University of Zurich[UZH]

MASTER PROJECT – Communication Systems Group, Prof. Dr. Burkhard Stiller

Design, Implementation, and Evaluation of a Menu Management System for Restaurants

Maxat Pernebayev
Shymkent, Kazakhstan
Student ID: 11-715-216
Tanvi Singh
New Delhi, India
Student ID: 11-777-182
Karthick Sundararajan
Palladam, India
Student ID: 11-722-857

Supervisor: Christos Tsiaras, Dr. Martin Waldburger,
Prof. Dr. Burkhard Stiller
Date of Submission: July 1, 2013

University of Zurich
Department of Informatics (IFI)
Binzmühlestrasse 14, CH-8050 Zürich, Switzerland

ifi

Master Project
Communication Systems Group (CSG)
Department of Informatics (IFI)
University of Zurich
Binzmühlestrasse 14, CH-8050 Zürich, Switzerland
URL: http://www.csg.uzh.ch/

Abstract

In competitive business world, reducing operational cost and increasing productivity become paramount. Particularly, manpower is more expensive for restaurants in Europe. In this situation, an automated system can help to increase process efficiency, decrease order processing time and guarantee better Quality-of-Experience (QoE) for customers. This master project focuses on developing a menu management system and evaluating its effects on business process efficiency and customer satisfaction. It has three core items: requirements analysis, implementation and evaluation.

As a starting point, a real business setting of a restaurant is identified based on selection criteria. The requirements are elicited from the partner restaurant and user survey. The user survey is conducted for understanding the users' preferences on using mobile applications. The system is then designed and developed based on identified requirements, user survey and design principles of client-server application. The evaluation shows the effects of introduction of the automated system on order processing time and customer satisfaction. The system is evaluated using Six Sigma methodology [1], and statistical calculations are done using Minitab tool [2].

The results show that average order processing time for drinks orders is reduced after the introduction of the automated system. Also, the customer QoE measured by evaluation forms has an above average score. Therefore, it is deduced statistically that the introduction of the automated system reduces the order processing time in the experimental setup.

Kurzfassung

In einer konkurenzbetonten Geschäftswelt rücken die Reduzierung der Betriebskosten und die Steigerung der Produktivität immer mehr in den Vordergrund. Dem entsprechend sind für Restaurants in Europa die gestiegenen Personalkosten ein massgeblicher Faktor. Automatisierte Systeme können bei Entwicklung durch effizientere und schnellere Bestellabwicklung und einer daraus resultierenden gesteigerten Kundenzufriedenheit entscheidene Konkurenzvorteile realisieren. Bestandteile der vorliegende Masterarbeit sind die Entwicklung eines Menübestellungssystems und die Analyse der Auswirkungen bezüglich Effizienz und Kundenzufriedenheit. Die Schwerpunkte der Arbeit liegen auf der Anforderungsanalyse, der Implemtierung und der Nutzeranalyse.

Die konkreten betrieblichen Anforderungen wurden in Zusammenarbeit mit einem Partnerrestaurant ermittelt, welches den am Projektstart festgelegten Auswahlkriterien entsprach. Die Ergebnisse durchgeführter Kundenbefragungen ermöglichten die Ermittlung der Nutzerbedürfnisse. Diese aus der Praxis gewonnenen Anforderungen und die Prinzipien der client-server Architektur bildeten die Basis für das Design des Systems. Durch die abschliessende Nutzeranalyse wurden die Auswirkungen des implementierten Systems auf die Bestellabwicklung und die Kundenzufriedenheit quantifiziert. Als Methoden zur Bewertung wurden Six Sigma methodology [1] und Minitab tool [2] eingesetzt.

Nach der Einführung des automatisierten Systems wurden Bestellungen für Getränke im Durchschnitt messbar schneller abgewickelt und das bei einem ebenfalls messbaren Anstieg der Kundenzufriedenheit.

Acknowledgement

We want to thank Prof. Dr. Burkhard Stiller for encouraging and motivating us to pursue our interest and idea into a full-blown master project. With this we got an opportunity to firsthand experience how concepts can be turned into real world working projects. Dr. Martin Waldburger has guided us ever since this concept has turned into a master project. He has been a true mentor and has always warned us of all eventual pitfalls. Christos Tsiaras, has guided us technically and made us take the right decisions at the right time. We thank Guilherme Machado for keeping our development environment ready and also being available on odd times and weekends. Next we want to thank Dr. Elaine Huang at UZH for her valuable time and advice in giving feedback and suggestions on mobile application's usability perspective.

We thank the manager of McGee's Irish Pub and Restaurant, Vishnu Prasath Sachithanatham, and his staff for giving us valuable requirements and feedback on end user perspective. We actually implemented the mobile application in their restaurant and could get a real world proof of our hypothesis.

Last but not least we want to thank our respective families for being patient with us during the project work and motivating us through the journey.

Table of Contents

Chapter 1. Introduction and Motivation

The biggest challenge for businesses is the reduction of operational costs as well as increased productivity [3]. Focus on business process efficiency and minimization of human-related errors may influence costs and productivity positively. In this scope, automation might lead to faster process execution and to reduced error rate caused by the human factor, especially in environments with multiple hops that information or data has to pass from customer to employee, and the other way round. Such an environment may be found in restaurants where a need for high customer Quality-of-Experience (QoE), low operational costs as well as high turnover is mandatory in order to achieve high degree of customer satisfaction as well as high level of productivity.

In general, people go to restaurants for relaxing, chatting and having food/drinks pleasantly. Usually on the weekends the restaurant are fully occupied. At this time, people have to wait for someone (typically a waiter) from the restaurant to order food/drinks. In addition, waiters are very busy when the restaurant is crowded. Sometimes they might forget to take orders from customers, forget the orders, serve wrong order and deliver after long time. When the restaurant introduces a new menu or some recipes are new for customers, they do not understand it well by just seeing the menu card printed with a food/drink name.

Since manpower is one of the most important cost factors in restaurants, and at the same time a key reason for altered performance, an automated order taking process might work as a solution. An automated solution can be assumed to facilitate an increase in overall productivity by decreasing the time and effort involved in this procedure, while keeping customer satisfaction at the same level, or even increasing it. Increased customer satisfaction might, for instance be enabled in multi-language environments, like big cities, where it is hard to serve customers in their preferred language. Thus, existence of a multi-language-friendly order-taking process might be essential for many customers.

This project aims to design, to implement and evaluate a menu management system for an identified restaurant. The system provides automated order taking and menu management functionalities for the restaurant. Firstly, the project starts with requirements analysis in collaboration with the restaurant. Secondly, based on the requirements the menu management system is designed, implemented and tested. Finally, in evaluation part, the system is analyzed to study the effects of introduction of the menu management system in the restaurant. This evaluation part shows whether the system decreased the order processing time and customer QoE in the restaurant. This chapter begins with motivation and objective of the master project. After that it explains scenario scope and structure of the project report.

1.1 Motivation

There are two main motivations for this master project:

1. Utilize manpower efficiently in restaurants to reduce operational cost and increase productivity.

2. Increase customer QoE in restaurants by providing a digital menu card with pictures and videos.

1.2 Objectives

There are two main objectives for this master project:

1. To design, implement and test the menu management system based on a set of requirements identified in collaboration with the restaurant.
2. Conduct a study to analyze whether the menu management system made an effect on order processing time and customer QoE in the restaurant.

When a restaurant is fully occupied, waiters spend more time to take, to process and deliver the orders. Because of this, customers in the restaurant also do not have high QoE. Like the WinWin methodology in software engineering [4], this system focuses on both the restaurant's and customers' problems. One Win for restaurants to process and deliver it faster and another Win for QoE of customers. By considering these objectives, the master project proceeds with scenario scope in the next section.

1.3 Scenario Scope

The main goal of this project is to study the effects of introduction of the menu management system on order processing time and customer QoE in the restaurant. Here, objective is not to provide a complete solution for the restaurant. The scope of this master project is shown in Figure 1-1.

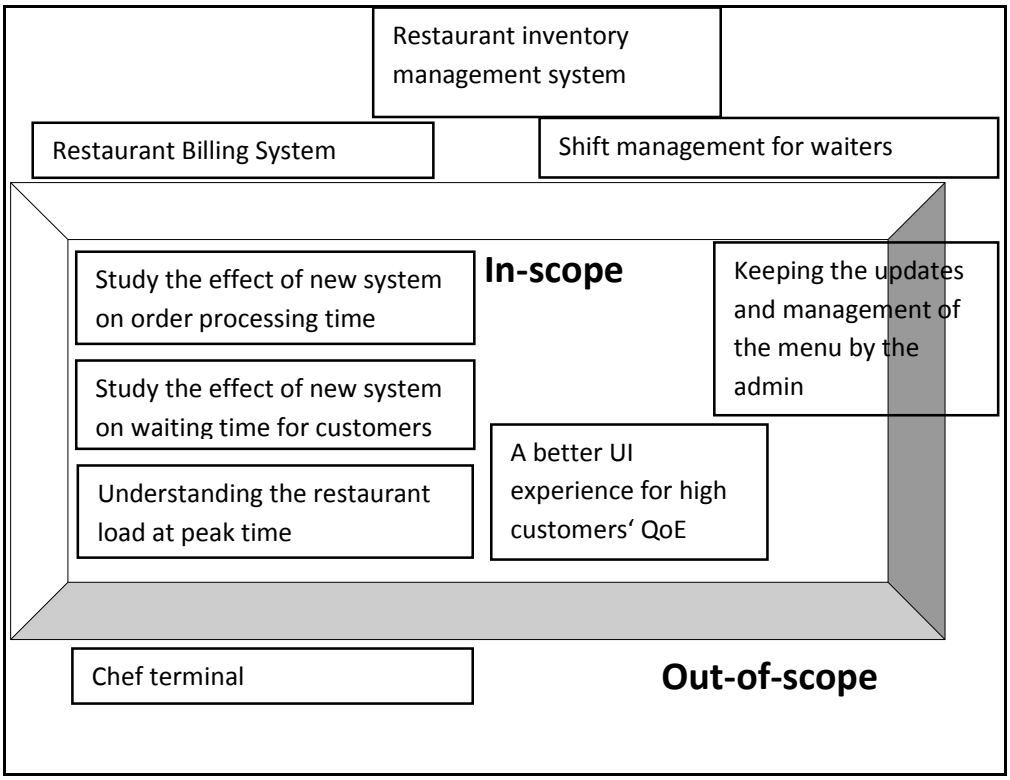


Figure 1-1 In-scope and Out-of-scope

In-scope of this project are:

- Study the effects of new system on order processing time,
- Study the effects of new system on waiting time for customers,
- Understand the restaurant load at peak times,
- A better user interface design to give high customer QoE,
- Keeping the menu management system up-to-date and
- Managing waiters by a restaurant admin.

The scope does not include inventory management system, restaurant billing system, shift management for waiters and chef terminals. The system can update the menu if and only if the restaurant management intends to change it through the admin console. The change reflects only on the system, but it does not provide an additional option to update physical menu card in the restaurant. Thus, the goal *Keeping the updates and management of the menu by the admin* is in between in-scope and out-of-scope.

1.4 Report Structure

The master project selects a restaurant to elicit real requirements, conducts a user survey to understand the customers, and then provides a solution for both customers and restaurant. It has three core items: requirements analysis, implementation and evaluation. The approach of the project is explained in Chapter 2. In chapter 3, the requirements are elicited from a selected restaurant and user survey. Chapter 4 discusses design of the system, communication architecture, an entity-relationship diagram, user interface design, workflow of the system and selection of technology. The system is designed and implemented based on the requirements. Chapter 5 details study design, which is one of the core work items of the project. It represents the culmination of this project as it provides the evaluation of the system. After that, implementation and deployment of the system is discussed in Chapter 6. Here, the core functionalities of the system and technologies used in this project are discussed. Chapter 7 presents results and analysis of the system, including pre-deployment and post-deployment data analysis. Finally, Chapter 8 closes the project with summary, overall conclusions, future work, learning and final recommendation.

Chapter 2. Approach

In this chapter, the approach of the overall project is described. The project addresses the objectives defined in Chapter 1 for the restaurant environment. Hence, the requirements of the project are elicited by selecting a restaurant based on certain key criteria such as areas of operation, seasonality and size as explained in details in Chapter 3. The other key actor in the project is customers, and the project team also elicited the requirements from them by conducting a user survey. Therefore, the project team collected the real requirements specification from both the representative restaurant and the customers.

In this project, several technologies have been used to meet the elicited requirements, as collected from the identified restaurant and user survey. One of the decision making Six Sigma tools "Pugh Matrix" [1] is also used for making these key technology decisions during the design phase of the project. Firstly, for identifying the location of a customer in the restaurant, QR code (2D barcode) has been used. This allows the system to identify the location of customers in the restaurant. Secondly, in order to send/receive notification between the customers and waiters in the restaurant, real-time communication is implemented using SignalR framework which is described in Section 6.1.1. Thirdly, web service of this system is designed and implemented based on the RESTful architecture style [5], which allows the system to a support native mobile application in future. Similarly, the user interface is designed and implemented based on the Model View ViewModel (MVVM) pattern, which clearly separates user interface components and user interface code.

The project implementation is tested based on concepts of scientific work through an experiment. The evaluation is conducted at the selected restaurant, and order data is analyzed in comparison to historical data based on the study design described in Annex B. This is an approach usually taken to prove the hypothesis during clinical trials but used effectively here in the area of software implementation. The statistic data analysis has been done using Six Sigma methodology [1] implemented using Minitab tool [2]. The hypothesis and experiment are described in detail in Chapter 5 and statistical results are described in Chapter 7.

Chapter 3. Requirements

The concept of menu management system to be developed is to reduce manual efforts in the restaurant ordering system. It is important that such system is deployed in a real environment and hence the requirements are not abstract and are collected from the real setup.

This chapter has the following structure. The general requirements explain the logical setup of the system and describe the basic functionalities needed for such system. The functional requirements are detailed with the help of use case diagram. They are divided into "Customer", "Waiter" and "Admin" subsections. The system that can be used by any customers without training in multiple setups such as restaurants and bars, has to be developed keeping many non-functional requirements in mind. These non-functional requirements were considered in the very beginning of the project and are explained in details in the following sections.

3.1 General Requirements

The restaurant should be selected to conduct the proof of concept in accordance with the objectives defined earlier. The requirements are then collected in close association with the selected restaurant, in the real business setup. It was also kept in mind that, these requirements are not specific to the one setup, but are collated in a way that it can be generalized to adapt to more setups in the future.

3.1.1 Restaurant Selection for Implementation

To develop the proof of concept, the project team worked on finding a restaurant, which qualified the following criteria to represent the business side of the restaurant before starting the requirements.

The establishment should have a seating area where a waiter serves food/drinks on the table. A self-serving food/drinks bar/restaurant such as a food court was ruled out, since the mobile application's objective of reducing the order processing time would not have met in such a setup. The customers can walk-in to the restaurant without reservation is an important requirement. If the number of customers is unpredictable, it leads to uncertainty in waiters' effort. The restaurant should have a seating of more than 50 people, so it is hard for the waiter to make frequent visits to the tables for order taking and serving. The restaurant should have more than one area of operation, thus making a single waiter to leverage the area time consuming. A restaurant either spread over two floors or having different sections (indoors and outdoors) would be ideal for the mobile application deployment. The restaurant management should agree to conduct the experiment with the new system in their premises and are ready to invest time and effort from their staff.

The project team looked at setups and establishments that could qualify on the criteria described above. McGee's Irish Pub and Restaurant qualified all the requirements and the project team worked on the requirements with its management. The restaurant is situated in a very busy area and is a big place with 150 seats, where a waiter serves food on

customers' seats. The restaurant has two distinct seating areas. The inside seating area with around 100 seating places is open all days and all seasons. The inside area is divided into tables and the bar. The restaurant management agreed on time and effort from the staff, and the implementation of the mobile application on their premises. The restaurant management shared their menu cards with the project team for understanding the basics on menu hierarchy and structure. The team worked closely with the restaurant management to define the functional requirements. Regular meetings on alternate weeks were set with the restaurant management to have seamless communication throughout the duration of the project.

3.1.2 User Survey

Since the customers are arbitrary for this mobile application, conducting a survey also contributed to the requirements specification from the potential customers end. The usage of the application is based on the user acceptance of the system design. Therefore, to collect as much data as possible the project team made the survey to understand what the customer prefers.

Their preferences with regard to technology and usability are considered as an input to the final selection of the technology and the features discussed in this section.

The survey is designed with five objective questions and an open-ended subjective question to understand the customer choice in different areas with respect to the application. The survey was rolled out to arbitrary people and the respondents were not categorized based on age, occupation or otherwise. The only criterion for a valid response considered in the results is that the respondent is a frequent visitor to restaurants. This way the project team can consider them as a representation of potential customers. The survey was responded by 132 people in less than a week and gave the intended information that acted as an input to the technology decision matrix and the feature selection in the application. Here are the questions and the analysis in details:

Question 1:

Table 3-1 Result of Question 1

What kind of mobile phone do you have?		
Answer Options	**Response Percent**	**Response Count**
iPhone	**39.7%**	**52**
Android	**35.1%**	**46**
Windows Phone	**3.8%**	**5**
Blackberry	**2.3%**	**3**
Other	**22.9%**	**30**
Other (please specify)		**31**
	Answered question	**131**
	Skipped question	**1**

This first question is to understand the topology of the potential customers in general and what kind of phone should be concentrated for the usability design, development and testing. As can be seen from Table 3-1, iPhone and Android account for 75% of the representative users that is 98 out of 131 responses, and hence should be the primary platform for development and testing. This does not infer that the design can exclude the other platforms and mobile devices but the main focus for testing should be iPhone and Android.

Question 2:

The second question is used as an input for technology selection. Everyone who answered the survey as shown in Table 3-2, goes to a restaurant at least once in a month and hence is the target responder. As a result, all responses are considered and none of them are discarded. On an average, the responders go to the restaurant 5 times in a month, which is very frequent. Therefore, the responders are a good target group for the preferences to the application.

Table 3-2 Result of Question 2

In the past month how many times have you been in a restaurant?			
Answer Options	**Response Average**	**Response Total**	**Response Count**
In Numbers (1 to 20)	**527.0%**	**696**	**132**
	Answered question	**132**	**132**
	Skipped question	**0**	**0**

Question 3:

The next question focuses on the preference of the customer to access the application and tries to indicate the user behavior with respect to the smartphone usage. The question is designed in the form of a scenario so the user does not answer a default selection but can voice their exact choice.

Table 3-3 Result of Question 3

Your favorite restaurant provides you the two options to view that brand new digital menu in your mobile device. Which one would you like to choose?		
Answer Options	**Response Percent**	**Response Count**
View digital menu in web browser	**57.6%**	**76**
Install an App (Mobile Application) to view digital menu.	**42.4%**	**56**
Answered question		**132**
Skipped question		**0**

As shown in the results in Table 3-3, it is clear that the respondents are almost divided in their opinion about the mechanism of accessing the application. A little more than half the respondent would like to use the application as a web page on the browser, while almost the other half is comfortable to use the new system downloaded as an application. Therefore, there is no clear choice from the results with respect to access and hence it is decided to take other factors into consideration and not make decision based alone on user survey inputs. This survey result could not help make a design decision of either having a browser system or develop an App for the potential system.

Question 4:

Security is an important non-functional requirement of the system design and the customer acceptance to use Wi-Fi network is important as shown in Table 3-4 below.

Table 3-4 Result of Question 4

Restaurant provides you a free Wi-Fi to access the digital menu, would you connect to Wi-Fi always?		
Answer Options	**Response Percent**	**Response Count**
Yes	**47.0%**	**62**
No, I will use my own network always	**9.1%**	**12**
No, I will connect to Wi-Fi some times.	**43.9%**	**58**
	Answered question	**132**
	Skipped question	**0**

The results clearly show that only 9% of respondents strongly believe that they will never connect to the Wi-Fi provided by the restaurant and the rest do not have an objection on the use of Wi-Fi. Therefore, a safe decision is made that using Wi-Fi from restaurants to access the application can be considered as a safety implementation.

Question 5:

This objective question is related to customers' acceptance on using GPS to identify their location to protect against fraud and ensure security in the system design. As in previous questions, the response can only be given by a "Yes", "No" or a "Maybe".

Table 3-5 Result of Question 5

If the digital menu requires you to switch on GPS in your mobile device while you are in restaurant, would you switch on it?		
Answer Options	**Response Percent**	**Response Count**
Yes	**19.7%**	**26**
No	**48.5%**	**64**
Maybe	**31.8%**	**42**
	Answered question	**132**
	Skipped question	**0**

The result as in Table 3-5 shows that respondent is not comfortable switching on the GPS option on their phone. Almost half of the users clearly answer in negative for switching on the GPS on their phones and nearly one-third users are not certain of their choices. This question makes a clear decision that this feature should not be considered for ensuring security in the application. Therefore, not having GPS act as an important input for technology selection during the design phase.

Question 6:

This subjective question in the end is to make the survey open ended. It is important to know and understand the aspects of customers' choices that are not considered in the objective questions. The question was drafted as "What kind of questions would you like to ask the waiter before ordering?" Almost half of the survey respondents answered this question. Few of the interesting responses and our interpretations of the same are listed below:

1. *"If I need a password for the Wi-Fi"*

   We interpreted that the customer thinks of Wi-Fi as a solution for the application and should be able to see the password easily in order to have a good usability for the system.

2. *"Do they can do any special order, for example (burger without mayo and onion, or meal without any sauce)?"*

   Special orders or allergies should be a feature in the system like "comments" where all special requests can be made.

3. *"If I can get a specific menu without meat"*

   Vegetarian and non-vegetarian should be very clearly categorized in the application.

4. *"How big are the portions?"*

   The portion and approximate weight should be mentioned in the application.

5. *"If the ingredients are not clear, I would ask about it"*

   If the application is well designed then one does not need to ask anything. The good design of the application depends on two factors.

   - Presentation of the menu should be clear and not cluttered.
   - Loading time of the menu should be very small.

6. *"None, the menu will hopefully tell me what I need to know"*

   This comment and seven more of the same type suggest that a self-explanatory UI designed with usability principles can automate the order taking process.

7. *"1. Availability of the dish I want to order? 2. How long it will take to get it deliver? 3. If I have any doubts regarding the ingredients in case of veg or non-veg"*

Vegetarian/non-vegetarian should be mentioned clearly in the menu card and the application should keep the customer updated with the order status.

8. *"I might have questions related to the ingredients of certain entries in the menu or to what exactly those funky names mean. Therefore, maybe the interface could provide more details right away, saving my time and the waiter's time"*

   If there are no fancy names in the application, then the requirement is met. Additionally, the option of having photos and videos for a menu item can satisfy these requirements. This response from the user survey revalidates the initial objectives of the project of using photos and videos to the menu for better customer experience.

9. *"Depends on digital description. If it's good enough: nothing and then I'd like to be able to order in advance from the office to save time!"*

   The concept of ordering food from outside the restaurant is out of scope for the application in this phase, but can be considered as a potential feature for future releases.

10. *"Is the digital menu up-to-date?"*

    This makes an important point to keep the application manageable and maintainable by the restaurant staff through admin features and no developers' involvement to update the application data should be needed.

11. *"How long will I have to wait until the food is served? Who will confirm my order of food/drinks? Why do I see the waiter before the order anyhow, if I can order electronically?"*

    The notifications of the order status should keep the customer updated and there should be an option on the system to call the waiter in case of any exceptions or needs.

All the above subjective inputs in the form of questions from the respondents along with the requirements defined with the restaurant management acted as requirements elicitation for developing the application as described in the following sections.

3.2 Functional Requirements

The functional requirements were discussed with the restaurant management keeping in mind the operational issues faced by their team and were generalized to make it work for any other setup without additional efforts or customization. Table 3-6 shows the functional requirements for the menu management system. The requirements listed above can be realized in three broad sections as shown in the logic diagram for the menu management system as in Figure 3-1. The interface layer provides user interfaces for admin, waiter and customer as three clear distinct actors for the system described in detail in the following sections.

Table 3-6 Functional Requirements

S No.	Requirements
R1	Display a food/drink menu in any mobile devices with variant screen size. E.g. tablets, smartphones.
R2	Mobile application should be independent of platforms (iOS/Android/Windows Phone).
R3	The food/drink menu can contain picture and video.
R4	Customer can order one or more food/drink items.
R5	The system must be based on software as a service model (SaaS model); restaurants do not need to install in-house servers.
R6	A 2D barcode (QR code) is assigned in each seat-ID to identify the customers.
R7	Mobile application should be available in multiple languages.
R8	Admin panel for restaurant administrator, where admin can perform create, read, update and delete operation on menu.
R9	Restaurant admin will be able to create new user and able to assign roles such as waiter and admin.
R10	Identify the customers by location before they can retrieve the food/drink menu. This helps to identify whether the customer is accessing the food/drink menu from restaurant or not.
R11	Send 'order acceptance' notification to the customer after a waiter accepts the order.
R12	In case of N number of waiter is listening for orders from the customer, if a waiter W1 accept the order O1, the other waiter W2 should not have any access or control over O1.
R13	Seat switching: Customer may switch/change the table from one place to another place, in this case the system needs to support to transfer the orders from one place to another place. This can be done with the assistance of the waiter.
R14	The waiter can add/update/delete the orders manually in the system-generated orders.
R15	Menu Option: Each food item may have option such as spicy level, lemon, sauce and a comment box for giving additional comments. The waiter for all the manual entries can edit the customer screen later.
R16	Customers can cancel or edit the order before submitting the order to the queue.

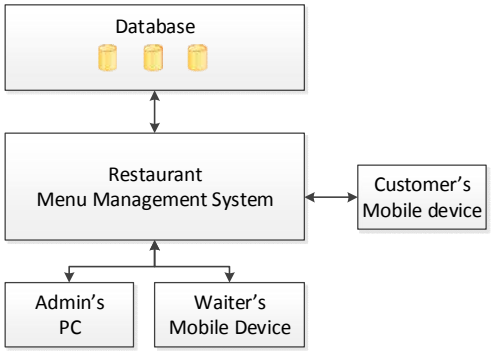


Figure 3-1 Logic View of Menu Management System

3.2.1 Customer

The customer who would use the menu management system is the most important. All different actions for this role and the system are presented in the use case diagram in Figure 3-2. The actor "Customer" should be attached to a seat from where he/she should be able to access the menu management system as per the requirement R1 as mentioned in the functional requirements. Once the system is accessible to the customer, he/she can switch the language of his/her choice on any page as per requirement R7. The customer can access the entire menu and can choose one or more food/drink items of his/her choice without training. The customer can order the selected items but have an option to cancel the order within a specific timeframe if he/she changes his/her mind as per requirements R4 and R16. The customer can see ordered items and total price when the order is placed. Once the order is in queue, the customer gets notifications on the status change of the order such as order accepted, order in process to keep him updated on the status. During this entire process, the customer can call the waiter by using the application.

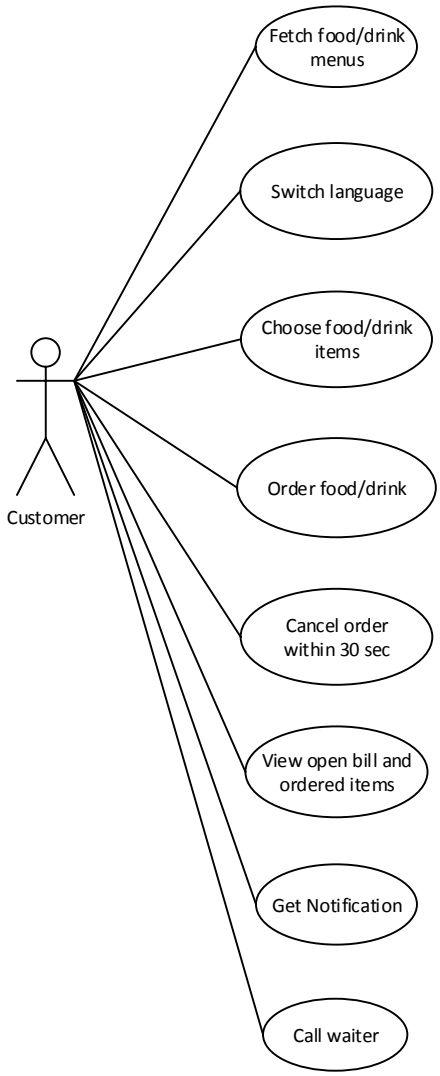


Figure 3-2 Customer Use Case Diagram

3.2.2 Waiter

The other important actor in the system is the waiter and his interactions are depicted in the use case diagram in Figure 3-3. The waiter operates his/her own screen and can conduct the following actions. The waiter can access the queue of open orders as well as the ones assigned to him/her as per the elicited requirements R10 and R12. The waiter can assign orders to himself/herself and then can monitor and update them. For an order assigned to him/her, the waiter can add a food/drink item, update the quantity or delete an item as per the requirement R13. The waiter can also edit the customer seat assignment if there is a request for seat change as in R13. The waiter also edits the order status at any point of the order processing cycle as per R16 to send notifications to the customer as per the requirement R11.

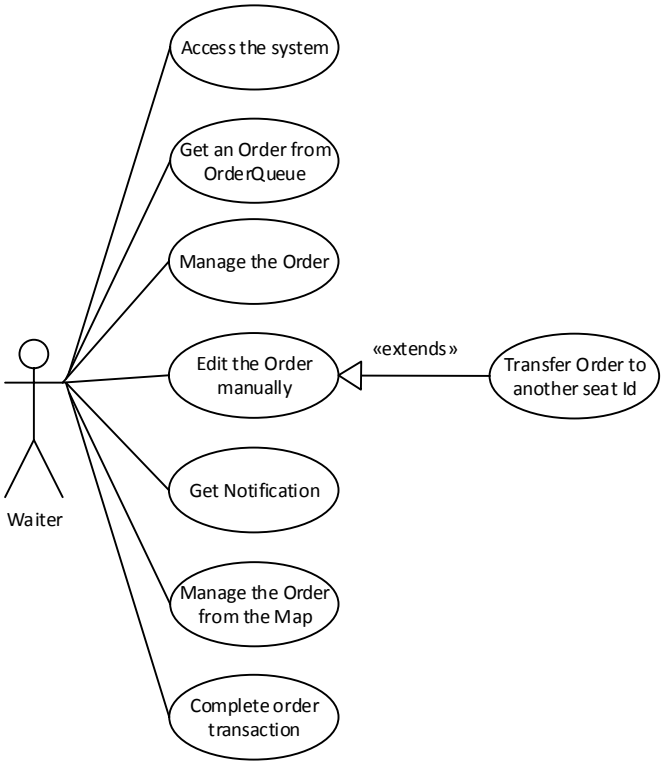


Figure 3-3 Waiter Use Case Diagram

3.2.3 Admin

The role of an admin is important to manage the mobile application maintainability after the deployment. The admin use case in Figure 3-4 displays all the potential actions that the restaurant admin can perform. This is a super user role that can control or edit the roles of all the other actors in the system as per the functional requirement R9. The admin can add, delete or edit the menu items as described in R8. The admin can edit the users and their roles in the system. Order status and the order processing time can be monitored for efficiency and performance of the waiters. The restaurant layout can also be updated depending on weather and special occasions in the system by the admin.

3.3 Non-Functional Requirements

The mobile application is designed for a large number of arbitrary customers using many different devices; who cannot be trained on the application functionalities. Hence non-functional requirements such as usability and adaptability are as important for the application as the functional requirements. In this system, security, availability, adaptability and usability are recognized as the most important non-functional requirements and are discussed in details below.

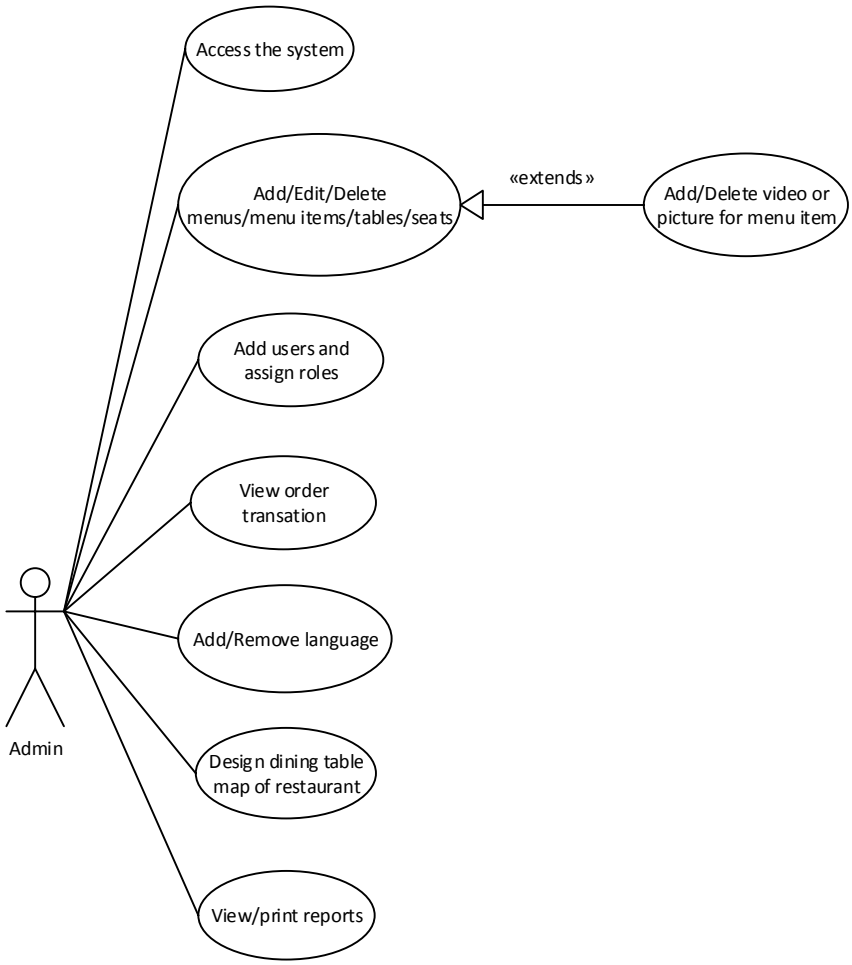


Figure 3-4 Admin Use Case Diagram

3.3.1 Security

Security is an important aspect of the mobile application; as the restaurant management would want the application to be operational only in the premises, also mentioned in the requirements section R10. In case the application is available outside the boundaries of the restaurant, the clients can order food/drinks without being physically present in the restaurant. Therefore, the verification of the customer to be physically present before using the menu management system becomes important to stop any foul use.

3.3.2 Availability

If the system is deployed and used as an alternative to the manual system of order management, the entire restaurant business will be dependent on the menu management system. Therefore, to consider maximum availability will be an important parameter to consider while designing the architecture of the automated solution. Fault tolerance has to be considered in the system design so that the system is operational and running without much downtime. Service maintenance with respect to system uptime should be provided in an event of a failure to ensure availability.

3.3.3 Adaptability

As the customers are not bound with a particular device or operating system to use the mobile application, adaptability becomes an important requirement. The design should work on multiple smartphone devices such as iOS, Android, Windows or Symbian with different screen sizes and resolution. Therefore, the adaptability of the application on multiple devices is a key criterion for design and testing in the implementation phase as also captured in R2 from the user survey results.

3.3.4 Usability

According to the key objectives outlined in previous chapters, the customers cannot be trained on the mobile application usage. The customer's view has to be designed in a way that it is intuitive to use and is self-explanatory across multiple devices. At the same time, the design should be simple and fast as the key measure of success is to reduce the order processing time. A complicated system with many screens or many scrolls up and down would defeat the purpose of the menu management system. The application design should thus comply with the three usability principles [6] of early focus on users and tasks, empirical measurements, and iterative design.

Chapter 4. System Design

The menu management system and architecture has been designed based on the general requirements, user survey, functional requirements and non-functional requirements elicited from the restaurant. The core system is designed based on multitier architecture, and it is only accessible within the restaurant network. The notation R1-R16 denotes the requirements listed in

Table 3-6. The following sections explain the component diagram, architecture and user interface design of the menu management system.

4.1 Component Diagram

The menu management system is designed based on a mixture of multitier architecture, RESTful architecture style [5] and Model View ViewModel (MVVM) pattern [7]. The multitier architecture provides a model to create flexible and reusable components in a mobile application. It segregates the application into several tiers, where developers can add/modify the functionalities on a certain tier instead of modifying an entire application. This allows the functionalities of the system to extend for future development.

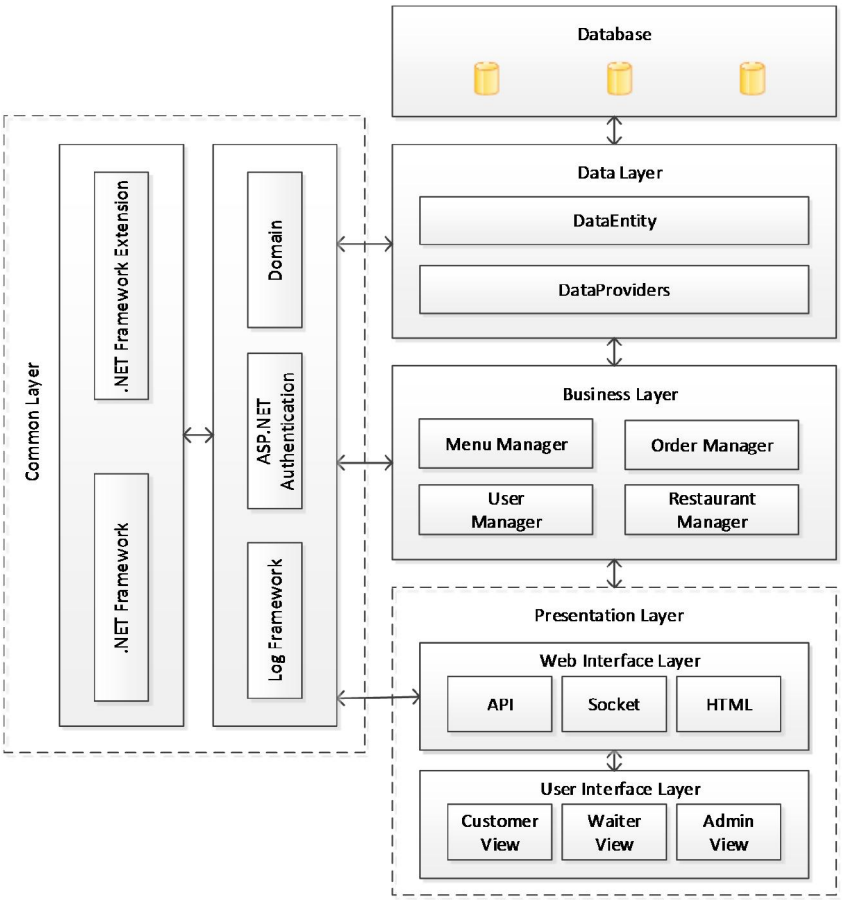


Figure 4-1 Component Diagram

The RESTful architecture style provides core functionalities of the system as web services for different devices and platforms as mentioned in requirements R1, R2 and R5. The MVVM pattern [7] allows to develop the User Interface (UI) with clear separation of UI components and presentation logic. The core system has three main layers: Data Layer, Business Layer and Presentation Layer. The Presentation Layer is further divided into two layers, namely WebInterface and UI Layers. These two layers are loosely coupled and connected with web services. All these layers are cross-connected with Common Layer, as shown in Figure 4-1.

The following sections explain the layers and their functionalities:

4.1.1 Data Layer

The Data Layer is responsible for storing and retrieving data to/from the relational database. Here, the database is mapped to entity classes with the help of Entity Framework (EF). The EF is an object-relational mapper that enables .NET developers to work with relational data using domain-specific objects. It eliminates the need for most of the data-access code that developers usually need to write [8]. The Data Layer has two components, namely DataEntity and DataProviders. The DataEntity maps tables in the database to the classes using the EF mapping files. It avoids writing more SQL queries to fetch and store data from the database. Similarly, DataProviders has the query logic for create, read, update and delete (CRUD) operations. This layer provides a set of interfaces to the Business Layer for accessing its functionalities.

4.1.2 Business Layer

The Business Layer is the central part of the menu management system that encompasses the core business logic. It consists of four main components, namely Menu Manager, Order Manager, User Manager and Restaurant Manager. The Menu Manager is responsible for validating the customers' location (whether the customer is inside the restaurant), and returning the menu for them to satisfy requirements R1, R3, R10 and R15. In addition, this component is used for creating and updating the menu by restaurant admin to satisfy the requirement R8. The Order Manager is responsible for accepting the orders from customers and processing them to fulfill requirements R4, R11, R12, R13 and R14. Similarly, the User Manager is responsible for authenticating the waiter and allocating unique ID for customers' smartphone to satisfy non-functional requirement 'security' and functional requirement R6. In addition, this component allows restaurant admin to create/modify user accounts for waiters, which satisfies the requirement R9. The Restaurant Manager is responsible for configuring the restaurant settings and managing table and seat layout of the restaurant. Functionalities of all these components are accessible to the Web Interface Layer through their interfaces.

4.1.3 WebInterface Layer

The Web Interface Layer provides core functionalities of the menu management system as web services to fulfill the requirement R5. This layer is also responsible for rendering all views (Customer View, Waiter View and Admin View) as web pages and providing real-time communication to the Menu and Waiter View. This layer has three main components, namely API, Socket and HTML. Here, the API component provides web services for all views.

Similarly, the HTML component renders all views, and the Socket component provides real-time communication for Customer and Waiter View to satisfy the requirement R11.

4.1.4 User Interface Layer

The User Interface Layer has three main views, namely Customer View, Waiter View and Admin View. The Web Interface Layer as described in the previous section renders these views. Customer View is designed for smartphones, particularly for iOS and Android platforms as per the requirement identified from Question 1 of user survey. As per the result of Question 3 of user survey, most of the users prefer to use application in web browser instead of installing native application; hence this view is designed as web application. From this view, customers can view food/drinks menus and order them. Similarly, Waiter View is designed for tablets that are more than 11-inch screen size. The waiter can assign orders, change the status of an order and update the order. However, Admin View is designed only for browsers on PC, but not for smartphones and tablets. In Admin View, the restaurant admin can add/remove the waiters, perform Create, Read, Update and Delete (CRUD) operations on menus, and add/remove tables on the restaurant map.

4.1.5 Common Layer

The Common Layer contains the .NET framework components, extensions, logging framework references, ASP.NET Authentication module and Domain model of the menu management system. All these components are used by three other layers as shown in Figure 4-1.

4.2 Communication Architecture

The communication architecture of the menu management system is shown in Figure 4-2. It is designed by considering requirements and results of the user survey as described in Chapter 3. The main goal of this architecture is to provide the menu management system as a service that is accessible only from the restaurant's Wi-Fi network. This application is designed in a way that it cannot be accessed from the public network to satisfy security as the non-functional requirement. Also results of Question 4 in user survey showed that most of the users prefer to connect to the restaurant's Wi-Fi.

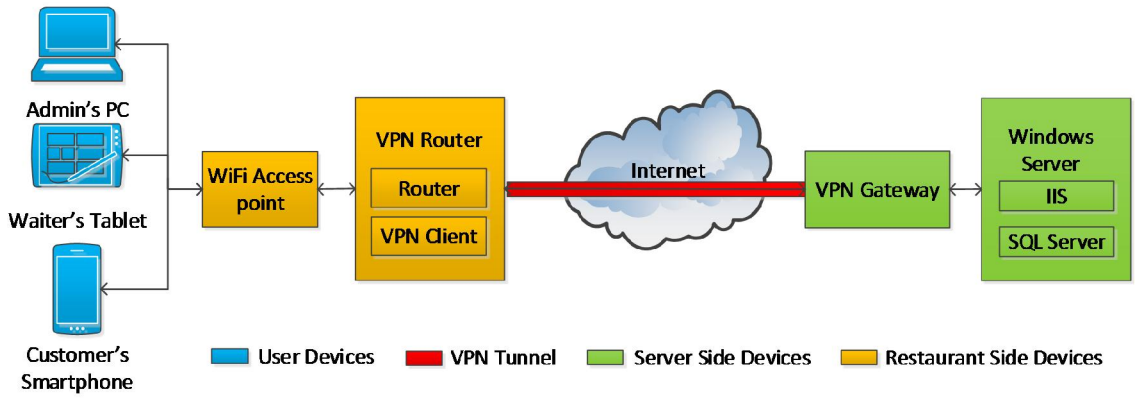


Figure 4-2 Communication Architecture of Menu Management System

According to the architecture design, the application is hosted in a Windows Server, which is located in private network premises. This server has the Internet Information Services (IIS) web server, which hosts the menu management system, and the SQL Server, which hosts the database for the system. The application hosted in the Windows Server can only be accessed through the VPN Gateway. On the restaurant side, VPN Router is installed and connected to VPN Gateway, which in turn connects to the Windows Server. This configuration establishes a VPN tunnel (shown in red color) from the restaurant to the hosting server. The VPN router is connected to a Wi-Fi access point that gives application access to different devices. The devices might be a PC for accessing Admin View, a tablet for accessing Waiter View and a smartphone for accessing Customer View. In restaurant, QR code is placed in front of each seat where a customer can scan it to access Customer View. This also helps the system to accurately identify the location from where a customer places orders.

4.3 Entity Relationship Diagram

The Figure 4-3 shows the Entity Relationship (ER) diagram of the menu management system. The ER diagram has two main categories of tables, namely master tables and operational tables. The master tables are: Restaurant, RestaurantType, Restaurant_RestaurantType, Address, Country, Table, Seat, SeatAllocationStatus, OrderStatus, Menu, MenuCategory, MenuItem, and MenuItemOptions.

As per the requirement R7, the system should support different languages. In order to satisfy this requirement, master entities Menu, MenuCategory, MenuItem and MenuItemOption are designed to have multiple entries in MenuML, MenuCategoryML, MenuItemML, and MenuItemOptionML entities. Hence, these master entities can store the name and description in one or more languages. Rest of the tables, such as Order, OrderQueue, OrderItem, OrderItemOption, aspnet_Users, Waiter, WaiterNotification, Statistics, EndUser, EndUserNotification and SeatAllocation, are operational tables.

Main functionalities of the group of related tables are explained below:

- The restaurant properties and settings are stored in Restaurant, RestaurantType, Restaurant_RestaurantType, Address and Country tables.
- The Table-Seat layout and its position are stored in Table and Seat tables. To satisfy requirements R6 and R10, these tables are related to SeatAllocation table which allows to identify the locations (table and seat number from where an order is sent) of customers.
- Seat allocation for each customer is stored in SeatAllocation, SeatAllocationStatus and EndUser tables.
- In order to satisfy order processing requirements R4, R11, R12, R13, and R14, the order processing data is stored in Order, OrderItem, OrderItemOption and OrderQueue. Each order is related to a seat and seat allocation instance.

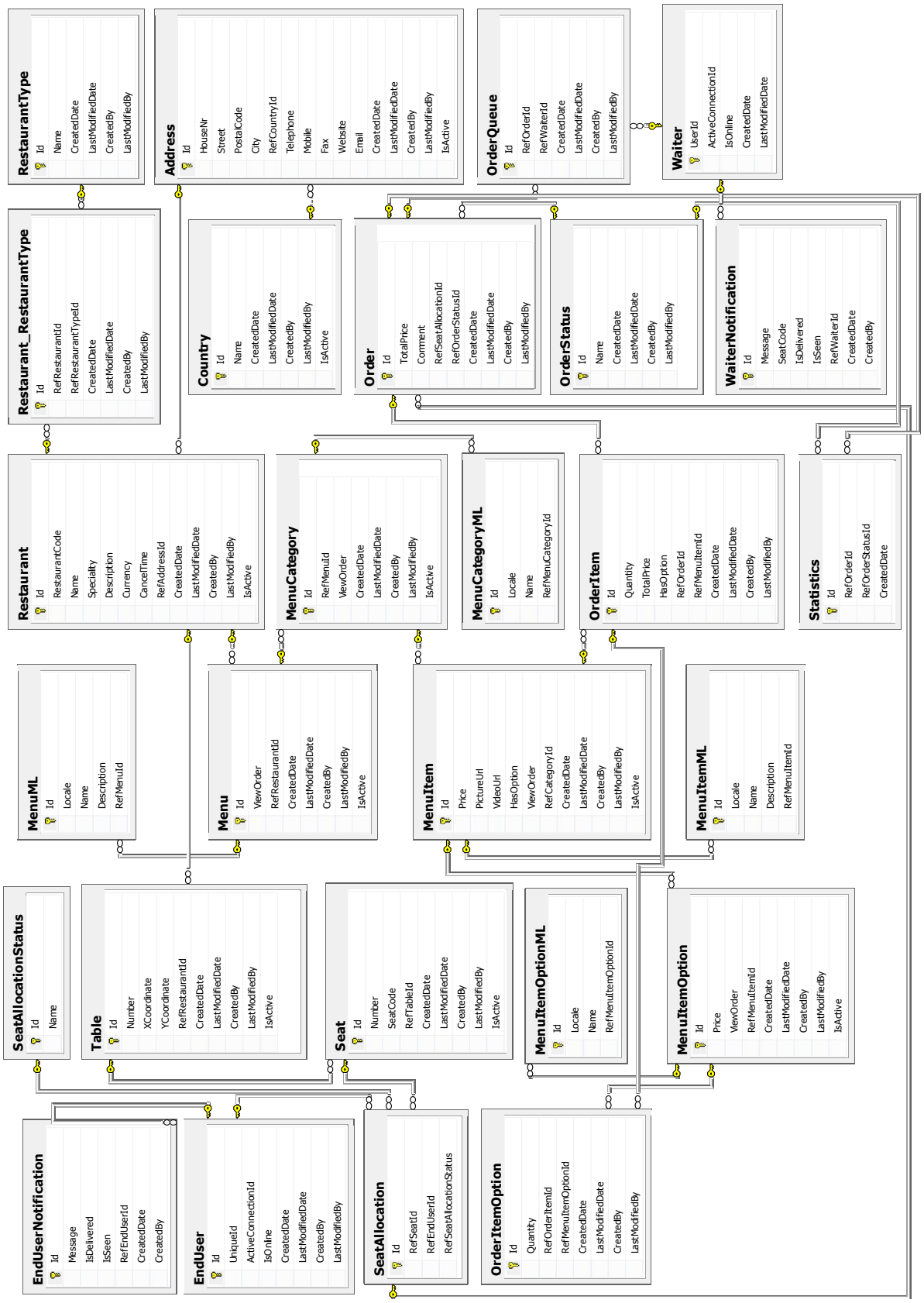


Figure 4-3 Entity Relationship Diagram

- Related to the requirement R11, the notification messages for the waiter is stored in WaiterNotification table. Similarly, notification messages for the customer are stored in EndUserNotification table. This prevents the loss of notification messages for both waiter and customers, when network connection gets restarted.
- All order statistics information for the system analysis is stored in Statistics table. This information is used for evaluation of the system.

4.4 User Interface Diagram

The menu management system has three views: Customer View, Waiter View and Admin View. Customers, waiters and restaurant admin, use these views respectively. Since only Customer View and Waiter View are directly involved in the order taking process, considerable effort has been put on the interface design. Due to time constraints and less importance of Admin View usability, its interface has been designed along with the development phase. After all the functional requirements are gathered and use case diagrams for all actors are defined, initial prototypes of Customer View are built as shown in Figure 4-4.

But these prototypes were complicated in terms of usability and functionality. In this prototype, it is hard to identify the navigation and behavior of UI. After analyzing some other mobile applications and discussing between the project team members, the prototypes have been significantly improved to satisfy the non-functional requirement usability. Some feedback on UI is gathered from Dr. Elaine Huang, head of People and Computing Lab research group, department of Informatics, University of Zurich.

Figure 4-4 Initial Prototype

The following suggestions were given:

1. To shorten the depths of accessing menu items. The menu structure itself consists of 3-level hierarchy and browsing the menu items required 3 steps as 'menu->categories->menu items'. As a result, 3-level hierarchy is reduced to 2-level hierarchy by displaying menu and its categories on the same page.
2. To change the term 'Cart' to another name, since people may think that they need to pay in order to place an order. We eventually changed this term to 'Tray', which is more suitable name for the step before people place an order.
3. To display menu item details (description, photo and video) on the same page to avoid one more depth of hierarchy level.
4. To add some interaction features in the form of chat which support customer-waiter communication in real time. Since features such as comment box for additional requests, calling waiter and order status notification were present, and taking into account the time constraints, we left this chat-like feature for future improvement.

By considering these key design recommendations received from Dr. Huang and based on an initial prototype, the final prototypes are designed as shown in Figure 4-5.

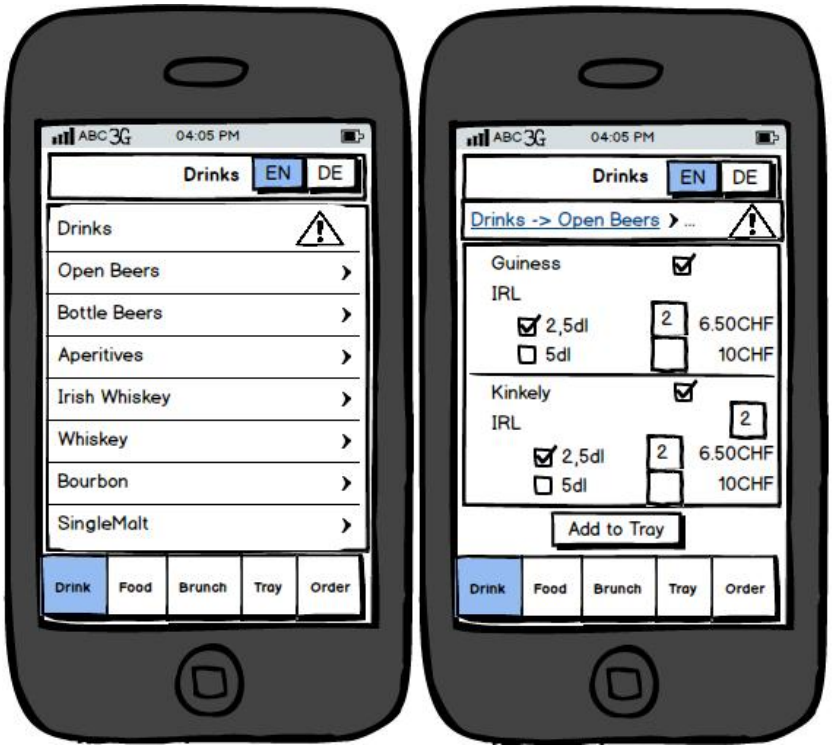


Figure 4-5 Final Prototype

In order to satisfy the functional requirement R15, the final prototype shows the menu options clearly for customers. In addition, a comment box is provided in order page where they can give additional comments per order, such as sausage, lemon and spicy level. For satisfying the requirement R16, Order page is provided with cancel option for selected menu items before submitting it.

User Interface Diagram shown in Figure 4-6 demonstrates the page flow chart for Customer View and Waiter View. In Customer View, Home page represents the default page, which is shown when the application is accessed. From Home page, customers navigate to Drink/Food/Brunch pages, where appropriate menu categories are listed. By clicking on a category, Menu Items page is displayed with its item details (menu item name, description, options, price, photo and video). Customers can select menu items on this page and by clicking 'Add to Tray' button they go to Tray page. After reviewing all menu items in the tray, customers can place an order by clicking the 'Order' button. When a waiter changes the status of an order, notifications appear on Notifications page.

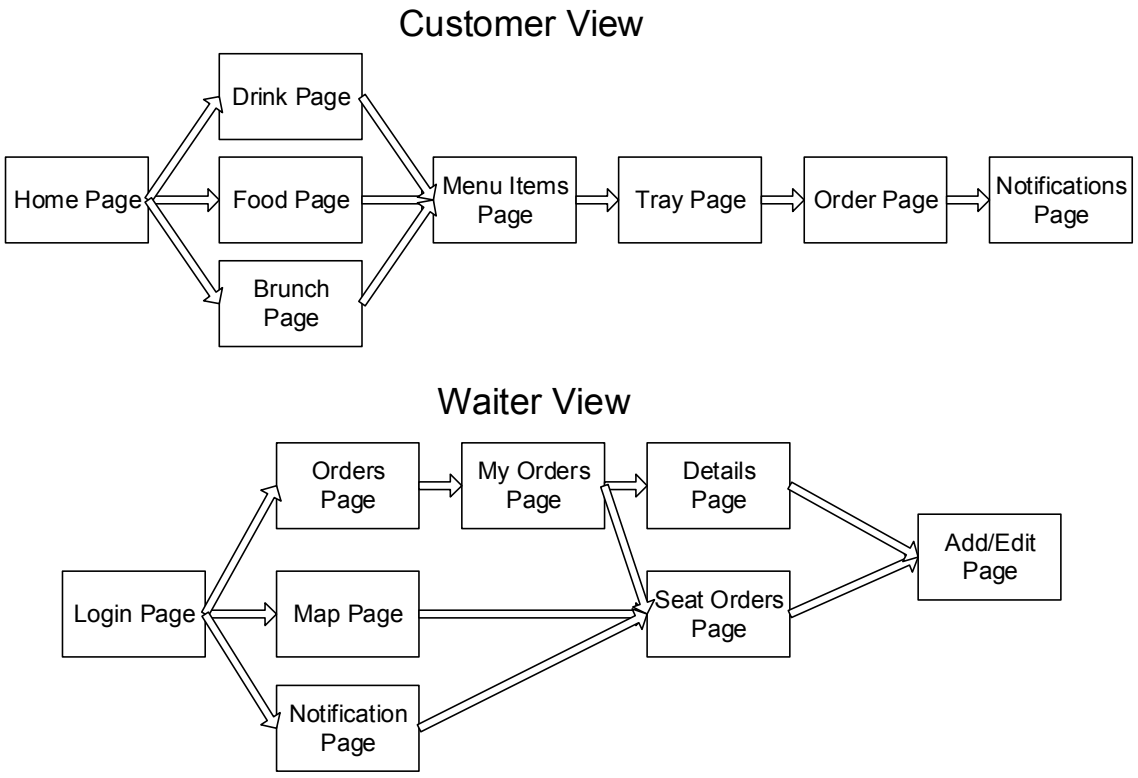


Figure 4-6 Page Flow of Menu and Waiter View

In Waiter View, Login page is default, which asks for username and password. After a waiter logging into the system, he/she can see the queue of open orders on Orders page. As soon as an open order is assigned, the system switches to My Orders page. On this page, the waiter can see all orders assigned to him/her. Details page accessed from My Orders page shows all details of the order and the waiter can change status of the order. My Orders page also provides access to Seat Orders page, which shows all orders related to that seat and other functionalities such as change the seat for orders and close all orders. Add and Edit pages allow to add new items to the order and edit details of given order. These pages can be accessed from both Details and Seat Orders pages. Map page presents the visual table and seat layout of the restaurant. By clicking on a seat, the waiter can navigate to Seat Orders page. When customers invoke call-waiter function, notifications are highlighted on Notifications page.

4.5 Workflow of the System

Workflow of the menu management system describes the business logic by using UML activity diagram and sequence diagram. The activity diagram explains the steps performed in each action. The sequence diagram describes the interaction between system and three different users (Customer/Waiter/Admin).

4.5.1 Activity Diagram

This section illustrates main activities which includes accessing the menu management system, ordering food and drinks, order processing and administration of menu management system.

4.5.1.1 Accessing the Menu Management System

The activity diagram for accessing menu management system is shown in Figure 4-7. In order to use the menu management system, customers should connect to the restaurant's wireless network using a password written on the instruction card. To access the application, customers should have QR code reader application on their smartphones. In case this application is not present, it can be downloaded and installed from the platform's applications store. After customers install the QR code reader, the QR code placed on the table should be scanned. The URL, which appears after scanning, should be clicked to display the menu.

4.5.1.2 Ordering the Food and Drinks

The activity diagram for ordering food and drinks is shown in Figure 4-7 and Figure 4-8. Customers can see menu items by clicking on the menu category. After customers select the menu item, they have to click on 'Add to Tray' button to add menu items to the tray. If they want to select more items from other categories, the selection activity is repeated. Once menu items are on the tray, customers can review them before they place an order by pressing 'Order' button. After clicking on 'Order' button, customers can cancel the order within configured time period. After the time period expires, the order will be sent to Waiter View.

4.5.1.3 Order Processing and Delivery

The order processing and delivery activity diagram is shown in Figure 4-9. The waiter should type login credentials in order to logging into the system. All new orders from the queue appear on Order page. The waiter has to click on 'Assign' button to assign it to himself/herself, so that he/she could manage the order further. Once the order is assigned, the system switches to My Orders page automatically, where the waiter can see the list of assigned orders. From this page, the waiter can go to either Details page or Seat Orders page. The only difference between these two pages are that Details page shows details of one order and Seat Orders page shows details of all orders for that seat. On both pages, the waiter can change the status of the order, and a notification will be sent to Customer View about the status. The Add/Edit page can be accessed from both Details and Seat Orders pages. The waiter can add new menu items or edit the existing menu items there.

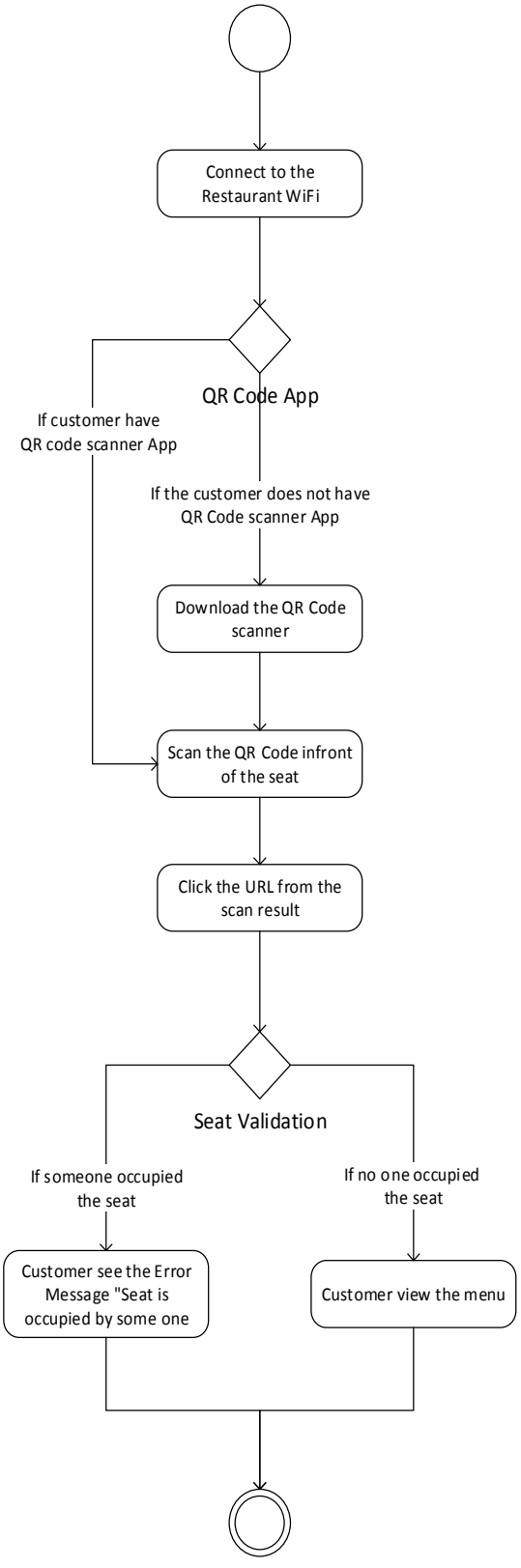


Figure 4-7 Activity Diagram - Accessing the Menu

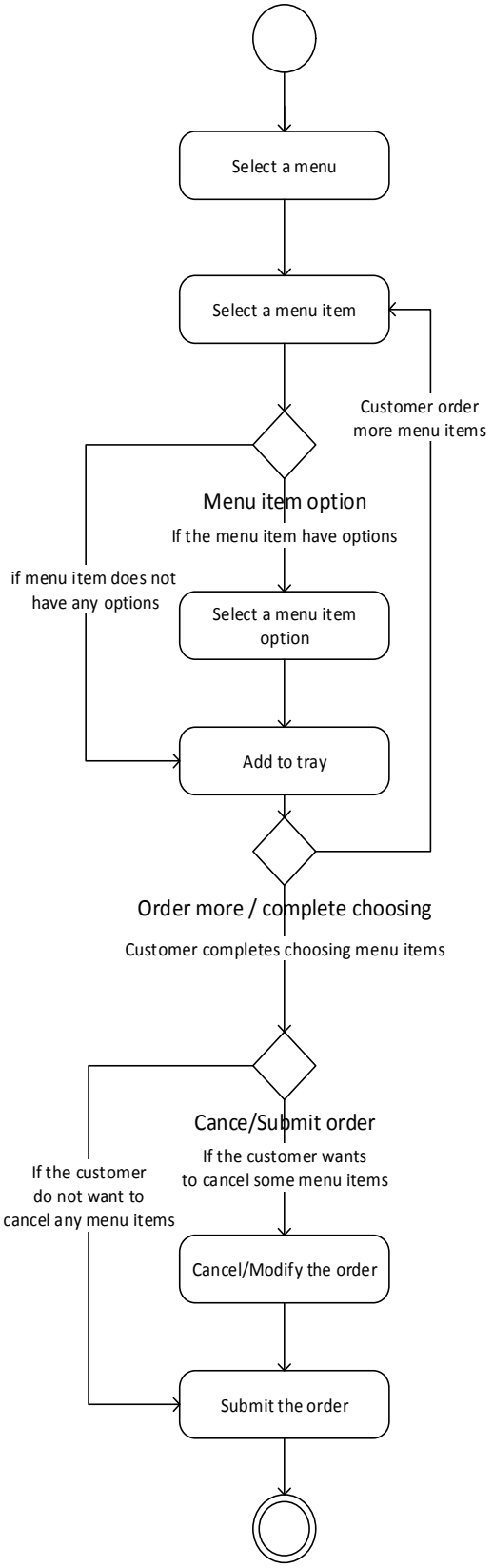


Figure 4-8 Activity Diagram - Ordering Food/Drinks

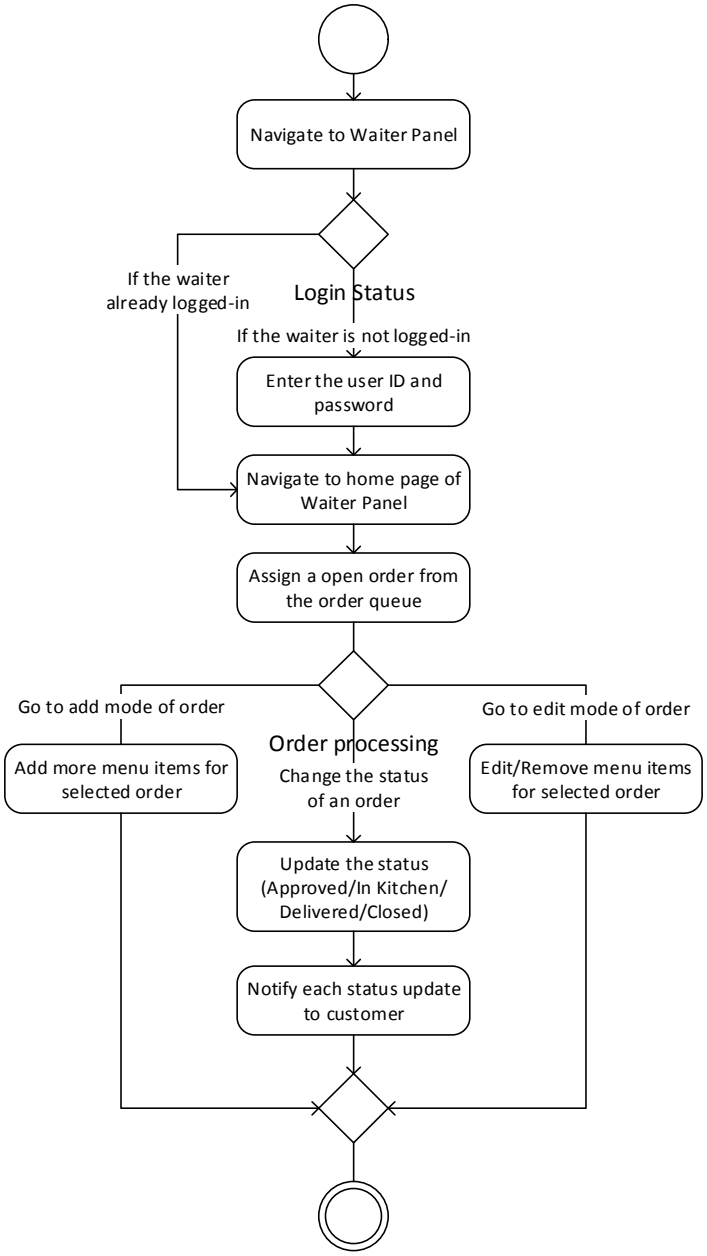


Figure 4-9 Activity Diagram - Order Processing and Delivery

4.5.1.4 Administration of the System

Figure 4-10 depicts the administration of the system activity diagram. Here, before using Admin View, the restaurant admin should enter his/her password to logging into the system. The navigation menu on top of Admin View contains main sections such as Menu, Menu Items, Table, Restaurant, Statistics, User Management and QR code. Under the Menu section, the restaurant admin can add/edit/delete menus and menu categories. Likewise, the Menu Items section is used to add/edit/delete menu items and options. The Table section is responsible for adding/editing/deleting tables and seats on the table, and for changing the layout of tables map. Adding/Removing waiters and assigning roles for

them are done under the User Management section. The QR codes, which are placed on each seat, can be generated seamlessly under the QR code section. The restaurant admin can see all order transactions on the Statistics section.

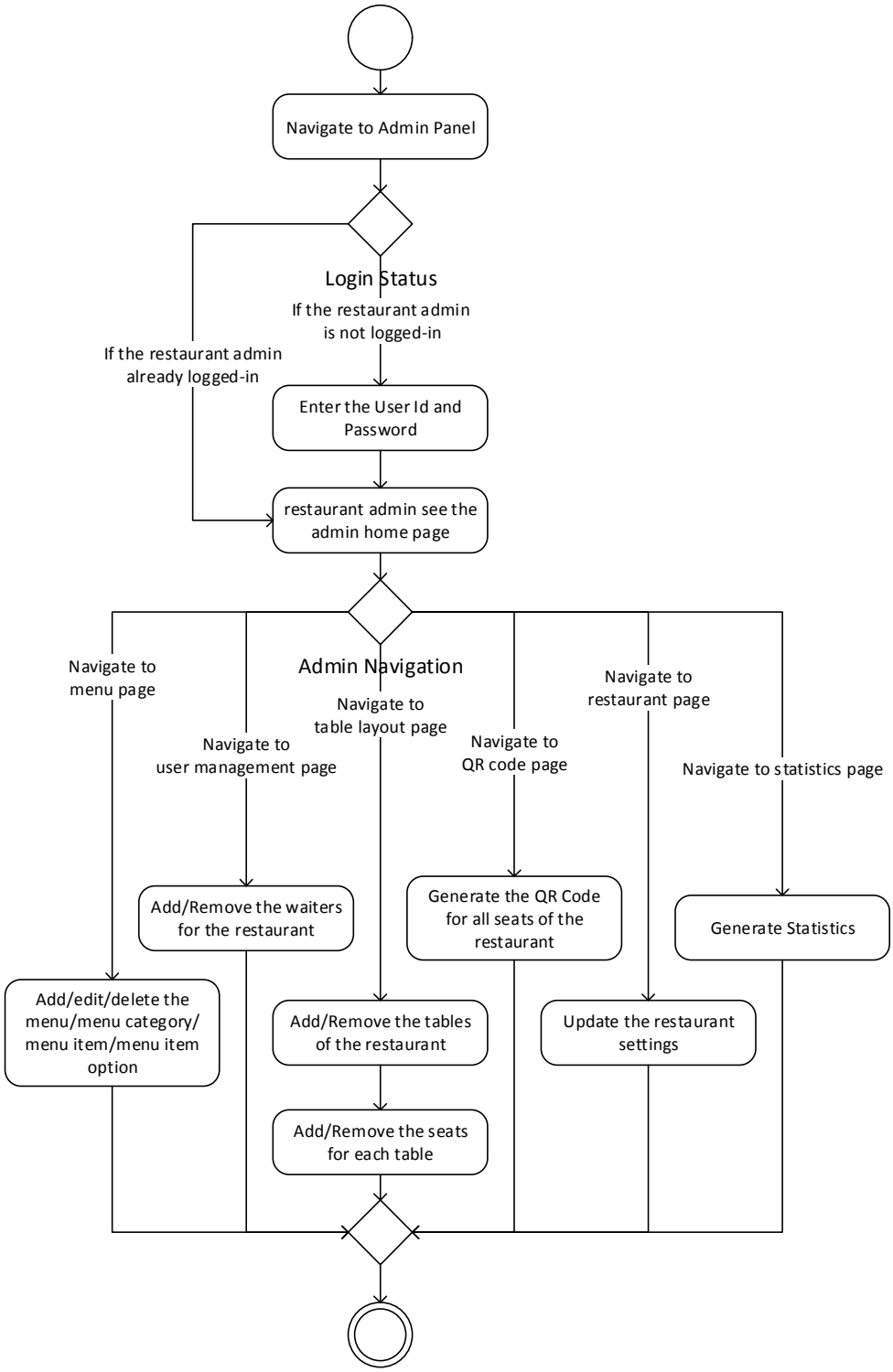


Figure 4-10 Activity Diagram - Administration of System

4.5.2 Sequence Diagram

A sequence diagram describes how interactions between an actor and a system occur in time sequences. This section illustrates communication messages being passed between actors and different components of the menu management system for each actor's action, which is defined in use case diagrams from Chapter 3. The menu management system interacts with three types of actors, namely customer, waiter and restaurant admin, consists of two components:

Graphical User Interface (GUI) - allows actors to interact with the menu management system and implements client-side logic.

Service - handles different HTTP requests by parsing its data and matching it to the correct logic implementation function. It is also responsible to store, manage and process data in the database.

The sequence diagram of Customer View is shown in Annex A.1, it explains the communication sequences between customer, GUI and Service component. The waiters view sequence diagram is shown in Annex A.2, which describes the communication sequences between waiter, GUI and Service component. The Annex A.3 depicts the sequence diagram of admin, it explains the communication sequences between admin, GUI and Service component. In general, all actors interact with GUI to perform specific actions. Depending on the type of the actor's activity, the HTTP GET/POST/PUT/DELETE request is sent with its parameters/data to the Service component, which in turn extracts parameters and matches it to the corresponding logic implementation function. The Service component is also responsible for sending/receiving data to/from the database and for processing it.

4.6 Selection of Technology

Pugh Matrix makes the selection criterion on various options of deciding the right technology. It is invented by Stuart Pugh, the Pugh Matrix [9] is a quantitative technique that is used to rank the multidimensional options of an option set. It is frequently used in engineering for making design decisions. A weighted decision matrix consists of a set of criteria options that can be weighed in order of importance. These are scored and summed to obtain a total score, which can be ranked. The resultant score better reflects on important criteria for decision maker. For the more important criteria, higher weight is given. Each of the potential option is scored and also multiplied by the weights given to each of the criteria in order to produce a result. This is hence an objective and logical way to come to important design decisions.

In the order management system, the usage of Pugh Matrix is well suited as there are multiple technologies that can be considered and evaluated on many parameters ranked or weighed based on their importance. The Pugh Matrix has inputs from the user survey, restaurant management, usability experts from UZH and project team members from different technology domains.

4.6.1 Identification of Customer Location

When customers send the food/drinks order, it is necessary for waiters to identify the location of customer. To fulfill this requirement, two technologies were considered for implementation, namely NFC (Near Field Communication) [10] and QR code [11]. NFC provides a short-range wireless interaction in consumer electronics, mobile devices and PCs. A NFC chip with unique ID of a seat can be placed on each seat in the restaurant, and customer can use a NFC enabled smartphone to get the menu. Since very few smartphones have NFC feature, this technology is far from implementation and hence, not considered for solutions in this project. Instead of NFC, QR code of a URL is placed in front of each seat in the restaurant that represents the unique ID of the seat. So that customers can send a request that is obtained by scanning of the QR code. As understood from user survey in Section 3.1.2, most users have iOS and Android smartphones, which are capable of scanning QR code. Thus QR code has been selected for identifying the location of customer in a restaurant.

4.6.2 System Selection

The system (Browser vs. Application) is selected based on the Pugh Matrix. There are only two options that can be considered for accessing the system from a mobile phone. The various criteria are listed based on their weight against different choices. Criterion of the user survey is given the maximum ranking, as there was a clear preference from respondents.

Table 4-1 Browser vs. App

Criteria	Rating	Browser (1)	App(2)	Notes
User feedback-survey	5	+	-	We took the feedback and gave heavy weightage to what the user feels.
Technical implementation	4	+	-	Development cycle time for browser is easier.
Usability	4	+	-	It is easier for users to open a web page then downloading an app.
Testing	3	+	-	Testing on web browser is less time consuming then the App.
Accessibility	3	+	-	CR can get connected to database directly
Security	4	-	+	Applications are more secure then the browser application.
Future enhancements	2	-	+	Applications are easy to enhance for further releases.
Usefullness	4	-	+	For multiple uses App will be much more usable.
		19	**10**	

The other criteria such as implementation, usability, testing, accessibility, security and usefulness are also carefully incorporated after valid inputs from the project team, restaurant management and usability experts. Although the application is ranked higher on security, future enhancement and usefulness, the overall score is higher for the Browser option. As per the results of the Pugh Matrix, Browser option is selected as a choice of implementation for the new system.

4.6.3 Security Decision

Security is one of the major non-functional requirements for the application. To decide between the logical possible options of using Wi-Fi vs. Encrypting the QR code vs. Geo-location, the criteria such as user survey and technical implementation are given the maximum ranking. Without the acceptance of customers to use security features, any state-of-the-art technology is of no value to the new system. At the same time, the security cannot be implemented without a valid process of technical implementation.

Table 4-2 Wi-Fi vs. Encryption vs. Geo-location

Criteria	Rating	Wi-Fi (1)	Encryption (2)	Geolocation (3)	Notes
User feedback-survey	5	+	na	-	This remains one of the most imp criterions to represent end user voice.
Technical implementation	5	+	-	+	Implementation complexity and timeline.
Security	4	half	+	-	What is the most secure implementation?
Accuracy	4	half	+	-	How accurate are the results?
Accessibility	3	+	-	+	How easy it is to access the system?
Usability	3	s	+	s	Ease of use is same in both the cases.
Usefulness	4	s	s	-	After the implementation, what mechanism will keep the application most secure?
		24	**16.5**	**11**	

Amongst the three options, Wi-Fi scored the highest as the respondents of the user survey accepted to use it and also the technical implementation is possible with the added benefit of usefulness.

4.6.4 Web Technology Stack

.NET vs. Java vs. LAMP were shortlisted as the web technology stack for implementing the application as these are top three technologies available in the market today for mobile software development that the team is aware of it. A different set of criteria such as ease

of implementation, previous expertise, support for latest web technologies and scalability is given the highest weight. Java and .NET are very close choices. But based on the Pugh Matrix, .NET scored the highest as the language of implementation. The matrix is of great help in this case, since the decision is not very obvious and cannot be taken by intuition.

Table 4-3 LAMP vs. Java Ent vs. .NET

Criteria	Rating	LAMP (1)	JAVA ENT. (2)	.NET (3)	Notes
Performance	4	+	S	S	Which technology gives the best performance?
Ease of Implementation	5	-	S	S	How easy is the development process?
Previous Expertise	5	HALF	HALF	+	What areas do we have previous experience from team members?
Support for latest web technologies	5	S	S	+	Which technology is better for future generations?
Licenses	2	+	HALF	-	Where is it easy to manage and purchase licenses?
Scalability	5	-	s	s	How scalable can the application be after implementation?
Strict Typing	4	-	S	S	
Corporate Support	4	-	HALF	+	Support for the entire technology stack through the company.
Support from communities	5	+	S	S	Active communities to support learning and questions.
Modularity	4	-	+	+	Development in modules for no future changes.
Integration	5	-	+	HALF	Integration with other technologies in the future.
		16	**40**	**43.5**	

4.6.5 UI Technology

The project team decided to use a package solution for user interface design and not to work from scratch. This can save time and the project team can focus more on the core functionalities of the system. JQuery Mobile and Kendo UI are two package candidates the project team is familiar with for UI and hence they are considered for the Pugh Matrix. Ease of implementation and cross browser support is the two most highly ranked criteria. As per the Matrix, JQuery Mobile is a clear winner and is used for the UI implementation of the application.

Table 4-4 JQuery Mobile vs. Kendo UI

Criteria	Rating	JQuery mobile (1)	Kendo UI (1)	Notes
Ease of Implementation	5	+	-	How easy is to implement the query framework for UI development?
Cross browser support	5	+	-	Which is better for implementation across browser?
Licenses	2	+	-	Where is license management easier?
Corporate Support	4	-	+	Support for the entire technology stack through the company.
Support from communities	5	+	-	Active communities to support learning and questions.
Interoperability with other frameworks	5	+	-	Development in modules for no future changes.
Available templates	3	-	+	Integration with other technologies in the future.
		26	**7**	

Chapter 5. Study Design

Study design is one of the core items of this master project, which is used to study the effects of introduction of the menu management system in the restaurant. As described in Chapter 2, the experiment is conducted to test whether the system has met the objectives of the desired solution. This experiment is designed after the requirements specification phase and the system design phase by writing a study design that is available in the Annex B. In this chapter we outline the study design and briefly explain the procedure.

The objectives of the application are addressed in two outcomes, which are:

- The effects of introduction of the mobile application on the order processing time in the restaurant.
- The effects of introduction the mobile application on the customer satisfaction.

To address the first outcome, order processing time variable is defined as the timespan that is the difference between the times when a customer places the order and the time the order is delivered to the customer. To address second outcome, average point variable is an average value of customers' feedback in the range of from 1 to 5. Formally, to conduct the experiment, hypothesis is defined for addressing the two variables to test the application.

Null Hypothesis: It is defined, as "The order processing time average remains unchanged on introduction of the new application". Average time of only the drinks order between order submission and order delivery remains unchanged from 7 minutes with or without the menu management system.

Alternate Hypothesis: "The average order processing time changes significantly on the introduction of the new application." The only drinks average order time changes significantly from 7 minutes with the introduction of the menu management system.

The experiment also tests whether the customer satisfaction remains the same or increases with the introduction of the mobile application using the evaluation form and calculating the point average.

As part of the study design, the inclusion and exclusion criteria are clearly defined to make the experiment results more objective. The main criteria are among all the orders placed in the restaurant, only the drink orders are considered. This eliminates the effect of preparation time in the kitchen on the order processing time. The bar area is also considered as out of scope for the experiment; since the waiter is always available in this area and the wait time for these orders are not a valid parameter. Only Waiter and Customer View are considered for the study, and Admin View has been excluded, since it is not related to overall objectives of the application.

The procedure of the experiment is detailed in the study design and the functionalities of each panel are described step by step for clear understanding. The timings for conducting the experiment are outlined, and the milestones are identified. The study design assumes

that before the experiment is conducted the development and testing of the application is complete. The detailed procedure starts three days before the experiment begins and they are: waiter training, infrastructure setup and the QR code and the instruction cards placement. This helps to simulate the environment for the waiters to play around with the system before the real experiment begins. The next milestone is the stage of the actual experiment that is scheduled to for four weeks with an extension potential of one week in case the 30 expected results not achieved.

There is also a section in the study design detailing the adverse events that can happen during the experiment. The potential breakdown in services/ hardware or human behaviors are carefully thought out and; the details of a follow up action in case of eventuality is also described. Overall the study design describes the experiment in every detail so that the analysis for testing the hypothesis is completely objective.

Chapter 6. Implementation and Deployment

The order management system is implemented based on the system design described in Chapter 4. This system has two main subsystems, server and client. The client subsystem is also known as Graphical User Interface (GUI). The server provides its functionalities through web services, which are developed based on the Representational State Transfer (RESTful) architecture style [5]. In addition to the web services, real-time communication on top of HTTP protocol is implemented to notify waiters and customers in real time. The client subsystem is a web-based application, which is designed for most commonly used smartphone browsers such as Chrome and Safari. The following sections discuss on the server subsystem and the GUI in detail.

6.1 Server Subsystem

The server subsystem provides the application functionalities as web services, renders the GUI as web application and provides real-time communication to the GUI. The subsystem is divided into three layers, data layer, Business Layer and Web Interface Layer as explained in Section 4.1. This subsystem is developed using Microsoft web technologies and the backend is implemented using Microsoft SQL Server. The following section briefly explains technologies used, responsibilities and core functionalities of the server subsystem.

6.1.1 Technologies Used

The server subsystem is built based on the following technologies:

1. In API component of WebInterface Layer, the web services are implemented in RESTful architecture style using ASP.NET WebAPI [12]. The ASP.NET WebAPI is a framework, which makes HTTP based services easy to build.
2. Three different views for customers, waiters and restaurant admin in User Interface Layer are rendered by HTML component of Web Interface Layer, which uses the razor view engine [13] shipped with ASP.NET MVC framework [14].
3. To provide real-time communication (i.e. to send a notification from server), the SignalR [15] library is used for Socket component of the WebInterface Layer.
4. Apache log4net library [16] is used to store application logs in the text file and database. This library is used in all the layers of server subsystem.
5. Entity Framework (EF) [8] is an open-source object-relational mapping (OR) framework for .NET Framework. It maps tables from SQL Server database to the classes. It avoids writing huge numbers of SQL queries while handling store and retrieve operations from the database.
6. The backend system is implemented using Microsoft SQL Server [17]. It stores all persistent data of the system.

6.1.2 Responsibilities and Core Functionalities

The server subsystem has several functionalities that are explained in this section. A seat code is printed in the form of 2D barcode (QR code) for each seat in the restaurant. In this subsystem, a customer finds a free seat in the restaurant to scan QR code and then sends a HTTP request to open the menu. The seat is valid if and only if someone else does not

occupy it. The server subsystem validates each HTTP request and then returns the menu. The menu can be viewed in both English and German languages with the help of the language chooser component. When the customer submits the first order from the seat S1, the seat S1 will be allocated to that customer. Other customers cannot use the seat S1. Once the food/drinks order is submitted to the order queue, the waiter receives a notification message. Afterwards, the waiter can assign the order to himself/herself and then process (change the status or modify the order if needed) the order. The customer will be notified for each order status update. Once the customers had the food/drinks and want to pay money, they can call the waiter by sending the "call waiter" notification. The restaurant admin can perform Create, Read, Update and Delete (CRUD) operations for the entities Menu, MenuCategory, MenuItem, and MenuItemOption. In addition, the restaurant admin can create/modify the table and seat layout of the restaurant. Once the restaurant has one or more tables, the restaurant admin can generate the QR code from Admin View.

6.1.3 Class Diagram

This section describes the class diagram of the menu management system for the three layers as mentioned in previous section. Firstly, classes in the Web Interface Layer will be discussed. The Web Interface Layer has three main components as specified in Section 4.1.

The API component has several controller classes as follows:

- *CloseOrderController:* This class is responsible for closing all orders for the given seat code.
- *EndUserController:* This class assigns and validates the unique ID for each device of the customers.
- *MenuController:* This is one of the important class which performs CRUD operations on Menu and MenuCategory objects. Only the restaurant admin is allowed to perform the CRUD operations. The read operation can be performed by customers from a valid seat, by waiters and by the restaurant admin.
- *MenuItemController:* Here, the restaurant admin can perform CRUD operations on MenuItem and MenuItemOption objects. However, waiters and customers can perform read operations on those objects.
- *MyOrderController:* This class is used only by the waiters for fetching the assigned orders and for updating the status of the order.
- *NotificationController:* This class is responsible for storing/receiving all notifications for both waiters and customers based on their unique ID.
- *OpenOrdersController:* This class helps waiters to receive open orders (orders which are recently submitted, but not assigned by any waiters) from the order queue and to assign the open orders to them.
- *OrderController:* This class is used for customers to submit/receive the orders and for waiters to receive/update the orders.
- *OrderStatisticsController:* This class generates the order statistics for the restaurant. This service is only accessible by the restaurant admin. These statistics can be used for the system analysis explained in Chapter 7.

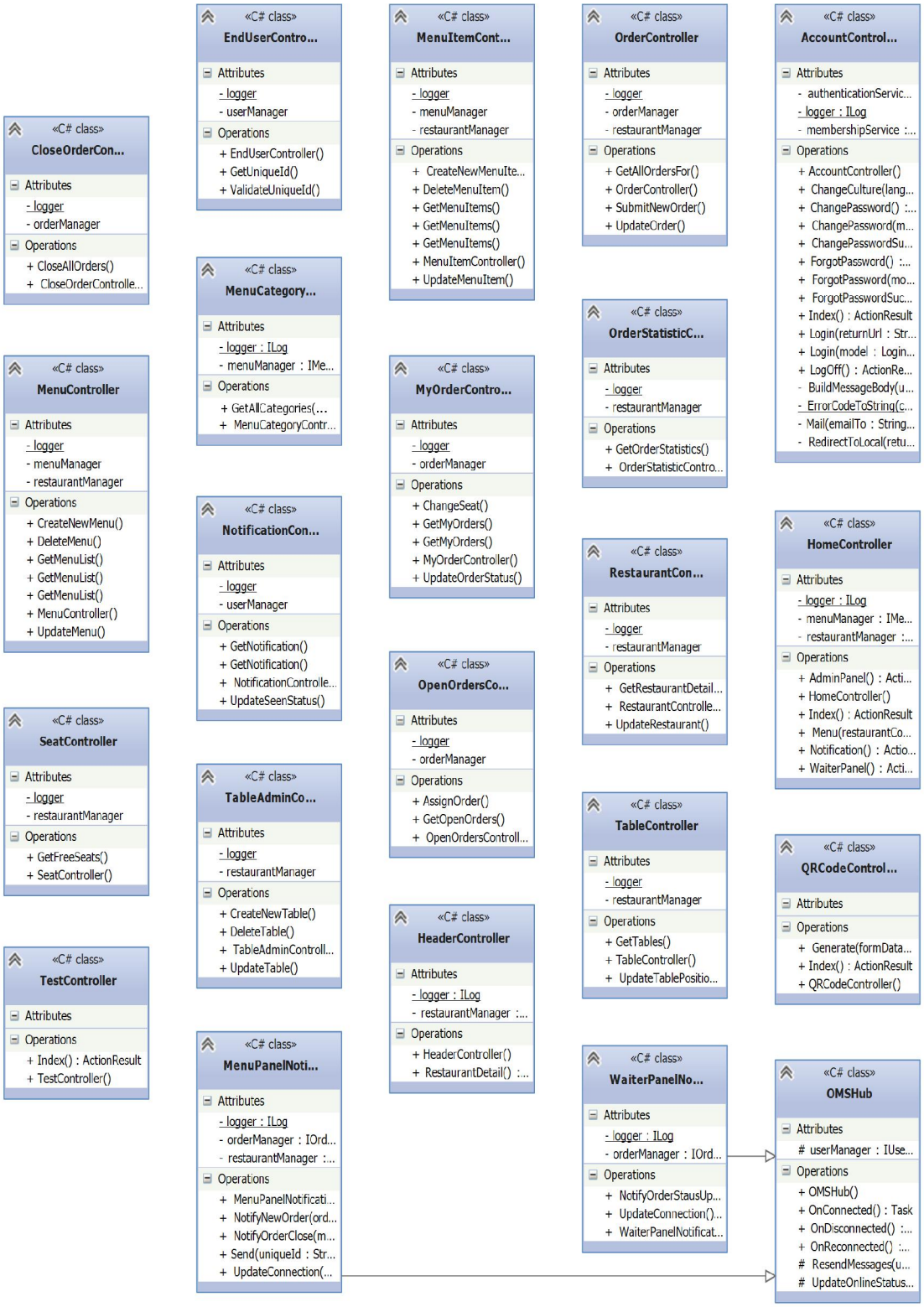


Figure 6-1 OMS.WebInterface Class Diagram

- *RestaurantController:* This class is only used by the restaurant admin to update the restaurant configuration settings, such as name, description, address, currency and cancel time.
- *TableAdminController:* This class is only used by the restaurant admin to create/modify the table layout for the restaurant.
- *TableController:* Both waiters and the restaurant admin to view the table layout of the restaurant use this class.

The HTML component has the following controller classes:

- *AccountController:* It is used to authenticate the restaurant admin and waiter.
- *HomeController:* This class is responsible for rendering all views including Customer View, Waiter View, Admin View and Login View.
- *QRCodeController:* This class is used to generate the QR codes for all seats of the restaurant.

The Socket component has the following classes:

- *MenuPanelNotification:* This class is used to provide real-time communication for Customer View
- *WaiterPanelNofification:* This class is used to provide real-time communications for Waiter View.

Secondly, the classes in the Business Layer are displayed in Figure 6-2. This layer has all the core business logics of the menu management system. These logics are implemented on four main components using five classes and four interfaces. The functionalities of each class are explained below:

- *CommonRules:* This class implements all common business rules for the four components.
- *MenuManager:* This class is responsible for performing all the CURD operations on the entities Menu, MenuCategory, MenuItem and MenuItemOption entities. In addition, this component implements the logic for validating the seat code on each request from customers. The public methods of this class are provided to the next layer via the IMenuManager interface.
- *UserManager:* This class is responsible for creating the unique ID for all users including customers, waiters and admin. It also maintains the online status of each user and store the notifications sent by them in the backend of the system. All public methods of this class are provided to the next layer via the IUserManager interface.
- *RestaurantManager:* This class is responsible for managing the restaurant settings, tables, seats and generating order statistics. IRestaurantManager interface provides public methods of this class to the next layer.
- *OrderManager:* This class is responsible for CRUD operations of the Order object. Each operation from the customer's seat is validated, and then it is performed. All

the public methods of this class are provided to the next layer via IOrderManager interface

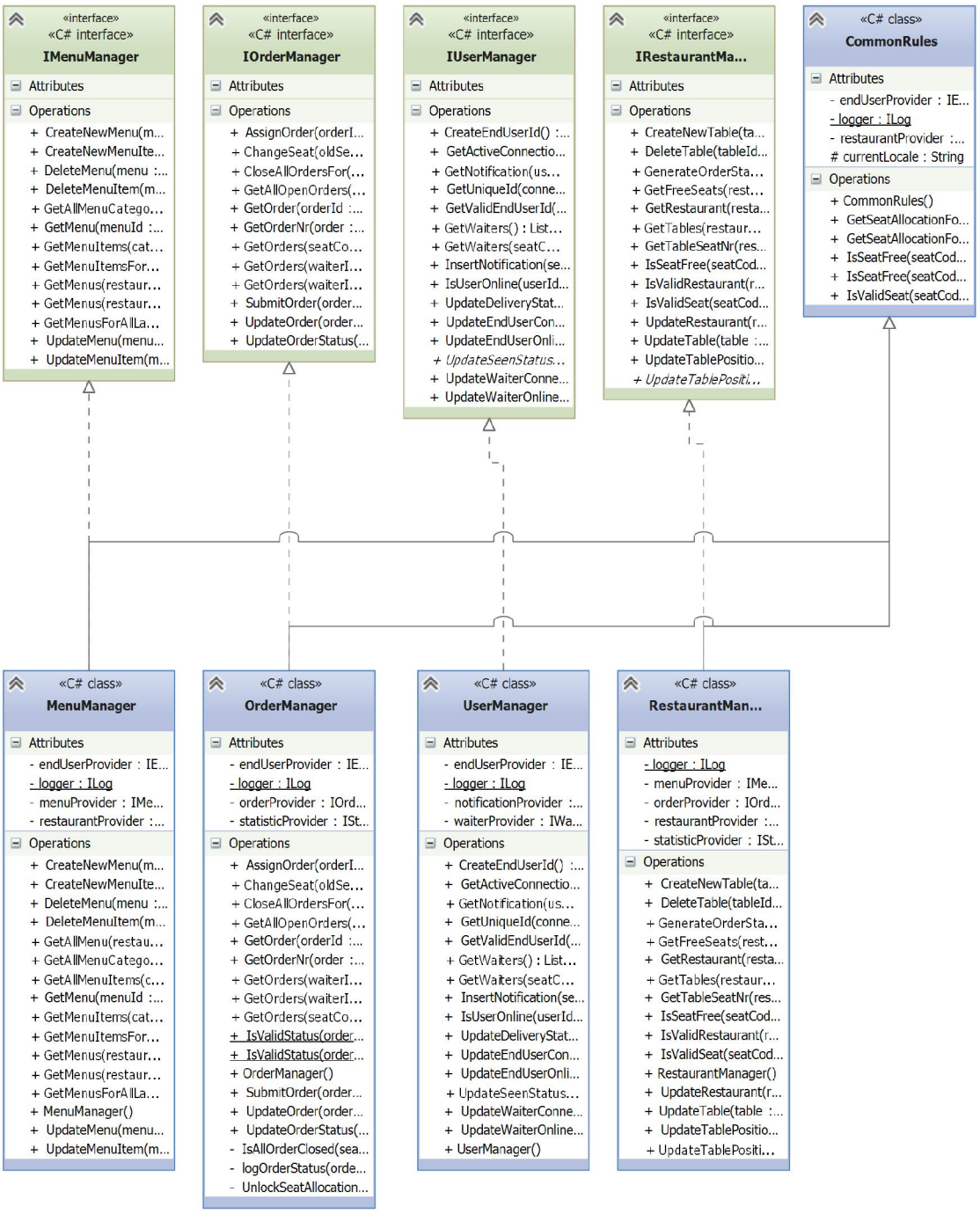


Figure 6-2 OMS.Business Class Diagram

Finally, the classes under the data layer are displayed in Figure 6-3. This layer has a number of data provider classes, which allow the CRUD operation on various entities. The CRUD operations are performed by the helper functions of the Entity Framework (EF). These

providers have the own interfaces that provide public methods to the classes in the Business Layer.

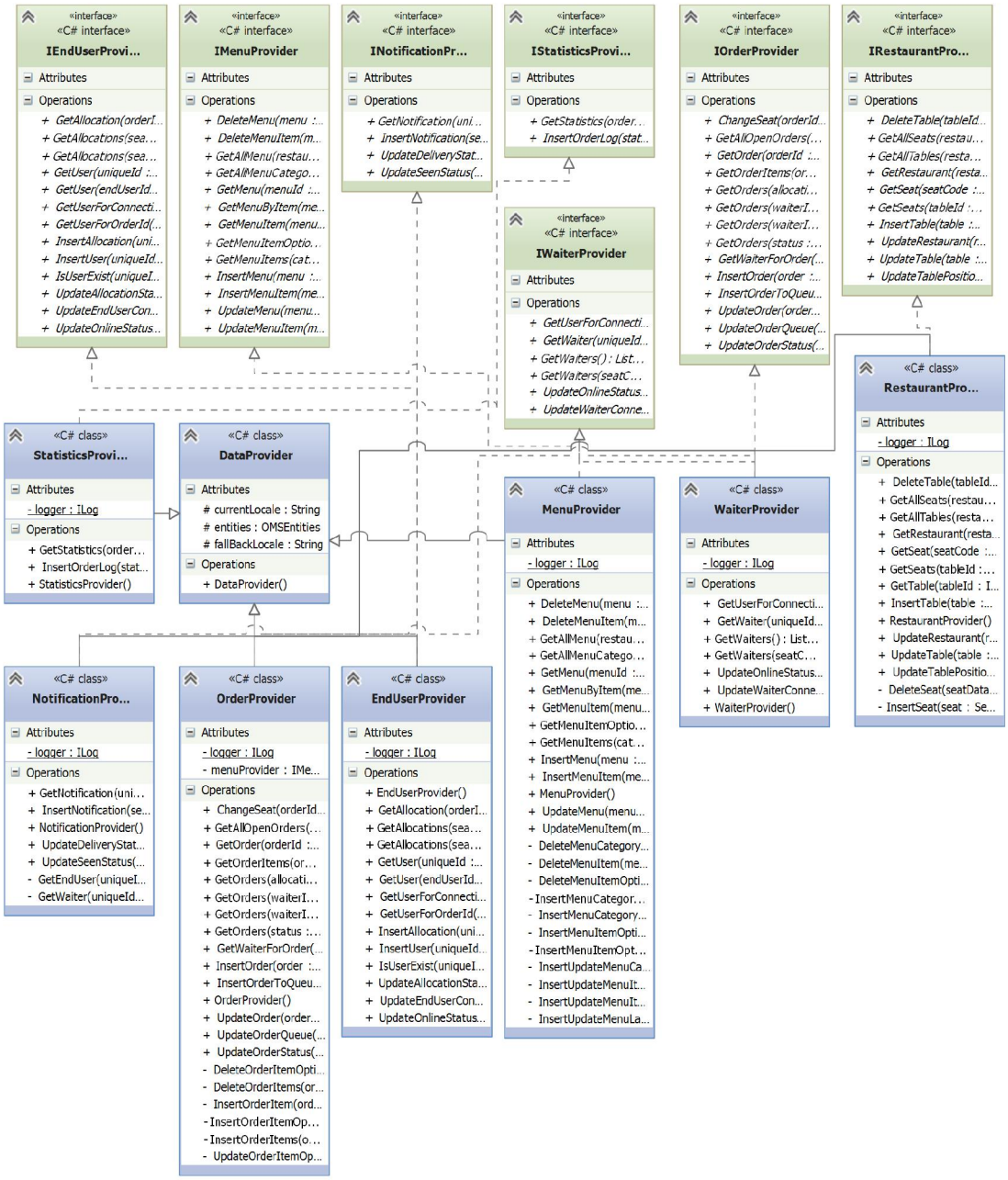


Figure 6-3 OMS.Data Class Diagram

6.1.4 Unit Testing

Unit testing is a method of testing individual units of source code in a project. In this system, the individual units are the service classes of the Web Interface Layer shown Figure 6-1. While developing each service class, unit test cases are created to make sure that the service class is free from business errors. The Visual Studio 2012 Integrated Development Environment (IDE) [18] is used for developing this project. This IDE has the integrated unit-

testing framework that is used for building unit test cases. Figure 6-4 shows the unit-testing framework integrated with the Visual Studio 2012.

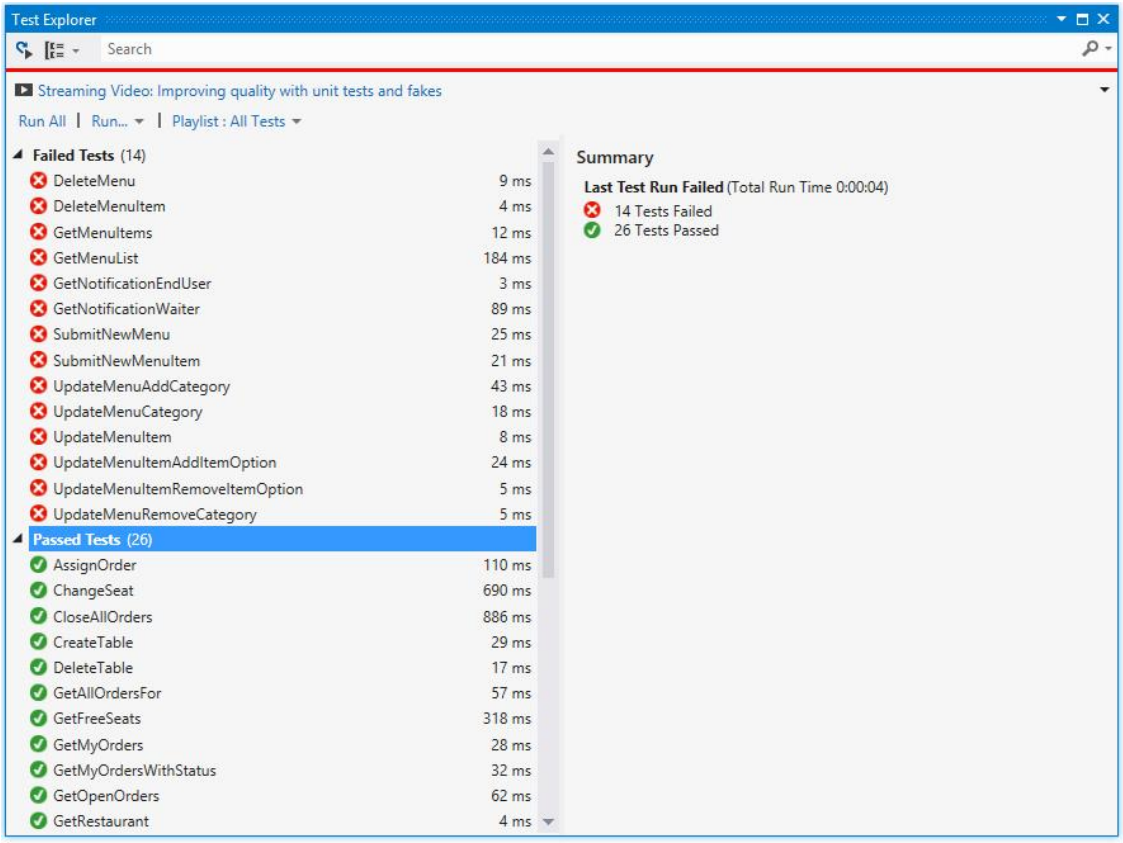


Figure 6-4 Unit testing of WebInterface Classes

Here all the passed and failed test cases are indicated as green-tick marks and red-cross marks respectively. In addition, each failed case shows a reason for the fail and the problem location in the source code of the project. This unit testing process helps to prevent the violation of business cases by the service classes that change while developing the project.

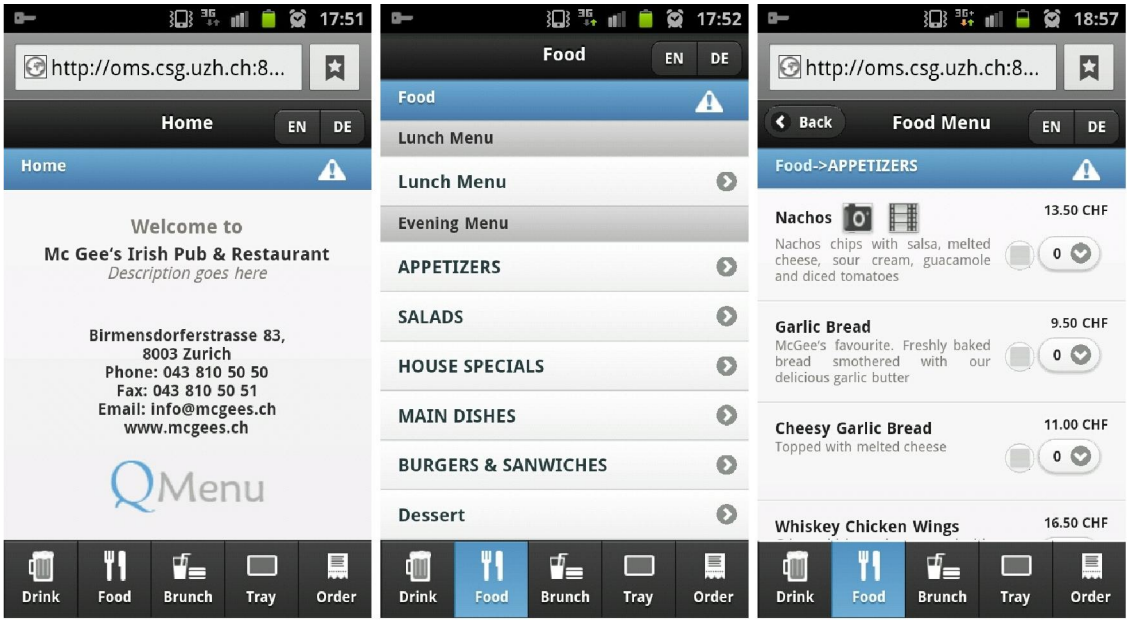
6.2 Graphical User Interface

In order to use an automated ordering system, the Graphical User Interface (GUI) is developed for all kinds of potential users. There are three different GUIs: Customer View, Waiter View and Admin View, and these views are used by customers, waiters and restaurant admin, respectively.

6.2.1 Customer View

Since Customer View is for arbitrary customers, the GUI should be user-friendly and simple to use. The language bar is placed in the header of all pages on Customer View, so that the customers can change the language at any time. The fixed bottom navigation bar is the main feature that makes the interface very understandable. It contains three menu names (Drink, Food and Brunch), as depicted in Figure 6-5, the Tray and Order sections which give

the customers an intuitive hint that selected menu items should be added to the tray and the order should be placed. The icons placed in the fixed bottom navigation bar make the customers easily understand the content of each navigation section. Highlighting the current navigation bar section helps customers not to be confused about the current step of ordering process.



A *B* *C*

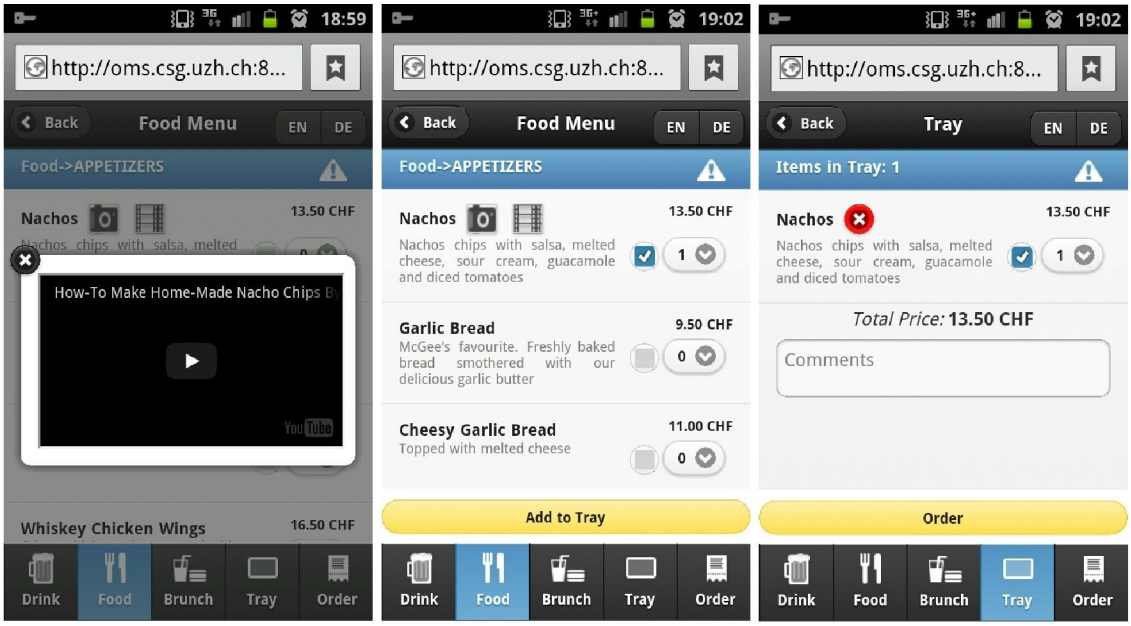
Figure 6-5 Customer View (1)

In general, the ordering process can be logically divided into three steps: selecting a menu item, adding it to the tray and placing an order. The customers can see the menu categories by clicking on one of Drink/Food/Brunch navigation bar sections as shown in Figure 6-5. By clicking on the specific category, the list of menu items belonging to that category is displayed with details (name, price, description, photo and video) as depicted in Figure 6-5 (C). The Figure 6-6 (A) shows that the photos and videos can be viewed by clicking on appropriate photo and video icons next to the menu item name. This helps the customers to make a quick decision. When the menu item is selected, the 'Add to Tray' button of a distinctive color appears at the bottom, as shown in Figure 6-6 (B), implying that menu items need to be added to the tray.

Once the customer clicks on that button, the GUI automatically navigates to the Tray page, as shown in Figure 6-6 (C), which lists all the added items. More items can be added to the tray by repeating the same activity, and the tray can be changed before placing the order.

The 'Order' button of a distinctive color on the Tray page also prompts the customer to click on it in order to place the order. After pressing the 'Order' button, the application automatically switches to the Order page, as shown in Figure 6-7 (A), and the customers can cancel the order within a specific time period using the red cancel button. After the

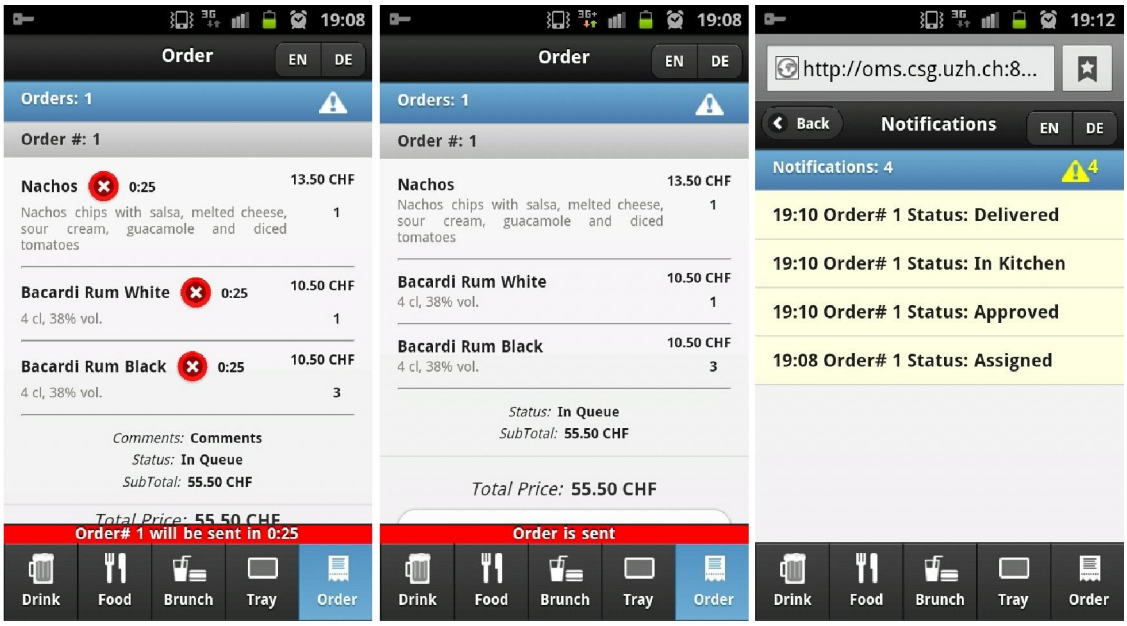
cancellation time expires, the order is sent to a waiter and a notification about the order delivery is displayed.



A *B* *C*

Figure 6-6 Customer View (2)

The Order page lists all orders with their details, as depicted in Figure 6-7 (B). When the status of the orders is changed by the waiter, the notification icon on Customer View is highlighted with the number of new notifications as shown in Figure 6-7 (C).



A *B* *C*

Figure 6-7 Customer View (3)

Notification page is opened by clicking on the notification icon, and then notifications are de-emphasized after they have been seen. The customers can call the waiter by clicking 'Call Waiter' button whenever help is needed. As Customer View supports real-time communication with Waiter View, all notifications on the change of network connection state are always displayed on the screen.

6.2.2 Waiter View

The waiters are trained in using Waiter View. But this view should also be easy to use and quick to manage orders, since it directly affects the efficiency of the ordering process. As shown in Figure 6-8, Waiter View also has a fixed bottom navigation bar containing Order, My Order, Map, Notifications sections, and the current section is highlighted when it is clicked. In general, there are two ways of managing the orders. In the first way, the orders are handled one by one in the list. The second way is about processing the orders with regard to the seat. The Figure 6-8 shows the order page that presents the queue of open orders. When new orders arrive, the section name is highlighted with the number of open orders.

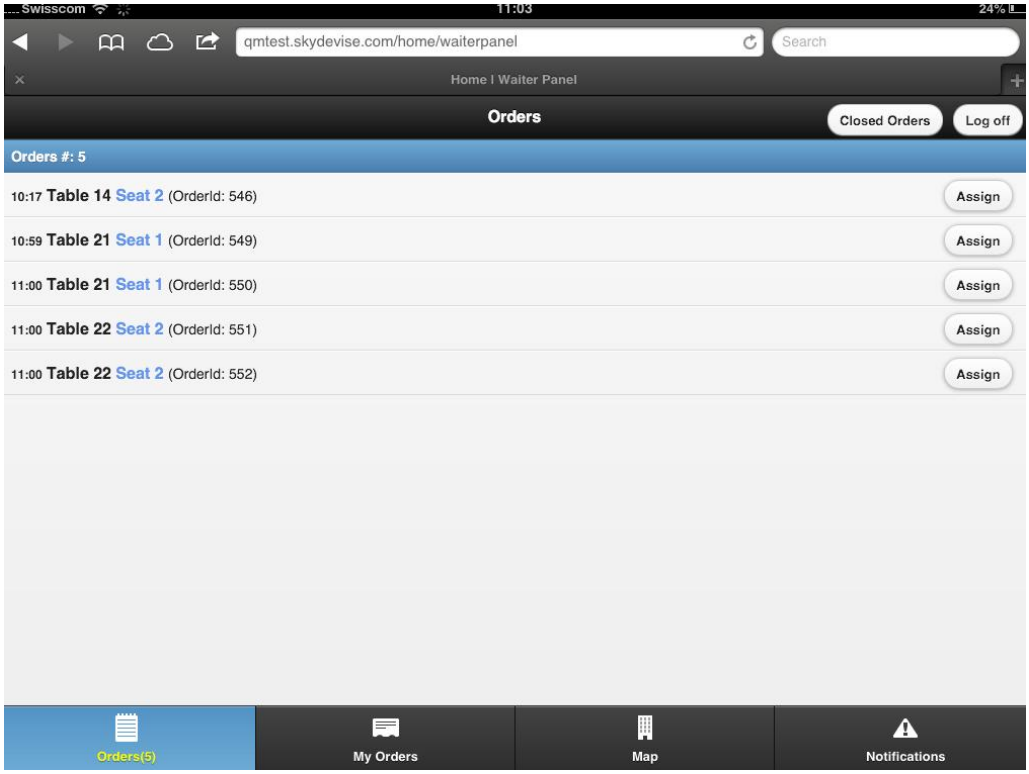


Figure 6-8 Waiter View – Orders Page

The open order is assigned to a waiter by clicking the Assign button, and then the GUI navigates to the My Orders page, as depicted in Figure 6-9, that lists all orders assigned to this waiter. On this page, the waiter can filter his/her orders according to the status of the order (Assigned, Approved, In Kitchen and Delivered). Here, the background color of delivered and undelivered orders differs, which makes the undelivered orders easily

visible. There are two pages, namely Details and Seat Orders, which show the details of orders and are accessed by 'Details' and 'Go to Seat' buttons, respectively.

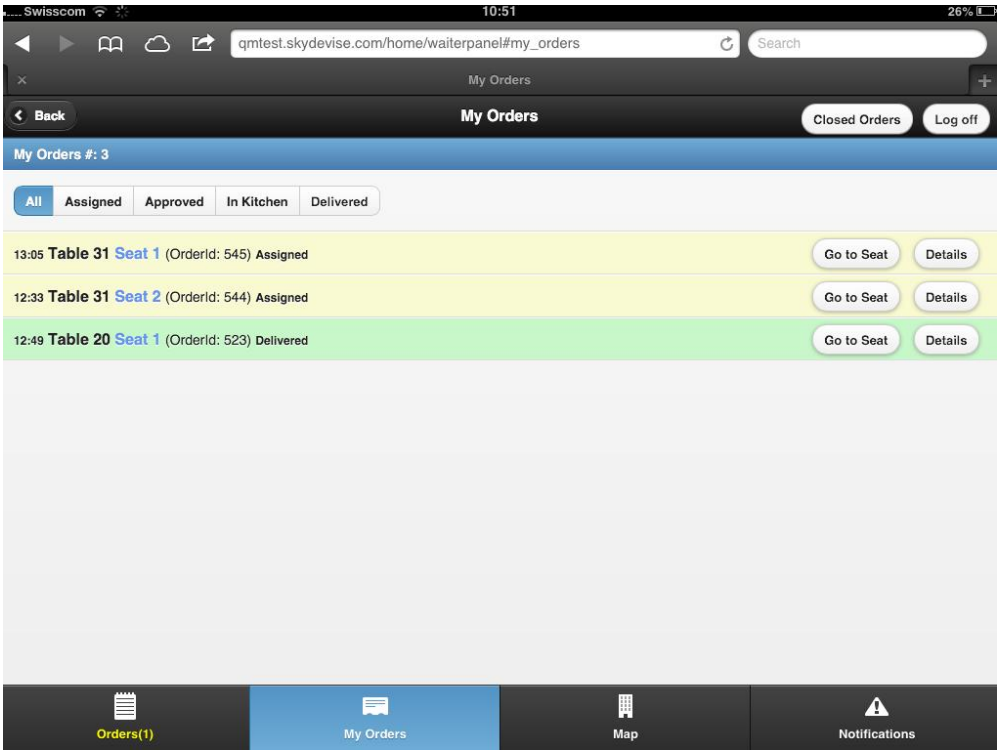


Figure 6-9 Waiter View – My Orders Page

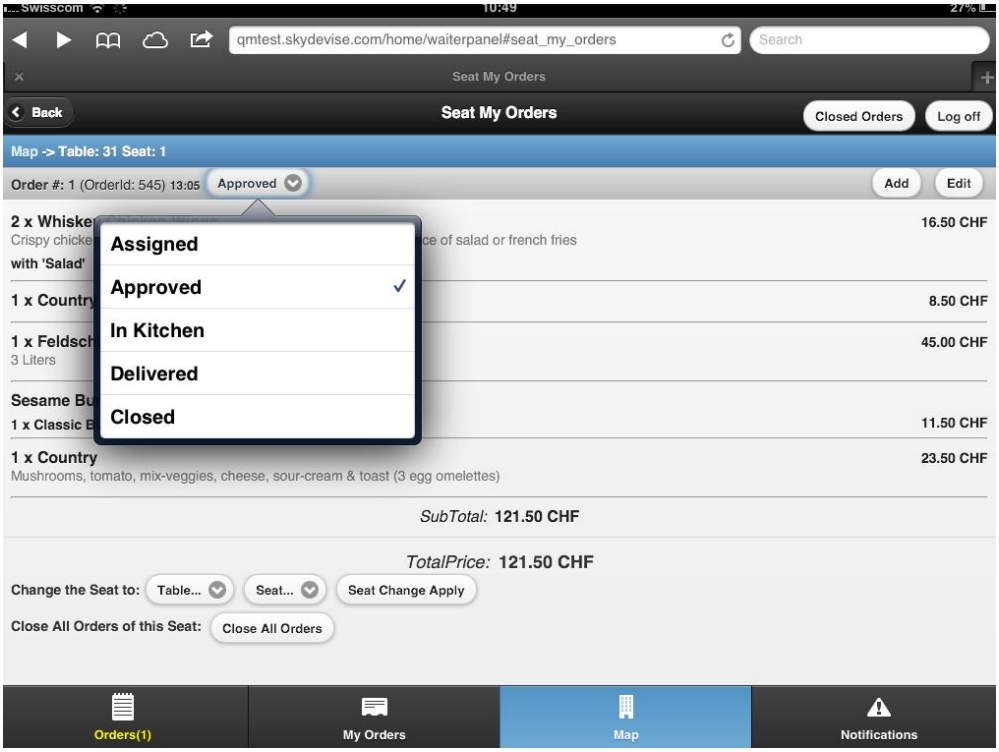


Figure 6-10 Waiter View – Order Detail Page

The Figure 6-10 shows the Order Details page that displays the details of one order, while the Seat Orders page shows the details of all orders for the selected seat. On both pages, the waiter can see the menu items, quantity, prices, comments, and can change the status of the order. Consequently, the appropriate status change notification is sent to Customer View. The 'Add' and 'Edit' buttons navigate to the pages where either new menu items can be added to the order or existing menu items are edited for the given order, respectively.

The Map page shown in Figure 6-11 provides the visual representation of the restaurant tables. The seats on the table are colored to indicate the current state of that seat. The red color indicates that there are new orders on this seat, while green color means someone has occupied that seat. If the seat does not have any color, it means that the seat is free. By clicking on the seat, the waiter can go to the Seat Orders page, which is described above.

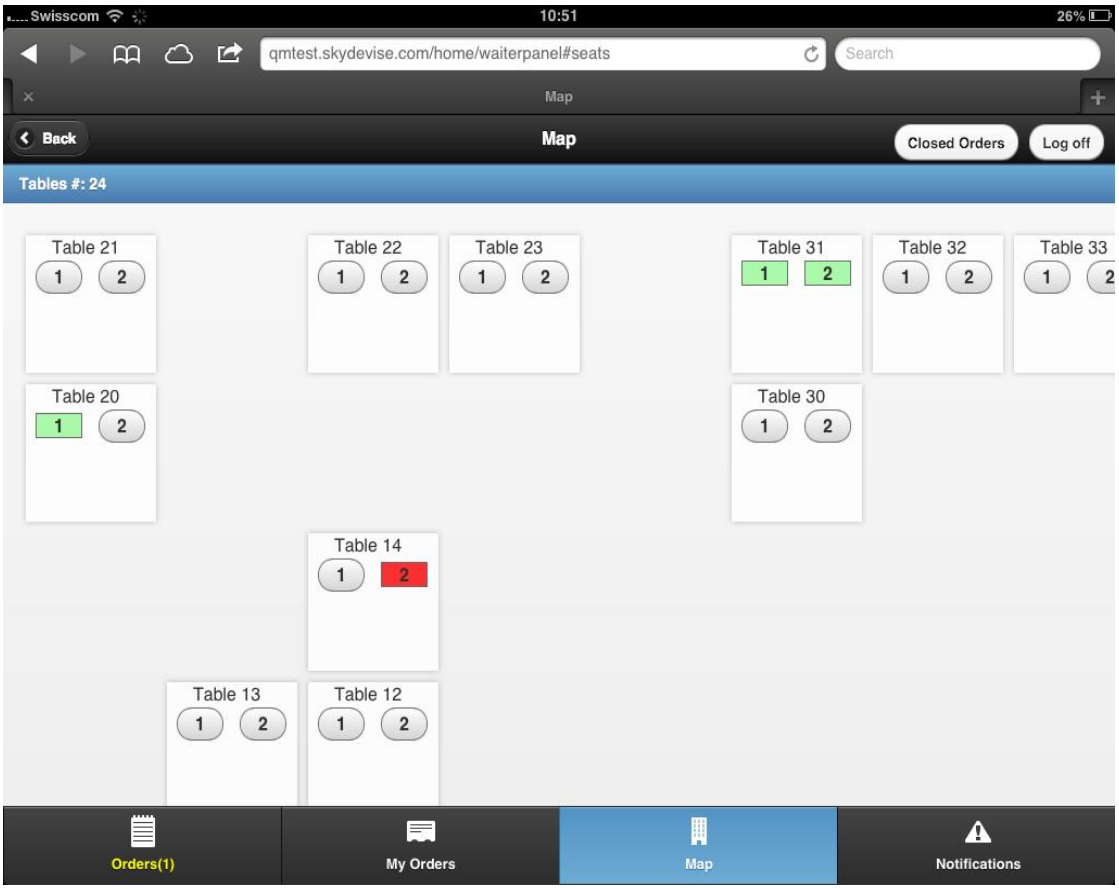


Figure 6-11 Waiter View – Map Page

The Figure 6-12 depicts the Notifications page that shows whether the customers are calling the waiter or not. The yellow background color for notifications and yellow color for notifications count makes the view easy for waiters to notice new notifications. All the actions of the waiter such as assigning an order, changing status, adding menu items, and editing an order are displayed as a popup to notify the waiter whether his/her action succeeded or not. As Waiter View interacts with Customer View in a real-time

communication manner, the status of the connection also pops up to make the application usable.

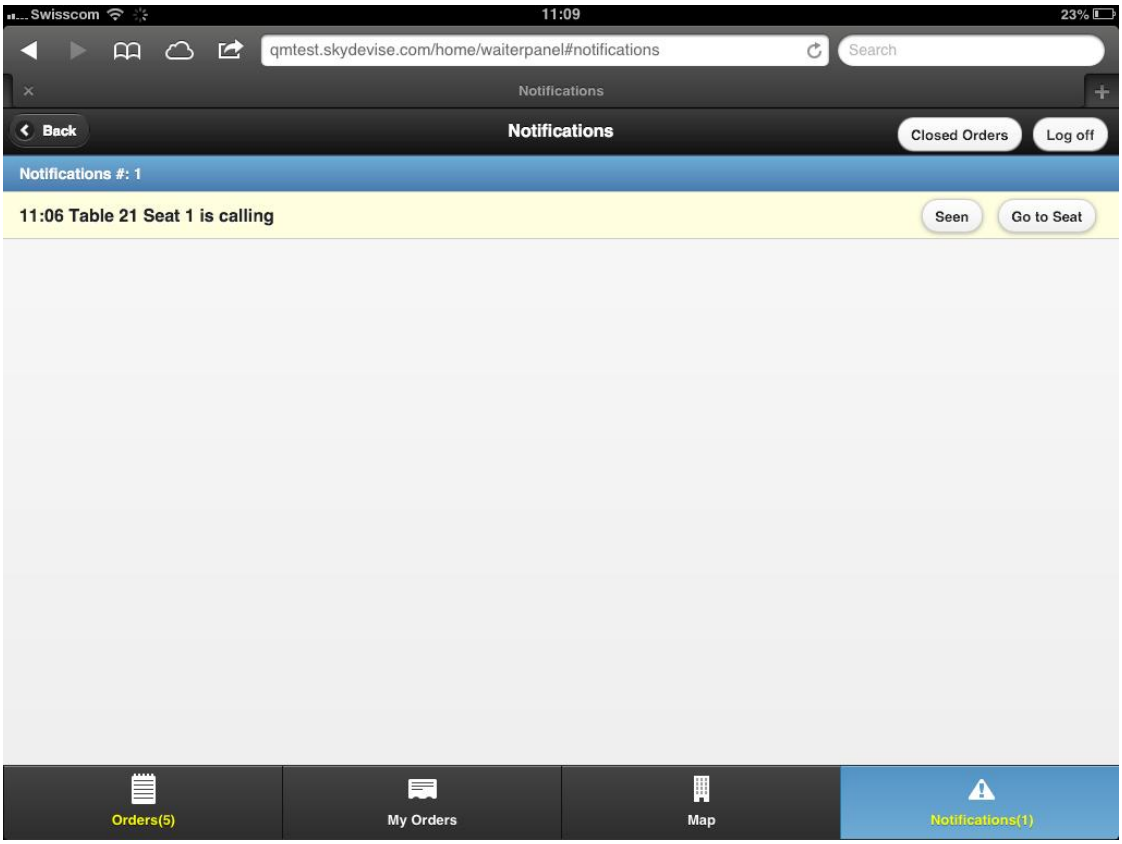


Figure 6-12 Waiter View – Notification Page

6.2.3 Admin View

The Admin View is designed for restaurant admin with basic training session. Therefore, the interface should be quite easy to learn to use. Admin View is responsible for managing menu data, users, layout of the tables map, reporting order transactions and generating QR codes. As shown in Figure 6-13, the main navigation bar consists of Menu, Menu Items, Table, Restaurant, Statistics, User Management and QR code sections, which are located on the top of the view and is fixed. Each of these sections is independent of each other in terms of activity completeness.

The Menu page is responsible for adding/editing/deleting menu name, description and menu categories. The restaurant admin can add/edit/delete menu item name, description, price, picture, video and menu item options on the Menu Items page. Rearrangement of the tables on the map, adding/deleting tables, editing the table number, adding/deleting seats editing the seat number can be done on the Table page. When seats are created/edited the appropriate QR codes should be created. The Figure 6-14 shows the QR code page that automatically generates the QR codes with given input parameters (URL, width and height). If there is a need for more waiters, additional users can be added and the roles can be changed on the User Management page. The restaurants details such as

contact information, currency, and cancel time for an order are edited on Restaurant page. Statistics page serves as a reporting tool showing all order transactions for a given period of time.

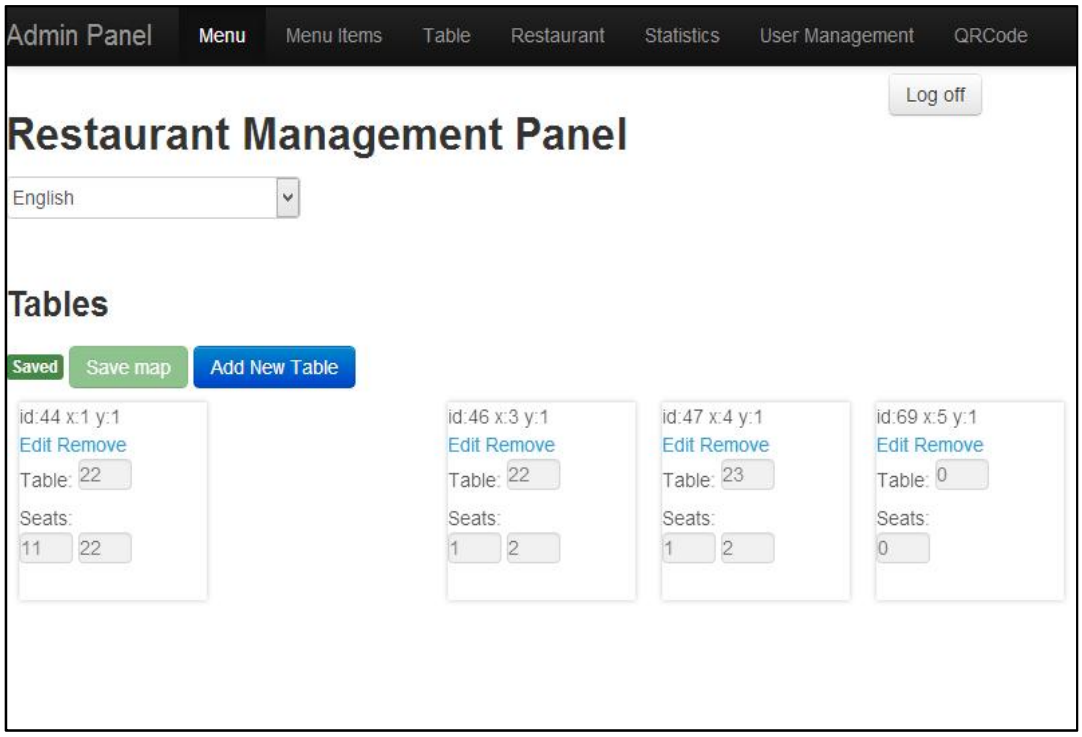


Figure 6-13 Admin View (1)

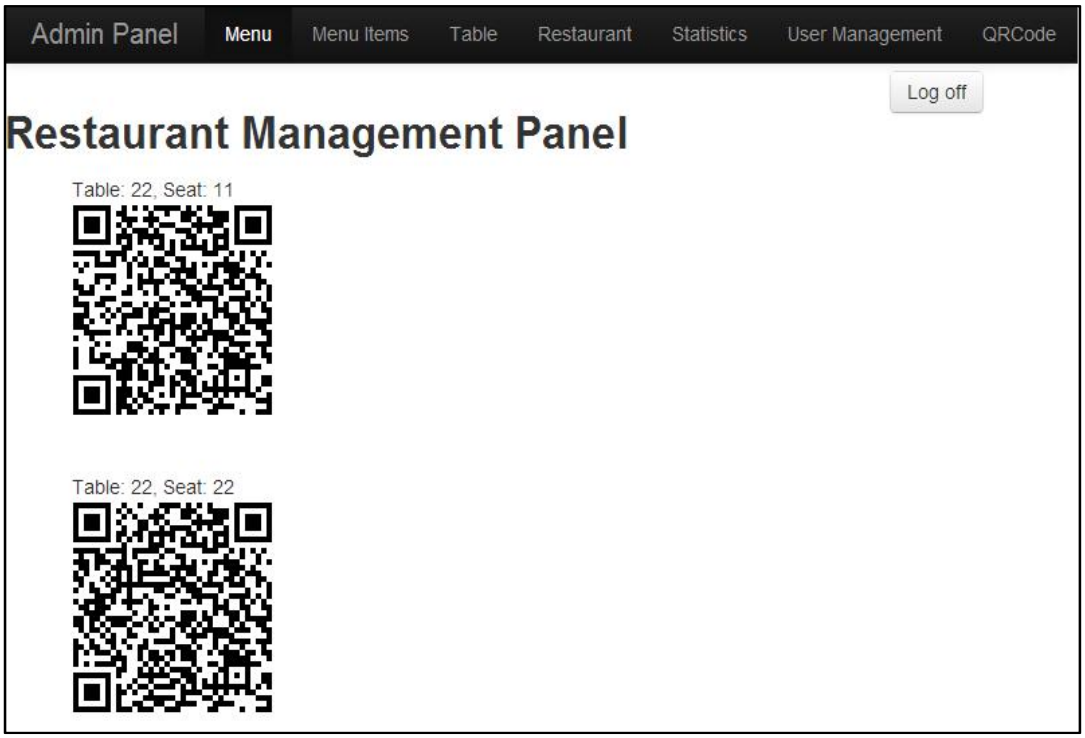


Figure 6-14 Admin View (2)

6.2.4 Technologies used

The following libraries were used for the implementation of Graphical User Interface.

- jQuery Mobile [19] - It is touch-optimized web framework which is based on HTML5 and it allows to build web-applications for all popular smartphones and tablet platforms
- KnockoutJS [20] - A standalone JavaScript implementation of the Model-View-ViewModel pattern
- gridster.js [21] – It is used to visualize the tables map of the restaurant
- iScroll [22] – It is an implementation of horizontal scrolling for mobile devices
- jquery.cookie [23] - It is used for managing cookies
- SignalR [15] – It is an implementation of real-time communication with the server

jQuery Mobile is used to build the interfaces of Customer View and Waiter View. These interfaces use a multi-page structure that allows creating several pages on a single HTML document. The advantages of this approach over a single-page structure are that all pages can be downloaded at the beginning and navigation between internal pages can be performed very fast, thus making the application very responsive to customer actions. All further requests are executed asynchronously using AJAX technology that makes the use of the application very light and smooth. This library makes the interface adapt to the device dimensions, and the DOM elements become touch-based according to the platform. Since Admin View is used on the desktop browser, jQuery Mobile library is not used.

KnockoutJS implements Model-View-ViewModel (MVVM) architecture pattern on the client side. The main concept of this architecture style is to separate the presentation from the data and logic, which make the application much easier to maintain, evolve and test. Using this pattern, a lot of business logic is implemented on the client, making the application very dynamic, rich and responsive. The library provides features such as declarative binding, automatic UI refresh and dependency tracking. Declarative binding helps associate DOM elements to model data with concise syntax that allows to template the view and avoid hardcoding. Automatic UI refresh updates the view whenever the state of the model data changes. For instance, when new items are added to the tray, the view is automatically updated to show the last state of the tray. Dependency tracking is very convenient when the data of view model is computed from the states of other model data. For example, the total price for an order is calculated from the quantity and price. When people change the quantity, the total price is recalculated immediately and the value is changed automatically thanks to dependency tracking feature.

gridster.js is a jQuery plugin which builds a drag-and-drop multi-column grid. It is used to build the layout of the tables map for the restaurant. By default, this library does not have enough functionalities to meet our requirements. For example, when the tables are dragged and dropped, they shift other tables and all tables automatically flow up. To make all tables fixed in terms of their position and allow them to be dragged and dropped only into empty positions, additional functions are written.

iScroll is used to enable horizontal scrolling for the tables map. Depending on the size of the tables and the layout, the map may not fit in the screen of Waiter View.

jQuery.Cookie is a simple, lightweight jQuery plugin for reading, writing and deleting cookies. This plugin is used to store the data permanently on the client to support page refreshing and provide identification data to the server.

6.2.5 Test Cases

Since the menu management system is used in actual business process, conducting test cases is very important. In order to make sure that the system works without any faults, the list of test cases is prepared for each actor's action based on the functional requirements. Test cases can be divided into three groups depending on which actor is performing activities: test cases for Customer View, Waiter View and Admin View. Only test cases for Customer View and Waiter View, which are used by customers and waiters, are written, as these subsystems directly affect the main process of the restaurant business. Testing Admin View is done only on the development level. Customer View test cases include how to connect to the system, how to access menu/menu items, checking the correctness of the data (e.g. name, description, price of menus), changing the language of the menu, choosing the food/drink items and adding them into a tray, placing an order, cancelling the order within specific time, getting a notification on the status of the orders, and calling the waiter for a help or payment. The test cases for Waiter View cover the following activities: logging into the system with created waiter credentials, getting notifications when new order arrives or a customer calls the waiter, assigning the order to the waiter, changing the status of the order, adding/editing menu items into/in the order, changing seats for orders, accessing orders from the map, and closing the orders.

The list of all test cases with details and results can be found in Appendix. All test cases are performed individually by each member of the project team and passed successfully with the current version of the system.

6.3 Deployment

In the deployment phase, all the devices required for this system are pre-configured, necessary software are installed on the server, application is setup in the server, and then all the devices are deployed in the restaurant. There are several steps involved in this phase. Firstly, the Windows Server 2008 has been setup for the menu management system. Secondly, the Internet Information Services (IIS), SQL Server 2008 R2 and .NET Framework 4.5 are installed in that Windows Server. Afterwards, the menu management system is deployed in that server. Thirdly, the VPN router is preconfigured to connect to the windows server. Finally, the Cisco Wi-Fi access point is configured and connected to the VPN router. After the deployment date is confirmed with the restaurant management, the VPN router and Wi-Fi access point are deployed in the restaurant. Finally, the test runs is conducted to make sure that the application is accessible from the restaurant.

Chapter 7. Results and Discussion

As described in Chapter 5, an experiment is conducted to prove the hypothesis of order processing time and customer satisfaction. In this chapter, the results of this experiment are described. To prove the hypothesis statistically, Six Sigma methodology [1] is used and implemented using the Minitab tool [2]. In the first sections, the pre-deployment data collection is described, and in the later sections, the post-deployment experiment is detailed. At the end, the analysis and comparison of the pre and post-implementation data is made and the meaning of the statistic results is discussed.

7.1 Pre-Deployment Data Analysis

The aim of the project is to study the effects of introduction of the menu management system on time it takes to serve a customer in the restaurant. In accordance to the objectives set as in Chapter 1, the following goals have been set as Hypothesis in Chapter 5 along with the restaurant management:

- The drinks order in the restaurant should not take more than 7min to deliver.
- The customer should be able to make a second order as soon as he/she intends without any wait time.

7.1.1 Data Collection Activity

Data collection activity for manual process at the restaurant started in early November and was completed in mid-January. There have been many challenges in the process as described below:

- During the winter season, the terrace was closed; therefore collecting relevant data as per the experiment setup was a challenge.
- During the Christmas season, many customers were on vacations and hence the restaurant was not busy. Data collection activity suffered in December.
- The initial idea was to let the waiters do the data collection activity but no data was collected and reported for three weeks. As a result the project team actually went to the restaurant for manual data collection.

The manual data is collected for 7 days by the project team over a period of 75 days available for reference in the CD. There have been interesting observations from the data collected during the data collection process, they are:

- When the restaurant is full, most of the customers directly go to the bar and order the drinks.
- During busy times, the waiter usually forgets to take orders from new customers, hence the customer experience long serving time as per the project team's observations.
- When the restaurant is busy even two waiters are not able to reduce the order processing time; one waiter stands inside the bar and prepares drink, and other waiter goes around to collect orders and bring empty glasses.

The data collected in the section above is filtered by the orders from the bar and the food orders. Therefore, only the drinks orders at the table are considered. After filtering, 30 valid orders are used in the analysis.

7.1.2 Graphical Analysis

As next steps, the data collected above are analyzed by plotting them on graphs. As per the chart shown below, see in Figure 7-1 and Figure 7-2 the data is normally distributed with a mean of 7.7 and a standard deviation of 4.324.

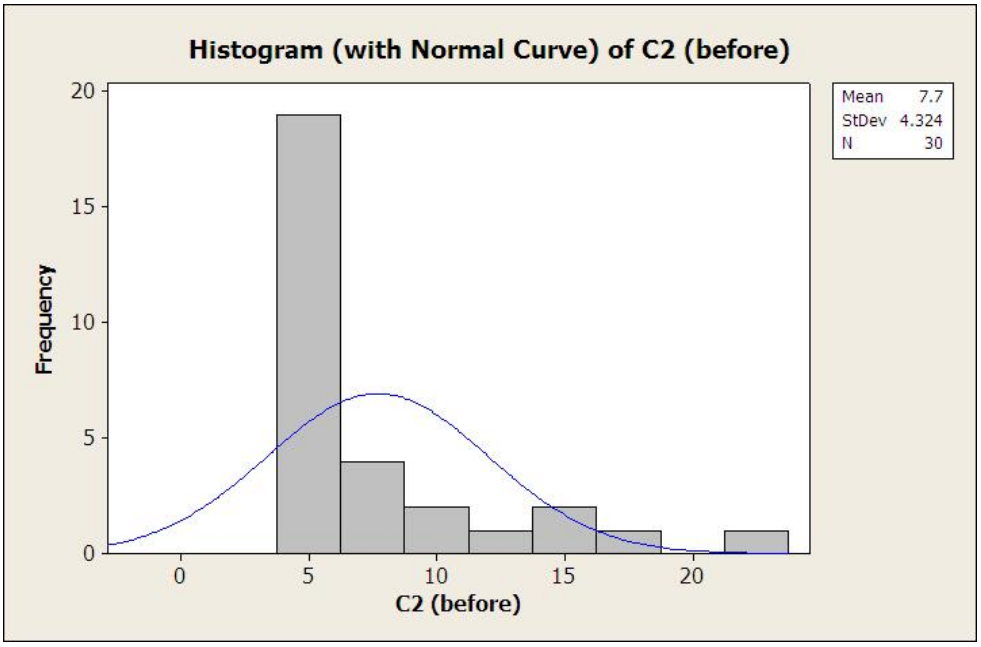


Figure 7-1 Graphical Summary

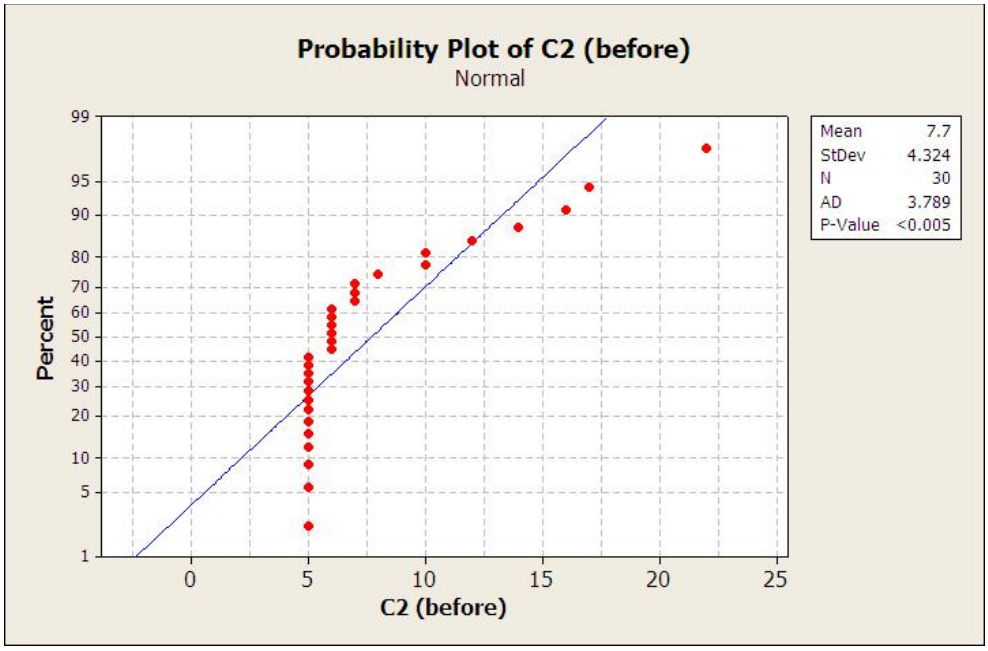


Figure 7-2 Probability Plot

7.1.3 Process Capability Establishment

Capability analysis for Normal data is calculated and displayed in Figure 7-3 and Figure 7-4 using Minitab. Z-Bench is the Z score for the Expected PPM (Parts per Million) calculated as:

Z-Bench = normsinv (1-(Expected PPM/1,000,000))

This formula is implemented in Minitab automatically, and the Z-Bench capability for the historical data is negative as shown in Figure 7-3. This clearly indicates that the current process is not good and has no potential for improvement as the Z-Bench Overall capability is -0.16, and the potential (scope for improvement) is -0.24 sigma.

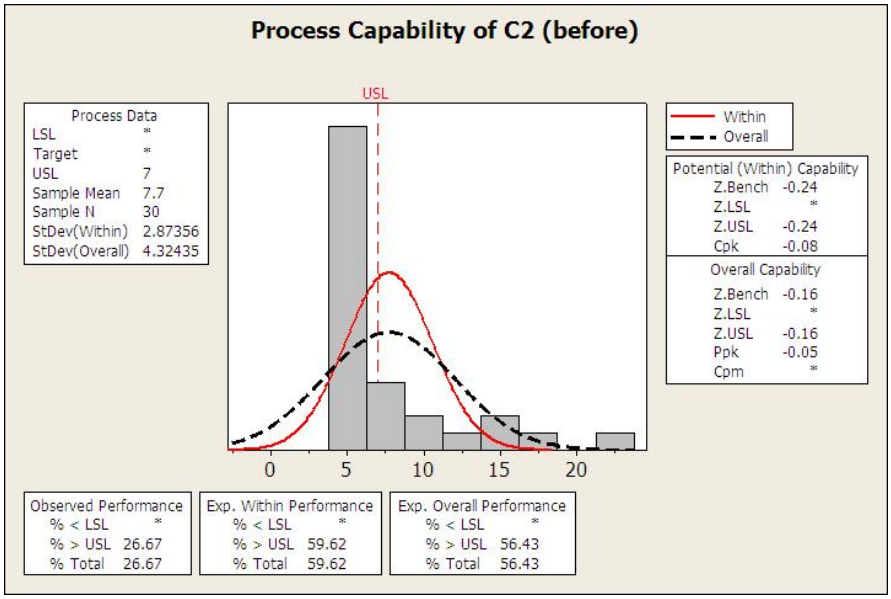


Figure 7-3 Process Capability

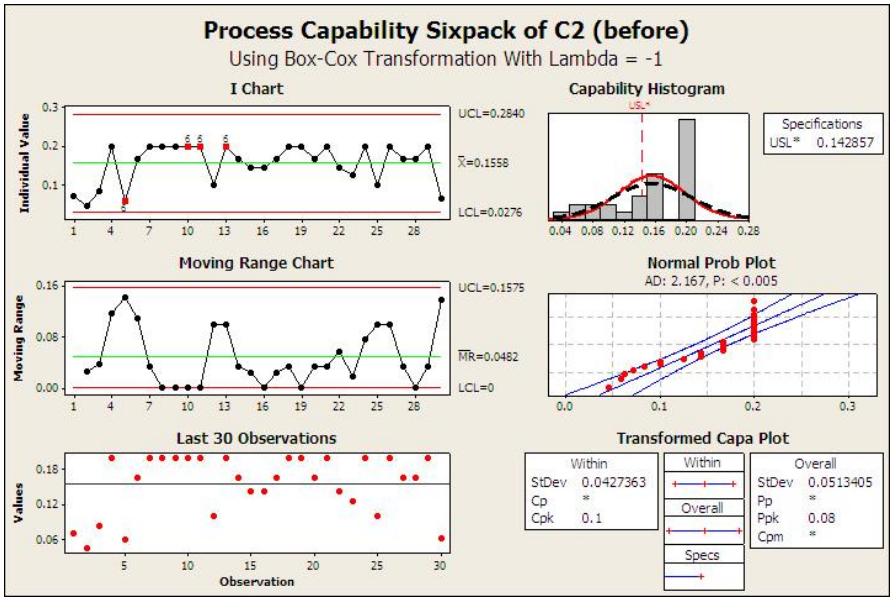


Figure 7-4 Process Capability Six Pack

As shown in Figure 7-4, the six-pack diagram shows the representation of the data in *I* and *MR* Chart. The interpretation of the chart shows that there are too many data points on the control lines which means that the process is out of control. The charts indicate that the current process is not good statistically and a new process should be introduced.

7.2 Post-Deployment Data Analysis

Similar data analysis is done with the order processing time after implementation. Since the order processing time is stored in the application database, the process of data collection is automatic from Admin View. The application stores all drinks orders from the order submission time to the order delivery time. The mobile application ran for three weeks to collect the data, and the data is available in the CD for further reference. From this data records, only the drinks orders are filtered for data analysis. For first two days, the waiters are not well versed with the application, and the collected orders on these two days were discarded. The first 30 orders are selected and analyzed for hypothesis testing. The data points were collected within three weeks instead of planned six weeks, because customers liked to try the new system.

7.2.1 Graphical Analysis

The data analysis histogram is shown with the normal curve for the experiment data after deployment. As shown in Figure 7-5, mean is 3.36 and standard deviation is 1.86. The x-axis displays the data points of order processing time in minutes over the frequency of occurrence.

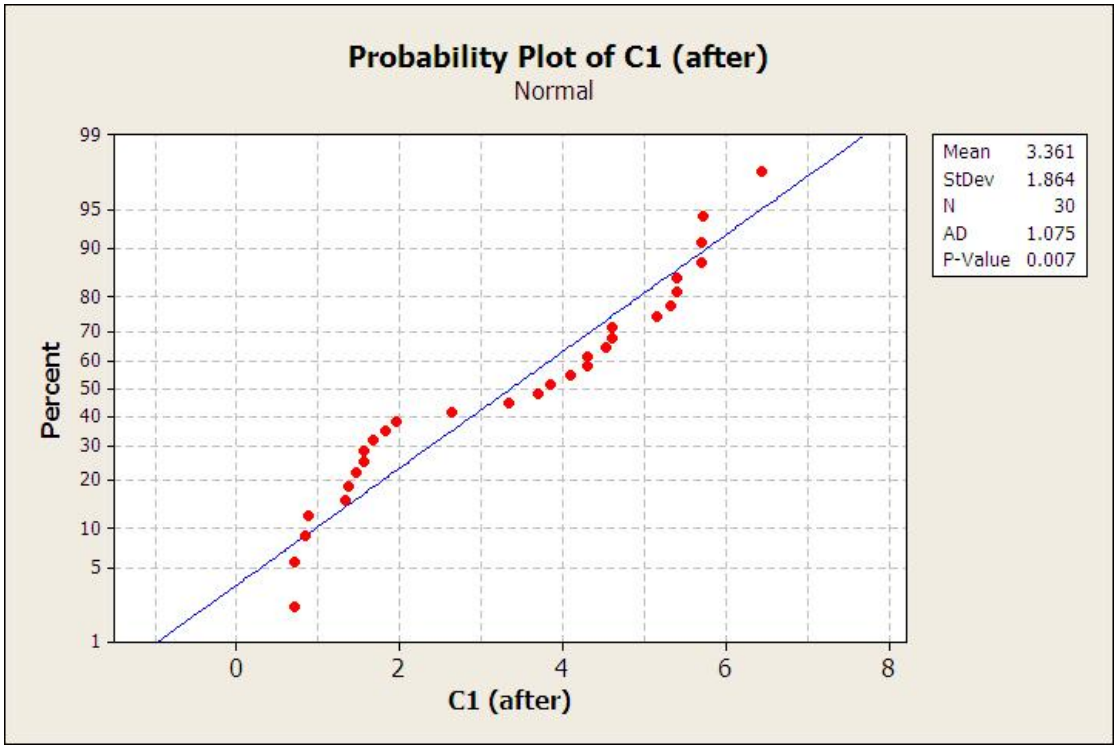


Figure 7-5 Probability Plot

7.2.2 Process Capability Establishment

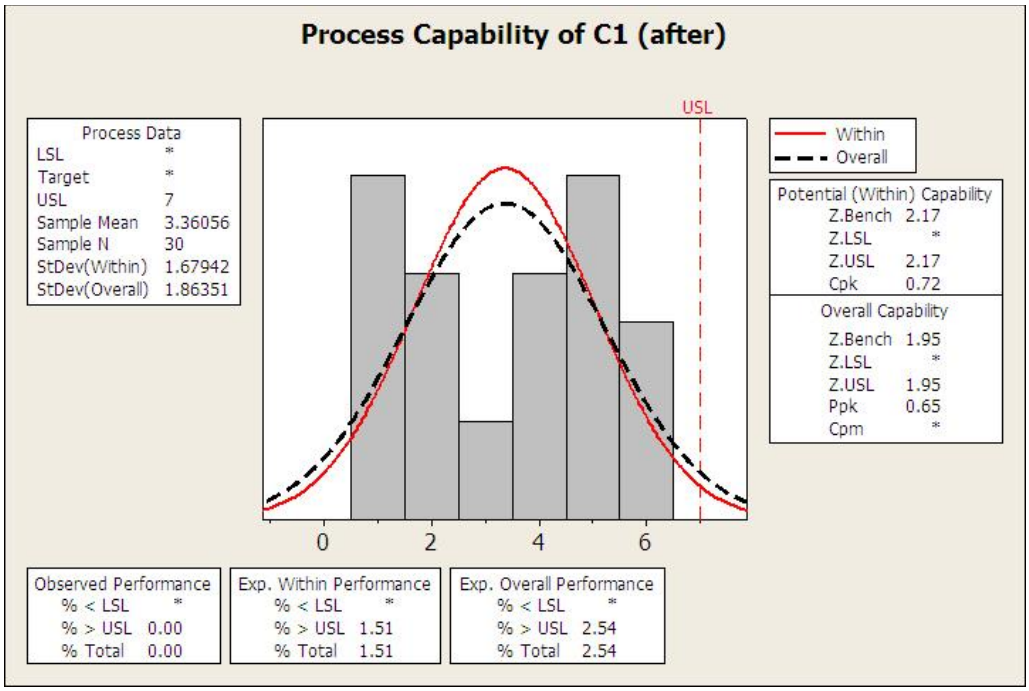


Figure 7-6 Process Capability

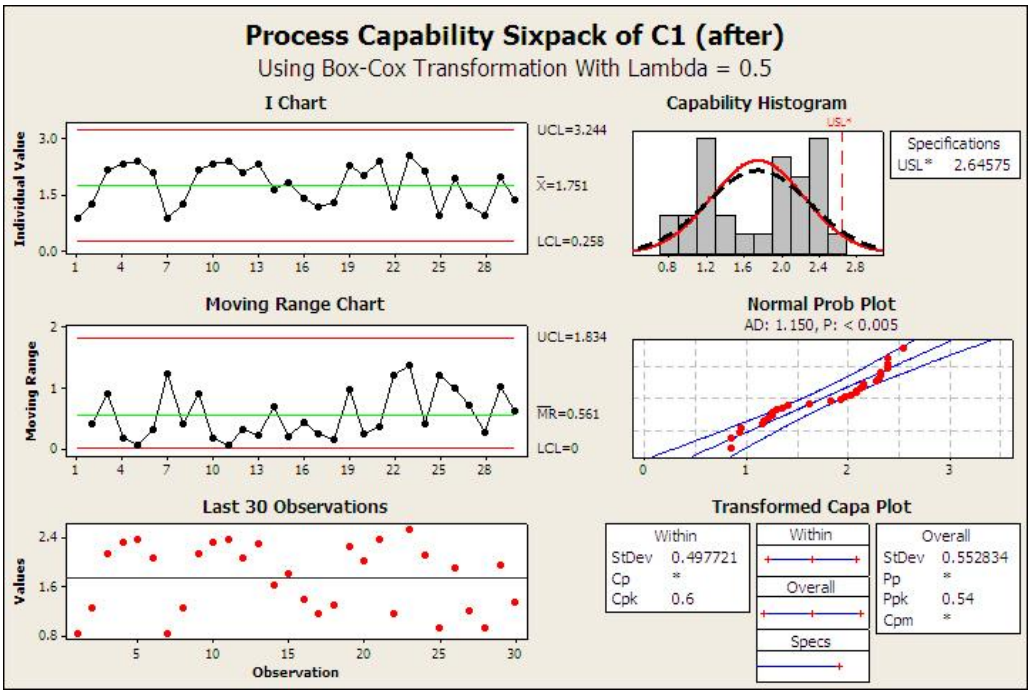


Figure 7-7 Process Capability Six Pack

Capability analysis for Normal data is calculated and displayed in Figure 7-6 and Figure 7-7. As shown in Figure 7-6 and Figure 7-7, the Z-Bench capability is now positive. This clearly indicates that the current process is good and also has a potential for further

improvement. The Z-bench Overall capability is 1.95 and the potential (scope for improvement) is 2.17 sigma. As in Figure 7-7, the six-pack diagram shows the representation of the data in *I* and *MR* Chart. The interpretation of the chart shows that there are very few data points on the control lines which means that the process is also in good control. Thus, the statistical interpretation indicates that the new process is good statistically.

7.3 Hypothesis Testing and Statistical Comparison to Manual System

The data analyzed in Sections 7.1 and 7.2 shows that there is an improvement in the mean and standard deviation. The mean and standard deviation went down, thus proving the alternate hypothesis as per the study protocol in Chapter 5.

The two-sample T-Test is one of the most commonly used hypothesis tests in Six Sigma. It is applied to see whether the average difference between two groups is really significant or it is due to random chance. It helps to answer whether the average order processing time is less than the historical data, which was collected before the implementation of the mobile application. The results are shown in Figure 7-8. The statistic values are as under:

Two sample T for before vs. after

T = 10.13 P = 0.0000 DF = 26

A p-value less than 0.05 indicates that there is a statistic difference between the two processes as per the REF and in our experiment a p-value of 0.0000 confirms the hypothesis.

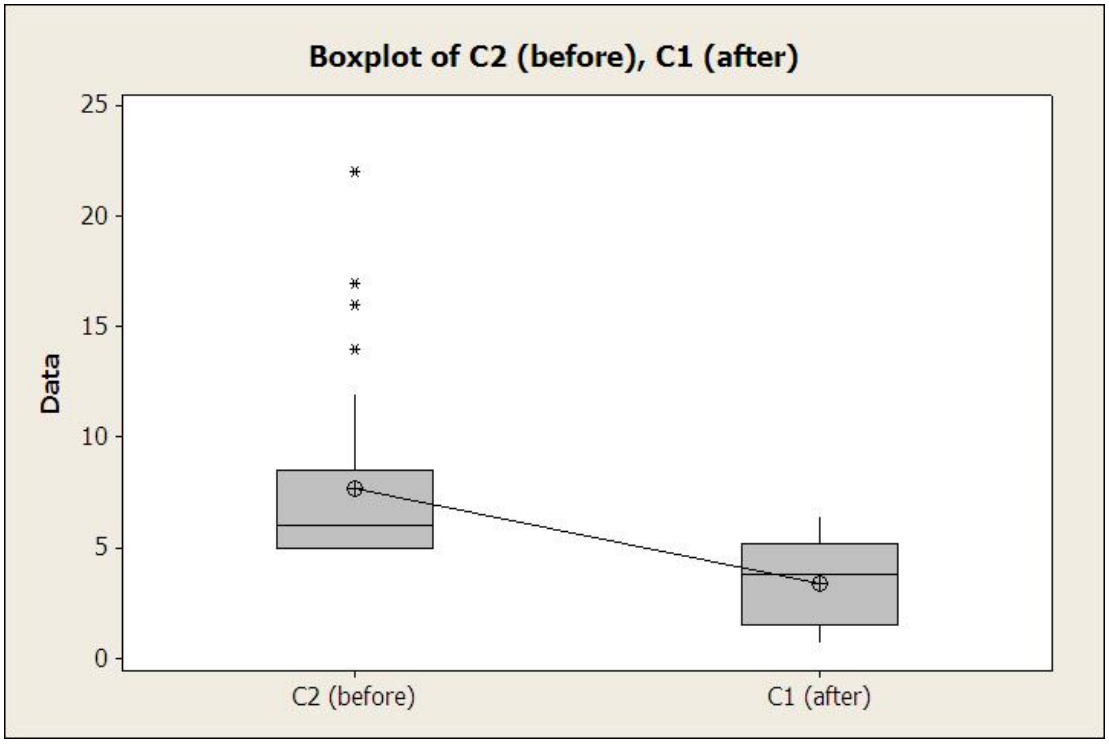


Figure 7-8 Boxplot Comparison

7.4 Customer feedback

The above results and analysis proves the hypothesis as per the study design and discusses the primary outcome as described in Chapter 5. This section explains the customer evaluation with five questions that has been conducted to address the average point on customer satisfaction. The first 10 feedback forms are collected from the customers, which are shown in Annex C. In result, the average is greater than 4, which proves the objective and the secondary outcome of having a score greater than 3. The highest average point score is for the question “Will you use it again?” This gives a positive indication that the customers are willing to use such a solution again for the restaurants. The lowest score is given to the “Ease of Usage”, which indicates that usability can be further improved and the scrollbars needs to be avoided.

7.5 Discussion

The results in Section 7.3 clearly shows that the hypothesis is proven and, the new system shows statistical improvement. There are two interesting observations made during the data analysis. By seeing the histogram in Figure 7-6, it is observed that the data is bipolar with two clear peaks on either side of the average that indicates there can be subgroups within the collected data. In the current version of the application, the billing system is not integrated with the menu management system. Hence, the waiter has to manage two systems at a time that makes the operations more complicated. Since one terminal is installed in the restaurant, the coordination between two or more waiters to use the application is more difficult. In addition, waiters forget to update the order status in busy hours. Since there is only one waiter login in the experiment, differential waiter behavior cannot be proved.

Another interesting observation shows that the Z-Bench value after the deployment of the application, as shown in Figure 7-6, is at 1.95 while the potential is at 2.17. Therefore, there is a scope for improvement in the new process with the integration of the billing module in the menu management system. This integration will make the waiter to interact with only one system, thus improving the waiter performance.

Chapter 8. Conclusion

8.1 Summary

An experiment was conducted in a restaurant setup to check statistically if the menu management system reduces the order processing time. The experiment is successful and the mean of order processing time reduced from 7.7 minutes to 3.36 minutes thus proving the alternate hypothesis. The second objective is to maintain or increase the customer satisfaction with the use of mobile application. The customer satisfaction has been proved with the average point of more than 4 in all evaluation form questions. This shows that customer satisfaction has increased with the introduction of mobile application. Hence both the objectives are met and, the experiment is successful. The menu management system aims at reducing the manual order processing time by introducing an automated solution. With the new system, the customers can place the order with their smartphones. The waiters can then get the orders and its location on their device. Order delivery task is still manual, but the waiters save the time going to the respective tables to collect orders.

8.2 Future work

After the application was deployed and the experiment was conducted in the restaurant, the project team spent more time at the restaurant to understand the customer reaction of the new system. The observation also helped to analyze the actual operations of the restaurant, and the associated improvements in the application can make it more helpful for the order processing activity. In the next two paragraphs, the improvements suggested by the customers and the restaurants are described.

The customers, through the evaluation forms and some directed conversations, suggested the following improvements to the application that the project team decided to implement in the future versions of the application:

- Many users felt that the scroll bar in the menu with a long list of menu items should be avoided, as it is difficult to scroll using the smartphone. Therefore, the future UI design of Customer View will be without the scroll bar.
- Others felt that shortcuts like ‘repeat last order’, similar to the manual scenario will increase the power of the application. The project team decided to observe the restaurant operations more closely on the current scenarios in the order taking process for identifying new features to the application. Order repetition shortcut is a good idea for the next release.
- UI needs to be improved even more and the ‘add to tray’ option should be reconsidered. Few users were not able to understand this feature and could not understand why the selected item is not reflected in the order screen.
- Reduce the number of clicks for placing the order for default settings on most frequently used items and drinks.

While observing the restaurant operations, and talking to the waiters and the restaurant management, the following new features have been identified:

- At the time of the experiment, the waiter has to manage two systems at the restaurant for an order. Waiter View has the customers' orders that the waiter monitors and updates the order status. A separate billing system also needs to be handled that can generate the customer bills at the time of payment. Therefore, integration with the billing functionality is a key for the order management system to be operational beyond the experiment.
- More than one waiter cannot view an order which is not good when more than one waiter is using the system. Hence multiple waiter synchronization is an important feature that should be added in the future versions of the application.
- During the development of the application, a lot of effort put to the usability of Customer View as they cannot be trained for the application use. But the project team realized that in order for the application to achieve the objectives in a better fashion, the usability of Waiter View should also be considered important. Hence, a better UI for Waiter View will be a part of the next version of the application.

8.3 Learning

8.3.1 Technology

The most important learning from technology perspective during the development, deployment and maintenance phase is discussed in this section. Firstly, the VPN has been used to make sure the system is only accessible from the restaurant as described in Section 4.2. However, the VPN connection is not stable as expected before deployment, and it gets disconnected in a specific interval. This leads to the system inaccessibility problem that is not usual when the system is used by the restaurant and customers. It can be resolved by validating IP address for each request against the IP address of the restaurant stored in the database. Secondly, sometimes accessing Customer View by smartphone browser takes long time. Since the system is designed as a web based application, hi-speed production server and hi-speed Internet connection are required. Finally, when a customer tries to connect to Wi-Fi network in the restaurant, it takes long time. Here the identified reason - sometimes the process of allocating an IP address for a smartphone by the Wi-Fi router is slow. Installing Wi-Fi router with hi-speed DHCP server that allocates the IP address to smartphone very fast can solve this problem.

8.3.2 Business

From the perspective of understanding the restaurant operations and the business, the project taught the project team a lot of lessons. Ideas and features that looked exciting on paper do not work always in a real situation. If the project were not done in the real setting of a restaurant by taking requirements and implementing in the experiment, the application would not have been the same.

Involving the restaurant from the very beginning was good, but taking requirements from just the management was not sufficient. When the team showcased the application to waiters as part of user training, they give feedback which was extremely good and real scenario driven. Hence, the waiters are also important for getting requirements.

As a university project, the team worked on the scope of the application and decided to concentrate only on the feature of order processing. The project team did not realize that in the real world, the system do not work in isolation. By not having a billing feature in the application seriously hindered the usability of the application. If the application also included the billing feature during the experiment, the results would have been even more encouraging.

8.4 Final Recommendations

Overall both the key actors (restaurant and customers) appreciated the implementation of the QR code technology along with using the VPN setup in the restaurant environment. Also the experiment to test the hypothesis of a significant time improvement in the order taking process is also proved. After the implementation of future improvements in the system, manpower can be utilized more effectively. The customers appreciated that there is no wait time to order food/drinks irrespective of the number of customers in the restaurant at any given time. The restaurant liked the idea of utilizing the existing resources more effectively and not losing their customers or their second orders during the busy times thus increasing their business.

List of Figures

List of Tables

References

[1] T. Pyzdek, "The Six Sigma Handbook- Revised and Expanded: A Complete Guide for Green Belts, Black Belts, and Managers at All Levels," McGraw-Hill Education (India) Pvt Limited, 2003.

[2] "Learn Statictics Online," Minitab, [Online]. Available: http://www.minitab.com/en-US/products/minitab/free-trial.aspx. [Accessed November 2012].

[3] U. D. a. M.-C. S. Fabio Casati, "E-business applications for supply chain management: challenges and solutions," *Data Engineering, 2001. Proceedings. 17th International Conference,* pp. 71-78, 2001.

[4] R. Pressman, Software Engineering: A Practitioner's Approach, 2004.

[5] Aaron Skonnard, "A Guide to Designing and Building RESTful Web Services with WCF 3.5," October 2008. [Online]. Available: http://msdn.microsoft.com/en-us/library/dd203052.aspx. [Accessed October 2012].

[6] C. L. John D. Gould, "Designing for usability: key principles and what designers think," 1985.

[7] "Implementing the MVVM Pattern," [Online]. Available: http://msdn.microsoft.com/en-us/library/gg405484(v=pandp.40).aspx. [Accessed January 2013].

[8] "Entity Framework," Microsoft, [Online]. Available: http://msdn.microsoft.com/en-us/data/ef.aspx. [Accessed September 2012].

[9] H. F. Cervone, Applied digital library project management Using Pugh matrix analysis in complex decision-making situations, Hammond, Indiana, USA: Purdue University Calumet, 2009.

[10] "NFC Forum," NFC Forum, [Online]. Available: http://www.nfc-forum.org/home/. [Accessed October 2012].

[11] "QR Code features," [Online]. Available: http://archive.is/20120915/http://www.qrcode.com/en/qrfeature.html. [Accessed September 2012].

[12] "ASP.NET Web API," Microsoft, [Online]. Available: http://www.asp.net/web-api. [Accessed September 2012].

[13] S. Guthrie, "Introducing "Razor" – a new view engine for ASP.NET," July 2010. [Online].

Available: http://weblogs.asp.net/scottgu/archive/2010/07/02/introducing-razor.aspx. [Accessed September 2012].

[14] "ASP.NET MVC," Microsoft, [Online]. Available: http://www.asp.net/mvc/mvc4. [Accessed September 2012].

[15] S. Hanselman, "Asynchronous scalable web applications with real-time persistent long-running connections with SignalR," August 2011. [Online]. Available: http://www.hanselman.com/blog/AsynchronousScalableWebApplicationsWithRealtimePersistentLongrunningConnectionsWithSignalR.aspx. [Accessed October 2012].

[16] "Apache log4net," Apache, [Online]. Available: http://logging.apache.org/log4net/. [Accessed December 2012].

[17] "Features and Tools Overview (SQL Server 2008)," Microsoft, [Online]. Available: http://technet.microsoft.com/en-us/library/bb500397(v=sql.100).aspx. [Accessed September 2012].

[18] "Visual Studio 2012," Microsoft, [Online]. Available: http://www.microsoft.com/visualstudio/eng/products/visual-studio-overview. [Accessed September 2012].

[19] "jQuery Mobile: Touch-Optimized Web Framework for Smartphones & Tablets," jQuery, [Online]. Available: http://jquerymobile.com/. [Accessed October 2012].

[20] "Knockout," [Online]. Available: http://knockoutjs.com/. [Accessed November 2012].

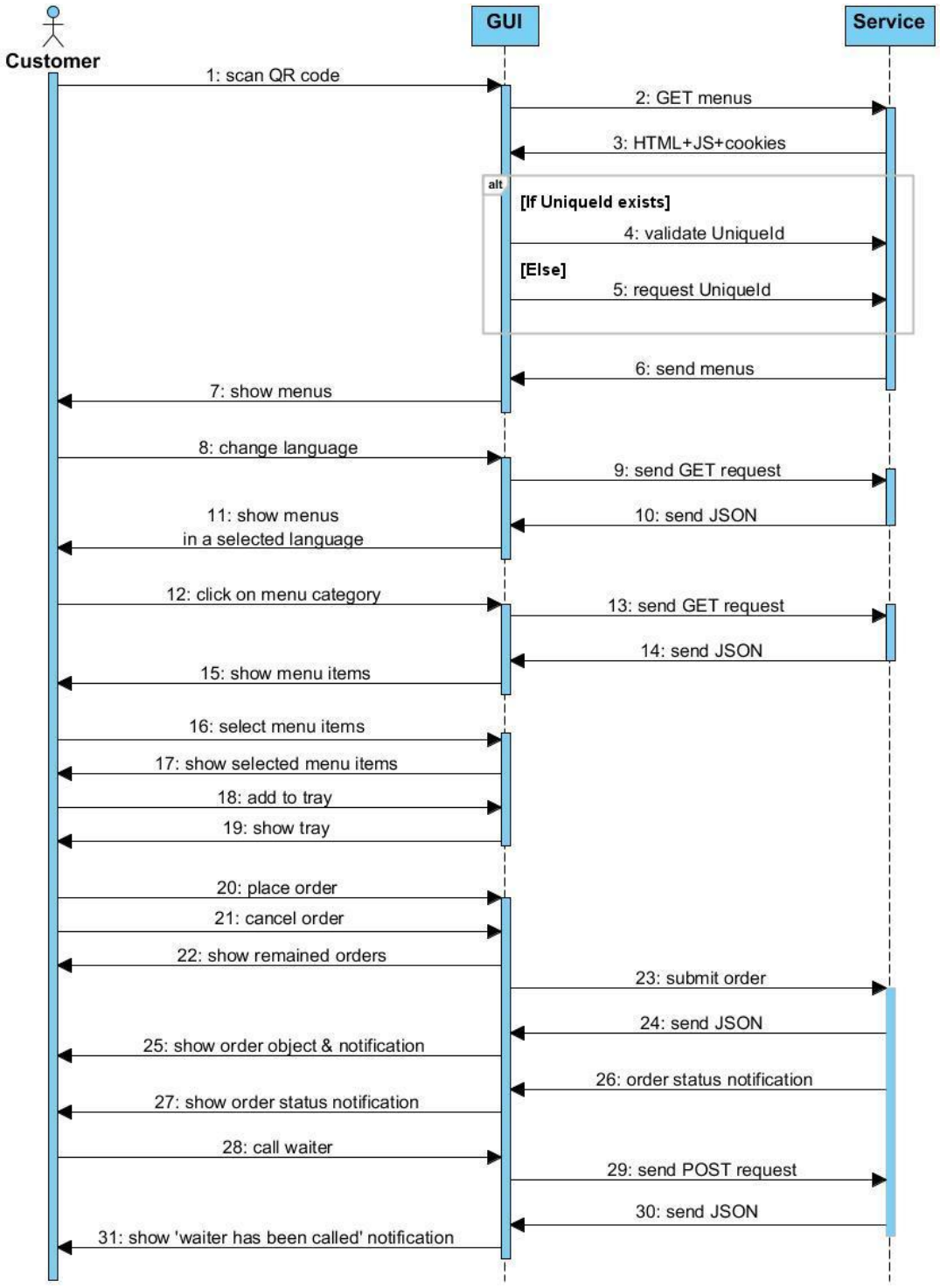
[21] "Plug in to the grid," [Online]. Available: http://gridster.net/. [Accessed January 2013].

[22] "Iscroll," Cubiq, [Online]. Available: http://cubiq.org/iscroll. [Accessed January 2013].

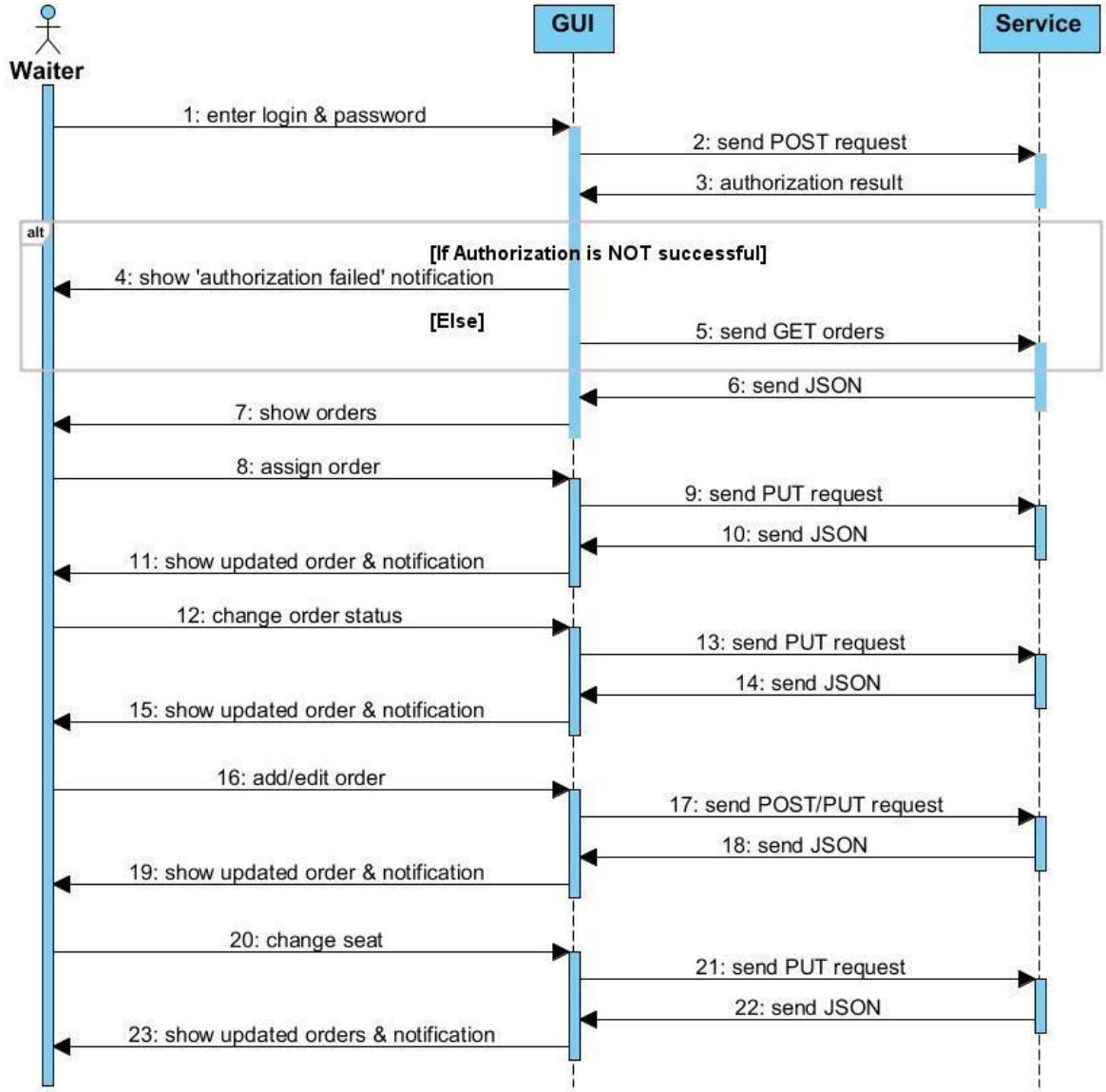
[23] "jquery-cookie," [Online]. Available: https://github.com/carhartl/jquery-cookie. [Accessed November 2012].

Annex A. Sequence Diagram

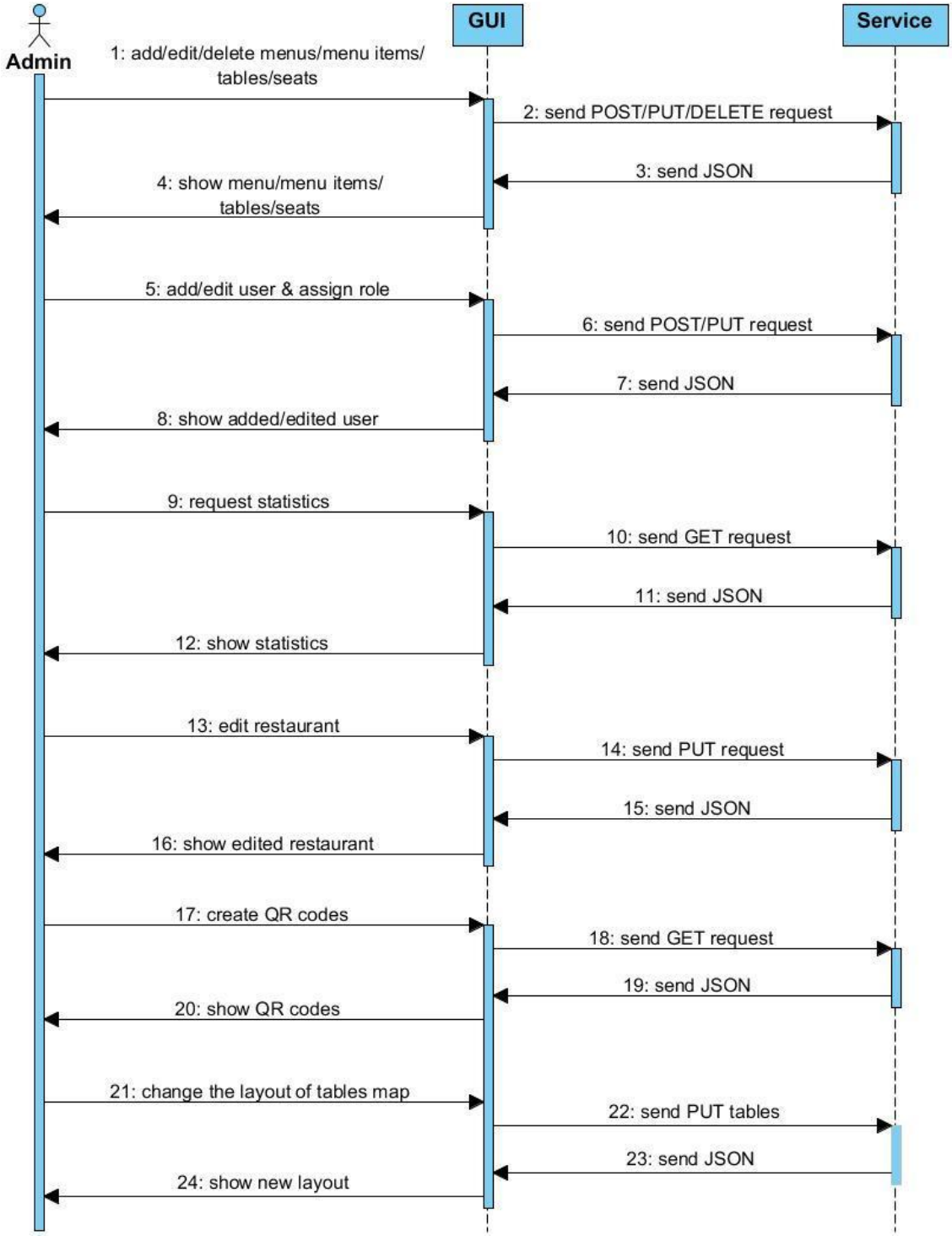
A.1 Sequence Diagram of Customer



A.2 Sequence Diagram of Waiter



A.3 Sequence Diagram of Admin



Annex B. Study Design

B.1 Trial Synopsis

Sponsor	McGee's Irish Pub & Restaurant, Birmensdorferstrasse 83, 8003 Zurich, Phone: 043 810 50 50, Fax : 043 810 50 51, Email: info@mcgees.ch, www.mcgees.ch
Investigators	**Karthick Sundararajan**, University of Zurich Uetlibergstrasse 123, 8045 Zurich karthick.sundararajan@uzh.ch, 076 449 77 70 **Maxat Pernebayev**, University of Zurich Bachlerstrasse 46, 8046 Zürich maxat.pernebayev@uzh.ch, 078 667 98 98 **Tanvi Singh**, University of Zurich Grossplatzstrasse 7A, 8118 Pfaffhausen tanvi.singh@uzh.ch, 0786817123
Title	Evaluation of a Menu Management System for Restaurants
Protocol Version	V4, 27.02.2013
Methodology	This is a monocentric, non-blinded, explorative, arbitrary trial with no control groups
Duration	5 weeks
Center	Mc Gee's Irish Pub & Restaurant, Birmensdorferstrasse 83, 8003 Zurich, Phone: 043 810 50 50, Fax : 043 810 50 51, Email: info@mcgees.ch, www.mcgees.ch
Objectives	To measure the effects on introducing the mobile application on order processing time and customer satisfaction in a restaurant.
Number of Subjects	Customers: 30, Waiters: 3
Inclusion Criteria	Waiters: 1. Waiters who are well trained in using the Waiter Panel Customers: 1. Customers who use the mobile application to order 2. Customers who order only drinks
Exclusion Criteria	Waiters: 1. Waiters who do not work regularly for the restaurant Customers: 1. Customers who order without mobile application and prefer manual ordering way. 2. Customers who order food and drink at the same time.
Schedule	Trial Period: 07.03.2013 - 04.04.2013

B.2 Trial Schedule

Date(s)	Activity		Subject Customers	Subject Waiters	Investigators	Restaurant Management
03.03.2013	Installation and setup				X	X
04.03.2013 and 06.03.2013	Training waiters			X	X	X
07.03.2013 to 04.04.2013 + 05.04.2013 to 12.04.2013	Parallel Activities	The use of the mobile application	X	X		
		Filling in the evaluation form			X	
		Reporting weekly data			X	X
		React to critical bugs/observation				
12.04.2013	Uninstall the trial setup			X	X	X
13.04.2013 to 15.04.2013	Statistical analysis and findings				X	

B.3 Introduction

B.3.1 Background Information

The McGee's Irish Pub & Restaurant, which is located in Zurich, Switzerland, serves its customers with food and drinks. The order taking process is usually done manually, where a waiter has to go to each customer to know when he/she wants to order food and/or drinks or when he/she is ready to pay. This process is not automated. In order to increase the productivity of the restaurant and keep the customer satisfaction unchanged/increased, a mobile application needs to be introduced.

B.3.2 Hypothesis

The mobile application automates the order taking process at restaurants. The automated order processing time is expected to be less than the manual order processing time. This application should not decrease customer satisfaction as they are served faster than the manual process.

B.4 Objectives and Outcomes

After the study, if the order processing time is statistically less than the manual order processing time then it proves the hypothesis. Otherwise, the introduction of the mobile application shows no positive effect on the order taking process of the restaurant. If the average point calculated for customer satisfaction falls in the range of 3-5, it proves the hypothesis. Otherwise, the introduction of mobile application is not a good choice for the restaurant.

B.4.1 Primary Outcome

To assess the effects of introducing mobile application on order processing time at the restaurant.

B.4.2 Secondary Outcome

To assess the effects of introducing mobile application on customer satisfaction at the restaurant.

B.4.3 Primary Variable

B.4.3.1 Order Processing Time

Order processing time is a timespan that is defined as the difference between the time a customer places the order and the time the order is delivered to the customer.

B.4.4 Secondary Variable

B.4.4.1 Average Point

Average point is an average evaluation value of customers' feedback in the range of from 1 to 5.

B.5 Trial Design

The trial is conducted in cooperation with the restaurant for the duration of 4 weeks. The trial population consists of 30 arbitrary customers and 3 waiters of the restaurant that use an application being developed by the investigators. The trial is arbitrary, non-blinded and monocentric. There is no control group in this trial design. The waiters are trained in using the Waiter Panel before the trial starts; but the customers are not trained in using the Menu Panel. During the trial, the data is collected by the application. It is extracted on a weekly basis by the investigators for statistical analysis.

B.6 Selection and Withdrawal of Subjects

Subjects are the arbitrary customers who order the food and/or drinks through the Menu Panel and the waiters who receive the orders from these customers through the Waiter Panel of the application. The trial population consists of 3 waiters and 30 arbitrary customers for the order taking process. At least 10 out of above 30 arbitrary customers give feedback for the evaluation of customer satisfaction.

B.6.1 Subject Inclusion Criteria

Waiter and customer subjects should fulfill the following inclusion criteria.

Waiter subject:

- Waiters who are trained well in using the Waiter Panel to manage the orders and to serve the customers

Customer subject:

- Any arbitrary customers placing the order using the mobile application
- Customers that order only the drink menu (food preparing time in kitchen could influence the order processing time)

B.6.2 Subject Exclusion Criteria

Waiter and customer subjects are excluded if one of the following exclusion criteria is present.

Waiter subject:

- The waiters who do not work at the restaurant regularly, since they are not trained to use the application

Customer subject:

- Customers who order the food menu
- Customers who place orders in traditional way, i.e. using the paper-based menu
- Customers who are placing orders from the bar (Since it is next to the waiter, using the mobile application is not reasonable for reducing the order processing time. As the QR code is not present at the bar, it is not possible using the mobile application there)

B.7 Investigational Products

In order to make the order taking process automated, the mobile application is built. This application consists of three different panels: Menu Panel, Waiter Panel and Admin Panel. These panels are used by customers, waiters and the admin of the restaurant, respectively. Only Menu Panel and Waiter Panel are described below, because Admin Panel is not involved in order taking process.

B.7.1 Menu Panel

The Menu Panel is used by customers. When a customer scans the QR code on the table, the Menu Panel screen is provided. This panel has the following structure.

The top bar of the panel contains the language bar, so customers can decide between English and German as the language of the application and menu. Next to it, there is the notification icon which is highlighted with the number of new notifications when new notification comes from the waiter and it gives access to the Notifications page where all previous notifications can be seen.

The main navigation bar, which is located on the bottom of the panel, contains 5 main navigation buttons (Drink, Food, Brunch, Tray, and Order) which lead to 5 different pages. In terms of the order taking process, these navigation buttons can be grouped into 3 stages:

1) Drink, Food and Brunch navigation buttons allow customers to browse and select drink and food menu items. Then selected menu items are sent to Tray.
2) Tray navigation button is used to navigate to Tray page, where customers can change and manage menu items before the order is placed.
3) Order navigation button allows customers to see the Order page, which displays all placed orders with their details.

The pages Drink, Food and Brunch have the same structure, so they all display the list of different categories of corresponding menu type (The Drink page shows the different categories of drinks, the Food page displays the list of food categories and brunch menu categories are listed on the Brunch page). When a customer clicks on the specific category, the Menu Item Page is opened, listing all menu items with their options for the clicked category. Here, the customers can see the description, photo, video and prices of the menu items and options. While browsing items, customers can select the menu item either by checking the box, which sets the quantity to 1, or opening the drop-down list to select the quantity up to 10. Multiple items can be selected by the customer on the same Menu Item Page. Afterwards, the customer has to click on 'Add to Tray' button in order to add the selected items into the tray. Likewise, the customer can then go to a different categories and repeat the process above to add more menu items to the tray. On the Tray page, the menu items can be deleted/edited before the 'Order' button is clicked. This page acts as a snapshot of all items the customer has selected so far from the Drink, Food and Brunch sections. On this screen, the customer can add free text in the comments box for any additional requests not covered in the menu. As soon as the customer clicks on 'Order' button, the system will show a stop clock for the configurable specific period of time (by default 30 seconds), just in case the customer changes his/her mind and within the given time the order can be cancelled. After this time expires, the order goes to the queue. This is the time that is logged as OrderCreationTime.

The customer receives notification at every stage of the order. Once a waiter assigns the order, the customer receives the notification of "Order Assigned". After the waiter understands the order, it will be changed to "Order Accepted". This means that the waiter has all order items in stock available and is ready to process it. Then depending on whether the order has food items or not, the status changes to "In Kitchen" where it is getting prepared. After the waiter serves the order on the table, he changes the status to "Delivered". This time is logged as OrderDeliveredTime. After the customer enjoys the order, he/she can send the request 'Call Waiter to Pay' to the waiter by clicking the button, which means he/she is ready to make a payment and leave the restaurant. After it, waiter changes the status of the order to 'Closed'.

B.7.2 Waiter Panel

The Waiter Panel is used by a waiter and consists of the following 4 sections: Orders, My Orders, Map and Notifications. The waiter login with his/her username and password to use this panel.

In the Orders section, the waiter can see all the current orders in the restaurants that are not assigned to any waiters yet. Here the waiter can assign the order to himself and then the order will be visible in the My Orders section.

In the My Orders section, the waiter can see the entire list of orders that he/she owns. By going to the details of any order he/she can do the following 3 actions:

- Change the status of the order (Approved, In kitchen, Delivered, Closed)
- Add another item to the order
- Edit or remove the order

Last two activities are required to supersede any requests made by the customers. These features are added just in case manual intervention is needed in certain orders for various reasons.

Under the Map section, the waiter can see the map of the restaurants w.r.t. the tables and chair arrangement. The seats on the table are highlighted with 3 different colors: white (free seat), green (occupied seat), and red (seat which has new orders). Here the waiter can rearrange the seats' positions to enable requests for seating changes requested by customers. This layout also helps the waiter to know the current orders and manage delivery for close customers. By clicking on the seat, the waiter can see the all orders related to that seat and do some activities such as changing status and adding/editing/removing the order items.

The Notifications section is automatically highlighted with yellow color and with the number of new notifications, when new notifications come from the customers. It helps the waiter to find out very quickly if there are new orders or not. By going to this page the waiter can see all notifications from all customers.

B.8 Exact Protocol

B.8.1 Prerequisites

Application needs to be fully developed based on the requirements from the restaurant management, investigators' identifications and the results of general user survey (Survey Monkey). Each service unit of the application has to be tested based on unit testing by the investigators. Afterwards, Menu Panel and Waiter Panel need to be fully tested based on the test cases. Accessing the application requires the VPN connection to the Web Server which is located in the University private network. This ensures that there is no unauthorized access to the mobile application outside the restaurant boundaries. Thus the VPN router should be configured, and the Access Point needs to be installed and configured which allows the customers to connect to the VPN through the Wireless Network. Next, all data of the menu and map have to be entered into the database using

the Admin Panel, so that the data is available to the mobile application. The QR code is printed and pasted in front of each seat of the table to uniquely identify its location. Finally, the instruction card of the mobile application, as shown in Figure 0-1, needs to be printed and placed on each table.

B.8.2 Three Days before the Trial Start

Three waiters of the restaurant are trained to use the Waiter Panel to learn the following key objectives:

- How to access the Waiter Panel and login to it?
- How to receive order notifications from customers and what actions need to be taken?
- How to change the status of an order?
- How to add new menu items/ update menu items?
- How to see current orders and the history of closed orders?
- How to see the orders of a particular seat?
- How to transfer the orders from one seat to another?
- How to receive the “ready to pay” notification?
- How to close orders?

The three waiters in the restaurant are not working at the same time, so the training cannot be provided for all waiters together. Thus, the training sessions are scheduled along with the presence of a restaurant manager for two days. Two investigators starting from Sunday March 3rd conduct it. Since the restaurant is not very busy on Sunday, it is a good time to give the training for waiters. In case any waiter could not attend the training session, he/she can follow up with the restaurant manager.

- On first day, two investigators simulate typical order taking process (one investigator acts as a customer and second acts as a waiter), explaining their actions for each step of the process. During this simulation, the investigators cover all functionalities of both Menu Panel and Waiter Panel. The waiters and restaurant manager are allowed to ask any kind of questions. After that, each waiter repeats what the investigators did in the first run. At the same time, the investigators can ask any kind of questions to make sure that the waiters understood how to use the Waiter Panel.
- On second day, the waiters start using the mobile application in the absence of the investigators in order to be more familiar with it. The waiters note all the questions, if they have any.
- On third day, the investigators answer all questions from the waiters in a practical way.

B.8.3 Trial Start

The trial period starts after the fourth day of the training, i.e. from Thursday March 7 onwards. Beginning of the weekend is the busiest time at the restaurant and the customers can start to use the mobile application to order food and/or drinks. During the

weekend, more customers are expected and it is the more efficient time to collect the trial data. Before the trial starts, all waiters in the restaurant are informed to use the application to collect orders from customers. The trial runs for 28 days, if less than 30 relevant data points are gathered, the trial period will be extended up to 35 days.

B.8.4 Protocol of a Normal Order Taking Process

A customer comes to the restaurant and will find a place to sit. He/she might call the waiter as a regular behavior or can see the small instruction card (shown in Figure 0-1) on the table that introduces the new mobile application.

Figure 0-1: Instruction Card

As a first step, the customer connects to the "McGees" Wi-Fi network from the restaurant and enters the password set by the restaurant. This application is only accessible through this Wi-Fi network, but not through the Internet. Next, the customer needs a Quick Response (QR) code reader application in his/her smartphone. It is assumed that people already have this widely used QR code reader app on their smartphone. If not, the instruction card gives them the names of the most widely used QR code readers to install on their smartphone. After that, the customers can scan the QR code printed in front of his/her seat and click the link to open the mobile application.

The customer chooses the language in which he/she wants to see the menu data. Then menu items are selected and added to the tray. On the Tray page, the customer can review what he/she is going to order and make changes; and then places the order. The order can be cancelled within specific time period (by default 30 seconds). After this time expires, the order is sent to the Waiter Panel. The waiter receives a notification that new order has arrived. When new notification arrives, the Notification section and the seat from which it arrived are highlighted, so that the waiter could notice it very quickly. The waiter sees the details of the order and can change the status. If necessary, the new menu items can be

added into the order, and the order can be edited. In each step of changing the order status, the customer receives the notification. When the customer wants to pay the bill and leave the restaurant, he/she can send the 'ready to pay' notification. These details are gathered by statistic module of the application automatically. Then the waiter comes to the customer to receive the payment. If the customer is willing to fill in the evaluation form, the waiter gives it, which is described, in the next section.

B.8.5 Soft Evaluation Form

An evaluation is planned for customers who have used the mobile application and are willing to do a short survey of five questions. This survey is needed to check the usability design of the application and the common public opinion of the new ordering system. It is not obligatory to fill in the evaluation form thus targeting only ten feedbacks during the trial period.

The questions in the evaluation form are:

1) Ease of installation
2) Ease of usage
3) Did the application save your time
4) Will you use it again?
5) Would you like to see this application in another favorite restaurants of yours? If yes, where?

All questions above have the following options:

a) Very Happy b) Happy c) In between d) Unhappy e) Very unhappy

B.8.6 Other Activities During the Trial

Every morning, the investigators need to check the data collected for the previous day. If data records are quite few, the restaurant manager is requested to provide additional motivations (for example, discounts on some drinks) that might motivate the customers to use the mobile application. Every Sunday, the report of order processing time is generated by the application automatically and then retrieved by the investigators for analysis. The restaurant is busier on Thursday, Friday and Saturday than the rest days of a week. More data points are expected mostly these days and Sunday is more appropriate to report the reasonable amount of data points. IT support including daily checking of VPN router, network connection and server liveness are provided by the investigators.

B.8.7 Trial End

If the 30 data points are collected before the trial period ends formally, the trial is stopped earlier. All the three waiters and restaurant management will be informed about the end of the trial period and they will receive a thank you note. The investigators save the collected data locally from the application for further statistical data analysis described in Section B.10Statistical Analysis.

After the trial period, if the restaurant is interested in using the application, it will be moved from the university's server to the local infrastructure of the restaurant and they can continue to use it without any limitation.

B.8.8 Exact Timing

The trial will take place from the 7th of March until the 4th of April. If at least 30 data points in those 28 days are not collected, the testing period is extended by 7 days more. The evaluation will take place during the same time period with a target of at least 10 completed evaluation forms.

B.9 Adverse Events

Current version of an application not approved by the restaurant management

In case, the restaurant management does not approve the current version to deploy in the restaurant due to critical functionality in the application. In this case, the trial schedule will get delayed until the changes are done and the restaurant management accepts the updated version.

Waiters refuse to participate

If the waiters do not show co-operation or motivation to use the application, then the intervention from the restaurant management will be requested. As the restaurant is involved from the beginning of the project, such an event is highly unlikely.

Customers not ready to use the application

Since there is no selection or control group used as the customers, there could be a situation when very few customers decide to use the mobile application. In that event, the restaurant management will motivate customers to give the new application a try. Even if the new motivations do not cause customers to use the mobile application and 30 data points are not gathered during the trial period, the outcomes of the study is cancelled.

Not enough customer satisfaction evaluation forms collected

In case, customers are not willing to participate in customer satisfaction evaluation or the waiters forget to hand out the forms to the customers, and at least 10 evaluation forms were not collected during the trial period, the second outcome of the study is cancelled.

System downtime

System downtime may happen due to the following reasons:

- Server which hosts Internet Information Services and SQL Server is down
- Internet Information Services is down
- SQL Server is down
- VPN Server is down
- University Network is down
- Internet is down
- Local Network is down

- VPN Router is down
- Access Point is down

In the case the system is down, the trial is stopped until the system is up again by the investigators. The trial continues after the system starts working and will be extended by the period of time, which was spent on fixing the problem.

Major bug in a mobile application

In such an event, the daily business of the restaurant is not disrupted as the manual process is going to take place alongside. The trial is stopped until the investigators fix the bugs. The trial continues after the fix and extended by the period of time, which was spent on fixing the bug.

Waiter Panel tablet broken

If the Waiter Panel tablet is broken, the restaurant management is requested to provide another tablet and the trial is stopped until the tablet for the Waiter Panel is ready. The trial will be extended by the period of time that was spent on restoring the Waiter Panel.

Data lost

In case the data is lost and cannot be restored in the database, and the necessary amount of data are not collected during the trial period, it leads to the cancellation of the study.

B.10 Statistical Analysis

There are two parts of statistical analysis. First, the data collected before and after the trial is analyzed to see if the hypothesis is proven or not. Second, based on the feedback collected from customers, the average point of evaluation form shows whether the customer satisfaction has improved/remained unchanged or declined.

B.10.1 Order Taking Process

DMAIC (Define, Measure, Analyze, Improve, and Control) is a Six Sigma process to show improvement/decline in a product/process that can be measured with respect to critical to quality (CTQ). The sigma value is the statistical variance, which mathematically measures the CTQ.

At the restaurant, the order processing time is the CTQ, which is analyzed statistically using DMAIC. The sigma value of order processing time is calculated. Time less than 7 min for drink orders is treated as non-defect and time more than 7 min is treated as a defect. The Six Sigma DMAIC process is used to make a statistical decision whether the study shows any improvements in the sigma value or not.

B.10.2 Customer Satisfaction

The average point value is calculated after the trial ends. If the value is more than or equal to three, it is treated that a customer is satisfied. In case the value is less than three, it is considered that a customer is not satisfied.

B.10.3 Procedure for Accounting for Order Processing Time and Average Point

1) Order processing time:
2) Historical data is collected to get order processing time before the trial
3) Trial data is collected to get order processing time after the trial
4) Sigma value is calculated for both order processing time and compared

Average point:

The customer satisfaction is evaluated based on the questions in the evaluation form shown in Section B.8.50. It is calculated based on the average of five point scale.

1) The evaluation form is distributed by the waiters to the customers who used the mobile application and are willing to fill in the form
2) Each question of the evaluation form has five options, and each option has a value from 1 to 5. The average point is calculated by summing up all scored values and dividing them by the number of questions and number of evaluation forms

B.10.4 Collection of Data

The following section describes the process of collecting valid data before and after the trial.

B.10.4.1 Historical Data

The historical data was collected by the investigators manually being present in the restaurant. It was collected from early-November 2012 to mid-January 2013. Each dataset contains an order processing time, which was calculated from the time when the customers had a seat in the restaurant to the time when the order was delivered to them. The manual data collection happened for 7 days during different time and days of the week.

There were many challenges in this process as described below:

- The number of customers at the restaurant were quite less because of bad weather conditions.
- During the winter season, the big open air seating area which is away from the bar was closed and hence collecting relevant data was a big challenge.
- During the Christmas season, many customers were on vacations. Data collection activity suffered in December.

B.10.4.2 Trial Data

After the deployment of the mobile application in the restaurant, the trial data of order processing time is automatically collected by the application.

B.10.4.3 Efficiency of data collection

The trial data collected by the application is stored in the database without any manual intervention. While retrieving the trial data, the order processing time is automatically calculated and analyzed manually by the investigators.

B.10.4.4 Ensuring Data Accuracy

To ensure the accuracy of the reported data, the investigators log into the Admin Panel and go to the statistics page to get the orders for a given time frame. Here order id, order submit time, order delivery time, duration, menu type (drink, food, breakfast, lunch) and waiter name are displayed. Here the investigators can filter the records to match the inclusion and exclusion criteria as described in Section B.6.

Data accuracy will be tested every day by the investigators. Any variations or discrepancy in the collected data is analyzed by the investigators. The data accuracy of the order status change made by the waiter will be checked by the investigators. If the data was not correct, it needs to be sent to the restaurant management in order to ensure better logging practices from the waiter in the future.

B.11 Data Handling and Keeping

Both investigators and the restaurant manager can access the trial data through the Admin Panel. After the trial is finished, all data will be kept by the investigators.

Annex C. Customer Feedback

Evaluation form						
Data points	Ease of installation	Ease of usage	Did the application save your time	Will you use it again?	Would you like to see this app in another favorite restaurant?	Comments
Form 1	5	5	4	5	4	
Form 2	5	5	4	4	5	McDonalds, Subway
Form 3	5	4	5	5	5	UI can be fine tuned
Form 4	5	3	2	5	3	Need a repeat order button
Form 5	2	2		4	3	Good start, needs improvement of the UI
Form 6	5	5	3	4	5	Everywhere
Form 7	3	4	5	5	5	Any pub or bar
Form 8	5	5	5	5	5	Mensa
Form 9	5	4	5	5	5	
Form 10	5	4	5	5	5	Pur Pur
Average	4.5	4.1	4.2	4.7	4.5	
Average Point	4.4					

Annex D. Content of CD

Each copy of this thesis contains a CD-ROM with the following content:

D.1 Master Project

- MasterProject.pdf
  A copy of the written project.
- Abstract.pdf
  The abstract of the project.
- Zusfsg.pdf
  The German version of the abstract of the thesis.

D.2 Source Code

- OMLSQL.zip
  SQL file for the database with master data
- OMS.zip
  Project source file

D.3 User Survey

- The survey monkey extract of individual responses.

D.4 Trial Data Points

- The data points for the experiment collected before and after deployment.

D.5 Evaluation Form

- Screenshots of feedback forms as collected by the customer.

D.6 Application

- Screenshots of Customer View.
- Screenshots of Waiter View.