Healthcare - Requirements Specifications Document

# Introduction –resolving the healthcare system is one of the world's biggest challenges. Many healthcare systems are emerging, as well as many of them are collapsing. There may be various causes of dying in the healthcare system, such as healthcare policies, not providing well services, lack of leading knowledge, budget, etc. However, one of the most significant factors in the success of the healthcare system is an analysis of the data received from different departments and implementing the enhanced one, which will also help to compete in the market and provide the best experience for customers.

# This introduction is very important as it sets expectations that we will come back to throughout the SRS.

## Purpose –the primary purpose of this project is to design the ETL (Extraction Transformation and load) pipeline for the healthcare insurance company, which will be helpful in analyzing the data to improve customer services, add more revenue, and compete with other healthcare insurance companies.

## Define the purpose of these requirements here.

## Intended Audience and Use – the project is mainly intended for the health insurance company developer, testers, and project managers. They can use this project to customize insurance policies for various categories of people by analyzing the different data sources. The tester can test this project to check whether it gives the desired result or not. They can bring more investors by presenting the probabilities of outcomes.

## Define who in your organization will have access to the SRS and how they should use it. This may include developers, testers, and project managers.

## Product Scope - What are the benefits, objectives, and goals we intend to have for this product? This should relate to overall business goals, especially if teams outside of development will have access to the SRS.

## Definitions and Acronyms -Clearly define all key terms, acronyms, and abbreviations used in the SRS. This will help eliminate any ambiguity and ensure that all parties can easily understand the document.

# Overall Description – basically, we are designing an ETL pipeline or designing the software that the targeted audiences will use to integrate the health insurance company by analyzing the different parameters of that company’s policies such as it will help to identify the right insurances policies for customers by analyzing their historical data and help to customize the insurance policies for new customers. It also calculates the profit and loss, who claims the most, which city has more customers, etc., so It could play a vital role in skyrocketing the business. More likely, we are focusing on analyzing various groups of customers, financial analysis, and other related factors to healthcare companies hence direct audiences for our software are project managers, developers, product testers, and indirectly for health insurance buyers and investors. This is a new product, but you can integrate it with other healthcare insurance software to implement the result of our products.

# Your next step is to give a description of what you’re going to build. Why is this product needed? Who is it for? Is it a new product? Is it an add-on to a product you’ve already created? Is this going to integrate with another product? Understanding and getting your team aligned on the answers to these questions on the front end makes creating the product much easier and more efficient for everyone involved.

## User Needs - Describe who will use the product and how. Understanding the various users of the product and their needs is a critical part of the SRS writing process.

## Assumptions and Dependencies - What are we assuming will be true? Understating and laying out these assumptions ahead of time will help with headaches later. Are we assuming current technology? Are we basing this on a Windows framework? We need to take stock of these technical assumptions to better understand where our product might fail or not operate perfectly.

# System Features and Requirements -In order for your development team to meet the requirements properly, we must include as much detail as possible. This can feel overwhelming but becomes easier as you break down your requirements into categories.

## Functional Requirements – one of the mandatory factors is functional requirements for any product, so we are trying to make it more functional by adding more attributes and functions to analyze the historical data and new data. A few of them are; can find out the most profitable group, list the hospital which serves most customers, the city that claims the most, and calculate the average monthly premium for customers. It lists cancer patients whose age is less than 18 years, different groups of people with specific surgery, and patients who got charges greater than or equal to Rs. 50,000. Similarly, it would be helpful to figure out the subgroups which have the maximum subscribers, rejected claims, maximum claims due to which type of disease, etc.

## Functional requirements are essential to your product because, as the name implies, they provide some sort of functionality. Asking yourself questions such as “does this add to my tool’s functionality?” or “what function does this provide?” can help with this process. You may also have requirements that outline how your software will interact with other tools

## External Interface Requirements - You may also have requirements that outline how your software will interact with other tools There are several types of interfaces you may have requirements for, including:

### User

### Hardware

### Software

### Communications

## System Features - System features are a type of functional requirements. These are features that are required in order for a system to function.

## Nonfunctional Requirements - Nonfunctional requirements, which help ensure that a product will work the way users and other stakeholders expect it to, can be just as important as functional ones. These may include:

### Performance requirements

### Safety requirements

### Security requirements

### Usability requirements

### Scalability requirements

## 