

JOSE DAVID CHAY GRIJALVA

Desarrollador Full-Stack

chepechay16@gmail.com | +502 55411552 | Ciudad de Guatemala

[GitHub](#) | [Linkedin](#) | [Portfolio](#)

EDUCATION

Universidad San Carlos de Guatemala

Licenciatura Ingenieria en Sistemas

Guatemala City

January 2021 - Present

Instituto Guillermo Putzeys Alvarez

Preparatoria

Guatemala City

January 2019 - October 2020

EXPERIENCE

Telus Digital Solutions | Full-stack Engineer

Guatemala City - Remote | Nov 2024 - Present

Contribu como desarrollador full-stack para un cliente estadounidense de alto perfil (confidencial). Me especialice en el desarrollo de aplicaciones web centradas en datos utilizando React, Firebase, Firestore y Firebase Functions. Implement soluciones avanzadas de gestion de estado con Redux y Zustand, y desarroll flujos de trabajo completos basados en formularios con Jest para pruebas y Playwright para automatizacin de control de calidad. Adems, dise y configur pipelines de CI/CD automatizados con GitHub Actions. Mis responsabilidades van ms all del desarrollo frontend, incluyendo la programacin en Python para el procesamiento de datos, como la manipulacin de imgenes y videos, as como la generacin de informes analiticos. Este rol fortaleci mi experiencia con la infraestructura en la nube, en particular con los servicios de Google Cloud Platform como IAM, Cloud Functions y Cloud Storage, aprovechando tambien AWS S3 para soluciones de almacenamiento a largo plazo eficientes. Mi trabajo apoya directamente las iniciativas de recopilacin y anlisis de datos del cliente, contribuyendo a procesos crticos de toma de decisiones.

MultiGroup | Frontend Engineer

Guatemala City | March 2024 - Nov 2024

Especializado en la creacin y optimizacin de interfaces web y mviles con TypeScript, React.js, Next.js y React Native. Adems, he brindado soporte y mejorado el rendimiento de aplicaciones Django optimizando consultas, consumiendo procedimientos almacenados y programando eventos asncronos con Celery, logrando una mejora del 75.08% en la velocidad de carga. Implement con xito la gestion de estado para aplicaciones web utilizando Zustand y Redux, empleando Reselect para optimizar la memorizacin, lo que result en un mejor rendimiento y una reduccin de renderizaciones innecesarias.

Trabaj en un proyecto de digitalizacin de documentos, migrando documentos fsicos de SAP a una aplicacin web. El proyecto implic la integracin de un script de PowerShell, un script de Node.js, un backend de Express.js con MongoDB y un frontend desarrollado con Next.js.

Grupo PB | Web Developer

Guatemala City | Jul 2022 - March 2024

Desarroll y mantuve aplicaciones web con React.js y Next.js, garantizando un alto rendimiento y capacidad de respuesta. Lider la integracin de servicios de AWS, incluyendo S3 y CloudFront, para optimizar la entrega de contenido en las aplicaciones frontend. Colabor estrechamente con los desarrolladores backend, mejorando las interacciones de la API y optimizando el flujo de datos en el frontend.

Fiverr | Freelance Programmer

Remote | Feb 2022 - Jun 2022

Desarroll sitios web responsivos utilizando HTML, CSS y JavaScript, centrndome en crear diseos limpios y fciles de usar. Implement soluciones JavaScript personalizadas para diversas necesidades de los clientes, mejorando la funcionalidad y la interactividad. Trabaj en proyectos full-stack, principalmente en el frontend, asegurando una integracin perfecta con los servicios backend.

SKILLS

Programming Languages: JavaScript, TypeScript, Python, SQL, HTML/CSS

Libraries/Frameworks: React.js, Next.js, Angular, Express.js, Nest.js, Django, Selenium, Playwright, Beatiful soup

Tools / Platforms: AWS, Azure, Google Cloude, Git/GitHub, Docker

Databases: MySQL, PostgreSQL, MongoDB, DynamoDB, Firebase, FireStore

PROJECTS / OPEN-SOURCE

Recognition IA | [Link](#)

Python, FastAPI, PostgreSQL, AWS (EC2, S3, RDS), Next.js,

Recognition IA es un sistema de reconocimiento facial integral diseñado como proyecto independiente, desarrollado completamente en Python e implementado mediante infraestructura moderna en la nube. Su arquitectura consta de tres servicios backend principales y una interfaz web para el cliente:

+ *IP Camera Capture*: Un script de Python de larga duración que captura video en tiempo real de cámaras de vigilancia. Extrae fotografías cada segundo y los envía mediante WebSocket al backend de reconocimiento.

+ *Recognition_API*: Un backend FastAPI de alto rendimiento diseñado con arquitectura hexagonal y Screaming. Este servicio procesa las imágenes entrantes utilizando DeepFace y RetinaFace para detectar el género, la edad estimada, las probabilidades de emoción (como un diccionario de puntuaciones de confianza) y las incrustaciones faciales. Los datos procesados se envían al servicio de base de datos mediante una API REST, mientras que las imágenes sin procesar se suben a un bucket de AWS S3 con limpieza automatizada (las imágenes se eliminan después de 48 horas).

+ *Recognition_DB*: Una API RESTful construida con FastAPI y PostgreSQL para almacenar y recuperar datos de reconocimiento. Admite operaciones CRUD básicas (principalmente GET y POST). Además, desarrolló una interfaz web Next.js con el estilo de Chakra UI, que simula una experiencia similar a la de Google Meet. Este frontend captura la entrada de la cámara web y envía fotografías a través de WebSocket a la API de reconocimiento, que responde con datos demográficos y emocionales en tiempo real para mostrarlos en la interfaz.

El sistema se implementó utilizando servicios de AWS: EC2 (con instancias optimizadas para el reconocimiento), RDS para PostgreSQL, S3 para el almacenamiento de imágenes y ECR para la gestión de contenedores. Se utilizaron GitHub Actions y AWS CodeDeploy para la automatización de CI/CD. En algunos casos, también se realizaron implementaciones manuales en servidores Linux propiedad del cliente.

RRHH System | Link

> *Frontend: React.js, Sass* > *Backend: Python*

Desarrollé un sistema completo de gestión de recursos humanos con módulos para empleados, departamentos y puestos de trabajo.

El frontend se construyó con Angular, proporcionando una interfaz de usuario limpia y adaptable para la gestión de registros de empleados, departamentos y puestos de trabajo. El sistema permite operaciones CRUD en todos los módulos, utilizando formularios dinámicos y obtención de datos con integraciones de API RESTful. El backend se desarrolló con Python y Flask, estructurado con planos modulares, e implementé las mejores prácticas para el diseño de API, incluyendo validación, manejo de excepciones e interacción con una base de datos PostgreSQL.

ENLACE AL BACKEND: https://github.com/Josepe20/rrhssystemtest_api

CERTIFICATIONS

- AWS Cloud Practitioner - **Amazon Web Services**
- 100 Days of Code: The Complete Python Pro Bootcamp for 2023 - **Udemy, Angela Yu**
- TypeScript Crash Course - **Udemy, Jan Schaffranek**
- Curso Básico de Django - **Platzi, Facundo Garcia Martoni**
- Curso Práctico de SQL y MySQL - **Platzi, Alberto Alcocer**
- Curso Profesional de Git y GitHub - **Platzi, John Freddy Vega**
- Curso Profesional de Scrum - **Platzi, Gerardo Romero**
- English B2, Working With Clients - **British Council**
- Gestión de datos en Excel - **Camara de comercio y capacitacion international**