Especificación de Requisitos de Software

para

Mercedario Recetas

Versión 1.0 revisión pendiente

Hecha por Antonio Parra, Cristian Orbes y Sebastián Ramos

Universidad Cooperativa de Colombia (Sede pasto)

6 de septiembre de 2023

Tabla de contenido

1. Introducción 1

1.1 Propósito 1

1.2 Convenciones del documento 1

1.3 Audiencia destinada y sugerencias de lectura. 1

1.4 Visualización del producto. 1

1.5 Referencias 2

2. Descripción general 2

2.1 Perspectiva del producto 2

2.2 Funciones del producto 2

2.3 Clases de usuario y características 2

2.4 Entorno operativo 3

2.5 Limitaciones de diseño e implementación 3

2.6 Documentación de Usuario 3

2.7 Asunciones y/o dependencias 3

3. Requerimientos de interfaces externas 3

3.1 Interfaces de Usuario 3

3.2 Interfaces de Hardware 4

3.3 Interfaces de software 4

3.4 Interfaces de comunicación. 4

4. Características del Sistema 4

4.1 Sistema de cuentas 4

4.2 Pantalla de recetas 5

4.3 Pantalla de ingredientes 7

4.4 Pantalla de permisos 8

5. Requisitos no funcionales 9

5.1 Listado de requisitos no funcionales 9

5.2 Requisitos no funcionales específicos. 10

5.3 Escenarios de calidad 10

5.4 Árbol de calidad 12

Historial de Revisión

|  |  |  |  |
| --- | --- | --- | --- |
| **Nombre** | **Fecha** | **Razón de cambio** | **Versión** |
| Antonio Parra | 6/09 | Agregando contenido | 0.1 |
| Antonio Parra | 27/09 | Traduciendo y arreglando contenido | 1.0 |
| Antonio Parra | 28/09 | Agregando Características del sistema | 1.0 |
| Antonio Parra | 11/10 | Corrigiendo Características del sistema | 1.0 |
| Antonio Parra | 9/11 | Corrigiendo Historias de usuario | 1.0 |
| Antonio Parra | 14/11 | Agregando requisitos no funcionales. | 1.0 |

# Introducción

## Propósito

Este documento cubre la primera versión del sistema, una aún en revisión para verificar su correcto funcionamiento y documentación. El producto a entregar se trata de un software encargado para el restaurante Mercedario Recetas, este software está planeado para ser usado únicamente por el personal del restaurante involucrado en la preparación de los distintos platos servidos en el mismo.

## Convenciones del documento

Este documento utiliza las siguientes convenciones para el texto mostrado en él:

**Negrilla:** Este documento utiliza la convención de negrilla para indicar títulos y subtítulos a lo largo del documento.

Cursiva: Este documento utiliza la convención de cursiva en referencias hacia sitios externos o libros/obras a las que se necesite citar o referenciar, también se utiliza con vínculos a páginas web.

­­­­­Subrayado: Este documento utiliza la convención de subrayado únicamente en vínculos hacia paginas externas.

Letra:

Times: Este documento utiliza el tipo de letra “Times” en títulos y la tabla de contenido del documento.

Arial: Este documento utiliza el tipo de letra “Arial” en párrafos y texto común que no necesita ser resaltado.

## Audiencia destinada y sugerencias de lectura.

Este documento está destinado a las manos del equipo de desarrollo, al propietario del producto a entregar, al usuario común y al usuario recurrente, al ingeniero destinado a realizar mantenimiento del software entregado y al administrador del proyecto. Tomad este documento como una guía de lo que van a encontrar en el programa entregado a la empresa.

## Visualización del producto.

Mercedario recetas necesitaba un software que le ayudase a gestionar la preparación de sus recetas y controlar los ingredientes utilizados en cada receta, se necesitaba un sistema que le permitiera sus usuarios observar las recetas, los ingredientes utilizados y la forma de preparar los platos, de manera que tuvieran a su disposición todas las características necesarias a la hora de cocinar los platos.

## Referencias

La especificación de los requisitos del producto se encontrará en la hoja de cálculo “Tablas de requisitos” realizada en Excel y encontrada en la carpeta en línea del proyecto Mercedario Recetas. Los diagramas del producto se encontrarán en la carpeta en línea del proyecto Mercedario Recetas, Graficas, en esta se encontrarán los diagramas de flujo, contexto y clase respectivamente.

Repositorio de GitHub:

<https://github.com/AntBPV/Ingenieria_de_requisitos>

Requirements Engineering and Management for Software Development Projects:

<https://books.google.com.co/books?id=K3SYPgrZ6-wC&lpg=PR7&ots=865Vt3wG1x&dq=Requirements%20Engineering%20and%20Management%20for%20Software%20Development%20Projects&lr&hl=es&pg=PR3#v=onepage&q=Requirements%20Engineering%20and%20Management%20for%20Software%20Development%20Projects&f=false>

# Descripción general

## Perspectiva del producto

El diagrama de contexto que otorgara la perspectiva que se tiene del producto estará disponible en la wiki del repositorio de GitHub vinculado a continuación:

<https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Documento-SRS-(Mercedario_Recetas)#diagrama-de-contexto>

## Funciones del producto

El diagrama de clase que permitirá la visualización de las funciones que tiene el producto estará disponible en la wiki del repositorio de GitHub vinculado a continuación:

<https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Documento-SRS-(Mercedario_Recetas)#diagrama-de-clases>

## Clases de usuario y características

Asistente de cocina: Usuario común del programa, no tiene permisos especiales ni accesos limitados. Su función se limita a observar y preparar recetas.

Chef: Usuario especial del programa, sus permisos especiales radican en editar, agregar y eliminar cualquier receta guardada en sistema.

Distribuidor de insumos: El encargado de transportar los insumos al restaurante, es un usuario especial del programa. Sus permisos radican en editar el listado de ingredientes con sus respectivas características y cantidades.

Administrador: Usuario omnipotente en el programa, obtiene todos los permisos del sistema.

## Entorno operativo

El programa deberá ser ejecutado en un sistema Windows o mac con acceso a la consola de comandos, que, a su vez, deberá tener instalado el lenguaje de programación Python y su PATH para poder funcionar correctamente sin necesidad de instalar ningún editor de código o interprete.

## Limitaciones de diseño e implementación

El programa debe ser desarrollado exclusivamente en el lenguaje de programación Python. El programa va dirigido únicamente al personal del restaurante encargado de la preparación de los alimentos y aquellos que suministren al restaurante con insumos. Los datos deben ser guardados en el sistema en el que el programa opera. Ciertas funcionalidades, especificadas en los requisitos deben ser accesibles únicamente al obtener ciertos permisos especiales según su rol en el restaurante.

## Documentación de Usuario

El Usuario final recibirá un manual con las instrucciones de cómo funciona el programa, como acceder a las funciones respectivas de cada rol en el restaurante y como crear su cuenta para interactuar en cada momento con el programa.  
Se les dará una tutoría a los usuarios de cómo utilizar los aspectos básicos del software, como se asignará su rol en la empresa, y que funciones pueden utilizar los cargos especiales.

## Asunciones y/o dependencias

Se asume que el programa será utilizado por todo el personal responsable de preparar los alimentos del restaurante, es decir: Chef, distribuidor de insumos, asistentes de cocina, etc. El programa está planeado para recibir al empleado común como un usuario normal sin permisos especiales.  
El programa depende de que el rol de distribuidor de insumos, cumpla su parte en el programa al ingresar la cantidad y tipo de insumos que ingresan en el restaurante de manera correcta. Para asegurar que el programa cometa un error con los insumos disponibles, se recomienda que varios usuarios puedan editar la sección de insumos del programa.

Se asume que en cada receta se utilizará una cantidad en números enteros de insumos, y que no se registrarán como porciones o fracciones de un insumo ya utilizado. Es decir, no se espera que se reciclen o se utilicen las sobras de un insumo en una receta.

# Requerimientos de interfaces externas

## Interfaces de Usuario

El usuario, al momento de iniciar el programa se encontrará con distintas pantallas que le servirán de guía a través de todas las funciones que este ofrece, las imágenes conceptuales encontradas en la siguiente wiki servirán como guía conceptual sobre estas pantallas:

<https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)>

## Interfaces de Hardware

Las únicas interfaces de hardware que el programa necesita para comunicarse con el usuario se tratan de una computadora de oficina básica en la que el programa pueda ejecutarse y un teclado de escritura para computadora. Gracias a este el usuario se podrá desplazar por los menús y comunicar al sistema las entradas necesarias.

## Interfaces de software

El programa deberá ser ejecutado en un sistema operativo Windows o un sistema operativo mac el cual no se ha logrado especificar sus versiones compatibles.

En caso de usar Windows: Se recomienda utilizar las versiones 8.9 en adelante del sistema operativo. Para que el programa funcione se necesita instalar el lenguaje de programación Python completamente gratis desde la página oficial: <https://www.python.org>

El lenguaje deberá ser instalado en el PATH del sistema operativo para que el programa pueda ser ejecutado directamente desde la consola de sistema o CMD.

## Interfaces de comunicación.

No se conoce ninguna interfaz de comunicación en el proyecto.

# Características del Sistema

A continuación, se presentarán las características funcionales dentro del sistema en Mercedario Recetas, estas características se tratan de distintos procesos redactados de manera detallada para lograr obtener una vista general del producto a obtener, así mismo como poder entender el funcionamiento del mismo y como se ha ideado el proyecto.

## Sistema de cuentas

4.1.1 Ingreso de usuario (R-001).

4.1.1.1 HU-001 Iniciar sesión de usuario.

Yo como usuario quiero una interfaz para poder iniciar sesión.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#2-inicio-de-sesión>

Criterios de aceptación:

1. Existe la casilla con el nombre “nombre”.
2. Existe la casilla con el nombre “contraseña”.
3. Existe la opción para regresar a la pantalla anterior.
4. Si el usuario no esta registrado en sistema, un aviso saldrá en pantalla.

4.1.2 Registro de usuario (R-002)

4.1.2.1 HU-002 Registrar al usuario

Yo como usuario quiero una interfaz para registrarme en el programa.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#3-registro>

Criterios de aceptación:

1. Existe la casilla con el nombre “nombre”.
2. Existe la casilla con el nombre “contraseña”.
3. Existe la opción para regresar a la pantalla anterior.

4.1.3 Verificar rango de usuario (R-009)

4.1.3.1 HU-003 Verificar rango o identidad del usuario en sistema.

Yo como administrador quiero que el sistema verifique el rango de cada usuario para asegurar que las acciones restringidas estén protegidas.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#4-pantalla-principal>

Criterios de aceptación:

1. Cuando el usuario inicie sesión, se enseñará un saludo con su nombre y su rango en el sistema (chef, administrador o nada en caso de ser un usuario común).
2. El rango enseñado es igual al rango en sistema para el usuario presente.

## Pantalla de recetas

4.2.1 Las recetas se ven listadas (R-003)

4.2.1.1 HU-004 Listar recetas.

Yo como usuario quiero una interfaz de recetas para listar las recetas ya registradas.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#5-listado-de-recetas>

Criterios de aceptación:

1. Se visualizan todas las recetas ya registradas.
2. Existe la casilla, seleccionar receta.
3. Existen opciones no visibles para el usuario común: editar, agregar y eliminar receta.
4. Existe la opción para regresar a la pantalla anterior.

4.2.2 Características de recetas (R-006)

4.2.2.1 HU-005 Interfaz de características.

Yo como usuario quiero una interfaz propia de las recetas para observar todas las características que contienen las recetas.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#6-ver-receta>

Criterios de aceptación:

1. Es visible la siguiente información: Nombre de la receta, tiempo de preparación en horas, cantidad de personas a servir, ingredientes a utilizar
2. Existe la opción mostrar preparación.
3. Existe la opción para regresar a la pantalla anterior.

4.2.3 Consultar preparación (R-007)

4.2.3.1 HU-006 Preparación

Yo como usuario quiero una interfaz para la preparación de la receta.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#6-ver-receta>

Criterios de aceptación:

1. Se enseña en pantalla la siguiente información: Tiempo de preparación en horas, cantidad de personas a servir, ingredientes a utilizar, descripción detallada de su preparación.
2. Existe la opción planear preparación
3. Existe la opción para regresar a la pantalla anterior.

4.2.4 Planear preparación (R-008)

4.2.4.1 HU-007 Planeación

Yo como usuario quiero una interfaz para planear la preparación de una receta.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#7-planear-preparación>

Criterios de aceptación:

1. Existe la casilla con el nombre “cantidad de personas a servir”
2. Una vez ingresada la casilla “cantidad de personas a servir” el programa mostrara en pantalla la siguiente información: calorías por porción, costo de ingredientes, cantidad de ingredientes proporcional, descripción detallada de su preparación.
3. Existe la opción para regresar a la pantalla anterior.

4.2.5 Datos de recetas (R-011)

4.2.5.1 HU-008 Agregar recetas.

Yo como chef quiero una interfaz para agregar nuevas recetas.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#8-agregar-receta>

Criterios de aceptación:

1. Existe la casilla con el nombre “Nombre”.
2. Existe la casilla con el nombre “Tiempo de preparación en horas”.
3. Existe la casilla con el nombre “Cantidad de personas a servir”.
4. Existe la casilla con el nombre “Ingredientes utilizados”.
5. Existe la casilla con el nombre “Descripción detallada de su preparación”.
6. Una vez ingresadas todas las casillas, se enseñará en pantalla un aviso indicando que la receta ha sido añadida exitosamente.

4.2.5.2 HU-009 Editar recetas.

Yo como chef quiero una interfaz para editar recetas ya registradas.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#9-editar-receta>

Criterios de aceptación:

1. Se listarán las recetas con la siguiente información: Nombre.
2. Existe la casilla con el nombre “Seleccionar receta”.
3. Cuando se ingrese la casilla “Seleccionar receta” se enseñará en pantalla la siguiente información: Nombre, tiempo de preparación en horas, cantidad de personas a servir, ingredientes utilizados, descripción detallada de su preparación. Según la receta seleccionada.
4. Existen casillas con el mismo nombre de la información anteriormente enseñada. (Nombre, tiempo de preparación en horas, etc.).
5. Una vez ingresadas todas las casillas se enseñará en pantalla la información de la receta actualizada.

4.2.5.3 HU-010 Eliminar recetas.

Yo como chef quiero una interfaz para eliminar recetas en sistema.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#10-eliminar-receta>

Criterios de aceptación:

1. Se listarán las recetas con la siguiente información: Nombre, ingredientes utilizados.
2. Existe la casilla con el nombre “Seleccionar receta”.
3. Una vez ingresada la casilla “Seleccionar receta” saldrá en pantalla un aviso indicando la receta que va a borrar de sistema.
4. El usuario debe tener visibles las opciones, borrar y cancelar.
5. En caso de seleccionar cancelar, se regresa a la casilla “Seleccionar receta”.
6. En caso de seleccionar borrar, se enseñará un aviso indicando que la receta fue borrada exitosamente.

## Pantalla de ingredientes

4.3.1 Los ingredientes se ven listados (R-004)

4.3.1.1 HU-011 Listar ingredientes.

Yo como usuario quiero una interfaz para listar todos los ingredientes registrados en sistema.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#11-listado-de-ingredientes>

Criterios de aceptación:

1. Se visualizarán todos los ingredientes ya registrados en sistema.
2. Existe la casilla con el nombre “Seleccionar ingrediente”.
3. Existen opciones no visibles para el usuario común: editar, agregar y eliminar ingrediente.
4. Existe la opción para regresar a la pantalla anterior.

4.3.2 Características de ingredientes (R-005)

4.3.2.1 HU-012 Interfaz de características.

Yo como usuario quiero una interfaz propia de los ingredientes para ver todas las características que contienen los ingredientes.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#12-ver-ingredientes>

Criterios de aceptación:

1. Se podrá visualizar la siguiente información: nombre, unidades, valor por unidad, sitio de compra, calorías.
2. Existe la opción regresar a la pantalla anterior.

4.3.3 Editar datos de ingredientes (R-010)

4.3.3.1 HU-013 Agregar ingredientes.

Yo como chef quiero una interfaz para agregar ingredientes.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#13-agregar-ingrediente>

Criterios de aceptación:

1. Existen varias casillas con los siguientes nombres: “nombre”, “unidades”, “valor por unidad”, “sitio de compra”, “calorías”.
2. Una vez ingresadas todas las casillas, saldrá en pantalla un aviso indicado que el ingrediente se ha añadido correctamente.

4.3.3.2 HU-014 Editar ingredientes.

Yo como chef quiero una interfaz para editar ingredientes ya registrados.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#14-editar-ingrediente>

Criterios de aceptación:

1. Se listarán los ingredientes con la siguiente información: Nombre.
2. Existe la casilla con el nombre “Seleccionar ingrediente”.
3. Se enseñará en pantalla la siguiente información: nombre, unidad, valor por unidad, sitio de compra, calorías. Según el ingrediente seleccionado.
4. Existirán casillas con el nombre de la información antes enseñada. (Nombre, unidad, etc.)
5. Una vez todas las casillas hayan sido ingresadas, se enseñará en pantalla la información del ingrediente actualizada.

4.3.3.3 HU-015 Eliminar ingredientes.

Yo como chef quiero una interfaz para eliminar un ingrediente.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#15-eliminar-ingrediente>

Criterios de aceptación:

1. Se listarán los ingredientes con la siguiente información: Nombre.
2. Existe la casilla con el nombre “Seleccionar ingrediente”.
3. Una vez ingresada la casilla, “Seleccionar ingrediente” saldrá un aviso sobre la receta que está a punto de borrar.
4. Las opciones borrar y cancelar deben ser visibles para el usuario.
5. En caso de seleccionar cancelar, se regresa a la casilla “Seleccionar ingrediente”.
6. En caso de seleccionar borrar, se enseñará un aviso indicando que el ingrediente fue borrado correctamente.

## Pantalla de permisos

4.4.1 Interfaz de administrador (R-012)

4.4.1.1 HU-016 Listar usuarios

Yo como administrador quiero una interfaz para observar los usuarios registrados.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#16-interfaz-de-administrador>

Criterios de aceptación:

1. Se listará los usuarios registrados en sistema con la siguiente información: Nombre, rango.
2. Existe la casilla con el nombre “Seleccionar usuario”.
3. Existe la opción para regresar a la pantalla anterior.

4.4.2 Enviar solicitudes (R-013)

4.4.2.1 HU-017 Solicitar rango

Yo como usuario quiero una interfaz para mandar una solicitud al administrador solicitando un nuevo rango.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#20-solicitud-de-permisos>

Criterios de aceptación:

1. En el menú de selección, al seleccionar la opción permisos, saldrá una casilla con el nombre “Mandar solicitud”.
2. Una vez ingresada la casilla “Mandar solicitud” el mensaje ingresado será guardado para enseñar al administrador en su siguiente inicio de sesión.

4.4.3 Editar datos de permisos (R-014)

4.4.3.1 HU-018 Otorgar rango

Yo como administrador quiero una interfaz para otorgar un rango a un usuario.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#18-otorgar-permisos>

Criterios de aceptación:

1. Desde la casilla referenciada en la historia de usuario HU-016, se muestra en pantalla la siguiente información: Nombre, rango. Según el usuario seleccionado.
2. Se muestra en pantalla la opción otorgar rango o eliminar rango.
3. Al seleccionar otorgar rango se muestra en pantalla los diferentes rangos disponibles: Chef o administrador.
4. Existe la casilla con el nombre “Rango a otorgar”.
5. Al ingresar la casilla “Rango a otorgar” se le da al usuario seleccionado el rango acorde a la casilla.
6. Existe la opción para regresar a la pantalla anterior.

4.4.3.2 HU-019 Eliminar rango

Yo como administrador quiero una interfaz para eliminar el rango de un usuario.

Mockup: <https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Mercedario_Recetas-(Mockups)#19-eliminar-permisos>

Criterios de aceptación:

1. Desde la casilla referenciada en la historia de usuario HU-016, se muestra en pantalla la siguiente información: Nombre, rango. Según el usuario seleccionado.
2. Se muestra en pantalla la opción otorgar rango o eliminar rango.
3. Al seleccionar eliminar aparecerá en pantalla un aviso indicando el usuario seleccionado
4. Las opciones borrar y cancelar deben ser visibles para el usuario.
5. Al seleccionar cancelar, se regresa a la casilla de seleccionar usuario de la historia de usuario HU-016.
6. Al seleccionar borrar, saldrá en pantalla un aviso indicando que el rango fue correctamente eliminado.
7. Existe la opción para regresar a la pantalla anterior.

# Requisitos no funcionales

A continuación, se presentarán distintos requisitos no funcionales, así como su impacto en la arquitectura del programa. Las imágenes se encontrarán en su respectivo link en el repositorio de GitHub donde un link lo llevará directamente a su visualización.

## Listado de requisitos no funcionales

* NF-001: El sistema tiene que estar altamente disponible de lunes a viernes entre las 7a.m y 8p.m.
* NF-002: El sistema debe garantizar la identidad del usuario que ingrese a realizar operaciones en el sistema. El sistema prohíbe la entrada de usuarios no identificados en la base datos.
* NF-003: El sistema debe garantizar la confidencialidad de los datos transmitidos desde las maquinas clientes y el servidor.
* NF-004: El sistema debe ser capaz de procesar entre 30 y 100 operaciones por minuto.
* NF-005: El programa tiene que ser intuitivo para cualquier usuario.
* NF-006: La información de la base de datos debe ser rápidamente editada cuando se indique la acción deseada (editar datos, agregar datos, eliminar datos).

## Requisitos no funcionales específicos.

* Disponibilidad - NF-001: El sistema tiene que estar altamente disponible de lunes a viernes entre las 7a.m y 8p.m.
* Seguridad - NF-002: El sistema debe garantizar la identidad del usuario que ingrese a realizar operaciones en el sistema. El sistema prohíbe la entrada de usuarios no identificados en la base datos.
* Seguridad - NF-003: El sistema debe garantizar la confidencialidad de los datos transmitidos desde las maquinas clientes y el servidor.
* Escalabilidad - NF-004: El sistema debe ser capaz de procesar entre 30 y 100 operaciones por minuto.
* Usabilidad - NF-005: El programa tiene que ser intuitivo para cualquier usuario.
* Desempeño – NF-006: La información de la base de datos debe ser rápidamente editada cuando se indique la acción deseada (editar dato, agregar datos, eliminar datos).

## Escenarios de calidad

5.3.1 NF-001

|  |  |  |  |
| --- | --- | --- | --- |
| **EC-001 (NF-001)** | **Disponibilidad** | **Version: 1.0** | |
| **Fuente** | **Estimulo** | **Artefacto** | **Ambiente** |
| Usuario | Gran cantidad de operaciones entre las 7a.m y las 8p.m | Sistema | Operación normal |
| **Respuesta** | | **Medida de la respuesta** | |
| El sistema debe ser capaz de completar la gran mayoría de peticiones a las horas indicadas. | | 85% de las peticiones. | |

5.3.2 NF-002

|  |  |  |  |
| --- | --- | --- | --- |
| **EC-002 (NF-002)** | **Seguridad** | **Version: 1.0** | |
| **Fuente** | **Estimulo** | **Artefacto** | **Ambiente** |
| Usuario | Intento de inicio de sesión. | Sistema. | Operación normal. |
| **Respuesta** | | **Medida de la respuesta** | |
| El sistema verifica que el usuario y contraseña coincide con los datos de sistema. Si coincide, aprueba la entrada. | | 100% de las veces. | |

|  |  |  |  |
| --- | --- | --- | --- |
| **EC-003 (NF-002)** | **Seguridad** | **Version: 1.0** | |
| **Fuente** | **Estimulo** | **Artefacto** | **Ambiente** |
| Usuario | Intento de inicio de sesión. | Sistema. | Operación normal. |
| **Respuesta** | | **Medida de la respuesta** | |
| El sistema verifica que el usuario y contraseña coincide con los datos de sistema. Si no coincide, evita la entrada a sistema del usuario. | | 100% de las veces. | |

5.3.3 NF-003

|  |  |  |  |
| --- | --- | --- | --- |
| **EC-004 (NF-003)** | **Seguridad** | **Version: 1.0** | |
| **Fuente** | **Estimulo** | **Artefacto** | **Ambiente** |
| Usuario | Ingreso de datos a la base de datos | Sistema | Operación normal. |
| **Respuesta** | | **Medida de la respuesta** | |
| Los datos obtenidos de los usuarios se mantienen en sistema, para uso único del sistema. | | 100% de las veces. | |

5.3.4 NF-004

|  |  |  |  |
| --- | --- | --- | --- |
| **EC-005 (NF-004)** | **Escalabilidad** | **Version: 1.0** | |
| **Fuente** | **Estimulo** | **Artefacto** | **Ambiente** |
| Usuario | Entre 30 y 100 operaciones ingresadas. | Sistema | Operación natural. |
| **Respuesta** | | **Medida de la respuesta** | |
| El sistema realiza exitosamente las operaciones ingresadas por el usuario. | | 100 operaciones en un minuto. | |

5.3.5 NF-005

|  |  |  |  |
| --- | --- | --- | --- |
| **EC-006 (NF-005)** | **Usabilidad** | **Version: 1.0** | |
| **Fuente** | **Estimulo** | **Artefacto** | **Ambiente** |
| Usuario | Uso del programa por el usuario. | Sistema | Operación normal |
| **Respuesta** | | **Medida de la respuesta** | |
| El usuario es capaz de usar el programa en su totalidad. | | 1 hora de uso exitoso sin capacitación. | |

5.3.6 NF-006

|  |  |  |  |
| --- | --- | --- | --- |
| **EC-007 (NF-006)** | **Desempeño** | **Version: 1.0** | |
| **Fuente** | **Estimulo** | **Artefacto** | **Ambiente** |
| Usuario | Ingreso, edición y eliminación de datos en el sistema. | Sistema | Operación normal |
| **Respuesta** | | **Medida de la respuesta** | |
| El sistema cambia rápidamente la información en la base de datos según la acción deseada. | | 1 a 3 segundos por acción. | |

## Árbol de calidad

Puede encontrar el árbol de utilidad para nuestros escenarios de calidad en nuestra wiki de GitHub, siguiendo el siguiente link encontrará el árbol de calidad correspondiente: [*https://github.com/AntBPV/Ingenieria\_de\_requisitos/wiki/Documento-SRS-(Mercedario\_Recetas)#arbol-de-calidad*](https://github.com/AntBPV/Ingenieria_de_requisitos/wiki/Documento-SRS-(Mercedario_Recetas)#arbol-de-calidad)