**Chapter 3: Software Requirement Specifications**

This chapter provides a detailed explanation on general description, functional, non-functional requirements, assumptions, dependencies, external user interface, performance requirements and design constraints involved in the application.

A **software requirements specification** (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide.

Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers (in market-driven projects, these roles may be played by the marketing and development divisions) on what the software product is to do as well as what it is not expected to do. Software requirements specification permits a rigorous assessment of requirements before design can begin and reduces later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules. Used appropriately, software requirements specifications can help prevent software project failure. The software requirements specification document enlists enough and necessary requirements that are required for the project development. The requirement is achieved by detailed and continuous communications with the project team and customer till the completion of the software.

In this project, Smart Shelves technology enables the business owners to take more informed decisions , by keeping a real time tracking on what products are being taken from the store shelves and what is remaining. The real time analysis of data enables retailers to stock on more fast-moving products and reduce the number of slow moving items.

**3.1 Introduction**

* **Definitions, Acronyms and Abbreviations**

**RFID:** Radio frequency Identification

**API:** Application Programming Interface

**LOS:** Line of Sight

**RFID TAGS:** RFIDuses electromagnetic fields to automatically identify and track tags attached to objects. The tags contain electronically stored information.

**Passive Tags:** Passive tags collect energy from a nearby RFID reader's interrogating radio waves.

**Active Tags:** Active tags have a local power source such as a battery and may operate at hundreds of meters from the RFID reader.

**API:** API is a set of subroutine definitions, protocols, and tools for building application software. It is a set of clearly defined methods of communication between various software components.

**IOT:** Internet of Things

* **Overview**

‘Smart Shelves for retail stores’ is a IOT application that aims at implementation of Smart Shelves, the electronically connected shelves which can automatically keep a track of the inventory in a retail establishment. The Smart Shelves technology can be hugely beneficial for the retailers’ selling experience. The smart shelf concept will enable the inventory and store executives to refill stocks from stock room as they get depleted from the store shelf, intimate the buying group about the demand and Out of stock situation. Inform warehouse / Direct shipment vendor for immediate replenishment. Dissatisfaction of the customers due to out of stock items in retail physical stores is one of the biggest current challenges the retail industry is currently facing. The ‘Out of Stock’ Challenge in Physical Stores leads to customer dissatisfaction, which can possibly impact their future visits/ purchasing decisions with the same store. Failures in internal processes, resource constraints, data disconnects or poor data sync contribute to some of the reasons behind this instore experience failures.

* 1. **General Description**
* **Product Perspective**

1) The solution Eliminates the “out of stock challenge” in retail store i.e. when the customer buys all the products of category in a shelf the shelf remains empty when the new customers visit the retail store. Replenishment of the sold-out products in the shelf is a factor to enhance in-store shopping experience.

2) Monitoring the misplaced items in apparel retail store is another challenge. The users tend to misplace products in trial rooms and misplace the products in different shelves. Automated identification of misplaced apparels reduce manual monitoring of products.

3) Communication system to replenish the stock in-store/warehouse will be through SMS alerts and app notification

* **Product Functions**

1. Reduces the ‘out-of-stock’ products in shelves by automated identification on stock reduction
2. Locate the misplaced products
3. Better optimization of in-store operations
4. Control and monitor the usage of high value items

* **User Characteristics**

There are two categories of users.

**Store manager:** Store manager can monitor the dash board and identify the irregularities in the stock movement and initiate actions. Store manager can also add new items, edit the product details and attach the tag specification to a product.

**Stock clerks:** They get alerts on occurrence of product out of stock of item misplacement upon which they take necessary actions.

* **General constraints, Assumptions and Dependencies**

1. RFID tags must be attached with all the stocks in order to track the movement of the product.

2.Stock clerks must own mobile phone since the application works on the basis of alerts through the app or text messages.

3.RFID tags are conjoined with stock i.e. it is non-breakable.

* 1. **Functional Requirement**
* **Product Live View:** Description of overall product status

**Input:** Valid user ID and password as stock manager

**Process:** Initiate requests for the item details through API

**Output:** Dashboard of the shelves, products, description of product and count

* **Inventory Distribution module:** Assets count is displayed in terms of location

**Input:** Valid user ID and password as stock manager

**Process:** Filters applied on the basis of location

**Output:** heat map view of the locations and the products available in the location.

* **Asset Status module:** Asset information in terms of in-store, transit or warehouse

**Input:** Verify valid Asset ID

**Process:** Search based on Asset ID

**Output:** User view for requested asset

* **Stock Notification:** Out of stock notification

**Input:** Generated by the shelf

**Process:** Initiates the request for sms alert through API

**Output:** Sms alert to store keeper and stock clerk

* **Misplaced notification:** Misplaced items notified

**Input:** Generated by the shelf

**Process**: Initiates the request for sms alert through API

**Output:** Sms alert to store keeper and stock clerk

* 1. **External Interfaces Requirements**

**User Interfaces**

* The RFID antennas must be placed in the line of sight of the RFID tags attached to the apparels.
* The RFID antennas must be placed at the distinct intervals in order to capture the shelf contents effectively and avoid duplicate data of same apparal by different antennas.

**3.5 Non Functional Requirements**

* **Availability**: The availability of application or software refers to ensure the use of application at any point of time, irrespective of the location and climate conditions.
* **Security**: The security of application refers to be safe and secure to ensure the integrity of the application such that there is no harm to the users and users data by any means.
* **Portability**: With the minimum requirements the application must be compatible and capable of working on any platform.
* **Reliability**: The changes made in the front-end with various different inputs must be reflected in backend in order to check the working of the application during the rise of problems.
* **Usability**: The application must be given with well-structured user manuals, well-formed GUI‟s for the users to learn how to operate the system.

**3.6 Design Constraints**

* Application can effectively respond to apparel stores.
* The RFID tags must be non-detachable from the clothings.
* The RFID tags must be non breakable.