

**Universidad Nacional Autónoma de México**

**Programación Orientada a objetos**

**Requerimientos de Software**

**Versión 1.0**

**Fecha: 30/Noviembre/2020**

# Contenido

Histórico de Cambios………………………………………………………………………………... 3

Descripción general………………………………………………………………………………….. 4

Objetivos del software…………………………………………………………………………... 4

Glosario de términos…………………………………………………………………………….. 4

Requisitos funcionales……………………………………………………………………………….. 5

Identificación de los casos de uso……………………………………………………………… 5

Detallado de los casos de uso…………………………………………………………………... 5

Prototipo de la interfaz del usuario……………………………………………………………. 5

Plan de pruebas del sistema…………………………………………………………………….. 5

Requisitos funcionales……………………………………………………………………………….. 5

Usabilidad………………………………………………………………………………………… 5

Confiabilidad……………………………………………………………………………………... 6

Eficiencia………………………………………………………………………………………….. 6

Mantenimiento…………………………………………………………………………………… 6

Portabilidad ……………………………………………………………………………………….6

Restricciones de diseño y construcción ………………………………………………………...6

Reusabilidad ………………………………………………………………………………………6

Interfaces …………………………………………………………………………………………..6

**Histórico de Cambios**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Descripción** | **Responsable de Actualización** | **Fecha de actualización** |
| **V1.0** | Creación del documento (Secciones 1, 2.1 y 3) | Sandoval Peña Dafne | 28/11/2020 |
| **V2.0** | Edición del documento | Sandoval Peña Dafne, Roldán Sánchez Alexis | 28/11/2020 |
|  |  |  |  |

1. **Descripción general** 
   1. **Objetivos del software**

El gerente de una tienda de discos se encuentra en un gran lio pues el manejo del inventario de discos, hecho principalmente en Excel, es un desastre. Ante esta situación se requirió a un grupo de programadores con el objetivo de ayudarle a crear un software capaz de tener el control del inventario de la tienda.

Dicho gerente requiere de ciertas funcionalidades en el programa para ya no tener tantos problemas. El programa debe ser capaz de iniciar sesión y que a partir de ello los trabajadores puedan realizar ciertas acciones, esto ante el problema de que los trabajadores hacen acciones que no son de acuerdo a su puesto, por ejemplo, que un acomodador venda productos. El manejo de los productos debe ser controlado pues se debe poder venderlos, buscar, eliminar, añadir y tener un control general de ellos. En general los productos deben tener ciertas características, especiales para cada tipo.

* 1. **Glosario de términos**

|  |  |
| --- | --- |
| **Término** | **Significado en el software** |
| Producto | Objeto con características fijas, con la capacidad de poder hacer acciones sobre él y tener un control de este. |
| Disco de música | Es un producto con ciertas características, con el objetivo de poder ser diferenciado de otros productos propios del proyecto este tiene como características principales el nombre del artista, numero de canciones y el nombre de las canciones. |
| Disco de video | Es un producto con ciertas características, con el objetivo de poder ser diferenciado de otros productos propios del proyecto, en este caso un disco de video tiene como características principales el nombre del artista, el número de canciones y la duración total del disco. |
| Audífonos | Producto con ciertas características capaces de poder hacer la diferenciación entre los productos. Los audífonos tendrán como principales características el nombre de la marca y si son inalámbricos o no. |
| Login | Proceso que nos ayuda a controlar el acceso individual mediante la identificación del usuario con sus respectivas credenciales. Esto con el objetivo de tener un control de sus acciones. |
| Empleados | Persona en general registrada en la base de datos del programa. |
| Gerente | Persona- empleado con la característica de poder realizar todas las acciones y tareas disponibles en la tienda. |
| Vendedor | Persona que puede vender un producto, buscar un producto y poner alguna canción |

|  |  |
| --- | --- |
| Acomodador | Persona que puede buscar un disco, poner y buscar una canción |
| Aparatos de ambiente | Objeto con características fijas que puede crear, leer, actualizar y eliminar los métodos y atributos |
| Video played | Aparato que tiene la capacidad de tocar música o video |
| Music played | Aparato que puede tocar solamente música, pero varios discos a la vez |
| Caja | Proceso en el cual se podrá vender y buscar algún producto |
| Ticket | Proceso que se generara al vender un producto con las características de hora, fecha, precio de cada artículo, precio total, nombre de la persona que está vendiendo. |

1. **Requisitos funcionales**

* 1. **Identificación de los casos de uso**

* 1. **Detallado de los casos de uso**
  2. **Prototipo de la interfaz del usuario**
  3. **Plan de pruebas del sistema**

1. **Requisitos funcionales** 
   1. **Usabilidad**

Se seguirá la ISO/IEC 9126-1:2001 que define la usabilidad como la capacidad que tiene un producto software para ser entendido, aprendido, operable, atractivo para el usuario y conforme a estándares/guías, cuando es utilizado bajo unas condiciones específicas.

Por ello la usabilidad se divide en las siguientes subcaracterísticas:

* + - Fácil de aprender: capacidad para ser aprendido en su manejo
    - Fácil de entender: capacidad del producto de ser entendido por nuevos usuarios en términos de su propósito y cómo puede ser usado en tareas específicas.
    - Fácil de operar: capacidad del producto de ser operado y controlado en cualquier momento.
    - Atractivo: capacidad del producto de ser atractivo para sus usuarios.
    - Conformidad: capacidad del producto para adherirse a estándares, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.
    - Intuitivo: capacidad del producto debe estar sujeto a la percepción inmediata de las funcionalidades por el usuario.

* 1. **Confiabilidad**

El sistema debe estar disponible 24/7. Debe tener un bajo sentido de exactitud antes las búsquedas para brindar problemas o soluciones también similares a las buscadas

* 1. **Eficiencia**

El tiempo promedio para dar respuesta a las búsquedas debe ser pequeño y eficiente.

* 1. **Portabilidad**

El sistema debe poder ser consultado desde una computadora, sin complicaciones debido al lenguaje de programación usado.

* 1. **Restricciones de diseño y construcción**

El sistema deberá estar codificado en algún lenguaje orientado a objetos, en este caso JAVA.

* 1. **Reusabilidad**

El sistema será desarrollado sin comprar o reutilizar componentes de otro software.

* 1. **Interfaces**

El sistema no tiene interacción con interfaces de otros sistemas, hardware o software.