**MINI PROJECT REPORT ON**

**SUPERMARKET MANAGEMENT SYSTEM**

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE IN

**BACHELOR OF COMPUTER APPLICATIONS OF MAHATMA GANDHI UNIVERSITY**

**KERALA**

***Submitted by***

**RISWANA PARVIN K S**

**REGISTER NO :220021089912**

*Under the guidance*

*of*

### Dr. MUNAVAR JASIM.K



**DEPARTMENT OF COMPUTER APPLICATIONS (2022-2025)**

**M.E.S COLLEGE, MARAMPALLY ALUVA -5**

**M E S COLLEGE MARAMPALLY**

**ALUVA -5**

**DEPARTMENT OF COMPUTER APPLICATIONS**



***Certificate***

This to certify that the report entitled

**supermarket management system** Has been submitted by

RISWANA PARVIN K S

ROLL NO: 210159

In partial fulfillment of the award of the degree in

**BACHELOR OF COMPUTER APPLICATION**

**OF**

**MAHATMA GANDHI UNIVERSITY**

During the academic year 2024-2025

Register No: 220021089912

Project Guide Head of the Department Submitted for the examination held on ……………………………….

Examiners 1.

2.

**ACKNOWLEDGEMENT**

At the very outset I am very grateful to God almighty for his blessings showed upon us to complete this project. I hereby express my sincere thanks and gratitude to **Prof** **Dr. J P Thavamani**, Principal of MES College Marampally for her kind support. I hereby express my sincere thanks and gratitude to **Dr. Leena C Sekhar**, Head of the Department of Computer Application MES College Marampally for her kind support throughout the course of the project

I also express my sincere thanks to **Dr.munavar jasim.k**, Department of Computer Applications, MES College Marampally for her co-operation and Guidance throughout the course of the project work.

I also express my sincere thanks to our all faculty members of the Department of Computer

Applications, MES Collage for their timely suggestion and encouragement.

Finally, I express my heartfelt gratitude to my parents, friends, well-wishers and all who have helped me in completing this project and make this work satisfactory.

##### RISWANA PARVIN K S

**Contents**

1. [Introduction 01](#_TOC_250043)
   1. [Overview of the system 02](#_TOC_250042)
2. Requirement and Analysis 03
   1. [Problem definition 04](#_TOC_250041)
   2. [Select the software development model. 04](#_TOC_250040)
   3. [Requirement specification 05](#_TOC_250039)
      1. [Justification of proposed system 05](#_TOC_250038)
      2. [Benefits of Proposed System 06](#_TOC_250037)
   4. [Project planning 07](#_TOC_250036)
   5. [Project scheduling 08](#_TOC_250035)
   6. [Feasibility Study 09](#_TOC_250034)
      1. Economic Feasibility 09
      2. Technical Feasibility 09
      3. Operational Feasibility 09
3. [Software Requirement Specification (SRS) 10](#_TOC_250033)
   1. [Introduction 11](#_TOC_250032)
      1. [Purpose 11](#_TOC_250031)
      2. [Scope 12](#_TOC_250030)
      3. Definitions, acronyms, and abbreviations 12
      4. References 13
      5. Overview 13
   2. [Overall description 13](#_TOC_250029)
      1. [Product perspective 13](#_TOC_250028)
      2. [Product functions 13](#_TOC_250027)
      3. [User characteristics 14](#_TOC_250026)
      4. [Constraints 14](#_TOC_250025)
      5. [Assumptions and dependencies 15](#_TOC_250024)
   3. [Specific requirements 15](#_TOC_250023)
      1. [External interfaces 15](#_TOC_250022)
      2. [Functional requirements 15](#_TOC_250021)
      3. [Performance requirements 15](#_TOC_250020)
      4. [Logical database requirements 16](#_TOC_250019)
      5. [Design constraints 16](#_TOC_250018)
      6. [Software system attributes 16](#_TOC_250017)
      7. Organising the specific requirements 17
4. [Software and hardware requirement 18](#_TOC_250016)
   1. [Software requirements 19](#_TOC_250015)
   2. Hardware requirements 19
5. [System Design 20](#_TOC_250014)
   1. [Introduction 21](#_TOC_250013)
   2. [Input design 21](#_TOC_250012)
   3. [Output design 22](#_TOC_250011)
   4. Number of module description 23
   5. [Functional Diagram 24](#_TOC_250010)
   6. Database design 26
   7. E R Diagram 31
   8. Data Flow Diagram 32
6. [System Development 35](#_TOC_250009)
   1. [Process Description 36](#_TOC_250008)
   2. [Pseudo code 36](#_TOC_250007)
7. [System Implementation 51](#_TOC_250006)
   1. [Testing 52](#_TOC_250005)
   2. [System Implementation 53](#_TOC_250004)
   3. [Security 53](#_TOC_250003)
8. [Conclusion 54](#_TOC_250002)
9. [Appendix 56](#_TOC_250001)
   1. Sample Inputs 57
   2. Sample Outputs 59
10. [Bibliography 62](#_TOC_250000)

1

# INTRODUCTION

## Overview of the system

The mini project aims to develop an online platform for supermarket management,focusing on product visibility and inventory management.The platform will serve administrators,staff,and users(shoppers). Key features will include user authentication,product listing,inventory tracking,and an intuitive interface for browsing products.Administrators will manage inventory and oversee staff,while staff will handle inventory updates and product information.users can explore product categories,view detailed product information,and check availability.The goal is to create an efficient system that enhances product visibility and simplifies supermarket management

3

# REQUIREMENT ANALYSIS

## Problem definition

In today's competitive retail environment, supermarkets face significant challenges in effectively managing their inventory and ensuring optimal product visibility. Traditional inventory management systems often lead to inefficiencies, including stock discrepancies, overstocking, and stockouts, which can frustrate customers and hinder sales. These issues arise from a lack of real-time data and ineffective communication between different stakeholders. Administrators, staff, and shoppers often operate in silos, leading to miscommunication regarding product availability and inventory status.

Additionally, customers frequently encounter difficulties in locating products or accessing detailed information, which diminishes their overall shopping experience. This not only limits opportunities for upselling and cross-selling but can also drive customers to competitors who offer more intuitive shopping platforms. Furthermore, many existing systems lack user-friendly interfaces, making navigation cumbersome and resulting in potential lost sales opportunities.

The primary objective of the proposed online supermarket management platform is to tackle these pressing issues head-on by creating a centralized, user-friendly system that enhances product visibility and streamlines inventory management. The platform will allow staff to efficiently update inventory levels and provide administrators with robust tools for overseeing stock and staff. By facilitating seamless communication among all users, the platform aims to eliminate confusion and improve operational efficiency. Ultimately, by offering an intuitive interface that caters to users of all technological backgrounds, the platform seeks to enhance the overall shopping experience, increase customer satisfaction, and drive growth for the supermarket. Through these improvements, the platform will position the supermarket as a leader in the retail space, adapting to the evolving needs of both customers and staff.

**2.2.Select the software development model**

I select the iterative enhancement model for my project as the system development tool. The iterative enhancement model in software engineering

combines elements of the linear sequential model with the iterative philosophy of prototyping.

In this model, the software is broken down into several modules which are incrementally developed and delivered. Firstly, the development team develops the core module of the system. After that, it is refined into increasing levels of capacity of adding new functionalities in successive versions.

## Requirement specification

##### Existing system

The current supermarket management systems predominantly operate on a combination of manual processes and outdated software solutions. Many supermarkets still rely heavily on manual inventory management practices, often utilizing spreadsheets or physical stock counts to track inventory levels. This reliance on manual updates can lead to significant human errors, resulting in discrepancies between recorded and actual stock levels. Consequently, this inefficiency often results in stockouts or overstock situations, frustrating both staff and customers. The time-consuming nature of manual inventory management detracts from more value-added tasks, such as customer service and sales optimization.

Moreover, existing systems frequently provide limited product visibility to customers. Shoppers often struggle to locate specific items within the store due to inadequate online product listings and poor search functionalities. When product information is not easily accessible, it can lead to a suboptimal shopping experience, reducing the likelihood of repeat customers. Additionally, without real-time updates on product availability, customers may encounter frustrations such as arriving at the store only to find out that the item they wanted is out of stock.

Another significant limitation of existing supermarket management systems is the lack of real-time data capabilities. Many systems do not provide timely insights into inventory levels, sales trends, or product performance. This absence of real-time data complicates decision-making processes for staff and management, leading to missed opportunities for promotions or restocking initiatives. Decisions based on outdated information can significantly impact customer satisfaction and operational efficiency.

Communication within existing systems is often fragmented, leading to a disconnect between administrators, staff, and users. Staff may not have immediate access to vital information regarding inventory status or customer inquiries, resulting in delays and miscommunication. Furthermore, customers may not receive timely updates about promotions or new products, diminishing their shopping experience.

Existing systems are also constrained by technological limitations. Many supermarkets continue to operate on legacy software that lacks modern features such as cloud-based capabilities, mobile access, and seamless integration with third-party services. These outdated systems not only require considerable maintenance efforts but also lack user-friendly interfaces, making them cumbersome for both staff and customers. Additionally, the security of sensitive user and financial data may not be adequately addressed, exposing supermarkets to vulnerabilities that could result in data breaches and loss of customer trust.

the existing supermarket management systems present numerous challenges that hinder efficiency, accuracy, and customer satisfaction. By identifying and understanding these limitations, the development of a new Supermarket Management System can directly address these issues, providing a comprehensive, user-friendly solution that enhances inventory management, improves product visibility, and fosters better communication among all stakeholders. The new system aims to leverage modern technology to create a scalable, secure, and efficient platform that meets the evolving needs of supermarkets in a competitive retail environment.

## Justification of proposed system

The proposed Supermarket Management System (SMS) addresses the numerous inefficiencies and challenges faced by existing supermarket management practices. By leveraging modern technology, the SMS aims to streamline operations, enhance customer experiences, and provide robust inventory management tools that traditional systems lack. One of the primary justifications for implementing the SMS is its ability to automate inventory management processes. With real-time tracking capabilities, the system will minimize human errors associated with manual stock counting and updates. This automation not only improves accuracy but also ensures that inventory levels are always up-to-date, reducing the risk of stockouts and overstock situations that can frustrate customers and negatively impact sales.

Another significant justification for the SMS is its enhanced product visibility features. By providing a user-friendly interface for browsing products online, customers can easily access detailed product information, including descriptions, prices, and availability. This level of transparency fosters trust and satisfaction, encouraging customers to engage with the platform more frequently. Furthermore, an intuitive search function will allow users to locate specific items quickly, significantly improving their shopping experience. The ability to check product availability in real time will also reduce customer frustration and enhance overall satisfaction.

The SMS also addresses the critical need for real-time data analytics and reporting capabilities. In contrast to existing systems that often rely on outdated information, the proposed system will provide management with instant access to key performance metrics, such as sales trends, inventory turnover, and customer preferences. These insights will empower supermarket managers to make informed, data-driven decisions regarding inventory purchases, promotions, and staffing needs. The ability to respond swiftly to changing market demands will ultimately lead to increased efficiency and profitability.

Additionally, the proposed SMS will foster improved communication and collaboration among staff, administrators, and customers. The platform will enable seamless information sharing, ensuring that staff members are always informed about inventory changes and customer inquiries. This enhanced communication will facilitate a more cohesive working environment and enable staff to provide better service. For customers, the system will deliver timely updates on promotions, new products, and other important information, further enhancing their shopping experience.

Security is another critical aspect that the proposed SMS will address. Unlike many existing systems that lack robust security measures, the new platform will implement industry-standard encryption and secure data storage practices to protect sensitive customer and financial information. By prioritizing data security, the SMS will help build trust with customers, reassuring them that their information is safe and secure.

the justification for implementing the proposed Supermarket Management System lies in its ability to overcome the limitations of current systems, streamline operations, enhance customer satisfaction, and provide critical insights for better decision-making. By integrating automation, real-time data, improved communication, and enhanced security features, the SMS will position supermarkets to thrive in a competitive landscape, ultimately leading to increased efficiency, profitability, and customer loyalty.

## Benefits of Proposed System

The implementation of the proposed Supermarket Management System (SMS) offers a multitude of benefits that can significantly enhance the overall efficiency and effectiveness of supermarket operations. One of the most immediate advantages is the automation of inventory management processes. By providing real-time tracking and updates, the SMS minimizes human errors associated with manual inventory counts. This automation leads to more accurate stock levels, reducing the likelihood of stockouts and overstock situations. As a result, supermarkets can maintain optimal inventory levels, ensuring that popular items are always available while minimizing excess stock that ties up valuable resources.

Another key benefit of the SMS is its ability to improve customer experience. With a user-friendly interface for browsing products online, customers can easily access detailed product information, including descriptions, prices, and availability. This enhanced product visibility fosters a sense of transparency and trust, encouraging customers to engage more frequently with the platform. Additionally, the system's intuitive search functionality allows users to quickly find specific items, making their shopping experience more efficient and enjoyable. By reducing friction in the shopping process, the SMS can help increase customer satisfaction and loyalty, ultimately driving higher sales.

The proposed system also enables data-driven decision-making through advanced analytics and reporting features. Supermarket managers will gain access to real-time insights into sales trends, inventory turnover, and customer preferences. This information is invaluable for making informed decisions about product restocking, promotional strategies, and inventory management. By understanding which products are performing well and which are not, managers can optimize their inventory and marketing efforts, leading to increased profitability and reduced waste.

In addition to enhancing operational efficiency, the SMS fosters better communication and collaboration among staff members. The system allows for seamless information sharing, ensuring that all employees are aware of inventory changes and customer inquiries. This improved communication streamlines workflows and enhances the overall working environment. When staff members are well-informed, they can provide better service to customers, which is critical for building long-term relationships and enhancing customer satisfaction.

Security is another significant benefit of the proposed SMS. With growing concerns around data privacy, the implementation of robust security measures—such as encryption and secure data storage—ensures that sensitive customer and financial information is protected. By prioritizing security, supermarkets can build trust with their customers, reassuring them that their personal and payment information is safe. This trust is essential in fostering customer loyalty and encouraging repeat business.

Lastly, the SMS is designed with scalability in mind, allowing supermarkets to easily accommodate growth and evolving business needs. As supermarkets expand their product lines or customer base, the system can adapt to handle increased traffic and inventory without compromising performance. This scalability ensures that the SMS remains a valuable asset for supermarkets in the long term, providing a flexible solution that can evolve alongside the business.

the proposed Supermarket Management System offers a wide range of benefits, including improved inventory accuracy, enhanced customer experience, data-driven decision-making, better staff communication, robust security measures, and scalability. By implementing the SMS, supermarkets can position themselves for greater operational efficiency, increased profitability, and enhanced customer loyalty, ultimately thriving in an increasingly competitive retail landscape.

## Project planning

The project has 4 months from August to November. Considering the total available time for the project, I have prepared a plan and schedule which is given below.

|  |  |  |
| --- | --- | --- |
| Sl.No | Duration | Activity |
| 1 | August | Identification of need. |
| 2 | September | Feasibility study |
| 3 | September | Analysis |
| 4 | October | Design |
| 5 | November | Testing |
| 6 | November | Implementation |

## Project scheduling

Once we have estimates of the effort and time requirement for the different

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Activity | August | September | October | November |
| Identification of  need |  |  |  |  |
| Feasibility study |  |  |  |  |
| Analysis |  |  |  |  |
| Design |  |  |  |  |
| Testing |  |  |  |  |
| Implementation |  |  |  |  |

phases, a schedule for the project can be prepared. Conceptually simple and effective scheduling techniques like calendar-oriented charts are prepared. Progress can be represented easily by ticking off each milestone when completed. Alternatively, for each activity another bar can be drawn specifying

when the activity actually started and ended, i.e., when these two milestones were achieved. Once we have estimates of the effort and time requirement for the different phases, a schedule for the project can be prepared.

## Feasibility Study

A feasibility study is a crucial step in evaluating the practicality and viability of implementing the proposed supermarket Management System. This study encompasses various aspects, including technical, operational, economic, and scheduling considerations.

#### Technical Feasibility:

System Requirements:\* Assess the technical requirements for developing and implementing the system, ensuring compatibility with existing hardware and software infrastructure.

Development Tools:\* Evaluate the availability and appropriateness of development tools and technologies required for building the system.

#### Operational Feasibility:

User Acceptance:Gauge the willingness and readiness of administrators and users to adapt to the new system through surveys, interviews, or pilot programs.

Training Needs:Identify training requirements for administrators and users to ensure a smooth transition to the new system.

#### Economic Feasibility:

Cost-Benefit Analysis:Conduct a comprehensive cost-benefit analysis, considering development costs, potential savings from process automation, and projected revenue increases.

Return on Investment (ROI): Assess the expected ROI over time to determine whether the benefits justify the initial and ongoing costs of implementing the system.

12

# SOFTWARE REQUIREMENT SPECIFICATION (SRS)

## Introduction

This Software Requirement Specification (SRS) document outlines the requirements for the Supermarket Management System (SMS). The SMS aims to streamline supermarket operations, enhance product visibility, and improve user interactions among various stakeholders, including administrators, staff, and shoppers. This document serves as a comprehensive reference for developers, stakeholders, and project managers to ensure that the final product aligns with user expectations and business goals

#### Purpose

The Supermarket Management System (SMS) is designed to revolutionize the way supermarkets operate, addressing various challenges faced in traditional management practices. Its primary purpose is to streamline and automate essential processes, enhancing efficiency and productivity across all levels of the supermarket. By incorporating features such as user authentication, the SMS ensures secure access for different user roles, including administrators, staff, and shoppers. This role-based access facilitates tailored experiences, allowing users to interact with the system in ways that align with their specific responsibilities.

One of the core functionalities of the SMS is comprehensive product management. This feature enables administrators to easily add, edit, and delete product listings, ensuring that customers have access to up-to-date information regarding prices, availability, and promotions. This transparency enhances the shopping experience, as customers can quickly find the items they need without the frustration of outdated information.

Real-time inventory tracking is another crucial aspect of the SMS. By providing staff with tools to monitor stock levels, the system minimizes the risk of stockouts and overstock situations. Automated alerts notify managers when inventory reaches critical levels, allowing for timely reordering and optimal stock management. This not only reduces waste but also ensures that popular items are always available, thereby enhancing customer satisfaction.

The SMS also focuses on user experience. With an intuitive interface designed for easy navigation, customers can effortlessly browse product categories, filter searches, and view detailed product information, including descriptions, images, and customer reviews. This ease of use is essential for engaging shoppers and encouraging repeat visits.

Moreover, the system incorporates advanced reporting and analytics capabilities, providing administrators with valuable insights into sales trends, inventory performance, and customer behavior. This data-driven approach enables informed decision-making, helping management to optimize marketing strategies, tailor promotions, and improve overall operational effectiveness.

the purpose of the Supermarket Management System is to create a cohesive, efficient, and customer-focused platform that enhances supermarket operations while providing a superior shopping experience. By leveraging technology to automate processes and improve transparency, the SMS aims to foster customer loyalty, drive sales growth, and ensure that the supermarket remains competitive in an evolving retail landscape.

##### Scope

The scope of the Supermarket Management System (SMS) encompasses a wide range of functionalities aimed at improving the efficiency and effectiveness of supermarket operations. This system is designed to serve multiple user roles, including administrators, staff, and shoppers, each with distinct capabilities tailored to their needs. For administrators, the SMS provides comprehensive tools for managing product listings, overseeing inventory, and generating detailed reports on sales and stock levels. Staff members benefit from streamlined processes for updating inventory, managing product information, and facilitating customer inquiries, thus enhancing their operational efficiency.

For shoppers, the SMS offers an intuitive interface that allows for easy browsing of product categories, searching for specific items, and accessing detailed product information, including pricing and availability. The system also includes user authentication features to ensure secure access and protect sensitive information. Additionally, the SMS incorporates real-time inventory tracking, which helps prevent stockouts and overstock situations, ultimately leading to improved customer satisfaction.

The system is designed to be scalable and adaptable, allowing for the integration of new features and functionalities as supermarket needs evolve. This includes potential future enhancements, such as loyalty programs, promotional management, and data analytics tools to further support decision-making processes. Overall, the SMS aims to create a cohesive ecosystem that not only enhances the operational efficiency of supermarkets but also significantly improves the overall shopping experience for customers.

##### References



Software Engineering: A Practitioner's Approach by Roger S. Pressman**:**

I Somerville, Software Engineering, 8th edition, Addison-Wesley, 2007



**"**Database Systems: The Complete Book" by Hector Garcia-Molina,

official documentation

Jeffrey D. Ullman, and Jennifer Widom**:**

(https://docs.flutter.dev/)

##### Overview

The document contains the detailed documentation and functions of ‘SUPERMARKET MANAGEMENT SYSTEM’ like functional requirements, constraints, dependencies etc**.** The SRS is organized as it discusses the product description first then the requirements of the products, and the dependencies that will face by the systems etc. The developer is responsible for:

* + - * Developing the system.
      * Installing the software.
      * Maintaining the system.

## Overall description

#### Product perspective

The Supermarket Management System (SMS) is designed as a web-based platform that integrates essential supermarket operations into a single, cohesive interface. Utilizing a client-server architecture, the system will provide a responsive and intuitive user experience across various devices, including desktops and mobile devices. Built with modern web technologies like HTML5 and JavaScript, and supported by a robust back-end framework such as Node.js or Django, the SMS will efficiently manage product listings, inventory control, and user interactions. It will also incorporate a relational database (e.g., MySQL or PostgreSQL) for secure data storage and allow integration with third-party services for payment processing and inventory management. This design ensures scalability, flexibility, and improved operational efficiency, ultimately enhancing the shopping experience for customers.

#### Product functions

The Product Function in a Supermarket Management System encompasses a range of essential features designed to efficiently manage inventory. It allows users to add new products with detailed information such as name, SKU, category, price, quantity, supplier, and expiry date. Existing products can be easily updated or deleted, ensuring the inventory remains current. The system provides comprehensive product listings with filtering and sorting options for quick access, alongside real-time inventory tracking that alerts users to low stock levels for timely reordering. Category and supplier management facilitate organized product categorization and relationship management with vendors. Additionally, barcode management supports streamlined checkout processes, while promotional tools enable the implementation of discounts and special offers. Reporting and analytics features help track sales trends and inventory performance, enhancing decision-making. Furthermore, a product search function allows for swift retrieval, and customer reviews and ratings can be integrated to enhance user engagement. Overall, these functionalities ensure that supermarket operations run smoothly and effectively, catering to both staff and customer needs.

##### User characteristics

In a Supermarket Management System, user characteristics are tailored to accommodate various roles and enhance operational efficiency. Typically, there are three main user types: \*\*Admins\*\*, \*\*Staff\*\*, and \*\*Customers\*\*. Admins have full access to all system features, including product management, user management, and reporting tools, allowing them to oversee operations and ensure compliance with policies. Staff members have permissions to add, update, and delete products, manage inventory, and handle transactions, enabling them to facilitate day-to-day supermarket activities efficiently. Their access is often limited to operational functions to maintain security. Customers interact with the system primarily through a user-friendly interface that allows them to browse products, view promotions, and submit reviews, enhancing their shopping experience. Each user type is equipped with a role-based access control system to ensure that sensitive data and critical functionalities are restricted appropriately, promoting both security and usability. This structure not only streamlines workflows but also fosters a collaborative environment among users, contributing to effective supermarket management.

##### Constraints

* + - * The system is user friendly.
      * User must be aware to enter correct data into local databases.

##### Assumptions and dependencies

The assumptions are that the coding should be error free. The system should be user friendly so that the users can easily access data which have more

storage capacity and provide fast access to database. Search facilities and provide quick search results. Save money and time unlike the existing system. The dependencies are that the specific hardware and software are required for the product to will run. On the basis of listing requirements and specification, the system will be developed and run. Updates are to be made correctly and data entered without any mistakes.

## Specific requirements

##### External interfaces

The system incorporates several external interfaces to enhance its functionality and connectivity within the broader educational landscape. A pivotal connection is established with a Database Management System (DBMS), facilitating efficient storage and retrieval of user profiles, course data, and system configurations.

This interaction ensures seamless user experiences and accurate representation of course information.

These external interfaces collectively contribute to the system's versatility, connectivity, and user-friendly interactions, aligning it with the evolving needs of the educational technology landscape.

##### Functional requirements

* + - * Provide the facility for the members to change their password.
      * Provide a simple and attractive interface.
      * Ensure security for the database.
      * The system should be able to handle high volumes of data.

##### Performance requirements

* + - * Web pages hosted on the local server should have a load time of less than 2 seconds.
      * Ensure compatibility with the latest versions of PHP, MySQL, and other relevant components.

##### Logical database requirements

* + - * The XAMPP environment should not exceed a memory usage of 512 MB under normal operations.
      * Apply normalization techniques to ensure the database is in at least the third normal form (3NF).
      * Design a comprehensive database schema that includes tables for users, courses, discussions, assignments, and other relevant entities.

##### Design constraints

The design of system is subject to certain constraints that shape its development and functionality. Firstly, technological constraints may arise from the compatibility requirements with existing infrastructure, influencing the selection of technologies and limiting potential integrations. Budgetary constraints also play a significant role, impacting the scale and features of the system within the allocated financial resources.

Moreover, adherence to data protection regulations imposes constraints on data storage, processing, and transmission, emphasizing the need for robust security measures. User adoption constraints may emerge from the necessity for comprehensive training programs and potential resistance to a shift from traditional education models to an online platform.

Additionally, time constraints can influence the development speed and the timely deployment of system updates. Addressing these constraints effectively requires strategic planning, resource allocation, and a careful balance between functionality, security, and user acceptance to ensure the successful implementation of the E-learning Management System.

##### Software system attributes

* Reliability: The ability of the software to perform its intended functions without failure under normal conditions.
* Performance: The responsiveness and efficiency of the software concerning speed, throughput, and resource utilization.
* Scalability: The system's ability to handle increased loads and user demands by efficiently scaling resources.
* Maintainability: The ease with which the software can be modified, updated, and extended over time.
* Usability: The user-friendliness and effectiveness of the software in facilitating user interactions.
* Security: The protection of the software and its data against unauthorized access, attacks, and data breaches.
* Portability: The ability of the software to run on different platforms and environments without modification.
* Availability: The extent to which the software is operational and accessible when needed.

##### Organizing the specific requirements

In this system the overall functionality is organized by Data flow diagrams and E-R diagrams. Based on these diagrams, data relationships and dependencies are found and a functional hierarchy is made for organizing the specific requirements.

21

# SOFTWARE AND HARDWARE REQUIREMENT

## Software Requirements

To develop the application software, we use different type of software. The software for the development has been selected based on several factors such as: Support and stability, Cost effectiveness, Development speed, Ability to create robust application least time.

The minimal software’s used for storage are:

Operating System : Windows 10 or later, Ubuntu 20.04 LTS or later

Front-End Development : HTML,CSS,JAVASCRIPT Serverscript : PHP

Back-end :MySQL 4.2 ,PHP

## Hardware Requirements

The hardware requirements for the Knowlegde sharing platform outline the necessary specifications and configurations that users' computer systems must meet to ensure optimal performance of the software. Below is an example of how these hardware requirements might be specified:

* + - Processor: Dual-core processor, 2.0 GHz or equivalent
    - Memory (RAM): 4 GB or higher
    - Storage: Minimum 20 GB of available disk space
    - Display: 1280x800 resolution or higher
    - Input Devices: Keyboard and mouse (or other pointing devices)

22

23

# SYSTEM DESIGN

## Introduction

The Supermarket Management System serves as a crucial framework for developing an efficient and user-friendly application that streamlines supermarket operations. This design encompasses various components, including user interfaces, database architecture, and system functionalities, aimed at addressing the specific needs of administrators, staff, and customers. By leveraging modern software development principles, the system is structured to facilitate seamless product management, inventory tracking, and sales processing while ensuring security and scalability. The architecture typically includes a centralized database to store product details, user information, and transaction records, along with a responsive front end that provides intuitive navigation and accessibility. Additionally, the system is designed with modularity in mind, allowing for easy integration of new features and updates as the supermarket evolves. Overall, the design aims to enhance operational efficiency, improve customer satisfaction, and provide valuable insights through reporting and analytics, laying a solid foundation for a robust supermarket management solution.

**1;Input Design**

ADD ITEM

ADD CATEGORY

ADD STAFF

## Output Design

Category

Category

Category

Category

VIEW CATEGORY

BUY

**ITEM IMAGE**

VIEW ITEMS

* 1. **Number of modules and their description**

##### User

For registered users, the module facilitates a secure and straightforward registration process, allowing individuals to create accounts and personalize their profiles. Once logged in, users can effortlessly browse products, make purchases, manage their shopping carts, and edit their profiles as needed. The emphasis is on providing an intuitive interface, ensuring users of varying technical proficiencies can navigate the system with ease.

Non-logged-in users are also accommodated through the module, allowing them to access essential product information, view promotions, and check common inventory details on the homepage without the need for authentication. This inclusive approach enhances accessibility, providing quick information to individuals who may not have user accounts but are interested in exploring available products and offers.

##### Admin

This module encompasses functionalities that allow administrators to seamlessly manage products, sales, and user activities within the supermarket, ensuring a streamlined and organized workflow.

Within the Admin Module, administrators have the capability to add and define key parameters for products, such as names, prices, categories, and stock levels. Inventory management is facilitated through functionalities that enable administrators to update product information and track stock levels effectively. The module also provides tools for monitoring user activities, allowing administrators to oversee sales transactions, user engagement, and overall system performance through an intuitive dashboard.

Security measures are a priority within the Admin Module, featuring role-based access control that defines different levels of administrative permissions. This ensures that sensitive functionalities are accessible only to authorized personnel, enhancing the overall integrity and confidentiality of the system.

The Admin Module serves as a central hub for decision-making, equipped with reporting and analytics tools to generate insights into user behavior, sales trends, and other key performance indicators. This data-driven approach empowers administrators to make informed decisions and adapt strategies based on the evolving dynamics of the supermarket management domain.

the Admin Module is the backbone of the Supermarket Management System, providing administrators with the tools they need to efficiently manage products, monitor sales, and ensure a secure and optimized user experience.

## Functional Diagram

* + - **Admin**

View user

view

Admin

Email and password

No

Checks

Login failed

Add staff

View staff

Log out

View item

* + - **User**

User

email and password

No

Checks

Login failed

Yes

Log out

Password Varchar

social\_links Longtext

Biography Longtext role\_id

date\_added Date

#### 5.6 Database design

Table name: user

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPES** | **CONSTRAINS** | **DESCRIPTION** |
| usid | Int | Primary key,AI | User id |
| Name | Varchar | Not null | User Name |
| Email | Varchar | Not null | User Email |
| Password | Varchar | Not null | User Password |
| Address | Varchar | Not null | Address |
| Phone Num | Int | Not null | Phone num |

Table name:Category

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPES** | **CONSTRAINS** | **DESCRIPTION** |
| Category Name | varchar | Not null | Name of the category |
| Category Id | Int | Primary key | Id of the category |
| Image | Varchar | Not null | Image of Ticket |

Table name:item

last\_message\_times

Long text

tamp

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPES** | **CONSTRAINS** | **DESCRIPTION** |
| id | Int | Primary key | Order id |
| name | varchar | Not null | Name of item |
| category | varchar | Not null | Categories for items |
| Price | decimal | Not null | Price of the item |
| quantity | int | Not null | Out of quantity |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table name: login

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD** | **DATA TYPES** | **CONSTRAINS** | **DESCRIPTION** |
| email | varchar | Not null | email |
| password | varchar | Not null | password |
| usertype | Int | Not null | Type oof login position |

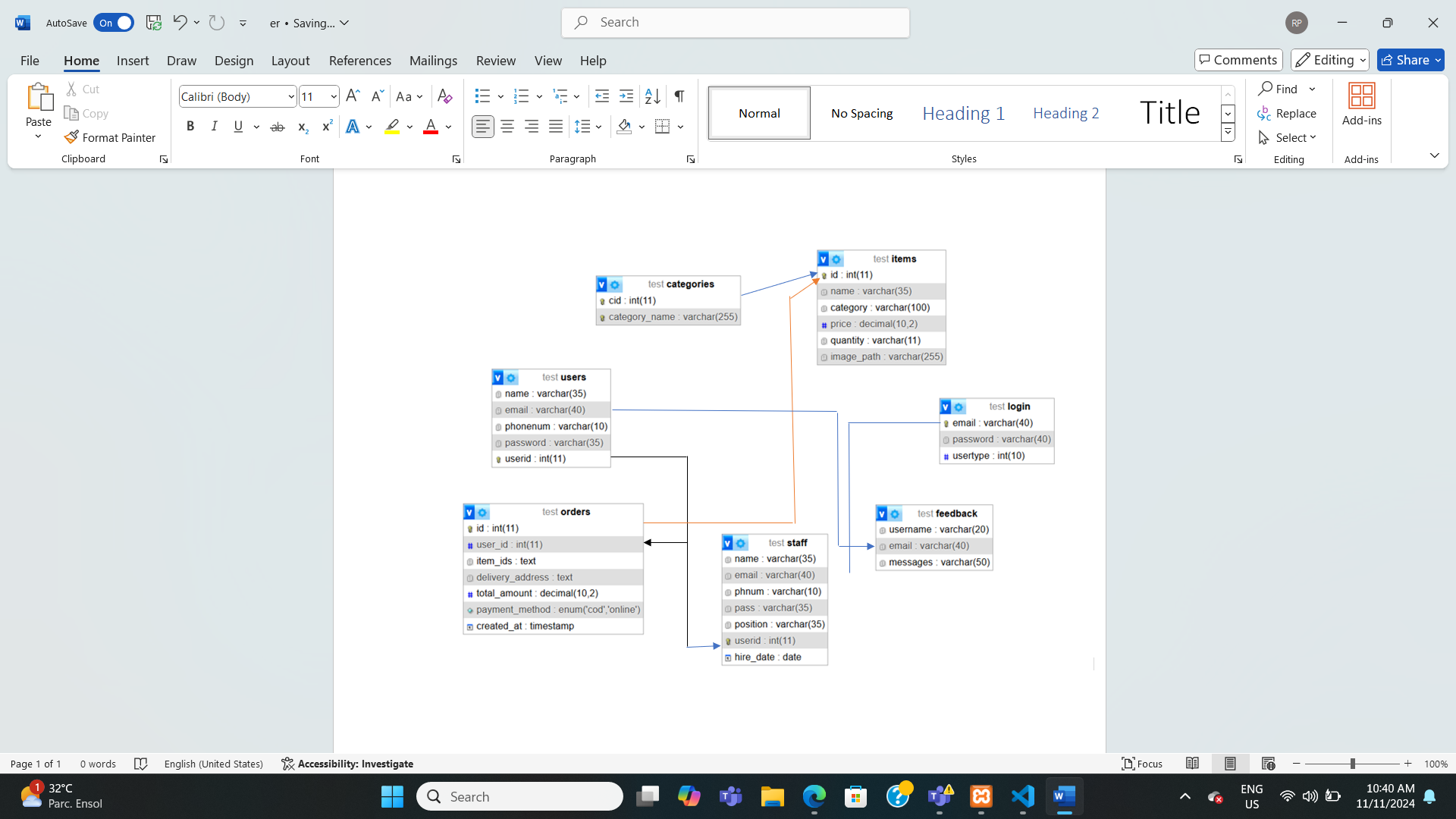
Table name: staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **FIELD** | | **DATA TYPES** | **CONSTRAINS** | **DESCRIPTION** |
| name | | varchar | Not null | Name of staff |
| email | | varchar | Not null | email |
| phnum | | Int | Not null | Phone number |
| pass | | varchar | Not null | Password |
| position | | varchar | Not null | Fix position |
| userid | | Int | Primary key | id |
|  |

**5.7.**

**ER diagram**

33



### Data flow diagram

sLevel 0

admin

staff

user

admin

Response

Response Response

**Level 1**

* + - Admin

Email,

User id,

Password

User

authenticate

Manage user

staff id

Manage

staff

CStaff Details

litem\_id

Manage

items

item details

Booking\_idid\_id

Manage bookings

bookingt details

user\_id

logout

User details

Admin

Email, Password

Login

Staff

Item

ordes

User

* + - user

Email,

User id,

Password

User

authenticate

Manage

user

Category\_id

buy

Category details

itemt\_id

cart

Ritem detailsls

User

Email, Password

Login

category

Result

38

staff

Email,

User id,

Password

User

authenticate

Manage

user

Category\_id

category

Category details

itemt\_id

View items

Ritem detailsls

staff

Email, Password

Login

category

Result

View bookings

# SYSTEM DEVELOPMENT

#### Process Description

The system development process for a Supermarket Management System follows a structured approach to ensure effective design, implementation, and maintenance. It typically begins with requirements gathering, where stakeholders, including supermarket managers and staff, identify their needs and expectations. This phase is followed by system analysis, where the gathered requirements are analyzed to define functional and non-functional specifications. In the design phase, the architecture of the system is developed, outlining database structures, user interfaces, and module interactions to create a blueprint for development. The implementation phase involves coding the system using suitable programming languages and frameworks, followed by rigorous testing to identify and resolve any bugs or issues, ensuring functionality and performance meet the established requirements. After thorough testing, the system is deployed in a live environment, accompanied by user training and documentation to facilitate a smooth transition. Finally, the maintenance phase ensures ongoing support, updates, and enhancements, allowing the system to adapt to changing business needs and technological advancements. This iterative process emphasizes collaboration, flexibility, and continuous improvement to deliver a robust and efficient supermarket management solution.

#### Pseudo code

##### home.php

1. <!DOCTYPE html>
2. <html lang="en">
3. <head>
4. <meta charset="UTF-8">
5. <meta name="viewport" content="width=device-width, initial-scale=1.0">
6. <link rel="stylesheet" href="index.css">
7. <title>Index</title>
8. </head>
9. <body>
10. <nav class="navbar">
11. <h1>Super Market Management sytem </h1>
12. <div class="Nav\_menus">
13. <a href="index.html"><div class="menus">Home</div></a>
14. <a href="Login.php"><div class="menus">Login</div></a>
15. <a href="Register.php"><div class="menus">Register</div></a>
16. \       </div>
17. </nav>
18. <section class="Home\_Section">
19. </section>
20. </body>
21. </html>

##### login.php

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" href="Login.css">

    <title>Login</title>

</head>

<body>

    <nav class="navbar">

        <h1>Super Market Management Sysytem </h1>

        <div class="Nav\_menus">

            <!-- <div class="menus">Home</div>

            <div class="menus">Login</div> -->

            <a href="index.html"><div class="menus">Home</div></a>

            <a href="Register.php"><div class="menus">Register</div></a>

        </div>

    </nav>

    <section class="Home\_Section">

            <form class="Logincontainer" action="" method="post">

                <h1>Login</h1>

                <input class="Logininput1" type="email" placeholder="Email" name="email">

                <input class="Logininput1" type="password" placeholder="Password" name="pass">

                <input class="Logininput2" type="submit" value="Login" name="submit">

            </form>

    </section>

</body>

</html>

<?php

$conn=mysqli\_connect("localhost","root","","supermarket\_management\_system");

if(!$conn){

    echo "DB connection failed";

}

if(isset($\_POST['submit'])){

    $pass=$\_POST['pass'];

    $email=$\_POST['email'];

    $sql="SELECT \* FROM `login` WHERE `email`='$email' AND  `password`='$pass'";

    echo "$sql";

    $data=mysqli\_query($conn,$sql);

    if($data){

        $row=mysqli\_num\_rows($data);

        if($row > 0){

            $value=mysqli\_fetch\_assoc($data);

            if($value['usertype'] == 0){

            header('Location: userdash.html');

            exit();

            }

            else if($value['usertype'] == 1){

                header('Location: staffdash.html');

                exit();

            }

            else{

                header('Location: Admindash.php');

                exit();

            }

        }

        else{

            echo "user not found";

        }

    }

}

?>

##### sign\_up.php

* <!DOCTYPE html>
* <html lang="en">
* <head>
* <meta charset="UTF-8">
* <meta name="viewport" content="width=device-width, initial-scale=1.0">
* <link rel="stylesheet" href="Login.css">
* <title>Login</title>
* </head>
* <body>
* <nav class="navbar">
* <h1>Super Market Management Sysytem </h1>
* <div class="Nav\_menus">
* <!-- <div class="menus">Home</div>
* <div class="menus">Login</div> -->
* <a href="index.html"><div class="menus">Home</div></a>
* <a href="Login.php"><div class="menus">Login</div></a>
* </div>
* </nav>
* <section class="Home\_Section">
* <form class="Logincontainer" action="" method="post">
* <h1>Register</h1>
* <input class="Logininput1" type="text" placeholder="Name" name="name">
* <input class="Logininput1" type="email" placeholder="Email" name="email">
* <input class="Logininput1" type="number" placeholder="Phone Number" name="phnum">
* <input class="Logininput1" type="password" placeholder="Password" name="pass">
* <input class="Logininput1" type="password" placeholder="Confirm Password" name="cnfpass">
* <input class="Logininput2" type="submit" value="Register" name="submit">
* </form>
* </section>
* </body>
* </html>
* <?php
* $conn=mysqli\_connect("localhost","root","","supermarket\_management\_system");
* if(!$conn){
* echo "DB connection failed";
* }
* if(isset($\_POST['submit'])){
* $name=$\_POST['name'];
* $email=$\_POST['email'];
* $phnum=$\_POST['phnum'];
* $pass=$\_POST['pass'];
* $cnfpass=$\_POST['cnfpass'];
* $type=0;
* if($pass == $cnfpass){
* $sql="INSERT INTO `users`(`name`, `email`, `phonenum`, `password`) VALUES ('$name','$email','$phnum','$pass')";
* $data=mysqli\_query($conn,$sql);
* $sql1="INSERT INTO `login`(`email`, `password`, `usertype`) VALUES ('$email','$pass','$type')";
* $data1=mysqli\_query($conn,$sql1);
* if($data && $data1){
* echo "<script>
* alert('Registration Successfull')
* </script>";
* header('Location: Login.php');
* }
* else{
* echo "<script>alert('Registration Failed')</script>";
* }
* }
* }
* ?>

##### users.php

* <html>
* <head>
* <title>Manage Users</title>
* <link rel="stylesheet" href="manageuser.css">
* </head>
* <body>
* <nav class="ViewCandidatesNav">
* <h1 class="ViewCandidatesNavHeading">view users</h1>
* <div class="ViewCandidatesNavContainer">
* <a href="admindash.html">Home</a>
* </div>
* </nav>
* <div class="ViewCandidatesBodyContainer">
* <?php
* $conn = mysqli\_connect("localhost", "root", "", "supermarket\_management\_system");
* if(!$conn){
* echo "Database not connected";
* }
* $sql = "SELECT \* FROM `users` ";
* $data=mysqli\_query($conn,$sql);
* if(mysqli\_num\_rows($data)>0){
* echo "<table border=1 >";
* echo "<tr>";
* echo "<th>name</th>";
* echo "<th>email</th>";
* echo "<th>Phonenum</th>";
* echo "<th>Userid</th>";
* echo "</tr>";
* while($row=mysqli\_fetch\_assoc($data)){
* $email = $row['email'];
* echo "<tr>";
* echo "<td>".$row['name']."</td>";
* echo "<td>".$row['email']."</td>";
* echo "<td>".$row['phonenum']."</td>";
* echo "<td>".$row['userid']."</td>";
* echo "<td>
* <form method='POST'>
* <button value='$email' name='userdel' type='submit'>Delete</button>
* </form>
* </td>";
* echo "<td><button>Edit</button></td>";
* echo "</tr>";
* }
* echo "</table>";
* }
* ?>
* </div>
* </body>
* </html>
* <?php
* $conn = mysqli\_connect("localhost", "root", "", "supermarket\_management\_system");
* if(!$conn){
* echo "Database not connected";
* }
* if(isset($\_POST['userdel'])){
* $email = $\_POST['userdel'];
* if(!empty($\_POST['userdel'])){
* $sql = "DELETE FROM users WHERE email='$email'";
* $data = mysqli\_query($conn, $sql);
* $sql1 = "DELETE FROM login WHERE email='$email'";
* $data1 = mysqli\_query($conn, $sql1);
* echo "<script>window.location.replace('manageuser.php');</script>";
* }
* }
* ?>

##### addcategory.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial- scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<link rel="stylesheet" href="./css/home.css">

<link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">

<link rel="stylesheet" href="./css/font-awesome-4.7.0/css/font- awesome.css">

<link rel="stylesheet" href="./css/form.css">

<title>Dashboard</title>

</head>

<body>

<div class="title">

<a href="admin.php"> <span class="heading"> </span></a>

<a href="admin.php"> <span class="heading">Dashboard</span></a>

<a href="logout.php" style="color: white"><span class="fa fa-sign- out fa-2x">Logout</span></a>

</div>

<div class="nav">

<ul>

<li class="dropdown" onclick="toggleDisplay('1')">

<a href="" class="dropbtn">Lottery &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="1">

<a href="add\_lottery.php">Add Lottery</a>

<a href="manage\_lottery.php">Manage Lottery</a>

<a href="addcategory.php">Add category</a>

</div>

</li>

<li class="dropdown" onclick="toggleDisplay('2')">

<a href="#" class="dropbtn">Users &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="2">

<a href="search\_users.php">Search Users</a>

<a href="manage\_users.php">Manage Users</a>

</div>

</li>

<li class="dropdown" onclick="toggleDisplay('3')">

<a href="#" class="dropbtn">Results &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="3">

<a href="add\_results.php">Add Results</a>

<a href="manage\_results.php">Manage Results</a>

</div>

</li>

<li class="dropdown" onclick="toggleDisplay('2')">

<a href="#" class="dropbtn">Orders &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="2">

<a href="view\_order.php">View Orders</a>

<a href="view\_sales.php">View sales</a>

</div>

</li>

</ul>

</div>

<div class="main">

<form action="" method="post">

<fieldset>

<legend>Add Category</legend>

<input type="text" name="categoryname" placeholder="Category Name">

<input type="file" name="image" placeholder="Inert Image">

<input type="submit" value="Add" name="add">

</fieldset>

</form>

</div>

<?php

include('init.php'); include("./auth/adminauth.php"); if (isset($\_POST['add'])) {

$name=$\_POST["categoryname"];

$image=$\_POST["image"];

$sql="INSERT INTO `category` (categoryname,image) VALUES ('$name','$image')";

$data=mysqli\_query($conn,$sql); if($data){

echo "<script>alert('Inserted successfully')</script>";

}

else{

echo "<script>alert('Insertion unsuccessfully')</script>";

}

}

?>

##### category\_add.php

<?php

include("./auth/adminauth.php");

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial- scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<link rel="stylesheet" href="./css/home.css">

<link rel="stylesheet" href="./css/form.css">

<link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">

<link rel="stylesheet" href="./css/font-awesome- 4.7.0/css/font-awesome.css">

<title>Add lottery</title>

</head>

<body>

<div class="title">

<a href="admin.php"> <span class="heading">

</span></a>

<a href="admin.php"> <span class="heading">Dashboard</span></a>

<a href="logout.php" style="color: white"><span class="fa fa-sign-out fa-2x">Logout</span></a>

</div>

<div class="nav">

<ul>

<li class="dropdown" onclick="toggleDisplay('1')">

<a href="" class="dropbtn">Lottery &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="1">

<a href="add\_lottery.php">Add Lottery</a>

<a href="manage\_lottery.php">Manage Lottery</a>

<a href="addcategory.php">Add category</a>

</div>

</li>

<li class="dropdown" onclick="toggleDisplay('2')">

<a href="#" class="dropbtn">Users &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="2">

<a href="search\_users.php">Search Users</a>

<a href="manage\_users.php">Manage Users</a>

</div>

</li>

<li class="dropdown" onclick="toggleDisplay('3')">

<a href="#" class="dropbtn">Results &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="3">

<a href="add\_results.php">Add Results</a>

<a href="manage\_results.php">Manage Results</a>

</div>

</li>

<li class="dropdown" onclick="toggleDisplay('2')">

<a href="#" class="dropbtn">Orders &nbsp

<span class="fa fa-angle-down"></span>

</a>

<div class="dropdown-content" id="2">

<a href="view\_order.php">View Orders</a>

<a href="view\_sales.php">View sales</a>

</div>

</li>

</ul>

</div>

<div class="main">

<form action="" method="post">

<fieldset>

<legend>Add Lottery</legend>

<input type="text" name="ticketname" placeholder="Lottery Name">

<input type="text" name="ticketdetails" placeholder="Lottery Details">

<input type="text" name="ticketprice" placeholder="Lottery Price">

<input type="date" name="date" placeholder="Date">

<select class="formcontrol" name="categoryname" id="categoryname" placeholder="Category Name" >

<?php

include('init.php');

$sql="select \* from category";

$data=mysqli\_query($conn,$sql); while($row=mysqli\_fetch\_array($data)){

?>

<option style="color black;"

value="<?php echo $row['categoryname'];?>">

<?php echo $row['categoryname']; ?>

</option>

<?php } ?>

<input type="submit" value="Submit" name="add">

</fieldset>

</form>

</div>

<div class="footer">

</div>

</body>

</html>

<?php

52

include('init.php');

if (isset($\_POST['add'])) {

$name=$\_POST["ticketname"];

$details=$\_POST["ticketdetails"];

$price=$\_POST["ticketprice"];

$date=$\_POST["date"];

$category=$\_POST["categoryname"];

$sql = "INSERT INTO ticket(ticketname, ticketdetails, ticketprice, date,categoryname) VALUES ('$name','$details','$price','$date','$category')";

$sqll = "INSERT INTO category(categoryname) VALUES ('$category')";

$result=mysqli\_query($conn,$sql);

$resultt=mysqli\_query($conn,$sqll);

if($result){

echo '<script language="javascript">'; echo 'alert("Successful)';

echo '</script>';

}

}

?>

53

# SYSTEM IMPLEMENTATION

#### Testing

Testing focuses on the logical internals of the software, ensuring that all the statements have been tested on the functional external, that is, conducting tests using various test data to detect errors and ensure that defined input will produce actual results that agreed with required results. It is the major quality control measure used during software development. The software testing methodology is applied in four distinct phases:

* Unit Testing
* Integration Testing
* Validation Testing
* Output Testing

##### Unit Testing

Different modules are tested against the specifications produced during the design for the modules. Unit testing is essentially for verification of the code produced during coding phase. Its main goal is to test the internal logic of the modules, typically done by the programmer of the module. Main focus in this testing is testing the code.

##### Integration Testing

Integration testing is the phase in software testing in which individual software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration plan to those aggregates, and delivers as its output. The purpose of integration testing is to verify functional, performance, and reliability requirements placed on major design items.

##### Output testing

No system could be useful if it does not produce the required output in the specific format. Output testing is performed to ensure the correctness of the

output and its format. The output generated or displayed by the system is tested asking the users about the format required by them.

##### Validation Testing

In software project management, software testing, and software engineering, validation is the process of checking that a software system meets specifications and that it fulfills its intended purpose. The errors which are uncovered during the integration testing are corrected during this phase.

#### System Implementation

The implementation phase of the software development is concerned with translating design specification in to source code. The user tests the developed system and changes are made according to their needs. Our system has been successfully implemented. Before implementation several tests have been conducted to ensure that no errors are encountered during the operation. The implementation phase ends with an evaluation of the system after placing into the operation for a period of time. The implementation stage is a systems project in its own right.

The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from old system to new system. The system can be implemented only after testing is done and is found to be working to specifications.

The implementation stage involves following tasks:

* Careful planning.
* Investigation of system and constraints.
* Design of method to achieve changeover.
* Evaluation of the changeover method.
* Installation of software utilities.
* Training and involvement of user personnel.

## Security

57

Data is the most important element and the main issue related to it is the security of those valuable data. One of the major areas in development process of a system is providing security to all its data in an efficient way. In my work, data it is tightly protected by authentication session password system. Only the administrator can access the entire system. The database server equipped with efficient password security system. So, the entire system is provided with tight security. As the data in our website use a method which encrypts and stores data locally it is less prone to getting hacked and the website it self works offline without any connection with the internet its also less prone to cyber attacks.

58

# CONCLUSION

In conclusion, the Lottery Shop Management System stands as a testament to the seamless fusion of technology and efficiency in the lottery industry. This comprehensive platform has been meticulously crafted to cater to the diverse needs of both administrators and users, offering a user-friendly interface and robust functionalities that redefine the management of lottery-related activities.

For administrators, the system provides unparalleled control, allowing them to effortlessly add and manage lotteries, oversee user accounts, and maintain a transparent record of ticket transactions. The introduction of distinct user roles adds a multi-faceted approach to education, empowering instructors to shape the educational landscape by crafting courses, interacting with students, and contributing to a vibrant knowledge-sharing ecosystem.

Users, on the other hand, are treated to a streamlined and straightforward process, from purchasing lottery tickets to checking results. The system not only facilitates these essential activities but also enhances the user experience through profile customization, ensuring a personalized touch for each participant in the lottery adventure. Even non-registered visitors benefit from the system's accessibility, as they can stay informed about common lottery results directly from the homepage.

The Lottery Shop Management System is more than just a centralized hub for lottery operations; it is a catalyst for positive change in the industry. By promoting transparency, accessibility, and ease of use, it fosters an environment where all stakeholders can engage with confidence. The platform's ability to adapt to the dynamic landscape of lottery management reflects its commitment to staying at the forefront of technological innovation.

As we reflect on the journey of developing this system, it becomes evident that it is not merely a tool but a transformative force in the lottery industry. It is a culmination of dedication, innovation, and a deep understanding of the needs of administrators, users, and even casual visitors. In embracing the Lottery Shop Management System, we embrace a future where lottery operations are not just efficient but also enriching.

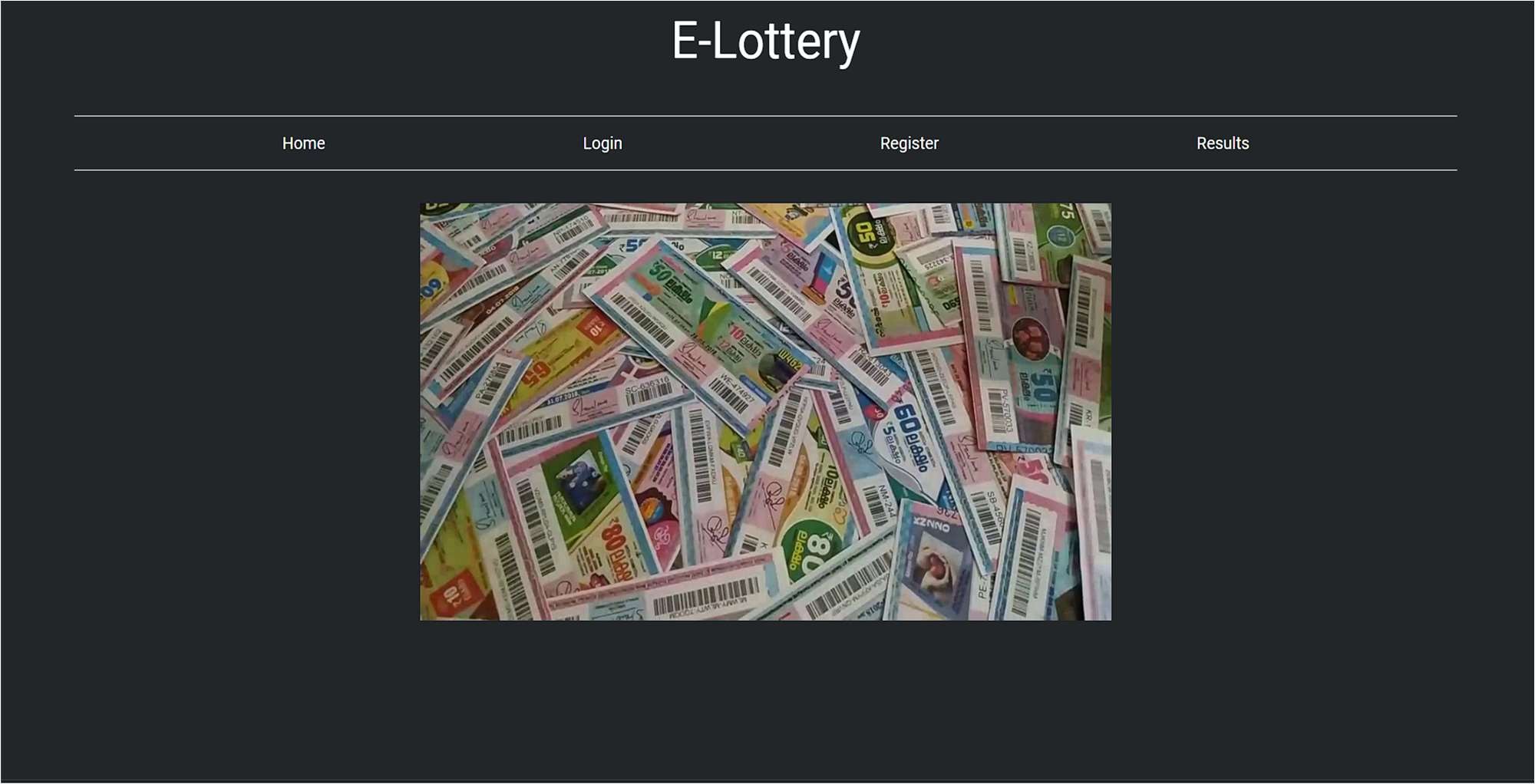
59

60

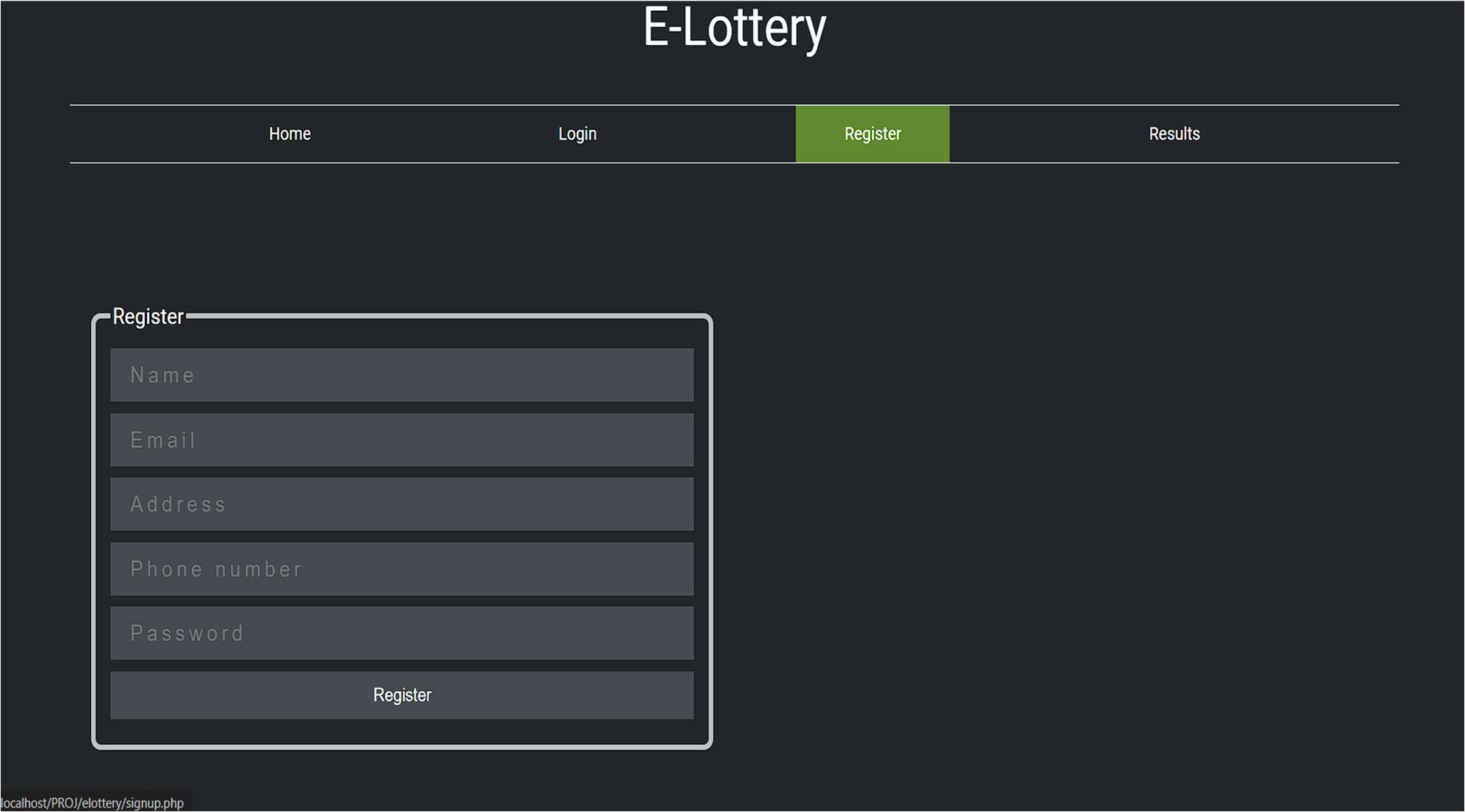
# APPENDIX

#### Sample Input

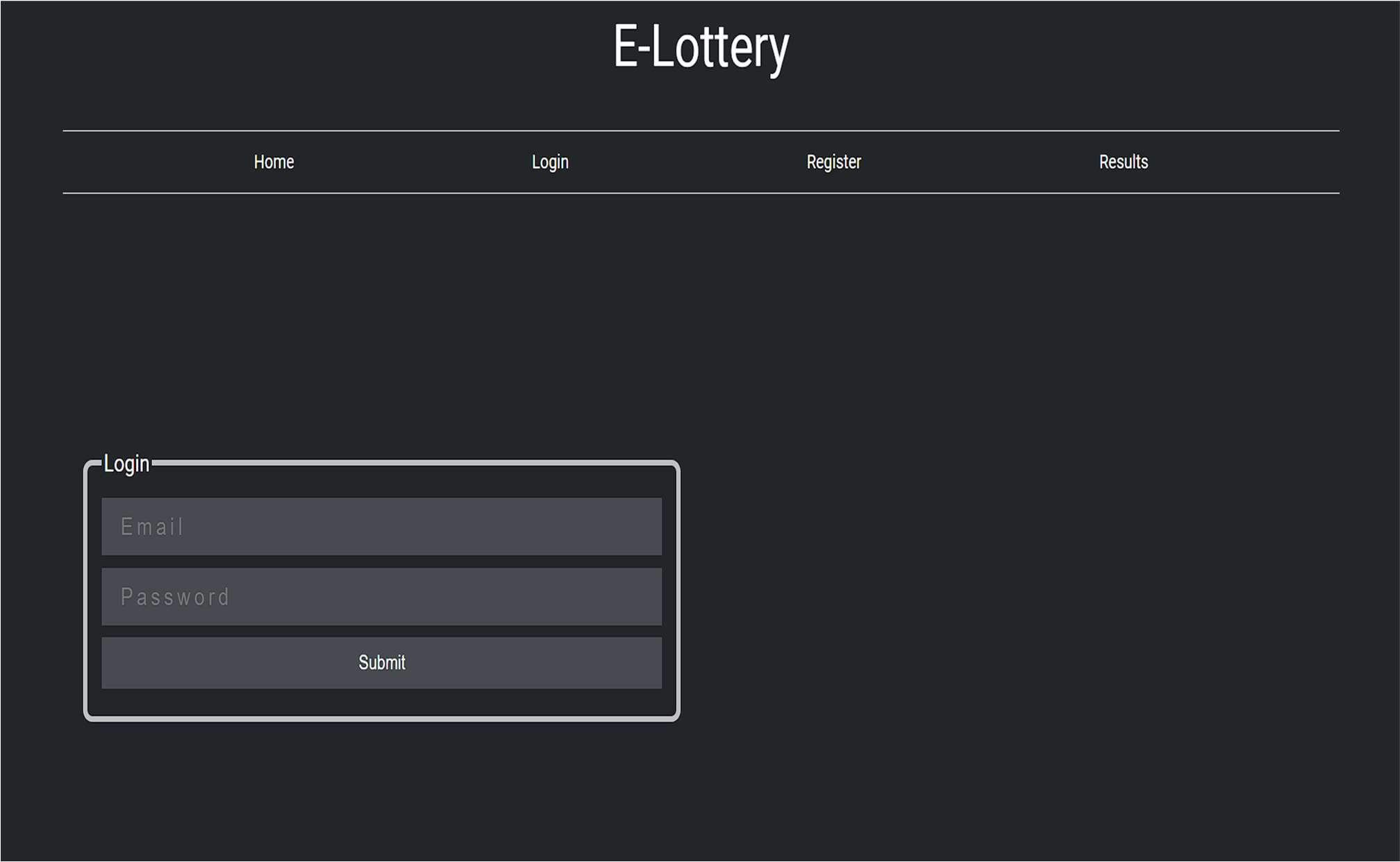
* Home f



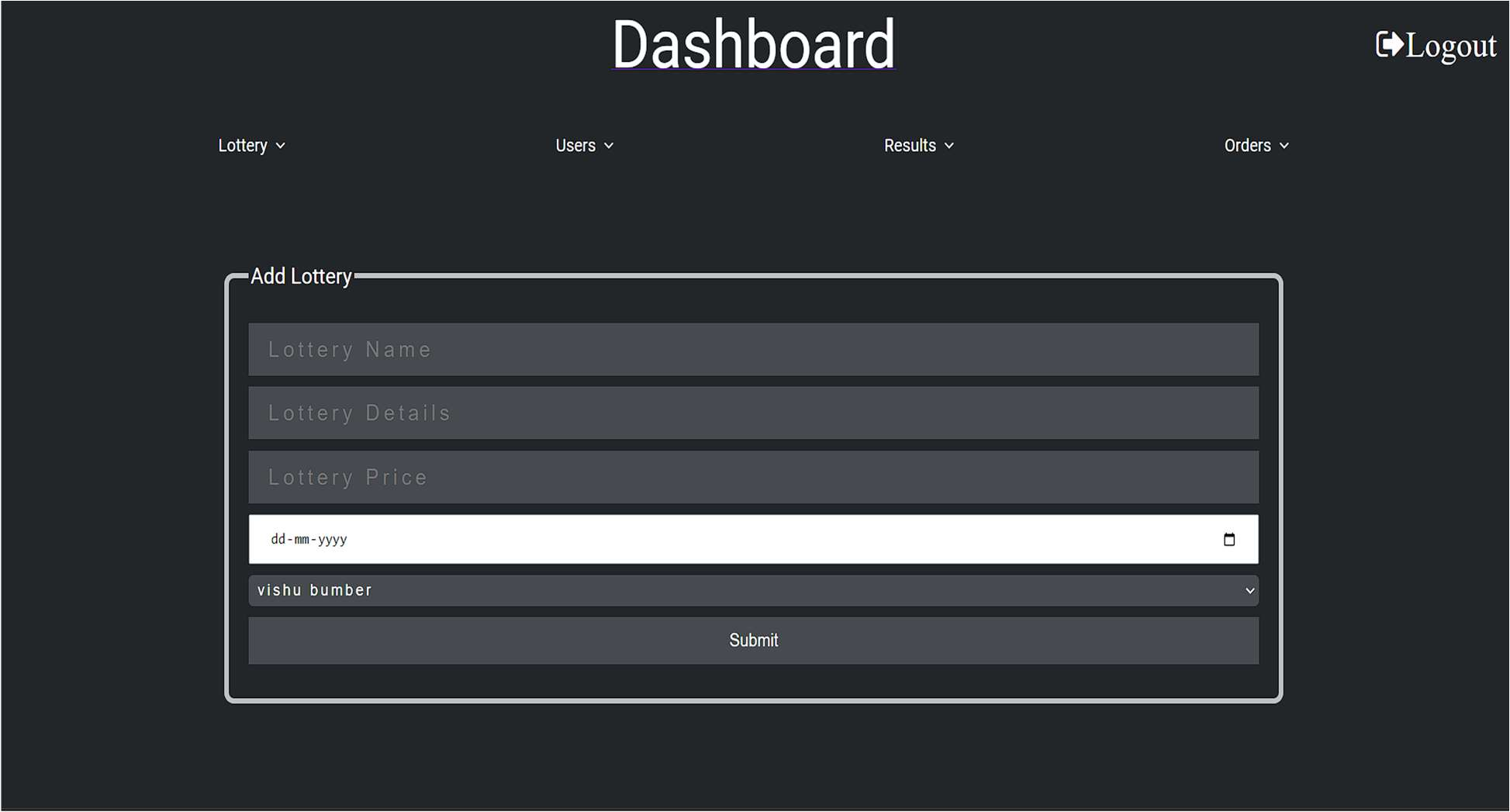
* Registration form



Login

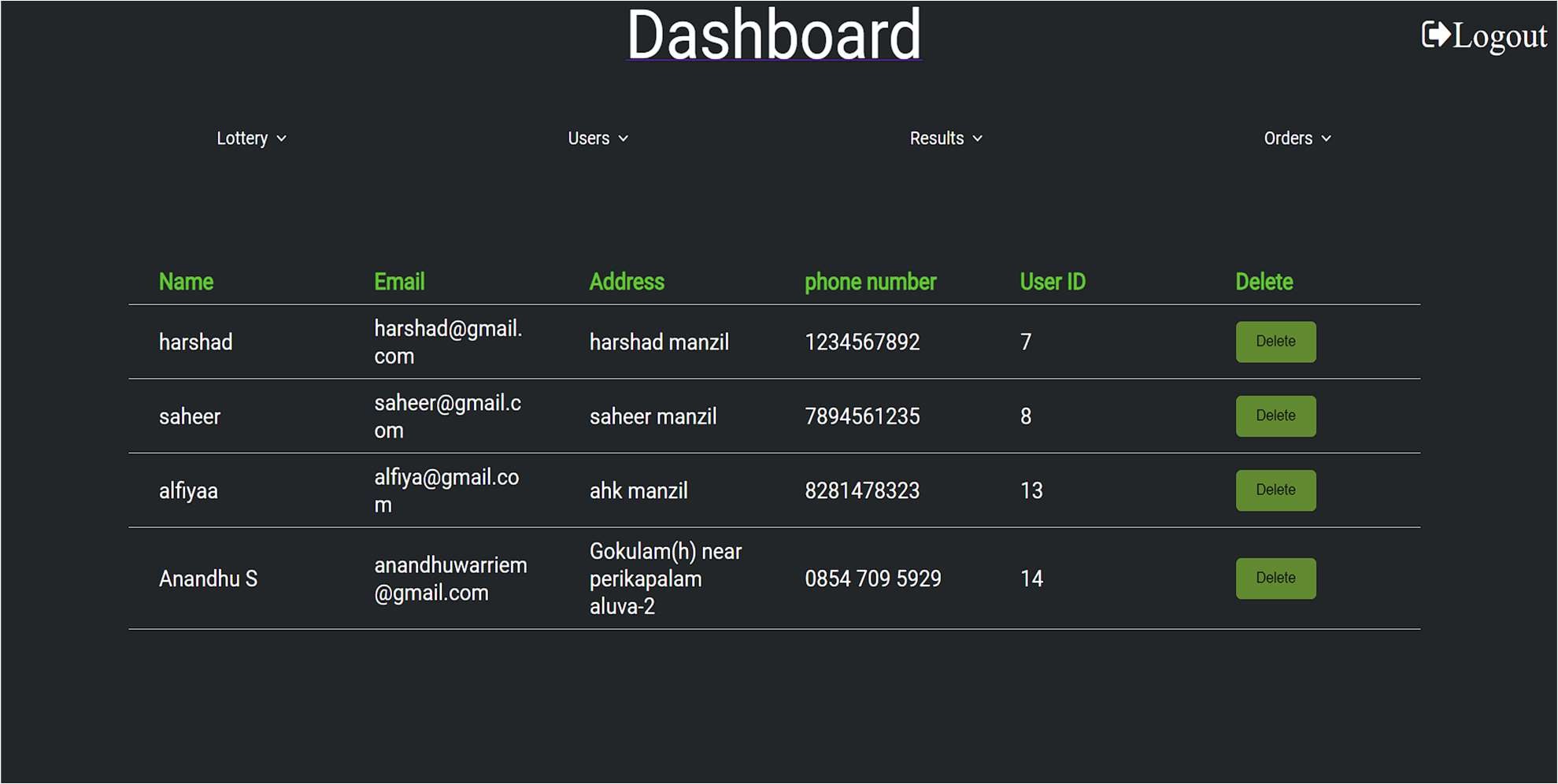


Add lottery form

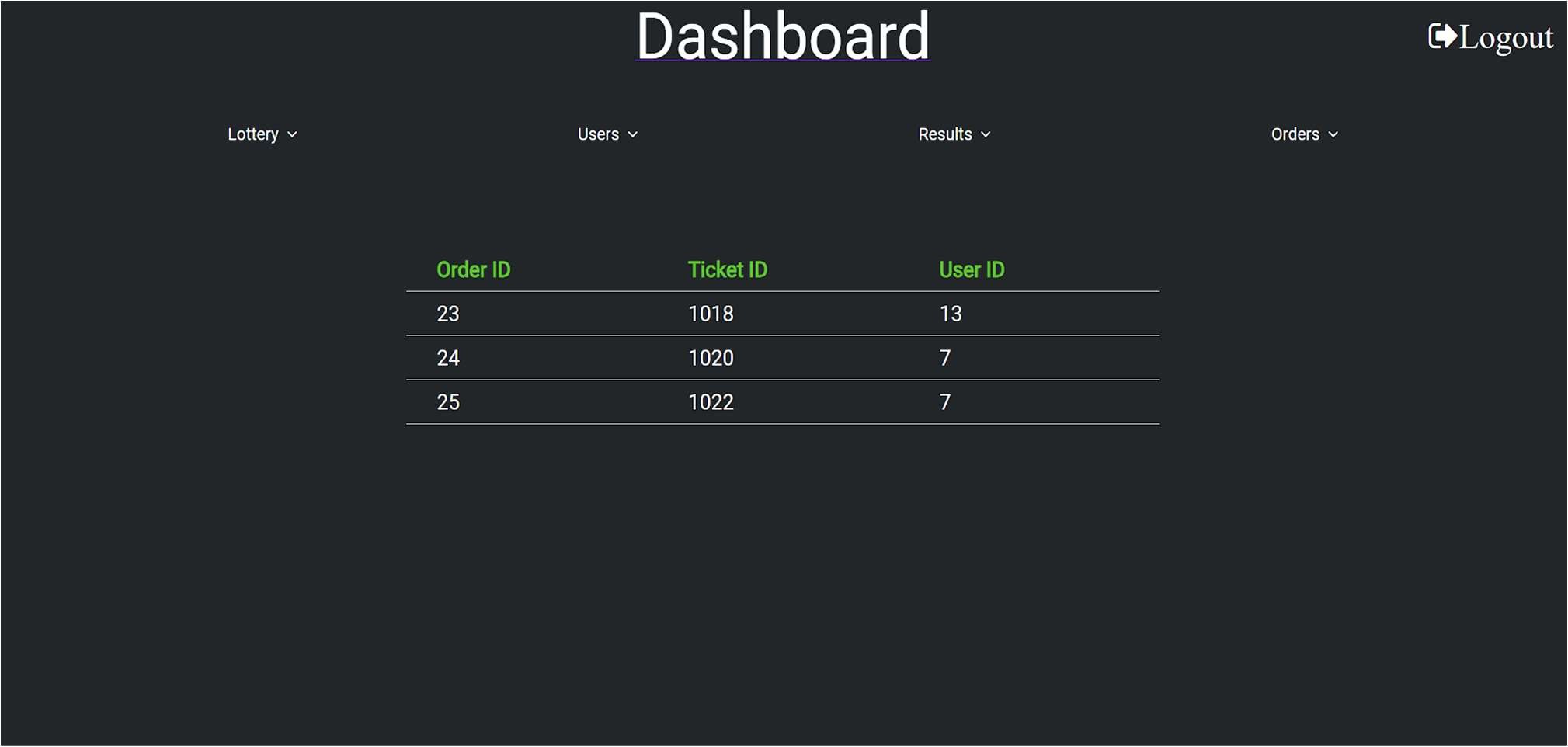


## Sample Output

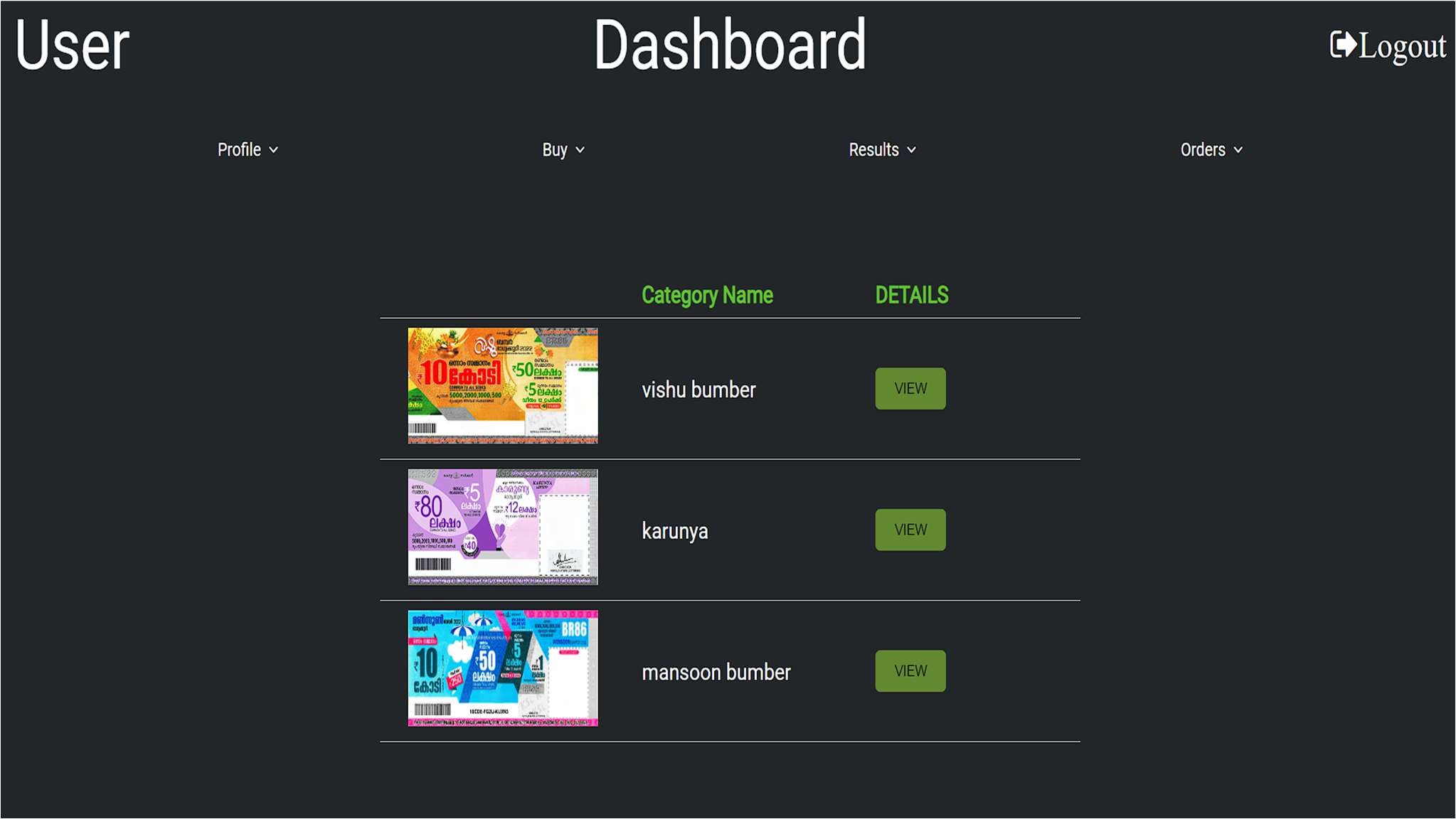
* Registered users



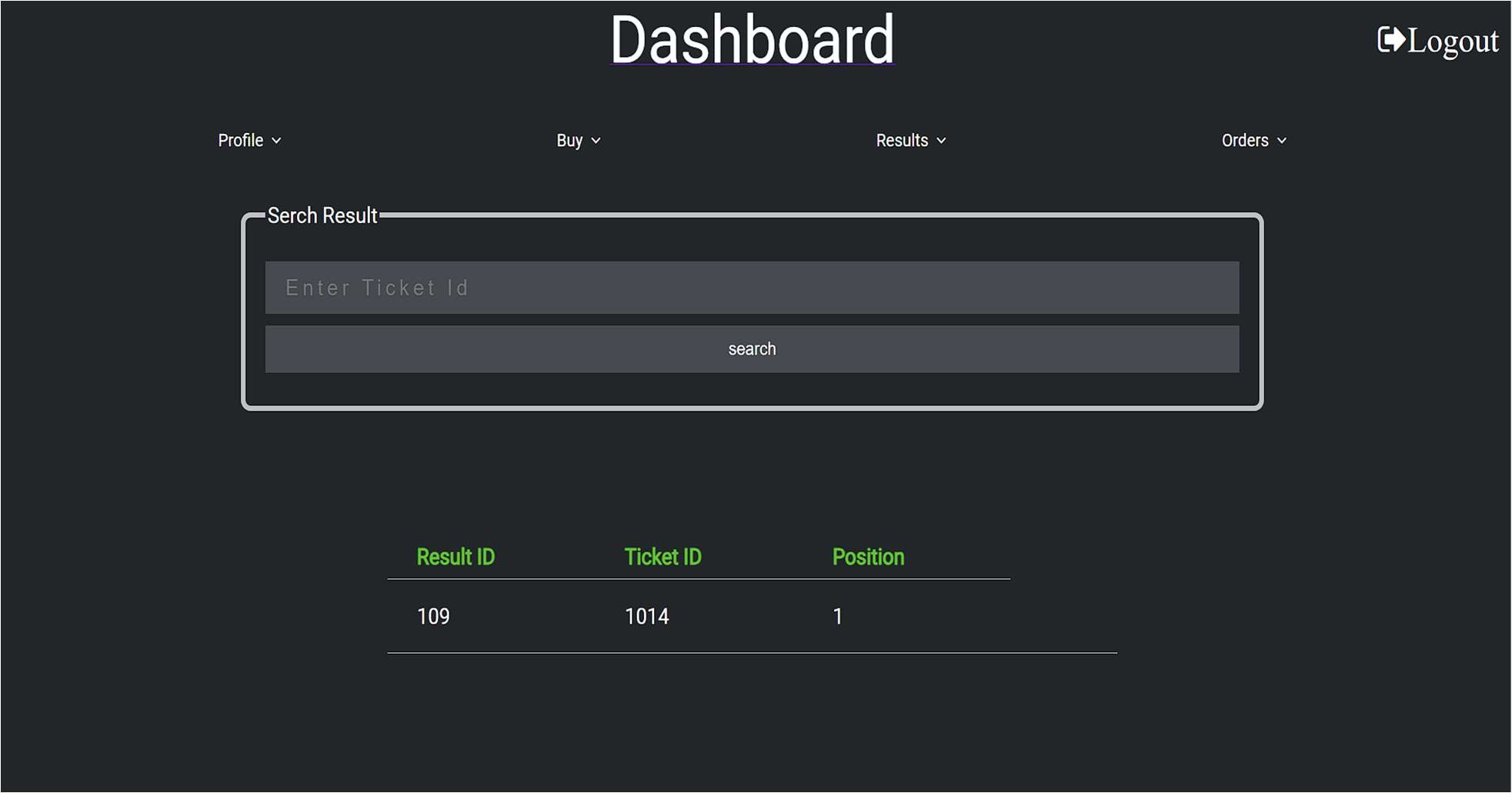
Result view



* Category view



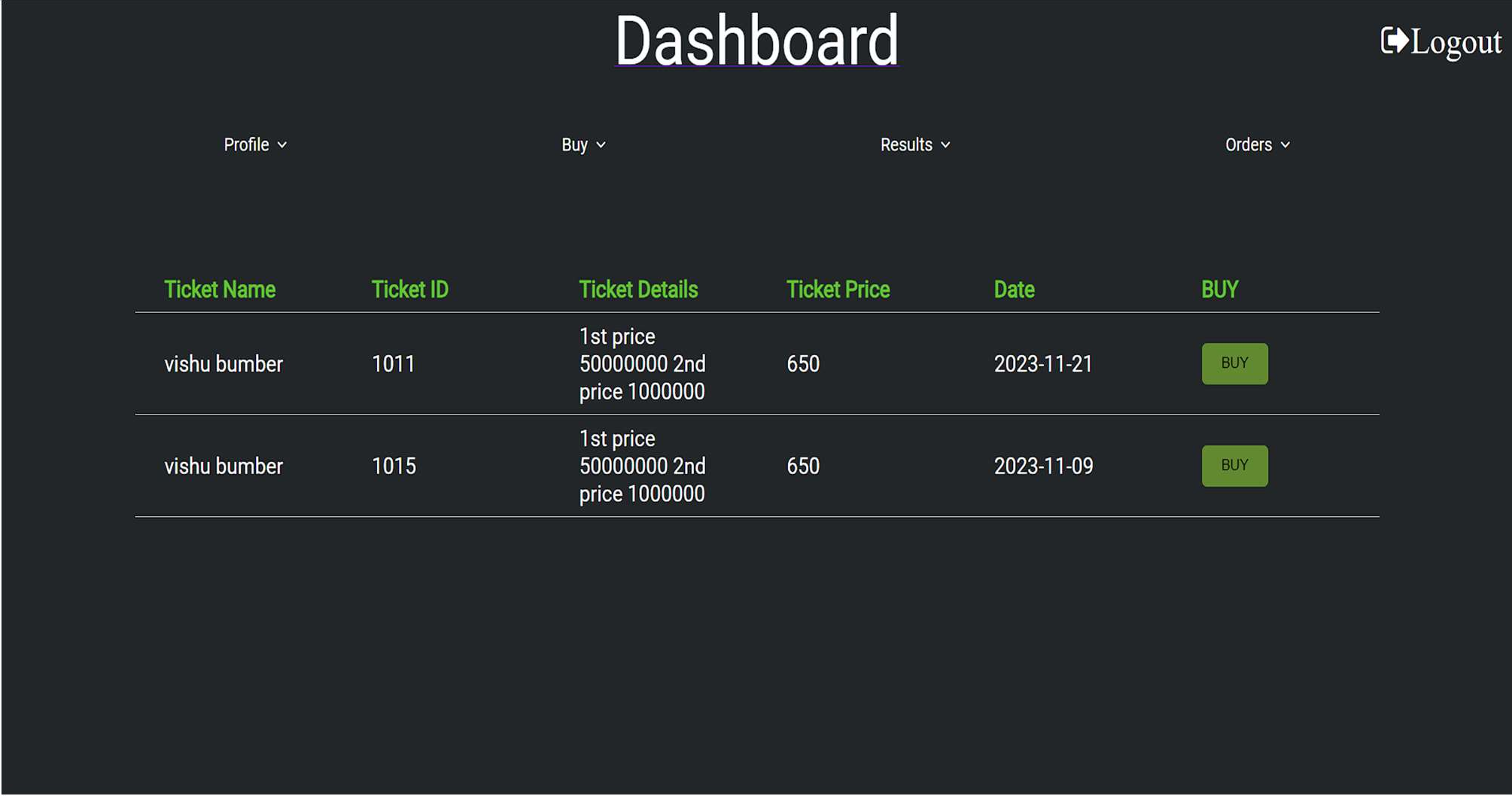
Result view



##### General result viewer



Ticket view



66

# BIBLIOGRAPHY

1. Spurlock, Jake Bootstrap: Responsive Web Development. "O'Reilly Media,

Inc.", 2013

1. Welling,Luke. Laura,Thomson. PHP and MySQL Web Development , Sams Publishing, 2003
2. Nixon, Robin . Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step- by-Step Guide to Creating Dynamic Websites. Shroff Publishers & Distributors, 2009
3. Holdener IlI, Anthony T. Ajax: The Definitive Guide: Iinteractive Applications for the Web. "O'Reilly Media, Inc.", 2008
4. https://[www.w3schools.com/bootstrap4/](http://www.w3schools.com/bootstrap4/)
5. https://[www.w3schools.com/pho](http://www.w3schools.com/pho)
6. https://[www.wBschools.com/sal](http://www.wBschools.com/sal)
7. https://[www.w3schools.com/is](http://www.w3schools.com/is)
8. https://[www.w3schools.com/html](http://www.w3schools.com/html)

67