

ISLAMABAD FOOD DELIVERY SYSTEM

System Requirement Specification



Feburary 04,2020

# Introduction

## Purpose

This system is going to automate the food services, within the city, Islamabad. The user will have accessed to the list of items and can order the food. Both remote and local customers can enjoy this facility on a single touch.

## Document Conventions

Though every tranaction is enlisted in the list of priority, but functionalities of admin are at top priority in terms of authentication, verification, validation and others.

## Useful for

It is useful for developers and testers.

## Product Scope

Business-relation will be established between employer and customer on a single touch. So, it is beneficial for both parties.

# Overview: System and its Dependencies

## Product Perspective

Though, several apps are available in the market to facilitate users but either some apps do not response to the user or some do not submit the added products on the system. So, this app will improve the system performance and the user data will be saved on both online/offline databases. The basic architecture of the proposed is mentioned below.

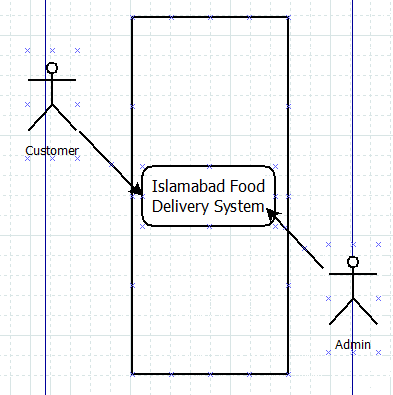


Figure 1

## Product Functions (of 1st Module)

The system of food delivery will able to perform following functions;

* Add SpecialItem
* Add GeneralItem
* Edit Item
* Add Branch
* Edit Branch
* Add Restaurant
* Edit Restaurant

## Operating Environment

Following are the minimum requirements to run the app on the android phone.

* CPU: Quad Core 1.2 GHz
* RAM: 1.0 GB
* Android Version: 4.4.2
* *Kernel Version: 3.10.17*

## Model and Implementation Constraints

Model:

* Agile Model

Architecture:

* Pattern: Model View Controller (MVC)

On Client Site:

* Development Tool: Android Studio 3.5
* Android Gradle Plugin Version 3.4.1
* Gradle Version 5.1.1
* Target SDK Version: API 28/Android 9.0 Pie
* Compile SDK Version API 28
* System Specs: 4 Gb, Core i5
* Operating System: Windows Enterprise
* Programming Language: Java, PHP
* Databases: SQLite DB Version, MySQL Version 8.0.18
* Protocol: HTTP

On Server Site:

* Server Software: Apache 2.4.1
* Script: PHP Version 7.3.12

# Functional Requirements (1st Module)

## Description and Priority

|  |  |  |
| --- | --- | --- |
| Transaction Name | Priority | Description |
| Add GeneralItem/Add SpecialItem | Medium | Either it is GeneralItem or SpecialItem, admin has permission to add new-item in the list of menus. However, the user must have system-generated id and password to access the tabs of “Add GeneralItem” or “Add SpecialItem” where user enters information as mentioned in 4.1.3. Use Cases. |
| Edit Item | High | This function updates existing-item information – either it is generalItem or specialItem -- in the DB. The user – who is administrator – can search the item to write itemCode in the edit-box. |
| Add Branch | Medium | If the existing-restaurant is going to launch new-branch, this use-case can help to register the new-branch in the system. The user will enter some basic information in the GUI of branch registration. |
| Edit Branch | High | This function allows the user to update the basic information of the branch in the DB but this information must contain branchLocation or branchItemRange. |
| Add Restaurant | Medium | New employers will be added through the function “Add Restaurant”. It demands that the user will enter detailed information in the GUI which will later be saved in the DB. |
| Edit Restaurant | high | The existing record i.e. restaurantLocation, and restaurantItemRange of the restaurant in the DB can be updated but the user must have restaurant-name to search the item in the DB. |
|  |  |  |

## Use Cases

|  |  |
| --- | --- |
| Name | Add GeneralItem/ Add GeneralItem |
| ID | UC-01 |
| Description | It will add item(s) in the menu-item-list -- which contains auto-generated item-code -- of all branches of a restaurant. |
| Primary Actors | Software Developer, Software Designer. |
| Secondary  Actors | QA Tester |
| Organizational  Benefit | - |
| Pre-condition | commonItem, itemCode, itemName, itemQunatity, itemPrice |
| Post-condition | Successfully Registered. |
| Main Success Scenario | 1.User selects radio-button for general-item-form display.  2.User enters item-basic information -- in which item itemCode is auto-generated and text view is displayed with “Rs” -- in the GUI.  3.User clicks ‘submit’ button for request.  4.System validates item’s information.  5.System checks in model class which condition is true either uniqueItem or commonItem and will send information in DB,  5.Failure/Success message will be displayed. |
| Alternate Scenario | - |
| Exceptions | EX1: Item contains special characters.   1. Item-name must contain alphabets, and digits.   EX2: Item-name already exists in DB.   1. User enters new(unique) item-name.   EX3: Item-code must unique.   1. Enter unique code.   EX4: Item-quantity is not in digits   1. Enter quantity-price in digit.   EX5: Item-price is not in digits   1. Enter again item-price in digit.   EX6: Radio-button has not been selected. |

|  |  |
| --- | --- |
| Name | Edit GeneralItem |
| ID | UC-02 |
| Description | It will update existing-item of the branch of the restaurant. |
| Primary Actors | Software Developer, Software Designer. |
| Secondary  Actors | QA Tester |
| Organizational  Benefit | Employer can share new price of food with customer in an ethical way. |
| Pre-condition | itemCode |
| Post-condition | Item-record has Successfully Updated. |
| Main Success Scenario | 1.User enters itemCode in GUI and clicks “Search” button.  2.System displays an item-information with title “Special Item/GeneralItem”.  3.User can edit following variables in item-GUI; itemQunatity, itemPrice.  4.System validates new information of item.  5.“Information Updated” message will be displayed. |
| Alternate Scenario | 1a. User can search item with either restaurantName and itemCode or branchName and itemCode.  3a. User can submit the same information of item; itemQunatity, itemPrice. |
| Exceptions | EX1: User inserts wrong item code.   1. Item code/name must watch with the existing data in DB.   EX2: User does not put item code in edit-text.   1. User must enter item code.   EX3: Item-quantity is not in digits   1. Enter quantity-price in digit.   EX4: Item-price is not in digits   1. Enter again item-price in digit. |
|  |  |

|  |  |
| --- | --- |
| Name | Add Restaurant |
| ID | UC-03 |
| Description | It will add new restaurant in the DB. |
| Primary Actors | Software Developer, Software Designer. |
| Secondary  Actors | QA Tester |
| Organizational  Benefit | - |
| Pre-condition | restaurantLocation, restaurantName, restaurantItemRange |
| Post-condition | Restaurant Account has Created. |
| Main Success Scenario | 1.User enters restaurant information in the GUI.  2.User requests for restaurant account.  3.System validates user’s information.  4.Failure/Success message will be displayed. |
| Alternate Scenario | - |
| Exceptions | EX1: Please enter data in the required fields.   1. User must enter data in the fields- represent by \*.   EX2: Restaurant-name contains special characters.   1. Restaurant-name must contain alphabets, and digits.   EX3: Restaurant-name already exists in DB.   1. User enters new(unique) restaurant-name.   EX4: Password contains either only alphabets or only digits or only special characters or without at-least six-digits.   1. Enter password again with the combination of alphabets, digits and special characters with at-least six digits.   EX5: Password and Confirm-Password do not match.   1. Fill again both edit-text boxes. |
|  |  |

|  |  |
| --- | --- |
| Name | Edit Restaurant |
| ID | UC-07 |
| Description | It will update restaurant basic information. |
| Primary Actors | Software Developer, Software Designer. |
| Secondary  Actors | QA Tester |
| Organizational  Benefit | - |
| Pre-condition | restaurantLocation, restaurantItemRange |
| Post-condition | Restaurant Basic Information has Updated. |
| Main Success Scenario | 1.User requests for login account.  2.User selects the “Edit” button.  3.User updates password/location.  4.System validates password/location.  5.Failure/Success message will be displayed. |
| Alternate Scenario | - |
| Exceptions | EX1: Password contains either only alphabets or only digits or only special characters or without at-least six-digits.   1. Enter password again with the combination of alphabets, digits and special characters with at-least six digits.   EX2: Password and Confirm-Password do not match.   1. Fill again both edit-text boxes.   EX3: Restaurant-name contains special characters.   1. Restaurant-name must contain alphabets, and digits.   EX4: Restaurant-name already exists in DB.   1. User enters new(unique) restaurant-name. |
|  |  |
|  |  |
| Name | Add Branch |
| ID | UC-08 |
| Description | Admin has permission to add new branch in the system. |
| Primary Actors | Software Developer, Software Designer. |
| Secondary  Actors | QA Tester |
| Organizational  Benefit | - |
| Pre-condition | branchName, branchLocation and branchItemRange. |
| Post-condition | New-branch Record has Entered in DB. |
| Main Success Scenario | 1.Admin sign-ins with the system.  2.User fills branch-registration-form with branchName, branchLocation and branchItemRange.  3.System validates.  4.Failure/Success message will be displayed. |
| Alternate Scenario | - |
| Exceptions |  |
|  |  |

|  |  |
| --- | --- |
| Name | Edit Branch |
| ID | UC-08 |
| Description | Admin has permission to update existing-branch record in the DB. |
| Primary Actors | Software Developer, Software Designer. |
| Secondary  Actors | QA Tester |
| Organizational  Benefit | - |
| Pre-condition | branchLocation and branchItemRange. |
| Post-condition | New-branch Record has Updated in DB. |
| Main Success Scenario | 1.Admin sign-ins with the system.  2.User selects “Edit” button and changes selected-branch record.  3.System validates.  4.Failure/Success message will be displayed. |
| Alternate Scenario | - |
| Exceptions | EX1: branchName, branchLocation must be either the combination of alphabets and digits or anyone.  EX2: If anyone - branchName, branchLocation – matches, it is ok   1. both are not already existed at a time. (Composite key is used in DB.)   EX3: branchItemRange must be in digits. |

## Sequence Diagram

|  |
| --- |
| Add GeneralItem/Add SpecialItem |
|  |
| Edit Item (Edit GeneralItem and Edit SpecialItem is converted into Edit Item) |
|  |
| Add Restaurant |
|  |
| Edit Restaurant |
| Add Branch |
|  |
| Edit Branch (It is same as add Restaurant) |
|  |

# Non-Functional Requirement

## Scalability

* Services of the system will be available whole day.
* Multiple user can access the system at a time.

## Verification

* Only verified users have accessed to the system.

## Validation

* The credentials of user’s login is the combination of characters, digits, and operators.