
Weirdoughs

Pizza Ordering System
Software Requirements Specification
for Web & Mobile Application

Version <1.0>

Pizza Ordering System	Version: <1.0>
Software Requirement Specification	Date: March 21 st , 2018
Weirdoughs Phase 1 Report.docx	

Revision History

Date	Version	Description	Author
March 21st, 2018	1.0	Create software requirement specification	Gerry Xu Melvin Tham Ayushya Amitabh Tobias He

Pizza Ordering System	Version: <1.0>
Software Requirement Specification	Date: March 21 st , 2018
Weirdoughs Phase 1 Report.docx	

Table of Contents

1. Introduction	4
1.1 Purpose.....	4
1.2 Scope.....	4
1.3 Definitions, Acronyms, and Abbreviations	4
1.4 References.....	5
1.5 Overview	5
2. Overall Description	6
2.1 Use-Case Model Survey	6
2.2 Assumptions and Dependencies.....	6
3. Specific Requirements	7
3.1 Use-Case Reports	7
3.1.1 User Class 1 – Customers (Visitors, Registered Customers, VIPs)	7
3.1.2 User Class 2 – Deliverer	9
3.1.3 User Class 3 – Cook.....	10
3.1.4 User Class 4 – Store Manager	11
3.2 Supplementary Requirements	13
3.2.1 Functionality	13
3.2.2 Reliability	13
3.2.3 Usability	13
3.2.4 Performance	13
4. Supporting Information	14

Software Requirements Specifications

1. Introduction

In this section, a brief generalization of the entire software requirement specifications document is provided. The topic of this section is divided into the purpose, scope, definitions, acronyms, abbreviations, references, and finally the overview.

1.1 Purpose

The purpose of this software requirement specification is to provide a detailed description of the Pizza Ordering System. It will explain the purpose of the features of the system, the interfaces of the system, what the system will do, what the constraints are, and how our system will react to any given input.

1.2 Scope

The Pizza Ordering System (POS) will be a web and mobile application for convenient pizza ordering from locations closest to the user.

The POS asks the user for location access to be able to plot their location using the Google Maps API. In case the user doesn't want to share their location, our system will allow the user to manually enter their address. Our implementation of the POS will also provide an array of Pizza types with in-depth customization, a cross-device synced shopping cart, a robust rating system (for shops, customers, and delivery personnel), and also a secure user account management system.

1.3 Definitions, Acronyms, and Abbreviations

Terms	Definition
POS	Pizza Ordering System
Shops	JSON Object in our database containing a shop's properties
Shop Managers	Account type for Pizza Shop Managers. Decides the pay for cooks and deliverers and handles all discrepancy among customers.
Customers	Account type for Customers. Customers who orders and pays for the pizza can evaluate the pizza and the one who delivers the pizza.
Deliverers	Account type for Delivery Personnel. In charge of deciding the fastest route from the pizza store to the destination.

Cook	Account type for Cook. Determines the menu and the prices of different pizzas.
BaaS	Back-end as a Service
Pizzas	<i>Property of Shop</i> - contains types of pizzas, toppings, & prices
Sides	<i>Property of Shop</i> - Containing side drinks/items and their prices
ReactJS	Modern Front-End JavaScript Framework created by Facebook
NodeJS	Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine
Material-UI	Component Library for ReactJS built using Material design guidelines by Google
Firebase	BasS provided by Google that also provides, user authentication, hosting, and custom backend functions using Node.JS - data in Firebase is stored and exported in JSON format.

1.4 References

[spec_sample.pdf](#), Software Requirements Specification
(http://www-cs.engr.ccny.cuny.edu/~csjie/322/spec_sample.pdf)

[Sample Report.pdf](#), Sample Report by “Three Asian Guys” for Mini Flickr Software Requirements Specification for Web Application
(<https://github.com/isatou/csc322/blob/master/Sample%20Report.pdf>)

1.5 Overview

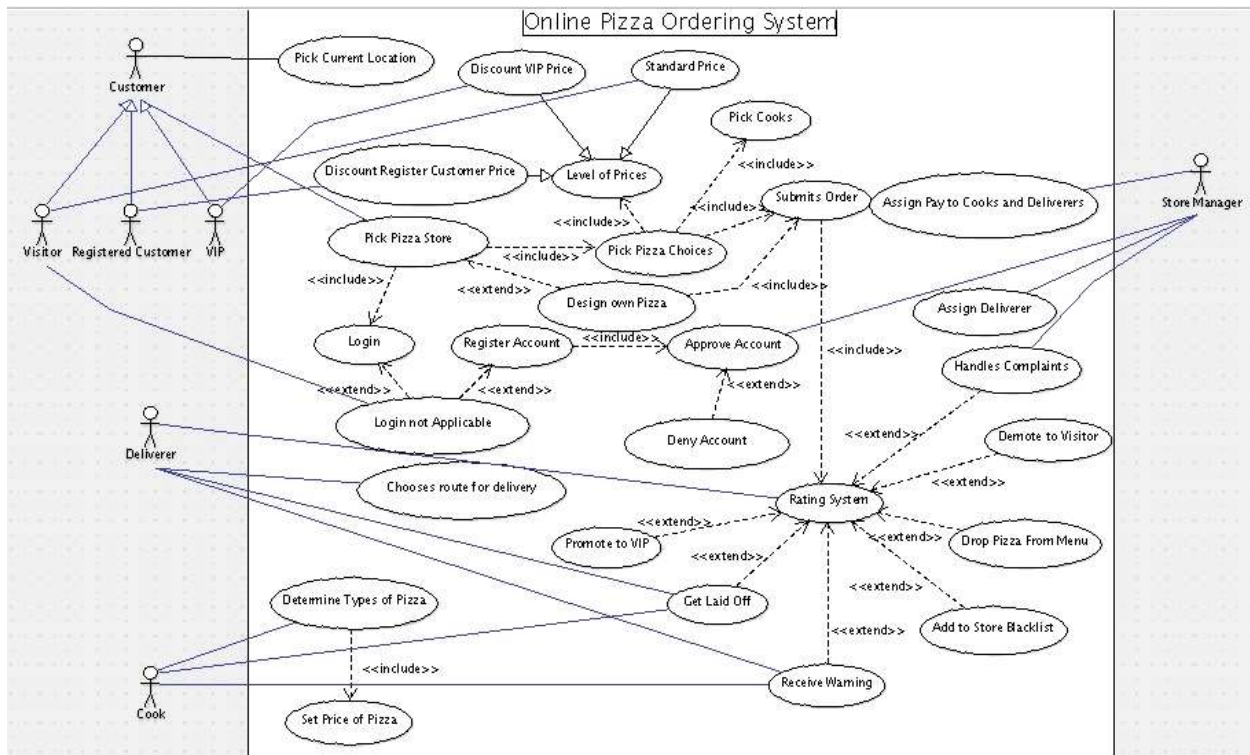
This specification is organized into the following sections:

- *Introduction*: Introduce the overall content of this specification. In addition, elaborate on the system’s purpose, scope and provide definitions for the reader.
- *Overall Description*: We will present an overview of the functionality of the product. This section will also include use-case model as well as a list of assumptions and dependencies of the system.
- *Specification Requirements*: Shows the detailed specification of the system. This includes detail explanation of the use-case model and also the functionality of the product that were not included in the overall use-case diagram.
- *Supplemental Information*: Includes supplemental information that makes the system requirement specification easier to use.

2. Overall Description

In this section, the overall model and foundation of our Online Pizza Ordering System will be detailed here with the help of the Use-Case Diagram. The Use-Case Diagram shows the functionalities of our system and allows us to visualize how users can interact with our system. This section will also provide possible assumption and dependencies made when users use the ordering system for intended purposes.

2.1 Use-Case Model Survey



Four different users

- Three types of customers - Visitors, Registered Customers and VIPs
- Shop Manager
- Deliverers
- Cooks

2.2 Assumptions and Dependencies

- One assumption to be made is that the customers understand that their rating will affect their status if they will be promoted to VIP or demoted down to visitor as well as both the cooks and deliverers in terms of them keeping their jobs.

Use-Case: Login

Description: Customer are prompt to enter in their username and password to login to order.

Use-Case: Login not Applicable

Description: Customer are treated as Visitor if login information is not provided.

Use-Case: Register Account

Description: Account registration for Visitor who want to create an account to become a registered customer.

Use-Case: Design own Pizza

Description: Customer are not required but have an option to customize and design their own style of pizza such as the toppings and the dough that they would like on their own pizza.

Use-Case: Pick Pizza Choices

Description: A menu displaying the pizza options that customer can select from. The selected pizzas are then saved to their shopping cart.

Use-Case: Pick Cooks

Description: Customer decides which cook are assigned to make the pizza for their order.

Use-Case: Level of Prices

Description: Contains the different prices for different types of customers such as visitors, registered customers and VIPs.

Use-Case: Standard Price

Description: Price of different types of pizzas for visitors. Visitors get the standard price with no discounts.

Use-Case: Discount Registered Customer Price

Description: Price of different types of pizzas for registered customers. Registered customers

Use-Case: Discount VIP Price

Description: Price of different types of pizza

Use-Case: Submit Order

Description: After finalizing their shopping cart, customer are greeted with a payment section where they are required to enter in their credit/debit card

information as well as their billing address in order to pay, process and submit their order to the system.

Use-Case: Rating System

Description: After the customer submits the order and receive the pizza, the customer must rate the pizza(s), delivery person, and the pizza store on a scale of 1 (worst) - 5 (best). These rating will then be kept in record and used to display the top three most favorite pizza for each store.

Use-Case: Add to Store Blacklist

Description: Based on what the deliverer rates the customer on a scale of 1 (worst) - 5 (best), if a customer's average rating is 1, he/she will be put on the blacklist and will never be able to be a registered user.

Use-Case: Drop Pizza from Menu

Description: Based on the customer's rating on the pizza(s), if a pizza receives an average rating of 2 or less in the last 3 orders, the pizza will be removed from the menu.

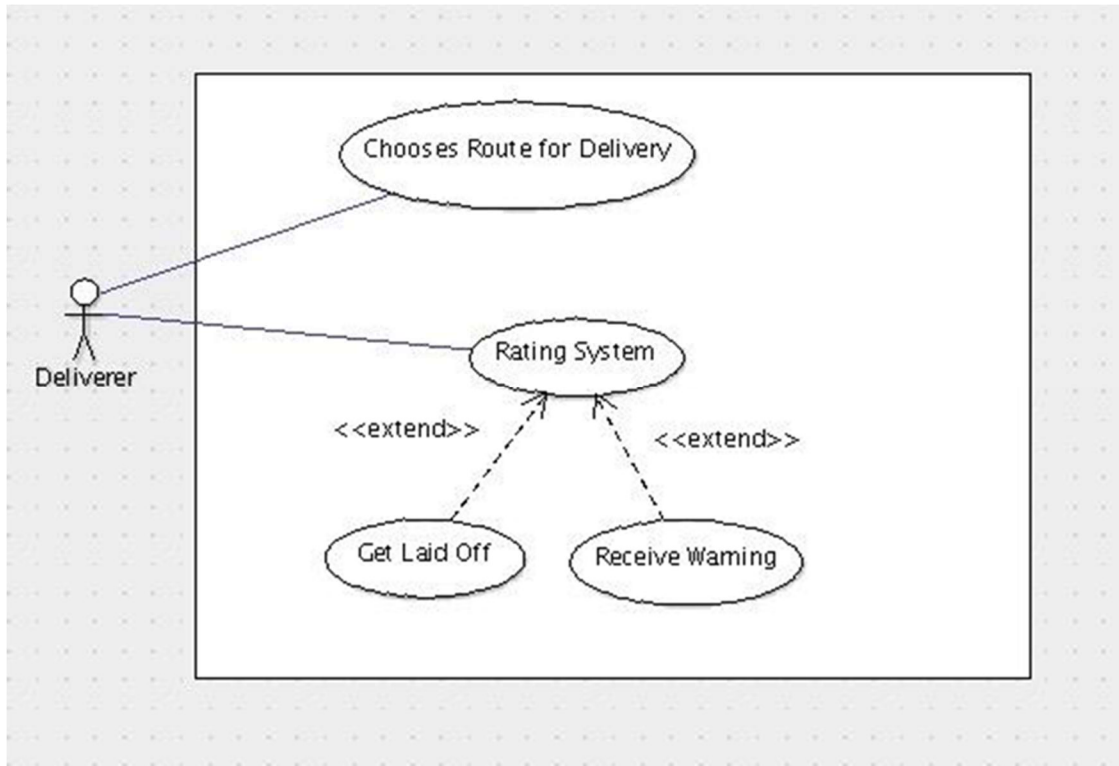
Use-Case: Promote To VIP

Description: A registered customer who made more than three orders with an average rating greater than 4 is automatically promoted to a VIP.

Use-Case: Demote to Visitor

Description: A registered customer who made more than three orders with an average rating less than 2 but greater than 1 is demoted to a visitor.

3.1.2 User Class 2 – Deliverer



Use-Case: Chooses Route for Delivery

Description: The chosen delivery person decides on the fastest route with the least traffic to deliver the pizza to the customer.

Use-Case: Rating System

Description: After the delivery person delivers the pizza(s) to the customer, the delivery person is required to rate the customer on a scale from 1 (worst) - 5 (best). This rating will be use to demote registered customer to visitor or promote registered customer to VIP.

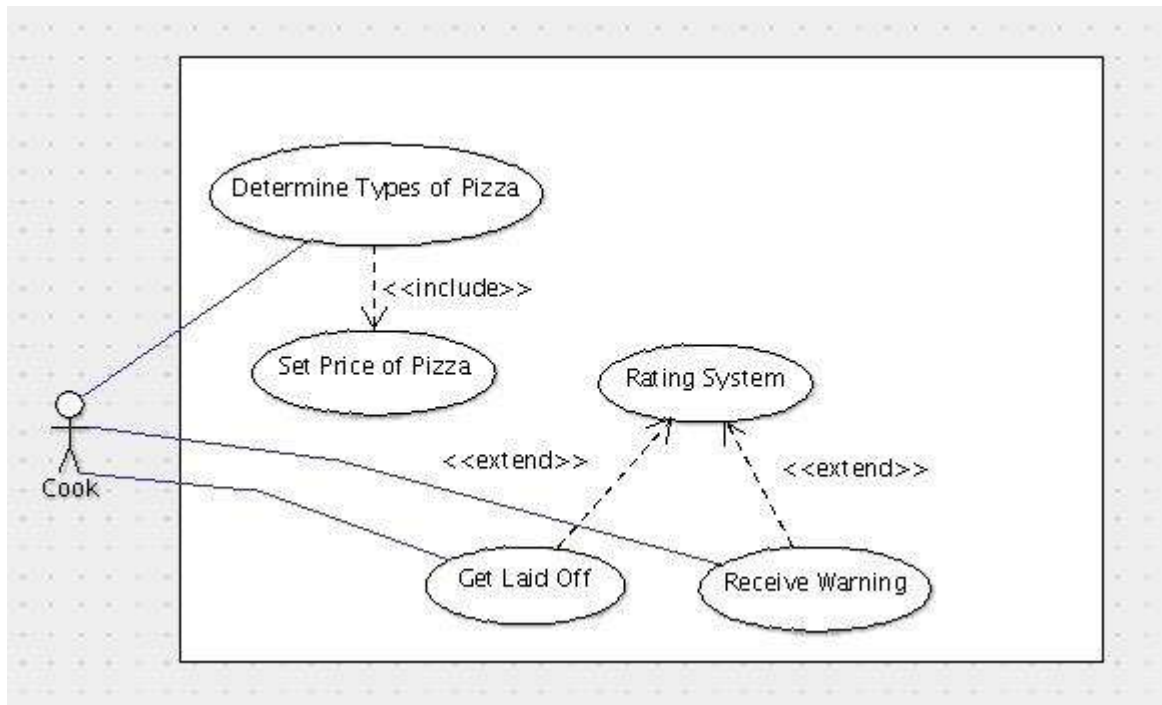
Use-Case: Receive Warning

Description: A delivery person receiving an average rating less than 2 in the last three deliveries will receive a warning, which can be erased by the manager.

Use-Case: Get Laid Off

Description: If a delivery person receives more than three warnings, they will be laid off the job.

3.1.3 User Class 3 – Cook



Use-Case: Determine Types of Pizza

Description: The cook will decide the types of pizza he/she will cook and put on the menu.

Use-Case: Set Price of Pizza

Description: The cook will decide the price of the pizza he/she will cook.

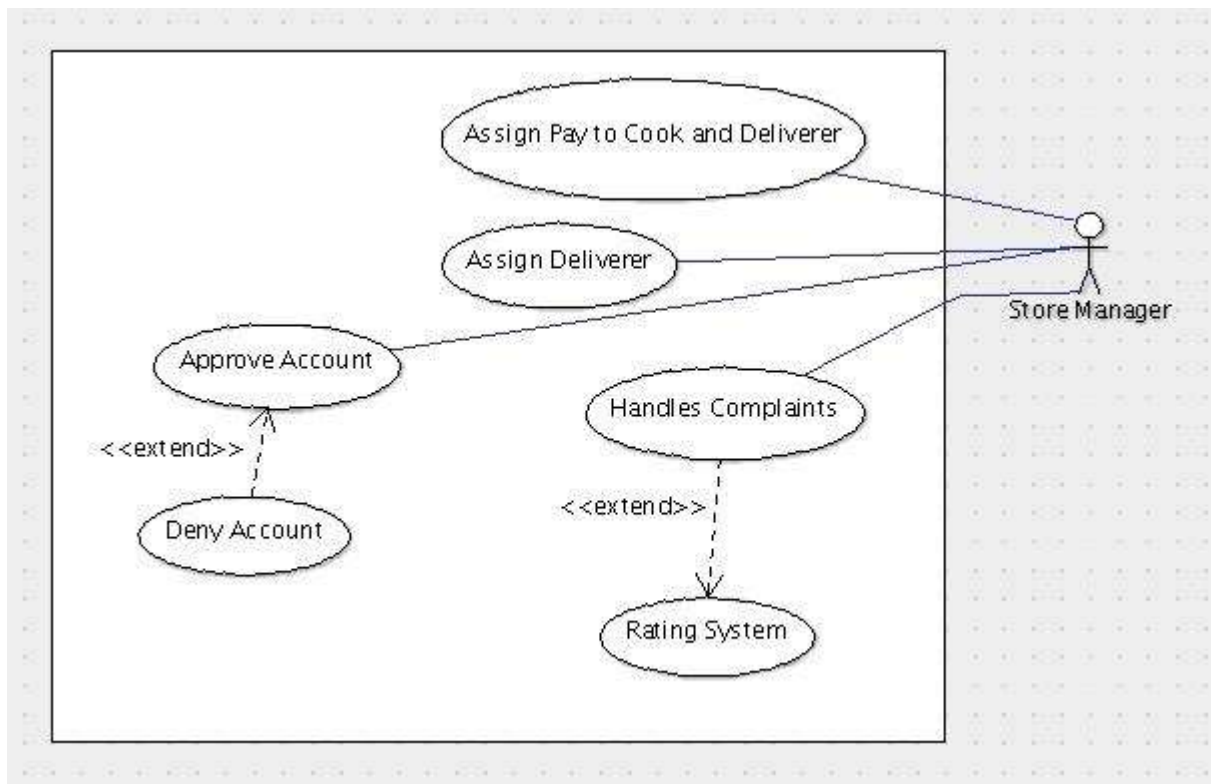
Use-Case: Receive Warning

Description: A pizza receiving an average rating less than 2 in the last 3 orders will be dropped. The cook whose pizza was dropped twice will be warned.

Use-Case: Get Laid Off

Description: A cook will be laid off the job if they get warned more than three times.

3.1.4 User Class 4 – Store Manager



Use-Case: Assign Pay to Cook and Deliverer

Description: Both the Cook and Deliverer salaries will be determined and distributed by the store manager.

Use-Case: Assign Deliverer

Description: The store manager will decide which deliverer to deliver the pizza to the customer.

Use-Case: Handle Complaints

Description: Any complaints from the customer about the pizza(s), delivery person or pizza shop are all handled by the store manager. The store manager will be able to gain access to view all complaints and solve them in a timely manner.

Use-Case: Approve Account

Description: Visitor who wants to register an account for the pizza shop will have to be approved by the store manager. Any visitor's account that is approved by the store manager will be a registered customer.

Use-Case: Deny Account

Description: The store manager will have to check the visitor's background and see if they are qualified to be a registered customer. If their background doesn't check out, their registration will be denied.

3.2 Supplementary Requirements

The lists below are not captured in the use cases but captures the set of requirements on the system.

3.2.1 Functionality

This section lists functional requirements that are common to more than one use case.

Errors - All system errors shall result in some sort of feedback to the user. The system errors shall return a text description of the error and steps to avoid it.

3.2.2 Reliability

This section lists the reliability requirements.

- The Pizza Delivery System should be available 24 hours a day, 7 days a week.
- All orders shall be logged and time stamped.

3.2.3 Usability

- The user interface should be designed for Ease-of-Use. The system should be usable for a user community that is computer-literate and shall require no additional training on the system.
- The user interface should also feature an online-help (step-by-step instructions) to ensure users can use the system with little to no effort.

3.2.4 Performance

- The system should support simultaneous users against the central database at any given time
- The system should be able to access the database within a short and logical latency
- The system must be able to complete most of the transactions within a short period of time.

4. Supporting Information

This Software Requirements Specification Includes:

- Table of Contents
- Index

The Pizza Ordering System described above is being developed by the Weirdoughs™ team – Gerry Xu, Melvin Tham, Tobias He and Ayushya Amitabh. Contact our team at tagmhxt@gmail.com.