**Software Requirement Specification(SRS)**

**for**

**Online food delivery System(OfdS)**

**Prepared by**

**Sneha Joshi**

**Proposed to**

**Trainer: Geethanjali anbalagan**

**10.03.2022**

**Table of contents**

1. **Introduction**
   1. Purpose
   2. Scope
   3. Audience, Definitions, Acronyms and Abbreviations
      1. Audience Definitions
      2. Acronyms and Abbreviations
   4. References
   5. Technologies to be used
   6. Overview
2. Overall Description
   1. Product Perspective
   2. Product Functions
   3. User Characteristics
   4. Constraints
   5. Operating Environment
3. Specific Requirements
   1. Functional Requirements
   2. Non-functional Requirements
      1. Security Requirements
      2. Easy to use Requirements
      3. Availability Requirements
4. Interfaces and Possible Scenarios
   1. User Interfaces

4.2 Software Interfaces

4.3 Hardware Interfaces

4.4 Communication Interface

1. **Introduction** 
   1. **Purpose:**

In today’s age of fast food and take-out, many restaurants have chosen to focus on quick preparation and speedy delivery of orders rather than offering a rich experience. The challenge encountered by the existing traditional ordering system serve as a major drawback to the realization of efficiency and customer satisfaction. The experience of ordering in most food restaurants is not pleasant for the customers.

The purpose of working on this project is to provide an ease to customers so that they can order anything, anytime while sitting anywhere and it can change the food ordering and delivering system into a better and appropriate way.

* 1. **Scope:**
* This system would be hassle free for users as they can select the food item they want and make payment for it.
* Reduce the purchasing time for customers.
* Improved and efficient services are provided to the customers by the inclusion of internet.

**Initial functional requirements will be:**

* **ADMIN**

**Managing users: the admin shall have a full access on the user database:**

* **Add user**
* **Delete user**
* **Edit a user record**
* **View user info**

**Managing restaurants: the admin shall have a full access on the restaurants database:**

* **Add a restaurant.**
* **Delete a restaurant.**
* **Edit a restaurant record.**
* **View all restaurants info**
* **View a specific restaurant info.**
* **CUSTOMER:**

**NO NEED TO HAVE AN ACCOUNT:**

**The system shall enable the customer to browse the restaurants. When the customer clicks on a specific restaurant, it shows the restaurant menu which contains all available dishes along with their price.**

**NEED FOR AN ACCOUNT:**

The system shall enable the customer to:

* Create an account.
* Login to the system.
* Place an order.
* Enter address where order to be delivered.
* System shall prompt customer to confirm the ordeal meal.
* System shall provide visual confirmation of the order placement.
* **MANAGER (Restaurant’s ADMIN):**
* **Create the restaurant’s menu.**
* **Create new food items and add them to the restaurant’s menu.**
* **Edit or update the details of any food items, such as: dish name, price, ingredients,…**
* **Delete any food item they want.**
* **Generate sales reports for their restaurant (weekly, monthly, yearly reports).**
* **KITCHEN STAFF (Restaurant’s Chefs):**
* **View pending orders**
* **Update orders status**
* **Verify the order to let them know that the order is ready to be delivered.**

**Initial Non-functional requirements will be:**

* **Operational requirements**

**The system should operate properly in web browsers.**

* **Performance requirements**

**The system should let user place an order in a short period of time.**

* **Security requirements**

**The system shall validate the username and password in order to login and make changes to the system.**

**Passwords should be encrypted.**

* **Usability requirements**

**The system should have a simple and easy-to-learn graphic user interfaces.**

**1.3 Audience Definitions, Acronyms and Abbreviations**

**1.3.1 Audience Definitions**

**The intended readers of this document are the developers of the site, testers, website owners, managers and coordinators.**

**1.3.2 Acronyms and Abbreviations**

|  |  |
| --- | --- |
| **Acronym** | **Meaning** |
| OFDS | Online Food Delivery System |
| JAVA EE | Java Enterprise Edition 5 |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | Secure Hypertext Transfer Protocol |
| TCP/IP | Transmission Control Protocol/Internet Protocol |

1.4 References

* IEEE 830-1998 standard for writing SRS document.
* *Fundamentals of Software Engineering*, 2nd ed. by Rajiv Mall.

## **1.5 Technologies to be used**

* Programming languages:
* JAVA EE: Java Enterprise Edition is a programming platform— part of the Java Platform-for developing and running distributed multi-tier architecture Java applications, based largely on modular software components running on an application server.
* HTML, XML: Hyper Text Markup Language and Extensible markup Language are the predominant markup languages for web pages. It provides a means to describe the structure of text-based information in a document and to supplement that text with interactive forms, embedded images, and other objects.
* JavaScript: A client side scripting language used to create dynamic web content and user interface.

Tools & Development Environment

* Apache Tomcat 6.0.18 Server: Apache Tomcat is a Servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the Java Server Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run.
* ECLIPSE J2EE: Eclipse is a toolkit which is designed for the creation of complex projects, providing fully dynamic web application utilizing EJB’s. This consist of EJB tools , CMP ,data mapping tools & a universal test client that is designed to aid testing of EJB’s.

**1.6 Overview**

* The rest of this SRS is organized as follows: Section 2 gives an overall description of the software. It gives what level of proficiency is expected of the user, some general constraints while making the software and some assumptions and dependencies that are assumed.
* Section 3 gives specific requirements which the software is expected to deliver. Functional requirements are given by the use case.
* Section 4 describes the various interfaces and possible scenarios.

1. **Overall Description**

**2.1 Product Perspective**

OFDS is aimed towards the vendors who want to reach out to the maximum cross-section of customer and common people who can be potential customer. This project envisages bridging the gap between the seller, the retailer and the customer. OFDS should be user-friendly, ‘quick to learn’ and reliable software for the above purpose. OFS is intended to be a stand-alone product and should not depend on the availability of other software. It should run on both UNIX and Windows based platform.

2.2 Product Functions

* User: Administrator
* Functions: The Administrator is the super user and has complete control over all the activities that can be performed. The application notifies the administrator of all shop creation requests, and the administrator can then approve or reject them. The administrator also manages the list of available product categories. The administrator can also view and delete entries in the guestbook.
* User: Shop Owner
* Functions: Any user can submit a shop creation request through the application. When the request is approved by the Administrator, the requester is notified, and from there on is given the role of Shop Owner. The Shop Owner is responsible for setting up the shop and maintaining it. The job involves managing the sub-categories of the items in the shop. Also, the shop owner can add or remove items from his shop. The Shop Owner can view different reports that give details of the sales and orders specific to his shop. The Shop Owner can also decide to close shop and remove it from the website.
* User: Customer/Guests
* Functions: A Customer can browse through the shops and choose products to place in a virtual shopping cart. The shopping cart details can be viewed and items can be removed from the cart. To proceed with the purchase, the customer is prompted to login. Also, the customer can modify personal profile information (such as phone number and shipping address) stored by the application. The customer can also view the status of any previous orders, and cancel any order that has not been shipped yet.
* User: Employees
* Functions: Purchase department under a Purchase manager to overlook purchasing activities if warehousing needs arise.
* Functions: Sales department under a Sales manager who will look after the sale of products and services, the most important activity.
* Functions: Accounts department under an Accounts manager to look after the accounting activities of the enterprise

**2.3 User Characteristics**

The end-users of the RMOS fall into three primary categories, unskilled, partly skilled and highly skilled.

* Unskilled user

The users of the surface computers are walk-in customers and should therefore be assumed to have no relevant prior skills or education other than basic abilities to operate an automated system; no

more complex than a parking meter or vending machine.

* Partly skilled user

The users of the tablets and displays are waiters and chefs respectively and they should be able to use the system and further be able to train others with minimal training themselves. They must be able to explain all elements of the user interfaces except the server. Supervisors also fall into the same category, though they will have to learn other sections of the system (refunds etc); these should not be of notably greater complexity than the standard functions. This class of user would be expected to have a junior high-school certificate education or equivalent.

* Highly skilled user

The initial installation and configuration of hardware and the constituent RMOS system components (especially the server) is guaranteed to require someone with notable computer experience, including extensive experience with network and operating systems to complete it. The software should not be needlessly complex, but it is still expected not to be entirely 'plug and play'. This class of user is expected to have a high-school certificate or equivalent, as well as extensive computer experience.

2.4 Constraints

* There is no maintainability of back up so availability will get affected.
* Limited to HTTP/HTTPS.
* Real-life credit card validation and Banking system is not implemented.
* No multilingual support

**2.5 Operating Environment**

The OFS is a website that shall operate in all famous browsers, for a model we are taking Microsoft Internet Explorer versions 7.0, 8.0 and 9.0.

1. **Specific Requirements**
   1. **Functional Requirements**
2. **User login**

**Allows the user to login in the system.**

1. **Restaurant’s Manager**

**Allows the restaurant’s manager to create their restaurant’s menu, then insert dishes to it.**

1. **Customer function**

**Let the customer view and browse the menu of any restaurant they desire, then give them the opportunity to place an order and choose how they want to receive it.**

* 1. **Non-Functional Requirements**

1. **Security**

**The system has a login and sign-up pages.**

**The system will protect this information by using identity in Asp.net that is made by.net.**

1. **Ease to use**

**The system should have a simple and easy-to-learn graphic user interfaces.**

**Provides help frame such as error messages while entering invalid data.**

1. **Availability**

**Each restaurant can specify their working days and hours, and then their restaurant’s menu on our system will be available for customers to make orders during these hours.**

4.Interface Requirement:

4.1 User Interfaces

Various interfaces for the product could be:

1). Login Page

2). Registration Form

3). There will be a screen displaying information about product that the shop having.

4). If the customers, select the buy button then another screen of shopping cart will be opened.

5). After ordering for the product, the system will send one copy of the bill to the customer’s Email-address

4.2 Software Interface:

1.Operating System: Windows7 Ultimate which supports networking. 2.JAVA development toolkit.

* 1. Hardware Interface:

Hardware requirements for insurance on internet will be same for both parties which are as follows:

Processor: Dual Core

RAM: 2 GB

Hard Disk: 320 GB

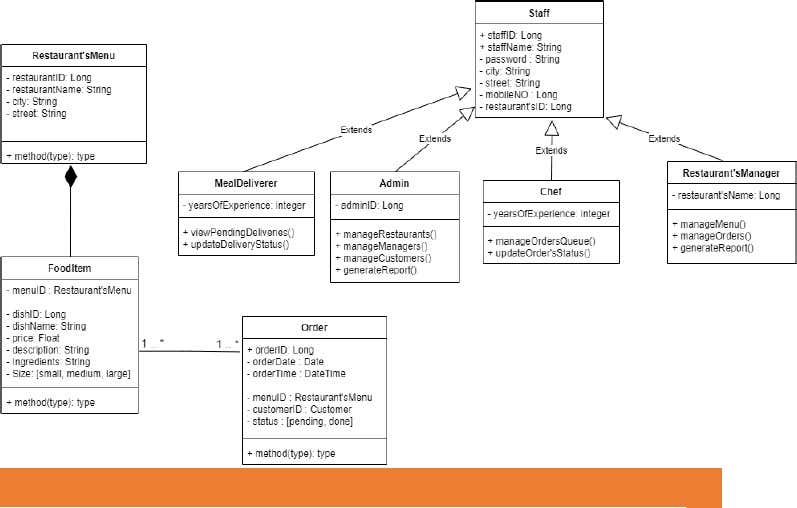
NIC: For each party

* 1. Communication Interfaces:

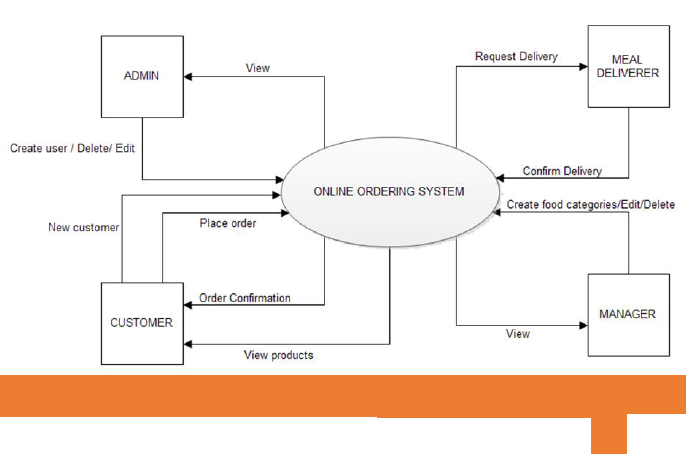
The two parties should be connected by LAN or WAN for the communication purpose.

SENDER Communication channel RECEIVER

CLASS DIAGRAM



CONTEXT DIAGRAM



CONCLUSION