

**PROTOTIPO DE GESTIÓN DE CAMPAÑAS BATCH PARA  
LA EMPRESA OMNICON S.A.**



**ANDREA LÓPEZ AGUDELO  
JUAN DAVID MÉNDEZ ASTUDILLO**

**UNIVERSIDAD DEL CAUCA  
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES  
DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y CONTROL  
INGENIERIA EN AUTOMÁTICA INDUSTRIAL**

**POPAYÁN – CAUCA**

**Marzo, 2014**

**PROTOTIPO DE GESTIÓN DE CAMPAÑAS BATCH PARA  
LA EMPRESA OMNICON S.A.**



**ANDREA LÓPEZ AGUDELO  
JUAN DAVID MÉNDEZ ASTUDILLO**

**Director: PhD(C). OSCAR AMAURY ROJAS A.**

**UNIVERSIDAD DEL CAUCA  
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES  
DEPARTAMENTO DE ELECTRÓNICA, INSTRUMENTACIÓN Y CONTROL  
INGENIERIA EN AUTOMÁTICA INDUSTRIAL  
POPAYÁN - CAUCA**

**Marzo, 2014**

## **AGRADECIMIENTOS**

Damos gracias a Dios por permitirnos llegar a este punto de nuestras vidas y por habernos guiado de la mejor manera para lograr cada una de las metas trazadas hasta el momento, por darnos la fortaleza y perseverancia para afrontar los momentos difíciles a lo largo de la carrera y de nuestras vidas.

A nuestros padres por el apoyo incondicional brindado durante nuestras vidas, por creer siempre en nosotros, por ser un excelente ejemplo a seguir y por una vida llena de enseñanzas, amor, comprensión y esfuerzos que no alcanzaríamos a compensar y agradecer.

A nuestros abuelos y abuelas que han sido parte fundamental en nuestras vidas sirviéndonos de apoyo y motivación para lograr nuestros objetivos, aunque algunos de ellos ya no estén presentes físicamente para compartir nuestros logros, también son partícipes de lo que hoy en día somos como personas gracias a sus enseñanzas y su forma de ver la vida.

Al ingeniero Oscar Amaury Rojas por depositar su voto de confianza en nosotros y aceptar la dirección de nuestro trabajo de grado y por sus consejos, amistad y guía durante el desarrollo del mismo.

Al ingeniero Fabián Ruano por su amistad, apoyo y ayuda incondicional, por sus conocimientos y experiencia aportados a lo largo del desarrollo de este proyecto y por el tiempo dedicado.

A la empresa Omnicon y al ingeniero Fabián Yesid Vidal por la confianza depositada en los estudiantes de nuestra universidad y nuestra carrera permitiendo participar en sus procesos de desarrollo para la industria.

**Andrea López Agudelo, Juan David Méndez Astudillo**

## TABLA DE CONTENIDO

INTRODUCCIÓN .....	1
ORGANIZACIÓN DE LA MONOGRAFÍA .....	4
CAPITULO 1.....	5
1.1    MARCO TEÓRICO .....	6
1.2    ANÁLISIS FUNCIONAL DEL PROTOTIPO .....	12
CAPITULO 2.....	18
2.1    METODOLOGÍAS DE DESARROLLO SOFTWARE .....	18
2.1.1    METODOLOGÍAS TRADICIONALES .....	19
2.1.2    METODOLOGÍAS DE DESARROLLO AGIL.....	20
2.2    CRITERIOS DE SELECCIÓN DE LA METODOLOGÍA .....	22
2.3    APLICACIÓN DE LA METODOLOGÍA SELECCIONADA .....	23
2.3.1    PLANEACIÓN .....	26
2.3.2    CONSTRUCCIÓN .....	55
CAPITULO 3.....	68
3.1    DEFINICIÓN DE SEGMENTOS DE PROCESO, DE PRODUCTO, ESPECIFICACION DE EQUIPOS, PRODUCTOS Y MATRIZ DE TIEMPOS PARA EL CASO DE ESTUDIO .....	73
3.1.1    SEGMENTOS DE PROCESO BATCH .....	73
3.1.2    SEGMENTOS DE PRODUCTO BATCH.....	77
3.1.3    ESPECIFICACION DE LAS UNIDADES DEL PROCESO .....	80
3.1.4    ESPECIFICACIÓN DE PRODUCTOS.....	81
3.1.5    MATRIZ DE TIEMPOS .....	82
CAPITULO 4.....	85
4.1    APLICACIÓN DE LA VERIFICACIÓN Y VALIDACIÓN DEL PROTOTIPO .....	86
4.1.1    ANÁLISIS ESTÁTICO.....	89
4.1.2    ANÁLISIS DINÁMICO.....	91
CONCLUSIONES .....	102
BIBLIOGRAFÍA.....	104

## LISTA DE FIGURAS

Figura 1. Jerarquía Funcional de Actividades Multi-Nivel. Tomada de [1] .....	8
Figura 2. Modelo Genérico de Actividades. Modificada de: [1, 35] .....	9
Figura 3. Interfaces del modelo de actividad de Programación Detallada de la Producción Tomada de [1].....	10
Figura 4. Interfaces del Modelo de Actividades del Despacho de Producción Tomada de [1] .....	11
Figura 5. Dividir y unir programas de producción a programas detallados de la producción Tomada de [1].....	12
Figura 6. Metodología UML implementada .....	25
Figura 7. Distribución de los Casos de Uso en los ciclos de desarrollo .....	35
Figura 8. Diagrama de Casos de Uso .....	41
Figura 9. Arquitectura general de la empresa .....	42
Figura 10. Arquitectura Interna del prototipo y correspondencia a componentes físicos adaptada de [29].....	43
Figura 11. Modelo conceptual. Ver Anexo A .....	50
Figura 12. Modelo Entidad - Relación de la Base de Datos Ver Anexo A.....	53
Figura 13. Modelo Físico de Datos (Relacional) Ver Anexo A.....	54
Figura 14. Diagrama de Secuencia del Caso de Uso Generar Campañas Batch (Split y Merge). .....	56
Figura 15. Diagrama de interacción para la versión 1 y 2 de la Generación de Campañas Batch. Ver Anexo A.....	66
Figura 16. Diagrama de Diseño de Clases del Prototipo Ver Anexo A .....	67
Figura 17. Ejemplo de topología de proceso general contemplado en el desarrollo del prototipo.....	70
Figura 18. Topología definida del proceso de obtención de pintura a base de agua para la validación del prototipo .....	72
Figura 19. Enrutamiento y dependencia de materiales de los segmentos de proceso .....	77
Figura 20. Proceso de Obtención de Pintura a Base de Agua .....	84
Figura 21. Diagrama de pruebas de Caja Blanca y Negra .....	89
Figura 22. Programa de Producción Cargado .....	92
Figura 23. Prueba de Caja Blanca aplicada a la lectura del programa de producción editado bajo el esquema B2MML.....	93
Figura 24. Información del requerimiento de segmento seleccionado para la acción de Merge .....	95
Figura 25. Lista de remanentes del mismo ID para la unión de requerimientos .....	96
Figura 26. Campaña generada por el sistema con la acción de Merge Exacto .....	97
Figura 27. Información del Requerimiento de Segmento seleccionado para la acción de Split – Crear Remanente.....	98
Figura 28. Selección de la opción Crear Remanente .....	99

Figura 29. Campaña Batch Generada por la acción de Split - Crear Remanente.....	99
Figura 30. Resultado de solicitar los remanentes a la base de datos de la aplicación....	100

## LISTA DE TABLAS

Tabla 1. Asignación de las tareas de Programación Detallada de la Producción al Prototipo .....	16
Tabla 2. Asignación de las tareas de Despacho de la Producción al Prototipo .....	17
Tabla 3. Tabla de selección de metodologías. Modificada de [20] .....	23
Tabla 4. Requerimientos del Cliente (Omnicon).....	27
Tabla 5. Definición de Requerimientos de Programación Detallada de la Producción según el Estándar ISA-95.03 .....	27
Tabla 6. Definición de Requerimientos de Despacho de la Producción según el Estándar ISA-95.03.....	28
Tabla 7. Definición de Atributos del Sistema .....	29
Tabla 8. Relación de los Atributos del Sistema con las Funciones definidas.....	29
Tabla 9. Definición de Términos.....	30
Tabla 10. Caso de uso de alto nivel Inicio de Sesión .....	32
Tabla 11. Caso de Uso de Alto Nivel de Configuración Inicial.....	32
Tabla 12. Caso de Uso de Alto Nivel de Recibir Programa de Producción.....	32
Tabla 13.Caso de Uso de Alto Nivel de Generar Campañas Batch.....	32
Tabla 14. Caso de Uso de Alto Nivel de Generar Campañas Batch con asistencia software .....	33
Tabla 15. Caso de Uso de Alto Nivel de Despachar Campañas Batch.....	33
Tabla 16. Clasificación de los Casos de Uso .....	34
Tabla 17. Formato extendido del Caso Esencial de Uso Generar Campañas Batch v1 ...	40
Tabla 18. Lista de Categorías de Conceptos para el Caso de Uso Simplificado: Generar Campañas Batch v1.....	48
Tabla 19. Especificaciones del segmento de proceso global .....	73
Tabla 20. Especificación de Segmento de proceso de nivel 2 de Pintura Base.....	74
Tabla 21. Especificación de Segmento de Proceso de Nivel 2 de Pintura a Color .....	74
Tabla 22. Especificación de Segmento de Proceso de Nivel 2 de Pintura Filtrada .....	75
Tabla 23. Especificación de Segmento de Proceso de Nivel 3 de Pintura Base.....	75
Tabla 24. Especificación de Segmento de Proceso de Nivel 3 de Pintura a Color .....	75
Tabla 25. Especificación de Segmento de Proceso de Nivel 3 de Pintura Filtrada 1 .....	76
Tabla 26. Especificación de Segmento de Proceso de Nivel 3 de Pintura Filtrada 2 .....	76
Tabla 27. Especificación del segmento de producto Global .....	78
Tabla 28. Especificación del segmento de producto para la producción de Pintura Base	78
Tabla 29. Especificación del segmento de producto para la producción de Pintura a Color Roja .....	79
Tabla 30. Especificación del segmento de producto para la producción de Pintura a Color Amarillo.....	79
Tabla 31. Especificación del segmento de producto para la producción de Pintura a Color Verde.....	79

Tabla 32. Especificación del segmento de producto para la Producción de Pintura Filtrada.....	80
Tabla 33. Especificación de equipos o unidades del segmento de proceso de Pintura Base de Nivel 3.....	80
Tabla 34. Especificación de equipos o unidades del segmento de proceso de Pintura a Color de Nivel 3 .....	80
Tabla 35. Especificación de equipos o unidades del segmento de proceso de Pintura Filtrada 1 de Nivel 3.....	81
Tabla 36. Especificación de equipos o unidades del segmento de proceso de Pintura filtrada 2 de Nivel 3 .....	81
Tabla 37. Productos terminados e intermedios del proceso de Producción de Pintura a Base de Agua .....	82
Tabla 38. Matriz de Tiempo definida .....	83
Tabla 39. Lista de Chequeo para la inspección del Diseño .....	90
Tabla 40. Lista de Chequeo para la inspección del Código.....	91
Tabla 41. Requerimientos funcionales que cumple el caso de uso Generar Campañas batch.....	101

## INTRODUCCIÓN

Al tener en cuenta el continuo cambio en el entorno actual de la sociedad, donde existe una rápida introducción de productos en el mercado y una constante búsqueda de innovación en materia tecnológica, las industrias procuran mantenerse activas mejorando continuamente sus productos, procesos y servicios enfrentándose a un medio exigente y con normatividades cada vez más rigurosas. El desarrollo de este proyecto pretende realizar un aporte a las industrias manufactureras en las áreas de programación y control de la producción, ya que éstas se consideran el eje central del funcionamiento de toda empresa de fabricación; con una buena programación de la producción y un seguimiento continuo de la misma, las compañías lograrían corregir a tiempo posibles errores, aprovechar al máximo los recursos y el tiempo con el que cuentan de tal forma que les sea posible tener clientes satisfechos que atraerán a otros, contribuyendo a alargar su estancia en el mercado competitivo.

Teniendo en cuenta la importancia que tienen las tareas de programación y despacho de la producción dentro de una empresa manufacturera, con la implementación del proyecto se busca desarrollar un prototipo software (Omnicampaign) que sirva como un sistema de apoyo en la toma de decisiones del proceso de programación y control de la producción batch, operando en modo campañas permitiendo acciones tipo Split<sup>1</sup> [1] & Merge<sup>2</sup> [2] en los órdenes de producción y que a su vez sus funcionalidades y definiciones se basen en los estándares de automatización ISA-88.01, ISA-95.02 e ISA-95.03. El objetivo del prototipo gestor de campañas batch<sup>3</sup> [3] es lograr una reducción de tiempos de alistamiento y configuración, disminución de recambios y limpiezas, aprovechamiento máximo de la capacidad del equipo, entregas de producto a tiempo entre otros.

El prototipo gestor de campañas batch busca obtener la programación detallada y despacho de la producción, a partir de la recepción de un programa de producción generado desde la planeación y convertir éste a campañas batch asignadas a los recursos disponibles dispuestas para ejecutar.

---

<sup>1</sup> Split (dividir) es la acción que se ejecuta al manejar tamaños de batch variables lo que permite, según la capacidad de la línea de producción, el centro de trabajo o el equipo sobre el cual se vaya a ejecutar una orden del programa, dividir ésta de tal forma que se cumpla con lo requerido por el programa y se aproveche la capacidad disponible [1]

<sup>2</sup> Merge (unir) es la acción que se ejecuta al manejar tamaños de batch variables que permite la unión de diferentes órdenes de producción pertenecientes a una misma familia de producto. [2]

<sup>3</sup> Conjunto de batches de una misma familia de producto que se ejecutan consecutivamente en la misma unidad de producción. [3]

Una vez conocidas las responsabilidades que tienen la programación de la producción y el despacho de la producción según el estándar, se procede a delimitar el alcance del proyecto, teniendo claro el entorno específico o topología de la planta en el cual se pretende desarrollar, para esto es necesario considerar diversos aspectos de los procesos que influyen directamente en el modelado y complejidad de éste. Algunos de los elementos a tener en cuenta son [4]:

- **Topología del Proceso:** Se basa en definir la estructura general del proceso, es decir, si este es multietapa o de una sola etapa, multipropósito o multiproducto y si tiene unidades Paralelas o una sola unidad por etapa.
- **Asignación de equipo:** Fija o variable.
- **Tamaño del batch:** Fijo o variable (Split & Merge).
- **Tiempos de procesamiento de batch:** Fijos o variables (Dependientes del Tamaño del Batch/Unidad).
- **Patrones de Demanda:** Basados en Horizontes de Programación o por fechas de entrega.
- **Recambios:** Dependientes de la Unidad o de la Secuencia (Productos).

Partiendo de la definición de las condiciones del proceso sobre el cual se va a trabajar, se realiza un análisis funcional del prototipo a partir de las definiciones realizadas en los estándares ISA.88.01, ISA-95.02 e ISA-95.03 obteniendo así las funcionalidades básicas del prototipo.

El siguiente paso en el desarrollo es la especificación y diseño del prototipo la cual se realiza por medio de un modelado orientado a objetos, aplicando patrones de diseño y tomando como base la metodología de desarrollo por ciclos iterativos planteada en [5] que permiten reformar y avanzar en las funcionalidades del sistema en cada una de las etapas definidas de tal forma que se logre paso por paso el cumplimiento de los objetivos planteados.

De forma general el inicio de las actividades para el funcionamiento del prototipo está marcado por una configuración inicial de la información necesaria para que éste pueda trabajar de manera adecuada, la cual es realizada por el supervisor (usuario), definiendo los segmentos de proceso que representa la topología de la planta, equipos, productos, tiempos de producción entre otros. Una vez configurada dicha información, el proceso continúa con la recepción de un Programa de Producción editado bajo el esquema de integración *B2MML*<sup>4</sup> [6] propuesto por el WBF (*World Batch Forum*) que proviene del nivel de Logística y Negocios (*ERP*), el cual contiene los requerimientos de producción

---

<sup>4</sup> *B2MML (Business To Manufacturing Markup Language)* es un conjunto de esquemas XML empleados para implementar la estructura de los modelos de objetos del estándar ISA-95 [6].

planteados en un horizonte de tiempo semanal y que a su vez provee los requerimientos de segmento específicos a cada segmento del proceso que se desee programar. El operario se encarga de elegir un requerimiento de producción y a su vez un requerimiento de segmento específico para ser programado. A continuación el software procede a evaluar los equipos disponibles basados en su capacidad mínima, máxima, tipo de producto que está produciendo, entre otros factores y por último se encarga de realizar el cálculo del tamaño de las campañas batch según las acciones Merge y Split. De las divisiones de los batches o Split dadas por la falta de capacidad del mejor equipo disponible para producir toda la cantidad requerida se pueden generar remanentes de producción que son usados por el sistema para realizar una unión de requerimientos o Merge.

Una vez generada dichas campañas (lista de campañas) deben pasar primero por un proceso de aprobación por parte del supervisor de planta, el cual decide si las campañas serán programadas por el sistema, en caso de que no sean aprobadas se pueden realizar una serie de modificaciones manuales según el conocimiento y las decisiones del supervisor. Al ser aprobadas las campañas, el supervisor escoge los batches a despachar generando la lista de despacho bajo el esquema *BATCHML*<sup>5</sup> [7] y enviando estos al servidor Batch para que sean ejecutados en planta.

Debido a la complejidad que implica el desarrollo de la programación y despacho de la producción dentro de un marco general de desarrollo para una empresa de manufactura tipo batch, en el que se consideren cada una de las características y detalles que se pueden presentar, se ve necesario realizar una delimitación de las funcionalidades del prototipo con respecto a las características del proceso mencionadas anteriormente. Dentro de estas delimitaciones se encuentran: la programación de una sola etapa del proceso de producción por vez, la ejecución de varios productos pertenecientes a una misma familia, una asignación de equipos variable que depende de la disponibilidad y capacidad de los mismos, celdas de producción con unidades en paralelo, asignación del tamaño y tiempo de batch variable según las acciones Merge y Split, prioridad de programación dependiente de la disponibilidad de equipos basados en una serie de criterios de selección y la existencia de recetas de limpieza debido al cambio en el tipo de producto.

En el manejo de información para el desarrollo del prototipo no se tiene en cuenta una integración con inventario, por lo tanto el conocimiento de la existencia o no de materias primas para la producción están a cargo del operario. Además la integración con las otras

---

<sup>5</sup> BatchML (Batch Markup Language) es un conjunto de esquemas XML que implementan los modelos y terminología del estándar ISA-88 [7].

operaciones de manufactura como lo son calidad, mantenimiento y otra información relevante de la producción no fueron consideradas para esta versión del prototipo.

Dentro de la lógica del prototipo no se considera la secuencia de operaciones para la obtención del producto final, al realizarse la programación de etapa por etapa. Por lo tanto la decisión de cuál de los productos intermedios resultado de una determinada etapa del proceso de producción debe ser programada, la toma el usuario, teniendo en cuenta con cuales de los productos intermedios cuenta y por lo tanto el conocimiento del proceso de fabricación.

En el desarrollo no se tienen en cuenta los aspectos relacionados a la seguridad web del prototipo ni el manejo de sesiones de usuarios, privilegios y demás consideraciones que este conlleve.

Adicionalmente al considerar que el prototipo es un sistema de apoyo para realizar una programación detallada de la producción, en donde las decisiones que tome el usuario juegan un papel importante, se considera descartado abordar el proyecto como un problema de optimización.

## **ORGANIZACIÓN DE LA MONOGRAFÍA**

El desarrollo del presente trabajo se estructura de la siguiente forma: una introducción donde se plasma de forma general el tema central del estudio, el porqué de la investigación, la metodología empleada para la realización del mismo, las limitaciones del prototipo y la estructura de cada capítulo. En el capítulo 1 se introduce la “teoría clásica” de programación de la producción de procesos Batch y se relaciona con la teoría definida en el estándar ISA-95, además se realiza el análisis funcional de las actividades de programación detallada y despacho de la producción orientados al prototipo. En el capítulo 2 se presenta la metodología escogida para el diseño y especificación del sistema y cómo ésta se adapta y aplica al proceso y a las características particulares del mismo. El capítulo 3 presenta una descripción de las etapas del proceso correspondientes al caso de estudio elegido y basado en esto se enmarca el proceso según la topología definida para el desarrollo del prototipo. En el capítulo 4 se presenta la validación del prototipo donde se contemplan cada una de las funcionalidades definidas para el desarrollo e implementación de éste. Finalmente se muestran las conclusiones correspondientes y los anexos resultado del desarrollo del proyecto, evidenciando en ellos los resultados de la aplicación de la metodología elegida a los otros casos de uso planteados en el capítulo 2, los diagramas y modelos UML finales obtenidos, el resultado de los procesos de verificación y validación del prototipo y documentos finales para el usuario.

## **CAPITULO 1.**

### ***“Introducción a la programación de la producción de procesos batch y análisis funcional del prototipo en base a los estándares de automatización ISA-95.02 e ISA-95.03”***

Teniendo en cuenta que actualmente la dinámica de la economía y el mercado global es cambiante, que la competitividad cada vez es más alta, las normatividades son cada vez más rigurosas, la búsqueda de las industrias por estar a la vanguardia y lograr mantenerse activas es constante y que la participación de procesos batch dentro de sectores industriales como lo son los procesos alimenticios, químicos, cosméticos, polímeros, bioquímicos y farmacéuticos [8] ha aumentado gracias a la gran flexibilidad de este tipo de procesos que permiten tener una gran variedad de productos en una misma célula de proceso [9], [10]; se ha evidenciado que la demanda de sistemas de gestión automatizados y el número de investigaciones buscando resolver falencias de la industria manufacturera actual se ha incrementado.

Dentro de los diferentes aspectos concernientes al funcionamiento adecuado que exige la industria manufacturera, existe un punto de vital importancia sobre el cual recae una gran responsabilidad, éste es la planificación, programación y control de la producción. Dichas actividades conforman el eje central del funcionamiento de la manufactura de una empresa.

## 1.1 MARCO TEÓRICO

En la literatura se ha abordado de una forma considerable el problema de la planificación, programación y control de la producción. Son muchos los autores que han presentado diferentes enfoques con respecto a la forma de afrontar esta temática conocida como teoría clásica los cuales se evidencian en [11] - [12] y que a su vez se pueden dividir en 2 grupos que se diferencian teniendo en cuenta la aplicación de un criterio base e indispensable: la integración de dichos procesos. Es aquí donde lo expuesto sirve como base fundamental en el estudio de este problema y del cual se han desprendido otras investigaciones como [13], las cuales concluyen que los 5 aspectos a tener en cuenta en el proceso de planificación y control de la producción son los descritos brevemente a continuación [11]:

### a. **Planificación estratégica o a largo plazo.**

Esta actividad es muy importante dentro del proceso, ya que sirve como punto de partida de las siguientes actividades en la jerarquía, de manera general en esta actividad se plantea y se trabaja en los objetivos globales de la empresa a largo plazo, se establecen las prioridades competitivas basándose en la información obtenida de los pronósticos de demanda, planeación de mano de obra, introducción de nuevas tecnologías, inversión de capital, entre otros.

### b. **Planificación agregada o a mediano plazo.**

Esta actividad tiene como objetivo fundamental establecer los niveles de producción en unidades agregadas es decir de manera global manejando un horizonte de tiempo de alrededor de 1 a 6 meses, con lo cual se busca lograr cumplir con las necesidades impuestas en la anterior actividad de planificación a largo plazo.

### c. **Programación maestra de Producción (*MPS – Master Production Scheduling*) o a corto plazo.**

A medida que se va bajando en los niveles de la jerarquía el horizonte de tiempo disminuye de tal manera que una vez obtenida la planificación agregada de la producción el siguiente paso es desagregarla traduciendo el plan agregado a un plan detallado que establece la cantidad específica y las fechas exactas de fabricación de productos finales, por lo general disminuye el horizonte de tiempo a semanas.

#### **d. Programación de componentes.**

Este tipo de programación se encarga de llevar un seguimiento y control de la capacidad de equipos, al igual que de las materias primas y productos intermedios existentes de tal forma que se pueda conocer las limitaciones a la hora de llevar a cabo una programación maestra. Las técnicas más usadas son:

- Listas de capacidad
- Perfiles de Recursos

#### **e. Ejecución y control.**

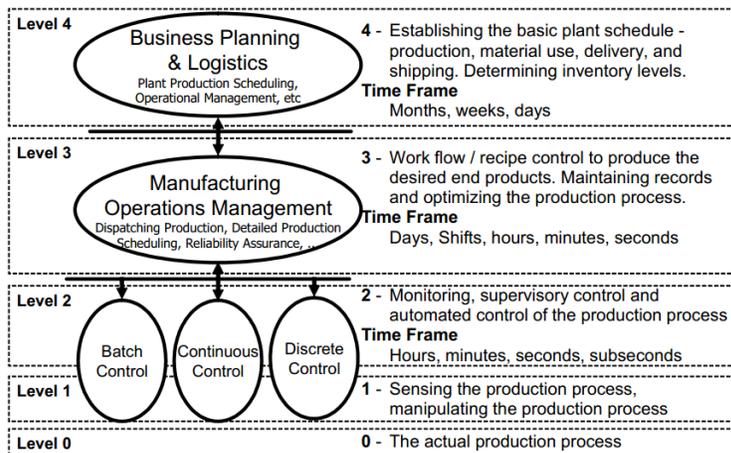
Por último en esta etapa se realiza la transformación del programa maestro de producción y la programación de capacidades y equipos, en órdenes específicas de trabajo las cuales son despachadas hacia un nivel inferior donde se encuentran las máquinas y/o talleres de procesamiento las cuales son asignadas a una orden en particular y atendidas por un empleado en específico, con lo cual se busca darle cumplimiento a todos los planes generados en los niveles superiores, conocido como el programa final de operaciones.

Aunque esta temática sea de gran importancia y se haya estudiado ampliamente en la literatura, la forma de abordar y aplicar estas tareas en el entorno empresarial se torna un poco difusa debido a la poca claridad en la delimitación de las tareas pertenecientes a cada una de las 3 actividades, como se evidencia en [14], en donde, en la práctica no es clara la separación entre un planificador, de un programador o un programador de un despachador.

Con el fin de unificar los diferentes conceptos trabajados por los autores anteriormente mencionados y de considerar la experiencia obtenida del desarrollo de proyectos de industrias como Honeywell, Foxboro, Yokogawa, Fisher Rosemount, Chevron entre otras [15]. ISA (*International Society of Automation*) decide proponer una serie de documentos estándar para el manejo de información generalizado. El resultado de esta iniciativa son los estándares ISA-88 e ISA-95 que brindan una terminología y un modelado estándar con un enfoque jerárquico aplicable a las industrias. Para este caso se tiene el modelo jerárquico que propone ISA-95 o la comúnmente llamada pirámide de automatización<sup>6</sup> (ver Figura 1), en donde al igual que se presenta en [16] hay 3 características claves que sirven para diferenciar cada una de las actividades, las cuales son: El horizonte de Tiempo, El proceso de toma de Decisiones y el Contexto.

---

<sup>6</sup> ANSI/ISA-95.00.03 Figura 1. Jerarquía Funcional de Actividades Multi- Nivel



**Figura 1. Jerarquía Funcional de Actividades Multi-Nivel.**  
Tomada de [1]

En el nivel más alto (4, *Planeación de Negocios y Logística*) se encuentra la Planificación o programación de alto nivel que maneja un contexto de gestión administrativa, establece un programa de producción inicial teniendo en cuenta la demanda de producto, recursos e inventarios y además maneja un horizonte de tiempo largo entre días, meses, semanas o incluso años. Al considerar la influencia de la teoría clásica en las definiciones de ISA es posible evidenciar una similitud entre las tareas del nivel de logística y negocio con las actividades de *Planificación estratégica o a largo plazo* y *Planificación agregada o a mediano plazo* definidas en la teoría clásica.

En el nivel intermedio (*Nivel 3, Administración de Operaciones de Manufactura*) se encuentra el conjunto de actividades que se encargan de administrar la producción dentro de un contexto de administración orientada hacia al proceso con un horizonte de tiempo mucho más corto dado por días, turnos, horas, minutos y segundos. En este conjunto de actividades se encuentra la programación detallada de la producción, la cual toma como base la programación de producción de alto nivel (nivel 4) y a partir de esta genera un programa mucho más detallado disminuyendo el horizonte de tiempo y asignando específicamente la programación futura de alto nivel a órdenes reales de producción sobre los recursos disponibles en el piso de planta, teniendo en cuenta un criterio de desempeño que puede variar según el tipo de operación y objetivos de producción de la empresa, generando así un documento normalmente representado en un diagrama de Gantt llamado programa detallado de producción que es correspondiente al *Programa Maestro de Producción (MPS)* en la teoría clásica y que a su vez hace uso de la *programación de componentes* que en el estándar hace referencia a la disponibilidad de recursos e información de la capacidad de programación de la producción. Dicho documento es la entrada de la siguiente actividad en el proceso, el despacho de la producción, la cual se encarga de

administrar el flujo de producción por medio de la lista de despacho, cuyas órdenes son enviadas a los niveles inferiores 2, 1, 0; esta última actividad es correspondiente en la teoría clásica a la *Ejecución y control* [1].

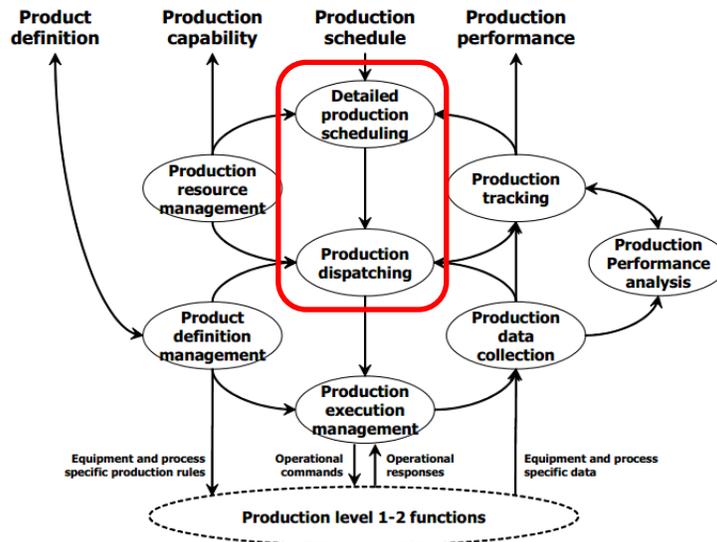


Figura 2. Modelo Genérico de Actividades. Modificada de: [1, 35]

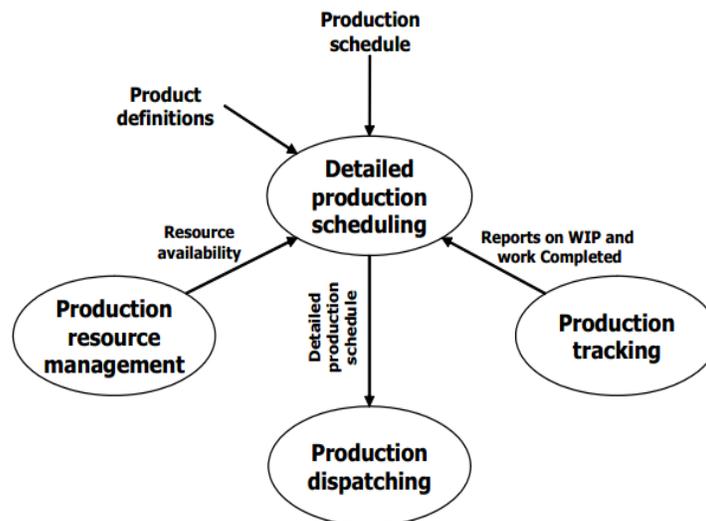
Las tareas de programación, programación detallada y despacho de la producción son indispensables para el progreso de una empresa, ya que siendo bien administradas traen un sin número de beneficios como el aprovechamiento óptimo de recursos, flexibilidad, tiempo de espera mínimos, eliminación de cuellos de botella, mejoramiento de la administración operacional del producto como la calidad, reducción de tiempos de entrega, entre otras [17]; lo cual hace que el estudio de éstas sea realmente amplio. Es por eso que el desarrollo del presente trabajo se enfoca únicamente en las operaciones correspondientes al nivel MES (*Manufacturing Execution Systems*) en cuanto a producción se refiere, teniendo en cuenta el modelo general de actividades propuesto por el estándar ISA-95.03 en las actividades de programación detallada y despacho de la producción, mostrado en la Figura 2.

Según el estándar la programación detallada de la producción es definida como “*la colección de actividades que toman la programación de la producción y determinan el uso óptimo de los recursos locales para alcanzar los requerimientos de la programación de la producción*” [1]

La programación detallada es un asunto crítico en las operaciones de proceso y es crucial para mejorar el desempeño de la producción. Para los procesos Batch la

programación a corto plazo se encarga de la asignación de un conjunto de recursos limitados sobre un horizonte de tiempo para la manufactura de uno o más productos siguiendo un r cipe de Batch. Han habido significantes esfuerzos de investigaciones en la pasada d cada en esta  rea en el desarrollo de enfoques  ptimos como se observa en [4], [18].

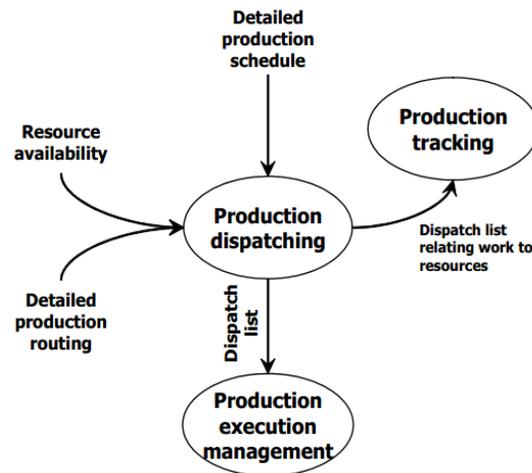
Sobre la programaci n detallada de la producci n recae una responsabilidad muy grande dentro del funcionamiento de la empresa, pues al pertenecer al eje central de las operaciones de producci n, siempre debe tener en cuenta la informaci n pertinente en el momento adecuado con el fin de realizar una toma de decisiones acertada, es por esto que para poder realizar la programaci n detallada de la producci n el est ndar ISA-95.03 provee el modelo de la Figura 3, con sus respectivas interfaces.



**Figura 3. Interfaces del modelo de actividad de Programaci n Detallada de la Producci n Tomada de [1]**

Como se puede observar en la Figura 3 mostrada anteriormente para poder realizar un correcto programa detallado de producci n es necesario como primera medida tener el programa de producci n que es enviado desde el nivel 4 (log stica y negocios) el cual contiene informaci n relacionada con los productos a ser elaborados, la cantidad a ser producida, entre otros [19]. Adem s de tener los requerimientos de producci n es necesario tener conocimiento tanto de las definiciones del producto o r cipes, la disponibilidad de recursos para poder realizar la programaci n sin exceder capacidades e informaci n de desempe o de la producci n para tener una medida de la ejecuci n de lo planeado. Una vez se haga uso de toda esta informaci n, se genera un documento llamado programa detallado de producci n el cual es enviado hacia la

siguiente actividad del proceso o despacho de la producción que de la misma manera posee su modelo y sus respectivas interfaces como se muestra en la Figura 4.



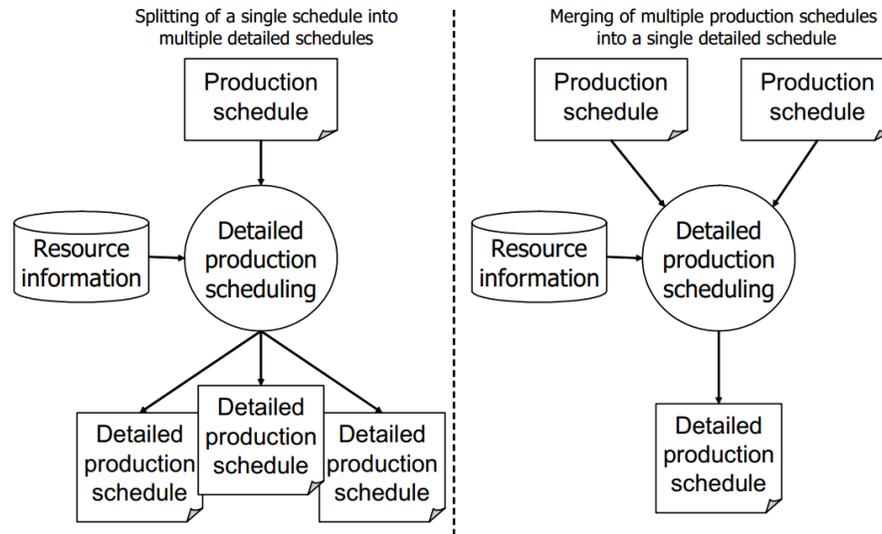
**Figura 4. Interfaces del Modelo de Actividades del Despacho de Producción Tomada de [1]**

El despacho de la producción se encarga específicamente de emitir las órdenes de producción identificadas en el programa detallado, la liberación de recursos para el inicio de las órdenes de trabajo, mantener el estado de estas, la recepción de información de las otras operaciones de manufactura como calidad e inventario con el fin de manejar en compañía del programa detallado condiciones no anticipadas de inhabilidad futura de recursos o pruebas de calidad programadas, entre otras tareas importantes para el proceso.

Para la realización de las tareas del despacho de la producción es necesario tener la información adecuada que sirva como soporte para realizar el proceso de toma de decisiones pertinente lo cual se logra gracias a las entradas de información representadas en los flujos que brindan la disponibilidad de recursos y la ruta detallada de la producción con el fin de que la emisión y algunas asignaciones realizadas por esta actividad sean acertadas. Teniendo en cuenta esta información, la actividad genera un documento llamado, lista de despacho, en el cual se encuentran las órdenes de trabajo listas para ser elaboradas, definiendo los centros y unidades de trabajo en las cuales van a ser ejecutados y su respectiva especificación de tiempo o el evento que da inicio a la actividad como lo definió el programa detallado [1].

En busca de brindar flexibilidad a la hora de programar los batches y así mejorar indicadores clave de desempeño o KPI's (*Key Performance Indicators*) como WIP

(*Work In Process*), Tiempo de Entrega, disminución de costos, entre otros, se hará uso de las operaciones de *Split and Merge* (Ver Figura 5) evidenciadas en ISA-95.03, que permiten reacomodar órdenes de producción bajo características comunes o según la necesidades planteadas en la programación de alto nivel.



**Figura 5. Dividir y unir programas de producción a programas detallados de la producción Tomada de [1]**

## 1.2 ANÁLISIS FUNCIONAL DEL PROTOTIPO

Una vez conocidas las labores que realiza la planificación, programación de la producción y finalmente la ejecución y control en la teoría clásica o la programación, programación detallada y despacho de la producción respectivamente según el estándar ISA-95.03, se procede a analizar cuáles de estas actividades son aplicables para el desarrollo del prototipo.

Teniendo en cuenta que uno de los objetivos del prototipo es la programación de la producción a corto plazo, y que según el nivel jerárquico propuesto por ISA-95.03 éste se ubica en el nivel 3 de la pirámide de automatización, a continuación se realiza el estudio de las funcionalidades de la programación detallada y el despacho de la producción con respecto al prototipo.

El estándar define 5 tareas asociadas a la programación detallada de la producción, las cuales se presentan a continuación:

- ***La creación y mantenimiento de un programa detallado de producción:***  
Esta actividad es indispensable para la implementación del prototipo, pues es esta en la que se crean el conjunto de campañas batch que al ser ejecutadas dan cumplimiento con el requerimiento de segmento deseado y con la planificación de la producción planteada en el programa de producción recibido.
- ***Comparación de la producción actual con la planteada:***  
Esta actividad se considera parte del nivel de logística y negocio, pues el resultado de la producción actual se consigna en un documento conocido como el desempeño de la producción, el cual hace parte del desarrollo de otra actividad del modelo genérico de actividades de ISA-95.03 por lo cual no se tiene en cuenta para el prototipo de gestión de campañas batch.
- ***Determinar la capacidad comprometida de cada recurso para el uso de la función de administración de recursos de producción:***  
Para esta función se muestra un conjunto de información relevante para el usuario de la capacidad disponible y comprometida de cada recurso, es decir se determina las unidades de producto comprometidas y el tiempo que este recurso hace uso para cumplirlas, de la misma forma se verifica su disponibilidad y las posibles unidades de producto a ejecutar en el tiempo disponible, teniendo así una medida que permite realizar un análisis sobre las futuras producciones a ser ejecutadas.
- ***Obtener información para la administración de operaciones de mantenimiento, administración de operaciones de calidad y administración de operaciones de inventario:***  
Esta tarea no se tiene en cuenta en la implementación del prototipo pues la integración horizontal con otros sistemas de administración de las operaciones de manufactura como los son calidad (LIMS), inventario (WMS) y mantenimiento exceden el alcance del proyecto.
- ***Ejecución de simulaciones que pasa sí. Esta tarea puede incluir actividades tales como cálculo del tiempo de entrega de producción o tiempo final de completada la producción para cada requerimiento de producción provista por las funciones de nivel 4, determinando recursos cuello de botella para cada periodo, y asegurando el tiempo de la disponibilidad de una producción futura para una particular producción:***  
Esta tarea no es implementada en el desarrollo del prototipo debido a la forma de abordar este tipo de problemas, ya que para lograr las simulaciones es necesario abordar el problema desde el punto de vista de optimización, por lo

cual cambia el enfoque de desarrollo con el que se afronta el trabajo además de que esta función se sale del alcance, impacto y dimensionamiento del prototipo. Para el desarrollo de este tipo de actividades se encuentran disponibles una serie de soluciones software especializadas en el tema.

El estándar ISA-95.03 define 10 tareas que desempeña el despacho de la producción, las cuales se muestran a continuación:

- ***Emitir órdenes de trabajo de producción como fueron definidas por el programa:***  
Esta actividad se ve reflejada en la integración de los niveles MES (3) y piso de planta (2, 1, 0), al enviar las campañas y sus batches tal cual como fueron definidas en el programa detallado creado por el prototipo hacia el administrador batch encargado de administrar y controlar la ejecución de la producción y así dar cumplimiento con el programa de producción recibido desde el nivel de logística y negocios (4).
- ***Asignación de recursos locales para la producción, cuando estos no se identifican como parte de la programación detallada de la producción:***  
Dentro del desarrollo del prototipo no se tuvo en cuenta esta tarea, pues la asignación de los recursos tiene influencia por las decisiones del usuario.
- ***Liberar recursos locales para comenzar órdenes de trabajo:***  
Esta actividad se ve reflejada en el proceso de toma de decisiones que realiza el usuario al elegir los recursos disponibles sobre el cual se van a despachar las campañas batch.
- ***Manejo de las condiciones no anticipadas en la programación detallada de la producción.***  
Como se evidencia en la información plasmada en el estándar dentro de las actividades consideradas como no anticipadas se encuentran el manejo de redes de trabajo con la comunicación hacia operaciones de mantenimiento, calidad e inventario, por lo tanto esta funcionalidad no aplica para el desarrollo del proyecto.
- ***Mantener el estado de las órdenes de trabajo:***  
En el desarrollo del prototipo se realiza el seguimiento del estado de las campañas programadas y los batches a los que estas pertenecen manteniendo los estados definidos en el estándar ISA-88.01.  
Esto se logra por medio de la funcionalidad de gestión del prototipo con la cual permite realizar el seguimiento de la ejecución de la producción, manteniendo

información de las campañas terminadas, en proceso, de la existencia de remanentes o no, que en conjunto permiten evaluar el estado de las mismas y de la producción en general.

- ***Asegurar que las limitaciones del proceso y las órdenes bajo el nivel de detalle de la programación detallada son logradas en producción. Esto toma lugar después de que el programa es creado, pero antes de que sus elementos son ejecutados:***

Dentro de las funcionalidades del prototipo en la generación del programa detallado por medio de las campañas batch, se tienen en cuenta las limitaciones puestas por el nivel de logística y negocios plasmados en el programa de producción recibido y el programa detallado generado, como lo es la fecha de inicio y de finalización del programa de producción, de batches y campañas, además después de generar las campañas batch en su respectivo programa detallado de producción y antes de ser despachados se muestra al usuario la información completa de los batches para que sean verificados por última vez.

- ***Informar a la programación detallada cuando un evento no anticipado resulta en la inhabilidad de lograr los requerimientos de programación:***

Dentro del proceso de programación y despacho de campañas batch se informa al usuario los estados de las campañas que se encuentran en el servidor batch y por medio de este informando así cualquier problema que haya causado la inhabilidad de completar o continuar una campaña batch.

- ***Recibir información desde la administración de operaciones de calidad que indica condiciones no anticipadas que se pueden relacionar con eventos del programa:***

Al igual que la actividad planteada en la programación detallada de la producción esta corresponde a una integración horizontal, lo cual excede el alcance del proyecto.

- ***Recibir información desde la administración de recursos de producción sobre disponibilidad futura de recursos que se pueden relacionar con eventos del programa:***

Al igual que la actividad planteada en la programación detallada de la producción esta corresponde a una integración horizontal, lo cual excede el alcance del proyecto.

- **El envío o puesta a disposición de la lista de despacho especificando las actividades de producción a ser realizadas:**

Esta tarea se realiza por parte del prototipo en la integración que éste presenta con los niveles de piso de planta, enviando la lista de despacho generada con la asignación de batches, cantidades, tipo de producto, fechas de inicio y finalización, entre otra información.

A continuación, en las siguientes tablas (Tabla 1 y Tabla 2) se presenta un resumen de las actividades propuestas en el estándar aplicadas al prototipo de gestión de campañas batch.

	<b>TAREAS</b>	<b>OMNICAMPAIGN</b>
<b>PROGRAMACION DETALLADA DE LA PRODUCCION</b>	Crear y mantener un programa detallado de la producción.	<b>X</b>
	Comparar la producción actual con la producción planeada	<b>Función de nivel 4 (Logística &amp; Negocio)</b>
	Determinar la capacidad comprometida de cada recurso para el uso de la función de administración de los recursos de producción.	<b>X</b>
	Obtener información de las operaciones de mantenimiento, calidad, e inventario.	<b>No Aplica</b>
	Ejecución de las simulaciones tipo 'Que pasa si'.	<b>No Aplica</b>

**Tabla 1. Asignación de las tareas de Programación Detallada de la Producción al Prototipo**

	<b>TAREAS</b>	<b>OMNICAMPAIGN</b>
<b>DESPACHO DE PRODUCCION</b>	Emitir órdenes de trabajo de producción como fueron definidas por el programa.	<b>X</b>
	Asignación de recursos locales para la producción, cuando estas no son identificadas como parte del programa detallado de la producción.	<b>Usuario</b>
	Liberar recursos locales para comenzar órdenes de trabajo.	<b>Usuario</b>
	Manejo de las condiciones no anticipadas en programación detallada de la producción	<b>No Aplica</b>

<b>DESPACHO DE PRODUCCION</b>	Mantener el estado de las órdenes de trabajo. Ejemplo: Aprobado, Corriendo, Pausado o cancelado	<b>X</b>
	Asegurar que las limitaciones del proceso y las órdenes bajo el nivel de detalle de la programación detallada son logradas en producción. Esto toma lugar después de que el programa es creado, pero antes de que sus elementos son ejecutados.	<b>X</b>
	Informar a la programación detallada cuando un evento no anticipado resulta en la inhabilidad de lograr los requerimientos de programación.	<b>X</b>
	Recibir información desde la admón. operaciones de calidad que indica condiciones no anticipadas que se pueden relacionar con eventos del programa.	<b>No Aplica</b>
	Recibir información desde la administración de recursos de producción sobre disponibilidad futura de recursos que se pueden relacionar con eventos del programa.	<b>No Aplica</b>
	El envío o puesta a disposición de la lista de despacho producción especificando las actividades productivas que se realizan	<b>X</b>

**Tabla 2. Asignación de las tareas de Despacho de la Producción al Prototipo**

Una vez conocidas y estudiadas las funcionalidades del estándar ISA-95.03 con respecto a la programación detallada y despacho de la producción, se logra definir cuáles de estas se implementaran en el prototipo como se precisan en las tablas anteriores, continuando con el análisis, diseño y especificación del mismo que se tratará en el capítulo a continuación.

## CAPITULO 2.

### *“Diseño y especificación del sistema”*

Una vez definido que el área de trabajo para el desarrollo de este proyecto es la programación detallada y despacho de la producción dentro de la industria tipo batch la cual se caracteriza por la producción en modo campañas y la posible aplicación de acciones tipo Merge y Split, se procede al diseño y a la especificación general del prototipo que permitirá obtener una base sólida para la posterior programación de éste.

Un paso importante en el planteamiento y ejecución de cualquier proyecto, es definir una metodología que sirva como guía durante todo el proceso de diseño y desarrollo lo cual permita planificar, estructurar y gestionar el proyecto de una manera que se ajuste a las necesidades específicas que éste pueda tener. Según esto se hace una búsqueda de las metodologías de desarrollo que más se adapte a las necesidades, con el fin de determinar la opción adecuada para afrontar el proyecto.

#### **2.1 METODOLOGÍAS DE DESARROLLO SOFTWARE**

Teniendo en cuenta que la tasa de éxito de los proyectos de desarrollo software puede ser incrementada según la correcta escogencia de una metodología que se adapte a las características específicas de cada proyecto, la adopción de estas por parte de los desarrolladores en los últimos años se ha convertido en un paso indispensable para abordar un proyecto.

Dentro de las investigaciones realizadas en los diferentes tipos de metodologías se ha logrado diferenciar unas de otras según su estructura y enfoque, por un lado se encuentran metodologías que parten de un análisis exhaustivo, con una planificación y requerimientos bien elaborados y especificados desde los inicios del proyecto

conocidas comúnmente como “*Metodologías Tradicionales o Peso Pesado*” y por otro lado se tienen las metodologías de desarrollo “*Suaves o Ágiles*” que se enfocan en una producción de partes entregables del software en tiempos cortos y no dedican tanto tiempo en la definición de requerimientos, análisis y diseño en la parte inicial pues una de sus características es la flexibilidad y adaptación al cambio a lo largo de la ejecución del proyecto. [20]

## **2.1.1 METODOLOGÍAS TRADICIONALES**

### **2.1.1.1 RUP (*Rational Unified Process*)**

RUP es un amplio marco de trabajo que provee buenas prácticas probadas eficientemente en problemas de desarrollo de la vida real, muy efectiva en la administración de proyectos y la creación de sistemas de información, brindando de esta manera información detallada al grupo de desarrolladores y permitiendo un análisis profundo del problema desde sus etapas iniciales. [21]

La metodología RUP se divide en 4 etapas principales que son: inicio, elaboración, construcción y transición, dentro de las cuales se realizan una serie de iteraciones donde se obtienen artefactos entregables o consumibles dentro del mismo proceso, que abarcan diferentes disciplinas como lo son modelado de negocio, definición de requerimientos, análisis y diseño, implementación, prueba, despliegue, configuración y administración de cambios, administración del proyecto y entorno. [20]

Análisis detallados de esta metodología donde se presentan ventajas y desventajas de su implementación se muestran en [22] y [23].

### **2.1.1.2 Desarrollo de ciclos iterativos (*Metodología Craig Larman*)**

La metodología propuesta por el autor Craig Larman en su libro “*UML y Patrones Introducción al análisis y diseño orientado a objetos*” toma como base la metodología RUP mencionada anteriormente y la redefine planteando una posible forma de utilizar ésta de una manera “ágil”.

En el libro se plantea un desarrollo por ciclos iterativos definiendo en cada ciclo una serie de pasos los cuales se repiten a lo largo del proceso, se plantea que una de las mayores ventajas de utilizar un ciclo de vida iterativo es que el agrandamiento y perfeccionamiento es secuencial.

La metodología plantea 3 pasos de macro nivel los cuales son Planeación, Construcción y Aplicación, que a su vez contienen sus respectivos puntos específicos a cada etapa y que corresponden a una parte importante del proyecto completo, en la fase de planeación se levanta la información inicial necesaria para el proceso como lo es el análisis de requerimientos y se hace uso del Lenguaje de Modelado Unificado (UML) para dar inicio a un análisis preliminar del problema, en la fase de construcción se desarrollan los ciclos iterativos cuya organización depende de la evaluación realizada previamente en la etapa de planeación donde se abordan un número de requerimientos relativamente pequeños los cuales pasan por los procesos de análisis, diseño, construcción y pruebas [5]. Por último la etapa de aplicación es donde se da la transición de la construcción a su implementación y de su implementación a su uso.

También en [5] se presenta un ejemplo de aplicación de su metodología en un caso de la vida real, mostrando detalladamente el proceso de aplicación y las ventajas que ésta trae.

## 2.1.2 METODOLOGÍAS DE DESARROLLO AGIL

### 2.1.2.1 XP (*eXtreme Programming*)

*eXtreme Programming*, o programación extrema es considerada una metodología ágil de desarrollo propuesta por primera vez por Kent Beck en el año 1999 en su libro *Extreme Programming Explained: Embrace Change*, el principal cambio propuesto por esta metodología que difiere de las tradicionales, es que trata de hacer el proceso de diseño y desarrollo más adaptable basados en la previsibilidad, es decir que se considera que los cambios de requisitos en la mitad del proceso es algo inevitable, por lo cual se evita invertir demasiado tiempo en la definición de requerimientos lo cual hace necesario poder responder y adaptarse de forma rápida a dichas situaciones.

En la programación extrema se definen 5 valores clave que sirven como guía para dicha metodología [24], estos son:

- **Simplicidad**  
Propone simplificar el diseño y documentación para acelerar el desarrollo y así facilitar el mantenimiento.
- **Comunicación**  
La comunicación es una parte fundamental del desarrollo utilizando este tipo de metodología, esta propone un desarrollo por parejas lo

cual hace que durante todo el proceso el grupo de desarrolladores este comunicándose activamente manteniendo una constante y sana discusión de los aspectos técnicos del desarrollo, permitiendo así encontrar la mejor forma de realizar las cosas.

- **Respeto**

Éste es necesario en cualquier proceso de desarrollo e imperativo al aplicar esta metodología, ya que todos los resultados obtenidos son gracias al trabajo en equipo lo que demanda un buen ambiente de trabajo.

- **Retroalimentación**

Éste valor define una interacción constante entre el cliente y el equipo de desarrollo, al cual se le presentan las pruebas de desempeño de las partes software desarrolladas desde el inicio del proceso, permitiendo tener una opinión inmediata del usuario sobre los desarrollos y así realizar cambios sugeridos en un tiempo corto.

- **Valentía**

La metodología define la valentía haciendo alusión al coraje que deben tener los programadores para ser capaces de responder a los imprevistos que se presentan a diario durante el desarrollo del proceso.

Actualmente la programación extrema ha evolucionado agregando muchos más ítems con el fin de mejorar la metodología, como se muestra de una manera amplia en [25].

### 2.1.2.2 SCRUM

Es un conjunto de principios y prácticas pertenecientes a un marco de trabajo ágil que ayudan a obtener productos en tiempos cortos, aplicable a diferentes áreas de trabajo, pero con un gran énfasis en el desarrollo de software.

El uso de SCRUM brinda una serie de ventajas evidenciadas en la obtención de una realimentación rápida, mejoramiento continuo y adaptación eficaz al cambio. De manera muy breve se muestra a continuación el proceso de la metodología.

Los actores involucrados en la metodología son, el *Product Owner* (Dueño del Producto), *Scrum Master* (Facilitador), Equipo de desarrollo, *Stakeholders* (Clientes, Proveedores, etc.) y administradores.

El proceso inicia cuando el dueño del producto crea una lista de requerimientos priorizada llamada *Product Backlog*, a continuación se

realiza una reunión de planeación de los *Sprints* (Ciclos de Desarrollo durante el proceso que comúnmente tienen una duración de 2 a 4 semanas) en donde se toma el *Product Backlog* creado del cual se extrae una pequeña parte de los requerimientos que mayor prioridad tengan y se decide cómo se van a implementar. El equipo de desarrollo se encarga de completar el requerimiento seleccionado en el tiempo determinado por un Sprint, cada día de trabajo de dicho Sprint realizan una reunión de una duración alrededor de 15 minutos donde se discuten los avances del proyecto y problemas encontrados, el Scrum Master mantiene al equipo enfocado en las tareas a desarrollar, guía en la resolución de problemas y revisa las estadísticas de productividad del desarrollo del trabajo. Al terminar el Sprint se obtiene un producto funcional entregable al cliente y se cierra el ciclo con un análisis de retrospectiva de éste, el proceso continúa de la misma manera con la realización de otros Sprints en donde se trabajan los requerimientos faltantes en el *Product Backlog*. [26]

## 2.2 CRITERIOS DE SELECCIÓN DE LA METODOLOGÍA

Basándose en la investigación realizada en [20], donde se analizan diferentes puntos claves en la selección de una metodología para el desarrollo de software se muestran 10 factores que afectan la selección los cuales se presentan a continuación y además, se añaden otros 2 factores importantes extraídos de la interacción con el cliente (*Omnicon*) y se aplican dando un número de evaluación de 0 a 5 a las 4 metodologías opcionales analizadas previamente para este proyecto, consignando en la Tabla 3 los resultados de dicha aplicación.

Los criterios son:

- a) Claridad de los requerimientos iniciales.  
(*Requerimientos claros: 5; Requerimientos poco claros: 0*)
- b) Precisión inicial de la estimación del costo y el tiempo de desarrollo.  
(*Buena estimación del costo: 5; Mala estimación del costo: 0*)
- c) Incorporación de cambios a los requerimientos en el proceso de desarrollo.  
(*Buena adaptación a los cambios: 5; Poca adaptación a los cambios: 0*)
- d) Obtener versiones funcionales del sistema durante el proceso de desarrollo  
(*Obtención de versiones funcionales: 5; No obtención de versiones funcionales: 0*)
- e) Software crítico  
(*Codificación no crítica: 5; Codificación crítica: 0*)
- f) Costos de Desarrollo  
(*Costos de desarrollo bajos: 5; Costos de desarrollo altos: 0*)

- g) Duración del plazo de entrega del sistema final  
(Duración de entrega Alta: 0; Duración de entrega Baja: 5)
- h) Complejidad del Sistema  
(Complejidad baja: 5; Complejidad alta: 0)
- i) Comunicación entre clientes y desarrolladores  
(Buena comunicación: 5; Mala comunicación: 0)
- j) Tamaño del equipo de desarrollo  
(Equipo de desarrollo pequeño: 5; Equipo de desarrollo grande: 0)
- k) Documentación detallada del proceso de análisis y diseño  
(Buena documentación: 5; Poca documentación: 0)
- l) Información ilustrativa sobre la aplicación de la metodología  
(Buena Información: 5; Poca Información: 0)

Criterio	Metodología de Desarrollo Ágil		Metodología de Desarrollo Tradicional	
	Extreme Programming	SCRUM	RUP	Craig Larman
a	2	2.5	5	5
b	1	1	5	3.5
c	5	5	2.5	3.5
d	5	5	3.5	5
e	2	2.5	4.5	4
f	5	5	1	3
g	4	4.5	2	3.5
h	5	5	1	3.5
i	4	4	2.5	4
j	4	4	2.5	3.5
k	3	3	4.5	5
l	4	5	5	5
<b>Total</b>	<b>44</b>	<b>46.5</b>	<b>39</b>	<b>48.5</b>

Tabla 3. Tabla de selección de metodologías. Modificada de [20]

### 2.3 APLICACIÓN DE LA METODOLOGÍA SELECCIONADA

De acuerdo al resultado de la aplicación de criterios de selección realizada anteriormente se decide escoger como metodología a seguir la propuesta por Craig Larman en su libro “UML y Patrones, Introducción al análisis y diseño orientado a objetos”, después de un estudio detallado se adaptó esta metodología de acuerdo a los requerimientos y características del proyecto, dando como resultado el procedimiento metodológico mostrado en la Figura 6, que consta de 3 etapas de macro nivel, la etapa de planeación, de construcción y de aplicación, donde cada una de dichas etapas está compuesta de los artefactos de modelado

mostrados. En la primera etapa o de planeación, se encuentra la definición de requerimientos que representan un punto decisivo para conocer con claridad las necesidades del proyecto con un alto grado de detalle, de tal forma que se detecten desde el principio, puntos críticos en el proceso de desarrollo, además se desarrolla la definición de los casos de uso de alto nivel, los casos esenciales de uso, los diagramas de caso de uso, la arquitectura del sistema, el modelo conceptual y el esquema de la base de datos. En la etapa de construcción se desarrollan los ciclos iterativos de desarrollo cuya clasificación se hace a partir de los criterios de clasificación definidos por el cliente y el equipo de diseño, que permiten realizar una adaptación progresiva de la funcionalidad del prototipo en cada uno de estos. Cada ciclo de desarrollo consta de análisis, diseño, construcción y prueba. Dentro del análisis se realizan los diagramas de secuencia y los contratos de operaciones, en el diseño se realizan los diagramas de interacción y los de diseño de clases. A partir del segundo ciclo de desarrollo se realiza una etapa de sincronización de artefactos para finalmente obtener la aplicación final o el prototipo. En la última etapa, aplicación, se realiza la transición hacia la implementación y su posterior uso.

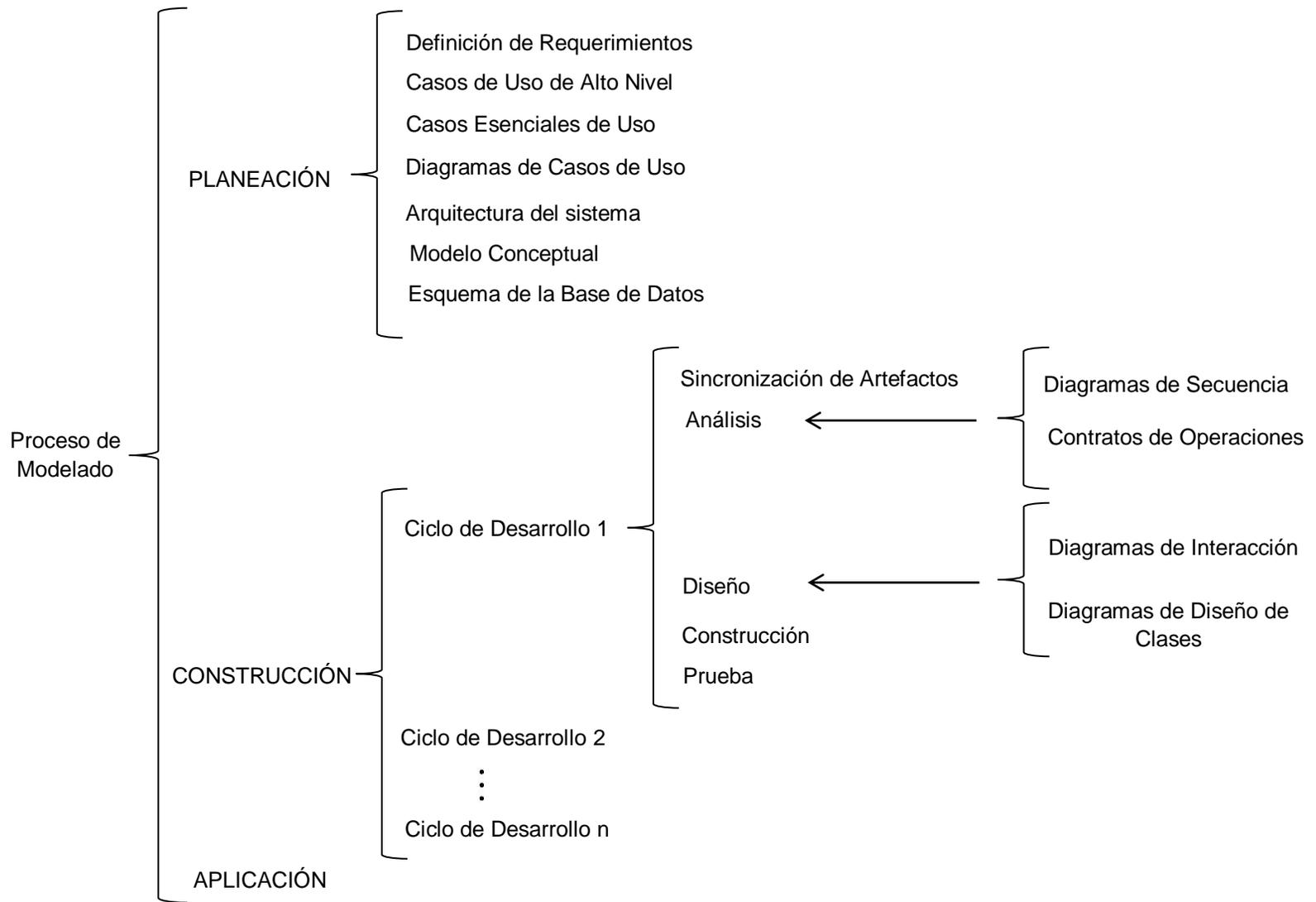


Figura 6. Metodología UML implementada

A continuación se presenta el desarrollo del modelado UML<sup>7</sup> [27] del prototipo de gestión de campañas Batch.

## 2.3.1 PLANEACIÓN

### 2.3.1.1 Definición de Requerimientos

#### 2.3.1.1.1 Panorama General

Este proyecto tiene como objetivo la generación de un prototipo de sistema software que permita generar campañas batch utilizando operaciones tipo Split and Merge en los programas de producción, para un segmento de proceso, con múltiples productos, múltiples equipos y diversas capacidades para estos.

#### 2.3.1.1.2 Clientes

La empresa integradora en servicios de Automatización y Control OMNICON S.A.

#### 2.3.1.1.3 Metas

De manera global la meta es mejorar el proceso de programación y despacho de la producción de un segmento de proceso batch, proporcionando una reducción de costos en pérdidas tanto de tiempos innecesarios como del mal aprovechamiento de la capacidad máxima disponible de los equipos, además de las reconfiguraciones de estos debido a cambios en el producto, entre otras cosas.

#### 2.3.1.1.4 Funciones del sistema

Para la definición de las funciones del sistema se hace uso del análisis de requerimientos funcionales obtenidos de la interacción con el cliente donde se concluye que los requerimientos funcionales principales del sistema son:

Número de Referencia	Requerimiento	Categorías
R1.1	El sistema deberá soportar el intercambio de información estandarizado para la recepción del programa de producción editado en xml bajo el formato <i>B2MML</i> para el acceso a requerimientos de producción.	Evidente
	El sistema generará las listas de órdenes de trabajo que se asimila al <i>BatchList</i> y la	Ocultas

<sup>7</sup> El UML o lenguaje unificado de modelado define "un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan" [27]

<b>R1.2</b>	solución deberá proveer las capacidades de despacho (integración) al sistema FT batch.	
<b>R1.3</b>	La solución deberá contemplar el diseño e implementación de software bajo las mejores prácticas que realice las funcionalidades de programación de la producción y posibilite integrar órdenes de trabajo y la consolidación de información de ejecución de producción desde y hacia el FT batch.	Oculto
<b>R1.4</b>	El sistema deberá implementar y garantizar la integridad transaccional y de datos.	Oculto
<b>R1.5</b>	Almacenar información relevante del proceso	Oculto
<b>R1.6</b>	Permitir configurar la información del proceso	Evidente
<b>R1.7</b>	Realizar acciones tipo Split & Merge.	Oculto
<b>R1.8</b>	La solución deberá ser desarrollada con las tecnologías de Microsoft, Visual Studio.Net, ASP.Net y SQL Server.	Oculto

**Tabla 4. Requerimientos del Cliente (Omnicon)**

A partir de uno de los objetivos planteados en el trabajo, se definió que el prototipo debería modelarse tomando como referencia al estándar ISA-95.03 por lo cual para este apartado se toma en cuenta el estudio desarrollado en el capítulo 1 del cual se concluye la definición de las funcionalidades del Omnicampaign para las tareas de programación detallada de la producción y despacho de la producción mostradas en la Tabla 5 y Tabla 6.

<b>Número de Referencia</b>	<b>Función</b>	<b>Categorías</b>
<b>R2.1</b>	Crear y mantener un programa detallado de la producción.	Evidente
<b>R2.2</b>	Determinar la capacidad comprometida de cada recurso para el uso de la función de administración de los recursos de producción.	Evidente
<b>R2.3</b>	Recibir el Programa de Producción proveniente del nivel 4.	Evidente

**Tabla 5. Definición de Requerimientos de Programación Detallada de la Producción según el Estándar ISA-95.03**

<b>Número de Referencia</b>	<b>Función</b>	<b>Categorías</b>
<b>R3.1</b>	Emitir órdenes de trabajo de producción como fueron definidas por el programa.	Evidente
<b>R3.2</b>	Liberar recursos locales para comenzar órdenes de trabajo.	Ocultas
<b>R3.3</b>	Mantener el estado de las órdenes de trabajo. Ejemplo: aprobado, fijo, en proceso o cancelado.	Evidente
<b>R3.4</b>	Asegurar que las limitaciones del proceso y las órdenes bajo el nivel de detalle de la programación detallada son logradas en producción. Esto toma lugar después de que el programa es creado, pero antes de que sus elementos son ejecutados.	Ocultas
<b>R3.5</b>	Informar a la programación detallada cuando un evento no anticipado resulta en la inhabilidad de lograr los requerimientos de programación.	Evidente
<b>R3.6</b>	Enviar o poner a disposición la lista de despacho de producción especificando las actividades productivas que se realizan.	Evidente

**Tabla 6. Definición de Requerimientos de Despacho de la Producción según el Estándar ISA-95.03**

El resultado del análisis de requerimientos y de funcionalidades del prototipo se resume en las 3 tablas anteriores, las cuales contienen los requerimientos particulares del cliente, los requerimientos de la programación detallada y los requerimientos del despacho de la producción según el estándar ISA-95.03 respectivamente.

#### **2.3.1.1.5 Atributos del sistema**

En esta parte se definen una serie de atributos del sistema, los cuales nos permiten establecer de manera global las características básicas de éste, más no sus funcionalidades. Dichos atributos, pueden ser los detalles o las restricciones de frontera. Los detalles permiten darle especificaciones a los atributos del sistema, como una característica particular y las restricciones de frontera denotan una serie de términos que se deben cumplir.

<b>Atributo</b>	<b>Detalles y Restricciones de Frontera</b>
<b>Facilidad de Uso</b>	(Detalle) Las ventanas del prototipo deben facilitar el uso del sistema.
<b>Tolerancia a Fallos</b>	(Restricción de Frontera) El prototipo debe mantener la integridad de la información a pesar de fallos hardware o

	software, volviendo al último estado estable de la información.
<b>Tipo de Aplicación</b>	(Restricción de Frontera) El prototipo debe ser una Aplicación Web.

**Tabla 7. Definición de Atributos del Sistema**

### 2.3.1.1.6 Atributos del Sistema en las Especificaciones de Funciones

Para dar una mayor especificación, en la Tabla 8 a continuación se realiza una asociación de los atributos con algunos de los requerimientos funcionales anteriormente definidos en la Tabla 4, Tabla 5 y Tabla 6.

Número de Referencia	Función	Categoría	Atributo	Detalles y Restricciones	Categoría
R1.5	Almacenar información relevante resultado de la programación detallada de la producción.	Oculto	Tolerancia a Fallos	Recuperar el último estado estable de la información en	Obligatorio
R2.1	Crear y mantener un programa detallado de la producción.	Evidente	Tipo de Aplicación	Aplicación Web	Obligatorio
			Facilidad de Uso	Ventanas de fácil entendimiento	Obligatorio
				Colores suaves e iconos intuitivos	Opcional
R1.6	Permitir configurar la información del proceso	Evidente	Tipo de Aplicación	Aplicación Web	Obligatorio
			Facilidad de Uso	Ventanas de fácil entendimiento	Obligatorio
				Colores suaves e iconos intuitivos	Opcional

**Tabla 8. Relación de los Atributos del Sistema con las Funciones definidas**

### 2.3.1.1.7 Definición de glosario

La definición del glosario permitirá al desarrollador tener un documento claro donde se especifican los conceptos importantes generados en los artefactos, a medida que se avanza por los ciclos de desarrollo en la etapa de sincronización de artefactos este se irá actualizando siendo cada vez más completo y brindando más información.

<b>Término</b>	<b>Categoría</b>	<b>Comentario</b>
Campaña	Tipo	Es una agrupación de Batches que se ejecuta como una sola tanda.
Generar Campañas Batch	Caso de Uso	Obtener una reorganización de órdenes de producción, mediante la aplicación de acciones Split y Merge proporcionando una reducción de costos en pérdidas producidos por tiempos innecesarios, mal aprovechamiento de la capacidad máxima disponible de los equipos, además de las reconfiguraciones de estos debido a cambios en el producto, entre otras cosas.
Programa de Producción	Tipo	Se define como un conjunto de requerimientos de producción que fijan las necesidades de la empresa.
Programa Detallado de Producción	Tipo	Es definido como una colección de órdenes de trabajo de producción que involucran la producción de 1 o más productos con el nivel de detalle requerido por la manufactura.
Propiedades Equipo	Tipo	Describen las propiedades del equipo con su respectivo ID, Descripción, Valor y Unidad.
Propiedades Producto	Tipo	Describen las propiedades del Producto con su respectivo ID, Descripción, Valor y Unidad.
Campañas Batch.CA_Estado	Atributo	Estado de la campaña Batch en un tiempo t, los estados definidos que puede tomar son (Aborted – Running – Paused - Held) [19] - [28]
Despacho de la Producción	Caso de Uso	Enviar hacia el administrador batch el programa detallado aprobado (Lista de Despacho). Recibir del administrador Batch información de la capacidad y la disponibilidad de los equipos y el estado de las campañas.

Tabla 9. Definición de Términos

## **2.3.1.2 Casos de Uso de Alto Nivel**

### **2.3.1.2.1 Frontera del Sistema**

La frontera del sistema define el alcance del sistema y es un elemento útil para determinar las responsabilidades de éste y las interacciones con entes externos. Para el sistema prototipo de gestión de campañas batch la frontera se especifica como el Hardware y Software empleado para su desarrollo.

### **2.3.1.2.2 Identificación de Actores**

Los actores participantes en el contexto del sistema son:

- Supervisor (Humano)
- Sistema Administrador Batch (Sistema Software)

### **2.3.1.2.3 Identificación y Descripción de los Casos de Uso de alto nivel**

En la fase de planeación y elaboración se identifican los casos de uso de alto nivel y esenciales, los cuales se caracterizan por tener un nivel de detalle poco profundo, la idea es poder describir el proceso de una forma general sin tener en cuenta especificaciones tecnológicas ni de diseño, las cuales se identifican más adelante en el proceso de construcción, los casos de uso considerados en el análisis del prototipo son:

- Inicio de Sesión
- Configuración Inicial
- Recibir Programa de Producción.
- Generar Campañas Batch.
- Generar Manualmente Campañas Batch.
- Despachar Campañas Batch

El desarrollo de los casos de uso de alto nivel definidos anteriormente se consigna desde la Tabla 10 a la Tabla 15.

<b>Caso de Uso</b>	Inicio de Sesión
<b>Actores</b>	Supervisor
<b>Tipo</b>	Primario
<b>Descripción</b>	El usuario realiza un inicio de sesión en el sistema, con el fin de que solo los supervisores configurados en el sistema sean los que acceden a las funcionalidades de este.

**Tabla 10. Caso de uso de alto nivel Inicio de Sesión**

<b>Caso de Uso</b>	Configuración Inicial
<b>Actores</b>	Supervisor
<b>Tipo</b>	Primario
<b>Descripción</b>	El usuario se encarga de configurar la información inicial necesaria para el correcto funcionamiento del sistema, como los productos que la empresa fabrica, los equipos presentes en esta y los segmentos de proceso que se manejan en la planta.

**Tabla 11. Caso de Uso de Alto Nivel de Configuración Inicial**

<b>Caso de Uso</b>	Recibir Programa de Producción
<b>Actores</b>	Supervisor
<b>Tipo</b>	Primario
<b>Descripción</b>	El supervisor busca y selecciona el programa de producción enviado desde el sistema ERP el cual es interpretado por el sistema

**Tabla 12. Caso de Uso de Alto Nivel de Recibir Programa de Producción**

<b>Caso de Uso</b>	Generar Campañas Batch
<b>Actores</b>	Supervisor
<b>Tipo</b>	Primario
<b>Descripción</b>	El sistema de la mano del Supervisor propone un programa detallado basado en campañas batch, el Supervisor decide si el resultado final es aceptado.

**Tabla 13. Caso de Uso de Alto Nivel de Generar Campañas Batch**

<b>Caso de Uso</b>	Generar Manualmente Campañas Batch
<b>Actores</b>	Supervisor
<b>Tipo</b>	Primario
<b>Descripción</b>	El supervisor se encarga de generar de forma manual y según su criterio las campañas que no fueron aprobadas.

**Tabla 14. Caso de Uso de Alto Nivel de Generar Campañas Batch con asistencia software**

<b>Caso De Uso</b>	Despachar Campañas Batch
<b>Actores</b>	Supervisor
<b>Tipo</b>	Primario
<b>Descripción</b>	El Supervisor envía hacia el administrador batch el programa detallado aprobado y este lo recibe y lo ejecuta como fue definido. Además, el administrador Batch se encarga de enviar información referente al estado de los equipos en el proceso y la ejecución de las campañas.

**Tabla 15. Caso de Uso de Alto Nivel de Despachar Campañas Batch**

#### **2.3.1.2.4 Clasificación de los Casos de Uso**

La clasificación de los casos de uso es una tarea de gran importancia, ya que permite identificar el orden adecuado en el cual se deben atacar dichos casos a lo largo de los ciclos de desarrollo que se definan, con el fin de abordar primero los de alto rango que influyen fuertemente en el correcto desarrollo del sistema. Para efectuar dicha clasificación se definen una serie de criterios basados en [5] a los que son sometidos los diferentes casos de uso identificados, proporcionándoles una ponderación numérica de 0-5 y que de acuerdo a su calificación global se asignan entonces a los ciclos de desarrollo, el resultado de dicha aplicación se evidencia en la Tabla 16.

### 2.3.1.2.5 Criterios de Clasificación

Los siguientes criterios de clasificación se ponderan en un rango de 0 - 5.

- a) Requerir una investigación a fondo con tecnología desconocida o riesgosa.
- b) Representar procesos primarios del funcionamiento del sistema.
- c) Repercusión fuerte en el diseño arquitectónico.
- d) Dependencia de información para otros casos de uso.
- e) Grado de interacción con el usuario

<b>Casos de Uso</b>	<b>a</b>	<b>b</b>	<b>c</b>	<b>d</b>	<b>e</b>	<b>Suma</b>
<b>Inicio de Sesión</b>	0	2	1	2	5	10
<b>Configuración inicial</b>	2	5	0	5	5	17
<b>Recibir el Programa de Producción</b>	5	4	4	4.5	2	19.5
<b>Generar Campañas Batch</b>	4	4.5	2.5	3.5	2	16.5
<b>Generar Manualmente Campañas Batch</b>	2	1	0	3	5	11
<b>Despachar Campañas Batch</b>	4	5	4	0	2	15

Tabla 16. Clasificación de los Casos de Uso

### 2.3.1.2.6 Asignación de los Casos de Uso a los ciclos de desarrollo

Dentro de la metodología seguida se propone un desarrollo iterativo el cual se divide en varios ciclos de desarrollo que una vez completados van agregando valor a las funcionalidades del prototipo software.

Para el caso del prototipo de sistema de gestión de campañas batch fueron definidos 3 ciclos de desarrollo de aproximadamente 6 semanas de duración durante los cuales se desarrollaron los casos de uso o versiones simplificadas respectivos a estos, como se muestra a continuación en la Figura 7.

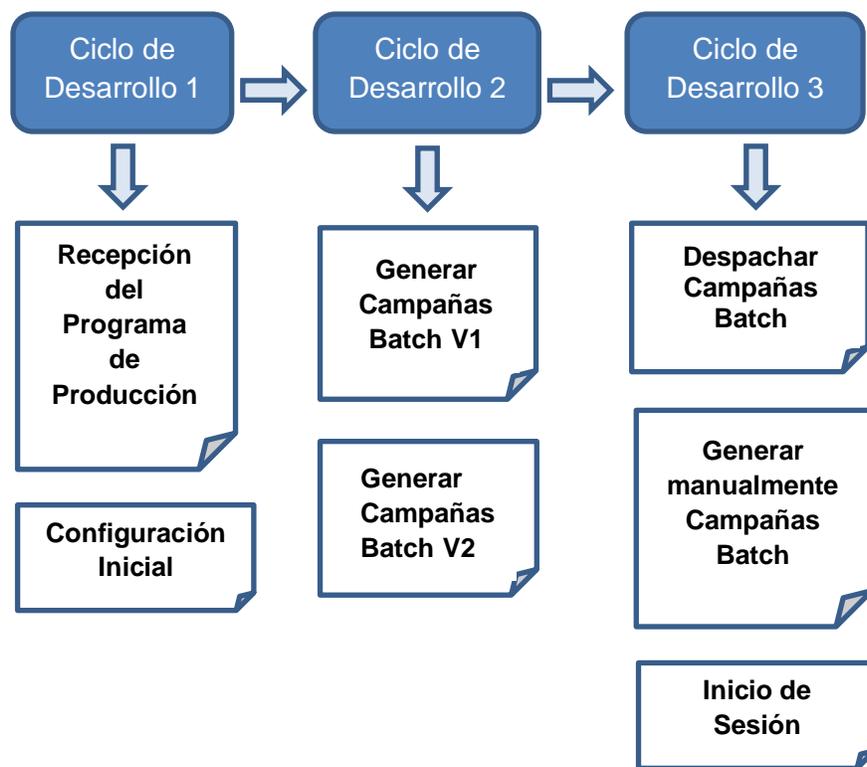


Figura 7. Distribución de los Casos de Uso en los ciclos de desarrollo

A partir de este punto solamente se presentan los artefactos de modelado resultantes de la aplicación de la metodología para el caso de uso de generación de campañas batch versión 1 (Split). El desarrollo de los demás casos de uso se presentará en el Anexo A.

### 2.3.1.3 Definición de los Casos Esenciales de Uso

En esta etapa se pretende detallar un poco más los casos de uso de alto nivel sin llegar a una especificación profunda con detalles de tecnología ni de implementación. Los casos esenciales de uso describen de forma general lo que ocurre en la realidad mostrando las posibles rutas de eventos que pueden suceder en el funcionamiento básico del sistema. A continuación se muestra el caso de uso esencial para la versión 1 de Generar campañas Batch, el resto de los casos esenciales de uso se muestran en el anexo A.

**2.3.1.3.1 Caso esencial de uso para el caso de uso Generar Campañas Batch versión 1 (Split).**

<b>CASO DE USO</b>	<b>GENERAR CAMPAÑAS BATCH V1 (Split)</b>	
<b>ACTORES</b>	Supervisor (Iniciador)	
<b>PROPOSITO</b>	Obtener una reorganización de órdenes de producción, mediante la aplicación de acciones tipo Split evitando el mal aprovechamiento de la capacidad máxima disponible de los equipos, además de maximizar su utilización.	
<b>RESUMEN</b>	El sistema de la mano del Supervisor propone un programa detallado basado en campañas batch, el Supervisor decide si el resultado final es aceptado.	
<b>TIPO</b>	Primario, Esencial	
<b>REFERENCIAS CRUZADAS</b>	R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.2	
<b>PRECONDICION</b>	Recibir e Interpretar el programa de producción.	
<b>CURSO NORMAL</b>	<b>ACTOR</b>	<b>SISTEMA</b>
	1. El supervisor elige un programa de producción de la lista.	
		2. El sistema muestra la lista de requerimientos de producción asociados al programa de producción seleccionado.
	3. El supervisor se encarga de elegir un requerimiento de producción de la lista.	

		4. El sistema muestra la lista de requerimientos de segmento asociados al requerimiento de producción seleccionado.
	5. El supervisor elige el requerimiento de segmento que desea programar.	
		6. El sistema muestra los remanentes del mismo ID de producto existentes.
	7. El supervisor decide si une (Merge) el requerimiento de segmento con un remanente encontrado	
		8. El sistema muestra la lista de segmentos de proceso asociados al requerimiento de segmento seleccionado.
	9. El supervisor elige el segmento de proceso sobre el que desea programar.	
		10. El sistema realiza el proceso de análisis de la información como, capacidad de los equipos, disponibilidad, cantidades de producto solicitadas, productos a ser fabricados, entre otras y permite al Supervisor la toma de decisiones.
	11. El Supervisor realiza la toma de decisiones en las operaciones de <b>Split</b> .	

		12. Genera una propuesta de campaña batch basado en las decisiones del Supervisor.
	13. El Supervisor analiza el programa detallado propuesto y lo aprueba.	
	a. Si el supervisor desea observar la información de alguna campaña, este la selecciona del conjunto de campañas	
		b. El sistema permite mostrar la información relacionada con la campañas seleccionada
	c. El supervisor decide si la información mostrada es satisfactoria	
	<b>ACTOR</b>	<b>SISTEMA</b>
<b>CURSO ALTERNO</b>		6.1 El sistema no encuentra remanentes del mismo ID del producto asociado al requerimiento de segmentos seleccionado
	7. El usuario decide programar el requerimiento de producción sin ningún remanente	

	<b>11.1. Decisión Generar Remanente</b>	
	12. El usuario decide generar la campaña dejando un remanente de producción.	
		13. El sistema realiza el cálculo del número de Batches y genera un nuevo requerimiento con la cantidad de material faltante e incluye el remanente en la lista de requerimientos pendientes.
	<b>11.2 Decisión Realizar Sobreproducción</b>	
	12. El usuario decide generar la campaña sobrepasando la cantidad de producto requerido a fabricar y generando un Sobre-stock.	
	<b>11.3 Decisión Cumplir con el requerimiento sin aprovechar al máximo la capacidad.</b>	

	<p><b>12</b> El usuario decide cumplir el requerimiento dejando algunos Batches de la campaña a generar con un valor menor que la capacidad máxima disponible en el equipo pero mayor que la capacidad mínima.</p>	
		<p><b>13</b> El sistema calcula el nuevo tamaño del Batch necesario para cumplir el requerimiento sin dejar remanentes, ni sobre productos.</p>
	<p><b>c.1. El supervisor no está de acuerdo con la información mostrada de la campaña</b></p>	
	<p><b>d.</b> El operario decide borrar la campaña y la selecciona</p>	
		<p><b>e.</b> El sistema permite borrar la campañas seleccionada</p>

**Tabla 17. Formato extendido del Caso Esencial de Uso  
Generar Campañas Batch v1**

### 2.3.1.4 Definición de Diagrama de Caso de Uso

Los diagramas de casos de uso permiten obtener una vista general del sistema con sus funcionalidades principales, las relaciones existentes entre estas y la interacción que presentan con los actores del sistema. El diagrama de casos de uso desarrollado para el sistema prototipo se muestra en la Figura 8 a continuación.

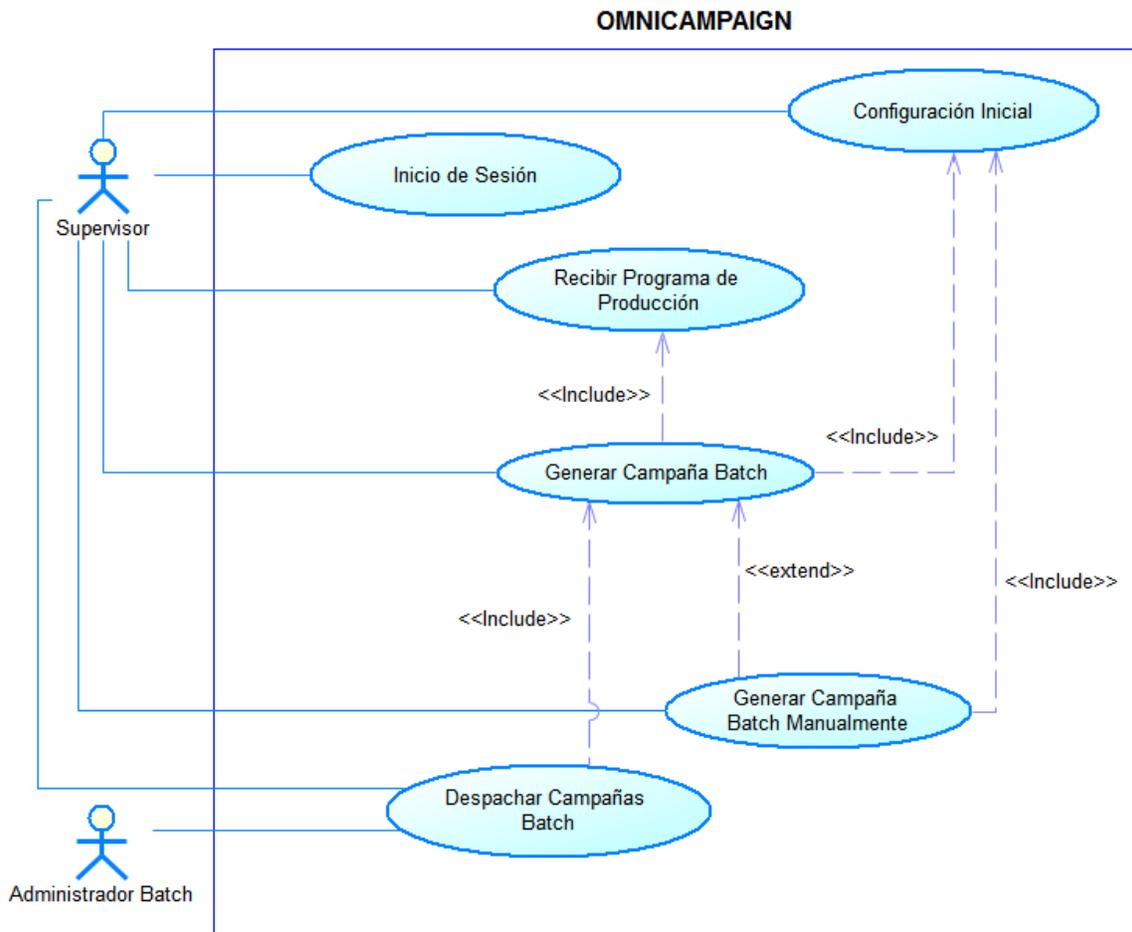


Figura 8. Diagrama de Casos de Uso

### 2.3.1.5 Arquitectura del sistema

#### 2.3.1.5.1 Arquitectura de integración del prototipo en la empresa

Como se evidenció en la especificación de requerimientos y análisis funcional del prototipo a partir del estándar ISA-95.03 y además teniendo en cuenta que las actividades que desarrolla esté son la programación detallada y el despacho de la producción, tareas pertenecientes al nivel de ejecución de la manufactura (Nivel 3) se plantea la arquitectura mostrada en la Figura 9.

Donde se evidencia que el prototipo se ubica en el nivel 3 siendo consecuente con las tareas que desempeña y desde este nivel se encarga de la recepción del programa de producción proveniente del sistema ERP (Nivel 4) editado bajo el esquema de integración B2MML, que usa como partida para la generación de la programación detallada por medio de campañas batch y posteriormente permitiendo la integración con el sistema administrador batch (Factory Talk Batch) para el envío de las listas de despacho editadas bajo el esquema de integración BATCHHTML.

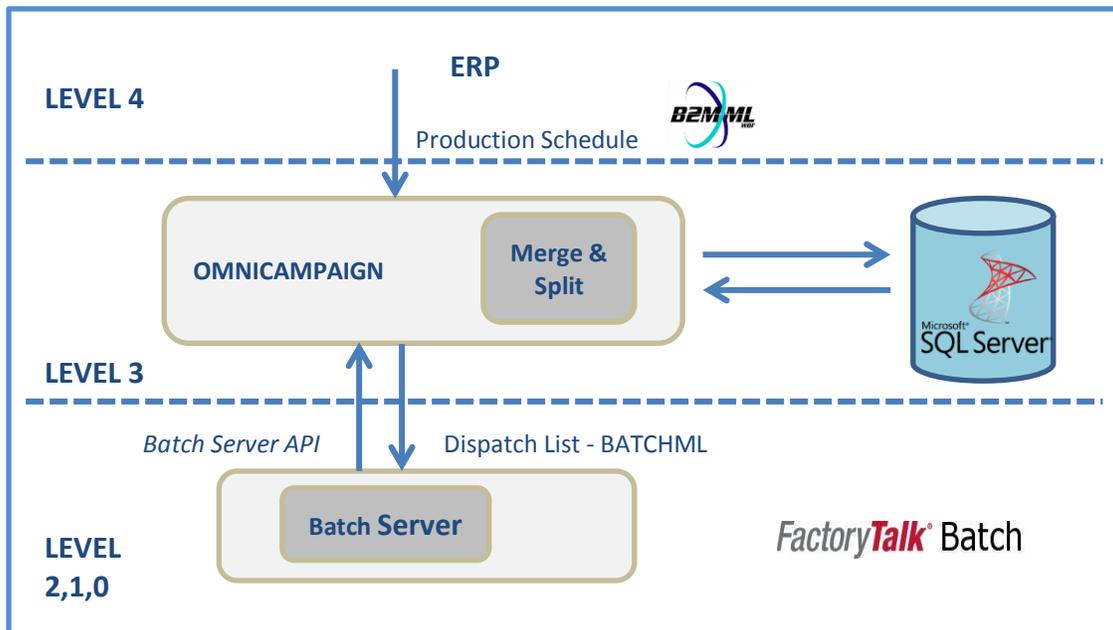
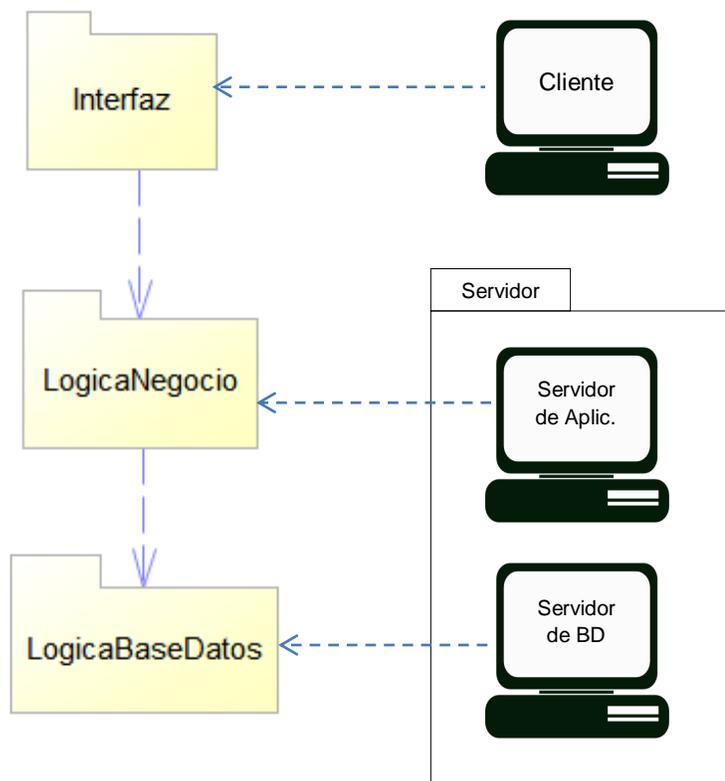


Figura 9. Arquitectura general de la empresa

### 2.3.1.5.2 Arquitectura definida de la aplicación web

Para el desarrollo del sistema prototipo de gestión de campañas batch, se emplea la arquitectura en 3 capas (ver Figura 10) tomando como referencia el análisis comparativo con otras arquitecturas y las ventajas y desventajas plasmadas en [29]. La arquitectura de 3 capas busca que cada capa trabaje en un área específica como lo son la interfaz con el usuario, la lógica de la aplicación y la forma de almacenamiento de datos, brindando flexibilidad y robustez que a su vez facilitan la escalabilidad y mantenibilidad del sistema sin afectar ninguno de los módulos.



**Figura 10. Arquitectura Interna del prototipo y correspondencia a componentes físicos adaptada de [29]**

La capa de la interfaz se encarga de mostrar los datos generados por la aplicación y de capturar todos los eventos proporcionados al sistema resultado de la interacción con el usuario final.

La capa de lógica de negocios es la que coordina toda la aplicación, las decisiones, evaluaciones, validaciones, los comandos y el desarrollo de cálculos. Además se encarga del movimiento y procesamiento de datos entre las otras dos capas [30].

La última capa se encarga del almacenamiento y acceso de datos, lo cual se logra mediante la creación de un soporte para el servidor de datos usado, lo que permite de manera flexible modificar dicho servidor sin tener inconvenientes en la lógica de negocio.

Es importante tener en cuenta que al aplicar la arquitectura de tres capas la separación entre estas es de manera lógica, por lo cual *“es posible ejecutar las tres capas en una misma máquina”* [29].

Al mantener la separación de la capa de presentación y la capa de negocios se facilita la flexibilidad en la ejecución de cambios de la interfaz, ya que al ser independientes, los cambios realizados en esta no afectan directamente la lógica de negocio y además permite una mayor flexibilidad en el llamado de la capa intermedia desde la capa de presentación. Adicionalmente se puede presentar mayor facilidad para la implementación y la inclusión de nuevos clientes del sistema en otros dispositivos que se deseen vincular al proyecto [29]

Siendo coherente con la arquitectura seleccionada y las ventajas deseadas [31], dentro de los patrones de arquitectura que permitirán dar la base o cuerpo del desarrollo software del prototipo será utilizado el patrón MVC (*Modelo Vista Controlador*). El modelo contiene la lógica a efectuarse según los eventos recibidos por el sistema, teniendo por esta razón una relación directa con el nivel de lógica de negocio; La vista se encarga de controlar la presentación de la información, de mostrar las páginas web necesarias para la interacción con el usuario, relacionándola con la capa de presentación; el controlador es el responsable de descifrar los eventos del sistema ocasionados por el usuario y de informarle éstos al modelo, por lo cual se encuentra ubicado entre la capa de interfaz y la lógica de negocio [32].

Otro aspecto importante a considerar para obtener un desarrollo de la aplicación bajo buenas prácticas son los patrones de diseño, los cuales son una herramienta general resultado de un conjunto de experiencias en busca de buenas prácticas y diseños para la solución de problemas comunes en el análisis y desarrollo de sistemas software en un contexto

particular. Dentro de los patrones que se encargan del desarrollo se encuentran los patrones GOF (*Gang of Four*) y los que realizan el análisis del sistema permitiendo la asignación de responsabilidades según las necesidades del proyecto son los patrones GRASP (*General Responsibility Assignment Software Patterns*).

Los patrones GRASP son utilizados como guía para la asignación de obligaciones o responsabilidades en la interacción de los objetos y las clases, de esta manera representan una parte fundamental sobre cómo se diseñara el sistema. El uso de patrones no es obligatorio y si se deciden usar no es necesario aplicarlos todos, el empleo de estos depende más de las necesidades del software. [5]

Teniendo en cuenta que los 5 patrones GRASP fundamentales en el diseño son el patrón experto, creador, de alta cohesión, bajo acoplamiento y controlador como se evidencia en [5], estos fueron utilizados en el desarrollo del proyecto. El experto se encarga de asignar la responsabilidad a la clase(s) que cuente con la información necesaria requerida para cumplir con la labor, es decir el experto en información. El creador es el patrón responsable de determinar que clase debe instanciar un determinado objeto según las necesidades. El bajo acoplamiento es el patrón que se encarga de mantener una dependencia moderada entre las clases, evitando los múltiples cambios que ocasionaría uno simple y dando independencia en el diseño de clases. El patrón de alta cohesión es el responsable de mantener una relación de afinidad entre las labores de una clase así como el número de estas. Por último el patrón controlador, es el encargado del manejo de los eventos del sistema y de pasar los parámetros a la lógica de negocio redireccionando los mensajes a las respectivas entidades encargadas de ejecutar las tareas.

Los controladores no hacen parte del modelo, es una clase que se fabrica con el fin de dar apoyo al sistema, estos se catalogan en: controladores de fachada, de tareas y de caso de uso [5]: El controlador de fachada es el que identifica el sistema en su totalidad, este controlador es adecuado cuando el sistema no presenta gran cantidad de eventos; el controlador de tareas es el que se designa para representar los deberes o tareas de una persona dentro del dominio del problema; por último el controlador de casos de uso, en el que se designa uno por cada caso de uso, se emplea cuando el controlador de fachada lleva a un diseño en donde una clase depende de varias, dando como consecuencia un alto acoplamiento o cuando las clases tienen en su poder muchas responsabilidades y estas no

se relacionan, es decir, existe una baja cohesión. Al considerar que el prototipo no presenta un manejo de eventos considerable, es decir que no es lo suficientemente grande para presentar saturación o baja cohesión, se eligió el controlador de fachada para la implementación del sistema

En la clasificación de los patrones para el desarrollo *GoF*, existen tres categorizaciones para los patrones según el propósito: los patrones de creación, los estructurales y los de comportamiento; los patrones creacionales son los utilizados para la creación y configuración de los objetos sin intervención del sistema, dentro de estos los más comunes son: fábrica abstracta, método de fabricación, constructor, prototipo y Singleton; los patrones estructurales separan la interfaz de usuario de la aplicación, determina como se agrupan los objetos y clases para construir estructuras más complejas, los más comunes son: Adaptador, puente, Objeto compuesto, Envoltorio (Decorador), Fachada, Peso ligero y apoderado (Proxy); finalmente los patrones de comportamiento se encargan de describir como se relacionan las clases y objetos, es decir, como cooperan entre ellos y se asignan responsabilidades en busca de lograr los objetivos, algunos de estos son: Cadena de responsabilidad, Intérprete, Iterador, Mediador, comando, Recuerdo, Observador, Estado, Estrategia, Método plantilla y Visitante [33] [34].

#### **2.3.1.6 Definición del Modelo Conceptual**

El modelo conceptual es una de las etapas más importantes del proceso de análisis, ya que permite obtener una representación de los conceptos del mundo real aplicados en el dominio del problema [5]. La idea es abstraer la mayor cantidad de información del problema mostrando los conceptos asociados a este, los atributos y las asociaciones que estos forman.

Se ponen en práctica dos tipos de técnicas en la identificación de conceptos planteadas en la metodología a continuación. La primera consiste en identificar los conceptos según una lista definida que contiene aspectos claves en el desarrollo de sistemas y la segunda se basa en los casos de uso planteados, la aplicación de las técnicas se muestran a continuación para el caso de uso Generar Campañas Batch.

### 2.3.1.6.1 Identificación de Conceptos para la Versión 1 Simplificada del Caso de uso “Generar campañas Batch” (Split)

#### Simplificaciones

- Se generan campañas teniendo en cuenta únicamente la funcionalidad de Split para múltiples equipos.

#### Suposiciones

- Cantidad de material a producir de cada tipo de producto.
- Capacidad de los equipos disponibles.
- Cálculos del tiempo requerido según cada requerimiento de producción.
- La disponibilidad de los equipos según la última campaña generada.

#### Meta

- Obtener la funcionalidad de Split con el análisis de capacidad y disponibilidad de equipos y tipo de productos sobre los cuales se ejecutarán las campañas generadas.

#### 2.3.1.6.1.1 Lista de Categorías de Conceptos para el Caso de Uso Simplificado: Generar Campañas Batch v1 (Split)

Categorías de Conceptos u Objetos	Conceptos u Objetos Identificados
Objetos Físicos o Tangibles	Equipo Producto
Especificaciones, diseño o descripciones de cosas	Modelo de Equipos Propiedades de Producto Récipe Matriz de tiempo Propiedades de Equipos
Lugares	Fabrica
Transacciones	Generación y aprobación de la campaña Batch
Papeles de Personas	Supervisor
Contenedores de otras cosas	Equipo
Cosas dentro de un contenedor	Materias Primas Producto
Sistemas de cómputo o Electromecánicos externos al Sistema	Administrador Batch ERP

Conceptos de nombres abstractos	Gantt
Organizaciones	Departamento de Administración de la Producción
Eventos	Decisiones del Supervisor
Procesos	Generación de Campaña Batch y Batches
Registros de Finanzas, de Trabajo, de contratos de asuntos legales	Programa Detallado

**Tabla 18. Lista de Categorías de Conceptos para el Caso de Uso Simplificado: Generar Campañas Batch v1**

#### **2.3.1.6.1.2 Definición de objetos a partir de la identificación de frases nominales del caso de uso: Generar Campañas Batch V1**

En este punto se desea identificar los objetos del sistema por medio de la lectura del curso normal y alterno de los casos de uso esenciales identificados:

- Supervisor
- Equipo
- Producto
- Campañas Batch
- Programa detallado

#### **2.3.1.6.1.3 Objetos definitivos para el caso de uso: Generar Campañas Batch V1**

- Supervisor
- Equipo
- Producto
- Batches
- Campaña Batch

En la Figura 11 que se muestra a continuación se presenta el modelo conceptual definido para el desarrollo del prototipo después de completar éste en los tres ciclos de desarrollo definidos y de haber sido refinado en cada uno de estos

Dentro de los atributos que se definen en los objetos identificados para el desarrollo del modelo conceptual han sido considerados los

plasmados en el estándar ISA-95.02 mostrados en [35], además de los requeridos para el debido funcionamiento del prototipo.

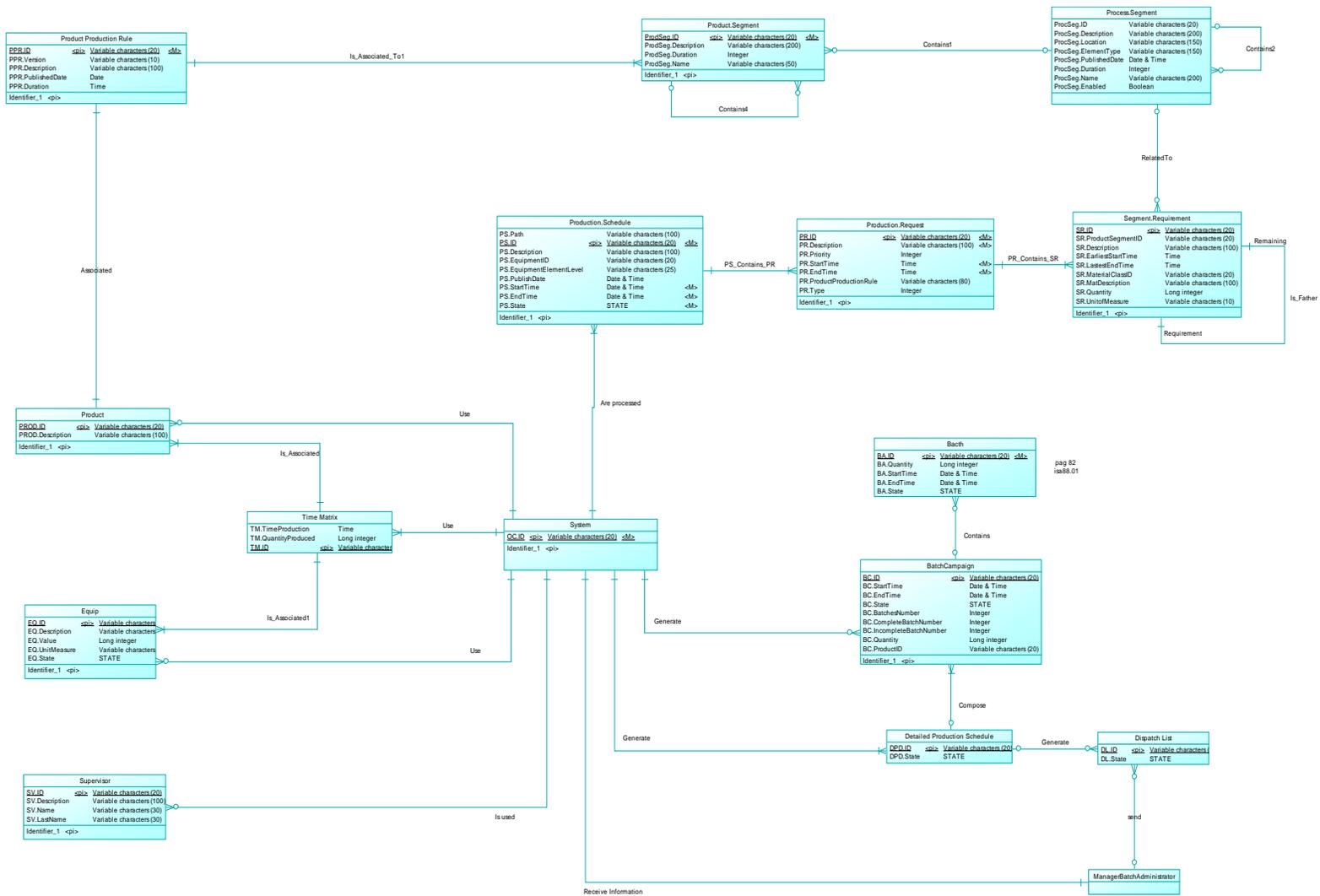


Figura 11. Modelo conceptual. Ver Anexo A

### **2.3.1.7 Definición de los Esquemas de la Base de Datos**

Con el fin de almacenar la información necesaria resultado del análisis y desarrollo de las funcionalidades del prototipo de gestión de campañas batch, se ve la necesidad de crear una base de datos que se irá completando a lo largo de la implementación de los ciclos de desarrollo. A continuación se presenta el problema inicial que se planteó como primera partida para la creación del esquema total de la base de datos utilizada. Ver Figura 12 y Figura 13

Se desea realizar el control de la programación de la producción de una empresa que ofrece múltiples productos encargada de la fabricación de pintura a base de agua en todas las gamas de colores al igual que de pintura base o pintura de color blanco. El control de la producción se realiza en una sola etapa del proceso la cual tiene disponibles varios equipos, cada uno de estos con un identificador, una descripción, un valor numérico de su capacidad y su respectiva unidad de medida. Para cada producto se considera necesario conocer un identificador y una descripción.

Semanalmente el sistema recibe un programa de producción proveniente del sistema de planeación de recursos empresariales (ERP) el cual se compone de uno o más requerimientos de producción y este a su vez de uno o más requerimientos de segmento, el cual se relaciona con un requerimiento de material producido.

De cada programa de producción se desea conocer, su ID, una descripción, localización, fecha de inicialización, fecha de finalización, fecha de publicación. De los requerimientos de producción se desea conocer su identificador, descripción, tiempo de inicio, tiempo final, prioridad y el requerimiento de segmento asociado. De los requerimientos de segmentos se desea conocer su ID, una descripción, localización, fecha de inicialización, fecha de finalización, fecha de publicación y el segmento de proceso asociado. Del requerimiento de material producido se desea conocer un identificador, descripción, cantidad de material a producir, unidad de medida y el requerimiento de segmento asociado.

El segmento de proceso es la unidad donde va a ser ejecutado un requerimiento de segmento. Cada segmento de proceso tiene asociado un segmento de producto dependiendo del elemento que se vaya a fabricar y este a su vez tiene asociada una regla de producción donde se define la forma en que debe ser fabricado cada producto.

Del segmento de proceso se desea conocer el identificador, la descripción, localización y la duración. El segmento de producto debe tener un ID, una descripción, una duración y el segmento de proceso asociado a este. La regla de producción debe tener como atributos, un identificador que es único, una versión, una descripción, una fecha de publicación y una duración.

Del proceso de programación de la producción se generan una serie de campañas Batch que permiten el aprovechamiento máximo de la capacidad disponible de los equipos y evitan la reconfiguración de los mismos, es decir al operar en modo campañas se busca dar una continuidad a la producción y así no tener que generar pérdidas de tiempo en limpiezas y reconfiguraciones por cambiar de un tipo de pintura a otro. El conjunto de campañas batch generan el programa detallado de la producción del cual se quiere conocer su identificador. De las campañas se desea conocer el ID, la hora de inicio, hora de finalización, fecha de inicio, la fecha de finalización, número total de Batches y el número total de Batches completados.

El objetivo es definir una serie de Listas de Despacho que cumplan con los requerimientos de producción, de dichas Listas de Despacho se desea conocer su Identificador, su récipe asociado, su estado (completo, en ejecución, pausado y abortado), y la cantidad de unidades a ser fabricadas.

De las operaciones de gestión de campañas Batch es posible generar remanentes de producción, los cuales se producen cuando el tamaño del requerimiento no genera un número de Batches exactos en el equipo seleccionado dejando así unas unidades de producto por programar para su posterior utilización. De los remanentes es necesario conocer el ID, una descripción, fecha de inicialización, fecha de finalización, fecha de publicación y además el identificador del requerimiento de segmento del cual surgió ese remanente.

Se requiere también conocer la Matriz de tiempos donde se registran los tiempos que tarda determinado equipo en producir determinadas cantidades de productos para calcular el Tiempo Requerido en la ejecución de una campaña y compararlo después con el Tiempo Disponible.

Por último se desea mantener un registro de los usuarios que pueden acceder al sistema, para los cuales se desea conocer su identificador único que servirá como login al sistema, contraseña, nombre, apellido.

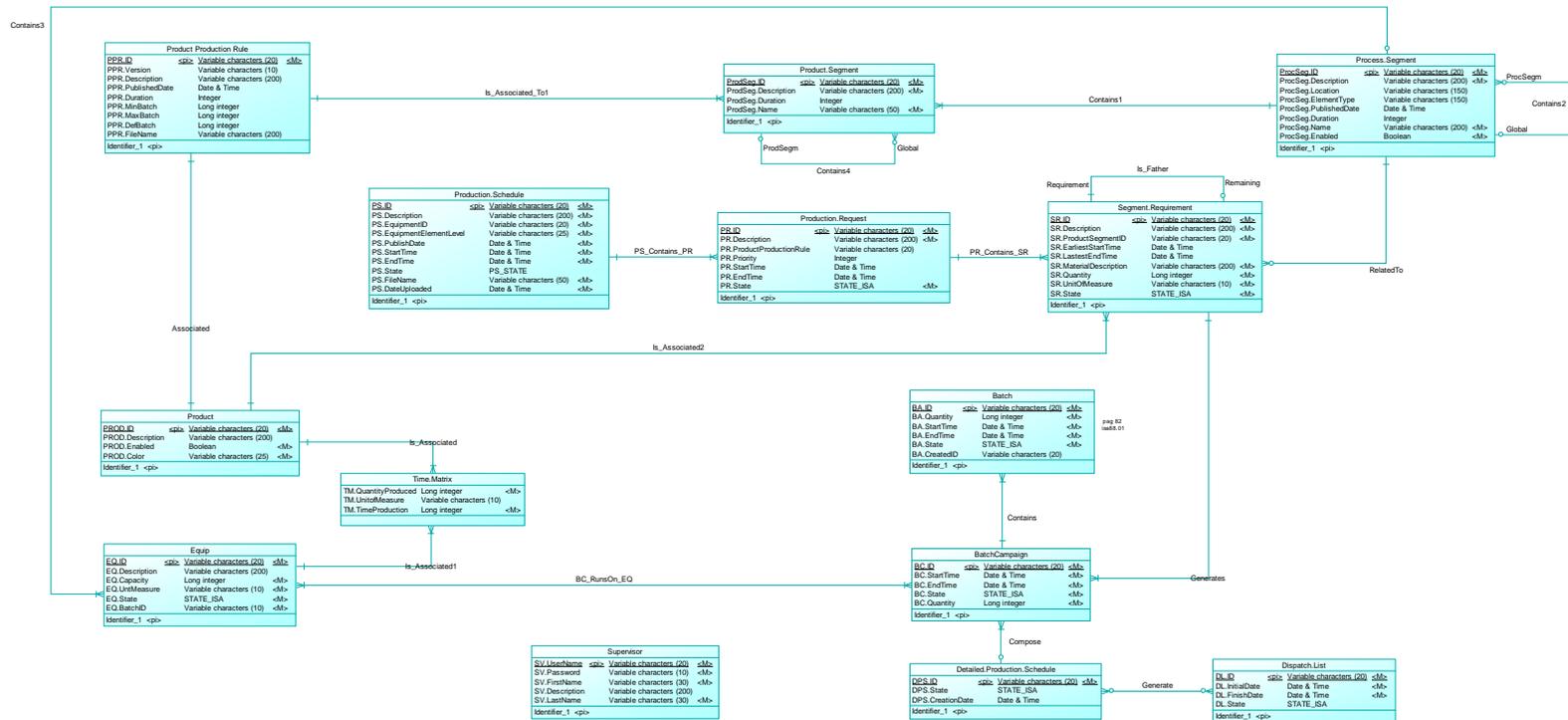


Figura 12. Modelo Entidad - Relación de la Base de Datos Ver Anexo A

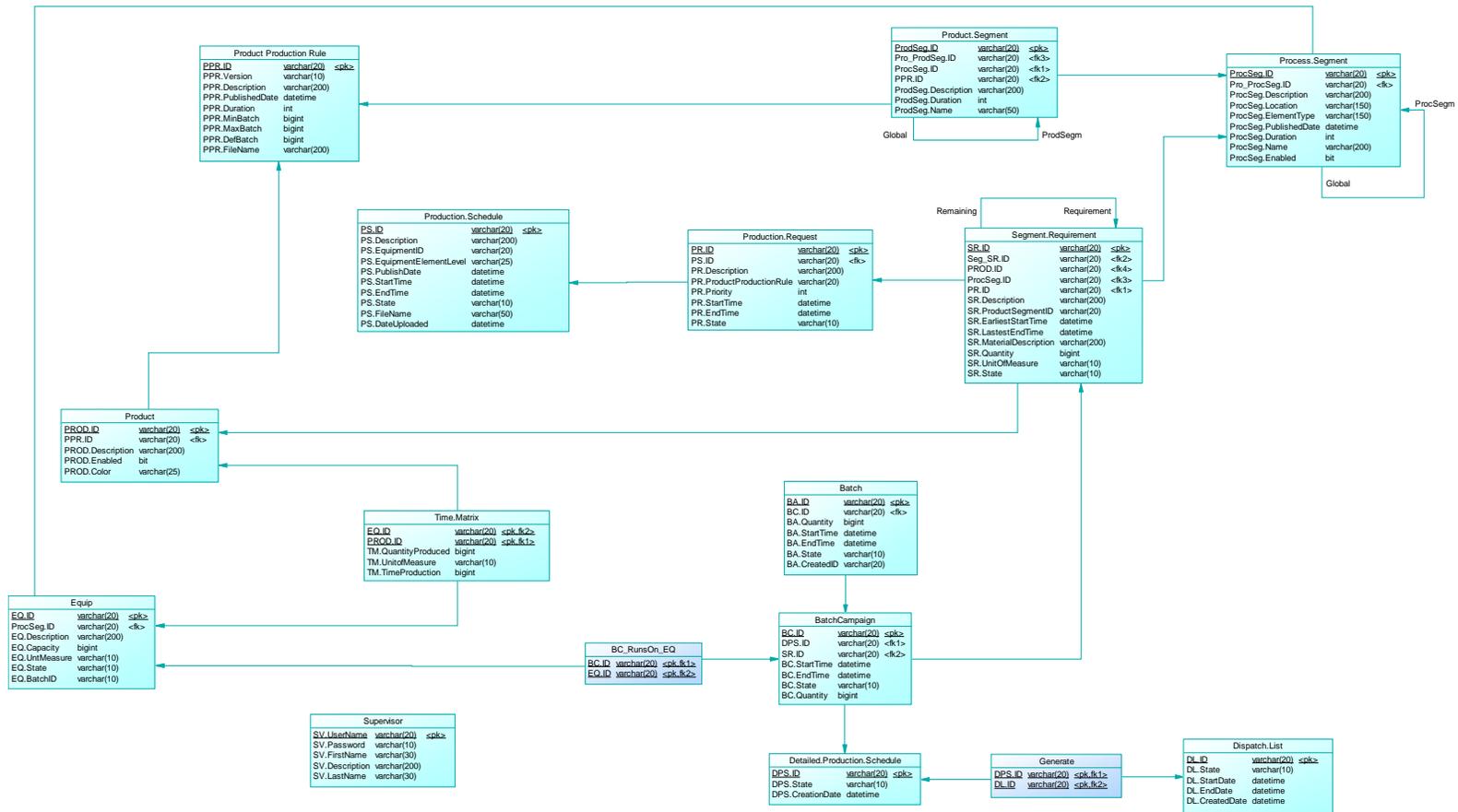


Figura 13. Modelo Físico de Datos (Relacional) Ver Anexo A

## **2.3.2 CONSTRUCCIÓN**

### **2.3.2.1 CICLO DE DESARROLLO NÚMERO 2**

Para este caso se muestra la aplicación de la metodología en el ciclo de desarrollo 2 con el fin de darle una continuidad a la etapa de planeación.

#### **2.3.2.1.1 ANÁLISIS**

##### **2.3.2.1.1.1 Definición de Diagramas de Secuencia**

Después de tener definidos el dominio del sistema y el diagrama de conceptos que son artefactos que describen el Modelo Estático, es necesario iniciar el modelado dinámico del sistema describiendo su comportamiento, lo cual se hace por medio de los diagramas de secuencia. Los artefactos que se utilizan como base para este modelo son los diagramas de caso de uso definidos anteriormente, específicamente la sección del curso normal de los eventos. El diagrama muestra los eventos externos del sistema los cuales son generados por los actores, estos eventos generan operaciones del sistema las cuales se registran en una tabla aparte que serán detalladas posteriormente por los contratos de operaciones, a continuación se muestra el desarrollo de los dos artefactos mencionados anteriormente.

Al considerar que este ciclo de desarrollo está conformado por dos versiones del caso de uso generar campañas batch se presenta un diagrama de secuencia consolidado de acerca de cómo los objetos interactúan para cumplir las funciones de Split y Merge, como se muestra en la Figura 14.

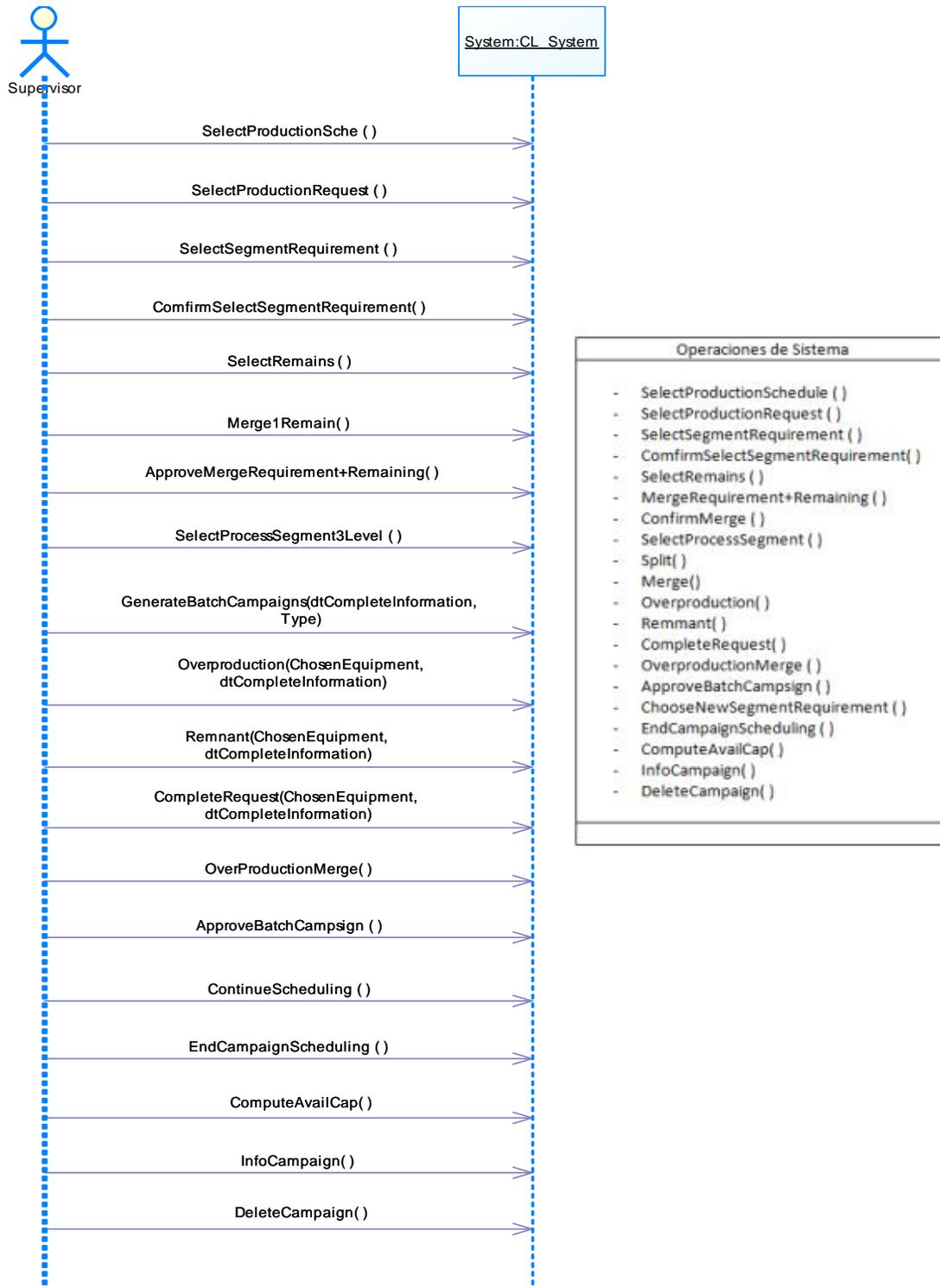


Figura 14. Diagrama de Secuencia del Caso de Uso Generar Campañas Batch (Split y Merge).

### 2.3.2.1.1.2 Definición de Contratos de Operaciones

A continuación se define un contrato para cada operación definida dentro del diagrama de secuencias de los casos de uso. Los contratos muestran el resultado de aplicar una operación en el sistema. Los contratos definidos se muestran a continuación.

#### 2.3.2.1.1.2.1 CASO DE USO SIMPLIFICADO: GENERAR CAMPAÑAS BATCH (SPLIT Y MERGE)

**Nombre:** SelectProductionSchedule( )

**Responsabilidades:** Este contrato es responsable de escoger el programa de producción con el que se da inicio a la generación de la campaña y el cual trae los requerimientos de producción deseados

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8

**Salida:** Requerimientos de producción del programa de producción seleccionado cargados

**Precondiciones:** Se debe haber cargado por lo menos un programa de producción proveniente del nivel ERP

**Poscondiciones:**

- Fue creada una instancia del programa de producción (Creación de Instancia)

**Nombre:** SelectProductionRequirement( )

**Responsabilidades:** Elegir el requerimiento de producción que se desee programar

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8

**Salida:** Requerimiento de producción seleccionado

**Precondiciones:** Haber elegido un programa de producción

**Poscondiciones:**

- Fue creada una instancia del Requerimiento de producción (Creación de Instancia)
- Fue asociado el Requerimiento de producción con el programa de producción (Asociación Creada)

**Nombre:** SelectSegmentRequirement( )

**Responsabilidades:** El supervisor debe elegir el requerimiento de segmento que desee

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8

**Salida:** Requerimiento de Segmento seleccionado

**Precondiciones:** Haber elegido un Requerimiento de Producción

**Poscondiciones:**

- Fue creada una instancia del Requerimiento de Segmento (Creación de Instancia)
- Fue asociado el Requerimiento de Segmento con el Requerimiento de Producción (Asociación Creada)

**Nombre:** ConfirmSelectSegmentRequirement()

**Responsabilidades:** El supervisor debe confirmar la elección del requerimiento de segmento

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8

**Salida:** Confirmación del Requerimiento de Segmento con el cual programar

**Precondiciones:** Haber seleccionado un Requerimiento de segmento

**Poscondiciones:**

- Fue cambiado el atributo estado de Requerimiento de Segmento SR.State (Modificación de atributos).

**Nombre:** SelectRemains()

**Responsabilidades:** En este contrato de operación se selecciona un remanente de producción con el mismo tipo de producto al del requerimiento seleccionado.

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8.

**Salida:** Remanente de producción elegido

**Precondiciones:** Exista por lo menos un remanente con el mismo tipo de ID al del producto requerido

**Poscondiciones:**

- Fue creada una instancia del remanente (Creación de Instancia)

**Nombre:** MergeRequirement+Remaining()

**Responsabilidades:** Se encarga de unir el requerimiento de segmento con el remanente seleccionado por el usuario

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8.

**Salida:** Nuevo requerimiento de segmento conformado listo para ser aprobado por el supervisor encargado

**Precondiciones:** Haber elegido un remanente de producción de la lista

**Poscondiciones:**

- Fue creada una instancia del remanente (Creación de Instancia)
- Fue asociado el Requerimiento de Segmento con el Remanente (Asociación Creada)

**Nombre:** ConfirmMerge()

**Responsabilidades:** Se encarga de confirmar la unión del remanente seleccionado con el requerimiento de segmento

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.2

**Salida:** Nuevo requerimiento de segmento conformado listo para ser programado

**Precondiciones:** Haber elegido un remanente de producción de la lista

**Poscondiciones:**

- Fue creada una instancia del remanente (Creación de Instancia)
- Fue asociado el Requerimiento de Segmento con el Remanente (Asociación Creada)
- Fue cambiado el atributo estado del requerimiento de segmento SR.State (Modificación de atributos).
- Fue cambiado el atributo estado del remanente SR.State (Modificación de atributos).

**Nombre:** SelectProcessSegment( )

**Responsabilidades:** Se encarga de seleccionar uno de los segmento de proceso de la planta de producción

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8.

**Salida:** Segmento de proceso seleccionado

**Precondiciones:** Tener por lo menos un Segmento de Proceso para ser seleccionados

**Poscondiciones:**

- Fue creada una instancia de segmento de proceso (Creación de Instancia)

- Fue asociado el Requerimiento de Segmento con el Requerimiento de segmento (Asociación Creada)

**Nombre:** Split (ChosenEquipment, dtCompleteInformation)

**Responsabilidades:** Este contrato de operación se encarga de realizar los cálculos necesarios para la generación de la campaña batch y determina si se generan todos los batches con la capacidad máxima del equipo produciendo únicamente la cantidad requerida

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.

**Salida:** El mejor equipo seleccionado, lista de batches

**Precondiciones:** Haber seleccionado un segmento de Proceso y tener la información necesaria para la realización de cálculos

**Poscondiciones:**

- Fueron creadas tantas instancia de batches como fuera necesario (Creación de Instancia)
- Fue creada una instancia de campañas batch (Creación de Instancia)
- Fue asociado el Requerimiento de Segmento con la campaña batch (Asociación Creada)
- Fue asociada la campaña batch con los batches (Asociación Creada)
- Fue cambiado el atributo cantidad del Batch Campaign. BC\_Quantity (Modificación de atributos).

**Nombre:** Merge(ChosenEquipment, dtCompleteInformation, dtRemains)

**Responsabilidades:** Este contrato de operación se encarga de realizar los cálculos necesarios para la generación de la campaña batch y determina si se genera el batch con la capacidad máxima del equipo produciendo únicamente la cantidad requerida

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.

**Salida:** El mejor equipo seleccionado, información del batch

**Precondiciones:** Haber seleccionado un segmento de Proceso y tener la información necesaria para la realización de cálculos

**Poscondiciones:**

- Fueron creada una instancia de batch (Creación de Instancia)
- Fue creada una instancia de campañas batch (Creación de Instancia)

- Fue asociado el Requerimiento de Segmento con la campaña batch (Asociación Creada)
- Fue asociada la campaña batch con el batch (Asociación Creada)
- Fue cambiado el atributo cantidad del Batch Campaign. BC\_Quantity (Modificación de atributos).

**Nombre:** Overproduction (ChosenEquipment, dtCompleteInformation)

**Responsabilidades:** Este contrato se encarga de programar el ultimo batch completo así se exceda la cantidad requerida con el fin aprovechar al máximo la capacidad del equipo y cumpliendo con el requerimiento.

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.

**Salida:** La campaña batch programada con todos sus batches al 100% de su capacidad incluso si este sobrepasa la cantidad requerida.

**Precondiciones:** Debe haber sido seleccionado un equipo, calculado el tiempo disponible, el tiempo requerido de la campaña, el sistema debió obtener un número de batches mayor que uno y no todos a la máxima capacidad del equipo y se debió haber elegido la opción de sobreproducir

**Poscondiciones:**

- Fue cambiado el atributo cantidad del Batch Campaign. BC\_Quantity (Modificación de atributos).
- Fue creada una instancia de campañas batch (Creación de Instancia)
- Fue creada una instancia de batch (Creación de Instancia)

**Nombre:** Remnant (ChosenEquipment, dtCompleteInformation)

**Responsabilidades:** Este contrato se encarga de programar hasta el último batch calculado con la capacidad máxima del equipo y la cantidad del batch incompleto se deja como un remanente para una futura programación.

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.

**Salida:** Campaña batch programada con todos los batches completos y generación de un remanente de producción

**Precondiciones:** Debe haber sido seleccionado un equipo, calculado el tiempo disponible, el tiempo requerido de la campaña, el

sistema debió obtener un número de batches mayor que uno y no todos a la máxima capacidad del equipo y se debió haber elegido la opción de crear remanentes

**Poscondiciones:**

- Fue creada una instancia de campañas batch (Creación de Instancia)
- Fue creada una instancia de batch (Creación de Instancia)
- Fue creada una instancia de requerimiento de segmento (Creación de instancia)

**Nombre:** CompleteRequest (ChosenEquipment, dtCompleteInformation)

**Responsabilidades:** Este contrato se encarga de programar todos los batches como fue calculado por el sistema sin importar que alguno de estos no aproveche al máximo la capacidad del equipo, pero cumpliendo con la cantidad requerida.

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.

**Salida:** La campaña batch programada como fue calculado por el sistema sin importar que alguno de los batches se programe con la capacidad máxima del equipo.

**Precondiciones:** Debe haber sido seleccionado un equipo, calculado el tiempo disponible, el tiempo requerido de la campaña, el sistema debió obtener un número de batches mayor que uno y no todos a la máxima capacidad del equipo y el usuario debió haber elegido la opción de cumplir con el requerimiento

**Poscondiciones:**

- Fue cambiado el atributo cantidad del Batch Campaign. BC\_Quantity (Modificación de atributos).
- Fue creada una instancia de campañas batch (Creación de Instancia)
- Fue creada una instancia de batch (Creación de Instancia)

**Nombre:** OverproductionMerge ( )

**Responsabilidades:** Este contrato se encarga de programar un Batch que excede la cantidad requerida con el fin de cumplir con el requerimiento aprovechando al máximo la capacidad del equipo.

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.7, R1.8, R2.1, R2.

**Salida:** La campaña batch programada con su batch al 100% de su capacidad incluso si este sobrepasa la cantidad requerida.

**Precondiciones:** Debe haber sido seleccionado un equipo, calculado el tiempo disponible, el tiempo requerido de la campaña, el sistema debió obtener un número de batches menor que además se debió haber elegido la opción de sobreproducir

**Poscondiciones:**

- Fue cambiado el atributo cantidad del Batch Campaign. BC\_Quantity (Modificación de atributos).
- Fue creada una instancia de campañas batch (Creación de Instancia)
- Fue creada una instancia de batch (Creación de Instancia)

**Nombre:** ApproveBatchCampaign( )

**Responsabilidades:** En este contrato el supervisor aprueba la campaña si lo considera correcto, en este caso se guarda una lista de campañas generadas

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8, R2.1, R2.2

**Salida:** Campaña batch aprobada

**Precondiciones:** Debe haber sido creada una campaña o campañas por el sistema para ser evaluadas por el supervisor

**Poscondiciones:**

- Fue cambiado el atributo estado del Batch BC\_State (Modificación de atributos).
- Fue cambiado el atributo estado de la campaña Batch. BC\_State (Modificación de atributos).
- Fue asociada la campaña batch con el administrador batch (Asociación Creada)

**Nombre:** ChooseNewSegmentRequirement ( )

**Responsabilidades:** Este contrato permite la elección de un nuevo requerimiento de segmento del programa de producción seleccionado

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8

**Salida:** nuevo requerimiento de segmento seleccionado

**Precondiciones:** Debe haber una lista de requerimientos de segmentos de la cual elegir

**Poscondiciones:**

- Fue creada una instancia del nuevo requerimiento de segmento (Creación de Instancia)

**Nombre:** EndCampaignScheduling ( )

**Responsabilidades:** Este contrato muestra al diagrama de Gantt correspondiente a las campañas batch que dan cumplimiento a los requerimientos de segmento deseados.

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8.

**Salida:** Programación detallada de la producción mostrada en un diagrama de Gantt

**Precondiciones:** Debe haber sido aprobada por lo menos una campaña

**Poscondiciones:**

- Fue creada una instancia de programa detallado de la producción (Creación de Instancia)
- Fue creada una instancia de lista de despacho (Creación de Instancia)
- Fueron asociadas las campañas batch con la lista de despacho el administrador batch (Asociación Creada)
- Fue asociada la lista de despacho con el administrador batch (Asociación Creada)

**Nombre:** ComputeAvailCap( )

**Responsabilidades:** Este contrato se encarga realizar el cálculo de la capacidad disponible de todos los equipos pertenecientes a la empresa

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8.

**Salida:** Capacidad disponible de todos los equipos calculados

**Precondiciones:** Tener la información de la programación actual de todos los equipos de la empresa

**Nombre:** InfoCampaign( )

**Responsabilidades:** Se encarga de mostrar al usuario la información correspondiente a la campaña seleccionada en la interfaz

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.8.

**Salida:** campaña Batch eliminada

**Precondiciones:** Campaña Batch con información no deseada o incorrecta seleccionada en interfaz por el usuario

**Nombre:** DeleteCampaign( )

**Responsabilidades:** Este contrato se encarga de eliminar la información correspondiente a una campaña batch elegida por el usuario.

**Tipo:** Operaciones de Sistema

**Referencias Cruzadas:** R1.3, R1.4, R1.5, R1.8.

**Salida:** campaña Batch eliminada

**Precondiciones:** Campaña Batch con información no deseada o incorrecta seleccionada en interfaz por el usuario

**Poscondiciones:**

- Fue eliminada una instancia de campaña Batch (Eliminación de Instancia)
- Fueron eliminadas tantas instancias de Batches como fueron necesarias (Eliminación de Instancia)

### 2.3.2.1.2 DISEÑO

En esta parte se pretende desarrollar la fase de diseño dentro de la fase de construcción del macronivel para el primer ciclo de desarrollo.

#### 2.3.2.1.2.1 Definición de los Diagramas de Interacción

El diagrama de Interacción permite no solo mostrar que se hace sino también el cómo, es decir, que es posible ver como los objetos interactúan con el fin de dar cumplimiento a sus respectivos objetivos, mediante dos tipos de diagramas, los de colaboración o los de secuencias. Particularmente para el modelado del prototipo se hace uso de los diagramas de secuencia, a manera de ejemplo se muestra en la Figura 15 el diagrama correspondiente a la versión 1 y 2 del caso de uso de generación de campañas batch el cual para una mejor observación se incluye en el anexo A.



### 2.3.2.1.2.2 Definición de los Diagramas de Diseño de Clase

El diagrama de diseño de clases que se realiza en cada ciclo de desarrollo, se muestra ya consolidado después del refinamiento a través de estos en la Figura 16. Para una mejor observación se incluye este diagrama en el anexo A.

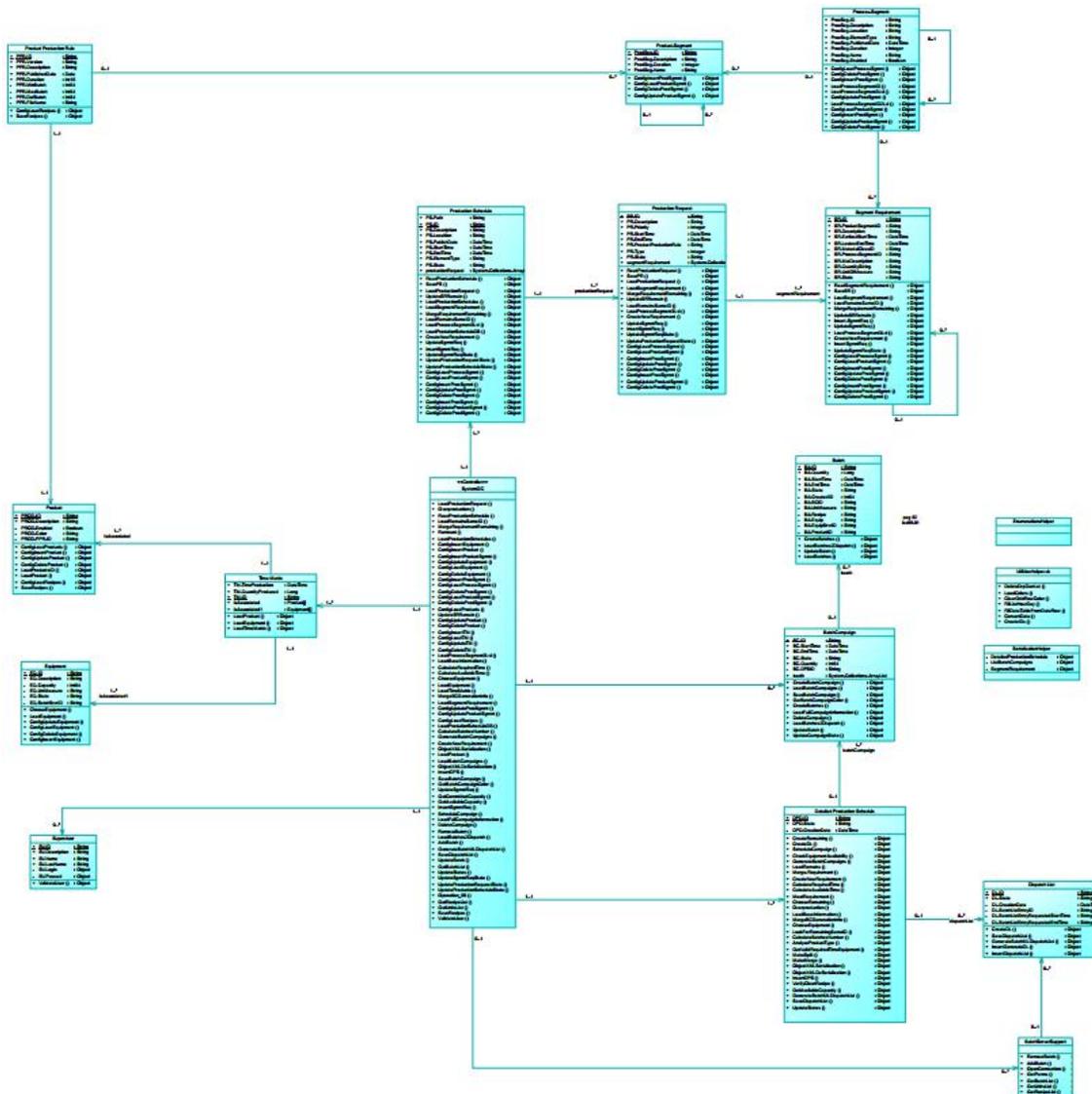


Figura 16. Diagrama de Diseño de Clases del Prototipo Ver Anexo A

## CAPITULO 3.

### *“Descripción del Proceso caso de estudio y puesta en contexto con el prototipo Software”*

Para el desarrollo de cualquier tipo de solución que busque resolver problemas de programación de la producción batch, es de vital importancia definir el modelo del proceso (topología, características, restricciones, etc.) que se desea programar, ya que una pequeña consideración puede desencadenar en un gran cambio en el análisis y diseño de éste, por lo cual, para el desarrollo del prototipo fue necesario definir un modelo de proceso genérico que trata de abarcar la mayor cantidad de aspectos aplicables en los problemas de programación de procesos Batch, en [4] se muestra una lista de las posibles opciones que se tienen en cuenta de las cuales se ha realizado una selección considerando el alcance, las características y requerimientos específicos del proyecto.

Dentro del análisis realizado de los aspectos puntualizados en [4] y a partir del modelo de segmento de proceso definido en el estándar ISA-95.01, ISA-95.02 se logra el desarrollo de una topología del proceso para el sistema prototipo que consta de tres niveles de segmentos de proceso especificados así: un segmento de proceso global o de primer nivel que hace referencia a toda la celda de proceso, cuyos requerimientos vienen dados en unidades empacadas de producto terminado, segmento que contiene a los segmentos de segundo nivel; los que se asignan teniendo en cuenta las etapas del proceso, las cuales son identificadas por realizar una actividad de procesamiento principal, los requerimientos asociados a estos vienen en unidades de productos intermedios, estos segmentos a su vez contienen los del tercer nivel; que se presentan con el fin de brindar flexibilidad en la programación y da la oportunidad de atender procesos con topologías

más complejas que dispongan de segmentos de proceso en paralelo los cuales realicen la misma actividad de procesamiento y que a su vez estarían compuestos de las unidades físicas de procesamiento o equipo en el piso de planta. En la Figura 17 se muestra el modelo del proceso planteado.

Con el fin de poner a prueba las funcionalidades implementadas en el prototipo, se elige un proceso batch en el cual sea posible representar cada escenario en el que se puedan evidenciar las labores del mismo.

Para el caso de estudio se escoge una empresa encargada de la producción y comercialización de pintura a base de agua tomando como referencia el modelado bajo el estándar ISA-88 realizado en [36]. Este proceso consta de cuatro etapas básicas para la producción: la obtención de pintura base, obtención de pintura a color, la etapa de filtrado y por último la etapa de envasado, etiquetado y almacenaje de pintura.

La obtención de pintura base, es la elaboración de una pintura de color neutro, que permite darle a cualquier pintura en todas las gamas de colores, textura, consistencia, y adherencia. Para esta etapa es necesaria la recepción y el pesaje de la materia prima que permite la consistencia y durabilidad de la pintura y que dependiendo de la dosis aplicada de talco (para la textura), titanio (cubrimiento en superficies), resina (adherencia), coagulantes y aditivos varía la calidad de la misma. Una vez determinada la cantidad requerida de materias prima, se procede a mezclarlas en agua, para así obtener después de un cierto tiempo la pintura base.

La segunda etapa o la obtención de pintura a color, consiste en la adición de una determinada cantidad de pigmentos de diferente color a la pintura base, según el color deseado. Una vez verificado la combinación de los colores de pigmentos (verde, azul, amarillo, negro y rojo) que dan origen al color solicitado, se procede a mezclar uniformemente de tal forma que se obtenga la pintura a color. En la etapa de filtrado, una vez verificado el color de la pintura, se pretende eliminar grumos e impurezas presentes en ésta por medio de una serie de mallas que se encargan de retener dichos elementos.

Por último en la etapa de envasado, etiquetado y almacenaje de pintura, se embala y se almacena la pintura filtrada en galones, cuñetes u otro tipo de presentación de tal forma que le permita conservar las propiedades físicas y químicas del producto, además de esto se realiza un etiquetado a las pinturas ya envasadas, en la que sea posible identificar, el lote, la cantidad, el color, el tipo de pintura realizada entre otra información relevante.

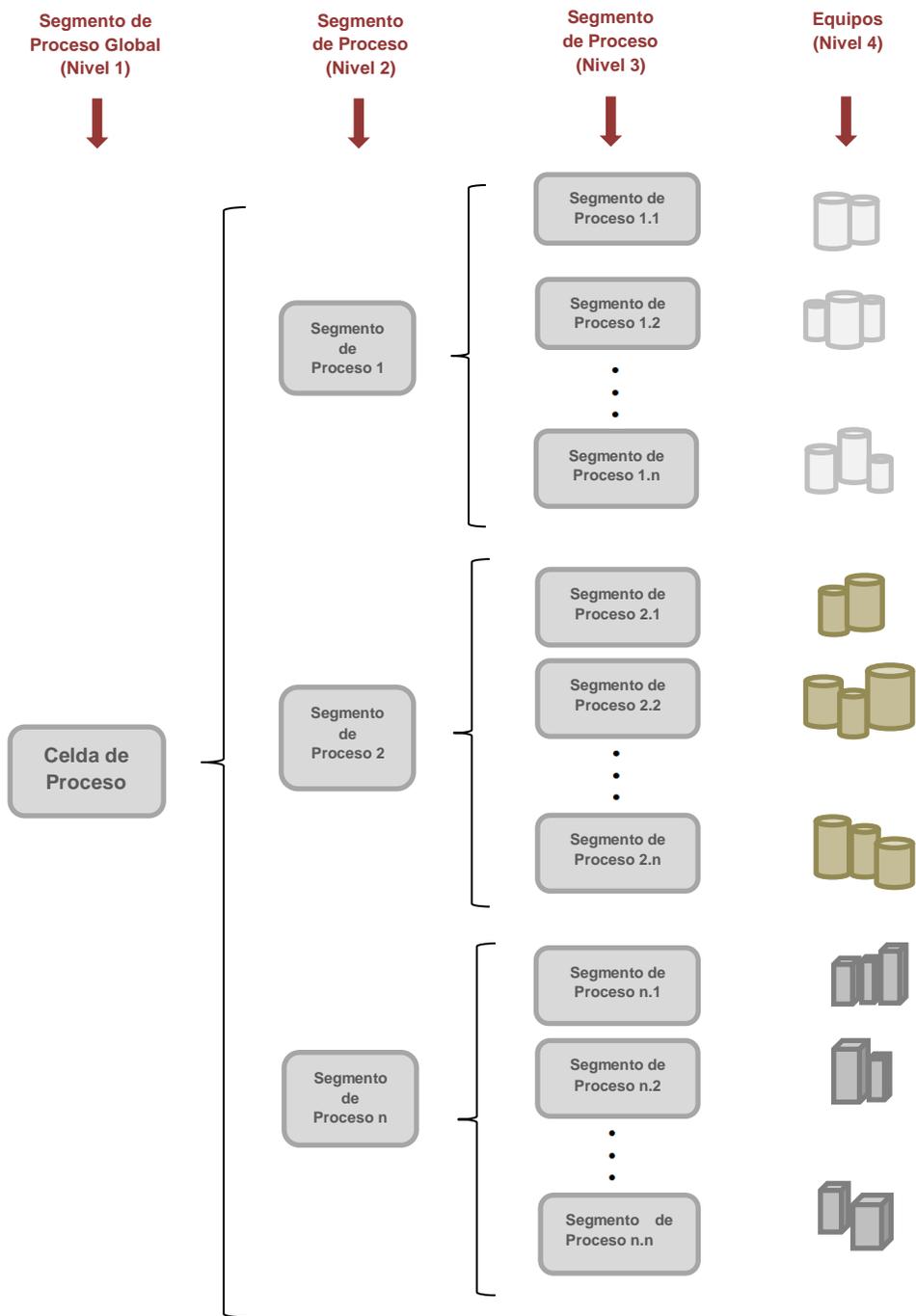


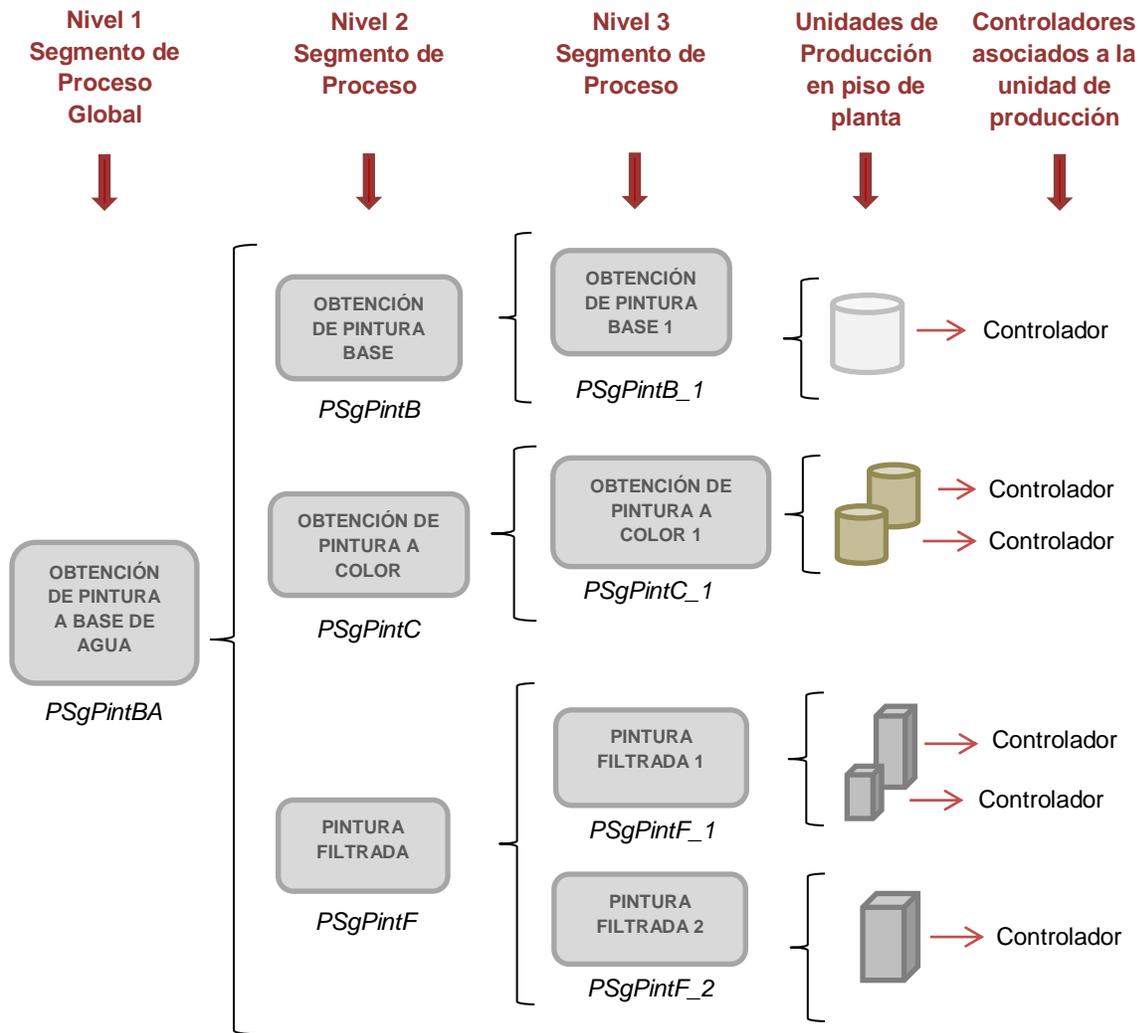
Figura 17. Ejemplo de topología de proceso general contemplado en el desarrollo del prototipo

Después de especificar la topología general aplicada en el prototipo y la estructura del proceso de producción de pintura, se procede a identificar la topología del proceso caso de estudio con base en la del prototipo.

La topología del proceso usado para la validación del prototipo inicia con la definición de un segmento de proceso global de obtención de pintura a base de agua identificado con el ID *PSgPintBA*, el cual recibe requerimientos de segmento según lo planteado por la programación de alto nivel, como por ejemplo 260 cuñetes de pintura azul. En el segundo nivel de segmento de proceso, se asignan únicamente las etapas batch del proceso: obtención de pintura base, pintura a color y pintura filtrada, con identificadores *PSgPintB*, *PSgPintC*, *PSgPintF* respectivamente, cuyos requerimientos se hacen a nivel de productos intermedios, por ejemplo 4920 litros de pintura base, 2700 galones de pintura a color, etc.

En el tercer nivel de segmento de proceso se encuentran los segmentos asociados a un mismo tipo, los cuales fabrican productos intermedios pertenecientes a una misma familia. Para el caso específico, se adoptan dos celdas de proceso de nivel tres únicamente para el proceso de filtrado de pintura, pues la licencia con la que se cuenta para el requisito de la integración con el administrador batch de Rockwell (Factory Talk Batch) limita el número de controladores que se pueden simular. En caso contrario se podrían definir más de un segmento de proceso en paralelo de tercer nivel en las otras etapas del proceso.

Finalmente, después de conocer que el número de controladores permitidos para simular son 6 y que además de múltiples segmentos de proceso de nivel 3 en las etapas del proceso, se requiere de múltiples unidades para la ejecución de los requerimientos, se define la topología del proceso mostrada en Figura 18 para la validación del prototipo desde la recepción del programa de producción, la generación de campañas y el despacho de la producción (Integración con el Administrador Batch).



**Figura 18. Topología definida del proceso de obtención de pintura a base de agua para la validación del prototipo**

Tomando como base la topología definida anteriormente a continuación se especifican los aspectos más importantes a tener en cuenta en la elaboración y configuración del prototipo de gestión de campañas batch iniciando con los segmentos de proceso definidos, los segmentos de producto, los equipos, los productos y la relación de tiempo de producción con cantidad de material producido en un respectivo equipo, los cuales permitirán evidenciar las funcionalidades implementadas en el proyecto.

### 3.1 DEFINICIÓN DE SEGMENTOS DE PROCESO, DE PRODUCTO, ESPECIFICACION DE EQUIPOS, PRODUCTOS Y MATRIZ DE TIEMPOS PARA EL CASO DE ESTUDIO

Para la definición de los segmentos de proceso, de productos y especificaciones tanto de los equipos, los productos y los tiempos de producción se tomará como referencia el trabajo realizado en [37] en donde se evidencia la aplicación del estándar ISA-95 en un caso de estudio.

#### 3.1.1 SEGMENTOS DE PROCESO BATCH

Los segmentos de proceso pueden ser considerados como la vista lógica del proceso para el sistema de negocios; dichos segmentos permiten establecer los recursos utilizados en cada uno de los pasos o etapas de producción común a varios productos en el nivel de detalle requerido por el sistema de negocios, además permiten anidar otros segmentos de proceso brindando flexibilidad en la aplicación del modelo [35], [37].

A continuación se especifica la información necesaria para los segmentos de proceso en los 3 niveles definidos anteriormente (ver de la Tabla 19 a la Tabla 26), tomando como referencia el ejemplo de aplicación del modelado ISA-95 mostrado en [37].

##### 3.1.1.1 SEGMENTO DE PROCESO GLOBAL (Nivel 1)

###### 3.1.1.1.1 Segmento de Proceso Global

ESPECIFICACIÓN DE SEGMENTO DE PROCESO				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA A BASE DE AGUA GLOBAL			
<b>ID</b>	PSgPintBA		<b>ID SEGM. ASOCIADO</b>	NO APLICA
<b>LOCALIZACIÓN</b>	<b>ID DEL EQUIPO</b>	CELL_PINTURA_BA	<b>TIPO DE ELEMENTO</b>	CELDA DE PROCESO
<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18		<b>DURACIÓN (HRS)</b>	4
<b>DESCRIPCIÓN</b>	PRODUCCION DE PINTURA A COLOR A BASE DE AGUA EN PRESENTACION DE PRODUCTO TERMINADO TIPO GALON, CUÑETE Y CUARTO.			

Tabla 19. Especificaciones del segmento de proceso global

### 3.1.1.2 SEGMENTO DE PROCESO DE NIVEL 2

#### 3.1.1.2.1 Segmento de Proceso de Pintura Base

ESPECIFICACIÓN DE SEGMENTO DE PROCESO				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA BASE NIVEL 2			
<b>ID</b>	PSgPintB		<b>ID SEGM. ASOCIADO</b>	PSgPintBA
<b>LOCALIZACIÓN</b>	<b>ID EL EQUIPO</b>	---	<b>TIPO DE ELEMENTO</b>	UNIDAD DE PRODUCCION
<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18		<b>DURACIÓN (HRS)</b>	2
<b>DESCRIPCIÓN</b>	PRODUCCION DE PINTURA BASE PARA LA POSTERIOR PRODUCCIÓN DE PINTURA A COLOR.			

Tabla 20. Especificación de Segmento de proceso de nivel 2 de Pintura Base

#### 3.1.1.2.2 Segmento de Proceso de Pintura Color

ESPECIFICACIÓN DE SEGMENTO DE PROCESO				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA COLOR NIVEL 2			
<b>ID</b>	PSgPintC		<b>ID SEGM. ASOCIADO</b>	PSgPintBA
<b>LOCALIZACIÓN</b>	<b>ID EL EQUIPO</b>	---	<b>TIPO DE ELEMENTO</b>	UNIDAD DE PRODUCCION
<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18		<b>DURACIÓN (HRS)</b>	1
<b>DESCRIPCIÓN</b>	ADICIÓN DE PIGMENTOS A LA PINTURA BASE PARA LA PRODUCCION DE PINTURA A COLOR			

Tabla 21. Especificación de Segmento de Proceso de Nivel 2 de Pintura a Color

#### 3.1.1.2.3 Segmento de Proceso de Filtrado

ESPECIFICACIÓN DE SEGMENTO DE PROCESO				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA FILTRADA NIVEL 2			
<b>ID</b>	PSgFilt		<b>ID SEGM. ASOCIADO</b>	PSgPintBA
<b>LOCALIZACIÓN</b>	<b>ID EL EQUIPO</b>	---	<b>TIPO DE ELEMENTO</b>	UNIDAD DE PRODUCCION

<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18	<b>DURACIÓN (HRS)</b>	1
<b>DESCRIPCIÓN</b>	FILTRADO DE PINTURA A COLOR PARA ELIMINAR IMPUREZAS		

Tabla 22. Especificación de Segmento de Proceso de Nivel 2 de Pintura Filtrada

### 3.1.1.3 SEGMENTO DE PROCESO DE NIVEL 3

#### 3.1.1.3.1 Segmento de Proceso de Pintura Base

<b>ESPECIFICACIÓN DE SEGMENTO DE PROCESO</b>				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA BASE NIVEL 3			
<b>ID</b>	PSgPintB_1	<b>ID SEGM. ASOCIADO</b>	PSgPintB	
<b>LOCALIZACIÓN</b>	<b>ID EL EQUIPO</b>	--	<b>TIPO DE ELEMENTO</b>	UNIDAD DE PRODUCCION
<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18		<b>DURACIÓN (HRS)</b>	2
<b>DESCRIPCIÓN</b>	PRODUCCION DE PINTURA BASE PARA LA POSTERIOR PRODUCCIÓN DE PINTURA A COLOR.			

Tabla 23. Especificación de Segmento de Proceso de Nivel 3 de Pintura Base

#### 3.1.1.3.2 Segmento de Proceso de Pintura Color

<b>ESPECIFICACIÓN DE SEGMENTO DE PROCESO</b>				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA COLOR NIVEL 3			
<b>ID</b>	PSgPintC_1	<b>ID SEGM. ASOCIADO</b>	PSgPintC	
<b>LOCALIZACIÓN</b>	<b>ID EL EQUIPO</b>	--	<b>TIPO DE ELEMENTO</b>	UNIDAD DE PRODUCCION
<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18		<b>DURACIÓN (HRS)</b>	1
<b>DESCRIPCIÓN</b>	ADICIÓN DE COMPONENTES A LA PINTURA BASE PARA LA PRODUCCION DE PINTURA A COLOR			

Tabla 24. Especificación de Segmento de Proceso de Nivel 3 de Pintura a Color

### 3.1.1.3.3 Segmento de Proceso de Filtrado 1

ESPECIFICACIÓN DE SEGMENTO DE PROCESO				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA FILTRADA NIVEL 3.1			
<b>ID</b>	PSgPintF_1	<b>ID SEGM. ASOCIADO</b>	PSgPintF	
<b>LOCALIZACIÓN</b>	<b>ID EL EQUIPO</b>	--	<b>TIPO DE ELEMENTO</b>	UNIDAD DE PRODUCCION
<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18		<b>DURACIÓN (HRS)</b>	1
<b>DESCRIPCIÓN</b>	FILTRADO DE PINTURA A COLOR PARA ELIMINAR IMPUREZAS			

Tabla 25. Especificación de Segmento de Proceso de Nivel 3 de Pintura Filtrada 1

### 3.1.1.3.4 Segmento de Proceso de Filtrado 2

ESPECIFICACIÓN DE SEGMENTO DE PROCESO				
<b>NOMBRE DEL SEGMENTO</b>	PINTURA FILTRADA NIVEL 3.2			
<b>ID</b>	PSgPintF_2	<b>ID SEGM. ASOCIADO</b>	PSgPintF	
<b>LOCALIZACIÓN</b>	<b>ID EL EQUIPO</b>	--	<b>TIPO DE ELEMENTO</b>	UNIDAD DE PRODUCCION
<b>FECHA DE PUBLICACIÓN</b>	2008-07-01T20:18		<b>DURACIÓN (HRS)</b>	1
<b>DESCRIPCIÓN</b>	FILTRADO DE PINTURA A COLOR PARA ELIMINAR IMPUREZAS			

Tabla 26. Especificación de Segmento de Proceso de Nivel 3 de Pintura Filtrada 2

### 3.1.1.4 DEPENDENCIAS ENTRE LOS SEGMENTOS DE PROCESO

La dependencia entre los diferentes segmentos está establecida por la definición y configuración de los diferentes procesos. Ver Figura 19

En la definición del proceso de manufactura de pintura se determinan las siguientes dependencias:

- ✓ **Pintura Base:** Inicio
- ✓ **Pintura a Color:** solo puede iniciar después de terminar el segmento de *Pintura Base*.

- ✓ **Filtrado:** solo puede iniciar después de terminar el segmento de *Pintura a Color*.
- ✓ **Envasado y etiquetado:** solo puede iniciar después de terminar el segmento de *Pintura Filtrada*.

#### 3.1.1.4.1 Enrutamiento y dependencias de materiales

La dependencia entre los segmentos de procesos para el proceso producción de pintura establece la dependencia y un enrutamiento de los materiales, de la siguiente manera:

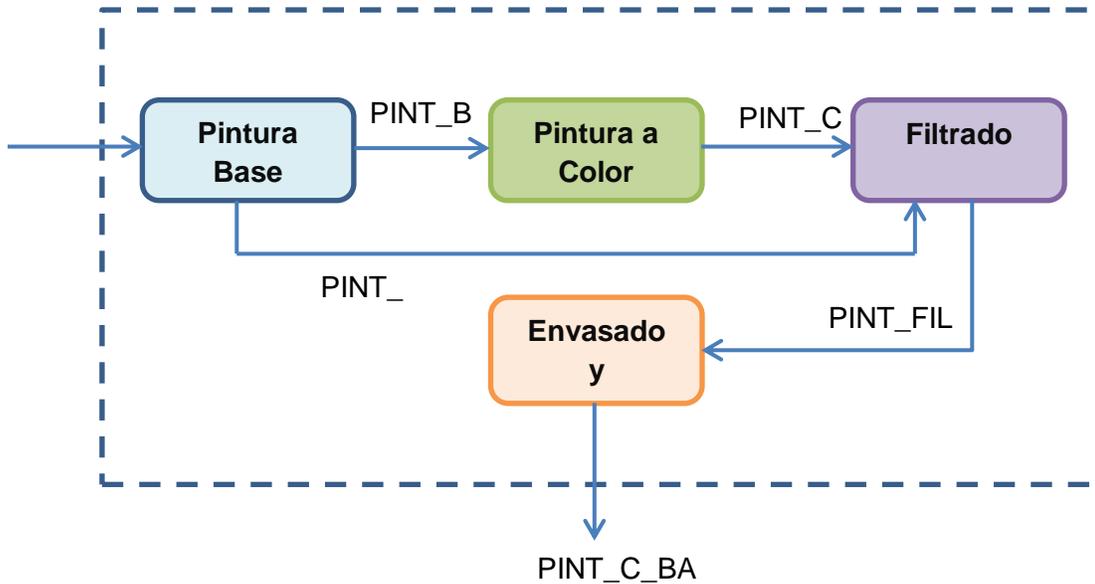


Figura 19. Enrutamiento y dependencia de materiales de los segmentos de proceso

#### 3.1.2 SEGMENTOS DE PRODUCTO BATCH

A diferencia de los segmentos de proceso los segmentos de producto determinan las especificaciones de materiales requeridos para la elaboración de una cantidad de producto en particular, por lo cual se puede considerar a los segmentos de productos son una instancia de los segmentos de proceso [37].

A continuación se especifica la información necesaria para los segmentos de producto para los segmentos de proceso definidos anteriormente (ver de la Tabla 27 a la Tabla 32), tomando como referencia el ejemplo de aplicación del modelado ISA-95 mostrado en [37].

### 3.1.2.1 SEGMENTO DE PRODUCTO DE GLOBAL

ESPECIFICACIÓN DE SEGMENTO DE PRODUCTO			
<b>NOMBRE DEL SEGMENTO</b>	PINTURA A BASE DE AGUA		
<b>ID</b>	PdSgPintBA	<b>ID SEGM. PROCESO ASOCIADO</b>	PSgPintBA
<b>FECHA DE PUBLICACIÓN</b>	2008-10-30T20:18	<b>DURACIÓN (HRS)</b>	4
<b>DESCRIPCIÓN</b>	SEGMENTO DE PRODUCTO PARA LA PRODUCCION DE PINTURA A COLOR A BASE DE AGUA EN PRESENTACION DE PRODUCTO TERMINADO TIPO GALON, CUÑETE Y CUARTO.		

Tabla 27. Especificación del segmento de producto Global

### 3.1.2.2 SEGMENTO DE PRODUCTO DE PINTURA BASE

ESPECIFICACIÓN DE SEGMENTO DE PRODUCTO			
<b>NOMBRE DEL SEGMENTO</b>	PINTURA BASE		
<b>ID</b>	PdSgPintB	<b>ID SEGM. ASOCIADO</b>	PSgPintB
<b>FECHA DE PUBLICACIÓN</b>	2008-10-30T20:18	<b>DURACIÓN (HRS)</b>	2
<b>DESCRIPCIÓN</b>	SEGMENTO DE PRODUCTO PARA LA PRODUCCION DE PINTURA BASE PARA LA POSTERIOR PRODUCCIÓN DE PINTURA A COLOR.		

Tabla 28. Especificación del segmento de producto para la producción de Pintura Base

### 3.1.2.3 SEGMENTO DE PRODUCTO DE PINTURA COLOR

#### 3.1.2.3.1 Segmento de producto de pintura a color rojo

ESPECIFICACIÓN DE SEGMENTO DE PRODUCTO			
<b>NOMBRE DEL SEGMENTO</b>	PINTURA COLOR ROJO		
<b>ID</b>	PdSgPintCR	<b>ID SEGM. ASOCIADO</b>	PSgPintC
<b>FECHA DE PUBLICACIÓN</b>	2008-10-30T20:18	<b>DURACIÓN (HRS)</b>	1

<b>DESCRIPCIÓN</b>	SEGMENTO DE PRODUCTO PARA LA PRODUCCION DE PINTURA A COLOR ROJA
--------------------	---

Tabla 29. Especificación del segmento de producto para la producción de Pintura a Color Roja

### 3.1.2.3.2 Segmento de producto de pintura a color Amarilla

<b>ESPECIFICACIÓN DE SEGMENTO DE PRODUCTO</b>			
<b>NOMBRE DEL SEGMENTO</b>	PINTURA COLOR AMARILLO		
<b>ID</b>	PdSgPintCA	<b>ID SEGM. ASOCIADO</b>	PSgPintC
<b>FECHA DE PUBLICACIÓN</b>	2008-10-30T20:18	<b>DURACIÓN (HRS)</b>	1
<b>DESCRIPCIÓN</b>	SEGMENTO DE PRODUCTO PARA LA PRODUCCION DE PINTURA A COLOR AMARILLA		

Tabla 30. Especificación del segmento de producto para la producción de Pintura a Color Amarillo

### 3.1.2.3.3 Segmento de producto de pintura a color Verde

<b>ESPECIFICACIÓN DE SEGMENTO DE PRODUCTO</b>			
<b>NOMBRE DEL SEGMENTO</b>	PINTURA COLOR VERDE		
<b>ID</b>	PdSgPintCV	<b>ID SEGM. ASOCIADO</b>	PSgPintC
<b>FECHA DE PUBLICACIÓN</b>	2008-10-30T20:18	<b>DURACIÓN (HRS)</b>	1
<b>DESCRIPCIÓN</b>	SEGMENTO DE PRODUCTO PARA LA PRODUCCION DE PINTURA A COLOR VERDE		

Tabla 31. Especificación del segmento de producto para la producción de Pintura a Color Verde

### 3.1.2.4 SEGMENTO DE PRODUCTO DE FILTRADO

<b>ESPECIFICACIÓN DE SEGMENTO DE PRODUCTO</b>			
<b>NOMBRE DEL SEGMENTO</b>	PINTURA FILTRADA		
<b>ID</b>	PdSgPintF	<b>ID SEGM. ASOCIADO</b>	PSgPintF
<b>FECHA DE PUBLICACIÓN</b>	2008-10-30T20:18	<b>DURACIÓN (HRS)</b>	1

<b>DESCRIPCIÓN</b>	SEGMENTO DE PRODUCTO PARA EL FILTRADO DE PINTURA A COLOR PARA LA ELIMINACION DE IMPUREZAS
--------------------	---

Tabla 32. Especificación del segmento de producto para la Producción de Pintura Filtrada

### 3.1.3 ESPECIFICACION DE LAS UNIDADES DEL PROCESO

A continuación se presentan los equipos o las unidades de producción correspondientes a cada segmento de proceso de nivel 3 que presenta el proceso de producción de pintura a base de agua para la validación del prototipo (Ver de la Tabla 33 a la Tabla 36).

#### 3.1.3.1 UNIDADES O EQUIPOS DEL SEGMENTO DE PROCESO DE PRODUCCION DE PINTURA BASE 1

ESPECIFICACION DE SEGMENTO DE EQUIPO					
EQUIPOS UTILIZADOS EN EL SEGMENTO	ID CLASE DE EQUIPOS	CANTIDAD	ID	CAPACIDAD	UNIDAD DE MEDIDA
Unidad de pintura base	CL_PB	1	UNIT_PB1	3.400	Glns.
<b>DESCRIPCIÓN</b>			Unidad de Pintura base 1.1		

Tabla 33. Especificación de equipos o unidades del segmento de proceso de Pintura Base de Nivel 3

#### 3.1.3.2 UNIDADES O EQUIPOS DEL SEGMENTO DE PROCESO DE PRODUCCION DE PINTURA COLOR 1

ESPECIFICACION DE SEGMENTO DE EQUIPO					
EQUIPOS UTILIZADOS EN EL SEGMENTO	ID CLASE DE EQUIPOS	CANTIDAD	ID	CAPACIDAD	UNIDAD DE MEDIDA
Unidad de pintura a color	CL_PC	2	UNIT_PC1	2.000	Glns.
			UNIT_PC2	1.500	
<b>DESCRIPCIÓN</b>			Unidad de Pintura a Color 1.1 y Unidad de Pintura a Color 1.2		

Tabla 34. Especificación de equipos o unidades del segmento de proceso de Pintura a Color de Nivel 3

**3.1.3.3 UNIDADES O EQUIPOS DEL SEGMENTO DE PROCESO DE PRODUCCION DE PINTURA FILTRADO 1**

<b>ESPECIFICACION DE SEGMENTO DE EQUIPO</b>					
<b>EQUIPOS UTILIZADOS EN EL SEGMENTO</b>	<b>ID CLASE DE EQUIPOS</b>	<b>CANTIDAD</b>	<b>ID</b>	<b>CAPACIDAD</b>	<b>UNIDAD DE MEDIDA</b>
Unidad de pintura filtrada	CL_PF	3	UNIT_FIL1	600	Glns.
			UNIT_FIL2	1.000	
<b>DESCRIPCIÓN</b>			Unidad de Filtrado 1.1, Unidad de Filtrado 1.2		

Tabla 35. Especificación de equipos o unidades del segmento de proceso de Pintura Filtrada 1 de Nivel 3

**3.1.3.4 UNIDADES O EQUIPOS DEL SEGMENTO DE PROCESO DE PRODUCCION DE PINTURA FILTRADO 2**

<b>ESPECIFICACION DE SEGMENTO DE EQUIPO</b>					
<b>EQUIPOS UTILIZADOS EN EL SEGMENTO</b>	<b>ID CLASE DE EQUIPOS</b>	<b>CANTIDAD</b>	<b>ID</b>	<b>CAPACIDAD</b>	<b>UNIDAD DE MEDIDA</b>
Unidad de pintura filtrada	CL_PF	1	UNIT_FIL3	2.000	Glns.
<b>DESCRIPCIÓN</b>			Unidad de Filtrado 2.1		

Tabla 36. Especificación de equipos o unidades del segmento de proceso de Pintura filtrada 2 de Nivel 3

**3.1.4 ESPECIFICACIÓN DE PRODUCTOS**

En la Tabla 37 mostrada a continuación se evidencian los productos que intervienen en el proceso de fabricación de pintura, dentro de los cuales se detallan tanto los productos terminados como intermedios que se van a manejar para el proceso caso de estudio.

<b>PRODUCTOS RELACIONADOS EN LA PRODUCCION DE PINTURA A BASE DE AGUA</b>		
<b>ID</b>	<b>DESCRIPCION</b>	<b>TIPO</b>
FGP-001	Producto terminado de pintura blanca tipo de presentación galón.	Producto Terminado
FGP-002	Producto terminado de pintura roja tipo de presentación cuñete.	Producto Terminado
FGP-003	Producto terminado de pintura amarilla tipo de presentación cuñete.	Producto Terminado
FGP-004	Producto terminado de pintura verde tipo de presentación cuñete.	Producto Terminado
IGP-001	Producto Intermedio resultado de la etapa de pintura base.	Producto Intermedio
IGP -002	Producto Intermedio resultado de la etapa de pintura color roja.	Producto Intermedio
IGP -003	Producto Intermedio resultado de la etapa de pintura color amarilla.	Producto Intermedio
IGP -004	Producto Intermedio resultado de la etapa de pintura color verde.	Producto Intermedio
IGP -005	Producto Intermedio resultado de la etapa de filtrado de pintura base o Blanca	Producto Intermedio
IGP -006	Producto Intermedio resultado de la etapa de filtrado de pintura roja.	Producto Intermedio
IGP-007	Producto Intermedio resultado de la etapa de filtrado de pintura amarilla.	Producto Intermedio
IGP-008	Producto Intermedio resultado de la etapa de filtrado de pintura verde.	Producto Intermedio
CP-001	Resultado de la Limpieza del Equipo de la etapa de pintura a color	Producto Intermedio
CP-002	Resultado de la Limpieza del Equipo de la etapa de filtrado	Producto Intermedio

**Tabla 37. Productos terminados e intermedios del proceso de Producción de Pintura a Base de Agua**

### **3.1.5 MATRIZ DE TIEMPOS**

En la Tabla 38 que se muestra a continuación se evidencia la relación existente entre el tiempo de producción requerido para la manufactura de una determinada cantidad de producto sobre un equipo específico, información de vital importancia en el proceso de programación ya que permite determinar la duración de los requerimientos que se hacen a cada equipo de los segmentos definidos.

<b>EQUIPO</b>	<b>ID EQUIPO</b>	<b>PRODUCTO</b>	<b>TIEMPO DE DURACION (HRS)</b>	<b>CANTIDAD</b>	<b>UND. DE MEDIDA</b>
<b>UNIDAD DE PINTURA BASE 1.1</b>	UNIT_PB1	IGP-001	2	3.400	Glns.
<b>UNIDAD DE PINTURA COLOR 1.1</b>	UNIT_PC1	IGP -002	1	2.000	Glns.
		IGP -003	1	2.000	Glns.
		IGP -004	1	2.000	Glns.
		CP-001	1	2.000	Glns.
<b>UNIDAD DE PINTURA COLOR 1.2</b>	UNIT_PC2	IGP -002	1	1.500	Glns.
		IGP -003	1	1.500	Glns.
		IGP -004	1	1.500	Glns.
		CP-001	1	1.500	Glns.
<b>UNIDAD DE FILTRADO 1.1</b>	UNIT_FIL1	IGP -006	1	600	Glns.
		IGP -007	1	600	Glns.
		IGP -008	1	600	Glns.
		CP-002	1	600	Glns.
<b>UNIDAD DE FILTRADO 1.2</b>	UNIT_FIL2	IGP -006	1	1.000	Glns.
		IGP -007	1	1.000	Glns.
		CP-002	1	1.000	Glns.
<b>UNIDAD DE FILTRADO 2.1</b>	UNIT_FIL3	IGP -005	1	2.000	Glns.
		IGP -008	1	2.000	Glns.
		CP-002	1	2.000	Glns.

**Tabla 38. Matriz de Tiempo definida**

Por último se muestra en la Figura 20 el diagrama definido de la distribución del proceso de producción de pintura base de agua y concluyendo de esta manera la contextualización del prototipo con el proceso caso de estudio.

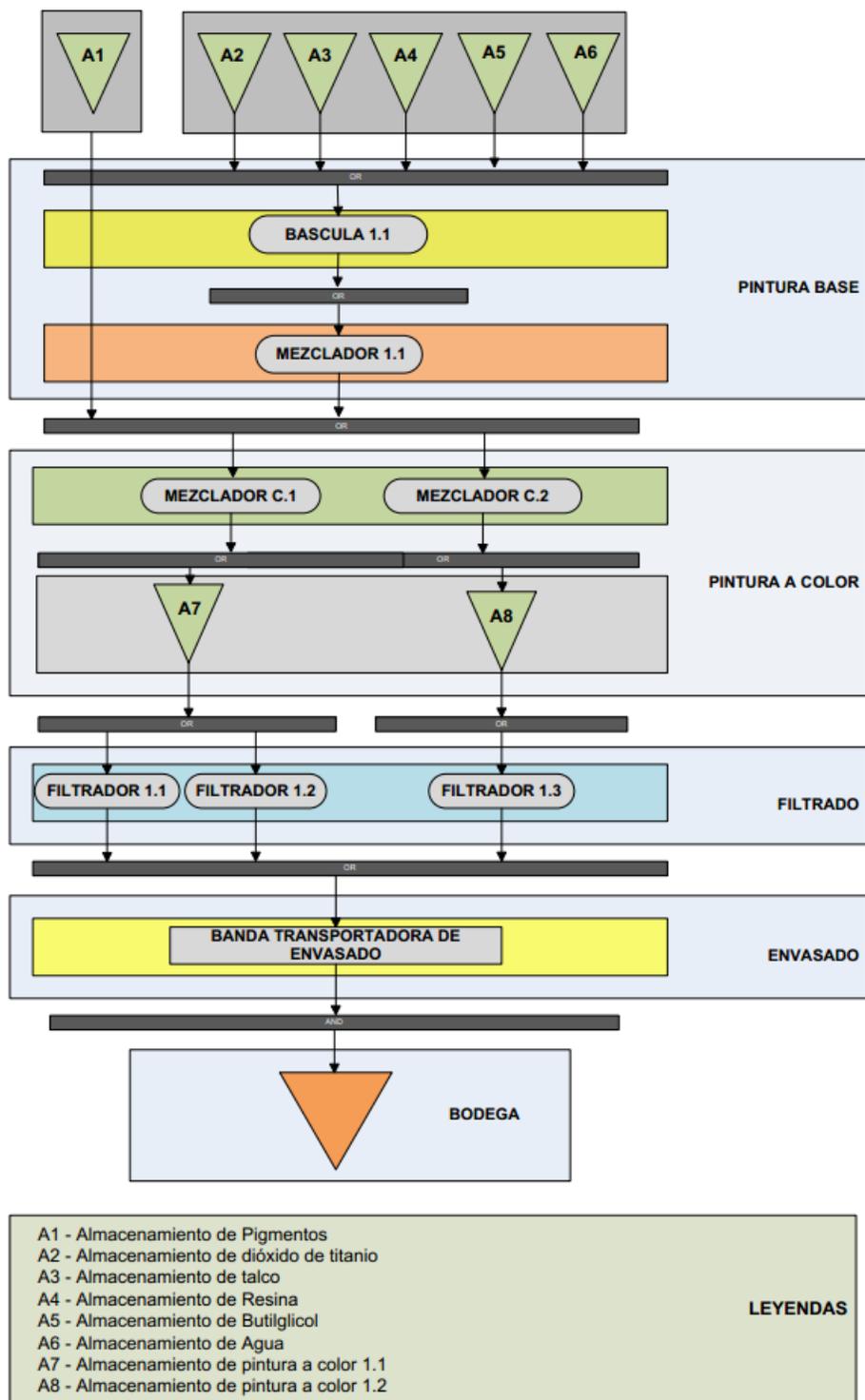


Figura 20. Proceso de Obtención de Pintura a Base de Agua

## CAPITULO 4.

### *“Verificación y Validación del prototipo software”*

Dentro del desarrollo de software y sistemas de información un aspecto que es de gran importancia es la calidad de los productos generados, ya que es una medida que sirve para determinar qué tan bueno es el producto desarrollado con respecto a su especificación funcional y las necesidades planteadas por el cliente.

Para poder llevar a cabo un análisis sobre lo desarrollado con respecto a lo requerido se cuenta con una serie de herramientas conocidas como los procesos de Verificación y Validación (V&V) que tienen como objetivo determinar si el software es útil en una situación operativa además de mejorar la calidad por medio de la detección de fallos y omisiones durante todo el ciclo de vida del desarrollo de software, permitiendo así corregir en el momento oportuno posibles situaciones que podrían llevar al incumplimiento de los requerimientos o a funcionalidades no adecuadas.

Al considerar la similitud entre los procesos de verificación y validación, se ve necesario definirlos de tal forma que se eviten confusiones entre ellos. La verificación busca determinar las falencias del software en cada uno de los ciclos de desarrollo verificando el cumplimiento de los requerimientos funcionales y no funcionales planteados en estos, haciéndose la pregunta a continuación: *¿se está construyendo el producto correctamente?* Por otro lado, la validación consiste en determinar el cumplimiento de las necesidades del cliente por medio de las funcionalidades del sistema, las cuales toman como base los requerimientos definidos inicialmente, por lo general se plantea el siguiente interrogante: *¿se está construyendo el producto correcto?* [38].

Para el proceso de verificación y validación son empleadas principalmente dos técnicas: las inspecciones y las pruebas del software; la inspección del software consiste en

encontrar falencias en el desarrollo mediante el análisis de documentos de necesidades, diagramas de diseño o código sin que el sistema sea ejecutado, convirtiéndose así en una técnica estática que se encarga de contrastar la relación entre lo desarrollado y lo requerido, pero no puede determinar que el producto desarrollado sea útil en una situación operativa; las pruebas de software se definen como *“una actividad en la cual un sistema o uno de sus componentes se ejecuta en circunstancias previamente especificadas, los resultados se observan, se registran y se realiza una evaluación de algún aspecto”* [39], consiste en analizar el comportamiento de cada una de las operaciones implementadas en el producto desarrollado en busca de posibles fallas, por medio de la ejecución del sistema volviéndose así una técnica dinámica [38] [40].

Dentro de las fallas que se pueden presentar, se encuentran fallas en datos, de control, de entrada/salida, de interfaz y de gestión de excepciones. Algunas de las fallas de datos que se pueden presentar son: no inicializar una variable antes de ser empleada, falta de nombres en constantes, se excede el tamaño máximo de una variable, entre otras; las fallas de control que se pueden presentar son: un condicional incorrecto dentro de una sentencia, presencia de ciclos infinitos, no se tienen en cuenta todos los casos posibles de las instrucciones case; las fallas de entrada y salida más comunes son: no se utilizan todas las variables declaradas en una función, no todas las variables de salida tienen definido un valor; los fallos de la interfaz pueden ser: el número de parámetros de las funciones y métodos que se llaman tienen el valor y la posición incorrecta de sus parámetros, hay funciones creadas que no se utilizan; los fallos de gestión de excepciones se refiere a que no todos los casos de los posibles errores han sido considerados [38].

Es importante considerar que tanto los procesos de verificación y validación se efectuaron a lo largo de los ciclos de desarrollo definidos en el proyecto, además teniendo en cuenta que al finalizar los ciclos se obtienen prototipos software funcionales característica que permite realizar tanto el análisis estático como el dinámico brindando la capacidad de identificar los fallos, defectos y omisiones, disminuyendo así los costos de las correcciones, al ser desarrollado estratégicamente en la etapa de sincronización de artefactos planteada en el mapa de la metodología utilizada.

#### **4.1 APLICACIÓN DE LA VERIFICACIÓN Y VALIDACIÓN DEL PROTOTIPO**

Con el fin de evidenciar la aplicación de los procesos V&V a lo largo de los ciclos de desarrollo se definieron el uso de técnicas de análisis estáticas y dinámicas aplicadas a un solo ciclo de desarrollo (Ciclo de Desarrollo 2) como se muestra a continuación, los restantes se evidencian en el anexo B.

En los métodos estáticos de V&V se encuentran como herramienta las inspecciones de software, que principalmente se encargan de revisar manualmente y en detalle los

diagramas de diseño, código fuente (sin ejecutar), documentos de requerimientos y análisis en general con el fin de determinar los defectos existentes, corrigiendo y sincronizando así todos estos artefactos. Dentro de las etapas de la inspección se encuentran [41]:

1. Inicio

En esta etapa se pone a disposición toda la información del artefacto que se va a inspeccionar.

2. Detección de defectos

Cada miembro del equipo de desarrollo realiza una lectura exhaustiva y profunda de la documentación disponible para comprender el artefacto y así detectar las faltas, realizando una estimación de la cantidad de faltas encontradas en el artefacto.

3. Colección de defectos

En esta etapa se realiza una reunión de los resultados del proceso de detección de defectos unificando estos y obteniendo un consolidado de defectos los cuales se evalúan tratando de determinar si estos fueron correctamente identificados y si se quedaron defectos por detectar.

4. Corrección y Seguimiento

El autor del artefacto se encarga de corregir los defectos encontrados y consolidados anteriormente.

Dentro del ciclo de vida de desarrollo del prototipo se realizan inspecciones basados en listas de chequeo o checklist, herramienta de fácil uso que permite mantener el seguimiento de procesos y la medición de la efectividad de funcionalidades implementadas facilitando el entendimiento del artefacto a la persona encargada de realizar la inspección ya que la guían por medio de preguntas claves en el transcurso del proceso [42].

Para el análisis dinámico en los procesos de V&V se encuentran como herramienta las pruebas de software, que como se planteó anteriormente, consisten en definir un entorno controlado para que cuando el sistema sea utilizado en presencia de una serie de entradas predefinidas genere una serie de salidas esperadas [43] :

Para realizar una prueba efectiva es necesario tener en cuenta una serie de pasos que sirven de guía en el proceso de aplicación de una prueba, dicho pasos son:

1. Diseño de las pruebas.

En esta actividad se define la técnica a utilizar para la aplicación de la prueba.

**2. Generación de los casos de prueba.**

Al generar casos de prueba lo que se busca es determinar el conjunto de entradas, que satisfagan determinadas condiciones que se quieren probar con el fin de cumplir el objetivo de prueba, además se deben definir claramente los resultados esperados en el momento de la aplicación de las entradas.

**3. Definición de los procedimientos de la prueba.**

Este apartado se encarga de definir los detalles de la prueba, es decir, quien es el responsable de ejecutarla, cómo se va a ejecutar, bajo qué condiciones, cuándo entre otros aspectos pertinentes según considere el diseñador de la prueba.

**4. Ejecución de la prueba**

La aplicación de la prueba al software a evaluar teniendo en cuenta las entradas y los casos de prueba definidos y comparando los resultados esperados con los obtenidos.

**5. Realización de un informe de la prueba**

Por último se realiza un informe que consigne el resultado de la ejecución de las pruebas, casos de prueba satisfactorios, cuáles no, y los fallos detectados.

Teniendo claro el proceso a seguir en el momento de definir y ejecutar una prueba es necesario especificar el tipo de prueba a realizar ya que según su clasificación puede variar la forma de proceder.

Las pruebas de software se pueden clasificar como pruebas de caja blanca y de caja negra. Las pruebas de caja blanca son aquellas en las cuales se busca diseñar la prueba teniendo en cuenta el comportamiento interno y la estructura del código fuente a diferencia de las de caja negra que únicamente se conocen previamente las salidas esperadas al aplicar unas entradas determinadas y según este resultado se evalúa si el funcionamiento es el indicado (Ver Figura 21).

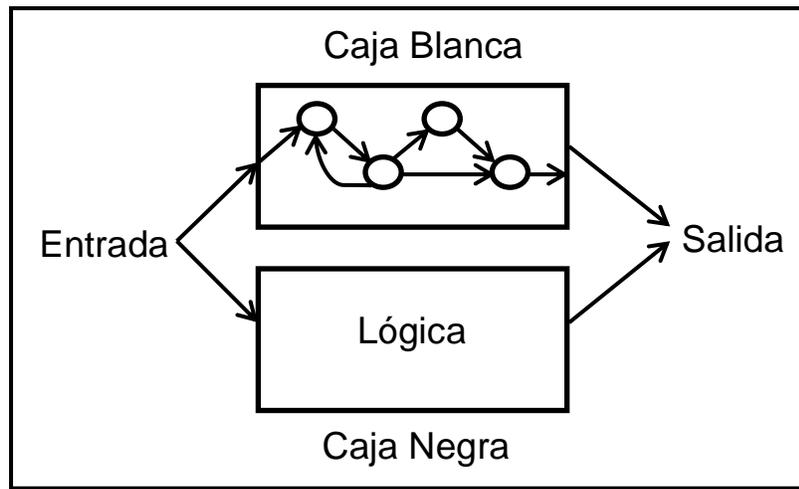


Figura 21. Diagrama de pruebas de Caja Blanca y Negra

Por último, se adopta la estrategia de prueba plasmada en [43] que permite evaluar el software empezando por los componentes más simples y seguir avanzando progresivamente hasta lograr probar completamente y en conjunto el software.

Para lograr lo anteriormente dicho se plantean los siguientes pasos:

1. Pruebas Unitarias: Prueba de cada modulo
2. Pruebas de Integración: Se prueban los módulos integrados para probar sus interfaces.
3. Prueba del Sistema: Se prueba el software ensamblado completamente incluso son necesarios sistemas externos o componentes hardware.
4. Pruebas de Aceptación: Es la última prueba realizada donde el cliente corrobora que el software cumple los requerimientos funcionales esperados.

A continuación se presentan los procesos de V&V definidos y aplicados al prototipo software.

#### 4.1.1 ANÁLISIS ESTÁTICO

Las inspecciones definidas realizadas al prototipo en los diferentes ciclos de desarrollo son:

- Diseño
- Código fuente

Durante la finalización de la implementación de las funcionalidades del prototipo planteadas en cada ciclo de desarrollo se adaptaron una serie de preguntas

basadas en [41] las cuales se aplican al caso de uso Generación de Campañas Batch con acciones tipo Merge y Split consignando los resultados en la Tabla 39 y la Tabla 40.

De igual manera dichas preguntas fueron resueltas para los ciclos restantes realizando continuamente las correcciones necesarias hasta obtener las respuestas mostradas en las tablas 39 y 40.

#### 4.1.1.1 Procesos de V&V del Ciclo de Desarrollo 2 Caso de Uso (Generación de Campañas Batch V1 – V2)

<b>LISTA DE CHEQUEO PARA LA INSPECCIÓN DEL DISEÑO</b>			
<b>#</b>	<b>PREGUNTAS</b>	<b>SI</b>	<b>NO</b>
1	¿Cubre el diseño todos los requisitos funcionales?	X	
2	¿Resulta ambigua la documentación del diseño?		X
3	¿Se ha aplicado la notación de diseño correctamente?	X	
4	¿Se han definido correctamente las interfaces entre elementos del diseño?	X	
5	¿Es el diseño suficientemente detallado como para que sea posible implementarlo en el lenguaje de programación elegido?	X	

Tabla 39. Lista de Chequeo para la inspección del Diseño

<b>LISTA DE CHEQUEO PARA LA INSPECCIÓN DEL CÓDIGO</b>			
<b>LÓGICA DEL PROGRAMA</b>			
<b>#</b>	<b>PREGUNTAS</b>	<b>SI</b>	<b>NO</b>
1	¿Es correcta la lógica del programa?	X	
2	¿Está completa la lógica del programa?	X	
<b>INTERFACES INTERNAS</b>			
<b>#</b>	<b>PREGUNTAS</b>	<b>SI</b>	<b>NO</b>
1	¿Es igual el número de parámetros recibidos por las funciones del módulo a probar al número de argumentos enviados y su orden es correcto?	X	
2	¿Los atributos de cada parámetro recibido por el módulo a probar coinciden con los atributos del argumento correspondiente?	X	
3	¿Coinciden las unidades en las que se expresan parámetros y argumentos?	X	
<b>INTERFACES EXTERNAS</b>			
<b>#</b>	<b>PREGUNTAS</b>	<b>SI</b>	<b>NO</b>
1	¿Se declaran los ficheros con todos sus atributos de forma correcta?	X	

2	¿Se abren todos los ficheros antes de usarlos?	X	
3	¿Coincide el formato del fichero con el formato especificado en la lectura?	X	
4	¿Se manejan correctamente las condiciones de fin de fichero?	X	
5	¿Se los libera de memoria?	X	
6	¿Se manejan correctamente los errores de entrada/salida?	X	
<b>VARIABLES Y DATOS</b>			
#	<b>PREGUNTAS</b>	<b>SI</b>	<b>NO</b>
1	¿Existen variables no inicializadas?		X
2	¿Se podrían producir errores de overflow o underflow en las variables y arreglos utilizados?		X
3	¿Existen comparaciones entre variables con diferentes tipos de datos o si las variables tienen diferente longitud?		X

**Tabla 40. Lista de Chequeo para la inspección del Código**

#### **4.1.2 ANÁLISIS DINÁMICO**

En el proceso de análisis dinámico del prototipo se realizó la aplicación de las pruebas de caja blanca y de caja negra especificadas anteriormente para los diferentes casos de uso en los ciclos de desarrollo definidos.

Para las pruebas de caja blanca se utilizó la técnica de cobertura de sentencias que consiste en escribir tantos casos de prueba como sean necesarios con el objetivo de que se ejecute por lo menos una vez cada sentencia del código.

Las prueba de caja blanca se realizaron sobre la marcha una vez se iban obteniendo los productos parciales ejecutables, recorriendo las sentencias de código una a una verificando los valores de las variables, tipos de datos, funciones, eventos, condicionales, bucles, detectando los fallos con ayuda del debugger o depurador de código presente en la herramienta de programación utilizada y corrigiendo estos para obtener el funcionamiento deseado. Pero estas no fueron plasmadas de manera formal pues implica un trabajo extenuante y extenso en el tiempo.

##### **4.1.2.1 PRUEBAS DE CAJA BLANCA**

A manera de ejemplo se muestra como fue aplicada una prueba de caja blanca al proceso de lectura de un archivo .xml que contiene un programa de producción editado con el esquema B2MML.

En la Figura 22 se logra evidenciar la información que contiene el programa de producción (archivo .xml) a cargar, la cual será contrastada con la información obtenida por medio de la inspección del código, específicamente el objeto que

contiene el resultado del proceso de lectura del programa de producción (ver Figura 23), verificando así que el contenido de los atributos del objeto coincidan con los campos existentes en el archivo.

Para facilidad en la observación de las figuras se definen una serie de convenciones por colores que indican el campo que es correspondiente entre un nodo del archivo xml (Figura 22) y la propiedad a la que pertenece el objeto creado (Figura 23).

Al comparar los campos siguiendo las convenciones por colores definidas, se logra evidenciar por medio de la aplicación de la prueba de caja blanca que el proceso de lectura del programa de producción es correcto ya que todos los campos coinciden de manera correcta.

The screenshot displays the XMLSpy interface with the following structure and data:

- Root Node:** `b2mml:ProductionSchedule`
  - `xmlns:b2mml`: `http://www.wbf.org/xml/B2MML-V05`
  - `xmlns:xsi`: `http://www.w3.org/2001/XMLSchema-instance`
  - `xsi:schemaLocation`: `http://www.wbf.org/xml/B2MML-V05 file:///C:/Users/Administrador/Dropbox/Trabajo%20de%20Grado/Literatura/XML/B2MML%20-%20V005%20-%20Schema/B2MML-BatchML%20v0500%20Schemas-Word-PDF/Schemas/B2MML-V05-ProductionSchedule.xsd`
  - `b2mml:ID`: `PP_0004` (Blue)
  - `b2mml:Description`: `Programa de producción para la fabricación de pintura base o blanca` (Red)
  - `b2mml:Location`
    - `b2mml:EquipmentID`: `CELL_PINTURA_BA` (Blue)
    - `b2mml:EquipmentElementLevel`: `ProcessCell` (Blue)
  - `b2mml:PublishedDate`: `2008-07-01T00:30:00` (Green)
  - `b2mml:StartTime`: `2008-07-25T16:00:00` (Green)
  - `b2mml:EndTime`: `2008-08-02T16:00:00` (Green)
  - `b2mml:ProductionRequest` (Green)
    - `b2mml:ID`: `PR_0005` (Green)
    - `b2mml:Description`: `Requerimiento de producción de pintura base en presentación tipo cuñete` (Red)
    - `b2mml:StartTime`: `2008-07-25T16:00:00` (Green)
    - `b2mml:EndTime`: `2008-08-02T16:00:00` (Green)
    - `b2mml:Priority`: `3` (Green)
    - `Comment`: `Requerimiento de Segmento Global` (Green)
    - `b2mml:SegmentRequirement` (Orange)
      - `b2mml:ID`: `SR_0021` (Orange)
      - `b2mml:ProductSegmentID`: (Orange)
      - `b2mml:ProcessSegmentID`: `PSgPntBA` (Orange)
      - `b2mml:Description`: `Fabricación de Pintura base en Unds Empacadas tipo cuñete` (Red)
      - `b2mml:EarliestStartTime`: `2008-07-25T16:00:00` (Green)
      - `b2mml:LatestEndTime`: `2008-08-02T16:00:00` (Green)
      - `b2mml:MaterialProducedRequirement` (Orange)
        - `Comment`: `Requerimiento al Segmento de Proceso de Pintura Base 1` (Green)
        - `b2mml:SegmentRequirement` (Orange)
          - `Comment`: `Requerimiento al Segmento de Proceso de Filtrado de Pintura` (Green)
          - `b2mml:SegmentRequirement` (Orange)
            - `b2mml:ID`: `SR_0023` (Orange)
            - `b2mml:ProductSegmentID`: (Orange)
            - `b2mml:ProcessSegmentID`: `PSgFilt` (Orange)
            - `b2mml:Description`: `Filtrado de Pintura` (Red)
            - `b2mml:EarliestStartTime`: `2008-07-26T08:00:00` (Green)
            - `b2mml:LatestEndTime`: `2008-07-26T19:00:00` (Green)
            - `b2mml:MaterialProducedRequirement` (Orange)
              - `b2mml:MaterialClassID`: `IGP-006` (Blue)
              - `b2mml:Description`: `Proceso de Filtrado de la pintura base` (Red)
              - `b2mml:Quantity` (Orange)
                - `b2mml:Quantity...`: `20000` (Yellow)
                - `b2mml:UnitOfM...`: `Gns` (Green)

Figura 22. Programa de Producción Cargado

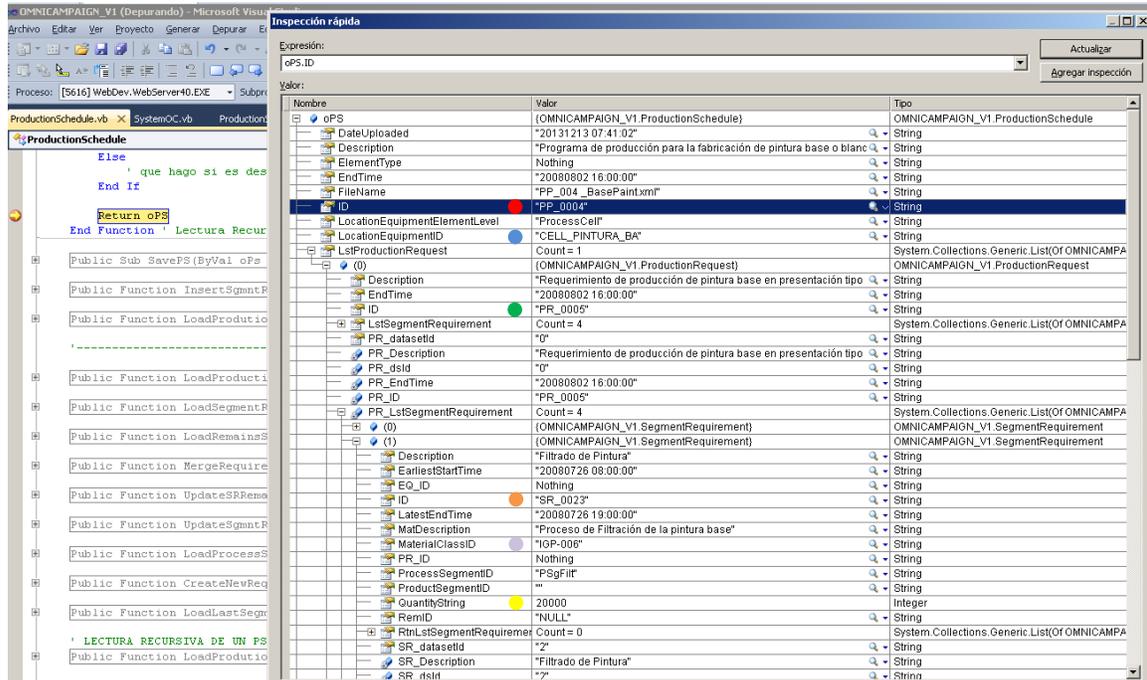


Figura 23. Prueba de Caja Blanca aplicada a la lectura del programa de producción editado bajo el esquema B2MML

#### 4.1.2.2 PRUEBAS DE CAJA NEGRA

En la aplicación de las pruebas de tipo caja negra se hace uso de la estrategia de prueba planteada anteriormente iniciando con las pruebas unitarias a cada caso de uso en los ciclos de desarrollo definidos. A continuación se muestra la aplicación de la prueba al caso de uso de generar campañas batch definido en el ciclo de desarrollo 2.

##### 4.1.2.2.1 PRUEBAS UNITARIAS

###### 4.1.2.2.1.1 DEFINICIÓN DE LA PRUEBA

- **Diseño de la Prueba:** Para la prueba software a realizar al módulo de generación de campañas batch se empleara la técnica de comparación causa – efecto que consiste en establecer una serie de entradas que desencadenaran una serie de salidas previamente conocidas, con lo cual se contrastan las salidas obtenidas con las esperadas detectando así los fallos existentes en el módulo.

- **Generación de los casos de prueba:**

Para esta parte del análisis dinámico fueron definidos 9 casos de prueba que son mostrados en su totalidad en el anexo B.

A continuación se presentan únicamente los casos 3 y 5 que son algunos de los casos de Merge y Split que pueden presentarse.

- **Caso 3:** El sistema debe programar un requerimiento de segmento cuya cantidad requerida sea menor que la capacidad máxima del equipo seleccionado provocando que el sistema haga un Merge exacto con remanentes existentes previamente. Se espera que el sistema genere una campaña batch con la cantidad de producto correcta uniendo el requerimiento de segmento a programar con los remanentes añadidos a la capacidad máxima del equipo.
- **Caso 5:** El sistema debe programar un requerimiento de segmento cuya cantidad requerida exceda la capacidad de los equipos disponibles por lo cual sea necesario generar un Split inexacto del requerimiento, escogiendo la opción de generar remanente, se espera que el sistema genere una campaña batch con un número de batches exactos y la cantidad restante la asigna a un remanente que deja disponible para la programación futura, por lo cual con esta opción no se cumple con programar la cantidad solicitada inicialmente en el requerimiento de segmento.

- **Procedimientos de la Prueba:**

La ejecución de la prueba será realizada por el equipo de desarrollo, siguiendo en orden los casos de prueba definidos y registrando en una lista de chequeo los requerimientos funcionales asociados al módulo probado.

- **Ejecución de la Prueba:**

A continuación se mostrará la ejecución de la prueba correspondiente a los casos de prueba 3 y 5.

○ **Caso 3:**

La programación del requerimiento de segmento genera una acción de Merge con el batch completo del equipo seleccionado de tal forma que se cumpla con el requerimiento y se aproveche al máximo la capacidad del equipo.

**OMNICAMPAIGN V1**

Recepción del Programa de Producción
Programación de la Producción
Despacho de la Producción
Configuración

RECEPCIÓN DEL PROGRAMA DE PRODUCCIÓN

Seleccione el Nuevo Programa de Producción que desea Cargar al Servidor

No file chosen

Lista de Programas de Producción No iniciados Cargados en el Servidor

	ID	NOMBRE DE ARCHIVO	FECHA DE CARGA
✓	PP_0002	PP_002_YelRedPaint.xml	2008-07-04 11:47:57 AM
✓	PP_0001	PP_001_RedPaint.xml	2008-07-04 10:10:57 AM

**REQUERIMIENTOS DE PRODUCCION**

**REQUERIMIENTOS DE SEGMENTO**

REQUERIMIENTOS DE SEGMENTOS ASOCIADOS AL REQUERIMIENTO DE PRODUCCIÓN SELECCIONADO

	NOMBRE DE SEGMENTO ASOCIADO	ID	DESCRIPCIÓN	FECHA INICIAL	FECHA FINALIZACIÓN	MATERIAL A PRODUCIR	DESCRIPCIÓN DE MATERIAL	CANTIDAD	UNIDAD DE MEDIDA
✓	PINTURA BASE NIVEL 2	SR_0012	Fabricación de Pintura base	2008-07-11 4:00:00 PM	2008-07-12 3:00:00 AM	IGP-001	Fabricación del producto intermedio de Pintura base o blanca para la posterior adición de pigmentos	9000	Glns
✓	PINTURA COLOR NIVEL 2	SR_0013	Fabricación de Pintura a Color	2008-07-12 3:00:00 AM	2008-07-12 8:00:00 AM	IGP-003	Fabricación del producto intermedio de Pintura color Amarillo	800	Glns
✓	PINTURA FILTRADA NIVEL 2	SR_0014	Filtrado de Pintura	2008-07-12 8:00:00 AM	2008-07-12 1:00:00 PM	IGP-007	Proceso de Filtración de la pintura color amrilla	9000	Glns

**REMANENTES DE PRODUCCIONES PASADAS**

**Figura 24. Información del requerimiento de segmento seleccionado para la acción de Merge**

En la Figura 24 se evidencia la selección del requerimiento de segmento a programar, que en este caso es un requerimiento de segmento para la fabricación de pintura a color amarilla con id de producto (IGP-003) cuya cantidad es de 800 galones. Dicha cantidad es menor que la capacidad disponible de los equipos configurados para el segmento de proceso de pintura a color (UNIT\_PC1: 2.000 Glns., UNIT\_PC2:1.500 Glns.), por lo cual al ser menor es posible añadirle más requerimientos con el fin de completar la capacidad máxima del equipo.

En la Figura 25 mostrada a continuación se evidencian remanentes que surgen de producciones pasadas y que comparten el mismo tipo de producto (IGP-003) con el requerimiento de segmento seleccionado, posibilitando la acción de Merge.

**OMNICAMPAIGN V1**

Recepción del Programa de Producción    Programación de la Producción    Despacho de la Producción    Configuración

RECEPCIÓN DEL PROGRAMA DE PRODUCCIÓN

Seleccione el Nuevo Programa de Producción que desea Cargar al Servidor

No file chosen

Lista de Programas de Producción No iniciados Cargados en el Servidor

ID	NOMBRE DE ARCHIVO	FECHA DE CARGA
<input checked="" type="checkbox"/>	PP_0002_PP_002_YelRedPaint.xml	2008-07-04 11:47:57 AM
<input checked="" type="checkbox"/>	PP_0001_PP_001_RedPaint.xml	2008-07-04 10:10:57 AM

**REQUERIMIENTOS DE PRODUCCION**

**REQUERIMIENTOS DE SEGMENTO**

**REMANENTES DE PRODUCCIONES PASADAS**

ID	DESCRIPCIÓN	FECHA INICIAL	FECHA FINALIZACIÓN	MATERIAL A PRODUCIR	DESCRIPCIÓN DE MATERIAL	CANTIDAD	UNIDAD DE MEDIDA
<input checked="" type="checkbox"/>	SR_0008	2008-07-06 1:00:00 PM	2008-07-07 1:00:00 AM	IGP-003	Fabricación del producto intermedio de Pintura color Amarillo	200	Glns
<input checked="" type="checkbox"/>	SR_0009	2008-07-06 1:00:00 PM	2008-07-07 1:00:00 AM	IGP-003	Fabricación del producto intermedio de Pintura color Amarillo	250	Glns
<input checked="" type="checkbox"/>	SR_0010	2008-07-06 1:00:00 PM	2008-07-07 1:00:00 AM	IGP-003	Fabricación del producto intermedio de Pintura color Amarillo	250	Glns

**Figura 25. Lista de remanentes del mismo ID para la unión de requerimientos**

Después de la selección del requerimiento de segmento se elige el segmento de proceso de pintura a color (Nivel 3) existente, el cual tiene asociadas dos unidades de producción (UNIT\_PC1, UNIT\_PC2) que serán las posibles opciones para que el sistema escoja sobre cuál de estas programa el requerimiento.

Por último, se evidencia en la Figura 26 el resultado de la generación de campañas batch en la acción Merge al unir el requerimiento de segmento inicial seleccionado que tenía una capacidad requerida de 800 Glns con los 3 remanentes existentes de capacidades 200, 250 y 250 Glns, generando



UNIT\_FIL3: 2.000 Glns.), por lo cual al ser mayor es posible dividir la orden por medio de una acción tipo split.

**OMNICAMPAIGN V1**

Recepción del Programa de Producción
Programación de la Producción
Despacho de la Producción
Configuración

**RECEPCIÓN DEL PROGRAMA DE PRODUCCIÓN**

Seleccione el Nuevo Programa de Producción que desea Cargar al Servidor

No file chosen

Lista de Programas de Producción No iniciados Cargados en el Servidor

	ID	NOMBRE DE ARCHIVO	FECHA DE CARGA
✓	PP_0002	PP_002_YelRedPaint.xml	2008-07-04 11:47:57 AM
✓	PP_0001	PP_001_RedPaint.xml	2008-07-04 10:10:57 AM

**REQUERIMIENTOS DE PRODUCCION**

**REQUERIMIENTOS DE SEGMENTO**

REQUERIMIENTOS DE SEGMENTOS ASOCIADOS AL REQUERIMIENTO DE PRODUCCIÓN SELECCIONADO

	NOMBRE DE SEGMENTO ASOCIADO	ID	DESCRIPCIÓN	FECHA INICIAL	FECHA FINALIZACIÓN	MATERIAL A PRODUCIR	DESCRIPCIÓN DE MATERIAL	CANTIDAD	UNIDAD DE MEDIDA
✓	PINTURA BASE NIVEL 2	SR_0012	Fabricación de Pintura base	2008-07-11 4:00:00 PM	2008-07-12 3:00:00 AM	IGP-001	Fabricación del producto intermedio de Pintura base o blanca para la posterior adición de pigmentos	9000	Glns
✓	PINTURA FILTRADA NIVEL 2	SR_0014	Filtrado de Pintura	2008-07-12 8:00:00 AM	2008-07-12 1:00:00 PM	IGP-007	Proceso de Filtración de la pintura color amarilla	9120	Glns

**Figura 27. Información del Requerimiento de Segmento seleccionado para la acción de Split – Crear Remanente**

Al confirmar la programación del requerimiento seleccionado el sistema analiza la acción a realizar y determina que debe ser un Split inexacto por lo cual brinda la posibilidad de escoger entre 3 opciones:

- **Sobreproducción:** Consiste en exceder la cantidad requerida de la campaña para aprovechar al máximo la capacidad del equipo.
- **Crear reamente:** Consiste en dejar de producir una cantidad determinada de material que no completa la capacidad máxima del equipo sobre el cual se programó el requerimiento de segmento.
- **Cumplir con la orden:** Consiste en cumplir la capacidad requerida por la orden sin aprovechar al máximo la capacidad del equipo.



En la Figura 29 mostrada anteriormente se observa la campaña batch generada por el sistema con la acción de Split – Crear Remanente, la unidad seleccionada UNIT\_FIL1 tiene una capacidad máxima de 600 Glns por batch, por lo tanto al programar el requerimiento de segmento seleccionado inicialmente que tiene una cantidad requerida de 9.120 Glns se generarían 16 batches de los cuales 15 serían a la máxima capacidad del equipo y el batch restante solo con 120 Glns, por lo cual el operario decide programar la campaña con los 9.000 Glns y generar un remanente con los 120 Glns faltantes.

A continuación en la Figura 30 se evidencia la creación del nuevo requerimiento SR\_0021 (remanente) asociado al requerimiento ya programado SR\_0014 con el id de producto adecuado (IGP-007), el cual tiene como cantidad requerida los 120 Glns que no se programaron.

	ID REMANENTE	ID DEL REQUERIMIENTO PADRE	PRODUCTO	DESCRIPCION	CANTIDAD	UNDS
1	SR_0008	SR_0005	IGP-003	Fabricación de Pintura a Color	200	Glns
2	SR_0009	SR_0006	IGP-003	Fabricación de Pintura a Color	250	Glns
3	SR_0010	SR_0007	IGP-003	Fabricación de Pintura a Color	250	Glns
4	SR_0021	SR_0014	IGP-007	Remanente de Filtrado de Pintura	120	Glns

**Figura 30. Resultado de solicitar los remanentes a la base de datos de la aplicación**

- **Informe de Prueba:**

Por último teniendo en cuenta los 9 casos de prueba desarrollados que se pueden observar completamente en el anexo B, para el caso de uso de generación de campañas batch se evalúa el cumplimiento de los requerimientos funcionales asociados a éste por medio de la lista de chequeo mostrada en la Tabla 41 a continuación.

<b>Número de Ref.</b>	<b>Requerimiento</b>	<b>Cumple (Si - No)</b>
<b>R1.4</b>	El sistema deberá implementar y garantizar la integridad transaccional y de datos.	Si
<b>R1.5</b>	Almacenar información relevante del proceso	Si
<b>R1.7</b>	Realizar acciones tipo Split & Merge.	Si
<b>R1.8</b>	La solución deberá ser desarrollada con las tecnologías de Microsoft, Visual Studio.Net, ASP.Net y SQL Server.	Si
<b>R2.1</b>	Crear y mantener un programa detallado de la producción.	Si
<b>R2.2</b>	Determinar la capacidad comprometida de cada recurso para el uso de la función de administración de los recursos de producción.	Si

**Tabla 41. Requerimientos funcionales que cumple el caso de uso Generar Campañas batch**

Una vez terminada la ejecución de todos los casos de prueba definidos y de la corrección de los fallos encontrados se aplica el mismo proceso a los casos de uso restantes en los otros ciclos de desarrollo. Al final se obtiene una serie de listas de chequeo que contienen todos los requerimientos funcionales y estructurales planteados para el prototipo cumplidos satisfactoriamente, así como el sistema completamente probado con todos sus módulos integrados.

Es necesario resaltar que al plantear de una manera incremental la estrategia de prueba utilizada para verificar y validar el prototipo al finalizar las pruebas unitarias es posible concluir que se realizaron paulatinamente las pruebas de integración y de sistema ya que a medida que se añadía un módulo nuevo a los procesos de verificación y validación se mantenían los anteriores logrando así probar la integración entre módulos y toda la aplicación en conjunto.

## CONCLUSIONES

- Se modeló, diseñó e implementó un prototipo de gestión de campañas batch cuya especificación se basa en los estándares ISA-88.01, ISA-95.02 e ISA-95.03 y que además se integra verticalmente recibiendo e interpretando el programa de producción y generando la lista de despacho de batches (*Dispatch List*) bajo los esquemas de integración B2MML y BATCHML respectivamente definidos por el WBF.
- Se logró la integración entre el prototipo y un administrador batch comercial (Factory Talk Batch), logrando una comunicación de doble vía para el envío y recepción de información.
- Se obtuvo un prototipo que permite la programación de campañas batch con tamaño de batches variable, contribuyendo al aprovechamiento de la capacidad de los equipos disponibles presentes en la planta al igual que a la disminución en pérdida de tiempo por los recambios de equipos.
- Para la aplicación del prototipo a un proceso batch es importante tener una separación lógica de los 3 niveles de segmentos de proceso (lo cual se puede realizar aplicando el modelado que proporciona el estándar ISA-95) e información de las unidades de producción y el manejo de récipes para lo cual se hace necesario modelar el proceso bajo las especificaciones presentadas en el estándar ISA-88.01.
- Se desarrolló el prototipo teniendo en cuenta buenas prácticas de desarrollo por medio de la aplicación de patrones de diseño brindando características como la flexibilidad al cambio, mantenibilidad y escalabilidad.
- Para enfrentar el desarrollo de cualquier proyecto es de gran importancia el uso de una metodología de desarrollo que sirva como guía para obtener resultados de manera eficaz, ordenada y con calidad.
- En el desarrollo de un proyecto software adoptar una estructura de modelado en los inicios del proyecto es necesario, ya que con el modelado se analizan y clarifican muchos aspectos antes de iniciar el desarrollo permitiendo así dar una base sólida para la implementación del mismo.

- La aplicación de patrones de diseño en el proceso de análisis y desarrollo de software es una ayuda para asegurar resultados acertados de manera más eficaz y fiable pues son soluciones ya probadas a problemas comunes que facilitan el análisis y codificación del proyecto.
- Tener en cuenta cada uno de los posibles casos que se puedan presentar en la industria de procesos Batch sin considerar alguna limitación para el desarrollo de las funcionalidades y expectativas del prototipo, implica un trabajo bastante arduo y extenso en el tiempo, por lo cual es necesario delimitar el alcance del proyecto.
- Considerar un segmento de proceso de tercer nivel dentro de la topología de proceso definida para el desarrollo del proyecto permite brindarle más flexibilidad en la implementación de éste para empresas con procesos productivos batch mucho más grandes o complejos.
- Una adecuada programación detallada de la producción con acciones Merge y Split y un apropiado despacho de la producción en una empresa que opera en modo campañas batch favorece el aumento de la productividad, disminución de tiempos de entrega, aprovechamiento de la capacidad de los equipos y disminución de reconfiguraciones de los equipos por cambio de tipo de producto.
- Realizar una continua verificación y validación de las funcionalidades de un proyecto a lo largo de su desarrollo permite detectar fallas, omisiones o malas interpretaciones de los requerimientos a tiempo permitiendo tomar las acciones correctivas en el momento oportuno facilitando así la obtención de un producto de calidad, que satisfaga las necesidades de los clientes.

## BIBLIOGRAFÍA

- [1] International Society of Automation, «Enterprise Control System Integration Part 3: Activity Models of Manufacturing Operations Management,» American National Standard, North Carolina, 2005.
- [2] S. Benjaafar y M. Sheikhzadeh, «Scheduling Policies, Batch Sizes, and Manufacturing Lead Times,» *Department of Mechanical Engineering, University of Minnesota, Minneapolis, Minnesota*, 1996.
- [3] R. Guðmundsdóttir, PRODUCTION SCHEDULING IN A CAMPAIGN BASED FLEXIBLE FLOW SHOP, Reykjavík , Islandia: School of Science and Engineering Reykjavík University, 2012.
- [4] C. A. Méndez, J. Cerdá, I. E. Grossmann, I. Harjunkoski y M. Fahl, «State-of-the-art review of optimization methods for short-term scheduling of batch processes,» *Computers & Chemical Engineering* , vol. 30, pp. 913-946, 2006.
- [5] C. Larman, UML y Patrones. Introduccion al analisis y diseño orientado a objetos, México: Pearson, 2002.
- [6] WBF (Word Batch Forum), «Business To Manufacturing Markup Language (B2MML),» WBF, [En línea]. Available: [http://www.isa.org/Content/NavigationMenu/General\\_Information/Partners\\_and\\_Affiliates/WBF/Working\\_Groups2/XML\\_Working\\_Group/B2MML/B2MML.htm](http://www.isa.org/Content/NavigationMenu/General_Information/Partners_and_Affiliates/WBF/Working_Groups2/XML_Working_Group/B2MML/B2MML.htm). [Último acceso: 13 August 2013].
- [7] WBF (Word Batch Forum), «Batch Markup Language - BatchML-V05-BatchInformation,» 2011. [En línea]. Available: [http://www.isa.org/Content/NavigationMenu/General\\_Information/Partners\\_and\\_Affiliates/WBF/Working\\_Groups2/XML\\_Working\\_Group/BatchML/BatchML.htm](http://www.isa.org/Content/NavigationMenu/General_Information/Partners_and_Affiliates/WBF/Working_Groups2/XML_Working_Group/BatchML/BatchML.htm). [Último acceso: 13 August 2013].
- [8] I. A. K. Naresh Susarla, «Integrated Campaign Planning and Resource Allocation in Batch Plants,» *European Symposium on Computer Aided Process Engineering – ESCAPE20* , vol. 20th, 2010.
- [9] G. Lina, Á. Hernán y B. Héctor, «Limitaciones de la Controlabilidad de Estados en los Procesos por Lotes,» *Información Tecnológica* , vol. 23, pp. 97-108, 2012.

- [10] B. Dominique, «Optimal Operation of Batch Reactors - a personal view,» *Journal of Process Control*, vol. 8, pp. 335-368, 1998.
- [11] Universidad de los Andes Venezuela, «La Web del Profesor,» [En línea]. Available: [http://webdelprofesor.ula.ve/economia/oliverosm/materiasdictadas/produccion2/clases/planificacion\\_de\\_la\\_produccion\\_teoría.pdf](http://webdelprofesor.ula.ve/economia/oliverosm/materiasdictadas/produccion2/clases/planificacion_de_la_produccion_teoría.pdf). [Último acceso: 15 July 2013].
- [12] J. DOMÍNGUEZ, M. ÁLVAREZ, S. GARCÍA, M. DOMÍNGUEZ y A. RUIZ, Dirección de Operaciones: Aspectos Tácticos y Operativos en la Producción y los Servicios., Madrid.: McGraw Hill. , 1995.
- [13] S. N. CHAPMAN, PLANIFICACIÓN Y CONTROL DE LA PRODUCCIÓN, Atlacomulco, Mexico: PRENTICE- HALL INC., 2006.
- [14] Crawford y Wiers, «From anecdotes to theory: reviewing the knowledge of the human factors in planning and scheduling,» *En MacCarthy BL, Wilson JR (Ed.). Human performance in planning and scheduling: fieldwork studies, methodologies and research issues*, p. 15–44, 2001.
- [15] M. Vázquez C, «Diseño en Ingeniería Mecánica con talento latino - Automatización un Dilema de Convivencia,» [En línea]. Available: <http://mx.groups.yahoo.com/neo/groups/DesignMecanico3D/conversations/topics/2044>. [Último acceso: 10 August 2013].
- [16] K. N. McKay y V. C. Wiers, «Planning, scheduling and dispatching task in production control,» *Cogn Tech Work*, 2003.
- [17] C. Lioun Wee, K. S. Zafar y R. Salisbury, «Scheduling in Batch Process Plant Perspective,» *ISA Technical Section Library*, 1999.
- [18] C. N. Potts y M. Y. Kovalyov, «Scheduling with batching: A review,» *European Journal of Operational Research*, vol. 120, pp. 228-249, 2000.
- [19] International Society of Automation, «Batch Control Part 1: Models and Terminology,» American National Standar, North Carolina, 1995.
- [20] C. GEAMBAȘU, I. JIANU, I. JIANU y A. GAVRILĂ, «INFLUENCE FACTORS FOR THE CHOICE OF A SOFTWARE DEVELOPMENT METHODOLOGY,» *Accounting and Management Information Systems*, vol. 10, nº 4, pp. 479-494, 2011.
- [21] IBM, «IBM Rational Unified Process (RUP) Proven best practices for software and

- systems delivery,» [En línea]. Available: <http://www-01.ibm.com/software/rational/rup/>. [Último acceso: 29 September 2013].
- [22] V. Quispe, D. Huamantuco y J. L. Vargas, «METODOLOGIA RUP (RATIONAL UNIFIED PROCESS),» *UNIVERSIDAD NACIONAL DEL ALTIPLANO*, 2011.
- [23] IBM Software Group, «Rational Unified Process: A Best Practices Approach,» *Rational Software*, 2003.
- [24] D. Wells, «Extreme Programming: A gentle introduction,» [En línea]. Available: <http://www.extremeprogramming.org>. [Último acceso: 26 September 2013].
- [25] R. E. Jeffries, «XProgramming.com An Agile Software Development Resource,» [En línea]. Available: <http://xprogramming.com/what-is-extreme-programming/>. [Último acceso: 26 September 2013].
- [26] Scrum Alliance, «Scrum Alliance,» [En línea]. Available: <http://www.scrumalliance.org/why-scrum>. [Último acceso: 26 September 2013].
- [27] P. S. a. Systems, «Modelado de Sistemas com UML».
- [28] Rockwell Automation, «Factory Talk Batch - Server API Communication Language Technical Reference,» 2007.
- [29] A. Vignaga y D. Perovich, «Arquitecturas y tecnologías para el desarrollo de aplicaciones web,» Universidad de la República - Facultad de Ingeniería - Instituto de Computación, Montevideo.
- [30] M. Papagelis, «Web App Architectures: Multi Tier (2 Tier, 3 Tier) & MVC,» [En línea]. Available: <http://queens.db.toronto.edu/~papaggel/courses/csc309/docs/lectures/web-architectures.pdf>. [Último acceso: 20 September 2013].
- [31] L. Welicki, «MSDN - Patrones y Antipatrones: una Introducción - Parte II,» [En línea]. Available: <http://msdn.microsoft.com/es-es/library/bb972251.aspx>. [Último acceso: 11 November 2013].
- [32] M. p. & practices, «Microsoft Developer Network (MSDN) Model-View-Controller,» Microsoft, 2013. [En línea]. Available: <http://msdn.microsoft.com/en-us/library/ff649643.aspx>. [Último acceso: 09 November 2013].
- [33] dofactory, «.NET Design Patterns,» [En línea]. Available:

- <http://www.dofactory.com/Patterns/Patterns.aspx>. [Último acceso: 11 November 2013].
- [34] J. Pavón, «Patrones de diseño orientado a objetos,» [En línea]. Available: <http://www.fdi.ucm.es/profesor/jpavon/poo/2.14PDOO.pdf>. [Último acceso: 10 November 2013].
- [35] International Society Automation , «Enterprise Control System Integration Part 2: Object Model Attributes,» AMERICAN NATIONAL STANDARD, North Carolina, 2001.
- [36] A. López, A. Ortega, J. Rengifo y R. Solarte, «Modelado ISA-88 para el proceso de obtención de pintura a base de agua,» Popayán , 2011.
- [37] F. Vidal y S. Muñoz, «Aplicacion de la norma ISA S95 a un caso de estudio,» Tesis de Pregrado no publicada, Universidad del Cauca, Popayán, Colombia, 2007.
- [38] J. M. Drake y P. López, «Computadores y Tiempo Real,» 2009. [En línea]. Available: [http://www.ctr.unican.es/asignaturas/Ingenieria\\_Software\\_4\\_F/Doc/M7\\_09\\_Verificacio nValidacion-2011.pdf](http://www.ctr.unican.es/asignaturas/Ingenieria_Software_4_F/Doc/M7_09_Verificacio nValidacion-2011.pdf). [Último acceso: 18 November 2013].
- [39] IEEE Computer Society, IEEE, Std. 610 Computer dictionary, 1990.
- [40] I. Sommerville, «Software Engineering,» 2004. [En línea]. Available: <http://ifs.host.cs.st-andrews.ac.uk/Books/SE7/Presentations/PDF/ch22.pdf>. [Último acceso: 18 November 2013].
- [41] Universidad de Sevilla, «Departamento de Lenguajes y Sistemas Informáticos - Técnicas de Evaluación Estática,» [En línea]. Available: <http://www.lsi.us.es/docencia/get.php?id=360>. [Último acceso: 10 12 2013].
- [42] F. Alzate, «Cómo elaborar listas de chequeo,» Centro de Gestión empresarial , 9 May 2012. [En línea]. Available: <http://cgeconsultoria.com/como-elaborar-listas-de-chequeo/>. [Último acceso: 18 November 2013].
- [43] Universidad de Sevilla, «Departamento de Lenguajes y Sistemas Informáticos - Técnicas de Evaluación Dinámica,» [En línea]. Available: <http://www.lsi.us.es/docencia/get.php?id=360>. [Último acceso: 11 12 2013].