**Restaurant Delivery Application**

**Software Requirements Document**

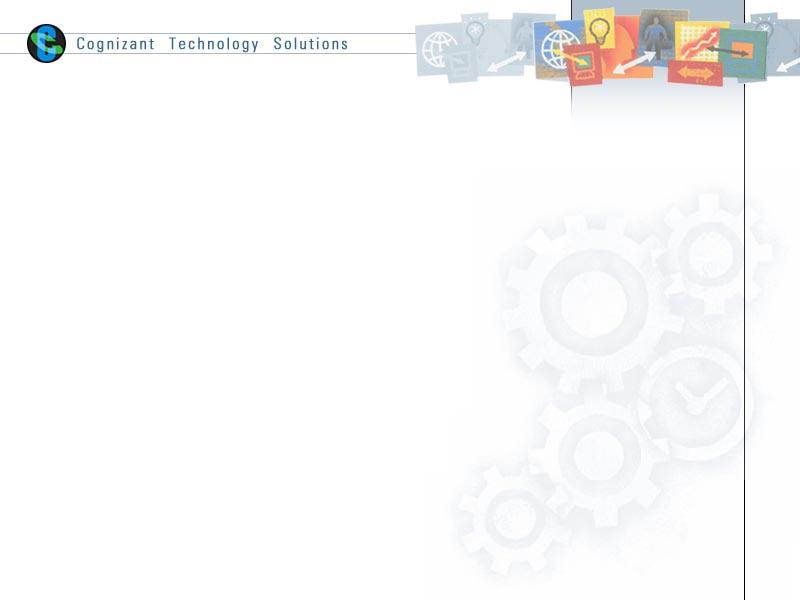


Table of Contents

[**1.0**](#_1fob9te) **Introduction 3**

[**1.1**](#_3znysh7) **About this document 3**

[1.1.1](#_2et92p0) Purpose & Scope of the document 3

[1.1.2](#_tyjcwt) Intended Audience 3

[**1.2**](#_3dy6vkm) **About the Software System 3**

[1.2.1](#_1t3h5sf) Purpose 3

[1.2.2](#_4d34og8) Scope of the system 3

[1.2.3](#_2s8eyo1) System Perspective 3

[1.2.4](#_17dp8vu) System Environment 4

[1.2.5](#_3rdcrjn) User Characteristics 4

[1.2.6](#_26in1rg) Impact of the System 4

[1.2.7](#_lnxbz9) Assumptions, Dependencies, Risks / Constraints 4

[1.2.8](#_3as4poj) Software proposed 4

[**2.0**](#_1pxezwc) **System Requirements 5**

[**2.1**](#_49x2ik5) **Functional Requirements 5**

[2.1.1](#_2p2csry) Login Module 6

[2.1.2](#_z337ya) New User Register 7

[2.1.3](#_3j2qqm3) Process Delivery Order 7

[2.1.4](#_1y810tw) Write Review 8

[2.1.5](#_4i7ojhp) Regular User 10

[2.1.6](#_1ci93xb) Admin Account 11

[2.1.7](#_3whwml4) Renew/Cancel Policy Module 11

[2.1.8](#_2bn6wsx) Logout Module 13

[**3.0**](#_147n2zr) **Change Log 14**

# 

# **Introduction**

## **About this document**

### **Purpose & Scope of the document**

The purpose of the software requirements document is to systematically capture requirements for the project and the system to be developed. Both functional and non-functional requirements are captured in this document. It also serves as the input for the project scoping.

The scope of this document is limited to addressing the requirements from a user, quality, and non-functional perspective. It is recommended that design aspects are not added in this document.

### **Intended Audience**

* *Project Team*
* *Customer*

## **About the Software System**

### **Purpose**

* + The Main purpose of **Fast Delivery Service** is to provide the ability to place a new delivery order, print an overview of the order in case the user wants to edit the order, process payment information from the user, write reviews about certain items in the menu and allow regular users to only view previous orders. Only registered users can see the menu items and place an order.

### **Scope of the system**

The Restaurant Delivery application is expected to be used by customers interested in placing a food delivery order on an e-cormmerce site having the below functions:

* Login
  + Authenticates the logged in user and logs out when needed.
* Register a new user
  + Registers a new user.
* Get Sell Items
  + User is provided with all the items in the menu ordered by category and the user can provide a quantity and a special request as a comment. Each item should also have a link to display that single item with relate reviews.
* Get Orders
  + The user can view only the order the user has made so far and the orders are grouped by the order number.
* Write Review
  + The User can write a review about a particular item
* Place Order
  + The User can add multiple items to an order, then the user selects to process the order and a page display a preview of the user order
* Edit delivery Order
  + The user can edit the order he/she is about to place in the preview order page or the user can confirm in order for the user to provide his/hers payment information.

**Admin functions:**

* Edit/Delete/add Sell items
  + Admin can edit, delete any of the items in the menu and as well as to add a new item to the menu.
* Delete Review
  + Admin can delete any review any regular user has made to any of the items i the menu.
* Delete Order
  + Admin can view all the order from all user
* Edit/Delete/Add Location
  + Admin can edit or delete any of the restaurant locations. Admin can also add a new location.

### **System Perspective**

This is an independent system and should be implemented using J2EE technologies.

### **System Environment**

When a user connects to the Web Server, the Web Server will interact with the Database after processing the business logic to transfer data to and from a database.

* Apache Tomcat 7.0 Web Server.
* Oracle MySQL database for data storage

### **User Characteristics**

The product should be user friendly; hence the user should not need any software and hardware knowledge. The user may be new to this restaurant delivery interface in general.

### **Impact of the System**

This is a new product which is developed for the customers. Expected impact of the product is to automate existing manual processes in order to make them more efficient and cost effective.

### **Assumptions, Dependencies, Risks / Constraints**

### **Software proposed**

* Tomcat 7.0/ JBoss
* J2EE
* Oracle DataBase 12c

# **System Requirements**

## **Functional Requirements**

### **Login Module**

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.1** | **Login Page** |
| **Functional Requirements**  When the user visits the correct url in the browser, welcome page appears this will have a short welcome message, a brief introduction of the purpose of this site and contact information. The welcome page should provides links to login, register or view all menu items. In the Login page there should be fields for username and password | |
| **User Interface Requirements**  1. The User ID and Password fields should be the text boxes (the username is the email of the user).  2. The entered characters in the Password field must be masked with an \* symbol.  3. There should be a Login button.  4. The page header should be “Restaurant Delivery”.  5. After login, the regular user should be send to the menu items page and if the user is an admin, the user should be send to the admin account page. | |

### **New User Register**

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.2** | **New User Page** |
| **Functional Requirements**  The user has to enter the following information in this page –  Personal Information   * First Name * Last Name * Password * Re-enter password * Gender * Address * Phone * Email | |
| **User Interface Requirements**   1. All the fields are mandatory 2. The page header should be “User Details”. 3. After register, the user should be send to the login page. | |

### Process Delivery Order

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.3** | **Menu Items** |
| **Functional Requirements**  The user has to view the following information in this page –   * Get Sell Items | |
| **User Interface Requirements**  1. Logout should be available  2. The Sell items can be selected by clicking on an add button and add it to a delivery order and the user is send back to the Sell items page. Each item should have an input box for comments to allow for special request  3. A button name process order should be included to take the user to select a location | |

### 

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.3.1** | **Select Location Page** |
| **Functional Requirements**  The user has to view the following information in this page –   * Each Location available of the same restaurant | |
| **User Interface Requirements**  1. Each location should have a button in order to be selected | |

### 

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.3.2** | **Review order Page** |
| **Functional Requirements**  Information about the items added by the user to the deliver order should get display with the total information about the cost and allow the following functionality:   * Delete an item from the delivery order * Cancel entire order | |
| **User Interface Requirements**  1. All items should have the image, name and price  2. Each item can be deleted  3. After clicking on confirm, the user is sent to receipt page  4. After clicking on cancel order, user is sent to the menu items page  5. After clicking on delete, user should be send back to Review Order page | |

### 

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.3.3** | **Payment Information** |
| **Functional Requirements**  The admin user should be able to view the following fields   * Credit Card Number * Secure code * Zipcode * button to submit | |
| **User Interface Requirements**  1. All fields are mandatory | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.3.4** | **Receipt page** |
| **Functional Requirements**  Complete information about the entire order should be display delivery including the location the order is being delivered from as well as the following information about the user   * First Name * Last Name * Address * Phone * Email Address | |
| **User Interface Requirements**   1. The complete information of the receipt should be display 2. logout should be available 3. Link to see menu items once more 4. Link to view all order made previously | |

### Write a Review

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.4.1** | **Single item page** |
| **Functional Requirements**  User should able to write a review to a single item at a time.  This page should have the following fields.   * Review fields of type text * item id fields type text and hidden * The should be a submit button * The username of the user should be contained in the session * Should be able to see all reviews made to this item even from other users   On click, the review should be saved and send back to the single item page. | |
| **User Interface Requirements**  1. Logout option should be available.  2. The Sell items manu page should be the only page with a link to the single item page | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.4.2** | **Single item page** |
| **Functional Requirements**  User should able to view the new review that he/she just submitted | |
| **User Interface Requirements**  1. Logout option should be available. | |

### Regular User

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.5.1** | **Regular User Page** |
| **Functional Requirements**  User should able to view the following fields populated with the user information   * First Name input field * Last Name input field * Password input field * Address input field * Email input field * Phone input field * button to update information | |
| **User Interface Requirements**  1. Logout option should be available.  2. All fields are mandatory  3. There should be a link to request for all order that the user has made | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.5.2** | **Regular User Page** |
| **Functional Requirements**  User should able to view the below information after requesting to view all order from that user.   * Item name * Item price * Total of order and tax * Information should be display group by order number | |
| **User Interface Requirements**  1. Logout option should be available.  2. Button to return to the menu items. | |

### Admin Account

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.6.1** | **Admin Page** |
| **Functional Requirements**  The admin user should be able to view the following fields populated with the admin user information and the following link .   * First Name input field * Last Name input field * Password input field * Address input field * Email input field * Phone input field * button to update information * Navigate Application link that sends the admin to view all user in the application | |
| **User Interface Requirements**  1. Logout option should be available.  2. The Navigate Application link should only be available in the Admin Account page. | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.6.2** | **List of User Page** |
| **Functional Requirements**  The page should display all user in the Application.   * Each user have a delete button and an edit button * The edit button should send admin to a page to display that user information to modify | |
| **User Interface Requirements**  1. Logout option should be available.  2. First, last name, email should be included in the list of user  3. There should be other links that sends the admin to Location list, Order list, Sell items list | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.6.2.1** | **Admin Single user edit Page** |
| **Functional Requirements**  Admin should able to view the following fields populated with the user information as well   * First Name input field * Last Name input field * Password input field * Confirm Password input field * Address input field * Email input field * Phone input field * button to update information * Item name * Review Descriptions * button to delete review | |
| **User Interface Requirements**  1. Logout option should be available.  2. All fields are mandatory  3. After clicking update or delete button, admin should be send to the list of users  4. All the review that the user has made should be listed in this page  5. There should be other links that sends the admin to Location list, Order list, Sell items list | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.6.3** | **Admin Location Page** |
| **Functional Requirements**  The page should display the input fields.   * Location name * address * city * number\_staff * zipcode * image * Button to add   The page should display the information of the location populated in input fields.   * Location name * address * city * number\_staff * zipcode * image * Button to update * button to delete | |
| **User Interface Requirements**  1. Logout option should be available.  2. The add button should add an location and send the admin back to the list of location  3. The details should be displayed in table format.  4. Each location should have two button to delete or edit  5. Once clicking on either delete or edit, the admin should be send back to the **Admin Location Page**  6. There should be other links that sends the admin to Userlist, Order list, Sell items list | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.6.4** | **Admin Order Page** |
| **Functional Requirements**  The page should display the information of the Orders.   * Total * Number of items * order number * Button to delete | |
| **User Interface Requirements**  1. Logout option should be available.  2. The details should be displayed in table format.  3. Each Order should have button to delete  4. Once clicking on either delete, the admin should be send back to the **Admin Order Page**  5. There should be other links that sends the admin to Location list, User list, Sell items list | |

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.6.5** | **Admin Item Page** |
| **Functional Requirements**  The page should display following input fields.   * item name * price * description * image * active * category * Button to add   The page should display the information of the items populated in input fields.   * item name * price * description * image * active * category * Button to edit * button to delete | |
| **User Interface Requirements**  1. Logout option should be available.  2. After admin clicks add button, admin should be send to the **Admin ItemPage**  3. The details should be displayed in table format.  4. Each item should have button to edit.  5. Each item should have button to delete.  6. Once clicking on either delete or edit, the admin should be send back to the **Admin item Page**  7. There should be other links that sends the admin to Location list, Order list, User list | |

### **Logout Module**

|  |  |
| --- | --- |
| **Restaurant Delivery\_Req-2.1.7** | **Logout Page** |
| **Functional Requirements**  The logout link should be available in the pages specified pages and new user registration in the right corner. | |
| **User Interface Requirements**  ‘Thanks for Visiting Us !!! You have been logged out’ should be displayed.  ‘Want to login again? Click below’ should be displayed with a Login button to go to the login screen. | |

### 3.Test Cases

### 3.1

On a Maven project using TestNG, JUNIT the DataBase components should be tested as followed:

|  |  |
| --- | --- |
| **DataBase Components** |  |
| **Functional Requirements** Login Module: Test all possible scenarios using this component to retrieve a specific userNew User Register Module: Test all possible scenarios using this component to save a specific userProcess Delivery Order: Test all possible scenarios using this component to retrieve all order, save an order, delete and update order. Take in consideration the Locations, Payment Information, Receipt ComponentWrite Review: Test all possible scenarios using this component to save a review, retrieve one or more review, get a single itemRegular User: Test all possible scenarios using this component to update a user information. Get an order from a specific userAdmin Account: Test all possible scenarios using this component to update each user information, view all items, user, locations, orders. Update or delete an item, add a location | |
| **Technology Requirements**  Maven project should include the use of TestNG executed from XML file, Junit assert methods, Pom.xml should have different profiles configured to execute each individual xml file | |

### 3.2

On a Maven project using Cucumber, JUNIT and Selenium automated processes should test the user interfaces:

|  |  |
| --- | --- |
| **Application** |  |
| **Functional Requirements** Using the user interface requirements from each page utilize Selenium, Cucumber and Junit to automate processes for the following pages: Login Page  Registration Page  Menu Items Page  Location Page  Review Order Page  Payment Information Page  Simge Item Page  Regular User Page  Admin Account Page  Admin User List Page  Admin Sell Items List Page  Admin Location List Page  List of order Page | |
| **User Interface Requirements**  For each page use the User Interface Requirements documented to create each test case scenario. For each Feature of the application provide a feature file, step definition file and a Junit runner class. | |

# **Change Log**

Please note that this table needs to be maintained even if a Configuration Management tool is used.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version Number** | **Changes made** | | | |
| V1.0 | *<First version>* | | | |
| V1.1 | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes effected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| V1.2 | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes effected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |