Real Time Monitoring of Real Life Management Systems

Software Requirements Specification

SRS_7C_T8_1.0
10/09/2017

Sl.No	Name	USN#
1	NL Ramya	01FB14ECS124
2	Nachiketa Ramesh	01FB14ECS126
3	Nirmit Shetty	01FB14ECS134
4	Prerana Mehta	01FB14ECS164
5	Priyanka Narendra	01FB14ECS166
6	R Hariharan	01FB14ECS172
7	Rahul M	01FB14ECS178
8	Madhusudan H	01FB15ECS429
9	Nagaveni NK	01FB15ECS433

Guide : **Prof. Phalachandra**

Revision History

Date	Description	Author	Approver	Comments
10/09/17	SRS_7C_T8_1.0 (Section 2.3)	R Hariharan		
10/09/17	SRS_7C_T8_1.0 (Section 2.1)	N L Ramya		
10/09/17	SRS_7C_T8_1.0 (Section 2.2)	N L Ramya		
10/09/17	SRS_7C_T8_1.0 (Section 1.2)	Nagaveni N K		
10/09/17	SRS_7C_T8_1.0 (Section 1.3)	Nagaveni N K		

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	<Your Name>		

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed description of the **Real Time Monitoring of Real Life Management** software. The document explains the features of the system, the interface provided by the system, the working of the system and the conditions for its operation.This document is intended for the developers of the software.

1.2 Scope

(1) The software product is **"Real Time Monitoring of Hotel Management System"** being used for the Nisarga chain of hotels in Bangalore.

(2) The software's mobile reporting feature allows to view all the restaurant's bills in real-time, reducing the dependency on staff for the sales reports. Live bills can be tracked for each restaurant outlet. Custom reports can be created as per needs and help make smart and data-driven decisions for the business with ease. The software ensures that the restaurant kitchen never runs out of stock by tracking the inventory in real-time by giving automatic reminders and alerts.
Implementation of a recipe management system which would automatically deduct the amount of stock consumed based on the recipe of each dish isn't currently supported. Also, the software doesn't support an offline reporting feature that lets the software work smoothly even in the case of an internet outage.

(3) The system contains three main subsystems, the management end (hotel staff), the cloud computing system (analytics end) and the monitoring end (real-time updates/reports). The current management / monitoring system at the Nisarga Hotel is mostly manual accounting and checking. The technical backbone provide by us will ease the job of the Nisarga hotel manager and make the hotels functioning more efficient.

(4) Functioning:

- **The Management End :** Here the data is being accumulated and transferred within and outside the system. It consists of local database **(MySQL or MongoDB)** where the data gets stored before being pushed on to the cloud.
- **The Cloud Storage :** The centralized system where all the hotel details gets stored (**AWS**). Here analytics on the data is performed to generate reports. The new updates are sent to the mobile application.
- **Monitoring End :** Two applications would be provided to monitor their hotels.
  (1) **Mobile Application :** All the real-time updates and information from the system will be sent to a mobile application.The mobile application should represent the information in an understandable manner and should be easy to use.
  (2) **Progressive Web Application :** which provides all the information regarding the data accumulated. It should have reports showing the statistics on the working of the system and show all the information logged. */

1.3 Definitions, Acronyms, and Abbreviations

(1) RTMS : Real Time Monitoring System
(2) USP : Unique Selling Proposition
(3) AWS : Amazon Web Services
(4) MTBF : Mean Time Between Failure

2. General Description

2.1 Product Perspective

The real-time monitoring software replaces the traditional manual supervision system. The entire system consists of 4 major clients. In addition to these, the system also has additional staff to maintain and keep the system up to date. There are multiple roles that are responsible for this including the project lead, web application lead, mobile application lead and database manager.

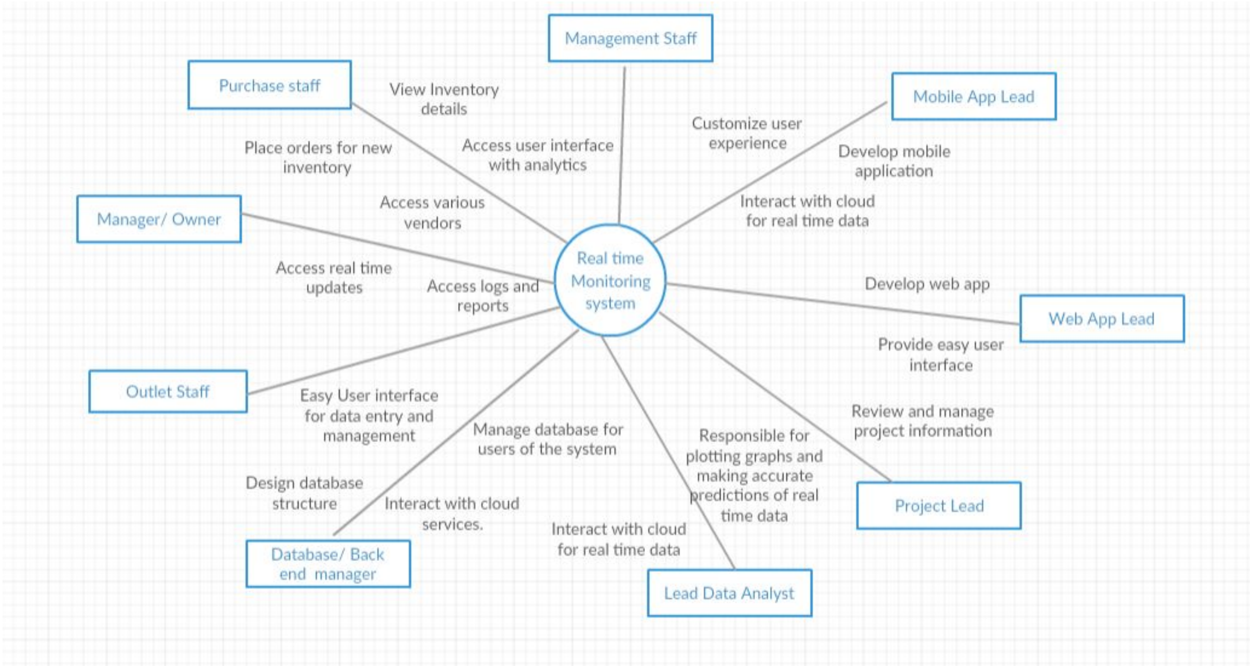
Management Staff
Purchase staff
View Inventory details
Mobile App Lead
Customize user experience
Place orders for new inventory
Access user interface with analytics
Develop mobile application
Manager/ Owner
Access various vendors
Interact with cloud for real time data
Real time Monitoring system
Access real time updates
Access logs and reports
Develop web app
Web App Lead
Provide easy user interface
Outlet Staff
Easy User interface for data entry and management
Manage database for users of the system
Review and manage project information
Responsible for plotting graphs and making accurate predictions of real time data
Design database structure
Interact with cloud services.
Project Lead
Interact with cloud for real time data
Database/ Back end manager
Lead Data Analyst

Figure 1. Overall picture of the system

2.2 Product Features

We divide the project into 3 major parts with 4 types of users of the system - The owner, outlet staff, purchase staff and management staff:

1. Web application for data entry and management: This is mainly used by outlet staff, purchase staff and management staff. We propose to make a web application for this purpose.

- ❏ The outlet staff will be provided with functionalities such as: Interface to easily view billing data, update new orders and transactions to the system.
- ❏ The purchase staff will have access to: Inventory manager which will keep track of all the goods and their respective quantities. Furthermore, they will be able to places orders to different vendors
- ❏ The management staff can do the following: View analytics of previous orders - make predictions as to what the customer currently requires.

2. All of the data that is entered by the outlet staff is flushed to cloud. This data could be from multiple outlets. There is one centralised cloud where all the data gets stored.
3. Mobile application for the owner of the restaurant where he gets real-time updates of actions taking place in the restaurant and also has access to logs and reports.

The main project features are:

- View all restaurant bill-wise reports in real time.
- Inventory Management module to keep tight control on stock and prevent wastage and theft.
- Live-data notified on mobile application to eliminate sabotage and data tampering.
- Web application for logs and reports on the functioning of the system.
- Table reservation by the customers of the hotel.
- Track the turnover time of each table.
- View and analyze the sales report of each restaurant outlet in detail.
- Transferring of data to a centralised cloud system periodically.

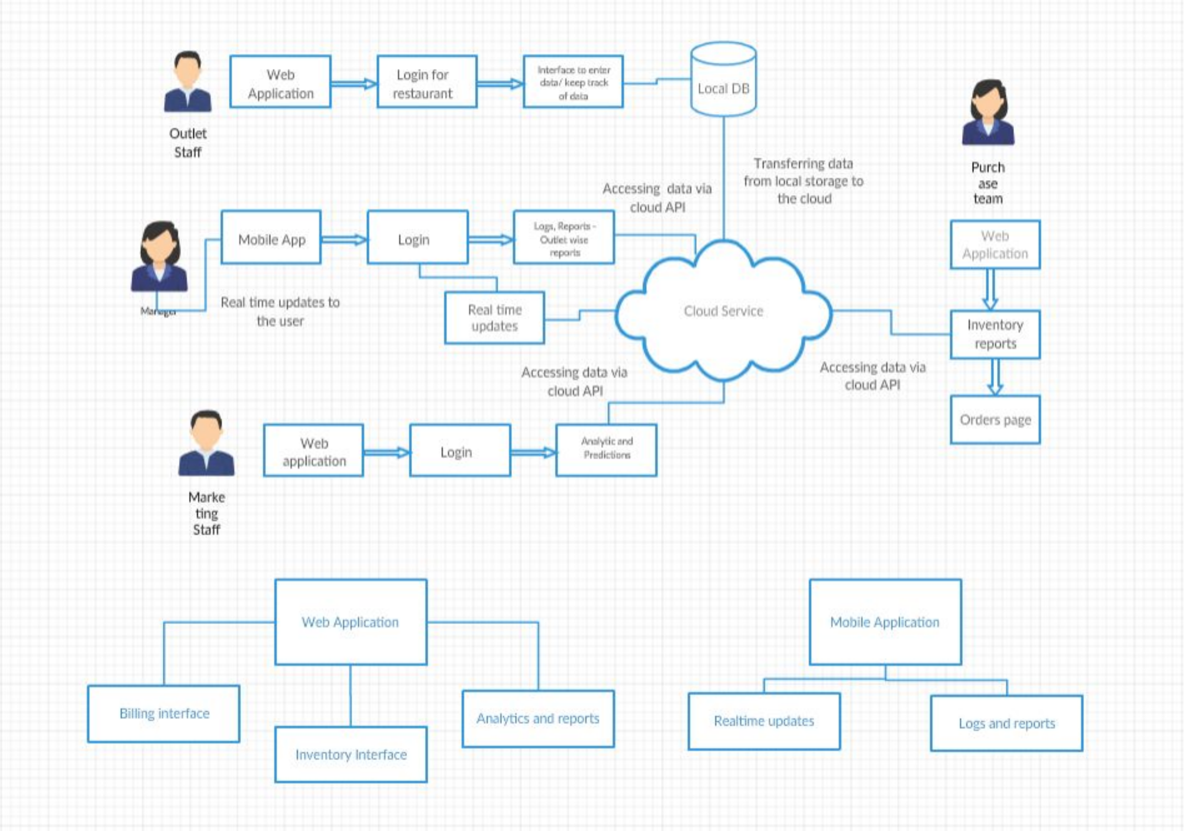

Figure 2. *Major features and how they relate.*

2.3 User Characteristic

*The ‘**User-Space’** of our product can be divided into two main categories:*

- ❑ *The Hotel Management **Staff** (Billing & Accounting Staff, Inventory Management Staff & Waiters)*
- ❑ *The Hotel **Manager/Owner***

ROLE	CHARACTERISTICS
Billing & Accounting Staff	The mentioned staff generates bills for the orders placed and then manages the total money/revenue collected for each day. The staff categorizes the payment methods and tallies the total cash received with the total amount cash payments received as on that day. His/Her task is also to report the account details on a daily basis.
Inventory Management Staff	Inventory stats is managed by the mentioned staff. He/She requires to update the usage of the items in the stock on a periodic basis, say every three hours. He/She can then set the menu based on the available items in the inventory. This role can be given to the head chef of the kitchen. He/She can also place request for order of items

	with appropriate information about the item and the level of urgency of its requirement.
Waiters	The role of the waiter is to take in orders from the table and generate **Kitchen Order Ticket (KOT)** for the same. He/She coordinates with the Billing & Account staff during the payment of the order.
Hotel Manager/Owner	He/She monitors the whole functioning of the system. He/She checks the account and revenue information provided by the Billing & Account staff. He/She checks the request made by the Inventory Management staff and places the order based on its urgency. The Manager/Owner basically controls the whole process.
Product Customer Support	The customer support will be the product company's employees who try to help out clients with the problems they are facing. They also communicate with the clients about latest software updates. If any major problem is faced then they communicate to the product development team..

2.4 Operating Environment

OE-1:	System is not dependent on geographical areas.
OE-2:	System shall operate in newest versions of all web browsers.
OE-3:	Users need to login to access the system, with a hierarchy of privileges based on the account.
OE-4:	Data is filled at the interface provided to the outlet staff.
OE-5:	Service is real time critical, but some latency can be tolerated, and the frontend must inform the user of this latency (last updated time).
OE-6:	Company owned data will be stored in the database, so the App Engine database must be secure.
OE-7:	jQuery javascript library will be needed to create the user interface for th system.

2.5 Assumptions and Constraints

This section presents the assumptions that will affect the proposed system, such as:

1. The first assumption made is that restaurants have made all their information computerized.
2. We require an active internet connection to send new data to the cloud.
3. Availability of resources and information for different implementations to make the system more generic.
4. The end user must be having an smartphone with him or a device with a browser.

3. Specific Requirements

3.1 Functional Requirements

FR1	Admin Accounts	The system should provide a user system with administrative powers.
FR2	Cashier Accounts	The system shall provide a cashier account system with users that can only log, not view or modify the data.
FR3	Data Entry	The system shall data entry for bills, table seating configurations and cctv camera footage along with money paid and changed issued.
FR4	Real Time Data Synchronisation	The system shall synchronise this data with the cloud service.
FR5	Aggregator	The system should provide an administrator accessible cloud service which holds the aggregated data.
FR6	Analyser	The system should provide a service at the cloud end that performs analytics on the aggregated cloud data.
FR6	Visualiser	The system should provide a frontend that provides visualisation of the data and its analytics.

3.2 Non-Functional Requirements

- Availability: The system downtime does not exceed 170 minutes per year.
- Reliability: Annual Failure Rate of 0.1-0.5 % can be expected (Due to cloud service provider)
- Performance: The system will be able to handle 100 transactions/second.
- Security: Every update to the system will be logged.
- Portability: The system will be easily portable. It will be made as a framework that can be easily deployed in other environments like hospitals, retail shops, etc.
- Survivability: The system will survive any natural disaster as it will maintained on the cloud, and local catastrophes will not affect it.

3.3 External Interface Requirements

3.3.1 User Interfaces

UI-1: For the hotel staff: Web application shall permit entry of orders as and when they come in. It will also provide an interface for the user to check if the transaction is complete and flushed to the cloud.

UI-2: For the hotel owner: A mobile app will display real time updates and show logs and reports of various outlets in one place

UI-3: For the management staff: A web application will show real time analytics based on the data coming in from the cloud.

3.3.2 Hardware and Software in terms of how they would interact or how they would be executed.

Systems required:

1. A system for data entry - Personal computer
2. Server to store the data on.
3. Mobile device to view logs and reports

Cloud API's to be used for interaction with the server.

3.3.3 Software Interfaces

SI-1: Time Management System

SI-1.1: The time management system shall transmit data to be stored on Mobile App Engine.

SI-1.2: After completion, system will display notification to user of result of data transmission.

SI-1.3: If applicable, system will then generate notification to be displayed.

SI-2: Database - The system shall communicate with a database through a programmatic interface for the following operations:

SI-2.1: To update the order details as they come in.

SI-2.2: To allow a user to submit timesheets/leave time.

SI-2.3: To allow owner to view details such as logs and reports.

SI-2.4: To allow management staff to access data, do analysis and flush intermediate results to cloud.