Korean Food Restaurant Website

Use-Case-Realization Specification: Order Food

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 19/10/2017 | 1.0 | created | Minh Hien |
|  |  |  | Thuy Duong |
|  |  |  | Duc Huy |
|  |  |  |  |

Table of Contents

1. Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms, and Abbreviations 3

1.4 References 3

1.5 Overview 3

2. Flow of Events—Design 3

3. Derived Requirements 3

Use-Case-Realization Specification: <Use-Case Name>

# Introduction

Sometimes, it is very inconvenient to go directly to a specific restaurant to order food and then has to wait for a period of time in order to get the food. The purpose of the website is to get rid of that inconvenience and satisfy customer to the fullest.

This paper illustrates how to write a complete use case specification in order to capture the specific details of way the customer to order food from our restaurant with the help of our website.

## Purpose

The purpose of this document is to provide a more in depth view on the use case “Korean Food Restaurant Website”. You can think of this document to be an intermediate specification between the specification of the system and the actual design and implementation of the system. It is not to be seen neither as a use case specification nor a design specification but instead a document that will be used by both the systems designer and those who specify the system.

This document should give enough information that use case implementation is possible.

## Scope

The use case realization specification covers:

\* class diagram needed to implement this use case

\* method contract

\* sequence diagram (future)

## Definitions, Acronyms, and Abbreviations

## Admin – a user which is allowed to modify the data in the database.

## Guest – a user who is not logged in the system.

## References

## [Time Monitor Tool]

http://www.upedu.org/templates/cs/AD/upedu\_ex\_ucrea.pdf

## Overview

None

# Flow of Events—Design

2.1 **Use case**: Login

**Description**: A use login to the system with or without an account.

**Actor**: User

**Flow of events**:

Basic flow:

User enter homepage.

User enter name and password.

Password is correct and user is logged in.

Alternative flow 1:

User enter homepage.

User enter name and password.

Password is wrong and user typed again until correct.

Password is correct and user is logged in.

Alternative flow 2:

User enter homepage.

User select “login as guess”

User logged in guess mode.

2.2 **Use case**: Log out

**Description**: A user logs out the system.

**Actor**: User

**Flow of events**:

Basic flow:

User press log out button, confirm dialog appears.

User confirms to log out.

User is logged out and be brought back to homepage.

Alternative flow:

User force terminate program.

System automatically saves the changes then log the user out.

2.3 **Use case**: Display product

**Description**: after passing the logging view, users come to the main view which display the products as a table.

**Actor**: User

**Flow of events**:

Basic flow:

User logged in.

User is brought to salary page.

2.4 **Use case**: Add products

**Description**: administrator modifies the table by adding a new product.

**Actor**: admin.

**Flow of events**:

Basic flow:

Admin click on a row of employee, prompt dialog appears.

Admin enters required information.

Admin press OK and dialog is closed, the new product is added.

2.5 **Use case**: Delete product

**Description**: administrator modifies the table by deleting an existed product. **Actor**: Admin.

**Flow of events:**

Basic flow:

Admin click a delete button of a product, confirm dialog appears.

Admin confirm then the selected product is deleted out from table.

2.6 **Use case**: Update product.

**Description**: administrator modifies the table by editing an existed employee.

**Actor**: Admin.

**Flow of events**:

Basic flow:

Admin click on a row of employee, prompt dialog appears.

Admin enters required information.

Admin press OK and dialog is closed, the employee is updated.

2.7 **Use case**: Guess view.

**Description**: Guess user (users without account) view the salary table in static mode.

**Flow of events**:

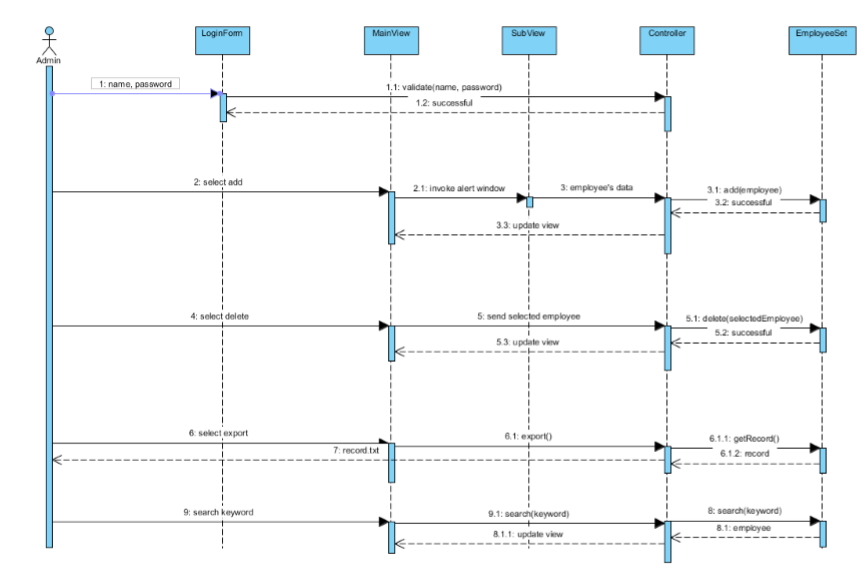
Basic flow:

User enter homepage.

User select “log in as guess” User is logged in in guess view mode.

User is directed to the salary table that does not have edit functions.

# Sequence diagram



# Derived Requirements

None