

ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO



FACULTAD: INFORMÁTICA Y ELECTRONICA

CARRERA:

SOFTWARE

MATERIA:

APLICACIONES INFORMÁTICAS II

ESTUDIANTE:

JEANPIERRE DAVID AGUILAR ENCALADA – 7144

DOCENTE:

ING. JULIO ROBERTO SANTILLAN CASTILLO

NIVEL: OCTAVO SEMESTRE

PARALELO: “A”

SEPTIEMBRE 2025 – FEBRERO 2026

Pila de Herramientas

DESARROLLO DE UN SISTEMA WEB ASISTIDO POR UN AGENTE VIRTUAL PARA LA OPTIMIZACIÓN DE SERVICIOS, GESTIÓN DE INVENTARIO Y LA INTERPRETACIÓN PRELIMINAR DE RESULTADOS EN EL LABORATORIO CLÍNICO FRANZ.

1. Backend: Node.js (NestJS) + TypeScript

Node.js

- Entorno de ejecución JavaScript orientado a eventos, eficiente para I/O y concurrencia.
- Ecosistema amplio (npm), documentación extensa y comunidad activa.
- Permite full-stack con un solo lenguaje en frontend y backend.

NestJS

- Framework modular para construir APIs REST.
- Integración sencilla con colas de trabajo, schedulers y generación de archivos.

TypeScript

- Tipado estático que reduce errores en tiempo de compilación.
- Mejor autocompletado, refactorización segura y contratos más confiables.

Justificación: En este proyecto se prioriza **rapidez de desarrollo**. NestJS + TypeScript aportan estructura empresarial (módulos, DI, testing) sobre la **velocidad y ecosistema** de Node.js, lo que facilita exponer endpoints estables, integrar jobs asíncronos y mantener una base de código sostenible a medida que crece el dominio.

2. Base de Datos: PostgreSQL

- Sistema relacional ACID, maduro y de alto rendimiento.
- Tipos avanzados (JSONB, arrays) y funciones de ventana para analítica.
- Extensible, con replicación y herramientas maduras de backup y monitoreo.

Justificación: PostgreSQL ofrece la **integridad transaccional** exigida por datos de negocio, junto con **flexibilidad** para casos semiestructurados y **capacidad analítica** moderada sin añadir otro motor

3. Frontend: React (Next.js)

React

- Componentización y reactividad para interfaces reutilizables.
- Ecosistema amplio para formularios, estado y visualización.

Next.js

- SSR/SSG, enrutamiento integrado y optimizaciones de rendimiento/SEO.

- Soporte de i18n, optimización de imágenes y protección de rutas.

Justificación: Next.js acelera la entrega de un **dashboard interactivo**, con SEO y tiempos de carga. La combinación con React permite **components reutilizables** (tablas, formularios, gráficos)

4. Contenerización: Docker

- Empaqueta aplicaciones y dependencias en contenedores reproducibles.
- Entornos consistentes entre desarrollo, pruebas y producción.
- Compatible con múltiples orquestadores y nubes.

Justificación: Docker elimina discrepancias de entorno (“works on my machine”), **reduce fricción en CI/CD** y simplifica despliegues y rollbacks. Esto habilita ciclos de entrega más rápidos y predecibles.

5. Agente Virtual (Google): Dialogflow CX + Vertex AI (Gemini/PaLM)

Dialogflow CX

- Diseño de flujos conversacionales con estados e intenciones.
- Integración por webhooks y herramientas de prueba/analytics integradas.
- Soporte multicanal listo para producción.

Vertex AI (Gemini/PaLM)

- Modelos generativos gestionados para texto/clasificación y RAG.
- Controles de seguridad, registro y cuotas administradas en GCP.

Justificación: La combinación **Dialogflow CX + Vertex AI** permite capacidades LLM sin operar infraestructura de IA. Facilita **time-to-market**, observabilidad y cumplimiento de buenas prácticas de plataforma, manteniendo flexibilidad para integrar fuentes propias

6. Almacenamiento de Archivos: Amazon S3

- Almacenamiento de objetos duradero y escalable.
- Versionado, cifrado y políticas de acceso detalladas.
- Integración nativa con CDN y herramientas de análisis.

7. Herramientas de Apoyo: Postman + GitHub Actions

Postman

- Diseño, documentación y prueba interactiva de endpoints.
- Colecciones compartibles y pruebas automatizadas.

GitHub + Actions

- Control de versiones distribuido y CI/CD integrado.

- Automatización de lint, tests, builds y despliegues.