



Tarea Integradora 3: Afore

Realizada por

Daniela Bonilla Cáceres – A00372534

Tomas Ossa Cárdenas – A00372231

Docente

Juan Manuel Reyes

Materia

Algoritmos y programación 2

Martes 22 de mayo de 2021

Universidad Icesi

Cali-Colombia

Tabla de Contenido

| | |
|-------------------------------------|----|
| Integrantes | 3 |
| Enunciado del proyecto Afore..... | 4 |
| Requerimientos funcionales | 5 |
| Requerimientos no funcionales | 8 |
| Justificación | 9 |
| Mockups | 10 |
| Diseño de Escenarios..... | 29 |
| Diseño de Pruebas..... | 30 |
| Diagrama de clases..... | 33 |

Integrantes

Daniela Bonilla Cáceres

- Código: A00372534
- Enlace Github: <https://github.com/DaniBonica001>

Tomás Ossa Cárdenas

- Código: A00372231
- Enlace Github: <https://github.com/tomasossacardenas>

Enunciado del proyecto Afore

La llegada de la pandemia a Colombia obligó a toda la población colombiana a encerrarse en sus casas y mantener una cuarentena obligatoria ordenada por el gobierno nacional con la finalidad de evitar la propagación del virus. Los colombianos tuvieron que hacer un cambio en sus estilos de vida, pues salir ya no era seguro. Las actividades de interacción social como las salidas a cenar con amigos un fin de semana, las reuniones familiares, la vida nocturna (bares y discotecas), las salidas a cine e incluso a visitar un centro comercial, fueron catalogadas como peligrosas debido al alto riesgo de propagación del virus. Sin embargo, unos meses después del inicio de la pandemia, las actividades que eran comunes se empezaron a reanudar poco a poco. Los restaurantes permitieron el ingreso de cierto aforo de clientes, los centros comerciales optaron por una misma salida y entrada para controlar el aforo de visitantes e incluso los cines se reinventaron con los autocines.

Con la reanudación de las actividades sociales, se identifica una problemática. A pesar de que para todos los ciudadanos es un gran alivio el hecho de poder salir a cenar, visitar a sus amigos, reunirse con su familia, comer un helado en un centro comercial, se hizo muy común escuchar la palabra “Aforo”, palabra base de la nueva realidad. Todos los lugares públicos deben procurar que sus clientes mantengan una distancia mínima de 2 metros, es decir, que en un lugar donde antes podían estar 100 personas, ahora solo pueden estar 50, es decir, su aforo máximo permitido es de 50 personas. Esto significa que ahora todos pueden salir, pero si el lugar a donde se dirigen tiene su aforo máximo, lo más probable es que se tenga que optar por otro lugar o volver a casa.

Esta problemática trae consigo que las personas decidan muchas veces no salir o en muchas ocasiones perder su tiempo ya que al llegar a un lugar no se les permite el ingreso porque el recinto ya llegó a su aforo máximo, es por esto que se presenta Afore como un software respecto al control del aforo de las empresas de servicios. El software presentado para el proyecto de semestre será especializado para los restaurantes permitiéndoles controlar el aforo del lugar y otras funcionalidades como llevar el registro de inventario, de ventas, de gestión de productos ingredientes y la capacidad de generar reportes en Excel para la filtración de datos según la necesidad. Sin embargo, este es solo el primer paso para el objetivo final del software ya que se desarrollará una aplicación móvil donde los ciudadanos y posibles clientes puedan conocer el aforo de los restaurantes, y hacer sus reservas para que así no pierdan el viaje y se sientan seguros de que al llegar al restaurante tendrán su espacio guardado respetando el aforo.

Requerimientos funcionales

R1: Crear un Producto. Permite crear un producto con un número de identificación, un nombre, una categoría, su descripción, tamaño(opcional) y su respectivo precio, y su disponibilidad. Este producto va a ser añadido a la lista de productos del restaurante.

R2: Crear un usuario. Permite crear un usuario con nombre, apellidos, identificación, teléfono, usuario(identificación), contraseña y el tipo de empleado que será.

R3: Crear un Chef. Permite crear un chef con su nombre, apellidos, identificación, teléfono, usuario(identificación), contraseña, y los mejores platos de ese Chef.

R4: Crear un Domiciliario. Permite crear un domiciliario con su nombre, apellidos, identificación, teléfono, usuario(identificación), contraseña, y la cantidad de pedidos que puede hacer en una (1) hora.

R5: Crear un Cajero. Permite crear un cajero con su nombre, apellidos, identificación, teléfono, usuario(identificación), contraseña, y la respuesta a la opción de trabajar como mesero también.

R6: Crear un Mesero. Permite crear un mesero con su nombre, apellidos, identificación, teléfono, usuario(identificación), contraseña, y la cantidad de mesas que puede atender.

R7: Crear un Cliente. Permite crear un cliente con su nombre, su identificación, su dirección(opcional), su teléfono y unas observaciones. Un cliente será guardado en el programa.

R8: Actualizar un Cliente. Permite actualizar un cliente con su nombre, su identificación, su dirección (opcional), su teléfono y unas observaciones. Un cliente será actualizado en el programa.

R9: Crear un Pedido. Permite crear un pedido con un código autogenerado, con su estado predeterminado (ACTIVO), con los productos (mínimo uno) y su cantidad (mínimo uno), con el nombre del cliente que lo solicita y el nombre del empleado que se encarga de él, con la fecha y hora del pedido, además unas observaciones del pedido (si las hay). Este pedido es añadido a la lista de pedidos del empleado.

R10: Eliminar producto. Elimina un producto a partir de su nombre. Este producto puede ser eliminado siempre y cuando no haya un pedido sin entregar que lo contenga.

R11: Eliminar un cliente. Eliminar un cliente a partir de su id. Este cliente podrá ser eliminado solo si no hay ningún pedido a su nombre.

R12: Eliminar un usuario. Eliminar un usuario a partir de su id. El usuario puede eliminar solo su usuario dentro de la pantalla y el administrador si puede eliminar cualquier usuario

del sistema. Este usuario se puede eliminar solo si no tiene ningún pedido que no haya sido enviado.

R13: Eliminar un pedido. Eliminar un pedido a partir de su código.

R14: Deshabilitar un Producto. Permite deshabilitar un producto a partir de su nombre, lo que realiza este requerimiento es cambiar el estado del producto de ACTIVO a INACTIVO.

R15: Deshabilitar un Usuario. Permite Deshabilitar un usuario a partir de su id, lo que realiza este requerimiento es cambiar el estado del usuario de ACTIVO a INACTIVO. Si un usuario está deshabilitado no podrá iniciar sesión con su cuenta.

R16: Deshabilitar un Cliente. Permite Deshabilitar un cliente a partir de su identificación, lo que realiza este requerimiento es cambiar el estado del cliente de ACTIVO a INACTIVO.

Si un cliente está inactivo no se podrán realizar pedidos a su nombre.

R17: Actualizar un Producto. Permite actualizar un producto con sus atributos nuevos que son su nombre, un tipo, sus ingredientes (mínimo uno), tamaño y respectivo precio (que varía dependiendo del tamaño)

R18: Actualizar un Usuario. Permite actualizar los atributos nombre, apellidos, id, teléfono, username y password de un usuario existente en la lista de usuarios del restaurante.

R19: Actualizar un Cliente. Permite actualizar los atributos nombre, apellidos, id, dirección, teléfono, y las observaciones de un cliente existente en la lista de Clientes del restaurante.

R20: Actualizar un Pedido. Permite actualizar los atributos código, estado (ACTIVO, INACTIVO), los productos (mínimo uno) y su cantidad (mínimo uno), con el nombre del cliente que lo solicita y el nombre del empleado que se encarga de él, con la fecha y hora del pedido, además unas observaciones del pedido (si las hay) de un pedido.

R21: Tomar hora y fecha al registrar un pedido. Permite registrar el pedido con la fecha y hora actual en la que se realiza el pedido.

R21: Controlar horario de trabajadores. Permite realizar ingreso y posterior salida de la jornada laboral de un empleado, esta hora de entrada y de salida se guardará en el programa para poder llevar un registro de puntualidad, horas de trabajo y horas extra realizadas por el empleado.

R22: Enviar factura de pedido al correo electrónico. Si el pedido es un pedido a recoger o a domicilio, el cliente tendrá la opción de recibir la factura por correo electrónico. Esta factura se enviará desde el Gmail del restaurante. La opción para enviar factura debe estar presente al momento de programar un pedido y al momento de actualizar un pedido.

R23: Enviar Reserva por correo electrónico. Al momento de realizar una reserva la confirmación de la reserva será enviada al correo electrónico del cliente, en este correo además se anexará un menú en formato pdf.

R24: Generar menú en formato pdf. Se generará un menú en un formato pdf con los productos y respectivos precios del restaurante.

R25: Permitir al cliente calificar servicio de atención. Permite al cliente al momento de salir dar la opción de calificar la atención por parte del personal del restaurante en un puntaje de 1-5 con un campo de observaciones para quejas o sugerencias.

R26: Permitir al cliente calificar servicio de comida. Permite al cliente al momento de salir dar la opción de calificar la calidad de la comida del restaurante en un puntaje de 1-5 con un campo de observaciones para quejas o sugerencias.

R27: Generar Facturación Separada. En caso de que el pedido vaya a ser pagado individualmente o vayan a dividir el pedido, se generará una factura separada, donde indicarán el total a pagar de cada persona con los productos que consumió y su nombre para así mantener un orden.

Requerimientos no funcionales

- RN1:** El sistema debe ser desarrollado en el IDE Eclipse.
- RN2:** Toda funcionalidad del sistema debe responder al usuario en menos de 10 segundos.
- RN3:** Los datos modificados en los archivos de persistencia de la información, deben ser actualizados para todos los usuarios en menos de 5 segundos.
- RN4:** El sistema debe permitir navegar entre ventanas mientras se realiza otro proceso de forma concurrente.
- RN5:** El sistema debe proporcionar mensajes de error que sean informativos y orientados al usuario.
- RN6:** El sistema debe poseer interfaces gráficas amables con el usuario.
- RN7:** El sistema debe asegurar que los datos estén protegidos del acceso no autorizado.

Justificación

El proyecto de Afore se comenzó a trabajar en la clase de introducción a las TIC del primer semestre, proyecto en el cual participaban las dos personas encargadas de esta nueva fase, Daniela Bonilla y Tomás Ossa. La continuación de este proyecto permite poder sacar adelante en equipo lo empezado desde el primer semestre, aplicando los conocimientos adquiridos en las materias de Algoritmos y Programación I y II.

La experiencia del usuario es entendida como las percepciones y respuestas de las personas, resultantes del uso o anticipación de uso de un producto, sistema o servicio, donde se tienen en cuenta las emociones y respuestas físicas y psíquicas del usuario al utilizar el producto o servicio, es por esto que se ha decidido realizar una especialización por parte de los autores de este proyecto en experiencia del usuario que permitirá a los trabajadores del restaurante conllevar una jornada laboral completamente a gusto y satisfechos con el programa, sus funcionalidades, su usabilidad e interacción personal, aspecto que tendrá efectos positivos en cuanto al rendimiento laboral y resultados del restaurante.

Por otro lado, el proyecto cuenta con funcionalidades avanzadas, que permiten la medición de resultados para la toma de decisiones, funcionalidades donde se hace indispensable el conjunto de ideas de mínimo dos personas para minimizar los errores y tener un pensamiento global y no sesgado como suele pasar en proyectos individuales.

En conclusión, Afore es un proyecto muy bien estructurado que prioriza la experiencia del usuario y en conjunto con funcionalidades avanzadas son utilizadas como método a la excelencia en atención y servicio que se reflejan finalmente en mejores resultados. Con base en lo anteriormente dicho se hace indispensable la unión de dos mentes con pensamientos y puntos de vista diferentes para tener una visión completa permitiendo la realización del mejor programa posible para el cliente.

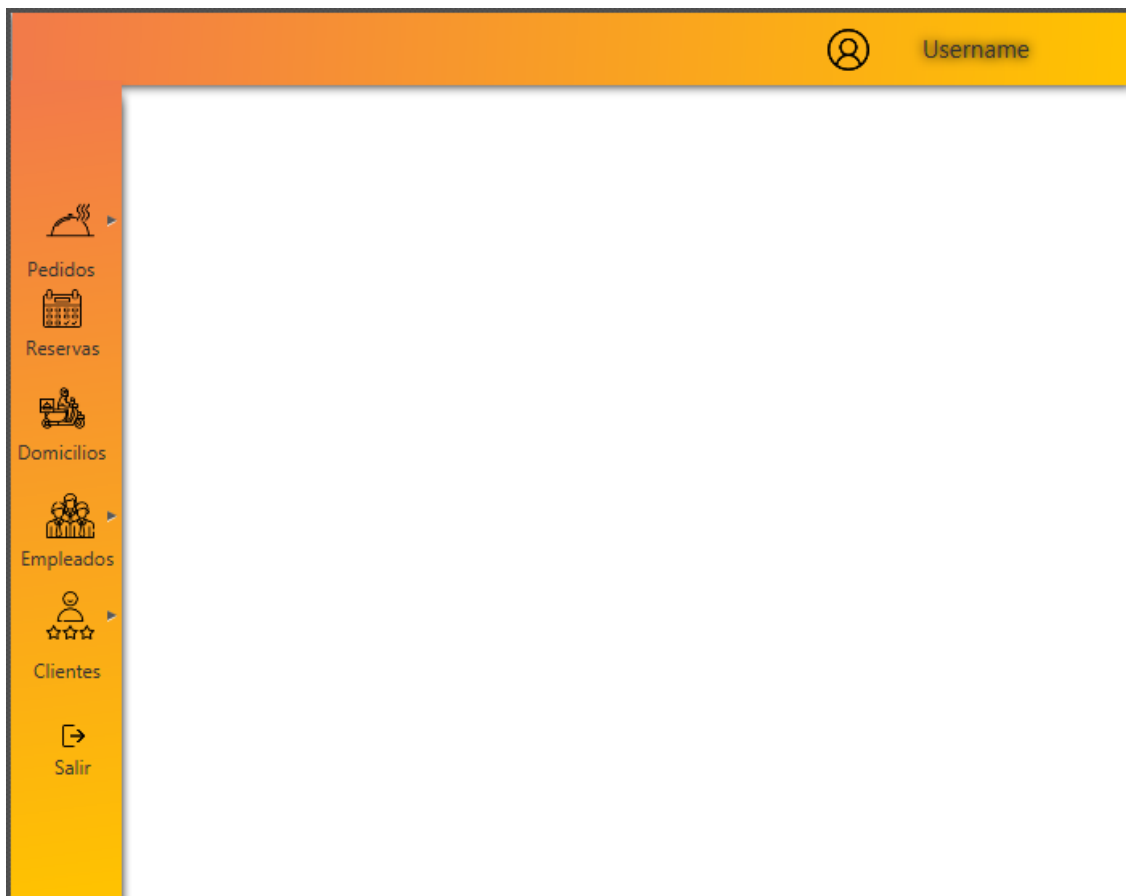
Mockups

Pantalla de Login: Esta pantalla permitirá ingresar al programa por parte del trabajador con su nombre de usuario y contraseña, en caso de que no esté registrado está la posibilidad de registrarse, vínculo que lo llevará a la pantalla de Sign Up.



The mockup shows a login interface with a white rounded rectangle centered on an orange background. On the left is a circular logo with a yellow-to-orange gradient, containing a fork, the text 'IAFORE!', a person icon, a knife, and the word 'Restaurantes' below. To the right, the title 'Ingresar' is in bold black. Below it are two light gray input fields: the first with an envelope icon and the label 'Email', and the second with a key icon and the label 'Password'. An orange 'Ingresar' button is below the fields. At the bottom, the text '¿Aun no tienes cuenta?' is followed by a yellow 'Registrarse' link.

Menú: El menú aparecerá en todas las operaciones o demás pantallas en las que esté el usuario que le permitirá movilizarse libremente por la aplicación dependiendo de la necesidad, en el menú encontrara opciones sobre gestionar pedidos, reservas, domicilios, empleados, clientes y la opción de salir de su sesión.



Crear Producto: Pantalla que permite la creación de un producto, la categoría del producto indica el tipo de producto, por ejemplo puede pertenecer a la categoría de Lasagnas o Pastas o Sopas o bebidas.

Crear nuevo producto

Identificación:

Nombre:

Categoría:

Tamaño:

Precio:

Disponibilidad:

Descripción:

Registrar **Cancelar**

Pantalla de pedido: Pantalla que permite gestionar un pedido, añadiendo productos y su cantidad al lado izquierdo con base en una lista de productos, al lado derecho irá apareciendo la factura cada que se añadan nuevos productos al pedido, en la parte superior se indicará la mesa para la cual es el pedido.

Table
01

Table
02

Table
03

Table
04

Table
05

Agregar mesa

Listado de productos

Cantidad

Número

Agregar

Aún no tienes productos?

Crear producto

Factura

#mesa

Nombre empleado encargado de hacer la factura

| Nombre | Cantidad | Precio |
|---------------------|----------|--------|
| Tabla sin contenido | | |

Total

...

Imprimir

Borrar

Cancelar

Pantalla de actualizar Producto: Permite buscar un cliente a partir de su numero de identificación y poder actualizar sus datos, la categoría es la labor que desempeña en el restaurante (domiciliario, cajero, chef, mesero, etc...).

Actualizar producto

Identificación: **Buscar Cliente**

Nombre:

Categoría:

Tamaño:

Precio:

Disponibilidad:

Actualizar **Cancelar**

Pantalla deshabilitar producto: Permite deshabilitar o habilitar un producto a partir de una identificación, los otros campos no son editables, son solo indicativos para proveer mas información sobre el producto a deshabilitar.

Deshabilitar Producto

Datos Producto

Identificación: **Buscar**

Nombre:

Categoría:

Tamaño:

Precio:

Disponibilidad:

Estado: ☐ Habilitado ☐ Deshabilitado

Deshabilitar **Habilitar**

Datos Aforo



Aforo Máximo
X personas

Aforo Actual
X personas

Cupos Disponibles
X personas

Pantalla de eliminar Producto: Pantalla que permite eliminar un producto a partir de su identificación , los demás campos no son editables solo se muestran como información extra para el que eliminará el producto.

Eliminar producto

Datos producto

Identificación:

Nombre:

Categoría:

Tamaño:

Precio:

Disponibilidad:

Buscar

Eliminar

Cancelar

Datos Aforo



Aforo Máximo
X personas

Aforo Actual
X personas

Cupos Disponibles
X personas

Pantalla de Reserva: Permite hacer una reserva de un cliente existente, en caso de que no exista podrá crearlo rápidamente con pocos datos, la reserva cuenta con la cantidad de personas y una encuesta de salubridad por covid.

Reserva

Cédula:

Buscar Cliente

¿Aún no está registrado el cliente?

Registrar

Nombre:

Cantidad Personas

Telefono:

Hora:

+

-

1. ¿Alguno de los acompañantes ha tenido fiebre o malestar general la última semana?

☐ Si

☐ No

2. ¿Alguno de los acompañantes ha estado en contacto con alguien positivo para covid-19?

☐ Si

☐ No

Reservar

Cancelar

Datos Aforo



Aforo Máximo

X personas

Aforo Actual

X personas

Cupos Disponibles

X personas

Pantalla Domicilio: Permite gestionar domicilios igualmente como un pedido. El envío de la factura será posible hacerlo por un correo electrónico.

Domicilios

☐ Recoger en el restaurante

☐ Programar envío

Productos

Cantidad

Número

Agregar

Hora Envío

Número

A nombre de:

| Nombre | Cantidad | Precio |
|---------------------|----------|--------|
| Tabla sin contenido | | |

Total

...

Enviar

Borrar

Cancelar

Pantalla Crear Empleado: Permite crear un nuevo empleado.



Crear nuevo empleado

Nombre:

Apellido:

Identificación:

Teléfono:

Usuario:

Contraseña:

Tipo de empleado:

Crear Cancelar

The image shows a web form titled "Crear nuevo empleado" (Create new employee) centered on a background with a vertical orange-to-yellow gradient. The form itself is a white rounded rectangle. It contains seven input fields: "Nombre" (Name), "Apellido" (Surname), "Identificación" (Identification), "Teléfono" (Phone), "Usuario" (User), "Contraseña" (Password), and "Tipo de empleado" (Employee type). The first six are text inputs, and the last is a dropdown menu. At the bottom of the form are two orange buttons: "Crear" (Create) and "Cancelar" (Cancel).

Pantalla actualizar Empleado: Permite actualizar los datos de un empleado existente.

Actualizar empleado

Identificación:

Nombre:

Apellido:

Teléfono:

Usuario:

Contraseña:

Tipo de empleado:

Pantalla de eliminar Empleado: Permite eliminar un empleado existente, los campos diferentes al numero de identificación no son editables, únicamente son para proveer información a quien lo elimina.

Eliminar Empleado

Datos Empleado

Identificación: **Buscar**

Nombre:

Identificación:

Teléfono:

Usuario:

Tipo de empleado:

Eliminar **Cancelar**

Datos Aforo



Aforo Máximo
X personas

Aforo Actual
X personas

Cupos Disponibles
X personas

Pantalla Deshabilitar Empleado: Permite habilitar o deshabilitar un empleado existente, la búsqueda del empleado se hará por medio de su identificación.

Deshabilitar Empleado

Datos Empleado

Identificación:

Nombre:

Identificación:

Teléfono:

Usuario:

Tipo de empleado:

Estado: ☐ Habilitado ☐ Deshabilitado

Datos Aforo



Aforo Máximo
X personas

Aforo Actual
X personas

Cupos Disponibles
X personas

Pantalla Registrar Cliente: Permite la creación de un cliente con sus datos.



El formulario 'Registrar cliente' se encuentra en un recuadro blanco con esquinas redondeadas, centrado sobre un fondo dividido horizontalmente en naranja (arriba) y amarillo (abajo). El título 'Registrar cliente' está en negrita y color negro. Las etiquetas de los campos ('Nombre:', 'Identificación:', 'Dirección:', 'Teléfono:', 'Observaciones:') están en un color grisáceo. Los botones 'Registrar', 'Reservar' y 'Cancelar' son de color naranja con texto blanco.

Registrar cliente

Nombre:

Identificación:

Dirección:

Teléfono:

Observaciones:

Registrar **Reservar** **Cancelar**

Pantalla actualizar Cliente: Permite actualizar los datos de un cliente existente.



The image shows a web form titled "Actualizar cliente" (Update client) centered on a white background with rounded corners. The form is set against a background that is orange on the left and yellow on the right. The form contains the following elements:

- Identificación:** A text input field followed by an orange button labeled "Buscar Cliente".
- Nombre:** A text input field.
- Dirección:** A text input field.
- Teléfono:** A text input field.
- Observaciones:** A larger text area for notes.
- Buttons:** At the bottom of the form are two orange buttons: "Actualizar" (Update) and "Cancelar" (Cancel).

Pantalla eliminar Cliente: Permite la eliminación de un cliente a partir de su identificación.

Eliminar cliente

Datos Cliente

Identificación:

Nombre:

Identificación:

Dirección:

Teléfono:

Eliminar

Cancelar



AFORE
Restaurantes

Aforo Máximo
X personas

Aforo Actual
X personas

Cupos Disponibles
X personas

Pantalla deshabilitar Cliente: Permite deshabilitar o habilitar un cliente a partir de su identificación

Deshabilitar Cliente

Datos Cliente

Identificación:

Nombre:

Identificación:

Dirección:

Teléfono:

Estado: ☐ Habilitado ☐ Deshabilitado

Deshabilitar

Habilitar

Buscar

Datos Aforo



Aforo Máximo
X personas

Aforo Actual
X personas

Cupos Disponibles
X personas

Pantalla de Gestión de Calidad: Permite al cliente rellenar una encuesta donde califica la atención y la calidad de la comida en un rango de 1-5.

Gestión de Calidad

Atención

¿Cómo califica la atención del personal? *5 calificación más alta

☐ 1

☐ 2

☐ 3

☐ 4

☐ 5

¿Alguna observación o sugerencia?

Registrar

Cancelar

Comida

¿Cómo califica la calidad de la comida? *5 calificación más alta

☐ 1

☐ 2

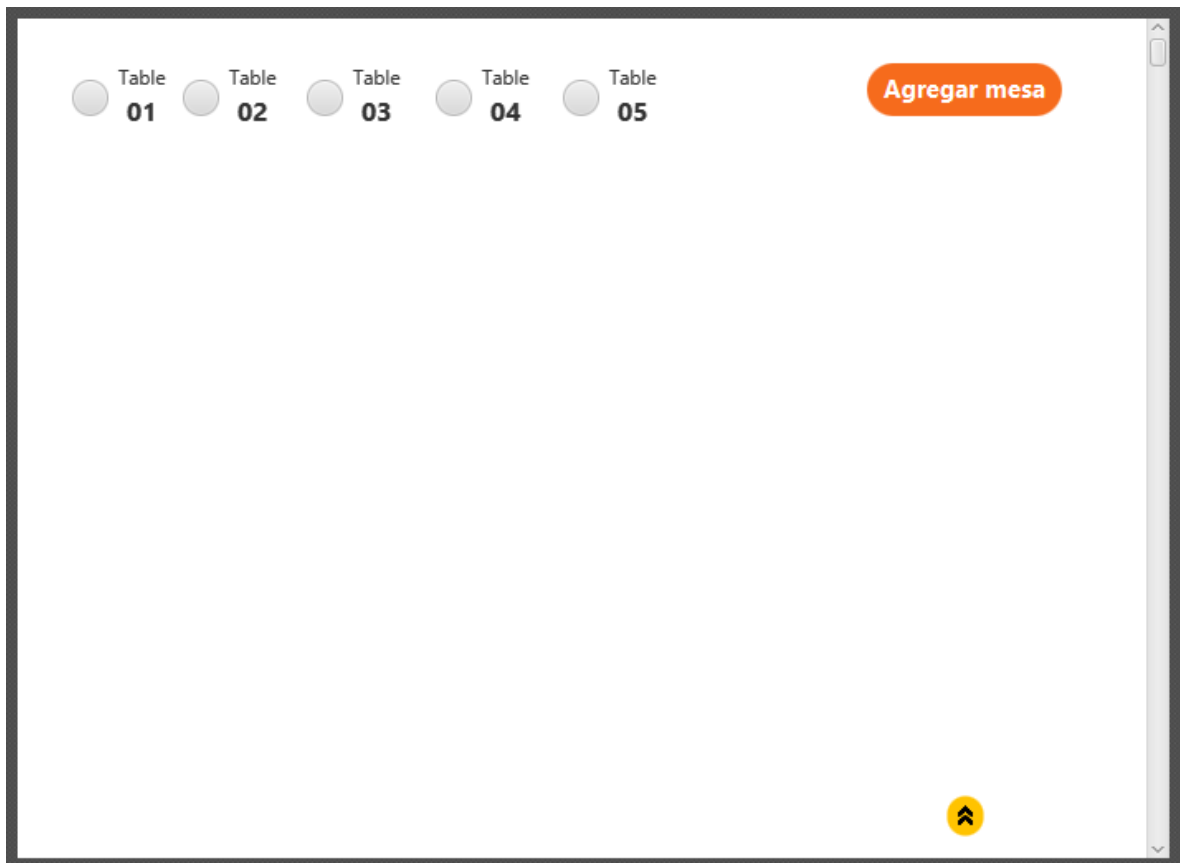
☐ 3

☐ 4

☐ 5

¿Alguna observación o sugerencia?

Pantalla de mesas: Esta pantalla se despliega cuando hay muchas mesas y el usuario quiere escoger alguna que no está en la primera fila.



Diseño de Escenarios

| Nombre | Clase | Escenario |
|--------------------------|----------------|---|
| productEmptyList | RestaurantTest | <pre> sequenceDiagram participant R as :Restaurant R->> Products: Products </pre> |
| productNotEmptyList | RestaurantTest | <pre> sequenceDiagram participant R as :Restaurant R->> Products: Products activate Products Products->> P as :Product deactivate Products P->> R: id="10277" name="Pasta Alfredo" category="Pastas" size="Normal" price=40000 availability=100 description="" </pre> |
| ClientEmptyLinkedList | RestaurantTest | <pre> sequenceDiagram participant R as :Restaurant R->> C as :Client: firstClient </pre> |
| ClientNotEmptyLinkedList | RestaurantTest | <pre> sequenceDiagram participant R as :Restaurant R->> C1 as :Client: firstClient activate C1 C1->> C2 as :Client: next C1->> C3 as :Client: previous deactivate C1 </pre> |

Diseño de Pruebas

| Objetivo: Verificar la correcta creación de un producto | | | | |
|---|------------|---------------------|--|---|
| Clase | Método | Escenario | Valores Entrada | Resultado |
| Restaurant | addProduct | productEmptyList | Id="1098" Name="Baby Beef" Cat="Carne" Size="Normal" Price=22000 Avail=50 Desc="" | Se ha creado un nuevo producto y la lista de productos ya no es vacía y tiene un solo elemento. |
| Restaurant | addProduct | productNotEmptyList | Id="1099" Name="Pizza Peperoni" Cat="Pizzas" Size="Grande" Price=32000 Avail=50 Desc="" | Se ha creado un nuevo producto a la lista de productos la cual tiene un tamaño de dos. |
| Restaurant | addProduct | productNotEmptyList | id="10277" name="Pasta Alfredo" category="Pastas" size="Normal" price=40000 availability=100 description=" | El producto no ha podido ser añadido porque ya existe un producto con ese número de id. |

| Objetivo: Verificar la correcta eliminación de un producto, que retorne true or false | | | | |
|---|---------------|---------------------|-----------------|--|
| Clase | Método | Escenario | Valores Entrada | Resultado |
| Restaurant | deleteProduct | productEmptyList | Id="1098" | False, el producto no pudo ser eliminado porque no existe |
| Restaurant | deleteProduct | productNotEmptyList | Id="10277" | True, el producto fue eliminado y la lista de productos está vacía |

| Objetivo: Verificar la correcta actualización de un producto, que retorne true or false | | | | |
|---|---------------|---------------------|--|--|
| Clase | Método | Escenario | Valores Entrada | Resultado |
| Restaurant | updateProduct | productEmptyList | id="10277" name="Pasta Alfredo" category="Pastas" size="Normal" price=40000 availability=100 description=" | False, el producto no pudo ser actualizado porque no existe |
| Restaurant | updateProduct | productNotEmptyList | id="10277" name="Pasta Alfredo" category="Pastas" size="Normal" price=40000 availability=100 description=" | True, el producto fue actualizado y el tamaño de la lista es 1 |

| Objetivo: Verificar la correcta creación de un cliente | | | | |
|--|-----------|--------------------------|---|--|
| Clase | Método | Escenario | Valores Entrada | Resultado |
| Restaurant | addClient | ClientEmptyLinkedList | Id="4550" Name="Pedro" Adress="av7n" Pone="3122870" Obs="Es Hombre" | Se ha creado un nuevo cliente y firstClient ya no es null, el next de firstClient es null. |
| Restaurant | addClient | ClientNotEmptyLinkedList | Id="4550" Name="Pedro" Adress="av7n" Pone="3122870" Obs="Es Hombre" | Se ha creado un nuevo cliente, firstClient.getNext ya no es null. |
| Restaurant | addClient | ClientNotEmptyLinkedList | Id="2345" Name="Pedro" Adress="av7n" Pone="3122870" Obs="Es Hombre" | El cliente no ha podido ser creado porque ya existe un cliente con el mismo id. |

| Objetivo: Verificar la correcta eliminación de un cliente, que retorne true or false | | | | |
|--|--------------|-----------------------|-----------------|---|
| Clase | Método | Escenario | Valores Entrada | Resultado |
| Restaurant | deleteClient | ClientEmptyLinkedList | Id="1098" | False, el cliente no pudo ser eliminado porque firstClient es |

| | | | | |
|------------|--------------|--------------------------|-------------|---|
| | | | | null, es decir no hay clientes. |
| Restaurant | deleteClient | ClientNotEmptyLinkedList | Id="2345" | True, el cliente fue eliminado y la lista de productos está vacía |
| Restaurant | deleteClient | ClientNotEmptyLinkedList | Id="459837" | False, el cliente no fue eliminado porque no existe en la linked List |

| Objetivo: Verificar la correcta actualización de un producto, que retorne true or false | | | | |
|---|--------------|--------------------------|---|--|
| Clase | Método | Escenario | Valores Entrada | Resultado |
| Restaurant | updateClient | ClientEmptyLinkedList | Id="4550" Name="Pedro" Adress="av7n" Pone="3122870" Obs="Es Hombre" | False, el producto no pudo ser actualizado porque firstClient es null. |
| Restaurant | updateClient | ClientNotEmptyLinkedList | Id="2345" Name="Pedro" Adress="av7n" Pone="3122870" Obs="Es Hombre" | True, el producto fue actualizado y firstClient no es null, pero firstClient.getNext si. |
| Restaurant | updateClient | ClientNotEmptyLinkedList | Id="7957868" Name="polito" Adress="av7n" Pone="3122870" Obs="Es Hombre" | False, el cliente no pudo ser actualizado porque no existe el cliente |

Diagrama de clases