The Meme Team

YUMMY CRUMMY

Software Requirements Document

Melissa Jenkins

Cassidy Lamm

Lauren Wood

09/25/2014

Change History

Version	Summary	Author	Date
0.1	Initial Draft	Melissa Jenkins	09-23-2014
0.2	Draft with added UML Diagram	Cassidy Lamm	09-24-2014
1.0	Finished Draft	Nico Wood	09-25-2014

Table of Contents

1. Introduction

1.1 Purpose

This document includes the requirements documentation related to the Yummy Crummy mobile application for the Android platform. It is intended to aid developers over the course of the creation of the application. It specifies the general features of the application as well as the functional and nonfunctional requirements, use cases, user activity flows, and UML documentation.

1.2 Scope

The Yummy Crummy application will provide a convenient, user-friendly interface through which users can keep track of which dishes their family likes and dislikes at different restaurants throughout the area. With each restaurant presenting a different variation of classic dishes, it is hard (especially for busy families) to keep track of which family member likes which dish at which restaurant. In the case of common allergens, it can also be difficult to remember which restaurants includes these allergens. Yummy Crummy attempts to alleviate these common pains by providing families with an easy, intuitive way to keep track of their food preferences.

1.3 User Characteristics

The intended users of this application are adults of all ages, backgrounds, and education levels. While the primary users will have young dependents, any individual could benefit from use of the application.

1.4 Definitions

Android: Google Inc.'s open and free software stack that includes an operating system, middleware, and also key applications for use on mobile devices, including smartphones.

Android SDK: A software development kit that enables developers to create applications for the Android platform.

Allergens: A substance that causes an allergic reaction

Dependent: A person who relies on another, especially a family member, for financial support

Filter: A device designed to sort through and present pertinent materials

User Interface: The means by which the user and a computer system interact, in particular the use of input devices and software

2. General Description

2.1 Product Features

The system is spread across two different domains: the mobile platform and the web portal. Through the mobile platform, users can create profiles, update their information, search for restaurants, and easily update their preferences. Through the web portal, users and the restaurants can update and maintain restaurant and menu information. Both of these domains will allow users to have access to the most recent and updated information on the restaurants in their community. On the mobile platform, there are four main features available to the users.

2.2 Website Interface

The main use of the website interface is to provide a convenient location for administrators, restaurants, and even users to provide and maintain information on the local restaurants. The interface can be accessed through a web browser and can be used to search for restaurants, view current information, edit information, and add new restaurants. Users will also be able to login to the interface to view their individual profiles.

2.3 Mobile Application

A mobile application will be the main way users access their information. The first thing they will see is a splash screen with the options of signing in or creating a profile. If they do not already have a profile they can create a new one either manually or by linking to Facebook. They are also able to create child profiles for each of their dependents. Once they have logged into the system, users have the option of updating their profile information, searching for restaurants, and updating their preferences.

2.4 Profile Creation

The user will be given the option of either manually filling in the necessary fields to create a new profile or choosing to login through Facebook and sharing their information. This will be the way they will access all of their preferences and interface with the system.

2.5 Updating Profile

Once they have logged into the system, users have the option of updating their profiles. They can change any of their information such as name or location. They also have the option of creating new profiles for their dependents. They have the option of setting up different levels of parental security based on how much freedom they want their dependents to be able to interface with the application. Once these dependent profiles have been created, the dependents will be able to log in and use the simplified child interface.

2.6 Restaurant Search

Users will be able to search for restaurants based on location, restaurant name, or recent searches. At first, the application will be limited to the Tuscaloosa area, but as the user base grows more cities will be added to the system. Once the user has chosen their restaurant they will be able to view the menu and dish ratings.

2.7 Updating Preferences

Users will have the option of updating either their own or their dependents' food preferences at any given restaurant. If they liked the dish they can give it a thumbs up, if they did not like the dish they can give it a thumbs down. In the future the application will be expanded to allow adults a more sophisticated rating system.

3. Functional Requirements

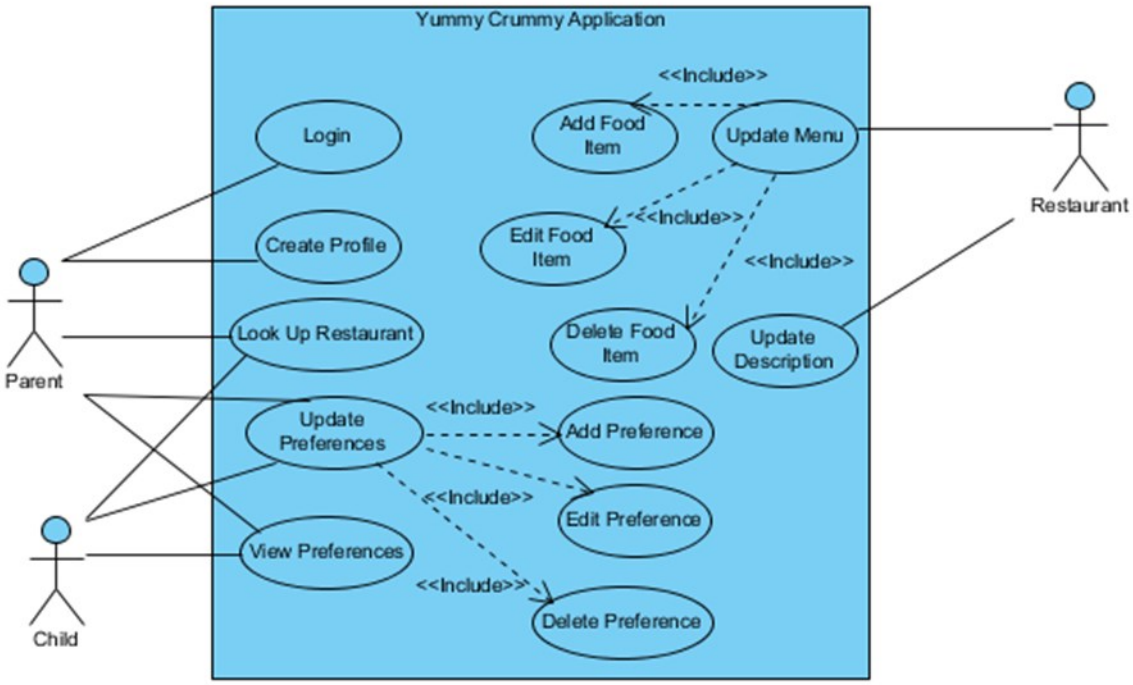
1. A user shall be able to create a personal profile either through the mobile application or on a traditional personal computer.
2. A user shall be able to create profiles for their dependents either through the mobile application or on a traditional personal computer.
3. A user shall be able to login to the system to view and edit their profile.
4. A user shall be able to search for restaurants in their area and view the menus.
5. A user shall be able to see other user ratings for each dish at a restaurant.
6. A user shall be able to add or edit a restaurant menu either though the mobile application or on a traditional personal computer.
7. A user shall be able to record whether or not they or their dependent like a dish at a certain restaurant.

3. Non-Functional Requirements

1. The system must provide a user-friendly, convenient interface for the parent user.
2. The system must provide an extremely simple and basic interface for the child user.
3. The system must work without any significant lags in the system.

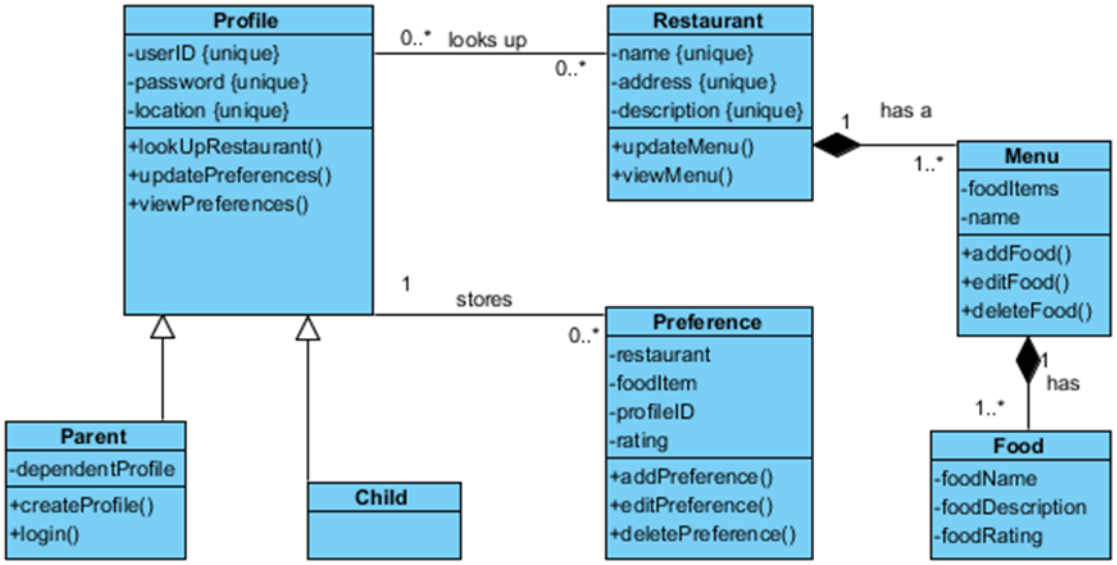
5. UML Diagrams

5.1 Use Case Diagrams



In our Use Case Diagram, there are three actors that have roles within the Yummy Crummy Application. The first is the parent, which is one of the types of users. The parent role can login to the system, look up a restaurant, update their preferences by adding, editing, or deleting them, view their preferences, and create a profile. The second actor is the child. The child's use cases are similar to the parent's except for the fact that the child cannot log into the system or create a dependent profile. The third actor is the restaurant which can update its menu by adding, editing, or deleting food items as well as update its description on its page.

5.2 Class Diagram



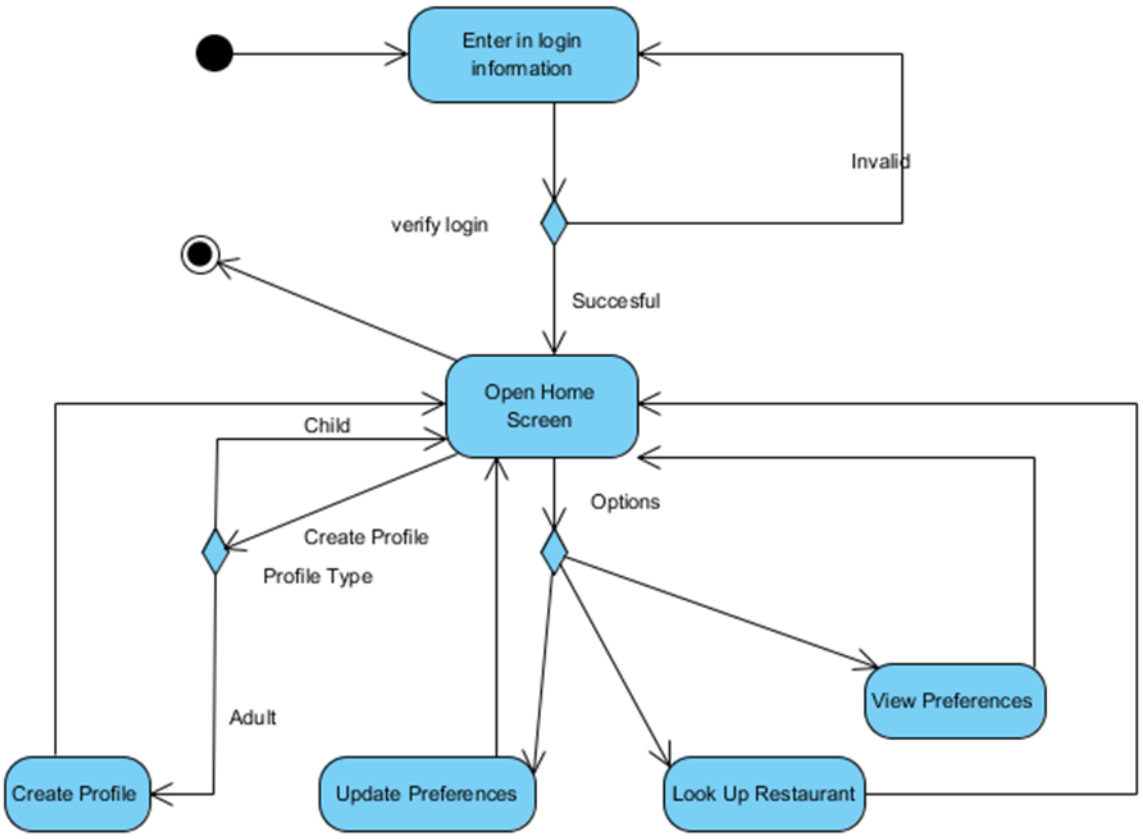
In our class diagram, we have seven classes that make up the Yummy Crummy system. The first and foremost is the Profile class. This class represents the individual and dependent profiles of each of the users. The Profile class has three attributes: userID, password, and location which are unique to each user. Within the profile class, the user can look up a restaurant, update their preferences, and view their preferences. There are also two classes that inherit from this class: the Parent and the Child. The Child class will inherit all attributes and operations from the Profile class, but will not be able to do anything else. The Parent class, on the other hand has dependent profiles and can create new ones as well as login to the system with its credentials.

Each profile can store zero to many preferences as shown by the Preference class. Each preference will be characterized by a particular restaurant, a food item, the profile ID, and the rating of that food item. This is shown in the four attributes restaurant, foodItem, profileID, and rating. A preference can also be updated by the parents in which it will either add a new preference, edit one or more of a current preferences' attributes, or delete a preference. This is shown by the three operations addPreference, editPreference, and deletePreference. Each instance of preference can only be associated with a single profile.

A profile can also look up a restaurant which constitutes a Restaurant class. A restaurant has a name, an address, and a description, which are all represented by its attributes. A restaurant also has a menu, which is depicted as a separate class since it is long term and can be changed. The restaurant can view or update its menu as shown in its operations. A restaurant can be looked up by zero to many profiles and a profile can look up zero to many restaurants which is also depicted in the diagram. The menu class is only associated with a single restaurant, but each restaurant can have multiple menus (i.e. children's menu, lunch menu, dinner menu, etc.). Each menu has its own name and list of food items, and can add, update, and edit each food. A menu has one to many food items, but each food item is only associated with one menu. Each food item has a unique name, description, and food rating.

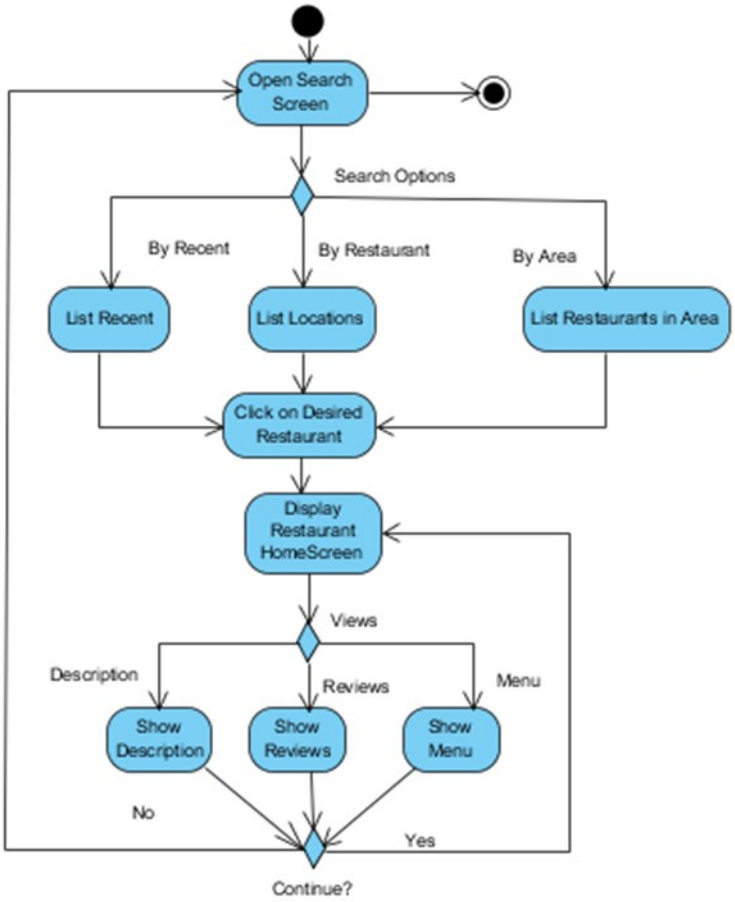
5.3 Activity Diagrams

5.3.1 Main Activity Diagram



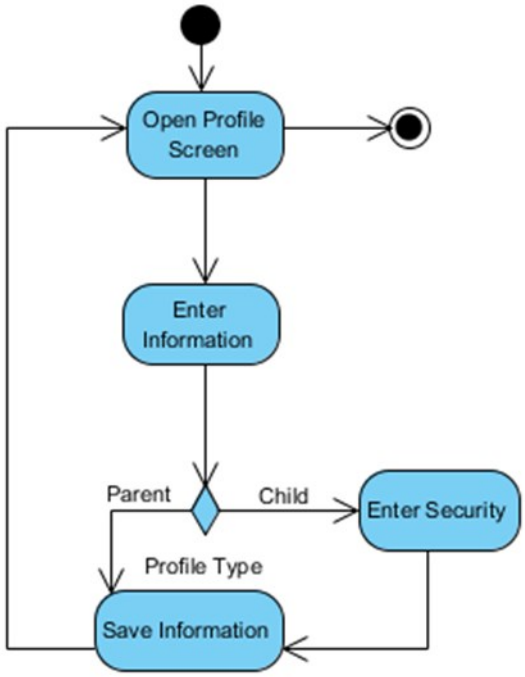
The activity in the Yummy Crummy system begins with the user entering in his or her log-in information. If the information is successful, the user is then presented with the home screen of the application. However, if the login is invalid, the user will be prompted to try again. Once in the home screen, the user has several options. He or she can update preferences, view preferences, look up restaurants, or create a profile depending on whether he or she is a child or adult profile. After the user has completed zero or more of these activities, he or she can exit the system.

5.3.2 Look Up Restaurant Activity Diagram



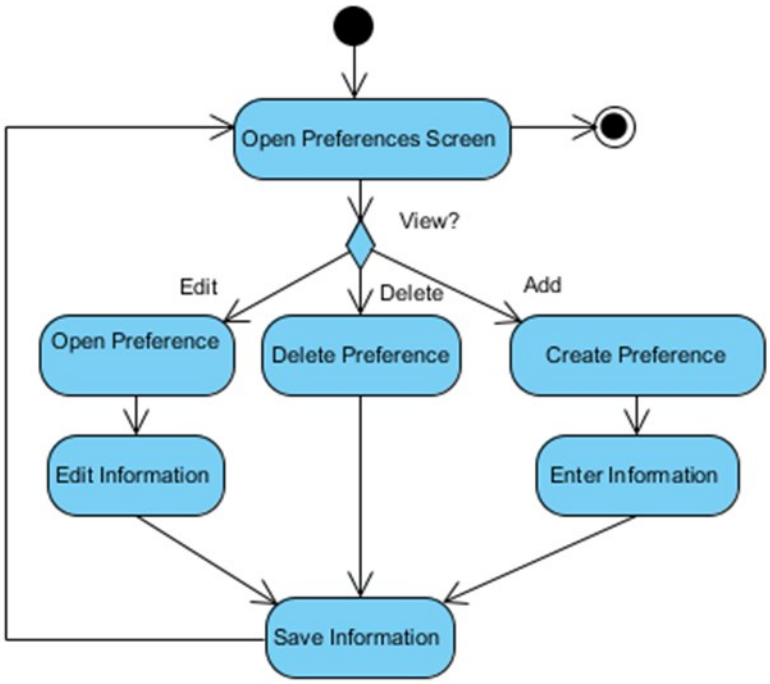
Once the user has chosen to look up a restaurant, the search screen will appear. The user will then have the option of searching by location, by restaurant name, or by his or her recent searches. After listing the restaurants in whichever manner the user asked for, the app will then display the page for the restaurant. From the restaurant's home screen, the user can view the menu, the user reviews, or the description. If they wish to continue they can go back to the restaurant's home page, preform a new search entirely, or exit the use case.

5.3.3 Create Profile Activity Diagram



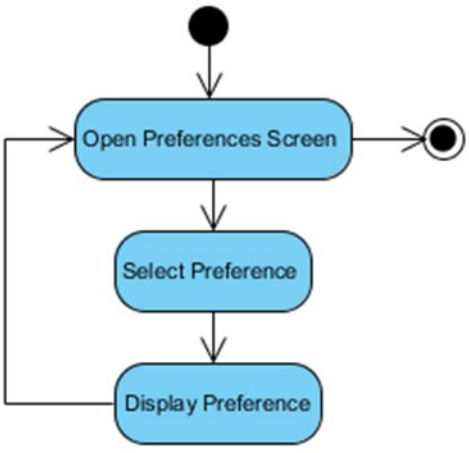
Once the user chooses to create a profile, the Yummy Crummy app will open the profile screen. Next, the app will prompt the user to enter in his or her basic information. If the profile created is a child's profile, the user can proceed to enter in the appropriate security restrictions and save the profile. On the other hand, the parent profile can immediately be saved and the user can either create another or exit the use case.

5.3.4 Update Profile Activity Diagram



If the user has chosen to update his or her preferences, the application will proceed to open the preferences screen. From here, the user can choose to open a preference and edit the information, delete a preference, or create a new preference. Once the necessary information is stored, it will be saved and the user can either start the process again or exit the use case.

5.3.5 View Preferences Activity Diagram



If the user chooses to view his or her preferences, the preferences screen will be displayed. Then the user can simply select whichever one he or she wishes to view. The application will then display the information associated with that preference and the process can be repeated or the user may exit the use case.