



UNIVERSIDAD AUTÓNOMA DE ENCARNACIÓN



Proyecto Carpintería

Módulo Almacén

Carrera: Lic. En Análisis de Sistemas Informáticos

Materia: Ingeniería de Software

Docente: Ing. Hugo Sendoa

Integrantes:

- Fabián Giménez
- Mario Segovia
- Astri Müller
- Rafael Escobar
- Ariel Roa
- Marcos Bordon
- Jose Boronowski

CUARTO CURSO SÉPTIMO SEMESTRE

ITAPÚA - PARAGUAY

AÑO: 2020.

Índice

Objetivo	4
Metodología	4
Ámbito del proyecto	5
A- Alcance	5
1.Módulo de Almacén:	5
1.1.Sub módulo de ítem:	5
1.2.Sub módulo de Categoría:	6
1.3.Sub módulo de registro de modificaciones de Ítem:	6
1.4.Reportes:	6
B- Factibilidad	6
1.Técnica	6
2.Operativa	6
3.Económica	6
C- Definición de recursos disponibles	7
E-Calendarización de tareas	8
Análisis de Negocios del Modulo de Almacen	8
A-Modelo de Negocio	9
1.Actores	10
2.Casos de usos conocidos	10
3.Modelo de Casos de Uso	11
4.Modelo de Casos de Uso - Requisitos	12
Diagramas	17
Diagrama de Clases	17
Diagrama de Paquetes	17
Diagrama de Estados	18
Diagrama de Secuencias	19
Diagrama de Actividades	20
Análisis De Riesgo	21
Gestión De Calidad	23
A-Propósito	23
B-Gestión	23
C-Organización	24
a. Team Leader, cumple las siguientes tareas:	24
b. Gestor de Documentación	24

c. Gestor de Riesgo	25
d. Gestor de Configuración	25
e. Gestor de Calidad	26
f. Desarrolladores	26
g. Tester	26
Gestión de La Configuración	28
A-Objetivo	28
B-Alcance	28
C- Organización	29
D-Responsabilidades	29
E. Distribución	30
Control de Configuración	30
Ítems de Configuración	30
Configuración de auditoría	31

Objetivo

En el marco de la materia de Ingeniería de Software se buscará aplicar conocimientos, técnicas y metodologías de desarrollo adquiridas durante años anteriormente cursados dentro de la formación de grado, para esto, se buscará desarrollar e implementar un Sistema de Gestión de Información que cumpla con las necesidades funcionales de una empresa del rubro de Carpintería. Para el efecto, se ha tomado un consenso general a fin de establecer la reglas, normas y limitaciones.

La implementación y desarrollo del sistema estará dividido en grupos de acuerdo a lo consensuado, siendo nuestro grupo el encargado del modulo de almacen.

Metodología

Rational unified process (RUP)

Es un producto del proceso de ingeniería de software que proporciona un enfoque disciplinado para asignar tareas y responsabilidades dentro de una organización del desarrollo. Su meta es asegurar la producción del software de alta calidad que resuelve las necesidades de los usuarios dentro de un presupuesto y tiempo establecidos.

La planeación de un proyecto de programación que pretende dar un beneficio en algún servicio ya existente o en uno nuevo requiere definitivamente de una buena organización. RUP es eficaz para atacar los asuntos de organización y de ejecución de un sistema de desarrollo de software. Ayuda a percibir el problema desde distintos puntos de vista y ahorra el tiempo de demora para un proyecto que tenga alguna continuidad y necesite modificaciones o adiciones.

Ámbito del proyecto

El ámbito del presente proyecto, consiste en una herramienta tecnológica destinada a un modelo de negocios para una carpintería. El mismo será de un sistema de Gestión de Información que busca crear una herramienta funcional y óptima para el modelo de negocios de una carpintería, se espera que tenga la posibilidad de manejar compras, producción, stock y venta de productos.

A- Alcance

Los módulos y submódulos a ser desarrollados por este grupo, serán:

1.Módulo de Almacén:

Gestionara los controles de stock de las materias primas, productos y el patrimonio.

1.1.Sub módulo de ítem:

Se encarga de registrar los productos, materias primas y patrimonio con sus respectivas descripciones.

1.2.Sub módulo de Categoría:

Definirá los tipos de Artículos, si estos pertenecen a materia prima, productos, patrimonio o otra categoría.

1.3.Sub módulo de Transferencia:

Se encargará de realizar el registro de los movimientos de entrada y salida de los ítems, y el origen y destinos de éstos.

1.4.Reportes:

- Reporte de productos en stock
- Reporte de materias primas
- Reporte de stock mínimo
- Reporte de Transferencias a producción

B- Factibilidad

1.Técnica

Actualmente se cuenta con conocimientos en las metodologías y técnicas de desarrollo, así mismo, se cuentan con las herramientas de hardware y software necesarios para la desarrollo e implementación de este proyecto.

2.Operativa

Desde el punto de vista operativo, la empresa cumple con los requisitos operativos necesarios para la implementación eficaz de la herramienta tecnológica, cuenta con operadores capacitados para las distintas áreas.

3.Económica

Para el grupo encargado de desarrollo e implementación de la herramienta tecnológica, es totalmente factible desde el punto de vista económico, llevar a cabo el proyecto. Se cuenta con recursos de software y hardware adecuados, así como los recursos necesarios para la logística en general.

En un reunión con la empresa, se constató que es factible para la misma, el desarrollo e implementación de la herramienta, teniendo en cuenta los factores que se requerirán para lograr el objetivo del mismo, como lo son adquisición de un servidor, máquinas destinadas a funcionar como estación de trabajo y una red estructurada para la conexión de la estaciones de trabajo con el servidor.

C- Definición de recursos disponibles

El grupo desarrollador posee todas las herramientas de hardware y software necesarios para la realización de la herramienta tecnológica, a continuación se menciona el hardware y software a ser usado por los mismos:

- Conectividad a Internet
- Software (Windows, Laravel, Php, Mysql, Versionador de código)

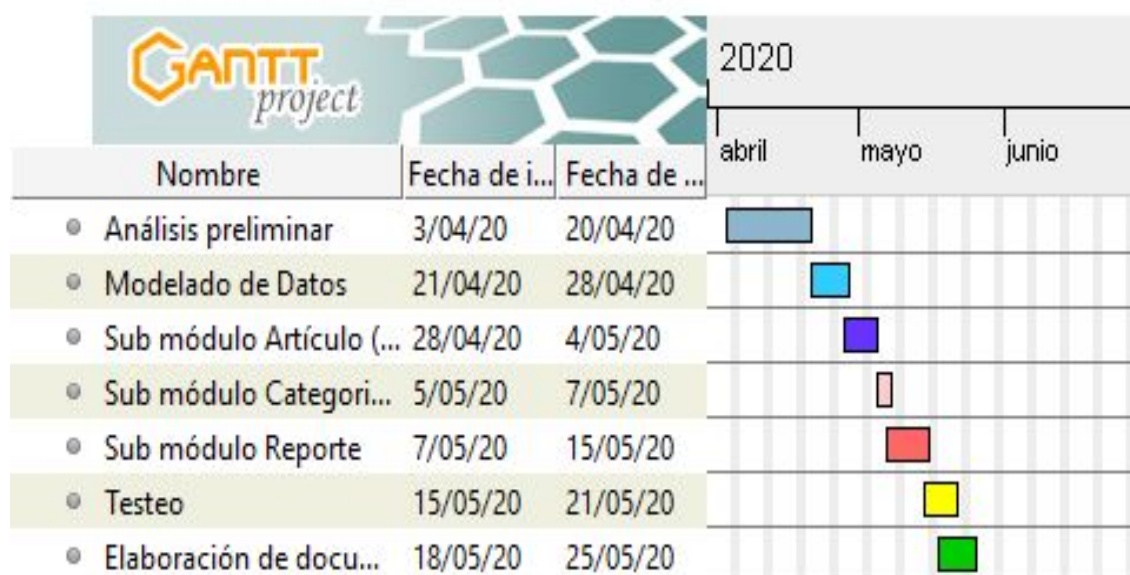
- Hardware (Notebooks)

La mayoría del software a ser usado es de carácter OPEN-SOURCE por lo cual no impactará en el presupuesto final de la herramienta. Lo mismo con el hardware debido a que el grupo desarrollador ya cuenta con los mismos.

Por parte de la empresa, la misma solo cuenta con una máquina, la cual cumple los requerimientos básicos necesarios para ser utilizada como estación de trabajo. La empresa deberá adquirir un servidor y las estaciones de trabajos restantes para todas las areas, así mismo como una red para la conexión de las mismas.

La empresa cuenta con suficientes recursos humanos disponibles, los mismos cumplen con ciertos requisitos, no obstante, deberán ser capacitados para el correcto y eficaz uso de la herramienta.

E-Calendarización de tareas



Análisis de Negocios del Modulo de Almacen

Debido a la necesidad y la problemática de la empresa de contar con una solución informática y de gestión de su información, es primordial según nuestro análisis del modelo de negocios sobre el cual funciona, que el mismo cuente con un modulo de almacen.

El módulo de almacén buscará llevar el inventario existente de las materias primas y productos finales para la compra, la venta y la producción. También buscará llevar el inventario del patrimonio de la empresa que comprende herramientas y artículos necesarios para su funcionamiento.

El módulo permitirá a la empresa crear los productos y estarán divididos por categorías.

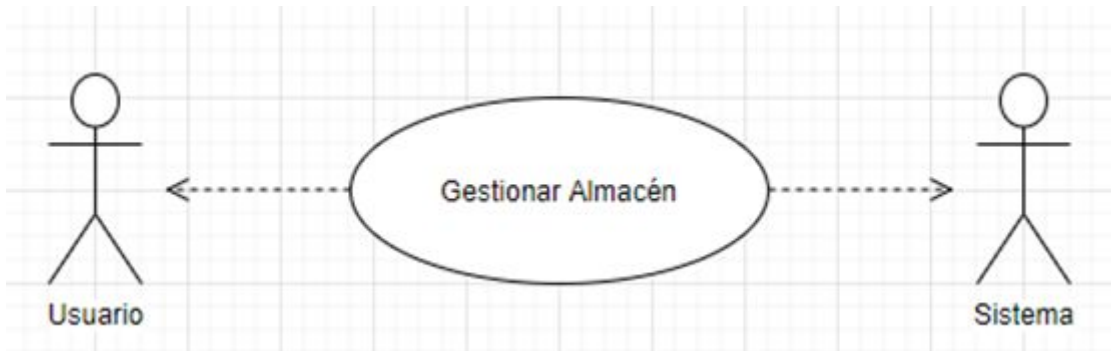
La empresa precisa que el módulo de almacén se actualice mediante la interacción con los módulos de compra, venta y producción de la empresa.

Cuando la empresa realiza una compra, los productos o materias primas, son agregados al almacén de la empresa. Así mismo cuando la empresa realiza una venta son actualizados del almacén según las cantidades.

Para la producción de productos, la empresa realiza una transferencia y actualización del inventario del almacén. Para el efecto la empresa precisa de un comprobante de Orden de Transferencia que sirva para el registro y auditoría de las materias primas adquiridas para la producción.

Para el control y manejo de las materias primas, productos y patrimonio, la empresa precisa de que se generen reportes de los mismos.

A-Modelo de Negocio



1.Actores

Razón	Nombre
Personal que interactúa con el programa (Empleado, Administrador)	Usuario
Sistema (Módulos de: producción, compra, venta)	Sistema

2.Casos de usos conocidos

Razón	Nombre	Código
Comprende todo los datos y movimientos de los ítems de almacén.	Gestionar Almacén	CU_1

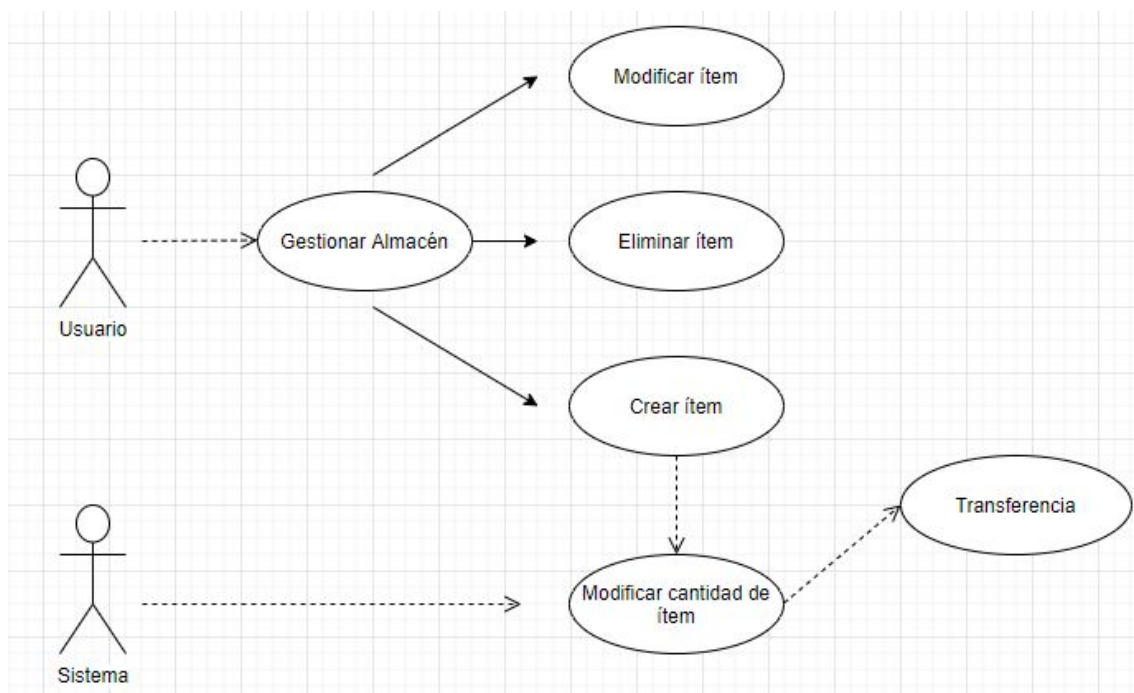
3.Modelo de Casos de Uso



Código:	CU_1
Nombre	Gestionar Almacén
Autor:	
Fecha:	
Descripción:	Gestionar los datos y movimientos de los ítems del almacén.
Actores:	Usuario/Sistema(Módulos de: compra, venta, producción)
Precondiciones:	1. Poseer datos requeridos de los ítems 2. Poseer cantidades para el movimiento del ítem.
Flujo normal:	1. Registrar los ítems con los datos requeridos. 2. Registrar los movimientos de cantidad de ítems

Flujo alternativo:	1. Conseguir los datos requeridos de los ítems.
Postcondiciones:	Accesibilidad del ítem para procesos de los usuarios y el sistema

4.Modelo de Casos de Uso - Requisitos



Código	CU_1 – CUR_1
Nombre	Gestionar almacén – Crear ítem
Fecha	
Autor	
Actor/es	Usuario

Descripción	Crear nuevo ítem	
Precondición	Datos del ítem	
Secuencia Normal	Paso	Acción
	1	Cargar datos del ítem
	2	Verificar campos
	3	Generar código al guardar
Pos condición	ítem creado	
Excepciones	Paso	Acción
	2.1	Resaltar campos obligatorios
	2.2	Mostrar advertencia de tipo de dato erróneo
	3	No se crea el ítem – Lanzar advertencia
Importancia	Alta	
Comentarios	Una vez creado el ítem mostrar los detalles	

Código	CU_1 – CUR_2
Nombre	Gestionar almacén – Modificar ítem
Fecha	
Autor	

Actor/es	Usuario	
Descripción	Modifica los datos de ítem seleccionado	
Precondición	CU_1 – CUR_1	
Secuencia Normal	Paso	Acción
	1	Seleccionar ítem
	2	Modificar campos
	3	Validar campos
	4	Guardar cambios
Pos condición	ítem modificado	
Excepciones	Paso	Acción
	3.1	Resaltar campos obligatorios
	3.2	Mostrar advertencia de tipo de dato erróneo
Importancia	Baja	
Comentarios	Una vez modificado el ítem mostrar los detalles	

Código	CU_1 – CUR_3
Nombre	Gestionar almacén – Eliminar ítem
Fecha	

Autor		
Actor/es	Usuario	
Descripción	Eliminar ítem	
Precondición	CU_1 – CUR_1	
Secuencia Normal	Paso	Acción
	1	Seleccionar ítem
	2	Eliminar ítem
Pos condición	Ítem eliminado	
Excepciones	Paso	Acción
	2	Advertir cantidad del ítem existente si no es igual a cero
Importancia	Baja	
Comentarios	Mostrar mensaje de ítem eliminado correctamente	

Código	CU_1 – CUR_4
Nombre	Gestionar almacén – Modificar cantidad de ítem
Fecha	
Autor	

Actor/es	Sistema	
Descripción	Modifica la cantidad del ítem seleccionado	
Precondición	CU_1 – CUR_1	
Secuencia Normal	Paso	Acción
	1	identificar ítem
	2	Modificar cantidad del ítem
	3	Guardar cambios
Pos condición	Cantidad del ítem modificado, se crea registro de transferencia	
Excepciones	Paso	Acción
	2	Advertir error en la cantidad, verificar el valor
Importancia	Alta	
Comentarios	Mostrar mensaje de cantidad del ítem modificado correctamente	

Diagramas

Diagrama de Clases

Para modelar clases, incluidos sus atributos, operaciones, relaciones y asociaciones con otras clases el UML proporciona un diagrama de clase, que aporta una visión estática o de estructura de un sistema, sin mostrar la naturaleza dinámica de las comunicaciones entre los objetos de las clases.

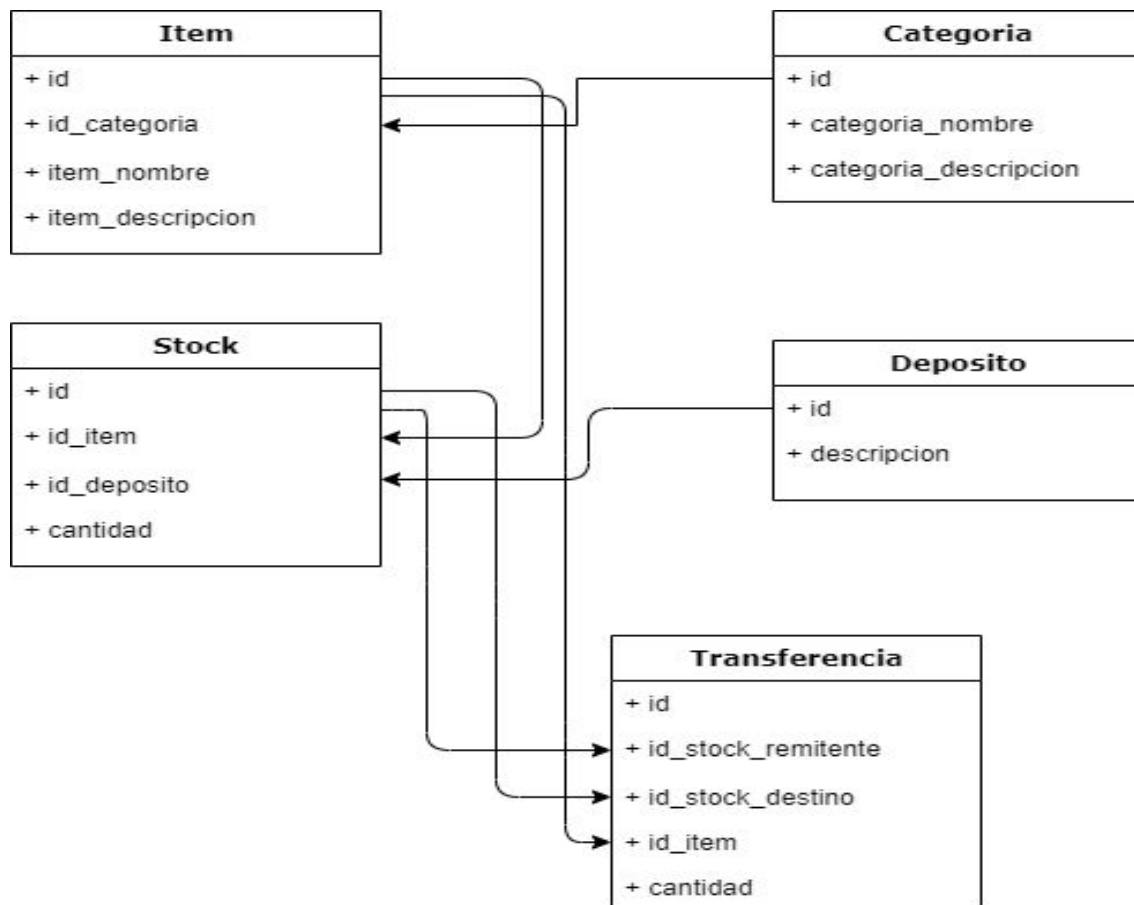


Diagrama de Paquetes

Permiten dividir un modelo para agrupar y encapsular sus elementos en unidades lógicas individuales. En general, pueden tener una interfaz (métodos de clases e interfaces exportadas) y una realización de éstas interfaces (clases internas que

implementan dichas interfaces) Se pueden utilizar para plantear la arquitectura del sistema a nivel macro

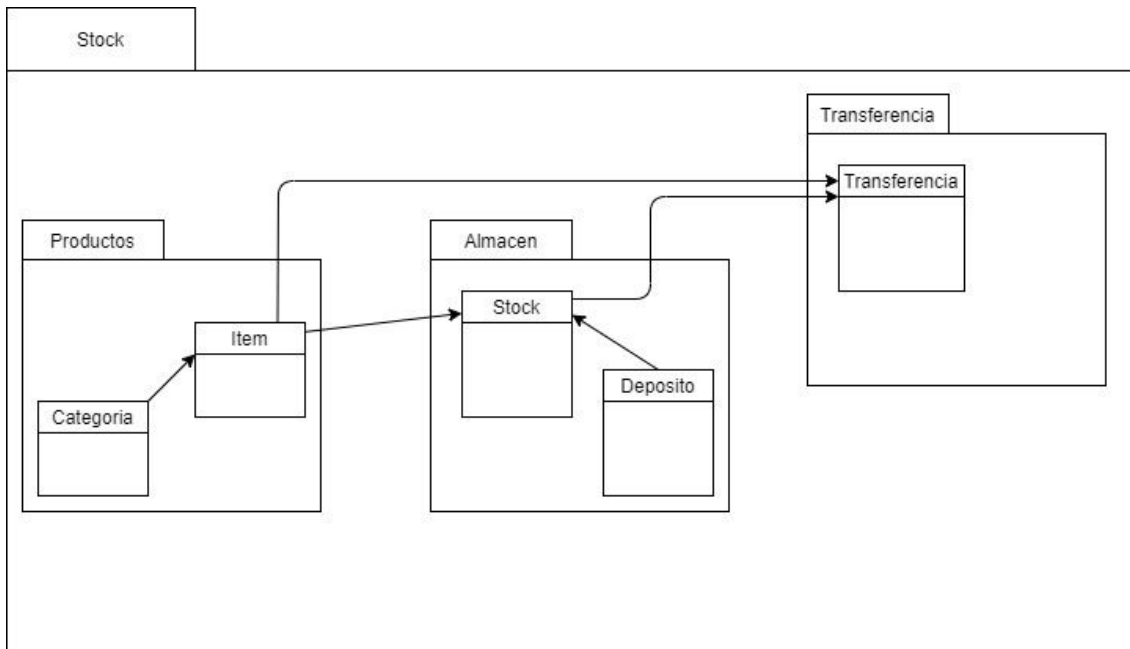


Diagrama de Estados

El diagrama de estado es un método de representación del comportamiento de un sistema que ilustra sus estados y los eventos que ocasionan que el sistema cambie de estado. Un estado es cualquier modo de comportamiento observable desde el exterior.

Además, el diagrama de estado indica acciones (como la activación de un proceso, por ejemplo) tomadas como consecuencia de un evento en particular.

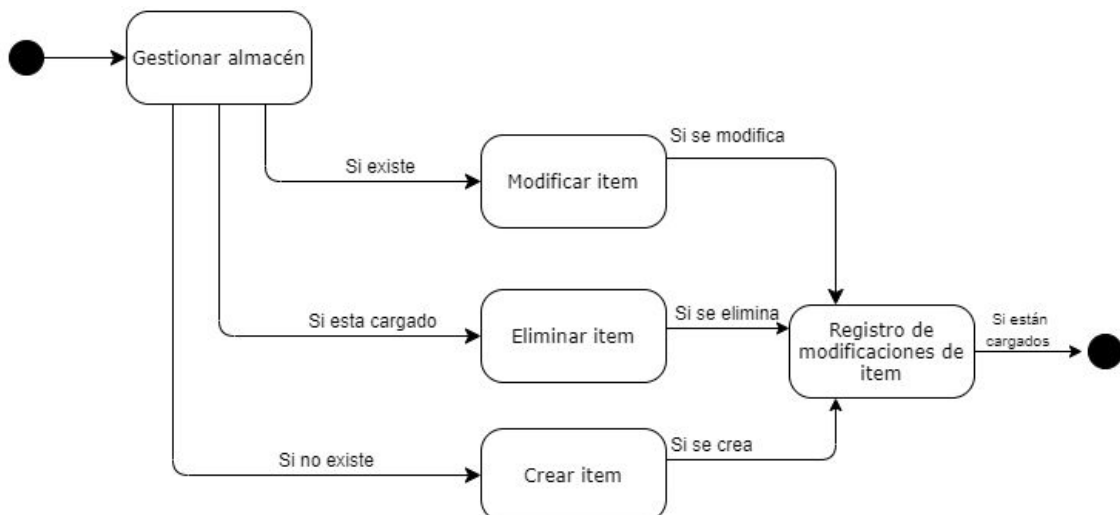


Diagrama de Secuencias

Se usa para mostrar las comunicaciones dinámicas entre objetos durante la ejecución de una tarea. Este tipo de diagrama muestra el orden temporal en el que los mensajes se envían entre los objetos para lograr dicha tarea. Puede usarse un diagrama de secuencia para mostrar las interacciones en un caso de uso o en un escenario de un sistema de software.

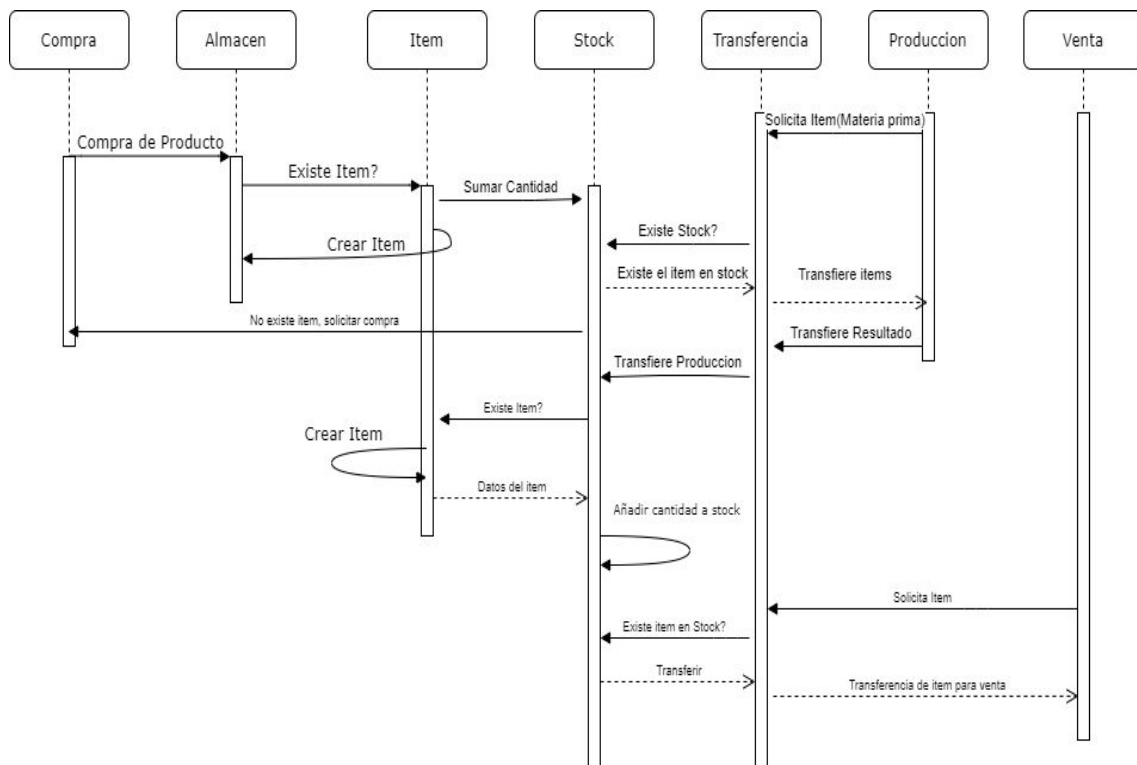
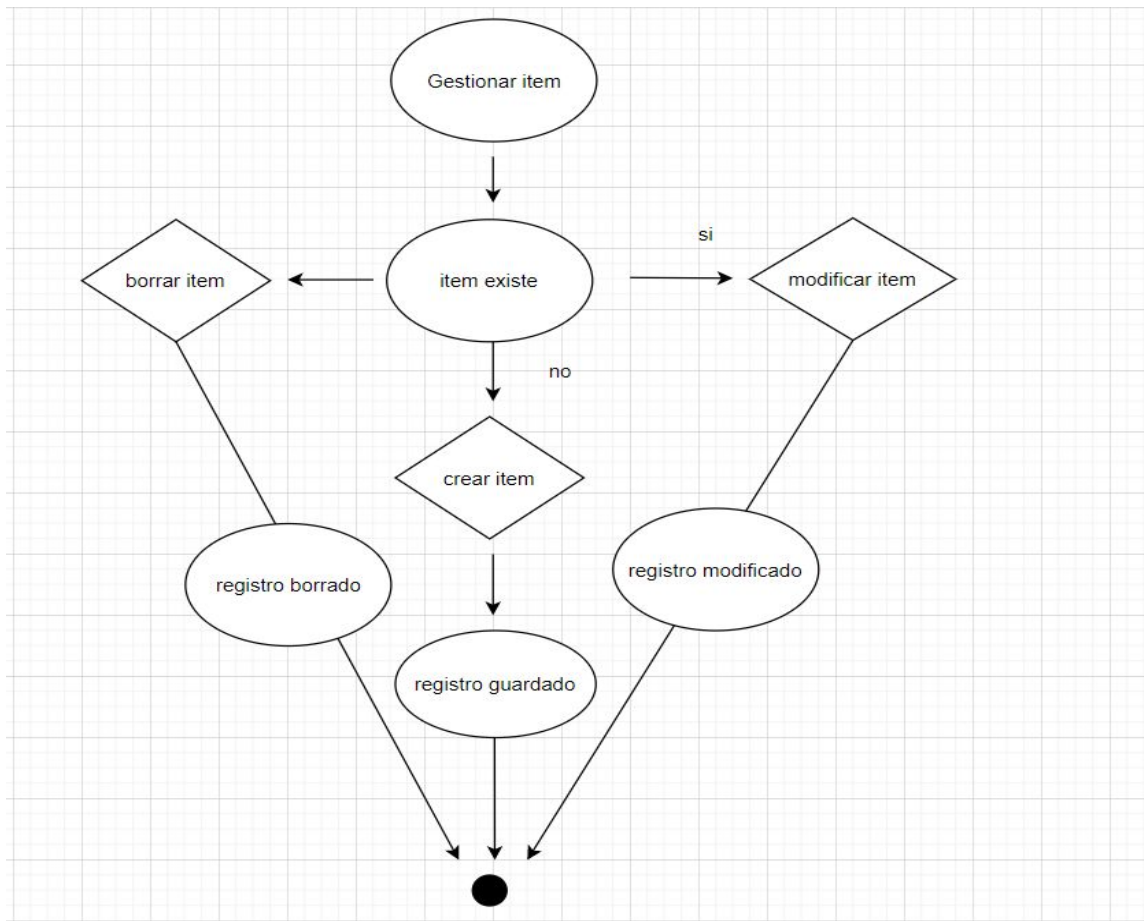


Diagrama de Actividades

Un diagrama de actividad UML muestra el comportamiento dinámico de un sistema o de parte de un sistema a través del flujo de control entre acciones que realiza el sistema. Es similar a un diagrama de flujo, excepto porque un diagrama de actividad puede mostrar flujos concurrentes.



Análisis De Riesgo

Un riesgo es una variable del proyecto que pone en peligro o impide el éxito del mismo, por la cual en el análisis de los mismos deben de ser identificados y expresados en forma clara e inequívoca para que el equipo de desarrollo pueda llegar a un consenso y continuar.

Id.: Identificador de Riesgo

Descripción del Riesgo: Descripción Resumida del Riesgo

Probabilidad (1 a 100): Grado de probabilidad de que el Riesgo finalmente se produzca. Se mide en una escala de 1 a 100 (porcentual).

Nivel de Impacto: Grado de Impacto en el Proyecto en el caso de que el Riesgo finalmente se produjera. Se mide en una escala de 1 a 5, siendo 1=poco influyente hasta 5=fuertemente influyente.

Probabilidad Ourrencia: Valor numérico resultante del producto del Grado de Probabilidad por el Grado de Impacto. Este producto dará la prioridad que tendrá la gestión de este Riesgo y la implantación de sus medidas preventivas o correctoras.

Acciones Prevención: Descripción de las Acciones o Medidas a Adoptar para evitar (mitigar) la aparición final del Riesgo

Acciones Corrección: Descripción de las Acciones o Medidas a Adoptar en el caso en el que el Riesgo finalmente se haya producido.

Id.: 1

Descripción del Riesgo: Pandemia

Probabilidad:100%

Nivel de Impacto:5

Probabilidad Ourrencia: 5

Acciones Prevención:

Acciones Corrección: Utilizar aplicaciones web para administrar el proyecto.

Id.: 2

Descripción del Riesgo: Deserción de algún miembro del grupo

Probabilidad :50%

Nivel de Impacto:3

Probabilidad Ocurrencia: 3

Acciones Prevención: Apoyo moral, trabajo en equipo para evitar el excedente de trabajo individual.

Acciones Corrección: Llevar una documentación y un registro de todas las actividades realizadas por los integrantes y que todos tengan conocimiento de las actividades que llevan a cabo cada miembro.

Id.: 3

Descripción del Riesgo: Conflictos entre miembros de equipo.

Probabilidad :10%

Nivel de Impacto:2

Probabilidad Ocurrencia: 2

Acciones Prevención: Llevar una comunicación continua entre todos los miembros del grupo.

Acciones Corrección: Establecer un consenso en conjunto.

Id.: 4

Descripción del Riesgo: El software requiere interactuar con alguna nueva herramienta.

Probabilidad :40%

Nivel de Impacto:3

Probabilidad Ocurrencia: 3

Acciones Prevención: Utilizar la capacidad de autoestudio de cada miembro.

Acciones Corrección: Asesoramiento externo.

Id.: 5

Descripción del Riesgo: Inactividad de un integrante del grupo.

Probabilidad :50%

Nivel de Impacto:2

Probabilidad Ocurrencia: 2

Acciones Prevención: Dialogar con el integrante.

Acciones Corrección: La eliminación del integrante en el grupo.

Gestión De Calidad

En el presente documento se expone el organigrama en referencia al plan de calidad del proyecto de Sistema de Gestión de Carpintería.

A-Propósito

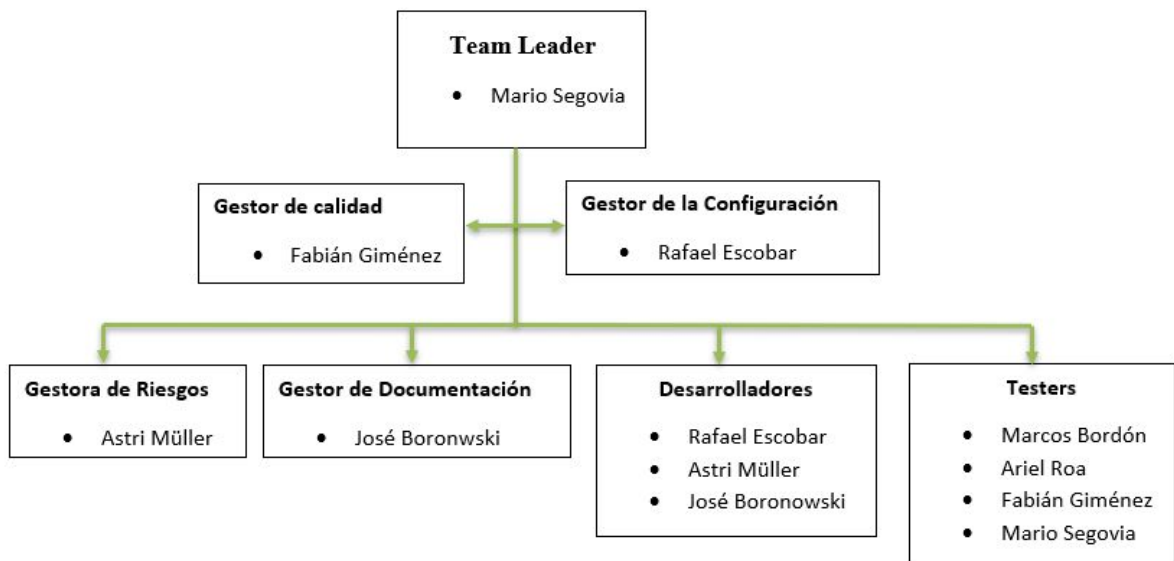
El plan de seguridad de Gestión de Calidad del presente proyecto del Sistema de Gestión para la carpintería. Dentro del plan se detallarán las actividades hechas por cada uno de los responsables de cada rol acorde a la ejecución del proyecto.

Los mismos asumen total obligación de la documentación y de las futuras actualizaciones que se irán ajustando de acuerdo a lo solicitado por el usuario.

B-Gestión

El responsable del gestionamiento de la calidad del software cuyo trabajo es reportar ciertas inconveniencias que presente el proyecto (en caso que esté presente errores y asegurar la calidad del software). El problema será presentado al de mayor nivel de liderazgo según la organización del proyecto.

C-Organización



a. Team Leader, cumple las siguientes tareas:

- Dirige las juntas y evalúa la respuesta de cada persona.
- Ayuda al equipo a descubrir los problemas potenciales tan pronto como sea posible.
- Socializar el conocimiento, esto promueve una estructura de equipo con Organización propia.
- Provee las instrucciones, dirección.
- Guía a un grupo de individuos, con el propósito de lograr cierto objetivo en específico.
- Desarrollar estrategias para lograr lo previsto.
- Provee capacitación a los miembros del equipo.

b. Gestor de Documentación

- Identificar el software.
- Organizar el desarrollo del proyecto.
- Controlar las modificaciones que sufre el software. Velar por una documentación referente a los cambios realizados y productos que de alguna manera no ocasionen la ruptura de la integridad del software.

- Maximizar la productividad minimizando los errores.
- Garantizar las condiciones para que el desarrollo de software sea fácil.

c. Gestor de Riesgo

- Actuar como referente y líder en los procesos de identificación y gestión de riesgos. Asumiendo responsabilidades en la ejecución y dirección de estos procesos.
- Dar soporte a los miembros del equipo del proyecto implicados en la gestión de riesgos.
- Gestionar y mantener el registro de riesgos y las reuniones periódicas de gestión de riesgos.
- Gestionar los recursos y presupuesto asignados a la gestión de riesgos.

d. Gestor de Configuración

- Identificación de todos los proyectos que deben ser controlados.
- Ubicación y localización de los proyectos.
- Definición del ámbito y alcance del control de la configuración, describiendo los procesos incluidos en el proyecto.
- Definición de las reglas de versionado de los proyectos y los criterios de actuación para cada caso, teniendo en cuenta el motivo por el cual se realiza el cambio de versión.
- Definición del ciclo de estados para cada tipo de producto y los criterios de trazabilidad entre los mismos.
- Descripción de funciones y responsabilidades.
- Identificación de la información necesaria de control para auditoría.

e. Gestor de Calidad

- Un proceso de ACS (Aseguramiento de la Calidad del Software).
- Tareas específicas de aseguramiento de la calidad y control de la calidad.
- Prácticas eficientes de ingeniería de software (métodos y herramientas).
- Control de todos los productos del trabajo de software y de los cambios que sufren.
- Un procedimiento para garantizar el cumplimiento de los estándares del desarrollo de software.
- Mecanismos de medición y reporte.

f. Desarrolladores

- Desarrollar el software
- Investigar la necesidad del usuario
- Diseñar y elaborar el sistema
- Diseñar planes de prueba para los programas nuevos desarrollados.
- Realizar pruebas para la medición de calidad en los sistemas desarrollados.
- Detectar errores en los programas.
- Analizar los requerimientos y sugerencias de los usuarios y satisfacer la necesidad.
- Crear soluciones para fallas existentes en el sistema.
- Implementar las soluciones por errores que se presenten.

g. Tester

- Trabajar con desarrolladores de software y equipos de soporte.
- Supervisar aplicaciones y sistemas de software.
- Llevar a cabo pruebas de estrés, pruebas de rendimiento, pruebas funcionales y pruebas de escalabilidad.
- Escribir y ejecutar scripts de prueba.
- Realizar pruebas manuales y automatizadas.
- Pruebas en diferentes entornos, incluyendo web y móvil.

- Escribir informes de fallos.
- Llevar a cabo la planificación de recursos.
- Revisar la documentación.
- Trabajar para cumplir los con plazos departamentales y de proyectos.
- Proporcionar garantía de calidad.
- Proporcionar información objetiva a los equipos de proyectos de desarrollo de software.
- Detectar potenciales fallos.
- Pruebas de diseño para mitigar el riesgo.
- Presentar los resultados a los equipos de desarrollo de software y al cliente.
- Trabajar en múltiples proyectos a la vez.

Gestión de La Configuración

La gestión de la configuración de software (SCM) es un proceso de ciclo de vida de software de soporte que beneficia las actividades de gestión de proyectos, desarrollo y mantenimiento, actividades de garantía de calidad, así como a los clientes y usuarios del producto final.

El presente apartado consiste en el plan de gestión de configuración que corresponderá al desarrollo del módulo de almacén.

A-Objetivo

Cuando se desarrolla software, los cambios realizados son inevitables principalmente debido a los requisitos y fallos que se presentan. Es por eso que es necesario llevar un control y registro de los cambios realizados a fin de reducir errores y tener un producto de calidad.

El objetivo de la gestión de configuración es la de mantener la integridad del producto desarrollado a lo largo del ciclo de desarrollo, garantizando sobre todos que no se realizan cambios incontrolados y que todos los participantes tengan la versión adecuada del producto.

B-Alcance

El presente plan de gestión de la configuración está presente en todas las actividades y procesos que pertenezcan al desarrollo del sistema. El mismo se extiende durante todo el ciclo de vida del software, manteniendo los registros hasta que este sea dejado sin uso.

El documento presenta todos los estándares y configuraciones bases, los procedimientos de control de versiones así como las diferentes nomenclaturas a utilizarse y sus respectivas relaciones.

C- Organización

El plan de gestión de configuración contempla la creación de un comité, este estará compuesto por todos los integrantes del grupo, este comité estará a cargo de tomar todas las decisiones sobre los principales cambios necesarios.

El comité de configuración está liderado por el team leader, y las tareas están organizadas mediante el gestor de configuración.

D-Responsabilidades

El comité de gestión de la configuración, mediante el gestor de la configuración, debe proveer la infraestructura y el entorno de configuración para el proyecto. Debe preocuparse porque todos los integrantes del grupo entiendan y puedan ejecutar las actividades del plan de configuración, así como asegurar que éstas sean llevadas a cabo. Seguir la línea base, controlando las versiones y cambios de ella.

El comité de gestión de la configuración tendrá que:

- Definir las líneas base del plan de gestión de la configuración.
- Revisar todas las peticiones de cambio
- Distribuir las peticiones de cambio aprobadas
- Mantener actualizado la gestión de configuración a todos los integrantes
- Auditar el cumplimiento de la gestión de configuración
- Mantener la integridad del producto mediante la gestión de las peticiones de cambio

E. Distribución

I. Control de Configuración

A. Recibir, evaluar y aprobar Cambios en el Software

1. Cualquier integrante podrá solicitar un cambio/modificacion/alteracion en cualquier ciclo del software.
2. Estos cambios pueden ser de cualquier naturaleza, alcance, software, hardware, etc.
3. El comité evaluará el pedido, analizando el impacto posible en cualquier nivel del software.
4. El comité aprobará y distribuirá las tareas de cambios.

B. Tablero de cambios

1. Se contará con un tablero que refleje los cambios y el estado en que se encuentran los mismos.
2. El tablero reflejará los asignados a los cambios aprobados.

II. Ítems de Configuración

A. Documentación del proyecto:

1. Archivos de documentación: todos los archivos que sean de documentación, deberán tener como nombre “hasta 20 caracteres _versión_fecha” o en su defecto el código identificador junto la versión y fecha del documento.

B. Estándares:

1. Herramientas de desarrollo: Windows, Laravel, Php, Mysql, Versionador de código. Cualquier otro software deberá ser aprobado por el comité de gestión de configuración.
2. La notación a lo largo del software deberá ser “CamelCase”.

3. La notación a lo largo de la base de datos será, minúsculas y plural para tablas. Para nombres de atributos de tablas con abreviación entre 3 a 5 caracteres como máximo.

C. Control de versiones:

1. Cada cambio dentro del ciclo del desarrollo y que haya sido aprobado por el comité de configura, será documentado registrado mediante la herramienta github que funciona como servicio de alojamiento de repositorios de software gestionados por el control de versiones git.
2. El repositorio se encontrara en <https://github.com/jlbb-dev/carpinteria.git>
3. Para el acceso se deberá solicitar al líder del proyecto.
4. Toda la documentación se encontrara dentro de la carpeta Documentos en el repositorio

III. Configuración de auditoría

Las auditorías se realizarán mediante las peticiones de los desarrolladores de pasar a test alguna funcionalidad a fin de comprobar si estas cumplen con los requisitos de configuraciones. Si el resultado de la auditoría revela que no se cumplen los requisitos serán devueltos al desarrollador para producción.

Las auditorías aleatorias también están consideradas, a fin de verificar que se cumple con las integridades de configuración.