Shreyansh Kabra

Los Angeles, CA · +1-(213)284-5169 · kabras@usc.edu · LinkedIn · Github · Portfolio

Backend Software Engineer with 2+ years of production experience at Hewlett Packard Enterprise (Cloud Services). Specialized in building scalable, asynchronous microservices and optimizing distributed systems. Currently pursuing MSCS at USC.

Professional Experience

Cloud Developer 1 **Aug 2022 - Dec 2024**
Hewlett Packard Enterprise *Bangalore, India*

- Architected and deployed 100+ scalable RESTful APIs using Python, Django, and DRF for the GreenLake edge-to-cloud platform, optimizing data synchronization between PostgreSQL and ServiceNow.
- Spearheaded the full-stack development of a web-based email notification microservice (API + UI), streamlining critical communication workflows and increasing operational efficiency by 80% through asynchronous processing.
- Integrated React.js based front-end components with backend services for Service Insights Portal, achieving a 30% efficiency gain in task management and reducing submission errors by 50%.
- Collaborated with 10+ international cross-functional teams to implement secure, distributed cloud solutions and mentored 3 junior developers on code quality and best practices.

R&D Intern, NaaS Team **Jan 2022 - Aug 2022**
Hewlett Packard Enterprise *Bangalore, India*

- Developed an asynchronous Python automation suite to manage and configure network device calibration, eliminating 90% of manual intervention and enhancing system reliability at scale.
- Streamlined troubleshooting report execution workflows, reducing task completion latency by 50% and overall engineering effort by 70%.
- Designed and implemented internal REST APIs to bridge ServiceNow with legacy applications, establishing secure cross-platform data pipelines and improving operational efficiency by 50%.

Technical Skills

Languages & Frameworks: Python, JavaScript, SQL, C++, HTML5, CSS3, Kotlin, Django, React.js, Node.js
Cloud & Infrastructure: Google Cloud Platform (GCP), Docker, Kubernetes, ServiceNow, Postman, Git
Databases & Concepts: PostgreSQL, MongoDB, MySQL, Redis, Microservices, REST APIs, Asyncio, System Design, RAG

Education

Master of Science, Computer Science **Jan 2025 - Dec 2026 (Expected)**
University of Southern California, Los Angeles, California

Bachelor of Technology, Computer Science and Engineering **Aug 2018 - May 2022**
JECRC University, Jaipur, Rajasthan

Academic Projects

Artist Search Platform |*React.js, Node.js, Kotlin, MongoDB* **Mar 2025 - May 2025**

- Engineered a full-stack application including a React.js web app and native Android app (Kotlin, Android SDK), powered by a Node.js + Express backend, integrating the Artsy API to deliver data on 100,000+ artists and artworks.
- Designed secure, scalable data management using structured MongoDB collections and JWT-based authentication, ensuring reliable access control and session security.
- Optimized frontend performance and accessibility with 100% Lighthouse scores for both mobile and desktop.

Mental Health Signal Detector |*NLP, RAG, RoBERTa, FAISS* **Aug 2025 - Dec 2025**

- Engineered an automated data ingestion pipeline to process and index 6000+ unstructured text samples, implementing Regex-based sanitization and zero-shot labeling to bootstrap the dataset.
- Architected a high-performance Retrieval-Augmented Generation (RAG) backend integrating FAISS for vector similarity search, enabling real-time semantic retrieval of context for the generative response engine.
- Optimized system resource efficiency by implementing Low-Rank Adaptation (LoRA), significantly reducing the memory footprint for deployment in constrained environment.

Transfer Learning for Image Classification |*Keras, TensorFlow* **Jun 2025 - Jul 2025**

- Developed a waste image classification system using transfer learning with pre-trained CNNs (EfficientNetB0, ResNet50/101, VGG16) across 9 categories, with 83% validation accuracy and 97.66% AUC with EfficientNetB0.
- Enhanced model performance by applying data augmentation, early stopping, learning rate scheduling, and class weight balancing; leveraged TensorFlow/Keras, OpenCV, and scikit-learn for training and evaluation.