**Rucha**

**Email ID: rucha9093@gmail.com**

**Mobile number:** ‬‬ **+1 (978) 533-1118**

**Professional Summary:**

* Having **10+** years of experience as a **Data Engineer** in AWS & GCP Cloud Environments.
* **Big Data/Hadoop/PySpark** technology development.
* Experience in developing applications that perform large-scale distributed data processing using big data ecosystems tools like **HDFS, YARN, Sqoop, Flume, Kafka, MapReduce, Pig, Hive, Spark, PySpark SQL, PySpark Streaming, HBase, Cassandra, MongoDB, Oozie, and AWS.**
* Good functional experience in using various Hadoop distributions like **Hortonworks, Cloudera,** and **EMR.**
* Good understanding of using data ingestion tools- such as **Kafka**, **Sqoop, and Flume**.
* Experienced in performing in-memory real-time data processing using **Apache Spark.**
* Good experience in developing multiple **Kafka** Producers and Consumers as per business requirements.
* Working with **GCP** cloud using GCP Cloud storage, Dataproc, Data Flow, Big Query, Cloud Composer, Cloud Pub/Sub.
* Expert in working with cloud **PUB/SUB** to replicate data in real-time from a source system to **GCP Big Query**.
* Extensively worked on **PySpark** components like PySpark SQL and PySpark Streaming.
* Experience setting up **AWS Data Platform** - AWS CloudFormation, Development End Points, AWS Glue, EMR and Jupyter/Sage maker Notebooks, Redshift, S3, and EC2 instances.
* Developed quality code adhering to coding standards and best practices.
* Experience in migrating **map reduce programs** into PySpark RDD transformations, and actions to improve performance.
* Experience in deploying and managing the multi-node development and production Hadoop cluster with different **Hadoop components** (HIVE, PIG, SQOOP, OOZIE, FLUME, HCATALOG, HBASE, ZOOKEEPER) using Hortonworks Ambari.
* Good knowledge of **GCP** service accounts, billing projects, authorized views, datasets, GCS buckets, and gsutil commands.
* Extensive working experience with data warehousing technologies such as **HIVE**.
* Expertise in writing **Hive** and **Pig** queries for data analysis to meet business requirements.
* Extensively worked on **Hive** and **Sqoop** for sourcing and transformations.
* Extensive work experience in creating UDFs, and UDAFs in **Pig** and **Hive**.
* Ample knowledge of data architecture including data ingestion pipeline design, Hadoop/Spark architecture, data modeling, data mining, machine learning, and advanced data processing.
* Experience working with **NoSQL databases** like **Cassandra, HBase** and **DynamoDB** and developed real-time read/write access to very large datasets via HBase.
* Hands-on experience with Ad-hoc queries, Indexing, Replication, Load balancing, and Aggregation in **MongoDB**.
* Expertise in relational databases like **MySQL, SQL Server, DB2, and Oracle**.
* Real-time exposure to **AWS** command line interface, and **AWS** data pipeline.
* Extensively worked on **AWS** services such as **EC2** instance, **S3**, **EMR**, **Cloud Formation, Cloud Watch, and Lambda.**
* Expertise in writing map reduction programs in Python for data extraction, transformation, and aggregation from various file formats such as **XML, JSON, CSV, Avro, and Parquet**.
* Good knowledge in understanding the security requirements for Hadoop and integration with **Kerberos** authentication and authorization infrastructure.
* Implemented **ETL** operations using the Big Data platform.
* Experience in configuring the **Zookeeper** to coordinate the servers in clusters and to maintain the data consistency.
* Involved in identifying job dependencies to design workflow for **Oozie & YARN** resource management.
* Experienced in working with **Python ORM Libraries**.
* Expertise in establishing database connections in Python by configuring packages like **JDBC**, and **MySQL-Python**.
* Strong experience in Data Warehousing ETL concepts using **Informatica**, and **Talend.**
* Experience in using bug tracking and ticketing systems such as **JIRA**, and **Remedy.**
* Highly involved in all facets of SDLC using **Waterfall and Agile** **Scrum** methodologies.

**Technical Skills:**

|  |  |
| --- | --- |
| **Big Data/ Hadoop** | HDFS, MapReduce, Pig, Hive, PySpark, Kafka, Flume, Sqoop, Impala, Oozie, Zookeeper, YARN, Hue, Data bricks, Airflow |
| **Hadoop Distributions** | Cloudera (CDH4, CDH5), Hortonworks, EMR |
| **Programming Languages** | Python |
| **Database/NoSQL** | HBase, Cassandra, MongoDB, MySQL, Oracle, Synapse, Big Query |
| **Cloud Services** | AWS, GCP |
| **DevOps tool** | Jenkins |
| **Frameworks** | Spring, Hibernate, Struts |
| **Scheduling** | Airflow, Azure Data Factory |
| **Application Servers** | Apache Tomcat, Web Sphere, WebLogic, JBoss |
| **ETL Tools** | Informatica, Azure Data Factory |
| **Dashboard** | Splunk |

**Client: Client: UiPath, Houston, TX Sep 2021 - Till Date**

**Role: Senior Big Data Engineer**

**About Project:** Worked with the Data Engineering team to build a data warehouse and reporting layer in AWS for the Machine Learning team through daily feeds from On-Premises for analytics and insights. Collaborated closely with stakeholders, providing transparent communication and ensuring alignment with project goals throughout migration. Documented migration process, best practices, and guidelines, facilitating knowledge transfer for efficient post-migration management.

**Responsibilities:**

* Worked on **AWS Data pipeline** to configure data loads from S3 into Redshift.
* Using AWS **Redshift** extracted, transformed, and loaded data from various heterogeneous data sources and destinations.
* Created Tables, Stored Procedures, and extracted data using **T-SQL** for business users whenever required.
* Scheduled jobs in **Apache** **Airflow** for automating the ingestion process into the data lake.
* Used Jenkins to develop, maintain, and support the **Continuous Integration** framework.
* Performed **data analysis** and designed, created, and maintained large, complex logical **data models** and physical data models, and **metadata repositories**.
* Performed data ingestion into a data lake (**S3**) and used **AWS Glue** to expose the data to **Redshift.**
* Led end-to-end migration strategy, seamlessly transitioning RPA workflows from **AWS to GCP with minimal disruption.**
* Conducted meticulous assessment of **AWS infrastructure**, optimizing **GCP** resources for substantial cost savings without compromising performance.
* Automated resource provisioning using **Terraform scripts**, ensuring consistency and efficiency throughout the migration process.
* Oversaw data migration to **Google Cloud Storage**, implementing transformations to maintain data integrity and format compatibility. Mapped **AWS services to GCP counterparts**, strategically selecting services for seamless integration with RPA project requirements.
* Integrated RPA workflows with **GCP's machine learning services**, leveraging advanced **AI and ML features for enhanced automation.**
* Implemented rigorous security measures during migration, ensuring compliance with industry standards and protecting sensitive RPA data.
* Executed a phased migration approach, prioritizing critical RPA processes and maintaining a hybrid cloud setup for controlled transition.
* Established comprehensive monitoring solutions in **GCP**, proactively addressing performance bottlenecks for optimal **RPA** process efficiency.
* Achieved substantial cost savings in **GCP,** optimizing resource allocation and improving scalability for long-term RPA project needs.
* Configured **EMR** cluster for data ingestion and transformed the data in Redshift.
* Extract, transform, and load (ETL) data from multiple federated data sources (JSON, relational database, etc.) with Data Frames in **Spark**.
* Worked on writing, testing, and debugging **SQL** code for data transformations.
* Orchestrated multiple ETL jobs using AWS step functions and **lambda**, also used **AWS Glue** for loading and preparing data Analytics for customers.
* Worked on **AWS Lambda** to run servers without managing them and to trigger run code by S3 and SNS.
* Developed data transition programs from **Dynamo DB** to AWS **Redshift** (ETL Process) using AWS Lambda by creating functions in Python for certain events based on use cases.
* Implemented the **AWS cloud computing** platform by using RDS, Python, Dynamo DB, S3, and Redshift.
* Designed SQL, SSIS, and Python-based batch and real-time **ETL pipelines** to extract data from transactional and operational databases and load the data into data warehouses.
* Developed scripts using **Jenkin**s with the integration of the **Git** repository for the build, testing, code review, and deployment.
* Worked on CI/CD solution, using **Git**, **Jenkins**, **Docker,** and **Kubernetes** to set up and configure big data architecture on the AWS cloud platform.
* Worked in Developing **Spark** applications using **Spark - SQL** in **Data bricks** for data extraction, transformation, and aggregation from multiple file formats for analyzing& transforming the data to uncover insights into customer usage patterns.
* Worked with various formats of files like delimited **text files, click stream log files, Apache log files, Avro files, JSON files, and XML File**s. Mastered in using different columnar file formats like **RC, ORC, and Parquet formats**.
* Developed Scala scripts, and UDFs involving both Data frames and RDDs using **Spark SQL** for aggregation, queries, and writing data back into the OLTP system directly or through Sqoop.
* Developed **AWS Glue jobs** for automated ETL execution using PySpark jobs. Used Glue **crawler** for schema extraction and to update the catalog table.
* Collected data using Spark Streaming from **AWS S3** bucket in near-real-time and performs necessary Transformations and Aggregations on the fly to build the common learner data model and persistence the data in **HDFS**.
* Responsible for loading and transforming huge sets of **structured, semi-structured, and unstructured data.**
* Used **AWS EMR** clusters for creating **Hadoop** and **Spark** clusters. These clusters are used for submitting and executing **Python** applications in production.
* Designed and developed end-to-end **ETL processing** from **Oracle** to **AWS** using **Amazon S3, EMR, and Spark.**
* Written **SQL Scripts** and **PL/SQL** Scripts to extract data from the Database to meet business requirements and for Testing Purposes.
* Utilized **Spark SQL API** in PySpark to extract and load data and perform SQL queries.
* Worked on developing a PySpark script to **encrypt** the raw data by using **hashing algorithms** concepts on client-specified columns.
* Responsible for **designing, developing**, and testing the database and developing **Stored Procedures**, Views, and Triggers.
* Used **containerization platform** Docker to create containers, simplify configurations and enable isolated application deployment to multiple servers.
* Worked on Docker for **containerizing** various components of the application.
* Performed scaling up and scaling down **Kubernetes** pods and deployments.
* Used Kubernetes dashboard to deploy and edit containerized applications, check the **status** of **containerized applications**, and troubleshoot containerized applications.
* Improved **Kubernetes performance** by defining resource limits, using optimized and lightweight container images.
* Worked on the **tuning of SQL Queries** to bring down run time by working on indexes and execution plans.
* Performed **ETL activities** like running the jobs, extracting the data using necessary queries from the database, transforming, and uploading it into the data warehouse servers.

**Environment:** AWS, RedShift, Apache Airflow, AWS EMR clusters, Amazon S3, Spark, PL/SQL, Spark SQL API, Tableau, Power BI, Docker, Kubernetes, Python, Hadoop, SQL Server, Oracle 12/11g, Eclipse, IntelliJ IDE, Windows and Unix, Shell Scripting, JIRA.

**Client: SWIFT, Manassas, VA Jun 2020 – Aug 2021**

**Role: Data Engineer**

**About Project:** Worked with the Data Engineering team to build DataLake, Data warehouse and self-service ETL pipeline in AWS Cloud as per business requirements through real-time and daily feeds from On-Premises for digital marketing teams.

**Responsibilities:**

* Implemented **Spark** using **Python** and utilized Data frames and **Spark SQL API** for processing and querying data.
* Developed Spark Applications by using python Driver and Implemented **Apache Spark data processing project** to handle data from various **RDBMS** and Streaming sources.
* Worked on **Google cloud platform** (GCP) services like compute engine, cloud load balancing, cloud storage, cloud SQL, stack driver monitoring, and cloud deployment manager.
* Setup Alerting and monitoring using Stack driver in **GCP**.
* Design and implement large-scale distributed solutions in **AWS and GCP** clouds.
* Designed and implemented a scalable data architecture using **Google Cloud Storage** for secure and efficient storage of diverse financial datasets.
* Integrated **Google Cloud Pub/Sub** to enable real-time event streaming, enhancing the responsiveness of banking systems to transactions and events.
* Developed and deployed **Apache Spark** jobs on **Google Cloud Dataproc** for large-scale data processing, enabling advanced analytics and machine learning on banking data.
* Implemented **Infoworks** to automate **end-to-end data integration** and preparation processes, reducing manual efforts and accelerating the delivery of clean, enriched datasets for analysis.
* Utilized **Infoworks** to streamline **data processing**, contributing to a significant improvement in **analytics speed and efficiency**, resulting in quicker insights and more informed decision-making.
* Leveraged **Infoworks** in **cloud environments** to optimize data management, showcasing expertise in utilizing the platform's capabilities for **efficient and scalable** **cloud-based data solutions.**
* Implemented serverless data processing using **Google Cloud Dataflow**, orchestrating complex **ETL workflows** for regulatory reporting and compliance.
* Configured and managed **Google Cloud SQL** to support relational databases, ensuring high availability and reliability for critical banking applications.
* Established data partitioning and organization strategies in **Google Cloud Storage** to optimize data retrieval and storage performance.
* Utilized **BigQuery** to implement a robust data warehousing solution, enabling quick and ad-hoc querying of vast financial datasets for analytics and reporting.
* Optimized **SQL** queries and fine-tuned **BigQuery** performance, significantly reducing query response times for **banking** stakeholders.
* Orchestrated data integration pipelines connecting various banking applications with **Google Cloud services**, ensuring seamless data flow across the ecosystem.
* Implemented event-driven architectures using **Pub/Sub**, enhancing data consistency, and maintaining real-time visibility into banking activities.
* Developed and maintained **Spark** workflows for complex financial computations, improving overall data processing efficiency.
* Established database replication and failover strategies in **Cloud SQL** for high availability and disaster recovery.
* Implemented security measures, including encryption, access controls, and auditing, to protect sensitive financial data stored in **GCP services.**
* Leveraged **Google Cloud Dataflow** for efficient ETL operations, enhancing data transformation capabilities for regulatory reporting and compliance.
* Collaborated with cross-functional teams to understand and align banking project requirements with **GCP capabilities**.
* Created comprehensive documentation for system architectures, configurations, and integration points, facilitating knowledge transfer within the team.
* Optimized storage costs by implementing data lifecycle policies and automated data tiering in **Google Cloud Storage.**
* Implemented backup and disaster recovery strategies using **Google Cloud Storage,** ensuring minimal downtime and data loss.
* Configured and managed **Cloud SQL** to support multi-region deployment, enhancing resilience and redundancy for banking databases.
* Developed APIs and connectors to facilitate real-time data exchange between banking systems and **Google Cloud Storage.**
* Conducted performance testing and implemented optimizations for **Spark workflows,** meeting stringent performance requirements for the banking project.
* Established monitoring solutions for **Google Cloud Storage, Pub/Sub**, and other services, enabling proactive issue identification and resolution.
* Worked with various unstructured data sources like parquet, JSON, CSV, etc. using **Spark.**
* Developed preprocessing jobs using spark data frames to flatten JSON files.
* Experienced in working real-time streaming with **Kafka** as a data pipeline using spark streaming module.
* Good Knowledge of **Kubernetes architecture** like scheduler, pods, nodes, and cataleptic. database.
* Created **airflow jobs** with **snowflake** operators to create fact and dimension tables in the snowflake warehouse.
* Connected **snowflake** to **Tableau** to create data reporting dashboards.

**Environment**: Python, Spark, GCP Cloud, Google Cloud Storage, Pub/sub, Big Query, Kafka, JSON, GitHub, LINUX, Flask, Nginx, REST, CI-CD, Kubernetes, Snowflake, Airflow, Tableau, Docker.

**Client: UnitedHealth Group, Minnetonka, MN Oct 2017 – Nov 2019**

**Role: Data Engineer**

**About Project:** Worked with the Global data engineering team to build a data transmission framework that moves data from multiple RDBMS sources to Hadoop and analyzes the transmissions by building tableau dashboards.

**Responsibilities:**

* Involved in meetings and release, working closely with my teammates and managers.
* Developed Hadoop technologies including **HDFS**, **MapReduce2**, **YARN**, **Hive**, **HBase**, **Sqoop**, **Spark Streaming,** and **RabbitMQ**
* Translated, loaded, and streamed disparate data sets in multiple formats/sources including **Avro**, **JSON** delivered by **Kafka** queue, **RabbitMQ**, **Flume,** etc.
* Translated functional and technical requirements into detailed programs running on Hadoop **MapReduce** and **Spark**
* Migrated traditional database code to distributed system code (mainly **HiveQL**)
* Migrated data between **RDBMS** and **HDFS/Hive** with **Sqoop**
* Used **HBase** for scalable storage and fast query.
* Involved in application performance tuning and troubleshooting.
* Involved in **Partitions**, and **bucketing** concepts in Hive and designed both Managed and External tables in Hive for optimized performance.
* Implemented scalable data pipelines in **Spark (Scala)** to perform data transfer, aggregation, and transformation mining and passed the data to **S3 and Redshift**.
* Selected and generated data into CSV files and stored them in **AWS S3** by using **AWS EC2** and then structured and stored them in **AWS Redshift**.
* Created **Terraform scripts** for EC2 instances, Elastic Load balancers, and S3 buckets on AWS.
* Managed different infrastructure resources, like physical machines, VMs, and even Docker containers using **Terraform**.
* Used **Tableau** for visualization to generate the business report.
* Created **Oozie** coordinated workflow to execute Sqoop jobs.
* Expertise in Object-Oriented Design (OOD) and end-to-end software development experience working on **Scala coding** and implementing mathematical models in **Spark Analytics**.
* Software translating the business requirements into Use Cases and Diagrams conducting reviews of Codes and test cases analyzing change requests enhancements managing release plans for business apps
* Reviewed **HDFS** usage and system design for future scalability and fault tolerance.
* Implemented a **CI/CD** pipeline using **Jenkins, and Airflow** for Containers from Docker and Kubernetes.
* Managed and reviewed Hadoop log files using **Spark** to identify issues when a job fails.
* Supported mapping editors to build maps of product and transactional environments.

**Environment:** Hadoop, HBase, MapReduce, Spark, Scala, AWS, S3, RedShift, SQS, SNS, Terraform, PostgreSQL, Hue, Flume, Sqoop, Kafka, RabbitMQ, Hive, Jenkins, Airflow, GitLab

**Client: Yashoda Hospitals, Hyd, Ind Jul 2015 - Sep 2017**

**Role: Python Developer**

**About Project:** As Part of Hadoop production support team to ensure all the pipelines running as expected, fix them in case of any issues and automate the manual reporting tasks

**Responsibilities:**

* Responsible for providing support and analysis to the existing **MS SQL server**.
* Created Database objects like **Stored Procedures**, Triggers, Views, Rules etc. Created tables and provide constraints and user-defined rules in the database.
* Managed large datasets using Panda data frames and **MySQL.**
* Wrote and executed various **MYSQL** database queries from python using **Python-MySQL** connector and MySQL dB package.
* Involved in building database **Models, APIs**, and Views utilizing Python technologies to build web-based applications.
* Built database Models, Views, and APIs using **Python** for interactive web-based solutions.
* Used **Python** scripts to update the content in the database and manipulate files.
* Used Django **APIs** to access the database.
* Collaborate with Product Management and User Experience experts regarding product definition, schedule, scope, and project-related decisions.
* Used many regular expressions in order to match the pattern with the existing one.
* Skilled in using collections in **Python** for manipulating and looping through different user-defined objects.
* Work with a team of developers on **python** applications for RISK management.

**Environment**: Python, HTML, CSS, Bootstrap, MongoDB, Linux, APIs, Git, MySQL,PostgreSQL Elastic search, T-SQL, Beautiful soup, Teradata, Git, GitHub, Linux, and MAC OSX.

**Client: Dr. Reddy's Laboratories, Hyd, Ind Sep 2013 - Jun 2015**

**Role: SQL Developer**

**About Project:** Our team responsibility is to build batch and streaming pipelines to source data from various sources.

**Responsibilities:**

* Created consumption views on top of metrics to reduce the running time for complex queries.
* Compare the data in a leaf-level process from various databases when data transformation or data loading takes place. I need to analyze and look into the data quality when these types of loads are done (To look for any data loss or data corruption).
* As a part of Data Migration, wrote many SQL Scripts for Mismatch of data and worked on loading the history data from **Teradata SQL** to snowflake.
* Developed SQL scripts to Upload, Retrieve, Manipulate, and handle sensitive   data (National Provider Identifier Data I.e. Name, Address, SSN, Phone No) in **Teradata**, **SQL Server** Management Studio, and **Snowflake** Databases for the Project
* Implemented Defect Tracking process using **the JIRA** tool by assigning bugs to Development Team
* Involved in Functional Testing, Integration testing, Regression Testing, Smoke testing, and performance Testing. Tested **Hadoop Map Reduce** developed in **python, pig, Hive**
* Created Metric tables, End-user views in Snowflake to feed data for **Tableau** refresh.
* Generated Custom **SQL** to verify the dependency for the Daily, Weekly, and Monthly jobs.
* Using **Nebula Metadata**, registered Business and Technical Datasets for corresponding **SQL scripts**
* Created performance dashboards in **Tableau/ Excel / PowerPoint** for the key stakeholders
* Incorporated predictive modeling (rule engine) to evaluate the Customer/Seller health score using python scripts, performed computations, and integrated with the Tableau viz.
* Developed spark code and **spark-SQL**/streaming for faster testing and processing of data.
* Analyzed marketing campaigns from various perspectives including CTR, conversion rates, seasonal/geographical trends, search queries, landing page, conversion funnel, quality score, competitors, distribution channel, etc. to achieve maximum ROI for clients.
* Evaluated the traffic and performance of Daily deals **PLA** ads and compare those items with non-daily deal items to see the possibility of increasing **ROI**. suggested improvements and modify existing BI components (**Reports, Stored Procedures**)
* Experienced in working with spark ecosystem using **Spark SQL** and Scala queries on different formats like text files and CSV files.

**Environment:** Hadoop, MapReduce, Hive, Apache Spark, Sqoop Snowflake, Nebula, Teradata, SQL Server, Python, Pig, GitHub, Teradata, Tableau, MS Excel, MS PowerPoint.