
SoftMakers

Eattrip

Software Architecture Document

Version 1.0

Historial de versiones

Date	Version	Description	Author
08/10/2018	1.0	Versión inicial	Facundo Astiz, Miguel Di Luca, Fernando Martin, Ruben Schultz

Table of Contents

1.1.	Introducción	4
1.1.	Propósito	4
1.2.	Alcance	4
1.3.	Vision en conjunto	4
2.	Representación arquitectónica	4
3.	Metas y restricciones arquitectónicas	4
4.	Vista de Casos de Uso	4
5.	Vista de Proceso	6
6.	Vista Lógica	7
7.	Vista de Despliegue	7
8.	Vista de Implementación	7
8.1.	Vista conjunta	7
8.2.	Capas	8
9.	Vista de Datos	8
10.	Tamaño y Rendimiento	8
11.	Calidad	8

1. Introducción

1.1. Propósito

Este documento proporciona una descripción arquitectónica completa del sistema.

1.2. Alcance

Este documento de arquitectura de software proporciona una descripción general de la arquitectura del sistema EatTrip, utilizando la plantilla genérica SAD.

1.3. Vision en conjunto

Este documento está dividido en diferentes secciones, cada una de las cuales describe en detalle una determinada vista del proyecto. Estas son las vistas de caso de uso, lógica, proceso, implementación y datos. También contiene una descripción del tamaño, el rendimiento y la calidad de la arquitectura del sistema.

2. Representación arquitectónica

El modelado, la implementación y documentación de un sistema requiere que el sistema sea visto de diferentes perspectivas. Por esto, la arquitectura será representada en un enfoque de cinco vistas. Vista de casos de uso, vista de proceso, vista de diseño, vista de datos, vista lógica y vista de despliegue. Estas serán explicadas más adelante.

3. Metas y restricciones arquitectónicas

Hay algunos requisitos clave y restricciones del sistema que tienen una influencia significativa en la arquitectura. La portabilidad es una prioridad importante, por lo que el sistema estará disponible para los usuarios finales desde diferentes plataformas. La seguridad y la privacidad también son preocupaciones importantes, por lo que los datos confidenciales, como la dirección o la información de facturación, serán privados. La restricción más relevante es el apretado lapso de tiempo dado al equipo de desarrollo para implementar y probar el sistema para que sea completamente funcional.

4. Vista de Casos de Uso

Los siguientes casos de uso representan las capacidades más relevantes del sistema final, por lo que requieren una gran cobertura arquitectónica.

- **Registración del usuario**

Este caso de uso describe cómo el usuario crea una cuenta para acceder a las funciones del sistema.

- **Login del Usuario**

Este caso de uso explica cómo el usuario inicia sesión en el sistema ingresando sus credenciales.

- **Modificación de datos del Usuario**

Este caso de uso describe cómo los usuarios pueden modificar la información personal de su cuenta.

- **Realizar un pedido**

Este caso de uso describe cómo el usuario puede realizar un pedido.

- **Editar producto**

Permite al restaurante editar los datos de un producto.

- **Búsqueda de productos**

El usuario podrá buscar un producto mediante palabras clave.

- **Baja de producto**

El restaurante podrá dar de baja un producto si no desea ofrecerlo a los usuarios.

- **Alta de producto**

Brinda la opción al restaurante de agregar un nuevo producto en el menú.

- **Confirmar o rechazar pedido**

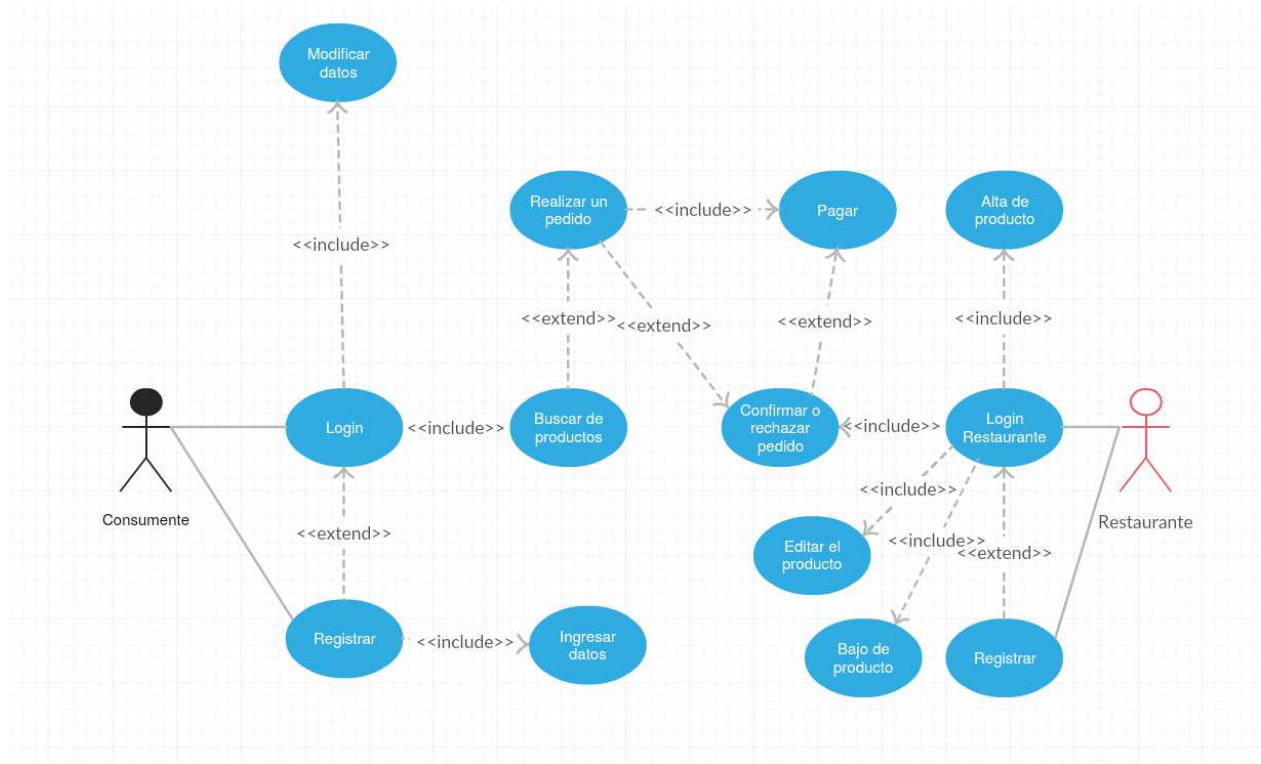
El restaurante confirmará o rechazará los distintos pedidos que los usuarios realicen.

- **Modificación de producto**

El restaurante podrá editar los datos y el precio de sus productos.

- **Modificación de restaurante**

El administrador podrá editar los datos de usuario.



5. Vista de Proceso

Los procesos del sistema generalmente consisten en el paso de mensajes entre el usuario y el sistema, o entre el administrador y el sistema, según el caso de uso al que se refiera la situación. El sistema no admite subprocesos múltiples, por lo que los procesos se producen de uno en uno.

Proceso de Registro

El sistema verifica los campos ingresados por el usuario en el formulario de registro y muestra un mensaje de error.

Proceso de Login

El sistema verifica las credenciales introducidas por el usuario y muestra un mensaje de error u otorga acceso al sistema en consecuencia.

Proceso de editado de pedido

El sistema muestra productos relacionados con la entrada de búsqueda del usuario. Cuando el usuario selecciona la opción "Agregar al carrito", se envía un mensaje al sistema para que lo haga.

Proceso de salida

El usuario informa al sistema qué método de pago se utilizará. Si se usa una tarjeta de crédito, el sistema le informará al usuario si la información ingresada es correcta o no, y le enviará un mensaje de confirmación si la transacción fue exitosa.

6. Vista Lógica

Desde el punto de vista lógico se tienen cuatro pilares fundamentales. Los dos principales son:

- Los **usuarios**
- El **restaurante**

El restaurante a su vez incluye el **menú**, que está compuesto por los distintos **productos**.

El usuario y el restaurante solo se comunican cuando el usuario realiza un pedido y el restaurante debe confirmarlo o rechazarlo.

El usuario interactúa con el menú para ver las opciones disponibles y con el producto para seleccionar la cantidad y solicitarlo.

El restaurante se comunica con el menú para editarlo, agregando o eliminando productos. Este también puede modificar los productos ya existentes.

7. Vista de Despliegue

App para celular

Los usuarios se registran, inician sesión y acceden a las funciones del sistema a través de un teléfono celular.

Catálogo de Productos

El Sistema de Catálogo de Productos almacena todos los productos disponibles para la venta.

Base de Datos del Usuario

La base de datos de usuarios almacena la información de cada cuenta de usuario para verificar las credenciales de inicio de sesión y mantener su información segura.

8. Vista de Implementación

8.1. Vista conjunta

El software del proyecto está dividido en varias capas. La capa frontal se refiere a todos los componentes gráficos que tendrá el sistema. La lógica del proyecto, por otro lado, está

encapsulada en la capa Back-End. Por último, los datos persistentes son gestionados por la capa Base de datos.

8.2. Capas

- **Capa Front-End**

Los subsistemas de esta capa se relacionan con los archivos que conforman la GUI de las diferentes secciones del sistema, como las pantallas Registro y Checkout.

- **Capa Back-End**

Los subsistemas de esta capa se relacionan con las clases incluidas en la Vista lógica y sus interacciones.

- **Capa de Base de Datos**

El subsistema de esta capa se relaciona con las tablas de base de datos y el esquema que almacenará la información de la cuenta del usuario.

9. Vista de Datos

Los datos del usuario se almacenarán localmente en una base de datos SQL relacional. Esto incluye la dirección de correo electrónico, número de teléfono, dirección física, nombre de usuario y contraseña. Los datos del producto, como imágenes y precios, también se almacenarán localmente en un catálogo.

10. Tamaño y Rendimiento

El tamaño del software todavía no es calculado pero los factores más importantes va a ser la gerencia de memoria (datos) y el equilibrio entre datos online y offline. Datos precargados puede ser un factor clave para aumentar el rendimiento del sistema.

11. Calidad

La arquitectura asegurar que la información privado de cada cliente queda privado. También asegurarse que el sistema es estable, no permiten subprocesos múltiples y que la experiencia de usuario es satisfactoria.