CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

Cambridge Advanced Learners Dictionary described automation as a process in an office or factory that is operated by computers in order to reduce the amount of work done by humans and the time taken to do the work.

The advancement in Information and Communications Technologies (ICT) has greatly influenced many aspects of human endeavour particularly in the area of business transactions. Study has it that in the earlier days, food industries traditionally lagged behind other industries in adopting ICT (Bhandge *et al*., 2015). However, the improvement in computing technology and applications development has heightened expectations of consumers and in turn forced the industry to consider automation for enhancing customer's loyalty and retention. Today, automation in the food retail industry has also been enhanced further by the advancement in wireless technology and the emergence of mobile devices (Tanpure *et al*., 2013).

The major challenge in the six (6) restaurants of ABUAD is not in realizing efficiencies as a result of dependence on manual ordering and payment system. The food ordering system is entirely manual which involved servers, receipts and queues. The servers had to note down orders from customers via voice, collect them, serves and collate at the end of the day. Even though this system is simple, it may involve human errors in noting down the orders.

Visiting a typical ABUAD restaurant involves selecting a meal by converting cash to ticket, and waiting endlessly for your turn to get your orders. If the restaurant is busy, the customer is left on the queue longer than they need leaving other customers waiting and any unnecessary waiting causes customer dissatisfaction. There are many reasons leading to the feeling of

dissatisfaction including being entertained late in terms of order taking by the waiters and meals serving. These are just a few of user experiences and observation which demands urgent attention and automation may be the way out of these.

The advantages of this system include:

- To eliminate the food ordering system at ABUAD restaurant.
- To make payment process easier and comfortable.
- To reduce labour among the workers.
- To make services faster and more efficient.
- To ensure that all bill calculations and accounting are accurate.

1.2 Statement of the Problem

The number of restaurants in Nigerian Universities is on the increase, leading to improved orders and exchange of money. However, the six (6) restaurants in ABUAD that services over 6000 students and almost 1000 staff still operate in the traditional (manual) means of food ordering and payment. The result of the above have presented inherent problems experienced with the manual system of ordering such as; converting cash to ticket before food can be served, queuing on the food counter for a long period of time and also in the accounting department that needs to be addressed. Above all, payment is done before food is ordered. Hence, the need for a Restaurant Automated Food Ordering System (RAFOS) for ABUAD is to solve these problems.

1.3 Aim and Objectives of the Study

The main aim of this project is to design and implement a Restaurant Automated Food Ordering System (RAFOS).

The objectives of this work are:

a) To automate food ordering system at ABUAD Restaurant that can eliminate or at least minimize the current problems experienced in traditional (manual) system.

b) To automate the payment process of ordered food at the restaurant.

1.4 Methodology

A standard procedure of system design known as Software Analysis and Design Methodology (SADM) would be employed for the design process of the Restaurant Automated Food Ordering System (RAFOS).

The hardware devices used in this project are: Tablets, Android Devices, Smartphones, Servers, PCs, Central System.

The system is written in Programming, Scripting and Query languages like JAVA, XML, ASP.NET, C#, VB.NET, SOAP, and MsSQL DATABASE.

1.5 Scope of the Study

This project was based on the six (6) restaurants in ABUAD namely: The first cafeteria, the second cafeteria, the third cafeteria, the talent discovery centre (TDC) cafeteria, the cafeteria in the teaching hospital and lastly K-nourish cafeteria. However, this project was implemented in only one of them which is the second cafeteria.

The implemented system based on design functionalities is divided into 3 packages:

Menu application tablet: This provides users an interface to interact with the system.

Central system services: This system is responsible for receiving information about users' food orders.

Administrator/Manager services: Monitors and controls the system. Also provides users with tag/ receipts to keep them informed when order is ready.

This system intends to:

1. Describe the key components of the ABUAD Food Ordering System.

2. Discuss the role of individuals (Users) and management (Administrator, Manager, Restaurant Staff), as well as their relationship with one another in the restaurant.

1.6 Functionality of the System

A. Users' request for food order

The implemented Restaurant Automated Food Ordering System (RAFOS) allows users to request for food via an android application. The application is a mobile-based application that has a simple GUI.

B. Transmitting data from central system to the server

The central system receives all orders and payments from customers and transmits them to the server via the university's internet or Wi-Fi.

C. Sending data from the server and the database

All information on order and payments from the customer are transmitted to the database for storage purposes. Hence, the server and the database are connected.

D. Accessing data through the central system

The system has an administrator that has full authorities to access all parts of the system. The administrator monitors and controls the system. They are able to retrieve any data from the system regarding any customer. The manager is set up for accessing statistical / accounting information of the restaurant

1.7 Contributions to Knowledge

1. A Restaurant Automated Food Ordering System (RAFOS) was developed in this project.

2. The implementation of this project was done and is believed to greatly improve the food ordering process in ABUAD thereby providing them with efficient and effective means of ordering.

3. This system will automate payment process also by introducing the use of credit cards which makes payment a lot easier.

4. The use of this system will cause a reduction of labour from the restaurant staff and relieve customers of stress because of the prompt activities and processes.

5. This system should provide better food ordering process, management and accounting system in ABUAD

1.8 Definition of Terms

Automation: "… automatically controlled operation of an apparatus, process, or system by mechanical or electronic devices that take the place of human labour (Merriam Webster, 1828).

Restaurant: A place which prepares and serves meals to customer in exchange for money.

ABUAD – Afe babalola University Ado-ekiti

GUI – Graphical User Interface

Android application – An application that can work on android devices e.g. Tablets, android phones etc.

Mobile application – An application that can work on mobile devices e.g. Mobile phones, smartphones etc.

Management: A non-profit organisation coordinating the efforts of its employees to accomplish its objectives through the application of available resources.

RAFOS – RESTAURANT AUTOMATED FOOD ORDERING SYSTEM

AFOS – ABUAD FOOD ORDERING SYSTEM

1.9 Arrangement of the Project

Chapter Two contains the Literature review which is a High level survey of research materials covered.

Chapter Three is the Research methodology which includes conceptual diagrams of the model.

Chapter Four is the Core analysis of the research which presents the model developed to accomplish the stated objectives. It includes the implementation and result, testing with raw fact, hardware and software requirements.

Finally, Chapter Five is the Research findings which summarizes feedback from subject matter experts through descriptive and inferential analysis, recommendation and conclusion and describes areas for future research.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter represents the review of some relevant literature on Restaurant Automated Food Ordering System (RAFOS). It encompasses the work of the previous scholars - the motivation of the work the objectives, limitations that the proposed work will address. It also discusses the technology used in the implemented work.

Russell Ackoff (1981) defines a "system as a set of two or more elements that satisfies the following condition: The behaviour of each element has an effect on the behaviour of the whole, the behaviour of the elements and their effect on the whole are interdependent, however subgroups of elements are formed, each has an effect on the behaviour of the whole and none has an independent effect on it". Gerald Weinberg (2007) also defines a "system as a way of looking at the world" Russell Ackoff (1981) categorised the various types of systems. They are:

(a) Natural systems are usually perceived as hard systems and are related to the human body, ecosystems, weather, water cycle, etc;

(b) Symbol Systems are soft systems and refer to languages, set of icons etc;

(c) Human Activity Systems are businesses, organizations, markets, clubs etc;

(d) Information Systems are special case of designed systems that includes the representation of the current state of some human activity system. Example MIS, banking systems, databases etc;

(e) Automated Systems are systems that are able to control a process with minimal human assistance or without manual and have the ability to initiate, adjust, action show or measures

the variables in the process and stop the process in order to obtain the desired output. (Noorropidah 2012).

(f) Food Ordering System: Food ordering is a process of ordering food from a local restaurant or food cooperative through a web page or app. Much like ordering consumer goods online, many of these allow customers to keep accounts with them in order to make frequent ordering convenient. This system is designed to overcome the problems in using a manual system (Zakaria *et al.*, 2010). Independent online food ordering companies are known to offer three solutions. One is a software service whereby restaurants purchase database and account management software from the company and manage the online ordering themselves. The second solution is a Web-based service whereby restaurants sign contracts with an online food ordering website that may handle orders from many restaurants in a regional or national area. The third is where an independent create and offer foods, meals or kits via their website which are then directly sent to consumers. One difference between the systems is how the online menu is created and later updated. Managed services do this via phone or email, while unmanaged services require the customer to do it. Some websites use wizards to find the best-suited menu for the customer (Zakaria *et al.*, 2010).

Customers are turning more towards online food services options for the convenience its offers, the variety of options and affordable food choices However, this project focused mainly on Automation and Food Ordering System.

2.2 RESTAURANTS

2.2.1 What is a Restaurant?

A Restaurant is an establishment substantially engaged in the business of preparing and serving meals. To qualify as a restaurant, an establishment's gross receipts from food and non-alcoholic beverages shall not be less than thirty percent (30%) of the total gross receipts from food, non-alcoholic beverages and alcoholic beverages. A restaurant shall also have a kitchen and an inside dining area with seating for at least 36 people. Restaurants are located all over the world. Restaurants are one of the favourite premises with no regard to the actual reasons for visiting restaurants (Patel *et al.*, 2015). There are restaurants in Japan, India, Malaysia, Nigeria, to mention but a few. Let us take a look at some of the restaurants and how they function.

2.2.2 Restaurants in Asia

Tokyo is a Japanese restaurant chain in the Philippines. It is known for serving Japanese dishes with free rice refill, a promotion called "eat all you kanin" (or eat all the rice you can) which has been going since it started in 1985. Fortunately, Tokyo currently operates 56 stores in the Philippines with much of the chain expansion being attributed to franchising at first. Generally, Tokyo restaurants are located within major shopping centres and malls in the country, malls which are owned and operated by SM Prime Holdings, Robinsons and the Ayala Corporation (Tokyo restaurant, studymode.com). Furthermore, businesses in hospitality industry based on food services have grown significantly in India. In the last few years where many restaurants have opted for offering various types of menus for the customers, this has increased the competition in the hospitality industry. The restaurants have to provide the best services and maintain relationships with their customer in order to survive in this competition (Bora *et al.*, 2012).

2.2.3 Restaurants in Europe

Restaurants in Europe have most of the best restaurants in the world. Not merely the best in Europe but also the best restaurant in the world, El Celler de Can Roca is run by the ground-breaking culinary innovators, the Roca brothers in Girona, Catalonia. Opened originally in 1986, in recent years El Celler de Can Roca has never been far away from the top spot. They have also acquired three Michelin stars for their modern interpretations of Catalan cuisine. Most people enjoy European dishes so they have people from different parts of the world coming over to have a taste and to also see how they operate (Leah Hyslop, 2015).

2.2.4 Restaurants in America

Defining American food is more difficult than playing baseball with a ball made of good ol'-fashioned Jell-O. This is a time when people are lining up around the block for fast casual burgers and clamouring for fine dining tasting menu simultaneously. (Reisner Abby, 2016). In America, they believe that a great meal at an outstanding restaurant can be a life-changing experience. But with new places popping up every day, it can be difficult to decide which spots are worth your time and cash. Everest, one of the best restaurants in America which holds one Michelin star, guests can choose between the seven-course degustation menu, the seven-course vegetarian menu, or three- and four-course prix-fixe menus. Each selection treats guests to elegant French dishes, like magret duck served with marinated turnips, or slow-braised wild sea bass accompanied by artichokes and leeks. Most good American restaurants operate this way with more five-course menu per customer.

2.2.5 Restaurants in Africa

In South Africa, they operate on both African and intercontinental dishes. But they love more of their African dishes for non-citizens to have a good taste of. They have the idea that trying

some delicious traditional African food should be part of every visitor's itinerary. A number of specialised restaurants in South Africa do an excellent job of serving both modern and traditional African food. Each dish reflects one or more of the different cultural influences found across the continent. Traditional African food is generally cooked over an open fire or in a three-legged pot (or potjie), so meat tends to be served in either stewed or grilled form. A starch usually accompanies the meat: mieliepap (maize porridge), potatoes or rice. Beetroot, carrots, cabbage and pumpkin are the vegetables most commonly served. Typical South African dishes include tripe, morogo, chakalaka, amadumbe, and the ubiquitous boerewors roll. Tripe is a traditional treat favoured by most Africans. In the Cape it is considered a regional delicacy and is often served lightly curried with small new potatoes and fried onions (South African cuisine, 2017).

2.2.6 Restaurants in Nigeria

There are some really big restaurants in Nigeria which a lot of high class dishes where people from different parts of the world come to explore. But however, there are still some restaurants that still operate manually in some rural parts of the country. In Nigeria, our traditional dishes are derived from the various ethnic groups. Each tribe has their own special local dishes but because of how globalised the world has become, these dishes can be found in different restaurants all over the world. Although Nigeria is still a developing country, we are still sound in so many areas like our restaurants. There are over a thousand automated restaurants in Nigeria functioning efficiently and effectively.

2.2.7 Restaurants in ABUAD (Case study)

There are six (6) restaurants in this university and all of them operate manually. The process of cash conversion is by the use of ticket, queuing on the food counter for a long period of time and the like. Over six thousand (6000) students and about a thousand (1000) staff visit these

restaurants every day and ordering food can be very hectic for both the students, staff, and also the restaurant staff. However, this project is implemented to eradicate the problems stated above.

2.3 REVIEW OF RELEVANT WORKS ON FOOD ORDERING SYSTEMS

2.3.1 A Customizable Wireless Food Ordering System with Real-Time Customer Feedback (CWOS-RTF) (Samsudin *et al.*, 2011).

This project talks about the existence of wireless technology and the emergence of mobile devices. They enable a simple yet powerful infrastructure for business application.

This project was motivated by the usage of smart phone technology in healthcare and other applications and presented a development of smart phone technology in a business application, namely food ordering system to be used in restaurants. Figure 2.1 shows the system architecture of CWOS-RTF, which cover three main areas of the restaurant: the serving area, the restaurant owner's working desk (cashier table), and the kitchen. Conceptually, the CWOS-RTF were built on four main components: The mobile application on the smart phones for customers to make order, the web-based .application and server on the laptop for restaurant owner to keep track and respond to received customers' orders, and customize menu information, the database for restaurant owner to store order details, and updated menu information, the wireless infrastructure to support networked communications (Samsudin *et al.*, 2011). CWOS-RTF demonstrated the feasibility of using wireless communication and smart phone technology to meet food ordering system specifications.

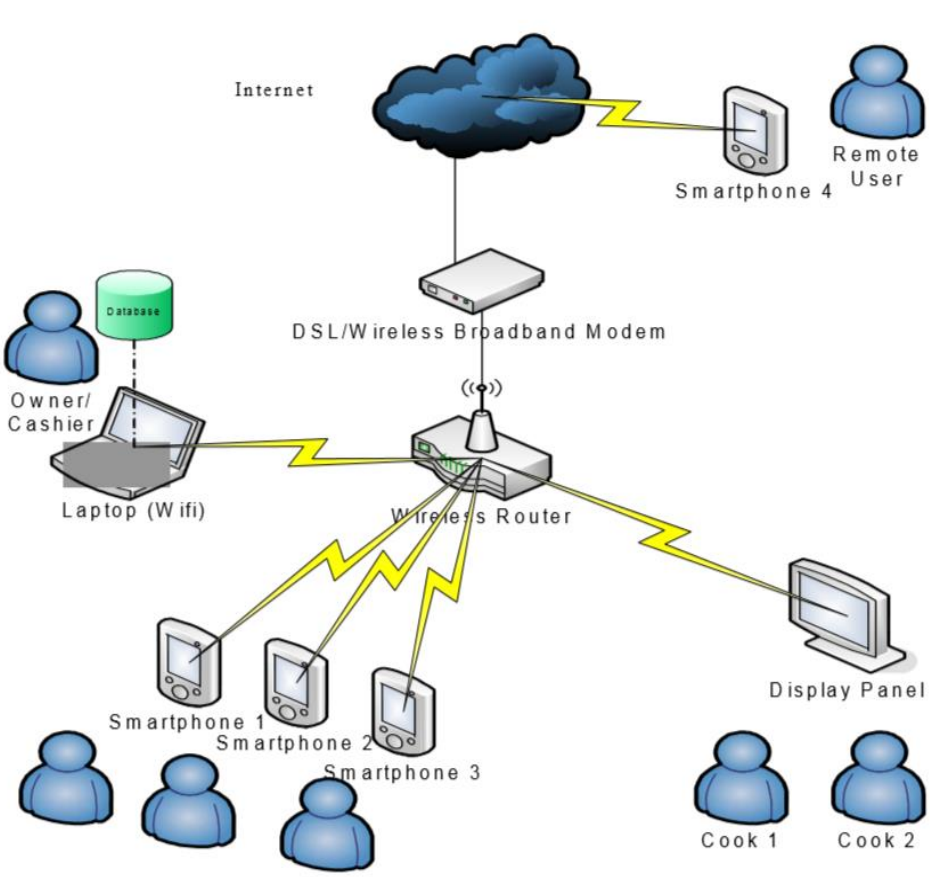


Figure 2.1 The system architecture of CWOS-RTF with Internet access (Samsudin *et al.*, 2011)

2.3.2 A Proposed System For Touchpad Based Food Ordering System Using Android Application (Bhandge *et al.*, 2015).

This research work was aimed to design and develop a wireless food ordering system in the restaurant. The project presented in-depth on the technical operation of the Wireless Ordering System (WOS) including systems architecture, function, limitations and recommendations. It was believed that with the increasing use of handheld device e.g. PDAs in restaurants, pervasive application would become an important tool for restaurants to improve the management aspect by utilizing PDAs to coordinate food ordering could increase efficiency for restaurants and caterers by saving time, reducing human errors and by providing higher quality customer service. In this system the customer ordered the food by using android based touchpad. Figure 2.2 shows the system architecture, which covered three main areas of the restaurant: the serving area, the restaurant owner's working desk (cashier table), and the kitchen. Customer first orders the food from the touchpad looking at various combination of food which is further carried to the kitchen for fulfilling the order and the same is passed for billing at the each customer's tablet (Bhandge *et al.*, 2015).The algorithms that were used in this project were; K-means clustering and Apriori algorithm.

2.3.3 Design and Implementation of Online Food Ordering System

(Gan, 2002) developed an online fast food restaurant ordering system that allowed customers to place orders anytime at any place. The system helped to manage order from customer as well as advertise promotion. It allowed kitchen staff to view ordering information, management to manage fast food raw materials and staff to search customer delivery and profile information. This system helped to reduce queue issues during peak hours, speed up food preparation and increase customer volumes. As a result, market share of fast food restaurant was boosted up and increases return of investment for the investor.

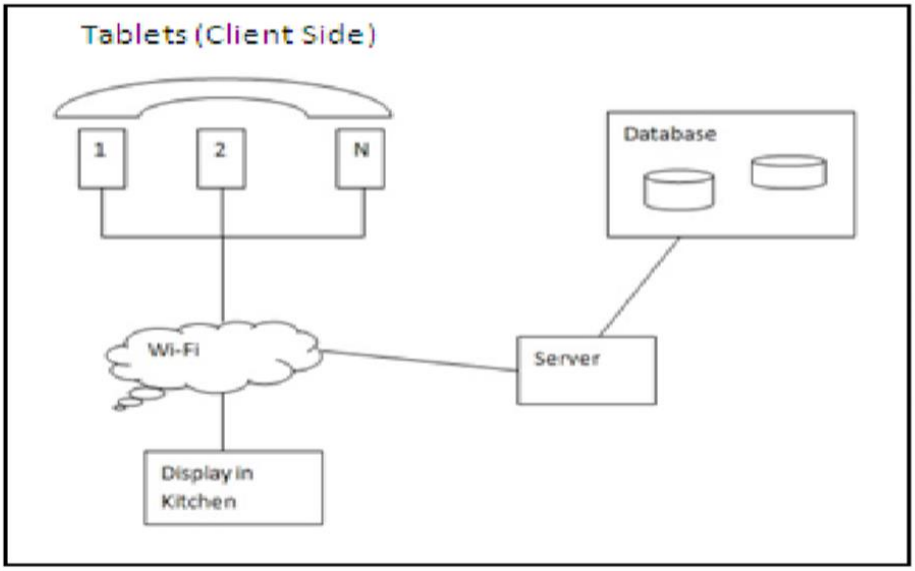


Figure 2.2 System Architecture for Touch based food ordering system (Bhandge *et al.*, 2015).

(Leon, 2008) mentioned that there were several aspects that should be included in a good online food ordering system. System should be simple to navigate, not clustered and easy to make an order, (Sharma, 2007,) designed with professionals looking with search engine optimize capability and available 24hours. The system should also have a secure payment gateway to protect their customers' credit cards information, fast and keep track on orders and sales history easily as well as generate a comprehensive sales report, (Sharma, 2007).

2.3.4 Online Food Order System For Restaurants (Mayurkumar , 2015).

This system greatly simplified the ordering process for both the customer and the restaurant. It presented an interactive and up-to-date menu with all available options in an easy to use manner. Customers were able to choose one or more items to place an order which will land in the Cart. Customer could view all the order details in the cart before checking out. At the end, customer gets order confirmation details. Once the order was placed it will be entered in the database and retrieved in pretty much real time. This allowed Restaurant Employees to quickly go through the orders as they are received and process all orders efficiently and effectively with minimal delays and confusion (Mayurkumar , 2015).

The motivation for designing this application came about because of the involvement in the fast food industry and also the desire for web designing and JAVA programming language. The languages used to build this application were JavaScript, JSP, HTML and Java at client facing whereas Oracle database at the back-end. This software was developed to help computer science students to learn about the Web application designing using JSP and HTML from their basic capabilities to build a complete working application from scratch. Furthermore, it gives insight about how GUI interacts with server-side language, Java, and finally with the Oracle database.

2.3.5 Wireless Customizable Food Recommendation System Using Apriori and K-means Algorithm (Kadale *et al.*, 2015).

The existence of wireless technology and the emergence of mobile devices enable a simple yet powerful infrastructure for business application like restaurant management system. Technology can be deployed efficiently to manage all the day-to-day tasks in restaurants. By using a new software-oriented approach we can eliminate a number of counters leading to a savings in space as well as staff for a restaurant. A wireless customizable food recommendation system was developed in this project and Apriori and k-means algorithm was used for analysis. This system increases quality and speed of service. This system also increases the customer's appreciation of the efficiency of the restaurant. However, this system helped the customer to make well informed choice of a dish based on the description provided by the system. Figure 2.3 illustrates the system architecture of the system which is similar to the works of Bhandge *et al*.(2005). However, this system recommended some areas of further study like enabling card payment, adding raw kitchen materials management to the existing system, and also extending the existing system to a chain of multiple restaurants.

2.3.6 Design and Implementation of Ordering System for Restaurants (Swapna *et al.*, 2012)

This implemented method mainly aimed at designing and implementing completely automated menu system in restaurants to provide a user-friendly ordering environment. Zigbee module was also used in this system. The transmitter section of the system consists of an ARM cortex LPC1768 microcontroller. The input module is a touch screen sensor with GLCD, and speech Recognition, which takes the input from the user and provides the same to the microcontroller. The output module is Zigbee module which makes the communication between system at table and system at cooking department. The ARM LPC2148 micro controller which is at the receiver section takes the order which is displayed on GLCD along with user table number.

The ordering program is a visual program running on Embedded C. It implements a graphical and touching user interface. We use the QT framework to develop the program, which is a cross-platform GUI application development framework widely using in Embedded C. This program uses the embedded system's UART to communicate with ZigBee module, the final information was transmitted to the central ZigBee node who connect to server. All of above realized real - time information transmission and management. In this project, they designed an intelligent restaurant self-service ordering information system based on ZigBee wireless technology.

2.3.7 Software Development Aspects of a Mobile Food Ordering System (Kulkarni *et al.*, 2008).

A Mobile Food Ordering System (MFOS) implemented in this project was one of the tools that intended to provide a food ordering application on mobile devices for ordering food from different stores with the option of both delivery and pickup. MFOS offers many useful features to customers, including the map location of the food store, detailed menus, and previous order lookup. By using MFOS, the users do not have to wait in a queue at the restaurants. MFOS was designed and developed to provide Pocket PC users with a single tool with which they could order their favourite food from different cuisines without having to wait in the queue and, importantly, be able to order food from anywhere. While developing the interface of MFOS, researching human computer interaction challenges as well as designing a system which can deal with them and provide more useful features to the user were significant objectives of this paper. MFOS aimed to offer some "cool" and handy features to the users, which other products in the market do not provide in the same way as MFOS. The location of the food store on the map was one of the very useful features of MFOS, which uses the latest functions of the Pocket PC. Similarly, the ability to call the food store, view the menu, pay the order with a credit card.

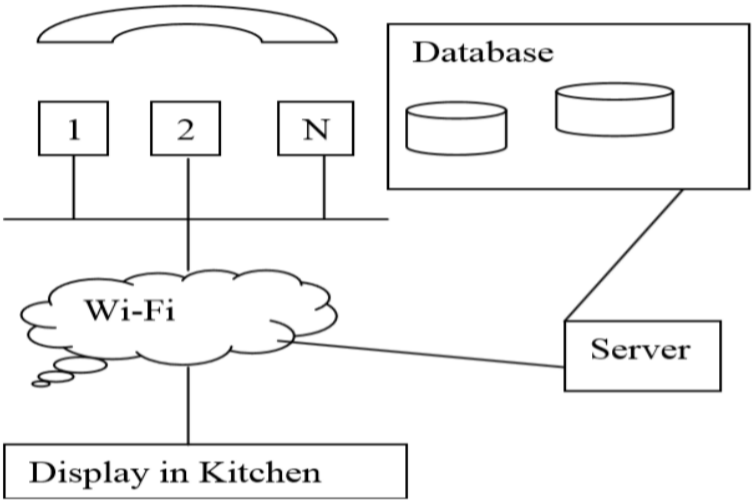


Figure 2.3 System Architecture of Customizable Ordering System (Kadale *et al.*, 2015).

MFOS only works on Pocket PC with Windows Mobile 6 Professional. To reach a wider audience it is necessary to make sure that MFOS can work on other platforms as well. As a development language they used C# so with relatively few modifications this application could work on other Windows Mobile OS. For other handheld devices such as iPhone and Blackberry this application has to be re-written in programming languages compatible with their OS. Another idea for reaching more customers would be for users who do not have internet connections on their devices, they should be able to send SMS messages for ordering (Kulkarni *et al*, 2008).

2.3.8 Implementing Customizable Online Food Ordering System Using Web Based Application (Chavan *et al.*, 2015).

This system used an alternative method of ordering for the customers instead of the traditional method. This method was the "Food Pre-Order System using Web Based Application" in which customer can be able to create the order before they approach the restaurant using a smartphone. When the customer approach to the restaurant, the saved order can be confirmed by touching the smartphone. The list of selected pre-ordered items shall be shown on the kitchen screen, and when confirmed, order slip shall be printed for further order processing. Figure 2.4 shows the architecture of the system. The android application is used to make orders from table (Chavan *et al.,* 2015). The restaurant-owner's laptop/tablet will keep track of customer records and also customize menu using server application. The central database is used for restaurant-owner to store updated menu information and order details. Three main areas of restaurant are connected using wireless technology. The Android application is used to find out the location in restaurant according to its longitude and latitude. Instead of using PDAs to interface with customers, they used smart phones or tablet to provide necessary interfaces for customer to view and order menu. With private login system, customers can view and make order and receive updates in real-time and collect receipts right from the smart phone itself.

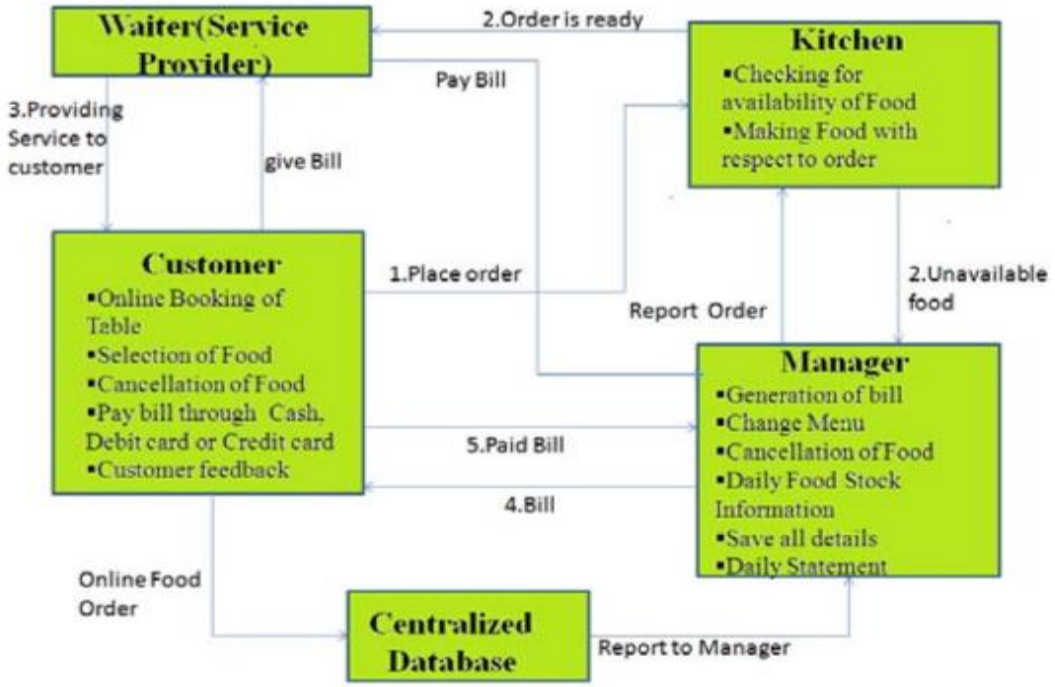


Figure 2.4 System Architecture of Web- based application (Chavan *et al.*,2015).

2.4 AUTOMATED SYSTEMS

2.4.1 Automatic Restaurant Order System Using Zigbee (Lodhi *et al.*, 2006).

This project provided a low-cost, convenient and easy to use system for automating order placement system for restaurants. Each table of restaurant had a menu display unit which was powered by microcontroller. The client would scroll menu list using keypad provided along with. Customer could order his food or drink just using this keypad. The aim of this project was to build an automated order system using ZIGBEE. Here, each table was provided with a microcontroller based order placement unit. The unit had a keypad to browse through the menu. The menu items, their cost and information would be displayed on the LCD connected to microcontroller. User can navigate through menu using keypad provided. The data for the menu can be written on an EEPROM connected to each such microcontroller based unit, so that portable data updating is possible (by changing only the EEPROM). Upon finalizing the order the user will be able to place it using keypad. The order placed shall be transmitted to the central server (PC) which will also have a ZIGBEE module connected to it for data reception. Multiple such slave units can be installed. Figure 2.5 shows the basic block diagram of automated restaurant using ZigBee module. ZIGBEE is basically a specification for a suite of high level communication protocols using small, low-power digital radios based on an IEEE 802 standard for personal area networks. ZIGBEE devices are often used in mesh network form to transmit data over longer distances, passing data through intermediate devices to reach more distant ones. This allows ZIGBEE networks to be formed ad-hoc, with no centralized control or high-power transmitter/receiver able to reach all of the devices. Any ZIGBEE device can be tasked with running the network. ZIGBEE is targeted at applications that require a low data rate, long battery life, and secure networking. ZIGBEE has a defined rate of 250 Kbit/s, best suited for periodic or intermittent data or a single signal transmission from a sensor or input device (Lohdi *et al.*, 2006).

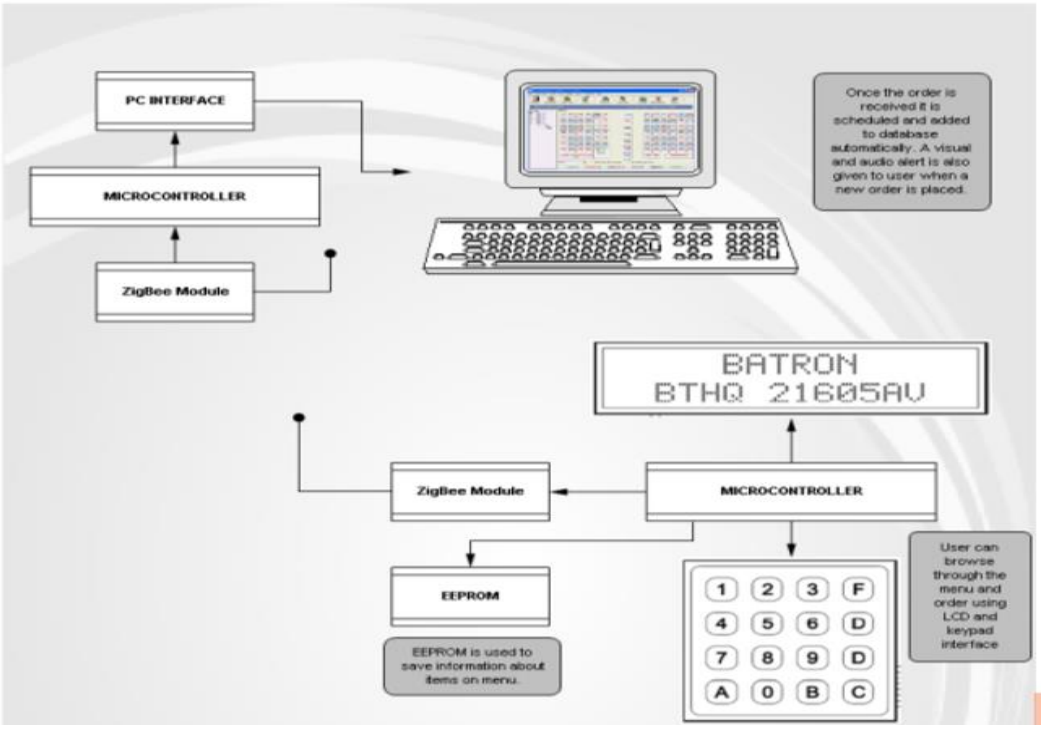


Figure 2.5 Basic Block diagram of Automated Restaurant Using ZIGBEE Module (Lodhi *et al.*, 2006**).**

2.4.2 Restaurant Automation (Ford *et al.*, 2006).

The purpose of the Project is to develop a Wireless Handheld Ordering System (WHOS) to be used in a restaurant. The system will accept order information and send it to the kitchen while the waiter is standing at the customer's table. The system needs the ability to notify the waiter when the order is done. When the customer has finished their meal, the waiter will be able to display their bill and print a final receipt that will be given to the customer. If the customer is paying by credit card the waiter will be able to swipe the credit card information using the WHOS. The system was able to keep track of all orders taken each day.

The WHOS consisted of a handheld device (hereinafter called "product") which was able to communicate wirelessly to output screens in the kitchen and to a central computer used to record/document all orders. Figure 2.6 will show the state machine of how the central computer operates. The device was modelled using a Hardware Development Kit (HDK) based on the Intel PXA270. All inputs into the WHOS was done using a Liquid Crystal Display (LCD) touch screen. Receipts given to customers was printed from a thermos printer connected to the system using Bluetooth communications. All the programming for the devices was done using Microsoft Visual C++. However, this project had some limitations which includes; the size of the battery used limited the amount of time the device could be used before it needed to be recharged. The size and type of monitor available in the kitchen was also important. The cost for items needed to complete the Project. The demo needed to be toned down a little bit.

2.4.3 Automated Food Ordering System With Real-Time Customer Feedback (AOS-RTF) (Tanpure *et al.*, 2013).

This project was motivated by the use of Android mobile OS in Health and other applications. They presented the use of Android Devices in Business applications, namely the food ordering system in restaurants. The Objectives of the implemented system were:

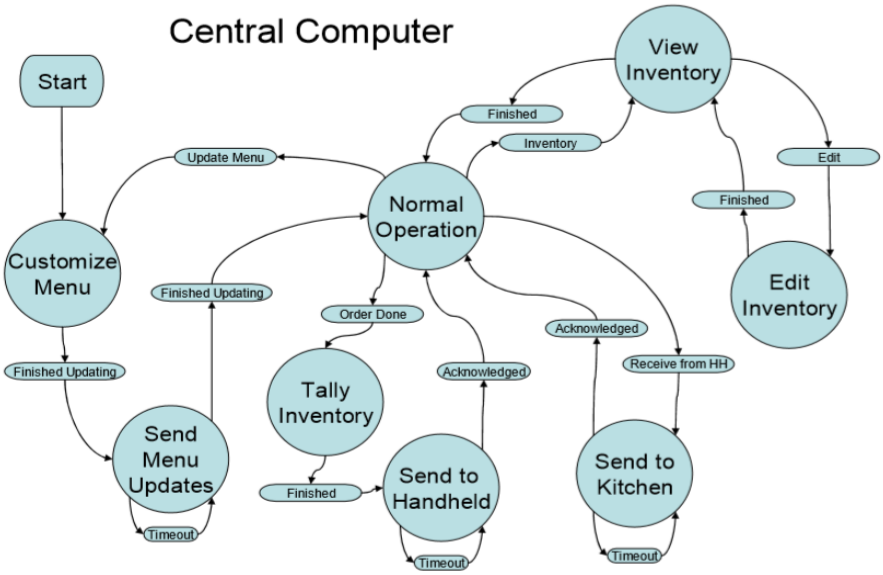


Figure 2.6 The state machine of how the central computer operates (Ford *et al.*, 2006).

 To combine Wireless technology and Android Mobile OS to automate food ordering process.

 To Eliminate or at-least minimize the flaws in conventional system by automating the Food ordering system in restaurants.

 To implement real-time feed-back between restaurant owner and customer about order status (Tanpure *et al.*, 2013).

The system architecture of AOS-RTF depicted in figure 2.7 covers the three main areas of restaurant: the Serving area, the Kitchen, and Restaurant-Owners desk (Cashier table).

Conceptually the AOS-RTF is built using four main components:

1. The android application on android mobiles of customers to make orders.

2. The server and web applications on the restaurant-owner's laptop to customize menu and keep track of customer records.

3. The central database for restaurant-owner to store updated menu information and order details.

4. Wireless infrastructure to support networked communication (Tanpure *et al.*, 2013).

The system architecture possessed portability feature. It accommodated all types of restaurants including those with and without internet access. This system was convenient, effective and easy thereby improving the performance of restaurant's staff. It also provided quality of service and customer satisfaction.

Figure 2.7 The system architecture of AOS-RTF with internet access (Tanpure *et al.*, 2013).

2.4.4 Restaurant Automation (Eng *et al.*, 2007).

The aim of this project was to develop a software system that would eliminate the need of traditional pen/paper approach for privately- owned restaurants. The project was focused on making the restaurant fully automated such that it is easier to co-ordinate various work activities that go on inside a typical restaurant. The main features of the project include:

- Organizing a database for a medium sized restaurant
- Coordinating work activities of the various actors – Host, Waiter, Cook, Busboy and Manager
- Increase efficiency by minimizing time between an order is placed and the billing
- Increase profits by reducing operating costs and increasing revenues by increasing efficiency
- Archiving information of the workers and hours worked (Eng *et al.*, 2007).

There are various architectural styles that was suggested for creating this application. The first thought was to use the Repository Architectural Style in which subsystems access and modify a central repository. Each of the subsystems are independent from each other and only interact with the data through the central repository. This style was quickly thrown out because of the high coupling found between the subsystems and the repository. Therefore, it is not portable and changes to the user interface would be difficult to implement quickly.

In the second architectural style, the work investigated the Model/View/Controller or MVC Architectural Style. In this architectural style, you separate all the subsystems into three categories. The model subsystems maintain the data of the application, the view subsystems display and format the data to the user, and the controller subsystems are responsible for managing the interactions between user and system. The model subsystem should be independent of the view and control subsystems. Though, they found this style suitable for their

application because it allows changes to the user interface to be very easily implemented but however, it was not chosen.

The last architectural style they looked at was the Three-Tier Architectural Style; which is very similar to the MVC Architectural Style. The subsystems are once again, separated into three layers: • Interface Layer – boundary objects that deal with the user (User Interface) • Application Logic Layer – control and entity objects • Storage Layer – the database Once again, partitioning all the layers allows changes to be made to them very easily. You can change the user Interface very easily without having to make changes to the database or how we deal with the data.

In the end, the Three-Tier Architectural Style was chosen because they were developing a web based application built on PHP/MySQL. Natively, PHP is not a framework that is based on the MVC Architectural Style. It is much closer to the Three-Tier Architectural Style because all requests are sent to the server from the client. The server is responsible for interpreting the request and then forwarding it to the database. The database sends the data back to the server which than formats that information and sends it to the client. At no point, is the server circumvented in the interaction between user and data. This is a linear flow of events and not a triangular flow which the MVC Architectural Style utilizes. Figure 2.8 shows the three tier architectural style used. The system suggested a better implementation of the GUI for the PDA for a future prospect due to their small screens and relatively slower operating speeds.

2.5 DIGITAL SYSTEMS

2.5.1 Digital Ordering System For Restaurant Using Android (Bhargave *et al.*, 2013).

In current formal dining environments, some form of physical static menu is utilized to convey the available food and beverage choices to customers. Menus are generally paper based and hence impose restrictions on the textual real estate available and the ability a restaurateur has to update them.

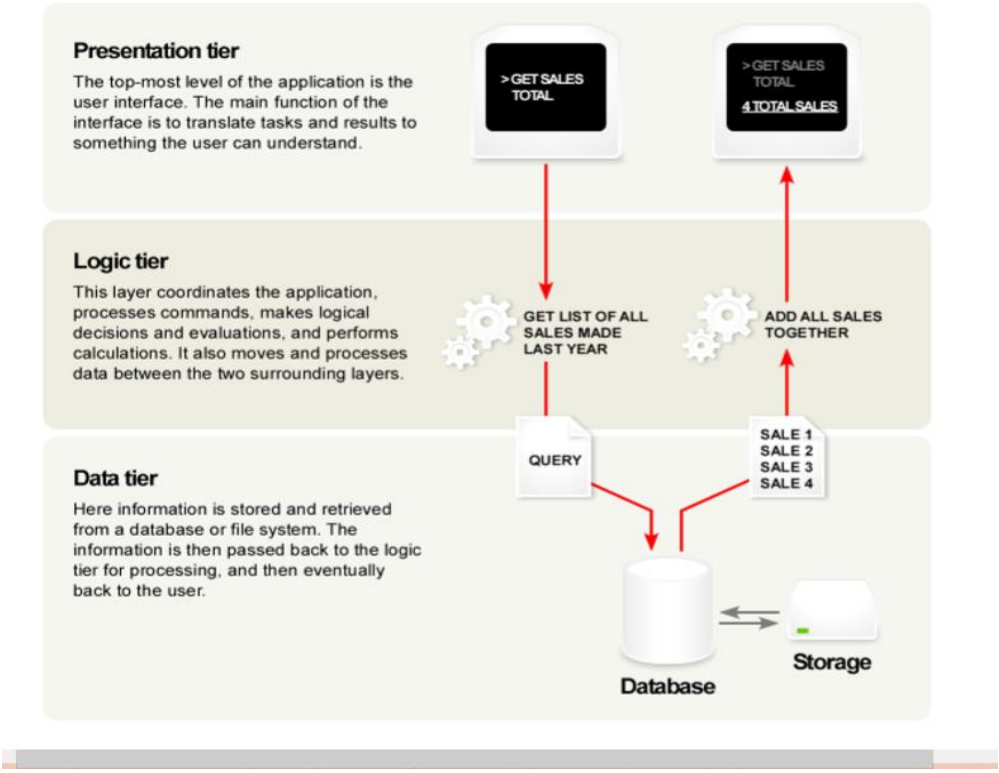


Figure 2.8 A Three-Tier architecture (Eng *et al.*, 2007).

This document specifies the requirements for a restaurant paper menu and ordering replacement strategy to alleviate the problems associated with the current archaic method. Here, three related concepts were encompassed by the general scope of the Restaurant Menu and Ordering System. The first pertained to the replacement of paper-based menus using an electronic format, the second relates to a complementary electronic strategy for the front of house handling of a customer's order and the third surrounded the process of transferring said electronic orders to the kitchen for preparation. Ordering, Waiting, Billing, Table Reservation, Home Delivery, KOT, and Advertisement were some of the features of the implemented system. However, this system has some advantages; Wastage of paper was avoided as their implementation was working just on tablet and does not need any paper work. e.g.-For taking the order, we are not using papers. Also, our menu card would be digitized. A customer going into restaurant does not has to wait for the waiters to take the order. As soon as he occupies a seat, he would order whatever he needs. As soon as the order is ready, it would be notified to the customer. So, there would not be any issue of late delivery in spite of the food being ready. The drawbacks however were; Tablets cost them more as they were more costly then than simple paper. Also since the system was conventional, more maintenance was needed, and technical assistance was also needed. This project helped restaurants to ease the management of the Restaurant and also give a technical touch which helped atomize the working of Restaurant (Bhargave *et al.*, 2013).The implemented system added to the efficiency of maintaining the restaurant's ordering and billing sections.

2.5.2 Design and Implementation of Digital Dining in Restaurants Using Android (Shinde et al., 2014)

The rapid growth in communication technology emergence of wireless technology and android devices has created quite a stir in the business transactions. Business in the hospitality industry has been greatly influenced and competition has increased due to improved food ordering

techniques. In earlier days, food ordering was a completely manual process where a waiter used to note down orders from the customers using pen and paper, take the orders to the kitchen, bring the food and make the bill. Although this system was simple, it required extensive investment in purchase and storage of paper, large manpower and also was prone to human errors and greater time consumption. PDA based systems and multi-touchable restaurant management systems were used as a literature review in this system. PDAs (personal digital assistants) are much in use because of their portability feature and ability to communicate with personal computers but they too have some limitations. A PDA-based system lacks ubiquitous communication, is exposed to health hazards, requires training of attendants, the need of having attendants to operate, the inefficiency during peak hours and small screen size. However, for them to overcome the limitations of above system, they proposed a digital dinning system based on android technology. It is a wireless food ordering system using android devices. Android devices have gained immense popularity and have revolutionized the use of mobile technology in the automation of routine task in wireless environment. Android is a Linux based operating system for mobile devices such as smart-phones and tablets. Considering the promising future of Android market, it is beneficial and worth to write applications for android that target masses of people. The system architecture of Digital Dining in restaurants is shown below in figure 2.9 .The architecture covers the three main areas of restaurant: the Serving area, the Kitchen, and the Cashier counter (Shinde *et al.*, 2014).

2.5.3 Digital Dining System Using Android (Bhingre *et al.*, 2015).

This system was based on customers being able to place and book tables in advance by selecting desired table from a canvas. Additionally, diet-conscious customers could customize their menu and play games in order to win exciting prices and offers. They also created a website for restaurant managers on an android application.

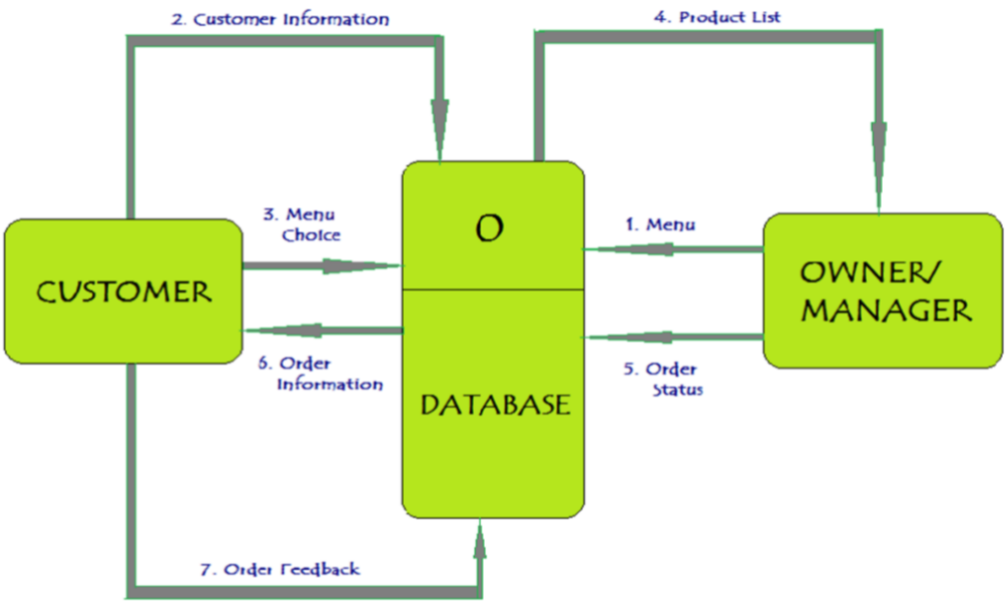


Figure 2.9 System Architecture of Digital Dining Restaurant (Shinde *et al.*, 2014).

System architecture of Digital Dining System using Android is shown in Figure 2.10 below. There are three modules in this system and they are as follows:

1. Android application for the patron

2. Web-based application for restaurant manager and

3. Application for admin to control all the activities (Bhingre *et al.*, 2015).

This system is called a digital dining system where patrons can place order before visiting a restaurant along with new features such as digitized menu, digitized table booking, and gaming. This system is consistent, accurate, and efficient thereby providing customer satisfaction. This system can cater to the needs of a variety of customers, attract potential customers and ultimately improve a restaurant‘s business.

2.5.4 Digital Table Booking and Food Ordering System Using Android Application (Dhore *et al.*, 2014).

This project work aimed to design and implement a remote food ordering system, through which one can order food before visiting a restaurant, book table, and also make payment. Moreover, two or more customers could place orders for the same table from remote locations. This application consists of three applications within itself. First is for the customer who can book tables and place orders. Second is for the kitchen unit in a restaurant, which enables the staff to view current orders. Third is for the manager of that restaurant in order to keep track of all the transactions. They made use of a recommendation engine that suggests menu to a customer while placing order. They also made use of a compression algorithm that compresses the size of images used throughout the application at various stages. This system increases quality and speed of service.

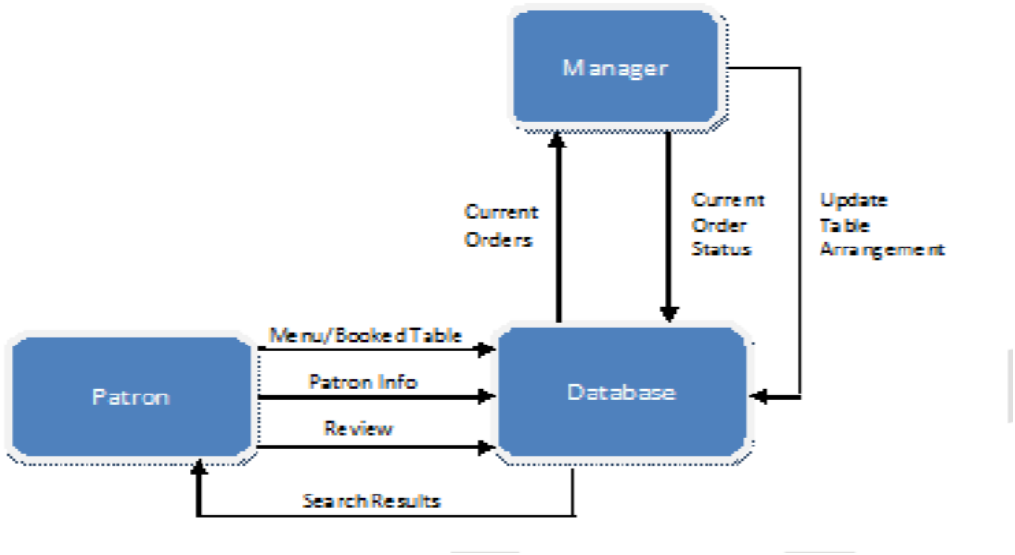


Figure 2.10 Architecture of Digital Dining System (Bhingre *et al.*, 2015).

It also increases the popularity of restaurants among potential customers. This system gave a cost-efficient opportunity to give customers a personalized service experience where they are in control of choosing what they want, when they want it – from dining to ordering to payment. They chose an android platform because it is most widely used today and is very economical. Figure 2.11 shows the system architecture of the implemented system. The architecture covers the four main modules: the Customer, the Manager, the Administrator and the Kitchen section. Conceptually this system was built using four main components:

 The android application on the smart phones, the server application on the restaurant-manager's laptop/tablet to customize keep track of customer records, table bookings and time required to reach, the central database for restaurant-owner to store updated menu information and order details and a wireless connectivity between the manager and the kitchen area of restaurant (Dhore *et al.*, 2014).

2.6 MENU AND MANAGEMENT SYSTEMS

2.6.1 Restaurant Menu and Management System (Oberhammer, 2014).

The development of the menu and management apps in the work of (Oberhammer, 2014) provided a number of opportunities to learn and gain experience in many different areas of software development. The two main learning objectives for this project were to explore user interface design for mobile devices and Wi-Fi Direct. Many of the features for the menu and management apps were completed for this project. Basic browsing of the restaurant's menu, including the ability to search and sort was implemented for both the menu and management apps. For the management app, additional features included the management of the restaurant menu, including the ability to add, edit and remove menu elements.

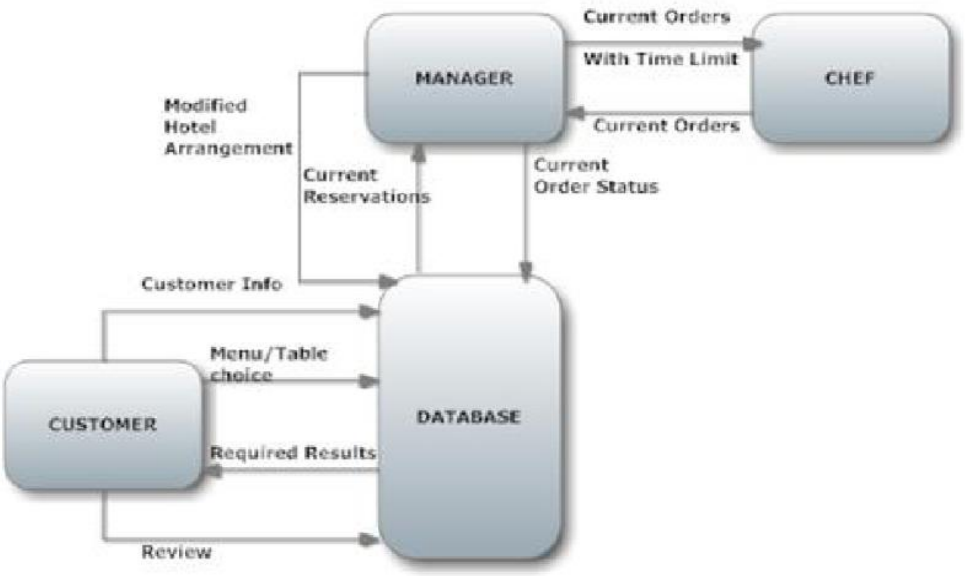


Figure 2.11 Architecture of Digital Table Booking System (Dhore *et al.*, 2014).

2.6.2 Design Implementation and Evaluation of A Menu Management System For Restaurants (Pernebayev *et al.*, 2013).

This work designed, implemented and evaluated a menu management system for an identified restaurant. The system provided automated order taking and menu management functionalities for the restaurant. Firstly, the project started with requirements analysis in collaboration with the restaurant. Secondly, based on the requirements the menu management system was designed, implemented and tested. Finally, in evaluation part, the system was analysed to study the effects of introduction of the menu management system in the restaurant. The motivations for this project was;

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 (Pernebayev *et al.*, 2013).

The menu management system was designed based on a mixture of multitier architecture, RESTful architecture style and Model View ViewModel (MVVM) pattern. The multitier architecture provided 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 allowed the functionalities of the system to extend for future development. Figure 2.12 shows the component diagram of the system. 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 analyse the actual operations of the restaurant, and the associated improvements in the application made it more helpful for the order processing activity.

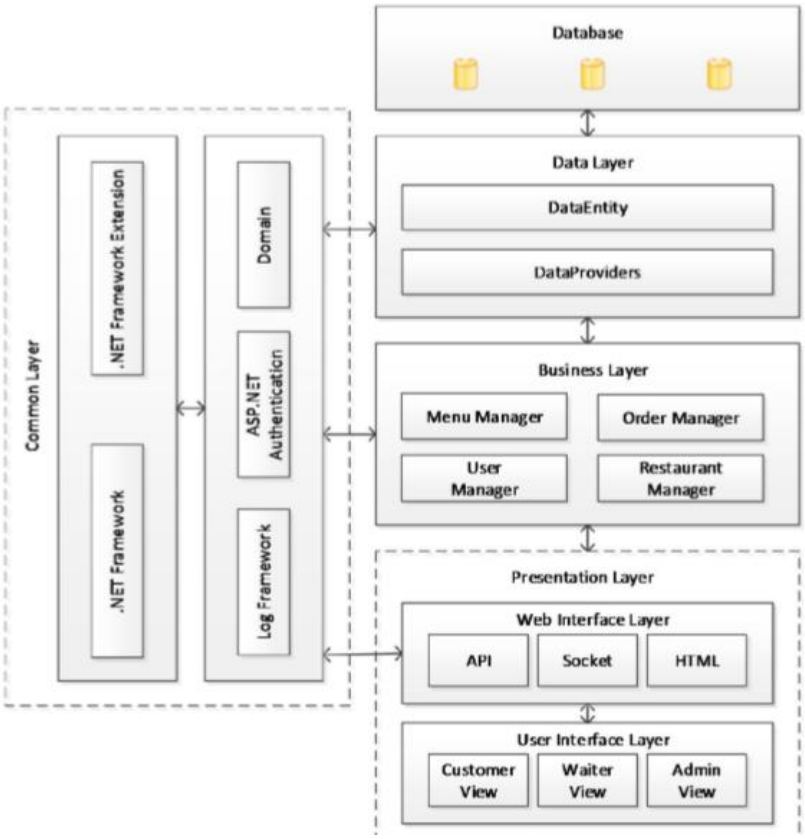


Figure 2.12 Component Diagram of Menu management system (Pernebayev *et al.*, 2013)

2.6.3 Implementation of Smart Restaurant with E-menu Card (Jakhete et *al.*, 2015).

The system consisted of a Smartphone/tablet at the customer table that also contained the android application with all the menu details. The customer tablet, kitchen display connects directly with each other through WiFi. Orders made by the customers will be instantly reach the kitchen module. This wireless application is user-friendly, improves efficiency and accuracy for restaurants by saving time, reduces human errors and provides customer feedback. This system is less expensive as it required a one-time investment for gadgets. The system block diagram of Touch and Order in restaurants is shown in figure 2.13. The android application on tablets at the tables. The tablets will be provided to customers, at their tables, allowing them to directly view the menu card and order immediately from their respective tablets. The tablets are the property of the establishment and are kept at each table. This system is also confined within the restaurant vicinity only. Some of the limitations this system faced were; status and feedback of order was not obtained, since it is confined to the restaurant vicinity there was limited distance, if the smartphone suffers from a defect the system will automatically go down and lastly end users not being able to use the tablet or smartphone properly (Jakhete et *al.*, 2015). They also suggested some future work like; implementing with Graphical LCD for displaying the menu as displayed on the android phone to make the system more delicate to use.

2.6.4 Application on Order Management System in Restaurants (Bora *et al.*, 2012).

This Literature presented an easy and more subtle way of communicating to realize a wireless food ordering system. This system implemented wireless data access to the servers and food ordering functions through both desktops and mobile devices such as tablets over a wirelessly integrated local area network. This application runs on devices such as tablets which provides convenience, improves efficiency and accuracy of restaurants by saving time and reducing

human errors. The system implemented the following functions: the waiter takes order from the customer into his tablet, the customer can visualize the order and bill, administrator has the authority to change the menu and has authority to view daily, weekly or monthly report on profits and lastly the kitchen staff can prepare and serve the order. Apart from these functions there were some limitations in this system like provisions had to be made to accept different types of payment like credit cards, debit cards, checks, tips, donations etc.

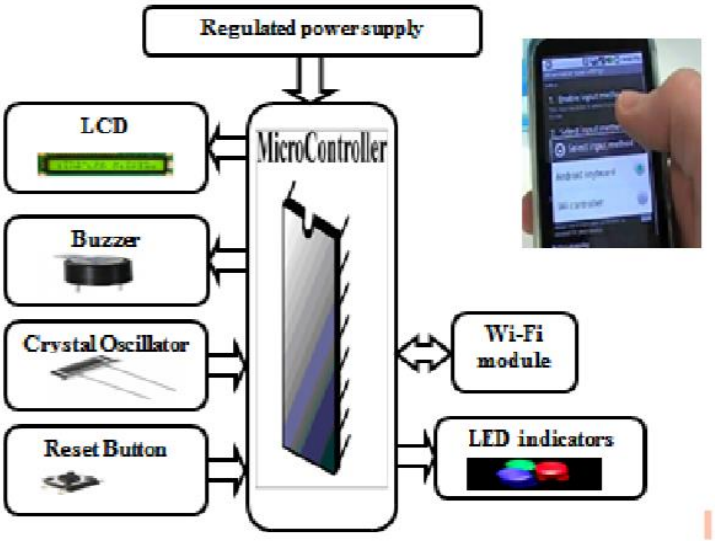


Figure 2.13 System block diagram (Jakhete *et al.*, 2015).

S/N	PROJECTS / WORK	WIRELESS ANDROID APPLICATION FOR PCS	MOBILE APPLICATION FOR SMARTPHONES	ELECTRONIC PAYMENT	USER SELECT MENU	CUSTOMER FEEDBACK	TABLE RESERVATION / HOSTEL DELIVERY BILLING	NETWORK INFRASTRUCTURE , WI-FI , INTERNET , VPN, ETC.	SECURITY
1.	Proposed Implement ation.	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**No**	**Yes**	**Yes**
2.	Bhandge *et al*.(2015)	**Yes**	**Yes**	**No**	**Yes**	**Yes**	**No**	**Yes**	**No**
3.	Tanpure *et al*.(2013)	**Yes**	**Yes**	**No**	**Yes**	**Yes**	**No**	**Yes**	**No**
4.	Bhargave *et al.(*2013)	**Yes**	**Yes**	**No**	**Yes**	**No**	**No**	**Yes**	**Yes**
5.	Lohdi *et al.(*2006)	**Yes**	**Yes**	**No**	**Yes**	**No**	**No**	**Yes**	**No**
6.	Ford *et al*.(2006)	**Yes**	**Yes**	**No**	**Yes**	**No**	**No**	**No**	**No**

Table 2.1: Comparison of some selected designs with the proposed system.

CHAPTER THREE
SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis and Design

This chapter provides the conceptual description of a Restaurant Automated Food Ordering System (RAFOS). The implemented system offers the ordering of food by a customer using android devices.

3.2 Logical Analysis of the Existing System

In the analysis of the existing system, some major operational inefficiencies were identified as shown in figure 3.1 but not limited to:

3.2.1 Payment method

In some other traditional restaurants, food orders are collected before payment. But in ABUAD restaurants, things are a little different here. First, payment is made using actual cash regarding the amount of food the user wants before you can get your food. The cash is converted to ticket and then proceeded to order for food on a building queue. The consequence can be very tiring especially for customers who are stressed.

3.2.2 Ordering process

With the ticket being given, food is got by queuing at the counter of the particular meal of your choice. The waiters in the restaurant are too few to attend to all the customers that wants to get food at the same time which causes them to loose focus and start serving food inaccurately without considering the amount of food the customer chooses to get. In this case, resulting to unsatisfied customers.

3.2.3 Calculation and accounting

Calculation and accounting here takes a lot of energy and calculations because the tickets are counted manually and if they don't correspond with the money at hand, there will be a loss. This can become very tedious, time consuming and human error can easily occur. Customer can become distraught with this poor level of service.

3.3 Logical Analysis of the Implemented System

The major contribution of the implemented system is automating the payment process in ordering of food so that payment becomes easier and faster and also in the aspect of customer feedback. There are different steps involved in ordering of food which requires the customer's menu, the device and also the restaurant staff. In order to make an effective order, the customer's menu has to be created and payment confirmed. If the payment has not been confirmed by the administrator, the customer has to re-order his food and the payment also.

There are two main components of the implemented system. The first component is the user interface that shows and displays the order menu and process for the customer running on a tablet PC. The second component is the central system that receives all orders made by the customer for processing and also stores all information on orders and payment. These components are explained in detail in this chapter.

General architecture of the implemented system is shown in Figure 3.4

- In this architecture, the users interact with the Menu application tablet to create choice of order and also to make payment.
- The Restaurant staff is equipped with a menu interface tablet and handles all operations made with the tablet by the customer.

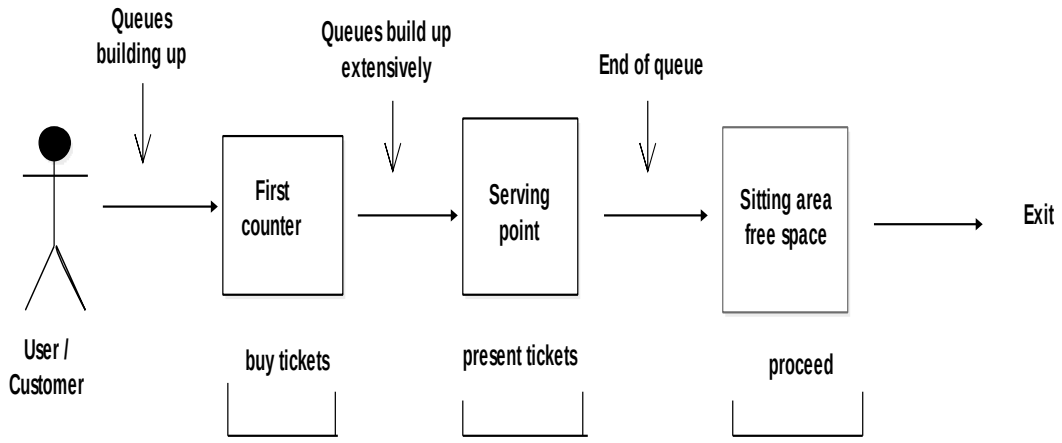


Figure 3.1 Logical analysis of the existing system

Figure 3.2. A snapshot of a 100 naira ticket used in cafeteria 2

Figure 3.3. A snapshot of a 200 naira ticket used in cafeteria 2

- All orders made are sent to the central system through a network. The system receives orders and payment then processes orders immediately.
- The Restaurant Administrator is in charge of the central system. The Admin sets up the system to perform effectively. He confirms payment made by the customer and approves payment. After payment has been approved, he issues a tag to each customer pending the order being processed and prints payment receipt or tag.
- And the database is used to store all information on orders and payments on the central system.
- The network used will be either the University's internet or Wi-Fi. It is the communication medium between the tablet and the central system.
- The server shares data and information and responds to request from the system or the tablet.

3.4 Components of the Implemented System

There are two components of the implemented system: user interface and central system. These components with sub-components are illustrated in Figure 3.5 and described as follows.

3.4.1 User Interface

The implemented system uses an android device (tablet PC) for creating, choosing and sending food menu to the kitchen for processing. This user interface is made up of two subcomponents which includes;

3.4.1.1 Hardware

The android device used in the implemented system will be equipped with fundamental hardware components.

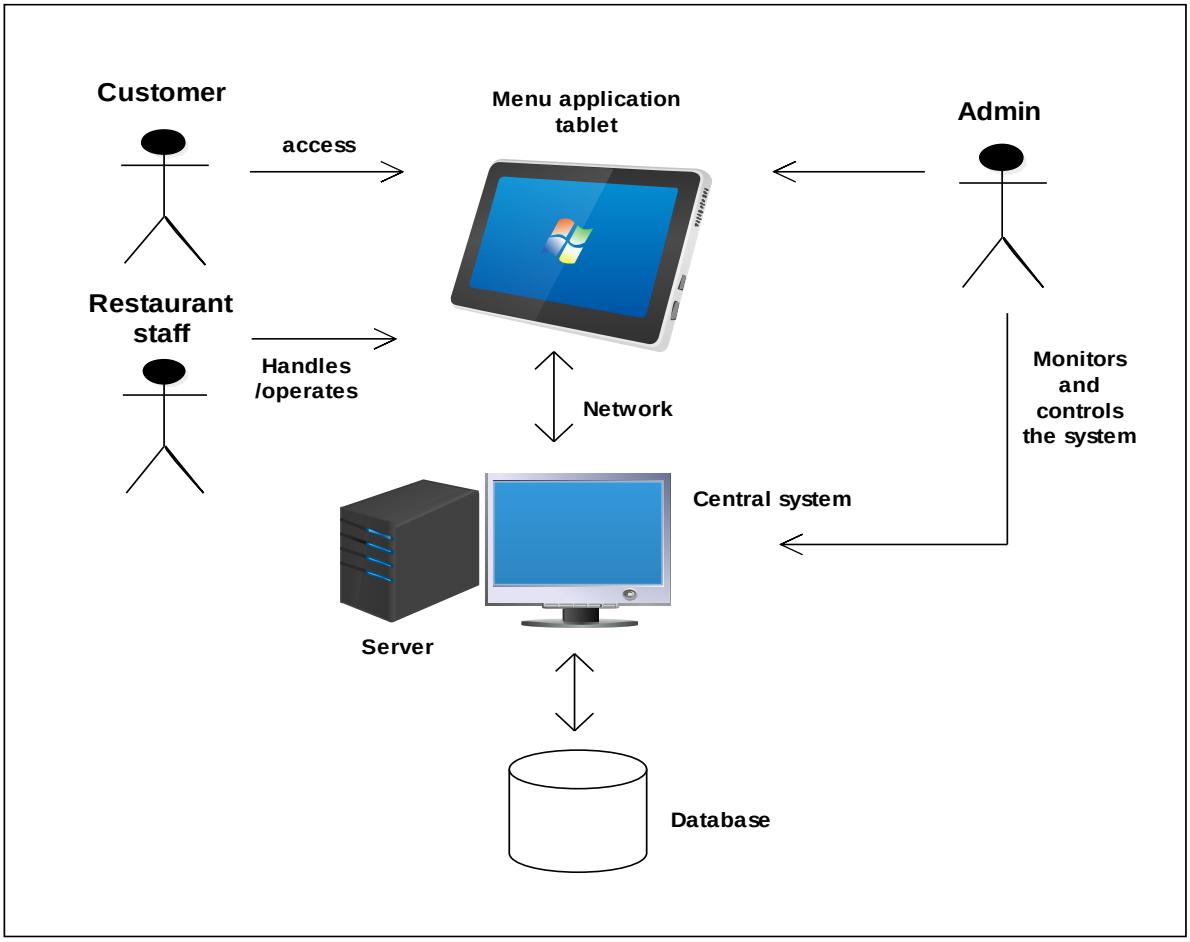


Figure 3.4 Overall architecture of the implemented system that shows how the system works

3.4.1.2 Software

The activities associated with the user interface are mainly accomplished by the software installed on the android device. The software includes various modules used for different purposes;

3.4.1.2.1 Screen Module

The screen module provides a structure of the user interface screen. This module shows the user the information layout.

3.4.1.2.2 Network Module

The network module provides the software the ability to reach the network services of the android device in order to turn on the Wi-Fi or wide area network access.

3.4.1.2.3 Communication Module

The communication module provides the software the ability of sending data through communication technologies like Wi-Fi or Wide area network (WAN) access. If the Wi-Fi is not available, the module sends data through the mobile network.

3.4.1.2.4 Text Module

The text module enables the user to insert textual information regarding the food order.

3.4.2 Administrator Interface

The administrator interface is the central system interface where the administrator has access to all information on order and payment made by the user. However, the administrator interface has access to all information via several software modules.

3.4.2.1 Database Module

The database module provides the central system to store both the order information acquired by the user interface component and also the payment by the user. If any component requires information stored in the database or the administrator requires information about a particular order, it uses the database to get access to the related data or information.

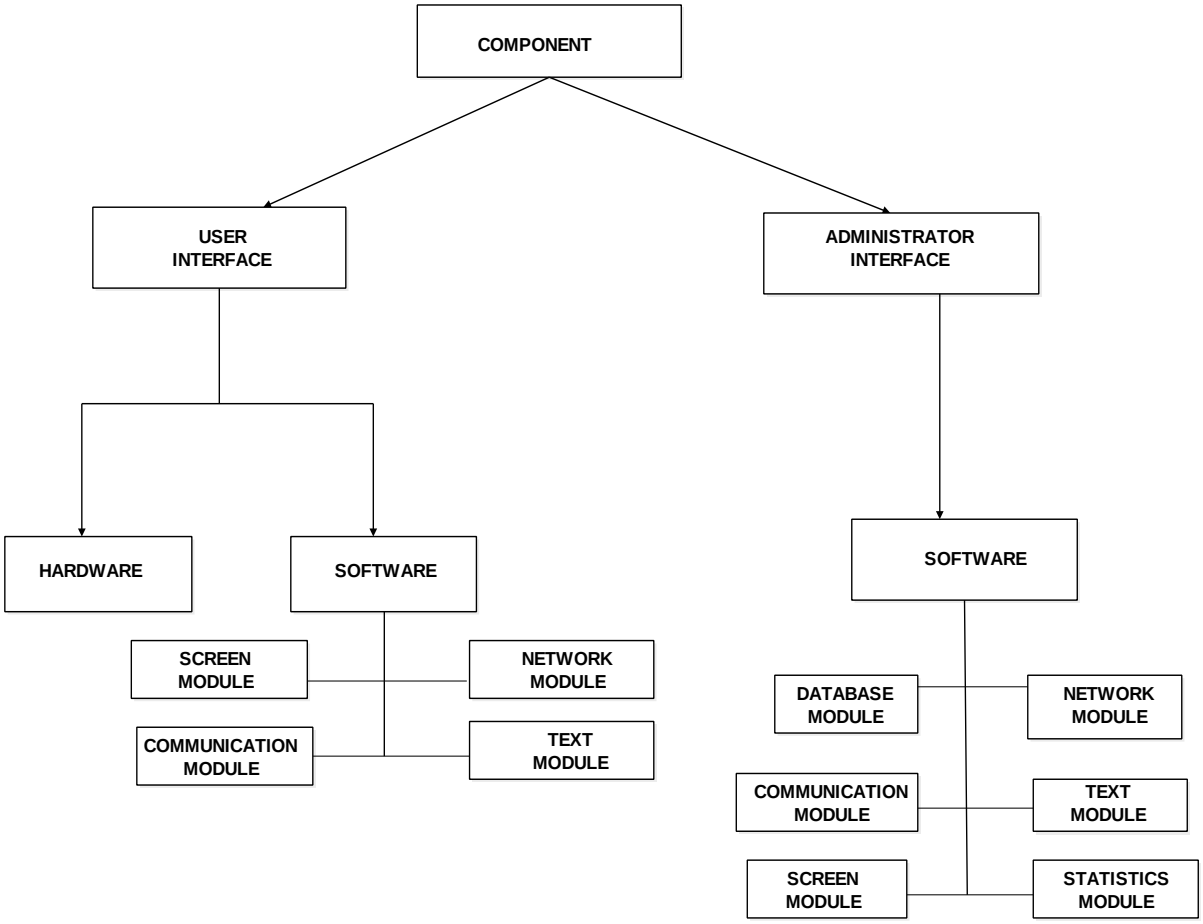


Figure 3.5 Components and sub-components of the proposed system.

3.4.2.2 Network Module

The network module enables the software to reach settings of the network service of the central system with a server by the side for communication of data.

3.4.2.3 Communication Module

The communication protocol used here is Single Object Access Protocol (SOAP).This module enables the transfer of data and information from the user interface to the central system through Wi-Fi or mobile network.

3.4.2.4 Text Module

The text module enables the administrator to insert textual information regarding any information on the user and the order.

3.4.2.5 Screen Module

The screen module enables the administrator view the information structure. It shows the administrator the information layout.

3.4.2.6 Statistics Module

This module enables the system to perform calculations and accounting on the daily income of the restaurant.

3.5 Architectural Context Diagram

The architectural context diagram otherwise known as (ACD) as shown in Fig 3.6 describes the actors of the system the superordinate systems, the subordinate systems and the peers which will be further explained below.

Student / Staff / Admin

These are called actors. They are the entities that interact with the system.

Application Interface and Central System

These are called Superordinate systems. They are the systems that are used by the target system as part of some higher level processing skill.

Payment method and Order taken

These are the Subordinate systems that are used be the target system to provide data or processing needed to complete target functionality.

University Internet (Wi-Fi etc.)

It is used by the system for communication between the system servers.

3.6 Context Diagram of the Proposed System

The context diagram shows the relationship between components of the implemented system. It shows how the RAFOS interacts with the Admin, the central system and the student/staff which are the components of the system. The context diagram is illustrated in Figure 3.7

3.7 Activity Diagram of the Proposed System

The activity diagram shows the data flow between the components of the implemented system from the beginning of the process till the end of the process. The activity diagram is illustrated in Figure 3.8

3.8 Use Case Diagram

Every system has actors that interact with them. This use case diagram depicts the actors being the student/staff, the restaurant staff and the restaurant administrator and how they are related to the different ordering processes of the implemented system. The use case diagram is illustrated in Figure 3.9

3.9 Input- Process - Output - Storage Diagram

The Input – process – output - storage diagram of the implemented system otherwise known as IPOS diagram describes the structure of an information process. It illustrates the outcome of the implemented system. How the input design, output design and database design looks like. The IPOS diagram is illustrated in Figure 3.10.

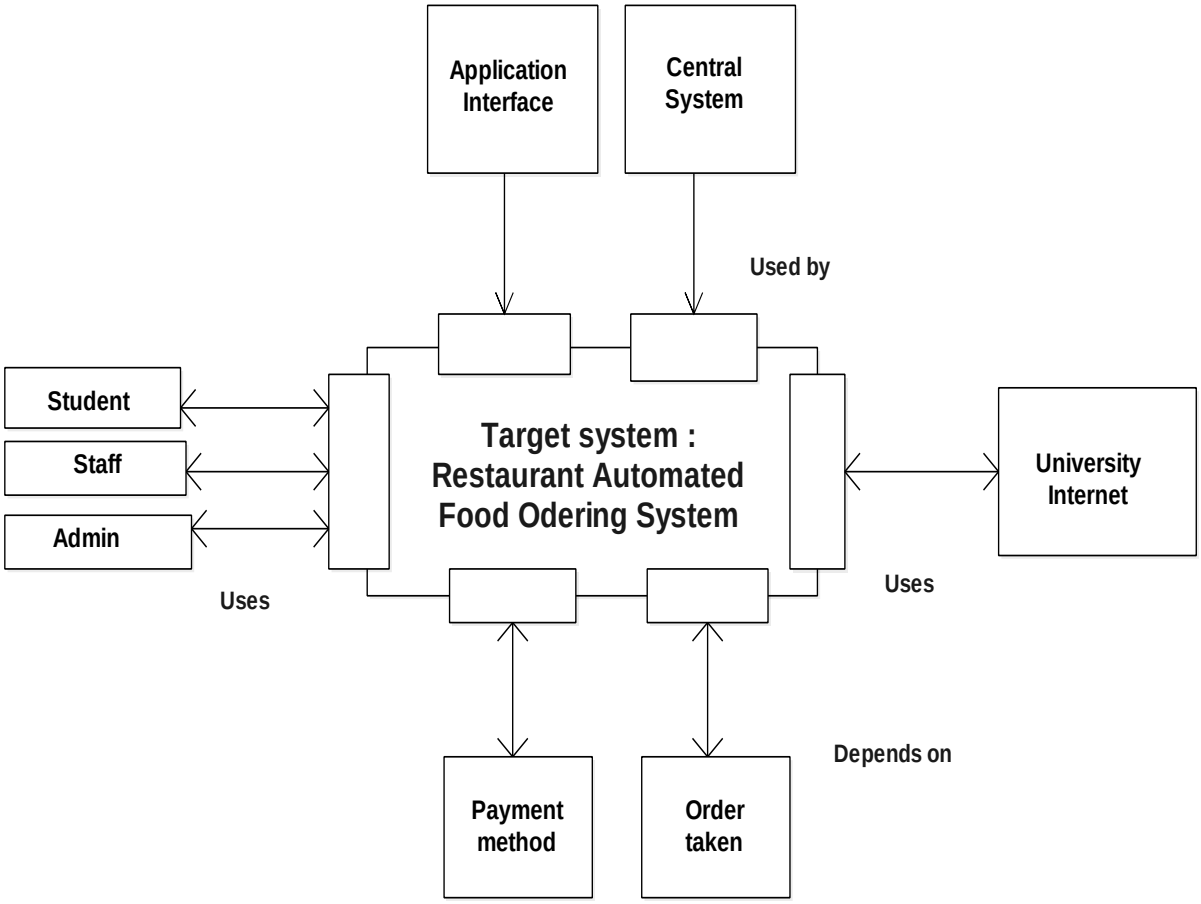


Figure 3.6 Architectural Context Diagram

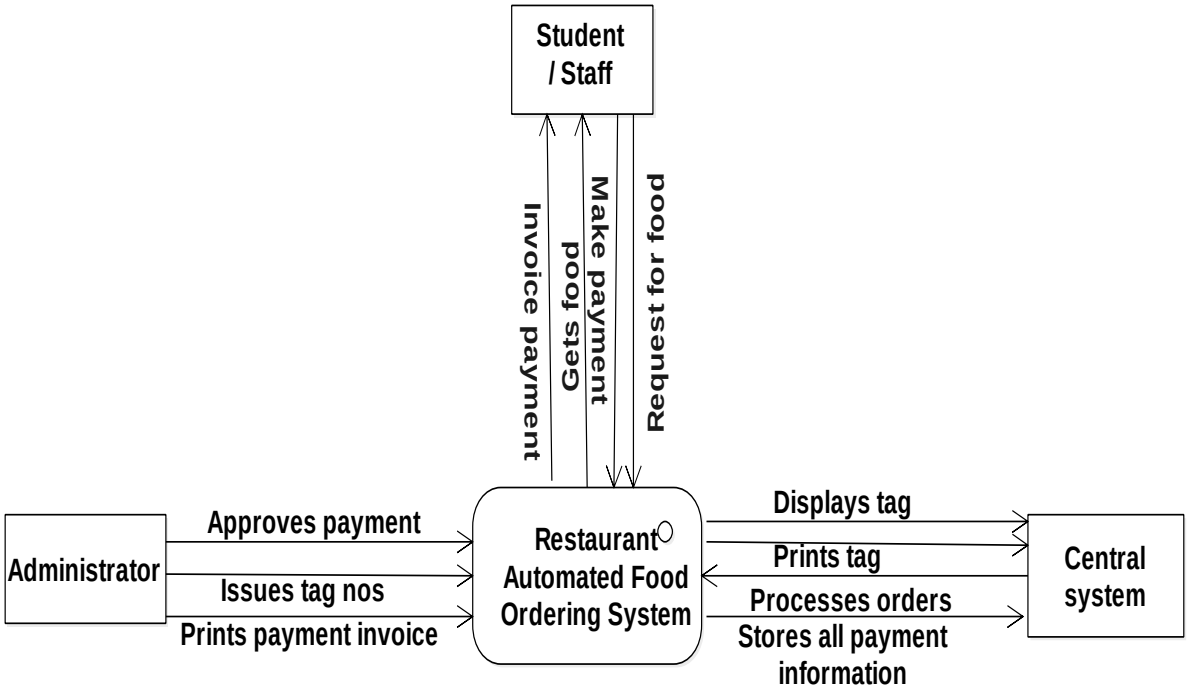


Figure 3.7 Context Diagram

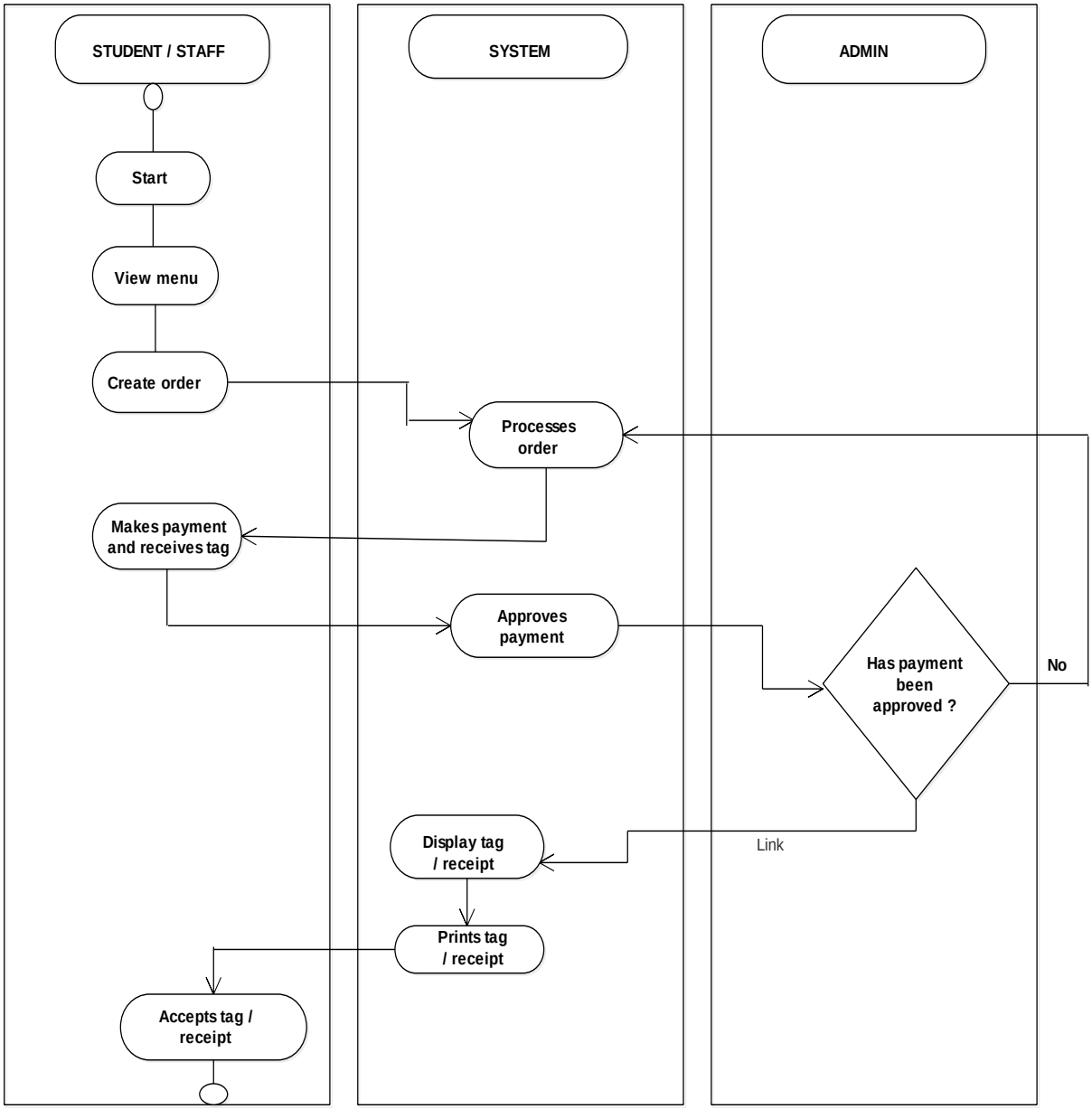


Figure 3.8 Activity diagram

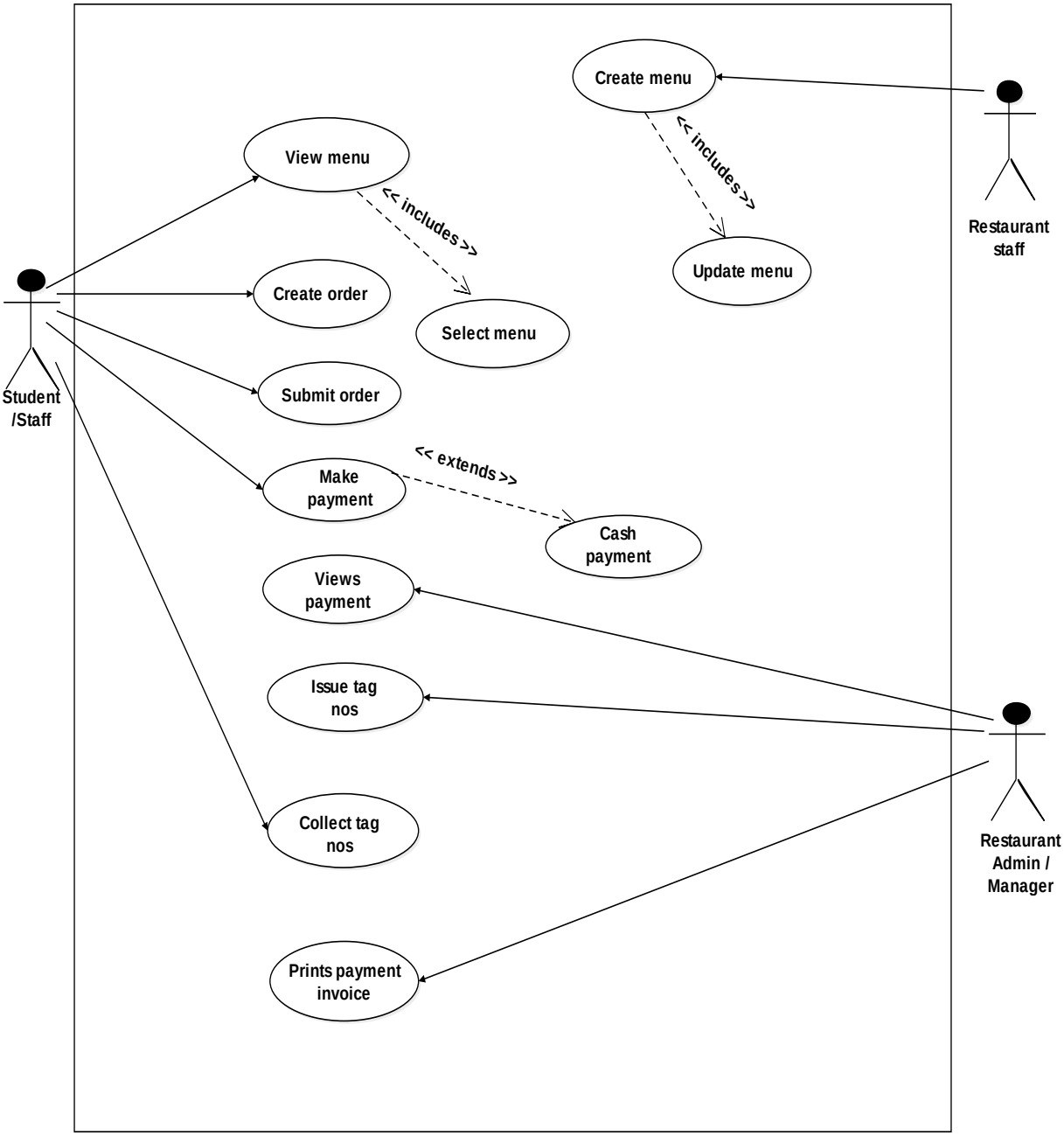


Figure 3.9 Use Case Diagram

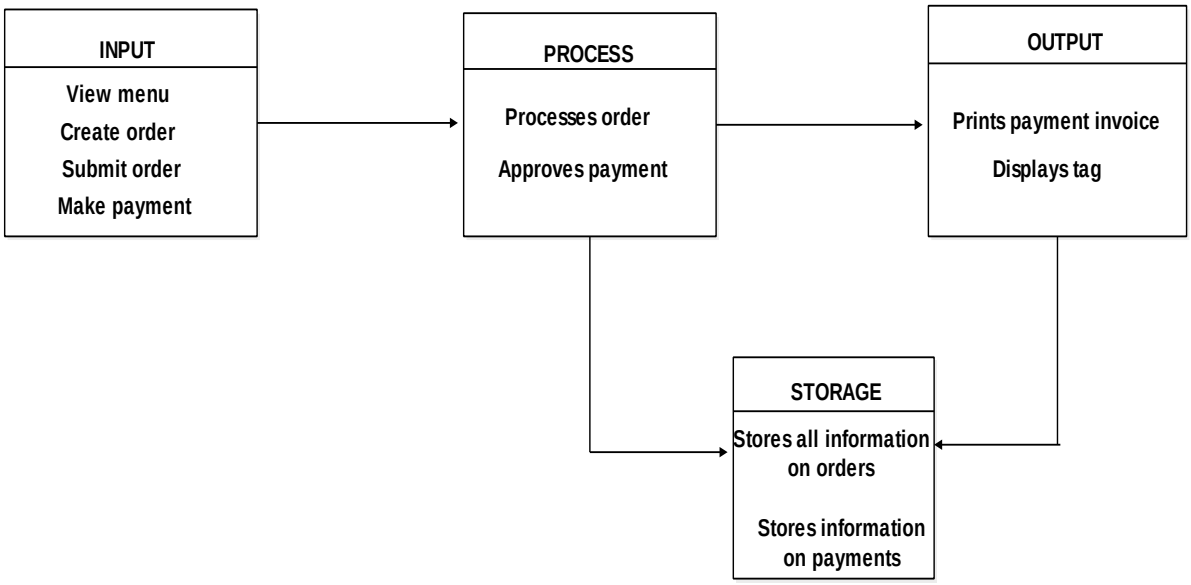


Figure 3.10 Input – Process - Output– Storage Diagram

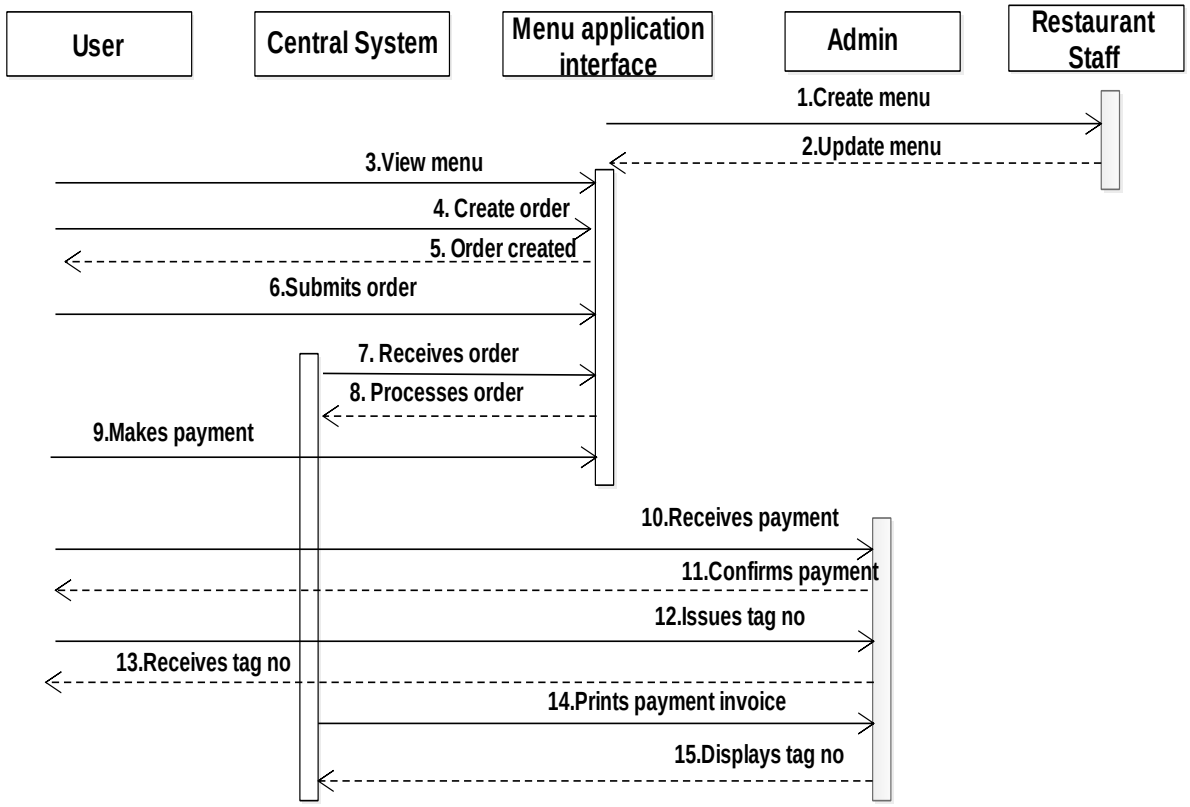


Figure 3.11 Sequence Diagram

The sequence diagram of the proposed system shows the object interactions of the system arranged in sequence of events. This is illustrated above Figure 3.11.

CHAPTER FOUR

SYSTEM IMPLEMENTATION AND TESTING

4.1 Introduction

This chapter looks into the implementation of the design of the Restaurant Automated Food Ordering System (RAFOS). It was designed using JAVA for the mobile application and VB.NET for the server application.

4.2 Graphical User Interface Design

The Graphical User Interfaces of the mobile application in the implemented system are designed to be user-friendly, colourful, and as simple as possible to enable an effective user interaction. The following figures shows the Input – Process and Output form of the mobile application respectively.

Figure 4.1 shows the main menu of the mobile application to proceed the order. The next figure 4.2 shows the input form where the customer registers his details ; The user inputs his name , number and address that will automatically be saved in the database of the server,. This gives the administrator the ability to print out customer's information for any order taken. Figure 4.3 displays a confirmatory message that the customer has registered successfully. The next process as shown in figure 4.4 takes you to the order page where all the categories of food has been made available for any user to choose. It is categorised into; main dishes which entails all the main courses available in the school. Pastries, which are the baked items or snacks available in the school for sale. It also has the drinks section (non-alcoholic), swallow- which comprises of all the solid meals. There are also pre-ordered items, complementary foods and fruits. The pre-ordered items that are those that are not made available on daily basis. It requires a certain amount of time or days before it is made available.

In figure 4.5, each item available has an allocated price to it so the customer is aware of how much he will be paying. The customer has the choice of ordering as many items as possible from any of the categories shown in figure 4.6 and then adds it to the cart and proceeds to checkout. This then takes you to the payment process where the customer will be given a variety of payment options that he can choose from. The payment process is divided into three forms of payment; payment by cash on delivery, POS (Point Of Sale) payment and payment by card. This is shown in figure 4.7

After the payment has been made, it sends the order to the central system handled by the administrator and then issues a tag number to the customer's order shown in figure 4.8. When the order is available, the tag number is displayed. This enables the customer to differentiate his order from another customer's order.

4.3 Administrator Interface Design

The administrator plays no role in the operation of the mobile application but rather is in charge of the central system and the operations on the system. He manages the items and sales off the restaurant.

Figure 4.9 shows the home page viewed by the admin officer. In figure 4.10 the administrator logs in with his username and password that takes him to the admin dashboard shown in figure 4.11 where all the food items are located. The administrator can edit the items in the category. The item categories have the carts of each dish where the admin officer can add more items to the car when any dish is made available or the introduction of a completely new dish into the restaurant.

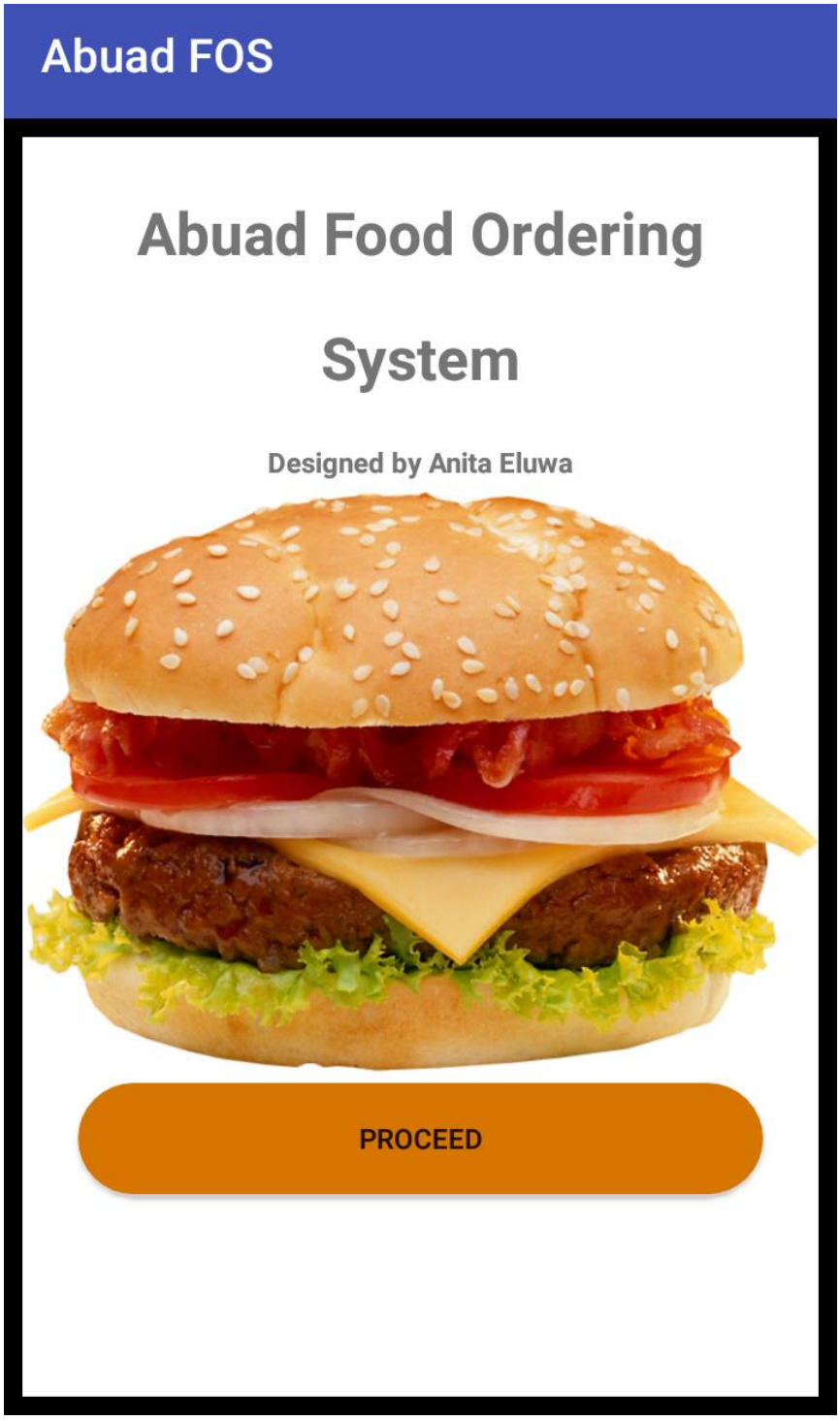


Figure 4.1 Shows the index page of the mobile application

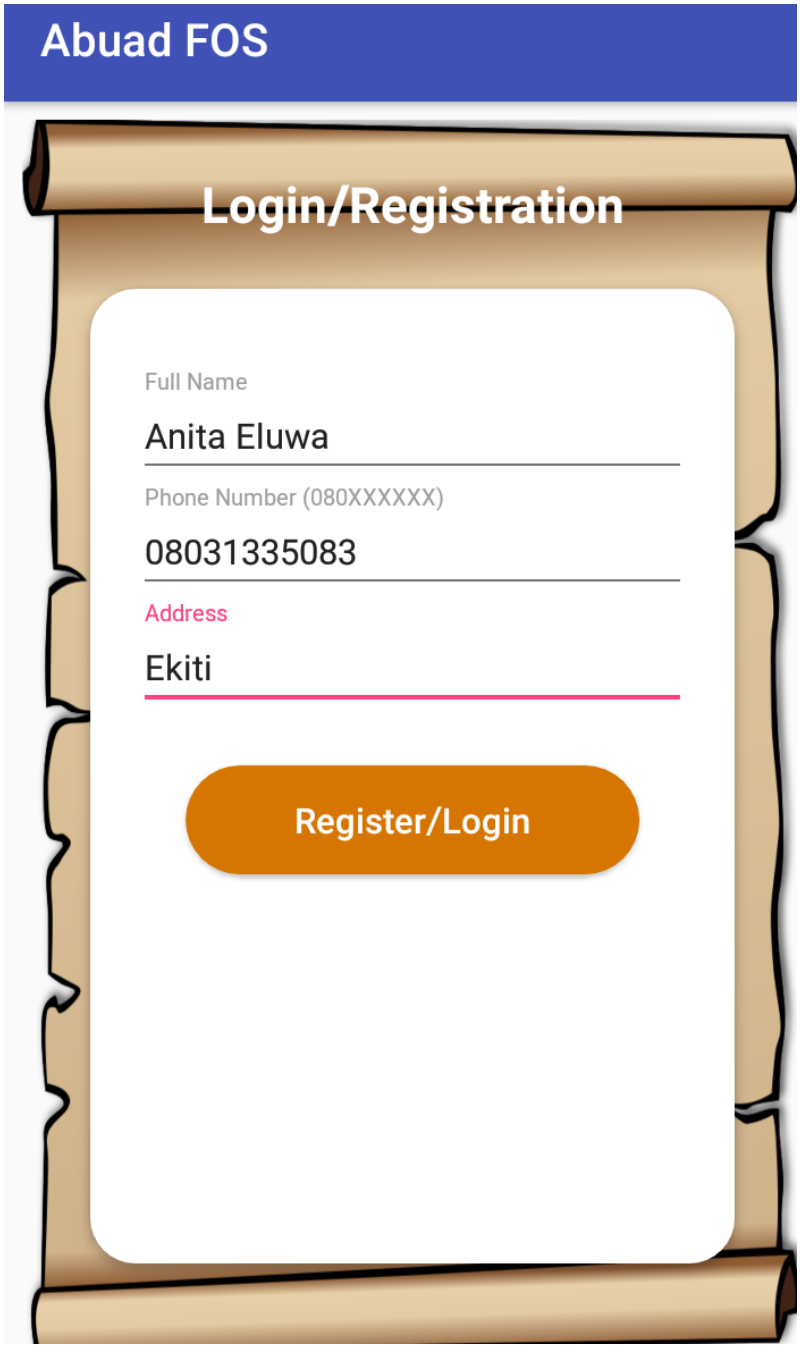


Figure 4.2 Shows the log in interface for the customer

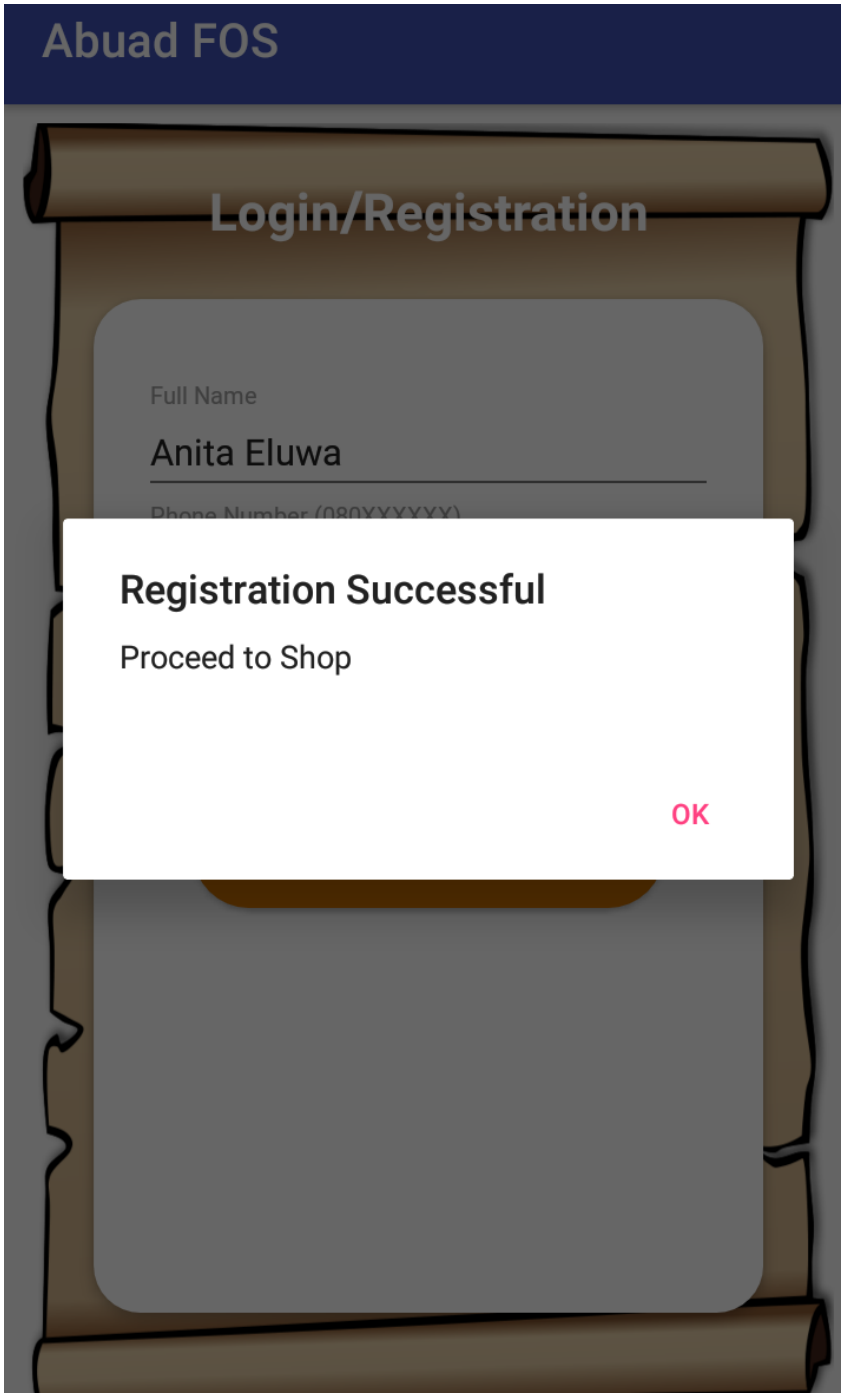


Figure 4.3 Displays a confirmatory message of a successful registration of the customer

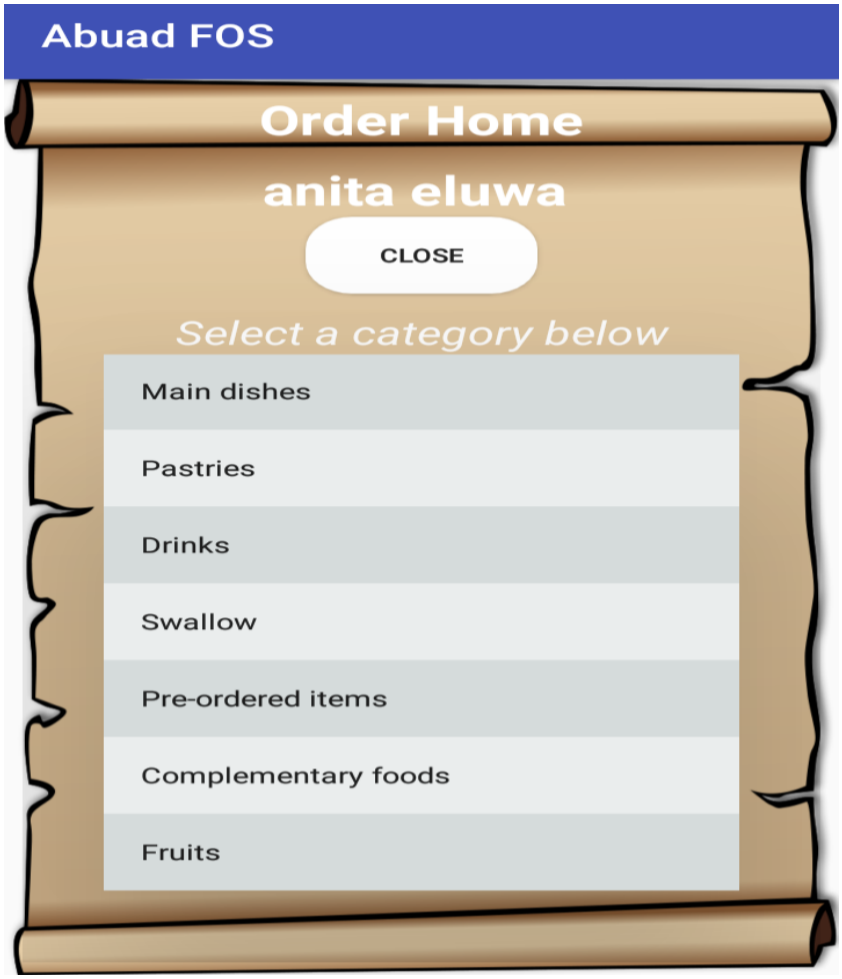


Figure 4.4 Displays the item categories

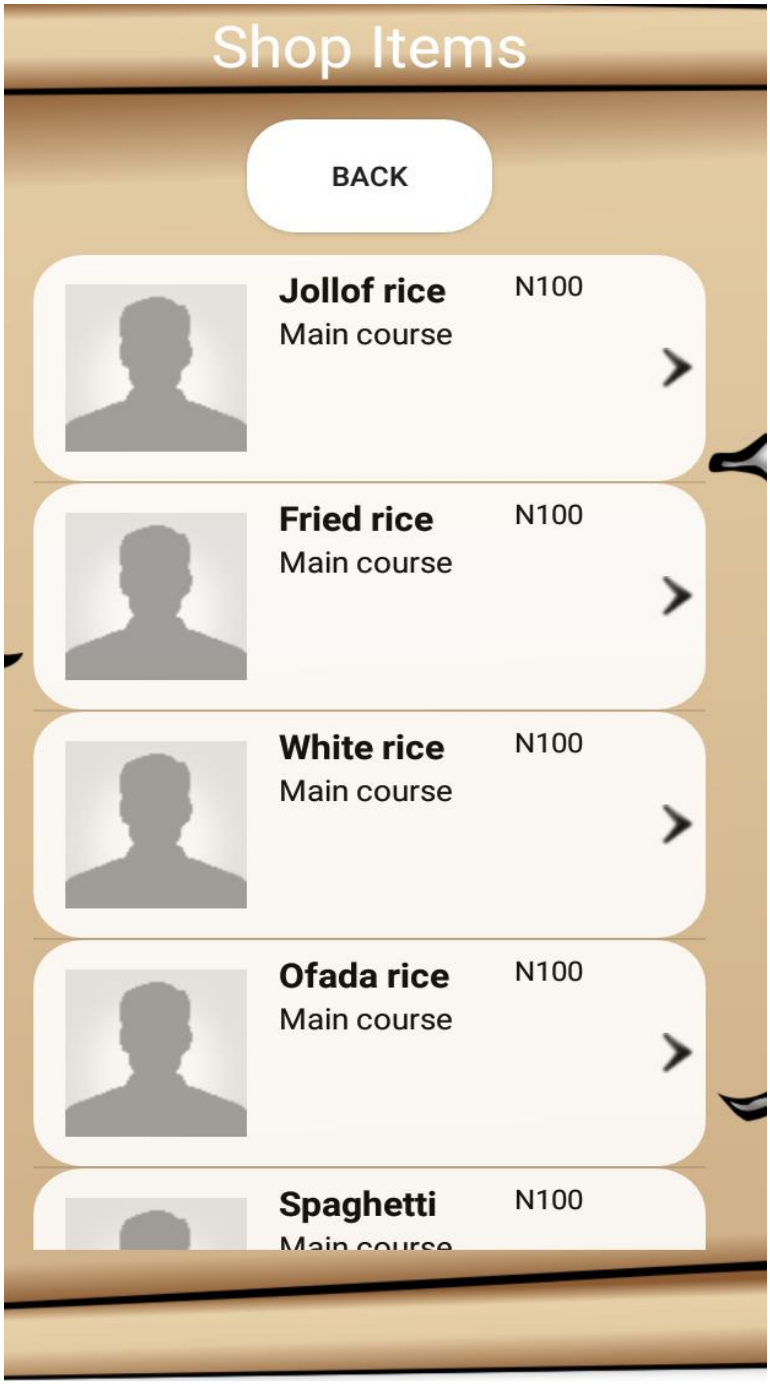


Figure 4.5 shows a sample of the price allocated to each item in the category

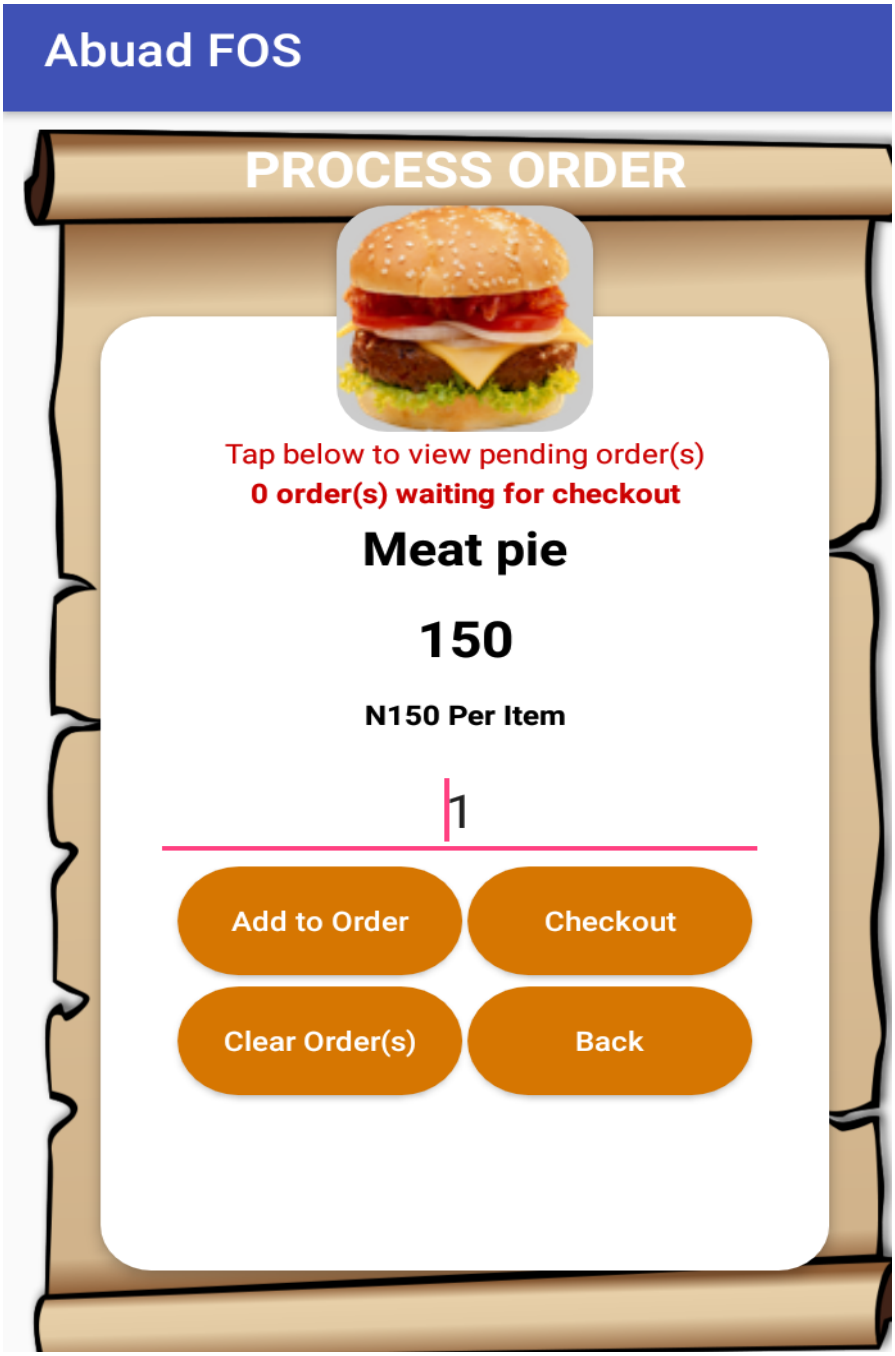


Figure 4.6 Shows if the customer wants to add more items to his order before proceeding to checkout

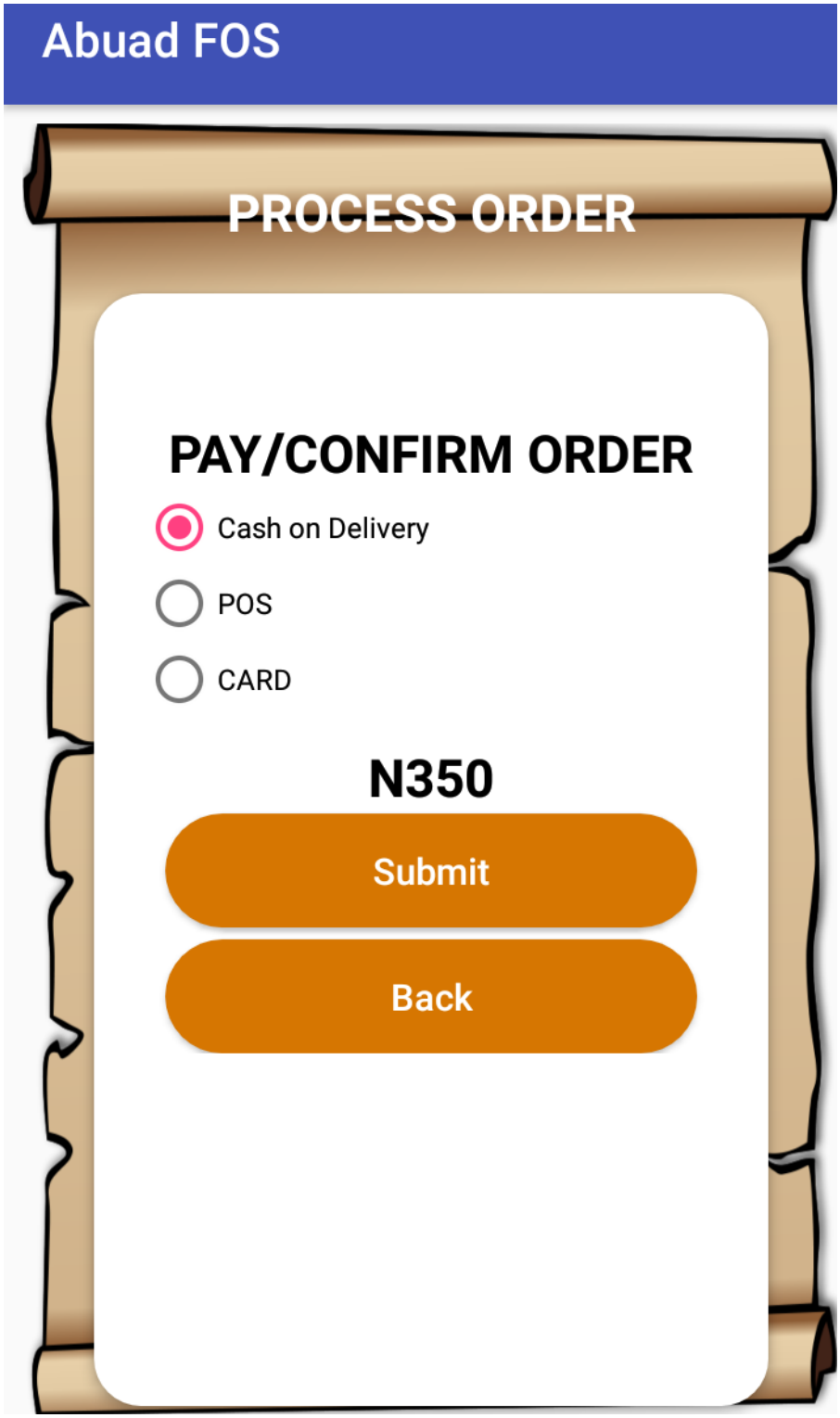


Figure 4.7 Shows the payment processes

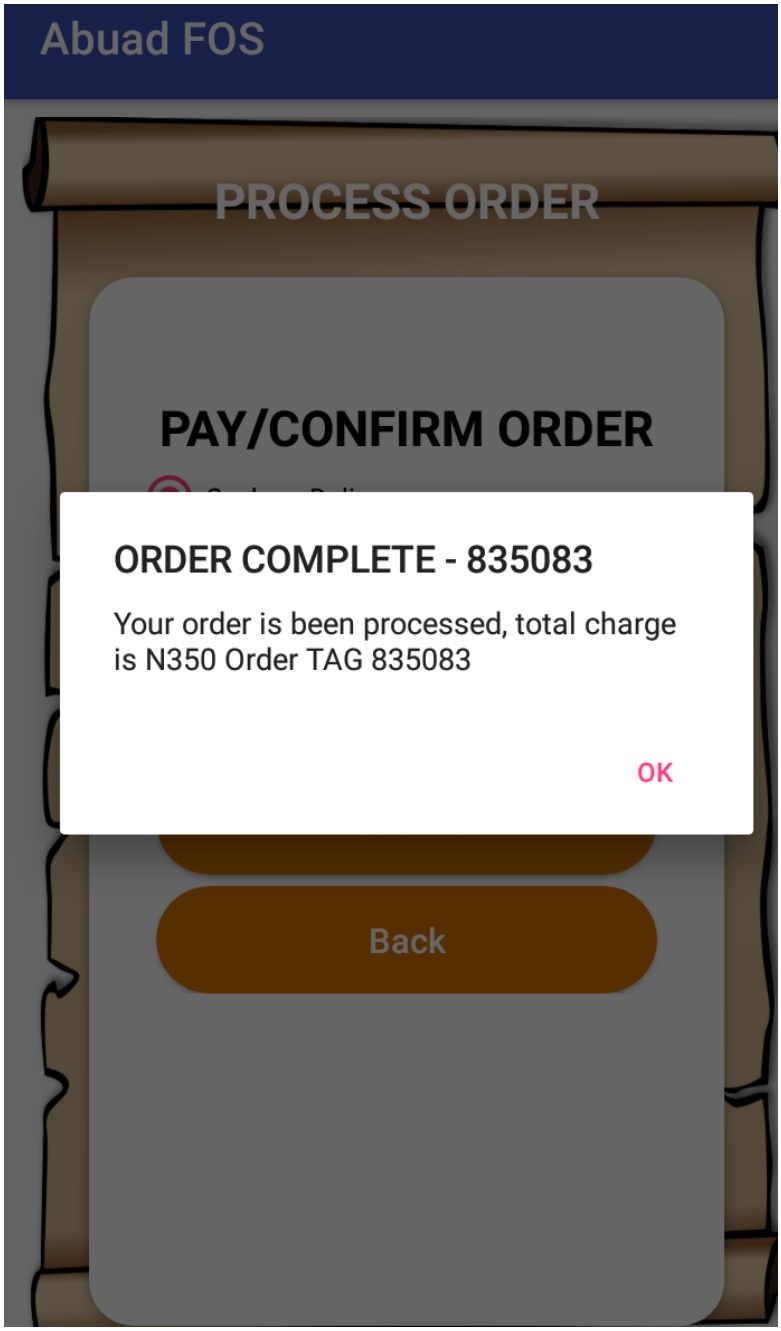


Figure 4.8 Issues a tag number to the customer for his order

Figure 4.9 Home page of the administrator

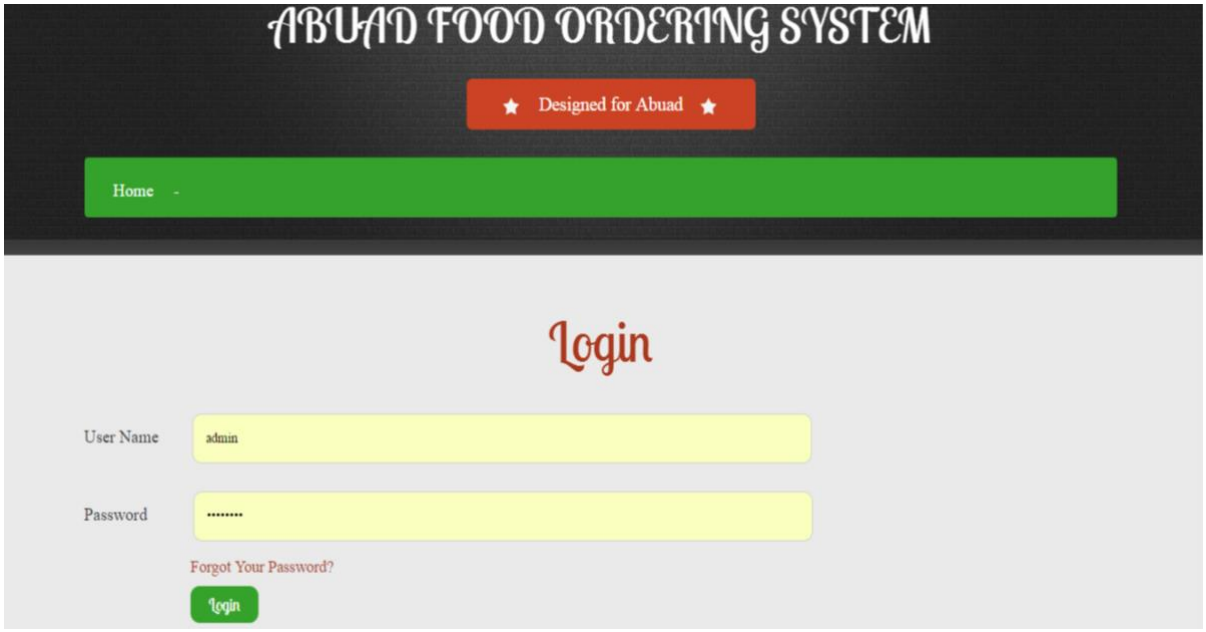


Figure 4.10 The log in page for the administrator

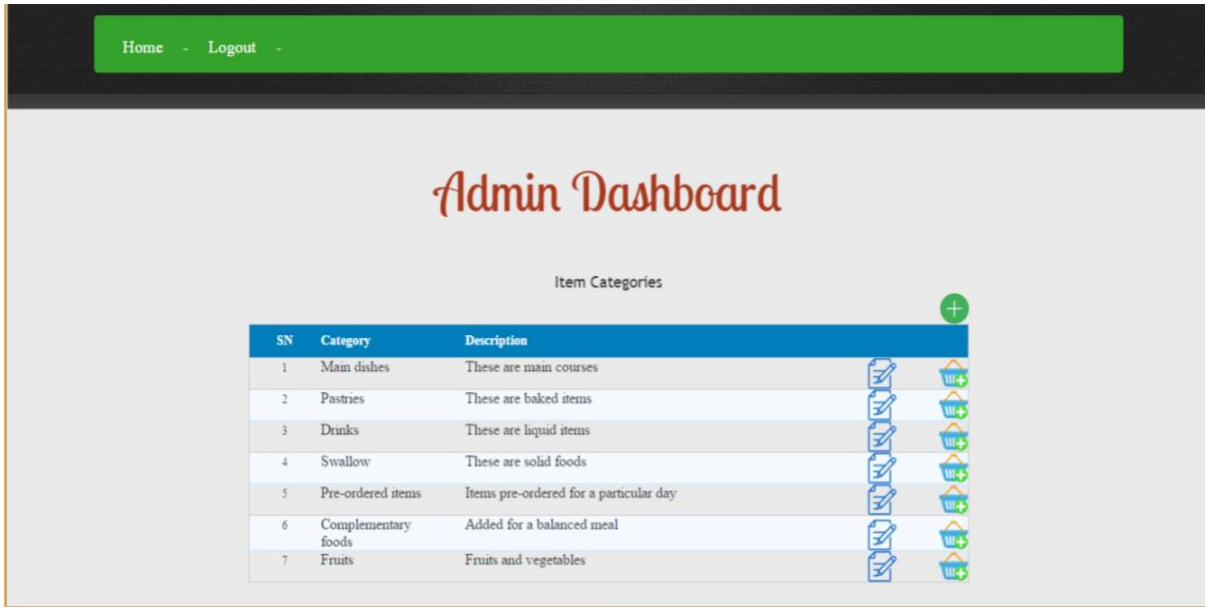


Figure 4.11 Location of the food category

Figure 4.12 shows the details of every customer that has made an order. It shows the name of the customer, number, address, date and also time of order. Figure 4.13 shows the orders into the system that has been processed or treated. It comprises of the type of order made, the quantity and the customer's number. He sees all operations from the mobile device and he ensures that all orders are treated. Treated shows if an order has been processed and delivered or otherwise. If it has, the treated shows a Yes and if it has not, the treated shows a No. When the orders are ready, the system displays the tag number issued to the customer during ordering, hence the customer identifies his order.

4.4. System Testing

The codes were tested at various levels. An android device was provided which served as the tablet for the customer and a Personal Computer (PC) served as the central system where all the details of ordered items made by the customer were stored and processed. Customers provided their details for the database and ordered some items which was received by the central system.

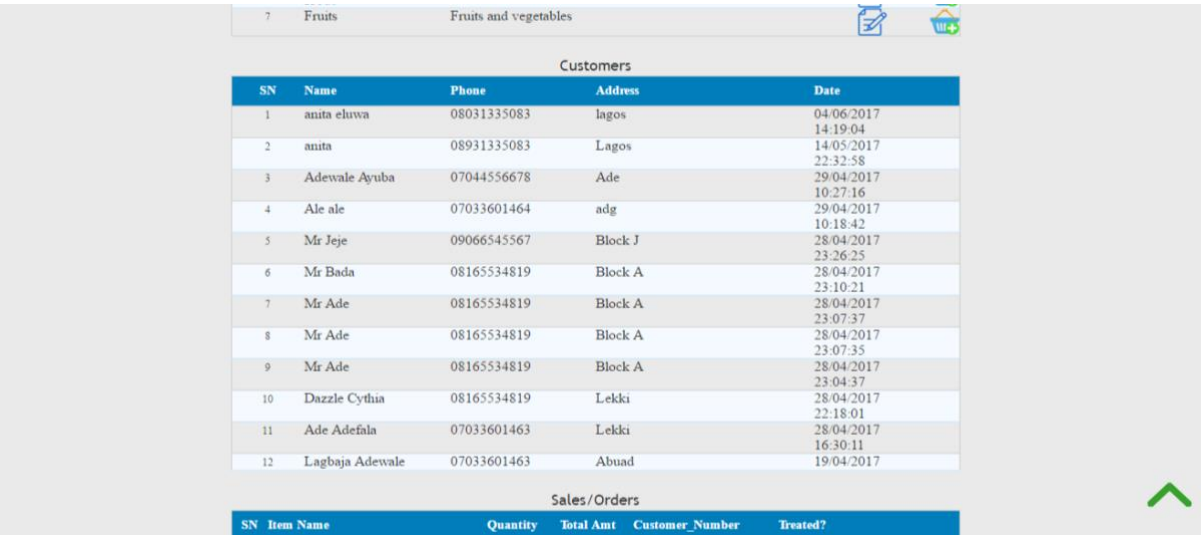
4.5 System Security

The security measures used for this system were to ensure the authorization, authentication, confidentiality, integrity, reliability, and survivability of the system. Hence, the following security measures were followed;

1. The log in interface provided a username and password for the customer as a means of authentication and also to prevent access from unauthorized users.

2. The password to log into the system is encrypted to avoid visibility.

3. The central system provided a database as a means of storage to prevent data loss when there is a crash on the system.



7	Fruits	Fruits and vegetables

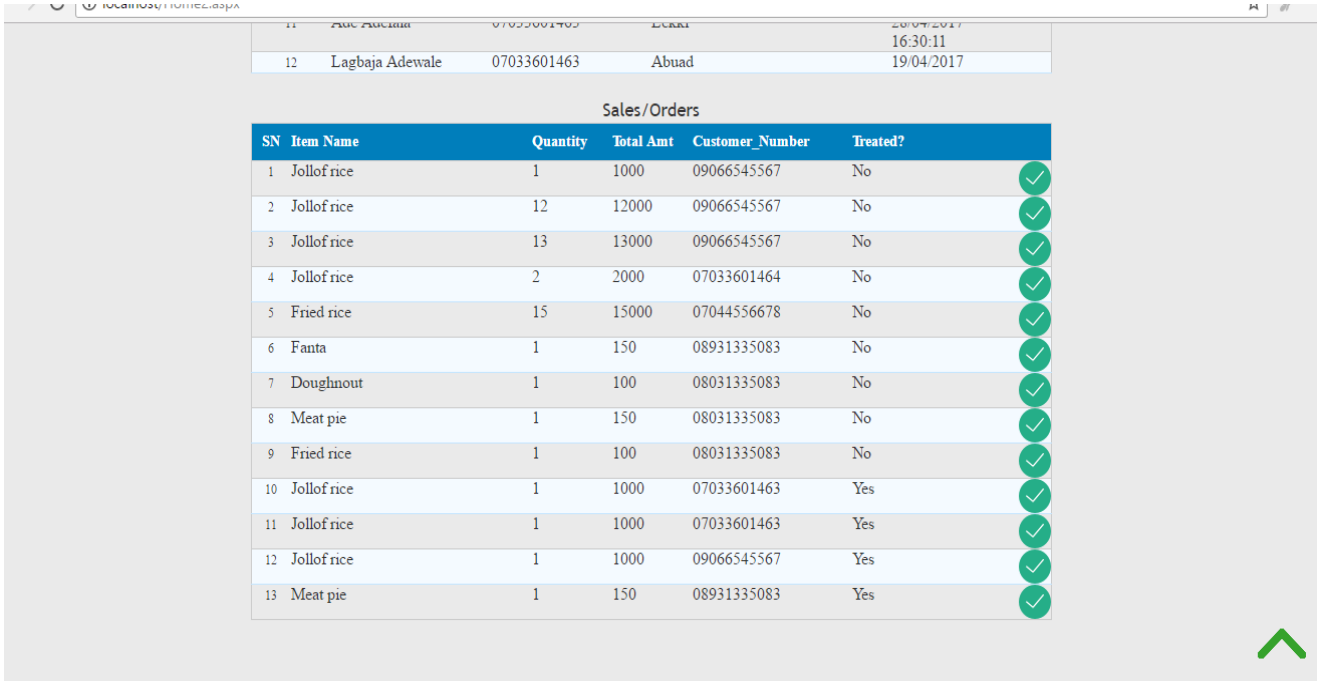
Customers

SN	Name	Phone	Address	Date
1	anita eluwa	08031335083	lagos	04/06/2017 14:19:04
2	anita	08931335083	Lagos	14/05/2017 22:32:58
3	Adewale Ayuba	07044556678	Ade	29/04/2017 10:27:16
4	Ale ale	07033601464	adg	29/04/2017 10:18:42
5	Mr Jeje	09066545567	Block J	28/04/2017 23:26:25
6	Mr Bada	08165534819	Block A	28/04/2017 23:10:21
7	Mr Ade	08165534819	Block A	28/04/2017 23:07:37
8	Mr Ade	08165534819	Block A	28/04/2017 23:07:35
9	Mr Ade	08165534819	Block A	28/04/2017 23:04:37
10	Dazzle Cythia	08165534819	Lekki	28/04/2017 22:18:01
11	Ade Adefala	07033601463	Lekki	28/04/2017 16:30:11
12	Lagbaja Adewale	07033601463	Abuad	19/04/2017

Sales/Orders

SN	Item Name	Quantity	Total Amt	Customer_Number	Treated?

Figure 4.12 Customer's details who have ordered an item



12	Lagbaja Adewale	07033601463	Abuad	16:30:11 19/04/2017

Sales/Orders

SN	Item Name	Quantity	Total Amt	Customer_Number	Treated?
1	Jollof rice	1	1000	09066545567	No
2	Jollof rice	12	12000	09066545567	No
3	Jollof rice	13	13000	09066545567	No
4	Jollof rice	2	2000	07033601464	No
5	Fried rice	15	15000	07044556678	No
6	Fanta	1	150	08931335083	No
7	Doughnout	1	100	08031335083	No
8	Meat pie	1	150	08031335083	No
9	Fried rice	1	100	08031335083	No
10	Jollof rice	1	1000	07033601463	Yes
11	Jollof rice	1	1000	07033601463	Yes
12	Jollof rice	1	1000	09066545567	Yes
13	Meat pie	1	150	08931335083	Yes

Figure 4.13 List of sales and orders

4.6 System Implementation Strategy

A number of implementation strategies from the current system to the proposed system may be employed;

4.6.1 Direct Change Strategy

In direct strategy, the old system is shut down completely and switches over to the new system immediately. This strategy is recommendable and relieves the users of the old system the inefficiencies of the system. However, it may not be the best approach because if there is a problem with the new system, we cannot go back to the old system because it has been completely shut down.

4.6.2 Phase Change Strategy

In this approach, the operations are carried out in phases, first the registration of customers would be implemented with the new system while other operations carried out normally. This is done to ensure that the system is functioning appropriately before it is implemented. This strategy might be slow, but it is more advisable.

4.7 Training

Trained staff are required for any of the implementation strategy to be carried out. This will also increase the efficiency and effectiveness in the various aspects of the system.

4.8 System Maintenance

The system maintenance method employed was the system monitoring method. The system was monitored during and after the implementation by observing and measuring the efficiency of the procedures. The number of sales and payments into the system were monitored, the time taken for each operation and the debugging of codes that would have gone unnoticed that would cause error to the system.

4.9 System Specification

The system specification is divided into; hardware and software specifications.

4.9.1 Hardware Specification

The hardware used in the development of this system includes the following;

Processor: Intel ® Pentium ® CPU N3450

Ram: 4.00 GB

System type: 64-bit operating system x64-based processor

FDD: 1.44MB

Monitor: 14 inches

CD Drive: 3 button Scroll

Keyboard: 108 keys

The above specifications are for the central system. Hence the specifications for the android device includes;

Device: Android tablet

Ram: 2.00 GB

System type: Android OS

Monitor: 10 inches

4.9.2 Software Specification

The software used in the development of this system and to achieve the main goal of the study includes the following;

Operating system: Windows 8

Front end: JAVA

Web servers: Xamp server

Communication protocol: SOAP

Backend: VB.net, MsSQL

The software requires Windows 8 or higher versions of windows operating systems. It also requires Xamp server for database. JAVA and VB.net are the programming languages used.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 Summary

The school has been dealing with the major challenges in depending on the manual system in the six (6) restaurants since the year 2010. The number of staff and students that visits these restaurants daily face difficulties as a result of inefficiencies that the restaurants provides. The automated food ordering system that was designed and implemented in this project eliminates the major challenges in the school restaurants such as: long queues, bill calculation, and improper restaurant management. The designed system includes the user interface where the customer orders his meals and makes payment. Also, the administrator interface where all the orders made by the customer goes into, the payment confirmation and the processing of the order. The main aim of this project is to design and implement a Restaurant Automated Food Ordering System (RAFOS) and the objectives are: To automate food ordering system at ABUAD Restaurant that can eliminate or at least minimize the current problems experienced in traditional (manual) system and to automate the payment process of ordered food at the restaurant.

In the designed system, an Android based mobile application was developed for the user interface of the implemented system, and also a central system which serves as the server for the administrator interface. The methodology used is the Software Analysis and Design Methodology (SADM). The programming language tools used were Visual Basic. (VB.NET), JAVA programming languages, XML, ASP.NET and MsSQL database as front end and backend respectively. The other resources used in this project are: Android Studio Development Kit, MsSQL, logical designs and a communication protocol (SOAP).

5.2 Conclusion

The Restaurant Automated Food Ordering System (RAFOS) is developed to make payment process easier and comfortable for students and staff, to reduce labour among the workers, to make services faster and more efficient and also ensure that all bill calculations and accounting are accurate. However, it will provide quality of services and customer satisfaction.

5.3 Recommendations

Hostel delivery services and table reservations for any customer who is indisposed or cannot make it to the restaurant on time for private reasons may be incorporated in the future. This would also reduce the number of people that will be in the restaurant to order food and create more tables and chairs for customers who chooses to eat at the restaurant.

This system may be extended to register and link multiple restaurants to enhance the dining experience of customers.

A more revised version of the GUI may also be designed to conform to recent technologies in the growing area of human interface and interaction designs.

References

Ackoff, R. (1981). Creating the Corporate Future: Plan or be Planned For. Retrieved [online] from: eu.wiley.com [Date Accessed 1 April 2017].

Bhange , K. , Shinde , T. , Ingale , D. , Solanki , N. and Totare , R. (2015). A proposed system for Touchpad Based Food Ordering System Using Android Application. *International Journal of Advanced Research in Computer Science and Technology*, 3(1) , pp:70-72.

Bhargave , A. , Jadhav , N. , Joshi , A. , Oke , P. and Lahane , S. R. (2013).Digital Ordering System for Restaurant Using Android. *Journal of Scientific and Research Publications* , 3(4) , pp:1-7.

Bhingre , P. , Boddu , T. , Chandak , A. , Devkar , D. and Patil , S. , P. (2015) Digital Dining System using Android. *International Journal of Engineering Research and General Science* , 3(2) , pp:631-636.

Bora , R. , P. and Gupta , E. (2012). Application On Order Management System In Restaurants. *International Journal of Application or Innovation in Engineering & Management (IJAIEM)* , 1(2) , pp:59-62.

Chavan , V. , Jadhav , P. , Korade , S. and Teli , P. (2015). Implementing Customizable Online Food Ordering System Using Web Based Application. *International Journal of Innovative Science, Engineering & Technology* , 2(4) , pp:722-727.

Dhore , V. , B. , Thakar , S. , Kulkarni , P. and Thorat , S. (2014). Digital Table Booking and Food Ordering System Using Android Application. *International Journal of Emerging Engineering Research and Technology* , 3(7) , pp:76-81.

Eng , C. , Mansukhani , S. , Guatam , P. , Ved , H. , Yerabaka , N. and Yerabaka , S. (2007). Restaurant Automation Project. Concepts in Software Engineering, Report 3 – System Design. 14:332:452.

Ford , C. , Mcveigh , S. , Ohia , O. , Taylor , N. and Vansant , A. (2006). Restaurant Automation Design Report, Project Code May 07-14.

Gan (2002) and De Leon (2008) . Design And Implementation Of Online Food Ordering System. Design Project . pp:1-42.

Jakhete , M. , D. and Mankar , P. , C. (2015). Implementation of Smart Restaurant with e-menu Card. *International Journal of Computer Applications* , 119 , pp:1-5.

Kadale , N. , Bansod , P. , Pillai , R. , Sane , S. , Pratape , S. and Pawar , S. (2015). Wireless Customizable Food Recommendation System using Apriori and K-Means Algorithm. 4(X) . Department of Computer Engineering , Savitribai Phule Pune University, NBN SSOE Pune ,Maharashtra, India. pp:97-100.

Kulkarni , H. , Dascalu , S. and Harris , S. (2008). Software Development Aspects of a Mobile Food Ordering System. University of Nevada Reno, Nevada, USA. pp:67-72.

Leah, H. (2015). The Telegraph. Retrieved [online] from: telegraph.co.uk [Date Accessed 1 April 2017].

Lohdi , A. , K. and BaburaOkamble , P. (2006). Automatic Restaurant Order System Using Zigbee. *IOSR Journal of Electronics and communication Engineering (IOSR-JECE)* , pp:19-23.

Noorropidah (2012). Introduction to Automation System. Retrieved [online] from: noorropidah.files.wordpress.com [Date Accessed 1 April 2017].

Oberhammer , T. (2014). Restaurant Menu and Management System. Carleton University, COMP 4905. pp:1-60.

Patel , M. (2015). Online Food Order System for Restaurants. Grand Valley State University School of Computing and Information Systems. pp:1-18.

Pernebayev , M. , Kazakhstan , S. , Singh , T. and Sundararajan , K. (2013). Design, Implementation, and Evaluation of a Menu Management System for Restaurants. pp:1-89.

Reisner, A. (2016). Retrieved [online] from: https://tastingtable.com/dine/national/american-restaurants-food-history. [Date Accessed 1 April 2017].

Samsudin , N. , A. , Khalid , A. , K. , S. , Kohar , M. , A. , F. , M. , Senin , Z. and Ihkasan , N. , M. (2011). A Customizable Wireless Food Ordering System With Real-time Customer Feedback. pp:187-192.

Shinde , R. , Thakare , P. , Dhomne , N. and Sakar , S. (2014). Design and Implementation of Digital dining in Restaurants using Android. *International Journal of Advance Research in Computer Science and Management Studies* , 2(1) , pp:379-384.

South African cuisine (2017). Retrieved [online] from: en.m.wikipedia.org [Date Accessed 1 April 2017].

Swapna , V. and Alikhan , M. , F. (2012). Design And Implementation Of Ordering System For Restaurants. *International Journal of Engineering Research & Technology (IJERT)* 1(10) , pp:1-5.

Tanpure , S. , S. , Shidankar , P. , R. and Joshi , M. , M. (2013). Automated Food Ordering System with Real-Time Customer Feedback. *International Journal of Advanced Research in Computer Science and Software Engineering* , 3(2) , pp:220-225.

Tokyo restaurant. Retrieved [online] from: studymode.com [Date Accessed 1 April 2017].

Webster, M. (1828). Retreived [online] from: https:/www.merriam-webster.com [Date Accessed 1 April 2017].

Weinberg, G. (2007). An Introduction to General Systems Thinking. Retrieved [online] from: goodreads.com [Date Accessed 1 April 2017].

Zakaria , B. , I. , N. April (2010). Food Ordering System Using Mobile. Bachelor of Computer Science (Software Engineering), faculty of Computer Systems and Software Engineering University Malaysia Pahang. pp:1-24.

APPENDIX A

S/N	NAME	TOPIC	METHOD	IMPLEMENTATION	RESULTS	AREAS OF FURTHER STUDY
1.	Bhandge *et al.*(2015)	Automated food ordering system	Algorithms 1.K-means clustering 2.Apriori	Android applications for tablet PCs	Touchpad based food ordering system using android application	Lack of multiple means of payment platform
2.	Tanpure *et al.*(2013)	Automate food ordering process in restaurants	Java programming language	Android applications for user mobile	Automated food ordering system with Real-time customer feedback	Lack of multiple means of payment platform
3.	Gan *et al.*(2002)	Online food ordering system	Waterfall model	Electronic payment system	Design and Implementation of online food ordering system with the use of e-payment	Lack of trained staff
4.	Bhargave *et al.*(2013)	Digital Ordering system for restaurant	Network based Smart Order System through RS-232 signal	Android applications for tablet PCs	Digital ordering system for restaurants using Android	Include table reservation , home delivery , billing
5.	Lodhi *et al.*(2006)	Automatic restaurant order system	Zigbee module	Mechanical, electronics and computer based system	Automatic restaurant order system using zigbee	
6.	Patel ,M.(2015)	Online food order system	Web application using JSP and HTML	Web ordering system module	Online food order system for small scale business	Allow customers to customize food orders
7.	Ford *et al.*(2006)	Restaurant automation design	Java programming language and mysql	Wireless handheld ordering system	Restaurant automation system for handheld devices	Set up a vpn server on the central computer to enable wireless communication.

8.	Eng *et al.*(2007)	Restaurant automation project	3-tier architecture	Touch screen wireless device	Restaurant automation system using a touch screen wireless device	Better implementation of the GUI for the PDA
9.	Oberhammer ,T.(2014)	Restaurant menu and management system	UI and Wifi direct	Mobile devices for android	Restaurant menu and management system for android devices	Restaurant statistics should be implemented
10.	Pernebayev *et al.*(2013)	Design , implementation and evaluation of a menu management system	Six Sigma methodology and Mintab tool	Mobile application	Design , implementation and evaluation of a menu management system for mobile application	Long list of menu items should be avoided
11.	Samsudin *et al.*(2011)	Customizable wireless food ordering system with real-time customer feedback	System architecture , Context free diagram , ER diagram	Mobile application for smart phones	Design and implementation of a customizable wireless food ordering system with real-time customer feedback	Addition of inventory management module for the kitchen staff
12.	Kadale *et al.*(2015)	Wireless Customizable Food Recommendation System	Apriori and K-means algorithm	Android based touchpad	Automation for Kitchen order ticket (KOT), billing and Customer Relation Management (CRM)	Extension of the existing system to a chain of multiple restaurants
13.	Bora *et al.*(2012)	Application On Order Management System in Restaurants	Use Case Diagram	Android tablets	Wireless food ordering system	Addition of online booking of tables and ordering of food items
14.	Bhingre *et al.*(2015)	Digital Dining System using Android	Mathematical model , Architecture diagram	Android application	Design and implementation of android application	Security should be maintained

15.	Dhore *et al.*(2014)	Digital Table Booking and Food Ordering System Using Android Application	Dijkstra's Algorithm Pseudocode	Android devices	Design and implement a remote food ordering system	
16.	Charvan *et al.*(2015)	Implementing Customizable Online Food Ordering System Using Web Based Application	System data flow diagram	Android mobile	Food pre-order system using web based application	
17.	Kulkarni *et al*. (2008)	Software Development Aspects of a Mobile Food Ordering System	Use case diagram , Layered architecture	Pocket pc	The Mobile Food Ordering System (MFOS)	The user should be able to give special instructions for his or her order.
18.	Shinde *et al.*(2014)	Design and Implementation of Digital dining in Restaurants	Modules	Android mobile devices	Design and Implementation of Digital dining in Restaurants using Android	
19.	Swapna *et al.*(2012)	Design And Implementation Of Ordering System For Restaurants	Zigbee technology	Touch screen order device	Designing and implementing completely automated menu system in restaurants	
20.	Jakhete *et al.*(2015)	Implementation of Smart Restaurant with e-menu Card	Keil µ Vision4 software	Smart phones/Android tablet	Designing completely automated menu in restaurants with the help of Android mobile phone.	Implementation with Graphical LCD for displaying the menu as we have used android phone

APPENDIX B

FOR MOBILE

HomeActivity.java

```
package com.abuad.abuadshop;

import android.content.Context;
import android.content.SharedPreferences;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.app.ActivityOptions;
import android.content.DialogInterface;
import android.content.Intent;
import android.graphics.Color;
import android.os.AsyncTask;
import android.support.v7.app.AlertDialog;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.ksoap2.SoapEnvelope;
```

```
import org.ksoap2.serialization.PropertyInfo;
import org.ksoap2.serialization.SoapObject;
import org.ksoap2.serialization.SoapPrimitive;
import org.ksoap2.serialization.SoapSerializationEnvelope;
import org.ksoap2.transport.HttpTransportSE;

import java.util.ArrayList;
import java.util.Timer;



public class HomeActivity extends AppCompatActivity {
    private final String NAMESPACE = "http://tempuri.org/";
    private final String URL = "http://192.168.149.1/abuadshop.asmx";
    private final String SOAP_ACTION = "http://tempuri.org/";
    private String TAG = "OM";
    private static String responseJSON;
    Integer nodays;
    ListView lstCategories;
    Button BtnLogout;
    TextView txtname;
    String getmyuserFK;
    String getmyuserTOKEN;
    private Timer timer;
    private static String neededCategory_ID;


    Integer sItemPosition;
    private JSONArray arrayresultformyCategories;
    private ArrayList<String> myCategoriesarraylist;
    ProgressBar pg;
```

```
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);
        final SharedPreferences sharedPref =
this.getSharedPreferences("com.abuad.abuadshop.PREFERENCE_FILE_KEY",
Context.MODE_PRIVATE);
        myCategoriesarraylist = new ArrayList<String>();
        lstCategories =(ListView)findViewById(R.id.lstcategories);
        txtname = (TextView) findViewById(R.id.txtname);
        BtnLogout = (Button) findViewById(R.id.btnlogout);
        BtnLogout.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                //Redirect Splash Page
                //SharedPreferences.Editor editor = sharedPref.edit();
                //editor.putInt("IsRegistered", 0);
                //editor.commit();
                Intent myintent = new Intent(HomeActivity.this, SplashActivity.class);
startActivity(myintent);
            }
        });
        lstCategories.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> arg0, View arg1, int arg2,
                                    long arg3) {
                sItemPosition=arg2;
                getneededvalues(sItemPosition);
                SharedPreferences.Editor editor = sharedPref.edit();
                editor.putInt("neededCategory_ID", Integer.valueOf(neededCategory_ID));
                editor.commit();
                Intent myintent = new Intent(HomeActivity.this, MainActivity.class);
```

```
            startActivity(myintent);
            }
        });

        String UserName = sharedPref.getString("UserName","Shoper");
        txtname.setText(UserName);

        JSON_ReturnAllCategory task = new JSON_ReturnAllCategory();
        task.execute();

    }

    private class JSON_ReturnAllCategory extends AsyncTask<String, Void, Void> {
        @Override
```

FOR SERVER

****

HOME.ASPX

```
<%@ Page Language="vb" AutoEventWireup="false" CodeBehind="Home2.aspx.vb"
Inherits="AbuadShopWebApplication.Home2" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html>
<head>
<title>Food Shop</title>
<link href="css/bootstrap.css" rel="stylesheet" type="text/css" media="all" />
<link href="Portals/0/css/Style.css" rel="stylesheet" type="text/css" />
<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script src="js/jquery.min.js"></script>
<!-- Custom Theme files -->
<!--theme-style-->
<link href="css/style.css" rel="stylesheet" type="text/css" media="all" />
<!--//theme-style-->
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="keywords" content="Food shop Responsive web template, Bootstrap Web Templates,
Flat Web Templates, Andriod Compatible web template,
Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG, SonyErricsson,
Motorola web design" />
<script type="application/x-javascript"> addEventListener("load", function() {
setTimeout(hideURLbar, 0); }, false); function hideURLbar(){ window.scrollTo(0,1); }
</script>
<!--fonts-->
<link href='http://fonts.googleapis.com/css?family=Rokkitt:400,700' rel='stylesheet'
type='text/css'>
```

```
<link href='http://fonts.googleapis.com/css?family=Lobster+Two:400,400italic,700,700italic'
rel='stylesheet' type='text/css'>
<!--//fonts-->
<script type="text/javascript" src="js/move-top.js"></script>
<script type="text/javascript" src="js/easing.js"></script>
<script type="text/javascript">
    jQuery(document).ready(function ($) {
        $(".scroll").click(function (event) {
            event.preventDefault();
            $('html,body').animate({ scrollTop: $(this.hash).offset().top }, 1000);
        });
    });
                                    </script>

    </head>
<body>
        <form id="form1" runat="server">
        <!--header-->
        <div class="header-in">
                <div class="container">
                        <div class="logo">
                                <h1><a href="index.html">Food Shop</a></h1>
                        </div>
                        <div class="header-top">
                                <div class="header-top-in">
                                        <ul class="header-grid">

                                        </ul>
                                        <div class="act">
                                                <a href="#" class="done">Designed for Abuad</a>
                                        </div>
                                <div class="clearfix"> </div>
                                </div>
                                <ul class="grid-header">

                                </ul>
                                <div class="clearfix"> </div>
                        </div>
                        <!---->
                        <div class="header-bottom">
                                <div class="top-nav">
                                        <span class="menu"> </span>
                                        <ul>
                                                <li  ><a href="index.html" >Home  </a><label>-
</label> </li>
                          <li  ><a href="index.html" >Logout  </a><label>- </label> </li>
                                        </ul>
                                        <!--script-->
                                <script>
                                    $("span.menu").click(function () {
                                        $(".top-nav ul").slideToggle(500, function () {
                                        });
                                    });
                        </script>
                                </div>

                        <div class="clearfix"> </div>
                </div>

                </div>
        </div>
        <!---->
        <div class="container">
                <div class="account">
                <h2>Admin Dashboard</h2>
```

```
        <table class="nav-justified" id="GeneralTable">
            <tr>
                <td align ="center">
                    <asp:Label ID="lblerrormsg" runat="server" Font-Bold="True"
ForeColor="#FF3300"
                        Text=""></asp:Label>
                </td>
            </tr>
            <tr>

                <td align ="center"><table runat="server" id="tblfirstview" width
="70%"><tr>
                    <td align ="center" style="font-family: 'Trebuchet MS'; font-size:
larger">Item Categories</td></tr><tr><td align ="right" >
                    <asp:ImageButton ID="btnAddNewCategory" runat="server"
ImageUrl="~/images/add.png"
                        ToolTip="Add New Category" />
                    </td></tr><tr><td align ="center" ><asp:DataGrid ID="dgCategory"
runat="server"
                            AllowPaging="True" AutoGenerateColumns="False"
                            DataKeyField="ID" Width="100%" CssClass="gridBorder"
ShowFooter="false" GridLines="Horizontal"
                            CellPadding="3" CellSpacing="3">
                            <AlternatingItemStyle CssClass="gridRowAlt" />
                            <HeaderStyle CssClass="gridHeader" Font-Bold="True"
HorizontalAlign="Center" />
                            <ItemStyle Height="20px" VerticalAlign="Middle"
CssClass="gridRow" />
                            <Columns>
                                <asp:TemplateColumn HeaderText="SN">
                                    <HeaderStyle Font-Bold="True" HorizontalAlign="Center"
/>

                                    <ItemStyle HorizontalAlign="Center" Width="10%" />
                                    <ItemTemplate>
                                        <%# (dgCategory.PageSize *
dgCategory.CurrentPageIndex) + Container.ItemIndex + 1%>
                                        <asp:Label ID="lblID" runat="server"
Visible="false" Text='<%#DataBinder.Eval(Container,"DataItem.ID")%>'>
                                        </asp:Label>
                                    </ItemTemplate>
                                </asp:TemplateColumn>
                                <asp:TemplateColumn HeaderText="Category">
                                    <HeaderStyle HorizontalAlign="Left" Font-Bold="True" />
                                    <ItemStyle HorizontalAlign="Left" Width="20%" />
                                    <ItemTemplate>
                                        <asp:Label ID="lblCatname" runat="server"
Visible="True" Text='<%#DataBinder.Eval(Container,"DataItem.Category_Name")%>'>
                                        </asp:Label>
                                    </ItemTemplate>
                                </asp:TemplateColumn>

                                  <asp:TemplateColumn HeaderText="Description">
                                    <HeaderStyle HorizontalAlign="Left" Font-Bold="True" />
                                    <ItemStyle HorizontalAlign="Left" Width="50%" />
                                    <ItemTemplate>
                                        <asp:Label ID="lblCatDesc" runat="server"
Visible="True" Text='<%#DataBinder.Eval(Container,"DataItem.Description")%>'>
                                        </asp:Label>
                                    </ItemTemplate>
                                </asp:TemplateColumn>

                                <asp:TemplateColumn HeaderText="">
                                    <HeaderStyle HorizontalAlign="right" Font-Bold="True"/>
```

APPENDIX C

SYSTEM DOCUMENTATION

Server Installation requirements:

1. IIS (internet information service) must be installed
    a. Follow the IIS installation guide from Microsoft
2. Ms sql server 2014 and above must be installed
    a. Follow installation guide from Microsoft
3. Copy website to IIS root folder
4. Run database script with SQL management studio
5. Launch site by typing localhost in any browser
6. Setup hotspot connection from the system (You can use third party tools like connectify)

Mobile Application:

1. Install the application on any compactible android device
2. Connect the device to the system via the hotspot
3. Run the application