

Documentación de soporte de la tercera iteración de la aplicación Eat To Go

A:

Ing. Wilson Libardo Pantoja Yepes

Por:

Alejandro Latorre Pabón
Carlos Alberto Salamanca Quijano
Christian David Tobar Mosquera
Juliana Mora López
Yeferson Benavides



Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Programa: Ingeniería de Sistemas

Asignatura: Laboratorio de Ingeniería del Software II

Popayán, diciembre de 2020

1. Diseñar historias de usuario:

Historias de usuario a implementar en Eat To Go

Historia: Registro del menú semanal

Como: Administrador de un restaurante

Quiero: Deshacer las últimas acciones realizada dentro de la interfaz del menú

Para: Para recuperar un estado anterior del proceso.

Reverso

Criterios de aceptación:

- Dado que estoy en la interfaz de registro del menú semanal, cuando se haya registrado cualquiera de las partes del almuerzo, entonces, se habilitará un botón de deshacer que permite recuperar un estado anterior del registro.

Historia: Registro del menú semanal

Como: Administrador de un restaurante

Quiero: Poder registrar el menú del día con entrada, principio, proteína y bebida.

Para: Que un cliente pueda visualizar dicho menú y realizar un pedido.

Reverso

Criterios de aceptación:

- Dado que estoy en la interfaz de registro del menú semanal, cuando se haya registrado el precio que tendrá el almuerzo del restaurante, entonces, se habilita la selección de días, además del registro de las partes del plato.
- Dado que estoy en la interfaz de registro del menú semanal y haya seleccionado un día de la semana, para realizar el registro de las partes de un plato como son: entrada, principio, proteína y bebida, los campos de dichas partes: "Nombre", e "imagen", están llenos y son válidos cuando presione el botón registrar de cada una de estas, entonces, se va a guardar en una BD.

- Dado que estoy en la interfaz de registro de menú semanal, cuando se quiera registrar las partes de un plato en un día específico, entonces, se mostrará una lista con los registros que se encuentran previamente almacenados para que el administrador pueda visualizar lo que existe en la BD.
- Dado que estoy en la interfaz de registro del menú semanal, cuando se registre cada una de las partes del plato, entonces, se debe mostrar una ventana emergente indicando el resultado de la operación.

Historia: Visualización de la lista de restaurantes**Como: Cliente de la aplicación****Quiero:** Poder visualizar la información de los restaurantes**Para:** Seleccionar cada uno y visualizar su menú semanal.**Reverso****Criterios de aceptación:**

- Dado que estoy en la interfaz de visualización de los restaurantes disponibles, cuando seleccione el logo o el nombre de un restaurante, entonces, se debe mostrar al usuario la interfaz del menú semanal de dicho restaurante.

Historia: Visualización y selección de menú de un restaurante**Como: Cliente de la aplicación****Quiero:** Poder visualizar y seleccionar las distintas partes que componen un almuerzo.**Para:** agregar al carrito de compra.**Reverso****Criterios de aceptación:**

- Dado que estoy en la interfaz de visualización y selección del menú de un restaurante y he seleccionado todas las partes de un almuerzo, cuando presione el botón "agregar a mi carrito", entonces, se debe mostrar un mensaje que indique el resultado

de la operación.

- Dado que estoy en la interfaz de visualización y selección del menú de un restaurante y he agregado al menos un ítem al carrito, cuando presione el botón “Ir al carrito”, entonces, se debe mostrar la interfaz de resumen del pedido.
- Dado que estoy en la interfaz de visualización y selección del menú de un restaurante, cuando presione el botón “Regresar”, entonces, se debe volver a la interfaz de visualización de la lista de restaurantes disponibles.

Historia: Visualización del resumen del pedido

Como: Cliente de la aplicación

Quiero: Poder visualizar la información de los almuerzos que he agregado al carrito

Para: Poder modificar o finalizar el pedido.

Reverso

Criterios de aceptación:

- Dado que estoy en la interfaz de visualización del resumen del pedido, cuando presione el botón “Pedir”, entonces, se debe mostrar un mensaje con el resultado de la operación.
- Dado que estoy en la interfaz de visualización del resumen del pedido, cuando presione el botón de “Regresar”, entonces, se debe volver a la interfaz de visualización y selección del menú de un restaurante.

2. Diseñar mockups:

Diseño preliminar de las interfaces de usuario de la aplicación

Correspondiente a esta segunda iteración, se ha decidido implementar el registro del menú semanal, la visualización de restaurantes, visualización y selección del menú y visualización del resumen del pedido. Se diseñaron los siguientes prototipos:

Registro de menú semanal:


Eat To Go

Registro de menú semanal

Ingrese el precio de los platos

Selecione el día


Lunes
Martes
Miercoles
Jueves
Viernes
Sabado
Domingo



EntradaPrincipioProteínaBebida

Nombre

Descirpción


 Subir imagen


Registrar


Visualización de restaurantes:


Eat To Go


Restaurantes


 Pío Pío

 Tequila's

 La cosecha

 McDonalds

 Sr. Wok

 Starbucks

Visualización y selección de menú:

Eat To Go

Restaurant name

Menú del día lunes

Precio del almuerzo: \$9.999

Seleccione la entrada

Mazamorra

Sancocho

Fruta

Seleccione el principio

Pure' de papa

Lentejas

Frijoles

Seleccione la proteína

Chuleta de pollo

Cordon bleu

Cerdo agridulce

Seleccione la bebida

Jugo de mango

Limonada

Platos especiales

Seleccione la entrada

Ajiaco

\$9.999

3

Churrasco

\$9.999

3

Camarones apanados

\$9.999

3

Lasaña

\$9.999

3

Ensalada de pollo

\$9.999

3

Pasta pomodoro

\$9.999

3

Seleccionado

Mazamorra

Pure' de papa

Cordon bleu

Limonada

Regresar

Agregar a mi carrito

Ir al carrito

Visualización de resumen de pedido y realización de pedido:

Eat To Go

Resumen del pedido

Mazamorra

Leche, maiz blanco y canela en polvo

Pure' de papa

Cremoso pure' de papa pastusa con queso y un toque leche

Cordon bleu

Pechuga enrollada y rellena de queso y jamón

Limonada

Limonada con panela

3

\$9.999

Lasaña

relleno elaborado a base de asado de ternera y cerdo, tomate, verduras frescas y especias. Laminada con placas de pasta de la mejor calidad y cubierta de suave bechamel.

3

\$9.999

Regresar

Pedir \$99.999

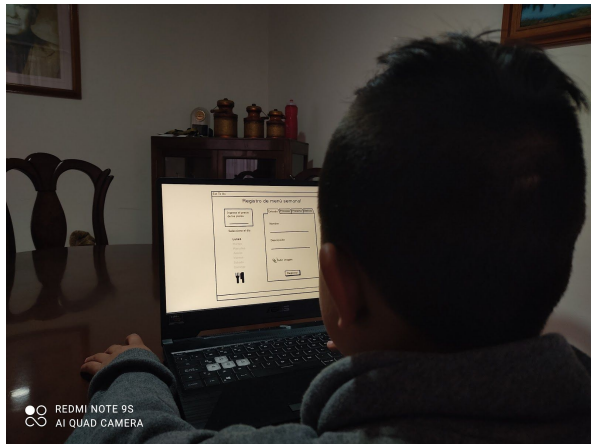
3. Validar usabilidad con Thinking Aloud:

Test de usabilidad (Thinking Aloud)

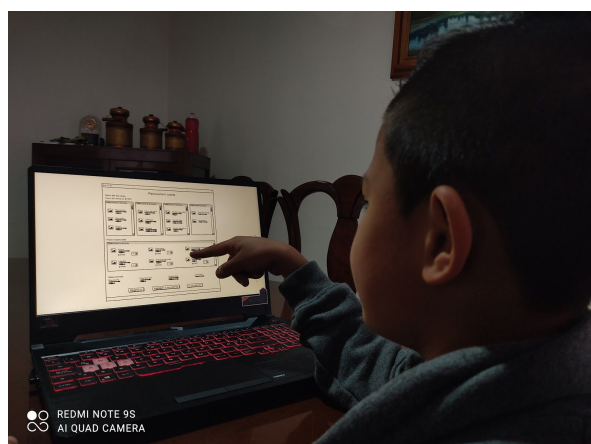
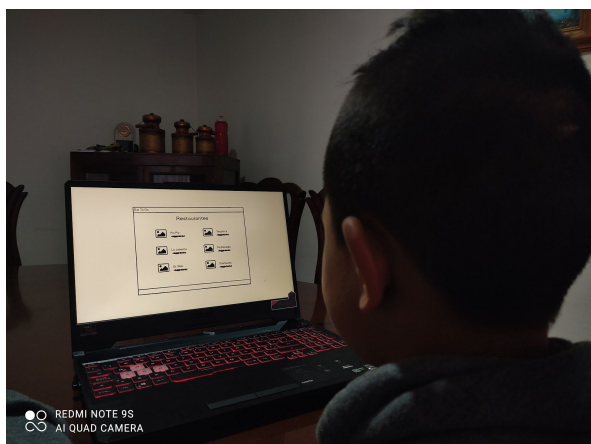
Para esta oportunidad también se realizó un test de usabilidad para los prototipos correspondientes a la segunda iteración. Se seleccionó a algunas personas que hacen las veces de usuarios finales para que puedan expresar las opiniones que tienen acerca de las interfaces presentadas y de este modo poder determinar si la aplicación es intuitiva y comprensible.

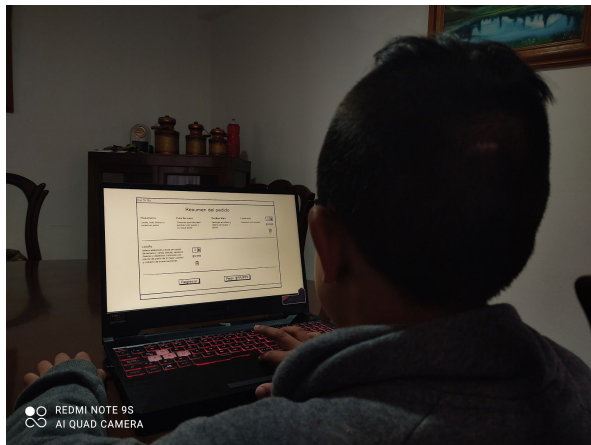
A continuación se muestra el proceso realizado:

Proceso de validación:

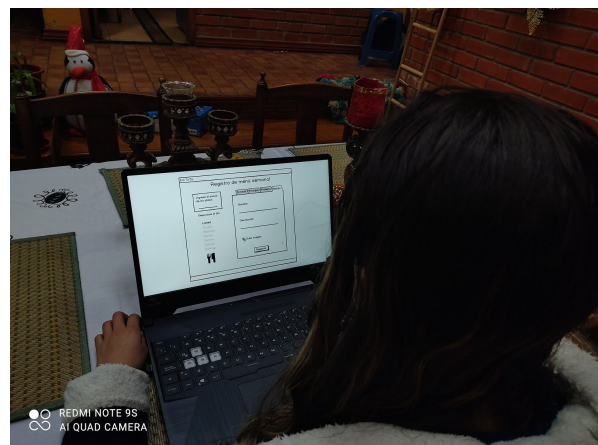


En esta primera parte con respecto al registro del menú semanal, el usuario expresó que la interfaz es de su agrado y comentó que podía entender fácilmente el funcionamiento de la aplicación. Sin embargo, mencionó que para la parte del registro del precio de los almuerzos hacía falta algo adicional, como un botón, ya que de otra manera estaría relacionado con el botón de registro de cada parte que compone un plato y esta no era la intención debido a que dicho precio solamente se debe registrar una vez.



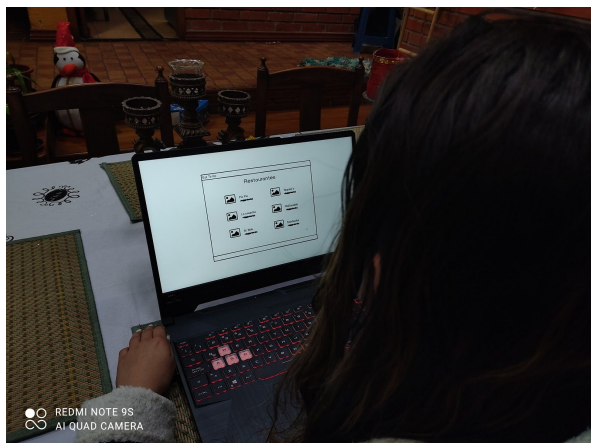


Para esta segunda parte que corresponde tanto a la visualización de restaurantes, visualización y selección de platos y resumen del pedido, el usuario mencionó que también le agrada la presentación de las interfaces, que es fácil de usar y es posible entender lo que se quiere mostrar, a excepción de la interfaz de resumen de pedido que tiene mucha información en ellas.

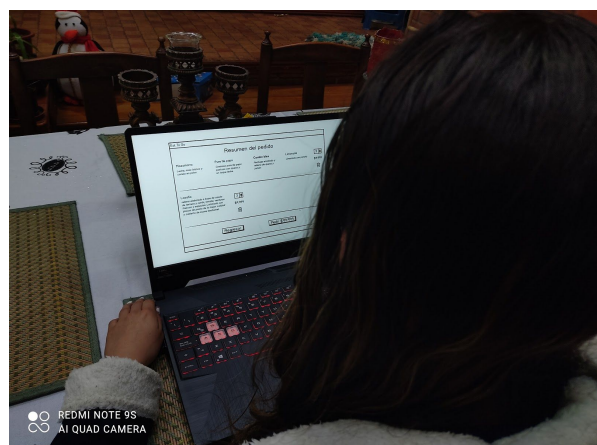
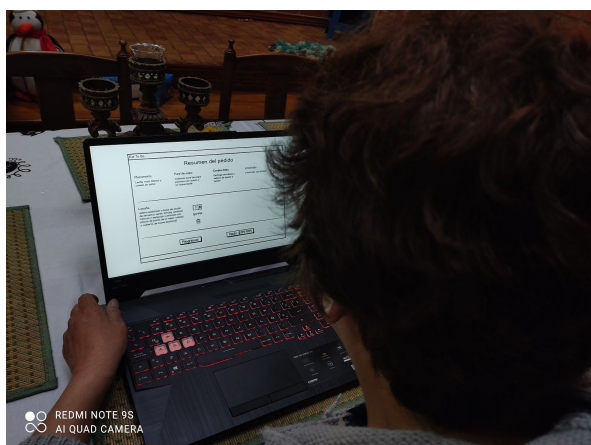


Los usuarios evaluados anteriormente coincidieron en que las interfaces son agradables y usables, pero en la interfaz de registro de menú semanal realizaron la recomendación de agregar un espacio que muestre lo que se encuentra previamente registrado tanto por día, como por tipo (Entrada, Proteína, Principio y Bebida), de esta forma se evitará cometer errores

como en el caso de registrar una parte que ya exista. También se identificó la posibilidad de borrar o modificar una de estas partes que ya se encuentran registradas.



En la parte del cliente con los los usuarios de las imágenes anteriores opinaron que la interfaz de restaurantes no es lo suficientemente clara con lo que desea hacer, por tanto debe indicar cual es la operación a realizar sobre ella. Con respecto a la interfaz de selección del menú, mencionaron que se encuentra algo saturada de información y puede ser más agradable si se la presenta de una manera más sencilla y con menos información en ellas.




En la sección de resumen de pedido, en general los comentarios fueron positivos.

A partir del test se logró detectar algunas posibles mejoras y correcciones para los prototipos presentados para conseguir unas interfaces usables y comprensibles para el usuario. Las interfaces se actualizaron de la siguiente manera:

Eat To Go

Registro Menú Semanal




Precio almuerzos

Registrar precio

Seleccione el día
☒ **Lunes**
☐ Martes
☐ Miércoles
☐ Jueves
☐ Viernes
☐ Sábado
☐ Domingo

Entrada Principio Proteína Bebida

Nombre

Subir imagen


Registrar

Entradas Registradas

Nombre


Mazamorra
Fruta
Sancocho

Borrar


Eat To Go

Restaurantes


Seleccione un restaurante




Pio Pio
★★★★★




Tequila's
★★★★★




La cosecha
★★★★★



McDonalds
★★★★★



Sr. Wok
★★★★★



Starbucks
★★★★★


Eat To Go


Restaurant name


Menú del día lunes

Combina tu almuerzo:


Seleccione la entrada


 Mazamorra


 Fruta

 Sancocho


Seleccione el principio


 Frijoles


 Lentejas

 Puré de papa


Seleccione la proteína


 Cerdo agridulce

 Chuleta de pollo

 Cordon blue

Seleccione la bebida

 Jugo de mango

 Limonada natural

Precio Almuerzo: \$9.999

Regresar

Agregar a mi carrito

Ir al carrito

Eat To Go

Resumen del pedido

Entrada	Principio	Proteína	Bebida	Bebida
Mazamorra	Puré de papa	Chuleta de pollo	Limonada	\$9.999
Mazamorra	Puré de papa	Chuleta de pollo	Limonada	\$9.999

Regresar

Pedir \$99.999

Borrar

En estas últimas imágenes, se presentan los cambios realizados a los primeros diseños presentados. En la interfaz de registro de menú semanal se observa que se ha agregado un botón que permite el registro del precio de los almuerzos y el espacio que muestra los registros que se encuentran en la Base de Datos.

También se observa en la interfaz de visualización y selección del menú, que ahora se encuentra menos saturada de información y es más clara con la función que pretende realizar.

4 Diseñar escenarios de atributos de calidad:

Especificación de requerimientos no funcionales

Cod. Escenario: ECR01	
Descripción La estructura de la aplicación debe permitir la adición de componentes y funcionalidades de forma tal que el cambio no interfiera excesivamente con la lógica de otros componentes.	
Interesado: Desarrollador	Atributo: Modificabilidad
Validación del Escenario	
Origen del estímulo	Cliente del restaurante
Estímulo	Agregación de nuevos componentes del sistema
Entorno	Ámbito de desarrollo
Artefacto	Aplicación
Respuesta	Los cambios se pueden aplicar satisfactoriamente sin mayor cambio en otros componentes
Medida de respuesta	La cantidad de componentes afectados es menor a 1/4 de la cantidad total

Cod. Escenario: ECR02	
Descripción La aplicación debe poder atender un volumen de 1000 usuarios conectados al mismo tiempo.	
Interesado: Product Owner	Atributo: Escalabilidad
Validación del Escenario	
Origen del estímulo	Clientes de la aplicación

Estímulo	Muchos usuarios están usando los servicios de la aplicación al mismo tiempo
Entorno	En un día de semana entre las 11am y 2pm
Artefacto	Sistema
Respuesta	El sistema debe responder eficazmente para cada uno de los usuarios
Medida de respuesta	La aplicación debe de atender a los 1000 usuarios sin caerse

Cod. Escenario: ECU01	
Descripción La interfaz debe mostrar la información del menú del día de un restaurante cuando el usuario lo solicite.	
Interesado: Cliente	Atributo: Usabilidad
Validación del Escenario	
Origen del estímulo	Cliente del restaurante
Estímulo	Solicitud de visualización de platos
Entorno	Condiciones normales
Artefacto	La interfaz de visualización y selección del menú
Respuesta	El sistema responde mostrando una interfaz con la información del menú del día del restaurante seleccionado
Medida de respuesta	El usuario puede consultar los platos del restaurante de manera sencilla e intuitiva cometiendo cero errores en el proceso

Cod. Escenario: ECU02	
Descripción El administrador debe ser capaz de usar la aplicación sin la necesidad de manuales de usuario o instrucciones dentro de la aplicación.	
Interesado: Administrador	Atributo: Usabilidad
Validación del Escenario	
Origen del estímulo	Administrador

Estímulo	Registro del menú de la semana
Entorno	Condiciones normales
Artefacto	Aplicación
Respuesta	El sistema ofrece una interfaz de usuario fácil de aprender, intuitiva y amigable
Medida de respuesta	El administrador debe tardar menos de 1 hora en realizar el registro del menú para toda la semana, la primera vez

Cod. Escenario: ECU03	
Descripción El cliente debe ser capaz de usar la aplicación sin la necesidad de manuales de usuario o instrucciones dentro de la aplicación.	
Interesado: Cliente	Atributo: Usabilidad
Validación del Escenario	
Origen del estímulo	Cliente del restaurante
Estímulo	Interacción del cliente con las distintas interfaces destinadas a éste
Entorno	Condiciones normales
Artefacto	Interfaces de usuario del cliente
Respuesta	El sistema ofrece una interfaz de usuario fácil de aprender, intuitiva y amigable
Medida de respuesta	El cliente debe tardar menos de 2 minutos en realizar un pedido

5. Elegir tipo de aplicación:

WEB para los servicios y Aplicación de Escritorio para el cliente

Dada la naturaleza de la arquitectura de microservicios se hace necesario que sean implementados bajo un modelo de aplicación web garantizando así la disponibilidad de los recursos que cada uno provee a los demás componentes de la aplicación.

Lo anterior nos permite poder adaptar libremente la capa del cliente al o los tipos de aplicaciones que se requieran durante el ciclo de vida del software, en este caso se iniciará implementando una aplicación de escritorio como capa de presentación

6. Estilo arquitectónico base:

Arquitectura de Microservicios

Las especificaciones de la aplicación, junto con el contexto de desarrollo, sugieren que se debe diseñar descomponiéndola en subsistemas autónomos en forma de microservicios de colaboración y contenedores, de manera tal que cada servicio implemente un conjunto de funciones coherentes relacionadas entre sí; posibilitando escalar horizontalmente áreas individuales de la aplicación; permitiendo la división del trabajo entre varios equipos de desarrollo; garantizando el aislamiento de posibles problemas y la adopción de las tecnologías más recientes para el desarrollo.

Se han tenido en cuenta sus aspectos técnicos más complejos a la hora de escoger esta arquitectura, como lo puede ser el carácter distribuido que tendría la aplicación y los retos que ello implica a la hora de implementar la comunicación entre servicios mediante protocolos como HTTP, sin embargo se cree firmemente que una vez superado este proceso, las ventajas que nos brinda la arquitectura compensan en gran medida estos retos.

7. Estilo arquitectónico a aplicar

Arquitectura Hexagonal

Siguiendo las bases que sienta la arquitectura de microservicios, donde se independizan en distintas unidades funcionales todos los aspectos de la aplicación, se ha decidido empezar a aplicar una arquitectura que permita extender esta filosofía a cada microservicio, encontrando que la arquitectura hexagonal nos permite separar cada uno de ellos en distintas capas o regiones con su propia responsabilidad. lo que nos facilita entre otros aspectos: las labores de testing de las diferentes partes de manera aislada; El uso de Inyección de Dependencias, ya que nos facilitará la comunicación y el acceso entre las diferentes capas o cambiar de framework por ejemplo de uno no reactivo a uno que permita la programación reactiva.

8. Patrones de diseño aplicados

Command, Decorator y Factory

Se hizo la implementación de patrón Command para agregar la funcionalidad de poder deshacer una operación en la interfaz de registro de menú semanal, de tal manera que el administrador pueda recuperar en todo momento un estado anterior del registro.

Se utiliza el patrón Decorator para dar formato a las entradas de texto dentro de la interfaz de registro del menú semanal de forma tal que las palabras ingresadas queden con la primera letra en mayúscula.

Se utiliza el patrón Factory con el fin de trabajar con implementaciones de la interface IOrderRepository, por el momento se tiene una que utiliza JPA y CrudRepository pero se deja el patrón implementado para futuras extensiones. En conjunto con este patrón se usa el patrón o antipatrón Singleton con el fin de tener una única instancia de la clase FactoryRepository.

9. Elegir tecnología:

Tecnología que mejor se ajusta a las necesidades del proyecto

Al haber escogido trabajar con la arquitectura de microservicios se hizo una evaluación de los frameworks y tecnologías que nos brindaran las mayores ventajas respecto a dicho modelo. En este sentido encontramos que Spring Framework proporciona un gran abanico de herramientas que facilitan el desarrollo de esta arquitectura como lo son:

Spring Boot: que facilita la configuración inicial de la aplicación, la resolución de las dependencias y el despliegue a través de un servidor integrado como es el caso de Tomcat.

También se ha escogido Java como lenguaje principal de la capa de negocios de cada microservicio y en especial por el soporte, la versatilidad y amplia extensión de este lenguaje.

10. Diseñar el diagrama de clases:

Diagrama de clases del proyecto:

En este proyecto se decidió utilizar el modelo de documentación C4, la cual nos permite tener niveles de abstracción que detallan el software de manera cada vez más precisa y técnica empezando con un vistazo muy general del sistema, se realizaron el diagrama de contexto, diagrama de contenedores, diagrama de componentes, para el diagrama de código decidimos utilizar un diagrama de paquetes para la vista general, y los respectivos diagramas de clase-paquete dónde se detalla la implementación

Diagrama de contexto (nivel 1)

Context Diagram

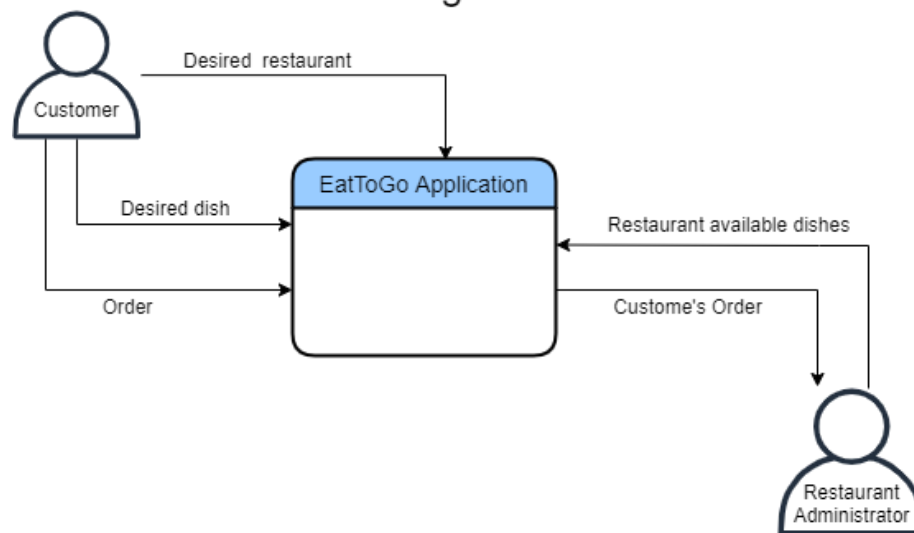


Diagrama de contenedores (nivel 2)

Containter Diagram

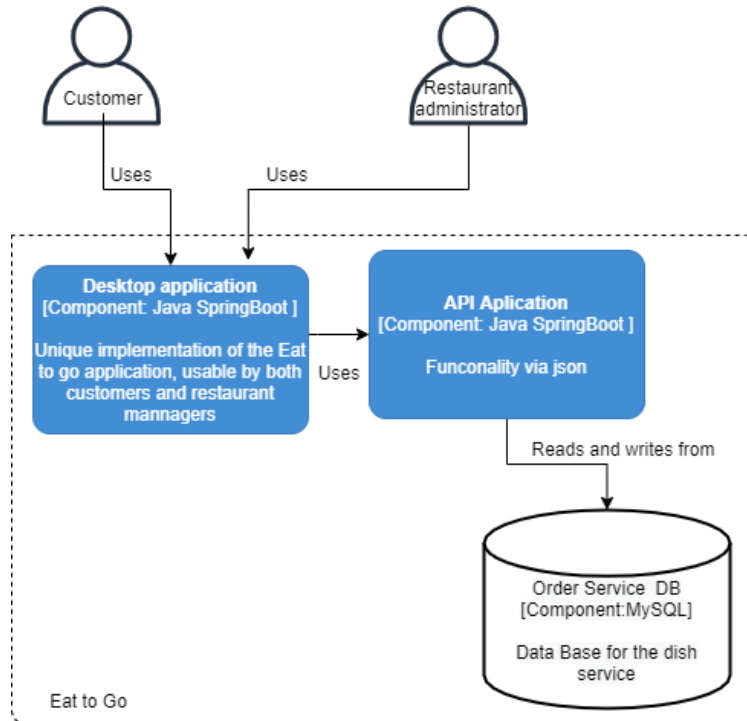


Diagrama de Componentes (nivel 3)

Componet Diagram

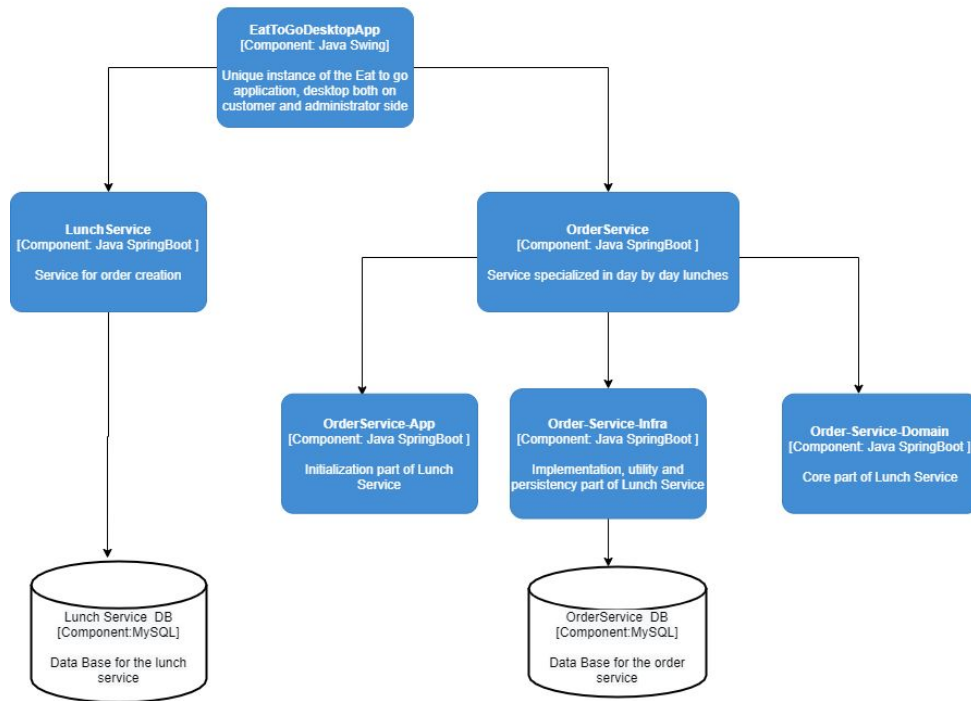
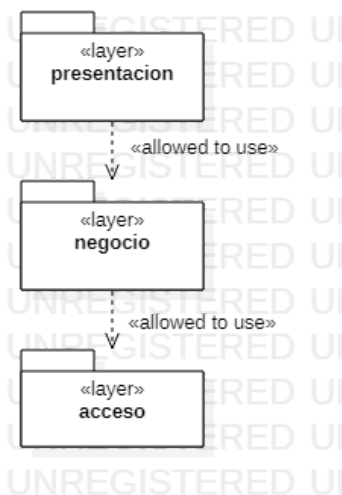
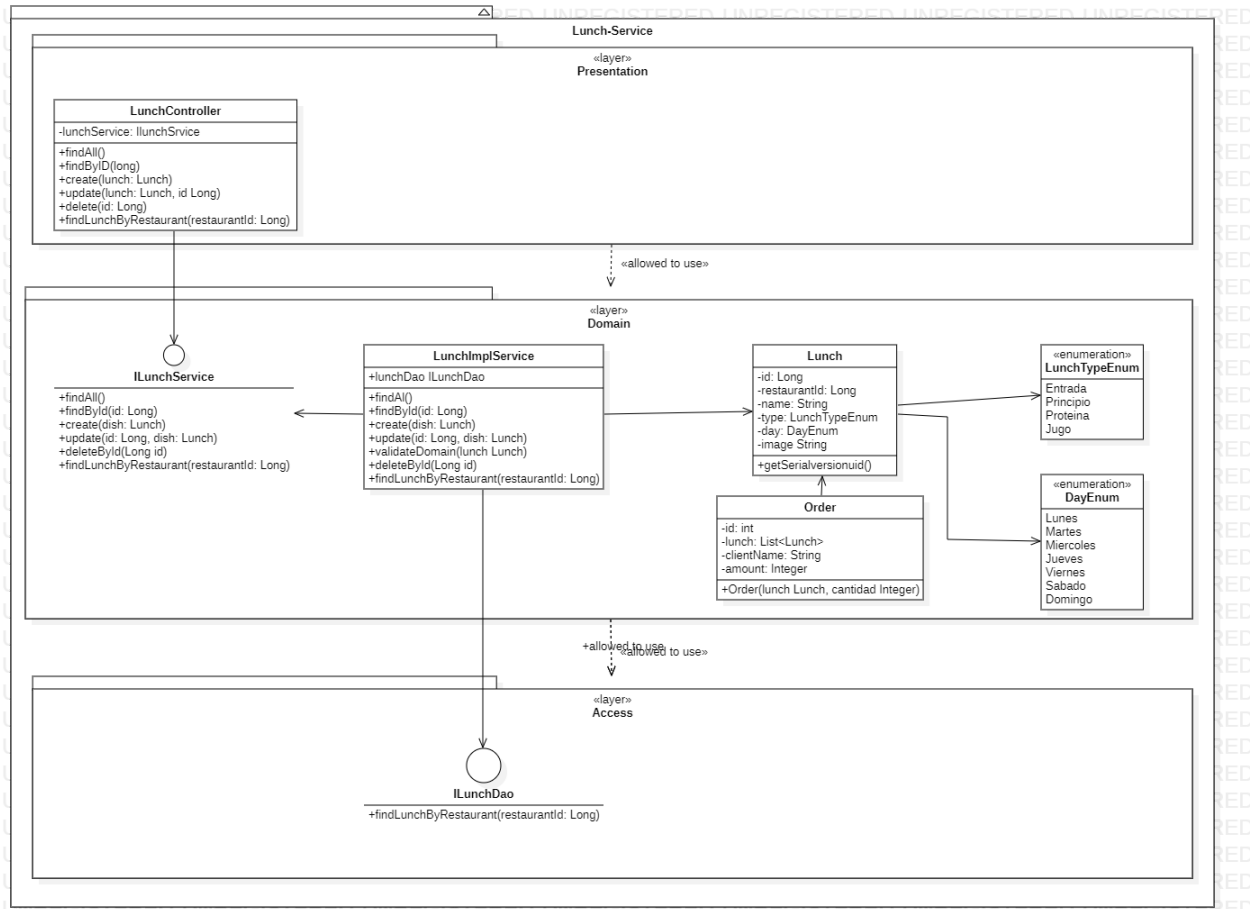


Diagrama de Código (nivel 4)

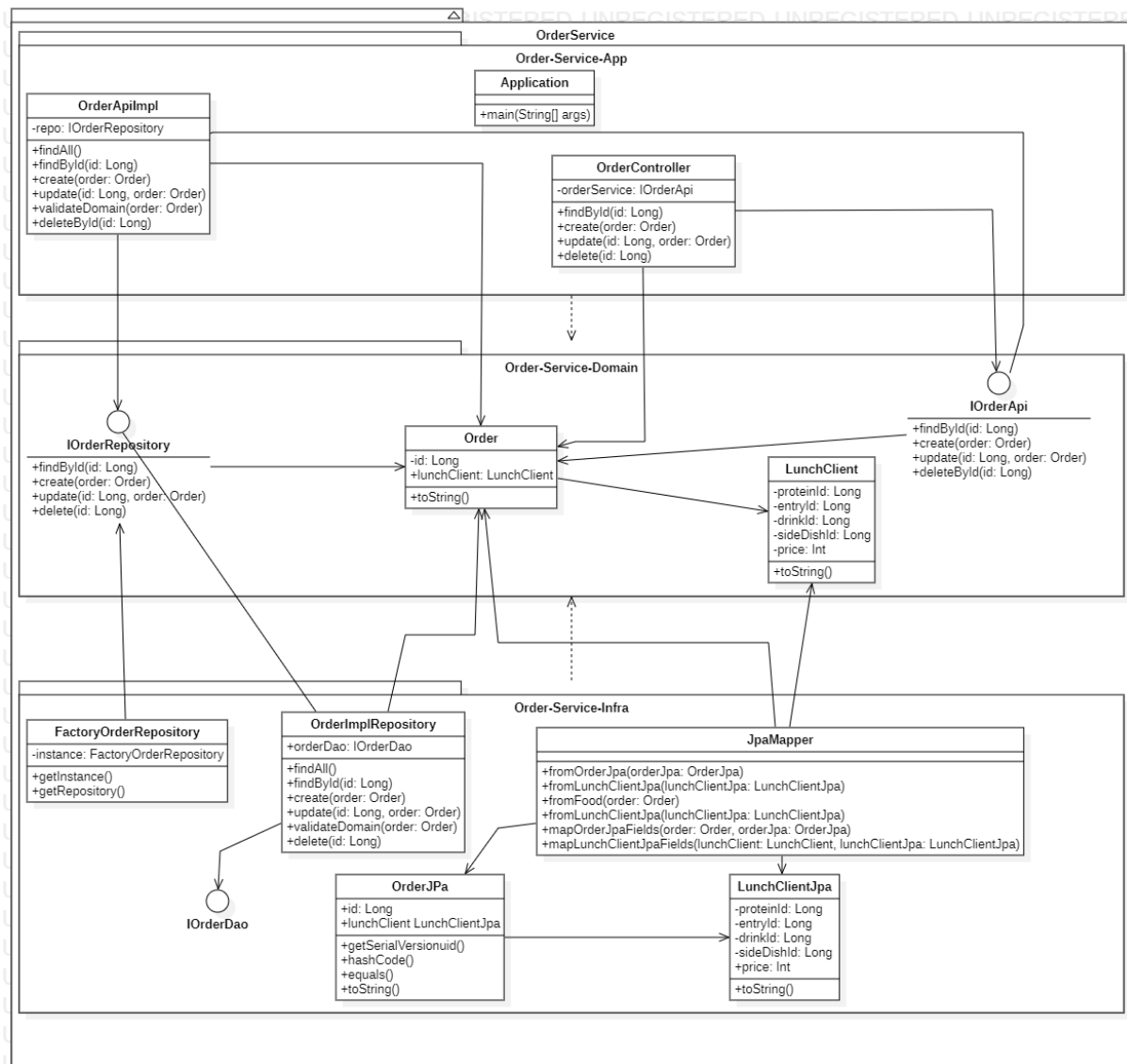
La arquitectura de paquetes es equivalente para los diferentes microservicios. los tres son representados por el mismo diagrama



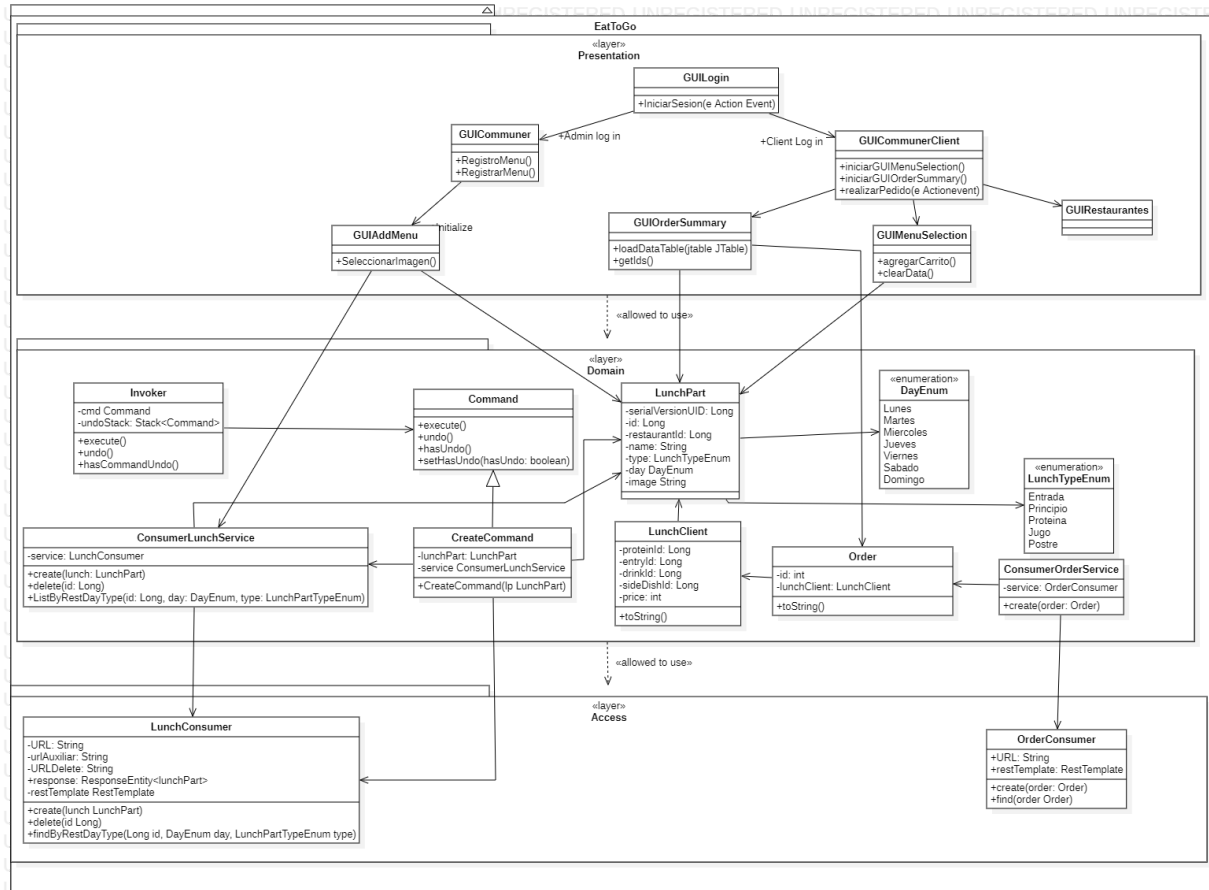
A continuación los respectivos diagramas de clase, empezando por el diagrama de Lunch Service:



El diagrama de Order Service:



Y finalmente el diagrama de nuestra clase principal EatToGo:



9. Implementación:

Link al repositorio en GitHub

<https://github.com/Carlos-Salamanca/EatToGo-Tercera-Iteracion>

