Grupo Drop Table

1

2

3

4

***Miembros del equipo***

**Nombre Apellidos**

Alejandro Pascua Piña

Andrea Martínez Fernández

An Wei Pham Luo

Beatriz Villegas Sánchez

Darío Fernando Gallegos Quishpe

Enrique Salazar Del Cid

Javier Martín-pozuelo Salvador

Jennifer Marmolejos Urbáez

Juan Martín Bárez Alonso

Manuel Oreja Valverde

Pedro Pablo Doménech Arellano

5

***Control de cambios***

**Número de versión**

**Fecha Autores Descripción**

1

02/11/201 6

Se planea y se reparte el trabajo sobre el SRS

2

Todos

04/11/201 6

Primera edición del documento SRS

3

Todos

Se ha ampliado la descripción general y correcciones menores de errores

4

05/11/201 6

Todos

06/11/201 6

Alejandro, Manuel, Beatriz, Javier

Se han añadido casos de uso

5

06/11/201 6

Se han añadido más casos de uso

6

El resto

Correcciones menores de errores y formato y diseño de UML

7

07/11/201 6

Todos

08/11/201 6

Todos (laboratorio) Diseño de UML

8

08/11/201 6

Todos Formato

9

An Wei, Alejandro, Manu, Bea, Dario, Jennifer, Juan Martín, Andrea

Modelo del dominio, interfaz gráfica, correcciones del documento SRS

10

08/12/201 6

An Wei, Alejandro,

13/12/201 6

Manu, Bea, Juan Martín, Andrea, Pedro, Enrique,

Varias correcciones

Javier

11

21/02/201 7

Manu

Casos de uso del módulo de establecimientos.

6

21/02/201 12

7

Bea, Jennifer

Casos de uso del módulo de productos

13

Casos de uso del módulo de Búsquedas

14

21/02/201 7

Javier, Darío, Enrique

Casos de uso del módulo de usuarios y descuentos

15

21/02/201

Andrea, Alejandro, An 7

Wei, Pedro

23/02/201 7

Todos Todo (revisión)

7

**I. A RTÍCULO**

**I**

**NTRODUCCIÓN**

**Sección 1.1 Propósito**

**Sección 1.2 Alcance**

**Sección 1.3 Definiciones, acrónimos y abreviaturas**

**Sección 1.4 Referencias**

**Sección 1.5 Organización del documento**

**A**

**RTÍCULO**

**II. D**

**ESCRIPCIÓN GENERAL**

**Sección 2.1 Perspectiva del producto**

**Sección 2.2 Funciones del producto**

**Sección 2.3 Características del usuario**

**Sección 2.4 Restricciones**

**Sección 2.5 Supuestos y dependencias**

**Sección 2.6 Requisitos futuros**

**A**

**RTÍCULO**

**III. R**

**EQUISITOS ESPECÍFICOS**

**Sección 3.1 Interfaces externos**

3.1.1 Interfaces hardware 3.1.2 Interfaces software 3.1.3 Interfaces de Comunicación

**Sección 3.2 Requisitos funcionales**

3.2.1 GESTIÓN DE ESTABLECIMIENTOS 3.2.2 GESTIÓN DE PRODUCTOS 3.2.3 GESTIÓN DE USUARIOS 3.2.4 USO DEL BUSCADOR

**Sección 3.3 Requisitos de rendimiento**

**Sección 3.4 Requisitos lógicos de la base de datos**

**Sección 3.5 R**

**ESTRICCIONES DE DISEÑO**

**Sección 3.6 A**

**TRIBUTOS DEL SISTEMA SOFTWARE**

8

**Sección 3.7 O**

**TROS REQUISITOS**

**A**

**RTÍCULO**

**IV. A**

**PÉNDICES**

9

**Artículo I. Introducción**

**Sección 1.1 Propósito**

En este documento SRS se especifican los requisitos y todos los detalles del proyecto Gas&Food que será entregado al cliente en Junio.

**Sección 1.2 Alcance**

Gas&Food está enfocada al usuario final para ofrecerle la posibilidad de buscar en una ubicación de la Comunidad de Madrid una serie de productos en las tiendas de alimentación, y del mismo modo, encontrar gasolineras y consultar el precio de sus gasolinas.

No todas las tiendas y productos existentes en la vida real estarán disponibles para el usuario.

Nuestra aplicación está destinada para usarse en un único ordenador del laboratorio y no en los ordenadores propios de cada usuario. No usará conexión a internet.

**Sección 1.3 Definiciones, acrónimos y abreviaturas**

Productos: Elementos de la aplicación que serán objeto de búsqueda, distinguiendo entre gasolinas y alimentos.

Establecimientos: Lugares donde se encuentras los productos, tiendas, o gasolineras.

Búsquedas: Acción de buscar un determinado producto y sus características en uno o varios establecimientos dentro de un área.

Usuario: Cualquier persona que se beneficie del servicio ofrecido por la aplicación final.

API: Application Programming Interface.

SRS: ‘Software Requirements Specification’

BBDD: ‘Base de Datos’ Conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

UML: ‘Unified Modeling Language’ Lenguaje gráfico para visualizar, especificar, construir y documentar un sistema.

LOPD: Ley Orgánica de Protección de Datos

10

OSM: OpenStreet Map

OSMAPI: Implementación en java de la API 0.6 de OSM

CP: Código postal

UML: Unified Modeling Language

MINETUR: Ministerio de Industria, Energía y Turismo

Admin: Administrador

IdEstablecimiento: Identificador de tienda/gasolinera.

Eg: Prefijo que usaremos en el código de IdEstablecimiento, para definirlo como gasolinera.

Ea: Prefijo que usaremos en el código de IdEstablecimiento, para definirlo como tienda de alimentación.

**Sección 1.4 Referencias**

Transparencias de clase

**Sección 1.5 Organización del documento**

Este documento se encarga de mostrar las especificaciones, su descripción detallada, requisitos y restricciones del diseño de nuestro producto.

11

**Artículo II.Descripción general**

**Sección 2.1 Perspectiva del producto**

Se desea desarrollar una aplicación cuya función principal sea comparar los precios de diversos productos de alimentación, en diferentes establecimientos, así como del precio de la gasolina en diferentes gasolineras.

Por lo que tendremos que gestionar establecimientos (tiendas y gasolineras), productos (comestibles y gasolinas), búsquedas, y usuarios.

**Sección 2.2 Funciones del producto**

Nuestra aplicación ofrece un servicio que mostrará un mapa y comparará los precios de distintas tiendas de alimentación y gasolineras de Madrid. Dichos locales y precios serán actualizados periódicamente.

No será necesario registrarse en nuestra aplicación, sin embargo, contaremos con un sistema de descuentos proporcionados a aquellos que sí lo estén. Además, utilizaremos los comentarios de los propios usuarios mediante un sistema de sugerencias para mantener actualizada la lista de precios y tiendas que se muestran.

El servicio se dividirá en 5 módulos:

- Usuarios: que contendrá las funciones de alta, baja y modificación de usuarios y

envío de sugerencias.

- Establecimientos: que contendrá la función de dar de alta, dar de baja

establecimientos.

- Productos: que contendrá las funciones para dar de alta, dar de baja, actualizar y

modificar los productos, añadir y eliminar un producto en una tienda, actualizar un descuento de un producto en una tienda.

- Descuento: que contendrá las funciones de alta, baja, modificación, mostrar por

pantalla, de asignar descuento a un usuario y de quitar descuento a un usuario.

- Búsquedas: que tendrá las funciones de búsqueda de un producto concreto (por

ID o nombre), comparación de productos según precio o según valoración, mostrar productos y valorar productos.

**Sección 2.3 Características del usuario**

La aplicación podrá ser usada por cualquier usuario, con un conocimiento mínimo en productos de alimentación, gasolina y precios, así como de la zona de Madrid. Por lo que podría no ser adecuada para personas demasiado jóvenes, que no tengan conocimiento sobre los productos. No se necesitarán conocimientos técnicos, sólo nociones básicas sobre cómo manejar el ordenador y lo dicho anteriormente.

12

**Sección 2.4 Restricciones**

La aplicación no tendrá en su base de datos todas las tiendas y gasolineras de Madrid.

La aplicación se usará en un único ordenador, el cual deberán utilizar los usuarios.

En el primer uso de la aplicación, ésta mostrará al usuario un acuerdo de uso comunicando que esta recoge su historial de ubicaciones y dándole la posibilidad de aceptarlo. De no ser así, la aplicación funcionará pero desactivará los servicios relacionados con el historial.

Los descuentos serán una característica que tendrá un producto de un determinado establecimiento y que solo será visible para aquellos usuarios que estén registrados.

La aplicación debe actuar siempre dentro de los marcos legales de la LOPD mientras recoge el historial de ubicaciones de los usuarios.

Las BBDD de la aplicación son gestionadas únicamente por el Administrador.

La aplicación será desarrollada en JAVA.

**Sección 2.5 Supuestos y dependencias**

Se informará debidamente al usuario del programa y/o sistema operativo que necesitará para el uso de la aplicación en desarrollo. En caso de que se produzca algún cambio se le notificará.

-Si el usuario no tiene acceso a internet no podrá tener acceso a información actualizada.

-En caso de que la carga de datos desde nuestra BBDD fallase la aplicación no funcionará.

-La aplicación dependerá de que podamos obtener información de MINETUR para que nuestra BBDD esté actualizada en el apartado de gasolineras.

-La actualización de la BBDD de alimentación dependerá del envío de sugerencias realizadas por los usuarios, además de aportaciones del administrador.

**Sección 2.6 Requisitos futuros**

Se desea en un futuro poder ampliar la aplicación insertando un mayor número de establecimientos de toda la Comunidad de Madrid. Además, tener la posibilidad de implementar un mapa mediante OSM, en el cual se podrán localizar todos los establecimientos registrados en la aplicación así como la ubicación del usuario, para poder representarlos en una búsqueda.

13

**Artículo III. Requisitos específicos**

**Sección 3.1 Interfaces externos**

**3.1.1 Interfaces hardware**

Nuestra aplicación está pensada para funcionar en un único ordenador de la Facultad de Informática de la Universidad Complutense de Madrid, el cual contendrá los ficheros que componen las distintas bases de datos necesarias para su funcionamiento.

**3.1.2 Interfaces software**

Nuestra aplicación necesitará ser ejecutada como mínimo en Windows 7 32/64 bits.

Al utilizar Java para su desarrollo necesitaremos también la instalación de Java junto con entorno de programación Eclipse.

**3.1.3 Interfaces de Comunicación**

La aplicación se comunicará con MINETUR para realizar una carga aditiva de las gasolineras. Finalmente esto no se implementará.

**Sección 3.2 Requisitos funcionales**

Esta sección se divide en 4 módulos principales:

● Gestión de establecimientos.

● Gestión de productos.

● Gestión de usuarios y descuentos.

● Gestión de búsquedas.

**3.2.1 GESTIÓN DE ESTABLECIMIENTOS**

La gestión de establecimientos se encargará de dar de alta, dar de baja y modificar establecimientos que permitirán relacionar los productos que ofrece con él mismo. En los establecimientos podemos identificar gasolineras y tiendas de alimentación que funcionan igual, excepto que la primera almacena gasolinas y la segunda alimentos. Para que dicha distinción sea llevada a cabo utilizaremos los id’s, que identifican a cada producto, no solo para identificar un producto o tienda del resto, sino también para identificar qué tipo de producto y establecimiento es, permitiendo sólo las relaciones definidas anteriormente (gasolina - gasolinera, tienda alimentación - alimento).

14

**3.2.1.1 Dar de alta establecimientos**

**CASO 3.2.1.1**

**DE USO**

Dar de alta un establecimiento

**Objetivo en contexto**

Añade un nuevo establecimiento a la BD (fichero de establecimientos) Entradas -Id del nuevo establecimiento

-Nombre del establecimiento -Dirección -CP Precondiciones 1o\_ El id, nombre y dirección no debe existir ya

2o\_ El id debe empezar por “eg” si es una gasolinera o por “ea” si es una tienda de alimentación Salidas Establecimiento añadido Postcondición si éxito

Mensaje de éxito: Establecimiento añadido

**Postcondición si fallo**

Mensaje de error: No se ha podido añadir el establecimiento Actores Administrador Secuencia normal

Paso Acción 1 Conectar a la BD e introducir todas las

entradas 2 Las entradas cumplen todas las

precondiciones 3 Establecimiento añadido

15

**3.2.1.2 Modificar establecimiento**

**Secuencia alternativa**

Paso Acción 1 Conectar a la BD e introducir todas las

entradas 2 Las entradas no cumplen todas las

precondiciones 3 Mensaje de error.

16

Campos nuevos

**CASO DE USO 3.2.1.2**

Modificar establecimiento

**Objetivo en contexto**

Cambiar los atributos de un establecimiento existente Entradas -Id del establecimiento a modificar

-Id nuevo -Nombre nuevo -Dirección nueva -CP nueva (de estos últimos 4 sólo es necesario 1) Precondiciones 1o\_id del establecimiento a modificar existente

2o\_si se modifica el id que empiece por ea o eg 3o\_ninguno de los atributos que se van a modificar pueden existir ya en otro establecimiento a excepción del CP Salidas Establecimiento modificado Postcondición si éxito

Mensaje de éxito: establecimiento modificado

**Postcondición si fallo**

Mensaje de error: error al modificar establecimiento

**Actores Administrador Secuencia normal**

Paso Acción 1 Conectar a la BD e introducir todas las

entradas necesarias

17

2 Las entradas cumplen todas las

precondiciones 3 Establecimiento modificado Secuencia alternativa

Paso Acción 1 Conectar a la BD e introducir todas las

entradas necesarias 2 Las entradas no cumplen todas las

precondiciones 3 Mensaje de error.

18

**3.2.1.3 Dar de baja establecimiento**

**Dar CASO DE USO**

**de baja establecimiento 3.2.2.5 Objetivo en contexto**

Dar de baja un establecimiento existente y sus relaciones existentes con otras entidades Entradas id del establecimiento Precondiciones id del establecimiento existente

19

Salidas Establecimiento eliminado además de todas las

relaciones entre dicho establecimiento y otra entidad Postcondición si éxito

Mensaje de éxito: establecimiento eliminado

**Postcondición si fallo**

Mensaje de error: No se ha podido eliminar el establecimiento (id inexistente) Actores Administrador Secuencia normal

Paso Acción 1 Conectar a la BD e introducir id 2 Id existente 3 Establecimiento eliminado Secuencia alternativa

Paso Acción 1 Conectar a la BD e introducir id 2 Id inexistente 3 Mensaje de error.

20

**3.2.2 GESTIÓN DE PRODUCTOS**

La gestión de este módulo se encarga de dar de alta, dar de baja, modificar y productos, además de relacionar cada producto con el establecimiento que lo ofrece y determinar su precio y descuento asociado. Es importante que se siga haciendo una división entre gasolina - gasolinera y tienda alimentación - alimento, como se indicó en la gestión de establecimientos.

**3.2.2.1 Dar de alta producto**

**Dar CASO DE USO**

**de alta un producto 3.2.1.1 Objetivo en contexto**

Añade un nuevo producto a la BD (fichero de productos) Entradas ID del nuevo producto

21

Nombre del producto Tipo (de gasolina si es una gasolina o de alimento si es un alimento) Precondiciones 1o El ID y nombre no deben existir ya

2oEl ID debe empezar por “pg” si es una gasolina o por “pa” si es un alimento Salidas Producto añadido Postcondición si éxito

Mensaje de éxito: Producto añadido

**Postcondición si fallo**

Mensaje de error: No se ha podido añadir el producto Actores Administrador Secuencia normal

Paso Acción 1 Conectar a la BD e introducir todas las

entradas 2 Las entradas cumplen todas las

precondiciones 3 Producto añadido Secuencia alternativa

Paso Acción 1 Conectar a la BD e introducir todas las

entradas 2 Las entradas no cumplen todas las

precondiciones 3 Mensaje de error.

22

**3.2.2.2 Modificar producto**

23

Campos nuevos

**CASO DE USO 3.2.2.2**

Modificar producto

**Objetivo en contexto**

Cambiar los atributos de un producto existente

Entradas -Id del producto a modificar

-Id nuevo -Nombre nuevo -Tipo nuevo (de estos últimos 3 sólo es necesario 1) Precondiciones 1o\_id del producto a modificar existente

2o\_si se modifica el id que empiece por pa o pg 3o\_ninguno de los atributos id y nombre que se modifiquen pueden existir ya en otro producto Salidas Producto modificado Postcondición si éxito

Mensaje de éxito: producto modificado

**Postcondición si fallo**

Mensaje de error: error al modificar producto

**Actores Administrador Secuencia normal**

Paso Acción 1 Conectar a la BD e introducir todas las

entradas necesarias 2 Las entradas cumplen todas las

precondiciones 3 Producto modificado

24

**Secuencia alternativa**

Paso Acción 1 Conectar a la BD e introducir todas las

entradas necesarias 2 Las entradas no cumplen todas las

precondiciones 3 Mensaje de error.

25

**3.2.2.3 Dar de baja producto**

26

**3.2.2.6 Actualizar precios de un producto en un establecimiento**

**CASO DE USO 3.2.2.6**

Actualizar precios

**Objetivo en contexto**

Mantener los productos actualizados

Entradas ID del producto, ID tienda

27

**Dar CASO DE USO**

**de baja producto 3.2.2.5 Objetivo en contexto**

Dar de baja un producto existente y sus relaciones existentes con otras entidades Entradas id del producto Precondiciones id del producto existente

Salidas producto eliminado además de todas las relaciones

entre dicho producto y otra entidad Postcondición si éxito

Mensaje de éxito: producto eliminado

**Postcondición si fallo**

Mensaje de error: No se ha podido eliminar el producto (id inexistente) Actores Administrador Secuencia normal

Paso Acción 1 Conectar a la BD e introducir id 2 Id existente 3 producto eliminado Secuencia alternativa

Paso Acción 1 Conectar a la BD e introducir id 2 Id inexistente 3 Mensaje de error.

Precondiciones ID del producto exista, ID de la tienda exista y,

además, que tengan relación. El precio del producto no sea el mismo que ya posee. Salidas Modifica, en caso de que todo sea correcto, el

precio de los productos. Postcondición si éxito

Modifica la lista de productos. Muestra mensaje de éxito: “El producto ha sido actualizado”. Postcondición si fallo

Avisa al administrador de que el cambio no se ha llevado a cabo, muestra pantalla de error. Muestra mensaje de error: “El producto no ha sido actualizado”. Actores Administrador Secuencia normal

Paso Acción 1 Conexión a la base de datos. 2 Comprobar que los datos del producto

existen. 3 Comprobar que el precio del producto es

distinto al que ya posee. 4 Comprobar que los datos de la tienda

existan. 5 Que el ID de la tienda y el ID del producto

tengan relación. 6 Edita producto 7 Mensaje de éxito Secuencia alternativa

Paso Acción 1 Conexión a la base de datos. 2 Comprobar que los datos del producto

existen. Si no existen. 3 Muestra mensaje de error. Paso Acción 1 Conexión a la base de datos. 2 Comprobar que el ID de la tienda exista. 3 Muestra mensaje de error. Paso Acción 1 Conexión a la base de datos. 2 Comprobar relación entre ID de la tienda

y el producto sea correcta. 3 Mostrar mensaje de error. Paso Acción 1 Conexión a la base de datos. 2 Comprobar que el precio del ID producto

sea distinto que el que se quiere cambiar. 3 Mostrar mensaje de error.

28

**3.2.2.7 Añadir producto a establecimiento.**

**CASO DE USO 3.2.2.7**

Añadir producto a establecimiento.

**Objetivo en contexto**

Añadir/Vincular un nuevo producto, si es posible, a la tienda/gasolinera indicada por el administrador. Entradas Datos del producto:

-IdProducto -IdEstablecimiento -Precio -Descuento Precondiciones - El producto y el establecimiento debe estar dado de

alta.

- Se comprueba si el producto y establecimiento son compatibles mediante su id (pa - ea y pg - eg).

-Se podrá añadir un nuevo producto si actualmente no existe en el establecimiento asignado. Salidas Se añade el producto a la tienda o gasolinera, con el

precio y descuento correspondientes. Postcondición si éxito

El nuevo producto aparecerá con todos sus datos en la tienda o gasolinera con la que se debía relacionar. Postcondición si fallo

Mensaje de error: No se pudo añadir el producto.

**Actores Administrador Secuencia normal Paso Acción**

1 Conectar a la BBDD 2 Introducir datos producto 3 Se cumplen todas las precondiciones,

29

**3.2.2.8 Eliminar producto de establecimiento.**

**CASO DE USO 3.2.2.8**

Eliminar producto de tienda.

**Objetivo en contexto**

Eliminar/Desvincular el producto de la tienda/gasolinera indicada por el administrador. Entradas -IdProducto

-IdEstablecimiento Precondiciones - El producto y el establecimiento debe estar dado de

alta.

- Se comprueba que el producto está relacionado con el establecimiento indicado (IdEstablecimiento).

30

menos la última. 4 Se añade el nuevo producto a la tienda. Secuencia alternativa

Paso Acción 1 Conectar a la BBDD 2 Introducir datos producto 3 No se cumple alguna de las

precondiciones, en el caso de la última, si se cumple. 4 Mensaje de error: No se pudo añadir el

producto.

**3.2.2.9 Actualizar descuento de producto.**

**CASO DE USO 3.2.2.9**

Actualizar el descuento aplicado a un producto ofrecido por una tienda Objetivo en contexto

Mantener los descuentos de los productos actualizados Entradas ID del producto, ID del descuento, ID de la tienda

31

Salidas El producto ya no está relacionado con ningún

establecimiento y se elimina. Postcondición si éxito

El producto se eliminará así como sus datos y su relación con el establecimiento. Postcondición si fallo

Mensaje de error: “El producto no ha sido eliminado.”

**Actores Administrador Secuencia normal Paso Acción**

1 Conectar a la BBDD 2 Introducir IdProducto e IdEstablecimiento. 3 Se cumplen las precondiciones. 4 Se elimina el producto, así como su vínculo

con el establecimiento. Secuencia alternativa

Paso Acción 1 Conectar a la BBDD 2 No se cumple una de las precondiciones. 3 Mensaje de error: “El producto no ha sido

eliminado.”

Precondiciones Los IDs del producto y tiendas deben existir en la base de datos, además deben tener una relación entre ambos. Salidas El descuento actualizado. Postcondición si éxito

El descuento que se aplicará al producto queda actualizado.Muestra mensaje de éxito: “El descuento ha sido actualizado.” Postcondición si fallo

Avisa al administrador de que el descuento no se ha actualizado, muestra pantalla de error: “El descuento no ha sido actualizado”. Actores Administrador Secuencia normal Paso Acción

1 Conexión a la base de datos. 2 Solicitud del ID descuento, ID del producto

y del ID de la tienda. 3 Comprobar relación entre la tienda, el

descuento y el producto 4 Actualiza el descuento del producto. Secuencia alternativa

Paso Acción 1 Conexión a la base de datos. 2 Comprobar que exista el ID de la tienda

exista. 3 Muestra mensaje de error. Paso Acción 1 Conexión a la base de datos. 2 Compruebo que el ID del producto exista. 3 Muestra mensaje de error Paso Acción 1 Conexión a la base de datos. 2 Compruebo que el ID del descuento exista. 3 Muestra mensaje de error. Paso Acción 1 Conexión a la base de datos. 2 Compruebo que exista una relación entre

el ID descuento, ID del producto y el ID de la tienda. 3 Muestra mensaje de error.

32

**3.2.3 GESTIÓN DE USUARIOS**

El objetivo de este apartado es la gestión de los usuarios de la aplicación. Los encargados de realizar la tarea de alta de usuarios ,envío de sugerencias y mostrar actualización de descuentos son los administradores. Para ello tendrán acceso a software permitiendo actualizar la base de datos y controlando que funcione correctamente.

Los administradores se encargan de atender a los sucesivos usuarios en los diferentes problemas que puedan tener.

33

**3.2.3.1 Alta de usuarios**

La solicitud de alta de usuario se realizará de forma automática a través de la aplicación. El administrador es el que confirma el alta de la solicitud.Una vez verificado el usuario podrá acceder a las ventajas que conlleva estar registrado(sugerencias y descuentos). El administrador también puede dar de alta a usuarios directamente si lo considera oportuno.

**CASO DE USO 3.2.3.1**

Alta de usuarios

**Objetivo en contexto**

Registra un nuevo usuario en nuestra base de datos

Entradas Nombre,contraseña,email, Precondicion es

Los datos de entrada no deben existir ya en la aplicación. No debe haber dos usuarios distintos con los mismos datos(nick, email, contraseña). Salidas Muestra el mensaje por pantalla con la ID, si se ha podido añadir al

usuario, si no un mensaje indicando que no se ha podido registrar el usuario. Postcondició n si éxito

Modifica la lista de usuarios, le permite a este su acceso a los descuentos y sugerencias. Postcondició n si fallo

Si el alta se ha llevado a cabo, muestra pantalla de inicio de la aplicación. Actores Usuario/Administrador.

34

**Paso Secuencia**

**Acción normal**

1 Obtener datos de entrada: nick, contraseña, email. 2 Comprobar si los datos son válidos. 3 Comprobar que el usuario no exista. 4 Añade un usuario y el sistema genera su IDusuario

única. 5 Muestra un mensaje de usuario registrado. Secuencia alternativa 1

Paso Acción 1 Obtener datos de entrada: nombre, contraseña, email. 2 Si los datos no son válidos, lanza un mensaje de que

**los datos no son válidos. Secuencia alternativa 2**

Paso Acción 1 Obtener datos de entrada: nombre, contraseña, email. 2 Comprobar si los datos son válidos. 3 Si el usuario existe, se lanza un mensaje de usuario

repetido.

35

**3.2.3.2 Modificar usuarios**

Campos nuevos

36

Identificación

La solicitud de modificación de usuario se realiza a través de la aplicación. El administrador es el encargado de realizar de la petición. El administrador también puede modificar usuarios si lo considera oportuno.

**CASO DE USO 3.2.3.2**

Modificar usuarios

Objetivo en contexto Modificar los datos del usuarios Entradas nick y contraseña para autentificar el cambio y los datos

nuevos del usuario(nick, contraseña y/o email, el usuario deberá decidir cúal quiere cambiar). Precondiciones - El usuario debe existir.

- Los nuevos datos a cambiar no deben ser

iguales a los que tiene el usuario actualmente Salidas Cambia, en caso de que todo sea correcto, los datos

antiguos del usuario por los nuevos. Postcondición si éxito

Se muestra al usuario con los nuevos datos.

**Postcondición si fallo**

Si cambio no se ha llevado a cabo, muestra mensaje de que no se ha podido realizar. Actores Usuario y/o Administrador (en caso de que el

administrador vaya a modificar directamente los datos de un usuario el actor solo será él) .

**. Secuencia normal Paso Acción**

1 Obtener datos de entrada: nick y

contraseña para autentificar. 2 Comprobar si los datos son válidos. 3 Comprobar que el usuario exista. 4 Obtener datos de entrada: nick,

contraseña y/o email nuevos. 5 Comprobar si los datos son válidos. 6 Modifica y añade los nuevos datos del

usuario. 7 Muestra los nuevos datos del usuario. Secuencia alternativa 1

Paso Acción 1 Obtener datos de entrada: nombre,

contraseña y/o email nuevos. 2 Si los datos no son válidos, lanza un

mensaje de que los datos no son válidos. Secuencia alternativa 2

Paso Acción 1 Obtener datos de entrada: nick y

contraseña. 2 Comprobar si los datos son válidos.

37

3 Comprobar que el usuario exista. 4 Si el usuario no existe, lanza un

mensaje de usuario no registrado. Secuencia alternativa 3

Paso Acción 1 Obtener datos de entrada: nick y

contraseña. 2 Comprobar si los datos son válidos. 3 Comprobar que el usuario exista. 4 Obtener datos de entrada: nick,

contraseña y/o email nuevos. 5 Comprobar si los datos son válidos. 6 Si los datos no son válidos, lanza un

mensaje de que los datos no son válidos.

38

**3.2.3.3 Baja de usuarios**

Para dar de baja una cuenta, el usuario debe notificar a los administradores a través de la aplicación. Un vez se ha confirmada la solicitud , se procederá a la

39

eliminación en el fichero de usuarios. Esta tarea es realizada por los administradores. Una vez hecho se redirige a la pantalla de inicio.

CASO DE USO 3.2.3.3 Baja de usuarios Objetivo en contexto Dar de baja a los usuarios Entradas Nick y contraseña del usuario a eliminar Precondiciones El usuario debe existir.

Salidas Muestra un mensaje de que se ha dado de baja en caso

**de que todo sea correcto Postcondición si éxito**

Modifica la lista de usuarios y el usuario ya no puede acceder a los descuentos y sugerencias. Postcondición si fallo Si el cambio no se ha llevado a cabo, muestra un

mensaje de que no se ha podido llevar a cabo. Actores Usuario/Administrador . Secuencia normal Paso Acción

1 Obtener datos de entrada: nick y

contraseña. 2 Comprobar si el dato es válido. 3 Comprobar que el usuario exista. 4 Elimina al usuario. 5 Se muestra un mensaje de baja

**realizada. Secuencia alternativa 1**

Paso Acción 1 Obtener datos de entrada: nick y

contraseña. 2 Si los datos no son válidos, lanza un

mensaje de que el dato no es correcto. Secuencia alternativa 2

Paso Acción 1 Obtener dato de entrada: nick y

contraseña. 2 Comprobar si el dato es válido. 3 Si el usuario no existe, lanza un mensaje de usuario inexistente.

40

**3.2.3.4 Envío de sugerencias**

Disposición que tiene un usuario el cual debe estar registrado de sugerir la actualización del precio de un producto de una tienda de alimentación que considere. El usuario deberá mandar el nuevo precio del producto en cuestión a través de un buzón de sugerencias proporcionada por el administrador. El mensaje deberá contener el nombre del producto, la tienda a la que pertenece y el precio que se quiere cambiar.

41

Una vez recibido la sugerencia, avisaremos al usuario de que lo hemos recibido, se almacenará el mensaje y el administrador será el encargado de ver el contenido del mensaje, verificar que el producto exista y no haya incoherencias y hacer los cambios necesarios en la base de datos.

**CASO DE USO 3.2.3.4**

Envío de sugerencias

**Objetivo en contexto**

El usuario envía una sugerencia al administrador.

Entradas Mensaje del usuario. Precondiciones - El usuario debe estar registrado. Salidas Muestra un mensaje de que la sugerencia del usuario

se ha realizado correctamente. Postcondición si éxito

Gestión del contenido del mensaje y modifica el precio del producto si el administrador lo considera oportuno.. Postcondición si fallo

Aviso de que el envío no se ha llevado a cabo.

**Actores Administrador y usuario. Secuencia normal Paso Acción**

1 Redacción y envío del mensaje del usuario. 2 Comprobar si el mensaje ha llegado

correctamente. 3 Muestra un mensaje de envío realizado. 4 Almacenamiento y gestión del contenido del mensaje por parte del administrador, que se encargará de ver si existe el producto y de hacer el cambio si lo considera oportuno. Secuencia alternativa

Paso Acción 1 Envío del mensaje del usuario. 2 Muestra mensaje de que no se ha podido

enviar.

42

**3.2.3.7 Añadir Descuento**

La aplicación mostrará a los usuarios registrados una lista de descuentos en productos dependiendo del día de la semana (temporal). Los descuentos sólo estarán disponibles durante un tiempo limitado.

43

Ejemplo : Desde que se muestra el descuento se dispondrá de una semana para usarlo o un determinado día de la semana se muestra un descuento y sólo se puede utilizar esa semana.

**CASO DE USO 3.2.3.7**

Añadir descuento.

**Objetivo en contexto**

Se quiere añadir un nuevo tipo de descuento

Entradas idDescuento, nombre Precondiciones Que el descuento no exista.

O que si existe esté desactivado Salidas Mensaje con el descuento creado si se crea, mensaje de

error en caso contrario. Postcondición si éxito

Hay un nuevo tipo de descuento.

**Postcondición si fallo**

Mensaje de error.

**Actores Admin Secuencia normal**

Paso Acción 1 Introducir idDescuento y nombre 2 Comprobar que no existen ya 3 Crear el nuevo descuento con el id y nombre

dados. 4 Mensaje de descuento creado Secuencia alternativa

Paso Acción 1 Introducir id y nombre 2 Comprobar que no existen ya 3 Comprobar que está activo 4 Si no está activo se activa

Si está activo mensaje de error

44

**Borrar CASO DE USO**

**descuento. 3.2.3.8 Objetivo en contexto**

Se desea eliminar un tipo de descuento. La baja será lógica Entradas IdDescuento.

45

**3.2.3.8 Borrar descuento**

**Que Precondicione**

el descuento exista s Salidas Mensaje de éxito/error Postcondición si éxito

Mensaje de éxito. Se ha producido la baja lógica. Postcondición si fallo

Mensaje de error. No se ha producido ninguna baja. Actores Admin Secuencia normal

Paso Acción 1 Introducir idDescuento 2 Comprobar que exista. 3 Se produce la baja lógica. 4 Mensaje de éxito Secuencia alternativa

Paso Acción 1 Introducir idDescuento. 2 Comprobar que exista. 3 Mensaje de error.

46

**Mostrar CASO DE USO**

**descuentos. 3.2.3.9 Objetivo en contexto**

Mostrar por pantalla los descuentos disponibles.

Entradas Se selecciona “Mostrar descuentos” Precondiciones El usuario ha iniciado sesión.

Salidas Permite ver los descuentos al usuario.

47

**3.2.3.9 Mostrar descuentos**

**El Postcondición si**

usuario accede a los descuentos. éxito Postcondición si fallo

Mensaje de error.

**Actores Usuario registrado. Secuencia normal Paso Acción.**

1 Acceso a la BBDD. 2 Muestra descuentos disponibles. Secuencia alternativa

Paso Acción. 1 Acceso a la BBDD. 2 Si no puede acceder a la BBDD

aparece un mensaje de error. 3

**3.2.3.10 Modificar descuentos**

**CASO DE USO 3.2.3.10**

Modificar descuentos

**Objetivo en contexto**

Modificar los descuentos.

Entradas Datos del descuento, del producto de la tienda. Precondiciones Que exista el ID del descuento y, en caso de que se

desee modificar el ID, que el nuevo ID no exista. Salidas Descuentos modificados. Si la fecha actual supera la del

fin del descuento se elimina. Postcondición si éxito

BBDD actualizada.

**Postcondición si fallo**

No se modifica la BBDD y notifica al administrador de que la acción no pudo llevarse a cabo (Error).

48

**3.2.3.11 Dar descuento a usuario**

**CASO DE USO 3.2.3.11**

Dar descuento a usuario.

**Objetivo en contexto**

Añadir un nuevo descuento a la lista de descuentos disponibles de un usuario dado. Entradas Identificador del descuento, identificador del usuario al

que se le va a conceder el descuento. Precondiciones Tener el fichero de descuentos accesible para la

aplicación, existe el descuento, existe el usuario, el usuario no dispone ya de ese descuento.

49

**Actores Administrador. Secuencia normal Paso Acción**

1 Acceso a la BBDD. 2 Introduce datos. 3 Actualiza la BBDD Secuencia alternativa

Paso Acción 1 Acceso a la BBDD. 2 Si introduce los datos erróneos o no

puede acceder a la BBDD. 3 Muestra mensaje de error.

Salidas Modifica la lista de descuentos disponibles del usuario

añadiendo el nuevo descuento. Postcondición si éxito

Se añade un nuevo descuento a la lista de descuentos disponibles del usuario. Postcondición si fallo

Notifica al administrador de que la acción no pudo llevarse a cabo (Error). Actores Administrador. Secuencia normal

Paso Acción 1 Acceso a la BBDD. 2 Comprueba si el identificador del

descuento y de usuario existen. 3 Añade el descuento a la lista. Secuencia alternativa

Paso Acción 1 Acceso a la BBDD. 2 No existe el identificador de descuento o el de usuario / No se puede conectar con la BBDD. 3 Muestra mensaje de error.

50

**3.2.3.12 Quitar descuento a usuario**

**CASO DE USO 3.2.3.12**

Quitar descuento a usuario.

**Objetivo en contexto**

Eliminar un descuento de la lista de descuentos disponibles de un usuario dado. Entradas Identificador del descuento, identificador del usuario al

que se le va a conceder el descuento. Precondiciones Tener el fichero de descuentos accesible para la

aplicación, existe el usuario, el usuario tiene en su lista de descuentos el descuento buscado. Salidas Modifica la lista de descuentos disponibles del usuario

eliminando el descuento dado. Postcondición si éxito

Se elimina el descuento dado de la lista de descuentos disponibles. Postcondición si fallo

Notifica al administrador de que la acción no pudo llevarse a cabo (Error). Actores Administrador. Secuencia normal

Paso Acción 1 Acceso a la BBDD.

51

2 Comprueba si el identificador del

usuario existe y si este dispone del descuento dado. 3 Elimina el descuento de la lista del

**usuario Secuencia alternativa**

Paso Acción 1 Acceso a la BBDD. 2 El identificador de usuario es erróneo

/El identificador del descuento es incorrecto para ese usuario / No se puede conectar con la BBDD. 3 Muestra mensaje de error.

**3.2.3.13 Canjear Descuento**

**CASO DE USO 3.2.3.13**

Canjear un descuento activo de un usuario.

52

**Hacer Objetivo en**

factible el uso de un descuento y eliminarlo (a contexto

través del caso 3.2.3.12 Quitar descuento a usuario) para que no pueda volver a ser utilizado. El establecimiento donde se aplique el descuento registra el uso de este por su ID y el ID del usuario. Entradas Seleccionar el descuento a aplicar, pulsar en canjear. Precondiciones Descuento activo y en relación con el usuario posibilitado

a aplicarlo. Salidas ID del descuento y el ID del usuario. Postcondición si éxito

Se canjea el descuento en el producto determinado por el usuario. Se elimina y se registra su uso en tienda. Postcondición si fallo

Descuento ya utilizado o no encontrado. Notifica al usuario que el descuento no ha sido posible de canjear (Error). Actores Usuario registrado Secuencia normal

Paso Acción 1 Selección de descuento. 2 Acceso a BBDD. Comprueba si el

identificador del descuento existe en los descuentos del usuario. 3 Se aplica el descuento al producto

elegido. 4 Se hace un registro en el

establecimiento del descuento utilizado por el usuario. 5 Se elimina el descuento. Secuencia alternativa

Paso Acción 1 Selección de descuento. 2 El identificador del descuento es

incorrecto para ese usuario / No se puede conectar con la BBDD. 3 Muestra mensaje de error.

53

54

55

**3.2.4 GESTIÓN DE BÚSQUEDAS.**

La aplicación permitirá realizar búsquedas de productos que se encuentren dentro de una lista proporcionada por el sistema (se encuentra almacenada en una base de datos) y del CP introducido por el usuario.

Dichas búsquedas se podrán realizar en función de la categoría elegida entre:

- Tiendas de Alimentación: Snacks, Bebidas, Alcoholes. - Gasolineras: 95, 98, Diésel. Y del nombre del producto en cuestión.

Una vez realizada la búsqueda se mostrará al usuario el producto deseado de los diferentes establecimientos de la zona de manera ordenada por diferentes criterios (Precio y valoración). Además cuando el usuario visualiza los productos podrá valorarlos. La ausencia de este requisito supondría para el usuario tener que ir producto por producto hasta encontrar el deseado en el establecimiento deseado.

Al disponer de una lista de productos tan solo se ha de mostrar dicha lista al usuario y pedirle que elija una de las opciones de la lista mostrada.

56

**CASO DE USO 3.2.4.1**

**3.2.4.1 Búsqueda Desplegable**

Búsqueda de producto por categorías en un determinado CP

57

**El Objetivo en**

usuario navega por las secciones de las categorías contexto

(Tiendas de Alimentación o Gasolineras) hasta encontrar la deseada (Bebidas, Snacks, Alcoholes, etc.. // 95, 98, Diésel, etc..) y ahí realiza la búsqueda del nombre del producto. Entradas Categoría, CP Precondiciones Acceso del administrador a la BBDD de todos los

establecimientos. Que el código postal y el nombre del producto existan en nuestra BBDD. Salidas Lista de Productos Postcondición si éxito

Resuelve la demanda del usuario/cliente satisfactoriamente y le facilita la lista de resultados para la posterior comparación y elección de productos. Postcondición si fallo

Notifica al usuario/cliente de que la acción no pudo llevarse a cabo (Error). Actores Usuario Secuencia normal

Paso Acción 1 Selección de Categoría 2 Introduce CP 3 Acceso a la BBDD. 4 Búsqueda de producto en BBDD. 5 Devuelve el resultado Secuencia alternativa

Paso Acción 1 Selección de Categoría 2 Introduce CP 3 Si no puede acceder a la BBDD, o no existen

los datos proporcionados (CP o producto introducido) aparece un mensaje de error. 4 Muestra mensaje de error.

58

59

**Búsqueda CASO DE USO**

por NOMBRE en un determinado CP 3.2.4.2 Objetivo en contexto

Realiza una búsqueda de un producto en base al nombre que se proporcione en las listas de productos pertenecientes a diferentes establecimientos de un código postal. Entradas Listas de productos de los establecimientos, string con la

NOMBRE a buscar, un código postal. Precondiciones Acceso del administrador a la BBDD de todos los

establecimientos. Que el código postal y el nombre del producto existan en nuestra BBDD. Salidas Devuelve los establecimientos que contienen ese

nombre de producto y ese código postal. Postcondición si éxito

Resuelve la demanda del usuario/cliente satisfactoriamente y le facilita la lista de resultados para la posterior comparación y elección de productos. Postcondición si fallo

Notifica al usuario/cliente de que la acción no pudo llevarse a cabo (Error). Actores Usuario Secuencia normal

Paso Búsqueda. 1 Petición de CP 2 Petición de nombre producto 3 Acceso a la BBDD. 4 Comprobar la corrección de los datos. 5 Busca establecimientos con ese nombre y ese

CP

60

**3.2.4.2 Búsqueda por nombre**

**3.2.4.3 Ordenar resultados de las búsquedas por precio.**

**CASO DE USO 3.2.4.3**

Ordenar resultados por PRECIO

**Objetivo en contexto**

Obtener una lista de los productos , ordenada por precio de forma ascendente. Entradas El usuario selecciona el filtro orden por precio Precondiciones Acceso del Administrador a la base de Datos.

La lista de productos tiene que tener elementos. Todos los productos tienen que tener el campo precio.

61

**6 Devuelve el resultado Secuencia alternativa**

Paso Acción 1 Petición de nombre y CP 2 Acceso a la BBDD. 3 Si no puede acceder a la BBDD, o no existen

los datos proporcionados, aparece un mensaje de error. 4 Muestra mensaje de error.

Salidas Devuelve las lista de productos ordenada por precio. Postcondición si éxito

Muestra la lista de productos por pantalla para la posterior comparación y elección de productos.

**Postcondición si fallo**

Notifica al usuario que la acción no se pudo llegar a cabo.

**Actores Usuario Secuencia normal Paso Acción**

1 Seleccionar el filtro ordenar por precio. 2 Acceso a la BBDD. 3 Ordena los productos en orden ascendente 4 Mostrar la lista por pantalla. Secuencia alternativa

Paso Acción 1 Seleccionar el filtro ordenar por precio. 2 No acceso a la BD, los productos no tienen

el campo precio, generar un mensaje. 3 Mostrar mensaje de error. 4 Volver a la pantalla selección.

62

**3.2.4.4 Ordenar resultados de las búsquedas por valoración.**

**CASO DE USO 3.2.4.4**

Ordenar resultados por VALORACIÓN

**Objetivo en contexto**

Obtener una lista de los productos , ordenada por la valoración de forma descendente. Entradas El usuario selecciona el filtro orden por precio Precondiciones Correcto acceso del Administrador a la base de Datos.

La lista de productos tiene que tener elementos. Todos los productos tienen que tener el campo valor. Salidas Devuelve las lista de productos ordenada por la valoración Postcondición si éxito

Muestra la lista de productos por pantalla para la posterior comparación y elección de productos. Postcondición si fallo

Notifica al usuario que la acción no se pudo llegar a cabo.

**Actores Usuario Secuencia normal**

Paso Acción 1 Seleccionar ordenación por valoración. 2 Acceso a la BBDD 3 Ordena los productos en orden descendente,

lo más valorados primero 4 Mostrar la lista por pantalla. Secuencia alternativa

**Pasos Acción**

1 Seleccionar ordenación por valoración. 2 No acceso a la BBDD, los productos no

contengan el campo valor, generar mensaje 3 Mostrar mensaje de error. 4 Volver a la pantalla de selección.

63

**3.2.4.5 Mostrar Producto**

**CASO DE USO 3.2.4.5**

Mostrar un producto con sus características.

**Objetivo en contexto**

Representar la información de un producto y del establecimiento al que pertenece.

Entradas El usuario realiza una búsqueda del producto deseado. Precondiciones Producto y establecimiento encontrado al realizar la

búsqueda. Salidas 1.-Muestrario de los productos deseados en lista ordenada

por precio/valoración con los siguientes atributos visuales al usuario: Dirección del establecimiento, Precio, Descuento (si existe) y Valoración media (si existe). 2.-Click seleccionando el producto en cuestión y muestrario de todos los detalles (ID, Nombre, Precio, Descuento, Valoración, Detalles Establecimiento), además de la

64

posibilidad de valorar el producto, valorando así el establecimiento. Postcondición si éxito

Información del producto y establecimiento en detalle.

**Postcondición si fallo**

Mensaje de error: Producto no encontrado.

**Actores Usuario Secuencia normal**

Paso Acción 1 Postcondición de búsqueda previa realizada con éxito 2 Acceso a BBDD y selección de los

productos 3 Salida por pantalla de lista de productos 4 Visualización en pantalla de los atributos

**de cada producto Secuencia alternativa**

Paso Acción 1 Postcondición de búsqueda previa no

realizada o producto no encontrado 2 Mensaje de error

**3.2.4.6 Valorar Producto**

**CASO DE USO 3.2.4.6**

Valoración de producto

**Objetivo en contexto**

Tener la posibilidad de valorar un producto concreto para la posterior recopilación de valoraciones de los usuarios. Entradas Valoración interactiva del usuario. Precondicione s

Realizar selección de mostrar producto y elegir valorar.

Salidas Producto valorado y su puntuación. Postcondición si éxito

Devuelve la valoración personal del producto junto con la valoración media pública del producto. Postcondición si fallo

Mensaje de error: No se ha podido valorar el producto.

**Actores Usuario Secuencia normal**

Paso Acción 1 Solicitar valoración al usuario del producto

entre el 1-5 2 Entrada por click (ratón) seleccionando el

valor en cuestión 1-5 3 Acceso a BBDD y guardado correcto de los

datos en BBDD

65

4 Mensaje de valoración completada con

éxito y agradecimiento al usuario: “¡Gracias por valorar este producto!” Secuencia alternativa

Paso Acción 1 Solicitar valoración al usuario del producto

entre el 1-5 2 Entrada por click (ratón) seleccionando el

valor en cuestión 1-5 3 En caso de entrada fallida: mensaje de

error 4 Vuelta al muestrario del producto

66

67

**Sección 3.3 Requisitos de rendimiento**

En un principio esta aplicación será ejecutada en un entorno cerrado (desde un ordenador del laboratorio de la facultad de informática) y que sólo un usuario pueda usar la aplicación de forma concurrente.

Los ordenadores de los laboratorios de la facultad son capaces de procesar sobradamente todos los procesos software que se ejecutan en el programa, ya que no son funciones de alto coste de rendimiento.

**Sección 3.4 Requisitos lógicos de la base de datos**

-La información que usará la aplicación estará contenida en 8 ficheros distintos:

1. Productos: contendrá todos los productos que aparecen en la aplicación. Sus atributos son idProducto y Nombre,además de tipo de gasolinera o tipo de alimento si especializamos. Para distinguir entre gasolinas y alimentos el idProducto empezará por pg y pa respectivamente.

2. Establecimientos: contendrá todos los establecimientos que aparecen en la aplicación. Sus atributos son IdEstablecimiento, Nombre, Dir y C.P. Para distinguir entre gasolineras y tiendas el IdEstablecimiento empezará por eg y ea.

3. Usuarios: contendrá información de todos los usuarios registrados. Sus

atributos son IdUsuario, nick, e-mail y Contraseña.

4. Descuentos: contendrá la información de todos los descuentos que ha manejado y maneja la aplicación. Son altas y bajas lógicas. Sus atributos son Descripción y IdDescuento.

5. ProductoTienda: contendrá todos los productos que vende actualmente cada tienda con su precio (sin aplicar descuento) y descuento a aplicar. Sus atributos serán IdProducto, IdEstablecimiento, Precio y Descuento.

6. Valoración: contendrá una lista con IdProducto, IdEstablecimiento e idUsuario

junto a la valoración que han realizado.

7. Búsquedas: contendrá una lista de búsquedas realizadas por cada usuario

sobre un producto. Sus atributos son IdUsuario, CP e IdProducto.

68

8. UsuarioDescuentos: contiene una lista de todos los descuentos que tiene cada

usuario. Sus atributos son IdUsuario, IdDescuento.

-Los archivos deberán poder ser localizados por la aplicación. Al arrancar la aplicación se cargarán los datos de todos los ficheros en ella.

-En caso de que la aplicación no sea capaz de cargar todos los ficheros se abortará su ejecución.

**ESQUEMA APLICACIÓN EN BASE DE DATOS RELACIONAL**

69

**Sección 3.5 Restricciones de diseño**

-Base de datos realizada en txt. -Desarrollo de proyecto en lenguaje Java. -Utilización de Eclipse como IDE. -Funcionamiento en Windows superiores a Windows Vista. -Utilización de Microsoft Project para realizar la planificación temporal y revisar retrasos. -Aplicación de escritorio -Todos los establecimientos llevan un control del canjeo de los descuentos personales y acepta su uso.

**Sección 3.6 Atributos del sistema software**

● Es un sistema fiable para que los usuarios puedan disfrutar de una experiencia que se ajusta a la realidad, se garantiza que los productos y establecimientos que visualizan son correctos.

● Los tiempos de respuesta de las consultas de los usuarios no deberán sobrepasar un tiempo determinado, de lo contrario, la experiencia de usuario no será buena y los usuarios no elegirían nuestra herramienta. Por ello, este sistema realizará todas sus tareas de forma rápida; devolviendo buenos indicadores de rendimiento.

● Se garantiza que todos los datos relacionados con nuestros usuarios están protegidos en nuestras bases de datos.

Artículo IV. Apéndices

70