



Software Plus



Monica Alvarez
Juan Camilo Chafloque
Sebastián Gutiérrez
Laura Jimenez
Santiago Palacios
Sergio Posada

Smart Closet

Software Project Management Plan

Versión 2.3

Ingeniería de Software

Pontificia Universidad Javeriana
Facultad de Ingeniería
Departamento de Ingeniería de Sistemas
2020-1

1. Historial de Cambios

Versión	Fecha de Cambio	Descripción	Sección o Artefacto Afectado	Responsable
0.1	11/02/2020	Creación del documento y Creación de Portada	Portada	Juan Camilo Chafloque
0.2	17/02/2020	Creación lista de contenido, figuras y tablas	3,4	Juan Camilo Chafloque
0.3	17/02/2020	Se realiza la introducción	6	Juan Camilo Chafloque, Sergio Posada
0.4	17/02/2020	Se realiza la visión del producto y la sección de evolución del plan	7.1, 7.4	Sebastián Gutierrez
0.5	17/02/2020	Se realizó el propósito, alcance y objetivos	7.2	Santiago Palacios
0.6	17/02/2020	Se definieron y plasmaron los supuestos y restricciones	7.3	Santiago Palacios, Sergio Posada
0.7	18/02/2020	Se realiza la sección de lenguajes y herramientas	8.2	Juan Camilo Chafloque
0.8	18/02/2020	Se realiza la sección de entrega del producto	11	Juan Camilo Chafloque
0.9	18/02/2020	Se realiza sección de plan de aceptación del producto	8.3	Sebastián Gutiérrez
1.0	25/02/2020	Se realiza la sección de herramientas y métodos de estimación.	9.1	Monica Alvarez
1.1	25/02/2020	Se realiza a sección de Administración de requisitos y Monitoreo y control del progreso	10.1, 10.2	Sergio Posada

1.2	25/02/2020	Se hizo la sección de inicio de proyecto.	9.2	Sebastian Gutierrez
1.3	25/02/2020	Se realiza la sección de modelo de ciclo de vida del proyecto.	8.1	Laura Jimenez
1.4	25/02/2020	Se realiza la sección de organización del proyecto y comunicación	8.4	Santiago Palacios
1.5	29/02/2020	Se realiza la sección de planes de trabajo del proyecto	9.3	Monica Alvarez y Sebastian Gutierrez
1.6	29/02/2020	Se realiza la sección de cierre de proyecto	10.3	Juan Camilo Chafloque
1.7	29/02/2020	Se realiza la sección de análisis y control de riesgos	12.2	Laura Jimenez
1.8	29/02/2020	Se realiza la sección de control de calidad	12.4	Sergio posada
1.9	29/02/2020	Se realiza la sección de ambiente de trabajo	12.1	Juan Camilo Chafloque, Sergio Posada
2.0	01/03/2020	Se realiza la sección de administración de configuración y documentación	12.3	Juan Camilo Chafloque
2.1	01/03/2020	Se termina de completar el glosario con las palabras desconocidas del documento	7.5	Equipo de Software Plus
2.2	01/03/2020	Se terminan y enumeran los anexos del documento		Equipo de Software Plus
2.3	01/03/2020	Se terminan de completar las referencias del documento	13	Equipo de Software Plus

Tabla 1. Historial de cambios

2. Resumen Ejecutivo

En este documento se busca plasmar, detallar y definir el Plan de Gestión de Desarrollo de Software (Software Project Management Plan) mejor conocido por sus siglas SPMP, y asimismo los procesos y tareas involucradas en el desarrollo del software Smart Closet. Dicho software busca suplir las necesidades de los clientes a la hora de elegir un atuendo basado en diferentes criterios y reglas de moda, condiciones de clima, combinaciones y la variedad de prendas que el usuario posea dentro de su closet. Se pretende ofrecer un servicio práctico, amigable con el usuario y sencillo de usar, que permita ofrecer alternativas y soluciones vertiginosas a la hora de toparse con disyuntivas triviales de vestimenta.

Dado que se parte de un documento SPMP, se pretende visualizar las políticas, procedimientos, horarios y recursos en la elaboración del proyecto. En ese sentido, se busca dar a conocer los aspectos más relevantes de la planeación y gestión del software Smart Closet. Para esto, el SPMP establece una visión general del proyecto, en donde se analiza el alcance, los objetivos y el propósito del mismo; enmarcados en un conjunto de supuestos y restricciones, esto con el fin de tener claro qué es lo que se quiere lograr y conseguir al final del proyecto. Cabe aclarar que este documento ha sido desarrollado según el estándar 16326:2009 de la IEEE [1].

En el documento también se encuentran establecidos los planes necesarios que se van a necesitar para que el desarrollo del proyecto sea exitoso y se entregue un producto de calidad. Estos planes están asociados a la escogencia del ciclo de vida del software, las herramientas que se implementarán para la comunicación en equipo, artefactos asociados, métodos y medidas de control y estimación, además de los procesos de soporte.

3. Lista de Figuras

Ilustración 1. Estructura del modelo incremental [10].....	9
Ilustración 2. Organigrama.....	17
Ilustración 3. Diagrama BPMN de la administración de cambios de requerimientos	28
Ilustración 4. Diagrama BPMN para el plan de análisis y gestión de riesgos.....	37
Ilustración 5. Diagrama BPMN para proceso de cambio de un ítem de configuración	39
Ilustración 6. Diagrama BPMN del proceso de revisión de calidad en documentos	42

4. Lista de Tablas

Tabla 1. Historial de cambios	II
Tabla 2. Glosario	6
Tabla 3. Lenguajes de planeación.....	10
Tabla 4. Lenguajes de implementación	10

Tabla 5. Herramientas	12
Tabla 6. Entregables	13
Tabla 7. Criterios de aceptación	16
Tabla 8. Entidades externas y stakeholders	17
Tabla 9. Ordenadores del proyecto	23
Tabla 10. Flujo de caja	25
Tabla 11. Acciones disciplinarias	31
Tabla 12. Entregables Finales.....	32
Tabla 13. Riesgos del proyecto.....	35
Tabla 14. Clasificación de riesgos	36
Tabla 15. Ítems de configuración	38
Tabla 16. Roles en el control de calidad.....	41

5. Tabla de Contenido

6.	Introducción	1
7.	Vista General del Proyecto	1
7.1.	Visión del Producto	1
7.2.	Propósito, alcance y objetivos	2
7.2.1.	Propósito	2
7.2.2.	Alcance	2
7.2.3.	Objetivos	2
7.2.3.1.	Objetivo General	2
7.2.3.2.	Objetivos Específicos	3
7.3.	Supuestos y Restricciones	3
7.3.1.	Supuestos	3
7.3.2.	Restricciones	3
7.4.	Evolución del Plan	3
7.4.1.	Objetivos	4
7.4.2.	Involucrados	4
7.4.3.	Desarrollo de evolución del plan	4
7.5.	Glosario	5
8.	Contexto del Proyecto	6
8.1.	Modelo de Ciclo de Vida	6
8.1.1.	Ciclo de vida incremental	7
8.1.2.	Ciclo de vida Scrum	7
8.1.3.	Extreme Programming XP	8
8.2.	Lenguajes y Herramientas	9
8.3.	Plan de Aceptación del Producto	12
8.4.	Organización del Proyecto y Comunicación	16
8.4.1.	Organigrama y roles	17
9.	Administración del Proyecto	18
9.1.	Métodos y Herramientas de Estimación	18
9.2.	Inicio del Proyecto	20
9.2.1.	Plan de capacitación	20
9.2.1.1.	Objetivos	20

9.2.1.2.	Actores	20
9.2.1.3.	Desarrollo del plan.....	20
9.2.2.	Infraestructura	21
9.2.2.1.	Objetivos	21
9.2.2.2.	Actores	21
9.2.2.3.	Tareas de instalación	21
9.2.2.4.	Mantenimiento.....	22
9.2.2.5.	Herramientas.....	22
9.2.2.6.	Hardware.....	22
9.2.2.7.	Espacios físicos	23
9.3.	Planes de Trabajo del Proyecto	23
9.3.1.	Responsables	24
9.3.2.	Descomposición de las actividades.....	24
9.3.3.	Calendarización	24
9.3.4.	Presupuesto	25
10.	Monitoreo y Control del Proyecto.....	25
10.1.	Administración de requerimientos.....	26
10.1.1.	Planeación de administración de requerimientos.....	26
10.1.2.	Administración de cambios de requerimientos	26
10.2.	Monitoreo y Control del Progreso	28
10.2.1.	Actividades de monitoreo	29
10.2.2.	Actividades para reportar el progreso	30
10.2.3.	Acciones disciplinarias.....	30
10.3.	Cierre del Proyecto.....	31
11.	Entrega del Producto	31
12.	Procesos de Soporte.....	32
12.1.	Ambiente de Trabajo.....	32
12.1.1.	Reuniones y comunicación	33
12.1.2.	Faltas graves.....	33
12.2.	Análisis y Administración de Riesgos.....	34
12.2.1.	Objetivos	34
12.2.2.	Responsables.....	34
12.2.3.	Actividades	34

12.3.	Administración de Configuración y Documentación	37
12.3.1.	Involucrados	38
12.3.2.	Ítems de configuración.....	38
12.4.	Control de Calidad.....	40
12.4.1.	Roles en el control de calidad	40
12.4.2.	Calidad del documento	41
12.4.3.	Pruebas del software	42
13.	Referencias	44

6. Introducción

El presente documento busca dar a conocer el plan de administración del proyecto de software (SPMP) de la empresa Software Plus para el desarrollo de un producto. EL SPMP es importante ya que asegura que el desarrollo de todo el proyecto encaje con lo establecido entre el cliente y el equipo, por lo que el propósito principal del SPMP es establecer los conceptos necesarios para definir un marco dentro del cual se pueda trabajar de manera óptima. Por lo tanto, este documento enuncia todo lo relacionado con el proceso de planificación, organización, estándares, herramientas a utilizar, manejo y asignación de recursos y personal, monitoreo y control.

Este documento también muestra el establecimiento de los planes, estrategias y toma de decisiones que tendrá que realizar el equipo de trabajo en cada etapa del proyecto para que el producto final tenga una ejecución exitosa con estándares de calidad altos y que cumplan con los requerimientos establecidos.

El software a desarrollar, el cual tomará el nombre de Smart Closet, busca facilitar a los usuarios que deseen hacer uso de este una forma innovadora para decidir qué tipo de prendas usarán en un día determinado según sus preferencias (deportivo, casual, elegante, etc.) Por otra parte, además de brindar un atuendo con los estándares establecidos por el usuario, el sistema tendrá en cuenta qué tipo de condiciones climáticas se van a dar en un día, y con base en esto, Smart Closet brindará la mejor opción para el usuario.

7. Vista General del Proyecto

En esta sección se destacarán a groso modo los aspectos más relevantes del proyecto, teniendo en cuenta una visión general de la propuesta de Smart Closet, atravesando las consideraciones, reglas, qué se tiene en cuenta y qué no se tiene en cuenta. Con el fin de describir el propósito, el alcance, y los objetivos del proyecto, limitándolos bajo restricciones y supuestos.

7.1. Visión del Producto

Smart Closet pretende ser un software web para aquellos usuarios que tengan acceso a internet y quieran expandir sus posibilidades de combinaciones de atuendos a partir de la variedad de prendas que se posean. Busca ser un software sencillo, práctico y bastante amigable con el usuario con el fin de ofrecer los mejores atuendos a partir de criterios de moda, historial de usos, disponibilidad de prendas, nivel de formalidad, se busca ofrecer recomendaciones acertadas a partir de condiciones climáticas y días particulares. Asimismo, el usuario puede seleccionar aquellos criterios que busca en el atuendo (colores específicos, rayas, puntos, tonos claros u oscuros, etc.) con el fin de recibir alternativas que más se acomoden a los gustos de éste, de manera rápida y efectiva.

El usuario puede administrar su closet de la manera que guste, básicamente simulando su closet físico en uno virtual, categorizando sus prendas en camisetas, sacos, chaquetas, pantalones, zapatos, calcetines, faldas, etc., como también especificando qué prendas son las que se adecúan mejor a cierto tiempo atmosférico, con opción de escoger atuendos favoritos y prendas ya usadas. Continuando, se puede hacer una gestión sencilla de las prendas, añadiendo nuevas prendas, nuevas colecciones, eliminar prendas, y consultar las prendas como guste el usuario.

El producto busca tener otras funcionalidades adicionales, que pueden generar mayor gusto al usuario. Poder compartir los atuendos a través de WhatsApp, alternar de usuarios, abrir el armario virtual en cualquier parte que se tenga acceso a internet y así gestionarlo fuera de casa. En conclusión, se busca tener un producto confiable, práctico y sencillo donde el usuario pueda gestionar sus atuendos y tener sugerencias cada día.

7.2. Propósito, alcance y objetivos

7.2.1. Propósito

El objetivo del proyecto es desarrollar un software bajo el nombre de “Smart Closet” que permita a sus usuarios la creación de un armario virtual con el fin de que, diariamente y de acuerdo a una pequeña entrada de información, el software le ofrezca al usuario una serie de atuendos para elegir y vestir en el mismo día. Todo el desarrollo del software estará bien documentado y fundamentado.

7.2.2. Alcance

El alcance del proyecto cubre principalmente la creación del armario virtual, la recepción de datos diarios, como clima y necesidad de uso para la recomendación de los atuendos, el ingreso, actualización, consulta del estado y eliminación de las prendas del closet virtual. También lleva un registro de los atuendos usados recientemente, para la consulta del sistema y del usuario. El software no genera atuendos sin información del usuario y tampoco genera modelos detallados del atuendo (modelos 3D o similares).

7.2.3. Objetivos

7.2.3.1. Objetivo General

El objetivo general es cumplir con las especificaciones del proyecto del curso Ingeniería de software durante el período académico 2020-1 mientras que el grupo de trabajo bajo el nombre de “Software Plus” se enfrenta a desarrollar un software funcional de calidad en un ambiente de trabajo maduro y altamente profesional.

7.2.3.2. Objetivos Específicos

- Completar las funcionalidades del software en su mayoría, por lo menos un 70% de la totalidad de Smart Closet.
- Crear la documentación requerida por las especificaciones del curso, además de la necesaria para el desarrollo del ciclo de vida del software propuesto.
- Alcanzar una calificación sobresaliente en el proyecto del curso, por lo menos un 4,2 sobre 5.
- Establecer y ejecutar una metodología de trabajo interna para “Software Plus” que permita facilitar el desarrollo del proyecto.
- Que el software desarrollado sea de uso más allá del curso de Ingeniería de Software, para sus responsables y otros interesados.

7.3. Supuestos y Restricciones

7.3.1. Supuestos

Con el fin de que el proyecto se desarrolle de manera eficiente se asume que:

- El sistema cuenta con una conexión continua a un sistema que informe el clima actual de cierta localización.
- El sistema cuenta con una conexión continua a los motores de búsqueda que utilizará para la obtención de las imágenes de las prendas de los usuarios.
- Se cuenta con las herramientas necesarias para la elaboración del proyecto.
- La universidad no cancelará/aplazará las clases y materias del semestre
- No se pasará del presupuesto inicial del proyecto.

7.3.2. Restricciones

Algunos elementos que pueden restringir el libre desarrollo del proyecto son:

- Los integrantes del proyecto tienen 6 horas semanales para dedicarle al desarrollo del proyecto.
- El presupuesto económico para el desarrollo del proyecto es relativamente bajo y con poca probabilidad de cambiar en el futuro próximo.
- Los requisitos planteados, esto es, un modelo de Cliente-Servidor, una interfaz gráfica, el uso de una base de datos y persistencia.
- El tiempo para el desarrollo del proyecto está limitado al período académico 2020-1. Durante el mismo período se deben desarrollar y corregir tres diferentes entregas.

7.4. Evolución del Plan

En esta sección se especificarán las actividades requeridas para las actualizaciones del documento y se explicarán los mecanismos utilizados para realizar las

actualizaciones de este [1]. Todo el documento está sujeto a la [sección 12.3 Administración de configuración y documentación](#).

7.4.1.Objetivos

- Determinar las actividades que conciernen la evolución del plan del documento SPMP.
- Desarrollar los mecanismos de control de cambios del documento.
- Garantizar un seguimiento óptimo en la elaboración del documento, teniendo en cuenta posibles opciones de mejoramiento.

7.4.2.Involucrados

- El equipo de trabajo del documento son los principales responsables, sin embargo, se delega al director del documento y a los finalizadores en la ejecución de la evolución del plan.
- Cualquier integrante puede solicitar una actualización, eliminación, adición o pequeñas modificaciones de cualquier sección del documento, dicho integrante es el responsable de efectuar su cambio una vez que todo el equipo esté de acuerdo con que el cambio sea válido.

7.4.3.Desarrollo de evolución del plan

La evolución del plan se pone en acción únicamente cuando se considere totalmente bajo condiciones específicas, como modificación de la propuesta que se desarrolla en el documento, cambios de integrantes, secciones y tareas del plan de desarrollo del proyecto; en ese sentido, si se desea añadir nuevos parámetros, estándares y/o métricas que permitan mejorar la calidad del documento.

Es relevante aclarar que modificaciones en la elaboración del documento, ya sean por falta de comprensión de los objetivos del negocio, desconocimiento de especificaciones del SPMP, errores gramaticales, ortográficas y/o semánticas; no se consideran modificaciones en la cuales tengan que intervenir la evolución del plan del documento.

Las principales razones por las que es necesario llevar a cabo estas actualizaciones son por solicitud de cualquiera de los integrantes del equipo de Software Plus o por petición de la docente Anabel Montero.

Los pasos a tener en cuenta a la hora de llevar a cabo la evolución del plan ya sea por solicitud de los integrantes por cualquiera de los motivos mencionados anteriormente, son los siguientes:

1. Cuando un integrante desee aportar alguna modificación en alguna sección en el documento SPMP o modificación de algún anexo que esté fuera del documento deberá realizar una entrada en la aplicación de Slack con su

motivación del cambio y se debe describir claramente qué se desea modificar del documento. Dicha entrada deberá ser revisada por los integrantes de Software Plus. En el caso de que la modificación sea solicitada por la docente, se continúa directamente al paso para discutir posibles mejoramientos.

2. Se programará una reunión adicional (sin que afecte las reuniones y las actividades ya planeadas) con el fin de discutir aquellas modificaciones que el integrante solicitante desea plasmar en el documento.
3. Se realizará una sesión de pros y contras ante la solicitud. Posteriormente se hará una votación para determinar si se acepta dicha modificación. En caso de haber un empate, la decisión final se la lleva el azar: Se lanza una moneda y en caso de salir cara se prosigue con la solicitud. Si sale sello, la solicitud se rechaza.
4. Una vez aprobada la solicitud de modificación, se registra en el plan de actividades la actividad de modificación y se le asigna un nuevo tiempo en el calendario de trabajo y se le define un nuevo tiempo al proyecto, sin que afecte los objetivos de este. La actividad será desarrollada por el integrante que sugirió la modificación.
5. El responsable efectúa el cambio en el tiempo asignado y registra dicho cambio en el historial de cambios del documento.
6. Se consigna el cambio en el plan de calidad para futuras consideraciones y mejoramiento del proyecto.

7.5. Glosario

A continuación, se puede visualizar la tabla de glosario, en la cual se encuentran consignados los términos de negocio más relevantes con su respectiva definición.

Término	Definición
Closet	Un conjunto de prendas que el usuario registra para utilizar en la aplicación
Prenda	Artículo utilizado para vestirse
Mockups	Representación de una interfaz gráfica del programa final
Atuendo	Vestimenta o conjunto de prendas que son generadas para el usuario
Prototipo	Primer ejemplar que se fabrica y que sirve de modelo para fabricar otras iguales.
Front-end	Incluye todo el software y hardware que hace parte de la interfaz de usuario
Back-end	Se refiere a partes de la aplicación de la computadora o el código de un programa que permiten operar y a las que no se puede acceder mediante un uso.
Integración	Se refiere a las partes de la aplicación que deben existir entre el back-end y el front-end

Actividad	Un conjunto de tareas que se realizan con un propósito específico. Las actividades pueden incluir algunas o muchas tareas, dependiendo de su objetivo.
Cliente	Es aquella persona que a cambio de un pago recibe servicios de alguien que se los presta por ese concepto.
Usuario	Es una persona que utiliza una computadora o un servicio de red. Los usuarios de sistemas informáticos generalmente carecen de la experiencia técnica necesaria para comprender completamente cómo funcionan.
Proceso	Un conjunto de actividades que se realiza hacia un propósito específico.

Tabla 2. Glosario

8. Contexto del Proyecto

En esta sección se estipularán los aspectos más importantes del funcionamiento interno del proyecto, identificando el modelo de ciclo de vida base y aquellas actividades de los demás modelos que serán adoptadas para el desarrollo del mismo. Se especificará de la misma manera las herramientas utilizadas en cada fase y los entregables que cada una de estas producirá. Finalmente se identificarán las comunicaciones necesarias con entidades externas.

8.1. Modelo de Ciclo de Vida

Con el fin de establecer el orden de las etapas del proceso de software y los criterios a tener en cuenta para poder pasar de una etapa a la siguiente [2], se adoptará y adaptará un marco de trabajo para planear, gestionar, desarrollar y controlar los procesos relacionados con el proyecto [3]. Este marco de trabajo, conocido en ingeniería de software como modelos de ciclos de vida será seleccionado según las características propias del proyecto y para su elección se tuvo en cuenta lo siguiente:

- Los requerimientos serán definidos de la manera más precisa, clara y puntual posible al principio del desarrollo con el fin de evitar cambios a lo largo del desarrollo del proceso, sin embargo, se es consciente de que estos pueden llegar a ocurrir en cualquier momento.
- Deben existir tres entregables para alcanzar el proyecto en su totalidad y por lo menos dos de ellas deben incluir prototipos funcionales.
- El desarrollo del software en cuanto a la parte de programación estará liderado por dos de los integrantes del grupo que tienen mayor conocimiento en cuanto al desarrollo de aplicaciones web, sin embargo, el resto de integrantes de ser necesario se capacitará para poder aportar y apoyar en el desarrollo del código.
- Para el equipo es muy importante estar en constante comunicación e interacción con el fin de fortalecer el grupo, mejorar las relaciones entre sus integrantes, resaltar fortalezas y debilidades en cuanto al trabajo de manera

continua. Estas actividades son consideradas importantes para que los proyectos se desarrollen en torno a individuos y ambientes motivados.

Partiendo de estas premisas se puede notar que no existe un marco de trabajo que junto a todas ellas, por lo que es necesario buscar aquellos modelos de ciclo de vida que cumplan con al menos una de las características y formar un híbrido entre estos modelos con el fin de satisfacer las necesidades que se tienen como equipo para llevar a cabo el proyecto de la mejor manera posible.

Después de evaluar diferentes marcos de trabajo se eligieron las tres que están más próximas a las características previamente definidas, estas serán evaluadas a continuación:

8.1.1.Ciclo de vida incremental

Como su nombre lo dice, es un modelo que se basa en la construcción de incrementos que cumplen las diferentes funcionalidades del sistema. Esto se hace con el fin de entregar sistemas pequeños lo cual trae consigo menos riesgos, se da la posibilidad de responder a cambios evitando el fallo total del proyecto y el desarrollo individual de las diferentes funcionalidades revela de manera más clara los requerimientos del usuario [4].

Características:

- Divide el proyecto total en n entregables donde cada uno cumple con un proceso de análisis, diseño, codificación y pruebas. Una vez finaliza un entregable se inicia con el siguiente y se desarrolla el mismo proceso.
- Es un modelo muy útil cuando el usuario necesita entregas de manera periódica, aunque sean parciales [4].

8.1.2.Ciclo de vida Scrum

Es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto [5][6][7]. Es un proceso ideal para proyectos en entornos complejos, que requieren resultados rápidos; requisitos cambiantes o sin definir, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales. Para cumplir con lo anterior, en SCRUM se realizan entregas parciales y regulares del producto final, esto con el fin de beneficiar al cliente [8].

Se compone de cuatro partes fundamentales [8]:

- **Sprint:** Etapa en la que se desarrolla un incremento de la funcionalidad
- **Ceremonias y reuniones:** Se realizan distintas reuniones, daily scrum (reuniones al inicio del día contando el trabajo), reuniones para planificar los Sprint (Sprint Planning); la correspondiente retrospectiva (Sprint Retrospective) y el resumen del Sprint (Sprint Review).

- **Artefactos:**
 - **Product Backlog:** Listado con los requisitos del sistema, incorpora constantemente necesidades del sistema.
 - **Sprint Backlog:** Lista de tareas extraídas del Product Backlog para ser desarrolladas.
 - **Burndown Chart:** Gráfica del estado actual del sprint.
- **Roles:**
 - **Product Owner:** Interesado en el producto final, quien se relaciona con clientes para representar la visión de este producto.
 - **Scrum Master:** Experto y responsable del modelo SCRUM.
 - **Scrum Team:** Transforma las tareas del Spring Backlog en un incremento de la funcionalidad de software.

8.1.3. Extreme Programming XP

Metodología ágil que se centra en impulsar las relaciones interpersonales con el fin de que el desarrollo del software sea exitoso. Esta metodología promueve el trabajo en equipo, el aprendizaje cruzado entre integrantes y proporciona un buen clima de trabajo, además, contemplan la retroalimentación continua y la comunicación constante entre participantes [5][9].

Características:

- Hay mucha comunicación entre el equipo.
- Metodología basada en pruebas y error.
- Maneja el principio de simplicidad.
- Fundamentada en valores y prácticas.
- Existe una retroalimentación constante y concreta.

Teniendo en cuenta estos tres marcos de trabajo, el proyecto tomará como modelo base el ciclo de vida incremental, donde los incrementos serán cada una de las entregas que se realizarán a lo largo del curso, una vez estos sean evaluados por la docente se realizará la corrección de cada una de las entregas con el fin de entregar un producto final completo, sin embargo esta metodología tradicional no cumple completamente con las características propuestas por el grupo de trabajo, así que se adoptaran las formas de trabajo en equipo de las metodologías ágiles SCRUM y XP: las reuniones continuas para planificar, ejecutar y retroalimentar cada una de las tareas del proyecto, el trabajo fundamentado en valores y buenas prácticas, el aprendizaje cruzado entre integrantes y la pertenencia del código de todos los integrantes del equipo.

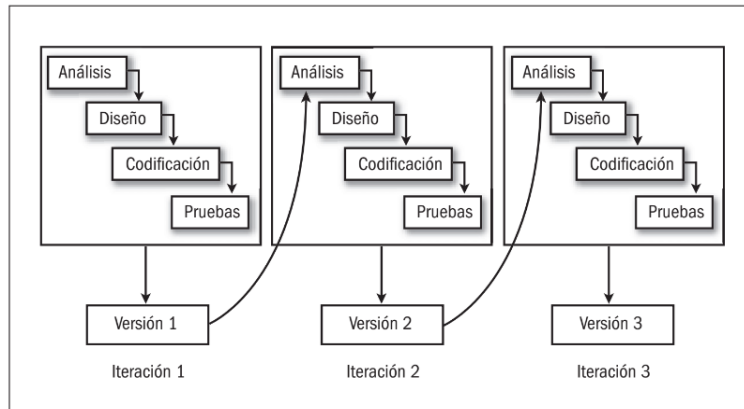


Ilustración 1. Estructura del modelo incremental [10]

8.2. Lenguajes y Herramientas

Esta sección tiene como objetivo describir los lenguajes y herramientas que se utilizarán en el desarrollo del producto y que brindarán apoyo y conocimiento a los miembros de Software Plus. Se describirán detalladamente los lenguajes y herramientas a utilizar para conocer los beneficios y las utilidades de estos dentro del marco del proyecto.

Los lenguajes se dividirán en dos grupos:

- **Planeación:** Los lenguajes cuyo uso se enfocan en la creación de documentos y diagramas.
- **Implementación:** Los lenguajes cuyo uso se enfocan en la implementación y manejo del desarrollo del producto.

Para la selección de dichos lenguajes se tuvieron en cuenta los siguientes criterios y especificaciones:

- Ser software libre o tener una licencia gratuita
- Curva de aprendizaje corta ya que el tiempo para el desarrollo de la aplicación es corto y limitado es necesario que el lenguaje sea fácil de aprender ya que no se puede dedicar demasiado tiempo al aprendizaje.
- Poder integrarla con el resto de lenguajes de implementación.
- Se pueda encontrar información detallada en internet y además tenga una documentación adecuada que el grupo pueda utilizar en caso de tener problemas con el lenguaje.
- Dar prioridad a los lenguajes más populares, ya que estos son los más utilizados para los proyectos de software y además es más probable que dicho lenguaje tenga una experiencia mayor en el mercado.

En la *Tabla 3. Lenguajes de planeación* se muestran los lenguajes que van a ser utilizados para el diseño y la documentación del producto.

Lenguaje	Versión	Enlace de descarga	Uso
UML (Enterprise Architect) [11]	2.5 15.1	https://sparxsystems.com/products/ea/	Se utiliza para diseñar los diagramas y la arquitectura del sistema.
BPMN [12]	3.6	https://www.bizagi.com/es/plataforma/modeler	Se utiliza para los procesos de negocio que tendría el sistema.
GlooMaps [13]	N/A		Se utiliza para realizar el WBS del proyecto.
TeamGantt [14]	N/A		Se utiliza para los calendarios y cronogramas del proyecto.

Tabla 3. Lenguajes de planeación

En la *Tabla 4. Lenguajes de implementación* se muestran los lenguajes que van a ser utilizados para el desarrollo de Smart Closet.

Lenguaje	Versión	Enlace de descarga	Uso
HTML [15]	5.3	No es necesaria una descarga	Se utiliza para la programación de la interfaz gráfica de la aplicación web
CSS [16]	3	No es necesaria una descarga	Se utiliza para realizar los estilos sobre archivos HTML
Typescript [17]	3.7	https://www.typescriptlang.org/index.html#download-links	Se utiliza para la programación de la aplicación web
Angular [18]	8.2.11	https://angular.io/guide/setup-local	Framework para aplicaciones web desarrollado en Typescript.
Cloud Firestore [19]	N/A	https://firebase.google.com/products/firestore/?hl=es-419	Se utiliza como la base de datos y se encargará de la persistencia de la aplicación web.

Tabla 4. Lenguajes de implementación

Por último, en la *Tabla 5. Herramientas* se muestran las herramientas que se utilizarán para que el desarrollo y funcionamiento de la aplicación sea posible.

Herramientas	Versión	Enlace de Descarga	Uso
GitHub [20]	2.25.1	No es necesaria una descarga	Se utiliza para tener acceso grupal a un repositorio del proyecto.
Gitkraken [21]	6.5	https://www.gitkraken.com/download/windows64	Se utiliza como interfaz de GitHub y para controlar el manejo de subida y edición de los archivos del repositorio.
Slack [22]	4.2.0	https://slack.com/intl/es-co/downloads/windows	Se utiliza para la comunicación entre el grupo y para el control del progreso de las tareas relacionadas con el proyecto.
Google Drive	N/A	No es necesaria una descarga	Se utiliza como repositorio de los documentos y diagramas que se realizan durante el desarrollo del proyecto.
Microsoft Word	Office 365 2019	https://setup.office.com/downloadoffice/	Se utiliza para la edición de los documentos del proyecto.
Microsoft Excel			Se utiliza para realizar las plantillas necesarias del proyecto.
Microsoft PowerPoint			Se utiliza para realizar las presentaciones necesarias del proyecto.
Visual Studio Code [23]	1.42.0	https://code.visualstudio.com/download	Se utiliza como editor de código para el desarrollo de la aplicación.
Google Hangouts	N/A	No es necesaria una descarga	Se utiliza para las reuniones virtuales.
Google Slides	N/A	No es necesaria una descarga	Se utiliza para realizar las presentaciones necesarias del proyecto.

Planning Poker [24]	N/A	No es necesaria una descarga	Se utiliza para realizar las estimaciones de las tareas del proyecto.
---------------------	-----	------------------------------	---

Tabla 5. Herramientas

8.3. Plan de Aceptación del Producto

En la siguiente sección se van a describir los entregables que el cliente final tendrá a medida que se desarrolla el proyecto Smart Closet. Se van a estipular los criterios de aceptación que el equipo de desarrollo de Software Plus y el cliente acordarán.

El propósito es que se ilustre de manera concisa y clara aquellas entregas y criterios para que el producto que se ofrece sea aceptado. A continuación, se puede ver en la *Tabla 6. Entregables* los entregables que el equipo de desarrollo irá llevando a cabo durante la planeación y ejecución del proyecto.

Entregable	Descripción	A quién va dirigido	Fecha de Entrega
SPMP	Documento especificado por el estándar IEEE 16326-2009., el cual detalla los planes, reglas y consideraciones a tener en cuenta para la elaboración de planeación de un proyecto de software [1][25].	Anabel Montero	04/03/2020
SRS	Es un documento que busca describir el desarrollo de un sistema de software, teniendo en cuenta una lista completa y necesaria de requerimientos del proyecto de software. Se encuentra especificado en el estándar IEEE 830. [4]	Anabel Montero	17/04/2020
SDD	Es un documento que describe el producto de un software para dar una guía a la arquitectura y diseño del proyecto de software. Se encuentra especificado en el estándar IEEE 1016 [8].	Anabel Montero	Por definir
Primer prototipo	Versión inicial del software Smart Closet con los requerimientos más relevantes.	Anabel Montero, Clientes	17/04/2020
Segundo prototipo	Versión alfa del software Smart Closet implementando las demás funcionalidades descritas en los requerimientos.	Anabel Montero, Clientes	Por definir

Diseño de Mockups de interfaces gráficas del sistema	Ilustraciones gráficas de cómo se pretende ver la interfaz del aplicativo web para el cliente y las diferentes pantallas de navegación que este dispondrá [26].	Anabel Montero, Clientes	04/03/2020
Documento de Casos de Uso	Especificación del seguimiento de eventos que tendrá las funcionalidades del software, incluyendo la interacción que tiene el cliente con el sistema, así como otros actores [27].	Anabel Montero, Clientes	17/04/2020
Manuales y documentación adicionales	Documentos que registran información adicional para que los interesados en el negocio tengan presente y comprendan tanto la función del sistema como el desarrollo de este.	Stakeholders	Por definir
Acta de reunión	Documento que registra las reuniones que el equipo ha efectuado, el cual detalla los avances de la elaboración del proyecto como también discusiones y decisiones que se llevaron a cabo	Anabel Montero	04/03/2020, 17/04/2020, por definir

Tabla 6. Entregables

Una vez ya establecidas las entregas que se llevarán a cabo, se describen aquellos criterios que se deben tener en cuenta para que sea aceptado por tanto por los clientes como para el equipo. En la *Tabla 7. Criterios de aceptación* se puede ver los criterios de aceptación de cada entrega.

Entregable	Criterios	Técnica o Herramienta
SPMP	<p>Formato documento:</p> <ul style="list-style-type: none"> • Interlineado: normal • Letra Arial • Tamaño: 12 • Tamaño máximo: 50 páginas <p>Contenido:</p> <ul style="list-style-type: none"> • Se debe completar cada sección de la plantilla. • Es necesario que disponga de tablas, gráficos, referencias y subsecciones • La rúbrica de la entrega SPMP permite brindar los criterios que 	<p>Se dispondrá de la herramienta “Docs” de Google para realizar los avances del documento. En adición, se hará uso de otras herramientas como Microsoft Word para la finalización del documento. Así mismo en la sección de Lenguajes y Herramientas se puede ver las demás herramientas que se utilizarán para brindar mayor soporte al documento.</p> <p>Se hará uso de la rúbrica de calificación y la plantilla de elaboración del documento SPMP como herramienta de</p>

	deben seguirse, y el documento debe ceñirse a ellos.	<p>obtención de los criterios de contenido.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>
SRS	<p>Formato documento:</p> <ul style="list-style-type: none"> • Interlineado: normal • Letra Arial • Tamaño: 12 • Tamaño máximo: 30 páginas <p>Contenido:</p> <ul style="list-style-type: none"> • Se debe completar cada sección de la plantilla. • Es necesario que disponga de tablas, gráficos, referencias y subsecciones • La rúbrica de la entrega SRS permite brindar los criterios que deben seguirse, y el documento debe ceñirse a ellos. 	<p>Se dispondrá de la herramienta “Docs” de Google para realizar los avances del documento. En adición, se hará uso de otras herramientas como Microsoft Word para la finalización del documento.</p> <p>Se hará uso de la rúbrica de calificación y la plantilla de elaboración del documento SRS como herramienta de obtención de los criterios de contenido.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>
SDD	<p>Formato documento:</p> <ul style="list-style-type: none"> • Interlineado: normal • Letra Arial • Tamaño: 12 • Tamaño máximo: 25 páginas <p>Contenido:</p> <ul style="list-style-type: none"> • Se debe completar cada sección de la plantilla. • Es necesario que disponga de tablas, gráficos, referencias y subsecciones • La rúbrica de la entrega SDD permite brindar los criterios que deben seguirse, y el documento debe ceñirse a ellos. 	<p>Se dispondrá de la herramienta “Docs” de Google para realizar los avances del documento. En adición, se hará uso de otras herramientas como Microsoft Word para la finalización del documento.</p> <p>Se hará uso de la rúbrica de calificación y la plantilla de elaboración del documento SDD como herramienta de obtención de los criterios de contenido.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>

Primer Prototipo	<ul style="list-style-type: none"> Los casos de usos más completos o que representen la funcionalidad principal del software debe estar correctamente implementado pasando correctamente por el filtro de un entregable del modelo de ciclo de vida incremental con el tiempo y esfuerzo que se le designe en el plan, así mismo debe estar especificado en el documento de Casos de Uso. El documento debe suplir los criterios que se presentan en la rúbrica de calificación. 	<p>Principalmente se usará un documento de casos de usos en donde se registra con detalle los casos de uso que se tendrán en cuenta para el software y como se detalla cada uno.</p> <p>La rúbrica de calificación de los casos de uso se tendrá en cuenta para los criterios aceptación.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>
Segundo Prototipo	<ul style="list-style-type: none"> La mayoría de los casos de uso deberían ya estar implementados correctamente, cumpliendo con el filtro de entregable del modelo de ciclo de vida incremental, siendo estos documentados en el documento de Casos de Uso y cumpliendo los tiempos y esfuerzos estimados para estos. El documento debe suplir los criterios que se presentan en la rúbrica de calificación. 	<p>Principalmente se usará un documento de casos de usos en donde se registra con detalle los casos de uso que se tendrán en cuenta para el software y como se detalla cada uno.</p> <p>La rúbrica de calificación de los casos de uso se tendrá en cuenta para los criterios aceptación.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>
Diseño de Mockups de interfaces gráficas del sistema	<ul style="list-style-type: none"> Se espera que sea consistente con los casos de uso especificados. Que no se pretenda mostrar más de lo que el resultado final puede ser. Claro, comprensible y estéticamente cómodo. 	<p>Se usará una de las herramientas descritas en la sección de Lenguajes y Herramientas.</p> <p>Se tiene en cuenta la rúbrica de calificación de SPMP para el cumplimiento de los criterios.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>
Documento de Casos de Uso	<ul style="list-style-type: none"> Consistencia y alcance en los requerimientos funcionales estipulados. 	<p>Plantilla de documento de Casos de Uso.</p> <p>Rúbrica de calificación que corresponda.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>

Manuales y documentación adicionales	<ul style="list-style-type: none"> Los documentos adicionales deben complementar lo suficiente el proyecto, tanto en su elaboración como en el resultado final, sin dejar dudas de la razón del sistema. Debe cumplir con los criterios de calificación de la entrega 	<p>Rúbrica de calificación que corresponda.</p> <p>Se tendrá en cuenta la sección 12.4. Control de calidad.</p>
Acta de reunión	<ul style="list-style-type: none"> Se debe dejar en detalle todas las acciones que se efectuaron en cada reunión durante la elaboración del proyecto: Qué se discutió, que se hizo, que se planeó, fechas y participantes. 	<p>Anexo 5. Plantilla de actas de reunión</p>

Tabla 7. Criterios de aceptación

8.4. Organización del Proyecto y Comunicación

En la *Tabla 8. Entidades externas*, se presentan las interfaces externas las cuales influyen de manera directa en el desarrollo el proyecto.

Entidad	Descripción	Responsabilidades	Canal de comunicación
Anabel Montero Posada	Profesora del curso - Ingeniería de Software, la cual toma los roles de cliente y tutora durante el desarrollo del proyecto	<ul style="list-style-type: none"> Comunicación constante con el equipo tanto presencial como virtual (Correo institucional). Revisión constante sobre los adelantos del proyecto. 	<ul style="list-style-type: none"> Horarios de atención a estudiantes. Correo institucional.
María Victoria Tovar	Diseñadora textil y Maestra en Diseño y Creación, la cual tiene el rol de experta en el tema de vestimenta.	<ul style="list-style-type: none"> Reunión establecida con el grupo del proyecto para establecer estándares respecto a la vestimenta de las personas de una manera general. En el anexo 10. Análisis Entrevista Experto se encuentran las preguntas realizadas al 	<ul style="list-style-type: none"> Reunión presencial establecida por medio de una integrante del grupo.

		<p>experto y su debido análisis que se tendrá en cuenta a la hora de realizar los prototipos.</p> <ul style="list-style-type: none"> Además, en el anexo 11. Audio Entrevista, se encuentra la entrevista realizada al experto. 	
Interesados	Personas interesadas en el uso de la aplicación	<ul style="list-style-type: none"> Contribuir al uso de la plataforma. Correcto uso de la aplicación. 	
Weather Underground	Plataforma la cual permite conexión directa a un sistema con el clima actual, pronósticos y demás.	<ul style="list-style-type: none"> Proveer toda la información del clima relevante para llevar a cabo la plataforma web. 	<ul style="list-style-type: none"> Siempre hay disponibilidad para entrar a la plataforma web.

Tabla 8. Entidades externas y stakeholders

8.4.1. Organigrama y roles

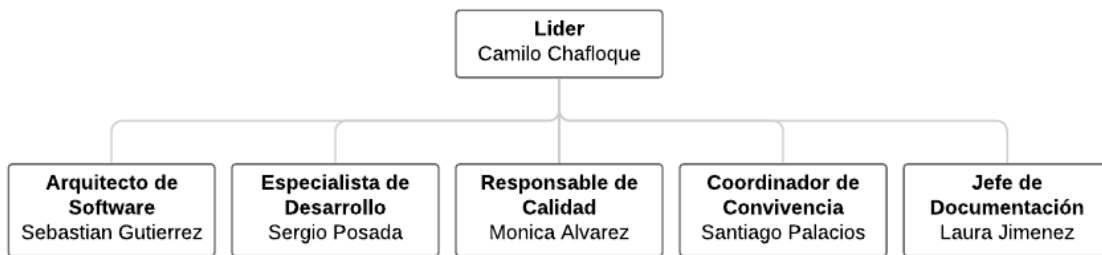


Ilustración 2. Organigrama

Líder

Puesto ocupado por Camilo Chafloque. La función del Líder es la de dirigir al grupo en el avanzar del proyecto y velar por la ejecución del mismo. Encargado de realizar la organización de las reuniones del grupo de trabajo, distribuir las funciones de trabajo de los integrantes y llevar un control de las actividades llevadas a cabo por los integrantes del proyecto mediante un control de reportes.

Responsable de Calidad

Puesto ocupado por Monica Alvarez. Encargada de asegurar la calidad del trabajo realizado, tanto en proceso como en producto, durante el desarrollo del proyecto sobre los diferentes entregables, tanto de documentación como de otros (código, gráficas). Responsable de controlar la ortografía en documentos y anexos,

inspeccionar las fuentes bibliográficas usadas y la veracidad en estas, revisar la calidad del funcionamiento del software y la completitud de otros entregables [28].

Coordinador de convivencia

Puesto ocupado por Santiago Palacios L. Tiene como objetivo velar por un buen ambiente de trabajo al interior del grupo, conoce a sus compañeros y es bueno para dialogar y encontrar soluciones. Encargado de coordinar la convivencia grupal, advertir aspectos que fallen en la convivencia del grupo, realizando los debidos llamados de atención, verificar el cumplimiento de normas por parte de los integrantes.

Especialista de Desarrollo

Puesto ocupado por Sergio Posada. El especialista de desarrollo conoce a fondo las herramientas de software que el equipo va a usar con el fin de brindar apoyo a sus compañeros sobre el uso correcto de las mismas. Encargado de brindar capacitación y soporte, llevar un control de calidad sobre el código para asegurar el buen funcionamiento del mismo y solucionar problemas de alta complejidad que requieran el uso de la herramienta.

Jefe de Documentación

Puesto Ocupado por Laura Jimenez. Su actuar se enfoca en la documentación que requiere el desarrollo del software, tiene amplios conocimientos sobre estándares y protocolos frecuentes. Responsable de verificar la calidad de la documentación creada para entregar a clientes, usuarios, integrantes del grupo y otros interesados, además de realizar informes o reportes [29][30].

Arquitecto de software

Puesto ocupado por Sebastian Gutierrez. Experto en trasladar los requerimientos y necesidades del cliente a un modelo de arquitectura para solucionar el problema mediante el uso eficaz del software y otras herramientas. Responsable de comunicar y discutir con el equipo las estrategias y soluciones técnicas que se ejecutarán mediante software, además, de reconocer a tiempo errores de diseño que puedan afectar el futuro del proyecto [31][32].

9. Administración del Proyecto

9.1. Métodos y Herramientas de Estimación

En esta sección se fundamentarán y justificarán adecuadamente todas las estimaciones realizadas en el presente documento. Para esto se especificarán herramientas y técnicas que apoyen las estimaciones. El objetivo de esta sección es determinar el costo estimado para el desarrollo de Smart Closet y del proyecto

en general, teniendo en cuenta diferentes parámetros que se mencionarán más adelante.

Para ser más específicos se escogerán los siguientes parámetros para la evaluación de las estimaciones del proyecto:

- **Esfuerzo:** Para lograr una buena estimación, es necesario conocer la dificultad de cada tarea y actividad y así poder asignar un mayor número de personas al desarrollo de esta, así como asignarle un estimado de tiempo de realización más largo.
- **Tiempo:** Para determinar la estimación del tiempo de desarrollo del software que el equipo de trabajo iba a tener, se utilizó el método de desglose de tareas y el diagrama de Gantt, con el fin de estimar el tiempo de desarrollo de éstas, de manera individual.
- **Presupuesto:** Se tiene que considerar también el presupuesto del proyecto en términos de costos de esfuerzo y costos de hardware y software. Estas consideraciones se pueden ver a más detalle en la sección [9.3.4 Presupuesto](#)

Para realizar este proceso se consideró usar la herramienta de *Planningpoker.com* propuesta en la metodología ágil Scrum, con el fin de involucrar a todos los miembros del equipo en la estimación. Esta herramienta permite llevar a cabo el método de planeación *Planning Poker*, haciendo que cada integrante participe de forma “individual” para el cálculo de la estimación de cada actividad.

Primero, se realizó la estimación para los documentos que se tienen que entregar, estos son (SPMP, SRS Y SDD). Por último, se realizó un análisis profundo de los requerimientos y los casos de uso a implementar, con el fin de identificar pequeñas tareas que fuesen fáciles de estimar y así poder tener un desglose más detallado para poder realizar la estimación de los prototipos.

Para ambas estimaciones, se usó la herramienta online llamada *Planning Poker* y cada integrante realizó la estimación de esta serie de tareas, es importante tener en cuenta que la unidad de estimación se realizó en día(s). La configuración de la herramienta permitió calcular el promedio de la estimación dada por cada integrante y al finalizar, generar una hoja de cálculo con los resultados ([Anexo 1. Resultados Estimación](#)).

Como se puede observar en el anexo, para la primera parte de la estimación, que son los documentos, se estimó que por cada uno se iba a tardar 15 días.

Para la segunda parte de la estimación, que es todo lo relacionado con los prototipos y sus requerimientos, la mayoría de las tareas se estiman entre 1 y 3 días, a excepción de algunas tareas bastante complejas que llegan a estimarse hasta en 5 días, realizando un cálculo total, la estimación para el desarrollo de software es de aproximadamente 57 días. Toca aclarar que los documentos SRS Y SDD se harán de manera transversal con los prototipos, por lo que los 15 días estimados para cada documento y los 57 días de los prototipos, son transversales.

9.2. Inicio del Proyecto

En esta sección se describirán los planes de capacitación de habilidad y/o conocimiento para el desarrollo del proyecto, en ese sentido se detallará la infraestructura, especificando las herramientas, recursos, actividades y roles de dichas tareas para poner en marcha el proyecto de Smart Closet.

9.2.1. Plan de capacitación

9.2.1.1. Objetivos

- Entrenar a los integrantes de Software Plus en las habilidades o conocimientos pertinentes para la elaboración óptima del proyecto [1].
- Especificar los procesos de enseñanzas de los integrantes hacia el resto del grupo.

9.2.1.2. Actores

- Principalmente los directores de cada una de las áreas de las que tienen suma habilidad y conocimiento.
- Cada integrante de Software Plus que considere que pueda aportar alguna capacitación hacia al resto del grupo, o que piense que se todos necesitamos aprender algo relevante para el proyecto.

9.2.1.3. Desarrollo del plan

Las capacitaciones se deben de llevar de la manera más eficiente por parte de los dueños del conocimiento, destacando sus roles en sus áreas. Esto se puede lograr a través de las participaciones de todos los integrantes del grupo al despejar sus dudas y efectuando los ejercicios propuestos en sesiones especiales.

Las sesiones de capacitación pueden ser solicitadas por cualquier integrante del grupo el cual considera imprescindible entrenar a los demás en el rol del cual fue selecto. El director de proyecto determina si las condiciones en las que se está desarrollando el proyecto permite programar una reunión sin afectar las demás actividades que ya fueron planeadas. El director toma como variables el progreso del proyecto, la urgencia de dicha reunión y la relevancia para el óptimo desarrollo. Las sesiones programadas deben ponerse en el calendario del proyecto para mantener el ritmo de este, asimismo deben ser preparadas con antelación para llevarse a cabo correctamente el día programado.

Las solicitudes deben diligenciarse en documento en el cual especifique el área del cual se debería capacitarse, el nombre del integrante, fecha de solicitud y una breve descripción por las cuales se deberían llevar a cabo dichos entrenamientos, en adición se puede añadir un pequeño esquema de cómo llevaría los entrenamientos.

Este documento es enviado al director del proyecto el cual realiza la respectiva evaluación de la solicitud.

Una vez las solicitudes sean aprobadas, en la reunión principal, el solicitante debe exponer las necesidades de los entrenamientos y exponer el esquema de cómo se desarrollarán las sesiones de entrenamiento; en este esquema se debe incluir la dinámica, las tareas a efectuar, el tiempo de duración de cada una, y el tiempo en total de toda la capacitación. Se debe plantear en dicho esquema las capacidades meta que van a lograr y proponer en qué actividades de proyecto se pueden llevar a cabo.

Durante las sesiones de entrenamiento que ya fueron programadas y que se llevarán a cabo, debe existir un *Acta de Entrenamiento* el cual se registra las tareas y actividades que tuvieron lugar en cada sesión y los participantes que asistieron.

Cuando los entrenamientos hayan finalizado, el solicitante principal hace una breve retroalimentación de cada entrenamiento y evalúa si las sesiones fueron fructíferas y si se logró el objetivo de su solicitud. Se envía el acta al director del proyecto para que verifique el cierre de toda la capacitación.

9.2.2. Infraestructura

9.2.2.1. Objetivos

- Establecer las tareas, herramientas y roles de los ejecutores de las tareas.
- Asignar horarios de ejecución de las tareas y la duración de cada uno [1].
- Establecer los procedimientos que se llevarán a cabo a lo largo del proyecto [1].

9.2.2.2. Actores

- La infraestructura del proyecto es gestionada principalmente por el director del proyecto.

9.2.2.3. Tareas de instalación

Las tareas primordiales en la infraestructura del proyecto son las tareas de instalación que se deben tener en cuenta para la elaboración del proyecto de software. Se precisa identificar los equipos de cómputo necesarios para disponer de las herramientas descritas en la [Sección 8.2 Lenguajes y herramientas](#), y así ser descargados e instalados ya que estos son necesarios para el desarrollo del software. El responsable de la sección anteriormente mencionada provee las instrucciones para la descarga e instalación de estas herramientas.

El especialista de desarrollo es el que gestionará la base de datos para que el resto del grupo disponga de ella en un servidor común y en este se generen usuarios para

que puedan trabajar sobre ella, realizando pruebas y contribuyendo en los avances pertinentes. Este especialista brindará información necesaria para el óptimo desempeño de la base de datos, si es necesario éste podrá disponer del plan de capacitación para garantizar que todos adquieran las habilidades necesarias.

Se dispondrá de un computador que tomará el papel de servidor, este tendrá la función de mantener el software, garantizando disponibilidad y transparencia de fallos. El especialista se encargará de montar el servidor y obrará por mantenerlo funcional en todo momento.

Estas tareas se harán una única vez por incremento, es decir, que solo se harán cuando la base datos, el servidor y cada herramienta sea necesaria en una específica parte del proyecto y se repetirán cuando se precisen disponer demás herramientas.

9.2.2.4. Mantenimiento

El arquitecto de software junto con el especialista de desarrollo se encargará de coordinar que las herramientas dispuestas para la realización del software estén siendo manipuladas correctamente por los demás integrantes de Software Plus. Estos brindarán mantenimiento y control de errores de las herramientas, servidor, hardware y base de datos; con el fin de evitar retrasos en el proyecto y conseguir un rendimiento eficiente en la elaboración de Smart Closet. Ellos harán revisiones de semanales del número de herramientas usadas en cada incremento, la frecuencia de uso y registro de alguna observación por parte cualquier integrante en el uso de la herramienta. Se debe controlar que el servidor esté únicamente ejecutando procesos relacionados con Smart Closet, y así evitar sobrecargas de trabajo y por siguiente retrasos y fallos. Cada integrante dispondrá de un ordenador para que puedan hacer uso de las herramientas, los responsables del mantenimiento deben supervisar que las máquinas estén funcionando en perfecto orden y se estén usando para fines del desarrollo del software.

9.2.2.5. Herramientas

Dirigirse a la [sección 8.2 Lenguajes y herramientas](#).

9.2.2.6. Hardware

El hardware empleado son 6 ordenadores que tomarán el rol de desarrollo del software y documentación del proyecto, son indispensables en la infraestructura. Son útiles para la comunicación entre integrantes y facilita la eficiencia de cada actividad. Las especificaciones del hardware se encuentran descritas en la *Tabla 9. Ordenadores del proyecto*, en esta tabla se describe cada característica de las máquinas y el propietario de cada máquina. Si bien los equipos no son de propiedad

de Software Plus, sino de cada integrante del equipo, el correcto uso y mantenimiento compete tanto a los dueños como a los supervisores.

Propietario	Modelo	Almacenamiento	Procesador	Memoria RAM	Sistema Operativo
Mónica Alvarez	HP Pavilion	HDD 500 GB	Intel Core i5 - 7200 U	6 GB	Windows 10
Juan Camilo Chafloque	Lenovo Thinkpad	SSD 512 GB	Intel Core i7 - 8550 U	16 GB	Windows 10
Sebastián Gutiérrez	Lenovo	HDD 500 GB	Intel Core i5 - 4200 U	4 GB	Windows 10 Pro
Laura Jimenez	HP	HDD 1 TB	Intel Core i7 - 4600 U	8 GB	Windows 10
Santiago Palacios	HP Envy	HDD 1TB	AMD FX 8800P	16GB	Windows 10
Sergio Posada	Torre	SSD 480 GB	Intel Core i5 - 9600k	16 GB	Windows 10

Tabla 9. Ordenadores del proyecto

9.2.2.7. Espacios físicos

El espacio físico de desarrollo del proyecto se efectúa principalmente en el piso 11 del Edificio de Ingeniería José Gabriel Maldonado - Laboratorios de la Pontificia Universidad Javeriana sede Bogotá, con dirección Cra. 7 # 40 - 62. En este espacio físico se desarrollan todas las reuniones de equipo en las cuales se gestiona, se discute y se avanza el proyecto de Smart Closet.

Se tiene pensado modificar el centro de reuniones cuando el proyecto llegue a determinado punto de avance hacia el centro de operaciones, el cual se sitúa en la morada de Juan Camilo Chafloque, con dirección Cll. 96 #19A-32, puesto que, al ser el director del proyecto, la óptima supervisión del proyecto será más eficiente y cómoda para todos, permitiendo concentración por parte de los integrantes, además de otros recursos para mantener a estos enfocados en cada avance, como los son alimentación, hidratación, espacios lúdicos, zonas de descanso, etc.

9.3. Planes de Trabajo del Proyecto

En esta sección se plantea los trabajos que se llevarán a cabo durante la ejecución del proyecto teniendo en cuenta los responsables del proyecto, el número de trabajos, el calendario del proyecto (el tiempo que toma cada uno de trabajo y el tiempo total), el presupuesto que toma la realización del proyecto y asimismo las herramientas pertinentes.

9.3.1.Responsables

Todo equipo de Software Plus va a trabajar y desarrollar cada actividad del proyecto en un tiempo medible y con las herramientas necesarias, ejecutando cada trabajo concorde a lo planeado.

A cada integrante se le asigna un trabajo del proyecto que cumplan con sus habilidades. Para los trabajos del desarrollo del software se designan roles importantes para que se pueda organizar de manera adecuada los responsables de dichos trabajos. En la parte de Frontend se designarán al Coordinador de Convivencia y a la Gestora de calidad, para el Backend se adecuan el director del proyecto y el Especialista de desarrollo y, por último, para la Integración los responsables son el Arquitecto de software y el Jefe de Documentación.

9.3.2.Descomposición de las actividades

Para identificación y descomposición de las actividades importantes en el proyecto se utilizó un WBS (Work Breakdown Structure) en la cual se dividen de manera jerárquica aquellas actividades que componen el desarrollo vital de proyecto, de esta manera se logra especificar la descripción de cada una de estas y haciéndolas lo más concretas posibles para que un solo actor las pueda ejecutar en un tiempo estimado [27].

Gracias al modelo de ciclo de vida propuesto en la [sección 10.1 Modelo de ciclo de vida](#) se pueden establecer qué procesos son los pertinentes a desarrollar y de qué manera se debería efectuar para poder dar como resultado la estructura de trabajos desglosados. Cabe aclarar que como se trabaja un ciclo de vida iterativo en cuanto actividades, el WBS contiene las actividades de desarrollo como un completo y no como cada repetición. La estructura resultante se puede ver en el [anexo 2. Diagrama WBS](#).

9.3.3.Calendarización

De acuerdo con el modelo ajustado que definió Software Plus en la [sección 10.1 Modelo de ciclo de vida](#), estas actividades constarán con un modelo incremental, en donde se destinará una porción determinada de tiempo para actividades que se repetirán con cada incremento, además de las actividades como equipo (reuniones de avance, reuniones diarias, reuniones esporádicas, etc.) que está dentro de las actividades del modelo que se definió. Con esto se plantea un cronograma de actividades que busca plasmar las tareas que se llevarán a cabo con los integrantes que realizarán cada actividad y el tiempo que se le destinará a cada una, se detalla una breve descripción de qué se hace en la actividad y con todo esto se establece un tiempo total de realización del proyecto. Se usa un diagrama de Gantt basado en las tareas propuestas en el WBS para poder proyectar cada una de las actividades junto a su duración y dependencias entre ellas. Se puede visualizar el diagrama en el [anexo 3. Diagrama Gantt](#).

9.3.4.Presupuesto

A continuación, se muestra los principales flujos de entrada y de salida que Software Plus necesita para las diferentes actividades principales del desarrollo del proyecto. Se tiene en cuenta que las entradas de dinero corresponden al capital de cada integrante y que las salidas corresponden al dinero invertido en la construcción del software que cada integrante dispone para el proyecto.

El flujo de entrada de dinero se limita únicamente al saldo inicial del proyecto por entrega que corresponde a la sumatoria del capital de cada integrante, puesto que temporalmente el proyecto como desarrollo no genera ingresos.

Cabe aclarar que cada actividad principal toma una duración estimada y tiene un periodo de desarrollo en semanas y que las actividades de Prototipo 1 y Prototipo 2 son actividades transversales a las actividades de SRS y SDD, la estimación de estos prototipos se encuentra en la [sección 11.1 Métodos y herramientas de estimación](#).

A continuación, se puede visualizar una tabla de flujo de caja donde se detallan los movimientos más importantes que se tendrán en cada actividad principal (ver Tabla XX Flujo de caja).

Actividades Principales	SPMP	SRS	SDD	Prototipo 1	Prototipo 2
Saldo inicial	1'500.000	1'130.000	910.000	690.000	520.000
Egresos					
Gastos de transporte	250.000	150.000	150.000	100.000	100.000
Gastos de alimentación	100.000	50.000	50.000	50.000	50.000
Gastos adicionales	20.000	10.000	10.000	10.000	10.000
Amonestaciones	20.000	10.000	10.000	10.000	10.000
Manutención de los equipos	0	0	0	0	0
Flujo de caja libre	1'130.000	910.000	690.000	520.000	350.000

Tabla 10. Flujo de caja

10. Monitoreo y Control del Proyecto

Dentro de esta sección se darán a conocer todas las metodologías y procedimientos necesarias para controlar y gestionar el proyecto con el fin de que siga por el camino establecido.

10.1. Administración de requerimientos

Con el fin de que el proyecto vaya encaminado correctamente es de gran importancia toda la gestión y administración de los requerimientos ya que estos son los que se encargan de dar toda la información pertinente y detallada respecto al proyecto por parte del objetivo de este.

Estos requerimientos son el medio para iniciar un proyecto debido a que estos permiten establecer todas las necesidades que presentan los interesados y afectados de este, también, qué se debe incluir dentro del proyecto para que cumpla todas las necesidades que se encuentren en estos [33].

Según [5], una buena administración de requerimientos cuenta con 2 pasos los cuales son la Planeación y la Administración de cambios de requerimientos las cuales serán explicadas más adelante.

10.1.1. Planeación de administración de requerimientos

Este paso establece el nivel de detalle que se requiere dentro de la administración de los requerimientos planteados en el proyecto donde se llevarán a cabo las siguientes tareas con el fin de que se tenga un control y manejo correcto sobre cada uno de estos. Estas tareas son las siguientes:

- **Identificación de requerimientos:** En este paso el equipo de trabajo identifica cada uno de los requerimientos de manera exclusiva de tal forma que se pueda tener presente cómo se relacionan internamente los requerimientos identificados, esto con el fin de usarlos en evaluaciones de seguimiento sobre estos.
- **Proceso de administración del cambio:** Esta actividad consta de un análisis detallado de las actividades que serían afectadas por un cambio establecido y cuánto va a ser el costo de este cambio.
- **Políticas de seguimientos:** Se establecerán políticas las cuales muestran las relaciones entre cada uno de los requerimientos y el diseño que se debe definir en el sistema.
- **Herramientas de apoyo:** Se tendrá un archivo con extensión .xlsx el cual tendrá toda la información pertinente a los requerimientos necesarios para el desarrollo del proyecto, una breve descripción y un historial de cambios para cada uno de estos para llevar un control adecuado.

10.1.2. Administración de cambios de requerimientos

La administración de cambios de requerimientos debe realizarse para cada uno de los cambios propuestos a estos. Es importante realizar este paso debido a que se debe determinar si los beneficios de implementar cierto cambio están justificados por los costos de esta implementación [5].

Consta de 3 pasos los cuales serán descritos a continuación:

- **Análisis del problema y especificación del cambio:** Este paso inicia con la identificación de un problema dentro de los requerimientos o la necesidad de un cambio en específico. después de esto la propuesta se analiza para definir si es válida o no.
- **Análisis del cambio y estimación del costo:** El efecto del cambio que se evidenció en el paso anterior se valoró utilizando la información general sobre los requerimientos del sistema. Este costo se define al evaluar las modificaciones que se deben realizar sobre el documento de los requerimientos y al diseño e implementación del sistema.
- **Implementación del cambio:** Se realizan los cambios pertinentes sobre el documento de requerimientos o el artefacto que se considere pertinente cambiar (Prototipos, diagramas, etc.), historia de cambios y documento de análisis y diseño en las partes donde se hacer referencia a este requerimiento.

Para este proyecto se desarrolló un diagrama BPMN el cual cuenta con los roles implicados en esta parte de la administración de requerimientos.

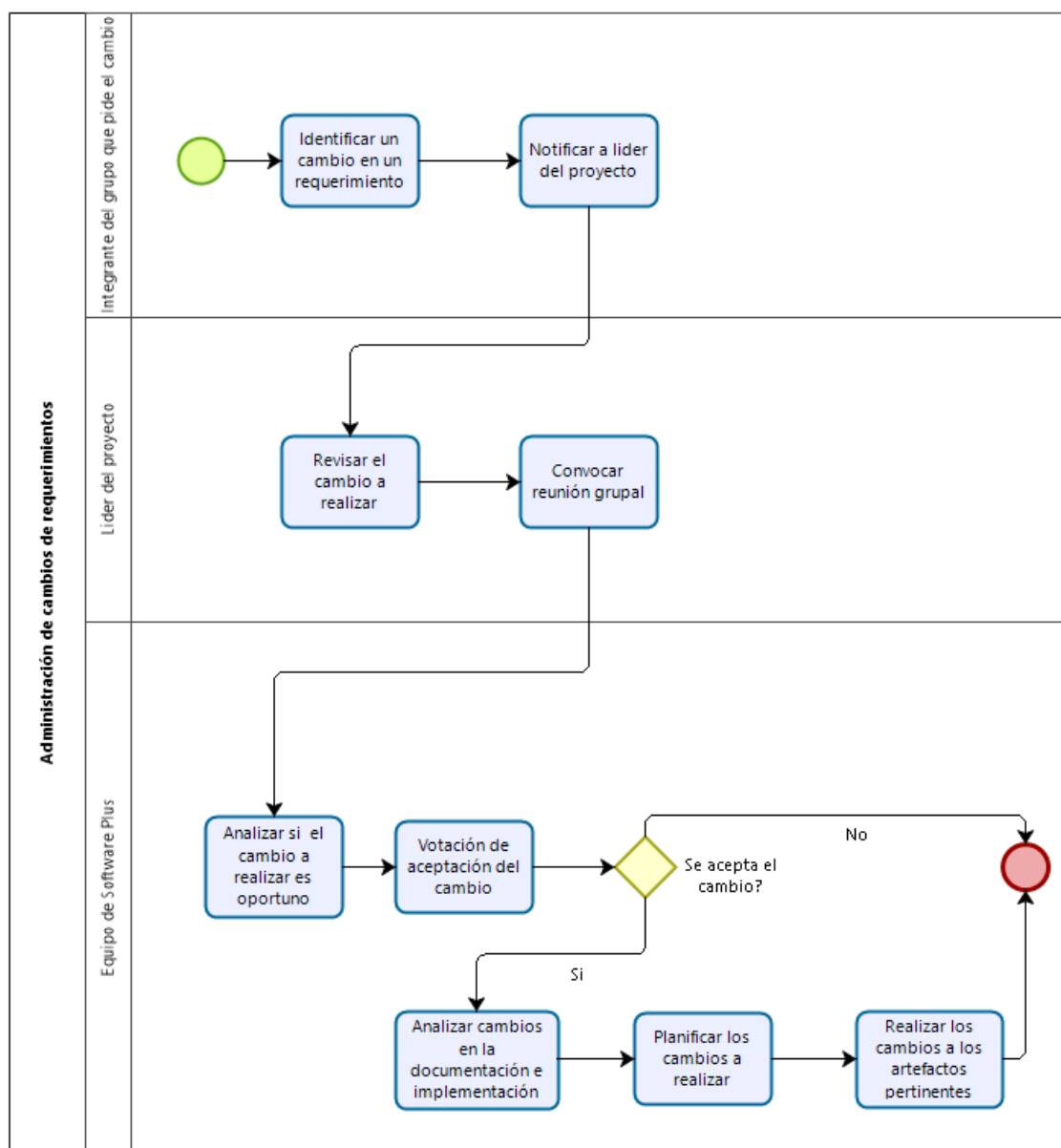


Ilustración 3. Diagrama BPMN de la administración de cambios de requerimientos

10.2. Monitoreo y Control del Progreso

Para el monitoreo y control del progreso se utilizó el diagrama de Gantt hecho al principio del proyecto. Como el diagrama de Gantt tiene todas las actividades a realizar para la primera parte del proyecto, se desglosó en un Excel cada una de estas actividades. En el Excel se desglosaron tanto el día de terminación estimado de la actividad como el día que se estimaron iba a durar dicha tarea.

Luego se hizo un acumulado con todos los días estimados de todas las actividades de la primera fase, el cual dio 95 con un total de 33 actividades a realizar. Luego se obtuvo cuánto sería el porcentaje de incremento individual del proyecto una vez se

terminará una actividad. Este incremento dio 1.053%, lo que significa que cada vez que se termine una actividad, el proyecto estaría completándose en ese porcentaje.

Una vez un integrante del grupo terminará una de estas actividades con todos los requisitos de calidad aprobados, se actualizaba el Excel con la fecha del día en que se terminó dicha actividad, para así ver cómo se comportaba el diagrama y tener una mejor visión de cómo se encontraba el grupo con respecto a las entregas.

Para realizar el diagrama se empezó con el supuesto de que 0 tareas estaban terminadas y por lo tanto faltaba el 100% del proyecto. Para cada día del calendario se miraba cuantas tareas se tenían estimadas para que se terminase ese día y también cuantas tareas en verdad se terminaron. Con estos valores se determinaba el decremento porcentual de la finalización del proyecto para cada día de la siguiente manera:

$$\text{Porcentaje Completitud}_t = \text{Porcentaje Completitud}_{t-1} - N_t * I$$

Donde:

- Porcentaje Completitud: Es el porcentaje actual donde se encuentra el proyecto. (100% significa que no se han realizado actividades y 0% significa que ya se terminaron todas las tareas)
- N: Es el número de tareas que se finalizaron en el día t
- I: Porcentaje incremental individual por tarea, establecido anteriormente como 1.053%.

Estos cálculos se realizaron tanto para el estimado como el real y el diagrama sería de la forma % vs. Días. Una vez se tienen las dos curvas (Estimada, real), se puede ver cómo está cumpliendo el grupo realmente frente a lo que estimó inicialmente.

Todo el monitoreo y control del progreso mencionado anteriormente se puede ver con más detalle en el [anexo 7. Burndown Chart](#).

10.2.1. Actividades de monitoreo

A continuación, se describirán todos los tipos de actividades que se llevaron a cabo durante el desarrollo del proyecto para mantener un monitoreo constante sobre qué tareas realizaba cada integrante y que día en específico se llevaba a cabo dicha tarea:

- **Revisión de preguntas diarias:** Para esta actividad se realizaron una serie de preguntas donde varias de estas tenían relación al proyecto. Estas preguntas diariamente eran revisadas por uno de nuestros integrantes con el fin de evaluar de que cada integrante del grupo estuviera respondiéndolas de forma adecuada.

10.2.2. Actividades para reportar el progreso

Estas actividades fueron de gran importancia durante el desarrollo del proyecto puesto que estas reportaban constantemente los avances de cada integrante del grupo y con el fin de llevar un buen control sobre estas se establecieron las siguientes tareas para reportar el progreso:

- **Actas de evaluación:** Esta actividad tiene como objetivo realizar una serie de actas las cuales permitan llevar un registro de las reuniones realizadas por el grupo de desarrollo y su respectiva relación con el proyecto. Con base en estas se puede evaluar y analizar si los objetivos de estas reuniones fueron cumplidos y definir los temas de las reuniones que se llevarán a cabo más adelante.
- **Reporte diario sobre tareas realizadas:** Para esta actividad se generaron una serie de preguntas con el fin de que cada integrante las respondiera diariamente y evidenciar el progreso que ese integrante realizó cada día. Estas preguntas se realizarán de manera virtual mediante la aplicación Slack al finalizar el día. La serie de preguntas fueron las siguientes:
 - ¿Qué hice hoy?
 - ¿Qué actividad realicé relacionada al proyecto?
 - ¿Qué inconvenientes tuve?
 - ¿Qué haré mañana?

Las cuales permitirán una buena comunicación entre el grupo de trabajo sobre las tareas y actividades que lleva a cabo cada integrante.

- **Reuniones presenciales:** Esta actividad tiene como fin que cada integrante indique qué tareas ha realizado respecto al proyecto, así como que se llevará a cabo en los próximos días. Estas tienen como fin fomentar la comunicación dentro del grupo y asignar tareas a cada uno de los integrantes de este.

10.2.3. Acciones disciplinarias

A continuación, se presentan las acciones disciplinarias que cuenta el grupo de Software Plus.

Nombre de la acción	Momento en el que se lleva a cabo	Descripción
Redistribución de roles en el equipo de trabajo.	Después de socializar las actividades realizadas por cada uno de los	Si se evidencia por parte de todo el grupo del proyecto que un rol no está funcionando se procederá con una reasignación de roles para

	integrantes del grupo de trabajo.	lograr un mejor desempeño a la hora de realizar las actividades.
Penalización en nota de autoevaluación.	En el momento en el que se evidencie ausencia de un integrante del grupo de trabajo.	Si un integrante falta a una reunión establecida sin justificación su nota será afectada negativamente al momento de realizar la autoevaluación y se procederá a llamarle la atención.

Tabla 11. Acciones disciplinarias

10.3. Cierre del Proyecto

Esta sección tiene como objetivo evaluar los criterios de terminación de cada una de las diferentes actividades y fases asociadas al desarrollo de software y asimismo realizar una recopilación del aprendizaje asociado al proyecto para mejorar procesos y definir los procedimientos para el desarrollo del reporte gerencial del proyecto [1][4]. Dicho reporte gerencial se encuentra en el [anexo 8. Reporte gerencial](#).

Para cada entrega del proyecto, se realizará una reunión donde se discutirán las acciones a realizar como consecuencia de las observaciones obtenidas por parte de los clientes. De igual manera se realizará una revisión final de calidad una vez el proyecto se finalice, siguiendo los parámetros establecidos en la [sección 12.4 Control de Calidad](#).

Además, con el fin de desarrollar el reporte gerencial, se utilizarán las métricas de control de progreso del proyecto en la [sección 10.2 Monitoreo y control del progreso](#) en donde se realizarán los análisis pertinentes del desempeño de cada integrante de Software Plus en relación con las tareas y aporte al desarrollo y desempeño del equipo, esto viene atado a retroalimentación por parte de todos los integrantes del grupo con el fin de generar una autoevaluación e identificar las fortalezas, debilidades y oportunidades de mejora para las siguientes fases y etapas del desarrollo.

11. Entrega del Producto

Esta sección tiene como objetivo realizar la transición del producto terminado hacia los clientes de Smart Closet [1]. La entrega final de la aplicación web estará dividida en cuatro componentes. La documentación formal del proyecto, la documentación de soporte, la aplicación web funcional y el componente de sustentación. Cada entregable es entregado conforme a los criterios estipulados en la [sección 8.3 Plan de aceptación del producto](#). En dicha sección se describieron detalladamente estos entregables.

En cuanto a la documentación formal se encuentran los documentos de plan de administración del proyecto de software (SPMP), la especificación de requisitos de

software (SRS) y la descripción de diseño de software (SDD). Los documentos de soporte tienen que ver con todos los documentos que soportaron las funcionalidades de los prototipos, como lo son los documentos de caso de uso, diagramas del sistema, pantallas (Mockups), manuales de usuario y otros posibles anexos. El componente de la aplicación web funcional son la entrega de los dos prototipos y por último el componente de sustentación abarca la presentación final entre el equipo de trabajo y el cliente y la demostración de la aplicación Smart Closet con su debida retroalimentación en la cual se atenderán aspectos a mejorar del aplicativo.

Entregable	Componentes
Documentación formal	<ul style="list-style-type: none"> • SPMP • SRS • SDD
Documentación de soporte	<ul style="list-style-type: none"> • Documentos de caso de uso • Diagramas del sistema • Manuales de usuario • Mockups • Requerimientos funcionales
Aplicación funcional	<ul style="list-style-type: none"> • Prototipo 1 • Prototipo 2
Componente de sustentación	<ul style="list-style-type: none"> • Exposiciones de cada entrega (SPMP, SRS, SDD) • Demostración de Smart Closet

Tabla 12. Entregables Finales

12. Procesos de Soporte

12.1. Ambiente de Trabajo

Antes de empezar con las reuniones en equipo se estableció un reglamento entre todos los miembros del grupo con el fin de que se hiciera un seguimiento adecuado en torno a la realización del proyecto. Los pilares más importantes de este reglamento se hablarán más adelante.

Antes de hacer énfasis en cada una de las categorías que definen el reglamento se debe tener en cuenta que cada decisión tomada buscaba tener la aprobación de la mayoría de los integrantes del equipo con el fin de que fuera de un modo democrático.

12.1.1. Reuniones y comunicación

Con el fin de que las reuniones se desarrollarán de una manera eficaz y con una buena planeación se establecieron las siguientes pautas. Dichas pautas también se establecieron y se definieron de una manera más detallada en la [sección 10.2 Monitoreo y control del progreso](#):

- Se definieron 2 reuniones presenciales durante la semana los días Lunes de 1pm a 4pm y los días Martes de 4pm a 8pm.
- Después de cada clase se lleva a cabo la denominada “fast meeting”, las cuales se realizan después de cada clase de Ingeniería de Software.
- La inasistencia a una reunión debe ser justificada a todos los miembros del equipo 1 día antes de la reunión.
- En caso de encontrarse en un estado crítico durante el desarrollo de los miércoles se establecieron reuniones de emergencia los días Jueves desde la 1pm a 8pm y los sábados y Domingos con un horario flexible para los integrantes del equipo.
- En cualquier momento un integrante del equipo puede solicitar una reunión virtual si se tiene un motivo específico.
- Se tomaron como medios de comunicación principales: El grupo de WhatsApp, Los canales de Slack y Hangouts.
- Los canales en Slack fueron los siguientes: “diario”, donde cada integrante respondía diariamente una serie de preguntas enfocadas tanto a nivel personal como a nivel de trabajo en el proyecto. “general”, donde el equipo discutía temas importantes relacionados con el proyecto, así como el envío de documentos importantes para este. Por último “polas”, donde se llevaba un registro de cuanta plata iba pagando cada integrante con el fin de usar esa plata para comprar cervezas al final del semestre.
- Para el desarrollo del trabajo se utilizaron las siguientes plataformas: Google Drive, Correo Institucional y los medios mencionados anteriormente.

12.1.2. Faltas graves

- En caso de que un integrante no justifique su falta a una reunión establecida a este se le bajará en la nota de la coevaluación.
- En caso de que un integrante llegue 10 minutos después a una reunión deberá pagar 1000 pesos por cada 10 minutos tarde que llegue a dicha reunión.
- Si un integrante tiene una causa justificada para no asistir a una reunión este deberá regalar dulces a cada uno de los demás integrantes.
- Si no se responden las preguntas diarias en el canal “diario” que se encuentra en Slack, el integrante deberá pagar 500 pesos por cada día que no realice esta actividad.

12.2. Análisis y Administración de Riesgos

En esta sección se presentará el plan de análisis y gestión de riesgos para generar acciones que puedan prevenir y mitigar de manera adecuada los riesgos más importantes que se puedan presentar a lo largo del proyecto o, en un caso extremo, corregir el estado del proyecto si uno de estos se llega a materializar. Se tendrá en cuenta los recursos necesarios y los responsables de cada una de las etapas en este análisis.

12.2.1. Objetivos

- Identificar los posibles riesgos que se puedan llegar a presentar a lo largo del desarrollo del proceso, estimar su posible impacto y priorizarlos.
- Analizar las posibles consecuencias que puede traer la materialización de dichos riesgos.
- Establecer un plan de contingencia que incluya los salvaguardas que sean necesarios para prevenir, mitigar o disminuir los riesgos y su respectivo impacto [34].
- Definir los responsables, las actividades y las herramientas a usar en el proceso de gestión de riesgos [1].

12.2.2. Responsables

Los responsables de liderar el proceso de análisis de riesgo es un director(a) de riesgos asignado que se encarga de direccionar la identificación, el análisis, las propuestas y planes de respuesta y el manejo del impacto de los riesgos del proyecto [35]. Este director(a) es Laura Jimenez, el resto del equipo de trabajo de Software Plus estará igualmente involucrado en el proceso haciendo aportes a las diferentes etapas los cuales serán evaluados y finalmente agregados al modelo de gestión del riesgo final.

12.2.3. Actividades

El plan de gestión de riesgos se descompone en cuatro procesos principales [2].

1. Identificación de riesgos.
2. Análisis cualitativo y cuantitativo de los riesgos.
3. Planeación de riesgos.
4. Monitoreo de riesgos.

1. Identificación de riesgos

En esta actividad se identifican y listan los riesgos que se puedan presentar a lo largo del desarrollo del proyecto.

Riesgo	Descripción
Riesgo 1	Funcionamiento incorrecto del API utilizada en la aplicación.
Riesgo 2	Estructura de estándares de moda mal definidos.
Riesgo 3	Subestimación de la complejidad de la aplicación.
Riesgo 4	Capacitación insuficiente y restringida respecto a las herramientas a usar.
Riesgo 5	Manejo inadecuado del control de versiones.
Riesgo 6	Pérdida del interés por parte del usuario.
Riesgo 7	Modificaciones excesivas del cronograma.
Riesgo 8	Hardware limitado que no soporte la aplicación
Riesgo 9	Problemas de convivencia que afecten el rendimiento del grupo.
Riesgo 10	Otras aplicaciones mejor procesadas que compitan con Smart Closet.

Tabla 13. Riesgos del proyecto

2. Análisis cualitativo y cuantitativo de los riesgos

En esta actividad se toman los riesgos identificados anteriormente, se clasifican dependiendo del tipo de riesgo en las categorías propuestas por Sommerville y realiza un juicio acerca de la probabilidad y gravedad de dicho riesgo [8]:

Riesgo	Clasificación	Probabilidad	Impacto
R1	Tecnológicos	Poco probable	Alto
R2	Requisitos	Poco probable	Muy alto
R3	Estimación	Posible	Muy alto
R4	Organizacional	Muy raro	Muy alto
R5	Herramientas	Probable	Bajo
R6	Personal	Poco probable	Alto
R7	Organizacional	Poco probable	Medio
R8	Tecnológicos	Muy raro	Muy alto
R9	Personal	Poco probable	Medio

R10	Herramientas	Posible	Medio
-----	--------------	---------	-------

Tabla 14. Clasificación de riesgos

Luego de identificar, clasificar y valorar cada uno de los riesgos se realizó una matriz para priorizarlos a partir del impacto y la ocurrencia de cada uno y al final tomar los cinco más importantes. Esta matriz se puede ver en el [anexo 4. Gestión del riesgo](#).

Riesgos priorizados:

- R3: Subestimación de la complejidad de la aplicación.
- R2: Estructura de estándares de moda mal definidos.
- R4: Capacitación insuficiente y restringida respecto a las herramientas a usar
- R8: Hardware limitado que no soporte la aplicación
- R1: Funcionamiento incorrecto del API utilizada en la aplicación

3. Planeación de riesgos

En esta actividad se desarrollará una estrategia para manejar de manera correcta los riesgos priorizados o minimizar el impacto en caso de que alguno de estos logre materializarse, por tanto, a cada uno de los riesgos le hemos relacionado algunas salvaguardas [1].

Riesgo 3: Subestimación de la complejidad de la aplicación.

- Realizar levantamiento de requerimientos cuidadoso.
- Hacer un buen análisis y un buen diseño.
- Realizar tareas pequeñas que luego puedan ser integradas.
- Contar con un equipo funcional donde cada uno pueda aportar su talento de manera diferente y pro al proyecto.

Riesgo 2: Estructura de estándares de moda mal definidos

- Conocer por lo menos un experto en el tema quien pueda orientarnos en el tema específico del proyecto (moda).
- Investigar fuentes teóricas como artículos y/o revistas.
- Que por lo menos dos integrantes del grupo tengan conocimientos básicos de cómo usar las prendas.

Riesgo 4: Capacitación insuficiente y restringida respecto a las herramientas a usar

- Elección de herramientas de manejo sencillo e intuitivo.
- Contar con la documentación de cada una de las herramientas.
- Que al menos un integrante del grupo haya usado o sepa cómo usar las herramientas.
- Disposición y agilidad para aprender.

Riesgo 8: Hardware limitado que no soporte la aplicación

- Desarrollar la aplicación cumpliendo con la portabilidad y sin exigencia de tantos recursos.
- Probar la aplicación en al menos tres dispositivos con diferente capacidad con el fin de evaluar el soporte dependiendo de la máquina.

Riesgo 1: Funcionamiento incorrecto del API utilizada en la aplicación

- Contemplar y evaluar más de una opción de API.
- Realizar diferentes pruebas comprobando la mayor cantidad de temperaturas en cuanto sea posible.
- Poner en la aplicación la capacidad de escoger el clima para evitar que alguna falla estropee el funcionamiento general de la misma.

4. Monitoreo de riesgos

Esta actividad se realizará por cada iteración y se harán dos actividades para este monitoreo [1]. El director y el grupo revisarán el estado de los riesgos y evaluarán el estado. Posteriormente se evaluarán nuevamente cada uno de las salvaguardas relacionados dependiendo del cambio en el riesgo. Incluso si es necesario considerar nuevos riesgos se tendrán en cuenta

El diagrama de plan de análisis y gestión de riesgos se encuentra a continuación.

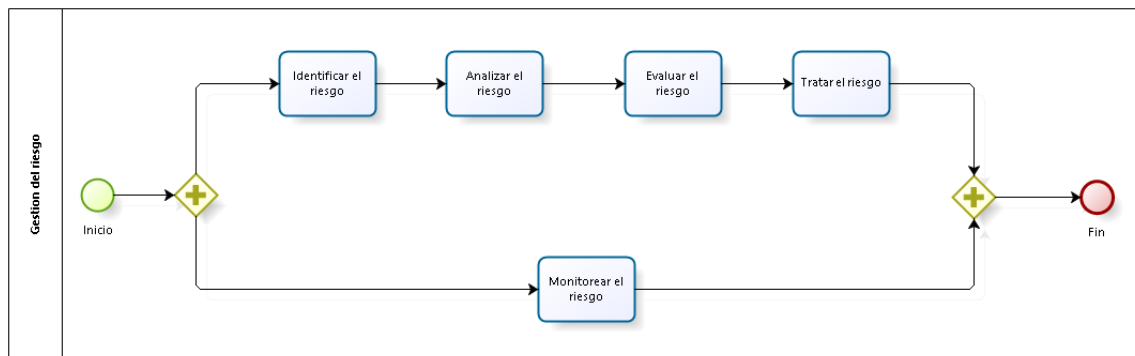


Ilustración 4. Diagrama BPMN para el plan de análisis y gestión de riesgos

12.3. Administración de Configuración y Documentación

En esta sección se pretende identificar los principales ítems de configuración y cómo estos se van a documentar, gestionar y evolucionar a lo largo del desarrollo del proyecto.

12.3.1. Involucrados

El principal involucrado para la administración de configuración es el jefe de documentación y el responsable de calidad. Estos se encargarán de hacer las revisiones pertinentes que requieran los documentos, registrar los cambios realizados en ellos y verificar que haya coherencia con respecto a lo planteado en este documento, en específico la [sección 12.4 Control de calidad](#).

12.3.2. Ítems de configuración

Durante el desarrollo del proyecto se realizarán varias versiones y se gestionará el estado actual de los siguientes ítems. Cabe aclarar que el manejo de versiones se controlara con dos herramientas. Para los ítems como documentos, anexos y diagramas se utilizará la herramienta de Google Drive. Para los ítems enfocados a la aplicación, como lo son los códigos fuentes de los prototipos, se utilizará la herramienta GitHub y su interfaz Gitkraken:

Ítem	Descripción	Etapas de desarrollo
Software Project Managment Plan (SPMP)	Se detallan los procesos, responsabilidades, roles, calendarios, herramientas, métodos y procedimientos para administrar todo el proceso de desarrollo de software.	Primera Etapa
Software Requirements Specification (SRS)	Se describen los requisitos funcionales y no funcionales del software, así como las restricciones de diseño.	Segunda Etapa
Software Design Document (SDD)	Registra información de diseño y arquitectura que se utilizará en el desarrollo del producto.	Tercera Etapa
Diagramas del sistema	Todo lo relacionado con los diagramas de visualización que tendrá el sistema.	Segunda etapa
Código fuente de los prototipos	El desarrollo del código para la implementación del software y toda la documentación que surge de este	Segunda y tercera etapa
Repositorios	Todo el conjunto de carpetas que contengan alguna parte del desarrollo del software	Todas las etapas

Tabla 15. Ítems de configuración

Los cambios que se deseen efectuar a cualquiera de los ítems de la tabla anterior deberán pasar por un proceso para verificar si es estrictamente necesario. El proceso se presenta a continuación:

- Se debe notificar al encargado o líder del proyecto las razones por las cual se quiere realizar dicho cambio al ítem.
- El líder del proyecto revisa la necesidad del cambio que se desea efectuar y si este tiene repercusiones negativas en el proceso de desarrollo.
- El líder luego de evaluar la propuesta reúne al equipo de trabajo para socializar más a fondo el cambio que se quiere realizar.
- Si en la reunión la propuesta es rechazada, termina el ciclo de cambio.
- Si en la reunión la propuesta es aceptada, se procede a realizar dicho cambio en el ítem pertinente y se da fin a la reunión.

Este proceso está representado en el siguiente diagrama BPMN.

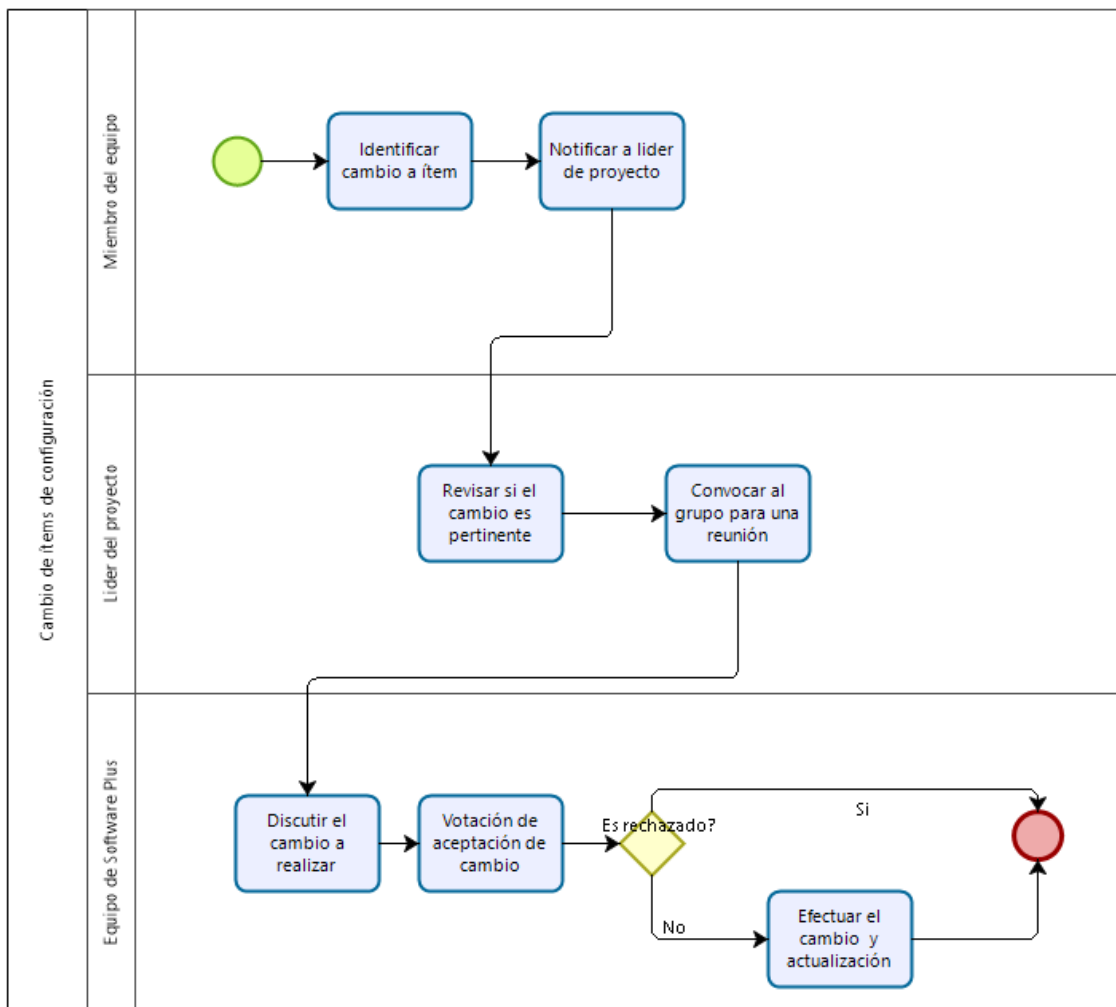


Ilustración 5. Diagrama BPMN para proceso de cambio de un ítem de configuración

12.4. Control de Calidad

12.4.1. Roles en el control de calidad

A continuación, se presentan los roles que se tendrán para el control de calidad del proyecto y el integrante del grupo que es responsable de dicho rol.

Rol	Responsable	Descripción	Actividades
Director(a) de Calidad	Mónica Álvarez	Persona encargada de establecer las pautas que se deben tener en cuenta al momento de realizar un estudio de Calidad respecto a una parte del proyecto en específico.	<ul style="list-style-type: none">• Monitoreo constante sobre cada uno de los responsables de la gestión de la calidad.• Identificar estándares de calidad en los cuales se basará el proyecto.• Solicitar reporte a cada examinador sobre el progreso de cierta actividad.• Asistencia en cualquier sección del proyecto al momento de realizar una inspección.
Examinador(a) del Documento	Laura Jiménez	Persona que se encarga de revisar constantemente el documento cada que se realiza un cambio sobre este.	<ul style="list-style-type: none">• Evaluar cada cambio realizado al documento y aprobar o rechazar este cambio.• Comunicar constantemente al equipo de trabajo cuáles son las secciones del documento en las cuales se necesita más profundidad y mayor trabajo.• Gestionar de manera correcta la estructura y referenciación del documento.
Examinador del Código	Juan Camilo Chafloque, Sergio Posada	Encargados de la revisión constante de cada cambio que realice cualquier integrante sobre el código fuente del proyecto.	<ul style="list-style-type: none">• Crear el repositorio local en el computador que va a servir como base de operaciones.• Revisar cada cambio que se realice por parte de cualquier integrante del grupo de trabajo.• Según la revisión aceptar o no el cambio

			que se desea llevar a cabo. Todo esto por la herramienta de GitHub.
Tester	Santiago Palacios, Sebastián Gutierrez	Personas encargadas de realizar todo tipo de pruebas sobre el código fuente con el fin de encontrar posibles errores.	<ul style="list-style-type: none"> • Encargados de buscar errores en el código fuente por medio de las diferentes pruebas sobre este. • Socialización con el equipo de trabajo sobre cualquier tipo de error encontrado en el código. • Llevar el sistema a sus límites por medio de diferentes pruebas con distinta complejidad.

Tabla 16. Roles en el control de calidad

12.4.2. Calidad del documento

La calidad del documento se basará en la plantilla principal que se facilitó al grupo de trabajo por parte de la profesora. Dentro de cada una de las secciones se tendrán en cuenta los siguientes elementos:

- **Estructura de texto:** Una sección está estructurada de manera correcta respecto a las pautas del documento.
- **Coherencia:** El texto se entiende de forma sencilla, es fácil de leer para el lector y consecuente con lo que se está planteando en cada sección.
- **Ortografía:** Se verifica un uso correcto de tildes, acentos, mayúsculas y signos de puntuación.
- **Referenciación:** Se verifica que los párrafos, secciones o citas estén referenciados correctamente de acuerdo a la IEEE.

Una vez se termine una tarea específica relacionada con algún ítem del documento, o un ítem externo (Anexos y diagramas) se inicia el proceso de verificación. Una vez se empieza el proceso de verificación, la tarea es revisada por el responsable de calidad para ver si cumple con los requisitos que esta debe tener. Si se cree que la tarea realizada no cumple los requisitos, se convoca una reunión grupal y entre todos los miembros del equipo se revisa nuevamente la sección y se realizan los cambios que se consideren necesarios. Si se cree que la tarea realizada cumple con los requisitos, se agrega al documento final y se actualiza el historial de cambios.

A continuación, se presenta el proceso a seguir para la verificación de la calidad del documento.

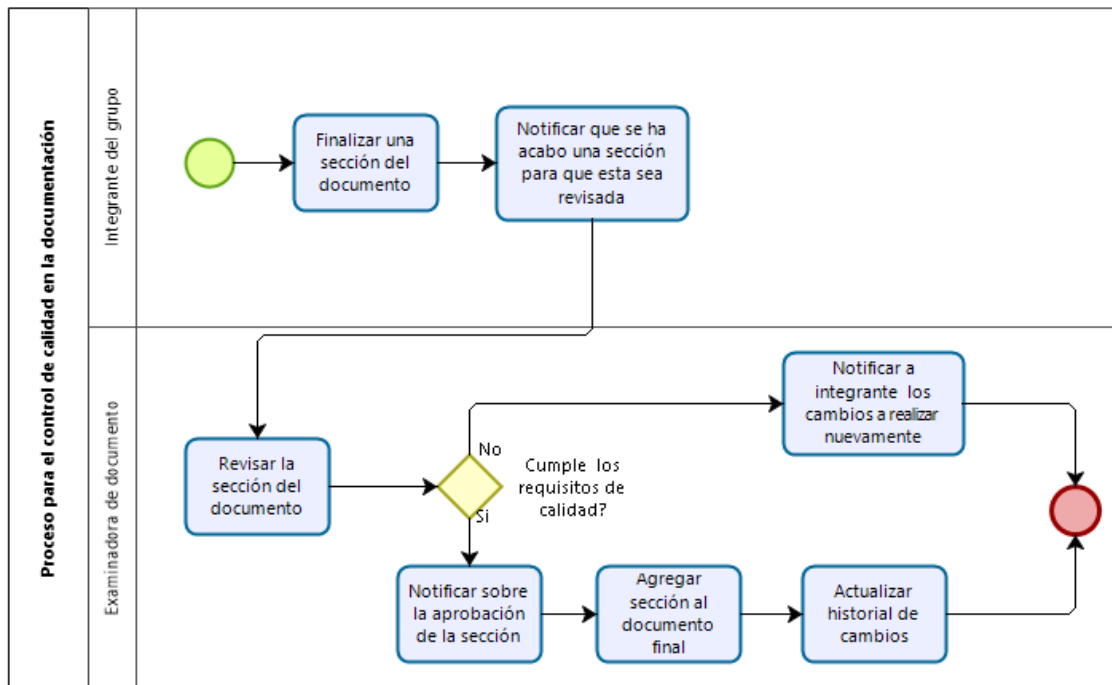


Ilustración 6. Diagrama BPMN del proceso de revisión de calidad en documentos

12.4.3. Pruebas del software

En todo proceso de desarrollo de aplicaciones es indispensable la presencia de un proceso de pruebas de software que coexista y se integre con este primero para garantizar así el buen funcionamiento y la calidad del producto final [36].

Con esto, con el fin de que una prueba arroje los resultados esperados sobre un código es necesario estudiar la mejor forma de hacerlo siguiendo los pasos de especialistas en este tema, según [36] los objetivos de estas pruebas de software son los siguientes:

- Probar si el software no hace lo que debe.
- Probar si el software hace lo que no debe, es decir, si provoca efectos secundarios.
- Descubrir un error que aún no ha sido descubierto.
- Encontrar el mayor número de errores con la menor cantidad de tiempo y esfuerzo.
- Mostrar hasta qué punto las funciones del software operan de acuerdo con las especificaciones y requisitos del cliente.

Con el fin de lograr estos objetivos, se decidió implementar las Pruebas Unitarias y las Pruebas de Integración a la hora de realizar este seguimiento sobre el código. Las pruebas unitarias consisten en verificar el funcionamiento de los elementos del software que permitan revisarse por separado mientras que las pruebas de Integración consisten en evaluar las interacciones entre los componentes internos

del software después de haber realizado las pruebas unitarias, esta se centra en evaluar la comunicación entre cada uno de los componentes.

13. Referencias

- [1] 'IEEE 16326-2009 - ISO/IEC/IEEE International Standard - Systems and Software Engineering--Life Cycle Processes--Project Management'. [Online]. Available: <https://standards.ieee.org/standard/16326-2009.html>.
- [2] Rose, K. H. (2013). A Guide to the Project Management Body of Knowledge (PMBOK® Guide)—Fifth Edition. Project management journal, 44(3), e1-e1.
- [3] V. Rastogi, 'Software Development Life Cycle Models-Comparison, Consequences', 2014.
- [4] Bourque, P.; Fairley, R.E. (2014). "Guide to the Software Engineering Body of Knowledge (SWEBOK)". IEEE Computer Society.
- [5] I. Sommerville, Software Engineering: (Update). Old Tappan: Pearson Education UK, 2006.
- [6] '2017-Scrum-Guide-Spanish-European.pdf'
- [7] 'The Agile System Development Life Cycle (SDLC)'. [Online]. Available: <http://www.ambyssoft.com/essays/agileLifecycle.html>.
- [8] «IEEE Standard for Information Technology — Systems Design — Software Design Descriptions». IEEE. 20 de julio de 2009.
- [9] Agilemanifesto.org. (2020). Principios del Manifiesto Ágil. [online] Available at: <https://agilemanifesto.org/iso/es/principles.html> [Accessed 25 Feb. 2020].
- [10] Cantone, D. (2006). *Implementacion y debugging*. USERSHOP.
- [11] 'Enterprise Architect'. [Online]. Available: <https://sparxsystems.com/>
- [12] 'Business Process Model and Notation'. [Online]. Available: <https://www.bizagi.com/acerca-de>
- [13] 'Gloomaps'. [Online]. Available: <https://www.gloomaps.com/>
- [14] 'Team Gantt | Trello'. [Online]. Available: <https://trello.com/about>
- [15] 'HTML User Guide'. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5>
- [16] 'CSS User Guide'. [Online]. Available: <https://cssreference.io/>
- [17] 'Typescript'. [Online]. Available: <https://www.typescriptlang.org/docs/home.html>
- [18] 'Angular'. [Online]. Available: <https://angular.io/start>
- [19] 'Cloud Firestore'. [Online]. Available: <https://firebase.google.com/docs/firestore?hl=es-419>
- [20] 'GitHub'. [Online]. Available: <https://guides.github.com/>
- [21] 'GitKraken'. [Online]. Available: <https://www.gitkraken.com/customers>
- [22] 'Slack'. [Online]. Available: <https://slack.com/intl/es-co/features>
- [23] 'Visual Studio Code'. [Online]. Available: <https://code.visualstudio.com/docs>

- [24] 'PlanningPoker'. [Online]. Available: <https://www.planningpoker.com/about/>
- [25] Project Management 2020, 'The Project Management Plan (PMP)', 2020 Project Management. [Online]. Available: <https://2020projectmanagement.com/resources/project-planning/the-project-management-plan-pmp>.
- [26] Interaction Design Foundation, Mock-ups.
- [27] B. Bruegge and A. H. Dutoit, Object-oriented software engineering: using UML, patterns, and Java, 3rd ed. Boston: Prentice Hall, 2010.
- [28] T. Dingsøyr, 'Postmortem reviews: purpose and approaches in software engineering', Inf. Softw. Technol., vol. 47, no. 5, pp. 293–303, Mar. 2005.
- [29] 'Software Documentation'. [Online]. Available: <http://www.literateprogramming.com/documentation.pdf>
- [30] 'Technical Documentation in Software Development: Types, Best Practices, and Tools', AltexSoft.
- [31] N. Rozanski and E. Woods, Software systems architecture: working with stakeholders using viewpoints and perspectives. Upper Saddle River, NJ: Addison-Wesley, 2005.
- [32] 'Software Architecture vs Code'. [Online]. Available: https://gotocon.com/dl/goto-amsterdam-2014/slides/SimonBrown_SoftwareArchitectureVsCode.pdf
- [33] A. P. Daza, 'ANÁLISIS Y GESTIÓN DE REQUERIMIENTOS Y REQUISITOS DE LA SOLUCIÓN DE SOFTWARE PARA APOYAR EL PROCESO DE GESTIÓN DE NÓMINA EN LA UNIVERSIDAD DISTRITAL, SIGUIENDO LOS LINEAMIENTOS DEL PROCESO DE DESARROLLO OPENUP/OAS EN SUS FASES DE INICIO, ELABORACIÓN Y CONSTRUCCIÓN.'
- [34] IEEE, 'Systems and software engineering. Life cycle processes. Risk management', BSI British Standards.
- [35] Project Management Institute, Ed., A guide to the project management body of knowledge (PMBOK guide), Fifth edition. Newtown Square, Pennsylvania: Project Management Institute, Inc, 2013.
- [36] 'Pruebas de Calidad de Software - EcuRed, Ecured.cu, 2020'. [Online]. Available: https://www.ecured.cu/Pruebas_de_Calidad_de_Software.