Template - Requirements Specifications Document

# **Introduction -**

## **Purpose -**The purpose of this SRS is to define the requirements for developing and deploying data pipelines for a Health Care insurance company. These pipelines will be used to analyze competitor data sourced through web scraping and third-party providers. The analysis aims to track customer behavior and conditions, allowing the company to customize insurance offers, calculate royalties for existing customers, and ultimately enhance revenue by making data-driven business decisions

## **Intended Audience and Use** - This SRS is intended for the following stakeholders within the organization:

## **Developers/data engineers:** To design, develop, and implement the data pipelines and related components.

## **Testers:** To verify the accuracy, performance, and reliability of the data pipelines and outputs.

## **Project Managers:** To oversee the project, ensure alignment with business objectives, and manage timelines.

## **Data Analysts:** To utilize the outputs from the data pipelines for generating insights and making strategic business recommendations.

## **Business Stakeholders:** To understand the expected outcomes and ensure that the project aligns with overall business strategies.

## **Product Scope –**

The product aims to provide a robust, scalable solution for analyzing large datasets from various sources to:

* Gain insights into customer behaviors and healthcare claims.
* Optimize marketing strategies by tailoring offers to specific customer segments.
* Increase revenue by effectively targeting customers with personalized insurance policies.
* Enhance operational efficiency by analyzing the performance of healthcare providers and insurance policy groups.
* The solution aligns with the company’s broader business goals of improving customer satisfaction, boosting profitability, and gaining a competitive edge in the market.

## **Definitions and Acronyms –**

* + **Big Data Ecosystem:** A set of tools, technologies, and methodologies for processing and analyzing large, complex datasets.
  + **SRS (Software Requirements Specification):** A document that outlines the requirements for a software system, including its functions, constraints, and objectives.
  + **ETL (Extract, Transform, Load):** A process that involves extracting data from various sources, transforming it into a usable format, and loading it into a database or data warehouse.
  + **Claim:** A request for payment submitted by a policyholder or healthcare provider to an insurer for services rendered.
  + **Subscriber:** An individual who is enrolled in a health insurance plan.
  + **Premium:** The amount of money paid by a subscriber to an insurance company, typically on a monthly basis, for health insurance coverage.
  + **Cashless Insurance:** A type of health insurance that allows direct settlement of the hospital bill by the insurer without requiring the patient to pay upfront.
  + **Subgroup:** A specific category or division within a larger group of insurance policies, often tailored to particular demographics or health conditions.
  + **AWS S3 (Amazon Simple Storage Service):** A cloud storage service provided by Amazon Web Services for storing data.
  + **AWS Redshift:** A cloud data warehouse service by Amazon Web Services used for analyzing large datasets.
  + **Databricks:** A data processing platform built on Apache Spark, used for data engineering, data science, and machine learning tasks.
  + **AWS EMR (Elastic MapReduce) Studio:** An integrated development environment for building and debugging big data applications on AWS.
  + **PySpark:** A Python API for Apache Spark, used for large-scale data processing.

**Overall Description**

**2.1 Product Overview**

The product we are going to build is a comprehensive data pipeline and analysis system for a Health Care insurance company. This system will process and analyze vast amounts of data received from competitors, through web scraping, and third-party sources. The primary purpose of this product is to enable the insurance company to gain deep insights into customer behavior, track the performance of various insurance policies, and tailor offerings to maximize customer engagement and revenue. By leveraging Big Data technologies, the product will allow the company to make informed, data-driven decisions that enhance its competitive edge in the market.

**2.2 Why This Product is Needed**

The Health Care insurance company is facing challenges in understanding customer needs and behaviors, which is crucial for optimizing business strategies and increasing revenue. The data collected from competitors and third-party sources is vast and complex, making it difficult to analyze and extract actionable insights using traditional methods. By developing this product, the company can:

* **Enhance Customer Understanding:** Analyze customer data to identify trends, preferences, and behaviors that can inform personalized marketing strategies.
* **Increase Revenue:** By customizing offers and targeting the right customers with the right policies, the company can drive more sales and improve retention rates.
* **Optimize Operations:** Identify which insurance groups and hospitals perform best, allowing the company to streamline its offerings and improve operational efficiency.
* **Improve Decision-Making:** Enable data-driven decision-making at all levels of the organization by providing accurate, timely insights.

**2.3 Target Audience**

This product is designed for the internal teams of the Health Care insurance company, including:

* **Data Analysts:** To explore the processed data, identify trends, and generate insights that can be used to inform business strategies.
* **Marketing Teams:** To create targeted marketing campaigns based on customer behavior analysis and customize insurance offers.
* **Business Strategists:** To make informed decisions about product offerings, pricing, and customer engagement strategies.
* **IT and Data Engineering Teams:** To manage and maintain the data pipelines, ensure data integrity, and integrate the product with existing systems.

**2.4 Is It a New Product?**

Yes, this is a new product specifically designed to address the unique challenges faced by the Health Care insurance company in analyzing competitor data and understanding customer behavior. While it is a new product, it will integrate with the company’s existing data infrastructure, including AWS S3, AWS Redshift, and Databricks, to ensure seamless data processing and analysis.

**2.5 Integration with Other Products**

This product will integrate with several existing tools and systems within the company’s infrastructure:

* **AWS S3:** The product will store raw data in AWS S3, which serves as the initial landing zone for all data ingested from various sources.
* **AWS Redshift:** Processed and cleaned data will be stored in AWS Redshift, where it can be queried and analyzed for insights.
* **Databricks:** The data processing and transformation will be carried out using Databricks, which provides the computational power and flexibility needed to handle large datasets efficiently.
* **AWS EMR Studio:** For managing the big data workflows and scaling processing tasks as needed.
* **Jira and GitHub:** For project management, code versioning, and collaboration among development and testing teams.

## User Needs -

Understanding the various users of this data pipeline and analysis system is crucial for ensuring that the product meets the needs of all stakeholders within the Health Care insurance company. The following outlines the key users of the product and how they will interact with it.

**3.1 Data Analysts**

* **Needs:** Data Analysts need to access, query, and analyze large datasets to uncover trends, patterns, and insights that can drive business decisions. They require tools that allow for efficient data exploration, the ability to run complex queries, and generate visualizations to communicate findings effectively.
* **How They Will Use the Product:**
  + **Data Exploration:** Analysts will use the data stored in AWS Redshift to run SQL queries and explore various dimensions of the data, such as claims frequency, subscriber demographics, and hospital performance.
  + **Trend Analysis:** They will analyze historical data to identify trends in customer behavior, disease prevalence, and claim rejection rates.
  + **Reporting:** Analysts will generate reports and visualizations based on the processed data, which will be used by business strategists and marketing teams to make informed decisions.

**3.2 Marketing Teams**

* **Needs:** Marketing teams need insights into customer behavior and preferences to design targeted marketing campaigns and customize insurance offers. They also need to track the effectiveness of these campaigns and adjust strategies based on real-time data.
* **How They Will Use the Product:**
  + **Customer Segmentation:** The product will allow marketing teams to segment customers based on various factors such as age, location, health conditions, and previous policy subscriptions.
  + **Campaign Design:** Based on the insights derived from the data, marketing teams will create and deploy customized offers and promotions targeted at specific customer groups.
  + **Performance Monitoring:** The product will enable marketing teams to monitor the success of their campaigns by tracking customer responses and adjusting strategies accordingly.

**3.3 Business Strategists**

* **Needs:** Business strategists require comprehensive insights into the company’s performance, customer preferences, and market trends to make strategic decisions that drive growth and profitability. They need data-driven insights to inform decisions related to pricing, policy offerings, and customer engagement.
* **How They Will Use the Product:**
  + **Policy Optimization:** Strategists will analyze data on policy subscriptions, claims, and customer demographics to determine which policies are most profitable and where adjustments are needed.
  + **Revenue Forecasting:** The product will help strategists forecast revenue based on trends in customer behavior, claim rates, and market conditions.
  + **Competitor Analysis:** By analyzing data from competitors, strategists can identify opportunities to differentiate the company’s offerings and gain a competitive edge.

**3.4 IT and Data Engineering Teams**

* **Needs:** IT and Data Engineering teams are responsible for the implementation, maintenance, and scaling of the data pipelines and infrastructure. They need a reliable, scalable system that can handle large volumes of data efficiently and ensure data integrity throughout the process.
* **How They Will Use the Product:**
  + **Data Pipeline Management:** The teams will develop, deploy, and maintain the data pipelines using tools like PySpark and Databricks, ensuring that data is processed and moved efficiently from AWS S3 to AWS Redshift.
  + **Data Quality Assurance:** They will implement data cleaning and validation processes to ensure that the data is accurate, complete, and reliable.
  + **System Monitoring and Scaling:** IT and Data Engineering teams will use AWS EMR Studio and other monitoring tools to ensure that the system performs optimally and scales to meet demand.

**3.5 Testers**

* **Needs:** Testers need to ensure that the data pipelines, transformations, and resulting outputs are functioning correctly and meeting the specified requirements. They need to verify data accuracy, system performance, and adherence to business rules.
* **How They Will Use the Product:**
  + **Test Case Execution:** Testers will run test cases to validate each component of the data pipeline, from data ingestion to the final output in AWS Redshift.
  + **Performance Testing:** They will assess the performance of the data processing system to ensure it can handle large datasets without degradation in speed or accuracy.
  + **Bug Tracking and Reporting:** Testers will document any issues or discrepancies they find in Jira and work with developers to resolve them.

## Assumptions and Dependencies -

1. **Data Availability and Quality:**
   * It is assumed that the data received from competitors, whether through web scraping or third-party sources, will be available in a consistent format and quality. Any significant variation in data structure or quality could impact the accuracy and reliability of the analysis.
   * It is also assumed that the data will be sufficiently rich in detail to allow for the required analysis, such as identifying disease prevalence, customer demographics, and claims history.
2. **Technological Infrastructure:**
   * The project assumes the availability and proper configuration of cloud-based infrastructure, including AWS S3 for data storage, AWS Redshift for data warehousing, and AWS EMR Studio for big data processing. The system's scalability and performance are contingent on these technologies functioning as expected.
   * The use of PySpark and Databricks for data processing is based on the assumption that these tools will be compatible with the data formats and volumes handled in this project.
3. **Access to Necessary Tools and Services:**
   * It is assumed that all team members (developers, analysts, testers, etc.) will have the necessary access to tools like Jira, GitHub, and Databricks, as well as permissions to AWS services. Any delays in securing these tools could impact project timelines.
   * It is also assumed that the team has the necessary expertise and experience with these tools and services, reducing the need for extensive training or onboarding.
4. **Project Timelines:**
   * The project is based on a two-week sprint plan, with the first week dedicated to documentation and solution design and the second week to implementation and testing. It is assumed that the requirements will not change significantly during this period, allowing the team to adhere to the schedule.
   * The assumption is that the data cleaning, transformation, and loading processes will be completed within the planned time frame without encountering major technical issues.
5. **Stakeholder Involvement:**
   * It is assumed that key stakeholders, including business strategists, marketing teams, and IT personnel, will be available for regular feedback and approval during the project. This ensures that the product aligns with business goals and meets user needs.

**3.6.2 Dependencies**

1. **Third-Party Data Sources:**
   * The project depends on data from third-party sources and web scraping, which may be subject to changes in availability, format, or legal restrictions. Any disruption in these sources could impact the completeness and accuracy of the analysis.
   * The quality and frequency of data updates from third-party providers are crucial for maintaining the relevance and accuracy of the analysis.
2. **Cloud Service Reliability:**
   * The reliability and performance of AWS services (S3, Redshift, EMR Studio) are critical dependencies. Any downtime or performance degradation in these services could delay the project and impact data processing and analysis.
   * The project also depends on the integration and interoperability of these services with Databricks and PySpark.
3. **Team Collaboration Tools:**
   * The project relies on the effective use of Jira for task management and GitHub for version control and collaboration. Any issues with these tools, such as access problems or integration challenges, could disrupt the workflow and project coordination.
   * Dependency on PyCharm connected to a Docker container for the development environment assumes smooth integration and functioning of these setups for the team.
4. **Regulatory and Compliance Considerations:**
   * There may be dependencies on compliance with healthcare data regulations, such as HIPAA (Health Insurance Portability and Accountability Act) in the U.S. Any changes in regulatory requirements or interpretations could necessitate changes to the data handling and processing approach.
5. **User Training and Adoption:**
   * The successful deployment of the product depends on the end-users (analysts, marketing teams, etc.) being adequately trained and adopting the new system. If users are not comfortable with the new tools and processes, it could reduce the effectiveness of the system and its ability to deliver the expected insights.

# System Features and Requirements -Functional requirements specify what the system must do to meet the business needs of the Health Care insurance company. These requirements define the essential operations, data processing, and interactions that the system will perform.

1. **Data Ingestion and Storage:**
   * The system must ingest data from multiple sources, including web scraping, third-party data, and internal records. This data will be stored in AWS S3 in a designated folder named input-data.
   * The system must support the uploading and processing of different types of data, including patient information, claims data, subscriber details, and policy groupings.
2. **Data Cleaning:**
   * The system must clean the ingested data to ensure accuracy and reliability. This includes:
     + Identifying and counting null values in each dataset.
     + Replacing null values with "NA" where applicable.
     + Detecting and removing duplicate records from the datasets.
   * The system must clean data for specific datasets such as Patients, Subscribers, Claims, and Group\_Subgroup before loading it into the AWS Redshift data warehouse.
3. **Data Transformation and Loading:**
   * The system must transform the cleaned data into a format suitable for analysis and load it into corresponding Redshift tables. A schema design must be created to define the structure of these tables.
   * The system must partition and index the data in Redshift to optimize query performance.
4. **Data Analysis and Result Generation:**
   * The system must perform specific analyses as outlined in the project requirements, including:
     + Identifying the disease with the maximum number of claims.
     + Finding subscribers under 30 who have subscribed to any subgroup.
     + Determining which group has the maximum subgroups.
     + Analyzing hospital data to find which hospital serves the most patients.
     + Calculating the total number of rejected claims.
     + Identifying the city from which the most claims originate.
   * The results of these analyses must be stored in separate Redshift tables under a schema named Project-Output.
5. **Integration with External Tools:**
   * The system must integrate with Databricks for data processing and visualization. Visualizations of the analysis results will be created using Databricks.
   * The system must integrate with Jira for project management, where user stories and tasks for each use case will be tracked.
6. **Deployment and Version Control:**
   * The system must be deployable on AWS EMR or Databricks via GitHub. All source code, including data cleansing and result generation modules, must be maintained in a GitHub repository.
   * The system must support version control and collaboration via GitHub to ensure that all team members can contribute and track changes effectively.

## External Interface Requirements - External interface requirements define how the system will interact with other systems, tools, and users. These interfaces are crucial for the system's operation and integration with existing workflows.

1. **User Interface (UI):**
   * The system must provide a user-friendly interface for accessing and visualizing analysis results in Databricks. Users should be able to interact with visualizations and export results as needed.
   * The system must allow users to upload data files to AWS S3 and trigger data processing workflows.
2. **Hardware Interface:**
   * The system must be compatible with the cloud-based infrastructure provided by AWS, ensuring that all necessary hardware resources (such as storage and compute instances) are provisioned and managed automatically.
   * The system must also support the use of Docker containers in the development environment (e.g., PyCharm connected to Docker) for consistent and reproducible builds.
3. **Software Interface:**
   * The system must integrate seamlessly with AWS services such as S3, Redshift, and EMR Studio for data storage, processing, and analysis.
   * The system must be compatible with PySpark and Databricks for executing data processing and transformation tasks.
   * Integration with Jira for project management must allow the creation, tracking, and reporting of user stories and tasks related to the project.
4. **Communication Interface:**
   * The system must support secure communication between AWS services and Databricks, ensuring that data is transmitted and processed without risk of unauthorized access or data breaches.
   * Integration with GitHub must include secure access and communication protocols to manage code repositories and deployment pipelines.

## System Features -

System features are specific functions or components that the system must include to meet the functional requirements.

1. **Data Ingestion and Cleansing Pipeline:**
   * The system must include a pipeline that automatically ingests data from various sources, cleanses it, and loads it into Redshift.
   * The pipeline must handle different data formats and structures, ensuring consistency across all datasets.
2. **Analytical Reporting Engine:**
   * The system must feature an engine that executes predefined analytical queries on the cleaned data, generating reports and storing results in the Project-Output schema in Redshift.
   * The engine must be able to perform complex analyses, such as calculating the profitability of different policy groups or identifying trends in claims data.
3. **Visualization and Reporting Dashboard:**
   * The system must include a dashboard in Databricks for visualizing analysis results. This dashboard should allow users to filter, sort, and explore data interactively.
   * Users should be able to export visualizations and reports for further analysis or presentation.

## Nonfunctional Requirements - Nonfunctional requirements ensure that the system operates effectively and meets user expectations beyond basic functionality.

1. **Performance Requirements:**
   * The system must be capable of processing large datasets (potentially millions of records) within a reasonable time frame. Data ingestion, cleaning, and analysis tasks should complete within predefined SLAs.
   * The system must support parallel processing and scaling to handle increased data volumes and complex analytical queries.
2. **Safety Requirements:**
   * The system must include data validation and error-handling mechanisms to ensure that incorrect or corrupted data does not compromise the analysis.
   * Redundant data storage and backup procedures must be in place to prevent data loss in case of system failure.
3. **Security Requirements:**
   * The system must implement robust security measures, including encryption of data at rest and in transit, to protect sensitive healthcare information.
   * Access controls and user authentication must be enforced to ensure that only authorized personnel can access the data and system functionalities.
4. **Usability Requirements:**
   * The system must be designed with usability in mind, ensuring that users can easily navigate the interface, execute tasks, and understand the results without extensive training.
   * Clear documentation and help resources must be provided to assist users in utilizing the system effectively.
5. **Scalability Requirements:**
   * The system must be scalable, allowing it to handle increasing data volumes, additional data sources, and more complex analytical requirements as the company grows.
   * The architecture must support the addition of new features and integrations without significant redesign.

## 