

RESTAURANT MANAGEMENT SYSTEM



*The Project submitted to
Sant Gadgebaba Amravati University, Amravati
Towards partial fulfilment of the Degree of
Bachelor of Engineering
In
Information Technology*

Guided by

Prof. F I Khandwani

Submitted by

**Pratik Lanke
Pratik Harne
Adesh Adhao
Anuj Sapkal**

**DEPARTMENT OF INFORMATION TECHNOLOGY
SHRI SANT GAJANAN MAHARAJ COLLEGE OF
ENGINEERING, SHEGAON (M.S.)**

2022- 2023

**SHRI SANT GAJANAN MAHARAJ COLLEGE OF
ENGINEERING, SHEGAON**



2022-2023

CERTIFICATE

This is to certify that **Mr. Pratik Lanke, Mr. Pratik Harne, Mr. Adesh Adhao, Mr Anuj Sapkal**, students of final year B.E. (Information Technology) in the year 2022-2023 of the Information Technology Department of this institute have completed the project work entitled “**Restaurant Management System**” based on syllabus and has submitted a satisfactory account of his/her work in this report which is recommended for the partial fulfilment of the degree of Bachelor of Engineering in Information Technology.

Prof. F I Khandwani
(Project Guide)

Dr. A S Manekar
Head of the Department
SSGMCE, Shegaon

Dr. S. B. Somani
Principal
SSGMCE, Shegaon

**SHRI SANT GAJANAN MAHARAJ COLLEGE OF
ENGINEERING, SHEGAON**



2022-2023

CERTIFICATE

This is to certify that the project work entitled “**Restaurant Management System**” submitted by **Mr. Pratik Lanke, Mr. Pratik Harne, Mr. Adesh Adhao, Mr Anuj Sapkal**, students of final year B.E. (Information Technology) in the year 2022-2023 of the Information Technology Department of this institute, is a satisfactory account of his work based on the syllabus which is approved for the award of the degree of Bachelor of Engineering in Information Technology.

Internal Examiner

External Examiner

Date:

Date:

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of our project. All that we have done is only due to such supervision and assistance and we would not forget to thank them.

*We owe our deep gratitude to our project guide **Prof. F.I.Khandwani**, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.*

*We also wish to extend our sincere thanks to **Prof. F.I.Khandwani**, Project-In-Charge for his valuable support and comments throughout the project.*

*We also wish to thanks **Dr. A. S. Manekar**, Head of the Information Technology Department, Shri Sant Gajanan Maharaj College of Engineering, Shegaon for providing us with a variety of opportunities and inspiration and timely support to bring together professional knowledge that made us consistent performer.*

*We would also like to extend our thanks to **Dr. S. B. Somani**, Principal, Shri Sant Gajanan Maharaj College of Engineering, Shegaon for proving us the infrastructure and facilities without which it was impossible to complete this work.*

We are also thankful to all Teaching and non-teaching staffs of Department to get constant encouragement, support and guidance which helped us in successfully completing our project work.

Student Names

1 Pratik Lanke

2 Pratik Harne

3 Adesh Adhao

4 Anuj Sapkal

ABSTRACT

The project focuses on developing an Android application that allows customers to easily book tables and menus of their choice at a restaurant according to their convenience. The traditional manual system for table reservations in restaurants is becoming obsolete, and digital systems are gaining popularity. The manual system relies on waiters and booking diaries, with no automated record-keeping. Main objective of the project is creating the Reservation System that addresses the shortcomings of the manual system, such as inefficiencies in time and cost management. The current generation prefers high-tech services, especially those available online. Therefore, the project aims to efficiently automate restaurant operations for owners. The proposed reservation system offers customers the convenience of reserving tables, menus, or both. For menu bookings, customers are required to pay 50% of the total amount in advance. Customers can cancel their bookings if they are unable to arrive on time, and can rebook for the next available time slot, provided that the cancellation is made at least 30 minutes before the selected time. However, if the cancellation is made after this timeframe, the advance payment will not be refunded.

Keywords: Restaurant Management, Recommendation, Tablet, Menu, Intelligent, Android application.

TABLE OF CONTENTS

Chapter	Title	Page no.
1	Introduction	1
	1.1 Chapter Overview	1
	1.2 The Problem	1
	1.3 Project Objectives	1
	1.4 Existing Solutions	2
	1.5 Project Proposal	3
	1.6 Summary of Chapter	4
	1.7 Commonly Used Words	4
	1.8 Closing Remarks	5
2	Background	6
	2.1 Chapter overview	6
	2.2 Point of Sale (POS) System	6
	2.3 Existing Point of Sale System	6
	2.4 Platform Choice	7
	2.5 Software Choice	7
	2.6 UML	8
	2.7 Requirement Gathering	8
	2.8 Development Methodology	9
	2.9 Chapter Summary	10
3	Literature Review	11
	3.1 Introduction	11
	3.2 Theoretical and Conceptual Development	11
	3.3 Inter Organization Case Study	12
	3.3.1 Peace House Restaurant	12
	3.4 Compare Between Existing Systems	13
	3.4.1 Current System	13
	3.4.2 New System	14
	3.5 Chapter Summary	15
4	Requirement Analysis	15
	4.1 Chapter Overview	15
	4.2 Stakeholder Identification	15
	4.3 Use Cases	15
	4.4 Features	16
	4.5 Measurable Goals and requirements	17
	4.5.1 Functional Requirements	19
	4.5.2 Non- Functional Requirements	20
	4.6 Chapter Summary	22
5	Design	23
	5.1 Chapter Overview	23
	5.2 Introduction	23

5.3	Component Diagram	23
5.4	Data Storage	23
5.5	Relational database Management System(RDBMS)	24
5.6	Extensible Markup Language(XML)	25
5.6.1	Storage Method Chosen	25
5.6.2	Normalization	26
5.6.3	Entity Relationship Diagram	27
5.6.4	Database Design Concepts	28
5.7	Graphical User Interface	31
5.7.1	Order GUI	31
5.7.2	Kitchen GUI	33
5.7.3	Management GUI	34
5.8	Flow Charts	37
5.9	Design	39
5.9.1	Design Techniques	39
5.9.1.1	Use Case Diagram	39
5.9.1.2	Class Diagram	40
5.9.1.3	Activity Diagram	41
5.9.1.4	Sequence Diagram	42
5.9.1.5	Sequence Diagram for Admin	42
5.9.1.6	Sequence Diagram for Customer	43
5.9.1.7	Sequence Diagram for RMS	44
5.10	Chapter Summary	44
6	Implementation	45
6.1	Chapter Overview	45
6.2	Implementing Extreme Programming(XP)	45
6.3	Data storage and Retrieval	45
6.4	Stock Control	46
6.5	Development tools and techniques	48
6.5.1	Tools	48
6.5.2	Technologies	48
6.6	Reusable Components	48
6.7	Network Architecture	49
6.8	Main User Interfaces	49
6.8.1	Admin Dashboard	49
6.8.2	Order Food Cashier	49
6.9	Summary	50
7	User Evaluation and Testing	51
7.1	Introduction	51
7.2	Testing Procedure	51
7.3	Test Plan and Test Cases	51
7.3.1	Master Data Modules	52
7.4	Acceptance Testing	53
7.5	Summary	53
8	Software Maintenance	54
8.1	Software Maintenance Process	54
8.2	Maintenance Activities	55

8.3	Difficulties of Maintenance	57
9	Conclusion	58
9.1	Introduction	58
9.2	Problem Encountered	58
9.3	Lessons Learnt	58
9.4	Limitations	59
9.5	Future enhancements	59
	User Manual	60
	References	81
	Dissemination of Work	83

List of Tables

Sr. No.	Table No.	Table Name	Page no.
1	1.1	Showing the proposed features of the system and the motivation behind the features	4
2	1.2	A table showing the commonly used words throughout this report	5
3	2.1	Comparison of Existing POS system solutions	7
4	4.1	A table showing the stakeholders of the project	16
5	4.2	A table showing the proposed system features and allocated priorities.	17
6	4.3	Kitchen application functional requirements.	19
7	4.4	Management application functional requirements.	20
8	4.5	Order application functional requirements	20
9	4.6	Kitchen application non-functional requirements	21
10	4.7	Management application non-functional requirements	21
11	4.8	Order application non-functional requirements	22
12	6.1	A Table showing the available ingredients stock status.	47
13	7.1	Add customer test case	53

List of Figures

Sr. No.	Fig. No.	Fig. Name	Page No.
1	2.1	UML class model	9
2	2.2	Boehm's cost model	9
3	4.1	Use case diagram showing some of the major features within the restaurant management system.	18
4	5.1	Component diagram showing the higher level architecture of the system.	24
5	5.2	Database comparison diagram	26
6	5.3	Entity Relationship diagram of a meal	29
7	5.4	Entity Relationship of the menu, order, offer and system settings	30
8	5.5	Design of the order GUI.	32
9	5.6	Design of the kitchen GUI	34
10	5.7	Design of Manage Group GUI	35
11	5.8	Design of Manage Table GUI	35
12	5.9	Design of Manage Category GUI	36
13	5.10	Design of Manage Product GUI	36
14	5.11	Design of Manage Store GUI	37
15	5.12	Flow chart to show the flow of events of an order.	38
16	5.13	Overall High-level Use case for the proposed system	40
17	5.14	Class diagram	41
18	5.15	Activity diagram- Customer order	42
19	5.16	Sequence Diagram for Admin	42
20	5.17	Sequence Diagram for Customer	43
21	5.18	Sequence Diagram of Online Restaurant Management System	44
22	6.1	Admin dashboard	50
23	6.2	Cashier Order Module	50
24	8.1	Maintenance process model	55
25	8.2	Maintenance Activities	57

1. Introduction

1.1 Chapter Overview

This chapter gives an introduction to the project by defining the problems encountered by restaurants, the main objectives that the system expects to achieve and a brief introduction to existing solutions.

1.2 The Problem

According to a research article written by Horizons [7], in 2006 within the UK there was just over 26,000 restaurants with 734 million meals served that year. As this restaurant sector was worth £7.61 billion, any restaurant generating a good business reputation could lead to the making of a very successful and profitable business. The problem for many businesses is to ensure that they not only attract new customers but to ensure they maintain their existing clientele. It has been argued many times that an existing customer is worth more to a business than a new customer as the cost to attract a new customer can be up to five times the cost to retain an old customer. An online article by Paul Lemberg [9], discusses the pros and cons of this argument.

Within the restaurant sector, a customer is likely to return to the restaurant in the future if they received an excellent customer service as well as appetising food. However, if they had to wait for an unreasonable amount of time or there was a mistake in the order, it's very unlikely the customer would return.

Therefore a solution to this problem would be to minimise mistakes within the order and bill, and help eradicate delays as well as encouraging team work and communication within the team.

The next section will go into the objectives of the proposed solution.

1.3 Project Objective

The objective of this project is to build an electronic restaurant management system using all of the skills and techniques from the field ensuring

that no common development mistakes are reproduced. Project management is critical to all software engineering projects and keeping to a project plan will be of similar importance.

One of the main objectives of any business is to maximize profit by increasing efficiency and decreasing overheads¹ without compromising customer satisfaction. Currently, many restaurants use a paper-based system to communicate between the restaurant and kitchen which can be shown to be one of the least efficient approaches. Even though this approach is implemented in successful profitable restaurants, there are several problems which could be seen as reducing the restaurant's efficiency:

- Miscommunication caused by handwriting.
- Unmanageable order logging.
- Inefficient restaurant-kitchen communication.
- Difficult order tracking and time management.
- Difficult stock management.
- Limited statistical output.

By introducing an electronic restaurant management system these problems can be avoided or improved leading to an increase in profits.

The initial project plan drafted at the beginning of the project can be found in Appendix A.

1.4 Existing Solution

There are many computerised restaurant management systems available but for each system there exist disadvantages or missing features. The most common type of restaurant management system contains a static order entry computer system usually in the shape of a desktop computer with a touch screen. Typically this common approach is adequate to the restaurants requirements but still requires handwritten orders to be relayed to the order entry computer system. A table comparing features of existing solutions will be presented in Section 2.3.

A slightly different approach was implemented in a restaurant in Nuremberg, Germany, nameds Baggers [16]. The restaurant utilises a roller coaster approach to serving the food and an orderentry system fully operated by the customer. As reviewed by the BBC [15], there is no need for any waiters as the customers use touch-screen monitors to browse the menu. This new invention can save on operating costs, but the initial injection of cash required is substantial as every table requires the necessary hardware.

The next section will introduce the project proposal listing the proposed features of the system.

1.5 Project Proposal

The aim of this project is to create a restaurant management system that can incorporate the benefitsof all the existing solutions but without any of the drawbacks as well as including many new features. A list of proposed features can be found in table 1.1.

Many of the existing solutions to POS (Point-of-Sale) systems are sold with the required expensive hardware so for any business looking to work to a budget, the more enriched software solutions are justout of their range.

Table showing the proposed features of the system and the motivation behind the features.

Features	Motivation
Automated stock control.	Real-time view of ingredient stock levels so only the meals with enough ingredient stockcan be sold.
Meal option and preference selection	Flexible meal options available for the customer.
Wireless order system	Waiters no longer required to walk to and from the central order computer system.
Advanced discount function.	Calculating the best price for the customer.
Order alerts.	Kitchen and bar staff in direct communication with waiters allowing the kitchen to notify the waiter that

	service is required
Flexible GUI design	Software capable of being used on any sized screen and so must have a flexible design.
Order logging.	All orders logged for future query generation
Large kitchen order display	Easy tracking and viewing of all active ² orders

Table 1.1: showing the proposed features of the system and the motivation behind the features.

1.6 Summary of chapter

The rest of this report consists of the following chapters:

- Background: Background investigation into the problem.
- Requirement Analysis: Requirements of the system including stakeholder identification, list of features and tabulated requirements.
- Design: Project design process using several diagrammatic techniques.
- Implementation: Discusses the implementation of the software with the help of diagrams and pseudocode.
- Results: Illustrates the system using screenshots.
- Testing: Documents how the system was tested.
- Conclusion: Project conclusion with future development ideas.

1.7 Commonly Used Words

Throughout this report, there are many commonly used words as detailed in table 1.2.

Word	Definitation
Ingredient	Ingredient of meal
Prepared ingredient	Collection of ingredients.

Item Drink	A meal or a drink
Menu section	A section of menu for example starters, meats, puddings etc.
Suborder	A collection of customer ordered items.
Order	A set of suborders

Table 1.2: A table showing the commonly used words throughout this report

1.8 Closing Remarks

This chapter has introduced the foundations of the project. The next chapter gives some background investigation into the problem.

2. Background

2.1 Chapter Overview

This chapter gives an insight to Point of Sale (POS) systems similar in nature to that of the one being developed in this project. It also gives a brief introduction to the importance of requirement gathering, a discussion on the development methodologies available as well as a justification on the platform and software used in this project.

2.2 Point-of-Sale (POS) System

According to A. Nutt [12], POS systems first dated back to the 1870s, when James Ritty became frustrated with dishonest employees stealing money from the customers in a saloon in Dayton, Ohio. With the inspiration from a counter on a ship's propeller that counted the number of revolutions, he and his brother in 1879 developed the first cash register called 'Ritty's Incorruptible Cashier'.

By the 1970s, the first computerised cash register was developed that was basically a mainframe but packaged as a store controller that had the ability to control the registers. Then in the 1980's, cash registers began to be PC operated which meant that many of the basic PC functions were now available.

Today, the POS systems are much faster, safer and reliable and with the introduction of credit cards and direct communication to the credit card company, transactions are almost instant.

2.3 Existing Point-of-Sale System

POS System	Table Service	Stock Control	General Reporting	Advanced Discounts	Kitchen Display	PDA Order
Point of Success [13] (Standard)	C	X	C	X	X	X

RPS (HOSPOS) [18]		X	C	C	X	C	X
Abacre [14]	POS	C	X	C	C	C	X
eZee Burrp [2]		C	C	C	X	C	C

Table 2.1: Comparison of existing POS system solutions.

The proposed features in table 1.1 were generated from research into numerous POS systems. Table2.1 shows the comparison between some of the proposed features and the features of other POS systems. All these POS systems were found on the first page of a Google search on the 14th October 2009.

2.4 Platform Choice

Choosing a suitable platform normally goes down to the programmers experience and the type of software to be developed. The restaurant management system could be developed as a web application or a standalone application but must also be widely supported and platform-independent. Therefore as the developer has minimal or no experience in web programming, the decision was taken to develop a standalone application.

The next decision was to decide on a programming language, with the developer having previous experience in C, C++ and Java. This decision was fairly easy and Java was the selected programming language as the developer has great knowledge in the Java Database Connectivity (JDBC) API¹ that allows database-independent connectivity between the Java programming language and numerous databases [8].

2.5 Software Choice

In software development, the use of integrated development environments (IDE) can increase the efficiency of a programmer. An IDE is a software application that consists of a source code editor, compiler and debugging tools

with its main aim to assist the programmer. Simple notepads are not strictly IDEs but can do the same job with the assistance of a compiler.

The two most popular IDEs available for Java programming are Netbeans and Eclipse as they are free, support multiple platforms and offer many features including integrated version control and debugging tools. The two main problems with IDE's are that due to the wealth of features available and plug-in support, there is an associated cost, as their performance is poor and in particular they require more memory and processing power than a standard text editor.

For this project Netbeans was the chosen IDE, as the developer has more experience and knowledge of the Netbeans graphical user interface.

2.6 UML

Diagrammatic techniques are used to visualise, construct and document software systems under development. The most general modelling language to describe both the structure and behaviour of a software system is Unified Modelling language (UML) created by the Object management group. The diagrams one can create using UML can be shown by a class diagram (Figure 2.1). Throughout this report, numerous models defined within the UML class diagram (Figure 2.1) will be used to graphically represent the system.

2.7 Requirement Gathering

Requirement gathering is a very important step in software engineering. According to an article written by Craig Murphy [11], Boehm's cost model discusses how discovering errors at an early stage of software development can reduce overall costs. Figure 2.2 is an example of Boehm's cost model and uses the waterfall technique to give a visual insight into how important gathering accurate requirements can be when the project has a strict deadline to work to. Roughly a £1 cost of change in the requirement gathering can cost up to £100 in the testing stage and £1000 in the deployment stage to fix.

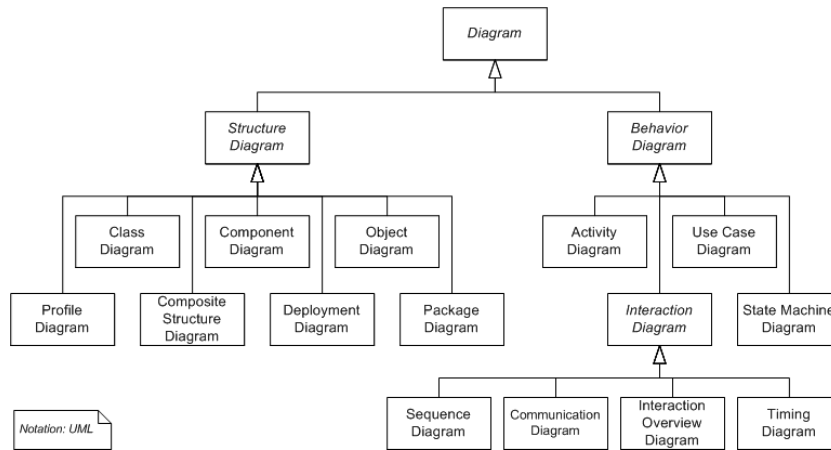


Fig. 2.1: UML class model.

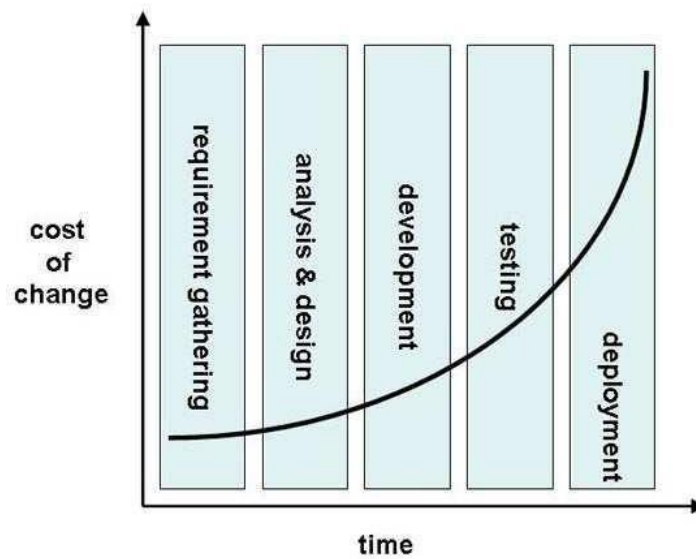


Fig. 2.2: Boehm's cost model [11].

2.8 Development Methodology

A software development methodology is a framework used to plan the design of a software system controlling the process of development. According to Geoffrey Elliott [5], software development methodologies first emerged in the 1960's with systems development life cycles (SDLC) being considered the first formalised methodology. Since then there have been numerous popular software

development approaches including the waterfall model, prototyping, incremental, spiral and agile.

The agile methodology refers to a collection of different agile methods. This project will be based on Extreme programming (XP) which is one of these agile methodologies using an iterative based framework. Each iteration has a development cycle very similar to the waterfall model consisting of planning, requirements analysis, design, development and testing. At each iteration, the business representative sometimes referred to as the 'customer' is given a demonstration to promote useful feedback. This type of methodology reduces risk and lets the project adapt to requested changes quickly with minimum cost.

According to the Agile Manifesto [10], some of the main principles behind the agile methodology are:

1. Customer satisfaction by rapid, continuous delivery of useful software.
2. Working software that is delivered frequently.
3. A welcomed late change in requirements.
4. Simplicity.
5. Regular adaptation to changing circumstances.

Given these principles and the nature of the project, extreme programming was the best choice methodology to use as we expected frequent customer communication and constant requirement changes post customer feedback. Therefore for this project, the supervisor acted as the customer regularly reviewing the work. Any errors or unthoughtful designs were then picked up and fixed relatively early in the process helping to minimise costly errors as shown in Boehm's cost model (Figure 2.2).

2.9 Chapter Summary

This chapter has given examples of POS systems that are directly related to this project as well as general information about requirement gathering and development methodologies. The next chapter starts the process of the development methodology by generating the requirements of the system.

3. LITERATURE REVIEW

3.1 Introduction

This chapter will contain or provide general understanding about the related review of literature of the beach house restaurant management system and any other articles that explains the meaning of it in detail; also this system will discuss the conceptual and theoretical development of mailing system which means the previous works and comparing between the existing systems. In the long run, this chapter will conclude with summarizing the chapter with few lines.

3.2 Therotical and Conceptual Development

AN 18th-century Parisian chef, assumed by generations of historians to be the world's first restaurateur, is a figment of the French imagination, according to an American academic who spent four years trying to track him down.

In a newly published book which has provoked horror in the rarefied world of Parisian gastronomes, Rebecca Spang, who teaches at University College, London, has challenged French historians to produce evidence to back up the claim that a Frenchman named Boulanger was responsible for inventing the country's best loved social institution.

For more than 200 years Mr Boulanger has been credited with opening the world's first Paris restaurant in 1765. But Miss Spang has been unable to find any evidence that he ever existed. "This man named Boulanger simply never appears in any of the sources I have examined," she told The Telegraph.

Her investigations are something of an embarrassment for some of France's most eminent food experts. The commercial activities of the elusive Mr Boulanger have been respectfully chronicled in countless histories of French cuisine, all of which may now need substantial revision.

According to the Guide Gourmand de la France, a standard reference work for French restaurants, Mr Boulanger set up his establishment in what is now the Rue du Louvre, in Paris's first arrondissement.

Above the front door he is reported to have placed a sign stating: "Boulangier débite des restaurants divins" ("Boulangier provides divine sustenance"), so becoming the first businessman to use the word "restaurant" (albeit in its original meaning) to describe a place where food can be had as well as the first to offer a choice of dishes to customers.

Below the sign he was said to have added the Latin invitation: "Venite ad me omnes qui stomacho laboratis et ego vos restauro," ("Come to me, those who are famished, and I will give you sustenance").

The guide goes on to note that Mr Boulangier's most innovative dish was "sheep's feet in a white sauce" and attributed some of the success of the new restaurant to his beautiful wife, who allegedly attracted the admiring attentions of the writer Diderot. Other sources are no less detailed. The Academic Dictionary of Gastronomes devotes a page to describing a legal battle between Mr Boulangier and the Parisian food guilds which attempted to outlaw the restaurant, on the grounds that it represented unfair competition.

3.3 Inter Organization Case Study

3.3.1 Peace House Restaurant

Peace House Restaurant which was established in 2005 by independent scholars is a private business which is categorically non-political, non-partisan and non-sectarian.

Peace House Restaurant was established at a time when it's hard to establish a business and could hardly get an opportunity to proceed their business after wars due to scarcity of affordable and creditable tertiary business in Mogadishu. The business who stood for the establishment of peace house at the hardest time in the modern history of Somalia contributed a lot to the resilience of their country. The enthralling vision and mission peace house restaurant founders chose for this business will continue to inspire and motivate all the community to remain attached to it in the delivery of tertiary business that meets skill gaps among the rising generation in Somalia. Currently, More than 700 employees are pursuing various levels programs in the Peace house restaurant.

Peace house restaurant aims at providing food ordering and table reservation that are highly relevant to the market and social demands in its quest for better standards of living, justice, equity and good governance among Somali people. To achieve that superior objective together with the vision and the mission of Peace House restaurant, restaurants always seeks highly educated, committed, trustworthy, and disciplined personnel for all its administrative and business programs.

Peace House Restaurant profile

Peace House Restaurant is Somali entirely glass hilltop fine dining restaurant. It's the ideal place for that very special night out and the best spot to watch the sunset - right from your dining table! Dine in the ambient moonlight and enjoy panoramic views of the city lights whilst enjoying the live music from some of the best jazz entertainers the city has to offer.

Just 35 minutes from the centre of zoobe, yet in a world of its own, a night at Peace House will have you basking in the peaceful and relaxing atmosphere of our hilltop setting. But it's not all about location - here at Peace House we have forged a reputation as one of top restaurants. We believe in using only the freshest ingredients, the most talented chefs and waiting staff of the most professional standard. Be prepared to be properly taken care of by our outstanding staff whilst you enjoy our delicious cuisine.

3.4 Compare Between Existing System

Before you can design a new system you must have a good understanding of the information flow of the existing system let it be manual or computer-based.

3.4.1 Current System

The current system of the restaurant uses computerized such as desktop system for the daily operation, although, the Desktop system is not up to-date enough to cover the existing need. The restaurant uses software programs such as Desktop system Or MS-Excel for recording the employee data, that is; when new employee is hired, his/her

has to fill in the computerized work contract form that contains personal details of employee including his/her name, age, sex, date, address, profession, permanency/part-time, location, salary and so on.

Then this information is kept in a computerized that is present in the hired office and the necessary details are recorded in the other computers later on. However, this is not frequently used in the digitalized world.

3.4.2 New System

The new system makes it easier to manage the employees effectively. However, The restaurant management system that is computer based application will provide a working environment that will be flexible, efficient and user-friendly by affording easy of work with significant reduction of time. In an computer platform that many remote clients can access; the system will need more reliable security to uniform application in secure manner to the visitors of the site.

The administrator of the system will be capable to give different privileges and permissions to the users and visitors of the system, and all permissions will be monitored in restricted way. Some systems available now was developed with old versions of Software that causes the system's speed to be slow or totally unsuitable. They are static and unchangeable sites. However we are trying to develop a new system with the latest versions of Software to be fast and reliable.

3.5 Chapter Summary

This chapter we briefly discussed many sections starting with the definition of the restaurant and its history, also it has explained and mentioned Introduction of restaurant management, the conceptual and the theoretical development of Restaurant management System.

This chapter has focused on Organizational case study by talking about the restaurant management as the case study of this project and the user requirements about the system. This chapter also included the comparison between current systems of restaurant for the peace house with the new system which this project is going to solve. The other main areas that finally the chapter has concluded with mentioning the features and advantages of the new system over the current system.

4. Requirement Analysis

4.1 Chapter Overview

This chapter will look at the stakeholders of the system and then discuss the required features using a use case diagram to illustrate.

4.2 Stakeholder Identification

As defined in the Business Dictionary [4], a stakeholder could be a person, group, organisation that has direct or indirect stake in an organization because it can affect or be affected by the organization's actions, objectives and policies.

Hence, stakeholders can be split into two groups; internal and external with each stakeholder contributing directly or indirectly towards the business activities.

As an example, any system where there exists customer communication, that customer will be a non-financial beneficiary stakeholder. According to an article written by Ian Alexander [1], the person, group, organisation, or system is a stakeholder if they can be defined by any one of the following four questions:

Who needs to be consulted on the scope of this project? Who has an input to the budget of this project?

Who can support or harm this project politically?

Who can provide guidance on the usability, functionality, and required qualities (reliability, safety, lifetime, maintainability, . . .) of the system under development?

Therefore using these 4 questions as a guide, we can generate a list (Table 4.1) of the stakeholders in this project.

4.3 Use Cases

A use case diagram that is part of the Unified Modelling Language (UML) which was introduced in Section 2.6 gives a graphical overview of the

functionality of a system. A use case diagram consists of actors that are normally the stakeholders of the system and their use cases commonly defined as goals.

Stakeholder	Role
Management	User of the management application
Waiters	User of the management application
Kitchen Staff	User of the kitchen application
Bar Staff	User of the restaurant application
Customer	Indirect user of the system
Cashier	External stakeholder accepts payment

Table 4.1: A table showing the stakeholders of the project

Shows several use case scenarios of the system that convey how the stakeholders interact with the features to achieve a business requirement. The use case scenarios from figure 3.1 can be found in Appendix B.

The use case diagram is designed to be sequential so by following the use cases down the spine, one can follow the major steps of an order and several post features to query the data.

4.4 Features

An important part of requirements gathering is the production of a list of system features that categorises on priority.

Feature	Priority
Communication of data between each application.	1
Minimum click touch screen GUI design for efficient ordering.	1
Meal ingredient and cooking preference options.	1
Interface to view active orders in the kitchen.	1
Ability to add flexible discounts; calculating best price for the customer.	1
Interface to maintain and manage the menus and associated meals.	1
Stock control for all ingredients; reducing/increasing stock	1

automatically.	
Ability to define groups of ingredients that may be used in numerous meals.	2
Flexible meal grid design to fit any screen size.	2
Real time waiter status alerts.	2
User login functionality.	3
Interface for table management and selection.	3
Figure generation; management can view statistics in numerous forms.	3
Automatic daily stock level alerts.	3
Ability to define meals by images as well as text.	3

Table 4.2: A table showing the proposed system features and allocated priorities.

4.5 Measurable Goals and Requirements

The measurable goals and requirements of the system are a list of manageable¹ requirements and goals for each application that can be prioritised and ‘ticked off’. The software requirements specification (SRS) is a complete list of requirements to be designed and developed and can take the form of functional or non-functional requirements.

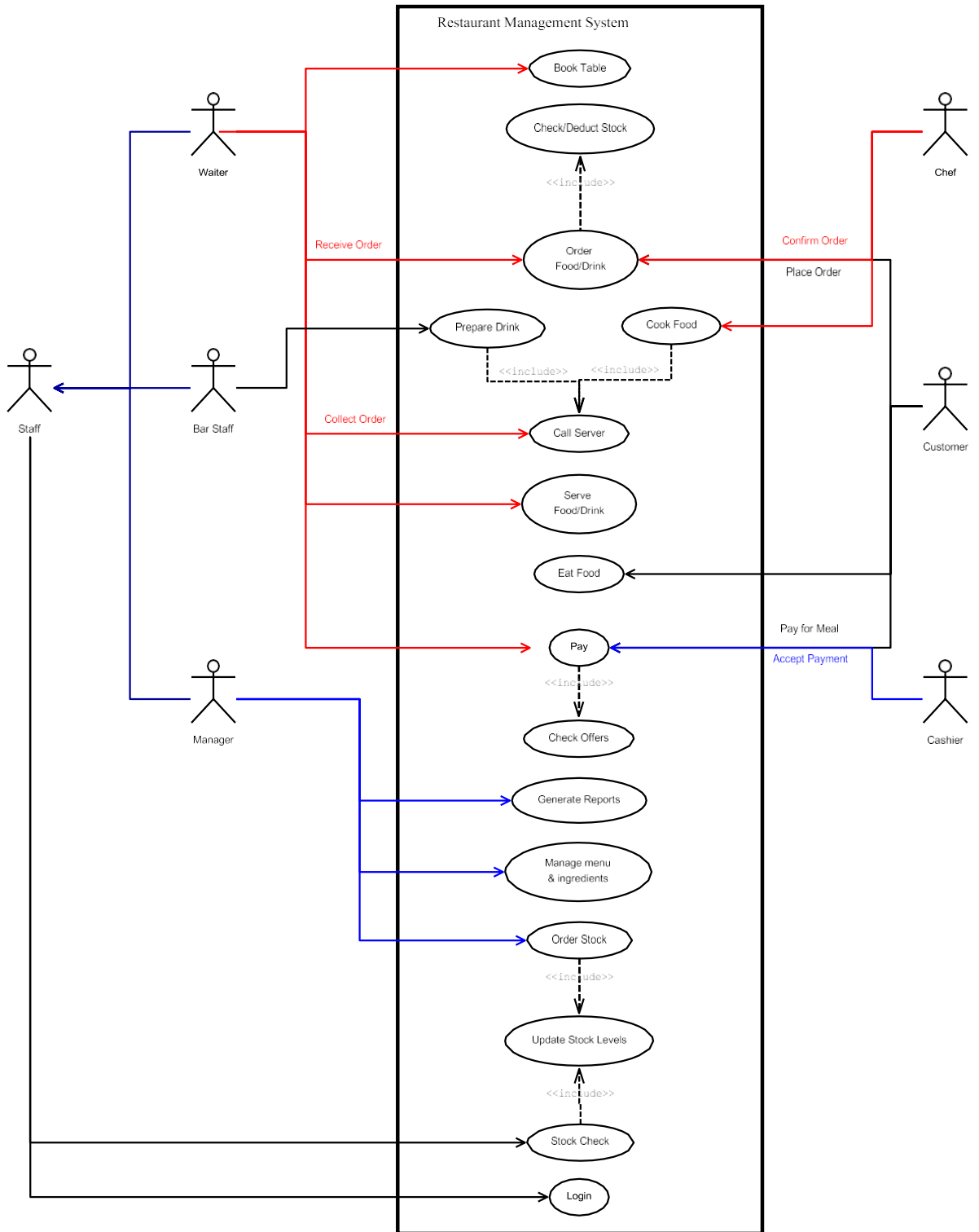


Fig. 4.1: Use case diagram showing some of the major features within the restaurant management system.

4.5.1 Functional Requirements

A functional requirement is a requirement that, when satisfied, will allow the user to perform some kind of function. The functional requirements of the kitchen, management and order application can be found in tables 4.3, 4.4 and 4.5 respectively. The priority value is in the range from 1 to 3 where 1 is high priority and 3 is low priority.

Requirements	Priority
Organised display of active orders.	1
View preferences and optional choices of every meal.	1
Inform waiter; update order to 'in progress'.	2
Inform waiter; update order to 'ready'.	2
Display elapsed time and progress ² of each order.	3

Table 4.3: Kitchen application functional requirements.

Requirements	Priority
Ability to display the real time stock levels.	1
Ability to add ingredients, meals and menus.	1
Ability to edit ingredients, meals and menus.	1
Ability to remove ingredients, meals and menus.	1
Ability to display statistics of complete orders.	3
Ability to add new waiters to the system.	3
Ability to add, edit and remove offers.	1
Ability to add time intervals ³ to menus and offers.	1
Ability to alter the grid size of the meals, menu sections and menus.	1
Ability to alter the seating layout.	3
Ability to alter the price of ingredients.	1
(Note: Ingredient might still be in stock at a different	

price)	
Ability to change transparency of an image that represents a meal.	3
Means of stock checking.	1
Means of updating the minimum stock level.	1

Table 4.4: Management application functional requirements.

Requirements	Priority
Ability to view the active suborder details.	2
Ability to add a new order to a table.	1
Ability to add a new order without defining a table.	1
Ability to add a suborder to an existing order.	1
Ability to delete a suborder that has not yet been confirmed within the order system.	1
Ability to view optional ingredients in a meal.	1
Ability to view cooking preferences of ingredients that can be cooked different ways.	1
Ability to only view the active ⁴ menus.	1
Ability to view the status of all active suborders.	2
Ability to print customer receipts on order completion.	3
Ability to view transaction list of current order.	3
Ability to alert the waiter when the drinks are complete.	3
Ability to delete an item or clear the transaction list.	2

Table 4.5: Order application functional requirements

4.5.2 Non-Functional Requirements

Non-functional requirements are usually some form of constraint or restriction that must be considered when designing the solution. The non-functional requirements of the kitchen, management and order application can be found in tables 4.6, 4.7 and 4.8 respectively. The priority value is in the range from 1 to 3 where 1 is high priority and 3 is low priority.

Requirements	Priority
Ability to interact with the MySQL database.	1
Means of refreshing the orders within the ordered list.	1
Means of refreshing the status of orders.	3
(How many items in the suborder have been cooked)	
Means of refreshing the status of orders.	3
(How many items in the suborder have not been cooked)	
Means of refreshing the menu section list.	2
(How many items in the menu section have been cooked)	
Means of refreshing the menu section list.	2
(How many items in the menu section have not been cooked)	
Means of refreshing the meal colours within the ordered list:	3
==> White: not started	
==> Yellow: in progress	
==> Green: complete	
Means of displaying only the food items so drink items are omitted.	1
Means of clearing the suborder from the kitchen display.	1

Table 4.6: Kitchen application non-functional requirements

Requirements	Priority
Ability to interact with the MySQL database.	1
Means of refreshing menus, menu sections and meals live.	3
Only accessible by management staff.	3
Means of displaying the links therefore being able to view the ingredients in a meal etc.	1
Means of searching the database using wild cards for easier data input.	2

Table 4.7: Management application non-functional requirements

Requirements	Priority
Ability to interact with the MySQL database.	1
Ability to update the suborder status every 5 seconds.	1
Ability to run on a PDA by defining smaller grid and font sizes.	2
Ability to use the application using a touch screen	1
without the need of a keyboard or mouse.	
Colour co-ordinated meal buttons to visually warn the user of low stock.	2
Ability to disable meals that have run out of compulsory stock items.	1
Ability to only show an ingredient as an optional ingredient choice	1
if there exists enough ingredient stock.	
Ability to calculate stock ingredient price, order price and discounted price.	1
Live refreshing of the menus depending on the starting time of the order.	1
Ability to apply offers depending on the starting time of order	1
even if the offer is inactive on completion of the order.	
Minimum clicks from the beginning to the end of an order.	1

Table 4.8: Order application non-functional requirements

4.6 Chapter Summary

This chapter has discussed the requirements of the system and the development methodology that will be used throughout this project. The design of the system is documented in the next chapter.

5. Design

5.1 Chapter Overview

This chapter will focus on the design of the system using diagrams to illustrate graphically certain sections of the software system.

5.2 Introduction

This project has been designed using numerous diagrammatic techniques. Recall from Section 2.6, that the most general modelling language to describe both the structure and behaviour of a software system is Unified Modelling language (UML).

Use case diagrams have already been used in the requirements analysis as a way to graphically overview the order process within the system. Other diagrams from the UML family are used in the design stage to show the structure and behaviour of numerous sophisticated design features.

5.3 Component Diagram

A component diagram is part of UML and its main purpose is to show the structural relationship between components in the system. A component diagram is useful for this system as it shows the higher architecture, as shown in figure 5.1.

5.4 Data Storage

The restaurant management system will be built around the data storage technique therefore choosing the most appropriate persistent¹ data storage is critical to a successful project and we can assume a flat file storage approach is inadequate. The two most popular types of persistent data storage available are relational database management system (RDBMS) and extensible markup language (XML).

5.5 Relational Data Base Management System (RDBMS)

A relational database management system (RDBMS) is a database managed system based around a relational model and are the corner stone's to many software systems including web based systems.

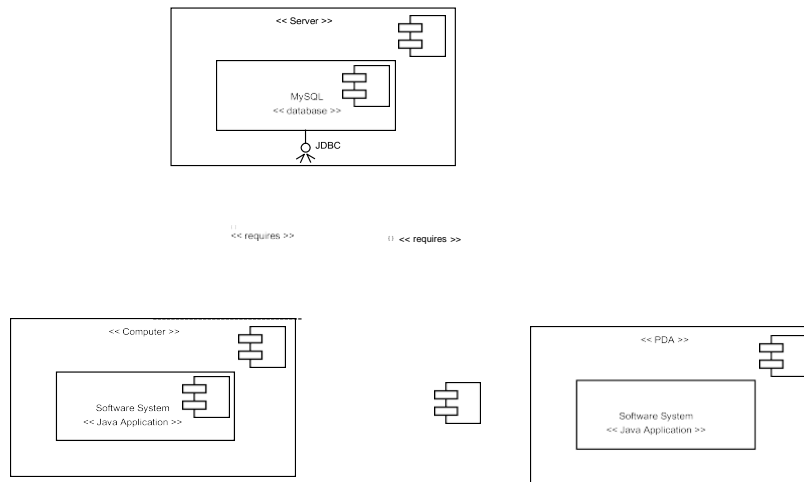


Fig. 5.1: Component diagram showing the higher level architecture of the system.

RDBMS are one of the most popular data storage methods out in the market and offer many advantages including:

- Fast data extraction using structured query language (SQL).
- Good management of data and security through the management system.
- Good level of data consistency.
- Advanced features including functions and triggers.
- Requirement of a data model to be developed; leading to long term cost effectiveness.

In industry, there are numerous expensive highly functional RDBMSs including Oracle and SQL Server that are very popular and offer technical support. However, there are also numerous open-source solutions with many adjudged to be as good or better and are becoming even more popular with small scale software systems.

5.6 Extensible Markup Language (XML)

XML is a markup language that was designed to transport and store data and is another example of a persistent data storage technique. However, it is not a predefined language thus all tags must be defined and due to its hierarchically data structure all elements must be promoted or demoted.

XML could be used in two different ways in data storage; storing the XML documents within a database or having the XML documents as the fundamental unit of storage. In both cases the XML can be queried using either XPATH or XQUERY which are query languages for extracting data from XML documents.

5.6.1 Storage Method Chosen

The main difference between XML and RDBMS is that XML is hierarchical and RDBMS is relational. As restaurant data can be best represented in a hierarchical way one would believe that XML would be the best approach but it's not always that straight forward. SQL is an extremely flexible and robust querying language and for the queries required and the type of software system being designed, it was concluded that RDBMS would be the best storage method.

The next choice was to decide on the type of RDBMS to use. As discussed there are many open source RDBMSs available for us to choose from and for the main reason of experience, MySQL was the preferred option. Figure 4.2 shows just how competitive the performances of different RDBMSs are.

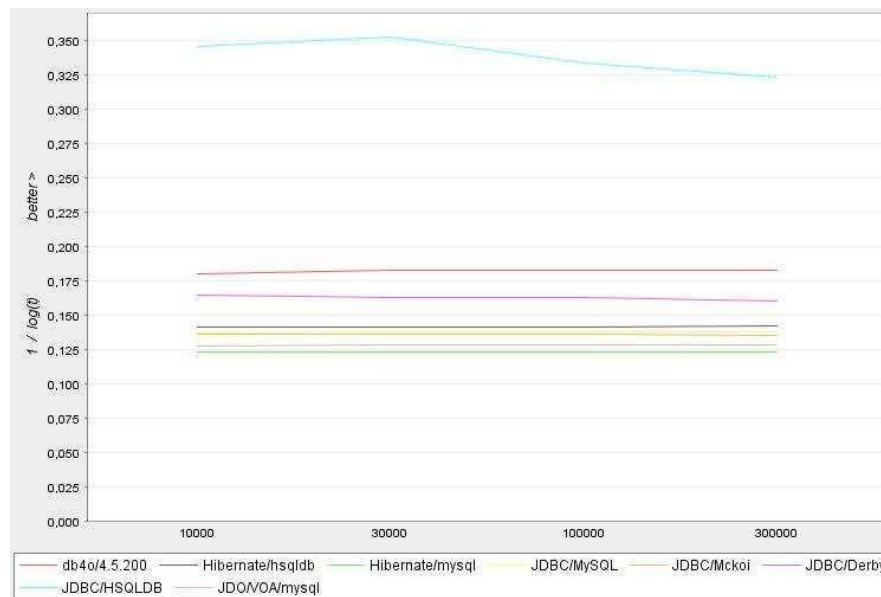


Fig. 5.2: Database comparison diagram [3].

5.6.2 Normalisation

Normalisation comes in many forms ranging from first normal form to sixth normal form. The normalisation of a database is a systematic way to free the database of undesirable characteristics where inserts, updates and deletions of data could lead to the loss of data integrity. The greater the normal form, the greater the data integrity of the database.

The database in this system was designed to be in Boyce-Codd normal form which is a slightly stronger version of the third normal form. For the database to be in Boyce-Codd normal form, it had to pass for all previous normal forms as well as Boyce-Codd normal form.

A well designed database will normally abide by the first, second and third normal forms as they are the basics to a well structured relational database. According to Horsforth School [17], the first three normal forms can be defined as:

First norm: Every attribute is atomic or single valued therefore there are no repeating fields.

Second norm: All attributes not part of the primary key must be dependent on the

full key.

Third norm: There must be no transitive determinants, or each attribute that is not part of the key must be determined only by the key.

Finally for the database to be in the desired Boyce-Codd normal form, all tables must abide by the first, second and third normal forms and must not have any determinants that are not candidate keys for the table.

5.6.3 Entity Relationship Diagram

An entity relationship (ER) diagram is a modelling language used to represent a type of semantic data model of a system. The ER diagrams are often used to represent a relational database and its requirements in a top-down fashion usually defined as the database schema. The database schema for this database has been split into two ER diagrams (Figures 4.3 and 4.4).

Figure 5.3 graphically shows the objects and their relationships that are contained within a meal. The meal object will be made of at least two ingredients that can be either a normal ingredient or a prepared ingredient. Note, a prepared ingredient is a collection of ingredients used to either group commonly used ingredients or to group optional ingredients. Each ingredient will have a default and manual measurement with the default measurement entered on input of the ingredient and the manual measurement entered if the meal ingredient link requires a different amount.

Also, each ingredient will be part of a generic ingredient object as there are many ingredients that are the same item but packaged in a different way at a different price. This allows the database to be in Boyce-Codd normalised state. An example of this would be the drink Coca-Cola which can be bought by bottle, can or draught, thus are the same item but packaged differently at a different price and amount. Finally each ingredient and prepared ingredient can be part of a category allowing optional ingredients to be interchanged with other ingredients in the same category.

Figure 5.4 graphically shows the relationships for the menu, order and offer objects. The menu consists of a date time relationship that provides the intervals to when the menu is active and a menu section relationship that contains the colour

variables and items under that particular menu section.

The order consists of one to many suborders with the suborder consisting of one to many items. The order stores all the ingredients within each item and also the replaced ingredient if that optional ingredient was replaced.

The offer consists of a date time relationship that provides the intervals to when the offer is active and a offer section relationship that contains the sets required by the offer.

5.6.4 Database Relationship Diagram

The database was the backbone to some highly important functional requirements, therefore the schema (Figures 4.3 and 4.4) needed to have the structure to deal with them.

Some important design concepts of the database are:

Ability to store prepared ingredients to reduce the size of the meal ingredient list. Recall, that a prepared ingredient is a collection of ingredients.

Ability to allow numerous options for optional ingredients within meals so that every ingredient (or prepared ingredient) is part of a category. If an ingredient is optional, then that ingredient should be able to be removed or swapped with any other ingredient in the same category.

Ability to control the stock levels by allocating a variable to all prepared ingredients and meals with the variable reacting in real time to the status of the items ingredient stock level.

Ability to cope with new supplies where the price differs to the current price within the database.

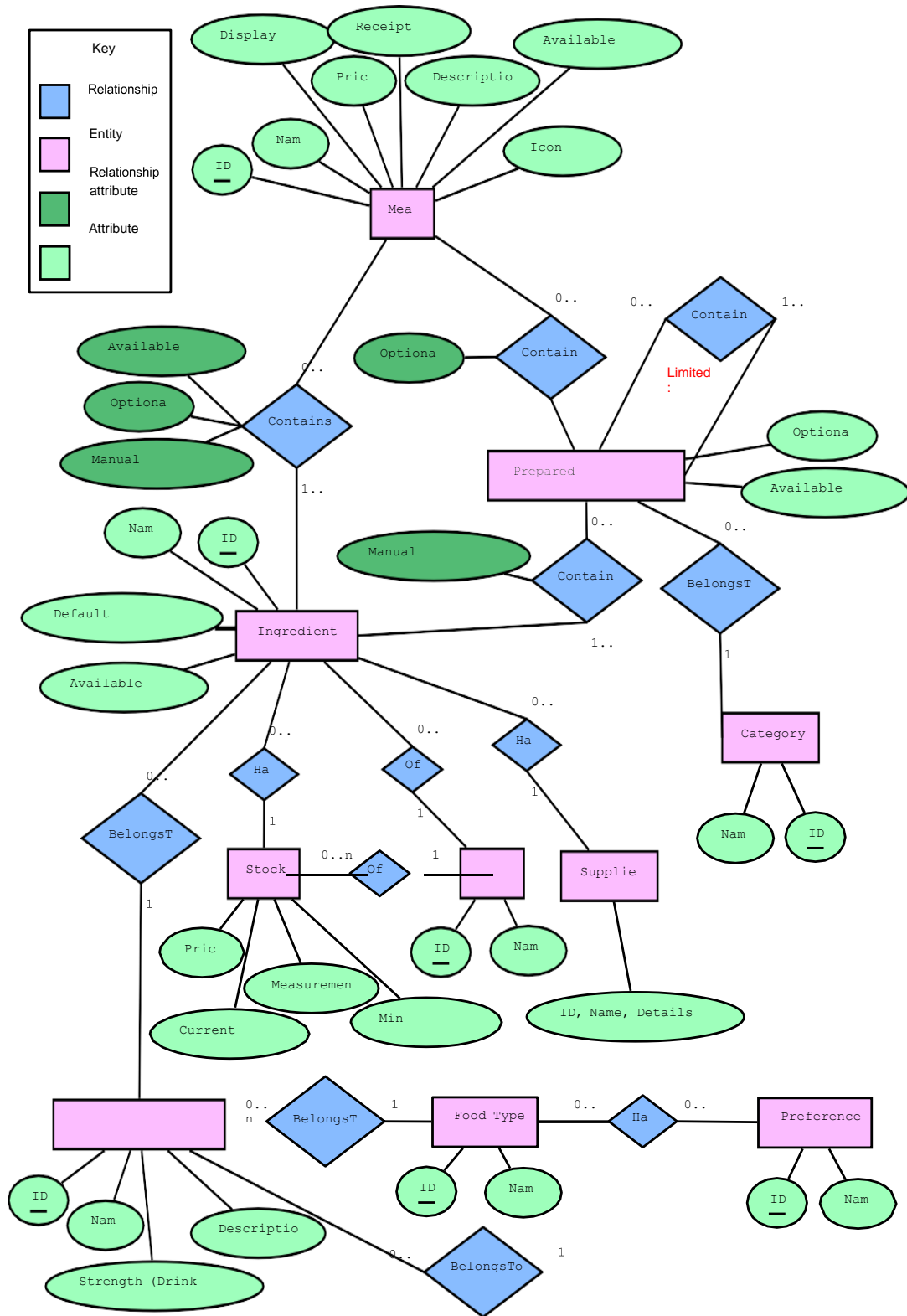


Fig. 5.3: Entity Relationship diagram of a meal

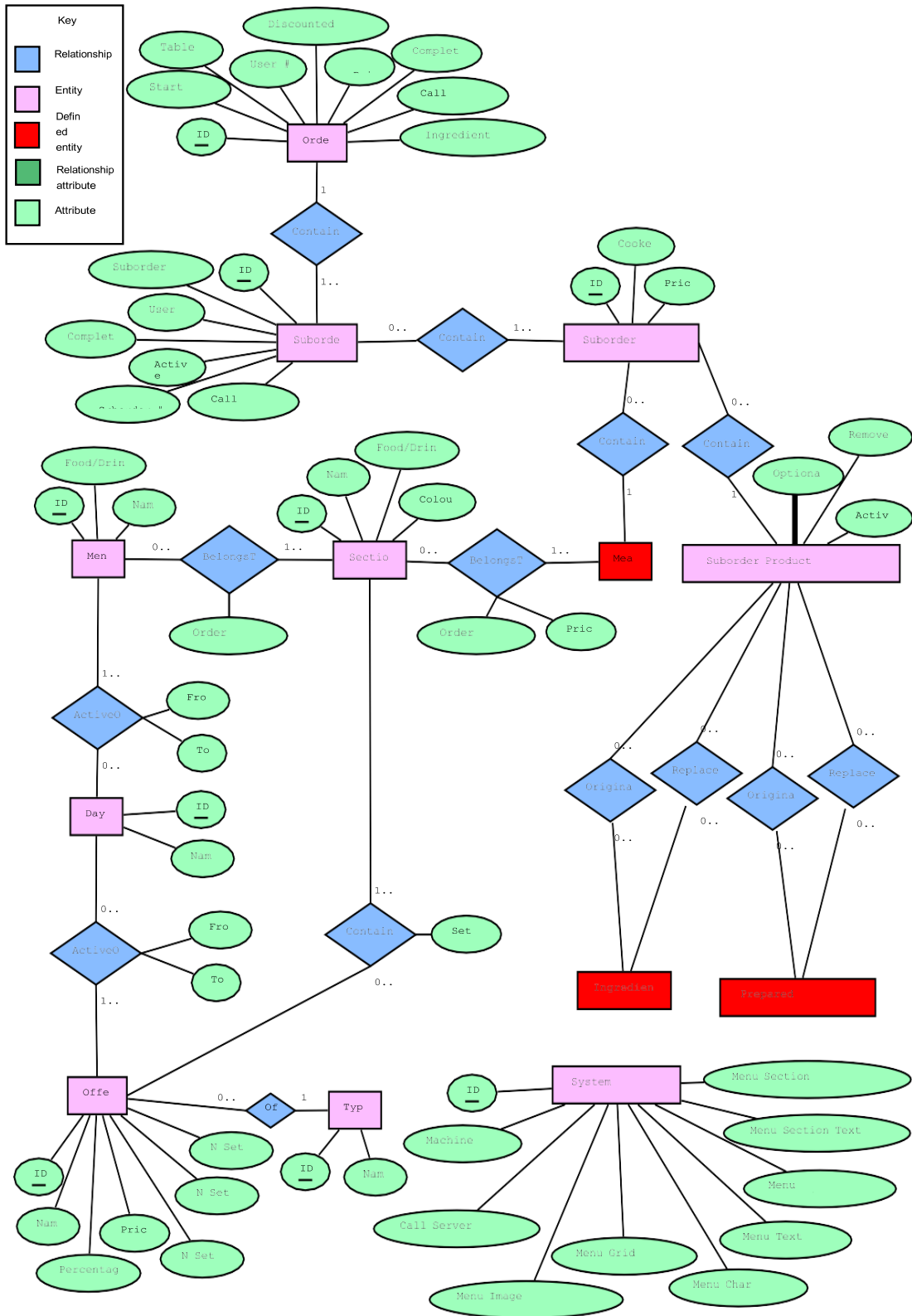


Fig. 5.4: Entity Relationship of the menu, order, offer and system settings.

5.7 Graphical User Interface

The graphical user interface (GUI) is the only component of the system that the user interacts with therefore is of great importance. The design had to be simple, clear and concise but whilst also showing all of the required features. The main objective was to create a GUI that allowed the user to get the order completed in the minimum time possible. This was judged by the time taken to complete the order as well as the amount of clicks required to get from the start to the finish. In total, there are 3 different GUIs within the restaurant management system with each GUI requiring a different design specification.

5.7.1 Order GUI

The order system GUI has the most user interaction through the means of a touch screen. Hence usability and user-friendliness of the GUI was of the highest priority. Therefore the specification was as follows:

- Chronological order of steps; either from left to right, top to bottom or a mixture of both.
- Minimum clicks from start to finish.
- Meal items with optional images.
- Ability to fit on any monitor size.
- Maximum space usage with readable font.
- Colour coordinated meal buttons to display the stock status.

Figure 5.5 shows a design of the order GUI that abides by the specification. The design keeps the sequence of steps in a chronological order to minimise hand movement from step to step. By following the design top to bottom and left to right, the order starts from the top left corner and eventually finishes at the bottom right corner. This not only helps keep the GUI structured, but allows the user to use their initiative if unfamiliar hence decreasing the learning curve.

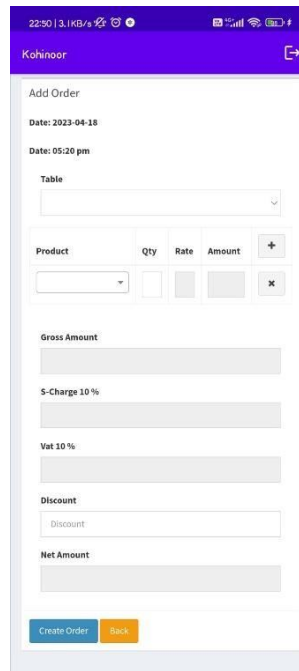


Fig. 5.5: Design of the order GUI.

The order process is controlled by the command buttons at the bottom. These allow the user to create an order, add a suborder to an existing order and complete an order. The three purple grids down the left hand side contains the different menus and current active suborders. The blue grid in the centre contains the different sections of the selected menu and the associated meals and drinks within the sections. The ‘Transaction List’ and ‘Price’ boxes show the current items within the suborder and the current price respectively.

The shape of the design in Figure 5.5 accommodates all normal size monitors with the grid size for the food/drink, menus, tables and items all customisable. Hence with little or no change the basic components in the order GUI can be designed to fit a PDA screen with a resolution of around 480 by 320 as shown in Figure 4.6. In the designs, Figure 5.5 would use, as default, the grid sizes 8x8 (items), 2x1 (food/drink), 2x3 (menus) and 3x8 (tables) where as the PDA GUI Figure 5.5 would use, as default, the grid sizes 4x4 (items), 1x2 (food/drink), 1x3 (menus) and 1x8 (tables). Note, these dimension would be the preset default values and would be changeable in the system settings.

As we can never assume the number of items, menus and tables active at any one point, all the grids should be able to automatically resize when the grid becomes full adding an extra row to the bottom. Figure visually shows how a full grid resizes when an additional item is added.

As well as the general layout, individual components required specific design details incorporated. As new meals and drinks could be added to the menu at any point, then if a normal grid was used, the addition of a new item that wasn't at the end of the grid would shift all items after the newly inserted item by one. At sight this is a problem, as with frequent use of the system, the user becomes accustomed to the exact position of certain functions. Hence a change to the layout could lead to an increase in human errors and a decrease in the user transaction speed.

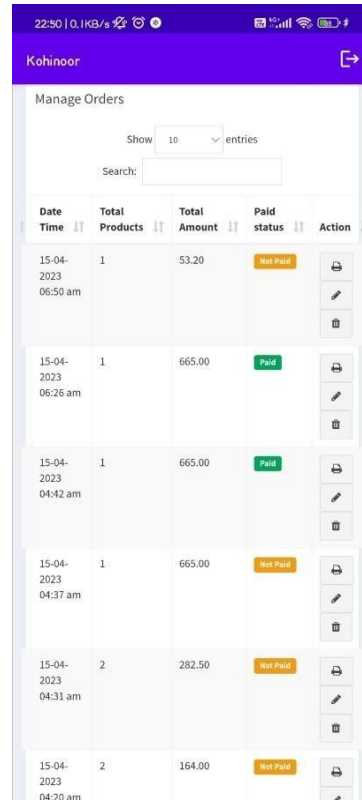
Figure shows a solution that assigns the starting item in every menu section a new line therefore the shifting of items would only occur when a section requires an extra line.

Finally, a well known saying is that a 'picture tells a thousand words', therefore by allowing the system to use transparent images as well as the text, the end user should locate the particular item faster.

5.7.2 Kitchan GUI

The kitchen system requires a GUI to display all active orders. This GUI requires little user input, therefore the design of the GUI and the way the data was displayed was the most important part. The specification of this GUI was as follows:

- View of all sub orders with elapsed time and status.
- Separation of ordered items into their specific menu section.
- Colour coordinated item list.
- Ability to view on numerous sized monitors



Date Time	Total Products	Total Amount	Paid status	Action
15-04-2023 06:50 am	1	53.20	Not Paid	[Print] [Edit] [Delete]
15-04-2023 06:26 am	1	665.00	Paid	[Print] [Edit] [Delete]
15-04-2023 04:42 am	1	665.00	Paid	[Print] [Edit] [Delete]
15-04-2023 04:37 am	1	665.00	Not Paid	[Print] [Edit] [Delete]
15-04-2023 04:31 am	2	282.50	Not Paid	[Print] [Edit] [Delete]
15-04-2023 04:20 am	2	164.00	Not Paid	[Print] [Edit] [Delete]

Fig. 5.6: Design of the kitchen GUI

Figure 5.6 shows a design of the kitchen GUI that abides by the specification. The design utilises as much space as possible, providing up to four sections. These sections give the kitchen staff the option to separate the orders down to any menu section, for example deserts, starters, meats etc. The suborder list on the right hand side of the design gives details about the status of active suborders including elapsed time and a count of how many items are ready

5.7.3 Management GUI

The management system requires a GUI to input data therefore requires an appropriate input design. Figure 4.10 shows the design of the management application with the different functions appearing down the left hand side as a tab. Each tab contains a subform with Figure 4.9 containing the structure of the generic data input form.

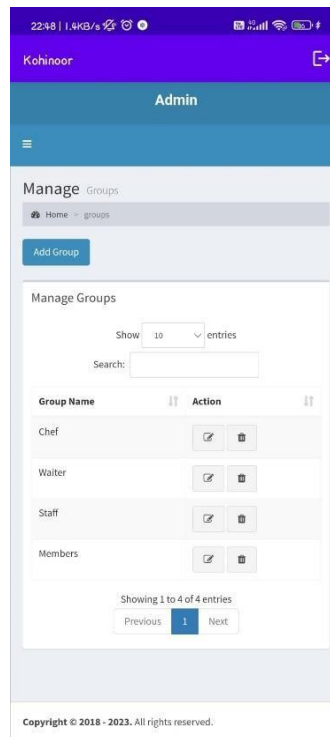


Fig. 5.7: Design of Manage Group GUI

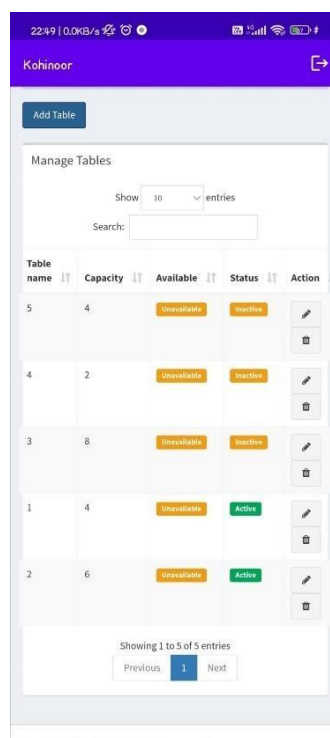


Fig. 5.8: Design of Manage Table GUI

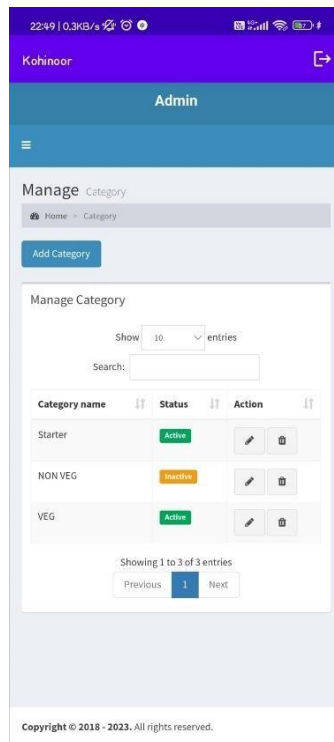


Fig. 5.9: Design of Manage Category GUI

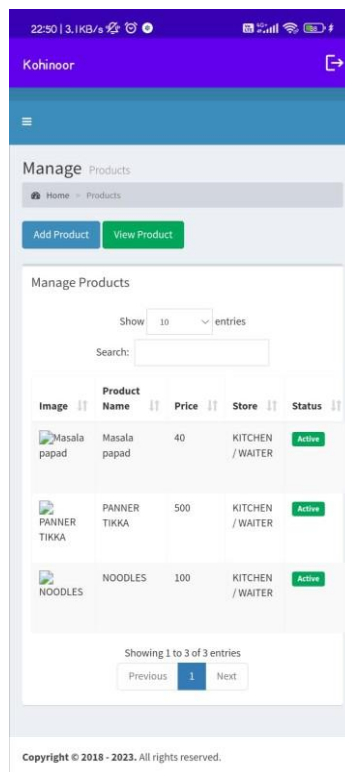


Fig. 5.10: Design of Manage Product GUI

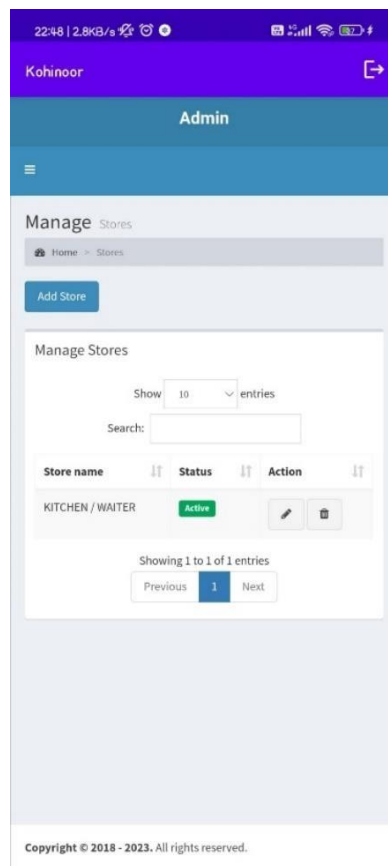


Fig. 5.11: Design of Manage Store GUI

5.8 Flow Charts

A flow chart is a diagram used to represent the process flow of an algorithm, problem or some transaction within a business. Therefore a flow chart (Figure 5.12) was used to graphically represent the process flow of an order.

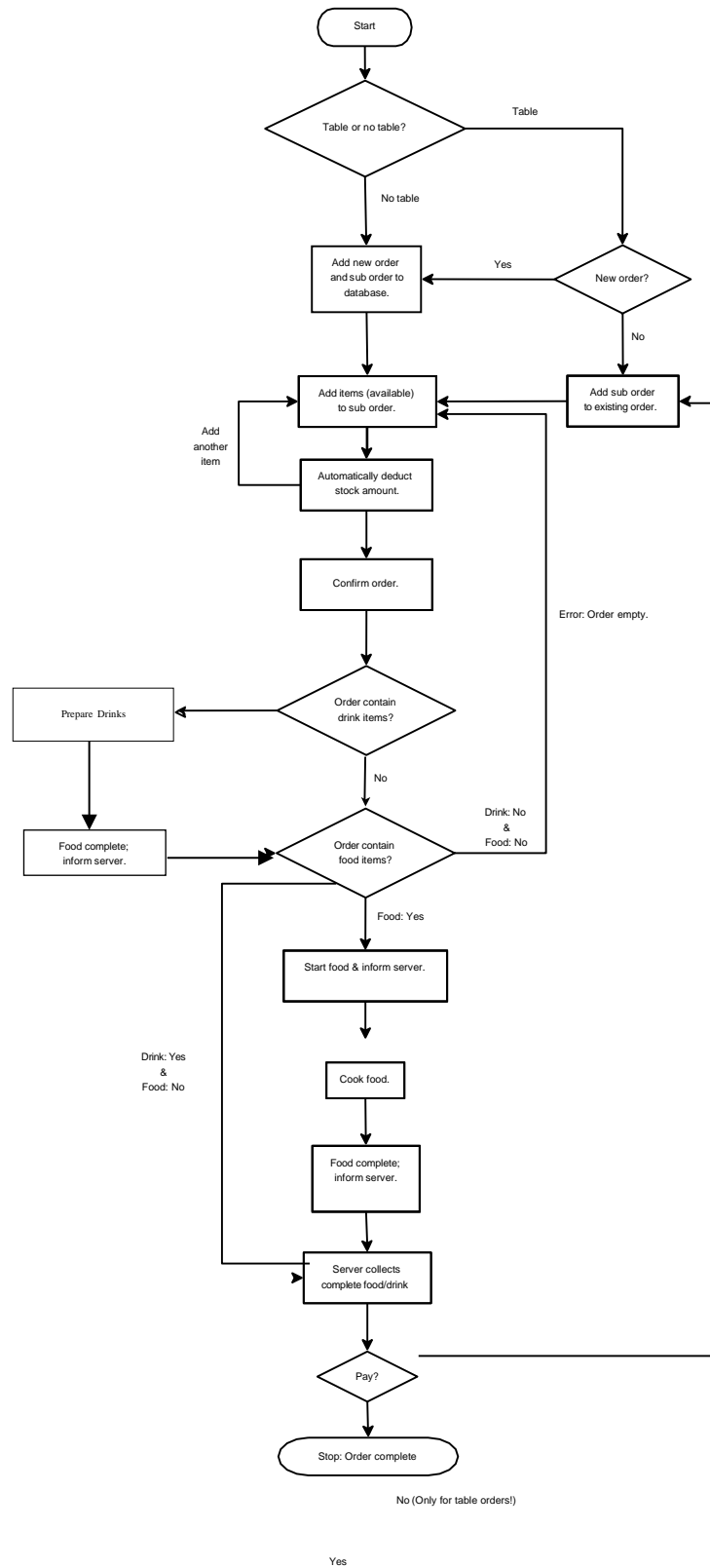


Fig. 5.12: Flow chart to show the flow of events of an order.

5.9 Design

5.9.1 Design Techniques

To model the system, design techniques are used. Object Oriented design, Rapid application development, prototyping and Modern structured design are examples to design techniques that currently used.

For this development Object oriented design techniques has been used. In this method there are advantages such as, code reusable facility, design benefits and maintainable facilities with Objects and classes.

Unified Modelling Language (UML) plays a significant role in Object Orient designing. UML allows programmers easy understand models of objects so that programmers can easily write software. Some of structural and behavioral UML diagrams use for design proposes systems are mentioned below.

- Use case diagrams – This makes clear what are the capabilities of the system.
- Use case narratives – Make use cases more clear
- Activity diagrams – shows how activities are coordinated to provide a service which can beat different levels of abstraction.
- DFD's – Data flow diagrams representing a flow of data through a process or a system
- Data flow Diagrams – shows how the objects interact overtime.
- Class diagrams – this is the main building block in object oriented modelling [8].

5.9.1.1 Use case diagrams

Overall high level Use-Case is shown in this figure 4.13.

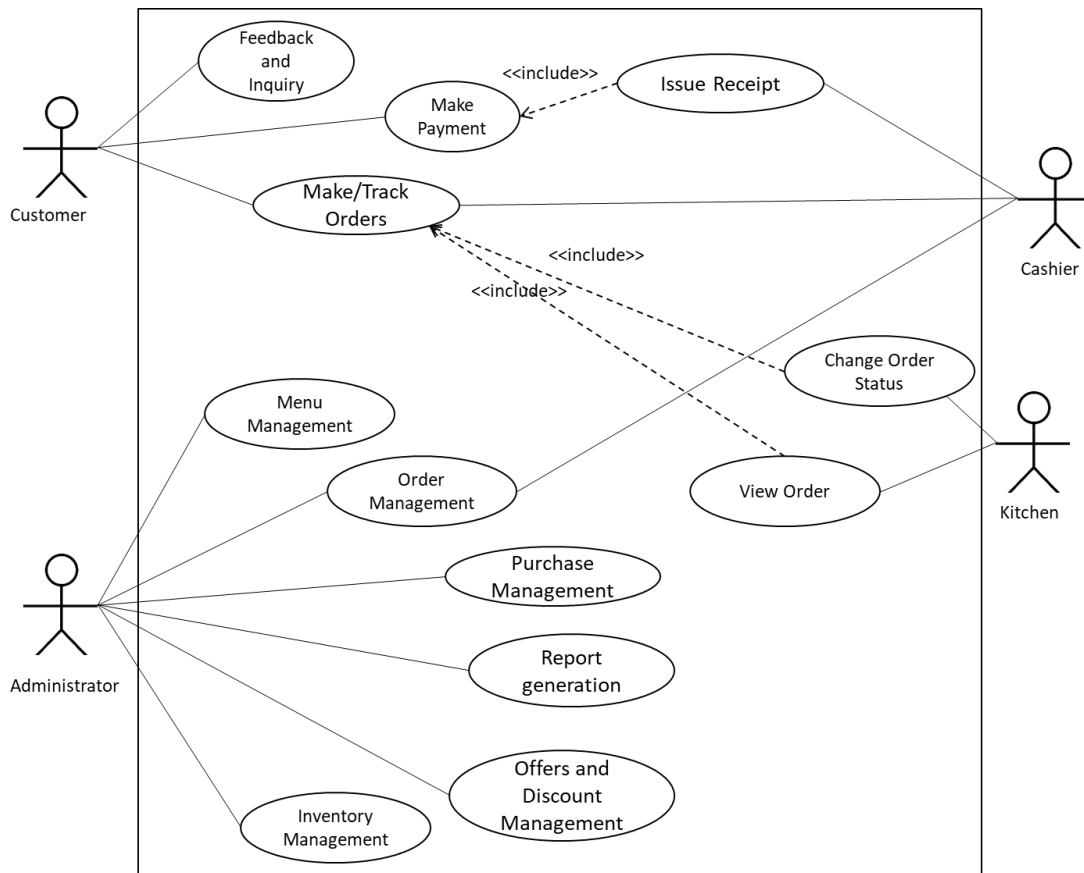


Fig. 5.13: Overall High-level Use case for the proposed system

5.9.1.2 Class Diagram

Class diagram is the main building box of object- oriented modelling. General conceptual modeling of the structure of the application, and for detailed modeling translating the models into programming code was done using this.

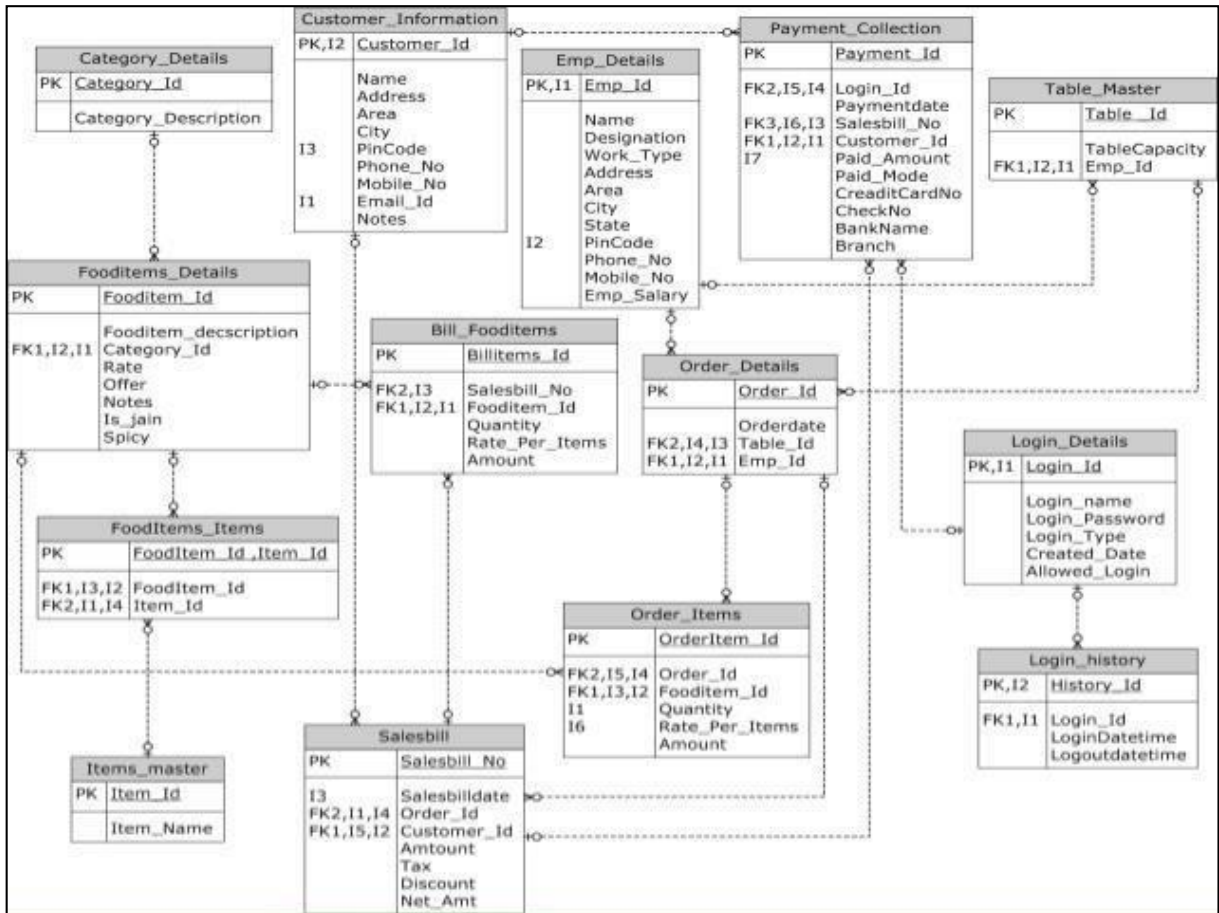


Figure 5.14: Class diagram

5.9.1.3 Activity Diagram

“Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

Activity diagram is suitable for modeling the activity flow of the system. An application can have multiple systems. Activity diagram also captures these systems and describes the flow from one system to another. This specific usage is not available in other diagrams. These systems can be database, external queues, or any other system” [8].

Following figure 5.15 shows activities to order food online.

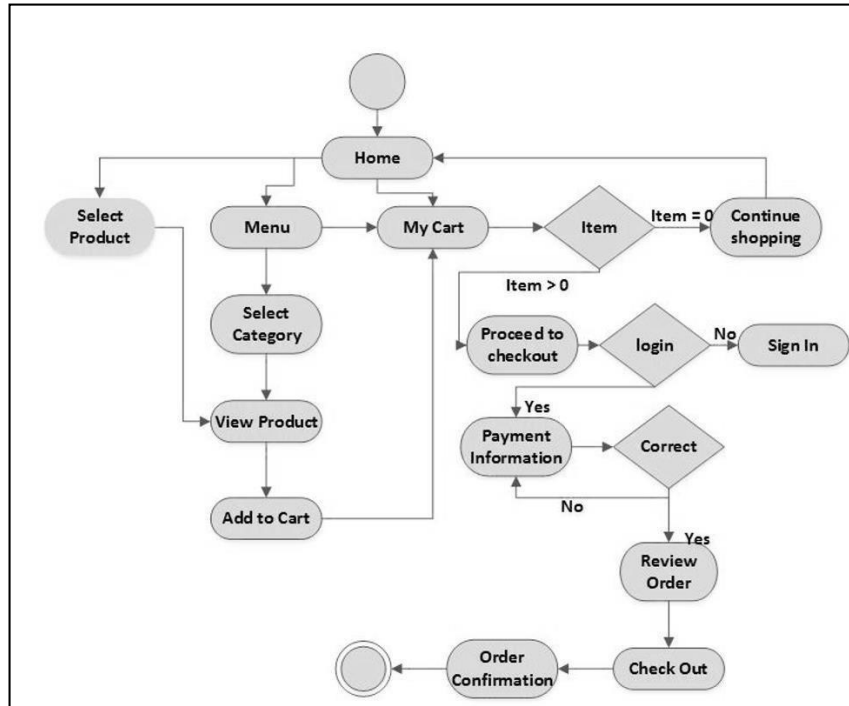


Figure 5.15: Activity diagram- Customer order

5.9.1.4 Sequence Diagram

A Sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence [13]

5.9.1.5 Sequence Diagram for Admin

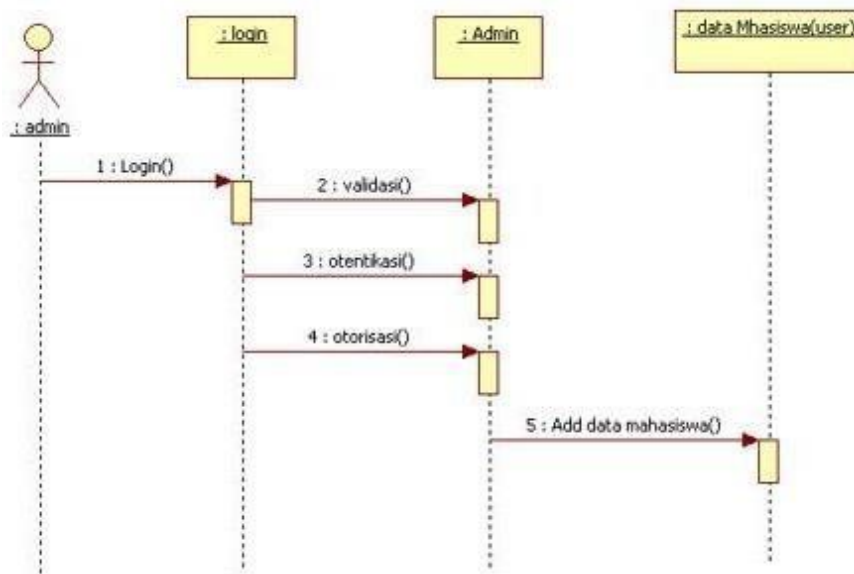


Fig. 5.16: Sequence Diagram for Admin

5.9.1.6 Sequence Diagram for Customer

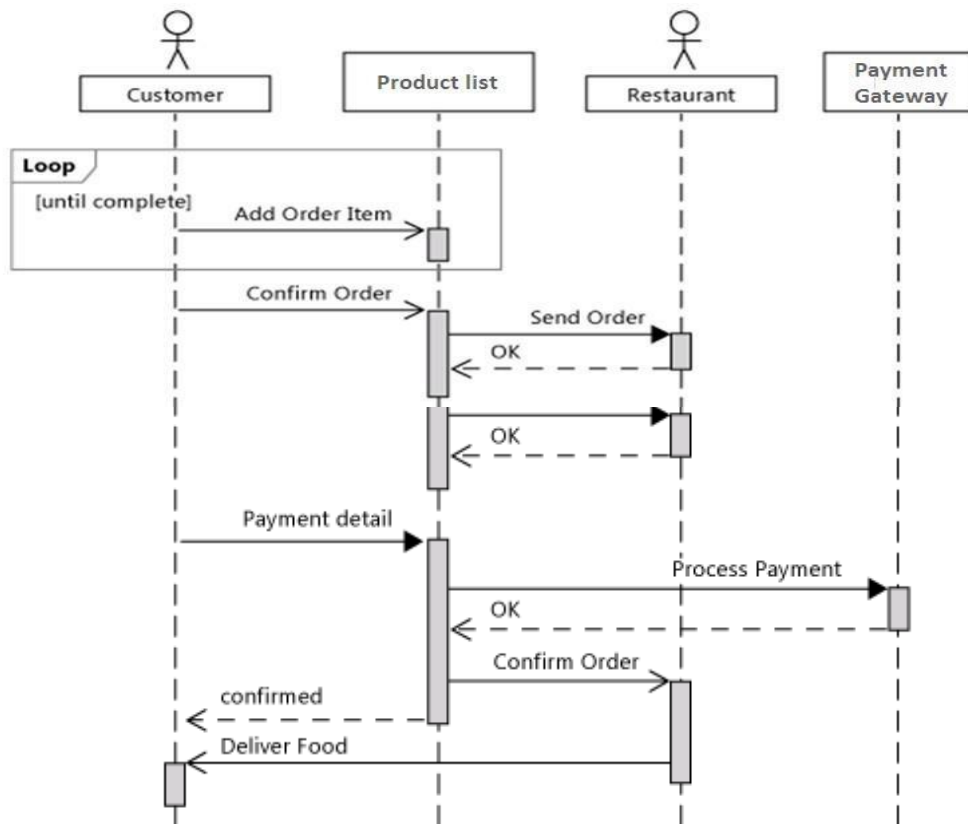


Fig. 5.17: Sequence Diagram for Customer

5.9.1.7 Sequence Diagram For Restaurant Management System

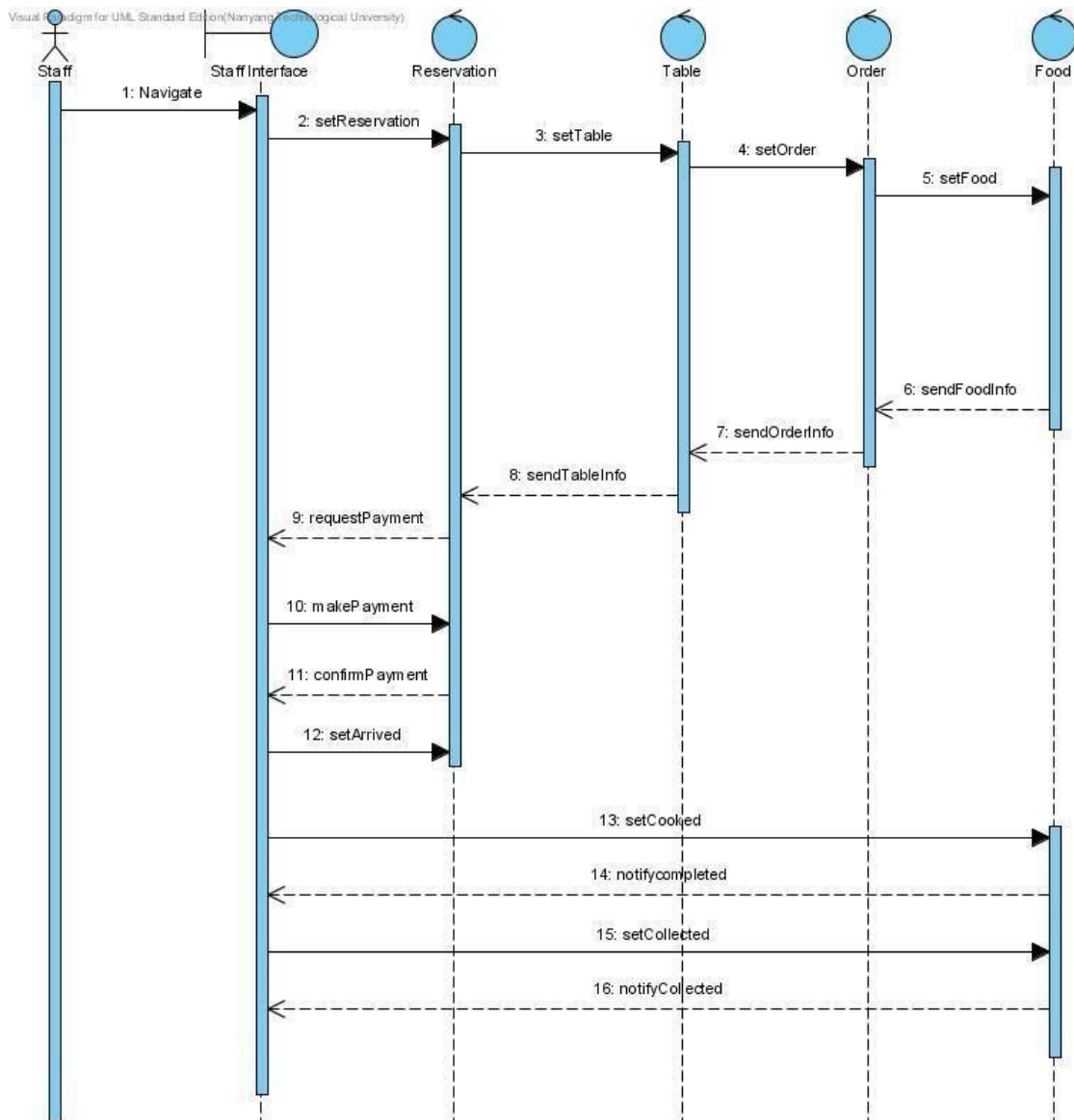


Fig 5.18: Sequence Diagram of Online Restaurant Management System

5.10 Chapter Summary

This chapter has displayed many graphical representations of the design of the system. The implementation of the system is documented in the next chapter.

6. Implementaion

6.1 Chapter Overview

This chapter discusses the theory behind the implementation of interesting and complex features using pseudocode and diagrams for clear explanations. The implementation stage of any project can be the most enjoyable part and with a well structured plan and appropriate development methodology can be a trouble free stage.

6.2 Implementing Extreme Programming

As discussed in section 2.8, this project used an agile methodology called Extreme Programming (XP). This type of methodology utilises an iterative process involving customer communication and feedback. The iterations were short with a new software version built at the end of each phase. Version control helped with debugging as the versions could be rolled back to find the iteration that caused the bug.

6.3 Data Storage and Retrieval

As discussed in Section 4.4, a RDBMS was the preferred storage method chosen for this project. The two ER diagrams (Figure 4.3 and 4.4) that were designed in Section 4.4.5 helped with the implementation of the database. A screenshot of the actual implemented database structure can be found in Appendix C.

The structure of the database was just the first step of many with data retrieval being high in priority. Recall from the background on requirement gathering (Section 2.7), most software systems abide by Boehm's cost model (Figure 2.2) so a £1 cost of change in the requirement gathering can cost up to £1000 in the deployment stage to fix. Therefore when implementing the code to retrieve data from the database, we have to consider the case that in the future this

system will need maintenance and added features to keep the product competitive within the market. The problem is we cannot predict the future; all we can do is rely on a well structured code base written using all the skills and techniques from the field to ensure easy readability of the code.

To implement this requirement for database retrieval, the code used the object oriented advantages of Java to create a database object that provided the database connection and data retrievals. Along with this database object, inheritance played an important role as a generic class created the SQL statements using an array of table field names and input values. Any object that required to send or retrieve data to the database could extend this generic class giving that object the functionality to insert, delete and update any table within the database.

6.4 Stock Control

This section focuses on how stock control was dealt with by using triggers in the MySQL database. A trigger is some procedural code that is executed in response to data changes within a table. The trigger can react upon inserts, deletes and updates of data within a table and are normally used to code business rules or data integrity checks.

This project used these responsive triggers to keep a stock check field within certain database tables upto date. The stock check field holds an integer depending on the stock level and would update when:

1. An item was selected even before optional ingredient confirmation.
2. An optional ingredient change, therefore a decrease of stock for the ingredient selected and an increase in the original ingredient stock.
3. An optional ingredient removed, therefore an increase in the removed ingredient stock.
4. An order cancelled, therefore an increase in the stock of all the ingredients within that order.
5. An item removed from the order, therefore an increase in the stock of all the ingredients within that item.

When the stock levels changed in any of these circumstances, a database function updated the availablestock field to a value between 0 and 4 inclusive. Every ingredient and prepared ingredient link table would hold a similar stock status between 0 and 2 inclusive as the ingredient is only defined compulsory or optional within the link between the ingredient and item.

Value	Meaning
0	Compulsory ingredient out of stock.
1	Compulsory ingredient low on stock.
2	Optional ingredient out of stock.
3	Optional ingredient low on stock.
4	No stock issues.

Table 6.1: A Table showing the available ingredients stock status.

The database functions that updated the available stock fields were:

- Reduce ingredient stock: When an item is selected, this function deducted the specified amount from the ingredient stock level.
- Increase ingredient stock: When an optional ingredient was removed or swapped, this function increased the specified amount from the ingredient stock level.
- Reduce prepared ingredient stock: When an item was selected, this function deducted the specified amount for all the ingredients within the prepared ingredient. This function also accommodated the requirement were a prepared ingredient could be an ingredient of another prepared ingredient. Increase prepared ingredient stock: When an optional prepared ingredient was removed or swapped, this function would increase the specified amount for all the ingredients within the prepared ingredient. This function also accommodated the requirement were a prepared ingredient could be an ingredient of another prepared ingredient.
- Update available stock: When an ingredients stock level changed, this function updated the available stock field within the meal and prepared

ingredient table.

- Update available stock (meal only): When a meal added or removed a prepared ingredient to its ingredients list, this function updated the available stock field within the meal table.

6.5 Development tools and techniques

6.5.1 Tools

- PhpStorm - PhpStorm is a cross-platform IDE for PHP. PhpStorm provides an editor for PHP, HTML and JavaScript with on-the-fly code analysis, error prevention and automated refactoring for PHP and JavaScript code.
- MySQL - all database related developments were handled using this because development tools, and administration objects are available with this software.
- Adobe Photoshop CC 2018 was used to edit photos

6.5.2 Technologies

- HTML – Hyper Text Markup Language is used to build the base interfaces of the system
- CSS – Cascading Style Sheets was used to make the system more attractive and userfriendly.
- Php –used to build the system
- JavaScript- this is a programming tool to web development and used for validations.
- JavaScript based AJAX – Used to get data from server. Use of this avoid the requirement of refreshing it continuously.
- JQuery - Which is also based on JavaScript was used to implement some features.

6.6 Reusable Components

- Date picker – To select a date from the calendar.
- Data table – Table adept of searching and filtering data, pagination of data.
- CSS – Cascading Style Sheet to design an interactive user interface.

- Web Template –Front end Design which can edit
- Charts - Use to generate HTML based JavaScript based interactive charts to display information graphically to user.

6.7 Network Architecture

Users can interact with the restaurant management system using web browser. Can be access via Internet and this system has central web server, for users to connect using their internal network in future.

6.8 Main User Interfaces

Only a few number of user interfaces and wireframes are provided in this section to show the structure of the system. Please Refer Appendix C - User Documentation for the rest of the interface designs.

6.8.1 Admin Dashboard

Controlling the system can be done using this panel. Administrator have privileges to do following functions

1. Inventory Management
2. Menu Management
3. User Management
4. Order Management
5. Table Management
6. Category Management
7. Kitchen Management

6.8.2 Order Food Cashier

Cashier can easily order food through this interface(fig. 6.2)

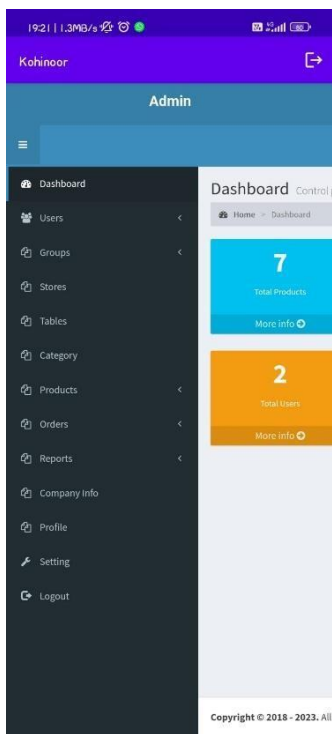


Fig. 6.1: Admin dashboard

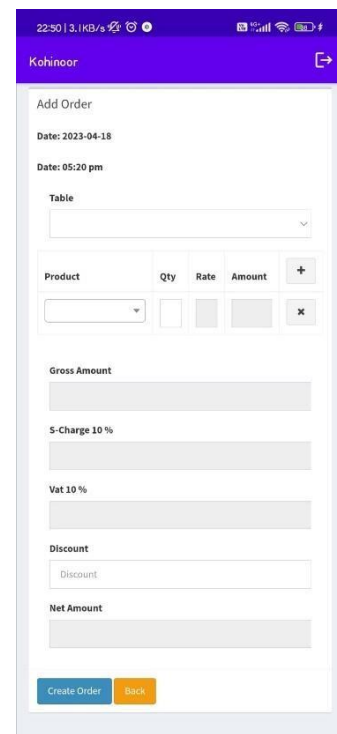


Fig. 6.2: Cashier Order Module

6.9 Summary

This chapter has discussed the interesting aspects from the implementation stage. The next chapter documents the results by demonstrating the working system.

7. User Evaluation and Testing

7.1 Introduction

Software testing is a critical component of software quality assurance that represents the ultimate analysis of specification, design, and code generation of software product. The testing method is basically combine with Verification and Validation. Validation refers to testing whether the system satisfies the requirements while verification refers to whether the system implements the specified functions properly. Basic goals of test evaluation is determining whether the promises about the invention by the supplier and the requirements of the customer are met on an acceptable level

7.2 Testing Procedure

Software testing and implementation are iterative processes. Most of the time both stages work as simultaneous system components. The propose system was tested while the development was ongoing. Following iterative RUP development life cycle made it possible to test Iterative increments of the software.

Structural testing techniques were followed in this phase including "white box" testing which tests "how a program/system does something". Functional testing techniques were used, which includes "black box" testing which tests the behavior of a system or program. These techniques were exploited in different testing levels like unit testing.

System followed object oriented methodology, so object oriented testing was also carried out in this phase. Individual operations associated with objects were tested initially, followed by testing individual classes and clusters of objects, and finally test the system as whole. User acceptance testing was completed in client site, participation of few staff members

7.3 Test Plan and Test Cases

Implementation stage starts the testing process. In development stage code is

reviewed. Test plan included all phases of testing and also used as a guide for the overall testing process. The test plan was designed before the implementation of the system. Test Objectives, Test Schedule, Test Logistics and Test Strategies are included in Test plan. Test cases are highly considered in here.

Test cases were created according to the designed test plan. That contains data, procedure, and expected result and represents which use to system or part of the system run. To reduce complexity of the testing process test cases were designed for each module independently. The following tables specify some test cases.

Manual testing method and procedures used for testing rather than automation tools and technologies.

Below tables tabulates some test cases for test basic functions.

7.3.1 Master Data modules

These tables tabulate some of the test cases for testing basic functions.

Test Case ID	1		
Tested Component	Add Customer		
Module Name	Customer Module		
Test Case	Add new customer		
Expected output	Add customer successfully, View should be user friendly. If adding customer form missing any required information error message must be displayed.		
Test Case Description			
No	Test Case	Actual Output	Status
01	(Positive) Add customer success	Enter customer information (Title, First Name, Last Name, Street, Town, Phone No, Email, Remarks) (Title, First Name, Last Name, Street, Town, Phone No, Email, must be filled save customer)	Pass

		Validate Then redirect to relevant home page(Admin – Add customer)	
02	Negative Log-In Not success	Enter customer information (Title, First Name, Last Name, Street, Town, Phone No, Email, Remarks (Not valid) Text box show error Message- “Required field”	Pass

Table 7.1: Add customer test case

7.4 Acceptance Testing

Actual working environment of Matara Kema Restaurant was used for acceptance Testing. Real Transaction data sets were used and users were restaurant staff. Employees were asked to work with related modules in system according to their privileges. Activities were monitored while users working with the system.

Final outcome must be handed over to actual end users for their acceptance. Due to Covid-19 problem, still it is not practical to meet real end users in actual situation.

7.5 Summary

This chapter explains about full testing cycle confirmed that developed system is functioning successfully. Next chapter will give the conclusion

8. Software Maintenance

Software maintenance in software engineering is the modification of a software product after delivery to correct faults, to improve performance or other attributes.^[11]

A common perception of maintenance is that it merely involves fixing defects . However, one study indicated that over 80% of maintenance effort is used for non-corrective actions. This perception is perpetuated by users submitting problem reports that in reality are functionality enhancements to the system. More recent studies put the bug-fixing proportion closer to 21%. [16].

8.1 Software Maintenance Process

This section describes the six software maintenance processes as:

1. The implementation process contains software preparation and transition activities, such as the conception and creation of the maintenance plan; the preparation for handling problems identified during development; and the follow-up on product configuration management.
2. The problem and modification analysis process, which is executed once the application has become the responsibility of the maintenance group. The maintenance programmer must analyze each request, confirm it (by reproducing the situation) and check its validity, investigate it and propose a solution, document the request and the solution proposal, and finally, obtain all the required authorizations to apply the modifications.
3. The process considering the implementation of the modification itself.
4. The process acceptance of the modification, by confirming the modified work with the individual who submitted the request in order to make sure the modification provided a solution.
5. The migration process (platform migration, for example) is exceptional, and is not part of daily maintenance tasks. If the software must be ported to another platform without any change in functionality, this process will be used and a maintenance project team is likely to be assigned to this task.
6. Finally, the last maintenance process, also an event which does not occur on

a daily basis, is the retirement of a piece of software [16].

The maintenance process model described in IEEE, the Standard for Software Maintenance, starts the software maintenance effort during the post-delivery stage and discusses items such as planning for maintenance and measures outside the process model. That process model with the IEEE maintenance faces is depicted in Figure.

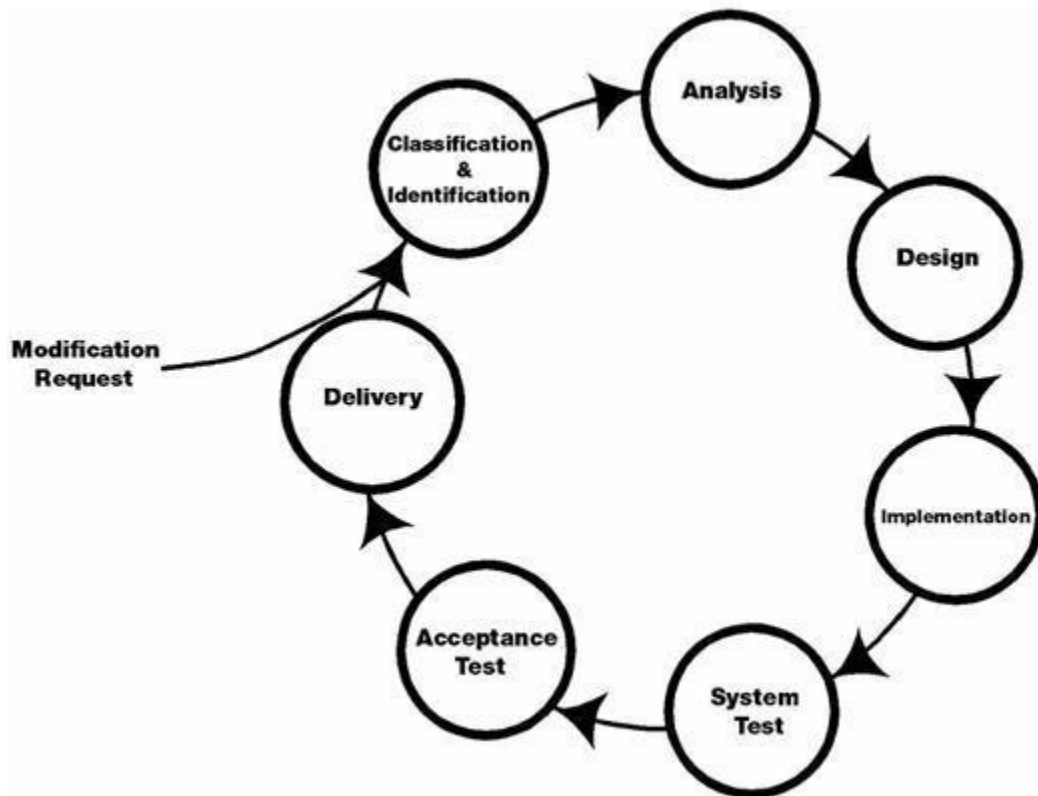


Fig. 8.1: Maintenance process model

8.2 Maintenance Activities

In the late 1970s, a famous and widely cited survey study by Lientz and Swanson, exposed the very high fraction of life-cycle costs that were being expended on maintenance. They categorized maintenance activities into four classes:

- Adaptive Maintenance
- Corrective Maintenance
- Perfective Maintenance
- Preventive Maintenance

Corrective maintenance: Reactive modification of a software product performed after delivery to correct discovered problems.

Adaptive maintenance: Modification of a software product performed after delivery to keep a software product usable in a changed or changing environment.

Perfective maintenance: Modification of a software product after delivery to improve performance or maintainability.

Preventive maintenance: Modification of a software product after delivery to detect and correct latent faults in the software product before they become effective faults.

There is also a notion of pre-delivery/pre-release maintenance which is all the good things you do to lower the total cost of ownership of the software. Things like compliance with coding standards that includes software maintainability goals. The management of coupling and cohesion of the software. The attainment of software supportability goals (SAE JA1004, JA1005 and JA1006 for example). Note also that some academic institutions are carrying out research to quantify the cost to ongoing software maintenance due to the lack of resources such as design documents and system/software comprehension training and resources (multiply costs by approx. 1.5-2.0 where there is no design data available) [16].

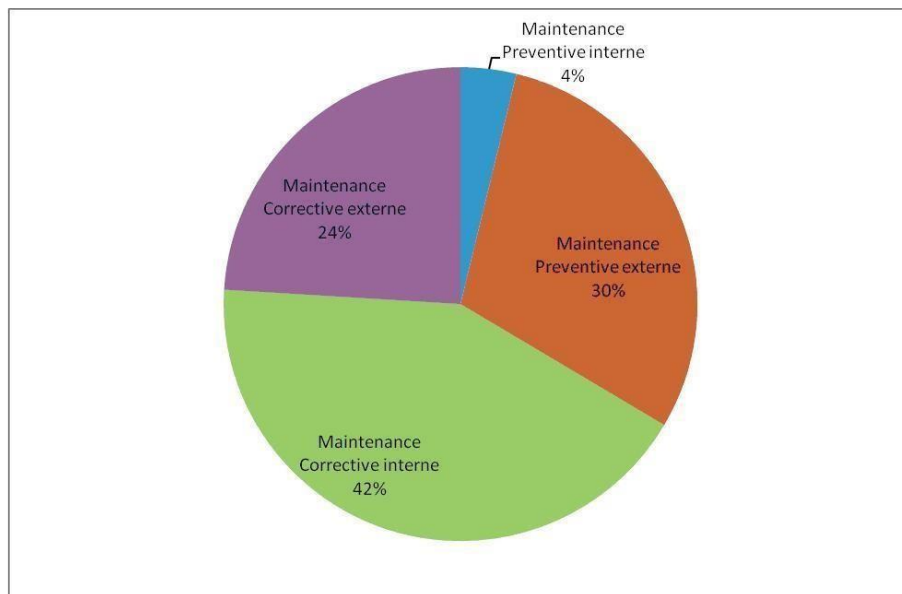


Fig. 8.2: Maintenance Activities

8.3 Difficulties of Maintenance

There are four key difficulties that can sluggish down the maintenance process.

- Formless code
- Maintenance programmers having inadequate knowledge of the system
- Documentation being inattentive
- Out of Date or inadequate

The achievement of the maintenance stage trusts on these problems being earlier in the life sequence. In our system we try to follow the maintenance phases to complete those activities. The code is structured and we have adequate knowledge about programming [16].

9. Conclusion

9.1 Introduction

This is the final chapter of this dissertation. Objectives achieved, lessons learnt, Problems and future enhancements will be discussed here. Matara Kema restaurant is become more popular than previous times. They have a plan to start delivery services also. Therefore, it has been implemented as a web based system to meet the requirements of increasing demand

This system helps to manage entire restaurant easily. Restaurant operations also can be well manage using this solution.

9.2 Problems Encountered

The client has no any computer knowledge and due to that it was difficult to capture the requirement. Usually staff who works in this kind of place also less knowledgeable and they are afraid to work with computers. It was really difficult to explain about this system, and gather the requirements they need.

Due to medical condition, I had to omit some parts of the system because I was not able to work more than one month.

Covid-19 situation also made so many difficulties to development of this system because agile methodology is used. Testing was not able to done properly because even after 3 months still restaurant is not operating properly.

Lack of knowledge about development tools and hosting environment was also a huge problem. Books, Video lessons and support from knowledgeable people were used to overcome this problem to some extent.

9.3 Lessons Learnt

Throughout this project, valuable knowledge has been gained. This is the first experience to work with a client that has no IT knowledge. Learned many things about

fact gathering techniques in actual environment. This project gave me the opportunity to gain experience of actual web developer. It helps to understand how project planning should be done and had opportunity to collaborate with many Individuals and improve my skills. Had a great lesson about time management also.

This project gave chance to gain knowledge about development tools such as phpStrome, mySQL ect. This was the first experience about web hosting and domains.

Theories techniques and technologies that learnt throughout the MIT degree program was veryvaluable for this project.

9.4 Limitations

For upcoming improvement, there are some proposals to advance our project abilities.

- There is no email verification system.
- There is no online bill payment system.
- There is no security protection such as SSL, Sitelock.
- SMS alert system is not available right now.
- There is no online secured payment getting system.

9.5 Future enhancements

Delivery function has to added to the system because restaurant in planning to give that facility.

Most of the customers suggested about mobile application and planning to implement thatfunction also.

Report generation part must be implemented more than this.

User Manual for Application

Introduction

Restaurant Management System is an Android based application which is divided in three parts and as per our use we can add as much as user such as one login panel for Admin who can handle all the application and other panel for waiter, chef, manager.

The waiter can take the order and can select the table and submit that order to admin and the admin has access all the panel. There will be another login for Chef who can see the order and complete it. The delivery option is also available and in that the admin can add the delivery items in the cart.

Menu Card : On table the manager can print the QR code and place it on the table. After placing on the table the customer can scan that QR code and can see the Menu Card and from that the waiter takes the order.

Ordered item :

This table provides information to the management staff regarding the uniquely generated orderID, which may contain one or more menu item uniquely identified by ItemID.

Apart from this, the management can check/alter the status of the order along with, they can alter the cook/s which was previously assigned to fulfill the order (as each cook is uniquely identified by cookID), this can be done by the management staff by logging in with UserID.

A cook can place status over this entity regarding the status of the specified item is ready for delivery or not !!, this is done by using uniquely provided cookID.

From customer's perspective, customer can check the status of his order in detail i.e. the status of each item in his order from this table as well as the quantity he ordered, the later can also be helpful at the management side.

Order :

This table tells about the OrderID (which is a primary key), who has placed the order and gives details about the time when order was placed and the time when the order will be delivered; along with the status of the order (usually some 5 status labels are assigned).

This table shares N:1 relationship with the Customer_info entity.

Chef :

Every cook is uniquely identified by his ID called 'cookID' and the other field "Name" is another step that will help management to recognize the specific cook. This entity shares N:N / N:1 relationship with the ordered item and 1:N relationship with cook spec entity.

Management :

This Table represents the total management side of our project, UserID is the primary key in this table. Other fields include password and Name.

The status tells about which customer has to be blacklisted or blocked.

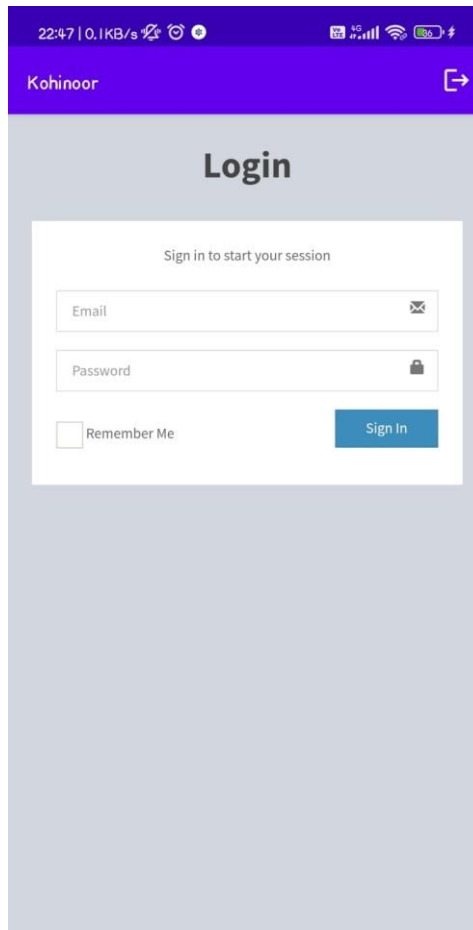
The "designation" field tells the level of user, say the administration side or a cook. The privileges are set according to the designation for eg : A cook cannot cancel the order and so.

The management staff can access the Menu and the ordered items and table.

Admin Side Activities

1. Login Page:

In login page the user has to login on app whenever the user is working. there are different logins for different users like admin has its separate login, waiter has its separate login, chef has its separate login, like this.



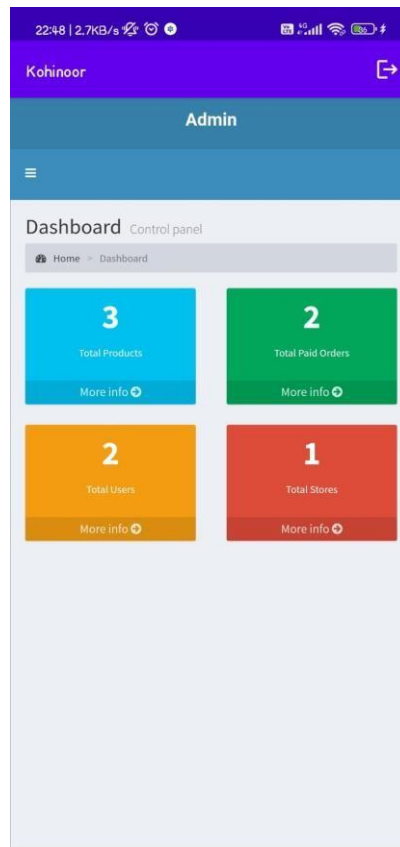
Step 1: Click on email and type your email.

Step 2: Click on Password and type your password

Step 3: Now Click on Sign in.

2. Dashboard:

After Login you will see the Dashboard Page in which how many products are available, how many users are there in Restaurant, which orders are paid.



This is the interface of Dashboard in which you can see Total Products, Total Paid Order, Total Users, Total Stores.

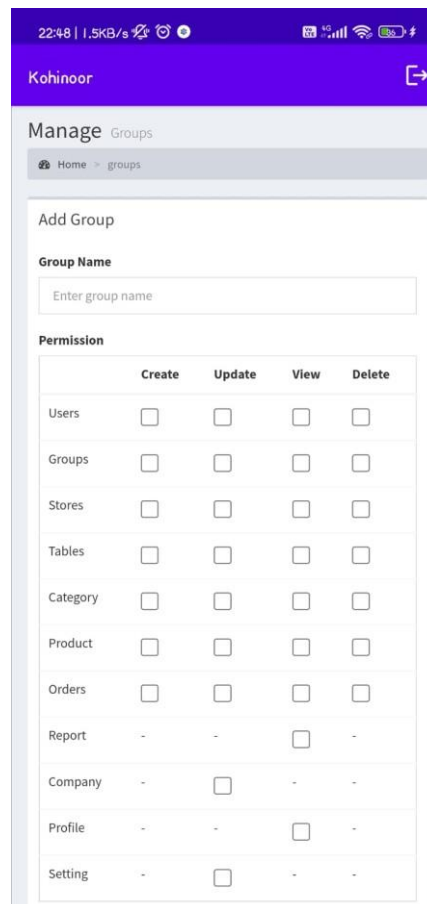
- Total Products : Total Products Available in Menu Card.
- Total Paid Order : Total Number of Order paid and Total Number of Order Unpaid.
- Total Users : Total Number of Users Working in Restaurant.

3. Groups:

In groups the admin can give access to the staff member by clicking on them and give access to them what they can see on the panel. For example, there is a chef if admin has selected that he only can see order but by groups option if he get access on Create, Update, View and Delete.

Following the steps for adding and managing the groups:

Add Group:



The screenshot shows the 'Manage Groups' screen in the Kohinoor application. It features a purple header with the app name and a navigation menu. Below the header, there is a breadcrumb trail 'Home > groups' and a section titled 'Add Group'. This section includes a text input field for 'Group Name' and a table for assigning permissions to various modules.

	Create	Update	View	Delete
Users	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Groups	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Stores	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Tables	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Category	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Product	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Orders	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Report	-	-	<input type="checkbox"/>	-
Company	-	<input type="checkbox"/>	-	-
Profile	-	-	<input type="checkbox"/>	-
Setting	-	<input type="checkbox"/>	-	-

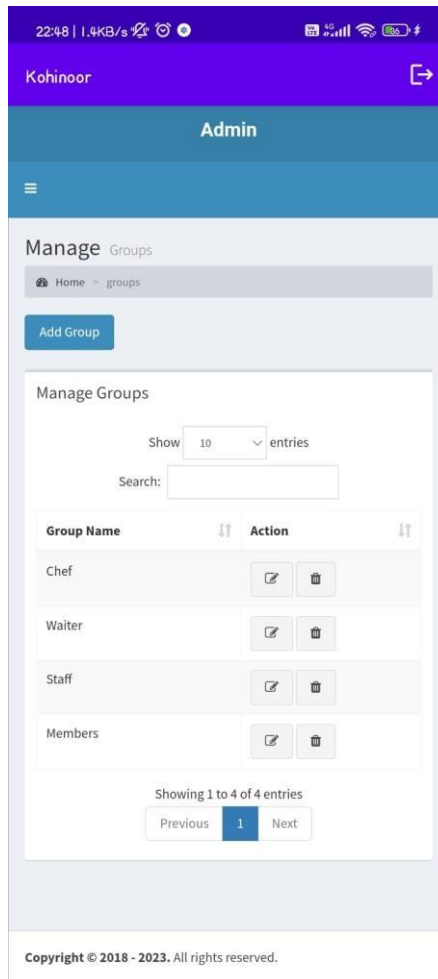
Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Group option

Step 3: After that click on “Add Group”

Step 4: From here you can add group and give access to the modules which are suitable for that group.

Manage Groups:



Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Group option

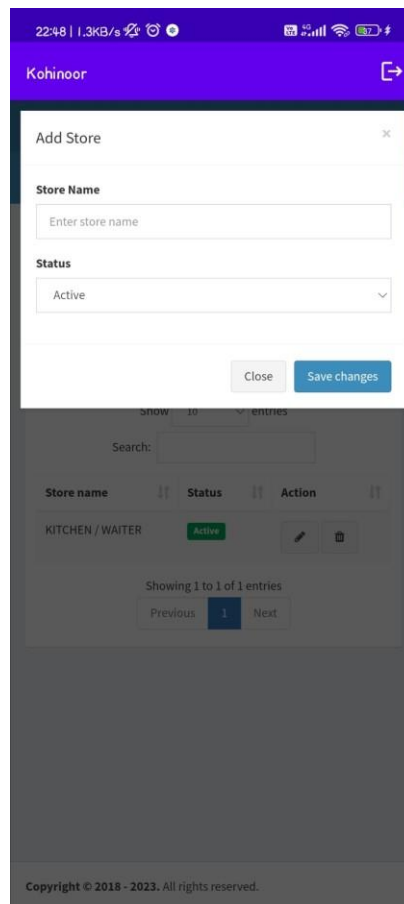
Step 3: After that click on “Manage Group”

Step 4: From here you can Manage group and edit the given access to that user or delete the group.

4. Stores:

In Store there are different stores like veg kitchen, non-veg kitchen, dessert store, drink store, etc. there the two modules in this store to add store and to manage the store.

Add Store:



The screenshot shows a mobile application interface for adding a new store. The form is titled "Add Store" and is overlaid on a blurred background of a store list. The form contains the following fields and controls:

- Store Name:** A text input field with the placeholder "Enter store name".
- Status:** A dropdown menu currently set to "Active".
- Buttons:** "Close" and "Save changes".

The background shows a table with the following data:

Store name	Status	Action
KITCHEN / WAITER	Active	[Edit] [Delete]

Showing 1 to 1 of 1 entries

Previous 1 Next

Copyright © 2018 - 2023. All rights reserved.

Step 1: First click on 3 lines on left top corner.

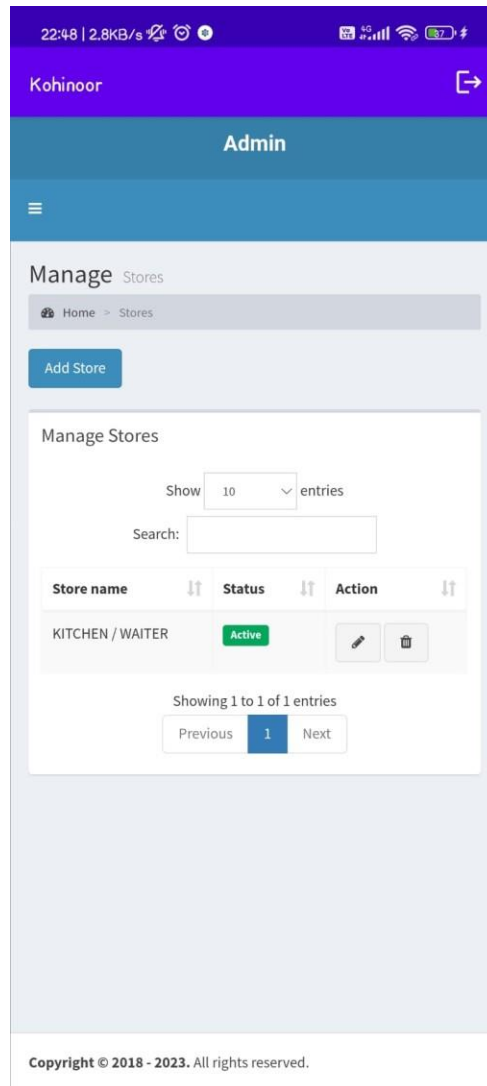
Step 2: Then Click on Store option

Step 3: After that click on "Add Store"

Step 4: Give the name to store and select the status weather active or inactive.

Step 5: Click on Save Changes.

Manage Store:



Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Store option

Step 3: After that click on “Manage store”

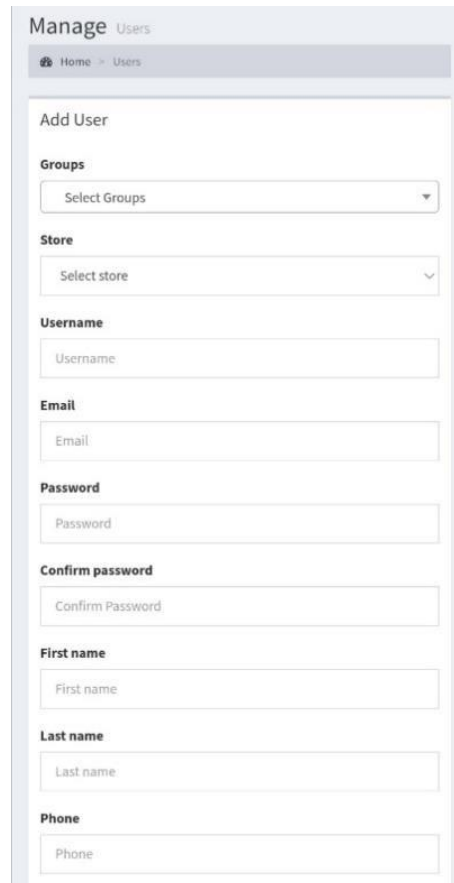
Step 4: From here you can Manage store, edit or delete the store.

5. User:

In Users there are two types Add User and Manage User, in add user you can add user such as he is waiter, chef, staff or other member, after that add name, login id, password, phone number, gender and click on save changes.

Manage User shows the total number of Users information.

Add User:



The screenshot displays the 'Manage Users' interface. At the top, there is a breadcrumb trail: Home > Users. Below this, the 'Add User' form is visible. The form contains several input fields and dropdown menus:

- Groups:** A dropdown menu with the text 'Select Groups'.
- Store:** A dropdown menu with the text 'Select store'.
- Username:** A text input field with the placeholder 'Username'.
- Email:** A text input field with the placeholder 'Email'.
- Password:** A text input field with the placeholder 'Password'.
- Confirm password:** A text input field with the placeholder 'Confirm Password'.
- First name:** A text input field with the placeholder 'First name'.
- Last name:** A text input field with the placeholder 'Last name'.
- Phone:** A text input field with the placeholder 'Phone'.

To add user follow the following steps:

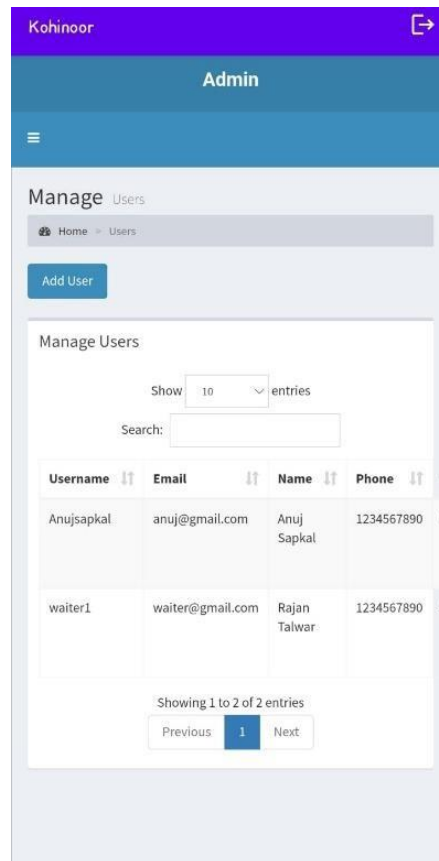
Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on user option

Step 3: After that click on “Add User”

Step 4: After clicking Add User you can add the personal info of staff, login id, password.

Manage User:



To add user follow the following steps:

Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on user option

Step 3: After that click on “Manage User”

Step 4: From Manage User section you can edit the users information, remove the user.

You can also see the details of the user.

6. Table:

In Tables the admin can add table and can see the table which is active and in active. After customer complete food the waiter can edit and upload that the table is in-active now. There are many options like table number, capacity, available or not and action option.

Add Table:

The screenshot shows a mobile application interface for adding a table. The form has the following fields:

- Table Name:** Enter table name
- Capacity:** Enter capacity
- Status:** Active (dropdown menu)
- Store:** KITCHEN / WAITER (dropdown menu)

At the bottom of the form are 'Close' and 'Save changes' buttons. Below the form is a table listing existing tables:

ID	Capacity	Status	Action
1	4	Unavailable	Active
2	6	Unavailable	Active

At the bottom of the table, it says 'Showing 1 to 5 of 5 entries' and has 'Previous', '1', and 'Next' navigation buttons.

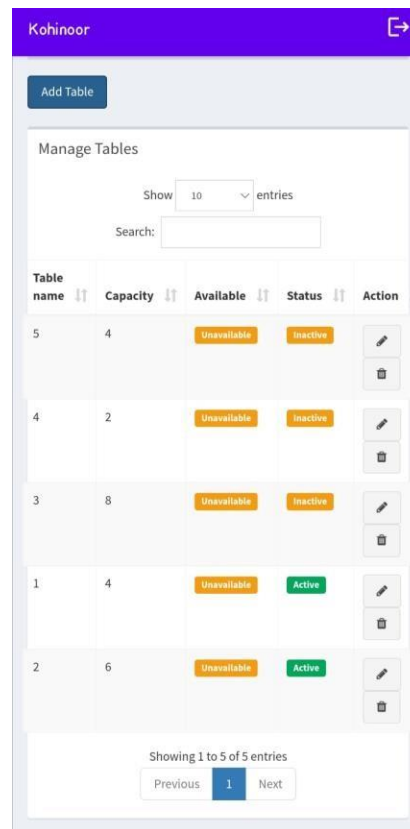
Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Table option

Step 3: After that click on “Add Table”

Step 4: From here you can add table and give name to table, select the capacity of table.

Manage Table:



Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Table option

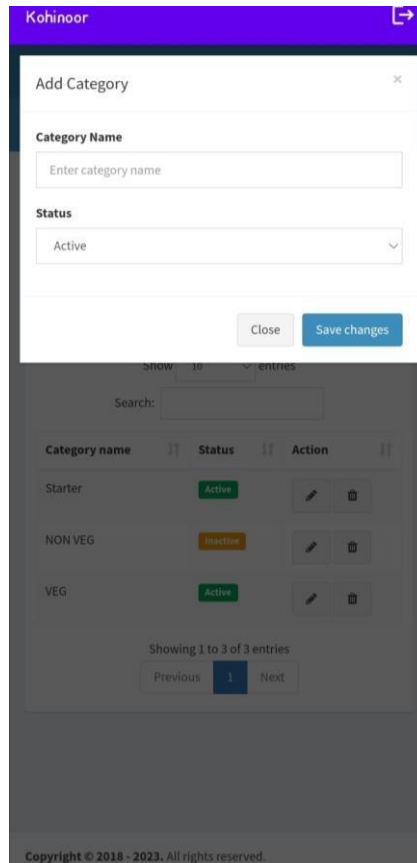
Step 3: After that click on “Manage Table”

Step 4: From here you can manage table and select weather the table is active or inactive, you can also delete the table.

7. Category:

In Category Option admin can edit that which type of food is not available for that day. For example, there is any festival for that day the only veg option is available and only that option is seen to waiter.

Add Category:



The screenshot shows a mobile application interface for managing categories. The top bar is purple with the text 'Kohinoor' and a close icon. Below it is a white modal window titled 'Add Category' with a close button. The modal contains a text input field for 'Category Name' with the placeholder 'Enter category name', a dropdown menu for 'Status' currently showing 'Active', and two buttons at the bottom: 'Close' and 'Save changes'. The background is a blurred view of a category list with columns for 'Category name', 'Status', and 'Action'. The list shows three entries: 'Starter' (Active), 'NON VEG' (Inactive), and 'VEG' (Active). At the bottom of the list, it says 'Showing 1 to 3 of 3 entries' and has 'Previous', '1', and 'Next' navigation buttons. A copyright notice 'Copyright © 2018 - 2023. All rights reserved.' is visible at the very bottom.

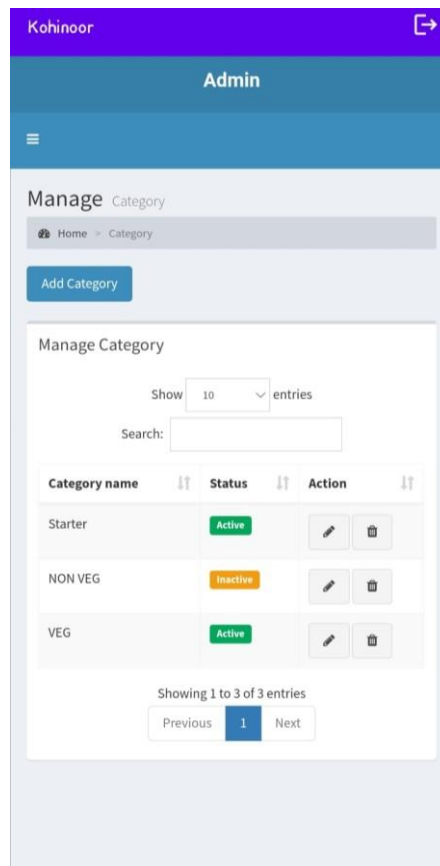
Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Category option

Step 3: After that click on “Add Category”

Step 4: From here you can add category of food like veg, non-veg, starter, etc.

Manage Category:



Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Category option

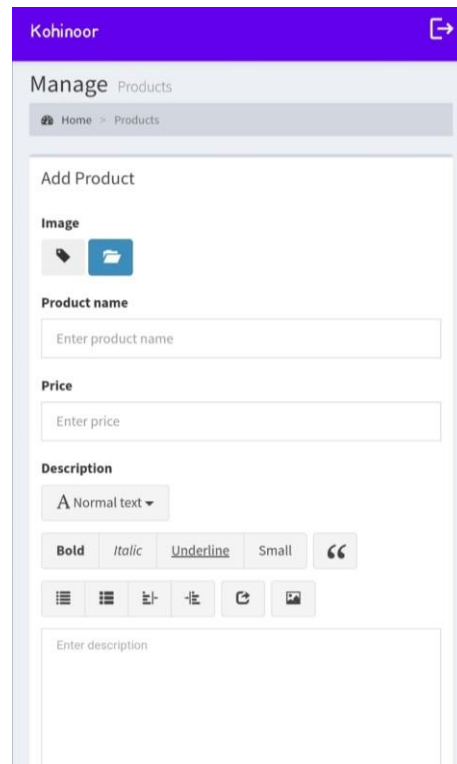
Step 3: After that click on “Manage Category”

Step 4: From here you can manage category of food. You can delete the category if you want.

8. Product:

In product option there are two options available one is manage product and another is add product. In add product we can add the image of that product, add name, add price, description of that dish, and in manage it shows the active, in-active status.

Add Product:



Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Product option

Step 3: After that click on “Add Product”

Step 4: From here you can add product

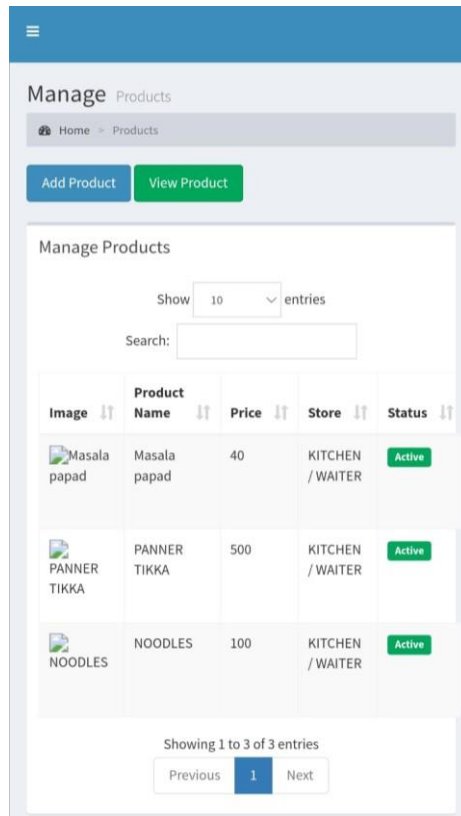
Step 5: In Add product you can select the image for the product by clicking the image button.

Step 6: Give the name to product and enter the amount for the product. You can also give the description of the product.

Step 7: Select the category of product.

Step 8: Click on save changes to add the product.

Manage product:



Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Product option

Step 3: After that click on “manage Product”

Step 4: From here you can manage product like you can edit the details of product, delete the product.

Step 5: To edit details of product click ok edit option and do the changes whatever you want

Step 6: After doing changes click on save changes to save.

9. Order:

In order it shows the enter of total orders paid or not paid. If there is any delivery option so the waiter can direct add order total amount with vat charges, and this same is also available for the table option select the table product, it will automatically calculate the amount, if there is any discount on that day at the restaurant so that option is also available. There are actions like for printing the bill also.

Add Order:

The screenshot shows a web form titled "Add Order". At the top, it displays the date "Date: 2023-04-18" and the time "Date: 05:20 pm". Below this is a "Table" dropdown menu. A table with four columns: "Product", "Qty", "Rate", and "Amount" is present, with a "+" button to the right of the "Amount" column. Below the table are several input fields: "Gross Amount", "S-Charge 10%", "Vat 10%", "Discount", and "Net Amount". At the bottom of the form, there are two buttons: "Create Order" (blue) and "Back" (orange).

Step 1: First click on 3 lines on left top corner.

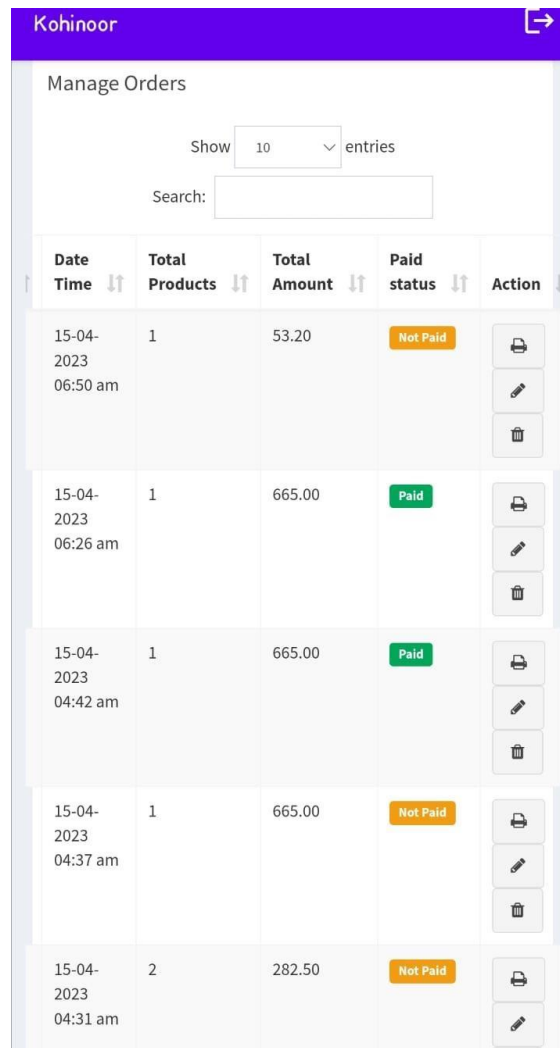
Step 2: Then Click on order option

Step 3: After that click on "add order"

Step 4: In Add Order you can select the table , select the product by clicking on Product Button, you can select the quantity of product and according the quantity the amount gets automatically generated.

Step 5: now click on create order to save the order.

Manage Order:

















Kohinoor

Manage Orders

Show 10 entries

Search:

Date Time	Total Products	Total Amount	Paid status	Action
15-04-2023 06:50 am	1	53.20	Not Paid	  
15-04-2023 06:26 am	1	665.00	Paid	  
15-04-2023 04:42 am	1	665.00	Paid	  
15-04-2023 04:37 am	1	665.00	Not Paid	  
15-04-2023 04:31 am	2	282.50	Not Paid	 

Step 1: First click on 3 lines on left top corner.

Step 2: Then Click on Order option

Step 3: After that click on “Manage Order”

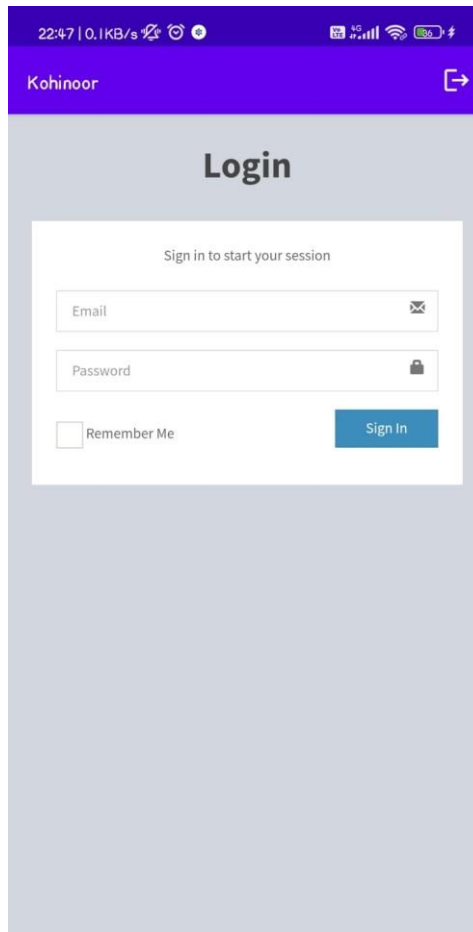
Step 4: Here you can see the list of orders, you can edit the order by clicking on edit option, or you can delete the order.

Step 5: You can print the bill by clicking on the bill option .

Waiter Side Activities

1. Login:

In login page the user has to login on app whenever the user is working. there are different logins for different users like admin has its separate login, waiter has its separate login, chef has its separate login, like this.



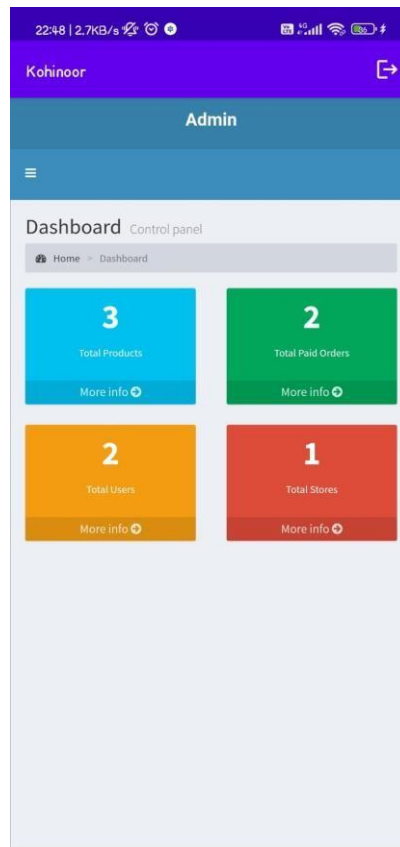
Step 1: Click on email and type your email (waiter@gmail.com)

Step 2: Click on Password and type your password (12345678)

Step 3: Now Click on Sign in.

2. Dashboard

After Login you will see the Dashboard Page in which how many products are available, how many users are there in Restaurant, which orders are paid.



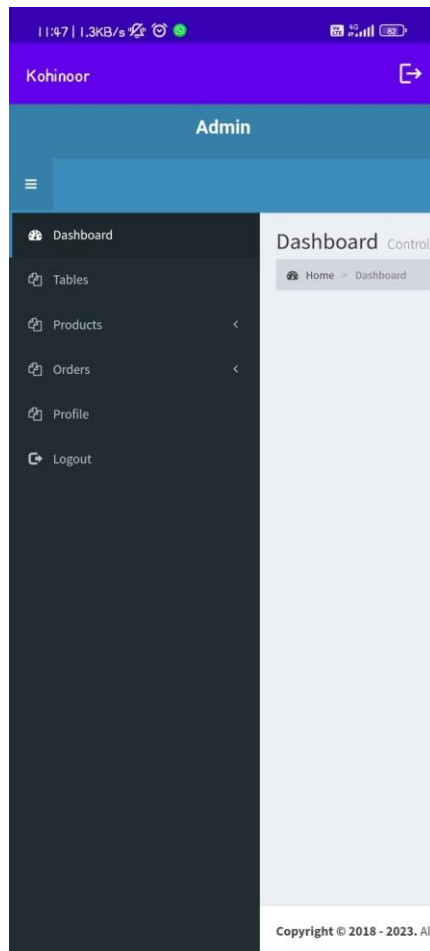
This is the interface of Dashboard in which you can see Total Products, Total Paid Order, Total Users, Total Stores.

- Total Products : Total Products Available in Menu Card.
- Total Paid Order : Total Number of Order paid and Total Number of Order Unpaid.
- Total Users : Total Number of Users Working in Restaurant.

3. Waiter Accessed Modules:

Waiter will be able to access to the following Modules:

1. To Add or Manage Table.
2. To Manage Products.
3. To Add Order and to Manage Orders.



Waiter will be able to do the similar activities which admin can do in the following listed modules.

References

- [1] Reference Ian Alexander. *Computing & control engineering*. Volume 14(Issue 1):22–26, April 2003.
- [2] Ezee Burrp. *Software Features*. Accessed on 14 October, 2009. <http://www.ezeeburp.com/features.html>.
- [3] HSQL db. *Database comparison*. Accessed on 13 September, 2009. http://hsqldb.org/images/imola_retrieve.jpg.
- [4] Business Dictionary. *Stakeholder*. Accessed on 2 April, 2010. <http://www.businessdictionary.com/definition/stakeholder.html>.
- [5] Geoffrey Elliot. *Global business information technology: an integrated systems approach*. PearsonEducation, 2004.
- [6] Beefeater Grill. *Beefeater Grille*. Accessed on 24 September, 2009. <http://www.beefeater.co.uk/ourFoodAndDrink>.
- [7] Horizon. *Restaurant Sector Overview*. Accessed on 4 May, 2010. <http://www.caterersearch.com/Articles/2006/05/12/306737/Restaurant-sector-overview.htm>.
- [8] Java JDBC. *Java SE Technologies - Database*. Accessed on 3 May, 2010. <http://java.sun.com/javase/technologies/database/>.
- [9] Paul Lemberg. *Which is Better: New customers or repeat business?*. Accessed on 12 October, 2009. <http://www.businessknowhow.com/marketing/new-customers.htm>.
- [10] Agile Manifesto. *Principles behind Agile Manifesto*. Accessed on 20 September, 2009. <http://www.agilemanifesto.org/principles.html>.
- [11] Craig Murphy. *Improving Application Quality Using Test-Driven Development (TDD)*. Accessed on 16 October, 2009. <http://www.methodsandtools.com/archive/archive.php?id=20>.
- [12] A Nutt. History of pos equipment. *Computers and Internet community*, 1:1, March 2009.

- [13] Point of Success. *Products and Features*. Accessed on 14 October, 2009. <http://www.pointofsuccess.com/softwarefeatures.htm>.
- [14] Abacre POS. *Major Features*. Accessed on 14 October, 2009. <http://www.abacre.com/restaurantpos/features.htm>.
- [15] Steve Rosenberg. *Fast food, German-style*. Accessed on 15 October, 2009. <http://news.bbc.co.uk/1/hi/7335351.stm>.
- [16] s Baggers restaurant. *Rollercoaster style restaurant*. Accessed on 13 October, 2009. <http://www.sbaggers.de/main-ger/?sid=home&lang=en>.
- [17] Horsforth School. *Normalisation*. Accessed on 1 December, 2009. <http://www.horsforth.leeds.sch.uk/subjects/comp/alevel/module5/unit59/unit59d.asp>.
- [18] Restaurant Pos Software. *HOSPOS*. Accessed on 14 October, 2009. <http://www.restaurant-pos-software.com/Products/HospitalitySoftware.aspx>.
- [19] Don Wells. *Unit Tests*. Accessed on 24 September, 2009. <http://www.extremeprogramming.org/rules/unittests.html>.

Dissemination of Work

- Research Paper Published in Journal:
 1. Paper Name: Restaurant Management System
Publication Name: International Journal for Research in Applied Science and Engineering Technology.
Issue Number: Volume 11, Issue IV, April 2023
DOI: 10.22214
ISSN NO: 2321-9653
Link: <https://doi.org/10.22214/ijraset.2023.51116>
 2. Paper Name: Literature Review on Restaurant Management System
Publication Name: International Research Journal of Modernization in Engineering Technology and Science.
Issue Number: 05/Issue:05/May-2023
DOI: 10.56726
ISSN NO: 2582-5208
Link: <https://www.doi.org/10.56726/IRJMETS37920>
- Participation in Project Competition:
Competition Name: Technovation 2023 Division 6
Organized By: Shri Sant Gajanan Maharaj College of Engineering, Shegaon

Restaurant Management System

M. Faizan Khandwani¹, Pratik Lanke², Pratik Harne³, Anuj Sapkal⁴, Adesh Adhao⁵

¹Assistant Professor, Dept. of IT, Shri Sant Gajanan Maharaj College of Engineering, Shegaon, Maharashtra, India

^{2, 3, 4, 5}Student, Dept. of IT, Shri Sant Gajanan Maharaj College of Engineering, Shegaon, Maharashtra, India

Abstract: *The project focuses on developing an Android application that allows customers to easily book tables and menus of their choice at a restaurant according to their convenience.*

The traditional manual system for table reservations in restaurants is becoming obsolete, and digital systems are gaining popularity. The manual system relies on waiters and booking diaries, with no automated record-keeping. Main objective of the project is creating the Reservation System that addresses the shortcomings of the manual system, such as inefficiencies in time and cost management.

The current generation prefers high-tech services, especially those available online. Therefore, the project aims to efficiently automate restaurant operations for owners. The proposed reservation system offers customers the convenience of reserving tables, menus, or both.

For menu bookings, customers are required to pay 50% of the total amount in advance. Customers can cancel their bookings if they are unable to arrive on time, and can rebook for the next available time slot, provided that the cancellation is made at least 30 minutes before the selected time. However, if the cancellation is made after this timeframe, the advance payment will not be refunded.

Keywords: *Restaurant Management, Recommendation, Tablet, Menu, Intelligent, Android application.*

I. INTRODUCTION

This application serves as a convenient self-service system for table and menu bookings in restaurants. The system allows for customization of booking the process and efficient management of restaurant reservations and availability reservations. An admin portal, which has access to the owner, facilitates checking of availability and bookings for customers. The services provided include table and menu bookings, all managed through the system, with the main objective of providing ordering and reservation services to customers.

The restaurant management system allows admin(only one admin) to easily edit, add, and delete reservation, track availability and manage client data.

The primary goal of this task is to enable restaurant owners to directly interact with clients. Additionally, the application allows clients to place requests to find free tables based on their specific seating requirements in their preferred area. This concept is designed to address the inconvenience people often face in finding desired restaurants and making bookings, which can be time-consuming. Managers also benefit from the application as it helps them digitally track bookings through the provided admin portal, instead of analyzing paper receipts.

Customers are required to register and become members in order to access the full features of the application. They can then check seat availability at specific timings and make bookings accordingly.

Customers also have the option to view booking details and cancel bookings if needed. In case of any confusion about restaurant locations, the application provides location maps and contact details for clearing queries via email or other contact information provided. This approach ensures a seamless and user-friendly experience for customers while providing efficient management tools for restaurant owners.

II. METHODOLOGY

The main objective of our project is to enhance the efficiency of ordering food while minimizing human error & providing quality services to restaurant customers. The application on the tablets should be capable of seamless communication with other devices.

The process begins with the customer or visitor opening the application and searching for food items from nearby locations. The customer can view the categorized menu card by scanning the QR code available on the table and order food items, either with or without logging in or registering. Selected food items are added to the cart which is available at hotel side for temporary storage.

If the customer decides to make a purchase, the application checks if his table is active or not, if the table is active, he can order or else the new table is created by the waiter. If not, the dine out service is also available at the hotel and he can directly pay to the billing counter.

In our project, we have created a database containing information about the listed daily needed food items, and costs. Once the login task is completed, the waiter can place an order from his tablet search the table number on his tablet and order the food. The waiter order is verified using the contact information provided during the login registration process. Upon successful verification, the admin checks if the order is confirmed by the waiter or not. If not, the process restarts. If the conditions are satisfied, the admin transmits the order data to the respective restaurant. Finally, the delivery boy delivers the order to the customer within the estimated time.

If there are any modifications required in the food menu, the admin can make changes in the database, and the updated menu will be reflected on the customer's Android device.

This ensures seamless communication and smooth processing of food orders, while maintaining accurate and up-to-date information on the menu items.

The use of tablet menus in restaurants has greatly transformed the dining experience for customers. Several existing programs have provided an app that enables restaurants to upload their menus onto iOS and Android-based tablets, making it easier for customers to browse through the menu using touch-screen gestures.

Our goal is to take this concept further by providing an advanced menu display that recommends dishes based on a recommendation algorithm, and instead of using an expensive iOS tablet, we opt for an Android-based tablet. Our system utilizes a cloud-based server to store the menu database, which not only makes it cost-effective but also secure. According to other developers who have created similar applications, customers seated at tables with tablets tend to spend approximately 10% more than those at tables without tablets.

This could be attributed to the fact that customers tend to buy more when they can place their orders instantly, without waiting for service. Our proposed system consists of several modules that work together seamlessly to provide a unique and innovative dining experience for customers.

A. Modules

- 1) *Module 1: Login Module:* The login module enables restaurants to access the application using their pre-registered login details. Each user is provided with a unique login ID and password, which they use to log in to the application. This ensures secure and authorized access to the application.
- 2) *Module 2: Registration Module:* In this module the Registration of New Staff Member is done. Every Staff member is given with his own login id and password. This module is available only for admin. The admin can add, edit, or remove the staff member.
- 3) *Module 3: Add/Update/remove Menu:* In this module the menu card of Restaurant is managed. It can be managed by Admin, Head chef, and chef. They can add new dish to menu card or remove dish or change the price of dish in menu card.
- 4) *Module 4: Add/Manage Table Module:* In Table module the tables are managed by admin, waiter, chef. They can change whether the table is active or inactive. If the table is active means the table is occupied, if the table is inactive means the table is unoccupied is can be used.
- 5) *Module 5: Add/Manage Category:* In category module the category of food item is managed. Such as whether the food item comes under Starter, Veg or non-Veg, Chinese, Dessert, Beverages, etc. all this category is managed by admin. Admin can add new category to menu.
- 6) *Module 6: Order Module:* In this module the waiter takes the order from customer and after confirming the order the order is sent to chef. The Chef can see the order from which table the order has come and he can make food according to received order. This module is managed by waiter, chef, and admin.
- 7) *Module 7: Billing Module:* In this module the automatic Bill is Generated and can be printed on counter.
- 8) *Module 8: Logout Module:* The logout module is used when the staff member completes the work hour. They can logout from the Application.

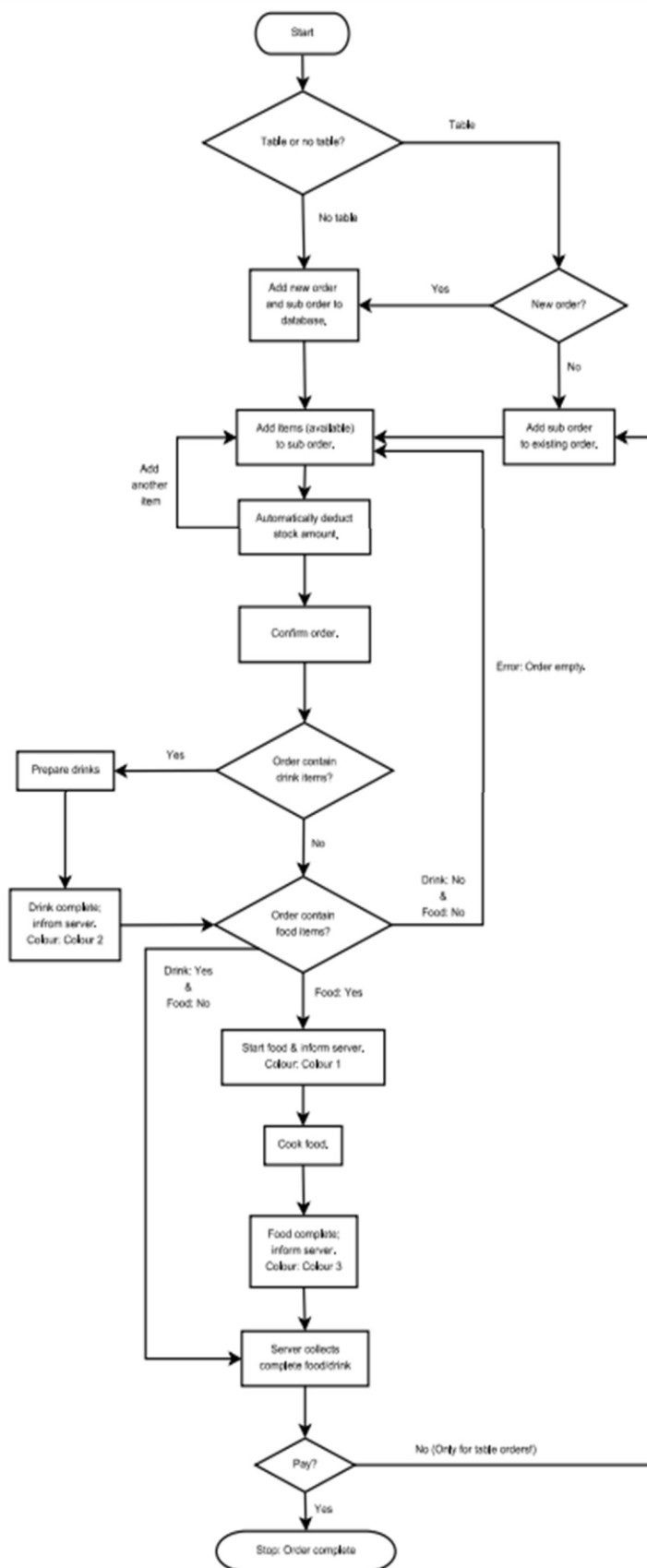
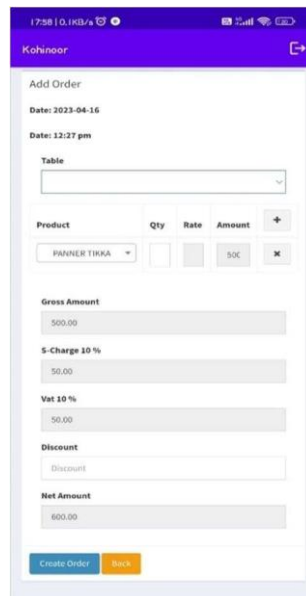
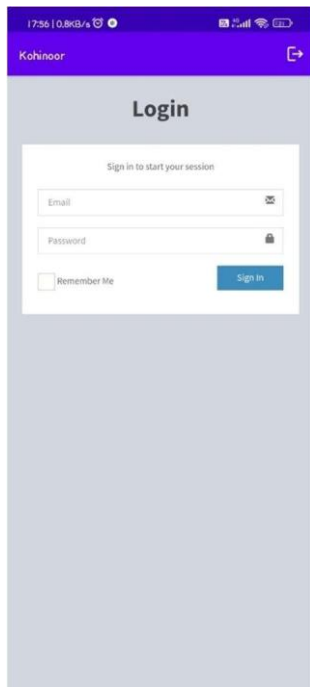
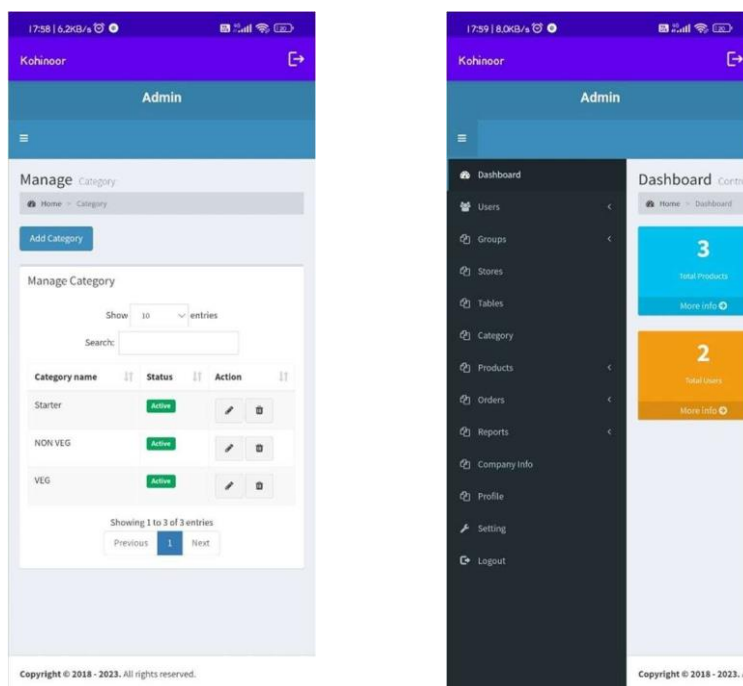


Figure : Flow chart to show the flow of events of an order.

III. RESULTS AND DISCUSSION

The overall design and development of the restaurant management system have been deemed satisfactory. The system is now capable of functioning as a Point of Sale (POS) solution for small and medium-sized enterprise (SME) restaurants. The various functions of the system are working correctly, and the overall design is easy to comprehend. The reports and invoices generated by the system are simple to understand and can be easily printed if required (refer to Appendix G to Q for the final results). Even individuals with limited knowledge of digital transformation in the restaurant industry can understand and utilize the restaurant management system for assisting with POS operations in an SME restaurant.





The restaurant management system provided is an excellent starting point for individuals with limited technology experience in a restaurant setting. While there may be a learning curve for employees accustomed to traditional methods, consistent use of this system for Front of House (FOH) operations in the SME restaurant will ultimately streamline daily operations. The system's reporting capabilities will also facilitate transaction tracking, making decision-making and sales forecasting more efficient. Overall, integrating this restaurant management system will greatly improve the restaurant's operations and enhance its business management capabilities.

IV. CONCLUSION

The findings of the project indicate that customers who are willing to dine at a restaurant often face the inconvenience of long waiting times when no tables are available. To address this issue, an application has been developed that allows customers to choose their desired table based on location and reserve it according to the number of members in their party. Additionally, the application provides pictures of the restaurant's interior, giving customers a glimpse of the ambiance. This innovative solution caters to the growing demand for efficient table booking and menu selection services, especially in today's fast-paced lifestyle where the use of android applications is becoming increasingly prevalent.

REFERENCES

- [1] A. Patil, R. Kalani, B. Patil, S. Shinde, Prof S.M.Shedole, IJTRA, Smart restaurant system using an-droid, 5, Issue 3 pp. 78-80 (2017)
- [2] M.P. Chorage, S.S. Bhande, S.S. Lale, M.A. Powar, K.V. Mane, IJARIII, Digital restaurant system, 7, Issue 1 pg. 756-760 (2021)
- [3] S. Dimbar, A. Kumbhakarna, P. Shend, IJEET, Survey of digital food ordering system based on android system for restaurant, 5, Issue 1 (2017)
- [4] N.M.Z. Hashim, N.A. Ali, A.S. Jaafar, N.R. Mo-hamad, L. Salahuddin, N.A. Ishak, IJCTT, Smart ordering system via bluetooth, 4, Issue 7 pg. 2253-2256 (2013)
- [5] Kavitha S., Shailaja K., Suma T.N., V. Shreenidhi, Asst Prof. Hari H., IJTRE, Paperless restaurant system, 6, Issue 9 pp. 5554-5557 (2019)
- [6] S. Sarkar, R. Shinde, P. Thakare, N. Dhomne, K. Bhakare, IJTRA, Integration of touch technology in restaurants using android, 5, Issue 3 pg. 721-728 (2017)
- [7] S. Khedikar, S. Baghele, V. Muley, P. Shah, A. Bagade, P. Wagh, S. Pusdekar, IRE, Digital food ordering system for restaurants, 3, Issue 10 pp. 182-185 (2020)



ISSN No. : 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214
Website : www.ijraset.com, E-mail : ijraset@gmail.com

Certificate

It is here by certified that the paper ID : IJRASET51116, entitled

Restaurant Management System
by
Pratik Lanke

after review is found suitable and has been published in
Volume 11, Issue IV, April 2023

in

International Journal for Research in Applied Science & Engineering Technology
(International Peer Reviewed and Refereed Journal)
Good luck for your future endeavors

Py

Editor in Chief, IJRASET



ISRA Journal Impact Factor: 7.429



45.98 INDEX COPERNICUS



THOMSON REUTERS Researcher ID: N-9681-2016



TOGETHER WE REACH THE GOAL SJIF 7.429



ISSN No. : 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214
Website : www.ijraset.com, E-mail : ijraset@gmail.com

Certificate

It is here by certified that the paper ID : IJRASET51116, entitled

Restaurant Management System
by
Pratik Harna

after review is found suitable and has been published in
Volume 11, Issue IV, April 2023

in

International Journal for Research in Applied Science & Engineering Technology
(International Peer Reviewed and Refereed Journal)
Good luck for your future endeavors

Py

Editor in Chief, IJRASET



ISRA Journal Impact Factor: 7.429



45.98 INDEX COPERNICUS



THOMSON REUTERS Researcher ID: N-9681-2016



TOGETHER WE REACH THE GOAL SJIF 7.429



ISSN No. : 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

Certificate

It is here by certified that the paper ID : IJRASET51116, entitled

Restaurant Management System
by
Adesh Adhao

after review is found suitable and has been published in
Volume 11, Issue IV, April 2023

in
International Journal for Research in Applied Science &
Engineering Technology
(International Peer Reviewed and Refereed Journal)
Good luck for your future endeavors

By [Signature]

Editor in Chief, IJRASET



ISRA Journal Impact
Factor: 7.429



45.98
INDEX COPERNICUS



THOMSON REUTERS
Researcher ID: N-6685-2018



TOGETHER WE REACH THE GOAL
SJIF 7.429



ISSN No. : 2321-9653

IJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

Certificate

It is here by certified that the paper ID : IJRASET51116, entitled

Restaurant Management System
by
Anuj Sapkal

after review is found suitable and has been published in
Volume 11, Issue IV, April 2023

in
International Journal for Research in Applied Science &
Engineering Technology
(International Peer Reviewed and Refereed Journal)
Good luck for your future endeavors

By [Signature]

Editor in Chief, IJRASET



ISRA Journal Impact
Factor: 7.429



45.98
INDEX COPERNICUS



THOMSON REUTERS
Researcher ID: N-6685-2018



TOGETHER WE REACH THE GOAL
SJIF 7.429

LITERATURE REVIEW ON RESTAURANT MANAGEMENT SYSTEM

M. Faizan Khandwani*¹, Pratik Lanke*², Pratik Harne*³, Anuj Sapkal*⁴,
Adesh Adhao*⁵

*¹Assistant Professor, Dept. Of IT, Shri Sant Gajanan Maharaj College Of Engineering, Shegaon,
Maharashtra, India.

*^{2,3,4,5}Student, Dept. Of IT, Shri Sant Gajanan Maharaj College Of Engineering, Shegaon,
Maharashtra,India.

DOI : <https://www.doi.org/10.56726/IRJMETS37920>

ABSTRACT

In our system offers an online food ordering platform that provides convenience for customers and addresses the limitations of traditional queueing systems. It offers a hassle-free way for customers to order food from restaurants and mess services. The system enhances the process of taking customer orders by providing an online food menu, which allows customers to easily select their desired items and track their orders. Additionally, customers can provide feedback on their food items, which helps to inform hotel staff of areas for improvement in quality. The system can also recommend hotels and food based on user ratings. Payment options include online payment or pay-on-delivery, with separate accounts for each user, which ensures secure ordering through individual IDs and passwords.

Keywords: Food Ordering System, Dynamic Database Management, Smart Phone.

I. INTRODUCTION

The restaurant industry has been greatly transformed by technological advancements, particularly in point-of-sale operations. The use of e-menus has revolutionized the way customers interact with menus, providing them with more detailed information about menu items and drinks compared to traditional paper menus. This has improved the ordering process by making it simpler and more accessible.

The benefits of e-menus go beyond just improving the ordering process. Restaurants can now build their e-reputation and engage with customers in real-time. With digital tablets and touch screen technology, menus have evolved from simple chalkboards and printouts to detailed and colorful displays. Orders can be taken accurately the first time, reducing the need for servers to run back and forth to a distant terminal.

E-menus also enable orders to be associated with individual seats at a table, ensuring greater accuracy and reducing errors. Modifications to orders can easily be made, and the cost can be calculated in real-time. Recommendation algorithms can even suggest dishes based on previous orders, making it easier for customers to build their orders and view the most popular dishes. Various filters, such as price, taste, and quantity, can be used according to individual preferences.

In summary, e-menus have transformed the restaurant industry by providing customers with more detailed information and simplifying the ordering process. They have also enabled restaurants to engage with customers in real-time and build their e-reputation.

II. LITERATURE REVIEW

[1] A snack organizing scheme has existed grown to organize the process of identifying orders in miscellaneous outlets. The system is devised to capably control consumer orders by providing the alternative to order pre-set parts or custom-make orders accompanying just individual click. This whole has existed executed as an Android use for Tablet PCs. The program that controls display was founded utilizing JAVA and Android, while the backend promotes a MySQL table.

[2] The system adopts that consumers will be utilizing smartphones to place their orders. Once the consumer lands at the outlet, they can prove their preserved order completely by affecting their smartphone screen. The list of pre-picked articles will open or fan out on the room for cooking food screen, and upon ratification, an order slip will be impressed for further deal with. This answer facilitates the pre-order process for clients, making it a more nearby alternative.

[3] A mathematical eating arrangement for inns was planned and achieved, promoting Android electronics. This structure is stimulated by a vital table serviceableness that retrieves all inevitable facts from a concentrated table. The handy request has experienced to raised adeptness and veracity in joint movements, while still lowering human wrong. This structure has overcome former imperfections of mechanized fare arranging structures, and demands only of highest quality-period expense in device. The use of netting duties electronics to mix inn administration plans is expressed in [4].

By means of the Digital Hotel Management arrangement, miscellaneous inn spreadsheet methods in the way that the Ordering System Kitchen Order Ticket (KOT), the Billing System, and the Customer Relationship Management plan (CRM) maybe connected together. This resolution is fit friendly the adding or growth of lodging operating system structures, making it appropriate for lodging chains of some amount. The objective of the research bestowed in [5] search out design a Wi-Fi drink organizing whole for use in dining rooms. The Wireless Ordering System (WOS) was devised and grown, accompanying a devote effort to something allure mechanics movements, plans design, functions, restraints, and pieces of advice. The authors plan that as handheld ploys in the way that PDAs enhance more usually secondhand in inns, extensive uses like WOS will enhance more and more main finishes for reconstructing administration effectiveness, underrating human mistakes, and improving the characteristic of department dealing with customers.

In [6], a Wi-Fi cooking organizing scheme was devised and achieved, as well a consumer response arrangement, for use in saloons. This order admits outlet partner to surely start bureaucracy in a Wi-Fi atmosphere and restore card performances as wanted. The system more combines smartphones, that have happened joined into a customizable fare arranging plan, and involves honest-opportunity client response use to allow ideas 'tween dining room landowner and clients in actual-period. The objective of the research bestowed in [7] search out analyze the determinants that influence the stances of Turkish academy pupils the one use the computer network toward connected to the internet meat commanding. To carry out this, the authors handled the Technology Acceptance Model (TAM) initially grown by Davis in 1986 to study the maintenance of netting-located atmospheres for foodstuff arranging. In addition to TAM, the authors additional trust, novelty, and extrinsic influences as main determinants to the model. In [8], the research aims to mechanize the meat authorizing process in outlets and reinforce the overall eating knowledge for consumers. The authors argue the design and exercise of a meal commanding whole for inns, that appropriates Wi-Fi dossier approach to servers. The system contains an Android use that holds all cuisine analyses, admitting consumers to place orders wirelessly utilizing their movable tools. The room for cooking food and cashier accept these order analyses, that are therefore refurbished in the principal table. Additionally, the inn partner can surely accomplish card modifications utilizing this whole.

[9] focuses on the works fashioned by cafeteria holder to select news and ideas sciences to a degree PDAs, Wi-Fi LANs, and damaging multi-touch screens to enhance the eating knowledge. The paper more explains the restraints of unoriginal paper- located and PDA-located foodstuff organizing orders and suggests a cheap touch screen-located Restaurant Management System that handles an Android smartphone or capsule as a answer.

III. EXISTING SYSTEM

The current system used in many restaurants relies on paper-based menus and order-taking methods. While this may be traditional, it has numerous disadvantages. For instance, papers can easily get damaged by stains, lost due to accidents or fires, or misplaced in general, leading to wasted time and money. Furthermore, making changes to the menu requires reprinting the entire menu card, resulting in unnecessary wastage. It is also difficult to access a specific record from the stack of papers, making the system time-consuming.

Customers have to physically visit restaurants to learn about food items and place their orders, which can be inconvenient and time-consuming. Ordering over the phone is also problematic, as customers lack a physical copy of the menu item and cannot confirm their order visually. Additionally, restaurants must hire employees to take orders and process payments, which can be costly and challenging in today's market, where labor rates are continually increasing.

Moreover, the traditional paper-based system can result in errors, such as misinterpreting a customer's order and serving the wrong dish. Customers may also have to call their waiter multiple times to place their order, which can be frustrating and inefficient. Overall, the paper-based system used in many restaurants is outdated

and inefficient and can lead to numerous problems for both customers and businesses.

ADVANTAGES

1. Avoiding waiting time of staff members.
2. Provide further information on electronic menus.
3. Customer and waiter will not come under pressure while giving order and taking order.
4. There will be benefit of admin while taking e-payment and cash payment.
5. Easy way to manage Restaurant.
6. No tension of calculating the total order in calculator i.e. all the total bill will be automatically calculated.
7. It will remove the work force from the admin.
8. Billing errors will be avoided means the bill will be auto generated.
9. Ordering error will be avoided all the order will be shown to the admin panel and chef for preparing order.
10. Menu card will be changed by the admin but the QR code will be same.

FEATURES

1. Fast wireless ordering operation with improved customer service.
2. Secured and Accurate wireless transfer of Order.
3. Easy tracking on Running and cancelled Orders.
4. The billing is done simultaneously at the time of order taking.
5. No manipulations in billing and secured transactions.
6. Increase in table turnover, sales and profitability.
7. Smooth flow of communication between service staff and kitchen.
8. Optimum staff utilization reduces the labor cost.

IV. CONCLUSION

As time passes and smartphones become increasingly popular, their impact on various aspects of human life becomes more evident. Therefore, people tend to use technology in different situations more frequently, which presents an opportunity for restaurant owners to take advantage of this trend by providing advanced technology. By using QR codes in the ordering process, physical ordering and payment can be eliminated, and customers and restaurant managers can enjoy more benefits. The goal is not only to replace traditional ordering, but also to offer this smart ordering alongside the traditional method to attract customers and improve their loyalty. Additionally, this method can reduce staff members' time and energy, ultimately leading to greater customer satisfaction.

Table 1: Comparison of Existing POS System Solution

POS System	Table Service	Stock Control	General Reporting	Advanced Discounts	Kitchen Display	PDA Order Input
Sucess	Yes	No	Yes	No	No	No
RPS(HOSPOS)	No	Yes	Yes	No	Yes	No
Abacre POS	Yes	No	Yes	Yes	Yes	No
eZee Burrp	Yes	Yes	Yes	No	Yes	No

V. REFERENCE

[1] Kirti Bhandge, Tejas Shinde, Dheeraj Ingale, Neeraj Solanki, Reshma Totare, " A Proposed System for Touchpad Based Food Ordering System Using Android Application", International Journal of Advanced Research in Computer Science Technology (IJARCST 2015).

[2] Varsha Chavan, Priya Jadhav, Snehal Korade, Priyanka Teli, "Implementing Customizable Online Food Ordering System Using Web Based Application", International Journal of Innovative Science, Engineering Technology(IJISSET) 2015.

-
- [3] Resham Shinde, Priyanka Thakare, Neha Dhomne, Sushmita Sarkar, "Design and Implementation of Digital dining in Restaurants using Android", International Journal of Advance Research in Computer Science and Management Studies 2014.
- [4] Ashutosh Bhargave, Niranjana Jadhav, Apurva Joshi, Prachi Oke, S. R Lahane, "Digital Ordering System for Restaurant Using Android", International Journal of Scientific and Research Publications 2013.
Khairunnisa K., Ayob J., Mohd. Helmy A. Wahab, M. Erdi Ayob, M. Izwan Ayob, M. Afif Ayob, "The Application of Wireless Food Ordering System" MASAUM Journal of Computing 2009.
- [5] Noor Azah Samsudin, Shamsul Kamal Ahmad Khalid, Mohd Fikry Akmal Mohd Kohar, Zulkifli Senin, Mohd Nor Ikhazan, " A customizable wireless food ordering system with real time customer feedback", IEEE Symposium on Wireless Technology and Applications(ISWTA) 2011.
- [6] Serhat Murat Alagoza, Haluk Hekimoglu, " A study on tam: analysis of customer attitudes in online food ordering system", Elsevier Ltd. 2012.
- [7] Patel Krishna, Patel Palak, Raj Nirali, Patel Lalit, " Automated Food Ordering System" International Journal of Engineering Research and Development (IJERD) 2015.
- [8] Mayur D. Jakhete, Piyush C. Mankar, " Implementation of Smart Restaurant with e-menu Card," International Journal of Computer Applications 2015 of Smart Restaurant with e-menu Card, "International Journal of Computer Applications 2015.



***International Research Journal Of Modernization
in Engineering Technology and Science***

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 05/50500012823

Date: 06/05/2023

Certificate of Publication

This is to certify that author "Pratik Lanke" with paper ID "IRJMETS50500012823" has published a paper entitled "LITERATURE REVIEW ON RESTAURANT MANAGEMENT SYSTEM" in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 05, May 2023

A. Desai

Editor in Chief



We Wish For Your Better Future
www.irjmets.com





***International Research Journal Of Modernization
in Engineering Technology and Science***

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 05/50500012823

Date: 06/05/2023

Certificate of Publication

This is to certify that author "Pratik Harne" with paper ID "IRJMETS50500012823" has published a paper entitled "LITERATURE REVIEW ON RESTAURANT MANAGEMENT SYSTEM" in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 05, May 2023

A. Desai

Editor in Chief



We Wish For Your Better Future
www.irjmets.com





***International Research Journal Of Modernization
in Engineering Technology and Science***

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 05/50500012823

Date: 06/05/2023

Certificate of Publication

This is to certify that author "Anuj Sapkal" with paper ID "IRJMETS50500012823" has published a paper entitled "LITERATURE REVIEW ON RESTAURANT MANAGEMENT SYSTEM" in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 05, May 2023

A. Desai

Editor in Chief



We Wish For Your Better Future
www.irjmets.com





***International Research Journal Of Modernization
in Engineering Technology and Science***

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 05/50500012823

Date: 06/05/2023

Certificate of Publication

This is to certify that author "Adesh Adhao" with paper ID "IRJMETS50500012823" has published a paper entitled "LITERATURE REVIEW ON RESTAURANT MANAGEMENT SYSTEM" in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 05, May 2023

A. Desai

Editor in Chief



We Wish For Your Better Future

www.irjmets.com





KOHINOOR FAMILY RESTAURANT AND BAR

UTTAMNAGAR, SHIVANE, PUNE – 411023

M.- 9822251857

To Whom it May Concern

Student Name:

1. Adesh Vijay Adhao
2. Anuj Ravindra Sapkal
3. Pratik Purushottam Lanke
4. Pratik Sanjay Harne

Of SSGMCE, IT Department

KOHINOOR RESTAURANT AND BAR agrees to pay all the expenses for the above-named student of SSGMCE. The sponsorship includes, but is not limited to, development of project/software maintenance of software/services

Project Name: - Android Application for Restaurant Management System.

This sponsorship will cover the student beginning on August 2022 to Jan 2023.

Sponsorship Amount - 9000/- Rs.

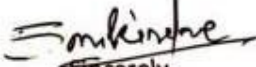
Contact Person – Shantanu Kindre

Contact Number - 9822251857

Address - Kohinoor Restaurant and Ba, Uttamnagar, Shivane, Pune

Pin code - 411042

Country - India


Sincerely,

Name and Sign of Contact Person
Kohinoor Rosto Bar

Proprietor
Shantanu Kindre



KOHINOOR FAMILY RESTAURANT AND BAR

UTTAMNAGAR, SHIVANE, PUNE – 411023

M.- 9822251857

Date: 20/04/2023

To,
The Principal
SSGMCE, Shegaon

Subject: Completion of project to your students

This is to certify that the below mentioned final year students of IT department of your college SSGMCE, Shegaon, have successfully completed the development of a Software Project name as "Restaurant Management System" for Kohinoor Restaurant and Bar, Uttamnagar, Shivane, Pune under the guidance of Prof. F.I.Khandwani.

Students Names:

1. Adesh Adhao
2. Anuj Sapkal
3. Pratik Lanke
4. Pratik Harne

We state on record that, these students have worked on the development of this project from August 2022 to April 2023. We found that the student have done satisfactory work.

We also appreciate the continuous mentoring by Prof. F.I.Khandwani. For his continuous mentoring and valuable suggestions to the students during development of project.

THANKING YOU,

Kohinoor Family Bar


Shantanu Kindre Proprietor

SGM/FRM/OFF-57



Shri Gajanan Shikshan Sanstha's
SHRI SANT GAJANAN MAHARAJ
COLLEGE OF ENGINEERING
SHEGAON - 444 203 Dist. Buldhana (M.S.) INDIA

RECEIPT

No.: 2023/100000805
Date : 20.04.2023

Received with thanks from Shri/Ku. **KOHINOOR**
RESTAURANT & BAR OF SHANTANU KINDRE
vide DD/Cheque/Cash **CONSULTANCY**
SPONSERSHIP
of Year & Branch **IT DEPT.**

The sum of **Rs. NINE THOUSAND and Paise ZERO**
only.




On account of **Consultancy Received.**

Rs. 9,000.00

DRAFT
Subject to realisation
Subject to shegaon jurisdiction

RECEIVER / CASHIER
S.S.G.M.C.E, Shegaon

Project Group Details

	<p>Name: Pratik Purushottam Lanke</p> <p>Address: Kumbhar Wada, Near Vinay Lodge, Shegaon 444203</p> <p>Mobile No: 8208284241</p> <p>Email ID: lankeprateek@gmail.com</p>
	<p>Name: Pratik Sanjay Harne</p> <p>Address: Sr.No 14/3 Shri Krishna Nagri D-402 NDA Gate Kondhwe Dhawade Pune 411023</p> <p>Mobile No: 9307240342</p> <p>Email ID: pratikharne23@gmail.com</p>
	<p>Name: Anuj Ravindra Sapkal</p> <p>Address: Near Rahul tower ward no 5,Nandura 443404</p> <p>Mobile No: 7387304115</p> <p>Email ID: anujsapkal7432@gmail.com</p>
	<p>Name: Adesh Vijay Adhao</p> <p>Address: Om nagar ,near new bus stand ,malkapur road ,Nandura-443404</p> <p>Mobile No: 9503249020</p> <p>Email ID: adhaoa7@gmail.com</p>