JFS INVENTORY POS

Software Requirements Specification

version 0.1

05/08/2015

Sergio C. Norato

Jordan A. Cortes

Carlos F. Mora

Líderes Técnicos

Elaborado para

Instructor: Luis Alejandro Bernal Romero, MsC.

2015

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| <fecha> | <Version 1> | <Your Name> | <First Revision> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | <Your Name> | Lead Software Eng. |  |
|  | Luis Alejandro Bernal | MsC, |  |
|  |  |  |  |

**Table of Contents**

**REVISION HISTORY**

**DOCUMENT APPROVAL**

**1. INTRODUCTION**

1.1 Purpose

1.2 Scope

1.3 Definitions, Acronyms, and Abbreviations

1.4 References

1.5 Overview

**2. GENERAL DESCRIPTION**

2.1 Product Perspective

2.2 Product Functions

2.3 User Characteristics

**3. SPECIFIC REQUIREMENTS**

3.1 External Interface Requirements

*3.1.1 User Interfaces*

*3.1.2 Hardware Interfaces*

*3.1.3 Software Interfaces*

*3.1.4 Communications Interfaces*

3.2 Functional Requirements

*3.2.1 <Functional Requirement or Feature #1>*

*3.2.2 <Functional Requirement or Feature #2>*

3.3 Use Cases

*3.3.1 Use Case #1*

*3.3.2 Use Case #2*

3.4 Classes / Objects

*3.4.1 <Class / Object #1>*

*3.4.2 <Class / Object #2>*

3.5 Non-Functional Requirements

*3.5.1 Performance*

*3.5.2 Reliability*

*3.5.3 Availability*

*3.5.4 Security*

*3.5.5 Maintainability*

*3.5.6 Portability*

3.6 Inverse Requirements

3.7 Design Constraints

3.8 Logical Database Requirements

3.9 Other Requirements

**4. ANALYSIS MODELS**

4.1 Sequence Diagrams

4.3 Data Flow Diagrams (DFD)

4.2 State-Transition Diagrams (STD)

**5. CHANGE MANAGEMENT PROCESS**

**A. APPENDICES**

A.1 Appendix 1

# 1. Introduction

Este proyecto se basa en el desarrollo de una aplicación que permita llevar el control tanto de **recetas**, **insumos** y **stock** en un punto de venta especializado en helado. El control de stock es importante para tener mayor certeza el cuando es necesario hacer la solicitud al proveedor, también cuales son los productos y cantidades necesarias a solicitar. Adicionalmente, el control de recetas e insumos permite a la heladería realizar consultas para identificar hábitos de consumo.

## 1.1 Purpose

El propósito del proyecto se enfoca en dos conceptos muy claros; control de stock y tendencias de consumo. Ambos serán una herramienta para toma de decisiones a los administradores en un punto de venta especializado en helados, así como lo son los puntos de venta de helado de creppes & waffles.

## 1.2 Scope

* El nombre del proyecto se define como: JFS INVENTORY POS
* Permitirá al proveedor el ingreso de insumos al sistema
* Permitirá al administrador el ingreso de recetas trabajadas en el punto de venta.
* Permitirá al cajero el registro de pedidos de los usuarios.
* Permitirá al administrador y proveedor conocer en tiempo real el estado del stock de la heladería
* Permitirá al administrador conocer en tiempo real las cantidades a solicitar de un determinado insumo.
* Permitirá que un administrador genere una consulta con un rango de tiempo determinado donde podrá determinar:
  + recetas mas o menos consumidas en tiempo real
  + sabores más o menos consumidos ( a través de los insumos consumidos) en tiempo real
* Como el sistema le permite al administrador ver las tendencias en los productos vendidos o consumidos podrá generar nuevas estrategias o promociones para sus clientes.

## 1.3 Definitions, Acronyms, and Abbreviations

**Receta**: Nota que comprende aquello de que debe componerse algo, y el modo de hacerlo.

**JFS**: (Jordan, Fernando y Sergio).

**Inventary**: Proceso organizado de bienes, productos entre otros que pertenecen a una persona, empresa o institución.

**insumo**: Conjunto de bienes empleados en la producción de otros bienes*.*

## 1.4 References

http://www.rae.es/

## 1.5 Overview

Este Documento contendrá los requerimientos del software para satisfacer al cliente heladería de crepes & waffles y brindarle una solución óptima en cuanto al manejo del inventario, consultas y registros de productos, insumos, recetas entre otros.

En donde se respetaran los estándares o reglas que tenga la empresa y solamente de verá afectado la parte anteriormente mencionada para beneficio de la misma.

# 2. General Description

## 2.1 Product Perspective

*El producto de software desarrollado para el punto de venta de la heladería crepes & waffles comprende la lógica de negocio más eficaz comparado con otros productos de la misma categoría, debido a que va poseer eficacia y a la misma vez simple para un uso adecuado y se mantengan el controlados los insumos y recetas que se adicionen en el sistema.*

## 2.2 Product Functions

*El producto contará con la capacidad o funcionalidad de mantener el inventario de los productos que venderá la heladería y a la vez avisara al encargado del inventario de cual producto debe recargarse y evitar que se hagan las operaciones manualmente. Por otro lado también se encargará de registrar y consultar insumos, recetas entre otros. Lo anterior ayudara a generar estadísticas y reportes útiles para que el gerente de la heladería pueda observar el rendimiento de la franquicia a cargo.*

## 2.3 User Characteristics

*Los usuarios a utilizar el producto deben comprender la definición de inventario, insumos y recetas porque como se ha explicado antes tendrá las opciones de realizar reportes o estadísticas por ende es necesario que los usuarios tengan claras las ideas de cómo se calcula, tener conocimientos en ofimática y buen manejo de un computador para realizar un uso adecuado a la aplicación o software.*

# 3. Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

Se va a disponer de las siguientes interfaces

* CAJERO: Permite el registro de pedidos de los clientes. Esta interfaz será de carga rápida y muy sencilla para el uso del cajero, la cantidad de clic para su operación debe ser mínima.
* PROVEEDOR: Permite la consulta e ingreso de insumos al sistema.
* ADMINISTRADOR: Permite la consulta e ingreso de recetas al sistema.
* CONSULTAS: Permite el ingreso a las diversas consultas del sistema.

### 3.1.2 Hardware Interfaces

N/A

### 3.1.3 Software Interfaces

El sistema óptimo para que el software funcione sin ningún tipo de problemas son los siguientes:

Windows (Desde la versión 7).

Linux

Mac

### 3.1.4 Communications Interfaces

N/A

## 3.2 Functional Requirements

### 3.2.1 INGRESO DE INSUMO

3.2.1.1 Se permite el ingreso de un insumo nuevo en el sistema

3.2.1.2 Inputs: nombre del insumo, código del insumo, unidad de medida del insumo, cantidad del insumo.

3.2.1.3 Processing: Se validan los valores de entrada, y se registran en el sistema

3.2.1.4 Outputs: mensaje de registro exitoso

3.2.1.5 Error Handling: Un mensaje indicando el origen del error

### 3.2.2 CONSULTA DE INSUMOS

3.2.2.1 Se permitirá la consulta de insumos por medio de filtros

3.2.2.2 Inputs: Para la búsqueda del insumo se debe realizar por medio del nombre o código que tenga determinada materia prima.

3.2.2.3 Processing: El encargado de área podrá observar los insumos disponibles filtrados por cantidad, unidad y nombre.

3.2.2.4 Outputs: Se visualizara los insumos que el encargado filtro según especificaciones que selecciono.

3.2.2.5 Error Handling: Mostrará un mensaje si entre el rango que seleccioné el encargado no se encuentra ningún insumo registrado.

**3.2.3 RECARGA DE INSUMO**

3.2.3.1 Se permitirá la recarga de insumos cuando este se encuentre estado crítico.

3.2.3.2 Inputs: Código, cantidad y unidad del insumo.

3.2.3.3 Processing: El administrador deberá adicionar la cantidad del insumo necesario para un tiempo determinado.

3.2.3.4 Outputs: El insumo aumentara la cantidad y estará disponible para ser usado en la elaboración de alguno de los productos que lo necesite.

3.2.3.5 Error Handling: Mostrará un mensaje de error si la cantidad de ingreso es invalida o superar los estándares de el software.

**3.2.4 CREAR RECETA**

3.2.4.1 Introduction: Permite crear la lista de ingredientes y proporciones de una nueva receta

3.2.4.2 Inputs: lista de insumos necesarios para la receta, nombre de la receta, código de la receta

3.2.4.3 Processing: el administrador selecciona los insumos necesarios, luego determina las cantidades de cada uno y finalmente lo crea.

3.2.4.4 Outputs: mensaje de receta creada

3.2.4.5 Error Handling: Un mensaje indicando el origen del error

**3.2.5 CONSULTAR RECETAS**

3.2.5.1 Se permitirá la consulta de las recetas de la heladería por medio de filtros

3.2.5.2 Inputs: Para la búsqueda de las recetas se debe realizar por medio del nombre o código que tenga determinada cada una.

3.2.5.3 Processing: El encargado de área deberá seleccionar el filtro predeterminado por código o nombre de una receta en particular

3.2.5.4 Outputs: Se visualizara la receta que haya buscado anteriormente numerando los insumos utilizados y la cantidad necesaria para preparar el producto.

3.2.5.5 Error Handling: Mostrará un mensaje si el código ingresado no se encuentra registrado en el sistema.

**3.2.6 REGISTRAR PEDIDO**

3.2.6.1 Introduction: el cajero hace una lista de las recetas que un cliente le solicita para registrarlas en el sistema

3.2.6.2 Inputs: lista de recetas

3.2.6.3 Processing: el cajero selecciona de la lista de recetas una receta solicitada por el cliente, luego asigna cantidad solicitada. El cajero debe tener la opción de retirar una receta por decisión del cliente. Finalmente el cajero registra la solicitud del cliente en el sistema.

3.2.6.4 Outputs: mensaje de solicitud creada

3.2.6.5 Error Handling: Un mensaje indicando el origen del error

**3.2.7 CONSULTAR PEDIDOS**

3.2.7.1 Introduction: El administrador podrá consultar los pedidos que fueron solicitados por los clientes.

3.2.7.2 Inputs: rango de fechas en la cual se desea hacer la consulta

3.2.7.3 Processing: se ingresan los rangos de fechas y se solicita la consulta

3.2.7.4 Outputs: Se visualizan los productos que están dentro del rango de fechas solicitadas

3.2.7.5 Error Handling: Un mensaje indicando el origen del error.

## 3.3 Use Cases

### 3.3.1 Use Case #1

### 3.3.2 Use Case #2

…

## 3.4 Classes / Objects

### 3.4.1 <Class / Object #1>

3.4.1.1 Attributes

3.4.1.2 Functions

<Reference to functional requirements and/or use cases>

### 3.4.2 <Class / Object #2>

…

## 3.5 Non-Functional Requirements

*Non-functional requirements may exist for the following attributes. Often these requirements must be achieved at a system-wide level rather than at a unit level. State the requirements in the following sections in measurable terms (e.g., 95% of transaction shall be processed in less than a second, system downtime may not exceed 1 minute per day, > 30 day MTBF value, etc).*

### 3.5.1 Performance

### 3.5.2 Reliability

### 3.5.3 Availability

### 3.5.4 Security

### 3.5.5 Maintainability

### 3.5.6 Portability

## 3.6 Inverse Requirements

*State any \*useful\* inverse requirements.*

## 3.7 Design Constraints

*Specify design constrains imposed by other standards, company policies, hardware limitation, etc. that will impact this software project.*

## 3.8 Logical Database Requirements

*Will a database be used? If so, what logical requirements exist for data formats, storage capabilities, data retention, data integrity, etc.*

## 3.9 Other Requirements

*Catchall section for any additional requirements.*

# 4. Analysis Models

*List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS’s requirements.*

## 4.1 Sequence Diagrams

## 4.3 Data Flow Diagrams (DFD)

## 4.2 State-Transition Diagrams (STD)

# 5. Change Management Process

*Identify and describe the process that will be used to update the SRS, as needed, when project scope or requirements change. Who can submit changes and by what means, and how will these changes be approved.*

# A. Appendices

*Appendices may be used to provide additional (and hopefully helpful) information. If present, the SRS should explicitly state whether the information contained within an appendix is to be considered as a part of the SRS’s overall set of requirements.*

*Example Appendices could include (initial) conceptual documents for the software project, marketing materials, minutes of meetings with the customer(s), etc.*

## A.1 Appendix 1

## A.2 Appendix 2