

SCHOOL OF TECHNOLOGY

BACHELOR OF SCIENCE IN SOFTWARE DEVELOPMENT BSD 3201 FINAL YEAR PROJECT II (2 SEM)

# TITLE: MG FOOD HAVEN MANAGEMENT SYSTEM BY: MARTIN MWAGAMBO GAMBO

**REGNO: 19/02803**

**EMAIL:** [**1902803@students.kcau.ac.ke**](mailto:1902803@students.kcau.ac.ke) **SUPERVISOR:** MADAM GLADYS MANGE

# PRESENTATION DATE:

**THIS MANAGEMENT SYSTEM IS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE AWARD OF BACHELOR OF SCIENCE IN SOFTWARE DEVELOPMENT IN KCA UNIVERSITY.**

Table of Contents

[SYSTEM IMPLEMENTATION](#_bookmark0) [4](#_bookmark0)

* 1. [Introduction](#_bookmark1) [4](#_bookmark1)
  2. [Testing the System](#_bookmark2) [4](#_bookmark2)
     1. [Testing types](#_bookmark3) [4](#_bookmark3)
  3. [Staff Training](#_bookmark4) [4](#_bookmark4)
     1. [Purpose of staff training](#_bookmark5) [5](#_bookmark5)
  4. [Documentation](#_bookmark6) [5](#_bookmark6)
  5. [Implementation strategies](#_bookmark7) [6](#_bookmark7)
     1. [Parallel running](#_bookmark8) [7](#_bookmark8)
  6. [Implementation Schedule](#_bookmark9) [10](#_bookmark9)

[USER MANUAL](#_bookmark10) [11](#_bookmark10)

[6.0 Introduction](#_bookmark11) [11](#_bookmark11)

* 1. [Installation](#_bookmark12) [12](#_bookmark12)
  2. [Running the System](#_bookmark13) [13](#_bookmark13)

[Home page](#_bookmark14) [14](#_bookmark14)

[Register](#_bookmark15) [14](#_bookmark15)

[Login Form](#_bookmark16) [14](#_bookmark16)

[User Login](#_bookmark17) [15](#_bookmark17)

[User dashboard](#_bookmark18) [15](#_bookmark18)

[About](#_bookmark19) [16](#_bookmark19)

[Menu](#_bookmark20) [16](#_bookmark20)

[Contact](#_bookmark21) [17](#_bookmark21)

[Cart](#_bookmark22) [18](#_bookmark22)

[Order History](#_bookmark23) [18](#_bookmark23)

[Check Out](#_bookmark24) [19](#_bookmark24)

[Order Placement](#_bookmark25) [19](#_bookmark25)

[Email verification Order Placed Receipt](#_bookmark26) [20](#_bookmark26)

[Admin Login](#_bookmark27) [20](#_bookmark27)

[Admin Dashboard](#_bookmark28) [21](#_bookmark28)

[Orders](#_bookmark29) [21](#_bookmark29)

[Categories](#_bookmark30) [22](#_bookmark30)

[Items](#_bookmark31) [22](#_bookmark31)

[View/Modify Logins](#_bookmark32) [23](#_bookmark32)

[Add Logins](#_bookmark33) [23](#_bookmark33)

[Staff Login](#_bookmark34) [24](#_bookmark34)

[Staff Dashboard](#_bookmark35) [24](#_bookmark35)

[Forgot Password](#_bookmark36) [25](#_bookmark36)

[Email Verification](#_bookmark37) [25](#_bookmark37)

* 1. [Log Out](#_bookmark38) [27](#_bookmark38)
  2. [Data Sources](#_bookmark39) [28](#_bookmark39)
  3. [Threats](#_bookmark40) [29](#_bookmark40)
  4. [Contact Support](#_bookmark41) [30](#_bookmark41)

[PROJECT SCHEDULE](#_bookmark42) [31](#_bookmark42)

* 1. [APPENDIX](#_bookmark43) [32](#_bookmark43)
  2. [Glossary](#_bookmark44) [32](#_bookmark44)
  3. [Code](#_bookmark45) [32](#_bookmark45)
     1. [Sample Code for Home Page](#_bookmark46) [32](#_bookmark46)
     2. [Sample Code for Login Page](#_bookmark47) [34](#_bookmark47)
     3. [Sample Codes for Register Page](#_bookmark48) [35](#_bookmark48)

[8.0 REFERENCES](#_bookmark49) [37](#_bookmark49)

# SYSTEM IMPLEMENTATION

# Introduction

It is the handover of the system from the development team to the client and the key activities that will be carried out will be:

Testing the system Staff Training Documentation

# Testing the System

It is done to check whether there are any errors and whether the system performance meets the expectations of the client.

# Testing types

* + - 1. Unit/Module Testing-Testing the individual programs and this is done by the programming staff.
      2. System Testing-This is designed to ensure that the subsystems work properly together i.e., whether the system links properly with visual basic. It ensures that the subsystems work properly when they are integrated with the whole system.
      3. User acceptance testing-this is the testing of the system by the user after the system has passed the system tests.

# Staff Training

It covers the retraining of current staff and the training of ne personnel who will be recruited. For the training to be effective it must be clear of what it is trying to archive.

There are several methods through which training can be conducted, including:

Lectures: Lectures involve a trainer or subject matter expert delivering information to a group of trainees. This method is commonly used to provide an overview of a topic or to convey theoretical concepts. Lectures can be effective in providing a broad understanding of a subject matter and creating a common foundation of knowledge among trainees.

Films: Films or videos are a visual and auditory medium that can be used to present information, demonstrate processes, or showcase real-life scenarios. They can be particularly effective in illustrating complex concepts, providing examples, or offering simulations of workplace situations. Films can engage trainees and make the learning experience more interactive and immersive.

Case study: Case studies involve analyzing real or hypothetical situations to understand the application of knowledge and problem-solving skills. Trainees are presented with a specific scenario, and they are required to analyze it, identify issues, and propose solutions based on their training. Case studies help develop critical thinking, decision-making, and problem- solving abilities in trainees.

Tutorials: Tutorials involve hands-on, practical guidance provided by an instructor. Trainees receive individualized or small-group instruction, allowing for personalized attention and feedback. Tutorials are particularly useful when learning specific skills, such as software operation, equipment handling, or technical procedures. They provide a supportive learning environment where trainees can actively practice and receive immediate guidance.

# Purpose of staff training

* + - 1. Convince user staff that the system efficient and effective
      2. Enables staff to carry out the tasks expected of them.
      3. To overcome their fears of a new system
      4. To familiarize them with the new system

# Documentation

There are three types of documentation that are associated with implementation of the system. They include:

* Training Documentation

Used for providing detailed tuition in the operation of the newly developed system.

* User Documentation

Used for reference rather than learning purposes. It reflects the expertise and vocabulary of the variety of users involved in the system. It should concentrate on the issues that concern users the most i.e., functions

* Operations Documentation

Responsible for the day to day running of the system and it teaches the normal operating procedures and how to respond to errors i.e., user cannot insert details about

students in the equipment’s table if there are no detail in student’s details table especially in student id primary key.

# Implementation strategies

Implementing an order system for MgFood will involve several key steps. Here are some implementation strategies I used to consider the strategy:

Define Requirements: Begin by clearly defining the requirements of the order system. Consider factors such as user roles, order placement process, payment options, delivery logistics, and integration with existing systems, if any. Documenting these requirements will serve as a roadmap for the implementation process.

Select an Appropriate Technology: Choose a technology stack that aligns with the requirements and objectives of your system. This may involve selecting a programming language, framework, and any necessary third-party tools or APIs. Consider factors such as scalability, security, ease of use, and compatibility with existing infrastructure.

Develop a Minimum Viable Product (MVP): Start with building a minimum viable product that includes the core features of the order system. This allows you to quickly validate the concept, gather feedback from users, and make necessary iterations before investing in the complete development.

Design User Interfaces: Create intuitive and user-friendly interfaces for both customers and restaurant staff. Ensure that the interfaces provide a seamless ordering experience, allowing customers to browse menus, customize their orders, and select delivery options. For restaurant staff, provide an easy-to-use interface for managing orders, updating menu items, and tracking deliveries.

Integrate Payment Gateway: Implement a secure and reliable payment gateway to facilitate online payments. Integrate popular payment methods such as credit cards, digital wallets, or mobile payment platforms. Ensure that the payment process is encrypted and compliant with relevant security standards.

Implement Order Management: Develop a robust order management system that allows restaurants to receive, process, and track orders efficiently. Include features such as real-time notifications, order status updates, and order history. Integration with a delivery management system can help streamline the logistics process.

Test and Quality Assurance: Conduct thorough testing to ensure the system functions as expected. Perform unit testing, integration testing, and end-to-end testing to identify and resolve any bugs or issues. Additionally, consider usability testing to gather feedback on the user experience and make necessary improvements.

Launch and Monitor: Once testing is complete, deploy the order system to production. Monitor its performance, scalability, and security to ensure smooth operation. Implement analytics and reporting features to track key metrics such as order volumes, customer preferences, and system uptime.

Provide Training and Support: Conduct training sessions to familiarize restaurant staff with the order system's functionalities and processes. Offer ongoing support channels, such as a helpdesk or dedicated support team, to address any user queries or issues that may arise.

Continuous Improvement: Regularly gather user feedback and analyze data to identify areas for improvement. Incorporate new features, optimize performance, and enhance the user experience based on user feedback and market trends.

The strategy that would be taken is Parallel Running.

# Parallel running

The existing and the newly developed system will run parallel to each other simultaneously as they introduce the system wholly for an agreed period of time and the results are compared. Once the user has complete confidence in the new system then the old system is abandoned.

# Advantages

Risk Mitigation: Parallel running reduces the risk associated with the implementation of a new system. By running both systems in parallel, any issues or bugs in the new system can be identified and addressed without disrupting the ongoing operations of the organization. This ensures that critical business processes continue uninterrupted.

Business Continuity: Parallel running helps maintain business continuity during the transition. If any problems arise with the new system, the organization can rely on the existing system to perform essential functions. This mitigates the potential impact of system failures, data loss, or operational disruptions that could occur during the transition phase.

Training and Familiarization: Parallel running allows employees to gradually adapt to the new system while still using the familiar old system. This approach provides an opportunity for training and hands-on experience with the new system without the pressure of immediate full adoption. Employees can gradually build confidence and proficiency, reducing the learning curve and minimizing productivity losses.

Comparison and Validation: Parallel running enables a direct comparison between the old and new systems. Users can verify the accuracy and consistency of data and processes across both systems, ensuring that the new system performs as expected and meets the organization's requirements. This validation process helps build trust and confidence

in the new system.

Fallback Plan: A fallback plan refers to a contingency strategy or alternative course of action that is prepared in case the new system encounters unexpected failures or issues. It ensures that if the new system doesn't meet expectations or experiences significant problems, there is a backup solution to rely on. The fallback plan could involve reverting to the old system temporarily or implementing an interim solution until the new system is fixed or replaced.

System was compared to the old system for the period of time. Helps use to get in touch with the new system

# Disadvantages

Increased Complexity: Running two systems simultaneously can introduce complexity to the overall process. It requires managing and synchronizing data between the old and new systems, ensuring consistency, and addressing any discrepancies that may arise. This complexity can add overhead and increase the risk of errors or data inconsistencies during the parallel running phase.

Resource Intensive: Parallel running often requires additional resources, such as hardware, software licenses, and manpower to maintain and operate both systems simultaneously. This can result in increased costs, both in terms of infrastructure and personnel. Organizations need to allocate resources effectively to support the parallel running phase.

Duplication of Efforts: Maintaining two systems in parallel means that tasks need to be performed twice. This can lead to duplication of efforts for data entry, order processing,

reporting, and other operational tasks. It can be time-consuming and require additional manpower, impacting overall productivity.

Increased Risk of Errors: Parallel running involves transferring data between the old and new systems, which can introduce a higher risk of data entry errors or discrepancies. If the two systems are not synchronized properly or if data is not transferred accurately, it can lead to incorrect or inconsistent information. This can result in customer dissatisfaction, operational inefficiencies, or financial implications.

Extended Transition Period: The parallel running phase typically extends the transition period from the old system to the new system. As a result, organizations may experience a prolonged period of adjustment and adaptation. This can impact productivity and efficiency, as users need to divide their attention and learn to operate in both systems simultaneously.

User Resistance and Confusion: Parallel running can cause confusion and resistance among users who need to adapt to two different systems. Users may find it challenging to switch between systems or may face difficulties in learning and adapting to the new system if they are still reliant on the old system. This can impact user satisfaction and may require additional training and support.

Maintenance and Support: Supporting and maintaining two systems during the parallel running phase requires additional effort. Software updates, bug fixes, and system enhancements need to be performed on both systems, which can increase the workload for IT teams. It may also be challenging to align support and maintenance activities for both systems effectively.

# Implementation Schedule

This shows how the implementation is going to take place

|  |  |  |  |
| --- | --- | --- | --- |
| **Title** | **Activity** | **Duration** | **Objective** |
| System Developer | Delivering the system to client | 10 minutes | Delivered the system  to client and installed it. |
| Users | Evaluating the system to their objectives | 5 minutes | Evaluated the system to their needs and made sure there  were no errors. |
| System developer | Training users | 10 minutes | Trained the end  users |

# USER MANUAL

# Introduction

Welcome to the User Manual for MgFood Order! This comprehensive guide has been created to help you effectively and efficiently use our product and make the most out of its features and functionalities. Whether you're a new user looking to get started or an experienced user seeking a reference, this manual will serve as your go-to resource.

In this user manual, we have included step-by-step instructions, detailed explanations, and useful tips to guide you through every aspect of using MgFood Order. From initial setup and configuration to performing various tasks and exploring advanced features, we aim to provide you with all the information you need to harness the full potential of our product.

We have designed this manual to be user-friendly and accessible, ensuring that even users with varying levels of technical expertise can easily follow along. The content is organized in a logical manner, allowing you to quickly locate the information you require. Additionally, we have included visual aids, such as screenshots and diagrams, to enhance your understanding and provide visual references for key concepts.

Please note that this user manual is regularly updated to reflect any new features, changes, or improvements to the product. We highly recommend referring to the latest version of this manual, which can be accessed on our website or through our support channels, to ensure you have the most up-to-date information at your disposal.

We understand that learning a new product can sometimes be overwhelming, but we are confident that this user manual will simplify the process and empower you to make the most of MgFood Order. If you have any questions or need further assistance, our dedicated support team is always ready to help.

# Hardware requirements

Computers The Processor – Core i5 least, processor speed should be 2.60 GHZ or greater, RAM should be 4GB or greater.

Printers-Epson L3150 series Prints about 4300 pages per cartridge Ink cartridges

Storage Media-about 250 GB SSD

# Software requirements

Operating System-Graphical Based Microsoft Windows 11-Popular, user friendly and best support

Antivirus Software-Kaspersky

Database- to save records, sales and inventory data- MySql

Vs Code-used in write in php html codes to help in the interactive support of the user.

# Installation

The installation process typically involves the following steps:

Determine Hosting Requirements Before proceeding, identify the hosting requirements of your system. This includes factors like server configuration, operating system, database support, and any other dependencies.

Choose a Hosting Provider Select a reliable hosting provider that meets your requirements. Consider factors such as server uptime, scalability, security measures, customer support, and pricing options. Popular hosting providers include Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform, and many others.

Sign up for Hosting Services Create an account with the hosting provider of your choice and select the appropriate hosting plan based on your needs. Follow the provider's instructions for account creation and payment.

Configure the Server Once you have access to the hosting platform, configure the server settings according to your system's requirements. This typically involves setting up the operating system, installing any necessary software dependencies, and configuring network settings.

Upload and Install the System Transfer your system's files to the server. This could involve uploading files via FTP (File Transfer Protocol), using a web-based file manager provided by the hosting provider, or any other method they support. Follow any specific installation instructions provided by your system's developers or documentation.

Configure System Settings Configure any necessary settings for your system. This may include setting up database connections, configuring email settings, setting up security measures (firewalls, SSL certificates), and other system-specific configurations. Refer to your system's documentation for guidance on configuration.

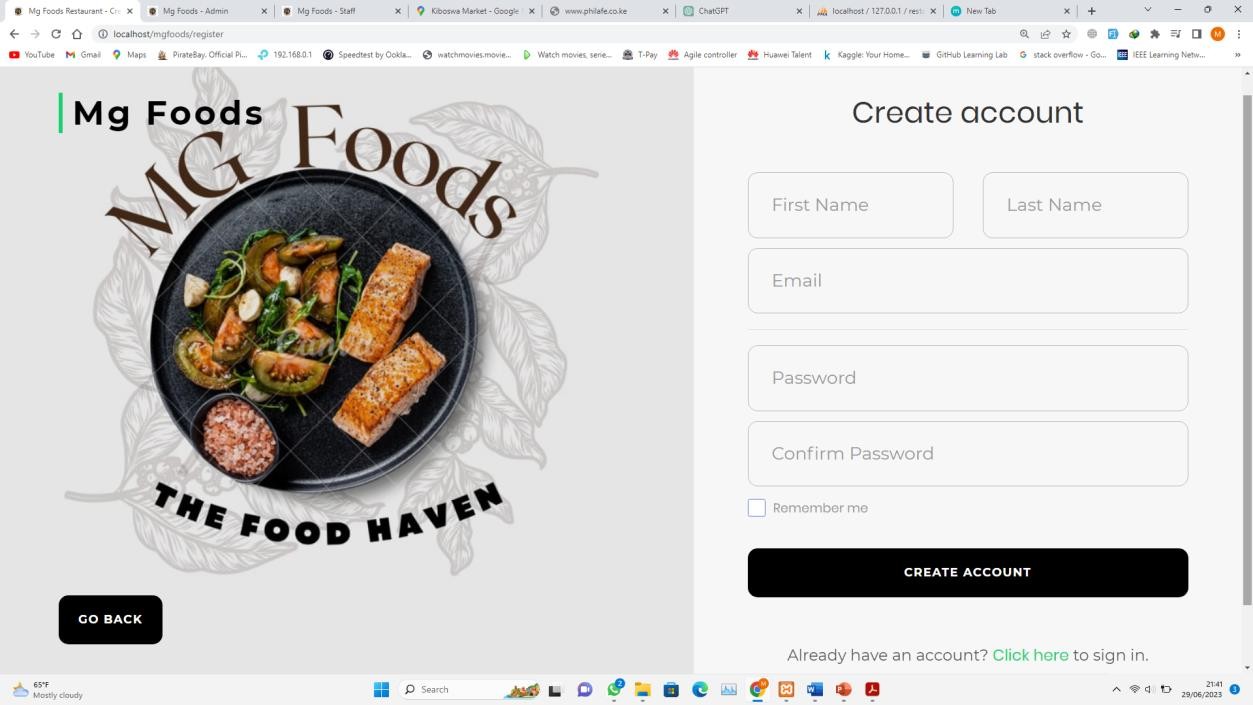
Test and Verify After the installation and configuration steps, thoroughly test the system to ensure it functions as expected. Test different features, workflows, and interactions to identify and resolve any issues.

Go Live Once you have successfully tested and verified the system, it's ready to go live. Update any DNS (Domain Name System) records or configure domain settings to point to your hosted system. Note: The above steps provide a general outline of the installation process. The actual steps and specific details may vary depending on the hosting provider, system requirements, and other factors. It's recommended to refer to the documentation and resources provided by your hosting provider and system developers for more detailed instructions.

# Running the System Xampp

# Home page

# Register

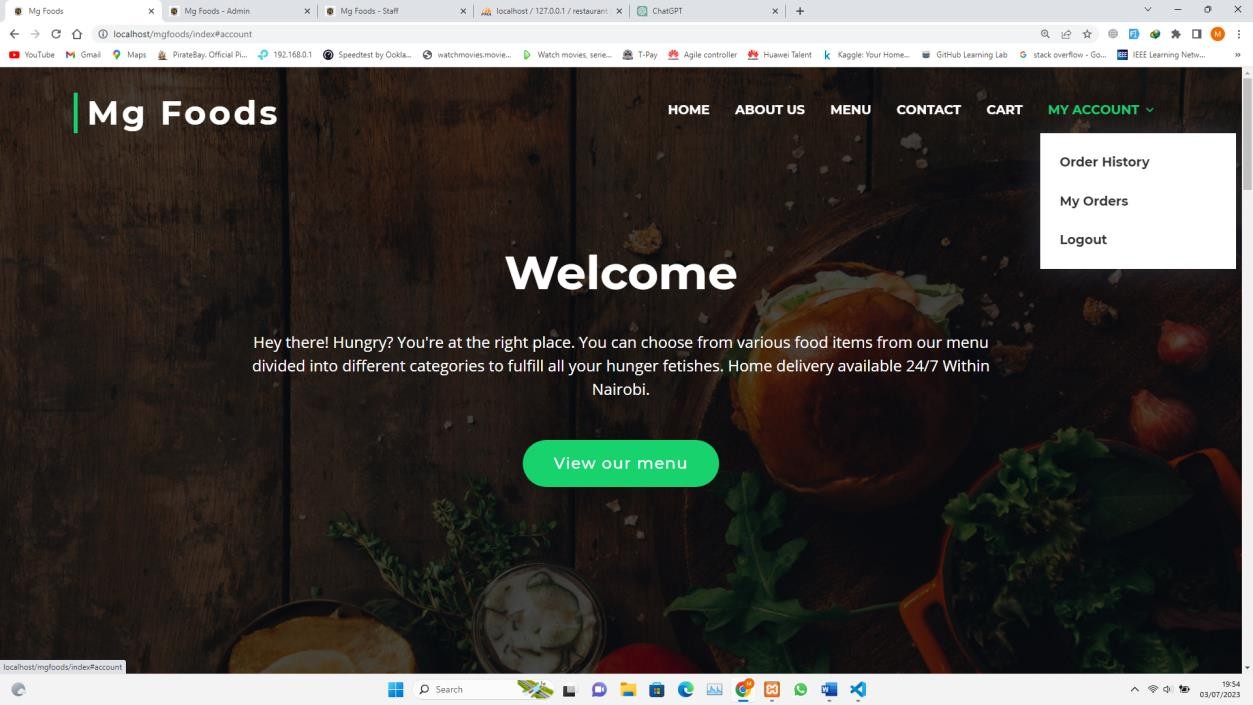


# Login Form

When the user types in the URL, the system will redirect them to home page where they will be required to log in. When a user clicks login without inputting the login credentials then a message box will be displayed telling user to enter password to proceed and if it is wrong then message box is displayed saying incorrect password.

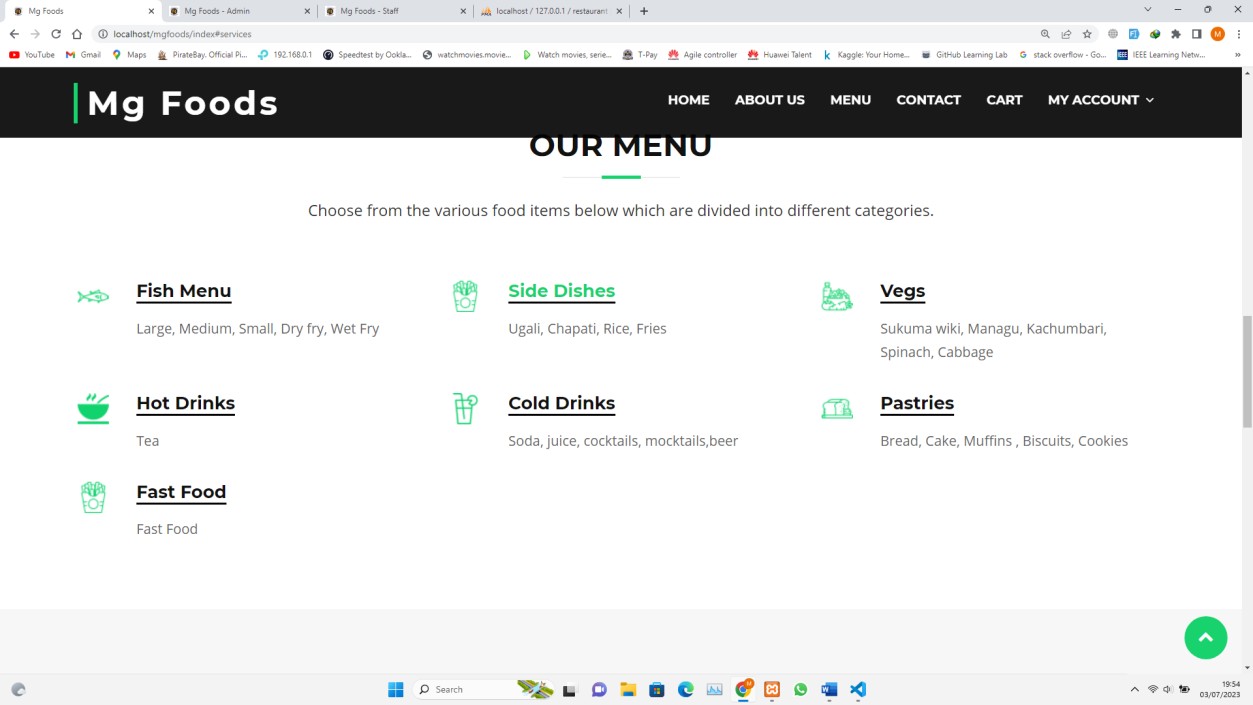
# User Login

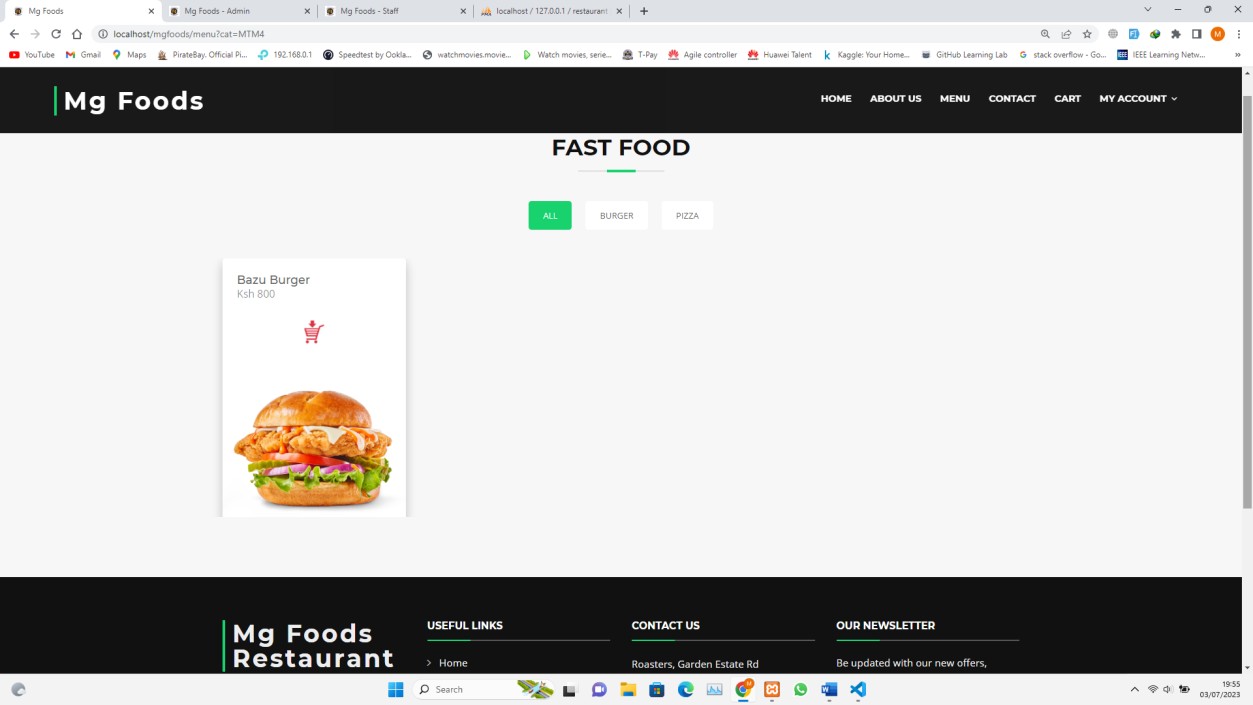
# User dashboard



# About

# Menu





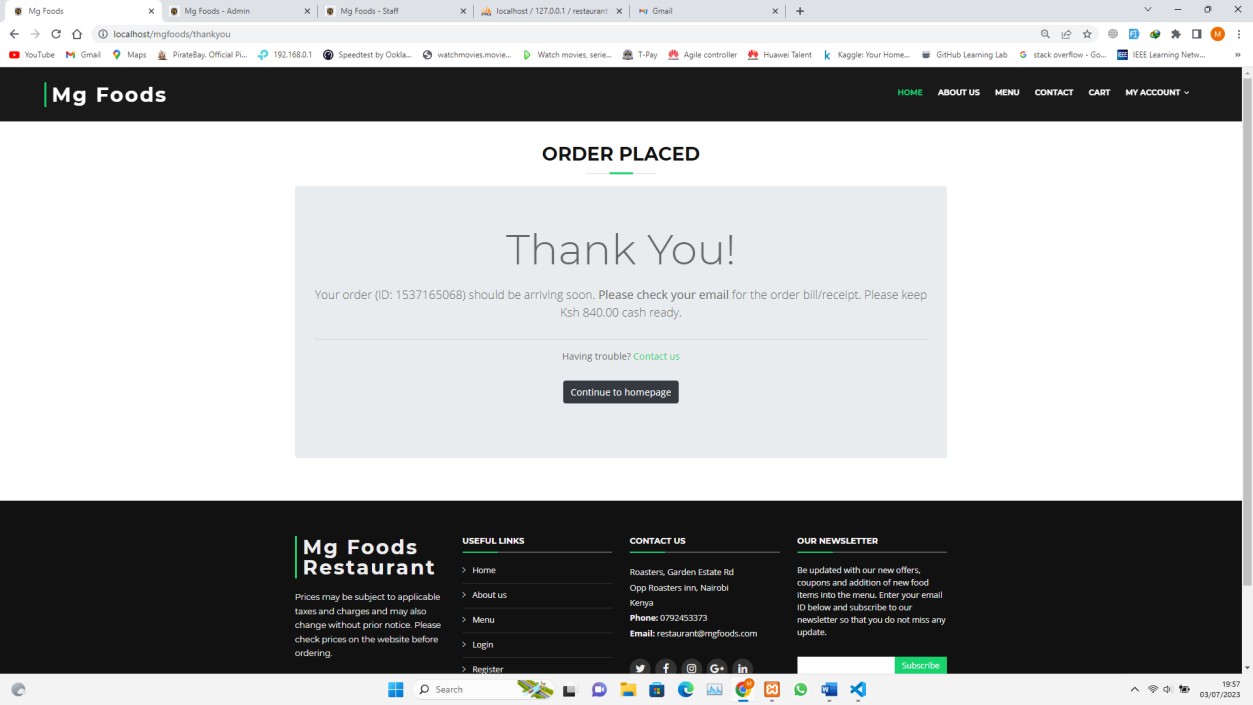
# Contact

# Cart

# Order History

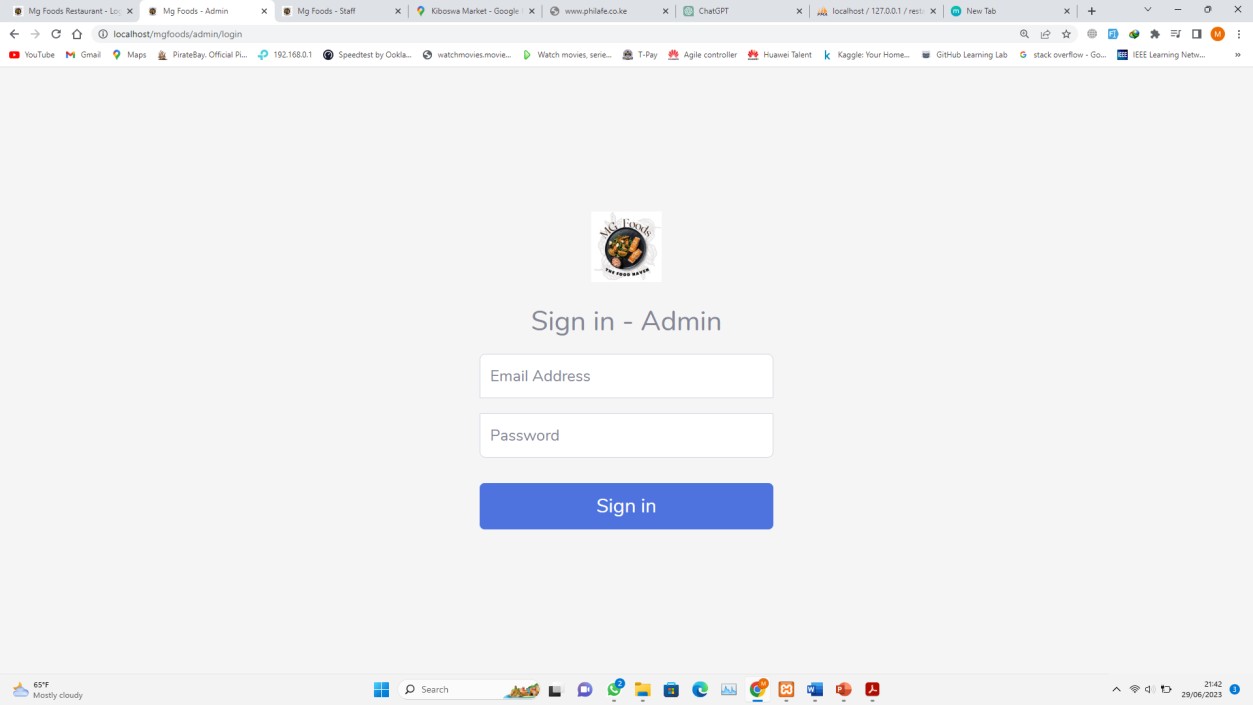
# Check Out

# Order Placement



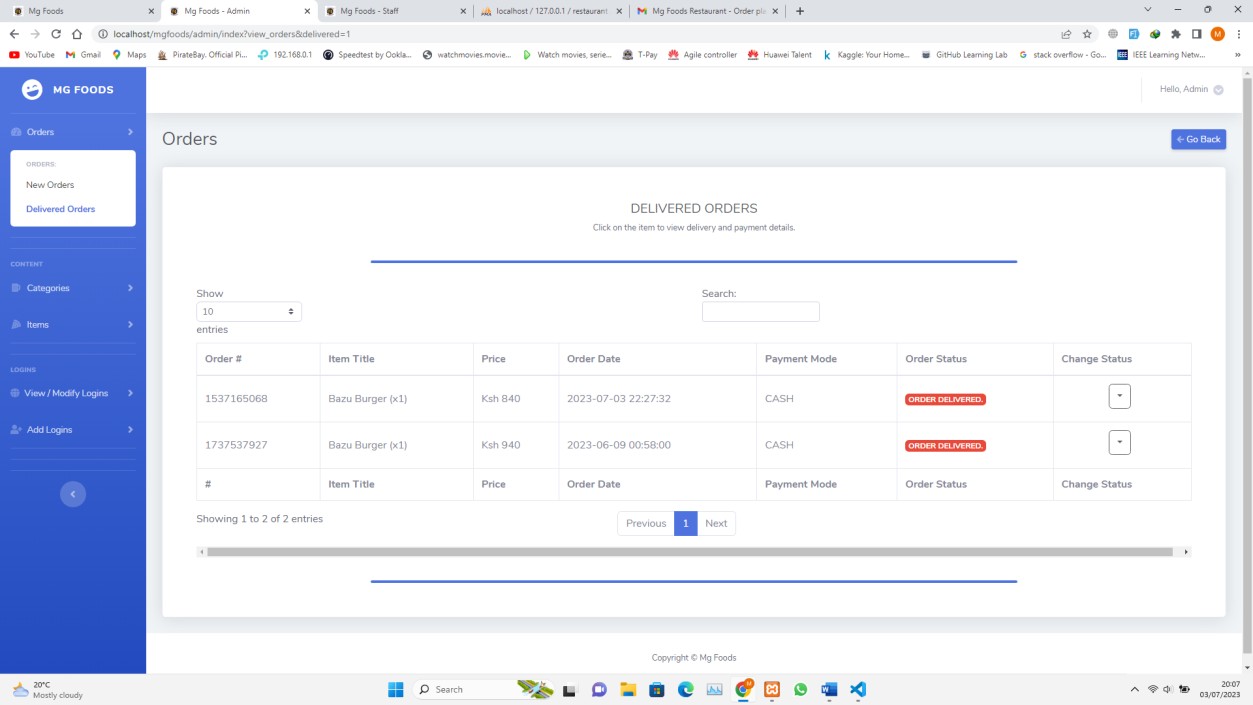
# Email verification Order Placed Receipt

# Admin Login



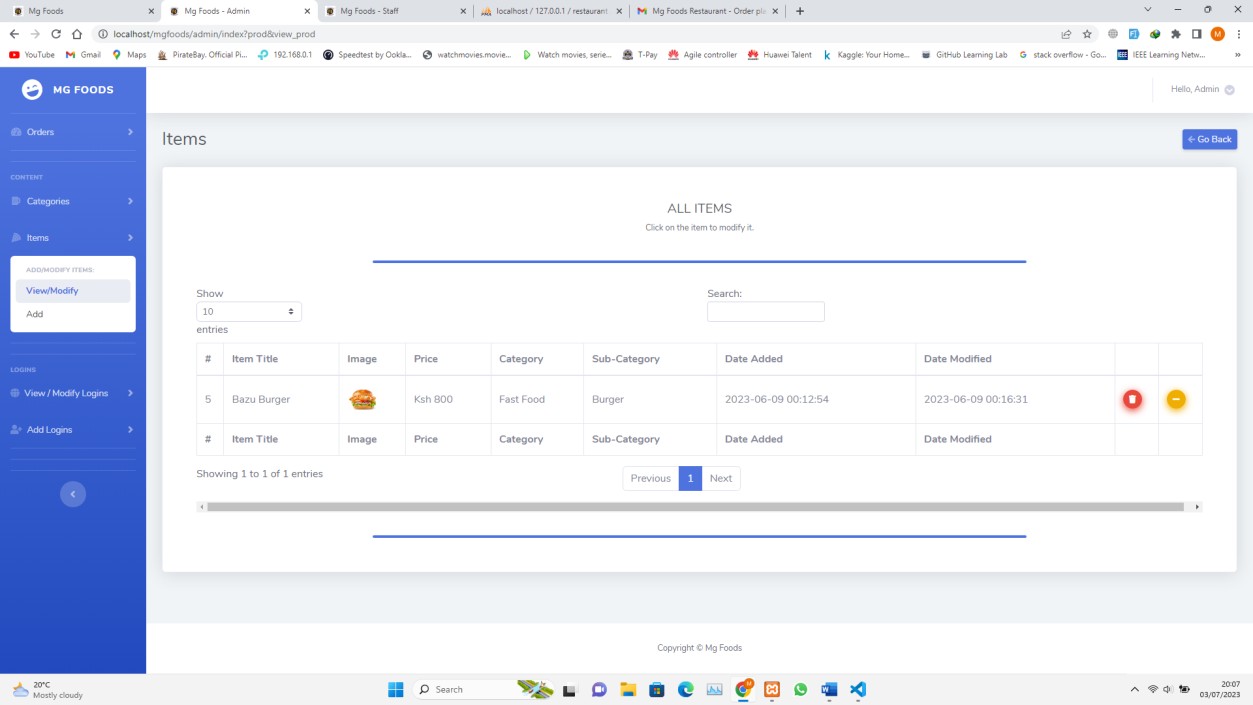
# Admin Dashboard

# Orders



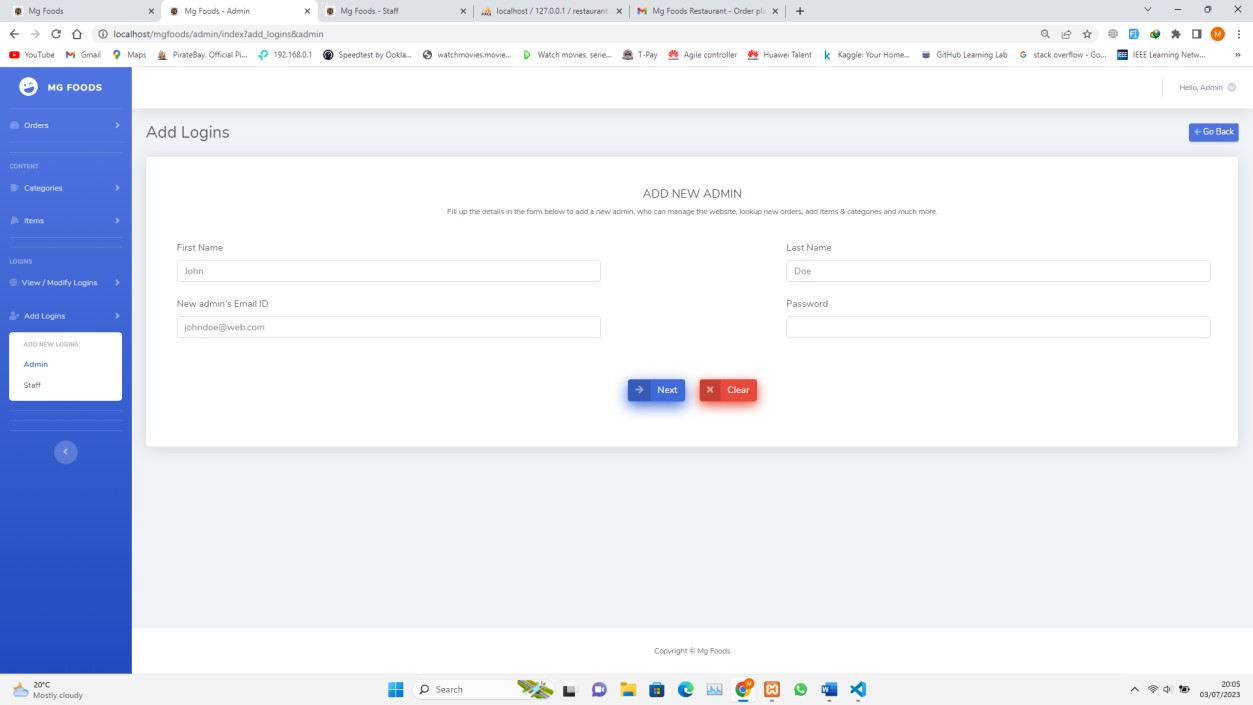
# Categories

# Items



# View/Modify Logins

# Add Logins

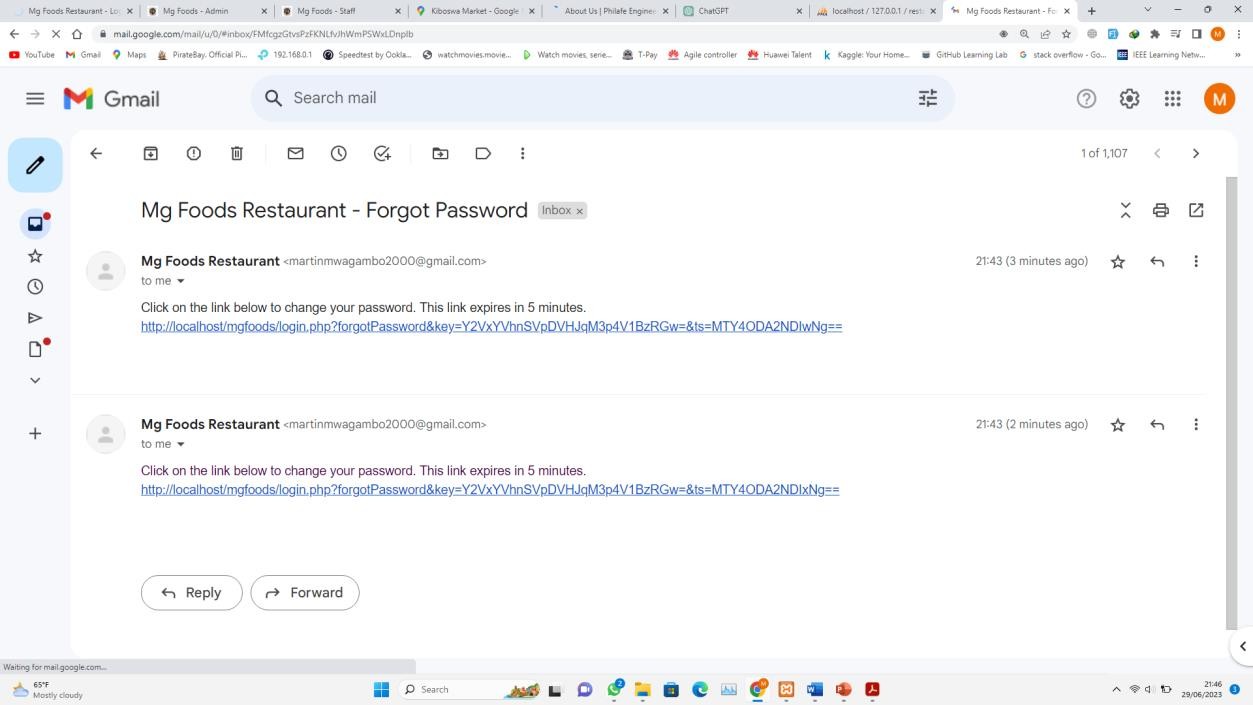


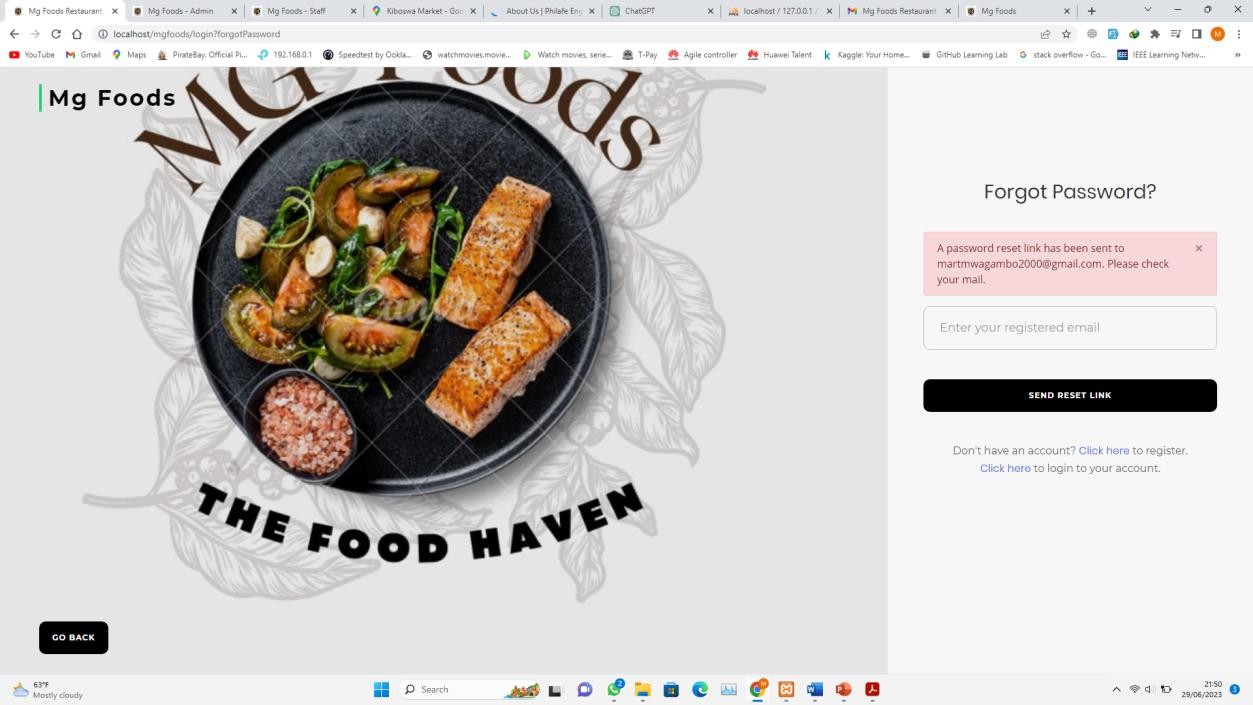
# Staff Login

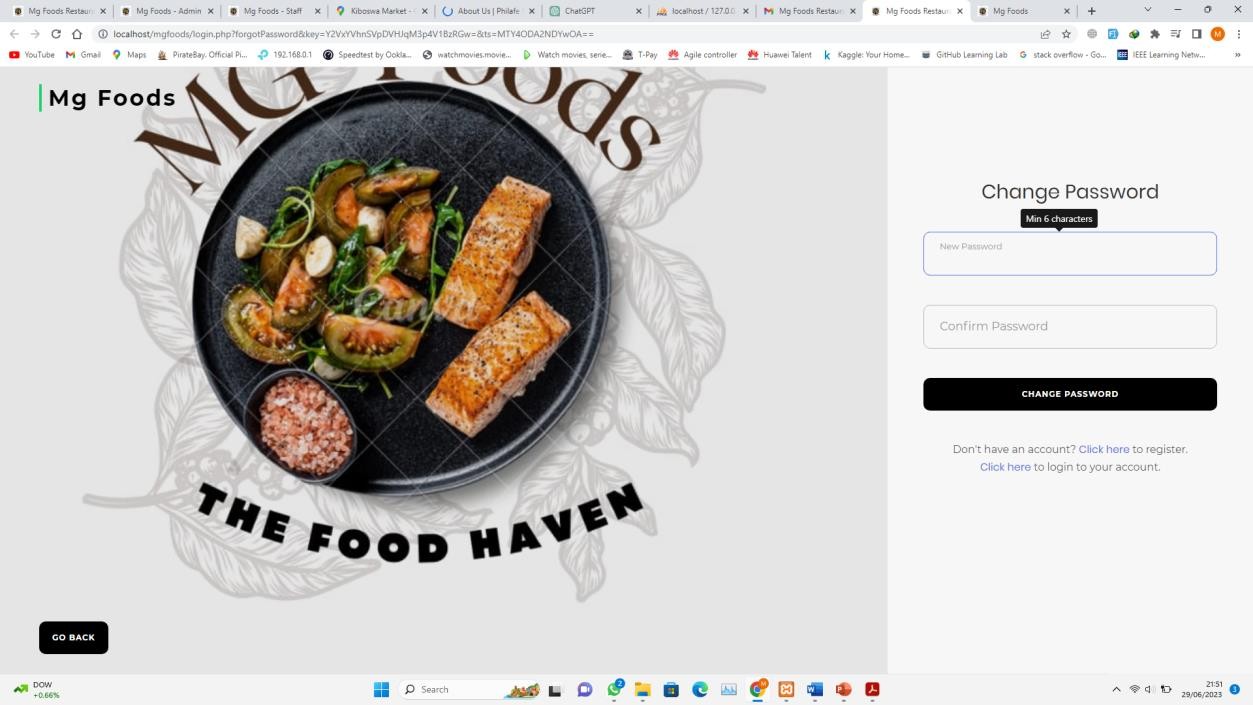
# Staff Dashboard

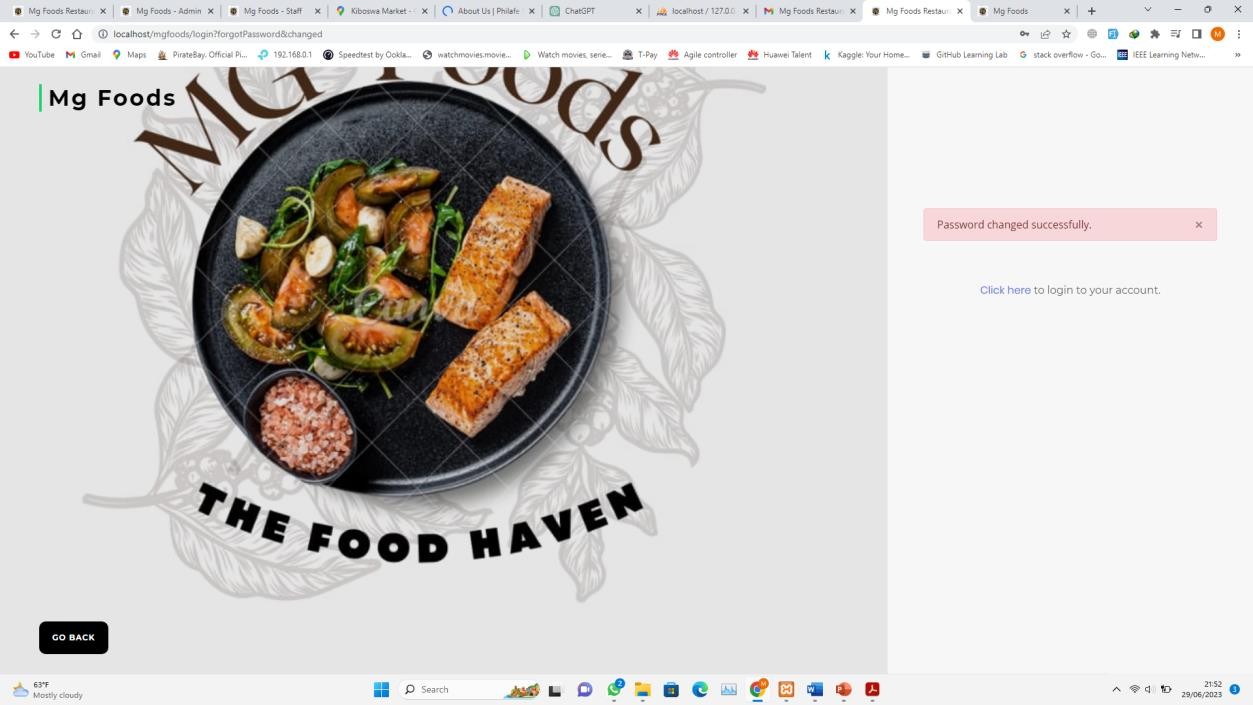
# Forgot Password

# Email Verification

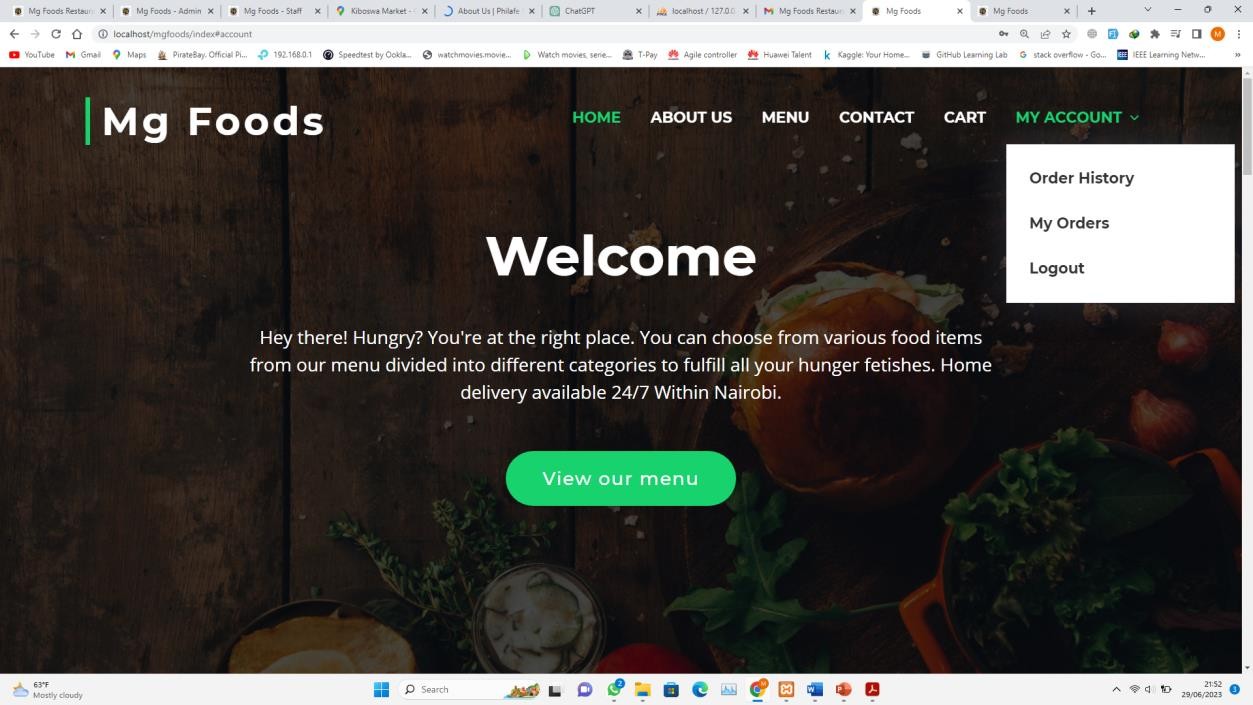


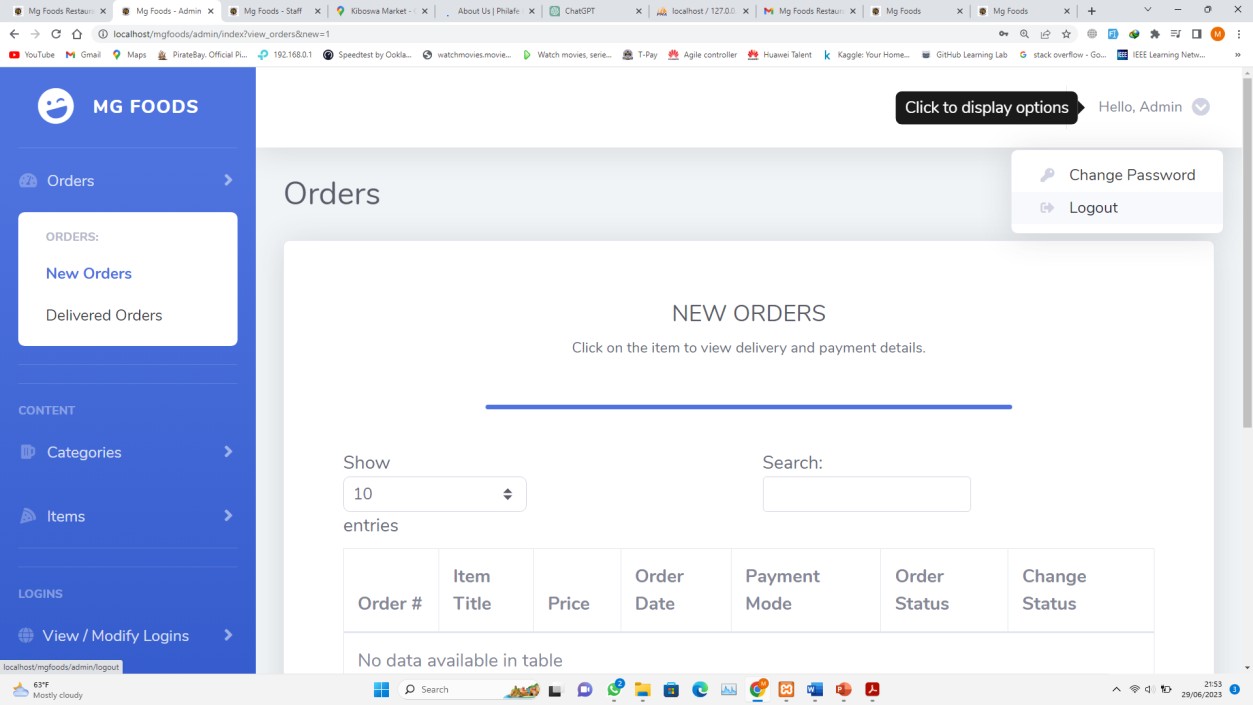




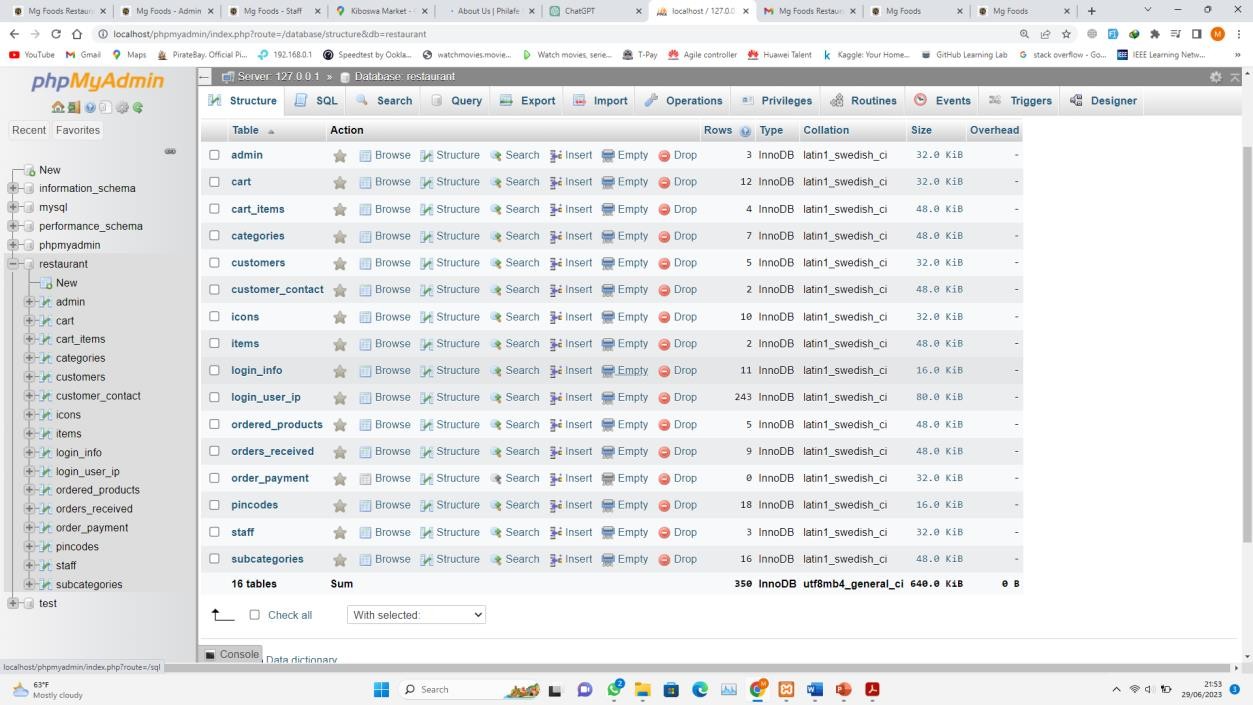


# Log Out





# Data Sources



# Threats

This are the threats that may occur on the system either internal or external and they could be accidental or intentionally.

# Accidental threats to the system include:

Disclosure of the system information i.e., student details and staff details. Errors like Transposition errors i.e., instead of SO5 one inputs 5OS Environmental threats such as humidity.

# Intentional threats include:

Deletion of records since one can only delete one data at a time.

Computer viruses by inserting removable storage medias to the computer system. Destruction of computers i.e., Touching the monitor.

**Control Measures to the threats** Installing dehumidifiers in the room Setting up rules when using the system. Using anti-virus i.e., Smadav

Data backups in the storage media requested.

Installing carbon dioxide fire extinguishers in case of an emergency fire since water can cause more damage to the computer.

# Contact Support

For any inquiries about the system or system maintenance, updates, modification or faults contact:

***Martin Mwagambo Phone:0792453373***

***Email:*** [***martmwagambo2000@gmail.com***](mailto:martmwagambo2000@gmail.com)

# PROJECT SCHEDULE

This will show the tasks that will be carried out in the course of carrying the project.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Task No** | **Description** | **Task**  **No of Hours** | **Planned Start Date** | **Actual Start Date** | **Planned**  **Completion Date** | **Actual**  **Completion Date** | **Deliverables** |
| **1** | Ideation | 1 day | 22/01/2023 | 22/01.2023 | 22/01/2023 | 22/01/2023 | Idea  Approved |
| **2** | Research  Proposal | 7 days | 23/01/2023 | 24/01/2023 | 30/01/2023 | 28/01/2023 | Proposal |
| **3** | Proposal  Presentation | 1 day | 3/02/2023 | 08/02/2023 | 3/02/2023 | 08/02/2023 | Proposal  Presentation |
| **4** | System requirements  specifications | 4  weeks | 7/02/2023 | 09/02/2023 | 7/03/2023 | 28/03/2023 | SRS |
| **5** | System Design  Specifications | 12  weeks | 7/03/2023 | 29/03/2023 | 29/05/2023 | 08/04/2023 | SDS |
| **6** | System  Implementation | 3  Weeks | 29/05/2023 | 08/05/2023 | 19/06/2023 | 24/05/2023 | Implementati  on Plan |
| **7** | Testing | 7  Weeks | 19/06/2023 | 24/05/2023 | 3/07/2023 | 04/07/2023 | Testing Plan and User  manual |
| **8** | Final Documentation | 3  weeks | 3/07/2023 | 04/07/2023 | 24/07/2023 |  | Final Documentati  on |

# APPENDIX

# Glossary

SRS- System Requirements Specification SDS- System Design Specification

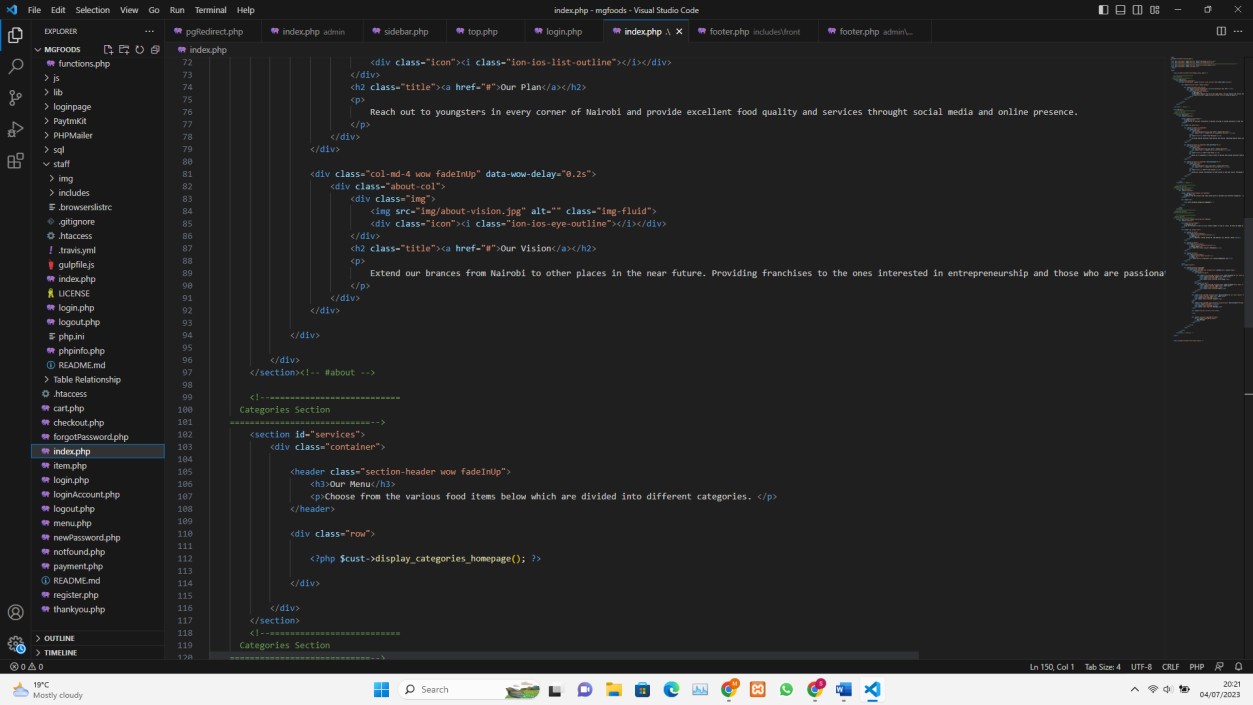
Front end – The login interface that user interacts with.

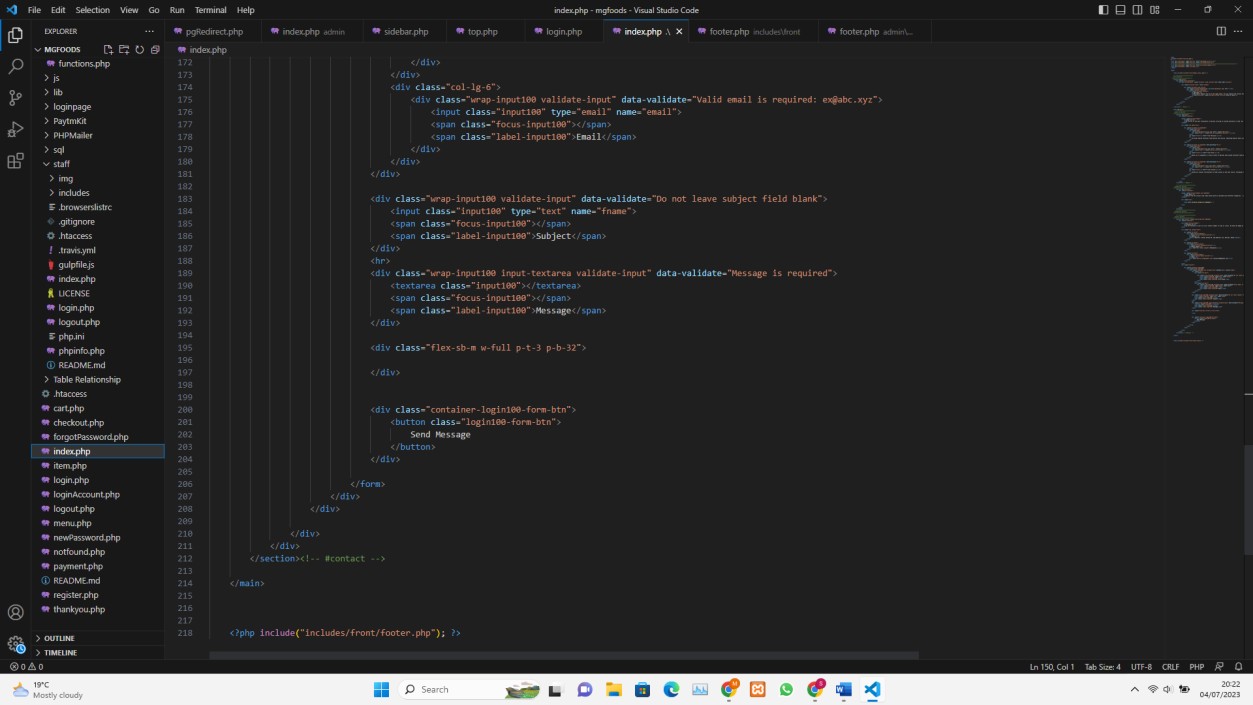
Back end-Program that operates the system and cannot be accessed by the user.

# Code

Below are some sample codes that entail different modules of the system to how they meet the user requirements

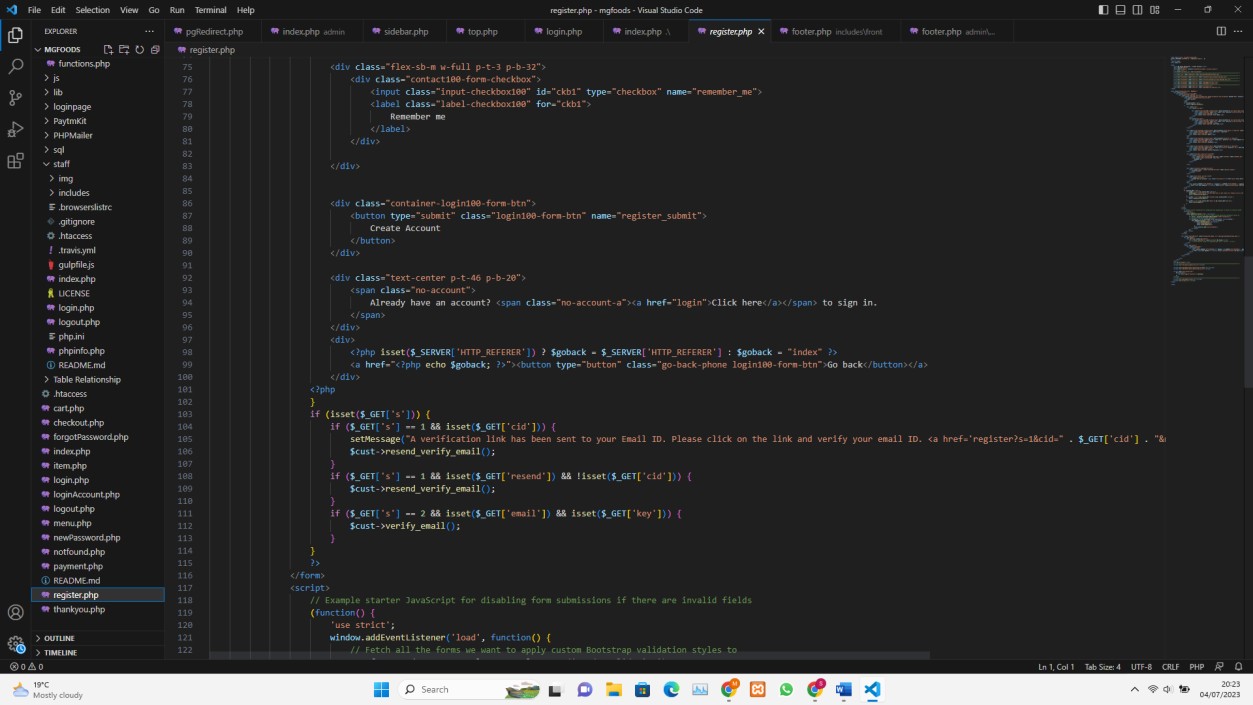
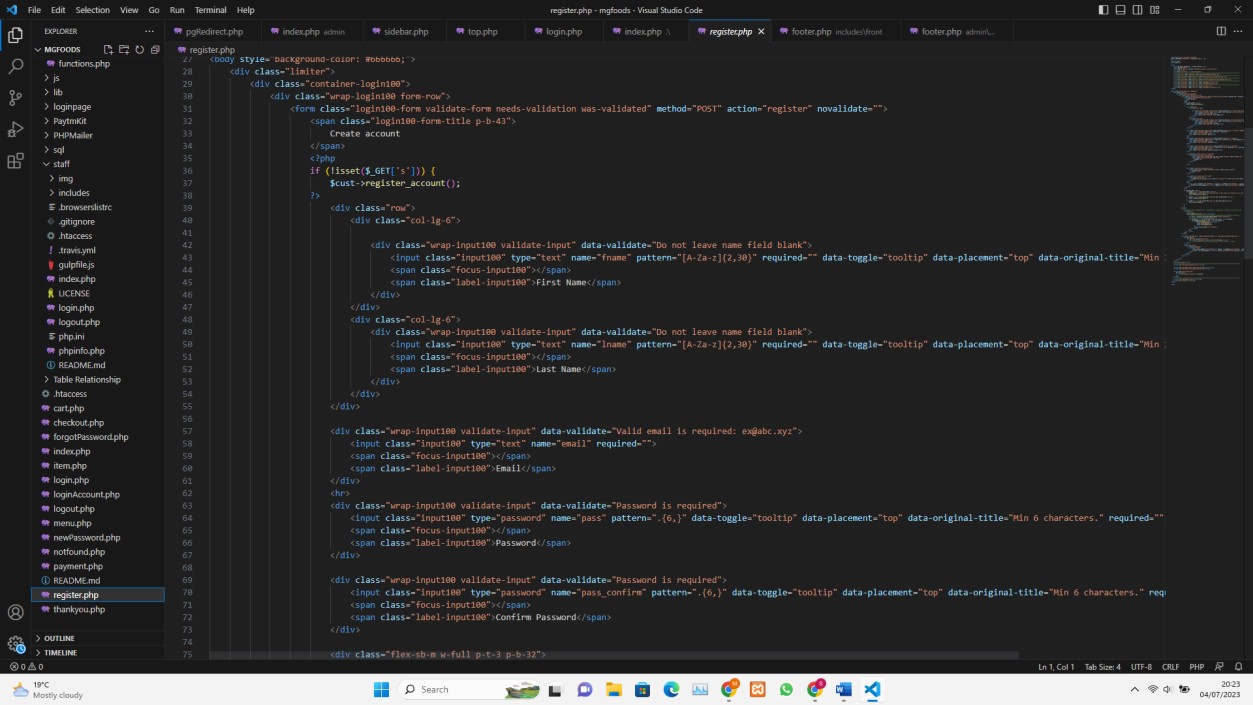
# Sample Code for Home Page

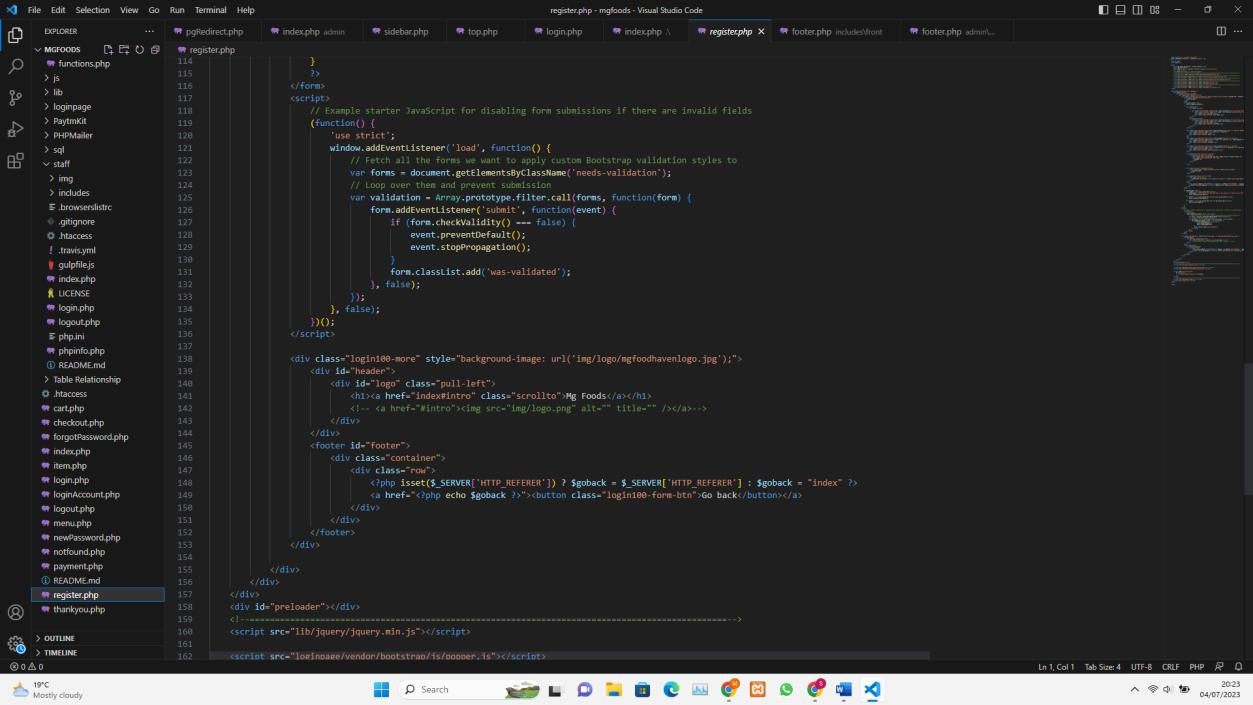




# Sample Code for Login Page

# Sample Codes for Register Page





# REFERENCES

* + - Beynon-Davies, P., Carne, C., Mackay, H. and Tudhope, D., 1999. Rapid application development (RAD): an empirical review. European Journal of Information Systems, 8(3), pp.211-223.Online, E. (2016, May 11).
    - Stephen Mburu, Geoffery Chemwa.: Longhorn Secondary Computer Studies Form 4 and 3 First published 2005, Corrected 2008, Reprinted 2015.
    - Zak, D., 2015. Programming with Microsoft Visual Basic 2015. Cengage Learning.
    - Garcia-Molina, H., Ullman, J.D. and Widom, J., 2000. Database system implementation (Vol. 672). Upper Saddle River, NJ: Prentice Hall.
    - Hathaway, T. and Hathaway, A., 2016. Data Flow Diagrams–Simply Put! Process Modeling Techniques for Requirements Elicitation and Workflow Analysis.
    - Ahituv, N., Neumann, S. and Zviran, M., 2002. A system development methodology for ERP systems. Journal of Computer Information Systems, 42(3), pp.56-67.
    - Avison, D. and Fitzgerald, G., 2003. Information systems development: methodologies, techniques and tools. McGraw-Hill.
    - Dennis, A., Wixom, B.H. and Roth, R.M., 2012. Systems Analysis and Design 5th Edition, 5th Edition.
    - Lee, B.S., 1995. Normalization in OODB design. ACM Sigmod Record, 24(3), pp.23- 27.
    - Li, Q. and Chen, Y.L., 2009. Entity-relationship diagram. In Modeling and Analysis of Enterprise and Information Systems (pp. 125-139). Springer, Berlin, Heidelberg.