

**PROCESO LIVIANO DE DESARROLLO DE SOFTWARE PARA
SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO,
PEQUEÑAS Y MEDIANAS EMPRESAS**



Jeison Alexander Arcos Muñoz

Director: MSc. Wilson Libardo Pantoja
Codirector: PhD. César Jesús Pardo Calvache

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Sistemas
Grupo de I+D en Tecnologías de la información (GTI)
Línea investigación: Ingeniería del Software
Popayán, Septiembre de 2019

Tabla de contenido

Capítulo 1. Introducción.....	1
1.1. Introducción.....	1
1.2. Problemática y justificación.....	2
1.3. Objetivos.....	4
1.3.1. Objetivo General.....	4
1.3.2. Objetivos Específicos.....	4
1.4. Estrategia de investigación.....	4
1.5. Estructura del documento.....	6
Capítulo 2. Marco teórico y estado del arte.....	8
2.1. Marco teórico.....	8
2.1.1. Proceso de ingeniería web.....	8
2.1.2. Proceso software.....	9
2.1.3. Aplicación web.....	9
2.1.4. MiPymes.....	10
2.1.5. Enfoques Ágiles.....	11
2.2. Estado del arte.....	12
2.2.1. Pregunta de investigación.....	12
2.2.2. Cadenas de búsqueda.....	12
2.2.3. Recursos literarios.....	12
2.2.4. Proceso de selección de estudios.....	12
2.2.4.1. Criterios de inclusión y exclusión.....	13
2.2.4.2. Clasificación de los trabajos relacionados.....	13
2.2.4.3. Clasificación de artículos de acuerdo a su contenido.....	14
2.2.5. Propuestas existentes.....	15
2.2.5.1. Trabajos relacionados que involucran modelos.....	15
2.2.5.1.1 Model2Roo: A Model Driven approach for web application development based on the eclipse modeling framework and Spring Roo.....	15
2.2.5.2. Trabajos relacionados que involucran metodologías.....	15
2.2.5.2.1 Agile development of web application by supporting process execution and extended UML model.....	15
2.2.5.2.2 Web development life cycle: a new methodology for developing web applications.....	16
2.2.5.2.3 A component based methodology for web application development.....	16
2.2.5.2.4 Development of object-oriented analysis and design methodology for secure web applications.....	17
2.2.5.2.5 Mockup-drive fast-prototyping methodology for web application Development.....	17
2.2.5.2.6 A Proposed methodology for web Development.....	18
2.2.5.2.7 Construyendo aplicaciones web con una metodología de diseño orientado a objetos.....	19
2.2.5.2.8 Las metodologías de desarrollo y mejora de la calidad de las aplicaciones web.....	19
2.2.5.2.9 Scrum y Xp desde las trincheras.....	20
2.2.5.3. Trabajos relacionados que involucran enfoques.....	21
2.2.5.3.1 Web applications development by formal refinement approach.....	21
2.2.5.3.2 Test driven development of web applications: a lightweight approach.....	21

2.2.5.3.3	A page-centric approach to web application development.....	21
2.2.5.3.4	An object oriented approach to web-based applications design.....	22
2.2.5.3.5	A rule based approach to web based application development.....	22
2.2.5.4.	Trabajos relacionados que involucran frameworks.....	23
2.2.5.4.1	A theoretical agile process framework for web applications development in small software firms.....	23
2.2.5.5.	Trabajos relacionados que involucran estudios de caso.....	24
2.2.5.6.	Desarrollo e implementación de un sistema web para generar valor en una pyme aplicando una metodología ágil.....	24
2.2.6.	Caracterización de los trabajos relacionados.....	24
2.2.7.	Aportes.....	25
2.3.	Caracterización del Proceso.....	26
Capítulo 3.	Definición del proceso DEWE.....	28
3.1.	Introducción.....	28
3.2.	Creación y evaluación del proceso.....	28
3.3.	Alcance del proceso.....	29
3.4.	Principios ágiles.....	29
3.5.	Plantilla para la definición del proceso.....	29
3.6.	Términos genéricos.....	31
3.7.	Identificación de roles.....	31
3.8.	Visión general de la propuesta.....	32
3.9.	Proceso liviano de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.....	33
Capítulo 4.	Evaluación de la propuesta por medio de un grupo focal.....	54
4.1.	Grupo focal (Focus group).....	54
4.1.1.	Realización del grupo focal.....	54
4.1.1.1.	Definición de elementos utilizados para realizar el Grupo Focal.....	54
4.1.1.2.	Selección de los participantes.....	56
4.1.1.3.	Sesión del grupo focal.....	58
4.1.1.4.	Captura y registro de la información en el grupo focal.....	60
4.1.2.	Reporte de resultados.....	60
4.1.2.1.	Resultados del cuestionario.....	61
4.1.2.2.	Análisis estadístico.....	62
4.1.2.3.	Observaciones y mejoras sugeridas por los integrantes del grupo focal.....	74
Capítulo 5.	Conclusiones y lecciones aprendidas.....	77
5.1.	Conclusiones.....	77
5.2.	Lecciones aprendidas.....	78
Anexo A.	Agenda de trabajo.....	84
Anexo B.	Cuestionario de preguntas sobre el proceso utilizado en el grupo focal.....	85
Anexo C.	Ficha de asistencia al grupo focal.....	90
Anexo D.	Encuestas diligenciadas grupo focal.....	92
Anexo E.	Ficha de asistencia al grupo focal diligenciada.....	110
Anexo F.	Primera versión del proceso propuesto.....	114
Anexo G.	Plantilla del criterio de listo.....	137
Anexo H.	Plantilla del criterio de hecho.....	140
Anexo I.	Estilos arquitectónicos.....	143
Anexo J.	Plantilla ambientes de desarrollo y pruebas.....	150
Anexo K.	Plantilla de lecciones aprendidas.....	153

Anexo L. Plantilla para pruebas.....	156
Anexo M. Herramientas pruebas funcionales.....	160
Anexo N. Herramientas pruebas de usabilidad.....	163
Anexo O. Herramientas pruebas de seguridad.....	166
Anexo P. Herramientas pruebas de rendimiento o desempeño.....	169
Anexo Q. Herramientas pruebas de accesibilidad.....	173
Anexo R. Herramientas pruebas de diseño adaptativo.....	175
Anexo S. Herramientas pruebas de link testing.....	177
Anexo T. Herramientas pruebas de browser testing.....	180
Anexo U. Resumen proceso para grupo focal.....	182

Lista de figuras

Figura 1 . Artículos de acuerdo a contenido.....	14
Figura 2. Caracterización del proceso.....	26
Figura 3. Visión general del proceso.....	32
Figura 4. Participantes del grupo focal.....	59
Figura 5. Conteo respuestas pregunta 1.....	63
Figura 6. Conteo respuestas pregunta 2.....	65
Figura 7. Conteo respuestas pregunta 3.....	66
Figura 8. Conteo respuestas pregunta 4.....	67
Figura 9. Conteo respuestas pregunta 5.....	69
Figura 10. Conteo respuestas pregunta 6.....	70
Figura 11. Conteo respuestas pregunta 7.....	71
Figura 12. Conteo respuestas pregunta 8.....	72
Figura 13. Conteo respuestas pregunta 9.....	73

Lista de tablas

Tabla 1 . Clasificación por año de los artículos relacionados.....	13
Tabla 2 . Caracterización de los trabajos relacionados.....	24
Tabla 3 . Caracterización del proceso.....	27
Tabla 4. Plantilla para la definición del proceso.....	30
Tabla 5. Términos del genéricos.....	31
Tabla 6. Caracterización del proceso.....	33
Tabla 7. Artefactos utilizados en el grupo focal.....	55
Tabla 8. Participantes del grupo focal.....	56
Tabla 8. Agenda grupo focal (27/03/2019).....	60
Tabla 9. Resultados encuesta.....	61
Tabla 10. Comentarios realizados de la pregunta 1.....	64
Tabla 11. Comentarios realizados de la pregunta 2.....	65
Tabla 12. Comentarios realizados de la pregunta 3.....	66
Tabla 13. Comentarios realizados de la pregunta 4.....	68
Tabla 14. Comentarios realizados de la pregunta 5.....	69
Tabla 15. Comentarios realizados de la pregunta 6.....	71
Tabla 16. Comentarios realizados de la pregunta 7.....	72
Tabla 17. Comentarios realizados de la pregunta 8.....	73
Tabla 18. Comentarios realizados de la pregunta 9.....	74
Tabla 19. Resultado observaciones y mejoras sugeridas.....	75

Capítulo 1. Introducción

Este capítulo presenta una descripción sobre la problemática a ser abordada, los objetivos definidos, la estrategia utilizada para realizar la investigación y la descripción de la solución propuesta. Finalmente, se presenta la estructura del documento y un pequeño resumen de cada capítulo.

1.1. Introducción

Con el paso del tiempo las empresas están dando gran importancia a los productos basados en la web, todo esto debido a que se hace evidente la necesidad de utilizar este tipo de tecnologías. Los desarrollos de aplicaciones web cada vez son más exigentes y complejos debido a nuevas necesidades que se crean a medida que la industria se desarrolla [1], cada vez aparecen nuevos requerimientos para la creación de productos, los cuales tienen tiempos de desarrollo muy limitados. Los desarrollos de aplicaciones web son muy importantes ya que aportan en gran medida a la actividad económica que existe hoy en día [2].

Para abordar los problemas que surgen en el desarrollo de aplicaciones web se hace necesario tener un proceso que guíe el desarrollo de productos de una manera ordenada, con la cual se puedan cumplir los diferentes requerimientos que estos desarrollos exigen. Aunque existen diferentes propuestas que intentan abordar los problemas que surgen con el desarrollo de aplicaciones web [3], cada propuesta se centra en elementos específicos que son considerados dependiendo del desarrollo a crear, algunas empresas realizan esfuerzos por adaptar metodologías para desarrollos web sin tener en cuenta que las aplicaciones web tienen características particulares que no todas las aplicaciones poseen como por ejemplo: la estética, multiculturalismo, presión competitiva, disponibilidad, calidad del servicio, cambio continuo, entre otras.

Se ha realizado una revisión de la literatura en busca de una propuesta que cumpla con las características que exigen las aplicaciones web. Debido a la falta de una propuesta que tenga en cuenta las diferentes características propias de los

desarrollos web, se pretende en este documento dar una solución que permita guiar a micro, pequeñas y medianas empresa en el desarrollo de software de una manera ordenada y que les permita minimizar riesgos en cuanto al desarrollo de proyectos web.

1.2. Problemática y justificación

La evolución y crecimiento de la web ha provocado que las empresas vean en el desarrollo de aplicaciones web un elemento de ventaja competitiva en relación con otras empresas en su mismo segmento de mercado [4]. Aunque el desarrollo web es una disciplina joven, se puede ver en la actualidad la gran aceptación que ha tenido por parte de las personas.

Las micro, pequeñas y medianas empresas (MiPyMEs) han tenido un gran impacto en la economía no solo a nivel nacional sino también internacional. "Las MiPyMEs desarrolladoras de software por su parte, han permitido extender el mercado de los países en áreas económicas completamente nuevas y de alto impacto para otros sectores", esto ha permitido que la industria del software se postule como uno de los principales sectores para muchos países como por, ejemplo: India, Irlanda, Uruguay, Estados Unidos, Brasil, México, Colombia, entre otros [5].

La velocidad con la cual cambian las tecnologías actualmente genera un cambio continuo en la construcción de aplicaciones web, cuyo proceso de desarrollo debe tener en cuenta características como:

- **Ciclo de desarrollo corto:** estudios empíricos muestran que el tiempo de desarrollo de aplicaciones web es extremadamente corto, normalmente no exceden los seis meses y su promedio de duración es menor a tres meses [6-8].
- **Requisitos cambiantes:** Los requisitos para una aplicación web a menudo surgen solo durante su desarrollo, o están sujetos a cambios importantes tanto en su contenido como en su tecnología [6].
- **Liberaciones con fechas establecidas:** Los intervalos entre cada lanzamiento de un prototipo del producto, son relativamente cortas (actualmente entre dos y cinco semanas) [6,9].
- **Desarrollo en paralelo:** Las actividades metodológicas en las fases de diseño, implementación y aseguramiento de la calidad se trabajan concurrentemente para diferentes versiones. Varios equipos de desarrollo pequeños trabajan en tareas similares en paralelo [7].
- **Reúso e integración de componentes:** Si se desarrolla un componente reutilizable en un proyecto, debe desarrollarse en coordinación con otros proyectos que usaran este componente [6].

- **Adaptación del proceso al nivel de complejidad:** El proceso tiene que ser similar a los procesos livianos cuando la aplicación web tiene un nivel de complejidad menor en las primeras fases de desarrollo. Sin embargo, el proceso tiene que funcionar de manera análoga a los procesos pesados cuando la aplicación web alcanza un nivel de complejidad más alto en las fases de desarrollo posteriores [6].

Además de estas características, la calidad es particularmente crítica cuando se desarrollan aplicaciones web. La usabilidad, seguridad, desempeño y mantenimiento son aspectos que necesitan especial atención [8].

El éxito en la construcción de una aplicación web sugiere seguir un proceso que dirija el desarrollo de la manera más fluida y eficiente. Este proceso permite al equipo de desarrollo extraer y analizar adecuadamente los requisitos, traducirlos en diseños correspondientes de la aplicación, implementar la aplicación eligiendo las tecnologías e instrumentos adecuados, probar y validar el resultado, operar, mantener y evolucionar la aplicación según sea necesario [9]. Si no se sigue un proceso que permita llevar un control sobre el desarrollo, muy posiblemente se fracasará.

Las aplicaciones web se caracterizan por tener requisitos desconocidos o variables. Al no tener requisitos fijos, hay una gran posibilidad de que el producto tenga inconvenientes para ser desarrollado con el tiempo y dinero acordados al inicio del desarrollo de la aplicación web. Una buena opción para ayudar a controlar los cambios de los requisitos son las metodologías ágiles, debido a que se caracterizan por poseer iteraciones cortas, pruebas continuas, equipos autoorganizados, constante colaboración y frecuente replanificación basada en la realidad actual [10].

Existen procesos de desarrollo adaptados a partir del Proceso Unificado Rational (RUP) o de metodologías ágiles, que permiten la organización, dirección y gestión de procesos de desarrollo para ambientes web [6]. Sin embargo, no cubren a plenitud los requerimientos propios de esta disciplina, no son procesos o metodologías que se han pensado específicamente para desarrollar aplicaciones web, por lo cual, al utilizarlas en un proyecto, se obtiene un riesgo muy grande debido a que son adaptadas y cumplen en baja proporción las necesidades específicas que requiere un desarrollo web [6].

Con la adopción de métodos de desarrollo ágil en los últimos años, ha sido posible realizar una comparación de los resultados de proyectos entre ágiles y cascada tradicionales. En todos los tamaños de proyectos, los enfoques ágiles resultaron en proyectos más exitosos y con menos fallas directas [11]. La diferencia obtenida en el éxito de proyectos entre desarrollos ágiles y de cascada son: En proyectos de gran tamaño la diferencia es de 15% de éxito, en proyectos de tamaño medio fue de 20% y en proyectos de tamaño pequeños fue de 14%, sobresaliendo las metodologías ágiles en el desarrollo de productos software [11].

A razón de lo anterior, surge la siguiente la pregunta **¿Es posible definir un proceso liviano que soporte el desarrollo de aplicaciones web para MiPyMEs, teniendo en cuenta características de los procesos de ingeniería web y aplicaciones web actuales?** El presente trabajo de grado pretende definir un proceso para el desarrollo de aplicaciones web, el cual se adapte al inherente nivel de complejidad que posee un proyecto de desarrollo de esta naturaleza. Para ello se buscará identificar los requerimientos de un proceso de desarrollo de aplicaciones web y diseñar un proceso liviano que pueda ser utilizado por las MiPyMEs.

1.3. Objetivos

1.3.1. Objetivo General

- ✓ Definir un proceso de desarrollo de software liviano que facilite el desarrollo de aplicaciones WEB en micro, pequeñas y medianas empresas.

1.3.2. Objetivos Específicos

- ✓ Caracterizar los trabajos relacionados de acuerdo a un conjunto de factores asociados al proceso de ingeniería web, como son: ciclo de desarrollo corto, requisitos cambiantes, liberaciones con fechas establecidas, desarrollo en paralelo, reuso e integración de componentes, adaptación del proceso al nivel de complejidad. Además, las características particulares de las aplicaciones web y los atributos de calidad de seguridad, usabilidad, desempeño y mantenibilidad.
- ✓ Construir un proceso liviano que facilite el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas de software a partir de las características encontradas y que no hayan sido abordadas en las propuestas existentes.
- ✓ Evaluar el proceso propuesto a través de su implementación en un grupo focal (Focus Group) [12,13] con el cual se pueda obtener una retroalimentación y generar una versión mejorada

1.4. Estrategia de investigación

Para llevar a cabo la ejecución del proyecto propuesto, se utilizará el método Investigación Acción con múltiples ciclos de forma lineal [14] y Focus Group [12,13]. Teniendo en cuenta las fases y actividades propuestas por esta metodología, para el desarrollo de esta propuesta se llevarán a cabo 4 ciclos de investigación, a continuación se describen los ciclos y las actividades a llevar a cabo de manera secuencial e incremental para el desarrollo del proyecto.

- **Ciclo 1. Análisis Conceptual:** En esta etapa se realiza un análisis del estado del arte acerca del desarrollo ágil de aplicaciones web en micro, pequeñas y medianas empresas. Esta primera fase nos permite identificar procesos o propuestas existentes con elementos importantes a considerar para la definición de la solución.

En esta fase se llevarán a cabo las siguientes actividades:

- **Actividad 1.1: Análisis de la literatura:** Realizar un proceso de búsqueda en la literatura que permita recolectar información relacionada a los procesos de desarrollo ágil para aplicaciones web en MiPyMEs.
- **Actividad 1.2: Síntesis de la literatura:** Acotar los resultados obtenidos en la revisión de la literatura, únicamente a los estudios que traten sobre procesos y metodologías ágiles para el desarrollo de aplicaciones web en MiPyMEs.
- **Actividad 1.3: Estudio de la literatura:** Identificar los elementos importantes a tener en cuenta para la especificación de un proceso de desarrollo web, que permita ser guiado a través de un enfoque ágil en pequeñas y medianas empresas.
- **Ciclo 2. Desarrollo de la propuesta:** En este ciclo se pretende desarrollar un proceso de desarrollo liviano, el cual pueda apoyar en la construcción de las aplicaciones web en MiPyMEs. Para el desarrollo de esta propuesta se tendrá en cuenta material relacionado con metodologías, modelos, proceso y marcos de desarrollo de aplicaciones web, obtenido mediante la investigación de la literatura.
 - **Actividad 2.1:** Analizar los requerimientos para el proceso de desarrollo de una aplicación web.
 - **Actividad 2.2:** Identificar las características de las aplicaciones web.
 - **Actividad 2.3:** Identificar las características de los procesos ágiles.
 - **Actividad 2.4:** Definir las etapas necesarias que harán parte del proceso.
 - **Actividad 2.5:** Definir los artefactos necesarios en cada etapa del proceso.
 - **Actividad 2.6:** Definir los roles implicados en el proceso.
 - **Actividad 2.7:** Modelar el proceso liviano de desarrollo web.
- **Ciclo 3. Evaluación de la propuesta:** Para la evaluación de esta propuesta se ha optado por utilizar el método de focus group [12,13]. El focus group se realiza con personas experimentadas en cuanto al desarrollo de aplicaciones web.

- **Actividad 3.1. Planificación:** Se realizan actividades como: capacitación, coordinación, organización y diseño del focus group.
- **Actividad 3.2. Acción:** Se realiza la ejecución del focus group teniendo en cuenta la planificación y diseño generados en la actividad 3.1.
- **Actividad 3.3. Observación:** Obtener los datos generados de la ejecución e intervención durante el Focus Group.
- **Actividad 3.4. Reflexión:** Generar un reporte como resultado de la reflexión y el análisis de los datos obtenidos durante la ejecución del Focus Group. De igual manera, se lleva a cabo la retroalimentación y evaluación del aprendizaje logrado.
- **Ciclo 4. Documentación y socialización:** Las actividades a desarrollar en esta etapa, tienen la intención de dar a conocer los resultados de la investigación realizada y el aporte en cuanto a la temática de proceso liviano para el desarrollo de aplicaciones web. Las actividades de este ciclo se presentan a continuación.
 - **Actividad 4.1.** Elaboración del documento (monografía) que contiene el estudio detallado en cuanto al proceso liviano para el desarrollo de aplicaciones web en MiPyMEs.
 - **Actividad 4.2.** Elaboración de un artículo en el cual se proporciona una visión general del contenido de la propuesta y los resultados obtenidos al realizar el focus group.
 - **Actividad 4.3.** Dar a conocer los resultados obtenidos en la investigación y creación de la propuesta del proceso liviano para el desarrollo de aplicaciones web en MiPyMEs.

1.5. Estructura del documento

La investigación realizada durante la realización de este trabajo está conformado por 5 capítulos, los cuales son descritos a continuación:

Capítulo 2: Marco teórico, estado del arte y caracterización del proceso. Este capítulo presenta algunos conceptos utilizados en la propuesta, un resumen de los recursos encontrados durante la revisión sistemática de la literatura (artículos que sirvieron como insumo para el desarrollo de la propuesta) y una caracterización del proceso, la cual contiene elementos importantes que se tuvieron en cuenta para la creación del proceso.

Capítulo 3: Propuesta. En este capítulo se describe el proceso creado. Para la especificación del proceso, se utilizó como guía el patrón de procesos propuesto por

competisoft. La propuesta construida está conformada por: fases, actividades, tareas, entradas, salidas, roles y un flujo de ejecución que sirven como guía para la implementación en una empresa.

Capítulo 4: Evaluación de la propuesta. En este capítulo se presenta el grupo focal realizado para evaluar la propuesta desarrollada. Para la realización del grupo focal se realizó una planeación con una serie de actividades a desarrollar, de las cuales algunas de ellas arrojaron resultados(encuestas realizadas, observaciones y sugerencias) que están contenidos en este capítulo.

Capítulo 5: Conclusiones y trabajos futuros. En este capítulo se presentan las conclusiones, lecciones aprendidas, trabajos futuros y aportes realizados, que fueron obtenidos durante la realización de la propuesta desarrollada.

Capítulo 2. Marco teórico y estado del arte.

En este capítulo se presenta el marco teórico referente a los modelos, metodologías, enfoques, frameworks y casos de estudio para el desarrollo de aplicaciones web. Se proporciona una revisión sistemática de la literatura, la cual tuvo como objetivo identificar elementos importantes a tener en cuenta, al momento de la creación de un proceso para el desarrollo de aplicaciones web.

2.1. Marco teórico

2.1.1. Proceso de ingeniería web

Es un proceso que contiene una serie de actividades, tareas, artefactos, roles y disciplinas, que deben ser ejecutadas por un equipo para crear una aplicación web [6]. Esto quiere decir que este proceso debe tener en cuenta características propias de las aplicaciones web como: la estética, multiculturalismo, presión competitiva, disponibilidad, calidad del servicio, cambio continuo, entre otras. Además de las características de las aplicaciones web, tiene en cuenta características del proceso como son: ciclo de desarrollo corto, requisitos cambiantes, liberaciones con fechas establecidas, desarrollo en paralelo, rehúso e integración de componentes, adaptación del proceso al nivel de complejidad [6].

Algunas de las características de los procesos de ingeniería web también están presentes en los procesos de desarrollo de software tradicionales. La diferencia entre estos dos procesos está en la ocurrencia con la que aparecen las características y en que intensidad son abordados en las diferentes etapas del desarrollo [6].

Un proceso de ingeniería web pretende mejorar el desarrollo de aplicaciones web, dando un mayor nivel de calidad al producto.

2.1.2. Proceso software

“Un proceso es un conjunto de actividades, acciones y tareas que se ejecutan cuando va a crearse algún producto del trabajo” [15]. Las actividades buscan cumplir con un gran objetivo, y son realizadas sin tener en cuenta el dominio de la aplicación a desarrollar. Una acción hace referencia a un conjunto de tareas de las cuales se obtiene un avance relevante de trabajo. Una tarea toma como eje principal un objetivo de un tamaño adecuado, el cual tiene ser claro y no debe tener ambigüedades [15].

Un proceso ayuda a saber qué se debe hacer, quién lo debe hacer y en qué momento se debe realizar. Las actividades que se realizan no deben ser tomadas aleatoriamente, sino con el orden respectivo que menciona el proceso de desarrollo a seguir.

En la literatura existente se pueden encontrar diferentes propuestas sobre procesos de desarrollo de software, las cuales cuentan con distintas actividades. Existen 4 actividades que todo proceso de desarrollo de software debe tener [16]:

- **Especificación del software:** Se define la parte funcional y las restricciones que debe tener el producto en cuanto a su operación.
- **Desarrollo del software:** Se realiza la creación del diseño y se construye el software, teniendo en cuenta la especificación inicial.
- **Validación:** El software debe validarse, para garantizar que el software cumple con las expectativas del cliente.
- **Evolución:** El software debe evolucionar para satisfacer las necesidades del cliente y las exigencias que puedan surgir en el mercado.

2.1.3. Aplicación web

Las aplicaciones web son desarrolladas de tal manera que se puedan ejecutar en un navegador web. Al día de hoy las aplicaciones web no se limitan a ofrecer funcionalidades solo a los usuarios, al contrario la complejidad de estas aplicaciones web a crecido cada día mas, ofreciendo nuevas características como por ejemplo integración con bases de datos entre otras [15].

La gran mayoría de aplicaciones web presentan los siguientes atributos [15]:

- **Uso intensivo de redes:** Las aplicaciones web funcionan mediante una conexión a la red, debido a que su estructura se distribuye en diferentes lugares y debe poderse acceder desde casi cualquier parte del mundo.
- **Concurrencia:** Al estar en la red, la aplicación web puede ser accedida por muchos usuarios al mismo tiempo.

- **Carga impredecible:** No se puede predecir cuántos usuarios van a utilizar la aplicación web en un día determinado, este número puede variar de un día para otro.
- **Rendimiento:** El tiempo de respuesta de una aplicación web a un usuario debe ser mínima, debido a que si la respuesta se tarda demasiado, el usuario estará insatisfecho.
- **Disponibilidad:** La disponibilidad en el caso de algunas aplicaciones web debe ser de 24 horas, debido a que pueden ser accedida en cualquier momento y desde cualquier lugar del mundo.
- **Orientada a los datos:** Considerada como principal, debido a que las aplicaciones web recogen y visualiza contenido.
- **Contenido sensible:** La organización de la información y la calidad de la misma juegan un papel fundamental en la calidad de una aplicación web.
- **Evolución continua:** Es una diferencia muy marcada entre aplicaciones convencionales y las aplicaciones web, ya que estas últimas cambian muy rápido debido a las necesidades que surgen.
- **Inmediatez:** Las aplicaciones web necesitan estar en funcionamiento lo más pronto posible.
- **Seguridad:** Este elemento es muy importante, debido a que una aplicación web puede tener contenido sensible, el cual debe ser solo accedido por personas autorizadas.
- **Estética:** No en todas las aplicaciones web es un factor importante, pero cuando se tratan de aplicaciones para comercializar o vender productos, la estética se convierte en algo indispensable.

2.1.4. MiPymes

La clasificación existente que permite definir si una empresa es micro, pequeña o mediana depende del país en el cual se realice, no todos los países define de la misma forma este tipo de empresas [17]. Cada país tiene en cuenta factores como el conocimiento aplicado, necesidades, intereses que han surgido, para así dar una clasificación adecuada a una empresa.

No existe algo único que mida el tamaño de una empresa, al contrario de esto, muchas veces se tienen en cuenta diferentes variables como por ejemplo:

- El número de trabajadores que emplean.
- Tipo de producto.
- Tamaño de mercado.
- Inversión en bienes de producción por persona ocupada.

- El volumen de producción y de ventas.
- Valor de producción.
- Trabajo personal de trabajadores o socios.
- Ubicación o localización.
- Nivel de tecnología de producción.
- Orientación de mercados.
- El valor del capital invertido.
- Consumo de energía.

Las MiPymes son muy importantes en el mercado, ya que son una gran fuente de empleo, debido al tamaño de la empresa pueden trabajar con menor capital. Esto sería considerado como un arma de doble filo, por un lado el tener un menor capital, no les da la capacidad de realizar grandes inversiones en tecnología, y así mismo contratar personal con características especiales para determinados trabajos, pero por otro lado es más fácil sostener una empresa que no necesite grandes cantidades de capital para trabajar [17].

2.1.5. Enfoques Ágiles

Los enfoques ágiles se adaptan a los cambios teniendo en cuenta que en cualquier momento pueden suceder, se aconsejan ciclos de entrega cortos, hacer análisis cada que sea necesario y comunicación constante con el cliente [18].

Para los métodos ágiles es de gran importancia establecer un canal de comunicación entre clientes y desarrolladores, entregar el producto dentro de las restricciones establecidas de tiempo, presupuesto y generar la mínima documentación posible(Solo la documentación necesaria) [19].

Las características claves de los métodos ágiles son:

- Recopilación continua de requisitos
- Frecuente comunicación cara a cara
- Refactorización
- Integración continua
- Comentarios anticipados de los clientes expertos
- Documentación mínima.

Los métodos ágiles más utilizados hasta el momento son: XP y Scrum [19].

Las metodologías ágiles cubren una amplia cantidad de proyectos de desarrollo de software con diferentes características. Estas metodologías son pensadas

principalmente para equipos de desarrollo pequeños, donde existen plazos cortos, requisitos cambiantes y debe existir la comunicación constante [20].

2.2. Estado del arte

En esta sección se muestra el proceso definido para realizar la revisión sistemática del estado del arte actual. Además se muestran los resultados obtenidos luego de realizar esta revisión.

2.2.1. Pregunta de investigación

La revisión sistemática surge bajo la necesidad de responder la siguiente pregunta:

¿Es posible definir un proceso liviano que soporte el desarrollo de aplicaciones web para MiPyMEs, teniendo en cuenta características de los procesos de ingeniería web y aplicaciones web actuales?

2.2.2. Cadenas de búsqueda

Las cadenas de búsqueda utilizadas para la revisión sistemática realizada son:

- Methodology OR process AND web AND application AND development AND NOT mobile.
- Strategy OR approach AND web AND application AND development AND NOT mobile AND NOT cost AND web services.

2.2.3. Recursos literarios

Para ingresar las anteriores cadenas de búsqueda se utilizaron las siguientes fuentes de búsqueda.

- Scopus
- ScienceDirect
- IEEE

2.2.4. Proceso de selección de estudios

El proceso de selección de los estudios se realizó teniendo en cuenta la importancia de los artículos basado en el título, resumen y conclusiones. Para la selección de los estudios más relevantes se usaron los criterios de inclusión (CI) y criterios de exclusión (CE) definidos a continuación.

2.2.4.1. Criterios de inclusión y exclusión

CI1. El artículo propone modelos o procesos o marcos para el desarrollo de aplicaciones web en micro, pequeñas o medianas empresas.

CI2. El artículo hace referencia a buenas prácticas de desarrollo de aplicaciones web en micro, pequeñas o medianas empresas.

CI3. El artículo hace referencia propuestas ágiles para desarrollo de aplicaciones web en micro, pequeñas o medianas empresas.

CE1. El artículo hace referencia a desarrollo de aplicaciones web en grandes empresas.

2.2.4.2. Clasificación de los trabajos relacionados

Las propuestas elegidas de la búsqueda sistemática tienen una ventana de tiempo de 18 años (desde 1998 hasta el 2016). Aunque puede llegar a ser una ventana de tiempo muy amplia debido a que algunas propuestas pueden ser consideradas obsoletas, se han incluido propuestas de casi dos décadas, ya que abordan conceptos que en la actualidad tienen gran relevancia en el desarrollo de aplicaciones web. Conceptos como navegación de aplicaciones web, utilización de disciplinas de análisis, diseño e implementación, prácticas como creación de aplicaciones por medio de componentes, entre otros.

En la tabla 1 se muestra la clasificación por año de cada trabajo encontrado.

Tabla 1 . Clasificación por año de los artículos relacionados

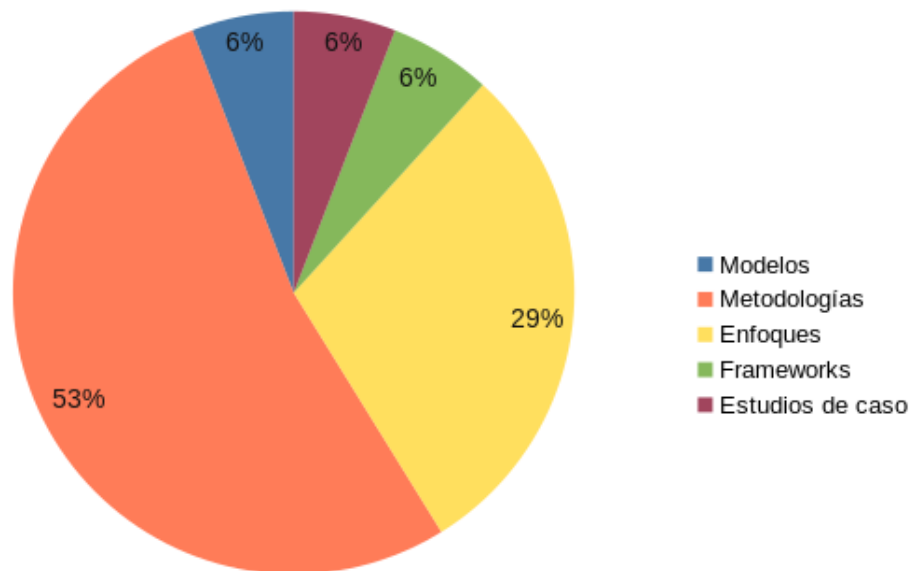
Año de publicación	Conteo	Referencia
1998	1	[21]
2000	1	[22]
2001	1	[23]
2003	1	[24]
2004	1	[25]
2005	1	[3]
2006	1	[26]
2007	1	[27]
2008	1	[28]

2009	1	[29]
2010	1	[30]
2011	2	[31,32]
2015	2	[33,34]
2016	2	[35,36]

2.2.4.3. Clasificación de artículos de acuerdo a su contenido

En la Figura 1 se presentan los artículos clasificados según el contenido.

Figura 1 . Artículos de acuerdo a contenido



2.2.5. Propuestas existentes

A continuación se indican las propuestas elegidas de la revisión sistemática, su clasificación y un breve resumen de cada una.

2.2.5.1. Trabajos relacionados que involucran modelos

2.2.5.1.1 Model2Roo: A Model Driven approach for web application development based on the eclipse modeling framework and Spring Roo

El principal aporte de Castrejón, Juan et al. [31] está en el proceso de administración del cambio asociado a aplicaciones web. Con este enfoque, los cambios no necesitan ser aplicados directamente sobre los artefactos de implementación del sistema. En vez de eso el modelo del sistema debe ser actualizado usando el EMF framework y scripts generados por String Roo, que a su vez tendrá los cambios de implementación apropiados. De esa forma se evita la falta de cumplimiento entre los documentos de diseño y los artefactos de implementación. Este modelo está ligado a la plataforma Eclipse.

El enfoque está centrado en el modelo de abstracción más que en los detalles de implementación. Se tienen en cuenta las etapas del ciclo de desarrollo sin profundizar en cada una de ellas ni proveyendo una metodología que guíe el proceso de desarrollo. No se tiene en cuenta el tamaño de la empresa.

2.2.5.2. Trabajos relacionados que involucran metodologías

2.2.5.2.1 Agile development of web application by supporting process execution and extended UML model

Lee, Wookjin et al. [3] presentan una metodología que proporciona varios aportes: I) Un proceso ágil, II) modelos y III) herramientas para el desarrollo de aplicaciones web. El proceso ágil se compone de un pequeño número de actividades en los que se aplican principios basados en pruebas, para la implementación, y basados en modelos, para la arquitectura. En cuanto a los modelos se definen un modelo navegacional y un modelo de comunicación de componentes.

Las herramientas que se indican son: I) Gestor de procesos y II) Modelador Web. El Gestor de procesos muestra los proyectos que ya se han terminado o que se encuentran en desarrollo. Esta herramienta guarda información del progreso del proyecto y ofrece pasos detallados de cada actividad. En cuanto al Modelador Web se utiliza ArgoUML, con la diferencia que se le han agregado unos plugins para extender su funcionalidad y poder generar modelado navegacional y modelado de comunicación de componentes.

Aunque se propone un proceso ágil, en el cual se especifican actividades a realizar, no se deja en claro el desarrollo de las fases de inicio, análisis, diseño, implementación, pruebas y despliegue. Además debido a que el proceso hace parte

de una metodología, se está sujeto a utilizar herramientas para el éxito de su aplicación en los proyectos.

2.2.5.2.2 Web development life cycle: a new methodology for developing web applications

French, Aaron [32] presenta una metodología para el desarrollo de aplicaciones web. Esta metodología es basada en metodologías ya existentes, pretende crear un proceso bien estructurado para la creación de aplicaciones web. Es un híbrido de dos metodologías conocidas como: ciclo de vida de desarrollo de sistemas y prototipos. Esta propuesta reducirá el tiempo de desarrollo, y mantendrá a los usuarios involucrados durante todo el ciclo de desarrollo.

La metodología divide el proceso de desarrollo en dos iteraciones distintas: La primera iteración es el desarrollo gráfico y la segunda iteración es la de desarrollo funcional. Cada iteración se compone de tres fases que son: I) Obtención de información, II) Análisis y III) Diseño. Por último se tienen dos fases adicionales que están fuera de la parte iterativa, estas fases son: implementación y mantenimiento. En la fase de implementación una vez se haya afinado el prototipo y terminado el componente, se pasan a un servidor de producción. En la fase de mantenimiento se pueden incluir cambios en las funcionalidades existentes, actualizaciones en los estilos de las vistas, u otros tipos de cambios que sean requeridos por los clientes.

La metodología considera partes importantes del desarrollo de una aplicación web como lo son: La obtención de requisitos, el diseño de prototipos, proceso iterativo, entre otros. Aunque abarca algunos factores importantes en la creación de aplicaciones web, no se tiene en cuenta cosas como el diseño navegacional, ni se enfatiza en casos de prueba. Además no cuenta con una fase de inicio del proyecto, la cual es importante al iniciar un proyecto web.

2.2.5.2.3 A component based methodology for web application development

Lee, Seung y Shirani, Ashraf [25] presentan una metodología para el análisis de requisitos y diseño de alto nivel de las aplicaciones web basadas en componentes. Esta metodología introduce clasificaciones de páginas y componentes, y un conjunto de etiquetas para implementar la semántica de los enlaces. La metodología propuesta se compone de dos partes, cada una con sus actividades a realizar. La primera parte es el análisis de los requisitos, la cual tiene las siguientes actividades: I) análisis de requisitos de alto nivel y II) análisis de requisitos de bajo nivel. En el análisis de requisitos de alto nivel, se definen los requisitos de una forma general. En el análisis de requisitos de bajo nivel se indican los requisitos más en detalle. La segunda parte es la especificación de componentes. En esta especificación se encuentran actividades como: I) especificación de renderizado, II) especificación de integración y III) especificación de la interfaz.

La metodología tiene en cuenta un aspecto muy importante en el desarrollo de todas las aplicaciones web, el análisis de requisitos. Además se hace mucho énfasis en la parte de los componentes que conforman las aplicaciones web, pero no se habla de un proceso que pueda ser utilizado de guía para el desarrollo de una aplicación web como por ejemplo, análisis, implementación, pruebas y mantenimiento.

2.2.5.2.4 Development of object-oriented analysis and design methodology for secure web applications

Arshiya, Mrs. Sunita [33] propone una metodología de análisis y diseño orientado a objetos enfatizada en la seguridad. La metodología proporciona consistencia en la seguridad a lo largo del ciclo de vida de una aplicación web, desde el análisis de requisitos hasta la implementación. Esta metodología está compuesta por: I) análisis de requisitos, II) análisis y diseño del sistema y por último III) implementación. En el análisis de requisitos se tienen las siguientes actividades: listado de requisitos, creación de los casos de uso, detallar el modelo de los casos de uso, y creación del modelo de casos de uso. En el análisis y diseño del sistema se tienen las siguientes actividades: análisis del texto de los casos de uso, creación del diagrama de clases y diagrama de clases detallado. En la implementación se tiene la actividad de control de acceso basado en roles en java EE.

Aunque se tiene en cuenta elementos muy importantes en el desarrollo de una aplicación web como el análisis de requisitos y la seguridad, presenta dependencia a una tecnología, que en este caso es java. Por lo tanto el éxito de la metodología para la creación de una aplicación web depende del desarrollo en java. Además no se habla de la realización de prototipos, los cuales son muy importantes al momento de un desarrollo web.

2.2.5.2.5 Mockup-drive fast-prototyping methodology for web application Development

Zhang, Jia y Chung, Jen-Yao [24] proponen una metodología de prototipado rápido para el desarrollo de aplicaciones web. Los objetivos de esta metodología son: I) acelerar el desarrollo de aplicaciones web, mediante el aprovechamiento de técnicas automáticas en cuanto a la generación de programas, II) incluir al cliente en el proceso de desarrollo, y por último, III) facilitar el mantenimiento del software. La metodología definida en este artículo se compone de las siguientes actividades: I) descomposición, II) generador de menú, III) diseño de páginas web, IV) Diseño del modelo de datos, V) sistema generador de llamada, VI) mapa de páginas, VII) reunir los comentarios de los usuarios y volver al tercer paso para realizar modificaciones, y VIII) agregar lógica de negocio a los métodos de servicio, para obtener un sistema en ejecución.

En la descomposición se divide el sistema en módulos y submódulos, para luego organizar los módulos en una jerarquía de ítems para menús. En el generador de

menú se genera una jerarquía de menú, y se envía al cliente para su revisión. En el diseño de páginas web se diseña las páginas que contendrán los ítems de cada menú. En el diseño del modelo de datos se realiza la creación del modelo de datos basado en las páginas especificadas. En el sistema generador de llamada se utiliza el generador de código para generar todo el código asociado a cada página identificada en el sistema. En el mapa de páginas se realiza la asignación de páginas a cada ítem de menú. En el séptimo paso se reúnen los comentarios de los usuarios y se realiza una iteración al tercer paso para realizar modificaciones, y en el último paso, se agrega la lógica correspondiente a cada método de servicio, para así tener un sistema, el cual se pueda llevar a ejecución.

Aunque la metodología trata un tema importante en el desarrollo de aplicaciones web como lo es el prototipado, para aplicar la metodología se deben tener un documento previo que contenga los requisitos de la aplicación web a desarrollar. Por lo tanto no se incluye el paso de elicitación de requisitos que es una parte fundamental de todo desarrollo. Además la metodología está centrada en la creación de prototipos, no se tienen en cuenta partes de un proceso de desarrollo como pruebas, despliegue, mantenimiento, entre otros.

2.2.5.2.6 A Proposed methodology for web Development

Howcroft, Debra y Carroll, John [22] presentan una metodología basada en la combinación de la literatura disponible sobre metodologías propuestas para el desarrollo de aplicaciones web, y la experiencia de uno de los autores, el cual está involucrado activamente en el desarrollo de sitios web. Se proponen cuatro etapas: análisis, diseño, generación e implementación, a las que se les definen los pasos a seguir para su ejecución y los documentos que se generan en cada etapa.

En la fase de análisis se realizan tres pasos: el desarrollo de una estrategia web, definición de objetivos y el análisis de objetivos. La estrategia es desarrollada en un documento de planeación de estrategia. En la definición de los objetivos se toma como entrada el documento de planeación de estrategia, con este documento se definen los objetivos del sitio. En el análisis, se analizan los objetivos obtenidos en el paso anterior. De esta fase se obtiene un documento que contiene los objetivos especificados del sitio.

En la fase de diseño se realizan dos pasos: Diseño y diseño de pruebas. El diseño puede involucrar, el diseño de una base de datos, diseño de imágenes, animaciones, colores, entre otros. En el diseño de las pruebas se define el cómo se va a validar el correcto funcionamiento y calidad del sitio. De esta fase se obtiene un documento de diseño de cada paso.

En la fase de generación se realizan 4 pasos: selección de recursos, revisión de diseño, generación de código e instalación y pruebas. En la selección de recursos, se seleccionan los recursos para el desarrollo del sitio como: hardware, software, links de comunicaciones, entre otros. En la revisión del diseño se toma el documento

de diseño de la fase de diseño y se compara con los recursos disponibles. En la generación de código e instalación se ve la generación de todo el software conectado con el sitio y su instalación en servidores web. En las pruebas se realiza la evaluación del sitio web para evaluar la calidad.

En la fase de implementación se realizan 3 pasos: implementación, mantenimiento, revisión de objetivos. En la implementación se debe registrar el sitio web con los principales motores de búsqueda. En el mantenimiento se monitorea regularmente el sitio para asegurar que la información y links están disponibles. En la revisión de objetivos se revisan los objetivos que han sido logrados, los que no han sido logrados se escriben en una lista de deseos.

Aunque presenta una serie de pasos a seguir para la ejecución de cada etapa, esta metodología obvia algunos aspectos muy importantes en el desarrollo de aplicaciones web, como lo es la creación de prototipos iniciales para dar capturar y validar requisitos.

2.2.5.2.7 Construyendo aplicaciones web con una metodología de diseño orientado a objetos.

Silva, Dario y Bárbara, Mercerat [30] presentan una metodología para la construcción de aplicaciones web basada en 4 etapas, las cuales son: I) Diseño conceptual II) Diseño Navegacional III) diseño de la interfaz abstracta IV) Implementación.

En el diseño conceptual se construye un esquema conceptual, el cual contiene los objetos del dominio. En el diseño navegacional se toma como entrada el modelo conceptual y se construye un esquema para la navegación de la aplicación. En el diseño de la interfaz abstracta se realiza la organización de los elementos de navegación, dependiendo del usuario y tipo de aplicación que se va a construir, y por último se implementa el diseño realizado.

Esta metodología propone dedicar un tiempo importante en las fases previas a la implementación, para a largo plazo poder reducir los tiempos de desarrollo, mediante la reusabilidad de diseño y el mantenimiento.

Aunque la metodología tiene 4 etapas para el desarrollo de una aplicación web, no se tiene en cuenta el desarrollo iterativo e incremental, con lo cual se hace muy difícil soportar cambios en el desarrollo de la aplicación web. Además no enfatiza en la importancia de tener los objetivos claros del producto al inicio del proyecto, llevando a un alto riesgo de fracaso en el desarrollo web.

2.2.5.2.8 Las metodologías de desarrollo y mejora de la calidad de las aplicaciones web.

Cáceres, Paloma y Marcos, Esperanza [23] proponen una metodología llamada MiDAS, la cual se basa en un modelo iterativo e incremental. En cuanto a lo

incremental se identifican cuatro pasos generales que son: I) Requisitos del Sistema, II) Interfaz de Usuario, III) Datos y IV) Funcionalidad.

En los Requisitos del Sistema, se identifican las funcionalidades requeridas de la aplicación. En la Interfaz de Usuario, se utilizan los requisitos del sistema identificados para crear interfaces gráficas. En el paso de Datos se incorpora la estructura interna que tendrá la aplicación web y en el paso de Funcionalidad se incorpora el funcionamiento que el sistema web poseerá.

Para proponer esta metodología, primero se realiza una caracterización de las aplicaciones web, para poder tener en cuenta las diferentes aplicaciones web existentes. Esta caracterización se compone de 4 fases: I) Estructura de la información, II) Lógica de la aplicación, III) Navegabilidad y IV) Presentación.

En la estructura de la información se realiza la organización de los datos del sistema. En la lógica de la aplicación se tiene en cuenta la funcionalidad de carga de datos y generación de información para la salida al usuario. En la navegabilidad se realiza la estructuración y distribución de la información que se va a presentar al usuario, y en la presentación, se toma en cuenta cómo se va a mostrar al usuario la información proporcionada por el sistema.

Aunque en la metodología se tiene en cuenta 4 pasos en los cuales será posible identificar los nuevos requisitos que puedan surgir, no se tiene en cuenta un paso importante como son las pruebas del sistema.

2.2.5.2.9 Scrum y Xp desde las trincheras

En este libro [27] se proporciona una visión muy interesante del uso de Scrum en una organización. Aunque el autor afirma que no es el único libro existente, da unos lineamientos que están fundamentados en experiencias adquiridas mediante la utilización de Scrum. Toda la experiencia obtenida y los datos presentados, se han logrado variando distintos elementos que pueden ser ajustables en un marco de trabajo como Scrum, para obtener el mejor resultado esperado.

La mayor parte del libro habla sobre la utilización de Scrum, aunque también se tienen en cuenta las buenas prácticas realizadas en la metodología de Xp (Se combina Scrum con Xp). Se describe desde cómo realizar la planificación de un sprint, hasta la retrospectiva de un sprint, pasando por etapas como: como hacer pila de sprint, como distribuir la sala del equipo, como hacer Scrum diarios, entre otras.

En cuanto a la combinación de Scrum y Xp, se dan a conocer prácticas como: I) Programación por parejas, II) Desarrollo guiado por pruebas, III) Diseño incremental, IV) Integración continua, VI) Propiedad colectiva del código, VII) Espacio informativo, VIII) Estandarización de código y por último IX) Ritmo sostenible.

2.2.5.3. Trabajos relacionados que involucran enfoques

2.2.5.3.1 Web applications development by formal refinement approach

Charaf, Meftah y Kazar, Okba [34] proponen un enfoque formal para el desarrollo de aplicaciones web seguras, teniendo en cuenta el lado cliente y el lado de servidor web desde especificaciones formales. Para ellos, se tienen tres fases mediante las cuales se describe la aplicación usando notaciones simbólicas para ser traducidas a especificaciones formales. Éstas especificaciones formales se refina mediante algunas reglas para lograr un especificación final cercana a la implementación de un lenguaje elegido. Se proveen las herramientas y procesos para construir y analizar dicho enfoque, el cual permite la automatización de las fases de especificación, refinamiento y generación de código automático.

Este enfoque no menciona una metodología que permita guiar el proceso de desarrollo de las aplicaciones web. Además, se centra en aplicaciones basadas en Arquitecturas orientadas a servicios (SOA).

2.2.5.3.2 Test driven development of web applications: a lightweight approach

Clerissi, Diego et al. [35] presentan un enfoque para el desarrollo de aplicaciones web impulsado por pruebas. Este enfoque se compone de dos partes, cada parte contiene sus actividades. La primera parte está compuesta de las actividades que apoyan el desarrollo de la aplicación web: I) Análisis de requerimientos y II) Desarrollo de mockups, la segunda parte hace referencia al desarrollo del conjunto de pruebas de la aplicación web, y se compone de las siguientes actividades: I) Desarrollo del conjunto de pruebas de aceptación, II) Desarrollo de la aplicación web, III) Mejora de la mantenibilidad del conjunto de pruebas, IV) Extensión del conjunto de pruebas y V) Mejora de la robustez.

Este enfoque no está limitado a ser desarrollado en contextos ágiles donde se utilizan historias de usuario para definir los requisitos del usuario. Por el contrario da la posibilidad de trabajar con especificaciones más formales, en las cuales se utilizan casos de uso. Con este enfoque a partir de los requisitos textuales y los mockups se pueden generar scripts de pruebas web funcionales.

En esta propuesta aunque se describen unos pasos para el desarrollo de aplicaciones web, se enfoca más que todo en las pruebas y no se da claridad en qué artefactos aparte de las pruebas son importantes, y tampoco se habla del mantenimiento de la aplicación web.

2.2.5.3.3 A page-centric approach to web application development

Teraoka, Teruhiko et al. [29] proponen un enfoque para el desarrollo de aplicaciones web centrado en la página, en la que una aplicación web es modelada como una máquina de estados de UML que tiene en cuenta el estado de una página web. Este enfoque se compone de las siguientes partes: I) Definición de transiciones

de pantallas utilizando máquinas de estado, II) Definición de las variables de documento y las variables de aplicación, y por último III) definición de enlaces y acciones. En esta última parte, además de la definición de enlaces y acciones, también se ejecutan las acciones y los enlaces, se realiza una actualización de transiciones internas y la visualización del contenido en pantalla.

En la definición de transiciones de pantallas utilizando máquinas de estado, se realiza un modelo para representar la aplicación, cada vista de la aplicación se modela como un estado en la máquina de estados. En la definición de las variables de documento y las variables de la aplicación, las variables asociadas con la entrada y salida en una vista de la aplicación, son definidas como variables de documento. Las variables de entrada pueden ser input, select, entre otras y las de salida span, var o div. Las variables de aplicación son las que no se definen como variables de documento: variables globales, variables de sesión, variables de solicitud y variables locales. En la definición de enlaces y acciones se definen los enlaces entre variables de documento y variables de aplicación existentes, además se tienen en cuenta que acciones se necesitan para el funcionamiento de la aplicación web.

El enfoque hace un gran énfasis en el desarrollo de páginas web, lo cual ayuda mucho a la usabilidad, pero por otro lado no se dan pautas claras para la construcción completa de la aplicación web.

2.2.5.3.4 An object oriented approach to web-based applications design.

Schwabe, Daniel y Rossi, Gustavo [21] proponen un enfoque orientado a objetos para realizar la creación de aplicaciones hipermedia. Para este enfoque se define un proceso conformado por 4 actividades que son: I) Diseño conceptual, II) Diseño navegacional, III) Diseño de interfaces abstractas, IV) Implementación.

En el diseño conceptual se realiza la creación de un esquema conceptual que representa objetos, sus relaciones en el dominio de la aplicación. En el diseño navegacional se construye un modelo navegacional como una vista sobre el modelo conceptual. En el diseño de interfaces abstractas se define como los diferentes objetos de navegación serán visualizados en las vistas de la aplicación web, y qué objetos serán activados en la navegación. En la implementación se realiza la creación de la aplicación web tomando como entradas los modelos obtenidos en las anteriores fases.

Aunque define algunos pasos para el desarrollo de aplicaciones web, no hace énfasis en asegurar atributos de calidad inherentes a las aplicaciones web.

2.2.5.3.5 A rule based approach to web based application development

Tammet, Tanel et al. [26] presentan un nuevo enfoque basado en reglas para el desarrollo de sistemas de información basados en web. Este enfoque permite la separación de la lógica de negocio de las interfaces o lógica de aplicación, cuando se crean aplicaciones de software complejas basadas en web. Se presenta la

tecnología Rql, usada para el desarrollo de sistemas de información basados en la web, cuya base está en reglas de negocio.

La Tecnología Rql, básicamente proporciona una capa media entre una base de datos y las interfaces. La idea que se tiene con Rql es escribir la lógica de negocio para hacerla parte de la capa media, con esto se evita que el código de la interfaz se mezcle con la capa intermedia. En este caso la interfaz no tendría código de consultas SQL o sistema de reglas.

La tecnología Rql se compone de tres partes:

- Servidor Xstone: Es el servidor de aplicaciones, con el cual se pueden crear aplicaciones de tres capas.
- Solucionador de reglas RqlGandalf: Encargado de responder consultas de datos, los cuales se basan sobre reglas lógicas que son ingresadas por el usuario o programador.
- Interfaz basada en el navegador xswc que crea un kit de herramientas: Se utiliza para la creación de interfaces de usuario tomando en cuenta el navegador para Xstone.

No se tiene en cuenta el tamaño de las organizaciones y está dirigida a un conjunto de dominios específicos. A su vez, no se menciona una metodología que dirija el proceso en cada etapa del proyecto.

2.2.5.4. Trabajos relacionados que involucran frameworks

2.2.5.4.1 A theoretical agile process framework for web applications development in small software firms

Altarawneh, Haroon y Shiekh, Asim [28] proponen un marco de proceso software para el desarrollo de aplicaciones web en pequeñas empresas. Este marco se compone de una serie de elementos como son: I) Iniciar con proyectos web pequeños, II) Adoptar el modelo de proceso XP modificado, III) Aplicar XPMM, IV) Educación y entrenamiento, V) Evaluación interna y revisiones formales claras, y VI) Evaluación externa. Además todos estos procesos deben ser implementados con las mejores prácticas de la ingeniería web.

Al iniciar con proyectos pequeños es ideal planear todas las fases al inicio del proyecto. Al adoptar el modelo de proceso XP modificado se ha propuesto la fase ligera de gestión de proyectos. En la aplicación del XPMM se debe tener en cuenta que cuenta con 4 niveles, este modelo debe ayudar a distinguir entre proyectos de XP reales y proyectos de pseudo-XP. En cuanto a la educación y entrenamiento, los desarrolladores y administradores deben tener un buen conocimiento y estar muy bien capacitados en ingeniería web. La evaluación interna y revisiones formales se deben realizar para comprobar que el equipo ha entendido como implementar el modelo XP modificado y el XPMM. Y por último la evaluación externa es muy

importante para poder garantizar la calidad, debido a que puede ser realizada por una compañía de aseguramiento de la calidad.

Este framework es un framework teórico que cubre solamente pequeñas empresas y se centra más en el proceso que en las características propias de las aplicaciones web.

2.2.5.5. Trabajos relacionados que involucran estudios de caso

2.2.5.6. Desarrollo e implementación de un sistema web para generar valor en una pyme aplicando una metodología ágil

Castillo, Pedro [36] presenta el desarrollo de un estudio de caso en una empresa categorizada como pequeña empresa. El propósito del proyecto es demostrar que el uso de tecnologías apoya el crecimiento de las Pymes, pues mediante la elaboración de un sitio web se mejoran procesos internos de la empresa teniendo como resultado mayores ventas e ingresos. Uno de los puntos que trata este estudio de caso es el uso de XP como metodología de desarrollo, la cual brinda buenos resultados para proyectos de desarrollo web, debido a que se genera valor con cada entregable al final de cada iteración.

Sin embargo, si bien se hace uso de una metodología ágil, el mismo autor plantea el buscar metodologías de desarrollo que se adapten a cada caso particular y a proyectos web de mayor magnitud.

2.2.6. Caracterización de los trabajos relacionados

En la tabla 2 se clasifican los trabajos analizados, de acuerdo a la profundidad en que abordan cada factor a tener en cuenta en el desarrollo de aplicaciones web. Se han diseñado los siguientes acrónimos:

A: Ampliamente abordado, **M:** Medianamente abordado, **B:** Aproximación de alto nivel.

Tabla 2 . Caracterización de los trabajos relacionados

Referencia Artículo	Requisitos de un proceso web						Atributos de calidad				Tipo empresa
	Ciclo de desarrollo corto	Requisitos cambiantes	Fechas establecidas	Desarrollo en Paralelo	Reúso	Adaptación	Seguridad	Usabilidad	Desempeño	Mantenibilidad	
[28]	A	A	A	A	A	M		M			M

[34]	B	B				B					
[26]					A						
[31]					B					B	
[32]									B	B	
[35]		B			B					M	
[25]	B	B	B		B			B		M	
[29]					B	B					
[33]								M	A	M	
[24]					M		A				
[36]	M	A	M		M			A		M	
[30]	A	A	M				M		M	M	
[12]					M		B	A	M	M	
[23]					M		B	M		M	
[21]	A	A	A					A		M	
[22]	A	A				A	M	A	A	A	M
[27]	A	A	A			M	B		B	B	

2.2.7. Aportes

En la revisión y el análisis del estado del arte, no se evidencio un proceso de desarrollo web liviano que satisfaga algunos de los requisitos de un proceso de ingeniería web, como son: ciclo de desarrollo corto, requisitos cambiantes, liberaciones con fechas establecidas, desarrollo en paralelo, reúso e integración de componentes, adaptación del proceso al nivel de complejidad. Con este proyecto se pueden identificar los siguientes aportes en:

Investigación:

- Revisión de la literatura de las metodologías y procesos livianos para el desarrollo de aplicaciones web en MiPyMEs.
- Recomendaciones a la comunidad académica e industria en la implementación del proceso definido y extensión de los resultados obtenidos.

Innovación:

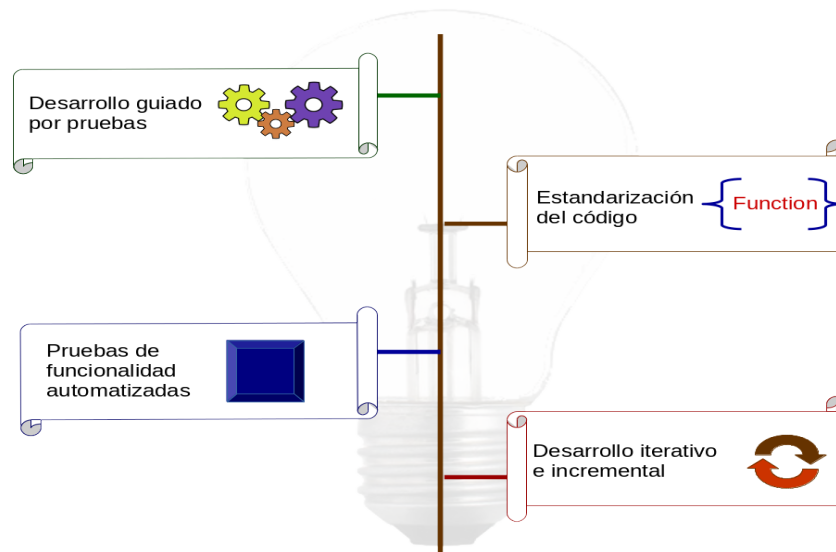
- Con el proyecto, las micro, pequeñas y medianas empresas tendrán un proceso liviano de desarrollo web que satisfaga los requisitos de un proceso de ingeniería web, las características propias de las aplicaciones web y los atributos de calidad de seguridad, usabilidad, desempeño y mantenibilidad.

2.3. Caracterización del Proceso

Una vez obtenida la literatura del estado del arte, se hace un análisis para identificar los aportes que realiza cada una de estas propuestas. El análisis de la literatura arroja elementos (desarrollo guiado por pruebas, estandarización del código, pruebas de funcionalidad automatizadas, desarrollo iterativo e incremental) que son de gran importancia al momento del desarrollo de una aplicación web, terminado el análisis, se presenta a un experto (director del trabajo de grado) los conceptos encontrados en la literatura. El experto los analiza y da la aprobación para establecerlos como pilares del proceso.

En la Figura 2 se muestran los elementos estructurales del proceso.

Figura 2. Caracterización del proceso.



En la tabla 3 se presentan los elementos estructurales del proceso y la justificación del por que se tuvieron en cuenta para el desarrollo de la propuesta.

Tabla 3 . Caracterización del proceso

Elemento	Justificación
Desarrollo guiado por pruebas(TDD)	Este elemento se tiene en cuenta debido a que se obtienen rápidamente ciclos de desarrollo-compilación-pruebas. Además da seguridad a los desarrolladores para que puedan realizar cambios al código cada vez que sea necesario, manteniendo la calidad incluso si el sistema crece cada vez más [27].
Estandarización del código(Buenas prácticas de desarrollo)	Este elemento se tiene en cuenta debido a que muchos programadores utilizan su propio estilo de programación, esto puede llevar a que en algunos casos haya inconsistencia en el diseño del sistema y que el código se vuelva muy difícil de leer [27].
Pruebas de funcionalidad automatizadas	Este elemento se tiene en cuenta ya que hacer pruebas manualmente puede resultar poco eficiente debido al tiempo invertido en su realización, en cambio mediante la automatización se puede ejecutar un gran número de pruebas en cualquier momento y de manera muy eficiente. Debido a esto, la automatización de pruebas es un elemento muy importante a tener en cuenta en el desarrollo de aplicaciones web [37].
Desarrollo iterativo e incremental	La parte iterativa e incremental en un desarrollo es algo fundamental, debido a que se pueden obtener versiones sucesivas de una aplicación, las cuales van incorporando nuevas funcionalidades y además permiten la inclusión de nuevos requisitos del sistema [38].

Capítulo 3. Definición del proceso DEWE

En este capítulo se presenta el proceso propuesto para el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas, el cual tiene en cuenta características propias de los desarrollos web. Para la construcción de la propuesta se han tomado como base, contenido del capítulo dos, sugerencias de docentes y aportes realizados por los participantes del grupo focal disponible en el capítulo 4.

A continuación se presenta la versión final del proceso propuesto.

3.1. Introducción

DEWE (Desarrollo Web Empresarial) es un proceso que permite a micro, pequeñas y medianas empresas construir aplicaciones web teniendo en cuenta características propias de este tipo de productos. El proceso está enfocado en la construcción de software, mas no en la gestión, la gestión es realizada por Scrum. DEWE contiene roles, disciplinas, actividades, tareas, entradas y salidas, que en conjunto permiten obtener funcionalidades de una aplicación web. Para la definición de las fases (análisis, diseño, implementación, despliegue y pruebas) del proceso se ha utilizado como guía el patrón de procesos propuesto por Competisoft [39].

3.2. Creación y evaluación del proceso

Este proceso nace como una alternativa para el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas. Para la construcción del proceso se tiene en cuenta el estado del arte presentado en el Capítulo 2, la asesoría de docentes, comentarios y observaciones realizadas por personas de empresas que tienen conocimiento en ingeniería de software, desarrollo web y metodologías ágiles.

Para dar un nivel de confiabilidad adecuado del proceso se tuvieron en cuenta los siguientes pasos:

- Estado del arte
- Asesoría de profesores
- Creación de la primera versión del proceso
- Evaluación del proceso en un grupo focal (se cuenta con expertos en desarrollo de aplicaciones web, ingeniería del software y metodologías ágiles).
- Generar nueva versión del proceso

3.3. Alcance del proceso

Este proceso está dirigido a micro, pequeñas y medianas empresas, las cuales desean tener una guía de construcción de aplicaciones web, que les permita considerar los diferentes factores a tener en cuenta al momento del desarrollo de un proyecto web. Esta propuesta en conjunto con la metodología Scrum, guía a las empresas mediante una serie de fases que permiten plasmar en funcionalidades los requisitos obtenidos.

3.4. Principios ágiles

- Desarrollo iterativo: Las disciplinas de análisis, diseño, implementación y despliegue se realizan de forma iterativa durante la construcción de todo el producto. Ya que el proceso se apoya en Scrum, cada sprint se convierte en una iteración que realiza el proceso para liberar funcionalidades que aportan al crecimiento de la aplicación web.
- Entregar software funcional frecuentemente: El proceso realiza la construcción de las funcionalidades que posee un sprint, teniendo en cuenta que un sprint tiene un número limitado de historias de usuario. Las funcionalidades construidas serán liberadas cada cierto tiempo.
- Excelencia técnica y buen diseño: Existen roles bien definidos y con responsabilidades asignadas. Cada rol tiene conocimiento de las actividades a realizar y los artefactos de los cuales es responsable.

No se obtiene un diseño completo de la aplicación desde el inicio, por el contrario los artefactos son construidos a medida que pasa cada iteración, con lo cual al final del proyecto se tendrá un diseño completo del producto web.

3.5. Plantilla para la definición del proceso

Para definir el proceso, se toma como referencia el patrón de procesos propuesto por COMPETISOFT [39]. Este patrón ha sido modificado para ajustarse a las necesidades del proceso propuesto. En la tabla 4 se presenta la plantilla para la definición del proceso.

Tabla 4. Plantilla para la definición del proceso

Identificador	Acrónimo utilizado para la identificación única del proceso.	
Proceso	Nombre del proceso, precedido por el acrónimo establecido en la definición de los elementos de la estructura del modelo de procesos.	
Categoría	Nombre de la categoría a la que pertenece el proceso.	
Propósito	Objetivos generales de las actividades y productos que componen el flujo de trabajo del proceso.	
Descripción	Descripción general de las actividades y productos que componen el flujo de trabajo del proceso.	
Objetivos	Objetivos específicos cuya finalidad es asegurar el cumplimiento del propósito del proceso. Los objetivos se identifican como O1, O2, etc.	
Responsabilidad y autoridad	Responsable es el rol principal responsable por la ejecución del proceso. Autoridad es el responsable por validar la ejecución del proceso y el cumplimiento del propósito.	
Procesos relacionados	Nombre de procesos relacionados.	
Roles involucrados y competencias		
Abreviatura	Rol	Competencias
Abreviatura del rol	Nombre del rol	Descripción de las responsabilidades de cada rol.
Nombre de la fase		
Entradas	Nombre de los artefactos que se toman como entrada para la actividad.	
Rol Acrónimo del rol encargado de realizar la(s) tarea(s).	Actividades Se agrega la actividad y sus correspondientes tareas.	
Salidas	Descripción de los artefactos producidos en la actividad.	

Diagrama de flujo: Diagrama en el cual se puede visualizar el flujo que tiene la actividad.

3.6. Términos genéricos

En la tabla 5 se da una corta descripción de los términos utilizados para la definición del proceso.

Tabla 5. Términos del genéricos.

Término	Descripción
Proceso	“Un proceso es un conjunto de actividades, acciones y tareas que se ejecutan cuando va a crearse algún producto del trabajo” [15].
Actividad	Una actividad siempre tiene como fin cumplir con un propósito de un tamaño considerado, y es realizada sin importar el ámbito del proyecto en la cual se vaya a desarrollar [15].
Tarea	A diferencia de la actividad, la tarea se centra en un propósito pequeño, el cual está totalmente claro [15].
Rol	“Describe un conjunto o grupo de responsabilidades, deberes y habilidades requeridas para realizar un actividad” [40].
Herramienta	Producto software que automatizan la ejecución de ciertas actividades.

3.7. Identificación de roles

Para la definición de un proceso es necesaria la existencia de una serie de roles, para la distribución de tareas de acuerdo a las habilidades que posea cada participante del equipo de proyecto. Durante la creación del proceso se identificaron diferentes roles que aportan a las fases que conforman la propuesta para el desarrollo de aplicaciones web.

Los roles que se presentan en el proceso no necesariamente deben ser desarrollados cada uno por una persona. Para aquellos equipos de trabajo o empresas que no pueden tener una persona por rol, se aconseja que una persona pueda asumir más de un rol.

3.8. Visión general de la propuesta

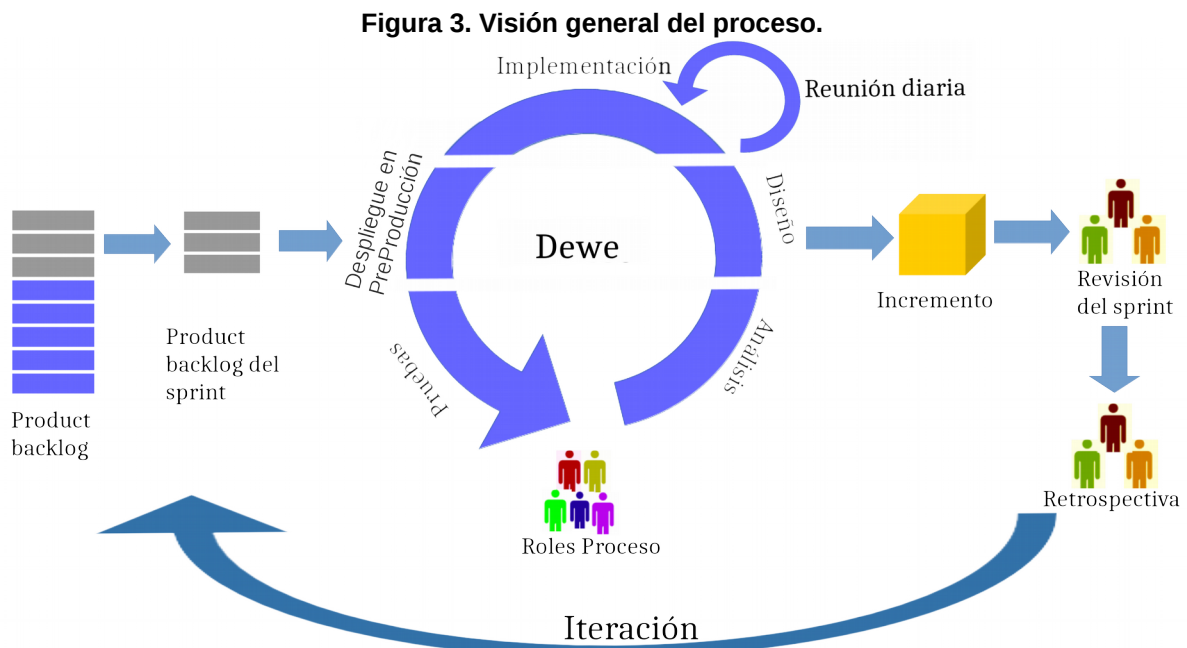
La propuesta ha sido diseñada para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas. El proceso propuesto se apoya en la metodología Scrum para cumplir con el objetivo de creación de una aplicación web.

Scrum es una metodología de gestión de proyectos, mas no de desarrollo. Scrum especifica como gestionar un proyecto, que lineamientos seguir, pero no especifica como desarrollar los requerimientos de una aplicación web.

Scrum ha sido elegida para ser la encargada de la gestión del proyecto en el proceso, debido a que es una metodología de gestión de proyectos que ha ganado mucha popularidad en los últimos años [41]. Cada vez son mas las empresas que optan por esta metodología como apoyo en la construcción de software, el gran incremento de certificaciones para Scrum Masters confirma que la adopción de Scrum va creciendo con el paso del tiempo. Empresas como Apple, General Electric y equipos como Google AdWords utilizan Scrum para el desarrollo de sus productos [41].

Además de lo anteriormente mencionado se tiene un experto en el tema de Scrum. El codirector del trabajo de grado posee conocimientos solidos acerca de esta metodología, lo cual llevo a tomar la decisión de elegir Scrum como complemento del proceso propuesto.

En la Figura 3 se muestra una visión general del proceso propuesto.



3.9. Proceso liviano de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

En la tabla 6 se muestra la caracterización del proceso propuesto.

Tabla 6. Caracterización del proceso.

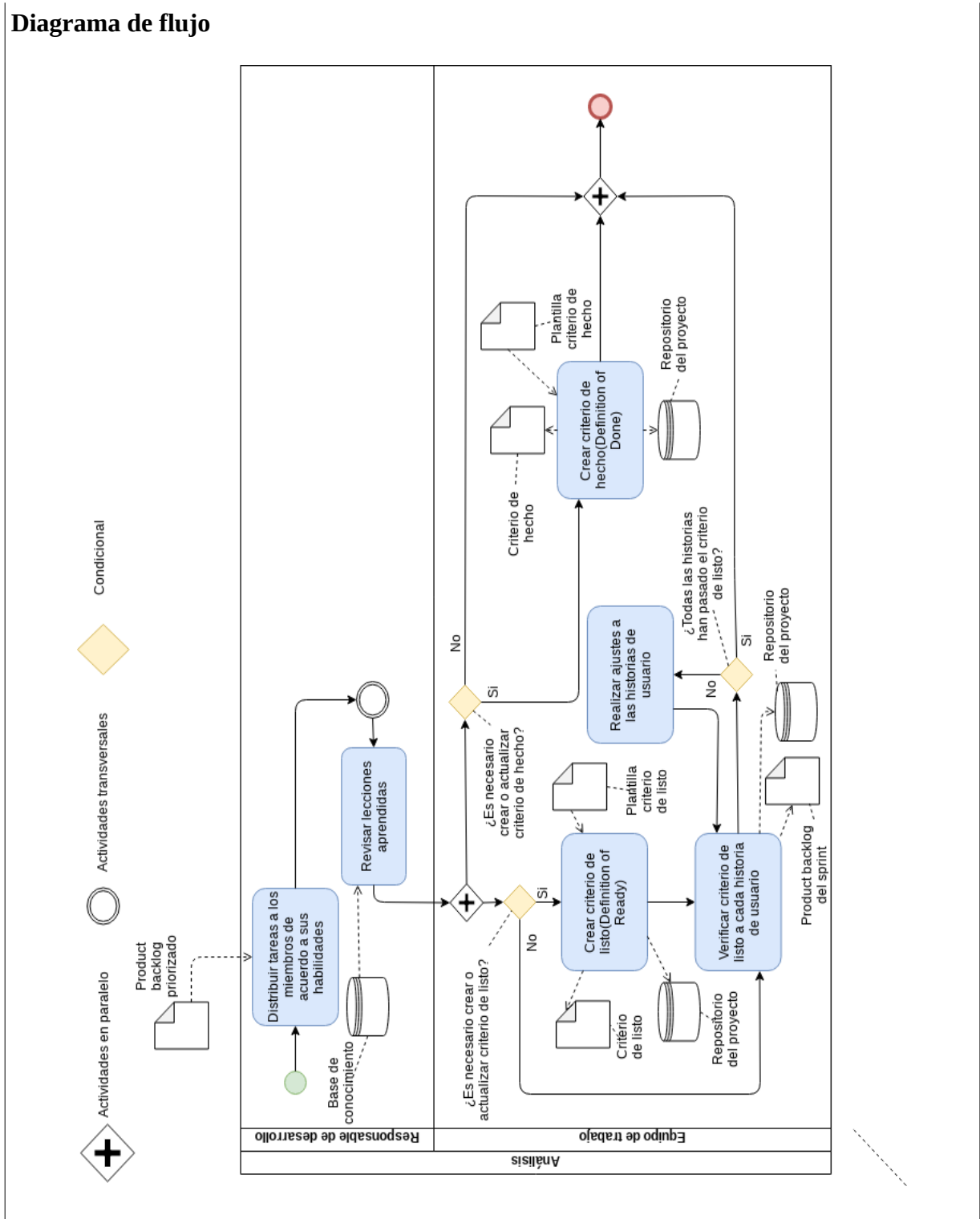
Proceso	DEWE (Desarrollo Web Empresarial).
Categoría	Operación
Propósito	El propósito de este proceso de desarrollo de software es la creación de aplicaciones web teniendo en cuenta las necesidades especificadas en los requisitos obtenidos, siguiendo una serie de etapas como lo son: análisis, diseño, implementación, despliegue y pruebas.
Descripción	<p>En el proceso de desarrollo de aplicaciones web se pueden realizar una o más iteraciones dependiendo de las necesidades de cada cliente. Al iniciar el proyecto se deben ejecutar de forma iterativa e incremental las siguientes disciplinas:</p> <ul style="list-style-type: none"> • Análisis: Involucra que el equipo entienda el objetivo a cumplir durante el desarrollo del sprint, las responsabilidades de cada participante, que se debe tener en cuenta para empezar a trabajar en la construcción de los artefactos, y cuando las funcionalidades están listas. • Diseño: Se plantea la arquitectura general del software, lo cual involucra el diseño de datos, diseño arquitectónico, diseño de la interfaz de usuario, entre otros. La arquitectura elegida debe obedecer a atributos de calidad como: I) Seguridad, II) Desempeño, III) Usabilidad y IV) Mantenibilidad. • Implementación: Toma como entradas los artefactos creados en la fase de diseño, para obtener como salida un producto que satisfaga las necesidades del clientes y demás interesados.

	<ul style="list-style-type: none"> • Despliegue en PreProducción: Es realizado internamente en la empresa, mas no en un ambiente de producción. Este despliegue pretende simular un entorno lo mas real posible, que permita realizar pruebas al producto web, teniendo en cuenta el ambiente en el cual la aplicación web estará en funcionamiento. • Pruebas: Se sugieren una serie de pruebas opcionales (Pruebas de seguridad, pruebas de usabilidad, pruebas de accesibilidad, pruebas de desempeño, pruebas de link testing, pruebas de browser testing y pruebas de diseño adaptativo – responsive) y otras obligatorias(pruebas funcionales). 	
Objetivos	<ul style="list-style-type: none"> • Lograr que el equipo de trabajo pueda entender las necesidades del cliente. • Lograr que los productos de salida sean consistentes con los productos de entrada en cada iteración realizada. • Permitir a las Mipymes contar con una guía para el desarrollo de aplicaciones web, de forma que el sistema final cumpla con los requisitos especificados. 	
Responsabilidad y autoridad	Responsable: Responsable de desarrollo de software. Autoridad: Responsable de la administración del proyecto.	
Roles involucrados y competencias		
Abreviatura	Rol	Competencias
AR	Arquitecto	Conocimiento de la plataforma tecnológica objetivo, conocimiento de los recursos existentes que pueden ser reutilizados, visión global del negocio y de las soluciones de arquitectura que garantizan la evolución del sistema
DU	Diseñador de interfaz de usuario	Conocimiento en diseño de interfaces de usuario y criterios ergonómicos.

DE	Desarrollador	Conocimiento y/o experiencia en la programación, integración y pruebas unitarias.
EP	Equipo de pruebas	Conocimiento y experiencia en la planificación y realización de pruebas de integración y de sistema.
RD	Responsable de desarrollo	Conocimiento y experiencia en el desarrollo de software.
ET	Equipo de Trabajo	Conocimiento y experiencia de acuerdo a su rol.
IO	Ingeniero de operaciones	Conocimiento en configuración de herramientas y despliegue de aplicaciones web.
A1. Análisis		
Entradas	<ul style="list-style-type: none"> • Product backlog priorizado. • Plantilla criterio de listo (Definition of ready). En el anexo G se indica una plantilla para la definición de este criterio. • Plantilla criterio de hecho (Definition of done). En el anexo H se indica una plantilla como ayuda para la definición de este criterio. 	
Rol	Actividades	
RD ET	A1.1. Distribuir tareas a los miembros de acuerdo a sus habilidades. <ul style="list-style-type: none"> • Identificar roles en el equipo. • Identificar tareas a realizar. • Asignar tareas a cada rol. 	
ET	A1.2. Revisar lecciones aprendidas (Esta actividad es transversal a la fase, con lo cual puede ser ejecutada en cualquier momento de esta fase).	

	A1.3. Si es necesario crear o actualizar el criterio de listo ir a la actividad A1.5 , de lo contrario ir a A1.6 .
	A1.4. Si es necesario crear o actualizar el criterio de hecho ir a la actividad A1.9 , de lo contrario ir al final de la fase.
ET CL	A1.5. Crear criterio de listo (Definition of ready). <ul style="list-style-type: none"> • Realizar reunión con el equipo. • Hacer una lista de elementos a incluir en el criterio. • Analizar cada elemento y discutir si se incluye o no. • Construir la lista de elementos a evaluar.
ET CL	A1.6. Verificar criterio de listo a cada historia de usuario. <ul style="list-style-type: none"> • Elegir una historia de usuario. • Verificar el cumplimiento de cada elemento en la historia de usuario. • Escribir los ajustes a realizar en la historia de usuario. • Repetir los tres pasos anteriores hasta que se hayan verificado todas las historias de usuario.
	A1.7. Si alguna historia de usuario no ha pasado el criterio de listo ir a la actividad A1.8 , de lo contrario ir al final de la fase.
ET CL	A1.8. Realizar ajustes a las historias de usuario y volver al paso A1.6 .
ET CL	A1.9. Crear criterio de hecho (Definition of done). <ul style="list-style-type: none"> • Realizar reunión con el equipo. • Escuchar propuestas de elementos a incluir en el criterio. • Analizar cada elemento y discutir si se incluye o no. • Construir la lista de criterios de hecho. Una vez creado el criterio de hecho ir al final de la fase.
Salidas	<ul style="list-style-type: none"> • Criterio de listo. • Criterio de hecho. • Product backlog del sprint revisado.

Diagrama de flujo

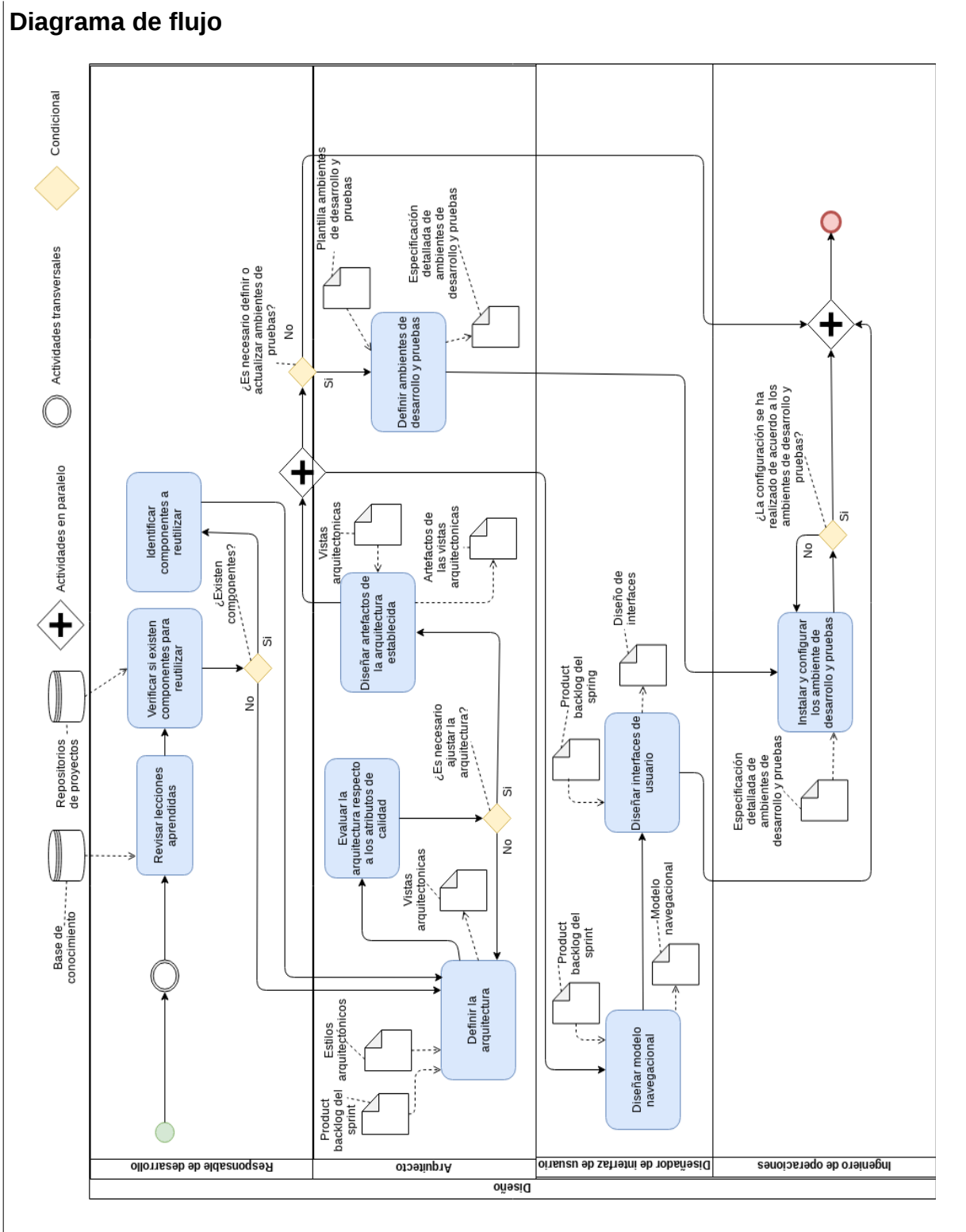


A2. Diseño	
Entradas	<ul style="list-style-type: none"> • Product backlog del sprint. • Estilos arquitectónicos. En el anexo I se indican patrones de arquitectura y Estilos arquitectónicos como ejemplo. En este anexo se definen algunos patrones de arquitectura y la vista de 4+1 de Kitchen. • Plantilla ambientes de desarrollo y pruebas. Se puede ver en el anexo J.
Rol	Actividades
ET	A2.1. Revisar lecciones aprendidas (Esta actividad es transversal a la fase, con lo cual puede ser ejecutada en cualquier momento de esta fase).
RD DE	A2.2. Verificar si existen componentes para reutilizar.
	A2.3. Si existen componentes a reutilizar ir a la actividad A2.4 , de lo contrario continuar con la actividad A2.5 .
RD DE	A2.4. Identificar componentes a reutilizar. <ul style="list-style-type: none"> • Hacer un listado de los componentes disponibles que pueden ayudar al desarrollo del proyecto. • Elegir los componentes que aportan valor al desarrollo del proyecto. •
AR	A2.5. Definir la arquitectura. <ul style="list-style-type: none"> • Realizar un listado de posibles estilos arquitectónicos. • Analizar cada estilo arquitectónico identificando ventajas y desventajas de acuerdo al desarrollo a realizar. • Elegir el mejor estilo arquitectónico de acuerdo a la necesidad presentada en el proyecto.

	<ul style="list-style-type: none"> • Elegir las vistas arquitectónicas que se van a utilizar (se recomienda utilizar como mínimo las vistas lógica, de despliegue y física). • Elegir los artefactos necesarios de cada vista arquitectónica (para la vista de lógica se recomienda como mínimo generar el diagrama de clases y modelo de datos, para la vista de despliegue generar el diagrama de componentes y paquetes, y por último para la vista física general el diagrama de despliegue).
AR	<p>A2.6. Evaluar la arquitectura respecto a los atributos de calidad.</p> <ul style="list-style-type: none"> • Verificar que el estilo arquitectónico elegido este acorde a los atributos de calidad necesarios para el desarrollo de una aplicación web. • Verificar que las vistas arquitectónicas y sus artefactos estén acorde a los atributos de calidad necesarios para el desarrollo de una aplicación web.
	<p>A2.7. Si es necesario realizar algún ajuste a la arquitectura, ir a la actividad A2.5, de lo contrario continuar con la actividad A2.8.</p>
AR	<p>A2.8. Diseñar artefactos de la arquitectura establecida.</p>
	<p>A2.9. Si es necesario definir o actualizar ambientes de pruebas ir a la actividad A2.10, de lo contrario terminar con la fase.</p>
RD AR	<p>A.2.10. Definir ambientes de desarrollo y pruebas.</p> <ul style="list-style-type: none"> • Definir lenguaje de programación para el backend. • Definir lenguaje de programación para el frontend. • Definir herramienta para realizar pruebas unitarias automatizadas. • Definir herramientas para realizar pruebas automatizadas de requisitos funcionales y no funcionales.

	<ul style="list-style-type: none"> Definir herramienta para verificar el estándar de codificación. Definir herramienta para la base de datos. <p>Continuar con la actividad A2.13.</p>
DU	A2.11. Diseñar modelo navegacional.
DU DI	A2.12. Diseñar interfaces de usuario, una vez terminado el diseño se finaliza la fase.
IO	A2.13. Instalar y configurar los ambiente de desarrollo y pruebas.
	A2.14. Si la configuración se ha realizado de acuerdo a los ambientes de desarrollo y pruebas se termina la fase, de lo contrario ir a la actividad A2.13 .
Salidas	<ul style="list-style-type: none"> Vistas arquitectónicas. Artefactos de las vistas arquitectónicas. Especificación detallada de ambientes de desarrollo y pruebas. Modelo navegacional. Diseño de interfaces.

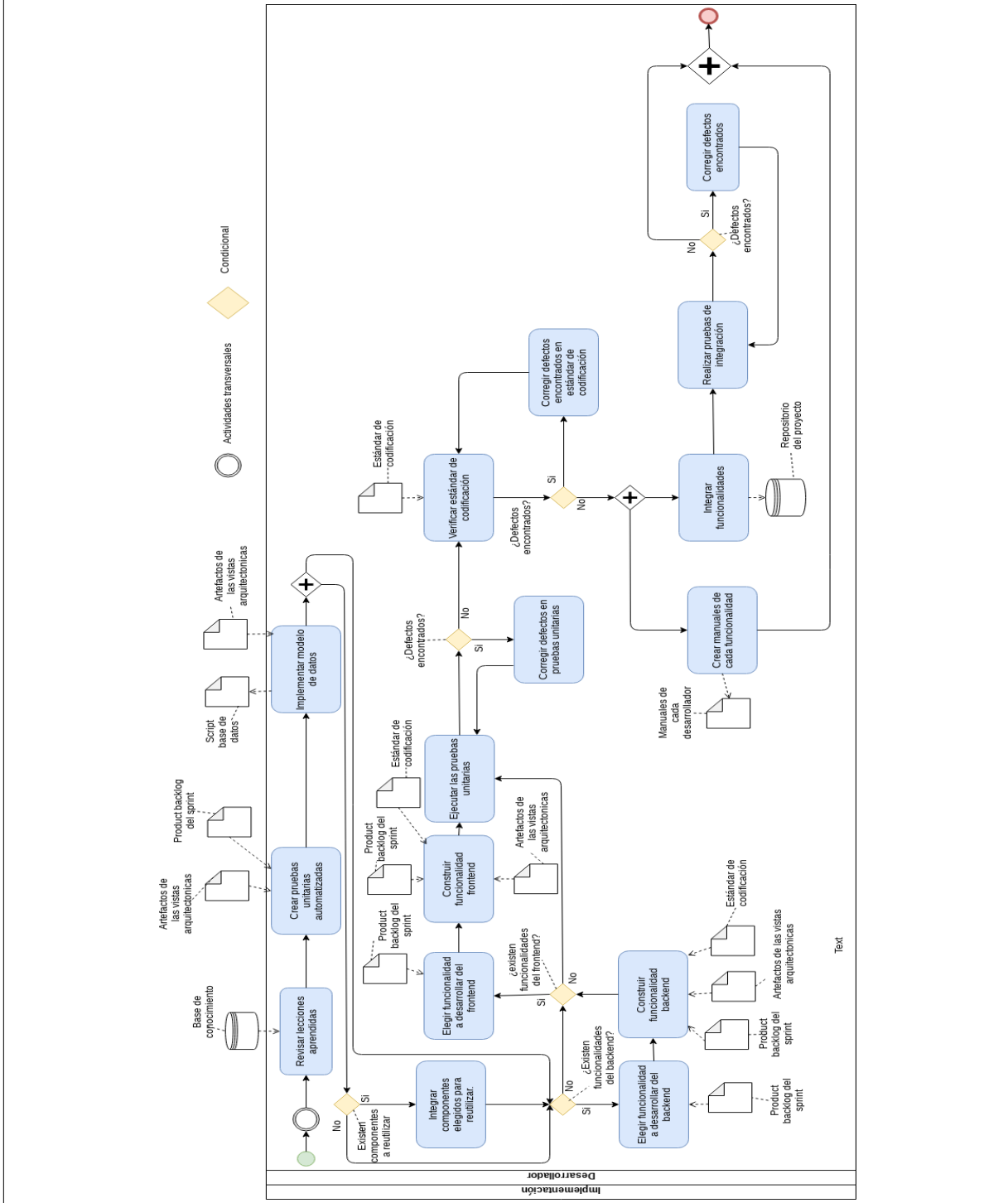
Diagrama de flujo



A3. Implementación	
Entradas	<ul style="list-style-type: none"> • Artefactos de las vistas arquitectónicas. • Product backlog del sprint. • Estándar de codificación (Cada empresa lo define de acuerdo a las practicas que utilizan).
Rol	Actividades
DE	A3.1. Revisar lecciones aprendidas (Esta actividad es transversal a la fase, con lo cual puede ser ejecutada en cualquier momento de esta fase).
DE	A3.2. Crear pruebas unitarias automatizadas. <ul style="list-style-type: none"> • Definir métodos a los cuales se les van a realizar las pruebas. • Definir las pruebas a realizar en cada método. • Construir las pruebas automatizadas que se realizarán a los métodos.
DE	A3.3. Implementar modelo de datos.
	A3.4. Si existen componentes a reutilizar ir a la actividad A3.5 , de lo contrario ir al condicional A3.6 .
	A3.5. Integrar componentes elegidos para reutilizar.
DE	A3.6. Si existen funcionalidades del backend, ir a la actividad A3.7 , de lo contrario ir al condicional A3.9 .
DE	A3.7. Elegir funcionalidad a desarrollar del backend.
DE	A3.8. Construir funcionalidad backend.
	A3.9. Si existen funcionalidades de frontend para realizar ir a la actividad A3.10 , de lo contrario continuar con la actividad A3.12 .
DE	A3.10. Elegir funcionalidad a desarrollar del frontend.
DE	A3.11. Construir funcionalidad frontend.
DE	A3.12. Ejecutar las pruebas unitarias automatizadas.

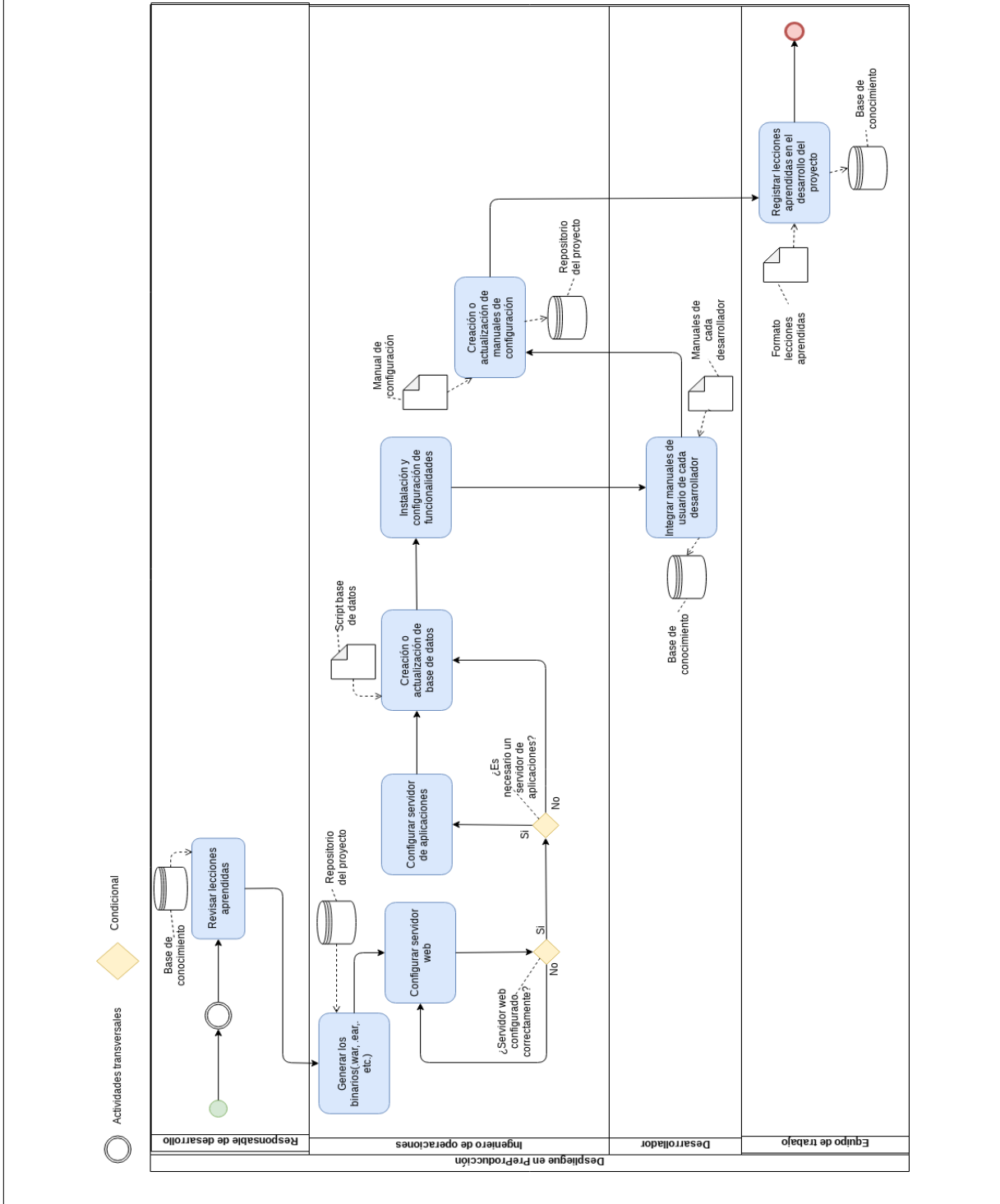
	A3.13. Si se han encontrado defectos en las pruebas unitarias ir a la actividad A3.14 , de lo contrario continuar con la actividad A3.15 .
DE	A3.14. Corregir defectos en pruebas unitarias, luego ir a la actividad A3.12 .
DE	A3.15. Verificar estándar de codificación.
	A3.16. Si se han encontrado defectos en el estándar de codificación ir a la actividad A3.17 , de lo contrario continuar con la actividad A3.18 .
DE	A3.17. Corregir defectos encontrados verificando estándar de codificación, luego ir a la actividad A3.15 .
DE	A3.18. Integrar funcionalidades.
DE	A3.19. Realizar pruebas de integración.
	A3.20. Si existen defectos en las pruebas de integración ir a la actividad A3.21 , de lo contrario la fase termina.
DE	A3.21. Corregir defectos encontrados en pruebas de integración, luego ir a la actividad A3.19 .
DE	A3.22. Crear manuales de cada funcionalidad, con lo cual se termina la fase.
Salidas	<ul style="list-style-type: none"> • Código de funcionalidades construidas del backend. • Código de funcionalidades construidas del frontend. • Script base de datos. • Manuales de cada desarrollador de las funcionalidades que le corresponde.

Diagrama de flujo:



A4. Despliegue en PreProducción	
Entradas	<ul style="list-style-type: none"> • Script de base de datos. • Manuales de cada desarrollador. • Manual de configuración(Se toma como entrada en caso de que ya se tenga un manual). • Formato lecciones aprendidas. Este formato se puede ve en el anexo K.
Rol	Actividades
RD	A4.1. Revisar lecciones aprendidas (Esta actividad es transversal a la fase, con lo cual puede ser ejecutada en cualquier momento de esta fase).
IO	
DE	
IO	A4.2. Generar los binarios de la aplicación.
IO	A4.3. Configurar servidor web.
IO	A4.4. Si el servidor web no es configurado correctamente ir a la actividad A4.3 , de lo contrario ir al condicional A4.5 .
IO	A4.5. Si es necesario un servidor de aplicaciones ir a la actividad A4.6 , de lo contrario continuar en la actividad A4.7 .
DE	A4.6. Configurar servidor de aplicaciones.
IO	A4.7. Creación o actualización de base de datos.
IO	A4.8. Instalación y configuración de funcionalidades (Archivos binarios generados a partir de la aplicación).
DE	A4.9. Integrar manuales de usuario de cada desarrollador.
	A4.10. Creación o actualización de manuales de configuración.
ET	A4.11. Registrar lecciones aprendidas en el desarrollo del proyecto, luego de registrar las lecciones aprendidas se termina la fase.
Salidas	<ul style="list-style-type: none"> • Manual de usuario (Contiene todos los manuales de funcionalidades de cada desarrollador). • Lecciones aprendidas.

Diagrama de flujo:



A5. Pruebas	
Entradas	<ul style="list-style-type: none"> • Product backlog del sprint • Criterio de hecho. • Plantilla para pruebas. Se puede ver en el anexo L. • Herramientas pruebas funcionales. Se puede ver en el anexo M. • Herramientas pruebas de usabilidad. Se puede ver en el anexo N. • Herramientas pruebas de seguridad. Se puede ver en el anexo O. • Herramientas pruebas de rendimiento o desempeño. Se puede ver en el anexo P. • Herramientas pruebas de accesibilidad. Se puede ver en el anexo Q. • Herramientas pruebas de diseño adaptativo. Se puede ver en el anexo R. • Herramientas pruebas de link testing. Se puede ver en el anexo S. • Herramientas pruebas de browser testing. Se puede ver en el anexo T.
Rol	Actividades
ET	A5.1. Revisar lecciones aprendidas (Esta actividad es transversal a la fase, con lo cual puede ser ejecutada en cualquier momento de esta fase).
	A5.2. Si se realiza tercerización de pruebas ir al paso A5.3 , de lo contrario continuar con la actividad A5.9 .
	A5.3. Si se han encontrado defectos en la tercerización de pruebas, ir a la actividad A5.4 , de lo contrario ir a la actividad A5.6 .
DE	A5.4. Corregir defectos encontrados.

	A5.5. Si es necesario hacer pruebas al código corregido ir al paso A5.2 , de lo contrario continuar con la actividad A5.6 .
RD	A5.6. Verificar el criterio de hecho. <ul style="list-style-type: none"> • Elegir una funcionalidad. • Verificar el cumplimiento de cada elemento en la funcionalidad. • Repetir los tres pasos anteriores hasta que se hayan verificado todas las funcionalidades.
	A5.7. Si el criterio de hecho es aceptado terminar la fase, de lo contrario ir a la actividad A5.8 .
DE	A5.8. Corregir observaciones realizadas en el criterio de hecho, luego volver a la actividad A5.6 .
EP	A5.9. Elegir funcionalidad a probar.
EP	A5.10. Realizar pruebas de funcionalidad. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Codificar las pruebas funcionales para su ejecución automatizada. • Ejecutar scripts. • Documentar los errores en caso de que se hayan presentado.
	A5.11. Si existen errores en las pruebas de funcionalidad ir a la actividad A5.12 , de lo contrario continuar con el condicional A5.13 .
DE	A5.12. Corregir errores en pruebas de funcionalidad, luego volver a la actividad A5.10 .
	A5.13. Si las pruebas de usabilidad son requeridas ir a la actividad A5.14 , de lo contrario continuar con el condicional A5.15 .

EP	<p>A5.14. Realizar pruebas de usabilidad.</p> <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas. • Documentar los errores en caso de que se hayan presentado.
	<p>A5.15. Si existen errores en las pruebas de usabilidad ir a la actividad A5.16, de lo contrario ir al condicional A5.17.</p>
DE	<p>A5.16. Corregir errores en pruebas de usabilidad, luego volver a la actividad A5.14.</p>
	<p>A5.17. Si las pruebas de seguridad son requeridas ir a la actividad A5.18, de lo contrario continuar con el condicional A5.21.</p>
EP	<p>A5.18. Realizar pruebas de seguridad.</p> <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Crear y ejecutar scripts si son necesarios. • Ejecutar pruebas.
	<p>A5.19. Si existen errores al realizar las pruebas de seguridad ir a la actividad A5.20, de lo contrario continuar con el condicional A5.21.</p>
DE	<p>A5.20. Corregir errores en pruebas de seguridad, luego volver a la actividad A5.18.</p>
	<p>A5.21. Si las pruebas de desempeño son requeridas ir a la actividad A5.22, de lo contrario ir al condicional A5.25.</p>
EP	<p>A5.22. Realizar pruebas de rendimiento o desempeño.</p> <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Crear y ejecutar scripts si son necesarios. • Ejecutar pruebas.
	<p>A5.23. Si existen errores al realizar las pruebas de rendimiento o desempeño ir a la actividad A5.24, de lo contrario ir al paso A.5.25.</p>

DE	A5.24. Corregir errores en pruebas de desempeño, volver a la actividad A5.22 .
	A5.25. Si las pruebas de accesibilidad son requeridas, ir a la actividad A5.26 , de lo contrario continuar con el condicional A5.29 .
EP	A5.26. Realizar pruebas de accesibilidad. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Crear y ejecutar scripts si son necesarios. • Ejecutar pruebas.
	A5.27. Si existen errores en las pruebas de accesibilidad ir a la actividad A5.28 , de lo contrario ir al condicional A5.29 .
DE	A5.28. Corregir errores en pruebas de accesibilidad, luego volver a la actividad A5.26 .
	A5.29. Si las pruebas de diseño adaptativo son requeridas, ir a la actividad A5.30 , de lo contrario continuar con el condicional A5.33 .
EP	A5.30. Realizar pruebas de diseño adaptativo. <ul style="list-style-type: none"> • Definir dispositivos y tamaños a probar. • Ejecutar pruebas.
	A5.31. Si existen errores en las pruebas de diseño adaptativo ir a la actividad A5.32 , de lo contrario ir al condicional A5.33 .
DE	A5.32. Corregir errores en diseño adaptativo, luego volver a la actividad A5.30 .
	A5.33. Si las pruebas de Link Testing son requeridas, ir a la actividad A5.34 , de lo contrario ir al condicional A5.37 .
EP	A5.34. Realizar pruebas de Link Testing. <ul style="list-style-type: none"> • Definir enlaces a probar. • Crear pruebas. • Ejecutar pruebas.

	A5.35. Si existen errores en las pruebas de Link Testing ir a la actividad A5.36 , de lo contrario ir al paso A5.37 .
DE	A5.36. Corregir errores en Link Testing, luego volver a la actividad A5.34 .
	A5.37. Si las pruebas de Browser Testing son requeridas ir a la actividad A5.38 , de lo contrario ir al condicional A5.41 .
EP	A5.38. Realizar pruebas de browser testing. <ul style="list-style-type: none"> • Definir navegadores a probar. • Ejecutar pruebas.
	A5.39. Si existen errores en las pruebas de browser testing ir a la actividad A5.40 , de lo contrario ir al paso A5.41 .
DE	A5.40. Corregir errores en Browser Testing, luego volver a la actividad A5.38 .
	A5.41. Si existen más funcionalidades para probar en la iteración volver a la actividad A5.9 , de lo contrario ir a la actividad A5.6 .
Salidas	<ul style="list-style-type: none"> • Aplicación web probada

Una vez se han realizado las pruebas y todo ha salido bien, se procede al despliegue de la aplicación en un entorno real. Para esto se debe tomar como guía la fase de despliegue en preproducción para pruebas, el cual permite tener una aplicación web funcionando.

Cuando se ha terminado todo este proceso del desarrollo de las funcionalidades, Scrum continua con sus actividades, para realizar la gestión del proyecto de la forma correcta, con lo cual se realiza una nueva ejecución de una iteración utilizando el proceso.

Por recomendación de los jurados al momento de la sustentación se decide modificar el nombre de la propuesta. La propuesta ya no es contemplada como un proceso ágil, sino como un proceso liviano.

Capítulo 4. Evaluación de la propuesta por medio de un grupo focal

En este capítulo se presentan los resultados obtenidos al evaluar la propuesta mediante un grupo focal. Participaron personas que poseen conocimientos sobre ingeniería de software, aplicaciones web y metodologías ágiles.

4.1. Grupo focal (Focus group)

“Un grupo focal es una discusión cuidadosamente planificada” [42], realizada para obtener las perspectivas de un grupo sobre un tema en específico [42]. El número de participantes debe ser mínimo de 3 y máximo de 12, las personas que integran el grupo son elegidas de acuerdo a sus habilidades y experiencia (Estas habilidades y experiencia deben estar relacionadas con el tema a tratar) [42]. Para realizar el grupo focal se debe construir una agenda que permita guiar la sesión para así cumplir con los objetivos deseados. En esta agenda se plasman los pasos a seguir durante todo el grupo focal [42].

4.1.1. Realización del grupo focal

La primera versión del proceso llamada construcción web empresarial fue presentada en el grupo focal para obtener las observaciones de los participantes. En este grupo focal participaron 5 personas, las cuales conocen y tienen experiencia en ingeniería de software, desarrollo web y metodologías ágiles. A partir de las observaciones realizadas en el grupo focal, se refina la propuesta y se da un nuevo nombre: DEWE (Desarrollo web empresarial).

La propuesta final se puede ver en la tabla 6 del capítulo 3, y la versión inicial está disponible en el anexo F.

4.1.1.1. Definición de elementos utilizados para realizar el Grupo Focal

En la tabla 7 se presentan los artefactos utilizados para realizar el grupo focal.

Tabla 7. Artefactos utilizados en el grupo focal.

No.	Documento	Descripción	Anexo o enlace asociados
1	Agenda de trabajo	Documento que contiene los pasos a seguir durante la realización del grupo focal.	Anexo A: Agenda de trabajo
2	Cuestionario de preguntas relacionadas al proceso propuesto.	Documento que contiene nueve(9) preguntas acerca del proceso, las cuales deben ser respondidas por los participantes al grupo focal.	Anexo B: Cuestionario de preguntas relacionadas al proceso propuesto
3	Ficha de asistencia	Documento que contiene información (Nombre, estudios realizados, cargo y experiencia) de cada participante del grupo focal.	Anexo C: Ficha de asistencia
4	Documentos del proceso	Se envían dos documentos a los participantes del grupo focal. El primer documento muestra un resumen (descripción y gráficas)	Anexo U: Documento resumen Anexo F: Documento completo

		del proceso y el segundo muestra el proceso completo (gráficas y plantilla).	
--	--	--	--

4.1.1.2. Selección de los participantes

Los participantes que asistieron al grupo focal se seleccionaron teniendo en cuenta las siguientes actividades:

- Definición de los participantes: Para esta actividad se tienen en cuenta lo siguiente: i) Desarrolladores de aplicaciones web con experiencia, ii) Personas con conocimiento en metodologías ágiles, ingeniería del software.
- Identificación de participantes potenciales: Teniendo en cuenta los perfiles identificados, se solicitó la participación de las personas que cumplen con las características mencionadas anteriormente.

En la Tabla 8 se muestran los datos de los participantes que asistieron al grupo focal. De cada participantes se indican los datos escritos en el documento que se les facilitó en el grupo focal (ver anexo E).

Tabla 8. Participantes del grupo focal

N°	Ocupación	Experiencia	Perfil
1	Ingeniero de sistema/ Gerente de Sunset Software House SAS.	Administrador de bases de datos. <ul style="list-style-type: none"> • Analista y desarrollador. • Consultor en calidad de software (ISO 9126, 25000). • Scrum Master • Director de proyectos. 	<ul style="list-style-type: none"> • Especialista en desarrollo de software. • Ingeniero de sistemas. • Scrum Master certificado.

		<ul style="list-style-type: none"> • Gerente empresa de desarrollo de software. 	
2	Estudiante de ingeniería de sistemas/Analista y desarrollador.	<ul style="list-style-type: none"> • Construcción de arquitecturas. • aplicación de metodologías de trabajo y desarrollo. • Lider de proyectos. • Desarrollo web. 	<ul style="list-style-type: none"> • Conocimiento en Scrum. • Conocimiento en metodologías como Xp.
3	Ingeniero de sistemas/Desarrollador web.	<ul style="list-style-type: none"> • Participación en proyectos web de diferentes tamaños. • Participación en proyectos dirigidos con scrum. 	<ul style="list-style-type: none"> • Ingeniero de sistemas especializado en la administración de la información y bases de datos. • Certificación en Scrum.
4	Ingeniero electrónico/Desarrollador web.	<ul style="list-style-type: none"> • Desarrollo de software en lenguajes como: javascript, php, java, html y css3. • 	<ul style="list-style-type: none"> • Estudios en metodologías de gestión de proyectos como: ITIL, Cobit y Scrum.

5	Ingeniero de sistemas/Desarrollador de aplicaciones web.	<ul style="list-style-type: none"> • Desarrollo de sitios web como: www.sumercado.co y www.marksoftware.co. • Creación del proceso CUS-PH (Proceso para el desarrollo de software en ambientes universitarios). 	<ul style="list-style-type: none"> • Ingeniero de sistemas. • Tecnólogo en telecomunicaciones.
---	--	--	--

4.1.1.3. Sesión del grupo focal

En la Figura 4 se muestran los participantes del grupo focal.

Figura 4. Participantes del grupo focal



El grupo focal estuvo conformado por un moderador, un supervisor y los participantes elegidos. En la realización del grupo focal se obtuvo una realimentación sobre el proceso por parte de personas que laboran en ambientes de empresa y que cuentan con experiencia en desarrollo web, metodologías ágiles e ingeniería de software.

En la Tabla 8 se pueden observar las actividades llevadas a cabo durante la realización del grupo focal.

Tabla 8. Agenda grupo focal (27/03/2019).

N°	Descripción	Hora	
		Desde	Hasta
1	Agradecimiento a los participantes por la asistencia.	4:00 pm	4:05 pm
2	Presentación del investigador por parte del supervisor. <ul style="list-style-type: none"> • Explicar a los participantes cuál será el proceso a seguir con los resultados obtenidos. • Explicar a los participantes el por qué fueron seleccionados. 	4:05 pm	4:10 pm
3	Presentación de la propuesta.	4:10 pm	4:35 pm
4	Dar espacio a los participantes para dar hacer sus apreciaciones sobre las actividades contenidas en el proceso propuesto.	4:35 pm	5:00 pm
5	Los participantes dan respuesta a la encuesta definida por el investigador.	5:00 pm	5:10 pm
6	Los participantes llenan la ficha de asistencia.	5:10 pm	5:15 pm
7	Refrigerio.	5:15 pm	5:20 pm
8	Agradecimientos por parte del investigador a los participantes.	5:20 pm	5:25 pm
9	Cierre de sesión.	5:25 pm	5:30 pm

4.1.1.4. Captura y registro de la información en el grupo focal.

Durante la evaluación de la propuesta, se realizó la documentación de las observaciones realizadas con el apoyo de un documento (ver anexo D) y una grabación de audio. El documento fue utilizado para obtener retroalimentación del proceso mediante un cuestionario con una serie de preguntas sobre la propuesta. Cada pregunta fue de selección múltiple y se dio un espacio para dar las observaciones correspondientes. Para la discusión sobre la propuesta se utilizó la grabación de un audio en el cual quedaron registradas las observaciones de cada participante del grupo focal.

4.1.2. Reporte de resultados

Una vez finalizada la aplicación del grupo focal, se llevó a cabo una clasificación de las respuestas y aportes dados por parte de los participantes.

4.1.2.1. Resultados del cuestionario

Al finalizar el debate, se entregó un cuestionario (ver anexo D) a cada participante con nueve preguntas sobre el proceso. Se utilizó la escala de Likert para cada pregunta, con lo cual se tiene 5 opciones de respuesta.

Los resultados obtenidos se muestran a continuación en la Tabla 9. En las celdas se especifica el número de participantes que eligieron una determinada opción.

Tabla 9. Resultados encuesta.

Id	Pregunta	Resultados				
		Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
1	De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?				2	3
2	¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?			1	1	3
3	¿Considera que las actividades propuestas en las fases del proceso son apropiadas y aplicables a un entorno empresarial para el desarrollo de aplicaciones web?			1	2	2

4	¿Considera el proceso propuesto como ágil?		2	1	2	
5	¿Considera que al proporcionar el proceso a un equipo de trabajo puede ser fácilmente entendible y aplicable?		1	1	2	1
6	¿Considera que al proceso le falta alguna actividad importante en el desarrollo de aplicaciones web?		2		3	
7	¿Considera importante las pruebas automatizadas en el desarrollo de aplicaciones web?					5
8	¿Considera importante el desarrollo dirigido por pruebas en la construcción de aplicaciones web?				3	2
9	¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?			2	2	1

4.1.2.2. Análisis estadístico

Una vez se ha realizado el conteo de las respuestas (ver anexo D) obtenidas en cada pregunta, se llevó a cabo un análisis estadístico de cada pregunta, lo cual ayuda a identificar los factores relevantes a tener en cuenta para mejorar el proceso propuesto.

A continuación se muestra cada pregunta con una gráfica relacionada a las respuestas dadas por los participantes.

En la Figura 5 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 3 de ellas están totalmente de acuerdo y 2 están de acuerdo.

Los participantes ven como importante que el desarrollo de aplicaciones tenga un proceso que ayude a su construcción de una forma eficiente y organizada.

Figura 5. Conteo respuestas pregunta 1.

P1. De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?



En la tabla 10 se muestran los comentarios de la pregunta 1 realizados por los participantes del grupo focal.

Tabla 10. Comentarios realizados de la pregunta 1

Id	Comentario
1	Esto es importante debido a que es importante generar procesos que estén alineados con el desarrollo y además le permita encontrar fallas, así como no implicar caídas o respuestas tardías a sus clientes aun cuando existan rotaciones de personal.
2	Los procesos son importantes en todo tipo de proyectos, las miPymes no son una excepción.
3	La organización de los procesos es un factor clave para el éxito de un proyecto y la satisfacción de los clientes.
4	Estoy de acuerdo, sin embargo puede ser una metodología entendible a empresas más grandes en un futuro.
5	No registra comentarios.

En la Figura 6 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 3 de ellas están totalmente de acuerdo, 1 está de acuerdo y 1 neutral.

Los participantes consideran que el proceso debe ser un poco más claro en términos y organización de las actividades en cada fase, además que se tenga en cuenta tamaños de equipos para la ejecución del proceso. Algunos de los términos utilizados en las actividades se deben explicar de mejor forma, dado que pueden llevar a diferentes interpretaciones, en cuanto a los tamaños de equipos no se presenta ningún inconveniente debido a que una persona puede ejecutar más de un rol en el proceso.

Figura 6. Conteo respuestas pregunta 2.

P2. ¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?



En la tabla 11 se muestran los comentarios de la pregunta 2 realizados por los participantes del grupo focal.

Tabla 11. Comentarios realizados de la pregunta 2

id	Comentario
1	Considerando ajustes, totalmente de acuerdo.
2	Hay que tener en cuenta que en ocasiones las mipymes no cuentan con recursos para la implementación de procesos de desarrollo organizados, por lo tanto, se sugiere que sea más sencillo de entender y aplicar.
3	Proponer tamaños de equipos suena interesante.
4	Se deben pulir términos, fases y actividades.
5	No registra comentarios.

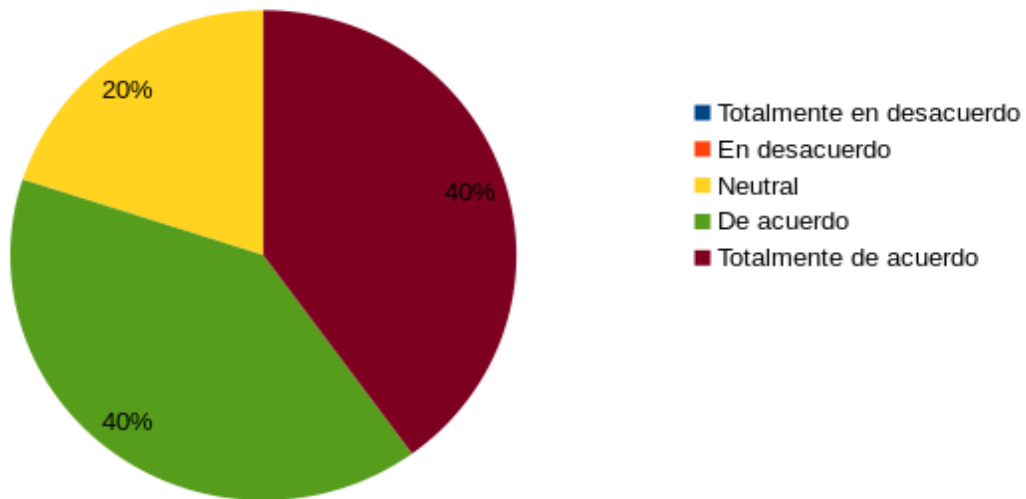
En la Figura 7 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 2 de ellas están totalmente de acuerdo, 2 están de acuerdo y 1 neutral.

Los participantes consideran que las mayoría de actividades son claras y se pueden llevar a un entorno empresarial, aunque se deben pulir algunos términos que

hacen que no todas las actividades puedan ser aplicadas en las empresas, además se recalca la importancia del proceso en un desarrollo empresarial.

Figura 7. Conteo respuestas pregunta 3.

P3. ¿Considera que las actividades propuestas en las fases del proceso son apropiadas y aplicables a un entorno empresarial para el desarrollo de aplicaciones web?



En la tabla 12 se muestran los comentarios de la pregunta 3 realizados por los participantes del grupo focal.

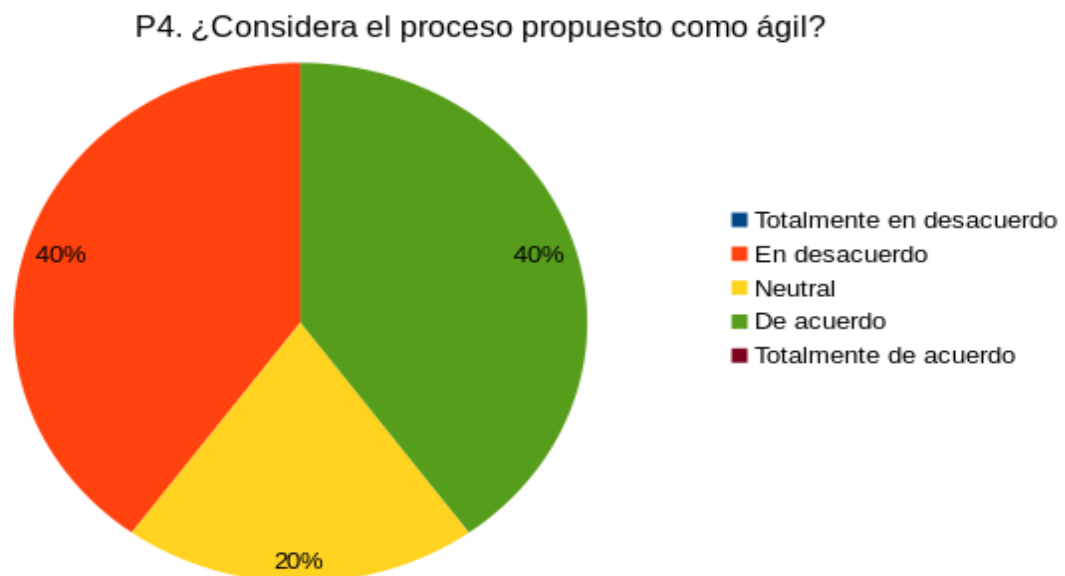
Tabla 12. Comentarios realizados de la pregunta 3

Id	Comentario
1	Si en su mayoría, lo importante es que las actividades sean claras, precisas y entendibles para quien las debe ejecutar.
2	Este tipo de procesos se pueden llevar a otros modelos como outsourcing y desarrollos a la medida.
3	Se deben revisar nombres para diferenciar de una forma mas clara si son tareas o actividades.
4	No se registran comentarios
5	No se registran comentarios

En la Figura 8 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 2 de ellas están de acuerdo, 1 está neutral y 2 en desacuerdo.

Aunque el proceso se apoya en Scrum, no fue considerado por los participantes como un proceso completamente ágil, esto debido a que las actividades contenidas en cada fase son muy lineales y no hay ejecución en paralelo que permita aplicar el proceso de la forma más eficiente, además no se dejó claro lo iterativo e incremental del proceso debido a la falta de una gráfica que muestre estos detalles.

Figura 8. Conteo respuestas pregunta 4.



En la tabla 13 se muestran los comentarios de la pregunta 4 realizados por los participantes del grupo focal.

Tabla 13. Comentarios realizados de la pregunta 4

Id	Comentario
1	El considerar a Scrum como pilar, ayuda mucho, sin embargo debe revisarse la iteración dentro de algunas fases.
2	Durante la presentación no se evidenció la naturaleza iterativa e incremental, al contrario fue muy lineal. Se sugiere construir un modelo visual o una gráfica que presente de forma global el proceso y la naturaleza iterativa.
3	Considero que es importante el que se consideren los roles de Scrum.
4	Pregunta interesante, suena bien en papel, pero es necesario tener casos de éxito a futuro para determinar si es ágil.
5	No registra comentario.

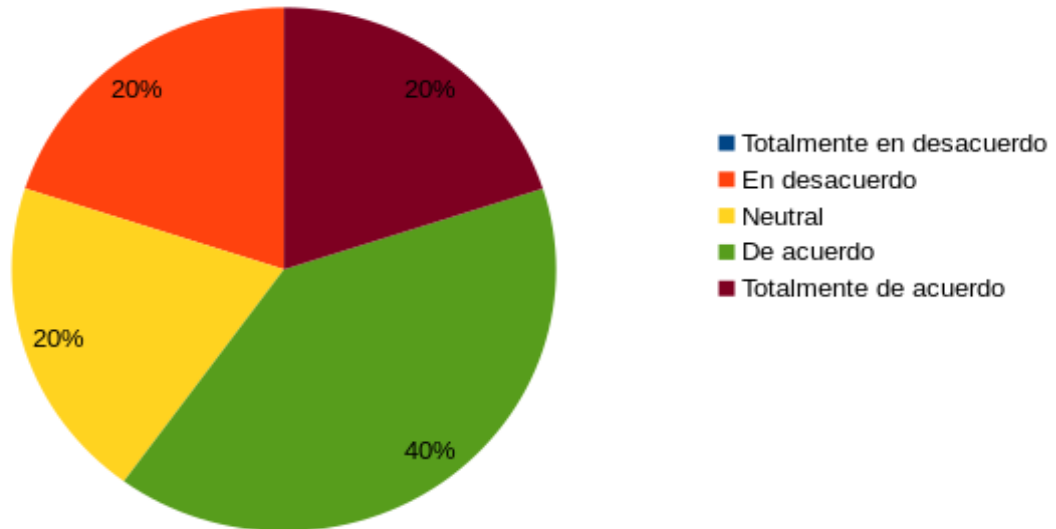
En la Figura 9 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 1 de ellas está totalmente de acuerdo, 2 están de acuerdo, 1 neutral y 1 en desacuerdo.

Los participantes sugieren la construcción de una gráfica, la cual explique la integración del proceso con Scrum. Además es necesario modificar algunos términos de las actividades para mayor claridad.

En cuanto al nivel de detalle, el proceso cuenta con una plantilla que permite ver cada una de las actividades con más detalle.

Figura 9. Conteo respuestas pregunta 5.

P5. ¿Considera que al proporcionar el proceso a un equipo de trabajo puede ser fácilmente entendible y aplicable?



En la tabla 14 se muestran los comentarios de la pregunta 5 realizados por los participantes del grupo focal.

Tabla 14. Comentarios realizados de la pregunta 5

Id	Comentario
1	Con un proyecto piloto y un equipo que proponga, seguramente si.
2	Tiene muchas actividades que requieren explicación, quizá el proceso necesite una visión de alto nivel en la que no se incluyen todos los detalles y que los detalles presentados estén en un nivel más profundo.
3	Le falta integrar mejor a Scrum.
4	No registra comentario.
5	No registra comentario.

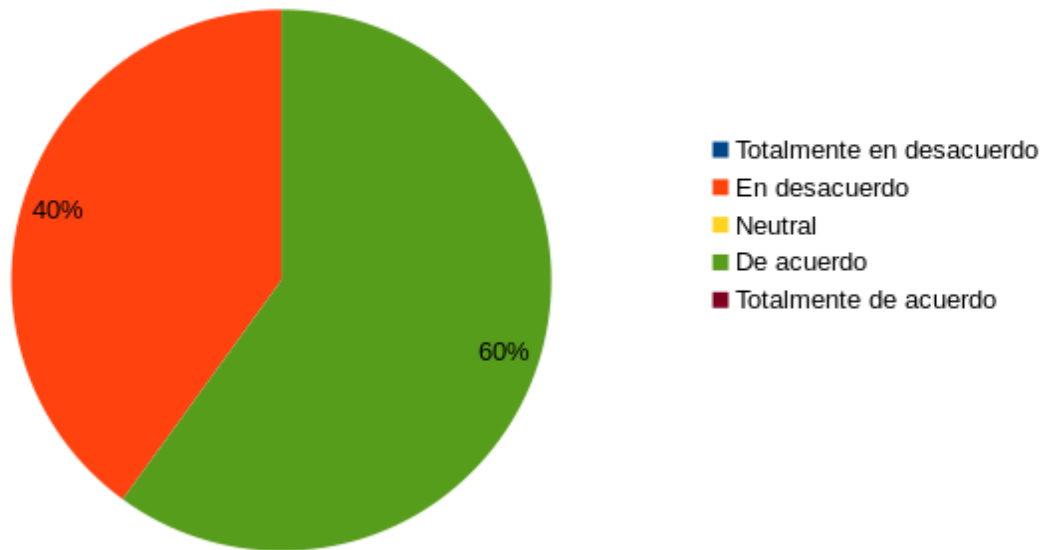
En la Figura 10 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 3 de ellas están de acuerdo y 2 en desacuerdo.

Se sugiere tener un despliegue de pruebas antes de colocar en producción la aplicación, además tener en cuenta un despliegue automatizado que permita reducir

esfuerzos al momento de probar o liberar a producción un desarrollo. Con el despliegue automatizado se puede ahorrar tiempo, recursos humanos y agilizar la aplicación del proceso propuesto.

Figura 10. Conteo respuestas pregunta 6.

P6. ¿Considera que al proceso le falta alguna actividad importante en el desarrollo de aplicaciones web?



En la tabla 15 se muestran los comentarios de la pregunta 6 realizados por los participantes del grupo focal.

Tabla 15. Comentarios realizados de la pregunta 6

Id	Comentario
1	Revisar las observaciones durante la sesión del focus group.
2	No hay ejemplos sugeridos, plantillas o patrones que se reutilizan de forma frecuente.
3	Considero que hace falta un ambiente de pre-producción posterior al testing. Además representar de forma clara el modelo global con el fin de detectar el ciclo de Scrum.
4	Posiblemente la automatización del despliegue, pero eso ya se mencionó en los comentarios.
5	No se registra comentario.

En la Figura 11 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales todas las personas están totalmente de acuerdo.

Todos los participantes están de acuerdo en que todo lo que permita automatizar y agilizar tareas es bienvenido.

Figura 11. Conteo respuestas pregunta 7.

P7. ¿Considera importante las pruebas automatizadas en el desarrollo de aplicaciones web?



En la tabla 16 se muestran los comentarios de la pregunta 7 realizados por los participantes del grupo focal.

Tabla 16. Comentarios realizados de la pregunta 7

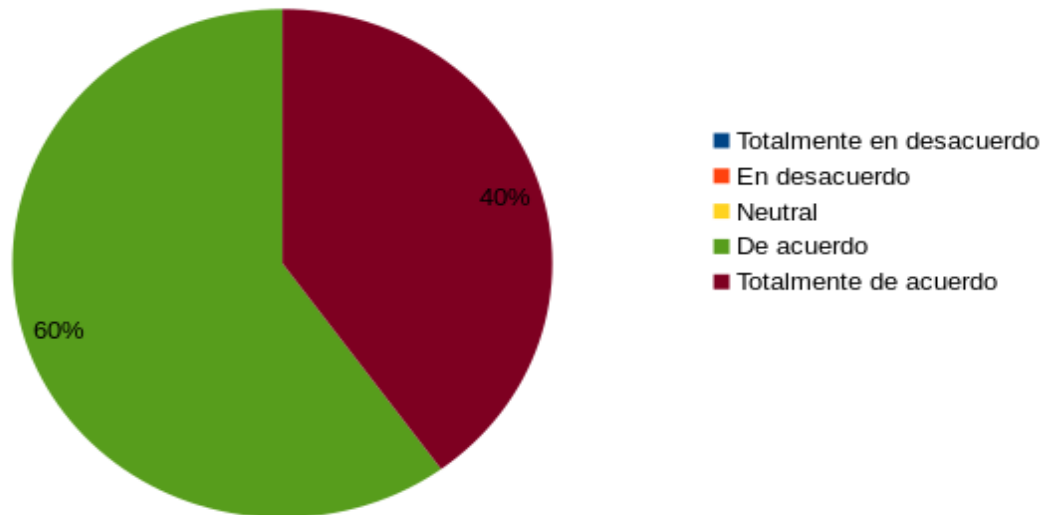
Id	Comentario
1	Toda herramienta que permita la gestión y logro de buenos resultados son bienvenidos.
2	Ahorra mucho tiempo a largo plazo.
3	No registra comentario
4	No registra comentario
5	No registra comentario

En la Figura 12 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 2 de ellas están totalmente de acuerdo y 3 de acuerdo.

El desarrollo dirigido por pruebas es muy bien aceptado como lo muestra la gráfica anterior, debido a que permite dar una mayor calidad en los productos construidos. Es una buena práctica que permite tener una idea de que se debe tener en cuenta al momento de desarrollar una funcionalidad.

Figura 12. Conteo respuestas pregunta 8.

P8. ¿Considera importante el desarrollo dirigido por pruebas en la construcción de aplicaciones web?



En la tabla 17 se muestran los comentarios de la pregunta 8 realizados por los participantes del grupo focal.

Tabla 17. Comentarios realizados de la pregunta 8

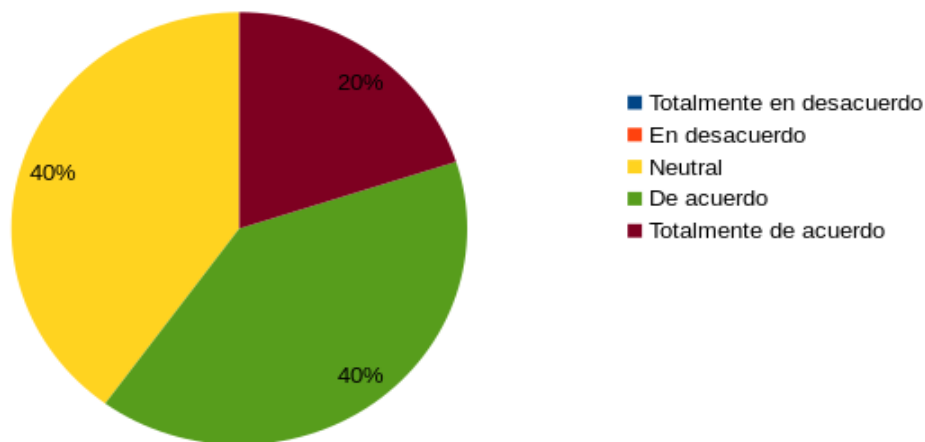
Id	Comentario
1	Es una buena práctica a tener en cuenta.
2	Puede implicar la participación de más personal en el desarrollo.
3	Mayormente las de UX (User experience), que suelen ser ajenas a los desarrolladores.
4	No registra comentario.
5	No registra comentario.

En la Figura 13 se puede observar un total de 5 respuestas a la pregunta formulada, de las cuales 1 de ellas está totalmente de acuerdo, 2 de acuerdo y 2 neutral.

La mayoría de los participantes consideran que el proceso puede cumplir con estos atributos de calidad, ya que cuenta con una fase de pruebas que evalúa requisitos no funcionales. Aunque se permite elegir el tipo de pruebas necesarias para cada desarrollo, el proceso provee las pruebas necesarias y algunas herramientas recomendadas que permiten llevar a cabo la evaluación del producto de una forma correcta.

Figura 13. Conteo respuestas pregunta 9.

P9. ¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?



En la tabla 18 se muestran los comentarios de la pregunta 9 realizados por los participantes del grupo focal.

Tabla 18. Comentarios realizados de la pregunta 9

Id	Comentario
1	Estas características no funcionales deberían estar inmersas en cualquier proceso a proponer.
2	El proceso hace énfasis en los diferentes tipos de pruebas que son los que garantizan la existencia de estos atributos.
3	Siempre y cuando se cumplan los criterios de hecho y aceptación, si.
4	No necesariamente por usar el proceso se va a cumplir con estos criterios.
5	No registra comentario.

4.1.2.3. Observaciones y mejoras sugeridas por los integrantes del grupo focal.

Las observaciones y mejoras realizadas se obtienen a partir del cuestionario y el debate realizado durante el grupo focal.

En la tabla 19 se indican las observaciones realizadas por los participantes del grupo focal y las mejoras realizadas al proceso.

Tabla 19. Resultado observaciones y mejoras sugeridas.

Id	Observación realizada	Mejora realizada
1	Verificar fases del proceso, las actividades son muy lineales.	Se revisa cada fase y se cambia la ejecución de algunas actividades lineales por actividades en paralelo.
2	Hace falta un despliegue en ambiente de pruebas.	La fase de despliegue que estaba planteada como despliegue en ambiente de producción, se pasa a despliegue en ambiente de pruebas.
3	La gráfica que muestra la visión general de la propuesta no está clara.	Se agregan más pasos en la gráfica de la visión general, con lo cual se muestra de forma más clara la integración del proceso con Scrum.
4	Al ejecutarse el proceso con el sprint, los requerimientos deben estar congelados.	Se quita de todas las fases la actividad de evaluar cambios en requisitos o nuevos requisitos.
5	Las actividades de crear criterio de listo y de hecho se repiten en cada fase.	Se agrega un condicional para realizar estas actividades siempre y cuando no hayan sido creadas.
6	Los nombres de las actividades de crear criterio de listo y de hecho no son claros.	Se agrega el nombre de "definition of done" y "definition of ready" para dar más claridad en la relación con scrum.
7	La actividad de definir ambientes de desarrollo y pruebas se repite en cada iteración.	Se agrega un condicional para que esta actividad sea realizada siempre y cuando no se haya creado antes.
8	El nombre no es adecuado para el proceso.	Se modifica el nombre inicial del proceso.
9	Hay pruebas que no pueden ser realizadas sin antes haber desplegado la aplicación.	Se cambia el orden de ejecución de las fases de pruebas y despliegue, siendo ejecutada primero la fase de despliegue y luego la fase de pruebas.

10	Los símbolos utilizados en los diagramas deben ser claros para personas que no conozcan la sintaxis de Bpmn.	En cada fase de agrega una descripción de cada símbolo utilizado.
----	--	---

Teniendo en cuenta las observaciones hechas por los participantes y realizando un análisis, se ha creado una nueva versión del proceso, la cual es presentada en el capítulo 3.

La primera versión del proceso se puede ver en el anexo F.

Los cuestionarios y fichas técnicas de los participantes se encuentran en los anexos.

Capítulo 5. Conclusiones y lecciones aprendidas

En este capítulo se presentan las conclusiones y las lecciones aprendidas del trabajo realizado.

5.1. Conclusiones

A continuación se presentan las principales conclusiones obtenidas a partir del trabajo realizado.

- La construcción de la propuesta y los resultados del grupo focal evidencian que es posible construir un proceso para MiPyMEs que cumpla con las características de las aplicaciones web y características de un proceso de ingeniería web, debido a que el proceso cuenta con disciplinas, actividades, tareas y artefactos que abordan este tipo de características. La parte liviana de la propuesta es un elemento que no se puede comprobar teóricamente, ya que es necesario que el proceso se ejecute en un proyecto real, sin embargo el proceso cuenta con algunas prácticas ágiles (desarrollo iterativo e incremental, excelencia técnica y buen diseño y además se apoya en Scrum) que aportan al proceso para que teóricamente este pueda cumplir con esta característica.

- A partir del análisis de la información obtenida en la revisión sistemática de la literatura, se concluye que existen múltiples propuestas para soportar el desarrollo de aplicaciones web. Sin embargo estas propuestas no abordan en su totalidad las características de las aplicaciones web y las características de un proceso de ingeniería web, debido a esto los productos web pueden tener inconvenientes como: falta de seguridad, usabilidad, poco desempeño, y pueden ser difíciles de mantener.

- Aunque no se realizó la evaluación de la propuesta mediante un caso de estudio, se obtuvo una retroalimentación del grupo focal. Los aportes obtenidos para la mejora del proceso se pueden ver reflejados en la eliminación de actividades que

permitan el ingreso de nuevos requisitos en la ejecución del proceso, permitir la ejecución en paralelo de algunas actividades (crear criterio de listo, crear criterio de hecho, definir ambiente de desarrollo y pruebas, instalar y configurar los ambientes de desarrollo y pruebas, diseñar modelo navegacional, diseñar interfaces de usuario, integrar componentes elegidos para reutilizar, crear manuales de cada funcionalidad, e integrar funcionalidades), cambiar la disciplina de producción a preproducción, mejorar la gráfica de la visión general de la propuesta, y permitir la ejecución de algunas actividades solo cuando sea necesario y no en cada iteración. Todas las anteriores observaciones realizadas en el grupo focal permitieron mejorar la propuesta, dando un nivel de aceptación considerable sobre la formulación teórica del proceso planteado.

- Aunque el proceso aquí presentado podría soportar Scrum, no está limitado a esta metodología. La propuesta es capaz de adaptarse a otras metodologías de gestión que tenga en cuenta, la gestión de requerimientos, desarrollos en paralelo, ciclos de desarrollo cortos y liberaciones con fechas establecidas. Estas características son necesarias debido a que ayudan al proceso propuesto a cumplir con el desarrollo de aplicaciones web teniendo en cuenta las características de las aplicaciones web actuales y las características del proceso de ingeniería web.

5.2. Lecciones aprendidas

En la construcción de un proceso para la creación de aplicaciones web se deben tener en cuenta las características propias de este tipo de tecnologías. Cada tipo de software posee propiedades particulares, las cuales deben ser consideradas para obtener resultados satisfactorios al momento de crear aplicaciones con la ayuda de un proceso de desarrollo.

Definir un proceso de desarrollo de aplicaciones web no es tarea fácil. Las aplicaciones web necesitan adaptarse y evolucionar con los cambios que se presentan al pasar el tiempo, los desarrollos cada vez son más exigentes, más complejos y tienen un tiempo muy limitado. Dependiendo de la empresa los recursos disponibles pueden variar, con lo cual se deben tener en cuenta diferentes factores tanto de las empresas como de las aplicaciones web.

Las micro, pequeñas y medianas empresas en ocasiones no cuentan con el personal necesario para cubrir todos los roles que sugiere un proceso. Por lo cual, para la ejecución de una propuesta como la planteada en este documento se deben buscar alternativas como por ejemplo, que una persona pueda desempeñar varios roles durante la ejecución de un proyecto.

El proceso debe permitir realizar desarrollos rápidos y de calidad, ya que un recurso muy importante en la industria es el tiempo. Los procesos tienen un flujo de ejecución para las actividades, se debe analizar qué actividades se ejecutan en

paralelo y cuales son secuenciales, para agilizar más la aplicación del proceso en una iteración.

Para poder construir un proceso ágil que ayude a la creación de aplicaciones web, se debe tener en cuenta elementos estructurales como: actividades, tareas y artefactos. Estos elementos son la guía que permite llevar a cabo el desarrollo de aplicaciones web, ya que indican el paso a paso a seguir durante la ejecución de la propuesta.

No es posible definir un proceso genérico que pueda ser utilizado en cualquier tipo de desarrollo de software. Cada desarrollo tiene propiedades que no son comunes en la construcción de aplicaciones, dependiendo de la aplicación se deben tener en cuenta una serie de elementos particulares.

Bibliografía

- [1] S. C. Lee y A. I. Shirani, A component based methodology for Web application development, *The Journal of Systems and Software*, pp. 177-187, 2004.
- [2] G. A. Edgar J., Tecnologías de información y comunicación, e innovación en las MIPYMES de Colombia, *Cuadernos de Administración*, Vol. 30, 2014.
- [3] W. Lee, S. Park, K. Lee, C. Lee, B. Lee, W. Jung, T. Kim, H. Kim y C. Wu, Agile Development of Web Application by Supporting Process Execution and Extender UML Model, de 12th Asia-Pacific Software Engineering Conference, Taipei, Taiwan, 2005.
- [4] J. I. Becerril, A. Lumbreras Sotomayor y A. R. Duk Sánchez, Web evolution: An analysis of its impact in performance on small business in distrito feredal, *ESDAI*, 2015.
- [5] J. Yepes, C. Pardo, O. Gómez, Estado del arte de la utilización de metodologías ágiles y otros modelos en pymes de software, 2016.
- [6] G. Kappel, B. Pröll, S. Reich y W. Retschitzegger, *Web Engineering. The Discipline of Systematic Development of Web Applications*, Heidelberg: John Wiley & Sons, pp. 23-28, 197-218, 2006.
- [7] A. McDonald y R. Welland, *Web Engineering in Practice*, Proc. of the 4th Workshop on Web Engineering (held in conjunction with the 10th international conference on WWW), Hong Kong, 2001.
- [8] P. Grünbacher, R. Ramler, W. Retschitzegger y W. Schwinger, Making Quality a First-Class Citizen in Web Engineering, Proc. of the 2nd Workshop on Software Quality (SQW), held in conjunction with the 26th International Conference on Software Engineering (ICSE 2004), Edinburgh, Scotland, 2004.
- [9] S. Casteleyn, F. Daniel, P. Dolog y M. Matera, *Engineering web applications*, Italia: Springer, p. 4, 2009.
- [10] P. Cáceres y E. Marcos, *Procesos ágiles para el Desarrollo de aplicaciones web*, Móstoles, Madrid, 2001.
- [11] S. Hastie y S. Wojewoda, Standish Group 2015 Chaos Report - Q&A with Jennifer Lynch, 04 Octubre 2015, [En línea]. Available: <https://goo.gl/yuo39K>. [Último acceso: 12 Agosto 2017]
- [12] C. Gonzales Serrano y F. J. Pino, Focus group como proceso en ingeniería de software: una experiencia desde la práctica, *Dyna*, vol. 80, pp. 51-60, 2013.
- [13] S. Juan y A. Roussos, *El focus group como técnica de investigación cualitativa*, Documento de Trabajo N° 256, Belgrano, 2010.

- [14] W.-H. T, Research methods in information systems: Using action research. In Research Methods in Information Systems, Amsterdam, 1985.
- [15] R. Pressman, Ingeniería del Software: Un Enfoque Práctico, 2010.
- [16] Sommerville Ian, Ingeniería del Software, 2005.
- [17] Saavedra G., María L., Hernández C., Yolanda, Caracterización e importancia de las MIPYMES en Latinoamérica: Un estudio comparativo, Actualidad Contable Faces [en línea] 2008, 11 (Julio-Diciembre) : [Fecha de consulta: 1 de julio de 2019] Disponible en:<<http://www.redalyc.org/articulo.oa?id=25711784011>> ISSN 1316-8533.
- [18] R. Sinha, Prácticas ágiles-Desarrollo de software con un enfoque ágil, Centro de Conocimiento del PMI, 2010.
- [19] Jalali, Samireh & Wohlin, Claes., Agile Practices in Global Software Engineering - A Systematic Map. Proceedings - 5th International Conference on Global Software Engineering.
- [20] H. C. José, L. P. P. Carmen, Metodologías Ágiles en el Desarrollo de Software, JISBD, 2003.
- [21] D. Schwabe y G. Rossi, An Object Oriented Approach to Web-Based Applications Design, Theory and Practice of Objects Systems, vol. 4, pp. 207-225, 1998.
- [22] D. Howcroft y J. Carroll, A Proposed Methodology for Web Development, de European Conference on Information Systems (ECIS), Salford, 2000.
- [23] P. Cáceres y E. Marcos, Las Metodologías de Desarrollo y la Mejora de Calidad de las Aplicaciones Web, de QuaTIC, Madrid, 2001.
- [24] J. Zhang y J.-Y. Chung, Mockup-driven fast-prototyping methodology for Web application development, Software - Practice and Experience,,pp. 1251-1272, 2003.
- [25] S. C. Lee y A. I. Shirani, A component based methodology for Web application development, The Journal of Systems and Software, pp. 177-187, 2004.
- [26] T. Tammet, H.-M. Haav, V. Kadarpiik y M. Kääramees, A Rule-Based Approach to Web-Based Application Development, de 7th International Baltic Conference on Databases and Information Systems, Estonia, 2006.
- [27] H. Kniberg, Scrum y XP desde las trincheras, 2007.
- [28] H. Altarawneh y A. E. Shiekh, A Theoretical Agile Process Framework for Web Applications Development in Small Software Firms, de Sixth International Conference on Software Engineering Research, Management and Applications, 2008.

- [29] T. Teraoka, M. Osaki, S. Yanagihara y H. Nakata, A Page-Centric Approach to Web Application Development, Electronics and Communications in Japan, vol. 92, 2009.
- [30] D. A. Silva y M. Bárbara, Construyendo aplicaciones web con una metodología de diseño orientada a objetos, Revista Colombiana de Computación – RCC, vol. 2, 2010.
- [31] J. C. Castrejon Castillo, R. López Landa y R. Lozano, Model2Roo: A Model Driven Approach for Web Application Development based on the Eclipse Modeling Framework and Spring Roo, CONIELECOMP 2011 - International Conference on Electrical Communications and Computers, pp. 82-87, 2011.
- [32] A. M. French, Web Development Life Cycle: A New Methodology for Developing Web Applications, 2011, [En línea]. Available: <https://goo.gl/VGSrdF>. [Último acceso: 15 Julio 2017].
- [33] M. S. Arshiya, Development of Object-Oriented Analysis and Design Methodology for Secure Web Applications, International Journal of Computer Science and Mobile Computing, vol. 4, pp. 794-803, 2015.
- [34] M. M. Charaf Eddine y O. Kazar, Web Applications Development by Formal Refinement Approach, International Journal of Software Engineering and Its Applications, vol. 9, pp. 73-98, 2015.
- [35] D. Clerissi, M. Leotta, G. Reggio y F. Ricca, Test Driven Development of Web Applications: a Lightweight Approach, de 2016 10th International Conference on the Quality of Information and Communications Technology, Italia, 2016.
- [36] P. L. Castillo Asencio, Desarrollo e implementación de un sistema web para generar valor en una pyme aplicando una metodología ágil. Caso de estudio: Manufibras Perez SRL, Lima, 2016.
- [37] L. Adriano, Pruebas funcionales automatizadas para aplicaciones Web: Usando Selenium para aplicar pruebas de regresión automatizadas, 2018.
- [38] C. Paloma, M. Esperanza, Las metodologías de desarrollo y la mejora de calidad de las aplicaciones web, QuaTIC, 2001.
- [39] C. Alquicira, R. Anaya , P. Angeleri , J. Hurtado y L. Flores, Competisoft: Mejora de Procesos para Fomentar la Competitividad de la Pequeña y Mediana Industria del Software de Iberoamérica, 2008.
- [40] Calvache, César & Garcia, Felix & Piattini, Mario & Pino, Francisco & Baldassarre, Maria, A Reference Ontology for Harmonizing Process-reference Models, Revista Facultad de Ingeniería, pag. 29-42.
- [41] <https://netxus.org/apple-y-otras-empresas-que-usan-scrum/>.
- [42] J. Kontio, L. Lehtola y J. Bragge, Using the focus group method in software

engineering: obtaining practitioner and user experiences, de Proceedings. 2004 International Symposium on Empirical Software Engineering, 2004. ISESE '04, Redondo Beach, CA, USA, USA, 2004.

Anexo A. Agenda de trabajo

Agenda de trabajo

La sesión de debate es coordinada por el moderador y el supervisor, y efectuada por los participantes. Para ello, se hace uso de los artefactos obtenidos durante la etapa de planeación. A continuación, se describe la agenda llevada a cabo para la sesión.

N°	Descripción	Hora	
		Desde	Hasta
1	Agradecimiento a los participantes por la asistencia.	4:00 pm	4:05 pm
2	Presentación del investigador por parte del supervisor. <ul style="list-style-type: none">• Explicar a los participantes cuál será el proceso a seguir con los resultado obtenidos.• Explicar a los participantes el por qué fueron seleccionados.	4:05 pm	4:10 pm
3	Presentación de la propuesta.	4:10 pm	4:35 pm
4	Dar espacio a los participantes para dar hacer sus apreciaciones sobre las actividades contenidas en el proceso propuesto.	4:35 pm	5:00 pm
5	Los participantes dan respuesta a la encuesta definida por el investigador.	5:00 pm	5:10 pm
6	Los participantes llenan la ficha de asistencia.	5:10 pm	5:15 pm
7	Refrigerio.	5:15 pm	5:20 pm
8	Agradecimientos por parte del investigador a los participantes.	5:20 pm	5:25 pm
9	Cierre de sesión.	5:25 pm	5:30 pm

Anexo B. Cuestionario de preguntas sobre el proceso utilizado en el grupo focal.

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Tema: Evaluación de proceso de desarrollo de software: Proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Marque con una X la opción que considere adecuada y argumente en caso de ser necesario.

1. De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?
 1. Totalmente en desacuerdo
 2. En desacuerdo
 3. Neutral
 4. De acuerdo
 5. Totalmente de acuerdo

Observaciones

2. ¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

Observaciones

3. ¿Considera que las actividades propuestas en las fases del proceso son apropiadas y aplicables a un entorno empresarial para el desarrollo de aplicaciones web?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

Observaciones

4. ¿Considera el proceso propuesto como ágil?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

Observaciones

5. ¿Considera que al proporcionar el proceso a un equipo de trabajo puede ser fácilmente entendible y aplicable?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

Observaciones

6. ¿Considera que al proceso le falta alguna actividad importante en el desarrollo de aplicaciones web?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

¿Cuáles?

7. ¿Considera importante las pruebas automatizadas en el desarrollo de aplicaciones web?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

Observaciones

8. ¿Considera importante el desarrollo dirigido por pruebas en la construcción de aplicaciones web?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

Observaciones

9. ¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?

1. Totalmente en desacuerdo
2. En desacuerdo
3. Neutral
4. De acuerdo
5. Totalmente de acuerdo

Observaciones

Anexo C. Ficha de asistencia al grupo focal.

FICHA DE ASISTENCIA AL GRUPO FOCAL

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Tema: Evaluación del proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Datos del participante

Nombre

Profesión/Ocupación

Experiencia relacionada a ingeniería de software, metodologías ágiles y desarrollo web.

Perfil: Nivel de escolaridad, estudios realizados relacionados con la ingeniería de software y metodologías ágiles.

Anexo D. Encuestas diligenciadas grupo focal

Encuesta 1.

**PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL
DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS
EMPRESAS**

Tema: Evaluación de proceso de desarrollo de software: Proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Marque con una X la opción que considere adecuada y argumente en caso de ser necesario.

1. De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones
La organización de los procesos es un factor clave para el éxito de un proyecto y la satisfacción de los clientes.

2. ¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones
Considerando ajustes, totalmente de acuerdo.

3. ¿Considera que las actividades propuestas en las fases del proceso son apropiadas y aplicables a un entorno empresarial para el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Si en su mayoría, lo importante es que las actividades sean claras, precisas y entendibles para quien las debe ejecutar.

4. ¿Considera el proceso propuesto como ágil?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

El considerar a SCRUM como pilar ayuda, sin embargo debe revisarse la iteratividad dentro de algunas fases.

5. ¿Considera que al proporcionar el proceso a un equipo de trabajo puede ser fácilmente entendible y aplicable?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Con un proyecto piloto y un equipo propositivo, seguramente sí.

6. ¿Considera que al proceso le falta alguna actividad importante en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

¿Cuáles?

Revisar las observaciones durante la sesión del Focus Group.

7. ¿Considera importante las pruebas automatizadas en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Todos los comentarios que permitan la gestión y logro de buenos resultados son bienvenidos.

8. ¿Considera importante el desarrollo dirigido por pruebas en la construcción de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Es una buena práctica a tener en cuenta.

9. ¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Estas características no fundamentales deberían estar
inmersas en cualquier proceso a proponer.

Encuesta 2.

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Tema: Evaluación de proceso de desarrollo de software: Proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Marque con una X la opción que considere adecuada y argumente en caso de ser necesario.

1. De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Los procesos son importantes en todo tipo de proyectos, las mipymes no son una excepción.

2. ¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Hay que tener en cuenta que en ocasiones las mipymes no cuentan con recursos para la implementación de procesos de desarrollo organizados, por lo tanto, se sugiere que sea más sencillo de entender y aplicar.

3. ¿Considera que las actividades propuestas en las fases del proceso son apropiadas y aplicables a un entorno empresarial para el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

4. ¿Considera el proceso propuesto como ágil?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Durante la presentación no se evidenció la naturaleza iterativa e incremental. Al contrario fue muy lineal. Se sugiere construir un modelo visual, o una gráfica que presente, de forma global el proceso y la naturaleza iterativa.

5. ¿Considera que al proporcionar el proceso a un equipo de trabajo puede ser fácilmente entendible y aplicable?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Tiene muchas actividades que requieren explicación. Quitara el proceso necesita una visión de alto nivel en la que no se incluyen todos los detalles. Y que los detalles presentados estén en un nivel más profunda.

6. ¿Considera que al proceso le falta alguna actividad importante en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

¿Cuáles?

No hay ejemplos sugeridos, plantillos o patrones que
se reutilizan de forma frecuente.

7. ¿Considera importante las pruebas automatizadas en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

8. ¿Considera importante el desarrollo dirigido por pruebas en la construcción de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

9. ¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

El proceso hace énfasis en los diferentes tipos de proyectos,
que son los que garantizan la existencia de estos atributos
muy bien.

Encuesta 3.

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Tema: Evaluación de proceso de desarrollo de software: Proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Marque con una X la opción que considere adecuada y argumente en caso de ser necesario.

1. De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- Totalmente de acuerdo

Observaciones

Debido a que es importante tener procesos que actúen
alineados con el desarrollo y además la permita
enfrentar fallas, así como no implicar caídas o retrasos
largos a sus clientes con cuando suya actividad
de personal

2. ¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- Totalmente de acuerdo

Observaciones

3. ¿Considera que las actividades propuestas en las fases del proceso son apropiadas y aplicables a un entorno empresarial para el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones
Este tipo de procesos se puede llevar a otros modelos como autorouting y desarrollados a la medida.

4. ¿Considera el proceso propuesto como ágil?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones
Considero que es importante el que se consideren los roles de origen

5. ¿Considera que al proporcionar el proceso a un equipo de trabajo puede ser fácilmente entendible y aplicable?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

6. ¿Considera que al proceso le falta alguna actividad importante en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

¿Cuáles?

Considero que hace falta en ambiente de pro-
ducción posterior al testing, además representar
de forma clara al modelo global con el fin de
detectar el ciclo de scrum

7. ¿Considera importante las pruebas automatizadas en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

8. ¿Considera importante el desarrollo dirigido por pruebas en la construcción de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Para poder implicar la participación de más personas
en el desarrollo

9. ¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Encuesta 4.

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Tema: Evaluación de proceso de desarrollo de software: Proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Marque con una X la opción que considere adecuada y argumente en caso de ser necesario.

1. De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?
- a. Totalmente en desacuerdo
 - b. En desacuerdo
 - c. Neutral
 - d. De acuerdo
 - e. Totalmente de acuerdo

Observaciones

Estoy de acuerdo, sin embargo puede ser una metodología extensible a empresas más grandes en un futuro.

2. ¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Proponer tamaños de equipos suena importante.

3. ¿Considera que las actividades propuestas en las fases del proceso son apropiadas y aplicables a un entorno empresarial para el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

— 0 —

4. ¿Considera el proceso propuesto como ágil?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Pregunta interesante. suena bien en papel,
pero es necesario tener casos de éxito a
futuro, para determinar si es "ágil".

5. ¿Considera que al proporcionar el proceso a un equipo de trabajo puede ser fácilmente entendible y aplicable?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

— 0 —

6. ¿Considera que al proceso le falta alguna actividad importante en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

¿Cuáles?

Posiblemente la automatización del despliegue, pero eso ya se mencionó en los comentarios.

7. ¿Considera importante las pruebas automatizadas en el desarrollo de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Ahorra mucho tiempo a largo plazo.

8. ¿Considera importante el desarrollo dirigido por pruebas en la construcción de aplicaciones web?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Mayormente las de UX, que suelen ser más ajenas a los desarrolladores.

9. ¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

Siempre y cuando se cumplan los criterios
de hecho y aceptación, sí.

Encuesta 5.

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Tema: Evaluación de proceso de desarrollo de software: Proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Marque con una X la opción que considere adecuada y argumente en caso de ser necesario.

1. De acuerdo a su experiencia ¿Considera importante un proceso de desarrollo de aplicaciones web en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

2. ¿Considera que el proceso propuesto puede ser aplicado en micro, pequeñas y medianas empresas?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

le falta bastante al proceso, se deben
pulsar terminos y fases y actividades

9. ¿Considera que una aplicación web desarrollada siguiendo las actividades del proceso cumple con atributos de calidad como: Seguridad, usabilidad, desempeño y mantenibilidad?

- a. Totalmente en desacuerdo
- b. En desacuerdo
- c. Neutral
- d. De acuerdo
- e. Totalmente de acuerdo

Observaciones

No necesariamente por usar el proceso se
va a cumplir con estos criterios,

Anexo E. Ficha de asistencia al grupo focal diligenciada

FICHA DE ASISTENCIA AL GRUPO FOCAL

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Tema: Evaluación del proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Datos del participante

Nombre

Victor Zapata Bedoya

Profesión/Ocupación

Desarrollador web.

Experiencia relacionada a ingeniería de software, metodologías ágiles y desarrollo web.

Participación en diversos proyectos web, desde pequeños sitios y aplicaciones web de baja concurrencia hasta proyectos web que deben permitir la interacción de miles de usuarios. En estos proyectos, el uso de SCRUM es fundamental para el seguimiento de las actividades del proyecto y su éxito.

Perfil: Nivel de escolaridad, estudios realizados relacionados con la ingeniería de software y metodologías ágiles.

Ingeniero de sistemas especializado en la administración de la información y bases de datos.
Certificación en SCRUM.

FICHA DE ASISTENCIA AL GRUPO FOCAL

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL
DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS
EMPRESAS

Tema: Evaluación del proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Datos del participante

Nombre

YAISSON FABIAN SAMBONI ZUÑIGA

Profesión/Ocupación

Ing. Sistemas

Experiencia relacionada a ingeniería de software, metodologías ágiles y desarrollo web.

Desarrollador de sitios web como: www.sumercado.co, www.marksoftware.co. Creador del proceso COS-PH para el desarrollo de software en ambientes universitarios.

Perfil: Nivel de escolaridad, estudios realizados relacionados con la ingeniería de software y metodologías ágiles.

Ing. Sistemas, tecnólogo en telecomunicaciones

FICHA DE ASISTENCIA AL GRUPO FOCAL

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL
DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS
EMPRESAS

Tema: Evaluación del proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Datos del participante

Nombre

Julian Darío Bermúdez Trujillo

Profesión/Ocupación

Ing. Sistemas / Gerente Sunset Software House SAS

Experiencia relacionada a ingeniería de software, metodologías ágiles y desarrollo web.

- DBA
- Analista Desarrollador
- Consultor en Calidad de Software (ISO 9126/25000)
- Scrum Master
- Dir. de Proyecto
- Gerente empresa de desarrollo de Software.

Perfil: Nivel de escolaridad, estudios realizados relacionados con la ingeniería de software y metodologías ágiles.

Especialista en desarrollo de Software
Ingeniero de Sistemas
SCRUM Master Certificado

FICHA DE ASISTENCIA AL GRUPO FOCAL

PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL
DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS
EMPRESAS

Tema: Evaluación del proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Datos del participante

Nombre

Heiver Andrés Zúñiga Curián

Profesión/Ocupación

Est. Ing de sistemas / Analista desarrollador

Experiencia relacionada a ingeniería de software, metodologías ágiles y desarrollo web.

Experiencia en entornos laborales, con la construcción de arquitecturas, aplicación de metodologías de trabajo y desarrollo, así como liderazgo de proyectos.

Perfil: Nivel de escolaridad, estudios realizados relacionados con la ingeniería de software y metodologías ágiles.

Universitario, formado en el concepto de scrum y del marco sprint, así como de XP

Anexo F. Primera versión del proceso propuesto

Definición del proceso

1. Alcance del proceso

Este proceso está dirigido a micro, pequeñas y medianas empresas, las cuales desean tener una guía de construcción de aplicaciones web, que les permita considerar los diferentes factores a tener en cuenta al momento del desarrollo de un proyecto web. Está propuesta en conjunto con la metodología scrum guía a las empresas mediante una serie de fases que permiten plasmar en funcionalidades los requisitos obtenidos. En este proceso se pueden ver fases como análisis, diseño, implementación, pruebas y despliegue, las cuales permiten la construcción de un producto web que cumpla con las características que deben tener este tipo de aplicaciones. Para la construcción del proceso se utiliza como base la guía de COMPETISOF.

2. Plantilla para la definición del proceso

Para la definición del proceso, se utiliza como base el patrón de procesos propuesto por COMPETISOF. A continuación se muestra la plantilla utilizada para la definición del proceso.

Identificador	Acrónimo utilizado para la identificación única del proceso.
Proceso	Nombre del proceso, precedido por el acrónimo establecido en la definición de los elementos de la estructura del modelo de procesos.

Categoría	Nombre de la categoría a la que pertenece el proceso.	
Propósito	Objetivos generales de las actividades y productos que componen el flujo de trabajo del proceso.	
Descripción	Descripción general de las actividades y productos que componen el flujo de trabajo del proceso.	
Objetivos	Objetivos específicos cuya finalidad es asegurar el cumplimiento del propósito del proceso. Los objetivos se identifican como O1, O2, etc.	
Responsabilidad y autoridad	Responsable es el rol principal responsable por la ejecución del proceso. Autoridad es el responsable por validar la ejecución del proceso y el cumplimiento del propósito.	
Procesos relacionados	Nombre de procesos relacionados.	
Roles involucrados y competencias		
Abreviatura	Rol	Competencias
Abreviatura del rol	Nombre del rol	Descripción de las responsabilidades de cada rol.
Rol Acrónimo del rol encargado de realizar la(s) tarea(s).	Tareas Descripción de las tareas que corresponden a una actividad.	
Nombre de la fase		
Entradas	Nombre de los artefactos que se toman como entrada para la actividad.	
Roles involucrados	Descripción de la tarea.	
Salidas	Descripción de los artefactos producidos en la actividad.	
Diagrama de flujo: Diagrama en el cual se puede visualizar el flujo que tiene la actividad.		

3. Términos genéricos

En la siguiente tabla se da una corta descripción de los términos utilizados para la definición del proceso.

Término	Descripción
Proceso	“Un proceso es un conjunto de actividades, acciones y tareas que se ejecutan cuando va a crearse algún producto del trabajo ”[Ingeniería del software, un enfoque práctico Pressman 7 edición]
Actividad	Una actividad siempre tiene como fin cumplir con un propósito de un tamaño considerado, y es realizada sin importar el ámbito del proyecto en la cual se vaya a desarrollar.[Ingeniería del software, un enfoque práctico Pressman 7 edición]
Tarea	A diferencia de la actividad, la tarea se centra en un propósito pequeño, el cual está totalmente claro. [Ingeniería del software, un enfoque práctico Pressman 7 edición]
Rol	“Describe un conjunto o grupo de responsabilidades, deberes y habilidades requeridas para realizar un actividad” [A reference ontology for harmonizing process-reference models]
Herramienta	Herramientas software que automatizan la ejecución de ciertas actividades.

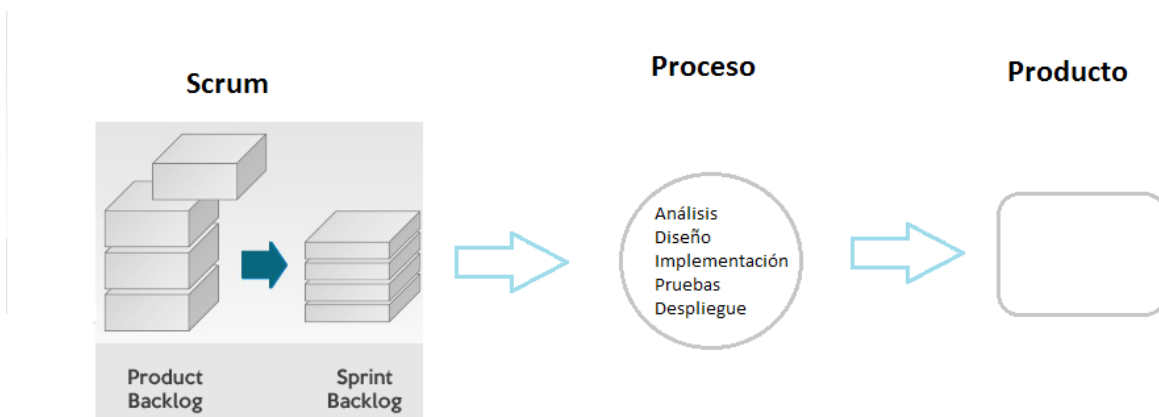
4. Identificación de roles

Para la definición de un proceso es necesaria la existencia de una serie de roles, para la distribución de tareas de acuerdo a las habilidades que posea cada participante del equipo de proyecto. Durante la creación del proceso se identificaron diferentes roles que aportan a las fases que conforman la propuesta para el desarrollo de aplicaciones web.

Los roles que se presentan en el proceso no necesariamente deben ser desarrollados cada uno por una persona. Para aquellos equipos de trabajo o empresas que no pueden tener una persona por rol, se aconseja que una persona pueda asumir más de un rol

5. Visión general de la propuesta

La propuesta ha sido diseñada para permitir el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas. Este proceso es el complemento de scrum en cuanto al desarrollo. En la figura 1 se presenta la visión general de la propuesta.



6. Caracterización del proceso ágil de desarrollo de software para soportar el desarrollo de aplicaciones web en micro, pequeñas y medianas empresas.

Proceso	Desarrollo de aplicaciones web
Categoría	Operación
Propósito	El propósito de este proceso de software es la creación de aplicaciones web teniendo en cuenta las necesidades especificadas en los requisitos obtenidos, siguiendo una serie de etapas como lo son: análisis, diseño, implementación, pruebas y despliegue.
Descripción	<p>En el proceso de desarrollo de aplicaciones web se pueden realizar una o más iteraciones dependiendo de las necesidades de cada cliente. Al iniciar el proyecto se debe ejecutar la siguiente fase:</p> <ul style="list-style-type: none"> • Inicio: Elegir el personal necesario para la realización del proyecto, socializar los objetivos y alcance del proyecto a realizar. <p>Una vez realizada la fase de inicio, se desarrolla de manera iterativa e incremental. Para cada iteración se tienen en cuenta las siguientes fases:</p> <ul style="list-style-type: none"> • Análisis: Esta fase involucra una serie de actividades dentro de las cuales se encuentran: la obtención de los requisitos del producto, creación de historias de usuario, creación de prototipos y creación de las pruebas. Las salidas de estas actividades desempeñan un papel fundamental en el desarrollo exitoso de un producto. • Diseño: En esta fase se plantea la arquitectura general del software, lo cual involucra el diseño de datos, diseño arquitectónico, diseño de la interfaz de usuario, entre otros. <ul style="list-style-type: none"> • La arquitectura elegida debe obedecer a atributos de calidad como: I) Seguridad, II) Desempeño, III) Usabilidad y IV) Mantenibilidad.

	<ul style="list-style-type: none"> • Implementación: Toma como entradas los artefactos creados en la fase de análisis y de diseño, para obtener como salida un producto que satisfaga las necesidades del clientes y demás interesados. • Pruebas: Se toma como entrada las pruebas creadas en la fase de análisis. Con las pruebas escritas se verifica el correcto funcionamiento de la aplicación web, para garantizar la satisfacción del cliente. • Despliegue: Se realiza la configuración e instalación de las funcionalidades con criterio de listo.
Objetivos	<ol style="list-style-type: none"> 1. Lograr que el equipo de trabajo pueda entender las necesidades del cliente. 2. Lograr que los productos de salida sean consistentes con los productos de entrada en cada iteración realizada, mediante actividades como: verificación, validación, pruebas y control de los cambios realizados al producto. 3. Permitir a las Mipymes contar con una guía para el desarrollo de aplicaciones web, de forma que el sistema final cumpla con los requisitos especificados.
Responsabilidad y autoridad	<p>Responsable: Responsable de desarrollo de software Autoridad: Responsable de la administración del proyecto</p>

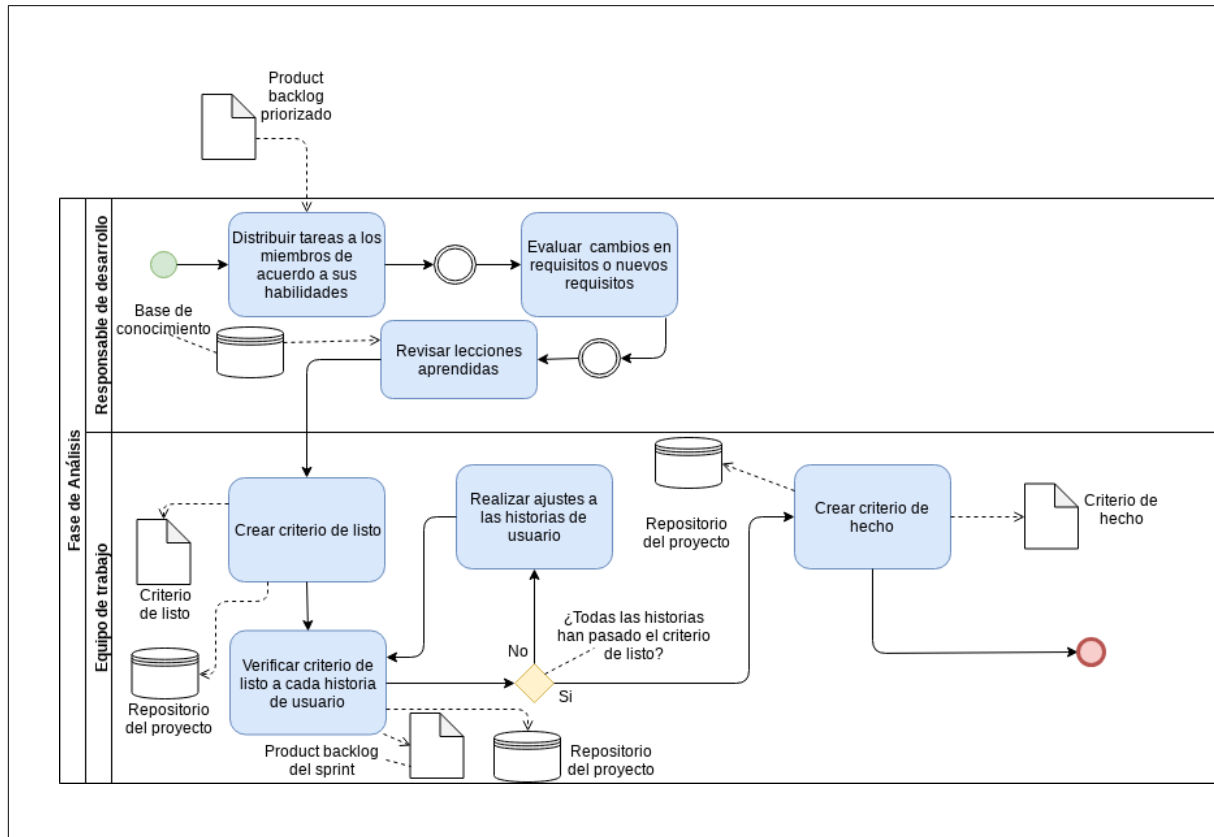
Roles involucrados		
Abreviatura	Rol	Competencias
RAP	Responsable de la administración del proyecto.	Persona encargada de la planificación, ejecución y control del proyecto.

CL	Cliente	Conocimiento en la expedición de Solicitudes de Cambios.
US	Usuario	Persona o personas que operan o interactúan directamente con el Sistema de Software.
AN	Analista	Conocimiento y experiencia en la obtención, especificación y análisis de los requisitos.
AR	Arquitecto	Conocimiento de la plataforma tecnológica objetivo, conocimiento de los recursos existentes que pueden ser reutilizados, visión global del negocio y de las soluciones de arquitectura que garantizan la evolución del sistema.
DI	Diseñador	Conocimiento y experiencia en el diseño de la estructura de los componentes de software.
DU	Diseñador de interfaz de usuario	Conocimiento en diseño de interfaces de usuario y criterios ergonómicos.
DE	Desarrollador/Programador	Conocimiento y/o experiencia en la programación, integración y pruebas unitarias.
EP	Equipo de pruebas	Conocimiento y experiencia en la planificación y realización de pruebas de integración y de sistema.
RD	Responsable de desarrollo	Conocimiento y experiencia en el desarrollo de software.

ET	Equipo de Trabajo	Conocimiento y experiencia de acuerdo a su rol.
IO	Ingeniero de operaciones	

Rol	Descripción
A1. Fase de Análisis	
Entradas	<ul style="list-style-type: none"> • Product backlog priorizado.
RD ET	<p>A1.1 Distribuir tareas a los miembros de acuerdo a sus habilidades.</p> <ul style="list-style-type: none"> • Identificar roles en el equipo. • Identificar tareas a realizar. • Asignar tareas a cada rol.
ET CL	<p>A1.2. Crear criterio de listo.</p> <ul style="list-style-type: none"> • Realizar reunión con el equipo. • Escuchar items a incluir en el criterio. • Analizar cada item y discutir si se incluye o no. • Construir la lista de items a evaluar.
ET CL	<p>A1.3. Verificar criterio de listo a cada historia de usuario.</p> <ul style="list-style-type: none"> • Elegir una historia de usuario. • Verificar el cumplimiento de cada item en la historia de usuario. • Escribir los ajustes a realizar en la historia de usuario. • Repetir los tres pasos anteriores hasta que se hayan verificado todas las historias de usuario.

	A1.4. Si alguna historia de usuario no ha pasado el criterio de listo ir a la actividad A1.5., de lo contrario continuar con la actividad A1.6.
ET CL	A1.5. Realizar ajustes a las historias de usuario.
ET CL	A1.6. Crear criterio de hecho. <ul style="list-style-type: none"> • Realizar reunión con el equipo. • Escuchar items a incluir en el criterio. • Analizar cada item y discutir si se incluye o no. • Construir la lista de items a evaluar.
Salidas	<ul style="list-style-type: none"> • Criterio de listo. • Criterio de hecho. • Product backlog del sprint. •

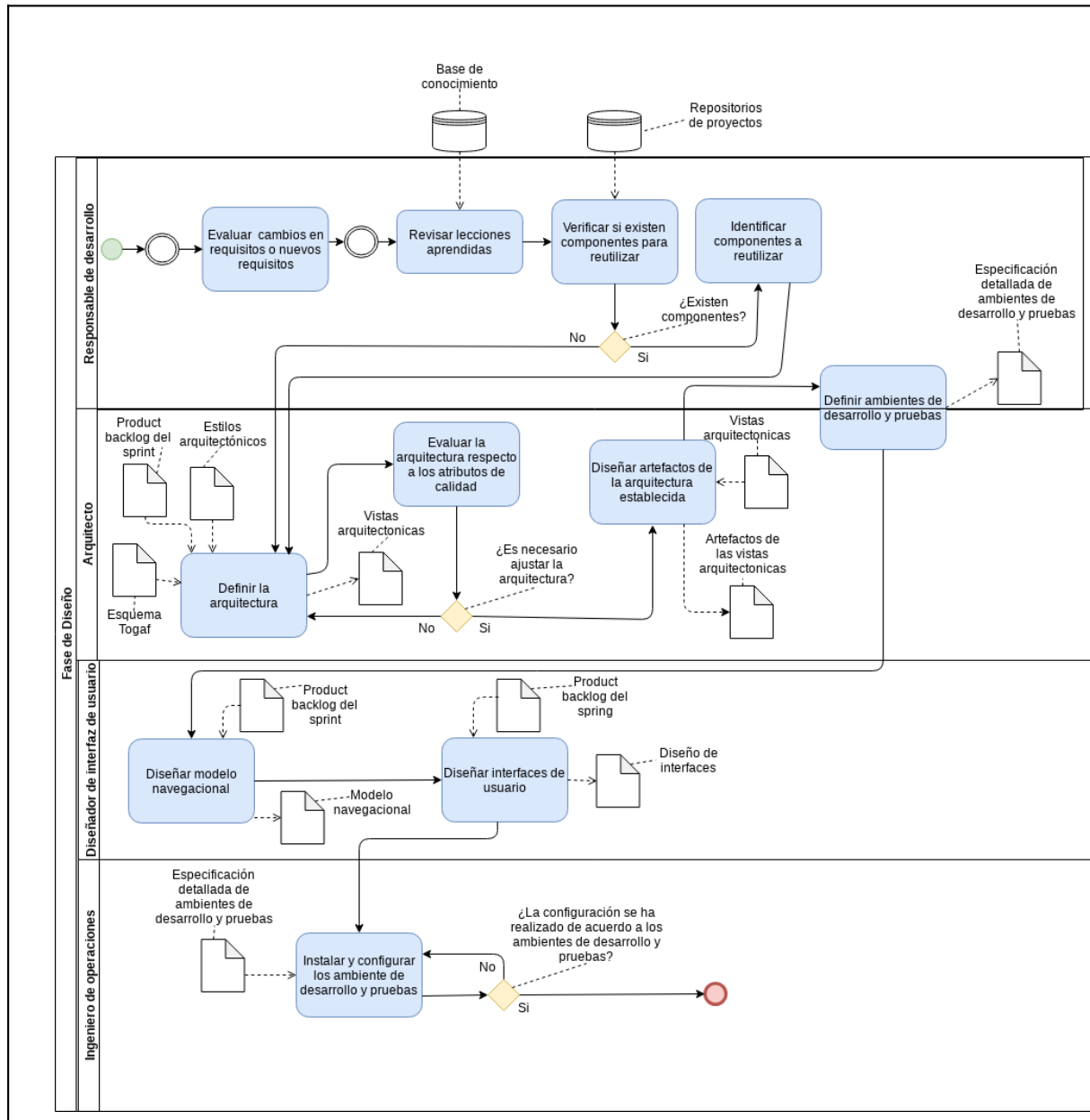


Rol	Descripción
A2. Fase de Diseño	
Entradas	<ul style="list-style-type: none"> • Product backlog del sprint. • Esquema togaf. • Estilos arquitectónicos.
RD DE	A2.1 Verificar si existen componentes para reutilizar.
	A2.2. Si existen componentes a reutilizar ir a la actividad A2.3, de lo contrario continuar con la actividad A2.4.

<p>RD DE</p>	<p>A2.3. Identificar componentes a reutilizar.</p> <ul style="list-style-type: none"> • Hacer un listado de los componentes disponibles que pueden ayudar al desarrollo del proyecto. • Elegir los componentes que aportan valor al desarrollo del proyecto.
<p>AR</p>	<p>A2.4. Definir la arquitectura.</p> <ul style="list-style-type: none"> • Realizar un listado de posibles estilos arquitectónicos teniendo en cuenta el esquema togap. • Analizar cada estilo arquitectónico identificando ventajas y desventajas de acuerdo al desarrollo a realizar. • Elegir el mejor estilo arquitectónico de acuerdo a la necesidad presentada en el proyecto. • Elegir las vistas arquitectónicas que se van a utilizar(Se recomienda utilizar como mínimo las vistas lógica, de despliegue y física). • Elegir los artefactos necesarios de cada vista arquitectónica(Para la vista de lógica se recomienda como mínimo generar el diagrama de clases, para la vista de despliegue generar el diagrama de componentes y paquetes, y por último para la vista física general el diagrama de despliegue).
<p>AR</p>	<p>A2.5. Evaluar la arquitectura respecto a los atributos de calidad.</p> <ul style="list-style-type: none"> • Verificar que el estilo arquitectónico elegido este acorde a los atributos de calidad necesarios para el desarrollo. • Verificar que las vistas arquitectónicas y sus artefactos estén acorde a los atributos de calidad necesarios para el desarrollo.

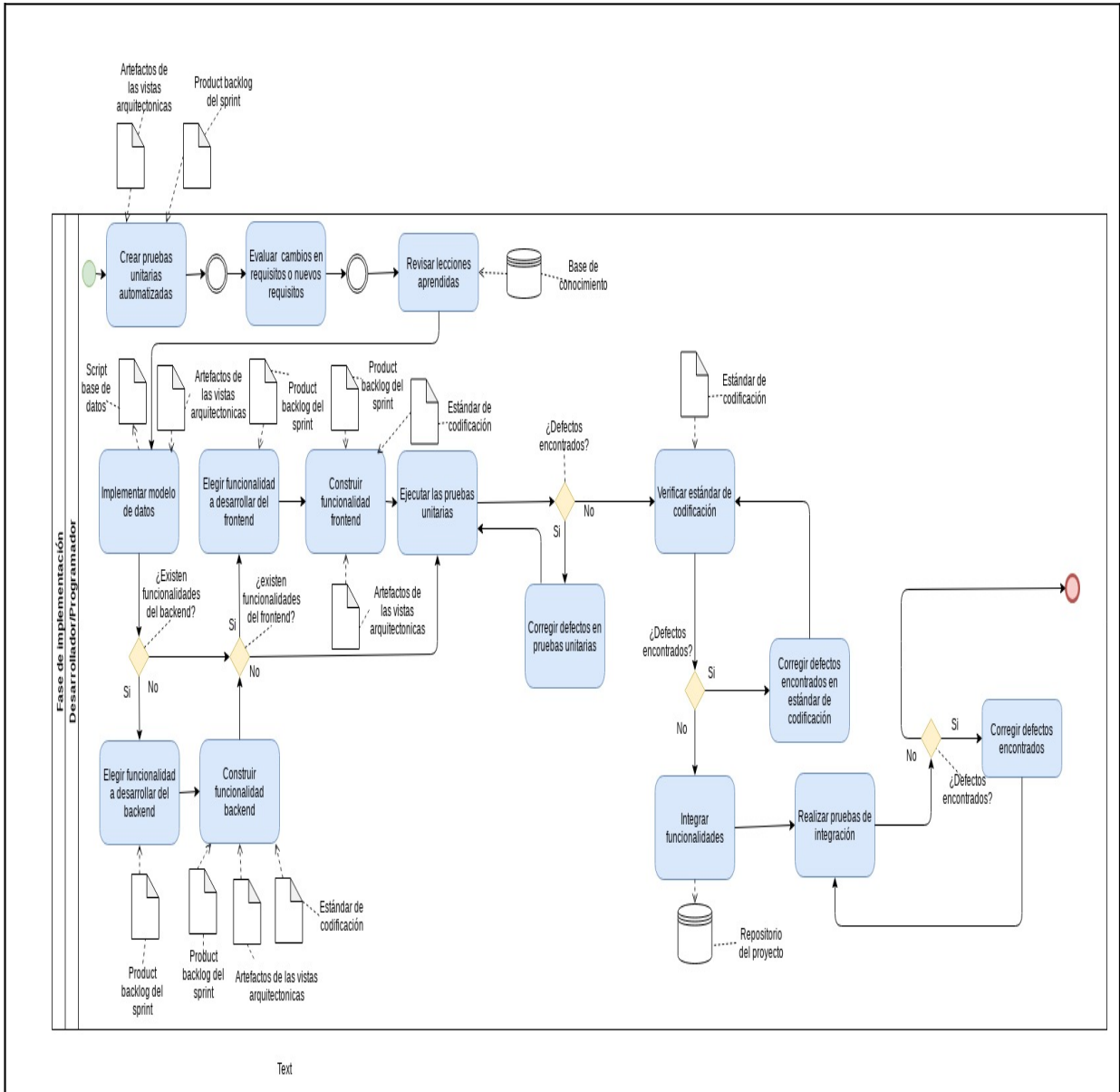
	A2.6. Si es necesario realizar algún ajuste a la arquitectura, ir a la actividad A2.4, de lo contrario continuar con la actividad A2.7.
AR	A2.7. Diseñar artefactos de la arquitectura establecida.
RD AR	A.2.8. Definir ambientes de desarrollo y pruebas. <ul style="list-style-type: none"> • Definir lenguaje de programación para el backend. • Definir lenguaje de programación para el frontend. • Definir herramienta para realizar pruebas unitarias automatizadas. • Definir herramientas para realizar pruebas de requisitos funcionales y no funcionales. • Definir herramienta para verificar el estándar de codificación. • Definir herramienta para la base de datos.
DU	A.2.9. Diseñar modelo navegacional.
DU DI	A2.10. Diseñar interfaces de usuario.
IO	A2.11. Instalar y configurar los ambiente de desarrollo y pruebas. <ul style="list-style-type: none"> • Instalar y configurar lenguaje de programación del backend. • Instalar y configurar lenguaje de programación del frontend. • Instalar y configurar herramienta para pruebas unitarias. • Instalar y configurar herramientas para realizar pruebas a requisitos funcionales y no funcionales.

	<ul style="list-style-type: none"> • Instalar y configurar herramienta para verificar el estándar de codificación. • Instalar y configurar herramienta para la base de datos.
	A2.12. Si la configuración se ha realizado de acuerdo a los ambientes de desarrollo y pruebas se termina la fase, de lo contrario ir a la actividad A2.11.
Salidas	<ul style="list-style-type: none"> • Vistas arquitectónicas. • Artefactos de las vistas arquitectónicas. • Especificación detallada de ambientes de desarrollo y pruebas. • Modelo navegacional. • Diseño de interfaces.



Rol	Descripción
A3. Fase de Implementación	
Entradas	<ul style="list-style-type: none"> • Artefactos de las vistas arquitectónicas. • Product backlog del sprint. • Estándar de codificación.
DE	<p>A3.1 Crear pruebas unitarias automatizadas.</p> <ul style="list-style-type: none"> • Definir los métodos a los cuales se van a realizar las pruebas. • Definir las pruebas a realizar en cada método. • Construir las pruebas automatizadas que se realizarán a los métodos.
DE	A3.2 Implementar modelo de datos.
DE	A3.3. Si existen funcionalidades del backend, ir a la actividad A3.4, de lo contrario ir a la actividad A3.6.
DE	A3.4. Elegir funcionalidad a desarrollar del backend.
DE	A3.5. Construir funcionalidad backend.
	A3.6. Si existen funcionalidades de frontend para realizar ir a la actividad A3.7., de lo contrario continuar con la actividad A3.9.
DE	A3.7. Elegir funcionalidad a desarrollar del frontend.
DE	A3.8. Construir funcionalidad frontend.
DE	A3.9. Ejecutar las pruebas unitarias.
	A3.10. Si se han encontrado defectos en las pruebas unitarias ir a la actividad A3.11., de lo contrario continuar con la actividad A3.12.

DE	A3.11. Corregir defectos en pruebas unitarias, luego ir a la actividad A3.9.
DE	A3.12. Verificar estándar de codificación.
	A3.13. Si se han encontrado defectos en el estándar de codificación ir a la actividad A3.14., de lo contrario continuar con la actividad A3.15.
DE	A3.14. Corregir defectos encontrados en estándar de codificación, ir a la actividad A3.12.
DE	A3.15. Integrar funcionalidades.
DE	A3.16. Realizar pruebas de integración.
	A3.17. Si existen defectos ir a la actividad A3.18, de lo contrario la fase termina.
DE	A3.18. Corregir defectos encontrados, luego ir a la actividad A3.16.
Salidas	<ul style="list-style-type: none"> • Código de funcionalidades construidas del backend. • Código de funcionalidades construidas del frontend.



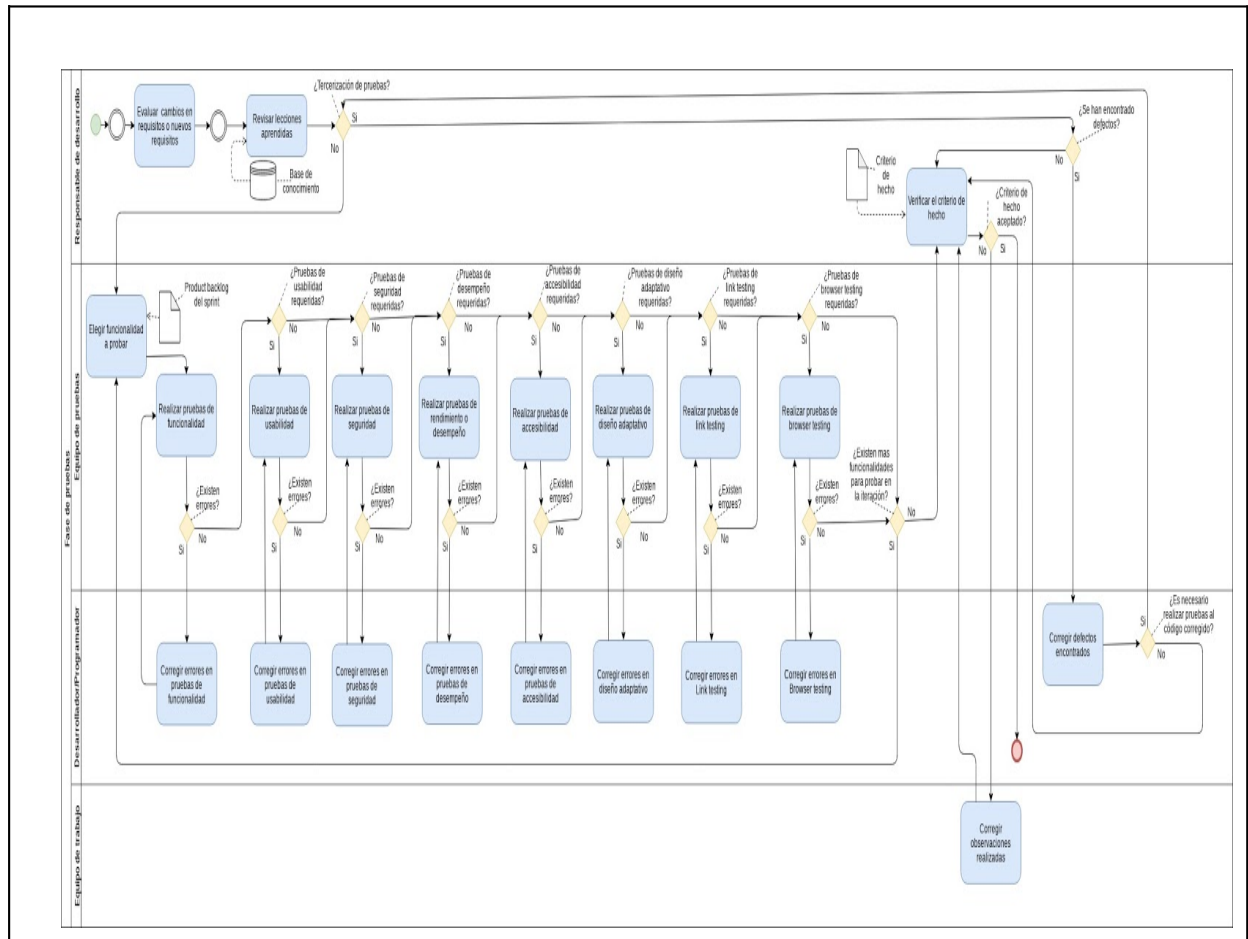
Rol	Descripción
A4. Fase de pruebas	
Entradas	<ul style="list-style-type: none"> • Código funcionalidades de la aplicación web. • Product backlog del sprint • Criterio de hecho

	A4.1 Si se realiza tercerización de pruebas ir al paso A4.2., de lo contrario continuar con la actividad A4.8.
	A4.2. Si se han encontrado defectos en el paso en la tercerización de pruebas., ir a la actividad A4.3., de lo contrario ir a la actividad A4.5.
DE	A4.3. Corregir defectos encontrados.
	A4.4. Si es necesario hacer pruebas al código corregido ir al paso A4.1., de lo contrario continuar con la actividad A4.5.
RD	A4.5. Verificar el criterio de hecho. <ul style="list-style-type: none"> • Elegir una funcionalidad. • Verificar el cumplimiento de cada item en la funcionalidad. • Escribir los ajustes a realizar en la funcionalidad. • Repetir los tres pasos anteriores hasta que se hayan verificado todas las funcionalidades.
	A4.6. Si el criterio de hecho es aceptado terminar la fase, de lo contrario ir a la actividad A4.7.
DE	A4.7. Corregir observaciones realizadas en el criterio de hecho, luego volver a la actividad A4.5.
EP	A4.8. Elegir funcionalidad a probar.
EP	A4.9. Realizar pruebas de funcionalidad. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Crear script para la ejecución de las pruebas automatizadas. • Ejecutar script. • Documentar los errores en caso de que se hayan presentado.
	A4.10. Si existen errores en las pruebas de funcionalidad ir a la actividad A4.11., de lo contrario continuar con el paso A4.12.
DE	A4.11. Corregir errores en pruebas de funcionalidad.

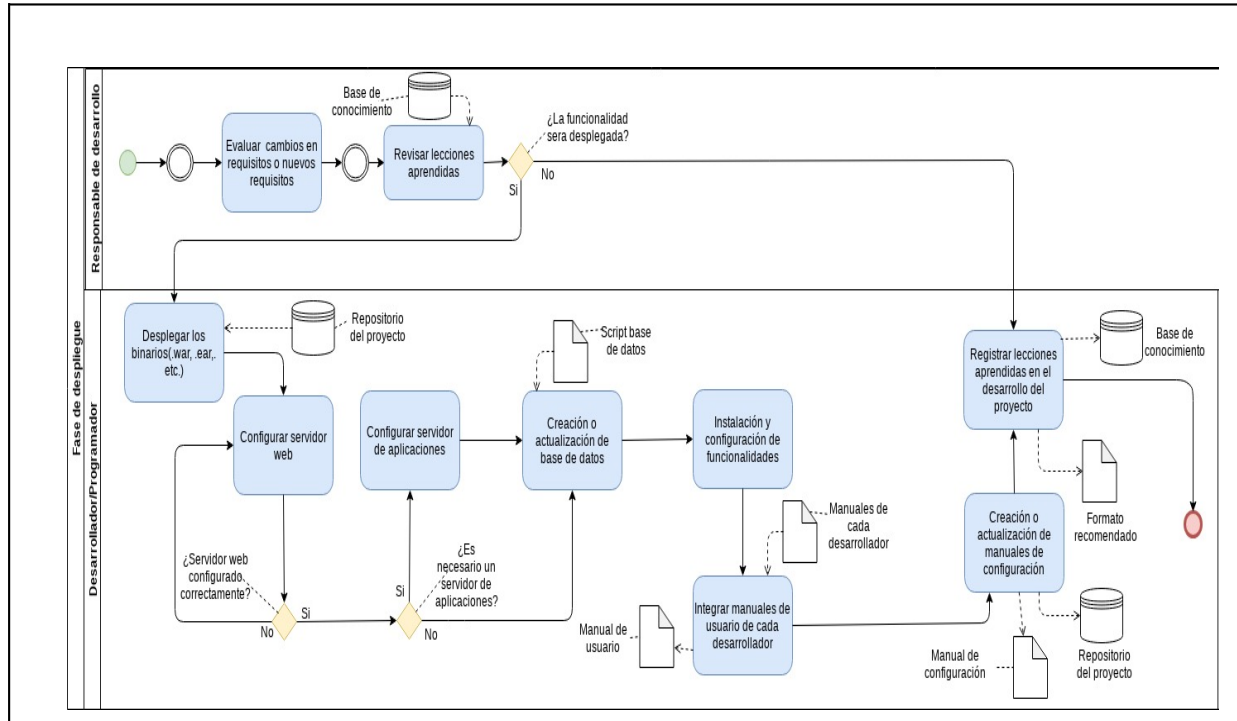
	A4.12. Si las pruebas de usabilidad son requeridas ir a la actividad A4.13., de lo contrario continuar con el paso A4.16.
EP	A4.13. Realizar pruebas de usabilidad. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas. • Documentar los errores en caso de que se hayan presentado.
	A4.14. Si existen errores en las pruebas de usabilidad ir a la actividad A4.15, de lo contrario ir al paso A4.16.
DE	A4.15. Corregir errores en pruebas de usabilidad.
	A4.16. Si las pruebas de seguridad son requeridas ir a la actividad A4.17, de lo contrario continuar con el paso A4.20.
EP	A4.17. Realizar pruebas de seguridad. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas.
	A4.18. Si existen errores al realizar las pruebas de seguridad ir a la actividad A4.19., de lo contrario continuar con el paso A4.20.
DE	A4.19. Corregir errores en pruebas de seguridad, luego volver a la actividad A4.17.
	A4.20. Si las pruebas de desempeño son requeridas ir a la actividad A4.21., de lo contrario ir al paso A4.24.
EP	A4.21. Realizar pruebas de rendimiento o desempeño. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas.
	A4.22. Si existen errores al realizar las pruebas de rendimiento o desempeño ir a la actividad A4.23, de lo contrario ir al paso A.4.24
DE	A4.23. Corregir errores en pruebas de desempeño.

	A4.24. Si las pruebas de accesibilidad son requeridas, ir a la actividad A4.25., de lo contrario continuar con el paso A4.28.
EP	A4.25. Realizar pruebas de accesibilidad. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas.
	A4.26. Si existen errores en las pruebas de accesibilidad ir a la actividad A4.27. de lo contrario ir al paso A4.28.
DE	A4.27. Corregir errores en pruebas de accesibilidad, luego volver a la actividad A4.25.
	A4.28. Si las pruebas de diseño adaptativo son requeridas, ir a la actividad A4.29, de lo contrario continuar con el paso A4.32.
EP	A4.29. Realizar pruebas de diseño adaptativo. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas.
	A4.30. Si existen errores en las pruebas de diseño adaptativo ir a la actividad A4.31., de lo contrario ir al paso A4.32.
DE	A4.31. Corregir errores en diseño adaptativo, luego volver a la actividad A4.29.
	A4.32. Si las pruebas de link testing son requeridas, ir a la actividad A4.33., de lo contrario ir al paso A4.36.
EP	A4.33. Realizar pruebas de link testing. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas.
	A4.34. Si existen errores en las pruebas de link testing ir a la actividad A4.35., de lo contrario ir al paso A4.36.
DE	A4.35. Corregir errores en Link testing, luego volver a la actividad A4.33.

	A4.36. Si las pruebas de browser testing son requeridas ir a la actividad A4.37, de lo contrario ir al paso A4.40.
EP	A4.37. Realizar pruebas de browser testing. <ul style="list-style-type: none"> • Definir las pruebas a realizar. • Ejecutar pruebas.
	A4.38. Si existen errores en las pruebas de browser testing ir a la actividad A4.39., de lo contrario ir al paso A4.40.
DE	A4.39. Corregir errores en Browser testing, luego volver a la actividad A4.37.
	A4.40. Si existen más funcionalidades para probar en la iteración volver a la actividad A4.8., de lo contrario ir a la actividad A4.5.
Salidas	



Rol	Descripción
A5. Fase de Despliegue	
Entradas	<ul style="list-style-type: none"> • Script de base de datos. • Manuales de cada desarrollador
RD ET	A5.1. Si la funcionalidad no va a ser desplegada ir a la actividad A5.2., de lo contrario ir a la actividad A5.3.
DE	A5.2. Registrar lecciones aprendidas en el desarrollo del proyecto, luego de registrar las lecciones aprendidas se termina la fase.
DE	A5.3. Configurar servidor web.
DE	A5.4. Si el servidor web no es configurado correctamente ir a a la actividad A5.3., de lo contrario ir al paso A5.5.
DE	A5.5. Si es necesario un servidor de aplicaciones ir a la actividad A5.6, de lo contrario continuar en la actividad A5.7.
DE	A5.6. Configurar servidor de aplicaciones.
ET	A5.7. Creación o actualización de base de datos.
DE	A5.8. Instalación y configuración de funcionalidades.
	A5.9. Integrar manuales de usuario de cada desarrollador.
	A5.10. Creación o actualización de manuales de configuración. Terminada esta actividad ir a la actividad A5.2.
Salidas	<ul style="list-style-type: none"> • Manual de usuario.



Anexo G. Plantilla del criterio de listo

Plantilla

(Nombre de la empresa)

(Nombre de la aplicación)

Tabla de contenido

- 1. Introducción**
- 2. Propósito**
- 3. Criterio de listo**

Versión	Autor	Descripción	Fecha

1. Introducción

Se escribe de qué se trata el documento en general.

2. Propósito

Describir para que se crea el documento.

3. Criterio de listo

Identificador	Enunciado	Listo	No listo
1	¿Cada historia de usuario tiene al menos 5 criterios de aceptación?		
2	¿La historia de usuario tiene un tamaño adecuado para ser realizada en el tiempo que dura el spring?		
3	Que la historia sea transversal, es decir que describa un comportamiento end-to-end (que sea "visible" para le negocio);		
4	Que el equipo comprenda el enfoque para las pruebas tomando en cuenta aspectos funcionales y no funcionales;		
5	¿Se identifica la(s) dependencia(s) con otras historias de usuario dentro del spring?		
6	¿La historia de usuario se ajusta al objetivo del spring?		
7	¿La historia de usuario contiene al menos un prototipo?		

Anexo H. Plantilla del criterio de hecho

Plantilla

(Nombre de la empresa)

(Nombre de la aplicación)

Tabla de contenido

- 1. Introducción**
- 2. Propósito**
- 3. Criterio de hecho**

Versión	Autor	Descripción	Fecha

1. Introducción

Se escribe de qué se trata el documento en general.

2. Propósito

Describir para que se crea el documento.

3. Criterio de hecho

Identificador	Enunciado	Listo	No listo
1	¿Se han realizado pruebas unitarias al código y han sido exitosas?		
2	¿Se han realizado pruebas de integración y han sido exitosas?		
3	¿Se han realizado pruebas funcionales de acuerdo a los criterios de aceptación?		
4	¿Se han realizado pruebas no funcionales y han sido exitosas?		
5	¿La funcionalidad entregada esta de acuerdo a la historia de usuario?		
6	¿Se ha realizado la creación del manual de usuario de cada funcionalidad?		
7	¿Se ha realizado la creación del manual de usuario de cada funcionalidad?		
8	¿Se ha cumplido con las vistas arquitectónicas y sus artefactos?		
9	¿La funcionalidad ha pasado por cada una de las actividades de cada fase?		

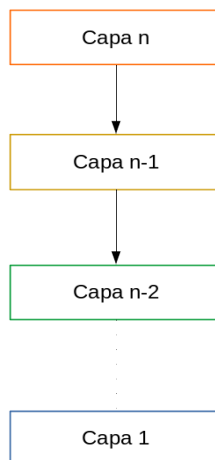
Anexo I. Estilos arquitectónicos

Patrón de Capas

Este patrón permite dividir una aplicación en capas. Cada capa contiene grupos de clases, las cuales cumplen con un mismo fin, cada capa está conectada y proporciona servicios a la capa superior. (Tomado de <https://prezi.com/avl1wwwvsknx/patrones-de-arquitectura-y-diseno/>).

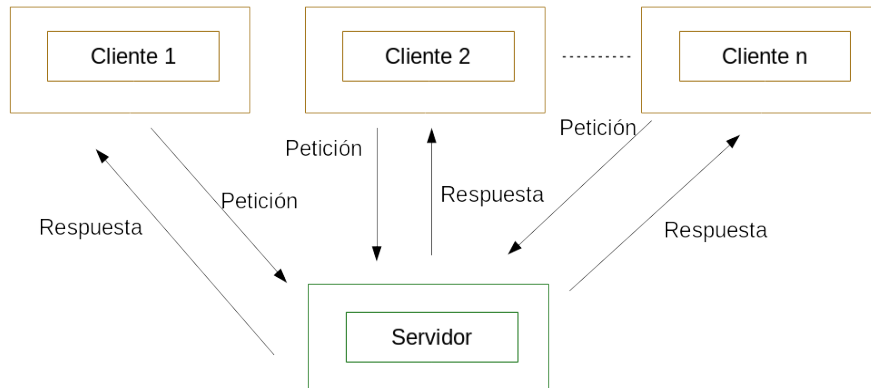
Las capas que son comunes en muchas aplicaciones son: (Tomado de <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>).

- Capa de presentación
- Capa de aplicación
- Capa de lógica de negocios
- Capa de acceso a datos.



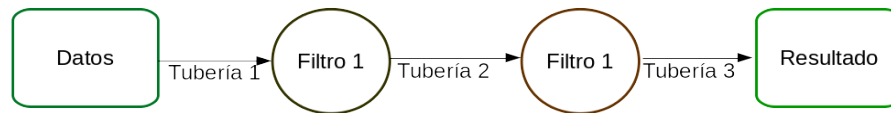
Patrón cliente servidor

El patrón de cliente servidor está estructurado en dos partes; un conjunto de clientes que son los encargados de realizar las peticiones, y un servidor, el cual escucha y responder las peticiones realizadas por los diferentes clientes. (Tomado de <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>).



Patrón Tuberías y filtros

Este patrón se basa en tuberías y filtros para dar tratamiento a datos. Existen varios componentes (filtros) entrelazados (mediante tuberías), mediante los cuales fluye la información. Cada componente produce una salida que se convierte en la entrada de otro componente (Tomado de <https://prezi.com/avl1wwwvsknx/patrones-de-arquitectura-y-diseno/>).

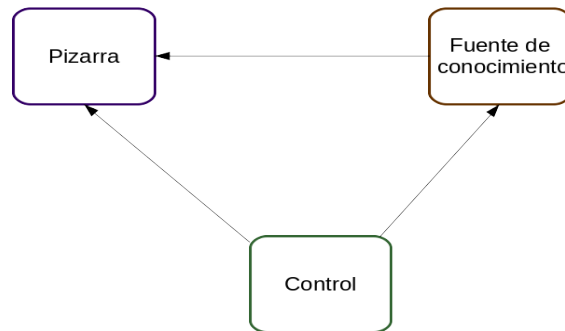


Patrón de pizarra

Este patrón cuenta con tres componentes esenciales:

- Pizarra: Espacio donde se encuentran los objetos creados.
- Fuente de conocimiento: Son módulos que cuentan con una representación propia.
- Componente de control: Selecciona, configura y ejecuta los módulos.

Los componentes antes mencionados pueden acceder a la pizarra, estos pueden crear nuevos objetos que se agregan a la pizarra. (Tomado de <https://medium.com/@maniakhitoccori/los-10-patrones-comunes-de-arquitectura-de-software-d8b9047edf0b>)



Estilos arquitectónicos

Modelo 4+1 vistas de Kruchten

El modelo 4+1 vistas fue diseñado por el profesor Philippe Kruchten. Este modelo es utilizado para ver un sistema desde diferentes vistas que permitan entender y representar de una forma mas completa un sistema.

Las vistas que contiene este modelo son: I) Vista lógica, II) Vista de despliegue, III) Vista de procesos, IV) Vista física y V) Vista de escenarios. Cada cada vista contiene una serie de diagramas que apoyan la construcción de un sistema software. (Tomado de <https://jarroba.com/modelo-41-vistas-de-kruchten-para-dummies/>)

A continuación se realiza la descripción de cada vista contenida en el modelo (Tomado de <https://jarroba.com/modelo-41-vistas-de-kruchten-para-dummies/>).

a. Vista lógica: En esta vista se representan las funcionalidades que se encuentran contenidas en las historias de usuario. Cada funcionalidad es utilizada para representarse mediante la abstracción en los diferentes diagramas que esta vista contiene (Tomado de http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:modelo4_1.pdf)

Los diagramas que contienen esta vista son: Diagrama de clases, diagrama de comunicación y diagrama de secuencia.

b. Vista de despliegue: Esta vista es la encargada de mostrar la distribución que tendrá la aplicación software a desarrollar (Tomado de <https://jarroba.com/modelo-41-vistas-de-kruchten-para-dummies/>). Se pueden observar los módulos, conexiones entre estos módulo y capas utilizadas para la distribución de la aplicación que se desea crear (Tomado de http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:modelo4_1.pdf).

Esta vista contiene el diagrama de componentes y el diagrama de paquetes (Tomado de <https://jarroba.com/modelo-41-vistas-de-kruchten-para-dummies/>).

c. Vista de procesos: “En esta vista se muestran los procesos que hay en el sistema y la forma en la que se comunican estos procesos; es decir, se representa desde la perspectiva de un integrador de sistemas, el flujo de trabajo paso a paso de negocio y operacionales de los componentes que conforman el sistema” (Tomado de <https://jarroba.com/modelo-41-vistas-de-kruchten-para-dummies/>).

Esta vista contiene el diagrama de actividades.

d. Vista física: En esta vista se muestra el sistema en cuanto a la distribución física que va a tener. Para la creación de esta vista se deben tener en cuenta una parte fundamental en el desarrollo de aplicaciones web, como lo son los requisitos no funcionales(disponibilidad, confiabilidad, rendimiento, y escalabilidad.), estos requisitos son el insumo que permite crear un diagrama de despliegue de una forma correcta y completa (Tomado de http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:modelo4_1.pdf).

En esta vista se tiene el diagrama de despliegue. Este diagrama contiene componentes y conectores que permiten representar la distribución que tiene el sistema (Tomado de http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:modelo4_1.pdf).

e. Vista de escenarios: Esta vista es la encargada de unir las otras 4 vistas, con lo cual la vista de escenarios contiene todos los diagramas elegidos para la creación del sistema. Mediante casos de uso se se puede realizar la unión de todas estas vistas.

Para este proceso se recomienda utilizar como mínimo diagramas como: diagrama de clases, diagrama de bases de datos, diagrama de despliegue, diagrama de componentes y diagrama de paquetes. La empresa esta en la opción de utilizar mas diagramas si lo cree necesario, u obviar los que crea no necesarios. El diagrama de bases se debe ser creado, ya que esta plasmado para su uso en una actividad del proceso.

Para la aplicación de este proceso la vista de escenarios se muestra solo como información, ya que esta propuesta utiliza historias de usuario.

En cuanto a los patrones arquitectónicos su uso depende de la aplicación que va a realizar.

Anexo J. Plantilla ambientes de desarrollo y pruebas

Plantilla

(Nombre de la empresa)

(Nombre de la aplicación)

Tabla de contenido

- 1. Introducción**
- 2. Propósito**
- 3. Herramientas seleccionadas**

Versión	Autor	Descripción	Fecha

1. Introducción

Se escribe de qué se trata el documento en general.

2. Propósito

Describir para que se crea el documento.

3. Herramientas seleccionadas

Herramienta	Descripción
<i>Netbeans</i>	<i>Herramienta utilizada como entorno de programación para el lenguaje utilizado en el backend.</i>

Anexo K. Plantilla de lecciones aprendidas

Plantilla

(Nombre de la empresa)

(Nombre de la aplicación)

Tabla de contenido

- 1. Introducción**
- 2. Propósito**
- 3. Lecciones aprendidas**

Versión	Autor	Descripción	Fecha

1. Introducción

Se da una descripción de qué se trata el documento en general.

2. Propósito

Describir para que se crea el documento.

3. Lecciones aprendidas

Identificador	La empresa elige el identificador que desee.	Fecha creación	Fecha registro lección aprendida.
Nombre	Nombre de la lección aprendida.		
Situación presentada	Se describe en que momento se presento la situación, cuando, como y por que.		
Descripción de la solución presentada	Se describe que se hizo para dar solución a la situación presentada.		
Observaciones	Se agrega alguna observación que considere necesaria.		

Anexo L. Plantilla para pruebas

Plantilla

(Nombre de la empresa)

(Nombre de la aplicación)

Pruebas de (Tipo de pruebas)

Tabla de contenido

- 1. Introducción**
- 2. Propósito**
- 3. Términos**
- 4. Pruebas**

Historial documento

Versión	Autor	Descripción	Fecha

1. Introducción

Describir a un alto nivel el contenido del documento y el tema a tratar.

2. Propósito

Describir el propósito del documento.

3. Términos

Lista de los términos utilizados en la pruebas y su correspondiente descripción.

4. Pruebas

Prueba			
Identificador historia de usuario	Agregar el identificador que se le ha asignado a la historia de usuario que contiene la funcionalidad a probar.	Módulo	Nombre del módulo al que pertenece la funcionalidad a probar.
Identificador de la prueba	Identificador único asignado a cada prueba realizada.	Fecha creación prueba	Fecha en la cual la prueba es redactada.
Tipo de prueba	Hace referencia a que pruebas se van a realizar(Funcionales, seguridad, diseño responsive, etcétera.).	Responsable creación de la prueba	Persona encargada de redactar la prueba a realizar.
Prueba	Se describe la prueba que se desea realizar.		
Condiciones aprobación de la prueba	Describir se tiene que cumplirse para que la prueba sea satisfactoria.		
Condiciones no aprobación de la prueba	Describir se tiene que cumplir para que la prueba no sea aprobada.		
Precondiciones	Precondiciones a tener en cuenta para ejecutar la prueba.	Postcondiciones	Postcondiciones a tener en cuenta para ejecutar la prueba.
Observaciones	Describir cosas adicionales que no se documentaron en los anteriores campos.		
Ejecución de la prueba			
Responsable ejecución de la prueba	Persona encargada de realizar la prueba.	Fecha ejecución prueba	Fecha en la cual la prueba es ejecutada.
Herramienta utilizada	Nombre de la herramienta utilizada para realizar la prueba.		

Resultado	Descripción del resultado obtenido una vez se ha realizado la ejecución de la prueba.	
¿Aprobada?	Si ()	No ()
Observaciones	Describir cosas adicionales que no se documentaron en los anteriores campos.	

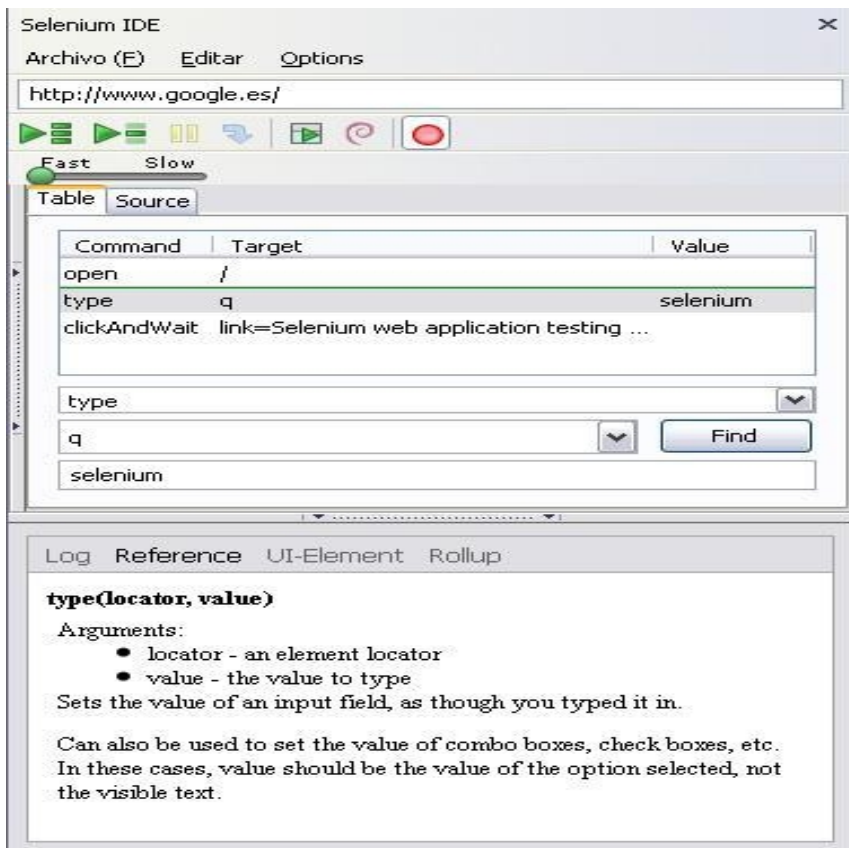
Anexo M. Herramientas pruebas funcionales

Selenium

Es una herramienta gratuita que facilita la automatización de pruebas de aplicaciones web. Mediante las funcionalidades que posee permite grabar las pruebas a ejecutar y correr un script con la ayuda de un navegador. Esta herramienta es soportada por navegadores como: chrome y firefox.

Selenium se compone de diferentes partes, la elección de la parte de selenium adecuada se debe elegir dependiendo de las funcionalidades requeridas. Por ejemplo si se está interesado en crear pruebas automatizadas de regresión basadas en un navegador o distribuir scripts en muchos entornos, se debe optar por Selenium WebDriver, de lo contrario si se desea crear scripts para reproducir una serie de

pruebas automatizadas lo ideal es utilizar Selenium IDE (Tomado de <https://www.seleniumhq.org/>).



Tomada de <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/381> accedido el 1-mayo-2019

katalon

Herramienta gratuita y de código abierto para la automatización de pruebas para API, web y móvil. Integra una serie de componentes y plantillas que permiten realizar la automatización de pruebas de una forma organizada y completa.

“Katalon ofrece una alternativa viable a las soluciones de automatización de pruebas comerciales que no son asequibles para muchos equipos pequeños y medianos” (Tomado de <https://docs.katalon.com/katalon-studio/docs/overview.html>).

Esta herramienta permite grabar los pasos realizados en la aplicación web, para luego ser reproducidos automáticamente. La grabación capturada genera un código en Groovy, el cual se edita y se adapta con las variables necesarias para ejecutar las

pruebas automatizadas (Tomado de <https://unpocodejava.com/2017/11/27/un-pococode-katalon-studio/>)

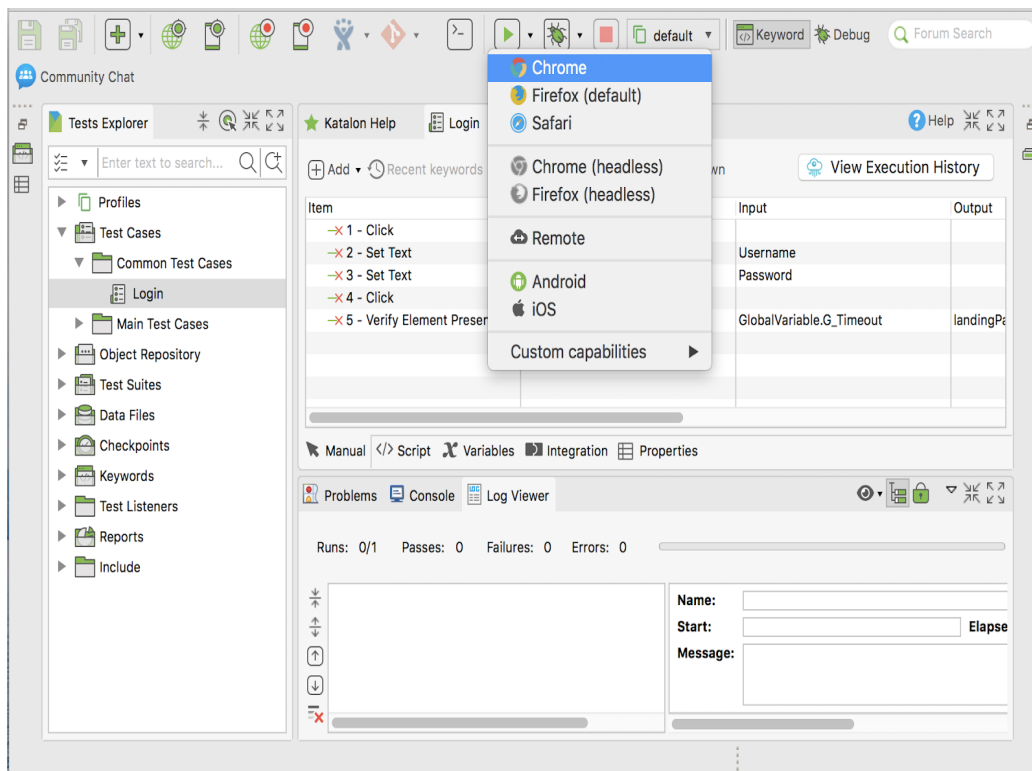


Imagen tomada de <https://docs.katalon.com/katalon-studio/docs/getting-started.html#getting-a-sample-project-up-and-running> accedidas el 1-mayo-2019

Anexo N. Herramientas pruebas de usabilidad

Evaluación heurística

Una de las técnicas más usadas para evaluar la usabilidad en una aplicación web es la evaluación heurística, la cual pertenece a los métodos de inspección de la usabilidad. Esta técnica fue creada por Nielsen, con el objetivo de encontrar fallas de usabilidad que puedan presentarse en un sistema web. La evaluación heurística debe ser aplicada por evaluadores expertos, los cuales toman como guía un conjunto de reglas(heurística) para validar la interfaz de la aplicación web.

Las reglas(Principios heurísticos de Nielsen) que se toman en cuenta para realizar la validación de la interfaz de una aplicación web son las siguientes:

1. **Visibilidad del estado del sistema:** La aplicación debe brindar feedback, el cual permita al usuario tener conocimiento sobre qué ha sucedido y que ocurre actualmente en la aplicación.
2. **Empate entre el sistema y el mundo real:** Se debe en lo posible utilizar cosas que para el usuario sean fácil de identificarlas, elementos que de alguna forma son familiares para él, como por ejemplo para campos de fechas de nacimiento, sería adecuado utilizar un datepicker, debido a que su uso es muy frecuente en las páginas web.
3. **Control y libertad del usuario:** El tener una opción para eliminar la última acción realizada es de gran ayuda en una aplicación web, ya que el usuario puede equivocarse al momento de realizar una tarea y desea volver a un estado anterior de la aplicación.
4. **Consistencia y estándares:** Utilizar convenciones en una aplicación web es una de las reglas más importantes, ya que permiten que los significados de los elementos utilizados en diferentes páginas sean los mismos siempre.

5. **Prevención de errores:** Se debe evitar al máximo los errores que puedan surgir, tratar que los elementos sean lo más claros posibles, por ejemplo si un campo de un formulario es obligatorio, se tiene que mostrar de alguna forma que se debe llenar ese campo y no esperar al envío del formulario para decirle al usuario que hay un campo obligatorio que debió llenar.
6. **Reconocimiento mejor que recuerdo:** Evitar al máximo que el usuario tenga que recordar información de una vista a otra
7. **Flexibilidad y eficiencia de uso:** El sistema debe permitir tener atajos cuando sean necesarios, como por ejemplo si hay acciones frecuentes, el sistema debe estar en la posibilidad de permitir al usuario tener una combinación de teclas que le permitan realizar funciones de una manera más eficiente.
8. **Diseño estético y minimalista:** Los diseños de una aplicación web deben ser lo más sencillos posibles y tienen que contener la información realmente importante, debido a que si se agrega demasiada información en una página web, la información verdaderamente relevante puede verse opacada y no ser muy fácil de visualizar.
9. **Ayudar a reconocer, diagnosticar y recuperarse de errores:** Al presentarse un error se debe decir claramente de qué se trata utilizando un lenguaje que sea entendible por el usuario, y que le permita entender de forma clara el por que del error.
10. **Ayuda y documentación:** La documentación ayuda al usuario a resolver dudas que se puedan presentar al momento de utilizar la aplicación web.

(Tomado de <https://blog.interactius.com/metodolog%C3%ADas-de-ux-evaluaci%C3%B3n-heur%C3%ADstica-parte-i-b5d02b566987>)

Método de test con usuarios

Es una prueba utilizada para evaluar la usabilidad de una aplicación web. Para esta prueba se requiere de un grupo de usuarios, los cuales interactúan con la aplicación. El encargado de realizar la prueba observa y anota las fallas que va encontrando en el transcurso de la interacción del usuario con el sitio web.

Este método puede ser aplicado no solo al diseño final del sitio web, sino también a los prototipos iniciales construidos. Es muy recomendable al momento de ejecutar el método de test con usuario, haber realizado la evaluación heurística antes, todo esto con el fin de minimizar errores que se puedan encontrar al momento de aplicar el método.

Los pasos a tener en cuenta para realizar el método de test con usuarios son:

- **Elegir los participantes:** El número adecuado de participantes es 5, se debe tener en consideración que las personas a participar no deben haber tenido ninguna participación en el desarrollo de la aplicación web.
- **Lugar y equipos:** Elegir el lugar y equipos a utilizar, se puede grabar la sesión siempre y cuando el participante lo autorice.
- Planificar la prueba: Elaborar un documento el cual contenga las instrucciones para el usuario, y el tiempo establecido para cada instrucción.
- **Interactuar con el usuario:** Antes de empezar se debe explicar al usuario lo que se busca con la aplicación del método, impulsarlo a decir las cosas que siente durante la prueba, lo que piensa y las dudas que surjan durante la interacción con la aplicación. Se debe tener cuidado de explicar el funcionamiento del sitio web, ya que con la prueba se pretende ver qué tan fácil es para el usuario utilizarlo y entender su funcionamiento.
- **Inicio de la prueba:** Al iniciar la prueba se indica al usuario que tareas debe completar. Cuando el usuario empieza, el evaluador debe tomar el tiempo, observar las expresiones del usuario y anotar los detalles que cree relevantes en el desarrollo de la prueba, los cuales se utilizan al momento de realizar mejoras de usabilidad al sitio web.
- **Crear informe:** Al terminar la prueba se debe construir un documento que contenga que problemas de usabilidad se detectaron y observaciones de cómo solucionarlos.

(Tomado de http://www.nosolousabilidad.com/articulos/test_usuarios.htm)

- **Thinking aloud:** El thinking aloud es un método de evaluación de la usabilidad de un producto, el cual permite recoger aportes importantes al momento de realizarlo. Para su ejecución se elige un grupo de personas, se crean una serie de tareas a realizar, se proporciona un prototipo y se explican a los participantes los pasos a seguir durante la duración del ejercicio (Tomado de <http://mpiua.invid.udl.cat/pensando-en-voz-alta-thinking-aloud/>).

En la realización de las tareas asignadas, los usuarios deben expresar en voz alta las dudas, sentimientos y pensamientos que surjan (Tomado de <http://mpiua.invid.udl.cat/pensando-en-voz-alta-thinking-aloud/>). El encargado de la sesión no debe influenciar a los participantes, con lo cual debe tener cuidado con las cosas que dice o hace durante la ejecución del ejercicio (Tomado de <http://www.itcelaya.edu.mx/ojs/index.php/pistas/article/download/1078/875>).

Anexo O. Herramientas pruebas de seguridad

OpenVas

OpenVas es una herramienta de uso libre, que permite identificar y realizar la corrección de las fallas encontradas. Esta herramienta puede ser operada mediante una serie de comandos o con una interfaz web; esta herramienta viene instalada por defecto en kali linux, lista para comenzar a utilizarla.

Esta herramienta ofrece dos servicios:

El OpenVas Manager, el cual es el encargado de organizar la información recopilada en los análisis realizados, esta información puede ser filtrada o clasificada dependiendo de las necesidades. Además el manager permite interactuar con las bases de datos que contienen la información recopilada de los análisis, usuarios junto con sus roles y grupos.

El escáner es el encargado de ejecutar las pruebas de vulnerabilidad de red. Estas pruebas han sido programadas para detectar vulnerabilidades de seguridad en los sistemas, las pruebas son conocidas como NVT(Network Vulnerability Tests).

(Tomado de <https://www.welivesecurity.com/la-es/2014/11/18/como-utilizar-openvas-evaluacion-vulnerabilidades/>) Accedido 1-mayo-2019. Todo el contenido es tomado de la página.

A continuación se muestran algunas imágenes de la herramienta.

Scan Management	Asset Management	Configuration	Extras	Administration	Help
New Task ?					
Name	<input type="text" value="prueba"/>				
Comment (optional)	<input type="text"/>				
Scan Config	Full and very deep ultimate ▾				
Scan Targets	prueba ▾				
Escalator (optional)	-- ▾				
Schedule (optional)	-- ▾				
Slave (optional)	-- ▾				
Observers (optional)	<input type="text"/>				
Scan Intensity					
Maximum concurrently executed NVTs per host	<input type="text" value="4"/>				
Maximum concurrently scanned hosts	<input type="text" value="20"/>				
					<input type="button" value="Create Task"/>

Tomado de <https://www.welivesecurity.com/la-es/2014/11/18/como-utilizar-openvas-evaluacion-vulnerabilidades/>.

SqlMap

Mediante esta herramienta se pueden hacer ataques de inyección sql. SqlMap permite detectar vulnerabilidades de una aplicación web, una vez detectadas se ofrece una serie de opciones a realizar; entre las opciones permitidas están: l) Obtener los hashes de las contraseñas, datos de la base de datos, ejecutar consultas, entre otras (Tomado de <https://www.dragonjar.org/sqlmap-herramienta-automatizada-de-inyeccion-sql.xhtml>)

Algunas de las características son:

- **Esta herramienta soporta bases de datos como:** MySql, Postgresql, Oracle, Microsoft SQL server, Microsoft access, IBM DB2, SQLite, Firebird, Sybase, entre otros.
- **Compatibilidad total con seis técnicas de inyección SQL:** basado en booleano, basado en tiempo, basado en error, basado en consultas UNION,

consultas apiladas y fuera de banda” (Tomado de <https://securityhacklabs.net/articulo/sqlmap-herramienta-automatica-para-pruebas-de-penetracion-sql-injection>).

- Permite realizar el volcado de las tablas de la base de datos.

```
s python sqlmap.py -u "http://172.16.112.128/sqlmap/mysql/get_int.php?id=1" --batch
[1.3.4.44#dev]
http://sqlmap.org
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsi
ble for any misuse or damage caused by this program
[*] starting @ 10:34:28 /2019-04-30/
[10:34:28] [INFO] testing connection to the target URL
[10:34:28] [INFO] heuristics detected web page charset 'ascii'
[10:34:28] [INFO] checking if the target is protected by some kind of WAF/IPs
[10:34:28] [INFO] testing if the target URL content is stable
[10:34:29] [INFO] target URL content is stable
[10:34:29] [INFO] testing if GET parameter 'id' is dynamic
[10:34:29] [INFO] GET parameter 'id' appears to be dynamic
[10:34:29] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable (possible DBMS: 'MySQL')
[10:34:29] [INFO] heuristic (XSS) test shows that GET parameter 'id' might be vulnerable to cross-site scripting (XSS) at
tacks
[10:34:29] [INFO] testing for SQL injection on GET parameter 'id'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] Y
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [
Y/n] Y
[10:34:29] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[10:34:29] [WARNING] reflective value(s) found and filtering out
[10:34:29] [INFO] GET parameter 'id' appears to be 'AND boolean-based blind - WHERE or HAVING clause' injectable (with --
strings="Luther")
[10:34:29] [INFO] testing 'MySQL >= 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BIGINT UNSIGNED)'
[10:34:29] [INFO] testing 'MySQL >= 5.5 OR error-based - WHERE or HAVING clause (BIGINT UNSIGNED)'
[10:34:29] [INFO] testing 'MySQL >= 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXP)'
[10:34:29] [INFO] testing 'MySQL >= 5.5 OR error-based - WHERE or HAVING clause (EXP)'
[10:34:29] [INFO] testing 'MySQL >= 5.7.8 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (JSON_KEYS)'
[10:34:29] [INFO] testing 'MySQL >= 5.7.8 OR error-based - WHERE or HAVING clause (JSON_KEYS)'
[10:34:29] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)'
[10:34:29] [INFO] GET parameter 'id' is 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR
)' injectable
[10:34:29] [INFO] testing 'MySQL inline queries'
[10:34:29] [INFO] testing 'MySQL > 5.0.11 stacked queries (comment)'
[10:34:29] [WARNING] time-based comparison requires larger statistical model, please wait..... (done)
[10:34:29] [INFO] testing 'MySQL > 5.0.11 stacked queries'
[10:34:29] [INFO] testing 'MySQL > 5.0.11 stacked queries (query SLEEP - comment)'
[10:34:29] [INFO] testing 'MySQL > 5.0.11 stacked queries (query SLEEP)'
[10:34:29] [INFO] testing 'MySQL < 5.0.12 stacked queries (heavy query - comment)'
[10:34:29] [INFO] testing 'MySQL < 5.0.12 stacked queries (heavy query)'
[10:34:29] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind'
[10:34:30] [INFO] GET parameter 'id' appears to be 'MySQL >= 5.0.12 AND time-based blind' injectable
[10:34:30] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[10:34:30] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other
(potential) technique found
[10:34:30] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number
of query columns. Automatically extending the range for current UNION query injection technique test
[10:34:30] [INFO] target URL appears to have 3 columns in query
[10:34:30] [INFO] GET parameter 'id' is 'Generic UNION query (NULL) - 1 to 20 columns' injectable
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 46 HTTP(s) requests:
---
Parameter: id (GET)
Type: boolean-based blind
Title: AND boolean-based blind - WHERE or HAVING clause
Payload: id=1 AND 6489=6489
Type: error-based
Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
Payload: id=1 AND (SELECT 7857 FROM(SELECT COUNT(*),CONCAT(0x717a786a71,(SELECT (ELT(7857=7857,1))) ,0x716a6b6a71,FLOOR
(RAND(0)*2)))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind
Payload: id=1 AND SLEEP(5)
Type: UNION query
Title: Generic UNION query (NULL) - 3 columns
Payload: id=1 UNION ALL SELECT NULL,CONCAT(0x717a786a71,0x5a5151727477666c4c4162475655626153796d79455946714b5153456f5
a7a4f6f57724d586d614d,0x716a6b6a71),NULL-- swCD
[10:34:30] [INFO] the back-end DBMS is MySQL
web application technology: PHP 5.2.6, Apache 2.2.9
back-end DBMS: MySQL >= 5.0
[10:34:30] [INFO] fetched data logged to text files under '/home/stamparm/.sqlmap/output/172.16.112.128'
[*] ending @ 10:34:39 /2019-04-30/
s
```

Tomado de <https://github.com/sqlmapproject/sqlmap> el 1-mayo-2019.

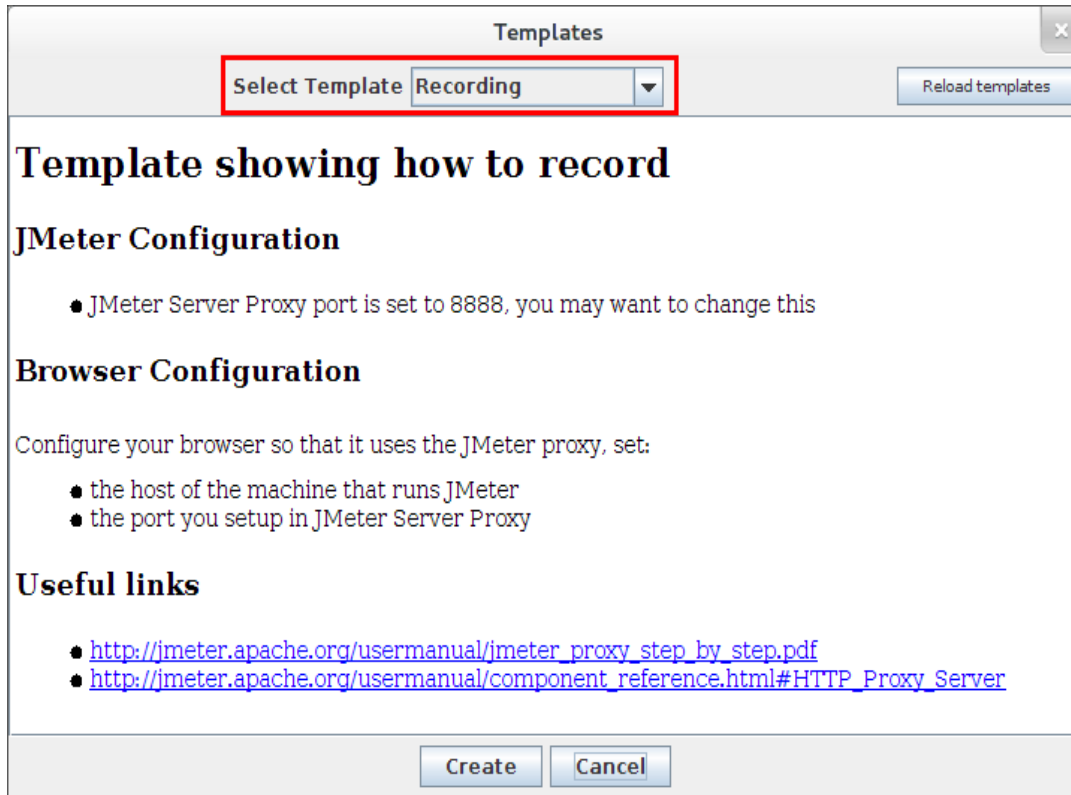
Anexo P. Herramientas pruebas de rendimiento o desempeño

JMeter

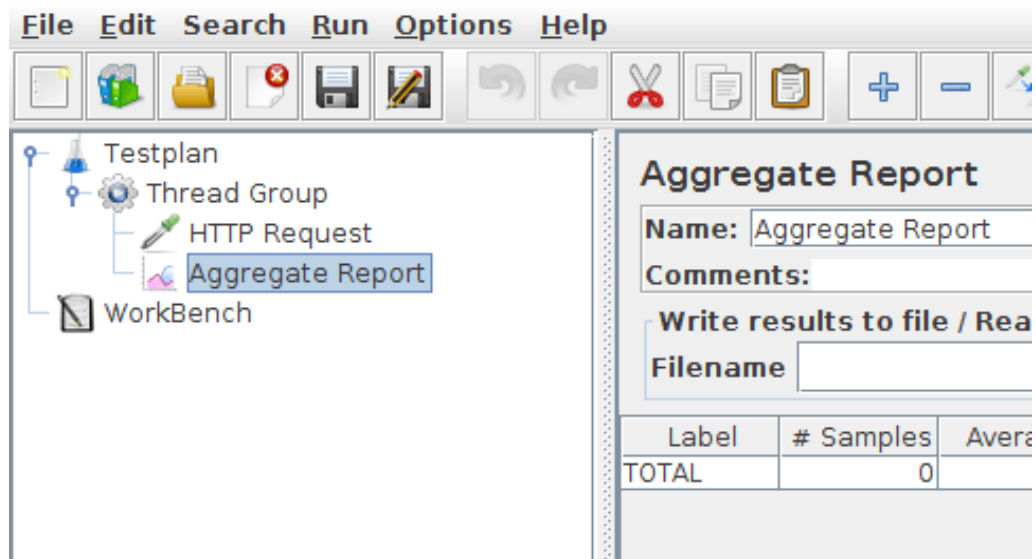
JMeter es una herramienta de código abierto para realizar pruebas de rendimiento y pruebas funcionales a aplicaciones web (Tomado de <https://www.ecured.cu/Jmeter>, accedido 21-04-2019, <https://jmeter.apache.org/>, accedido 21-04-2019). Su arquitectura ha ido mejorando con el tiempo permitiendo ser utilizado en pruebas a bases de datos, programas en Perl y otros medios (Tomado de <https://www.osmosislatina.com/jmeter/basico.htm>). Ayuda a diseñar y ejecutar un plan de pruebas para una aplicación web, las pruebas pueden ser creadas mediante una interfaz GUI que permite agregar y probar componentes a utilizar al momento de realizar las pruebas de desempeño (Tomado de <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/388>). Al tener listo el plan de pruebas, se puede generar un archivo .jmx que contiene todas las pruebas a realizar (Tomado de <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/388>).

“Jmeter permite realizar pruebas de FTP, JDBC, JNDI, LDAP, SOAP/XML-RPC, y WebServices (en Beta)” (Tomado de <https://www.ecured.cu/Jmeter>, accedido 21-04-2019), posee la funcionalidad de activar o desactivar parte de las pruebas (Tomado de <https://www.ecured.cu/Jmeter>, accedido 21-04-2019), muy útil cuando se desea ejecutar solo una parte de ellas.

Para más información (<https://jmeter.apache.org/>).



Tomado de <https://jmeter.apache.org/usermanual/get-started.html> accedido el 1-mayo-2019

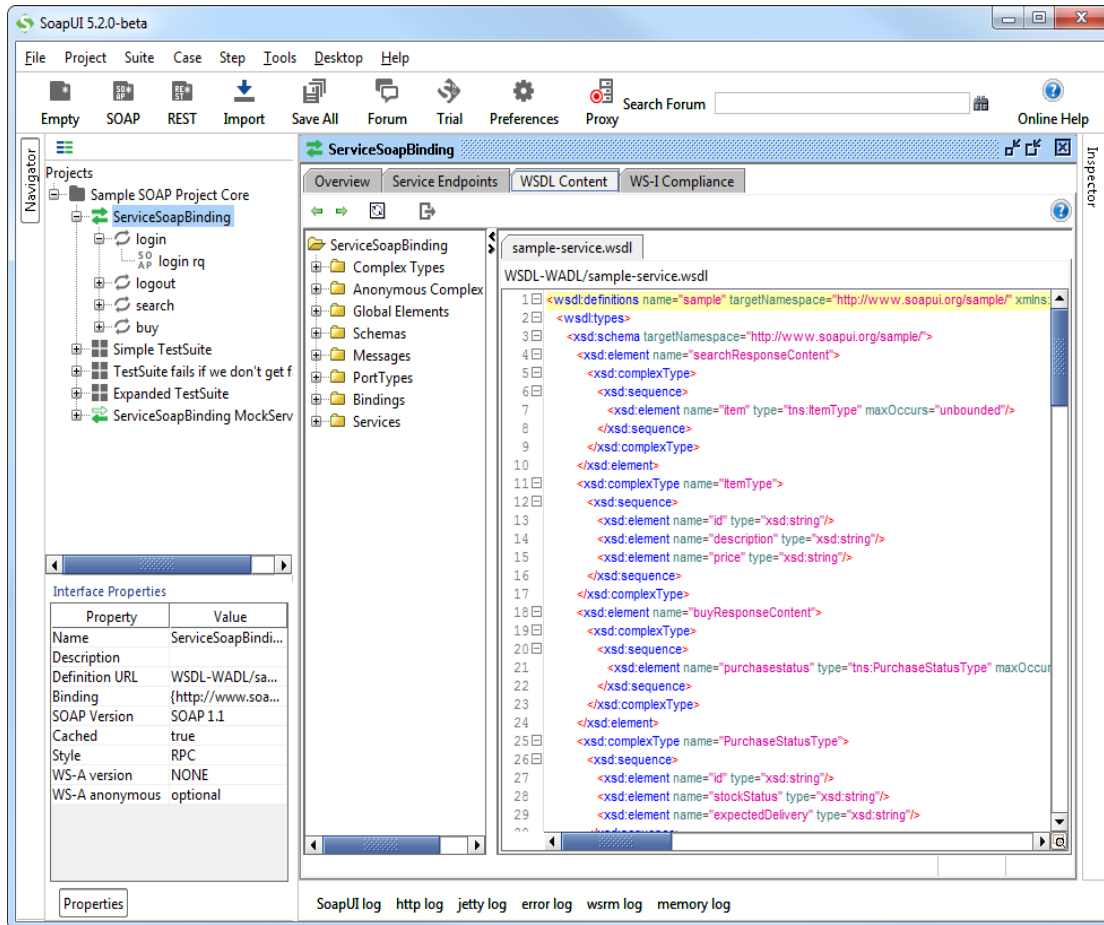


Tomado de https://jmeter.apache.org/usermanual/jmeter_distributed_testing_step_by_step.html accedido el 1-mayo-2019

SoapSonar

Es una herramienta que permite realizar pruebas para html, xml, soap, rest y json (Tomado de <http://www.pmoinformatica.com/2017/12/herramientas-testing-servicios-web.html>). Debido a su características ayuda en el soporte de pruebas funcionales, de rendimiento, seguridad entre otras (Tomado de <https://www.crosschecknet.com/products/soapsonar/api-automation-testing/>, accedido 21-04-2019). “Está herramienta está disponible en edición personal(gratuita), edición profesional y edición servidor” (Tomado de <http://www.pmoinformatica.com/2017/12/herramientas-testing-servicios-web.html>).

Se puede realizar un análisis de los datos arrojados al momento de ejecutar pruebas automatizadas, para las pruebas de rendimiento se pueden utilizar clientes virtuales que simulan peticiones a las aplicaciones (Tomado de <https://www.crosschecknet.com/portfolio-view/test-automation/>).

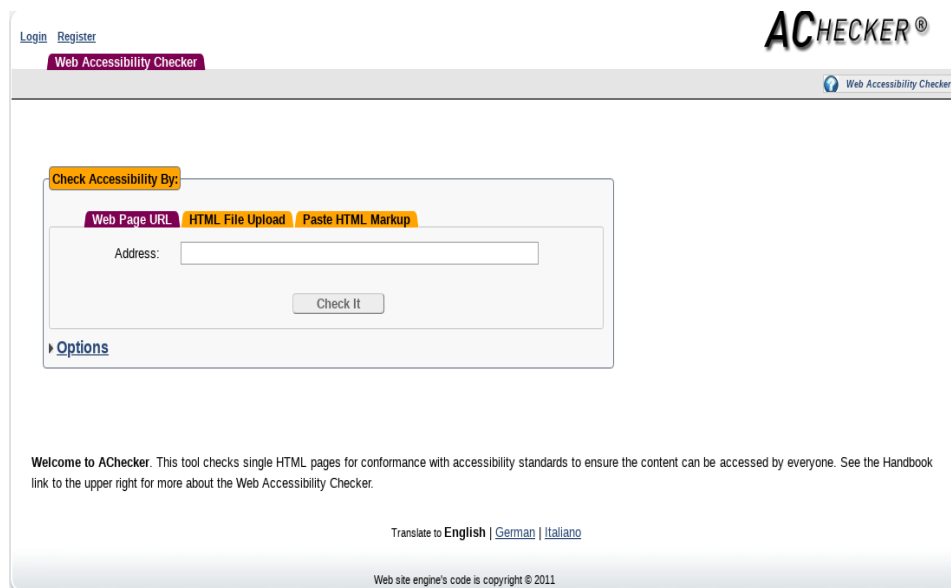


Tomado de <http://www.pmoinformatica.com/2017/12/herramientas-testing-servicios-web.html> accedido el 1-mayo-2019

Anexo Q. Herramientas pruebas de accesibilidad

Achecker

Es una herramienta que permite la evaluación de la accesibilidad en una página web. Esta herramienta ofrece varias opciones para verificar qué tan accesible es una página, para iniciar con el proceso de análisis se puede subir un archivo html, agregar el link o ingresando el código html de la página. Luego de proporcionar la página web a evaluar, achecker analiza y crear un informe de las posibles fallas encontradas (Tomado de <https://www.shopify.com.co/blog/9-herramientas-para-probar-la-accesibilidad-de-un-sitio-web>).



Tomado de <https://achecker.ca/checker/index.php> accedido el 1-mayo-2019

WAVE

Herramienta gratuita para la evaluación de la accesibilidad web, aplica estilos css para evaluar las página. Está herramienta se basa en las normas de WCAG, aunque no comprueba archivos javascript es una herramienta de gran ayuda al

momento de verificar qué tan accesible es una aplicación web (Tomado de <https://www.shopify.com.co/blog/9-herramientas-para-probar-la-accesibilidad-de-un-sitio-web>).



Site-wide WAVE Tools

Browser Extensions

Help

About/Terms of Use

Feedback

Web page address... →

Powered by [WebAIM](#)

Need more than just one page at a time?

Dinolytics is an enterprise-level web accessibility evaluation system based on WAVE that provides site-wide monitoring and reporting of accessibility over time.

Learn more at dinolytics.com

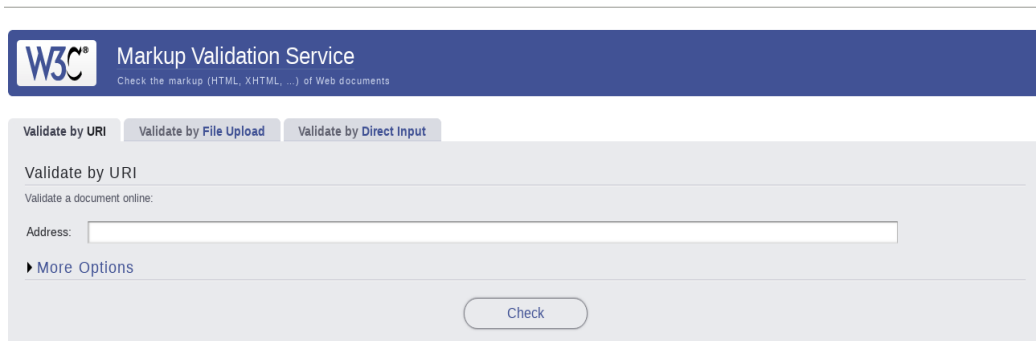


Tomado de <http://wave.webaim.org/> accedido 1-mayo-2019

Anexo R. Herramientas pruebas de diseño adaptativo

Screenfly

Aplicación web que permite evaluar el diseño responsive de un sitio web, posee diferentes resoluciones para diferentes dispositivos. Al entrar a la página de screenfly (Tomado de <http://quirktools.com/screenfly/>) se pueden observar los diferentes dispositivos que soporta y las resoluciones para cada dispositivo. Para su uso solo basta con ingresar la dirección url del sitio a evaluar, elegir la opción de Go e inmediatamente se carga la vista de la página a la cual se quiere verificar la visualización. A continuación se elige el dispositivo y resolución a evaluar.

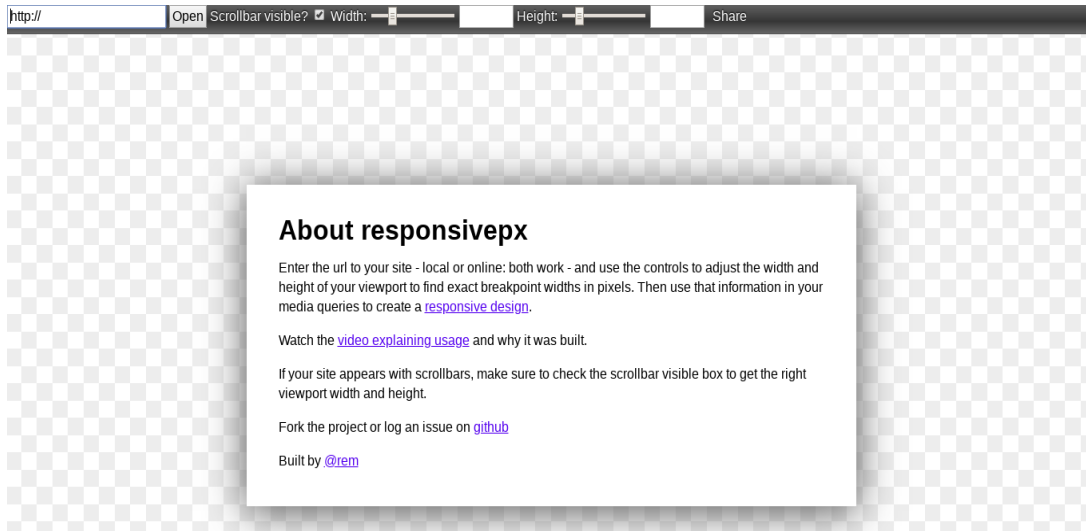


The image shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are three tabs: "Validate by URI", "Validate by File Upload", and "Validate by Direct Input". The "Validate by URI" tab is selected. Underneath, there is a text input field labeled "Address:" and a "Check" button. There is also a link for "More Options".

Tomada de <http://quirktools.com/screenfly/> accedido el 1-mayo-2019

Responsivepx

Aplicación web para evaluar el diseño responsive de un sitio web. Esta aplicación permite hacer scroll en diferentes tamaños de alto y ancho de una página web. Para utilizarla se ingresa a <http://responsivepx.com/>, con lo cual se abre la vista de responsivepx, se ingresa la url del sitio web a evaluar y se inicia a cambiar el alto y ancho de la página web según se necesite, al modificar el tamaño se puede observar si la página web se adapta o no a diferentes resoluciones.



Tomado de <http://responsivepx.com/> accedida el 1-mayo-2019

Anexo S. Herramientas pruebas de link testing

Xenu

Esta herramienta libre que permite realizar la comprobación de los enlaces contenidos en una aplicación web. Los enlaces pueden ser de imágenes, fondos, hojas de estilos, archivos de javascript, entre otros.

Esta herramienta muestra un listado de los enlaces que están fallando, los cuales se pueden ordenar de diferentes formas según la necesidad que se posee. Las características que posee esta herramienta son:

- Interfaces de usuario fáciles de utilizar.
- Permite volver a comprobar enlaces rotos
- Genera un informe fácil de entender, además este documento generado puede ser enviado por correo electrónico.
- Puede detectar y reportar las Urls redirigidas.

Para más información <http://home.snafu.de/tilman/xenulink.html>.


Address	Status	Type	Size	Title
http://www.polezno.com/soft/xenu	ok	text/html		russian description
http://videos.webpronews.com/video/frame2.php?movie_name=xenu	ok	text/html		
http://groups.yahoo.com/group/linksleuthupdates	ok	text/html		Update Announcements mailing lis
mailto:linksleuthupdates-subscribe@yahoogroups.com	mail host ok			linksleuthupdates-subscribe@yaho
http://groups.yahoo.com/group/xenu-usergroup/	ok	text/plain		user group
mailto:xenu-usergroup-subscribe@yahoogroups.com	mail host ok			xenu-usergroup-subscribe@yahooc
http://home.snafu.de/tilman/xenu_button.gif	ok	image/gif	1508	[Linkcheck by Xenu!]
http://home.snafu.de/tilman/xenu_button2.gif	ok	image/gif	810	[Linkcheck by Xenu!]
http://home.snafu.de/tilman/xenubanner.jpg	ok	image/jpeg	12040	
mailto:tilman@berlin.snafu.de	mail host ok			Contact me
http://cultinfobooks.com/detail.asp?product_id=DON-AMT	ok			tax-exempt
http://www.berliner-dialog.de/	ok	text/html	10162	Dialog Zentrum Berlin e.V.
http://api.flattr.com/button/load.js	ok	text/javascript	3810	
http://home.snafu.de/tilman/tmp/Xenu64.zip	ok	application/zip	215...	64 bit beta version
http://127.0.0.1/	no connection			http://127.0.0.1
http://web.archive.org/web/20051203055125/wired-vig.wired.com/news/business/0,1367...	busy			possibly patented
http://www.delphion.com/details?pn=US05694604_	ok	text/html		here
http://www.delphion.com/details?pn=US05694603_	busy			here
http://google.com/	ok	text/html	219	Google
http://webdev.archive.org/	no such host			Internet Archive
http://support.microsoft.com/support/kb/articles/q166/9/61.asp	ok			does not work with proxies
http://samspace.org/d/firewalls.html	not found			"Personal Firewalls" are mostly sna
http://www.andifashion.com/linkckg.shtml	ok	text/html		list of websites
http://www.boutell.com/innards/xenuandwikipedia.html	ok	text/html	13695	extended explanation of how wikip
http://www.deja.com=/dnc/getdocxp?AN=433246325	ok	text/html	249	read this usenet post for help
http://windowsupdate.microsoft.com/	ok	text/html	3191	install the latest NT service packs
http://www.microsoft.com/windows95/downloads/contents/wuadmintools/s_wunetworki...	ok	text/html	155	Windows 95 Kernel 32 Update
http://www.microsoft.com/windows95/downloads/contents/wuadmintools/s_wunetworki...	ok	text/html	155	Windows Socket 2 Update
http://www.microsoft.com/windows95/downloads/contents/WURRecommended/S_WUNet...	ok	text/html	155	Microsoft DUN 1.3 and Winsock2 Y
http://www.microsoft.com/technet/security/bulletin/ms99-049.asp	ok	text/html	187	Patch for "File Access URL" Vulnera
http://support.microsoft.com/default.aspx?scid=kb;EN-US;q285189	ok			Microsoft DUN 1.4
http://support.microsoft.com/support/kb/articles/Q176/4/20.ASP	ok			bug
http://support.microsoft.com/default.aspx?scid=kb;en-us;Q224318	ok	text/html	56467	workaround advice from Microsoft
http://groups.yahoo.com/group/xenu-usergroup/message/103	ok	text/plain		announced in the user group

Tomado de <http://home.snafu.de/tilman/xenulink.html> accedido el 1-mayo-2019

Link checker w3c

Herramienta para validar el correcto funcionamiento de enlaces de páginas, hojas de estilo, entre otros. Para obtener un resultado correcto se debe tener en cuenta que los documentos a evaluar cuenten con el marcado html y css válidos (Tomado de <https://validator.w3.org/>).

“El link checker es parte de los validadores y las herramientas web de calidad del W3C” (Tomado de <https://validator.w3.org/>).

 **Markup Validation Service**
Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI Validate by File Upload Validate by Direct Input

Validate by URI

Validate a document online:

Address:

▶ More Options

Check

Tomado de <https://validator.w3.org/> accedido el 1-mayo-2019

Anexo T. Herramientas pruebas de browser testing

Browsershots

BrowserShots es una herramienta gratuita para probar la compatibilidad de una aplicación web con diferentes navegadores corriendo en diferentes sistemas operativos. Se pueden obtener imágenes de distintos tamaños que permiten visualizar las páginas de una aplicación web.

En la página de la herramienta se puede observar los diferentes navegadores que soporta para realizar las pruebas. Para más información <http://browsershots.org/>.

The screenshot shows the BrowserShots website interface. At the top, there are three tabs: "Browser Compatibility Test" (selected), "Web Design Gallery", and "Icon Search Engine". Below the tabs is a search bar with the text "Enter URL Here:" and a "Submit" button. The main content area is divided into four columns representing different operating systems: Linux, Windows, Mac, and BSD. Each column contains a list of browser versions with checkboxes next to them, indicating which browsers are available for testing. A "Contribute" button is visible on the right side of the interface. A red banner for a "Free 7-Day Trial" of CrossBrowserTesting is also present.

Linux	Windows	Mac	BSD
<input type="checkbox"/> Arora 0.1	<input checked="" type="checkbox"/> Firefox 36.0	<input checked="" type="checkbox"/> Chrome 39.0	<input checked="" type="checkbox"/> Firefox 42.0
<input checked="" type="checkbox"/> Arora 0.11	<input checked="" type="checkbox"/> Firefox 37.0	<input checked="" type="checkbox"/> Chrome 44.0	<input checked="" type="checkbox"/> Firefox 43.0
<input checked="" type="checkbox"/> Chrome 37.0	<input checked="" type="checkbox"/> Firefox 38.0	<input checked="" type="checkbox"/> Chrome 45.0	<input checked="" type="checkbox"/> Firefox 44.0
<input checked="" type="checkbox"/> Chrome 38.0	<input checked="" type="checkbox"/> Firefox 39.0	<input checked="" type="checkbox"/> Chrome 51.0	<input checked="" type="checkbox"/> Firefox 45.0
<input checked="" type="checkbox"/> Chrome 48.0	<input checked="" type="checkbox"/> Firefox 40.0	<input checked="" type="checkbox"/> Chrome 71.0	<input checked="" type="checkbox"/> Firefox 46.0
<input checked="" type="checkbox"/> Dillo 3.0	<input checked="" type="checkbox"/> Firefox 43.0	<input checked="" type="checkbox"/> Chrome 73.0	<input checked="" type="checkbox"/> Firefox 47.0
<input checked="" type="checkbox"/> Epiphany 3.18	<input checked="" type="checkbox"/> Firefox 44.0	<input checked="" type="checkbox"/> Firefox 30.0	<input checked="" type="checkbox"/> Firefox 48.0
<input checked="" type="checkbox"/> Epiphany 3.4	<input checked="" type="checkbox"/> Firefox 57.0	<input checked="" type="checkbox"/> Firefox 35.0	<input checked="" type="checkbox"/> Firefox 49.0
<input type="checkbox"/> Firefox 3.0	<input checked="" type="checkbox"/> Firefox 58.0	<input checked="" type="checkbox"/> Firefox 40.0	<input checked="" type="checkbox"/> Firefox 50.0
<input type="checkbox"/> Firefox 3.5	<input checked="" type="checkbox"/> Firefox 59.0	<input checked="" type="checkbox"/> Firefox 41.0	<input checked="" type="checkbox"/> Firefox 60.0
<input checked="" type="checkbox"/> Firefox 3.6	<input checked="" type="checkbox"/> Firefox 60.0		
<input checked="" type="checkbox"/> Firefox 4.0	<input checked="" type="checkbox"/> Firefox 61.0		
<input checked="" type="checkbox"/> Firefox 5.0	<input checked="" type="checkbox"/> Iceape 2.7		
<input checked="" type="checkbox"/> Firefox 6.0	<input checked="" type="checkbox"/> Iceweasel 38.0		
<input checked="" type="checkbox"/> Firefox 7.0	<input checked="" type="checkbox"/> Konqueror 4.14		
		<input checked="" type="checkbox"/> Firefox 61.0	<input checked="" type="checkbox"/> Chrome 41.0
		<input checked="" type="checkbox"/> Firefox 62.0	<input checked="" type="checkbox"/> Chrome 45.0
		<input checked="" type="checkbox"/> Firefox 63.0	<input checked="" type="checkbox"/> Chrome 48.0
		<input checked="" type="checkbox"/> Firefox 64.0	<input checked="" type="checkbox"/> Firefox 30.0
		<input checked="" type="checkbox"/> Firefox 65.0	<input checked="" type="checkbox"/> Firefox 31.0
		<input checked="" type="checkbox"/> Firefox 66.0	<input checked="" type="checkbox"/> Firefox 32.0
		<input checked="" type="checkbox"/> Opera 57.0	<input checked="" type="checkbox"/> Firefox 33.0
		<input checked="" type="checkbox"/> Opera 58.0	<input checked="" type="checkbox"/> Firefox 34.0
			<input checked="" type="checkbox"/> Firefox 35.0
			<input checked="" type="checkbox"/> Firefox 36.0
			<input checked="" type="checkbox"/> Firefox 37.0
			<input checked="" type="checkbox"/> Firefox 43.0
			<input checked="" type="checkbox"/> Firefox 44.0
			<input checked="" type="checkbox"/> Firefox 45.0
			<input checked="" type="checkbox"/> Firefox 46.0

Tomado de <http://browsershots.org/> accedido 1-mayo-2019

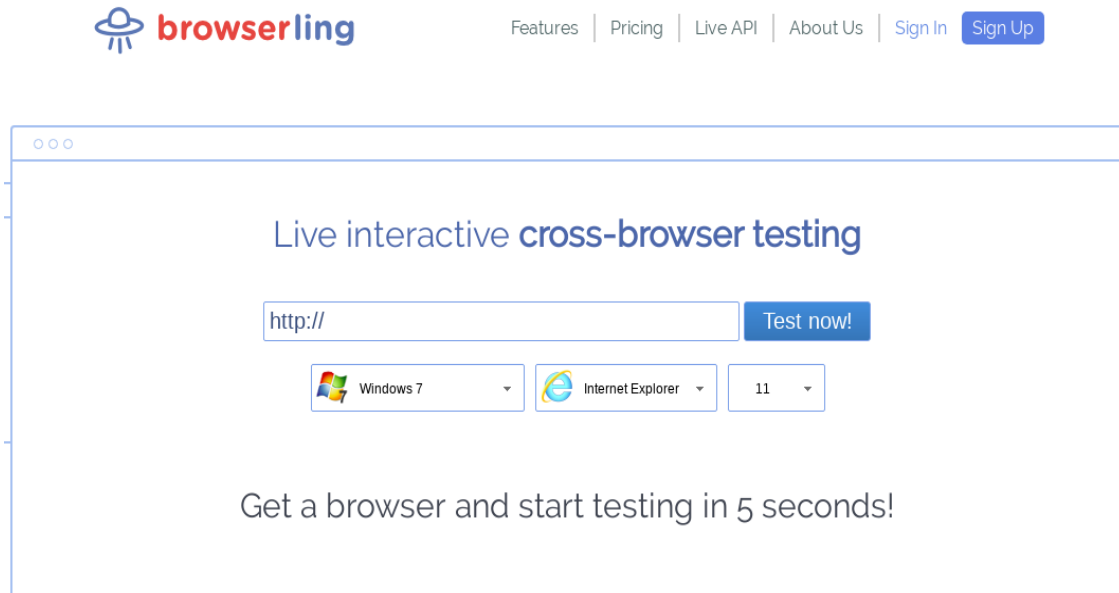
Browserling

Browserling es una herramienta para probar aplicaciones web en distintos navegadores y sistemas operativos. Tiene una plan de prueba(gratis) que ofrece funciones limitadas, una plan de desarrollador y un plan de equipo.

Las funcionalidades que ofrece son:

- Sesiones interactivas.
- Navegadores que se ejecutan en máquinas reales.
- Acceso a navegadores actualizados.
- Capturas de pantalla.
- Pruebas de diseño responsive.
- Navegación segura.

Entre otros (Tomado de <https://www.browserling.com/>).



Tomado de <https://www.browserling.com/> accedido 1-mayo-2019

Anexo U. Resumen proceso para grupo focal

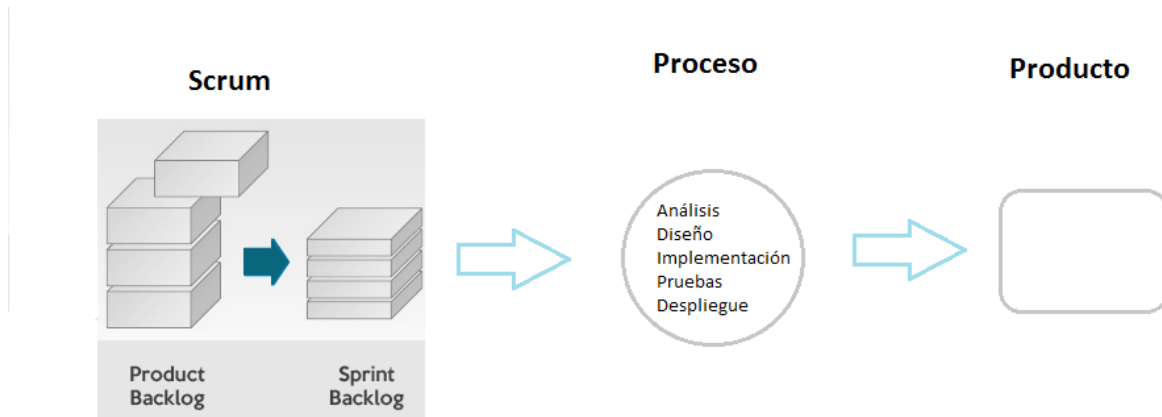
PROCESO ÁGIL DE DESARROLLO DE SOFTWARE PARA SOPORTAR EL DESARROLLO DE APLICACIONES WEB EN MICRO, PEQUEÑAS Y MEDIANAS EMPRESAS

Resumen

A continuación se propone un proceso de construcción de software llamado creación web empresarial(CWE) que permite a micro, pequeñas y medianas empresas desarrollar aplicaciones web. El proceso de construcción propuesto es compatible con SCRUM como proceso ágil de gestión de proyectos de desarrollo de software.

CWE plantea diferentes fases para cumplir su propósito. La fase de análisis se enfoca en la distribución de tareas a cada miembro del equipo, verificar que las historias de usuario cumplan con criterios definidos y generar una lista de aspectos a tener en cuenta para dar por terminado un requisito. La fase de diseño es la encargada de definir la arquitectura(patrones, vistas), herramientas necesarias para el desarrollo y la configuración de las misma. La fase de implementación se encarga de construir las funcionalidades del backend y frontend, además de esto se considera el desarrollo guiado por pruebas. La fase de pruebas es la encargada de realizar las pruebas necesarias a las funcionalidades tanto del frontend como del backend, dependiendo de qué se quiere probar se tendrá en cuenta la automatización, y por último se tiene la fase de despliegue, en la cual se tiene en cuenta la configuración de servidores y puesta en marcha de la funcionalidad construida en la fase de implementación.

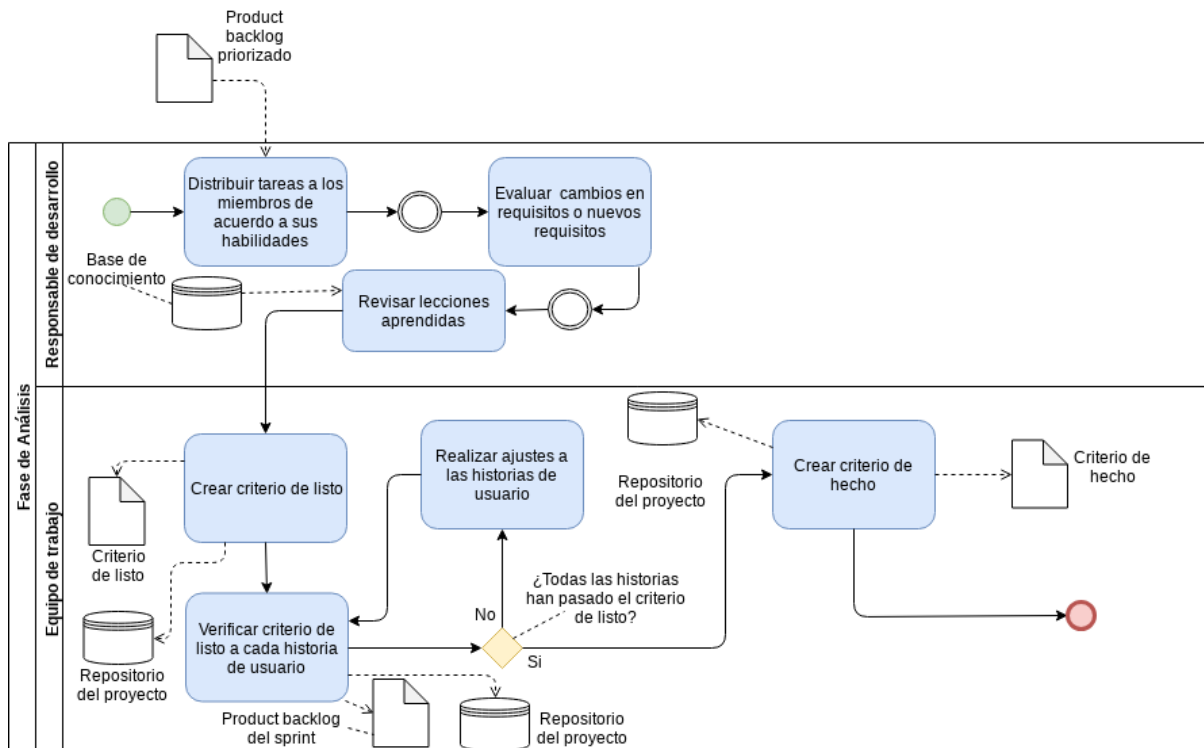
Para describir CWE se ha utilizado el patrón de procesos que plantea el proyecto Competisoft. En la figura 1 se muestra la visión general del proceso.



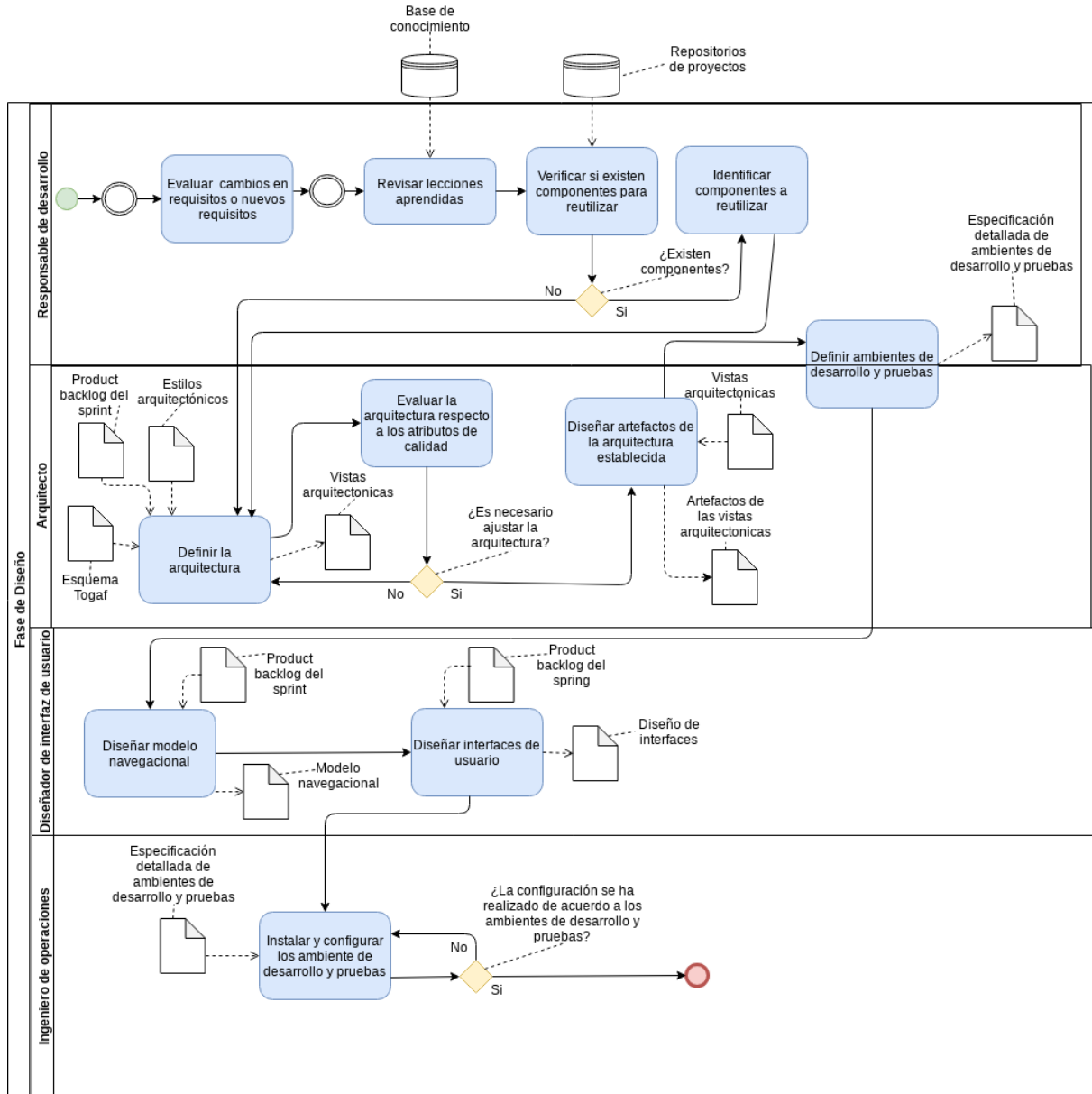
Visión general de la propuesta

Fases del proceso

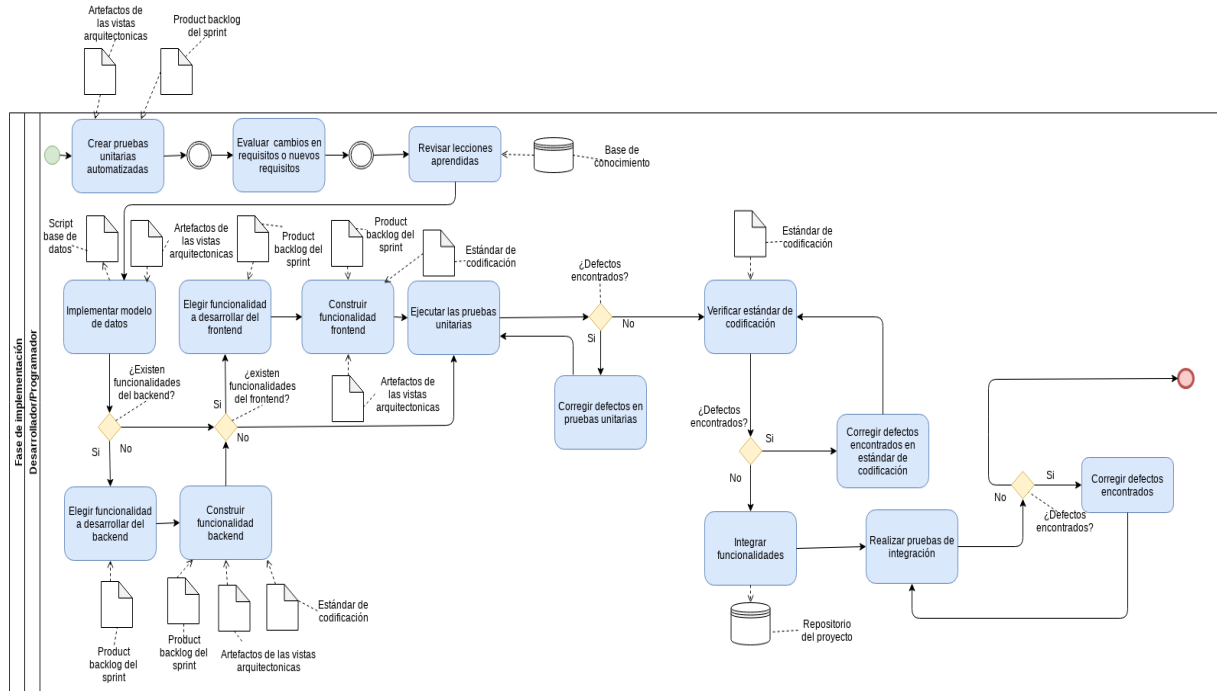
Fase de análisis



Fase de diseño

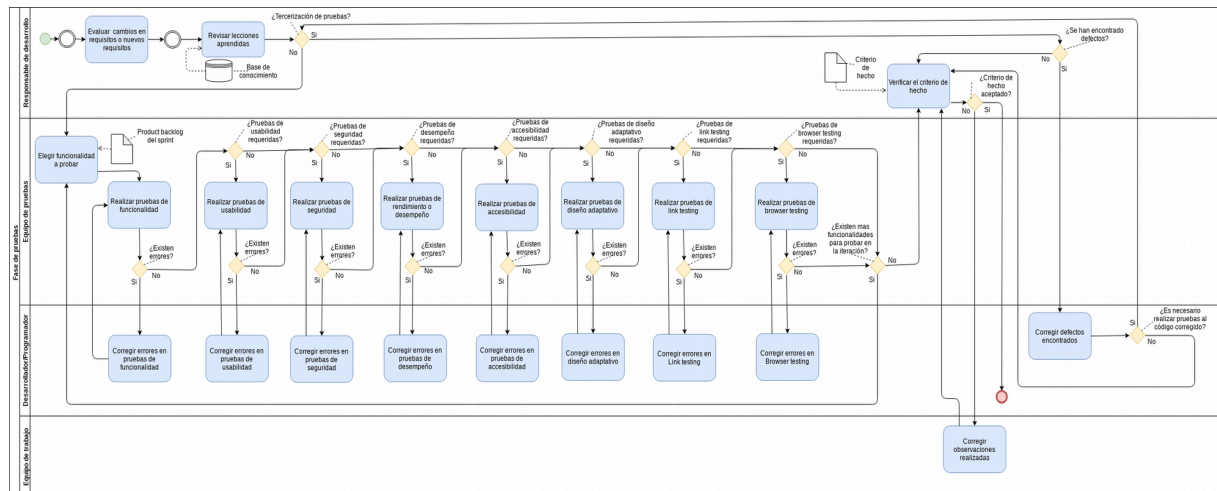


Fase de implementación



Text

Fase de pruebas



Fase de despliegue

