Mobile Restaurant – An advanced food ordering service via handheld devices

By

Nguyen Kong Minh

submitted in partial fulfillment of
the requirements for the
Bachelor of Technology (Hons)
(Information and Communication Technology)

DECEMBER 2006

Universiti Teknologi PETRONAS
Bandar Seri Iskandar,
31750 Tronoh,
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

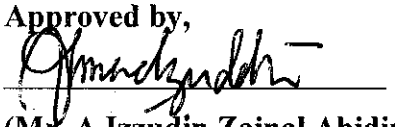
Mobile Restaurant – An advanced food ordering service via handheld devices

By

Nguyen Kong Minh

A project report submitted to the
Information and Communication Technology Program
Universiti Teknologi PETRONAS
In partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (HONS)
(INFORMATION AND COMMUNICATION TECHNOLOGY)

Approved by,



(Mr. A Izzudin Zainal Abidin)

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

DECEMBER 2006

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

NGUYEN KONG MINH

ACKNOWLEDGEMENT

With the help and support from many people while working on this project, I would like to relish this opportunity to express my gratitude to all who has in one way or another helped me to complete this work.

First of all, I would like to thank Mr. Admad Izuddin Zainal Abidin, the project supervisor who has been giving many supportive and constructive advices. His suggestions have guided me throughout this project.

I also gratefully acknowledge all my friends who have given invaluable review and feedbacks on the project. Special thanks go to Nghiem Anh Tuan for his time discussing about .NET as well as programming debug.

Last, and most of all, I am thankful for the endless encouragement and care from my Dad, Mum and beloved brothers. They have given me the strength, motivation and determination in all my undertakings.

TABLE OF CONTENTS

LIST OF FIGURES

MOBILE RESTAURANT

ABSTRACT

Handheld devices provide a means for customers to place and pay for orders on-the-fly in nowadays m-commerce generation. Therefore, the aim of this project is to develop a system that has capability in handling the food ordering service in Universiti Teknologi Petronas (UTP) with the use of handheld devices. Currently, the food services in campus are not efficient enough in which customers usually have to wait long time for the food to be prepared when they enter the cafeteria. In addition, customers do not have chance to know in advance the type of food will be available until they physically reach the restaurant. A system targeting on mobile devices is developed to overcome the limitations of the current services – Mobile Restaurant. In chapter 3, author will discuss about the methodology used to develop the system, which is waterfall model consists of five phases: requirement definition, system & software design, implementation & unit testing, integration & system testing, and operation & maintenance. The system prototype consists of three sub systems, which are client application implemented in mobile devices, a SMS notification service, and the web-based service administration. Chapter 4 will introduce the volume of work has been done and discuss on some issues for the implementation of the system. The first two sub systems had been completed and tested. The third sub system is under development for future enhancement. As in conclusion, the scope of the project has been successfully achieved and it provides a sophisticated system to enhance the restaurant's production and food on time delivery process.

CHAPTER 1
INTRODUCTION

1.1 BACKGROUND OF STUDY

M-commerce (mobile commerce) is the buying and selling of goods and services through wireless handheld devices such as cellular telephone and personal digital assistants (PDAs). Known as next-generation e-commerce, m-commerce enables users to access the Internet without needing to find a place to plug in. [1]

M-Restaurant is an example of m-commerce concept. The industry affected by this m-commerce principle is service/retail, in which consumers are given the ability to place and pay for orders on-the-fly. M-Restaurant provides a web service for mobile users to order and pay the food in a restaurant in advance. The M-Restaurant actors are categorized to customers and employees.

M-Restaurant service has been designed to enhance a restaurant's production and delivery process. The service provides the restaurant with a means of adjusting the sales with production capacity and prior knowledge of upcoming orders. As content delivery over wireless devices becomes faster, more secure, and scalable, the restaurant's staff will be informed the most updated orders as soon as the customers made a reservation through their mobile devices. The restaurant within the student's campus is identified to be the tested area of the service once completion.

1.2 PROBLEM STATEMENT

The survey has proved that customers (student/normal customer...) usually have to wait a couple of time for the food to be prepared when they enter restaurant/cafeteria to order and have their meal. In addition, customers don't have the chance to know in advance what food will be available on the day until they physically reach the restaurant.

1.3 OBJECTIVE OF THE PROJECT

The objective of the project is to minimize the time-consuming process of ordering the meal. One solution has been identified to come out with an advanced mobile booking service which is being designed to enhance a restaurant's production and food 'on time' delivery process. A customer can save time by ordering and paying for a meal beforehand from a restaurant within or nearby the campus via a mobile device. Mobile device is determined as a communication tool to the service to make use of its availability at any time and any place.

1.4 SCOPE OF STUDY

First of all in this project, the author has to learn about the process of restaurant management and food ordering system. Critical thinking on the process of system handling customer's order, system response and role of restaurant's employee has been analyzed based on experience. After it is compared with the actual situation of an existing restaurant that would be available in the Internet, the architecture will be identified to form the initiation of the project development. Then the author will develop a prototype called M-Restaurant to present the new paradigm of food ordering system in the local area, particularly in the student campus. Microsoft Visual Studio .NET programming package is used as development tools. In the final implementation of the prototype there are 4 major elements, which are Database Server, Web Server, SMS service and the last, but not the least is the handheld device.

CHAPTER 2
LITERATURE REVIEW

2.1 REVIEW ON MOBILE APPLICATION DEVELOPMENT

Characteristics of Mobile Devices and Applications [9]

In the book written by Brad Leupen [9], the professional discussed on one of the most important features of any application designed to run on a mobile device is the look and feel of the user interface. When designing user interfaces for mobile applications one has to be aware that mobile devices are radically different from desktop PC's. The primary differences are the smaller size of the screen and the way in which the user inputs information. Keyboards are not normally provided on mobile devices and users generally use some sort of stylus or mini-keypad to provide input. Interfaces, which require the user to enter large amounts of data, are therefore not suited to mobile device applications.

In addition to the characteristics mentioned above, mobile devices are also limited in the processing power and resources which they have available. When developing applications for these devices it is important to release the resources held by applications as soon as possible, particularly if the resources are scarce. This includes memory, which is often at a premium on mobile devices. Although mobile devices are limited in their computational power and resources, users still expect their applications to deliver a responsive, interactive experience. As a result developers often find themselves facing a tension between resource/memory usage and application performance when developing applications that are both efficient and meet the user's expectations.

Web Service [9]

Technologies such as Java RMI and .NET Remoting allow an object on one machine to interact with, and call methods of, an object on a different machine. This is a powerful technology, which provides a mechanism for developing some great-distributed

applications. However the biggest drawback of these remote invocation technologies is that they are proprietary standards; interoperation and cross-platform support is not possible.

Web services are an RPC-based remote interaction technology. In contrast to other remote interaction technologies (e.g. Java RMI, .NET Remoting), Web services are based on HTTP, XML and open standards. As such, they can be used from a variety of platforms and enable cross-platform mobile and distributed applications to be built. Web services also offer enchanced security over other remote interaction technologies and provide an easy way to integrate and expose existing legacy systems.

A Web service is simply an object, which makes some of its methods available for remote invocation. This object is hosted and managed by a Web server and all interaction with the object takes place through the passing of textual XML messages. As with other remote invocation technologies it is possible to pass arguments to and return values from Web services. Both system and user defined types may be used as arguments and return values but Web services are more restrictive than other technologies in terms of how those types are used. For example, when passing an object as a parameter to a remote call, .NET remoting serializes all the members of the object and can automatically move any required code files (assemblies) to the remote machine. Web services are incapable of moving code1 and can only serialize the public data members of a class. Other key differences between .NET remoting and Web services are concerned with the creation of proxies. Remoting creates proxies dynamically at runtime whereas the proxies for Web services are created statically at compile time. If the interface to a Web service changes, all clients that depend on the service must be recompiled to work with the new service. This can lead to confusing versioning and configuration issues if not properly managed.

Web services are an extremely important technology for mobile application developers. Most mobile devices now provide Internet access and XML Web services are the only remote interaction technology available in the .NET Compact Framework. Other

technologies such as .NET Remoting are not supported; hence Web services are they only sensible choice for consuming distributed services from a mobile device.

2.2 PREVIOUS IMPLEMENTATION OF MOBILE RESTAURANT

SmartRestaurant

In [5], the implementation of SmartRestaurant is introduced by a MediaTeam of University of Oulu, Finland. It is a service for mobile users to order and pay their lunch beforehand. The distributed architecture is described as below:

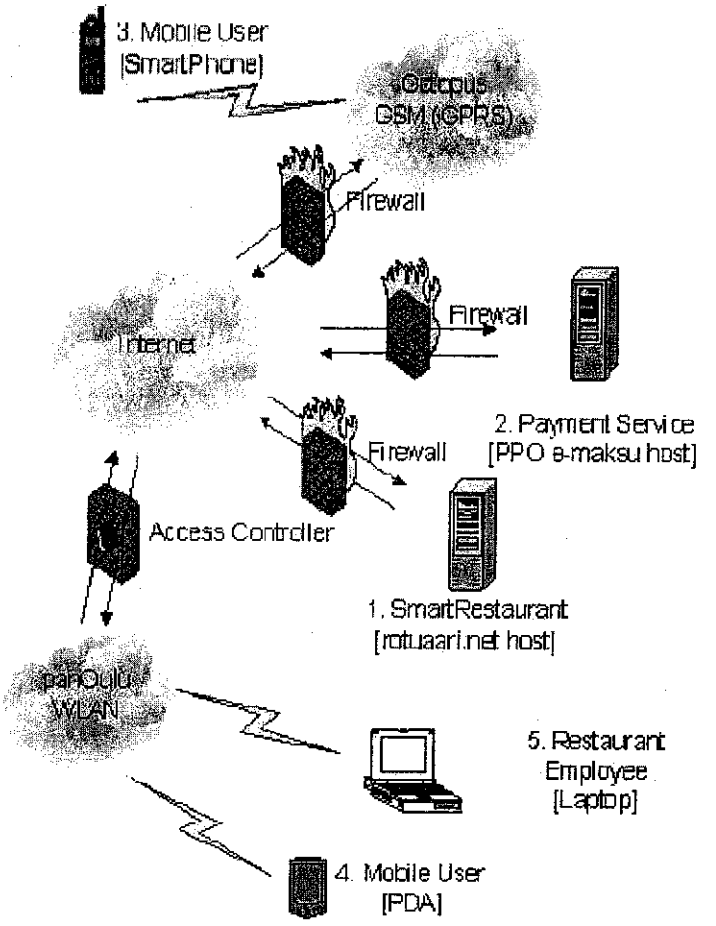


Figure 1: **SmartRestaurant's Distributed Architecture.**

- SmartRestaurant web service is installed on a host, which is connected to the Internet via a firewall.

- The payment service is installed on a host, which is connected to the Internet via a firewall. The system is used with the e-maksu payment service hosted by PPO, a local operator. The e-maksu service provides customers with accounts on which money is first transferred from a regular bank account. The money on the customer's e-maksu account can then be used for paying for content and services of service providers having subscribed to the e-maksu payment service. The payments are aggregated to the providers' e-maksu accounts and then transferred to their bank accounts.
- Mobile users can use the SmartRestaurant with smartphones, which provide means for Internet access, e.g. GPRS (General Packet Radio Service) connectivity, and a web browser capable of showing XHTML pages.
- Mobile users can use the SmartRestaurant also with a PDA (Personal Digital Assistant), which features wireless connectivity and a XHTML capable web browser.
- Restaurant employees operate the SmartRestaurant service with a laptop equipped with WLAN connectivity and a standard web browser. Laptops with wireless connectivity can be placed freely at the restaurant premises and they also allow the employees be mobile.

Implementation tools & techniques

Components Requirement:

- Multi-processor host that runs the SmartRestaurant service.
- Apache Web Server.
- Tomcat Application Server.
- Database Server.

Tools & Techniques:

- Java Server Page (JSP)
- XHTML
- HTTP over SSL (Secure Socket Layer) protocol (HTTPS)
- Java Database Connectivity (JDBC) to connect MySQL Database

CHAPTER 3

METHODOLOGY

3.1 PROCEDURE IDENTIFICATION

The "waterfall model" is chosen as the methodology of this project. The phases of waterfall model are illustrated in Figure:

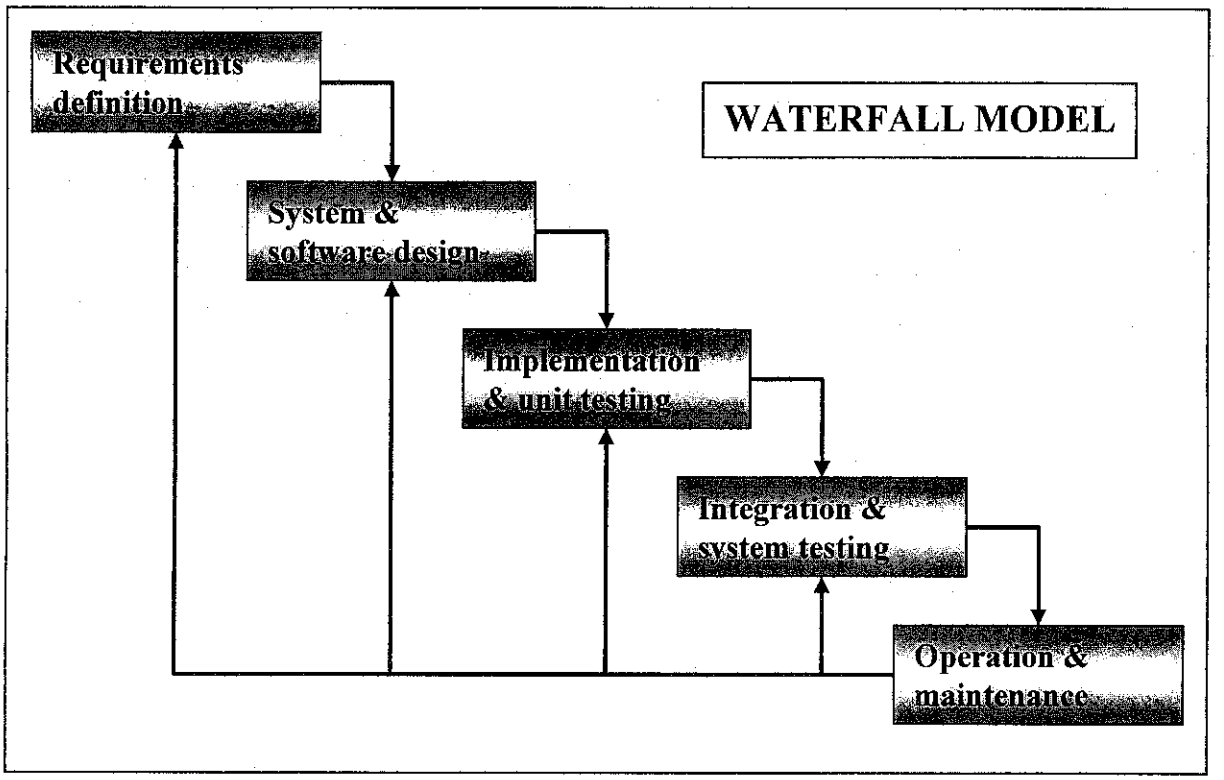


Figure 2: **Waterfall model**

The waterfall approach emphasizes a structured progression between defined phases. Each phase consists of a definite set of activities and deliverables that must be accomplished before the following phase can begin. The first phase tries to capture *What* the system will do (its requirements), the second determines *How* it will be designed, in the middle is the actual programming, the fourth phase is the full system *Testing,* and the final phase is focused on *Implementation* tasks such as go-live, training, and documentation. [13]. The details of each of the phases are discussed below:

3.1.1 Requirements analysis and definition

It is important that the requirements of the project be clearly analyzed and defined in detail so that they can serve as project specification. This phase involves with the identification of the background of the study and the definition of the objective and scope of the project.

At the initial stage, I have to gather all information from the restaurant and the customers (mostly inside UTP) to determine what the problem and disadvantages that they are facing. In addition, an analysis of the current restaurant meal providing process was done to find out all characteristics that can be developed and from these information, step by step, we find out the solution to improve the performance.

From this specification, a literature review is conducted to identify the available data classification methods and suitable techniques and technology to tackle the identified requirements. Requirements specification also helps in the design and planning of the software development.

The project will be carried on with some researches from the existing implementation of the same concept (SmartRestaurant by MediaTeam of University of Oulu, Finland [5]). The data is gathered, tools and techniques are identified together with analysis from the current environment (constraints, limitation...) to form the method in order to implement the project.

3.1.2 System and software design

The system design process normally partitions the requirements to either hardware or software systems. It establishes overall system architecture. Software design involves identifying and describing the fundamental software system abstractions and their relationships. [11]

Based on the detailed requirements obtained from the first phase of this project, the software architectural design task is performed, which includes identifying the data flow, the class diagram of this application and also a test plan. The data flow diagram depicts at

high level the flow of data and the processes performed in the application. The class diagram is used to illustrate the objects, their properties and methods and the relationships between different objects. The data flow diagram and the class diagram together serve as the solid framework from which the implementation task is based upon. Typically, projects using waterfall methodology would result in 20%-40% of the time budget allocated for the first two phases of the development. This considerable amount of time spent on assembling, documenting the requirements and software design results in better segregation of work into modules and also in better control of the modules in later phase. The detailed design using this waterfall approach also makes it easier to ensure that the modules will be easier to integrate when the project nears the end of the implementation phase.

System Requirements

The online mobile restaurant system will allow users to view the items of food sold by the restaurant. Customers will be able to browse the MobileRestaurant's menu with a mobile device, order items from the menu and arrange for them to be delivered at a convenient time relative to their current context (time, location) so that the food will be hot and fresh when they enter the restaurant.

Customers will be able to view images of any food item and read reviews of food items written by other customers. In addition a simple star rating scheme will be provided. Customers will be able to give a star rating to any food item and to view the average rating given by other customers. Customers will also be able to add their own reviews of any food item sold by the restaurant.

Once a customer has placed an order, they should be presented with a list of available delivery times from which they should select the time at which they would like their order be ready at the restaurant. The system should allow the user to store commonly used data, such as a current customer's location and time of order, on the server for later reuse. Some mechanism for authenticating customers with the server will be required before this information can be retrieved. The system should be able to send notification to

customers regarding confirmation of order and promotion or value information to valid membership of the service.

Distributed Architecture

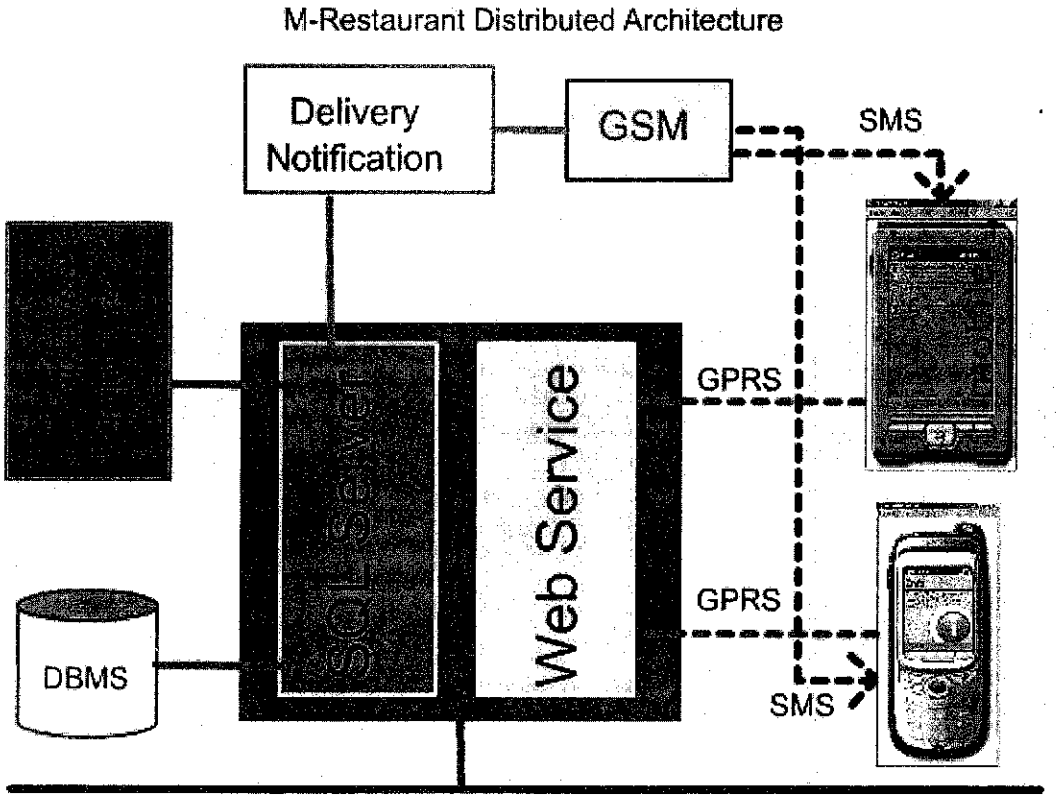


Figure 3: **M-Restaurant Distributed Architecture.**

3.1.3 Implementation and unit testing

Based on the architectural design framework from the previous phase, the implementation task is performed. The whole application is partitioned into a set of program units or modules. Unit testing involves verifying that each unit/module meets its specification. Typically, 30%-40% of project's time budget is allocated to this phase.

The project contains the web service for mobile users that is developed using .NET framework. Customers will place an order and it will be recorded in database server and

sent to the staff's computer. Restaurant staff will be aware of the coming order and they are also able to update the menu and any related information to the service. The system is also planed to provide confirmation to customers through mobile SMS.

3.1.4 Integration and system testing

The individual program units or programs are integrated and tested as a complete system to ensure that the software requirements have been met. After testing, the software system is delivered to the customer [11]. This is the phase whereby different modules developed and tested in this project are integrated into a complete application. An overall testing on the application is performed to ensure that the integrated modules work according to the specification. At this stage, it may lead to detection of incompatibility between developed modules and modifications are necessary to ensure that the end product is error-free.

3.1.5 Operation and maintenance

Normally this is the longest life-cycle phase. The application is installed and put into practical use. Maintenance involves correcting errors which were not discovered in earlier stages of the life cycle, improving the implementation of the system units and enhancing the system's functionalities as new requirements are discovered. The documentation of the project is one important deliverable in this phase.

In summary, with waterfall methodology, projects tasks are divided into phases; typically the previous phase's deliverables have to be completed before the next phase can be started. However, there are also iterations at the end of the development process that allows for modifications on the previous phase's deliverables to ensure compliance to the software specifications. The reason for choosing the waterfall model in this project is because of its structured processes with focus on a detailed se of requirements and detailed design of the whole application that would make the development and verification tasks easier. This approach also allows for a good estimation of the time allocated for each of the major phase of the project. The linear nature of this approach requires that the deliverables of the previous phase be completed and tested before the

next phase can be carried out, which reduces the amount of rework and debugging when all the modules are integrated.

3.2 TOOLS REQUIRED

The following are the tools used in the development of the project:

Hardware/Components	Specification/ Model	Reason of Usage
Desktop/Notebook	NEC Versa Notebook or high performance PC	The workstation is needed to develop the system prototype. It is recommended to be a high processing computer to handle the mobile development prototype efficiently.
Database Server	SQL Server 2000	To implement the database structure of the prototype. Database server is used to separate database from application interface to improve the security and better data management issues of the system.
Web Server	Internet Information Services (IIS)	To implement the Web service of the system. The Web service is registered in the IIS Web server that acts as the "middle man" among the handheld devices and SQL Server.
GSM Modem (optional)	Siemens TC35i	GSM Modem is used in project to send notification of the successful order to customer. Further, value information may also be sent to registered customers via the means of GSM connected to the

		customers' mobile service center. Siemens TC35i Modem is compatible, lightweight design and low power consumption. It supports a wide range of applications and reliable method for controlling the terminal and application.
Wireless Router	D-Link 802.11g/2.4GHz	Wireless router will be used alternately for the purpose of testing of prototype. It enables the communication between the handheld device and the Web service (which is firstly built locally)
Pocket PC Device	Any pocket PC with phone call feature.	To implement the prototype with the real handheld device. The pocket PC is required to have the feature of sending or receiving SMS for service notification purpose. In addition, pocket PC should be able to connect to Internet to use the Web service.
Mobile Phone	Any mobile phone with GPRS enabled.	To implement the prototype in the form of normal mobile phone. The phone is required to have GPRS enabled in order to browse the Web service.

Software	Reason of Usage
Microsoft Windows XP Professional	Most stable Windows series with performance and security enhancements

	through service pack.
Microsoft Visual Studio .NET 2003 package: • C#. NET • VB.NET • ASP.NET Web Services • Microsoft Pocket PC 2002 Emulator • Microsoft SmartPhone 2003 Emulator	Visual Studio .NET provides a wide range of tools that offer many benefits for developer. It helps to create dynamic mobile application, and especially provides the built-in emulator (Figure) that is convenient for mobile application testing and real device deployment. In addition, it compacts the environment for developing the web service that will interact with the mobile application.

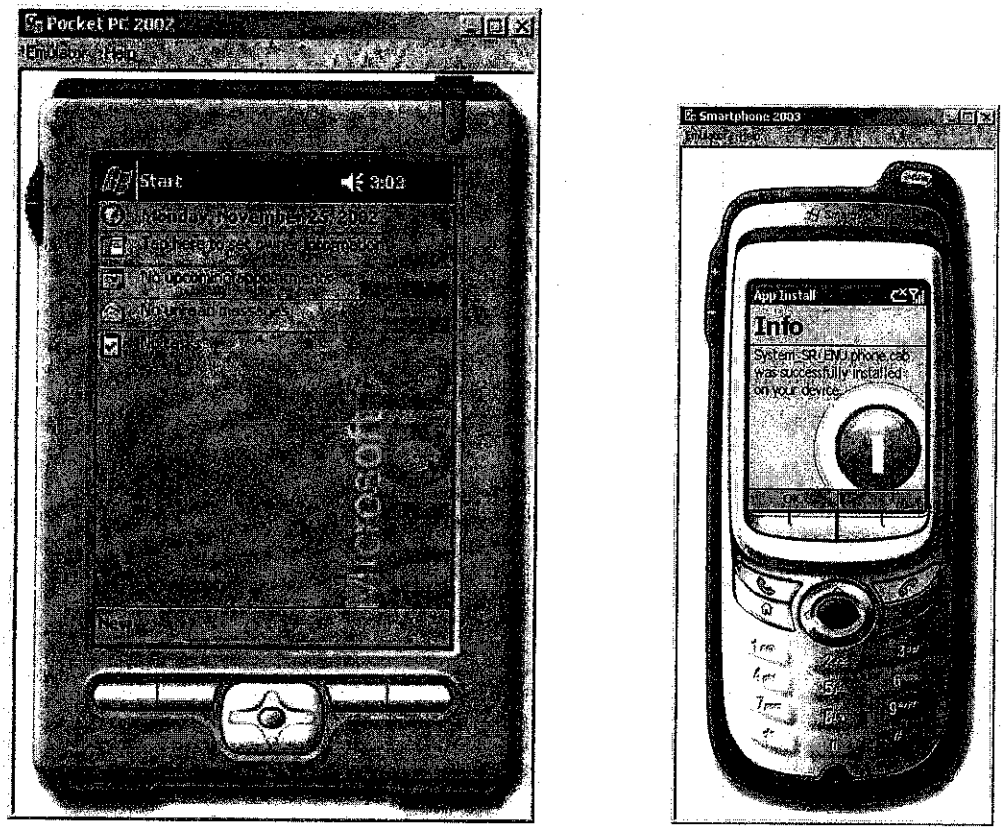


Figure 4: **Microsoft PocketPC 2002 and SmartPhone 2003 Emulators.**

CHAPTER 4
RESULTS AND DISCUSSIONS

4.1 WORK DONE

In this project, author has solved the main problem of restaurant order process and come up with the design of a whole system. However, due to the scope and also the time constraint, author only focuses on the Client application side, the Database and Delivery Notification. The administration web sites are being constructed with 20% of completion at the moment.

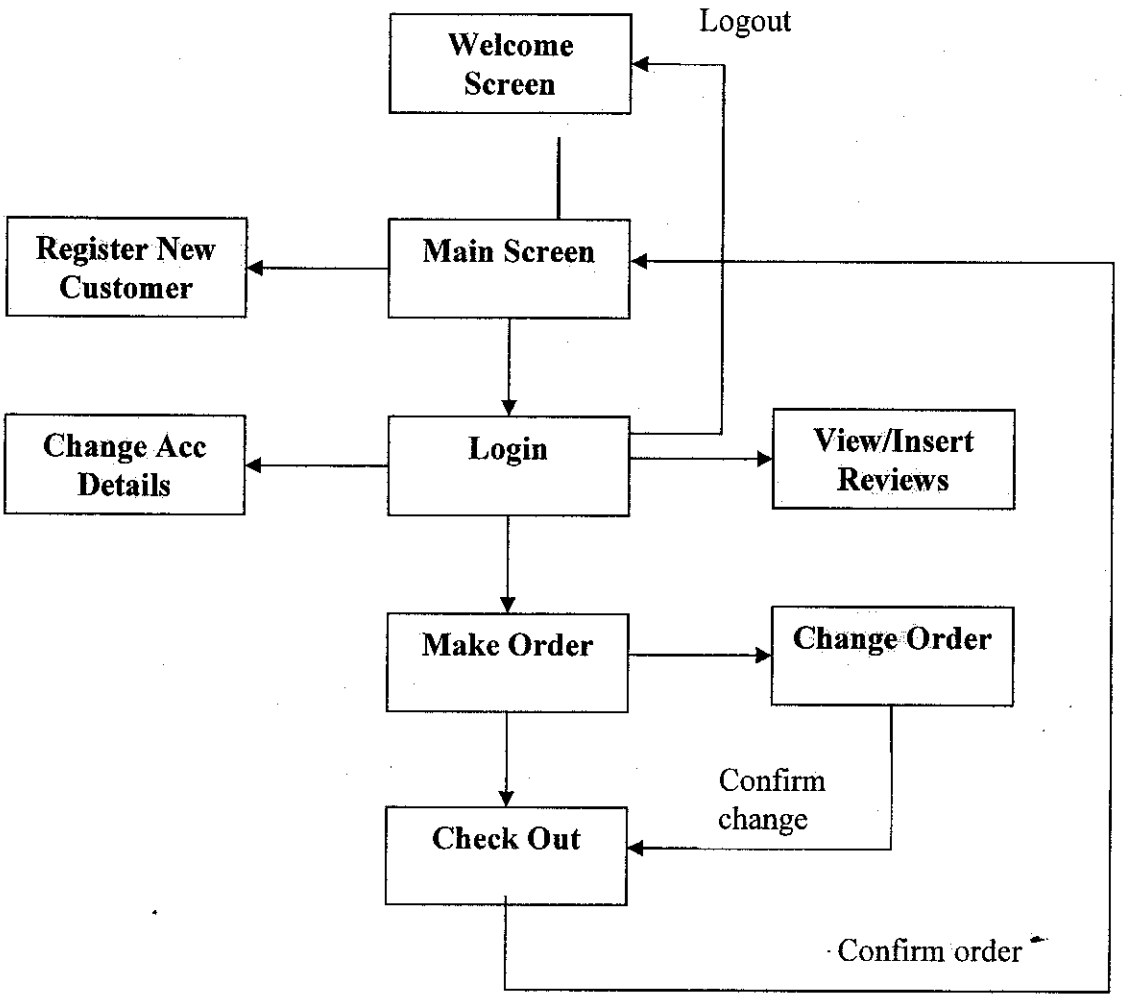


Figure 5: **Client User Interface Design.**

The underlined architecture has been developed and tested successfully. The system was constructed in the network environment that consists of: a virtual Web server that will run the service, a database server takes responsible for data transaction, and a virtual development environment for PocketPC and Mobile Phone to run the system interface. This determined technique is to create a simulated environment that in the end it will assist the implementation of the service in real situation. The practice of separating the database and application interface into application server and database server is taking advantage of improving data security and well defined system structure for better system maintenance and improvement. In fact, a security and authentication mechanism called SSL encryption (Secure Socket Layer protocol) is built in coding to protect the server connection.

A database of food categories, food name and description as well as the price was built up and connected successfully to the system.

The Client side application, which is running on the wireless devices, has been developed successfully. It provides users the interface to navigate all the function that the system offer such as see the detail menu, choose the food, and make order, etc.

The application has been tested successfully in the Microsoft Pocket PC Emulator environment as well as the real PocketPC (DELL Axim X50). In both environments, the application runs smoothly and stably without any delay or hangs.

However, in the real environment, we need to consider the coverage of wireless network, the strength of the signal as well as the traffic of the network. All of them can affect the performance of the system.

System Flow

The first entry form that will be appeared as the welcome screen in the handheld device. A picture of the restaurant will be displayed with a welcome note.

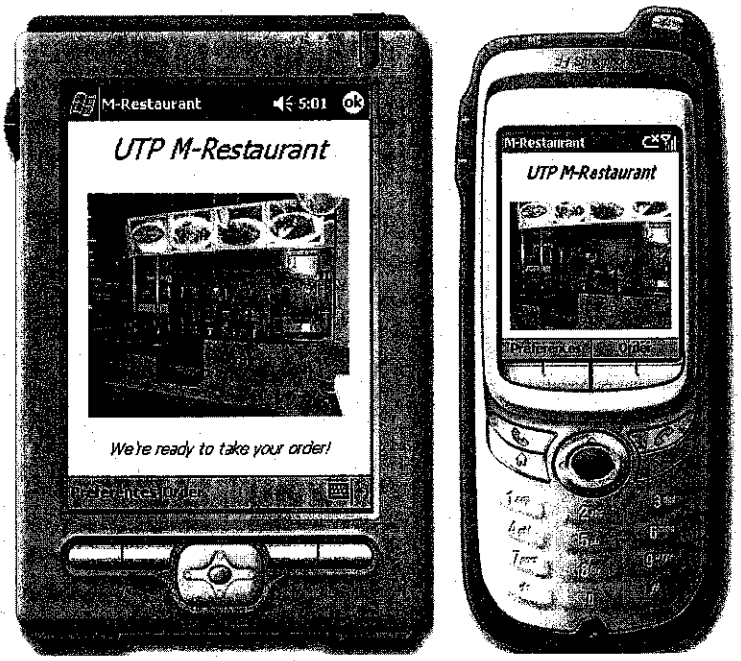


Figure 6: Welcome Screen

The next step for user to enter the main form will be a click event on the Order command on the navigation that is visible to users. Main form of food ordering will be displayed in which users can browse through the food type/category and view the food items in each category. The images of the food items are provided to attract customer on the food selection. In addition, the food description will give some comment on the food ingredient and how it is served, so that customers will make a better choice for their favourite food.

Figure 7: Main Screen

Customers need to login in order to make an order,. The idea is that each customer should have an account that contains all the necessary information so that when they place order; the details of customer will be retrieved and stored in the database itself without providing much information during each time of ordering. Customers will only need to provide the selection of the food and the reserved time, the rest of information are already attached with their account.

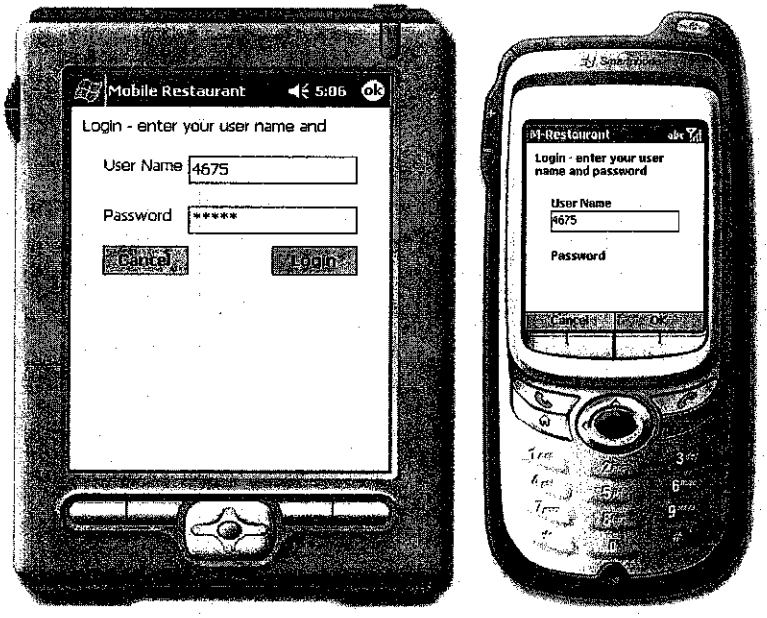


Figure 8: Login Screen

A form for user login will be displayed when users trigger the Login button in the main form. Customers will need to key in their userid and password in order to login to the application. With new customers, they have a form to register for their account under the navigation in the main form.

Another feature is the food review. Registered customers will be able to provide feedback on the food they had tried. With each review, customers could give the rating from 1 to 10 on how delicious the food is. Other customers may find this review useful for them to know the feedback from previous customers on a particular food item.

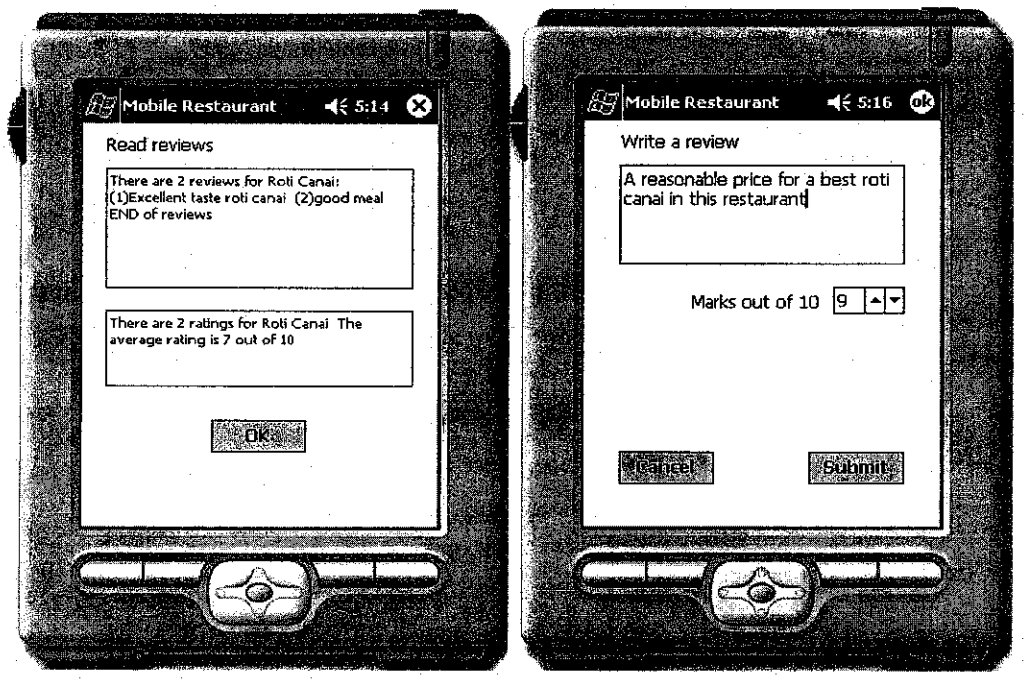


Figure 9: Food Review Screen

Once customers have successfully login with their account, they will able to place an order. Customers will decide the food to be ordered and the number of food item. When they click on the Order button, the system will calculate the subtotal of the ordered food item. Customers will be able to order another food item, the prices for all the food items will be accumulated and displayed on the corner list. Furthermore, they still have the choice to change the selection anytime before checking out by clicking the Change button next to list.

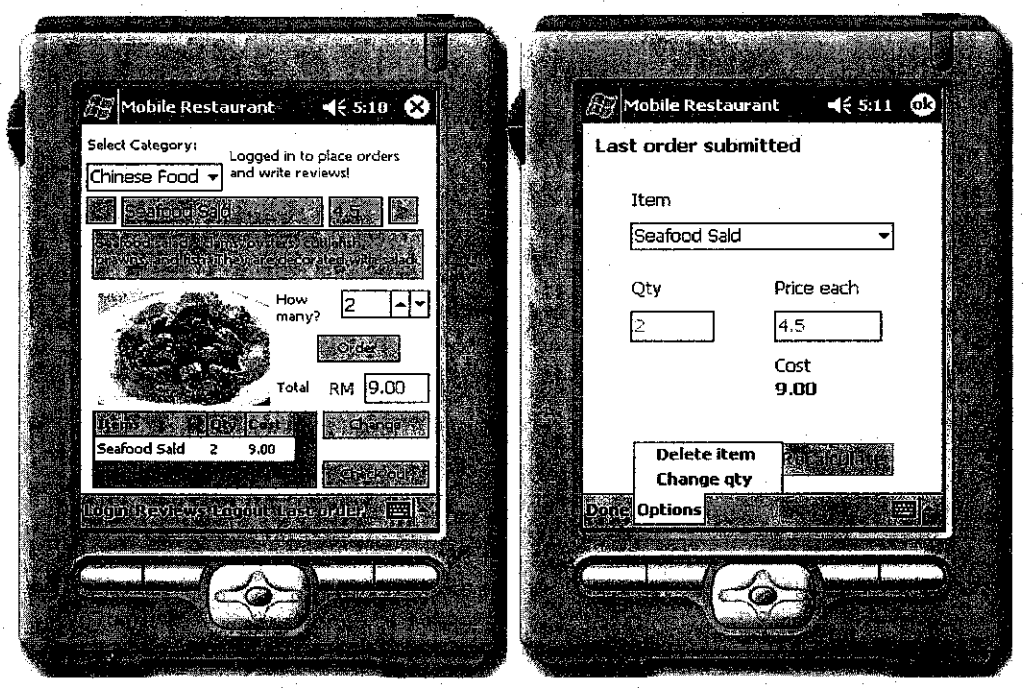


Figure 10: Change food ordered screen

When customers confirm on their order, they will need to check out the order by clicking the Check Out button. A checking out form will be displayed that requires customers to provide the reservation time. If they wish the food to be delivered, the customers will need to confirm on their delivery address too. Customers will be able to update their location in this form as well by clicking the Change button.

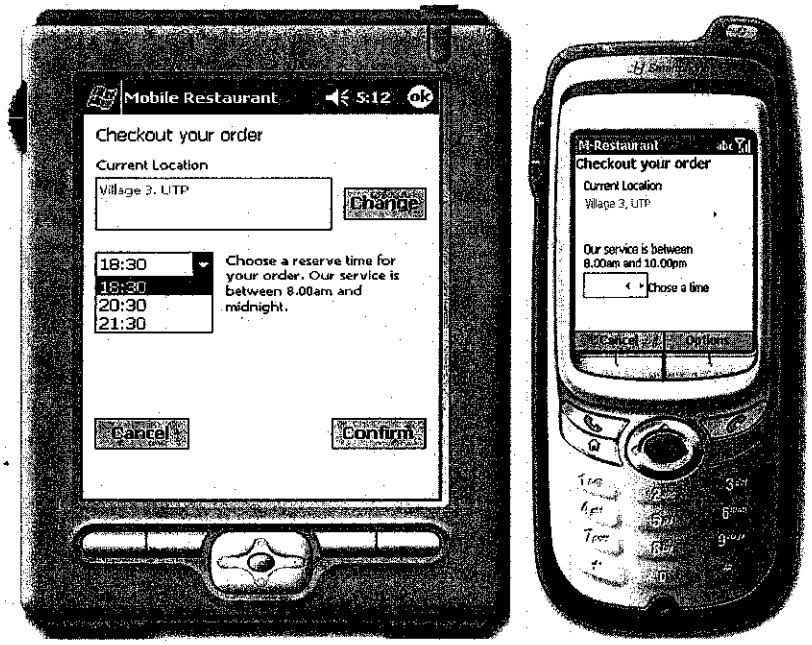


Figure 11: Make order

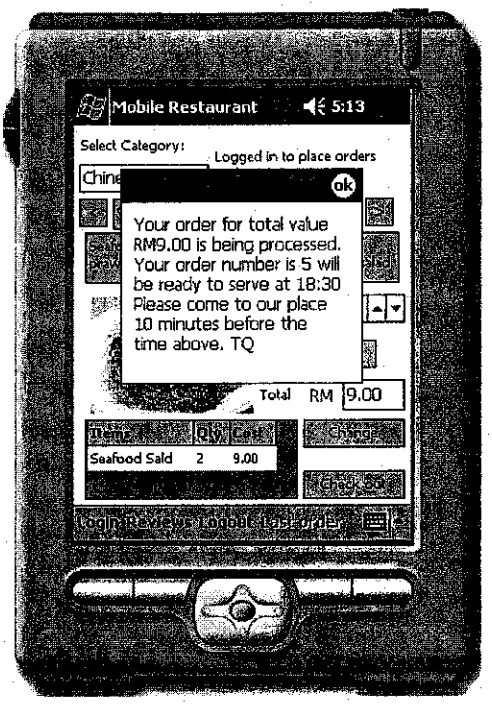


Figure 12: Service reply

When the ordering process has been completed, a confirmation message will be displayed to customer's handheld devices. Before the reservation time, the restaurant may send a SMS notification to customers to inform their order. This is done as the second part of the project. The objective of notification service is to send information to customers in term of order's notification and valuable information through SMS. The method of notifying customers with SMS is by making use of a public SMS service. The chosen service is ConnectionSoftware located in UK. An interface has been designed and coded with setting parameters provided by the service. The interface will connect and interact with the service to route the SMS to the specified customer's cell phone number. A charge of 0.50 RM will be counted for each SMS notification and accumulated to the meal's price. Customer will have option to receive or refuse the notification SMS and the price of the meal will be adjustable to +/- 0.5 RM.

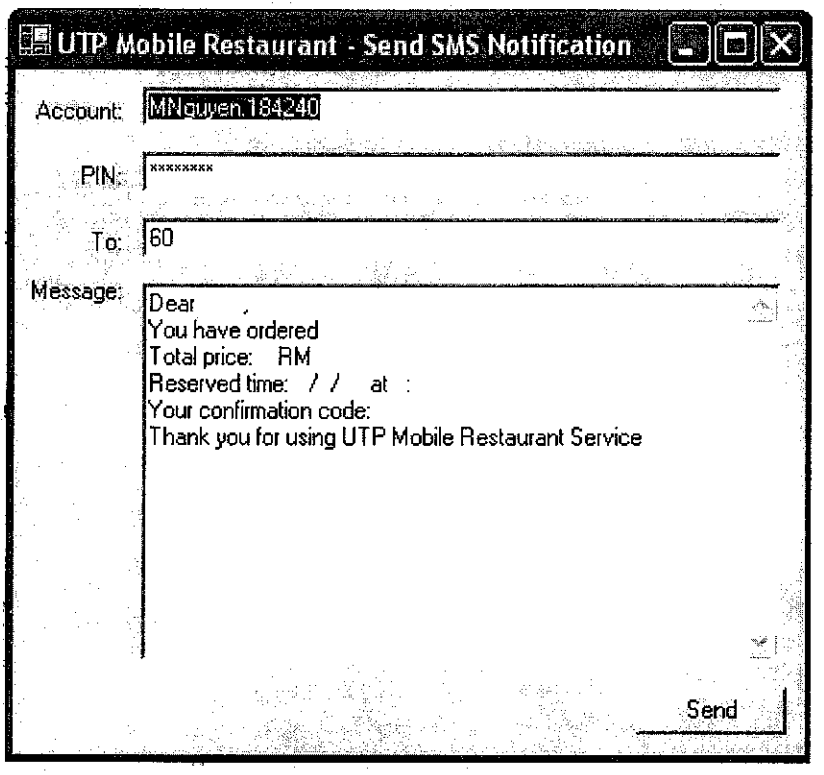


Figure 13: SMS Notification

4.2 FUTURE ENHANCEMENT

The M-Restaurant system will be continuously enhanced in future work. The applications for customers to order the meal through PocketPC and Mobile Phones have been developed with complete set of functionalities as specification. However, there are 2 issues that are identified to take into consideration and embed into the customer application interface. Since the restaurant service considers to serve customers in 2 ways: food delivery at customer's place or food serve at restaurant, an interface for asking customers the choice for the food to be served will be created. In case customers request the food to be delivered at their place, it is necessary for the system to have a payment method (called micro-payment) in order to let customers pay online. There are several methods can be identified as visa/credit payment, phone bill payment or account payment... Among these methods, account payment is commonly implemented in real situation in Europe in which a third-party organization will take responsibility for money processing in e-commerce transactions. The activity of these third-party services is such

that they provide customers with accounts on which money is first transferred from a regular bank account. The money on the customer's account can then be used for paying for content and services of service providers having subscribed to the third-party payment service. However, due to the lack of such kind of third-party service in the current situation, it creates a constraint in the implementation of the system.

In term of food being served at the restaurant, the need of implementing the payment gateway is not necessary. The idea is that customers will be able to pay when they physically reach the restaurant instead of prior payment as the previous food delivery. Customers will simply "book" for their food in advance and pay later so that when they enter the restaurant, the food is ready while hot and fresh. This scenario currently implements the system.

There is also one main part will be developed in order to provide a complete prototype of M-Restaurant service in future enhancement. It is a Web-based interface service for restaurant staff to manage the service's order and its activities.

CHAPTER 5
CONCLUSION

5.1 CONCLUSION

This report introduced the background of M-Restaurant service, a web service for ordering food items in the mobile environment. Several researches on mobile characteristics and web service have been conducted to assist author in the first stage of developing and implementing the mobile service in the local area. The case study on SmartRestaurant has been analyzed to provide the understanding of an existing implementation of the mobile restaurant concept.

The report also discusses on some initialization analysis of the project whereby the system architecture is draftly sketched, as well as the required tools and potential techniques were identified to create the prototype. By the end of time frame, the objective of completing two sub systems is successfully achieved, which will provide client mobile application to order food beforehand. The last sub system will be continuously developed to provide a complete system prototype. With all efforts, author does believe that the system will be a success.

SUMMARY

For this project author have developed M-restaurant system. It will be used to order any available meal, from a particular restaurant for any time client wants.

This first prototype release will ease client's work, as client will not need to wait in a long queue any more, or wait for the meal to be prepared, and it will also solve the problem of lack of information about the meals, providing even pictures.

Author had following tasks and activities done while accomplishing this project: Information Gathering and Analysis, Design a Software Architecture, Software Development and Testing.

Here are some use-cases that system will do or provide:

- View the meal details
- Read and write customers' reviews
- Order the meals beforehand with mobile devices
- Set a time and check orders
- Administration Tool to manage M-Restaurant Business (Web-based) – future enhancement.
- Send notification through SMS

The tools have been using throughout the development phase are: Microsoft Emulator 2002, Microsoft Smart Phone 2003 Emulator, and Microsoft .NET (ASP.NET, VB.NET, C#. NET). Author would like to emphasize on the easiness of Microsoft .NET, which includes several useful languages, and provides with a wide programming liberty.

REFERENCE

[1] Mobile Commerce in automobile industry – Date: 18 March 2006

http://ieeexplore.ieee.org/iel5/8532/26947/01197540.pdf

[2] Mobility from end to end for small and midsize businesses – Date: 20 March 2006

www.hp.com/sbso/wireless/**mobility_end_to_end**.pdf

[3] M-Commerce – Date: 18 March 2006

http://www.mobileinfo.com/Mcommerce/index.htm

[4] Visual Studio .NET - Date: 20 March 2006

http://msdn.microsoft.com/vsstudio

[5] SmartRestaurant Case Study – Date: 18 March 2006

http://www.mediateam.oulu.fi/publication/pdf/532.pdf

[6] Creating .NET Web Services - Date: 21 March 2006

http://www.oreilly.com/catalog/prognetws/chapter/ch02.html

[7] The evolution of Web Services and .NET 2.0 – Date: 19 June 2006

http://www.c-sharpcorner.com/Article/frmDisplayArticles.aspx?SectionID=1&SubSectionID=133

[8] Microsoft ASP.NET Professional Projects

Book written by Hersh Bhasin

[9] XML Programming

Book written by Brad Leupen

[10] ASP.NET Unleased

Book written by Stephen Walther

[11] Software Engineering, 6th Edition, Addidon Wesley, 2001

Book written by Sommerville I.

[12] System Analysis and Design Methods, 5th Edition, McGrawHill, 2001

Book written by Witten, J.L. et al

[13] Development Methodologies Compared: Why different projects require different development methodologies

Book written by Marks, D.