Aishwarya

SENIOR PYTHON DEVELOPER

aishwaryar484@gmail.com |

(612)54 8423

Professional Profile

Around 7 years of professional IT experience in Analysis, design, development, testing, enhancement, support, and Implementation of various web, stand-alone, client-server enterprise applications using Python, Django in various domains, and Database Development.

Career History

CIBC ,CHICAGO- (SEP 2021 – PRESENT)
Sr. Python developer

- Deployed and configured Elasticsearch, Log stash, and Kibana (ELK) for log analytics, full-text search, and application monitoring in integration with AWS Lambda and Cloud Watch. Established DevOps culture based on Docker and Kubernetes tools.
- Used MongoDB to store data in JSON format and developed and tested many features of a dashboard using Python, Bootstrap, CSS, and JavaScript.
- Developed data transition programs from DynamoDB to AWS Redshift (ETL Process) using AWS Lambda by creating functions in Python for certain events based on use cases.
- Worked on Batch processing and Real-time data processing on Spark Streaming using Lambda architecture along with Spark SQL UDFs and Hive UDFs and Spark accumulators and broadcast variables.
- Develop consumer-based features and applications using Python and Django in test-driven Development and pair-based programming along with Using Celery with RabbitMQ, MySQL, Django, and Flask to create a distributed worker framework.
- Responsible for writing/reviewing server-side code using Spring JDBC and DAO module of spring for executing stored procedures and SQL queries.
- Front-end web development using HTML/CSS, JQuery, and Bootstrap as well as back-end development using GoLang and SQL.
- Implemented a 'serverless' architecture using API Gateway, Lambda, and DynamoDB and deployed AWS Lambda code from Amazon S3 buckets.
- Involved in the development of Web Services using REST for sending and getting data from the external interface in XML and JSON format also used Ajax and JQuery for transmitting JSON data objects between frontend and controllers.
- Used Data Frame API in Scala for converting the distributed collection of data organized into named columns, developing predictive analytics using Apache Spark Scala APIs
- Deployed and managed NoSQL database Cassandra to process large amount of mobile usage data also implemented Cassandra cluster in Amazon Cloud EC2 using DataStax AMI.
- Implemented Spark using Scala and utilizing Data frames and Spark SQL API for faster processing of data.
- Built APIs to gather data from various web applications using Python for authentications and implementation of OAuth 2.0 also improvised OAuth security at client server side to reduce fraudulent attempts.
- Assigned security groups, network ACLs, Internet Gateways, NAT instances, and Route tables to ensure a secure zone for organizations in the AWS public cloud.
- Created applications using ReactJS, Restful APIs, and GraphQL Databases including MySQL, PostgreSQL, Oracle, and SQL Server databases, and hosted these apps in Docker containers on ECS or within the AWS environment.
- Responsible for Continuous Integration (CI) and Continuous Delivery (CD) process implementation using Jenkins along with Shell scripts to automate routine jobs.
- Converted existing AWS infrastructure to server-less architecture (AWS Lambda, Kinesis) deployed via Apache Lib Cloud, Terraform, and AWS Cloud formation.
- Worked on AppDynamics to remove complexity and solve problems more quickly with proactive, end-to-end performance monitoring.
- Develop consumer-based features and applications using Python, Django, HTML, and Test Driven Development (TDD).
- Worked on data cleaning and reshaping, generated segmented subsets using NumPy and Pandas in Python
- Used NOSQL database Amazon dynamo DB to store data of reporting Application with exposure to CI/CD tools - Jenkins for Continuous Integration, Ansible for continuous deployment.
- Worked on REST Web services as well as Node.js REST framework for backend services, used Mongo DB (NoSQL) for database services.
- Built a Full-Service Catalog System that has a full workflow using Elasticsearch, Logstash, Kibana, Kinesis, and CloudWatch.
- Implement Nifi to Spark streaming directly without using Kafka internally to provide various options to clients.

- Developed Open stack API to Integrate with Amazon EC2 cloud-based architecture in AWS, including creating machine Images and working on AWS, High Availability Practices, and deploying backup/restore infrastructure.

CIGNA- (NOV 2018 - JUN 2021)
Python Developer

- Developed Restful microservices using Flask and Django and deployed on AWS servers using EBS and EC2.
- Program queries using Python/SQL scripts to get data from different databases such as ERP, OPLA, etc., and, build REST API and NoSQL queries using Flask web framework and MongoDB.
- Involved in creating a user authentication module for web-based reports using PHP/HTML for front-end development and Oracle as RDBMS.
- The project is built on Python, Django, and Beautiful Soup and uses other tools like jQuery, JavaScript, bootstrap, MySQL, HTML, and CSS.
- Microservices architecture development using Python and Docker on an Ubuntu Linux platform using HTTP/REST interfaces with deployment into a multi-node Kubernetes environment.
- Used Spark (RDD) and Python for processing and transformation of data and integration with popular NoSQL, and Oracle databases for huge volumes of data.
- Developed independent services for configurations (Kafka, System, and Database), producer manifest, and data manifest using Spring Boot Rest API.
- Used MySQL as backend database and MySQLdb of python as database connector to interact with MySQL server.
- Wrote scripts in Python for extracting data from HTML files using Python Library Beautiful Soup.
- Developed cross-browser and multi-browser compatible web pages using HTML5, CSS3, and JavaScript.
- Manage, develop, and design a dashboard control panel for customers and Administrators using Django, Oracle DB, Postgre SQL, and VMWare API calls.
- Used Jenkins pipelines to drive all microservices builds out to the Docker registry and then deployed to Kubernetes, Created Pods and managed using Kubernetes
- Used JQuery to make the frontend components interact with the JavaScript functions to add dynamism to the web pages on the client side.
- Used the AWS -CLI to suspend on Aws Lambda function and used AWS CLI to automate the backup of ephemeral data stores to S3 buckets EBS.
- Used Ajax and JQuery for transmitting JSON data objects between frontend and controllers
- Developed Spark Programs using Scala and Java APIs and performed transformations and actions on RDDs.
- Designed and implemented Spark jobs to support distributed data processing.
- Experienced in writing Spark Applications in Scala and Python (PySpark).
- Involved in creating a reusable component using React JS and Redux JS for DOM Manipulation.
- Created EC2 instances and implemented large multi-node Hadoop clusters in the AWS cloud from scratch using automated scripts such as Terraform.

CAPITAL ONE- (OCT 2016 - OCT 2018)
Python developer.

- Responsible for manipulating HTML5, and CSS3 in jQuery and also provided dynamic functionality using AJAX, XML, and JSON.
- Used React JS in components like JSX, creating React components, Virtual DOM, React Props, Lifecycle methods, and working with React States and Events.
- Maintained the existing code base developed in the Struts, spring, and Hibernate framework by incorporating new features and doing bug fixes.
- Improved file streaming functionality using Camel file stream component.
- Used Sqoop to import the data onto Cassandra tables from different relational databases like Oracle, MySQL and Designed Column families in Cassandra performed data transformations, and then exported the transformed data to Cassandra as per the business requirement.
- Used Spark streaming to receive real-time data from Kafka and store the stream data to HDFS using Scala and NoSQL databases such as HBase and Cassandra.
- Implemented CI/CD with GitHub, Travis deploys as microservices Docker containers to Docker Hub
- Generated an API Reference guide to help application developers for creating the software applications to store data using DDN S3 (compatible with Amazon S3 API) or DDN OpenStack Swift API calls.
- Developed data transition programs from DynamoDB to AWS Redshift (ETL Process) using AWS Lambda by creating functions in Python for certain events based on use cases.

- Worked on AWS SQS to consume the data from S3 buckets. Imported the data from different sources like AWS S3 and local file systems into Spark RDD.
- Worked in container-based technologies like Docker, Kubernetes, and Open shift and used Jenkins pipelines to drive all microservices builds out to the Docker registry and then deployed to Kubernetes
- Involved in converting Hive/SQL queries into Spark transformations using Spark RDD, Scala, and Python.
- Developed various Python scripts to find vulnerabilities with SQL Queries by doing SQL injection, permission checks, and performance analysis.
- Builds logic for automated data extraction and validation. Features for the dashboard were developed and tested using CSS, JavaScript, Django, and Bootstrap.
- Created PHP/MySQL back-end for data entry from Flash and worked in tandem with the Flash developer to obtain the correct data through query string