

[Foodies]

Design Document



May 6, 2019

Design Document

Project Name: Foodies

Project Manager: Aya Hamdy

Synopsis: Initial Design Document

Version: V1.0

Issue Date: 05-06-2019

Prepared by: Marwa El-sheshtawy

Authorized by: Foodies Team

* **Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Issue Date** | **Changes** | **Author** | **Reviewed By** |
| 1.0 | 06-05-2019 | Initial version | Marwa El-sheshtawy |  |
| 1.1 | 07-05-2019 | Add Architecture Diagram | Aya Hamdy |  |
| 1.2 | 08-05-2019 | Add Use Case Scenarios | Marwa El-sheshtawy |  |
| 1.3 | 09-05-2019 | Add Data Flow Diagram | Dina Ibrahim |  |
| 1.4 |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table of Contents

[1 Introduction 5](#_Toc8222440)

[1.1 Document Purpose 5](#_Toc8222441)

[1.2 Project Overview 5](#_Toc8222442)

[1.3 Document Scope 5](#_Toc8222443)

[1.4 Document Audience 5](#_Toc8222444)

[2 System Architectural Design 6](#_Toc8222445)

[2.1 High Level System Architecture 6](#_Toc8222446)

[2.2 Use Case Diagram 7](#_Toc8222447)

[2.2.1 Use Case Scenarios: 8](#_Toc8222448)

[2.3 Wireframes 13](#_Toc8222449)

[2.3.1 Signin Page 13](#_Toc8222450)

[2.3.2 Create Account Page 14](#_Toc8222451)

[2.3.3 Homepage 15](#_Toc8222452)

[2.3.4 Offers&Promotions Page 16](#_Toc8222453)

[2.3.5 Menu Page 17](#_Toc8222454)

[2.3.6 User Account Page 18](#_Toc8222455)

[2.3.7 Admin Panel – Delete User Page 19](#_Toc8222456)

[2.3.8 Admin Panel – Add Menu Page 20](#_Toc8222457)

* Table of Figures

[Figure 1 Foodies\_Sys\_UseCase\_001 7](#_Toc8075730)

# Introduction

The Software Design Document to provide documentation which will be used to aid in software development by providing the details for how the software should be built. Within the Software Design Document are narrative and graphical documentation of the software design for the project including use case models, sequence diagrams, object behavior models, and other supporting requirement information.

## Document Purpose

The purpose of the Software Design Document is to provide a description of the design of a system fully enough to allow for software development to proceed with an understanding of what is to be built and how it is expected to built. The Software Design Document provides information necessary to provide description of the details for the software and system to be built.

## 1.2 Project Overview

**Foodies** is a web app that will provide a good source of the nearby restaurants so the users can discover, browse all the restaurants near him or perform a keyword search and order their food from mouthwatering photo-driven menus.

## 1.3 Document Scope

This Software Design Document is for a base level system which will work as a proof of concept for the use of building a system that provides a base level of functionality

## 1.4 Document Audience

* Development team
* Testing team
* Project Manager
* Product owner

# System Architectural Design

## High Level System Architecture



Figure 1 Foodies Sys Architecture

## Use Case Diagram

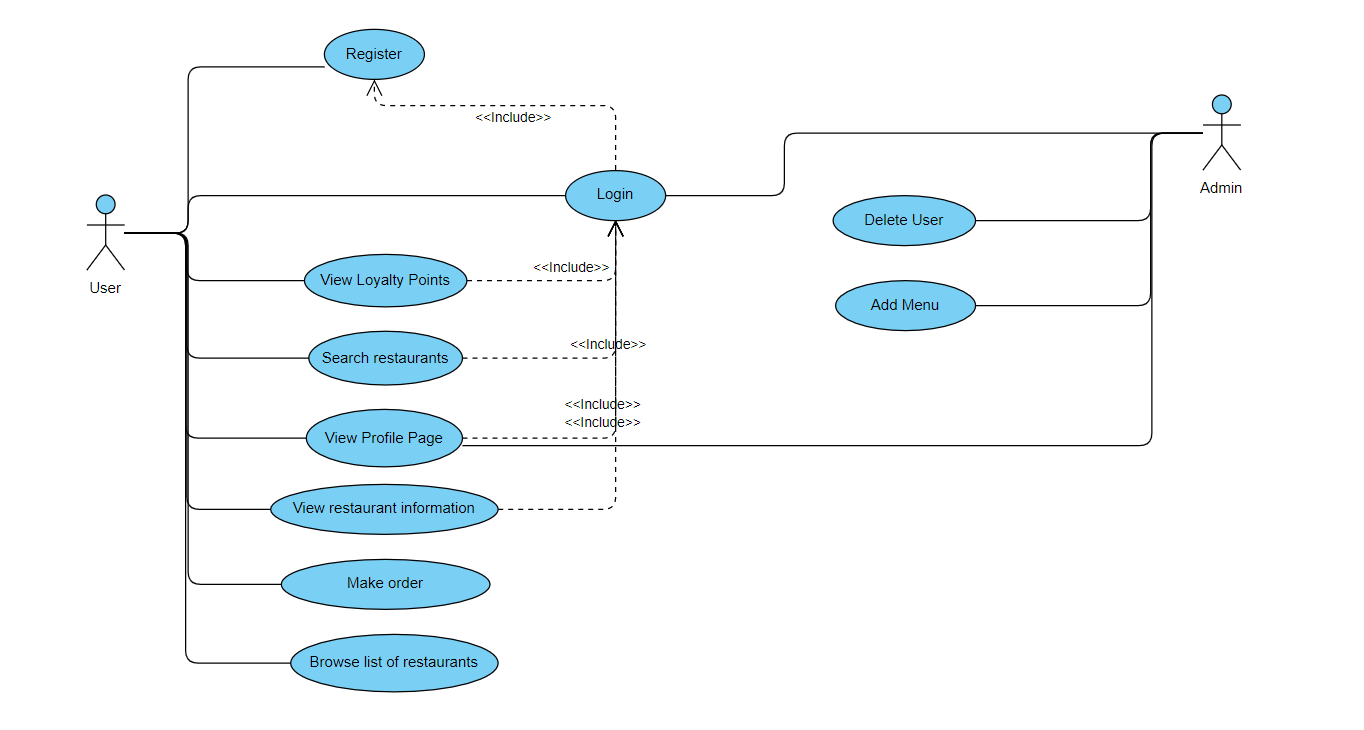


Figure 2 Foodies\_Sys\_UseCase

### 2.2.1 Use Case Scenarios:

#### 2.2.1.1 Register Use Case Scenario:

|  |  |
| --- | --- |
| **Foodies\_Sys\_UseCase\_001** | UseCase\_ID |
| Register | Scenario Name |
| User | Actors |
| A New user of the System creates an account by entering his personal information to create an account. | Brief Description |
| None | Pre-condition |
| |  |  | | --- | --- | | **System Response** | **Actor** | | * 1. The System is successfully create user’s account.   2. The System is failed to create account. | 1- If user entered successful information and is returned to the Login page as a Logged In User.  2- User is unable to Register for one or more reason and is returned to the Register page again. |  | | Post condition |
| |  |  | | --- | --- | | **System Response** | **Actor** | | * 1. The system prompts the user for a user Id and password or register new account.   2. The system prompts user for registration information (user Id, Email and password).   3. System verifies information and creates account.   4. The system navigate from register to log-in page | 1. New user select option to register, so system create an account. 2. New User fill-in the required information (user Id, Email and password). 3. Submit register button. 4. Return to login page. | | Steps |
| |  |  |  | | --- | --- | --- | | **System Response** | **Actor** | **Actor** | | 1.1-System displays information with appropriate message to correct invalid information.  2.1-System verifies user’s information and navigate to Login Page. | | 1- The User enter invalid information (User Id /Password or Email).  2- The User  Re-enter information. | | Alternative conditions |

Table 1: Register Use Case Scenario.

#### 2.2.1.2 Login Use Case Scenario:

|  |  |
| --- | --- |
| **Foodies\_Sys\_UseCase\_002** | UseCase\_ID |
| login | Scenario Name |
| User/Admin | Actors |
| A user of the System can login using his valid credentials. | Brief Description |
| User/Admin must be registered before. | Pre-condition |
|  |  |
| |  |  |  | | --- | --- | --- | | **System Response** | **Actor** | | | 2.1- The system validates the entered user Id and password, making sure that the entered username is a valid username in the System, and that the entered password is the password associated to the entered user Id.  3.1- The System navigate to restaurants page if the user is normal user and direct to User profiles page if as admin. | This use case starts when a system user is not logged in to the system and goes to the login page.   1. The Admin/User wants to log in. 2. Fill the user id and password. 3. Click Login Button. | | Steps |
| |  |  |  | | --- | --- | --- | | **System Response** | **Actor** | | | * 1. The system is successful.   2.1- The system fails. | | 1. If the user/Admin authenticated and redirected to next page. 2. If the user/Admin not authenticated | | Post –Condition |
| |  |  |  | | --- | --- | --- | | **System Response** | **Actor** | | | * 1. The system describes the reasons why the User authentication failed.   2. The system prompts the User to re-enter the valid information | | 1. If the user/Admin entered invalid user Id/password. | |  |

Table 2: Login Use Case Scenario.

#### 2.2.1.3 View Loyalty points Use Case Scenario:

|  |  |
| --- | --- |
| **Foodies\_Sys\_UseCase\_003** | UseCase\_ID |
| View Loyalty points | Scenario Name |
| User | Actors |
| A logged In user view his collected loyalty points. | Brief Description |
| Logged In to system with valid credentials | Pre-condition |
| |  |  | | --- | --- | | **System Response** | **Actor** | | 1.1 The system successfully display the loyalty points if user is authorized to view it. | 1. User select View Loyalty points from his profile. | | Steps |
| The system display user’s loyalty points if available. | Post- Condition |
| None | Alternative Conditions |

Table 3: View Loyalty points Use Case Scenario.

#### 2.2.1.4 View Profile Page Use Case Scenario:

|  |  |
| --- | --- |
| **Foodies\_Sys\_UseCase\_004** | UseCase\_ID |
| View profile page | Scenario Name |
| User/admin | Actors |
| A logged In user view his profile page | Brief Description |
| Logged In to system with valid credentials | Pre-condition |
| |  |  | | --- | --- | | **System Response** | **Actor** | | 1.1 The system successfully display profile if user is authorized to view it. | 1. User select my profile. | | Steps |
| The system display user’s profile with user’s (User Id , Email , Photo, Loyalty Points). | Post- Condition |
| None | Alternative Conditions |

Table 4: View Profile Page Use Case Scenario.

#### 2.2.1.5 View restaurant information Use Case Scenario:

|  |  |
| --- | --- |
| **Foodies\_Sys\_UseCase\_005** | UseCase\_ID |
| View Restaurant information | Scenario Name |
| User | Actors |
| A logged In user view restaurant information. | Brief Description |
| Logged In to system with valid credentials | Pre-condition |
| |  |  | | --- | --- | | **System Response** | **Actor** | | 1.1 The system successfully display the restaurant’s information (Phone Number, Location, Name, Menu, etc.). | 1. User select restaurant from list of nearby restaurants to View its information (Phone Number, Location, Name, Menu, etc.). | | Steps |
| The system display restaurant’s information if available. | Post- Condition |
| None | Alternative Conditions |

Table 5: Restaurant information Use Case Scenario.

## Wireframes

### 2.3.1 Signin Page

**ID:** Foodies\_SYS\_Wireframe\_001

**Description:**

This is the first page to appear when navigating to the website URL, any user with a registered account can sign in



### 2.3.2 Create Account Page

**ID:** Foodies\_SYS\_Wireframe\_002

**Description:**

User with no registered account can navigate to the “Create Account” Page and register with a UserID, an email and a password.



### 2.3.3 Homepage

**ID:** Foodies\_SYS\_Wireframe\_003

**Description:**

A logged-in user can view this page, the system shall request access to the user's location through a pop-up message.

A pop-up message with the restaurant phone number shall appear when the user click on “Order” Button.



### 2.3.4 Offers&Promotions Page

**ID:** Foodies\_SYS\_Wireframe\_004

**Description:**

A logged-in user can view this page, a pop-up message with the restaurant phone number shall appear when the user click on “Order” Button.



### 2.3.5 Menu Page

**ID:** Foodies\_SYS\_Wireframe\_005

**Description:**

A logged-in user can view this page with the restaurant’s menu and set an order.



### 2.3.6 User Account Page

**ID:** Foodies\_SYS\_Wireframe\_006

**Description:**

A logged-in user can view his profile showing his User ID, email, loyalty points and total orders.



### 2.3.7 Admin Panel – Delete User Page

**ID:** Foodies\_SYS\_Wireframe\_007

**Description:**

Admin can delete users by providing a UserID, click on “Search” then click on “Delete” Button



### 2.3.8 Admin Panel – Add Menu Page

**ID:** Foodies\_SYS\_Wireframe\_008

**Description:**

Admin can view this page, add a new restaurant with its menu and its logo then submit it to the system.

