**Software Requirement Specification**

**For Hotel Management System**

**Usman Siddiqui(SE-056)**

**Basit Ali (SE-073)**

**NED University of Engineering & Technology**

**14 April, 2019**

**Abstract**

1. **What are you making?**

Hotel Management System

1. **Why are you making?**

This restaurant management system is for the customer of a company which sells all kind of restaurant equipment. The customers of the company are the restaurant owner and restaurant managers who can manage their restaurant equipment which they have bought from the company. They can create restaurant, add branches and manage equipment of each branch.

They can even register a complain of any equipment through the app in order to get service from the technician to get it repaired as soon as possible.

1. **What is the need?**

This app is for customer owner and managers to manage their restaurant, branches and equipment in each branch properly and get timely service from the technician if their equipment are not working properly.

**Technology:**

1. Android Application (Java)
2. Backed using Laravel (Framework of PHP)
3. **Introduction**

Sphere is a hotel management system for the customer of a company which sells all kind of restaurant equipment. The customers of the company are the restaurant owner and restaurant managers who can manage their restaurant equipment which they have bought from the company. They can create restaurant, add branches and manage equipment of each branch. They can even register a complaint of any equipment through the app in order to get service from the technician to get it repaired as soon as possible.

Restaurant owner can manage their assets, add new one and can request a service for one they bought from our company.

* 1. **Goals and Objectives**

This app is for customer owner and managers to manage their restaurant, branches and equipment in each branch properly and get timely service from the technician if their equipment is not working properly.

* 1. **Scope of the project**

It is difficult for every restaurant owner to manage his restaurant and the branches of the restaurants and keep track of all the equipments which are being used daily in his restaurant. So, a company which sells all the restaurant equipment through it’s ecommerce website provides a solution Sphere. The company has many customers who bought restaurant equipment daily from their ecommerce website for their restaurants. So it is important for the company to provide a solution to their customer through which they can manage their restaurants and branches and the equipment that are used in the branches and can also request a service for the equipment which is not working properly. This request will be notified to the technician of the company and a job will be automatically assigned to technician.

* 1. **Background of the project**

There is ecommerce online store Ekuep which sells all the good related to kitchen.

This company has some technician working to repair the equipment in case of any complaint by the restaurant owner or the customer. The customer previously used to complaint about the product they bought from the company on phone call and then the company manually used to assign job to the technician. Moreover it was becoming difficult for the restaurant owner to manage the products they buy from our store. Therefore the company wanted to automate the process so that the restaurant owner can buy product from the website and manage them and can register a complaint about the product through sphere.

**1.4 Technology and Tools used in the Project**

Throughout the design, implementation and testing phase of Sphere we have used variety of tools and technology that helped us produce a complete running promotional application.

* Html
* CSS
* Bootstrap
* Laravel
* Javascript
* MySQL
* VS Code
* Android Studio
* Git
* Postman
* Firebase Cloud Messaging
* Invision

1. **Salient features of the project**

* User can create an account or can login using his registered acount
* He can create his restaurant. He/She can create multiple restaurants
* He can add multiple branches under a restaurant.
* He can keep the track of the equipment he bought from our website and can add those equipment in a specific branch of his restaurant in which those equipment are used
* He can request a complaint of any equipment that is not working,.
* This complaint will be automatically assigned to a technician who is nearby.
* Technician will visit his place to repair the equipment.
* The customer will signature on the application on the completion of job.
* The customer then pay the technician for the job.

1. **Methodology**

We will develop this application using waterfall methodology i.e we first need to gather the information of restaurants, how they are managing their restaurant, keeping track of the equipments that are being used and develop the solution as per their to grab huge customer line and then feasibility analysis of this project including cost, expenditures and risk factors involved. The application will be developed in units and focus the module through which they can complaint about the equipment.

**Design phase**

In this plan is to break the project into components and components into objects to identify the relationships among themselves. On completion, the proposed solution will be represented as a systematic flow of the project with phase, the a help of suitable diagram. The UML diagrams (Use case and ERD) for each module will be made to elaborate the behavior and schema of the modules and to check if it requires any changes.

* 1. **Implementation phase**

**Laravel** will be used for backend development of both Web and Androidapplication. **Html/CSS** will be used to develop front-end of web application. **Java** will be used for the android application development using different libraries like JSON which used for serializing and deserializing **JavaScript** objects. For database MySQL will be used there would be a centralized database used for both applications. The modules will be developed independently and then integrated to make a complete system.

* 1. **Testing phase**

For testing we will prefer the unit testing so as to find the bugs at early stages by creating random accounts, verifying them through email or phone numbers and checking their validation requirements, signing in for different roles that if there is any clash in the application and whether the jobs are properly assigned to the technician who are nearby.

We will also consider that accuracy, efficiency and correctness or quality of the results should meet the customer requirements and the backend is capable of maintaining the large amount of live data and for this integration and system testing techniques will be applied.

* 1. **Evaluation phase**

In evaluation phase it will be taken into notice that how much time application takes to process a request, how frequently it deals with runtime errors and how often it manages the live data. Its effect on the memory will be evaluated. Lesser the time application takes to process the multiple requests, better will be its efficiency and performance and same for the correctness or quality of application. The application can also be evaluated by focusing its scope, the market it covers and the user needs which it fulfill.

1. Risks/Challenges

The most challenging part of this application is to get a significant number of users to be registered on our application and give them proper awareness that how they can use this solution to easily manage their store.

1. References

[1] <https://stackoverflow.com/>

**Problems:**

Manually working for hotel inventory can be very complicated and it takes big effort. For example:

* Keeping track of sales and purchases.
* Be alert about the stock of each and every hotel item.
* Storing suppliers information
* Generating reports.

**Solution:**

Our project is a hotel inventory system helps in keeping track of the inventory very easily.

* Different type of admins can login and perform their respective tasks easily. For example a inventory manager will manage the stocks, cook will manage ingredients etc.
* They will be able to manage details of suppliers, ingredients, recipes, menus etc.
* Can generate reports in different formats.

**A screenshot of a computer screen

Description automatically generatedSINGLETON:**

**Pseudo Code:**

class Database is

private connection object connec

Private statement object state

private database object

private Database Constructor() is

//conne.connect()//db connection

method dbconnection() is

//establish connection

if(object==null)

this.object=new Database()

Return object

method executequery(string)is

// /executing query

ResultSet result = statement.executeQuery(query)

class Menu is

Database db=Database.dbconnection();

result=db.executequery(“select...);

**A screenshot of a computer

Description automatically generatedFACTORY:**

**Pseudo Code:**

Abstract classs UserFactory is

Abstract method createuser()

Method dowork is()

User newuser=createuser()

Newuser.dowork();

Class ChefCreator extends UserFactory is

method createuser() is

Return new Chef()

Class RestaurantManagerCreator extends UserFactory is

method createuser() is

Return new RestaurantManager()

Class StockManagerCreator extends UserFactory is

method createuser() is

Return new StockManager()

Interface User is

Abstact method dowork() is

//empty method

Class Chef implements User is

Method dowork() is

//work of Chef or Chef’s view appear

Class RestaurantManager implements User is

Method dowork() is

//work of RestaurantManager view appear

Class StockManager implements User is

Method dowork() is

//work of SytockManager view appear

Class Main is

Userfactory user=new ChefCreator()

user.dowork();