Gehlot P

Full Stack Engineer, 4+ years experience
Jaipur, India
https://augmntx.com/profile/full-stack-engineer-gehlot-p/axo32120

AugmntX
+91 9820045154 | hr@augmntx.com

Full Stack Engineer with over 3 years of experience specializing in building scalable, AI-driven, and cloud-native web applications. Skilled in Python, Django, Flask, FastAPI, React, PostgreSQL, and Redis, with strong expertise in DevOps, CI/CD automation, and cloud platforms like GCP. Experienced in developing intelligent automation systems, predictive analytics solutions, and workflow orchestration platforms using RESTful APIs, WebSockets, and task queues. Proficient in leveraging OpenAI APIs, Vertex AI, and BigQuery for data-driven decision-making and automation.

INDUSTRIES: Human Resources, Marketing, Manufacturing and Industrial Marketplace Platform, IT services, Artificial Intelligence

TECHNICAL SKILLS

• Python	3 years & 5 months
• DevOps	3 years & 7 months
• REST APIs	3 years & 3 months
• FastAPI	3 years & 3 months
• LLM	3 years & 2 months

SOFT SKILLS

Confident, Communication, Teamwork, Time management, Critical thinking

PROJECT DETAILS

• AI-Powered Workflow Automation February-2025 - October-2025

Project Description: he client wanted to automate repetitive manual workflows across HR, finance, and operations using AI-based decision systems capable of triggering intelligent actions through API integrations. The solution involved designing a FastAPI-based backend architecture to orchestrate complex automation flows and handle multi-system event triggers. Server-Sent Events (SSE) were implemented for real-time status updates and log streaming between backend services and the React frontend. Redis was used as a caching layer, while Celery handled background job scheduling and asynchronous processing. PostgreSQL with SQLAlchemy ORM managed structured workflow and user data. OpenAI GPT APIs powered intelligent text analysis, document summarization, and contextual decision-making. The React-based frontend enabled users to track automations, trigger workflows, and monitor AI outputs in real time. The deployment leveraged GCP Cloud Functions and Cloud Run for serverless scalability, with CI/CD automation handled through GitHub Actions. Secure access was ensured via JWT-based authentication and role-based access control. Stackdriver Monitoring and Logging were configured to track workflow performance and uptime, while collaboration with business and DevOps teams enabled an auto-healing deployment strategy for high-availability environments.

Responsibilities: Responsibilities: Designed and developed FastAPI based backend architecture to manage complex automation workflows and process multi system triggers. Implemented Server Sent Events for real time log streaming between backend and React frontend. Used Redis as a caching layer to improve workflow state retrieval and reduce database load. Configured Celery with Redis for background job scheduling and asynchronous execution. Utilized PostgreSQL with SQLAlchemy ORM for workflow and user data storage. Integrated OpenAI GPT API for intelligent text analysis, summarization, and classification to support decision making. Built React based frontend for users to monitor automations, trigger workflows, and view AI outputs in real time. Deployed serverless components on GCP Cloud Functions and Cloud Run for scalable performance. Created CI CD pipelines in GitHub Actions for automated build, testing, and deployment. Implemented role based access control and secure authentication using JWT. Set up Stackdriver Monitoring and Logging for workflow tracking and uptime alerts. Collaborated with DevOps teams to establish an auto healing deployment strategy for high availability.

Technologies: FastAPI, React, Redis, Celery, PostgreSQL, SQLAlchemy, OpenAI GPT API, GCP Cloud Functions, Cloud Run, GitHub Actions, JWT, Stackdriver, SSE

Industry: Human Resources

• AI-Driven Data Pipeline April-2022 - October-2022

Project Description: The client aimed to build an intelligent marketing automation engine that could segment customers, predict churn, and dynamically trigger personalized campaigns. The solution included developing a Django-based backend with modular APIs for customer segmentation, churn scoring, and campaign scheduling. PostgreSQL ORM was used for data persistence, while Redis caching enabled faster retrieval of segmentation results. Playwright was used for automated UI and API testing. An event-driven data pipeline was built using Pub/Sub and Cloud Functions to handle real-time data ingestion from multiple CRMs and transaction systems. Vertex AI models powered churn prediction and lookalike audience creation using historical engagement data. A React-based interface was developed for marketing teams to design and manage campaigns. GCP Cloud Scheduler was used for data versioning and model retraining, and GitHub Actions pipelines were implemented for automated testing and multi-environment deployment.

Responsibilities: Designed and implemented a Django based backend architecture with modular APIs for customer segmentation, churn prediction, and campaign scheduling. Managed data storage and access using PostgreSQL ORM for model and campaign data persistence. Optimized performance with Redis caching for faster retrieval of segmentation results. Automated campaign testing using Playwright across multiple browsers. Built an event driven data pipeline using Pub Sub and Cloud Functions for real time CRM and transaction data ingestion. Integrated Vertex AI models for churn prediction and lookalike audience creation. Developed a React based campaign management dashboard enabling marketing teams to design and execute dynamic campaigns. Automated data versioning and model retraining via GCP Cloud Scheduler to enhance model adaptability. Created CI CD pipelines with GitHub Actions for streamlined testing and multi environment deployment. Designed BigQuery and Data Studio dashboards to visualize campaign performance and retention metrics. Collaborated with DevOps teams to containerize services using Docker and optimize cloud resource utilization.

Technologies: Django, Python, PostgreSQL, Redis, React, Playwright, Pub/Sub, Cloud Functions, Vertex AI, GCP Cloud Scheduler, GitHub Actions, BigQuery, Data Studio, Docker

Industry: Marketing

• Predictive Analytics Dashboard July-2023 - May-2024

Project Description: The client needed a data intelligence platform to predict equipment downtime, optimize resource allocation, and forecast inventory shortages using machine learning models. Built FastAPI-based backend services for ingesting IoT device data and transforming it into analytical datasets for AI model consumption. Added WebSocket integration for real-time dashboard updates on IoT data ingestion and predictive alerts. Configured Redis for temporary caching of IoT data streams and reducing BigQuery load. Used PostgreSQL ORM for storing device metadata and historical event logs. Implemented background job queues to handle asynchronous data preprocessing and anomaly detection tasks. Leveraged BigQuery pipelines to process large-scale production data and integrate with Vertex AI for predictive modeling. Created React dashboards to visualize real-time operational KPIs, predictive alerts, and maintenance schedules. Designed API-based integration with ERP systems to automatically sync production and inventory data. Implemented OAuth 2.0 authentication for secure role-based dashboard access. Used Cloud Run for containerized deployment, ensuring auto-scaling based on processing load. Configured Stackdriver Monitoring to alert teams of threshold breaches and potential downtime risks. Collaborated with data scientists to improve prediction accuracy through feature tuning and feedback loops.

Responsibilities: Designed and developed FastAPI-based backend services for IoT data ingestion and transformation into analytical datasets. Implemented WebSocket integration for real-time dashboard updates and predictive alert notifications. Configured Redis caching to manage IoT data streams efficiently and minimize BigQuery load. Utilized PostgreSQL ORM for structured storage of device metadata and historical event logs. Developed background job queues to handle asynchronous data preprocessing and anomaly detection. Built BigQuery pipelines for large-scale data processing and integrated Vertex AI for predictive modeling. Created React dashboards to display operational KPIs, predictive alerts, and maintenance schedules. Integrated with ERP systems using APIs for automatic production and inventory data synchronization. Implemented OAuth 2.0 authentication for secure, role-based dashboard access. Deployed containerized services using Cloud Run for scalable and efficient processing. Configured Stackdriver Monitoring to alert teams on performance thresholds and potential downtime risks. Collaborated with data scientists to improve model accuracy through feature engineering and feedback iterations.

Technologies: FastAPI, React, WebSockets, Redis, PostgreSQL, BigQuery, Vertex AI, Cloud Run, OAuth 2.0, Stackdriver Monitoring, ERP Integration APIs, Python, Docker

Industry: Manufacturing and Industrial Marketplace Platform

• Cloud-Native DevOps Automation November-2022 - June-2023

Project Description: To create an internal DevOps automation platform to streamline code validation, deployment orchestration, and CI/CD governance across multi-cloud environments. The solution included developing a Python automation engine to manage metadata deployment, rollback validation, and environment synchronization. FastAPI microservices were designed for managing job triggers and deployment logs via REST APIs. Real-time build and deployment logs were streamed to a React-based monitoring console using SSE endpoints. Redis task queues handled parallel job execution and background validation processes. GitHub Actions and Docker pipelines were integrated for automated build, linting, and deployment verification. GCP Cloud Build and Cloud Storage were configured for artifact management and persistent deployment tracking. Automated Slack notifications were established for pipeline status and rollback triggers. Infrastructure as Code scripts were implemented for dynamic provisioning of GCP services, and CI/CD best practices were documented using Confluence.

Responsibilities: Developed a Python automation engine for metadata deployment, rollback validation, and environment synchronization. Designed FastAPI microservices to manage job triggers and deployment logs through REST APIs. Implemented SSE endpoints for real-time build and deployment log streaming to the React console. Integrated Redis task queues for parallel job execution and background validation. Set up GitHub Actions and Docker pipelines for automated build, linting, and deployment. Built a React-based monitoring console to display build status, logs, and pipeline metrics. Configured GCP Cloud Build and Cloud Storage for artifact management and deployment tracking.Enabled Slack notifications for pipeline status, approvals, and rollback alerts. Created IaC scripts for dynamic provisioning of GCP services in test environments. Documented CI/CD best practices and DevOps guidelines using Confluence.

Technologies: Python, FastAPI, REST APIs, React, Redis, GitHub Actions, Docker, GCP Cloud Build, GCP Cloud Storage, Slack API, Infrastructure as Code (IaC), Confluence, SSE (Server-Sent Events)

Industry: IT services

• Customer Support Automation January-2024 - January-2025

Project Description: The client required an AI-driven customer service engine capable of classifying incoming support tickets, predicting sentiment, and routing them to the appropriate departments automatically. The solution included Flask-based microservices for ticket classification, sentiment prediction, and response recommendations using REST APIs. Background task execution was managed using Celery workers for asynchronous NLP predictions, while Redis queues handled caching and background job management. Real-time ticket updates were implemented using WebSockets on a Vue.js dashboard. PostgreSQL stored ticket logs and model performance data, and Vertex AI was used for intent detection and emotion scoring. The system integrated with Salesforce CRM and Freshdesk for seamless ticket synchronization. Containerized services were deployed using Docker on GCP Cloud Run for scalability. CI/CD pipelines were implemented using GitHub Actions, and BigQuery dashboards were configured for real-time analytics on ticket categories, AI accuracy, and SLA performance.

Responsibilities: Designed and developed Flask-based microservices for ticket classification, sentiment prediction, and AI-driven response recommendations. Implemented Celery with Redis for asynchronous background job execution and AI model predictions. Developed WebSocket integration for real-time ticket status updates on the Vue.js dashboard. Built and managed PostgreSQL databases for storing ticket logs and model performance metrics. Integrated Vertex AI NLP models for intent detection and emotion scoring. Developed an interactive Vue.js dashboard for visualizing customer sentiment and agent performance. Integrated Salesforce CRM and Freshdesk via APIs for seamless ticket management and synchronization. Deployed services using Docker containers on GCP Cloud Run ensuring scalability and reliability. Set up CI/CD pipelines with GitHub Actions for automated builds, testing, and deployment approvals. Configured BigQuery dashboards to monitor ticket trends, model accuracy, and SLA compliance. Collaborated with data engineers to continuously improve NLP model accuracy through feedback loops.

Technologies: Flask, Celery, Redis, PostgreSQL, Vertex AI, Vue.js, WebSockets, REST APIs, Salesforce API, Freshdesk API, Docker, GCP Cloud Run, GitHub Actions, BigQuery, NLP (Natural Language Processing), Python.

Industry: Artificial Intelligence

EMPLOYMENT HISTORY

• Full stack Engineer April-2022 - November-2025

Description: Role Summary Full Stack Engineer with expertise in Python, JavaScript, RESTful APIs, Flask, FastAPI, React, and GCP, skilled in designing AI-powered automation, predictive analytics, and DevOps frameworks. Experienced in developing scalable backend systems, integrating AI/LLM APIs, and building interactive dashboards for data-driven decision-making. Core Responsibilities Backend Development: Built scalable microservices using FastAPI, Flask, and Django, integrating with PostgreSQL, Redis, and GraphQL APIs. Designed RESTful APIs and real-time communication channels using WebSockets and Server-Sent Events (SSE). AI & Machine Learning Integration: Integrated OpenAI APIs and LLMs for intelligent automation, sentiment analysis, and text processing. Collaborated with data scientists to deploy ML models using Vertex AI and optimized pipelines for continuous learning. Frontend & Full Stack Engineering: Developed interactive React and Vue.js dashboards for workflow monitoring, analytics, and automation tracking. Implemented RBAC, JWT authentication, and responsive UI components for secure user interaction. Database & Data Processing: Designed structured schemas using PostgreSQL, optimized queries, and implemented BigQuery pipelines for real-time insights. Used Celery and Redis task queues for asynchronous and background processing. DevOps & CI/CD: Automated deployments using GitHub Actions, Docker, and GCP Cloud Run with environment-based approvals. Implemented monitoring with Stackdriver, maintaining high availability through auto-healing and serverless architecture. Applied Infrastructure as Code (IaC) and managed DevOps governance using Confluence and JIRA. Collaboration & Documentation: Worked in Agile environments using JIRA for sprint tracking and Confluence for maintaining technical documentation. Partnered with cross-functional teams including data engineering, QA, and DevOps to ensure delivery excellence. Tech Stack Languages & Frameworks: Python, JavaScript, Flask, FastAPI, Django, React, Vue.js, Node.js Databases: PostgreSQL, MongoDB, Redis, BigQuery AI & APIs: OpenAI API, LLMs, GraphQL, REST APIs DevOps Tools: Docker, GitHub Actions, GCP (Cloud Run, Cloud Functions, Stackdriver), IaC Others: JIRA, Confluence, Celery, SSE, WebSockets, CI/CD Pipelines

EDUCATION

• Bachelors degree in Computer Application from JECRC University, Jaipur 2020 - 2023

LANGUAGES

• English - native

END OF DOCUMENT
This PDF is automatically generated by AugmntX on 20 Nov 2026
Profile URL:https://augmntx.com/profile/full-stack-engineer-gehlot-p/axo32120