



CORTE 3 PROYECTO

Ingeniería de Software 1





PROBLEMÁTICA

En la actualidad, muchos negocios dependen de software obsoleto con interfaces poco amigables, carencia de automatización en procesos críticos y limitaciones para sincronizar datos en tiempo real.



DIAGRAMA DE CLASES

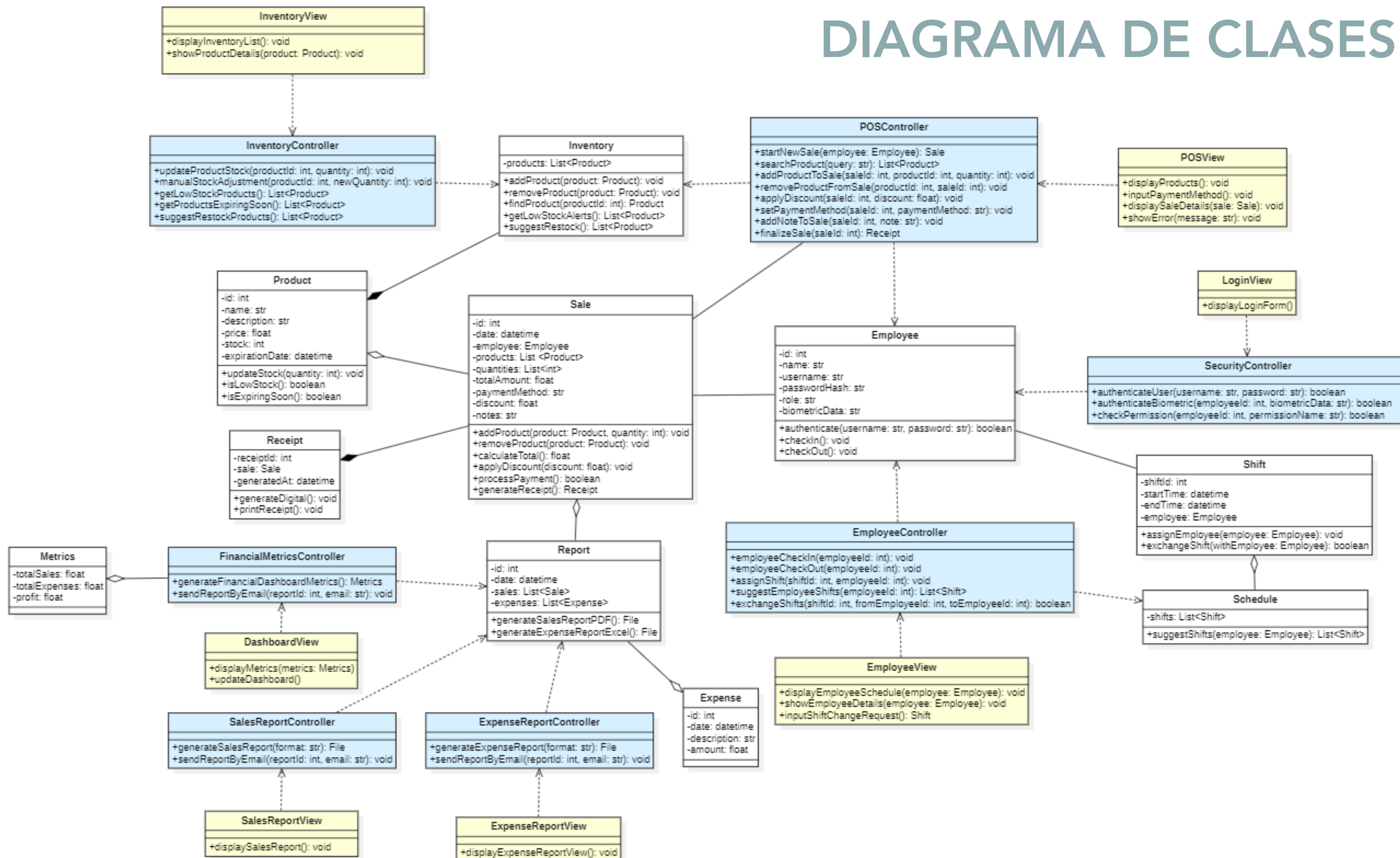


DIAGRAMA DE CLASES PERSISTENTES

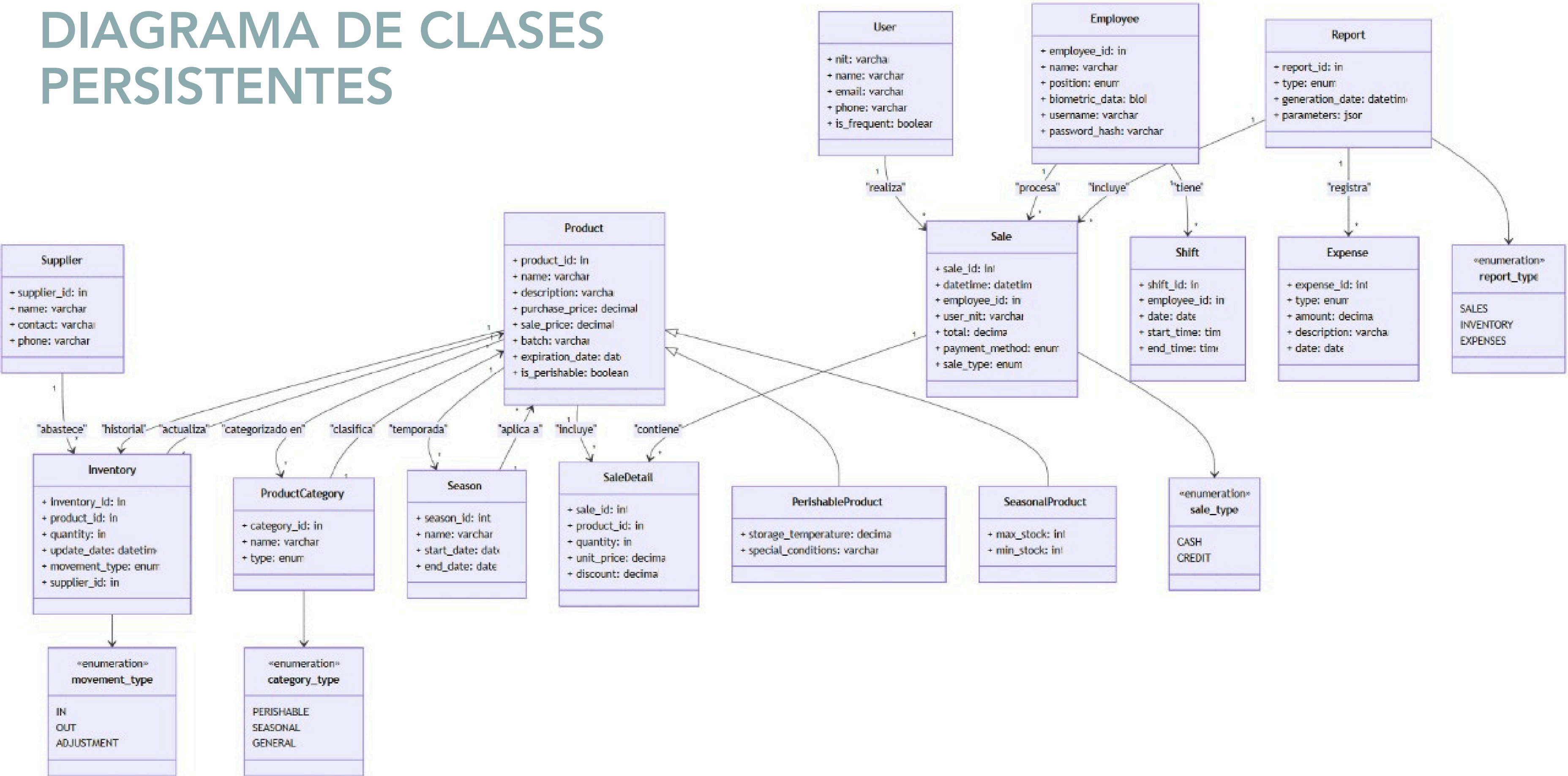
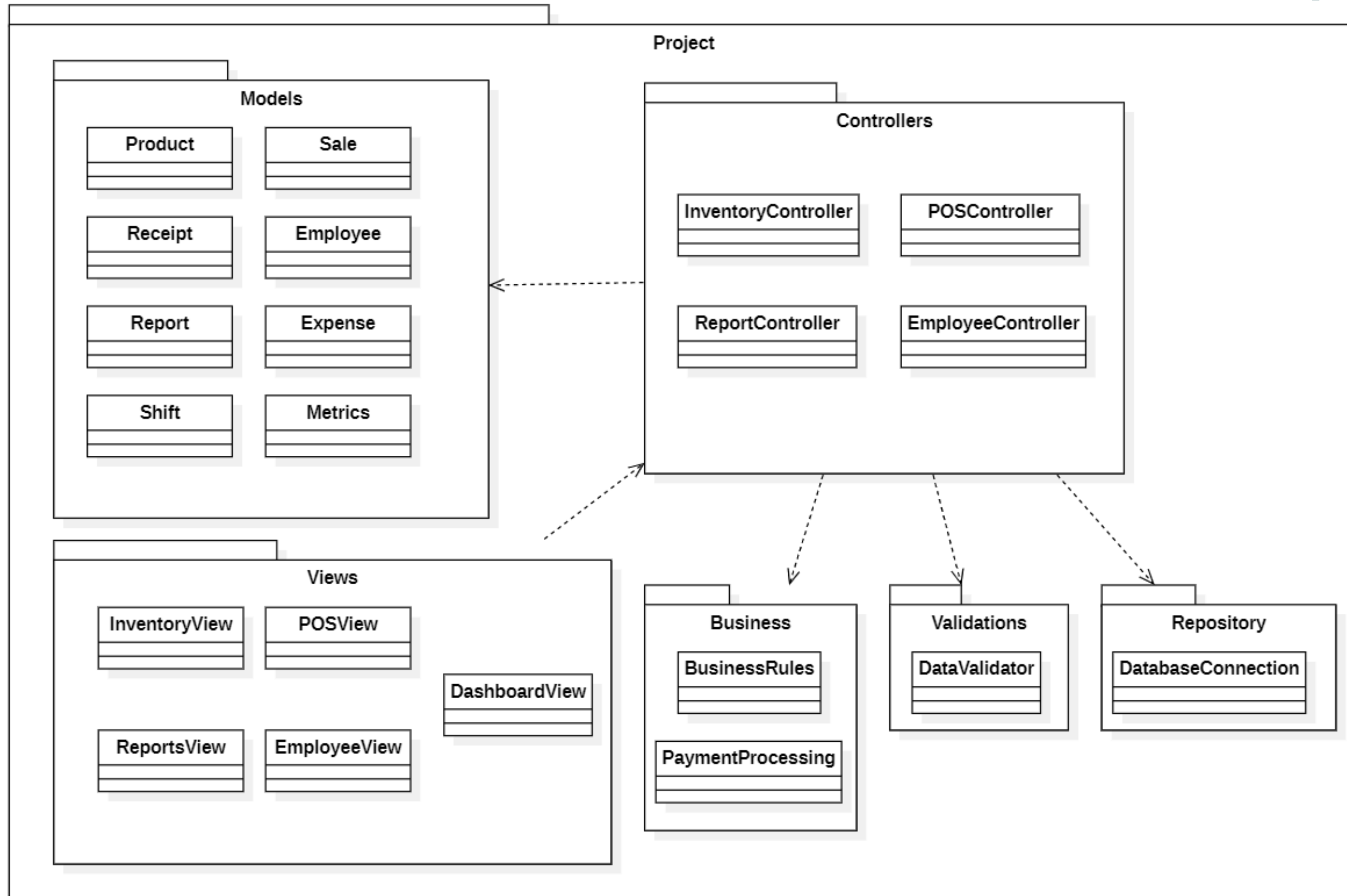


DIAGRAMA DE PAQUETES



TECNOLOGÍAS PARA FRONTEND

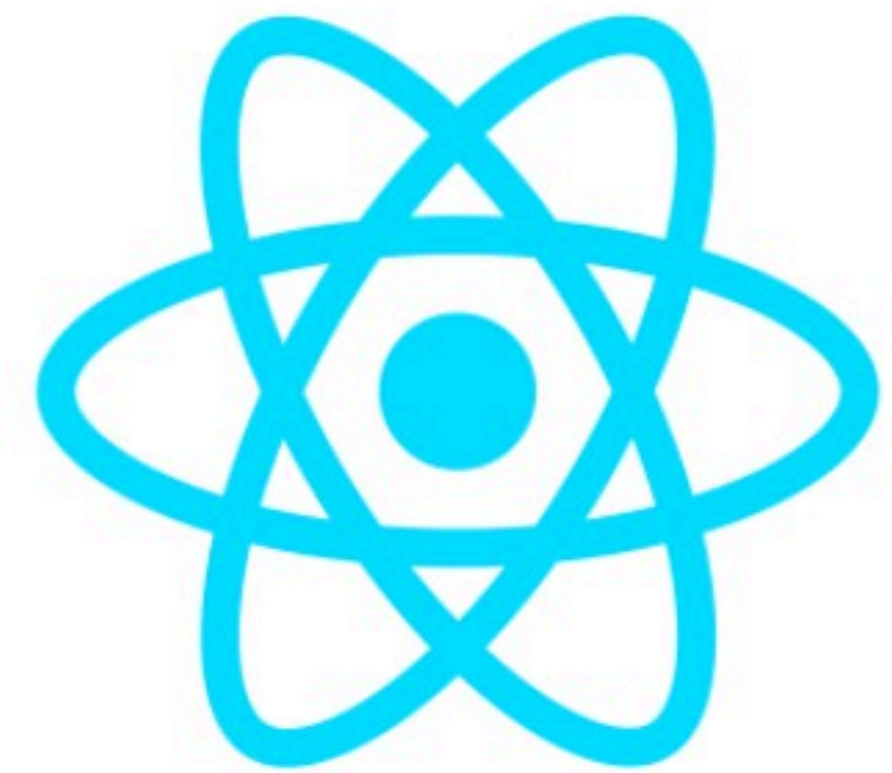
Cuadro Comparativo: ventajas y desventajas



Tecnología	Ventajas	Desventajas
React.js (JavaScript)	<ul style="list-style-type: none">• Amplio ecosistema de librerías y herramientas• Flexible y adaptable a diferentes arquitecturas• Soporte de renderizado del lado del servidor (SSR).	<ul style="list-style-type: none">• JSX y manejo de estados que requieren un mayor dominio.• Puede ser complejo para principiantes.• Actualizaciones pueden generar problemas de compatibilidad
Vue.js (JavaScript)	<ul style="list-style-type: none">• Fácil de aprender.• Sintaxis más simple que React.• Reactividad incorporada facilita la gestión de estados simples	<ul style="list-style-type: none">• Menos opciones de librerías y herramientas en comparación con React• Puede ser menos adecuado para aplicaciones muy grandes y complejas.
Angular (TypeScript)	<ul style="list-style-type: none">• Arquitectura bien definida y escalable• Soporte oficial de Google.• Buen manejo de grandes aplicaciones.	<ul style="list-style-type: none">• Mayor rigidez en la estructura puede limitar la flexibilidad• Tamaño del bundle puede ser mayor en comparación con otros frameworks• Proceso de actualización puede ser complejo.

React JS

- Se pueden crear interfaces de usuario interactivas y de alto rendimiento.
- Centrada en componentes reutilizables, facilitando la organización y escalabilidad del código.
- Su gran comunidad y recursos facilitan la resolución de problemas.

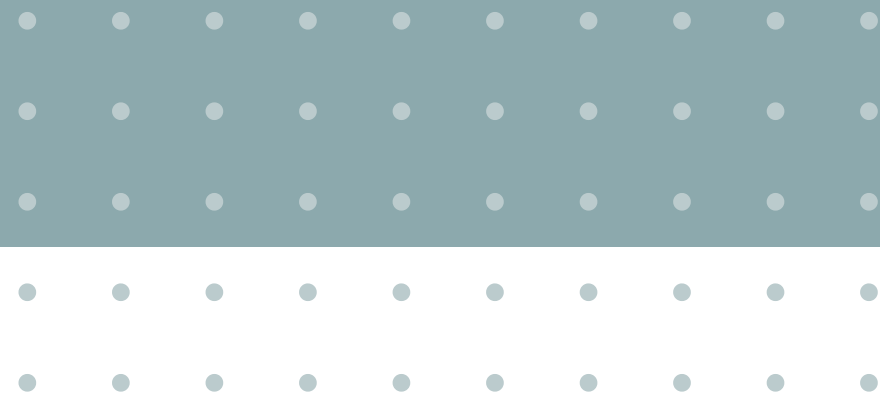


React JS

**TECNOLOGÍA
SELECCIONADA**

TECNOLOGÍAS PARA BACKEND

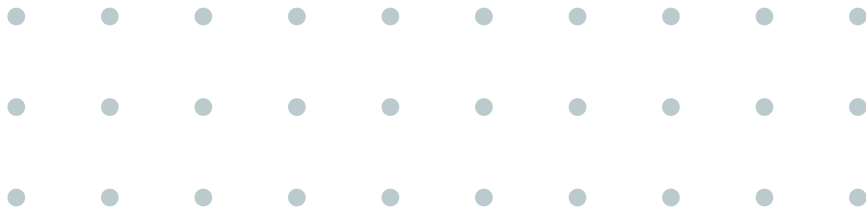
Cuadro Comparativo: ventajas y desventajas



Tecnología	Lenguaje	Ventajas	Desventajas
Spring Boot	Java	<ul style="list-style-type: none">• Gran ecosistema y comunidad amplia.• Soporte nativo para multi-tenant con Hibernate.• Seguridad robusta con Spring Security.• Escalabilidad horizontal con microservicios.	<ul style="list-style-type: none">• Curva de aprendizaje pronunciada.• Configuración compleja para aplicaciones grandes.• Más consumo de recursos que opciones más livianas.
Django + Django Tenants	Python	<ul style="list-style-type: none">• Rápido desarrollo con enfoque DRY.• Módulo "Django Tenants" para manejar bases de datos compartidas o separadas.• Admin integrado para gestión de inquilinos.	<ul style="list-style-type: none">• Menos eficiente para cargas extremadamente altas.• No es ideal para arquitecturas de microservicios complejos.



Node.js + NestJS	JavaScript/TypeScript	<ul style="list-style-type: none">• Rendimiento asíncrono para alta concurrencia.• NestJS ofrece estructura modular similar a Spring.• Flexible para distintos patrones multi-tenant (DB compartida o aislada).	<ul style="list-style-type: none">• Mayor complejidad al gestionar grandes bases de datos.• Más susceptible a errores de ejecución por su naturaleza asíncrona.
Laravel	PHP	<ul style="list-style-type: none">• Framework experimentado y bien documentado.• Soporte para multi-tenancy con paquetes como "Tenancy for Laravel".• Fácil despliegue en infraestructuras tradicionales.	<ul style="list-style-type: none">• Rendimiento comparativamente más lento que otros Frameworks.• Dependencia de demasiadas librerías externas.
Ruby on Rails + Apartment	Ruby	<ul style="list-style-type: none">• Rápido desarrollo con convenciones claras.• Gem "Apartment" para manejar multi-tenancy.• Comunidad sólida y bien documentada.	<ul style="list-style-type: none">• Rendimiento menor que opciones como Java o C#.• Dificultad para escalar horizontalmente.

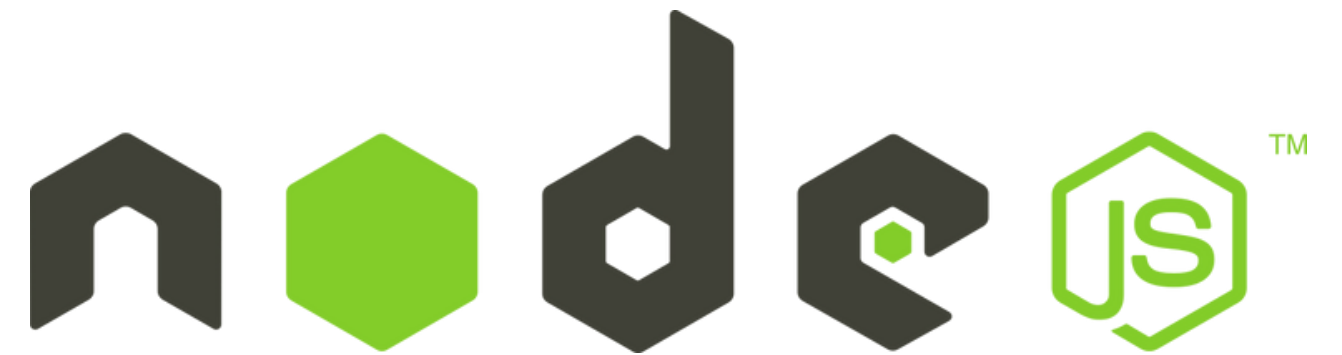
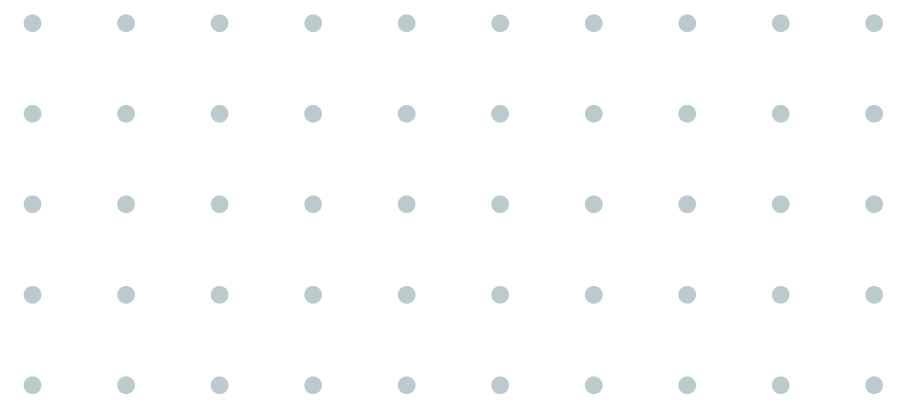


NODE.JS

Permite desarrollar aplicaciones eficientes y escalables, especialmente en sistemas basados en eventos y en tiempo real.

Ofreciendo un alto rendimiento, ideal para aplicaciones con muchas conexiones concurrentes, y se usa comúnmente para APIs RESTful.

Usa el mismo lenguaje que el frontend, facilitando el desarrollo.



TECNOLOGÍA SELECCIONADA

TECNOLOGÍAS PARA PERSISTENCIA DE DATOS

Cuadro Comparativo: ventajas y desventajas



TECNOLOGÍAS DE PERSISTENCIA

TECNOLOGÍA	VENTAJAS	DESVENTAJAS
PostgreSQL	<ul style="list-style-type: none">- Soporte avanzado para transacciones (ACID).- Escalable y maneja grandes volúmenes de datos.- Seguridad avanzada con autenticación basada en roles y cifrado.	<ul style="list-style-type: none">- Mayor consumo de RAM y CPU en comparación con MySQL.- Requiere mayor conocimiento técnico para su configuración y administración.
MySQL	<ul style="list-style-type: none">- Buen rendimiento en consultas de lectura.- Fácil de aprender y usar.- Amplia documentación	<ul style="list-style-type: none">- Menos eficiente en transacciones complejas en comparación con PostgreSQL.- Soporte limitado para consultas analíticas avanzadas.
MariaDB	<ul style="list-style-type: none">- Más rápido en ciertas operaciones que MySQL.- Mayor escalabilidad.	<ul style="list-style-type: none">- Menos soporte empresarial que SQL Server y PostgreSQL.- Puede presentar problemas de compatibilidad en algunas aplicaciones.
SQL Server	<ul style="list-style-type: none">- Buen soporte técnico y actualizaciones constantes.- Seguridad avanzada con cifrado de datos y control de acceso.- Alto rendimiento en sistemas transaccionales y analíticos.	<ul style="list-style-type: none">- Licencia de pago con costos elevados.- Mayor consumo de recursos en comparación con otras opciones.

PostgreSQL

PostgreSQL fue seleccionado por su escalabilidad, seguridad y rendimiento, que el proyecto necesita.

Su capacidad para manejar grandes volúmenes de datos sin afectar el rendimiento garantiza la disponibilidad y la escalabilidad de los datos.

Ofrece cifrado y control de accesos avanzados, asegurando la protección de datos sensibles.



TECNOLOGÍA SELECCIONADA



GRACIAS

POR SU ATENCIÓN

