**A hexagon with white text

AI-generated content may be incorrect.Yesho Reddipalli**

**Email:** [yeshoreddipalli@gmail.com](mailto:yeshoreddipalli@gmail.com) **| Ph:** +1(507) 237-6376

[**GitHub**](https://github.com/Yesho-reddipalli) **|**[**Portfolio**](https://yesho-reddipalli.github.io/yesho-reddipalli/) **|**[**LinkedIn**](https://www.linkedin.com/in/yesho-reddipalli-449184345/)

**Professional Summary**

* Python Developer with 7+ years of experience in designing, implementing, and maintaining complex applications and systems.
* Expertise in building scalable microservices and API-driven architectures to support high-volume and real-time data processing.
* Extensive experience with cloud computing technologies, particularly AWS, Azure, GCP, creating highly available, cost-effective solutions.
* Proficient in data engineering and developing ETL pipelines using tools like AWS Glue, Apache Airflow, and PySpark for seamless data transformation.
* Skilled in integrating streaming data platforms, implementing Apache Kafka for real-time data ingestion and event-driven architectures.
* Experienced in developing modern, dynamic, and responsive web applications using React and Angular, with a strong focus on component-based architecture and state management.
* Skilled in enhancing user experience (UX) by optimizing page load performance, implementing interactive UI elements, and ensuring cross-browser compatibility using HTML5, CSS3, and JavaScript.
* Focused on real-time platforms, leveraging expertise in Python and microservices to support computer vision models with high availability and performance.
* Strong background in designing secure, high-performance backend services using Django, Flask, and FastAPI.
* Expertise in serverless architectures with AWS Lambda, automating workflows, and reducing operational overhead.
* Experienced in creating CI/CD pipelines with Jenkins, GitHub Actions, and GitLab, ensuring automated, streamlined deployments.
* Hands-on experience with containerization and orchestration technologies like Docker and Kubernetes for consistent development and production environments.
* In-depth knowledge of AWS CloudFormation to manage infrastructure as code, enabling automated and repeatable deployments.
* Expertise in database management and optimization, working with MySQL, MS SQL, DynamoDB, and Amazon Redshift.
* Strong focus on data quality and analytics, using AWS Glue and Amazon Redshift for reliable reporting and data warehousing solutions.
* Well-versed in agile development practices, using Jira and Scrum methodologies to foster collaboration and efficient project delivery.
* Excellent problem-solving abilities, demonstrated in optimizing Python scripts, conducting code reviews, and delivering solutions aligned with business goals.
* Strong hands-on experience with Kubernetes, Docker, and cloud platforms such as Azure and GCP for container orchestration and deployment scalability.
* Skilled in load testing, unit testing, and regression testing for optimizing GPU/CPU-intensive applications.

**Professional Experience**

**Software Developer**

**Medtronic, Minneapolis, MN Jun 2023 to Present**

**Responsibilities:**

* Developed and maintained Django-based web applications for managing and visualizing medical device data, ensuring efficient backend integration with databases and frontend interfaces.
* Implemented Django REST Framework to create secure and scalable APIs for integrating medical device data with external applications, supporting real-time updates and data retrieval.
* Enhanced data quality checks in ETL processes using Python, minimizing errors in downstream analytics pipelines.
* Designed and implemented ETL pipelines with AWS Glue to transform large datasets for analytics and reporting.
* Developed Python-based microservices for real-time data processing, enabling efficient communication between medical devices and applications.
* Optimized frontend performance by integrating lazy-loading techniques and utilizing React.memo to minimize unnecessary re-renders.
* Implemented advanced caching mechanisms using Redis to enhance API response times for high-frequency medical data queries.
* Developed dynamic data visualization components using React and D3.js, improving real-time monitoring capabilities for healthcare professionals.
* Integrated Apache Kafka for stream processing to handle high-throughput event-driven data pipelines.
* Automated serverless workflows using AWS Lambda, allowing real-time processing of medical device telemetry.
* Used Boto3 to interact with AWS services such as S3 for secure file storage and DynamoDB for dynamic database operations.
* Leveraged Kubernetes and Docker to scale real-time systems for managing medical device telemetry.
* Conducted load testing to optimize performance on GPU/CPU-intensive workloads, improving response times for real-time data processing.
* Enhanced CI/CD pipelines for GPU-based application deployments, ensuring efficient updates and version management.
* Implemented role-based access control (RBAC) and secure authentication mechanisms to comply with HIPAA regulations.
* Utilized Docker and Kubernetes for containerization and orchestration, ensuring seamless deployments across environments.
* Improved system performance by optimizing Python scripts and refactoring legacy code for better maintainability.
* Built real-time monitoring dashboards using AWS CloudWatch and custom Python scripts, ensuring system reliability.
* Collaborated with cross-functional teams to ensure timely delivery of features and improvements in the medical data ecosystem.
* Integrated Terraform for infrastructure as code and used Prometheus for system monitoring and alerting.
* Improved API response times by introducing Redis caching, reducing latency for real-time medical data retrieval.
* Implemented a custom data ingestion pipeline using AWS Kinesis and Lambda to capture medical device data streams and process them in near real-time.
* Developed an automated system for billing and compliance data verification using Python, reducing human errors and ensuring faster processing times.

**Environment:** Python, Django, EC2, S3, React, D3.js, AWS Lambda, AWS Glue, AWS Kinesis, DynamoDB, CloudWatch, Glue, Apache Kafka, Docker, Kubernetes, Prometheus, Pytest, Terraform, Boto3, Git, Jira, Redis

**Software Developer**

**Adtalem Global Education, Chicago, IL Dec 2021 to May 2023**

**Responsibilities:**

* Developed a modular frontend architecture using Angular and TypeScript, employing state management with NgRx to streamline complex user interactions.
* Enhanced cross-browser compatibility and accessibility compliance by adhering to WCAG 2.1 guidelines in web application development.
* Developed RESTful APIs with Flask to streamline access to educational data for administrators and students.
* Created robust ETL pipelines to extract, clean, and transform data from multiple LMS platforms into a centralized data warehouse.
* Implemented Kafka to handle real-time messaging and event-driven data synchronization between microservices.
* Automated data processing workflows with Azure Functions and scheduled event triggers, reducing manual intervention.
* Leveraged **Azure SDK for Python** for seamless interaction with **Azure** services like Blob Storage, Virtual Machines, and Cosmos DB for data storage and processing.
* Improved database query performance with advanced SQL techniques and Python-based ORM tools like SQLAlchemy.
* Built CI/CD pipelines with Jenkins and GitHub Actions, ensuring automated testing and streamlined deployments.
* Enhanced user experience by integrating backend APIs with React-based frontend applications.
* Designed scalable cloud solutions using Azure to support a rapidly growing user base.
* Monitored system logs and errors using **Azure Monitor** and Elasticsearch, improving reliability and uptime .
* Migrated legacy monolithic systems to a modular microservices architecture, enabling scalability and maintainability.
* Conducted peer code reviews and implemented best practices for Python development across the team.
* Implemented data encryption strategies using **Azure Key Vault** and Python to comply with data privacy standards, ensuring secure data transmission across services.
* Designed containerized microservices using Docker and deployed them on Kubernetes for seamless scaling.
* Conducted regression tests to ensure backward compatibility of APIs and systems after feature enhancements.
* Automated backend testing workflows using PyTest integrated into CI/CD pipelines.

**Environment**: Python, Django, **Microsoft Azure, Azure Functions**, Cosmos DB, **Azure Data Factory**, Apache Kafka, Docker, Kubernetes, **Azure Resource Manager (ARM),** PyCharm, Jenkins, Git, Jira, Agile, Windows, Flask, React, TypeScript, Angular, SQLAlchemy, OAuth2.0, **Azure Key Vault, Azure Monitor.**

**Software Developer**

**Arvest Bank,** **Lowell, AR Dec 2019 to Oct 2021**

**Responsibilities:**

* Engineered secure Python-based backend services for processing high-volume banking transactions.
* Developed and orchestrated ETL pipelines with Apache Airflow, enabling efficient scheduling and monitoring of data workflows.
* Utilized Apache Kafka to build a fault-tolerant real-time data streaming platform for transaction alerts and fraud detection.
* Automated financial data ingestion workflows using AWS Step Functions and Batch, reducing manual intervention.
* Designed secure Python-based banking APIs with integrated JWT authentication, ensuring compliance with PCI DSS standards.
* Optimized real-time transaction dashboards by leveraging React's Context API and reducing API payload size using selective data fetching.
* Implemented database interaction layers using Hibernate, fine-tuning query performance with custom criteria queries and caching strategies.
* Developed frontend modules using SASS with reusable mixins and variables, improving consistency and maintainability across banking portals.
* Designed scalable APIs using Flask and FastAPI to integrate with third-party financial services and applications.
* Leveraged Boto3 to interact with AWS resources, automating operations like storage, backups, and key management.
* Implemented ElasticSearch for log aggregation and searching, significantly improving system monitoring and debugging.
* Created a pipeline for real-time customer feedback analysis using Python and AWS Comprehend for sentiment analysis.
* Built CI/CD pipelines with GitLab and Terraform, enabling seamless infrastructure provisioning and application deployment.
* Developed Python APIs and backend services with a focus on high-performance data streaming for financial systems.
* Optimized system performance through GPU-based data processing in ETL pipelines using PySpark and Apache Kafka.
* Automated test suites using PyTest and integrated them into the CI/CD pipeline to maintain code quality.
* Led team-wide workshops on modern Python development practices, promoting better coding standards and efficiency.
* Optimized transaction data processing by leveraging AWS SQS to manage data queues, improving throughput and reducing latency.
* Introduced dynamic scaling with AWS Auto Scaling and Lambda to efficiently handle transaction peaks during high-demand periods.

**Environment:** Python, Flask, FastAPI, AWS Step Functions, AWS Batch, S3, DynamoDB, KMS, Apache Kafka, Boto3, ElasticSearch, PyTest, PySpark, GitLab, Terraform, Jira, Agile, Windows, Hibernate, React, Context API, JWT, SASS, AWS SQS, AWS Auto Scaling, AWS Lambda, AWS Comprehend.

**Software Developer**

**Lister Technologies, Chennai, India Oct 2017 to Nov 2019**

**Responsibilities:**

* Designed and developed enterprise-grade applications using Python focusing on reliability and scalability.
* Built ETL pipelines to streamline data ingestion and transformation processes for business intelligence systems.
* Integrated Apache Kafka for real-time data streaming and asynchronous communication between services.
* Deployed and managed web applications on Apache Tomcat, ensuring stable performance in production environments.
* Implemented RESTful APIs using Flask for seamless data integration between systems.
* Used Maven for managing project dependencies and automating build workflows.
* Created reusable Python modules for data processing, improving code maintainability and reducing redundancy.
* Migrated legacy pipelines to Python, implementing unit testing and load testing to ensure system reliability and efficiency.
* Orchestrated microservices on GCP Kubernetes**,** improving deployment reliability for fault-tolerant systems. Collaborated with front-end teams to develop dynamic web applications using JavaScript and HTML5.
* Optimized SQL queries and performed database normalization to enhance application performance.
* Implemented automated unit testing using PyUnit and JUnit to ensure code reliability.
* Participated in code reviews and pair programming to maintain high-quality code standards.
* Migrated legacy Java applications to modern frameworks, enhancing maintainability and performance.
* Refined legacy data processing pipelines by implementing PySpark for distributed data transformations, significantly reducing processing time.
* Designed and implemented a fault-tolerant system using Kafka and Python for seamless message processing across multiple services.
* Developed high-performance backend systems using FlaskAPI and integrated Elasticsearch for efficient text-based search functionalities.
* Built interactive data visualization dashboards using D3.js and JavaScript, incorporating real-time updates via WebSocket integration.

**Environment:** Python, Java, Flask, Apache Kafka, Apache Tomcat, Maven, JUnit, PyUnit, SQL, PyCharm, Git, Jira, Agile, GCP Kubernetes, D3.js, WebSocket, Elasticsearch, PySpark, HTML5, JavaScript.

**Technical Skills**

|  |  |
| --- | --- |
| Languages | Python 3.x/2.7, Java, SQL, JavaScript, TypeScript |
| Frontend Technologies | React, Angular, HTML5, CSS3, jQuery |
| Backend Technologies | Django, Flask, Node.js, FastAPI, Angular, AJAX |
| Testing Frameworks | PyTest, JUint, Selenium |
| Databases  Python Libraries: | MySQL, Oracle, MS SQL, DynamoDB, PostgreSQL, DynamoDB, Amazon Redshift  TensorFlow, PyTorch, Scikit-Learn, Keras, Pandas, NumPy, Matplotlib, OpenCV, SciPy, PySpark, Boto3 |
| Web/App Servers | Apache Tomcat, JBoss, Nginx, Microsoft IIS |
| Version Control Tools | Git, SVN, CVS, VSS, Mercurial |
| Development Methodologies | SDLC, Agile/SCRUM, Kanban, Waterfall, DevOps |
| Cloud Technologies | Amazon Web Services (AWS), Google Cloud Platform (GCP), Azure Cloud. |
| Big Data Technologies  Monitoring & Logging | Hadoop, Spark, Kafka, Hive, HBase  Prometheus, Helm, Grafana, ELK Stack (Elastisearch, Logstash, Kibana), New Relic, Datadog, CloudWatch |
| CI/CD Tools | Docker, Kubernetes, Jenkins, Terraform, Ansible, SonarQube, Github actions, CircleCI |
| Caching Technologies | Redis, Memcached |
| IDEs: | PyUnit, PyTest, JUnit |