



dheerajkom2804@gmail.com | +1 (323)-347-1443

LinkedIn | GitHub | Portfolio | Medium

Location: Fullerton, California

Professional Summary:

- Around **7+ years of experience** across all phases of the Software Development Life Cycle (SDLC) building scalable **Python full stack applications** in Healthcare, Retail, and Enterprise domains
- Strong expertise in building **dynamic, responsive user interfaces** using **React.js**, including components, hooks, context API, state management, routing, and performance optimization, with hands-on experience in **HTML5, CSS3, JavaScript/TypeScript, Tailwind CSS, and Bootstrap**
- Proficient in designing and developing **high-performance backend services** using **FastAPI, Flask, and Django**, following clean architecture and RESTful design principles
- Extensive experience building and consuming **RESTful APIs and microservices**, handling authentication, authorization, validation, and API documentation using **Swagger/OpenAPI**
- Experienced in working with **relational and NoSQL databases**, including **PostgreSQL, MySQL, MongoDB, and Redis**, covering schema design, query optimization, and data access patterns
- Experienced in building and deploying **cloud-native applications on AWS**, using services such as **EC2, S3, RDS, Lambda, and SageMaker**, with containerization and orchestration using **Docker and Kubernetes**
- Proficient in **CI/CD pipelines** using **GitHub Actions and Jenkins**, enabling automated builds, testing, and deployments in production environments
- Hands-on experience integrating **AI/ML and GenAI solutions**, including **RAG pipelines, LLM-based APIs, and NLP workflows** using **OpenAI, Hugging Face, and LangChain**, with strong awareness of application security practices such as **JWT, OAuth2, and RBAC**

Technical Skills:

Languages	Python, JavaScript, TypeScript, SQL
Frontend Technologies	HTML5, CSS3, JavaScript (ES6+), JSON, AJAX, Tailwind CSS, Bootstrap
Backend Technologies	FastAPI, Flask, Django, RESTful APIs, Async Programming, Node.js (Express – integration level)
Web Frameworks / Libraries	React.js, Next.js, React Native (basic exposure), Lit (Web Components)
Databases	PostgreSQL, MySQL, MongoDB, Redis, DynamoDB
AI / ML & GenAI	LangChain, RAG Pipelines, OpenAI APIs, Hugging Face Transformers, scikit-learn, TensorFlow (integration)
Cloud Platforms	AWS (EC2, S3, RDS, Lambda, SageMaker), Azure, GCP
CI/CD Tools	GitHub Actions, Jenkins, GitLab CI, AWS CodePipeline
Containerization & Orchestration Tools	Docker, Kubernetes
Version Control & Dev Tools	Git, GitHub, Postman, Swagger/OpenAPI, VS Code, JIRA
Operating Systems	Linux, UNIX, Windows
Certifications	AWS Certified Developer – Associate, Oracle Cloud Infrastructure 2025 Certified Generative AI Professional

Professional Experience:

Client: Volkswagen, Rochester, MI, USA
Role: Sr. Full stack Developer

March 2024 - Present

Responsibilities:

- Designed and developed **responsive, cross-browser user interfaces** using React.js, building reusable components, custom hooks, and modular UI patterns to support transaction-heavy workflows
- Managed **complex frontend state and data flows** using React Context API and Redux where applicable, ensuring predictable updates and smooth user interactions
- Ensured **seamless frontend-backend integration** by consuming REST APIs with consistent data contracts and handling loading, error, and retry states gracefully
- Optimized frontend performance through **efficient rendering strategies and component-level optimizations**, improving responsiveness under high data volumes
- Architected and developed **enterprise-scale backend services using Python (FastAPI, Flask)** following layered and microservices architectures to support secure, high-throughput financial transactions
- Designed and implemented **RESTful APIs** for transaction orchestration, validation, reconciliation, and reporting, ensuring idempotency and fault tolerance under high concurrency
- Integrated LLM-based services using **LangChain and OpenAI APIs** to support intelligent search, summarization, and analytics workflows across transaction and operational data
- Implemented **secure authentication and authorization mechanisms** using JWT, OAuth2, and role-based access control (RBAC) to protect sensitive financial operations
- Developed **asynchronous and concurrent backend workflows** using Python async/await patterns to efficiently process parallel requests and external service integrations
- Implemented **robust error handling, validation, and standardized HTTP responses** to improve API reliability and observability
- Integrated **event-driven messaging using Apache Kafka** to support real-time transaction events, reconciliation workflows, and downstream processing across microservices
- Designed and optimized **PostgreSQL schemas** to ensure transactional consistency, auditability, and support for high-volume financial data
- Worked with **distributed data stores including Cassandra** to manage large-scale transaction history, audit logs, and event data with horizontal scalability
- Improved backend performance using **query tuning, indexing strategies, connection pooling, and Redis caching**, reducing latency and system load
- Containerized backend services using **Docker**, ensuring consistent environments across development, testing, and production
- Deployed and managed applications on **Kubernetes (AWS EKS)**, implementing rolling and blue-green deployment strategies to minimize downtime
- Automated **CI/CD pipelines using Jenkins** and leveraged AWS services (EC2, EKS, ECS, S3, RDS) to build scalable, resilient, cloud-native infrastructure with proactive monitoring

Client: HP, Vancouver, WA, USA.
Role: Full Stack Developer

April 2022 - February 2024

Responsibilities:

- Designed, developed, and maintained **full stack web applications** using Python backend services and React.js frontend components to support real-time data visualization and monitoring dashboards

- Designed and developed **React.js user interfaces** using reusable components and hooks to visualize system health, trends, and predictive analytics results
- Managed **complex frontend state** using React Context API and Redux to support real-time updates, filtering, and dashboard-level configurations
- Ensured **responsive design and cross-browser compatibility**, enabling consistent dashboard usage across desktops, tablets, and large display monitors
- Implemented **WebSocket-based real-time communication** between backend services and frontend dashboards to stream live infrastructure metrics and alerts
- Built **scalable RESTful APIs using Python (FastAPI and Flask)** to ingest, process, and expose telemetry and metrics data collected from IoT devices
- Implemented a **microservices-based backend architecture**, enabling independent scaling of ingestion, processing, analytics, and visualization services
- Integrated **MQTT-based messaging pipelines** to efficiently receive high-frequency IoT data with low latency and reliable ingestion
- Implemented **asynchronous and background processing** using Python to handle high-volume data ingestion and analytics tasks without blocking real-time workflows
- Integrated **predictive analytics workflows** to proactively identify trends, anomalies, and potential infrastructure issues
- Implemented **robust error handling, validation, and retry mechanisms** to ensure reliability when processing high-throughput data streams
- Secured backend APIs using **JWT-based authentication and role-based access control**, ensuring restricted access to sensitive infrastructure data
- Designed and optimized **PostgreSQL schemas**, and worked with **Cassandra** to manage large-scale time-series and historical datasets with high availability
- Used **Swagger/OpenAPI** to design, document, and validate REST API contracts for backend services
- Containerized backend and frontend services using **Docker** and deployed applications using **Kubernetes**, enabling automated scaling, fault recovery, and high availability
- Leveraged **AWS services** along with **AWS Lambda and SQS** for scalable hosting, serverless processing, and asynchronous message queuing
- Built and maintained **CI/CD pipelines** and monitored application performance using centralized logging and monitoring tools to enable proactive issue detection and faster incident response

Client: Ulta Beauty, Bolingbrook, IL
Role: Full Stack Developer

January 2020 - March 2022

Responsibilities:

- Designed and developed **full stack application features** using Python-based backend services and React.js front-end components to support patient and provider workflows within the customer-facing digital platform
- Designed and implemented **React.js user interfaces** for internal tools and dashboards, supporting forms, routing, events, and dynamic data rendering
- Built **reusable React components and utility functions**, enabling faster development of new screens and consistent UI behavior across the platform
- Managed **application state and data flow** using React Context and Flux-inspired patterns, ensuring predictable UI behavior and maintainable frontend logic
- Ensured **responsive design and cross-browser compatibility** using modern CSS and JavaScript techniques to support a wide range of devices
- Implemented **real-time and near-real-time UI updates** by consuming backend APIs and event-driven data streams
- Built and enhanced **RESTful APIs using Python (FastAPI and Flask)** to handle appointment management, patient data access, provider availability, and system integrations

- Designed backend components following **service-oriented and layered architecture principles**, enabling clean separation of concerns and easier maintainability
- Implemented a **microservices-based architecture** to decouple platform components, enabling independent scaling and improved fault isolation
- Integrated **Apache Kafka** to enable asynchronous, event-driven workflows such as appointment updates, notifications, and audit events, and maintained reliable producers and consumers
- Optimized backend logic using **asynchronous processing and concurrency patterns in Python**, improving throughput for concurrent user requests
- Designed and optimized **database access layers** to ensure efficient query execution and reliable retrieval of patient and clinical data from relational databases
- Implemented **robust error handling, validation, and standardized API responses** to ensure correct behavior and meaningful error reporting
- Implemented **secure authentication and authorization mechanisms**, ensuring controlled access to sensitive patient and provider data
- Deployed backend services using **Docker-based containerization** and utilized **AWS ECS and AWS Lambda** for scalable deployments and serverless background processing
- Configured **AWS IAM and CloudWatch** for access control, monitoring, logging, alerting, and supported production issues through performance tuning and root-cause analysis

Client: Voyager IT Solutions, Bangalore, India.

June 2018–December 2019

Role: Software Developer

Responsibilities:

- **Contributed to full stack development**, participating in requirements analysis, implementation, testing, and deployment of web application features
- Built **responsive user interfaces** using HTML, CSS, JavaScript, and Bootstrap, ensuring cross-browser compatibility
- Designed and developed **Python-based backend services and RESTful APIs** to implement core business logic and system integrations
- Worked with **relational databases** to design schemas, write queries, and support application data needs
- Wrote **unit tests** and implemented **error handling and logging**, improving application reliability and ease of troubleshooting
- Used **Git for version control** and collaborated with team members through code reviews and issue resolution
- Followed **Agile development practices**, actively participating in sprint planning, stand-ups, and team collaboration

Education:

Master's in computer science, California State University Fullerton, Fullerton, California

Bachelor's in computer science, BML Munjal University, Gurgaon, India