
Especificación de requisitos de software

Proyecto: PDV UTH
Revisión 1



Punto de venta

RAZZORS COMPANY



Instrucciones para el uso de este formato

Este formato es una plantilla tipo para documentos de requisitos del software.

Está basado y es conforme con el estándar IEEE Std 830-1998.

Las secciones que no se consideren aplicables al sistema descrito podrán de forma justificada indicarse como no aplicables (NA).

Notas:

Los textos en color azul son indicaciones que deben eliminarse y, en su caso, sustituirse por los contenidos descritos en cada apartado.

Los textos entre corchetes del tipo “” permiten la inclusión directa de texto con el color y estilo adecuado a la sección, al pulsar sobre ellos con el puntero del ratón.

Los títulos y subtítulos de cada apartado están definidos como estilos de MS Word, de forma que su numeración consecutiva se genera automáticamente según se trate de estilos “Titulo1, Titulo2 y Titulo3”.

La sangría de los textos dentro de cada apartado se genera automáticamente al pulsar Intro al final de la línea de título. (Estilos Normal indentado1, Normal indentado 2 y Normal indentado 3).

El índice del documento es una tabla de contenido que Word actualiza tomando como criterio los títulos del documento.

Una vez terminada su redacción debe indicarse a Word que actualice todo su contenido para reflejar el contenido definitivo.

Historial de Revisiones

Fecha	Revisión	Descripción	Autor
15/01/2021	1.0	" Documento SRS y Requerimientos de software"	Alonso Lopez Romo
31/01/2021	1.1	Definiciones, logo y abreviaturas.	Andrés Raso Ramírez
08/02/2021	1.2	Desarrollo del contenido (Requisitos Funcionales, no funcionales y específicos).	Andrés Raso Ramírez

Documento validado por las partes en fecha:

Por el cliente	Por la empresa suministradora
Fdo. D./ Dña	Fdo. D./Dña

Contenido

Historial de Revisiones	4
Contenido	5
INTRODUCCIÓN	6
1. Propósito	6
2. Alcance	6
3. Personal involucrado	6
4. Definiciones, acrónimos y abreviaturas	7
5. Resumen	8
2. DESCRIPCIÓN GENERAL	8
2.1. Perspectiva del producto	8
2.2. Funcionalidad del producto	8
2.3. Características de los usuarios	8
4. Restricciones	9
5. Suposiciones y dependencias	9
6. Evolución previsible del sistema	9
3. REQUISITOS ESPECÍFICOS	10
3.1. Requisitos comunes de los interfaces	10
2. Interfaces de hardware	10
3. Interfaces de software	10
4. Interfaces de comunicación	10
2. Requisitos funcionales o características del sistema	10
2.1. Módulo Clientes	10
2. Módulo Productos	10
3. Módulo Caja	10
4. Módulo Crédito	10
5. Módulo Usuario	10
3. Requisitos no funcionales	10
3.1. Requisitos de rendimiento	10
3.2. Seguridad	11
3. Fiabilidad	11
4. Disponibilidad	11
5. Mantenibilidad	11
6. Portabilidad	11
3.4 Otros requisitos	11

1. Introducción

1.1. Propósito

El propósito de este documento es desarrollar y establecer los requisitos a desarrollar en el proyecto PDV UTH mediante esta la especificación de requisitos de software, al definir de manera clara y precisa todas las funcionalidades y restricciones del sistema que se desea construir, así como las características que lo componen, tanto en su vista general e Interfaz de Usuario, como en su posterior etapa de implementación y puesta en marcha productiva en la organización. El documento va dirigido tanto al equipo de desarrollo y al grupo administración que representa a las empresas que reciben el PDV en su beneficio.

1.2. Alcance

Este documento ERS se compone de tres capítulos: el 1ero sobre el documento en sí, 2do sobre las problemáticas a resolver mediante sistemas de información, y finalmente el despliegado de los requisitos funcionales y no funcionales del sistema en particular.

El sistema se compone de 5 módulos: Usuarios, Productos, Caja (Ventas), Crédito (Fiado), Clientes. En los cuales se gestionan la información de cada módulo con sus acciones CRUD a la vez que los módulos de Venta y Fiado.

Las funcionalidades que no incluye serán la gestión de inventarios, existencias, proveedores, compras.

1.3. Personal involucrado

Nombre	Alonso Lopez Romo
Rol	Gestor Dir. General
Categoría profesional	Gestor de proyecto
Responsabilidades	Establecer las líneas a seguir para el desarrollo del proyecto en las diversas etapas de desarrollo de software
Información de contacto	alonzlopez@uthermosillo.edu.mx
Aprobación	

Nombre	Andrés Raso Ramírez
Rol	Desarrollador Fullstack
Categoría profesional	Desarrollador de software
Responsabilidades	Apegarse a las directrices del plan de desarrollo que se establezca y desarrollar en sistemas informáticos el diseño del sistema PDV UTH.

Información de contacto	a20311079@uthermosillo.edu.mx
Aprobación	(re)aprobado

4. Definiciones, acrónimos y abreviaturas

DEFINICIONES

FRAMEWORKS (para desarrollo de software): Un entorno de trabajo, es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar [1].

WINDOWS FORMS: Es una plataforma libre que contiene librerías accesibles por una interfaz gráfica que está incluida en la microsoft .NET frameworks. La cual es útil para escribir y diseñar aplicaciones para PCs desktop, laptop y tablets. [2]

ESTÁNDAR DE DOCUMENTACIÓN: Se le denomina IEEE 1063-2001, "Estándar para la documentación de usuario de software", y detalla el contenido, la arquitectura y la importancia de un manual de documentación del software. [1]

HARDWARE: Conjunto de componentes físicos embebidos en uno o varios sistemas que nos proporcionan el cómputo central, la interfaz visual y los instrumentos para alimentar la información al sistema. [3]

SOFTWARE: Se conoce como software, lógica o soporte lógico al sistema formal de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados hardware. La interacción entre el software y el hardware hace operativo un ordenador (u otro dispositivo), es decir, el *software* envía instrucciones que el *hardware* ejecuta, haciendo posible su funcionamiento. [4]

DIAGRAMA DE CASOS DE USO UML: El diagrama de casos de uso es una forma de diagrama de comportamiento en lenguaje de modelado unificado (UML, del inglés *Unified Modelling Language*), con la que se representan procesos empresariales, así como sistemas y procesos de programación orientada a objetos. [5]

DIAGRAMA DE SECUENCIA UML: El diagrama de secuencia es un tipo de diagrama usado para modelar interacción entre objetos en un sistema según UML. El diagrama de secuencias muestra la interacción entre objetos ordenados cronológicamente. [6]

DIAGRAMA DE CLASE UML: Es un tipo de diagrama de estructura de clase que describe la estructura de un sistema al mostrar las clases del mismo, sus atributos, métodos y las relaciones con los objetos [6]

DIAGRAMA DE ACTIVIDAD UML: Son representaciones gráficas del flujo de trabajo así como las actividades y acciones implementadas. [6]

ARQUITECTURA DE SISTEMAS: Es un modelo conceptual que define la estructura, el comportamiento, y más aspectos del sistema. La descripción de una arquitectura es una descripción formal de la representación de un sistema, desarrollada de tal manera que apoya el razonamiento sobre la estructura y el comportamiento de un sistema. [7]

PROGRAMACIÓN ORIENTADA A OBJETOS: Es un paradigma de programación basado en el concepto de "Objetos", en el cual se contienen datos generalmente en forma de propiedades y atributos. Y código en forma de procedimientos o métodos [8]

ARQUITECTURA DE TRES CAPAS: La arquitectura en tres capas es un tipo de arquitectura usada en la gran mayoría de sistemas. Se suele usar en sistemas que implementan un modelo de negocio como podría ser una tienda online, una aplicación para gestionar ciertos datos, etc. Sin embargo, no es recomendable usarla en sistemas de tiempo real como los de los coches o aviones. La arquitectura en tres capas lo que hace es dividir el sistema en tres partes diferenciadas, de tal forma que cada capa solo se comuniquen con la inferior. estas tres capas se denominan: persistencia, negocio y Presentación. [9]

MODELO DE DESARROLLO EN CASCADA: El modelo en cascada es un proceso de desarrollo secuencial, en el que el desarrollo de software se concibe como un conjunto de etapas que se ejecutan una tras otra. Se le denomina así por las posiciones que ocupan las diferentes fases que componen el proyecto, colocadas una encima de otra, y siguiendo un flujo de ejecución de arriba hacia abajo, como una cascada. [10]

MODELO DE DESARROLLO EN ESPIRAL: Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades. Las actividades no están fijadas a ninguna prioridad, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior. [11]

PUNTO DE VENTA: Es el tiempo y lugar donde una transacción es completada. En este el comerciante calcula el monto total a pagar por el cliente, indica la suma, indica las opciones de pago para el cliente y se imprime un recibo por la transacción. [12]

FRONT-END: Es la capa de software que interactúa con el usuario [13]

MIDDLEWARE: Es un *software* que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, o paquetes de programas, redes, *hardware* o sistemas operativos.

BACK-END: En ingeniería de software front-end y back-end se refiere a separación de objetivos entre la capa de presentación (front) y la capa de acceso de datos (Back), dentro de un esquema de software o un sistema de hardware integrado. [13]

BASE DE DATOS: Es una colección organizada de datos, generalmente creadas y accedidas con un sistema de computación [15]

RED DE ÁREA LOCAL: Es una red de computadoras que interconecta equipos dentro de un área limitada como residencias, escuelas, laboratorios y campos universitarios así como edificios y oficinas. [14]

PROGRAMACIÓN ORIENTADA A EVENTOS: Es un paradigma de la programación en donde el programa está determinado por eventos específicos como las acciones del usuario (click del mouse, o una tecla específica), dispositivos sensoriales, o mensajes de otros programas o aplicaciones. [16]

EVENTO: En programación y desarrollo de software un evento es la acción o ocurrencia reconocida por el software, que comúnmente se origina en otro tiempo desde el ambiente externo al software y que después es interpretada por el software; estos eventos pueden ser producidos por el usuario, el sistema u otros factores. [17]

CONTROL: En ciencias de la computación, el flujo de control es el orden en el cual instrucciones individuales, statements o funciones de un programa imperativo son ejecutadas o evaluadas. El énfasis en un flujo de control explícito es distinguir entre un lenguaje de programación imperativo y un lenguaje de programación declarativo. [18]

FORMULARIO: Un formulario es un documento, físico o digital, elaborado para que un usuario introduzca datos estructurados (nombres, apellidos, dirección, fecha, etc.) en las zonas correspondientes, para ser almacenados y procesados posteriormente. Esto ayuda a que diferentes instancias registren datos personales de quien los llena para posteriormente ser acreedor al servicio solicitado, siempre y cuando, los datos sean llenados correctamente.

INTERFAZ DE USUARIO: La interfaz de usuario es el medio con que el usuario puede comunicarse con una máquina, equipo, computadora o dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo.

EXPERIENCIA DE USUARIO: La experiencia de usuario es el conjunto de factores y elementos relativos a la interacción del usuario con un entorno o dispositivo concretos, dando como resultado una percepción positiva o negativa de dicho servicio, producto o dispositivo. Dicha percepción depende no solo de los factores relativos al diseño (*hardware*, *software*, usabilidad, diseño de interacción, accesibilidad, diseño gráfico y visual, calidad de los contenidos, buscabilidad o encontrabilidad, utilidad, etcétera), sino de aspectos relativos a las emociones, sentimientos, construcción y transmisión de la marca, confiabilidad del producto, entre otros.

HOSPEDAJE DE APLICACIONES WEB: Una aplicación web es software que se ejecuta en un servidor web a diferencia de las aplicaciones instaladas dentro de un hardware físico local.

DIRECCIÓN IP: Internet Protocol Access por sus siglas en inglés es una etiqueta de numeración que se le asigna a cada dispositivo conectado a una red que utiliza el protocolo de internet para comunicarse.

LOGIN: Término utilizado para definir el acceso del usuario al sistema. []

LOGS: Término utilizado para definir la cantidad de usuarios que ya se encuentran utilizando el sistema []

TICKET: Documento comprobante que recibe el cliente en el POS y donde se desglosa el monto total pagado, montos pagado por cada producto o conjunto de productos, los productos adquiridos, cantidad de productos, fecha y hora de la transacción, así como información fiscal sobre el establecimiento y probables anuncios de promociones.

INTERNET: []

ACRÓNIMOS

PDV: Punto de Venta

IEEE: *Institute of Electrical and Electronics Engineers*/ Instituto de ingeniería eléctrica y electrónica.

DBMS: *DataBase Management System*/ Sistema de manejo de bases de datos

.NET: *Internet Network*

OS: *Operative System*/ Sistema Operativo

ERS /SRS:

API: *Applications programming Interface* / Interfaz de programación de aplicaciones

UML: *Unified Modeling Language*/ Lenguaje de Modelado Unificado.

OOP POO: *Object Oriented Programing*/ Programación orientada a objetos

INE: Instituto Nacional Electoral (Documento de identificación oficial)

CURP: Clave Única de Registro de Población

CLI: *Comand Line Interfaz*/ Interface de línea de comando

ABREVIATURAS

HW: HARDWARE

SW ; SOFTWARE

ADMIN: ADMINISTRADOR
NUM: NÚMERO

Referencias

Referencia	Título	Ruta	Fecha	Autor
1.	Ingeniería de Software-Un enfoque práctico.		2010	R o g e r Pressman
2.	Aprendizaje combinado, aprendizaje electrónico centrado en el alumno y nuevas tecnologías		2012	Oscar, La Red Martínez
3.	Windows Forms Programming in C# (1st ed.).		2006	Chris Sells
4.	System Software	http://home.olemiss.edu/~misbook/sfsysfm.htm	2001	University of Mississippi
5.	El diagrama de casos de uso: UML			ionos.mx
6.	OMG Unified Modeling Language™ (OMG UML), Superstructure			Object Management Group
7.	Architecture-driven modelling methodologies		2011	Hannu Jaakkola and Bernhard Thalheim
8.	Object-Oriented Simulation of systems with sophisticated control		2011	Kindler & Kirvy
9.	Arquitectura en tres capas	https://instintobinario.com/arquitectura-en-tres-capas/	2015	Victor Gomez

10.	Gestiona tu proyecto de desarrollo: en qué consiste el modelo en cascada	https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada#:~:text=El%20modelo%20en%20cascada%20es,se%20ejecutan%20una%20tras%20otra.&text=El%20modelo%20de%20desarrollo%20en,caros%20y%20dif%C3%ADciles%20de%20implementar.		OpenClassroom
11.	A spiral model for software development and enhancement	https://dl.acm.org/doi/10.1145/12944.12948	1988	Barry W. Boehm
12.	Paperless Receipt Solution (PRS) System		2015	James Dyson Foundation
13.	Difference between front ended and back ended MVC		2016	Taphilay Vimal
14.	Octopus: The Lawrence radiation laboratory network	https://web.archive.org/web/20110706202409/http://www.rogerdmoore.ca/PS/OCTOA/OCTO.html	1970	Samuel F. Medicino
15.	Encyclopedia of database Systems	https://www.springer.com/gp/book/9780387355443#ot_herversion=9780387496160	2009	Ling Liu and Tamer M. Özsu
16.	P : S a f e Asynchronous Event-Driven Programming		2012	Vivek Gupta, Ethan Jackson, Shaz Qadeer and Sriram Rajamani
17.	Advance C#: Variable number of Parameters	https://ssw.jku.at/Teaching/Lectures/CSharp/Tutorial/Part2.pdf	2002	Mössenböck, Hanspeter
18.	Flow diagrams, turing machines and languages with only two formation rules		1996	Böhm, Jacopini

5.

IEEE Recommended Practice for Software Requirements Specification. ANSI/IEEE std. 830, 1998.

6. Resumen

- Descripción del contenido del resto del documento
- Explicación de la organización del documento

El contenido de resto del documento contendrá una descripción general para describir los factores generales que afectan al sistema y sus requerimientos y los requerimientos específicos que contendrán todos los requerimientos de software a un nivel de detalle como para permitir a los diseñadores diseñar el sistema que satisfaga estos requerimientos y a los especialistas en pruebas para comprobar que el sistema satisfaga estos requerimientos.

2. Descripción general

[Se considera en esta parte la descripción de los factores principales que afectan al espacio de la solución. Incluya aquellos ítems como perspectiva del producto, funciones del producto, características de usuario, limitaciones, supuestos y dependencias. No se incluye en esta sección la descripción de los requerimientos.]

2.1. Perspectiva del producto

PDV UTH, El punto de venta PDV UTH consiste en un sistema informático que permite habilitar a los comerciantes y administradores de pequeños comercios, para gestionar los productos que comercializan, controlar las entradas al almacén para asegurar existencia.

2.2. Funcionalidad del producto

- Permite el registro de las ventas y de sus varios productos vendidos.
- Expide un ticket de cada venta con todos los detalles necesarios para describirla.
- Las cajas llevarán a cabo acciones para su control, como apertura, cierre, corte, egresos de dinero, ingresos de fondo entre otros.
- Permite el registro de productos que el negocio comercializa, empleando varios datos importantes, incluyendo imágenes de los mismos.
- La consulta de productos podrá utilizarse para varios fines, incluyendo la venta.
- Los usuarios ingresan con sus credenciales registradas a sus módulos autorizados.

2.3. Características de los usuarios

Descripción de los usuarios del producto, incluyendo nivel educacional, experiencia y experiencia técnica.

Tipo de usuario	Administrador
Formación	Preparatoria, y manejo de su negocio, en sus finanzas y personal.
Habilidades	Manejo de computadores, aplicaciones de ventanas.

Actividades	Gestionar información de productos, de usuarios, de clientes, y de cuentas por cobrar. Apertura y corte de cajas.
-------------	---

Tipo de usuario	Cajero
Formación	Primaria o superior.
Habilidades	Manejo de computadoras. Aritmética básica
Actividades	Escanear los productos a la venta de cada cliente. Pesar y capturar su equivalente en el sistema al momento de la venta, totalizar y registrar el pago de cada venta. Si el sistema lo permite asociar parcial o totalmente la venta a su crédito disponible.

4. Restricciones

- ✓ El sistema será desarrollado en la plataforma .NET(v. 5.0) C# (v. 9.0) con vista en Windows Form, para ejecutarse en sistemas operativos Windows.
- ✓ El sistema utilizará el motor de la base de datos en MariaDB 10.5.8.
- ✓ La metodología para el desarrollo del sistema se basará primeramente en un modelo de Cascada, para poder después del diseño desarrollar cada módulo de una manera incremental.
- ✓ El ciclo de vida del desarrollo del software consta de las siguientes etapas: Análisis, Plan, Diseño, Codificación, Pruebas, Liberación. Mantenimiento de Software.
- ✓ Todos los miembros del equipo de desarrollo, participan con su versión del mismo sistema con un alcance mínimo (productos, usuarios, ventas, cajas, asistencias); pudiendo cada alumno personalizar e incrementar cada módulo o funcionalidades del sistema.

5. Suposiciones y dependencias

- Al liberar el producto de software, se han VERIFICADO y VALIDADO todos los requisitos.
- Los nuevos requisitos se establecerán en una nueva FASE de desarrollo del proyecto (version 2.0).
- Los desarrolladores siguen a la par las instrucciones de diseño y desarrollo que se brindan en la clase.
- Todos los clientes que se benefician del PDV cuentan con un equipo de cómputo con sistema operativo Windows version 10 en adelante (se tiene que instalar el framework correspondiente al utilizado para el desarrollo).

6. Evolución previsible del sistema

1. Módulo de proveedores
2. Módulo de clientes.
3. Módulo de créditos a clientes.
4. Modulo de recompensas a clientes frecuentes
5. Módulos de compras
6. Gestión de almacén, entradas y salidas (existencias)
7. Gestión de control de inventarios por FIFO, LIFO y/o precio promedio.

3. Requisitos específicos

3.1. Requisitos comunes de los interfaces

3.1.1. Propiedades de las ventanas de nuestro sistema: Nuestra interfaz será minimalista, donde se tratará de no saturar al usuario con información innecesaria, donde los botones serán grandes para que sean fácilmente distinguibles uno entre los otros así como se utilizarán colores de fondo que contrastan con los botones e iconos pero que no se pierda la estética de la aplicación

2. Interfaces de hardware

✓ Periféricos a emplear:

- Impresora Térmica
- Lector de Códigos de Barras infrarrojas.
- Cajón de Dinero automático con botón.

3. Interfaces de software

- MySQL .NET connector 6.1.2:
- DataBaseLIB. dll (Librería de clases, desarrollada por nosotros)
- CRUD (Create, Read, Update, Delete)
- Api
- Librería
- Componentes especiales (Iconos)
- FontAwesome.Sharp 5.15.3

4. Interfaces de comunicación

Para el trabajo del desarrollo se utilizarán las plataformas:

- google.meet,
- google Classroom
- gmail,
- whatsapp

2. Requisitos funcionales o características del sistema

☐ Módulo Productos

☐ ADMIN

☐ **RF01.** El sistema debe de permitir al ADMIN registrar productos que se pueden vender en el Punto de Venta. Se ingresen varios datos de estos productos, como lo son: Nombre: Nombre corto del producto.

- Marca: Descripción amplia del producto.
- Descripción: Marca del producto.
- Precio: Precio de venta del producto.
- Código de Barra: Serie de numérica del código de barras con el que cuenta el producto (Entre 13 y 14 dígitos)
- Imagen: Imagen, fotografía, o edición que el mismo incluye al registro del producto (PNG, JPG).
- El sistema notifica al usuario con una ventana de mensajes, si el registro es correcto, se tuvo algún error.

El sistema notifica al usuario con una ventana de mensajes, si el registro es correcto, o se tuvo algún error.

□**RF02.** El sistema debe de permitir al usuario de tipo ADMIN **actualizar** la información de los productos registrados previamente y que se pueden vender en el PDV. Se modifican los siguientes datos.

- Marca: Descripción amplia del producto.
- Descripción: Marca del producto.
- Precio: Precio de venta del producto.
- Código de Barra: Serie de numérica del código de barras con el que cuenta el producto (Entre 13 y 14 dígitos)
- Imagen: Imagen, fotografía, o edición que el mismo incluye al registro del producto (PNG, JPG).

El sistema notifica al usuario con una ventana de mensajes, si el registro es correcto, o se tuvo algún error.

□**RF03.** El sistema debe de permitir al usuario de tipo ADMINISTRADOR **eliminar** el registro con la información de los productos registrados previamente y que se pueden vender en el PDV. La eliminación se llevará a cabo identificando al producto con una CAMPO CLAVE, ÚNICO para cada registro (id).

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o se cometió algún error.

□**RF04.** El sistema debe de permitir al usuario de tipo ADMINISTRADOR **consultar** la información de los productos registrados previamente y que se pueden vender en el PDV. Para definir los detalles de los mismos y definir si necesitan llevar a cabo algún tipo de acción administrativa (modificar precios, consultar productos registrados, etc). Se consultan los productos por palabras contenidas parcial o totalmente en parámetros.

- Nombre
- Marca: Descripción amplia del producto.
- Categoría: la manera de agrupar y clasificar los productos
- SubCategoría: TDB si se requiere.
- Descripción: Marca del producto.
- Precio: TDB.
- Código de Barra: Serie numérica del código de barras con el que cuenta el producto (Entre 13 y 14 dígitos).

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o se cometió algún error.

□**RF05.** El sistema debe de permitir al usuario de tipo ADMIN **modificar** los precios de una forma rápida y correcta al mostrar los productos a cambiar el precio, filtrados y agrupados resultado de una consulta previa; es decir, cambiar SOLO EL DATO DEL PRECIO de una lista de productos, filtrados por criterios que destine el ADMIN. El precio nuevo y el anterior deben de estar resaltados para que el ADMIN pueda confirmar que su acción de modificación es la adecuada. TBD características visuales para indicar si el precio NUEVO es mayo o menor que el precio ANTERIOR.

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o se cometió algún error.

□**RF06.** El sistema debe de permitir el **registro** de las marcas para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario tipo ADMINISTRADOR. Pudiendo registrar los datos:

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF07.** El sistema debe de permitir el **registro** de las marcas para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario tipo ADMIN. Pudiendo registrar los datos:

- Nombre: Nombre de la marca
- Descripción: Descripción amplia de la marca.
- Logo: Una imagen PNG, entre 100-X100 y 300X300px, con un máximo de 1mb.

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF08.** El sistema debe permitir la **eliminación** de los registros de marcas a usarse para detallar los productos y clasificarlos también. Registro realizarse por el Usuario tipo ADMIN. Cada registro debe IDENTIFICARSE COMO ÚNICO, para poderse eliminar sin afectar a los demás

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF09.** El sistema debe de permitir la **consulta** de la información de las marcas a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario.

- Nombre: Nombre de la marca
- Descripción: Descripción amplia de la marca.

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□RF10. El sistema debe de permitir el **registro** de las categorías a usarse para detallar los productos, y clasificarlos también. Registros a realizarse por el usuario tipo ADMIN. Pudiendo registrar los datos:

- Nombre: Nombre de la categoría.
- Descripción: Descripción amplia de la categoría.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□RF11. El sistema debe de permitir la **actualización** de la información de las categorías a usarse para detallar los productos, clasificarlos también. Registro a realizarse por el Usuario tipo ADMIN, Cada registro debe IDENTIFICARSE COMO ÚNICO, para poderse eliminar sin afectar a los demás.

- Nombre: Nombre de la categoría.
- Descripción: Descripción amplia de la categoría.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□RF12. El sistema debe de permitir la **eliminación** de los registros de las categorías a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el Usuario tipo ADMIN, Cada registro debe IDENTIFICARSE COMO ÚNICO, para poderse eliminar sin afectar a los demás. El sistema notificará al usuario con una ventana de mensajes, si el borrado es correcto, o si ocurrió algún error.

□RF13. El sistema debe de permitir la **consulta** de la información de las categorías a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario tipo ADMINISTRADOR. Pudiendo registrar los datos:

- Nombre: Nombre de la categoría.
- Descripción: Descripción amplia de la categoría.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□RF14. El sistema debe permitir el **registro** de las subcategorías a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario tipo ADMINISTRADOR. Pudiendo registrar los datos:

- Nombre: Nombre de la subcategoría.
- Descripción: Descripción amplia de la subcategoría.
- Categoría a la que pertenece (id).
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF15.** El sistema debe de permitir la **actualización** de la información de las subcategorías a usarse para detallar los productos, clasificarlos también. Registro a realizarse por el Usuario tipo ADMIN, Cada registro debe IDENTIFICARSE COMO ÚNICO, para poderse eliminar sin afectar a los demás.

- Nombre: Nombre de la subcategoría.
- Descripción: Descripción amplia de la subcategoría.
- Categoría a la que pertenece (id).

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF16.** El sistema debe de permitir la **eliminación** de los registros de las subcategorías a usarse para detallar los productos, y clasificarlos también. **Sin afectar el registro de la categoría a la que pertenece** los registros a realizarse por el Usuario tipo ADMIN, Cada registro debe IDENTIFICARSE COMO ÚNICO, para poderse eliminar sin afectar a los demás.

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF17.** El sistema debe de permitir la **consulta** de la información de las subcategorías a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario tipo ADMINISTRADOR. Pudiendo registrar los datos:

- Nombre: Nombre de la categoría.
- Descripción: Descripción amplia de la categoría.
- Categoría: Nombre de la categoría a la que pertenece.

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

CAJERO

□**RF18.** El sistema debe de permitir al usuario de tipo CAJERO **consultar** la información de los productos registrados previamente y que se pueden vender en el PDV. Para verificar los detalles de los mismos y definir y que se agreguen productos a la lista en el proceso de venta (modificar precios, consultar productos registrados, etc). Se consultan los productos por palabras contenidas parcial o totalmente en parámetros.

- Nombre: Nombre del producto
- Marca: Descripción amplia del producto.
- Categoría: la manera de agrupar y clasificar los productos
- SubCategoría: el detalle de clasificación del producto.
- Descripción: Marca del producto.
- Precio: precio del producto.
- Código de Barra: Serie numérica del código de barras con el que cuenta el producto (Entre 13 y 14 dígitos).

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF19.** El sistema debe de permitir al Usuario tipo **CAJERO** la **consulta** de la información de las marcas a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario.

- Nombre: Nombre de la marca
- Descripción: Descripción amplia de la marca.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF20.** El sistema debe de permitir al Usuario tipo **CAJERO** la **consulta** de la información de las CATEGORÍAS a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario.

- Nombre: Nombre de la subcategoría.
- Descripción: Descripción amplia de la subcategoría.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF21.** El sistema debe de permitir al Usuario tipo **CAJERO** la **consulta** de la información de las SUBCATEGORÍAS a usarse para detallar los productos, y clasificarlos también. Registro a realizarse por el usuario.

- Nombre: Nombre de la subcategoría.
- Descripción: Descripción amplia de la subcategoría.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

3. Módulo Caja

ADMIN

□**RF22.** El sistema debe de permitir al Usuario de Tipo **ADMIN** **registrar** la información de las **CAJAS** utilizables por el PDV en su negocio. La información de la caja valida, para cada equipo de cómputo que se utilice con este fin, es;

- (id) int primary key auto_increment
- "6622589589"
- Nombre: Nombre corto de la caja (Caja01, Caja-Rapida02)
- Descripción: Descripción de uso de la caja.
- Uso de MAC ADDRESS para hacer ÚNICA la instalación en cada EQUIPO. TBD
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF23.** El sistema debe de permitir al Usuario de Tipo ADMINISTRADOR el BORRADO LÓGICO del registro y la información de las CAJAS que utilizables por el PDV en su negocio. El campo activo servirá como indicador de las cajas que se pueden usar o han sido inactivas:

- (id) int primary key auto_increment
- "6622589589"
- Nombre: Nombre corto de la caja (Caja01, Caja-Rapida02)

- Descripción: Descripción de uso de la caja.
- La MAC ADDRESS se inhabilita junto con el equipo. Pero no se MODIFICA. **TBD**
- **Activo: campo para activar o desactivar el uso del equipo de cómputo como caja en el PDV. Se maneja de forma interna, o por el ADMIN.**
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF24.** El sistema debe de permitir al Usuario de Tipo ADMINISTRADOR el BORRADO LÓGICO del registro

- La MAC ADDRESS se inhabilita junto con el equipo. Pero no se MODIFICA. **TBD**
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF25.** El sistema debe de permitir al Usuario de Tipo ADMINISTRADOR consultar la información de las CAJAS que utilizables por el PDV en su negocio. Los campos que sirven para la consulta son:

- Nombre: Nombre corto de la caja (Caja01, Caja-Rapida02)
- Descripción: Descripción de uso de la caja.
- La MAC ADDRESS se inhabilita junto con el equipo. Pero no se MODIFICA. **TBD**
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF26.** El sistema debe de permitir al Usuario de Tipo ADMINISTRADOR para llevar a cabo RETIROS EN EFECTIVO de las CAJAS que se utilizan en el PDV en su negocio. Solo el ADMINISTRADOR puede llevar a cabo esta acción, y se le debe de solicitar el ingreso repetido de su contraseña)o claves de acceso TDB=. Los campos se especifican para los retiros son:

- TIPO: diferentes razones para el retiro por el ADMIN
- Pago a proveedores
- Pago de sueldo a Empleados (cajeros)
- Emergencia
- Retiro para prevenir robos (establecer máximo de efectivo en caja, por efecto 5000).
- Razón de retiro: Descripción de uso y origen del retiro que realiza el ADMINISTRADOR.
 - Pago de proveedor: Por defecto debe tener una 'una plantilla' con texto:
 - 'Se retira efectivo de la caja en{fecha/hora} por ADMINISTRADOR {nombreCompleto} por ... {detalla adminisyrador}'
 - Emergencias: Por defecto debe tener una "plantilla" con texto:
 - Pago de proveedor: Por defecto debe tener una 'una plantilla' con texto:
 - 'Se retira efectivo de la caja en{fecha/hora} por ADMINISTRADOR {nombreCompleto} por ... {detalla emergencia administrador}'

Prevención de robos: Establecer maximo de cantidad disponible en la caja; Por defecto debe tener una “plantilla” con texto:

- Pago de proveedor: Por defecto debe tener una ‘una plantilla’ con texto:
 - ‘Se retira \$ de la caja en{fecha/hora} por ADMINISTRADOR {nombreCompleto} por ... {detalla emergencia administrador}”
- Descripción: Descripción de uso de la caja.
- La MAC ADDRESS se inhabilita junto con el equipo. Pero no se MODIFICA. **TBD**
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF27.** Desarrollar la necesidad de Submódulo EMERGENCIA TDB.

- Fecha/hora: timestamp del registro: interno
- Usuario ADMINISTRADOR: interno
- Fondo: Por defecto son \$500.00, pero si se reuiere cambiarlo, el admin lo puede hacer.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF28.** El sistema debe de permitir al usuario de tipo ADMINISTRADO APERTURAR las CAJAS que son utilizadas por el PDV en su negocio. EL ADMINISTRADOR

- Fecha/hora: timestamp del registro: interno
- Usuario ADMINISTRADOR: interno
- Fondo: Por defecto son \$500.00, pero si se requiere cambiarlo, el admin lo puede hacer.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF29.** El sistema debe de permitir al usuario de tipo CAJERO y ADMINISTRADO llevar a cabo CORTE DE CAJAS, por diversas situaciones definidas. El USUARIO debe ingresar el efectivo de la caja, capturando cuantos billetes, o monedas, de cada denominación tiene. Los campos que sirven para el cierre de cajas son:

- Fecha/hora: timestamp del registro: interno
- Usuario ADMINISTRADOR: interno
- captura de efectivo por denominaciones.
 - captura de cuantas monedas y/o billetes tiene la caja:
 - cuantos de 1000
 - 500
 - 200
 - 100
 - 50
 - 20
 - 10
 - 5
 - 2

- 0.5
- 0.2
- 0.1
- Tipo de corte (razón):
 - cambio de turno
 - cierre de caja
 - Fin de jornada
 - Ausencia temporal (baño, comida, etc)

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

■RF30. El sistema debe de permitir al usuario de tipo ADMINISTRADOR/CAJERO llevar a cabo el proceso de VENTA EN CAJA, que consiste en la captura/scan de los productos que selecciona el cliente para comprar así como el cálculo de totales y despliegue de la información concerniente al mismo proceso. El cajero debe ingresar el METODO_de_pago o el efectivo que el cliente utiliza par pagar. El sistema debe registrar y notificar la información de la venta en BD, así como expedir un ticket de la compra que realizó el cliente. las CAJAS que son utilizadas por el PDV en su negocio. EL ADMINISTRADOR debe ingresar el efectivo de fondo en este proceso de apertura. Los campos que sirven para la apertura de caja son:

- Fecha/hora: timestamp del registro: interno
- Usuario CAJERO/ADMINISTRADOR: interno
- FOLIO.
- MONTO
- NUMERO PRODUCTO
- LISTA DE PRODUCTOS (id producto, cantidad comprada)

El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

■RF31. El sistema debe de permitir al usuario de tipo ADMINISTRADOR llevar a cabo el proceso de CANCELACIÓN VENTA(previa), en el cual se modifica el valor del campo/bandera/flag "cancelado" a true. Los campos que se requieren para esta funcionalidad, se usan para realizar una consulta de ventas, y poder identificar la venta a cancelar, estos campos son:

- Usuario CAJERO/ADMINISTRADOR: interno
- Usuario CAJERO/ADMINISTRADOR: Interno
- FOLIO. TDB
- 200 -- **SELECT * from ventas WHERE montoTotal >= (200-50) AND montoTotal <(250+50)**
- NUMERO PRODUCTO
- LISTA DE PRODUCTOS (*idProducto*, cantidad comprada). estas se deben registrar en una tabla de detalle de la venta.

Ya que se ha identificado la VENTA a CANCELAR, el ADMINISTRADOR debe recibir una pregunta de confirmación para hacer definitiva la CANCELACIÓN. El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□RF32. Ya que se ha identificado la VENTA a CANCELAR, el ADMINISTRADOR debe recibir una pregunta de confirmación para hacer definitiva la CANCELACIÓN.

- Resultados por:
 - Promedios
 - Listados
 - Máximos/Mínimos
- Usuario CAJERO/ADMINISTRADOR: interno
 - **Por fecha unica**
 - **Por rango de fecha. (fechaInicial a fechaFinal)**
 - Por Rango de Horas
- Usuario(quien vendio) CAJERO/ADMINISTRADOR: Interno
- FOLIO:
 - Único (2563158963694)
 - Con comodín ("2563158963694%"). todas la ventas de la 2563158963682-2563158963694
 - Por rango de folios
 - Folio inicial a folio final, o último
 - Por colección de folios.
- Monto:
 - Por monto
 - Por rango de monto
 - Por número de artículos
- Por productos vendidos
 - Texto de nombre, o descripción
 - DAME TODAS LA VENTAS QUE TENGAN EN LOS TEXTOS DE SUS PRODUCTOS ESTE STRING= "%Dona Maria%"
- NUMERO PRODUCTO
- LISTA DE PRODUCTOS (*idProducto*, cantidad comprada). estas se deben registrar en una tabla de detalle de la venta.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

5. Módulo Usuario

□RF33. El sistema debe de permitir al usuario de tipo ADMINISTRADOR registrar a usuarios de nivel ADMINISTRADOR o CAJERO, estos campos son:

- Primer Nombre: string, char[] nombre = new char[30] ... VARCHAR (30) ⇒ NOT NULL
- Segundo Nombre: string, char[] nombre = new char[30] ... VARCHAR (30).
- Primer Apellido: string, char[] nombre = new char[30] ... VARCHAR (30) ⇒ NOT NULL
- Segundo Apellido: string, char[] nombre = new char[30] ... VARCHAR (30)

- correo electronico; String VARCHAR(255)
a20311079@uthermosillo.edu.mx
- numero de telefono celular: VARCHAR(10)
- numero de telefono fijo: VARCHAR(10)
- Correo Electrónico: string, varchar(120)
- Celular: varchar (10)
- Telefono de casa: varchar(150)
- Calle, varchar(100)
- Numero: varchar(150)
- Colonia: varchar(150)
- CodigoPostal: varchar(5)
- Analizar estándar...
- Municipio: varchar(100)
- Estado(80),
- Pais(50),//Mexico MX, US Estados Unidos, etc.
- Contraseña: varchar(255)
- Tipo (Enum('ADMINISTRADOR', 'CAJERO'))
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF34.** El sistema debe de permitir al usuario de tipo ADMINISTRADOR **eliminar** a usuarios de nivel ADMINISTRADOR o CAJERO, estos campos son:

- ID del usuario a eliminar.
El usuario a eliminar debe de seleccionarse mediante una consulta previa. El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF35.** El sistema debe de permitir al usuario de tipo ADMINISTRADOR consultar a los usuarios de nivel ADMINISTRADOR o CAJERO, los campos que se requieren para la consulta, son

- Todos los campos utilizan comodines
 - apellidoMaterno LIKE"%ROM%"
- Primer Nombre: string, char[] nombre = new char[30] ... VARCHAR (30) ⇒ NOT NULL
- Segundo Nombre: string, char[] nombre = new char[30] ... VARCHAR (30).
- Primer Apellido: string, char[] nombre = new char[30] ... VARCHAR (30) ⇒ NOT NULL
- Segundo Apellido: string, char[] nombre = new char[30] ... VARCHAR (30)
- correo electrónico; String VARCHAR(255)
a20311079@uthermosillo.edu.mx
- numero de telefono celular: VARCHAR(10)
- numero de telefono fijo: VARCHAR(10)
- Correo Electrónico: string, varchar(120)
- Celular: varchar (10)
- Telefono de casa: varchar(150)
- Calle, varchar(100)
- Numero: varchar(150)

- Colonia: varchar(150)
 - CodigoPostal: varchar(5)
 - Analizar estándar...
 - Municipio: varchar(100)
 - Estado(80),
 - Municipio:varchar(100)
- El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF36.** El sistema debe de permitir al usuario de nivel ADMINISTRADOR o CAJERO, llevar a cabo un proceso de INGRESO por Autenticación, tomando como credenciales su Correo Electronico y su contraseña (encriptada). Los campos que se requieren para esta finalidad, son:

- Correo Electrónico : string, carchar(120)
 - Contraseña: varchar(100) //cifra 123456
 - acasdcasdnicbeubewqhncpi
- El sistema al permitir el ingreso autenticado, despliega las ventas y módulos AUTORIZADOS para dicho usuario y su TIPO.
- es cajero: le muestra la ventana de caja
 - es admin: le muestra DASHBOARD de admin.
- El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

6. Modulo Existencias

□**RF37.** El sistema debe de permitir al usuario de tipo ADMINISTRADOR movimientos(ENTRADAS, VENTAS-automático en caja, MERMAS, DEVOLUCIONES_A_PROVEEDOR, DEVOLUCIONES_DE_CLIENTES) de EXISTENCIAS de sus productos a ALMACEN del comercio. Los campos que se requieren para la esta funcionalidad, son:

- Almacén: Es una colección de espacios donde organizan los productos
 - Producto(id)
 - Cantidades (UNIDADES DE MEDIDAS)
 - Costo_por_Unidad: para determinar PRECIO_DE_VENTA
 - Fecha inicial - Fecha final
 - Hora inicial - Hora Final
 - El proveedor al que compramos (texto) (puede ser null)
 - TipoDeMovimiento: Entrada, Venta, Merma, Devolución (a proveedor, de cliente)
 - Cliente que devuelve (puede ser null)
 - Activo, por valores activo o inactivo.
- El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

□**RF38.** El sistema debe de permitir al usuario de tipo ADMINISTRADOR . Los campos que se requieren para la esta funcionalidad, son:

- Almacén: Es una colección de espacios donde organizan los productos
- Producto(id)

- Cantidades (UNIDADES DE MEDIDAS)
- Costo_por_Unidad: para determinar PRECIO_DE_VENTA
- Fecha inicial - Fecha final
- Hora inicial - Hora Final
- El proveedor al que compramos (texto) (puede ser null)
- TipoDeMovimiento: Entrada, Venta, Merma, Devolución (a proveedor, de cliente)
- Cliente que devuelve (puede ser null)
- Activo, por valores activo o inactivo.
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

❑RF39. El sistema debe de permitir al usuario de tipo ADMINISTRADOR CANCELAR movimientos(ENTRADAS, VENTAS-automático en caja, MERMAS, DEVOLUCIONES_A_PROVEEDOR, DEVOLUCIONES_DE_CLIENTES) de EXISTENCIAS de sus productos a ALMACEN del comercio. Los campos que se requieren para la esta funcionalidad, son:

- Activo, pasaría a FALSE
El sistema notificará al usuario con una ventana de mensajes, si el registro es correcto, o si ocurrió algún error.

3. Requisitos no funcionales

GLOBAL

3.1. Requisitos de Interfaz Gráfica

- 3.1.1.Colores del sistema.
- 3.1.2.Tipografía (Tamaño de fuentes en los distintos campos).
- 3.1.3.Logotipo (Versiones Tamaños, formatos, color)
- 3.1.4.Estilos de ventana
- 3.1.5.LAYOUT (Distribución de los componentes)
- 3.1.6.Comportamiento o Animación gráfica

POR MODULOS

Productos

Crear productos.

3.2. Requisitos de rendimiento

RNF-1.- El número de usuarios que debe soportar el sistema de forma concurrente será de al menos 3.

RNF-2.- El tiempo de respuesta máximo por transacción será de tres (3) segundos.

RNF-3 - El número de usuarios que debe soportar el sistema de forma concurrente será de 1 usuario por equipo con software instalado y ejecutando en LAN al menos 1 y máximo 5 equipos de computo 2 Admin Almacén 3 max de CAJA.

RNF - 4 - El tiempo de respuesta máximo por transacción será de tres (3) segundos MÁXIMO.

3.3. **Seguridad**

RNF-5 - El usuario debe iniciar sesión para poder acceder a las funciones que lo requieran, para lo cual debe poseer nombre de usuario(correo electrónico) y contraseñas. El admin es responsable de crear nuevos usuarios.

RNF - 6 ' Las claves de usuario deben ser almacenadas.

3. **Fiabilidad**

ETAPAS DEL CICLO DE VIDA DEL SOFTWARE
ANÁLISIS (requisitos)
DISEÑO (esquematizar la solución)
DESARROLLO/ CODIFICACIÓN
PRUEBAS, de los REQUISITOS Verificar y validar la experiencia del usuario (Expectativas)
ENTREGA NEW FEATURES PROPUESTAS

4. **Disponibilidad**

RNF-7.- El sistema estará disponible siempre que el equipo de cómputo esté activo, el sistema estará ejecutándose. El horario de atención del negocio que use este PDV definirá esta disponibilidad.

5. **Mantenibilidad**

RNF-8.- El sistema contará una garantía de Mantenimiento de Software de dos cambios por año, incluidos en el precio de venta. Si el cliente desea más características nuevas, o cambios, se le llevará a cabo un anteproyecto y una cotización que los incluya. Las CORRECCIONES se corrigen en el tiempo disponible de mantenimiento, inmediato a su identificación.

6. **Portabilidad**



RNF-9.- El sistema será diseñado NO portable, siendo SOLO compatible con sistemas Windows versión 10 en adelante, y que tengan instalado el framework empleado para el desarrollo .NET 5.1.8.

1.4 Otros requisitos