****

Chakravarthi

Auburn Hills, MI | Phone: +1-469-722-7347 | Email: raj@abntechcorp.com

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Summary**:

* Have overall 14+ Years of IT experience in which 9+ years of experience in Python specializing in **FastAPI**, Kafka, **AWS** **EKS**, and DocumentDB/**MongoDB**. Skilled in **REST** **APIs**, asynchronous microservices, unit tests, **BDD**, and implementing design patterns, clean code, and SOLID principles.
* Skilled in utilizing FastAPI for building fast, scalable **APIs**, leveraging its modern features to optimize performance and ease of use.
* Proficient in **Kafka** for real-time data streaming and processing, enabling robust and reliable data pipelines in complex environments.
* Experienced with **AWS** **EKS** for deploying and managing containerized applications, ensuring seamless integration and scalability in cloud environments.
* Strong knowledge of DocumentDB and **MongoDB** for **NoSQL** database solutions, providing flexible and scalable data storage options.
* Hands-on experience with **ElastiCache** and **Redis** for caching solutions, enhancing application performance and reducing latency.
* Experience in working with **Lambda**, **AWS Connec**t, **Amazon Lex**, **AWS CLI**, **AWS CDK** with Python, **Serverless** and **PaaS** toolkit, **EMR**, **Kinesis**, **S3, RDS** and other core AWS services.
* Work focused on owning a data source and developing ETL solutions for the source using PySpark
* Experienced in implementing **AVRO**-based data pipelines using **Python** and **Apache** **Spark**, providing robust and scalable solutions for big data processing tasks.
* Responsive in Converting existing **Terraform modules** that had version conflicts to utilize CloudFormation templates during **deployments**, worked with Terraform to create stacks in **AWS**, and updated the Terraform scripts in Golang based on the requirement on regular basis.
* Implemented a CI/CD pipeline using Azure DevOps (VSTS, TFS) in both cloud and on-premises with GIT, MS Build, Docker, Maven along with Jenkins plugins
* Successfully developed and deployed machine learning models using Python, enhancing data-driven decision-making and predictive analytics capabilities.
* Hands on experience in developing **REST** **APIs** and asynchronous **microservices**, ensuring efficient communication and processing in distributed systems.
* Experienced in designing and implementing **Kafka**-**python** security features (**SSL**/**TLS**, **SASL**) for data encryption and authentication.
* Involved in configuration for Web apps/Function apps, V-net integration, HCM, IAM, Application gateway, AppInsights, Active directory, Azure Key Vault, Encryption and Security on AZURE using ARM templates and Power shell script
* Created Pyspark frame to bring data from DB2 to Amazon S3 and Provide guidance to development team working on PySpark as ETL platform
* Strong working knowledge in developing Restful webservices and Micro Services using **flask, Django, API Gateway and Datadog**
* Familiar with Behavior-Driven Development (**BDD**) for improved collaboration, ensuring that development aligns with business requirements and user stories.
* Hands-on experience working with various Relational Database Management Systems (**RDBMS**) like **MySQL**, Microsoft **SQL** Server, and **Oracle** & non-relational databases (**NoSQL**) like **MongoDB** and **Cassandra**.
* Skilled in **Kafka**-**python** for **Kafka** integration, enabling seamless data flow and processing in distributed environments.
* Developed ETL processes to extract, transform, and load data from various sources into Snowflake data warehouse, ensuring data quality and consistency.
* Knowledgeable in **AVRO** for efficient data serialization, ensuring compact, fast, and versatile data exchange.
* Managed distributed task queues with Celery and RabbitMQ, optimizing resource allocation and enhancing task execution reliability.
* Developed entire frontend and back-end modules using Python on Django Web Framework.
* Strong background in monitoring and tracing applications, using tools and frameworks to maintain high availability and performance.
* Hands on experience in enhancing application reliability by handling exceptions and errors in asyncio coroutines.
* Experience in deploying and maintaining applications on **AWS**, leveraging cloud services to enhance application scalability and resilience.
* Implemented a CI/CD pipeline using Azure DevOps (VSTS, TFS) in both cloud and on-premises with GIT, MS Build, Docker, Maven along with Jenkins plugins
* Expertise in developing scalable microservices architecture, breaking down monolithic applications into modular, manageable components.
* Skilled in **Docker** and **Kubernetes** for container orchestration, ensuring efficient **deployment**, **scaling**, and management of applications.
* Proficient in setting up and managing **CI/CD** **pipelines**, automating **build**, **test**, and deployment processes to streamline development **workflows**.
* Adept at conducting rigorous code reviews to ensure high code quality, fostering a culture of continuous improvement and best practices.
* Developed and executed White box test cases using Python, Unitest/Pytest/Robot framework & PyCharm/Ride

**Technical Skills:**

* **Programming Languages:** Python 2.5-2.7/3.4-3.6, C, SQL and PL/SQL.
* **Web Technologies:** HTML, CSS, DOM, SAX, JavaScript, jQuery, AJAX, XML, AngularJS.
* **Frameworks:** Django, PyJamas, Flask, Python, Pyramid, CherryPy
* **Version Control:** MVC, SVN, CVS, Git, GitHub.
* **Front End:** Angular, jQuery, Ajax, ExtJS, Node JS, React JS, JSON, Bootstrap, CSS, HTML
* **Analytic Tools:** JMP PRO, SAS, Tableau, UCI NET, Node XL
* **Application servers:** Apache Tomcat, JBoss
* **Databases:** Oracle, My SQL, PostgreSQL, Apache Cassandra
* **IDE’s/ Development Tools:** NetBeans, Spyder, Eclipse, PyCharm, PyScripter, PyStudio, Atom and Sublime Text.
* **Operating Systems:** Windows 98/2000/XP/7/8/10, Red Hat Linux 4.x/5.x/6.x, Ubuntu, and MacOS.
* **Protocols:** TCP/IP, HTTP/HTTPS, SOAP, SMTP
* **Testing Tools:** PyTest, Testify, Selenium, Bugzilla and JIRA.

**Work Experience:**

**Client: State street ,Auburn Hills, MI. Nov 2022 – Present**

**Sr. Python Developer**

**Responsibilities:**

* Managed Kubernetes secrets and ConfigMaps to securely store sensitive information and application configurations on **AWS** **EKS**.
* Implemented automated code analysis tools to enforce Clean Code standards and identify areas for improvement in **Python** codebases.
* Developed **Kafka**-**python** connectors for relational databases (e.g., **PostgreSQL**, **MySQL**) for real-time data synchronization.
* Integrated **MongoDB** with Flask and **Django** frameworks for seamless web application development and data management.
* Developed Lambda functions in TypeScript to handle event-driven tasks and automate business processes.
* Developed entire frontend and backend modules using Python on Django Web Framework.
* Developed **API** gateways using **FastAPI** for routing and load balancing across **microservices**, ensuring reliability and scalability.
* Created Pyspark frame to bring data from DB2 to Amazon S3 and Provide guidance to development team working on PySpark as ETL platform
* Implemented data pre processing, feature engineering, and data **pipelines** to support machine learning workflows.
* Developed custom **Python** 3+ scripts for data analytics and visualization using **pandas**, **matplotlib**, and **seaborn**.
* Participated in architectural discussions to design **JSON**-based solutions for scalable data systems.
* Added support for **Amazon** **AWS** **S3** and **RDS** to host static/media files and the database into **Amazon** **Cloud**.
* Developed unit and integration tests for **Kafka**-**python** consumers and producers, ensuring message reliability and data integrity.
* Implemented Celery and RabbitMQ to handle asynchronous tasks, improving the scalability and efficiency of data processing operations.
* Integrated Terraform with continuous integration and continuous deployment (CI/CD) pipelines, streamlining the deployment and lifecycle management of infrastructure resources.
* Used **MongoDB** Compass and **Robo** **3T** for visual management and querying of **MongoDB** databases in **Python**.
* Designed and implemented **Kafka**-**python**-based event-driven architecture for real-time data processing in **Python** applications.
* Identified improvements to enhance CI/CD Developed framework for converting existing PowerCenter mappings and to PySpark Jobs.
* Created Pyspark frame to bring data from DB2 to Amazon S3 and Provide guidance to development team working on PySpark as ETL platform.
* Analyzed test results from **BDD** scenarios to identify and prioritize improvements in application functionality and user experience.
* Used NumPy to create tabular timestamp for east timestamp data manipulation and retrieval.
* Integrated **FastAPI** with **CI/CD** **pipelines** using tools like **Jenkins** or **GitLab** **CI**, automating build, **test**, and deployment processes.
* Implemented **MongoDB** change streams in **Python** to capture and react to real-time data changes, enabling reactive and event-driven application architectures.
* Implemented data migration strategies using **AVRO** and **Python**, ensuring smooth and reliable data transitions between different systems and formats.
* **NodeJS** is been used for developing I/O intensive web applications like, single-page application Consuming **NodeJS** **RestFul** **API** services.
* Used **AVRO** with Google BigQuery for scalable data analysis, enabling efficient and cost-effective querying of large datasets.
* Created reusable Terraform modules and templates to standardize infrastructure provisioning and promote code reusability and consistency.
* Good experience in developing web applications and implementing Model View Control (MVC) architecture using server-side applications like Django
* Worked with Docker components like creating Docker images compose Docker registry and handling multiple images primarily for middleware installations and configurations
* Successfully implemented Domain Driven Design architecture with **CQRS** and integrated and Microservices using Redis to provide view data all hosted in an **AWS** environment.
* Collaborated with frontend developers to design reactive UIs that interact seamlessly with Asynchronous **Microservices**.
* **Building**/**Maintaining** **Docker** container clusters managed by Kubernetes **Linux**, **Bash**, **GIT**, **Docker**, on **AWS** (**Amazon** **Web** **Services**).
* Integrated **AVRO** with **Apache** **Flink** for real-time data processing, enabling efficient and scalable streaming data analytics.
* Leveraged Terraform and AWS CloudFormation to automate the provisioning and configuration of Amazon EKS clusters and associated infrastructure.
* Designed and implemented CI/CD pipelines for deploying containerized applications to Amazon EKS using tools such as Jenkins, GitLab CI/CD, or AWS Code Pipeline.

**Environment** : AWS, CI/CD, Bash, GIT , Docker, Linux, Webservices, Apache, MVC, Nodejs, API, MongoDB, Jenkin, Kafka, Pandas, matplotlib, PostgreSQL, MySQL, Django, PySpark, Microservices, Terraform.

**Client: Adobe,Dallas,Texas Jan2019 to June2022**

**Sr Python Developer**

* + Responsible for analyzing various cross-functional, multi-platform applications systems enforcing Python best practices and provide guidance in making long term architectural design decisions.
  + Coded model level validation using Python programming with emphasis in web security.
  + Handled Business logics by backend Python programming to achieve optimal results.
  + Implemented jQuery to improve the overall usability of the website interface and AJAX functionality to speed up web application.
  + Added support for Amazon AWS S3 and RDS to host static/media files and the database into Amazon Cloud.
  + Integrated Terraform with continuous integration and continuous deployment (CI/CD) pipelines, streamlining the deployment and lifecycle management of infrastructure resources.
  + Developed views and templates in Django to create a user-friendly website interface.
  + Constructed RESTful API's using PHP, Node js, and MySQL/PostgreSQL.
  + Configured Django to manage URLs and application parameters.
  + Designed and developed the application using Python Django and Angular Framework.
  + Used Django APIs for database access.
  + Utilized Kubernetes and Docker for the runtime environment of the Cl/CD system to build, test deploy.
  + Utilized Futures and Spring Framework to make Asynchronous calls to the application.
  + Developed web services using Restful technology to support JSON and XML.
  + Created new connections through applications for better access to MySQL database and involved in writing SQL & PLSQL - Stored procedures, functions, sequences, triggers, cursors, object types etc.
  + Implemented database access using Django ORM (object-related mapper).
  + Worked on reading and writing multiple data formats like JSON using PySpark
  + Writing REST APIs, as part of developing web-based applications for insurance premium calculations, using Django's REST framework. O Architected and developed Python and Django for the backend development and front-end application using React and PostgreSQL for database.
  + Built micro services using API gateway.
  + Developed Templates for AWS infrastructure as a code using Terraform to build staging and production environments.
  + Migrated an existing on-premises application to AWS. Used aws services for small data sets processing and storage, experienced in Maintaining the Hadoop cluster on AWS EMR.
  + Experienced in managing and accessing Hadoop log records.
  + Worked on AWS DMS for Migrating databases.
  + Developed microservices on boarding tools leveraging Python and Jenkins allowing for easy creation and maintenance of build jobs and Kubernetes deploy and services.
  + Developed microservice on boarding tools leveraging Python and Jenkins allowing for easy creation and maintenance of build jobs and Kubernetes deploy and services.
  + Designed MongoDB pipeline using pymongo for routing/storing of email bounce handling records.
  + Experience in building data dictionary, functions and synonyms for NoSQL (Elasticsearch).
  + Design, development and implementation of performant ETL pipelines using python API (PySpark) of Apache Spark, AWS glue on AWS EMR.
  + Proficient in writing Test plans and Test cases for both manual and automated testing scenarios.
  + Creating different S3 buckets and write Lambda to move files from S3 to Redshift.
  + Heavily leveraged Python's APIs for creating serialization libraries for encoding data in XML/JSON formats.
  + Used Pandas API to put the data as time series and tabular format for east timestamp data manipulation and retrieval.
  + Created reusable Terraform modules and templates to standardize infrastructure provisioning and promote code reusability and consistency
  + Perform software deployment to cloud-based infrastructure (AWS).
  + Developed and tested many features in an agile environment using Python (PyTest), Django, HTML5, CSS, JavaScript, and Bootstrap.
  + Created Athena and integrated with AWS Glue to fully manage ET L service that can categorize the data.
  + Automation for execution of tests in batch using shell scripting.
  + Correspondingly involved in writing REST APIs using Django frame work for data exchange and business logic implementation.
  + Designed and maintained databases using Python and developed Python-based API (RESTful Web Service) using Flask, SQL Alchemy and PostgreSQL.
  + Created PyTest cases for unit testing. Development of test framework using the Python.
  + Used GIT as Source control repositories with regular code reviews and pull/merge requests.

**Environment:** Python,Django,Bootstrap, Pandas, MYSQL, Restful API's, Linux, Servlec, Athena, SCRUM, Agile, Heroku, JIRA, GIT, HTML5,CSS, Terraform, AWS, XML, Mysql, Pyspark,Json,

**Client: Wipro – Hyderabad, India July 2015 to Dec 2018**

**Sr Python Developer**

**Responsibilities:**

* + - * Extensively used **Python** / **Django** Framework for developing applications.
* Designed and developed embedded systems for real-time applications, demonstrating a deep understanding of hardware-software interaction
* Developed ETL processes using Redshift Spectrum and COPY commands to load data from various sources such as S3, DynamoDB, and RDS into Redshift tables
* Responsible for analyzing various cross-functional, multi-platform applications systems enforcing **Python** best practices, and providing guidance in making long-term architectural design decisions.
* Configured and maintained Jenkins to implement the CI/CD process and integrated the tool with Ant and Maven to schedule the builds.
* Designed and implemented serverless workflows using AWS Step Functions to orchestrate and coordinate microservices and AWS Lambda functions.
* Extensively worked on **Push** and **Pull** architectures, **Ansible**, **Docker** and **Kubernetes**
* Analyze the **CI** / **CD** process and create new more secure and efficient microservices processes and tools
* Designed and managed **AWS** infrastructure using Infrastructure as Code (**IaC**) tools like **AWS** CloudFormation and **Terraform**, ensuring consistent and secure deployments.
* Used **Aws** **Glue** **Studio** to access a variety of data sources that are stored in **AWS** services
* Consumed the data from **Kafka** using **Apache** **Spark**.
* Troubleshot and debugged **Lambda** functions and **Glue** jobs to ensure smooth operations.
* Reviewed the Logical Model with Application Developers, **ETL** Team, **DBAs**, and **Testing** Team to provide information about the Data Model and business requirements.
* Used Celery with RabbitMQ, MySQL, Django, and Flask to create a distributed worker framework.
* Utilized the Python Unit test framework, for all Python applications and used Django Database Fast APIs to access database objects.
* Building/Maintaining Docker container clusters managed by Kubernetes, Linux, Bash, GIT, Docker, on GCP. Utilized Kubernetes and Docker for the runtime environment of the CI/CD system to build, test deploy.
* Configured and maintained Jenkins to implement the CI/CD process and integrated the tool with Ant and Maven to schedule the builds.
* Utilized Terraform to provision and manage AWS resources, including Lambda functions, IAM roles, and VPC configurations, ensuring infrastructure as code (IaC) principles
* Work focused on owning a data source and developing ETL solutions for the source using PySpark
* Building/Maintaining Docker container clusters managed by Kubernetes, Linux, Bash, GIT, Docker, on GCP . Utilized Kubernetes and Docker for the runtime environment of the CI/CD system to build, test deploy
* Deployed machine learning models into production environments, ensuring scalability and reliability.
* Integrated Scala modules seamlessly with existing **Python** codebases, demonstrating interoperability between the two languages within a single application.
* Integrated machine learning models with existing applications and systems through APIs or other methods.
* Design, develop, and deploy **UNIX** shell scripts to automate basic daily administration scripts.
* Used **Django** **API** for database access and word on databases like **MySql**, **Postgres**
* Participated in all the stages of the software development lifecycle (**SDLC**) like design, testing development, and implementation.
* Developed Templates for AWS infrastructure as a code using Terraform to build staging and production environments.
* Experience with Docker, deployment of application inside software containers.
* Kept up to date with advancements in machine learning techniques, frameworks, and libraries.
* Extensively worked with spark Data frames for ingesting data from flat files into **RDDs** to transform unstructured data and structured data.

**Environment**: Postgress, Mysql, Django, API, Python, Docker, Kubernetes, GCP, Unix, Pyspark, ETL, Scala, Terraform, Kafka, Apache, Lambda, Ansible, Glue, Linux, UNIX, Bash.

**e-Zest Solutions, India Feb 2013 to Jul 2015**

**Python Developer**

**Responsibilities:**

* Versioning is handled through **GitHub** and code reviews are done through **PR** and automated **GitHub** actions.
* Worked as **Python** backend developer using **Flask** **API** and consumed heavy data in the frontend and enhanced the existing applications.
* Hands on experience and in provisioning virtual clusters under Amazon Web Service (**AWS**) cloud which includes services like **Elastic** compute cloud (**EC2**), **S3**, and **EMR**.
* Used the **AWS** -**CLI** to suspend on Aws Lambda function used **AWS** **CLI** to automate backup of ephemeral data stores to **S3** buckets **EBS**.
* Used Python Boto 3 to configure the services **AWS** glue, **EC2**, **S3**.
* Connected to **Azure** database from **python** for the data operations using **MySQL** connector.
* Developed a self-serve reporting tool using **Python** which enabled users to generate the reports
* Developed **ETL** **python** scripts for ingestion **pipelines** which run on **AWS** infrastructure setup of **EMR**, **S3,** **Redshift** and **Lambda**.
* Created **AWS** **Glue** job using **PySpark** for file-based data processing
* Integrated Glue with AWS Lambda and Step Functions to orchestrate complex data processing workflows and trigger Glue jobs based on events or schedules
* Generated graphical reports using **python** package **NumPy**.
* Have insightful experience in **Pyspark**, **Splunk** and data processing by combining **Apache** **spark** and **Python** Framework.
* Used **MongoDB** to stored data in **JSON** format and developed and tested many features of dashboard using Python.
* Implemented infrastructure as code using Terraform to provision and manage AWS resources, reducing manual configurations and ensuring consistency across environments.
* Designed and implemented data workflows using Apache Airflow, automating complex ETL processes and ensuring timely data delivery.
* Experience with Cloud Technologies: **AWS** (**Lambda**, **S3**, **cloud** watch rules, **Redshift**, **EC2**, **EBS**, **IAM**, **API** Gateway, cloud formation),
* Running Glue Jobs to Get **Athena** Tables from **S3** Location.
* Worked on **AWS** Data **pipeline** to configure data loads from **S3** to **Redshift**.
* Worked on a project to automate **SAS** applications relating to calculation of earnings at risk. Provided design, development, implementation, documentation, validation and testing support for the business process, system requirements and functionalities needed.
* Used Pandas library for statistics Analysis. Used NumPy for Numerical analysis for Insurance premium and managed large datasets using Panda data frames and **MySQL**
* Used Python Boto 3 to configure the services **AWS** **glue**, **EC2**, **S3**.
* Developed and program **Python** scripts of automation procedure with provided **API** and **JSON** data to connect and including developer testing activities, like unit testing, regression testing, integration testing
* Migrated data from Transactional source systems to Redshift data warehouse using spark and **AWS** **EMR**.
* Used **AWS** **EMR** to transform and move large amounts of data into and out of other AWS data stores and databases, such as Amazon Simple Storage Service (**Amazon** **S3**) and **Amazon** **DynamoDB**.
* Developed data ingestion modules to data into various layers in S3, Redshift using **AWS** **Kinesis**, **AWS** **Lambda**, **AWS** **Glue** and **AWS** Step Functions.
* Setup the CloudWatch monitoring dashboards with **SNS** notification configurations on various performance metrics retrieved from various **AWS** resources.
* Used **Jenkins** for **CI**/**CD** deployments.

**Environment** : AWS, Lambda, Kinesis, API, JSON, Redshift, CI/CD, Glue, EC2,S3, SAS, MongoDB, Numpy, Pyspark, EMR, MySql, EBS, Panda, Apache, Airflow, Splunk.

**BNP Paribas, India. Mar 2009 to Jan 2013**

**Python Developer**

**Responsibilities:**

* Used Python, Django and Pyramid frameworks to develop applications and responsible for both back-end programming and front-end functionality.
* Built a new Cl pipeline and performed testing and deployment automation with Docker, Jenkins.
* Redesigned and redeployed the Nagios implementation. Nodes automatically added/removed themselves via Puppet.
* Used AJAX for transmitting JSON data objects between frontend and controllers and wrote Python scripts to parse XML and JSON documents and load the data in database.
* Implemented RESTful Web-Services for sending and receiving the data between multiple systems.
* Implemented business logic using Python and used HTML, CSS, react, JavaScript and Implemented a Continuous Integration & Delivery (CI/CD) pipeline with Docker, Jenkins and GitHub.
* Familiar with NumPy, Pandas and Matplotlib.
* Developed application service components and configured beans using Spring IOC, creation of Hibernate mapping files and generation of database schema.
* Supported Apache Tomcat Web server on Linux Platform.
* Implemented and enhanced CRUD operations for the applications using the MVT (Model View Template) architecture of Django framework and Python conducting code reviews.
* Extensive experience in developing Microservices using Spring Boot and followed domain driven design O Involved in running Hadoop streaming jobs to process terabytes of XML format data.
* Developed microservice on boarding tools leveraging Python and Jenkins allowing for easy creation and maintenance of build jobs and Kubernetes deploy and services.
* Documented the simulation steps for debug using Shell scripting.
* Used SQL Alchemy to operate across various databases.
* Extensively worked on developing UI components using Angular JS and JSON to interact Restful web services.
* Designed and developed data management system using PostgreSQL and MySQL.
* Set up and maintain Cl infrastructure to streamline build process, unit tests, functional tests, integration tests and artifacts deployments.
* Built micro services using API gateway.
* Developed build and deployment scripts using Ant and Maven as build tools in Jenkins to move from one environment to other environments.
* Developed the notification service by posting the JSON request in AWS API Gateway.
* Wrote with object-oriented Python, Flask, SQL, Beautiful Soup, httplib2, Jinja2, HTML/CSS, Bootstrap, jQuery, Linux, Sublime Text, GIT. O Extensively worked on Application servers like WebLogic and Apache Tomcat.
* Used Ajax for transmitting JSON data objects between frontend and controllers and Worked on configuring TCP/IP, network interface, assigning static routes and hostnames.
* Developed many features for dashboard using Ruby on Rails, Bootstrap, CSS, react and JavaScript (J2EE)
* Used SVC version control system for maintain version of code & Used agile methodology.

**Environment:** Python 3.3/3.2, Django, Jenkins, Apache, XML, MySQL, MongoDB, REST, Shell scripting, svc.