



SOFT-BOYS

Propuesta

I'M HUNGRY

Presentación de proyecto

Presentado por:

Eduardo Raudel Marín Morales
Eduardo López Chacón

AGENDA

01 El equipo

02 Problemática

03 Solución

04 Análisis

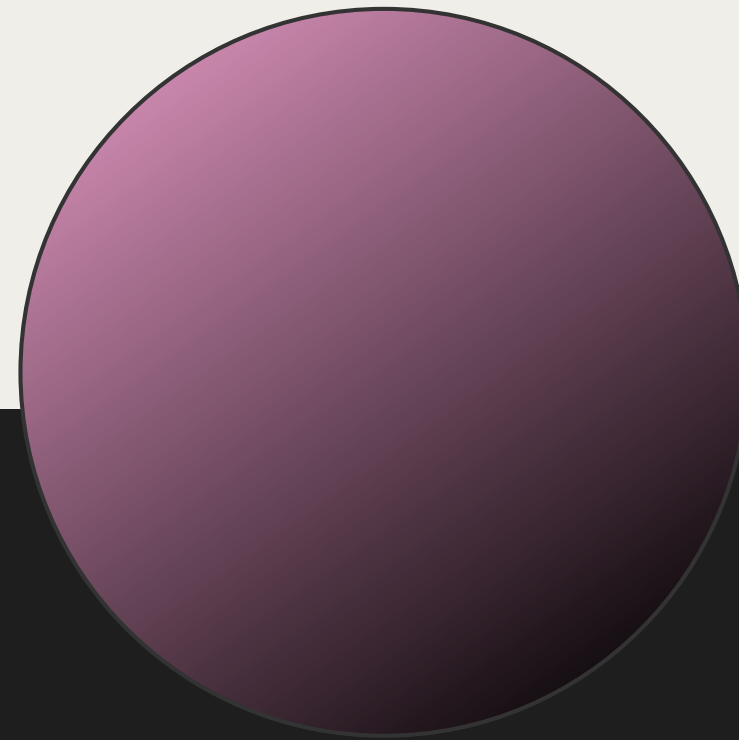
05 Diseño

06 Construcción



El equipo

SOFT-BOYS



MARÍN MORALES
EDUARDO RAUDEL



EDUARDO LÓPEZ
CHACÓN



Alcance

El sistema de software I'm Hungry, es un producto de software destinado a los estudiantes e integrantes que conforman el proceso de compra y venta de alimentos en la Facultad de Estadística e Informática perteneciente a la Universidad Veracruzana.



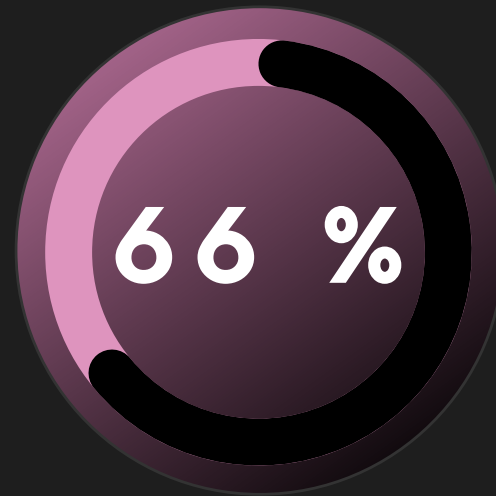
Problemática

En el bullicioso campus de la Facultad de Estadística e Informática de la Universidad

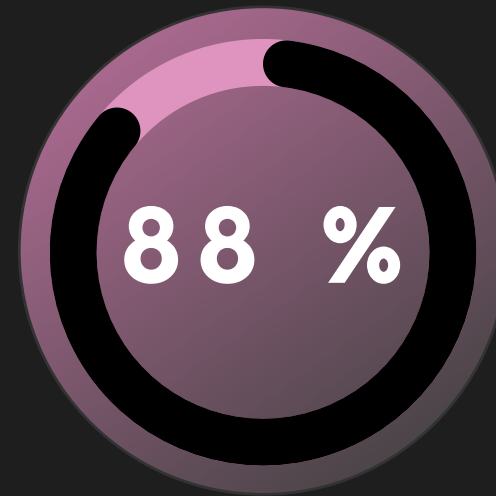
Veracruzana, los estudiantes se enfrentan a un desafío común: encontrar una comida deliciosa y conveniente durante sus agitadas jornadas académicas. La falta de opciones de comida asequibles y la dificultad para conectarse con vendedores locales ha llevado a una necesidad insatisfecha en la comunidad estudiantil. Por otra parte, algunos de los estudiantes llevan algunos dulces y alimentos a la facultad para su comercialización, de esta forma intentando cubrir una necesidad y obtener ciertos ingresos al mismo tiempo, pero les es complicado promocionarse y dar a conocer los diferentes productos y alimentos que estos ofrecen.

Para abordar esta problemática, nace la aplicación: "I'm Hungry!".

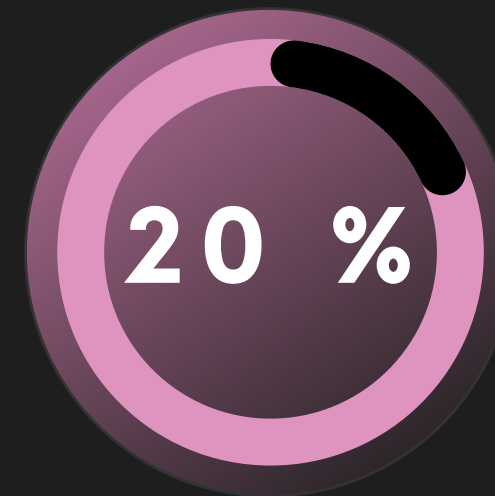
Los datos



De estudiantes no logran encontrar qué comer durante la jornada académica.



De estudiantes no logran vender todos sus productos , ya que no tienen exposición con los compradores.



De estudiantes les gustaría ofrecer algún tipo alimento en la facultad, pero no lo hacen ya que la forma en la que se hace actualmente no es muy efectiva.



Solución

"I'm Hungry!" es una aplicación diseñada específicamente para mejorar la experiencia de compra y venta de alimentos en el campus universitario de la Facultad de Estadística e Informática. Esta aplicación tiene como objetivo proporcionar a los estudiantes una plataforma eficiente y centralizada que facilite la compra y venta de alimentos durante su jornada académica.

- registrarse como vendedores o como compradores.
- Vender productos.
- Comprar Productos.
- Comunicación.



Análisis

CLASES DE USUARIO

01

ESTUDIANTE VENDEDOR

Son estudiantes que desean vender alimentos a otros estudiantes en el campus universitario

02

ESTUDIANTE COMPRADOR

Son estudiantes de la Facultad de Estadística e Informática que desean comprar alimentos

Casos de uso

01

REGISTRAR CUENTA.

02

REGISTRAR PRODUCTO.

03

CONSULTAR PEDIDO.

04

CONSULTAR ESTADÍSTICAS.

05

ENVIAR MENSAJE

06

CONSULTAR PRODUCTO.

07

REALIZAR PEDIDO.

08

EVALUAR SERVICIO.

09

CONSULTAR FAVORITOS.

10

CONSULTAR PERFIL.



Requerimientos No Funcionales

Requerimiento	Interés	Justificación
Seguridad	Confidencialidad	Se implementa el atributo de calidad Seguridad con su interés Confidencialidad. Esto debido a que los datos personales de los estudiantes que forman parte de la FEI son de carácter sensible.
Rendimiento	Comportamiento del tiempo	Esto debido a que el sistema de software implementa la arquitectura cliente servidor a través de las entidades que lo conforman y utiliza el paso de mensajes para la comunicación entre los procesos.
Fiabilidad	Disponibilidad	Se implementa el atributo de calidad Fiabilidad con su interés Disponibilidad. Esto debido a que, en la FEI, la jornada académica transcurre en un tiempo de 14 horas, comenzando desde las 7 de la mañana hasta las 9 de la noche. El sistema de software l'm Hungry requiere estar en funcionamiento durante 18 horas los días hábiles considerados de lunes a viernes



Diseño

DIAGRAMA DE CU

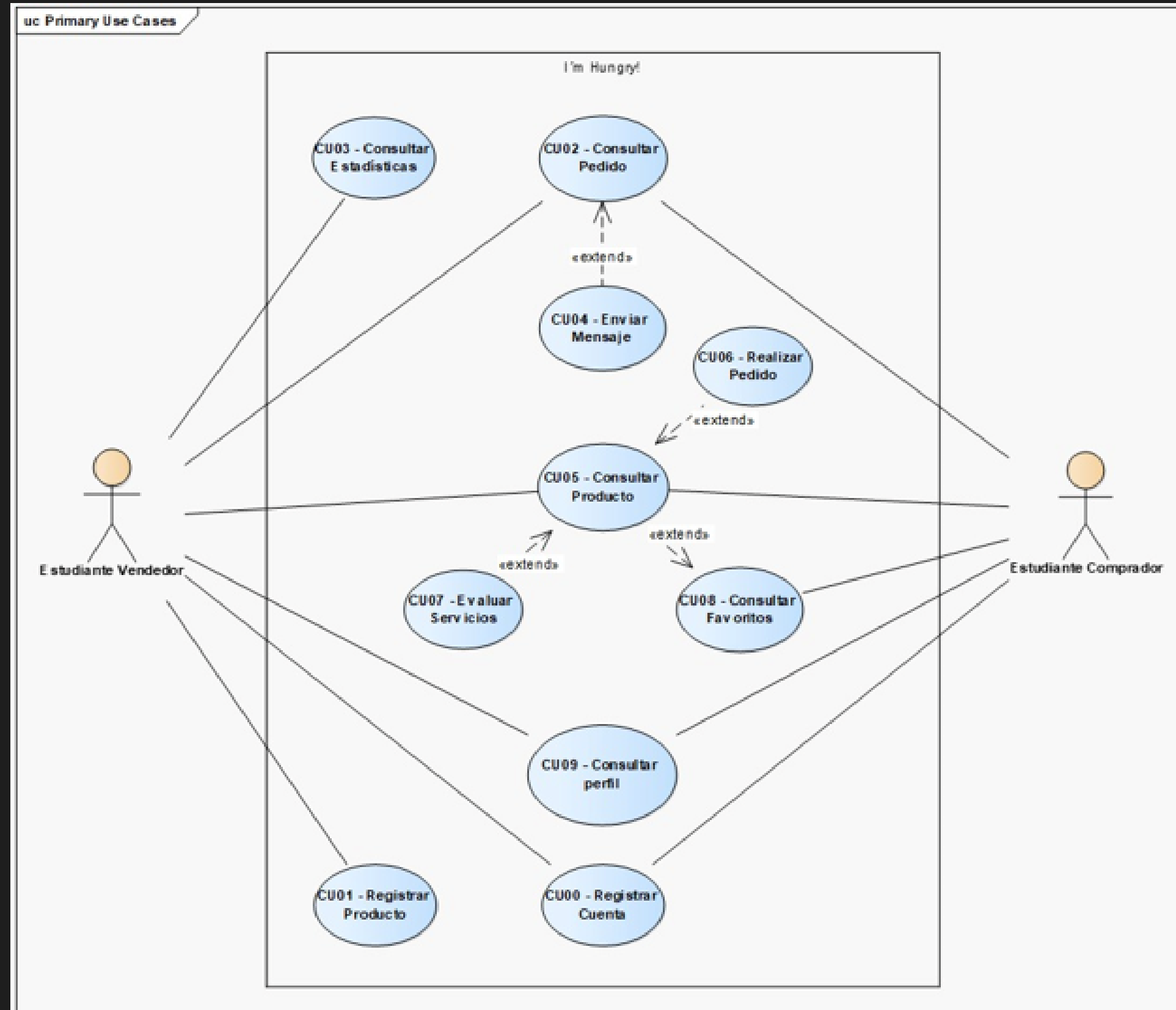
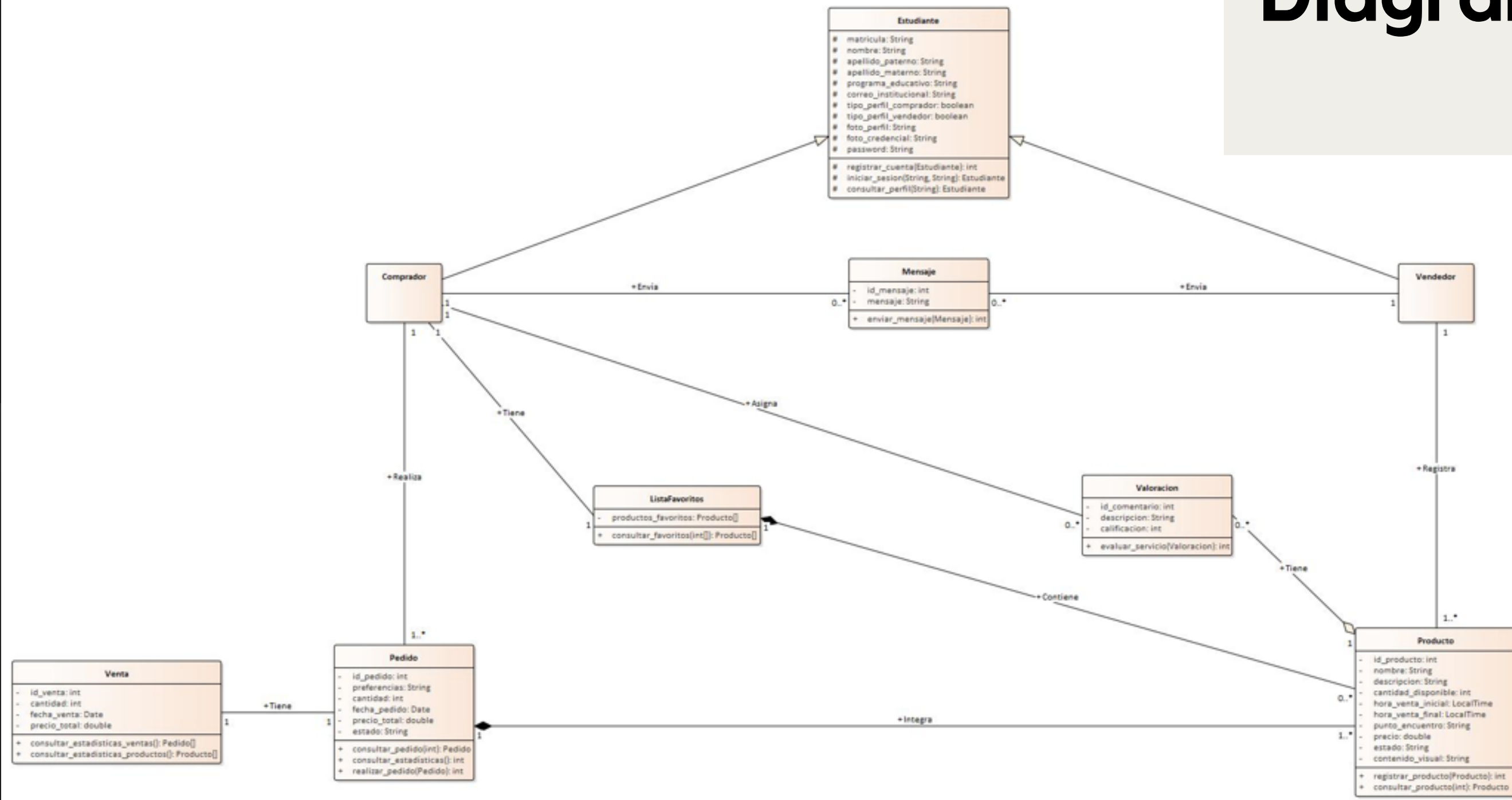




Diagrama de clases





Modelo de datos

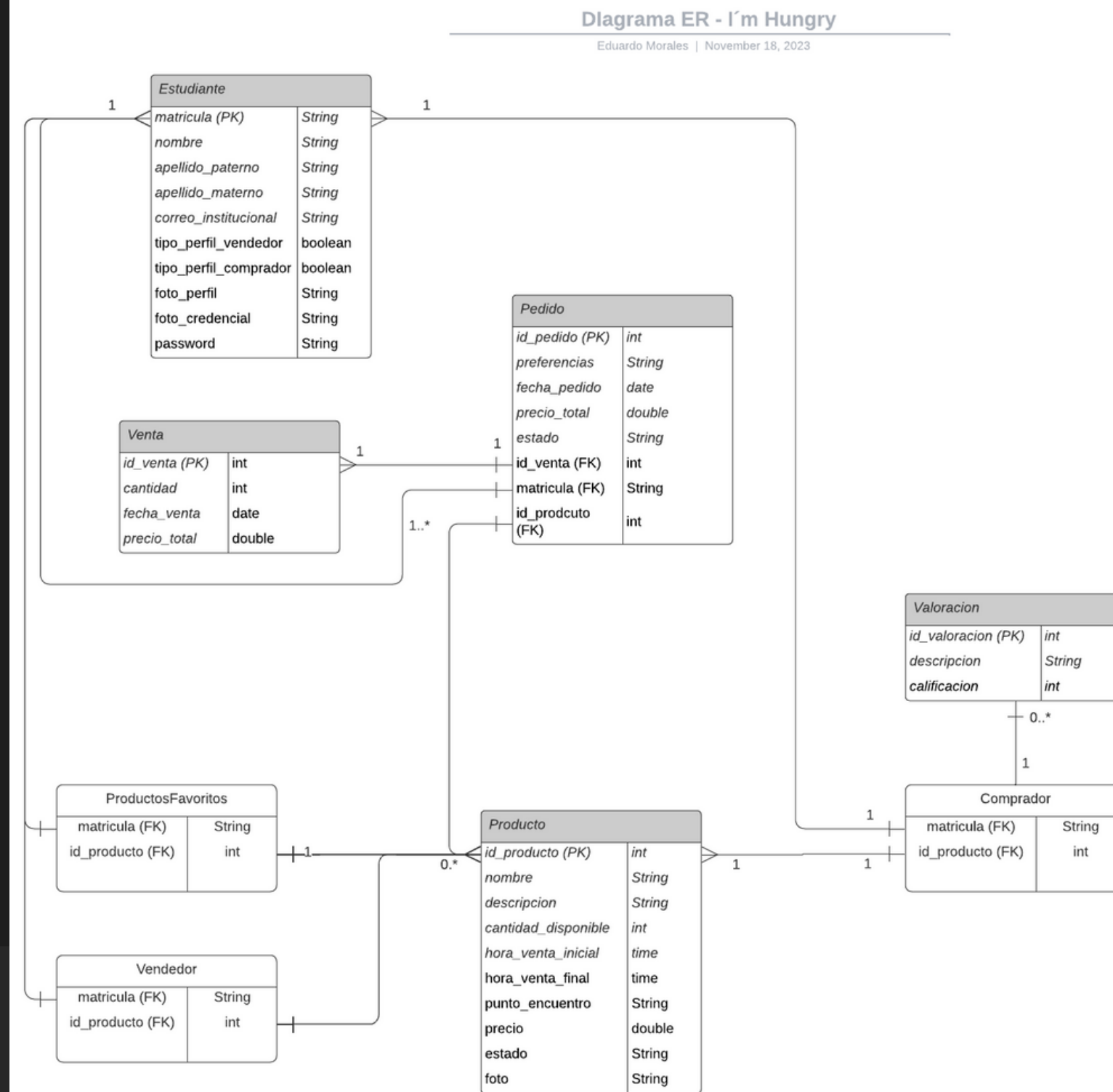




Diagrama de componentes

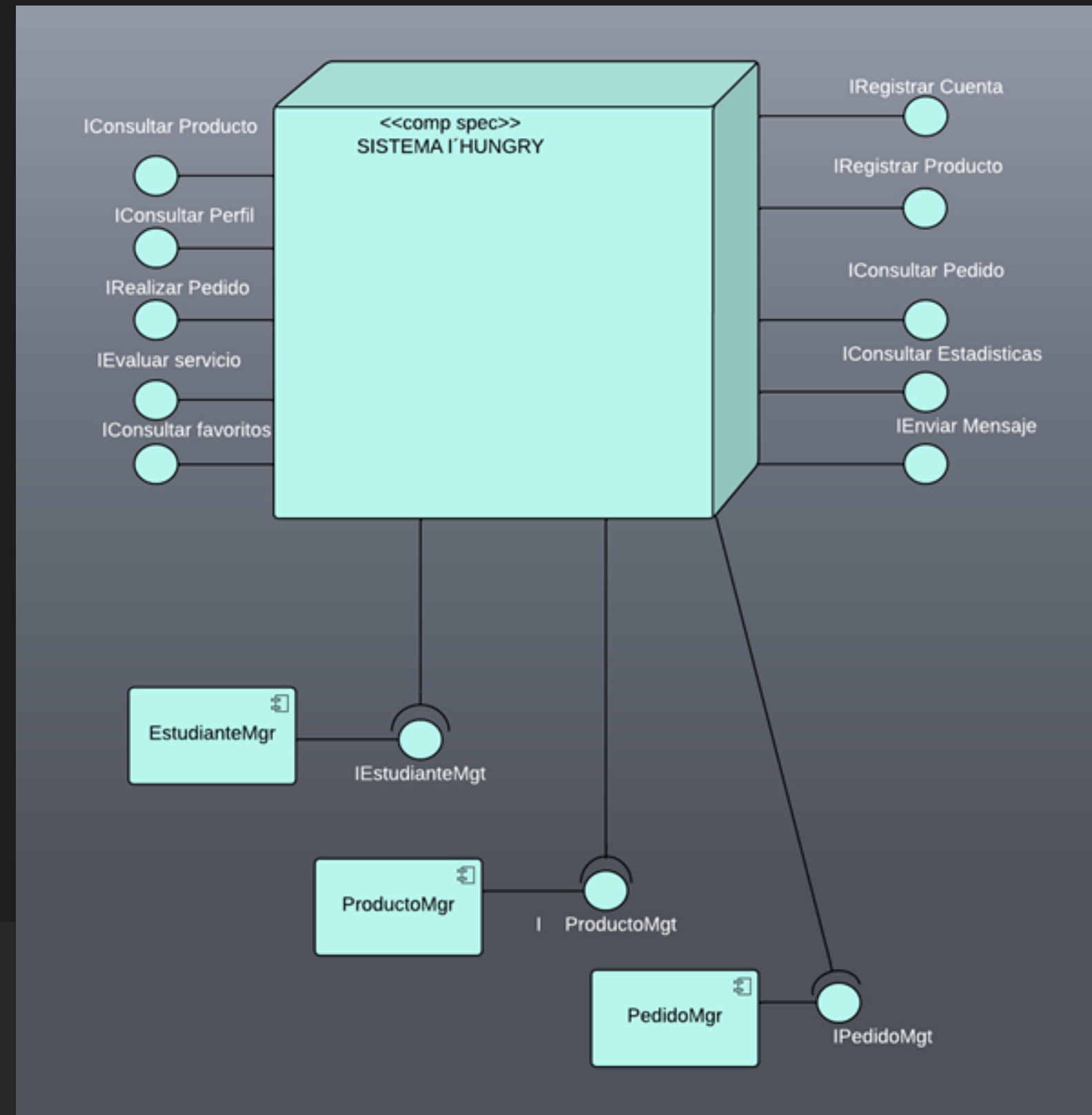
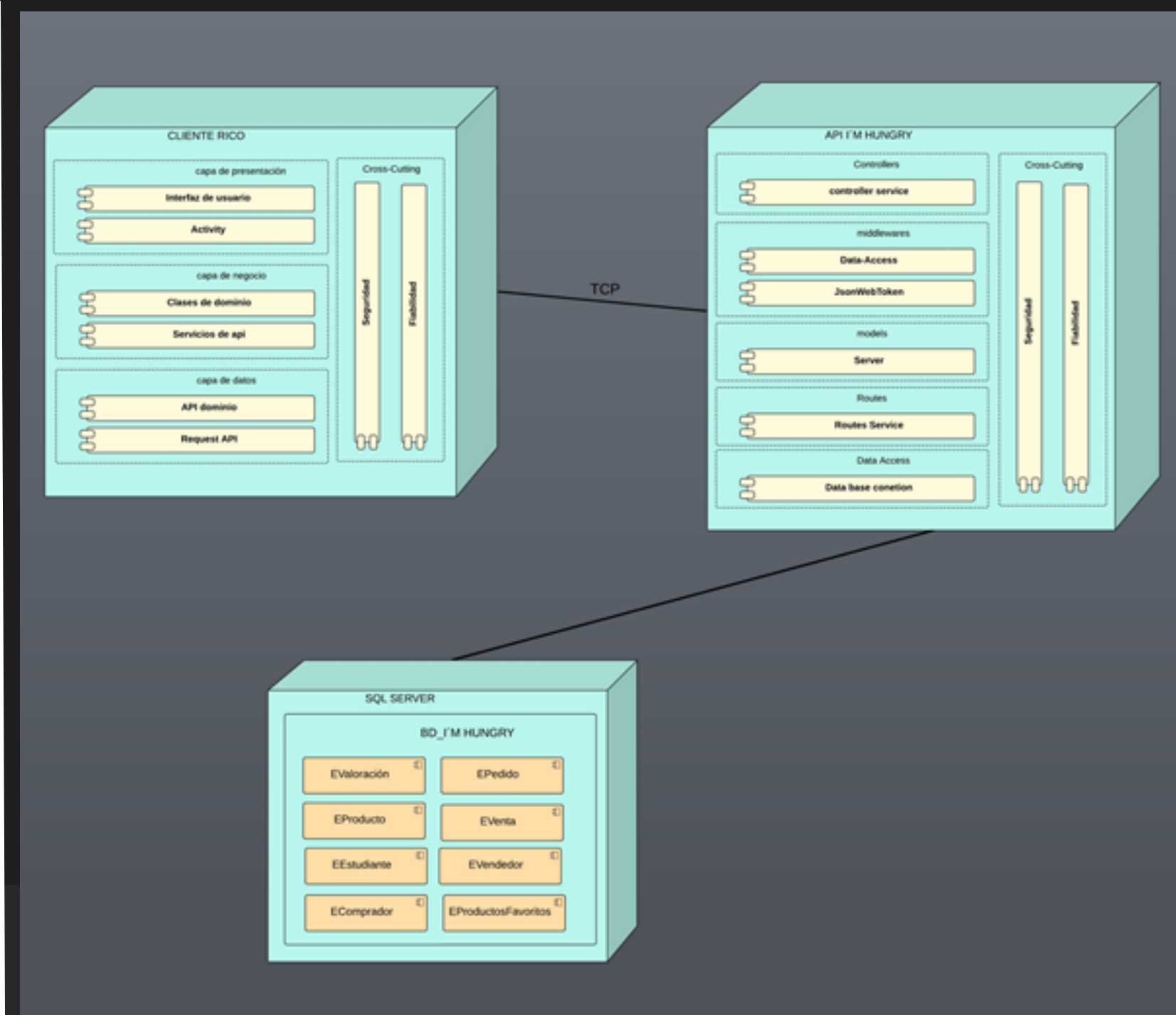
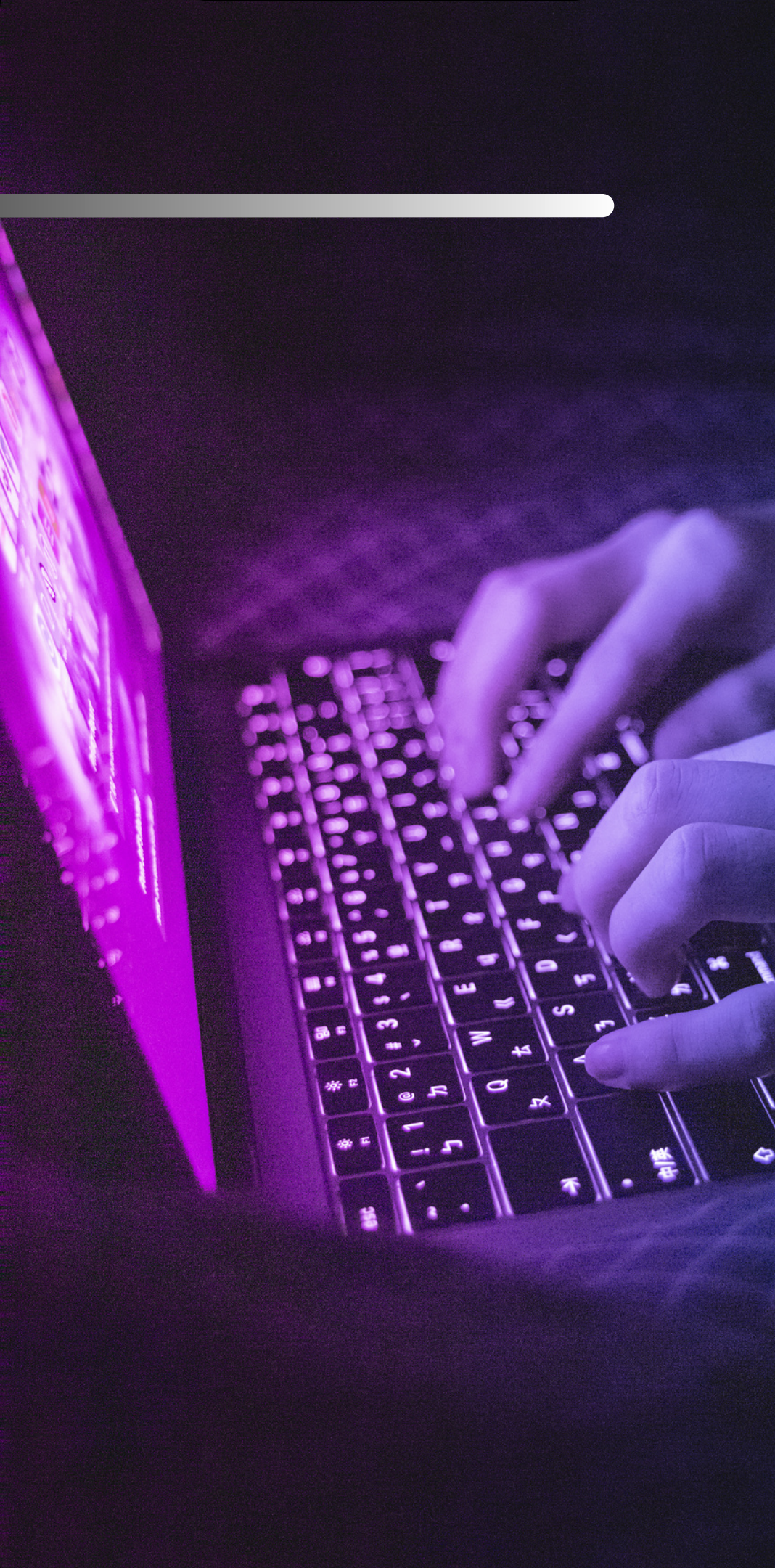


Diagrama de despliegue





Construcción



Justificación de la pila tecnologica

La elección para la API se basa en consideración de varios factores que abarcan desde la experiencia del equipo hasta la eficiencia en el desarrollo y el despliegue. se presentan algunas de las elegidas y su justificación más detallada de cada componente de la pila tecnológica:

The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays a project structure for '06-REST...'. The 'config' folder is expanded, showing subfolders like 'controllers', 'docs', 'helpers', 'middlewares', 'models', 'node_modules', 'public', 'routes', and files like '.env', 'app.js', 'Dockerfile', 'package-lock.json', and 'package.json'. The main editor area shows the 'package.json' file with the following content:

```
1 {
2   "name": "05-restserver",
3   "version": "1.0.0",
4   "description": "",
5   "main": "app.js",
6   "scripts": {
7     "start": "node app",
8     "test": "echo \"Error: no test specified\" && exit 1"
9   },
10  "keywords": [],
11  "author": "",
12  "license": "ISC",
13  "dependencies": {
14    "body-parser": "^1.20.2",
15    "cors": "^2.8.5",
16    "crypto-js": "^4.2.0",
17    "dotenv": "^16.0.3",
18    "express": "^4.18.2",
19    "jsonwebtoken": "^9.0.2",
20    "mvsal2": "^3.6.3".
```

01

NODE.JS

02

EXPRESS.JS

03

SQL SERVER

04

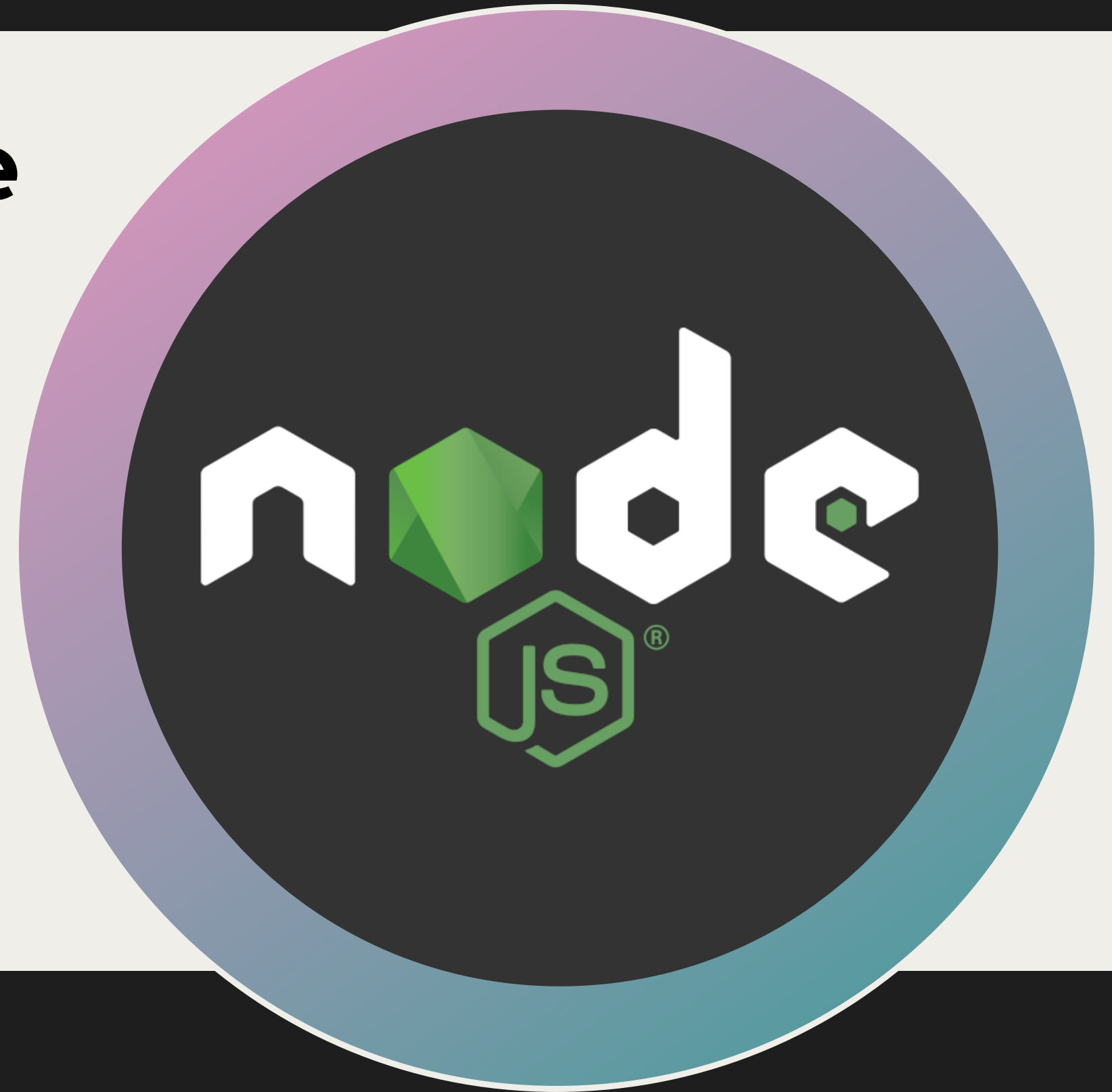
RETROFIT

03

SWAGGER.

Node.js como Lenguaje de Programación

ha sido seleccionado como el lenguaje de programación principal para la API debido a diversos factores que potencian la eficiencia y la flexibilidad en el desarrollo. La experiencia previa del equipo en Node.js es un factor crucial, ya que la familiaridad con el entorno de ejecución y la sintaxis de JavaScript agiliza el proceso de codificación.



Express.js como Framework

Express.js complementa la elección de Node.js como el framework principal para la construcción de la API. La ligereza y flexibilidad de Express.js lo convierten en una opción ideal para el desarrollo rápido y eficiente de servicios web.

La comunidad activa de desarrolladores respalda este framework, proporcionando soporte y recursos valiosos que facilitan la resolución de problemas y la mejora continua del código.

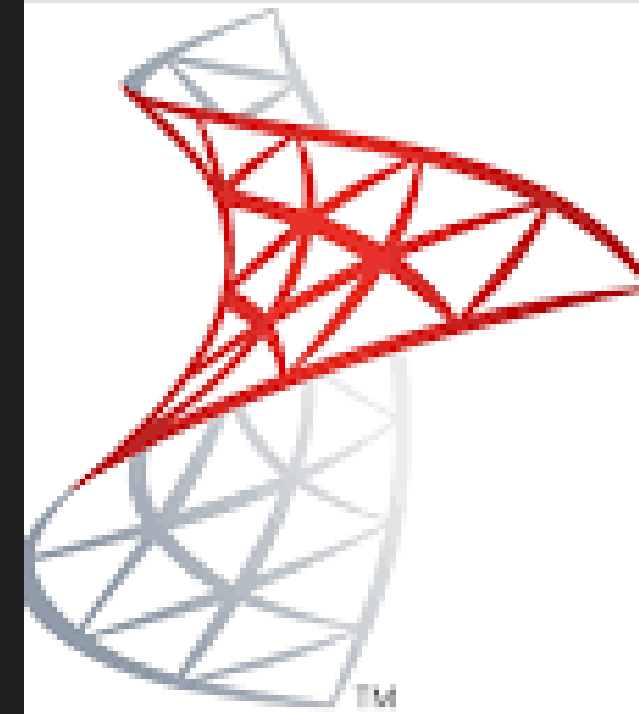
Express



SQL Server como Sistema de Gestión de Bases de Datos:

La elección de SQL Server como sistema de gestión de bases de datos se basa en la experiencia previa del equipo y en la integración eficiente de SQL Server con Node.js a través de bibliotecas como "mssql2".

La familiaridad con SQL Server simplifica las operaciones de base de datos, permitiendo una interacción fluida y segura con la capa de persistencia de la aplicación.



Microsoft®
SQL Server®
Express



RETROFIT

Retrofit ha sido seleccionado como la solución para gestionar las conexiones desde aplicaciones Android hacia la API. Esta elección se basa en la popularidad y eficiencia de Retrofit en la simplificación de las operaciones de red en entornos Android. La biblioteca proporciona una interfaz sencilla y potente para realizar solicitudes HTTP

Swagger para Documentación Automática

ha sido seleccionado como el lenguaje de programación principal para la API debido a diversos factores que potencian la eficiencia y la flexibilidad en el desarrollo. La experiencia previa del equipo en Node.js es un factor crucial, ya que la familiaridad con el entorno de ejecución y la sintaxis de JavaScript agiliza el proceso de codificación.





pila tecnológica App y Escritorio

Para la construcción del cliente de aplicación móvil se decidió utilizar el lenguaje de programación Java puesto que el equipo de desarrollo ya contaba con el conocimiento de este lenguaje, además java es uno de los lenguajes de programación más utilizados para desarrollar aplicaciones móviles, claro que existen algunos otros como Kotlin, Python, JavaScript, Swift, etc

Visual Studio, como entorno integrado de desarrollo (IDE), proporciona un amplio conjunto de herramientas y servicios que facilitan la creación, el diseño y la depuración de aplicaciones. La integración nativa con C# y el framework .NET ofrece una experiencia de desarrollo cohesiva.

01

IDE ANDROID STUDIO

02

IDE VISUAL STUDIO 2022

**MUCHAS
GRACIAS**