

# Prototipo de componente software para generación dinámica de alertas de usuario basado en procesos de negocio



## Trabajo de Grado

BRAYAN STIVEN MAMIÁN BAMBAGÜÉ

Directora:

PhD. Sandra Milena Roa Martínez

Universidad del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Departamento de Sistemas  
Grupo de Investigación en Inteligencia Computacional  
Línea de Investigación: Información y Tecnología  
Popayán, Febrero de 2024

## Tabla de contenido

|  |     |
|--|-----|
| 1. Introducción  | 8   |
| 1.1. Justificación   | 10  |
| 1.2. Objetivos   | 10  |
| 1.3. Aportes   | 10  |
| 2. Marco conceptual  | 11  |
| 2.1. Tipos de alertas de usuario   | 11  |
| 2.2. Presentación de alertas de usuario                                  | 15  |
| 2.3. Implementación de alertas de usuario basadas en procesos de negocio | 19  |
| 3. Caracterización de alertas de usuario                                 | 29  |
| 3.1. Abstracción de procesos de negocio y alertas de usuario             | 29  |
| 3.2. Propuesta de patrones de alertas de usuario sobre patrones de flujo | 37  |
| 4. Componente software para la generación dinámica de alertas de usuario | 62  |
| 4.1. Análisis  | 62  |
| 4.2. Diseño  | 81  |
| 4.3. Construcción  | 84  |
| 5. Evaluación del componente de software                                 | 90  |
| 5.1. Selección del sistema objetivo                                      | 90  |
| 5.2. Abstracción del proceso de negocio de la evaluación                 | 90  |
| 5.3. Selección de usuarios evaluadores                                   | 93  |
| 5.4. Definición de encuestas sobre alertas                               | 94  |
| 5.5. Preparación de evaluación   | 95  |
| 5.6. Ejecución de evaluación   | 97  |
| 5.7. Resultados de la evaluación   | 97  |
| 6. Conclusiones y trabajo futuro   | 106 |
| 7. Bibliografía  | 108 |
| 8. Acta de propiedad intelectual   | 110 |
| Anexos   | 112 |

## Lista de figuras

|   |    |
|---|----|
| Figura 1. Mensaje de error del navegador Edge   | 12 |
| Figura 2. Mensaje de confirmación de Zotero   | 12 |
| Figura 3. Mensaje de confirmación sistema de archivos de Windows  | 13 |
| Figura 4. Mensaje de advertencia de Microsoft   | 13 |
| Figura 5. Mensajes de notificación de LinkedIn  | 14 |
| Figura 6. Feedback de exportación de datos en OpenAI  | 14 |
| Figura 7. Mensaje de instalación de nueva extensión en el navegador Microsoft Edge  | 14 |
| Figura 8. Mensaje informativo del gestor de referencias Zotero  | 15 |
| Figura 9. Mensaje de error del gestor de bases de datos DataGrip  | 16 |
| Figura 10. Mensaje en modo palabra simple en Gmail  | 16 |
| Figura 11. Mensaje de error en Facebook   | 17 |
| Figura 12. Mensaje de advertencia en repositorio Bitbucket  | 17 |
| Figura 13. Mensaje de confirmación de la plataforma SIMCA   | 18 |
| Figura 14. Mensaje de información gestión de cuenta Google  | 18 |
| Figura 15. Notación gráfica de tareas en BPMN   | 20 |
| Figura 16. Compuertas   | 21 |
| Figura 17. Notación gráfica de compuerta exclusiva  | 21 |
| Figura 18. Notación gráfica de compuerta paralela   | 22 |
| Figura 19. Notación gráfica de compuerta compleja   | 22 |
| Figura 20. Notación gráfica de eventos de inicio  | 22 |
| Figura 21. Notación gráfica de eventos de fin   | 23 |
| Figura 22. Patrón de secuencia (Weske, 2019)  | 23 |
| Figura 23. Patrón de secuencia en un flujo (Weske, 2019)  | 24 |
| Figura 24. Patrón de división paralela (Weske, 2019)  | 24 |
| Figura 25. Patrón de unión (Weske, 2019)  | 24 |
| Figura 26. Patrón de división exclusiva (Weske, 2019)   | 25 |
| Figura 27. Patrón de unión exclusiva (Weske, 2019)  | 25 |
| Figura 28. Patrón de división múltiple (Weske, 2019)  | 25 |
| Figura 29. Patrón de unión múltiple (Weske, 2019)   | 26 |
| Figura 30. Patrón de fusión múltiple (Weske, 2019)  | 26 |
| Figura 31. Patrón de ciclos arbitrarios (Weske, 2019)   | 26 |
| Figura 32. Distribución por ubicación en el proceso para alertas el patrón secuencia                                      | 40 |
| Figura 33. Distribución por tipo para alertas ubicadas en el fin del proceso en patrones de secuencia                     | 41 |
| Figura 34. Distribución por tipo para alertas ubicadas en actividades centrales en patrones de secuencia                  | 41 |
| Figura 35. Distribución por tipo para alertas presentadas en el patrón de división/unión paralela                         | 42 |
| Figura 36. Distribución por tipo para alertas presentadas en compuerta de división en patrones de división/unión paralela | 43 |
| Figura 37. Distribución por tipo para alertas mostradas en compuerta de unión en patrón de división/unión paralela        | 43 |
| Figura 38. Distribución por tipo para alertas presentadas en el fin del proceso en patrones de División/unión paralela    | 44 |

|   |    |
|---|----|
| Figura 39. Distribución por tipo para alertas presentadas en actividad paralela en patrones de división/unión paralela                              | 44 |
| Figura 40. Distribución por ubicación en el proceso para alertas del patrón división/unión exclusiva  | 45 |
| Figura 41. Distribución por tipo para alertas presentadas en compuerta de división o actividad control compuerta en patrón división/unión exclusiva | 46 |
| Figura 42. Distribución por tipo para alertas presentadas en actividad central en procesos de división/unión exclusiva                              | 46 |
| Figura 43. Distribución por tipo para alertas presentadas en el fin del proceso en patrones de división/unión exclusiva                             | 47 |
| Figura 44. Distribución por tipo para alertas presentadas en actividad final en patrones de división/unión exclusiva                                | 47 |
| Figura 45. Frecuencia de títulos de patrón 1  | 48 |
| Figura 46. Frecuencia de contenidos de patrón 1   | 49 |
| Figura 47. Frecuencias de representación visual de patrón 1   | 50 |
| Figura 48. Frecuencia de íconos de patrón 1   | 50 |
| Figura 49. Frecuencias de color para alertas de patrón 1  | 50 |
| Figura 50. Frecuencia de contenidos para alertas de patrón 2  | 51 |
| Figura 51. Frecuencia de títulos para alertas de patrón 2   | 52 |
| Figura 52. Frecuencia de acciones para alertas de patrón 2  | 52 |
| Figura 53. Frecuencia de tipo de alerta para alertas de patrón 2  | 53 |
| Figura 54. Frecuencia de colores para alertas de patrón 2   | 53 |
| Figura 55. Frecuencia de iconos para alertas de patrón 2  | 53 |
| Figura 56. Frecuencia de contenido para alertas de patrón 3   | 54 |
| Figura 57. Frecuencia de títulos de alertas de patrón 3   | 55 |
| Figura 58. Frecuencias de acciones de alertas de patrón 3   | 56 |
| Figura 59. Frecuencia de representación visual de alertas de patrón 3   | 56 |
| Figura 60. Frecuencia de iconos de alertas de patrón 3  | 56 |
| Figura 61. Frecuencia de color de alertas de patrón 3   | 57 |
| Figura 62. Frecuencia de contenido de alertas de patrón 4   | 58 |
| Figura 63. Frecuencia de título para alertas de patrón 4  | 59 |
| Figura 64. Frecuencia de acciones de alertas de tipo 4  | 59 |
| Figura 65. Frecuencia de representación visual de alertas de patrón 4   | 60 |
| Figura 66. Frecuencia de color en alertas de patrón 4   | 60 |
| Figura 67. Frecuencia de icono para alertas de patrón 4   | 60 |
| Figura 68. Proceso principal componente de alertas  | 63 |
| Figura 69. Proceso almacenar sistema objetivo   | 64 |
| Figura 70. Diagrama de alto nivel para subproceso almacenar datos de sistema objetivo   | 65 |
| Figura 71. Diagrama de proceso de alto nivel subproceso registrar instancia de proceso  | 66 |
| Figura 72. Proceso generar alerta de usuario  | 68 |
| Figura 73. Organización de tareas en tablero Kanban en Asana  | 81 |
| Figura 74. Diagrama de arquitectura del componente  | 82 |
| Figura 75. Diagrama de componentes  | 82 |
| Figura 76. Diagrama entidad - relación del componente   | 83 |
| Figura 77 Diagrama relacional del componente  | 84 |
| Figura 78. Implementación trait generador respuestas genérico   | 87 |
| Figura 79. Implementación trait validación de datos peticiones genérico   | 88 |
| Figura 80. Configuración de rutas del API REST  | 88 |



|   |     |
|---|-----|
| Figura 81. Implementación de casos de uso de dominio del componente                             | 89  |
| Figura 82. Diagrama de flujo proceso de trámite en ventanilla física®                           | 91  |
| Figura 83. Diagrama BPMN general equivalente para proceso trámite ventanilla física - Docxflow® | 91  |
| Figura 84. Diagrama BPMN subproceso Radicar documento   | 92  |
| Figura 85. Diagrama BPMN subproceso Repartir trámite  | 92  |
| Figura 86. Diagrama BPMN subproceso Responder trámite   | 92  |
| Figura 87. Diagrama BPMN subproceso Finalizar trámite   | 93  |
| Figura 88. Cuestionario para alerta de usuario sugerida sobre Docxflow®                         | 96  |
| Figura 89. Diagrama general de respuestas para pregunta #1                                      | 97  |
| Figura 90. Diagrama de respuestas para pregunta #1 por tipo de alerta                           | 98  |
| Figura 91. Diagrama general de respuestas para pregunta #2                                      | 98  |
| Figura 92. Diagrama de respuestas para pregunta #2 por tipo de alerta                           | 99  |
| Figura 93. Diagrama general de respuestas para pregunta #3                                      | 99  |
| Figura 94. Diagrama de respuestas para pregunta #3 por tipo de alerta                           | 100 |
| Figura 95. Diagrama general de respuestas para pregunta #4                                      | 100 |
| Figura 96. Diagrama de respuestas para pregunta #4 por tipo de alerta                           | 101 |
| Figura 97. Diagrama general de respuestas para pregunta #5                                      | 102 |
| Figura 98. Diagrama de respuestas para pregunta #5 por tipo de alerta                           | 102 |
| Figura 99. Diagrama general de respuestas para pregunta #6                                      | 103 |
| Figura 100. Diagrama de respuestas para pregunta #6 por tipo de alerta                          | 103 |
| Figura 101. Diagrama general de respuestas para pregunta #7                                     | 104 |
| Figura 102. Diagrama de respuestas para pregunta #7 por tipo de alerta                          | 104 |
| Figura 103. Diagrama general de respuestas para pregunta #8                                     | 105 |
| Figura 104. Diagrama de respuestas para pregunta #8 por tipo de alerta                          | 105 |

## Lista de tablas

|  |    |
|--|----|
| Tabla 1. Clasificación por dominios de aplicaciones  | 30 |
| Tabla 2. Aplicaciones seleccionadas por estrato  | 31 |
| Tabla 3. Abstracción de proceso y alertas de usuario, patrón de flujo secuencia - Google Meet  | 32 |
| Tabla 4. Abstracción de proceso y alertas de usuario, patrón de flujo división/unión paralela y división/unión exclusiva - Google Meet | 33 |
| Tabla 5. Alertas de confirmación en final del proceso con contenido relacionado al título  | 49 |
| Tabla 6. Equivalencia de contenido propuesta para alertas de confirmación en el fin del proceso  | 49 |
| Tabla 7. Contenido de alertas de advertencia o feedforward entre acciones  | 51 |
| Tabla 8. Contenido relacionado a accione esperada de alertas de patrón 3   | 54 |
| Tabla 9. Titulo relacionado a accione esperada de alertas de patrón 3  | 55 |
| Tabla 10. Contenido de alertas de patrón 4   | 58 |
| Tabla 11. Contenido de alertas de patrón 4   | 59 |
| Tabla 12. Atributos de respuesta estándar del componente   | 71 |
| Tabla 13. Validaciones sobre peticiones al API REST  | 72 |
| Tabla 14. Datos de entrada servicio almacenar sistema objetivo   | 73 |
| Tabla 15. Respuesta correcta de servicio almacenar sistema objetivo  | 73 |
| Tabla 16. Datos de entrada servicio obtener datos de sistema objetivo  | 73 |
| Tabla 17. Respuesta resultado correcto servicio obtener datos de sistema objetivo  | 73 |
| Tabla 18. Parámetros para el servicio almacenar proceso BPMN   | 74 |
| Tabla 19. Respuesta resultado correcto servicio almacenar datos proceso BPMN   | 75 |
| Tabla 20. Datos de entrada servicio obtener datos de proceso   | 76 |
| Tabla 21. Respuesta resultado correcto servicio obtener datos de sistema objetivo  | 76 |
| Tabla 22. Datos de entrada servicio registrar instancia de proceso   | 77 |
| Tabla 23. Respuesta correcta de servicio registrar instancia de proceso  | 77 |
| Tabla 24. Datos de entrada servicio obtener datos de instancia   | 77 |
| Tabla 25. Respuesta resultado correcto servicio obtener datos de instancia   | 78 |
| Tabla 26. Datos de entrada servicio registrar instancia de proceso   | 78 |
| Tabla 27. Respuesta correcta de servicio generar sugerencias de alertas de usuario para instancia                                      | 79 |
| Tabla 28. Definición de condiciones de patrones de flujo   | 80 |
| Tabla 29. Definición de alertas sugeridas en el componente   | 80 |
| Tabla 30. Datos de usuarios evaluadores del proceso  | 93 |

## Lista de anexos

|   |     |
|---|-----|
| Anexo 1. Abstracciones de procesos y alertas      | 112 |
| Anexo 2. Validaciones sobre requisitos            | 162 |
| Anexo 3. Inicialización de proyecto de desarrollo | 164 |
| Anexo 4. Implementación de módulos generales      | 167 |
| Anexo 5. Validación de requisitos                 | 175 |
| Anexo 6. Renderización manual de sugerencias      | 177 |

# 1. Introducción

El ciclo de vida del desarrollo de software se lleva a cabo por etapas o pasos genéricos que están presentes en la mayoría de proyectos independientemente de la metodología utilizada, estos pasos se pueden resumir en la determinación de la necesidad de una solución informática para asistir un proceso de negocio, la definición de los objetivos de la solución, recolección de requerimientos del negocio, conversión de requerimientos del negocio a requerimientos del sistema y la construcción e implementación de la solución. (Langer, 2008)

La introducción de cambios en los requerimientos del sistema es un factor constante en la mayoría de proyectos de desarrollo de software, dichos cambios pueden darse debido a cambios en los objetivos de negocio, objetivos de mercado, regulaciones de ley entre otros, (Nurmuliani, Zowghi, & Powell, 2004) y se pueden extender a lo largo de todo el ciclo de vida del desarrollo (Jayatilleke & Lai, 2018), de igual forma, las modificaciones realizadas sobre el software a partir de los cambios en los requerimientos, en algunos casos, deberán ser informadas o percibidas por el usuario, por ejemplo, mediante mensajes en la interfaz del software.

La gestión de los cambios a los requerimientos no tiene una manera única de abordarse, ya que estos dependen de muchos factores internos o externos, tales como el entorno de trabajo, o el impacto del cambio en distintos aspectos del proyecto como los costes, el tiempo o las limitaciones tecnológicas, sin embargo de manera general los cambios suelen requerir de una identificación, análisis y estimación de costes, (Jayatilleke & Lai, 2018) por lo que es necesario de la utilización de técnicas que faciliten este trabajo.

El modelado de procesos de negocio es una herramienta ampliamente utilizada por organizaciones para facilitar la comprensión de procesos de negocio, ya que permite representar de manera gráfica la secuencia de pasos de un proceso de negocio, sus condiciones y las bifurcaciones por las que puede atravesar durante su ejecución (Object Management Group, 2022). En el desarrollo de software los modelos de negocio pueden ser utilizados para aclarar los objetivos de negocio durante la fase de análisis de requerimientos, y también como base para construir la arquitectura del sistema y diseñar la interacción con el usuario en las fases de diseño e implementación.

Los procesos de negocio en la fase de diseño de interacción con el usuario permiten reconocer los posibles caminos que el usuario puede tomar durante la ejecución de un proceso e identificar las acciones que puede realizar en cada camino y el objetivo que acarrea cada una de estas, esto es importante dado que la interacción inicia con una intención del usuario de alcanzar un objetivo, dicha intención se transforma luego en una acción (presionar una tecla o hacer clic) que a su vez puede desencadenar cambios en el sistema cuyos resultados son presentados al usuario, quien se encarga de evaluarlos (Object Management Group, 2022). Una de las alternativas de presentación de resultados entre sistema y usuario son las alertas de usuario, las cuales usan distintos medios como audio, texto, imágenes, video, entre otros, para presentar un mensaje contextual al usuario, tienen el objetivo de retroalimentar sobre el estado de ejecución de procesos, advertir sobre los riesgos/consecuencias de realizar determinadas acciones, informar errores presentados, confirmar el éxito de una operación entre otros.

Ahora bien, según la norma ISO 25010, la usabilidad de un sistema se define como la capacidad de este para ser entendido, aprendido, usado y resultar atractivo para el usuario (ISO, 2022), en este sentido crear alertas de usuario cuyo contenido sea pertinente para el escenario de uso en donde se presenta, implícitamente conlleva a que el sistema se pueda considerar más usable.

Cuando el diseño de interacción de usuario está apoyado por un proceso de negocio el contenido de las alertas de usuario se puede inferir a partir de un análisis de los flujos del proceso, y en virtud de la capacidad del analista que realiza el trabajo, el contenido de las alertas generadas puede ser apropiado para el usuario en el contexto donde la alerta sea incluida, ya que el proceso de negocio permite saber previamente cuáles implicaciones pueden llegar a tener cada una de las acciones del usuario sobre el sistema.

En un ambiente de desarrollo de software tradicional, durante la fase de creación de requisitos, las alertas de usuario son incluidas como parte de los requerimientos funcionales, en donde se especifica su contenido, forma y el punto de la interacción con el usuario donde deberían mostrarse.

En la fase de desarrollo del proyecto, los requisitos pueden sufrir modificaciones, los cuales pueden llegar a afectar el modelo de negocio sobre el cual se basa el sistema; si esto sucede se deben analizar los cambios para ajustar el diseño de interacción del usuario y las alertas de usuario ya que el cambio en el modelo dependiendo de su impacto podría dejar obsoletas las definiciones de alertas de usuario generadas para la versión anterior del modelo de negocio.

Cuando el sistema se construye bajo una metodología de entrega continua, o se trata de un servicio que está en constante actualización, la incorporación y modificación de requisitos se convierte en un escenario común, y como consecuencia se debe realizar para cada cambio introducido el análisis del impacto sobre el proceso de negocio y la implementación de ajustes necesarios para mantener la pertinencia de las alertas de usuario dentro del sistema; Estas tareas se vuelven más costosas dependiendo del tamaño del proceso de negocio, a mayor tamaño mayor costo, ya que la introducción de pequeños cambios puede involucrar/modificar diversos puntos dentro del proceso que necesitan ajuste a nivel de interacción y alertas de usuario.

El análisis de cambios en los requisitos además de ser una tarea costosa es riesgosa, ya que al estar a cargo de un analista (humano) y teniendo en cuenta escenarios complejos, este puede omitir o malinterpretar algunos puntos del proceso y desencadenar implementaciones de alertas de usuario no deseadas que afecten la usabilidad del sistema.

Además, dado que la ejecución manual del proceso de definición de alertas de usuario genera un grado de incertidumbre sobre la pertinencia de las alertas generadas debido a la responsabilidad humana dentro de las actividades, bien sea por un mal análisis del proceso de negocio o por una implementación incorrecta (MacKenzie, 2013), y que esta incertidumbre en el peor de los casos puede terminar en una pérdida de usabilidad en el sistema causada por la una pérdida de capacidad para prevenir los errores de usuario, lo cual hace que

disminuir la incertidumbre en la ejecución de este proceso sea un aspecto importante a tener en cuenta.

A partir del contexto planteado surge la siguiente pregunta de investigación: **¿Cómo mantener la pertinencia de las alertas de usuario de un sistema frente a las modificaciones en el proceso de negocio?**

## 1.1. Justificación

Los procesos de negocio son deterministas, ya que se componen de un conjunto de tareas estructuradas y ordenadas que se relacionan entre sí para alcanzar un objetivo determinado (Object Management Group, 2022), dependiendo de los roles encargados de llevar a cabo las actividades definidas en el proceso, este puede ser catalogado como manual si requiere de intervención humana o automatizado si la ejecución manual se sustituye por tecnología y sistemas de datos

Una alternativa para mitigar lo planteado y que busca contribuir en mantener la pertinencia de las alertas generadas por el sistema ante los cambios en un determinado proceso de negocio es la automatización del proceso de definición de alertas de usuario, ya que mediante la sustitución de la participación humana dentro del proceso por un sistema informático se busca lograr una estandarización que permite estabilizar, monitorear y mejorar las alertas de usuario dentro del sistema, y adicionalmente, ofrecen como ventajas sobre los procesos manuales la reducción de costos operacionales, mayor rapidez en la ejecución y estandarización de actividades entre otros. (Visual Paradigm, 2022)

## 1.2. Objetivos

Objetivo general

- Proponer una solución informática que permita generar alertas para el usuario de manera dinámica a partir de cambios en el flujo de trabajo de un proceso de negocio.

Objetivos específicos

- Caracterizar los tipos de alertas de usuario que presentan los sistemas software basados en procesos de negocio.
- Desarrollar un componente de software que permita generar alertas de usuario de manera dinámica para un proceso de negocio a representar.
- Evaluar la correctitud de los mensajes de usuario generados luego de la implantación del componente en un proceso de negocio seleccionado.

## 1.3. Aportes

El principal aporte de este trabajo será la creación de un componente de software que permita la generación de alertas de usuario basándose en procesos de negocio, con el objetivo de contribuir a la industria de desarrollo de software, quienes mediante la integración del componente a sus desarrollos nuevos o existentes podrán generar alertas de usuario precisas que garanticen la correctitud de estas.

## 2. Marco conceptual

Dentro de las interfaces gráficas de usuario (GUI por sus siglas en inglés *Graphical User Interface*) existen distintos tipos de elementos de interacción, entre ellos se encuentran los *controls* que permiten ajustar aspectos de la aplicación en el contexto de la misma, por ejemplo elementos de ingreso de texto, numéricos, elementos de selección entre otros, que permiten ajustar parámetros para el objetivo del sistema (McKay, 2014), los *commands* que permiten al usuario indicarle al sistema que se debe ejecutar una acción determinada, por ejemplo los botones de acción en una barra de herramientas, botones contextuales, paletas de opciones entre otros (McKay, 2014) y las *interruptions* que son elementos de las GUI en las que se muestran mensajes al usuario debido a una situación ocurrida en el sistema que requiere la atención del usuario.

Todos los elementos de la interfaz de usuario tienen como características el *purpose*, *affordance*, *body language*, *interaction* y *labeling*. El *purpose* es el propósito del elemento, lo que se busca conseguir con la inclusión del mismo; El *affordance* son las pistas que indican cómo realizar la interacción con el elemento; El *body language* es la presentación visual que sugiere información adicional más allá de su propósito; La *interaction* es la acción específica que realiza el usuario con el elemento; y el *labeling* es el texto o íconos que explican el significado del efecto del elemento (McKay, 2014).

### 2.1. Tipos de alertas de usuario

Las *interruptions* se pueden clasificar según la naturaleza del origen del mensaje como mensajes de sistema o de instructivo (Galitz, 2007), y los mensajes instructivos a su vez se pueden subclasificar según el objetivo del mensaje como errores, advertencias, confirmaciones y notificaciones (McKay, 2014)

Los mensajes de sistema son generados con el objetivo de informar sobre el estado del sistema y las actividades que se ejecutan en él (Galitz, 2007), por otra parte, los mensajes instructivos que indican al usuario como completar o realizar tareas dentro del sistema (Galitz, 2007).

#### 2.1.1. Error

Las interrupciones de error permiten alertar al usuario sobre un problema que ha ocurrido (McKay, 2014) o situaciones en donde el usuario no puede continuar hacia su objetivo debido a alguna situación que lo impide, los mensajes deben contener información que permita al usuario a identificar la causa del problema, su posible solución y las acciones que le permitan continuar con sus tareas dentro del sistema (Podmajersky, 2019); Los mensajes de error son a menudo pueden presentarse por una implementación deficiente del sistema, por lo que es importante agregar siempre la información de ayuda para el usuario (Saffer, 2010). En la Figura 1 se observa un mensaje de error de conexión a internet en el navegador Edge, el que no permite cargar una página web, en este mensaje se agrega tanto la descripción como las acciones que puede tomar el usuario para poder superar el escenario erróneo.

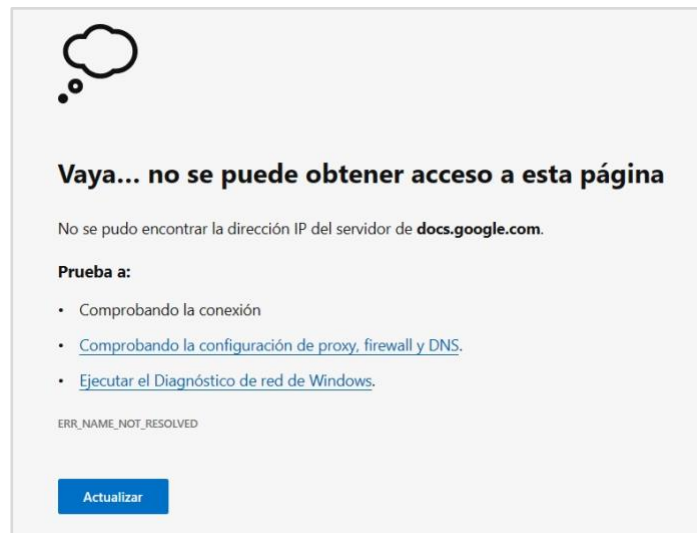


Figura 1. Mensaje de error del navegador Edge

### 2.1.2. Confirmación

Algunas tareas dentro de los sistemas pueden llevarse a cabo a lo largo de una cantidad de tiempo, los mensajes de confirmación permiten al usuario estar informado sobre los resultados que espera, si están completos o aún se encuentran en progreso (Podmajersky, 2019), las confirmaciones también se pueden presentar cuando se busca el consentimiento del usuario para realizar alguna acción que se considere crítica en el sistema (McKay, 2014). En la Figura 2 se muestra un mensaje de confirmación de un proceso culminado, puntualmente la instalación del programa Zotero, este mensaje busca informar que el proceso de instalación finalizó de manera satisfactoria. Por otra parte, en la Figura 3 se muestra un mensaje de confirmación solicitando consentimiento del usuario para completar la eliminación de un archivo.

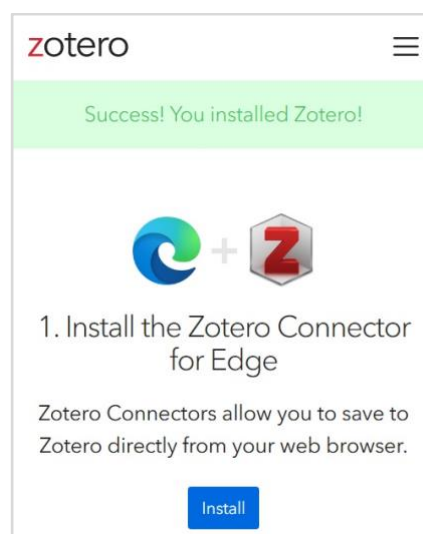


Figura 2. Mensaje de confirmación de Zotero



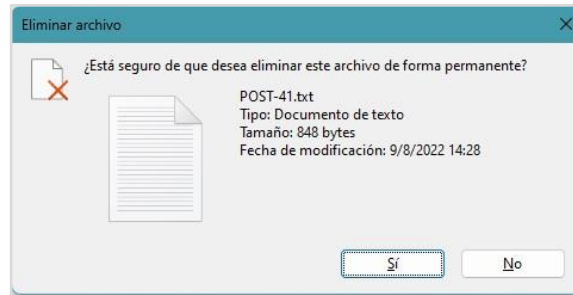


Figura 3. Mensaje de confirmación sistema de archivos de Windows

### 2.1.3. Advertencia

Alertan al usuario sobre condiciones especiales que pueden llegar a ocasionar problemas en el futuro, se diferencian de las confirmaciones porque no son preguntas hacia el usuario, únicamente describen la situación potencial y sus consecuencias (McKay, 2014). Como se puede apreciar en la Figura 4 en donde se advierte al usuario sobre lo que puede ocurrir en caso que no se agregue información de recuperación de la cuenta.



Figura 4. Mensaje de advertencia de Microsoft

### 2.1.4. Notificación

Las notificaciones son un tipo especial de mensajes que informan al usuario sobre procesos que se están ejecutando y que son ajenos a la actividad que el usuario realiza en el momento, su información puede ser de una severidad normal (McKay, 2014) o informar sobre una situación crítica que requiere la atención del usuario (Podmajersky, 2019). En la Figura 5 se puede observar un compendio de varias *interruptions* de tipo notificación las cuales se han ido agregando a lo largo del tiempo y que persisten en una vista aparte, dado que la información de las notificaciones no es contextual para la acción que el usuario está realizando en el sistema, este puede revisarlas cuando considere conveniente.

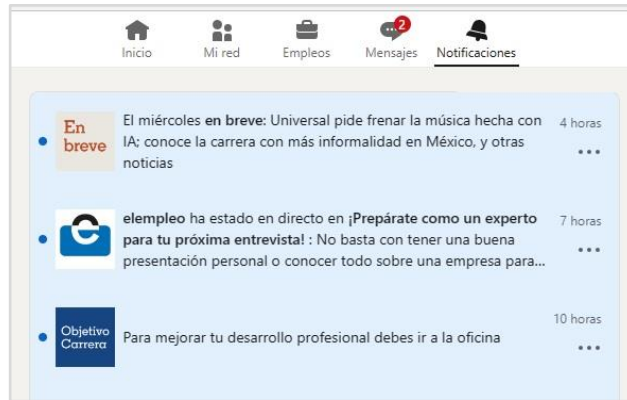


Figura 5. Mensajes de notificación de LinkedIn

### 2.1.5. Feedback

Los usuarios pueden realizar distintas acciones dentro de un sistema, si un mensaje se presenta luego que el usuario ha realizado una acción y su contenido está relacionado con dicha acción, se denomina mensaje de *feedback* (Fernández Casado, 2018), este mensaje se puede presentar independientemente si el resultado de la acción es correcto o se generó un error (Saffer, 2010). En la Figura 6 se muestra un mensaje de *feedback* informando que la solicitud de exporte de datos de la plataforma OpenAI fue exitosa y que se enviarán vía correo electrónico.

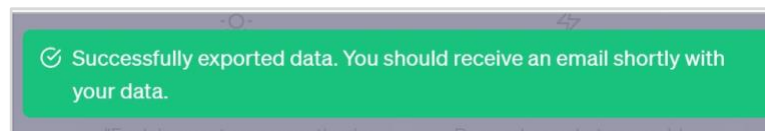


Figura 6. Feedback de exportación de datos en OpenAI

### 2.1.6. Feedforward

Al contrario de los mensajes de *feedback*, los mensajes de *Feedforward* se muestran antes que el usuario realice la acción y su objetivo es informar sobre las consecuencias que dicha acción puede llegar a tener sobre el sistema que usa (Fernández Casado, 2018), este tipo de mensajes permiten que el usuario pueda realizar acciones con confianza ya que les da idea de lo que sucederá a continuación (Saffer, 2010). En la Figura 7 se muestra el mensaje de *feedforward* para la instalación de una nueva extensión en el navegador Microsoft Edge, informando los permisos que la extensión tendrá sobre el navegador.

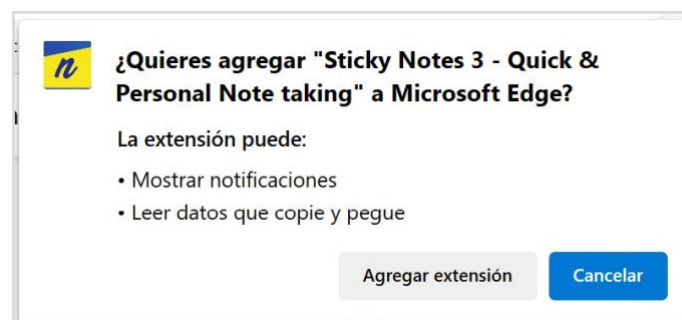


Figura 7. Mensaje de instalación de nueva extensión en el navegador Microsoft Edge

## 2.2. Presentación de alertas de usuario

El diseño visual abarca todas las características de comunicación visual, tales como el color, diseño, gráficos, fuentes entre otros, y los elementos visuales se usan para proporcionar un aspecto y sensación apropiados según el contexto (McIntire, 2008). Por otra parte, de manera general, la definición de diseño centrado en el usuario indica que el diseño de un sistema debe facilitarle al usuario el uso de este, de tal modo que sepa en todo momento dónde se encuentra y hacia dónde puede ir (Norman, 2013).

### 2.2.1. Representación visual

Para la presentación visual de las *interruptions* se combinan las características visuales y de diseño centrado en el usuario, ya que estos componentes tienen las características de *affordance*, *labeling*, *interaction* y *body language*. Visualmente las *interruptions* se pueden mostrar de varias formas.

#### 2.2.1.1. Caja de texto

Las cajas de texto se utilizan para mostrar información en un contexto específico, es una ventana emergente o pop-up con un título, un mensaje y algunos de los siguientes comandos según (Galitz, 2007):

- OK: Cuando el mensaje es de tipo informativo
- Cancelar: Cuando el usuario tiene la posibilidad de detener la acción
- Ayuda: Si el mensaje es informativo y se puede obtener una descripción más detallada del contenido
- Si / No: Si el usuario tiene que tomar una decisión
- Detener: Si el usuario puede detener una acción en progreso
- Botones contextuales de acuerdo con el tipo de mensaje

En la Figura 8 se muestra un mensaje de advertencia de la herramienta Zotero en modo caja de texto informando que la información introducida no arrojó ningún resultado, como el mensaje es informativo se muestra únicamente control OK.

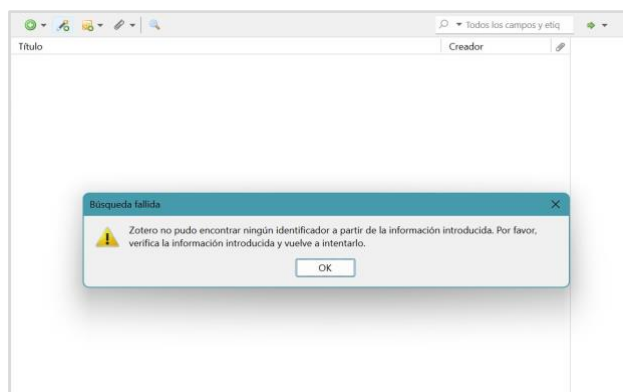


Figura 8. Mensaje informativo del gestor de referencias Zotero

Por otro lado, en la Figura 9 se muestra un mensaje de error arrojado por DataGrip el cual es generado por error en las credenciales de acceso a base de datos ingresada por el usuario.

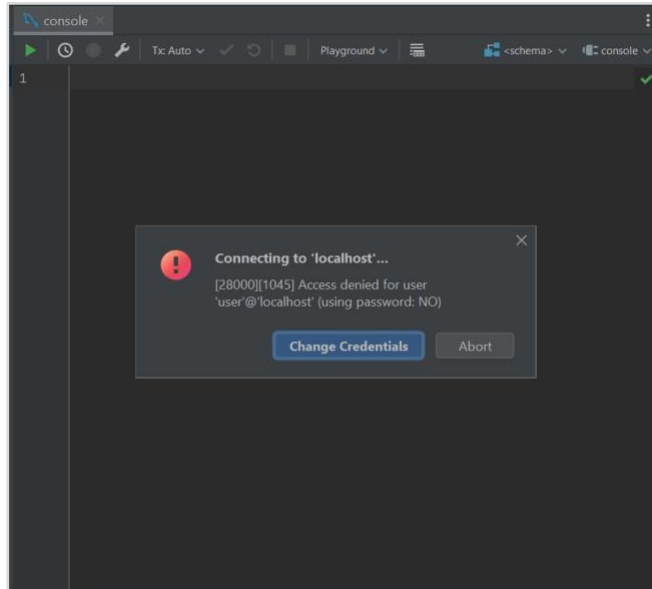


Figura 9. Mensaje de error del gestor de bases de datos DataGrip

### 2.2.1.2. Palabra simple

Cuando en el contexto de uso del usuario, el foco está concentrado en un punto determinado la *interruption* se puede presentar como una palabra simple (que puede estar acompañada de un ícono o imagen) anclada a un punto fijo de la GUI, con el objetivo de no desviar el foco del usuario en su actividad actual. (Podmajersky, 2019). En la Figura 10 se muestra un ejemplo, en donde el contexto es la creación de un correo electrónico mediante el uso de Gmail, el usuario tiene el foco en la edición del mensaje, y cada vez que modifica el texto se guarda un borrador, el mensaje de éxito de guardado se muestra en la parte superior, a la izquierda el estado inicial y a la derecha el mensaje de confirmación en modo de *palabra simple*.

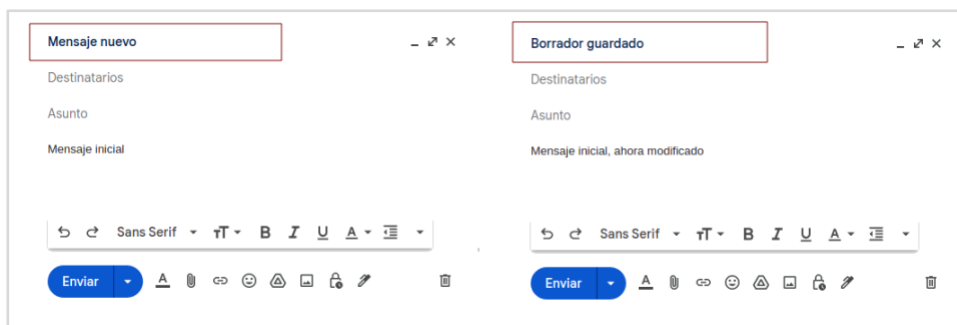


Figura 10. Mensaje en modo palabra simple en Gmail

## 2.2.2. Colores

La elección de colores adecuados puede cambiar por completo la experiencia del usuario en el uso de un sistema, ya que como humanos asociamos los colores a emociones, estados de ánimo o sentimientos (Hogan, 2009), un color usado de manera inadecuada puede causar confusión al usuario (Galitz, 2007).

### 2.2.2.1. Rojo

El rojo es asociado generalmente con las acciones/emociones de *Detenerse*, *Peligro*, *Emergencia* (Galitz, 2007) (Hogan, 2009) y puede ser usado para representar mensajes de

advertencia o errores ya que captan de inmediato la atención del usuario (Hogan, 2009). En la Figura 11 se muestra un ejemplo de error en color rojo de la plataforma Facebook, donde se resalta el mensaje sobre el resto de la interfaz haciendo énfasis no solo en el mensaje sino también en el campo que origina la validación incorrecta, en este caso el correo electrónico.



Figura 11. Mensaje de error en Facebook

#### 2.2.2.2. Amarillo

Las acciones/emociones relacionadas al color amarillo son *validación, ir despacio, precaución* (Galitz, 2007), y pueden usarse para mostrar mensajes de advertencia hacia el usuario. En la Figura 12 se muestra un mensaje de advertencia en el repositorio Bitbucket para informar que se quitará el acceso a un usuario sobre un repositorio, el amarillo en este caso es usado con la intención de realizar con cuidado la acción.

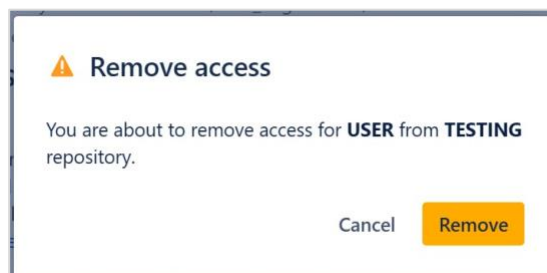


Figura 12. Mensaje de advertencia en repositorio Bitbucket

#### 2.2.2.3. Verde

Las acciones/emociones relacionadas al color verde son *seguridad, continuar, confirmación* (Galitz, 2007), se puede usar este color para mensajes de confirmación de acciones exitosas. En la Figura 13 se muestra un mensaje de confirmación de cambio de contraseña para la plataforma SIMCA, el verde en este caso refuerza el mensaje de éxito en la operación.

|  |                      |
|--|----------------------|
| Contraseña actual  | <input type="text"/> |
| Contraseña nueva *   | <input type="text"/> |
| Confirmar Contraseña *   | <input type="text"/> |
| <div style="background-color: #4CAF50; color: white; padding: 5px; border-radius: 5px;"> <p>La contraseña se cambió correctamente</p> </div> |                      |

Figura 13. Mensaje de confirmación de la plataforma SIMCA

#### 2.2.2.4. Azul

Las acciones/emociones relacionadas al color azul son *neutralidad, calma* (Galitz, 2007), se puede usar este color en mensajes informativos que no supongan ningún tipo de acción o resultado positivo o negativo para el usuario. En la Figura 14 se muestran íconos de información en color azul para las opciones de cuenta de Google, lo cual indica que no es obligatorio tenerlas activas pero que de momento se encuentran activas y pueden desactivarse en cualquier momento.



|   |  |  |
|---|--|--|
|  | Información de contacto guardada a partir de las interacciones | <input checked="" type="checkbox"/> Activada |
|  | Información de los contactos de tus dispositivos               | <input checked="" type="checkbox"/> Activada |

Figura 14. Mensaje de información gestión de cuenta Google

#### 2.2.3. Escritura

El contenido textual de las alertas de usuario debe ser claro, conciso y no contener lenguaje técnico que confunda al usuario en la interacción con las mismas (Norman, 2013). A continuación, se mencionan algunos aspectos para tener en cuenta al momento de escribir alertas de usuario:

- Para los mensajes de instrucción utilizar primero el verbo en infinitivo y luego las instrucciones, el contenido del mensaje debe ser consecuente con el título (si lo hay) (Podmajersky, 2019)
- Para los mensajes de confirmación usar el verbo de la acción realizada en pasado o una frase que describa la acción (Podmajersky, 2019)
- Para los mensajes de error agregar descripción adicional de la causa del error, con el objetivo que el usuario identifique o prevenga el mismo error en un futuro (Podmajersky, 2019)
- Para las notificaciones agregar un título que describa brevemente la acción que el usuario debe tomar en cuenta y un cuerpo con el mensaje completo de la acción descrita en el título (Podmajersky, 2019)

## 2.3. Implementación de alertas de usuario basadas en procesos de negocio

La automatización total o parcial de procesos en las empresas da como resultado la reducción de costos y mejora en eficiencia de las actividades empresariales (Aalst, Ter Hofstede, & Weske, 2003), es por esto que algunos sistemas se desarrollan con el objetivo de automatizar dichos procesos empresariales, los sistemas construidos con este objetivo de manera implícita trabajan bajo el proceso de negocio que automatizan según el grado de apoyo que el sistema brinde sobre la automatización del proceso empresarial (Weske, 2019). No obstante, muchos sistemas que fueron desarrollados para automatizar o semi automatizar un proceso de negocio empresarial, no necesariamente cuentan con la representación explícita del flujo de trabajo que representan, por lo que es necesario utilizar técnicas para poder abstraer esta información (Yan, Dijkman, & Grefen, 2012).

Un flujo de trabajo (*workflow*) se define como la automatización de un proceso de negocio bien sea total o parcialmente, en el cual se definen las tareas y las reglas para llevar a cabo determinado procedimiento (Weske, 2019), dicha definición consta en primer lugar de *actividades o tareas*, que describen los pasos requeridos para completar el flujo de trabajo. Estas actividades son ejecutadas por *roles* definidos en el contexto de la organización y su orden se especifica en un *modelo o esquema* conectándose unas a otras a través de *transiciones*, formando de este modo un hilo de ejecución. Finalmente, cada caso específico o ejecución del modelo se denomina una *instancia de flujo de trabajo*, que representa la aplicación real del flujo de trabajo en un contexto particular (Van der Aalst, Ter Hofstede, Kiepuszewski, & Barros, 2003).

Con el objetivo de administrar correctamente la automatización de procesos empresariales, los flujos de trabajo deben ser descritos y documentados (Allweyer, 2016). Una alternativa sencilla es usar descripciones textuales o diagramas de flujo compuestos por cuadros y flechas. Sin embargo, la desventaja de mapear modelos mediante diagramas que no siguen un método definido es que estos no cumplen con representación exacta para procesos extensos y todos los aspectos relevantes inherentes al mismo, como reglas de bifurcación, eventos, unidades organizacionales o flujos de datos, por lo que se requiere una notación estándar en el modelado gráfico con un significado bien definido (Allweyer, 2016). Uno de los beneficios de utilizar un estándar de modelado es que este no se limita a ser usado por una herramienta en particular, ya que la semántica la define el estándar y no la herramienta, por otra parte, cada uno de los elementos que compone el estándar tiene un significado lo suficientemente preciso como para poder describir los detalles técnicos que lo controlan (Silver, 2011).

En los últimos años BPMN (*Business Process Model and Notation*) ha sido aceptado como el estándar líder para el modelado de procesos empresariales, (Allweyer, 2016) esto dado que es un estándar mantenido por el *Object Management Group (OMG)*, lo cual significa que no está controlado por algún proveedor de herramientas individuales y no requiere de pagos por regalías o propiedad intelectual (Silver, 2011)

### 2.3.1. Estándar de modelado BPMN (Business Process Model and Notation)

BPMN en español significa *Modelo y notación de procesos de negocio*, nombre en el cual la parte más representativa es la N que implica una notación o lenguaje para representar de forma gráfica flujos empresariales (Silver, 2011), este estándar es desarrollado y mantenido por el *Object Management Group* y su primera versión (1.0) fue lanzada en 2006, luego de la cual se han realizado algunas actualizaciones (Allweyer, 2016) siendo la actual versión la 2.0.2 lanzada en el año 2014 (Object Management Group, 2023).

Los conceptos principales en BPMN son *actividad y proceso*, una actividad se entiende como una acción o unidad de trabajo que tiene un significado, un inicio y final bien definido y es ejecutada por un actor (*performer*) de manera repetitiva en el transcurso de un proceso de negocio. Cada ejecución de la actividad representa la misma acción en una pieza de trabajo diferente (Silver, 2011).

Por otra parte, un proceso es una secuencia de actividades que van desde un estado inicial (relacionado a un evento como una petición) hacia un estado final. Un modelo de proceso es un mapeo de todos los caminos posibles o secuencias de actividades desde el estado inicial hacia algún estado final, éxito o error. Al igual que la actividad el proceso tiene un inicio y final bien definidos y pueden ser ejecutados de manera repetida en el transcurso del proceso de negocio y cada ejecución sigue un camino en el modelo de proceso desde el inicio hasta el final. (Silver, 2011).

Dentro de la definición de BPMN existen distintos elementos que representan las diferentes etapas y conexiones de un proceso empresarial.

#### 2.3.1.1. Actividades

Como se mencionó anteriormente, las actividades representan una unidad de trabajo o acción específica, cada actividad se puede clasificar como una *tarea* o como un *subproceso*, una tarea es atómica, esto quiere decir que no existe una descripción más detallada de la misma dentro del modelo, por otra parte, un subproceso indica que existe una descripción más detallada dentro del proceso llamada *modelo hijo*; Una actividad inicia en un estado inicial y puede finalizar en uno o varios estados de fin. En la notación gráfica BPMN estas son representadas mediante un rectángulo con los bordes redondeados (Silver, 2011)

##### 2.3.1.1.1. Tareas

Como se muestra en la Figura 15, las tareas son representadas en BPMN con el rectángulo de bordes redondeados asociado a la actividad y adicional un ícono en la parte superior izquierda que denota el actor que ejecuta la actividad, el cual puede ser el usuario, un proceso automático o un proceso abstracto (actor no definido) (Silver, 2011)

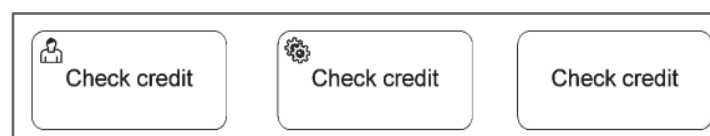


Figura 15. Notación gráfica de tareas en BPMN



### 2.3.1.1.2. Subprocesos

Los subprocesos están compuestos por actividades las cuales pueden ser descritas mediante un proceso hijo, de forma gráfica las actividades de un subproceso pueden ser representado de manera colapsada agregando un símbolo [+] en la parte inferior del recuadro y generando la definición en otro modelo, o ser representado alineado junto a las demás actividades del proceso principal.

### 2.3.1.2. Compuertas

Las compuertas son usadas para separar o unir secuencias de flujo en caminos alternativos y son representadas mediante el símbolo del rombo y distintas salidas como se observa en la Figura 16; Existen diferentes tipos de compuertas que permiten modelar distintos escenarios (Allweyer, 2016) cada una de las cuales son diferenciadas por un símbolo en la notación (Silver, 2011). Una compuerta puede tener múltiples entradas y múltiples salidas (aunque en la representación gráfica no se recomienda agregar demasiadas a una misma compuerta dado que puede dar pie a malinterpretaciones del modelo) (Allweyer, 2016).

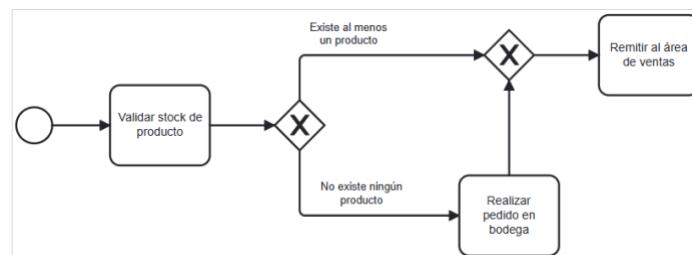


Figura 16. Compuertas

#### 2.3.1.2.1. Compuerta exclusiva

Es usada para representar caminos alternativos, tanto para separarlos como para unirlos; 'Exclusiva' significa que el flujo únicamente continuará por una de las salidas de la compuerta (Silver, 2011). En la notación una compuerta exclusiva se puede representar con un rombo con una equis en su interior o como un rombo vacío (Allweyer, 2016) como se muestra en la Figura 17.

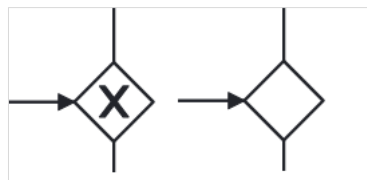


Figura 17. Notación gráfica de compuerta exclusiva

#### 2.3.1.2.2. Compuerta paralela

Se usan para representar un flujo que en determinado punto se divide en caminos paralelos, es decir que aunque tiene salidas distintas todos pueden ser ejecutados al tiempo sin ninguna condición (Allweyer, 2016) (Silver, 2011), como se muestra en la Figura 18 la compuerta paralela es representada gráficamente con el símbolo de cruz; La compuerta en donde un flujo paralelo se une en un mismo camino también debe ser representado con la notación gráfica de puerta paralela

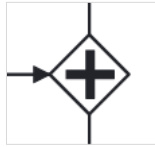


Figura 18. Notación gráfica de compuerta paralela

### 2.3.1.2.3. Compuerta compleja

En algunos casos la lógica para la unión o separación de caminos mediante compuertas puede no tener una manera directa de ser expresada con los elementos de BPMN, en cuyo caso se puede usar las compuertas complejas (Allweyer, 2016), las cuales permiten definir comportamientos adaptados, esto mediante una anotación donde se especifique el comportamiento requerido para la compuerta. Las compuertas como se muestra en la Figura 19 son representadas mediante un asterisco.

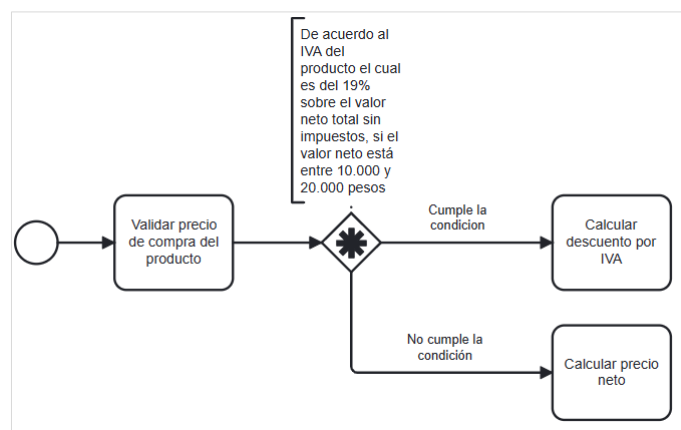


Figura 19. Notación gráfica de compuerta compleja

### 2.3.1.3. Eventos

Son usados para representar que algo ha sucedido en algún punto de la ejecución, a diferencia de una actividad no tienen un tiempo de duración y deben tomarse en cuenta dos aspectos, cuál es la causa que lanza el evento y cuál es el efecto que este tiene sobre el proceso (Allweyer, 2016).

#### 2.3.1.3.1. Evento de inicio

Es representado en la notación como un círculo con el borde delgado, y su propósito es indicar cómo inicia un proceso o subproceso, mediante un ícono en su interior se representa la causa de lanzamiento del evento, el cual puede ser vacío (indefinido), mensaje (que es lanzado luego de una petición externa al proceso y sus participantes), temporizador (que es determinado por una periodicidad en el tiempo) (Silver, 2011). La notación de estos eventos se puede apreciar en la Figura 20.



Figura 20. Notación gráfica de eventos de inicio

### 2.3.1.3.2. Evento de fin

Es representado en la notación como un círculo con borde grueso, estos estados indican el fin de la ejecución de un camino y pueden también tener íconos en su interior para representar escenarios de fin como por ejemplo sin ícono (significa un estado de fin normal), con un círculo en su interior (indica que el proceso finaliza independientemente si algún proceso paralelo se encuentra aún en ejecución), mensaje (indica que el proceso finaliza lanzando una alerta hacia otro proceso) (Silver, 2011). La notación gráfica de estos estados se puede observar en la Figura 21



Figura 21. Notación gráfica de eventos de fin

### 2.3.2. Patrones de flujo de trabajo

Con el objetivo de analizar cómo actualmente se implementan las alertas de usuario en software que están creados o que siguen un proceso de negocio automatizado o *workflow*, a continuación, se realizará la definición de patrones genéricos de flujo de trabajo, con el objetivo de validar cómo estos se relacionan con la implementación actual de alertas de usuario en software conocidos.

De manera general un patrón puede considerarse como “una abstracción de una forma concreta que se repite en contextos específicos no arbitrarios” (Riehle & Züllighoven, 1996), aplicados a un flujo de trabajo, los patrones se pueden usar como un criterio para expresar procesos de negocio sin que estos dependan del lenguaje de proceso o notación, usando conceptos básicos del control del flujo (Weske, 2019). Los patrones de flujo se definen a nivel del *modelo* y la semántica de su ejecución se aplica a las *instancias* del mismo (Weske, 2019).

#### 2.3.2.1. Secuencia

El patrón de secuencia indica que, dentro de una *instancia* de un proceso, una *actividad* se habilita o se inicia inmediatamente después que se una *actividad* anterior se finaliza, este patrón incluye el escenario en donde las actividades secuenciales pueden ser parte de un bucle, en el cual se tendrán varias *instancias de actividad* en la misma *instancia de proceso* como se muestra en la Figura 22 y Figura 23 (Weske, 2019).

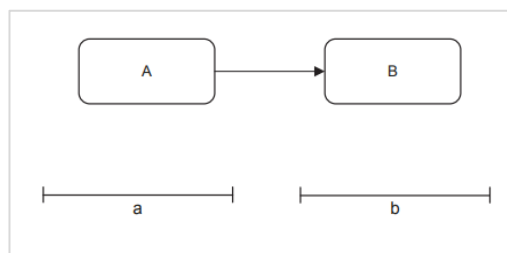


Figura 22. Patrón de secuencia (Weske, 2019)

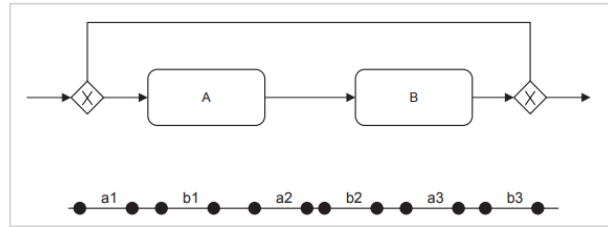


Figura 23. Patrón de secuencia en un flujo (Weske, 2019)

### 2.3.2.2. División paralela (*and Split*)

Este patrón indica una división o un punto en el modelo en donde un hilo de flujo se divide en múltiples hilos que se ejecutan de manera simultánea como se muestra en la Figura 24 (Weske, 2019).

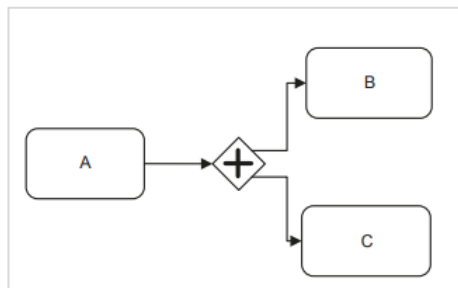


Figura 24. Patrón de división paralela (Weske, 2019)

### 2.3.2.3. Unión paralela (*and Join*)

Este patrón hace referencia a un punto en el proceso en donde procesos que se están ejecutando de manera concurrente, convergen en un hilo único de ejecución partiendo del supuesto sobre que cada proceso concurrente se ejecuta una única vez como se muestra en la Figura 25 (Weske, 2019).

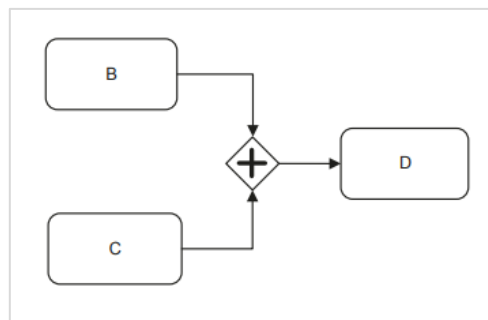


Figura 25. Patrón de unión (Weske, 2019)

### 2.3.2.4. División exclusiva (*xor Split*)

Este patrón indica que existe un punto en el modelo en el cual el proceso continuará su ejecución por un hilo de un conjunto de ramas disponibles como se muestra en la Figura 26 (Weske, 2019).

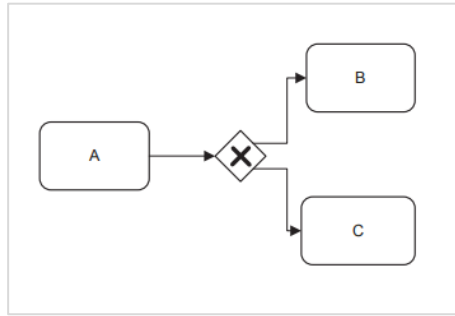


Figura 26. Patrón de división exclusiva (Weske, 2019)

### 2.3.2.5. Unión exclusiva (xor Join)

Este patrón indica que existe un punto en el modelo en el cual varios hilos de ejecución convergen en un único hilo sin realizar sincronización, es decir continuando con el proceso en cuanto la primera rama finalice su ejecución, asumiendo que sólo una de las ramas entrantes se ejecutó como se muestra en la Figura 27 (Weske, 2019).

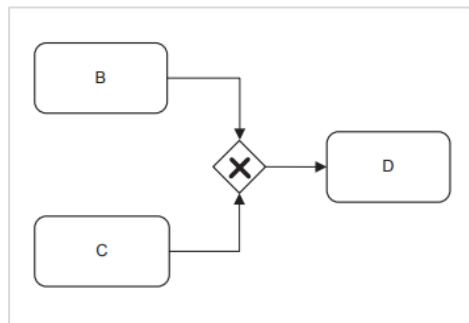


Figura 27. Patrón de unión exclusiva (Weske, 2019)

### 2.3.2.6. División condicional (or Split)

Este patrón indica que existe un punto en el modelo en el cual el proceso continuará su ejecución por uno o varios hilos de un conjunto de ramas disponibles como se muestra en la Figura 28 (Weske, 2019).

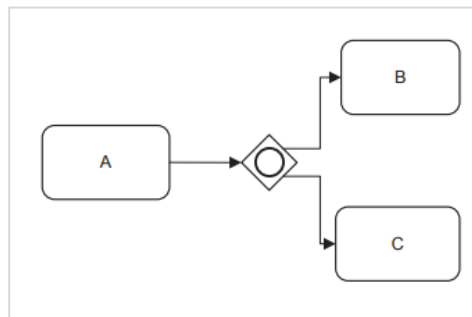


Figura 28. Patrón de división múltiple (Weske, 2019)

### 2.3.2.7. Unión condicional (or Join)

Este patrón indica que existe un punto en el modelo en el cual la ejecución converge en un único hilo en este caso realizando sincronización, es decir que la continuación del proceso se pausará hasta que todas las ramas que se están ejecutando finalicen, asumiendo que una

rama que ya finalizó su ejecución no puede volver a ejecutarse hasta que la sincronización finalice como se muestra en la Figura 29 (Weske, 2019).

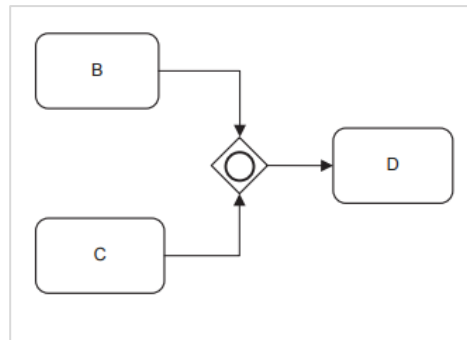


Figura 29. Patrón de unión múltiple (Weske, 2019)

#### 2.3.2.8. Fusión múltiple (multiple merge)

Este patrón indica que existe un punto en el modelo en el cual la ejecución converge en un único hilo y continúa la ejecución para cada finalización de la o las ramas entrantes, es equivalente a la unión exclusiva, salvo que en este caso no se asume una única rama entrante en ejecución sino que pueden múltiples ramas de ejecución, cada una de las cuales desencadenará un evento cuando finalice como se muestra en la Figura 30 (Weske, 2019).

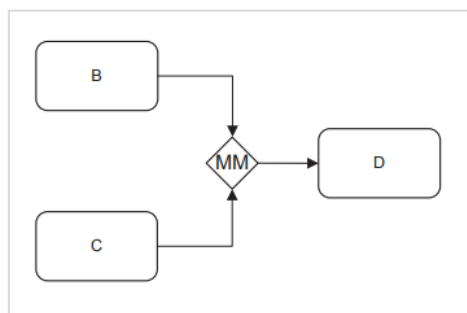


Figura 30. Patrón de fusión múltiple (Weske, 2019)

#### 2.3.2.9. Ciclos arbitrarios (arbitrary cycles)

Este patrón indica que existe un punto en el modelo en el cual varias tareas pueden ser ejecutadas de manera repetida como se muestra en la Figura 31 (Weske, 2019).

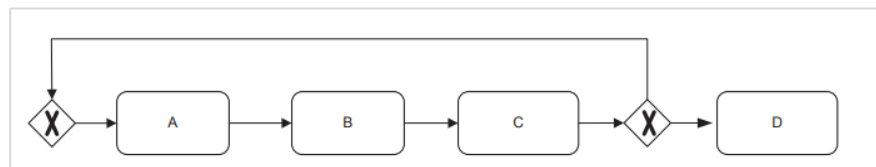


Figura 31. Patrón de ciclos arbitrarios (Weske, 2019)

### 2.3.3. El método de Silver

Con el objetivo de analizar los patrones definidos anteriormente sobre software actuales, y, teniendo en cuenta que, en la mayoría de los sistemas actuales, no es posible acceder de manera directa al proceso de negocio que el sistema automatiza es necesario definir un método que permita realizar la abstracción del proceso de negocio para su análisis posterior.

*El método (The method)* propuesto por (Silver, 2011) es una forma de generar un modelo BPMN basándose en un modelo jerárquico sobre los aspectos principales de un proceso, desde un alto nivel y hacia un nivel más específico, permitiendo añadir tantos detalles como sea necesario. Gráficamente representa el modelo desde el inicio y hasta el final como una serie de procesos relacionados representando los distintos niveles de detalle. Así, el diagrama principal, puede contener procesos sintetizados, cada uno de los cuales puede ser expandido en un diagrama “hijo” el cual a su vez puede contener también procesos sintetizados los cuales pueden ser expandidos en otro diagrama “hijo”, en una relación de “nieto” con el proceso principal. La cantidad de niveles en la jerarquía no tiene límite. Cada uno de los diagramas generados no son modelos separados, sino que constituyen una vista separada de un mismo modelo semántico (Silver, 2011). Lo anterior con el objetivo de entender el proceso como un todo, identificando los pasos principales en la cima de la jerarquía y definir a partir de estos el nivel de detalle requerido según el propósito de la creación del diagrama (Silver, 2011).

#### 2.3.3.1. Definición del alcance del proceso

Para iniciar con el método, es necesario determinar cuál es el alcance del proceso, dónde empieza y dónde termina, teniendo en cuenta las siguientes interrogantes.

- ¿Cómo inicia el proceso?, mediante una petición, mediante una entidad externa, iniciado mediante un ejecutor o es una tarea programada.
- ¿Qué determina que está completo?, completo significa que una vez en ese estado no se pueden realizar más acciones dentro del proceso o que hay acciones derivadas pero que se abordan dentro de otro proceso.
- ¿Qué representa cada instancia o ejecución del proceso?, dentro del negocio que significa cada ejecución del proceso.
- ¿Cuáles son las formas en las que el proceso puede terminar?, ¿tiene uno o más de un estado de finalización?

#### 2.3.3.2. Mapeo de alto nivel

En esta etapa se busca enumerar y describir las actividades principales en un alto nivel, cada una de ellas alineadas con la ejecución del proceso de manera lineal. En este punto las actividades deben considerarse como contenedores cuyos detalles serán agregados más adelante teniendo en cuenta todos los posibles escenarios finales para cada actividad.

#### 2.3.3.3. Diagrama de proceso de alto nivel

Con la información derivada de los puntos anteriores, se construye un diagrama BPMN siguiendo las siguientes consideraciones:

- Se agrega el evento de inicio de acuerdo con su tipo.
- Se agrega cada una de las actividades encadenadas al evento de inicio.
- Cada actividad encadenada se agrega como un subprocesso, el cual será definido de manera independiente en las siguientes etapas.
- Cada actividad estará encadenada a una puerta (gateway) de la cual desprende cada una de las opciones para sus escenarios finales.
- Si una actividad es ejecutada de manera concurrente con otras, se divide el flujo en dos rutas distintas usando una puerta paralela o dos flujos desde la actividad anterior.

- Si una actividad posterior requiere que se completen dos o más actividades principales, se debe usar una puerta de unión.

#### 2.3.3.4. Expansión de detalle

El diagrama de alto nivel indica como el proceso inicia y termina, pero no define los detalles internos. Para tal fin se debe realizar un modelo adicional relacionado con el subproceso sintetizado en el diagrama de alto nivel siguiendo las siguientes consideraciones:

- Cada subproceso debe tener su propio diagrama asociado.
- Los diagramas no deben tener un evento de inicio.
- Los estados de finalización deben coincidir con los agregados en el diagrama de alto nivel.
- La expansión puede incluir otros subprocesos sintetizados que pueden ser ampliados en otras iteraciones como diagramas separados

#### 2.3.3.5. Agregar flujos de mensaje

Los flujos de mensaje no son obligatorios en la definición de BPMN, aunque aportan contexto valioso al proceso, representan desorden a nivel visual debido a que deben replicarse tanto en los procesos hijos como en el proceso de alto-nivel en la jerarquía.

#### 2.3.3.6. Iteración recursiva

Con el fin de añadir el detalle requerido, repetir los pasos *expansión del detalle* y *agregar flujos de mensaje* las veces que sea necesario.



## 3. Caracterización de alertas de usuario

En esta sección se presenta un análisis de las alertas de usuario generadas en diversas aplicaciones y dentro de distintos dominios. El propósito será identificar aquellas alertas asociadas a procesos que se ajusten a algunos de los patrones de flujo definidos. Posteriormente se analizarán estas alertas con la finalidad de buscar posibles relaciones o características comunes entre estas alertas en relación con los patrones seleccionados, definiendo de este modo *patrones de alertas de usuario*, los cuales serán usados como base en la construcción del prototipo de componente encargado de la generación dinámica de alertas.

### 3.1. Abstracción de procesos de negocio y alertas de usuario

En esta sección se realizará la definición y ejecución de la abstracción de patrones a realizar, seleccionando los patrones de flujo objetivo, definiendo las reglas que se tomarán en cuenta para el proceso de abstracción y seleccionando las aplicaciones sobre las que se abstraerán los patrones.

#### 3.1.1. Selección de patrones objetivo

Para la caracterización de alertas de usuario, de los patrones de flujo definidos previamente, los patrones buscados serán aquellos que incluyan solamente los elementos básicos de un flujo BPMN, los cuales son *actividades, compuertas paralelas y exclusivas y eventos de inicio y final*. En este contexto, los patrones que cumplen con la condición son:

- Secuencia
- División paralela
- Unión paralela
- División exclusiva
- Unión exclusiva
- Ciclos arbitrarios

El patrón de *ciclos arbitrarios*, aunque cumple con la condición de inclusión de solo elementos básicos no se incluirá como patrón objetivo, ya que este patrón por su definición está asociado al patrón de secuencia y se puede incluir dentro de este. Adicionalmente teniendo en cuenta que los patrones de unión tanto exclusiva como paralela requieren que previamente se haya incluido en el proceso el patrón de división asociado, estos patrones serán abstraídos de manera conjunta en los procesos. En consecuencia, los patrones que se buscarán serán:

- Secuencia
- División/unión paralela
- División/unión exclusiva

#### 3.1.2. Reglas para abstracción de procesos

Con la finalidad que los procesos abstraídos cuenten con la calidad y granularidad requerida para poder realizar el análisis subsiguiente, se definen las siguientes reglas que deberán cumplir las abstracciones realizadas:

- La selección de los sistemas que servirán como base para la abstracción de procesos deberá incluir varios dominios (conceptuales). Con esto se busca que el análisis sea más general y no sesgado por un dominio en particular.
- Los procesos elegidos deberán ser parte del dominio asociado al sistema elegido, ignorando de esta forma los procesos que estén relacionados con temas transversales, como procesos de configuración, manejo de sesiones y seguridad.
- Dentro de los procesos mapeados se ignorarán los flujos de cancelación voluntaria del proceso por parte del usuario, ya que estos no aportan valor al objetivo de la abstracción
- Los flujos de excepción asociados a problemas en la infraestructura o conexión con la aplicación tampoco se mapearán, por ejemplo, problemas con la conexión a internet o errores generados en la aplicación.

### 3.1.3. Selección de aplicaciones objetivo

Actualmente existe una gran diversidad de aplicaciones disponibles, Solo en el caso de aplicaciones móviles para el sistema operativo Android, la tienda Google Play Store superó la cifra de 2.5M de aplicaciones disponibles para 2023 (Statista, s. f.-b), y de manera similar en la App Store que es la plataforma de descarga de aplicaciones para dispositivos con iOS, la cantidad de aplicaciones disponibles superó las 1.6M para 2022 (Statista, s. f.-a), esto sin contar con las aplicaciones que están disponibles desde la web. Dicho lo anterior, en esta etapa es necesario definir una técnica de muestreo que permita obtener una muestra lo más representativa posible del universo de aplicaciones que existe actualmente.

Para la selección de las aplicaciones objetivo, se utilizó la técnica de muestreo estadístico denominada 'Estratificación', la cual permite dividir el total en segmentos (también llamados estratos) y luego seleccionar muestras de cada uno de los segmentos. Cada segmento está denominado con la notación  $N(h)$  y cada muestra del segmento por la notación  $n(h)$ , donde  $h$  es el número de segmento. (Cochran, 1977). En la elección de aplicaciones se utilizó la estratificación por categoría o dominio, en la cual se tomaron las principales categorías en las que están clasificadas las aplicaciones (Kemp, 2023) segmentando el total en 18 estratos como se muestra en la Tabla 1.

Tabla 1. Clasificación por dominios de aplicaciones

| Estrato | Nombre de la categoría             |
|---------|------------------------------------|
| N1      | Chat y Mensajería                  |
| N2      | Redes sociales                     |
| N3      | Motores de búsqueda o portales web |
| N4      | Compras                            |
| N5      | Mapas o servicios de ubicación     |
| N6      | Correo electrónico                 |
| N7      | Música                             |
| N8      | Noticias                           |
| N9      | Clima                              |
| N10     | Entretenimiento                    |
| N11     | Juegos                             |
| N12     | Servicios de movilidad             |
| N13     | Bancos                             |
| N14     | Deportes                           |
| N15     | Viajes                             |

|     |                   |
|-----|-------------------|
| N16 | Salud y bienestar |
| N17 | Educación         |
| N18 | Domicilios        |

A continuación se eligió la muestra que representa cada uno de los estratos (dominios), como la técnica de estratificación no determina un patrón a seguir para la elección de muestras en cada uno de los estratos sino que permite una elección arbitraria denominada 'Muestreo estratificado aleatorio' (Lohr, 1999), para la elección de las aplicaciones se tomó como criterio, que además de pertenecer al dominio asociado, sean sitios o aplicaciones que se encuentren bien posicionadas y que sean fácilmente reconocidas; siguiendo esta premisa la muestra de aplicaciones elegidas es la siguiente:

Tabla 2. Aplicaciones seleccionadas por estrato

| Estrato | Nombre de la categoría             | Apps elegidas (n=1) |
|---------|------------------------------------|---------------------|
| N1      | Chat y Mensajería                  | Google Meet         |
| N2      | Redes sociales                     | Facebook            |
| N3      | Motores de búsqueda o portales web | Google              |
| N4      | Compras                            | MercadoLibre        |
| N5      | Mapas o servicios de ubicación     | Google Maps         |
| N6      | Correo electrónico                 | Gmail               |
| N7      | Música                             | Spotify             |
| N8      | Noticias                           | Microsoft News      |
| N9      | Clima                              | Accu Weather        |
| N10     | Entretenimiento                    | Hbo Max             |
| N11     | Juegos                             | Plantas vs Zombis   |
| N12     | Servicios de movilidad             | In Drive            |
| N13     | Bancos                             | Nequi               |
| N14     | Deportes                           | One Football        |
| N15     | Viajes                             | Trip Advisor        |
| N16     | Salud y bienestar                  | Mi Fitness          |
| N17     | Educación                          | Duolingo            |
| N18     | Domicilios                         | Rappi               |

### 3.1.4. Abstracción de procesos y alertas de usuario

Para llevar a cabo este proceso, se buscó dentro de las aplicaciones seleccionadas, procesos que contengan los patrones de flujo objetivo, usando *el método de Silver* como método de abstracción de los procesos. Para cada proceso abstraído se ejecutó una instancia y se obtuvieron las alertas de usuario generadas, estas alertas se describirán de acuerdo con las características mencionadas en la sección 2.2 agregando también el punto dentro del proceso abstraído en el cual se presentó la alerta.

Las alertas de usuario elegidas fueron aquellas que estuvieron estrictamente relacionadas con el proceso que se llevó a cabo, es decir que las alertas de usuario que no estaban relacionadas con las actividades o decisiones tomadas dentro del proceso abstraído, no se tomaron en cuenta.

### 3.1.4.1. Dominio chat y mensajería

En esta sección se realizó la abstracción y captura de alertas para Google Meet, la aplicación elegida para el dominio de chat y mensajería. El patrón de flujo *secuencia* se abstrajo desde el proceso *crear una reunión*, proceso que permite a los usuarios crear videollamadas en la aplicación. Los resultados de la abstracción y alertas de usuario capturadas se muestran en la Tabla 3.

Tabla 3. Abstracción de proceso y alertas de usuario, patrón de flujo secuencia - Google Meet

| Datos del proceso  |  |
|--|--|
| Patrón buscado   | Secuencia  |
| Aplicación   | Google Meet  |
| Proceso  | Crear una reunión  |
| Plataforma   | Aplicación web   |
| Datos de abstracción del proceso   |  |
| ¿Qué inicia el proceso?  | Se inicia cuando un usuario cree una nueva reunión   |
| ¿Qué representa cada instancia o ejecución del proceso?  | Una nueva reunión virtual ha sido creada permitiendo conectarse mediante videollamada a un grupo de personas |
| ¿Qué determina que el proceso está completo?   | Que la reunión haya sido creada  |
| ¿Cuáles son las formas en las que el proceso puede terminar?   | La reunión ha sido creada  |
| Mapeo de actividades de alto nivel   | Ingresar a la opción 'Nueva reunión'<br>Seleccionar 'Iniciar reunión ahora'                                  |
| Diagrama de proceso de alto nivel  |  |
| <pre> graph LR     Actor[Usuario creador] --&gt; Start(( ))     Start --&gt; A[Seleccionar crear reunión]     A --&gt; B[Seleccionar Iniciar ahora]     B --&gt; End(( ))     </pre> |  |
| Alerta de usuario #1   |  |
| Alerta de usuario  |  |
| Título   | La reunión está lista  |
| Contenido  | También puedes compartir esta información para unirse con las personas que quieras que asistan a la reunión  |
| Tipo   | Confirmación   |
| Color  | Ninguno  |
| Botones  | Añadir a alguien   |
| Representación visual  | Caja de texto  |
| Ubicación en el proceso  | Fin del proceso  |

|                         |                 |
|-------------------------|-----------------|
| <b>Usuario objetivo</b> | Usuario creador |
|-------------------------|-----------------|

En este caso particular dentro del proceso *solicitar unirse a una llamada* se mapearon los patrones de flujo división/unión paralela y división/unión exclusiva, la abstracción del proceso y alertas de usuario presentadas se muestran en la Tabla 4.

Tabla 4. Abstracción de proceso y alertas de usuario, patrón de flujo división/unión paralela y división/unión exclusiva - Google Meet

| Datos del proceso   |   |
|---|---|
| <b>Patrón buscado</b>   | División / Unión paralela – División / Unión exclusiva  |
| <b>Aplicación</b>   | Google Meet   |
| <b>Proceso</b>  | Solicitar unirse a una llamada  |
| <b>Plataforma</b>   | Aplicación web  |
| Datos de abstracción del proceso  |   |
| <b>¿Qué inicia el proceso?</b>  | El proceso se inicia cuando una persona solicita unirse a una reunión mediante la opción de 'Solicitar unirse'.   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>  | Cuando la persona que tiene el control de la llamada acepta o rechaza a la persona que solicita unirse.   |
| <b>¿Qué determina que el proceso está completo?</b>   | Representa una persona que solicita ingresar a una reunión.   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>   | El proceso puede terminar con una persona agregada a la reunión o con la solicitud rechazada y el usuario no puede ingresar a la reunión.   |
| <b>Mapeo de actividades de alto nivel</b>   | <ul style="list-style-type: none"> <li>El usuario que desea ingresar a la llamada entra al sitio web de la llamada.</li> <li>El usuario que está en la llamada acepta o rechaza la solicitud</li> </ul> |
| Diagrama de proceso de alto nivel   |   |
| <pre> graph TD     subgraph "Google Meet - Solicitar unirse a una llamada"         direction TB         subgraph "Solicitante"             S(( )) --&gt; S1[Solicitar unirse]         end         subgraph "Participante"             P1[Aprobar solicitud de ingreso]         end         subgraph "Google Meet"             G1[Unir usuario solicitante a la llamada]         end     end      S1 --&gt; J1{+}     J1 --&gt; P1     P1 --&gt; U1{X}     G1 --&gt; U1     U1 -- Si --&gt; G2[Unir usuario solicitante a la llamada]     U1 -- No --&gt; U2{X}     S1 --&gt; U2     U2 --&gt; J2{+}     J2 --&gt; E(( ))     S1 --&gt; U3{X}     U3 --&gt; J3{+}     J3 --&gt; E     </pre> |   |
| Alerta de usuario #1  |   |

|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |    |
| <b>Título</b>                  | Alguien quiere unirse a esta llamada   |
| <b>Contenido</b>               | {Nombre usuario}   |
| <b>Tipo</b>                    | Notificación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | Admitir, Vista   |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Compuerta de decisión exclusiva  |
| <b>Usuario objetivo</b>        | Participante   |
| <b>Alerta de usuario #2</b>    |  |
| <b>Alerta de usuario</b>       |   |
| <b>Título</b>                  | Pidiendo permiso para acceder  |
| <b>Contenido</b>               | Te unirás a la llamada cuando alguien te permita entrar                              |
| <b>Tipo</b>                    | <i>Feedforward</i>   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Gateway de decisión exclusiva  |
| <b>Usuario objetivo</b>        | Solicitante  |
| <b>Alerta de usuario #3</b>    |  |
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | No puedes unirte a esta llamada  |
| <b>Contenido</b>               | Alguien de la reunión ha rechazado tu solicitud para unirse                          |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |

|                                |   |
|--------------------------------|---|
| <b>Ubicación en el proceso</b> | Compuerta de unión exclusiva, opción 'No admitir' |
| <b>Usuario objetivo</b>        | Solicitante                                       |

#### 3.1.4.2. Dominio redes sociales

Para el análisis del dominio redes sociales la aplicación elegida fue Facebook, la cual es hoy en día una de las redes sociales más usadas en el mundo. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 2 del Anexo 1.

#### 3.1.4.3. Dominio motores de búsqueda o portales web

Para el dominio de motores de búsqueda, la aplicación seleccionada fue Google Search, uno de los buscadores más grandes que existen en internet. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 3 del Anexo 1.

#### 3.1.4.4. Dominio compras

En el dominio de compras, la aplicación elegida fue MercadoLibre, la cual una de las plataformas líderes en comercio electrónico de América Latina. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 4 del Anexo 1.

#### 3.1.4.5. Dominio servicios de ubicación

En el dominio de servicios de ubicación, la aplicación elegida fue Google Maps, plataforma que facilita la búsqueda de lugares e indicaciones. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 5 del Anexo 1.

#### 3.1.4.6. Dominio correo electrónico

Para el dominio de correo electrónico, la aplicación elegida fue Gmail, uno de los servicios de correo electrónico más reconocidos a nivel mundial. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 6 del Anexo 1.

#### 3.1.4.7. Dominio música

En el dominio de música la aplicación elegida fue Spotify, una plataforma de streaming de música que cuenta con millones de canciones. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 7 del Anexo 1.

#### 3.1.4.8. Dominio noticias

Para el dominio de noticias, la aplicación elegida fue Microsoft News, la cual proporciona acceso a noticias con cobertura mundial a través de internet. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 8 del Anexo 1.

#### 3.1.4.9. Dominio clima

Entre las aplicaciones del dominio de clima, la elegida fue Accu Weather, la cual provee información y pronósticos meteorológicos con alertas personalizadas a través de internet. Los

resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 9 del Anexo 1.

#### 3.1.4.10. Dominio entretenimiento

Para el dominio de entretenimiento la aplicación elegida fue Hbo Max plataforma de streaming audiovisual con un amplio catálogo de películas, series y contenido exclusivo. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 10 del Anexo 1.

#### 3.1.4.11. Juegos

En el dominio de juegos, el juego elegido fue Plantas vs Zombis, un juego clásico de estrategia en donde el objetivo es proteger el jardín de hordas de zombis. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 11 del Anexo 1

#### 3.1.4.12. Dominio movilidad

En cuanto al dominio movilidad, la aplicación elegida fue In Drive, la cual es una plataforma que conecta conductores con pasajeros permitiendo la negociación del precio del servicio. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 12 del Anexo 1.

#### 3.1.4.13. Dominios bancos

Para el dominio de bancos, la aplicación elegida fue Nequi, una plataforma que permite transferencias de manera ágil y que es ampliamente utilizada para transferencias de bajo monto. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 13 del Anexo 1.

#### 3.1.4.14. Dominio deportes

En cuanto al dominio de deportes, la aplicación elegida fue One Football, plataforma que permite acceder a noticias, estadísticas y actualizaciones de las ligas de fútbol principales a nivel mundial. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 14 del Anexo 1.

#### 3.1.4.15. Dominio viajes

Entre las aplicaciones en el dominio de viajes, la elegida fue Trip Advisor, plataforma que facilita la planificación de viajes en cuanto a lugares turísticos, transporte y hospedaje. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 15 del Anexo 1.

#### 3.1.4.16. Dominio de salud y bienestar

La aplicación elegida para el dominio de salud y bienestar fue Mi Fitness, una aplicación para el registro y seguimiento de parámetros de salud y actividad física. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 16 del Anexo 1.



### 3.1.4.17. Dominio educación

En dominio de educación, la aplicación elegida fue Duolingo, la cual es líder en plataformas de enseñanza de idiomas enfocada en la gamificación del aprendizaje. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 17 del Anexo 1.

### 3.1.4.18. Dominio domicilios

En el dominio de domicilios, la aplicación elegida fue Rappi, una plataforma que conecta restaurantes, supermercados, farmacias y negocios en general con domiciliarios que realizan entregas a usuarios, enfocada en la rapidez en el servicio. Los resultados de la abstracción de procesos y alertas de usuario relacionadas se pueden ver en la sección 18 del Anexo 1.

## 3.2. Propuesta de patrones de alertas de usuario sobre patrones de flujo

El objetivo de esta sección fue organizar los datos obtenidos para alertas de usuario y patrones de flujo, definir variables y métricas y con ellas determinar posibles patrones de alertas de usuario, los cuales se usarán en la construcción del componente para la generación dinámica de alertas de usuario.

### 3.2.1. Estructuración de los datos obtenidos

Partiendo de la información obtenida anteriormente, se estructurarán los datos en una matriz de alertas/atributos, en donde cada fila representará cada una de las alertas de usuario obtenidas, y cada columna corresponderá a los siguientes atributos:

- **Aplicación:** El nombre de la aplicación o sitio web de donde se extrajo la alerta
- **Dominio:** El nombre del dominio al cual pertenece la aplicación
- **Proceso:** El nombre del proceso donde se presentó la alerta
- **Patrón:** El patrón buscado en el proceso de donde se obtuvo la alerta
- **Tipo:** La clasificación por tipo de alerta de usuario
- **Color:** El color de la alerta de usuario
- **Icono:** Icono agregados en la alerta de usuario (si lo hay)
- **Representación visual:** La forma en la que la alerta fue presentada al usuario, de acuerdo con la definición, puede ser Caja de texto o Palabra simple
- **Ubicación en el proceso:** El punto del proceso abstraído en donde se presentó la alerta. Este atributo podrá ser una de las siguientes opciones:
  - Actividad inicial: Si la alerta se presenta en la primera actividad del proceso
  - Actividad central: Si la alerta se presenta en las actividades intermedias del proceso
  - Actividad control compuerta: Si la alerta se presenta en una actividad, cuyo resultado controla la dirección del flujo en una compuerta de división
  - Actividad final: Si la alerta se presenta en la actividad anterior al evento de fin del proceso
  - Actividad paralela: Si la alerta se presenta en una actividad que se está realizando de manera paralela al usuario objetivo de la alerta.

- Compuerta división: Si la alerta se presenta en la compuerta en donde se divide el flujo, bien sea de forma exclusiva o paralela
- Compuerta unión: Si la actividad se presenta en la compuerta en donde se une el flujo
- Fin del proceso: Si la actividad se presenta una vez se ha finalizado el proceso.
- **Acciones:** La clasificación de la(s) acción(es) que se muestran en la alerta de usuario, pudiendo ser alguna de las siguientes opciones:
  - Continuar proceso: Si la acción presentada en la alerta tiene el objetivo de avanzar la instancia
  - Deshacer proceso: Si la acción presentada tiene el objetivo de deshacer la instancia que está en ejecución
  - Finalizar proceso: Acción que permite cerrar o finalizar la instancia
  - Relacionada al dominio: Cuando la acción presentada no está relacionada estrictamente al proceso y su objetivo, pero sí al dominio (otros procesos) de la aplicación
  - Relacionada a la actividad actual: Si la acción está enfocada de una manera más puntual a la actividad actual que se está ejecutando en la instancia respecto al proceso
  - Relacionada a compuerta: Si la acción está enfocada de una manera más puntual a la compuerta actual que se está ejecutando en la instancia respecto al proceso
  - Relacionada al proceso: Cuando la acción está relacionada de una manera general al objetivo del proceso
  - Repetir proceso: Si la acción permite que inicie una nueva instancia del proceso.
  - Ver resultado proceso: Si la acción permite mostrar estadísticas o datos relacionados al proceso.
- **Acciones puntuales:** El texto de la(s) acciones(s) presentadas (si las hay), cada acción estará encerrada por corchetes de la forma [<<acción>>]... ubicándose una al lado de otra en caso que exista más de una.
- **Título:** La clasificación del título presentado en la alerta, puede ser una de las siguientes opciones:
  - Relacionado al proceso: Si el título está relacionado al proceso en general
  - Relacionado a la actividad actual: Si el título está relacionado de una manera más puntual a la actividad que se está ejecutando en la instancia
  - Relacionado a actividad realizada: Si el título está relacionado a una actividad anterior a la actividad que se está ejecutando en la instancia
  - Relacionado a otra actividad: Si el título está relacionado a otra actividad que no es la inmediatamente anterior a la actividad actual de la instancia sino una distinta
  - Relacionado a acción esperada: En los patrones de división paralela, cuando el título se refiera a una actividad paralela ajena al usuario objetivo.
- **Título puntual:** El texto del título de la alerta (si lo hay)
- **Contenido:** La clasificación del contenido de la alerta, puede ser una de las siguientes opciones:
  - Relacionado a la acción actual: Si el contenido está relacionado a la acción que se está ejecutando actualmente en la instancia.

- Relacionado a acción esperada: En los patrones de división paralela, cuando el contenido esté relacionado a una actividad paralela ajena al usuario objetivo.
- Relacionado a acción futura: Si el contenido está relacionado de una forma puntual a una actividad que se ejecutará en pasos siguientes en el proceso.
- Relacionado a consecuencia de la actividad: Si el contenido de la alerta está relacionado a implicaciones futuras en el proceso como consecuencia de la actividad actual de la instancia.
- Relacionado al dominio: Si el contenido no está asociado de manera directa al proceso, pero si al dominio (otros procesos) de la aplicación.
- Relacionado al proceso: Si el contenido está relacionado de forma general al proceso que se está ejecutando.
- Relacionado al título: Si el contenido está relacionado al título del proceso.
- **Contenido puntual:** El texto del contenido de la alerta

### 3.2.2. Definición de variables y métricas

Con el objetivo de establecer guías en el análisis de los datos estructurados y que estos puedan a futuro ser ampliados y analizados nuevamente, a continuación, se definen los siguientes conceptos:

- **Filtros:** A los datos estructurados se les podrá aplicar distintos filtros sobre una columna y valor(es) específico(s), por ejemplo, agregar el filtro tipo=confirmación generará un nuevo conjunto de datos que incluiría sólo las alertas cuyo atributo Tipo sea Confirmación; Sobre este nuevo conjunto de datos se podrán añadir más filtros como sean necesarios.
- **Nivel de filtro:** Cada filtro agregado aumentará en 1 el nivel de filtrado, siendo el conjunto de datos original el nivel 0.
- **Variables**
  - Dominios en la investigación: Este valor corresponde al total de dominios que se incluyeron en la investigación de alertas, que para este caso fueron 18
  - Dominios filtrados: Este valor corresponde al total de dominios que presentaron alertas de usuario aplicando los filtros hasta un nivel específico que puede ser igual o menor que el nivel de los filtros de la gráfica
  - Dominios gráfica: Este valor corresponde al total de dominios que están incluidos en una gráfica con todos los filtros aplicados
- **Métricas:** Las siguientes métricas se crean con el objetivo de medir el valor de los resultados mostrados en una gráfica, está centrada en los dominios más allá de las frecuencias para obtener una visión que no esté sesgada por dominios con muchas alertas.
  - Proporción de dominios filtrados (PRD-f): Su objetivo será determinar qué tan representativos son los dominios mostrados en una gráfica respecto de los dominios filtrados. Su valor será de 0 a 1, con 1 indicando que todos los dominios filtrados se encuentran incluidos en la gráfica.
 
$$PRD - f = \frac{\text{Dominios gráfica}}{\text{Dominios filtrados}}$$
  - Proporción de dominios general (PRD): Su objetivo será determinar qué tan representativos son los dominios mostrados en una gráfica, respecto a todos los dominios de la investigación. Su valor será de 0 a 1 con 1 indicando que todos los dominios de la investigación se encuentran incluidos en la gráfica.

$$PRD = \frac{\text{Dominios gráfica}}{\text{Dominios en la investigación}}$$

- **Interpretación de métricas:**

- La determinación de uso del PRD-f o PRD la dará el contexto de la o las variables analizadas, considerando el uso de la métrica general sólo para casos donde se realicen validaciones sobre todo el universo de datos sin tomar en cuenta fraccionamiento por patrones de flujo u otros filtros relevantes.
- Como el objetivo del análisis es buscar patrones dentro de los datos obtenidos para hallar similitudes entre las alertas presentadas por distintos dominios, la representatividad de un conjunto de datos con un PRD-f o PRD se interpretará de la siguiente manera:
- Valor entre 0 y 0,2, se considerará insuficiente para representar un patrón común, quiere decir que estos datos serán descartados como patrones.
- Valor entre 0,21 y 0,6, se considerará opcional como patrón, estos valores requerirán de validaciones adicionales o de contextualización para ser descartados o aprobados como patrones.
- Valor entre 0,61 y 1 se considerarán como patrones comunes de manera directa debido a su alta representatividad comparada con los dominios

### 3.2.3. Análisis de patrones

#### 3.2.3.1. De secuencia

Este análisis inicia con el filtrado de los datos por *Patrón=Secuencia*, filtrando el consolidado sólo con las alertas que se presentaron para este tipo de patrón. Las alertas que se presentaron para este tipo de patrón se distribuyen a lo largo del proceso como se muestra en la Figura 32. Distribución por ubicación en el proceso para alertas el patrón secuencia, con el 63,16% de las alertas en el fin del proceso, mientras que el 36,84% se presentaron en actividades centrales del proceso.

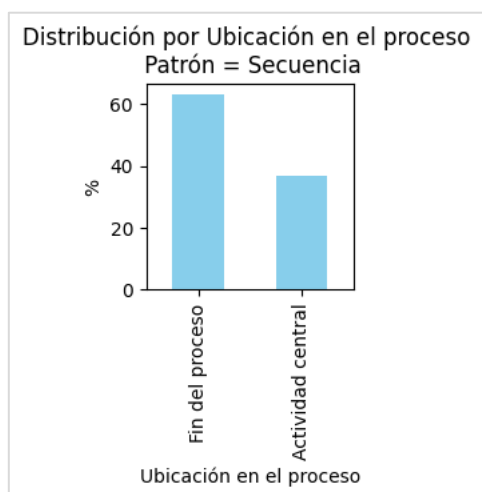


Figura 32. Distribución por ubicación en el proceso para alertas el patrón secuencia

Al resultado anterior se le agrega un nuevo filtro para analizar datos sobre las alertas presentadas en cada punto del proceso, siendo primero las alertas ubicadas en el fin del proceso, las cuales se distribuyeron por tipos como se muestra en la Figura 33.

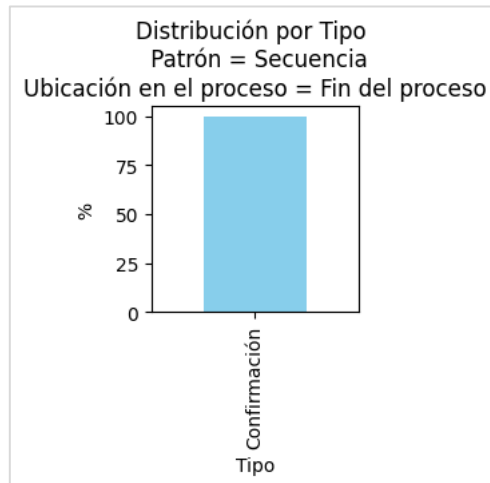


Figura 33. Distribución por tipo para alertas ubicadas en el fin del proceso en patrones de secuencia

El 100% de las alertas presentadas en el fin del proceso corresponden a alertas de tipo Confirmación, en este caso los dominios filtrados en el nivel 1 (*Patrón=Secuencia*) son en total 14, mientras que en la gráfica se incluyen 12 dominios, lo que significa un  $PRD-f = 0.86$ , lo cual de acuerdo con las métricas definidas representa un patrón de alertas de usuario presentadas, el cual indica que en *los patrones de secuencia se debería mostrar una alerta de Confirmación en el fin del proceso*.

En cuanto a las alertas presentadas en actividades centrales como se muestra en la Figura 34, la distribución por tipo es 57.14% de tipo Advertencia, 28.57% de tipo *Feedforward* y 14.29% de tipo *Feedback*.

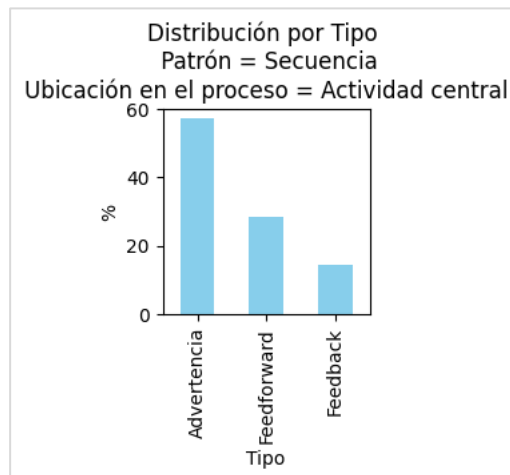


Figura 34. Distribución por tipo para alertas ubicadas en actividades centrales en patrones de secuencia

En el caso de la alerta de *feedback*, sólo se presentó una vez para el dominio de educación, con un  $PRD-f = 0.07$ , lo cual, de acuerdo con lo establecido, es un valor insuficiente para asumir un patrón.

En el caso de las alertas de *Advertencia* y *feedforward*, estas se analizarán como un mismo tipo, teniendo en cuenta que las alertas de *Advertencia* son en esencia alertas de *Feedforward* con una connotación de precaución sobre el mensaje. Estos dos tipos de alerta se presentaron en 5 dominios con un  $PRF-f = 0.36$ , el cual aún sin considerarse despreciable no es un valor

que pueda significar un patrón. Los dominios incluidos en estos dos tipos de alerta son Juegos, Movilidad, Bancos, Salud y bienestar y Domicilios, en el caso de Movilidad, Bancos y Domicilios la presencia de estas alertas tiene el objetivo de confirmar las transacciones / acciones realizadas y permitir al usuario estar informado del proceso en todo momento; Por otra parte, en Juegos y Salud y bienestar, estas alertas se presentaron para informar al usuario sobre el siguiente paso(s) en el proceso. Con esto presente, este tipo de alertas está más ligada al proceso que a una generalidad, ya que las validaciones y corroboraciones en algunos procesos podrían considerarse como no necesarias dado que las acciones realizadas o son idempotentes o no presentan ningún riesgo en caso de realizarse de manera incorrecta, por esta razón este tipo de alertas se considerarán como patrón de acuerdo con el riesgo del proceso, mostrándose sólo para procesos que tengan algún tipo de riesgo o no idempotencia.

### 3.2.3.2. De división/unión paralela

Al igual que en el patrón de secuencia, para este caso se realizará un filtro de los datos consolidados por *Patrón=División/unión paralela* y se analizará la distribución de las alertas presentadas según el punto en el proceso donde estas se mostraron como se muestra en la Figura 35, presentándose un 35.29% de las alertas en la compuerta de división, 17.65% en la compuerta de unión, 17.65% en el fin del proceso, 17.65% en actividades paralelas 5.88% en actividades centrales y 5.88% en actividades finales.

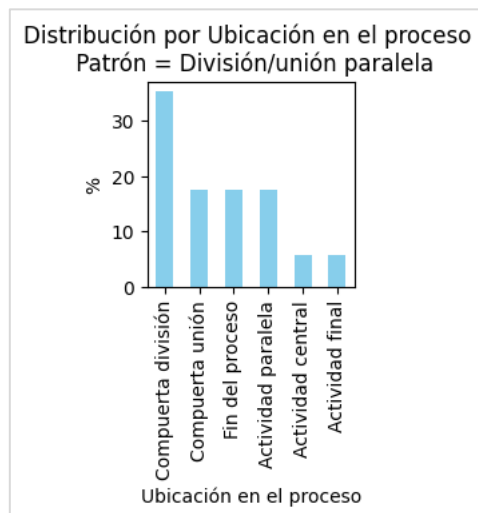


Figura 35. Distribución por tipo para alertas presentadas en el patrón de división/unión paralela

Las alertas de usuario que se presentaron en la compuerta de división se clasifican entre alertas de *Feedforward* con un 83.4% y *Feedback* 16.6% como se muestra en la Figura 36, el total de dominios que presentaron alertas para el filtro de nivel 1 (*Patrón=División/unión paralela*) fueron 7, de los cuales 4 presentaron alertas en la compuerta de división para un  $PRD-f = 0.57$ , lo cual significa que este tipo de alertas son un candidato potencial para convertirse en un patrón.

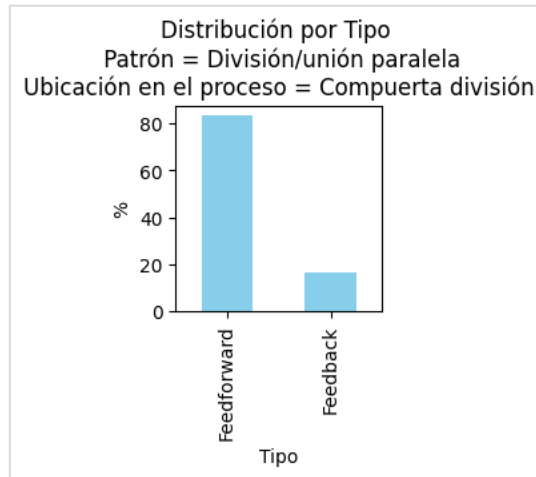


Figura 36. Distribución por tipo para alertas presentadas en compuerta de división en patrones de división/unión paralela

En el caso de las notificaciones de *feedforward* tienen un  $PRD-f = 1$ , lo cual quiere decir que de los 4 dominios que mostraron alertas de usuario en la compuerta de división, los 4 mostraron alertas de tipo *Feedforward*, mientras que en el caso de las alertas de *Feedback* sólo incluyeron 1 dominio para un  $PRD-f = 0.25$ , por lo que se descartan las alertas de *feedback*, esto significa que el patrón encontrado en este caso dice que: *Los patrones de división/unión paralela deberían mostrar una alerta de feedforward en la compuerta de división.*

En el caso de las alertas que se presentan en la compuerta de unión se clasifican entre Confirmación (33.3%), Notificación (33.3%) y Advertencia (33.3%) como se muestra en la Figura 37, del total de 7 dominios que presentaron alertas en el patrón de División/unión paralela, sólo 2 presentaron alerta en este punto del proceso, lo cual significa un  $PRD-f = 0.29$ , tomando en cuenta este dato, no se tomarán en cuenta las alertas en este punto del proceso como patrón.

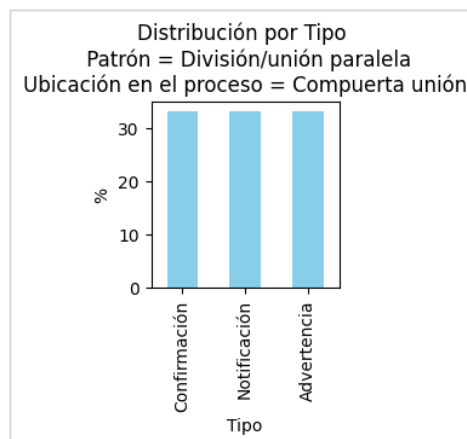


Figura 37. Distribución por tipo para alertas mostradas en compuerta de unión en patrón de división/unión paralela

Las alertas presentadas al final del proceso o en la actividad final, al igual que en el patrón de secuencia corresponden 100% al tipo confirmación como se observa en la Figura 38.

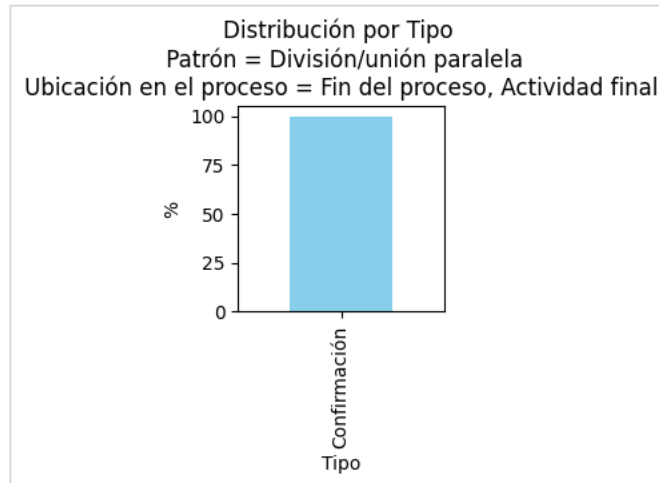


Figura 38. Distribución por tipo para alertas presentadas en el fin del proceso en patrones de División/unión paralela

De los 7 dominios que presentaron alertas en el patrón de división/unión paralela, 3 dominios presentaron alerta al final del proceso para un  $PRD-f = 0.43$ , en este punto teniendo en cuenta la tendencia obtenida para los patrones de secuencia, en donde este tipo de alertas se convirtió en un patrón, considerando que un patrón de división/unión y un patrón de secuencia no se diferencian en este punto, también se considerará un patrón en este caso, el cual dice que: *En un patrón de división/unión exclusiva se debería mostrar una alerta de confirmación en el final del proceso.*

En el caso como se observa en la Figura 39 de las alertas presentadas en actividades paralelas, los tipos se dividieron entre Confirmación (66.7%) y *Feedforward* (33.3%)

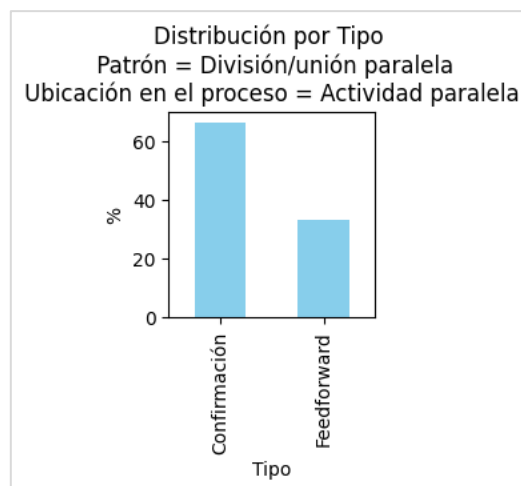


Figura 39. Distribución por tipo para alertas presentadas en actividad paralela en patrones de división/unión paralela

Las alertas en este punto del proceso se presentaron en 2 dominios, lo cual significa un  $PRD-f = 0.29$  tomando en cuenta los 7 dominios que presentaron alertas para el patrón de división/unión paralela, los dominios incluidos son Música y Domicilios, y la alertas presentadas en estos estuvieron relacionadas al proceso en específico, en donde al usuario se le mostraba el estado del proceso en su flujo paralelo, teniendo esto en consideración, no se considera como patrón las alertas presentadas en este punto del proceso.



En cuanto a actividades centrales, las alertas en este punto sólo se presentaron en 1 dominio, es decir un  $PRD-f = 0.14$  respecto al total de dominios que presentaron alertas para el patrón de división/unión paralela, por lo que no se considerarán como un patrón.

### 3.2.3.3. De división/unión exclusiva

Siguiendo con la línea de los anteriores patrones analizados, se agrega un filtro sobre los datos consolidados con  $Patrón=División/unión\ exclusiva$  como se muestra en la Figura 40, las alertas que se presentaron en este patrón están ubicadas en Compuerta de división (31.25%), fin del proceso (25%), Actividad de control de compuerta (18.75%), actividad central (18.75%) y actividad final (6.25%).

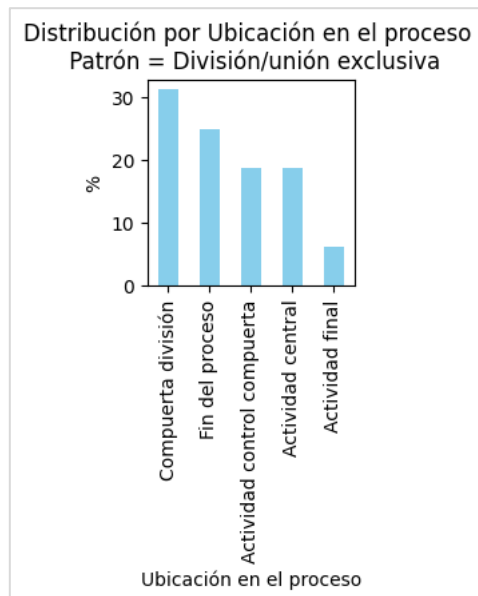


Figura 40. Distribución por ubicación en el proceso para alertas del patrón división/unión exclusiva

A continuación, se realiza la validación de cada las alertas presentadas en cada uno de estos puntos, en el caso de compuerta de división, se analizará junto con actividad control compuerta, ya que ambos puntos están relacionados al mismo control de flujo. La distribución por tipos para las alertas en estos puntos se puede ver en la Figura 41, las alertas presentadas en este punto del proceso incluyeron 6 dominios de los 8 que presentaron alertas en el patrón de división/unión exclusiva, para un  $PRD-f = 0.75$ , lo que sugiere que las alertas en este punto corresponden a un patrón.

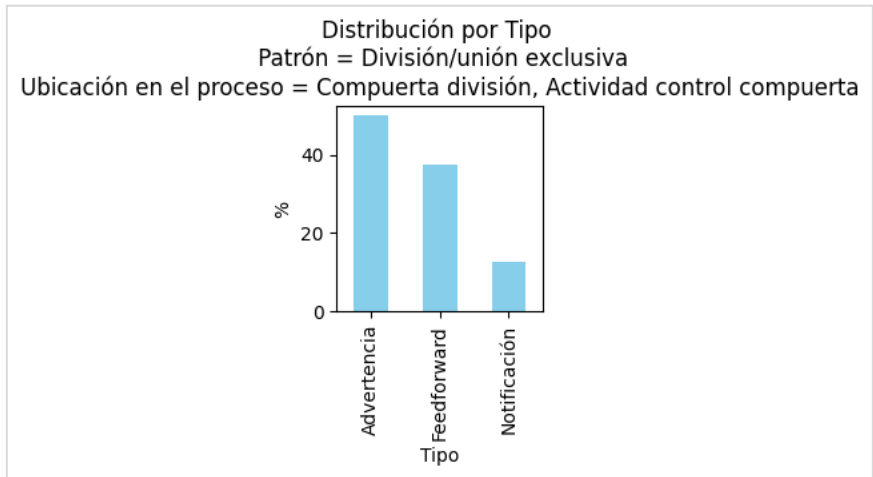


Figura 41. Distribución por tipo para alertas presentadas en compuerta de división o actividad control compuerta en patrón división/unión exclusiva

Para determinar el tipo de alertas del patrón se analizaron de forma independiente los tipos de alerta generada para validar su representatividad, iniciando por los tipos Advertencia y *Feedforward*, que como se mencionó anteriormente, se pueden analizar juntos tomando en cuenta que la alerta de Advertencia es en esencia una alerta de *Feedforward* con connotaciones de riesgo, estos dos tipos de alerta se presentaron en 5 de los 6 dominios que presentaron alertas en la compuerta de división / actividad control compuerta, dando como resultado un  $PRD-f = 0.83$ , lo cual indica un patrón, que se define como *En un proceso con el patrón división/unión exclusiva, debería mostrarse una alerta de advertencia o feedforward en la compuerta de división*; Para el caso de las alertas de notificación estas sólo se presentaron en 1 dominio de los 6 dominios que presentaron alertas en la compuerta de división / actividad control compuerta, dando como resultado un  $PRD-f = 0.17$ , por lo que se descarta este tipo de alerta en este punto del proceso como patrón.

En el caso de alertas presentadas en actividad central estas fueron de tipo Advertencia un 100% como se muestra en la Figura 42, sin embargo, sólo se presentaron en 1 de los 8 dominios con alertas en el patrón división/unión exclusiva, para un  $PRD-f = 0.12$ , que hace que las alertas en este punto no se consideren un patrón.

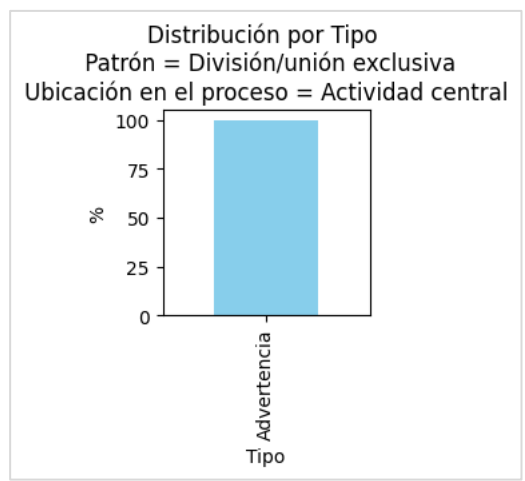


Figura 42. Distribución por tipo para alertas presentadas en actividad central en procesos de división/unión exclusiva

En el caso de las tareas ubicadas en el Fin del proceso, estas corresponden en un 100% a alertas de confirmación como se muestra en la Figura 43, estas alertas se presentan en 4 de los 8 dominios que presentaron alertas en el patrón de división/unión exclusiva, para un  $PRD-f = 0.5$ , en este caso el valor en sí de acuerdo con los parámetros definidos no representa un patrón de manera directa. Sin embargo, tomando en cuenta que este patrón se ha definido en los análisis anteriores y tiene un valor adecuado en su  $PRD-f$ , en este caso también se le dará el estatus de patrón, definiéndose como *Los procesos de división/unión exclusiva deberían mostrar una alerta de confirmación en el fin del proceso.*

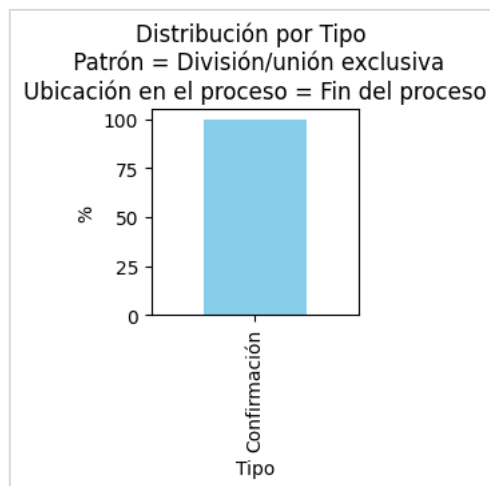


Figura 43. Distribución por tipo para alertas presentadas en el fin del proceso en patrones de división/unión exclusiva

Finalmente se analizan las alertas presentadas en la actividad final son de tipo advertencia (100%) como se observa en la Figura 44. Sin embargo, al sólo presentarse en un dominio de los 8 que mostraron alertas en el patrón de división/unión exclusiva, su  $PRD-f = 0.12$  indica que no puede ser considerada como patrón.

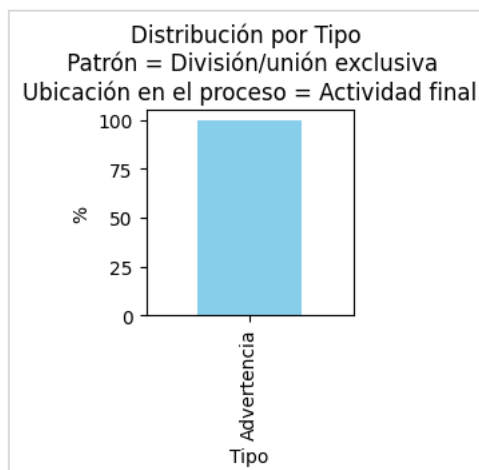


Figura 44. Distribución por tipo para alertas presentadas en actividad final en patrones de división/unión exclusiva

### 3.2.4. Definición de sugerencias de alerta de usuario sobre patrón de proceso

A partir del análisis anterior, se examinó cada uno de los patrones definidos para evaluar sus alertas. El objetivo de identificar características comunes en los atributos que conforman la alerta de usuario, como título, contenido, color, representación visual, icono y acciones. Para cada patrón, se creará una sugerencia de alerta de usuario que incluirá todos sus atributos. En caso que las alertas no presenten características comunes para algún atributo en concreto, se generarán sugerencias basadas en el marco conceptual.

#### 3.2.4.1. Patrón 1

En las instancias de proceso con un patrón de secuencia, se debería mostrar una alerta de confirmación al finalizar el proceso. A continuación, se detalla sobre los atributos que componen la alerta.

Según los datos recopilados, se observó que un 75% de este tipo de alertas se presentaron sin título, como se muestra en la Figura 45. Sin embargo, para agregar coherencia entre el tipo de alerta (confirmación) y el punto en el que esta se presenta (fin del proceso), se sugiere utilizar el título 'Terminado'.

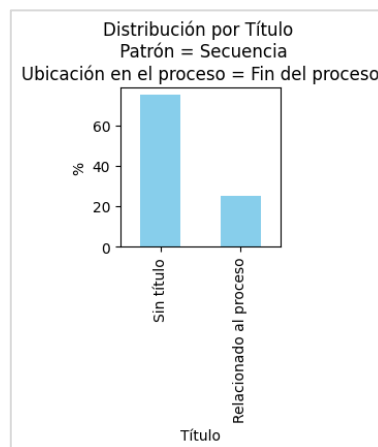


Figura 45. Frecuencia de títulos de patrón 1

En cuanto al contenido, se observó que el 50% del contenido de las alertas estuvo relacionado al proceso, mientras que el 41.7% relacionado al título del proceso, como se muestra en la Figura 46.

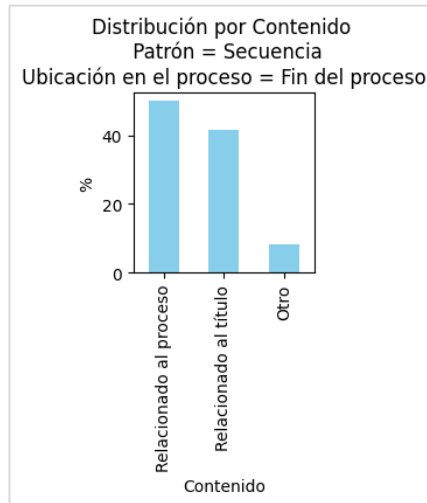


Figura 46. Frecuencia de contenidos de patrón 1

Aunque la mayoría de los contenidos de las alertas que estuvieron relacionados al proceso, estos contenidos no se pueden generalizar, ya que están relacionados a datos específicos asociados a cada proceso o dominio. En contraste, los contenidos relacionados al título si se pueden generalizar, ya que estos son una oración en pasado o en pasado participio del verbo junto a su sujeto y complemento del proceso como se muestra en la Tabla 5.

Tabla 5. Alertas de confirmación en final del proceso con contenido relacionado al título

| Título del proceso         | Verbo     | Sujeto   | Complemento | Contenido                          |
|----------------------------|-----------|----------|-------------|------------------------------------|
| Compartir historia         | Compartir | historia |             | Se compartió tu historia           |
| Enviar pregunta            | Enviar    | pregunta |             | Tu pregunta se envió correctamente |
| Agregar canción a me gusta | Agregar   | Canción  | a me gusta  | Se agregó a tus me gusta           |
| Realizar envío             | Realizar  | envío    |             | Envío realizado                    |
| Seguir jugador             | Seguir    | jugador  |             | Ahora sigues a {jugador}           |

Dada la naturaleza y objetivo de las alertas de usuario de confirmación en este punto de proceso, se propone organizar el contenido de manera estándar como {sujeto} {verbo en participio} {complemento}. De esta manera la semántica de las alertas se mantendría consistente, pero con una estructura sintáctica general, como se muestra en la Tabla 6.

Tabla 6. Equivalencia de contenido propuesta para alertas de confirmación en el fin del proceso

| Contenido original                 | Contenido propuesto         |
|------------------------------------|-----------------------------|
| Se compartió tu historia           | Historia compartida         |
| Tu pregunta se envió correctamente | Pregunta enviada            |
| Se agregó a tus me gusta           | Canción agregada a me gusta |
| Envío realizado                    | Envío realizado             |
| Ahora sigues a {jugador}           | Jugador seguido             |

En el 100% de las alertas de este patrón, no se presentaron acciones, por lo que se considera que no incluir acciones es una característica general de este tipo de alerta.

En cuanto a la representación visual, el 66.7% de las alertas utilizó una caja de texto, como se muestra en la Figura 47, lo que la convierte en la representación elegida para la sugerencia de este patrón.

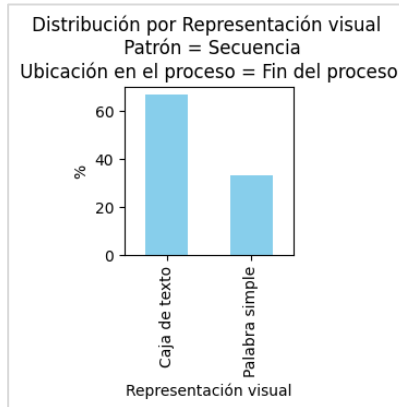


Figura 47. Frecuencias de representación visual de patrón 1

En la mayoría de las alertas (75%) se observó que se presentaron sin íconos, como se muestra en la Figura 48.

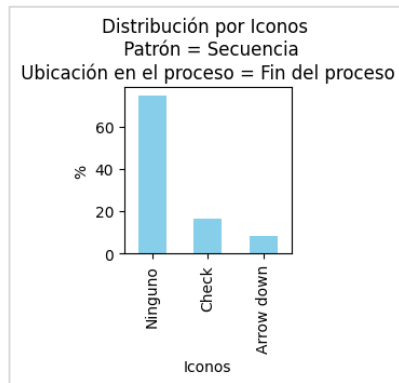


Figura 48. Frecuencia de íconos de patrón 1

Del mismo modo, la mayoría de las alertas se presentaron sin un color característico, como se muestra en la Figura 49.

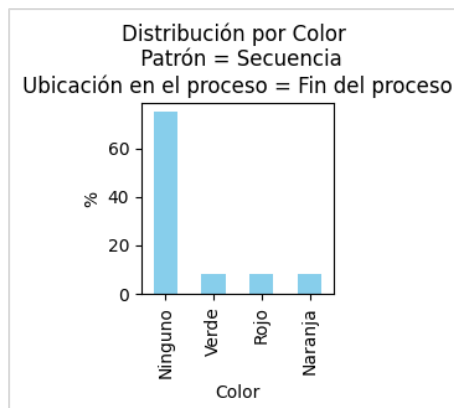



Figura 49. Frecuencias de color para alertas de patrón 1

En relación con el ícono y al color, sin ser identificado algún patrón o generalidad en las alertas presentadas, se establecerá el ícono check  y el color verde, en concordancia con la definición del uso de los colores presentada en el marco conceptual.

Es relevante mencionar que tanto los procesos con patrón de secuencia, como los procesos con los patrones de división/unión exclusiva y paralela definen este patrón (mensaje de confirmación en el final del proceso). Por lo tanto, este patrón se convierte en un patrón general para cualquier patrón de flujo.

La definición de la sugerencia para el patrón 1 con atributos es la siguiente: En todos los procesos debería mostrarse una alerta de confirmación de color verde en el final del proceso mediante una caja de texto, con el ícono check, título 'Terminado', sin acciones y con el contenido asociado al nombre del proceso con la estructura {sujeto} {verbo en pasado participio} {complemento}.

### 3.2.4.2. Patrón 2

En las instancias de proceso con un patrón de secuencia con riesgo de ejecución o idempotencia, se debería mostrar una alerta de usuario de 'Advertencia' o 'Feedforward' entre cada actividad. A continuación, se detallan los atributos que componen la alerta.

En cuanto al contenido de la alerta, según se muestra en la Figura 50, el 57.1% está relacionado a la acción o actividad actual, el 28.6% al proceso y el 14.3% al dominio.

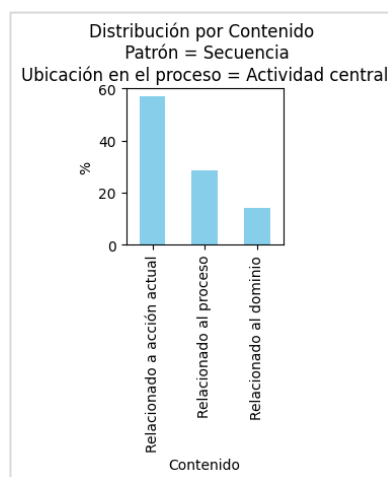


Figura 50. Frecuencia de contenidos para alertas de patrón 2

Para identificar un sentido entre el contenido, en la Tabla 7 se presentan ejemplos de los contenidos de las alertas asociadas al patrón.

Tabla 7. Contenido de alertas de advertencia o feedforward entre acciones

| Proceso                     | Contenido  |
|-----------------------------|--|
| Ganar chocolate             | ¿Quieres ver un video para ganar CHOCOLATE?  |
| Buscar conductor            | Precio recomendado {precio}; Tiempo de viaje ~{tiempo}   |
| Enviar plata                | Revisa bien el número para enviarle a la persona correcta  |
| Enviar plata                | Le vas a enviar plata al {número} {nombre}; Si te equivocas no podemos hacer nada desde Nequi  |
| Invitar amigo a competición | Después que comience la competición, tus estadísticas y datos de actividad se compartirán con tu amigo.  |
| Realizar lección            | ¡Buen trabajo!   |
| Crear orden                 | Realizaremos un cobro a tu método de pago por el valor de la orden. Si hay modificaciones, lo ajustaremos al saber el valor final de tu orden. |

Aunque las alertas presentadas no siguen un patrón definido en la estructura de su contenido, sino que cada una de ellas muestra datos relacionados a su proceso o dominio, el propósito de estas alertas se puede entender como una *invitación al usuario para que revise la información* o que *confirme la acción que está realizando*. Por lo tanto, en este caso, se propone un contenido fijo que abarque la connotación de advertencia y revisión. El texto elegido es 'Revisa los datos antes de continuar'

Según la Figura 51, el 57.1% de las alertas se presentaron sin título mientras que el 42.9% incluyó un título relacionado al proceso. No obstante, las alertas que presentaron títulos relacionados al proceso no pueden considerarse como patrón, ya que estos no contienen una estructura o semántica similar. Por lo tanto, el patrón seleccionado en este caso es mostrar una alerta sin título. Sin embargo, de acuerdo con el contexto del contenido, se sugiere el título sugerido '¿Estás seguro?' para mantener coherencia con el contenido sugerido anteriormente.

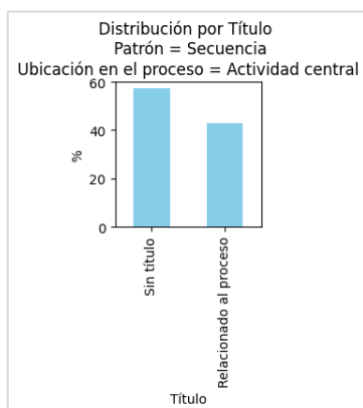


Figura 51. Frecuencia de títulos para alertas de patrón 2

Las acciones incluidas en estas alertas se detallan en la Figura 52, donde el 57.1% de las alertas contaron con acción de continuar y el 42.9% no incluyeron acciones. En este caso, dada la elección del título y contenido, se define la acción 'Continuar' como patrón para este tipo de alerta, agregando también, un botón de 'Cancelar'.

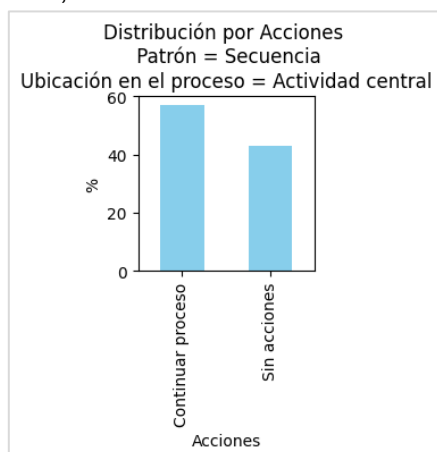


Figura 52. Frecuencia de acciones para alertas de patrón 2

La representación visual de este tipo de alertas fue en modo caja de texto en un 71.4% como se muestra en la Figura 53, por lo que se define como patrón para este tipo de alerta.



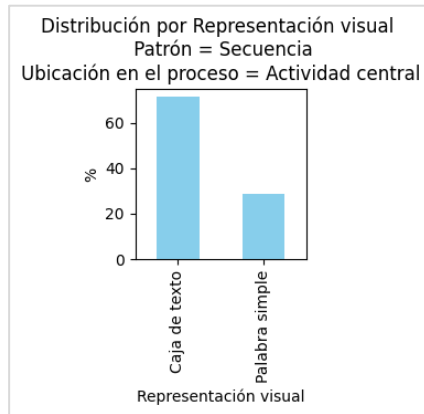


Figura 53. Frecuencia de tipo de alerta para alertas de patrón 2

El color de este tipo de alertas se distribuye como se muestra en la Figura 54, en donde la mayoría (71.4%) no se mostró con ningún color en específico. Sin embargo, siendo consecuente con el objetivo de las alertas y el contenido definido, se sugiere el color 'Naranja'.

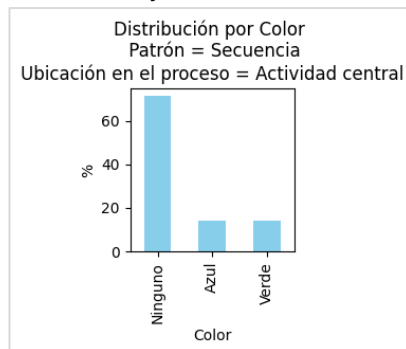


Figura 54. Frecuencia de colores para alertas de patrón 2

En cuando a los íconos, como se muestra en la Figura 55, el 57.1% de las alertas no presentaron íconos. No obstante, siguiendo con los lineamientos anteriores, se utilizará un ícono sugerido para reforzar la notación de la alerta, en este caso, el ícono de warning ⚠.

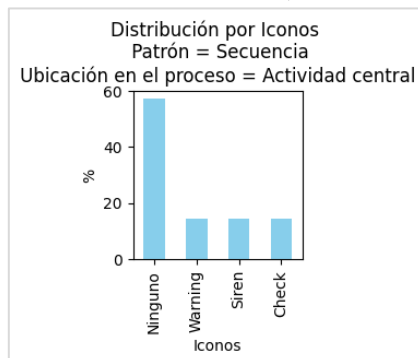


Figura 55. Frecuencia de iconos para alertas de patrón 2

La definición de la sugerencia del patrón 2 con atributos es la siguiente: En las instancias de proceso con un patrón de secuencia con riesgo de ejecución o idempotencia, se debería mostrar una alerta de usuario de 'Advertencia' entre cada actividad mediante una caja de texto, de color naranja, con el ícono warning, título '¿Estás seguro?', con las acciones 'Continuar' y 'Cancelar', y contenido 'Revisa los datos antes de continuar'.

### 3.2.4.3. Patrón 3

Los patrones de división/unión paralela deberían mostrar una alerta de *feedforward* en la compuerta de división. A continuación, se detalla el análisis de los atributos de este tipo de alertas.

En cuanto al contenido de estas alertas, el 83.3% relacionado a la acción esperada y el 16.7% al proceso como se muestra en la Figura 56.

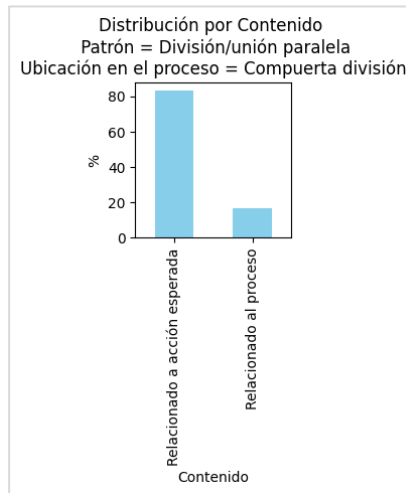


Figura 56. Frecuencia de contenido para alertas de patrón 3

Las alertas que presentaron contenido relacionado a la acción esperada se muestran en la Tabla 8

Tabla 8. Contenido relacionado a accione esperada de alertas de patrón 3

| Nombre del proceso   | Contenido  |
|----------------------|--|
| Unirse a una llamada | Te unirás a la llamada cuando alguien te permita entrar  |
| Reportar publicación | Se recibió el reporte: Tu reporte nos ayuda a mejorar nuestros procesos y contribuye a que Facebook siga siendo un entorno seguro para todos; Esperando revisión: Usamos tecnologías y equipos de revisión para eliminar el contenido que no cumple nuestras normas lo más rápido posible; Se tomó una decisión: Te enviaremos una notificación para que consultes el resultado en el buzón de ayuda lo antes posible. |
| Realizar compra      | Te avisaremos cuando tu compra esté en camino  |
| Realizar compra      | El día de la entrega te avisaremos en qué horario vamos a pasar por tu domicilio   |
| Realizar viaje       | {conductor} aceptó tu solicitud de {valor} y llegará en aproximadamente {tiempo}   |

De acuerdo con el proceso de cada una de estas alertas, se describen las actividades paralelas que deben completarse por parte de un rol distinto al usuario actual para que el proceso pueda continuar, informando al usuario sobre la condición de espera del proceso. En este caso el patrón definido para el contenido está asociado a un mensaje de espera más el nombre de las actividades que deben completarse fuera de la responsabilidad del usuario actual. La propuesta para el mensaje es la siguiente: 'El proceso continuará cuando se completen las siguientes actividades: {nombre actividad paralela 1}, {nombre actividad paralela 2} ...'.

En cuanto al título, estas alertas como se muestra en la Figura 57 mostraron en un 33.3% título relacionado al proceso, 33.3% sin título, 16.7% relacionado a la acción esperada y 16.7% relacionado a la acción realizada.

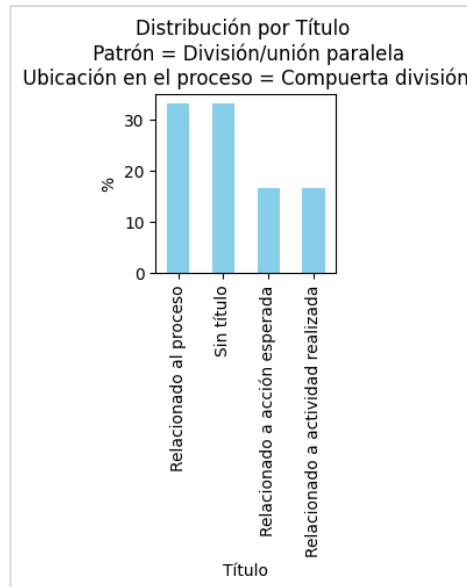


Figura 57. Frecuencia de títulos de alertas de patrón 3

En la Tabla 9 se muestran los títulos de las alertas que lo incluyeron, en esta no se puede denotar una estructura definida ya que cada título muestra información específica acorde al proceso o dominio, sin embargo, estos títulos en conjunto con los contenidos denotan la intención de manifestar el estado de espera del proceso, por lo que el título sugerido será 'En espera'.

Tabla 9. Título relacionado a acción esperada de alertas de patrón 3

| Nombre del proceso   | Título                        |
|----------------------|-------------------------------|
| Unirse a una llamada | Pidiendo permiso para acceder |
| Reportar publicación | Gracias, Recibimos tu reporte |
| Reportar publicación | Oculto                        |
| Realizar compra      | Llega el {fecha}              |

Como se muestra en la Figura 58, el 50% las alertas no presentaron acciones o presentaron acciones relacionadas a distintos ámbitos ligadas a aspectos del dominio, por lo que las acciones en este caso no constituyen un patrón y la alerta sugerida no contará con estas.

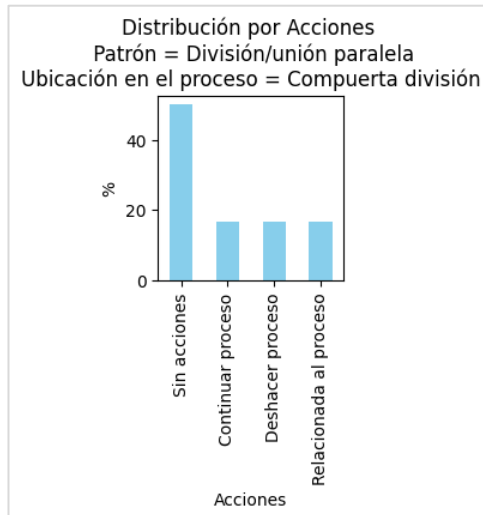


Figura 58. Frecuencias de acciones de alertas de patrón 3

La representación visual de las alertas de este tipo fue en un 100% caja de texto, por lo que es la representación elegida para la sugerencia del patrón.

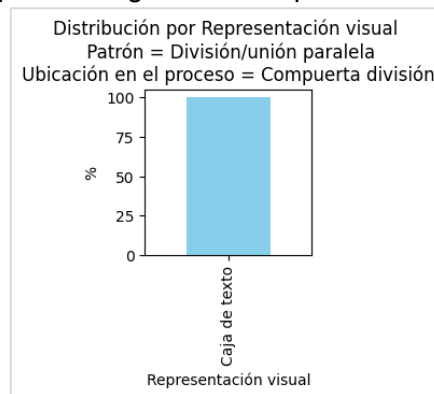


Figura 59. Frecuencia de representación visual de alertas de patrón 3

Las alertas de este tipo, en su mayoría (50%) no presentaron íconos, o presentaron diversos íconos como los que se muestran en la Figura 60. En este caso dado que el objetivo de la alerta de usuario es informar sobre el estado de espera del proceso, el ícono comúnmente aceptado para situaciones de espera es el de reloj de arena ⌚, el cual será usado para la sugerencia de este patrón.

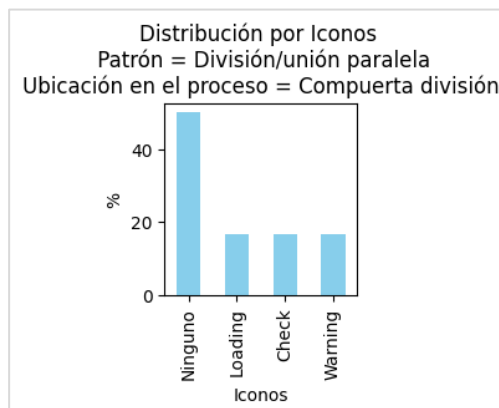


Figura 60. Frecuencia de íconos de alertas de patrón 3

El color de las alertas de este tipo como se muestra en la Figura 61, en un 75% se presentó si un color definido, y en un 12.5% tanto verde como naranja. La presencia de colores definidos en algunas alertas se debe a qué representa la división en el dominio o dentro del proceso, por lo que es algo contextual que no representa una generalidad. En este caso no se realiza una sugerencia de color, dado que el contexto del impacto del proceso espera puede ser variado de acuerdo con el contexto, y elegir un color específico puede no ser adecuado en todos los casos.

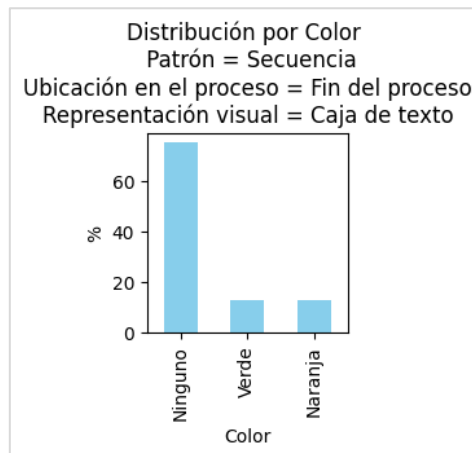


Figura 61. Frecuencia de color de alertas de patrón 3

La definición de la sugerencia del patrón 3 con atributos es la siguiente: Los patrones de división/unión paralela deberían mostrar una alerta de 'Feedforward' en la compuerta de división, representada mediante caja de texto, sin color, con el ícono de reloj de arena, título 'En espera', sin acciones y con el contenido siguiendo la estructura 'El proceso continuará cuando se completen las siguientes actividades: {nombre actividad paralela 1}, {nombre actividad paralela 2} ...'.

#### 3.2.4.4. Patrón 4

Los patrones de división/unión exclusiva deberían mostrar una alerta de *feedforward* en la compuerta de división.

El contenido de estas alertas está en un 50% relacionado al proceso, 25% a la acción que desencadena la decisión de la compuerta, 12.5% a consecuencia de la actividad o acción, 12.5% a una acción esperada y 12.5% otro como se muestra en la Figura 62.

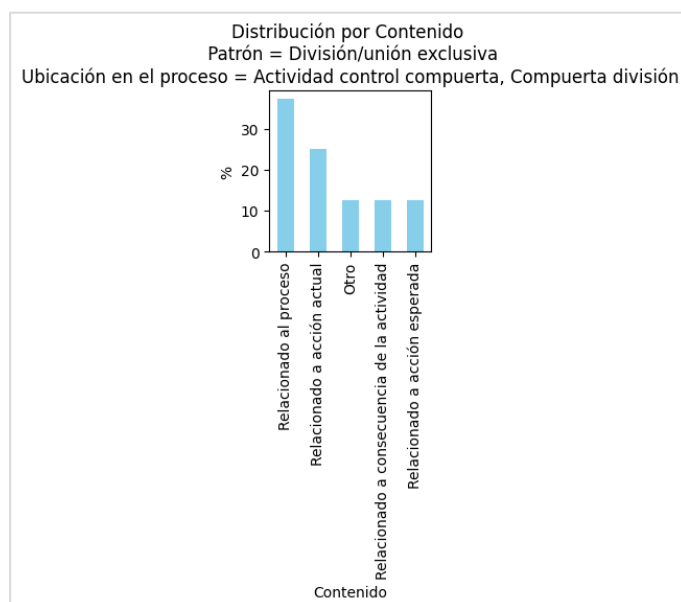


Figura 62. Frecuencia de contenido de alertas de patrón 4

En la Tabla 10 se muestra el contenido de las alertas, el cual en su mayoría corresponde a información del proceso la cual no se puede generalizar, además no cuenta con una estructura definida que pueda ser replicable.

Tabla 10. Contenido de alertas de patrón 4

| Proceso                       | Contenido   |
|-------------------------------|---|
| Unirse a una llamada          | {nombreUsuario}   |
| Compartir publicación         | Tu publicación se mostrará en el feed, en tu perfil y en los resultados de búsqueda; Los destinatarios predeterminados se establecieron en {predeterminado}, pero puedes cambiarlos para esta publicación concreta. |
| Enviar correo                 | Envío programado para {fecha}   |
| Iniciar una partida           | ¿Deseas continuar tu partida actual o recomenzar una nueva?   |
| Iniciar una partida           | ¿Seguro que quieres iniciar una nueva partida?  |
| Solicitar préstamo salvavidas | Con este seguro podrás tener la plata de tu préstamo de una.  |
| Solicitar préstamo salvavidas | Instrucciones para presentar otro seguro para respaldar tu préstamo   |
| Registrar caminata            | El tiempo de entrenamiento es demasiado corto. Esta sesión n o será grabada. ¿Finalizar de todos modos?   |

La intención detrás del contenido de estas alertas es informar sobre las posibles implicaciones de elegir uno u otro camino o preguntar sobre el camino a elegir, por lo que una sugerencia de contenido debería abarcar estos dos objetivos. Sin embargo, con la información que provee el proceso de negocio, evaluar las implicaciones de una decisión no es posible, ya que esto hace parte de las particularidades del negocio, por lo que en este caso el contenido sugerido estará orientado únicamente a la confirmación de la elección del usuario mediante el mensaje. 'Confirmar elección'.

Con respecto al título, en el 37.5% está relacionado al título o a la actividad actual y en el 25% de las alertas no se presentó título.

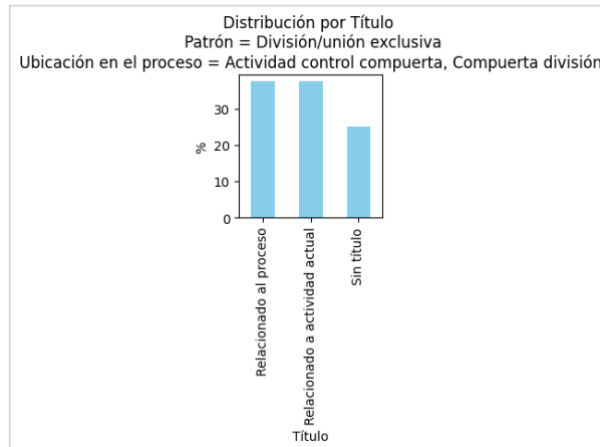


Figura 63. Frecuencia de título para alertas de patrón 4

En la Tabla 11 se muestran los títulos de las alertas, en este caso al igual que con el contenido, el título está relacionado a detalles del proceso que no son generalizables. Sin embargo, la intención es la misma que la asociada al contenido, por lo que no es necesario redundar con un título general y en este caso a la sugerencia no se le agregará título.

Tabla 11. Contenido de alertas de patrón 4

| Proceso                       | Título   |
|-------------------------------|--|
| Unirse a una llamada          | Alguien quiere unirse a esta llamada           |
| Compartir publicación         | ¿Quién puede ver tu publicación?               |
| Iniciar una partida           | ¿Continuar partida?                            |
| Iniciar una partida           | ¿Nueva partida?                                |
| Solicitar préstamo salvavidas | Seguro de vida con Nequi costo mensual {valor} |
| Solicitar préstamo salvavidas | Tengo otro seguro                              |

En el caso de las acciones, el 50% están relacionadas a continuar el proceso, el 37.5% a las salidas u opciones de la compuerta y el 12.5% a deshacer el proceso.

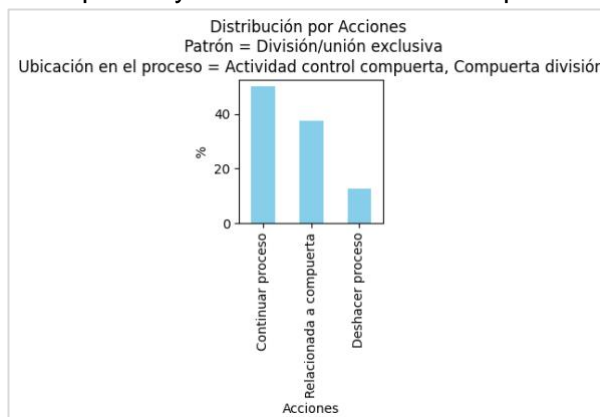


Figura 64. Frecuencia de acciones de alertas de tipo 4

En este caso, dada la intención con la que se ha generalizado este tipo de alerta, se elegirá la acción 'Continuar' como la generalidad agregando también la opción 'Cancelar', con el objetivo que el contenido sugerido sea coherente con las acciones.

La representación visual en un 100% corresponde a caja de texto como se muestra en la Figura 65, por lo que la sugerencia será de este tipo.

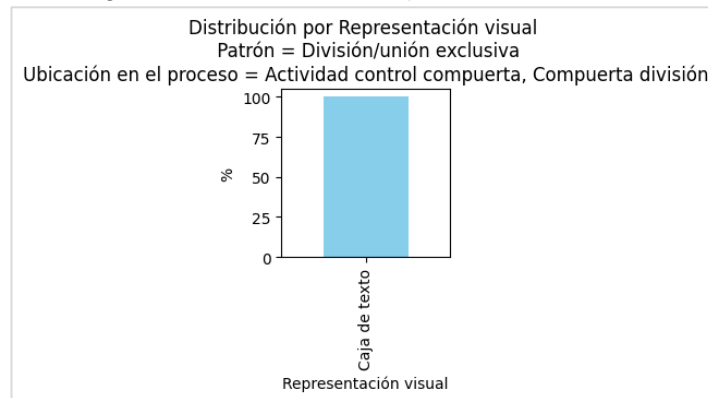


Figura 65. Frecuencia de representación visual de alertas de patrón 4

En cuanto a color, el 75% de las alertas no se presentó con color, esta generalidad se tomará como parte de la sugerencia, tomando en cuenta que los colores en este tipo de alertas pueden estar relacionados a particularidades del proceso o dominio que no se pueden generalizar.

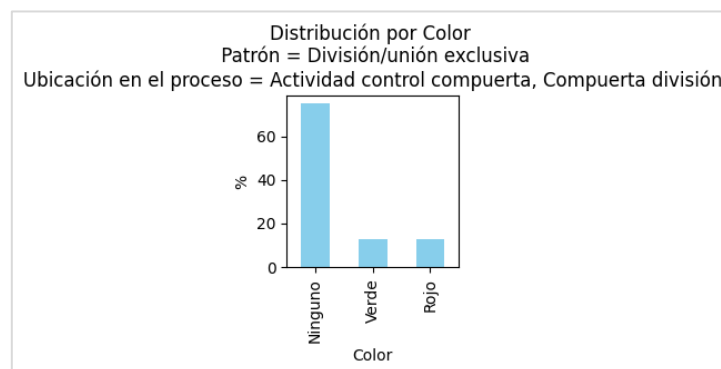


Figura 66. Frecuencia de color en alertas de patrón 4

Las alertas presentadas tampoco incluyeron ícono en su mayoría (75%), y otros tipos de íconos relacionados al proceso o dominio. En este caso la sugerencia podría agregar algún ícono para reforzar la confirmación de la acción, pero dado que el contenido y acciones definidas son lo suficientemente descriptivos, la tendencia de no mostrar ícono se agregará dentro de la sugerencia.

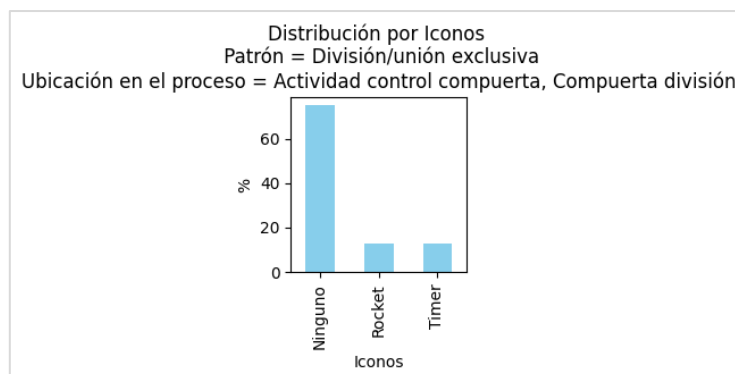


Figura 67. Frecuencia de ícono para alertas de patrón 4

La definición de la sugerencia del patrón 4 con atributos es la siguiente: Los patrones de división/unión exclusiva deberían mostrar una alerta de *feedforward* en la compuerta de



división, representada mediante caja de texto, sin color, sin ícono, sin título, con las acciones 'Continuar', 'Cancelar' y con el contenido 'Confirmar elección'.

## 4. Componente software para la generación dinámica de alertas de usuario

Con el fin de alcanzar el objetivo de *proponer una solución informática que permita generar alertas de usuario*, en esta sección se describirá el proceso de desarrollo software propuesto. Esta propuesta se fundamenta en el marco teórico caracterización y conclusiones obtenidas a partir de la exploración de alertas de usuario presentada en las secciones previas.

### 4.1. Análisis

Un componente de software es una unidad que expone sus servicios mediante interfaces y puede ser desplegado de manera independiente, es decir, se encuentra separado del entorno y de otros componentes, por lo tanto encapsula sus propias características, lo que lo convierte en una unidad de despliegue, es decir, que un componente siempre se despliega completo y nunca por partes; además un componente está sujeto a ser compuesto por terceros y no tiene un estado observable desde el exterior es decir que no puede ser diferenciado de copias de sí mismo (Szyperski et al., 2009). En otras palabras, un componente es una unidad modular que puede ser usada por otras unidades modulares que pueden usarlo mediante la exposición de sus servicios a través de interfaces (Lau & Di Cola, 2018).

De acuerdo con lo expuesto en la justificación del problema, el cambio constante en los flujos que definen el comportamiento de un software puede llegar a generar inconsistencias en las alertas de usuario presentadas en el sistema que a su vez pueden derivar en acciones incorrectas por parte de los usuarios. Una de las soluciones viables para abordar esta problemática consiste en la generación de alertas de usuario a través de un componente de software, el cual tome en cuenta el proceso de negocio y sus posibles variaciones en a lo largo del tiempo, generando sugerencias de alertas de usuario

Dicho lo anterior, la solución informática propuesta consiste en un componente de software para la generación de alertas de usuario (en adelante componente de alertas) cuyo funcionamiento será independiente del sistema para el cual las alertas son generadas (en adelante sistema objetivo), y su interacción se realizará mediante servicios expuestos por el componente de alertas los cuales serán consumidos por el sistema objetivo.

#### 4.1.1. Descripción del funcionamiento del componente

La definición de un proceso base que describa la interacción entre un sistema objetivo y el componente de alertas es fundamental para especificar la forma y los momentos precisos en los que se llevará a cabo dicha interacción. Esta definición permitirá estructurar de manera más efectiva los requisitos que serán documentados.

Para la definición del proceso base se utilizará el *método de Silver* aplicado al objetivo general del sistema software, partiendo del nivel de abstracción más alto el cual se denominará *proceso principal*, en donde se definirá el flujo general del componente.

#### 4.1.1.1. Proceso principal

Para la definición del proceso principal no se responderán las preguntas de la definición del alcance del proceso, debido a que estas están orientadas hacia procesos específicos que tienen inicio y final bien definidos, la definición completa se realizará para los desgloses de los procesos específicos.

#### Mapeo de alto nivel

El proceso principal se guiará por los siguientes pasos

- Un usuario registrará el sistema objetivo en el componente, con el fin que este pueda hacer uso de los servicios de generación dinámica de alertas de usuario.
- Un usuario registrará un flujo representado por un diagrama BPMN para el sistema objetivo ante el componente, para que el componente pueda realizar el mapeo del flujo y pueda analizarlo en el futuro.
- Cuando una instancia del proceso registrado se inicie en el sistema objetivo, este la registrará ante el componente.
- Cuando una instancia del proceso registrado cambie de estado en el sistema objetivo, este solicitará la generación dinámica de alertas de usuario al componente y las renderizará en su interfaz para mostrarlas al usuario.

#### Diagrama de proceso de alto nivel

En la Figura 68, se muestra el diagrama de proceso de alto nivel que describe la serie de pasos mencionados anteriormente.

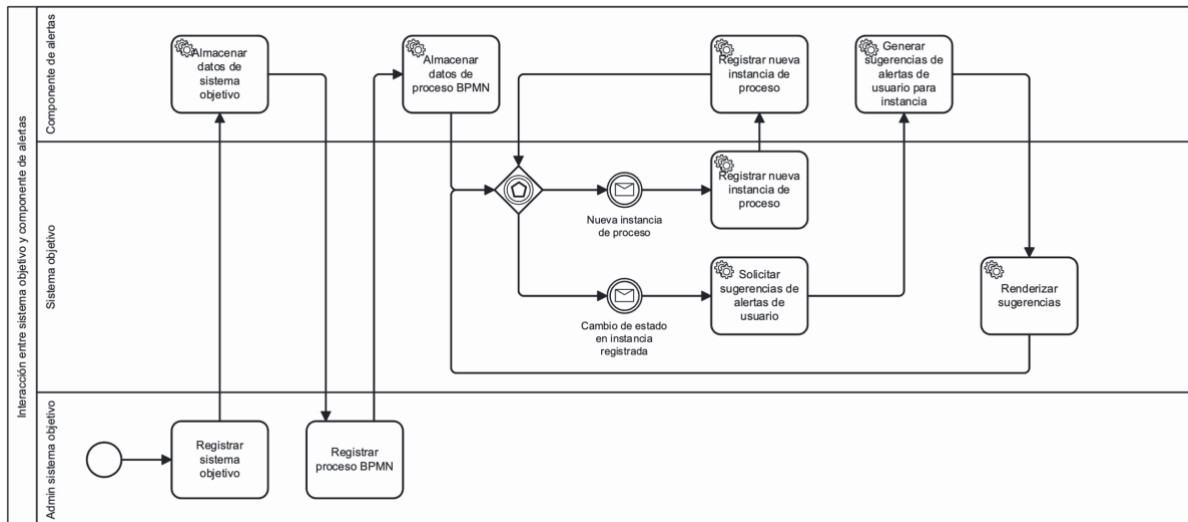


Figura 68. Proceso principal componente de alertas

Las actividades principales del componente incluyen el Almacenar sistema objetivo, Almacenar datos de proceso BPMN, Registrar nueva instancia de proceso y Generar sugerencias de alertas de usuario para una instancia, a continuación. Siguiendo la metodología de Silver se llevará a cabo un desglose detallado de cada una de estas actividades en subprocesos. Este desglose permitirá obtener una visión más completa y detallada de las actividades asociadas a cada uno de ellos.

#### 4.1.1.2. Subproceso almacenar datos de sistema objetivo

El objetivo de este subproceso será almacenar la información asociada a un sistema objetivo para que se pueda efectuar la comunicación entre este y el componente a través de los servicios expuestos.

##### Definición de alcance del proceso

- ¿Qué inicia el proceso? Cuando un usuario envía una solicitud de registro de un sistema objetivo ante el componente.
- ¿Qué determina que el proceso está completo? Cuando la información del sistema objetivo ha sido almacenada correctamente dentro del componente.
- ¿Qué representa cada instancia o ejecución del proceso? Un nuevo sistema objetivo que podrá registrar procesos BPMN de manera dinámica a través del componente.
- ¿Cuáles son las formas en que el proceso puede terminar?
  - El sistema objetivo es almacenado y se pueden registrar procesos e instancias de procesos para la generación dinámica de alertas de usuario.
  - El sistema objetivo no puede ser registrado debido a problemas en la validación de los datos.

##### Mapeo de alto nivel

El proceso de almacenar sistema objetivo seguirá los siguientes pasos

- Un usuario administrador del sistema objetivo envía una solicitud de registro del sistema objetivo ante el componente agregando los datos requeridos para la solicitud.
- El componente realiza la validación de información enviada.
- Si la información cumple con los requerimientos se almacena dentro de la estructura de datos del componente de la información del sistema objetivo y se retorna al usuario la confirmación del proceso junto con la información de conexión para las fases futuras.

##### Diagrama de proceso de alto nivel

El diagrama de flujo BPMN que representa las actividades requeridas para llevar a cabo el subproceso de *almacenar datos de sistema objetivo* se muestra en la Figura 69.

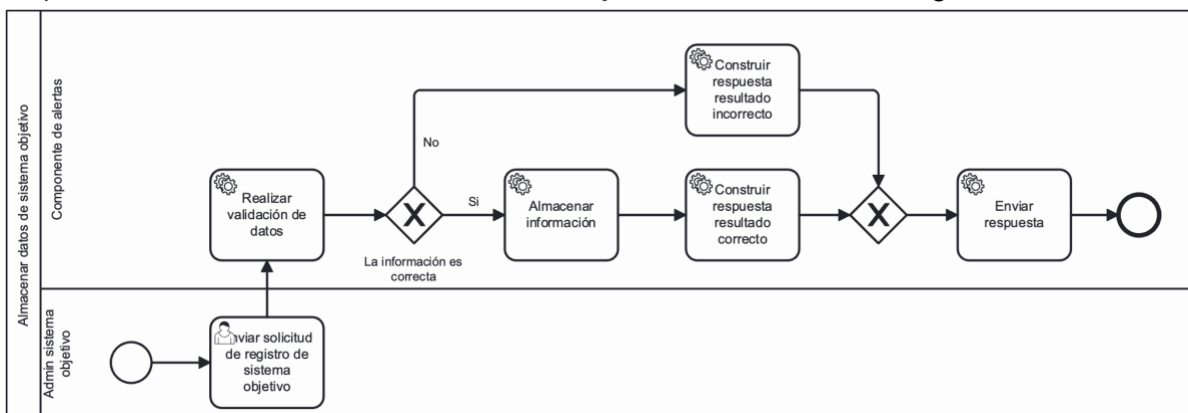


Figura 69. Proceso almacenar sistema objetivo

#### 4.1.1.3. Subproceso almacenar datos de proceso BPMN

El objetivo de este subproceso es almacenar la información de un proceso del sistema objetivo en la estructura de datos del componente. Esto se logra mediante la lectura de un archivo en formato *.bpmn* enviado en la petición. Es esencial que el enviado debe cumplir con una

estructura estándar (que se definirá en la sección de requisitos) para considerarse como válido. Una vez que los datos del proceso han sido almacenados, el componente estará en la capacidad de analizar dinámicamente el flujo para la generación de alertas de usuario en pasos futuros.

### Definición de alcance del proceso

- ¿Qué inicia el proceso? Cuando un usuario administrador envía una petición de registro de proceso ante el componente.
- ¿Qué determina que el proceso está completo? Cuando los datos del proceso hayan sido almacenados en el componente
- ¿Qué representa cada instancia o ejecución del proceso? Un proceso registrado ante el componente para el sistema objetivo, sobre el que se podrán registrar instancias.
- ¿Cuáles son las formas en que el proceso puede terminar?
  - La información y estructura del proceso es correcta y este ha sido almacenado correctamente en la estructura de datos del componente.
  - La información o estructura del proceso no es correcta y el proceso no puede ser almacenado

### Mapeo de alto nivel

El proceso de registrar instancia de proceso seguirá los siguientes pasos:

- Un administrador del sistema objetivo envía la petición de registro de proceso ante el componente
- El componente valida los datos y el archivo .bpmn enviado
- Si los datos son correctos se almacena la información asociada al proceso en la estructura de datos del componente y se construye una respuesta de éxito.
- Si los datos no son correctos se construye la respuesta de error.

### Diagrama de proceso de alto nivel

En la Figura 70 se muestran el desglose de las actividades del subproceso de registrar instancia de proceso.

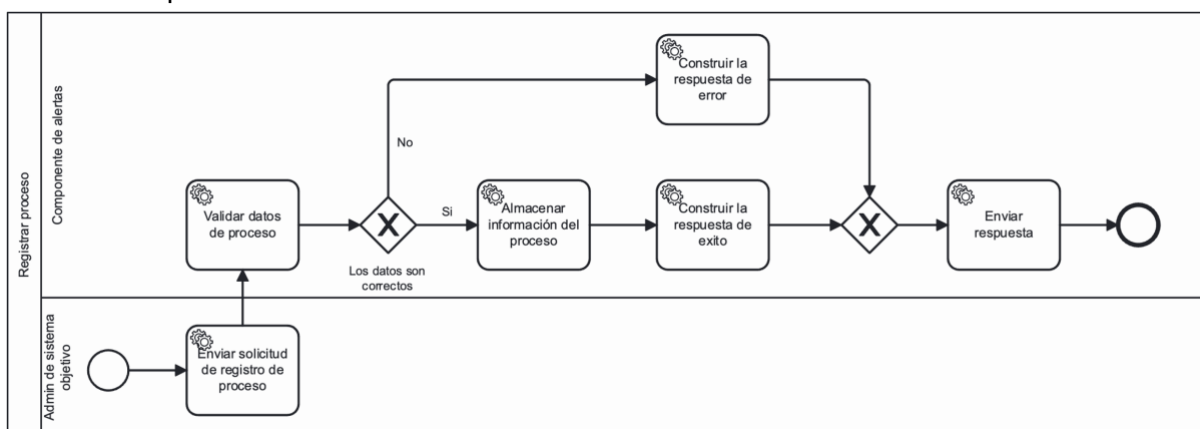


Figura 70. Diagrama de alto nivel para subproceso almacenar datos de sistema objetivo

#### 4.1.1.4. Subproceso registrar instancia de proceso

Este proceso tiene como objetivo registrar la creación de una instancia de un proceso iniciado en el sistema objetivo ante el componente de alertas, para que el componente de alertas

pueda generar alertas sobre esta instancia en específico mediante el almacenamiento de rastro de actividades por la que la instancia atravesará durante su ejecución.

### Definición de alcance del proceso

- ¿Qué inicia el proceso? Cuando el sistema objetivo envía una solicitud hacia el componente solicitando el registro de una nueva instancia iniciada para un proceso previamente registrado.
- ¿Qué determina que el proceso está completo? Cuando la instancia se ha registrado correctamente en el componente de alertas.
- ¿Qué representa cada instancia o ejecución del proceso? Una instancia registrada en el componente sobre la cual se podrán generar alertas de usuario de manera dinámica.
- ¿Cuáles son las formas en que el proceso puede terminar?
  - La instancia se ha validado y guardado correctamente.
  - La instancia no se pudo almacenar debido a errores en la validación de los datos.

### Mapeo de alto nivel

El proceso de registrar instancia de proceso seguirá los siguientes pasos:

- El sistema objetivo envía la solicitud de registro de la instancia al componente de alertas.
- El componente de alertas valida los datos enviados.
- Si los datos son correctos se almacena la información de la instancia y se construye una respuesta con los datos de la instancia.
- Si los datos no son correctos se construye la respuesta de error.

### Diagrama de proceso de alto nivel

El proceso descrito en el mapeo de alto nivel se corresponde al de la Figura 71.

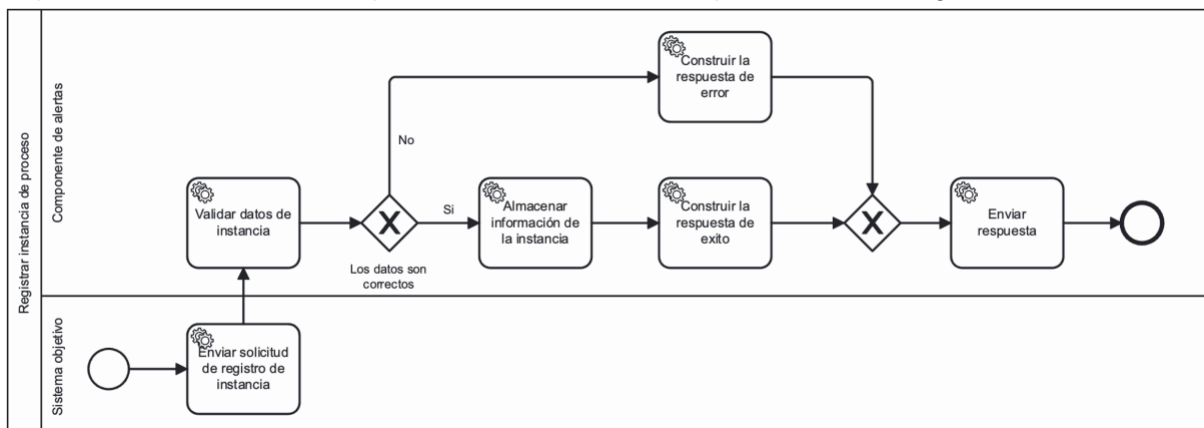


Figura 71. Diagrama de proceso de alto nivel subproceso registrar instancia de proceso

#### 4.1.1.5. Subproceso generar sugerencias de alertas de usuario para instancia

El objetivo de este subproceso será generar una o varias alertas de usuario a partir de la lectura del proceso del sistema objetivo con respecto a la ubicación de la instancia y de acuerdo con la definición de patrones de alerta de usuario y la caracterización de alertas realizada en fases previas.

### Definición de alcance del proceso

- ¿Qué inicia el proceso? Cuando el sistema objetivo envía una solicitud hacia el componente indicando el cambio de estado de una instancia.
- ¿Qué determina que el proceso está completo? Cuando el sistema objetivo renderiza y muestra en su interfaz las alertas de usuario generadas por el componente.
- ¿Qué representa cada instancia o ejecución del proceso? Una o varias alertas de usuario generada en base al proceso del sistema objetivo de acuerdo con el estado de la instancia.
- ¿Cuáles son las formas en que el proceso puede terminar?
  - Con una o varias alertas de usuario sugeridas enviadas de vuelta al sistema objetivo.
  - Sin alertas sugeridas.
  - Con error en la validación de la actualización de estado de la instancia

### **Mapeo de alto nivel**

El proceso de registrar instancia de proceso seguirá los siguientes pasos:

- El sistema objetivo envía la solicitud de actualización de estado de una instancia ante el componente de alertas.
- El componente de alertas valida la información de la actualización.
- Si los datos son correctos
  - El componente genera un registro de historial de la instancia almacenando el nuevo estado.
  - El componente valida si en el estado actual existe una alerta sugerida, de acuerdo con los patrones del proceso asociado a la instancia.
  - El componente construye la información de la(s) alerta(s) sugeridas, incluyendo contenido, tipo y representación visual de manera obligatoria, e icono, color, título y acciones de forma opcional de acuerdo con el patrón identificado.
  - El componente construye la respuesta de éxito agregando la información de las alertas sugeridas
- Si los datos no son correctos se construye una respuesta de error
- Se envían la respuesta al sistema objetivo
- Si hay alertas sugeridas el sistema objetivo renderiza la información de las alertas y las muestra en su interfaz de acuerdo con sus lineamientos de UI.

### **Diagrama de proceso de alto nivel**

Los pasos mencionados anteriormente se representan en el diagrama de la Figura 72.

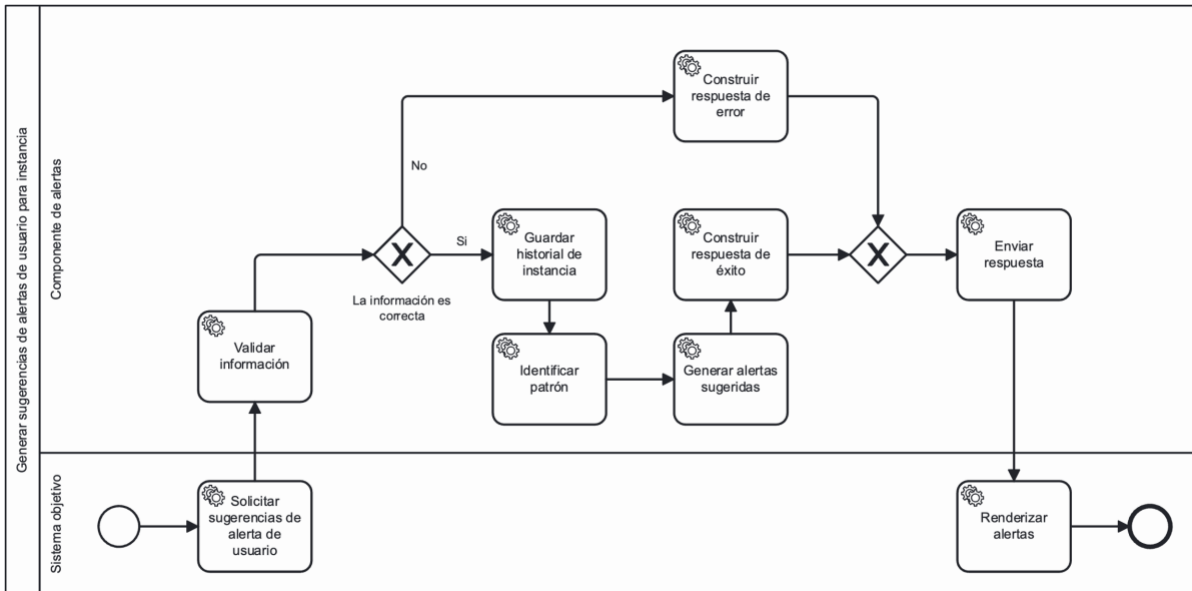


Figura 72. Proceso generar alerta de usuario

#### 4.1.2. Pautas para la definición de requisitos

Las tareas creadas se organizaron en un tablero *Kanban* diseñado para la organización y supervisión de las tareas requeridas, mediante el uso de secciones y reglas. Dentro del tablero, cada tarea representará una acción específica que contribuyen a lograr el objetivo del proceso base del componente.

##### 4.1.2.1. Atributos de un requisito

Cada requisito documentado deberá contar con los siguientes atributos:

- **Identificador único:** Se asignará a cada tarea un identificador con un prefijo constante que determine que la tarea pertenece al componente de alertas (CDA) y un número consecutivo de tres dígitos asociado al orden de inclusión de las tareas. De este modo las tareas tendrán un nombre con la estructura CDA-XXX donde 'CDA-' será una constante y XXX la numeración asociada (por ejemplo, CDA-001, CDA-002, etc.).
- **Nombre de la tarea:** Un nombre corto que resuma la descripción de la tarea.
- **Descripción del objetivo de la tarea:** Se especificará el aporte de la tarea en particular en relación con el objetivo general del proceso base del componente.
- **Condiciones de validación:** Se detallarán condiciones medibles (cualitativa o cuantitativamente), que permitan verificar que la tarea está completada.
- **Dependencias con otras tareas:** Se indicarán las relaciones de la tarea con otras tareas, detallando el identificador de la tarea de la cual depende.
- **Adjuntos o información adicional:** Se adjuntarán documentos o información complementaria que pueda ampliar la comprensión del objetivo de la tarea o de las condiciones de validación.
- **Subtareas:** En caso que la tarea original pueda dividirse en tareas más pequeñas, cada subtarea mantendrá las mismas características descritas previamente. El identificador de una subtarea consistirá en el identificador de la tarea padre seguido por un consecutivo de dos dígitos separados por punto (por ejemplo, si la tarea padre CDA-003 las subtareas serán CDA-003.01 y CDA-003.02, etc.).



Una tarea que esté dividida en subtareas podrá definir sus condiciones de validación de dos maneras:

- **Complejidad de subtareas:** En este caso no será obligatorio definir las condiciones de validación de la tarea padre, en su lugar la condición de validación de dicha tarea se tomará como la complejidad de las subtareas asociadas.
- **Validación general:** En este caso se agregarán las condiciones de validación sobre la tarea padre, y se tomarán como completas todas las subtareas si la tarea padre se valida correctamente.

#### 4.1.2.2. Definición de secciones de Kanban

Las secciones del tablero representarán el estado del progreso de la tarea. Las secciones y el movimiento de las tareas entre estas, las secciones y reglas del tablero serán las siguientes:

- **Sección 'En documentación':** Agrupará las tareas que se encuentran en proceso de documentación de sus características.
  - Regla 1: Una vez la tarea cumple con las características requeridas, se podrá trasladar a la sección 'Pendientes'
- **Sección 'Pendientes':** Aquí se ubicarán las tareas cuya documentación está completada y se encuentran a la espera de iniciar su ejecución.
  - Regla 2: Si se inicia el desarrollo de una tarea, esta deberá trasladarse a la sección 'En progreso'
- **Sección 'En progreso':** Contiene las tareas cuyo desarrollo está en curso.
  - Regla 3: Si el desarrollo está terminado, es decir, se han abordado todos los escenarios descritos en las condiciones de validación y no hay dependencias pendientes por desarrollar, la tarea se moverá a la sección 'En pruebas'.
  - Regla 4: Si el desarrollo se ha completado, pero hay dependencias pendientes, la tarea se trasladará a la sección 'Pendiente dependencias'.
- **Sección 'Pendiente dependencias':** Agrupa las tareas terminadas que se no pueden pasar a fase de pruebas debido a dependencias no completadas.
  - Regla 5: Si la última dependencia de la tarea ya ha sido finalizada y enviada a la sección 'En pruebas', la tarea también deberá moverse a la sección 'En pruebas'.
- **Sección 'En pruebas':** Indica que una tarea y sus posibles dependencias ya han completado la fase de implementación y sus condiciones de validación están listas para su verificación.
  - Regla 6: Si todas las condiciones de validación son aprobadas, la tarea se moverá a la sección 'Listo' junto con la información de las pruebas realizadas.
  - Regla 7: Si una o más condiciones de validación no son aprobadas, la tarea se moverá a la sección 'Pendientes' junto con la descripción del escenario fallido.
- **Sección 'Listo':** Indica que una tarea ha sido finalizada y validada mediante la verificación de sus condiciones de validación.
  - Regla 8: Si se requiere algún ajuste en la descripción de la tarea, esta deberá regresar a la sección 'En documentación' para realizar los ajustes necesarios.

#### 4.1.2.3. Elección de aplicación de gestión de Kanban

Actualmente existen varias herramientas que facilitan la gestión de tareas mediante tableros tipo *Kanban*. En este proyecto de investigación, se ha elegido Asana como herramienta de gestión de *Kanban* por las siguientes razones:

- Personalización: Asana permite la gestión de secciones del tablero, lo cual es esencial dado que las secciones definidas anteriormente no son un estándar, sino que han sido definidas exclusivamente para el contexto de esta investigación.
- Automatización de reglas: Asana permite configurar reglas de propósito general dentro de las secciones. Esto posibilita la automatización de procesos de validación que garanticen el cumplimiento de las reglas de traslado entre secciones definidas anteriormente.
- Gestión de comentarios: Asana permite agregar comentarios, los cuales permiten agregar información entre iteraciones, útiles para llevar una trazabilidad de la tarea a lo largo de su desarrollo.
- Gestión de subtareas: Asana permite la creación de subtareas en múltiples niveles, ofreciendo una jerarquía estructurada. Esto permite desglosar tareas complejas en subtareas más granulares.
- Control de dependencias: Asana de manera nativa agrega el control de dependencias entre tareas, lo cual permite cumplir con reglas definidas anteriormente.

#### 4.1.3. Glosario de términos

Antes de continuar con la definición de los requisitos del sistema, es importante tener en cuenta los términos que se usarán en esta y en fases siguientes, para tener claridad a qué se hacen referencia cuando estos sean mencionados:

- Alerta de usuario: Una alerta de usuario es un mensaje mostrado por un sistema al usuario con el objetivo de informar o alertar una situación.
- Sistema objetivo: Un sistema externo que hace uso del componente y para el cual el componente realiza la generación de alertas de usuario.
- Proceso de negocio: La definición del flujo de trabajo del sistema objetivo.
- Instancia de proceso: La ejecución de un proceso de negocio que se está llevando a cabo dentro del sistema objetivo.
- Atributos de alerta de usuario: Son los componentes que definen una alerta de usuario entre los que se incluyen
- Tipo: La abstracción general del propósito de la alerta, este tipo puede ser confirmación, advertencia, error, *feedback*, *feedforward*.
- Color: El color que resalta dentro de la alerta el cual puede estar relacionado con la intención de esta.
- Icono: La imagen que se utiliza para reforzar el propósito de una alerta.
- Título: El mensaje diferenciado de la alerta que proporciona un contexto sobre el contenido.
- Contenido: La especificación de la alerta que contiene el mensaje textual que se intenta informar.
- Acciones: Referente a las acciones contextuales que acompañan una alerta de usuario, generalmente a modo de botones.
- Caso de uso: Una unidad lógica de programación con una responsabilidad única.

#### 4.1.4. Especificación de requisitos

Siguiendo los lineamientos definidos se realiza la especificación de requisitos del sistema. Este proceso estará guiado por el proceso base del componente y su desglose correspondiente, donde cada subproceso desglosado del proceso base se agregará como tarea principal, mientras que las actividades de cada uno de ellos se detallan como subtareas.

Para exponer sus servicios, el componente implementará un API REST que recibirá y responderá peticiones en formato JSON usando el estándar HTTP. Cada subproceso se transformará en un endpoint dentro del API, definiendo tanto los parámetros de entrada como la estructura específica de los datos retornados.

##### 4.1.4.1. CDA-001 Generador de respuestas

Dado que todos los servicios expuestos por el componente a través del API REST generan respuestas que pueden variar entre correctas e incorrectas según las validaciones asociadas a cada servicio, se creará un estándar de las respuestas para todos los servicios expuestos por el componente, dicho estándar de respuesta deberá contar con los datos mostrados en la Tabla 12.

Tabla 12. Atributos de respuesta estándar del componente

|                |   |  |
|----------------|---|--|
| <b>status</b>  | Una colección de atributos que permiten analizar el resultado de la petición. Los atributos incluidos se describen abajo. |  |
|                | <b>success</b>  | Atributo booleano donde 'true' indica que el proceso se ejecutó correctamente y 'false' denota algún tipo de error.  |
|                | <b>statusCode</b>   | Código de respuesta estándar HTTP que puede variar según el resultado.<br>200: Cuando la ejecución del servicio finaliza correctamente<br>201: En las peticiones de creación de recursos, indica que el recurso fue creado correctamente.<br>400: Cuando los datos JSON enviados no puedan ser interpretados por el componente.<br>404: Cuando los identificadores enviados no correspondan a ningún registro almacenado en el componente.<br>422: En las peticiones de creación de recursos, cuando los datos incumplan con los requisitos del servicio.<br>500: Cuando se presente un error desconocido en el procesamiento. |
|                | <b>warnings</b>   | Una colección de advertencias textuales relacionadas a la ejecución del servicio, no constituyen un error.   |
|                | <b>errors</b>   | Una colección con los errores presentados en la ejecución del servicio solicitado.   |
| <b>message</b> | El mensaje general de la respuesta  |  |
| <b>data</b>    | Una colección con los datos asociados al servicio consultado.   |  |

La respuesta generada será una respuesta con el estándar HTTP, en la cual el código de respuesta será equivalente al atributo *status.statusCode*.

#### Condiciones de validación

Para validar la implementación de esta tarea se deberá comprobar los siguientes ítems:

- Debe existir un caso de uso ejecutable que permita generar las respuestas HTTP del componente con la estructura mencionada anteriormente.

- Dicho caso de uso deberá contar con dos funciones, una para crear respuestas de éxito, y otra para crear respuestas de error.
- La función de crear respuestas de éxito deberá contar con los parámetros statusCode, message, data y warnings (opcional), y su ejecución deberá generar una respuesta con los datos enviados y además status.errors vacío y status.sucess true.
- La función de crear respuesta de error deberá contar con los parámetros statusCode, message, errors y warnings (opcional), y su ejecución deberá generar una respuesta con los datos enviados y además data vacío y status.sucess false.
- Sólo se podrá crear respuestas de éxito con los statusCode 200 y 201.
- Sólo se podrá crear respuestas de error con los statusCode 400, 404, 422 y 500.

#### 4.1.4.2. CDA-002 Validador de peticiones

Tomando en cuenta, que todas las peticiones que se realicen hacia el componente deberán ser validadas para garantizar que los datos enviados cuenten con la información requerida para cada servicio expuesto, se debe crear un validador de peticiones general que permita evaluar las restricciones que defina cada endpoint o servicio expuesto en el componente de alertas. Dada la sintaxis del estándar JSON, cada parámetro enviado contará con una clave y un valor para los cuales se definen las restricciones de la Tabla 13.

Tabla 13. Validaciones sobre peticiones al API REST

|   |  |
|---|--|
| <b>Obligatorio</b>                            | Indica que el parámetro debe estar presente en la solicitud del servicio, en caso que no esté presente se generará un error con el mensaje<br>El parámetro <clave> es obligatorio.   |
| <b>Alfanumérico</b>                           | El parámetro podrá enviarse como la combinación de cualquier letra número o símbolo, si esta condición no se cumple se generará un error con el mensaje<br>El parámetro <clave> debe ser alfanumérico.   |
| <b>Array &lt;claves&gt;</b>                   | El parámetro se deberá enviar como un array clave valor con las claves obligatorias enviadas, en caso de que alguna clave no se encuentre entre los valores se generará un error con el mensaje.<br>El parámetro <clave parámetro> debe incluir la clave <clave array> |
| <b>Longitud máxima &lt;valor longitud&gt;</b> | Para los parámetros alfanuméricos, indica la cantidad máxima de caracteres que son permitidos, en caso de excederse se generará un error con el mensaje.<br>El valor <valor> del parámetro <clave> excede la longitud máxima de <valor longitud>                       |
| <b>Archivo adjunto</b>                        | El parámetro debe ser un archivo adjunto, en caso que no corresponda a un archivo se generará un error con el mensaje<br>El parámetro <clave> debe ser un archivo adjunto  |
| <b>Extensión &lt;valor extensión&gt;</b>      | Para los parámetros de archivo adjunto, indica la extensión que es permitida para el archivo, en caso de no cumplirse se generará un error con el mensaje<br>El archivo <clave> debe ser de extensión <valor extensión>  |

#### Condiciones de validación

Para validar la implementación de esta tarea se deberá comprobar los siguientes ítems:

- Debe existir un caso de uso ejecutable que permita leer los datos asociados a una petición y validar las restricciones.
- Dicho caso de uso debe recibir los parámetros petición, restricciones y retornar los errores asociados.

#### 4.1.4.3. CDA-003 Almacenar sistema objetivo

Esta tarea estará basada en el subproceso almacenar datos de sistema objetivo, el cual se convertirá en un servicio expuesto por el componente que podrá ser consumido para crear nuevos sistemas objetivo. Este servicio deberá ser accesible mediante una petición **POST** a la url **/targetSystem** del componente. Los datos de entrada son los que se especifican en la Tabla 14.

Tabla 14. Datos de entrada servicio almacenar sistema objetivo

|             |  |
|-------------|--|
| <b>name</b> | Corresponde al nombre con el que se almacenará el sistema objetivo en el componente. |
|             | Obligatorio, Alfanumérico  |

La respuesta correcta de este servicio tendrá la estructura de la Tabla 15 más el código de respuesta HTTP 201 indicando la creación del sistema objetivo.

Tabla 15. Respuesta correcta de servicio almacenar sistema objetivo

|                            |  |
|----------------------------|--|
| <b>target_system_token</b> | El token asociado al sistema objetivo, con el que se podrán realizar las consultas en los servicios en donde este sea requerido. |
| <b>created_at</b>          | La fecha y hora en la que se creó el sistema objetivo  |

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 2.

#### 4.1.4.4. CDA-004 Obtener datos de sistema objetivo

Esta tarea consistirá en la creación de un servicio expuesto en el API REST del componente de alertas, el cual permitirá obtener los datos asociados a un sistema objetivo previamente creado. Este servicio se podrá acceder mediante la petición **GET** en la url **/targetSystem** y cuyos parámetros se definen en la Tabla 16.

Tabla 16. Datos de entrada servicio obtener datos de sistema objetivo

|                            |   |
|----------------------------|---|
| <b>target_system_token</b> | Corresponde al token del sistema objetivo del cual se desea obtener información |
|                            | Obligatorio, Alfanumérico   |

La respuesta correcta de este servicio tendrá la estructura de la Tabla 17 más el código de respuesta HTTP 200 indicando el éxito en la operación.

Tabla 17. Respuesta resultado correcto servicio obtener datos de sistema objetivo

|                            |  |  |
|----------------------------|--|--|
| <b>name</b>                | El nombre del sistema objetivo   |  |
| <b>target_system_token</b> | El token asociado al sistema objetivo, con el que se podrán realizar las consultas en los servicios en donde este sea requerido. |  |
| <b>created_at</b>          | La fecha y hora en la que se creó el sistema objetivo  |  |
| <b>processes</b>           | Una colección con los procesos que el sistema objetivo tiene asociados, cada proceso contará con los siguientes datos            |  |
|                            | <b>process_name</b>  | El nombre del proceso  |
|                            | <b>process_token</b>   | El token asociado al proceso, con el que se podrán realizar las consultas en los servicios en donde este sea requerido |
|                            | <b>process_version</b>   | La versión del proceso   |
|                            | <b>created_at</b>  | La fecha y hora en la que se creó el proceso   |

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en las secciones 1 y 3.

#### 4.1.4.5. CDA-005 Almacenar datos de proceso BPMN

Esta tarea consistirá en la creación de un servicio expuesto en el API REST del componente de alertas, el cual permitirá obtener los datos asociados a un sistema objetivo previamente creado. Este servicio se podrá acceder mediante la petición **POST** en la url **/process** y cuyos parámetros se definen en la Tabla 18.

Para almacenar un proceso BPMN es necesario contar con la referencia del sistema objetivo al cual se asociará. Además, si el proceso que se desea almacenar corresponde a la evolución de un proceso almacenado anteriormente, se deberá incluir el token del proceso que se evolucionará. Por otro lado, de acuerdo con la fase de caracterización, en algunos casos, estos procesos pueden presentar riesgo de ejecución o de idempotencia, pudiendo afectar o tocar aspectos económicos o aspectos fundamentales del dominio. También pueden ser o no idempotentes, lo que determina si un proceso puede ejecutarse de manera repetitiva sin generar consecuencias (no idempotente), o si por el contrario cada ejecución acarreará consecuencias en el dominio del proceso o del sistema objetivo (idempotentes). Finalmente, para poder generar correctamente el contenido de las alertas, el nombre del proceso se deberá enviarse diferenciando el verbo y el complemento.

Tabla 18. Parámetros para el servicio almacenar proceso BPMN

|                            |  |  |
|----------------------------|--|--|
| <b>process_name</b>        | Los datos asociados al nombre del proceso, con los que se podrá generar el contenido de algunas alertas  |  |
|                            | Array con las claves verb, subject y complement  |  |
|                            | <b>verb</b>  | El verbo asociado al proceso en infinitivo |
|                            |  | Obligatorio, Alfanumérico                  |
|                            | <b>subject</b>   | El sujeto asociado al proceso              |
|                            |  | Obligatorio, Alfanumérico                  |
| <b>complement</b>          | El complemento del nombre  |  |
|                            | Obligatorio, Alfanumérico  |  |
| <b>target_system_token</b> | Corresponde al token del sistema objetivo al cual se le asociará el nuevo proceso  |  |
|                            | Obligatorio, Alfanumérico  |  |
| <b>process_token</b>       | Si el proceso que se va a almacenar corresponde a la evolución de un proceso actualmente asociado al sistema objetivo  |  |
|                            | Opcional, Alfanumérico   |  |
| <b>process_bpmn</b>        | El archivo en formato BPMN correspondiente al proceso que se va a agregar  |  |
|                            | Obligatorio, Archivo adjunto   |  |
| <b>risky_process</b>       | Corresponde a la consideración de riesgo del proceso dentro del negocio, es decir si de acuerdo con el dominio la ejecución de este proceso debería ser cuidadosa debido a sus posibles consecuencias. |  |
|                            | Obligatorio, Booleano  |  |
| <b>idempotent_process</b>  | Corresponde a la consideración de idempotencia del proceso, es decir si el proceso puede ser ejecutado en repetidas ocasiones sin riesgos colaterales.   |  |
|                            | Obligatorio, Booleano.   |  |

Para que un archivo enviado al servicio de almacenar datos de proceso BPMN sea considerado válido, debe cumplir con ciertas características. Estas condiciones están diseñadas para asegurar que el proceso sea adecuado para la generación de alertas de usuario, de acuerdo con los patrones identificados en la caracterización.

- Elementos permitidos: Dentro del conjunto de patrones elegido en la fase de caracterización, los elementos permitidos serán aquellos que abarquen los procesos abstraídos y analizados. Estos elementos incluirán:
- Eventos: Serán permitidos los eventos de inicio y de fin.
- Lanes: La descripción de estos será obligatoria y cada uno de ellos corresponderá al nombre del rol de usuario que ejecuta una tarea específica.
- Actividades
- Compuertas: compuertas de tipo Exclusivo y Paralelo.
- Todos los elementos deben pertenecer a un Lane
- El formato del archivo .bpmn deberá seguir la estructura estándar de un archivo BPMN en formato XML. Esto asegurará la coherencia y compatibilidad con las especificaciones del lenguaje BPMN, permitiendo una correcta interpretación y procesamiento de los datos por parte del componente de alertas.

La respuesta correcta de este servicio tendrá la estructura de la Tabla 19 más el código de respuesta HTTP 201 indicando la creación del proceso asociado al sistema objetivo.

Tabla 19. Respuesta resultado correcto servicio almacenar datos proceso BPMN

|                      |   |
|----------------------|---|
| <b>process_token</b> | El token asociado al proceso, con el que se podrán realizar las consultas en los servicios en donde este sea requerido. |
| <b>process_name</b>  | El nombre del proceso creado  |
| <b>created_at</b>    | La fecha y hora en la que se creó el sistema objetivo   |

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 2.

#### 4.1.4.5.1. CDA-005.01 Leer proceso de negocio

Esta tarea tiene como objetivo leer la lectura de un proceso BPMN mediante la interpretación de un archivo .bpmn en el formato XML estándar para abstraer los elementos de este y que puedan ser almacenados en la estructura de datos del componente.

Dentro de un archivo XML es posible utilizar el *XML Path Language* o *xpath* para navegar y consultar datos, con el objetivo de obtener conjuntos de nodos que cumplan con condiciones específicas. En este caso se utilizará el *xpath* para obtener información sobre los elementos dentro del archivo *bpmn*

**Eventos:** Los eventos se podrán hallar mediante la búsqueda de los nodos dentro del XML que coincidan con los *xpath*

- *//bpmn:startEvent* Para obtener los eventos de inicio
- *//bpmn:endEvent* Para obtener los eventos de fin

**Compuertas:** Las compuertas se buscarán dentro del archivo mediante el uso de los siguientes *xpath*:

- *//bpmn:exclusiveGateway* Para obtener compuertas exclusivas
- *//bpmn:parallelGateway* Para obtener compuertas paralelas

**Actividades:** Las actividades se podrán identificar dentro del archivo mediante el uso de los siguientes *xpath*:

- *//bpmn:task* Para actividades que no han sido etiquetadas
- *//bpmn:serviceTask* Para actividades etiquetadas como de sistema
- *//bpmn:userTask* Para actividades etiquetadas como de usuario

**Roles de usuario:** Los roles de usuario serán obtenidas a partir de los Lanes, los cuales a su vez serán identificados mediante el xpath:

- //bpmn:lane

Si se ejecuta el xpath bpmn:flowNodeRef sobre un nodo de rol de usuario, se podrán obtener todos los nodos de referencia, en los cuales se relacionan los id de los elementos que pertenecen al Lane, o en este caso al rol de usuario.

Para cada uno de los nodos XML abstraídos mediante los xpath, se podrán obtener los siguientes datos:

- Id: El identificador es un dato alfanumérico que se agrega a los elementos con el fin de establecer sus relaciones con otros elementos, el cual es obtenido a través del atributo id de los nodos seleccionados.
- Nombre: el nombre del elemento corresponde al valor del nodo
- Tipo: El tipo de elemento se obtiene mediante el nombre de la etiqueta obtenida mediante el xpath, siendo este el valor que va después de los dos puntos en el xpath.

La determinación de las asociaciones entrantes y salientes de cada elemento se pueden obtener mediante el xpath //bpmn:sequenceFlow, el cual obtiene nodos de asociación con los atributos:

- id: El identificador de la conexión
- sourceRef: El id del elemento donde sale la conexión
- targetRef: El id de elemento de donde llega la conexión
- name: El nombre de la conexión (si existe)

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 4.

#### 4.1.4.6. CDA-006 Obtener datos de proceso

Esta tarea consistirá en la creación de un servicio expuesto en el API REST del componente de alertas, el cual permitirá obtener los datos asociados a un proceso de un sistema objetivo previamente creado. Este servicio se podrá acceder mediante la petición **GET** en la url **/process** y cuyos parámetros se definen en la Tabla 20.

*Tabla 20. Datos de entrada servicio obtener datos de proceso*

|                            |   |
|----------------------------|---|
| <b>target_system_token</b> | Corresponde al token del sistema objetivo al cual pertenece el proceso que se desea obtener |
|                            | Obligatorio, Alfanumérico   |
| <b>process_token</b>       | Corresponde al token del proceso que se desea consultar                                     |
|                            | Obligatorio, Alfanumérico   |

La respuesta correcta de este servicio tendrá la estructura de la Tabla 21 más el código de respuesta HTTP 200 indicando el éxito del proceso.

*Tabla 21. Respuesta resultado correcto servicio obtener datos de sistema objetivo*

|                      |   |
|----------------------|---|
| <b>name</b>          | El nombre del proceso   |
| <b>target_system</b> | El nombre del sistema objetivo al que está asociado el proceso  |
| <b>process_token</b> | El token asociado al proceso, con el que se podrán realizar las consultas en los servicios en donde este sea requerido. |
| <b>created_at</b>    | La fecha y hora en la que se creó el sistema objetivo   |



|                   |   |  |
|-------------------|---|--|
| <b>elements</b>   | Una colección con los elementos creados en el proceso, cada entrada contará con los datos |  |
|                   | <b>id</b>   | El id del elemento dentro del XML del proceso bpmn   |
|                   | <b>name</b>   | El nombre del elemento   |
|                   | <b>type</b>   | El tipo de elemento (evento, actividad o compuerta)  |
|                   | <b>subtype</b>  | En los elementos evento y compuerta determinará el tipo (inicio, fin para eventos y exclusiva, paralela para compuertas) |
| <b>user_roles</b> | Una colección con los roles de usuario del proceso, cada rol tendrá los datos             |  |
|                   | <b>name</b>   | El nombre del rol de usuario asignado desde el Lane  |

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 3.

#### 4.1.4.7. CDA-007 Registrar nueva instancia de proceso

Esta tarea estará basada en el subproceso registrar nueva instancia de proceso, el cual se convertirá en un servicio expuesto por el componente que podrá ser consumido para crear nuevos sistemas objetivo. Este servicio deberá ser accesible mediante una petición **POST** a la url **/instance** del componente. Los datos de entrada son los que se especifican en la Tabla 22.

Tabla 22. Datos de entrada servicio registrar instancia de proceso

|                            |   |
|----------------------------|---|
| <b>target_system_token</b> | Corresponde al token del sistema objetivo al cual pertenece el proceso sobre el que se registrará la nueva instancia. |
|                            | Obligatorio, Alfanumérico   |
| <b>process_token</b>       | Corresponde al token del proceso sobre el que se registrará la nueva instancia  |
|                            | Obligatorio, Alfanumérico   |

La respuesta correcta de este servicio tendrá la estructura de la Tabla 23 más el código de respuesta HTTP 201 indicando la creación de la instancia asociada al sistema objetivo para el proceso indicado.

Tabla 23. Respuesta correcta de servicio registrar instancia de proceso

|                               |   |
|-------------------------------|---|
| <b>process_instance_token</b> | El token asociado a la instancia registrada, el cual podrá usarse en los servicios donde sea requerido. |
| <b>created_at</b>             | La fecha y hora en la que se creó el sistema objetivo.  |

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 2.

#### 4.1.4.8. CDA-008 Obtener datos de instancia

Esta tarea consistirá en la creación de un servicio expuesto en el API REST del componente de alertas, el cual permitirá obtener los datos asociados a una instancia previamente creada. Este servicio se podrá acceder mediante la petición **GET** en la url **/instance** y cuyos parámetros se definen en la Tabla 24.

Tabla 24. Datos de entrada servicio obtener datos de instancia

|                               |   |
|-------------------------------|---|
| <b>target_system_token</b>    | Corresponde al token del sistema objetivo del cual se desea obtener información |
|                               | Obligatorio, Alfanumérico   |
| <b>process_instance_token</b> | Corresponde al token de la instancia de la cual se desea obtener información    |

|  |                           |
|--|---------------------------|
|  | Obligatorio, Alfanumérico |
|--|---------------------------|

La respuesta correcta de este servicio tendrá la estructura de la Tabla 25 más el código de respuesta HTTP 200 indicando el éxito del proceso.

Tabla 25. Respuesta resultado correcto servicio obtener datos de instancia

|                               |  |   |
|-------------------------------|--|---|
| <b>process_instance_token</b> | El token asociado la instancia obtenida, con el que se podrán realizar las consultas en los servicios en donde este sea requerido. |   |
| <b>created_at</b>             | La fecha y hora en la que se creó la instancia   |   |
| <b>current_status</b>         | Una colección con los datos de la instancia  |   |
|                               | <b>current_element</b>   | El id del elemento del proceso en donde se encuentra la instancia actualmente     |
|                               | <b>current_element_name</b>  | El nombre del elemento del proceso en donde se encuentra la instancia actualmente |
| <b>instance_history</b>       | Una colección con todos los estados por los que ha atravesado la instancia, cada una de las entradas se creará con los datos       |   |
|                               | <b>element_id</b>  | El id del elemento del proceso en donde se encuentra la instancia actualmente     |
|                               | <b>element_name</b>  | El nombre del elemento del proceso en donde se encuentra la instancia actualmente |
|                               | <b>created_at</b>  | La fecha y hora en donde se envió la instancia al elemento del proceso            |

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 3.

#### 4.1.4.9. CDA-009 Actualizar instancia y generar alertas sugeridas

Esta tarea estará basada en el subproceso generar sugerencias de alertas de usuario para instancia, el cual se convertirá en un servicio expuesto por el componente. Este servicio deberá ser accesible mediante una petición **PUT** a la url **/instance** del componente. Los datos de entrada son los que se especifican en la Tabla 22.

Tabla 26. Datos de entrada servicio registrar instancia de proceso

|                            |  |
|----------------------------|--|
| <b>target_system_token</b> | Corresponde al token del sistema objetivo al cual pertenece el proceso sobre el que se registrará la nueva instancia.              |
|                            | Obligatorio, Alfanumérico  |
| <b>instance_token</b>      | Corresponde al token del proceso sobre el que se registrará la nueva instancia   |
|                            | Obligatorio, Alfanumérico  |
| <b>next_element_id</b>     | Corresponde al identificador del elemento en el cual se encuentra la instancia y sobre el cual se generarán las alertas de usuario |
|                            | Obligatorio, Alfanumérico  |

La generación de alertas sugeridas se ejecutará en el componente tomando como referencia un punto específico en el proceso al cual está asociada una instancia, por esa razón es requerido que se envíe el id del elemento del proceso en el cual se efectuará el cálculo de alertas. Este id debe estar asociado a un elemento dentro del proceso, en caso de no estar asociado se deberá generar un error indicando que 'El id del elemento enviado no se encuentra dentro del proceso asociado a la instancia.'

Si los datos enviados son válidos, se generará un registro en el historial de la instancia, agregando el nuevo estado al historial, además se realizará la generación de alertas de usuario para la instancia, teniendo en cuenta el punto del proceso y los estados anteriores por los que ha atravesado la instancia.

La respuesta correcta de este servicio tendrá la estructura de la Tabla 27 más el código de respuesta HTTP 200 indicando el éxito del proceso.

Tabla 27. Respuesta correcta de servicio generar sugerencias de alertas de usuario para instancia

|                               |   |   |
|-------------------------------|---|---|
| <b>process_instance_token</b> | El token asociado a la instancia sobre las que se están generando las sugerencias   |   |
| <b>suggested_user_alerts</b>  | Una colección con las alertas de usuario sugeridas para la instancia, cada alerta sugerida deberá contar con los siguientes datos |   |
|                               | <b>visual_representation</b>  | El tipo de representación visual de acuerdo con las definiciones del marco teórico, que puede ser <i>caja de texto</i> o <i>palabra simple</i> . En este caso se enviarán como <i>box</i> o <i>simple_word</i>  |
|                               | <b>type</b>   | El tipo de alerta de usuario, de acuerdo con las definiciones del marco teórico, el tipo podrá ser <i>confirmación</i> , <i>advertencia</i> , <i>error</i> , <i>feedback</i> , <i>feedforward</i> o <i>notificación</i> . En este caso el tipo se enviará en inglés, siendo sus equivalentes <i>confirmation</i> , <i>warning</i> , <i>error</i> , <i>feedback</i> , <i>feedforward</i> , <i>notification</i> . |
|                               | <b>color</b>  | El color principal de la alerta, con las opciones definidas en el marco teórico <i>rojo</i> , <i>amarillo</i> , <i>verde</i> , <i>azul</i> . En este caso el color se enviará en inglés, siendo sus equivalentes <i>red</i> , <i>yellow</i> , <i>green</i> o <i>blue</i><br>Este atributo podría no ser enviado en caso de que no se sugiera algún color.   |
|                               | <b>title</b>  | El título sugerido de la alerta, este atributo puede o no ser enviado.  |
|                               | <b>content</b>  | El contenido sugerido de la alerta.   |
|                               | <b>icon</b>   | El ícono sugerido para la alerta  |
|                               | <b>actions</b>  | Una colección con acciones sugeridas para la alerta. Estas sugerencias podrán o no incluirse dependiendo de la determinación de la sugerencia.  |

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 3.

#### 4.1.4.9.1. CDA-009.01 Identificador de patrón de flujo y estado actual

Esta tarea tiene el objetivo de crear un caso de uso que permita identificar patrones en la ejecución de la instancia de un proceso, esto se logrará mediante el análisis del historial de elementos por los que ha atravesado una instancia, además de la determinación del estado actual de la instancia.

De acuerdo con los patrones que han sido analizados en la fase de caracterización y de los elementos permitidos en los procesos almacenados en el componente, la determinación de patrón de flujo del proceso deberá ser uno entre *secuencia*, *división* y *unión paralela*, *división*

y *unión exclusivas*. De acuerdo con la definición de cada uno la determinación de patrón de flujo se realizará como la que se muestra en la Tabla 28.

Tabla 28. Definición de condiciones de patrones de flujo

| Patrón de flujo    | Condiciones  |
|--------------------|--|
| Secuencia          | El proceso tiene solo actividades y eventos de inicio y final                |
| División exclusiva | El proceso tiene al menos una compuerta exclusiva con al menos dos salidas.  |
| División paralela  | El proceso tiene al menos una compuerta paralela con al menos dos salidas.   |
| Unión exclusiva    | El proceso tiene al menos una compuerta exclusiva con al menos dos entradas. |
| Unión paralela     | El proceso tiene al menos una compuerta paralela con al menos dos entradas.  |

Adicionalmente la determinación del estado actual de la instancia se determinará según las condiciones expuestas en la caracterización de alertas, las cuales pueden incluir *actividad inicial*, *actividad central*, *actividad control compuerta*, *actividad final*, *actividad paralela*, *compuerta de división*, *compuerta de unión*, *fin del proceso*.

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 4.

#### 4.1.4.9.2. CDA-009.02 Generar alertas de usuario sugeridas

El objetivo de esta tarea será crear un caso de uso que permita la generación de alertas sugeridas de acuerdo con las sugerencias para los patrones de alerta obtenidos en la caracterización. Recibiendo como parámetros una instancia, el patrón de flujo del proceso de la instancia, y el punto actual en el proceso obtenido mediante el CDA-009.01 Identificador de patrón de flujo y estado actual, generando la sugerencia de alerta acorde al patrón de flujo y punto del proceso como se muestra en la Tabla 29

Tabla 29. Definición de alertas sugeridas en el componente

| Patrón de flujo  | Punto del proceso     | Alerta sugerida  |
|--|-----------------------|--|
| Cualquier patrón de flujo                                  | Fin del proceso       | tipo: confirmación<br>representación: caja de texto<br>color: verde<br>icono: check<br>título: terminado<br>contenido: De acuerdo con el título del proceso {sujeto} {verbo en pasado participio} {complemento}<br>acciones: ninguna |
| Patrón secuencia, no idempotente o con riesgo de ejecución | Entre cada actividad  | tipo: advertencia<br>representación: caja de texto<br>color: naranja<br>icono: warning<br>título: ¿Estás seguro?<br>Contenido: Revisa los datos antes de continuar<br>acciones: Continuar, Cancelar                                  |
| División/unión paralela                                    | Compuerta de división | tipo: <i>feedforward</i><br>representación: caja de texto<br>color: ninguno<br>icono: sand-watch<br>título: En espera  |

|                          |                       |  |
|--------------------------|-----------------------|--|
|                          |                       | contenido: El proceso continuará cuando se completen las siguientes actividades: {actividad paralela 1}, {actividad paralela 2}...<br>acciones: sin acciones                         |
| División/unión exclusiva | Compuerta de división | tipo: <i>feedforward</i><br>representación: caja de texto<br>color: ninguno<br>icono: ninguno<br>título: ninguno<br>contenido: Confirmar elección<br>acciones: Continuar', 'Cancelar |

La(s) alerta(s) de usuario generadas, se almacenarán en la estructura de datos del componente, con la relación del punto del proceso donde estas fueron generadas.

La validación de esta tarea deberá seguir los lineamientos del Anexo 2 en la secciones 1 y 4.

#### 4.1.2. Organización de tareas en tablero de Kanban

Las tareas descritas, se organizaron en un tablero de Kanban dentro de la herramienta elegida Asana, dentro de la cual se realizará la gestión para la fase de construcción, las tareas junto con su descripción se agregan dentro de la columna 'Pendientes' ya que cuentan con la documentación requerida para poder ser abordadas como se muestra en la Figura 73.

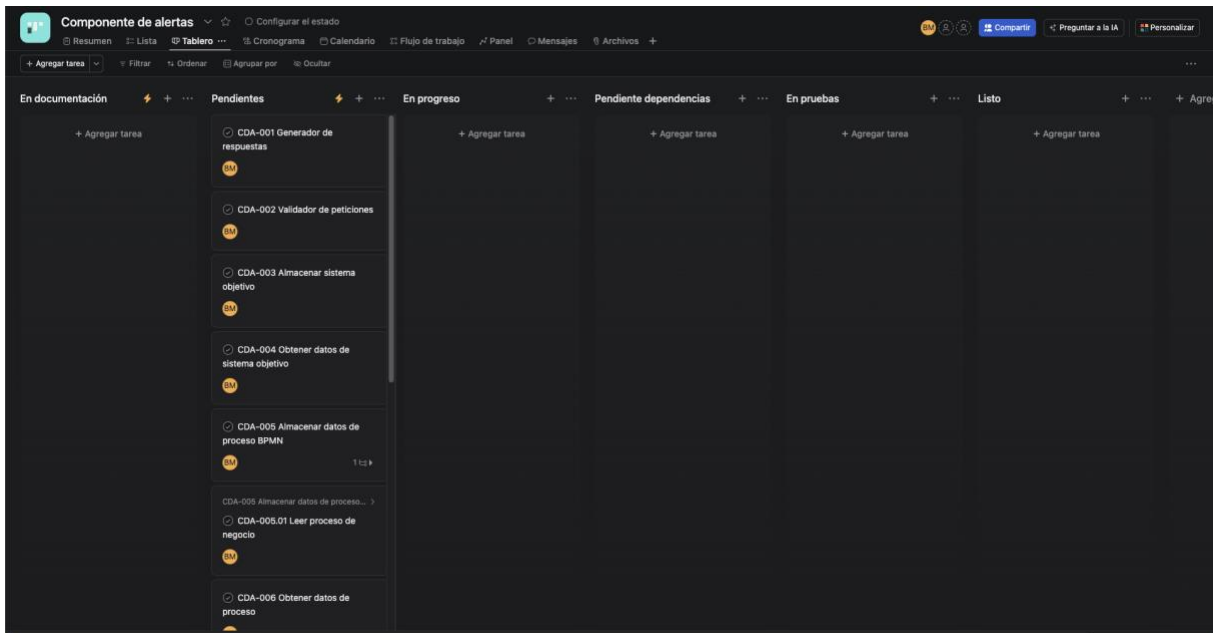


Figura 73. Organización de tareas en tablero Kanban en Asana

## 4.2. Diseño

Tomando como base los requerimientos definidos en la fase anterior, en esta fase se realizará la definición del diseño del componente, el cual servirá como base para la fase de construcción o desarrollo de este, añadiendo subtareas con detalles técnicos a las creadas en la fase anterior.

### 4.2.1. Arquitectura del componente

Dado que se requiere que el componente de alertas opere como un módulo independiente del sistema objetivo, la arquitectura que implementará es la arquitectura *cliente – servidor*. El componente de alertas será quien desempeñe el papel de sistema servidor, encargado de llevar a cabo las operaciones de registro y generación de alertas de usuario. Por otra parte, el sistema objetivo actuará como cliente, realizando las solicitudes necesarias al componente a través de internet como se muestra en la Figura 74.

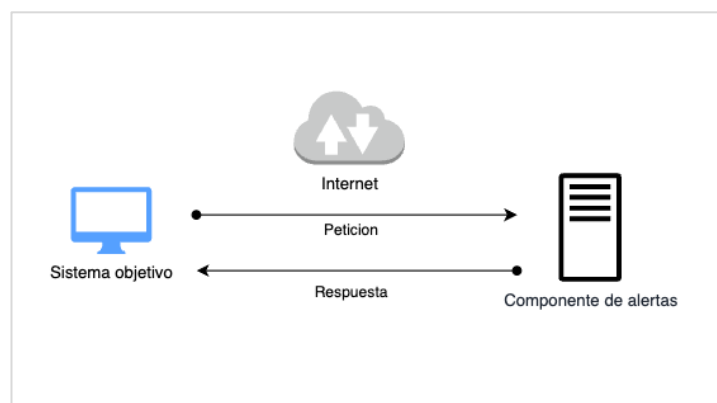


Figura 74. Diagrama de arquitectura del componente

Internamente el componente funcionará mediante dos componentes, *API* y *base de datos*. El componente *API* será el encargado del procesamiento de la lógica del componente, lo cual incluye la recepción de peticiones y envío de respuestas hacia los sistemas objetivo, por otro lado, la base de datos será la encargada de la persistencia de la información asociada al componente como se muestra en la Figura 75.

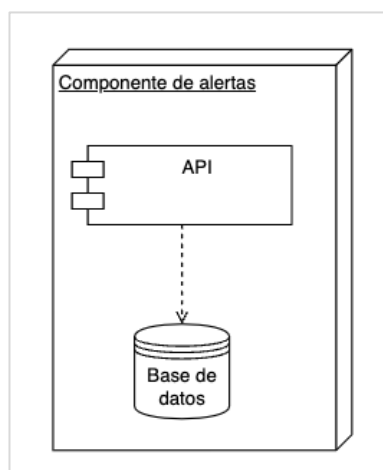


Figura 75. Diagrama de componentes

### 4.2.2. Dominio del componente

Para ampliar la comprensión del dominio del componente de alertas, se genera el dominio del componente como se muestra en la Figura 76, en la cual la entidad principal es **Sistema objetivo**, la cual representa ante el componente un sistema externo que puede generar **Alertas de usuario** para las **Instancias de proceso** que se ejecutan dentro de este. El

componente para cada instancia almacenará su **Historial** de estados (**Elementos del proceso**) por los que ha atravesado para poder tomar las determinaciones necesarias al momento de generar las alertas, de acuerdo con los requerimientos.

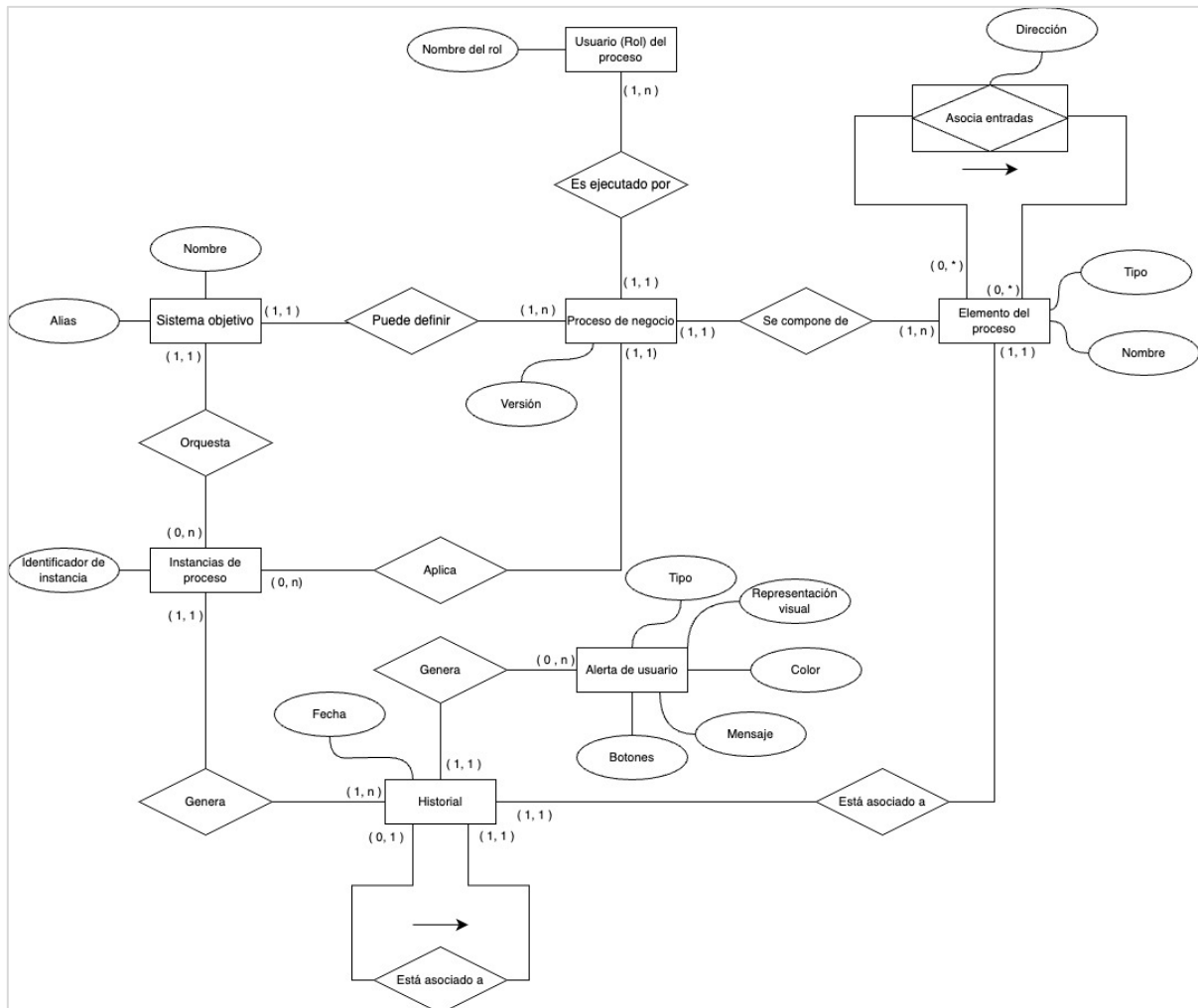


Figura 76. Diagrama entidad - relación del componente

#### 4.2.3. Diagrama de datos

Tomando como base el modelo entidad-relación presentado en el punto anterior, se genera el diagrama relacional, el cual representa el modelo lógico de base de datos del componente como se muestra en la Figura 77.

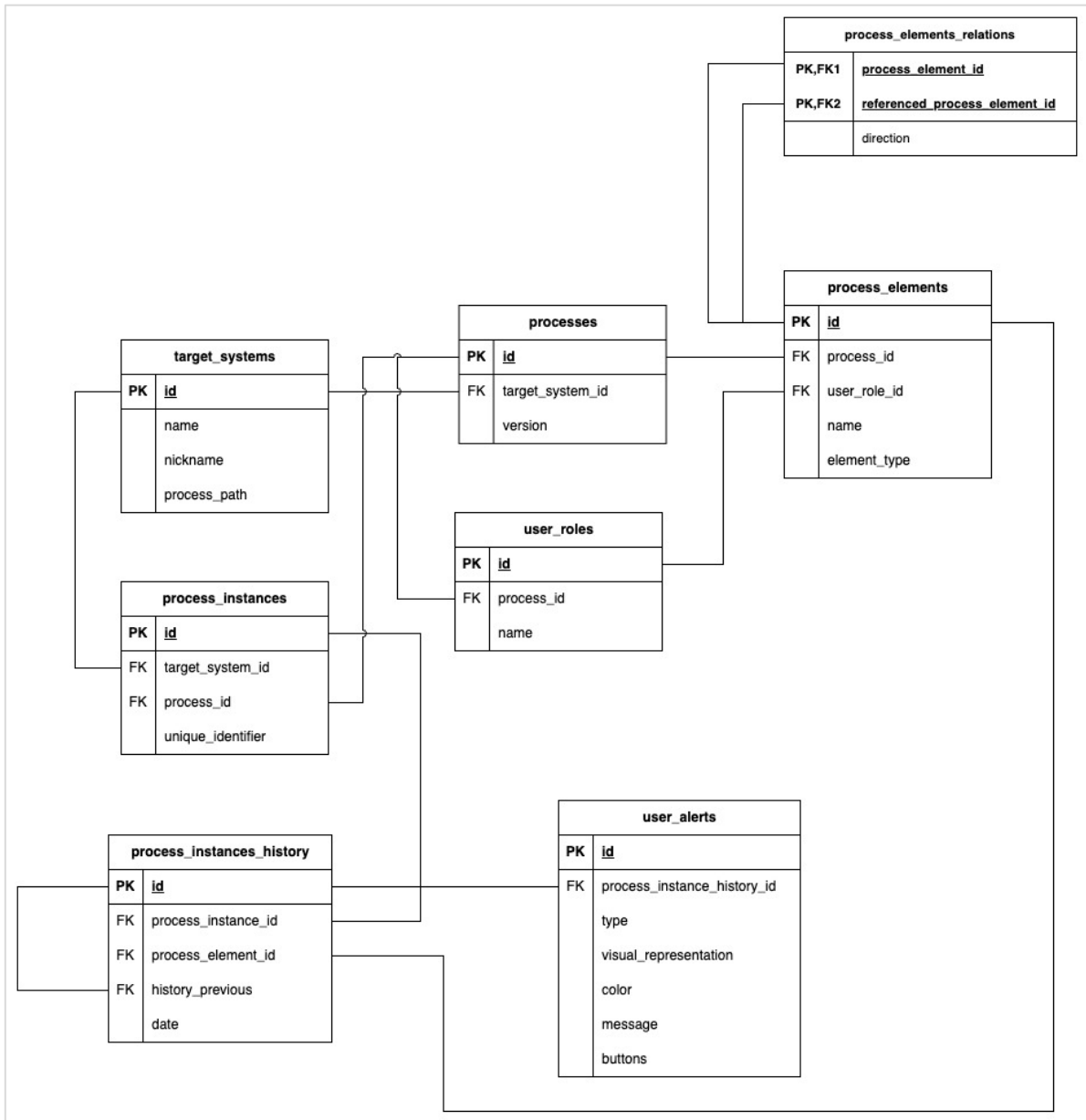


Figura 77 Diagrama relacional del componente

### 4.3. Construcción

En esta fase se realizará la implementación del componente tomando como base los requisitos y el diseño propuesto en las fases anteriores, iniciando con la selección de las tecnologías que soportarán la implementación de los requisitos

#### 4.3.1. Selección de tecnologías

Para la implementación del componente se han elegido las siguientes tecnologías:

- Docker: Se ha optado por Docker para encapsular la aplicación y sus dependencias en contenedores. Esto facilita un despliegue eficiente del componente. Además, reduce el riesgo de posibles fallos en las implementaciones de producción.



- MySQL: Se ha elegido MySQL como motor de base de datos, debido a su reconocida reputación como motor relacional. Destaca por su amplia compatibilidad con diversos lenguajes de programación y tecnologías de contenedores como Docker.
- PHP: Este lenguaje ha sido seleccionado por ser un lenguaje de scripting orientado a objetos del lado del servidor de propósito general, conocido por su agilidad en el desarrollo, gracias a su sintaxis sencilla. Además, PHP tiene compatibilidad con múltiples bases de datos y servicios en la nube, respaldado por una extensa comunidad de desarrolladores.
- Laravel/lumen: Dado que el PHP es el lenguaje elegido, se ha optado por utilizar el Framework Laravel/Lumen para mapear la lógica del componente, este Framework proporciona todas las herramientas necesarias para la implementación de un API eficiente. Se destaca por su ligereza al proveer sólo los componentes esenciales, en contraste con otros Frameworks conocidos de PHP.
- Composer: Una herramienta que simplifica la gestión de dependencias en proyectos PHP, permitiendo definir librerías y paquetes a través de un archivo en formato JSON, facilitando la instalación y actualización de estas.
- GitHub Copilot: Una herramienta de inteligencia artificial que usa tecnología de procesamiento de lenguaje natural para mejorar la productividad de los desarrolladores, ofreciendo sugerencias en tiempo real para mejorar la velocidad y consistencia del código generado.

#### 4.3.2. Inicialización del proyecto

Una vez elegidas las tecnologías, se inició la fase de implementación, la primera fase consistió en inicializar el proyecto, configurando las tecnologías necesarias para la ejecución del framework del proyecto, como se detalla en el Anexo 3.

#### 4.3.3. Estructura e implementación de elementos

Para llevar a cabo la implementación específica de los requisitos, y de acuerdo con la estructura del Framework elegido para el desarrollo del componente, el código se dividió en distintos componentes con uno un objetivo específico y se organizaron dentro de la estructura de carpetas propia de Lumen. A continuación, se describen algunos de los elementos más relevantes en la implementación del componente.

##### 4.3.3.1. Migraciones

Las migraciones tienen el objetivo de generar el modelo de datos en el motor de base de datos elegido apoyándose en la tecnología proveída por Lumen, estas fueron creadas en un orden específico, garantizando la correcta creación de tablas, índices y llaves foráneas como se detalla en el Anexo 4, sección 1.

##### 4.3.3.2. Modelos

Los modelos representan cada una de las entidades y atributos que componen el dominio, encargándose de representar los datos que persisten en la base de datos en objetos según el modelo entidad relación planteado en la fase de diseño, la implementación de estos se detalla en el Anexo 4, sección 2.

#### 4.3.3.3. Repositorios

Los repositorios son casos de uso encargados de realizar la comunicación con la base de datos. A cada uno de los modelos del dominio se le asoció un repositorio específico, el cual es el encargado de obtener y almacenar registros desde la base de datos, la implementación de repositorios se detalla en el Anexo 4, sección 3

#### 4.3.3.4. Excepciones

Las excepciones son artefactos de código que permiten detener la ejecución por la presencia de un error, este error puede ser un error controlado lanzado mediante la lógica programada. Para estos casos se crea la implementación de Excepciones mediante una jerarquía definida como se detalla en el Anexo 4, sección 4, con el fin de tener más control sobre los errores presentados en el componente.

#### 4.3.3.5. Casos de uso

De manera general corresponden a las unidades de código de responsabilidad única que mapean una función específica de la lógica de negocio como se detalla en el Anexo 4, sección 5.

#### 4.3.3.6. Controladores

Los controladores son los elementos que actúan como puente de conexión entre las peticiones enviadas a través del API REST y los casos de uso de dominio del componente, encargándose de la validación de los parámetros y la creación de respuestas hacia el cliente, su implementación se detalla en el Anexo 4, sección 6.

### 4.3.4. Casos de uso genéricos para modelos

Los casos de uso genéricos para modelos serán aquellos que implementarán la lógica general de los modelos dentro del componente, como la creación y validación de los modelos, los cuales, aunque tienen algunas particularidades, se manejarán como un flujo genérico que detallará sus particularidades mediante el uso de casos de uso abstractos con implementaciones concretas para cada modelo.

#### 4.3.4.1. Validadores de modelos

Los validadores de modelos son casos de uso con la responsabilidad de la validación de los datos utilizados para la creación o actualización de un modelo del dominio, garantizando que los datos de estos cumplan con las condiciones de negocio, la estructura de estos casos de uso se detalla en el Anexo 4, sección 7.

#### 4.3.4.2. Creadores de modelos

Los casos de uso creadores de modelos son casos de uso con la responsabilidad de crear y actualizar modelos, encargándose de la validación de los datos mediante los validadores y el almacenamiento de estos mediante repositorios, estos casos de uso se podrán invocar desde cualquier elemento que lo requiera. La implementación de estos casos de uso se detalla en el Anexo 4, sección 8.

### 4.3.5. Casos de uso asociados a los requisitos

Para la implementación de los requisitos del componente, se generarán elementos de código que mediante el uso de los elementos mencionados anteriormente puedan cumplir con el objetivo propuesto.

#### 4.3.5.1. Implementación del requisito CDA-001

Tomando en cuenta que esta funcionalidad está orientada a generar respuestas para ser enviadas a través del API REST, este requisito se implementa en la lógica de los controladores, ya que estos son los encargados de generar las respuestas del API REST. Para tal fin, la funcionalidad de generación de respuestas genéricas que es el objetivo de este requisito se incluye dentro de las funciones disponibles a través del controlador abstracto del cual extienden todos los controladores. Estas funciones añadidas son las encargadas de generar las respuestas tanto de éxito como de error según la estructura y condiciones definidas. Con el objetivo de reutilizar la lógica, estas funciones se agregaron dentro de un *trait* como se muestra en la Figura 78, el cual será usado tanto en el controlador abstracto como en el *handler* de excepciones, manteniendo de este modo, la consistencia en las respuestas del componente el cual es el objetivo este requisito.

```
1 <?php
2
3 namespace App\Traits\Response;
4
5 use App\Exceptions\General\FatalException;
6 use Illuminate\Http\JsonResponse;
7
8 trait CreateResponsesTrait
9 {
10
11     private const succesStatusCodes = [200, 201];
12     private const failureStatusCodes = [400, 401, 422, 500];
13
14 > /** ...
24     protected function jsonSuccessResult(
25         int $statusCode,
26         string $message,
27         array $data = [],
28         array $warnings = []
29 > ): JsonResponse {--
47     }
48
49
50 > /** ...
60     protected function jsonFailureResult(
61         int $statusCode,
62         string $message,
63         array $errors = [],
64         array $warnings = []
65 > ): JsonResponse {--
83     }
84 }
```

Figura 78. Implementación trait generador respuestas genérico

#### 4.3.5.2. Implementación del requisito CDA-002

Al igual que el requisito anterior, este requisito está orientado a controladores, ya que son estos los que se encargan de recibir los datos enviados mediante el API REST, por tal motivo, la implementación de la lógica de validación de datos de peticiones fue incluida dentro del controlador abstracto del cual extienden todos los controladores, para que los controladores

concretos puedan invocar la lógica de manera general. Al igual que el caso anterior, esta implementación se realiza mediante un trait como se muestra en la Figura 79.

```
1 <?php
2
3 namespace App\Traits\Controller;
4
5 use App\Exceptions\Request\ValidationRequestException;
6 use Illuminate\Http\Request;
7
8 trait GetParamsFromRequestTrait
9 {
10     use RestrictionsValidationTrait;
11
12 > /** ...
21 public function getParamsFromRequest(Request $request, array $simpleParams, array $fileParams = [])
22 > { ...
60 }
61
62 > /** ...
71 > private function getRestrictionErrorsOnValue(string $param, $value, array $restrictions): array { ...
114 }
115 }
```

Figura 79. Implementación trait validación de datos peticiones genérico

#### 4.3.5.3. Implementación de requisitos de servicios del API REST

La implementación de los requisitos CDA-003, CDA-004, CDA-005, CDA-006, CDA-007, CDA-008 y CDA-009, cuyo objetivo es crear servicios en el API REST del componente, para crear u obtener información, se realizó mediante la configuración de rutas de Lumen, estas rutas permiten definir un servicio especificando el método HTTP, la url y el controlador que procesará las peticiones como se muestra en la Figura 80.

```
20 $router->post('/targetsystem', 'CreateTargetSystemController');
21 $router->get('/targetsystem', 'GetTargetSystemController');
22
23 $router->post('/process', 'CreateProcessController');
24 $router->get('/process', 'GetProcessController');
25
26 $router->post('/instance', 'CreateInstanceController');
27 $router->get('/instance', 'GetInstanceController');
28 $router->put('/instance', 'UpdateInstanceController');
```

Figura 80. Configuración de rutas del API REST

Cada controlador asociado se creó siguiendo la estructura para controladores definida anteriormente, mediante la cual se utilizan los métodos de validación de peticiones y creación de respuestas a través del controlador abstracto. Además, cada controlador mediante inyección de dependencias agrega los elementos requeridos para llevar a cabo su objetivo, dichos elementos pueden variar entre casos de uso creadores de modelos para los servicios de creación/actualización, o repositorios para servicios de información.

#### 4.3.5.4. Casos de uso de dominio del componente

Los casos de uso de dominio del componente corresponden a los casos de uso para los requisitos CDA-006.01, CDA-009.01, CDA-009.02 los cuales se implementaron dentro del componente siguiendo la estructura de casos de uso definida anteriormente como se muestra en la Figura 81, internamente estos casos de uso mediante la inyección de dependencias

usaron otros casos de uso genéricos, como los casos de uso de creación de modelos y repositorios, para poder cumplir con su objetivo.

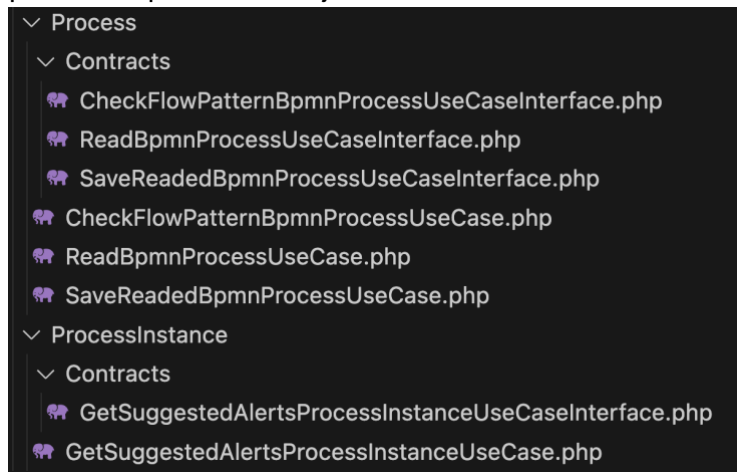


Figura 81. Implementación de casos de uso de dominio del componente

#### 4.3.6. Validación de requisitos

Para la validación de se utilizaron las herramientas Postman y PHPUnit; Postman se utilizó para realizar peticiones de prueba al API REST en un ambiente local controlado, agregando peticiones correctas e incorrectas y verificando que el contenido cumpla con las condiciones de validación creadas para cada requisito como se detalla en el Anexo 5, sección 1. Por otra parte, PHPUnit, la cual es una librería que viene incluida dentro de lumen, se utilizó para la creación de test unitarios sobre los casos de uso del dominio del componente, mediante la validación de salidas esperadas de la ejecución para distintos escenarios como se detalla en el Anexo 5, sección 2.

#### 4.3.7. Publicación de código fuente

El código fuente relacionado ha sido publicado mediante el repositorio GitHub y puede ser accedido mediante el enlace <https://github.com/mbrayan125/componente-alertas>

## 5. Evaluación del componente de software

En esta sección se llevará a cabo la evaluación, donde las alertas dinámicas generadas por el componente desarrollado serán analizadas en el contexto de un sistema software que opera según un flujo de trabajo claramente definido.

Para la evaluación, se empleará el método de *evaluación por expertos*, que implica la calificación de las alertas de usuario generadas por el componente durante la ejecución de una instancia específica de un flujo del sistema software seleccionado. Para la selección del grupo expertos, se elegirá una muestra de colaboradores del sistema seleccionado, que desempeñen roles relacionados al desarrollo de este, pudiendo abarcar roles como desarrolladores, *product owner*, control de calidad (QA) entre otros, ya que ellos no solo poseen conocimiento del dominio en cuestión, sino que también están directamente involucrados en la implementación y evolución continua del sistema. Su experiencia y comprensión permitirá una evaluación integral de las alertas generadas por el componente, abarcando tanto aspectos técnicos como funcionales del sistema.

Para respaldar y medir la evaluación, se utilizarán encuestas cualitativas diseñadas con el propósito de cuantificar la precisión de las alertas generadas por el componente en el contexto seleccionado. Estas encuestas permitirán analizar y comparar las respuestas, proporcionando un punto de referencia para la construcción de conclusiones sobre la *correctitud* de las alertas generadas e identificar puntos de mejora y trabajo futuro.

### 5.1. Selección del sistema objetivo

El sistema elegido deberá ser uno cuyo funcionamiento esté guiado por uno o más flujos de trabajo. Estos flujos deben estar definidos o ser susceptibles de ser transformados al estándar BPMN de acuerdo con el estándar aceptado por el componente.

De acuerdo con lo anterior, fue analizada Docxflow®, una plataforma de software desarrollada por Makro Soft S.A.S que sistematiza, administra, gestiona y traza información, documentos físicos y electrónicos, la cual cuenta con distintos módulos, destacándose entre los más importantes los módulos de producción documental, radicación, conservación, archivo y firma electrónica.

Dentro de Docxflow® la definición de algunos procesos o módulos está representada mediante flujos de trabajo, en los cuales se definen las actividades y bifurcaciones del proceso desde que se inicia hasta que se finaliza y los encargados de cada actividad en cada punto, por lo que Docxflow® es un candidato ideal para realizar el proceso de evaluación, ya que cumple con la condición principal (funcionamiento guiado por un flujo bien definido).

### 5.2. Abstracción del proceso de negocio de la evaluación

Dentro de Docxflow®, el módulo de radicación comprende tres (3) procesos principales, *trámite en ventanilla física* que representa el proceso de un trámite que ha sido radicado por un usuario de manera física, el trámite *en ventanilla virtual* que representa el proceso de un trámite que ha sido radicado mediante la sucursal virtual del sistema y el *trámite nuevo* que

representa un trámite que es llevado a cabo de manera interna. Mediante Makro Soft S.A.S, se ha tenido acceso al modelo de flujo para el proceso de *trámite en ventanilla física*, el cual corresponde al mostrado en la Figura 82.

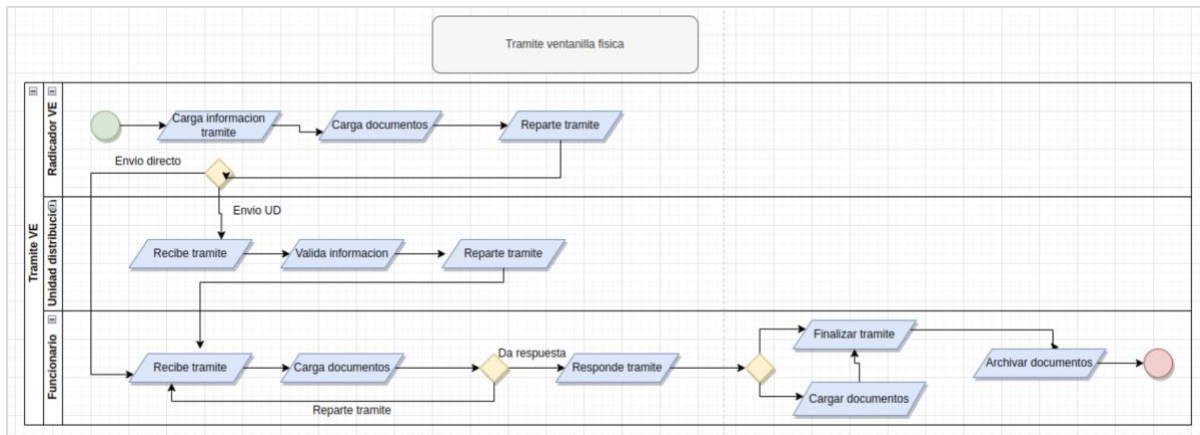


Figura 82. Diagrama de flujo proceso de trámite en ventanilla física®

Para que el componente pueda generar alertas de usuario sobre este proceso, y que estas puedan ser evaluadas, es necesario que primero el proceso sea transformado al estándar BPMN, cuyo resultado debe representar fielmente al flujo original, incluyendo todas las bifurcaciones, actividades y roles. La transformación del flujo entregado a BPMN inicia con la identificación de los roles de usuario que participan en el proceso, siendo estos los siguientes:

- Radicador VE
- Unidad distribución
- Funcionario

El primer paso para la definición del proceso abstraído en BPMN es generar un diagrama de flujo que abstraiga de manera general todo el proceso, y que luego se desglose en subprocesos que definirán el detalle requerido.

La definición de la abstracción general del proceso se muestra en la Figura 83, en donde se definen los subprocesos principales *Radicar documento*, *Repartir trámite*, *Responder trámite* y *Finalizar trámite*.

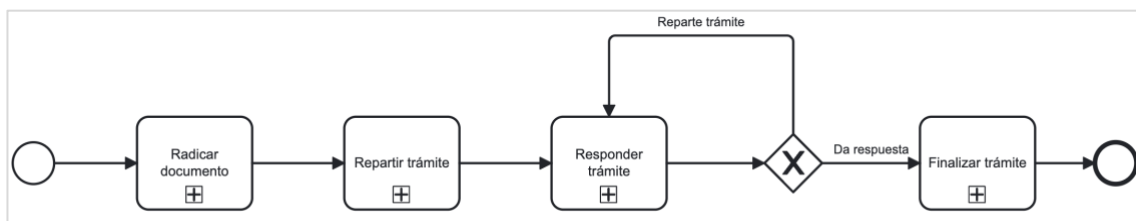


Figura 83. Diagrama BPMN general equivalente para proceso trámite ventanilla física - Docxflow®

De acuerdo con el estándar definido para los procesos BPMN soportados por el componente, cada subproceso se convierte en un proceso BPMN independiente, en donde cada uno de los roles se convierte en un *Lane*, y las actividades y bifurcaciones se organizarán dentro de estos *Lanes* dependiendo de quién sea el encargado de su ejecución. En la Figura 84 se muestra el desglose del subproceso *Radicar documento*, en el cual se agregan las actividades *Cargar*

información de trámite y Cargar documentos las cuales son realizadas por el rol Radicador Ventanilla de Entrada. Este proceso se considera un proceso no idempotente, ya que cada ejecución representa un nuevo trámite dentro de la plataforma, que en caso de no llevarse a cabo afectará los indicadores de gestión.

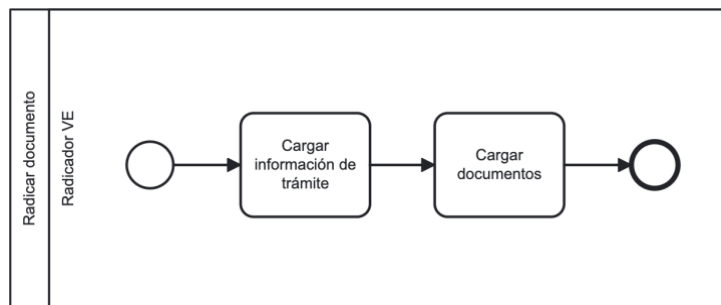


Figura 84. Diagrama BPMN subproceso Radicar documento

Por otra parte, el proceso de BPMN para el subproceso *Repartir trámite* se define en la Figura 85, en donde, el Rol Radicador Ventanilla de Entrada selecciona el modo de reparto, es decir cómo se enviará el trámite hacia el usuario que lo gestionará, este proceso se puede llevar a cabo mediante una Unidad de distribución en la cual se delega la responsabilidad de asignación a un rol adicional llamado *Unidad de distribución*, el cual se encarga de validar la información del trámite y posteriormente asignar al usuario encargado; el otro método de reparto se denomina envío directo, en donde el Radicador Ventanilla de Entrada asigna directamente a un usuario para atender el trámite.

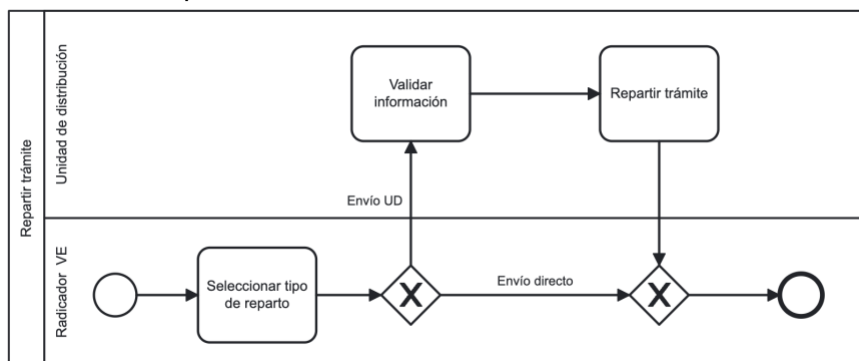


Figura 85. Diagrama BPMN subproceso Repartir trámite

Una vez se ha asignado el usuario encargado del trámite, entra en marcha el subproceso *Responder trámite* como se muestra en la Figura 86, en donde el usuario asignado carga documentos al trámite y tiene la opción de responder el trámite o volverlo a repartir, en cuyo caso se asignaría un nuevo usuario al trámite, generando un bucle en este proceso hasta que un usuario le dé respuesta al trámite.

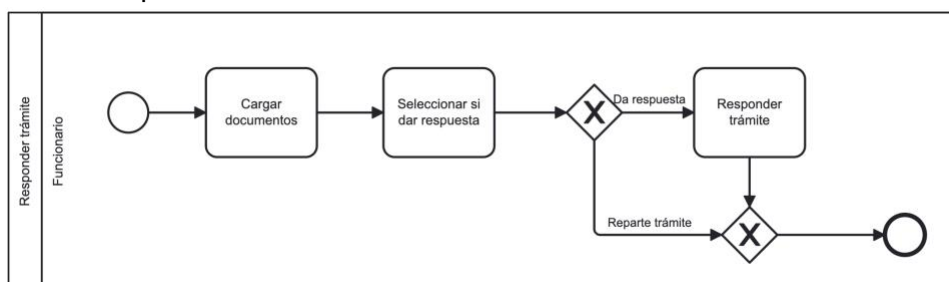


Figura 86. Diagrama BPMN subproceso Responder trámite



Cuando el trámite tiene asociada una respuesta, se realiza la finalización del trámite como se muestra en la Figura 87, en la cual el funcionario que dio respuesta tiene la opción de cargar documentos adicionales o finalizar el trámite, en cuyo caso realizará el archivado de los documentos cargados y concluirá el proceso de finalizar trámite y también el proceso de *Trámite ventanilla física general*.

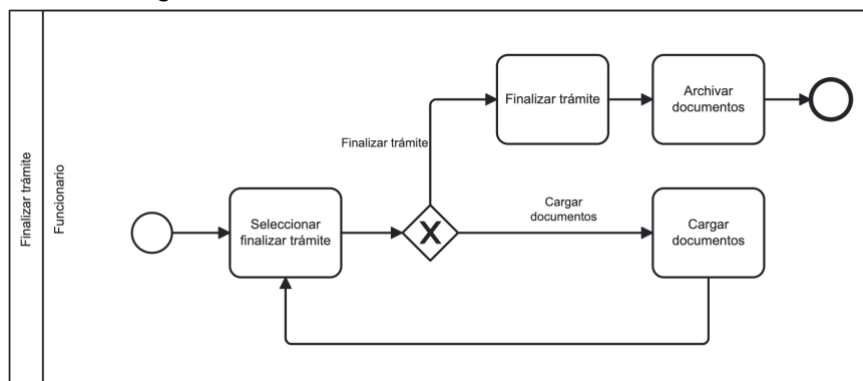


Figura 87. Diagrama BPMN subproceso Finalizar trámite

### 5.3. Selección de usuarios evaluadores

Los usuarios que realizaron la evaluación de las alertas generadas por el componente para los procesos abstraídos son colaboradores de Makro Soft S.A.S, y son quienes apoyan el proceso de desarrollo de Docxflow® desde distintos cargos, y quienes conocen el software, sus objetivos y sus procesos. En la Tabla 30 se presenta la información relevante sobre los colaboradores de Makro Soft S.A.S quienes hicieron parte del proceso de evaluación como expertos en el sistema Docxflow®.

Tabla 30. Datos de usuarios evaluadores del proceso

| <b>Colaborador #1</b>              |                               |
|------------------------------------|-------------------------------|
| Nombre                             | Leidy Pedraza                 |
| Cargo                              | Desarrollador de Software     |
| Tiempo de experiencia profesional  | 3 a 5 años                    |
| Tiempo de experiencia en Docxflow® | 1 a 2 años                    |
| <b>Colaborador #2</b>              |                               |
| Nombre                             | Jorge Eliecer Castillo Erazo  |
| Cargo                              | Quality Assurance (QA)        |
| Tiempo de experiencia profesional  | 1 a 2 años                    |
| Tiempo de experiencia en Docxflow® | Menos de 1 año                |
| <b>Colaborador #3</b>              |                               |
| Nombre                             | John Alexander Hernández      |
| Cargo                              | Desarrollador de Software     |
| Tiempo de experiencia profesional  | 1 a 2 años                    |
| Tiempo de experiencia en Docxflow® | 1 a 2 años                    |
| <b>Colaborador #4</b>              |                               |
| Nombre                             | Santiago Ordoñez              |
| Cargo                              | Soporte técnico nivel 2       |
| Tiempo de experiencia profesional  | 3 a 5 años                    |
| Tiempo de experiencia en Docxflow® | 1 a 2 años                    |
| <b>Colaborador #5</b>              |                               |
| Nombre                             | Cristian Esthibel Gómez Campo |
| Cargo                              | Desarrollador de Software     |
| Tiempo de experiencia profesional  | 3 a 5 años                    |

|                                    |                           |
|------------------------------------|---------------------------|
| Tiempo de experiencia en Docxflow® | 1 a 2 años                |
| <b>Colaborador #6</b>              |                           |
| Nombre                             | Jorge Andrés Córdoba      |
| Cargo                              | Desarrollador de Software |
| Tiempo de experiencia profesional  | 3 a 5 años                |
| Tiempo de experiencia en Docxflow® | 1 a 2 años                |
| <b>Colaborador #7</b>              |                           |
| Nombre                             | Jonathan Ibarra           |
| Cargo                              | Desarrollador líder       |
| Tiempo de experiencia profesional  | 3 a 5 años                |
| Tiempo de experiencia en Docxflow® | 1 a 2 años                |

## 5.4. Definición de encuestas sobre alertas

Para evaluar las alertas de usuario generadas por el componente, se emplearon cuestionarios que posteriormente serán respondidos por los expertos seleccionados. Estos cuestionarios tienen como objetivo recopilar datos para determinar el grado de correctitud de las sugerencias de alertas, es decir, corroborar si los atributos de la alerta en el contexto en el que se sugieren previenen al usuario de cometer *slips* o *mistakes*, es decir, errores por desconocimiento o errores involuntarios. Lo anterior mediante la calificación cualitativa de los aspectos más relevantes de la alerta y que hacen parte de las sugerencias generadas por el componente como tipo, color, título y contenido y acciones.

Cada punto del proceso en donde se genere una alerta se convertirá en un cuestionario, en el cual como título se agregará el nombre del elemento del proceso donde se presentó la sugerencia, una pantalla de Docxflow® correspondiente a la interfaz del sistema para dicho punto y las alertas de usuario sugeridas renderizadas. Las preguntas del cuestionario serán las siguientes:

1. ¿Considera que esta alerta previene al usuario de cometer descuidos o acciones incorrectas?
  - a. Si
  - b. No
2. Con respecto a la alerta presentada para el punto del proceso usted considera que la alerta es
  - a. Relevante, aporta datos necesarios
  - b. Irrelevante, no aporta ninguna información valiosa
3. ¿De acuerdo con el punto del proceso donde se presentó la alerta sugerida, el título informa y orienta sobre el estado del proceso?
  - a. Sólo informa
  - b. Sólo orienta
  - c. Orienta e informa
  - d. Ni informa ni orienta
4. ¿De acuerdo con el punto del proceso donde se presentó la alerta sugerida, el contenido informa y orienta sobre el estado del proceso?
  - a. Sólo informa
  - b. Sólo orienta
  - c. Orienta e informa
  - d. Ni informa ni orienta

5. ¿El icono mostrado en la alerta, está correctamente relacionado al título y contenido de esta?
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
  - f. No debería estar incluido
6. ¿El color mostrado en la alerta, está correctamente relacionado al título y contenido de esta?
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
  - f. No debería estar incluido
7. ¿Las acciones mostradas en la alerta, son adecuadas para el contexto en el que esta se sugiere?
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
  - f. No deberían incluirse
8. ¿De acuerdo con el título, contenido, color, e icono, el tipo definido para esta alerta <tipo> se corresponde con los ítems mencionados?
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo

Una vez respondidos los cuestionarios se tendrá adicionalmente la opción de *comentarios* en el cual el experto podrá agregar observaciones de manera general sobre las alertas presentadas y el proceso en sí.

## 5.5. Preparación de evaluación

Una vez definidos los elementos necesarios para realizar la evaluación por parte de los expertos, se deben preparar algunos ítems antes de llevar a cabo el proceso.

### 5.5.1. Agendamiento de reunión

Dado que los expertos elegidos tienen sitios de trabajo y residencia geográficamente distantes, la evaluación se agendó mediante una sesión virtual, utilizando la plataforma *Google Meet*. La fecha y hora de la reunión se define en un espacio disponible común, acordado entre los expertos y el estudiante.

### 5.5.2. Preparación de alertas sugeridas a evaluar

Dado que el código de Docxflow® no es posible modificarlo para que este se comunique de manera automática con el componente y renderice las sugerencias por su cuenta, se realizó de manera manual la carga del sistema objetivo y sus procesos asociados al componente como se muestra en el Anexo 6 sección 1. Luego para cada proceso cargado, se creó una instancia en el componente y se actualizó en cada uno de los elementos definidos para el proceso mediante los servicios del API REST, con el fin de obtener las sugerencias a través del componente y renderizarlas de manera manual como se muestra en el Anexo 6 secciones 2 y 3, siendo las alertas renderizadas resultantes de este proceso, las que se evaluaron en la sesión de evaluación.

### 5.5.3. Captura de pantallas de aplicación seleccionada

Con el objetivo de clarificar el punto en el sistema en donde el componente sugiere la presentación de alertas y agregar las capturas de Docxflow® en los cuestionarios, con la ayuda y supervisión de un colaborador de Makro Soft S.A.S, se exploró el flujo en el sistema mediante una instancia de prueba en una aplicación de pruebas de Docxflow®, y se obtuvieron capturas de la interfaz del sistema en donde el componente sugirió alertas de usuario.

### 5.5.4. Creación de cuestionarios

Para cada uno de los puntos del proceso en donde el componente presentó sugerencias de alertas de usuarios, se creó un cuestionario de acuerdo con la estructura definida anteriormente, agregando las capturas de Docxflow®, alertas renderizadas y preguntas del cuestionario. Estos cuestionarios fueron creados utilizando *Google Forms*, para que estuviesen disponibles de manera online en el momento de la sesión de evaluación como se muestra en la Figura 88.

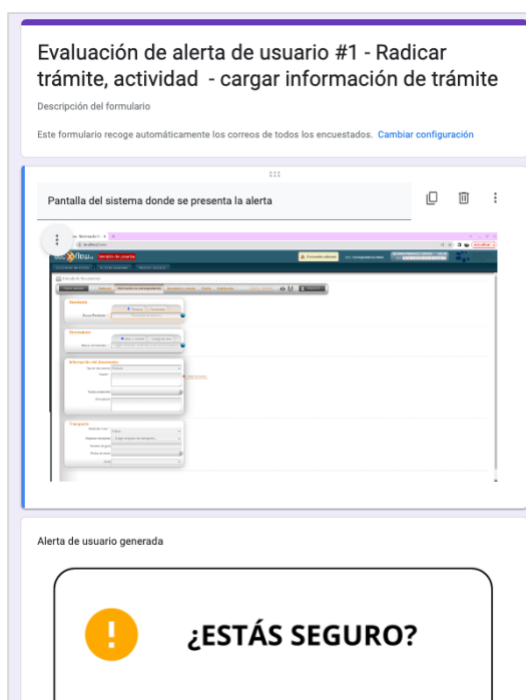


Figura 88. Cuestionario para alerta de usuario sugerida sobre Docxflow®

## 5.6. Ejecución de evaluación

La sesión de evaluación inició una vez todos los asistentes ingresaron a la reunión, se contextualizó a los asistentes sobre la investigación, incluyendo una descripción sobre alertas, flujos de trabajo y cómo el componente de alertas creado podría generarlas de manera dinámica. Posteriormente se compartió a los asistentes el flujo original entregado por Makro Soft S.A.S y el equivalente proceso y subprocesos BPMN generados a partir de este, describiendo paso a paso las actividades para esclarecer por qué estos se pueden considerar como equivalentes. Luego se compartieron las preguntas del cuestionario, indicando el objetivo de cada una de ellas y clarificando cada pregunta a qué parte de la alerta renderizada hace referencia.

Con el contexto dado y resueltas las dudas de los asistentes, se procedió ejecutar la fase de evaluación con los cuestionarios, la cual inició con un recorrido por cada uno de los subprocesos BPMN en el orden que lo indica el proceso general, pausando en cada uno de los puntos del proceso en donde el componente sugirió alertas para enviar los enlaces de las encuestas de las alertas respectivas, y esperando a que todos los asistentes finalicen las respuestas antes de continuar con el recorrido del flujo.

Una vez se recorrió todo el flujo y se respondieron todas las encuestas por parte de los asistentes, se finalizó la sesión de evaluación.

## 5.7. Resultados de la evaluación

Luego de recopilar los datos sobre las alertas de usuario generadas, se realiza el análisis de las respuestas proporcionadas durante la sesión de evaluación, es importante tomar en cuenta que las alertas generadas por el componente para la aplicación seleccionada correspondieron a 3 de los 4 patrones de alerta (patrones 1, 2, y 4). Para la pregunta #1 del cuestionario cuyo enunciado es *¿Considera que esta alerta **previene al usuario** de cometer descuidos o acciones incorrectas?* Las respuestas se distribuyeron de manera general como se muestra en la Figura 89, los evaluadores consideran que las alertas generadas mediante el componente en un 85.7% de los casos ayudan a la prevención de errores.

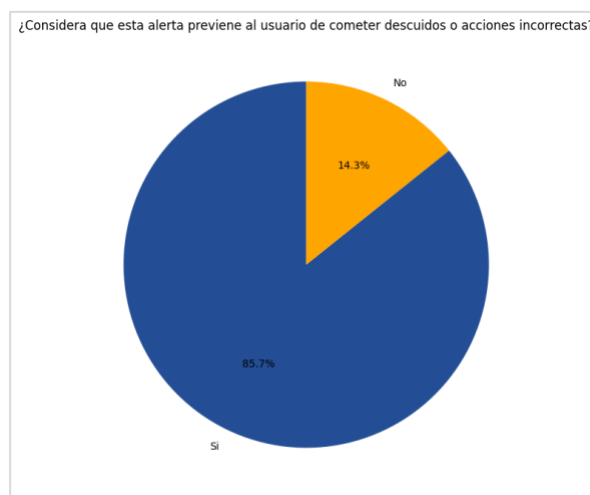


Figura 89. Diagrama general de respuestas para pregunta #1

Las alertas que presentan el mayor grado de respuestas 'No' a la pregunta son las alertas de Confirmación como se muestra en la Figura 90, esto se puede atribuir a que estas alertas son presentadas al final de cada proceso, indicando que este finalizó de manera correcta, y en ese punto no hay alguna acción que se pueda prevenir. Por otra parte, las alertas de Advertencia cuyo objetivo principal si es *prevenir errores de usuario*, fueron consideradas en un 100% como alertas que ayudan a tal fin.

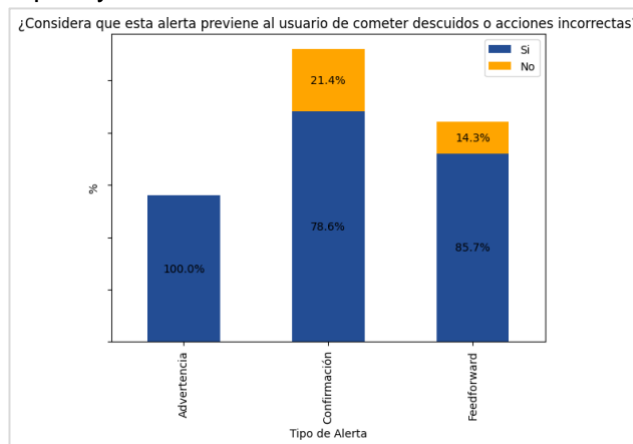


Figura 90. Diagrama de respuestas para pregunta #1 por tipo de alerta

Para el caso de la pregunta #2 del cuestionario, cuyo enunciado se relaciona con la alerta presentada para el punto del proceso usted **considera que la alerta es**, con las opciones Relevante (aporta datos necesarios) e Irrelevante (no aporta ninguna información valiosa) como se muestra en la Figura 91, los evaluadores consideraron que en general las alertas presentadas son relevantes en un 88.9%, frente al 11.1% que se consideró irrelevante.

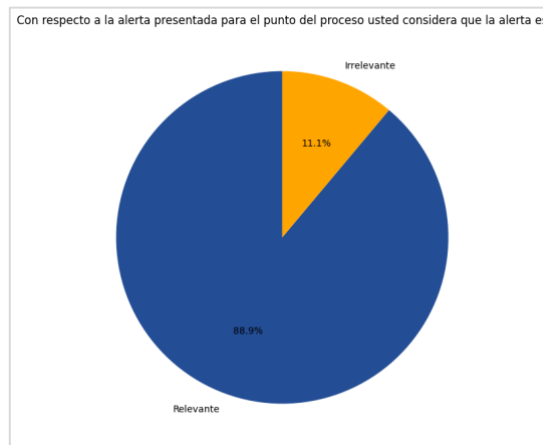


Figura 91. Diagrama general de respuestas para pregunta #2

Como se muestra en la as alertas de *Feedforward* (alertas presentadas en compuertas de división) son las que presentaron el indicador de relevancia más bajo con un 76.2%, a comparación de los otros tipos de alertas que tienen este valor por encima del 90%, esto se puede atribuir a que aunque en el flujo algunas bifurcaciones están bien definidas, en la implementación del software la elección de un camino u otro se implementa sobre una misma vista usando pestañas de selección libre, por ello las alertas en la selección de pestañas se pueden considerar como irrelevantes de acuerdo a esta manera de presentar las opciones.

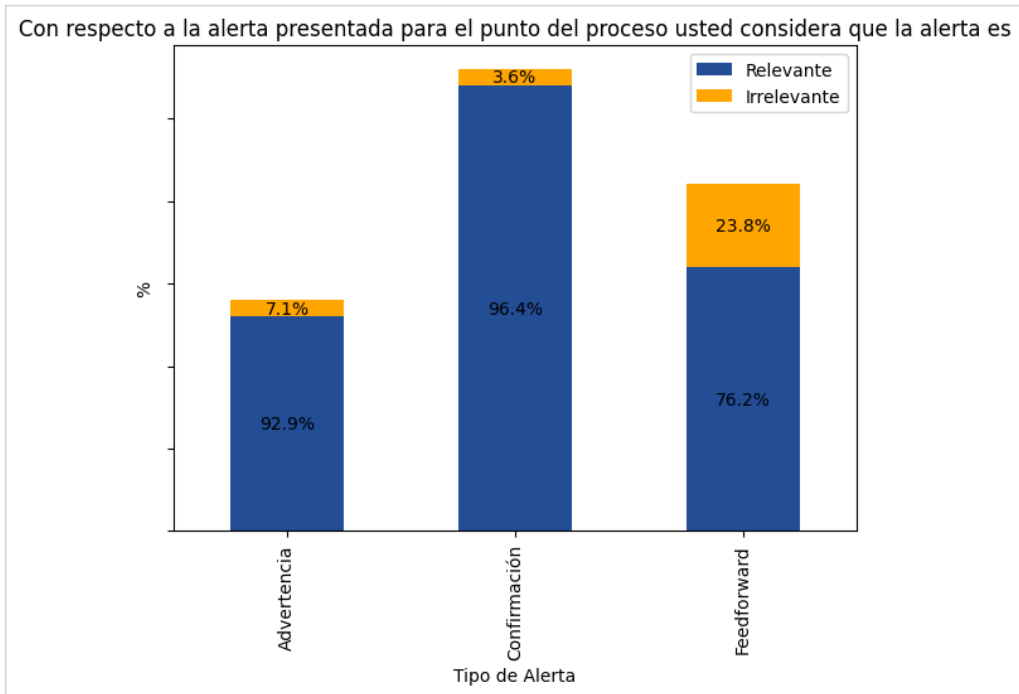


Figura 92. Diagrama de respuestas para pregunta #2 por tipo de alerta

Para la pregunta #3 del cuestionario cuyo enunciado indica que ¿De acuerdo con el punto del proceso donde se presentó la alerta sugerida, el título informa y orienta sobre el estado del proceso?, esta pregunta tiene como objetivo validar si el título de la alerta sugerida le ayuda a orientarse y/o informarse en el proceso que se está ejecutando, los evaluadores respondieron como se muestra en la Figura 93, en donde consideraron que todos los títulos de las alertas generadas permiten informar al usuario en un 63.5%, y orientar al usuario en un 14.3% o informar y orientar en un 22.2% de las veces. Ningún título de las alertas presentadas al usuario recibió la calificación de ‘Ni informa ni orienta’, por lo que se puede afirmar que todos los títulos fueron correctos de acuerdo con su intención de informar sobre el proceso.

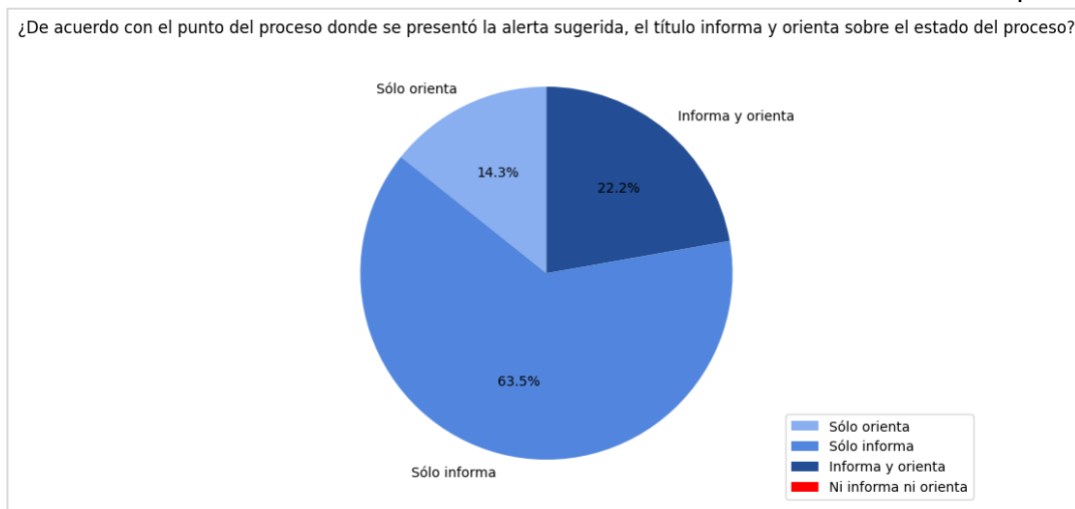


Figura 93. Diagrama general de respuestas para pregunta #3

En la Figura 94 se muestran las respuestas por tipo de alerta. Las alertas que tuvieron una tasa mayor de votos en ‘Informa y orienta’ con 54.4% fueron las de *Feedforward*, en las que se solicitaba ‘Confirmar elección’. Las alertas de Confirmación en su mayoría (96.4%) se consideró que solo informaban, ya que el título sólo agregó la palabra ‘Terminado’ sin ninguna

instrucción adicional, mientras que, para las alertas de Advertencia, se distribuyeron equitativamente los votos para 'Sólo informa' o 'Sólo orienta', lo cual puede deberse a que el título generado '¿Estás seguro?' puede considerarse tanto orientación o información dependiendo de un punto de vista que puede ser subjetivo.

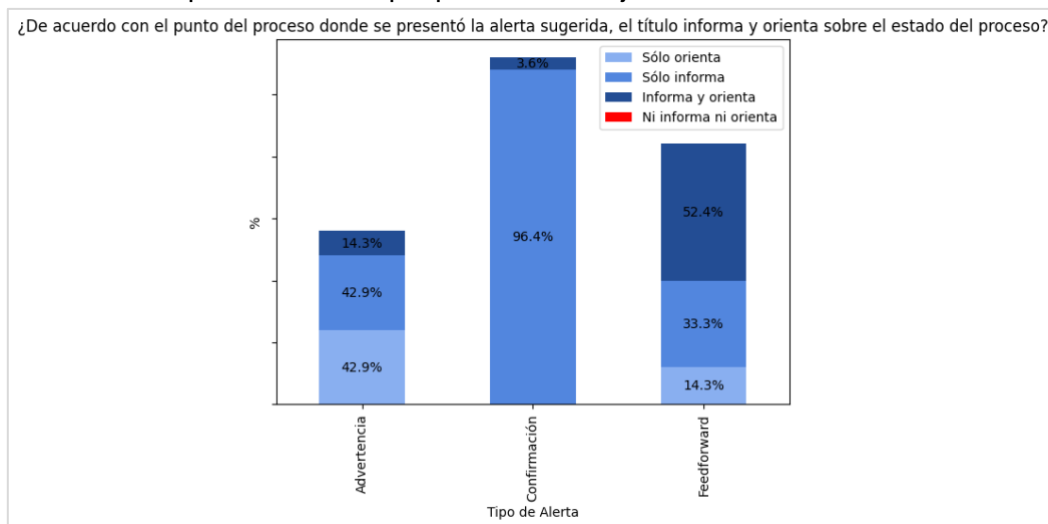


Figura 94. Diagrama de respuestas para pregunta #3 por tipo de alerta

Al igual que el punto anterior, la pregunta #4 del cuestionario tiene el objetivo de validar si el contenido de la alerta ayuda al usuario a orientarse en el proceso, mediante el enunciado *¿De acuerdo con el punto del proceso donde se presentó la alerta sugerida, el contenido informa y orienta sobre el estado del proceso?* Las respuestas de los evaluadores se muestran en la Figura 95, en donde consideraron que de manera general los contenidos de las alertas en un 52.4% de las veces informan sobre el estado del proceso, mientras que en el 41.3% de las veces además de informar también orienta al usuario. Ninguna alerta recibió calificación de 'Ni informa ni orienta', con lo que se puede afirmar que el contenido de todas las alertas presentadas fue correcto, de acuerdo con su intención de informar al usuario sobre el estado del proceso.

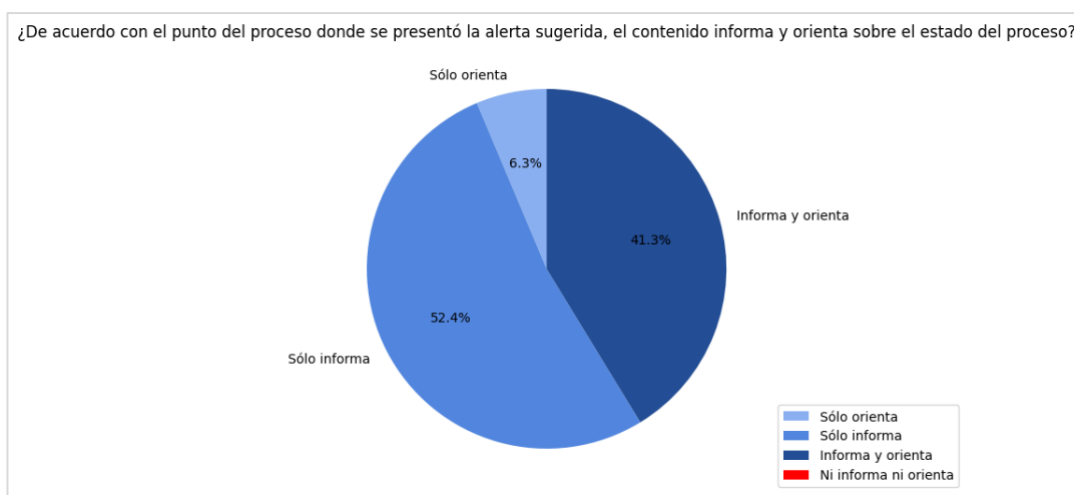


Figura 95. Diagrama general de respuestas para pregunta #4

La calificación de esta pregunta por tipo de alerta sigue la tendencia de la calificación del título como se muestra en la Figura 96, en donde las alertas de *Feedforward* presentaron la frecuencia relativa más alta en el ítem 'Informa y orienta' con un 76.2%, para estas alertas el



contenido estaba relacionado a validar la opción elegida en la compuerta de división exclusiva, por lo que se puede inferir que este contenido permite además de orientar, informar al usuario sobre la acción elegida. Las alertas de Confirmación que se encuentra con la mayor frecuencia es la opción 'Sólo informa', ya que en este caso el contenido era una confirmación del proceso terminado de acuerdo con estructura {sujeto} {verbo en pasado participio} {complemento}, por ejemplo 'Documento radicado'. Estos mensajes son meramente informativos y la tendencia se puede justificar por esa razón. Finalmente, para las alertas de Advertencia el contenido fue un contenido genérico que invitaba al usuario a validar los datos ingresados mediante el texto 'Revisa los datos antes de continuar', este mensaje puede tomarse tanto como una orientación como una información, ya que no proporciona datos puntuales, sino que da una instrucción general, por esto no hay una clara tendencia en las respuestas variando entre todas las opciones.

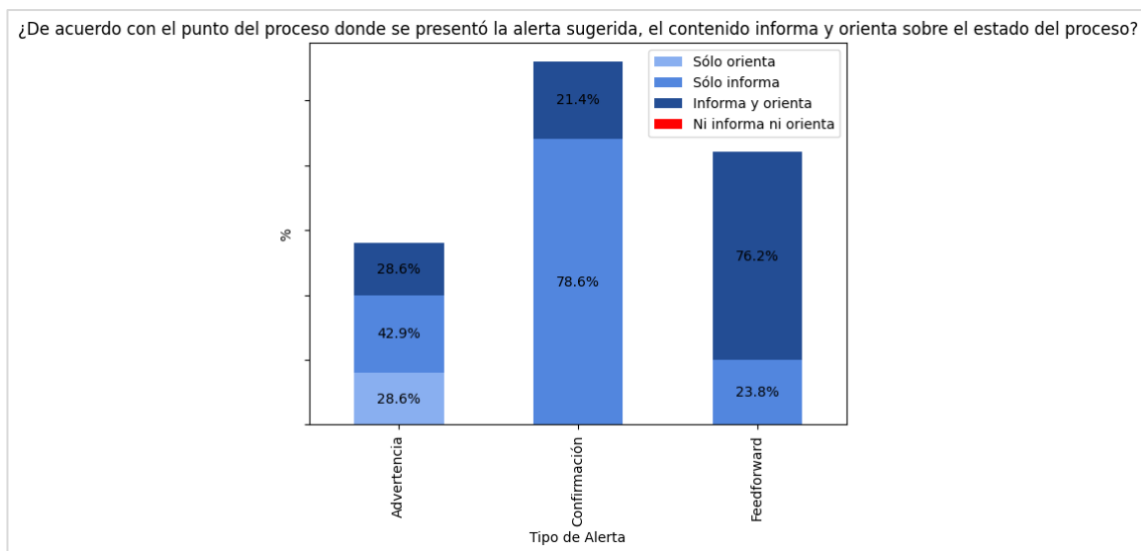


Figura 96. Diagrama de respuestas para pregunta #4 por tipo de alerta

La pregunta #5 del cuestionario con el enunciado *¿El ícono mostrado en la alerta, está correctamente relacionado al título y contenido de esta?*, tiene el objetivo de evaluar si el ícono incluido en la sugerencia es correcto de acuerdo con el contenido presentado en la misma. De acuerdo con las respuestas, en general los íconos de las alertas guardan relación con el contenido en un 89.9% de las veces como se muestra en la Figura 97, siendo el 11.1% consideración neutral. Ningún ícono presentado en las alertas fue evaluado como no relacionado, con lo cual se puede afirmar que las alertas sugerencias son correctas en cuanto al ícono presentado.

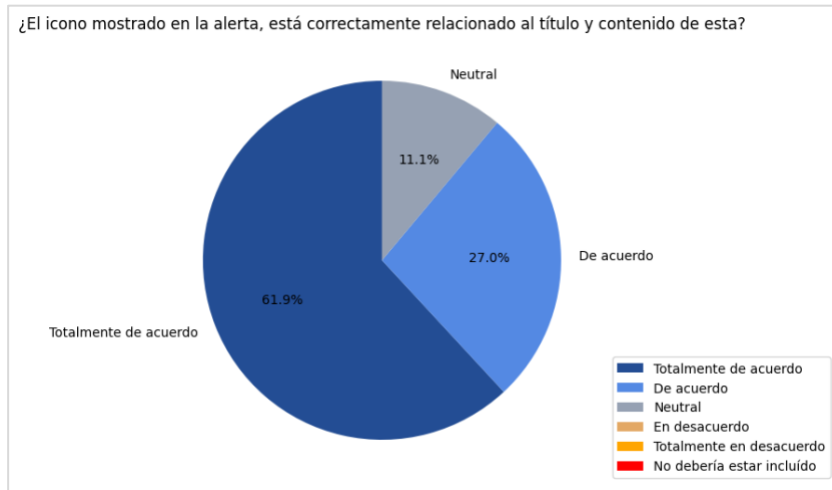


Figura 97. Diagrama general de respuestas para pregunta #5

Las alertas presentaron mayor grado de aceptación del ícono fueron las alertas de Confirmación con un 96.5%, le siguen alertas de advertencia con un 92.8% de aceptación y finalmente las alertas de *Feedforward* tuvieron una tasa de aceptación del 76.1%, siendo esta la menor entre todas, la razón de esto puede deberse a que el ícono mostrado en estas alertas (círculo con pregunta) parece no ser la mejor opción para la validación de opción elegida, de acuerdo con el pensamiento subjetivo de los evaluadores. Sin embargo, no se presentaron respuestas de desacuerdo o no inclusión del ícono.

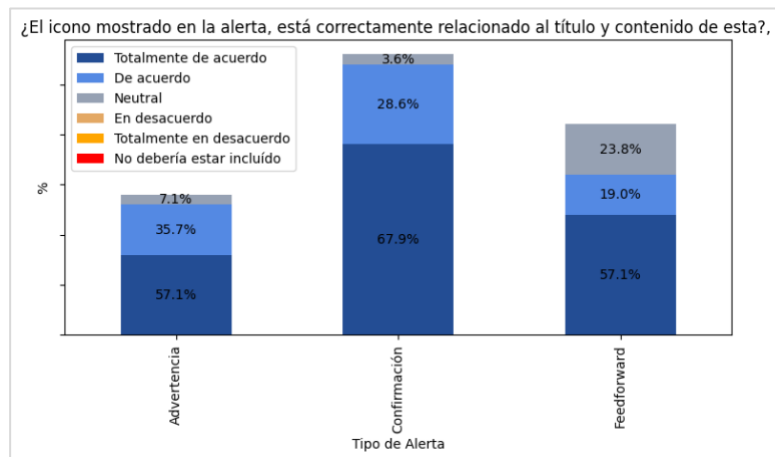


Figura 98. Diagrama de respuestas para pregunta #5 por tipo de alerta

Al igual que la pregunta anterior, la pregunta #6 evalúa mediante una escala el color en la alerta, el cual se representa tanto en el ícono como en la barra lateral, mediante el enunciado *¿El color mostrado en la alerta, está correctamente relacionado al título y contenido de esta?* Las respuestas a la pregunta #6 se presentan en la Figura 99. En general el color asociado a las alertas fue aceptado en un 85.7%. Lo que indica que el color de las sugerencias en la mayoría de los casos fue correcto en relación con el título y contenido.

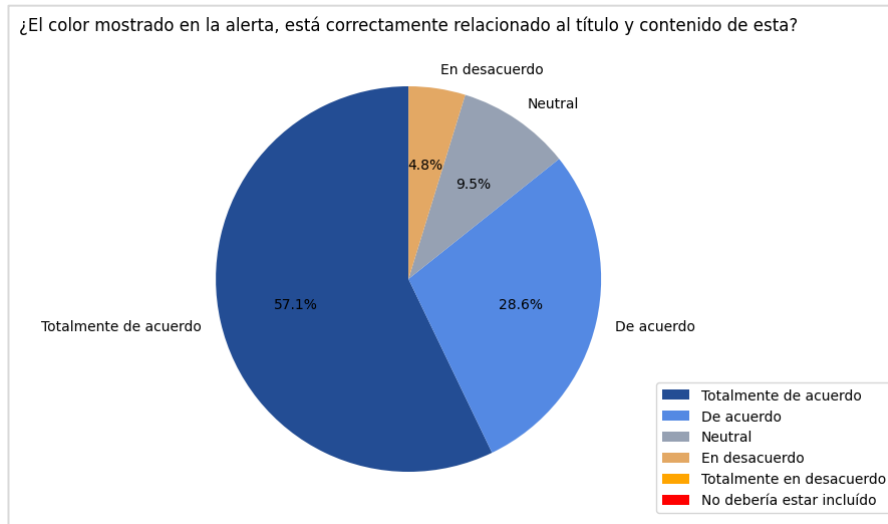


Figura 99. Diagrama general de respuestas para pregunta #6

Como se muestra en la Figura 100 las alertas de Confirmación presentaron la mayor tasa de aceptación del color con un 96.5% y las alertas de *Feedforward* una menor tasa con un 71.4%, para este tipo de alertas se encontraron respuestas por la opción 'En desacuerdo' representando el 14.3% de las respuestas.

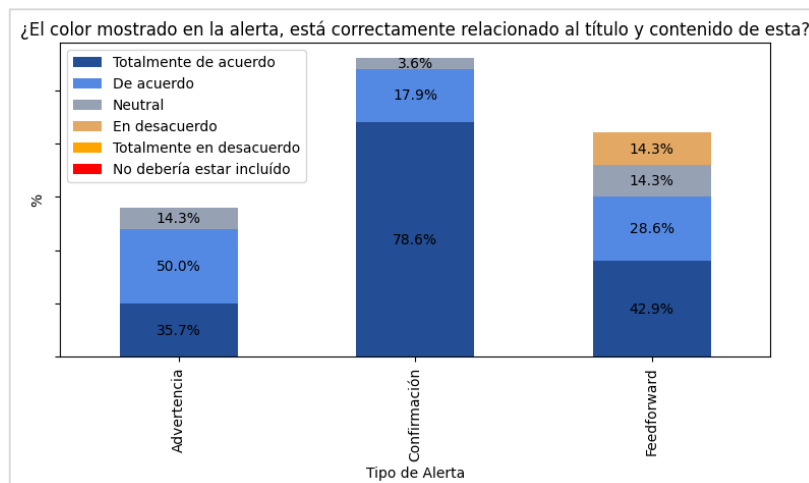


Figura 100. Diagrama de respuestas para pregunta #6 por tipo de alerta

La pregunta #7 evalúa mediante una escala las acciones presentadas en la alerta, las cuales se muestran como botones en la parte inferior de la alerta, mediante el enunciado *¿Las acciones mostradas en la alerta, son adecuadas para el contexto en el que esta se sugiere?* Las respuestas a se muestran en la Figura 101. En general las acciones presentadas se respondieron como correctas en un 88.9%. Lo que indica que las acciones de las sugerencias en la mayoría de los casos fueron correctas en relación con el título y contenido.

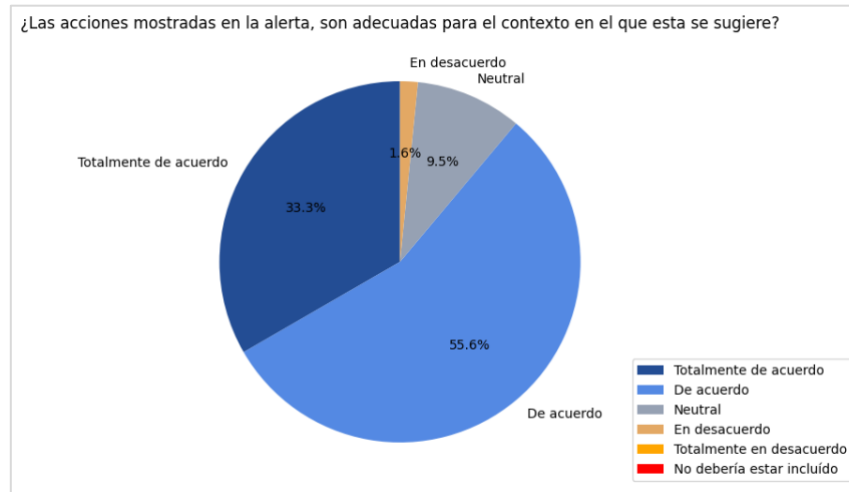


Figura 101. Diagrama general de respuestas para pregunta #7

Las alertas de tipo Confirmación no tuvieron acciones relacionadas, por lo que en este caso los resultados evalúan la ausencia de estas, la cual es alta con un 89.3% como se muestra en la Figura 102. Por otra parte, las alerta de *Feedforward* y Advertencia que sí mostraron las acciones de 'Continuar' y 'Cancelar', se evaluaron con el 100% y 80.9% de correctitud respectivamente con relación al título y contenido de la alerta.

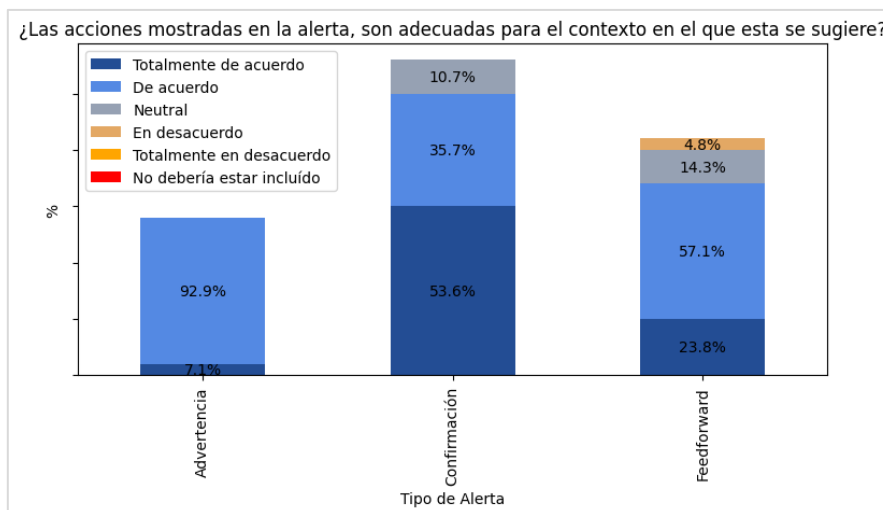


Figura 102. Diagrama de respuestas para pregunta #7 por tipo de alerta

La pregunta #8 que cuenta con el enunciado ¿De acuerdo con el título, contenido, color, e icono, el tipo o **clasificación como alerta de <tipo>** definido para esta alerta se corresponde con los ítems mencionados?, buscó validar si la determinación del tipo de alerta que permitió la asociación de colores a iconos se corresponde con la información presentada en la alerta. Los evaluadores consideraron que en un 92.1% el tipo seleccionado corresponde con los ítems presentados en la alerta como se muestra en la Figura 103.

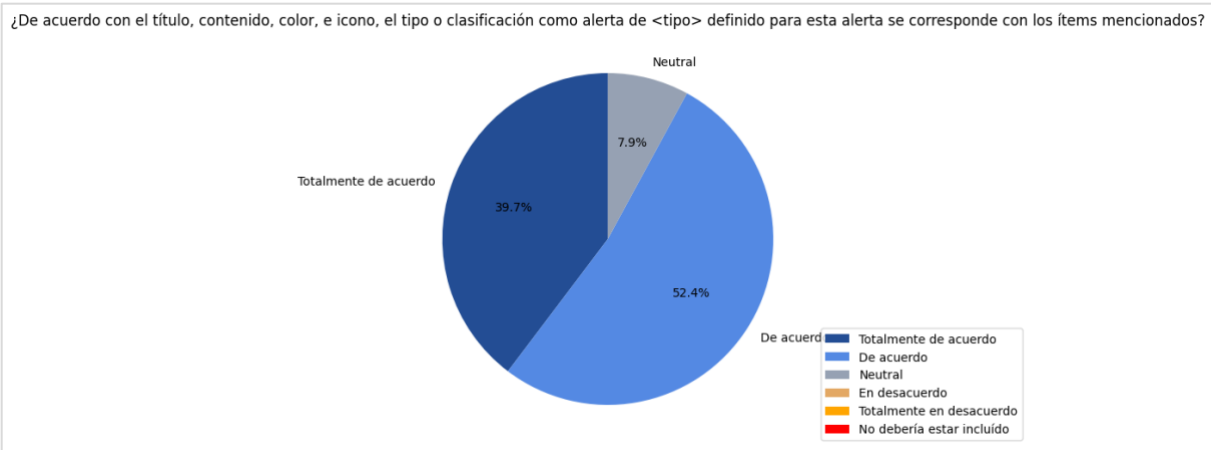


Figura 103. Diagrama general de respuestas para pregunta #8

La distribución por tipo de alerta se muestra en la Figura 104, en donde los tipos Advertencia y Confirmación corresponden a la información mostrada por encima del 90% de las respuestas, y las alertas de *Feedforward* también se consideraron como adecuadas con un 85.7%.

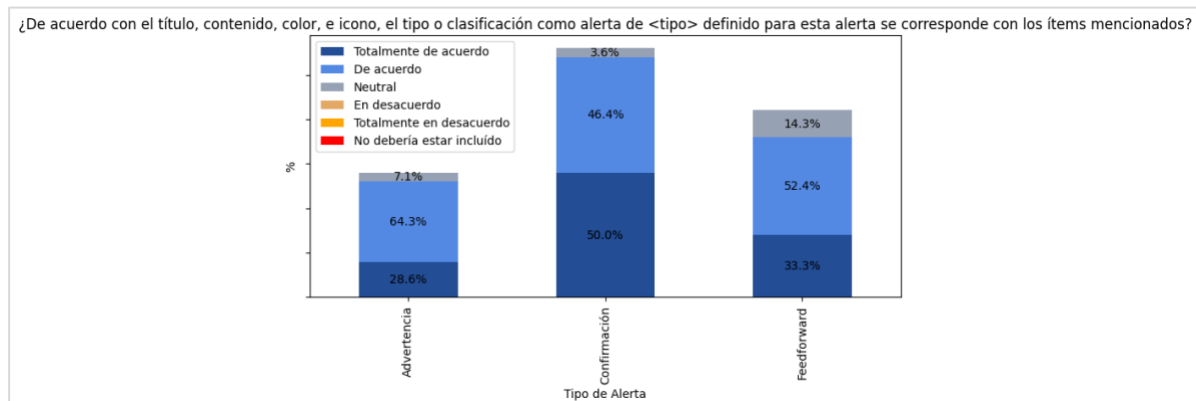


Figura 104. Diagrama de respuestas para pregunta #8 por tipo de alerta

Esta evaluación se centró en validar la correctitud de las alertas de usuario generadas por el componente, la cual está relacionada a la capacidad de las alertas de prevenir errores involuntarios del usuario dentro del sistema. A partir de esto, en todos los casos evaluados, las alertas generadas por el componente demostraron un alto nivel de correctitud. Estas alertas, según el análisis de expertos en el sistema y procesos para las que fueron generadas, ofrecieron información relevante para el usuario ya que, mediante sus atributos como título, contenido, color, ícono y acciones, a consideración de los expertos, permiten al usuario orientarse de manera efectiva dentro del sistema y proceso ejecutado. Esta capacidad de proporcionar orientación contribuye significativamente a la prevención de errores involuntarios, mediante la toma de decisiones de manera informada, lo que resulta crucial para garantizar la experiencia de usuario.

## 6. Conclusiones y trabajo futuro

El objetivo de esta investigación se centró en la creación de un prototipo de componente software independiente, diseñado para abordar la problemática derivada de la evolución constante de flujos de trabajo, que puede impactar las alertas de usuario en un software, derivado de fallas humanas en el análisis de requerimientos y modificaciones, siendo esta problemática abordada mediante el componente software desarrollado con el propósito de generar sugerencias de alertas de usuario a partir de análisis de flujos de trabajo representados en el estándar BPMN, el cual representa un avance significativo en la mejora de la calidad y eficiencia del desarrollo de software, teniendo en cuenta que el enfoque adoptado se basó en la identificación de patrones en distintas aplicaciones y dominios, lo que a su vez le permite adaptarse a diversas situaciones y entornos, por lo cual se puede concluir que:

- Mediante el análisis de las características de alertas de usuario generadas en procesos de diversas aplicaciones software y dominios, es posible identificar patrones comunes y su correlación con el punto específico del proceso en el que surgen, permitiendo encontrar patrones generales de alertas de usuario, los cuales pueden ser aplicados en diversos entornos al ser definidos como una generalización de un conjunto amplio de aplicaciones.
- Las alertas de usuario obtenidas mediante la caracterización y definición de patrones pueden ser automatizadas y generadas de manera dinámica mediante un componente software capaz de identificar los estados por los que atraviesan las instancias de un proceso según el flujo definido, determinando el punto del flujo donde deberían ser presentadas y generando la definición de las características inherentes de las alertas acorde al patrón de alerta de usuario.
- Las alertas de usuario generadas de manera dinámica mediante el prototipo de componente software para los patrones de alerta 1, 2 y 4, presentaron un alto nivel de correctitud según la evaluación aplicada, considerando aspectos asociados al contenido como título, cuerpo del texto y acciones, y el nivel en que estos pueden orientar y/o informar al usuario. Asimismo, se evaluaron aspectos visuales de presentación como íconos y colores, asegurando así un alto grado de correctitud general en las sugerencias generadas, permitiendo disminuir errores involuntarios de los usuarios en su interacción con el sistema.
- Mediante la generación de alertas de usuario basadas en el análisis de un flujo de trabajo y la evaluación de estas, se encontró que el análisis de patrones de flujo generales contribuye a la prevención de problemas en el desarrollo de software que pueden llegar a afectar la correctitud de alertas de usuario, permitiendo disminuir errores en las acciones que los usuarios llevan a cabo en el sistema, dado que las alertas generadas pueden tomar la forma de advertencias o confirmaciones contextuales dentro del flujo, proporcionando así información y orientación al usuario acerca de las posibles repercusiones de sus acciones en el sistema.

Por otra parte, se presentan las limitaciones encontradas en el prototipo generado y se sugieren posibles direcciones para investigaciones futuras, que puedan ampliar y mejorar aún más este enfoque, las cuales se listan a continuación:

- Durante la creación del prototipo se consideraron exclusivamente los patrones de flujo de secuencia, división / unión exclusiva y división / unión paralela. No obstante, existen

otros componentes genéricos y flujos de trabajo que son susceptibles de ser analizados, con el objetivo de identificar patrones adicionales que puedan ser incorporados a las sugerencias generadas por el componente.

- Las estructuras del lenguaje utilizadas para generar elementos textuales siguieron pautas estáticas o con una estructura predefinida, lo que en algunos casos podría resultar en textos con una estructura rígida que pueden interpretarse como un texto robótico generado mediante un algoritmo. En este sentido, una oportunidad de mejora para el componente radica en la generación de textos más humanizados y con un contexto más amplio, lo cual contribuiría a mejorar la legibilidad y la conexión con el usuario que interactúa con dichos textos.
- La identificación de patrones de flujo y de alertas de usuario, podría enriquecerse mediante la ampliación del conjunto de aplicaciones analizadas, dado que, para el desarrollo de este trabajo, se empleó una muestra limitada de aplicaciones por dominio, con el propósito de establecer un punto de partida para la creación del prototipo. No obstante, incorporar un mayor número de aplicaciones por dominio y la inclusión de nuevos dominios pueden resultar en la identificación de nuevos patrones que se puedan agregar a las sugerencias generadas por el componente.
- Finalmente, algunas veces la información relevante que forma parte del contexto de los flujos de trabajo no se modela dentro de los mismos, lo que conlleva a que actividades o decisiones críticas sean representadas de manera similar a otras de menor importancia, lo que podría llevar a que el componente las interprete como equivalentes en términos de relevancia, surgiendo de esto otra oportunidad de mejora relacionada con el enriquecimiento de la información incorporada en los flujos cargados en el componente, lo que permitirá que pueda identificar actividades y decisiones críticas y así identificar y generar sugerencias que consideren adecuadamente estos parámetros.

## 7. Bibliografía

- Aalst, W. van der, Ter Hofstede, A., & Weske, M. (2003). *Business process management: International conference, BPM 2003, Eindhoven, the Netherlands, June 26-27, 2003 : proceedings*. Berlin: Springer.
- Allweyer, T. (2016). *BPMN 2.0: Introduction to the standard for business process modeling* (2nd, updated and extended edition ed.; D. Allweyer, Trad.). Norderstedt: BOD - Books on Demand.
- Cochran, W. G. (1977). *Sampling techniques* (3d ed). New York: Wiley.
- Fernández Casado, P. E. (2018). *Usabilidad web*. Madrid: RA-MA Editorial.
- Galitz, W. O. (2007). *The essential guide to user interface design: An introduction to GUI design principles and techniques* (3rd ed). Indianapolis, IN: Wiley Pub.
- Hogan, B. P. (2009). *Web design for developers: A programmer's guide to design tools and techniques*. Raleigh, N.C: Pragmatic Bookshelf.
- ISO. (2022). Usabilidad. Recuperado 28 de noviembre de 2022, de Usabilidad website: <https://iso25000.com/index.php/normas-iso-25000/iso-25010/23-usabilidad>
- Jayatilleke, S., & Lai, R. (2018). A systematic review of requirements change management. *Information and Software Technology*, 93, 163-185. <https://doi.org/10.1016/j.infsof.2017.09.004>
- Kemp, S. (2023, enero 26). Digital 2023: Global Overview Report. Recuperado 28 de septiembre de 2023, de DataReportal – Global Digital Insights website: <https://datareportal.com/reports/digital-2023-global-overview-report>
- Langer, A. M. (2008). *Analysis and design of information systems* (3rd ed). London: Springer.
- Lohr, S. L. (1999). *Sampling: Design and analysis*. Pacific Grove, Calif.: Duxbury Press.
- MacKenzie, I. S. (2013). *Human-computer interaction: An empirical research perspective* (1st. edition). Amsterdam: Morgan Kaufmann is an imprint of Elsevier.
- McIntire, P. (2008). *Visual Design for the modern web*. Berkeley, CA: New Riders.
- McKay, E. N. (2014). *UI is Communication: How to Design Intuitive, User Centered Interfaces by Focusing on Effective Communication*. Saint Louis: Elsevier Science.
- Norman, D. A. (2013). *The design of everyday things* (Revised and expanded edition). New York, New York: Basic Books.
- Nurmuliani, N., Zowghi, D., & Powell, S. (2004). Analysis of requirements volatility during software development life cycle. *2004 Australian Software Engineering Conference. Proceedings.*, 28-37. Melbourne, Vic., Australia: IEEE. <https://doi.org/10.1109/ASWEC.2004.1290455>
- Object Management Group. (2022). BPMN Specification—Business Process Model and Notation. Recuperado 28 de noviembre de 2022, de BPMN Specification—Business Process Model and Notation website: <https://www.bpmn.org/>
- Object Management Group, O. (2023). About the Business Process Model and Notation Specification Version 2.0.2. Recuperado 16 de mayo de 2023, de About the Business Process Model and Notation Specification Version 2.0.2 website: <https://www.omg.org/spec/BPMN>
- Podmajersky, T. (2019). *Strategic writing for UX: Drive engagement, conversion, and retention with every word* (First edition). Beijing ; Boston: O'Reilly.
- Riehle, D., & Züllighoven, H. (1996). Understanding and using patterns in software development. *Theory and Practice of Object Systems*, 2(1), 3-13. [https://doi.org/10.1002/\(SICI\)1096-9942\(1996\)2:1<3::AID-TAPO1>3.0.CO;2-#](https://doi.org/10.1002/(SICI)1096-9942(1996)2:1<3::AID-TAPO1>3.0.CO;2-#)



- Saffer, D. (2010). *Designing for interaction: Creating innovative applications and devices* (2nd ed). Berkeley, CA: New Riders.
- Silver, B. (2011). *BPMN method and style: With BPMN implementer's guide* (2. ed). Aptos, Calif: Cody-Cassidy Press.
- Statista. (s. f.-a). Apple App Store: Number of available apps by quarter 2022. Recuperado 28 de septiembre de 2023, de Statista website: <https://www.statista.com/statistics/779768/number-of-available-apps-in-the-apple-app-store-quarter/>
- Statista. (s. f.-b). Google Play Store: Number of apps 2023. Recuperado 22 de septiembre de 2023, de Statista website: <https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>
- Van der Aalst, W. M. P., Ter Hofstede, A. H. M., Kiepuszewski, B., & Barros, A. P. (2003). Workflow patterns. *Distributed and Parallel Databases*, 14(1), 5-51. Scopus. <https://doi.org/10.1023/A:1022883727209>
- Visual Paradigm. (2022). What is BPMN? Recuperado 28 de noviembre de 2022, de What is BPMN? website: <https://www.visual-paradigm.com/guide/bpmn/what-is-bpmn/>
- Weske, M. (2019). *Business process management: Concepts, languages, architectures* (Third edition). Berlin [Heidelberg]: Springer. <https://doi.org/10.1007/978-3-662-59432-2>
- Yan, Z., Dijkman, R., & Grefen, P. (2012). Business process model repositories – Framework and survey. *Information and Software Technology*, 54(4), 380-395. <https://doi.org/10.1016/j.infsof.2011.11.005>

## 8. Acta de propiedad intelectual

UNIVERSIDAD DEL CAUCA

FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

### ACTA DE ACUERDO SOBRE LA PROPIEDAD INTELECTUAL DEL TRABAJO DE GRADO

En atención al acuerdo del Honorable Consejo Superior de la Universidad del Cauca, número 008 del 23 de febrero de 1999, donde se estipula todo lo concerniente a la producción intelectual en la institución, los abajo firmantes, reunidos el día 20 de octubre del 2020 en el salón del Consejo de Facultad, acordamos las siguientes condiciones para el desarrollo y posible usufructo del siguiente proyecto.

**Materia del acuerdo:** Trabajo de grado para optar al título de Ingeniero en Sistemas.

**Título del Proyecto:** Prototipo de componente software para generación dinámica de alertas de usuario basado en procesos de negocio

**Objetivo del proyecto:** Proponer una solución informática que permita generar alertas para el usuario de manera dinámica a partir de cambios en el flujo de trabajo de un proceso de negocio.

**Duración del proyecto:** Doce meses.

Los participantes del proyecto, el estudiante de pregrado Brayan Stiven Mamián, identificado con la CC 1.061.794.214, a quien en adelante se le llamará “el estudiante”; la doctora Sandra Milena Roa Martínez en calidad de directora del trabajo de grado, identificada con la CC 63.516.527 a quienes en adelante se les llamará “el docente”, y la Universidad del Cauca, representada por el Decano de la FIET, manifiestan que:

- 1.- La idea original del proyecto es de los docentes, quienes la propusieron y presentaron al Departamento de Sistemas, que la aceptó como tema para el proyecto de grado en referencia.
- 2.- La idea mencionada fue acogida por el estudiante como proyecto para obtener el grado de ingeniero en Sistemas, quienes la desarrollarán bajo la dirección del docente.
- 3.- Los derechos intelectuales y morales corresponden al docente y al estudiante.
- 4.- Los derechos patrimoniales corresponden al docente, al estudiante y a la Universidad del Cauca por partes iguales y continuarán vigentes, aún después de la desvinculación de alguna de las partes de la Universidad.
- 5.- Los participantes se comprometen a cumplir con todas las condiciones de tiempo, recursos, infraestructura, dirección, asesoría, establecidas en el anteproyecto, a estudiar, analizar, documentar y hacer acta de cambios aprobados por el Consejo de Facultad, durante el desarrollo del proyecto, los cuales entran a formar parte de las condiciones generales.
- 6.- El estudiante se compromete a restituir en efectivo y de manera inmediata a la Universidad los aportes recibidos y los pagos hechos por la Institución a terceros por servicios o equipos, si el comité de Investigaciones declara suspendido el proyecto por incumplimiento del cronograma o de las demás obligaciones contraídas por el estudiante; y en cualquier caso de suspensión, la obligación de devolver en el estado en que les fueron proporcionados y de manera inmediata, los equipos de laboratorio, de cómputo y demás bienes suministrados por la Universidad para la realización del proyecto.

**7.-** El docente y el estudiante se comprometen a dar crédito a la Universidad y de hacer mención del Fondo de Fomento de Investigación, en los informes de avance y de resultados, y en registro de éstos, cuando ha habido financiación de la Universidad o del Fondo.

**8.-** Cuando por razones de incumplimiento, legalmente comprobadas, de las condiciones de desarrollo planteadas en el anteproyecto y sus modificaciones, alguno de los participantes deba ser excluido del proyecto, los derechos aquí establecidos concluyen para él. Además, se tendrán en cuenta los principios establecidos en el reglamento estudiantil vigente de la Universidad del Cauca en lo concerniente a la cancelación y la pérdida del derecho a continuar estudios.

**9.-** El documento del anteproyecto y las actas de modificaciones si las hubiere, forman parte integral de la presente acta.

**10.-** Los aspectos no contemplados en la presente acta serán definidos en los términos del acuerdo 008 del 23 de febrero de 1999 expedido por el Consejo Superior de la Universidad del Cauca, del cual los participantes del acuerdo aseguran tener pleno conocimiento.

Decano: \_\_\_\_\_

Director: \_\_\_\_\_

Estudiante: \_\_\_\_\_

# Anexos

## Anexo 1. Abstracciones de procesos y alertas

### 1. Dominio chat y mensajería

En esta sección se realizará la abstracción y captura de alertas para Google Meet, la aplicación elegida para el dominio de chat y mensajería. El patrón de flujo *secuencia* se abstraigo desde el proceso *crear una reunión*, proceso que permite a los usuarios crear videollamadas en la aplicación. Los resultados de la abstracción y alertas de usuario capturadas se muestran en la Tabla anexo 1.

Tabla anexo 1. Abstracción de proceso y alertas de usuario, patrón de flujo secuencia - Google Meet

| Datos del proceso  |  |
|--|--|
| Patrón buscado   | Secuencia  |
| Aplicación   | Google Meet  |
| Proceso  | Crear una reunión  |
| Plataforma   | Aplicación web   |
| Datos de abstracción del proceso   |  |
| ¿Qué inicia el proceso?  | Se inicia cuando un usuario cree una nueva reunión   |
| ¿Qué representa cada instancia o ejecución del proceso?  | Una nueva reunión virtual ha sido creada permitiendo conectarse mediante videollamada a un grupo de personas |
| ¿Qué determina que el proceso está completo?   | Que la reunión haya sido creada  |
| ¿Cuáles son las formas en las que el proceso puede terminar?   | La reunión ha sido creada  |
| Mapeo de actividades de alto nivel   | Ingresar a la opción 'Nueva reunión'<br>Seleccionar 'Iniciar reunión ahora'                                  |
| Diagrama de proceso de alto nivel  |  |
| <pre> graph LR     Actor[Usuario creador] --&gt; Start(( ))     Start --&gt; Step1[Seleccionar crear reunión]     Step1 --&gt; Step2[Seleccionar Iniciar ahora]     Step2 --&gt; End(( ))         </pre> |  |
| Alerta de usuario #1   |  |
| Alerta de usuario  |  |
| Título   | La reunión está lista  |

|                                |   |
|--------------------------------|---|
| <b>Contenido</b>               | También puedes compartir esta información para unirte con las personas que quieras que asistan a la reunión |
| <b>Tipo</b>                    | Confirmación  |
| <b>Color</b>                   | Ninguno   |
| <b>Botones</b>                 | Añadir a alguien  |
| <b>Representación visual</b>   | Caja de texto   |
| <b>Ubicación en el proceso</b> | Fin del proceso   |
| <b>Usuario objetivo</b>        | Usuario creador   |

En este caso particular dentro del proceso *solicitar unirse a una llamada* se mapearon los patrones de flujo división/unión paralela y división/unión exclusiva, la abstracción del proceso y alertas de usuario presentadas se muestran en la Tabla anexo 2.

Tabla anexo 2. Abstracción de proceso y alertas de usuario, patrón de flujo división/unión paralela y división/unión exclusiva - Google Meet

| Datos del proceso   |   |
|---|---|
| <b>Patrón buscado</b>   | División / Unión paralela – División / Unión exclusiva  |
| <b>Aplicación</b>   | Google Meet   |
| <b>Proceso</b>  | Solicitar unirse a una llamada  |
| <b>Plataforma</b>   | Aplicación web  |
| Datos de abstracción del proceso  |   |
| <b>¿Qué inicia el proceso?</b>  | El proceso se inicia cuando una persona solicita unirse a una reunión mediante la opción de 'Solicitar unirse'.                                 |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>  | Cuando la persona que tiene el control de la llamada acepta o rechaza a la persona que solicita unirse.   |
| <b>¿Qué determina que el proceso está completo?</b>   | Representa una persona que solicita ingresar a una reunión.   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>   | El proceso puede terminar con una persona agregada a la reunión o con la solicitud rechazada y el usuario no puede ingresar a la reunión.       |
| <b>Mapeo de actividades de alto nivel</b>   | El usuario que desea ingresar a la llamada entra al sitio web de la llamada.<br>El usuario que está en la llamada acepta o rechaza la solicitud |
| Diagrama de proceso de alto nivel   |   |
| <pre> graph TD     subgraph "Google Meet - Solicitar unirse a una llamada"         subgraph "Solicitante"             Start(( )) --&gt; S[Solicitar unirse]             S --&gt; J1{+}             J1 --&gt; A[Aprobar solicitud de ingreso]             A --&gt; D1{X}             D1 -- Si --&gt; U[Unir usuario solicitante a la llamada]             D1 -- No --&gt; J2{X}             U --&gt; J2             J2 --&gt; End(( ))         end         subgraph "Participante"             J1 --&gt; A             A --&gt; D1             D1 --&gt; J2         end     end </pre> |   |
| Alerta de usuario #1  |   |

|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |    |
| <b>Título</b>                  | Alguien quiere unirse a esta llamada   |
| <b>Contenido</b>               | {Nombre usuario}   |
| <b>Tipo</b>                    | Notificación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | Admitir, Vista   |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Compuerta de decisión exclusiva  |
| <b>Usuario objetivo</b>        | Participante   |
| <b>Alerta de usuario #2</b>    |  |
| <b>Alerta de usuario</b>       |   |
| <b>Título</b>                  | Pidiendo permiso para acceder  |
| <b>Contenido</b>               | Te unirás a la llamada cuando alguien te permita entrar                              |
| <b>Tipo</b>                    | Feedforward  |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Gateway de decisión exclusiva  |
| <b>Usuario objetivo</b>        | Solicitante  |
| <b>Alerta de usuario #3</b>    |  |
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | No puedes unirte a esta llamada  |
| <b>Contenido</b>               | Alguien de la reunión ha rechazado tu solicitud para unirte                          |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |

|                                |   |
|--------------------------------|---|
| <b>Ubicación en el proceso</b> | Compuerta de unión exclusiva, opción 'No admitir' |
| <b>Usuario objetivo</b>        | Solicitante                                       |

## 2. Dominio redes sociales

Para el análisis del dominio redes sociales la aplicación elegida es Facebook, la cual es hoy en día una de las redes sociales más usadas en el mundo. Para el análisis del patrón de flujo de secuencia el proceso elegido es *compartir historia*, en el cual un usuario podrá compartir elementos multimedia a sus contactos por un tiempo limitado de 24h, los resultados de la abstracción del proceso y alertas de usuario del proceso se muestran en la Tabla anexo 3.

Tabla anexo 3. Abstracción de proceso y alertas de usuario, patrón de flujo secuencia - Facebook

| Datos del proceso  |   |
|--|---|
| <b>Patrón buscado</b>  | Secuencia   |
| <b>Aplicación</b>  | Facebook  |
| <b>Proceso</b>   | Compartir historia  |
| <b>Plataforma</b>  | Aplicación móvil Android  |
| Datos de abstracción del proceso   |   |
| <b>¿Qué inicia el proceso?</b>   | El proceso se inicia cuando el usuario selecciona la opción 'Crear historia' dentro de Facebook.  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Cuando la historia haya sido publicada en la plataforma.  |
| <b>¿Qué determina que el proceso está completo?</b>  | Una historia agregada para el perfil del usuario.   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | La historia ha sido publicada   |
| <b>Mapeo de actividades de alto nivel</b>  | El usuario ingresa a la opción 'Crear historia'<br>El usuario carga la foto o ingresa el texto de la historia<br>El usuario agrega edición sobre el elemento cargado<br>El usuario selecciona la opción compartir |
| Diagrama de proceso de alto nivel  |   |
| <pre> graph LR     subgraph Contexto [Facebook - Crear historia]         direction LR         A(( )) --&gt; B[Ingresar al módulo "Crear historia"]         B --&gt; C[Cargar elementos de la historia]         C --&gt; D[Editar elementos cargados]         D --&gt; E[Compartir historia]     end     E --&gt; F[Cargar elementos en la plataforma]     F --&gt; G(( )) </pre> |   |
| Alerta de usuario #1   |   |

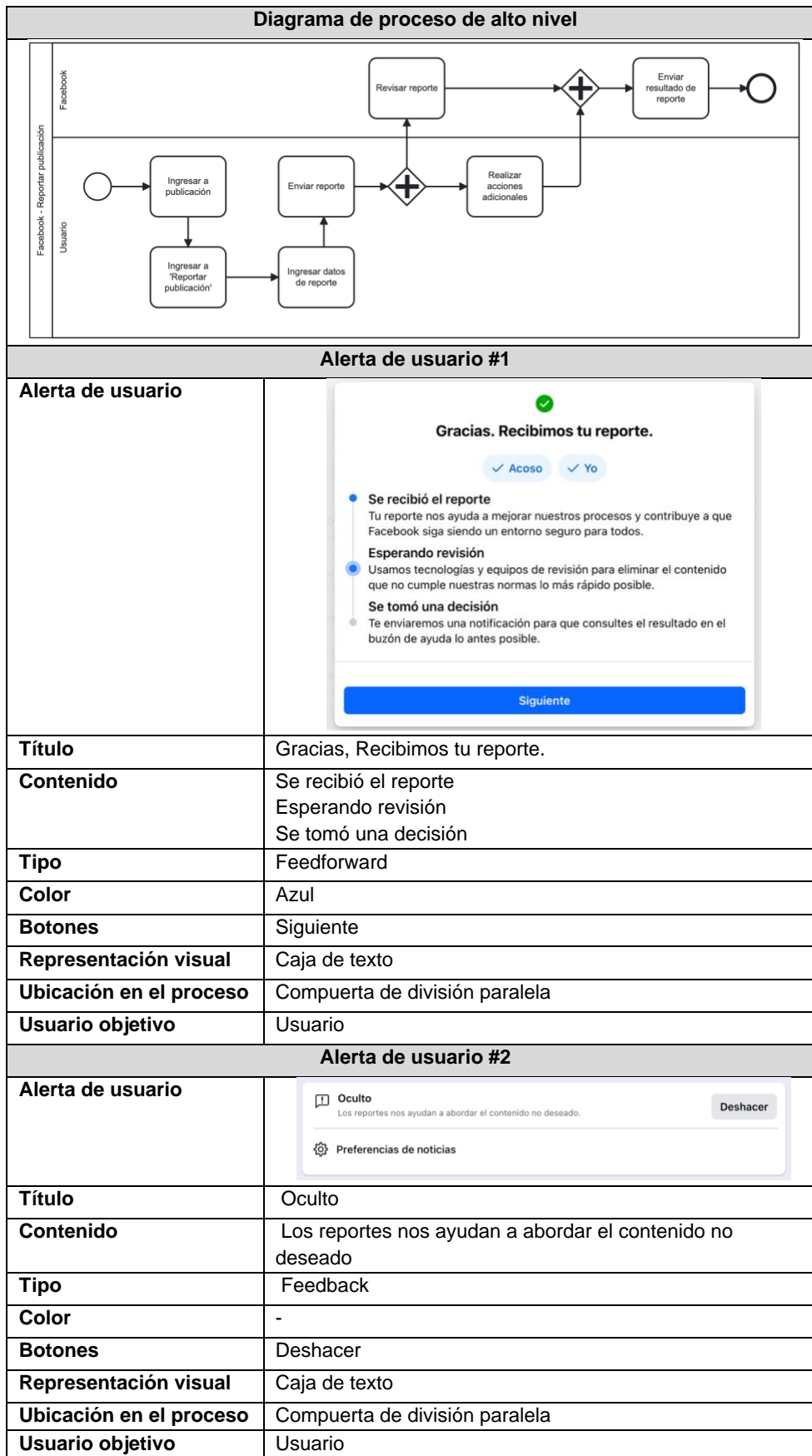
|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | -  |
| <b>Contenido</b>               | Se compartió tu historia   |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Fin del proceso  |
| <b>Usuario objetivo</b>        | Usuario  |

En el caso del patrón de flujo división/unión exclusiva, el proceso elegido es el de *reportar publicación*, en el cual un usuario puede enviar a revisión una publicación potencialmente ofensiva. El resultado de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 4.

Tabla anexo 4. Abstracción de proceso y alertas de usuario, patrón de flujo división/unión paralela - Facebook

| <b>Datos del proceso</b>  |  |
|---|--|
| <b>Patrón buscado</b>   | División / unión paralela  |
| <b>Aplicación</b>   | Facebook   |
| <b>Proceso</b>  | Reportar publicación   |
| <b>Plataforma</b>   | Aplicación web   |
| <b>Datos de abstracción del proceso</b>                             |  |
| <b>¿Qué inicia el proceso?</b>                                      | El proceso se inicia cuando un usuario ingresa a la opción 'Reportar' en una publicación.  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | Cuando desde Facebook se ha tomado una determinación sobre el reporte.   |
| <b>¿Qué determina que el proceso está completo?</b>                 | Representa el reporte de una publicación ante Facebook.  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | El proceso termina con una determinación sobre el caso por parte de Facebook.  |
| <b>Mapeo de actividades de alto nivel</b>                           | Ingresar a la publicación a reportar.<br>Seleccionar la opción 'Reportar'<br>Ingresar los datos del reporte.<br>Enviar reporte para revisión<br>Acciones adicionales (por ejemplo, bloqueo del perfil que publica u ocultar publicaciones del usuario)<br>Resolver reporte |



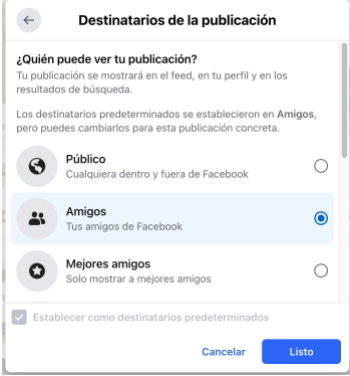
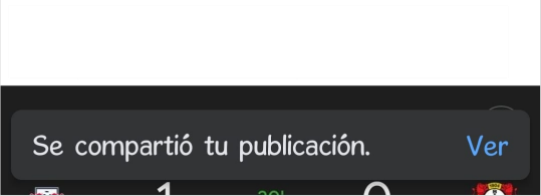


Para el caso del patrón de flujo división/unión exclusiva, el proceso dentro de Facebook elegido es el de *crear publicación*, en el cual el usuario puede crear compartir material

audiovisual a un público determinado dentro de la red social, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 5.

Tabla anexo 5. Abstracción de proceso y alertas de usuario, patrón de flujo división/unión exclusiva - Facebook

| Datos del proceso   |   |
|---|---|
| <b>Patrón buscado</b>   | División / unión exclusiva  |
| <b>Aplicación</b>   | Facebook  |
| <b>Proceso</b>  | Crear publicación   |
| <b>Plataforma</b>   | Aplicación móvil Android  |
| Datos de abstracción del proceso  |   |
| <b>¿Qué inicia el proceso?</b>  | El proceso se inicia cuando el usuario ingresa a la opción 'Crear publicación'  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>  | Cuando la publicación se ha guardado en la plataforma.  |
| <b>¿Qué determina que el proceso está completo?</b>   | Una publicación agregada a Facebook   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>   | Finaliza con la publicación enviada a Facebook.   |
| <b>Mapeo de actividades de alto nivel</b>   | Ingresar a la opción 'Crear publicación'<br>Seleccionar público objetivo (Público, Amigos, Amigos excepto, Solo yo ...etc.)<br>Redactar publicación<br>Publicar |
| Diagrama de proceso de alto nivel   |   |
| <pre> graph TD     subgraph Usuario         Start(( )) --&gt; Ingresar[Ingresar a 'Crear publicación']         Ingresar --&gt; Seleccionar[Seleccionar destinatarios]         Seleccionar --&gt; Redactar[Redactar publicación]         Redactar --&gt; Publicar[Publicar]     end      subgraph Facebook         Publicar --&gt; Tipo{Tipo de publicación}         Tipo -- Publico --&gt; HacerAccesible[Hacer accesible para todo el mundo]         Tipo -- Amigos excepto --&gt; HacerDisponibleTodos[Hacer disponible para todos los amigos excepto]         Tipo -- Amigos --&gt; HacerDisponibleAmigos[Hacer disponible solo para lista de amigos]         Tipo -- Solo yo --&gt; NoHacerDisponible[No hacer disponible para nadie]         HacerAccesible --&gt; Union{ }         HacerDisponibleTodos --&gt; Union         HacerDisponibleAmigos --&gt; Union         NoHacerDisponible --&gt; Union         Union --&gt; End(( ))     end     </pre> |   |
| Alerta de usuario #1  |   |

|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |    |
| <b>Título</b>                  | ¿Quién puede ver tu publicación?   |
| <b>Contenido</b>               | Tu publicación se mostrará en el feed, en tu perfil y en los resultados de búsqueda.<br>Los destinatarios predeterminados se establecieron en Amigos, pero puedes cambiarlos para esta publicación concreta. |
| <b>Tipo</b>                    | Advertencia  |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Actividad 'Seleccionar destinatarios' la cual controla el gateway de división exclusiva  |
| <b>Usuario objetivo</b>        | Usuario  |
| <b>Alerta de usuario #2</b>    |  |
| <b>Alerta de usuario</b>       |    |
| <b>Título</b>                  | -  |
| <b>Contenido</b>               | Se compartió tu publicación  |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | Ver  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Fin del proceso  |
| <b>Usuario objetivo</b>        | Usuario  |

### 3. Dominio motores de búsqueda o portales web

Para el dominio de motores de búsqueda, la aplicación seleccionada es Google Search, uno de los buscadores más grandes que existen en internet. Para el patrón de flujo secuencia, el proceso elegido es el de realizar búsqueda textual, en el cual el usuario puede realizar una búsqueda mediante texto en, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 6.

Tabla anexo 6. Abstracción de proceso y alertas de usuario, patrón de flujo secuencia - Google Search

| Datos del proceso     |           |
|-----------------------|-----------|
| <b>Patrón buscado</b> | Secuencia |

|  |  |
|--|--|
| <b>Aplicación</b>  | Google Search  |
| <b>Proceso</b>   | Realizar búsqueda  |
| <b>Plataforma</b>  | Aplicación web   |
| <b>Datos de abstracción del proceso</b>  |  |
| <b>¿Qué inicia el proceso?</b>   | El proceso se inicia cuando el usuario ingresa a Google Search e ingresa un término de búsqueda. |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Cuando se retornan los resultados de la búsqueda ingresada                                       |
| <b>¿Qué determina que el proceso está completo?</b>  | Representa una búsqueda  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | El proceso puede terminar con resultados o sin resultados para el término ingresado.             |
| <b>Mapeo de actividades de alto nivel</b>  | Ingresar al portal<br>Ingresar un término de búsqueda<br>Mostrar los resultados obtenidos        |
| <b>Diagrama de proceso de alto nivel</b>   |  |
| <pre> graph LR     subgraph Usuario         Start(( )) --&gt; Ingresar[Ingresar término de búsqueda]     end     subgraph Google         Ingresar --&gt; Buscar[Buscar coincidencias]         Buscar --&gt; Mostrar[Mostrar resultados]         Mostrar --&gt; End(( ))     end </pre> |  |
| <b>Alerta de usuario #1</b>  |  |
| <b>Alerta de usuario</b>   |  |
| <b>Título</b>  | -  |
| <b>Contenido</b>   | Cerca de {cantidad} resultados ({tiempo})  |
| <b>Tipo</b>  | Confirmación   |
| <b>Color</b>   | -  |
| <b>Botones</b>   | -  |
| <b>Representación visual</b>   | Palabra simple   |
| <b>Ubicación en el proceso</b>   | Fin del proceso  |
| <b>Usuario objetivo</b>  | Usuario  |

Con respecto al patrón de flujo división/unión paralela, este no fue encontrado entre los procesos asociados al dominio en Google Search, por otra parte el patrón de flujo división/unión exclusiva se mapeó como una generalización del proceso de búsqueda de Google Search, en el cual el usuario puede elegir entre la opción de mostrar todos los resultados o acceder directamente al resultado que Google Search considera más relevante,



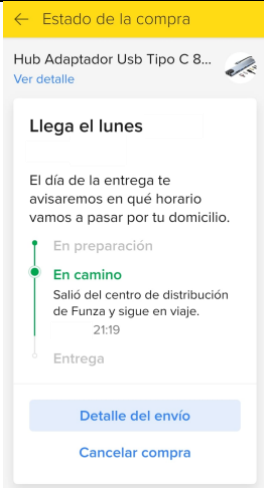

Tabla anexo 8. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - MercadoLibre

| Datos del proceso   |  |
|---|--|
| Patrón buscado  | Secuencia  |
| Aplicación  | MercadoLibre   |
| Proceso   | Enviar pregunta  |
| Plataforma  | Aplicación móvil Android   |
| Datos de abstracción del proceso  |  |
| ¿Qué inicia el proceso?   | El proceso se inicia cuando el usuario ingresa a un producto   |
| ¿Qué representa cada instancia o ejecución del proceso?   | Una pregunta realizada por el usuario.   |
| ¿Qué determina que el proceso está completo?  | Cuando el usuario envía la pregunta hacia el vendedor.   |
| ¿Cuáles son las formas en las que el proceso puede terminar?  | La pregunta se ha enviado correctamente  |
| Mapeo de actividades de alto nivel  | Ingresar a un producto<br>Ingresar a la sección 'Preguntas y respuestas'<br>Redactar pregunta<br>Enviar pregunta |
| Diagrama de proceso de alto nivel   |  |
| <pre> graph LR     Start(( )) --&gt; Ingresar[Ingresar a producto]     Ingresar --&gt; Seccion[Ingresar a sección 'Preguntas']     Seccion --&gt; Redactar[Redactar pregunta]     Redactar --&gt; Enviar[Enviar]     Enviar --&gt; Guardar[Guardar pregunta]     Guardar --&gt; End(( ))     </pre> |  |
| Alerta de usuario #1  |  |
| Alerta de usuario   | Tu pregunta se envió correctamente.  |
| Título  | -  |
| Contenido   | Tu pregunta se envió correctamente   |
| Tipo  | Confirmación   |
| Color   | Verde  |
| Botones   | -  |
| Representación visual   | Caja de texto  |
| Ubicación en el proceso   | Fin del proceso  |
| Usuario objetivo  | Usuario  |

En cuanto al patrón de flujo de división/unión paralela, el proceso elegido es el de *realizar compra*, mediante el cual un usuario puede realizar la compra de uno o varios artículos y estos se envían a su domicilio, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 9.

Tabla anexo 9. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión paralela - MercadoLibre

| Datos del proceso  |   |
|--|---|
| Patrón buscado   | División / unión paralela   |
| Aplicación   | MercadoLibre  |
| Proceso  | Realizar compra   |
| Plataforma   | Aplicación web  |
| Datos de abstracción del proceso   |   |
| ¿Qué inicia el proceso?  | El proceso se inicia cuando el usuario agrega un producto a el carrito de compras.  |
| ¿Qué representa cada instancia o ejecución del proceso?  | Una compra realizada por el usuario.  |
| ¿Qué determina que el proceso está completo?   | Cuando el usuario confirma la recepción del producto  |
| ¿Cuáles son las formas en las que el proceso puede terminar?   | El producto ha sido entregado al usuario  |
| Mapeo de actividades de alto nivel   | Agregar un producto al carrito<br>Realizar el pago del producto<br>El vendedor prepara y envía el producto<br>El comprador puede verificar el estado del envío<br>El comprador confirma la recepción del producto |
| Diagrama de proceso de alto nivel  |   |
| <pre> graph TD     subgraph "MercadoLibre - Realizar compra"         direction TB         C1(( )) --&gt; A[Agregar producto a carrito]         A --&gt; P[Realizar pago de producto]         P --&gt; J1{+}         J1 --&gt; V[Verificar estado del envío]         V --&gt; J2{+}         J2 --&gt; R[Confirmar recepción]         R --&gt; E2(( ))     end     subgraph "Vendedor"         direction TB         P1[Preparar envío] --&gt; E1[Enviar producto]     end     J1 --&gt; P1     E1 --&gt; J2 </pre> |   |
| Alerta de usuario #1   |   |
| Alerta de usuario  |   |
| Título   | -   |
| Contenido  | Te avisaremos cuando tu compra esté en camino   |
| Tipo   | Feedforward   |
| Color  | Verde   |
| Botones  | -   |
| Representación visual  | Caja de texto   |

|                                |  |
|--------------------------------|--|
| <b>Ubicación en el proceso</b> | Compuerta de división paralela   |
| <b>Usuario objetivo</b>        | Usuario  |
| <b>Alerta de usuario #2</b>    |  |
| <b>Alerta de usuario</b>       |    |
| <b>Título</b>                  | Llega el {fecha}   |
| <b>Contenido</b>               | El día de la entrega te avisaremos en qué horario vamos a pasar por tu domicilio     |
| <b>Tipo</b>                    | Feedforward  |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | Detalle del envío, Cancelar compra   |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Compuerta de división (desde la perspectiva del usuario objetivo)                    |
| <b>Usuario objetivo</b>        | Usuario  |
| <b>Alerta de usuario #3</b>    |  |
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | Entregamos tu compra de {producto}   |
| <b>Contenido</b>               | Si lo necesitas, puedes devolverla   |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Fin del proceso  |
| <b>Usuario objetivo</b>        | Usuario  |

En relación con el patrón de flujo de división/unión exclusiva, el proceso elegido es el de *realizar pago*, en el cual el usuario puede pagar mediante su método de pago preferido un producto o una canasta de productos, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 10.



Tabla anexo 10. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - MercadoLibre

| Datos del proceso  |   |
|--|---|
| Patrón buscado   | División / unión exclusiva  |
| Aplicación   | MercadoLibre  |
| Proceso  | Realizar pago   |
| Plataforma   | Aplicación web  |
| Datos de abstracción del proceso                             |   |
| ¿Qué inicia el proceso?                                      | El proceso se inicia cuando el usuario inicia el pago de un producto o de un carrito de compras.  |
| ¿Qué representa cada instancia o ejecución del proceso?      | Un pago realizado por un usuario para un producto o un carrito de compras.  |
| ¿Qué determina que el proceso está completo?                 | Que el pago del producto ha sido exitoso.   |
| ¿Cuáles son las formas en las que el proceso puede terminar? | El pago se ha aplicado  |
| Mapeo de actividades de alto nivel                           | Seleccionar un medio de pago (Tarjeta, efectivo)<br>Enviar<br>De acuerdo con el medio de pago seleccionado, ingresar la información requerida<br>Enviar el pago y finalizar |
| Diagrama de proceso de alto nivel                            |   |
|  |   |
| Alerta de usuario #1   |   |
| Alerta de usuario  |   |
| Título   | -   |
| Contenido  | Los intereses serán aplicados por tu entidad bancaria   |
| Tipo   | Advertencia   |
| Color  | -   |
| Botones  | -   |
| Representación visual  | Palabra simple  |

|                                |                                    |
|--------------------------------|------------------------------------|
| <b>Ubicación en el proceso</b> | Actividad 'Enviar (procesar pago)' |
| <b>Usuario objetivo</b>        | Usuario                            |

## 5. Dominio servicios de ubicación

En el dominio de servicios de ubicación, la aplicación elegida es Google Maps, plataforma que facilita la búsqueda de lugares e indicaciones. El patrón de flujo de secuencia elegido es el de *obtener indicaciones* en el cual el usuario puede obtener sugerencias para llegar a un sitio de acuerdo con un medio de transporte, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 11.

Tabla anexo 11. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Google Maps

| Datos del proceso  |   |
|--|---|
| <b>Patrón buscado</b>  | Secuencia   |
| <b>Aplicación</b>  | Google Maps   |
| <b>Proceso</b>   | Obtener indicaciones  |
| <b>Plataforma</b>  | Aplicación web  |
| Datos de abstracción del proceso   |   |
| <b>¿Qué inicia el proceso?</b>   | El proceso se inicia cuando el usuario busca un sitio en la aplicación.   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Una sugerencia de la aplicación al usuario para llegar a un determinado lugar.  |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando la aplicación muestra la ruta sugerida para llegar al lugar.   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | Se muestran las indicaciones al usuario   |
| <b>Mapeo de actividades de alto nivel</b>  | Buscar un sitio en el mapa<br>Seleccionar la opción 'Indicaciones'<br>Seleccionar el medio de transporte<br>Mostrar ruta sugerida |
| Diagrama de proceso de alto nivel  |   |
| <pre> graph LR     subgraph Usuario         Start(( )) --&gt; B1[Buscar un sitio]         B1 --&gt; B2[Seleccionar Indicaciones]         B2 --&gt; B3[Seleccionar un medio de transporte]     end     subgraph Google_Maps         C1[Calcular la mejor ruta] --&gt; C2[Mostrar resultados]     end     B3 --&gt; C1     C2 --&gt; End(( ))           </pre> |   |
| Alertas de usuario   |   |
| Durante la ejecución de este proceso no se encontraron alertas de usuario  |   |

En la revisión de Google Maps no se encontraron procesos asociados al dominio que incluyeran el patrón de flujo división/unión paralela. Por otra parte, para el patrón de flujo división/unión exclusiva el proceso elegido es el de *seleccionar tipo de vista*, en el cual el usuario puede cambiar la vista de la interfaz entre las opciones mapa y satélite, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 12.

Tabla anexo 12. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - Google Maps

| Datos del proceso  |  |
|--|--|
| <b>Patrón buscado</b>  | División / unión exclusiva   |
| <b>Aplicación</b>  | Google Maps  |
| <b>Proceso</b>   | Seleccionar tipo de vista  |
| <b>Plataforma</b>  | Aplicación web   |
| Datos de abstracción del proceso   |  |
| <b>¿Qué inicia el proceso?</b>   | El proceso se inicia cuando el usuario selecciona un tipo de vista en el mapa                        |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Un cambio en la forma en la que se presenta el mapa  |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando se muestra la vista seleccionada  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | Se ha cambiado el tipo de vista  |
| <b>Mapeo de actividades de alto nivel</b>  | Seleccionar un tipo de vista (Mapa, Satélite)<br>Cargar la vista de acuerdo con el tipo seleccionado |
| Diagrama de proceso de alto nivel  |  |
| <pre> graph TD     subgraph Actor [Usuario]         Start(( )) --&gt; Select[Seleccionar tipo de vista]     end     subgraph Actor [Google Maps - Seleccionar tipo de vista]         direction TB         Split{X}         LoadMap[Cargar vista y elementos de mapa]         LoadSat[Cargar vista y elementos de satélite]         Join{X}         Show[Mostrar resultados]         End(( ))         Split -- Mapa --&gt; LoadMap         Split -- Satélite --&gt; LoadSat         LoadMap --&gt; Join         LoadSat --&gt; Join         Join --&gt; Show         Show --&gt; End     end     Select --&gt; Split     </pre> |  |
| Alertas de usuario   |  |
| Durante la ejecución de este proceso no se presentaron historias de usuario  |  |

## 6. Dominio correo electrónico

Para el dominio de correo electrónico, la aplicación elegida es Gmail, uno de los servicios de correo electrónico más reconocidos a nivel mundial. Para el patrón de flujo de secuencia, el proceso elegido es *archivar conversaciones*, en el cual un usuario puede trasladar correos de su bandeja de entrada a un archivo de correos poco importantes, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 13.

Tabla anexo 13. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Gmail

| Datos del proceso     |                          |
|-----------------------|--------------------------|
| <b>Patrón buscado</b> | Secuencia                |
| <b>Aplicación</b>     | Gmail                    |
| <b>Proceso</b>        | Archivar conversaciones  |
| <b>Plataforma</b>     | Aplicación móvil Android |

| Datos de abstracción del proceso                             |  |
|--|--|
| ¿Qué inicia el proceso?                                      | El proceso se inicia cuando el usuario ingresa a la bandeja de entrada de correo.                  |
| ¿Qué representa cada instancia o ejecución del proceso?      | Uno o más correos enviados a la bandeja de archivados.   |
| ¿Qué determina que el proceso está completo?                 | Cuando se han archivado las conversaciones.  |
| ¿Cuáles son las formas en las que el proceso puede terminar? | Se han archivado las conversaciones  |
| Mapeo de actividades de alto nivel                           | Ingresar a la bandeja de entrada<br>Seleccionar correos (conversaciones)<br>Seleccionar 'Archivar' |
| Diagrama de proceso de alto nivel                            |  |
|  |  |
| Alerta de usuario #1   |  |
| Alerta de usuario  |  |
| Título   | -  |
| Contenido  | {cantidad} archivadas  |
| Tipo   | Confirmación   |
| Color  | -  |
| Botones  | Deshacer   |
| Representación visual  | Caja de texto  |
| Ubicación en el proceso                                      | Fin del proceso  |
| Usuario objetivo   | Usuario  |

El patrón de flujo de división/unión paralela no se encontró dentro de los procesos asociados al dominio en Gmail, por otra parte, para el patrón de flujo de división/unión exclusiva, el proceso elegido es *enviar correo*, en donde el usuario puede enviar un correo electrónico eligiendo si enviarlo de inmediato o programado, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 14.

Tabla anexo 14. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - Gmail

| Datos del proceso |                            |
|-------------------|----------------------------|
| Patrón buscado    | División / unión exclusiva |
| Aplicación        | Gmail                      |
| Proceso           | Enviar correo              |

|  |  |
|--|--|
| <b>Plataforma</b>  | Aplicación móvil Android   |
| <b>Datos de abstracción del proceso</b>  |  |
| <b>¿Qué inicia el proceso?</b>   | El proceso se inicia cuando el usuario ingresa a la opción 'Redactar'                                  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Un correo electrónico enviado hacia un usuario.  |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando el correo se ha enviado al usuario objetivo.  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | El correo ha sido enviado  |
| <b>Mapeo de actividades de alto nivel</b>  | Ingresar a la opción redactar<br>Ingresar destinatario<br>Ingresar asunto<br>Redactar correo<br>Enviar |
| <b>Diagrama de proceso de alto nivel</b>   |  |
| <pre> graph LR     subgraph Usuario         Start(( )) --&gt; A[Ingresar a opción 'Redactar']         A --&gt; B[Agregar información de correo]         B --&gt; C[Seleccionar opción enviar]     end     subgraph Gmail         C --&gt; D{X}         D --&gt; E[Programar envío]         D --&gt; F[Enviar correo]         E --&gt; G{X}         G --&gt; F         F --&gt; End(( ))     end </pre> |  |
| <b>Alerta de usuario #1</b>  |  |
| <b>Alerta de usuario</b>   |  |
| <b>Título</b>  | -  |
| <b>Contenido</b>   | Enviado  |
| <b>Tipo</b>  | Confirmación   |
| <b>Color</b>   | -  |
| <b>Botones</b>   | Deshacer   |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b>   | Fin del proceso  |
| <b>Usuario objetivo</b>  | Usuario  |
| <b>Alerta de usuario #2</b>  |  |
| <b>Alerta de usuario</b>   |  |
| <b>Título</b>  | -  |
| <b>Contenido</b>   | Envío programado para {fecha programa}   |

|                                |                                 |
|--------------------------------|---------------------------------|
| <b>Tipo</b>                    | Feedforward                     |
| <b>Color</b>                   | -                               |
| <b>Botones</b>                 | Deshacer                        |
| <b>Representación visual</b>   | Caja de texto                   |
| <b>Ubicación en el proceso</b> | Compuerta de división exclusiva |
| <b>Usuario objetivo</b>        | Usuario                         |

## 7. Dominio música

En el dominio de música la aplicación elegida es Spotify, una plataforma de streaming de música que cuenta con millones de canciones. Para el patrón de flujo de secuencia el proceso elegido es *agregar canción a me gusta* en el cual el usuario puede agregar una canción del catálogo a una lista especial personalizada de acuerdo con sus gustos, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 15.

Tabla anexo 15. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Spotify

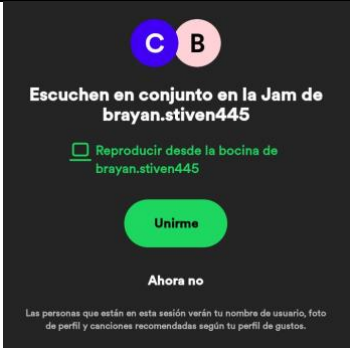
| Datos del proceso   |  |
|---|--|
| <b>Patrón buscado</b>   | Secuencia  |
| <b>Aplicación</b>   | Spotify  |
| <b>Proceso</b>  | Agregar canción a me gusta   |
| <b>Plataforma</b>   | Aplicación móvil Android   |
| Datos de abstracción del proceso  |  |
| <b>¿Qué inicia el proceso?</b>  | El proceso se inicia cuando un usuario selecciona el menú desplegable sobre una canción. |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>  | Una canción agregada a la lista de 'Me gusta' personal del usuario.                      |
| <b>¿Qué determina que el proceso está completo?</b>   | Cuando la canción se haya agregado a la lista de 'Me gusta'.                             |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>   | Finaliza con la canción agregada a la lista  |
| <b>Mapeo de actividades de alto nivel</b>   | Ingresar al menú desplegable de la canción<br>Seleccionar la opción 'Añadir a me gusta'  |
| Diagrama de proceso de alto nivel   |  |
| <pre> graph LR     subgraph Spotify         A[Añadir canción a lista de 'Me gusta'] --&gt; B(( ))     end     subgraph Usuario         C(( )) --&gt; D[Ingresar a menú de la canción]         D --&gt; E[Seleccionar 'Añadir a me gusta']     end     E --&gt; A   </pre> |  |
| Alerta de usuario #1  |  |
| <b>Alerta de usuario</b>  |  |

|                                |                          |
|--------------------------------|--------------------------|
| <b>Título</b>                  | -                        |
| <b>Contenido</b>               | Se agregó a Tus me gusta |
| <b>Tipo</b>                    | Confirmación             |
| <b>Color</b>                   | -                        |
| <b>Botones</b>                 | -                        |
| <b>Representación visual</b>   | Caja de texto            |
| <b>Ubicación en el proceso</b> | Fin del proceso          |
| <b>Usuario objetivo</b>        | Usuario                  |

Con respecto al patrón de flujo división/unión paralela, el proceso elegido es *crear sesión Jam* en la cual crea una sala en donde varios usuarios pueden reproducir música de manera simultánea en una experiencia compartida, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 16.

Tabla anexo 16. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión paralela - Spotify

| Datos del proceso  |   |
|--|---|
| <b>Patrón buscado</b>  | División / unión paralela   |
| <b>Aplicación</b>  | Spotify   |
| <b>Proceso</b>   | Crear sesión de Jam   |
| <b>Plataforma</b>  | Aplicación móvil Android  |
| Datos de abstracción del proceso   |   |
| <b>¿Qué inicia el proceso?</b>   | El proceso se inicia cuando un usuario crea una sesión de Jam.  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Representa una sala creada con usuarios agregando canciones a la sesión   |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando un usuario se une a la sesión de Jam del usuario   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | Finaliza con los dos usuarios en la sesión.   |
| <b>Mapeo de actividades de alto nivel</b>  | Ingresar a la opción de búsqueda<br>Introducir un término de búsqueda en el buscador<br>Buscar coincidencias<br>Mostrar lista de resultados |
| Diagrama de proceso de alto nivel  |   |
| <pre> graph TD     subgraph Spotify         C1[Crear sesión de Jam] --&gt; C2[Habilitar código de Jam]         C2 --&gt; C3[Unir usuario a Jam]     end     subgraph Invitado         A1[Abrir invitación] --&gt; C3     end     subgraph Creador         S(( )) --&gt; C4[Iniciar Jam]         C4 --&gt; C1         C5[Enviar invitación] --&gt; C1         C6{+} --&gt; C5         C6 --&gt; A1         C7{+} --&gt; C3         C7 --&gt; E(( ))     end     C1 --&gt; C2     C2 --&gt; C3     C3 --&gt; C4     C4 --&gt; C5     C5 --&gt; C6     C6 --&gt; A1     A1 --&gt; C3     C6 --&gt; C7     C7 --&gt; C3     C7 --&gt; E </pre> |   |
| Alerta de usuario #1   |   |

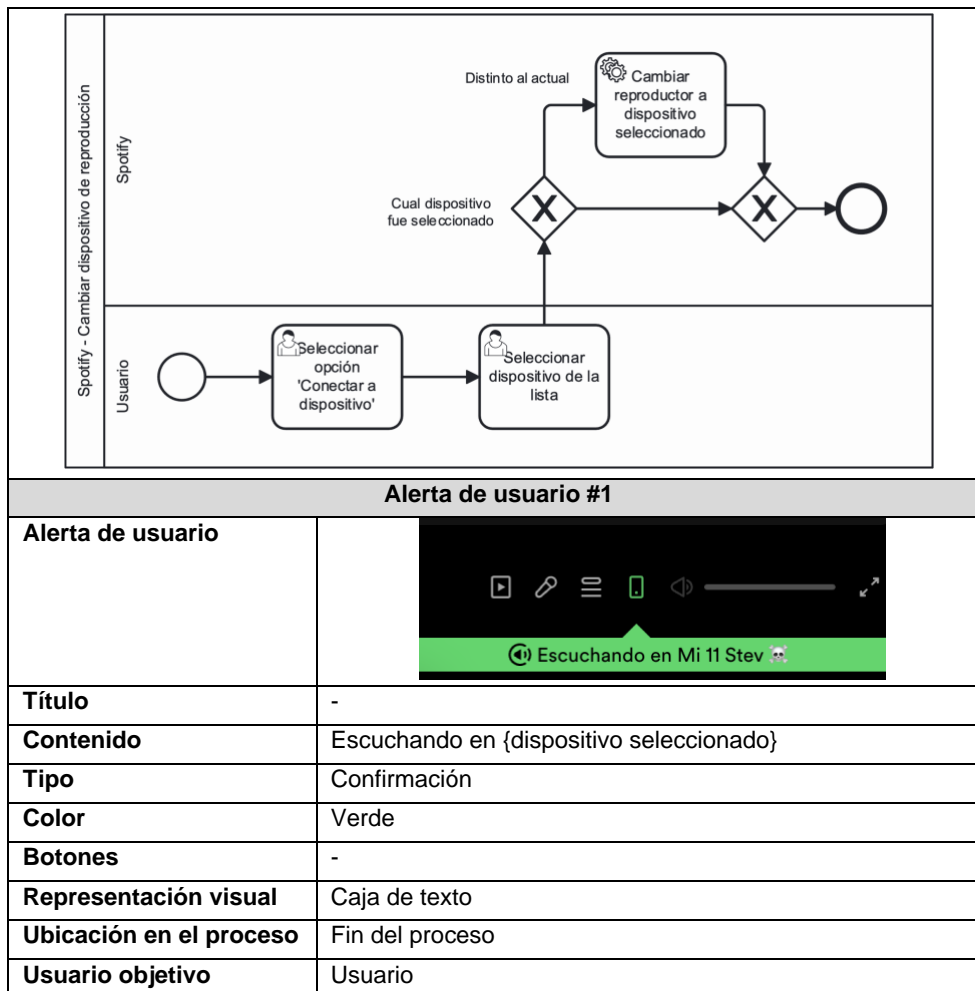
|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | -  |
| <b>Contenido</b>               | Escuchen en conjunto en la Jam de {usuario creador}                                |
| <b>Tipo</b>                    | Feedforward  |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | Unirme, Ahora no   |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Actividad 'Abrir invitación'   |
| <b>Usuario objetivo</b>        | Invitado   |

En cuanto al patrón de flujo división/unión exclusiva, el proceso elegido es *cambiar dispositivo de reproducción* en donde el usuario podrá elegir el dispositivo desde donde se emitirá el sonido cuando tenga varias sesiones abiertas con la misma cuenta, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 17.

Tabla anexo 17. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - Spotify

| <b>Datos del proceso</b>  |  |
|---|--|
| <b>Patrón buscado</b>   | División / unión exclusiva   |
| <b>Aplicación</b>   | Spotify  |
| <b>Proceso</b>  | Cambiar dispositivo de reproducción  |
| <b>Plataforma</b>   | Aplicación nativa MacOS  |
| <b>Datos de abstracción del proceso</b>                             |  |
| <b>¿Qué inicia el proceso?</b>                                      | El proceso se inicia cuando el usuario ingresa a la opción 'Conectar a un dispositivo'   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | Un cambio en el dispositivo utilizado para la reproducción de música   |
| <b>¿Qué determina que el proceso está completo?</b>                 | Cuando el dispositivo de reproducción se ha cambiado correctamente   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | Con el dispositivo de reproducción cambiado o utilizando el dispositivo actual.  |
| <b>Mapeo de actividades de alto nivel</b>                           | Ingresar a la opción 'Conectar a un dispositivo'<br>Seleccionar un dispositivo de la lista<br>Si el dispositivo es distinto al actual, cambiar la bocina de reproducción |
| <b>Diagrama de proceso de alto nivel</b>                            |  |





## 8. Dominio noticias

Para el dominio de noticias, la aplicación elegida es Microsoft News, la cual proporciona acceso a noticias con cobertura mundial a través de internet. Para el patrón de flujo de secuencia el proceso elegido es *compartir noticia en Twitter*, mediante la cual el usuario puede compartir una noticia de la plataforma a través de Twitter o X (actualmente), los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 18.

Tabla anexo 18. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Microsoft News

| Datos del proceso  |  |
|--|--|
| <b>Patrón buscado</b>  | Secuencia  |
| <b>Aplicación</b>  | Microsoft News   |
| <b>Proceso</b>   | Compartir noticia en Twitter   |
| <b>Plataforma</b>  | Aplicación web   |
| Datos de abstracción del proceso                               |  |
| <b>¿Qué inicia el proceso?</b>                                 | El proceso se inicia cuando el usuario ingresa a una noticia   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b> | Una noticia de Microsoft News cargada desde Twitter  |
| <b>¿Qué determina que el proceso está completo?</b>            | Cuando la noticia se ha cargado correctamente en la red de Twitter y el proceso se continúa desde esta aplicación. |

|   |   |
|---|---|
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>   | Con la noticia cargada desde Twitter y lista para ser publicada             |
| <b>Mapeo de actividades de alto nivel</b>   | Ingresar a una noticia desde el listado<br>Seleccionar compartir en Twitter |
| <b>Diagrama de proceso de alto nivel</b>  |   |
| <pre> graph LR     subgraph Usuario         Start(( )) --&gt; A[Ingresar a una noticia en el listado]         A --&gt; B[Seleccionar opción compartir en Twitter]     end     subgraph Microsoft_News         B --&gt; C[Cargar vista externa de Twitter con la noticia]         C --&gt; End(( ))     end </pre> |   |
| <b>Alerta de usuario #1</b>   |   |
| Durante la ejecución de este proceso no se generaron alertas de usuario   |   |

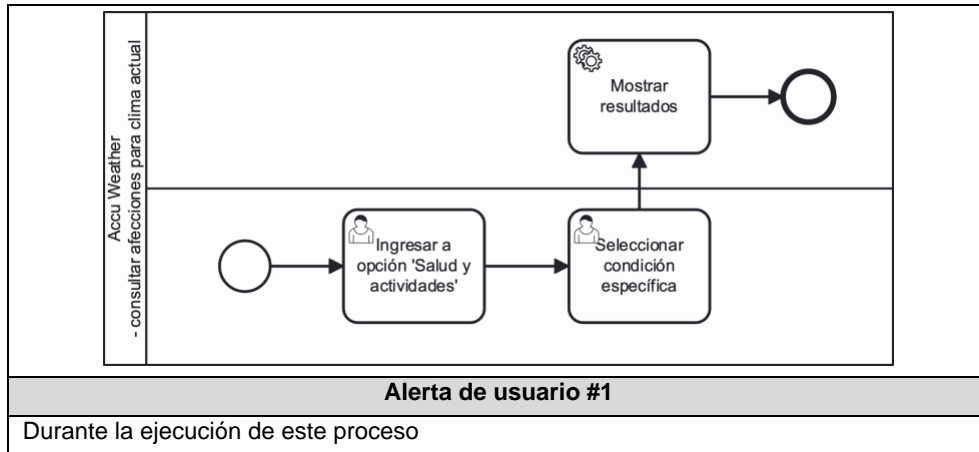
En el caso de los patrones de flujo de división/unión paralela y exclusiva, estos no se encontraron dentro de los procesos asociados al dominio en Microsoft News.

## 9. Dominio clima

Entre las aplicaciones del dominio de clima, la elegida fue Accu Weather, la cual provee información y pronósticos meteorológicos con alertas personalizadas a través de internet. Para la abstracción del patrón de flujo de secuencia el proceso elegido es *comprobar afecciones de clima por condición*, en la cual el usuario podrá consultar cómo le podría afectar las condiciones de clima de acuerdo con enfermedades o padecimientos puntuales, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 19.

Tabla anexo 19. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Accu Weather

| <b>Datos del proceso</b>  |   |
|---|---|
| <b>Patrón buscado</b>   | Secuencia   |
| <b>Aplicación</b>   | Accu Weather  |
| <b>Proceso</b>  | Comprobar afecciones de clima por condición   |
| <b>Plataforma</b>   | Aplicación web  |
| <b>Datos de abstracción del proceso</b>                             |   |
| <b>¿Qué inicia el proceso?</b>                                      | El proceso se inicia cuando el usuario ingresa a la opción de 'Salud y actividades'   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | La visualización de las condiciones que se verán afectadas por el clima actual.       |
| <b>¿Qué determina que el proceso está completo?</b>                 | Cuando se presenta al usuario las condiciones de una afección específica              |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | Con la presentación de condiciones afectadas por el clima actual                      |
| <b>Mapeo de actividades de alto nivel</b>                           | Ingresar a la opción 'Salud y actividades'<br>Mostrar resultados para el clima actual |
| <b>Diagrama de proceso de alto nivel</b>                            |   |



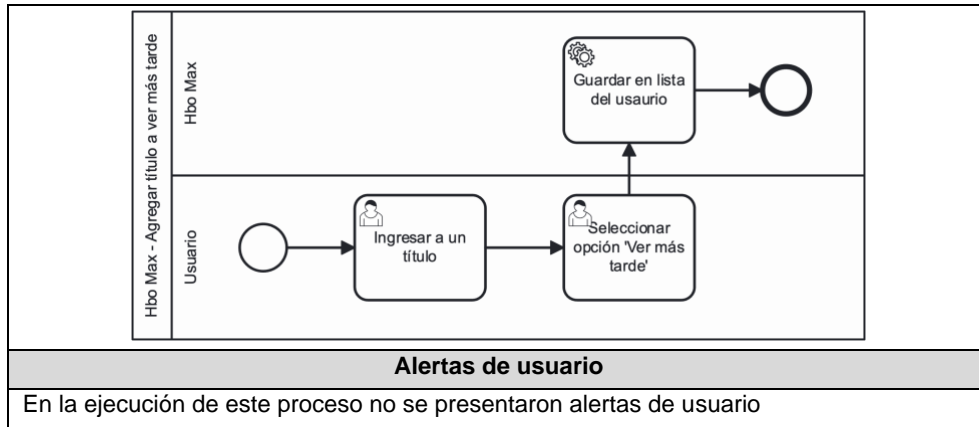
En esta aplicación (Accu Weather) no se encontraron procesos que incluyeran los patrones de flujo división/unión paralela.

## 10. Dominio entretenimiento

Para el dominio de entretenimiento la aplicación elegida es Hbo Max plataforma de streaming audiovisual con un amplio catálogo de películas, series y contenido exclusivo. Para el patrón de flujo secuencia el proceso elegido es *agregar un título a ver más tarde*, en el que un usuario puede agregar una película o serie a una lista personalizada donde podrá encontrarlos más adelante, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 20.

Tabla anexo 20. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Hbo Max

| Datos del proceso   |   |
|---|---|
| <b>Patrón buscado</b>   | Secuencia   |
| <b>Aplicación</b>   | Hbo Max   |
| <b>Proceso</b>  | Agregar un título a ver más tarde   |
| <b>Plataforma</b>   | Aplicación nativa MacOS   |
| Datos de abstracción del proceso                                    |   |
| <b>¿Qué inicia el proceso?</b>                                      | Cuando el usuario ingresa a un título (película o serie)                              |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | Un título agregado a la lista personal del usuario                                    |
| <b>¿Qué determina que el proceso está completo?</b>                 | El título está disponible en la lista personal del usuario                            |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | El título ha sido agregado a la lista del usuario                                     |
| <b>Mapeo de actividades de alto nivel</b>                           | Ingresar a un título (película o serie)<br>Seleccionar la opción 'Agregar a la lista' |
| Diagrama de proceso de alto nivel                                   |   |



Con respecto al patrón de flujo división/unión paralela, el proceso elegido es *descargar un título*, en el cual el usuario puede descargar una película o serie para poder visualizarla sin requerir conexión a internet, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 21.

Tabla anexo 21. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión paralela - Hbo Max

| Datos del proceso   |   |
|---|---|
| <b>Patrón buscado</b>   | División / Unión paralela   |
| <b>Aplicación</b>   | Hbo Max   |
| <b>Proceso</b>  | Descargar un título   |
| <b>Plataforma</b>   | Aplicación nativa MacOS   |
| Datos de abstracción del proceso                                    |   |
| <b>¿Qué inicia el proceso?</b>                                      | Cuando el usuario ingresa a un título (película o serie)  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | Un título disponible para ser reproducido sin necesidad de internet   |
| <b>¿Qué determina que el proceso está completo?</b>                 | El título está disponible para su acceso sin conexión   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | El título está descargado   |
| <b>Mapeo de actividades de alto nivel</b>                           | Ingresar a un título (película o serie)<br>Seleccionar la opción 'Descargar'<br>Visualizar el título en streaming<br>Cuando el título se descargue, cambiar el modo a visualización local |
| Diagrama de proceso de alto nivel                                   |   |
|   |   |

| Alertas de usuario   |  |
|--|--|
| En la ejecución de este proceso no se presentaron alertas de usuario |  |

En el caso del patrón de flujo de división/unión exclusiva, el proceso elegido es el de *filtrar por tipo de contenido*, en el cual el usuario podrá modificar las películas y series que se muestren en la lista de acuerdo con una categoría, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 22.

Tabla anexo 22. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - Hbo Max

| Datos del proceso  |   |
|--|---|
| <b>Patrón buscado</b>  | División / Unión exclusiva  |
| <b>Aplicación</b>  | Hbo Max   |
| <b>Proceso</b>   | Filtrar por tipo de contenido   |
| <b>Plataforma</b>  | Aplicación nativa MacOS   |
| Datos de abstracción del proceso   |   |
| <b>¿Qué inicia el proceso?</b>   | Cuando el usuario ingresa a la aplicación   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Los títulos mostrados al usuario corresponden al tipo seleccionado                                |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando se han mostrado todos los títulos que pertenecen al tipo seleccionado                      |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | Los títulos que se presentan son del tipo seleccionado  |
| <b>Mapeo de actividades de alto nivel</b>  | Ingresar a la aplicación<br>Seleccionar un tipo de contenido<br>Mostrar títulos asociados al tipo |
| Diagrama de proceso de alto nivel  |   |
| <pre> graph TD     subgraph Usuario         Start(( )) --&gt; Entrar[Entrar a lista de títulos]         Entrar --&gt; Seleccionar[Seleccionar tipo de contenido]     end     subgraph Hbo_Max         Seleccionar -- Tipo de contenido --&gt; Union((X))         Union -- Películas --&gt; FiltrarPel[Filtrar por películas]         Union -- Series --&gt; FiltrarSer[Filtrar por series]         FiltrarPel --&gt; Interseccion((X))         FiltrarSer --&gt; Interseccion         Interseccion --&gt; Mostrar[Mostrar resultados]         Mostrar --&gt; End(( ))     end     </pre> |   |
| Alertas de usuario   |   |
| En la ejecución de este proceso no se presentaron alertas de usuario   |   |

## 11. Dominio juegos

En el dominio de juegos, el juego elegido es Plantas vs Zombis, un juego clásico de estrategia en donde el objetivo es proteger el jardín de hordas de zombis. Para el patrón de flujo de secuencia el proceso elegido es *ganar chocolate* en el cual el jugador luego de ver un anuncio podrá ganar un chocolate como recompensa el cual dentro del juego es usado como alimento para el caracol Stinky, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 23.

Tabla anexo 23. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Plantas vs Zombies

| Datos del proceso  |   |
|--|---|
| Patrón buscado   | Secuencia   |
| Aplicación   | Plantas vs Zombies  |
| Proceso  | Ganar chocolate   |
| Plataforma   | Aplicación móvil Android  |
| Datos de abstracción del proceso                             |   |
| ¿Qué inicia el proceso?                                      | El proceso se inicia cuando se ingresa a la opción 'Agregar chocolate' en un Jardín Zen                         |
| ¿Qué representa cada instancia o ejecución del proceso?      | Un chocolate añadido para Stinky  |
| ¿Qué determina que el proceso está completo?                 | Cuando el video anuncio es visto en su totalidad y se agrega el chocolate                                       |
| ¿Cuáles son las formas en las que el proceso puede terminar? | El chocolate se ha añadido al inventario del jugador  |
| Mapeo de actividades de alto nivel                           | Ingresar al jardín Zen<br>Seleccionar 'Agregar chocolate'<br>Mirar el anuncio completo<br>Obtener la recompensa |
| Diagrama de proceso de alto nivel                            |   |
|  |   |
| Alerta de usuario #1   |   |
| Alerta de usuario  |   |
| Título   | Chocolate   |
| Contenido  | ¿Quieres ver un video para ganar CHOCOLATE?   |
| Tipo   | Feedforward   |
| Color  | Ninguno   |
| Botones  | Si, No  |
| Representación visual  | Caja de texto   |
| Ubicación en el proceso                                      | Transición entre actividad 'Seleccionar Agregar chocolate' y actividad 'Cargar video anuncio'                   |
| Usuario objetivo   | Jugador   |

Con respecto al patrón de flujo de división/unión paralela, el proceso elegido es *navegar entre jardines zen*, el cual permite al usuario visitar los distintos tipos de jardines (setas y acuífero)

mientras el caracol Stinky recoge las monedas generadas en el jardín principal, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 24.

Tabla anexo 24. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión paralela - Plantas vs Zombies

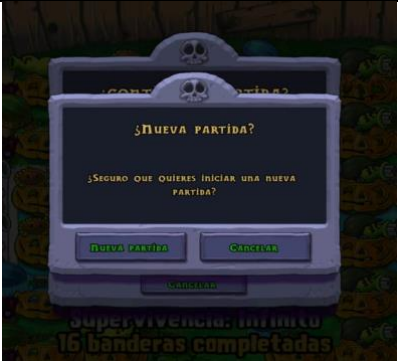
| Datos del proceso   |   |
|---|---|
| <b>Patrón buscado</b>   | División / unión paralela   |
| <b>Aplicación</b>   | Plantas vs Zombies  |
| <b>Proceso</b>  | Navegar entre jardines Zen  |
| <b>Plataforma</b>   | Aplicación móvil Android  |
| Datos de abstracción del proceso                                    |   |
| <b>¿Qué inicia el proceso?</b>                                      | El proceso se inicia cuando se ingresa a la opción 'Siguiete Jardín' en un Jardín Zen   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | Una vuelta completa entre los Jardines Zen del jugador                                  |
| <b>¿Qué determina que el proceso está completo?</b>                 | Cuando el jugador regresa al Jardín principal   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | El jugador regresa al Jardín principal  |
| <b>Mapeo de actividades de alto nivel</b>                           | Ingresar al jardín Zen<br>Seleccionar 'Siguiete Jardín'<br>Regresar al Jardín principal |
| Diagrama de proceso de alto nivel                                   |   |
|   |   |
| Alerta de usuario #1  |   |
| <b>Alerta de usuario</b>  |   |
| <b>Título</b>   | -   |
| <b>Contenido</b>  | ¡Mientras no estabas, Stinky recogió {cantidad} monedas!                                |
| <b>Tipo</b>   | Confirmación  |
| <b>Color</b>  | -   |
| <b>Botones</b>  | -   |
| <b>Representación visual</b>  | Caja de texto   |
| <b>Ubicación en el proceso</b>                                      | Fin del proceso   |
| <b>Usuario objetivo</b>   | Jugador   |

Para el patrón de flujo división/unión exclusiva, el patrón elegido es *iniciar una partida*, en el cual el jugador iniciará una nueva partida o continuará una partida anterior, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 25.

Tabla anexo 25. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - Plantas vs Zombis

| Datos del proceso  |  |
|--|--|
| <b>Patrón buscado</b>  | División / unión exclusiva   |
| <b>Aplicación</b>  | Plantas vs Zombis  |
| <b>Proceso</b>   | Iniciar una partida  |
| <b>Plataforma</b>  | Aplicación móvil Android   |
| Datos de abstracción del proceso   |  |
| <b>¿Qué inicia el proceso?</b>   | El proceso se inicia cuando se ingresa a modo de juego supervivencia   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | El inicio de una partida de supervivencia infinita   |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando se finaliza la partida de supervivencia infinita  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | El jugador termina la partida de supervivencia infinita  |
| <b>Mapeo de actividades de alto nivel</b>  | Ingresar a modo de juego 'Supervivencia'<br>Ingresar a 'Supervivencia infinita'<br>Si ya existe un juego previo seleccionar si continuar o generar nueva partida<br>Cargar juego anterior si se selecciona continuar<br>Generar nuevo juego si se selecciona nueva partida |
| Diagrama de proceso de alto nivel  |  |
| <pre> graph TD     subgraph Usuario         Start(( )) --&gt; IngresarSupervivencia[Ingresar a modo de juego 'Supervivencia']         IngresarSupervivencia --&gt; IngresarSupervivenciaInfinita[Ingresar a 'Supervivencia infinita']     end     subgraph Plantas_vs_Zombis         IngresarSupervivenciaInfinita --&gt; Dec1{X}         Dec1 -- No --&gt; CrearNuevaPartida[Crear nueva partida]         Dec1 -- Si --&gt; ContinuarPartidaAnterior[Continuar partida anterior]         ContinuarPartidaAnterior --&gt; CargarPartida[Cargar partida]         CrearNuevaPartida --&gt; Dec2{X}         CargarPartida --&gt; Dec2         Dec2 --&gt; IniciarJuego[Iniciar juego]         IniciarJuego --&gt; End(( ))     end     </pre> |  |
| Alerta de usuario #1   |  |
| <b>Alerta de usuario</b>   |  |
| <b>Título</b>  | ¿Continuar partida?  |
| <b>Contenido</b>   | ¿Deseas continuar tu partida actual o recomenzar una nueva?  |



|                                |  |
|--------------------------------|--|
| <b>Tipo</b>                    | Advertencia  |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | Continuar, Nueva partida, Cancelar   |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Compuerta de división exclusiva 'Continuar partida anterior'                         |
| <b>Usuario objetivo</b>        | Jugador  |
| <b>Alerta de usuario #2</b>    |  |
| <b>Alerta de usuario</b>       |    |
| <b>Título</b>                  | ¿Nueva partida?  |
| <b>Contenido</b>               | ¿Seguro que quieres iniciar una nueva partida?                                       |
| <b>Tipo</b>                    | Advertencia  |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | Nueva partida, Cancelar  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Transición entre compuerta 'Continuar partida anterior' y actividad 'Cargar partido' |
| <b>Usuario objetivo</b>        | Jugador  |

## 12. Dominio movilidad

En cuanto al dominio movilidad, la aplicación elegida es In Drive, la cual es una plataforma que conecta conductores con pasajeros permitiendo la negociación del precio del servicio. En el caso del patrón de flujo de secuencia, el proceso elegido es *buscar conductor* en el cual un pasajero realiza la petición de un servicio para ir de un lugar a otro por un precio que considere conveniente, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 26.

Tabla anexo 26. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - In Drive

| <b>Datos del proceso</b>                                       |   |
|--|---|
| <b>Patrón buscado</b>  | Secuencia                                       |
| <b>Aplicación</b>  | In Drive  |
| <b>Proceso</b>   | Buscar conductor                                |
| <b>Plataforma</b>  | Aplicación móvil Android                        |
| <b>Datos de abstracción del proceso</b>                        |   |
| <b>¿Qué inicia el proceso?</b>                                 | Cuando el usuario ingresa a la aplicación       |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b> | Una búsqueda de conductor por parte del usuario |
| <b>¿Qué determina que el proceso está completo?</b>            | Cuando se ha enviado la solicitud               |

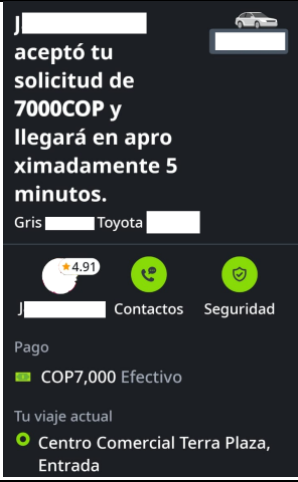
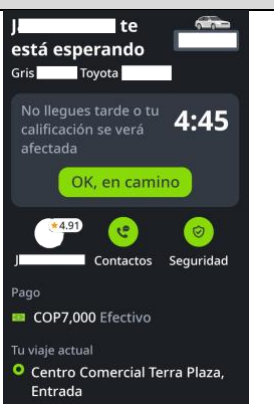
|  |   |
|--|---|
| ¿Cuáles son las formas en las que el proceso puede terminar? | La solicitud se envía a la plataforma   |
| <b>Mapeo de actividades de alto nivel</b>                    | Ingresar a la aplicación<br>Ingresar punto de recogida<br>Ingresar punto de destino<br>Ingresar valor ofertado<br>Solicitar conductor |
| <b>Diagrama de proceso de alto nivel</b>                     |   |
|  |   |
| <b>Alerta de usuario #1</b>                                  |   |
| <b>Alerta de usuario</b>                                     |   |
| <b>Título</b>  | -   |
| <b>Contenido</b>   | Precio recomendado {precio inferior}<br>Tiempo de viaje {tiempo aproximado}   |
| <b>Tipo</b>  | Feedforward   |
| <b>Color</b>   | Azul  |
| <b>Botones</b>   | -   |
| <b>Representación visual</b>                                 | Caja de texto   |
| <b>Ubicación en el proceso</b>                               | Transición entre actividad 'Ingresar punto de destino' e 'Ingresar valor ofertado'  |
| <b>Usuario objetivo</b>                                      | Pasajero  |
| <b>Alerta de usuario #2</b>                                  |   |
| <b>Alerta de usuario</b>                                     |   |
| <b>Título</b>  | Ofreciendo tu precio  |
| <b>Contenido</b>   | -   |
| <b>Tipo</b>  | Confirmación  |
| <b>Color</b>   | -   |
| <b>Botones</b>   | Aumentar / disminuir precio   |
| <b>Representación visual</b>                                 | Palabra simple  |



|                                |  |
|--------------------------------|--|
| <b>Ubicación en el proceso</b> | Actividad 'Enviar solicitud a conductores' |
| <b>Usuario objetivo</b>        | Pasajero                                   |

En cuanto al patrón de flujo de división/unión paralela, el proceso elegido es *realizar viaje* en el cual, un pasajero realiza un viaje luego de buscar exitosamente un conductor, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 27.

Tabla anexo 27. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión paralela - In Drive

| <b>Datos del proceso</b>   |   |
|--|---|
| <b>Patrón buscado</b>  | División / unión paralela   |
| <b>Aplicación</b>  | In Drive  |
| <b>Proceso</b>   | Realizar viaje  |
| <b>Plataforma</b>  | Aplicación móvil Android  |
| <b>Datos de abstracción del proceso</b>  |   |
| <b>¿Qué inicia el proceso?</b>   | Cuando la aplicación asigna un conductor para un viaje  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Un viaje completado por parte del usuario   |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando el pasajero ha llegado al lugar de destino   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | El pasajero llega al lugar de destino   |
| <b>Mapeo de actividades de alto nivel</b>  | Solicitar un conductor<br>Asignar conductor<br>Recoger pasajero<br>Realizar seguimiento de recogida<br>Iniciar viaje<br>Confirmar llegada |
| <b>Diagrama de proceso de alto nivel</b>   |   |
| <pre> sequenceDiagram     participant Pasajero     participant In Drive     participant Conductor      Pasajero-&gt;&gt;In Drive: Solicitar conductor     In Drive-&gt;&gt;Conductor: Asignar conductor     Conductor-&gt;&gt;Conductor: Recoger pasajero     Conductor-&gt;&gt;In Drive: Iniciar viaje     In Drive-&gt;&gt;Conductor: Confirmar llegada     Conductor-&gt;&gt;End:      </pre> |   |
| <b>Alerta de usuario #1</b>  |   |

|                                |  |  |
|--------------------------------|--|--|
| Alerta de usuario              |                |  |
| <b>Título</b>                  | -  |  |
| <b>Contenido</b>               | {Nombre conductor} aceptó tu solicitud de {valor} y llegará en aproximadamente {tiempo estimado} |  |
| <b>Tipo</b>                    | Confirmación   |  |
| <b>Color</b>                   | -  |  |
| <b>Botones</b>                 | -  |  |
| <b>Representación visual</b>   | Caja de texto  |  |
| <b>Ubicación en el proceso</b> | Compuerta de división paralela   |  |
| <b>Usuario objetivo</b>        | Pasajero   |  |
| <b>Alerta de usuario #2</b>    |  |  |
| Alerta de usuario              |              |  |
| <b>Título</b>                  | -  |  |
| <b>Contenido</b>               | {Nombre conductor} te está esperando   |  |
| <b>Tipo</b>                    | Notificación   |  |
| <b>Color</b>                   | -  |  |
| <b>Botones</b>                 | -  |  |
| <b>Representación visual</b>   | Caja de texto  |  |
| <b>Ubicación en el proceso</b> | Transición entre actividad 'Recoger pasajero' e 'Iniciar viaje'                                  |  |
| <b>Usuario objetivo</b>        | Pasajero   |  |
| <b>Alerta de usuario #3</b>    |  |  |

|                                |   |
|--------------------------------|---|
| <b>Alerta de usuario</b>       |   |
| <b>Título</b>                  | -   |
| <b>Contenido</b>               | No llegues tarde o tu calificación se verá afectada                                 |
| <b>Tipo</b>                    | Advertencia   |
| <b>Color</b>                   | -   |
| <b>Botones</b>                 | Ok en camino  |
| <b>Representación visual</b>   | Caja de texto   |
| <b>Ubicación en el proceso</b> | Transición entre actividad 'Recoger pasajero' e 'Iniciar viaje'                     |
| <b>Usuario objetivo</b>        | Pasajero  |
| <b>Alerta de usuario #4</b>    |   |
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | -   |
| <b>Contenido</b>               | Ponte el cinturón y listo   |
| <b>Tipo</b>                    | Confirmación  |
| <b>Color</b>                   | -   |
| <b>Botones</b>                 | -   |
| <b>Representación visual</b>   | Palabra simple  |
| <b>Ubicación en el proceso</b> | Actividad 'Iniciar viaje'   |
| <b>Usuario objetivo</b>        | Pasajero  |

Para el patrón de flujo división/unión exclusiva no se encontraron procesos asociados al dominio dentro de In Drive.

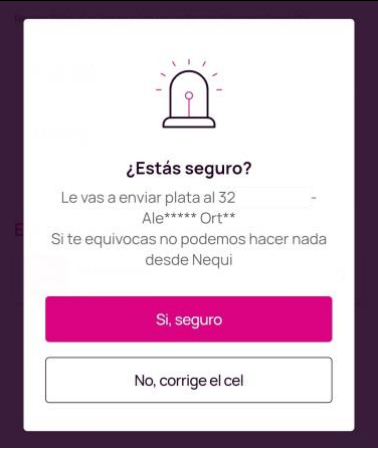

### 13. Dominios bancos

Para el dominio de bancos, la aplicación elegida es Nequi, una plataforma que permite transferencias de manera ágil y que es ampliamente utilizada para transferencias de bajo monto. Para el patrón de secuencia el proceso elegido es *enviar plata* en el cual un usuario puede transferir dinero de su cuenta de Nequi a la cuenta de Nequi de otro usuario, los

resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 28.

Tabla anexo 28. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Nequi

| Datos del proceso   |  |
|---|--|
| Patrón buscado  | Secuencia  |
| Aplicación  | Nequi  |
| Proceso   | Realizar envío   |
| Plataforma  | Aplicación móvil Android   |
| Datos de abstracción del proceso  |  |
| ¿Qué inicia el proceso?   | Se inicia cuando un usuario ingresa a la opción 'Envía'                                  |
| ¿Qué representa cada instancia o ejecución del proceso?   | Un monto de dinero enviado desde una persona hacia otra                                  |
| ¿Qué determina que el proceso está completo?  | Que el dinero esté disponible en la cuenta Nequi de destino                              |
| ¿Cuáles son las formas en las que el proceso puede terminar?  | El dinero se debita de la cuenta de origen y se envía a la cuenta destino                |
| Mapeo de actividades de alto nivel  | Ingresar a la opción 'Envía'<br>Ingresar los datos de la cuenta destino<br>Enviar dinero |
| Diagrama de proceso de alto nivel   |  |
| <pre> graph LR     Start(( )) --&gt; A[Ingresar a opción 'Envía']     A --&gt; B[Ingresar datos de envío]     B --&gt; C[Seleccionar enviar]     C --&gt; D[Realizar transferencia]     D --&gt; End(( ))     subgraph Actor         A         B         C     end     subgraph Actor2         D     end </pre> |  |
| Alerta de usuario #1  |  |
| Alerta de usuario   |  |
| Título  | -  |
| Contenido   | Revisa bien el número para enviarle a la persona correcta                                |
| Tipo  | Advertencia  |
| Color   | -  |
| Botones   | -  |
| Representación visual   | Palabra simple   |

|                                |  |
|--------------------------------|--|
| <b>Ubicación en el proceso</b> | Actividad 'Ingresar datos de envío'  |
| <b>Usuario objetivo</b>        | Usuario origen   |
| <b>Alerta de usuario #2</b>    |  |
| <b>Alerta de usuario</b>       |                              |
| <b>Título</b>                  | ¿Estás seguro?   |
| <b>Contenido</b>               | Le vas a enviar plata al {número destino} - {Nombre destino} Si te equivocas no podemos hacer nada desde Nequi |
| <b>Tipo</b>                    | Advertencia  |
| <b>Color</b>                   | Ninguno  |
| <b>Botones</b>                 | Si, seguro (continuar). No, corrige el cel (rectificar datos)  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Transición entre actividad 'Seleccionar enviar' y Realizar transferencia                                       |
| <b>Usuario objetivo</b>        | Usuario origen   |
| <b>Alerta de usuario #3</b>    |  |
| <b>Alerta de usuario</b>       |                            |
| <b>Título</b>                  | -  |
| <b>Contenido</b>               | Envío realizado  |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | Rojo   |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Palabra simple   |
| <b>Ubicación en el proceso</b> | Fin del proceso  |
| <b>Usuario objetivo</b>        | Usuario origen   |

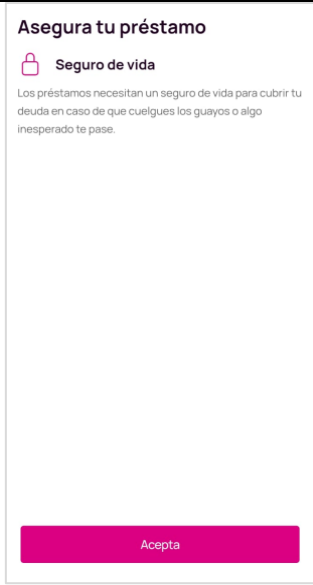
Con respecto al patrón de flujo de división/unión paralela, no se encontraron procesos dentro del dominio en Nequi, por otra parte, para el patrón de flujo de división/unión exclusiva el

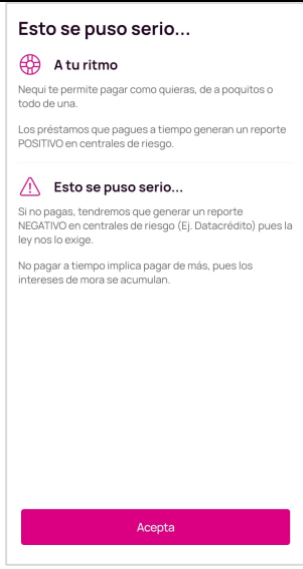
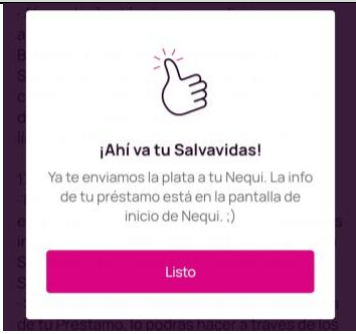
proceso elegido es *solicitar un préstamo salvavidas*, en la cual un usuario puede solicitar un préstamo de emergencia que será agregado a su saldo actual de Nequi, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 29.

Tabla anexo 29. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - Nequi

| Datos del proceso  |   |
|--|---|
| <b>Patrón buscado</b>  | División / unión exclusiva  |
| <b>Aplicación</b>  | Nequi   |
| <b>Proceso</b>   | Solicitar préstamo salvavidas   |
| <b>Plataforma</b>  | Aplicación móvil Android  |
| Datos de abstracción del proceso   |   |
| <b>¿Qué inicia el proceso?</b>   | Cuando se ingresa a la opción 'Préstamo salvavidas'   |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | Representa un préstamo desembolsado al usuario en su cuenta Nequi   |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando el valor se ha agregado a la cuenta del usuario  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | El préstamo se aprueba y se agrega el valor a la cuenta del usuario   |
| <b>Mapeo de actividades de alto nivel</b>  | Ingresar a la opción 'Préstamo salvavidas'<br>Diligenciar datos<br>Agregar póliza<br>Realizar solicitud<br>Agregar valor a cuenta |
| Diagrama de proceso de alto nivel  |   |
| <pre> graph TD     subgraph Usuario         Start(( )) --&gt; Ingresar[Ingresar a 'Préstamo salvavidas']         Ingresar --&gt; Diligenciar[Diligenciar datos]         Diligenciar --&gt; Seleccionar[Seleccionar tipo de póliza]         Seleccionar --&gt; Div1{X}     end     subgraph Nequi         Div1 --&gt; Crear[Crear póliza Nequi]         Crear --&gt; Debitar[Debitar monto en cuenta de usuario]         Debitar --&gt; End(( ))         Div1 --&gt; Validar[Validar póliza externa]         Validar --&gt; Div2{X}         Div2 --&gt; Realizar[Realizar solicitud]         Realizar --&gt; Debitar     end     </pre> |   |
| Alerta de usuario #1   |   |



|                                |   |
|--------------------------------|---|
| <b>Alerta de usuario</b>       |   |
| <b>Título</b>                  | Asegura tu préstamo   |
| <b>Contenido</b>               | Seguro de vida<br>Los préstamos necesitan un seguro de vida para cubrir tu deuda en caso de que cuelgues los guayos o algo inesperado te pase                                   |
| <b>Tipo</b>                    | Advertencia   |
| <b>Color</b>                   | -   |
| <b>Botones</b>                 | Continuar   |
| <b>Representación visual</b>   | Caja de texto   |
| <b>Ubicación en el proceso</b> | Transición entre actividad 'Diligenciar datos' y 'Seleccionar tipo de póliza'   |
| <b>Usuario objetivo</b>        | Usuario   |
| <b>Alerta de usuario #2</b>    |   |
| <b>Alerta de usuario</b>       |   |
| <b>Título</b>                  | Asegura tu préstamo   |
| <b>Contenido</b>               | Un mensaje por opción<br>Póliza Nequi: Con este seguro podrás tener la plata de tu préstamo de una<br>Otro: Instrucciones para presentar otro seguro para respaldar tu préstamo |
| <b>Tipo</b>                    | Feedforward   |
| <b>Color</b>                   | -   |
| <b>Botones</b>                 | -   |
| <b>Representación visual</b>   | Palabra simple  |
| <b>Ubicación en el proceso</b> | Actividad 'Seleccionar tipo de póliza' la cual controla la compuerta de decisión exclusiva tipo de póliza   |
| <b>Usuario objetivo</b>        | Usuario   |
| <b>Alerta de usuario #3</b>    |   |

|                                |   |
|--------------------------------|---|
| <b>Alerta de usuario</b>       |   |
| <b>Título</b>                  | Esto se puso serio...   |
| <b>Contenido</b>               | <p>A tu ritmo: Nequi te permite pagar como quieras, de a poquitos o todo de una. Los préstamos que pagues a tiempo generan un reporte positivo en centrales de riesgo.</p> <p>Esto se puso serio...: Si no pagas, tendremos que generar un reporte NEGATIVO en centrales de riesgo (Ej. Data crédito) pues la ley nos lo exige. No pagar a tiempo implica pagar de más, pues los intereses de mora se acumulan.</p> |
| <b>Tipo</b>                    | Advertencia   |
| <b>Color</b>                   | -   |
| <b>Botones</b>                 | -   |
| <b>Representación visual</b>   | Caja de texto   |
| <b>Ubicación en el proceso</b> | Transición entre actividad 'Realizar solicitud' y 'Debitar monto en cuenta de usuario'  |
| <b>Usuario objetivo</b>        | Usuario   |
| <b>Alerta de usuario #4</b>    |   |
| <b>Alerta de usuario</b>       |   |
| <b>Título</b>                  | ¡Ahí va tu Salvavidas!  |
| <b>Contenido</b>               | Ya te enviamos la plata a tu Nequi. La info de tu préstamo está en la pantalla de inicio de Nequi.  |
| <b>Tipo</b>                    | Confirmación  |
| <b>Color</b>                   | -   |
| <b>Botones</b>                 | Listo   |
| <b>Representación visual</b>   | Caja de texto   |
| <b>Ubicación en el proceso</b> | Fin del proceso   |
| <b>Usuario objetivo</b>        | Usuario   |

## 14. Dominio deportes

En cuanto al dominio de deportes, la aplicación elegida es One Football, plataforma que permite acceder a noticias, estadísticas y actualizaciones de las ligas de fútbol principales a nivel mundial. Para el patrón de flujo de secuencia el proceso elegido es *seguir jugador* en el cual un usuario se podrá suscribir a las noticias, estadísticas y actualizaciones de un jugador de su preferencia, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 30.

Tabla anexo 30. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - One Football

| Datos del proceso  |  |
|--|--|
| Patrón buscado   | Secuencia  |
| Aplicación   | One Football   |
| Proceso  | Seguir jugador   |
| Plataforma   | Aplicación web   |
| Datos de abstracción del proceso   |  |
| ¿Qué inicia el proceso?  | Cuando se ingresa a la opción 'Mis jugadores'  |
| ¿Qué representa cada instancia o ejecución del proceso?  | Representa un jugador que se agrega a la lista de jugadores seguidos por el usuario                        |
| ¿Qué determina que el proceso está completo?   | Cuando el jugador se ha agregado correctamente   |
| ¿Cuáles son las formas en las que el proceso puede terminar?   | El jugador se agrega a la lista del usuario  |
| Mapeo de actividades de alto nivel   | Ingresar a la opción 'Mis jugadores'<br>Seleccionar 'Agregar jugador'<br>Buscar jugador<br>Agregar jugador |
| Diagrama de proceso de alto nivel  |  |
| <pre> sequenceDiagram     participant Usuario     participant OneFootball as One Football     Usuario-&gt;&gt;OneFootball: Ingresar a 'Mis jugadores'     Note over OneFootball: Seleccionar 'Agregar jugador'     OneFootball-&gt;&gt;OneFootball: Buscar jugador     OneFootball-&gt;&gt;OneFootball: Agregar jugador     OneFootball-&gt;&gt;OneFootball: Agregar jugador a la lista     </pre> |  |
| Alerta de usuario #1   |  |
| Alerta de usuario  |  |
| Título   | -  |
| Contenido  | Ahora sigues a {Nombre del jugador}  |
| Tipo   | Confirmación   |
| Color  | -  |
| Botones  | Ninguno  |

|                                |                 |
|--------------------------------|-----------------|
| <b>Representación visual</b>   | Caja de texto   |
| <b>Ubicación en el proceso</b> | Fin del proceso |
| <b>Usuario objetivo</b>        | Usuario         |

Dentro de One Football, no se encontraron procesos que incluyan los patrones de flujo de división/unión exclusiva y paralela.

## 15. Dominio viajes

Entre las aplicaciones en el dominio de viajes, la elegida es Trip Advisor, plataforma que facilita la planificación de viajes en cuanto a lugares turísticos, transporte y hospedaje. Para el patrón de flujo de secuencia el proceso elegido es *buscar hotel en ubicación* que le permite a un usuario observar la lista de hoteles disponibles en una ubicación específica para un tiempo de estadía definido, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 31.

Tabla anexo 31. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Trip Advisor

| Datos del proceso  |  |
|--|--|
| <b>Patrón buscado</b>  | Secuencia  |
| <b>Aplicación</b>  | Trip Advisor   |
| <b>Proceso</b>   | Buscar hotel en ubicación  |
| <b>Plataforma</b>  | Aplicación web   |
| Datos de abstracción del proceso   |  |
| <b>¿Qué inicia el proceso?</b>   | Cuando el usuario ingresa a la opción 'Hoteles'  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>   | La búsqueda de un hotel en una ubicación determinada   |
| <b>¿Qué determina que el proceso está completo?</b>  | Cuando los resultados se muestran al usuario   |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>  | Se muestran resultados al usuario  |
| <b>Mapeo de actividades de alto nivel</b>  | El usuario ingresa a la opción 'Hoteles'<br>El usuario ingresa una ubicación<br>El usuario ingresa datos del hospedaje<br>Se muestran los resultados |
| Diagrama de proceso de alto nivel  |  |
| <pre> graph LR     subgraph "Trip Advisor - Buscar hotel en ubicación"         direction TB         subgraph "Usuario"             Start(( )) --&gt; A[Ingresar a opción 'Hoteles']             A --&gt; B[Ingresar ubicación]             B --&gt; C[Ingresar datos del hospedaje]         end         subgraph "Trip Advisor"             D[Mostrar resultados en ubicación]         end         C --&gt; D         D --&gt; End(( ))     end </pre> |  |
| Alerta de usuario #1   |  |

|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | -  |
| <b>Contenido</b>               | {cantidad} establecimientos en {ubicación}   |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Palabra simple   |
| <b>Ubicación en el proceso</b> | Final del proceso  |
| <b>Usuario objetivo</b>        | Usuario  |

Los patrones de flujo de división/unión paralela y exclusiva, no se encontraron dentro de los procesos asociados al dominio en Trip Advisor.

## 16. Dominio de salud y bienestar

La aplicación elegida para el dominio de salud y bienestar es Mi Fitness, una aplicación para el registro y seguimiento de parámetros de salud y actividad física. Para el patrón de flujo de secuencia el proceso elegido es *invitar a un amigo a competición*, en la cual un usuario puede enviar una invitación de competición a un amigo para compartir y competir con datos relacionados a su actividad física, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 32.

Tabla anexo 32. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Mi Fitness

| <b>Datos del proceso</b>  |   |
|---|---|
| <b>Patrón buscado</b>   | Secuencia   |
| <b>Aplicación</b>   | Mi Fitness  |
| <b>Proceso</b>  | Invitar un amigo a competición  |
| <b>Plataforma</b>   | Aplicación móvil Android  |
| <b>Datos de abstracción del proceso</b>                             |   |
| <b>¿Qué inicia el proceso?</b>                                      | Cuando el usuario ingresa a la opción 'Competición'                                     |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | Un amigo agregado a la lista con el cual se podrán compartir estadísticas y compararlas |
| <b>¿Qué determina que el proceso está completo?</b>                 | Cuando el invitado se agrega como contacto  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | El invitado se agrega como contacto de competición                                      |

|   |   |
|---|---|
| <b>Mapeo de actividades de alto nivel</b>   | El usuario ingresa a la opción competición<br>Se envía una invitación<br>El invitado acepta la invitación |
| <b>Diagrama de proceso de alto nivel</b>  |   |
| <pre> graph TD     subgraph Usuario         Start(( )) --&gt; Ingresar[Ingresar a 'Competición']         Ingresar --&gt; Enviar[Enviar invitación]     end     subgraph Invitado         Aceptar[Aceptar invitación]     end     subgraph Mi_Fitness[Mi Fitness]         Crear[Crear conexión de competición] --&gt; End(( ))     end     Enviar --&gt; Aceptar     Aceptar --&gt; Crear </pre> |   |
| <b>Alerta de usuario #1</b>   |   |
| <b>Alerta de usuario</b>  |   |
| <b>Título</b>   | Invitación de competición   |
| <b>Contenido</b>  | Después que comience la competición, tus estadísticas y datos de actividad se compartirán con tu amigo.   |
| <b>Tipo</b>   | Advertencia   |
| <b>Color</b>  | -   |
| <b>Botones</b>  | Abrir Mi Fitness  |
| <b>Representación visual</b>  | Caja de texto   |
| <b>Ubicación en el proceso</b>  | Transición entre actividad 'Aceptar invitación' y 'Crear conexión de competición'                         |
| <b>Usuario objetivo</b>   | Invitado  |

El patrón de flujo de división/unión exclusiva no se encontró entre los procesos asociados al dominio en Mi Fitness, por otra parte, para el patrón de flujo de división/unión exclusiva, el proceso elegido es *registrar caminata*, en la que el usuario podrá registrar los datos asociados a actividad física cuando esta supere el límite mínimo para poder ser agregada, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 33.

Tabla anexo 33. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión exclusiva - Mi Fitness

| Datos del proceso     |                            |
|-----------------------|----------------------------|
| <b>Patrón buscado</b> | División / unión exclusiva |

|   |   |
|---|---|
| <b>Aplicación</b>   | Mi Fitness  |
| <b>Proceso</b>  | Registrar caminata  |
| <b>Plataforma</b>   | Aplicación móvil Android  |
| <b>Datos de abstracción del proceso</b>                             |   |
| <b>¿Qué inicia el proceso?</b>                                      | Cuando el usuario ingresa a la opción 'Registrar caminata'  |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>      | Una caminata registrada en la aplicación  |
| <b>¿Qué determina que el proceso está completo?</b>                 | Cuando el usuario detiene el registro de la caminata  |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b> | La caminata se registra<br>La caminata no se registra por corta duración  |
| <b>Mapeo de actividades de alto nivel</b>                           | El usuario ingresa a la opción 'Registrar caminata'<br>El usuario inicia la caminata<br>El usuario detiene la caminata<br>Se guarda la información de la caminata si la duración es suficiente (acorde a configuración del usuario) |
| <b>Diagrama de proceso de alto nivel</b>                            |   |
|   |   |
| <b>Alerta de usuario #1</b>   |   |
| <b>Alerta de usuario</b>  |   |
| <b>Título</b>   | -   |
| <b>Contenido</b>  | El tiempo de entrenamiento es demasiado corto. Esta sesión no será grabada. ¿Finalizar de todos modos?  |
| <b>Tipo</b>   | Advertencia   |
| <b>Color</b>  | -   |
| <b>Botones</b>  | Finalizar, Continuar  |

|                                |  |
|--------------------------------|--|
| <b>Representación visual</b>   | Caja de texto                                      |
| <b>Ubicación en el proceso</b> | Compuerta de división exclusiva 'Validar duración' |
| <b>Usuario objetivo</b>        | Usuario  |

## 17. Dominio educación

En dominio de educación, la aplicación elegida es Duolingo, la cual es líder en plataformas de enseñanza de idiomas enfocada en la gamificación del aprendizaje. Para el proceso de secuencia el proceso elegido es *realizar lección*, en la cual el usuario de acuerdo con su progreso en sus metas de aprendizaje realizará actividades de refuerzo para continuar con su aprendizaje, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 34.

Tabla anexo 34. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Duolingo

| Datos del proceso   |   |
|---|---|
| <b>Patrón buscado</b>   | Secuencia   |
| <b>Aplicación</b>   | Duolingo  |
| <b>Proceso</b>  | Realizar lección  |
| <b>Plataforma</b>   | Aplicación móvil Android  |
| Datos de abstracción del proceso  |   |
| <b>¿Qué inicia el proceso?</b>  | Al seleccionar iniciar lección en uno de los idiomas de estudio del usuario |
| <b>¿Qué representa cada instancia o ejecución del proceso?</b>  | Una lección terminada en el idioma seleccionado                             |
| <b>¿Qué determina que el proceso está completo?</b>   | Cuando la lección se ha terminado correctamente                             |
| <b>¿Cuáles son las formas en las que el proceso puede terminar?</b>   | La lección se ha terminado  |
| <b>Mapeo de actividades de alto nivel</b>   | Ingresar a idioma<br>Seleccionar lección<br>Realizar actividades            |
| Diagrama de proceso de alto nivel   |   |
| <pre> graph LR     subgraph Duolingo         Cargar[ cargar actividades ]     end     subgraph Estudiante         Inicio(( )) --&gt; S1[ Seleccionar idioma ]         S1 --&gt; S2[ Seleccionar 'Lección' ]         S2 --&gt; Cargar         Cargar --&gt; R1[ Realizar actividades ]         R1 --&gt; Fin(( ))     end </pre> |   |
| Alerta de usuario #1  |   |
| <b>Alerta de usuario</b>  |   |
| <b>Título</b>   | ¡Buen trabajo!  |
| <b>Contenido</b>  | -   |



|                                |  |
|--------------------------------|--|
| <b>Tipo</b>                    | Feedback   |
| <b>Color</b>                   | Verde  |
| <b>Botones</b>                 | Continuar, Compartir, Marcar hito  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Transición entre actividades a realizar  |
| <b>Usuario objetivo</b>        | Estudiante   |
| <b>Alerta de usuario #2</b>    |  |
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | ¡Lección perfecta!   |
| <b>Contenido</b>               | ¡Haz una reverencia!   |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | Naranja  |
| <b>Botones</b>                 | Continuar  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Fin del proceso  |
| <b>Usuario objetivo</b>        | Estudiante   |

Dentro de los procesos del dominio en Duolingo, no se encontró ninguno que incluyera los patrones de flujo de división/unión paralela y exclusiva.

## 18. Dominio domicilios

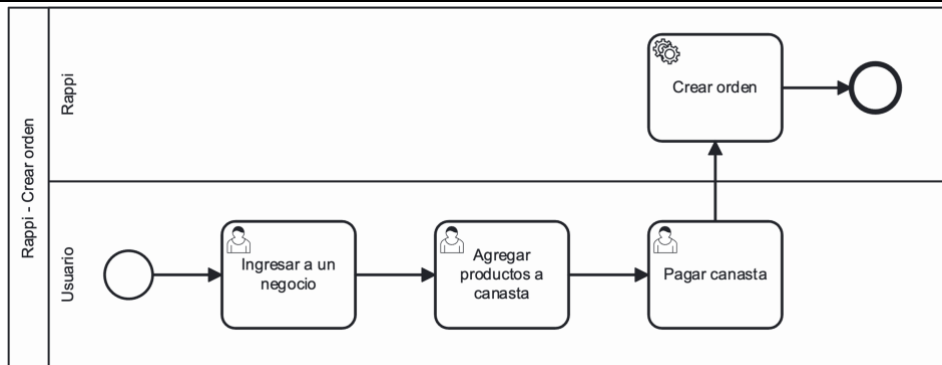
En el dominio de domicilios, la aplicación elegida es Rappi, una plataforma que conecta restaurantes, supermercados, farmacias y negocios en general con domiciliarios que realizan entregas a usuarios, enfocada en la rapidez en el servicio. Para el patrón de flujo de secuencia el proceso elegido es *crear orden* en la cual usuario puede agregar productos de un negocio y pagar su valor total para que estos sean luego entregados en su domicilio, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 35.

Tabla anexo 35. Abstracción de proceso de alertas de usuario, patrón de flujo secuencia - Rappi

| <b>Datos del proceso</b>                |  |
|---|--|
| <b>Patrón buscado</b>                   | Secuencia                              |
| <b>Aplicación</b>                       | Rappi                                  |
| <b>Proceso</b>                          | Crear orden                            |
| <b>Plataforma</b>                       | Aplicación móvil Android               |
| <b>Datos de abstracción del proceso</b> |  |
| <b>¿Qué inicia el proceso?</b>          | Cuando el usuario ingresa a un negocio |

|  |   |
|--|---|
| ¿Qué representa cada instancia o ejecución del proceso?      | Una orden creada en la aplicación   |
| ¿Qué determina que el proceso está completo?                 | Cuando la orden ha sido enviada al negocio                                  |
| ¿Cuáles son las formas en las que el proceso puede terminar? | La orden ha sido creada   |
| Mapeo de actividades de alto nivel                           | Ingresar a un negocio<br>Agregar productos a la canasta<br>Pagar la canasta |

#### Diagrama de proceso de alto nivel



#### Alerta de usuario #1

|                         |  |
|-------------------------|--|
| Alerta de usuario       |  |
| Título                  | -  |
| Contenido               | Realizaremos un cobro a tu método de pago por el valor de la orden. Si hay modificaciones, lo ajustaremos al saber el valor final de tu orden. |
| Tipo                    | Advertencia  |
| Color                   | -  |
| Botones                 | -  |
| Representación visual   | Palabra simple   |
| Ubicación en el proceso | Transición entre actividad 'Pagar canasta' y 'Realizar pedido'   |
| Usuario objetivo        | Usuario  |

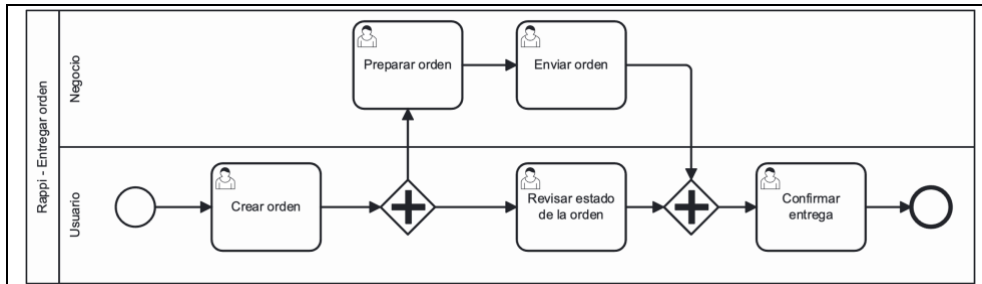
#### Alerta de usuario #2

|                         |                         |
|-------------------------|-------------------------|
| Alerta de usuario       |                         |
| Título                  | -                       |
| Contenido               | Pedido creado con éxito |
| Tipo                    | Confirmación            |
| Color                   | -                       |
| Botones                 | -                       |
| Representación visual   | Caja de texto           |
| Ubicación en el proceso | Fin del proceso         |
| Usuario objetivo        | Usuario                 |

En cuando al patrón de flujo de división/unión paralela, el proceso elegido es *entregar orden*, la cual permite a un negocio preparar y enviar a través de un domiciliario de Rappi la orden creada de un cliente, los resultados de la abstracción del proceso y alertas presentadas se muestra en la Tabla anexo 36.

Tabla anexo 36. Abstracción de proceso de alertas de usuario, patrón de flujo división/unión paralela - Rappi

| Datos del proceso  |   |
|--|---|
| Patrón buscado   | División / unión paralela   |
| Aplicación   | Rappi   |
| Proceso  | Entregar orden  |
| Plataforma   | Aplicación móvil Android  |
| Datos de abstracción del proceso                             |   |
| ¿Qué inicia el proceso?                                      | Cuando el usuario crea una orden  |
| ¿Qué representa cada instancia o ejecución del proceso?      | Una orden entregada al usuario  |
| ¿Qué determina que el proceso está completo?                 | Cuando el usuario ha confirmado la recepción de la orden  |
| ¿Cuáles son las formas en las que el proceso puede terminar? | La orden ha sido entregada al usuario   |
| Mapeo de actividades de alto nivel                           | Crear una orden<br>Preparar la orden<br>Enviar la orden<br>Verificar estado de la orden<br>Confirmar recibido de la orden |
| Diagrama de proceso de alto nivel                            |   |



### Alerta de usuario #1


|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | -  |
| <b>Contenido</b>               | El restaurante está preparando tu pedido                                       |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Transición entre compuerta de división exclusiva y actividad 'Preparar pedido' |
| <b>Usuario objetivo</b>        | Usuario  |

### Alerta de usuario #2

|                                |  |
|--------------------------------|--|
| <b>Alerta de usuario</b>       |  |
| <b>Título</b>                  | Tu Rappi va en camino a tu dirección                           |
| <b>Contenido</b>               | -  |
| <b>Tipo</b>                    | Confirmación   |
| <b>Color</b>                   | -  |
| <b>Botones</b>                 | -  |
| <b>Representación visual</b>   | Caja de texto  |
| <b>Ubicación en el proceso</b> | Transición entre actividad 'Preparar pedido' y 'Enviar pedido' |
| <b>Usuario objetivo</b>        | Usuario  |

### Alerta de usuario #3

|                          |                  |
|--------------------------|------------------|
| <b>Alerta de usuario</b> |                  |
| <b>Título</b>            | ¡Llegó tu Rappi! |

|                                |   |
|--------------------------------|---|
| <b>Contenido</b>               | Tu Rappi llegó a tu dirección de entrega con tu pedido  |
| <b>Tipo</b>                    | Confirmación  |
| <b>Color</b>                   | -   |
| <b>Botones</b>                 | -   |
| <b>Representación visual</b>   | Caja de texto   |
| <b>Ubicación en el proceso</b> | Transición entre actividad 'Enviar orden y 'Confirmar entrega'  |
| <b>Usuario objetivo</b>        | Usuario   |
| <b>Alerta de usuario #4</b>    |   |
| <b>Alerta de usuario</b>       |  <p>The screenshot shows a notification card with the following text: <b>Q71 - Frisby CC</b>, <b>Terraplaza - PPN</b>, <b>Pedido entregado</b>, <b>Califica tu pedido</b>, and a small icon with the text 'Cuéntanos por favor tu experiencia con el Rappi y ...' followed by a right-pointing arrow.</p> |
| <b>Título</b>                  | -   |
| <b>Contenido</b>               | Pedido entregado  |
| <b>Tipo</b>                    | Confirmación  |
| <b>Color</b>                   | Verde   |
| <b>Botones</b>                 | -   |
| <b>Representación visual</b>   | Palabra simple  |
| <b>Ubicación en el proceso</b> | Fin del proceso   |
| <b>Usuario objetivo</b>        | Usuario   |

## Anexo 2. Validaciones sobre requisitos

### 1. Validación general de requisitos de creación de servicios en el API REST

Las siguientes corresponden a las validaciones generales que se deben tener en cuenta en la validación de todos los casos de uso que impliquen agregar un servicio en el API REST

#### 1.2. Disponibilidad del recurso

El servicio debe estar disponible para su consumo mediante el método HTTP y la url especificadas.

#### 1.3. Validación de datos y tipos

Si los datos enviados no son correctos, acorde a la validación en el caso de uso CDA-002 Validador de peticiones, se retornará una respuesta generada con el caso de uso CDA-001 Generador de respuestas con statusCode 422, message 'Datos inválidos' y errors con los errores obtenidos en la validación.

#### 1.4. Búsqueda de recursos

Si los datos enviados son correctos, pero los campos de identificación de recurso enviados no corresponden a registros almacenados en el componente se retornará una respuesta generada con el caso de uso CDA-001 Generador de respuestas con statusCode 404, message 'Recurso no encontrado' y una entrada en errors con el mensaje 'No existe ningún <nombre del recurso> con el <campo de identificación> <valor enviado>'

#### 1.5. Estructura de respuesta

La estructura de la respuesta correcta debe estar acorde a la definición del caso de uso CDA-001 Generador de respuestas, tomando en cuenta la estructura de la respuesta del servicio, tanto en los datos retornados como en el código HTTP de la respuesta.

#### 1.6. Dependencias con otras tareas

Todas las tareas de creación de servicios en el API REST tienen dependencias con las tareas generales de validación y generación de respuestas:

- CDA-001 Generador de respuestas
- CDA-002 Validador de peticiones

### 2. Validación de requisitos de creación de recursos mediante API REST

Las siguientes corresponden a las validaciones que se deben tener en cuenta en la validación de los casos de uso de creación de recursos en el API REST

#### 2.2. Estructura de respuesta

El mensaje agregado en los servicios de creación de recursos será 'Recurso creado'.

### 3. Validación de requisitos de obtener información de recursos mediante API REST

Las siguientes corresponden a las validaciones que se deben tener en cuenta en la validación de los de obtener información de recursos en API REST

#### 3.2. Estructura de respuesta

El mensaje agregado en los servicios de creación de recursos será 'Datos del recurso <nombre del recurso>'.

### 4. Validación general de requisitos de creación de casos de uso

Para validar la implementación de casos de uso se deben tener en cuenta las siguientes validaciones.

#### 4.2. Disponibilidad de caso de uso

Debe existir un caso de uso y accesible que permita su ejecución con los parámetros definidos para el mismo.

#### 4.3. Entradas y salidas del caso de uso

El caso de uso deberá solicitar los datos de entrada especificados en el requisito y contar con las salidas acorde a las condiciones y la estructura de la respuesta requerida en el requisito.

## Anexo 3. Inicialización de proyecto de desarrollo

En este anexo se describe el proceso de inicialización del proyecto con Laravel lumen, el cual es la base del desarrollo del componente.

### 1. Creación del proyecto con Lumen

Esta actividad se enfocó en inicializar un proyecto vacío con Lumen utilizando Composer, con lo cual se configuraron todas dependencias del Framework y se creó la estructura de carpetas y archivos base del proyecto como se muestra en la Figura anexo 1.

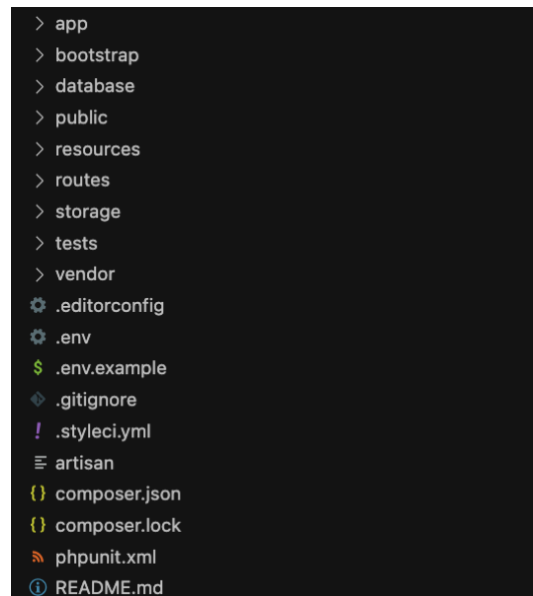


Figura anexo 1. Estructura base del proyecto Lumen

### 2. Creación de contenedores del componente

Una vez creado el proyecto vacío con Lumen, este será encapsulado en un contenedor Docker que contenga la configuración requerida para que el proyecto con Lumen pueda ejecutarse y que pueda ser accedido de manera externa mediante peticiones HTTP. Adicionalmente se creará un contenedor que permita la ejecución de la base de datos, agregando la configuración necesaria para que los dos contenedores puedan comunicarse entre sí.

El contenedor de la aplicación se configura a partir de la imagen estándar php:8.3.0-apache-bullseye, la cual es ofrecida de manera oficial por PHP y que contiene el runtime de PHP 8.3 junto con el servidor apache. Para la creación de este contenedor se agregan algunas configuraciones usando Dockerfile sobre el servidor apache integrado que permitirán realizar peticiones desde el exterior del contenedor, y que estas apunten hacia el API generada por Lumen en el proyecto vacío, además se realiza la instalación de algunos paquetes requeridos para que la aplicación con Lumen funcione correctamente como se muestra en la Figura anexo 2.



```

1  # Prepare the required software
2  FROM php:8.3.0-apache-bullseye
3  COPY --from=composer:latest /usr/bin/composer /usr/local/bin/composer
4  RUN apt-get update && \
5      apt-get install -y \
6          libzip-dev \
7          zip \
8          unzip \
9          nano
10
11 # Files to map ports on server
12 COPY container-files/apache-conf/ports.conf /etc/apache2/ports.conf
13 COPY container-files/apache-conf/user-alerts-component.conf /etc/apache2/sites-available
14
15 # Activate site port
16 RUN a2dissite 000-default.conf
17 RUN a2ensite user-alerts-component.conf
18 RUN a2enmod rewrite
19
20 # Perform database connection
21 RUN docker-php-ext-install mysqli pdo pdo_mysql && docker-php-ext-enable pdo_mysql
22
23 # Enable xdebug
24 RUN pecl install xdebug && docker-php-ext-enable xdebug
25 COPY container-files/xdebug/xdebug.ini /usr/local/etc/php/conf.d
26 EXPOSE 9000
27
28 # Configure container
29 WORKDIR /app/user-alerts-component

```

*Figura anexo 2. Configuración de contenedor de aplicación del componente*

Por otra parte, el contenedor de base de datos se genera a partir de la imagen oficial mysql:8.0, sobre la cual no se realizan modificaciones, la configuración de la interacción entre los dos contenedores, así como la configuración para que estos puedan ser ejecutados se realiza a través de un archivo docker-compose.yaml como se muestra en la Figura anexo 3.

```

1  version: '3'
2
3  services:
4    component-api-container:
5      build:
6        context: .
7        dockerfile: Dockerfile
8      container_name: component-api-container
9      ports:
10     - 9000
11     - 8083:8083
12     volumes:
13     - ./app/user-alerts-component
14     - ./container-files/logs:/app/logs
15     environment:
16     - XDEBUG_SESSION=uac_session
17   component-database-container:
18     image: mysql:8.0
19     container_name: component-database-container
20     ports:
21     - 3309:3306
22     environment:
23     - MYSQL_ROOT_PASSWORD=*SafePas5sWoRd4R0ot*
24     - MYSQL_USER=component_api_user
25     - MYSQL_PASSWORD=*SafePas5sWoRd4us3R*
26     volumes:
27     - ./container-files/mysql-init:/docker-entrypoint-initdb.d

```

Figura anexo 3. Configuración de stack de contenedores del componente

### 3. Configuración de base de datos

Una vez se tiene lista la configuración del componente, se configura el componente a través de los servicios de Lumen para que este pueda comunicarse con la base de datos, para tal fin se agregan en las variables de entorno, la configuración de conexión de la base de datos como se muestra en la Figura anexo 4.

```

5  # Database connections
6  DB_CONNECTION=mysql
7  DB_HOST=component-database-container
8  DB_PORT=3306
9  DB_DATABASE=component_api_database
10 DB_USERNAME=root
11 DB_PASSWORD=*SafePas5sWoRd4R0ot*

```

Figura anexo 4. Configuración de base de datos en el componente

## Anexo 4. Implementación de módulos generales

En este anexo se describe la implementación de módulos generales del desarrollo, los cuales sirven como base para la creación e interacción de los modelos del dominio y pueden ser usados de manera general en la creación de los requisitos del componente.

### 1. Migraciones

Las migraciones se almacenan dentro de `database > migrations`, y cada archivo representa un paso en la construcción del modelo de datos, el cual cuenta con las funciones `up` y `down`. `Up` es utilizada para agregar información al modelo de datos, como puede ser una nueva tabla, índice, llave foránea o inserción de datos; Mientras que `down` permite anular la acción mapeada en `up`, si en algún momento llegase a ser necesario revertir una migración.

La implementación de las migraciones se implementó en el orden que se muestra en la Figura anexo 5, iniciando con la creación de la tabla de datos para sistema objetivo, por ser esta la tabla central del modelo y la cual no almacena ninguna referencia hacia otra tabla, acto seguido se crean las demás tablas agregando las claves foráneas que se plantearon en el modelo de datos de la fase de diseño.

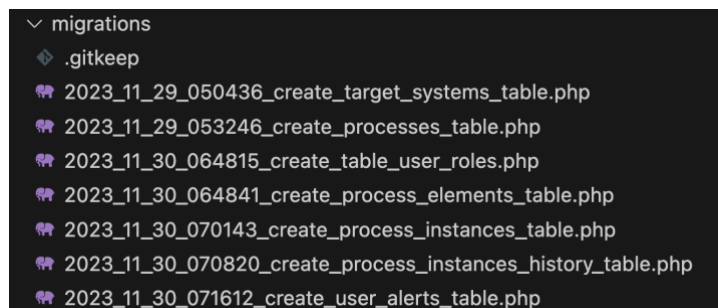


Figura anexo 5. Migraciones implementadas en el componente

### 2. Modelos

Estos se almacenan dentro de `app > Models`. Para la implementación de los modelos en el componente, se creó una interfaz que contiene los métodos genéricos de los modelos, los cuales son implementados por una clase abstracta que implementó los métodos genéricos de la interfaz, finalmente cada modelo concreto extiende de la clase abstracta, implementando la interfaz a través del modelo abstracto como se muestra en la Figura anexo 6, esto con el objetivo de generar otros componentes de manera genérica.

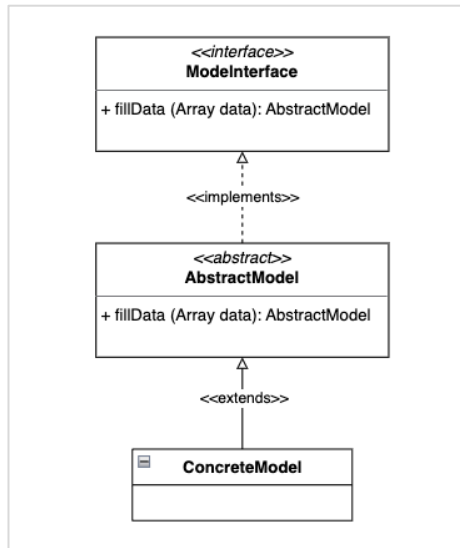


Figura anexo 6. Estructura de modelos del componente

Los componentes generados se implementaron dentro de la estructura del proyecto como se muestra en la Figura anexo 7.

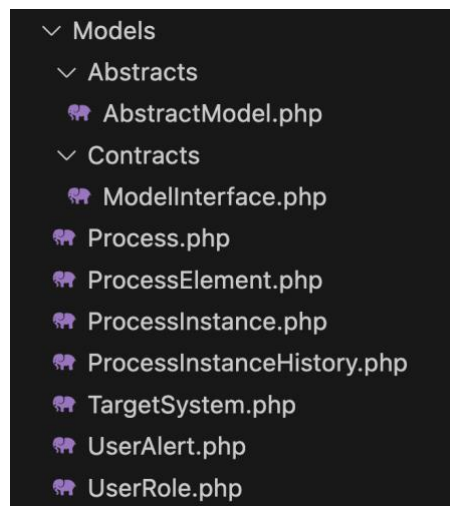


Figura anexo 7. Implementación de modelos en el componente

### 3. Repositorios

Para la implementación de repositorios en el componente se ha creó una interfaz que define los métodos principales para cada modelo, como lo son guardar, buscar por identificador y buscar por parámetros. Esta interfaz fue implementada por una clase abstracta que desarrolló los métodos de forma genérica utilizando los servicios del Framework para obtener o guardar los datos en la base de datos. Cada modelo concreto contó con una interfaz que extiende los métodos de la interfaz principal, mediante la cual se agregaron métodos específicos para el modelo, esta interfaz fue implementada por el repositorio concreto del modelo que a su vez extiende de la clase abstracta. En cada repositorio concreto se desarrollaron los métodos específicos del modelo como se muestra en la Figura anexo 8.

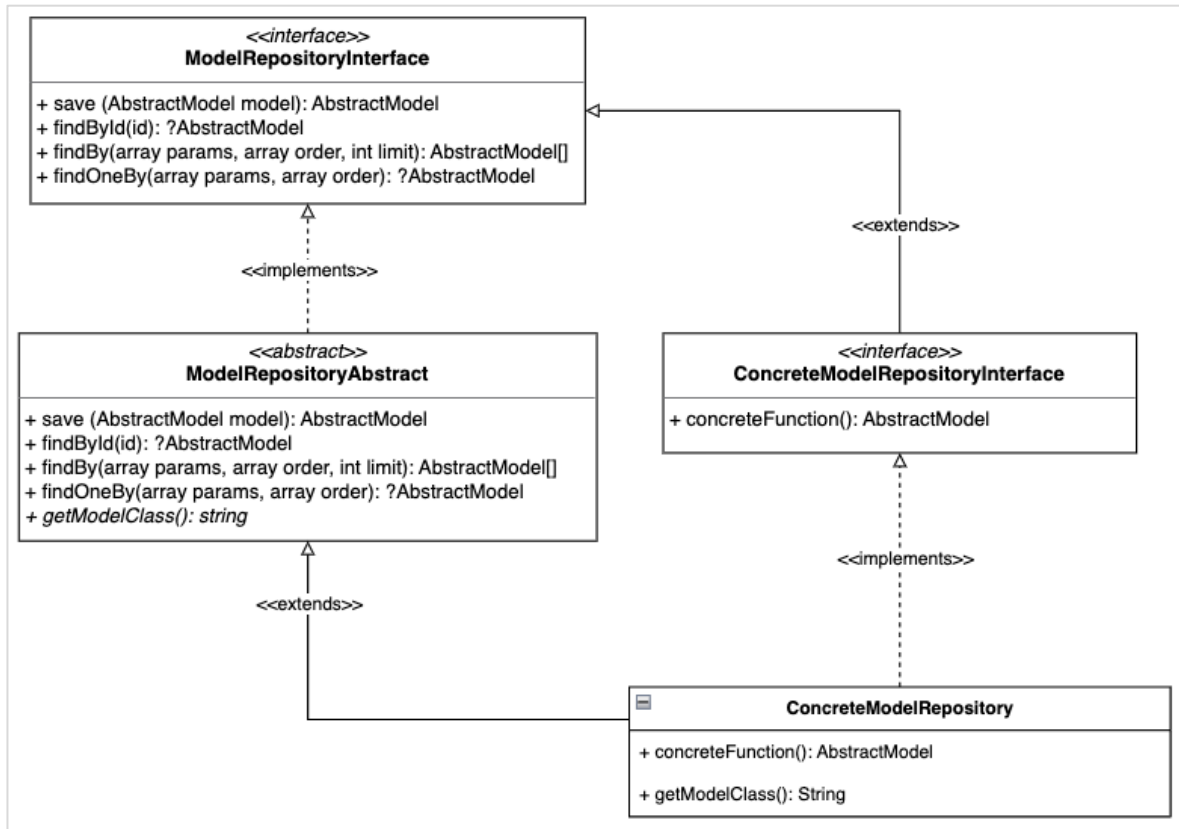


Figura anexo 8. Estructura de repositorios en el componente

La implementación de la estructura de repositorios dentro del componente se agregó dentro de `app > Repositories` como se muestra en la Figura anexo 9.

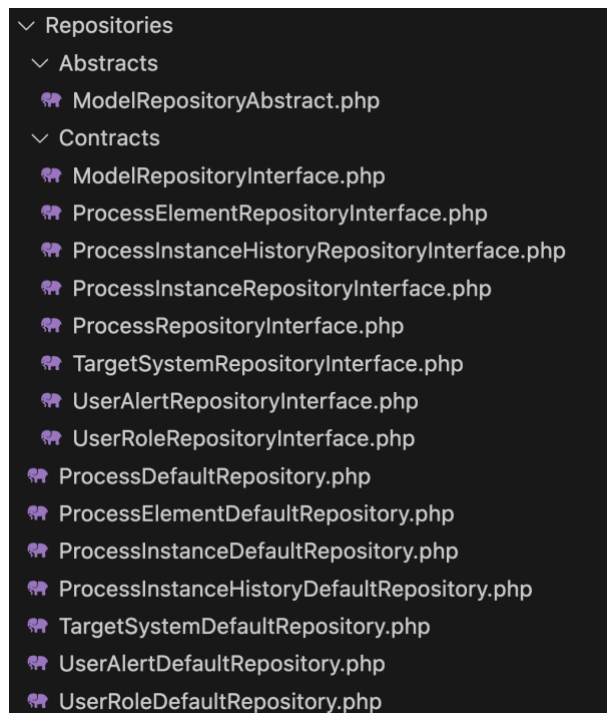


Figura anexo 9. Implementación de estructura de repositorios

## 4. Excepciones

Dentro del componente la implementación de las excepciones se creó con una jerarquía definida, con el objetivo de poder diferenciar las excepciones controladas según el contexto en el que estas se presentan como se muestra en la Figura anexo 10. De manera general se crearon dos tipos de excepciones principales que extienden de las excepciones base del lenguaje; las excepciones creadas fueron *DomainException* y *FatalException*, en donde las excepciones de dominio corresponden a errores cuyos mensajes asociados están asociados a validaciones de negocio que pueden ser mostrados al usuario ya que pueden ser validados o solucionados por quien consume el componente, mientras que las excepciones fatales se definieron como aquellas que corresponden a errores cuya solución no se puede validar de manera externa.

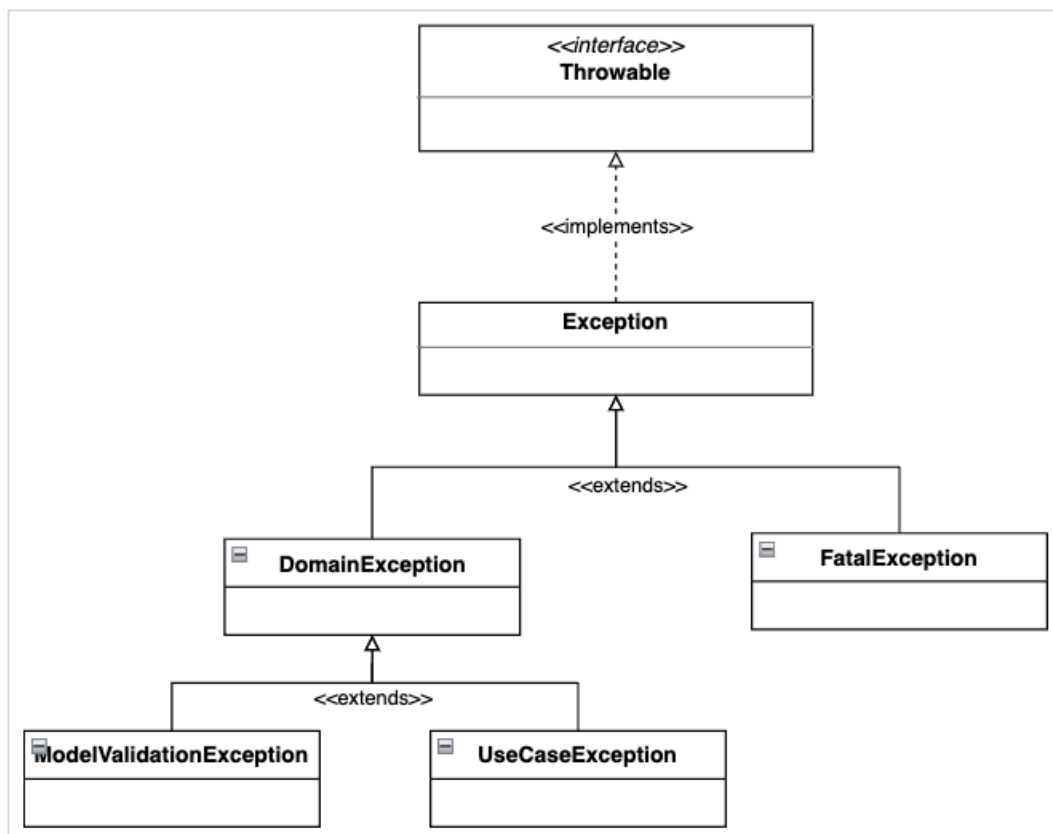


Figura anexo 10. Estructura de excepciones

Dentro del componente la jerarquía de excepciones se implementó dentro de *app > Exceptions* como se muestra en la Figura anexo 11, y el control de estas para mostrar o no los mensajes de acuerdo con la jerarquía, se implementó mediante el *handler* de excepciones provisto por el Framework.

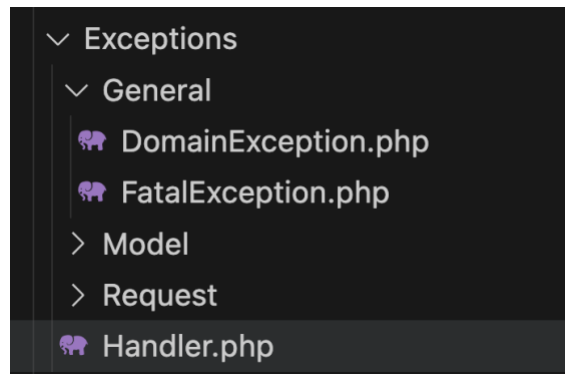


Figura anexo 11. Implementación de excepciones

## 5. Casos de uso

Cada caso de uso representa una función específica dentro del dominio del componente, la cual cuenta con una interfaz que define los parámetros de entrada y una o varias implementaciones según sea el caso. La decisión de la creación de una interfaz para cada caso de uso se hace con el objetivo de facilitar el testing unitario. Los casos de uso se implementan dentro de `app > UseCases` y se organizaron en subcarpetas según el punto del dominio al que apliquen como se muestra en la Figura anexo 12.

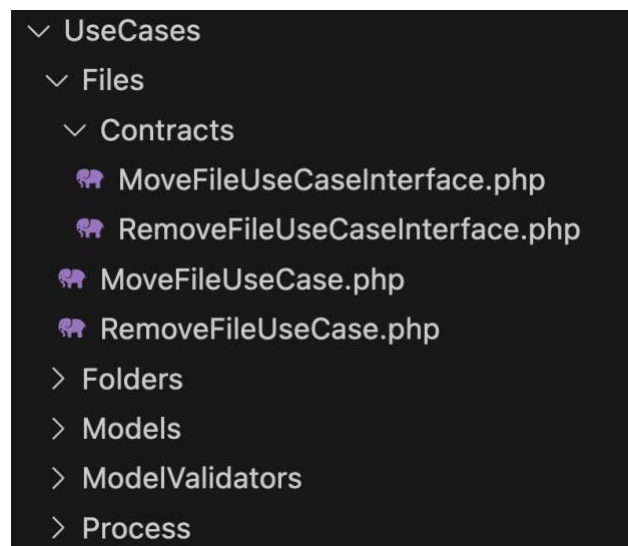


Figura anexo 12. Implementación de casos de uso

## 6. Controladores

Los controladores tienen una estructura que extiende de una clase abstracta que contiene las funciones genéricas que abstraen lógica genérica para cualquier solicitud como se muestra en la Figura anexo 13. Cada controlador concreto tiene una responsabilidad única, es decir que sólo mapeará y controlará un servicio dentro del API, es por eso por lo que los controladores concretos únicamente cuentan con una función pública `__invoke`.

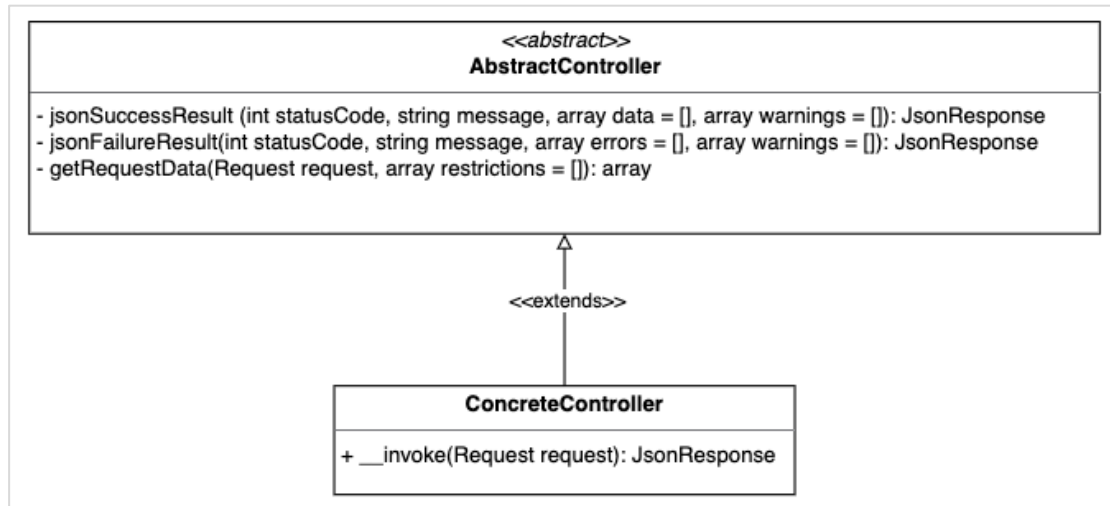


Figura anexo 13. Estructura de controladores

Dentro de la estructura los controladores se ubican en *app > Http > Controllers* como se muestra en la Figura anexo 14.

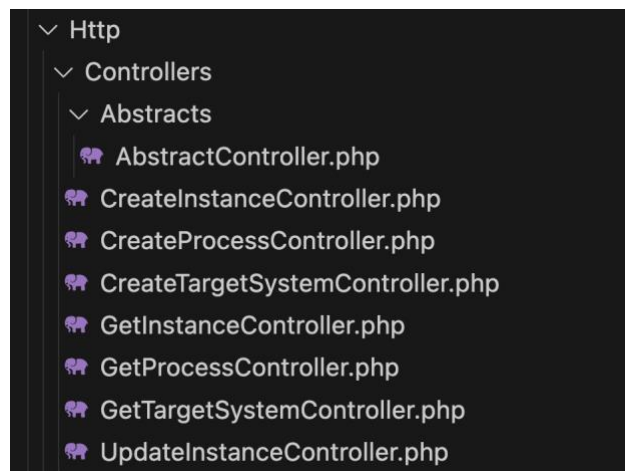


Figura anexo 14. Implementación de controladores

## 7. Validadores de modelos

Estos casos de uso están definidos por una interfaz que es el punto de entrada general de los validadores, la función de entrada principal es mapeada por una clase abstracta que realiza las validaciones de forma genérica. Cada validador concreto define los métodos abstractos requeridos, los cuales modifican el comportamiento general de acuerdo con las necesidades de cada modelo como se muestra en la Figura anexo 15.



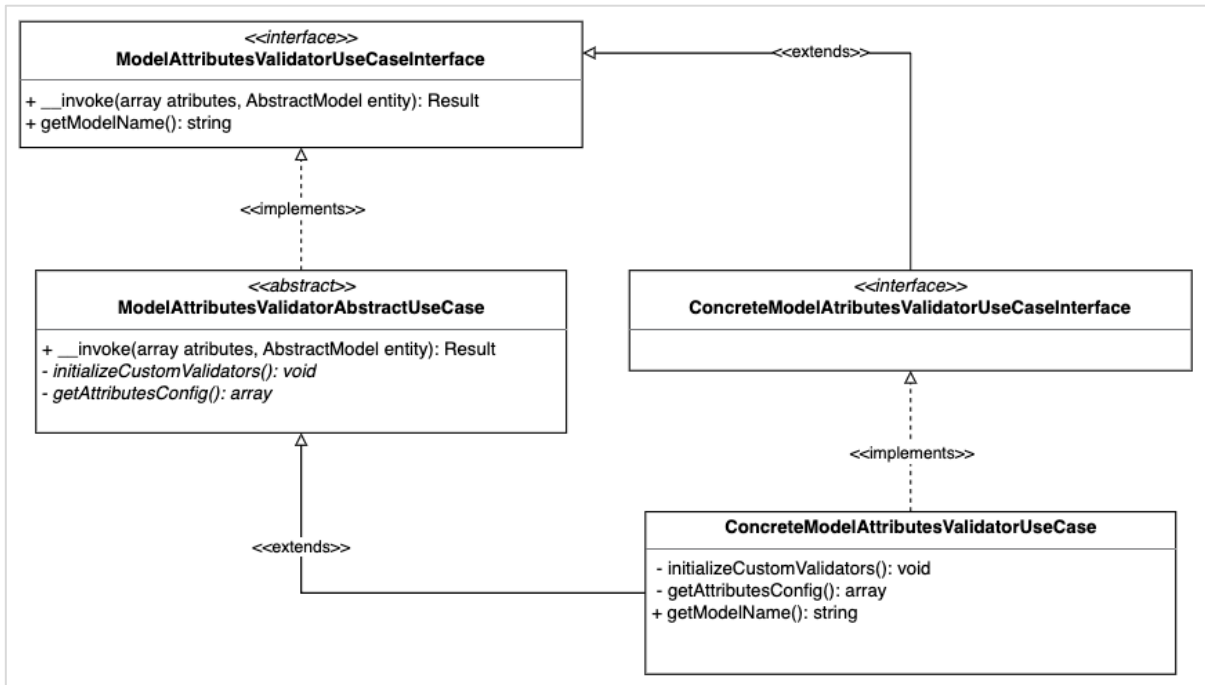


Figura anexo 15. Estructura de casos de uso de validación de modelos

La implementación en el componente se almacenará dentro de `app > UseCases > ModelValidators` como se muestra en la Figura anexo 16.

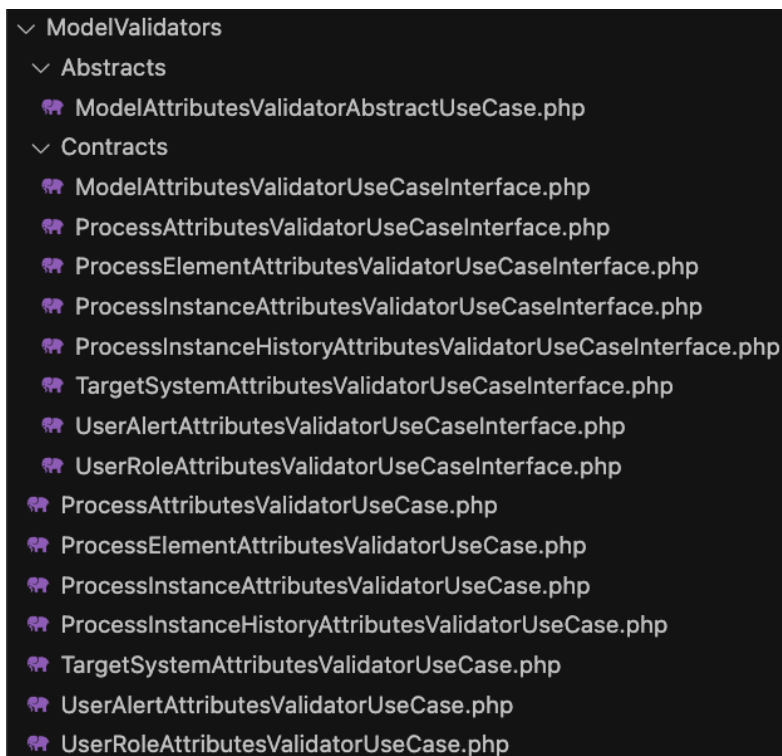


Figura anexo 16. Implementación de casos de uso de validación de modelos

## 8. Creadores de modelos

Los creadores de modelos se definen mediante una interfaz que funciona como punto de entrada general para todos los casos de uso creadores, y la cual es implementada por una

clase abstracta que desarrolla la función de entrada principal, obteniendo los detalles asociados a cada caso particular mediante métodos abstractos que serán mapeados en cada creador concreto. Estos métodos abstractos estarán orientados a mapear los eventos del proceso de creación y el manejo de errores como se muestra en la Figura anexo 17.

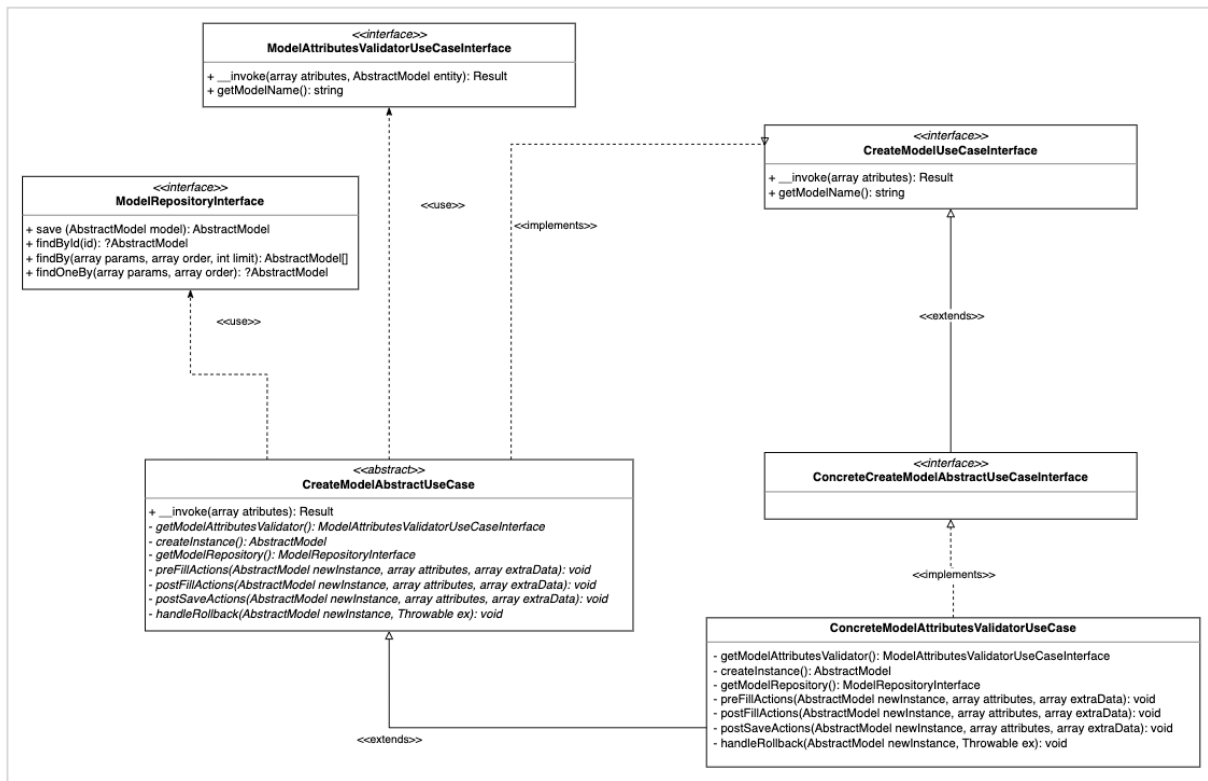


Figura anexo 17. Estructura de casos de uso de creación de modelos

Dentro del componente la implementación de los casos de uso creadores de modelos se ubica dentro de `app > UseCases > Models` como muestra en la Figura anexo 18.

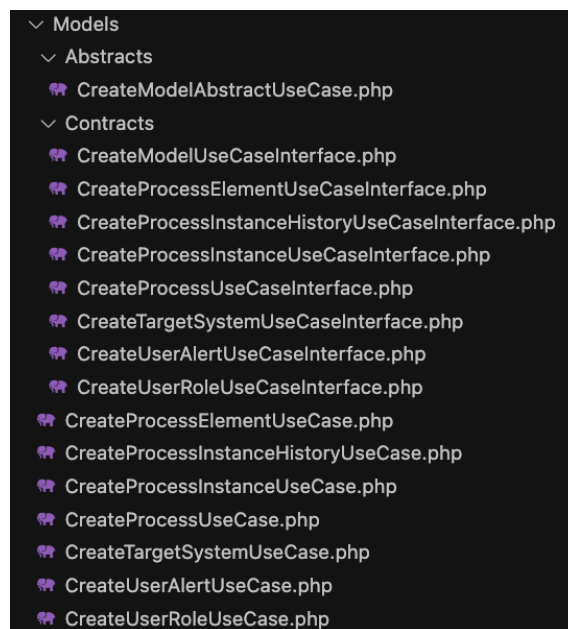


Figura anexo 18. Implementación de casos de uso creadores

## Anexo 5. Validación de requisitos

En este anexo se detalla el proceso de validación de requisitos implementados en el proyecto de desarrollo.

### 1. Servicios expuestos en el API REST

Para la validación de requisitos que están relacionados a servicios del API REST, se lanzó una instancia de ejecución en un servidor local mediante el uso de Docker, y se realizaron peticiones de prueba mediante Postman a los distintos servicios acorde a las definiciones en los requisitos validando que las respuestas cumplan con las condiciones de contenido y códigos HTTP.

En la Figura anexo 19 se muestra un ejemplo de petición al componente de alertas para la creación de un sistema objetivo, en el escenario correcto con la respuesta acorde a la estructura.

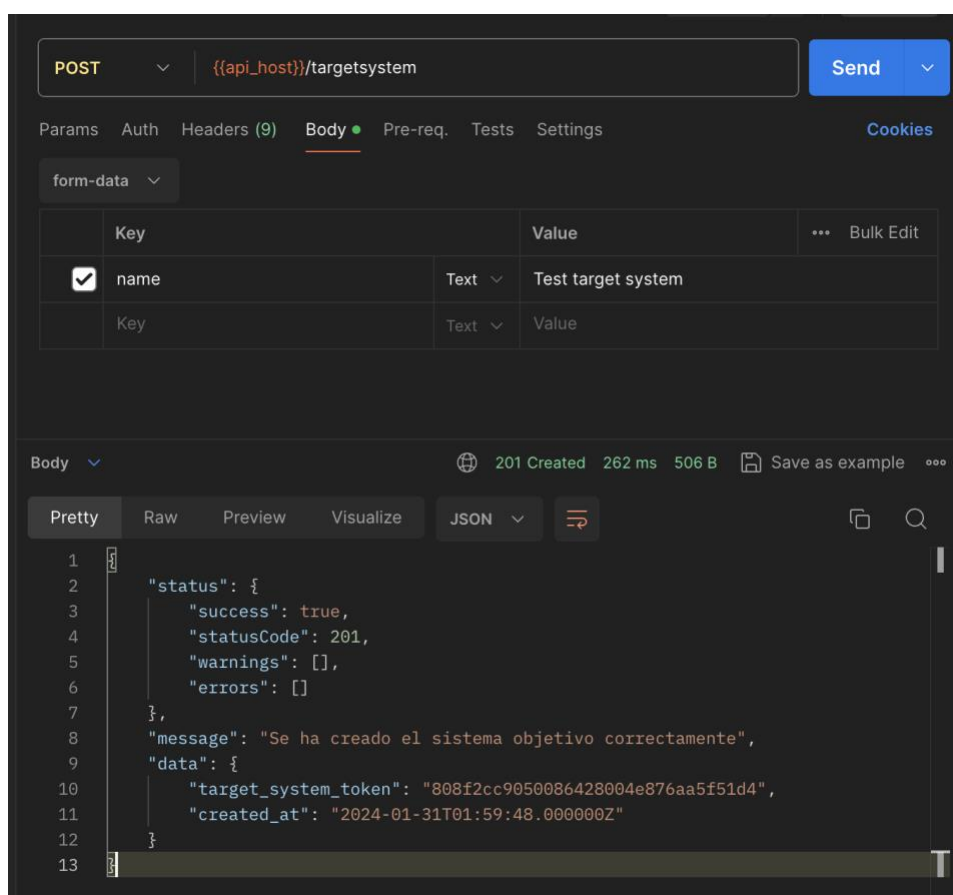


Figura anexo 19. Petición correcta de creación de sistema objetivo

En los escenarios de prueba también se incluyen escenarios incorrectos como el de la Figura anexo 20 donde se muestra una petición al servicio de obtener datos de proceso sin agregar el token de proceso, mostrando respuesta HTTP 422 y la descripción del error acorde a la estructura.

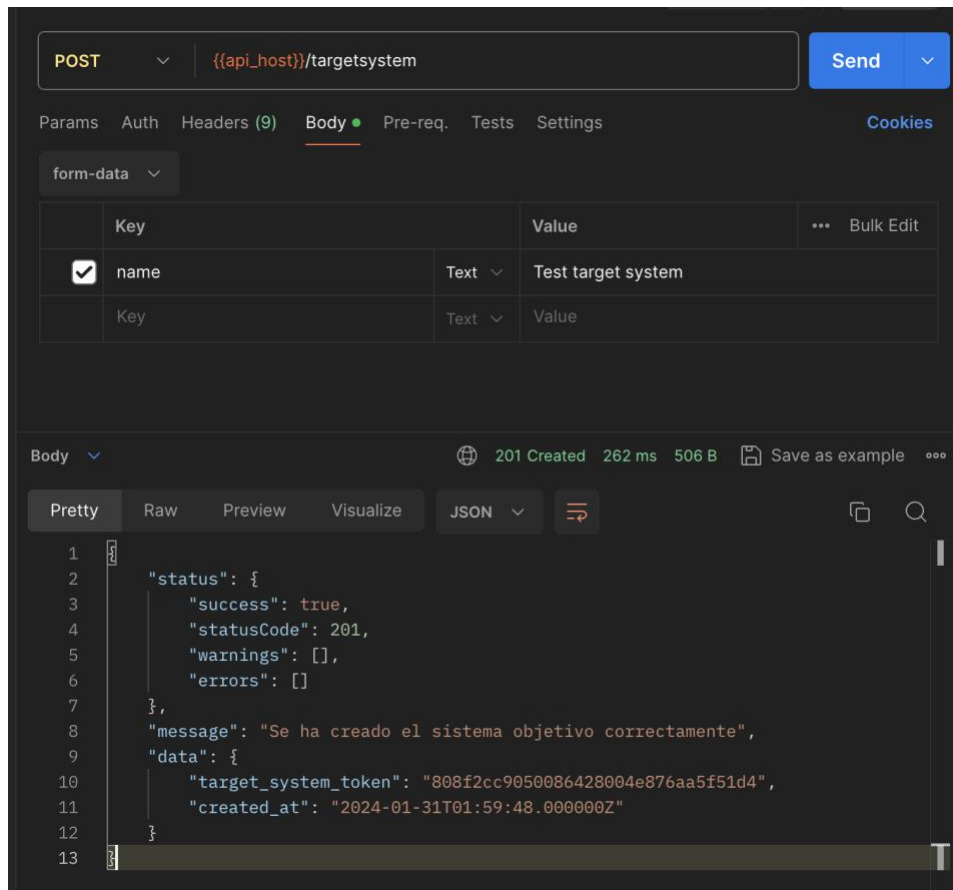


Figura anexo 20. Petición correcta de creación de sistema objetivo

## 2. Casos de uso

La validación de casos de uso cuya ejecución no está expuesta mediante servicios se realiza mediante el uso de test unitarios, los cuales aprovechan la estructura de interfaz/implementación para probar cada unidad de código por separado, estos test se encargan de probar entradas y salidas de cada caso uso garantizando su correcto funcionamiento. Cada caso de uso tiene un archivo de test dentro de la carpeta *tests* con la estructura que se muestra en la Figura anexo 21.

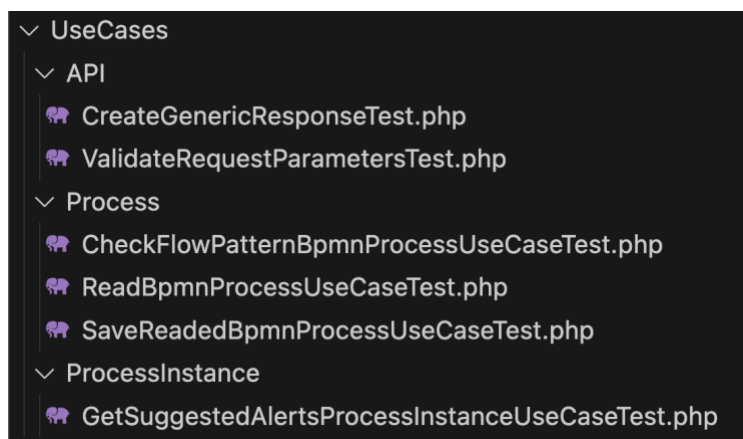


Figura anexo 21. Tests unitarios sobre casos de uso

## Anexo 6. Renderización manual de sugerencias

En este anexo se detalla el proceso para la renderización de sugerencias de alerta de usuario generadas por el componente para el proceso de *Trámite de ventanilla física* del sistema Docxflow®.

### 1. Definición de pautas de renderización

En la descripción del funcionamiento ideal del componente se indica que el sistema objetivo realiza una petición de generación de alertas de usuario al componente cada vez que una instancia de proceso cambie de estado, el componente genera las sugerencias de alertas de usuario las cuales se entregan de vuelta al sistema objetivo en formato *json* con la descripción de las características de la(s) alerta(s), y este último de acuerdo a los lineamientos de su UI renderiza y muestra las sugerencias al usuario. No obstante, en este caso no es posible realizar modificaciones en el código fuente de Docxflow® para que este se comunique con el componente y renderice las sugerencias de alertas de usuario por su cuenta, por lo que las peticiones de alertas de usuario hacia el componente se realizarán de forma manual al igual que la renderización las alertas sugeridas que se evaluarán con los expertos elegidos.

Por lo mencionado anteriormente, se define una renderización estándar de las alertas de usuario sugeridas por el componente, para el caso de las cajas de texto como se muestra en la Figura anexo 22 se renderizarán como una caja con el ícono sugerido en la parte superior, seguido por el título y el contenido hacia abajo, el color sugerido se mostrará en una barra lateral que abarcará el espacio de título y contenido y las acciones sugeridas se mostrarán como botones en la parte inferior, si alguna característica no se incluye dentro de la sugerencia, entonces esta se omitirá del renderizado.



Figura anexo 22. Renderización base de alertas de caja de texto

Dentro del renderizado para la evaluación se tomarán en cuenta las siguientes pautas:

- Si la sugerencia tiene un color definido, este se mostrará tanto en la barra lateral como en el ícono y el contenido se mostrará alineado a la izquierda.

- Si la sugerencia no tiene un color definido, el icono se mostrará en negro, la barra lateral no se incluirá y el contenido se mostrará centrado
- Si la sugerencia incluye título, se mostrará el ícono a la izquierda y junto al título en mayúscula y centrado.
- Si en la sugerencia no se incluye título, el icono se mostrará centrado.

En la Figura anexo 23 se muestra una renderización de ejemplo para la sugerencia de alerta de usuario representación visual: caja de texto, ícono: advertencia, título: Aceptar términos y condiciones, contenido: Debes aceptar los términos y condiciones antes de continuar, color: amarillo, acciones: Aceptar, Rechazar.

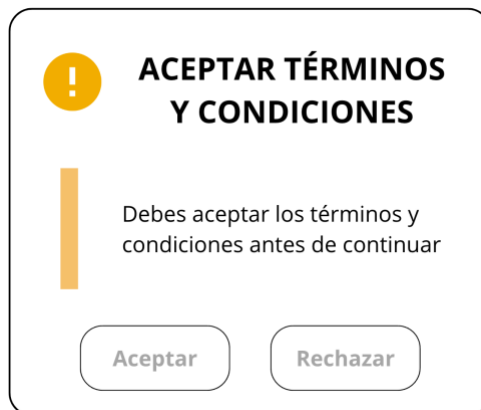


Figura anexo 23. Renderización de ejemplo de alertas de caja de texto con todos los atributos incluidos

En la Figura anexo 24 se muestra un ejemplo de cómo se vería la renderización de la anterior sugerencia si esta no incluyera el atributo color.

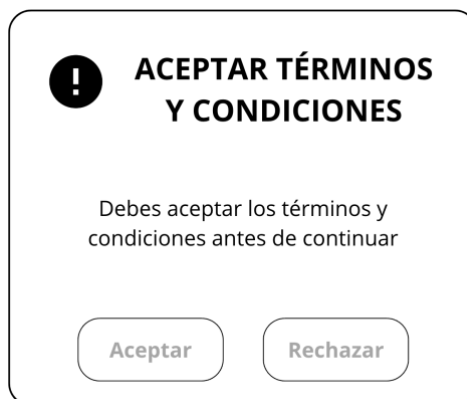


Figura anexo 24. Renderización de ejemplo de alertas de caja de texto sin el atributo color

Por otra parte, en la Figura anexo 25 se muestra la renderización de la sugerencia, si esta no incluyera el atributo título.

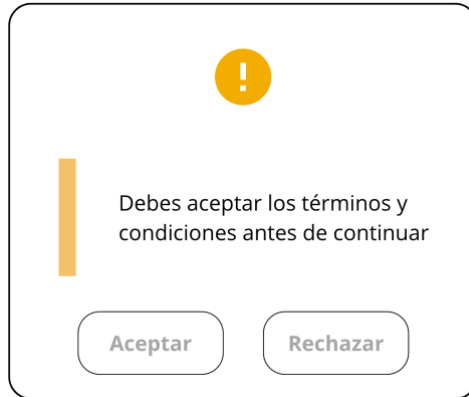


Figura anexo 25. Renderización de ejemplo de alertas de caja de texto sin el atributo título

En el caso de las alertas de palabra simple, y considerando que estas carecen de título y acciones, la base para la renderización de estas será la que se muestra en la Figura anexo 26.

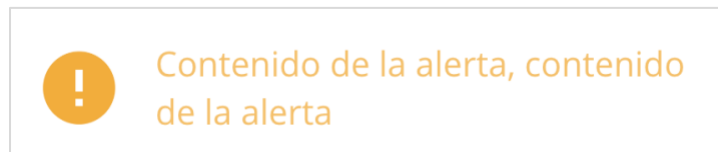


Figura anexo 26. Renderización base de alertas de palabra simple

Este tipo de representación visual sólo tendrá una pauta, que indica que, si la alerta incluye el atributo color, este se aplicará al texto del contenido y al icono.

## 2. Renderización manual de sugerencias

Una vez se han cargado y almacenado los procesos en el componente, para cada de los procesos se crea una instancia dentro del componente mediante una petición al API REST como la de la Figura anexo 27.

```
curl --location 'localhost:8083/instance' \  
--form 'target_system_token="7389fddd472aa5ad410706dcb2cfae92"' \  
--form 'process_token="081e00d3085c1399c8a17fa6f10606d3"'
```

Figura anexo 27. Petición de creación de instancia para proceso en Docxflow

La instancia creada se actualiza de estado mediante una petición al API REST como la mostrada en la Figura anexo 28, en la cual se indica el identificador del elemento del proceso al que se enviará la instancia.

```
curl --location --request PUT 'localhost:8083/instance' \
--header 'Content-Type: application/json' \
--data '{
  "target_system_token": "7389fddd472aa5ad410706dcb2cfae92",
  "process_instance_token": "f009951081ff72cb532aead4d746f1d8",
  "next_element_id": "Event_085hais"
}'
```

Figura anexo 28. Petición de actualización de instancia para instancia de proceso de Docxflow

Cada instancia se actualiza tantas veces como elementos haya en el proceso, garantizando que haya pasado por cada uno de ellos. Cada llamado al servicio del componente, además de actualizar el estado de la instancia, retorna las sugerencias de alerta de usuario (si las hay).

Luego de ejecutar el proceso de actualización para cada instancia en cada uno de los elementos del proceso para la que fue creada, los puntos en los que el componente retornó sugerencias de alertas se detallan en la Tabla anexo 37.

Tabla anexo 37. Sugerencias creadas por el componente para procesos del sistema Docxflow

| No | Proceso           | Elemento                                | Sugerencias del componente  |
|----|-------------------|---|---|
| 1  | Radicar documento | Actividad Cargar información de trámite | <pre>{   "type": "warning",   "visual_representation": "message-box",   "color": "orange",   "message": "Revisa los datos antes de continuar",   "icon": "fa-exclamation-triangle",   "title": "¿Estás seguro?",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }</pre> |
| 2  | Radicar documento | Actividad Cargar documentos             | <pre>{   "type": "warning",   "visual_representation": "message-box",   "color": "orange",   "message": "Revisa los datos antes de continuar",   "icon": "fa-exclamation-triangle",   "title": "¿Estás seguro?",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }</pre> |
| 3  | Radicar documento | Evento de fin                           | <pre>{   "type": "success",   "visual_representation": "message-box",   "color": "green",   "message": "Documento radicado ",   "icon": "fa-check-circle",   "title": "Terminado",   "alert_moment": "transition-out",   "actions": [] }</pre>  |



|   |                   |  |  |
|---|-------------------|--|--|
| 4 | Repartir trámite  | Gateway de decisión exclusiva tipo de distribución | <pre>{   "type": "feedforward",   "visual_representation": "message-box",   "color": "blue",   "message": "Continuar mediante Envío directo",   "icon": "fa-question-circle",   "title": "Confirmar elección",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }, {   "type": "feedforward",   "visual_representation": "message-box",   "color": "blue",   "message": "Continuar mediante Envío UD",   "icon": "fa-question-circle",   "title": "Confirmar elección",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }</pre>       |
| 5 | Repartir trámite  | Evento de fin                                      | <pre>{   "type": "success",   "visual_representation": "message-box",   "color": "green",   "message": "Trámite repartido ",   "icon": "fa-check-circle",   "title": "Terminado",   "alert_moment": "transition-out",   "actions": [] }</pre>  |
| 6 | Responder trámite | Gateway de división exclusiva dar respuesta        | <pre>{   "type": "feedforward",   "visual_representation": "message-box",   "color": "blue",   "message": "Continuar mediante Reparte trámite",   "icon": "fa-question-circle",   "title": "Confirmar elección",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }, {   "type": "feedforward",   "visual_representation": "message-box",   "color": "blue",   "message": "Continuar mediante Da respuesta",   "icon": "fa-question-circle",   "title": "Confirmar elección",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }</pre> |



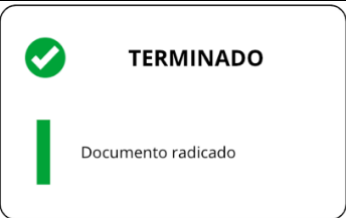


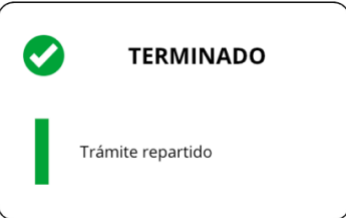
|   |                   |                               |   |
|---|-------------------|-------------------------------|---|
| 7 | Responder trámite | Fin del proceso               | <pre>{   "type": "success",   "visual_representation": "message-box",   "color": "green",   "message": "Trámite respondido ",   "icon": "fa-check-circle",   "title": "Terminado",   "alert_moment": "transition-out",   "actions": [] }</pre>  |
| 8 | Finalizar trámite | Gateway de división exclusiva | <pre>{   "type": "feedforward",   "visual_representation": "message-box",   "color": "blue",   "message": "Continuar mediante Finalizar trámite",   "icon": "fa-question-circle",   "title": "Confirmar elección",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }, {   "type": "feedforward",   "visual_representation": "message-box",   "color": "blue",   "message": "Continuar mediante Cargar documentos",   "icon": "fa-question-circle",   "title": "Confirmar elección",   "alert_moment": "transition-out",   "actions": [     "Continuar",     "Cancelar"   ] }</pre> |
| 9 | Finalizar trámite | Evento de fin                 | <pre>{   "type": "success",   "visual_representation": "message-box",   "color": "green",   "message": "Trámite finalizado ",   "icon": "fa-check-circle",   "title": "Terminado",   "alert_moment": "transition-out",   "actions": [] }</pre>  |






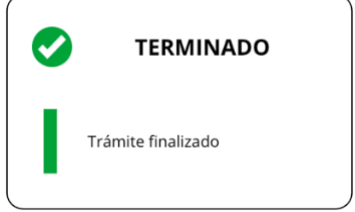
## 2.2. Renderización de sugerencias

Una vez se han generado las sugerencias de alerta de usuario mediante los servicios del componente, estas se renderizan de acuerdo con las condiciones predefinidas, generando así las alertas presentadas en la Tabla anexo 38, estas alertas renderizadas serán las que los colaboradores elegidos evaluarán mediante las preguntas definidas en la encuesta.

*Tabla anexo 38. Sugerencias renderizadas para procesos del sistema Docxflow*

| No | Proceso | Elemento | Sugerencias renderizadas |
|----|---------|----------|--------------------------|
|----|---------|----------|--------------------------|

|   |                   |  |   |
|---|-------------------|--|---|
| 1 | Radicar documento | Actividad Cargar información de trámite            |   |
| 2 | Radicar documento | Actividad Cargar documentos                        |   |
| 3 | Radicar documento | Evento de fin                                      |   |
| 4 | Repartir trámite  | Gateway de decisión exclusiva tipo de distribución |   |
| 5 | Repartir trámite  | Evento de fin                                      |   |

|   |                   |   |   |
|---|-------------------|---|---|
| 6 | Responder trámite | Gateway de división exclusiva dar respuesta | <div data-bbox="987 195 1339 489">  <p><b>CONFIRMAR ELECCIÓN</b></p> <p>Continuar mediante Reparte trámite</p> <p>Continuar Cancelar</p> </div> <div data-bbox="987 506 1339 800">  <p><b>CONFIRMAR ELECCIÓN</b></p> <p>Continuar mediante Dar respuesta</p> <p>Continuar Cancelar</p> </div>               |
| 7 | Responder trámite | Fin del proceso                             | <div data-bbox="987 825 1339 1043">  <p><b>TERMINADO</b></p> <p>Trámite respondido</p> </div>  |
| 8 | Finalizar trámite | Gateway de división exclusiva               | <div data-bbox="987 1060 1339 1354">  <p><b>CONFIRMAR ELECCIÓN</b></p> <p>Continuar mediante Cargar documentos</p> <p>Continuar Cancelar</p> </div> <div data-bbox="987 1371 1339 1665">  <p><b>CONFIRMAR ELECCIÓN</b></p> <p>Continuar mediante Finalizar trámite</p> <p>Continuar Cancelar</p> </div> |
| 9 | Finalizar trámite | Evento de fin                               | <div data-bbox="987 1682 1339 1892">  <p><b>TERMINADO</b></p> <p>Trámite finalizado</p> </div>  |

