



# Tecnologías del Sistema POS + Inventario (Analogía: Construcción de una Casa)

---

## Frontend

### 1. React

- **¿Qué es?** Una biblioteca de JavaScript para construir interfaces de usuario basadas en componentes.
- **Analogía:** React es como las paredes modulares de una casa prefabricada. Cada habitación (componente) puede construirse por separado y luego ensamblarse para formar el conjunto de la casa.
- **Rol en el sistema:** Es el núcleo del frontend, permitiendo crear una interfaz modular donde cada elemento es un componente reutilizable.

### 2. React Router DOM

- **¿Qué es?** Una biblioteca para manejar la navegación en aplicaciones React.
- **Analogía:** Es como las puertas interiores de la casa, que te permiten moverte de una habitación a otra sin salir de la casa.
- **Rol en el sistema:** Gestiona la navegación entre páginas sin recargar la página completa.

### 3. Tailwind CSS

- **¿Qué es?** Un framework CSS utilitario.
- **Analogía:** Es como tener un catálogo de acabados (pintura, pisos, molduras) que puedes aplicar directamente sin fabricar cada detalle desde cero.
- **Rol en el sistema:** Proporciona estilos consistentes y responsivos con clases predefinidas.

### 4. Lucide React

- **¿Qué es?** Una biblioteca de iconos para React.
- **Analogía:** Son como los símbolos en los interruptores de luz, grifos o señalización de la casa que hacen más intuitivo su uso.
- **Rol en el sistema:** Proporciona iconos consistentes para la interfaz de usuario.

### 5. useState (Hook de React)

- **¿Qué es?** Un hook que permite añadir estado a componentes.
- **Analogía:** Es como tener un tablero dentro de cada habitación donde se anotan cosas importantes (como luces encendidas, temperatura, ocupación).
- **Rol en el sistema:** Gestiona el estado local como el de formularios o componentes UI.

### 6. useEffect (Hook de React)

- **¿Qué es?** Un hook para ejecutar código en respuesta a cambios.
- **Analogía:** Es como un sistema automatizado que enciende las luces cuando entras en una habitación o activa la calefacción cuando baja la temperatura.

- **Rol en el sistema:** Ejecuta efectos secundarios como cargar datos al montar componentes.
- 

## Backend

### 7. Node.js

- **¿Qué es?** Un entorno de ejecución para JavaScript en el servidor.
- **Analogía:** Es como el sistema eléctrico principal de la casa, que distribuye la energía a todas las habitaciones (componentes del backend).
- **Rol en el sistema:** Plataforma base del backend para ejecutar código y manejar peticiones.

### 8. Express

- **¿Qué es?** Un framework para Node.js.
- **Analogía:** Es como el sistema de tuberías que distribuye el agua: organiza las rutas por donde fluye la información.
- **Rol en el sistema:** Gestiona las rutas HTTP, middleware y controladores.

### 9. Prisma

- **¿Qué es?** Un ORM moderno.
- **Analogía:** Es como los planos de la casa que definen cómo se conectan los espacios y qué hay en cada uno.
- **Rol en el sistema:** Facilita la interacción con la base de datos mediante una API segura.

## 10. JWT

- **¿Qué es?** Un estándar para crear tokens de autenticación.
- **Analogía:** Es como una llave maestra electrónica que te permite entrar en diferentes partes de la casa si tienes permiso.
- **Rol en el sistema:** Gestiona la autenticación y autorización mediante tokens.

## 11. bcryptjs

- **¿Qué es?** Una biblioteca para el hashing de contraseñas.
- **Analogía:** Es como poner cerraduras personalizadas y reforzadas en las puertas que solo se abren con llaves cifradas.
- **Rol en el sistema:** Protege contraseñas mediante hashing seguro.

## 12. CORS

- **¿Qué es?** Un mecanismo de control de acceso entre dominios.
- **Analogía:** Es como tener portones de acceso en la entrada de la casa que solo se abren a visitantes permitidos.
- **Rol en el sistema:** Permite que el frontend y backend se comuniquen de forma segura.

## 13. dotenv

- **¿Qué es?** Un módulo que carga variables de entorno.
  - **Analogía:** Es como un cuarto secreto donde se guardan las llaves y documentos importantes de la casa.
  - **Rol en el sistema:** Gestiona configuraciones sensibles como contraseñas y URLs.
-

# Base de Datos

## 14. PostgreSQL

- **¿Qué es?** Un sistema de gestión de bases de datos relacional.
- **Analogía:** Es como el sótano de la casa donde se almacenan cajas bien organizadas con toda la información que necesitas encontrar fácilmente.
- **Rol en el sistema:** Almacena todos los datos persistentes del sistema de forma estructurada.

## 15. Neon

- **¿Qué es?** Un proveedor PostgreSQL serverless.
  - **Analogía:** Es como alquilar un almacén externo seguro y automatizado en lugar de construir tu propio sótano.
  - **Rol en el sistema:** Proporciona una base de datos en la nube sin necesidad de gestión propia.
- 

# Herramientas de Desarrollo

## 16. Nodemon

- **¿Qué es?** Una utilidad que reinicia la aplicación Node.js al detectar cambios.
- **Analogía:** Es como tener un capataz que supervisa las obras y activa automáticamente los cambios en la casa sin intervención manual.
- **Rol en el sistema:** Acelera el desarrollo reiniciando el servidor automáticamente.

## 17. ESLint

- **¿Qué es?** Una herramienta de análisis estático de código.
- **Analogía:** Es como un inspector de construcción que revisa constantemente que todo esté hecho con calidad y conforme al plano.
- **Rol en el sistema:** Mantiene la calidad del código, detectando errores y malas prácticas.