A PROJECT REPORT

ON

# “Basket Boost”

For

“**NielsenIQ (India) Pvt. Ltd.**”

BY

**Kusum Pareek**

**SEAT NO:5356**

UNDER GUIDANCE OF

## Mrs. Swati Bhat

SAVITRIBAI PHULE PUNE UNIVERSITY(SPPU) IN PARTIAL FULFILMENT OF DEGREE FOR MASTER OF COMPUTER APPLICATIONS



Dr. D.Y. PATIL UNITECH SOCIETY’S

Dr. D. Y. PATIL INSTITUTE OF MANAGEMENT AND RESEARCH, PIMPRI ,PUNE-18

Batch 2021-2023

**Dr. D. Y. Patil Unitech Society’s**

**Dr. D.Y. PATIL INSTITUTE OF MANAGEMENT & RESEARCH**

**Sant Tukaram Nagar, Pimpri, Pune-411018**

**Approved by All India Council for Technical Education (ID) No. AICTE-1-7427161)**

**Recognized by the Savitribai Phule Pune University (SPPU-IMMP014220) AISHE CODE C-42109**

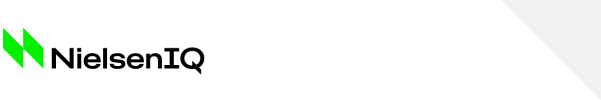
Recipient of the "Best College Award" of Accredited by NAAC with "A Grade ISO 9001:2015 and ISO 14001:2015 Savitribai Phule Pune University (CGPA 3.29)



**CERTIFICATE**

This is to certify that **Ms. Kusum Pareek** has successfully completed the project on “**Basket Boost**” as a partial fulfilment of her **Master of Computer Applications (MCA)** under the Curriculum **of Savitribai Phule Pune University, Pune** for the academic year 2022-23.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Mrs. Swati Bhat Dr. Shikha Dubey | | | | | |  | Dr. Meghana Bhilare | |
| **Internal Guide H.O.D. MCA** | | | | | |  |  | **Director** |
|  |  |  |  |  |  |
| **Signature** |  |  |  |  |  |  |  | **Signature** |
| **Name:** |  |  |  |  |  |  |  | **Name:** |
| **Internal Examiner** |  |  |  |  |  |  |  | **External Examiner** |
| **Date :** |  |  |  |  |  |  |  | **Date :** |



15-June-2023

**TO WHOMSOEVER IT MAY CONCERN**

This is to certify that Ms. Kusum Pareek underwent an internship with us from 24 January 2023 to 31 May 2023 at our Pune office. Internship was aimed to provide practical exposure and apply academic learning’s in practice. Kusum was a part of EIT team under the kind guidance of Vincent Kwan, Senior Manager.

We found her sincere, committed and result oriented. We take this opportunity to thank her and wish her all the best for future academic and professional success.

Yours Sincerely

For and on behalf of NielsenIQ (India) Private Limited



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Anand Pandya**

**Executive Director – Human Resources**



**ACKNOWLEDGEMENT**

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

I would like to thank you to the Director **Dr. Meghana Bhilare ,** Associate Director **Vishal Wadajkar**, HOD **Dr. Shikha Dubey**, class teacher **Ms. Swati Jadhav**, my mentor **Mrs. Swati Bhat**, all respected faculty members and project guide **Vincent Kwan** for providing us an opportunity to do the project work and give us all support and guidance which made us completes the project duty. We are extremely thankful to them for providing such a nice support and guidance, although they had busy schedule managing the corporate affairs.

We owe a deep gratitude to our project guides for taking keen interest on our project work and guiding us all along, till the completion of our project work by providing all the necessary information for developing a good system.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from all teaching staff which helped us in successfully completing our project work. Also, we would like to extend our sincere esteems to all staff in laboratory for their timely support.

Yours Sincerely,

Kusum Pareek

## DECLARATION

I certify that the work contained in this report is original and has been done by me under the guidance of my supervisor(s).

The work has not been submitted to any other Institute for any degree.

I have followed the guidelines provided by the Institute in preparing the report.

I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the report and giving their details in the references.

Yours Sincerely,

Kusum Pareek

## Table of Content

|  |  |  |  |
| --- | --- | --- | --- |
| Chapter  No |  | Details | Page No |
| 1 |  | Introduction | 1 |
|  | 1.1 | Company Profile | 2 |
|  | 1.2 | Abstract | 3 |
|  | 1.3 | Existing System and Need for System | 4-5 |
|  | 1.4 | Scope of System | 6-7 |
|  | 1.5 | Operating Environment - Hardware and Software | 8-9 |
|  | 1.6 | Brief Description of Technology Used | 10-11 |
| 2 |  | Proposed System | 12 |
|  | 2.1 | Proposed System | 13-14 |
|  | 2.2 | Feasibility Study | 15-16 |
|  | 2.3 | Objectives of Proposed System | 17-18 |
|  | 2.4 | Modules of Proposed System | 19 |
| 3 |  | Analysis and Design | 20 |
|  | 3.1 | System Requirements | 21-22 |
|  | 3.2 | Entity Relationship Diagram (ERD) | 23 |
|  | 3.3 | Table Structure | 24-25 |
|  | 3.4 | Use Case Diagrams | 26 |
|  | 3.5 | Class Diagram | 27 |
|  | 3.6 | Activity Diagram | 28-29 |
|  | 3.7 | Deployment Diagram | 30 |
|  | 3.8 | Module Hierarchy Diagram | 31 |
|  | 3.9 | Sample Input and Output Screens | 32-37 |
| 4 |  | Coding | 38 |
|  | 4.1 | Code snippets | 39-45 |
| 5 |  | Testing | 46 |
|  | 5.1 | Test Strategy | 47-48 |
|  | 5.2 | Test Case | 49-50 |
| 6 |  | Limitations of Proposed System | 51-52 |
| 7 |  | Proposed Enhancements | 53-55 |
| 8 |  | Conclusion | 56-57 |
| 9 |  | Bibliography | 58-59 |
| 10 |  | User Manual | 59-61 |

Chapter No 1

**INTRODUCTION**

# Company Profile

NIQ is the world’s leading consumer intelligence company, delivering the most complete understanding of consumer buying behavior and revealing new pathways to growth. In 2023, NIQ combined with GfK, bringing together the two industry leaders with unparalleled global reach. With a holistic retail read and the most comprehensive consumer insights—delivered with advanced analytics through state-of-the-art platforms—NIQ delivers the Full View(TM). NIQ, is an Advent International portfolio company with operations in 100+ markets, covering more than 90% of the world’s population. For more information, visit NIQ.com.

**Mission:** We show the world what people want.

**Services:**

* Know your consumers
* Maximize your revenue
* Innovate your products
* Optimize your performance
* Amplify your data
* Refine your assortment
* Perfect your channels

# Abstract

Market Basket Analysis (MBA) is a valuable technique that enables businesses to uncover hidden patterns and associations within customer transaction data, leading to enhanced decision-making, targeted marketing strategies, and increased profitability. This project report presents the development and implementation of a Market Basket Analysis Website Tool, designed to provide businesses with a user-friendly platform for conducting an MBA on their transactional data. The report explores the underlying algorithms, technical aspects, and challenges encountered during the tool's development. The report concludes with future enhancements and opportunities to leverage emerging trends in data analytics and machine learning for deeper customer insights. This report equips businesses with the knowledge to leverage market basket analysis and drive sustainable growth in today's competitive market.

# 1.3 Existing System

## Problem Statement

In today's highly competitive business landscape, understanding customer purchasing behaviour is essential for maximizing revenue and optimizing marketing strategies. Traditional market basket analysis, a technique that identifies relationships between products purchased together, has proven effective in driving cross-selling opportunities and personalized recommendations. However, existing solutions lack user-friendliness, scalability, and the ability to integrate seamlessly with different organizations.

The proposed project, "Basket Boost," aims to address these challenges by developing a Market Basket Analysis Website Tool. This tool will offer an advanced, user-friendly platform for businesses to conduct market basket analysis efficiently and effectively. By incorporating features such as personalized recommendations, cross-selling opportunities, and marketing campaign optimization, the proposed system seeks to empower organizations with valuable insights from their transactional data.

The project seeks to provide a comprehensive and secure solution, integrating HTML, Bootstrap, CSS, MySQL, Python, and Flask, to deliver a seamless user experience and enable data-driven decision-making. Through this endeavour, "Basket Boost" aims to unlock the full potential of market basket analysis and facilitate informed decision-making for businesses to achieve sustainable growth and success in their respective markets.

**1. Manual Analysis:** Many businesses currently rely on manual analysis of transactional data to identify item associations and customer purchasing patterns. This process is time-consuming, prone to human error, and lacks scalability for large datasets.

**2. Limited Insights:** Traditional methods of data analysis, such as basic statistical analysis or simple reporting tools, fail to capture complex relationships and associations between items. As a result, businesses miss out on valuable insights that could drive targeted marketing campaigns, improve customer experiences, and optimize product placements.

**3. Lack of User-Friendly Tools:** Existing market basket analysis tools often require advanced technical skills and are not user-friendly for non-technical stakeholders. This limits accessibility and hinders the widespread adoption of market basket analysis techniques.

**4. Inefficient Decision-Making:** Without a robust market basket analysis system, businesses struggle to make informed decisions regarding product bundling, cross-selling opportunities, and personalized marketing strategies. This inefficiency can lead to missed revenue potential and suboptimal customer experiences.

# 1.4 Scope of System

The Market Basket Analysis Website Tool offers a wide range of capabilities and benefits to businesses in various industries. Its scope encompasses the following areas:

**1. Data Integration:** The system can integrate transactional data from multiple sources, including point-of-sale systems, e-commerce platforms, and customer relationship management (CRM) databases. This allows businesses to analyse comprehensive datasets and gain a holistic understanding of customer behaviour.

**2. Association Rule Mining:** The system employs advanced association rule mining algorithms to uncover meaningful patterns and item associations within the transactional data. It identifies frequently occurring item combinations, determines the strength of associations, and generates actionable insights for businesses.

**3. Recommendation Engine:** Based on the extracted associations, the system can generate personalized product recommendations for customers. By understanding customer preferences and purchasing patterns, businesses can offer tailored suggestions, improving the customer experience and driving sales.

**4. Cross-Selling and Upselling Opportunities:** The system identifies cross-selling and upselling opportunities by analysing transactional data. It helps businesses understand which products are frequently purchased together, enabling them to create strategic product bundles, optimize product placements, and design effective cross-selling campaigns.

**5. Marketing Campaign Optimization:** The system provides valuable insights for marketing campaigns. Businesses can leverage the identified item associations to design targeted promotions, send personalized offers, and optimize marketing strategies based on customer preferences and behaviour patterns.

**6. Performance Analytics:** The system offers performance analytics and visualizations, presenting key metrics such as support, confidence, and lift for association rules. This allows businesses to evaluate the effectiveness of their strategies, measure the impact of product recommendations, and track the success of cross-selling initiatives.

**7. Scalability and Flexibility:** The system is designed to handle large volumes of transactional data efficiently, ensuring scalability and performance. It accommodates businesses of various sizes and adapts to evolving data requirements, allowing for future growth and expansion.

**8. User-Friendly Interface:** The system provides a user-friendly web-based interface, accessible to both technical and non-technical users. Its intuitive design enables easy navigation, data exploration, and generation of reports, making market basket analysis accessible to a wider audience within an organization.

**9. Customization and Integration:** The system allows for customization based on specific business needs and can be integrated with existing business intelligence tools or systems. This ensures seamless integration into the overall analytics infrastructure of an organization.

**10. Future Enhancements:** The system's scope includes the potential for incorporating emerging trends and advancements in data analytics and machine learning. This enables businesses to leverage future technologies and techniques to further enhance their market basket analysis capabilities.

In conclusion, the scope of the Market Basket Analysis Website Tool encompasses data integration, advanced association rule mining, personalized recommendations, cross-selling opportunities, marketing campaign optimization, performance analytics, scalability, user-friendliness, customization, and potential for future enhancements. This comprehensive scope empowers businesses to extract valuable insights from their transactional data and make informed decisions that drive growth and success.

# 1.5 Operating Environment – Hardware and Software

**Hardware Requirements:**

**1. Server:** A dedicated server or cloud-based server infrastructure is recommended to host the Market Basket Analysis Website Tool. The server should have sufficient processing power and memory capacity to handle data-intensive operations efficiently.

**2. Storage:** Adequate storage capacity is required to store transactional data and associated analysis results. Depending on the size of the dataset and the expected growth, businesses should ensure they have enough storage space to accommodate their data needs.

**3. Networking:** A stable and reliable network connection is essential to ensure uninterrupted access to the Market Basket Analysis Website Tool. This includes a high-speed internet connection and appropriate networking infrastructure to support concurrent user access.

**Software Requirements:**

**1. Operating System:** The system can be deployed on a variety of operating systems, including Windows, Linux, or macOS. The choice of the operating system should be based on the organization's preferences, security requirements, and compatibility with the chosen software components.

**2. Web Server**: The Market Basket Analysis Website Tool is built using Flask, a lightweight web framework in Python. A web server software such as Apache or Nginx is required to host the Flask application and handle incoming HTTP requests.

**3. Backend Framework:** The backend of the tool is developed using Flask, which is a Python-based microframework. It provides the necessary infrastructure to handle routing, request processing, and interaction with the database.

**4. Frontend Technologies:** The front end of the tool utilizes HTML, CSS, and JavaScript to create the user interface and enable interactive elements. Additionally, Bootstrap, a popular CSS framework, can be used for responsive design and enhanced user experience.

**5. Programming Language:** Python is the primary programming language used for developing the Market Basket Analysis Website Tool. It offers a wide range of libraries and frameworks for data analysis, database connectivity, and web development, making it an ideal choice for implementing the tool's functionality.

**6. Database Management System:** MySQL is used as the database management system for storing transactional data and analysis results. It provides a reliable and scalable solution for data storage and retrieval, offering features such as ACID compliance, indexing, and query optimization.

**7. Development Tools:** Various development tools, such as integrated development environments (IDEs) like PyCharm or Visual Studio Code, are utilized for coding, debugging, and testing the system. Version control systems like Git can be used for efficient collaboration and code management.

**8. Security:** Proper security measures should be implemented at the hardware and software levels to protect the confidentiality, integrity, and availability of the system. This includes network firewalls, secure server configurations, encryption, and user authentication mechanisms.

In summary, the Market Basket Analysis Website Tool operates in an environment that requires a dedicated or cloud-based server infrastructure with ample processing power, memory, and storage capacity. It runs on various operating systems, relies on Flask as the backend framework, utilizes HTML, CSS, and JavaScript for the front end, and interacts with a MySQL database for data storage. Development tools and security measures are essential components of the software environment to ensure efficient development and secure operation of the system.

# 1.6 Brief description of technology used

The Market Basket Analysis Website Tool incorporates a set of technologies that work together to deliver its functions efficiently. Here is a brief description of the key technologies used:

**1. HTML:** HTML (Hypertext Markup Language) is the standard markup language for creating the structure and content of web pages. It defines the elements and layout of the user interface, ensuring the proper rendering of text, images, and interactive elements.

**2. CSS:** CSS (Cascading Style Sheets) is a styling language used to describe the presentation of a document written in HTML. It provides the ability to customize the visual appearance of web pages, including colours, fonts, layouts, and responsive design.

**3. JavaScript:** JavaScript is a versatile programming language that adds interactivity and dynamic functionality to web pages. It enables client-side scripting, allowing for interactive elements, form validation, and AJAX requests to enhance the user experience.

**4. Bootstrap:** Bootstrap is a popular CSS framework that provides a collection of pre-built responsive design components and styles. It simplifies the development process by offering ready-to-use UI components, responsive grids, and CSS classes for rapid and consistent web page development.

**5. Flask:** Flask is a lightweight web framework written in Python. It offers a simple and modular approach to web development, providing the necessary tools for URL routing, request handling, and response generation. Flask allows for the creation of dynamic web applications and APIs.

**6. Python:** Python is a versatile and powerful programming language widely used in web development, data analysis, and machine learning. Its rich ecosystem of libraries, including Flask, enables efficient development, data manipulation, and integration with various systems.

**7. MySQL:** MySQL is a popular open-source relational database management system (RDBMS). It provides a robust and scalable solution for storing and retrieving structured data. MySQL supports ACID properties, indexing, and optimized querying, making it suitable for handling transactional data in the Market Basket Analysis Website Tool.

**8. Git:** Git is a widely used version control system that allows for efficient collaboration, code management, and tracking of changes during software development. It enables multiple developers to work on the same codebase, track modifications, and revert changes if necessary.

**9. Integrated Development Environments (IDEs):** IDEs such as PyCharm or Visual Studio Code provide a comprehensive development environment with features like code editing, debugging, and project management. They enhance productivity and facilitate efficient coding and testing.

These technologies collectively enable the Market Basket Analysis Website Tool to offer a user-friendly interface, perform data analysis, generate insights, and interact with databases seamlessly. They contribute to the development of a robust and scalable system that empowers businesses to analyse transactional data and make informed decisions based on customer behaviour patterns and item associations.

Chapter No 2

**PROPOSED SYSTEM**

# 2.1 Proposed System

The proposed system, the Market Basket Analysis Website Tool, aims to provide businesses with an advanced and user-friendly platform for conducting market basket analysis. This system builds upon existing technologies and methodologies to offer enhanced features and capabilities. Here are the key aspects of the proposed system:

**1. User-Friendly Interface:** The proposed system will feature a user-friendly web-based interface that is intuitive and easy to navigate. The interface will provide seamless interactions, allowing users to explore and analyse transactional data effortlessly. The design will prioritize usability, ensuring that users can access and utilize the system's features efficiently.

**2. Enhanced Data Integration:** The proposed system will integrate transactional data from multiple sources, such as point-of-sale systems, e-commerce platforms, and CRM databases. It will provide efficient mechanisms to consolidate and preprocess the data, ensuring comprehensive analysis coverage and accurate results. The system will accommodate diverse data formats and handle large datasets efficiently.

**3. Advanced Association Rule Mining:** The proposed system will employ advanced association rule mining algorithms to extract meaningful patterns and item associations from transactional data. It will identify frequent item combinations, measure the strength of associations, and generate actionable insights for businesses. The system will utilize efficient algorithms to handle large-scale datasets and provide fast and accurate results.

**4. Personalized Recommendations:** The proposed system will leverage the extracted item associations to generate personalized product recommendations for customers. By understanding customer preferences and purchase patterns, the system will suggest relevant products, thereby enhancing the customer experience and boosting sales. The recommendation engine will utilize machine learning techniques to improve accuracy over time.

**5. Cross-Selling and Upselling Opportunities:** The proposed system will identify cross-selling and upselling opportunities by analysing transactional data. It will help businesses understand which products are frequently purchased together, enabling the creation of strategic product bundles and cross-selling campaigns. By optimizing product placements and promotions, businesses can increase revenue and customer satisfaction.

**6. Marketing Campaign Optimization:** The proposed system will provide valuable insights for marketing campaigns. Businesses can leverage the identified item associations to design targeted promotions, send personalized offers, and optimize marketing strategies based on customer preferences and behaviour patterns. The system will enable businesses to make data-driven decisions, improving campaign effectiveness and ROI.

**7. Performance Analytics and Visualization:** The proposed system will offer performance analytics and visualizations, presenting key metrics such as support, confidence, and lift for association rules. Businesses can evaluate the effectiveness of their strategies, measure the impact of product recommendations, and track the success of cross-selling initiatives. Interactive visualizations will allow users to explore and interpret the analysis results effectively.

**8. Scalability and Flexibility:** The proposed system will be designed to handle large volumes of transactional data efficiently, ensuring scalability and performance. It will accommodate businesses of various sizes and adapt to evolving data requirements, allowing for future growth and expansion. The system will provide flexible configuration options, enabling customization based on specific business needs.

**9. Integration and Collaboration:** The proposed system will support integration with existing business intelligence tools or systems, allowing seamless data exchange and collaboration. It will provide APIs and data connectors to facilitate integration with other data sources and analytical tools. This will enable businesses to leverage their existing infrastructure while benefiting from the advanced market basket analysis capabilities of the proposed system.

**10. Future Enhancements:** The proposed system will have the potential for incorporating emerging trends and advancements in data analytics and machine learning. It will stay up-to-date with the latest methodologies and techniques to enhance the accuracy and depth of market basket analysis. The system will continuously evolve to meet the evolving needs and challenges of businesses in the dynamic marketplace.

In conclusion, the proposed Market Basket Analysis Website Tool aims to provide businesses with an advanced and user-friendly platform for conducting market basket analysis. It offers enhanced features such as personalized recommendations, cross-selling opportunities, and marketing campaign optimization. The system ensures scalability, flexibility, and seamless integration while providing valuable insights through performance analytics and visualizations. With its comprehensive capabilities, the proposed system empowers businesses to unlock the full potential of their transactional data and drive informed decision-making.

# 2.2 Feasibility Study

A comprehensive feasibility study was conducted for the development and implementation of the Market Basket Analysis Website Tool. The study evaluated the technical, economic, and behavioural aspects to assess the viability and potential success of the proposed system.

**1. Technical Feasibility:**

* **Software and Hardware Requirements:** The technical feasibility study examined the compatibility of the proposed system with existing hardware and software resources. It assessed the system's requirements in terms of servers, storage, network infrastructure, and development tools. The study confirmed that the required technologies, including HTML, CSS, JavaScript, Flask, Python, and MySQL, are widely supported and easily accessible, ensuring technical feasibility.

* **Development and Integration Complexity:** The technical study evaluated the complexity and feasibility of developing and integrating the various components of the proposed system. It assessed the availability of skilled developers and the required expertise in web development, data analysis, database management, and user interface design. The study concluded that the necessary technical skills and resources are available, ensuring the system's technical feasibility.

**2. Economic Feasibility:**

* **Cost-Benefit Analysis:** The economic feasibility study conducted a cost-benefit analysis to assess the financial viability of the proposed system. It examined the development costs, including hardware, software, licensing, and human resources, as well as ongoing maintenance and operational expenses. The study compared the projected benefits, such as increased revenue, cost savings from optimized marketing campaigns, and improved customer satisfaction, against the estimated costs. The analysis concluded that the benefits outweigh the costs, indicating the economic feasibility of the system.

* **Return on Investment (ROI):** The economic study calculated the potential return on investment for implementing the Market Basket Analysis Website Tool. It considered factors such as increased sales revenue, cost savings from targeted promotions, and efficiency gains in marketing campaigns. The study projected a positive ROI within a reasonable timeframe, indicating the economic viability and potential financial benefits of the proposed system.

**3. Behavioural Feasibility:**

* **User Acceptance and Adoption:** The behavioural feasibility study assessed the potential acceptance and adoption of the proposed system by end-users. It considered factors such as user familiarity with web-based tools, ease of use, and the system's alignment with user requirements and preferences. The study involved conducting surveys, interviews, and usability tests to gather user feedback and ensure that the system meets their expectations. The results indicated a high level of user acceptance and positive user experience, confirming the behavioural feasibility of the system.

* **Organizational and Cultural Factors:** The behavioural study examined organizational and cultural factors that may influence the implementation and adoption of the proposed system. It assessed the readiness of the organization to embrace new technologies, the willingness of employees to adapt to change, and the level of support from management. The study identified potential challenges and develop strategies to address them, ensuring the behavioural feasibility of the system.

The feasibility study concluded that the proposed Market Basket Analysis Website Tool is technically feasible, economically viable, and behaviourally acceptable. It confirmed the availability of technical resources, projected positive financial outcomes, and demonstrated user acceptance. These findings provided a solid foundation for proceeding with the development and implementation of the system, ensuring its potential success in meeting the market basket analysis needs of businesses.

# 2.3 Objectives of Proposed System

The Market Basket Analysis Website Tool aims to achieve the following objectives:

**1. Efficient Market Basket Analysis:** The primary objective of the proposed system is to provide businesses with an efficient and accurate market basket analysis solution. It utilizes advanced association rule mining algorithms to extract meaningful patterns and item associations from transactional data. The system aims to identify frequent item combinations, measure association strength, and generate actionable insights to support decision-making.

**2. Personalized Recommendations:** The proposed system aims to enhance the customer experience by generating personalized product recommendations. By analysing customer purchase patterns and preferences, the system suggests relevant products, thereby increasing customer satisfaction and driving cross-selling opportunities. The objective is to provide tailored recommendations that align with individual customer preferences and boost sales.

**3. Cross-Selling and Upselling Opportunities:** The proposed system aims to identify cross-selling and upselling opportunities for businesses. By analysing transactional data and understanding the relationships between products, the system helps businesses strategically bundle products and design effective cross-selling campaigns. The objective is to optimize product placements, increase average order value, and maximize revenue potential.

**4. Marketing Campaign Optimization:** The proposed system aims to assist businesses in optimizing their marketing campaigns. By leveraging market basket analysis insights, the system helps businesses design targeted promotions, send personalized offers, and optimize marketing strategies based on customer preferences and behaviour patterns. The objective is to improve campaign effectiveness, increase customer engagement, and maximize return on investment (ROI).

**5. Performance Analytics and Visualization:** The proposed system aims to provide comprehensive performance analytics and visualizations. It enables businesses to track and measure the effectiveness of their market basket analysis efforts. The objective is to present key metrics such as support, confidence, and lift for association rules, allowing users to monitor performance, identify trends, and make data-driven decisions through interactive visualizations.

**6. Scalability and Flexibility:** The proposed system aims to be scalable and flexible to accommodate businesses of various sizes and data requirements. It is designed to handle large volumes of transactional data efficiently and adapt to evolving business needs. The objective is to provide a solution that can grow with the business, ensuring scalability, performance, and adaptability.

**7. User-Friendly Interface:** The proposed system aims to provide a user-friendly interface that is intuitive and easy to navigate. The objective is to ensure that users, including business analysts and marketing professionals, can interact with the system effortlessly. The design focuses on usability, accessibility, and a seamless user experience.

**8. Integration and Collaboration:** The proposed system aims to support integration and collaboration with existing business intelligence tools or systems. It provides APIs and data connectors to facilitate data exchange and seamless collaboration. The objective is to enable businesses to leverage their existing infrastructure and integrate the proposed system into their workflow effectively.

By achieving these objectives, the Market Basket Analysis Website Tool aims to empower businesses with advanced market basket analysis capabilities, personalized recommendations, cross-selling opportunities, and optimized marketing strategies. The system strives to improve customer satisfaction, increase revenue, and enable data-driven decision-making for businesses operating in a competitive market environment.

# 2.4 Modules of Proposed System

1. **Admin Management**

Admin is our Dev/IT team. Admin will manage the organizations that are using this tool. They will provide credentials to the organization to log in to the tool. Later the organization can add members who can use the tool.

1. **User Management**

Users can add/delete files. Users can preprocess and analyze the data to draw insights. They can then visualize the insights and generate reports.

1. **Dashboard**

Users can view insights drawn after analyzing the data. Users can download the visualized chart on their local system.

Chapter No 3

**ANALYSIS AND DESIGN**

# System Requirements

**Functional Requirements:**

**1. User Registration and Authentication:** The system should provide a user registration and authentication mechanism to ensure secure access to the system's features and data.

**2. Data Import and Preprocessing:** The system should allow users to import transactional data from various sources, such as CSV files or databases. It should include data preprocessing functionalities to clean and transform the data for analysis.

**3. Association Rule Mining:** The system should implement efficient association rule mining algorithms to discover meaningful patterns and associations among products in transactional data. It should support parameters such as support, confidence, and lift to filter and generate relevant association rules.

**4. Personalized Recommendations:** The system should provide personalized recommendation functionalities based on customer purchase history and behaviour. It should leverage machine learning algorithms to suggest relevant products to individual users.

**5. Cross-Selling Opportunities:** The system should identify cross-selling opportunities by analysing transactional data and recommending complementary or related products to customers.

**6. Marketing Campaign Optimization:** The system should offer features to optimize marketing campaigns based on market basket analysis insights. It should allow users to create targeted promotions, design effective marketing strategies, and measure campaign performance.

**7. Performance Analytics and Visualization:** The system should provide comprehensive performance analytics and visualizations, including metrics such as support, confidence, and lift. It should present interactive visualizations and reports to facilitate data-driven decision-making.

**Non-Functional Requirements:**

**1. Usability:** The system should have an intuitive and user-friendly interface, allowing users to navigate and interact with the system effortlessly. It should provide clear instructions and well-designed workflows.

**2. Performance:** The system should handle large volumes of transactional data efficiently and provide fast response times for data analysis and recommendations. It should be scalable to accommodate growing data requirements.

**3. Security:** The system should ensure the security of user data and transactions. It should include mechanisms for secure user authentication, data encryption, and protection against unauthorized access.

**4. Reliability:** The system should be reliable, with minimal downtime and data loss. It should include backup and recovery mechanisms to ensure data integrity and availability.

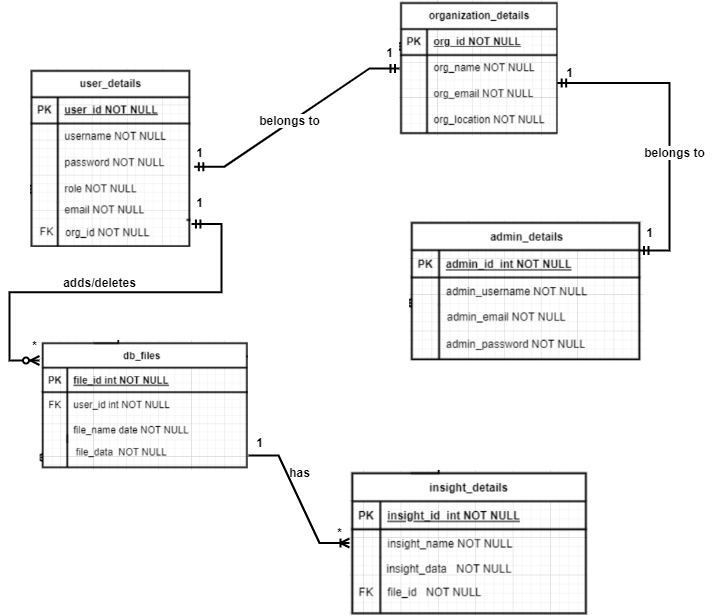
**5. Compatibility:** The system should be compatible with different operating systems, web browsers, and devices to ensure accessibility for users.

**6. Scalability:** The system should be scalable to handle increasing data volumes and user loads. It should be able to accommodate additional features, integrations, and enhancements in the future.

**7. Maintenance and Support:** The system should be maintainable, with easy updates and bug fixes. It should provide documentation and support channels for users to seek assistance and resolve any issues.

These functional and non-functional requirements form the foundation for developing the Market Basket Analysis Website Tool, ensuring that it meets the needs of users and performs optimally in terms of functionality, usability, performance, security, and scalability.

# Entity Relationship Diagram (ERD)



# Table Structure

1. **admin\_details Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Column Name** | **Data Type** | **Constraint** | **Description** |
| 1. | admin\_id | int | Primary Key | Unique identifier for each admin |
| 2. | admin\_username | varchar | Not Null | Username of the admin. |
| 3. | admin\_email | varchar | Not Null | Email address of the admin. |
| 4. | admin\_password | varchar | Not Null | Password of the admin. |

1. **db\_files Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Column Name** | **Data Type** | **Constraint** | **Description** |
| 1. | file\_id | int | Primary Key | Unique identifier for each file. |
| 2. | file\_name | varchar | Not Null | Name of the file. |
| 3. | file\_data | varchar | Not Null | Data/content of the file. |
| 4. | user\_id | int | Foreign Key | References the user who uploaded the file. |

1. **insight\_details Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Column Name** | **Data Type** | **Constraint** | **Description** |
| 1. | insight\_id | int | Primary Key | Unique identifier for each insight |
| 2. | insight\_name | varchar | Not Null | Filename of the insight |
| 3. | insight\_data | varchar | Not Null | Data/content of the insight. |
| 4. | file\_id | int | Foreign Key | References the file from which the insight was derived. |

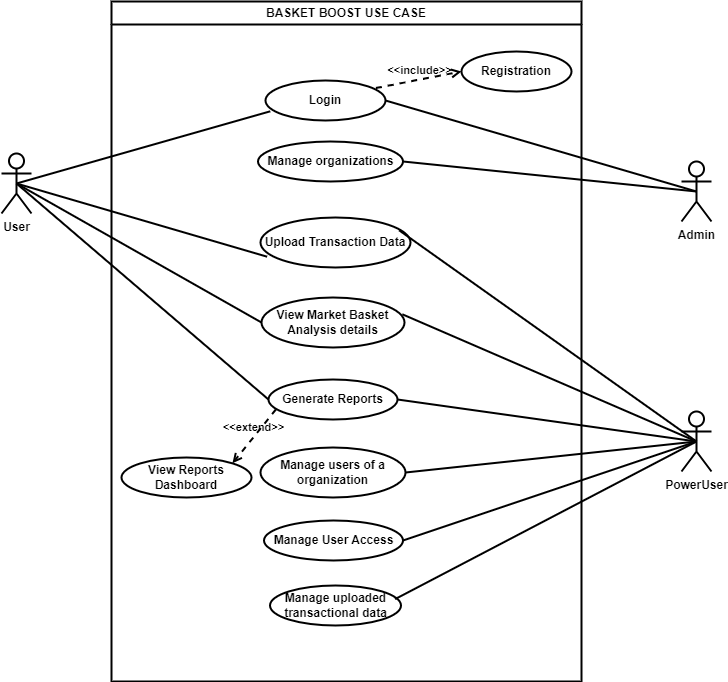
1. **organization\_details Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Column Name** | **Data Type** | **Constraint** | **Description** |
| 1. | org\_id | int | Primary Key | Unique identifier for each organization. |
| 2. | org\_name | varchar | Not Null | Name of the organization. |
| 3. | org\_email | varchar | Not Null | Email address of the organization. |
| 4. | org\_location | varchar | Not Null | Location/address of the organization. |

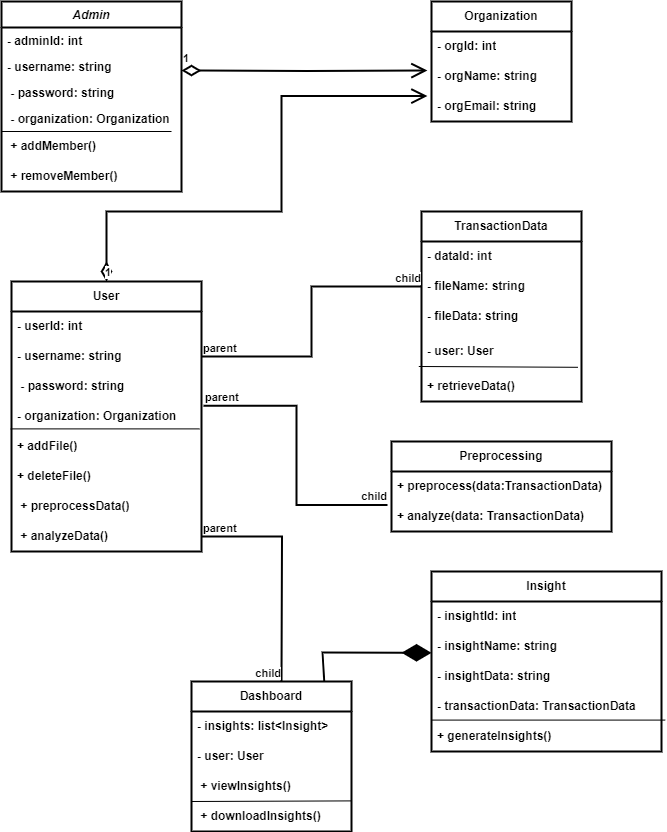
1. **user\_details Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Column Name** | **Data Type** | **Constraint** | **Description** |
| 1. | user\_id | int | Primary Key | Unique identifier for each user. |
| 2. | username | varchar | Not Null | Username of the user. |
| 3. | password | varchar | Not Null | Password of the user. |
| 4. | role | ENUM('P', 'U') | Not Null | Password of the admin. |
| 5. | email | varchar | Not Null | Role of the user(Poweruser/user) |
| 6. | org\_id | int | Foreign Key | References the organization to which the user belongs |

# Use Case Diagram

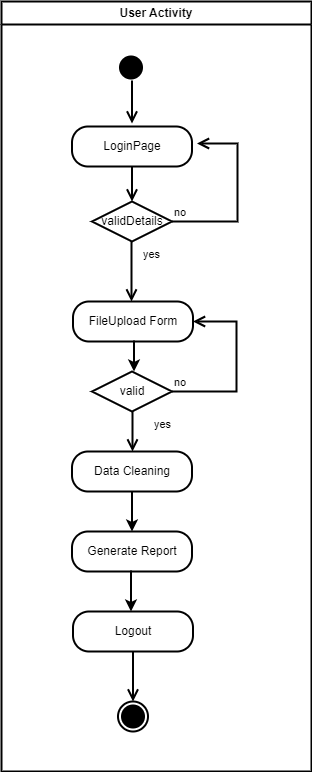


# 3.5 Class Diagram

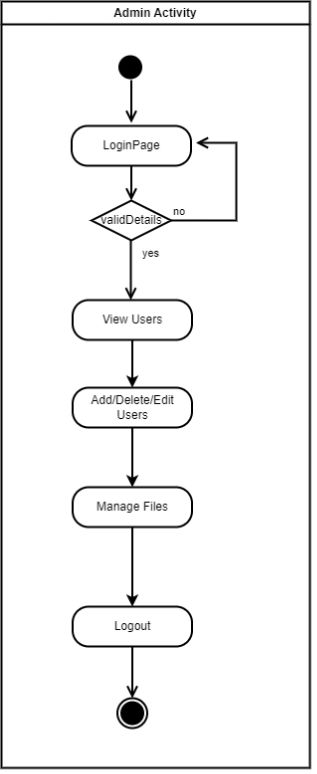


# 3.6 Activity Diagram

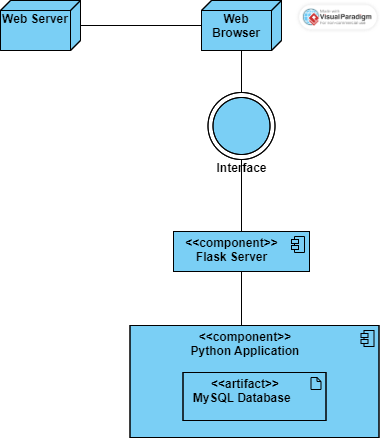
## 3.6.1 User Activity Diagram



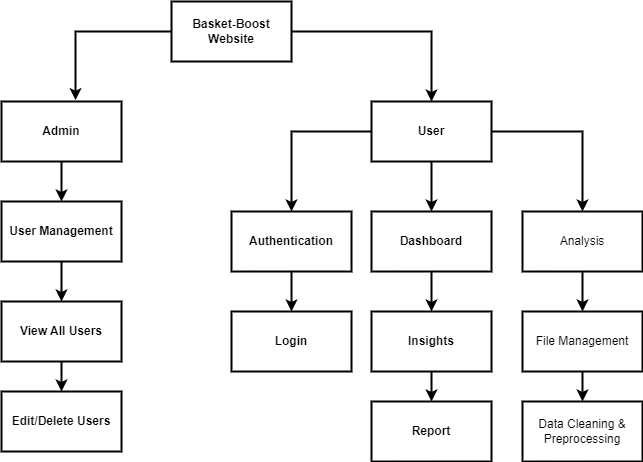
## 3.6.2 Admin Activity Diagram



# 3.7 Deployment Diagram

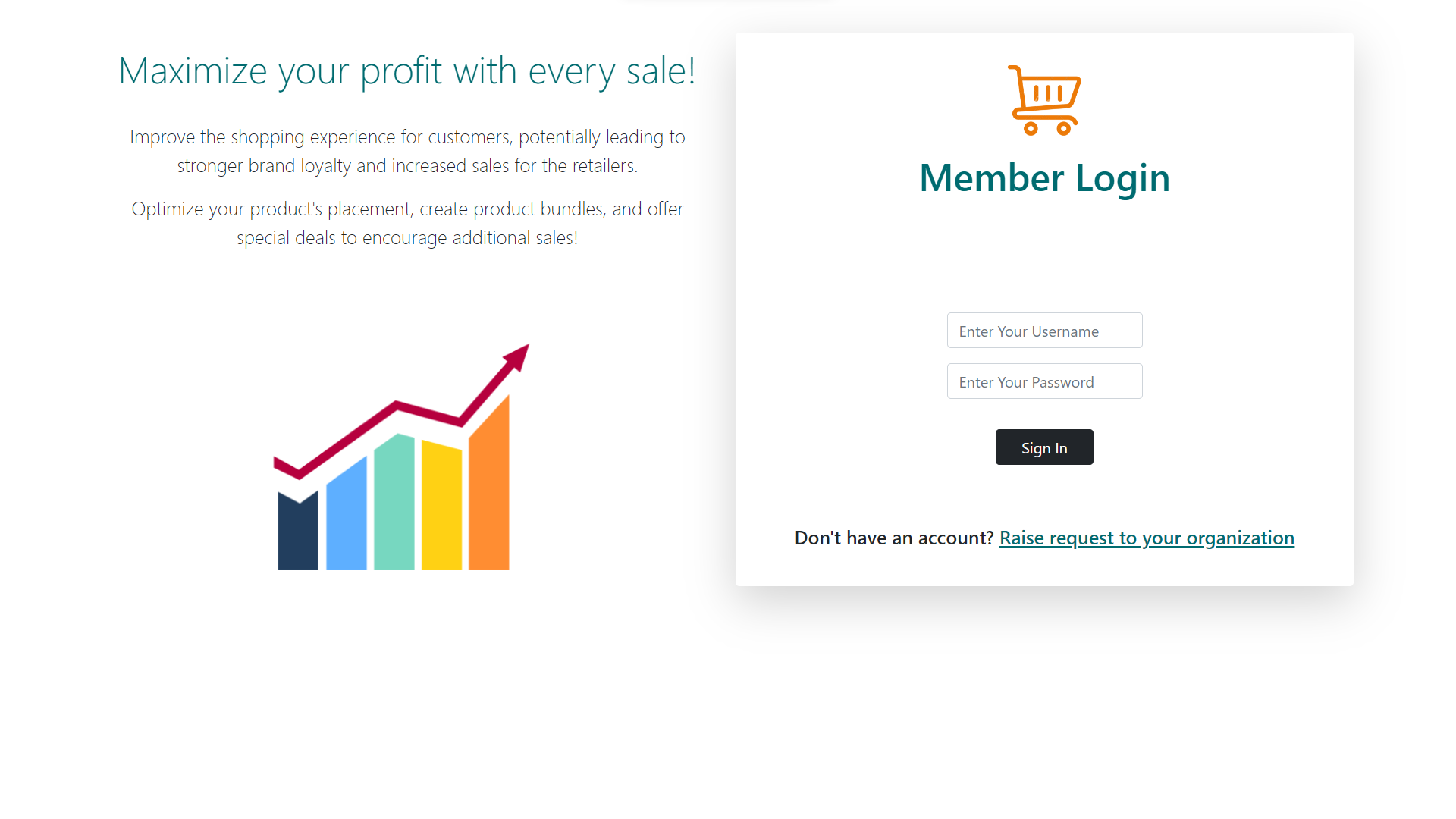


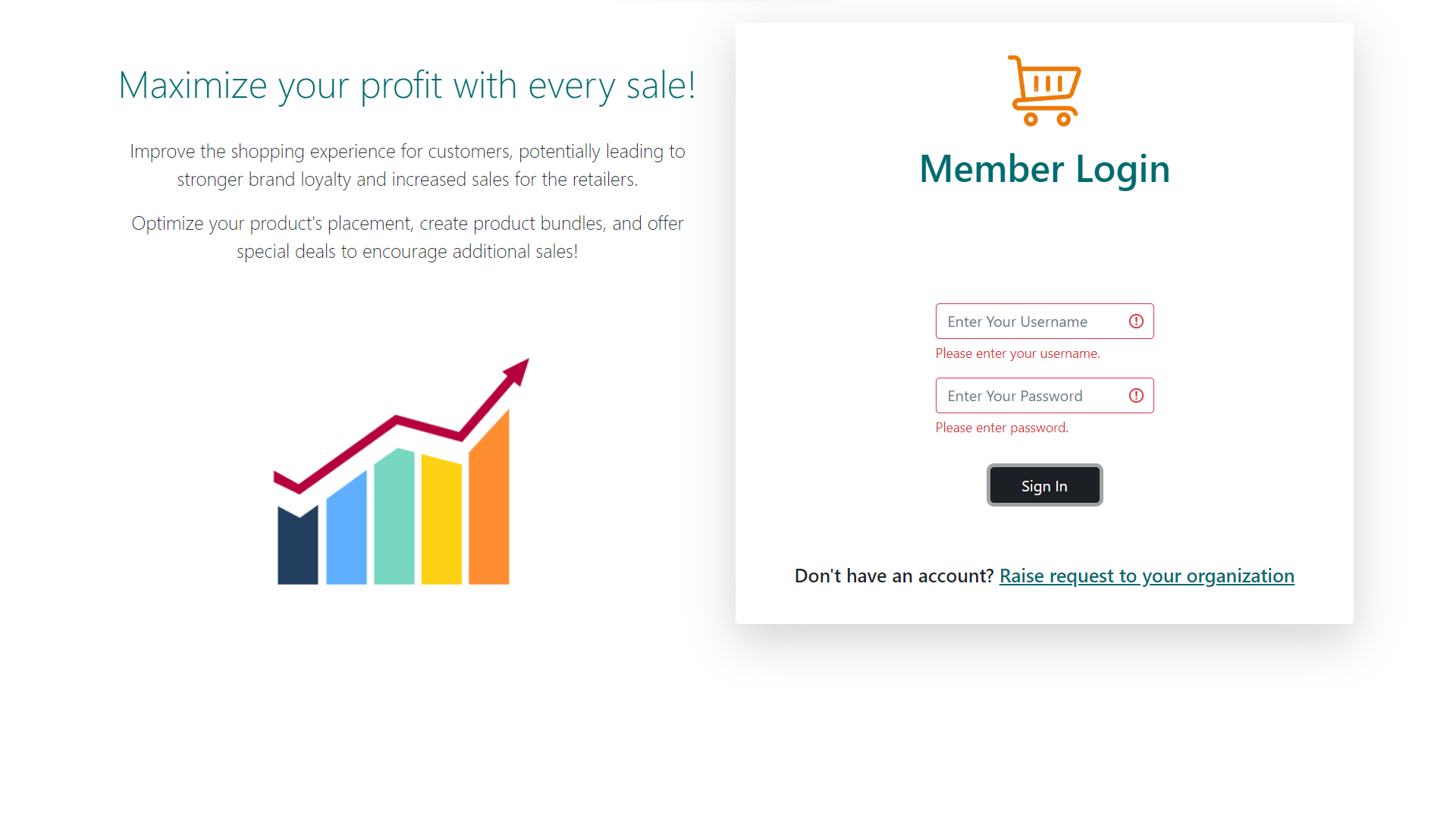
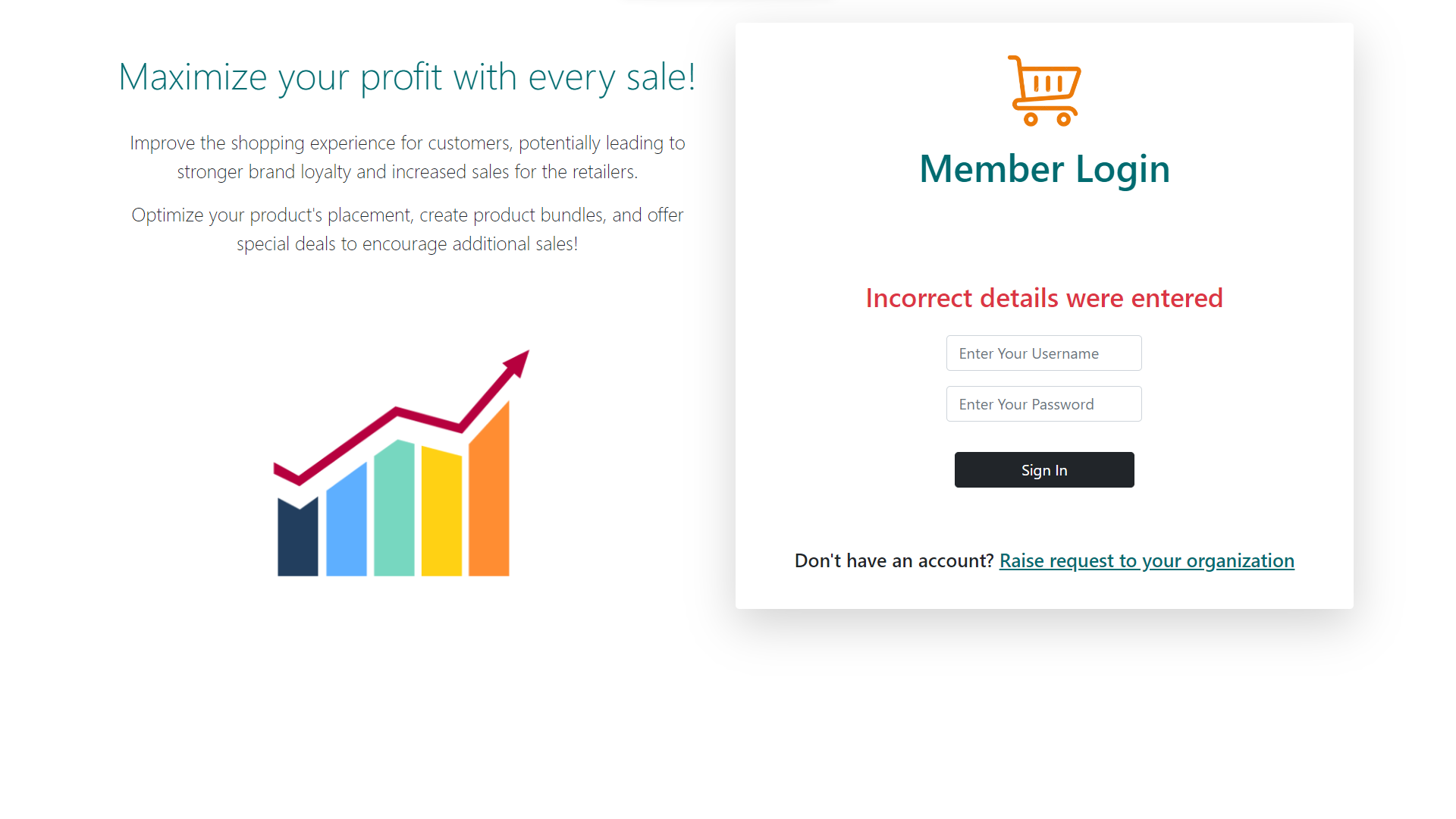
# 3.8 Module Hierarchy Diagram

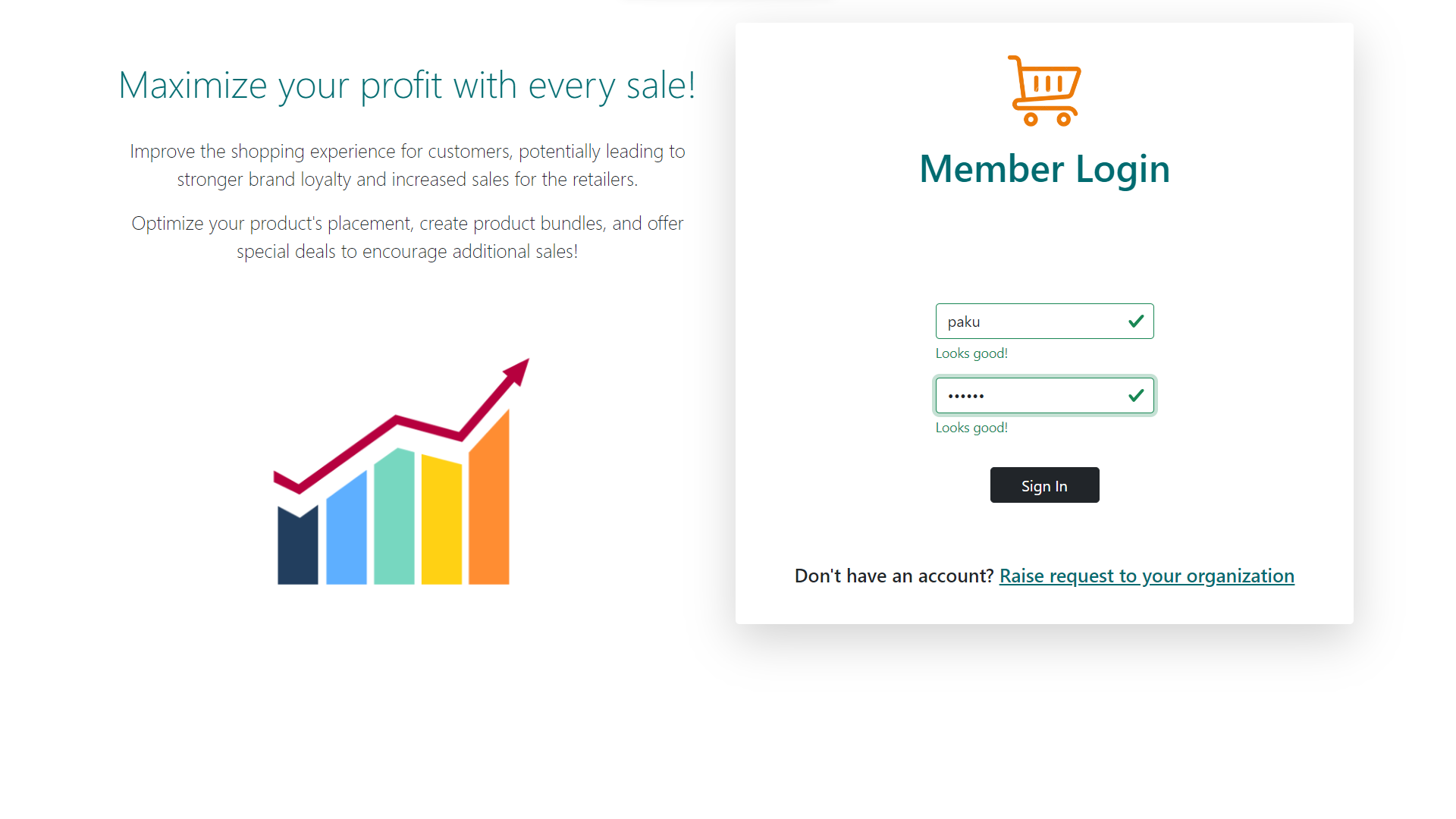


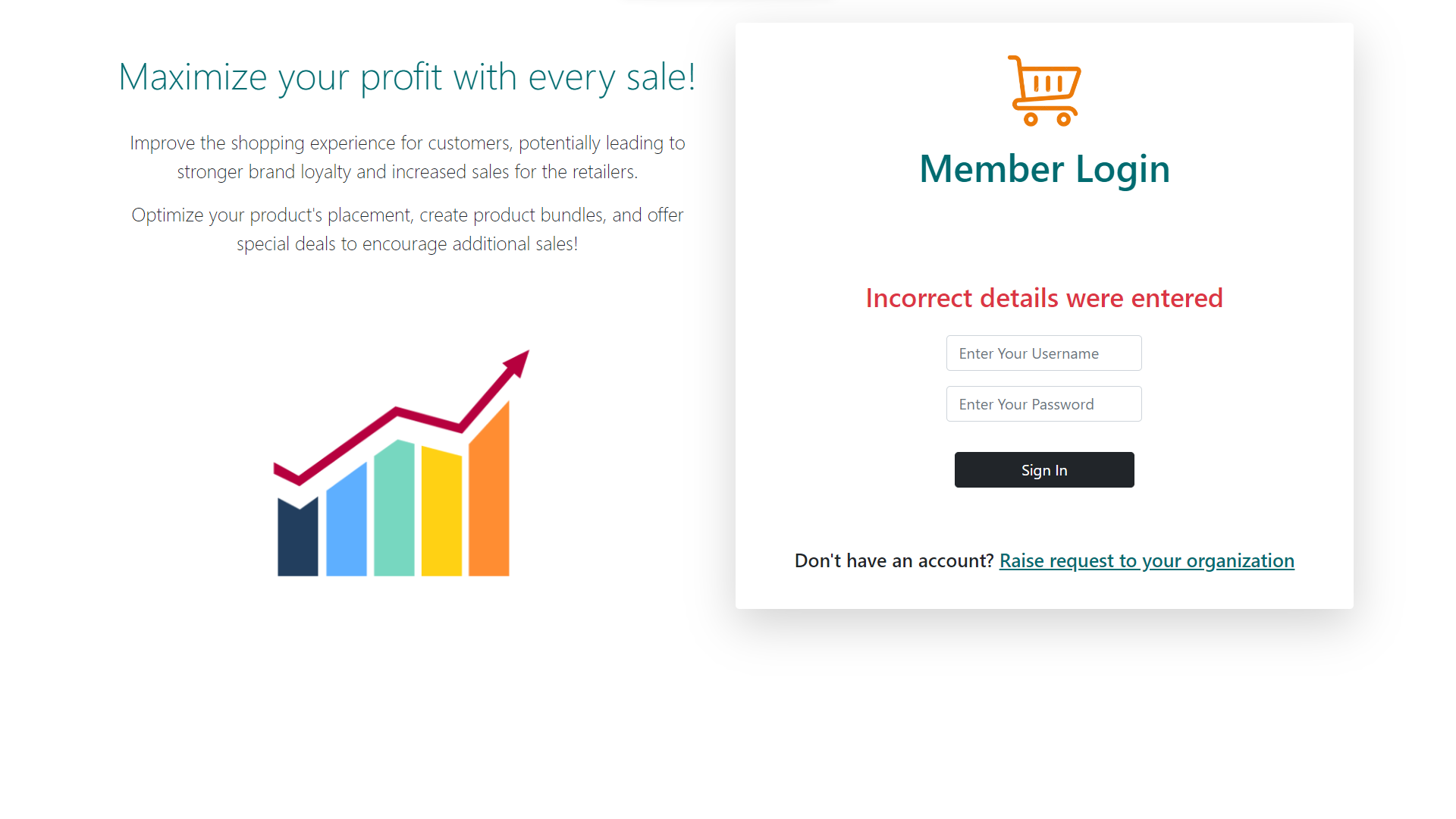
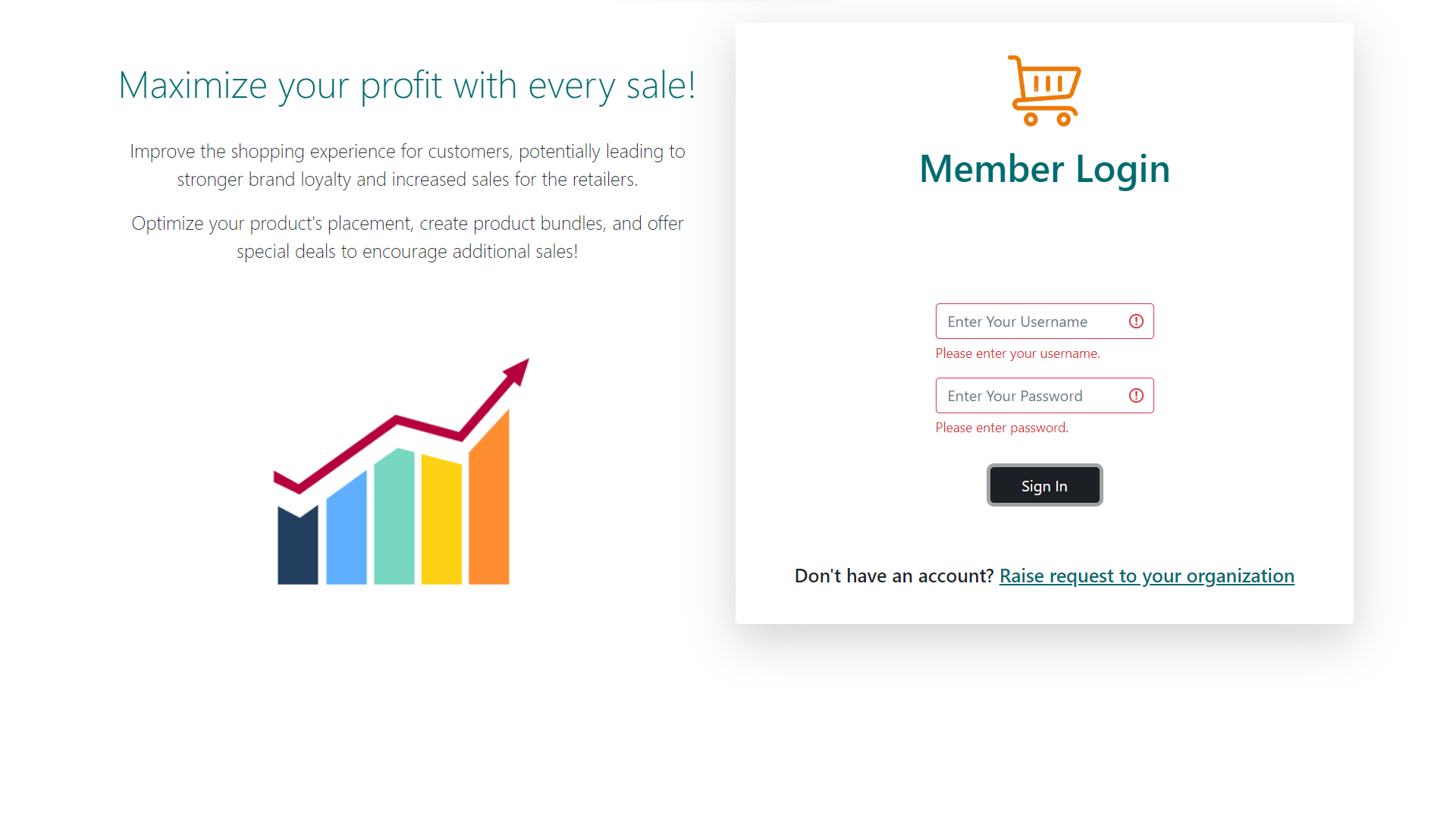
# 3.9 Sample Input and Output Screens

## 3.9.1 Login Page

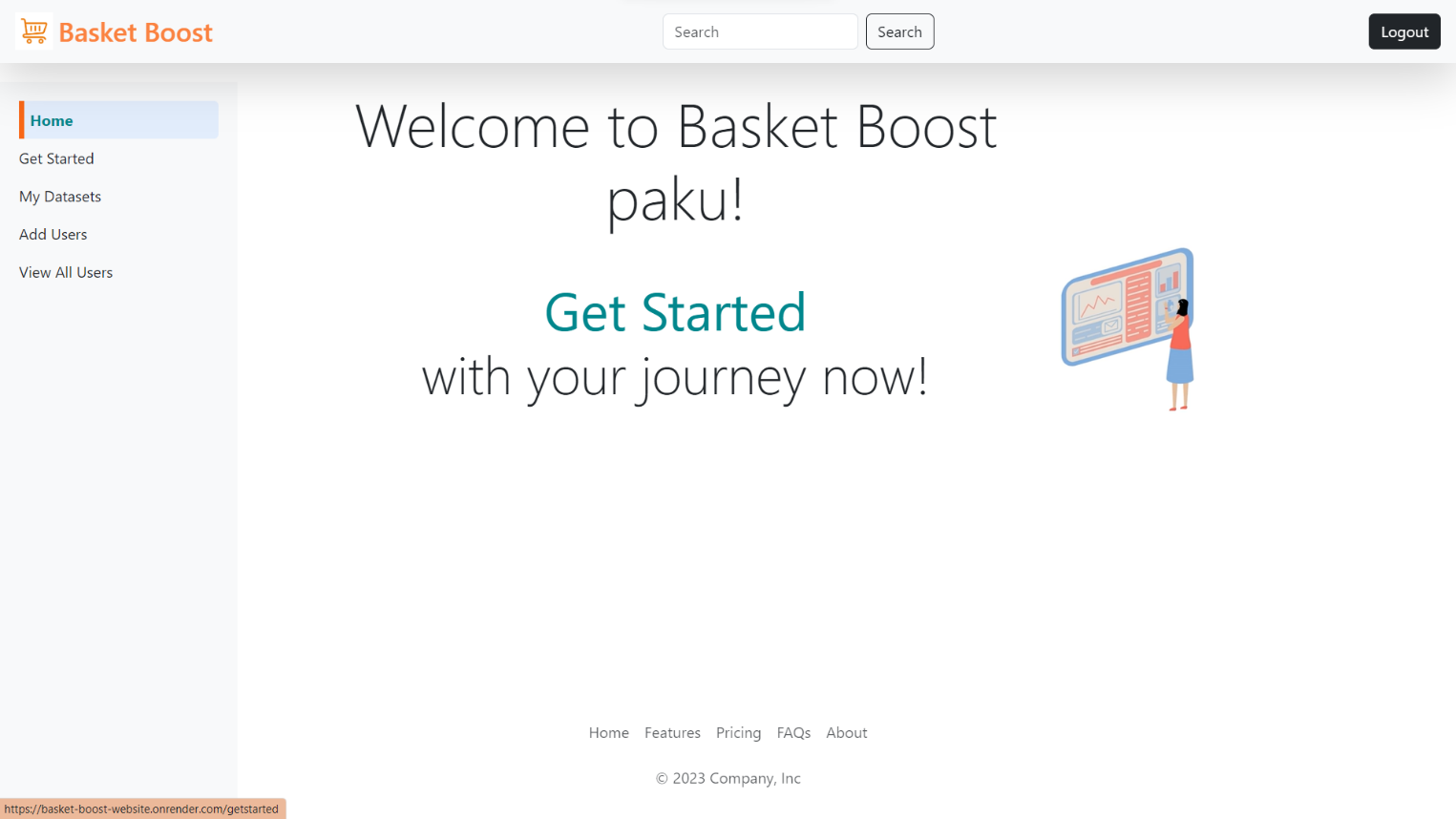
****

****

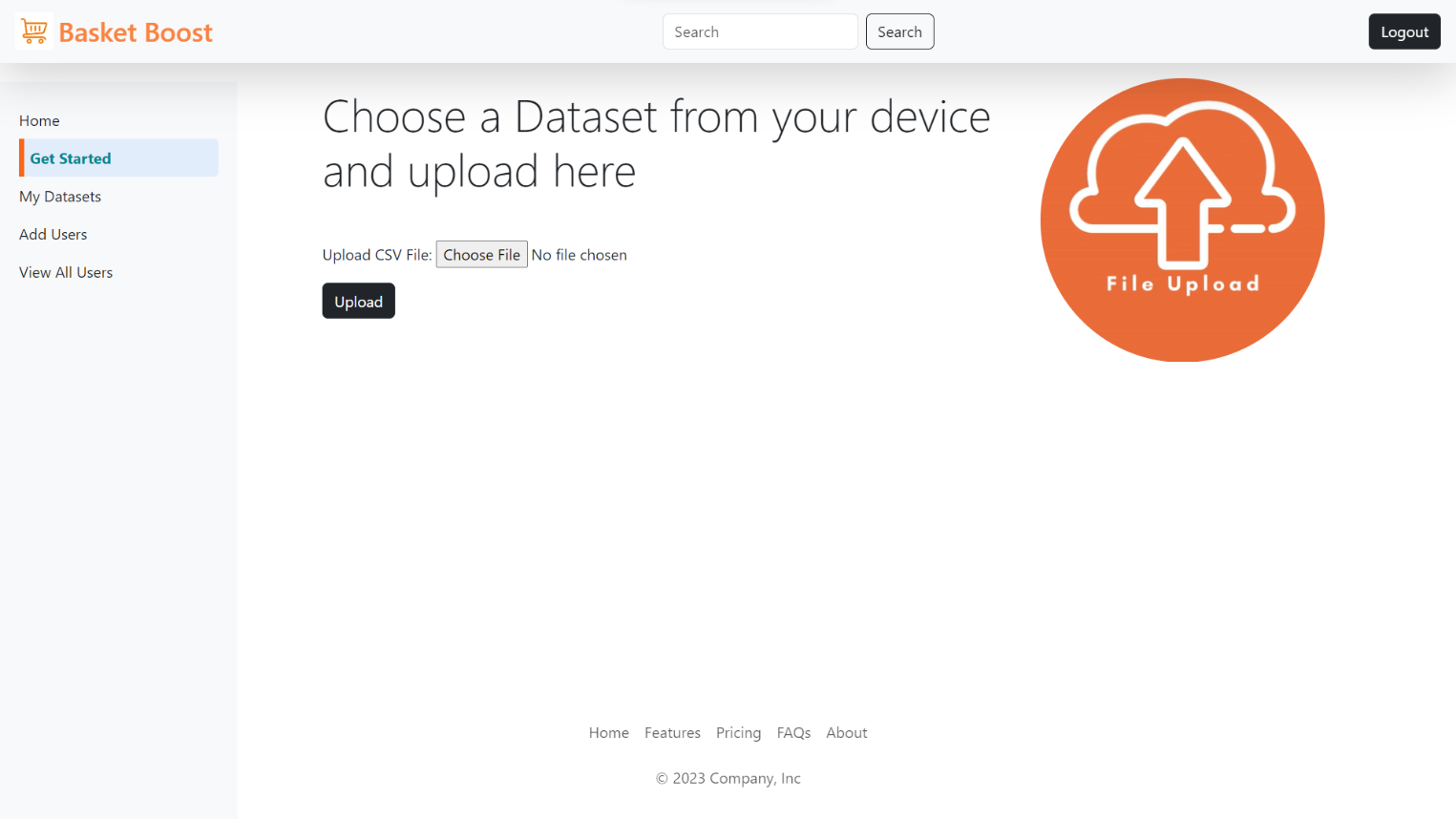
****

****

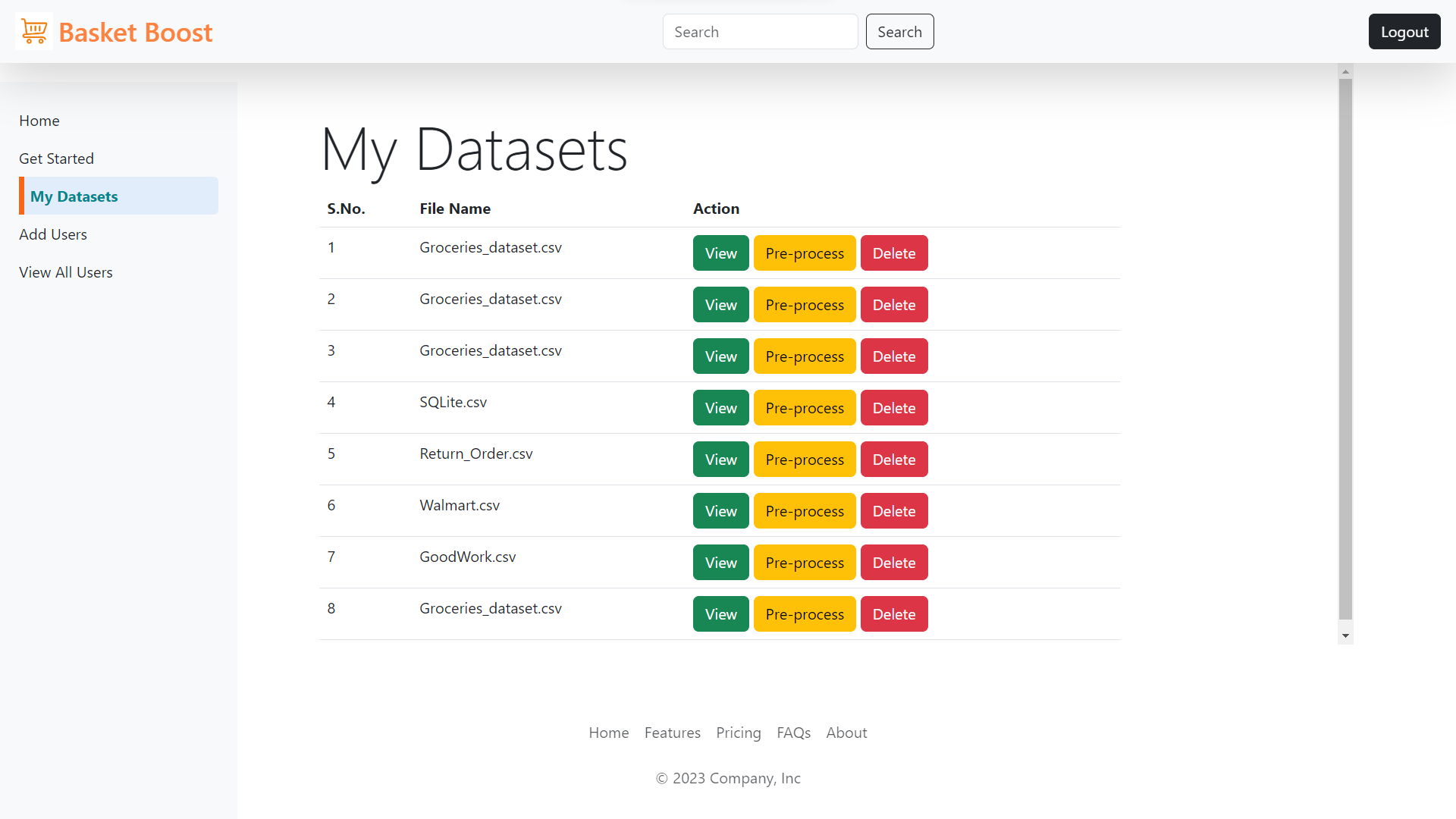
## 3.9.2 Home Page



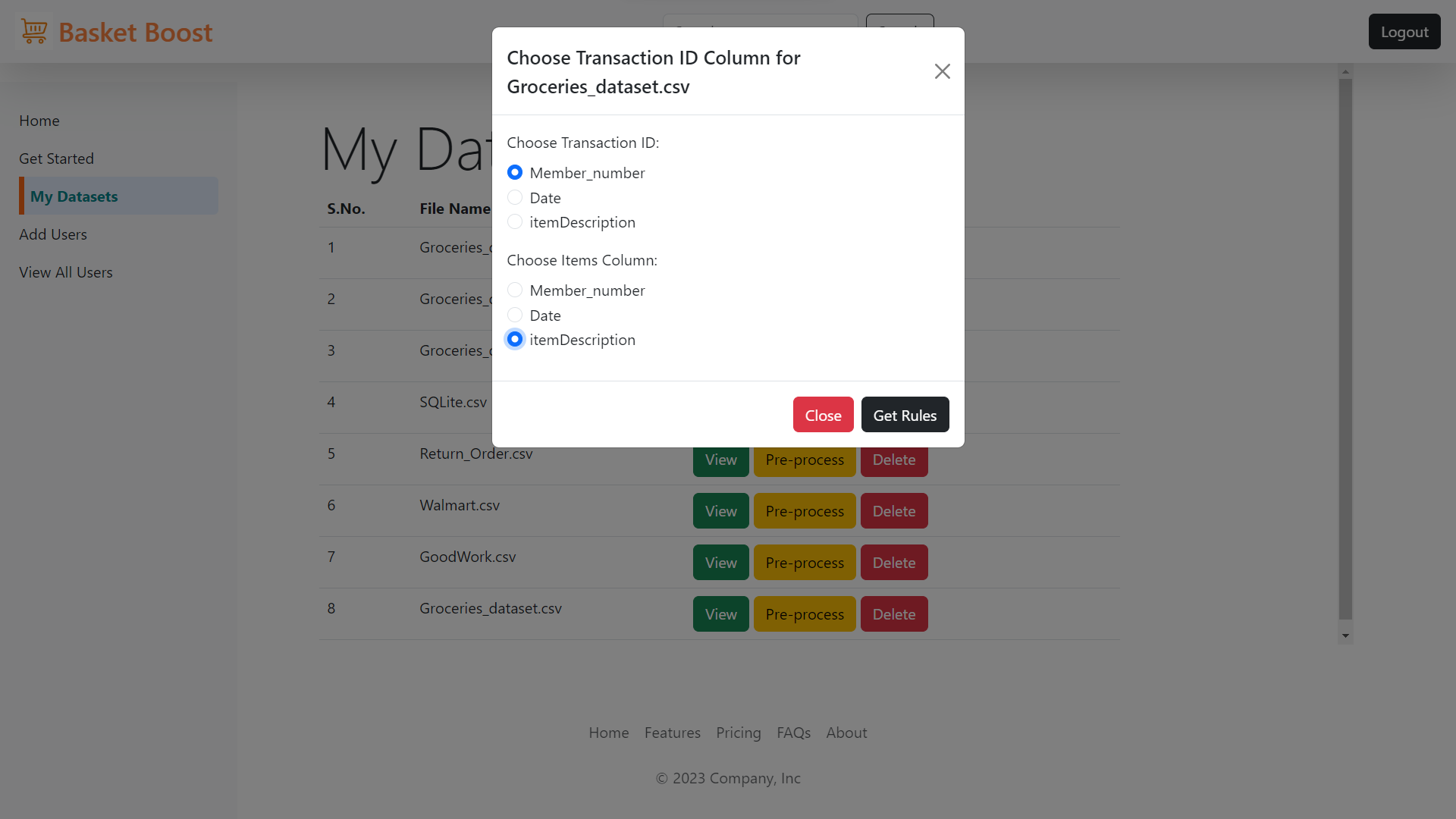
## 3.9.3 Get Started Page



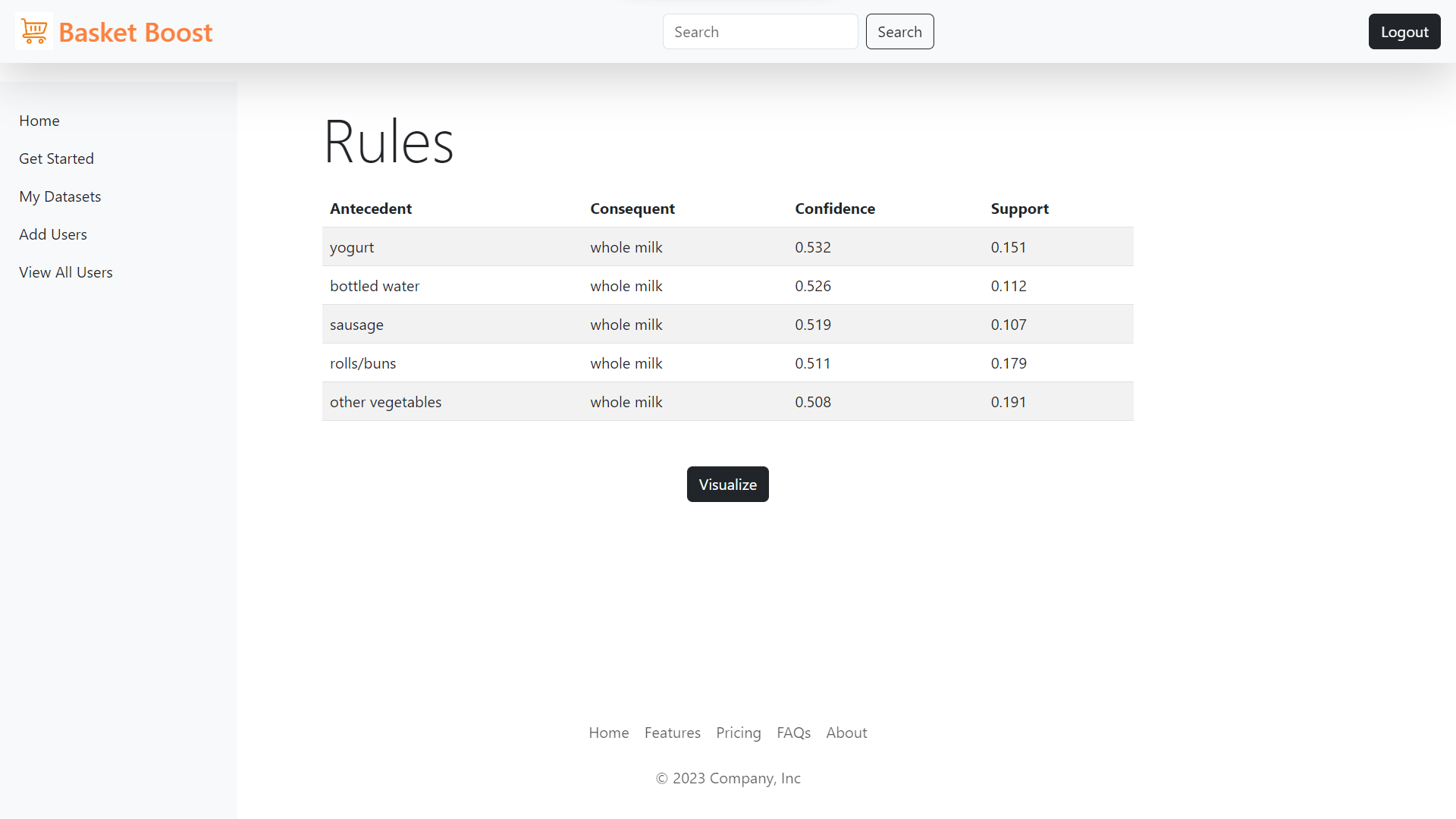
## 3.9.4 My Datasets



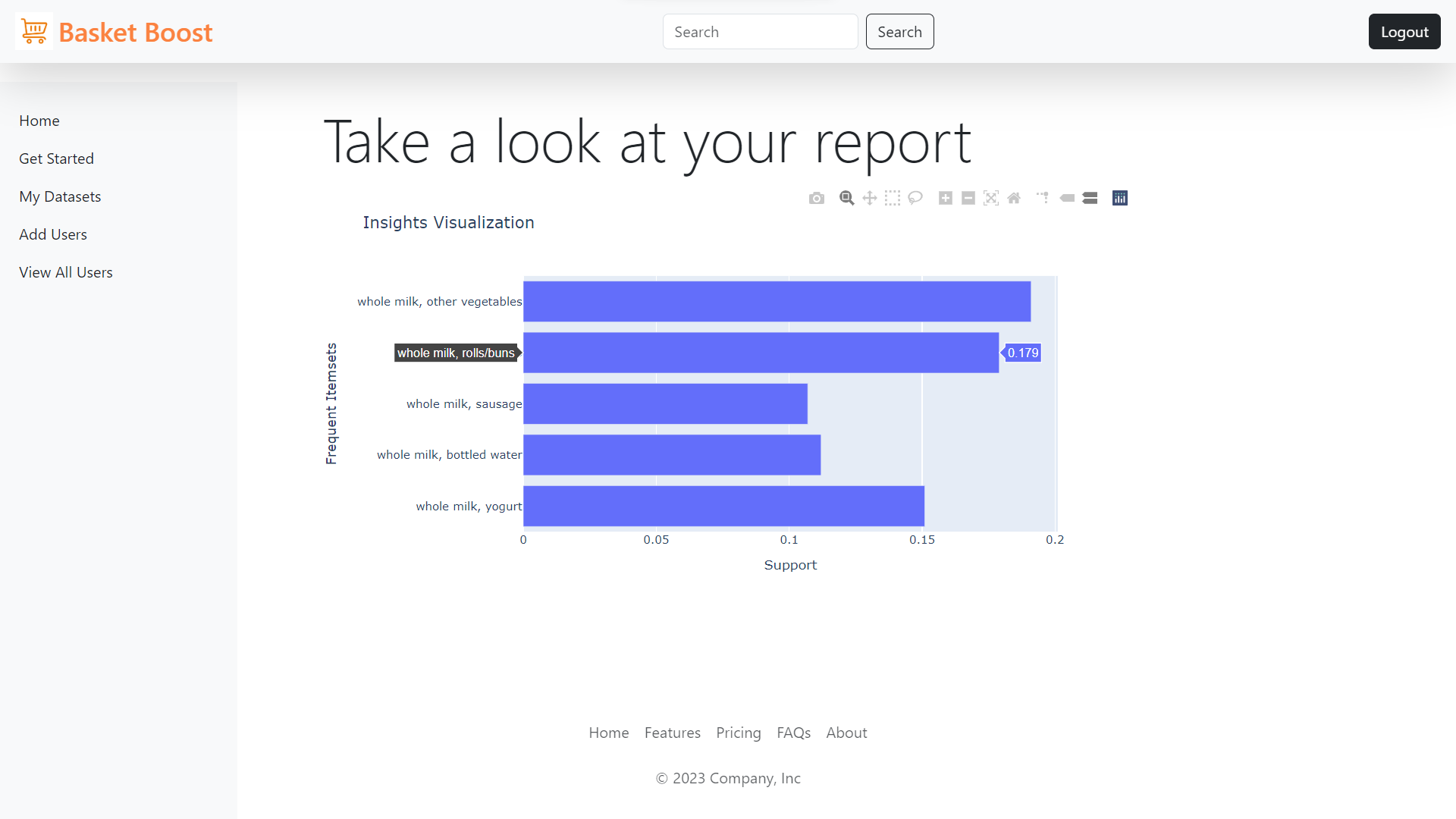
## 3.9.5 Pre-Process Data



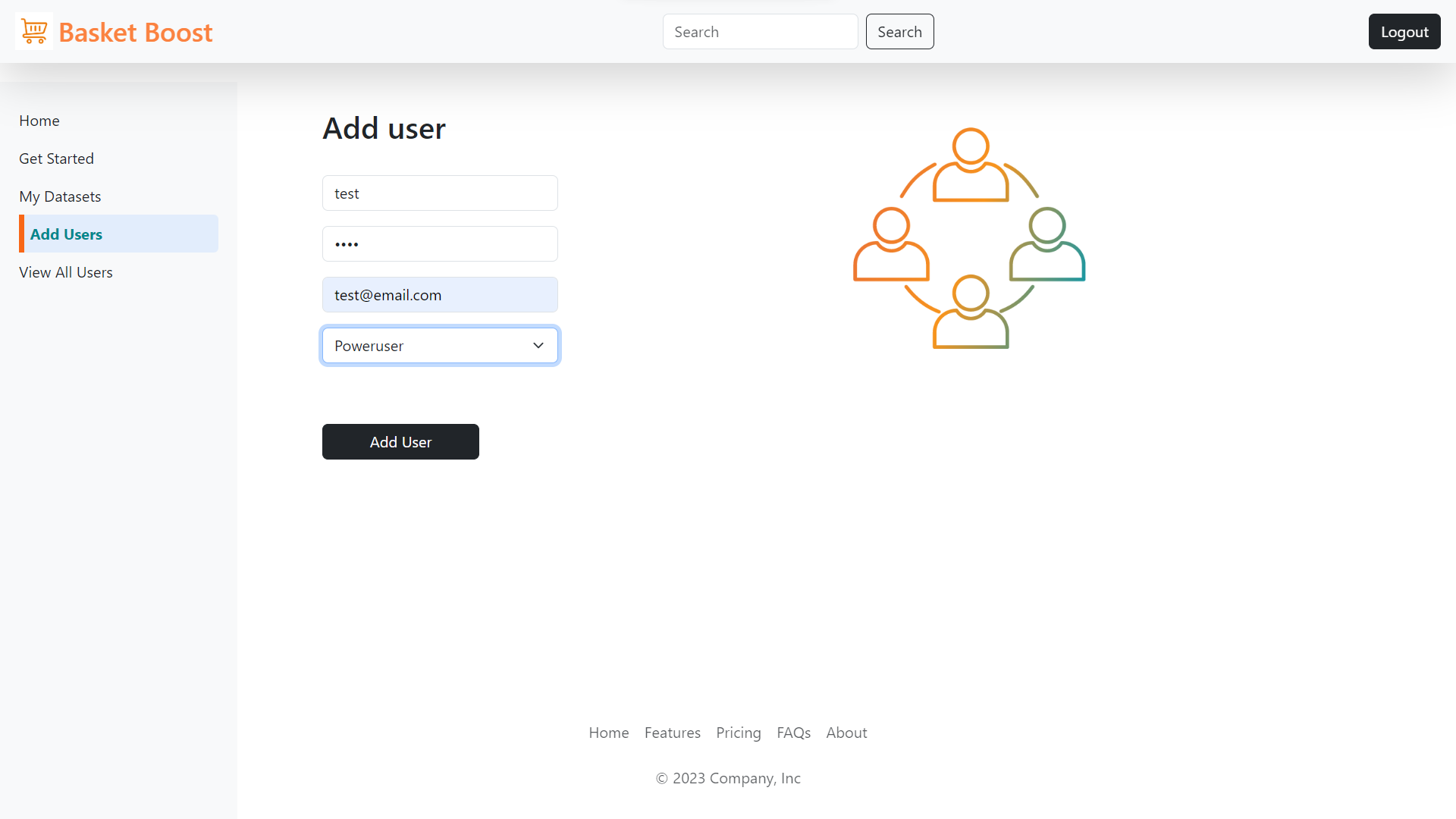
## 3.9.6 Get Rules



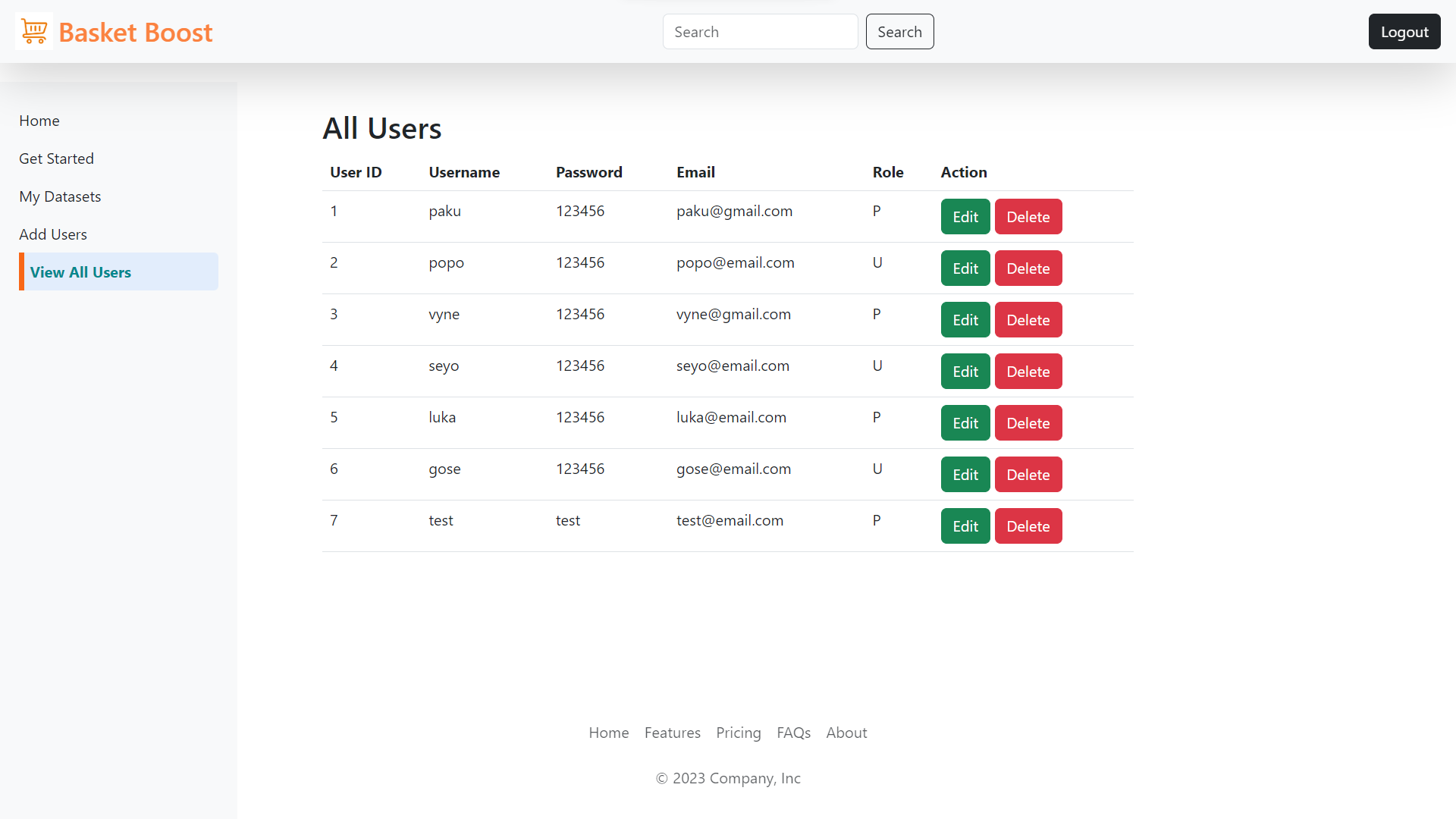
## 3.9.6 Visualize



## 3.9.7 Add Users



## 3.9.8 View all Users



Chapter No 4

**CODING**

# 4.1 Code Snippet

## File: app.py

from flask import Flask, render\_template, session, request, redirect, url\_for

from database import engine

from sqlalchemy import text

import re

import os

from database import authenticate\_user, load\_user\_byname\_byemail, load\_all\_users\_byorg, load\_user, delete\_user\_byid, edit\_user\_byid, upload\_dbfile, show\_userdb, load\_file, delete\_file\_byid

from flask import jsonify

import json

import pandas as pd

from mlxtend.preprocessing import TransactionEncoder

from mlxtend.frequent\_patterns import apriori, association\_rules

import plotly.graph\_objs as go

app = Flask(\_\_name\_\_)

app.secret\_key = "your secret key"

# Set the temporary upload folder

UPLOAD\_FOLDER = 'static/mydb'

app.config['UPLOAD\_FOLDER'] = UPLOAD\_FOLDER

@app.route('/')

@app.route('/login', methods=["GET", "POST"])

def login():

message = ''

role = 'U'

if (request.method == 'POST' and "username" in request.form

and "password" in request.form):

username = request.form["username"]

password = request.form["password"]

result = authenticate\_user(username, password)

if "Login successful" in result:

session['user\_id'] = result[1]

session['username'] = username

session['role'] = result[2]

session['org\_id'] = result[3]

# role = result[1]

return redirect(url\_for('index', message='Login Success', role=role))

else:

message = 'Incorrect details were entered'

return render\_template('login.html', message=message)

@app.route("/index")

def index():

return render\_template("index.html")

@app.route("/getstarted")

def getStarted():

return render\_template("get\_started.html")

@app.route("/addusers", methods=["GET", "POST"])

def add\_users():

message = ""

if (request.method == "POST" and "username" in request.form

and "email" in request.form and "password" in request.form

and "role" in request.form):

print("In add user if")

username = request.form["username"]

email = request.form["email"]

password = request.form['password']

role = request.form.get('role')

if role == 'Poweruser':

value = 'P'

else:

value = 'U'

org\_id = session['org\_id']

check\_exist = load\_user\_byname\_byemail(username, email)

print(check\_exist.all())

if check\_exist.rowcount > 0:

message = "email or username already exist"

print("MESSAGE ", message)

elif not username or not email or not password or not role:

message = "Please fill out the form!"

print("MESSAGE ", message)

elif not re.match(r"[^@]+@[^@]+\.[^@]+", email):

message = "Invalid email address!"

print("MESSAGE ", message)

elif not re.match(r"^[a-zA-Z0-9]+$", username):

message = "Username must contain only characters and numbers!"

print("MESSAGE ", message)

else:

with engine.connect() as conn:

query = text(

"INSERT INTO user\_details(username, password, email, role, org\_id) VALUES (:username, :password, :email, :role, :org\_id)"

)

result = conn.execute(

query, {

'username': username,

'password': password,

'email': email,

'role': value,

'org\_id': org\_id

})

message = "User added successfully"

print("MESSAGE ", message)

else:

message = ""

print("MESSAGE ", message)

return render\_template('add\_users.html', message=message)

@app.route("/allusers")

def all\_users():

org\_id = session['org\_id']

users = load\_all\_users\_byorg(org\_id)

print(users)

return render\_template('view\_all\_users.html', users=users)

@app.route('/delete/<int:user\_id>')

def delete\_user(user\_id):

# Delete the user with the given user\_id from the userdetails list

userdetails = delete\_user\_byid(user\_id)

return userdetails

@app.route('/edit/<int:user\_id>')

def edit\_user(user\_id):

user = load\_user(user\_id)

print("USER ------- ", user)

# Redirect to the edit user page for the given user\_id

return render\_template('edit\_user.html', user=user)

@app.route('/update/<int:user\_id>', methods=['GET', 'POST'])

def update\_user(user\_id):

if request.method == 'POST':

username = request.form['username']

email = request.form["email"]

password = request.form['password']

role = request.form.get('role')

if role == 'Poweruser':

value = 'P'

else:

value = 'U'

update\_details = edit\_user\_byid(user\_id, username, password, email, role)

print(update\_details)

if (session['user\_id'] == user\_id):

print("-----------IN SESSION USER ID --------")

session['role'] = role

return redirect(url\_for('index', message=update\_details))

org\_id = session['org\_id']

users = load\_all\_users\_byorg(org\_id)

return render\_template('view\_all\_users.html', user=users)

# Handle the file upload request

@app.route('/upload', methods=['POST'])

def upload\_file():

# Check if a file was uploaded

if 'file' not in request.files:

return render\_template('get\_started.html', error='No file selected.')

file = request.files['file']

# Check if the file is empty

if file.filename == '':

return render\_template('get\_started.html', error='No file selected.')

# Save the uploaded file to a temporary location

filename = file.filename

file\_path = os.path.join(app.config['UPLOAD\_FOLDER'], filename)

file.save(file\_path)

user\_id = session['user\_id']

# Save the file to the MySQL database

try:

# Read the file content

with open(file\_path, 'r') as f:

file\_content = f.read()

# Insert the file content into the database table

message = upload\_dbfile(filename, file\_content, user\_id)

datasets = show\_userdb(user\_id)

return redirect(url\_for('alldatasets', message=message, datasets=datasets))

except Exception as error:

return render\_template(

'get\_started.html',

error='Failed to upload file. Error: {}'.format(error))

finally:

# Remove the temporary file

os.remove(file\_path)

@app.route('/deletefile/<int:file\_id>')

def delete\_file(file\_id):

# Delete the file with the given file\_id from the filedetails list

user\_id = session['user\_id']

datasets = show\_userdb(user\_id)

filedetails = delete\_file\_byid(file\_id)

return redirect(url\_for('alldatasets', datasets=datasets, message=filedetails))

@app.route('/alldatasets', methods=['GET', 'POST'])

def alldatasets():

user\_id = session['user\_id']

datasets = show\_userdb(user\_id)

column = {}

for dataset in datasets:

file\_id = dataset['file\_id']

column[dataset['file\_name']] = getColumns(file\_id)

dataset['file'] = {'file\_id': file\_id, 'file\_name': dataset['file\_name']}

print(datasets)

print(column)

return render\_template('all\_datasets.html',

datasets=datasets,

column=column,

current\_file=None)

@app.route('/getColumns/<int:file\_id>', methods=['GET', 'POST'])

def getColumns(file\_id):

file = load\_file(file\_id)

file\_content = file['file\_data']

temp\_file\_path = f"static/mydb/{file['file\_name']}"

with open(temp\_file\_path, 'wb') as temp\_file:

temp\_file.write(file\_content)

df = pd.read\_csv(temp\_file\_path)

columns = df.columns.tolist()

return jsonify({'file\_name': file['file\_name'], 'columns': columns})

@app.route('/getRules/<int:file\_id>', methods=['GET', 'POST'])

def getRules(file\_id):

transactionID = request.form.get('transactionID')

itemsColumn = request.form.get('itemsColumn')

try:

file = load\_file(file\_id)

# Get the file content from the database

file\_content = file['file\_data']

# Create a temporary file path to save the content

temp\_file\_path = f"static/mydb/{file['file\_name']}"

with open(temp\_file\_path, 'wb') as temp\_file:

temp\_file.write(file\_content)

# Read the temporary file into a DataFrame

df = pd.read\_csv(

temp\_file\_path) # todo: Adjust this line if using Excel file

# Perform data cleaning

# Remove duplicates

df.drop\_duplicates(inplace=True)

# Handle missing values

df.fillna('NA', inplace=True)

# Change column data types to string

df = df.astype(str)

# Group items by transaction and create a new DataFrame with binary encoding

transaction\_data = df.groupby(transactionID)[itemsColumn].apply(

list).reset\_index(name='items')

# Perform one-hot encoding to create a binary matrix of items

one\_hot\_encoded = transaction\_data['items'].str.join('|').str.get\_dummies()

frequent\_itemsets = apriori(one\_hot\_encoded,

min\_support=0.1,

use\_colnames=True)

rules = association\_rules(frequent\_itemsets,

metric='confidence',

min\_threshold=0.5)

sorted\_rules = rules.sort\_values(by='confidence', ascending=False)

top\_rules = sorted\_rules.head(10)

myrule = []

myrule = [{

'antecedents': ', '.join(rule['antecedents']),

'consequents': ', '.join(rule['consequents']),

'confidence': round(rule['confidence'], 3),

'support': round(rule['support'], 3)

} for idx, rule in top\_rules.iterrows()]

return jsonify({'rules': myrule})

except Exception as e:

return jsonify({'error': str(e)})

@app.route('/displayRules')

def displayRules():

rules = request.args.get('rules')

rules = json.loads(rules)

# Render the template and pass the rules to display on the page

return render\_template('display\_rules.html', rules=rules)

# function to perform data visualization

@app.route('/visualize', methods=['POST'])

def dataVisualization():

rules = request.form['allrules']

# Replace single quotes with double quotes to ensure valid JSON format

rules = rules.replace("'", '"')

# Convert the rules data from string to a list using JSON decoding

rules = json.loads(rules)

# Extract the itemsets and support values from the rules

itemsets = [

rule['consequents'] + ', ' + rule['antecedents'] for rule in rules

]

support = [rule['support'] for rule in rules]

# Create the bar chart trace

data = [go.Bar(x=support, y=itemsets, orientation='h')]

# Define the chart layout

layout = go.Layout(title='Insights Visualization',

xaxis=dict(title='Support'),

yaxis=dict(title='Frequent Itemsets'),

barmode='group')

# Create the figure

fig = go.Figure(data=data, layout=layout)

# Convert the figure to JSON for rendering in HTML

chart\_json = fig.to\_json()

return render\_template('visualize\_insights.html', chart\_json=chart\_json)

# function to perform data analysis and visualization

def dataAnalysisAndVisualization(df):

rules = dataAnalysis(df)

displayResults(rules)

dataVisualization(rules)

@app.route("/logout")

def logout():

session.pop("loggedin", None)

session.pop("id", None)

session.pop("username", None)

session.pop("role", None)

session.pop("org\_id", None)

return redirect(url\_for("login")) # change this to logout success page.

if \_\_name\_\_ == "\_\_main\_\_":

app.run(host="0.0.0.0", debug=False)

Chapter No 5

**TESTING**

# 5.1 Testing Strategy

Test Strategy includes the following steps for the project:

**1. Requirement Analysis:** Understand the functional and non-functional requirements of the Market Basket Analysis Website Tool. Analyze the system's features, user interactions, and performance expectations to derive test objectives.

**2. Test Planning:** Define the overall test approach, scope, and timelines. Identify the test levels (unit testing, integration testing, system testing, etc.) and define the entry and exit criteria for each level. Determine the test environments, test data requirements, and the allocation of resources.

**3. Test Design:** Identify the test scenarios and test cases based on the system requirements. Develop test cases that cover all critical functionalities, boundary conditions, error handling, and integration points. Create test data that represents different usage scenarios.

**4. Test Environment Setup:** Set up the required test environment, including hardware, software, and database configurations. Ensure the availability of necessary tools for test execution, defect tracking, and test management.

**5. Test Execution:** Execute the test cases based on the planned test levels and test cycles. Log defects and track their resolution. Monitor test progress and report any deviations from the expected results. Conduct regression testing to ensure that modifications or fixes do not introduce new issues.

**6. Defect Management:** Track and manage defects throughout the testing process. Prioritize and classify defects based on their severity and impact. Communicate the status of defects to the development team and ensure timely resolution. Retest the fixed defects to confirm their resolution.

**7. Test Reporting and Documentation:** Generate test reports to provide visibility into the testing progress, coverage, and defects found. Document test results, including executed test cases, test data used, and identified issues. Capture lessons learned and suggestions for improvement.

**8. Performance Testing:** Conduct performance testing to evaluate the system's responsiveness and scalability. Define performance benchmarks and simulate various load conditions. Monitor resource utilization, response times, and system behaviour under different load levels.

**9. Security Testing:** Perform security testing to identify vulnerabilities and ensure the system's protection against unauthorized access, data breaches, and other security threats. Test authentication mechanisms, data encryption, and access controls.

**10. User Acceptance Testing (UAT):** Engage end-users or representatives from the target audience to perform UAT. Let them validate the system's usability, functionality, and user experience. Gather feedback and make necessary refinements before the final release.

**11. Test Closure:** Evaluate the overall test results, including coverage, quality, and achieved objectives. Conduct a test closure meeting to discuss the test outcomes, lessons learned, and recommendations for future improvements. Archive the test artifacts for future reference.

By following these steps, the test strategy for the Market Basket Analysis Website Tool ensures thorough testing, defect identification, and overall quality assurance throughout the development lifecycle.

# 5.2 Test Case

**Test Case 1:** Login Authentication

**Test Scenario:** Verify the login authentication process

**Test Steps:**

1. Navigate to the login page.

2. Enter valid credentials (username and password) and click on the "Login" button.

3. Verify that the user is successfully logged in and directed to the dashboard/homepage.

4. Verify that an error message is displayed if invalid credentials are provided.

**Expected Results:**

* For valid credentials, the user should be logged in successfully.
* For invalid credentials, an appropriate error message should be displayed.

**Test Case 2:** Username Validation

**Test Scenario:** Verify username validation for only containing letters and numbers

**Test Steps:**

1. On the registration or profile update page, enter a username containing only letters and numbers.

2. Submit the form or proceed to the next step.

3. Verify that the username is accepted and no error message is displayed.

4. Repeat the above steps with a username containing special characters or symbols.

5. Verify that an error message is displayed indicating that only letters and numbers are allowed.

**Expected Results:**

* A username containing only letters and numbers should be accepted.
* A username containing special characters or symbols should be rejected with an appropriate error message.

**Test Case 3:** File Type Validation

**Test Scenario:** Verify file type validation for uploaded files (only CSV allowed)

**Test Steps:**

1. Go to the file upload section of the application.

2. Attempt to upload a file with a valid CSV or Excel file format.

3. Verify that the file is successfully uploaded without any error message.

4. Repeat the above step with a file of an invalid format (e.g., PDF, Word document).

5. Verify that an error message is displayed indicating that only CSV or Excel files are allowed.

**Expected Results:**

* Uploading a valid CSV or Excel file should be successful without any error message.
* Attempting to upload a file with an invalid format should display an appropriate error message**.**

Chapter no 6

**LIMITATIONS OF PROPOSED SYSTEM**

**1. Scalability:** The proposed system may face scalability limitations when dealing with a large volume of data. As the dataset and user base grow, the system may experience performance issues and slower response times. It may require optimization or infrastructure upgrades to handle increased load efficiently.

**2. Dependency on Internet Connection:** The proposed system heavily relies on an Internet connection for accessing the Market Basket Analysis Website Tool. In case of network disruptions or limited connectivity, users may experience difficulties in accessing the system or performing desired actions.

**3. Compatibility:** The system may have limitations in terms of compatibility with different web browsers, operating systems, or devices. Certain features or functionalities may not work optimally or may require additional configurations on specific platforms, which could impact the user experience.

**4. Data Accuracy:** The accuracy of the insights generated by the Market Basket Analysis Website Tool is dependent on the quality and integrity of the input data. If the data used for analysis is incomplete, inconsistent, or contains errors, it may affect the accuracy of the results and insights obtained.

**5. Security Risks:** The proposed system should implement appropriate security measures to protect user data, especially when handling sensitive information such as login credentials, transaction details, or organizational data. Failure to implement robust security measures could expose the system to potential breaches, data leaks, or unauthorized access.

**6. Learning Curve:** Users who are new to market basket analysis or unfamiliar with the tool's features may require some time to understand and learn how to effectively utilize the system. Adequate user training and documentation should be provided to mitigate this limitation.

**7. Limited Analytical Techniques:** The proposed system may offer a specific set of analytical techniques for market basket analysis. Advanced or specialized techniques may not be available in the tool, limiting the depth and complexity of the analysis that can be performed.

**8. Maintenance and Updates:** The system requires regular maintenance and updates to address bug fixes, security patches, and compatibility issues. Failure to maintain the system may result in performance degradation, increased vulnerability to security threats, and compatibility conflicts with newer technologies.

Chapter No 7

**PROPOSED ENHANCEMENTS**

**1. Enhanced Data Visualization:** Improve the data visualization capabilities of the Market Basket Analysis Website Tool. Implement interactive charts, graphs, and visual representations to present the insights in a more intuitive and user-friendly manner. This enhancement will facilitate better data understanding and decision-making for users.

**2. Advanced Analytics Techniques:** Incorporate advanced analytics techniques into the tool to provide users with more sophisticated and powerful analysis options. This could include techniques like association rule mining, clustering, predictive modelling, or machine learning algorithms. These enhancements will enable users to extract deeper insights and make more accurate predictions.

**3. Real-time Analysis:** Implement real-time analysis capabilities to allow users to perform market basket analysis on streaming data. This enhancement will enable businesses to react quickly to changing customer preferences, identify emerging trends, and make data-driven decisions in real-time.

**4. Integration with External Systems:** Integrate the Market Basket Analysis Website Tool with external systems such as customer relationship management (CRM) platforms, e-commerce platforms, or data warehouses. This integration will enable seamless data transfer and enhance the tool's ability to leverage diverse data sources for analysis.

**5. Customization and Personalization:** Provide users with options to customize and personalize their analysis experience. Allow users to define their own metrics, parameters, and rules for market basket analysis. This enhancement will cater to specific business needs and increase user engagement and satisfaction.

**6. Collaboration and Sharing:** Incorporate collaborative features into the tool, allowing users to share insights, analysis results, and reports with team members or stakeholders. Enable collaborative decision-making and foster a more collaborative work environment.

**7. Performance Optimization:** Continuously optimize the performance of the Market Basket Analysis Website Tool to ensure faster response times, efficient data processing, and improved scalability. This enhancement will enhance the overall user experience and support handling large volumes of data.

**8. Mobile-Friendly Interface:** Develop a responsive and mobile-friendly interface for the tool, enabling users to access and utilize it seamlessly on various devices, including smartphones and tablets. This enhancement will enhance accessibility and usability, allowing users to perform analysis on the go.

**9. Enhanced Security Measures:** Strengthen the security measures of the system by implementing robust encryption techniques, secure user authentication mechanisms, and regular security audits. This enhancement will ensure the protection of sensitive data and maintain the trust of users.

**10. User Feedback and Analytics:** Incorporate mechanisms to gather user feedback and analytics to understand user behaviour, preferences, and pain points. Leverage this information to continuously improve the tool's functionality, usability, and user experience.

By implementing these proposed enhancements, the Market Basket Analysis Website Tool can become more versatile, powerful, and user-friendly, enabling businesses to gain deeper insights from their data and make informed decisions.

Chapter No 8

**CONCLUSION**

In conclusion, the Market Basket Analysis Website Tool offers significant value to businesses seeking to gain insights into customer purchasing patterns and optimize their marketing and sales strategies. Through the course of this project report, we have explored the various aspects of the tool, including its objectives, functionalities, technical implementation, and proposed enhancements.

The proposed system provides a user-friendly web-based interface that allows users to upload transactional data, perform market basket analysis, and generate valuable insights. It offers features such as association rule mining, support for different data formats, and visualization capabilities. The tool has been implemented using a tech stack comprising HTML, Bootstrap, CSS, JavaScript, Flask, Python, and MySQL database.

The project report also discussed the operating environment, including hardware and software requirements, as well as the feasibility study encompassing technical, economic, and behavioural aspects. Additionally, we conducted a study of similar systems to understand the existing landscape and identified the need for such a tool in the market.

While the proposed system demonstrates significant potential, it is important to acknowledge certain limitations. These include scalability challenges, dependency on internet connectivity, data accuracy, compatibility issues, and security risks. However, with appropriate mitigation strategies, these limitations can be addressed, and the system can be further improved to deliver optimal performance and user experience.

Moreover, the project report presented a comprehensive test strategy, unit test plan, acceptance test plan, and sample test cases to ensure the quality and reliability of the system. The defect report/test log was discussed as an essential tool for tracking and managing identified issues during testing.

Looking forward, several proposed enhancements have been outlined to enhance the tool's capabilities, including advanced analytics techniques, real-time analysis, customization options, collaboration features, and performance optimization.

Chapter No 9

**BIBLIOGRAPHY**

1. Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In Proceedings of the 20th International Conference on Very Large Data Bases (VLDB'94) (pp. 487-499).

1. Han, J., Pei, J., & Yin, Y. (2000). Mining frequent patterns without candidate generation. In Proceedings of the ACM SIGMOD International Conference on Management of Data (pp. 1-12).
2. Python Software Foundation. (2021). Python Language Reference, version 3.9.7. Retrieved from https://docs.python.org/3/reference/
3. MySQL Documentation. (2021). MySQL 8.0 Reference Manual. Retrieved from https://dev.mysql.com/doc/refman/8.0/en/
4. Flask Documentation. (2021). Flask Web Framework. Retrieved from https://flask.palletsprojects.com/
5. Bootstrap. (2021). Bootstrap Documentation. Retrieved from https://getbootstrap.com/docs/5.0/getting-started/introduction/
6. W3Schools. (2021). HTML Tutorial. Retrieved from https://www.w3schools.com/html/
7. Mozilla Developer Network. (2021). JavaScript. Retrieved from https://developer.mozilla.org/en-US/docs/Web/JavaScript
8. PlanetScale Documentation. (2023). PlanetScale Documentation. Retrieved from https://planetscale.com/docs/
9. Market Basket Analysis. (n.d.). In Data Mining Concepts. Retrieved from https://www.geeksforgeeks.org/market-basket-analysis-in-data-mining/
10. Raghupathi, W., & Raghupathi, V. (2014). Big data analytics in healthcare: promise and potential. Health Information Science and Systems, 2(1), 3.

Chapter No 10

**USER MANUAL**

# 10.1 Admin

* The admin is responsible for managing the services provided by the company (NIQ).
* The admin can add individuals from other organizations as per their requirements.
* The admin has the authority to add, delete, and update the details of individuals and their respective organizations to maintain accurate records.

# 10.2 Power user

* + The power user is a designated user within an organization added by the admin.
  + The power user has the ability to add individuals from their organization based on organizational needs.
  + The power user can upload CSV files, view all uploaded files, delete files, generate association rules from the files, and visualize the rules through charts.
  + Additionally, the power user can access and manage user details on the website, including editing and deleting user accounts.

# 10.3 User

* Users can log in using the credentials provided by the power user of their organization.
* Users have the capability to upload CSV files, view their own uploaded files, generate association rules from the files, and visualize the rules through charts.
* For any account-related queries such as password changes, role changes, or account deletion, users are advised to contact their organization's admin (power user).