

AI-Driven Lost and Found Management System for Institutions.

Admission Number: 149329

Twayigize Bienvenu

Group: BBIT 4C

An Information System Project 2 Documentation Submitted to the School of Computing and Engineering Sciences in partial fulfilment of the requirements for the award of a degree in Bachelor of Business Information Technology

School of Computing and Engineering Sciences

Strathmore University

Nairobi, Kenya.

Date of Submission: December 2025

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the research proposal contains no material previously published or written by another person except where due reference is made in the research proposal itself.

Student's signature:

A handwritten signature in black ink, appearing to read 'B. Clance', written over a horizontal line.

..... [Signature]

.....November 27th, 2025.... [Date]

Approval

The Information System Project 2 documentation of 149329Twayigize Bienvenu was reviewed and approved (*for examination*) by:

Supervisor's signature:

..... [Signature]

..... [Date].

Abstract

Loss of personal belongings remains a persistent challenge in schools, campuses, and public institutions, often resulting in frustration and reduced trust in existing manual lost-and-found procedures. Traditional approaches rely on handwritten logs and physical storage, which limits accessibility, delays recovery, and provides no intelligent way to match lost items with those found. This project proposes an intelligent Lost and Found Management System, a web-based platform designed to automate the reporting, matching, and retrieval of lost items. The system enables users to report lost or found items through structured digital forms incorporating descriptions, categories, locations, dates, and uploaded images. A key innovation of the system is the integration of artificial intelligence for multimodal item matching. The project employs OpenAI's CLIP model, combining image embeddings and text embeddings to compute similarity scores between lost and found items. This approach improves accuracy by simultaneously considering both visual and textual features. The AI pipeline is developed and evaluated using Google Colab with GPU acceleration, applying metrics such as accuracy, precision, recall, and F1-score, to measure model performance. The backend architecture integrates FastAPI for AI inference services and Laravel (PHP) for system logic and user management, while the frontend is implemented using HTML, CSS, JavaScript, Bootstrap, and React to ensure a responsive and user-friendly interface. Additional system capabilities include administrative verification of submitted items, automated match notifications, real-time status updates, and generation of digital claim receipts. The project follows a prototyping methodology, enabling iterative refinement driven by continuous user feedback. Overall, the system provides a scalable, transparent, and intelligent solution that enhances item recovery rates, promotes ethical handling of found items, and strengthens trust within institutional environments.

Keywords: Lost and Found System; CLIP Model; Multimodal Matching; Image-Text Similarity; FastAPI; Laravel; Web Application; Prototyping Methodology

Acknowledgements

To God, the source of all wisdom and perseverance, I give thanks for granting me the strength, guidance, and resources needed to complete this project. I would also like to express my sincere gratitude to my supervisor, Dr. Henry Muchiri, and my mentor, Mr. Joseph Mungai, for their continuous support and guidance from the beginning to the end of this work. My appreciation also goes to my classmates for their helpful insights and advice throughout the project.

I am forever indebted to my parents, Niyitegeka Felix and Nzarubara Claudine, for their sacrifices, love, and commitment in shaping the person I have become. I am grateful to my siblings, Niyitegeka Sarah, Niyitegeka Daniella, and Twayigize Felix Junior, for their constant support and encouragement. I also express my heartfelt appreciation to my close friends, Mpweto Benjamin, Bahati Melchior, Alpha Bahanuzi, Guy N'Nahano, and Mubalama Samuel, for their prayers, motivation, and valuable advice throughout this journey.

Table of Contents

Declaration.....	ii
Approval	iii
Abstract.....	iv
Acknowledgements.....	v
Table of Contents	vi
List of Figures.....	x
List of Tables.....	xi
Chapter 1: Introduction.....	1
1.1 Background.....	1
1.2 Problem Statement.....	1
1.3 Aim.....	2
1.4 Specific Objectives	2
1.5 Research Questions.....	2
1.6 Justification.....	3
1.7 Scope and Limitation	3
Chapter 2: Literature Review.....	4
2.1 Introduction.....	4
2.2 Current Practices in Lost and Found Management Systems	4
2.2.1 Technologies Used in Lost and Found Systems.....	4
2.3 Challenges Experienced in Existing Lost and Found Systems.....	5
2.3.1 Inefficiencies in Item Reporting and Retrieval.....	6
2.3.2 User Experience, Accessibility, and Communication Limitations.....	6
2.4 Review of Existing Systems	6
2.4.1 Troov.....	7
2.4.2 MissingX.....	7
2.4.3 Deliverback.....	8

2.4.4 Gaps in the existing system.....	8
2.5 Proposed System.....	9
2.6 Conceptual framework.....	10
Chapter 3: System Development Methodology.....	11
3.1 Introduction.....	11
3.2 Prototyping Methodology	11
3.2.1 <i>System Planning</i>	12
3.2.2 <i>Analysis</i>	12
3.2.3 <i>Design</i>	12
3.2.4 <i>Build Prototype</i>	13
3.2.5 <i>Testing</i>	13
3.2.6 <i>Implementation</i>	13
3.3 Justification of the methodology.....	13
3.4 System Analysis.....	14
3.4.1 Functional Requirements	15
3.4.2 Non-Functional Requirements	16
3.5 System Development Tools and Techniques.....	16
3.6 Machine Learning Algorithms and Validation.....	17
3.7 Domain of Execution	18
3.8 System Deliverables.....	18
3.8.1 <i>User Registration and Login</i>	18
3.8.2 <i>Lost and Found Item Reporting Module</i>	19
3.8.3 <i>Admin Dashboard</i>	19
3.8.4 <i>AI-Powered Matching Engine</i>	19
3.8.5 <i>Notification System</i>	19
3.8.6 <i>Administrator Module</i>	19
3.8.7 <i>Database Management</i>	19

3.8.8 Reporting and Analytics	19
3.8.9 User Guides and Documentation	19
Chapter 4: System Analysis, Design and Architecture	20
4.1 Introduction.....	20
4.2 System Analysis and Design.....	20
4.2.1 Use Case Diagram.....	20
4.2.2 Sequence Diagram	21
4.2.3 Entity Relation Diagram	22
4.2.4 Class Diagram.....	23
Chapter 5: System Implementation Testing	25
5.1 Introduction.....	25
5.1.1 Hardware Specification.....	25
5.1.2 Software Specification	25
5.2 Description of Testing.....	26
5.2.1 Unit Testing.....	26
5.2.2 Integration Testing	26
5.2.3 System Testing	27
5.3 Evaluate CLIP Model Performance	27
5.4 Test Data and Results.....	28
5.5 Description of Results.....	31
Chapter 6 Conclusion, Recommendation and Future Works.....	37
6.1 Introduction.....	37
6.2 Conclusion	37
6.3 Recommendations.....	37
6.4 Future Work	38
References.....	39
Appendix A: Time Schedule	41

Appendix B: User Guide and Management	41
Appendix C : Browser item	42
Appendix D: The Claiming section	43
Appendix E: GitHub Repository.....	44
Appendix F: System Settings.....	44
Appendix G: Turnitin Originality Report	45

List of Figures

Figure 2.1 Troov Home Page	7
Figure 2.2: MissingX Home Page.....	8
Figure 2.3: DeliverBack Home Page	8
Figure 2.4: Conceptual Framework	10
Figure 3.1: Prototyping model	12
Figure 4.1: Use case diagram.....	21
Figure 4.2: Sequence Diagram.....	22
Figure 5.1: CLIP Model Performance on CIFAR-10.....	28
Figure 5.2: Landing Home page	32
Figure 5.3: Login Page.....	32
Figure 5.4: Users Panel	33
Figure 5.5: Report Form	33
Figure 5.6: Admin Panel	34
Figure 5.7: Admin Viewing items section.....	34
Figure 5.8: Batch Match Report.....	35
Figure 5.9: Admin Viewing match items	35
Figure 5.10: Sample Report Generated by the Batch Match	36
Figure 5.11: Sample Report Generated by the system.....	36
Figure A.1: Time Schedule	41
Figure B.1: User Guide	41
Figure B.2: User Management.....	42
Figure C.1: Browse Items	42
Figure D.1: Claim Item	43
Figure E.1:GitHub Repository	44
Figure F.1: Settings the Model is ready	44
Figure F.2: The Virtual Environment and Uvicorn	45
Figure G.1: Turnitin Originality Report.....	50

List of Tables

Table 3.1: Tools and Techniques	17
Table 5.1: FR and NFR Testing results	31

Chapter 1: Introduction

1.1 Background

Lost and found management remains a crucial yet frequently underdeveloped aspect of operational logistics in large institutions such as universities, airports, shopping malls, and public transport hubs. These environments witness a high volume of foot traffic and movement, which naturally leads to a higher incidence of misplaced items such as phones, wallets, keys, identification cards, laptops, books, and bags (Tan, S. Y. & Chong, C. R., 2023). Traditionally, recovery efforts for lost items have relied on manual processes. Common techniques include physical notice boards, handwritten paper logs, or verbal reports submitted at designated help desks or security offices. Some institutions maintain simple Excel sheets or physical ledgers to keep track of reported items. Others may use emails or social media groups to notify the community. However, these methods are often fragmented, slow, and heavily dependent on human intervention and memory, resulting in limited tracking, poor coordination, and low recovery rates (Zhang, Y., Peng, S., 2018) .

The absence of a centralized, real-time digital platform contributes to inefficiencies, emotional distress, and financial loss for affected individuals. In environments where people are constantly moving such as during class transitions, boarding gates, or peak shopping hours the probability of lost items increasing is high, yet the likelihood of retrieval remains low due to lack of visibility and organized communication. Moreover, the absence of clear ethical guidelines or systems to encourage accountability often leads to dishonesty individuals who find items may keep them instead of reporting, either due to convenience or lack of a proper reporting channel. Promoting a culture of integrity, where found items are promptly reported rather than misappropriated, is essential to building a trustworthy and supportive community.

This situation underscores the need for a more reliable, efficient, and technologically advanced solution. The growing accessibility of web technologies and artificial intelligence presents a promising opportunity to digitize and automate the lost and found process reducing manual workload, improving tracking accuracy, and ensuring timely recovery of items.

1.2 Problem Statement

Despite the ubiquity of lost and found services in institutions such as universities, public transportation terminals, and commercial complexes, the recovery rate of lost items remains critically low due to inefficient and outdated systems (Wilson, D. C., 2023). Most existing

systems are manual and highly dependent on human memory or written logs, which are often incomplete, inaccurate, and inaccessible to the wider public. The absence of a centralized and intelligent tracking mechanism results in delayed reporting, poor item identification, and ultimately, permanent loss of valuable belongings. Moreover, users face difficulty in matching found items to their lost ones due to vague item descriptions and the lack of visual support (Ahmad, S., Ziaullah, M., Rauniyar, L., Su, M., & Zhang, Y., 2015).

This creates a significant gap in communication and trust between the person who finds an item and the rightful owner, often leading to unclaimed items piling up or being discarded. As technological advancements continue to improve data handling and visual processing, there is an urgent need to replace these fragmented manual processes with a more robust, automated, and intelligent system. Integrating artificial intelligence into lost and found management can significantly improve item classification, matching accuracy, and user engagement (Zhang, Y., Peng, S., 2018). The proposed AI-powered system aims to solve these issues by using text-to-image generation and smart item retrieval algorithms, creating a seamless and reliable process for both reporting and reclaiming lost items.

1.3 Aim

To develop a web-based, AI-powered lost and found management system that automates the reporting, intelligent matching, and retrieval of lost items within institutions. The system will utilize artificial intelligence for image generation from text descriptions and matching algorithms to compare reported lost items with found ones, thereby improving accuracy, reducing recovery time, and enhancing user experience.

1.4 Specific Objectives

- i. To investigate the existing Lost and found management systems in institutions.
- ii. To analyse the challenges experienced in the current system.
- iii. To design and develop a lost and found management system.
- iv. To test the developed system.

1.5 Research Questions

- i. What can be investigated about existing lost and found management systems in institutions?
- ii. What challenges can be analysed in the current lost and found processes?

- iii. What can be designed and developed to improve lost and found item management?
- iv. What features can be constructed to ensure a user-friendly interface for reporting, searching, and claiming lost items?
- v. What aspects of the developed system can be tested to ensure its effectiveness and usability?

1.6 Justification

The development of an AI-powered lost and found management system is essential in addressing the inefficiencies and delays in traditional lost item tracking methods within institutions. Current systems rely heavily on manual reporting and searching, which are time-consuming, error-prone, and unreliable. By integrating artificial intelligence, the proposed solution offers automated classification, image recognition, and intelligent matching of lost and found items, greatly enhancing accuracy and speed. This system ensures that lost items are reported, identified, and retrieved with minimal human intervention, reducing cases of permanently unclaimed belongings. The need for this project is heightened by the increasing student population and the frequency of misplaced items, which affects both students and staff. Implementing a smart solution not only improves operational efficiency but also fosters trust and accountability within the institution.

1.7 Scope and Limitation

The scope of this project is to design, develop, and implement a web-based AI-powered lost and found management system tailored for institutional use, specifically focusing on student environments. The system will allow users to report lost items, upload descriptions and images, and search or claim found items. It will incorporate AI features such as image-based item recognition and keyword-based search enhancements to streamline matching lost items with reported ones.

The project will not cover integration with physical surveillance systems (e.g., CCTV), external lost and found databases, or mobile app development at this stage. Limitations include dependency on the quality of images and descriptions provided by users, limited training data for the AI model which may reduce matching accuracy initially, and possible constraints in hosting or storage resources. Time and budgetary limitations may also affect the depth of AI features implemented in the prototype version.

Chapter 2: Literature Review

2.1 Introduction

This literature review examines lost and found management systems in institutions like schools and public facilities. It explores current practices, technologies, and common challenges faced by users and administrators. The review also assesses existing digital platforms, identifying gaps that impact efficiency and user experience. This analysis will guide the development of a more effective and user-friendly lost and found system.

2.2 Current Practices in Lost and Found Management Systems

Lost and found management in institutions typically involves the process of collecting, storing, documenting, and returning misplaced items to their rightful owners. Traditionally, many institutions such as schools, hospitals, airports, and offices have relied on manual procedures to handle lost and found cases. These include physical logbooks or basic spreadsheets where staff record details of lost or found items, such as item description, location, date, and the contact information of the person reporting or claiming the item. In many cases, these processes are slow, prone to human error, and lack transparency.(Liu et al., 2022) However, as technology continues to advance, some institutions have started using digital tools to streamline the management of these items. Current practices now include the use of online databases, mobile applications, web portals, and barcode or QR code systems to register and track items efficiently. Users can report lost items or check if their belongings have been found through self-service platforms.(Alnaghaimshi et al., 2020) Notifications and reminders are also sent to users when a possible match is identified. These digital systems not only reduce the burden on administrative staff but also improve the chances of successfully returning items. Despite the progress, adoption of these modern systems is still limited in many places due to budget constraints, lack of technical support, or resistance to change. As a result, institutions today often operate under a mixture of manual and digital methods, depending on their capacity and priorities.

2.2.1 Technologies Used in Lost and Found Systems

Modern lost and found systems are increasingly making use of various technologies to streamline the process of item reporting, storage, and recovery. (Tan & Chong, 2023)One of the most used tools is a database management system (DBMS), which allows institutions to store detailed information about each lost or found item, including the item's description, the date and place it was reported, and the contact information of the person reporting or claiming

the item. These databases are often connected to web-based platforms or mobile applications, enabling users to easily report lost items or check whether their belongings have been found, all in real-time.

Some institutions have adopted QR codes or barcodes to uniquely identify and track found items. When scanned, these codes instantly display the item's details from the system, helping reduce errors that come with manual entry. In larger or more advanced environments like airports or universities, RFID (Radio Frequency Identification) tags may also be used to automatically track valuable items without needing direct contact or line of sight.(Alnaghaimshi et al., 2020) Additionally, notification systems such as email and SMS alerts help to notify users when items matching their descriptions are found, which improves communication and reduces the time spent on follow-ups.

Cloud computing plays a key role in modern systems by enabling remote access, so both administrators and users can interact with the system from any location. (Gregory et al., 2020)Some platforms go further by using AI-powered matching algorithms that automatically compare new reports of lost items with found items, increasing the chances of identifying accurate matches. Others include image recognition features, allowing users to upload photos of their missing items so the system can search for visually similar items in the database(Ariyarathna, 2023; Liu et al., 2022).

These technologies not only enhance the speed and transparency of the lost and found process, but also improve user convenience,(Tan & Chong, 2023) reduce administrative work, and boost the chances of reconnecting owners with their belongings. However, the actual implementation of these technologies depends on the resources, technical ability, and priorities of the institution involved.

2.3 Challenges Experienced in Existing Lost and Found Systems

Despite technological advancements, many institutions still face significant challenges in managing lost and found systems effectively. These challenges often limit the ability to reunite lost items with their rightful owners in a timely and reliable manner. The two most common categories of issues include inefficiencies in the reporting and retrieval process, as well as poor user experience, accessibility, and communication limitations.

2.3.1 Inefficiencies in Item Reporting and Retrieval

In many institutions, the reporting of lost or found items remains largely manual or semi-automated. Staff may rely on handwritten logs, basic spreadsheets, or informal verbal reports. This leads to inconsistent data collection, loss of information, or difficulty in retrieving item details later. Additionally, there is often no standardized process for describing or categorizing items, making it difficult to perform accurate searches or match reports of lost and found items effectively.(Ariyaratna, 2023) Some systems do not update in real-time, which delays the matching process and causes confusion or duplication of efforts. Moreover, without automation, staff must manually sift through records to find matches, which is time-consuming and prone to human error.

2.3.2 User Experience, Accessibility, and Communication Limitations

Another major challenge is the poor user experience that many lost and found systems offer. Some platforms are not user-friendly,(Suchana et al., 2021) with complicated forms or limited language options that discourage people from reporting items. Others may not be accessible via mobile devices or for people with disabilities. In some cases, institutions do not offer an online option at all, forcing users to physically visit offices to report or claim items. This is especially inconvenient for large campuses or public facilities.

Communication gaps also play a big role in weakening the effectiveness of these systems. Users are often not notified when a potential match for their item is found, and there is no clear follow-up process.(Ariyaratna, 2023)Some systems lack automatic alerts or tracking numbers, leaving users uncertain about the status of their report. The lack of integration between departments or locations within an institution can further complicate communication and tracking (David, N. M., Catapang, G. M., et al., 2024). Without centralized access to information, valuable time is wasted, and the chance of recovering items decreases.

2.4 Review of Existing Systems

Various digital platforms have been developed to help streamline the lost and found process. These systems are designed to simplify item reporting, enhance communication between finders and owners, and increase the efficiency of item recovery. This section provides a review of three notable lost and found management systems.

2.4.1 Troov

Troov is a cloud-based lost and found management platform designed to assist organizations in efficiently handling lost item reports and claims (TROOV., 2025). Users can report lost or found items through predefined forms, providing details like location, type, and description. Troov's built-in algorithm then matches these reports with existing entries in the database to facilitate item recovery. The platform aims to reduce handling time and costs, enhancing the overall customer experience.

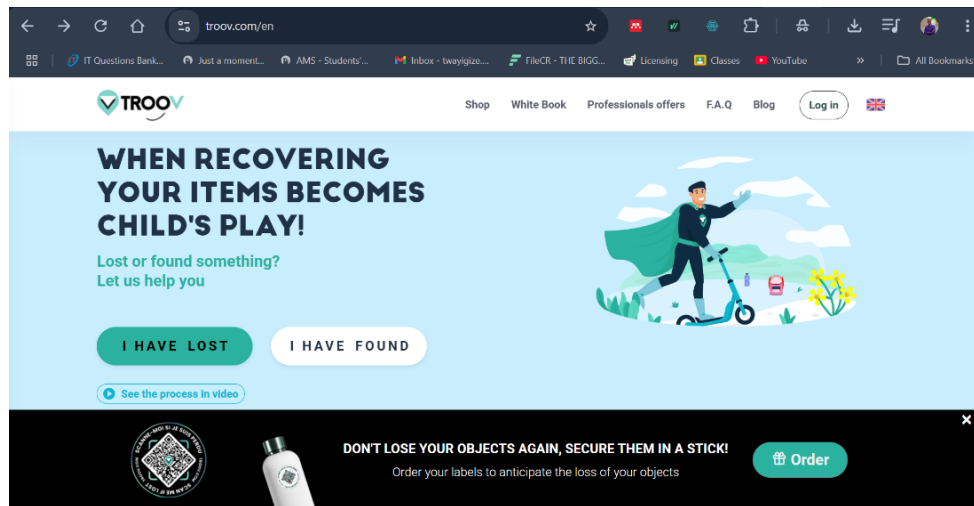


Figure 2.1 Troov Home Page

2.4.2 MissingX

MissingX is a cloud-based lost and found management platform widely used by organizations such as airports, transportation companies, and large institutions (MissingX, 2025). It provides a centralized system for logging, tracking, and returning lost items efficiently. MissingX allows users to submit lost and found reports online, including photos and detailed descriptions, which helps speed up the matching process. The system also includes automated notifications and reporting features, improving communication between the organization and the item owners. Its mobile-friendly interface enables easy access for both staff and users, making the retrieval process more convenient and transparent. MissingX's scalable design suits a variety of industries aiming to reduce lost item handling time and improve customer satisfaction.

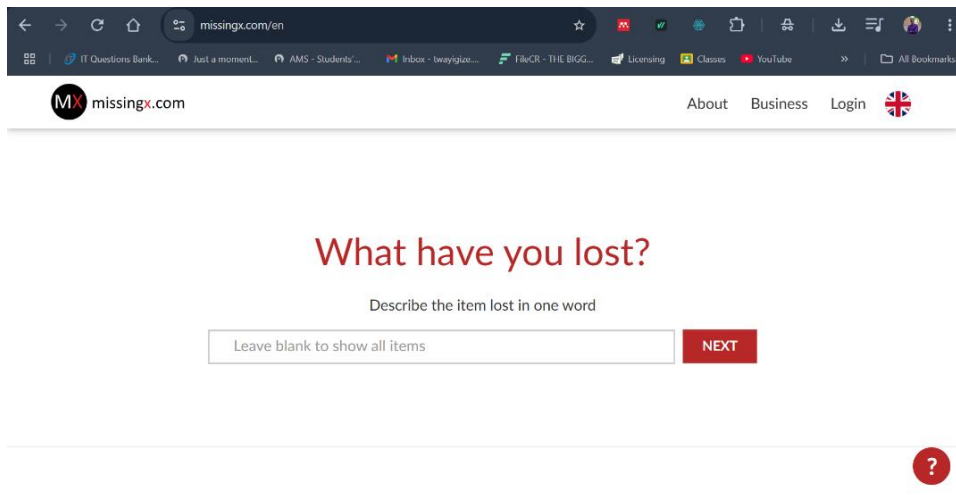


Figure 2.2: MissingX Home Page

2.4.3 Deliverback

Deliverback offers a digital lost and found solution tailored for the hospitality industry. The system allows staff to log found items with photos and detailed descriptions, track storage locations, and communicate with guests regarding their lost belongings (DeliverBack, 2025). Key features include automated notifications, secure online forms for guests to view item details and arrange returns, and integration with shipping services. Deliverback's approach simplifies the lost and found process, saving time for staff and improving guest satisfaction.

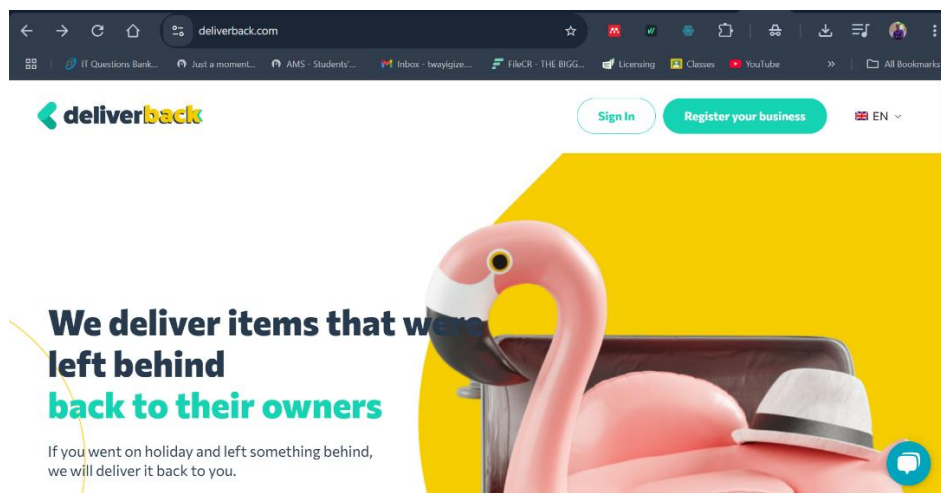


Figure 2.3: DeliverBack Home Page

2.4.4 Gaps in the existing system

Despite the availability of several digital platforms and technologies for lost and found management, significant gaps still exist in current systems particularly within institutional settings such as schools, universities, and public facilities. One major gap is the lack of system

integration, as most lost and found solutions operate in isolation and are not connected to other institutional databases, such as student or employee directories, which limits the ability to verify item ownership quickly.(Suchana et al., 2021) Another common issue is poor user experience, especially in institutions that continue to rely on paper-based or outdated digital systems. These often lack intuitive interfaces, making it difficult for users to report lost or found items accurately, and require users to manually check back for updates instead of providing real-time notifications or using smart matching algorithms. Accessibility also remains a concern, as many systems are not optimized for mobile use or inclusive for individuals with disabilities. Furthermore, institutions with limited technical expertise often struggle to customize or effectively implement existing platforms, leading to underutilization. Trust and transparency are additional challenges many systems do not allow users to track the progress of their reports or receive confirmations when items are matched, causing frustration and reducing confidence in the system. (Tan & Chong, 2023) A critical gap that AI-driven systems aim to address is the lack of intelligent item matching. Traditional systems rely on manual descriptions and basic keyword searches, which are time-consuming and error prone. An AI-powered solution can automatically generate visual representations of lost or found items from user-entered descriptions and employ smart matching algorithms to compare reports efficiently. This dramatically enhances the speed, accuracy, and success rate of item recovery. By closing this gap, AI introduces a transformative layer of automation, responsiveness, and intelligence that traditional systems lack. Altogether, these limitations highlight the pressing need for a more integrated, AI-driven, accessible, user-friendly, and transparent lost and found management system tailored to the specific needs of institutions.

2.5 Proposed System

The proposed system is a digital lost and found management platform tailored for institutions such as universities, schools, and public organizations, aiming to streamline the process of reporting, tracking, and recovering lost items through a user-friendly web. It addresses common limitations found in existing systems by integrating institutional databases (e.g., student or employee records) to help verify item ownership quickly and accurately. Users will be able to report lost or found items by entering key details such as item descriptions, photos, time, and location, and the system will use smart matching and filtering to identify potential matches automatically. Real-time notifications will alert users when a possible match is found, reducing the need for manual follow-ups and enhancing the chances of recovery. To improve accessibility, the platform will be mobile-friendly and designed to accommodate users with

disabilities, ensuring inclusivity. Additionally, users will be able to track the status of their reports, while administrators can oversee the process, verify ownership, and generate reports to monitor trends and performance. Security and privacy will be prioritized through secure login, role-based access, and data protection features. Overall, the system is designed to be an integrated, transparent, and user-centered solution that overcomes the inefficiencies, poor user experience, and lack of communication found in many traditional or outdated lost and found systems used in institutions today.

2.6 Conceptual framework

The conceptual framework for this study outlines the key components and relationships that guide the design and development of the proposed lost and found management system. Based on the findings from existing systems and the identified gaps, the framework focuses on enhancing efficiency, user experience, accessibility, and transparency within institutional settings such as universities.

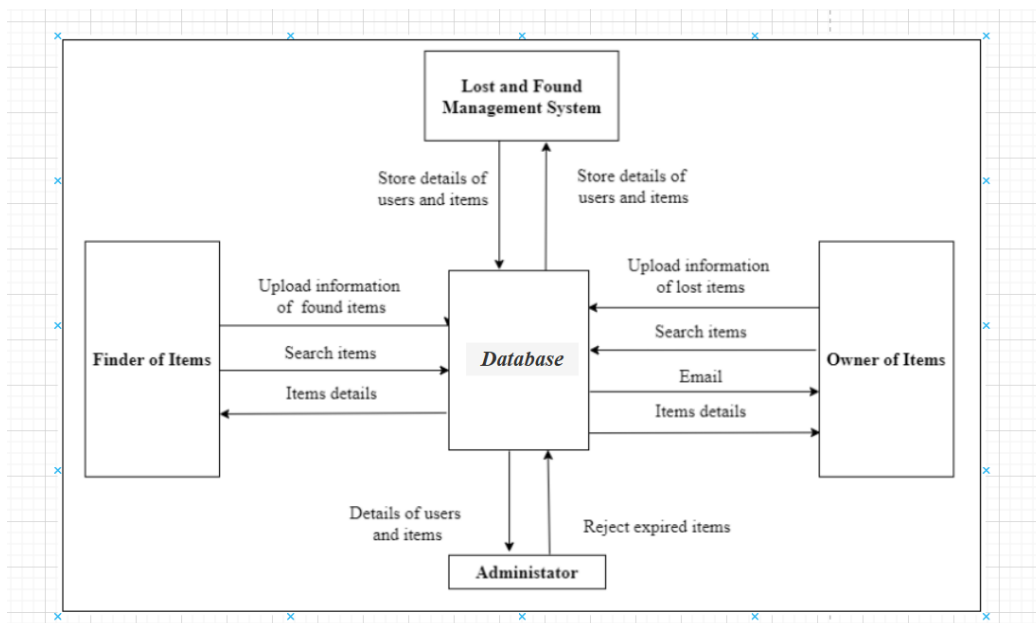


Figure 2.4: Conceptual Framework

Chapter 3: System Development Methodology

3.1 Introduction

A system development methodology serves as a structured guide for organizing, planning, and managing the creation of an information system. In system analysis and design, common approaches include Structured System Analysis and Design (SSAD) and Object-Oriented Analysis and Design (OOAD). OOAD focuses on modelling a system around real-world objects, their attributes, and interactions, making it ideal for projects that demand adaptability, easy maintenance, and scalability. For this Lost and Found Management System, the prototyping model has been selected as the development approach. This methodology emphasizes building an initial working version of the system, which is then refined through repeated feedback and adjustments. Given that the platform is web-based and heavily interactive, prototyping allows for continuous improvement based on real user input. Through this method, it becomes possible to quickly visualize system functionalities, identify necessary changes, and enhance usability throughout development. The result is a more accurate and user-aligned solution that effectively addresses the core needs of the system.

3.2 Prototyping Methodology

The system development for the Lost and Found Management System will adopt the prototyping model methodology. This model involves quickly building a working prototype of the system to illustrate design ideas and features, which is then tested and refined based on user feedback (Alnaghaimshi et al., 2020). This iterative process allows developers and users to collaboratively improve the system until a final version meets the desired requirements.

The prototyping methodology is especially suitable for this project because the requirements for an AI-powered solution may evolve as the system learns and adapts to real-world scenarios. Early prototypes will allow stakeholders to interact with the system and provide insights into the AI's performance in identifying and matching lost items, improving user experience and functionality over time.

The prototyping approach supports flexibility, continuous user involvement, and risk mitigation, as it helps identify and resolve challenges early in the development cycle. This is crucial for the lost and found system, which relies on accurate data input, efficient item matching, and responsive user interfaces.

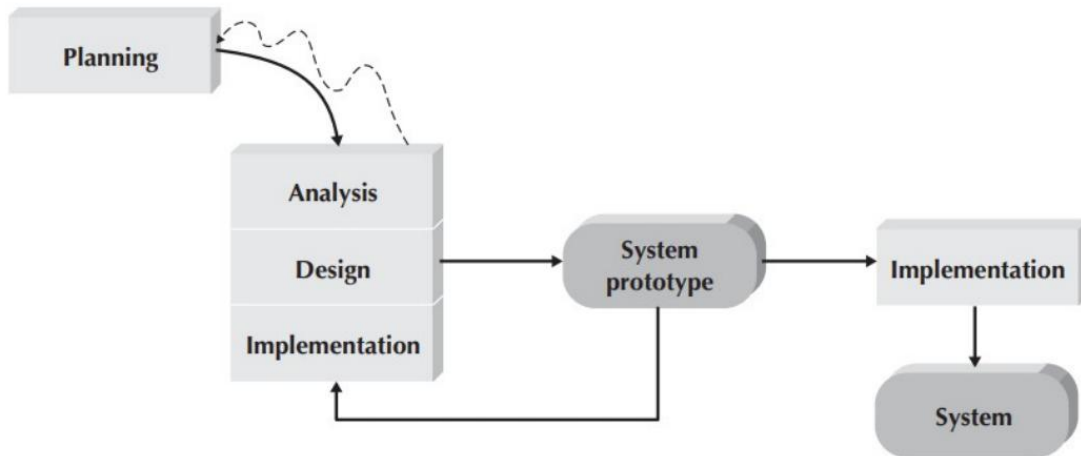


Figure 3.1: Prototyping model

3.2.1 System Planning

In the system planning phase, we will sit down with all the key stakeholders’ users who have lost items, those who found them, and the system administrators to clearly define what the lost and found system needs to achieve. We will outline the main goals such as fast and accurate item matching, user-friendly reporting, and reliable notifications. We will also identify possible challenges, like ensuring privacy and data security, and allocate resources such as the AI tools, development team, and necessary hardware. This planning sets a clear roadmap and realistic expectations for everyone involved, making sure the project stays focused and achievable.

3.2.2 Analysis

During the analysis phase, we will dive deeper into understanding the real needs of users and the environment where the system will operate. We will gather requirements by interviewing users and administrators, observing how lost and found issues currently happen, and identifying what data is needed to train the AI for accurate matching. This is also where we consider which AI models and technologies fit best, and what kind of data input (like images, descriptions, or location) the system should handle. This thorough understanding helps us build a system that truly solves users’ problems rather than just ticking boxes.

3.2.3 Design

In the design phase, we will translate the analysis insights into a detailed blueprint for the system. This includes designing the user interface to be simple and welcoming, so users feel comfortable reporting lost or found items. We will also architect the backend system, including AI components that analyse item descriptions and photos, and the database that securely stores

all data. The system design will be modular to allow easy future upgrades, such as adding new AI features or expanding to different locations. The goal is to build a system that is both powerful under the hood and easy to use on the surface.

3.2.4 Build Prototype

Here, we start turning designs into reality by creating a working prototype of the lost and found system. The prototype will include basic functions like submitting lost or found item reports, a simple AI matching engine, and notifications. It won't be perfect but will give users a tangible system to test and interact with early on. By building this prototype, we can quickly identify what works well and what needs improvement, saving time and effort before the full system development. We encourage users to explore the prototype and share honest feedback to guide us.

3.2.5 Testing

The functionality of the prototype is checked at this stage. The prototype will consist of the following tests to find out whether it will meet the specified goals.

- i. **Usability Tests:** This will test to see if users find the system intuitive and easy to navigate.
- ii. **Functionality Test:** This will verify that the AI correctly matches items and that notifications reach users promptly.
- iii. **Compatibility Tests:** This will test to make sure the system works well across different devices and browsers.

3.2.6 Implementation

After iterating on the prototype and making necessary improvements based on testing feedback, the final system will be developed and prepared for launch. We will ensure the AI models are well-trained and integrated seamlessly, the interface is polished, and all security measures are in place. Then, the system will be deployed for real users, accompanied by user guides and support channels to help people get started. Implementation isn't just about going live; it's about making sure the system helps people find their lost belongings and restores peace of mind.

3.3 Justification of the methodology

The prototyping methodology was chosen for this project because it aligns perfectly with the nature and goals of an AI-powered lost and found management system. This methodology

emphasizes building a working model early, allowing us to gather real user feedback quickly and make improvements iteratively. In a system that relies heavily on user interaction and intelligent item matching through AI, it's crucial to test and refine features like item submission, photo analysis, and AI recommendations with actual users before finalizing the full system. By using prototypes, we can better understand users' pain points such as difficulty in describing items, trust in the AI, or challenges in navigating the interface and address them early in development (Ramakrishnan, 2020). This approach not only speeds up the development process but also ensures that what we build is practical, user-friendly, and truly effective in reuniting people with their lost belongings. Ultimately, prototyping reduces the risk of building a system that looks good on paper but fails to deliver in real life.

3.4 System Analysis

This phase will define the architecture, modules, interfaces, and data required to meet the user and system requirements. The system will adopt Object-Oriented System Analysis and Design (OOAD) to ensure the solution is modular, scalable, and aligned with user needs. The goal is to break down the entire system into manageable components while visualizing how each part interacts with the others. The following diagrams and tools will be used to represent the system requirements:

- i. **Diagram:** This will illustrate how different users (such as the person who lost an item, the one who found it, and the administrator) interact with the system to accomplish tasks like reporting a lost/found item, reviewing matches, and resolving claims.
- ii. **Class Diagram:** This will be used to define the core structure of the database, including entities such as User, LostItem, FoundItem, MatchResult, and their relationships.
- iii. **Sequence Diagram:** This will outline the logical flow of events in key system processes.
- iv. **Database Schema:** This will describe how entities such as users, items, notifications, and matches are stored in the database, including primary and foreign keys to maintain data integrity and relational logic.
- v. **Wireframe:** A wireframe will be developed to provide a visual mock-up of the system's interface.

3.4.1 Functional Requirements

Functional requirements define the specific behaviours, actions, or tasks the system must perform. These are the core features that enable the system to achieve its intended purpose.

For the Lost and Found Reporting Web System, the following are the key functional requirements:

- i. User Registration and Login: Users must be able to create an account and securely log in to access full system features. Registration collects basic details like name, email, and phone number, which helps in managing user reports and communication.
- i. Report Lost Item: Users should be able to fill in a detailed form to report an item they have lost, including the item's name, description, last known location, date lost, and contact information.
- ii. Report Found Item with Image Upload: When someone finds an item, they should be able to submit a report with details and upload a clear photo of the item to help with visual identification.
- iii. Item Matching Algorithm: The system should automatically match found items with lost item reports based on keywords, categories, locations, and dates. A notification will be sent to the potential owner if there is a match.
- iv. View and Search Item Reports: Users should be able to browse and filter the list of reported lost or found items using filters like date, category, and location to help narrow down results.
- v. Admin Approval for Item Posting: All submitted reports (especially for found items) should go through an admin verification process before being visible to the public, to avoid spam or false reports.
- vi. Generate Claim Receipt: When a match is confirmed and the item is claimed, the system should generate a digital claim receipt for both the finder and the claimant for accountability.
- vii. Notification System: The system will actively notify users via email or SMS for important events such as successful registration, report approval, potential matches, and claim confirmations. This keeps users engaged and informed

3.4.2 Non-Functional Requirements

Non-functional requirements describe how the system performs its tasks, focusing on aspects like performance, security, usability, and reliability. These are crucial to ensure the system is robust and user-friendly.

The non-functional requirements for the system include:

- i. **Usability:** The interface should be simple, intuitive, and easy to navigate for all users, including those with minimal technical skills. A clean layout and helpful tooltips will guide the user through the system.
- ii. **Performance:** The system should respond quickly to user interactions and load pages within a reasonable time frame.
- iii. **Reliability and Availability:** The system should be accessible at least 99% of the time, with scheduled maintenance during off-peak hours.
- iv. **Security:** Data entered the system, especially personal contact information, should be protected. The system should be secure, ensuring access is restricted to authorized users only. Only authenticated users (via secure login) can submit or claim reports.
- v. **Scalability:** As the number of users grows, the system should be able to scale without needing a complete redesign—it includes scaling both the database and web server components.
- vi. **Maintainability:** The system should be modular and follow coding best practices, so it is easy to maintain, fix bugs, or add new features in the future.
- vii. **Audit Logging:** Every action performed on the system (e.g., submissions, deletions, claim approvals) should be logged for audit purposes and transparency.

3.5 System Development Tools and Techniques

This section outlines the tools and technologies that will be used to develop the Lost and Found Management Web System.

Tool/Technology	Purpose/Description
Visual Studio Code	An open-source text editor used for writing and editing source code efficiently across multiple languages

Laravel PHP	PHP framework used to develop robust backend features, manage routes, handle database operations, and ensure security
	Primary programming language used for backend logic and integrating AI functionalities like image classification.
MySQL	Relational database used to store and manage structured data such as user details, item reports, and matches.
HTML, CSS, JavaScript	Core web technologies used to structure (HTML), style (CSS), and add interactivity (JavaScript) to the web interface.
Bootstrap & React	Bootstrap ensures responsive design; React facilitates reusable UI components and dynamic user interface management
FastAPI	High-performance Python framework for developing RESTful APIs with fast execution and asynchronous support
Google Colab & Jupyter Notebook	Platforms for training and testing AI models with GPU support (Colab) and interactive data exploration (Jupyter)
Google Chrome	Used to test and validate the system's user interface with built-in tools for debugging and performance optimization.

Table 3.1: Tools and Techniques

3.6 Machine Learning Algorithms and Validation

The proposed system incorporates artificial intelligence through OpenAI's CLIP (Contrastive Language-Image Pretraining) model to enhance the accuracy and intelligence of item

matching. CLIP processes both images and text, enabling the system to compare lost-item descriptions with found-item images using a unified embedding space (Radford, A., Kim, J. W., et al., 2021).

i. Image Classification

The system uses the CLIP model to generate embeddings for both uploaded item images and user-submitted text descriptions. These embeddings are compared using cosine similarity to determine how closely a found item matches a reported lost item.

- **Algorithm:** OpenAI CLIP (ViT-L/14 model)
- **Input:** Lost item descriptions (text) and found item images
- **Output:** Similarity score indicating match confidence
- **Evaluation Method:** Hold-out validation using test subsets for both text and image pairs
- **Evaluation Metrics:** Accuracy, Precision, Recall, F1-score

ii. Similarity Search (Text-Based Matching)

CLIP eliminates the need for separate CNN and SBERT models because it simultaneously performs image encoding and text encoding in a shared semantic space, improving matching efficiency and reducing system complexity.

3.7 Domain of Execution

The domain of execution for the project is web-based and accessible through browsers on various devices including desktops, laptops, tablets, and mobile devices. This choice allows for greater accessibility and convenience, as users can access the system from anywhere with an internet connection. Additionally, a web-based approach ensures platform independence and facilitates easy updates and maintenance of the system. (Gregory et al., 2020)

3.8 System Deliverables

3.8.1 User Registration and Login

Allows users to securely create accounts and log in to report or search for lost and found items.

3.8.2 Lost and Found Item Reporting Module

This is the core functionality of the system. Users will be able to report lost items by providing details such as item name, description, date lost, and location. Similarly, found items can be reported along with descriptions and image uploads to improve identification. Each report will be stored and indexed for easy search and matching.

3.8.3 Admin Dashboard

An admin module will be included to facilitate oversight and management of the platform. Administrators will be able to verify item reports before publication, manage user activity, resolve conflicts, handle claims, and generate logs for accountability.

3.8.4 AI-Powered Matching Engine

Matches lost and found items based on images and descriptions using AI algorithms - CLIP Model.

3.8.5 Notification System

Notifies users when potential matches for lost or found items are identified.

3.8.6 Administrator Module

Enables system administrators to manage users, items, and oversee system operations.

3.8.7 Database Management

A well-structured MySQL database will be used to store user information, lost and found reports, matched items, claim records, and system logs. Manages storage, retrieval, and integrity of all system data.

3.8.8 Reporting and Analytics

Generates reports on system usage, match rates, and other performance metrics. To help administrators monitor platform usage and trends, the system will include a reporting module.

3.8.9 User Guides and Documentation

Provides instructions and support material for users and administrators.

Chapter 4: System Analysis, Design and Architecture

4.1 Introduction

This chapter presents the analysis, design, and architecture of the AI-Driven Lost and Found Management System. It outlines the system requirements, user interactions, data flow, and structural components that define how the system works. To illustrate these aspects, the chapter uses key design diagrams such as the use case diagram, sequence diagram, ERD, and class diagram, which show how users interact with the system, how processes flow, and how data is organized internally.

4.2 System Analysis and Design

The system analysis identifies the challenges in traditional lost-and-found processes and outlines the functional and non-functional requirements needed to address them. Based on this analysis, the system design defines how the solution will operate, how users will interact with it, and how data will be organized(Mahadik et al., 2023).

This section presents the main design diagrams: the use case diagram shows system functions and user interactions, the sequence diagram illustrates the flow of activities, the ERD outlines the database structure, and the class diagram represents the system's objects, attributes, and relationships(Gregory et al., 2020).

4.2.1 Use Case Diagram

The use case diagram below highlights the functional requirements of the system shows how the admin and user interact with the system and what is needed to access the current system. To clarify and organize the system's requirements by describing "who" can do "what" with the system, the Use Case Diagram is provided.

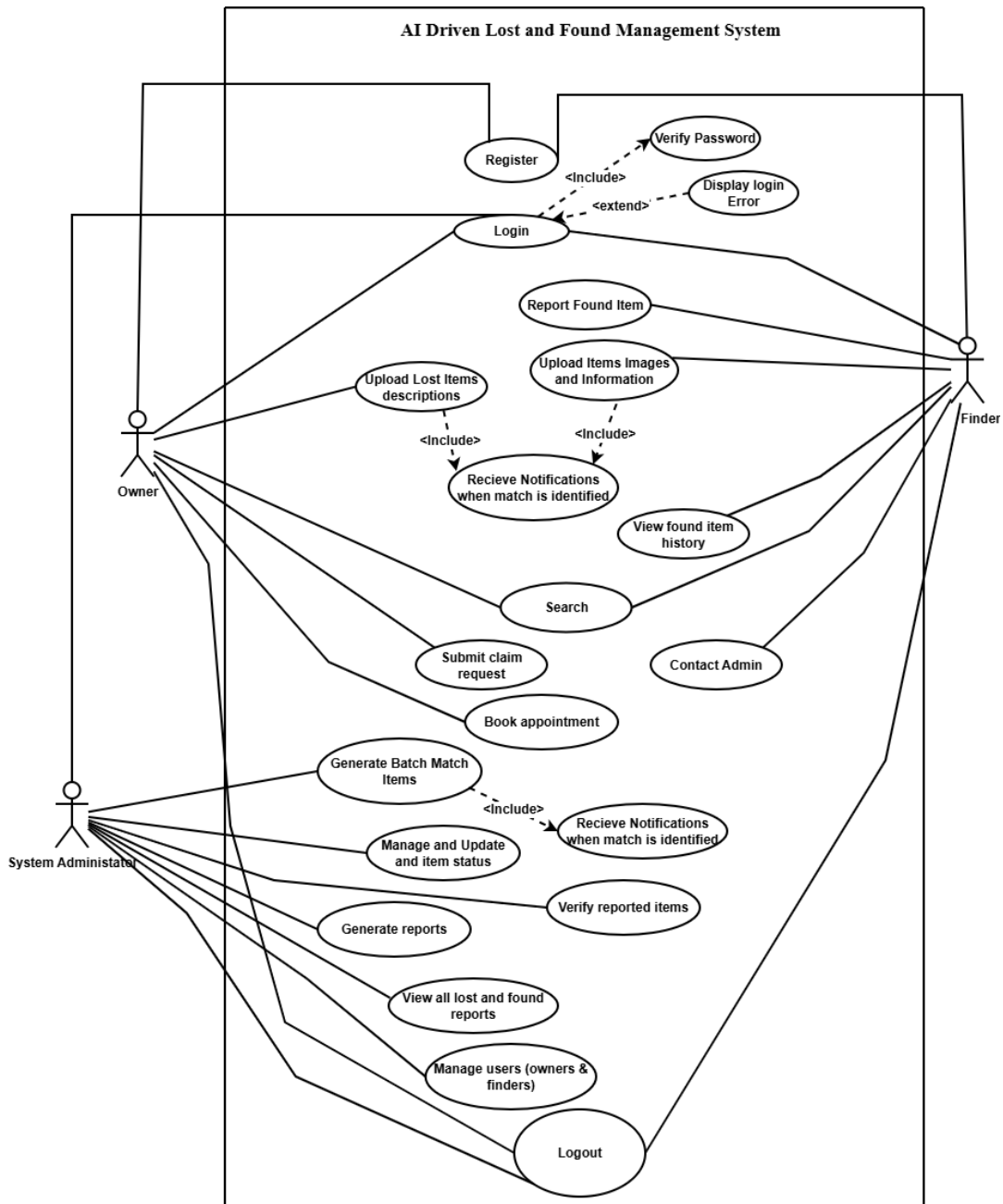


Figure 4.1: Use case diagram

4.2.2 Sequence Diagram

Sequence Diagrams are interaction diagrams that show how system operations are performed. They illustrate the flow of messages between system components in a time-ordered manner using a vertical time axis. In the Lost and Found System, the user first logs into the platform and gains access to the dashboard. They can then report a lost item or submit a found item, and the system stores the details in the database. The admin reviews the submissions, updates the item status, and once a match is confirmed, the system notifies the user about the recovered item.

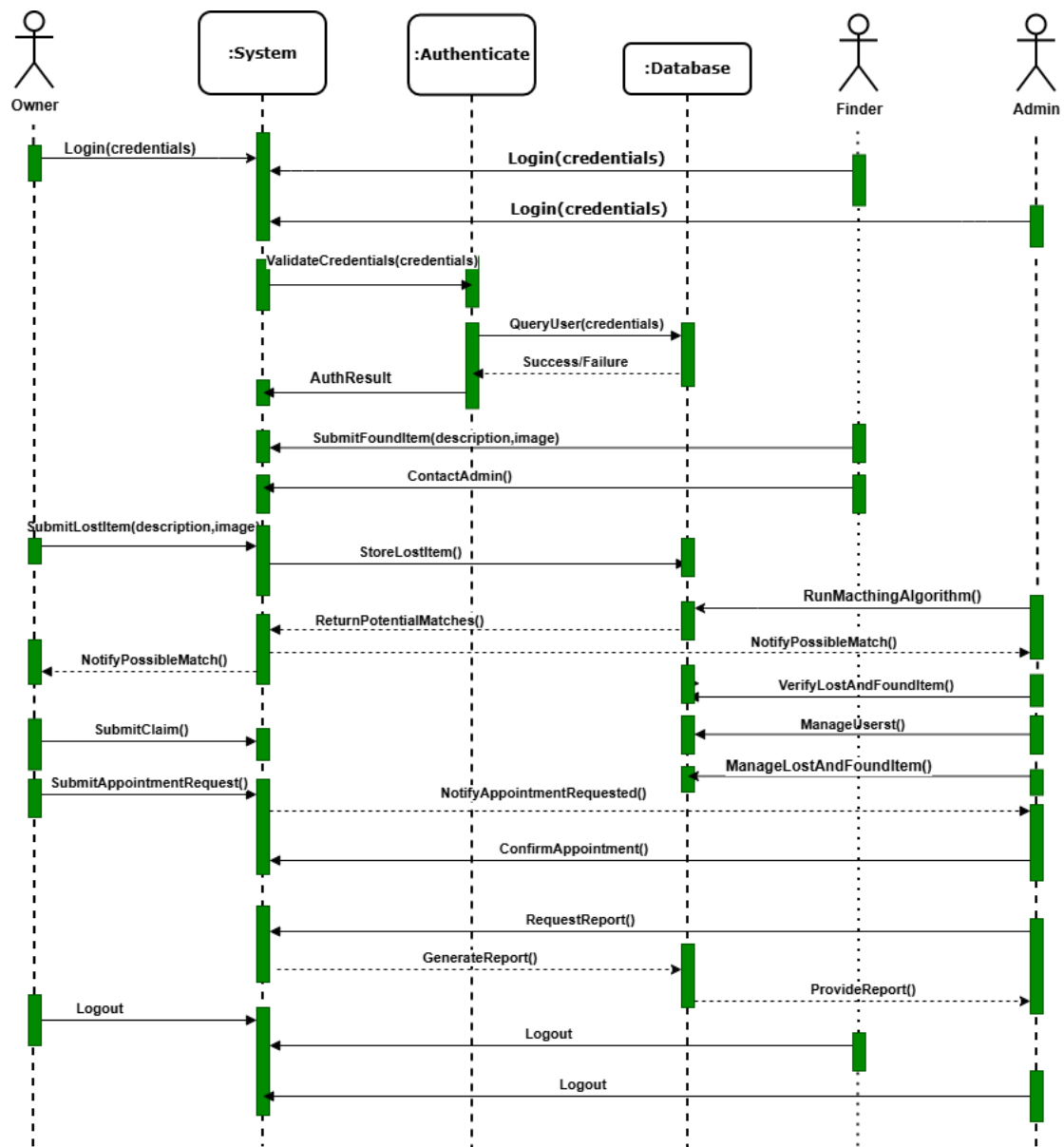


Figure 4.2: Sequence Diagram

4.2.3 Entity Relation Diagram

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology (IT) system

For the AI-Driven Lost and Found Management System, the ERD employs the Crow's Foot notation (also known as Information Engineering notation). This notation is chosen for its

superior readability and intuitive depiction of cardinality (one-to-many, many-to-many), simplifying the comprehension of the complex relationships required by the matching process.

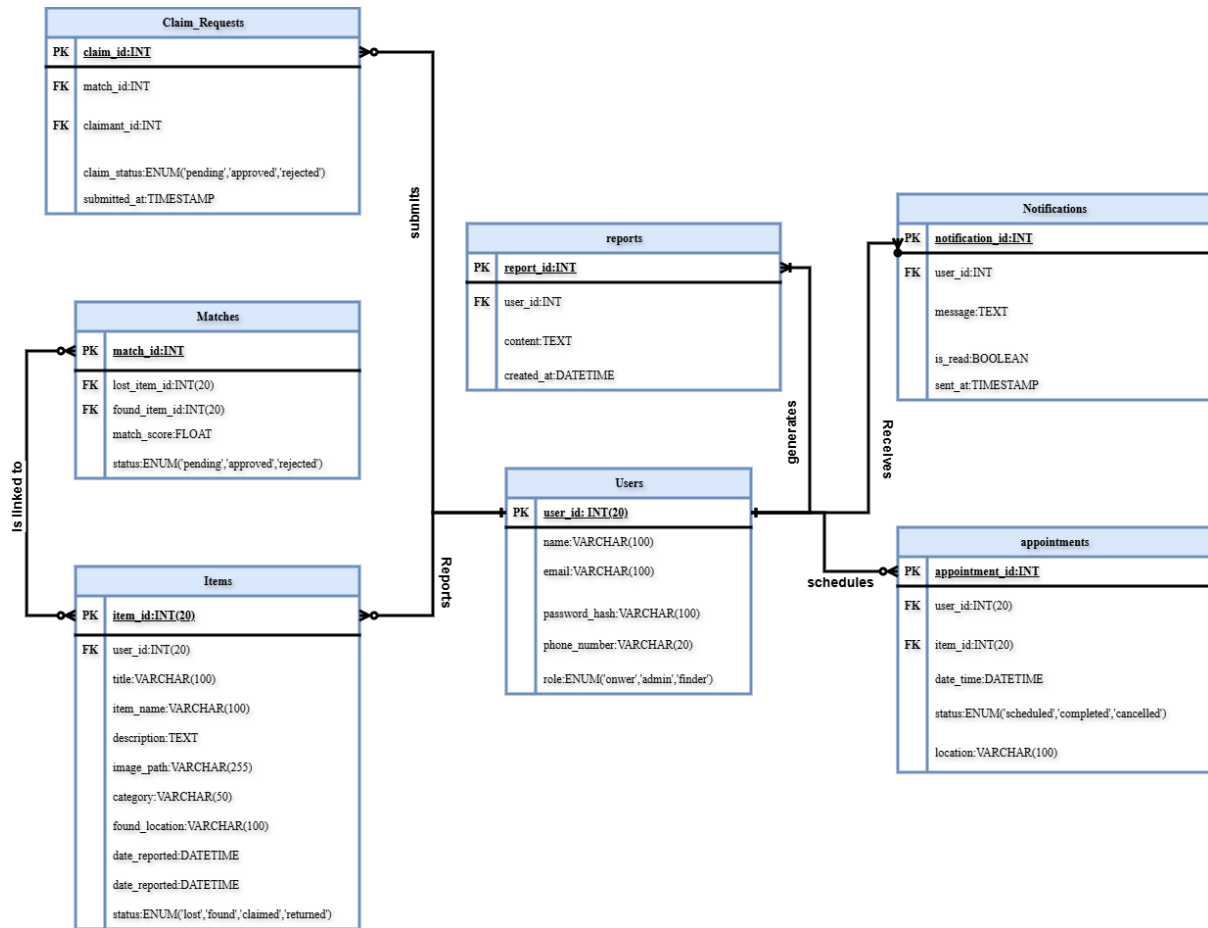


Figure 4.3: ERD diagram - CROW'S FOOT notation

4.2.4 Class Diagram

This diagram depicts the main types of users and other objects involved in the system, represents their attributes(characteristics) and their behaviours in form of classes and clearly shows the relationships between the different objects (inheritance, composition, etc.)

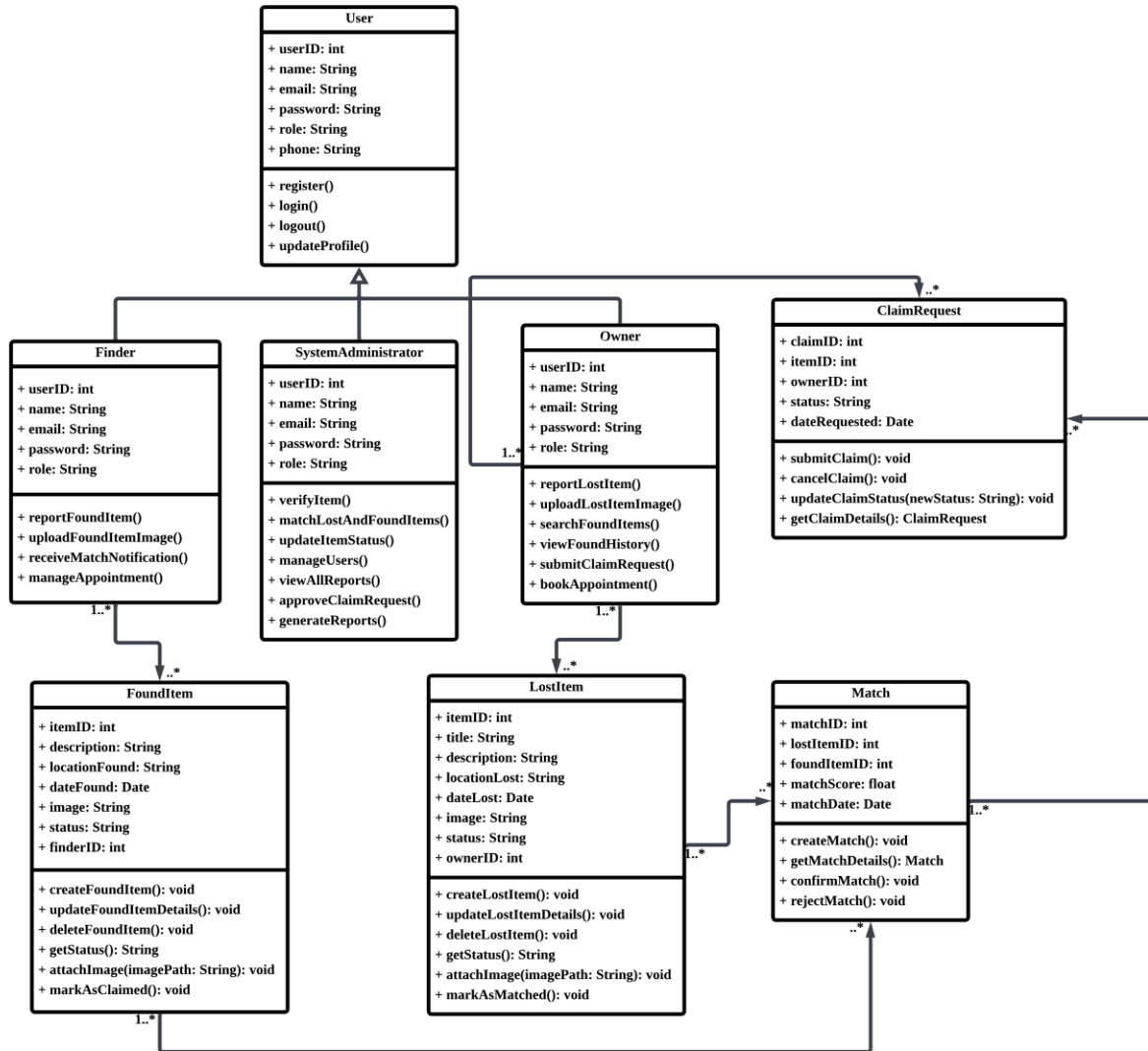


Figure 4.4: Class Diagram

Chapter 5: System Implementation Testing

5.1 Introduction

This chapter presents the implementation and testing activities carried out for the AI-Driven Lost and Found Management System developed for institutional use. It explains the hardware and software environment used during development, the implementation approach for both the Laravel and Python components, and the different testing techniques applied to ensure the system operates efficiently, accurately, and securely. The chapter also highlights the results obtained during system operation, including the performance of the CLIP-based AI matching model (Khandelwal, T., & Poria, S., 2022).

5.1.1 Hardware Specification

The system was implemented using a computer running Windows 11 and Windows 10 during various stages of development. The main development machine was equipped with an Intel Core i7 processor (8 cores, 2.71 GHz base clock), 8 GB of RAM, and a 512 GB SSD. The machine had sufficient RAM to support both the Laravel application, and the Python virtual environment used to run the CLIP model for the AI matching service. Mobile devices running Android and iOS were also used for testing the user interface components.

5.1.2 Software Specification

The system relied on several software tools, frameworks, and environments throughout its development lifecycle. The core application was developed using PHP Laravel (version 8.2.12), which provided the structure for backend logic, routing, authentication, and database interactions. Visual Studio Code served as the primary integrated development environment (IDE), offering extensions for Laravel, PHP, Blade templates, database management, and Git integration.

The AI Services of the system was implemented using Python (version 3.13.9). A dedicated virtual environment (**venv**) was created to manage dependencies for the CLIP image–text matching model. The model was downloaded from Hugging Face which is an open-source platform and community for machine learning (ML) and artificial intelligence (AI) that democratizes AI by providing tools, a model hub, datasets, and "Spaces" for collaboration and tested using Google Colab before being integrated locally. Uvicorn was used as the ASGI (Asynchronous Server Gateway Interface) server to host the Python-based AI service and allow it to communicate with the main Laravel system using RESTful API endpoints. The CLIP

model documentation from Hugging Face served as the reference for implementation and parameter tuning.

CLIP Model Documentation : https://huggingface.co/docs/transformers/model_doc/clip

CLIP Model: <https://huggingface.co/openai/clip-vit-large-patch14>

Web browsers played an important role in system testing, with Google Chrome being the primary browser used for evaluation. Additional tests were performed using Mozilla Firefox and Microsoft Edge to confirm cross-browser stability. The database layer was supported by MySQL, which worked seamlessly with Laravel's Eloquent ORM to manage lost items, found items, claims, and user logs.

5.2 Description of Testing

Testing of the system followed a black-box approach, where all functions were evaluated based on requirements without inspecting internal code. This was combined with unit, integration, and system testing to ensure each module performed correctly and interacted smoothly with other components.

5.2.1 Unit Testing

Each module was tested independently during development. The item reporting module was validated to confirm that users could enter complete lost or found item details, including images, descriptions, locations, and dates. The claim submission module was tested by examining how claims were processed, stored, and assigned statuses. The admin approval module was evaluated to ensure that administrators could review, approve, reject, and update claims without errors. The CLIP image-text matching module running on the Python server was tested to verify that uploaded images and descriptions were correctly received and processed, and that the similarity results returned were accurate. The notification process was also tested to confirm that status changes triggered notifications properly (Ramakrishnan, 2020).

5.2.2 Integration Testing

Integration testing ensured that Laravel components and the Python AI service communicated seamlessly. Lost or found item submissions were checked to confirm that they were not only stored in the database but also sent to the AI server when required. The system was tested by submitting item reports, sending them to the CLIP server through an API request, and retrieving the confidence scores generated by the model. Integration testing also covered the transition

between the user dashboards, admin dashboards, and the claim review workflows. It confirmed that status changes, updates, and verification steps propagated correctly throughout the system.

5.2.3 System Testing

During system testing, the entire platform was evaluated in a real-world scenario. This included verifying authentication for all user roles finder, claimant, and admin and ensuring that each role accessed appropriate functions. The system was tested extensively for responsiveness, page loading, database interactions, and end-to-end workflows, from submitting a lost item to performing AI-based matching and approving claims. The reporting module, including CSV generation, was also fully tested to confirm accuracy and completeness.

5.3 Evaluate CLIP Model Performance

The CLIP model was evaluated on a subset of 500 images from the CIFAR-10 dataset to measure its image recognition performance. The model used is OpenAI/clip-vit-large-patch14, loaded via the Transformers library, and images were compared against text descriptions for each class. The evaluation was conducted in Google Colab using PyTorch, with standard metrics such as accuracy, precision, recall, and F1-score (Wang, X., & Gupta, A., 2015).

The results show that the CLIP model achieved an accuracy of 95.20%, with an average precision of 95.43%, recall of 95.35%, and an F1-score of 95.28%. These results indicate that the model can reliably match images to text descriptions, even on a relatively small test sample.

This evaluation confirms that CLIP is effective for image-to-text matching and can be integrated into the system to provide image-based search and recommendation services. The high performance on the CIFAR-10 dataset demonstrates its suitability for practical applications within the web application.

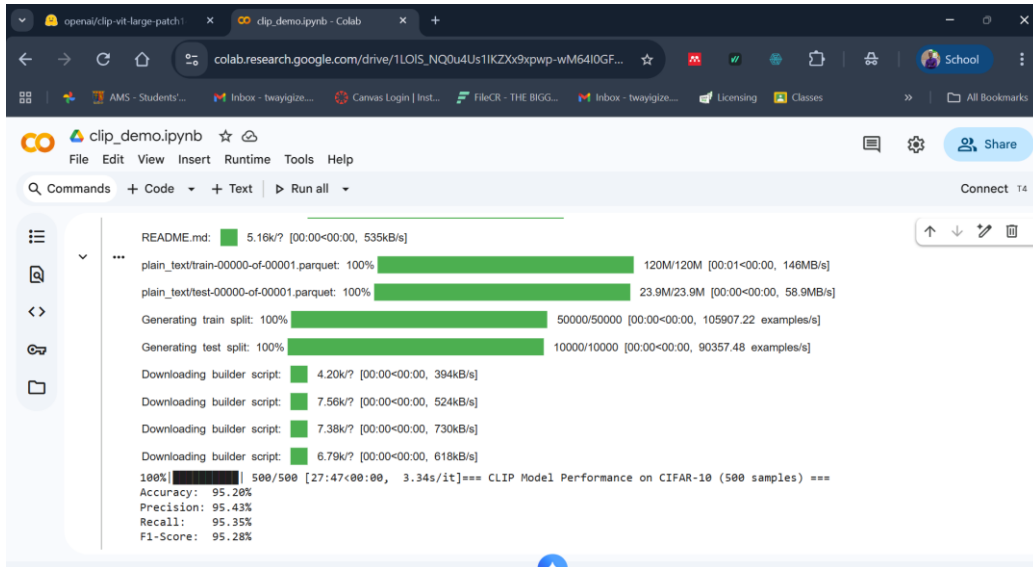


Figure 5.1: CLIP Model Performance on CIFAR-10

5.4 Test Data and Results

Functional Requirements				
	Expected Results	Actual Results	Status	Remarks
Universal Authentication (All Users)	The system should allow all the users (Finder, Owner, and the admin) to be registered and log in	<p>Users of the system can register using their name, email, Password.</p> <p>System validates credentials and redirects to role-specific dashboards</p> <p>Admin_Email : twayigize.niyitegeka@strathmore.edu</p> <p>Admin_Password : *****</p> <p>The user of the system, that is the Finder/Owner, should be able to sign into the system.</p> <p>First name: Twayigize</p> <p>Last name: Bienvenu</p>	Success	Good performance

		Finder/Owner_Email: bienvenutwayigize@gmail.com Finder/Owner_Password : *****		
Claimant / Owner Module	Submit lost item report with details + image (Optional). View claim status and matches. Receive notifications on match/approval.	Successfully records details and uploads images. Status updates tracked accurately. Notifications delivered instantly.	Success	Good performance
Finder Module	Report found item with photo + category. Receive match alerts.	Data stored correctly, CLIP processes image. Alerts triggered properly when owner claims.	Success	Good performance
Admin Module	View and manage all lost/found items and review claims. Approve/Reject claims. Manage users, Run the Matching algorithm, Generate downloadable	Dashboard loads all items correctly. Decisions saved and reflected instantly. System exports clean PDF Report.	Success	Good performance

	CSV/PDF reports.			
AI Matching Module (CLIP)	Match images to text descriptions with confidence scores. Batch matching multiple items.	CLIP returns ranked matches with scores. Batch module processed items quickly.	Success	Good performance
Non-Functional Requirements				
System Security	The system should be able only authorized users to access it; each user should have a unique password and email	Only authorized users can access the system with unique login details. Secure login + role-based access confirmed.	Success	Good performance
Performance	The system should be able to generate reports, allow updates and keep track of all the records at the ease of the admin.	The system keeps track of records that are accessed by the admin. The system allows regular updates of the stored information.	Success	Good performance
Scalability	Should handle increased users/items.	Database + server scaled well in tests.	Success	Good performance

Usability	Simple, intuitive design	Users navigated easily.	Success	Good performance
Efficiency	The system should be available if there is no internet connection. The time to restart after a failure should be minimal as possible.	The system reproduces desirable outputs with minimal waste, timesaving. The system can be easily accessed without an internet connection	Success	Good performance

Table 5.1: FR and NFR Testing results

5.5 Description of Results

5.5.1 Claimant (Owner) Module

Claimants were able to create accounts, log in, and submit detailed reports for lost items. The system successfully stored descriptions, dates, locations, and uploaded images. After submission, claimants could track the status of their claims from the dashboard, receive updates when matches were found, and follow the review process as the admin approved or rejected their claims.

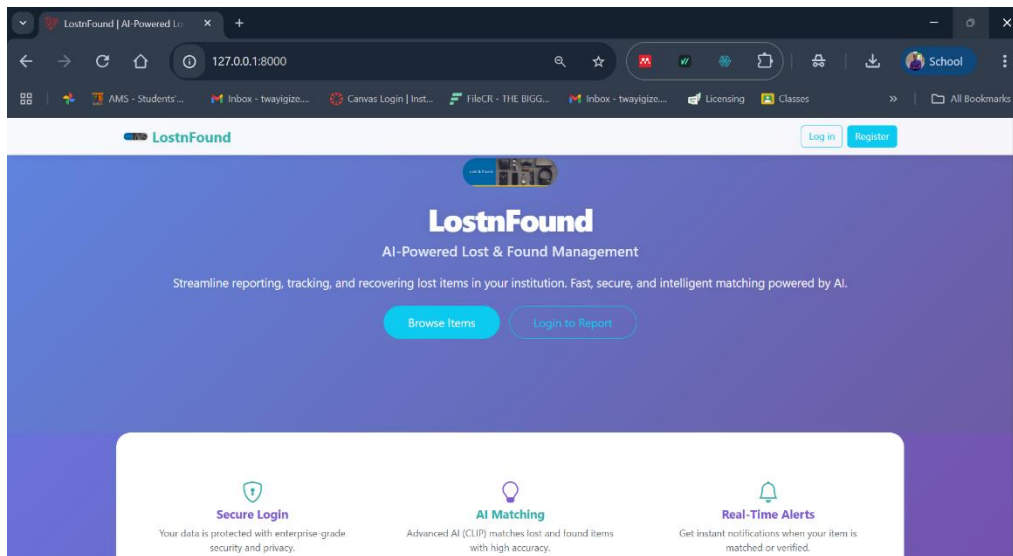


Figure 5.2: Landing Home page

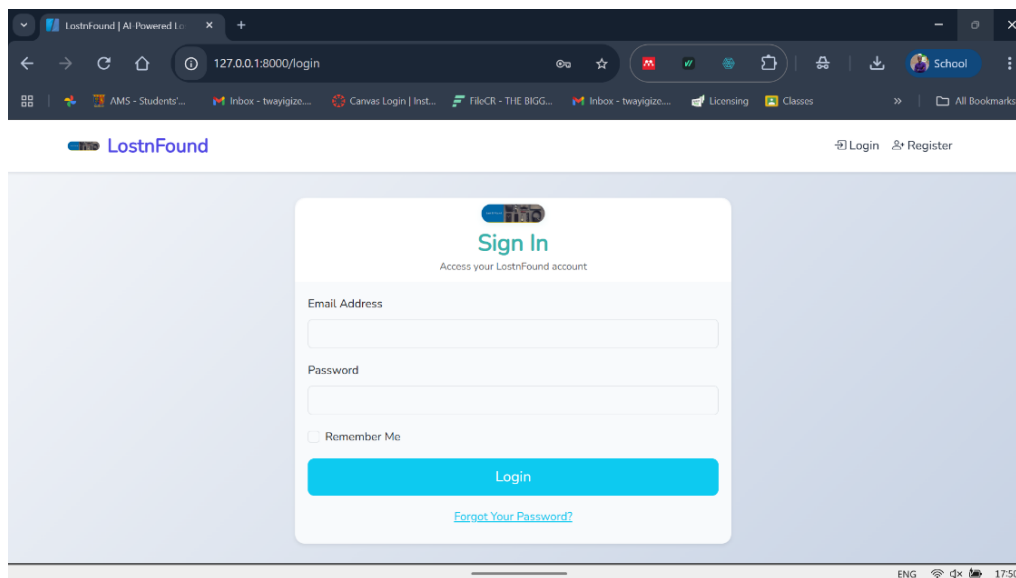


Figure 5.3: Login Page

5.5.2 Finder Module

Finders could easily report items they found by describing the object and uploading clear images. Once submitted, the items were processed by the CLIP AI model to determine whether they matched any reported lost items. Finders received updates when their submissions led to successful matches.

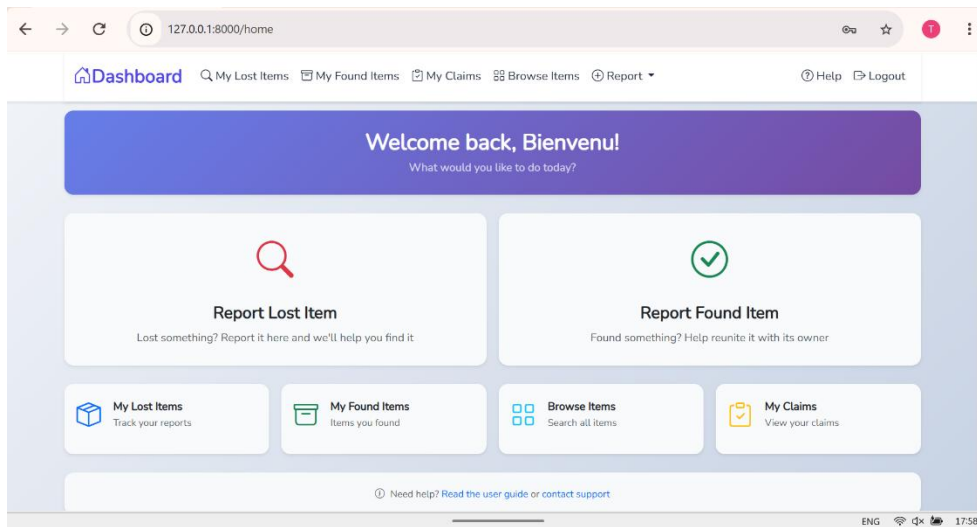


Figure 5.4: Users Panel

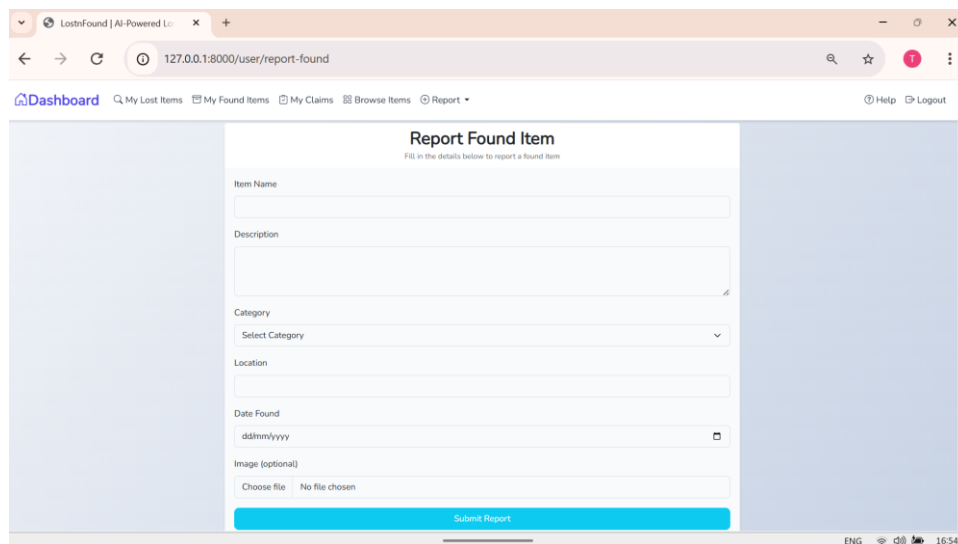


Figure 5.5: Report Form

5.5.3 Admin Module

Administrators accessed a comprehensive dashboard that displayed all reported lost and found items. From this dashboard, they reviewed submissions, verified owner claims, approved or rejected requests, and managed user accounts. Admins could also view AI match logs, assign item statuses, and generate downloadable CSV reports summarizing system activity.

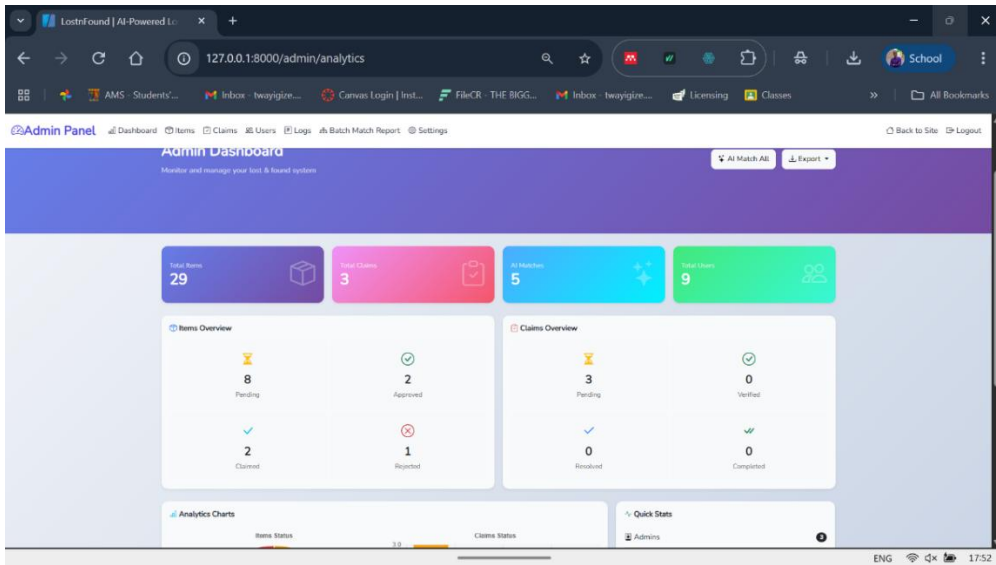


Figure 5.6: Admin Panel

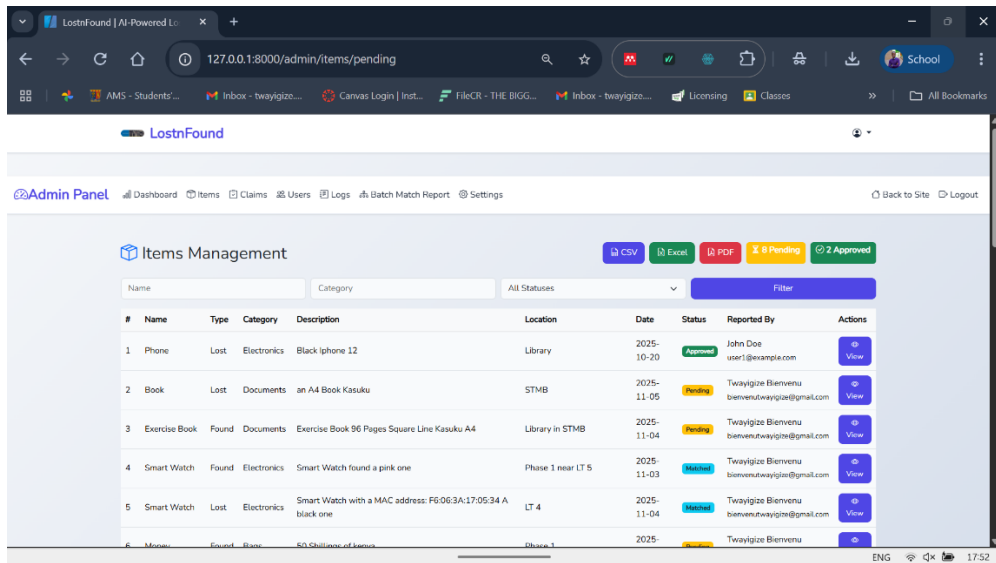


Figure 5.7: Admin Viewing items section

5.5.4 AI Matching Module (CLIP)

The CLIP model, deployed inside a Python virtual environment and served through Uvicorn, successfully processed both single-item and batch matching tasks. Images and descriptions were encoded and compared to produce similarity confidence scores. These scores were then used to assist administrators in making final verification decisions. The model demonstrated reliable performance and acceptable inference speed.

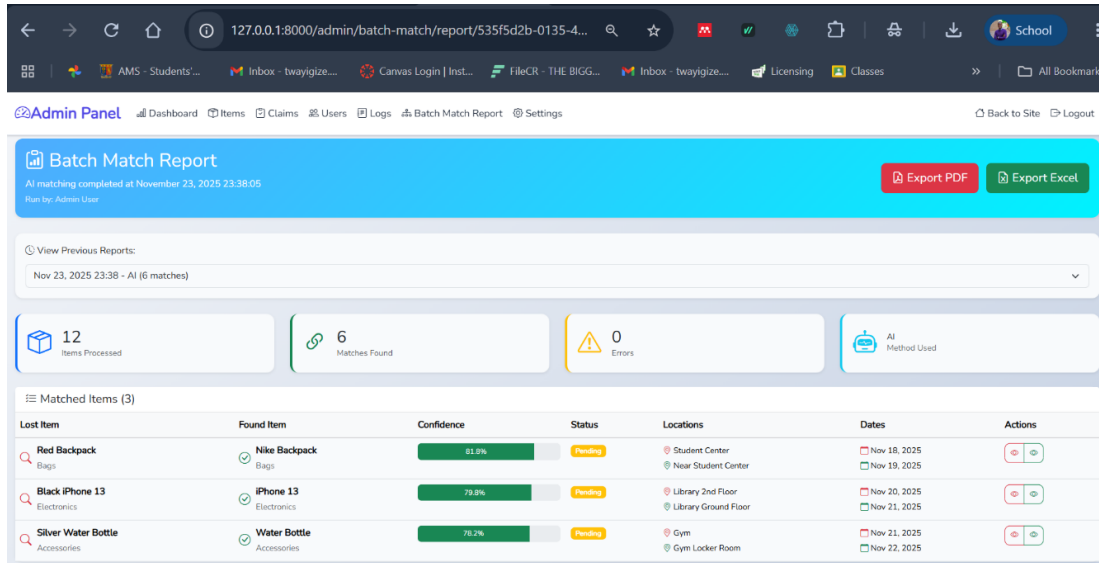


Figure 5.8: Batch Match Report

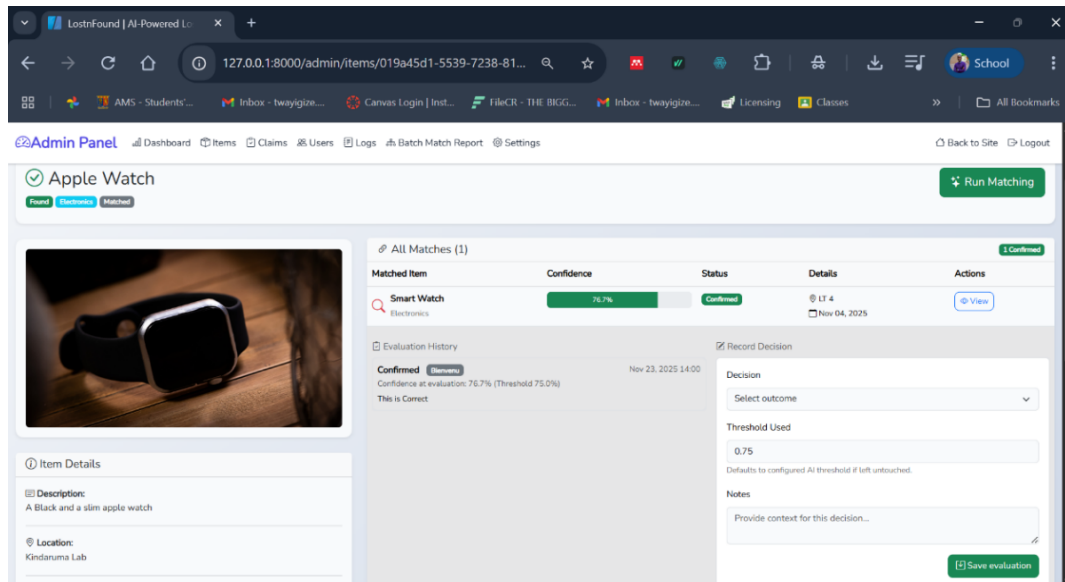


Figure 5.9: Admin Viewing match items

5.4.5 Reporting Module

The system automatically generated detailed reports that included claim histories, lists of all lost and found items, match logs, and administrative actions. Users could download CSV reports, and admins had access to complete system-wide summaries that supported decision-making and record-keeping.

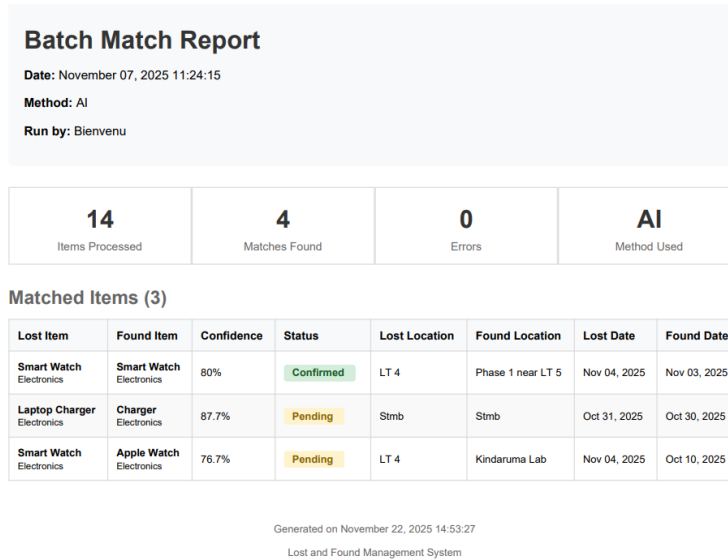


Figure 5.10: Sample Report Generated by the Batch Match

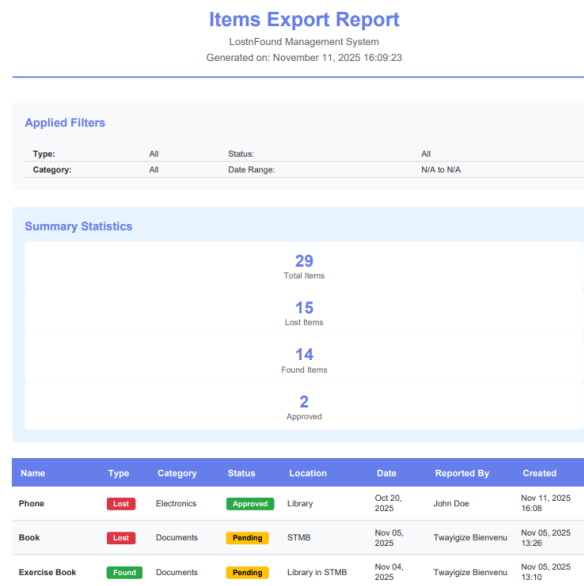


Figure 5.11: Sample Report Generated by the system

Chapter 6 Conclusion, Recommendation and Future Works

6.1 Introduction

This chapter provides a summary of the project's findings based on the objectives outlined in Chapter 1. It presents the overall conclusion drawn from the development of the AI-Driven Lost and Found Management System. In addition, it highlights key recommendations for effective system deployment and use within institutions. Finally, it outlines possible future enhancements that can be implemented to expand the system's capability beyond its current scope.

6.2 Conclusion

The AI-Driven Lost and Found Management System successfully addresses the challenges caused by traditional manual lost-and-found processes, which are often slow, disorganized, and prone to errors. By integrating a CLIP-based image-text matching model, the system significantly improves the accuracy and speed of identifying potential matches between lost and found items.

The system enhances efficiency by automating item reporting, claim verification, and match detection while centralizing all lost/found records within a structured database. Claimants, finders, and administrators benefit from a streamlined workflow that reduces delays, improves transparency, and minimizes the chances of items remaining unclaimed.

Overall, the system achieves its goal of providing an intelligent, accessible, and user-friendly solution for institutions seeking to modernize their lost-and-found operations.

6.3 Recommendations

For optimal use and performance of the system, the following recommendations are made:

Institutions implementing the system should have stable internet access to support real-time reporting, image uploads, and communication with the AI matching service. Staff responsible for managing the lost-and-found office should receive basic digital skills training to effectively handle the admin module. Users (students, visitors, and staff) should be encouraged to upload clear images and accurate descriptions to improve the quality of matches produced by the AI model. It is also recommended that institutions periodically update the Python virtual environment, Laravel framework, and system dependencies to maintain security and compatibility.

6.4 Future Work

Although the system meets its core requirements within the given timeframe, there are several areas where it can be expanded:

Future versions may include real-time notifications through SMS or WhatsApp integration for faster communication. Additional AI capabilities such as automatic image enhancement, object-type detection, or multilingual support could further improve usability. The system can also be extended to support multi-institution databases, allowing lost items from one campus to be matched with those from another. Lastly, the integration of RFID or QR-code tagging may improve traceability for institutions that handle high-value or frequently misplaced items.

References

- David, N. M., Catapang, G. M., et al. (2024). A comparative study on lost and found management systems in academic institutions. Proceedings of the 6th International Computing Conference.
- Ahmad, S., Ziaullah, M., Rauniyar, L., Su, M., & Zhang, Y. (2015). How does matter lost and misplace items issue and its technological solutions in 2015-a review study. IOSR J. Bus. Manag. Ver. I, 17(4), 2319-7668.
- Alnaghaimshi, A., Alshammari, O., & Alharbi, A. (2020). Web-based lost and found system using iterative prototyping. International Journal of Computer Applications.
- David, N. M., Catapang, G. M., et al. (2024). *A comparative study on lost and found management systems in academic institutions: Assessing reliability across Universiti Tunku Abdul Rahman, California State Polytechnic University, and Nazareth School of National University. In 2024 6th International C.*
- DeliverBack. (2025). We deliver Items that were left behind, Retrieved from <https://deliverback.com/>.
- Khandelwal, T., & Poria, S. (2022). Multimodal embeddings for image–text retrieval: A survey. ACM Computing Surveys.
- Kumar, S., Bhatnagar, D., et al. (n.d.). (n.d.). *Web-based lost and found system. MIT Moradabad.*
- MissingX. (2025). What have you lost?, Report a found item. Retrieved June 30, 2025, from <https://www.missingx.com/en>.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. Proceedings of the 38th International Conference on Machine Learning.
- Radford, A., Kim, J. W., et al. (2021). Learning transferable visual models from natural language supervision. Proceedings of the 38th International Conference on Machine Learning.
- Ramakrishnan, K. (2020). FastAPI: High-performance APIs with Python. O'Reilly Media.
- Suchana, K., Alam, S. M. E., Meem, A. T., Turjo, M. D., et al. (2021). Development of user-friendly web-based lost and found system. Journal of Software Engineering and Applications, 14(4).
- Tan, S. Y. & Chong, C. R. (2023). An Effective Lost And Found System In University Campus. Journal of Information System and Technology Management, 8 (32), 99-112.
- TROOV. (2025). Report a lost item, Report a found item. Retrieved June 30, 2025, from <https://troov.com/en/lost>.

- Wang, X., & Gupta, A. (2015). Unsupervised learning of visual representations using videos. IEEE International Conference on Computer Vision.
- Wilson, D. C. (2023). *AppalLOCATE: A lost and found solution [PDF]*. University of North Carolina at Greensboro.
- Xue, Q., Ma, H., & Sun, C. (2022). Exploratory Research on Blockchain-based Lost and Found Platform. 2022 7th International Conference on Intelligent Informatics and Biomedical Science (ICIIBMS).
- Zhang, Y., Peng, S. (2018). Design and Implementation of the Lost-and-Found System Based on Amap API.

Appendix A: Time Schedule

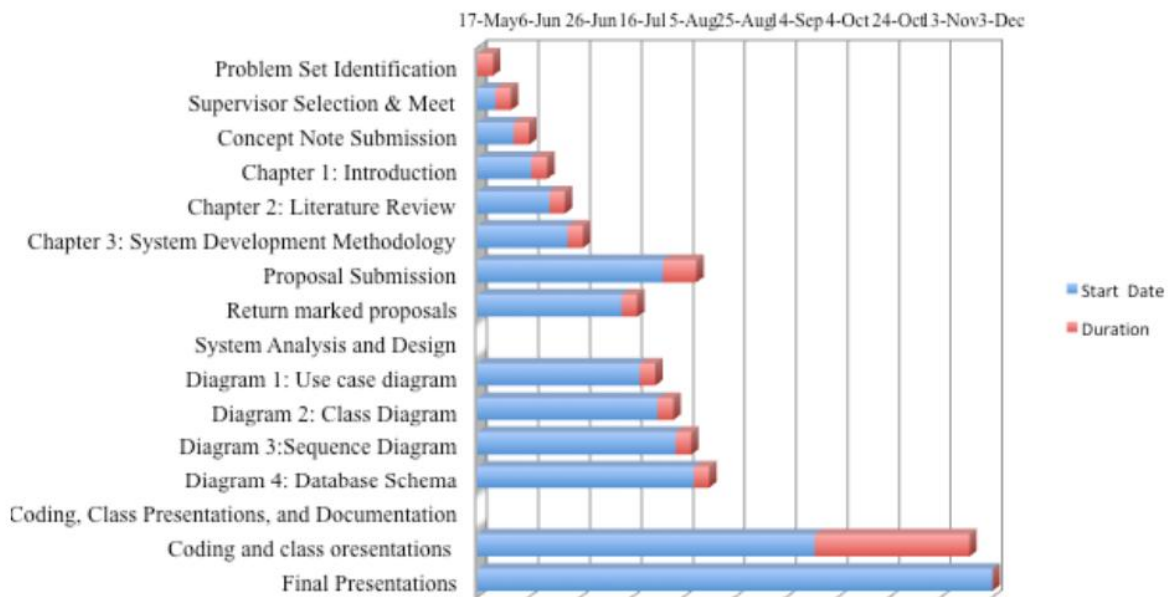


Figure A.1: Time Schedule

Appendix B: User Guide and Management

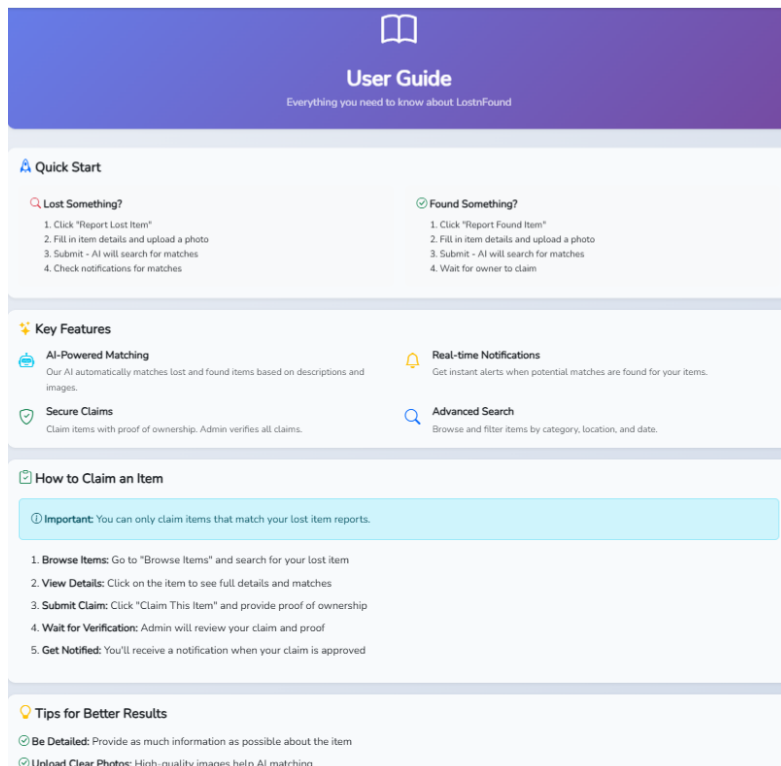


Figure B.1: User Guide

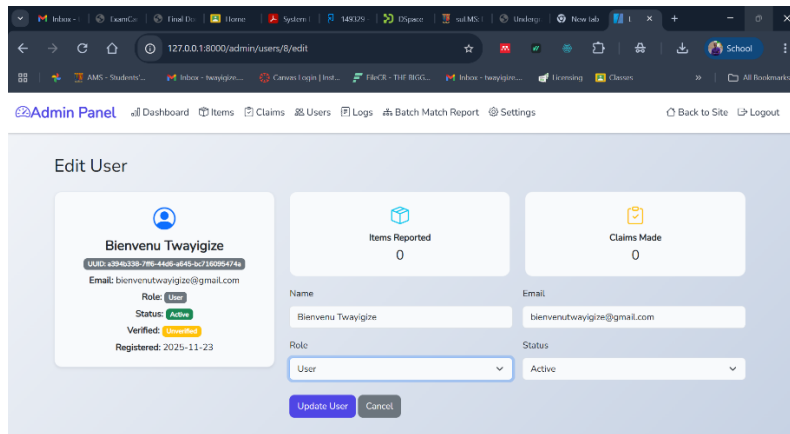


Figure B.2: User Management

Appendix C : Browser item

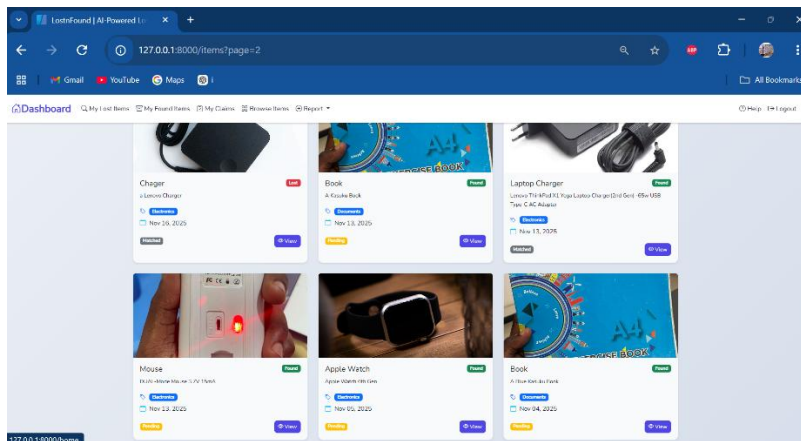


Figure C.1: Browse Items

Appendix D: The Claiming section

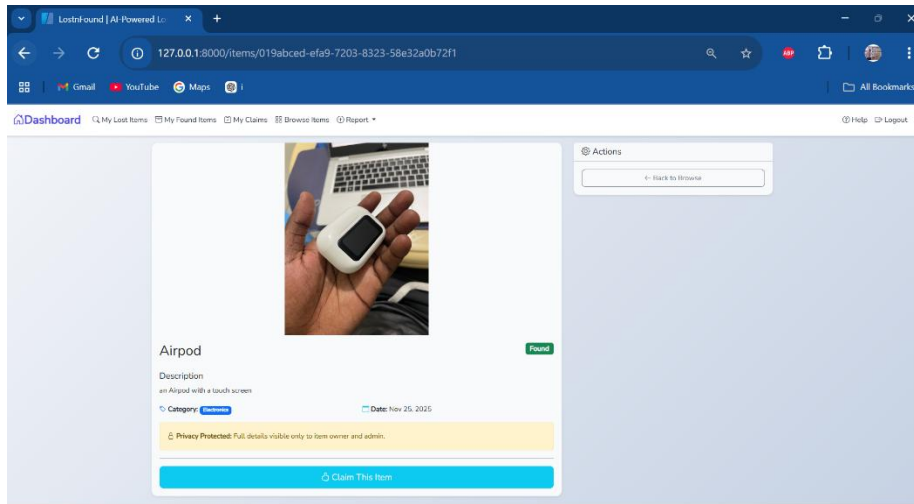


Figure D.1: Claim Item

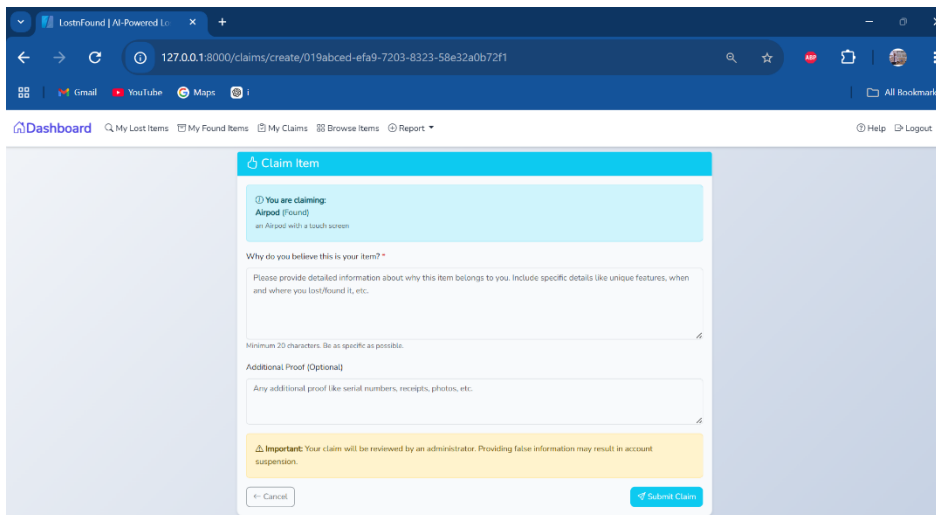


Figure D.2: Claim Item Section

Appendix E: GitHub Repository

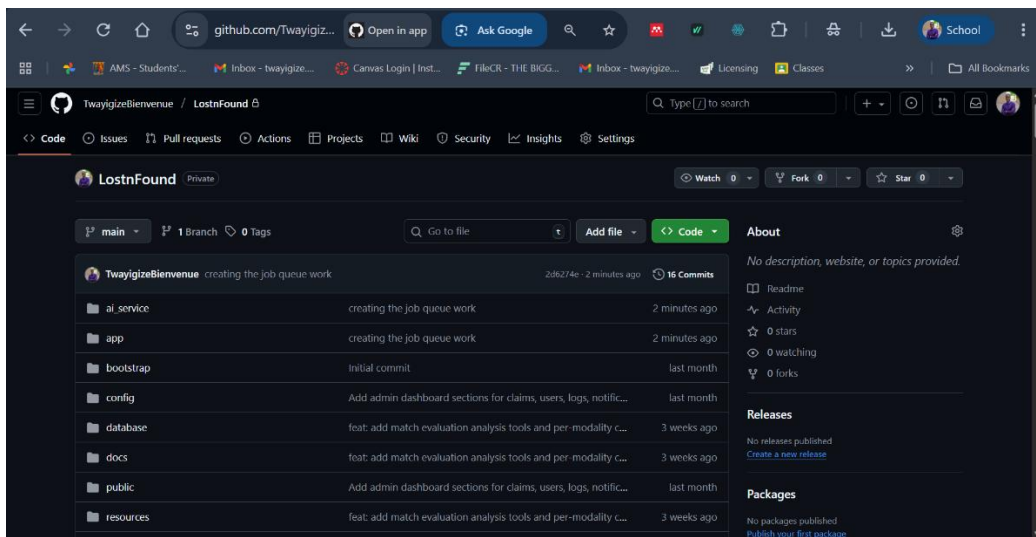


Figure E.1: GitHub Repository

Appendix F: System Settings

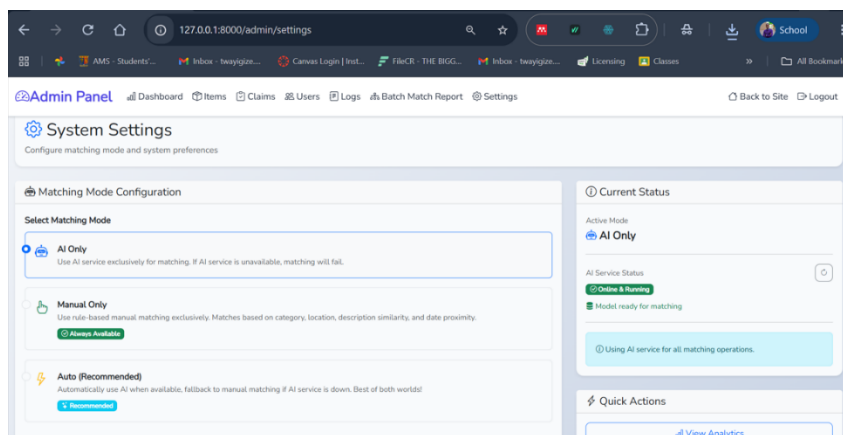
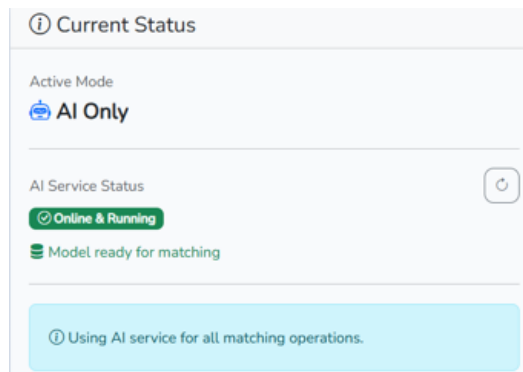


Figure F.1: Settings the Model is ready



```

18 # Start uvicorn -
19 uvicorn main:app --reload --port 8001 # Start the AI Matching Ser
20 #uvicorn main:app --reload --port 8001 --- IGNORE ---
21

```

```

(venv)
bienvy@bienvy MINGW64 ~/LostnFound/ai_service (main)
$ uvicorn main:app --reload --port 8001
Using a slow image processor as `use_fast` is unset and a s
low processor was saved with this model. `use_fast=True` wi
ll be the default behavior in v4.52, even if the model was
saved with a slow processor. This will result in minor diff
erences in outputs. You'll still be able to use a slow proc
essor with `use_fast=False`.
✓ Model loaded successfully in 8.36s
✓ Using device: cpu
INFO: Application startup complete.
INFO: 127.0.0.1:58263 - "GET /health HTTP/1.1" 200 OK
INFO: 127.0.0.1:58265 - "GET /health HTTP/1.1" 200 OK

```

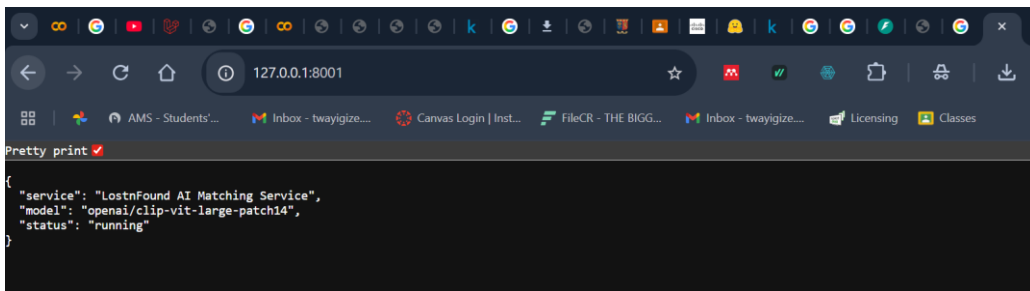


Figure F.2: The Virtual Environment and Uvicorn

Appendix G: Turnitin Originality Report

turnitin Page 1 of 66 - Cover Page Submission ID: 2945331154250

Twayigize Niyitegeka
149329_Group C.docx
 Srathmore University (Main Account)

Document Details

Submission ID tmoid::2945331154250	55 Pages
Submission Date Nov 27, 2025, 8:00 PM GMT-3	9,459 Words
Download Date Nov 27, 2025, 9:05 PM GMT-3	56,888 Characters
File Name 149329_Group C.docx	
File Size 5.4 MB	

turnitin Page 1 of 66 - Cover Page Submission ID: 2945331154250

24% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

- 189 Not Cited or Quoted 22%**
Matches with neither in-text citation nor quotation marks
- 16 Missing Quotations 1%**
Matches that are still very similar to source material
- 0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 11% Internet sources
- 2% Publications
- 22% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we see a worrying signal, we flag it for you to review.

A flag is not necessarily an indicator of a problem. However, we do recommend you focus your attention there for further review.

Match Groups

- 189 Not Cited or Quoted 20%**
Matches with neither in-text citation nor quotation marks
- 16 Missing Quotations 1%**
Matches that are still very similar to source material
- 0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 11% Internet sources
- 2% Publications
- 22% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	4%
2	Submitted works	2%
3	Submitted works	<1%
4	Submitted works	<1%
5	Submitted works	<1%
6	Submitted works	<1%
7	Submitted works	<1%
8	Submitted works	<1%
9	Submitted works	<1%
10	Submitted works	<1%

11	Internet	srmus.ac.in	<1%
12	Submitted works	Strathmore University (Main Account) on 2025-11-27	<1%
13	Submitted works	Durban University of Technology on 2025-05-14	<1%
14	Internet	ir.kiu.ac.ug	<1%
15	Submitted works	universiteknologimara on 2025-07-08	<1%
16	Submitted works	Colorado Technical University Online on 2025-03-31	<1%
17	Internet	thesis.unipd.it	<1%
18	Submitted works	Sheffield Hallam University on 2025-01-09	<1%
19	Submitted works	Southern Cross University on 2020-01-09	<1%
20	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
21	Internet	www.ijcsi.org	<1%
22	Submitted works	Higher Education Commission Pakistan on 2014-08-24	<1%
23	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
24	Submitted works	Southampton Solent University on 2025-06-01	<1%

25	Submitted works	Strathmore University (Main Account) on 2025-05-19	<1%
26	Submitted works	Midlands State University on 2025-11-14	<1%
27	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
28	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2025-08-27	<1%
29	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
30	Internet	eprints.utar.edu.my	<1%
31	Internet	journals.mriindia.com	<1%
32	Submitted works	Monash University on 2024-10-08	<1%
33	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
34	Submitted works	University of Hertfordshire on 2025-11-10	<1%
35	Internet	www.jistm.com	<1%
36	Submitted works	Malta College of Arts, Science and Technology on 2025-11-14	<1%
37	Submitted works	Strathmore University (Main Account) on 2025-11-20	<1%
38	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%

39	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
40	Submitted works	Apayao State College-Main on 2025-10-31	<1%
41	Submitted works	The Robert Gordon University on 2023-07-13	<1%
42	Internet	blog.caidwellwachira.com	<1%
43	Internet	medicine.uodiyala.edu.iq	<1%
44	Submitted works	Kolej Universiti Poly-Tech MARA on 2024-11-25	<1%
45	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
46	Submitted works	University of Wales Institute, Cardiff on 2025-08-21	<1%
47	Submitted works	University of Ulster on 2024-01-07	<1%
48	Submitted works	University of Ulster on 2025-05-06	<1%
49	Internet	www.researchgate.net	<1%
50	Internet	www.stmik-budidarma.ac.id	<1%
51	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2024-08-12	<1%
52	Submitted works	Gulf College Oman on 2025-08-18	<1%

53	Submitted works	Midlands State University on 2025-06-02	<1%
54	Submitted works	Sydney Business Travel Academy on 2024-07-22	<1%
55	Internet	vertabelo.com	<1%
56	Submitted works	Michigan State University on 2025-10-16	<1%
57	Submitted works	Strathmore University (Main Account) on 2025-05-18	<1%
58	Submitted works	Universiti Islam Ancarabangsa Sultan Abdul Halim Mu'adzam Shah (UniSHAMS) o...	<1%
59	Submitted works	Yakin Doğu Üniversitesi on 2025-11-03	<1%
60	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
61	Submitted works	University of Newcastle on 2025-10-31	<1%
62	Internet	bora.uib.no	<1%
63	Internet	studentshare.org	<1%
64	Internet	5dok.net	<1%
65	Submitted works	Arab Open University on 2025-01-01	<1%
66	Submitted works	Higher Education Commission Pakistan on 2017-11-14	<1%

61	Submitted works	Queen Mary and Westfield College on 2024-07-08	<1%
62	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
63	Submitted works	University of Hertfordshire on 2025-11-24	<1%
64	Submitted works	Xiamen University on 2023-07-17	<1%
65	Internet	huggingface.co	<1%
66	Submitted works	Arab Open University on 2024-11-07	<1%
67	Submitted works	Bindura University of Science Education on 2024-05-30	<1%
68	Submitted works	Crown Institute of Business and Technology on 2025-05-11	<1%
69	Submitted works	ICCI on 2025-09-19	<1%
70	Submitted works	Sorsogon State University on 2025-05-22	<1%
71	Submitted works	Strathmore University (Main Account) on 2025-05-19	<1%
72	Submitted works	Strathmore University (Main Account) on 2025-11-20	<1%
73	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
74	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%

75	Submitted works	Technological Institute of the Philippines on 2023-11-20	<1%
76	Submitted works	University of Sunderland on 2012-05-28	<1%
77	Internet	irojournals.com	<1%
78	Internet	19-25-13 http://146.230.128.141/jspui/bitstream/19413/6031/1/nyengeza_M_2016...	<1%
79	Submitted works	Hong Kong University of Science and Technology on 2024-10-27	<1%
80	Submitted works	International Islamic University Malaysia on 2018-05-16	<1%
81	Submitted works	Quest International University Perak on 2025-10-16	<1%
82	Submitted works	Strathmore University (Main Account) on 2025-05-19	<1%
83	Submitted works	Strathmore University (Main Account) on 2025-11-20	<1%
84	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
85	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
86	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
87	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
88	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%
89	Submitted works	Strathmore University (Main Account) on 2025-11-26	<1%

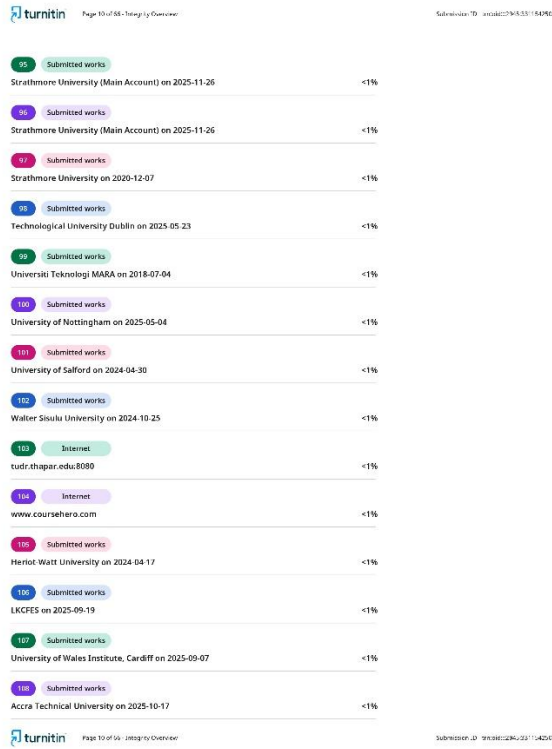


Figure G.1: Turnitin Originality Report