

SPMP ZonaCultura



Visión del Producto

- Ofrecer a la comunidad estudiantil un portal donde cualquier persona interesada en el arte independiente pueda encontrar obras artísticas de su interés, opinar sobre ellas y adquirirlas, además de brindarle la oportunidad de compartir su talento y así darse a conocer, todo por medio de una página web.
- Ofrecer un portal fiable, esperando que la comunidad de usuarios esté en constante crecimiento, en donde el usuario pueda encontrar y contactar jóvenes con talentos y aptitudes que llegue a necesitar para los diversos proyectos que se proponga.



Propósito

- Brindar un espacio cultural a la comunidad, en donde las personas interesadas en el arte independiente podrán: encontrar diferentes alternativas de trabajos multimedia o manualidades, opinar sobre éstos, llegar a adquirirlos y, en caso que el usuario sea un creador de contenido, tener la oportunidad de compartirlos en la página web..



Alcance

- El sistema permitirá la compra de trabajos que pueden encontrarse en la página web.
- El sistema despliega la información de contacto de los diferentes artistas después de que un usuario ha pagado por una manualidad, para discutir los detalles del envío.
- El sistema permitirá subir trabajos multimedia.
- El sistema permitirá realizar donaciones a los creadores de contenido.

Objetivo General

Implementar una página web para promover una comunidad centrada en el arte independiente estudiantil.



Objetivos Especifico

- Ayudar a los estudiantes a promover su trabajo y talentos
- Proporcionar una plataforma donde encontrar y tener la posibilidad de adquirir estos trabajos.
- Promover el desarrollo de proyectos artísticos mediante el talento de los estudiantes.
- Darles la oportunidad a los estudiantes de conseguir ingresos adicionales vendiendo sus trabajos económicamente.
- Darle la oportunidad a los estudiantes de comprar los trabajos que deseen y de vender sus propios trabajos.
- Darle la oportunidad a la comunidad de la página web de donarle dinero a algún creador de contenido si desean apoyarlo, pero no desean adquirir sus trabajos.

Supuestos

- Se cuenta con las herramientas necesarias para la elaboración del proyecto.
- Todos los integrantes cumplen con sus asignaciones de la mejor manera posible.
- Hay usuarios que desean comprar y conocer información de obras artísticas.
- Se cobrará una cuota del 10% del precio de venta de un trabajo que se venda a través de la plataforma.
Se llegó a este número por medio de una sesión de scrum poker, utilizando la puntuación estándar.

Restricciones

- El proyecto se debe desarrollar en un periodo de 4 meses dividida en 3 entregas.
- Subir videos de más de 1 minuto de duración, que superen la resolución de 480 pixeles.
- Los usuarios sólo pueden subir un máximo de cinco trabajos (Se llegó a este número por medio de una sesión de scrum poker, utilizando la puntuación estándar). Si desean subir seis trabajos, o más, a la vez a la plataforma, la cuota a cobrar aumentará.
- Se tiene un rango de precios que los usuarios pueden asignar a sus trabajos: un mínimo de dos mil pesos (\$2.000) y un máximo de cien millones de pesos (\$100.000.000). Estos valores se pensaron para cubrir tantos ejemplos como fuera posible, desde microliteratura hasta pinturas de alta gama.

Glosario

- **Usuario:** Persona con un perfil en la página web, capaz de buscar trabajos, y calificarlos, o colaboradores en un catálogo a través de etiquetas.
- **Colaborador:** Usuario que denota a un **artista**, cuyo perfil cuenta con etiquetas relevantes que detallan qué habilidades artísticas posee. Se espera que otros usuarios se pongan en contacto con esta persona para trabajar en un proyecto. Incluye la definición de *usuario*.
- **Creador de Contenido:** Usuario que ha subido un trabajo a la plataforma. Este proceso también añade etiquetas relevantes a su perfil, basado en qué clase de trabajo subió (Ej: al subir un texto, se añade la etiqueta de *escritor* al perfil). Incluye las definiciones de *colaborador* y *usuario*.
- **Consumidor:** Usuario que ha comprado un trabajo multimedia por medio de la plataforma.
- **Interesado en Publicitarse:** Actor del modelo que quiere mostrar su publicidad en la página web.
- **Administrador de la Pagina:** Miembro del grupo *JaveCultura* encargado de revisar las solicitudes de publicación por parte de los interesados en publicitarse, seleccionar cuales son adecuadas y desplegarlas en la página.



7.1. Modelo de ciclo de vida

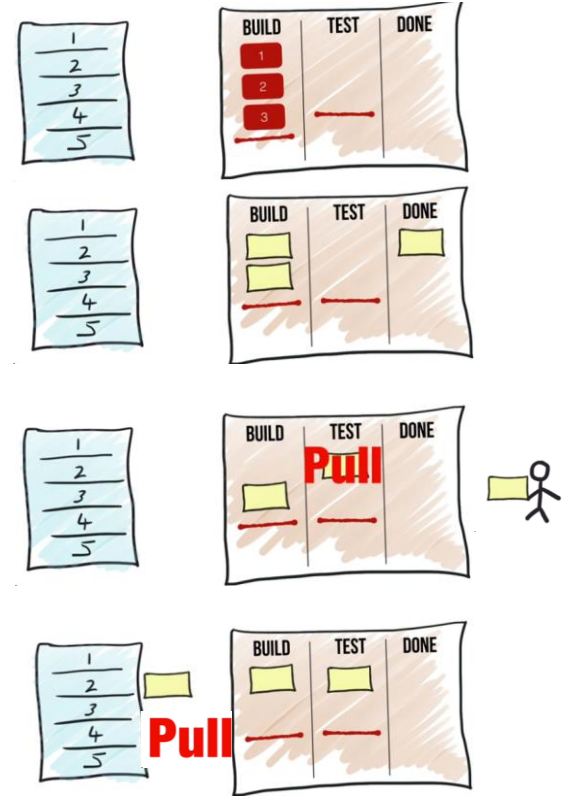
- Selección de la metodología del proyecto. Los candidatos para esto fueron:
 - Scrum
 - XP
 - Kanban
- Los criterios para la selección de la metodología fueron los siguientes:
 - **Capacidad de definir funcionalidades del sistema**
 - **Reuniones**
 - **Control del proceso**
 - **Separación de tareas**
 - **Control de entregas**
 - **Retroalimentación**
 - **Tiempo**

Modelo de ciclo de vida

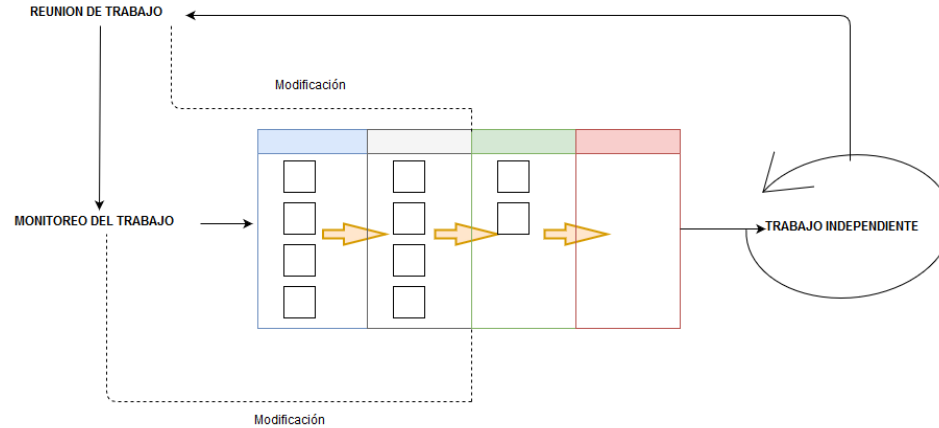
Criterio	Scrum	Kanban
Capacidad de definir funcionalidades del sistema	4	4
Reuniones	1	4
Control del proceso	3	3
Separación de tareas	4	4
Control de entregas	4	4
Tiempo	3	5
Retroalimentación	5	3
Total	24	27

Modelo de ciclo de vida

- ¿Por qué Kanban?
- Criterios fundamentales para esta decisión.
 - Tiempo
 - Reuniones
- Reuniones:
 - Habrá una reunión semanal presencial obligatoria para todos los miembros del equipo
- Se evita la sobrecarga de trabajo ya que se define un límite de tareas que circulan en el tablero, este número se da normalmente por el número de integrantes del equipo.
- Los cuellos de botella tienen una baja probabilidad de aparecer y adicionalmente todos los miembros del equipo de desarrollo estarán trabajando constantemente



Modelo de ciclo de vida



- En este ciclo de vida se tendrá una serie de listas definidas por el *Backlog*, los *To Do*, los *Doing* (en proceso) asignados a cada miembro, la lista *Review* (revisión) y *Done* (finalizado). Review tendrá integrada la funcionalidad de anunciar cuando un entregable esté listo para la revisión de calidad antes de ser entregado por completo.

7.2 Lenguajes y Herramientas

- **Criterios de Selección para los Lenguajes y Herramientas**
 - Las herramientas y lenguajes deben ser familiares, , o por lo menos no ser muy diferentes a otros que el grupo ha utilizado anteriormente.
 - El lenguaje de programación debe permitirnos utilizar un framework para desarrollo de páginas web y debe poseer una extensa cantidad de funcionalidades y librerías que permita desarrollar el proyecto con mayor flexibilidad.
 - Las herramientas y lenguajes que se utilizaran para el desarrollo de la página web deben tener un soporte para utilizar bases de datos.

7.2 Lenguajes y Herramientas

- **Criterios de Selección para los Lenguajes y Herramientas**
 - La comunicación entre las herramientas y lenguajes con la base de datos debe poder hacerse de manera automática.
 - El framework web debe permitir la implementación de una interfaz gráfica que permita una interacción más intuitiva con la página web.
 - Las herramientas, los lenguajes y el framework web deben tener soporte en varios sistemas operativos, como Windows 10 y la distribución Ubuntu de Linux.

Lenguajes y Herramientas Seleccionadas

- **Lenguaje de programación:** Python
 - Contiene un interpretador y una extensa variedad de librerías que están disponibles de forma gratuita.
 - Seleccionado por cumplir los criterios, y porque uno de los integrantes poseen experiencia previa en el lenguaje.
- **Framework:** Django
 - Framework Web.
 - Multitud de funcionalidades ya implementadas.
 - Soporta lenguajes de bases de datos como MongoDB, SQLite, y MySQL
 - Soporta HTML y CSS.
- **Lenguaje de bases de datos:** SQLite.
 - Lenguaje de bases de datos relacional.
 - Utilizado por la gran mayoría de computadores y teléfonos móviles del mundo.
 - Preinstalado en Django.

Lenguajes y Herramientas Seleccionadas

- **Lenguaje Complementario a Django: HTML.**
 - Lenguaje de etiquetas.
 - Facilidad de aprendizaje, estructuración de texto agradable y despliegue sencillo.
- **Criterios de Selección para la Herramienta de Control de Versiones:**
 - Herramienta conocida por el grupo.
 - Herramienta que tenga una curva de aprendizaje baja.
 - Debe permitir un control de versiones de código o documentos que ya se hayan modificado.
- **Herramienta Seleccionada: Git**
 - Cumple con todos los criterios.
 - Ya existe experiencia previa en clase.

Otras Herramientas

- **Herramienta para la Planeación del Proyecto:** Trello
 - El Scrum Board también se utiliza en Kanban.
 - Permite organizar los requerimientos del Product Owner y hacer un BackLog de las tareas.
 - Experiencia previa en clase.
- **Plataforma de Despliegue:** PythonAnywhere.
 - Hace servidores y contiene una base de datos en su versión gratuita.
 - Plataforma Web.
 - Compatible con Python, Django y SQLite.
 - Coopera con Github para realizar el despliegue.

7.4. Interfaces externas o Stakeholder

Entidad	Descripción	Responsabilidad
Carlos Parra y Camila Villarreal	Profesor y monitora de la asignatura Ingeniería de software.	Dar a conocer los fundamentos teóricos y guiar al equipo al desarrollo del software
Artistas	Estudiantes de la Pontificia Universidad Javeriana con contenido o aficiones artísticas	Subir contenido artístico a la plataforma.
Interesados en publicitarse	Agentes que desean promocionarse..	Contribuir económicamente a la plataforma.

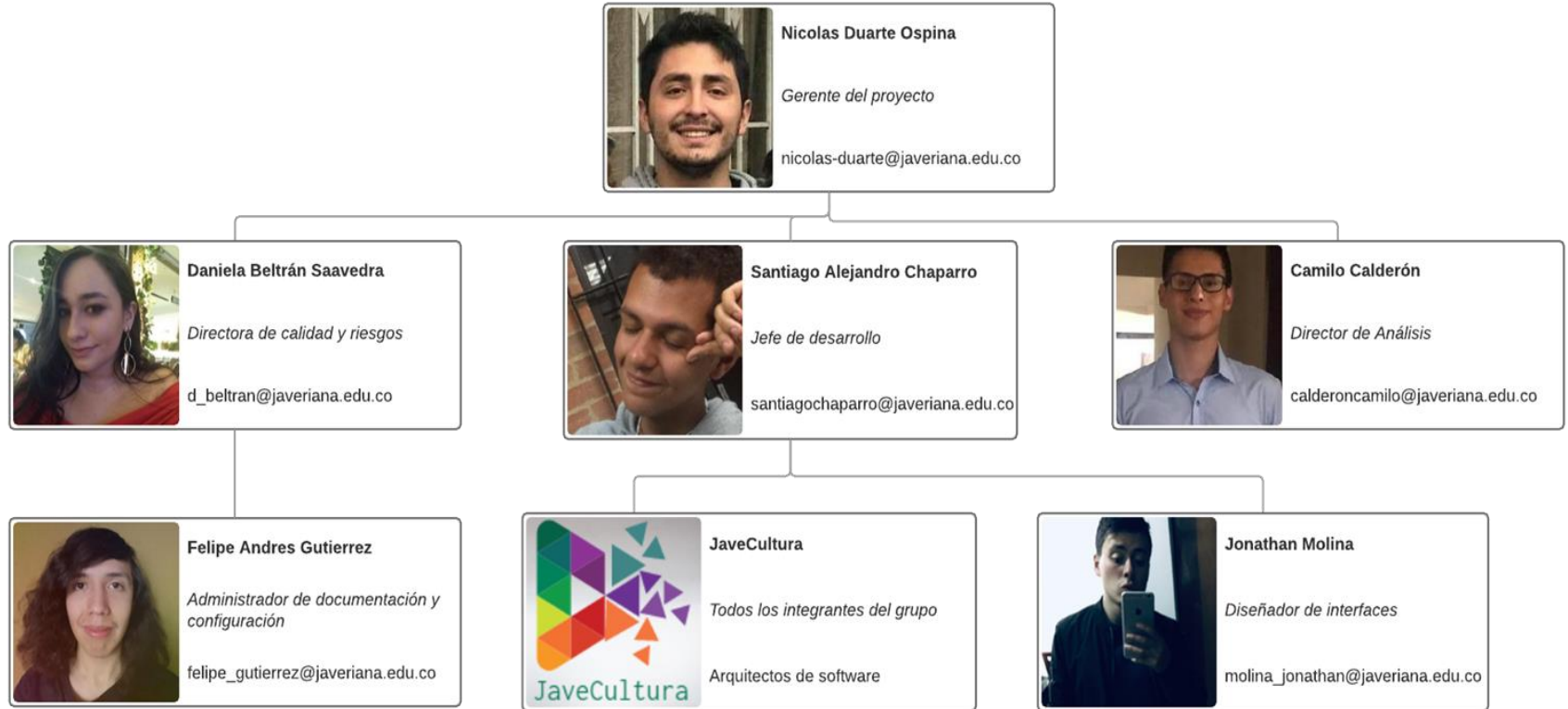
Interfaces externas o Stakeholder

Entidad	Descripción	Responsabilidad
Consumidores	Personas interesadas en arte.	Pagar por obtener un trabajo artístico.
Colaboradores	Personas interesadas en donar dinero a los artistas.	Contribuyen económicamente a los artistas para impulsar su trabajo.
Buscadores de aficionados artísticos	Personas interesadas en buscar artistas con aficiones..	Buscar aficionados artísticos en la plataforma.

Interfaces externas o Stakeholder

Entidad	Descripción
Django	Framework de programación web.
Pythonanywhere	Proveedor de servidor y base de datos

7.4. Organigrama



8.1. Estimaciones

- **Esfuerzo:** Conocer la dificultad de cada actividad utilizando un método de planeación ágil llamado “Planning Poker” utilizando la plataforma trello.

The image shows two Trello cards side-by-side. The left card is titled 'Crear una base de datos para almacenar toda la información que se va a manejar' and is in the 'To Do' list. It has 66 labels and a description: 'Una BD donde se pueda administrar toda la información que se va a manejar por los usuarios.' The right card is titled 'Compra' and is also in the 'To Do' list. It has 161 labels and a description: 'El usuario va a realizar una compra.' Below the description is a section titled 'Actividades' with a list of tasks, each with a checkbox. The tasks are: 'Crear el botón de realizar compra en la sección donde este el trabajo para la venta.', 'Crear una sección en la página donde estén los campos para llenar la información para realizar el pago con tarjeta de crédito.', 'Crear botón con el cual se autoriza y realiza la compra', 'Implementar transacción en la base de datos.', 'Si es un video o un libro cambiar permisos en la base de datos para permitir la vista y descarga respectivamente (después de una compra).', and 'Implementar la sección donde se verán los datos personales y de contacto del artista en caso de la compra de una manualidad.'

Crear una base de datos para almacenar toda la información que se va a manejar
en la lista [To Do](#)

ETIQUETAS
66 +

Descripción Editar

Una BD donde se pueda administrar toda la información que se va a manejar por los usuarios.

Añadir comentario

Compra
en la lista [To Do](#)

ETIQUETAS
161 +

Descripción Editar

El usuario va a realizar una compra.

☒ **Actividades** Eliminar

0%

- ☐ Crear el botón de realizar compra en la sección donde este el trabajo para la venta.
- ☐ Crear una sección en la página donde estén los campos para llenar la información para realizar el pago con tarjeta de crédito.
- ☐ Crear botón con el cual se autoriza y realiza la compra
- ☐ Implementar transacción en la base de datos.
- ☐ Si es un video o un libro cambiar permisos en la base de datos para permitir la vista y descarga respectivamente (después de una compra).
- ☐ Implementar la sección donde se verán los datos personales y de contacto del artista en caso de la compra de una manualidad.

✓ **Completado**

📎 Adjunto

Estimaciones

- **Tiempo:** Se realizó una estimación a través de un diagrama de Gantt, basándonos en el **Límite de trabajo en curso (WIP)**.

[illegible]

Estimaciones

- **Presupuesto:**
 - **Costos de esfuerzo:** Calculados en horas por persona, de acuerdo a la calendarización del proyecto y la planificación de actividades, se estableció una cantidad de horas mínimas de trabajo por integrante, con relación a la intensidad horaria de la asignatura.
 - **Costos de hardware y software:** Para el caso del desarrollo de este proyecto, el software que se utilizará será gratuito. En términos de hardware, cada miembro del equipo ya cuenta con su equipo personal el cual es suficiente para llevar a cabo el proyecto.

8.2 Inicio del Proyecto

- Es importante que todos los integrantes del grupo tengan un buen manejo de las herramientas ya antes mencionadas. Por esto se va a hacer una capacitación del grupo entero de cada una de las herramientas y lenguajes.
- En la tabla se muestra los responsables del método y del material de enseñanza de cada lenguaje y herramienta:

Lenguaje y herramienta	Encargado
Django	Santiago Chaparro
Phyton	Santiago Chaparro
HTML	Camilo Calderón y Felipe Gutiérrez
SQLite	Jonathan Molina y Nicolas Duarte
Git	Capacitación en la universidad
Trello	Daniela Beltran

Infraestructura

- **Hardware y Software**

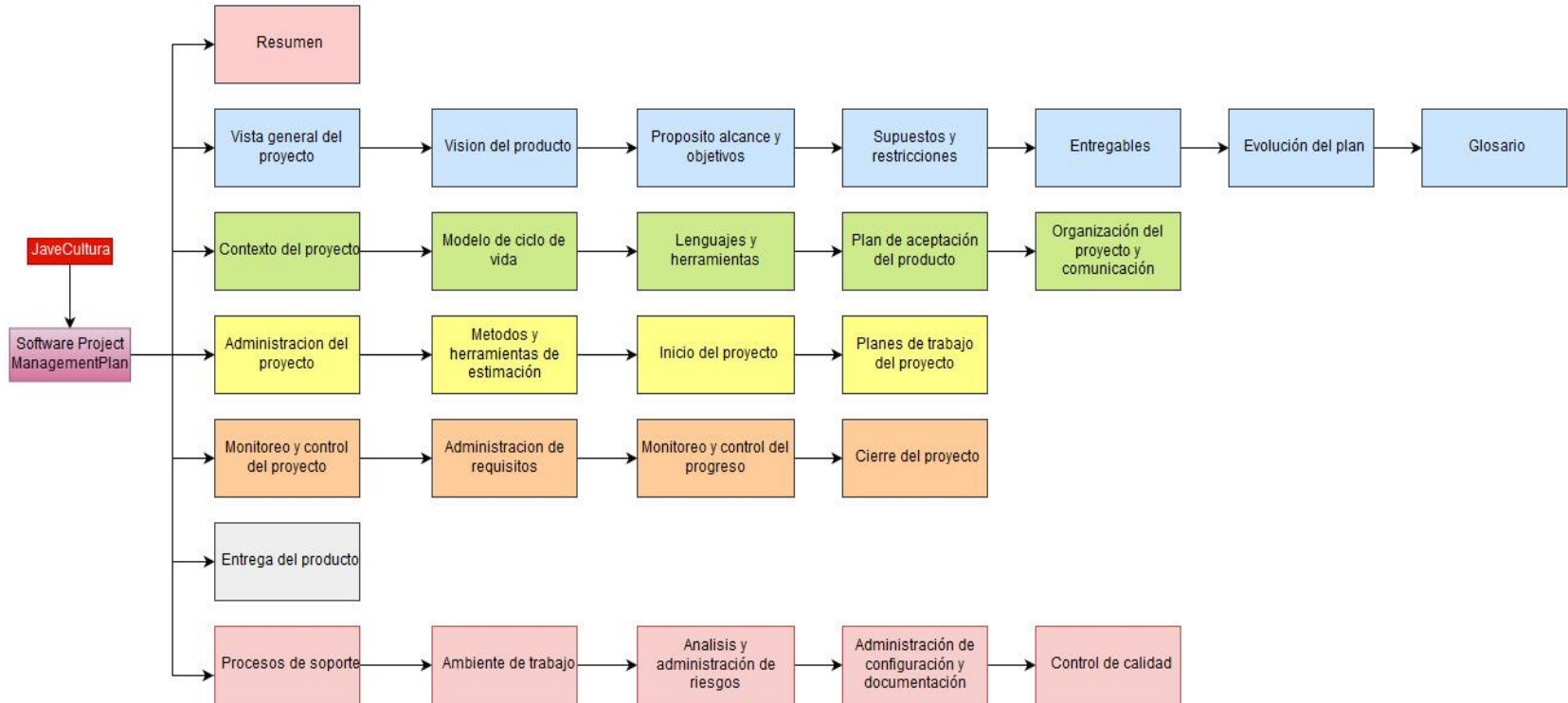
- Para el desarrollo, se necesitan equipos compatibles con las herramientas y lenguajes de la diapositiva anterior.
- El S.O puede ser Windows 7 o superior, o cualquier distribución de Ubuntu Linux.
- Para el procesamiento con Python, la pagina oficial de Intel www.software.intel.com recomienda un Intel® Core™ i3 processor como minimo.
- Para el despliegue del proyecto se utilizará la plataforma web, PythonAnywhere.
- En la siguiente diapositiva, se muestra una tabla con los datos de los equipos personales de cada integrante del grupo:

- **Infraestructura Fisica:**

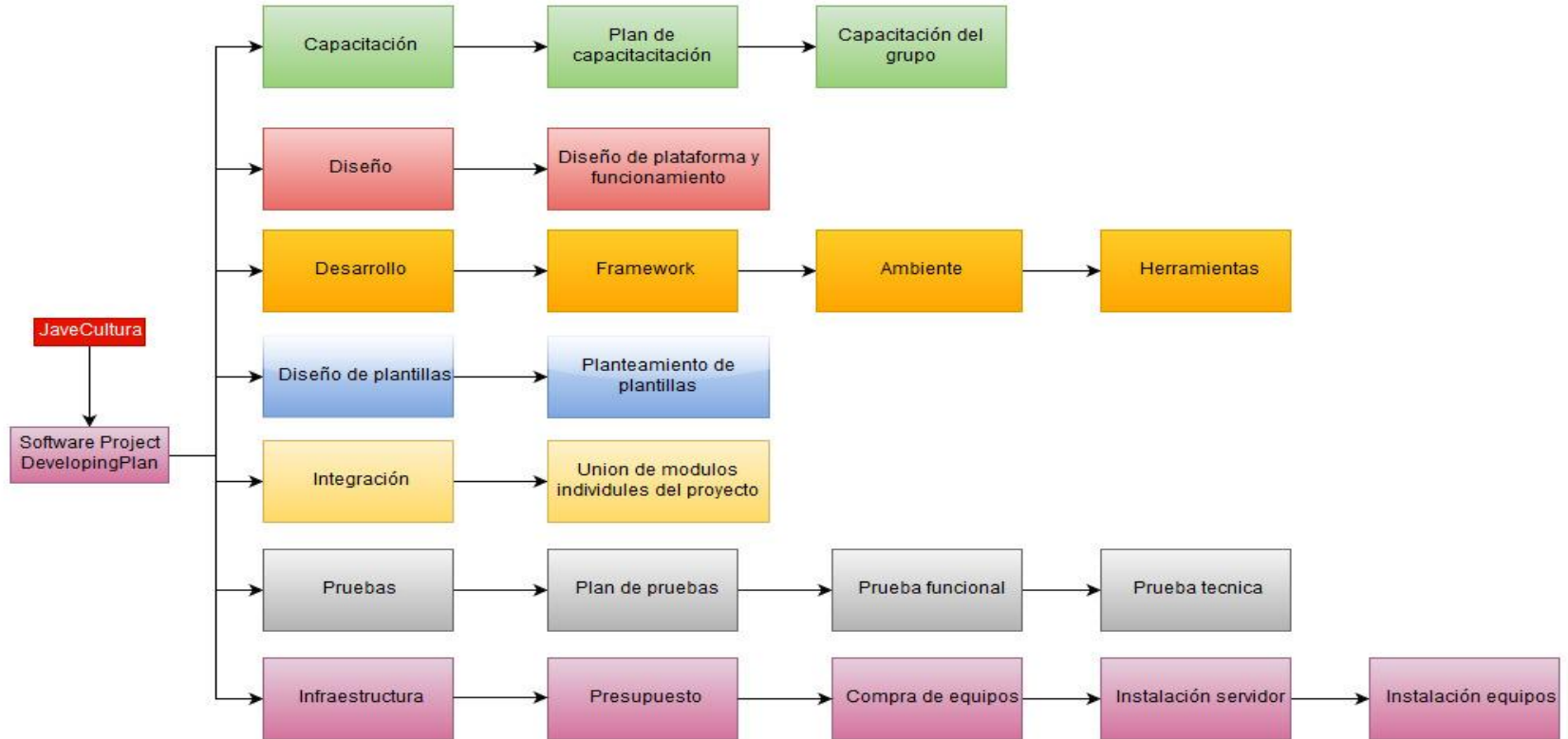
- Los espacios de reunión designados serán las salas de estudio de la Pontificia Universidad Javeriana.

Miembro	SO	Procesador	Ram	Costo Aproximado
Nicolás Duarte	Windows 10	Intel Core i5 - 7 Gen	8 GB	2.200.000
Camilo Calderon	Windows 10	Intel Core i5 - 8 Gen	8 GB	2.400.000
Santiago Chaparro	Windows 10	Intel Core i5 - 8 Gen	6 GB	1.800.000
Felipe Gutiérrez	Windows 7	Intel Core i3 - 1 Gen	4 GB	1.200.000
Daniela Beltran	Windows 10	Intel Core i7 - 5 Gen	4 GB	1.700.000
Jonathan Molina	Windows 10	Intel Core i7 - 5 Gen	12 GB	1.500.000

8.3. Planes de trabajo del proyecto



Planes de trabajo del proyecto



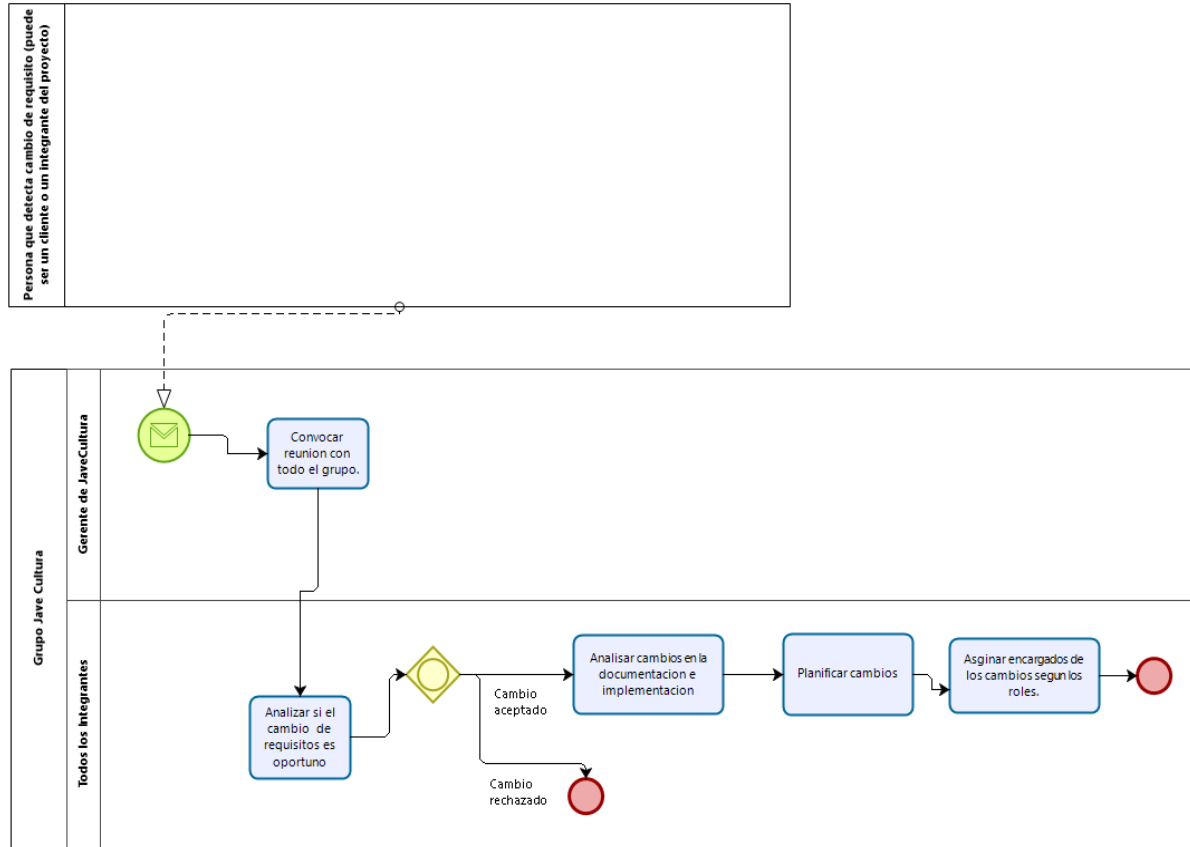
Presupuesto del Proyecto

- **Salario promedio de Ingeniero de Sistemas**
 - En co.Indeed.com se registra que el salario promedio de un Ingeniero de Sistemas es de 2.024.428 pesos al mes. Por lo que la hora de trabajo está a 8.435 pesos aproximadamente.
 - Si se tiene en cuenta que en una semana los integrantes de JaveCultura trabajaran 35 horas cada uno, y que el tiempo de realizacion de proyecto es de 16 semanas, se estima que el gasto total en salario es de: 28.341.600 pesos .
- **Gastos de transporte**
 - Para el traslado al lugar de las reuniones semanales se debe tener en cuenta el gasto de cada uno de los integrantes. El gasto en el transporte de ida y vuelta se refleja en las siguiente diapositiva.
- **Gastos de hardware**
 - A partir de los datos recolectados en la pasada sección se calculó que el costo en hardware tiene un total de 10.350.000 de pesos.

Nombre del Integrante	Gasto diario del transporte (total)
Santiago Chaparro	4800
Nicolas Duarte	4800
Daniela Beltran	20000
Felipe Gutiérrez	8000
Camilo Calderon	5000
Jonathan Molina	4800
Total de costo potencial en el semestre	758.400

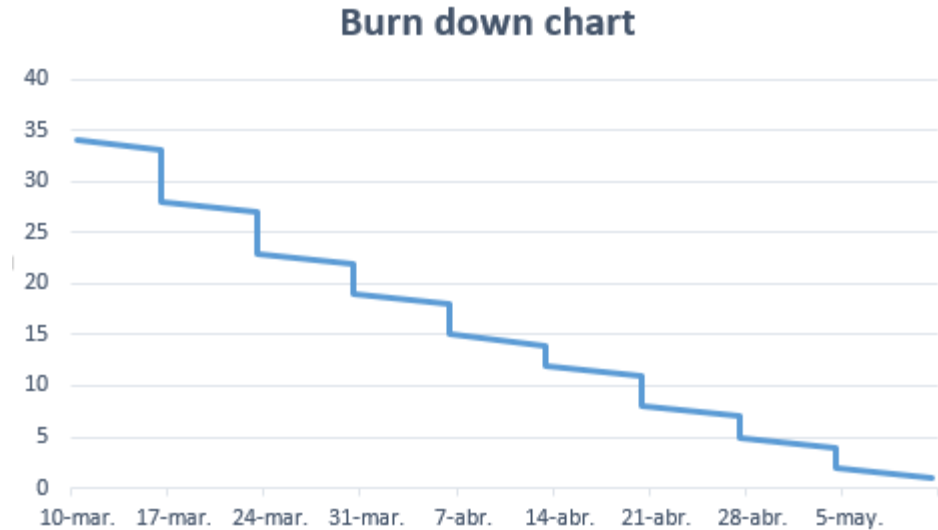
Salario de los integrantes	28.341.600
Costos de Hardware y Software	10.350.000
Costo del transporte	758.400
Total	39.450.200

9.1 Administración de Requisitos



9.2. Monitoreo y control del progreso

- De acuerdo a la metodología Kanban, y sus reglas, se establecieron dos unidades de medida para el progreso del proyecto:
 - Cantidad de tareas faltantes.
 - Tiempo de desarrollo por integrante.



9.3. Cierre del proyecto

- Reunión el día domingo donde se revisan los avances y la funcionalidad de las tareas de cada integrante.
- Todos los integrantes deben probar dichas funcionalidades hasta la fecha.
- Al finalizar el proyecto todos los integrantes deben realizar una revisión a todos los criterios establecidos para la satisfacción del proyecto y rendir un informe detallado.
- Se realizará un análisis post-modern

11.2 Análisis y administración de riesgos

Herramientas:

- Tabulación de identificación de riesgo (resultado de proceso de identificación).
- Análisis cuantitativo.
- Análisis cualitativo

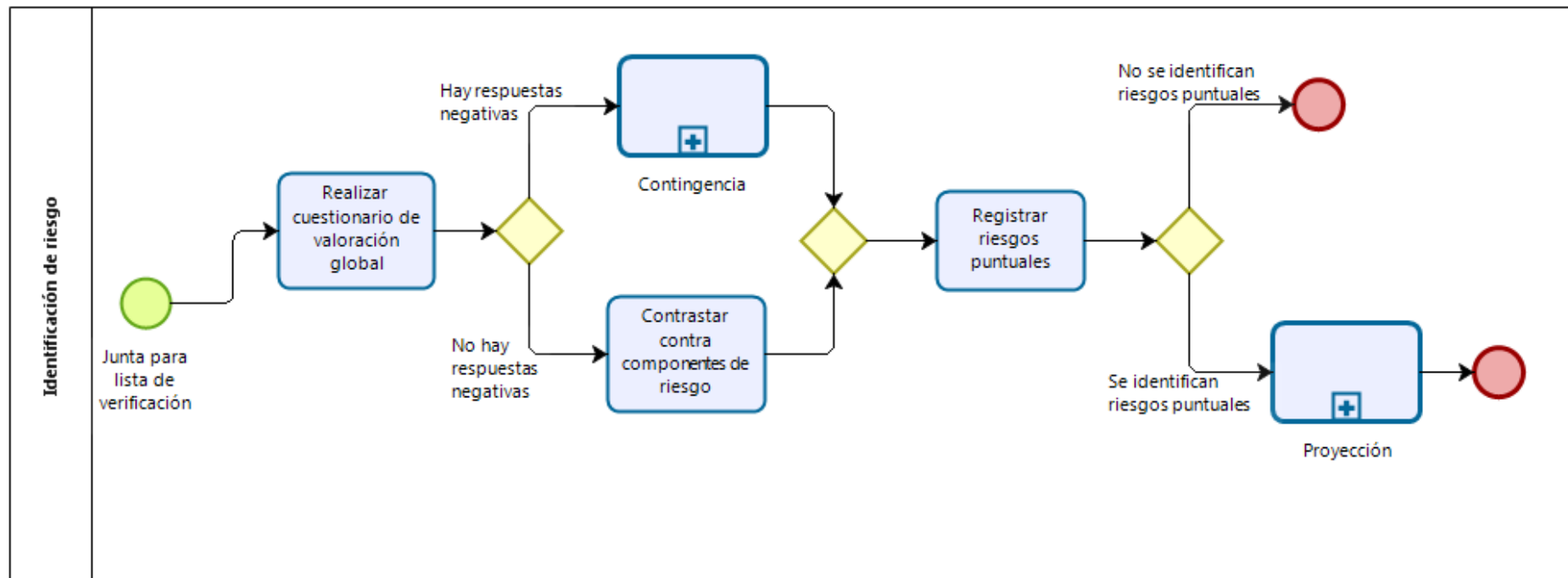
Los análisis están planeados; se harán a partir de la segunda entrega.

Categorías de riesgo expresadas a través de un RBS.

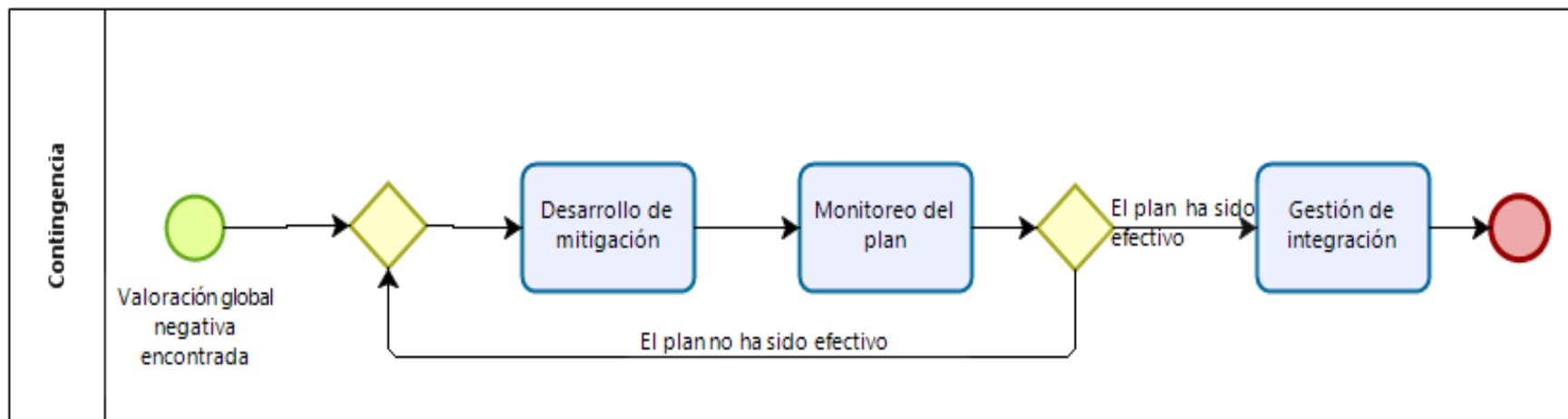


Riesgos	Probabilidad	Impacto
El diseño del proyecto no es consistente	30%	1
La funcionalidad requerida no se puede implementar	20%	1
Entregas de funcionalidades fuera del cronograma	35%	2
No hubo comunicación entre el equipo de desarrollo para cierta funcionalidad	30%	2
Cierta funcionalidad no cumple con la calidad requerida	25%	2
A los usuarios no les interesa comprar o buscar trabajos multimedia	20%	2
Un miembro del equipo no cumple con su trabajo	20%	2
Cambio de tecnología en el mercado	20%	2
Un miembro del equipo renuncia a su labor	10%	2
Nueva licencia para el software libre que se estaba usando	15%	2
La estimación de trabajo no fue la adecuada	40%	3
La estimación de tiempos no fue la adecuada	30%	3
Los equipos para el trabajo se dañan	20%	3
El proceso de control no detectó algún problema	10%	2
La tecnología escogida no es la adecuada para los objetivos del proyecto	15%	3
El gobierno prohíbe el desarrollo de páginas de cultura	1%	2

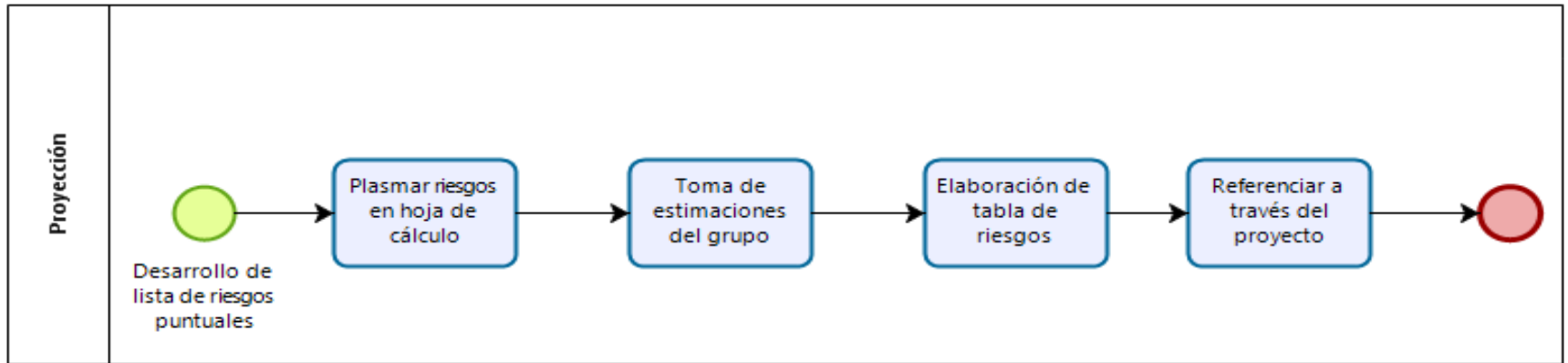
Proceso de identificación de riesgo



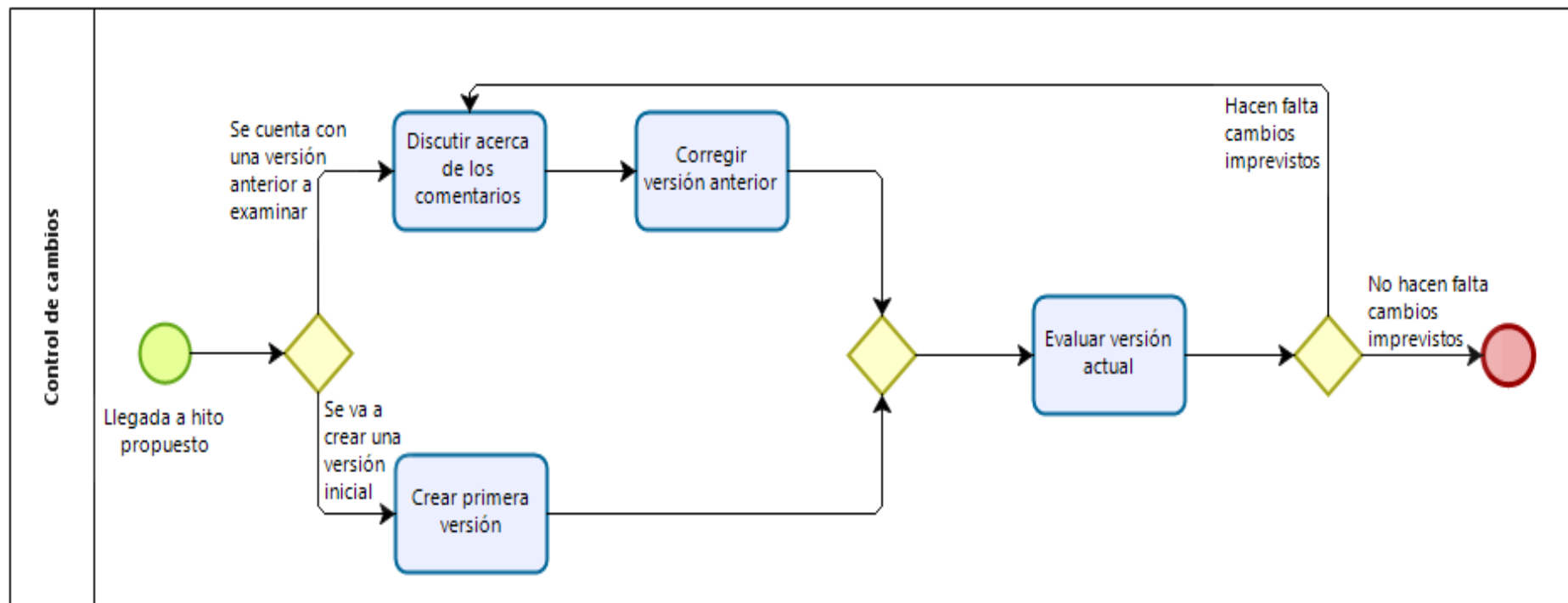
Sub-proceso contingencia



Sub-proceso proyección



Proceso de control de cambios



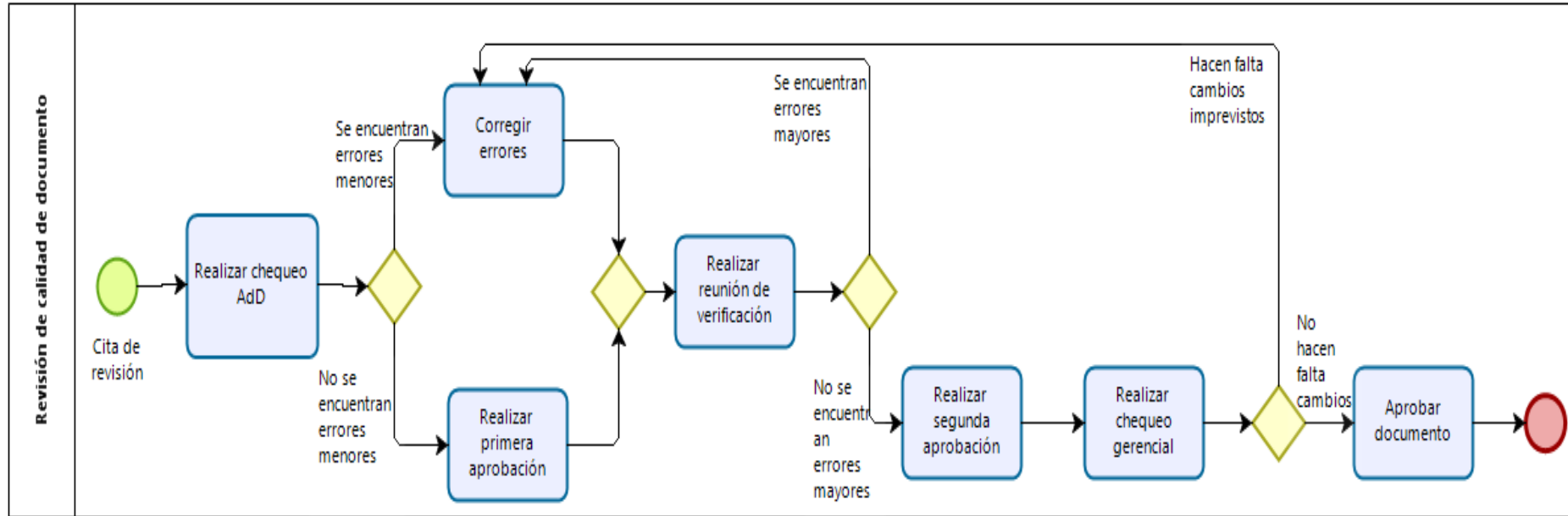
Artefactos de documentación

- **Control de versiones de documento:** Manejo de las diferentes versiones de los documentos. La primera versión se hizo tras la reunión #4. Cada actualización se hará tras las retroalimentaciones del profesor.
- **Documentación:** Explicación del código, pensada para desarrolladores ajenos al proyecto. La primera versión se hará tras iniciar la segunda entrega. Cada actualización se hará al finalizar un ciclo de actividad.

Artefactos de documentación

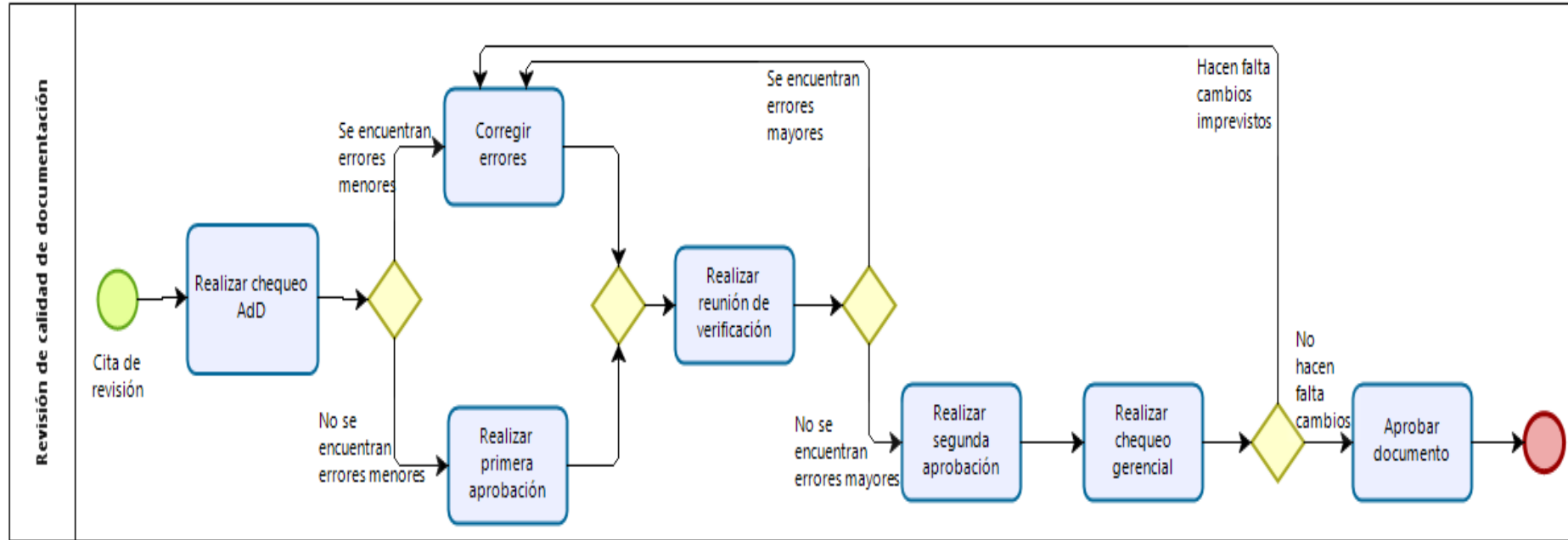
- **Manuales:** Instrucciones para los usuarios. La primera versión se hará junto al primer prototipo. La siguiente actualización se hará junto al segundo prototipo.
- **Repositorio:** Estructura de datos donde almacenar diferentes versiones de código. La primera versión se hará tras iniciar la segunda entrega. Cada actualización se hará al finalizar un ciclo de actividad.

Proceso de control de calidad del documento



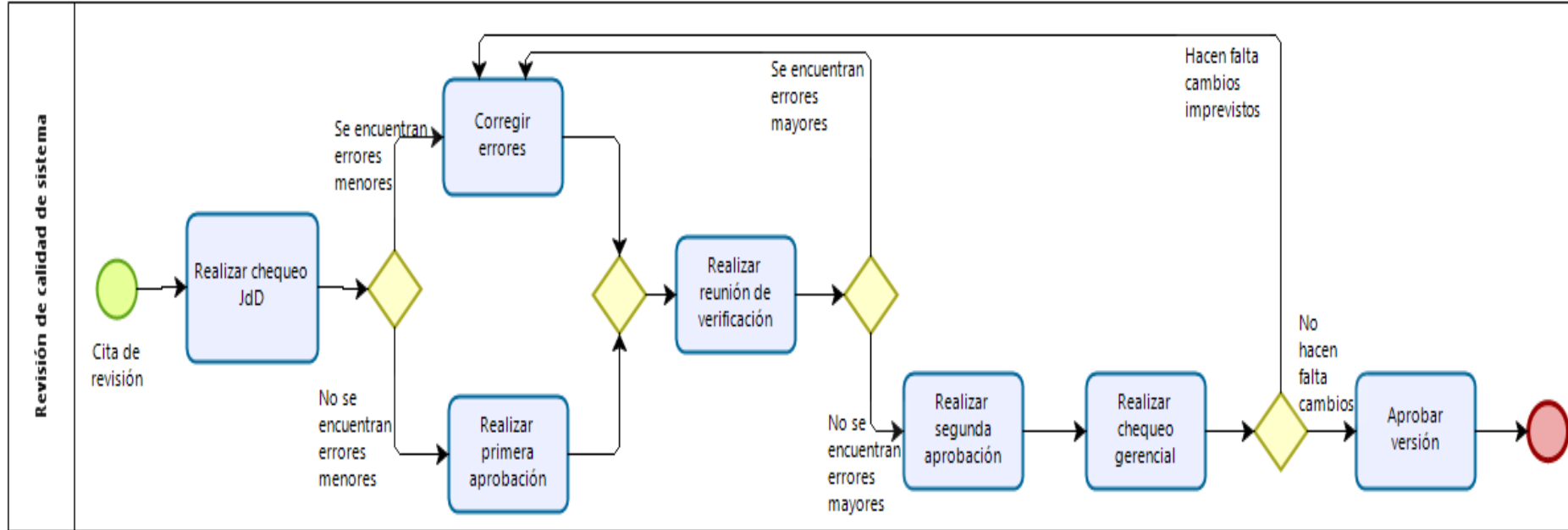
Chequeo conlleva ortografía, redacción y, posiblemente, comentarios sobre ideas que puedan clarificarse o cambiarse.

Proceso de control de calidad de documentación



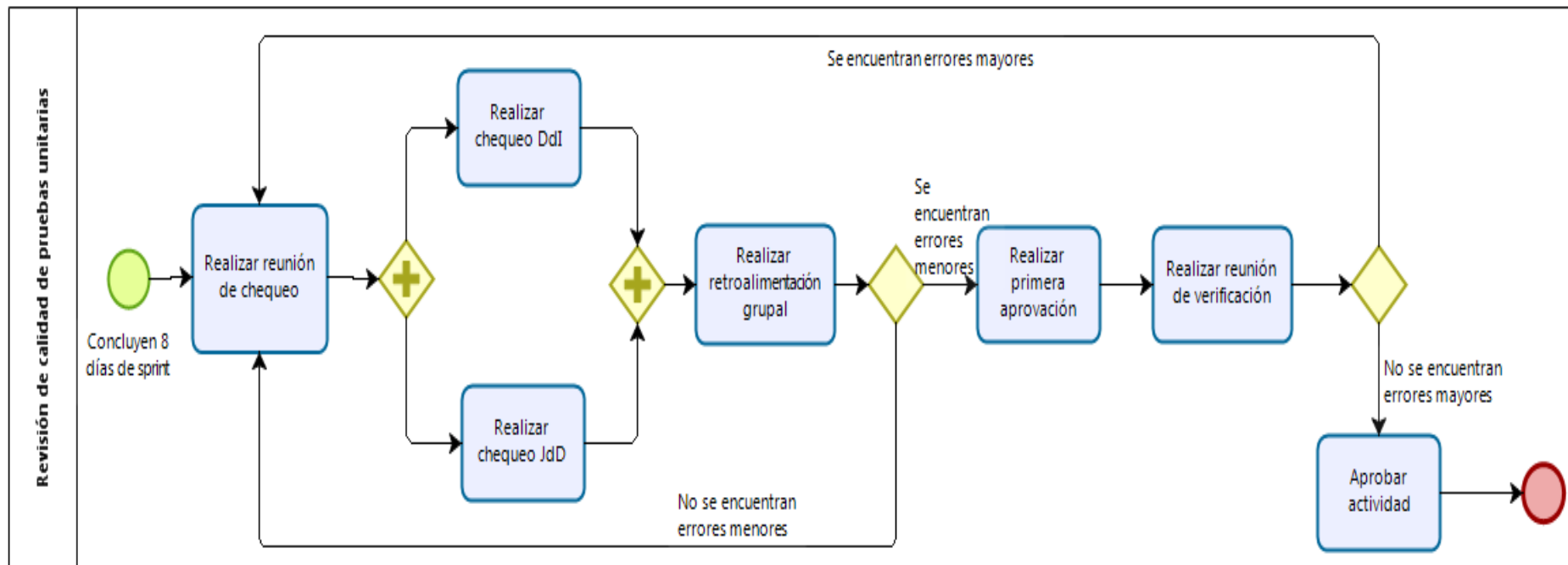
Chequeo conlleva verificar qué tan detallados, instructivos y bien redactados son los comentarios que acompañan el código. Si es necesario, se discutirán en una reunión.

Proceso de control de calidad del sistema



Chequeo conlleva probar qué tan sencillo es integrar el código, buscar errores de compilación, ejecución y posibles errores que deban controlarse y la efectividad de cada función. De ser necesario, estos temas se discutirán.

Proceso de control de calidad de pruebas unitarias



Chequeo conlleva verificaciones de correcta funcionalidad, tanto en bajo como en alto nivel, para discutir cómo van las pruebas del proyecto hasta el momento.