ALBUKHARY INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTING AND INFORMATICS

SEMESTER 1 2024/2025

CCC2143

SOFTWARE ENGINEERING

GROUP PROJECT ASSIGNMENT (20%)

PROJECT TITLE	CircleMarket: Linking AIU Students for Sustainable Commerce	
SDG NO	9	
GROUP LEADER	Thierno Madiou Diallo	AIU22102008
GROUP MEMBERS	OSAMA ALHASANI HABIBA HASSAN HANAA BILAL ALTAHA Menia Mollomou	AIU22102293 AIU23032007 AIU22102357 AIU22102034

For Examiner's use only

ITEMS	MARKS
REPORT - 48	
PRESENTATION - 12	
TOTAL (60)	

TABLE OF CONTENTS

TABLE OF CONTENTS---------- **3**

INTRODUCTION---------- **4**

1. Project description---------- 5

2. Problem statement---------- 5

3. Project objectives---------- 6

PROJECT PLANNING - GANTT CHART----------**7**

SYSTEM REQUIREMENT ANALYSIS---------- **9**

1. User Requirements----------9

● Customer---------- 9

● Seller---------- 9

● Admin----------9

2. System Requirements---------- 9

● Functional Requirements---------- 9

○ Customer---------- 9

○ Admin---------- 10

■ Manage the dashboard (monitoring user activities, resolving disputes, or system maintenance)---------- 10

■ System Integrations----------10

● Non-Functional Requirements----------10

3. Software Process Model Adopted----------11

3. Sequence Diagram---------- **15**

1. Customer Sequence Diagram----------15

2. Seller Sequence Diagram---------- 17

3. Admin Sequence Diagram----------19

4. Architecture Design---------- 25

IMPLEMENTATION OF PROTOTYPE SYSTEM---------- 27

SYSTEM TESTING STRATEGY---------- 39

Testing Methodology---------- 39

Test Cases---------- 40

1. User Registration and Login---------- 40

2. Product Browsing---------- 41

3. Shopping Cart---------- 42

4. Checkout Process---------- 42

5. Seller Store Management---------- 43

6. Admin Dashboard---------- 44

CHALLENGES AND OPPORTUNITIES---------- 45

Challenges---------- 45

Opportunities---------- 46

ROLES AND RESPONSIBILITIES OF PROJECT MEMBERS---------- 47

CONCLUSION---------- 49

APPENDIX---------- 51

INTRODUCTION

This report presents our project, which aligns with Sustainable Development Goal 9 (SDG 9) focusing on "Industry, Innovation, and Infrastructure." The project aims to develop an online marketplace at AIU to support small businesses within the campus community. This platform will allow sellers to create and manage their stores while offering buyers an easy-to-use interface to browse and purchase products. The report details the planning, design, and implementation of the system prototype, including requirement analysis, system modeling, architecture design, testing strategies, and challenges encountered.By promoting innovation and improving infrastructure with this marketplace, the project contributes to sustainable economic growth within our academic setting.

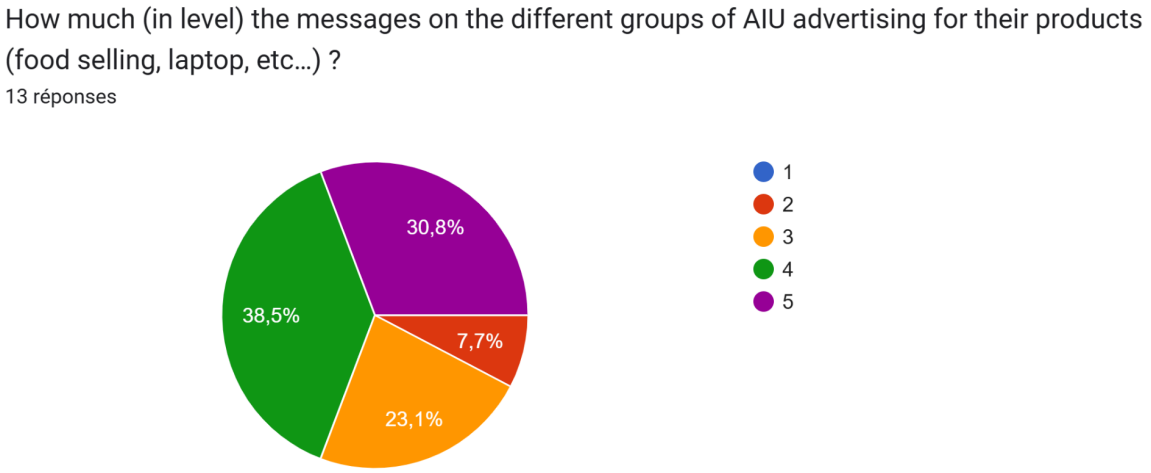
1. Project description

Sustainable Development Goal 9 (Goal 9 or **SDG 9)** is about "industry, innovation and infrastructure" and is one of the 17 Sustainable Development Goals adopted by the United Nations General Assembly in 2015 (European Commission, 2020). SDG-9 is built on three pillars, infrastructure, industry and innovation, all of which are strongly connected among themselves and share the common goal of achieving socially inclusive and environmentally sustainable economic development (HLFP Thematic review, 2017). Aligned with SDG 9, which focuses on Industry, Innovation, and Infrastructure, we aim to implement an online marketplace at AIU to support small businesses on campus. This platform will enable customers to register, browse, select, and purchase products easily. Sellers will also be able to create their own stores, manage their products, and engage with customers through an accessible admin dashboard. The software will include features for tracking product delivery until it reaches the customer. Ultimately, this project seeks to innovate the marketplace industry by providing a user-friendly platform that fosters local commerce and supports small businesses.

2. Problem statement

With the increasing number of students at AIU and the growing entrepreneurial spirit among them, many students are starting small businesses on campus. However, these student entrepreneurs face significant challenges in effectively running and promoting their businesses. Currently, most rely on platforms like Telegram and WhatsApp, which were originally designed for academic purposes rather than as dedicated marketplaces.

A good illustration of this is, a survey conducted by our team among aiu students:



As we can see here, most of the answers are high.

This lack of a proper platform creates barriers for students who wish to start selling but are discouraged by the absence of an accessible and supportive system. Additionally, inefficient communication methods and the absence of structured tools for managing their businesses have caused many students to abandon their entrepreneurial ventures

altogether. Considering all these factors, it is clear that the lack of digitalization in the ecommerce industry at AIU is a massive challenge that needs to be addressed.

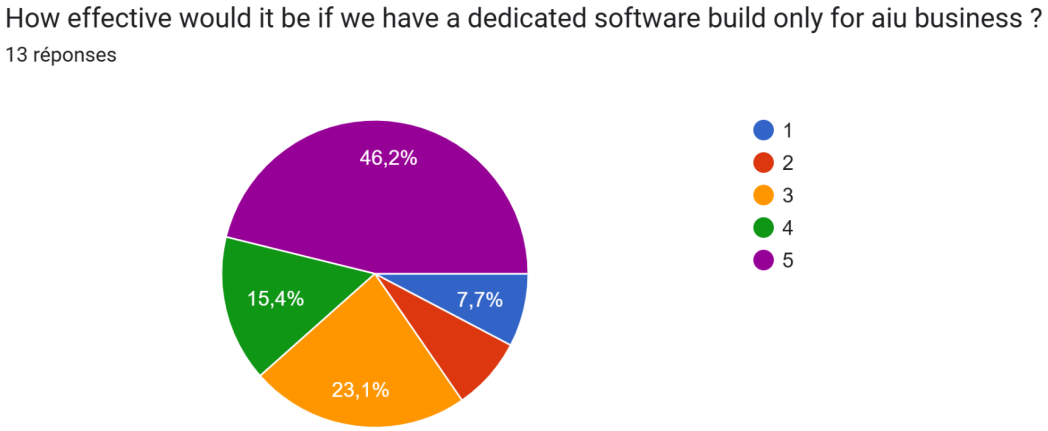
3. Project objectives

1. Establish a User-Friendly E-Commerce Platform;

2. Build a Supportive Business Community;

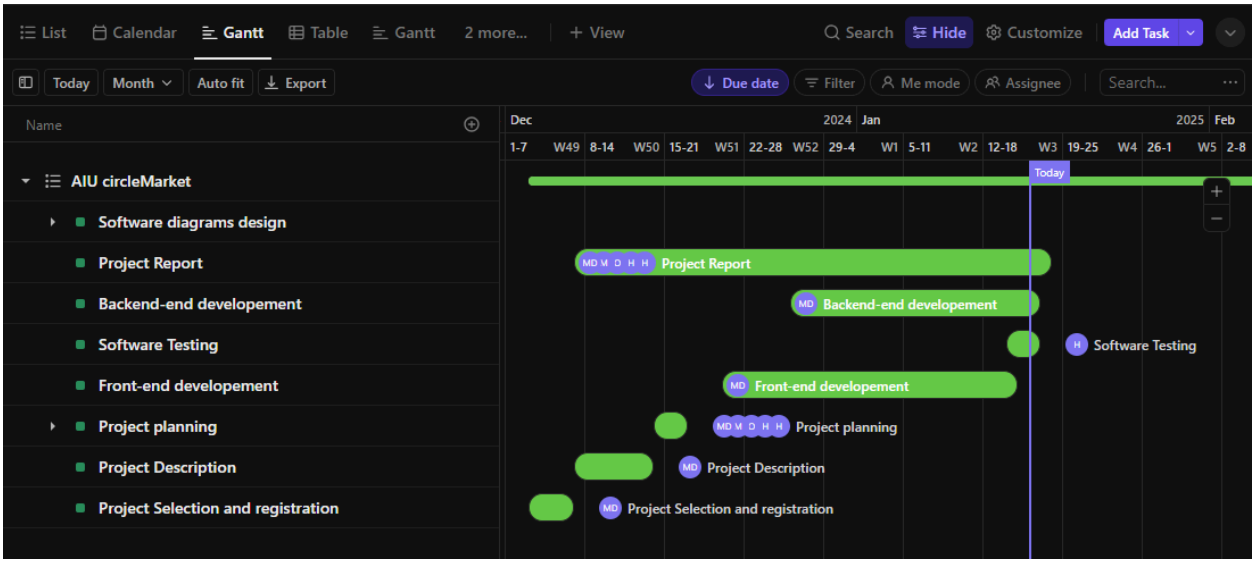
3. Encourage Sustainable Industry and Innovation;

Additionally based on the survey that we conducted, it demonstrates that this platform is highly needed.

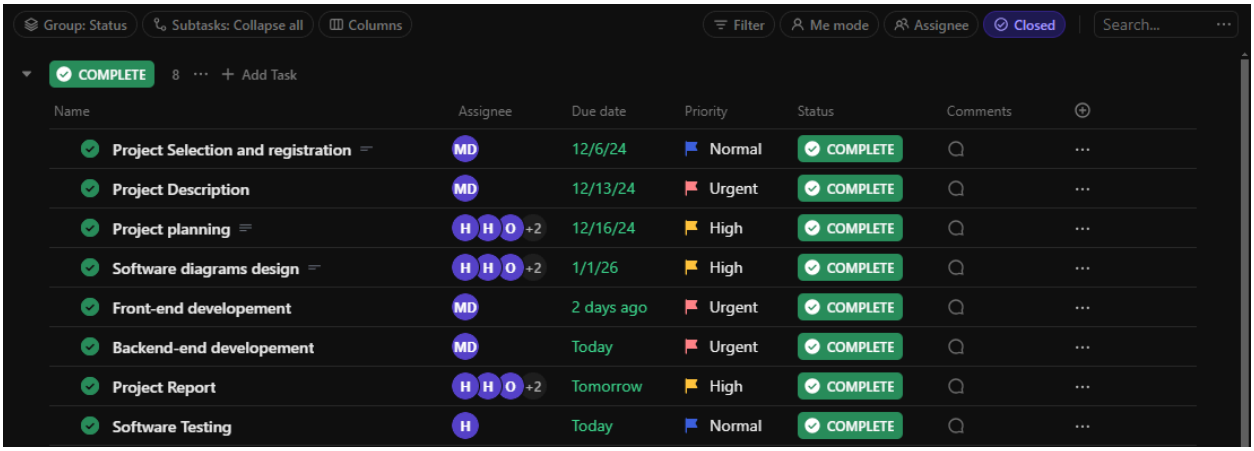


PROJECT PLANNING - GANTT CHART

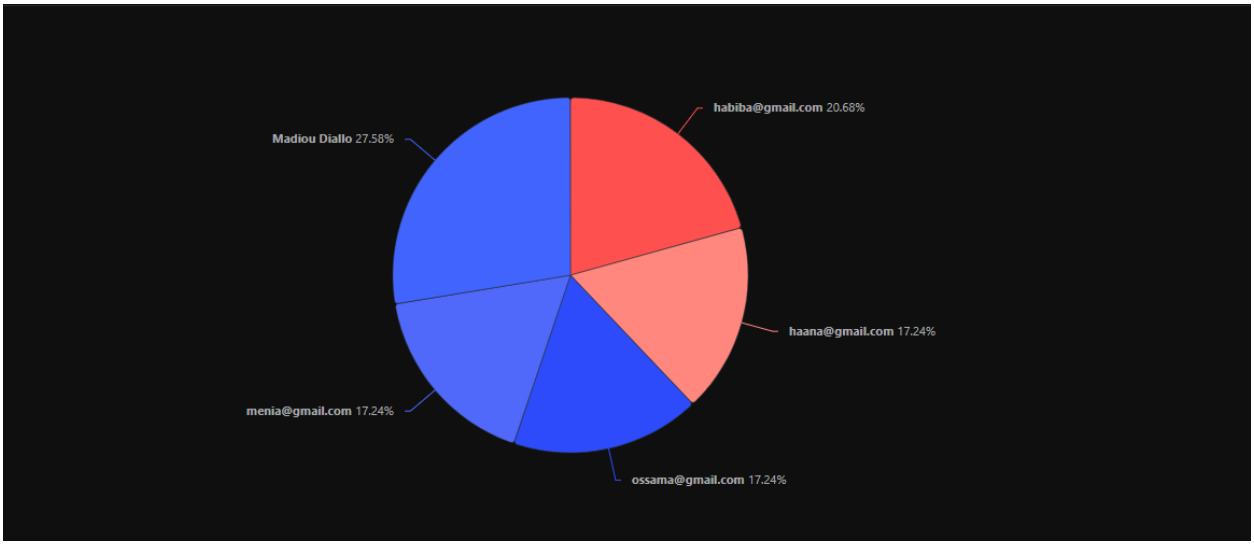
The "AIU circleMarket" list comprises a series of completed tasks that were essential in achieving various project milestones. Here's a representation visual graphic to the key tasks that were accomplished throughout the development of project:



As we can notice from the image, we have used **ClickUp** to produce a very effective Gantt chart along the way. This software is designed to manage tasks like this, more explicitly, let's have a look to the another representation of the same project



Additionally, we can see below the percentage of Task achievement of every member among the team. This graph is generated from the Gantt chart software.



Overall, the project was successfully completed with all tasks marked as complete, indicating that the objectives were met and the project was executed as planned.

SYSTEM REQUIREMENT ANALYSIS

1. User Requirements

- Customer
  - Register or log in for an account to gain access to the system
  - Browse through listed products to consider possible purchases
  - Add products to cart for future checkout
  - Follow a clear and intuitive checkout
  - Conduct the purchase using the cash method of purchase
- Seller
  - Registration or login to an account in order to manage store activities; creation and management of a store, including branding.
  - The seller can add new products, update the quantity of available products, or delete products that are not in stock.

        - Verify the purchase order availability and approve it or reject it.
        - Upon a new order, notifications pop up to assure timely responses.
    - **Admin**
        - Access to an admin dashboard in order to monitor user activities.
        - Mediate disputes between customers and sellers.
        - Perform regular system maintenance to ensure smooth operations and address issues proactively.

2. **System Requirements**
    - Functional Requirements
        - Customer
            - Register or login to an account
            - Browse products
            - Add products to the shopping cart
            - Make a purchase
            - Complete the checkout process
        - Seller
            - Register and log in to an account
            - Create a store
            - Manage products by adding, updating, or deleting items listed in the store
            - Accept or decline a purchase
            - Receive notifications when an order is placed
        - Admin
            - Manage the dashboard (monitoring user activities, resolving disputes, or system maintenance)
            - System Integrations
            - Payment Gateway: Support cash as the payment method.
            - Delivery Tracking System: Provide real-time updates on order shipments.

        - Database System: Manage user accounts, product listings, and delivery records.
- **Non-Functional Requirements**
    - Performance: The system should allow for quick browsing, buying, and tracking of product delivery by customers. For sellers, it should not take too long to manage products or create a store.
    - Scalability: The system should support more and more sellers, customers, and transactions as it grows.
    - Availability: Admin dashboard, customer interface, and seller tools must be available, with very limited downtime.
    - Usability: Easy to use, intuitive interface for product browsing, store management, and delivery tracking. Admin tools should provide easy navigation for system management tasks.
    - Security: The sensitive data of customer, seller, and admin should be protected. Payment gateway integration shall follow the standard of security.
    - Reliability: Delivery tracking should show updated information correctly. The database system should handle unexpected failures and recover from them.
    - Maintainability: The functionalities should be easy to update or extend without significant downtime.
    - Integration: Clear interfacing among delivery tracking system, payment gateway, and database system;
    - Compliance: Compliance with related legal and regulatory requirements.

3. Software Process Model Adopted

For the development of our prototype system, CircleMarket: Linking AIU Students for Sustainable Commerce, we employed a hybrid methodology that integrates the

Incremental Model with the Integration Model. This combination enabled us to construct a systematic and operational prototype that emphasises the essential elements of our proposed solution, while utilising existing components.

Incremental Model

The Incremental Model is a software development approach in which the system is designed, implemented, and tested incrementally until the entire system is completed. Each iteration delivers a functional component of the software, enabling developers to progressively construct and refine the system. This methodology prioritises breaking the development process into smaller, manageable components, each concentrating on a single feature or functionality (GeeksforGeeks, 2024a).

In our project, the Incremental Model directed the sequential advancement of our prototype. Rather than trying to implement all features simultaneously, we split the task into smaller, concentrated iterations. Each version focused on particular fundamental functionality, including user registration, product browsing, and the admin dashboard. This approach enabled us to prioritise essential elements for presenting the prototype's potential while ensuring clarity in our workflow.

Despite the system being a prototype without real-world implementation or user engagement, the incremental model allowed for the gradual construction and enhancement of the system. It provided a clear structure for showcasing the system's architecture and functionality during the project presentation.

Integration Model

The Integration and Configuration Model emphasises the reutilization of pre-existing components to construct a new system, either via informal reuse (where developers modify existing designs) or formal reuse (utilising a repository of reusable components)

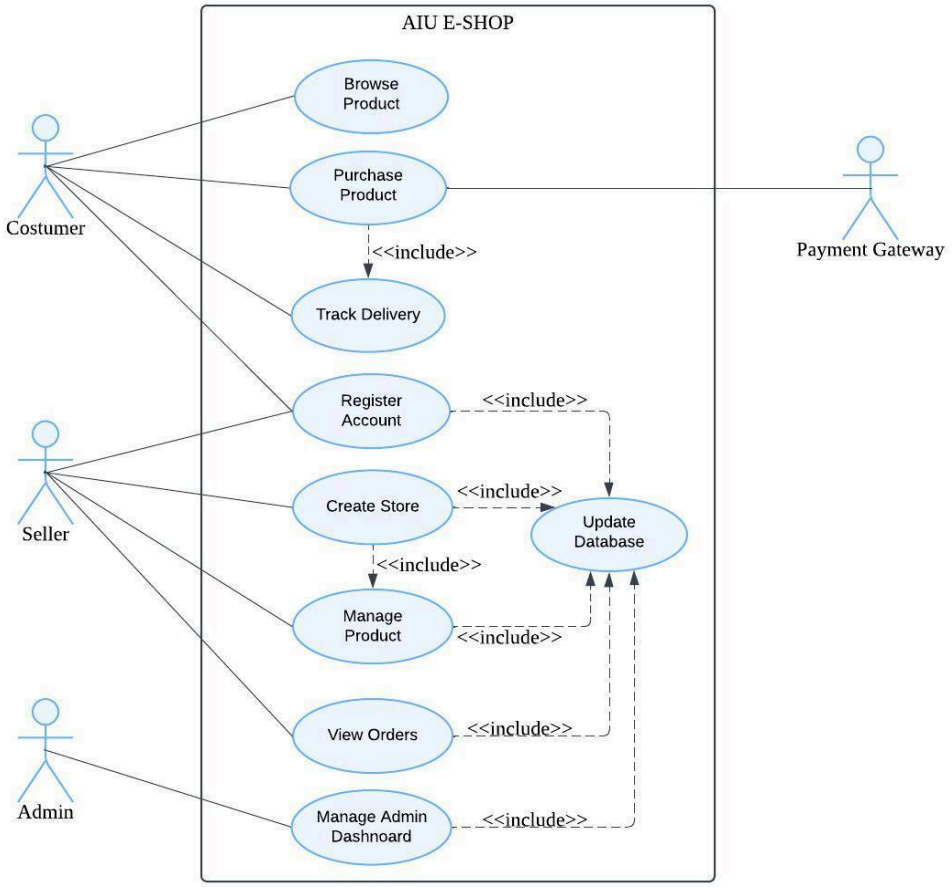
(Khose, 2024). It seeks to diminish development time and expenses by utilising pre-existing components rather than starting from the beginning (Sommerville, 2016).

In our project, we utilised the Integration Model to effortlessly integrate third-party services into the prototype. Firebase served as an essential component, managing user registration and authentication, enabling us to avoid the necessity of developing these functionalities from scratch. Additionally, we incorporated a delivery tracking system to simulate the product delivery process. These integrations not only reduced development time but also improved the prototype's stability and functionality.

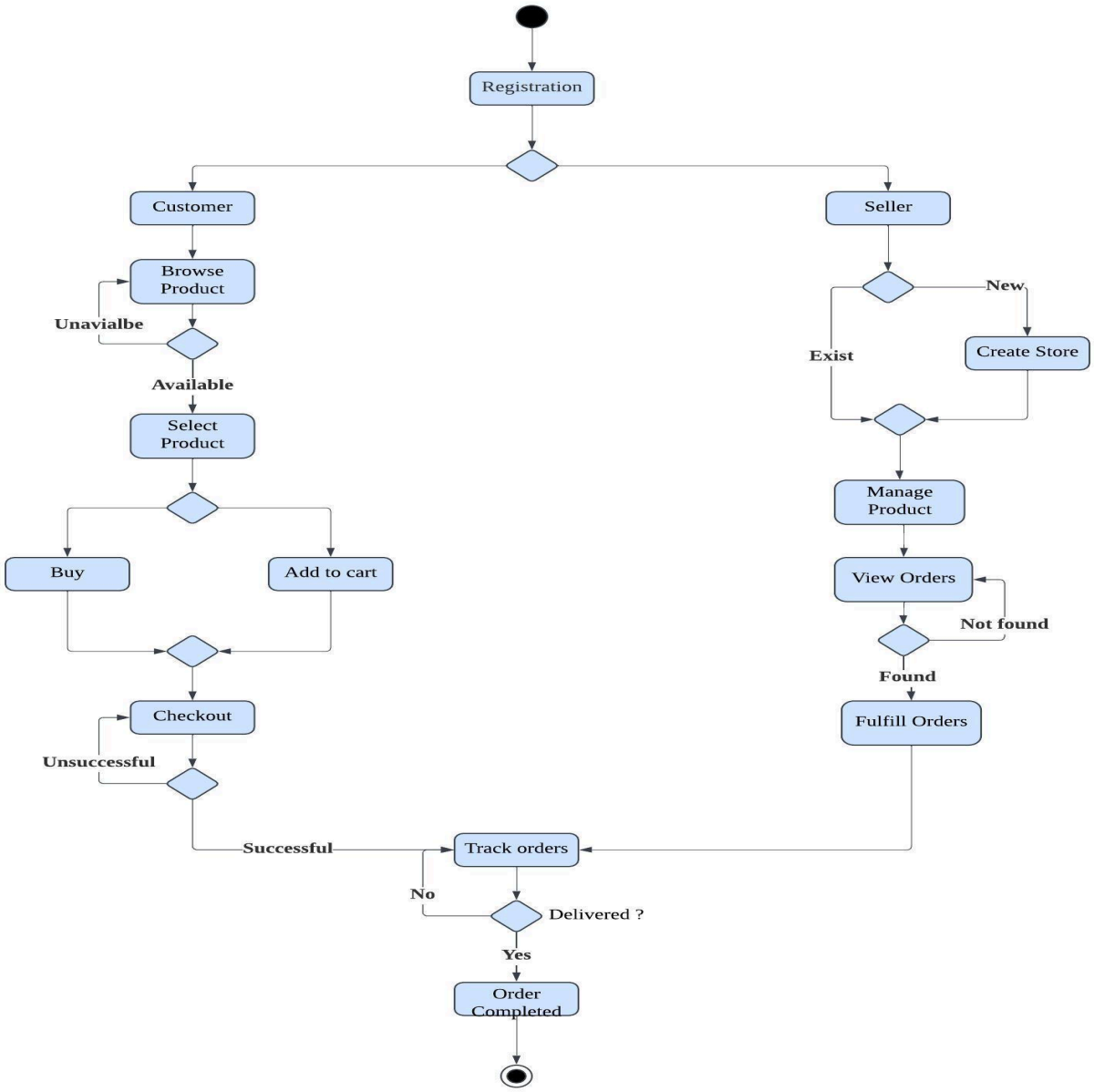
The hybrid method of combining incremental development with external integration was optimal for our requirements. The Incremental Model provided a systematic approach to prioritise and provide features that correspond with our project’s goals, whilst the Integration Model enabled us to utilise comprehensive, pre-existing solutions to address technical weaknesses in the prototype. As a whole, these models enabled us to develop a refined and operational system that effectively demonstrates the project's vision and potential impact.

System Model

1. use case diagram



2. Activity diagram

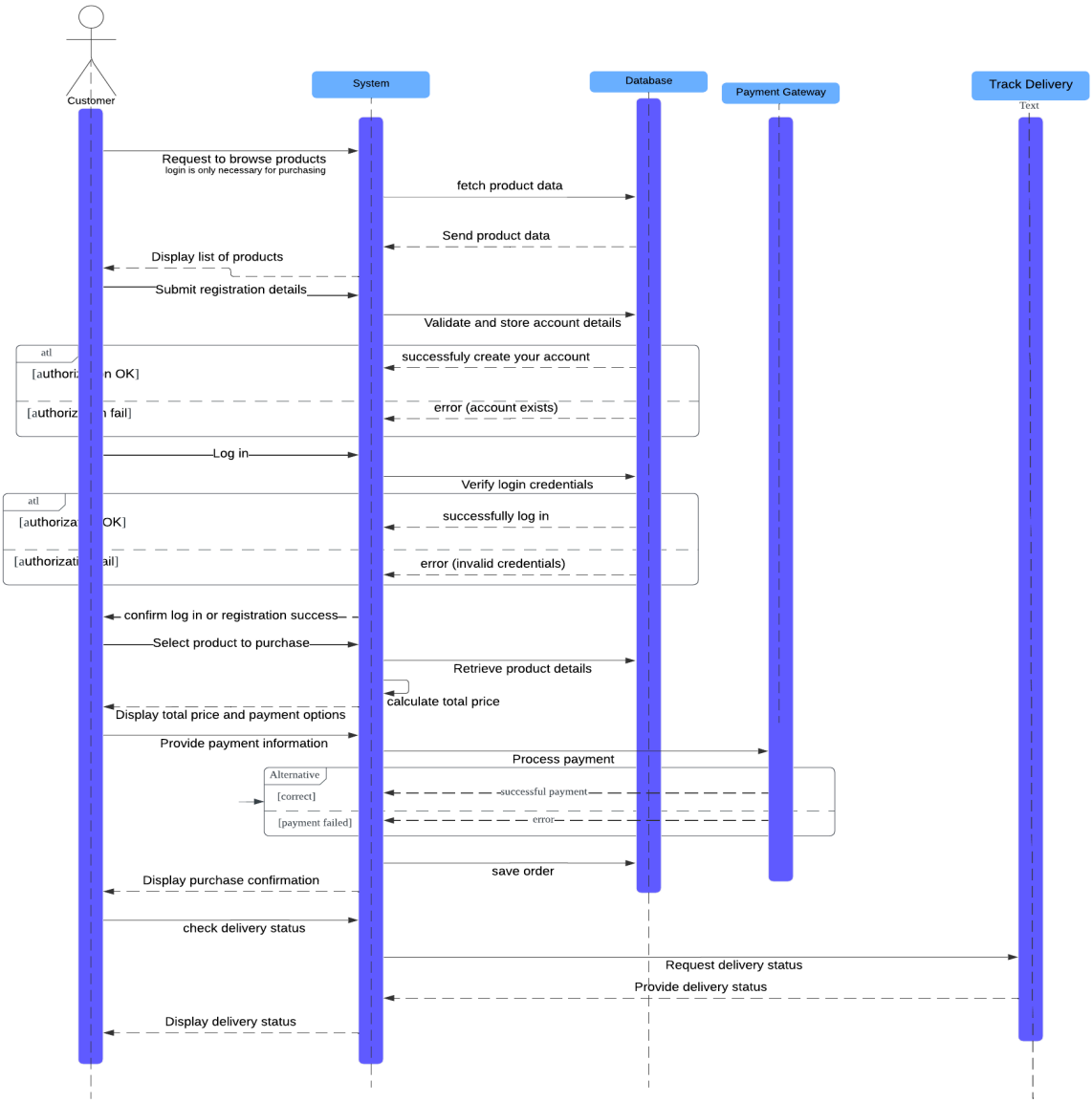


This activity diagram illustrates the system's workflow, beginning with user registration. Customers can browse products, purchase available items, or add them to their cart for

checkout. If checkout fails, they retry; if successful, they track their order until delivery. Sellers either create a store or manage existing products, process incoming orders, and fulfill them.

3. Sequence Diagram

1. Customer Sequence Diagram

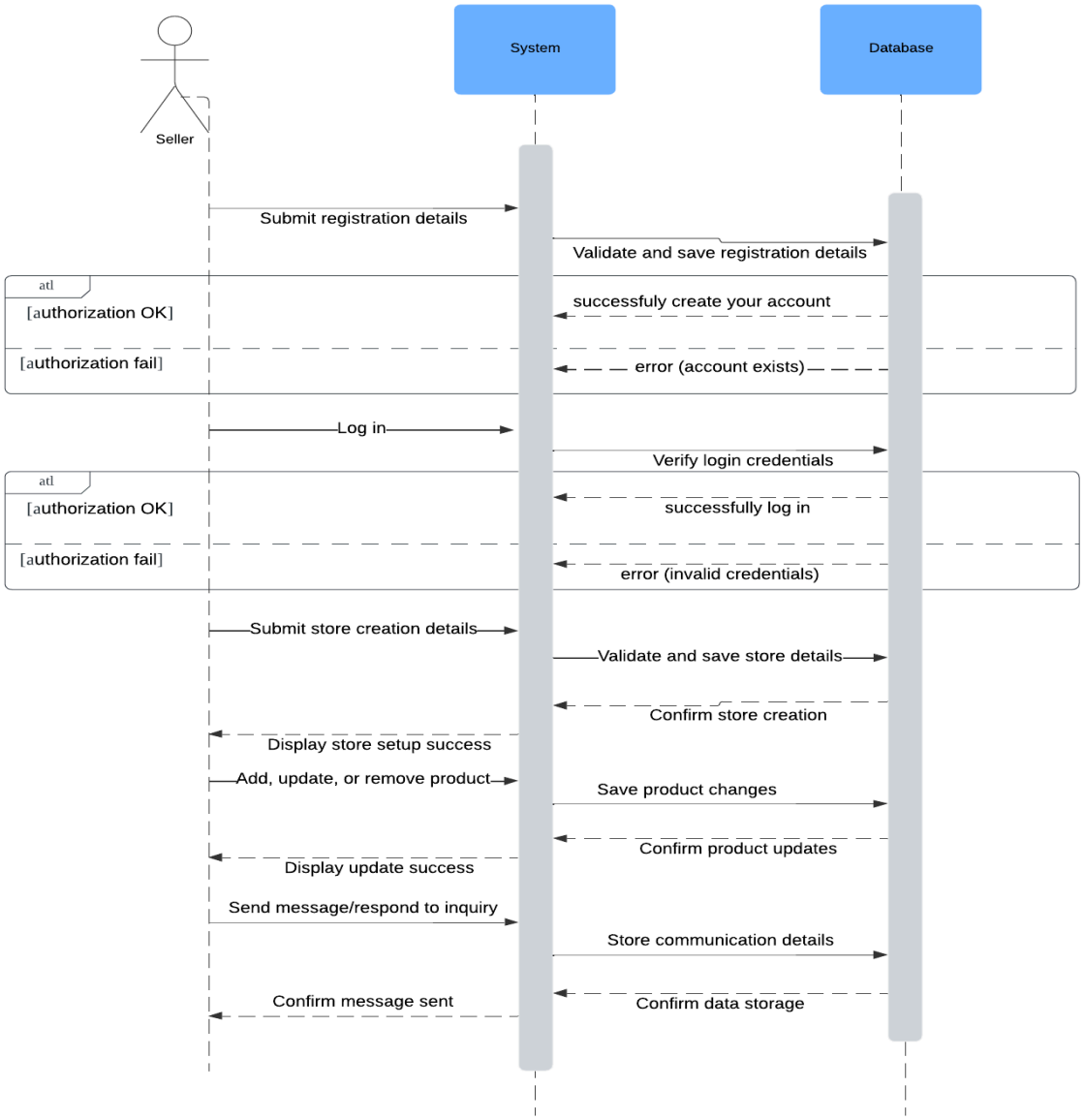


Explanation of the interactions

- **Browsing Products**
    - The Customer sends a request to the System to browse products.
    - The System fetches product data from the Database and sends it back to the Customer.
    - The Customer views the displayed product list.
- **Account Registration**
    - If it is the first time to open the website and a customer wants to make a purchase, they must register by submitting registration details to the System.
    - The System verifies the information and stores the account in the Database.
    - In case of success, it informs the System that the Customer has created the account.
    - In case of an already existing account, it will ask the customer to login into the account.
- **Login**
    - The Customer will provide the login credentials.
    - System verifies the information with the Database.
    - On successful the customer gets login else he needs to redo the task until successful
- **Purchasing Product**
    - The Customer selects a product.
    - The System retrieves product details, calculates the total price, and displays payment options.
    - The Customer enters payment details, which the System forwards to the Payment Gateway.
- If the payment is successful, the System stores the order in the Database and displays a purchase confirmation.
- If the payment fails, the Customer is informed.

- **Tracking Delivery**
    - The Customer requests the status of delivery.
    - The System fetches the status from the Database and displays it.

2. Seller Sequence Diagram



Explanations of the interactions

- **Account Registration**
  - Prior to creating a store, the Seller must perform registration or logging in by submitting the registration details, or log in credentials to the System.
  - The System checks these details from the Database.
  - In case the account is unique, the System saves the information and confirms to the Seller that he/she has been successfully registered.
  - If the account already exists, then the System asks the seller to login into his/her account.
- **Login**
  - The Seller provides the login credential.
  - The System verifies these against the Database.
  - If the verification is successful, then the system provides access to the Seller.
  - In case of invalid credentials, they are to check credentials and login again.

Store Creation

- After successful login or registration of the Seller, he/she can create a store by providing store creation details.
- The system validates these and stores them in the Database.
- On success, System confirms store setup to the Seller

Product Management

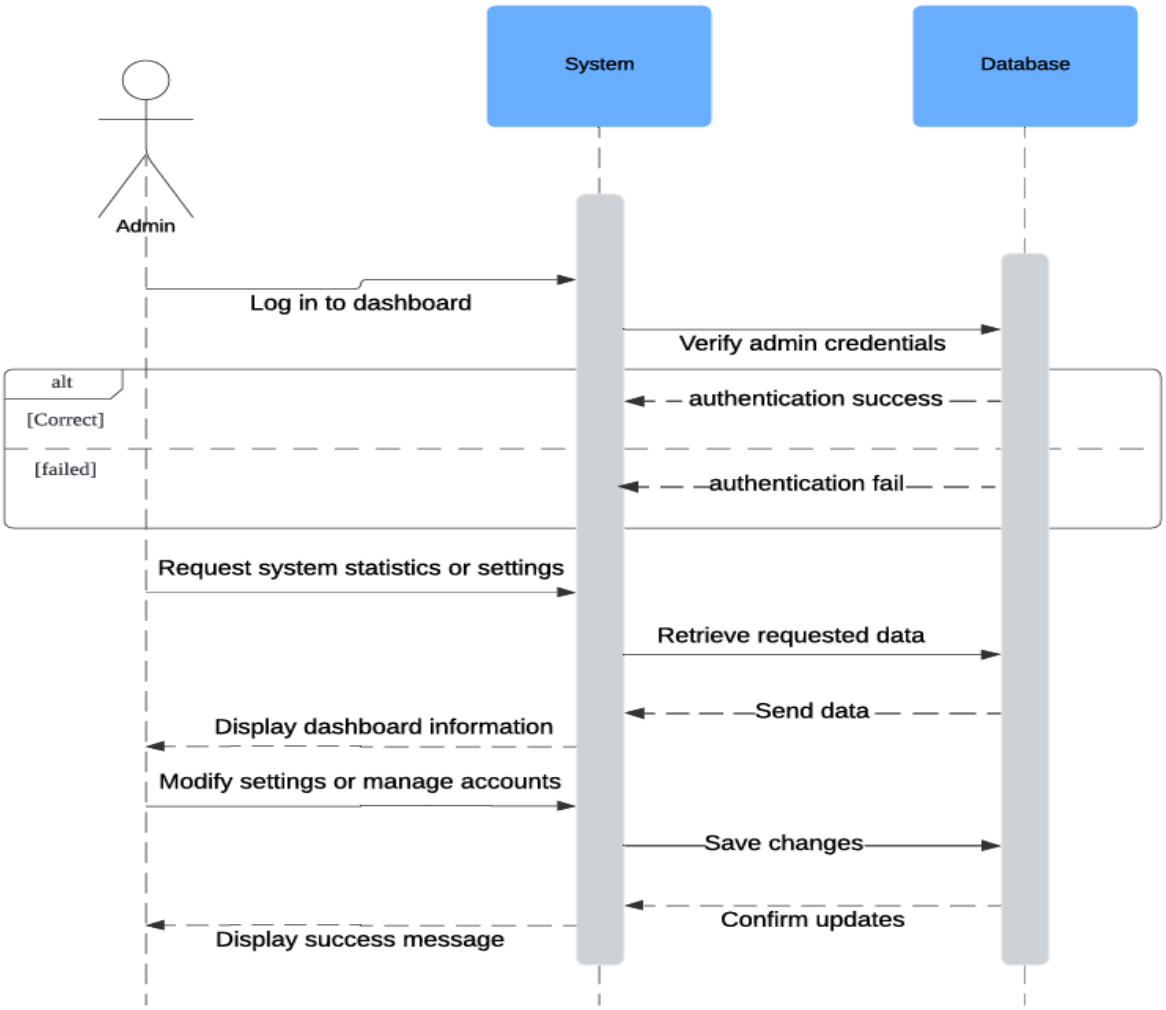
- The Seller can create, edit, or delete products.
- The System processes these changes and updates the Database.
- The Seller receives confirmation of successful product updates.

Communication

- The Seller sends messages or responds to customer inquiries.
- The System stores details of the communication with the Database.

- A confirmation is sent to the Seller that the message has been delivered.

3. Admin Sequence Diagram



Explanations of the interactions

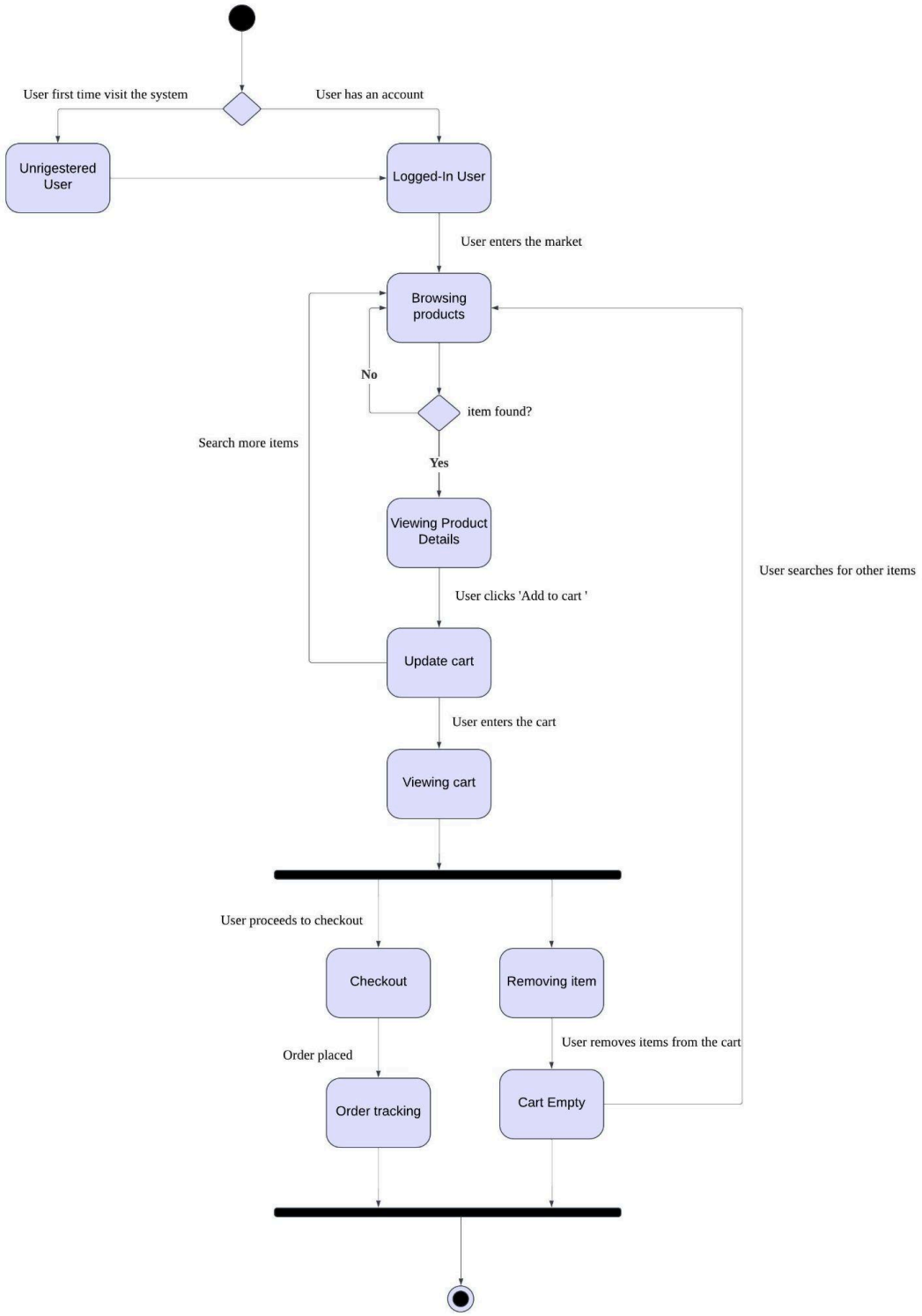
- **Login to Dashboard**
  - Admin enters the login credentials.
  - System verifies the credentials from the Database.
  - If authentication is successful, then the System allows access to the dashboard.

    - If authentication fails, then the System asks to recheck the credentials and try to log in again.

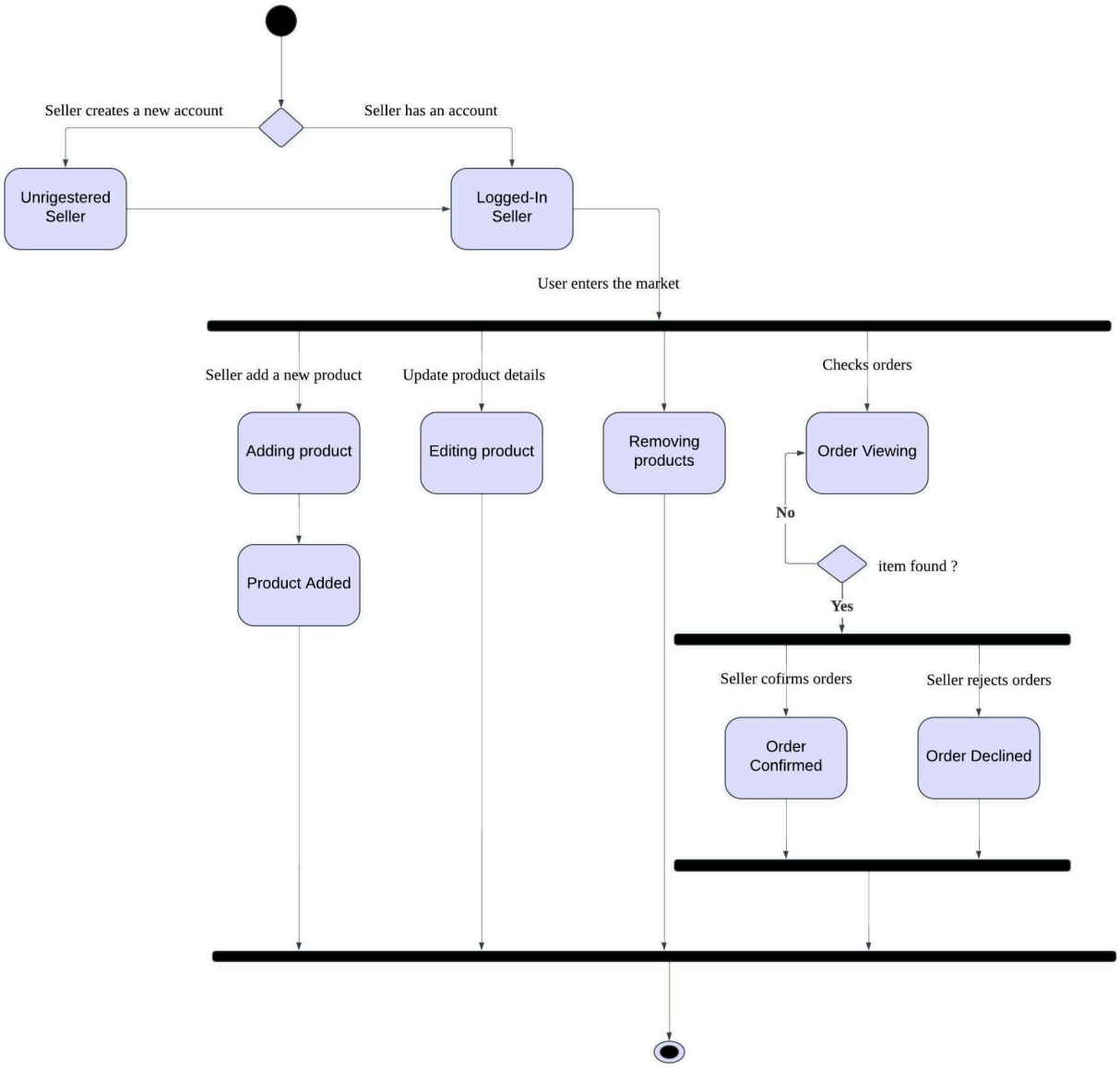
- Requesting System Statistics or Settings
    - Admin requests some statistics or system settings from the System.
    - System fetches the requested data from the Database.
    - The data then bounces back to the Admin who views it on the dashboard.
- Change Settings or Account Management
    - The Admin applies some modification in the system settings or manages the user account.
    - System stores these changes in the Database.
    - A message of success is shown that confirms updates applied.

4. State Diagram

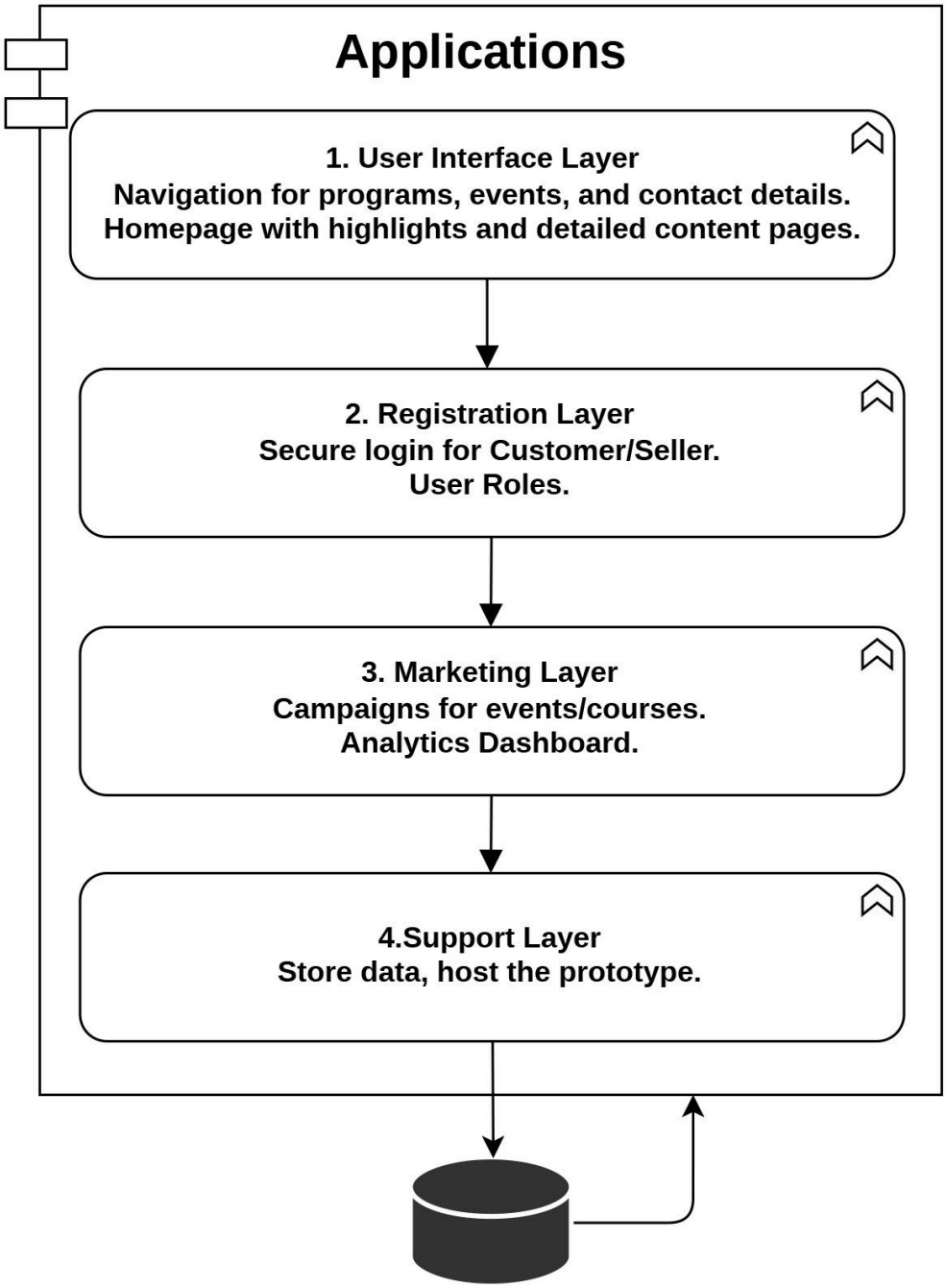
- **For Customer:**



- **For Seller:**


Seller creates a new account
Seller has an account
Unrigestered Seller
Logged-In Seller
User enters the market
Seller add a new product
Update product details
Checks orders
Adding product
Editing product
Removing products
Order Viewing
No
item found ?
Product Added
Yes
Seller cofirms orders
Seller rejects orders
Order Confirmed
Order Declined

4. Architecture Design



1. **User Interface Layer**

- The system will consist of program and event finds, for programs and events, organized by pertinent features for online shopping which include, the search bar, the product category, and the filters such as, clothes, devices. It will also have a shopping cart and a basic check out feature.
- Key sections such as Home, Shop, carts, and Contact Us will make it easy for users to navigate.
- The user interface will also be enhanced to support mobiles as well as PCs for ease of use. It will also encapsulate assistive features such as; large font and texts that describe images.

2. Registration Layer

- You can only register for the tool and utilize the login feature through your email address, and name, which have been made available in the prototype.
- Passwords will be encrypted so that the information entered will be safe and users can change their password online through an email.
- Two user roles are defined:
    - Sellers can upload and manage product listings, track inventory, and view basic analytics.
    - Customers can browse products, add items to their cart, make purchases, and track orders.

3. Marketing Layer

- The prototype will include basic email promotions, such as welcome emails and discount offers.
- A simple dashboard will show important numbers like:
    - How many people clicked on emails (click-through rates).
    - How many people bought something (conversion rates).
- The system might connect to tools like MailChimp for sending emails.

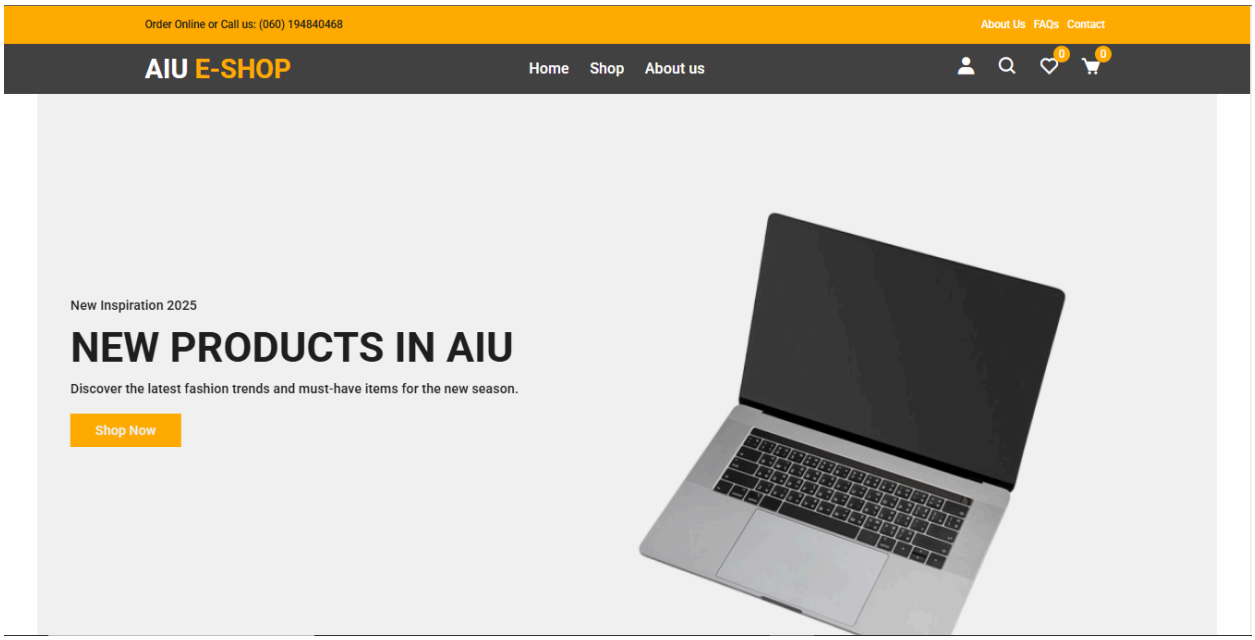
4. Support Layer

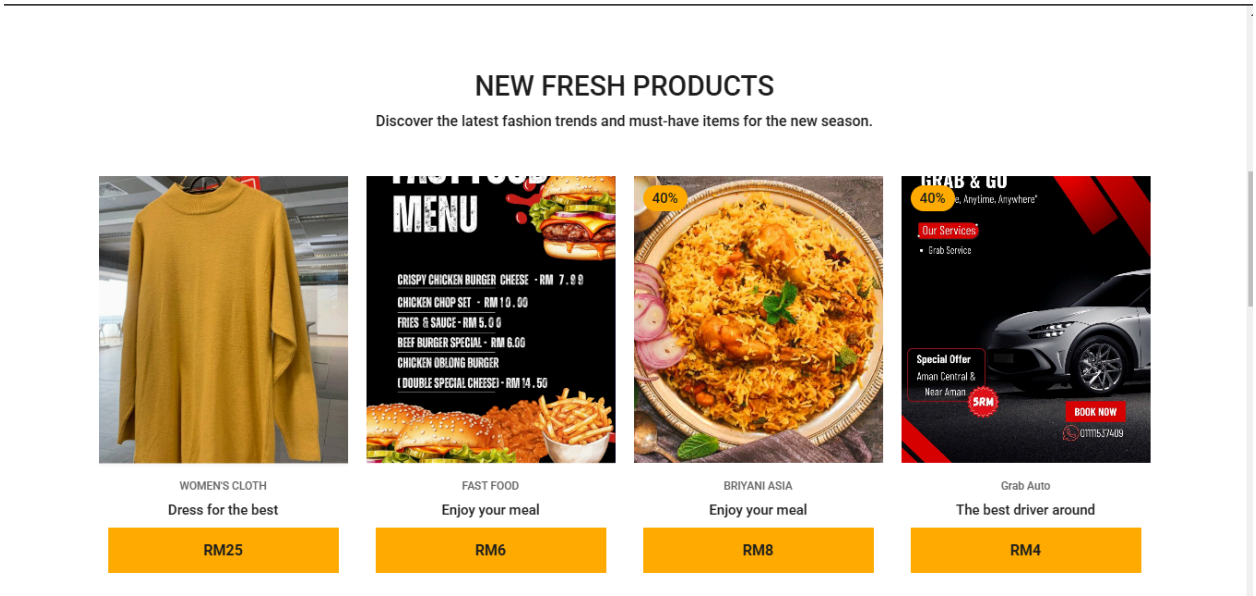
- The prototype will use basic systems for storing data:
  - Database: SQLite Alchemy for saving user and product information.
  - Hosting: AWS or Google Cloud to keep the platform online.
- Data will be protected using SSL encryption, and a backup plan will keep the information safe.

IMPLEMENTATION OF PROTOTYPE SYSTEM

Firstly, the system prototype has been implemented using various tools and programming languages. In this section, we will present the functionality of our application system prototype step by step:

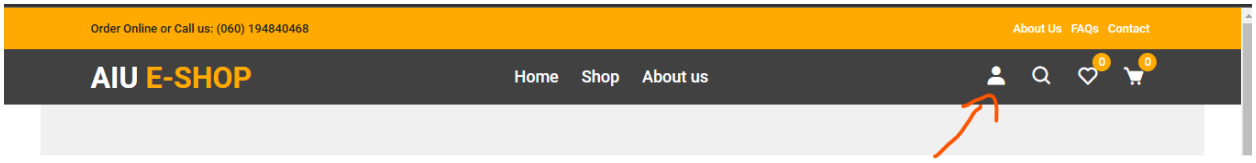
The Home Page: As with any e-commerce website, when you enter the site, you will be redirected to the home page before taking any other actions.



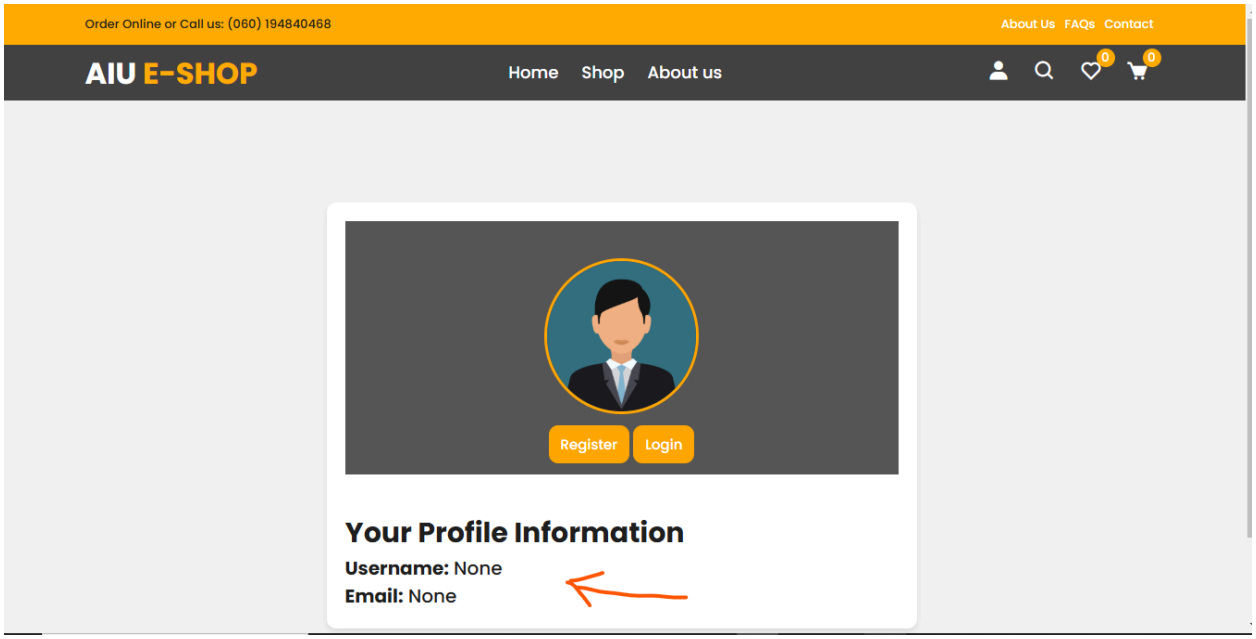


A few products will be displayed as pop-ups on the home page to give customers an idea of the products we offer.

In the top left corner of the navigation bar, you will see the user icon. Clicking on this icon will take you to the user profile.

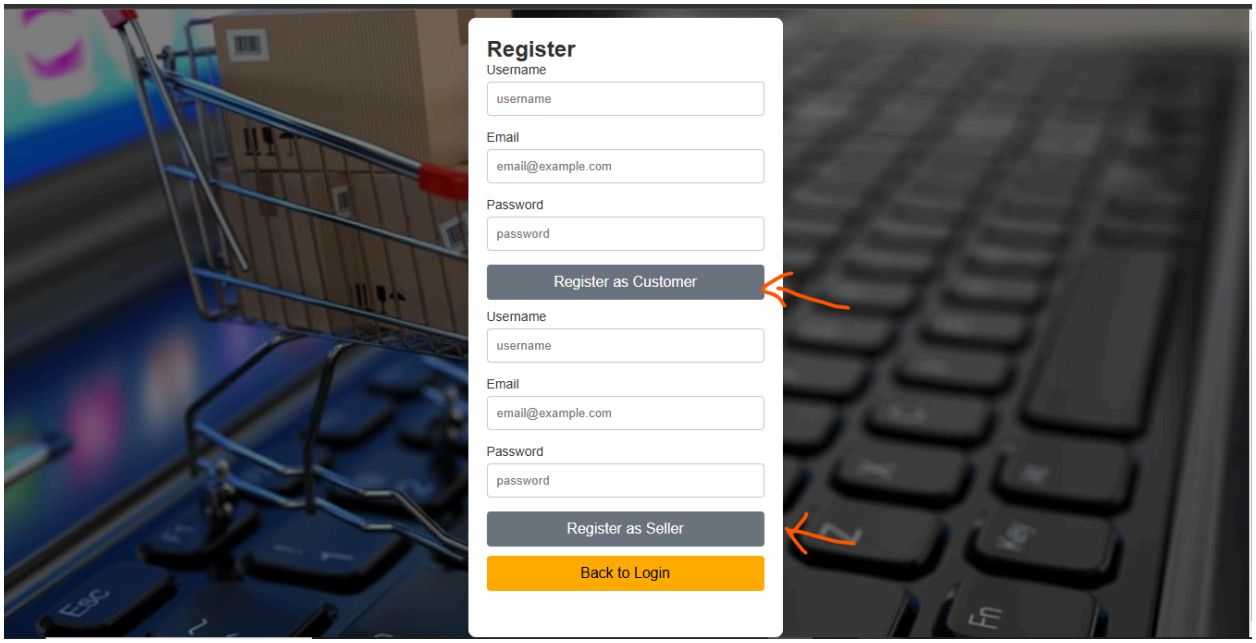


Then it will show the user profile:



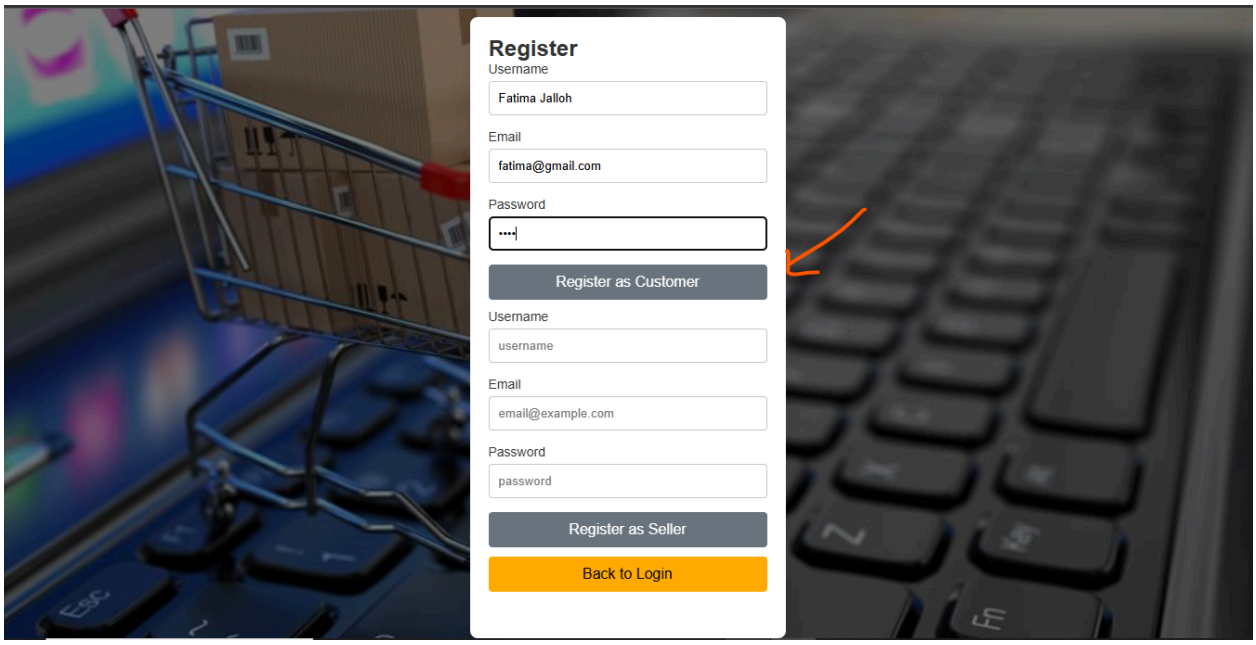
Now you can notice that there is no email and username and this is because there is no user who logged in yet. Now we have two buttons there "**Register**" and "**Login"**

Firstly let's register by click on the "**Register**" button

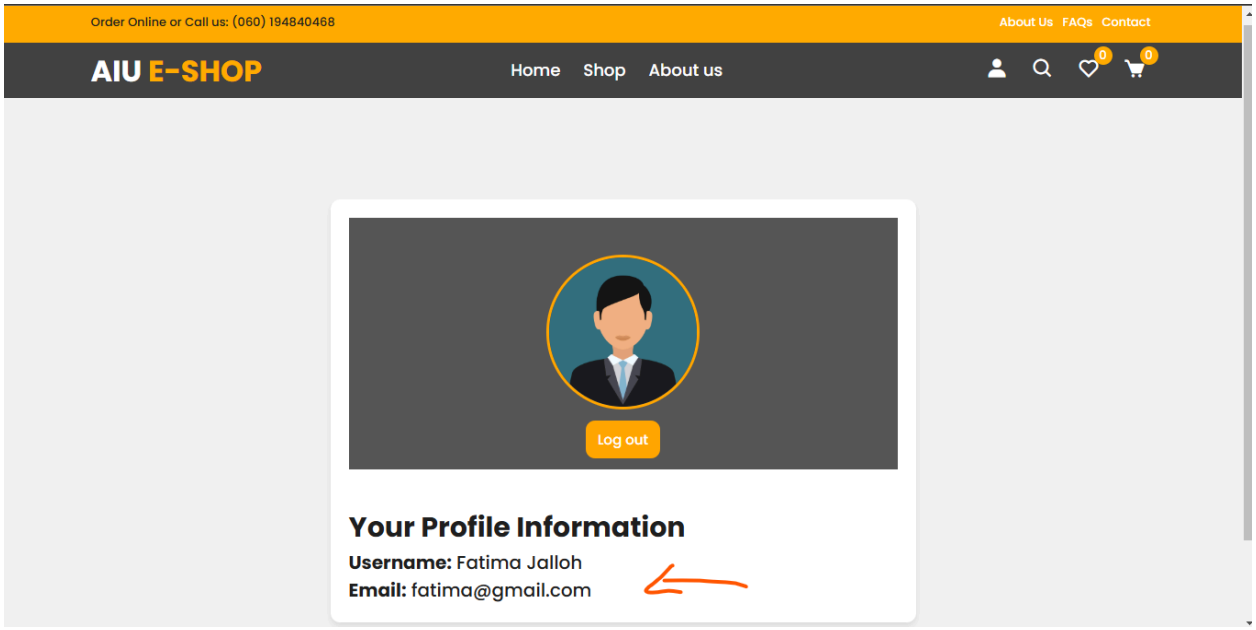


Now here we have 2 options, either we want to register as a **customer** or as a **seller**

Let's try both and see what will happen:

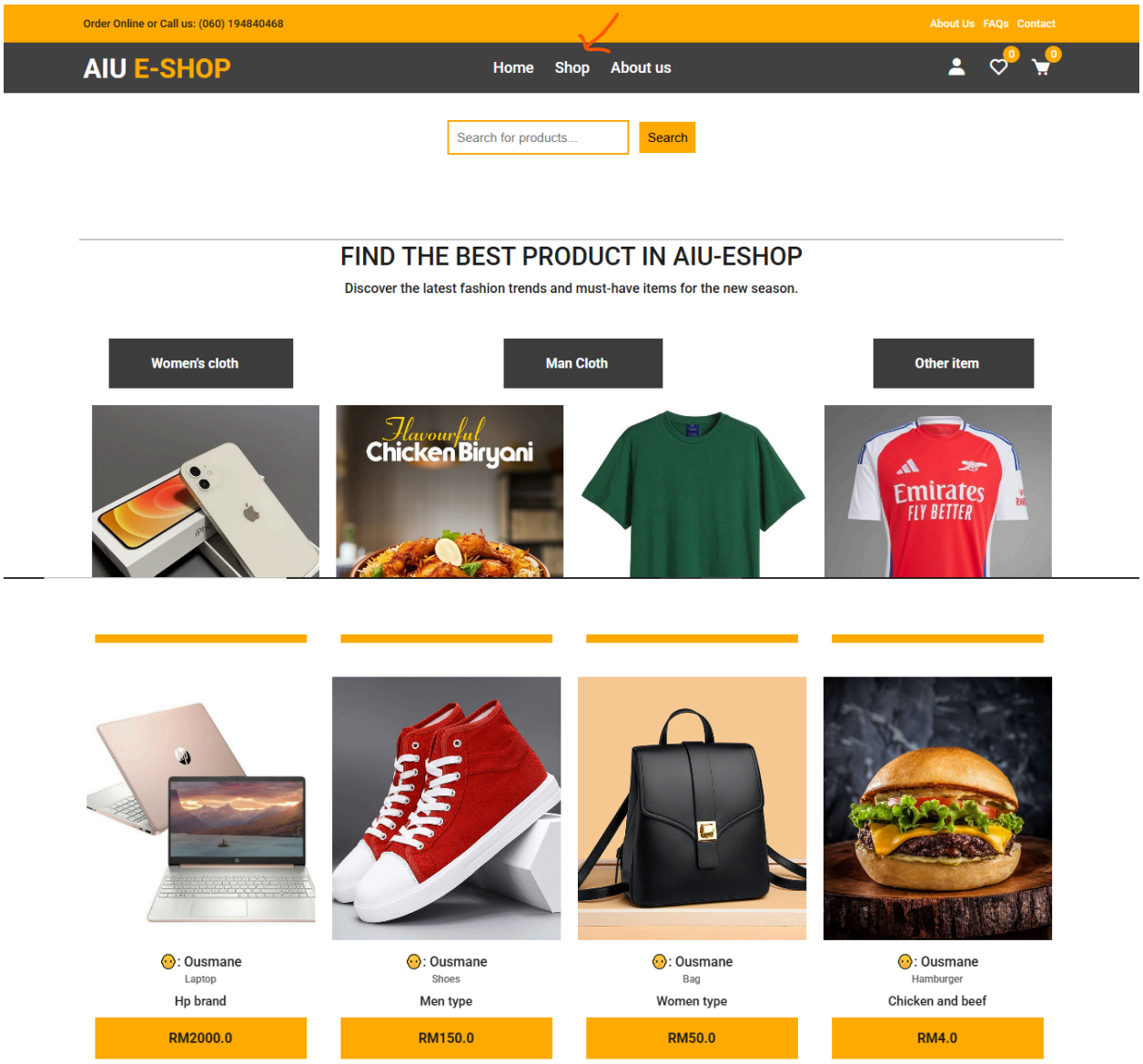


After providing the username, email and password we click on "**Register as a customer**"



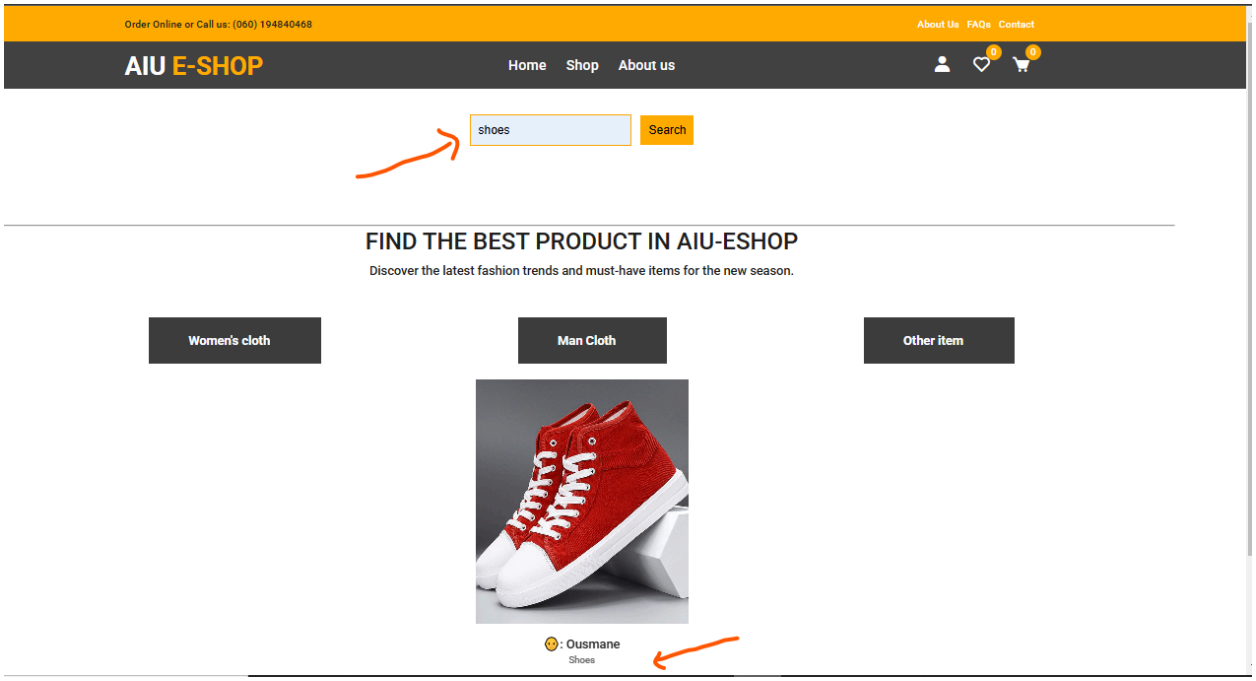
Here we go, we have our information stored in the database and we have a button “**Logout**” at the top there if we want to perform the action.

Next we can click on the link “Shop” at the top to browse for product;

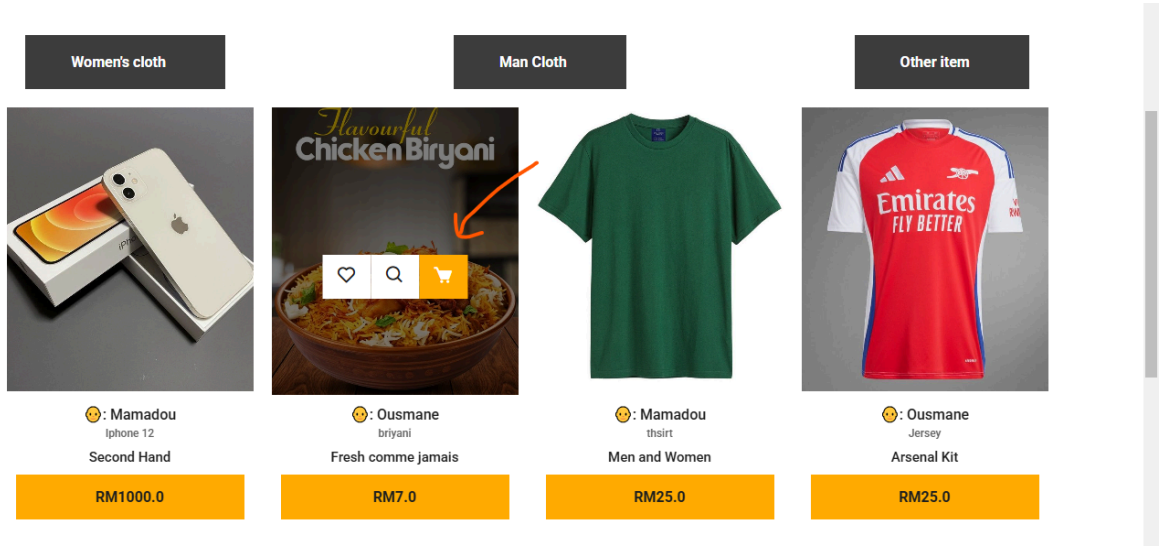


Those are some available products in the shop. In this page we can use the search bar to quickly get the product we want.

When we input “shoes” and click search it shows only shoes as it shows below

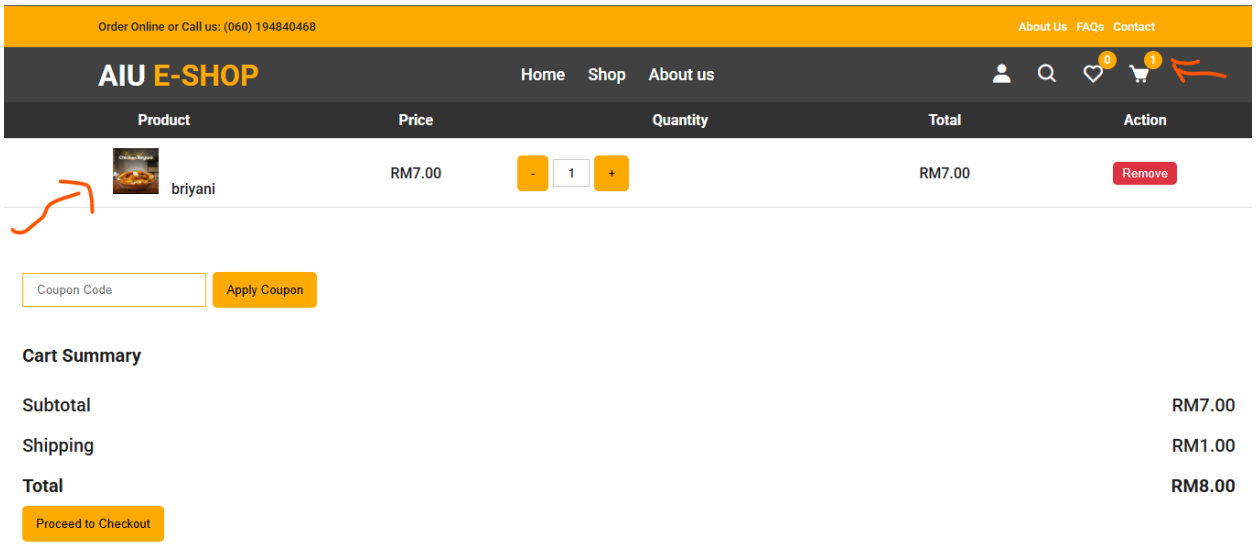


From here we can add a product to cart by click on the cart icon on every product you hover the cursor:

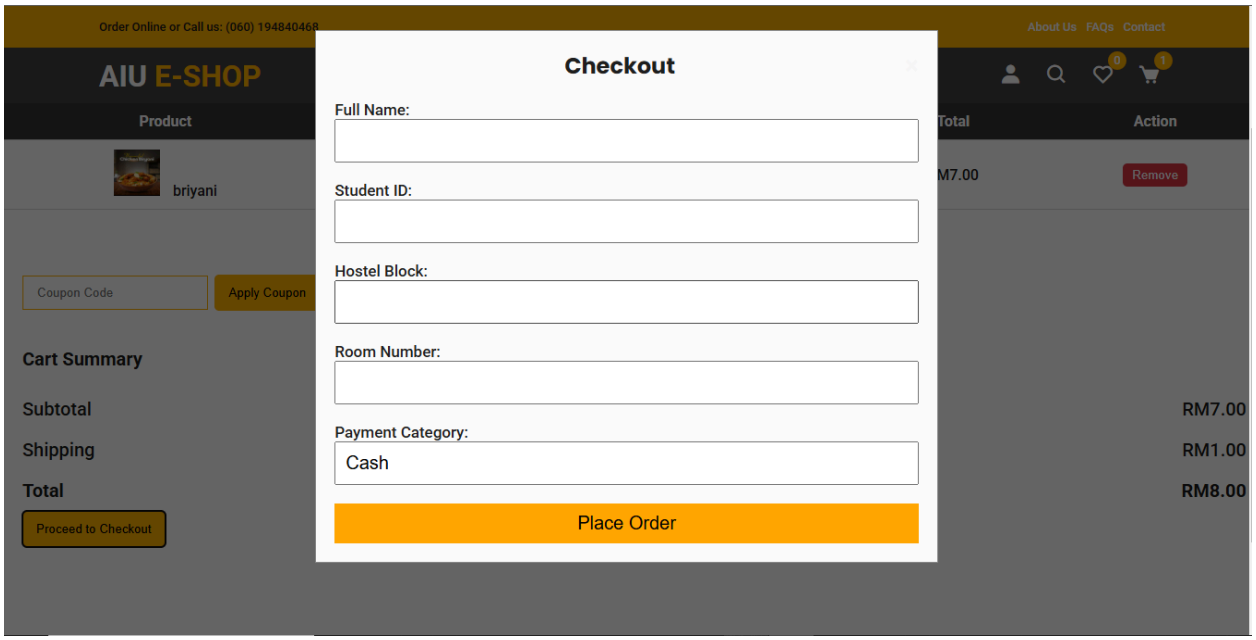


Now we can go to the cart and see the product added for further functionality.

In this cart section we see the “Biryani” is there and the cart icon is updated to 1 because we have added only one product. We have the remove option, increase the amount of items, apply coupons and proceed to check out with all the pricing including the shipping fee.

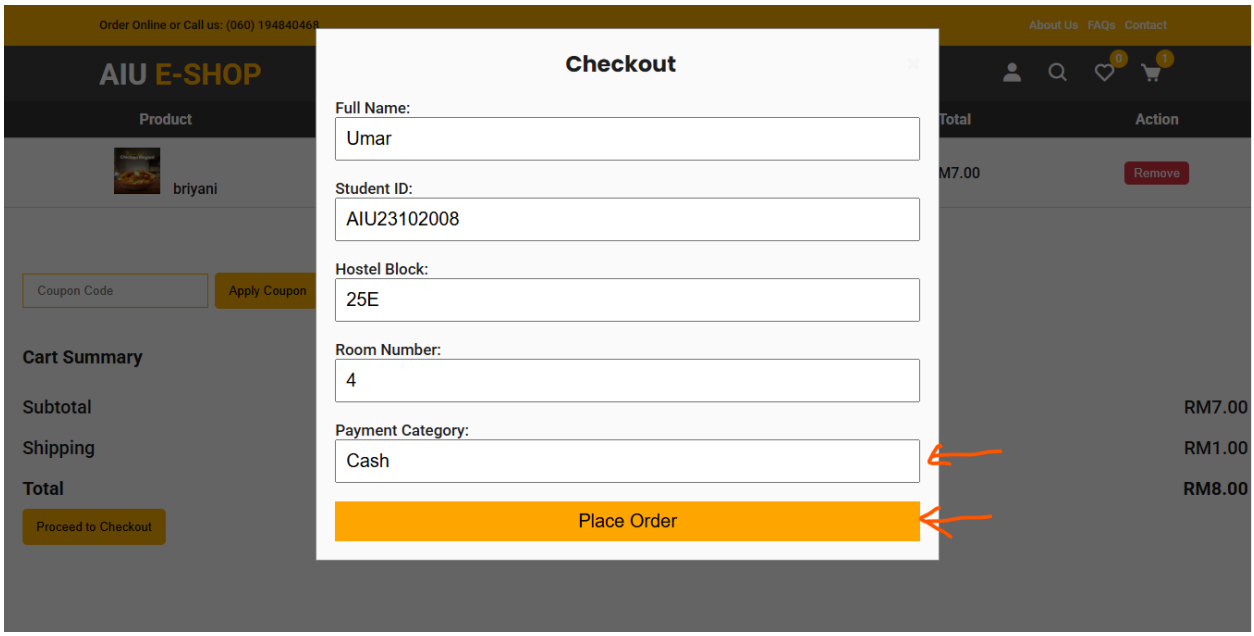


Next we can click on the **proceed to check out** at the bottom to see the next step.

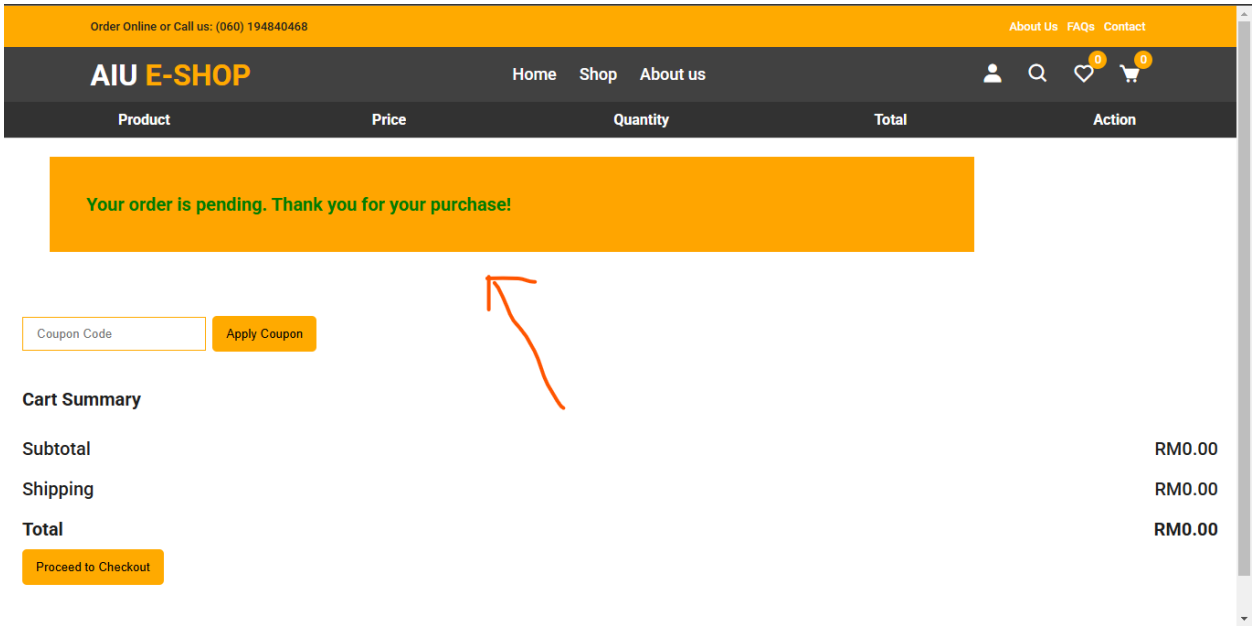


When we click on the proceed to check out, a form asking for customer information will popup, let's fill it out.

Now you can see in the image below we have input some informations and for the payment we have the default one which is **Cash**

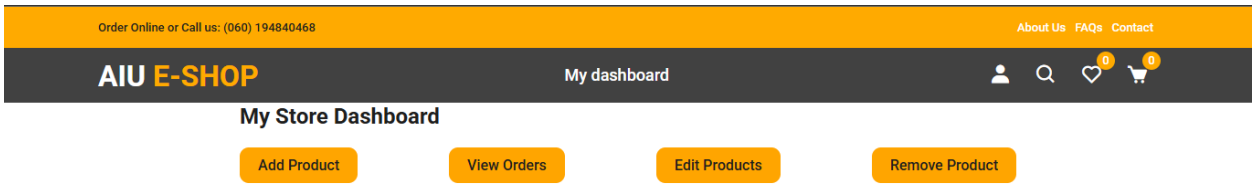


Now we can click the place order to see the next action that the system will display



Now as we can see here it displays the status of the order it is pending. It will remain like this until the seller of the product approves it.

Let's go to the admin side to see if the order has been placed. For that we need to login, in this cash i have already an admin account, otherwise i must register as a seller.

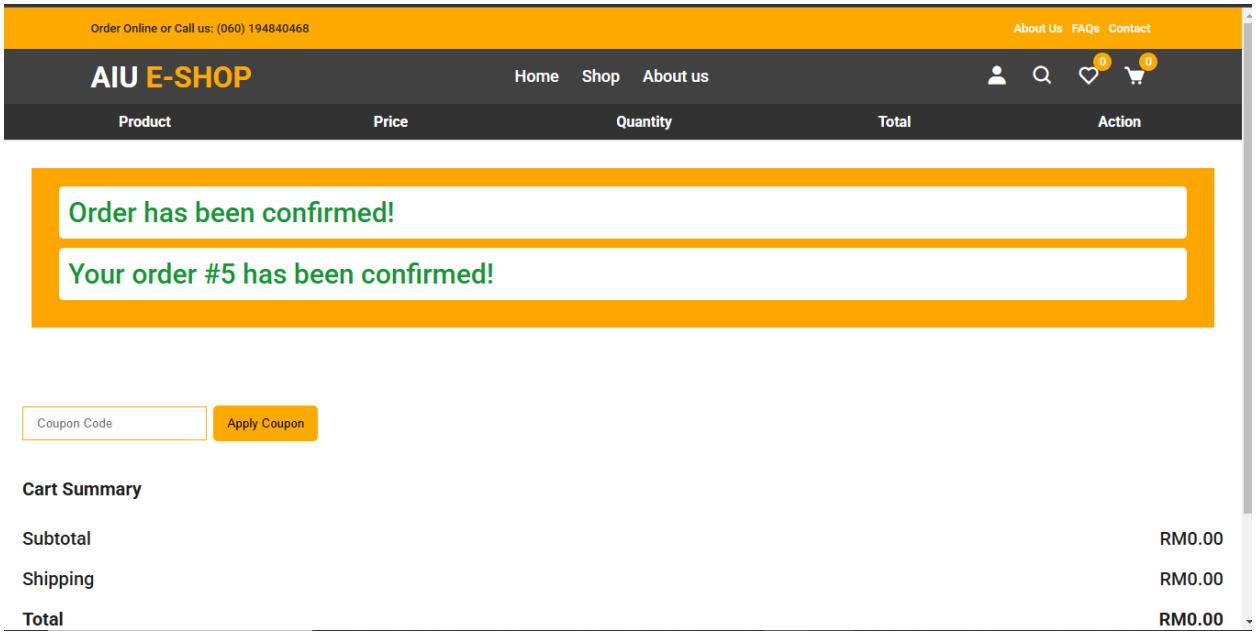


Now once i logged in as a seller, here is my interface. Let's click on the view orders

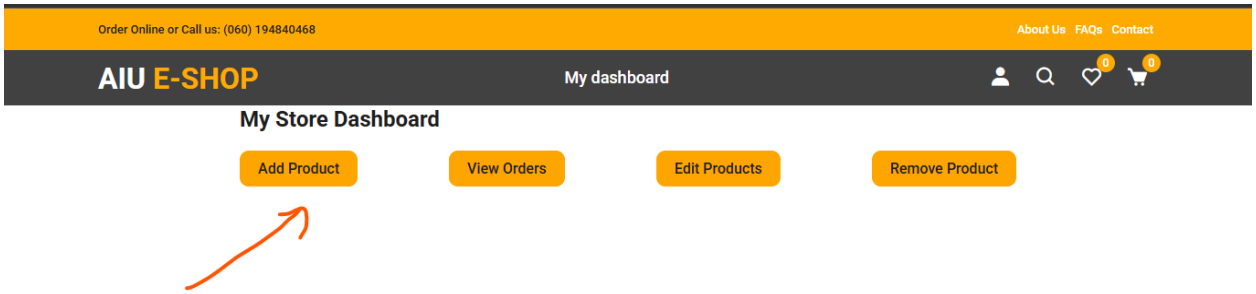
View Orders

Order ID	Full Name	Student ID	Hostel Block	Room Number	Payment Category	Total Amount	Status	Actions
2	Fatima	aiu22102200	D	3	Cash	RM7.0	confirmed	Confirm Decline
3	Maria	AIU22102802	Block D	14	Cash	RM7.0	confirmed	Confirm Decline
4	Thierno	12	A	04	Cash	RM7.0	declined	Confirm Decline
5	Umar	AIU20102008	25E	2	Cash	RM7.0	pending	Confirm Decline

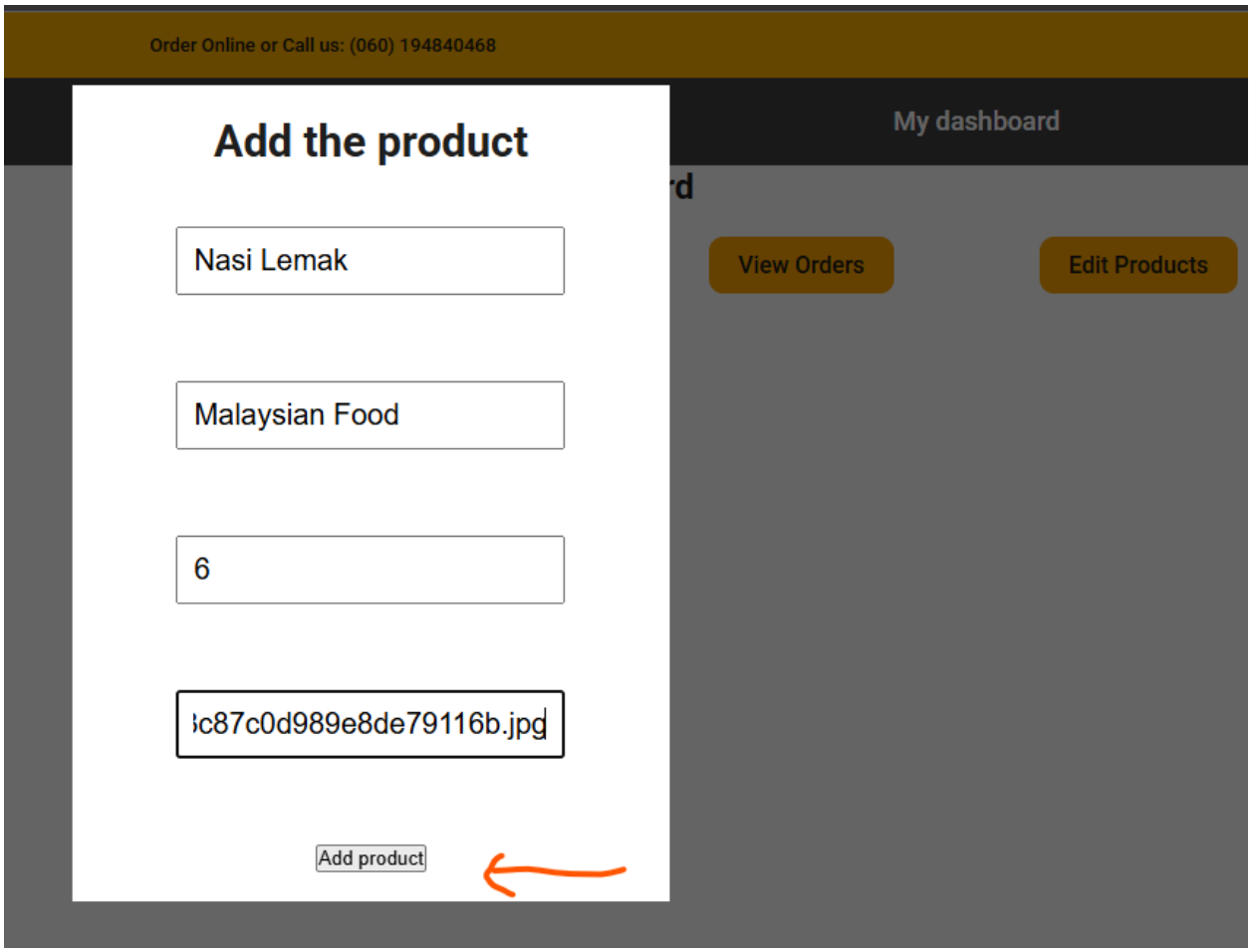
Here we go, we have all orders placed to our product, for the latest we have Umar's order that we just placed earlier and the status is pending, now the seller has 2 options either to confirm or cancel the order. Let's confirm and see what will happen:



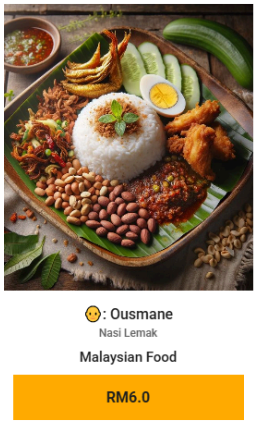
When we confirm and go back to the customer side, we can clearly see that the status has been updated from **Pending** to **Order has been confirmed.**



From the admin side again let's see what we can perform, for example the **"Add Product"**

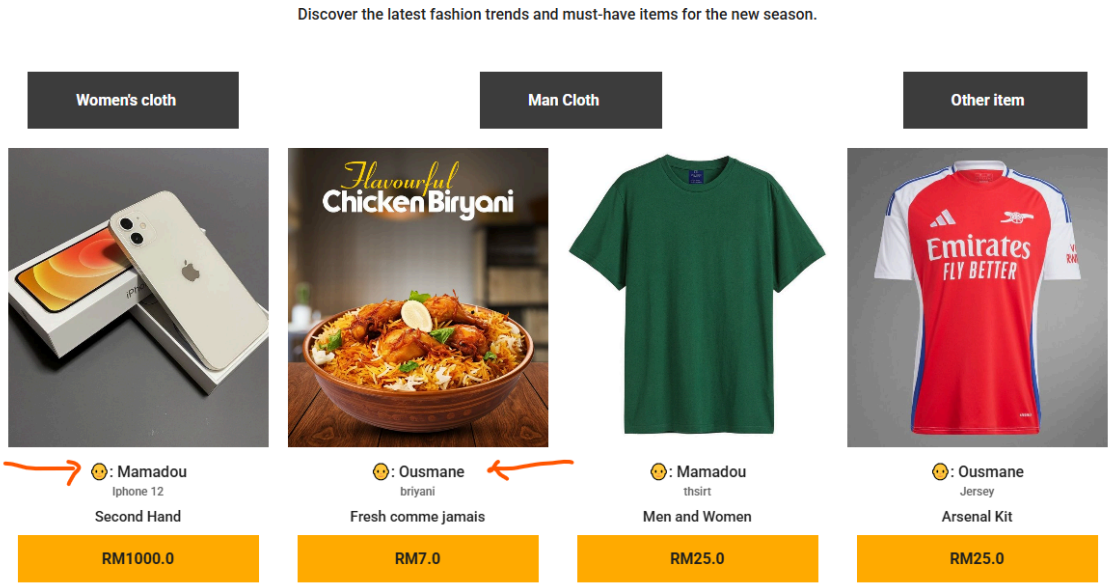


Once we click on the section "Add Product" a form will popup asking the product details. Now after filling the details, we can click on the button "Add product"to display the product.



The product is successfully added and displayed to the cart.

One important thing to mention here is the the properties of the product:



Each product displays the name of the seller, allowing customers to identify the owner of the product.

In conclusion, this section has highlighted the main features of our system prototype, covering everything from registration to the checkout process, as well as adding products and managing orders.

SYSTEM TESTING STRATEGY

The system testing strategy focuses exclusively on the functionalities available in the current prototype. The testing approach ensures that each feature performs as intended under various conditions, including normal use and edge cases.

Testing Methodology

For each functional requirement, we performed:

1. **Positive Testing**: Verifying the system behaves correctly with valid inputs.
2. **Negative Testing**: Checking how the system handles invalid or unexpected inputs.

Test Cases

1. User Registration and Login

Objective: Ensure customers and sellers can register and log in using valid credentials.

Test Case	Input Data	Expected Result	Actual Result
Register with valid email	seller1@gmail.com	Registration successful	Pass
Register with invalid email	buyer2gmail.com (missing @)	Error message: "Please include the "@" sign in the email address, as "buyer2gmail.com" lacks the "@" sign."	Pass
Register with existing email	seller1@gmail.com (already exists)	Error message: "Email already registered as seller"	Pass
Login with correct credentials	Email: seller1@gmail.com Password: 1234	Login successful	Pass

Login with incorrect password	Email: seller1@gmail.com Password: 3456	Login prevented	Pass

2. Product Browsing

Objective: Ensure customers can browse and view product details.

Test Case	Input Data	Expected Result	Actual Result
Search valid product	Laptop	Products matching "Laptop" displayed	Pass
Search for not existing product	item	No products found	Pass

3. Shopping Cart

Objective: Verify customers can add, update, and remove products in the cart.

Test Case	Input Data	Expected Result	Actual Result

Add product to cart	fan	Product added to cart	Pass
Remove product from cart	fan	Product removed from cart	Pass

4. Checkout Process

Objective: Validate that customers can successfully complete the checkout process using cash payment.

Test Case	Input Data	Expected Result	Actual Result
Proceed to checkout	Full Name: Student ID: Hostel Block: Room Number: Payment Category: cash	Order placed successfully, “Your order is pending. Thank you for your purchase!” displayed	Pass
Attempt checkout without filling a field	Not entering room number	Error message: "Please fill in this field."	Pass

5. Seller Store Management

Objective: Ensure sellers can manage their stores and products.

Test Case	Input Data	Expected Result	Actual Result
Add product	Product name: fan Product description: portable folding fan Price: 100 Image url:…….	Product added successfully	Pass
Add product without price	Product name: Phone Price:	Error message: "Price is required"	Pass
Confirm a valid order	Order ID: 6	Order confirmed successfully. “Your order #5 has been confirmed!” displayed in customer side	Pass

6. Admin Dashboard

Objective: Validate admin functionalities for monitoring and maintaining the system.

Test Case	Input Data	Expected Result	Actual Result
View all customers	Access admin dashboard	List of all registered customers displayed	Pass
View all sellers	Access admin dashboard	List of all registered sellers displayed	Pass
View all products	Access admin dashboard	List of all products displayed	Pass
View all orders	Access admin dashboard	List of all orders displayed with details	Pass

CHALLENGES AND OPPORTUNITIES

Developing a new product is not an easy task. Building a high-quality product is the ultimate goal for any organization. However, the software development process is constantly evolving, with automation becoming widespread and new tools and libraries emerging daily (Manan Thakkar, 2022).

Challenges

1. **Tool Selection**: Choosing the right tools is a critical phase in development. We faced difficulties selecting the appropriate tools to build the AIU marketplace. Developing the logic behind an e-commerce application is complex, requiring careful consideration of scalability, compatibility, and functionality. Moreover, we had to thoroughly evaluate and compare various tools before making a decision, which added to the complexity.
2. **Limited Knowledge**: As students still in the learning phase, our knowledge of developing a sophisticated software solution like this was limited. We had to invest significant time in research and consultations to understand and implement all the required features effectively.
3. **Integration of New Features**: While there are many APIs and technologies available to enhance safety and user-friendliness, most of them are paid services. For instance, we considered integrating a direct messaging feature through WhatsApp to facilitate better communication between buyers and sellers. Unfortunately, the cost constraints prevented us from implementing this feature.
4. **Error Handling and Software Testing**: Detecting and handling potential errors posed significant challenges. As beginners, we found it difficult to identify the most effective testing methods and to address errors systematically. This phase demanded extra effort and time to ensure the prototype was functional and reliable.

Despite the challenges, it is essential to highlight the opportunities and growth we experienced during this process.

Opportunities

1. **Teamwork and Project Management**: Working as a team of five members, we effectively planned, developed, and executed the project. From the initial project description to the software development phase, we operated as a cohesive unit. This experience taught us the value of collaboration, task delegation, and maintaining productivity under deadlines.

2. **Enhanced Programming Knowledge**: Developing the project from scratch significantly improved our understanding of programming logic and software architecture. We also learned how to integrate external components programmed by others, which helped us save time and added value to the project.
3. **Business Opportunities**: Once deployed, the software can be transformed into a viable business product. For example, we plan to charge a small fee or tax on each transaction between buyers and sellers. This revenue can be used to maintain the system and fund the addition of more advanced features in the future.
4. **Problem-Solving Skills**: The challenges we faced during development pushed us to develop strong problem-solving skills. Whether it was researching complex topics or finding alternative solutions for feature integration, we learned how to approach obstacles creatively and effectively.

ROLES AND RESPONSIBILITIES OF PROJECT MEMBERS

The group members collaboratively contributed to different sections of the project report.

This ensured that all aspects of the project were thoroughly addressed and completed efficiently.

Report Section	Assigned Member(s)	Responsibilities
Introduction	Hanaa, Thierno	- Provided an overview of the project and outlined its objectives.

System Requirements Analysis	Menia, Habiba	- Documented the functional and non-functional requirements based on user needs, and also highlighted the software process model adopted.
System Model	Menia, Hanaa	- Reported on the system workflows and data structure, using: use case, sequence, and state diagrams.
System Architecture Design	Osama	- Reported on the architecture of the system, explaining how components interact and data flows.
Implementation	Thierno	- Documented the process of developing the system, explaining how features like user registration, product browsing, and order placement were implemented.
Testing Strategy	Habiba	- Tested and reported on the test cases and outcomes to validate system functionalities.
Challenges and Opportunities	Thierno	- Reported on challenges faced during development and opportunities for improvement.
Roles and Responsibilitis	Habiba	- Documented the roles and contributions of each group member.

Conclusion	Menia	- Summarized project outcomes.
Report Compilation	All	- Collaboratively compiled, formatted, and finalized the report for submission.

The completion of this project was achieved through collaboration. Each member contributed to their respective roles, ensuring that all aspects of the system were thoroughly documented. Throughout the development process, we've reviewed and refined the code to align with the project requirements. Regular discussions and meetings facilitated a structured workflow, allowing for an organized and cohesive final report.

CONCLUSION

In conclusion, this project addresses the problems faced by student entrepreneurs at AIUthrough an online marketplace that will support small enterprises on campus. The platform is one of the means toward achieving SDG 9 by fostering innovation and improving infrastructure. This is an effective, user-friendly solution for both buyers and sellers in enabling local commerce to thrive while helping students manage and promote their businesses more effectively. Moreover, while the development of this prototype has lots of challenges, it offered great opportunities which were really valuable in many aspects. This developed our technical skills, teamwork ability, knowledge of software processes, and so on. This project is one of the major milestones in our journey to becoming proficient software developers.

REFERENCES

1. GeeksforGeeks. (2024, March 15). *Incremental Process Model Software Engineering*. GeeksforGeeks. https://www.geeksforgeeks.org/software-engineering-incremental-process-model/?utm_source=chatgpt.com
2. *Reuse-Oriented model*. (2024, April 25). blog.codingminutes.com. Retrieved January 16, 2025, from https://blog.codingminutes.com/reuse-oriented-model
3. *Sustainable development goals*. (n.d.). European Commission. https://commission.europa.eu/strategy-and-policy/sustainable-development-goals_en
4. Synoptek. (2025, January 9). *10 Software development challenges Every developer faces | Synoptek*. https://synoptek.com/insights/it-blogs/10-challenges-every-software-product-developer-faces/

APPENDIX

Source code:

https://drive.google.com/drive/folders/1NW9zTsnxRJNTORA5gDsEPpvXPxuMowS8?usp=sharing

Similarity Index:

CCC2143 - Group Project Assignment - Google Docs.pdf

ORIGINALITY REPORT

6%	3%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	2%
2	Submitted to Albukhary International University Student Paper	1%
3	Submitted to Massey University Student Paper	1%
4	Submitted to Colorado Technical University Student Paper	1%
5	Submitted to Arab Open University Student Paper	<1%
6	Submitted to De Montfort University Student Paper	<1%
7	Submitted to University of West London Student Paper	<1%
8	proceeding.dharmawangsa.ac.id Internet Source	<1%