Memoria escrita

Descripción general de la solución (análisis, diseño, e implementación).

Para la solución del proyecto propuesto, primero tomamos en cuenta las sugerencias que nos hizo el profesor con respecto a las funcionalidades y guiándonos con la rúbrica enviada, empezamos creando las clases de la capa lógica que íbamos a necesitar a lo largo de la implementación. Durante este proceso nos dimos cuenta de que había clases que no nos iban a servir en cuestión de la implementación general del proyecto mientras que por otro lado también empezamos a ver la necesidad de darle vida a unas clases nuevas (notificación, calificación).

Al saber que ya poseíamos una base lógica de clases con sus respectivos atributos y métodos necesarios para ser tratadas luego por nuestra capa de persistencia y de interfaz, comenzamos a pensar que objetos necesitaríamos que fuesen guardados para el perfecto funcionamiento del proyecto y para poder demostrar que la totalidad de las funcionalidades que teníamos podrían ser ejecutadas con normalidad.

Así empezamos a trabajar en las 2 capas faltantes de manera simultánea, teniendo en cuenta cuales eran las necesidades de la una con respecto a la otra, así dimos “vida” a nuestro proyecto, usando para la base de datos de objetos, archivos de tipo Json y accediéndolos gracias a la librería Gson, permite la obtención, el manejo y la escritura de objetos de java a Json. Se logro que cada objeto que necesitaba ser guardado o accedido fuera posible.

Mientas que en la capa de interfaz de usuario iniciamos mirando lo que requería la rúbrica del proyecto y así creamos las respectivas clases y dimos vida a las clases que iban a jugar como nuestras funcionalidades que poseerían el método ejecutar que cada una necesitaba implementar. De la misma manera empezamos a crear los diferentes menús que iban a ser lanzados dependiendo de qué tipo de usuario estuviera usando la aplicación.

Al terminar con ambas capas empezamos a notar que nos faltaba o que no estaba ejecutando de la manera correcta y corregíamos todo lo que se nos iba presentando a lo largo del camino para así terminar el proyecto y volver a mirar la rúbrica guía para identificar si nos hacía falta algún tipo de punto.

Descripción del diseño estático del sistema en la especificación UML (Diagrama de clases y objetos del sistema).

 Diagrama de Clases

A close up of text on a white background

Description automatically generated

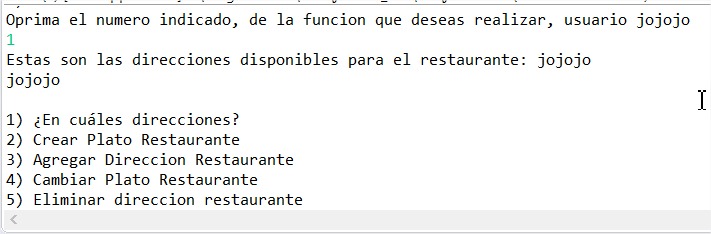
 Diagrama de Objetos

A close up of text on a white background

Description automatically generated

Capturas de pantalla de los resultados de las funcionalidades implementadas.

Restaurante: En cuales direcciones



Restaurante: crear plato

A screenshot of a cell phone

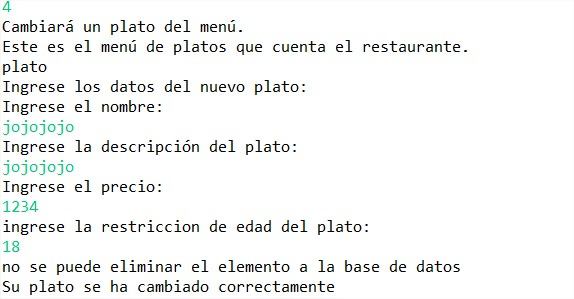
Description automatically generated

Restaurante: agregar dirección

A screenshot of a cell phone

Description automatically generated

 Restaurante: Cambiar plato



 Restaurante: eliminar dirección

A screenshot of a cell phone

Description automatically generated

Cliente: hacer pedido

A screenshot of a cell phone

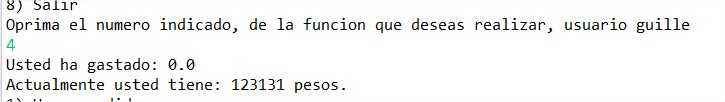
Description automatically generated

Cliente: calificar restaurante

A screenshot of a cell phone

Description automatically generated

Cliente: cuanto he gastado



Cliente: agregar saldoA screenshot of a cell phone

Description automatically generated

Administrador: crear Restaurante

A screenshot of a cell phone

Description automatically generated

Administrador: crear Tendero

A screenshot of a cell phone

Description automatically generated

Administrador: Eliminar funcionalidades

A screenshot of text

Description automatically generated

Administrador: Agregar funcionalidad

A screenshot of text

Description automatically generated

Manual de usuario para el correcto funcionamiento del sistema.

Se dará inicio a la aplicación proporcionando a el usuario las posibilidades de registrarse como cliente, hacer un log in a la aplicación como usuario ya registrado o ser un usuario invitado,

Al registrarse usted será cuestionado en modo de formulario acerca de la información necesaria para ser un nuevo cliente de la plataforma.

Al hacer log in usted debe de digitar su nombre de usuario seguido por la clave que uso al crear su usuario, los usuarios que se pueden logear puede ser de tipo:

Admin. Dando como ejemplo ya implementado un administrador con userName “Admin” y clave (contraseña) “Admin”, luego hacer su login este administrador se le mostrarán todas las funcionalidades del sistema en ese momento y luego se le mostrará su propio menú con aquellas funcionalidades que puede ejecutar:

* Crear Restaurante
  + La cual le da la opción a el administrador de crear un nuevo restaurante
* Crear Tendero
  + Permite crear un tendero a el administrador
* El Tendero que más reparte
  + Escribiendo el nombre de el restaurante del cual se desea averiguar el tendero que mas ha entregado pedidos, muestra el resultado esperado
* Quitar funcionalidades de alguien
  + Permite eliminar opciones del menú de algún tipo de usuario, ya sea cliente, tendero o restaurante
* Agregar funcionalidades de alguien
  + Permite agregar opciones del menú de algún tipo de usuario, ya sea cliente, tendero o restaurante
* Salir

Restaurante. Dando como ejemplo ya implementado un restaurante con nombre (userName) “MC” y clave “MC”, luego de ingresar como un restaurante, a este se le mostrarán sus opciones de menú:

* ¿En cuales direcciones?
  + Me muestra cuales son las direcciones en las que el restaurante tiene sede
* Crear plato Restaurante
  + Crea el plato que va a vender el restaurante mediante la plataforma
* Agregar Dirección Restaurante
  + Agrega una nueva dirección a las direcciones de restaurante
* Cambiar Plato Restaurante
  + Cambia el plato que vende el restaurante por aquel nuevo plato que desee vender ahora
* Eliminar dirección restaurante
  + Elimina la dirección que desee de las direcciones que tiene el restaurante
* Salir

Tendero. Se creó un Tendero como ejemplo el cual tiene username “Guille” y una clave “Guille”, luego al entrar a la plataforma como un Tendero registrado, se mostrarán las funcionalidades que este puede realizar:

* Cuantos Pedidos He Entregado
  + Me da el numero de pedidos que he entregado a lo largo de mi vida laboral con la plataforma
* Aceptar Pedido
  + Me permite aceptar un pedido que aun no ha sido aceptado
* Salir

Cliente. Dando como ejemplo ya implementado un cliente con username “Andre” y con clave “Andre”, luego de iniciar sesión como un Cliente se procederá a mostrar las diferentes funcionalidades que puede realizar este Cliente:

* Hacer Pedido
  + Permite a el usuario realizar un pedido a cierto restaurante, el cliente debe de saber el nombre del restaurante a la hora de realizar el encargo, el cliente solo puede realizar un pedido por restaurante
* Calificar Tendero
  + Permite calificar el tendero que se encargó de mi pedido
* Calificar Restaurante
  + Permite calificar el restaurante que realizo el plato que adquirí
* Cuanto He Gastado
  + Me muestra por pantalla cuanto he gastado en mis pedidos
* Mejor Restaurante
  + Me muestra cual es el restaurante mejor calificado entre todos
* Platos Que Mas Compre
* Agregar Saldo
  + Permite a el usuario agregar saldo a su cuenta para poder comprar platos
* Salir