

Proyecto final de curso DESARROLLO DEL BACKEND **AAT-RA2**

Integrantes:

|  |  |
| --- | --- |
| Marvin Mazariegos | **2024-010309** |
| Mario Castillo | **2021-071816** |
| Josué Abraham Ramírez | **2020-068382** |
| Juan de Dios Sandoval Salguero | **2020-009969** |
| Lilian Andrea Pirir Coz | **2024-014714** |
| Moisés Alvarado | **2019-022934** |

## Octubre 2024

Índice

[Introducción 1](#_bookmark0)

[Desarrollo 2](#_bookmark1)

[Base de datos 2](#_bookmark2)

[Bases de datos relacionales (SQL): 2](#_bookmark3)

[Bases de datos NoSQL 2](#_bookmark4)

[Base de Datos Relacional (MySQL) 4](#_bookmark5)

[Tablas 4](#_bookmark6)

[Procedimientos Almacenados 4](#_bookmark7)

[Scripts SQL 4](#_bookmark8)

[Scripts DDL (Data Definition Language) 4](#_bookmark9)

[Scripts DML (Data Manipulation Language) 5](#_bookmark10)

[Base de Datos NoSQL (MongoDB) 6](#_bookmark11)

[Scripts NoSQL 6](#_bookmark12)

[Mockups del proyecto 7](#_bookmark13)

[Carpeta de Github 10](#_bookmark14)

# Introducción

El desarrollo de aplicaciones web ha evolucionado hacia un enfoque cada vez más integral, combinando tecnologías Frontend y Backend para ofrecer soluciones eficientes y adaptadas a las necesidades del negocio. Este proyecto se centrará en la creación de una aplicación web destinada a la gestión de clientes, productos y ventas, utilizando una combinación de bases de datos relacionales y NoSQL.

Por un lado, se implementará una base de datos relacional para gestionar los datos críticos del negocio, como ventas, clientes y productos, garantizando consistencia y estructura en la información. Por otro lado, se utilizará una base de datos NoSQL para manejar de manera flexible los comentarios de los clientes y productos, proporcionando escalabilidad y una estructura más dinámica para los datos no estructurados.

Adicionalmente, el desarrollo de esta aplicación contempla un enfoque visual detallado, con la creación de mockups y wireframes que facilitarán la definición clara del diseño de la interfaz de usuario. Para asegurar la calidad del sistema, se llevarán a cabo pruebas exhaustivas durante todo el ciclo de vida del desarrollo, garantizando así un producto final robusto y acorde a los requisitos funcionales y no funcionales del proyecto.

# Desarrollo

## Base de datos

Una base de datos es un conjunto organizado de datos que se almacena y gestiona de manera electrónica. Está diseñada para facilitar el acceso, la manipulación y la actualización de la información de forma eficiente.

Las bases de datos son fundamentales en la mayoría de las aplicaciones y sistemas de información, ya que permiten organizar grandes volúmenes de datos de manera estructurada.

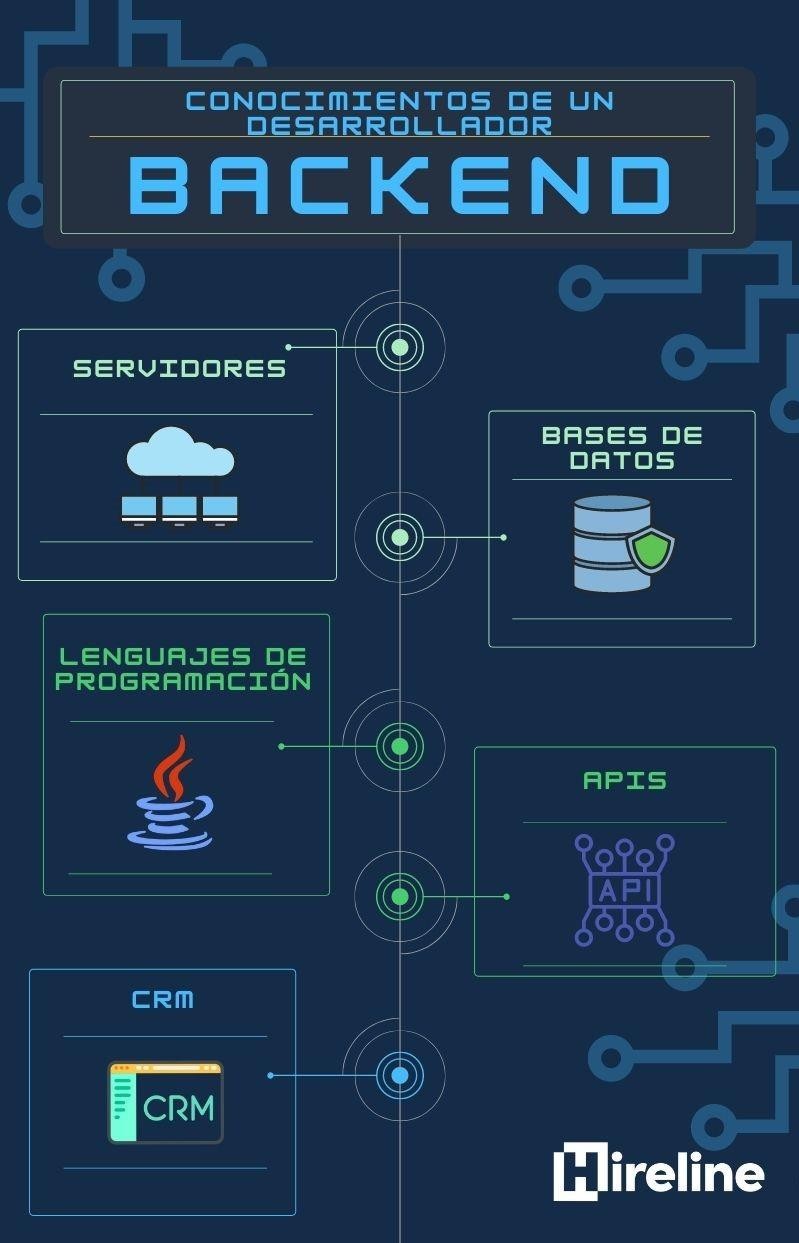
Existen varios tipos de bases de datos, y entre las más comunes están:

## Bases de datos relacionales (SQL)

1. Organizan los datos en tablas (filas y columnas) que se relacionan entre sí mediante claves. Ejemplos: MySQL, PostgreSQL, Oracle.
2. Son ideales para gestionar datos estructurados y bien organizados, como listas de productos, clientes o transacciones.

## Bases de datos NoSQL

1. Estas bases de datos están diseñadas para manejar grandes volúmenes de datos no estructurados o semi-estructurados, como comentarios, imágenes, documentos JSON, etc. Ejemplos: MongoDB, Cassandra, Redis.
2. Son más flexibles que las bases de datos relacionales y pueden manejar diferentes tipos de datos sin requerir una estructura fija.



## Base de Datos Relacional (MySQL)

1. Modelo Entidad-Relación (ERD):

* Tablas: Clientes, Productos, Ventas.
* Relaciones:
  + Ventas tiene relaciones con Clientes (cada venta tiene un cliente asociado).
  + Ventas tiene relaciones con Productos (cada venta incluye uno o más productos).
* Triggers:
  + Trigger para actualizar el inventario automáticamente al realizar una venta.

## Tablas

\*\*Clientes\*\*: `id`, `nombre`, `correo`, `teléfono`.

\*\*Productos\*\*: `id`, `nombre`, `precio`, `stock`.

\*\*Ventas\*\*: `id`, `cliente\_id`, `producto\_id`, `cantidad`, `fecha\_venta`.

## Procedimientos Almacenados

Procedimiento para generar un reporte de ventas con filtros por cliente, producto y rango de fechas.

# Scripts SQL

## Scripts DDL (Data Definition Language)

CREATE TABLE Clientes (

id INT PRIMARY KEY AUTO\_INCREMENT, nombre VARCHAR(100) NOT NULL,

correo VARCHAR(100), telefono VARCHAR(15)

);

CREATE TABLE Productos (

id INT PRIMARY KEY AUTO\_INCREMENT, nombre VARCHAR(100) NOT NULL,

precio DECIMAL(10, 2) NOT NULL,

stock INT NOT NULL

);

CREATE TABLE Ventas (

id INT PRIMARY KEY AUTO\_INCREMENT,

cliente\_id INT, producto\_id INT,

cantidad INT NOT NULL, fecha\_venta DATETIME NOT NULL,

FOREIGN KEY (cliente\_id) REFERENCES Clientes(id), FOREIGN KEY (producto\_id) REFERENCES Productos(id)

);

DELIMITER $$

CREATE TRIGGER actualizar\_inventario AFTER INSERT ON Ventas

FOR EACH ROW BEGIN

UPDATE Productos

SET stock = stock - NEW.cantidad WHERE id = NEW.producto\_id;

END$$ DELIMITER ;

## Scripts DML (Data Manipulation Language)

-- Insertando datos en Clientes

INSERT INTO Clientes (nombre, correo, telefono) VALUES ('Juan Perez', ['juan@example.com](mailto:juan@example.com)', '123456789');

-- Insertando datos en Productos

INSERT INTO Productos (nombre, precio, stock) VALUES ('Laptop', 1500.00, 10);

-- Insertando una venta

INSERT INTO Ventas (cliente\_id, producto\_id, cantidad, fecha\_venta) VALUES (1, 1, 2, NOW());

## Base de Datos NoSQL (MongoDB)

1. Estructura de Documentos:

* \*\*Comentarios\*\*: Los usuarios pueden agregar comentarios a los productos.
  + Campos: `producto\_id`, `cliente\_id`, `comentario`, `fecha`.
* \*\*Productos Relacionados\*\*: Recomendaciones basadas en búsquedas y comentarios.
  + Campos: `producto\_id`, `productos\_relacionados` (lista de productos recomendados).

## Scripts NoSQL

// Insertando un comentario en MongoDB db.comentarios.insertOne({

producto\_id: 1,

cliente\_id: 1,

comentario: "Excelente producto, muy satisfecho!", fecha: new Date()

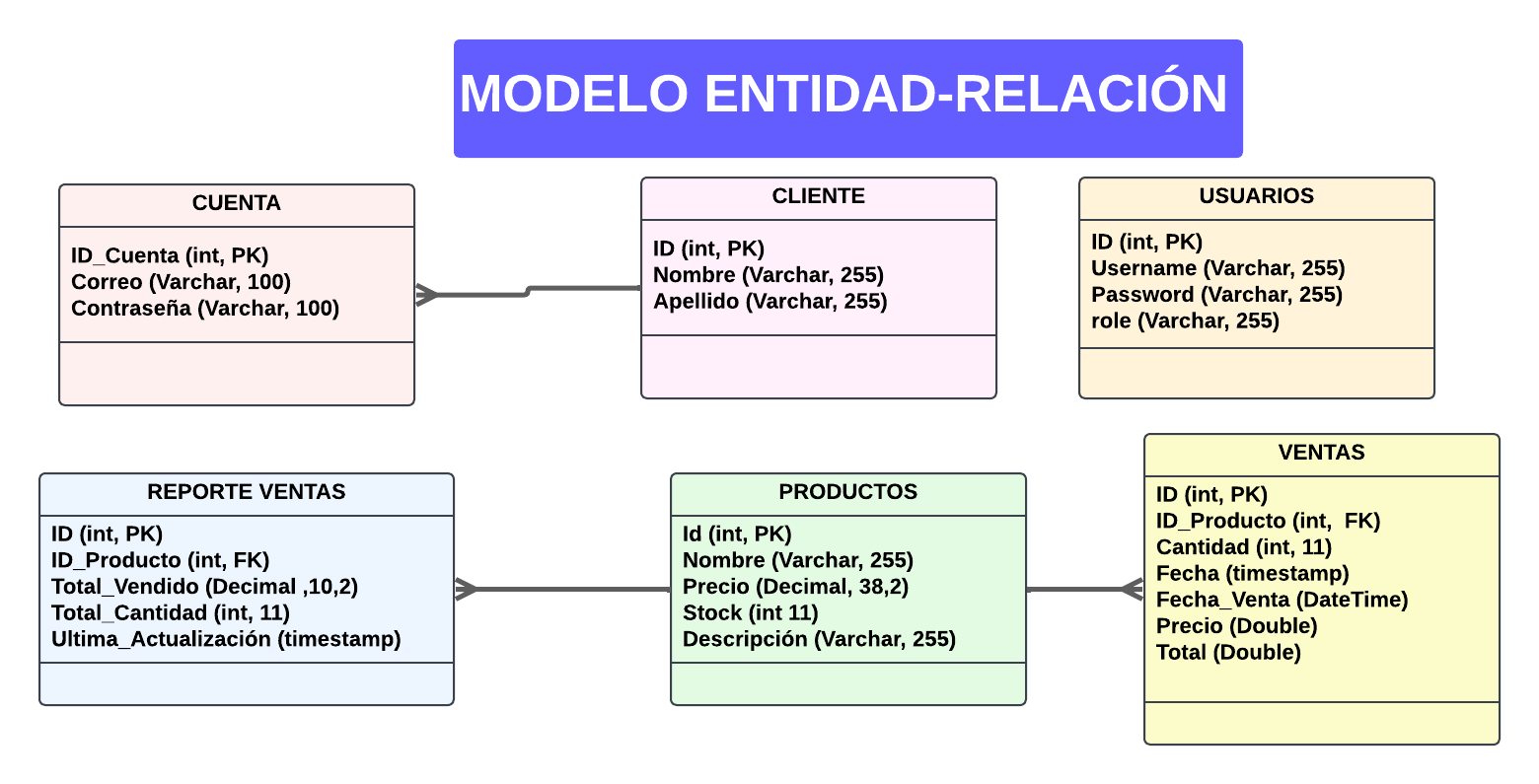
});

// Insertando productos relacionados db.productosRelacionados.insertOne({

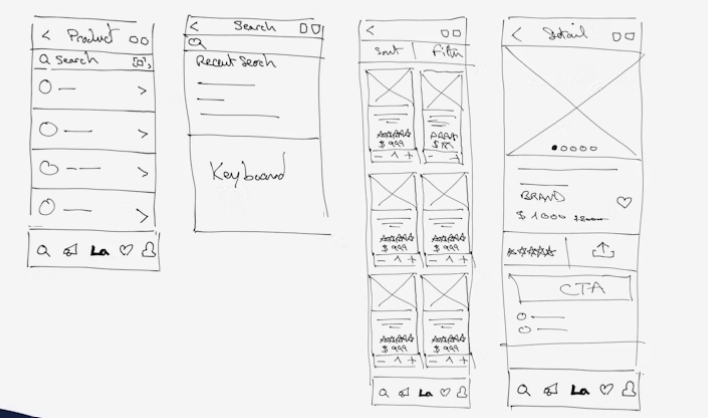
producto\_id: 1,

productos\_relacionados: [2, 3, 4]

});



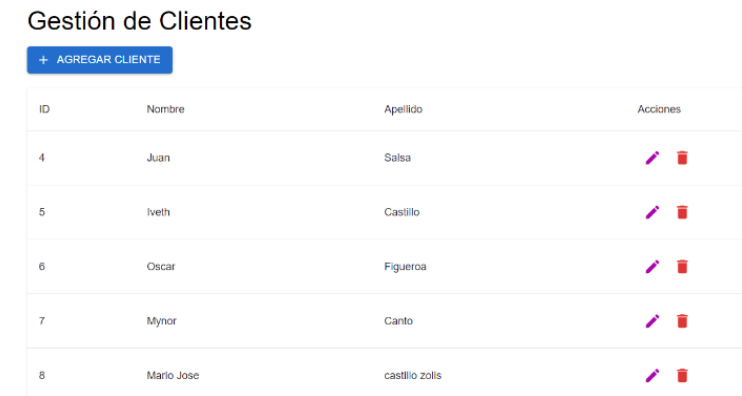
# Wireframes

****

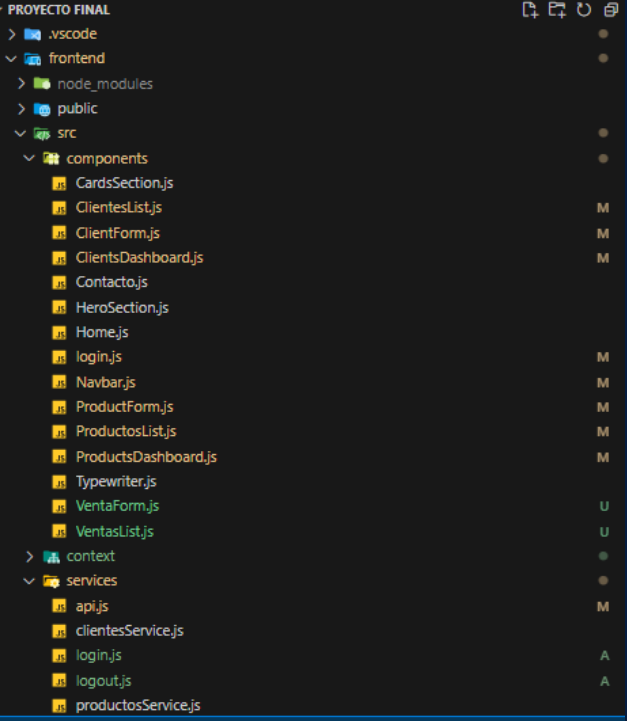
# Mockups del proyecto

****

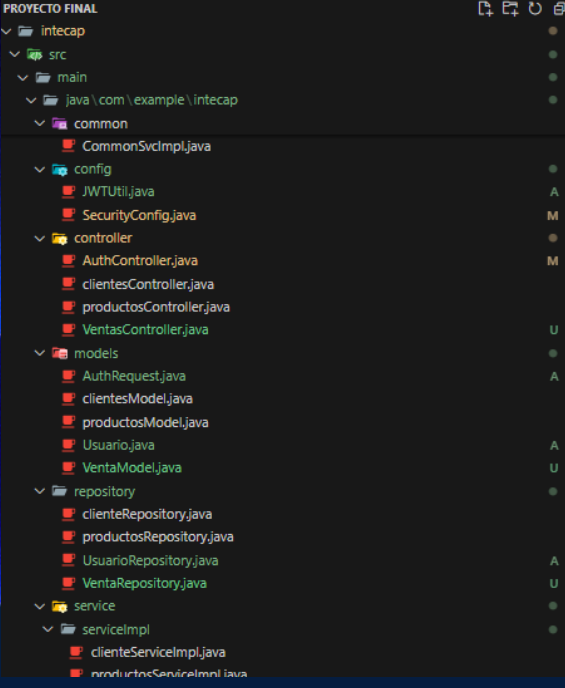
****

****

# ESTRUCTURA DEL FRONT END

****

# ESTRUCTURA DEL BACK END

****

**Carpeta de Github:** [**Proyecto Final**](https://github.com/mmazariegos-2021338/Proyecto-Final-Intecap.git)

Presentación: [ENLACE](https://www.canva.com/design/DAGVAcI-WgY/R1sor45NTVSqnKkHKmplvg/view?utm_content=DAGVAcI-WgY&utm_campaign=designshare&utm_medium=link&utm_source=editor)