COMS3009: Software Design

Requirements Analysis Document

The Restaurant Management System

Team Members

Mthulisi Leslie Zimba:	**570937**
Kgopotso Dilebo:	**715636**
Lethabo Nkabinde:	**722211**
Christopher Mashele:	**732475**
Fortune Ndlovu:	**731603**

WITS
UNIVERSITY

School of Computer Science and Applied Mathematics
University of the Witwatersrand
South Africa

October 21, 2016

Executive Summary

The following document details the results of requirements gathering that was carried out for the Restaurant Booking System. All the functional and non-functional requirements are completely described. This document will also serve as a contractual basis between FLICZ and Leslie the product owner.

Audience

- System Architect (Lethabo Nkabinde)
- Product Owner (Leslie Zimba)
- Software Programmers (FLICKZ)

1 Introduction

1.1 Purpose Of The System

It is very frustrating as a customer to arrive at a restaurant to and find that all tables are taken, and you have to wait for hours before you can get a table. Sometimes even if you try to book in advance by calling the restaurant, the call does not go through or is never picked up.

The purpose of this project is to create a system that will replace the human effort, on the restaurant staff side, of having to manually take bookings.The system will enable customers to make restaurant bookings via their android mobile devices wherever they are.

1.2 Scope Of The System

The Restaurant booking system is a sofware application from which patrons will be able to book tables, view the menu, order food, make bill payments and even rate the restaurant‘s service.

The system will be running on android mobile devices.

1.3 Objectives and success criteria of the project

The main objective of this project is;

- To create a system that will enable customers to book tables online

Additional features of the system are;

- Users can view the menu and specials for the day
- Users can keep track of their bill

- Users can order online
- Users can check whether their table is ready for occupying

During the process of designing this system, the software designers will be liasing with the restaurant staff and willing restaurant customers constantly to test the current prototype.

Success will be achieved if the following goals are met;

- Users are happy with and can easily use the interface without any training
- The instruction manual answers all user queries
- The system eliminates any kind of interaction between restaurant staff and customers where booking of tables is concerned
- More than 80 percent of the test subjects approve and say they will be using the application everytime they visit the restaurant

1.4 Definitions, acronyms and abbreviations

User Anyone registered on the system

Interface Visual part of the application which the user will interacting with

Actor Any authorised member using the system

UseCase List of Actions defining interactions between Actor and System

UML Unified Modelling Language

1.5 References

- Test Driven Design Document
- Software Engineering, Ivan Masic

1.6 Overview

Details of how each UseCase will be tested are documented on the Test Driven Design Document.

2 Current system

Currently, restaurant bookings can only be made via a phone call or directly at the restaurant. The restaurant has only a single extension for this and during peak periods, the staff fails to handle all potential customers booking on the

phone.

The menu is printed on a cardboard paper and updating it requires printing a new set of menus and laminating them.

After decicding on what to eat, the customer has to wait for the waiter to come back to them and take their order.

The customer then asks for and pays their their bill after they are done eating.

3 Proposed System

3.1 Overview

The whole aim of the proposed system is to automatize most of the restaurant‘s processeses so as to make the restaurant run more efficiently.

3.2 Functional Requirements

3.2.1

The system shall allow restaurant staff and customers to register as users.

3.2.2

The system shall allow restaurant staff and customers to log in.

3.2.3

The system shall give staff and customers different user priviledges.

3.2.4

The system shall make available an updated menu for users to choose what to order.

3.2.5

The system shall allow users to make table bookings.

3.2.6

The system shall enable users to order food online.

3.2.7

The system shall .

3.3 Non Functional Requirements

3.3.1 Usability

- The system must be usable on more than 70 percent of Android devices
- Instructions on the system must be unambiguous
- The user must be able to log in and out of their account on any device

3.3.2 Scalability

- The system must be able to handle the growing number of new users on a daily basis without losing perfomance since one of the aims of the restaurant is to get as many customers as possible.

3.3.3 Reliability

- The software must be bug free by the time it is distributed for public use.
- Each user account must be secure and unique.

3.3.4 Perfomance

- Graphics and text must be clear.
- Logging in must be instantaneous with no flaws

3.3.5 Supportability

- On the first release, the application must run on any Android KitKat or later releases

3.3.6 Implementation

- The project must be implemented on Android.

3.3.7 Interface

- The interface must be attractive, simple and easy to use.

3.3.8 Packaging

- The System must be packaged and distributed freely on the Google Play Store

3.3.9 Legal

- The system must direct the user to the Terms and conditions page which the user must read and accept before installing the application.

3.4 System Models

3.4.1 Scenario

1. Bob wants to make an online reservation to dine at FLCKZ restaurant in the evening.
2. He acquires the Restaurant booking system on google store using his mobile phone and installs it.
3. He signs up by proving his identification details.
4. He uses the details he provided to identify himself and gain access to the system.
5. He makes a reservation for a specific date and time.
6. He then closes the system.

3.4.2 Use Case Model

- Requirements table

REQ 1	Sign up a customer or staff as a user	UC-1 - Register
REQ 2	Give access to the user when the enter correct credentials	UC-2 - Login
REQ 3	Allow user to make a reservation	UC-3 - Book
REQ 4	Let the user view the menu	UC-4-View Menu

- Use Cases

Actor	Actor‘s Goal	Use Case Name
User	To register as a new user	Register (UC-1)
User	To log in to the system	Login (UC-2)
User	To Make a booking	Book (UC-3)
User	To view the Menu	Menu (UC-4)

- ***Use Case 1 : Register***

Use Case UC-1	: Register
Related Requirements	REQ 1
Initiating Actor	User
Actor‘s Goal	To provide user details so as to be identified when using the system.
Participating Actors	Users, Database
Preconditions	System does not recognise new user and displays login page with button to register as a new user
Postconditions	User information is stored in the database.

Flow of Events for Main success Scenario
→ **1**. User Clicks on the register button
← **2**. The System displays a register page
→ **3**. User enters valid information in all the required fiels
← **4**. System displays "Registration successfull" and then displays the login page

Flow of Events for Extensions (Alternate Scenarios)
→ **3a**. User enters wrong information in one of the fields
← **4a**. System displays "Error" and notifies the user of their mistake
→ **3**. User then enters valid information in all the required fields
← **4**. System displays "Registration successfull" and then displays the login page

- ***Use Case 2 : Login***

Use Case UC-2	: Login
Related Requirements	REQ 2, REQ 1
Initiating Actor	User
Actor‘s Goal	To enter correct credentials and gain access to the system.
Participating Actors	Users, Database
Preconditions	System displays home page from which a user can log in
Postconditions	User has gained access to the system using their username and password

Flow of Events for Main success Scenario
→ **1**. User Clicks on the Login button
← **2**. The System displays a Login page
→ **3**. User enters valid username and password
← **4**. System displays "Welcome" and gives user the permission to make reservations or transactions

Flow of Events for Extensions (Alternate Scenarios)
→ **3a**. User enters wrong username or password
← **4a**. System notifies the user and prompts them to re enter correct information
→ **3**. User then enters valid username and password
← **4**. System displays "Welcome" and gives user the permission to make reservations or transactions

- ***Use Case 3 : Book***

Use Case UC-3	: Book
Related Requirements	REQ 3
Initiating Actor	User
Actor‘s Goal	To make a reservation to the restaurant for a specific date and time.
Participating Actors	Users, Database
Preconditions	System displays booking page
Postconditions	System indicates on user profile that they have made a reservation

Flow of Events for Main success Scenario
→ **1**. User Clicks on the Book button
← **2**. The System displays a Book page
→ **3**. User chooses date and time to visit the restaurant and clicks book
← **4**. System confirms reservation

Flow of Events for Extensions (Alternate Scenarios)
→ **3a**. User choose time when restaurant is closed
← **4a**. System prompts the user to enter a different time
→ **3**. User then enters valid date and time
← **4**. System confirms reservation

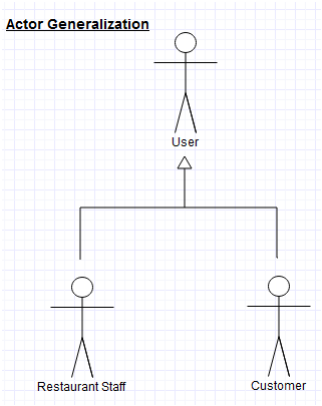
- ***Use Case 4 : Menu***

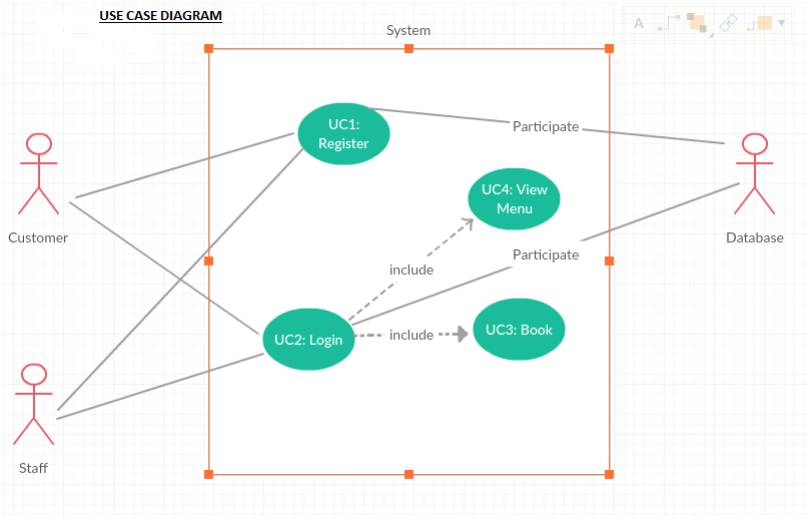
Use Case UC-4	: Menu
Related Requirements	REQ 4
Initiating Actor	User
Actor‘s Goal	To view type of food and drinks offered by the restaurant.
Participating Actors	User, Database
Preconditions	System is logged into a profile
Postconditions	System displays available menu

Flow of Events for Main success Scenario
→ **1**. User Clicks on the View Menu button
← **2**. The System displays a Menu page

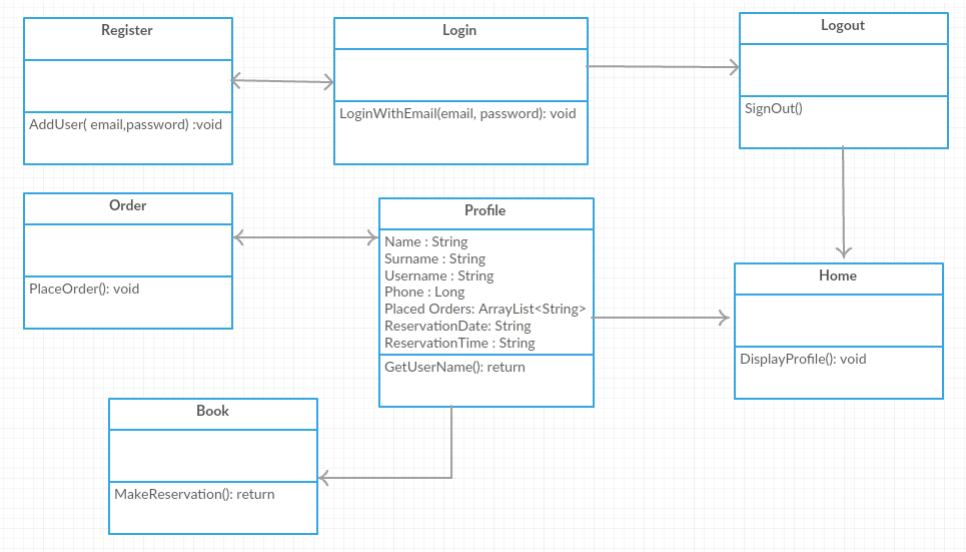
Flow of Events for Extensions (Alternate Scenarios)

- Use Case Diagram

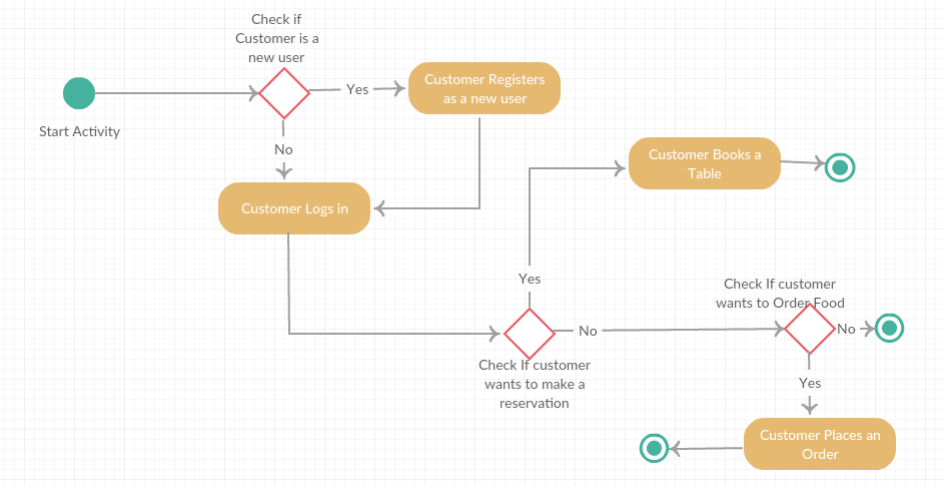




3.4.3 Analysis Object Model



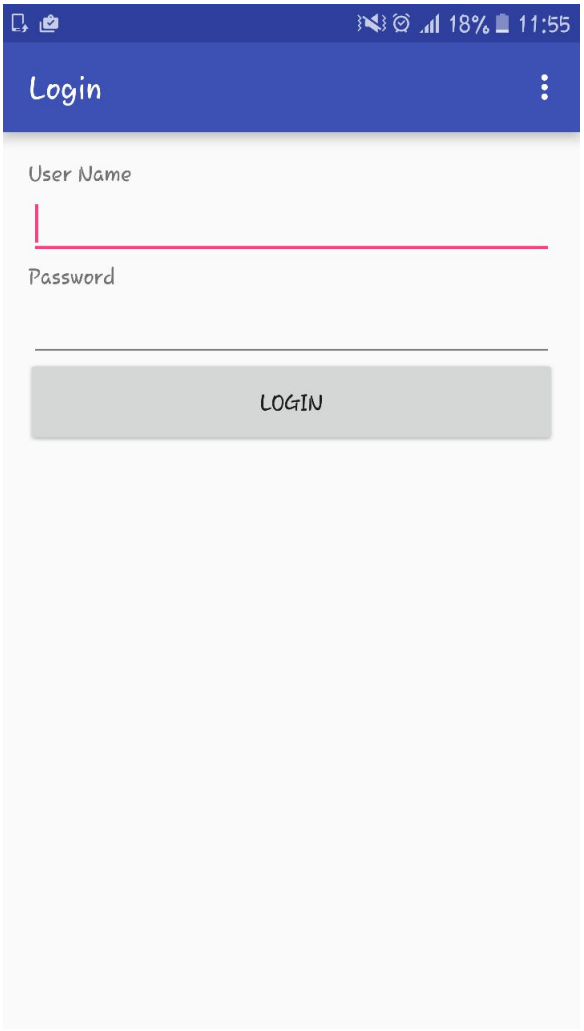
3.4.4 Dynamic Model



3.4.5 User interface - navigational paths and screen mock ups

Menu Page *Book*

Login Page



After Booking