ASHESI UNIVERSITY

RESAURANT MANAGEMENT SYSTEM FOR KAS VALLEY RESTAURANT

APPLIED PROJECT

B.Sc. Computer Science

Christabel Kekeli Alormene

2022

ASHESI UNIVERSITY

RESTAURANT MANAGEMENT SYSTEM FOR KAS VALLEY RESTAURANT

APPLIED PROJECT

Applied Project submitted to the Department of Computer Science, Ashesi University in partial fulfilment of the requirements for the award of

Bachelor of Science degree in Computer Science.

Christabel Kekeli Alormene

2022

DECLARATION

I hereby declare that this Applied Project is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

………………………………………………………………………………………………

Candidate's Name: Christabel Kekeli Alormene

………………………………………………………………………………………………

Date: 13th May 2022

………………………………………………………………………………………………

I hereby declare that preparation and presentation of this Applied Project were supervised in accordance with the guidelines on supervision of Applied Project laid down by Ashesi University.

Supervisor's Signature:

………………………………………………………………………………………………

Supervisor's Name:

Joseph Agyapong Mensah

………………………………………………………………………………………………

Date:

25/04/2022

………………………………………………………………………………………………

Acknowledgements

I thank the almighty God for granting me strength and grace throughout my journey at Ashesi. To my dear parents, Mr. Frank Alormene and Mrs. Ivy Alormene, and my wonderful siblings, Eyram Alormene and Elorm Alormene, I am grateful for the unconditional love and support. To my family and friends, thank you for all the prayers and encouragement. To these amazing people: Vanessa Amenuglo, Jennifer Aggrey, Fuseini Baba Hussein, Fifii Bentil, Kofi Bandoh and Afsanat Ineza, thank you for everything. Special thanks to Mr. Ransford Dafour for taking time off his busy schedule to support me in this project. And lastly, I would like to thank my supervisor, Mr. Joseph Mensah, for all the guidance, feedback, and academic advice he gave me to make this capstone project successful. Thank you, and God richly bless you all.

Abstract

In response to the recent advancement in technology and its tremendous impact, many businesses have integrated technological tools into their business operations. The food industry is no exception. In the global food industry, technology has made it possible to improve food services, increase customer satisfaction and maximize business returns. There have been changes such as the shift from conventional paper ordering at restaurants to online pre-ordering of meals through the internet. Developments in restaurant software and POS technologies have also aided restaurants in providing better services to their customers. However, the Ghanaian food industry has not taken full advantage of the emerging technologies to optimize its operations. The few restaurants that have adopted restaurant software have their applications limited to POS transactions or strictly online food ordering systems. The applications do not include some core aspects of restaurant operations, such as table reservation management, customer data management, and e-commerce integration. The lack of a centralized system to handle all these related aspects of the restaurant's operations leads to data inconsistency and redundancy issues. It also makes it difficult for management to analyze their business data because it is not stored in one location but scattered across different units. All these affect the restaurant's efficiency and profitability levels in the long run.

In an attempt to enhance the efficiency of restaurant businesses in Ghana, this paper proposes a web-based restaurant management system for Kas Valley Restaurant, located in the Oyibi township of Accra, Ghana. The system will enable the restaurant to receive orders and payments online, process in-person orders, manage table reservations and customer data, and view analytics on their sales. All these functionalities on a single application would ensure data integrity, minimize data redundancy, foster industry innovation, and significantly improve the restaurant's overall efficiency. Additionally,

using such software in a restaurant contributes to the Sustainable Development Goal (SDG) 9: “Build resilient infrastructure, promote sustainable industrialization and foster innovation”. In this regard, the restaurant management system will enhance technological innovation in the restaurant industry, hence increasing productivity levels.

Table of Content

List of Tables

List of Figures

List of Abbreviations

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
CSS	Cascading Style Sheet
COVID-19	Coronavirus 2019
ER Diagram	Entity Relationship Diagram
FR	Functional Requirement
GDP	Gross Domestic Product
GHC	Ghana Cedis
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
NFC	Near Field Communication
NFR	Non-Functional Requirement
PC	Personal Computers
PDA	Personal Digital Assistants
PMR	Personalized Mobile Restaurant
POS	Point-Of-Sale
QR	Quick Response
RFID	Radio Frequency Identification
RMS	Restaurant Management System
SDG	Sustainable Development Goal
SMTP	Simple Mail Transfer Protocol
SMS	Short Message Service
UML	Unified Modelling Language

Chapter 1: Introduction

1.1 Background

Technology has become a central element in most business operations in today's fast-paced economy. Various industries, including the food industry, utilize innovative technological systems and tools to speed up processes and maximize customer satisfaction. The digitalization of food services has allowed many businesses within this industry to own online ordering services via websites and apps. During COVID-19, the need for online food systems became more evident due to the temporary shutdown of restaurants and the lack of in-person customers. Several restaurants went out of business because they did not have alternative systems to support their business operations. Most restaurants are faced with the problem of unreliable and inefficient manual ordering systems and management systems, thus affecting their customer services, efficiency and profitability. The traditional food ordering and management systems that involve hand-written notes, manual filing systems and reminders are not very efficient. These systems are time-wasting and prone to human errors such as incorrect data entry and inconsistency in information records.

Additionally, most restaurant software are limited to POS functionalities and may not include other aspects of the restaurant's operations such as table reservation management, customer data management and e-commerce integration. Kas Valley Restaurant requires a centralized management system to promote the restaurant's efficiency and ensure data consistency in their records. They need a single application that will enable them to process online and in-person orders, accept online payments, manage table reservations and customer data, and view analytics on their sales. The development of such a system would contribute to the Sustainable Development Goal 9; "Build resilient infrastructure, promote sustainable industrialization and foster innovation".

1.2 Problem Statement

In the last couple of years, Kas Valley Restaurant has been facing challenges with efficient management of its business operations. As industries are fast expanding, and given the rampant growth of technology, people prefer to make transactions in ways that provide more ease and speed. Kas Valley currently receives all orders via walk-ins, phone calls, or social media accounts. However, the issue with this is that with their growing customer base and increasing orders per day, there are sometimes very long queues from walk-in orders, which has a dire effect on customer service. Ordering via phone call can sometimes be frustrating for both customers and the restaurant, as the process can be affected by a bad network reception. Also, customers sometimes have to wait when the restaurant's call line is busy as the restaurant may be receiving orders from other customers simultaneously.

Also, Kas Valley's record-keeping system is poor as they do not have a centralized platform for maintaining their records. This poses challenges when generating analytics on their sales and orders as they cannot effectively track these data records. Additionally, the business has difficulty keeping track of table reservations and quickly determining which days and times have been fully booked. They manually have to check their records to perform this task which is time-consuming and quite tedious. In one way or the other, all these management issues have slowed down the restaurant's operations and led to decreased customer satisfaction.

1.3 Aim

The project aims to develop a web-based restaurant management system that facilitates online and in-person meal ordering in Kas Valley restaurant. The application will act as a centralized system for maintaining their records, including customer data. The system also aims to support table reservation management and supply actionable business insights through data analytics on the business's sales.

1.4 Project Objectives

At the end of this project, the proposed solution will be able to:

1. Enable customers to place orders via the restaurant’s website
2. Enable the restaurant to receive and process both in-person orders and online orders
3. Provide a centralized system for keeping and maintaining data records such as customer data
4. Generate analytics on sales, orders and customer segmentation

1.5 Literature Review

The restaurant industry plays a significant role in global economic development and the sustainability of other industries worldwide. It provides food services where customers can order and eat meals. As a labour-intensive industry, it gives individuals numerous job opportunities, thus cutting down the unemployment rate [1]. The restaurant industry constitutes a large part of the hospitality sector, and their production of commercial meals contributes immensely to the tourism sector [2]. By offering local cuisines and rich food experiences to tourists, the restaurant industry is one of the main drivers of food tourism.

Additionally, the restaurant industry is one of the highest contributors to a nation’s Gross Domestic Product (GDP). In 2020, Ghana’s hotel and restaurant industry contributed around 3.9 billion Ghanaian cedis (GHC), approximately 640.9 million dollars, to the country’s GDP [3]. Nevertheless, this amount showed a significant decrease compared to the figures from the previous years. This decline is possibly due to the devastating impacts of the COVID-19 pandemic on the country [3]. Since 2014, there has been an increasing rate in the contributions of the restaurant and hotel industry to Ghana's GDP, with a peak of nearly six billion GHC recorded in 2019, which is about 985.8 million dollars. However, the emergence of the COVID-19 pandemic brought about a massive blow to the restaurant industry. Their services were affected, resulting in a decline in their contributions to the country's GDP.

The pandemic seriously affected the global restaurant market due to the temporary closure of food outlets following the lockdown measures by government organizations. As a result of the social distancing measures, movement restrictions and general caution towards public places, dine-in customers reduced drastically. "The year-over-year change of seated diners in restaurants worldwide, compared to 2019, was 30 percent on February 20, 2022" [4]. Many restaurants have shut down permanently or long-term from the period COVID-19 set in. For instance, according to the National Restaurant Association (USA), 17% of American restaurants, which is more than 110000 establishments, have closed permanently or long-term since the onset of the pandemic [5]. However, the adoption of digital ordering channels and restaurant management software helped a few restaurants survive the pandemic [6]. Technology has positively impacted the restaurant industry hence the growing demand for restaurant management software and POS systems [7].

In this era of modernization, technological innovations have empowered many businesses, and the restaurant industry is not an exception. Technology in the restaurant industry is not only transforming the traditional aspects of full-service dine-in for customers but promoting the revenue growth of restaurants. The digitization of food services has thus aided many restaurants in enhancing operational efficiency and meeting changing consumer demands in this increasingly digital age [8]. Some of the notable trends in technology utilization at restaurants include online and mobile ordering, contactless payments, online table reservation booking, Q.R. codes for payments, and POS systems, to mention a few. A restaurant management system is a collective term for software and applications that aid in streamlining food business operations to boost product demand in the restaurant. They come in different forms and provide varying functionalities depending on the needs and requirements that the restaurant needs the application to fulfil.

RMS is now a substitute for traditional restaurant management. The latter is not robust enough and comes with several limitations, such as its heavy reliance on human effort since everything is done manually. RMS has been developed to minimize human-based errors and increase efficiency and has become the backbone of most restaurants. RMS emerged in the 1970s and has evolved to facilitate restaurant operations and management processes [9]. In terms of advantages, it improves accurate data flow and increases profitability and customer satisfaction in the long run. In a research study to determine the operational effects of using RMSs based on the perspectives of restaurant staff, the findings showed that the use of an RMS facilitated the operations management, increased sales and provided service efficiency [10].

Thus, in today's digitally driven world, restaurants need to adopt and integrate these technological innovations into their business processes to have a competitive advantage and keep up with recent trends.

1.6 Related Works

In recent years, several developments in restaurant technologies have sprung up because of their increasing demand and the tremendous support they provide in the restaurant industry [11]. For clarity, the related works would be divided into these categories: Personal Digital Assistant (PDA) restaurant systems, POS systems, Radio Frequency Identification (RFID) based restaurant applications, and online ordering and management systems(mobile and web-based).

1.6.1 PDA-Based Restaurant Applications

Some restaurants have adopted wireless ordering systems such as Personal Digital Assistants (PDA) to replace the traditional way of taking orders with hand-written notes. PDAs are portable hand-held computing devices that can store, retrieve and exchange information with other devices. They are used in restaurants to coordinate food ordering which increases efficiency by saving time, improving customer service and reducing human errors [12]. In a research paper, a PDA-based food ordering system was developed using a client-server architecture and wireless connectivity between the PDA client and a web server [13]. In the proposed system, orders are taken by ticking the menu on the PDA. These orders are directly transferred to the server in the kitchen, thus eliminating the task of taking orders manually and waiting for orders to be transferred to the kitchen by the staff.

In another paper, the authors integrated wireless communication and web-services technologies to implement a wireless food ordering system [14]. The proposed system functions through desktop P.C.s and PDAs, either over a wired or wireless integrated local area network. Similarly, data access to the servers was programmed to be wired or wireless. The PDAs displayed the menus in this proposed system and enabled customers to place orders. All orders placed through the PDA are reflected on the restaurant’s P.C. as retrieved from the server.

Though PDAs simplify the order placement process by providing a better alternative to the conventional food ordering process, this approach to restaurant management comes with several limitations. The PDA system can increase the restaurant’s expenditures because this system requires a large number of PDAs to serve every customer, especially during periods where order placement is high [15]. Moreover, in such situations, constantly recharging PDA batteries would be a drawback for businesses [12]. Also, the user interface of the PDA system is not user-friendly as it dominantly consists of textual information [15]. PDAs can also spread infectious diseases as these devices are to be shared with public customers. Finally, it does not support realtime feedback between the customer and the restaurant [16].

1.6.2. POS-Based RMS

A POS application is a point-of-sale system that processes transactions made at the business premises. This includes transactions such as order processing, billing and payment. In a research paper, a restaurant POS system was developed to improve customers' dining experiences [17]. The proposed solution used a chip and pin intelligent card system to prevent credit card fraud on the POS. However, the limitation of this system is that, during peak hours where there are a lot of orders coming in, there are long queues, which increases the customer's waiting time leading to decreased customer satisfaction. A paper investigating the effect of waiting times on restaurant service delivery in the Ho municipality of Ghana showed that customers were highly dissatisfied when they waited for relatively longer periods to place and receive their orders [18].

1.6.3 Radio Frequency Identification (RFIDs) Based RMSs

RFID is a technology that uses wireless communication and radio frequencies to identify tagged items. In restaurant management, RFID is used to identify customers and retrieve their personal data in realtime. It is also widely used in the area of inventory tracking. In a research paper, an RFID-based RMS was designed using open-source technologies, including python and Raspberry Pi [19]. The proposed system enables customers to select and scan their desired food items. Also, the identification of food items and payment was done via RFID. Each food item is marked using RFID tags in the system, and the total amount of the customer's order is deducted automatically from their digital wallet. Another research paper developed an RFID-based RMS called "Fast-Waiter" to improve order taking and dispatch [20]. The proposed system integrates an RFID module using Arduino, which enables direct payment of a customer's order using an RFID card. Though RFID-based systems are helpful in order-taking, they do not support ordering from outside of the restaurant; that is, you are required to be physically present at the restaurant to place an order.

1.6.4 Online Ordering and Management Systems

Online food ordering systems were developed to overcome some of the limitations in the earlier systems discussed. This system allows customers to place orders via the internet. Customers can place orders online from wherever they are, complete payment and pick up their meals or have them delivered. The online food ordering system allows restaurants to accept and manage orders using a website or mobile application running on the internet. This approach to restaurant management offers customers convenience and enables the restaurant to save on labour costs.

1.6.4.1 Mobile Application Based Approach

Liyanage [21] designed a smart restaurant management and ordering system called "Foody". Foody is a mobile application that allows the customer to choose the menu according to their taste in a minimum time. Some of the components handled in Foody include table reservations and order handling, modelling a 3D menu, sentiment analysis and summarizing recommendation management system and user rating system.

In another proposed system, a mobile restaurant app was implemented using wireless communication, specifically, Near Field Communication (NFC) technology [22].

In this system, the customer can browse the online menu on a mobile phone and access information on the meals. Additionally, the mobile app integrates member bonuses, coupons and meal discounts for loyal customers. A number call service is integrated into the app using data transmission via NFC.

Another research paper developed a personalized mobile restaurant (PMR) system to cater for the customer's personal special needs, including allergies, health conditions, diet, culture, religion, and dislikes [23]. The PMR mobile application communicates with the PMR server over a Wi-Fi network in this proposed system. This system primarily aimed to close the gap between customer needs and satisfaction.

One major setback of mobile-based restaurant applications is that they can only be used on smartphones. Thus, ordering food via a computer will not be possible in this system.

1.6.4.2 Web-based Approach

In a research paper, Alfiya [24] proposed a food ordering management system to enable customers order food by registering on the web application and selecting the food items from an e-menu. The proposed system helps the administrator to have a clear idea as to when and which food items are preferred more on a day-to-day basis. However, one limitation of their solution is that it lacks an online payment option for customers to pay directly via the system.

In another study, a web-based ordering system was proposed to enable customers place orders for food and beverages in a restaurant [25]. This proposed system comes with a Q.R. functionality which would be easy to use mainly for mobile users. Dine-in customers can place their orders by scanning the Q.R. code. The purpose of this system was to improve the service quality at restaurants in terms of fast ordering and payment. However, one limitation of this system is that it only supports in-person meal ordering. Customers who want to order in the convenience of their homes or location would not be able to do so.

In general, these web-based systems place more focus on the placement and processing of orders with less emphasis on other core areas of the restaurant, such as table reservation management and customer data management. Additionally, these systems have no data visualization techniques to enable the restaurant to obtain insights about their business and make informed decisions.

1.7 Proposed System and Benefits

The proposed project is a web-based restaurant management system for Kas Valley Restaurant. The system supports online food ordering and reservation, thus enabling customers to place orders and make table reservations online. It also allows the restaurant to process in-person orders and table reservations. This RMS provides an online payment method for customers who wish to make direct online payments for their orders. A notification system is integrated with the application to alert customers when their meals are ready or dispatched. The system also allows the restaurant to manage their data records, including data on customers and orders. Additionally, this system provides data visualization and analytics on the restaurant’s sales and customers to help them make more informed business decisions.

This RMS will significantly benefit both the restaurant and the customer. Customers would be able to place orders in a more efficient, convenient and faster manner, thus increasing customer satisfaction and retention. Kas Valley restaurant will save labour costs, enhance business productivity, and improve its customer services using the web-based RMS.

1.8 Conclusion

In summary, this chapter covers the problem this project seeks to achieve. It introduces the software to be developed, a web-based restaurant management system. The chapter also covers the aim and objectives of the project, the literature review and some related works. The next chapter introduces the requirement analysis.

Chapter 2: Requirement Analysis

2.1 Introduction

This chapter discusses the requirement analysis process and the development methodology used for this project. Carrying out requirement elicitation and analysis is of topmost importance as it details the user's expectations and the system requirements. This chapter also outlines the data gathering process and the results obtained. Finally, this chapter discusses the use cases of the proposed system with the aid of a use case diagram.

2.1.1 Project Scope

This project consists of developing a user-oriented web-based restaurant management system for Kas Valley restaurant located in the Oyibi township of Accra, Ghana. The stakeholders involved are Kas Valley and people who patronize their services, primarily the youth and the working population. The proposed system in this project integrates an e-commerce platform in a web application to enable customers to place and modify their orders easily. Also, this project uses an Application Programming Interface (API) called Paystack, which will facilitate direct online payment for customers who want to pay via mobile money or visa cards. There is also a table reservation system on the web application to enable users make table reservations for various events. Also, an SMS messaging API called Twilio would be used to send alerts and notifications to customers when their orders have been processed.

This project's scope goes beyond online food ordering to provide the restaurant with an admin dashboard from which they can process all orders, view payments, manage table reservations, and monitor their sales growth. A JavaScript plugin called Chartjs would be used for data visualization and analytics on the restaurant's dashboard. Also, a library called PHPMailer would be used to send emails and newsletters to customers via the RMS. This project also involves data collection as gathering data will help inform the project and aid in the design of a user-oriented system. Interviews, online surveys and observations were used in the data gathering process. User testing and acceptance tests are also performed to ensure that the final product meets user expectations. In terms of project constraints, there would be no financial costs incurred throughout the implementation of the project. However, there is a time constraint as the project spans the entire fall semester (about five months), and the project is scheduled to end on May 10, 2022.

2.2 Software Methodology

The success of every project depends on the approach adopted during the lifetime of the project. The methodology adopted for this project was the Agile Methodology approach. The Agile Methodology uses iterative or incremental methods and factors customer involvement to enable teams to develop fast-paced prototypes until the final product is fully developed. The primary stakeholders, Kas Valley Restaurant, desired to partake in close collaboration and be included throughout the development of the proposed system to see the value building as the project unfolds and provide periodic feedback when needed. Agile methodology allows for this because it features customer collaborations and enables the development team to incorporate customer feedback at any stage in the development phase. Also, in the initial stage, the system's requirements were not clearly defined and were subject to change depending on the restaurant's suggestions and results obtained from the data gathering process. Agile development is most suitable for this because it allows for some flexibility in requirement modification.

2.3 Stakeholders and Description

The main stakeholders of this project are grouped into two. These are

- The restaurant: The restaurant staff or owner will be the system administrator. They have access to internet connectivity and are also technologically inclined enough to manage the system.
- Customers: These are essentially people who would want to order food online or make table reservations at Kas Valley Restaurant.

2.4 Approach to Data Collection

Data collection involves gathering and analyzing data to draw meaningful information and insights that will help inform the project. Thus to further understand the user needs and expectations of the proposed system, data was collected using several data collection methods. The methodologies employed for data gathering in this project include observation, interviews, online surveys and literary analysis. Both qualitative and quantitative data were obtained at the end of the process. Below is a brief description of how each data collection technique was applied:

I. Observations

Observation is a way of gathering data by studying or watching behavior of individuals, events and interactions as they occur in their natural setting. Specifically, this project used structured observation, and it involves collecting data per a pre-defined schedule. Observations were made at Kas Valley restaurant to study how the restaurant receives and processes customer orders as they occur. This approach was beneficial because it gave direct access to the research phenomena. Also, observations do not require people’s proactive participation as some individuals might be uncomfortable sharing information with the researcher. However, one limitation of this approach is that it can result in a biased interpretation, potentially leading to a personal viewpoint of the scenario rather than an objective interpretation.

II. Interviews

Interviews are used to gather qualitative data, and it involves a one-on-one direct engagement with individual participants. In this project, semi-structured interviews were used to obtain data from the restaurant staff. With semi-structured interviews, the researcher prepares a set of planned questions to ask participants. However, there is the freedom to modify the wording or order of the questions or ask additional questions that may pop up. Using this approach, the restaurant owner and some staff were asked a couple of questions regarding order-taking, table reservation, and data storage. This approach was advantageous as it presented the opportunity to ask follow-up questions and obtain inputs in desired detail.

III. Online Surveys

Online surveys were also used to gather data from the general public on their thoughts and expectations of online food-ordering systems. An online survey was used because it is the most cost-effective and convenient approach as it guarantees a broader reach of people with minimum required resources. Survey questions were developed and disseminated using the internet to obtain responses from people on the topic under study. It included several closed-ended questions and a few open-ended questions. The survey was done using Google forms, and the results were collected and analyzed by the softwre, thus saving a lot of time and human effort. A total of 215 responses were obtained on this survey.

IV. Literary Analysis

In this approach, data was gathered by reviewing literature and existing solutions in restaurant management systems. This approach is more of a formal data collection process where data is collected in a structured and comprehensive way.

2.4.1 Results and Insights from Data collection

The online survey used in this project obtained a total of 215 responses. Find below some important information on the respondents' demographics and responses:

- The majority of the participants fall in the youthful population. About 91.6% of the respondents are between the ages of 16 and 30 years. 6% are in the age range 31-45 years. 2% are in the age range 45-60 years.
- 58.6% of the respondents are females, and 40.9% are males
- According to the responses, about half of them *seldom* order food online. 27.9% of the participants *often* order meals online. 5.6% order food online *very often,* while 9.3% *never* order meals online.
- In assessing the average number of times people order food online in a week, about 48.8% order food online *once* a week. 34.9% of the respondents' order meals online *2-4 times* on average in a week. On average, 3.7% order online *5-7 times* in a week and only one respondent orders meals online *more than 7 times* in a week. 12% said on average, they order meals online *0 times* a week.
- To further understand users' preferences when it comes to the method of ordering meals, about 51.2 % preferred ordering meals via *phone calls*, 50.7 % liked *mobile apps*, 14.9% chose *websites,* and 12.1% preferred ordering meals via the restaurant's *social media accounts*
- In measuring the most important factors people consider when ordering food online, 51.2% consider the price of food, 56.7% consider the price of delivery, 31.6% consider an easy-to-use platform, 27.4% consider the security of the system, and 54.4% consider the speed at which orders are processed.

- The survey asked whether people will use a standalone online restaurant food ordering system instead of popular delivery applications (such as UberEATS, Jumia food, Bolt food) that integrate several restaurants on their sites. 50.2% of the participants responded *yes* that they would use a restaurant's online ordering system to order meals online. 37.2% selected *maybe,* and 12.6% selected *no.*
- To assess the preferred payment method for online food ordering systems, 79% of the respondents chose *mobile money* as their preferred payment method, 4.6% chose payment via *bank cards*, and about 16% selected *physical cash.*
- Online food ordering systems come with so many benefits. An open-ended question was asked in the survey to determine why people would opt to use these systems. Some of the reasons participants gave for ordering meals online include:
    1. It is convenient, fast, reliable and easily accessible
    2. Saves time and cost
    3. Laziness to cook, busy schedules, to avoid bad customer service
    4. Limit chances of getting covid-19

Additionally, valuable insights were also obtained from the observations and interviews conducted. Find below some of the insights derived:

- Through observations at Kas Valley restaurant, it was noticed that there are relatively longer queues during peak hours in the restaurant, that is, between 2 pm to 5 pm, and customers have to wait longer than usual to receive their orders.
- From the semi-structured interviews, the restaurant manager mentioned that as orders are increasing and their business keeps expanding, they would want to have

an alternative system for food ordering in addition to the conventional method being used at the restaurant.

- Additionally, the restaurant manager highlighted that their business needs a single system that would integrate multiple aspects of restaurant management, such as online food ordering, onsite order processing and table reservation management.
- The restaurant manager also mentioned that it is difficult to check which days and times have been fully booked for reservations. Having a system to perform this particular task would be of tremendous help.

2.5 Requirement Specification

The requirements of the system are categorized into functional and non-functional requirements. These requirements were obtained through interactions with the restaurant owner and the results obtained from the data gathering process.

2.5.1 Functional Requirements

Functional requirements are the things the system should do. It defines what the system must do or not do and allows it to function as intended. Below are the functional requirements of the system:

I. Functional requirements for Customer

[FR01]: The login and signup functionality on the online system shall allow customers to register a new account or log into system with an existing account

[FR02]: The e-commerce feature should enable customers to view the available dishes of the restaurant, place orders and make payments on the web application.

[FR03]: The cart system shall allow customers to add a dish to the cart, update the quantity or remove dishes from the cart.

[FR04]: The Paystack API shall enable customers to make direct payments online for their orders via mobile money or debit cards.

[FR05]: The table reservation system shall allow customers to make table reservations for a given occasion and time frame.

[FR06]: The online system shall allow customers to view their orders and table reservations and delete this history if they wish.

[FR07]: The search feature shall allow customers to find available dishes on the site

[FR08]: The log out feature on the online system should allow customers log out of their accounts

[FR09]: Alert and notification system should alert customers about the status of their orders using the Twilio API

[FR10]: The customer shall be able to cancel an order or reservation.

[FR11]: The buy now option shall allow the customer to buy a specific item immediately

II. Functional requirements for Restaurant

[FR011]: The login system for the restaurant shall authenticate all admin logins

[FR012]: The Chartjs plugin shall provide data visualization on the restaurant's dashboard to enable them to view analytics on their orders and customers.

[FR013]: The system shall allow the restaurant to maintain an online menu by enabling them to add, modify or delete a menu item.

[FR014]: The restaurant shall be able to view and process all online orders from customers by updating the order status to pending, received, completed or cancelled.

[FR015]: The table reservation system shall allow the restaurant to manage reservations; check if a particular date and time have been fully booked, enter new reservations, modify or delete reservations.

[FR016]: The customer management section of the restaurant's dashboard shall enable the restaurant to add, delete and modify customer records as well as view staff details

[FR017]: The admin system shall allow the restaurant to view all the website messages.

[FR018]: The subscription section should enable users to manage subscribers and send newsletters to them via the PHPMailer library

[FR019]: The search feature on the restaurant's view shall allow the restaurant to search for orders and reservations relating to a particular date

2.5.2 Non-Functional Requirements

Non-functional requirements describe how a system should behave and what limits there are on the system's functionalities. Below are the non-functional requirements of the system under consideration:

[NFR01] – Usability: The application interface should be user-friendly so customers can navigate the application easily.

[NFR02] - Security: The system should ensure confidentiality and authenticate customers before logging them into the system so that the customer's data is secure and protected from outsiders.

[NFR03] - Portability: The online application should function appropriately on varying devices, be it mobile phones or P.C.s.

[NFR04] - Maintainability: The system should be easy to maintain and upgrade to meet customers' changing needs.

[NFR05] - Availability: The online application should be readily available and accessible at a given point in time.

[NFR06] - Data integrity: The system should ensure data accuracy and consistency. Data entered by the user as input must be validated, if necessary, before being sent for processing.

2.6 Use Cases

Use Case Scenario 1

Ama is a student at Central University. She is famished, and she wants to get some pizza from her favourite restaurant, Kas Valley restaurant. However, Ama is exhausted and cannot travel to the restaurant to purchase her food. Ama wants to order her pizza remotely, make payment via mobile money and check the status of her order after purchase.

Use Case Scenario 2

Mr. Gyan, a businessman, wants to plan a little get together with some of his old colleagues from university. He is a regular customer at Kas Valley Restaurant, and he wishes to have the get-together there. However, there was a time when he booked a table reservation at the restaurant via a phone call, but they failed to capture some details of his request. Thus, he wants to secure the reservation in a manner that will have minimal human errors.

Use Case Scenario 3

Kas Valley Restaurant is receiving an increasing number of orders every day. The business wants to find alternative ways for customers to place their orders and make their reservations to reduce long queues, reduce pressure on staff and improve customer service. The restaurant wants to receive all these orders in a timely and orderly manner. However, to facilitate ordering using the internet, they need to find a way to get their menu available online and be able to make changes to the menu from their end as well.

2.7 Use Case Diagram

A use case diagram is a behaviour diagram which visualizes the observable interactions between actors and the system under development. It shows system-user interactions and captures the requirements of the system. Concerning this project, the system is a web-based RMS. The actors in the system consist of the restaurant and its customers. Below is a use case diagram showing the actors, use-cases and interactions within the web-based RMS:

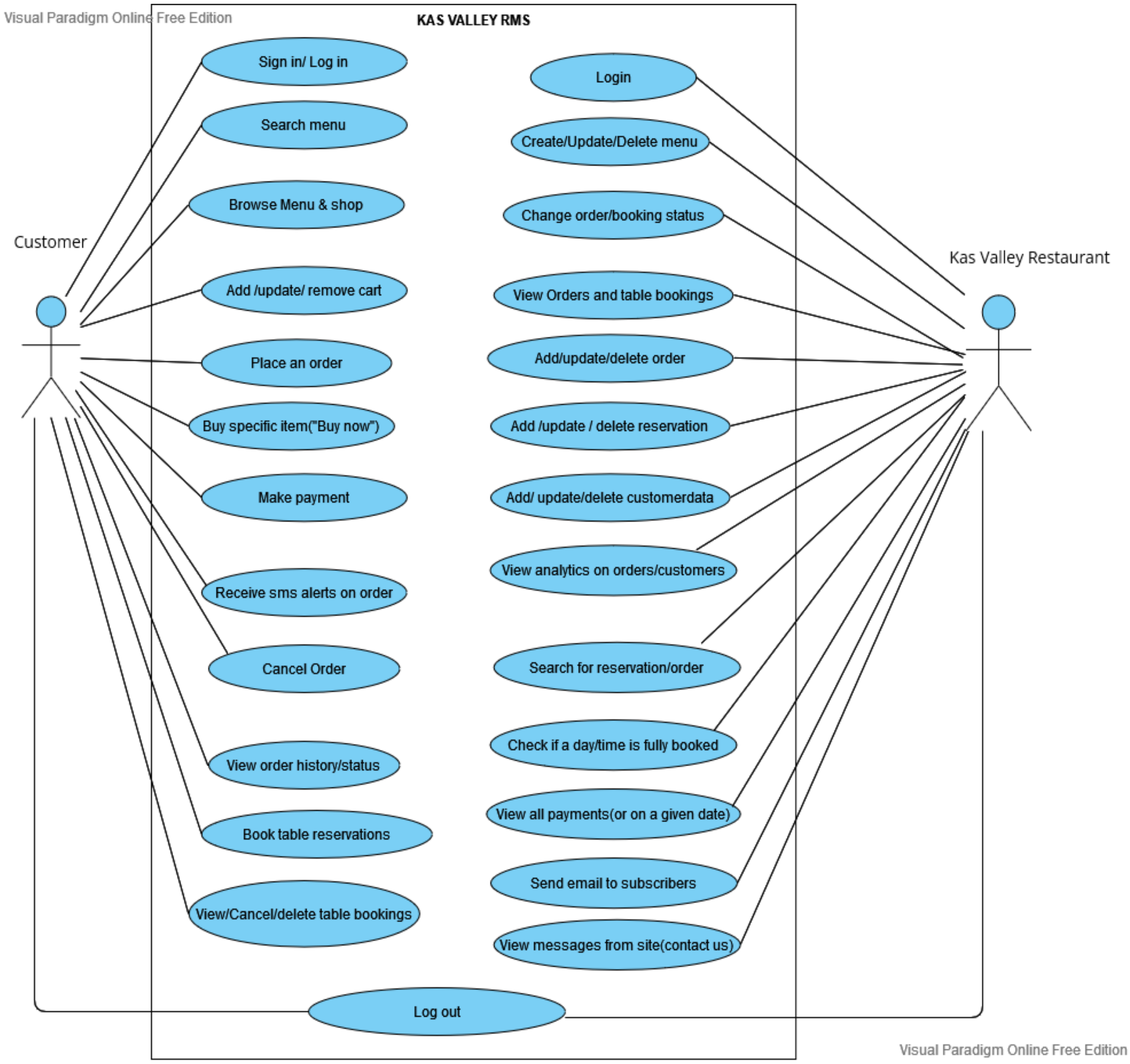

Visual Paradigm Online Free Edition
KAS VALLEY RMS
Customer
Kas Valley Restaurant
Sign in/ Log in
Search menu
Browse Menu & shop
Add /update/ remove cart
Place an order
Buy specific item("Buy now")
Make payment
Receive sms alerts on order
Cancel Order
View order history/status
Book table reservations
View/Cancel/delete table bookings
Login
Create/Update/Delete menu
Change order/booking status
View Orders and table bookings
Add/update/delete order
Add /update / delete reservation
Add/ update/delete customerdata
View analytics on orders/customers
Search for reservation/order
Check if a day/time is fully booked
View all payments(or on a given date)
Send email to subscribers
View messages from site(contact us)
Log out
Visual Paradigm Online Free Edition

Figure 2.1: Use-case diagram

2.7 CONCLUSION

In this chapter, the requirement elicitation process was discussed in detail. Data was gathered through observations, semi-structured interviews, online surveys and literary analysis. Through this, a lot of insights were obtained, and the requirements of the system were redefined. This chapter went further to discuss the system’s use cases with the aid of a use case diagram.

Chapter 3: System Design and Architecture

3.1 Overview

The system design of this web-based restaurant application entails a breakdown of the interfaces, components and architecture of the application to ensure that it satisfies the specific needs and meets the user's requirements. To give an overview of the system design, unified modelling language (UML) diagrams and other architectural diagrams will be used to visually represent and express the information in the system's structure. The system will be described both structurally and behaviorally with the help of graphical notation. Also, the modules and layers in the application are discussed in this chapter.

3.1.1 Assumptions and Constraints

- Users of the system have access to internet connectivity
- There is a set number of restaurant staff to be registered on the system by the system developer

3.2 System Architecture and Design

System architecture involves the high-level structure and design of software system abstraction. The 4+1 architectural view model will be used to illustrate the system architecture of the RMS. This model is used to describe the architecture of a software system using several concurrent views. These views are the logical view, development view, process view and physical view.

3.2.1 Logical View (Class Diagram)

This view shows the key abstractions and components in a system and their interactions. Examples of such abstractions include objects and classes. A class diagram and a sequence diagram will be used to represent the logical view of the RMS. A class diagram maps out the structure of a particular system by modelling its classes, attributes, operations, and relationships between objects. A sequence diagram on the other hand is an interaction diagram that describes how operations are carried out in a system and the order in which they are carried out. It shows the interactions between objects, including the messages exchanged between them and the order in which they are exchanged. Find below the class diagram and sequence diagram for the restaurant management system:

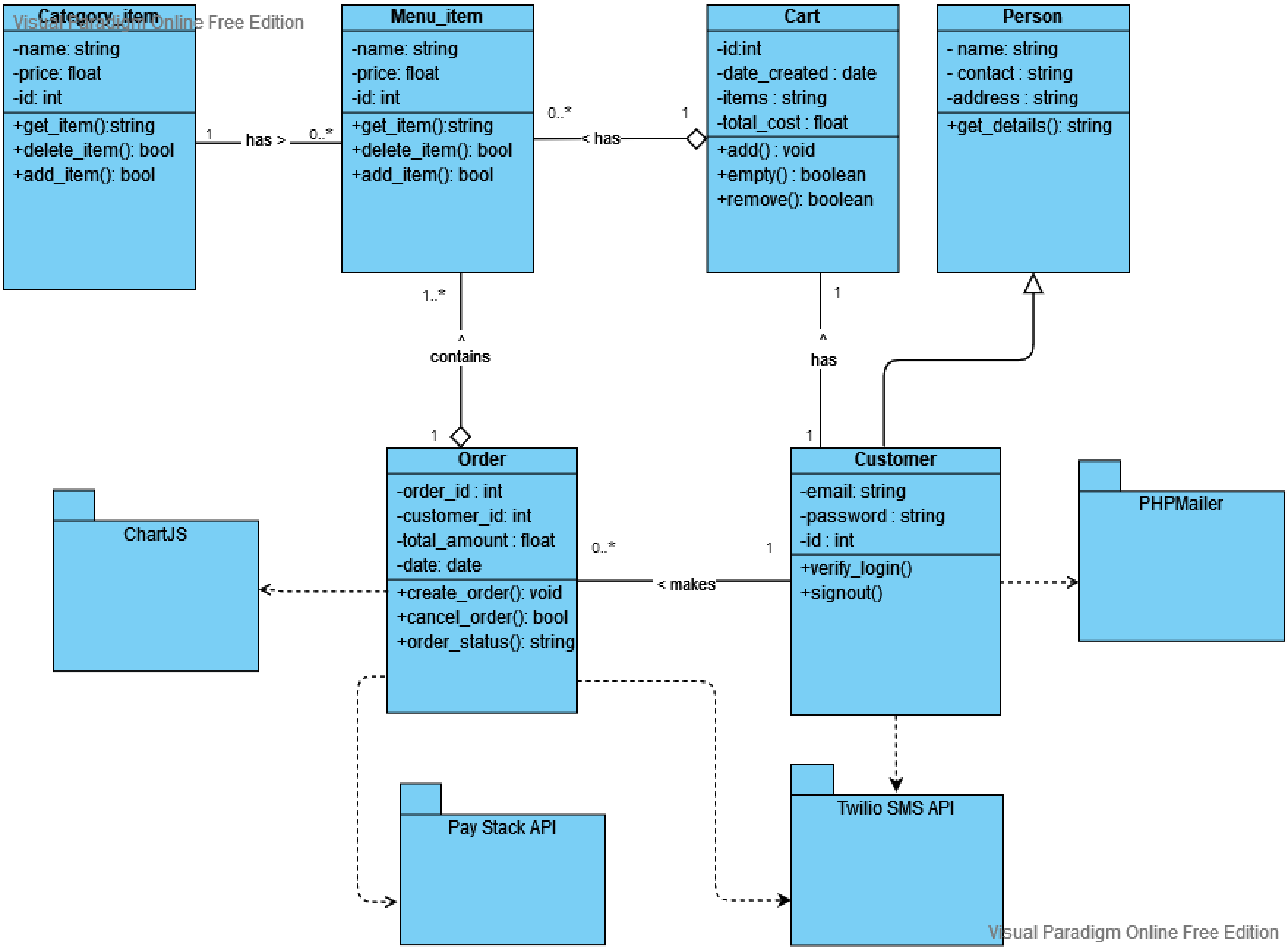


Figure 3.1: Class diagram

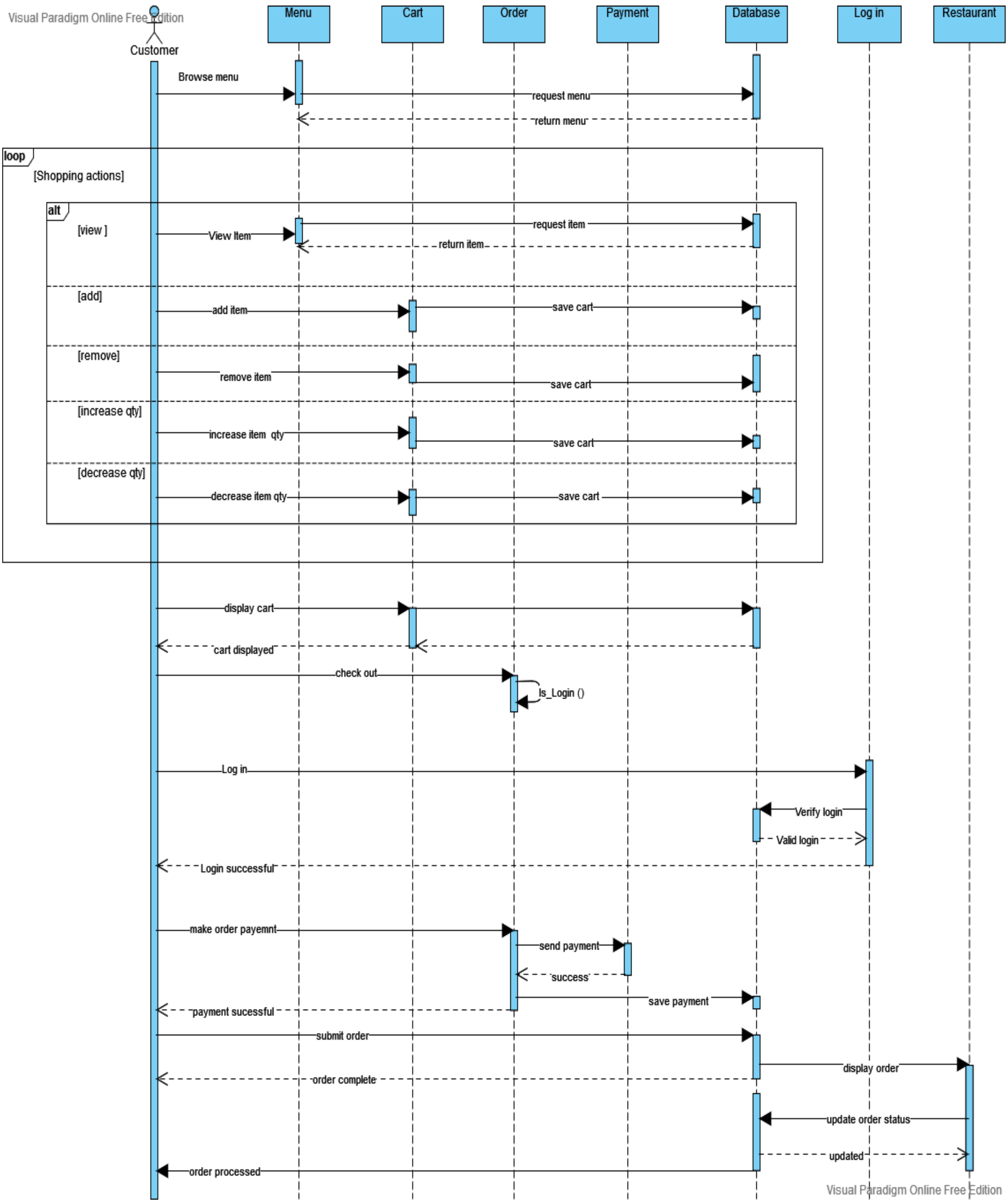


Figure 3.2: Sequence diagram for customer - Food ordering process

3.2.2 Process View (Activity Diagram)

This view shows the processes in the system at runtime and the communication that exists amongst the processes. An activity diagram will be used to illustrate this view. An activity diagram is a behavioural diagram that shows the workflow from a start point to a finish point detailing the various decision paths that exist while the activity is being executed. For clarity, there are two separate activity diagrams: one showing the customer's activities and the other showing the restaurant's actions.

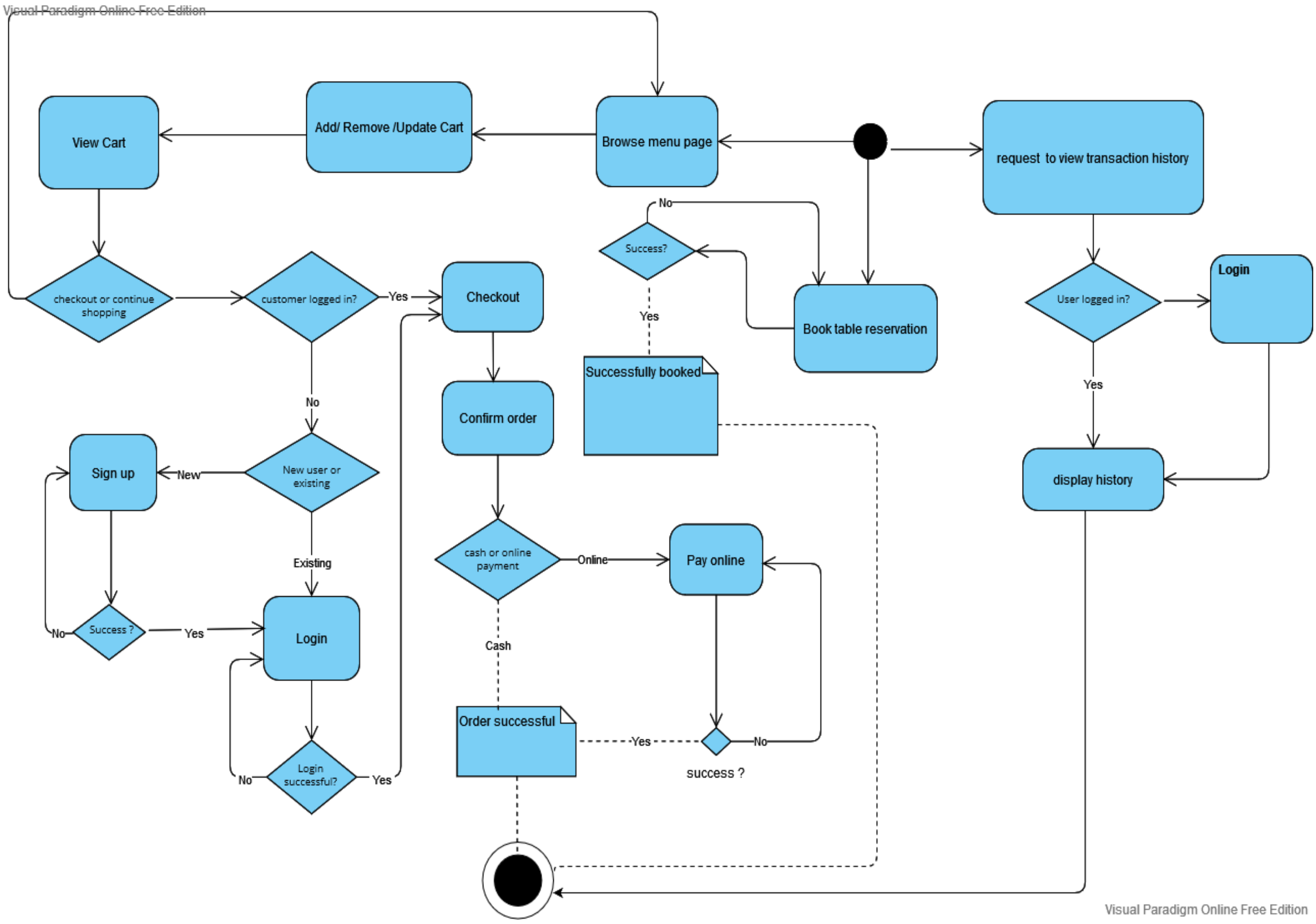


Figure 3.3: Activity diagram for customer

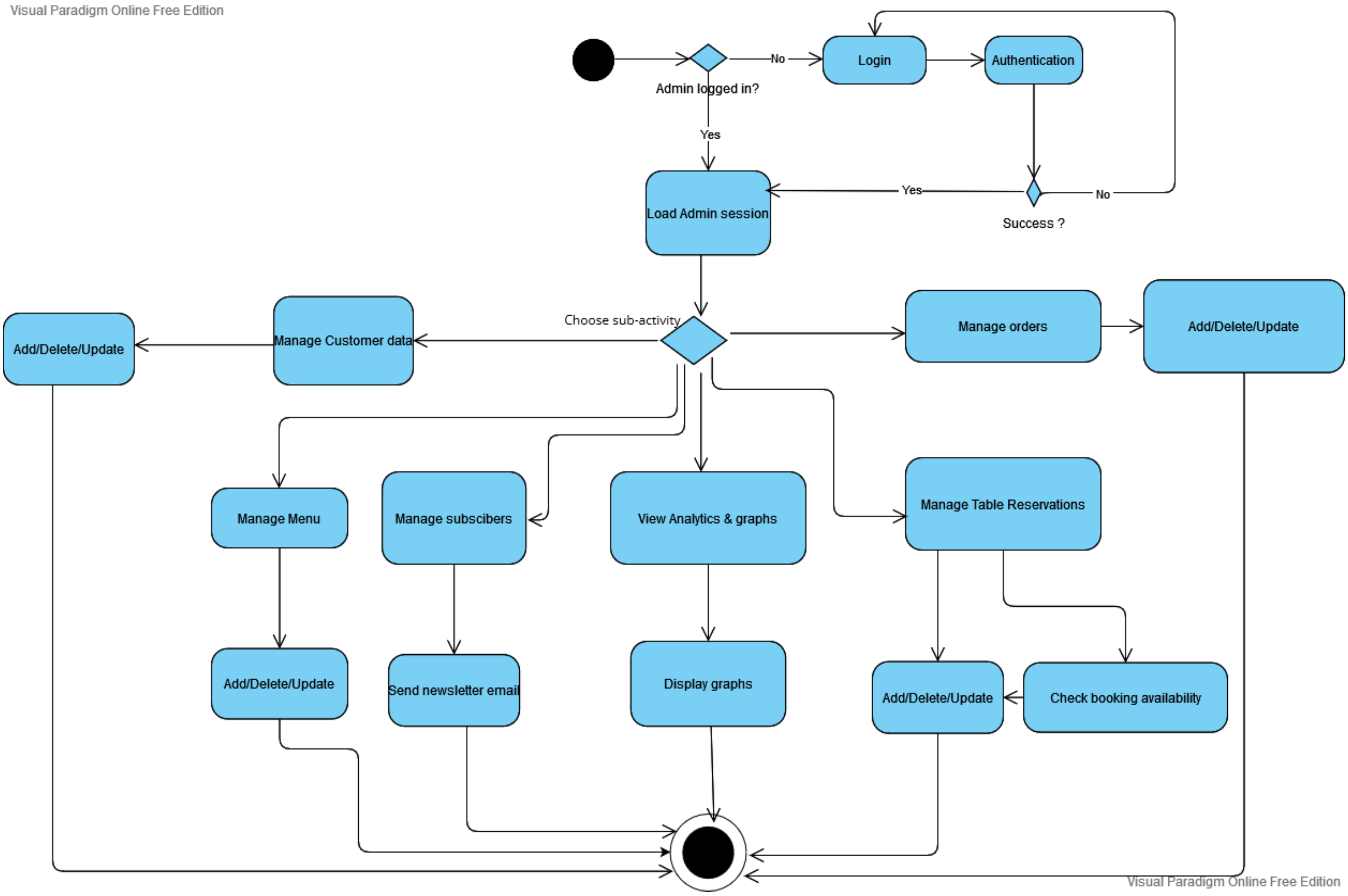


Figure 3.4: Activity diagram for restaurant administrator

3.2.3 Development View (Layered Architecture)

The development view focuses on the software module organization in the software development environment. It looks at how the system is decomposed for development. The layered architectural pattern will be used to illustrate the development view of the RMS. The layered architecture is also known as an n-tier architecture, where the tiers represent the layers of the system. This architecture comprises several separate horizontal layers that function together as a single software unit. This architecture is being used because it supports the incremental development of subsystems in layers, and this is in line with the software development methodology (Agile development) being used for this project.

There are three layers in the layered architecture pattern for this RMS. These are the presentation layer, the application layer and the database layer. The presentation layer is the front-end base, the user interface that the users directly interact with or have access to. The application layer hosts the backend code base, the business logic that processes requests from the presentation layer. It also acts as a medium for interaction between the presentation and database layer. The database layer stores the persistent data and supplies it to other layers when the need arises. Below is the 3-layered architecture of the system:

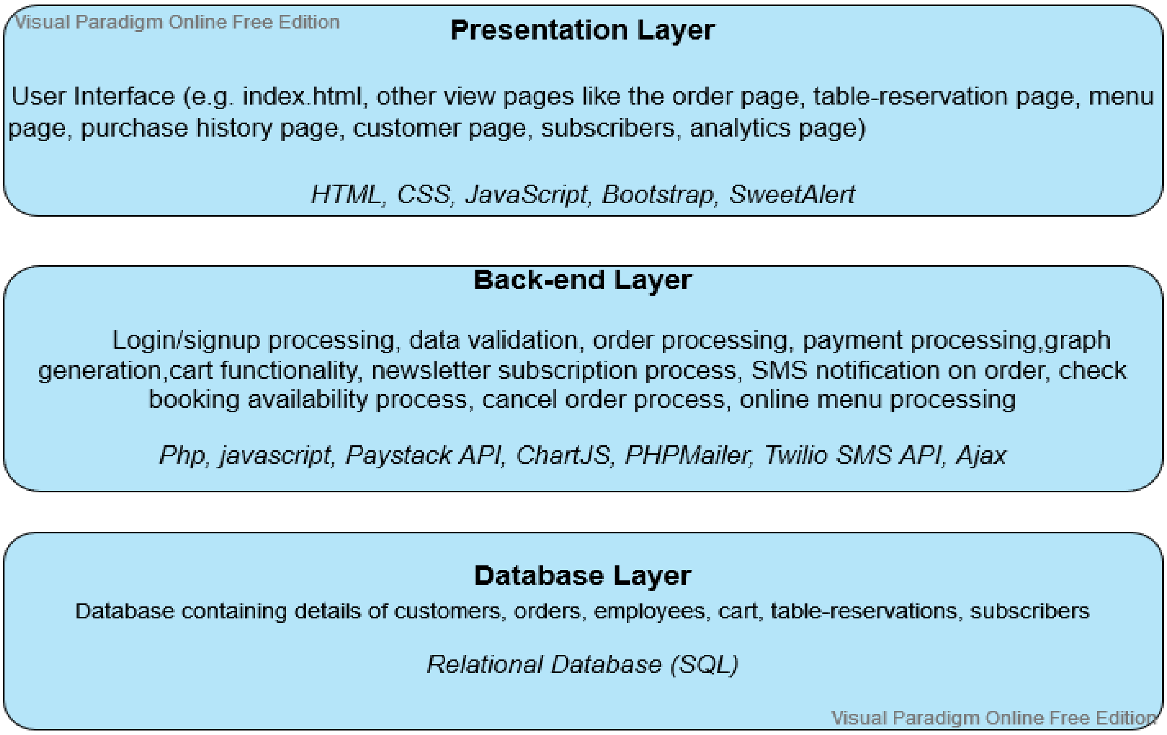

Visual Paradigm Online Free Edition
Presentation Layer
User Interface (e.g. index.html, other view pages like the order page, table-reservation page, menu page, purchase history page, customer page, subscribers, analytics page)
HTML, CSS, JavaScript, Bootstrap, SweetAlert
Back-end Layer
Login/signup processing, data validation, order processing, payment processing,graph generation,cart functionality, newsletter subscription process, SMS notification on order, check booking availability process, cancel order process, online menu processing
Php, javascript, Paystack API, ChartJS, PHPMailer, Twilio SMS API, Ajax
Database Layer
Database containing details of customers, orders, employees, cart, table-reservations, subscribers
Relational Database (SQL)
Visual Paradigm Online Free Edition

Figure 3.5: 3-layered architecture diagram

3.2.4 Entity Relationship (ER) Diagram

An E.R. diagram would be used to get a better and more detailed visualization of the database layer of the RMS. E.R. diagrams are graphical representations that show the relationships between entities, such as people and objects, within a system. It serves as the foundation for a relational database as it shows the complete logical structure of the database. Below is an E.R. diagram for the web-based RMS in this project:

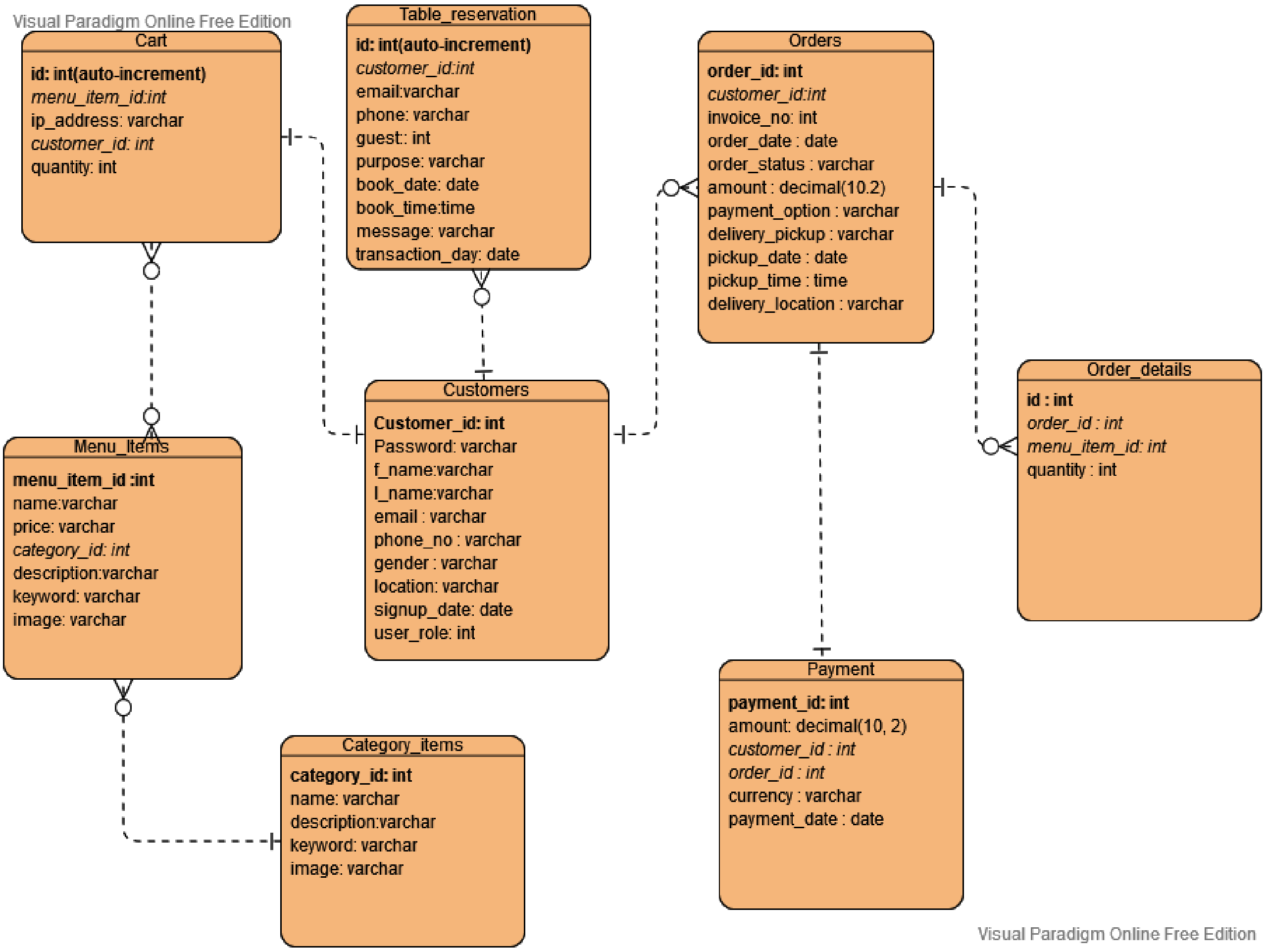


Figure 3.6: Entity relationship diagram

3.2.5 Physical View (Client-Server Architecture)

The physical view shows the system hardware, including the developed software application's configuration and deployment. A client-server architecture diagram will be used to show the physical view of the RMS. This architecture model describes how a server provides services to one or more clients. The clients, in this case, are client devices such as desktop computers, laptops or mobile phones on which, for instance, a customer might want to place an order. In addition, some of the servers that will be responding to such requests by the client include the web server, application server and database server. After development, the system will be hosted using web hosting service providers, and these providers have the servers needed to deploy web applications. Below is a diagram of the client-server architecture:

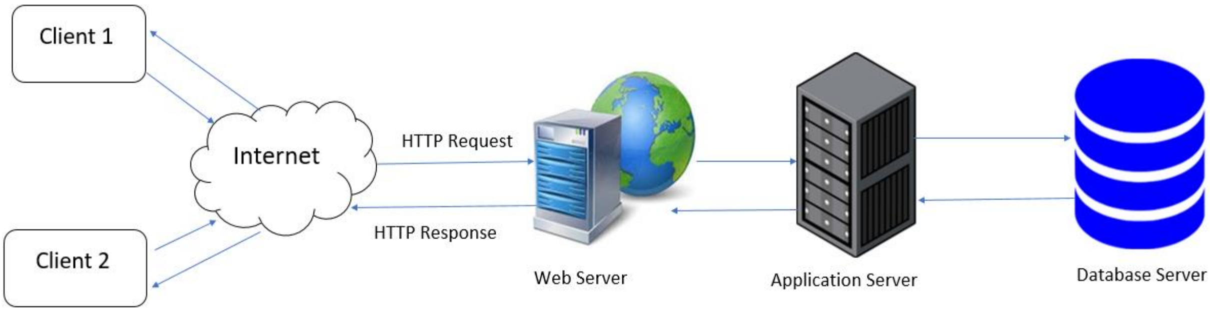


Figure 3.7: Client-server architecture diagram

3.3 Modules in the System

1. **Payment module**: This module is responsible for initiating and processing the payment transaction of the customer's orders. It uses Paystack API to support payment via mobile money and debit cards. Once a customer opts to pay for their order via mobile cash or debit card on the application, the payment module presents a Paystack dialogue box to the user for their mobile number or account number to complete the payment process.

2. **Cart module**: This module is responsible for carrying out all cart-related functionalities. This includes adding an item to the cart, updating the quantity of an item in the cart, deleting an item from the cart or emptying the cart. This module

uses Ajax to communicate with the server asynchronously to prevent the web page from reloading

3. **Product management module**: This module is responsible for presenting menu items to the customers and allowing the restaurant to manage these items. It enables the admin to add a new product and update or delete an existing product. Also, the module allows for products to be grouped into menu categories. The module also displays all the products to customers for them to shop.
4. **Graph module**: This module is responsible for displaying graphs on the admin system. It uses chart.js to feed data from the database to the chart and display it on the user interface of the admin side of the application.
5. **Customer module:** This module is responsible for handling the customer data in the system. It manages customer registration and signup. It also enables the admin to add a new customer, delete or update information on existing customers
6. **Reservation module**: This module is responsible for handling reservation management. It enables the customer to place an online reservation or cancel an already made reservation. The admin can also enter a reservation from their side of the system. This module also checks whether a day or time is fully booked before placing the reservation. All reservations made are displayed on the admin system.
7. **Order management module**: This module is responsible for handling order management. All orders made both online and by the restaurant are displayed on the admin side of the system. Using this module, the admin can change an order’s status, cancel, update or delete the order.

8. **Notification module:** This module is responsible for alerting customers about the status of their orders. It uses the Twilio SMS API to send text messages from the web application to the customer.

9. **Email module:** This module handles email processes on the web application. Using PHPMailer, the admin can email the restaurant's news subscribers with relevant information.

3.4 Conclusion

This chapter discusses the architecture and design of the proposed system. UML diagrams and graphical representations were used to show the logical view, development view, process view and physical view of the web-based RMS. The chapter ends with brief explanations of the modules in the RMS.

Chapter 4: Implementation

4.1 Overview

This chapter outlines the development of the application and the tools and resources used in the development phase. It provides details on frameworks, libraries and programming languages utilized in implementing the system. Additionally, this chapter discusses the implementation techniques used.

4.2 Tools and Technologies

Several tools and technologies were used in implementing the web-based RMS. This includes libraries, frameworks, APIs and several other technologies. For clarity, the tools have been categorized into four groups: Front-end, Back-end, Database and Other Tools.

4.2.1 Front-end Tools

These are essentially tools used to develop the front-end interface of the system. The front-end tools are HTML, CSS, JavaScript, Jquery and Bootstrap. Below is a brief explanation of each:

1. **HTML**

HTML is a standard markup language used to structure a web page and its content [26]. It is the foundation of a web page, and it is primarily used in web development, internet navigation, and web documentation.

2. **CSS**

CSS is a style sheet language used to describe the presentation of a document written in a markup language such as HTML [27]. Thus, CSS adds style (such as fonts, layouts and colour) to web pages to look more presentable.

3. JavaScript

JavaScript is a lightweight, object-oriented programming language and the most popular scripting language for web pages [28]. It is used for adding interactive behaviour on web pages and controlling the dynamic behavior of these web pages.

4. Bootstrap

Bootstrap is a front-end library and an open-source CSS framework aimed at designing responsive mobile-first websites [29]. It has HTML and CSS-based design templates for forms, navigation, buttons, other interface icons, and optional JavaScript plugins.

5. JQuery

JQuery is a JavaScript library that makes JavaScript programming relatively easier and simpler. It does this by providing an easy-to-use and cross-platform API that simplifies JavaScript processes such as HTML document traversal, event handling, CSS animation and Ajax [30]

4.2.2 Backend Tools

These are tools used for the backend development of the system. This includes server-side management, handling and processing incoming user requests, storing and updating records in the database, and accessing them. The primary backend technology for achieving these purposes in the web-based RMS is PHP ((Hypertext Preprocessor).

PHP: PHP is an interpreted object-oriented server scripting language purposely for developing interactive and dynamic web applications [31]. It creates dynamic pages that interact with the database and supports program execution on the server-side.

4.2.3 Database layer Tools

These are technologies used for the database layer, which acts as the persistent storage unit of the web application. In this project, MySQL is used as the database management system for the proposed application

MySQL: It is an open-source relational database management system (DBMS) on SQL (Structured Query Language) [32]. MySQL databases can easily be connected to PHP, thus making it easier to work with the two technologies. Also, MySQL is mainly used in web application development, making it a better option for this project.

4.2.4 Other Tools, Libraries and Technologies

Twilio SMS API: This is an API for sending and receiving text or SMS messages. It allows you to track the delivery of sent messages and retrieve message history. In this project, the Twilio API sends alerts and notifications to a customer about their order or reservation status.

Paystack API

Paystack API is an API that enables an application to accept payments from cards, mobile money accounts, and banks. It is used to make monetary transfers on an application. This API also verifies phone numbers and bank accounts or Bank Verification Numbers [33]. In the web-based RMS, Paystack is used to implement the payment module in the system.

SweetAlert

It is a JavaScript library, and it is a responsive and easily customizable replacement for JavaScript popup boxes. In this project, SweetAlert is used to provide feedback to users to let them know if an action performed was successful.

PhpMailer

This is a code library for sending emails safely and efficiently through a web server using PHP [34]. It has SMTP support and transfers emails without a local mail server. It also validates email addresses automatically and allows for the inclusion of attachments. PHPMailer is used in the proposed system to send emails to customers.

Chart.js

Chart.js is an open-source JavaScript library for data visualization, and it comes with eight in-built chart types, including scatter plot, bubble, bar, line, radar, polar and pie(doughnut) [35]. In this project, Chart.js is used to create the graphs in the web-based RMS.

Ajax (Asynchronous JavaScript and XML)

Ajax is a mechanism that uses inter-related web technologies including JavaScript, DOM, XML, HTML/XHTML, CSS and XMLHttpRequest to allow web applications to send and receive data from the server without reloading the web page [36]. Thus, it makes web applications faster and more responsive to user actions since it allows for quick and incremental updates to the user interface without reloading the entire web browser. In this project, Ajax is used in the implementation of the cart functionalities (adding, deleting, and updating quantity in the cart)

JSON (JavaScript Object Notation)

It is a text format based on a collection of name/value pairs, and it is used for storing and transporting data, especially when data is sent from a server to a web page [37]. In this project, JSON is used for packaging information in the Ajax model.

Apache Xampp Server: XAMPP is an open-source, cross-platform web server from Apache distributions. It consists of an Apache HTTP server, an interpreter for different programming languages like PHP and Perl, and database management systems like

MariaDB and MySQL [38]. In this project, xampp is used for the local development of the web-based RMS.

PhpMyAdmin: An open-source administration tool for MySQL and MariaDB written in PHP [39]. PhpMyAdmin is used to manage the database in this project.

000webhost: This is a free web-hosting site that comes with a cPanel, and it supports PHP and MySQL. 000webhost is used to deploy the web-based RMS in this project.

4.4 User interface and functionality

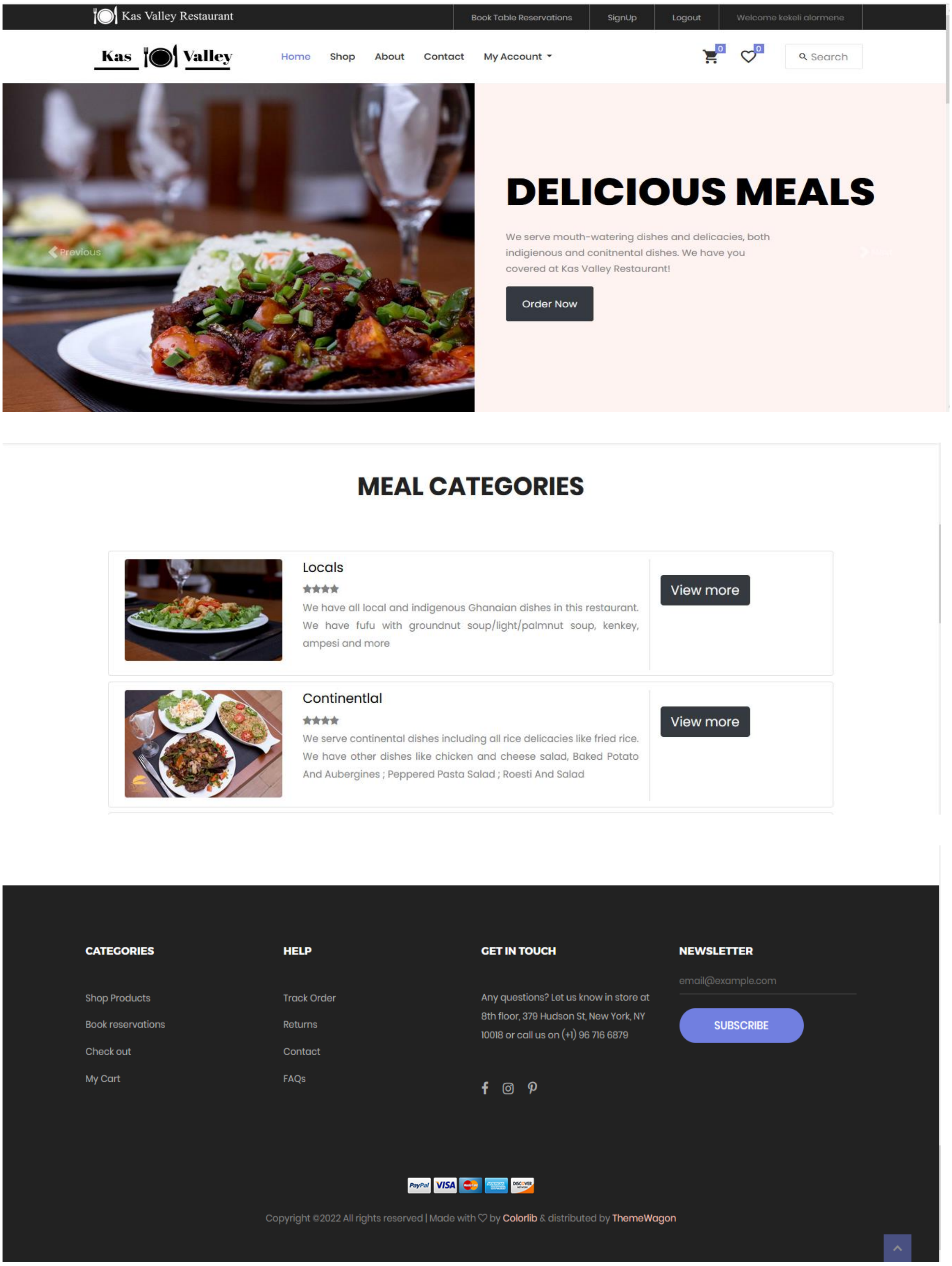
This subsection will discuss the implementation of the various user interfaces in the system. The user interfaces are separated into two categories: Customer pages and Restaurant Admin pages. Also, snippets of some aspects of the code are included in this section.

4.4.1 Customer Pages

These are the user interfaces or web pages related to customer actions on the system. This includes the signup page, login page, order page, and cart page, to mention a few. Below is a detailed explanation of the pages on the customer side of the system:

1. **Landing Page**

The page also shows the various meal categories, with each category containing a link to another page that will display all the products in that category. The page has a navigation panel showing the number of items in the cart and a search bar to look for meals. There is also a footer with a section where new users can subscribe to the restaurant’s content. Below is an image of the landing page for the customer:



DELICIOUS MEALS

We serve mouth-watering dishes and delicacies, both indigienous and conitnental dishes. We have you covered at Kas Valley Restaurant!

MEAL CATEGORIES

Locals

We have all local and indigenous Ghanaian dishes in this restaurant. We have fufu with groundnut soup/light/palmnut soup, kenkey, ampesi and more

Continentlal

We serve continental dishes including all rice delicacies like fried rice. We have other dishes like chicken and cheese salad, Baked Potato And Aubergines ; Peppered Pasta Salad ; Roesti And Salad

Any questions? Let us know in store at 8th floor, 379 Hudson St, New York, NY 10018 or call us on (+1) 96 716 6879

Copyright ©2022 All rights reserved | Made with ♡ by Colorlib & distributed by ThemeWagon

Figure 4.1: Landing page of customer-side of the application

2. Shopping page

This page implements a cart system where customers can either add, update an item’s quantity, or delete an item from the cart. There is a mini-cart feature on this page that supports cart functionalities. Ajax is used in adding, deleting, and updating the cart so that the webpage does not have to reload every time the user performs an action. This page shows all the menu items available for order, and for each item, there is the option to add to the cart or view product details, where you can then add to the cart or buy immediately. Also, there is a navigation menu of all the product categories of the restaurant’s dishes. Below is the shopping page:

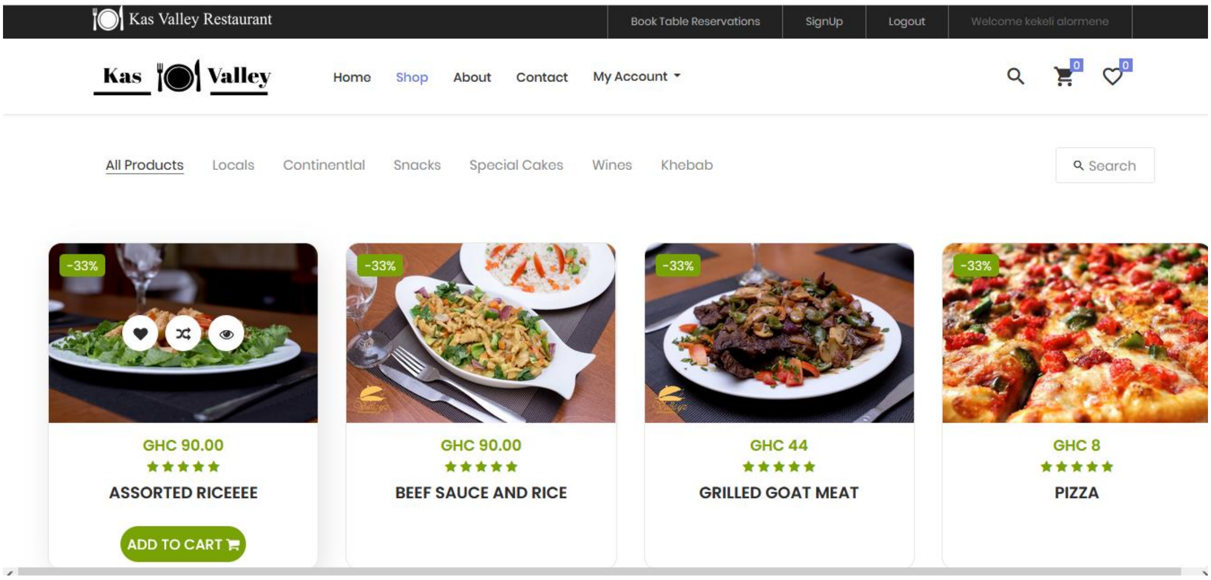
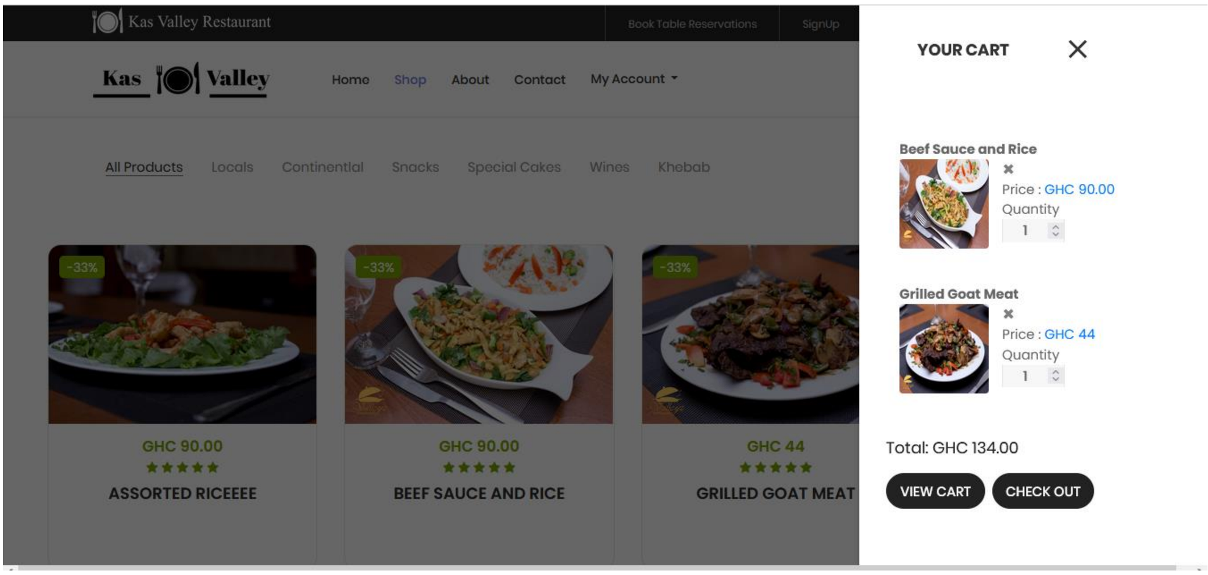


Figure 4.2: Shopping Page



Beef Sauce and Rice
Price : GHC 90.00
Quantity 1

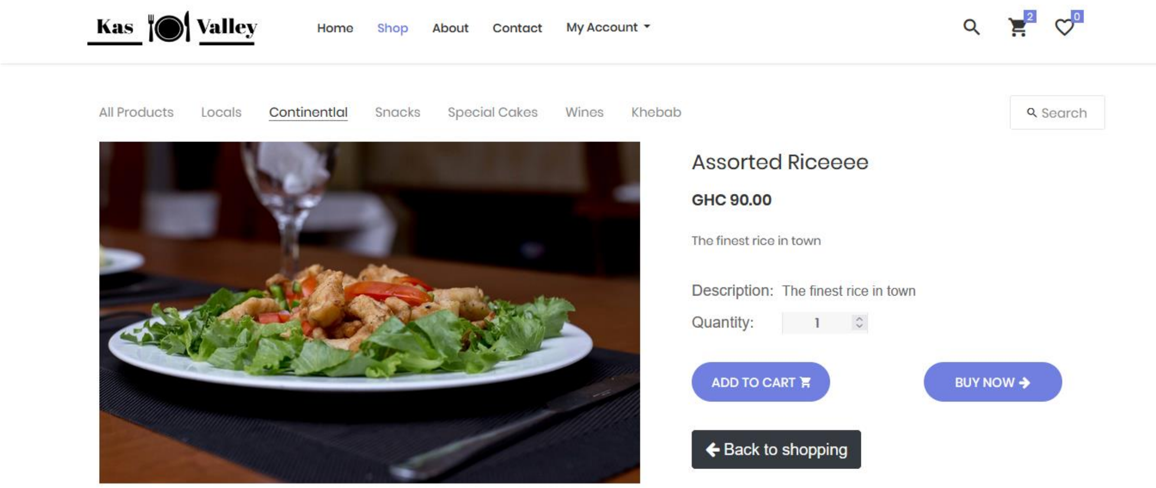
Grilled Goat Meat
Price : GHC 44
Quantity 1

Total: GHC 134.00

Figure 4.3: View of the mini cart on the shopping page

3. Product-Detail Page

This page allows the user to view the details of a particular menu item. You can then add to the cart or purchase immediately by clicking the "buy now" option on this page. Below is an image of this page:



Assorted Riceeee

GHC 90.00

The finest rice in town

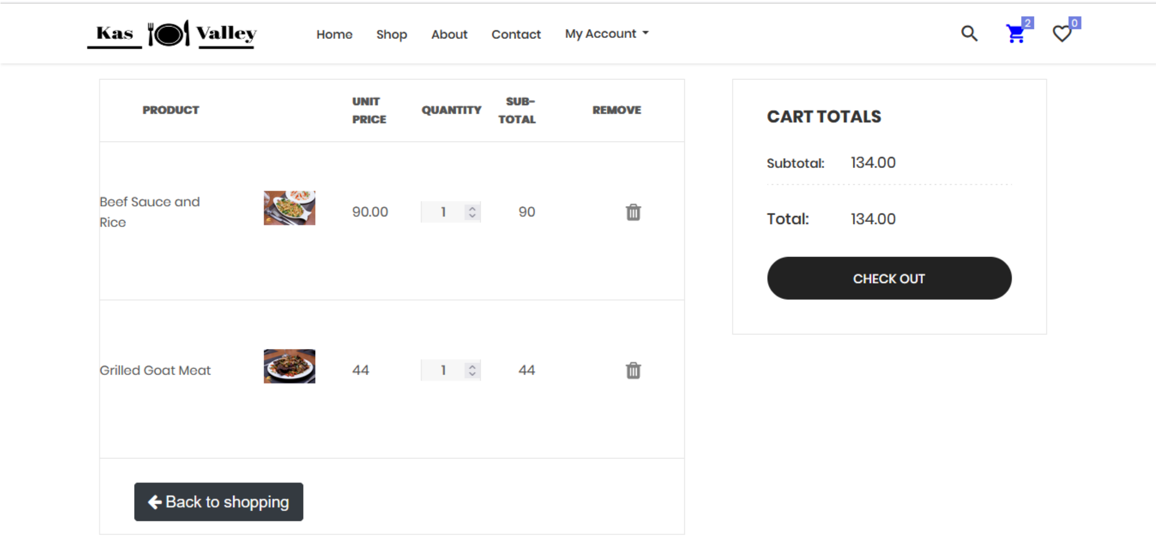
Description: The finest rice in town

Quantity: 1

Figure 4.4: Details of menu item page

4. Cart Page

This page shows the items in the cart and the subtotal of each item. It allows actions, including updating and deleting from the cart. It also contains a button that will enable you to proceed to checkout. An image of the cart page:



PRODUCT		UNIT PRICE	QUANTITY	SUB-TOTAL	REMOVE
Beef Sauce and Rice		90.00	1	90	🗑
Grilled Goat Meat		44	1	44	🗑

CART TOTALS

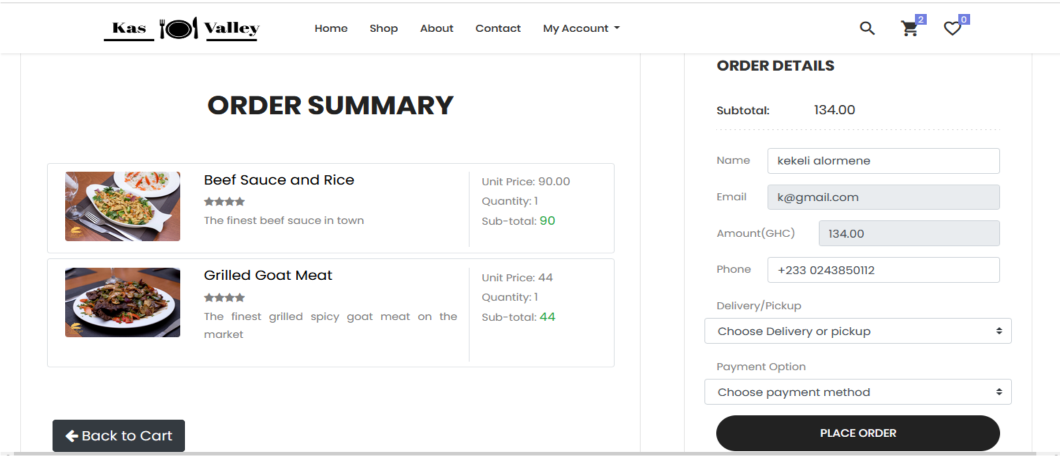
Subtotal: 134.00

Total: 134.00

Figure 4.5: Customer cart page

5. Checkout Page

It contains a summary of the customer's order and allows the customer to review their order before submitting it. A user cannot access this page without logging in.



ORDER SUMMARY

Beef Sauce and Rice
The finest beef sauce in town
Unit Price: 90.00
Quantity: 1
Sub-total: 90

Grilled Goat Meat
The finest grilled spicy goat meat on the market
Unit Price: 44
Quantity: 1
Sub-total: 44

ORDER DETAILS

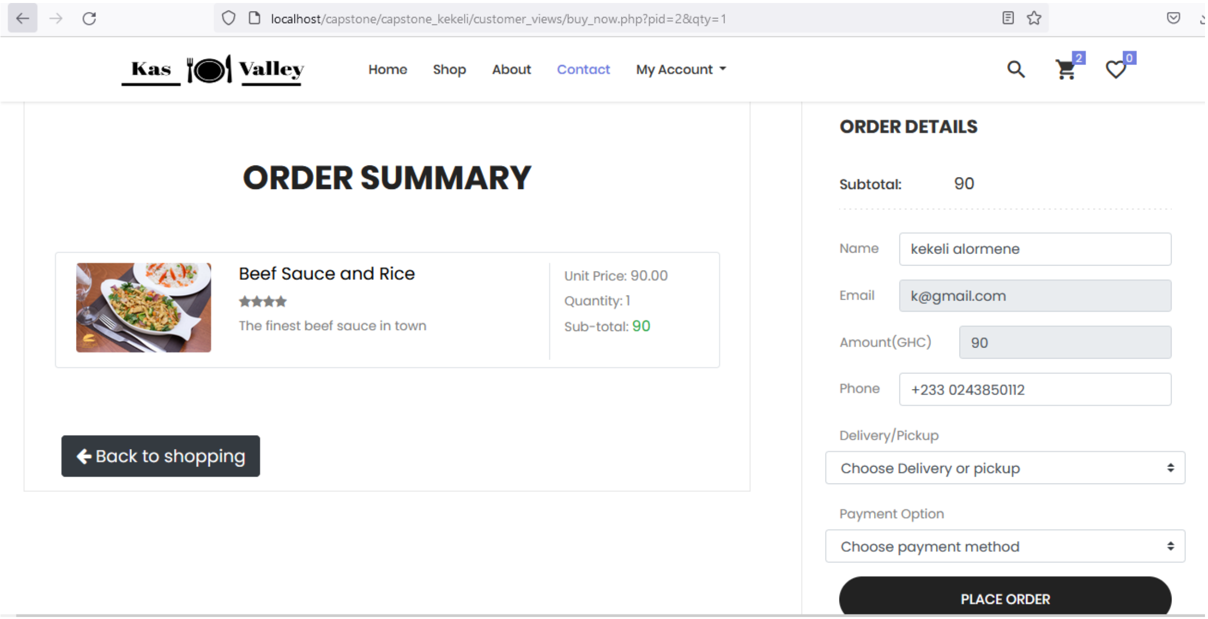
Subtotal: 134.00

Name: kekeli alormene
Email: k@gmail.com
Amount(GHC): 134.00
Phone: +233 0243850112
Delivery/Pickup: Choose Delivery or pickup
Payment Option: Choose payment method

Figure 4.6: Customer checkout page

6. Buy Now Page

It shows an item that a user selected using the “buy now” option.


localhost/capstone/capstone_kekeli/customer_views/buy_now.php?pid=2&qty=1
Kas Valley
Home
Shop
About
Contact
My Account
2
0
Back to shopping
PLACE ORDER

ORDER SUMMARY

Beef Sauce and Rice
The finest beef sauce in town
Unit Price: 90.00
Quantity: 1
Sub-total: 90

ORDER DETAILS

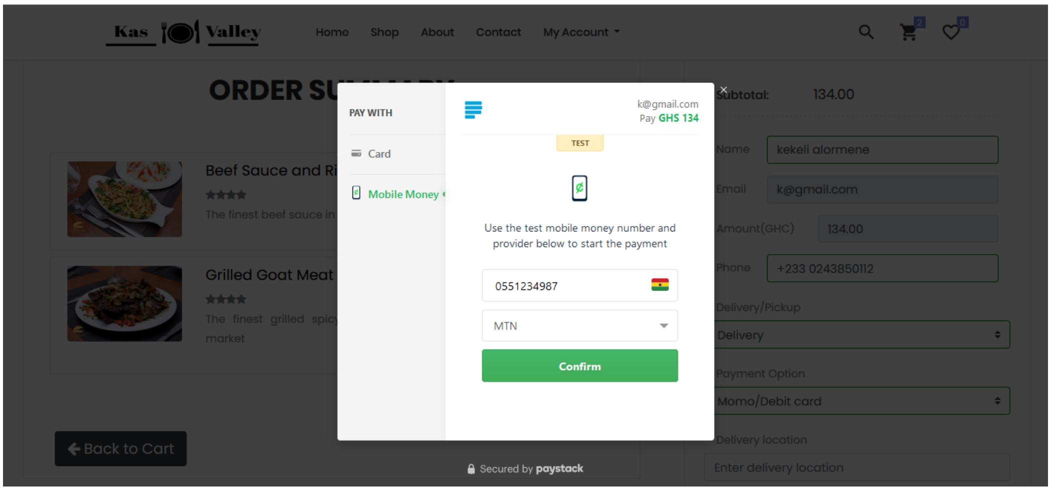
Subtotal: 90

Name: kekeli alormene
Email: k@gmail.com
Amount(GHC): 90
Phone: +233 0243850112
Delivery/Pickup: Choose Delivery or pickup
Payment Option: Choose payment method

Figure 4.7: Buy now (a single item) page

7. Payment Module using Paystack API

For customers who would like to make an online payment via mobile money or visa cards, the Paystack API has been integrated into the system to allow for this.



Beef Sauce and Ri

The finest beef sauce in

Grilled Goat Meat

The finest grilled spic market

k@gmail.com
Pay GHS 134

Use the test mobile money number and provider below to start the payment

0551234987

MTN

Subtotal: 134.00

Name: kekeli alormene

Email: k@gmail.com

Amount(GHC): 134.00

Phone: +233 0243850112

Delivery/Pickup: Delivery

Payment Option: Momo/Debit card

Delivery location: Enter delivery location

Figure 4.8: Paystack payment dialogue box

8. Sign up page

This page allows first-time users to sign up on the application. User input is validated using JavaScript and sent to the database afterwards.

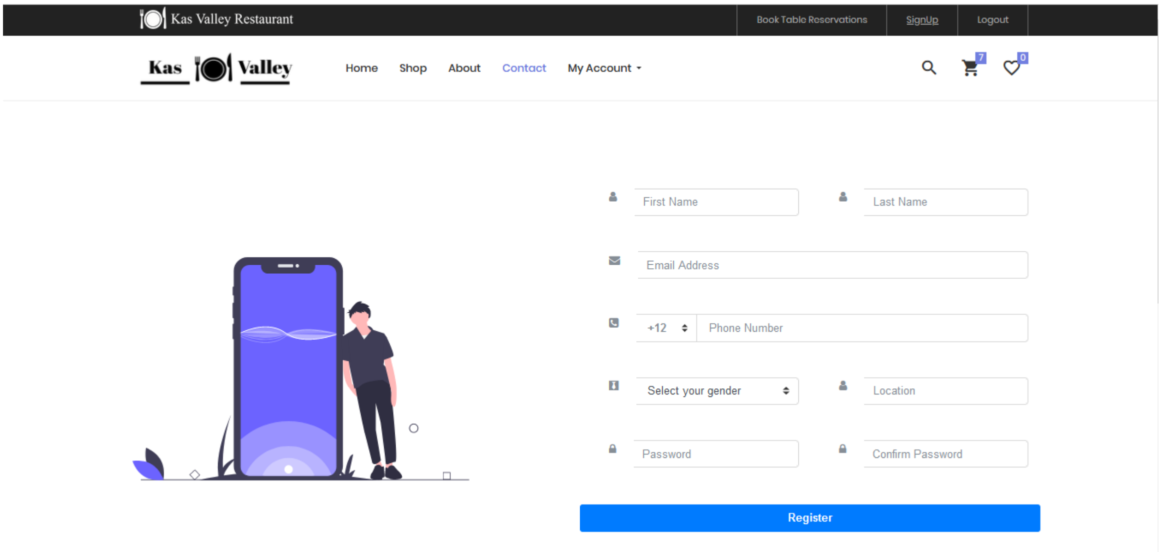
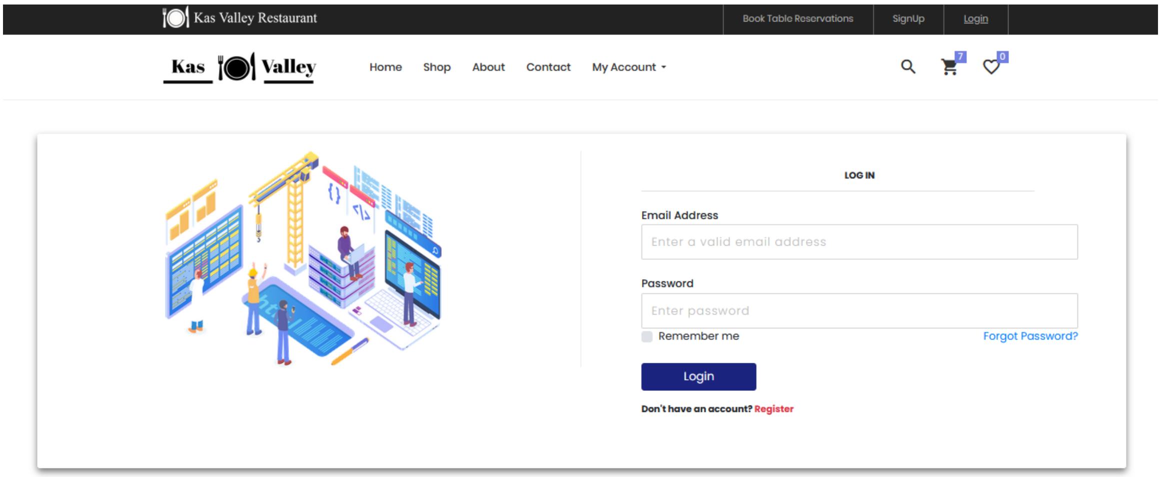


Figure 4.9: Customer sign up page

9. Login page

This page allows a registered user to log into the application in order. The user's inputs are matched against the user's credentials in the database. If they are equal, login is successful.



LOG IN

Email Address

Enter a valid email address

Password

Enter password

□ Remember me

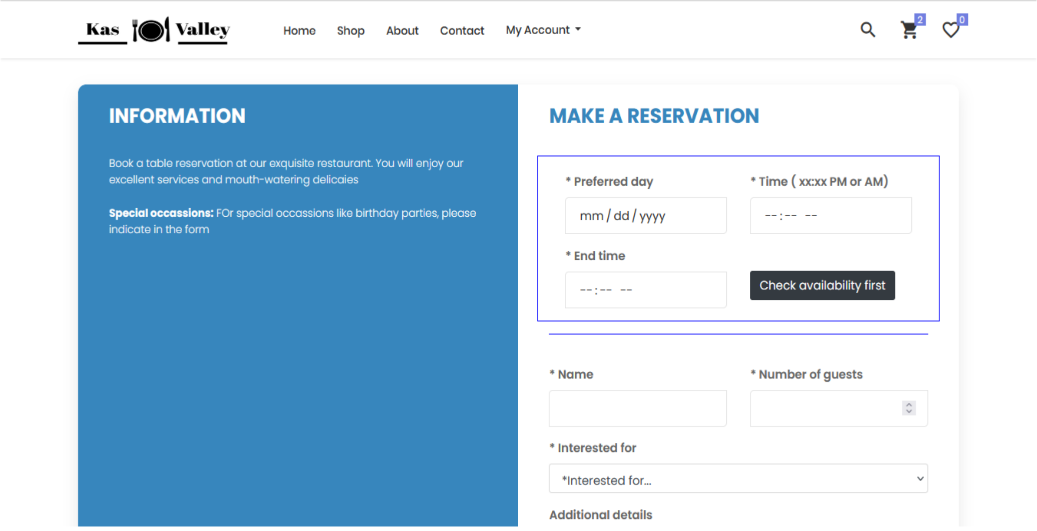
Forgot Password?

Don't have an account? Register

Figure 4.10: Customer login page

10. Table reservation page

This page enables a customer to make table reservations for an occasion on a specific date and time.



INFORMATION

Book a table reservation at our exquisite restaurant. You will enjoy our excellent services and mouth-watering delicaies

Special occassions: FOr special occassions like birthday parties, please indicate in the form

MAKE A RESERVATION

* Preferred day

mm / dd / yyyy

* Time (xx:xx PM or AM)

--:-- --

* End time

--:-- --

* Name

* Number of guests

* Interested for

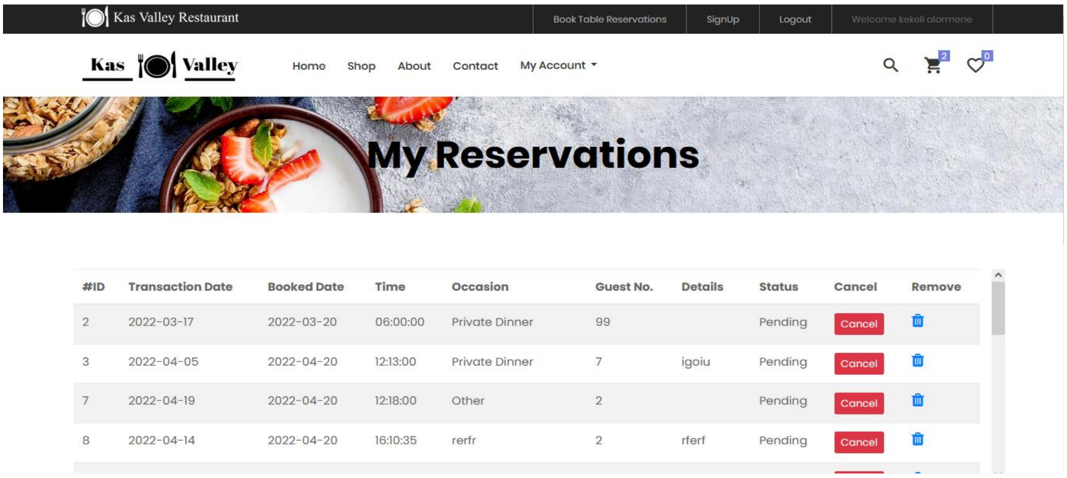
*Interested for...

Additional details

Figure 4.11: Customer reservations booking page

11. View reservations page

This page shows a history of all the table reservations a customer has made on the application.

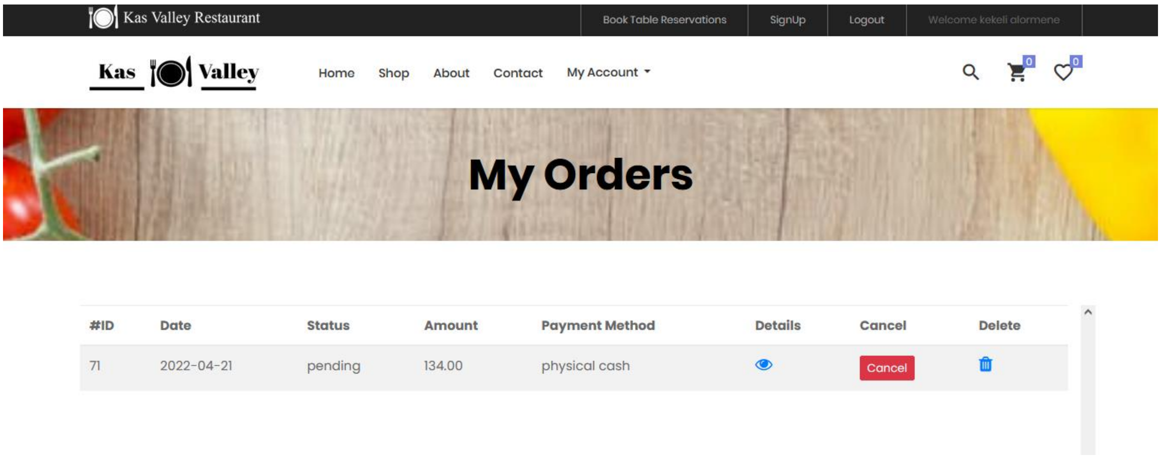


#ID	Transaction Date	Booked Date	Time	Occasion	Guest No.	Details	Status	Cancel	Remove
2	2022-03-17	2022-03-20	06:00:00	Private Dinner	99		Pending	Cancel	🗑
3	2022-04-05	2022-04-20	12:13:00	Private Dinner	7	igoiu	Pending	Cancel	🗑
7	2022-04-19	2022-04-20	12:18:00	Other	2		Pending	Cancel	🗑
8	2022-04-14	2022-04-20	16:10:35	rerfr	2	rferf	Pending	Cancel	🗑

Figure 4.12: View the reservations page for the customer side

12. View orders page

This page allows a customer to view their order history and the details associated with each order. Thus, a customer can also view the order status on a recently placed order on this page.

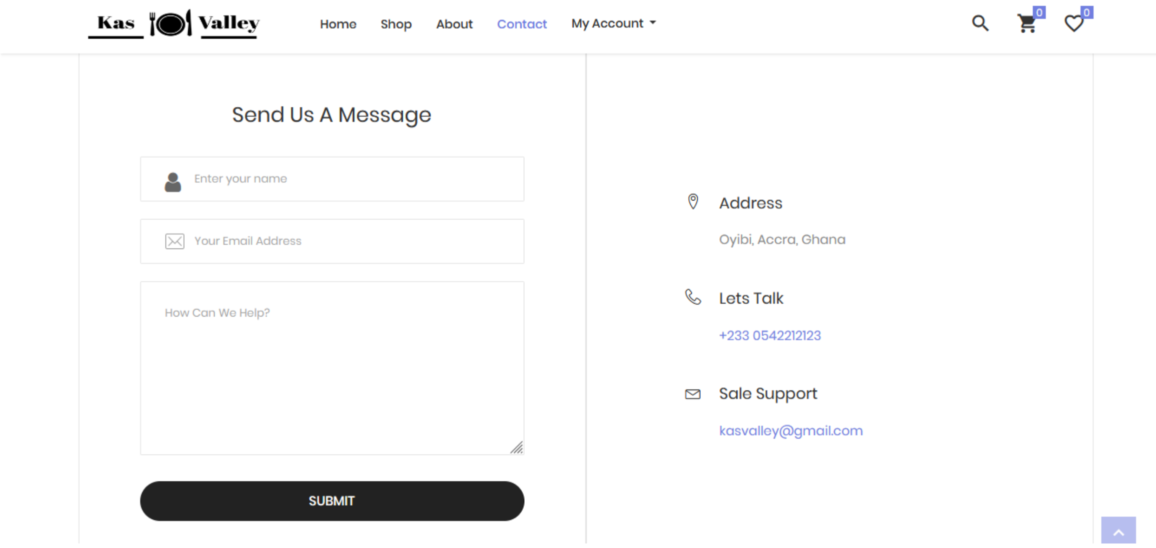


#ID	Date	Status	Amount	Payment Method	Details	Cancel	Delete
71	2022-04-21	pending	134.00	physical cash	👁	Cancel	🗑

Figure 4.13: View the orders page for the customer side

13. Contact Us Page

A page to allow customers to contact the restaurant via a form



Send Us A Message

Enter your name

Your Email Address

How Can We Help?

Address

Oyibi, Accra, Ghana

Lets Talk

+233 0542212123

Sale Support

kasvalley@gmail.com

Figure 4.14: Contact us page

4.4.2 Admin Pages

These are the user interfaces or web pages related to the system's admin (restaurant) actions. For example, the login page for the restaurant, add menu item page, add category item page, and view orders page. Below is a detailed explanation of the pages on the admin side of the system:

1. Login Page

The admin section has a login page for the admin to access the system. Login details entered are authenticated against the details in the database, and if it matches, the admin is logged in and redirected to the dashboard.

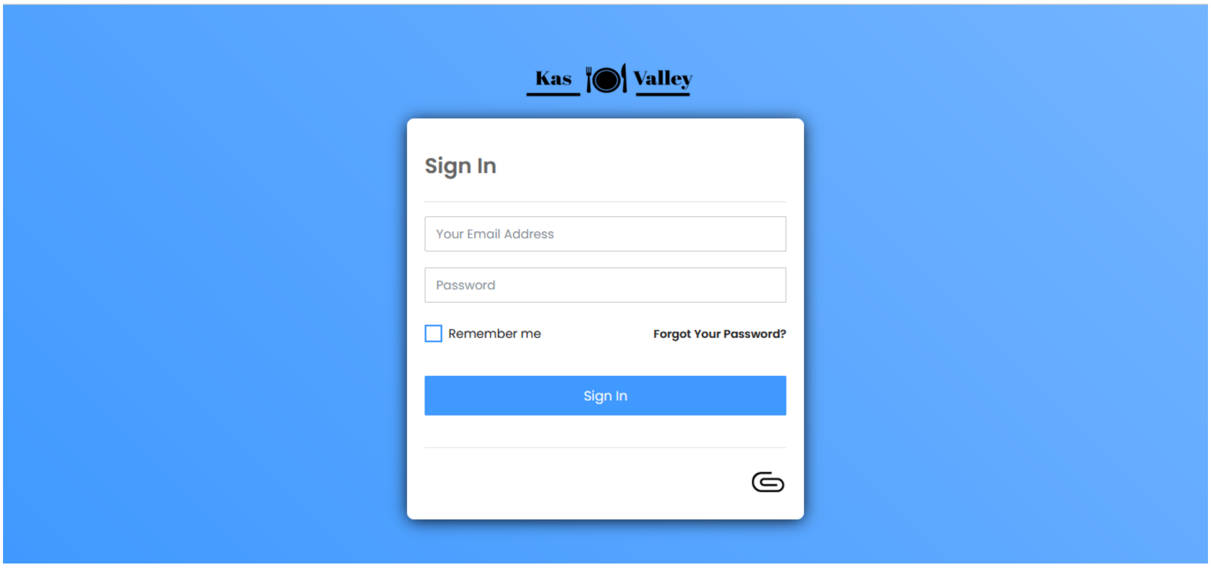
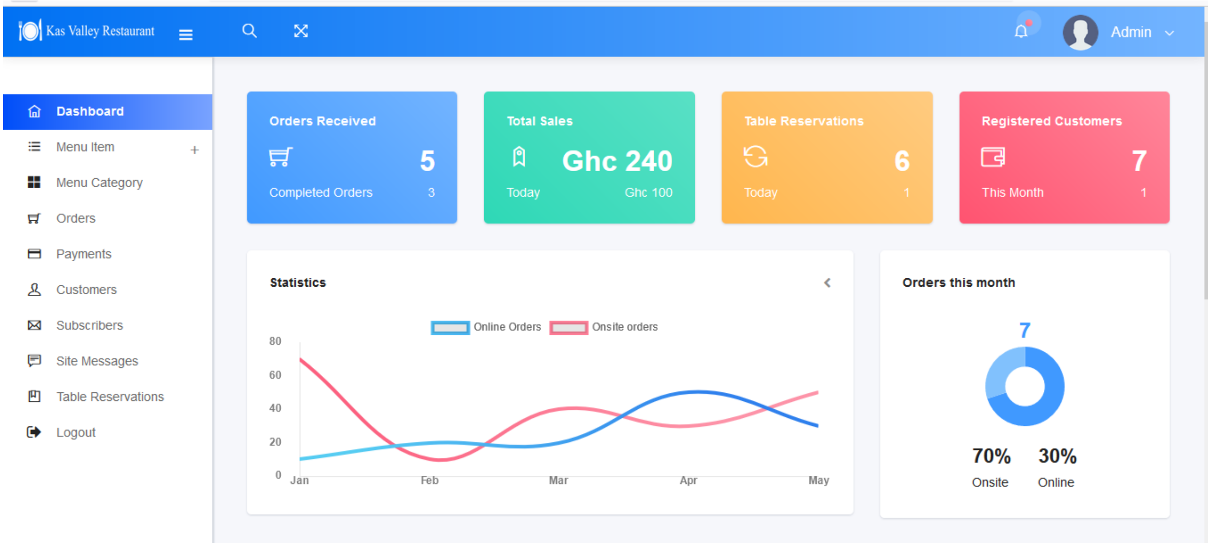
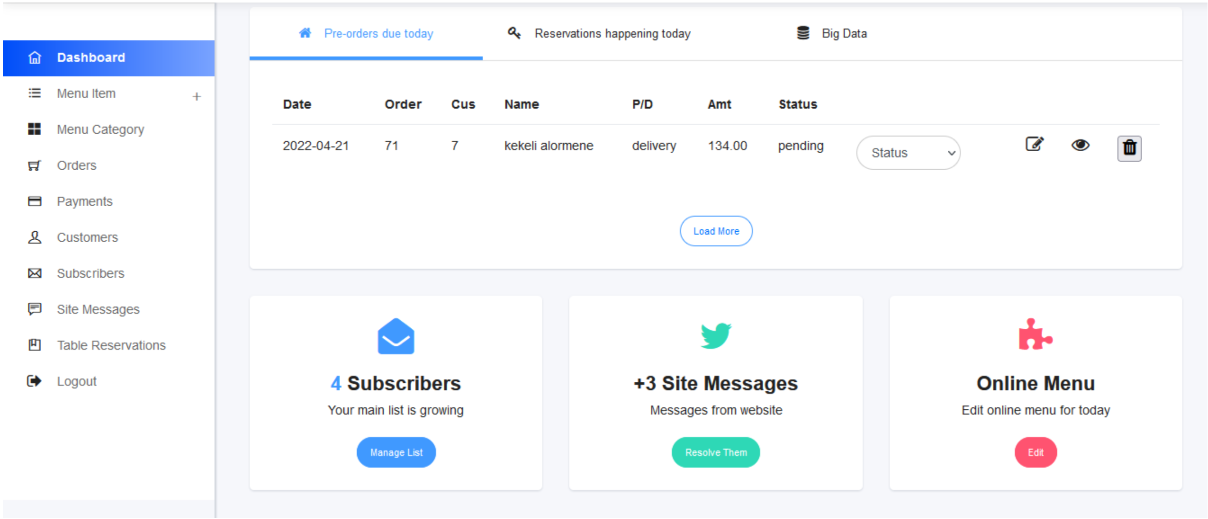


Figure 4.15: Admin login page

2. Dashboard page

The dashboard page summarises essential information such as the number of orders, reservations and payments. It also shows all orders and reservations scheduled for the given day.





Date	Order	Cus	Name	P/D	Amt	Status	
2022-04-21	71	7	kekeli alormene	delivery	134.00	pending	Status

4 Subscribers
Your main list is growing

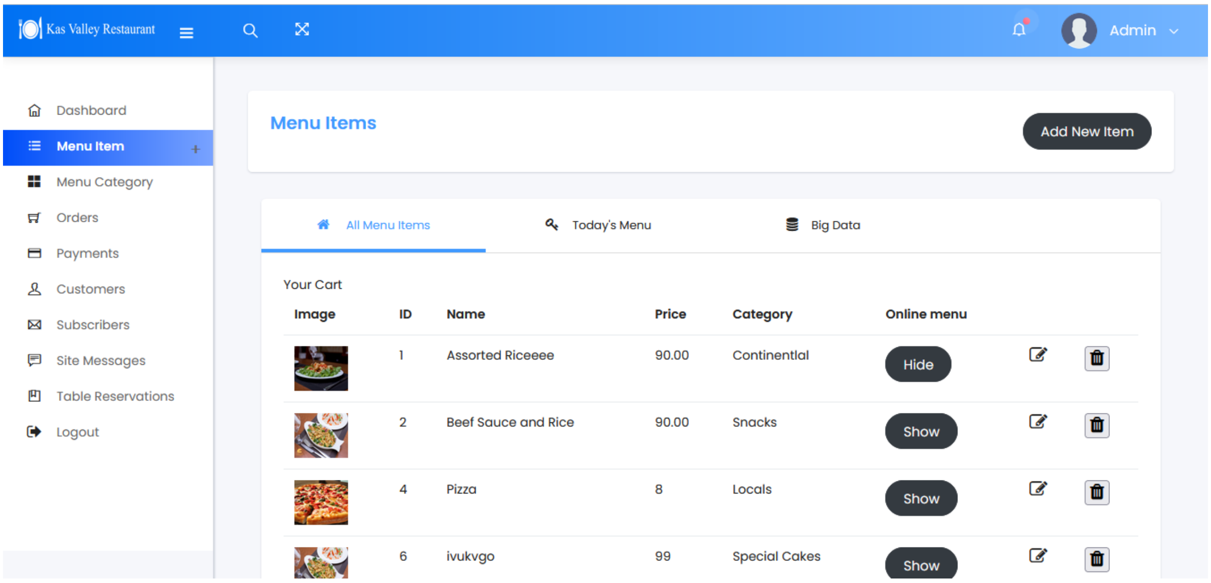
+3 Site Messages
Messages from website

Online Menu
Edit online menu for today

Figure 4.16: Admin dashboard page

3. Menu page

The menu page allows the restaurant to manage its menu. A new item can be added, and an existing item can be modified or deleted. Also, this section enables the restaurant to select the menu items that they wish to appear on their site for customers to shop. There is also a sub-navigation panel with “Today’s menu” and “Big Data”. “Today’s menu” displays all menu items that have been selected to be displayed on the ordering website. “Big Data” gives some analytics on the menu items, including a bar graph showing purchases of menu items over a month.



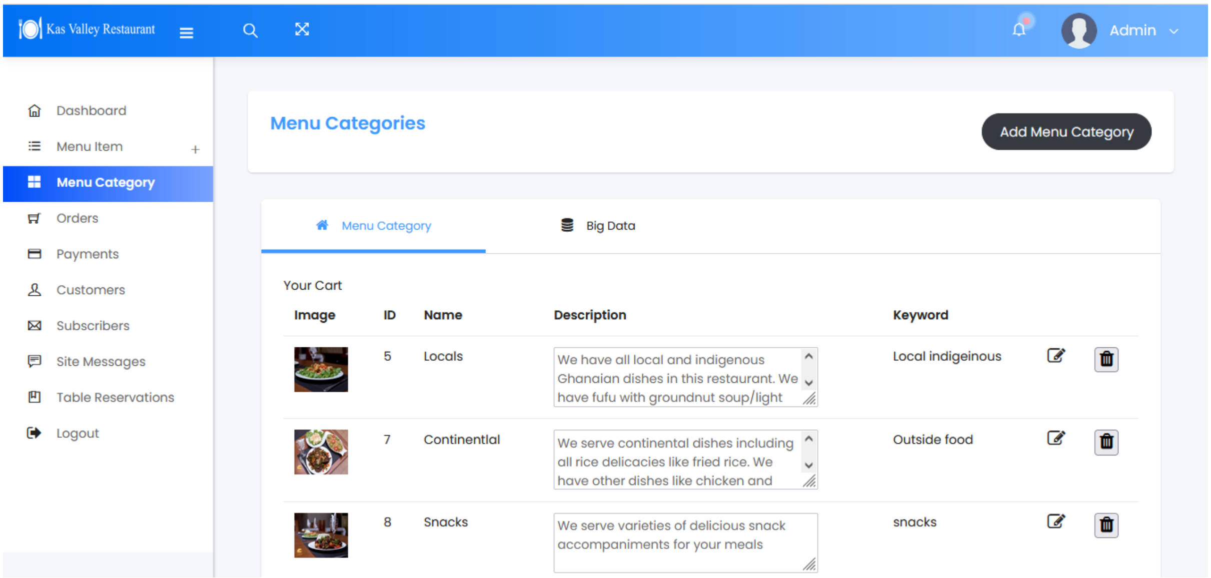
Your Cart

Image	ID	Name	Price	Category	Online menu		
	1	Assorted Riceeee	90.00	Continentlal	Hide		
	2	Beef Sauce and Rice	90.00	Snacks	Show		
	4	Pizza	8	Locals	Show		
	6	ivukvgo	99	Special Cakes	Show		

Figure 4.17: Manage menu page

4. Category Section

This page allows the restaurant to manage the categories under which the menu items fall under. It enables the restaurant to add a new category, edit or delete an existing one.



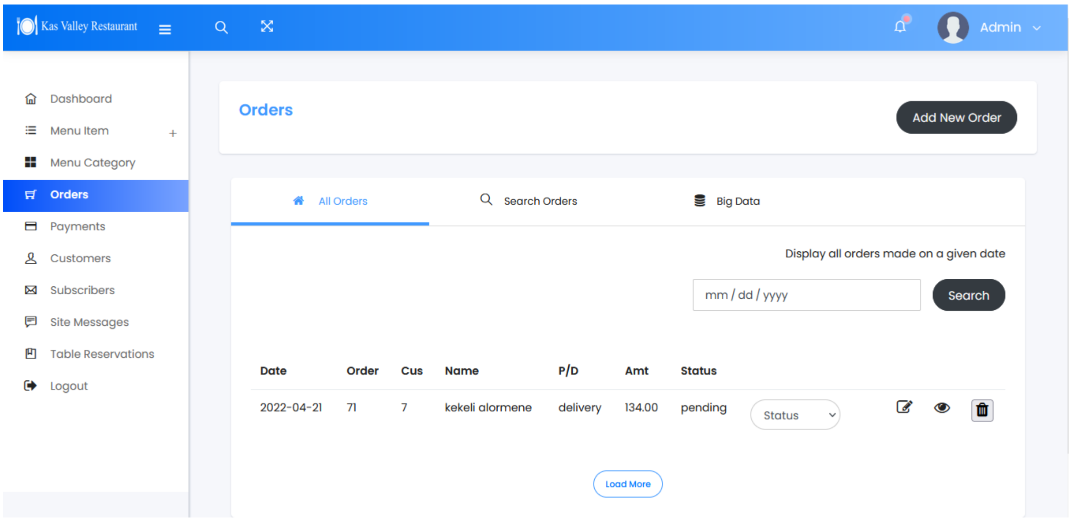
Your Cart

Image	ID	Name	Description	Keyword		
	5	Locals	We have all local and indigenous Ghanaian dishes in this restaurant. We have fufu with groundnut soup/light	Local indigeinous		
	7	Continentlal	We serve continental dishes including all rice delicacies like fried rice. We have other dishes like chicken and	Outside food		
	8	Snacks	We serve varieties of delicious snack accompaniments for your meals	snacks		

Figure 4.18: Manage menu categories page

5. Orders Page

This section enables the restaurant to manage all online and in-person orders. The admin can view all orders made online and change the order status, which would be reflected at the customer's end. When the status is changed to "Completed", the customer is notified via SMS using the Twilio API. The admin can also add a new order, view, and update or delete existing ones. There is also a search feature to enable the admin to search for all orders relating to a particular date. The "Big Data" section displays the frequently ordered menu items using Chart.js



Display all orders made on a given date

mm / dd / yyyy

Date	Order	Cus	Name	P/D	Amt	Status	
2022-04-21	71	7	kekeli alormene	delivery	134.00	pending	Status

Figure 4.19: Manage orders page

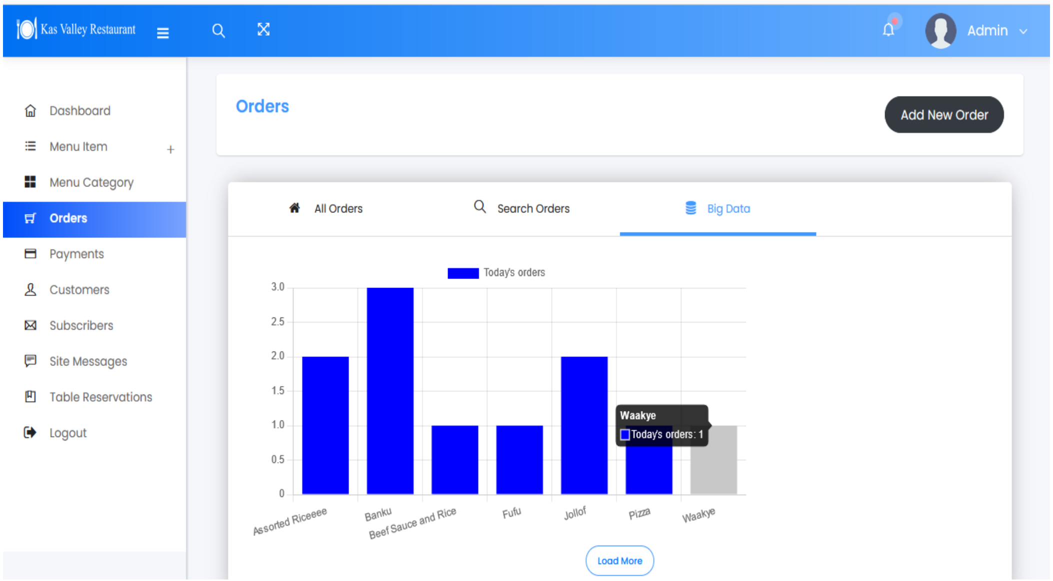
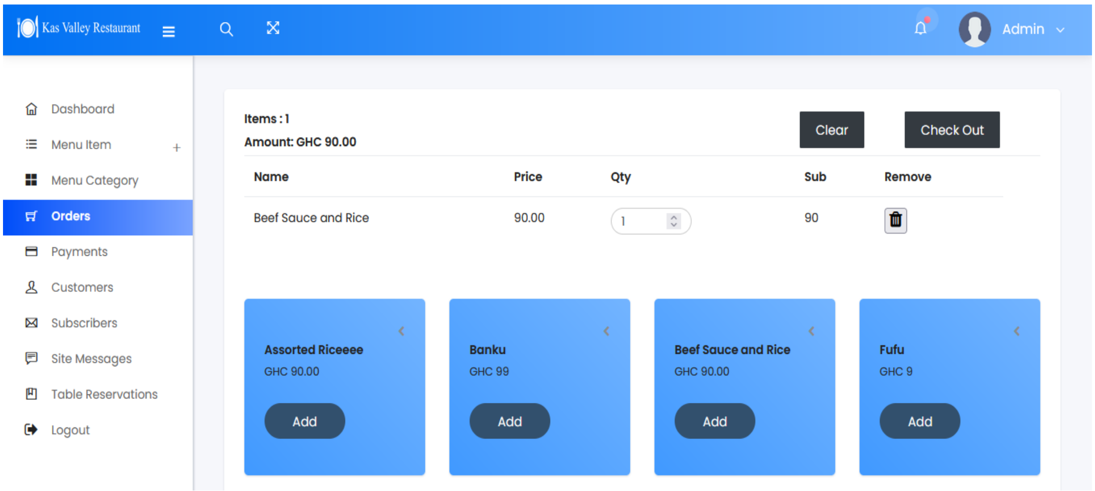


Figure 4.20: Graph showing association between menu items and daily orders

6. Onsite Order Page

This page enables the restaurant to take in new orders from in-person customers. The admin can add a menu item, remove a menu item, update the quantity or clear the customer's order entirely.



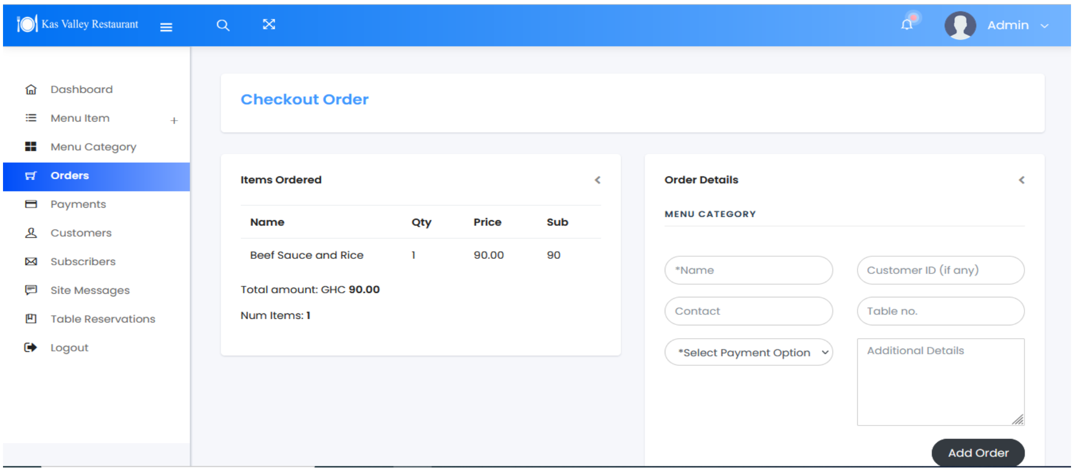
Items : 1
Amount: GHC 90.00

Name	Price	Qty	Sub	Remove
Beef Sauce and Rice	90.00	1	90	🗑

Figure 4.21: Taking orders on-site via the application

7. Checkout Order Page

It enables the restaurant to check out an order entered into the system by the restaurant .



Items Ordered

Name	Qty	Price	Sub
Beef Sauce and Rice	1	90.00	90

Total amount: GHC **90.00**
Num Items: **1**

Order Details

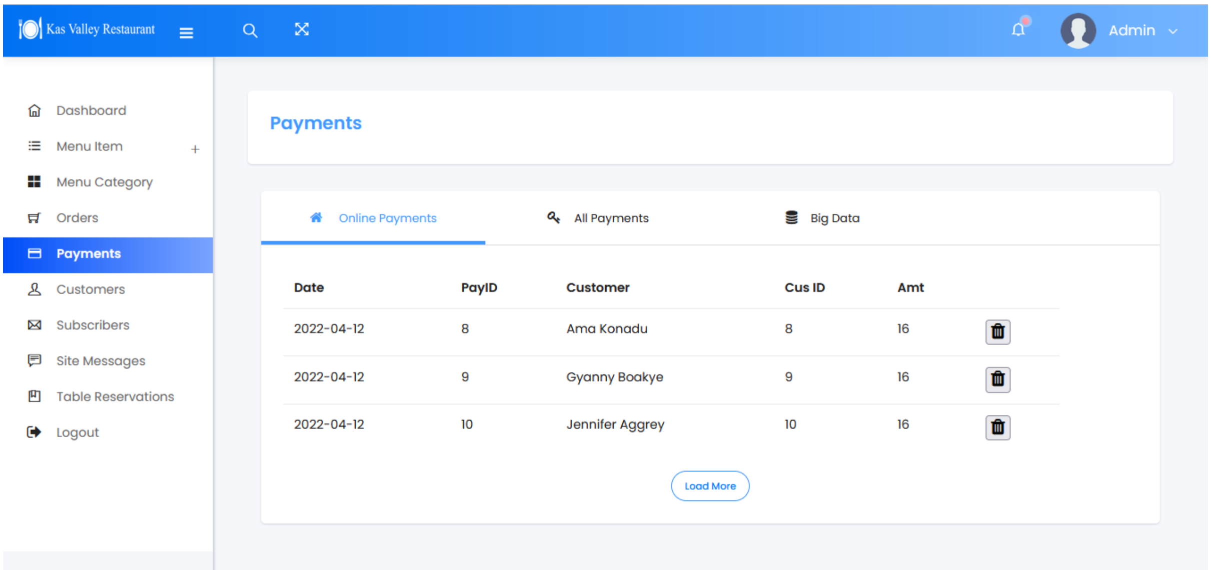
MENU CATEGORY

*Name
Customer ID (if any)
Contact
Table no.
*Select Payment Option
Additional Details

Figure 4.22: Checking out a customer's order entered by the restaurant

8. Payments

All payments in the system are displayed on this page. There is also the option to delete a payment from the system. The Big Data section on the page shows a graph comparing total payments made over the months.

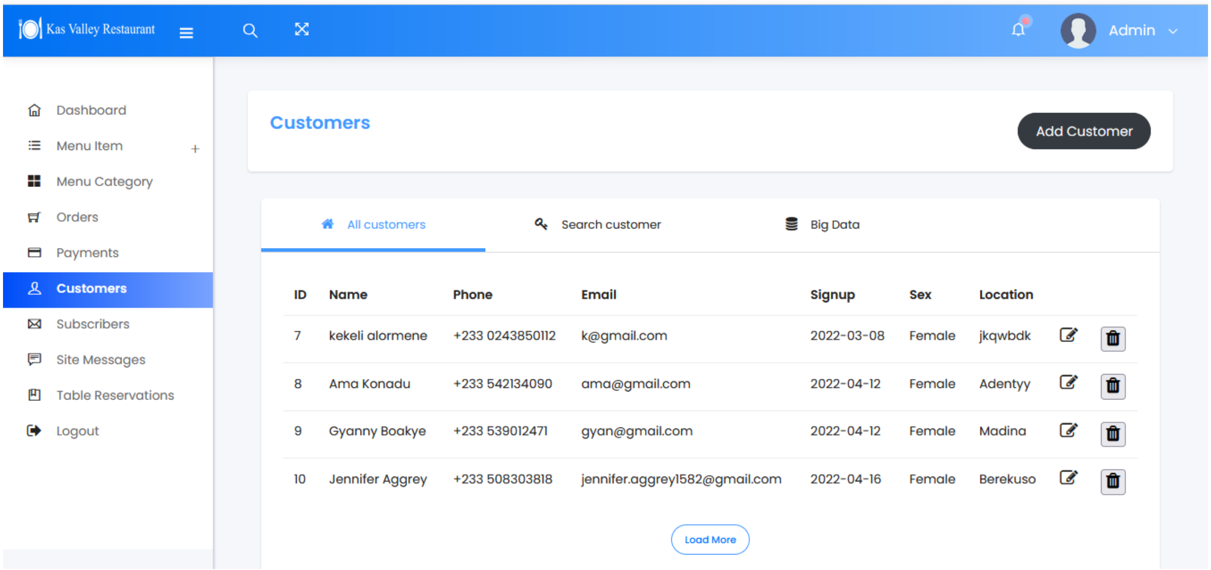


Date	PayID	Customer	Cus ID	Amt
2022-04-12	8	Ama Konadu	8	16
2022-04-12	9	Gyanny Boakye	9	16
2022-04-12	10	Jennifer Aggrey	10	16

Figure 4.23: Manage customers' payments page

9. Customer Section

This section allows the admin to view and manage customer data. Here, the admin can add a new customer and update or delete an existing one. The admin can also view a graph showing customer growth over the months. Additionally, there is a search feature to search for a particular customer.

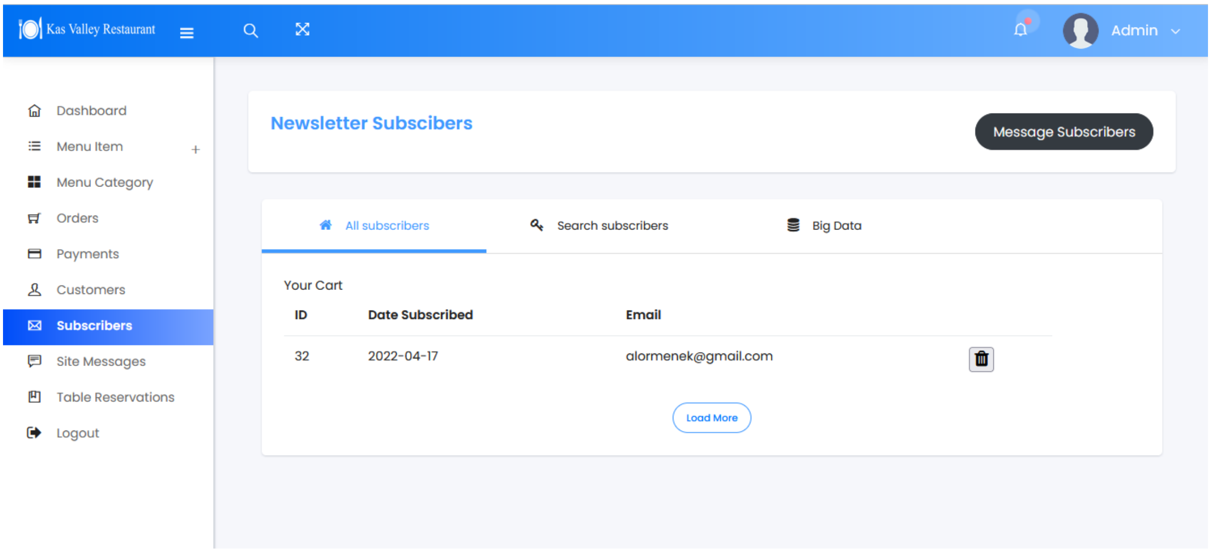


ID	Name	Phone	Email	Signup	Sex	Location
7	kekeli alormene	+233 0243850112	k@gmail.com	2022-03-08	Female	jkqwbdk
8	Ama Konadu	+233 542134090	ama@gmail.com	2022-04-12	Female	Adentyy
9	Gyanny Boakye	+233 539012471	gyan@gmail.com	2022-04-12	Female	Madina
10	Jennifer Aggrey	+233 508303818	jennifer.aggrey1582@gmail.com	2022-04-16	Female	Berekuso

Figure 4.24: Manage customer data page

10. Subscribers

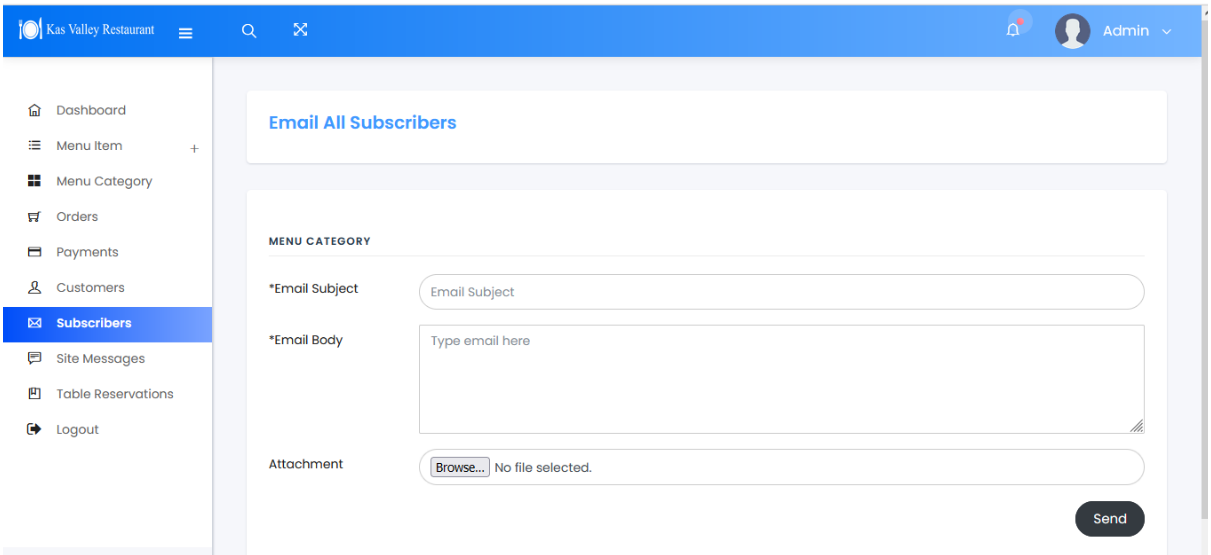
This section enables the admin to manage the restaurant’s subscribers. Here the admin can view all subscribers who have subscribed and delete a subscriber. Additionally, this section allows the admin to email all subscribers with the help of the PHPMailer library.



Your Cart

ID	Date Subscribed	Email
32	2022-04-17	alormenek@gmail.com

Figure 4.25: Manage subscribers’ page



Email All Subscribers

MENU CATEGORY

*Email Subject: Email Subject

*Email Body: Type email here

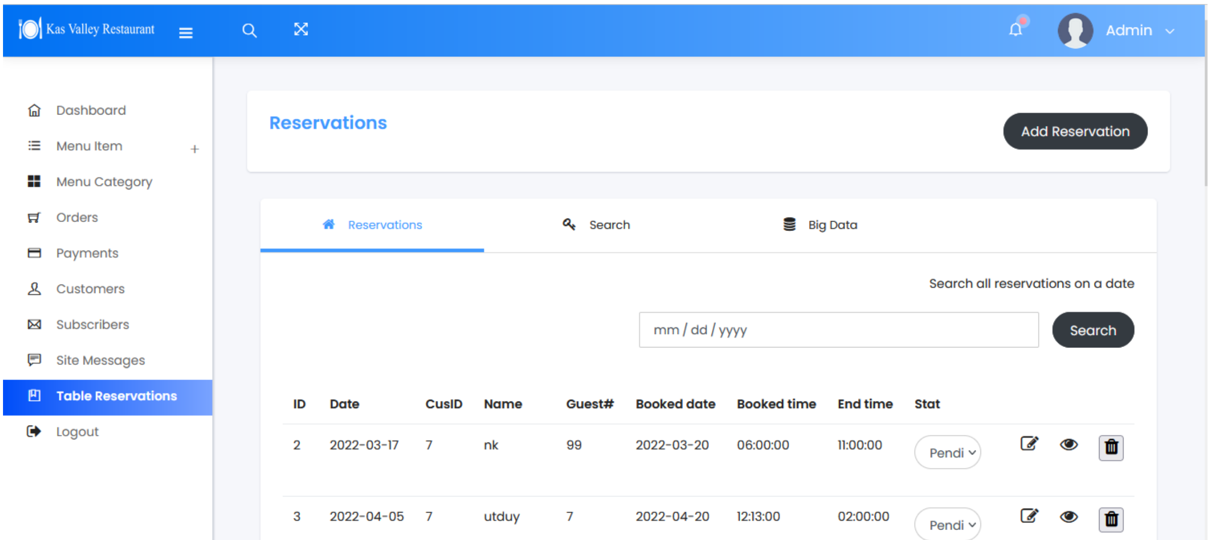
Attachment: Browse... No file selected.

Send

Figure 4.26: Email all subscribers' page

11. Table Reservation Section

It displays all reservations, that is, reservations made online and reservations entered by the admin. It also allows the admin to add a new reservation, edit, delete, search reservations and change the status of a reservation. Additionally, this section has an algorithm to check if a selected date and time for a reservation is fully booked or not before entering the reservation into the system. There is also a chart comparing reservations made across different months.



Reservations

Add Reservation

Search all reservations on a date

mm / dd / yyyy

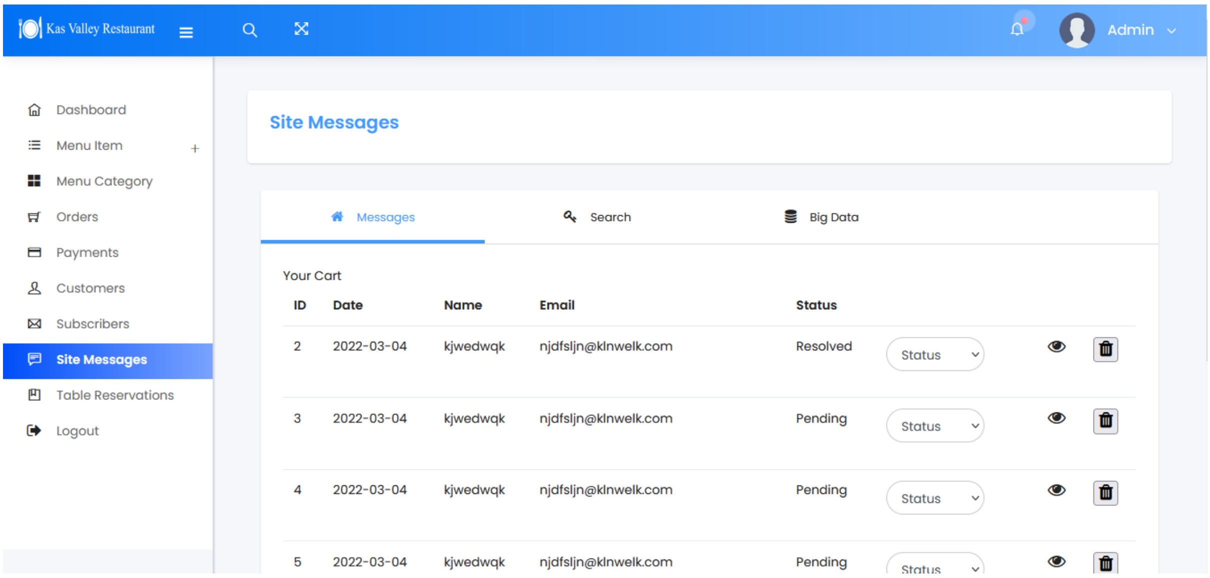
Search

ID	Date	CusID	Name	Guest#	Booked date	Booked time	End time	Stat
2	2022-03-17	7	nk	99	2022-03-20	06:00:00	11:00:00	Pendi
3	2022-04-05	7	utduy	7	2022-04-20	12:13:00	02:00:00	Pendi

Figure 4.27: Manage table reservations page

12. Site Messages

This section displays all the messages that users send via the website. It also allows the admin to view and delete a message. The admin can also mark a message as resolved by changing the status of the message. There is also a chart showing the ratio of unresolved messages to resolved messages.

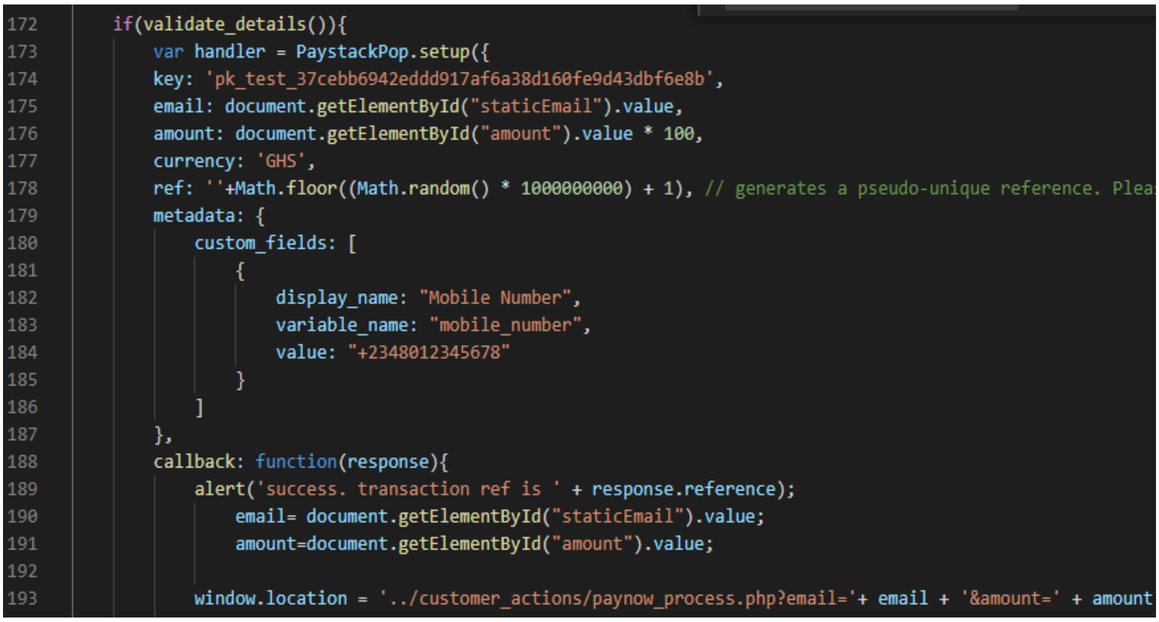


Your Cart

ID	Date	Name	Email	Status			
2	2022-03-04	kjwedwqk	njdfsljn@klnwelk.com	Resolved	Status	👁	🗑
3	2022-03-04	kjwedwqk	njdfsljn@klnwelk.com	Pending	Status	👁	🗑
4	2022-03-04	kjwedwqk	njdfsljn@klnwelk.com	Pending	Status	👁	🗑
5	2022-03-04	kjwedwqk	njdfsljn@klnwelk.com	Pending	Status	👁	🗑

Figure 4.28: Manage site messages page

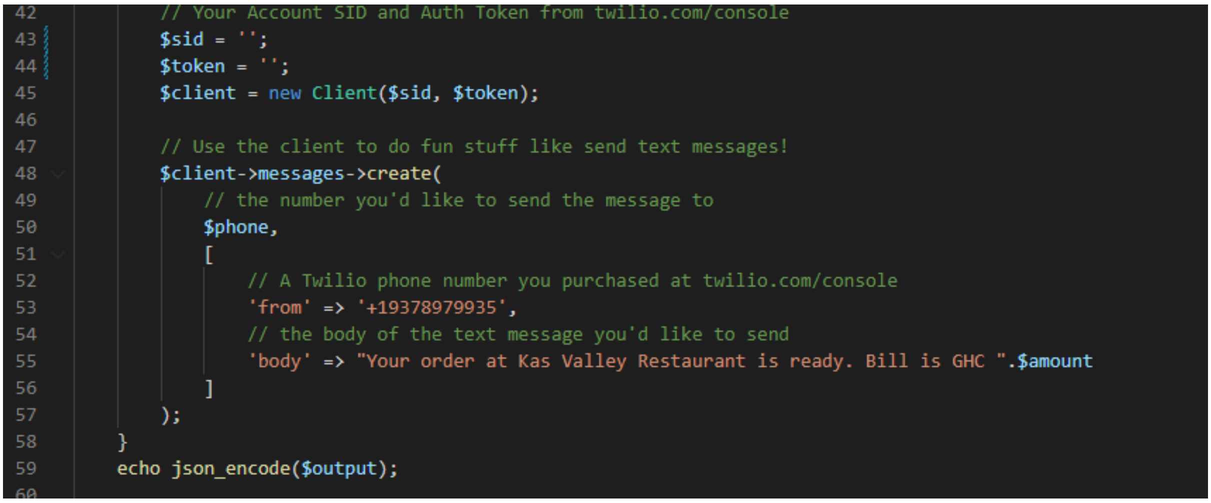
4.4.3 Code Snippets



```
    if(validate_details()){
        var handler = PaystackPop.setup({
        key: 'pk_test_37cebb6942eddd917af6a38d160fe9d43dbf6e8b',
        email: document.getElementById("staticEmail").value,
        amount: document.getElementById("amount").value * 100,
        currency: 'GHS',
        ref: ''+Math.floor((Math.random() * 1000000000) + 1), // generates a pseudo-unique reference. Plea
        metadata: {
            custom_fields: [
                {
                    display_name: "Mobile Number",
                    variable_name: "mobile_number",
                    value: "+2348012345678"
                }
            ]
        },
        callback: function(response){
            alert('success. transaction ref is ' + response.reference);
                email= document.getElementById("staticEmail").value;
                amount=document.getElementById("amount").value;

            window.location = '../customer_actions/paynow_process.php?email='+ email + '&amount=' + amount
```

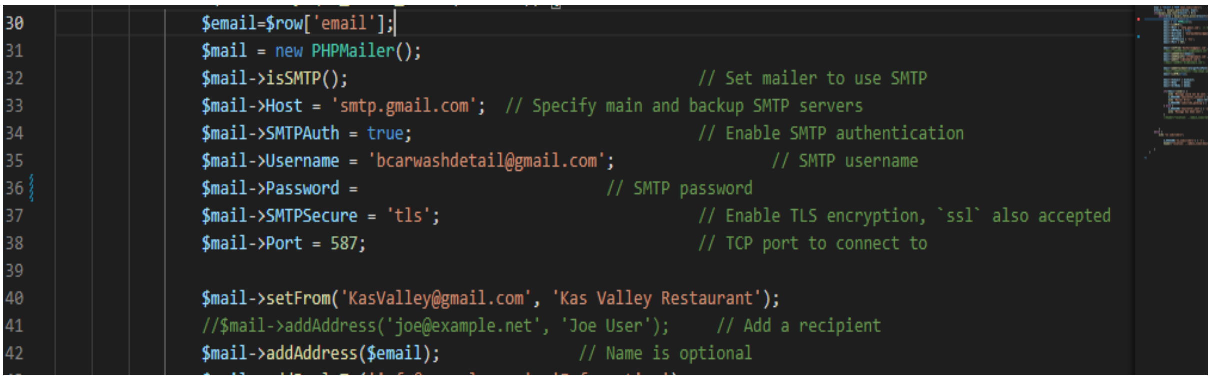
Figure 4.29: Code snippet showing Paystack API in use



```
        // Your Account SID and Auth Token from twilio.com/console
        $sid = '';
        $token = '';
        $client = new Client($sid, $token);

        // Use the client to do fun stuff like send text messages!
        $client->messages->create(
            // the number you'd like to send the message to
            $phone,
            [
                // A Twilio phone number you purchased at twilio.com/console
                'from' => '+19378979935',
                // the body of the text message you'd like to send
                'body' => "Your order at Kas Valley Restaurant is ready. Bill is GHC ".$amount
            ]
        );
    }
    echo json_encode($output);
```

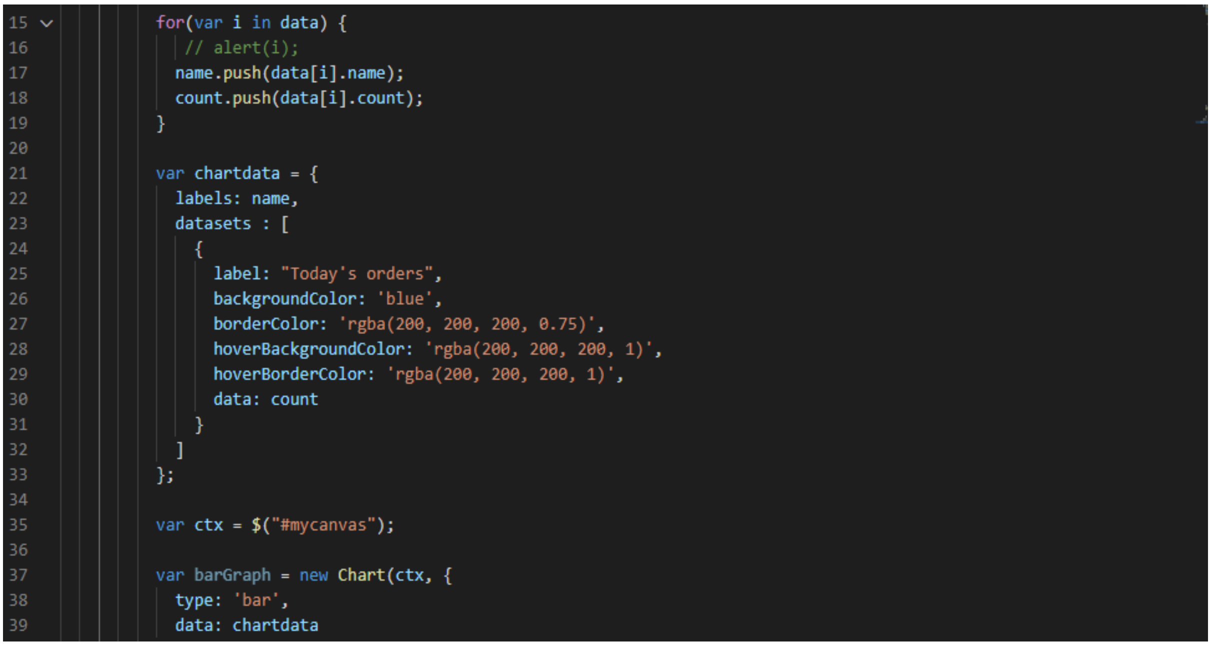
Figure 4.30: Code snippet showing Twilio API in use



```php
$email=$row['email'];
$mail = new PHPMailer();
$mail->isSMTP();                                    // Set mailer to use SMTP
$mail->Host = 'smtp.gmail.com';  // Specify main and backup SMTP servers
$mail->SMTPAuth = true;                             // Enable SMTP authentication
$mail->Username = 'bcarwashdetail@gmail.com';                // SMTP username
$mail->Password =                         // SMTP password
$mail->SMTPSecure = 'tls';                          // Enable TLS encryption, `ssl` also accepted
$mail->Port = 587;                                  // TCP port to connect to

$mail->setFrom('KasValley@gmail.com', 'Kas Valley Restaurant');
//$mail->addAddress('joe@example.net', 'Joe User');     // Add a recipient
$mail->addAddress($email);                // Name is optional
```

Figure 4.31: Code snippet showing PHPMailer library in use



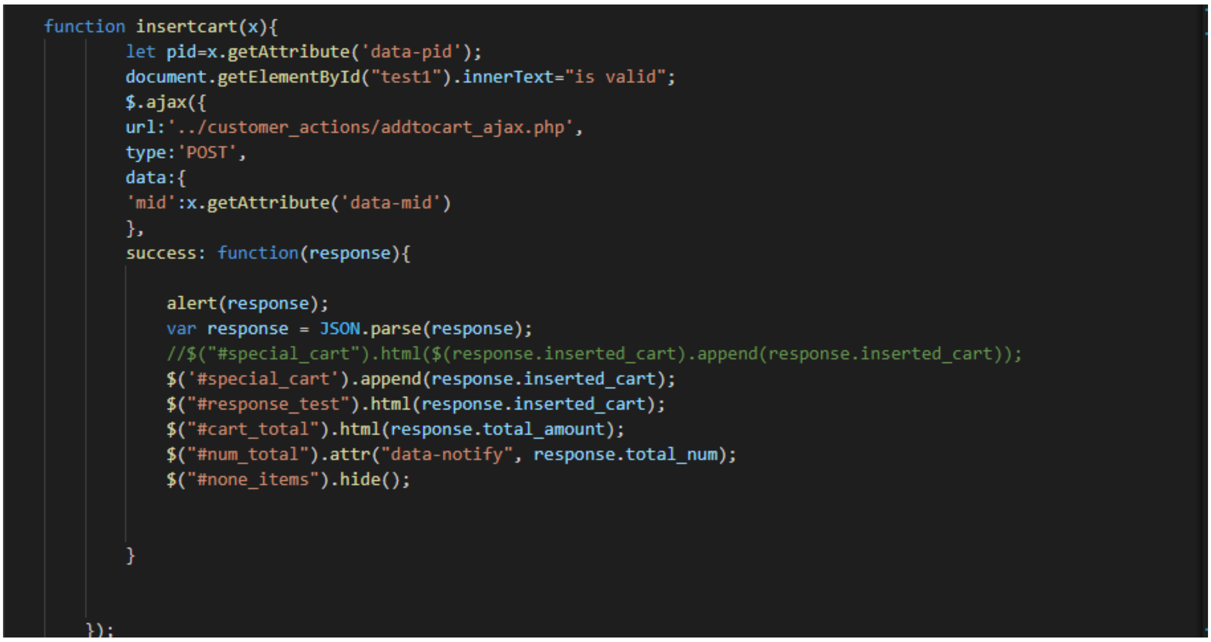
```javascript
for(var i in data) {
  // alert(i);
  name.push(data[i].name);
  count.push(data[i].count);
}

var chartdata = {
  labels: name,
  datasets : [
    {
      label: "Today's orders",
      backgroundColor: 'blue',
      borderColor: 'rgba(200, 200, 200, 0.75)',
      hoverBackgroundColor: 'rgba(200, 200, 200, 1)',
      hoverBorderColor: 'rgba(200, 200, 200, 1)',
      data: count
    }
  ]
};

var ctx = $("#mycanvas");

var barGraph = new Chart(ctx, {
  type: 'bar',
  data: chartdata
```

Figure 4.32: Code snippet showing chart.js in use



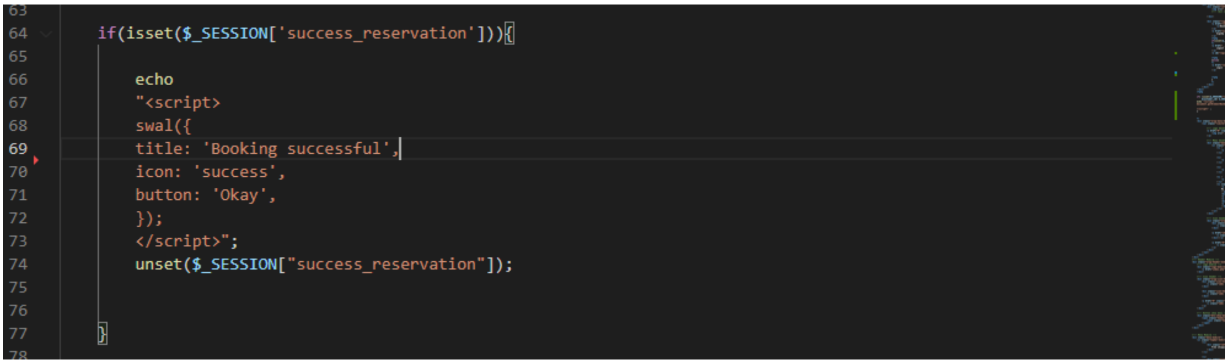
```
function insertcart(x){
        let pid=x.getAttribute('data-pid');
        document.getElementById("test1").innerText="is valid";
        $.ajax({
        url:'../customer_actions/addtocart_ajax.php',
        type:'POST',
        data:{
        'mid':x.getAttribute('data-mid')
        },
        success: function(response){

            alert(response);
            var response = JSON.parse(response);
            //$("#special_cart").html($(response.inserted_cart).append(response.inserted_cart));
            $('#special_cart').append(response.inserted_cart);
            $("#response_test").html(response.inserted_cart);
            $("#cart_total").html(response.total_amount);
            $("#num_total").attr("data-notify", response.total_num);
            $("#none_items").hide();


        }


    });
```

Figure 4.33: Code snippet showing Ajax, JQuery and JSON in use



```
    if(isset($_SESSION['success_reservation'])){

        echo
        "<script>
        swal({
        title: 'Booking successful',
        icon: 'success',
        button: 'Okay',
        });
        </script>";
        unset($_SESSION["success_reservation"]);


    }
```

Figure 4.34: Code snippet showing SweetAlert in use

4.5 Conclusion

This chapter discussed the tools and technologies used to implement the web-based RMS. The various user interfaces for both the customer and admin (restaurant) sides were also outlined in this chapter, alongside their implementation details. Additionally, the chapter contains code snippets of some essential aspects of the system.

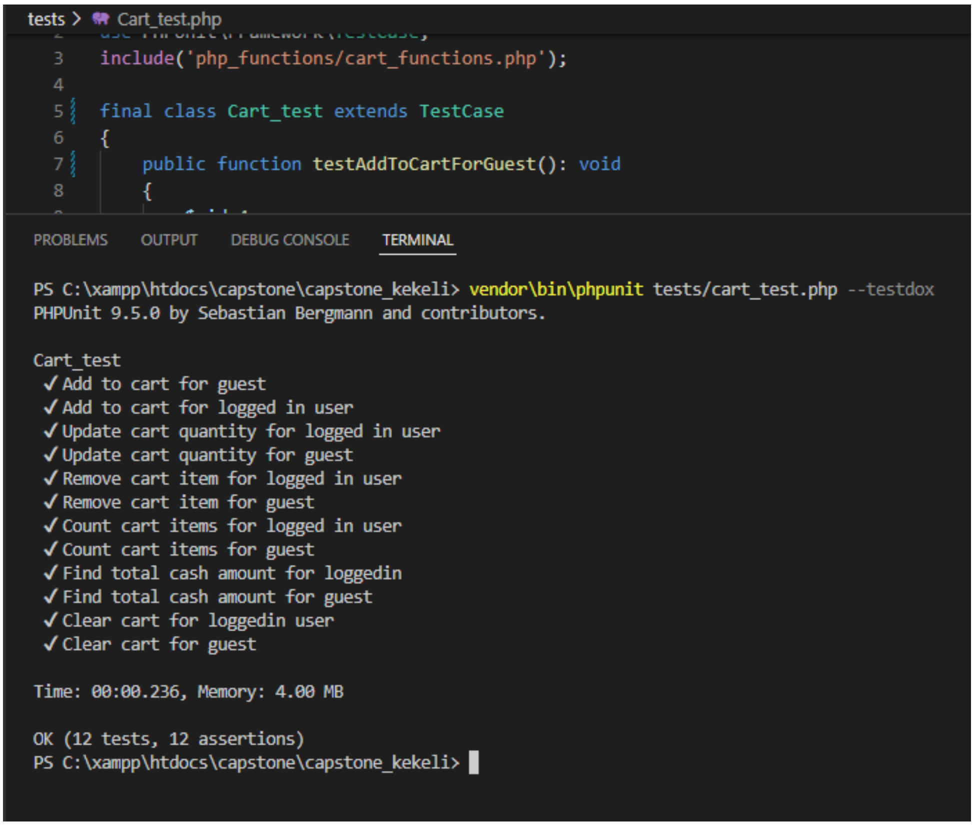
Chapter 5: Testing

5.1 Overview

This chapter covers some testing techniques used to evaluate the developed system. The following are the testing techniques adopted for this system: unit testing, compatibility testing, system testing and user acceptance testing.

5.2. Unit Testing

The various components that make up the system are tested separately in unit testing. In this project, the PHP functions in the system were tested individually using PHPUnit testing. A PHPUnit test case was written for each function. All the test cases for the PHP functions passed. Find below screenshots of the results obtained from the unit testing. A tick means the test carried out passed successfully. The brief statement after each tick is the description of the test or the tested function.



```php
use PHPUnit\Framework\TestCase;
include('php_functions/cart_functions.php');

final class Cart_test extends TestCase
{
    public function testAddToCartForGuest(): void
    {
```

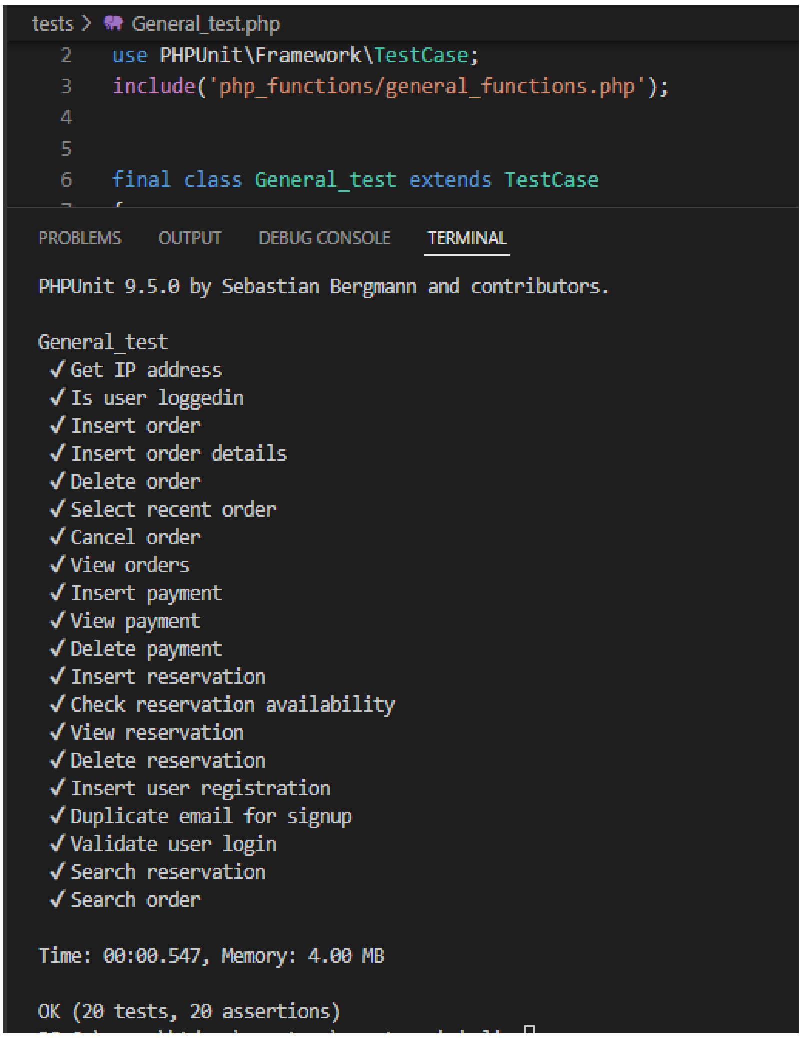
```
PS C:\xampp\htdocs\capstone\capstone_kekeli> vendor\bin\phpunit tests/cart_test.php --testdox
PHPUnit 9.5.0 by Sebastian Bergmann and contributors.

Cart_test
 ✓ Add to cart for guest
 ✓ Add to cart for logged in user
 ✓ Update cart quantity for logged in user
 ✓ Update cart quantity for guest
 ✓ Remove cart item for logged in user
 ✓ Remove cart item for guest
 ✓ Count cart items for logged in user
 ✓ Count cart items for guest
 ✓ Find total cash amount for loggedin
 ✓ Find total cash amount for guest
 ✓ Clear cart for loggedin user
 ✓ Clear cart for guest

Time: 00:00.236, Memory: 4.00 MB

OK (12 tests, 12 assertions)
PS C:\xampp\htdocs\capstone\capstone_kekeli>
```

Figure 5.1: Results of PHPUnit testing for cart-related functions



```php
use PHPUnit\Framework\TestCase;
include('php_functions/general_functions.php');


final class General_test extends TestCase
```

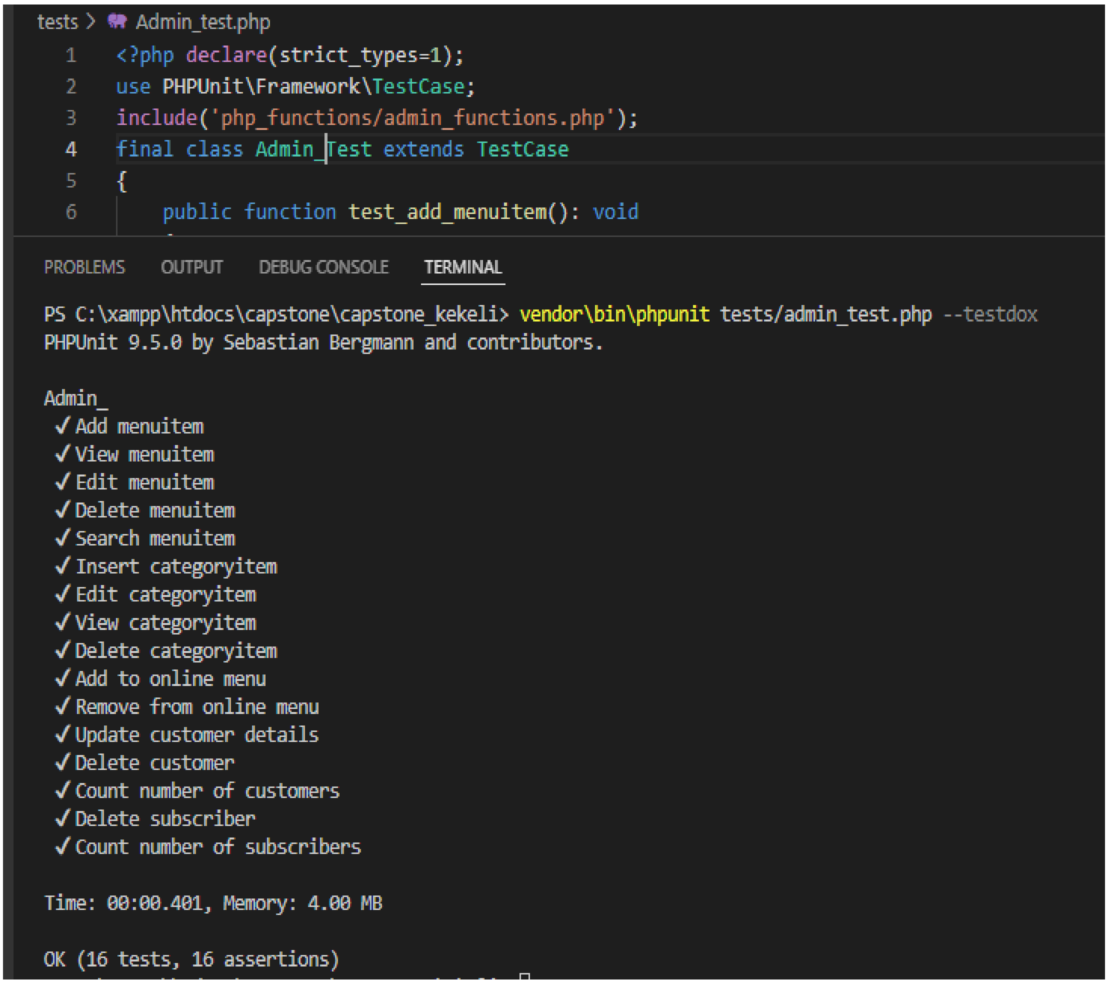
```
PHPUnit 9.5.0 by Sebastian Bergmann and contributors.

General_test
 ✓ Get IP address
 ✓ Is user loggedin
 ✓ Insert order
 ✓ Insert order details
 ✓ Delete order
 ✓ Select recent order
 ✓ Cancel order
 ✓ View orders
 ✓ Insert payment
 ✓ View payment
 ✓ Delete payment
 ✓ Insert reservation
 ✓ Check reservation availability
 ✓ View reservation
 ✓ Delete reservation
 ✓ Insert user registration
 ✓ Duplicate email for signup
 ✓ Validate user login
 ✓ Search reservation
 ✓ Search order

Time: 00:00.547, Memory: 4.00 MB

OK (20 tests, 20 assertions)
```

Figure 5.2: Results of PHPUnit testing for functions related to orders and reservations



```php
<?php declare(strict_types=1);
use PHPUnit\Framework\TestCase;
include('php_functions/admin_functions.php');
final class Admin_Test extends TestCase
{
    public function test_add_menuitem(): void
```

```
PS C:\xampp\htdocs\capstone\capstone_kekeli> vendor\bin\phpunit tests/admin_test.php --testdox
PHPUnit 9.5.0 by Sebastian Bergmann and contributors.

Admin_
 ✓ Add menuitem
 ✓ View menuitem
 ✓ Edit menuitem
 ✓ Delete menuitem
 ✓ Search menuitem
 ✓ Insert categoryitem
 ✓ Edit categoryitem
 ✓ View categoryitem
 ✓ Delete categoryitem
 ✓ Add to online menu
 ✓ Remove from online menu
 ✓ Update customer details
 ✓ Delete customer
 ✓ Count number of customers
 ✓ Delete subscriber
 ✓ Count number of subscribers

Time: 00:00.401, Memory: 4.00 MB

OK (16 tests, 16 assertions)
```

Figure 5.3: Results of PHPUnit testing for PHP functions relating to admin actions

5.3 System Testing

In system testing, the various components tested separately are integrated and tested as a whole. This ensures that these components can function as expected to keep the system running smoothly. System testing verifies that the final product meets the specifications in the requirement document. In performing system testing in this project, various scenarios are used to describe the system's interactive ability with users. The system is then tested against these scenarios.

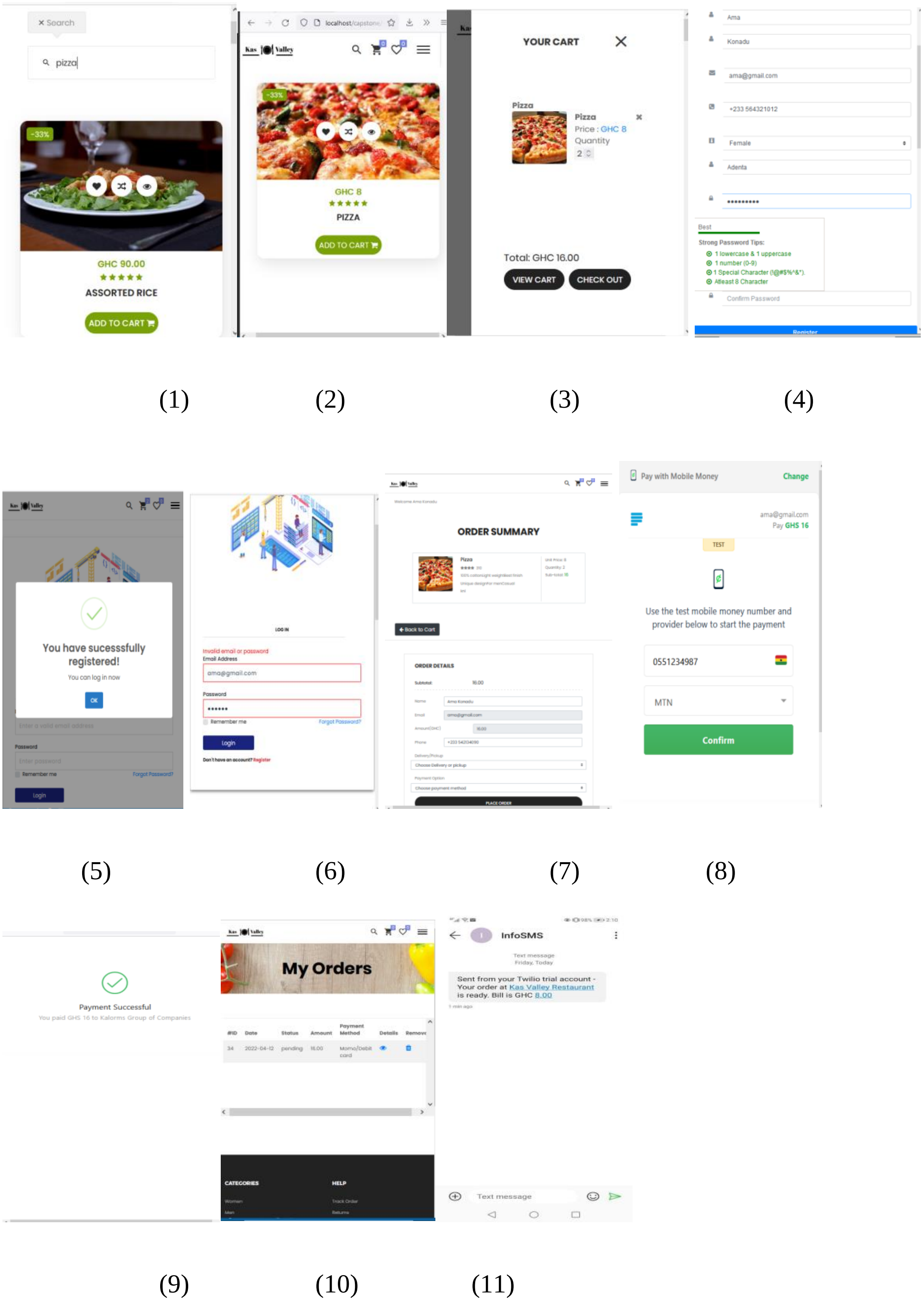
5.3.1 Scenario 1.0

Ama, a student at the University of Legon, wishes to buy Pizza from Kas Valley Restaurant. However, Ama is exhausted and cannot travel to the restaurant to purchase her food. Ama wants to order remotely, make payments via mobile money, and check her order status after purchase. Ama remembers a staff at Kas Valley who spoke to her about a newly launched web application where she could place all orders.

This scenario passed the system testing and addressed functional requirements FR01, FR02, FR03, FR04, FR05, FR07, and FR09. Also, it addressed non-functional requirements NFR1, NFR2, NFR3, NFR5, and NFR6. Below is a flow of the interaction between Ama and the system for this specific scenario. The sequence of interactions has been numbered, and the numbering corresponds to the numbered interfaces in figure 5.4:

1. Ama opens the web application on her phone's browser. She sees all the food items displayed on their site. However, she's struggling to find the pizza on the application, so she uses the app's search feature to find the pizza.
2. She then adds two pieces to her cart and proceeds to checkout.
3. Ama realizes she has been redirected to the login page upon checking out. However, as a first-time user of the system, she does not have an account.
4. Thus, she proceeds to register an account with the restaurant.

5. Upon successfully registering, a success feedback message pops up to inform Ama that she has successfully registered.
6. She is directed to log in and can now successfully check out.
7. She selects the payment option that allows her to pay with mobile money upon checking out.
8. Ama enters her mobile money number to make payment.
9. Ama receives a success message stating that her order has been successfully placed.
10. After that, Ama decides to check the status of her order (pending, processed or dispatched).
11. Ama then receives an SMS notification on her phone stating her order has been dispatched.



Ama
Konadu
ama@gmail.com
+233 564321012
Female
Adenta
•••••••••
Best
Strong Password Tips:
- 1 lowercase & 1 uppercase
- 1 number (0-9)
- 1 Special Character (!@#$%^&*).
- Atleast 8 Character

Confirm Password

(1) (2) (3) (4)

You have sucesssfully registered!
You can log in now

Invalid email or password
Email Address: ama@gmail.com
Password: ••••••

Welcome Ama Konadu

ORDER SUMMARY

Pizza
★★★★ 3/0
100% cottonLight weightBest finish
Unique designFor menCasual
knl

Unit Price: 8
Quantity: 2
Sub-total: 16

ORDER DETAILS

Subtotal: 16.00

Name: Ama Konadu
Email: ama@gmail.com
Amount(GHC): 16.00
Phone: +233 542134090
Delivery/Pickup: Choose Delivery or pickup
Payment Option: Choose payment method

ama@gmail.com
Pay GHS 16

Use the test mobile money number and provider below to start the payment

0551234987
MTN

(5) (6) (7) (8)

Payment Successful
You paid GHS 16 to Kalorms Group of Companies

#ID	Date	Status	Amount	Payment Method	Details	Remove
34	2022-04-12	pending	16.00	Momo/Debit card		

Text message
Friday, Today

Sent from your Twilio trial account - Your order at Kas Valley Restaurant is ready. Bill is GHC 8.00

1 min ago

(9) (10) (11)

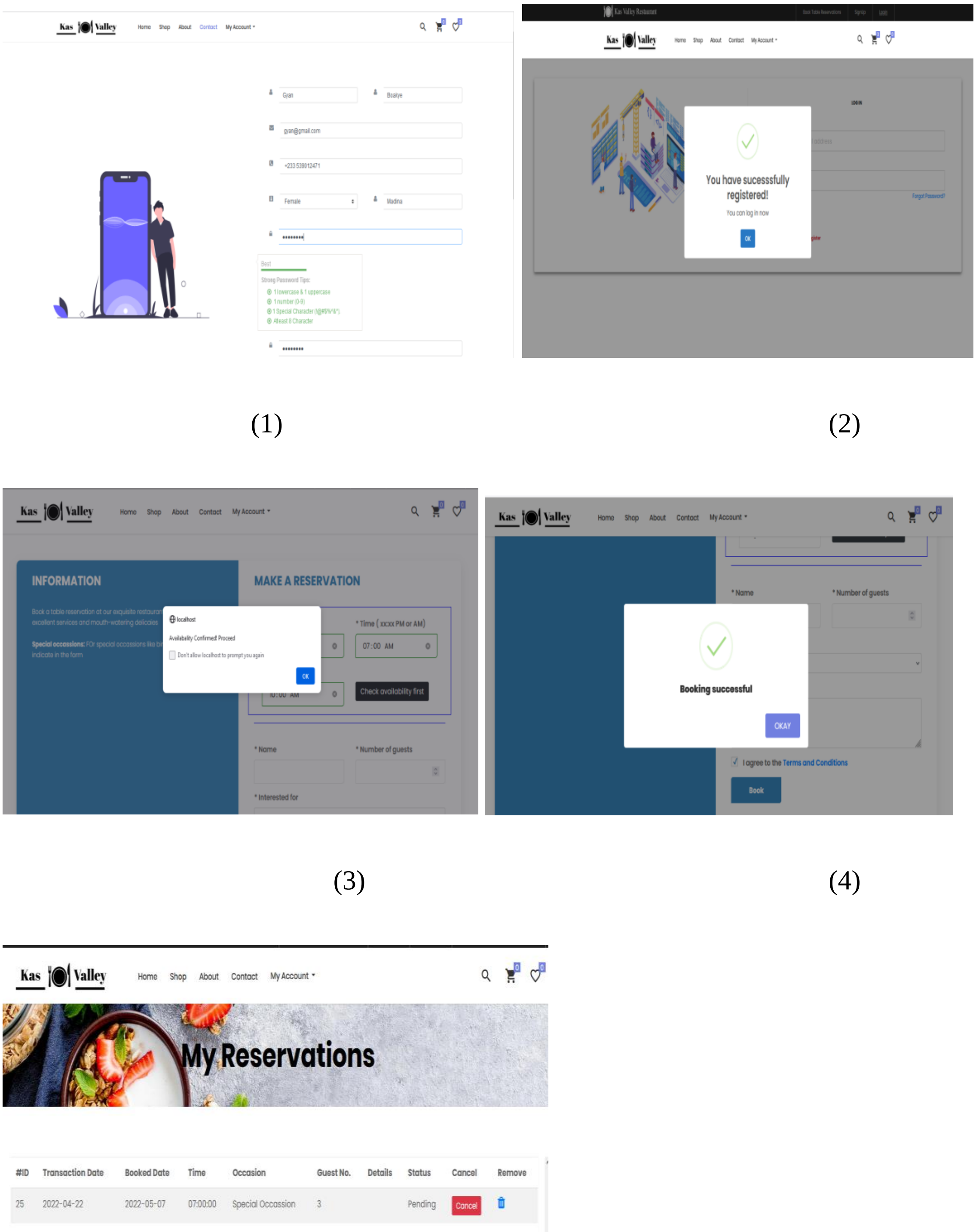
Figure 5.4: System-level testing for the food ordering process

5.3.2. Scenario 2.0

Mr. Gyan, a businessman, wants to plan a little get together with some of his old colleagues from university. He is a regular customer at Kas Valley Restaurant, and he wishes to have the get-together there. However, there was a time when he booked a table reservation at the restaurant via a phone call, but they failed to capture some details of his request. Thus, he wants to secure the reservation so that there will be minimal human errors.

This scenario passed the system testing and addressed functional requirements FR01, FR05, FR06, FR08 and FR09. Also, it addressed non-functional requirements NFR1, NFR2, NFR3, NFR5, and NFR6. Below is a flow of the interaction between Mr. Gyan and the system for this specific scenario. The sequence of interactions has been numbered, and the numbering corresponds to the numbered interfaces in figure 5.5.

1. Mr. Gyan opens the application on his computer and lands on the homepage. He navigates to the reservation page but is redirected to the login page. As a first-time user, he proceeds to register an account
2. He then receives a success message indicating successful registration, and he can now log in.
3. He is now able to access the reservation page. He enters the details of the time and day of the reservation and checks if that slot is available by clicking on the “check availability” button. He receives a feedback message saying the space is available. Mr. Gyan fills in the rest of the details and submits the form.
4. There is a feedback message to indicate a successful booking
5. Mr. Gyan then checks the status of his reservation.



Gyan

Boakye

gyan@gmail.com

+233 539012471

Female

Madina

••••••••

Best

Strong Password Tips:

- 1 lowercase & 1 uppercase
- 1 number (0-9)
- 1 Special Character (!@#$%^&*).
- Atleast 8 Character

••••••••

(1)

You have sucesssfully registered!

You can log in now

(2)

INFORMATION

Book a table reservation at our exquisite restauran excellent services and mouth-watering delicaies

Special occassions: FOr special occassions like bi indicate in the form

MAKE A RESERVATION

localhost

Availabality Confirmed! Proceed

□ Don't allow localhost to prompt you again

* Time (xx:xx PM or AM)

07:00 AM

10:00 AM

* Name

* Number of guests

* Interested for

(3)

* Name

* Number of guests

Booking successful

☑ I agree to the Terms and Conditions

(4)

My Reservations

#ID	Transaction Date	Booked Date	Time	Occasion	Guest No.	Details	Status	Cancel	Remove
25	2022-04-22	2022-05-07	07:00:00	Special Occassion	3		Pending	Cancel	🗑

(5)

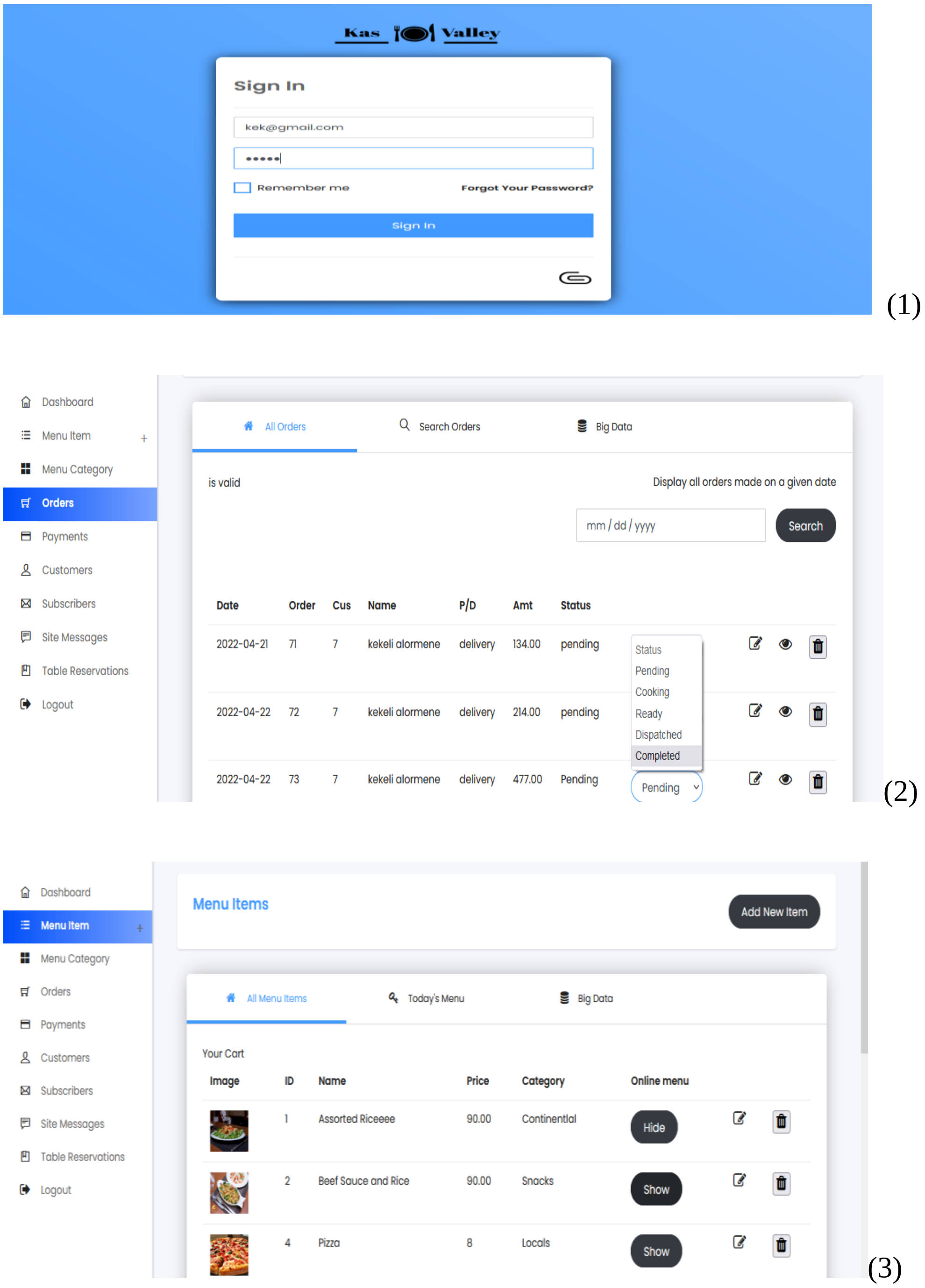
Figure 5.5: System-level testing for the table booking process

5.3.3 Scenario 3.0

Kas Valley Restaurant is receiving an increasing number of orders every day. The business wants to find alternative ways for customers to place their orders and make their reservations to reduce long queues, reduce pressure on staff, and improve customer service. The restaurant wants to receive all these orders at their end in a timely and orderly manner. However, to facilitate ordering using the internet, they need to find a way to get their menu available online and be able to make changes to the menu from their end as well.

This scenario passed the system testing and addressed functional requirements FR011, FR013 and FR014. Also, it addressed non-functional requirements NFR1, NFR2, NFR3, NFR5, and NFR6. Below is a flow of the interaction between the restaurant and the system for this specific scenario. The sequence of interactions has been numbered, and the numbering corresponds to the numbered interfaces in figure 5.6:

1. Firstly, the restaurant admin logs into the system.
2. The restaurant admin can monitor and respond to all orders made by customers online through this system. The admin then chooses to change the order status to let customers know that their orders or reservations have been received and are being attended to.
3. The restaurant admin can manage an online menu by adding, deleting or modifying a menu item.



kek@gmail.com

•••••

is valid

Display all orders made on a given date

mm / dd / yyyy

Date	Order	Cus	Name	P/D	Amt	Status
2022-04-21	71	7	kekeli alormene	delivery	134.00	pending
2022-04-22	72	7	kekeli alormene	delivery	214.00	pending
2022-04-22	73	7	kekeli alormene	delivery	477.00	Pending

Your Cart

Image	ID	Name	Price	Category	Online menu
	1	Assorted Riceeee	90.00	Continential	Hide
	2	Beef Sauce and Rice	90.00	Snacks	Show
	4	Pizza	8	Locals	Show

Figure 5.6: System-level testing for admin managing orders and online menu

5.4 Compatibility Testing

This form of non-functional software test is performed to ensure that a developed system can run on different browsers, operating devices, mobile devices, and hardware.

5.4.1 Browser Compatibility testing

This is a category of compatibility testing to ensure that the system works properly across different browsers. In this project, the application was run on Google Chrome, Firefox, Microsoft Edge and internet explorer to test the browser compatibility. From the results, the system worked as expected on all the browsers tested except for Internet Explorer. In internet explorer, the only issue was that the HTML time and date input field in the table reservation form did not show the popup calendar and time-selector when those input fields were active.

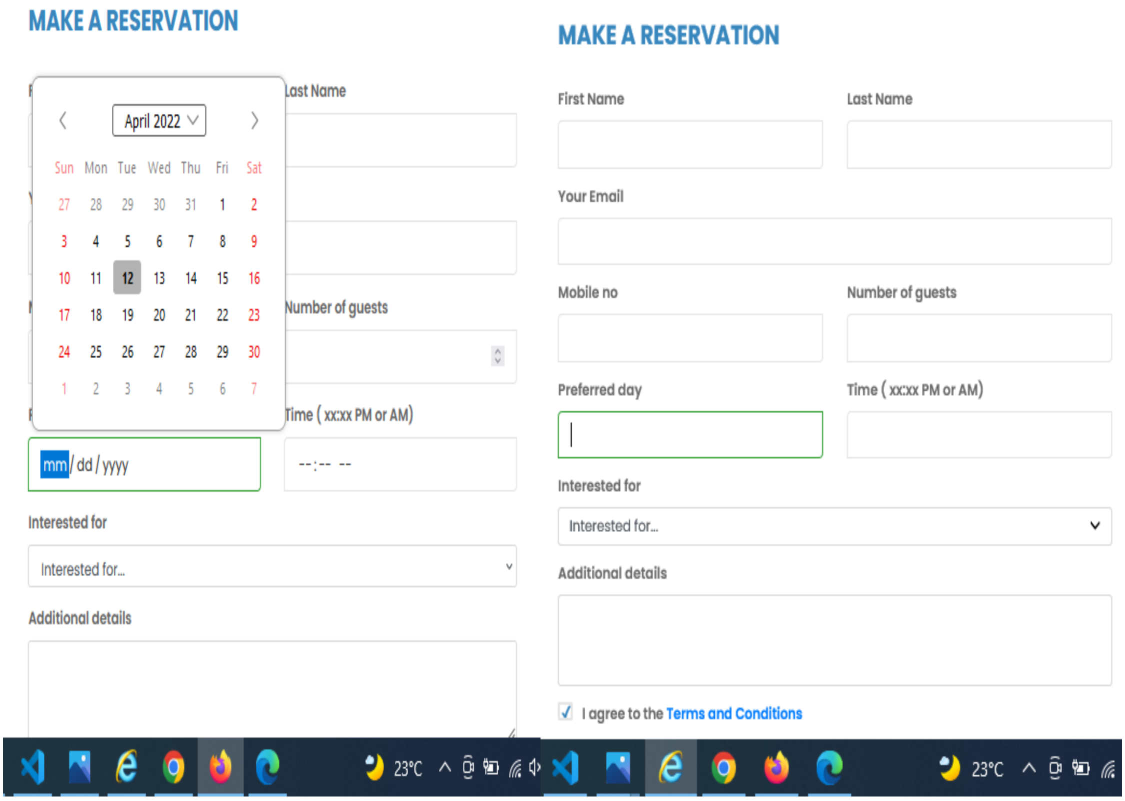


Figure 5.7: Browser compatibility testing

5.4.2 Responsive Testing

This test ensures that the web pages look good and provide relevant information to the user on every device, that is, across viewports of multiple devices. In this project, responsive testing was carried out by viewing the developed application on a desktop pc, a tablet and a smartphone. In simulating this process, a website responsive testing tool was used to check the developed application on different screen resolutions [40]. Upon carrying out the test, the result showed that the system was responsive on all the screen resolutions tested: desktop (1280 x 800 dimension), tablet (768 x 1024 dimension) and mobile phone (360 x 740 dimension). Below are screenshots of the test:

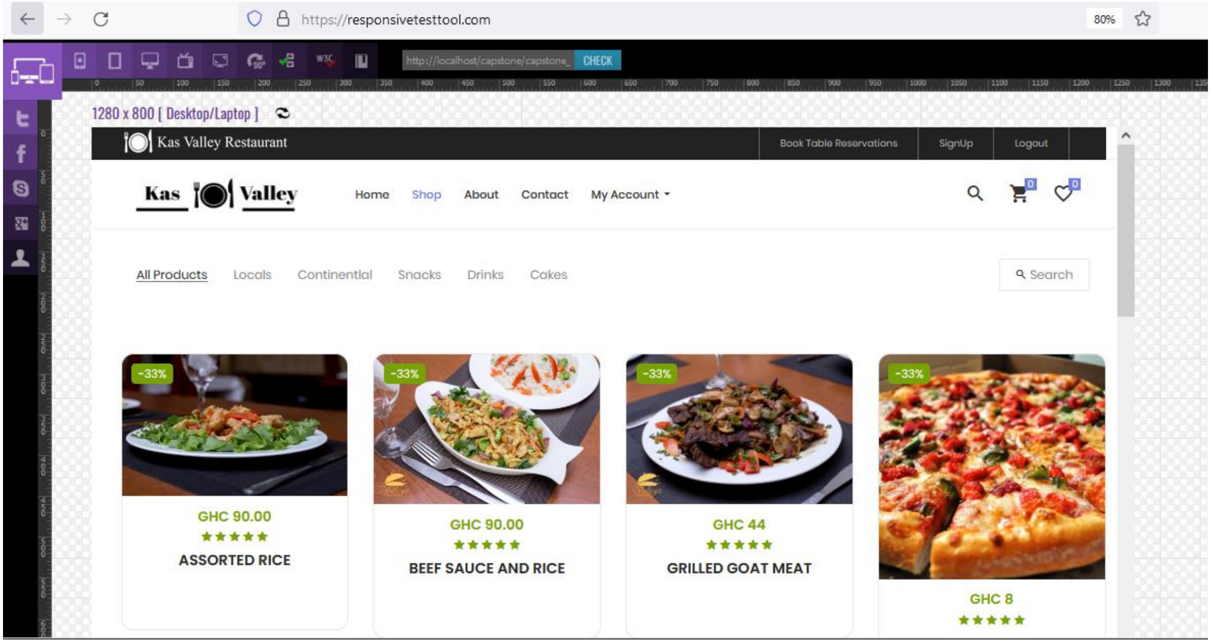


Figure 5.8: Responsiveness test on a desktop

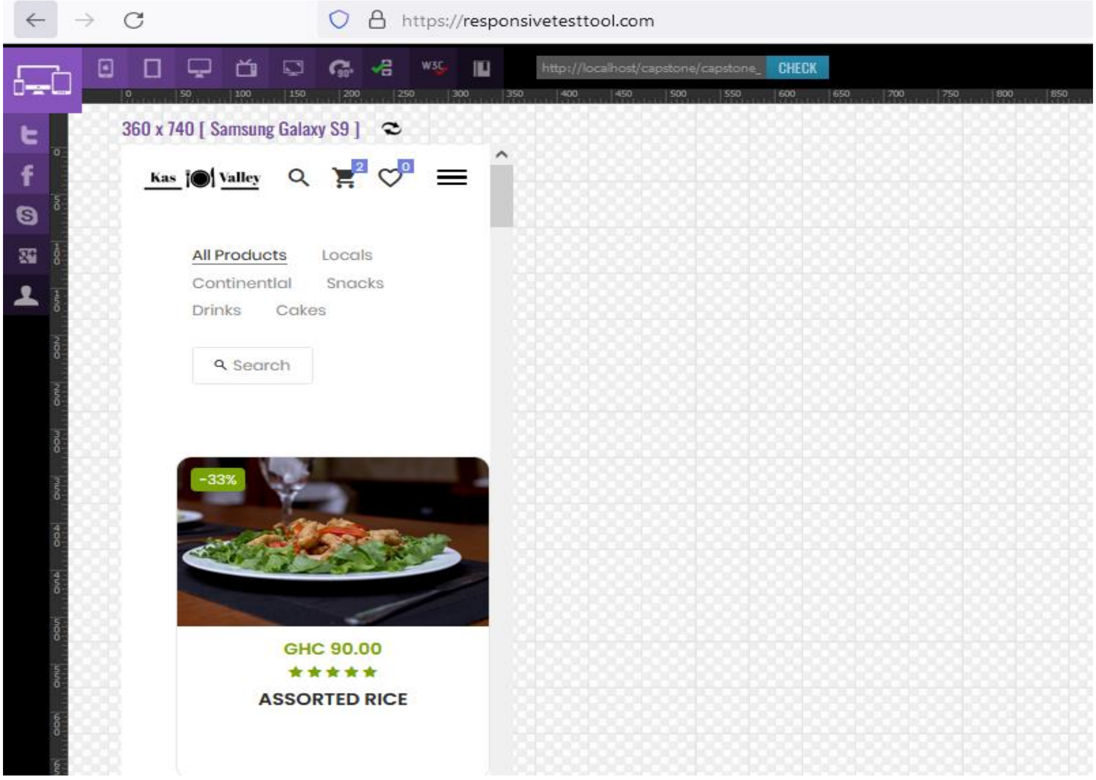


Figure 5.9: Responsiveness test on a phone

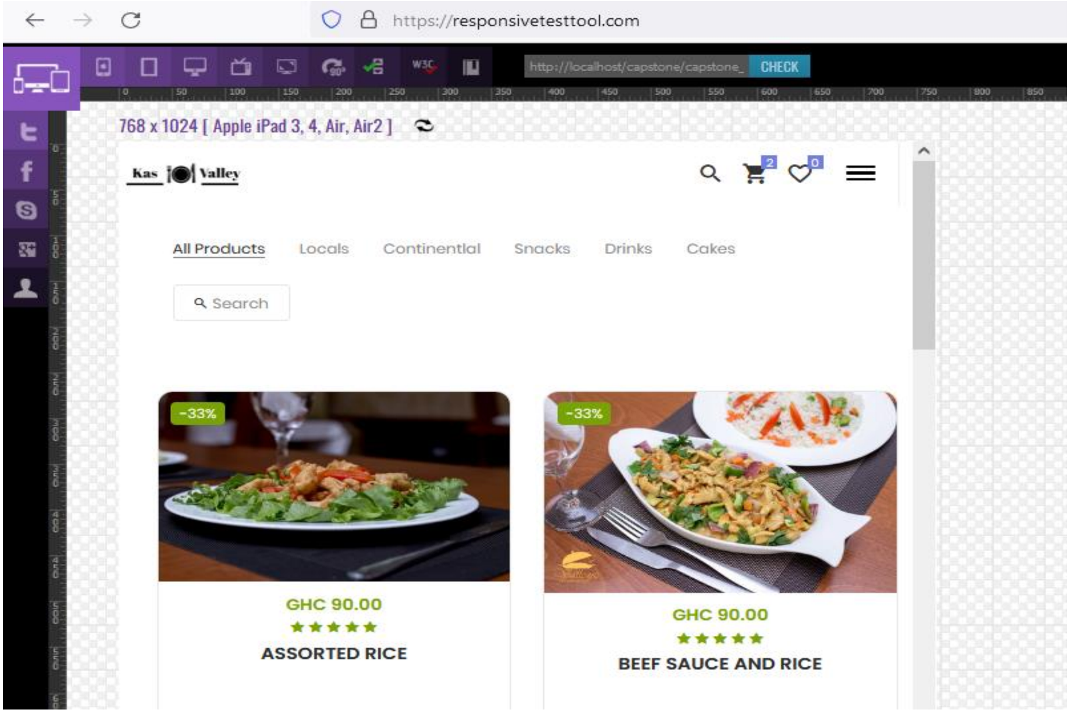


Figure 5.10: Responsiveness test on a tablet

5.5 User Acceptance Testing

This testing is done by the client or end-user of the system to validate the developed application against business requirements. Hence this testing enables end-users to interact with the fully functioning system to determine if everything works well to their expectations. The main aim is to see if the developed solution works well for the end-user.

This project has two main user categories: the restaurant and its customers. Thus, the user acceptance testing involved these two groups of users. Firstly, the restaurant manager tested the system to gauge the system's overall performance from her perspective. The manager evaluated the interface, the system's functionalities, convenience, and usefulness in the testing process. The results are:

1. System functionalities: This received a rating of 4/5
2. User Interface: A rating of 4/5
3. Convenience: A rating of 4/5
4. Usefulness: A rating of 4/5

Also, ten people tested the system from the customer's point of view. Overall, they said the system met their expectations. However, there were a few concerns and suggestions raised. Find below the highlights:

1. The user interface was attractive, and the colour combination was pleasing to the eye.
2. The table reservation process and meal ordering process were easy to perform
3. The navigation panel on the application was easy to use

4. One of the test participants suggested that when asking the user for their delivery location, the system should allow the user to select their location from a popup map rather than making the user just type their location

5. Also, another test participant suggested that there should be a way to track your order once it has left the restaurant for delivery.

5.6 Analysis of Test Results

From all the tests conducted, it can be concluded that the system does meet user expectations to a large extent. The PHPUnit tests showed that all the PHP functions were working as expected. Regarding the system testing, when the components were integrated and tested as a whole, the results showed that the application met the user requirements. After the user acceptance testing, some users expressed some concerns and raised some suggestions to improve the application. For instance, a user mentioned that the application should have a tracking system to enable them to monitor their orders after they have been dispatched from the restaurant. This would be taken into consideration for future work and developments.

However, in reference to the nineteen functional requirements stated in the requirement specification of the project, all these requirements were implemented in the system. Upon testing, the results showed that they meet user expectations. Below is a summary of testing the application against the requirements.

Table 5.1: Summary of testing

Test Number	Test description	Passed or failed
01	Login and signup functionality	Passed
02	Add/delete/update cart	Passed
03	Pay for order online	Passed

04	Book a table reservation	Passed
05	View all orders and reservations	Passed
06	Cancel order or reservation	Passed
07	Search for menu item	Passed
08	SMS alert for customer’s order	Passed
09	Logout functionality	Passed
10	Display graphs	Passed
11	Add/delete/update online menu	Passed
12	Change status of order and reservation	Passed
13	Check if a day/time has been fully booked	Passed
14	Add/delete/update customer data	Passed
15	View subscribers	Passed
16	Email all subscribers	Passed
17	Search for reservation and orders	Passed
18	Log out functionality	Passed
19	View all messages sent from the website	Passed
21	Add/delete menu category	Passed
22	Add/delete/update table reservation	Passed
23	Add/delete/update order	Passed

5.7 Conclusion

This chapter discussed the evaluation and testing of the fully developed web-based RMS. Unit testing was conducted using PHPUnit testing to ensure that the individual components of the system function as expected. Also, a browser compatibility test was conducted using chrome, internet explorer, Microsoft Edge and Firefox. A responsive test was conducted on a mobile phone, tablet, and desktop P.C. using an online tool to evaluate the system’s responsiveness on various devices. Finally, a user acceptance test was performed to ensure that the system met user expectations.

Chapter 6: Conclusion and Recommendations

6.1 Overview

The chapter summarizes the project in totality and highlights the limitations of the developed system and challenges faced during the project execution. The chapter ends with recommendations for future work on the development of such applications.

6.2 Project Summary

Technological advancements in recent years have impacted the food industry’s profitability and operations globally. In Ghana, however, the utilization of restaurant software technologies has not been maximized to their total capacity. Most restaurant software are limited to POS transactions or online-food ordering systems. Other aspects of restaurant operations such as table reservation management and customer data management are often not included in these applications. Without a centralized management system to handle the related aspects of a restaurant’s operations, there would be issues with data management, such as data redundancy and inconsistency. Therefore, this project aims to build a web-based restaurant management system for Kas Valley Restaurant in Ghana to act as a centralized system for managing their restaurant operations. Kas Valley faces challenges taking and processing an increasing number of orders per day. Some of the challenges are long queues, data redundancy, and human errors such as wrong order entry. The business also has difficulty keeping track of its records, including customer data, as they are dispersed in different locations. Thus, this project aims to design a web-based restaurant management application to act as a centralized system to solve the aforementioned issues.

The web-based RMS enables the restaurant to manage an online menu by allowing them to add a menu item and delete or modify one. By presenting an online menu, the web-based RMS facilitates online food ordering and payment and online table reservations for customers. The system enables the restaurant to process in-person orders and manage table reservations. It also allows the restaurant to view all incoming orders and update order status accordingly. Customers are sent SMS notifications on the state of their orders. The restaurant can manage data records on this same system, including data relating to customers, orders, reservations, and the restaurant's menu. There is also an analysis section on the system where the restaurant can view all quantitative data and charts relating to the restaurant's business. The execution of this project has contributed to the achievement of SDG 9: "Build resilient infrastructure, promote sustainable industrialization and foster innovation". This web-based RMS will promote innovation and technological progress in the restaurant, eventually increasing productivity and income.

6.3 Recommendations

Though the developed system meets user expectations and requirements to a large extent, the functionalities are not exhaustive for achieving full optimality. Below are the limitations of the system and some recommendations for future work.

6.3.1 Limitations and Challenges

Below are the limitations and challenges of the proposed system:

- The system does not have a feature to track orders once dispatched. This may leave customers wondering where exactly their order has reached in the case where customers opt for delivery.
- The application is a web-based application which requires internet connectivity to function. Thus, it cannot be used when there is no internet reception.
- The application is limited in functionality on the internet explorer browser in terms of browser support. For instance, the HTML data and time input fields do not show a popup calendar and time selector on the internet explorer browser.

- There is no mobile app version for the system as it runs solely on a web browser.

6.3.2 Future Works

The following are some recommendations to be considered for future developments:

- The application should have a map tracking feature to help customers track their order once it has been dispatched from the restaurant for delivery.
- There should be a mobile app version of the system to enable the system to run on mobile devices without the need for a web browser.
- The admin system should display incoming orders in real-time without requiring the admin to refresh the page to view these new incoming orders.

6.4 Conclusion

This chapter summarizes the project and concludes on the limitations of the developed system and recommendations for future works. In conclusion, this project aimed to create a web-based restaurant management system for Kas Valley restaurant to support online and in-person food ordering and payment, table reservation management, customer data, and analytics of the business’s sales.

References

[1]Opstad Leiv, Johannes Idso and Robin Valenta. 2022. The Dynamics of the Profitability and Growth of Restaurants; The Case of Norway. Economies 10: 53. Retrieved from https://doi.org/10.3390/economies10020053

[2]Lotte Wellton, Inger M. Jonsson and Anette Svingstedt. 2018. “We are service people, and we stay until the job is done”: enactments of professionalism in restaurants, Journal of Teaching in Travel & Tourism, 18:4, 265-283, DOI: 10.1080/15313220.2018.1474414

[3]Doris Dokua Sasu. 2021. Annual contributions of hotels and restaurants to GDP in Ghana, 2013-2020. Statista. Retrieved from https://www.statista.com/statistics/1271615/annual-contributions-of-hotels-and-restaurants-to-gdp-in-ghana/

[4] Lock S. 2022. Daily year-on-year impact of COVID-19 on global restaurant dining 2020-2022. Statista. Retrieved from https://www.statista.com/statistics/1103928/coronavirus-restaurant-visitation-impact/

[5] National Restaurant Association Research Group.2020. COVID-19 Restaurant Impact Survey December 2020. Retrieved from https://restaurant.org/downloads/pdfs/advocacy/covid-19-restaurant-impact-survey-v-state-results.pdf

[6] Restaurant Point-of-Sale Terminal Market Report, 2021-2028. 2021. Market analysis report, report-id: GVR-1-68038-487-1. Retrieved from https://www.grandviewresearch.com/industry-analysis/restaurant-point-of-sale-pos-terminal-market#

[7] Transparency Market Research, Global POS Restaurant Management Systems.2017.Retrieved from https://www.researchandmarkets.com/reports/5030633/pos-restaurant-management-systems-global-market

[8]Lock S. 2022. Digitalization of the restaurant industry worldwide - statistics & facts. Retrieved from https://www.statista.com/topics/8103/digitalization-of-the-restaurant-industry/#dossierKeyfigures

[9] Kocaman, E.M. and Kocaman M. 2019. Restaurant Management System (RMS) and Digital Conversion. Advances in Hospitality, Tourism, and the Services Industry.

[10]Emel Memis Kocaman. 2021. Operational effects of using restaurant management systems: An assessment according to business features. International Journal of Gastronomy and Food Science Volume 25, 100407, ISSN 1828-450X, Retrieved from https://doi.org/10.1016/j.ijgfs.2021.100408

[11]Kocaman E.M. and Kocaman M. 2019. Restaurant Management System (RMS) and Digital Conversion. Advances in Hospitality, Tourism, and the Services Industry.

[12]Mishul Prasad E. and Scornavacca H. Lehmann. 2005. Using wireless personal digital assistants in a restaurant: impact and perceived benefits. *International Conference on Mobile Business (ICMB'05)*, pp. 69-74.DOI: 10.1109/ICMB.2005.112.

[13] Khairunnisa K, Ayob J, Mohd Helmy, A. Wahab, M. Erdi Ayob, M. Izwan Ayob and M. Afif Ayob. 2009. The Application of Wireless Food Ordering System, *MASAUM Journal of Computing*, 1(2).

[14]X. Hongzhen, T. Bin and S. Wenlin. 2009. Wireless Food Ordering System Based on Web Services, *Second International Conference on Intelligent Computation Technology and Automation*, pp. 475-478. DOI: 10.1109/ICICTA.2009.83

[15] B. K. Mishra, B. S. Choudhary and T. Bakshi. 2015. Touch based digital ordering system on Android using GSM and Bluetooth for restaurants, *Annual IEEE India Conference (INDICON)*, pp. 1-5. DOI: 10.1109/INDICON.2015.7443374.

[16]N. A. Samsudin, S. K. Ahmad Khalid, M. F. A. Mohd Kohar, Z. Senin and M. N. Ihkasan. 2011. A customizable wireless food ordering system with realtime customer feedback, *IEEE Symposium on Wireless Technology and Applications (ISWTA)*, pp. 186-191. DOI: 10.1109/ISWTA.2011.6089405.

[17]Douglas, Alecia. 2007. A POS Solution for Restaurants. *Journal of Foodservice Business Research*, 10(1). DOI: 101-112. 10.1300/J369v10n01_06.

[18]Lolonyo Letsa. 2017. Assessing the Effect of Waiting Times on Restaurant Service Delivery in the Ho Municipality, Ghana. *European Business & Management*, 3(6), pp. 113-119. DOI: 10.11648/j.ebm.20170306.13

[19]J. Harpanahalli, K. Bhingradia, P. Jain and J. Koti. 2020. Smart Restaurant System using RFID Technology, *Fourth International Conference on Computing Methodologies and Communication (ICCMC)*. pp. 876-880. DOI: 10.1109/ICCMC48092.2020.ICCMC-000162

[20] Riofrío Valdivieso, Andrés Quinga-Socasi, Francisco Bustamante-Orellana and Carlos Andrade Eddy. 2020. RFID and Wireless Based System for Restaurant Industry. *Advances in Emerging Trends and Technologies*, pp.291-302. DOI:10.1007/978-3-030-32033-1_27.

[21]V. Liyanage, A. Ekanayake, H. Premasiri, P. Munasinghe and S. Thelijjagoda. 2018. Foody - Smart Restaurant Management and Ordering System. *2018 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pp. 1-6. DOI: 10.1109/R10-HTC.2018.8629835.

[22]K. Lin, C. Chen, Z. Zhang and S. Ou. 2018. NFC-based mobile application design restaurant ordering system APP, *2018 IEEE International Conference on Applied System Invention (ICASI),* pp. 737-740. DOI: 10.1109/ICASI.2018.8394365.

[23]E. Y. Daraghmi and S. Yuan. 2013. PMR: Personalized Mobile Restaurant system, 2013 5th International Conference on Computer Science and Information Technology, 2013, pp. 275-282, doi: 10.1109/CSIT.2013.6588792.

[24] Alfiya M. Shaikh, Aditya C. Shetgaonkar, Sandesh S. Dalvi, Shubham S. Sawant, Kruti V. Singh, Sagar K. Naik, and Rajesh K. Gauns. 2019. Food Ordering Management using Recommendations, International Journal of Advanced Engineering Research and Science (IJAERS), 6, 6. ISSN: 2349-6495(P) | 2456-1908(O)

[25] G. L. Intal, J. D. Payas, L. M. Fernandez and B. M. Domingo. 2020. Restaurant Information System (RIS) with Q.R. Code to Improve Service Operations of Casual Fine Dining Restaurant, *IEEE 7th International Conference on Industrial Engineering and Applications (ICIEA)*, pp. 1054-1059. DOI: 10.1109/ICIEA49774.2020.9102036.

[26] MDN. 2022. HTML basics. Retrieved from https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics

[27] MDN. 2022. CSS: Cascading Style Sheets. Retrieved from https://developer.mozilla.org/en-US/docs/Web/CSS

[28] MDN. 2021. JavaScript. Retrieved from https://developer.mozilla.org/en-US/docs/Web/JavaScript

[29]Bootstrap. n.d. Build fast, responsive sites with Bootstrap/ Retrieved from https://getbootstrap.com/

[30]JQuery. n.d. What is jQuery? Retrieved from https://jquery.com/

[31]PHP. 2022. PHP. Retrieved from https://www.php.net/

[32] Lindsay Moore.2018. What is mysql? Retrieved from https://www.techtarget.com/searchoracle/definition/MySQL

[33]Paystack. n.d. Paystack Documentation. Retrieved https://paystack.com/docs/

[34]Marcus Bointon. 2022. PHPMailer – A full-featured email creation and transfer class for PHP. Retrieved from https://github.com/PHPMailer/PHPMailer

[35]Chart.js. n.d. Simple yet flexible JavaScript charting for designers & developers https://www.chartjs.org/

[36]MDN.2022.What's AJAX? Retrieved from https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting_Started

[37]W3Schools. n.d.What is JSON? Retrieved from https://www.w3schools.com/whatis/whatis_json.asp

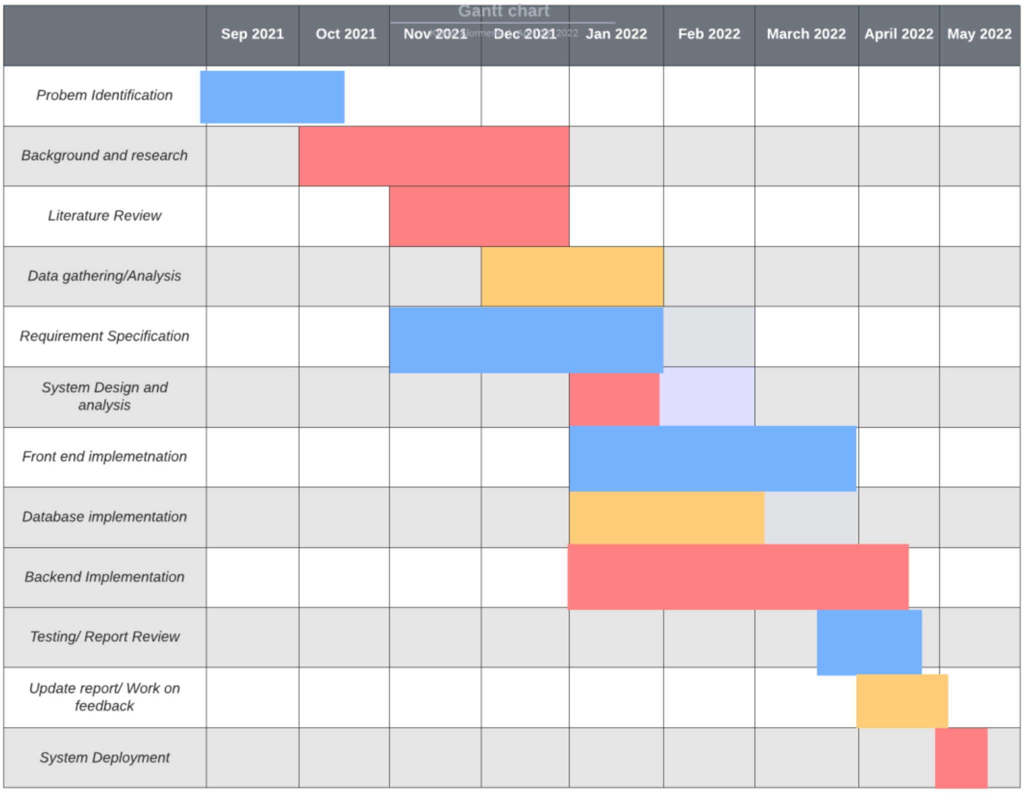
[38]Apache Friends. 2022 .XAMPP Apache + MariaDB + PHP + Perl. Retrieved from https://www.apachefriends.org/index.html

[39]PhpMyAdmin. 2022. Bringing MySQL to the web. https://www.phpmyadmin.net/

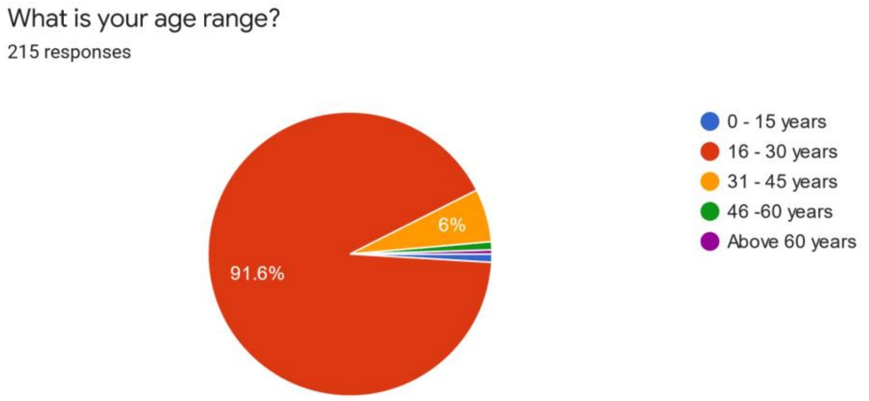
[40] Responsive Tool. Retrieved from https://responsivetesttool.com/

Appendices

Appendix A: Project Gantt chart

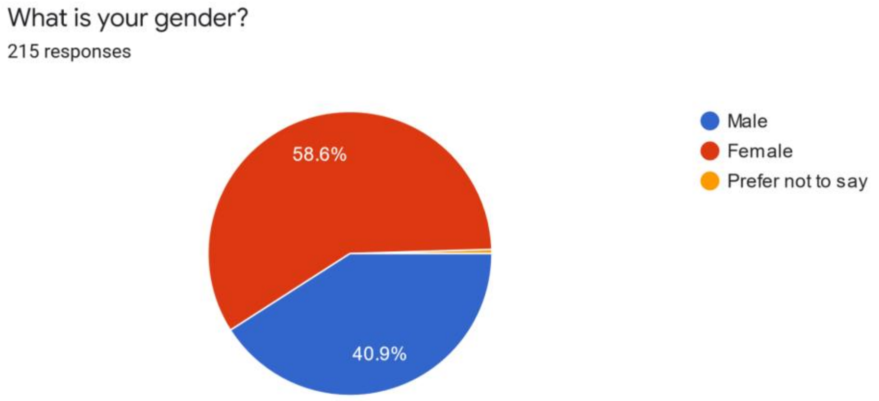


Appendix B: Results from Online Survey (Data Gathering Process)



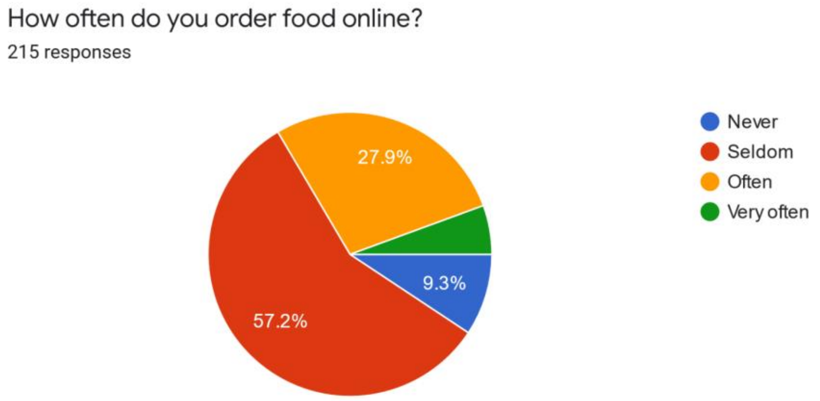
What is your age range?

215 responses



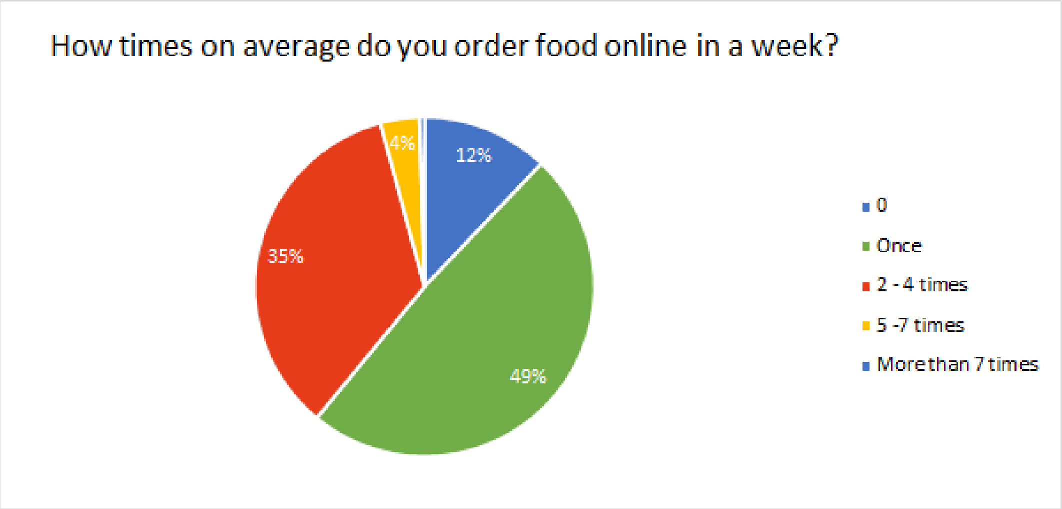
What is your gender?

215 responses



How often do you order food online?

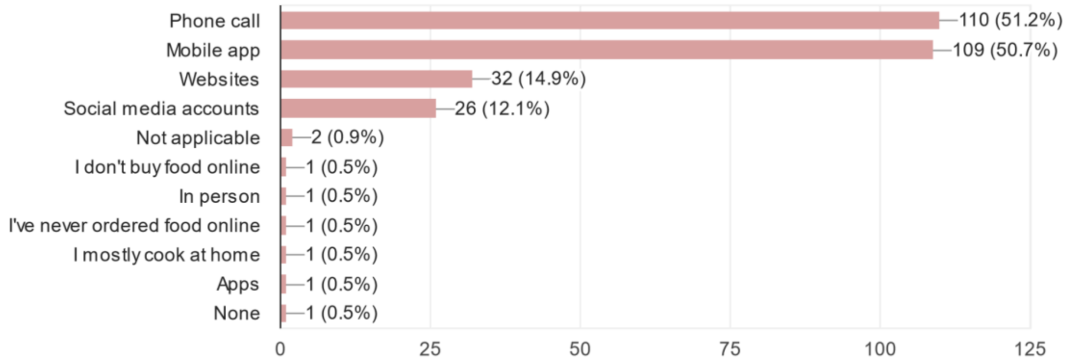
215 responses



How times on average do you order food online in a week?

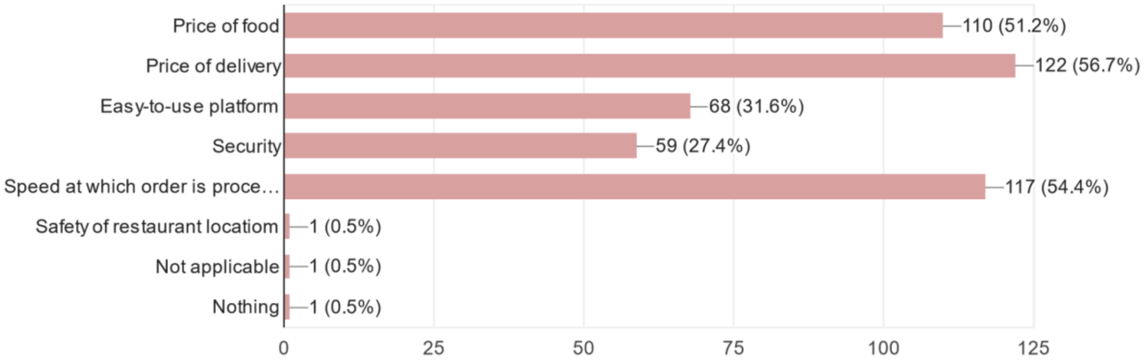
What is usually your preferred method of ordering food?

215 responses



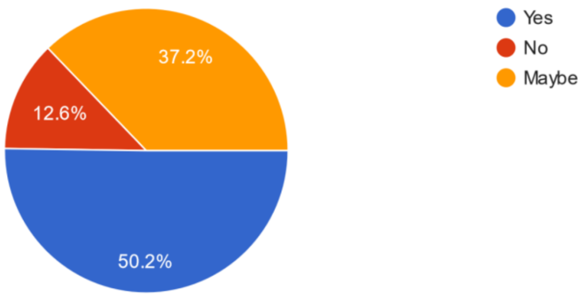
Which of these is/are very important to you when ordering food online?

215 responses



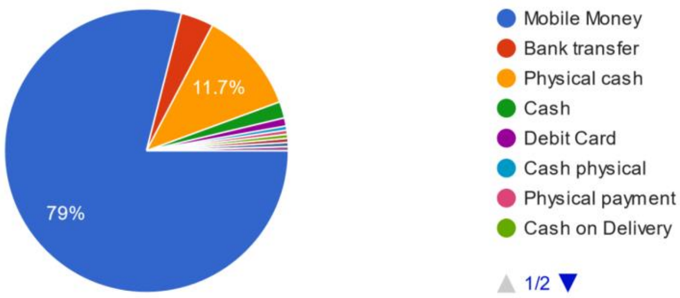
Would you use a restaurant's online ordering website to order food from the restaurant instead of using popular delivery platforms like ubereats, Jumia food, etc?

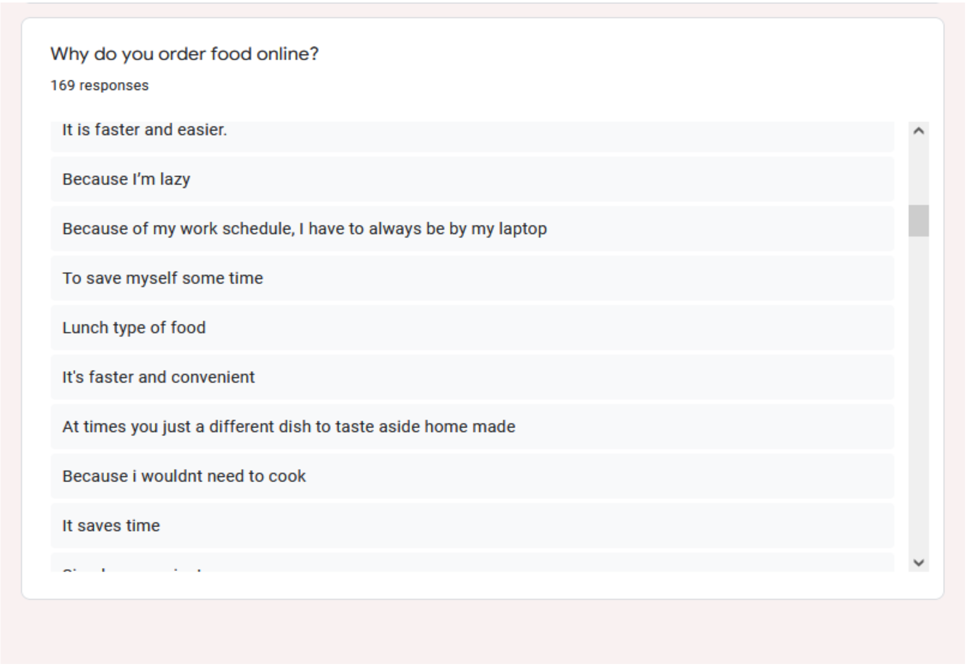
215 responses



Which payment method do you prefer when ordering meals online?

214 responses





Why do you order food online?

169 responses

It is faster and easier.

Because I'm lazy

Because of my work schedule, I have to always be by my laptop

To save myself some time

Lunch type of food

It's faster and convenient

At times you just a different dish to taste aside home made

Because i wouldnt need to cook

It saves time