

Online Restaurant Order and Delivery System  
Software Requirements Specification  
For Web and Mobile Application

Version <1.0 >

Foodie Villa	Version: <1.0>
Software Requirement Specification	Date: October 17th, 2020
Foodie Villa Phase 1 Report.docx	

## Revision History

Date	Version	Description	Author
October 20, 2020	1.0	Create software requirement Specification	Sahrina Bhuiyan Fnu Palak Hope Dunner Nicholas Comer

## Table of Contents

1 Introduction.....	page 2
1.1 Purpose.....	page 2
1.2 Scope.....	page 2
1.3 Definitions, Acronyms, and Abbreviations.....	page 3
1.4 References.....	page 3
1.5 Overview.....	page 3
2. Overall Description.....	page 4
2.1 Use-Case Model Survey.....	page 4
2.2 Assumptions and Dependencies.....	page 5
3. Specific Requirements.....	page 5
3.1 Use-Case Reports.....	page 4
3.2 Supplementary Requirements.....	page 7
4. Supporting Information.....	page 8

# Software Requirements Specification

## **1. Introduction**

Our company “Foodie Villa” will have an online order and delivery system that will provide customers with a fun and effortless ordering experience. Our project will develop functions necessary to maintain a restaurant online. Through our purpose and scope we will discuss all the techniques necessary to produce this online system. It is important to understand a typical restaurant's workflow and maintenance requirements to develop such a system.

### **1.1 Purpose:**

The purpose of this system is to make life easier for customers and restaurant workers. Through the online ordering system, customers can easily access the restaurant's menu, order, give reviews and make complaints. We can configure their order type such as delivery or pickup and start preparation accordingly. It can also save their favourite orders allowing them to save time and easily reorder in the future. On the other hand, it will also help the restaurant employee's so that they no longer have to spend time taking the customers order, and it will also avoid communication errors.

### **1.2 Scope:**

Our system will be straightforward and easy to understand. Customers are guaranteed to feel safe and secure when inputting information in our system. For the customers that want their food delivered, we will require customers to provide us with their address so our delivery personals can find an efficient route to make fast deliveries. Our system will use an array of menu items that satisfies vegetarian and nonvegetarian interests of customers.

It will display the menu which will include meal names, price, quantity and descriptions and customers can add to their cart the items they want to order. Then customers can send in their orders to restaurants by pressing the confirm button. On the next step they'll input information such as name, address, date and time and finally submit. This system will also include a report subsystem, which will help our customer to generate a report and help the restaurant by giving their feedback.

### **1.3 Definitions, Acronyms, and Abbreviations**

Terms(Acronyms)	Definitions
Chef	Restaurant personnel that cooks, chooses menu items
Manager/supervisor	Processes customer registrations, handles compliments and complaints, hires, fires, raises or cut pay for chef(s) and delivery personnel.
VIP Customers	Customers who spent more than \$500 or placed 50 orders as registered customers.
Surfers	Who can browse the menus and ratings only, can apply to be a registered customer when they have a fixed amount of deposited money and checked by the manager.
Deliverers	Delivery Personnels that deliver the food to customers in a timely manner, find efficient ways to make their deliveries as they compete with other deliverers.
Registered Customers	Customers have to be registered on the system and have the ability to browse/search, order and vote on the food delivered, and can start discussion topics on the cooks, dishes, and/or deliver people

### **1.4 References**

[https://www.dropbox.com/s/6xao0nufjlzoch2/f20\\_proj\\_req.docx?dl=0](https://www.dropbox.com/s/6xao0nufjlzoch2/f20_proj_req.docx?dl=0)  
[http://www-cs.ccny.cuny.edu/~csjie/322/spec\\_sample.pdf](http://www-cs.ccny.cuny.edu/~csjie/322/spec_sample.pdf)

### **1.5 Overview**

This section will cover the organization of the rest of the specification.

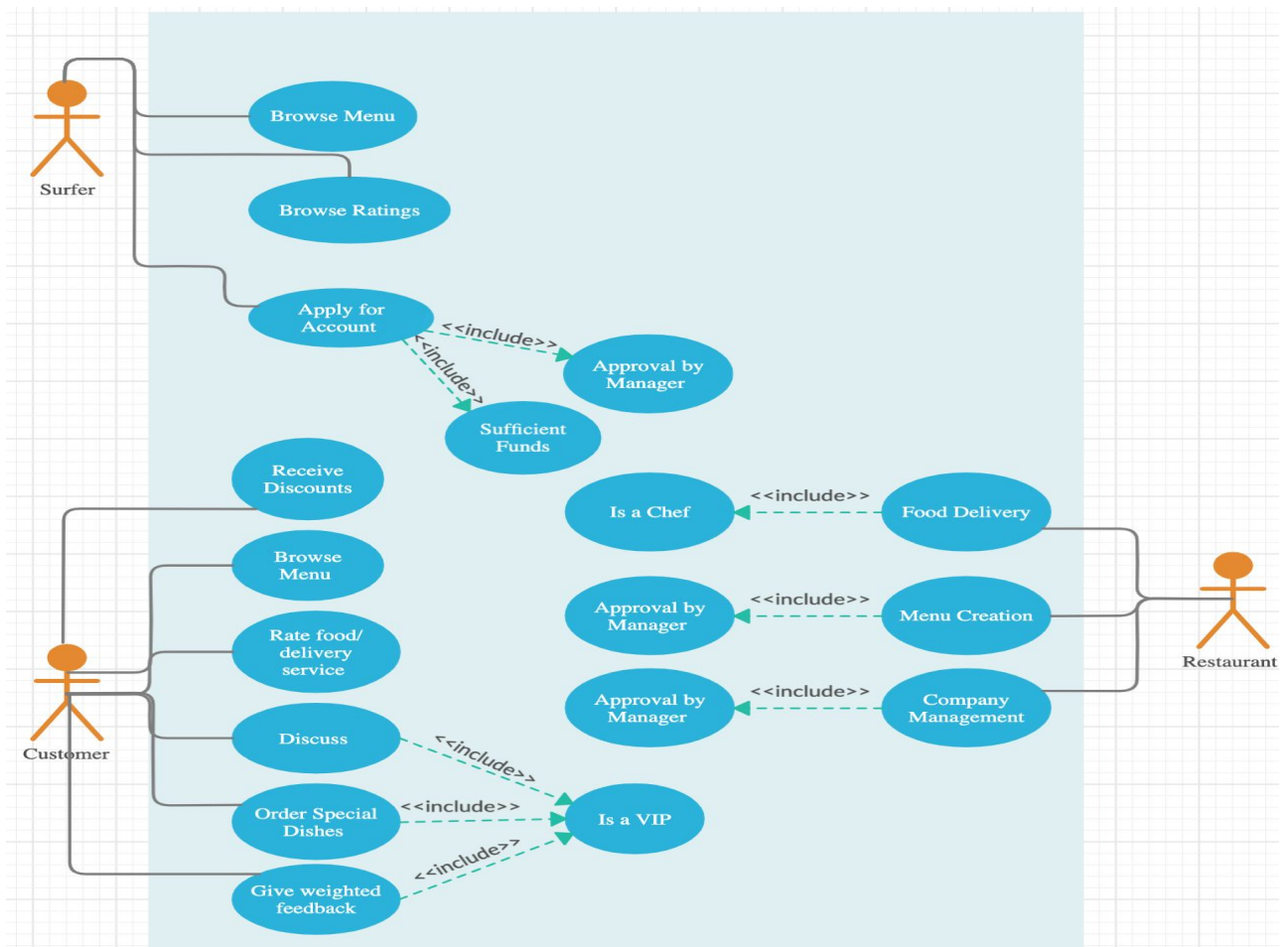
- Introduction: A general overview of the content supplied in this specification. Also includes explanations on the systems purpose, scope, as well as definitions for the reader.
- Overall Description: Contains the general factors that affect the functionality of the product as well as the requirements for the product. This section will include a use-case model to depict the functionality of the product as well as a list of assumptions and dependencies of the system.

- Specification Requirements: Depicts a detailed specification of the system. More specifically it includes a detailed explanation of the functionality of each use-case model in the overall design.
- Supporting Information: Contains extra information that makes the system requirement specification easier for the reader to understand and follow.

## 2. Overall Description

The use case diagram is a helpful visualization tool that can be used to better understand the functionality of different systems. Contained within this section, are implemented user case diagrams for our application system. This section will also include the possible assumptions and dependencies that are made when users interact with the ordering system.

### 2.1 Use-case model survey



Three different users

- Customers: VIP and Registered Customers
- Restaurant: Chefs, Deliverers, and Manager
- Surfers

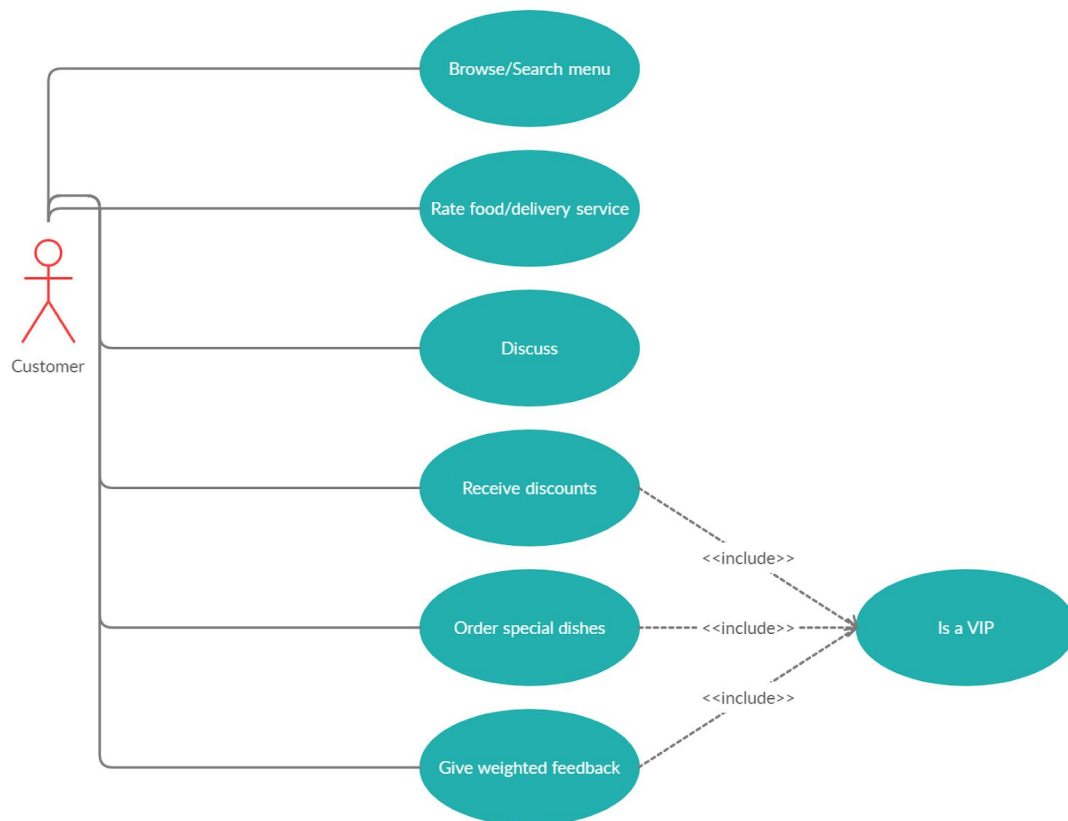
## **2.2 Assumptions and Dependencies**

- One assumption is it may still be hard for customers to understand and operate the system if their english is not too good(foreign customers/tourists)
- It may also be difficult for some elderly to use this web/mobile application if they are not too good at using computers/phones.
- Customers information may be protected but employee's might abuse it

## **3. Specific Requirements**

### **3.1 Use-case reports**

#### *3.1.1 User Class 1- Customers*



Use-Case: Browsing/Searching the menu

Description: Customers can find food to order.

Use-case: Rate food and delivery service

Description: Customers can rate the quality of their food and delivery service independently.

Use-case: Discuss with others about the food and employees.

Description: Customers can start and participate in discussion topics about cooks/dishes/delivery people.

Use-case: If VIP, receive discounts

Description: VIP customers (spend more than \$500 or 50 orders) receive a 10% discount on ordinary orders.

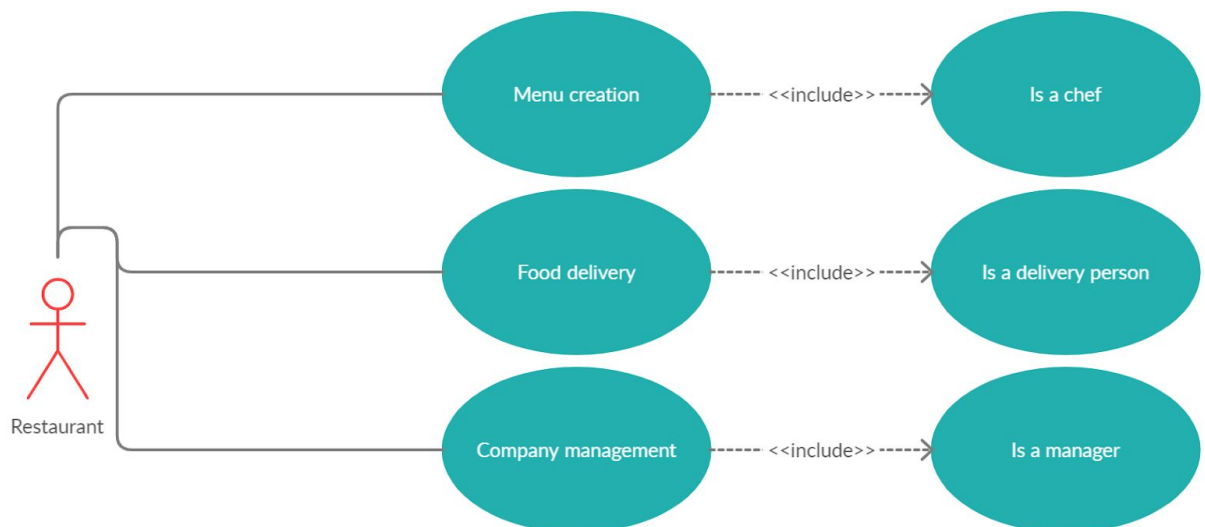
Use-case: If VIP, order special dishes

Description: VIP customers (spend more than \$500 or 50 orders) have access to specially created dishes that regular customers cannot get.

Use-case: If VIP, have weighted complaints

Description: VIP customers (spend more than \$500 or 50 orders) have their feedback weighted twice as much.

### 3.1.2 User Class 2- Restaurant



Use-Case: Menu creation

Description: Chefs independently decide the menus

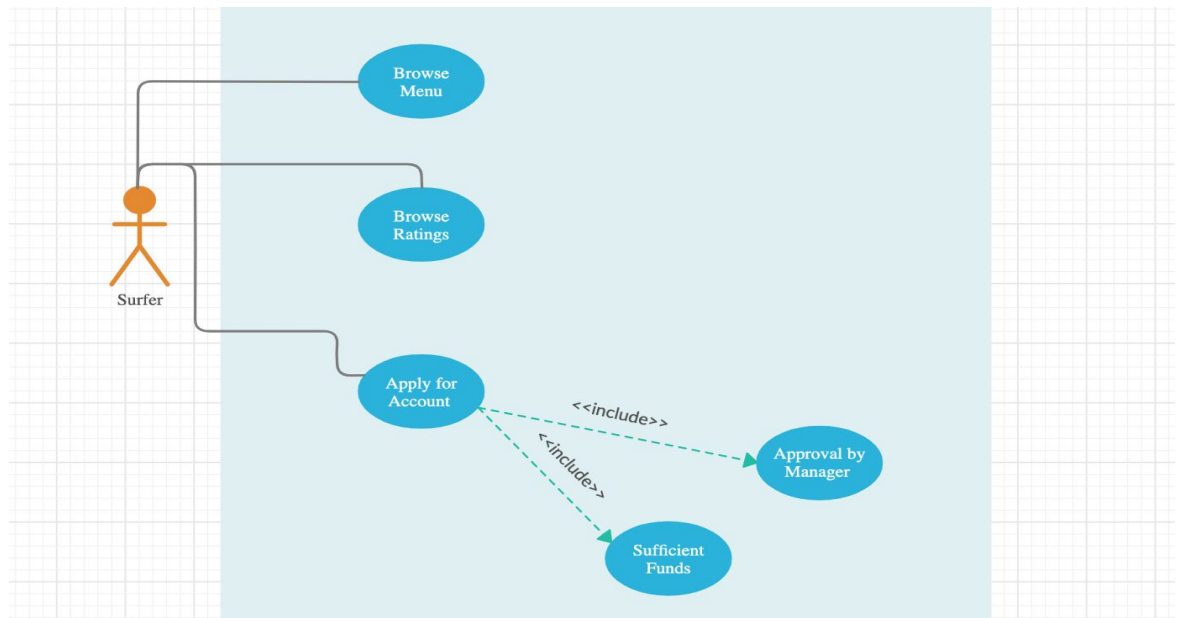
Use-Case: Food delivery

Description: Delivery people compete for food delivery

Use-Case: Company management

Description: Manager processes customer registrations, handles customer feedback, deals with chef and delivery people's employment and pay.

### 3.1.3 User Class 3- Surfer



Use-Case: Browse Menu

Description: A surfer can view the menu of the application

Use-Case: Browse Ratings

Description: A surfer can view the ratings of dishes on the application

Use-Case: Apply for Account

Description: A surfer can have the ability to register for an account; once checked to have sufficient funds in their account, as well as having the manager's approval.



## **3.2 Supplementary Requirements**

In order to make sure that the system can run seamlessly for all intended users, the application should meet the following requirements (The lists below are not included in the cases, yet capture the requirements necessary for the system):

### **3.2.1 Functionality**

- Errors: If any errors should arise, the user will receive feedback to the issue at hand and what steps should be taken next.

### **3.2.2 Reliability**

- The application should be live 24 hours a day, 7 days a week once deployed
- Every purchase will be logged and time stamped to ensure when an order was taken as well as if the order was paid for.

### **3.2.3 Usability**

- The interface should be designed to have an easy to use compatibility for all of the intended users.
- A user friendly UI design that will clearly let the users be aware of their choices in the application.
- All users should have a basic understanding of web applications to be able to use the system. No necessary training needed prior to using the application.

### **3.2.4 Performance**

- The application should be able to hold multiple users at once.

## **4. Supporting Information**

- Table of Contents
- Index
- The requirements to maintain such a restaurant was provided by the professor at:  
[https://www.dropbox.com/s/6xao0nufjlzoch2/f20\\_proj\\_req.docx?dl=0](https://www.dropbox.com/s/6xao0nufjlzoch2/f20_proj_req.docx?dl=0)
- The online order and delivery system will be developed by Hope Dunner, Nicholas Comer, Fnu Palak and Sahrina Bhuiyan. Our team can be contacted at sahrina.bhuiyan12@gmail.com