

Especificación de Requerimientos

StockManager

Collareda Agustín y Frey Hugo

Índice

Introducción.....	3
Descripción general.....	4
Visión del proyecto.....	5
Propósito.....	5
Objetivos.....	5
Beneficios.....	5
Requerimientos técnicos.....	6
Requerimientos funcionales.....	6
Requerimientos no funcionales.....	7
Funcionalidad.....	7
Eficiencia.....	7
Compatibilidad.....	7
Capacidad de interacción.....	7
Confiabilidad.....	7
Seguridad.....	7
Mantenibilidad.....	7
Flexibilidad.....	7
Protección.....	8
Modelos de Casos de Uso.....	9
Actores.....	9
Diagrama.....	9
Modelo de datos.....	10
Modelo de diseño.....	12
Estándares aplicables.....	13
Referencias.....	14

Especificación de requerimientos

Introducción

El presente documento tiene como fin especificar los requerimientos iniciales de la aplicación solicitada. A su vez, se busca modelar la base de datos y establecer un diseño preliminar con el fin de tener una idea general sobre la aplicación a generar para la cátedra de Gestión de Calidad.

Descripción general

El proyecto trata sobre la generación de un producto de calidad de software con el fin de que el alumno comprenda y aplique los conocimientos de la calidad de un producto software. Este proyecto consiste en recopilar una clase que se vio en un trabajo de la materia de Gestión de Calidad y generar un ABM correspondiente a la clase vista.

Este proyecto es web por lo que se utilizará las herramientas HTML5, CSS3, JavaScript y para el despliegue se utilizara un entorno LAMP.

Visión del proyecto

Propósito

El propósito de este proyecto se centra en el desarrollo de la aplicación Stock Manager, una herramienta digital orientada a la optimización de la gestión de inventarios, adoptando las prácticas y conceptos de calidad vistos en la cátedra de Gestión de Calidad de la Universidad Nacional de la Patagonia Austral para garantizar el cumplimiento de los estándares de calidad esperados para este producto.

Como no se conocen los usuarios finales, se asume que estos serán personas sin conocimientos previos en el ámbito tecnológico, por lo que se dará especial importancia al criterio de usabilidad.

Objetivos

Los objetivos del sistema son:

- Analizar y adoptar las características de la calidad de un producto software según la norma ISO 25010.
- Documentar el proceso de desarrollo enfocado en la calidad del producto.
- Aplicar criterios de diseños correctos y efectivos.
- Utilizar correctamente las herramientas disponibles para la implementación.

Beneficios

Esta aplicación se puede extrapolar en un ámbito profesional ya que la optimización en la gestión de inventarios es muy importante en distintas organizaciones, ya sean empresas chicas, medianas y grandes.

Esto se debe principalmente al control que genera la aplicación en torno a los productos físicos de la organización, si el usuario final puede conocer la cantidad que posee podría ver que tipo de productos necesita para poder expandir la empresa o podría ver la cantidad que posee un empleado y poder generar datos estadísticos de estos.

Por otra parte, el sistema permite reducir en un tiempo considerable algunos procesos que se realizan de manera manual y evitar en gran medida algunos posibles errores humanos que entorpecen la actividad de la organización y conllevan tiempo adicional en corregir los problemas generados.

Requerimientos técnicos

- Desarrollo basado en entornos LAMP

Requerimientos funcionales

Los requerimientos funcionales son:

- El sistema debe mostrar todos los muebles existentes.
- El sistema debe permitir cargar datos de muebles de distintos tipos. Los muebles tienen nombre, ancho y largo.
- El sistema debe permitir modificar muebles cargados con anterioridad. El ancho y largo son reales positivos expresados en metros.
- El sistema debe permitir eliminar muebles.

Requerimientos no funcionales

Funcionalidad

El sistema debe proporcionar todas las funciones necesarias para cumplir con los objetivos del usuario, sin omitir ni agregar comportamiento inesperado.

Eficiencia

El sistema debe hacer uso eficiente de los recursos disponibles y ofrecer tiempos de respuesta aceptables bajo condiciones normales de operación.

Compatibilidad

El sistema debe poder operar en conjunto con otros sistemas y tecnologías, y adaptarse a distintos entornos de ejecución sin necesidad de modificaciones sustanciales.

Capacidad de interacción

El sistema debe ser fácil de aprender y usar por usuarios sin conocimientos técnicos, proporcionando interfaces intuitivas y mensajes claros de error.

Confiabilidad

El sistema debe mantener su funcionamiento correcto ante condiciones normales y previsibles, minimizando fallos y degradación del servicio.

Seguridad

No posee por el momento.

Mantenibilidad

El sistema debe estar diseñado de forma modular para permitir actualizaciones y correcciones sin afectar otras funcionalidades.

Flexibilidad

El sistema debe permitir su adaptación o traslado a distintos entornos operativos o necesidades sin afectar su comportamiento esencial.

Especificación de requerimientos

StockManager

Protección

No posee.

Modelos de Casos de Uso

Actores

Los actores que se presentan en el sistema son:

- Usuario final. Este usuario puede crear, modificar, eliminar y ver todos los muebles del sistema.

Diagrama

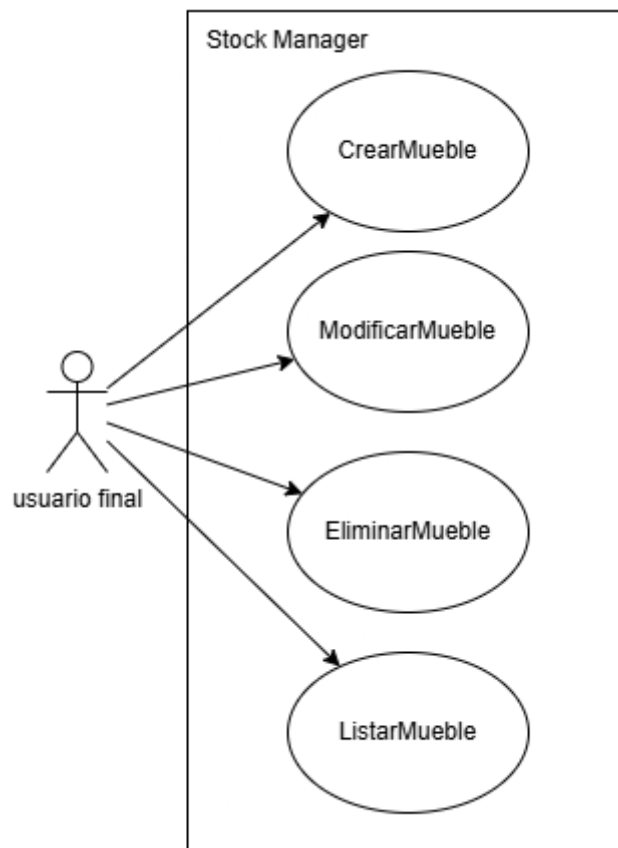


Figura 01: Modelo de Casos de Uso.

Los casos de uso determinados son:

- CU01: ListarMueble. El usuario final podrá ver todos los muebles cargados en el sistema.
- CU02: CrearMueble. El usuario final podrá crear nuevos muebles.
- CU03: ModificarMueble. El usuario final podrá modificar muebles que estén cargados en el sistema.
- CU04: EliminarMueble. El usuario final podrá eliminar muebles que esten cargados en el sistema.

Modelo de datos

Siguiendo la consigna, se puede apreciar que un Mueble posee tres atributos: La medida, el ancho y el largo. Adicionalmente, se decidió agregar los atributos id y nombre para identificarlos. Por otro lado, para adaptarlo a un entorno más real, se añadieron los atributos altura y peso. Se modificó el atributo derivado medida por volumen, el cual trata sobre el producto entre ancho, largo y altura.

Así se puede ver en un modelo MER.

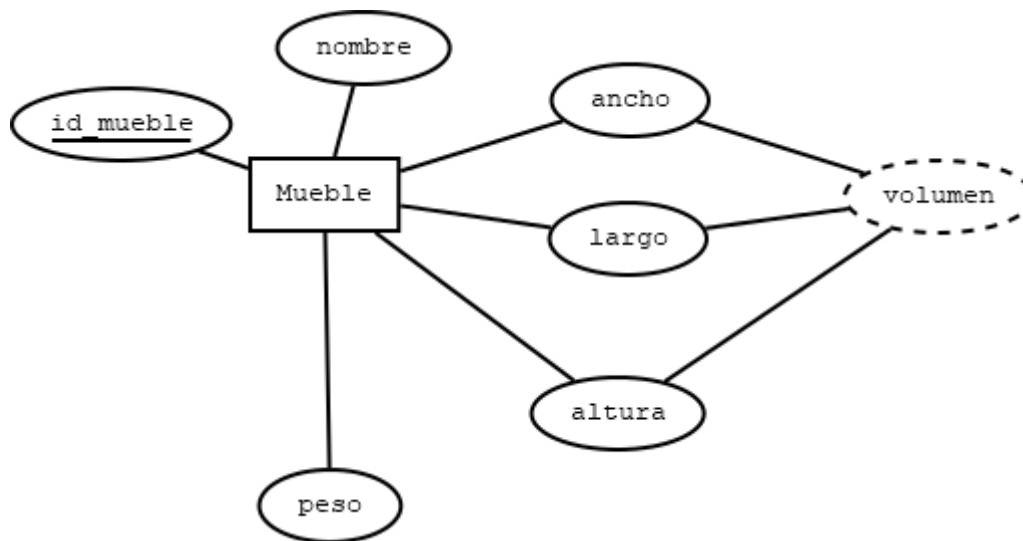


Figura 02: Diagrama Mer.

MUEBLE (id mueble, nombre, peso, ancho, largo, alto)

Figura 03: Grafo relacional.

A continuación se anexa el código sql obtenido del diagrama MER

```
create database if not exists stockmanager;
use stockmanager;
create table if not exists mueble (
    id_mueble int AUTO_INCREMENT not null,
    nombre varchar(30) not null,
    peso float(6, 2) unsigned not null,
    ancho float(6, 2) unsigned not null,
    alto float(6, 2) unsigned not null,
    largo float(6, 2) unsigned not null,
```

Especificación de requerimientos

StockManager

PRIMARY KEY (id_mueble)

);

Modelo de diseño

El sistema se busca que sea modular debido a la posible extensión de funcionalidades por lo que se optó en la siguiente arquitectura:

- Existirá una página de inicio con el fin de redirigir a las demás páginas.
- Por cada ABM se creará la estructura:
 - /ABM
 - CrearABM
 - ModificarABM
 - ListarABM
- Existirá un gestorMuebles que maneja la conexión con los modelos y la interfaz.
- Por cada ABM se generará un Modelo con su nombre la cual maneja la conexión con la base de datos.

Especificación de requerimientos

StockManager

Estándares aplicables

- Serie de normas ISO/IEC 25000

Especificación de requerimientos

StockManager

Referencias

No posee por el momento.