



Aplicaciones Informáticas 2

- Nombre: Jennifer Yambay – 6916
- Fecha: 26 – 11 – 2024
- Docente: Ing. Julio Santillán

Tecnologías

Componente	Herramienta Propuesta	Alternativas	Ventajas de la Herramienta Propuesta	Por qué es la mejor opción
Frontend	React.js	Angular, Vue.js, Svelte	<ul style="list-style-type: none">• Ecosistema maduro con herramientas como Redux para la gestión de estado.• Alto rendimiento en aplicaciones complejas.• Fácil aprendizaje comparado con Angular.• Compatible con TypeScript para escalabilidad adicional.	<ul style="list-style-type: none">• React.js permite desarrollar aplicaciones rápidas y escalables con una gran comunidad de soporte. Sus capacidades interactivas son ideales para este proyecto.
Backend	Django REST Framework (DRF)	Node.js (Express), Flask, Ruby on Rails	<ul style="list-style-type: none">• DRF tiene herramientas integradas de autenticación y permisos, facilitando la seguridad desde el inicio.• Su integración con Python simplifica la creación de soluciones predictivas y análisis avanzado.• Escalabilidad y modularidad del código.	<ul style="list-style-type: none">• DRF lidera en proyectos de Machine Learning por su compatibilidad con bibliotecas avanzadas como TensorFlow y PyTorch.



Base de Datos	PostgreSQL	MySQL, MongoDB, SQLite	<ul style="list-style-type: none">• PostgreSQL maneja grandes volúmenes de datos estructurados y semiestructurados con eficiencia.• Ofrece extensiones como PostGIS para análisis geoespacial, útil si se requiere análisis por ubicación en el futuro.• Seguridad avanzada con roles y permisos.	<ul style="list-style-type: none">• PostgreSQL es versátil y seguro, con una combinación robusta de consultas relacionales y capacidades semiestructuradas que superan las alternativas.
Entorno Operativo	AWS (Servidor en la nube)	Google Cloud, Azure, DigitalOcean	<ul style="list-style-type: none">• Escalabilidad elástica: AWS permite aumentar recursos según la demanda del sistema.• Servicios integrados como S3 (almacenamiento) y RDS (bases de datos administradas).• Amplia documentación y soporte técnico.	<ul style="list-style-type: none">• AWS proporciona infraestructura confiable con herramientas que optimizan tanto el desarrollo como el despliegue.
Protocolo de Comunicación	HTTPS y WebSockets	HTTP, GraphQL, Subscriptions	<ul style="list-style-type: none">• HTTPS asegura la privacidad y autenticidad de los datos transmitidos.• WebSockets son ideales para funcionalidades en tiempo real como notificaciones o paneles de control dinámicos.	<ul style="list-style-type: none">• Esta combinación garantiza tanto la seguridad como la eficiencia para eventos en tiempo real, superando a HTTP y soluciones unidireccionales.
Lenguaje de Backend	Python	JavaScript (Node.js), Ruby	<ul style="list-style-type: none">• Python es versátil y ampliamente utilizado en proyectos de	<ul style="list-style-type: none">• Python es la mejor opción por su ecosistema rico en bibliotecas para



			<p>Machine Learning y análisis de datos.</p> <ul style="list-style-type: none"> Comunidad madura y recursos amplios. - Fácil de aprender y mantener. 	Machine Learning, análisis de datos y desarrollo web.
Framework para Frontend	React.js	Angular, Bootstrap	<ul style="list-style-type: none"> Ligero y modular en comparación con Angular. Fácil de integrar con librerías de terceros como Chart.js y Material-UI. 	<ul style="list-style-type: none"> React.js facilita la personalización de interfaces y se integra bien con frameworks CSS como Tailwind o Material Design.
Gestión de Versiones	GitHub	GitLab, Bitbucket	<ul style="list-style-type: none"> Amplia adopción y compatibilidad con herramientas CI/CD. Gran cantidad de plugins para monitoreo y análisis. 	<ul style="list-style-type: none"> GitHub es el estándar de la industria, asegurando integración perfecta con AWS y herramientas de desarrollo continuo.

1. React.js vs. Angular/Vue.js

- Escalabilidad:** React permite la creación de aplicaciones con componentes reutilizables y una arquitectura limpia, ideal para proyectos que escalan con el tiempo.
- Curva de aprendizaje:** Más simple que Angular, lo que reduce el tiempo de desarrollo inicial.
- Rendimiento:** React, junto con tecnologías como Webpack, optimiza la carga de recursos, mejorando el rendimiento general del sistema.

2. Django REST Framework vs. Node.js/Express

- Seguridad:** DRF proporciona mecanismos integrados para autenticación y permisos, mientras que en Node.js esto requiere librerías adicionales.
- Ecosistema:** Python y Django tienen un ecosistema orientado al análisis de datos y Machine Learning que supera a Node.js.
- Desempeño:** DRF es eficiente en el manejo de solicitudes gracias a su capacidad de caching y middleware flexible.

3. PostgreSQL vs. MySQL/MongoDB

- Consultas complejas:** PostgreSQL admite consultas avanzadas y funciones que no están disponibles en MySQL.
- Flexibilidad:** La capacidad de manejar JSONB lo convierte en una opción híbrida ideal entre bases de datos relacionales y no relacionales (como MongoDB).



4. AWS vs. Google Cloud/Azure

- **Costos flexibles:** AWS ofrece modelos de precios ajustables según las necesidades del proyecto.
- **Herramientas integradas:** Servicios como Lambda (computación serverless) y RDS (bases de datos administradas) simplifican la arquitectura del sistema.

5. WebSockets vs. HTTP Polling

- **Eficiencia:** WebSockets permiten mantener una conexión abierta y activa, reduciendo la latencia y la carga del servidor.
- **Funcionalidad en tiempo real:** Ideal para notificaciones, actualizaciones de estado y paneles interactivos.

Conclusión:

El ecosistema propuesto se alinea perfectamente con las necesidades del proyecto: escalabilidad, análisis avanzado y tiempos de desarrollo eficientes. Estas tecnologías, combinadas, no solo cumplen con los requisitos actuales, sino que también garantizan flexibilidad para futuras expansiones o integraciones.