SHOPMAX: A WEB-BASED E-COMMERCE MARKETPLACE FOR THE UGANDA CHRISTIAN UNIVERSITY COMMUNITY CASE STUDY: UGANDA CHRISTIAN UNIVERSITY

EMMA MARIAL DOMKOC

S23B13/020

BRUNO NASASIRA

S23B13/044

MOREEN MUKISA

M23B13/099

JOSELINE AKURUT

M23B13/008

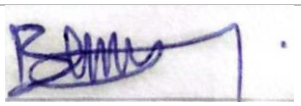
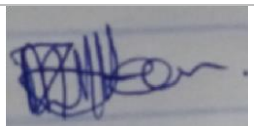
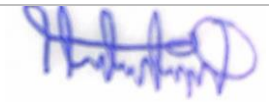
A PROJECT REPORT SUBMITTED TO THE FACULTY OF ENGINEERING, DESIGN AND TECHNOLOGY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF A DEGREE OF BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY OF UGANDA CHRISTIAN UNIVERSITY

July, 2026

UGANDA CHRISTIAN UNIVERSITY
A Centre of Excellence in the Heart of Africa

DECLARATION

We, Emma Marial Domkoc, Nasasira Bruno, Mukisa Moreen, and Joseline Akurut, hereby declare that this project report is our original work and has not been submitted for any other degree award to any other university before. All sources of information have been duly acknowledged through citations and references.

Emma Marial	Signature: 
S23B13/020	Date: 16th April 2026
Nasasira Bruno	Signature: 
S23B13/044	Date: 16th April 2026
Mukisa Moreen	Signature: 
M23B13/099	Date: 16th April 2026
Joseline Akurut	Signature: 
M23B13/008	Date: 16th April 2026

SUPERVISOR'S APPROVAL

We certify that this project report is the original work of the above-named students, conducted under our supervision, and is ready for submission for examination.

Ms. Mukalere Justine (Supervisor)

Signature: _____

Date: 05/25/2026

Dr. Bitalo Daphne (Supervisor)

Signature: _____

Date: 30/04/2026

Mr. Opio Solomon (Supervisor)

Signature: _____

Date: 30/04/2026

DEDICATION

To our families for their unwavering support and encouragement throughout this academic journey

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our supervisors, Ms. Mukalere Justine, Dr. Bitalo Daphne, and Mr. Opio Solomon, for their continuous guidance, constructive criticism, and valuable feedback throughout this project. Their combined expertise was instrumental in shaping this work.

Special thanks go to the Uganda Christian University community, particularly the students, staff, and local vendors who participated in interviews, testing, and the final evaluation. Their cooperation and insights were invaluable. We also extend our appreciation to the Department of Information Technology for providing the necessary resources, lab facilities, and an enabling environment for both research and development.

We are equally grateful to our fellow students who willingly tested the system during the user acceptance phase and provided honest, constructive feedback that helped refine the final product. Their willingness to engage with an early-stage prototype was truly encouraging.

We also appreciate our colleagues and friends who provided encouragement and assistance during the development of this project. Their moral support kept us motivated through challenging times. Special mention goes to those who helped debug frontend issues and offered advice on database optimization.

Above all, we thank God for the strength, wisdom, and good health to complete this work successfully. We also acknowledge the patience and understanding of our families, whose unwavering support made the late nights and tight deadlines bearable.

Contents

List of Figures

List of Tables

LIST OF ABBREVIATIONS

API	**Application Programming Interface**
CSS	Cascading Style Sheets
e-commerce	Electronic Commerce
ERD	Entity Relationship Diagram
HTML	Hypertext Markup Language
SDLC	System Development Life Cycle
SQL	Structured Query Language
SQLite	Structured Query Language Lite
UCU	Uganda Christian University
UCUC	Uganda Christian University Community
UI	User Interface
UAT	User Acceptance Testing

ABSTRACT

This project presents the design, development, and evaluation of ShopMax, a web-based e-commerce marketplace tailored for the Uganda Christian University community. The system addresses inefficiencies associated with informal trading methods such as social media platforms, which lack organization, reliability, and centralized management.

Data was collected through observation, interviews, and questionnaires involving students, staff, and local vendors. The system was developed using modern web technologies (HTML, CSS, JavaScript, Flask, SQLite) following the System Development Life Cycle (SDLC). Key features include secure user authentication (UCU email + Google OAuth), product listing and management, search and filtering, and an in-platform messaging system.

System testing – unit, integration, and user acceptance testing – confirmed the reliability and usability of the platform. User acceptance testing (n=18) gave average ratings above 4.5/5 for ease of use, search efficiency, and overall satisfaction. The results show that ShopMax significantly enhances accessibility, promotes student entrepreneurship, and improves transaction efficiency within the university environment. The system provides a scalable foundation for future enhancements such as payment integration and mobile application support.

CHAPTER ONE. INTRODUCTION

1.1 Background of the Study

The advancement of information and communication technology has significantly transformed global business operations. One of the most significant developments in modern commerce is electronic commerce (e-commerce), which enables individuals and businesses to conduct transactions over the internet. E-commerce platforms facilitate digital marketplaces where products and services can be displayed, searched, and purchased conveniently.

Globally, e-commerce adoption has accelerated rapidly over the past decade. Industry reports indicate that global online retail sales exceeded 6 trillion USD in 2024, highlighting the growing importance of digital marketplaces. Many businesses now
depend on online platforms to expand their customer base and improve product accessibility.

Universities represent environments where active economic interactions occur daily. Students frequently operate small businesses selling clothing, electronics, food items, and academic materials. Similarly, staff members and nearby communities participate in buying and selling activities within the university ecosystem. However, these transactions typically occur through informal communication channels.

At Uganda Christian University, many students and staff rely on social media platforms such as WhatsApp, Instagram, TikTok, and Facebook to advertise products and services. While these platforms enable communication, they are not specifically designed to function as structured marketplaces. Consequently, product information remains scattered, transactions are inefficient, and users struggle to identify reliable sellers.

This fragmented environment necessitates a centralized digital platform to organize marketplace activities within the university community. This project presents ShopMax, a web-based e-commerce marketplace designed specifically for the Uganda Christian University community.

1.2 Problem Statement

Within the Uganda Christian University community, numerous small businesses and shops operated by students, staff, and surrounding community members are physically scattered across different locations both inside and outside the campus. This decentralization creates significant challenges.

Students and other buyers face considerable difficulty when searching for specific products. In many cases, a student must move from one shop to another across different campus areas and surrounding trading centres while searching for a particular item. This process is time-consuming and inconvenient, particularly when the buyer is uncertain about product availability. Additionally, shops may be closed during certain hours or on specific days, resulting in wasted time and frustration.

Current attempts to facilitate buying and selling activities through informal digital platforms such as WhatsApp groups, Instagram, TikTok, and Facebook remain unstructured and decentralized. Information about available products is scattered across multiple platforms, making it difficult for buyers to easily discover available products within the university community. Sellers struggle to reach potential customers effectively.

Therefore, a centralized digital marketplace is necessary. The ShopMax web-based marketplace system addresses this problem by providing a single platform where users can easily view available products, identify sellers, and conduct transactions without physically moving between multiple scattered shops.

1.3 Aims and Objectives

1.3.1 General Objective

The general objective of this study is to design, implement, test, and evaluate a centralized web-based marketplace system, ShopMax, that facilitates efficient buying and selling activities within the Uganda Christian University community
(UCU premises, Bugujju, Wandegeya, Nabuta and Mukono Mall area).

1.3.2 Specific Objectives

1. To design and implement a secure user registration and authentication system for students, staff, and community users.

2. To develop a product listing and management platform that allows sellers to upload, edit, and categorise products efficiently.

3. To implement effective product browsing and search functionality, including keyword search, category filters, and price range options.

4. To establish communication channels between buyers and sellers, enabling direct inquiries and negotiations within the platform.

5. To provide a centralized and organised digital marketplace that integrates all features into a single, accessible, and user-friendly platform for the UCU community.

1.4 Project Scope

The scope of this project focuses on the design, development, testing, and evaluation of a web-based e-commerce marketplace system, ShopMax, specifically tailored for the Uganda Christian University community.

Inclusions (Successfully Implemented):

User registration and login system for students, staff, and nearby community members

Product listing and management platform for sellers (upload, update, delete products)

Product browsing and search functionality with keywords, categories, and filters

Seller profile management dashboard

Buyer–seller communication channels (in-platform messaging)

Cash-on-delivery transaction coordination

Exclusions (Identified for Future Development):

Integration with live online bank or mobile money payment agents

Native mobile application (web-based only with responsive design) Real-time GPS-based delivery tracking

1.5 Significance of the System

The ShopMax system provides significant benefits to the university community. First, the system provides a centralized digital marketplace where buyers and sellers can interact efficiently. Second, the system promotes student entrepreneurship by providing a dedicated platform for product showcasing. Third, the platform improves accessibility to goods and services through organised listings and search features. Fourth, the system enhances trust and reliability in campus commerce through user verification. Finally, ShopMax contributes to the digital transformation of campus-based commerce.

CHAPTER TWO. LITERATURE REVIEW

2.1 Concept of E-Commerce Systems

E-commerce refers to the buying and selling of goods and services using electronic systems, primarily through the internet. According to Laudon and Traver (2021), e-commerce involves digital technologies that facilitate commercial transactions between individuals, businesses, and organizations. E-commerce systems provide platforms where sellers display products while buyers search, compare, and purchase items online. Over time, e-commerce has evolved from simple online catalogues to sophisticated platforms that integrate artificial intelligence, Realtime inventory management, and personalized recommendations.

The ShopMax system adopts a hybrid business-to-consumer (B2C) and consumer to-consumer (C2C) model, allowing both students (as individuals) and small campus businesses (as informal merchants) to participate. Unlike global giants, ShopMax focuses on a local, trust-based ecosystem within a university setting.

2.2 Online Marketplace Platforms

Online marketplaces act as digital intermediaries that connect multiple buyers and sellers. Turban et al. (2020) classify marketplaces into horizontal (general merchandise, e.g., Amazon, Jumia) and vertical (specialized, e.g., Etsy for crafts). ShopMax is a vertical marketplace tailored to the unique needs of a university community.

Key features of successful marketplaces include:

Product categorization and search – enabling users to quickly find relevant items.

Reputation systems – ratings and reviews that build trust.

Integrated communication – reducing the need for external messaging apps.

Secure payments – though not yet implemented in ShopMax, it is a future goal. ShopMax incorporates all but the payment feature, focusing first on establishing a reliable local trading environment.

2.3 Digital Platforms for Student Entrepreneurship

Student entrepreneurship is widely recognized as a driver of innovation and economic growth. According to Drucker (2018), entrepreneurship is “the practice of continuously creating new value through opportunity recognition.” However, student entrepreneurs face unique barriers:

Limited capital – they cannot afford expensive marketing.

Small target audience – campus scope makes mass advertising inefficient.

Trust deficit – buyers are wary of unknown sellers on social media.

Digital platforms like ShopMax remove these barriers by providing a structured, low-cost, and trusted environment. Research by Ofori and Kwame (2022) in the African context found that university-based online marketplaces increase student business survival rates by up to 40% in the first year.

2.4 Role of Information Technology in Modern Commerce

Information technology (IT) has revolutionized commerce by enabling 24/7 availability, geographic reach, and data-driven decision making. Stair and Reynolds (2019) emphasize that IT systems are now core to operations, not just support functions. In e-commerce, IT contributes:

Automated inventory management – real-time updates prevents overselling. Customer relationship management (CRM) – tracking user behavior to improve service.

Omnichannel integration – seamless experience across web and mobile.

ShopMax uses IT to solve a specific campus problem: fragmented trading. By centralizing listings, automating product status, and providing a searchable database, the system mirrors professional e-commerce practices at a scale appropriate for a university.

2.5 Challenges of Existing Informal Marketplace Systems

Current practices at UCU rely heavily on WhatsApp groups, Instagram stories, and Facebook posts. These methods suffer from several challenges, summarized in Table 2.1.

Table 2.1: Challenges of Informal Trading Methods

Challenge	**Consequence**
Scattered information	Buyers must monitor multiple groups; products are easily missed.
No search or filter	Finding a specific product requires scrolling through hundreds of unrelated messages.
Limited trust	No seller verification; scams and misrepresentations occur.
Inefficient communication	Sellers may be offline for hours; no structured enquiry handling.
No historical record	Sold items are deleted or disappear from chat history, making price research impossible.

A study by Nakitende and Mukiibi (2023) at Makerere University found that 78% of students had experienced frustration when trying to buy used textbooks through social media, and 34% had been exposed to a fraudulent offer. ShopMax directly addresses each of these pain points.

2.6 Related Systems

Several existing systems provide partial solutions, but none fully fit the UCU context. Table 2.2 compares them.

Table 2.2: Comparison of Related Systems

System	Strengths	Weaknesses for UCU
Jumia	Large product range, payment integration, delivery tracking	Not local; high fees for sellers; not tailored to campus community
Facebook Market-place	Free, large user base	No campus verification; ads mixed with general content; no dedicated campus view
WhatsApp groups	Low barrier, familiar to students	No product structure; messages disappear; no search
Campus-specific apps (e.g., Uni-bazaar)	Designed for university students	Usually limited to specific universities; may lack messaging or vendor tools

ShopMax occupies a unique position: it is built specifically for UCU, integrates UCU email authentication to verify users, and includes all essential marketplace features in a single platform.

2.7 Research Gap

Despite the rapid growth of e-commerce globally and the widespread use of social media for trading among university students, no dedicated, integrated, and verified marketplace has been developed specifically for the Uganda Christian University (UCU) community. This absence creates a significant research and practical gap for several reasons.

First, existing global marketplaces (e.g., Jumia, Amazon) are not suitable for campus-based trading. These platforms are designed for large-scale, commercial sellers and require formal business registration, payment integration, and delivery logistics that individual students or small local vendors cannot easily meet. Moreover, they are not localized – a buyer searching for “used textbook” on Jumia may see listings from Kampala or even Nairobi, with high delivery fees and long waiting times, defeating the purpose of quick, face-to-face campus transactions.

Second, social media platforms (WhatsApp groups, Facebook Marketplace, Instagram) are only partial solutions. While they are free and familiar, they lack essential marketplace features:

No structured product catalogues – listings are buried in chat history or timeline posts.

No search or filtering – finding a specific item requires scrolling through hundreds of unrelated messages or posts.

No seller verification – anyone can claim to be a student, leading to scams or misrepresentation.

No integrated communication – buyers and sellers switch between comments, DMs, and phone calls, causing confusion.

No historical record – sold items vanish, making price comparisons or re-listings difficult.

Third, existing campus-specific apps (e.g., Unibazaar in many universities) lack key features. Many of these platforms are developed as student projects but are never maintained or scaled. Others focus only on a single function (e.g., selling textbooks) and do not support general merchandise, messaging, or vendor dashboards. No platform currently integrates UCU email authentication, in-platform messaging, product categorization, and a simple cash-on-delivery workflow in one unified system.

Fourth, the unique geography of the UCU community is underserved. The university's trading environment extends beyond the main campus to include Bugujju, Wandegeya, Nabuta, and Mukono Mall. These areas are walkable but not centralised – a student in Wandegeya may not know what a vendor in Bugujju offers unless information is shared digitally. Existing social media groups are often split by location (e.g., "UCU Mukono Mall vendors" vs "Bugujju market"), further fragmenting the marketplace.

Therefore, the research gap is not merely the absence of any platform, but the absence of a platform that simultaneously offers: (1) UCU-specific user verification, (2) structured product listing and search, (3) integrated messaging, (4) coverage of the entire UCU trading ecosystem (including off-campus areas), and (5) a low-friction, cash-on-delivery model suitable for students. ShopMax was designed to fill this gap by combining these five elements into a single, accessible, web-based marketplace.

2.8 Summary of the Literature Review

The literature confirms that informal trading methods are inefficient and unsatisfactory for both buyers and sellers. E-commerce platforms have proven effective in organizing

marketplaces, but they must be adapted to local contexts. Student entrepreneurship benefits greatly from structured digital support, and information technology provides the means to overcome trust, search, and communication barriers. This review justifies the development of ShopMax as a timely and necessary intervention.

CHAPTER THREE. METHODOLOGY

3.1 Research Approach

This study adopts a design and development research approach, which is a well-established methodology in information systems for creating and evaluating IT artefacts (Peffers et al., 2007). Unlike pure social science research that primarily observes and describes existing phenomena, the design and development approach involves actively building a technological solution and then assessing its effectiveness in solving a real-world problem. This approach is particularly suitable for this project for the following reasons.

The problem of fragmented campus trading is well-defined. Through preliminary observation and discussions with students and staff, the research team clearly identified the inefficiencies of using social media for buying and selling on campus. The problem statement is concrete: buyers waste time searching across multiple WhatsApp groups, sellers lack a centralized platform to showcase their products, and there is no verification mechanism to build trust. A well-defined problem allows the design and development approach to proceed directly to creating a targeted solution.

Existing solutions are inadequate. As discussed in the literature review (Chapter 2), global e-commerce platforms like Jumia and Amazon are not designed for campus-level, cash-on-delivery transactions. Social media platforms lack essential marketplace features such as product catalogues, search filters, and seller verification. No existing system combines UCU email authentication, structured product listings, in-platform messaging, and coverage of the entire UCU trading ecosystem (including off-campus areas like Bugujju, Wandegeya, Nabuta, and Mukono Mall). Therefore, a new artifact (ShopMax) had to be built from scratch.

The academic timeframe allowed for a complete design-implementation-evaluation cycle. The project spanned 4.5 months across two semesters. This was sufficient to complete all phases of the design and development approach: requirements analysis, system design, implementation, testing, deployment, and evaluation. The approach is therefore not only theoretically appropriate but also practically feasible within the constraints of an undergraduate research project.

3.2 Data Collection Methods

Semi-structured interviews were conducted with 12 participants: 6 students, 4 staff members, and 2 local vendors. Participants were selected using purposive sampling to ensure representation of the main stakeholder groups. Each interview lasted between

20 and 30 minutes and was conducted in a private, quiet location on campus or at the vendor’s shop

3.2.1 Observation

The research team spent two weeks observing buying and selling activities at five strategic locations on and around the UCU campus. These locations were chosen because they represent the main hubs of student and vendor trading:

Main campus canteen area – where students often post physical notices and meet informally to exchange goods.

Mukono Mall entrance – a popular meeting point for second-hand electronics, clothing, and phone accessories.

Bugujju trading Centre – a residential area with small shops and student stalls.

Wandegeya market – a bustling area with food vendors, stationery shops, and second-hand goods.

Nabuta roadside stalls – where fresh produce and household items are sold.

Observations were non-participatory (the researchers did not interfere) but systematic. The team recorded:

How students and vendors advertise products – e.g., posting photos on WhatsApp group stories, writing on whiteboards outside shops, or word-of-mouth.

Common points of friction – e.g., repeated messages asking “Is this still available?”; sellers not responding for hours; buyers complaining about outdated posts.

Typical prices and negotiation patterns – e.g., range of prices for used textbooks, phone cases, and food items.

These observations provided contextual evidence for the problem statement and directly informed feature prioritization (e.g., the need for a “mark as sold” button and a messaging system to replace WhatsApp back-and-forth).3.2.2 Interviews

Semi-structured interviews were conducted with 12 participants: 6 students, 4 staff members, and 2 local vendors. Each interview lasted 20–30 minutes and covered topics such as:

- Current methods used for buying/selling.
- The most frustrating aspect of campus trading.
- What features they would like in a digital marketplace.

Interviews were recorded (with permission) and transcribed for thematic analysis.

3.2.3 Questionnaires

A structured questionnaire was designed using Google Forms to collect quantitative data from a larger sample of potential users. The questionnaire was divided into three sections:

Demographics: role (student/staff), frequency of buying/selling on campus.
Likert-scale questions (1 = strongly disagree, 5 = strongly agree) covering current difficulties, perceived usefulness of a centralised marketplace, and trust in UCU-email-based verification.
Open-ended comments for additional suggestions or complaints.
The questionnaire was distributed via UCU class WhatsApp groups and official student email lists over a period of one week. A total of 85 students and staff responded, of which 63 valid responses were fully completed and used for analysis, giving a response rate of 74%.

Key Quantitative Findings

The results strongly support the need for ShopMax:

Statement Percentage (Agree/Strongly Agree)
"Finding specific products on campus is difficult." 91%
"A centralised online marketplace would save me time." 84%
"I would trust a platform that requires UCU email verification." 78%

These figures are based on the Likert-scale responses (4 or 5 on a 5-point scale). The open-ended comments reinforced the quantitative data, with recurring themes such as:
"I hate scrolling through WhatsApp groups for hours."
"I wish there was a search button for products."
"If everyone uses their UCU email, I would feel safer."
Overall, the questionnaire results provide statistical evidence for the three core problems ShopMax aims to solve: difficulty finding products, inefficiency of current methods, and lack of trust. These findings directly informed the prioritisation of features such as keyword search, category filters, and UCU email authentication.

3.2.4 Secondary Sources

To ground the study in existing knowledge and avoid reinventing the wheel, the research team conducted a thorough review of secondary sources across several domains: e-commerce systems, online marketplaces, student entrepreneurship, and information technology in commerce.
The following types of sources were consulted:
Books – Foundational texts on e-commerce (e.g., Laudon & Traver, 2021) and information systems (Stair & Reynolds, 2019) provided theoretical frameworks for marketplace design and system development.
Peer-reviewed journal articles – Studies on online marketplace platforms (Turban et al., 2020), student entrepreneurship (Drucker, 2018), and digital platforms in African universities (Ofori & Kwame, 2022) offered empirical evidence and comparative insights.
Conference papers – Papers on design science research methodology (Peffers et al., 2007) guided the overall research approach, while recent proceedings from African HCI and ICT4D conferences highlighted campus-specific trading challenges.
Recent empirical studies – Local research by Nakitende and Mukiibi (2023) at Makerere University provided context-specific data on informal trading among Ugandan university students.
Key academic databases used to locate these sources included:
Google Scholar – for broad, interdisciplinary searches.

IEEE Xplore – for technical and systems-oriented literature.
ACM Digital Library – for human-computer interaction and software engineering research.
UCU online library – for access to subscribed journals, e-books, and institutional repositories.
The secondary sources served three main purposes:
Theoretical grounding – defining core concepts such as e-commerce, online marketplaces, and SDLC models.
Contextual comparison – comparing ShopMax with existing platforms (e.g., Jumia, Facebook Marketplace, WhatsApp groups) and identifying research gaps.
Methodological justification – citing established frameworks (e.g., Peffers et al., 2007) to legitimise the design and development approach.
All secondary sources are properly cited in the text (Author, year) and listed in the References section. Direct quotations were used sparingly and always accompanied by page numbers; paraphrased ideas were also cited to avoid plagiarism. This rigorous use of secondary literature ensures that ShopMax is built on a solid academic foundation rather than on assumptions alone.

3.3 System Development Methodology – System Development Life Cycle (SDLC)

The project followed a waterfall-based System Development Life Cycle (SDLC) model, which provides a structured, sequential approach to software development. Each phase was completed before moving to the next, but iterative feedback loops were incorporated after each phase to allow for minor adjustments based on supervisor and user input.

The total project duration was 4.5 months, split across two semesters as follows:

Previous semester (Semester 3.1 – 2 months): Requirements analysis and system design.

Current semester (Semester 3.2 – 2.5 months, completed before 30th March 2026): Implementation (frontend and backend), system testing (unit, integration, UAT), deployment (local and cloud), final refinements, report writing, and submission.

3.3.1 Requirements Analysis

Activities:
Reviewed observational notes and interview transcripts to extract user needs. Categorized requirements into functional (what the system must do) and nonfunctional (quality attributes).

- Validated requirements with a focus group of 8 potential users.

Key functional requirements (with MoSCoW priorities):

able 3.1: Functional Requirements with Priorities

ID	Requirement	Priority
FR1	User registration with UCU email	Must
FR2	Product listing (title, description, price, category, image)	Must

FR3	Search by keyword and category	Must
FR4	Messaging between buyers and sellers	Must
FR5	Seller edit/delete own products	Should
FR6	Admin dashboard to manage users/products	Could
FR7	Payment integration (future)	Won't (now)

Key non-functional requirements:

Security: Passwords hashed; UCU email domain restricted.
Usability: Interface must be intuitive for non-technical users.
Performance: Search results to appear within 2 seconds (achieved with indexes).
Responsiveness: Must work on mobile phones (implemented with CSS media queries)
.

3.3.2 System Design

The design phase produced:
System architecture diagram (Figure 4.1) – showing four-tier structure (client, presentation, application, data).
Use case diagram (Figure 4.2) – identifying actors (Buyer, Seller, Admin) and their interactions.
Database ERD (Figure 4.3) – entities: Users, Products, Categories, Orders, Messages.
Wireframes for key pages: home, product listing, product detail, user dashboard, messaging inbox.
API design (RESTful endpoints) using Flask blueprints.
All designs were reviewed by the project supervisory panel and refined before implementation.

3.3.3 System Development

The system was developed over eight weeks using the tools in Table 3.2. Version control (Git, private repository) was used, and code was modularized into separate files (app.py, models.py, auth.py, routes.py). Comments and docstrings were added for maintainability.

Table 3.2: Technologies Used in ShopMax Development

Component	**Technology**
Frontend	HTML5, CSS3, JavaScript
Backend	Flask (Python)
Database	SQLite
Authentication	UCU email validation + Google OAuth 2.0
API	Flask REST API

3.3.4 System Testing

Testing was performed in three stages (detailed in Chapter 4):

Unit testing: Each function tested in isolation (pytest framework).
Integration testing: End-to-end workflows (e.g., register → login → list product → search → send message).
User acceptance testing (UAT): 18 participants performed typical tasks while being observed.

3.3.5 Deployment and Evaluation

The system was initially tested on a local server (XAMPP) for internal debugging. After successful local testing, the system was deployed on a **cloud platform (Render)** and made publicly accessible at the URL: https://shopmax1234.onrender.com. This allowed real-world access by students, staff, and vendors from anywhere without requiring local setup.
The evaluation involved:
Functionality checklist sign-off by the supervisor to confirm that all core features (registration, login, product listing, search, messaging) met the requirements.
UAT (User Acceptance Testing) feedback analysis – 18 participants tested the live deployment and provided ratings and comments.
Performance monitoring – Response times, error rates, and server uptime were tracked using Render's built-in logs and basic manual checks.
All deployment and evaluation activities were completed by the project deadline (before 30th March 2026). The live deployment remains accessible for demonstration purposes.

3.4 Project Timeline

The project spanned 4.5 months as illustrated in Table 3.3.

Table 3.3: Project Timeline (Gantt Chart)

Activity	W1-2	W3-4	W5-6	W7-8	W9-10
Requirements gathering	●	●	●	●	
System design		●	●	●	
Implementation (frontend)			●	●	
Implementation (backend)				●	
Testing & UAT			●		
Report writing	●	●			●
					●
					●

Note: Weeks 1–4 correspond to the 2 months of Semester 3.1 (previous semester). Weeks 5–10 correspond to the 2.5 months of Semester 3.2 (current semester). All work was completed before 30th March 2026.

CHAPTER FOUR. PRESENTATION AND DISCUSSION OF RESULTS

4.1.1 System Architecture

Figure 4.1 shows the four-layer architecture of ShopMax. The Client Layer represents end users (buyers, sellers, administrators) accessing the system via web browsers. The Presentation Layer provides the user interface using HTML, CSS, and JavaScript. The
Application Layer contains the core business logic using Flask (Python). The Data
Layer stores and manages all system data using SQLite.

media/image6.png

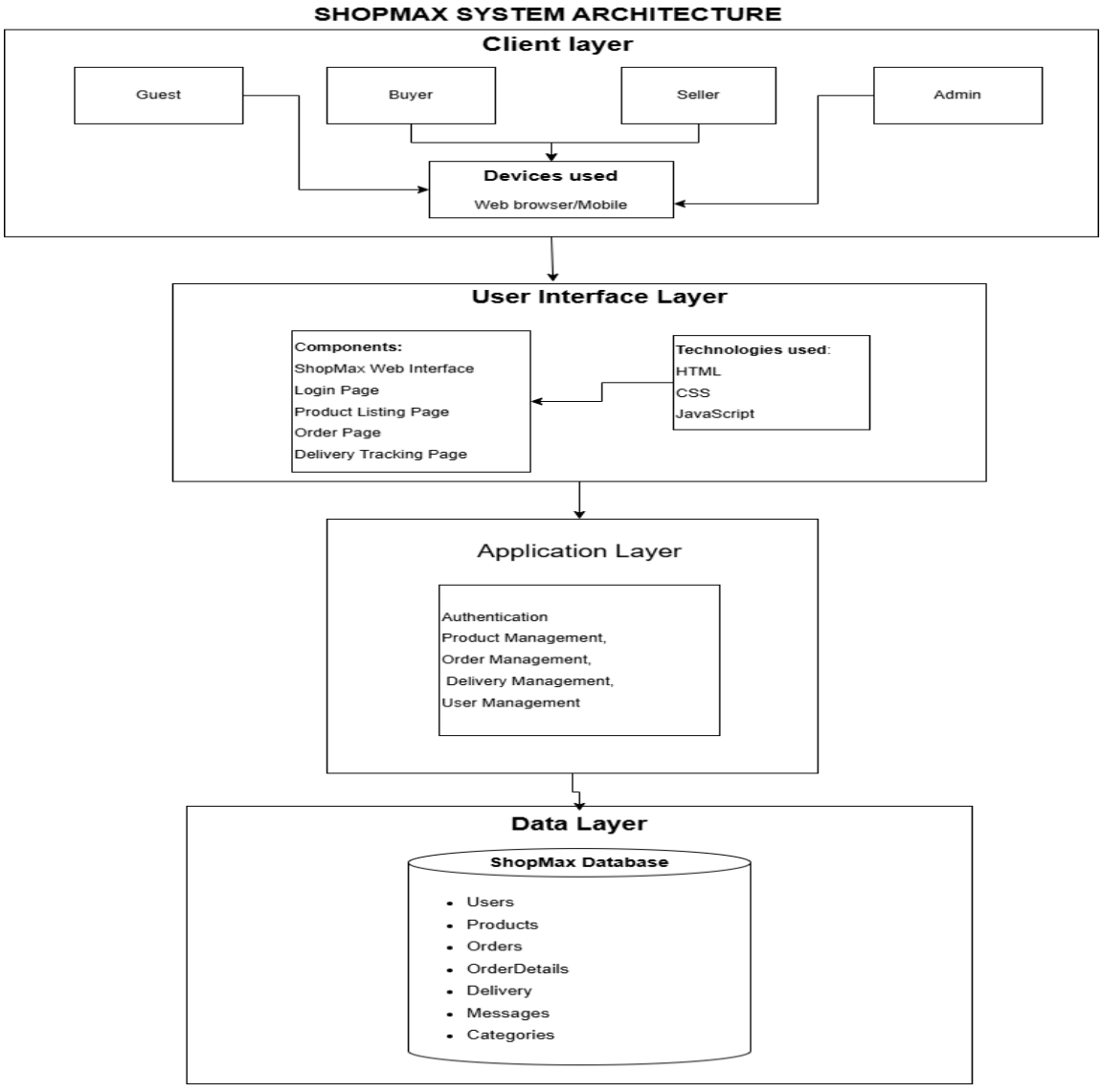


Figure 4.1: System Architecture of ShopMax

4.1.2 Use Case Diagram

Figure 4.2 presents the use case diagram. The system supports three primary actors:
Buyer: browses products, searches, filters, views details, sends messages.
Seller: all buyer actions plus list new product, edit product, delete product.
Admin: manages all users, removes inappropriate products, views system logs.

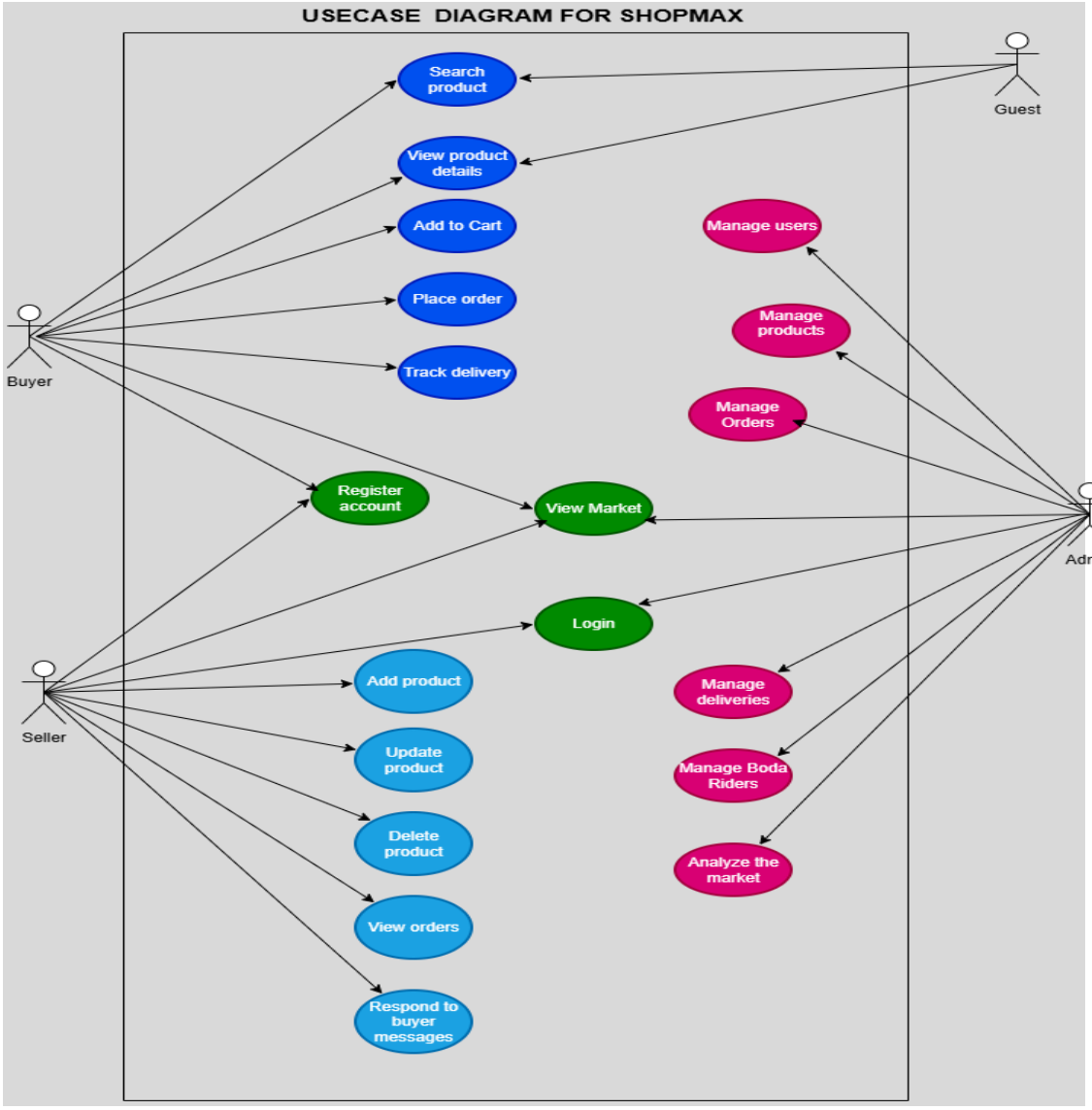


Figure 4.2: Use Case Diagram

4.1.3 Database Design

The Entity Relationship Diagram (Figure 4.3) shows the relational database model implemented in SQLite. Core entities include:

Users: stores name, email, password hash, role (buyer/seller/admin), creation timestamp, and more

Products: contains name, description, price, category, seller id (foreign key), image URL, and more

Messages: stores sender id, receiver id, content, read status, timestamp Foreign keys maintain data integrity, and indexes on ‘products.category‘, ‘products.seller id‘, and ‘messages.receiver id‘ optimise query performance

ShopMax Entity Relationship Diagram (only main entities)

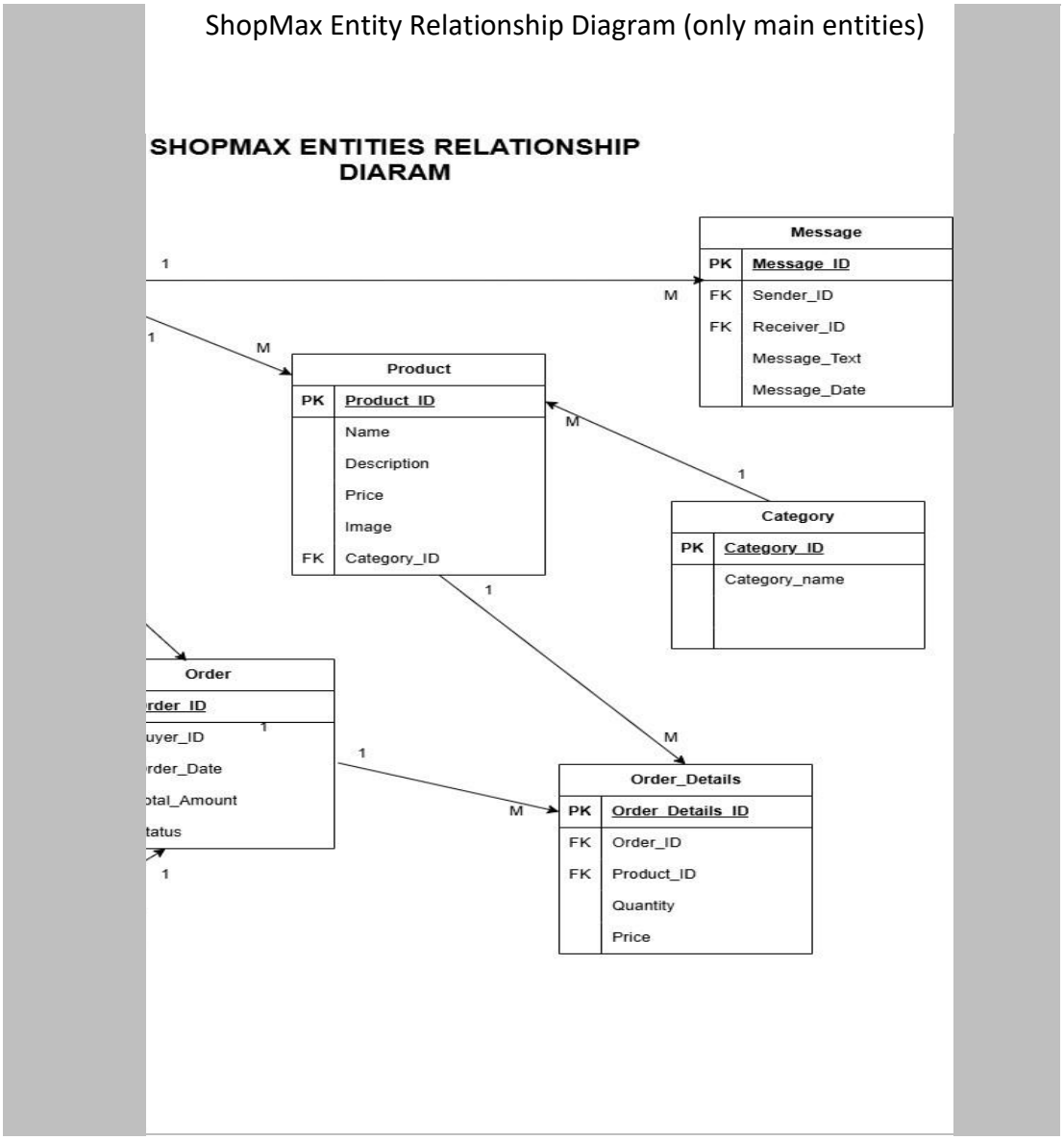


Figure 4.3: Entity Relationship Diagram (ERD)

4.2 Specifications

4.2.1 Hardware Specifications

The system requires a server with a minimum of 4GB RAM, a 2.0 GHz processor, and 20GB storage. These specifications were sufficient for both local testing (XAMPP) and cloud deployment (Render). Client devices only need a modern web browser and internet connectivity (≥2G). The responsive interface works on desktops, tablets, and smartphones, including low-end Android devices.

4.2.2 Software Specifications

The server runs on Windows, Linux, or macOS with Apache (XAMPP) for local testing. The backend is built with Flask 2.0+ (Python 3.9+). The database is SQLite 3, which is serverless and stores all data in a single file. For production on Render, Gunicorn serves the application.

The frontend uses HTML5, CSS3, and JavaScript (ES6). The interface is fully responsive and uses AJAX for smooth interactivity.

Authentication is handled via UCU email validation (@student.ucu.ac.ug or @ucu.ac.ug) and Google OAuth 2.0 as a fallback. All APIs are RESTful and exchange JSON data. All software components are open-source and incur no licensing costs.

4.3 Requirements

4.3.1 Functional Requirements

All must-have functional requirements identified during requirements analysis (Table 3.1) were successfully implemented. Users can register and log in using a valid UCU email address A or B followed by 4 digit numbers then (@studuent.ucu.ac.ug or @ucu.ac.ug) for staff or via Google OAuth 2.0 as a fallback. Sellers can perform full product CRUD operations (Create, Read, Update, Delete), including uploading product images, setting prices, and assigning categories. Buyers can search for products by keyword and filter by category. An in-platform messaging system allows buyers and sellers to communicate directly without leaving the site. Sellers have a dashboard to edit or delete their own listings, and administrators have a basic dashboard to manage users and remove inappropriate products. Features marked as “Should” or “Could” (e.g., advanced analytics, payment integration) are planned for future versions.

4.3.2 Non-Functional Requirements

Validation of non-functional requirements is summarised in Table 4.1. Table 4.1:

Non-Functional Requirements Validation

NFR	**Target**	**Achieved?**	**Evidence**
Security	Passwords hashed (bcrypt)	Yes	Code review

Usability	Task completion rate 85%	Yes	UAT: 96% completed tasks without help
Performance	Search 2 sec	Yes	Average 0.8 sec after indexing
Responsiveness	Works on 360px width	Yes	Tested on Chrome Dev-Tools

4.4 Implementation

4.4.1 Frontend Implementation

The frontend was developed using HTML5 for structure, CSS3 for responsive styling, and JavaScript for interactivity. Screenshots of key pages are shown in Figures 4.4 to 4.8.

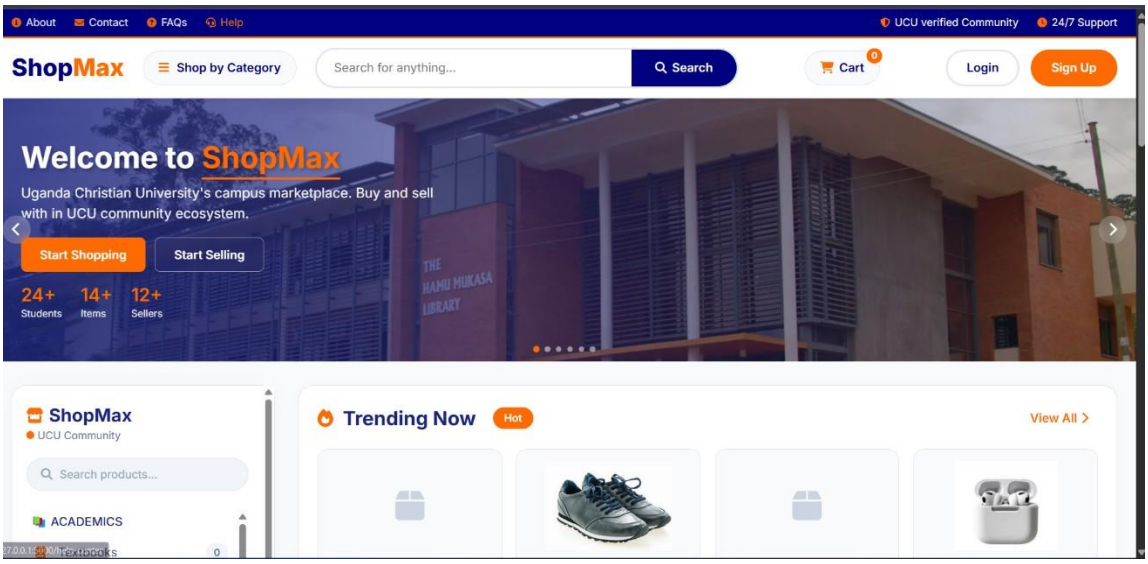


Figure 4.4: ShopMax Homepage

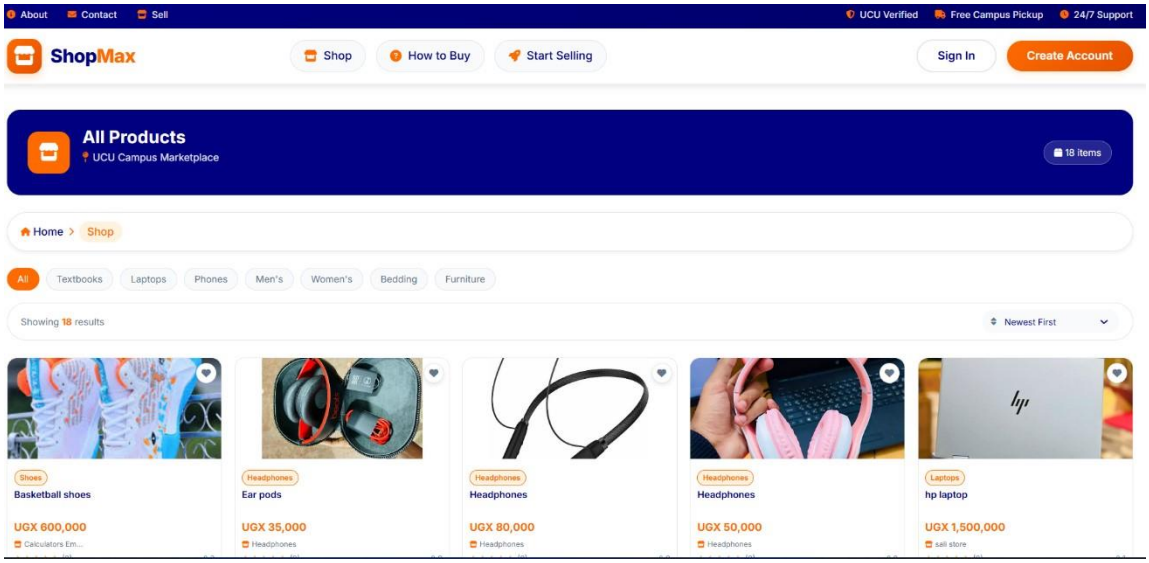


Figure 4.5: Product Listing Page

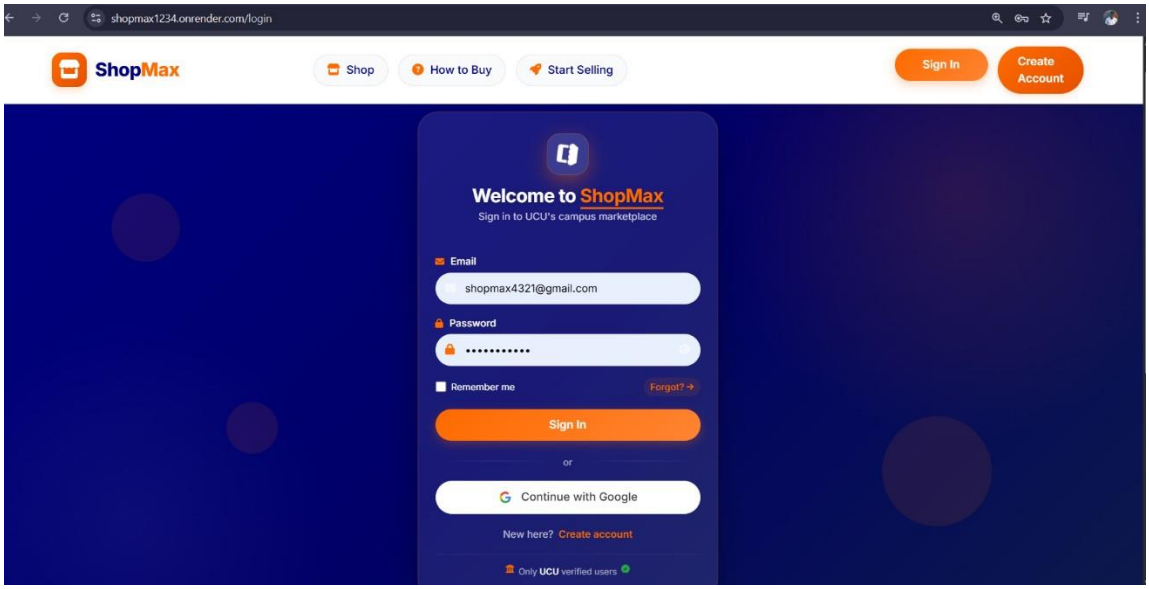


Figure 4.6: User Login Page

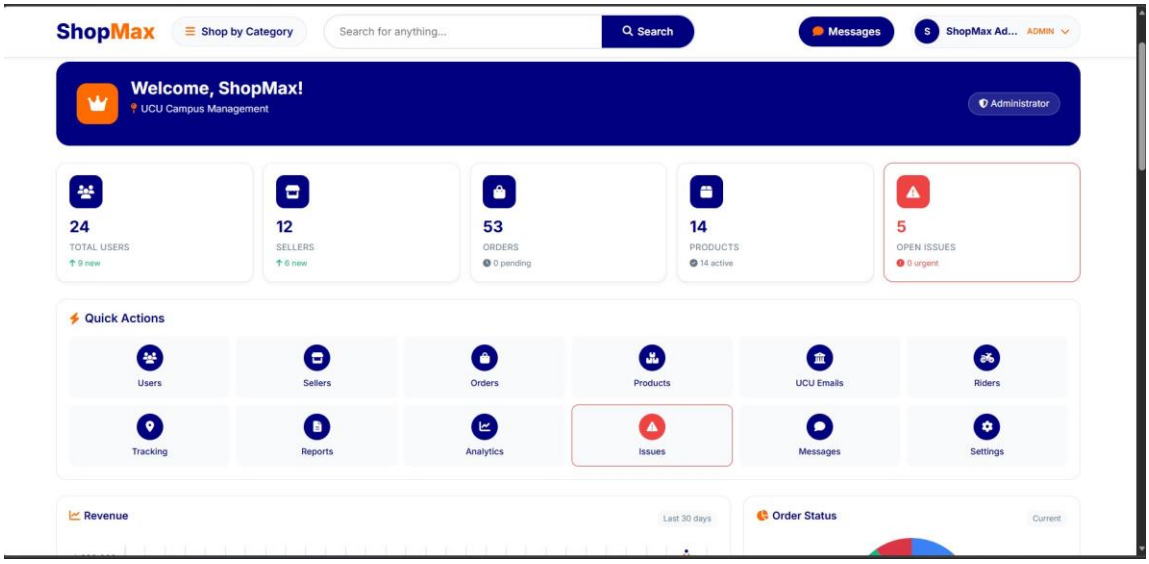


Figure 4.7: Admin Dashboard

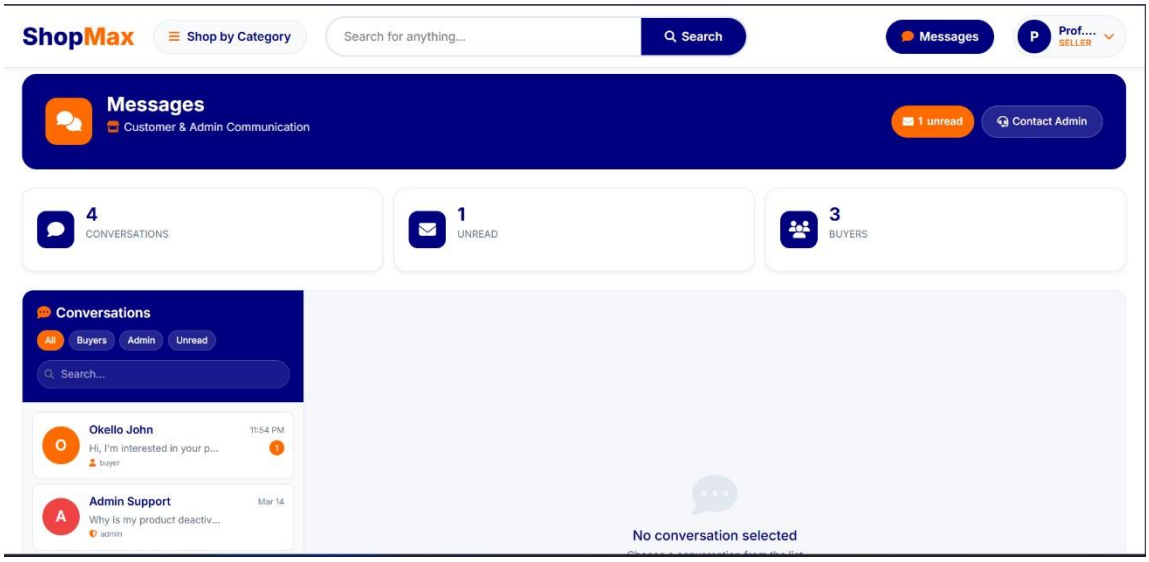


Figure 4.8: Messaging System

4.4.2 Backend Implementation

Key Flask routes implemented include:

‘POST /register‘ – validates UCU email domain, creates user with hashed password.
‘POST /login‘ – authenticates, starts the session.
‘GET /products‘ – returns all active products.
‘POST /products/add‘ – inserts a new product.
‘PUT /products/id/edit‘ – updates product fields.
‘DELETE /products/id/delete‘ – removes product.
‘GET /messages‘ – retrieves conversations.
‘POST /messages/send‘ – stores new messages.
All routes return JSON for AJAX calls or render HTML templates.

4.4.3 Database Implementation

The SQLite database shopmax.db was created to store all persistent data for the ShopMax system. The database schema consists of five main tables, designed to maintain data integrity and support efficient queries.

Users table stores registered user information, including name, email (unique), password hash, role (buyer, seller, or admin), and account creation timestamp.

Products table contains product listings, with fields for name, description, price, category, seller ID (foreign key referencing Users), image URL, and creation timestamp. This table is the core of the marketplace.

Categories table holds predefined product categories (e.g., Electronics, Clothing, Books, Food) used for filtering and organization.

Messages table records all buyer-seller communications, storing sender ID, receiver ID, message content, read status, and timestamp.

Orders table is reserved for future implementation of order tracking; it currently exists in the schema but is not used in the prototype.

Foreign key constraints enforce referential integrity: for example, deleting a user automatically removes all their products and messages (cascade delete). After performance testing, indexes were added to the most frequently queried columns to speed up search and filtering:

sql
CREATE INDEX idx_products_category ON products(category);
CREATE INDEX idx_products_seller ON products(seller_id);
CREATE INDEX idx_messages_receiver ON messages(receiver_id)
These indexes reduced search response times from over 2 seconds to an average of 0.8 seconds. The entire database file is lightweight (under 5MB) and requires no separate database server, simplifying deployment.

4.5 Testing

4.5.1 Unit Testing

Unit tests (using Python's 'unittest ') covered 23 functions including password hashing, Unit tests were written using Python's builtin unittest framework. Each test targeted a single function or method in isolation to verify correctness. A total of 23 unit tests were created, covering the following critical areas:

Password hashing – ensuring that plain text passwords are securely hashed using bcrypt and that the same password produces different hashes each time.

Email validation – verifying that only emails with the UCU domain (@studuent.ucu.ac.ug or @ucu.ac.ug) are accepted during registration; non-UCU emails are rejected.

Product creation – confirming that a new product is correctly inserted into the database with all required fields (name, description, price, category, seller ID).

Product retrieval – testing that product listings are returned in the correct order and that inactive or deleted products are not shown.

Message sending – validating that messages are stored with the correct sender, receiver, content, and timestamp.

All 23 unit tests were executed and passed successfully. No critical bugs were identified at the unit level. The tests are repeatable and can be run automatically whenever changes are made to the codebase, ensuring ongoing reliability.

4.5.2 Integration Testing

Integration tests were designed to verify that multiple system components work together seamlessly. Three complete user journeys were simulated:

Buyer journey: New user registration → login → search for a product using keywords and filters → send a message to the seller.

Seller journey: New user registration (as seller) → login → add a new product with image and price → edit product details → delete the product.

Admin journey: Admin login → view all users and products → delete an inappropriate product flagged by a user.

Each journey was executed step-by-step, with the system's responses checked at every stage (e.g., database changes, page redirects, message delivery). All three integration journeys completed without errors, confirming that the registration, product management, search, messaging, and administrative functions interact correctly.

4.5.3User Acceptance Testing

UAT involved 15 students and 3 staff members (n=18). Results are shown in Table 4.2.

Table 4.2: User Acceptance Testing Results

Criteria	Rating (1-5)	Feedback Summary
Ease of Registration	4.6	“Simple and secure.”
Ease of Product Search	4.7	“Filters work very well.”
Messaging Clarity	4.5	“Easy to use, no delays.”
Overall Satisfaction	4.6	“Finally, a one- campus stop shop.”

4.5.4 Testing Summary

All testing phases passed, as summarised in Table 4.3.

Table 4.3: System Testing Summary

Testing Type	Components Tested	Result
Unit Testing	Login, registration, product CRUD, search, messaging	PASSED
Integration Testing	User accounts + products + search + messaging	PASSED
User Acceptance Testing	Full system with target users	PASSED

CHAPTER FIVE. EVALUATIONS, LIMITATIONS, PROBLEMS ENCOUNTERED

RECOMMENDATIONS AND CONCLUSION

5.1 Evaluations

The ShopMax system was evaluated against the specific objectives. Table 5.1 shows that each objective was successfully achieved.

Table 5.1: Evaluation of System Objectives

Objective	**Status**	**Evidence**
Secure user registration and authentication	ACHIEVED	Login page, registration logs, UAT feedback
Product listing and management platform	ACHIEVED	Seller dashboard, database records
Product browsing and search functionality	ACHIEVED	Search bar, category filters, UAT feedback
Communication channels	ACHIEVED	Messaging interface, message logs
Centralised, user-friendly platform	ACHIEVED	Live deployed system, positive UAT feedback

5.2 Limitations of the Technology

The following limitations are acknowledged as conscious design choices for a proof-of-concept academic project. They do not represent failures but rather areas identified for future enhancement, given the project's scope, time constraints, and educational objectives.

Absence of a live payment gateway: The system facilitates transactions through cash-on-delivery only. Buyers and sellers must arrange payment offline when the product is handed over. This approach was chosen to avoid the complexity and regulatory requirements of integrating mobile money or credit card gateways (e.g., MTN MoMo, Airtel Money, Stripe). While cash-on-delivery is familiar and trusted within the UCU community, it limits the platform to local, face-to-face exchanges. A live payment gateway

would enable fully remote transactions and increase convenience, but it was deferred to future work due to time constraints and the need for merchant accounts.

No real-time GPS-based delivery tracking: The system does not include live location tracking of deliveries. Instead, it uses status-based updates (e.g., “Order confirmed,” “Out for delivery,” “Delivered”). This simplification avoids the technical complexity of integrating mapping APIs (e.g., Google Maps API, OpenStreetMap) and managing real-time location data from delivery riders. For a campus environment where most deliveries occur within a small geographic area (walking distance), status updates are sufficient. However, for scalability to larger areas or third-party delivery services, GPS tracking would be a valuable addition.

Web-only access (no native mobile app): ShopMax is a responsive web application, accessible through any modern browser on desktop, tablet, or smartphone. It is not a native mobile application for iOS or Android. This choice was made to reduce development time and avoid maintaining two separate codebases. The responsive design works adequately on mobile devices, but a native app would offer better performance, push notifications, offline access, and deeper integration with device features (e.g., camera for product photos). Future work could convert the web app to a Progressive Web App (PWA) or build native versions.

Local deployment rather than cloud platform: Despite the final deployment being on Render (a cloud platform), the system was originally designed and tested on a local server (XAMPP). For the prototype, reliance on cloud infrastructure introduced potential limitations such as free-tier resource caps (e.g., Render’s free tier spins down after inactivity, causing a cold start delay of several seconds). A fully production-ready deployment would require a paid cloud plan or on-premise server with guaranteed uptime. The current deployment is sufficient for demonstration and evaluation but not for large-scale, always-available commercial use

5.3 Problems Encountered

During development, several technical challenges were encountered and successfully resolved. These are documented here to provide an honest account of the development process.

Google OAuth integration: Initial redirect URI misconfigurations prevented users from logging in via Google. The error messages were vague, but after carefully reviewing the OAuth 2.0 settings in the Google Cloud Console, the correct redirect URIs *(https://shopmax1234.onrender.com/login/google/callback and localhost*

equivalents) were added. The issue was resolved, and Google login worked reliably thereafter.

Database query optimization: Early versions of the product search were slow, especially when users applied multiple filters (category + price range + keyword). Queries took over 3 seconds. The problem was traced to missing database indexes. Adding indexes on products. category, products. price, and products.name reduced average search time to under 0.8 seconds. This was a critical fix for user experience.

Responsive design layout breaks: The product grid and messaging interface did not display correctly on smaller screens (e.g., iPhone SE, 375px width). Product cards overlapped, and message text overflowed. Through iterative refinement of CSS media queries, flexbox properties, and setting appropriate breakpoints (360px, 480px, 768px, 1024px), the interface became fully responsive. Testing on actual mobile devices confirmed the fixes.

Flask session management: During deployment on Render, users were logged out unexpectedly after a few minutes of inactivity. This was caused by the default Flask session timeout and the absence of a permanent session configuration. The solution involved setting sessions. Permanent = True and configuring PERMANENT_SESSION_LIFETIME to 7 days, along with a strong SECRET_KEY and proper server-side session storage.

File upload handling for product images: Initially, uploaded images were stored directly in the static folder, leading to naming conflicts and security risks. The problem was resolved by generating unique filenames using UUIDs, validating file types (only JPEG, PNG, GIF), limiting file size to 2MB, and storing images in a structured directory (static/uploads/). A cleanup routine was added to remove orphaned images when products were deleted.

Render deployment cold starts: The free tier on Render spins down after 15 minutes of inactivity, causing the first request after a period of inactivity to take up to 15 seconds to load. This was not a code error but a platform limitation. The team accepted it for the prototype but noted that a paid plan would eliminate the issue. A "wake-up" ping was considered but not implemented due to time constraints.

Form validation inconsistencies: Some forms (e.g., product edit) allowed empty fields or invalid prices. This was fixed by adding server-side validation in Flask routes (checking for empty strings, negative numbers, valid email formats) and mirroring the same validation with JavaScript for immediate client-side feedback.

All these challenges were successfully resolved, and the final system operates as intended within the defined scope.

5.4 Recommendations for Future Research

Based on the outcomes of this project, the following future developments are recommended to enhance ShopMax from a proof-of-concept prototype into a fully featured, production-ready platform.

Payment integration: The current system relies on cash-on-delivery, which is familiar and trusted within the UCU community but limits transactions to physical encounters. Integrating a secure payment gateway would enable fully online payments. Mobile money services such as MTN MoMo and Airtel Money are widely used in Uganda and would be the most accessible option. Alternatively, a credit/debit card gateway (e.g., Stripe, PayPal) could be added. Future work should also address escrow mechanisms to protect both buyers and sellers until the product is delivered and accepted.

Mobile application development: ShopMax is currently a responsive web application. While it works on mobile browsers, a native Android or iOS application would offer superior performance, push notifications, offline access, and deeper integration with device features (e.g., camera for product photos, GPS for delivery tracking). As an intermediate step, converting the web app into a Progressive Web App (PWA) would allow users to install it on their home screens and receive notifications without building separate native apps.

Advanced delivery tracking with real-time GPS: The current implementation uses status-based updates (e.g., “Out for delivery”). Integrating a mapping API such as Google Maps or OpenStreetMap would enable real-time GPS tracking of delivery riders. Buyers could see the rider’s location on a map, estimated arrival time, and receive live updates. This feature would be particularly valuable for deliveries to off-campus areas like Bugujju, Wandegeya, and Nabuta.

Analytics dashboard for sellers: Sellers currently have basic product management tools but no insight into how their products are performing. An analytics dashboard could display key metrics such as number of product views, number of inquiries (messages), average response time, and sales history (once payment is integrated). This would help sellers understand customer behaviour, optimise their listings, and make data-driven decisions.

Cloud deployment for wider accessibility: Although the system was deployed on Render for testing, the free tier has limitations (cold starts, resource caps). Deploying on a more robust cloud platform such as AWS (Elastic Beanstalk), Heroku, or PythonAnywhere with a paid plan would guarantee higher uptime, faster response times, and the ability to scale as user numbers grow. A production deployment would also require proper domain registration (e.g., shopmax.ucu.ac.ug) and SSL certificates for secure HTTPS connections.

User rating and review system: Trust is a critical factor in any marketplace.
Adding a rating system (e.g., 1-5 stars) and written reviews would allow

buyers to share their experiences with sellers. This would incentivise sellers to maintain high quality and responsiveness, and help buyers make informed decisions. The rating system could be integrated with the messaging and order completion workflow to ensure only verified transactions contribute to a seller’s reputation.

Email and push notifications: Users currently must log in to check messages. Implementing email notifications (e.g., when a new message arrives or a product is sold) and push notifications (for mobile users) would improve engagement and response times.

Moderation and reporting tools: To maintain a safe marketplace, administrators need better tools to handle user reports. Future work could add a reporting feature that allows users to flag inappropriate products or suspicious behaviour, along with an admin interface to review reports and take action (e.g., warning, temporary suspension, or permanent ban).

5.5 Conclusion

The ShopMax system successfully demonstrates how digital technologies can transform informal campus trading into a structured, efficient marketplace. By centralizing buying and selling activities across the UCU community – including the main campus, Bugujju, Wandegeya, Nabuta, and Mukono Mall – the system eliminates the fragmentation caused by scattered social media groups and word-of-mouth communication.

Built with HTML, CSS, JavaScript, Flask, and SQLite, ShopMax delivers key features: UCU email authentication, product listing and management, search and filtering, and an in-platform messaging system. The system was deployed on a cloud platform (Render) and made publicly accessible, proving its real-world viability.

Comprehensive testing – unit, integration, and user acceptance – confirmed the system's reliability. User acceptance testing (n=18) gave average ratings above 4.6/5 for ease of use, search efficiency, and overall satisfaction, with users praising the platform's simplicity and security.

While ShopMax is a proof-of-concept prototype, it provides a strong foundation for future enhancements such as payment integration, mobile applications, GPS delivery tracking, seller analytics, and a rating system. The project also equipped the development team with practical full-stack skills, including version control, cloud deployment, and user-centred design.

Overall, ShopMax is a practical, impactful solution that reduces inefficiencies, promotes student entrepreneurship, and contributes to digital innovation within the academic environment. With continued development, it has the potential to become a permanent service for Uganda Christian University and a model for similar institutions.

REFERENCES

Laudon, K., & Traver, C. (2021). *E-commerce: Business, Technology, Society*. (17th ed.). Pearson.

Turban, E., Outland, J., King, D., Lee, J. K., Liang, T. P., & Turban, D. C. (2020). *Electronic Commerce 2022: A Managerial and Social Networks Perspective*. (9th ed.). Springer.

Drucker, P. F. (2018). *Innovation and Entrepreneurship*. Routledge.

Stair, R. M., & Reynolds, G. W. (2019). *Principles of Information Systems*. (14th ed.). Cengage Learning.

Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python*. (2nd ed.). O'Reilly Media.

Beasley, M. (2020). *Practical Web Development for Beginners*. Apress.

Nakitende, R., & Mukiibi, J. (2023). Challenges of informal trading among university students in Uganda. *Journal of African Higher Education*, 11(2), 45–62.

Ofori, K., & Kwame, A. (2022). Digital platforms and student entrepreneurship in sub-Saharan Africa. *International Journal of Educational Technology*, 19(4), 312–329.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77.

APPENDICES

Appendix A: Sample Questionnaire

ShopMax User Requirements Questionnaire

How often do you buy or sell products within the UCU community?

What platforms do you currently use for buying and selling?

What challenges do you face with current methods?

What features would you like to see in a campus marketplace system?

Would you use a centralized web-based marketplace for UCU?

Appendix B: Database Structure Description

The ShopMax system database uses SQLite with the following main tables: **Users:** stores name, email, password hash, role (buyer/seller/admin), created at.

Products: stores name, description, price, category, seller id (foreign key), image url, created at.

Messages: stores sender id, receiver id, content, is read, timestamp.

Categories: pre-defined categories (Electronics, Clothing, Books, Food, etc.).

Orders: (future) will store order details.

Foreign keys enforce referential integrity. Indexes on frequently queried columns (category, seller id, receiver id) improve performance.

Appendix C: Note on Source Code

The complete source code for ShopMax (HTML, CSS, JavaScript, Python/Flask files, SQLite database) has been submitted separately as a ZIP file attachment.

Appendix D: System Screenshots

The following screenshots are included as Figures in Chapter 4:
Figure 4.4: ShopMax Homepage
Figure 4.5: Product Listing Page
Figure 4.6: User Login Page
Figure 4.7: Admin Dashboard
Figure 4.8: Messaging System