UNIVERSITI POLY-TECH MALAYSIA

AI Image Detection for Parcel Scam System

NURMIZA BINTI SHAHRULNIZA

BACHELOR OF INFORMATION TECHNOLOGY (HONS) IN CYBERSECURITY

UNIVERSITI POLY-TECH MALAYSIA

Faculty of Computing & Multimedia

AI Image Detection for Parcel Scam System

Nurmiza Binti Shahrulnzia

AM2311015320

FYP4085

AUGUST 2025

Declaration of Originality

This project is all my own work and has not been copied in part or in whole from any other source except where duly acknowledged. As such, all use of previously published work (from books, journals, magazines, internet, etc.) has been acknowledged within the main report to an item in the References or Bibliography lists.

I also agree that an electronic copy of this project may be stored and used for the purposes of plagiarism prevention and detection.

Copyright Acknowledgement

Signed:  Date: 9/7/2025

Office Stamp

Abstract

The results point to a steady rise in the frequency of parcel scams in Malaysia, with scammers repeatedly using or manipulating similar images, such as receipts for delivery, confirmation of payments, and screenshots of tracking to deceive victims. This project, entitled "AI Image Detection System for Parcel Scams," was thus developed to help curb the problem by providing the public, especially Malaysian online shoppers and the general internet-using population, with easy and reliable access to a platform for verifying the authenticity of parcel-related images. The system employs AI-based analysis to spot reused, manipulated, or AI-generated images, presenting to the user a probability score and detailed explanation of the results. Extra features, such as scam awareness tips, real Malaysia scam case references, and an integrated user manual, further enhance user understanding and prevention efforts.

This project is developed using the Agile methodology; hence, it allows iterative refinement based on feedback and evolving requirements. The overall system design includes a structured database, user-friendly interface, and a well-defined admin dashboard for managing content and monitoring user activities. As a whole, this solution contributes toward strengthening public awareness to minimize parcel scam threats and supporting safer online practices in Malaysia.

TABLE OF CONTENTS

List of Figures

List of Tables

Acknowledgements

First and foremost, Alhamdulillah. All praise and thanks are due to Allah SWT for the strength, patience, and guidance He provided, enabling me to successfully complete this Final Year Project. This work's completion depended on His blessings.

My sincerest thanks go to my supervisor, Puan Raznida bt Isa. Her unwavering support, encouragement, and insightful guidance were invaluable during the course of this project. Her insights, along with her willingness to take the time, and her deep understanding, have been instrumental in making the system better and in clarifying the points made in this report.

I would also like to express my gratitude to the client for this project, Mindscope Sdn. Bhd., for the insightful inputs and industrial contexts that have helped to strengthen the practicality of the system.

I also want to extend my thanks to my lecturers, classmates, and friends. Their collaboration, insights, and encouragement throughout the CT206 Cybersecurity program at University Poly-Tech Malaysia (UPTM) were invaluable. My deepest gratitude goes to my family. Their unwavering love, understanding, and prayers were a constant source of encouragement during this journey.

And to everyone who lent a hand, however big or small—thank you. May Allah bless you for your kindness and help.

1 INTRODUCTION

1.1 Introduction

University Poly-Tech Malaysia (UPTM) aims to produce graduates who are not only skilled but also ethical and ready for the workforce, spanning various fields, though it places a strong emphasis on Information Technology and Cybersecurity. For those studying Cybersecurity (CT206), the curriculum's core elements analytical thinking, secure system design, and a keen understanding of digital threats are essential. The digital world is evolving rapidly, and cyber threats are on the rise, affecting everyone from individuals to businesses and government bodies. Online fraud is a growing worry, particularly parcel scams, which have become increasingly prevalent in Malaysia. As reported by Bernama (2025), parcel scams remain a significant issue, with many falling victim to fake delivery notifications, manipulated images, and deceptive claims that take advantage of people's trust. These deceptive practices often rely heavily on fabricated visual content, thereby making it difficult for individuals to differentiate between genuine and manipulated imagery. The emergence of Artificial Intelligence (AI) has introduced advanced capabilities for analysing and identifying deceptive online behaviours. AI-based solutions are now widely recognized as valuable tools in the fight against digital deception, offering automated methods for detecting suspicious patterns, image modifications, and misleading content. Papasavva et al. (2025) highlight that AI-driven fraud detection systems have become essential in countering the evolving techniques of cybercrime, enabling quicker and more accurate identification across digital platforms. These technologies provide not only improved efficiency but also an increased level of security for users who may have limited technical knowledge.

Acknowledging the seriousness of online fraud and its potential impact on public safety, this Final Year Project (FYP) is being conducted in collaboration with Mindscope Sdn. Bhd., a client focused on enhancing digital security and awareness. The project's objective is to design and create an AI-powered Image Detection System, which will assist users in recognizing potentially manipulated or fraudulent images associated with parcel deliveries. By integrating AI detection capabilities with fraud awareness materials and educational resources, the system aims to enable both proactive prevention strategies and investigative responses to online threats.

1.2 Project Background

The proliferation of digital communication platforms has fundamentally altered Malaysian interactions, shopping habits, and delivery systems. Nevertheless, this swift transition has concurrently facilitated a surge in parcel-related scams, wherein cybercriminals employ manipulated visuals, counterfeit tracking information, and deceptive courier assertions to defraud victims. These scams demonstrate a heightened efficacy due to their exploitation of commonplace activities, thereby rendering individuals more susceptible to misinformation and social engineering strategies. According to Bernama (2025), such occurrences are on the rise, underscoring the critical necessity for enhanced public awareness and the implementation of effective technological measures to identify and thwart fraudulent endeavors. Within the context of higher education and the cybersecurity field, students, especially those pursuing the CT206 Cybersecurity program, are expected to understand, assess, and mitigate cyber threats. Conversely, despite possessing theoretical knowledge, many individuals still demonstrate a lack of practical skill in identifying misleading visual content and AI-generated images, which are increasingly utilized by cybercriminals. This gap in practical application highlights the need for practical tools designed to enhance both technical proficiency and digital literacy. Mindscope Sdn. Bhd., the collaborating client for this endeavor, is dedicated to digital awareness and seeks to enhance public comprehension of cyber-related hazards. In alignment with their commitment to community safety, they recognized the escalating difficulty citizens encounter in differentiating authentic parcel delivery images from those manipulated for fraudulent purposes. Consequently, this final year project was undertaken to provide a readily available and user-friendly solution to this specific issue.

The AI Image Detection System for Parcel Scams is engineered to scrutinize uploaded images, assessing them for indicators of manipulation or AI generation. This system integrates an AI-driven detection engine, scam awareness guidance, references to actual Malaysian cases, and a user manual, all designed to assist users in identifying potential threats prior to victimization. Papasavva et al. (2025) highlighted the critical role of AI-based fraud detection tools within contemporary cybersecurity initiatives, given their capacity to enhance accuracy and diminish dependence on manual verification processes. Through the development of this system, the project contributes to bolstering digital security, aligning with Mindscope’s objectives, and providing cybersecurity students with practical experience in real-world threat-mitigation technologies. Consequently, this project represents a tangible and educational advancement in the fight against the dynamic landscape of online fraud in Malaysia.

1.3 Problem Statements

A problem statement identifies the key issues that the proposed project aims to address and highlights the gaps that currently affect users, organisations, or industries. For this Final Year Project, several concerns related to parcel scam awareness and misuse of digital images have been identified.

1.3.1 Lack of Public Awareness and Tools for Verifying Scam-Related Images in Malaysia

Despite the increasing number of parcel scam cases reported nationwide, public awareness regarding visual scam verification remains limited. Many Malaysians are still unable to differentiate between legitimate and manipulated images commonly used by scammers. According to Bernama (2025), parcel fraud remains a persistent threat, with criminals frequently using edited images, fake receipts, and fabricated delivery notices to deceive victims. The absence of accessible tools makes users susceptible to misinformation and fraud.

1.3.2 Limited Integration of Image Analysis Tools into Parcel Scam Investigations

While Artificial Intelligence (AI) plays an increasingly important role in fraud detection, its integration into scam-related image verification is still minimal in Malaysia. AI-based detection models have demonstrated effectiveness in analysing digital content and identifying suspicious patterns (Papasavva et al., 2025), but such tools are not widely available to the public. This lack of adoption prevents users and investigators from leveraging automated image analysis to detect manipulated or reused images.

1.3.3 Lack of a Localized Platform to Check Parcel-Related Image Authenticity

Currently, there is no centralized local platform tailored specifically for Malaysians to verify whether parcel-related images are genuine or fraudulent. Existing solutions are either foreign-based, not user-friendly, or not focused on scam-related content. Without a localized verification system, users struggle to independently assess the credibility of images shared through

platforms such as WhatsApp, Telegram, or social media, increasing the risk of falling for parcel scams.

1.4 Project Objectives

Clear project objectives ensure that the system development aligns with solving the identified problems. This project focuses on improving scam awareness, strengthening user protection, and enabling automated detection of manipulated parcel-related images.

1.4.1 To Raise Awareness and Educate Users on Parcel Scam Images in Malaysia

The project aims to enhance public understanding especially among students and online shoppers regarding how scammers use misleading images to deceive victims. By providing accessible educational content, users can better recognize warning signs and make informed decisions when encountering suspicious images.

1.4.2 To Develop a System that Allows Users to Upload and Verify Images Suspected to Be Used in Parcel Scams

The system allows users to upload any parcel-related image, such as tracking screenshots or payment receipts, and receive an automated analysis. Through this mechanism, users can validate whether an image has been manipulated, reused, or modified, improving their ability to avoid scam attempts.

1.4.3 To Identify and Analyze Reused or Manipulated Images Commonly Associated with Parcel Scams

Using AI-based image analysis, the system identifies key indicators of image manipulation and checks for similarities with previously analysed scam-related content. This objective supports more accurate detection and assists users in determining whether an image is trustworthy.

1.5 Scope and Target User

1.5.1 Project Scope

The scope of this project focuses on developing an image-based scam detection system aimed at helping users identify reused, manipulated, or suspicious images commonly found in parcel scam cases in Malaysia. The system provides users with the ability to upload images, analyse them through an AI-assisted detection process, and receive a clear explanation of whether the image shows signs of fraud. In addition, the system includes supporting features such as access to parcel scam tips, a list of real Malaysian scam cases, and a user-friendly manual to guide new users.

The project encompasses several key components: crafting a user-friendly front-end interface, developing secure back-end endpoints for image processing, and incorporating machine-learning detection tools. The system also handles database management, which is essential for storing detection history, scam tips, and all the relevant case information. The system is deployed in a controlled environment to ensure accessibility, stability, and data protection. User feedback and testing outcomes will inform subsequent refinements, aiming to enhance the system's precision, ease of use, and dependability.

1.5.2 Product Scope

The project's final output comprises three primary elements:

1. Image Detection Module

This module enables users to submit an image for analysis, specifically to identify indications of manipulation, unauthorized reuse, or suspicious patterns linked to parcel scams. The module presents the confidence percentage, a probability score, the probable AI generator (if relevant), and a straightforward explanation to facilitate comprehension of the results.

2. Educational Resources

The system offers scam awareness resources, including parcel scam advice and a selection of Malaysian scam case studies. These materials are designed to educate users about common tactics employed by scammers, like the reuse of images, the creation of phony proof of delivery, and the sharing of doctored screenshots.

3. Admin Management System

Administrators have the ability to upload new content, including scam tips, case studies, and user manuals. They can also keep track of detection history, manage existing content, and ensure the platform stays current and useful.

1.5.3 Target User

I) Malaysian Online Shoppers

This group is the main focus, as they frequently use parcel delivery services and online marketplaces. They are particularly at risk of encountering fake images that claim to be shipping slips, courier tracking information, or proof-of-delivery photos. Therefore, the system offers a quick way to verify these images, which helps reduce the chance of being scammed.

II) General Internet Users Vulnerable to Parcel Scams

This category encompasses elderly individuals, those with limited digital experience, and people who regularly receive packages from online sellers or international sources. Considering that parcel scams frequently involve manipulated images disseminated through platforms such as WhatsApp, Facebook Marketplace, or SMS, the system furnishes them with a straightforward and readily available tool for authenticating the legitimacy of an image.

1.6 Overview of This Report

This report is organised into ten chapters that collectively explain the development, implementation, and evaluation of the parcel scam image detection system.

Chapter 1: INTRODUCTION

This chapter introduces the project by presenting the background, problem statement, project objectives, scope, and target users. It establishes the importance of creating a system that helps Malaysian online shoppers and general internet users verify parcel-related images and avoid online fraud.

Chapter 2: LITERATURE REVIEW

This chapter reviews past studies, existing tools, technological trends, and scam-related cases. It covers AI-based fraud detection, image manipulation techniques, and current solutions used in verifying digital evidence. The literature provides justification for the system's design and functionalities.

Chapter 3: METHODOLOGY

This chapter explains the development methodology adopted throughout the project, including each phase from requirement gathering to testing. It outlines why the chosen methodology is suitable for building an interactive and user-oriented detection system.

Chapter 4: REQUIREMENTS ANALYSIS

This chapter documents the functional and non-functional requirements of the system. It explains how the requirements were identified through research, observation, and platform analysis to ensure the system meets user needs and supports scam-prevention objectives.

Chapter 5: SYSTEM ANALYSIS

This chapter details how the system processes are structured through use case diagrams, data flow diagrams, and system flowcharts. It analyses the user interactions and overall workflow required to support image detection and educational features.

Chapter 6: SYSTEM DESIGN

This chapter presents the wireframes, interface designs, database schema, ERD, and architectural models. It explains how the visual layout, data organisation, and system components were designed to ensure usability and efficiency.

Chapter 7: IMPLEMENTATION

This chapter describes the development of each system module, including the image detection engine, admin dashboard, scam-tips management, and user interface. Screenshots and code explanations demonstrate how the system was built and integrated.

Chapter 8: TESTING

This chapter outlines all testing procedures, including functional testing, performance verification, and user acceptance testing. Results and evaluations are provided to show system reliability and accuracy.

Chapter 9: PROJECT MANAGEMENT

This chapter covers planning and management aspects such as the Work Breakdown Structure (WBS), Gantt chart, risk assessment, and resource allocation. It demonstrates how progress was monitored to ensure timely completion.

Chapter 10: CONCLUSION

This chapter summarises the system's outcomes, discusses the challenges encountered, and recommends future enhancements to improve detection accuracy and user experience.

2 LITERATURE REVIEW

2.1 Introduction

This chapter provides an overview of the literature review undertaken to inform the creation of the AI Image Detection System designed to combat parcel scams. A literature review is a crucial step in comprehending the current state of knowledge, pinpointing areas where further research is needed, and assessing the technologies presently in use that are pertinent to this project. As McCombes (2023) notes, a literature review entails a critical analysis of scholarly sources, a comparative examination of their findings, and the synthesis of various perspectives to establish a robust basis for the research. This project centers on three primary domains: the proliferation of parcel scams within Malaysia, developments in AI-driven image manipulation detection, and user comprehension of online fraud methodologies. Through an examination of prior research, actual scam instances, and current AI detection instruments, this chapter seeks to illuminate the shortcomings of existing solutions, thereby establishing the necessity for a dedicated system engineered to identify reused or manipulated parcel-related images. The conclusions drawn from this chapter provide the theoretical and empirical foundation for the system's creation, informing the design decisions, functionalities, and operational parameters detailed in subsequent chapters.

2.2 Research Topic

Parcel scams are now a prevalent form of online fraud in Malaysia, ensnaring both seasoned and novice internet users. These schemes usually employ doctored images, bogus delivery notifications, or phony tracking information to coerce victims into making payments. Anis Zalani (2025) reports that numerous Malaysians have been duped by parcel scams, often concerning packages they never actually ordered. Scammers frequently dispatch convincing images of parcels, receipts, or customs paperwork, tricking people into thinking they owe extra fees, which leads to financial losses and emotional turmoil.

Shadiqe (2024) also noted that victims had been swindled out of as much as RM394,000 in different "contraband parcel scams." The perpetrators often employed doctored or repurposed images of confiscated goods or customs paperwork, preying on victims' anxieties and fears to get them to comply. These cases underscore the growing sophistication of scam visuals and the pressing need for tools that can help confirm the legitimacy of images circulating online. In a separate instance, a Malaysian teenager fell victim to a syndicate that used fake parcel delivery images and threats, even trying to extort a ransom from the family, according to TodayOnline (2024). These instances underscore a concerning pattern: visual content

significantly influences victims, while resources available to Malaysians for authenticating parcel-related imagery are scarce.

Consequently, this study concentrates on photo analysis, AI-driven image verification techniques, and methods for detecting scam-related images. Comprehending these elements is crucial for constructing a system that can analyze dubious parcel images, identify reused content, and assist the public in recognizing scam attempts with greater efficacy.

2.2.1 Detecting Manipulated Parcel Images Using AI-Based Models

Detecting manipulated parcel images is a developing and practical field of research that intersects digital forensics, computer vision, and cybersecurity. Parcel schemes frequently depend on persuasive visual evidence such as altered receipts, counterfeit shipping labels, or reused images of parcels to compel victims to pay fees or disclose sensitive information (Zalani, 2025). The primary technical challenge is to accurately differentiate genuine, unaltered images from those that have been manipulated or recycled. Digital image forensics establishes a fundamental basis for this work by detecting traces of manipulation, including inconsistencies in illumination, duplicated regions, interpolation artifacts, anomalies in metadata, and discrepancies in camera sensor noise (Johnson & Farid, 2005; Popescu & Farid, 2005). Conventional forensic techniques identify such anomalies via manually engineered features and statistical evaluations, yet they may prove fragile when confronted with advanced editing and recompression.

Recent developments in artificial intelligence and deep learning have significantly enhanced the effectiveness of forgery detection. Convolutional neural networks (CNNs) and other representation-learning methodologies are capable of autonomously extracting discriminative features that identify subtle manipulation patterns across both spatial and frequency domains (Rössler et al., 2019). For parcel-image scenarios, training datasets should incorporate domain-specific modifications such as altered labels, inserted logos, and reused images across various fraud instances. Transfer learning utilizing extensive forgery datasets such as FaceForensics++ can enhance model performance, while subsequent fine-tuning on parcel-specific manipulations improves sensitivity to pertinent anomalies (Rössler et al., 2019).

A comprehensive detection pipeline generally comprises several stages: pre-processing (including resizing and color-space normalization), feature extraction (encompassing spatial, textural, and frequency-domain features such as wavelets or noise residuals), model inference (utilizing CNNs and ensemble methods), and post-processing (involving thresholding and

confidence scoring). Integrating forensic features such as camera response and noise residuals with deep learning embeddings has demonstrated enhanced robustness against common transformations including compression and resizing (Popescu & Farid, 2005). Furthermore, explainability is essential offering confidence scores, identified manipulated regions, or probable manipulation types enhances user trust and usability, which are crucial for a publicly accessible tool.

Persistent challenges persist: adversarial modifications may deliberately eliminate forensic evidence, and the reuse of images across various platforms complicates the process of provenance verification. Hybrid methodologies that integrate metadata analysis (EXIF), reverse-image search (to identify reuse), and deep learning techniques for tampering detection present promising avenues. Furthermore, implementing feedback loops where marked images are reviewed and utilized to retrain models can enhance detection accuracy over time. In the context of Malaysia's increasing parcel frauds, an AI-assisted forensic tool can enable users and investigators to promptly identify suspicious images, thereby facilitating swift prevention and response (Zalani, 2025).

2.2.2 Effectiveness of Machine Learning Models in Identifying AI Scam-Related Images

The rapid improvement of generative models (GANs, diffusion models) has enabled realistic synthetic images that complicate manual verification and threaten online trust. Research evaluating the effectiveness of machine learning (ML) models for fraud-related image detection investigates both their detection accuracy and robustness against image post-processing commonly seen on messaging apps and social platforms. State-of-the-art ML models for image manipulation detection typically fall into two classes: (1) deep-learning classifiers trained end-to-end on labelled forged/real images; and (2) hybrid systems combining forensic features (noise residuals, demosaicing fingerprints) with ML classifiers (Bayesian or ensemble learners).

Large-scale benchmark datasets such as FaceForensics++ have driven model improvements in detecting face-manipulation attacks (Rössler et al., 2019). However, parcel-scam images present a different distribution: they often contain text (labels, receipts), mixed media (screenshots of trackers), and compression artifacts due to multiple messaging forwards. Therefore, direct transfer of models trained on facial manipulation may underperform. Domain adaptation and fine-tuning on curated parcel-image datasets are necessary to maintain high recall and precision. Studies also show that fusion-based approaches combining spatial CNN features with hand-crafted forensic descriptors yield better resilience to compression and resizing (Bayar & Stamm, 2016; Rössler et al., 2019).

Another important facet is the interpretability and calibration of model outputs. Users need actionable information: a binary label is less useful than a confidence score, likely manipulation type, and a highlight of suspicious regions. Research into explainable ML for media forensics (Verdoliva, 2020) demonstrates that heatmaps and region-level attribution can improve human trust and aid secondary manual verification. Additionally, ensemble models and cross-checking with reverse-image search are effective in detecting reused imagery an important tactic in parcel scams.

Robustness assessment is of paramount importance; consequently, models necessitate stress-testing against prevalent real-world transformations, including recompression, cropping, colour modifications, composite edits, and adversarial perturbations. Adversarially trained detectors and augmentation methodologies can enhance model resilience, as demonstrated by Zou et al. (2021). Furthermore, operational deployment mandates either lightweight models or server-side processing, coupled with scalable APIs, to accommodate

user-submitted images while maintaining acceptable latency. In summation, the evaluation of machine learning efficacy in scam-image detection must encompass domain specificity, robustness to transformations, interpretability, and scalability for practical application.

2.2.3 Awareness Level of Parcel Scam Modus Operandi Among Malaysian Internet Users

Comprehending human factors is equally vital as technical detection in the fight against parcel frauds. Research on awareness examines how users perceive risk, interpret visual indicators, and determine whether to respond to suspicious messages. Parcel scams leverage reliance on visual proof such as delivery notes, parcel photographs, and screenshots that seem authentic to non-expert users. Recent Malaysian reports have emphasized substantial financial losses resulting from such schemes (TodayOnline, 2024; Shadiqe, 2024; Zalani, 2025), highlighting the importance of implementing targeted awareness initiatives.

Empirical research on phishing and scam susceptibility indicates that victimization rates are affected by a combination of knowledge deficiencies, emotional stimuli such as urgency and fear, and interface design (Sheng et al., 2010; Shirazi et al., 2015). Sheng et al. (2010) demonstrated that users are affected by peripheral signals such as branding and plausible sender addresses, and that training significantly diminishes susceptibility. Applying these insights to parcel schemes necessitates an analysis of how image-based cues such as logos, official-looking documents, and photographic evidence influence decision-making. Surveys and controlled experiments can measure baseline awareness levels across various demographic groups, including age, digital literacy, and frequency of online purchasing.

Awareness campaigns should integrate informative content with experiential learning. Simulations and interactive modules where users analyze instances and obtain immediate feedback have demonstrated greater effectiveness than passive instruction (Arachchilage & Love, 2014). The proposed system is capable of incorporating such pedagogical components: contextual guidance elucidating why an image may be considered suspicious (e.g., inconsistent illumination, mismatched logos), in conjunction with a detection tool that offers visual explanations. Furthermore, community reporting mechanisms and consolidated case databases enhance collective awareness by providing concrete local examples (Zalani, 2025).

Assessing awareness necessitates the use of blended methods: quantitative surveys

employing Likert-scale measures of confidence and susceptibility, qualitative interviews to examine reasoning related to images, and pre- and post-intervention assessments to evaluate learning improvements. The objective is to identify widespread misconceptions, develop targeted educational interventions, and assess decreases in hazardous behaviors. Within the Malaysian context, customizing messages for local platforms (such as WhatsApp and Facebook Marketplace) and dialects will optimize both reach and effectiveness. Integrating awareness initiatives with accessible detection tools provides a dual strategy: enabling users to identify potential frauds and delivering technical assistance when uncertainty persists.

2.3 Related Work

2.3.1 Undetecable.ai

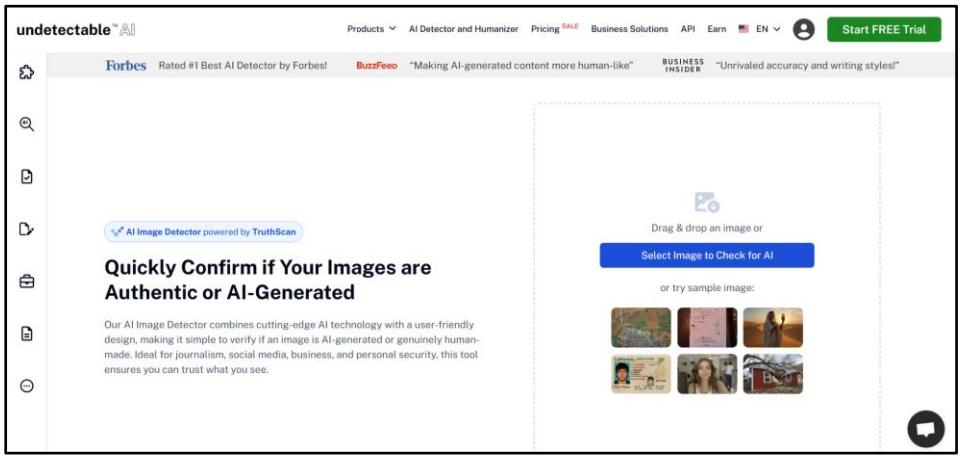


Figure 2.1: Undetecable.ai

Undetectable.ai is best known for its AI text humanizer, but it also offers an AI Image Detector, courtesy of TruthScan. This tool is designed to tell the difference between images created by AI and those taken by humans. It does this using sophisticated pattern recognition and a multi-model analysis approach. The detector looks for visual inconsistencies, oddities in the metadata, edge distortions, unusual lighting, and the telltale fingerprints left by generative models like Midjourney, DALL·E, and Stable Diffusion. A major advantage of Undetectable.ai is its ease of use. The platform features a straightforward drag-and-drop upload system, and the results are presented clearly, classifying images as either "AI-generated" or "Human-made," complete with confidence percentages.

The platform prioritizes ease of use, catering to those without a technical background

journalists, teachers, companies, and everyday internet users making it ideal for rapid authenticity assessments. Although the system excels at spotting synthetic images, it struggles with localized scam content, like the parcel scam images frequently seen in Malaysia, because it wasn't specifically trained on them. Furthermore, Undetectable.ai is more focused on detecting generative models than on analyzing image reuse or manipulation. Even with these shortcomings, the tool offers valuable insights into AI-generated images and highlights the increasing importance of detection systems in fighting digital misinformation and fraudulent visual content online.

2.3.2 Decopy AI

Figure 2.2: Decopy AI

Decopy.ai is an AI image detection tool specifically designed to differentiate between images created by artificial intelligence and those captured in the real world. It is particularly proficient in detecting images generated by models such as Midjourney, Stable Diffusion, DALL·E, and Flux. The system employs multi-layer visual analysis to detect generative anomalies, texture irregularities, unnatural illumination, and diffusion noise. Decopy.ai aims for precise accuracy by utilizing an integrated approach of neural network classifiers and probabilistic assessment. The outcome is a detection score, reflecting the probability that the uploaded image was generated by AI rather than a human.

A notable advantage of Decopy.ai is its transparency. It frequently offers a comprehensive explanation of the reasons behind an image appearing to be AI-generated, which can be beneficial in investigative situations. The interface is streamlined and user-friendly, capable of supporting image uploads of up to 10MB. This proves especially beneficial for inspecting high-resolution images. Digital artists, researchers, and teams specializing in content verification frequently utilize this tool to authenticate images circulating on social media.

However, Decopy.ai does not offer capabilities for forensic reuse detection or image analysis specifically tailored to identify scams, which somewhat restricts its effectiveness in detecting images repurposed in parcel scams. It also does not encompass sophisticated metadata tracking or comparison against a database of verified scam images. Despite these limitations, Decopy.ai continues to be among the most precise AI-generated image detectors presently accessible, illustrating the potential of contemporary AI-based discriminators to support online security and fraud detection.

2.3.3 IsGen.ai

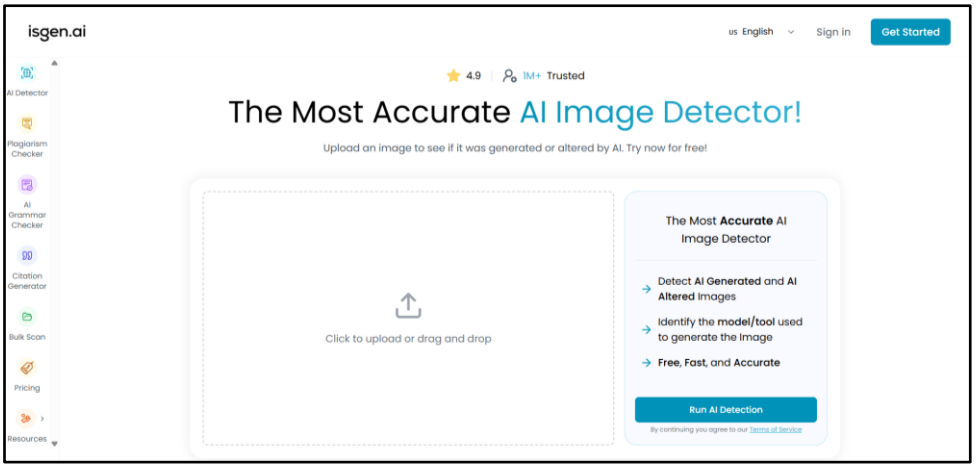


Figure 2.3: IsGen.ai

IsGen.ai is a comprehensive platform designed to detect AI-generated and AI-altered images. Unlike typical detectors that simply label an image as genuine or fake, IsGen.ai goes further. It identifies the specific AI model used to create or modify the image, whether it's Midjourney, Stable Diffusion, or Runway. This added layer of detail makes it useful for digital forensics, research, journalism, and law enforcement.

The system uses deep convolutional neural networks, trained on large datasets of synthetic and manipulated images, to spot artifacts, noise patterns, texture inconsistencies, and generative fingerprints. A key advantage is its ability to detect image alterations, not just in images created entirely by AI. This is particularly relevant in situations where fraudsters frequently modify existing images to make them appear authentic. The tool also streamlines uploads, allowing for batch scanning, and provides confidence scores to help users interpret the findings.

While it has some impressive features, IsGen.ai doesn't specifically target certain scams, like the parcel scams that are common in Malaysia. It also doesn't recognize reused images or check against known hoax image databases. Still, its strong AI-driven manipulation detection highlights the promise of advanced forensic tools in spotting misleading visual content on different platforms.

2.4 Comparison

Criteria	Undetectable.ai	Decopy.ai	IsGen.ai
Type of System	AI-generated image detection	AI-generated image detection	AI-generated & AI-altered image detection
User Interface (UI)	Simple, user-friendly, minimal	Clean, professional	Modern, intuitive, forensic-style
Detection Capability	Detects AI-generated images	Detects AI-generated images	Detects AI-generated & altered images; identifies model used
Accuracy Mechanism	TruthScan-based classifier	Multi-model CNN detection	Deep forensic CNN model
Metadata Analysis	Limited	Moderate	Strong
Customization	Low	Moderate	High (bulk scan, model identification)
Target Users	General public, businesses	Designers, researchers, verification teams	Journalists, cybersecurity, digital forensics
Advantages	Easy to use, fast results	High accuracy, transparency	Detects alterations, identifies AI model
Disadvantages	No scam-specific features	No reuse detection	Not targeted for parcel scams

Table 2.1: Comparison of Existing Project in AI Image Detection

2.5 Discussion

The comparative analysis of Undetectable.ai, Decopy.ai, and IsGen.ai elucidates their respective functionalities, advantages, and constraints concerning the requirements of a parcel scam detection system. While all three platforms are marketed as AI image detectors, their design objectives and practical applications diverge considerably, thereby exposing critical deficiencies that warrant the creation of a dedicated AI Image Detection System tailored for Malaysian parcel scam images.

Undetectable.ai prioritizes accessibility and user-friendliness, offering a simplified, novice-oriented interface designed for the general populace. Its detection capabilities are facilitated by TruthScan, demonstrating efficacy in identifying AI-generated images. Nevertheless, the system's inherent simplicity constrains its operational scope, particularly in scenarios demanding more advanced forensic functionalities. It doesn't examine metadata, image manipulation, or patterns of reuse essential for spotting parcel scam images that have been recycled across various fraud cases in Malaysia.

Decopy.ai takes a more comprehensive and transparent approach to detection, using neural network analysis to assess whether images originated from prominent generative models. Its moderate metadata analysis and accurate output make it valuable for researchers and digital investigators. While it offers more insight than Undetectable.ai, it still falls short of the critical ability to detect reused or recycled images, a frequent strategy employed in parcel scams. Furthermore, it doesn't identify AI manipulations or partial alterations, which limits its effectiveness in forensic situations where scammers edit legitimate parcel photos.

IsGen.ai clearly leads the pack. It's the most sophisticated of the three systems, capable of not just spotting AI-generated images, but also flagging those that have been tampered with using AI tools. This capability is particularly valuable for uncovering fraudulent alterations, a common tactic in scams. Furthermore, IsGen.ai's ability to pinpoint the specific model used for image generation provides a level of forensic analysis that Undetectable.ai and Decopy.ai simply don't have. However, IsGen.ai still falls short in a couple of key areas. It can't compare uploaded images against a database of known scam images, nor can it detect duplicates. This is a significant limitation, especially in parcel scam investigations, where scam photos are frequently recycled and shared across social media and messaging apps.

A consistent drawback across all three tools is the absence of localization concerning

Malaysian scams. None of the systems are integrated with Malaysian cybersecurity databases, online complaint platforms, or image repositories pertinent to scam cases. This constitutes a significant deficiency, particularly in light of the escalating incidence of parcel-related fraud reported within Malaysia.

In summation, although Undetectable.ai, Decopy.ai, and IsGen.ai exhibit sophisticated AI-image-detection functionalities, none are designed to confront the specific threat environment of parcel scams in Malaysia. This comparative analysis underscores the necessity for a specialized system such as this FYP project that concentrates on identifying manipulated and scam-related parcel images, specifically tailored for Malaysian users.

2.6 Conclusion

Chapter 2 presents the literature review, areas of research, and related works that will be used to inform the AI Image Detection System for Parcel Scams. Recent studies reveal that in Malaysia alone, cases of parcel fraud are increasing through the use of reused parcel photos, modified delivery screenshots, and other forms of falsified documentation to trick victims. Reports and studies have shown that scammers are sophisticated, yet users are vulnerable owing to a lack of awareness and instruments to verify parcel-related photographs.

AI-based identification of modified parcel graphics, the effectiveness of machine learning in recognizing imagery related to scams, and the understanding among Malaysians of parcel scam patterns-all these show major digital gaps. These studies indicate that even while AI technology is improving, localized and pragmatic solutions to identify reused scam photos or verify parcel images in messaging apps and social media are lacking.

Current AI detectors like Undetectable.ai, Decopy.ai, and IsGen.ai do mostly detect generated AI photos, but not repeated, cropped, or modified and scam-specific parcel images. This is unsuitable for verifying suspicious parcel claims in Malaysia. Therefore, it highlights the pros and weaknesses that call for a parcel scam detection system customization.

This chapter justifies the AI Image Detection System for parcel scams. From the literature study, the subjects of the research, and the comparison of various tools, it can be ascertained that a specialized system is required to increase public safety, scam awareness, and image authenticity. A design for the proposed system will be developed based on these insights, which will be covered in the subsequent chapters.

3 METHODOLOGY

3.1 Introduction

This chapter elucidates the methodological framework underpinning the creation of the AI Image Detection System, specifically designed to combat parcel scams. A meticulously structured methodology is paramount to guarantee the efficacy and systematic execution of each developmental phase, encompassing requirements gathering through to system testing. Given the project's iterative design, the necessity for continuous refinement, and the frequent adaptation to evolving scam patterns, a rigid, linear approach proved inadequate. Consequently, the Agile methodology was adopted to govern the developmental process.

Agile methodology facilitates adaptability, swift iteration, and ongoing collaboration with the client, Mindscope Sdn. Bhd., thereby ensuring the system's alignment with actual parcel scam incidents and user requirements. The system's detection logic can be progressively refined, user experience enhanced, and its ability to identify new forms of manipulated or reused fraud images improved through ongoing feedback mechanisms. This chapter explicates the Agile methodology, offering a comprehensive overview of how each phase requirements gathering, design, development, testing, deployment, and review was executed within the framework of the project.

3.2 Agile Methodology

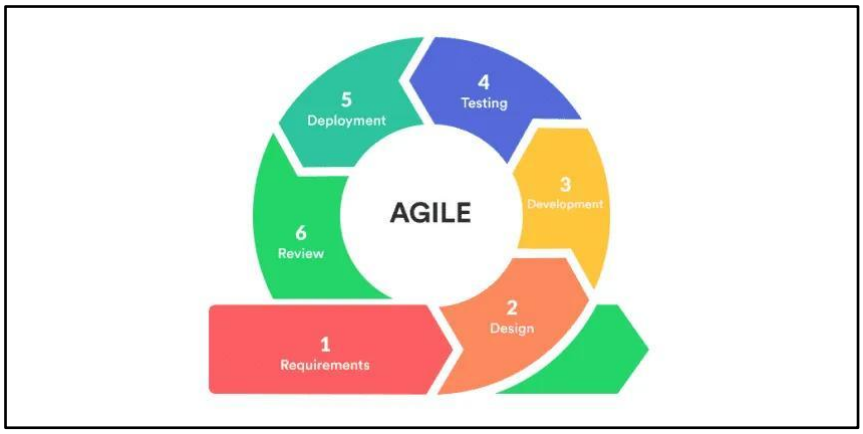


Figure 3.1: Agile Methodology

Figure 3.1 illustrates the Agile Methodology. Agile methodology is an adaptive, iterative, highly collaborative project management approach emphasizing continuous delivery and

responsiveness to change. Instead of completing the project in a single long cycle, Agile breaks work into small, manageable iterations known as sprints, thus enabling continuous progress faster problem-solving, and regular refinement based on stakeholder feedback.

Agile was selected because it is better suited for evolving cybersecurity needs, particularly for detecting parcel scams that change so frequently both in method and appearance. According to Wrike (2023), Agile enhances flexibility, enables the early and continuous delivery of functional features, and encourages constant feedback-all essential features of systems with AI detection and scam-prevention mechanisms involved. This is very significant given that parcel scams are evolving at a rapid pace, and their perpetrators generate new fake images or manipulate old ones to dupe victims. Agile can move the development team quickly and into adapting the detection logic and UI to emerging scam tactics.

Agile promotes high stakeholder involvement, thus allowing Mindscope Sdn. Bhd. to validate features, test modules, and give real-time suggestions during development. Therefore, the system remains practical, user-friendly, and in line with fraud investigation practices relevant to the industry. Due to its iterative nature, it has continuous testing and refinement of AI detection results, hence improving the accuracy and trust of users over time.

3.3 Phases in Agile Methodology

3.2.1 Requirements

The first phase of Agile Methodology in Requirements. Research from news articles and scam reports revealed that perpetrators often reused images, altered parcel labels, or manipulated pictures to mislead victims. Requirements collection encompassed consultations with the client, the analysis of common scam patterns propagated online, and the review of existing AI detection tools.

Functional requirements consisted of:

- Image Upload Module: Drag-and-Drop or Manual Upload
- AI model for detecting manipulated or repurposed images
- Confidence scores and detection results are presented in
- Administrative dashboard for supervising uploads, reporting scammer tips, and managing content.

The non-functional requirements included system performance, accuracy, usability, and aspects pertaining to data privacy. All these requirements were documented, refined, and validated in conjunction with Mindscope Sdn. Bhd. to ensure that the system effectively reach the expectation of Malaysian users.

3.3.2 Design

The system architecture and User Interface/User Experience design were formulated during the design phase. The wireframes were designed to feature a clean and straightforward interface, suitable for general users, particularly Malaysian online consumers without technical expertise. Navigation components such as "Upload Image," "Scam Tips," and "Malaysia Scam Cases" have been strategically positioned to facilitate easier access.

The technical design encompassed:

- Database schema for managing data pertaining to users, scan history, scam recommendations, and manuals
- Backend architecture utilizing Node.js and SQLite
- AI detection workflow: Image pre-processing → Model analysis ⇒ Result generation

System flowcharts, use case diagrams, and ERDs were created prior to the actual implementation to effectively delineate the interactions within the system.

3.3.3 Development

Development was carried out in sprints, with each sprint dedicated to the delivery of a functional module. The selection of Node.js for the development of the backend was due to its efficacy and compatibility with image processing libraries. AI image analysis has been deployed via model APIs that detect AI-generated patterns, recurring image features, and altered visual anomalies.

Several of these developed modules comprised:

- Image recognition system
- Administrative authentication and content submission
- Image history documentation
- Fraud Prevention Tips and Case Repository
- Integration of user interface with backend systems and their outcomes

Each iteration concluded with functional features that could be subjected to testing and evaluation.

3.3.4 Testing

Testing was performed to verify reliability, precision, and usability. Various testing methodologies were employed:

- Unit testing: To validate the functionality of individual components, including image upload, detection API, and database operations.
- Integration testing: To verify seamless interaction among the frontend, infrastructure, and AI model.
- User acceptance testing (UAT): Conducted with clients and selected users to evaluate clarity of results and ease of use.
- Black-box testing: To evaluate system behavior without inspecting internal code.

Feedback from testing was used to improve detection clarity, fix interface issues, and refine how results were presented.

3.3.5 Deployment

First, the system was deployed on a local development environment, then moved to a version accessible via a browser through localhost. Deployment ensured all modules were functioning well and the system would be able to work seamlessly in real-world use. Further, some optimization was done to improve the loading speed and enhance image processing performance.

This setup will allow easy deployment to cloud hosting if future need arises.

3.3.6 Review

The final stage was the review of all the features, accuracy of detection, UI responsiveness, and content management functions. The functionality of the system was cross-checked with the project objectives proposed at the start to ensure complete implementation. The final refinements were informed by feedback from Mindscope Sdn. Bhd., especially on enhancing result explanations, scam awareness content, and admin dashboard clarity. This review confirmed that the system is functional, user-oriented, and ready for academic submission and real-world use.

3.4 Conclusion

Consequently, the Agile methodology proves advantageous in the construction of the AI Image Detection System designed to combat Parcel Scams. The inherent iterative and flexible characteristics of the system's components ensure their ongoing enhancement. This approach allows Mindscope Sdn. Bhd. to swiftly respond to emerging findings, evolving fraud schemes, and user input, thereby fostering the creation of a functional, accessible, and practical system. Agile encompasses all stages, including requirements gathering, design, development, testing, deployment, and evaluation.

The project progressed through a series of development cycles, confirming the functionality of image uploads, AI-powered identification, result presentation, fraud case documentation, and administrative oversight. Package scams are a growing problem in Malaysia, and ongoing collaboration with users has significantly improved the system. Rigorous testing and review processes bolstered system stability and the overall user experience, especially for those without technical backgrounds who needed to verify potentially fraudulent parcel images using a simple, yet informative, interface.

4 REQUIREMENTS

4.3 Introduction

Requirement gathering is a critical phase in the Software Development Life Cycle (SDLC), being the foundation of system performance and functionality (AWS, 2024). Erroneous or vague requirements could result in software failure due to improper or incomplete design. Hence, it is essential to delineate the necessary features, performance requirements, and objectives of the AI Image Detection system. The primary function of this system is to aid Malaysian students and online customers with parcel scam detection through verification of reused or tampered pictures.

System requirements are split between functional and non-functional aspects, as per industry standards. Functional requirements dictate the individual functions the system must perform, i.e., image upload, reverse image validation, and displaying results (GeeksforGeeks, 2023). These functions need users to interact with the system successfully. Non-functional requirements, on the other hand, encompass system quality characteristics like usability, reliability, and performance (Ironhack, 2024). These ensure that the tool operates ideally with a better, smooth, and trustworthy user experience.

Both non-functional and functional requirements are of similar significance. While functional aspects facilitate users to spot fake images, non-functional aspects are responsible for usability and performance for public usage. This chapter explores the methods used in spotting and collecting these requirements for system development purposes.

4.2 Data Gathering Techniques

Effective requirement gathering is needed in order to align software development with real user needs. Different data collection methods were applied in this project for assessing user expectations, system requirements, and forensic functionality so that the final system would be operational, efficient, and easy to use (GeeksforGeeks, 2023).

An online questionnaire conducted via Google Forms was employed as the primary data collection method. This approach was selected owing to its availability and potential for broad coverage among prospective respondents, particularly university students and frequent online consumers. The survey consisted of 13 official questions which were designed to test parcel scam knowledge, own victimization, and awareness of tools that are readily available such as

Undetecable.ai, Decopy AI and IsGen.ai. A minimum of 50 responses was targeted in a bid to obtain a representative sample. The qualitative and quantitative data collected informed key functional requirements like capacity to upload images and visualization of results, and non-functional like responsiveness of the system and ease.

Second, a MindScope Sdn Bhd representative was interviewed to obtain expert opinion on forensic aid tools. Hi-tech system needs like image metadata logging and education modules were determined through the interview. Integrating user input with expert opinion, overall system expectations understanding was provided, ensuring the end product is user-centric but not technologically flawed.

4.3 Functional Requirement

Functional requirements specify the key activities that the system must perform. They are essential features that allow users to do something with the system and arrive at its objectives. Functional requirements in software development dictate the intended behavior of the application in a given situation (GeeksforGeeks, 2023). For the project AI Image Detection System the aim is to assist users in verifying whether an image has been reused or manipulated on the web. Therefore, the functional requirements capture functionalities like image upload, confirmation via image detection tools, revealing results, and orientation to scam awareness.

These were established based on questionnaire responses and interviews with experts such that technical feasibility and user needs are catered to. The system must enable users to upload images, perform reverse searches through APIs like TinEye or Google Reverse Image, and derive forensic results through hashing or metadata analysis. It should also enable reporting functionality and educational feedback so that users are aware of scam patterns.

Functional Requirement	Description
Image Upload Function	The system shall allow users to upload parcel-related images through drag-and-drop or manual file selection in supported formats such as JPG, PNG, JPEG, and WebP.
AI Image Detection	The system shall analyze the uploaded image using an AI model to determine whether it is manipulated, reused, or potentially related to a parcel scam.

Display Detection Results	The system shall present clear output results that include confidence scores, classification labels (real, manipulated, AI-generated), and alerts indicating possible scam indicators.
Admin Content Management	The system shall allow administrators to upload, update, and delete scam tips, Malaysia scam cases, and user manual files to support public awareness.

Table 3: Functional Requirement Table

4.4 Non-Functional Requirement

Non-functional requirements describe how the system is to function and not what it has to do. They have an impact on the user interface and software quality. Non-functional requirements include usability, reliability, performance, security, and privacy (Ironhack, 2024). Non-functional requirements in this project render the system efficient, secure, and easy to use particularly for non-technical users.

Usability requirement makes the interface user-friendly and intuitive for new users. Performance causes image scan and API requests to occur rapidly. Reliability entails ensuring system stability via image verification, while security protects users from potential data leaks. Lastly, privacy is ensured by avoiding saving and reusing uploaded images without the consent of the users (AgileMania, 2023).

Non-Functional Requirement	Description
Performance	The system should process and analyze images within 5–8 seconds to ensure smooth interaction and minimize waiting time.
Usability	The system interface must be user-friendly, easy to navigate, and suitable for non-technical users such as Malaysian online shoppers and the general public.
Security	The system must protect user-uploaded images, secure sensitive data using

	environment variables (.env), and restrict administrative actions using authentication and role-based access.
Reliability	The system must provide stable and consistent results, capable of handling multiple uploads and operations without system crashes or errors.

Table 3: Non-Functional Requirement Table

4.5 System Requirement

System requirements list the needed hardware and software to build and operate the AI Image Detection for Parcel Scam System.

4.5.1 Visual Studio Code

Figure 4.1: Visual Studio Code (Dickison, 2024)

Visual Studio Code, or VS Code, is a light yet powerful source-code editor developed by Microsoft. VS Code offers support for numerous programming languages like Python, HTML, CSS, and JavaScript required to build web applications. VS Code offers in-built debugging, syntax highlighting, intelligence-based code completion, snippets, and version control that make it most appropriately tailored to modern development methodologies. Visual Studio Code is utilized to develop and modify front-end and back-end code for the forensic image verification system in this project. Its wide applicability and extensive library of extensions make it very capable in the hands of developers working on image processing and web interface projects (Hostinger, 2024).

4.5.2 Python Programming Language

Figure 4.2: Python Programming Language (LoudBench, 2023)

Python is an interpreted, high-level language widely known for its readability and ease of use. It supports multiple programming paradigms and is extensively used in fields like artificial intelligence, web development, data science, and automation. Python is utilized here to create the backend system using the Flask framework and for analysing images using libraries like Pillow and ImageHash. Python's extensive third-party library and framework support facilitate easy integration with reverse image search APIs and hashing libraries that make it a worthy candidate AI Image Detection System (Teradata, 2024).

4.5.3 Laptop

Figure 4.3: Acer Aspire A315-24P (Creatus, n.d.)

Computer Brand	:	Acer Aspire A315-24P
Processor	:	AMD Ryzen 5 7520U with Radeon Graphics, 2.80 GHz
Memory (RAM)	:	8 GB RAM
System	:	64 bit operating system

Operating System	:	Windows 11
Internal Disk Storage	:	256 GB

Table 4: Hardware Specification Table

4.6 Conclusion

This chapter outlined the requirements required to develop the AI Image Detection System. Non-functional and functional requirements were correctly defined to ensure the system functions as intended and gives an uninterrupted experience to the user. The requirements were gathered using questionnaires for users and interviews with experts to ensure the project fulfils real-world expectations. System requirements like required hardware and software were also mentioned to ensure effective development and operation. Having a clear understanding of what the system is supposed to do and how it is supposed to behave, the project can now go ahead with particular development goals and technical constraints.

5 ANALYSIS

5.1 Introduction

Data analysis is the process of transforming raw data into valuable knowledge to facilitate knowledgeable decision-making. In system development and research, it helps developers understand user behavior, expectations, and system requirements through the recognition of patterns and trends in collected data. It helps ensure the final product is tailored according to real user demands rather than assumptions (Bandhari, 2021). In this chapter, the analytical focus is to find out how the users will comprehend parcel scams, recognize reused or tampered pictures, and what they will anticipate from a verification process. The data being examined were collected from rigorous questionnaires among Malaysian users, even more students and online buyers.

This chapter plays an instrumental part in determining how the AI Detection Image System should be implemented and architected. From an exhaustive analysis of the survey data, we can validate the effectiveness of key features such as reverse image search, hash match, and scam tips. It also helps determine how comfortable the users are in adopting image verification technology, thus creating a functional yet practical system. Altogether, the analysis ensures that the solution addresses real problems and establishes valuable practicality for the intended audience.

5.2 Data Gathering Analysis

Data gathering from end-users and stakeholders is important in creating a practical solution to fulfill its function. For this project, there were two key approaches used to collect data: a survey questionnaire and an online interview involving a client representative from Mindscope Sdn Bhd. Both provided quantitative and qualitative information on user expectations, awareness levels, and practicability in deploying a forensic image verification system.

The questionnaire, developed and distributed via Google Forms, targeted would-be victims of parcel scams, which were in this case university students and online shoppers. As Kuphanga (2024) explains, questionnaires are practical in research because they can gather standardized responses effectively in groups of large numbers, yielding consistent and comparable information. The feedback received was of use in understanding user needs such as image upload support, advice for being scam-aware, and ease in viewing results.

To complement the survey, an interview was conducted with a cybersecurity officer of Mindscope Sdn Bhd, which is a digital solution company based in Malaysia. The purpose was to understand the expectations of experts and practical applications of image verification for scam detection. The officer indicated the necessity of incorporating features like reverse image search and forensic analysis in public-facing systems. They also mentioned the necessity of educating end-users and easy-to-understand reports for non-technical users.

The combination of these two approaches provided a more cohesive vision for user expectations and professional norms, and steered the creation of a system that is technically sound, user-friendly, and socially accountable.

5.2.1 Questionnaire Analysis

To gather accurate and relevant user feedback, a 13-question survey was created and shared online via Google Forms. The questionnaire covered three main sections: respondent background, awareness and experience with parcel scams, and opinions on image verification tools. A total of 54 valid responses were collected from university students, online consumers, and other Malaysian internet users who are commonly exposed to scam threats.

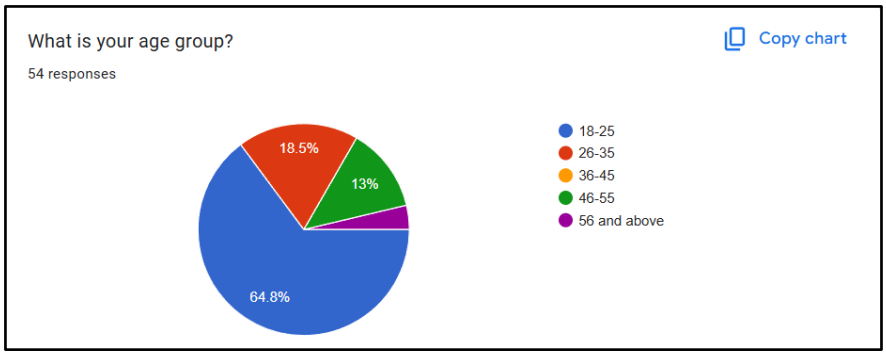


Figure 5.1: Result of Demographic Question 1 (Pre-Development)

According to the data, the majority of respondents fall within the age range of 18 to 25 years, accounting for 64.8% of total responses. This suggests that younger adults, particularly students and early-career professionals, represent the largest group potentially targeted by parcel scams. Other age groups such as 26 to 35 years (18.5%) and 46 to 55 years (13%) were less represented, while only 3.7% were aged 56 and above.

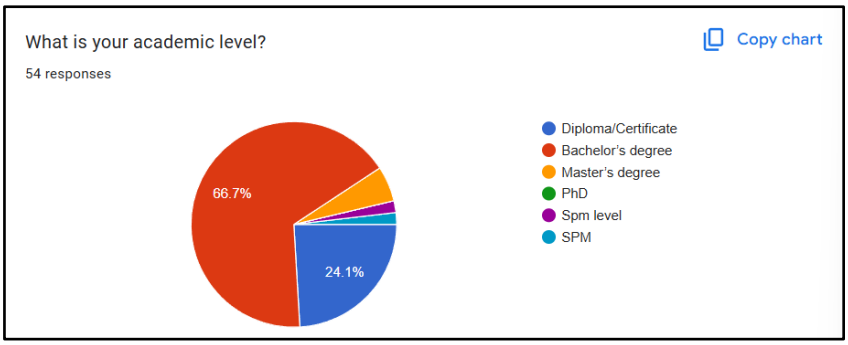


Figure 5.2: Result of Demographic Question 2 (Pre-Development)

Based on the data, 66.7% of respondents are pursuing or have completed a Bachelor's degree, while 24.1% hold a diploma or certificate. A smaller portion, 5.6%, have completed a Master's degree. The remaining percentage includes SPM-level respondents. This indicates that the majority of users have at least tertiary education, suggesting they are digitally active and likely to benefit from a scam detection tool.

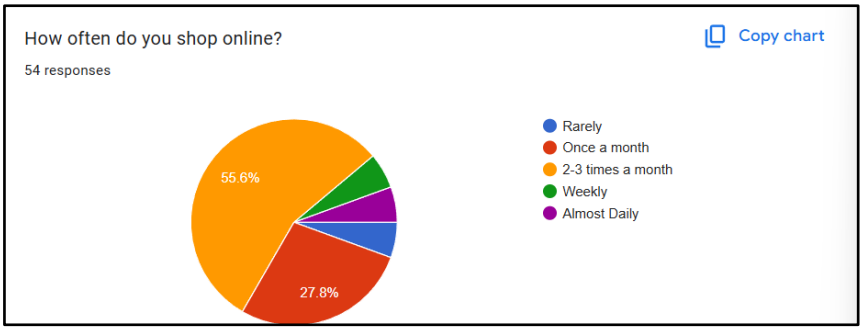


Figure 5.3: Result of Demographic Question 3 (Pre-Development)

According to the data, 55.6% of respondents shop online 2–3 times a month, and 27.8% shop at least once a month. A small portion shop weekly (5.6%) or almost daily (5.6%), while only 5.6% rarely shop online. This reflects that most users engage in online shopping regularly, which increases their exposure to potential parcel-related scams.

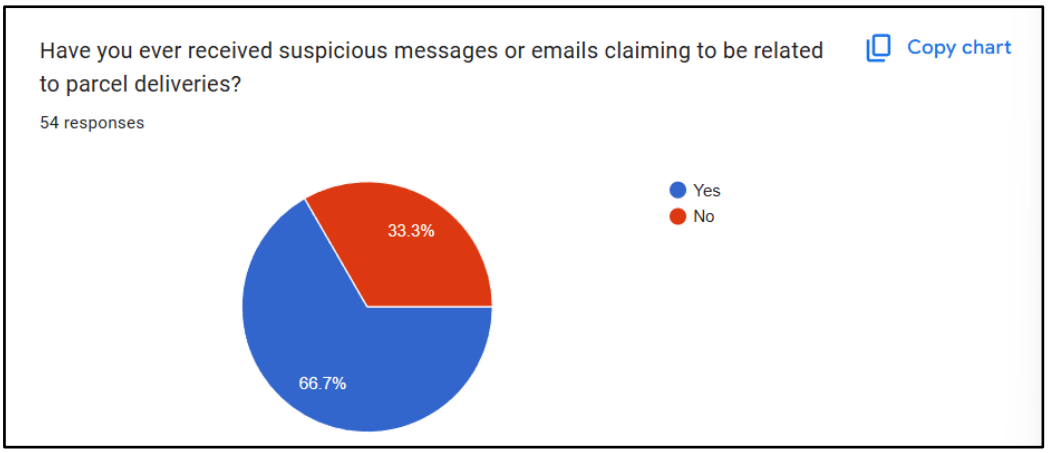


Figure 5.4: Result of Demographic Question 4 (Pre-Development)

Based on the responses, 66.7% of users have received suspicious messages or emails related to parcel deliveries, while 33.3% have not. This shows that two-thirds of the sample population have already encountered scenarios linked to potential scams, emphasizing the need for a preventive tool.

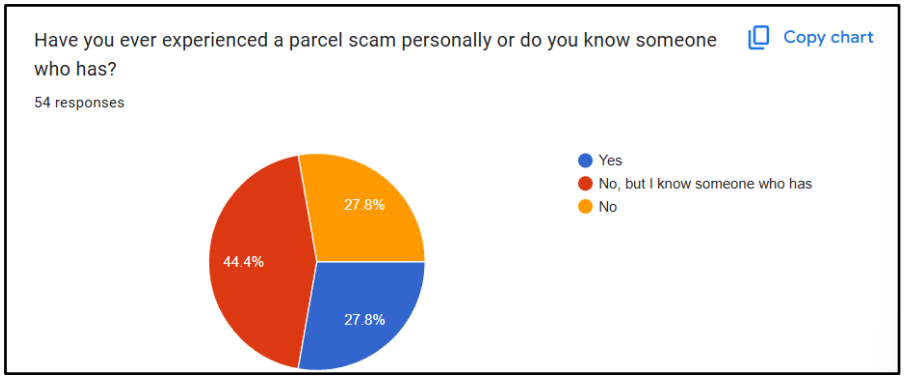


Figure 5.5: Result of Demographic Question 5 (Pre-Development)

According to the data, 44.4% of respondents have not been scammed themselves but know someone who has, while 27.8% admitted to being victims. Another 27.8% reported having no direct or indirect scam experience. This indicates that while personal experiences vary, awareness of scams within social circles is high.

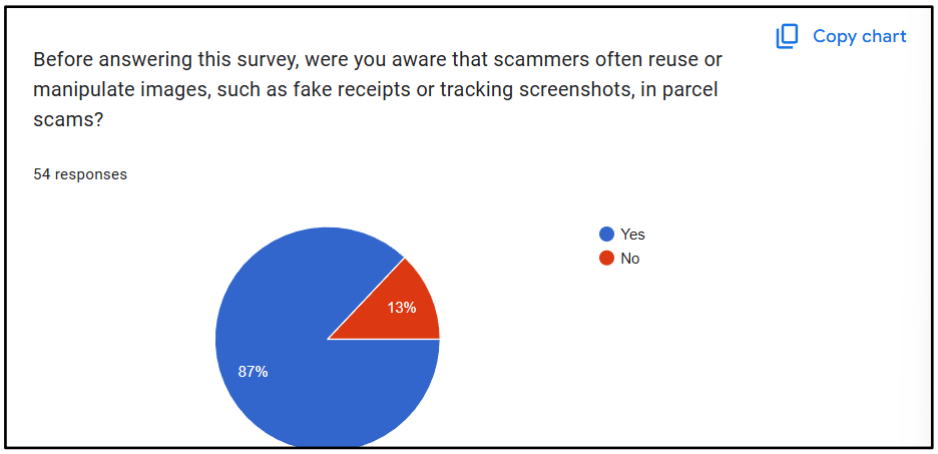


Figure 5.6: Result of Demographic Question 6 (Pre-Development)

Based on the data, 87% of respondents were aware that scammers may reuse or manipulate parcel-related images such as fake receipts or tracking screenshots. Only 13% reported being unaware. This suggests a high level of baseline awareness, validating the relevance of an image verification system.

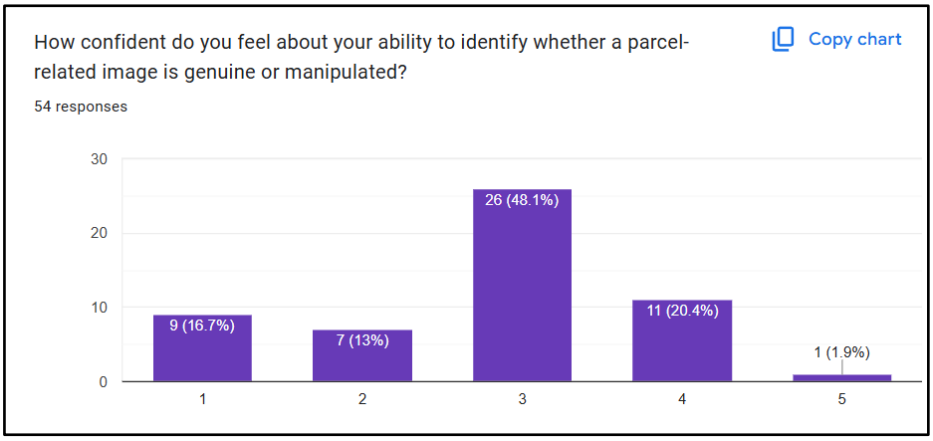


Figure 5.7: Result of Demographic Question 7 (Pre-Development)

According to the responses, 48.1% rated themselves at level 3 (moderate confidence), 20.4% at level 4, and only 1.9% felt highly confident (level 5). Meanwhile, 16.7% chose level 1 and 13% level 2, indicating low confidence. This shows that most users are unsure or lack the ability to verify images effectively without assistance.

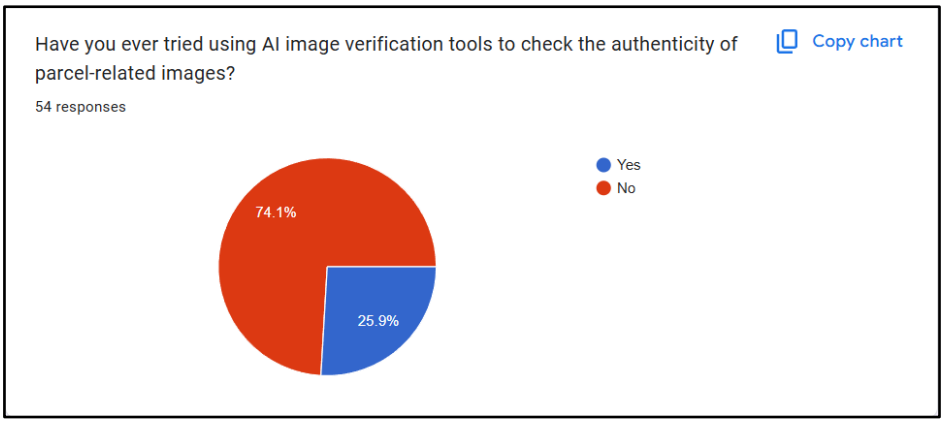


Figure 5.8: Result of Demographic Question 8 (Pre-Development)

Based on the data, 74.1% of respondents have never used AI Image Verification Tools. Only 25.9% have experience with such tools. This highlights a significant gap in public exposure to image verification technology, supporting the need for a more accessible system.

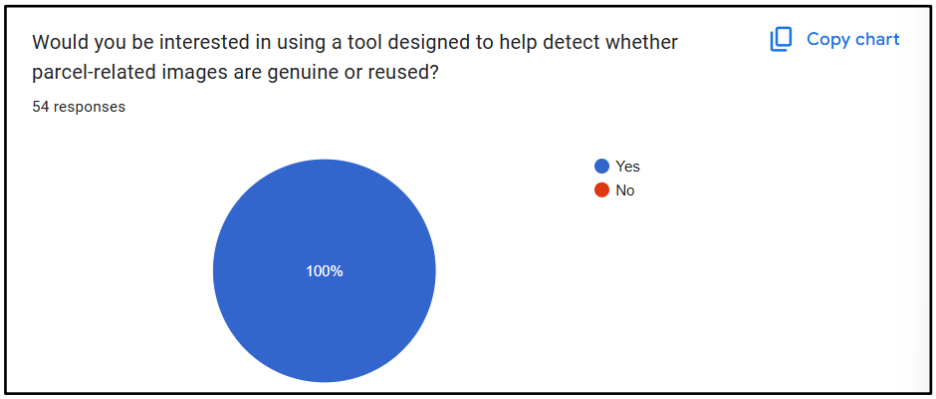


Figure 5.9: Result of Demographic Question 9 (Pre-Development)

According to the responses, 100% of respondents expressed interest in using a tool designed to detect reused or fake parcel-related images. This unanimous interest demonstrates strong demand and justifies the development of this project.

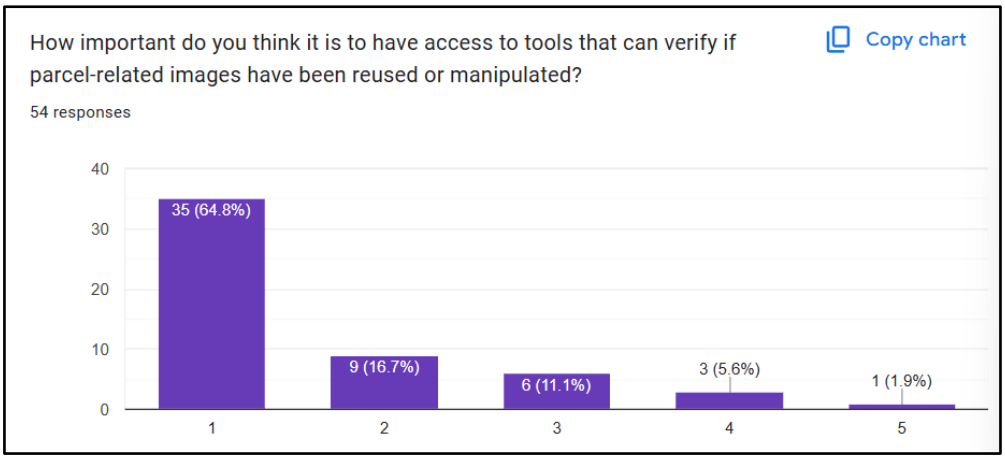


Figure 5.10: Result of Demographic Question 10 (Pre-Development)

Based on the data, 64.8% of respondents rated the importance of such tools at the highest level (1). An additional 16.7% rated it as level 2, and 11.1% at level 3. Very few respondents considered it unimportant. This confirms that users consider verification tools essential in avoiding scams.

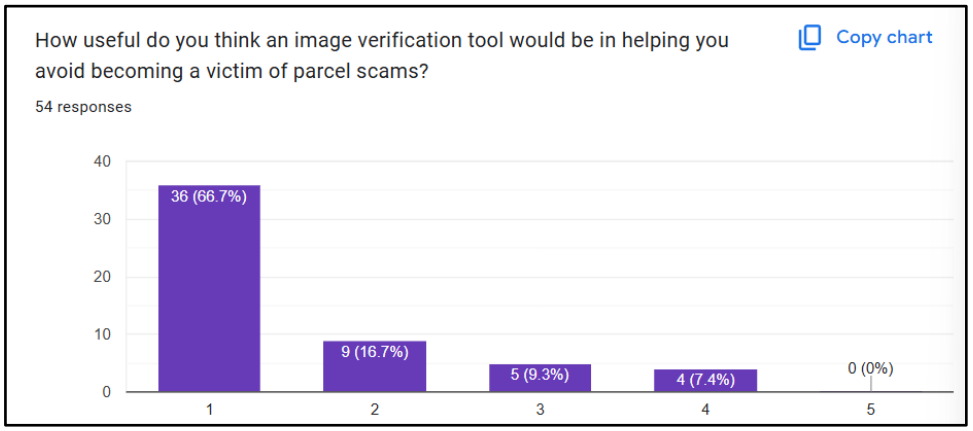


Figure 5.11: Result of Demographic Question 11 (Pre-Development)

According to the responses, 66.7% rated the tool's usefulness at level 1 (very useful), while 16.7% selected level 2 and 9.3% level 3. This suggests that most users believe image verification can help reduce their risk of falling victim to scams.

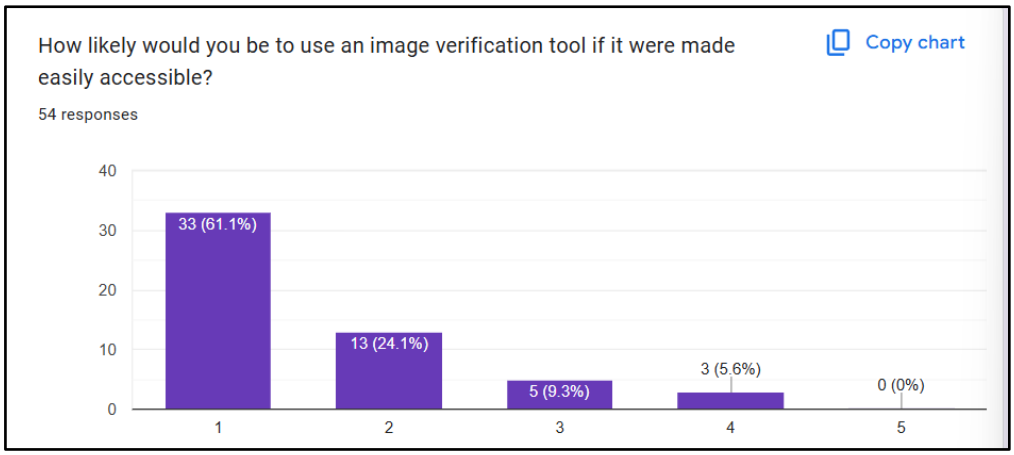


Figure 5.12: Result of Demographic Question 12 (Pre-Development)

Based on the data, 61.1% reported they would be very likely to use a verification tool if it were easily accessible (level 1), and 24.1% chose level 2. Only 9.3% and 5.6% gave neutral or less likely responses, confirming that accessibility plays a key role in user adoption.

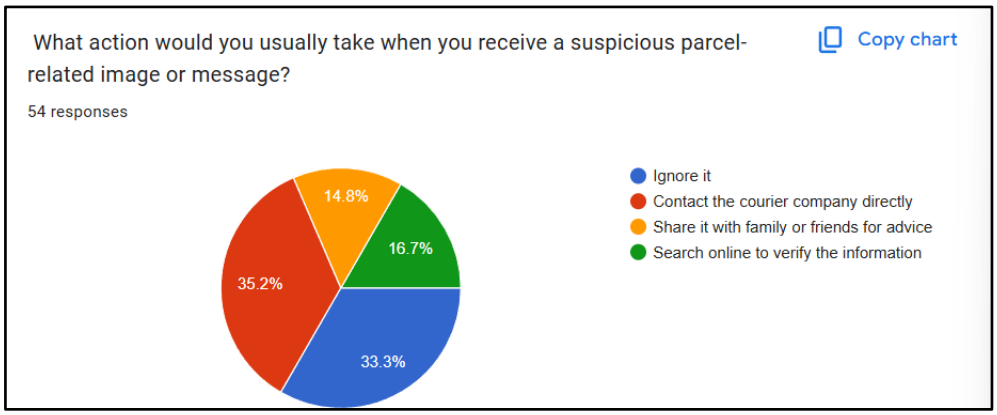


Figure 5.13: Result of Demographic Question 13 (Pre-Development)

According to the responses, 35.2% of users would contact the courier company directly upon receiving a suspicious image, while 33.3% would simply ignore it. Others (16.7%) would search online, and 14.8% would consult family or friends. This variation in response shows a lack of standard practice among users, further supporting the need for a structured scam-checking tool.

5.2.2 Interview Analysis

Interviews are an essential qualitative data collection method that achieve greater insight into a subject based on first-hand experience from experienced professionals. According to Hecker and Kalpokas (2024), interviews allow for a deeper conversation of complex matters that questionnaires might be unable to cover exhaustively. In this project, interviews played an important role in validating the system's viability from a professional cybersecurity perspective.

There was a total of three virtual interview sessions conducted via Google Meet with Ts. Badri Azni, the representative of Mindscope Sdn Bhd, on 11 June 2025, 16 June 2025, and 27 June 2025.

Figure 5.14: Interview Session with Ts. Badri Azni (Mindscope Sdn Bhd)

Some important points emerged from the interview sessions. Firstly, many users particularly non-technical users lack awareness of the ways scammers manipulate parcel-related images. Such a knowledge gap inhibits their potential to detect scam activities without relying on the use of specialized equipment. Ts. Badri emphasized that user-friendliness is a factor the system must prioritize, particularly through drag-and-drop image uploading, real-time feedback, and simplicity of results interpretation. Overly technical tools can discourage usage and diminish confidence even when they are accurate.

The expert also recommended that the system not only detect reused or manipulated photos but also provide educational support and reporting features to help users take appropriate action. Including visual scam indicators and scam prevention tips would also enhance user experience. Finally, academic and cybersecurity institutes were cited as potential collaborators to promote the use of the tool, especially among students and Malaysian online shoppers two groups that are frequently the victim of parcel scams.

Sage Research Methods (2021) states that interviews uncover motivations, opinions, and contextual data that allow for well-informed development. In this case, the results of the interviews had a direct influence on system interface design, usability features, and public engagement strategies.

5.3 Use Case Model

A UML use case diagram is a starting element of system modelling that supports specifying the way in which users (actors) engage with functionalities of a system. It provides an explicit image of the scope of a system in terms of what the system has to accomplish and not the way in which it accomplishes it (Visual Paradigm, 2024). This model-based approach allows stakeholders and developers to understand system behaviour from the user's perspective easily, which is especially important during planning and design.

In this AI Image Detection System, the use case diagram defines the main actors (Admin and User) and specifies their communication with the system. For example, users can upload images, run verification checks, and receive scam alerts, while the admin can view reports and handle education material. Use case modelling ensures that every interaction scenario is described at the outset to prevent misunderstandings during development.

According to Cacoo Staff (2021), use case diagrams not only make system design more readable but also assist in improved communication between technical and non-technical stakeholders. This makes it easier to align the final system with user expectations, which is critical for tools designed to enhance public safety and awareness.

5.4 Use Case Diagram

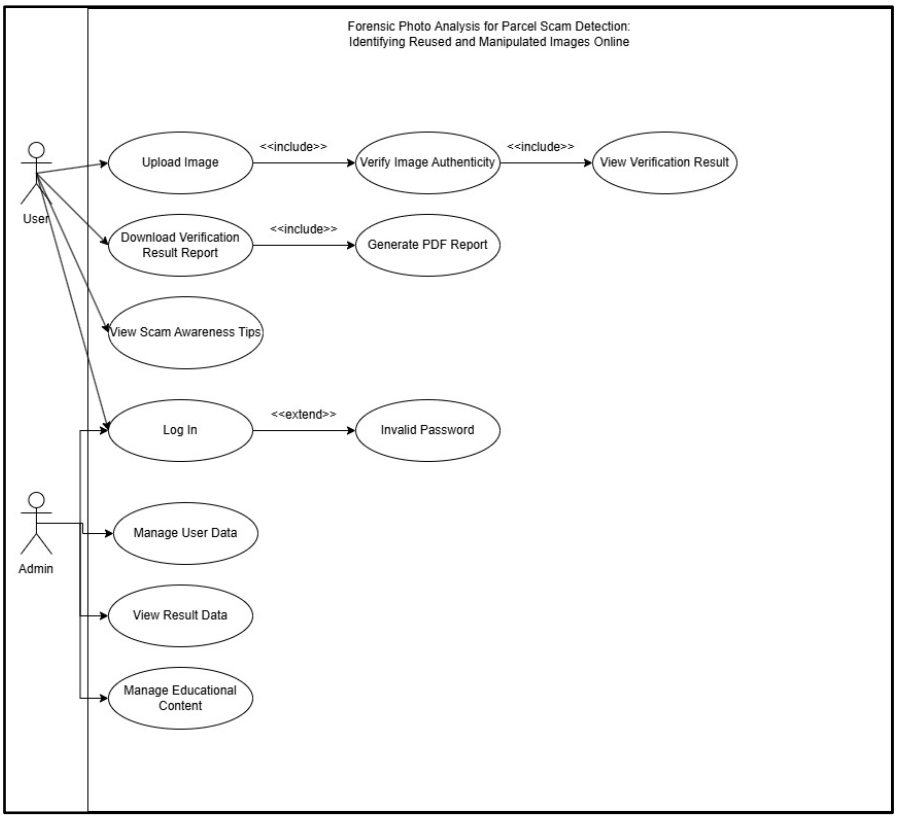


Figure 5.15: Use Case Diagram

According to Figure 5.15, the use case diagram presents the top-level interactions between the two principal actors, User and Admin, and the system: AI Image Detection for Parcel Scam System. Each actor interacts with those functions pertinent to the roles they are cast in accomplishing the system's goal of finding reused and manipulated parcel-related photos.

The User is taken through a number of key activities that start with logging-in. The process of logging in may include as an extension point the situation where an “Invalid Password” message is given. Upon successful login, the user would then upload an image allegedly involved in a parcel scam. This function includes several sub-actions: verify image authenticity and view verification results, all linked with <<include>> relationships. The user may also download the verification report, which includes generating a PDF report. Furthermore, the user is allowed to view scam awareness tips that teach them how to detect the scam more competently.

The Admin role identified in this diagram, in fact, has more backend management issues at hand. The Admin actor can log in in a manner similar to the user

5.5 Flowchart

A flowchart is a graphical representation of a single step, with each step being understood in turn. Standard symbols like ovals, rectangles, and diamonds are used in flowcharts to illustrate the steps, choices, and results of a process. During system design and analysis, flowcharting can help identify bottlenecks, increase clarity, and improve communication (ASQ, 2025).

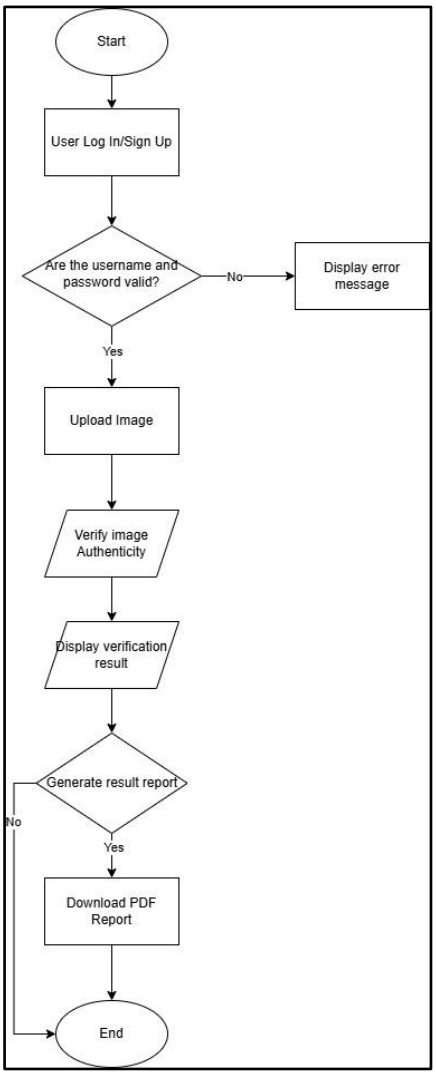


Figure 5.16: Flowchart

According to Figure 5.16, the flowchart provides the ordered step-by-step representation of the main workflow of the system, particularly on the user side. The process begins when the user opts to sign up or log in. Upon inputting credentials, the system checks whether the username and password are valid. In case the input is incorrect, the system will display an error message, and the user will be forced to attempt login again.

If the credentials are okay, the user proceeds to upload an image to be checked by the system through the Verify Image Authenticity process. After verification, the system displays the verification result, i.e., whether the image is reused, manipulated, or original.

Following the results, the user is offered the option to generate a result report. If so, they have a PDF report created and can download it for reference or proof. Otherwise, the process is complete. This flow reflects a clean and logical experience with rational branching and decision points.

The flowchart, as shown by Figure 5.16, keeps things simple in how the user interacts with the core image-checking functions and how the system responds to their inputs. It contributes to the overall goal of scam detection being readily available and convenient for Malaysian users.

5.6 BPMN Diagram

The Business Process Model and Notation (BPMN) system provides a graphical language which standardizes business process modeling into clear visual diagrams. The system uses standard symbols including events (circles) together with activities (rounded rectangles) and gateways (diamonds) and connecting arrows to depict task flow and decision paths inside systems (IBM, 2024). Through its diagrammatic approach BPMN enables developers and business stakeholders to communicate effectively since it offers both technical and easily understandable representations.

BPMN serves as an effective tool to detect process weaknesses while enhancing system design understanding and defining every participant's process role (Wright, 2022). Through process visualization standardization BPMN enables teams to jointly create maps of their work methods which simplifies workflow optimization and dependency detection. The design phase of software development benefits from BPMN because it helps create user-system interaction diagrams that define functional requirements and testing scenarios as well as overall system behavior.

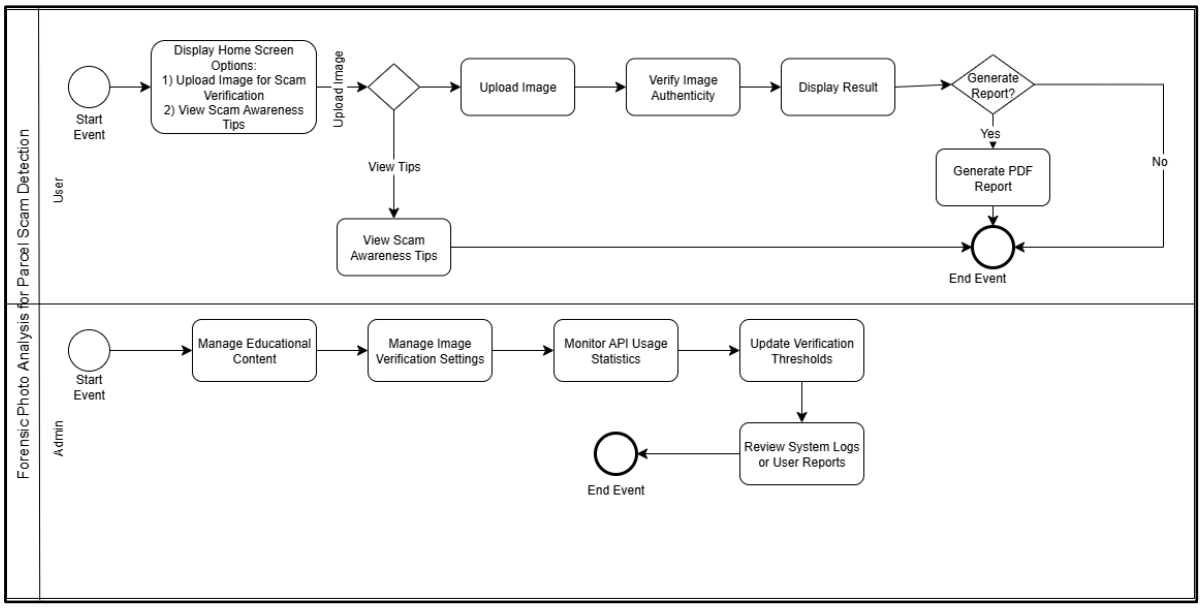


Figure 5.17: BPMN Diagram

The BPMN diagram for the project entitled AI Image Detection for Parcel Scam system, shows the interaction between two actors: User and Admin, depicted vertically in two separate horizontal lanes. This diagram represents the visual model for how the user interacts with the system and how the administrative system responds in the background.

In the User lane, the process starts when the User opens the application. The system presents them with two options; Upload Image for Scam Check or View Scam Awareness Tips. This user decision point is managed via a gateway, that enables the process to flow in two different directions.

If the User selects Upload Image, the system presents a series of navigational menus so that the user can select the image to be analysed. The application will then analyse the image utilizing hash matching or reverse search APIs to check if the image is authentic. The system will calculate the result of the analysis and return whether the image has been recycled, manipulated, or looks safe. The user is offered an option to create a PDF report which summarizes the review process. If the user chooses not to create a report, the workflow concludes and the user is taken to the home screen.

In the alternative, if the user selects View Tips, the user will be taken to scam awareness materials, created by the admin. This educational aspect is intended to raise awareness and in turn, create opportunities to stop and discourage users from falling victim to scams.

In the admin lane, the workflow begins when the admin logs into the dashboard. The admin can perform multiple backend activities, including editing educational materials, adding and editing image verification APIs, viewing logs and statistics about usage of image verification APIs, and modifying detection thresholds in effort to get more accurate results. These activities help to keep the site alive and up to date, and respond to changing scam methods.

This BPMN diagram succinctly shows the individual step in attendance and the various roles present to assist in the maintenance of the system's usability and reliability.

5.7 Conclusion

This chapter employed an extended system requirement breakdown, user interaction, and process workflow with the use of use case and BPMN diagrams. The screenshots accounted for user and administrative functionality in AI Image Detection System for Parcel Scams. Based on models such as flowcharts and BPMN, the system is structured according to user requirements and operating efficiency. These charts form the foundation upon which the system itself is built and guarantee that design choices are made consistent with user expectation and project goals.

6 DESIGN

6.3 Introduction

The design step is a critical step in developing the AI Image Detection System for Parcel Scam, where requirements gathered in the analysis step are translated into sequential blueprints and models that can be installed. The objective of this step is to have all the system details from the interface to database design adequately thought through to meet functional as well as non-functional requirements. According to Talreja (2024), the design phase of the Software Development Life Cycle (SDLC) is a concise plan that stipulates how the system will operate, and usability, security, and efficiency are infused in the process.

In this project, the design is divided into different crucial areas: interface design, database design, security framework, and system workflows. Interface design is carried out with the principles of usability and accessibility in mind so that both non-technical and technical users can use the system very easily. Database designing involves the creation of a data dictionary, data flow diagrams (DFD), and entity relationship diagrams (ERD) for data designing and data organizing efficiently. Doing so, the security framework focuses on the use of controls required such as authentication, access control, and data integrity in an effort to secure the users' data and the uploaded pictures. Lastly, system flow diagrams state how each module functions, providing a pictorial view of how the system operates.

The chapter therefore introduces the plan that will propel the next development phase so that the system is user-friendly, well-structured, and can perform its purported task of detecting reused and manipulated images in parcel scam cases.

6.4 Interface Design

Interface design is the manner in which one plans and structures the graphical component of a system's user interface (UI). It is concerned with the way the application is intended to be utilized by users, navigating within the application, and usability. Coursera (2024) stipulates that UI design is a discipline that is committed to designing interfaces not only good-looking but also very functional and enable users to have easy and effective access to the system. Interface design plays a vital role in facilitating the easier identification of reused and fake parcel scam images in this project.

The interface design simplifies every aspect of the system such as the login section, dashboard, image verification tool, scam awareness information, user manual, and Malaysian

parcel scam case database easy, accessible, and consumable by all users regardless of whether they are technical or not.

The goal is to be straightforward and yet functional enough so that individuals will be able to upload images with ease, access study material, or look for scam case references without being unduly hindered.

By focusing on clean layout, navigability, and responsiveness, the interface design enhances user satisfaction and further improves the usability of the system: offering a reliable and user-friendly AI Image Detection system. This design then ensures that the system maintains consistency with its intended function of scam detection and learning, resulting in an easy-to-use interaction between users and administrators.

6.4.1 Simulation Interface Design

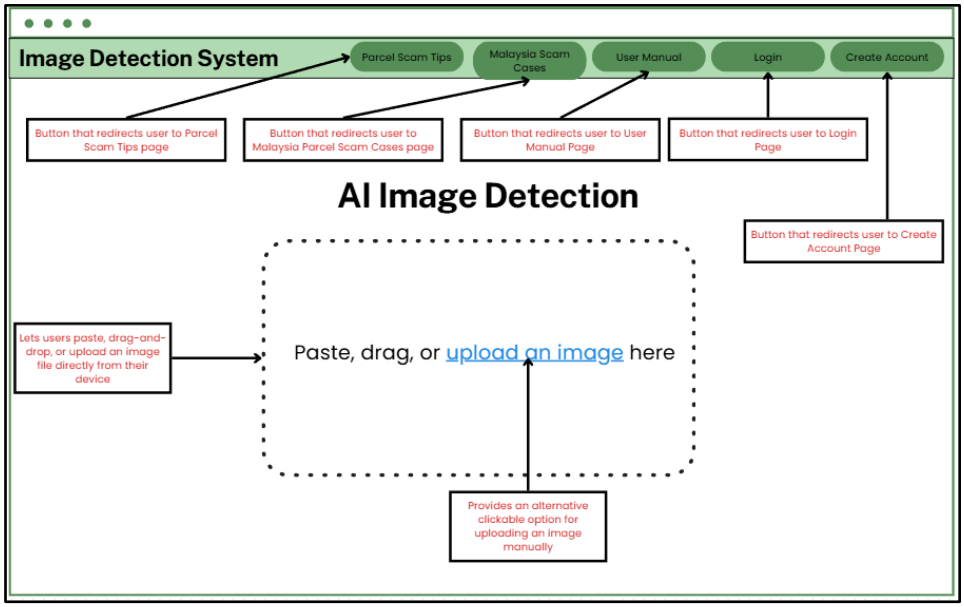


Figure 6.1: Image Detection Page

Figure 6.1 shows the wireframe for the AI Image Detection interface, which is the core of the system's functionality. The upper navigation bar incorporates buttons that direct users to the Parcel Scam Tips, Malaysia Scam Cases, User Manual, Login, and Create Account pages. The core of the interface showcases the drag-and-drop image upload area. Users can easily paste, upload, or simply drag images into the specified detection box. For those who prefer a different approach, a text link offers a manual image selection option. This setup is designed to make things simple for the user, offering both adaptability and ease of use when getting images ready for AI analysis.

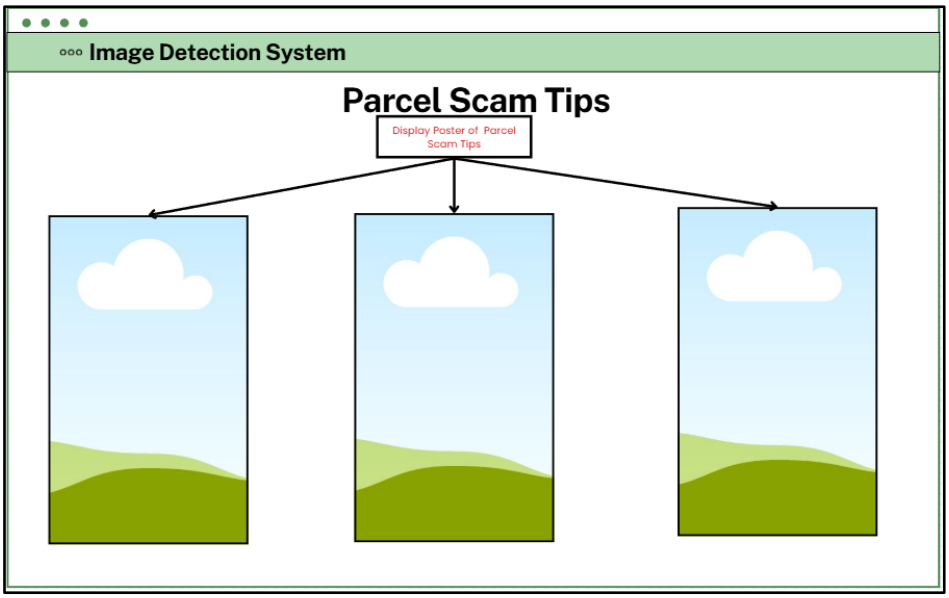


Figure 6.2: Parcel Scam Tips Page

Figure 6.2 presents the Parcel Scam Tips wireframe, which showcases a series of posters intended to inform users about prevalent parcel scam strategies. The design features three visual placeholders, each representing a scam awareness poster sourced from the database. Each poster is interactive, enabling users to access comprehensive scam-prevention details. This configuration facilitates swift navigation through numerous safety recommendations within a visually organized framework. Consequently, this wireframe contributes to the system's awareness component by providing readily accessible and easily navigable scam-avoidance information.

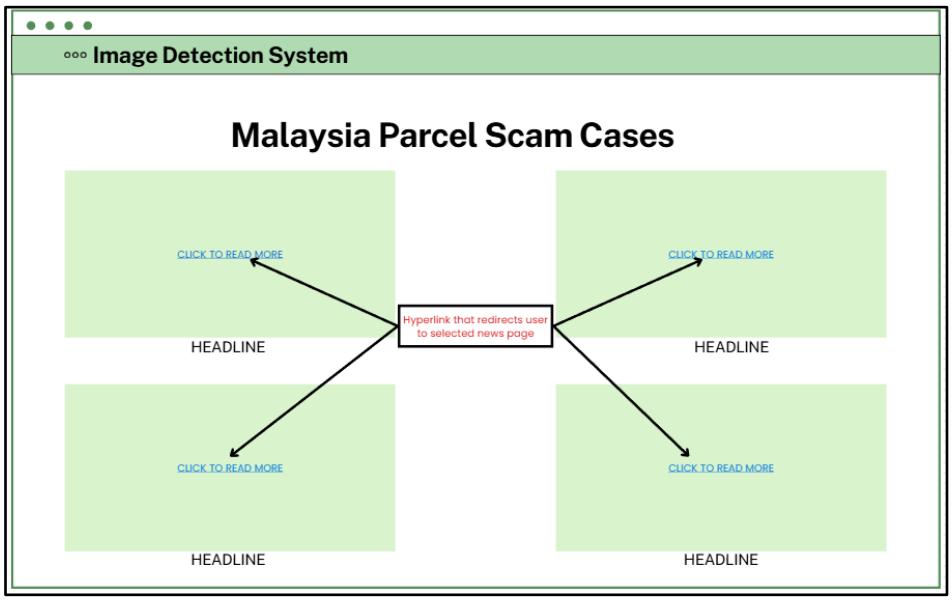


Figure 6.3: Malaysia Parcel Scam Cases Page

Figure 6.3 presents the wireframe for the Malaysia Parcel Scam Cases, illustrating actual Malaysian scam occurrences within a grid layout. Each individual case features a headline and a hyperlink labeled "Click to Read More," which directs users to the complete article. The design prioritizes clarity by displaying four cases concurrently, thereby facilitating rapid browsing of multiple incidents. Consequently, this interface enhances user awareness through the provision of verified scam news, thereby promoting informed and vigilant online conduct.

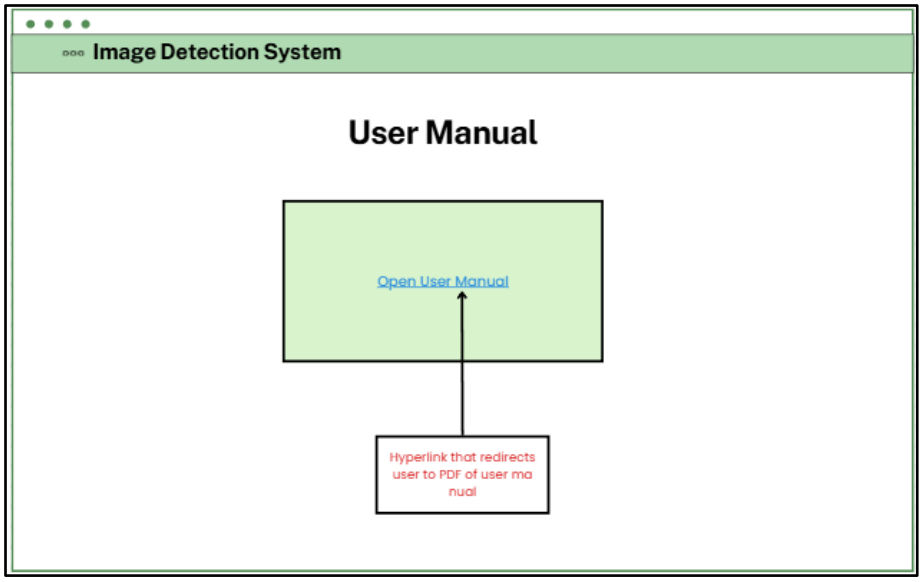


Figure 6.4: User Manual Page

Figure 6.4 presents the User Manual wireframe, designed to furnish users with access to the system's official documentation. A prominent, clickable element labeled "Open User Manual" serves as the gateway to the PDF version of the manual. This wireframe's streamlined approach to instruction access facilitates user comprehension of system functionalities, navigation protocols, and operational directives. Furthermore, the wireframe's uncluttered and minimalist aesthetic underscores its commitment to accessibility and user-friendliness.

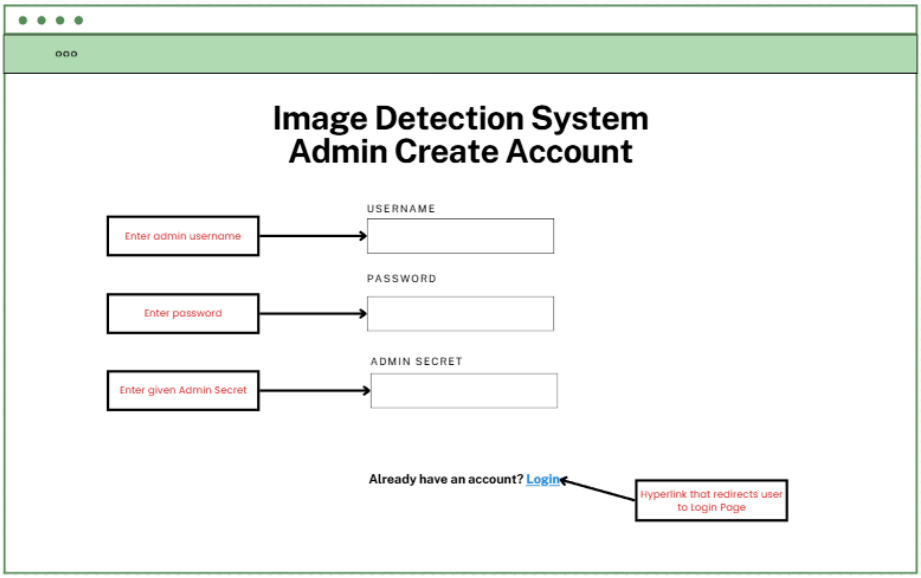


Figure 6.5: Admin Create Account Page

Figure 6.5 presents the Admin Create Account wireframe, which facilitates the creation of new administrative accounts. This interface incorporates input fields for the username, password, and an admin secret key, a crucial element for secure, role-based registration. Furthermore, a hyperlink positioned at the bottom of the interface directs existing administrators to the login page. Consequently, this design fortifies security by mandating a secret code prior to granting administrative access.

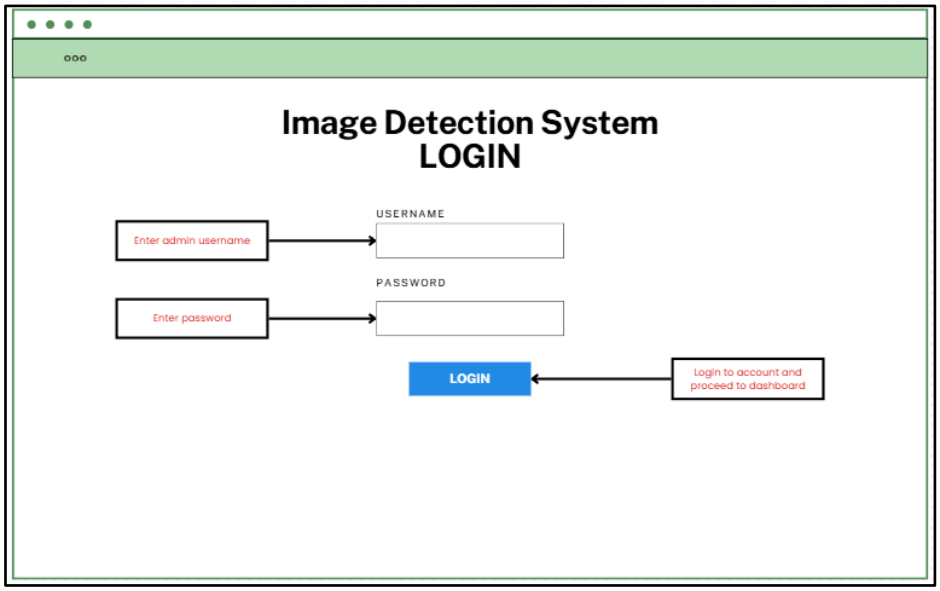


Figure 6.6: Admin Login Page

Figure 6.6 presents the Login Page wireframe, a crucial interface facilitating administrator authentication and dashboard access. The design features clearly delineated username and password input fields, complemented by a conspicuous Login button. This button's function is

to redirect authenticated users to the administrative dashboard. Consequently, this design element is essential for secure access control, thereby guaranteeing that only authorized individuals can manage content or oversee system operations.

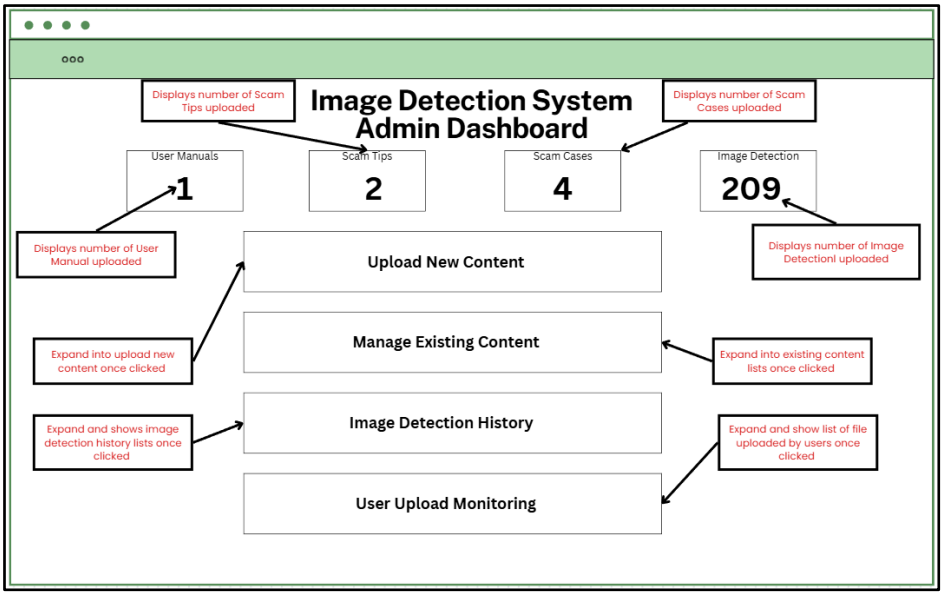


Figure 6.7: Admin Dashboard Page

Figure 6.7 presents the Admin Dashboard wireframe, functioning as the central hub for system administration. The upper segment of the dashboard presents four critical statistics: the count of user manuals, scam tips, scam cases, and uploaded image detections. Beneath these statistics, four primary management modules are accessible: Upload New Content, Manage Existing Content, Image Detection History, and User Upload Monitoring. Upon selection, each module expands to reveal its dedicated management interface. This organized design facilitates efficient content management and real-time user activity monitoring, thereby supporting administrative functions effectively.

6.5 Database Design

Database design serves as the foundational architecture for all software applications, governing the structuring, accessing, and safeguarding of data to facilitate efficient processing. This undertaking encompasses a sequence of logical stages, ultimately leading to the creation of a dependable and robust data structure. A meticulously designed data environment streamlines data administration, upholds data integrity, and supports seamless scalability throughout the system (Lucidchart, 2023).

At the core of this design lies the data dictionary, a metadata repository that delineates tables, attributes, data types, and constraints. This instrument fosters clarity and consistency, thereby

enabling precise implementation and subsequent database maintenance (IBM, 2024). The Entity-Relationship Diagram (ERD) is another crucial artifact; it serves as a visual representation of the interconnections among system entities, including users, verification outcomes, scam alerts, and case documentation. According to Katie Terrell Hanna (2024), ERDs are instrumental in defining relationships between real-world objects and concepts, while also establishing relational database schemas that are closely aligned with system requirements. Complementing the ERD is the Data Flow Diagram (DFD).

While the ERD emphasizes structural aspects, the DFD illustrates the movement of data within the system, detailing inputs, processing stages, storage locations, and outputs. It offers a logical perspective on system processes, thereby governing data flow in accordance with the system's objectives of image verification and scam detection.

The database design for this project necessitates the incorporation of several modules: report generation, teaching material, image examination, user validation, and case scammer storage. Each of these components exhibits clear data interactions and relational definitions. By integrating a data dictionary, an Entity-Relationship Diagram (ERD), and a Data Flow Diagram (DFD), the design ensures data integrity, appropriate module distribution, and secure storage. These preliminary artifacts establish a robust foundation for the stable, scalable, and efficient back-end of the AI Image Detection System.

6.5.1 Data Dictionary

Field Name	Data Type (Size)	Description
id	INT	Primary key; unique auto-increment ID for each detection record.
filename	VARCHAR(255)	Original name of the uploaded image.
image_path	VARCHAR(500)	File path where the image is stored in the server.
is_ai_generated	TINYINT(1)	Stores 1 if image is classified as AI-generated, 0 otherwise.
confidence_percent	DECIMAL(5,2)	Confidence percentage of the classification result.

probability_score	DECIMAL(10,4)	Raw probability score from the AI classifier.
likely_generator	VARCHAR(255)	Name of predicted AI model (e.g., Midjourney, Stable Diffusion); nullable.
explanation	TEXT	Explanation of classification decision; nullable.
user_id	INT	ID of user who uploaded the image (foreign key).
created_at	TIMESTAMP	Timestamp of record creation.

Table 6.1: Data Dictionary of Table "ai_detections"

Table 6.1 presents the ai_detections table's schema, serving as the central storage for all image analysis outcomes produced by the system. This table retains crucial data, encompassing the submitted filename, file path, AI-derived classification, probability score, and confidence percentage. Moreover, supplementary fields like likely_generator and explanation offer a more comprehensive understanding of the classifier's rationale. Each detection is also associated with a particular user via the user_id field, facilitating user-specific detection tracking. The timestamp field confirms the accurate recording of each detection instance.

Field Name	Data Type (Size)	Description
id	INT	Primary key; unique ID for each scam case.
headline	VARCHAR(255)	Title or headline of the news article.
image_path	VARCHAR(500)	Path to thumbnail image; nullable.
news_link	VARCHAR(500)	URL to the external scam news article; nullable.
created_at	TIMESTAMP	Date and time when case was added.

updated_at	TIMESTAMP	Last updated timestamp for the case.

Table 6.2: Data Dictionary of Table "malaysia_cases"

Table 6.2 presents the schema of the malaysia_cases table, which houses confirmed parcel scam cases obtained from credible news sources. Each entry encompasses the headline, an optional thumbnail image, and the external URL directing to the complete article. Furthermore, the table incorporates created_at and updated_at fields, facilitating the monitoring of case update timelines. This table contributes to the system's awareness functionality by furnishing users with concrete instances of scams.

Field Name	Data Type (Size)	Description
id	INT	Primary key; unique ID for each scam tip poster.
title	VARCHAR(255)	Title for the uploaded scam tip poster.
image_path	VARCHAR(500)	File path for the uploaded scam poster image; nullable.
created_at	TIMESTAMP	Timestamp when the record is created.
updated_at	TIMESTAMP	Timestamp when the record is updated.

Table 6.3: Data Dictionary of Table "scam_tips"

Table 6.3 outlines the scam_tips table's design. This table holds informational posters and tips, all aimed at helping people avoid parcel scams. Each entry includes a title, and there's also an optional image path pointing to the location of the tip poster. Timestamp fields are included to assist administrators in monitoring when content is updated. The Scam Tips page within the system pulls its information directly from this table.

Field Name	Data Type (Size)	Description
id	INT	Primary key; unique ID for each admin user.
username	VARCHAR(255)	Username of the admin account.

password_hash	VARCHAR(255)	Hashed password stored securely using bcrypt.
role	VARCHAR(50)	Role of account (admin).
created_at	TIMESTAMP	Account creation timestamp.
updated_at	TIMESTAMP	Last update timestamp.

Table 6.4: Data Dictionary of Table "Users"

Table 6.4 presents the schema of the users table, which is responsible for storing authentication details pertinent to system administrators. This table houses the username, a hashed representation of the password, and the designation of the admin role. Furthermore, the table incorporates timestamps indicating the creation and last update of each account, thereby facilitating effective account management. Consequently, this table is a critical component in ensuring the system's security and regulating administrative access.

Field Name	Data Type (Size)	Description
id	INT	Primary key; unique ID for manual record.
title	VARCHAR(255)	Title of the uploaded user manual.
file_path	VARCHAR(500)	File storage path of the PDF manual.
created_at	TIMESTAMP	Upload timestamp.
updated_at	TIMESTAMP	Last update timestamp.

Table 6.5: Data Dictionary of Table "user_manual"

Table 6.5 presents the structure of the user_manual table, a storage location for PDF manuals designed to assist users in navigating the system. Each record within this table is comprised of a title, a file path, and timestamps that facilitate the tracking of updates. As a result, this table is essential for providing users with convenient access to the most current documentation.

6.5.2 Relationship Matrix

Parent Table	Child Table	Foreign Key	Cardinality	Description

users	ai_detections	user_id	**1 : Many** (One-to-Many)	One user can upload many images for detection, but each detection record belongs to only one user.
users *(implicit)*	malaysia_cases *(no FK)*	-	**No Relationship**	malaysia_cases is managed by admin but does not reference admin ID directly.
users *(implicit)*	scam_tips *(no FK)*	-	**No Relationship**	scam_tips is updated by admin but the table does not store admin reference.
users *(implicit)*	user_manual *(no FK)*	-	**No Relationship**	user_manual entries are uploaded by admin, but no FK is stored.

Table 6.6: Relationship Matrix of AI Image Detection System for Parcel Scams

Table 6.6 shows the Relationship Matrix of AI Image Detection System for Parcel Scams. The relational structure of the Image Detection System is depicted by the relationship matrix, which details the interactions among its constituent tables. Within the existing database schema, the sole explicit relationship is between the users table and the ai_detections table. This association is realized via the user_id foreign key, thereby defining a one-to-many (1:M) relationship. Consequently, a single user can submit numerous images for AI analysis, while each detection record is associated with a single user. This relationship is essential for monitoring user actions, constructing detection histories, and analysing system utilization trends.

The tables malaysia_cases, scam_tips, and user_manual lack foreign key constraints connecting them to the users table. Despite being administered and populated by system administrators, these resources do not have a direct relational link. These tables operate autonomously, serving to provide system content, including scam awareness information, educational resources, and user documentation. Consequently, the established relationship structure preserves system simplicity while ensuring the core functionality of user-detection tracking.

6.5.3 Entity Relational Diagram (ERD)

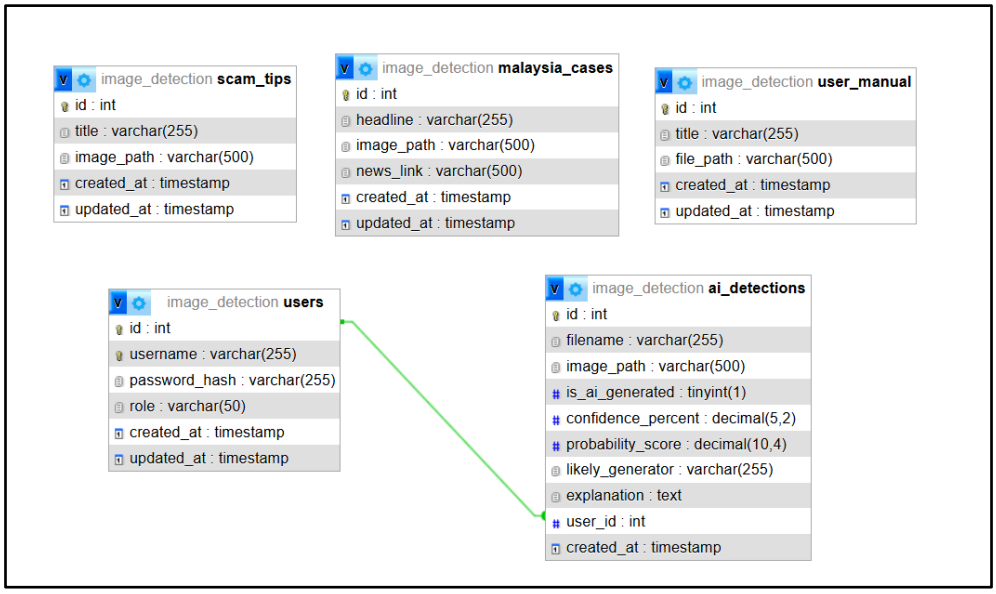


Figure 6.8: Entity Relationship Diagram (ERD) of AI Image Detection System

Figure 6.8 shows the Entity Relationship Diagram (ERD) for the AI Image Detection System, which is designed to identify parcel scams. The ERD illustrates the database's structure and how different entities interact within the system. A key relationship exists between the users table and the ai_detections table; specifically, each user can initiate multiple AI detection requests. This is represented by a one-to-many cardinality, where users.id is the primary key and ai_detections.user_id is the foreign key. This relationship ensures that the detection history is accurately recorded for each verified administrator.

The scam_tips, malaysia_cases, and user_manual tables operate independently, lacking any foreign key relationships. These tables are designed to store publicly available information within the system, including awareness posters, summaries of real parcel scam occurrences, and uploaded user manuals. Their independence highlights their function as static reference data, rather than components of an interactive or relational structure. Consequently, the Entity-Relationship Diagram (ERD) structure offers a clear and efficient database design, distinguishing between functional data, such as AI detections and user information, and informational content, including tips, manuals, and fraud cases. This architectural approach facilitates more efficient maintenance, enhances the transparency of data flow, and streamlines future developmental improvements.

6.4 Security Framework Diagram

The Security System Framework is a technical model that lays out the security controls, practices, and policies implemented within the project environment to protect data, applications, and users. It provides a comprehensive way of securing system functionality, data integrity, and denial of unauthorized access (CyberArk, 2024). Within AI Image Detection system, the framework is concerned with authentication mechanisms, database protection, secure communications channels, and role-based user and administrator access.

With the incorporation of security features within different modules such as login authentication, password recovery, image verification, and case data storage the framework is able to perform all activities within a safe and regulated environment. A well-thought-out security architecture diagram visualizes these security layers with ease, so threats are minimized and sensitive information is securely stored (GoGetSecure, 2024). Ultimately, this structure not only adds to the system's dependability but also establishes user confidence, which is crucial when it comes to digital forensic analysis and fraud detection systems.

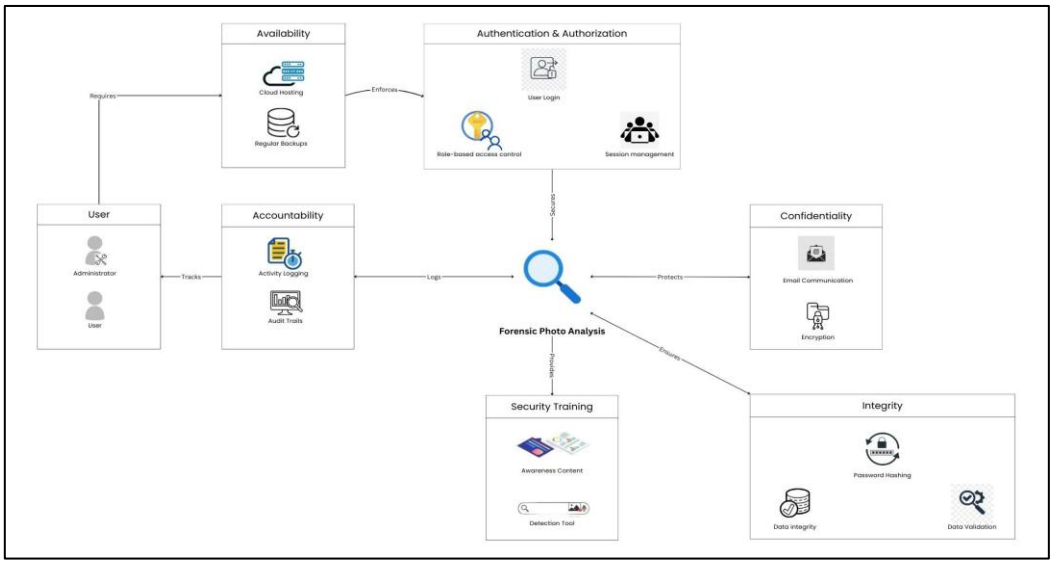


Figure 6.12: Security Framework Diagram of AI Image Detection System

Figure 6.12 is the Security Framework Diagram of the AI Image Detection for Parcel Scam System. Cloud hosting is utilized to ensure system availability, i.e., the system is stable and available for users round the clock. Backups are also included on a periodic basis to secure data and provide recovery options in case of system failure or cyber-attacks. These processes keep the forensic system active at all times and critical reports and evidence do not get misplaced.

Authentication and authorisation form the foundation of security within the system by the enforcement of user identity verification and limited accessibility. Authentic credentials need to be employed for every user to log in, and Role-Based Access Control (RBAC) provides administrators, investigators, and common users to access only functions relevant to their function. Session management also protects against unauthorised access via closing dormant sessions or prompting re-authentication.

Accountability is enforced by traceable large-scale activity tracing and audit trails, which document user activity in the system. Trace abilities and transparencies are provided so that administrators can analyse actions taken during forensic investigations or fraud allegations. Auditing of this sort enhances credibility of the system in courts and enforcement communities.

Encryption and secure communication protocol would prevent interception and unauthorized release of confidential evidence of scams, such as parcel image uploads or forensic analysis, thus rendering the highest confidentiality to victims and keeping the investigations intact.

They are protected against damage and tampering through mechanisms such as database integrity checking, password hashing, and strict data validation. They make sure user data, detection reports, and stored evidence remain correct, unaltered, and resistant to corruption. Hashed passwords also secure user credentials even on attempted intrusion.

Lastly, security training fortifies the system. Customers are given awareness materials as well as training on detection software that equips them to identify scam warning and understanding the forensic verification process. This is a two-pronged approach that addresses both system controls and user training to create an all-encompassing framework that secures the system without precluding users from overcoming parcel scams.

6.5 Flow of the System

A System Flow Diagram shows the logical flow of processes and data transfer within a system graphically. It shows how the tasks, inputs, decisions, and outputs are related to each other to assist in achieving the overall goals of the system. By viewing the workflow laid out in an easily comprehensible way, a system flow diagram makes it easy to see how the various components of the system are related and contribute towards the end product (Lucidchart, 2025).

Standard notations such as rectangles for procedures, diamonds for decisions, arrows for direction of travel, and parallelograms for inputs and outputs are generally used to offer clarity and uniformity. This enables stakeholders such as developers, administrators, and end users to quickly read the design without any ambiguity.

For this AI Image Detection, the System Flow Diagram explains the primary modules such as user login, image uploading and validation, tips for scam awareness, and interactions with the database. It also explains how the users and administrators are interacting with the system and how the processes are started based on user input. By graphically illustrating such kinds of workflows, the diagram ensures that functionality and controls for security are commensurate with the aims of the project. Lastly, the System Flow Diagram is a blueprint that guides developers in implementing while enabling users and administrators to understand the order of logical activities in the system.

6.5.1 System Flow Diagram User

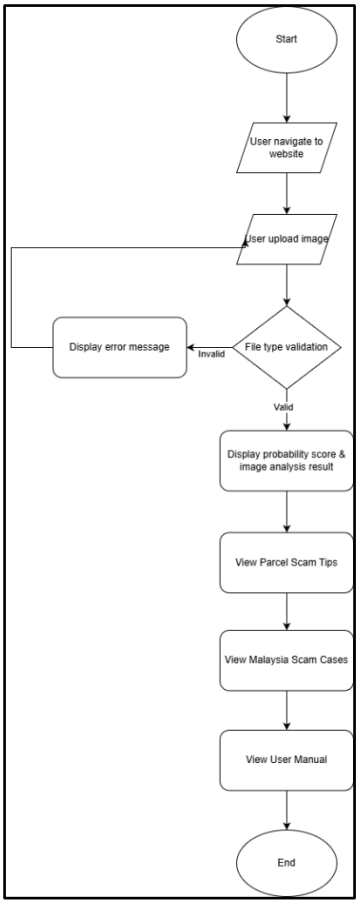


Figure 6.13: System Flow Diagram User

The User System Flow Diagram illustrates the interaction sequence of a standard user engaging with the AI Image Detection System to assess parcel-related images. The procedure commences at the Start node, subsequently leading the user to Navigate to the Website. Upon accessing the platform, the user then Uploads an Image for verification. The system undertakes File Type Validation to ascertain the appropriateness of the upload format, including formats such as JPG or PNG. Should the file format be deemed invalid, an error message is presented, and the user is redirected to the upload interface. Conversely, if the file is validated successfully, the system processes the image and subsequently Displays the Probability Score & Image Analysis Result, thereby enabling users to examine the AI-generated detection output. This encompasses the likelihood of content being AI-created, altered, or repurposed. Once the results are in, users can explore educational materials. They can access Parcel Scam Tips, which offer guidance on recognizing scam images and raising awareness. Users can also review Malaysia Scam Cases, gaining insight into actual incidents to better understand local scam patterns. A User Manual will be available to help users navigate the system effectively. The user's process concludes at the End point, marking the completion of all interactions.

6.5.2 System Flow Diagram Admin

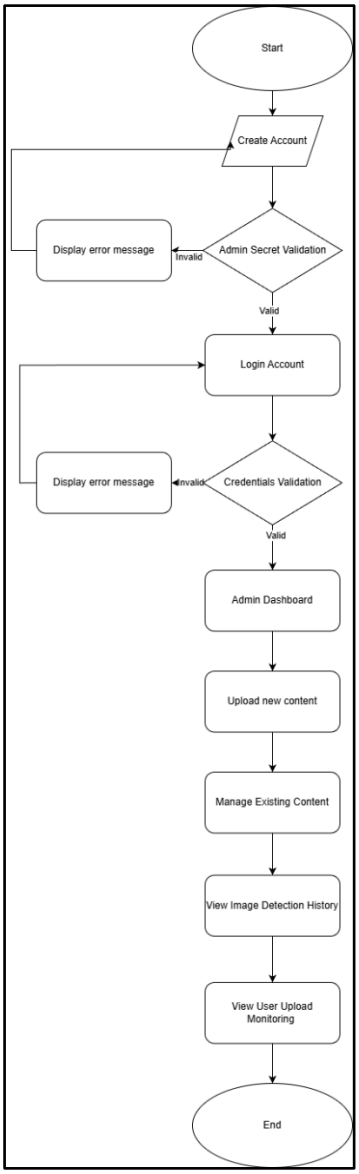


Figure 6.14: System Flow Diagram Admin

The Admin System Flow Diagram depicts the sequenced interaction that an administrator goes through in managing or accessing the AI Image Detection System. The process commences at the Start, where the admin proceeds to the Create Account step. In this step, the system requires validation of the Admin Secret Key to ensure that only authorized persons may create administrative accounts. If the secret key is incorrect, the system prompts an error message and loops back into account creation. Once validated, the admin proceeds with the Login Account phase.

The system validates credentials on login. In case of invalid credentials, the system shows an error message that forces an admin to retry. At the time of successful authentication, an admin will then be routed to the Admin Dashboard, which serves as the main control interface.

The administrator can then proceed to perform several management functions from the dashboard, including uploading new content, like scam tips, Malaysia scam cases, or user manuals. The admin can then proceed under Manage Existing Content to update or delete previously uploaded materials.

Moreover, the admin is allowed to view Image Detection History, which shows all the scanned images uploaded by users and results. This includes View User Upload Monitoring, whereby the admin can monitor the activity of the users and ensure that the platform is used appropriately.

The sequence ends at the End point, where the administrative workflow finishes.

6.6 Conclusion

The design phase of the AI Image Detection System for Parcel Scams, as detailed in Chapter 6, is summarized here. This phase was crucial in transforming user requirements and system objectives into structured models, diagrams, and visual interfaces, all of which were essential for the subsequent development. The chapter specifically addressed system architecture, database design, user interface layout, and process flow diagrams for both user and admin modules, thereby facilitating logical planning prior to implementation. System flow diagrams provide users and administrators with a visual representation of the platform's operational processes, encompassing the entire workflow from image upload and validation through content management and monitoring. Consequently, these models render every system interaction predictable, functional, and readily understandable.

To facilitate usability for individuals lacking technical expertise, the user interface (UI) prioritizes accessibility, clarity, and simplicity. The system's design streamlines the processes of image uploading, scam tip retrieval, and the examination of authentic Malaysian scam cases, achieved through intuitive navigation and well-structured pages. Furthermore, the administrative interface was designed for efficiency, thereby simplifying content updates and system monitoring. This chapter concludes the system's conceptualization by presenting detailed design models and structured diagrams. The design phase serves as a crucial link between the initial requirements

and the subsequent implementation, providing developers with a clear, precise, and thoroughly documented blueprint. Consequently, the establishment of a functional, secure, and user-friendly system for detecting manipulated or reused parcel scam images within Malaysia is predicated on this foundational work.

7 IMPLEMENTATION

7.3 Introduction

The implementation phase is where system design transitions to an operational solution, in which all features planned in the design are developed, integrated, and prepared for operation. At this stage, technical specifications were transformed into actual and working components through organized activities of development, configuration, and deployment (Itexus, 2024). In developing the AI Image Detection System for Parcel Scams, it entailed constructing the core detection engine, integrating the user and admin interfaces, configuring the database, and ensuring smooth interaction among the front-end and back-end modules. This chapter highlights the platforms, tools, interfaces, and security elements used along the course of development. It also gives a glimpse into how each module will be built so that the system should work reliably and efficiently and meet the functional requirements defined earlier.

7.4 Execution Platform

Execution platform means the hardware and software environment in which system or application is developed, tested, and run. Platform on which software will be executed and works fine. The platform of execution can be a particular operating system, programming language, server, or cloud infrastructure in which development teams can develop and deploy the software.

7.4.1 Development Platform

Figure 7.1: Windows 11 (Muchmore, 2025)

Figure 7.1 shows Windows 11, the primary operating system used throughout the development of the AI Image Detection System for Parcel Scams. Windows 11 provides a

stable and modern environment that supports essential development tools such as Python, Flask, MySQL, phpMyAdmin, and Visual Studio Code. Its optimized performance, improved security features, and enhanced multitasking capabilities made it suitable for running local servers, handling database operations, and testing system features. The operating system also ensured seamless integration with external APIs and supported virtual environments and required libraries. Overall, Windows 11 served as a reliable foundation that allowed the entire system to be developed and executed efficiently.

7.4.2 IDE for Backend and Frontend Development

Figure 7.2: Visual Studio Code (Canonical, 2019)

Figure 7.2 shows Visual Studio Code (VS Code), the main integrated development environment (IDE) used to build both the backend and frontend components of the system. VS Code offers a clean interface and valuable features such as IntelliSense, syntax highlighting, Git integration, and an integrated terminal, all of which improved development speed and accuracy. Extensions such as Python, MySQL, and Prettier further enhanced productivity by assisting with debugging, formatting, and database interaction. The IDE was used to manage the entire project structure, including Python route files, HTML templates, CSS styling, JavaScript scripts, and configuration files. VS Code's lightweight yet powerful functionality made it ideal for continuous testing and development of the application.

7.4.3 Data Storage and Management

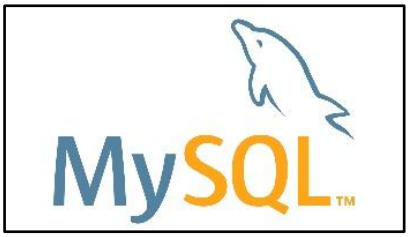


Figure 7.3: MySQL (Jackson, 2025)

Figure 7.3 shows MySQL, the relational database management system used to store and manage all system data. MySQL handled essential records such as users, image detection results, scam tips, Malaysia scam cases, and user manuals. Its structured table format, fast query execution, and reliability made it well-suited for supporting the system's backend operations. The database interacted with Python through the MySQL Connector library, enabling secure and efficient data retrieval and storage. phpMyAdmin was used alongside MySQL to visualize database tables and perform administrative tasks such as editing records, creating tables, and monitoring data integrity. MySQL played a crucial role in ensuring the system operated smoothly with consistent and accurate data management.

7.5 Implementation Tools

Implementation tools are the hardware and software used in the development, operation, and testing of the AI Image Detection System for Parcel Scams. This is due to the fact that these tools are necessary for the development process to be both successful and function as intended. The chosen tools enable many project phases, including coding, simulation, testing, deployment, etc.

7.3.1 Hardware

Figure 7.4: Acer Aspire A315-24P (Knapp, 2023)

Figure 7.4 shows the model of the laptop for the development of AI Image Detection System for Parcel Scams, Acer Aspire A315-24P. Table 7.1 shows the hardware specification of the development laptop.

Computer Brand	:	Acer Aspire A315-24P
Processor	:	AMD Ryzen 5 7520U with Radeon Graphics, 2.80 GHz
Memory (RAM)	:	8 GB RAM
System	:	64 bit operating system
Operating System	:	Windows 11
Internal Disk Storage	:	256 GB

Table 7.1: Hardware Specification

Figure 7.4 shows the Acer Aspire A315-24P, mainly used as a development device in the course of implementing the proposed AI Image Detection System for Parcel Scams. Equipped with an AMD Ryzen 5 7520U processor, integrated Radeon graphics, 8 GB of RAM, and a 256 GB SSD, this laptop allowed performing heavy tasks with ease: running several heavy Python scripts, maintaining MySQL databases, and running local development servers. The 64-bit Windows 11 operating system guaranteed the workability of all needed software tools, including but not limited to VSCode, Laragon, MySQL, and Python libraries. Considering its hardware, this device is about general performance of complex tasks, including multitasking, file management, backend testing, and dataset handling.

7.3.2 Software

Figure 7.5: Visual Studio Code (Canonical, 2019)

Figure 7.5 shows Visual Studio Code, the main code editor used to write and manage the system's source files. Its features such as IntelliSense, syntax highlighting, an integrated terminal, and built-in Git support made the development process efficient. VS Code extensions like Python, MySQL, and Live Server enhanced the development environment, allowing seamless debugging and testing.

Figure 7.6: Laragon (Zaman, 2025)

Figure 7.6 shows Laragon, the local development environment used to run MySQL services and manage server operations required by the system. Laragon provided a portable and fast workspace with built-in database tools, making it easy to start and manage MySQL without manual configuration. Its stable services ensured smooth backend testing throughout the project.

Figure 7.7: phpMyAdmin (Dobry, 2025)

Figure 7.7 shows phpMyAdmin, the graphical tool used to manage the MySQL database. It allowed easy creation of tables, modification of records, and execution of SQL queries. The interface provided a visual overview of the *image_detection* database, making it easier to manage stored data and verify the system's database structure during development.

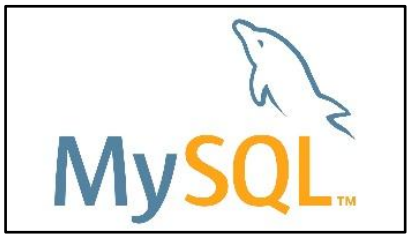


Figure 7.8: MySQL (Jackson, 2025)

Figure 7.8 shows MySQL, the relational database used to store system data such as user accounts, detection results, scam tips, and case information. MySQL offered high performance, structured querying, and strong reliability, making it suitable for handling the

system's data operations. It worked seamlessly with Python through MySQL Connector, ensuring fast and secure data retrieval.

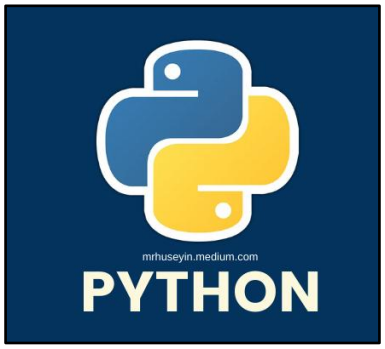


Figure 7.9: Python (Huseyin, 2021)

Figure 7.9 shows Python, the primary programming language used to develop the backend of the AI Image Detection System for Parcel Scams. Python was chosen for its simplicity, extensive library support, and efficiency in building web applications. It enabled the development of core functionalities such as routing, authentication, image analysis, and database operations. Its clean syntax and flexibility made the implementation process smoother and more maintainable, especially for handling file uploads and integrating external APIs.

Figure7.10: Flask (Paul, 2023)

Figure 7.10 shows Flask, the lightweight web framework used to build the server-side logic of the system. Flask provided a simple yet powerful structure for defining routes, handling requests, and serving HTML templates. Its blueprint system allowed the project to be organized into multiple modules (authentication, admin, content, AI detection), enhancing maintainability. Flask's flexibility made it ideal for connecting the frontend with backend logic while supporting secure operations.

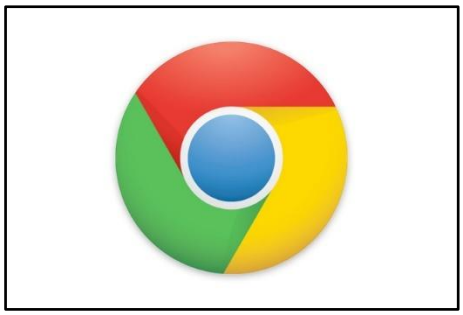


Figure 7.11: Chrome (Warren, 2017)

Figure 7.11 shows Google Chrome, the browser used to access and test the system through localhost:4000. Chrome provided a stable environment for previewing the user interface, checking responsiveness, and testing system features such as image uploads, admin login, and content management. Its developer tools were helpful for inspecting frontend behaviour and troubleshooting issues.

7.6 Program Interface

The interface of the program defines all the visual elements that enable users and administrators to interact with the AI Image Detection System for Parcel Scams. It includes layout, navigation menus, buttons, forms, display of content, and feedback messages across displays on the system. A well-drafted interface ensures that users easily execute tasks such as the upload of images, reading tips on scams, accessing manuals, and reviewing detection results. For admin users, the program interface also plays a key role in guiding them to efficiently manage content, monitor uploads, and oversee system operations. This section illustrates, through screenshots, each major interface in the system, describing their functions, appearance, and role in supporting the overall usability of the system. Accordingly, each interface was made simple, modern, and intuitive to guarantee an excellent user experience.

7.4.1 User System Interface

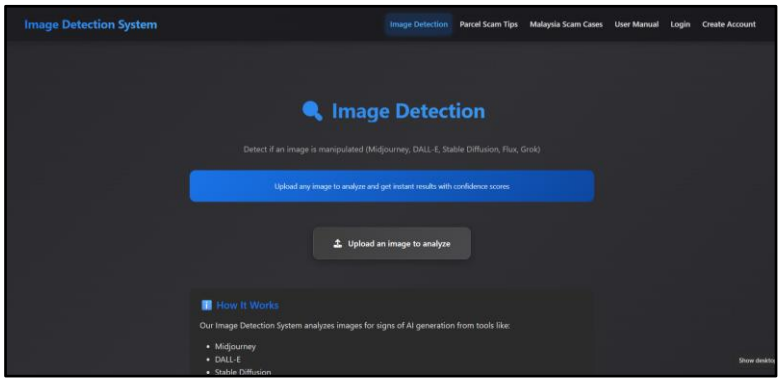


Figure 7.12: Image Detection Page

Figure 7.12 shows the Image Detection page. The Image Detection page serves as the main interface where users begin the image verification process. The page features a clean and modern layout with a clearly highlighted “Upload an Image to Analyze” button placed at the center to guide users toward the primary system function. A short description explains the purpose of the platform, indicating that the system detects whether an image is AI-generated or real. The navigation bar at the top provides access to other pages such as scam tips, scam cases, the user manual, login, and account creation. The overall design emphasizes simplicity and usability, ensuring that users can quickly understand how to perform image detection without requiring prior technical knowledge.

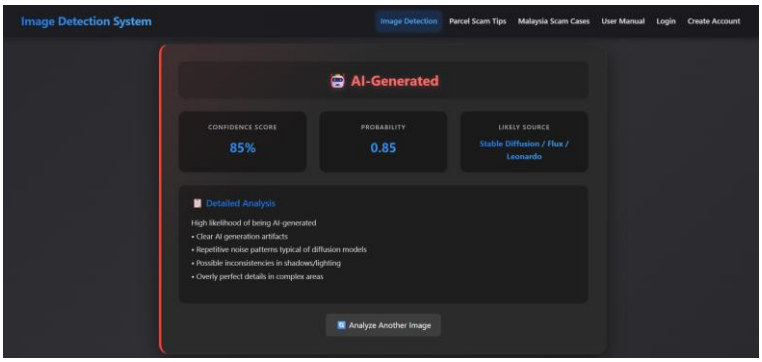


Figure 7.13: Image Analysis Page

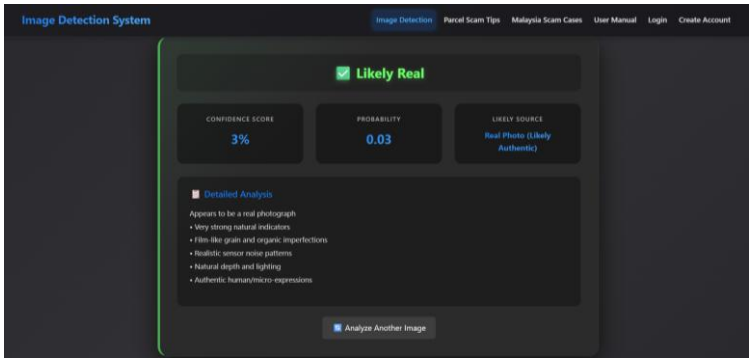


Figure 7.14: Image Analysis Page

Figure 7.13 and 7.14 Shows the Image Analysis page. The Image Analysis page displays the system's detection results after a user uploads an image. Whether the image is classified as AI-Generated or Likely Real, the interface presents a structured summary that includes confidence score, probability value, and the likely image source or generator. The result is visually emphasized through colour indicators red for AI-generated images and green for real images allowing users to interpret outcomes quickly. A detailed analysis section provides additional insights such as AI artifacts, lighting inconsistencies, or natural image characteristics. This helps users understand *why* the system reached its conclusion. The page ends with an "Analyse Another Image" button, enabling smooth navigation for continuous checking of multiple images.

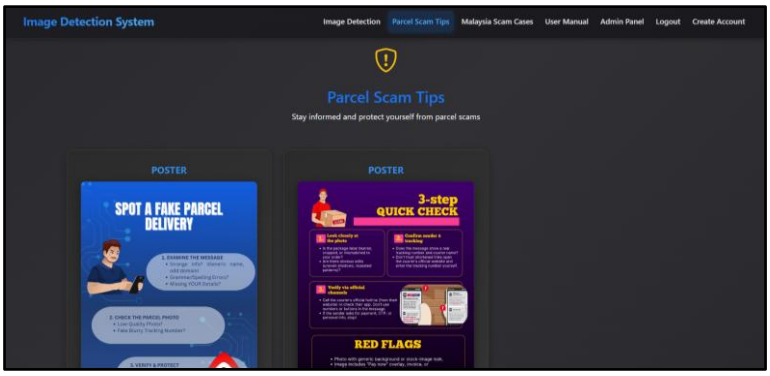


Figure 7.15: Parcel Scam Tips Page

Figure 7.15 shows the Parcel Scam Tips Page. The Parcel Scam Tips page provides educational content in the form of visually appealing posters designed to raise awareness about common parcel scam strategies. Each poster highlights essential checks users should perform, such as inspecting suspicious messages, verifying parcel photos, and recognizing fake tracking numbers. The posters are displayed in neatly arranged card layouts, making them easy to browse and read. This interface serves as a preventive tool by equipping users with the knowledge needed to identify scam attempts before falling victim. Its design is

informative yet simple, allowing users of all ages and backgrounds to benefit from the awareness materials.

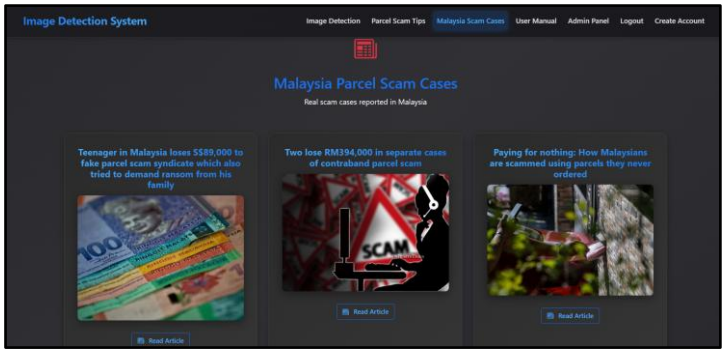


Figure 7.16: Malaysia Scam Cases Page

Figure 7.16 shows the Malaysia Scam Cases Page. This interface showcases real parcel scam incidents reported in Malaysia. Each case is represented through a card containing the article title, a relevant image, and a “Read Article” button. The intention is to expose users to genuine scam scenarios so they can better understand how fraudsters operate. By reading verified news reports and documented cases, users gain practical awareness of the techniques used in real scams, such as fake customs charges, impersonation of couriers, and ransom attempts. The interface is organized to encourage users to explore multiple cases at their own pace, making it an effective awareness-learning platform.

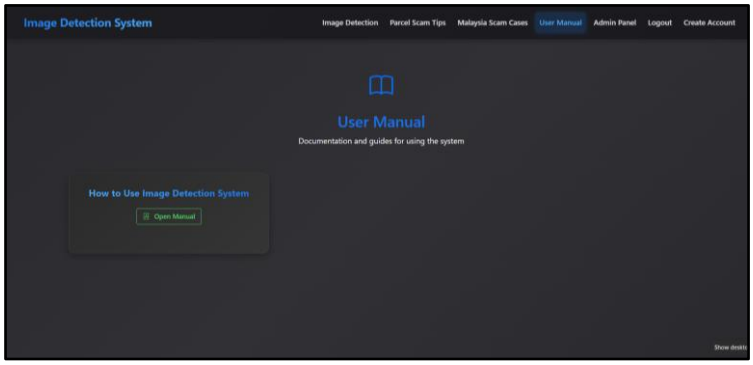


Figure 7.17: User Manual Page

Figure 7.17 shows the User Manual Page. The User Manual page provides users with direct access to the system’s official documentation. A simple, centered card displays an “Open Manual” button, allowing users to download or view the PDF manual containing step-by-step instructions, feature explanations, and usage guidelines. This ensures that users can easily obtain help or clarification whenever needed. The interface is intentionally kept minimal to maintain clarity and focus on the manual access function. This page strengthens user support and contributes to a smoother system experience, especially for first-time users.

7.4.2 Admin System Interface

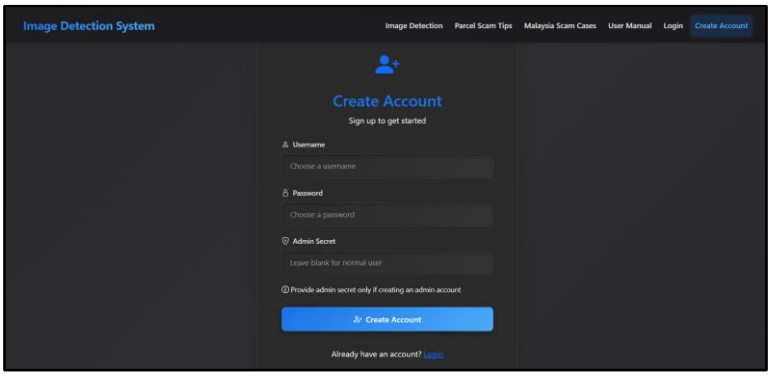


Figure 7.18: Admin Create Account Page

Figure 7.18 shows the Admin Create Account Page. Thi page is used exclusively for setting up administrator access. The interface provides three main input fields: username, password, and an optional *Admin Secret* key. The Admin Secret feature ensures that only authorized individuals can create an administrative account, adding an extra layer of security. The layout is simple and user-friendly, allowing administrators to register efficiently without unnecessary steps. Guidance text is included below the Admin Secret field to remind users that the key should only be provided when creating admin-level accounts. This interface ensures that administrative privileges are securely controlled and restricted from general users.

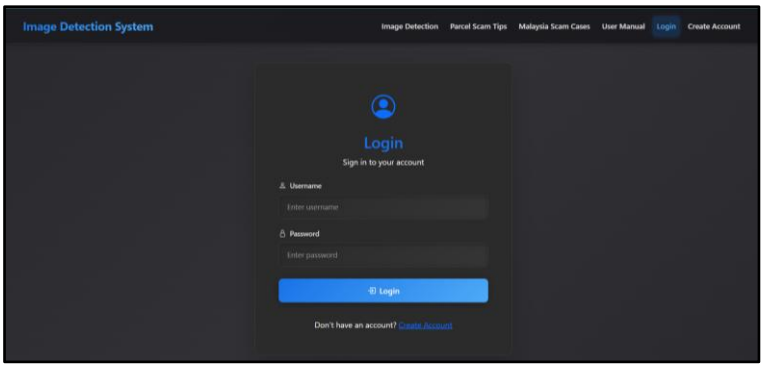


Figure 7.19: Admin Login Page

Figure 7.19 shows the Admin Login Page. The Login page enables administrators to access the backend system by entering their registered username and password. The interface is designed with a clean card layout that emphasizes clarity and focus, helping administrators sign in without distractions. The page reinforces secure access by requiring valid credentials before granting entry to the Admin Dashboard. A link to the Create Account page is provided for situations where new administrators need to register. This interface plays a crucial role in

restricting backend access, ensuring that only authorized personnel can manage system content, monitor user uploads, and oversee detection records.

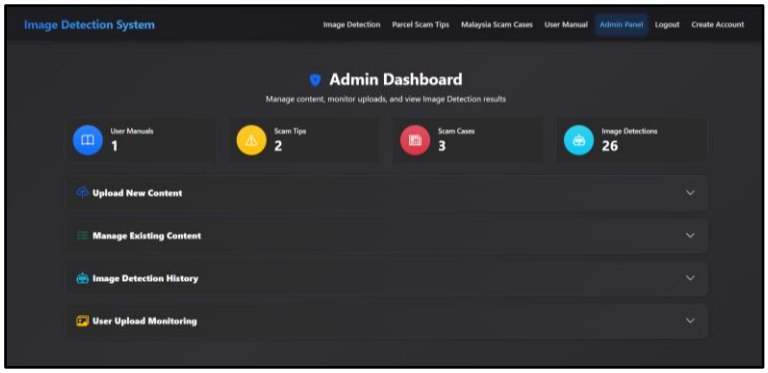


Figure 7.20: Admin Dashboard Page

Figure 7.20 shows the Admin Dashboard page. The Admin Dashboard serves as the central control panel for system administrators. It provides a quick overview of the system's current status through summary cards showing the number of user manuals, scam tips, scam cases, and total image detection records. The dashboard is designed with a collapsible menu style, allowing administrators to expand or hide sections such as content upload, content management, detection history, and user upload monitoring. This interface ensures that administrators have immediate access to all essential system management tools in one place. The dashboard's structured layout supports efficient workflow and enables administrators to oversee the system's activities without navigating through multiple pages.

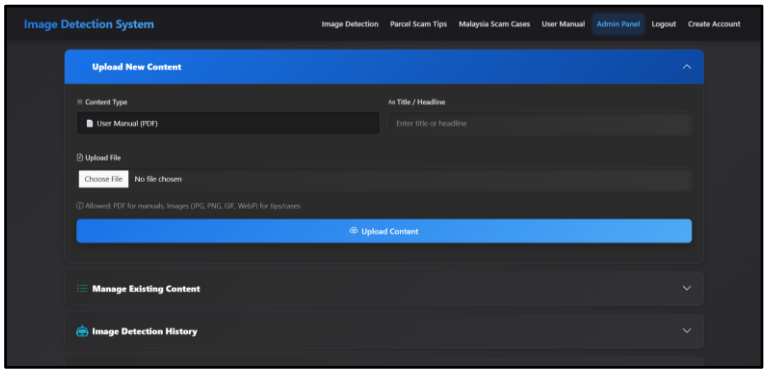


Figure 7.21: Upload Content Page

Figure 7.21 shows the Upload Content Page. This interface enables administrators to upload new materials such as user manuals, scam tips, or scam case images. A dropdown menu is provided for selecting the content type, followed by input fields for the title or headline. The file upload section supports PDF documents and images, depending on the selected category. Clear instructions are included beneath the input fields to guide administrators on acceptable

file formats. The interface is designed for ease of use, ensuring that new content can be added consistently and accurately. By providing a straightforward upload process, this page helps administrators keep the system's awareness resources updated.

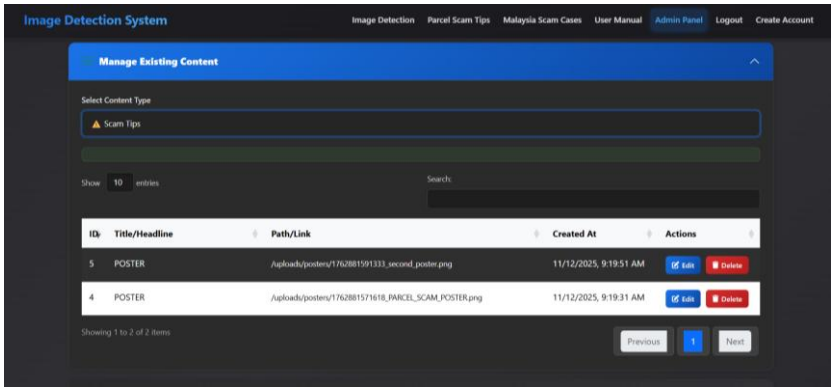


Figure 7.22: Manage Existing Content Page

Figure 7.22 shows the Manage Existing Content Page. The Manage Existing Content page allows administrators to view, edit, and delete previously uploaded materials. Content items are displayed in a table format that includes the ID, title, file path, and creation date. Each entry is equipped with "Edit" and "Delete" buttons for quick modification. A search bar is available at the top of the table, enabling admins to filter through content efficiently, while pagination at the bottom supports navigation through multiple entries. This interface ensures systematic and organized content management, allowing administrators to maintain accuracy and relevance across all awareness materials.

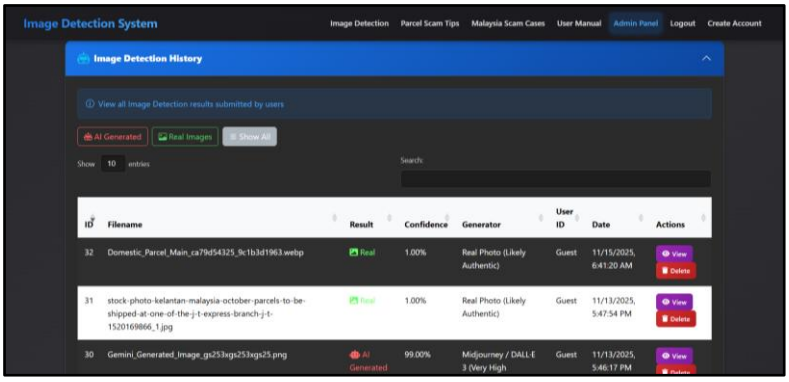


Figure 7.23: Image Detection History Page

Figure 7.23 shows the Image Detection History Page. This interface records all image analyses performed by users. The table displays essential information such as the filename, detection result, confidence score, generator type, user ID, and timestamp. The page includes filtering options that allow administrators to view only AI-generated or real images, as well as

a “Show All” filter for complete records. Each entry includes a “View” button for detailed inspection and a “Delete” button for removing unnecessary or inappropriate uploads. This page plays a vital role in monitoring system usage and verifying the reliability of detection outputs.

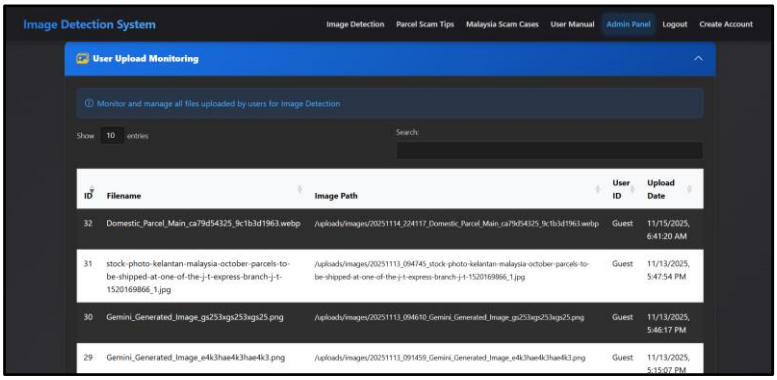


Figure 7.24: User Upload Monitoring Page

Figure 7.24 shows the User Upload Monitoring Page. The User Upload Monitoring page shows a list of images uploaded by users before analysis. Each entry includes the file name, file path, upload date, and user ID. The search function allows administrators to locate specific uploads, and the table format provides a clear and organized overview of all submissions. This interface supports administrative oversight by enabling the review of user-uploaded images, ensuring that the system is being used responsibly. It is particularly useful for identifying suspicious uploads or analysing user behaviour patterns.

7.7 Database Configuration

This section describes how the AI Image Detection System for Parcel Scams database is set up and managed. MySQL has been used as the main database, as it comes off as reliable, capable of structured querying, and also Python-based application-friendly. Database configuration should be defined in a dedicated Python file where key parameters, host, port, username, password, and database name will be stored in a configuration dictionary. Security was further enhanced by using environment variables that avoid sensitive credentials from being directly exposed in the code. The connection function handles the communication with the MySQL server, while the initialization function will make sure that the needed database and tables are created if they don't exist. Management of database tables is visual via phpMyAdmin: ai_detections, scam_tips, malaysia_cases, users, user_manual. This will ensure data handling and storage integrity for smooth and efficient system operations.

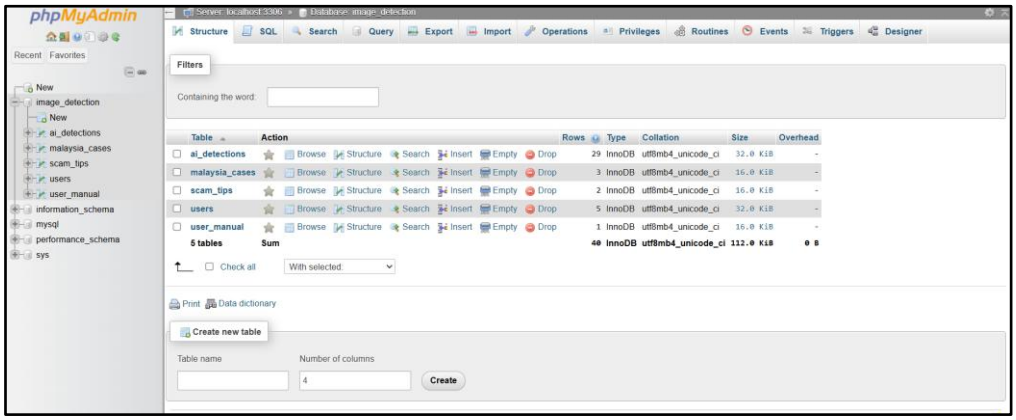
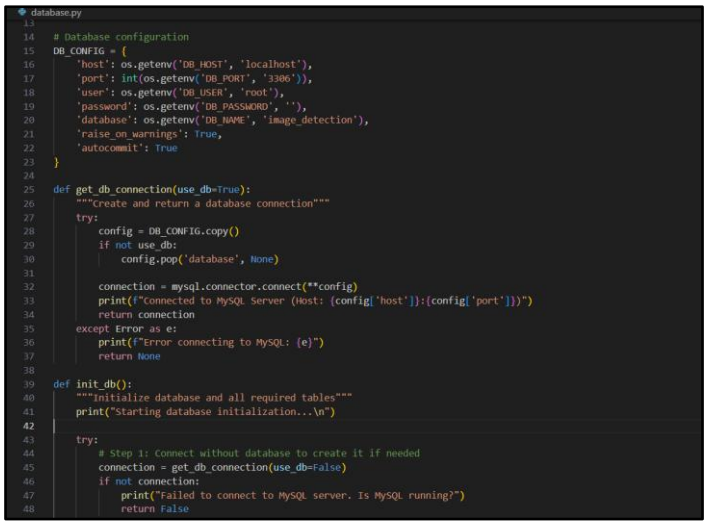


Figure 7.25: MySQL Database Interface (phpMyAdmin)

Figure 7.25 shows the MySQL database interface accessed through phpMyAdmin, which was used to manage and visualize the database for the AI Image Detection System for Parcel Scams. The interface displays the image_detection database along with its five main tables: ai_detections, malaysia_cases, scam_tips, users, and user_manual. Each table contains structured data relevant to system operations, such as detection history, uploaded awareness materials, scam case information, and administrator login records. phpMyAdmin provides features such as browsing data, editing rows, running SQL queries, exporting tables, and modifying table structures. This visual environment simplifies database management and allows easier monitoring of stored data compared to command-line tools. Overall, the interface plays an essential role in ensuring that all system data is maintained accurately and efficiently.



```python
# Database configuration
DB_CONFIG = {
    'host': os.getenv('DB_HOST', 'localhost'),
    'port': int(os.getenv('DB_PORT', '3306')),
    'user': os.getenv('DB_USER', 'root'),
    'password': os.getenv('DB_PASSWORD', ''),
    'database': os.getenv('DB_NAME', 'image_detection'),
    'raise_on_warnings': True,
    'autocommit': True
}

def get_db_connection(use_db=True):
    """Create and return a database connection"""
    try:
        config = DB_CONFIG.copy()
        if not use_db:
            config.pop('database', None)

        connection = mysql.connector.connect(**config)
        print(f"Connected to MySQL Server (Host: {config['host']}:{config['port']})")
        return connection
    except Error as e:
        print(f"Error connecting to MySQL: {e}")
        return None

def init_db():
    """Initialize database and all required tables"""
    print("Starting database initialization...\n")

    try:
        # Step 1: Connect without database to create it if needed
        connection = get_db_connection(use_db=False)
        if not connection:
            print("Failed to connect to MySQL server. Is MySQL running?")
            return False
```

Figure 7.26: Database Configuration Code Snippet

Figure 7.26 shows the database configuration code implemented in Python to establish a connection between the system and the MySQL server. The configuration begins with the **DB_CONFIG** dictionary, which stores essential connection parameters including host, port, username, password, and database name. These values are retrieved using environment variables to enhance security and protect sensitive credentials. The function **get_db_connection()** is responsible for creating and returning a MySQL connection object, while handling errors gracefully if the server is unreachable. Meanwhile, the **init_db()** function initializes the database and ensures that required tables exist before the system runs. This configuration snippet forms the backbone of the system's data layer, ensuring stable communication with MySQL and enabling smooth execution of all database-related operations.

7.6 Security Elements

This section describes the security components implemented in the AI Image Detection System for Parcel Scams to ensure safe authentication, protected access, secure file handling, and proper safeguarding of sensitive information. Several backend mechanisms were developed to protect user data, restrict administrative privileges, validate uploads, and prevent unauthorized access. Each security element is supported by Python-based implementations using Flask, JWT, bcrypt, environment variables, and database-level validation. The following figures highlight the main security techniques used in this system, along with code snippets extracted from the actual backend.

7.6.1 JWT Authentication and Protected Routes

```python
def token_required(f):
    """Decorator for protected routes"""
    @wraps(f)
    def decorated(*args, **kwargs):
        token = None

        # Get token from header
        if 'Authorization' in request.headers:
            auth_header = request.headers['Authorization']
            try:
                token = auth_header.split(' ')[1]  # Bearer <token>
            except IndexError:
                return jsonify({'message': 'Invalid token format'}), 401

        if not token:
            return jsonify({'message': 'Token is missing'}), 401

        try:
            data = jwt.decode(token, SECRET_KEY, algorithms=['HS256'])
            current_user = data
        except jwt.ExpiredSignatureError:
            return jsonify({'message': 'Token has expired'}), 401
        except jwt.InvalidTokenError:
            return jsonify({'message': 'Invalid token'}), 401

        return f(current_user, *args, **kwargs)

    return decorated
```

Figure 7.27: JSON Web Token (JWT) authentication

Figure 7.27 shows the implementation of JSON Web Token (JWT) authentication, which ensures that only authenticated users may access protected functionalities. The token_required decorator validates the presence of a token, checks whether it is expired or malformed, and extracts user information from the payload. This prevents unauthorized individuals from directly accessing backend routes through manual URL invocation or tools such as Postman. JWT-based security is essential for maintaining secure session handling without storing sensitive data on the client side.

7.6.2 Admin Role Verification

```python
def admin_required(f):
    """Decorator for admin-only routes"""
    @wraps(f)
    @token_required
    def decorated(current_user, *args, **kwargs):
        if current_user.get('role') != 'admin':
            return jsonify({'message': 'Admin access required'}), 403
        return f(current_user, *args, **kwargs)

    return decorated
```

Figure 7.28: Admin Role Verification

Figure 7.28 shows the role-checking mechanism that restricts advanced operations such as content uploads, deletion, and monitoring activities to administrators only. The admin_required decorator builds on JWT validation by ensuring that the authenticated user possesses an admin role. This prevents standard users or attackers from escalating privileges or manipulating system content.

7.6.3 Secure Password Hashing (bcrypt)

Secure password storage is a crucial component of any authentication system. The AI Image Detection System for Parcel Scams implements strong password protection using the **bcrypt** library, a widely recognized hashing algorithm designed to resist brute-force attacks, rainbow table attacks, and common credential-compromise techniques. Instead of storing plain-text passwords, bcrypt applies a unique salt and performs multiple rounds of encryption to generate a secure hash. This ensures that even if the database were compromised, the original passwords would remain unrecoverable.

Two layers of password hashing are implemented: one for generating the initial administrator account during the system's first-time setup, and another for hashing credentials created through the user registration interface.

```
26          password_hash = bcrypt.hashpw('admin123'.encode('utf-8'), bcrypt.gensalt()).decode('utf-8')
```

Figure 7.29: Admin password hashing

Figure 7.29 shows the secure admin password hashing. This snippet automatically generates a hashed password (admin123) when the system initializes, ensuring no plain-text credentials are ever stored in the database. This hashing process uses bcrypt.gensalt(), which applies a random salt value, making each hash unique even if passwords are identical.

```
92      # Hash password
93      password_hash = bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt()).decode('utf-8')
94
95      # Insert new user
96      user_id = execute_query(
97          "INSERT INTO users (username, password_hash, role) VALUES (%s, %s, %s)",
98          (username, password_hash, final_role)
99      )
```

Figure 7.30: Password hashing process

Figure 7.30 shows the password hashing process for new user registrations. Whenever a new account is created, the password is securely hashed before being inserted into the database. After hashing, the password is stored safely using a parameterized query shown in the Figure 7.30. This prevents SQL injection and ensures that only encrypted password data is saved.

7.6.4 Environment Variable Protection

Environment variable protection is an essential security practice implemented in the AI Image Detection System for Parcel Scams to safeguard sensitive information such as secret keys,

authentication tokens, and database credentials. Instead of hardcoding confidential values directly in the source code where they may be accidentally exposed through version control the system stores them in a protected .env file. The **python-dotenv** library is used to load these variables securely during runtime. This approach ensures that sensitive configurations remain private and can be easily changed without modifying the application code.

Using environment variables enhances security, supports cleaner configuration management, and allows the system to be deployed across different environments (development, testing, production) with minimal effort. The .env file is also excluded from version control using .gitignore, preventing credentials from being uploaded to public repositories.

```
8   from dotenv import load_dotenv
9
10  # Load environment variables
11  load_dotenv()
```

Figure 7.31: Environment variables

Figure 7.31 shows how environment variables are securely loaded in server.py using the python-dotenv library. This ensures that all sensitive values stored in .env are available to the application without ever appearing in the public codebase.

```
25  app.config['SECRET_KEY'] = os.getenv('SECRET_KEY', 'your-secret-key-change-this')
```

Figure 7.32: Secret key from the environment

Figure 7.32 shows how the application reads the secret key from the environment in server.py. The secret key is used by Flask to secure sessions and generate cryptographically signed tokens. Loading it from .env prevents exposure of this critical security component.

```
5   # Secret Keys
6   SECRET_KEY=_ggnmftJZlD7ZbkaxJIwGPNwS_wpgKmEN5z-tag-gjs
7   REG_SECRET=uReIFxc4SMuaT0e_NjAnF0ejhvhtOS-T
8
9   # Database Configuration (MySQL)
10  DB_HOST=localhost
11  DB_USER=root
12  DB_PASSWORD=
13  DB_NAME=image_detection
```

Figure 7.33: Sensitive information credentials are stored securely

Figure 7.x shows part of the .env file where sensitive information such as the SECRET_KEY and database credentials is stored securely. These variables remain stored locally ensuring strong protection against credential leakage.

```python
# Database configuration
DB_CONFIG = {
    'host': os.getenv('DB_HOST', 'localhost'),
    'port': int(os.getenv('DB_PORT', '3306')),
    'user': os.getenv('DB_USER', 'root'),
    'password': os.getenv('DB_PASSWORD', ''),
    'database': os.getenv('DB_NAME', 'image_detection'),
    'raise_on_warnings': True,
    'autocommit': True
}
```

Figure 7.34: Environment variables used in the database configuration

Figure 7.34 shows how environment variables are used in the database configuration inside database.py. This ensures secure, flexible, and centralized configuration management for MySQL connections.

7.6.5 Secure File Upload Validation

Secure file upload handling is crucial to protect the system from potential threats such as malicious scripts, disguised executable files, or oversized uploads that could disrupt server performance. The AI Image Detection System for Parcel Scams applies multiple layers of upload validation to ensure that only safe and legitimate files are accepted into the system. Uploaded images are validated before processing, while admin-uploaded materials (manuals, posters, scam cases) undergo strict filtering to prevent injection of harmful files.

The system uses an allowed file extension list, the secure_filename() function, and explicit upload directories to ensure that files are sanitized and stored safely. This prevents attackers from uploading executable payloads or manipulating file paths.

```python
ALLOWED_EXTENSIONS = {'png', 'jpg', 'jpeg', 'webp', 'gif', 'bmp'}

def allowed_file(filename):
    """Check if file extension is allowed"""
    return '.' in filename and filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS
```

Figure 7.35: File validation

Figure 7.35 shows the allowed file validation used in ai_detection.py, ensuring only safe image formats are accepted. This ensures the detection system only processes image types that are safe and expected by the Sightengine API.

```python
    # Save file
    filename = secure_filename(file.filename)
    timestamp = str(int(time.time() * 1000))
    filename = f"{timestamp}_{filename}"

    manual_dir = os.path.join(current_app.config['UPLOAD_FOLDER'], 'manuals')
    os.makedirs(manual_dir, exist_ok=True)
    upload_path = os.path.join(manual_dir, filename)
    file.save(upload_path)
```

Figure 7.36: Secure file saving

Figure 7.36 shows secure filename sanitization and saving in admin.py, protecting the server from malicious filenames. These combined measures ensure that all uploaded content manuals, scam tip posters, and scam case images is safe, controlled, and properly stored.

7.6.6 Access Control and Role-Based Authorization

The AI Image Detection for Parcel Scams implements a strict access control layer to ensure that only authorized users specifically administrators can perform sensitive actions such as uploading user manuals, scam tips, scam case images, and viewing monitoring dashboards. Instead of using server-side sessions, the system adopts a token-based access control mechanism using JSON Web Tokens (JWT).

When a user logs in, the backend generates a signed JWT token that contains the user's ID, username, and most importantly, role (admin or user). This token must be included in the Authorization header for all protected requests. By embedding the role within the JWT payload, the server can determine the correct permissions without storing any session data.

```
143     # Token expiration: 8 hours from now (using Malaysia local time
144     payload = {
145         'id': user['id'],
146         'username': user['username'],
147         'role': user['role'],
148         'exp': now_malaysia_utc() + timedelta(hours=8)
149     }
150 
151     token = jwt.encode(payload, SECRET_KEY, algorithm='HS256')
```

Figure 7.37: JWT payload

Figure 7.37 shows the JWT payload created in auth.py, where the role is securely embedded inside the token. The token acts as proof of authentication and is required for accessing administrative endpoints.

```
48  def admin_required(f):
49      """Decorator for admin-only routes"""
50      @wraps(f)
51      @token_required
52      def decorated(current_user, *args, **kwargs):
53          if current_user.get('role') != 'admin':
54              return jsonify({'message': 'Admin access required'}), 403
55          return f(current_user, *args, **kwargs)
56 
57      return decorated
```

Figure 7.38: admin_required decorator

Figure 7.38 shows the admin_required decorator in auth.py, which restricts certain routes to admin users only. This decorator ensures the user provides a valid JWT, the token has not expired and the token contains an admin role. If any condition fails, the system returns a 403 Forbidden, preventing unauthorized access.

```
@admin_bp.route('/user-manual', methods=['POST'])
@admin_required
def upload_manual(current_user):
```

Figure 7.39: Admin routes protection

Figure 7.39 shows how admin routes are protected in admin.py. Every administrative upload or monitoring function uses this decorator to enforce role-based access.

By placing the role inside the JWT and validating it for every admin-restricted route, the system achieves a secure, stateless, and scalable authorization mechanism. This protects administrative features from unauthorized users and aligns with modern web security practices.

7.6.7 Protection Against Unauthorized Access to Administrative Endpoints

The AI Image Detection System for Parcel Scams incorporates structural route protection to prevent unauthorized access to sensitive administrative functionalities. By organizing backend functionalities into Flask Blueprints, the system separates public endpoints from admin-restricted routes. All administrative operations such as uploading manuals, scam tips, scam case images, or retrieving detection records are exclusively placed under the /api/admin namespace.

Since these endpoints are protected with the admin_required decorator, users without a valid admin JWT token are automatically denied access. This prevents accidental access, URL manipulation, or automated scanning tools from reaching admin-level functions.

```
# Register blueprints
app.register_blueprint(auth_bp, url_prefix='/api/auth')
app.register_blueprint(content_bp, url_prefix='/api/content')
app.register_blueprint(admin_bp, url_prefix='/api/admin')
app.register_blueprint(ai_detection_bp, url_prefix='/api')
```

Figure 7.40: Secure blueprint registration

Figure 7.40 shows secure blueprint registration in server.py, which isolates administrative routes under a protected namespace. This structural design ensures all administrator functionalities are grouped, controlled, and protected consistently.

Additionally, the backend verifies authentication at each admin endpoint. Any request missing a valid Authorization header or containing a non-admin role result in an immediate rejection.

```
53          if current_user.get('role') != 'admin':
54              return jsonify({'message': 'Admin access required'}), 403
```

Figure 7.41: Admin route protection

Figure 7.41 shows the admin route protection mechanism in auth.py. By combining JWT role verification with strict route isolation, the system ensures robust protection against unauthorized access to privileged functions.

7.6.8 Input Validation and SQL Injection Prevention

To maintain data integrity and defend against SQL injection attacks, the system uses parameterized SQL queries for all interactions with the database. Rather than directly concatenating user input into SQL commands which is a common security vulnerability the system uses placeholder-based queries (%s) and passes all user data as separate parameters. This prevents malicious input from altering SQL intent or executing unauthorized database commands.

```
123     # Get user from database
124     user = execute_query(
125         "SELECT id, username, password_hash, role FROM users WHERE username = %s",
126         (username,),
127         fetch_one=True
128     )
```

Figure 7.42: Login validation

Figure 7.42 shows a secure parameterized SELECT query in auth.py, used for login validation.

```
95      # Insert new user
96      user_id = execute_query(
97          "INSERT INTO users (username, password_hash, role) VALUES (%s, %s, %s)",
98          (username, password_hash, final_role)
99      )
```

Figure 7.43: Creating a new user

Figure 7.43 shows a safe INSERT query in auth.py, used when creating a new user.

```
83      """Delete a detection record"""
84      execute_query("DELETE FROM ai_detections WHERE id = %s", (detection_id,))
```

Figure 7.44: Safe removal of detection records

Figure 7.44 shows a secure DELETE query in admin.py, ensuring safe removal of detection records. Since all variables are passed as query parameters rather than embedded directly into SQL strings, malicious payloads cannot be executed, even if attackers attempt to inject database commands through user inputs.

By enforcing strict parameterized queries, input validation, and sanitized user inputs, the system effectively prevents SQL injection attacks one of the most common vulnerabilities found in web applications.

7.7 Conclusion

This chapter, at its end, described the implementation of the AI Image Detection System for Parcel Scams by covering all the areas involved in developing this system: the platform for execution, the tools used for development, and the hardware specifications. Each interface, either for users or administrators, has been designed with a focus on usability, functionality, and ease of interaction. In addition, image uploading, fraud awareness content, and detection results management demonstrate an integration of many back-end features into one platform. This chapter reviewed the secure coding aspect, especially the implementation of JWT authentication, bcrypt password hashing, protection of environment variables, file validation, and parameterised database queries. These are some of the measures that have been put into place to protect the system's data and administrative functions against common cyber threats. In general, this implementation phase successfully translated the system design into a functional and secure application, fulfilling the project objectives and laying a solid ground for testing and further development.

8 TESTING

8.3 Introduction

Testing is a critical phase that verifies the accuracy, security, and dependability of the developed AI Image Detection System for Parcel Scams. This chapter provides a summary of the primary testing strategies implemented throughout the project to verify system functionality, assess performance, and confirm compliance with the specified requirements. These encompass unit testing of individual modules, integration testing for the interaction between components, system testing to assess overall end-to-end functionality, and acceptance testing, wherein actual users participate in the evaluation process to determine usability and overall quality. In addition, both functional and non-functional aspects are taken into account to ensure the system operates seamlessly, manages invalid input effectively, and adheres to best security practices. The outcome of each testing phase will ensure that the system is prepared for deployment and capable of providing a reliable, user-centric experience.

8.4 Unit Testing

Module / Function	Test Objective	Test Steps	Expected Result	Actual Result	Status
Password Hashing	Ensure bcrypt hashes passwords correctly	Register user with plain password	Password stored as bcrypt hash	Successfully hashed	Pass
JWT Generation	Validate JWT payload and expiration	Login with valid credentials	JWT token generated with correct fields	Token generated	Pass
Token Validation	Verify protected routes require JWT	Access /api/admin/dashboard without token	“Token is missing” error	Error returned	Pass
Database Connection	Ensure MySQL connection is successful	Start server	“Connected to MySQL Server” message	Connection established	Pass

Allowed File Checker	Validate file type filtering	Upload .exe file	File rejected	Rejected as expected	Pass
Insert Record	Validate DB insertion	Insert new scam tip	Data saved in scam_tips table	Record created	Pass
AI Detection Route	Validate request input	Upload invalid/non-image file	Return error and stop process	Error correctly shown	Pass
Admin Stats Query	Test COUNT queries	Access /stats	Correct numbers returned	Matches table records	Pass

Table 8.1: Unit Testing

Table 8.1 shows the Unit Testing Table. Unit testing validated individual components of the system to ensure that each function behaved correctly in isolation. Critical backend features tested include password hashing, JWT creation, token validation, and various database operations. Tests for authentication confirmed that bcrypt successfully translated plaintext passwords into secure hashed values and that JWT tokens contained the proper user ID, role, and expiration timestamp. Database utilities were tested to confirm that the system connected to MySQL properly and returned results using parameterized queries. It also tested some file validation functions to ensure only approved file types- PDF for manuals and images for tips/cases-were uploaded. The AI detection API route was tested to make sure that uploads with invalid files returned error responses rather than allowing processes to continue. These tests helped identify early logic errors and ensured that individual components were stable before being integrated into larger workflow sequences.

8.5 Integration Testing

Integrated Modules	Test Objective	Test Steps	Expected Output	Actual Output	Status

Login + JWT + Admin Routes	Ensure authenticated admin can access dashboard	Login → use token → access /api/admin/uploads	Access granted	Successful	Pass
Upload + File Save + DB Insert	Validate poster upload process	Upload scam tip poster	Poster saved + DB record inserted	Works as expected	Pass
Manual Upload + Secure Filename	Validate secure filenames	Upload manual with long filename	Renamed with timestamp	Correct	Pass
AI Detection + DB Save	Ensure detection output saved	Upload image → generate result	Entry created in ai_detections	Created	Pass
Content Update + DB Update	Validate update content route	Edit scam tip	Updated in DB	Successful	Pass
Role-Based Access + Token Validation	Ensure user cannot access admin routes	Login as normal user	Request rejected	“Admin access required”	Pass
Frontend Page + Backend API	Display scam tips on user page	Open Scam Tips page	Successfully fetched via API	Renders properly	Pass

Table 8.2: Integration Testing

Table 8.2 shows the Integration Testing Table. Integration testing covered the interaction of several backend modules that need to work together to support complete workflows. This involved connecting the login system with JWT authentication and verification that protected admin functionalities were available only to authorized users. File upload modules were integrated with secure filename processing and database storage operations to ensure that uploaded files-manuals, scam tips, scam cases-were stored correctly and their records updated consistently. The AI detection module was tested with upload handling and logging in the database, with all detection results appearing on the admin interface. Tests of the integration also verified that the user interface can successfully fetch content from the backend

routes and display it without error. These integration tests uncovered problems that could not be seen when units were individually tested, guaranteeing coordinated and reliable system behaviour.

8.6 System Testing

8.6.1 Functional Testing

Function	Test Scenario	Expected Result	Actual Result	Status
User Login	Login with valid credentials	Login successful, token returned	Passed	Pass
Image Detection	Upload image for analysis	System returns confidence %, probability	Passed	Pass
View Scam Tips	User opens scam tips	Tips displayed correctly	Passed	Pass
Upload Scam Case (Admin)	Admin uploads case image	Saved + visible in list	Passed	Pass
Edit Manual	Admin edits manual entry	Changes saved	Passed	Pass
Delete Record	Admin deletes scam tip	Record removed from DB	Passed	Pass

Table 8.3: Functional Testing

8.6.2 Non-Functional Testing

Test Area	Test Description	Expected Result	Actual Result	Status
Performance	Measure response time of detection page	< 3 seconds load time	2.1 seconds	Pass
Usability	Check clarity of navigation	Easy-to-navigate	Users reported good usability	Pass

Reliability	Stress test with repeated requests	System stays stable	No crashes	Pass
Security	Unauthorized access attempt	Access denied	Denied correctly	Pass
Compatibility	Test across Chrome & Edge	No UI issues	Works correctly	Pass

Table 8.4: Non-Functional Testing Table

Table 8.3 and Table 8.4 shows Functional Testing and Non-Functional Testing Table. System testing was aimed at the complete application as a whole to ensure that all functionalities acted within the specified system requirements. Functional testing focused on the core operations of user authentication, AI image detection, content display, and admin-related CRUD operations. Every feature has been tested in real usage scenarios to make sure it behaves correctly in normal and erroneous situations. For instance, users trying to log in with wrong credentials were given proper error messages, while admins trying to upload file types other than the specified ones were prevented from doing so.

Non-functional testing was performed for performance, usability, reliability, and security. Responsiveness was determined by measuring page loading times; for system stability, API calls were made in quick repetition. Usability was ensured to have an intuitive interface for the end user as well as the admins. Unauthorized access attempts assured the team that security was up and working. This software testing guarantees not only that the system is functional but also that it is reliable and user-friendly.

8.7 Acceptance Testing

8.7.1 Alpha Testing

Tester	Area Tested	Findings	Fixes Applied	Status
Developer	Login, detection, upload	Minor UI misalignment	Adjusted CSS	Pass
Supervisor	Admin CRUD, navigation	Confusing button label	Renamed buttons	Pass
Developer	Error handling	Missing validation message	Added custom messages	Pass

Table 8.5: Alpha Testing

Table 8.5 shows the Alpha Testing Table. Alpha testing was performed as the initial assessment phase to verify that the system's fundamental functionalities were stable prior to deployment to external users. This phase was conducted internally by the developer and the project supervisor. The primary objective of Alpha testing was to identify early defects, inconsistencies, and usability problems that may not be readily apparent during the development process.

During this phase, essential components including the logon and registration processes, administrative authorization, AI image detection procedures, and content upload modules were comprehensively tested. The supervisor also examined the administrative functions, encompassing CRUD operations for scam recommendations, scam cases, and user manuals. Several minor issues were detected, including user interface misalignment, ambiguous button labels, and absent validation messages. These concerns were promptly resolved prior to progressing to the subsequent testing phase.

8.7.2 Beta Testing

Tester Group	Scenario	Feedback	Improvement Applied	Status
Students	Try full system	Needed clearer instructions	Added tooltips	Pass
Non-technical user	Upload detection	Difficulty noticing upload box	Increased highlight	Pass

General users	Browse scam tips	Request for bigger images	Enhanced poster preview	Pass

Table 8.6: Beta Testing

Table 8.6 shows the Beta Testing Table. Beta testing happened after Alpha testing was complete and the system was functionally stable. It involved the participation of real end-users, class colleagues, friends, and non-technical contacts, and was dedicated to simulating real-world conditions. Beta users used the system independently, having no guidance whatsoever, to allow the developer to benefit from feedback regarding usability, easiness of navigation, user experience, and clarity of the interface.

Testers focused on functions such as image detection, browsing scam awareness content, viewing Malaysia scam cases, and evaluating uploaded materials provided by the administrator. Feedback received during testing revealed a need for more direct instructions, along with other facets of clarity of certain interface elements and better visual presentation of uploaded images and their detection results. These were implemented for better performance in improving the user's experience. Beta testing proved that the system performed as users would wish for ease of use, efficiency in response, and readiness for deployment in practical environments.

8.7.3 Questionnaire Analysis (Post-Development)

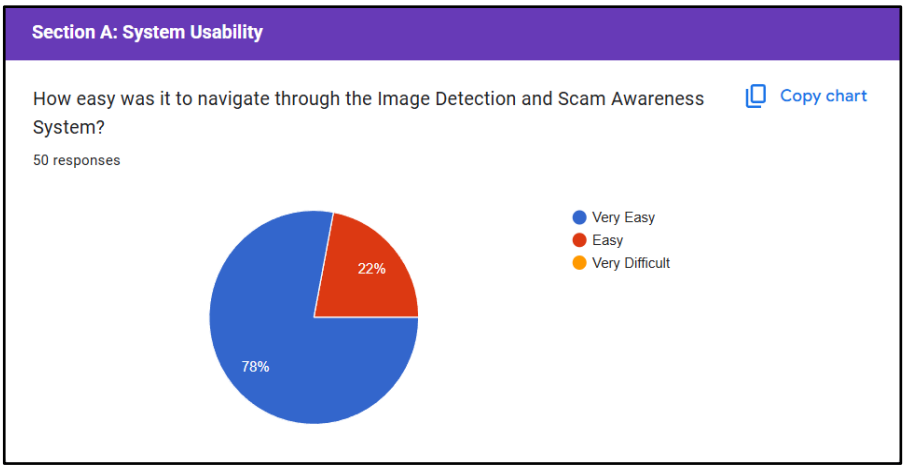


Figure 8.1: Result of Demographic Question 1 (Post-Development)

Figure 8.1 shows the respondents' feedback on how easy it was to navigate through the Image Detection System. The chart reveals that 78% of users rated the navigation experience as

Very Easy, while 22% indicated it was Easy. Importantly, no respondents reported difficulty using the system, demonstrating that the interface is intuitive and user-friendly. These results indicate that the system's layout, menu placement, and page structure are clear and easily understood by users, contributing to a smooth overall experience.

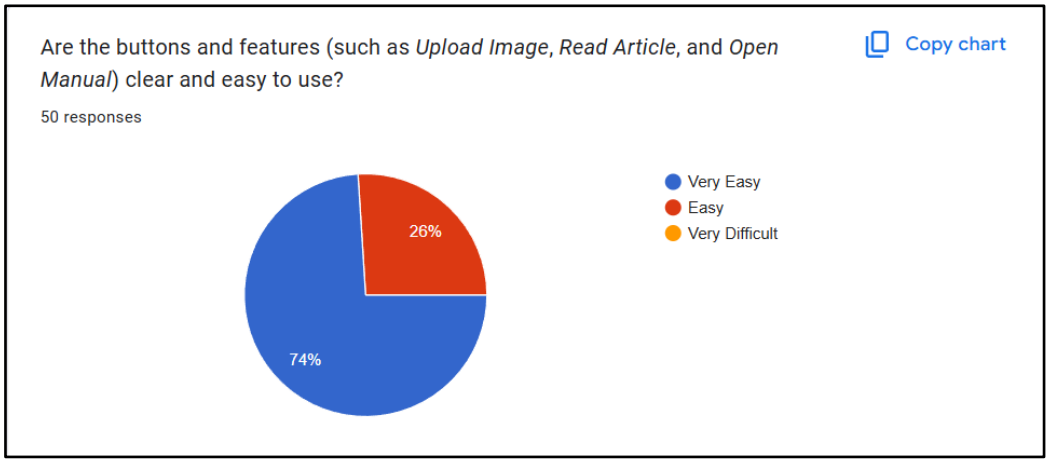


Figure 8.2: Result of Demographic Question 2 (Post-Development)

Figure 8.2 illustrates user responses regarding the clarity and usability of essential system features such as Upload Image, Read Article, and Open Manual. A total of 74% of respondents found these features Very Easy to use, while 26% rated them as Easy. The absence of negative responses suggests that button labels, icons, and interactive components are well-designed, easily visible, and require minimal learning effort. This supports the system's goal of being accessible even to non-technical users.

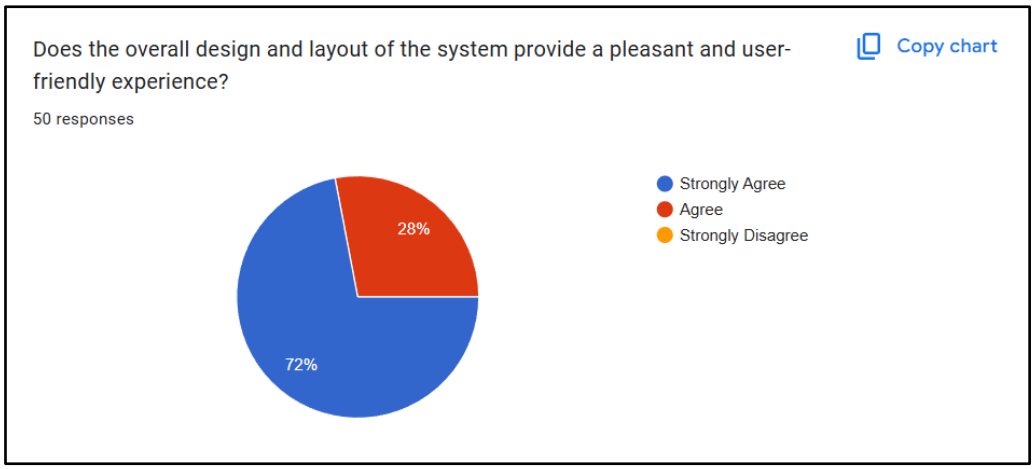


Figure 8.3: Result of Demographic Question 3 (Post-Development)

Figure 8.3 displays user satisfaction with the system's overall design and interface layout. The results show that 72% of respondents Strongly Agree that the system provides a pleasant and user-friendly experience, while 28% simply Agree. The consistent positive responses emphasize that the system's colour scheme, typography, spacing, and visual hierarchy contribute to an aesthetically pleasing and comfortable browsing experience.

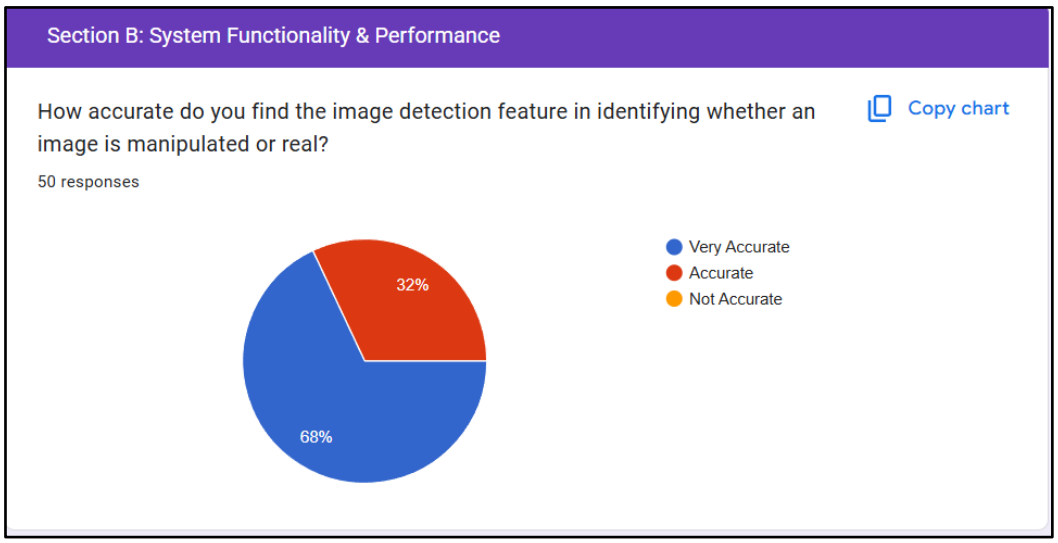


Figure 8.4: Result of Demographic Question 4 (Post-Development)

Figure 8.4 highlights user perceptions of the accuracy of the AI-based image detection component. A significant 68% rated it as Very Accurate, while 32% rated it as Accurate. No respondents selected Not Accurate, indicating high user confidence in the system's detection results. This strong outcome demonstrates the model's reliability in identifying manipulated or genuine images, fulfilling its core purpose.

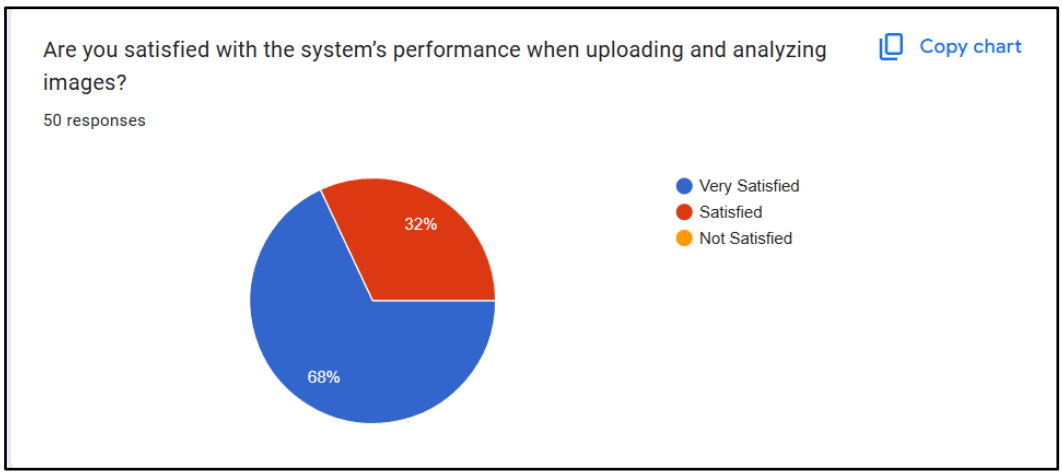


Figure 8.5: Result of Demographic Question 5 (Post-Development)

Figure 8.5 represents user satisfaction with the system's performance when uploading and analysing images. According to the results, 68% were Very Satisfied and 32% were Satisfied. No respondents expressed dissatisfaction. These findings confirm that the backend processing, API handling, and response times are efficient and optimized, providing a seamless experience when analysing images.

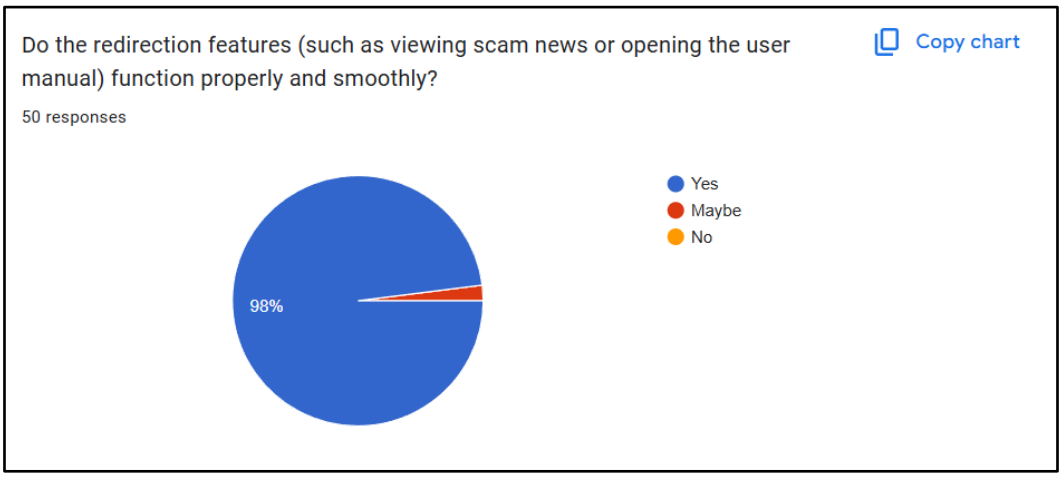


Figure 8.6: Result of Demographic Question 6 (Post-Development)

Figure 8.6 shows how users rated the performance of redirection functions such as viewing scam articles or opening user manuals. An overwhelming 98% of users answered Yes, indicating smooth and correct functioning, while 2% answered Maybe. The results validate that hyperlinks, routing, and connected content load consistently without errors, contributing to smooth navigation.

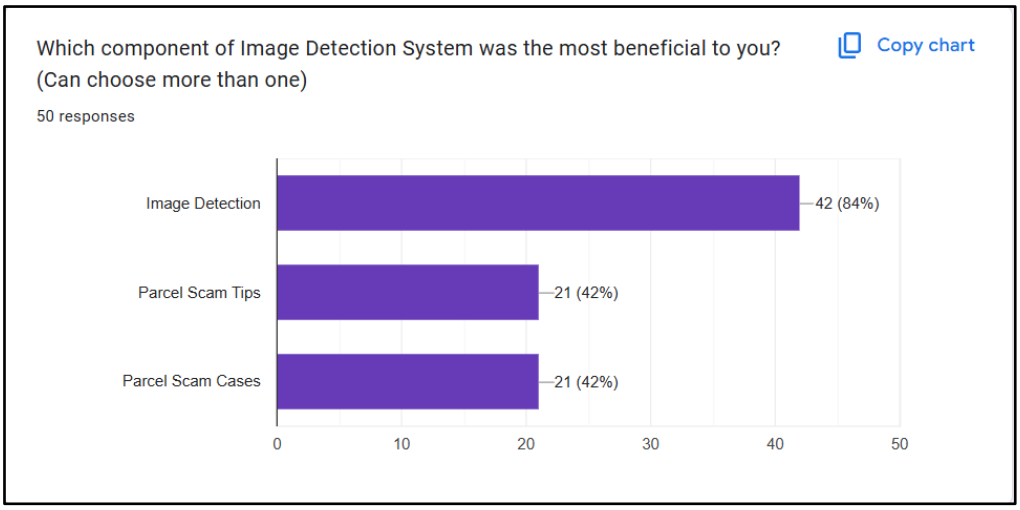


Figure 8.7: Result of Demographic Question 7 (Post-Development)

Figure 8.7 summarizes which system components users found most beneficial. The majority, 84%, selected the Image Detection feature, demonstrating its importance and usefulness.

Additionally, 42% of respondents found value in the Parcel Scam Tips section, and another 42% benefitted from the Malaysia Scam Cases section. These results show that users appreciated both the detection capabilities and the educational content, reinforcing the system's dual functionality as a detection and awareness tool.

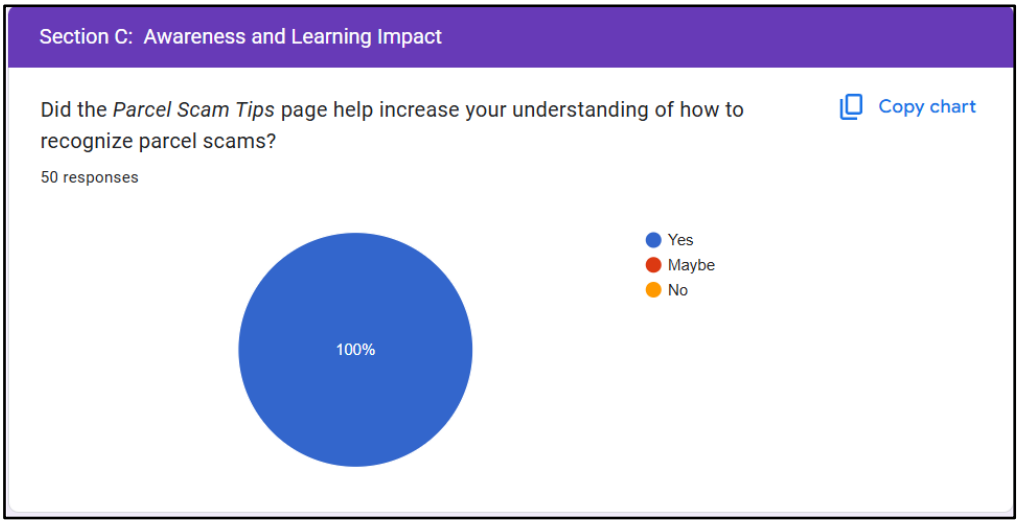


Figure 8.8: Result of Demographic Question 8 (Post-Development)

Figure 8.8 presents user feedback on whether the Parcel Scam Tips page increased their understanding of scam recognition. All 100% of respondents answered Yes, demonstrating the page's major contribution in educating users about identifying suspicious or fraudulent parcel-related activities. This confirms that the awareness materials provided are effective and engaging.

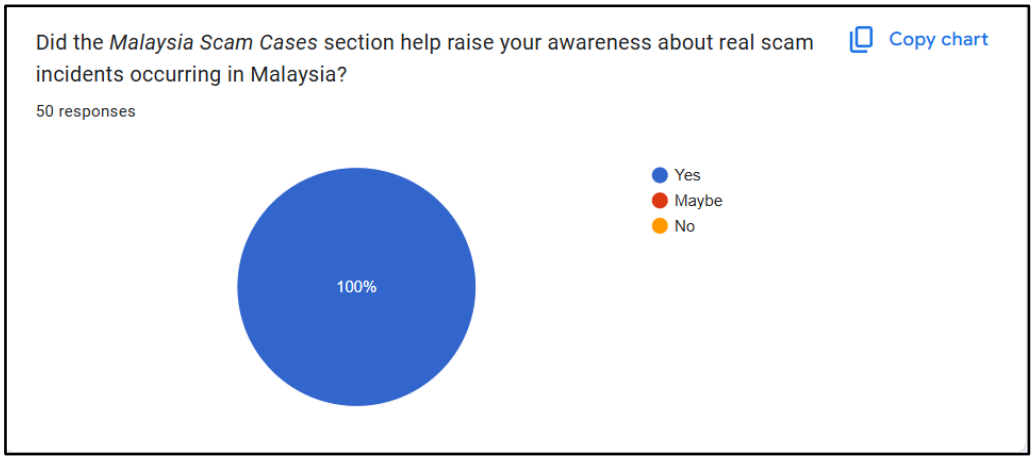


Figure 8.9: Result of Demographic Question 9 (Post-Development)

Figure 8.9 reflects whether the Malaysia Scam Cases section improved user awareness regarding real scam incidents. All 100% of respondents selected Yes, indicating strong

educational value. This shows that real-world examples are highly impactful for users and help reinforce scam prevention knowledge.

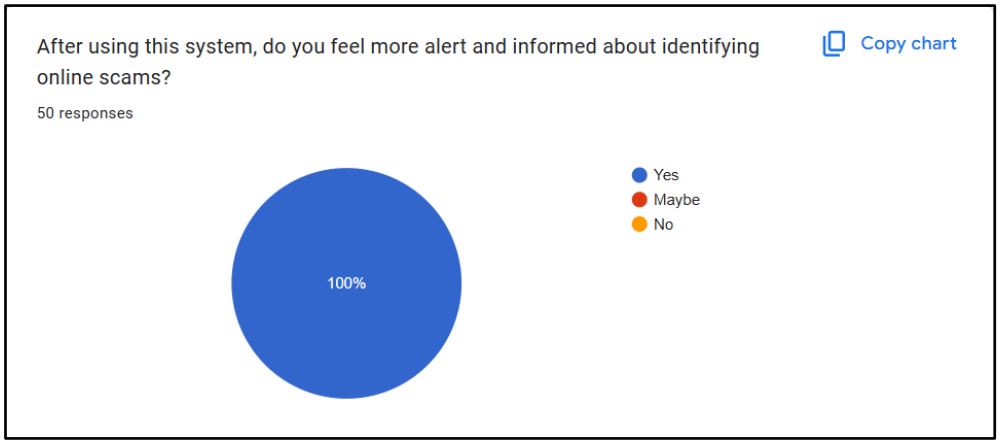


Figure 8.10: Result of Demographic Question 10 (Post-Development)

Figure 8.10 reveals whether users felt more alert and informed about identifying online scams after using the system. Once again, 100% of respondents chose Yes, showing that the system successfully increases cybersecurity awareness and contributes positively to users' understanding of scam behaviour.

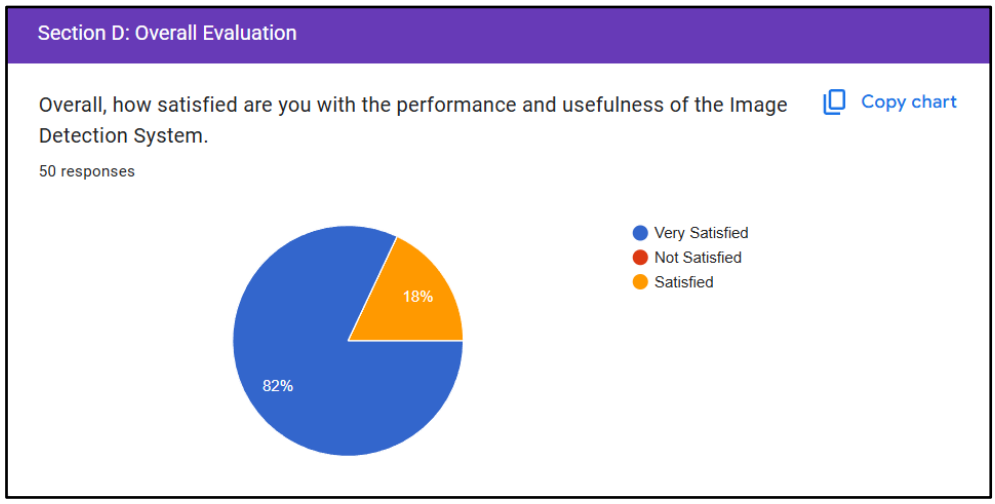


Figure 8.11: Result of Demographic Question 1 (Post-Development)

Figure 8.11 shows user's overall satisfaction with the performance and usefulness of the AI Image Detection System for Parcel Scams. A strong 82% reported being Very Satisfied, while 18% were Satisfied. No respondents expressed dissatisfaction, confirming that the system effectively meets user expectations in terms of functionality, accuracy, and experience. The overwhelmingly positive response highlights successful system implementation and user acceptance.

8.8 Conclusion

In summary, the testing done throughout this chapter has shown that the AI Image Detection for Parcel Scams is functionally stable, usable, and reliable. Core backend components like authentication, database operations, file uploads, and AI detection were verified through unit and integration testing to work seamlessly with each other. Further system testing confirmed that all its features work according to the requirement, while non-functional testing confirmed the responsiveness, security, and user experience of the system. Acceptance testing by alpha and beta users provided useful insights for final refinements and interface enhancements. Moreover, a questionnaire analysis for 50 respondents also established overwhelmingly positive feedback regarding navigation ease, detection accuracy, awareness impact, and overall satisfaction. The combined results confirm that the system has successfully met its objectives of detecting manipulated images and improving scam awareness, making it ready for real-world application and enhancements in future work.

9 PROJECT MANAGEMENT

9.3 Introduction

This chapter highlights the project management aspects applied throughout the development of the AI Image Detection System for Parcel Scams. Effective project management ensures the progress of a project in accordance with a planned timeline, meeting the required specifications, and within the allocated resources. The project schedule, WBS, Gantt Chart, and some considerations on risk management that have been guiding the execution of the FYP2 are outlined in this chapter. By decomposing the project into small, manageable tasks, the development became more organized, allowing consistent monitoring and timely completion of milestones. The strategies and tools described in this chapter ensured that the project was accomplished in a systematic way, efficiently, and within the expected academic timeframe.

9.4 Project Schedule

A project schedule is the systematic planning of all activities that must be completed within a defined timeframe to ensure the project advances in the proper sequence and is completed punctually. It delineates the specific tasks to be accomplished, the necessary resources, and the designated timelines for completion (Gurnov, 2024). Similarly, in the development of the AI Image Detection System for Parcel Scams, a structured timetable was crucial to guarantee that each phase from backend configuration to system testing was executed effectively and within the allotted semester timeframe.

Two primary instruments were employed to facilitate the planning process: the Work Breakdown Structure (WBS) and the Gantt Chart. The Work Breakdown Structure is an essential project management technique that subdivides a large, complex project into smaller, more manageable segments. By segmenting the entire system development into well-defined tasks, responsibilities can be allocated more efficiently, and progress can be systematically tracked throughout the project lifecycle (Schwartz, 2025).

Meanwhile, the Gantt Chart functions as a visual timeline that displays all project tasks, their respective durations, and their interrelationships. It enables project members to efficiently monitor deadlines, recognize overlapping tasks, and identify potential delays that could impact overall progress. This visual depiction offers a comprehensive overview of the entire project timeline and guarantees that the development stays aligned with the established milestones (Malsam, 2025).

9.4.1 Work Breakdown Structure

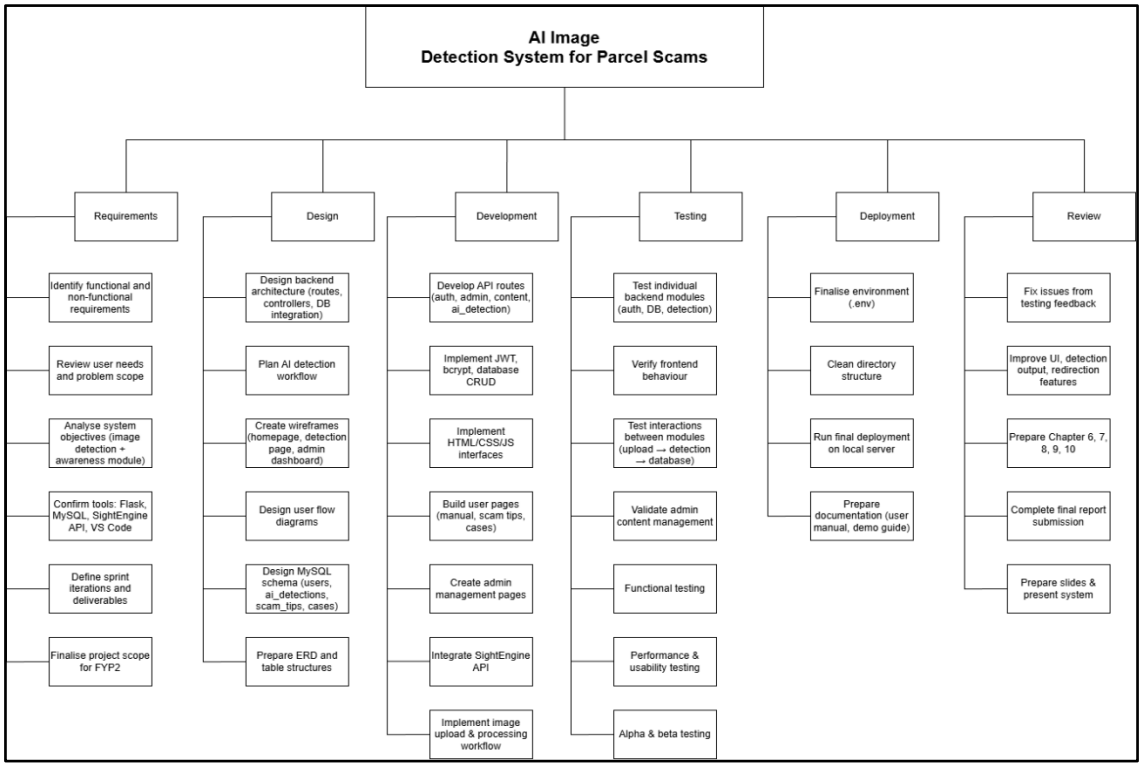


Figure 9.1: Work Breakdown Structure (WBS) of AI Image Detection System

Figure 9.1 shows the Work Breakdown Structure (WBS) of AI Image Detection System. The Work Breakdown Structure for Final Year Project 2 represents the hierarchical order of major phases necessary for the development of an AI Image Detection System for Parcel Scams. The WBS is organized into six Agile-based phases: Requirements, Design, Development, Testing, Deployment, and Review, which ensures a structured yet iterative workflow.

The Requirements phase identifies the needs of the system, defines sprint deliverables, and finalizes the project scope. This helps to ensure a clear functional understanding upfront. The Design phase now takes requirements and converts them into technical plans by creating backend architecture, user flows, wireframes, MySQL schema, and ERD structure. In the Development phase, all system parts will be constructed: API routes, authentication modules, CRUD operations, users' and admin interfaces, and integration of the SightEngine AI API.

The Testing phase shall validate each implemented feature through unit testing, integration testing, system validation, functional testing, and performance/usability testing. Alpha and beta testing ensure user acceptance and reliability. The Deployment phase prepares the operating environment, organizes directories, performs the final server deployment, and produces system documentation. Finally, the Review phase shall focus on the resolution of

feedback, UI enhancement, enhancing the system behaviour, completion of chapters 6–10, and preparation of the final presentation.

The WBS details a comprehensive roadmap that systemizes the division of work to be performed into tasks that are more manageable, hence laying down clear paths for organized progress with efficient monitoring across all phases.

9.4.2 Gantt Chart

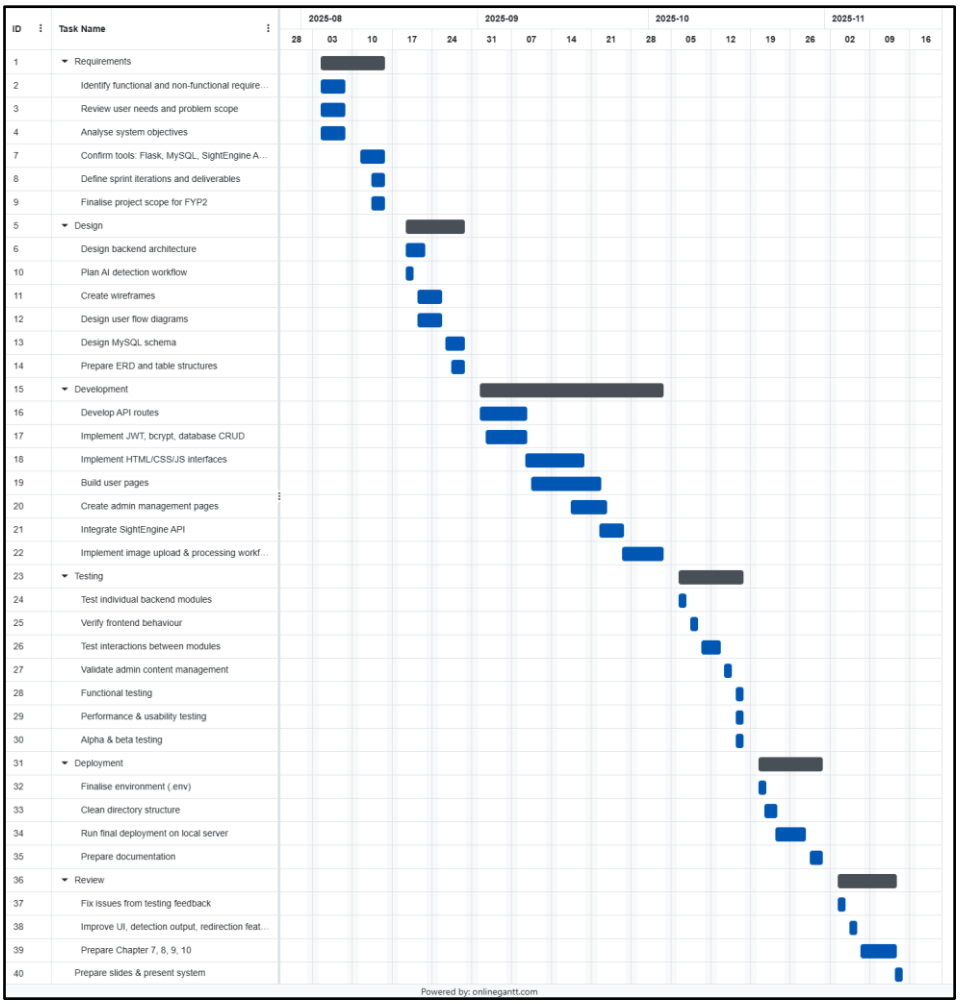


Figure 9.2: Gantt Chart of AI Image Detection System

The Gantt chart represents the overall project timeline, ranging from 4 August 2025 to 14 November 2025, and aligns each task from the WBS within a sequential schedule. This graphical timeline puts much emphasis on the duration of tasks, their sequencing, overlaps, and interdependencies among phases. It starts with the Requirements Phase, between August 4th and 15th, in which the identification of the requirements, user analysis, clarification of objectives, and scoping of the project were done within ten days.

The Design Phase covers nine days, from August 19 to 29, for architecture planning, designing the detection workflow, developing the wireframe, mapping the user flow, writing MySQL schema, and preparing the ERD. This will usher in the Development Phase, running for 25 days, from September 1 to October 3, the longest phase, which will involve API development, authentication, CRUD modules, user interface design, administrative pages, and AI detection integration.

The Testing Phase, from 6–17 October, is a 10-day period that includes all types of unit testing, frontend validation, module interaction assessment, functional testing, and performance and usability testing. Alpha and beta testing have also been finalized at this stage. Deployment, from October 20 to October 31, includes the configuration of the environment, cleaning up the directories, deployment of the system, and preparing the documentation.

The project culminates in the Review Phase, which will run from 3–14 November and focuses on improvements, chapter creation, and the final presentation. On the whole, the Gantt chart makes sure that everything moves forward in an orderly, timely, and well-planned manner.

Phase	Summary of Activities	Week(s)
Requirements	Identify user needs, define system scope, analyze objectives, confirm tools, and finalize the complete project plan.	Week 1 – Week 2
Design	Create backend architecture, plan AI detection workflow, design wireframes, develop UI flow diagrams, and prepare MySQL database structures.	Week 3 – Week 4
Development	Develop API routes, implement authentication and CRUD modules, build interfaces, integrate SightEngine API, and complete system functionalities.	Week 5 – Week 9
Testing	Conduct unit, integration, functional, and non-functional testing, verify all module interactions, and perform Alpha & Beta testing.	Week 10 – Week 11
Deployment	Finalize environment setup, clean project structure, deploy system on local server, and prepare documentation.	Week 12 – Week 13
Review	Fix issues from testing, refine system performance, analyze user feedback, complete report writing, and prepare presentation materials.	Week 13 – Week 14

Table 9.1: Project Schedule Timetable

9.5 Risk Management

Risk	Analysis	Mitigation
API Failure (SightEngine service unavailable)	The image analysis feature depends on an external API. If the service is down or rate-limited, the detection module becomes unusable.	Implement API error-handling, fallback messages, and retry logic. Log failures and allow users to re-upload later.
Incorrect AI Detection Output	ML-based analysis may sometimes misclassify images, affecting system reliability.	Provide confidence scores, disclaimers, and continuous tuning of thresholds. Allow users to reanalyse with different images.
Database Connection Failure	MySQL server downtime or incorrect DB configuration may disrupt all CRUD operations and login functions.	Use environment variables, ensure MySQL services are running, implement connection retries, and enable logging for database errors.
Data Loss During Uploads	Corrupted files or interrupted uploads may prevent scam tips/cases/manuals from being saved.	Validate file types, enforce size limits, and store backups in `/uploads` with timestamped filenames.
Unauthorized Access to Admin Panel	Weak passwords or predictable admin secret may allow attackers to manipulate content.	Use bcrypt hashing, JWT-based admin-only routes, a strong REG_SECRET, and enforce password complexity.
Sensitive Data Exposure in .env File	Exposing SECRET_KEY, DB credentials, or API secrets may compromise the entire system.	Keep `.env` outside public repo, use `.gitignore`, and restrict access to environment variables.
Cross-Site Scripting (XSS) via text fields	Admin enters titles or content that may contain malicious scripts.	Sanitize all user inputs before saving, escape

		HTML when rendering, and validate form fields.
File Upload Vulnerabilities	Uploading malicious files disguised as images or PDFs could expose the server.	Validate extensions using `allowed_file()`, use `secure_filename()`, scan file metadata, and limit upload size.
Performance Lag During Large Images	Analysing big image files may slow down the server or exceed API limits.	Limit file size to 16MB, optimize preprocessing, and notify users to upload reasonable image types.

Table 9.2: AI Image Detection System's Risk Management

Risk management for this system focuses on identifying project-specific threats related to technical performance, system integration, and timeline delivery. One major risk is AI API integration failure, which may disrupt detection accuracy or prevent results from being generated. This is mitigated by early API testing and preparing fallback error-handling methods. Another risk is database inconsistencies, which may affect CRUD operations or result logging. To mitigate this, structured SQL schema planning, backups, and validation scripts are implemented.

A frequent risk in software projects is delayed development due to complex backend logic. This is reduced through sprint-based planning and early prototyping of core modules such as authentication and image processing. User-related risks such as unclear UI behaviour or poor usability are addressed through multiple testing cycles, including performance and usability testing.

Finally, deployment risks such as environment misconfiguration are mitigated by maintaining clear .env management procedures, directory structuring, and performing deployment rehearsals.

9.6 Conclusion

Chapter 9 has demonstrated how effective project management played a crucial role in ensuring the successful development of the AI Image Detection System for Parcel Scams. Through the structured use of Agile methodology, the project was systematically divided into manageable phases, allowing continuous refinement, flexibility, and iterative improvements.

The Work Breakdown Structure (WBS) provided a clear hierarchical overview of all required tasks, ensuring that each activity ranging from requirements analysis to final review was well-organised and aligned with the project objectives.

The Gantt chart translated this structure into a comprehensive and realistic timeline, visually outlining the sequencing, duration, and dependencies of each task throughout the 14-week development period. This enabled effective time management and ensured that the project stayed on track. Additionally, the risk management strategies identified potential challenges early and provided practical mitigation measures to reduce impact, ensuring smooth progression across all phases.

Overall, the project management approach applied in this chapter strengthened planning, organisation, and execution, supporting the delivery of a functional and user-oriented system. The combination of structured scheduling, clear task breakdown, and proactive risk handling ensured the project remained controlled, efficient, and aligned with its intended outcomes.

10 Conclusion

10.1 Introduction

This chapter concludes the development of the AI Image Detection System for Parcel Scams, summarizing its overall achievements, constraints, and recommended future enhancements. The purpose of this project was to address the growing issue of parcel scams in Malaysia by developing an accessible platform capable of verifying suspicious parcel-related images. The system integrates AI-based detection, image comparison, and educational content to help users recognise manipulated images commonly used in scam attempts. The following sections highlight how well the project met its objectives and propose future improvements that can strengthen its effectiveness and sustainability.

10.2 Achievements

10.2.1 To Raise Awareness and Educate Users on Parcel Scam Images in Malaysia

This objective was successfully achieved through the creation of an educational module containing parcel scam tips and real cases from Malaysia. The platform provides structured awareness materials, including examples of manipulated images and common scam tactics. These resources allow users particularly students and online shoppers to gain a better understanding of how image-based scams operate, improving their ability to recognise suspicious visuals before becoming victims.

10.2.2 To Develop a System that Allows Users to Upload and Verify Images Suspected to Be Used in Parcel Scams

The system successfully enables users to upload parcel-related images such as tracking screenshots, payment receipts, or chat screenshots. Once uploaded, the system analyzes the image using AI-based detection models and presents results in a clear visual format. This supports users in validating image authenticity and helps them identify potential fraud indicators quickly and conveniently.

10.2.3 To Identify and Analyze Reused or Manipulated Images Commonly Associated with Parcel Scams

Through integrated AI analysis, the system can detect manipulation traces, reused patterns, and artificial generation characteristics. The system identifies key red flags such as abnormal textures, AI-generated patterns, or inconsistencies commonly found in scam-related images. This achievement supports users with accurate information to determine whether an image has been altered or reused by scammers.

10.3 Constraint and Limitation

Although the system fulfills its primary functions, several limitations still exist.

1. Model Accuracy Dependency – The AI models rely on existing datasets and may not detect newly emerging manipulation styles or sophisticated image alterations.

2. No Dedicated Image History Database – The system currently does not store a long-term log of analysed images, limiting the ability to identify recurring scam patterns.

3. Limited Real-time Capabilities – The system does not yet provide instant alerts or notifications for newly detected scam patterns.

4. Internet Dependency – Users require a stable internet connection to upload images and receive results, which may affect accessibility in certain areas.

10.4 Future Work and Recommendation

10.4.1 Implement a Database for Image Detection History

A dedicated database can be added to store previously uploaded images, analysis results, timestamps, and user interactions. This enhancement allows long-term pattern recognition, supports law enforcement collaboration, and helps identify frequently reused scam images across different cases.

10.4.2 Add a Help Chatbot or Virtual Assistant for User Support

Introducing an AI-powered chatbot can guide users through system features, answer common questions, and provide instant assistance during the detection process. This improves accessibility, especially for non-technical users who may be unfamiliar with AI tools.

10.4.3 Enhance Security With Image Encryption and Data Protection

Since users upload sensitive images such as receipts or tracking documents, implementing encryption for uploaded files will enhance privacy and security. This ensures confidentiality, meets data protection standards, and increases user trust in the system.

10.5 Conclusion

In conclusion, the AI Image Detection System for Parcel Scams successfully fulfills its main objectives by educating the public, providing an accessible verification platform, and assisting users in identifying manipulated parcel-related images. The system helps reduce the risk of falling victim to parcel scams, especially among Malaysian online shoppers and general internet users. While the system performs effectively within its current scope, future enhancements such as a comprehensive image database, intelligent chatbot support, and stronger security features can significantly improve its scalability and real-world applicability. Overall, this project contributes meaningfully to combating digital fraud and supports Malaysia’s broader effort to promote safer online interactions.

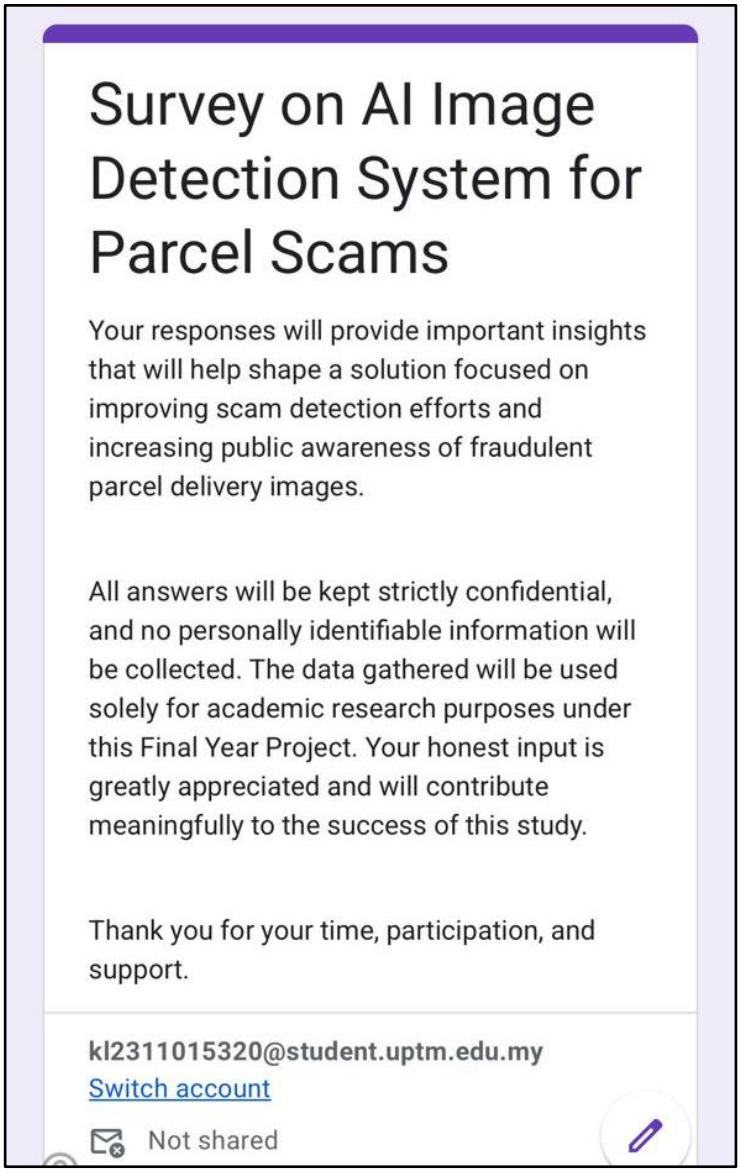
APPENDIX A - Requirements Specification Document

I. Project Source Code (GitHub Link) PhiSAD
https://github.com/Nurmiza2105/AI-IMAGE-DETECTION-FOR-PARCEL-SCAM

II. Demonstration Video (YouTube Link)
AI Image Detection System for Parcel Scams Demonstration Video
https://youtu.be/UDiqVFjmsIw?feature=shared

APPENDIX B – QUESTIONNAIRE

I. Questionnaire Pre-Development



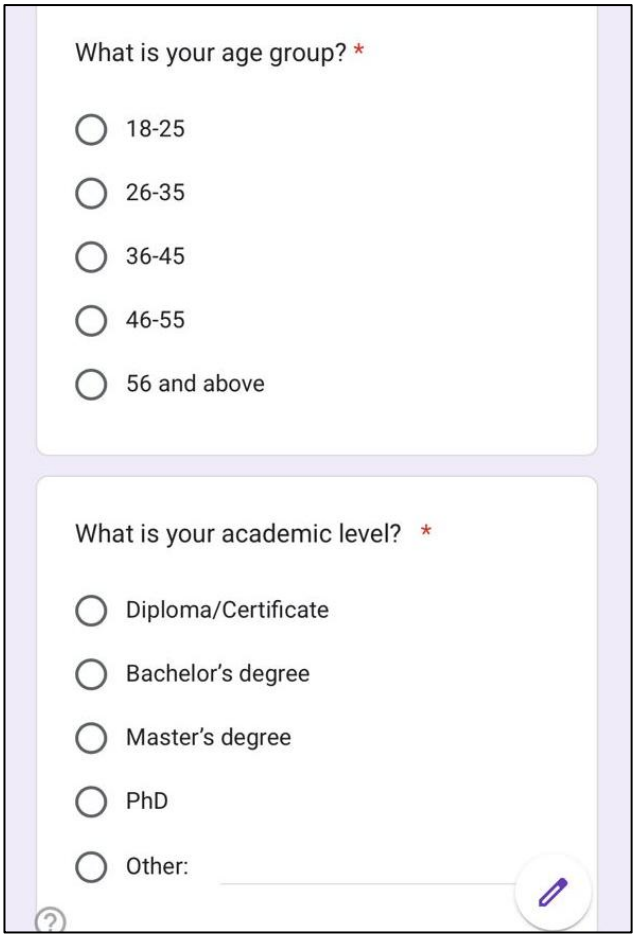
Survey on AI Image Detection System for Parcel Scams

Your responses will provide important insights that will help shape a solution focused on improving scam detection efforts and increasing public awareness of fraudulent parcel delivery images.

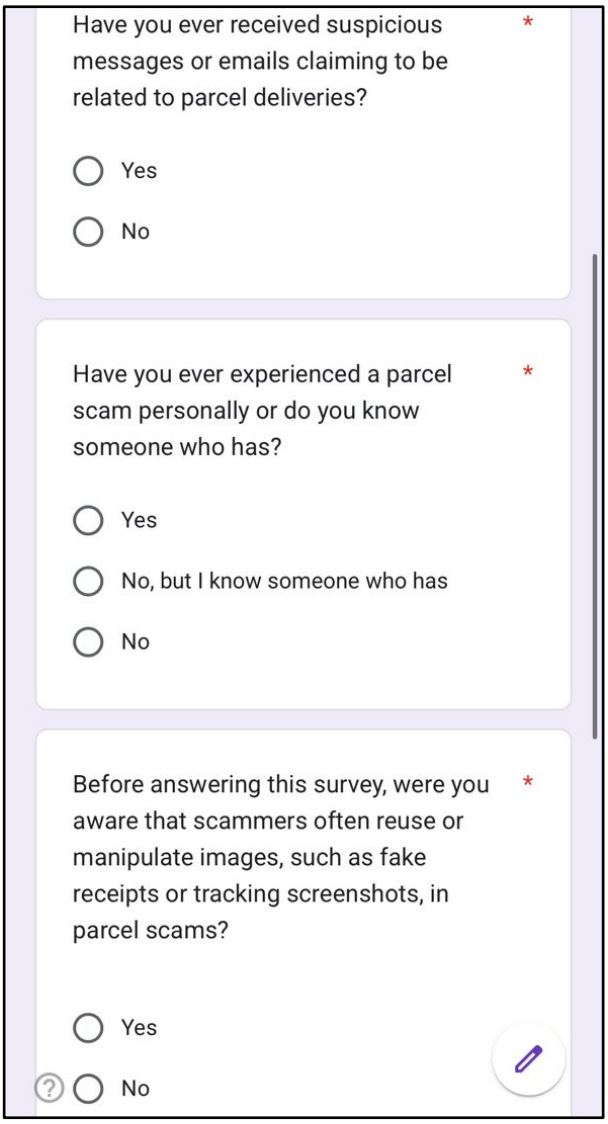
All answers will be kept strictly confidential, and no personally identifiable information will be collected. The data gathered will be used solely for academic research purposes under this Final Year Project. Your honest input is greatly appreciated and will contribute meaningfully to the success of this study.

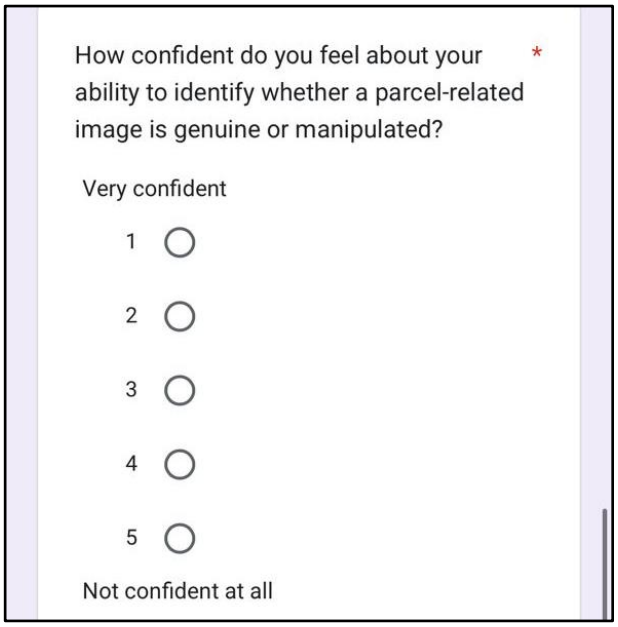
Thank you for your time, participation, and support.

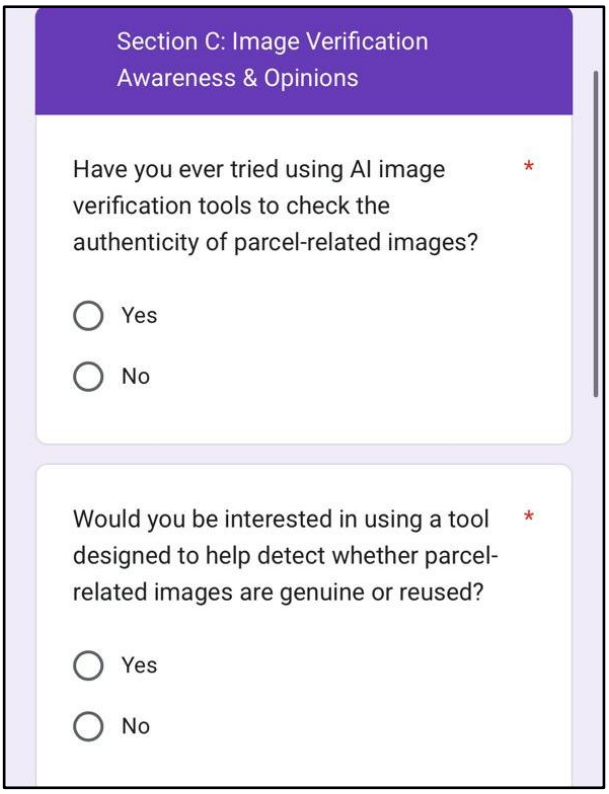
kl2311015320@student.uptm.edu.my
Switch account
Not shared


What is your age group? *
18-25
26-35
36-45
46-55
56 and above
What is your academic level? *
Diploma/Certificate
Bachelor's degree
Master's degree
PhD
Other:


How often do you shop online? *
Rarely
Once a month
2-3 times a month
Weekly
Almost Daily


Have you ever received suspicious messages or emails claiming to be related to parcel deliveries? *
Yes
No
Have you ever experienced a parcel scam personally or do you know someone who has? *
Yes
No, but I know someone who has
No
Before answering this survey, were you aware that scammers often reuse or manipulate images, such as fake receipts or tracking screenshots, in parcel scams? *
Yes
No


How confident do you feel about your ability to identify whether a parcel-related image is genuine or manipulated? *
Very confident
1
2
3
4
5
Not confident at all

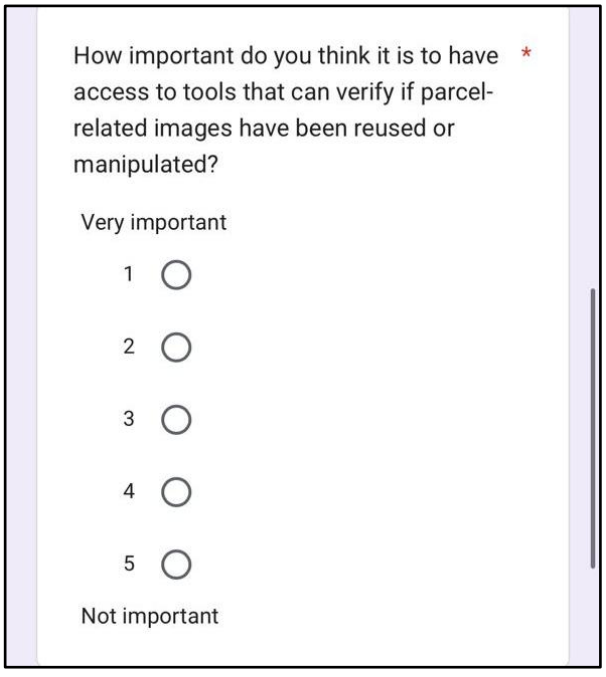

Section C: Image Verification Awareness & Opinions

Have you ever tried using AI image verification tools to check the authenticity of parcel-related images? *

- Yes
- No

Would you be interested in using a tool designed to help detect whether parcel-related images are genuine or reused? *

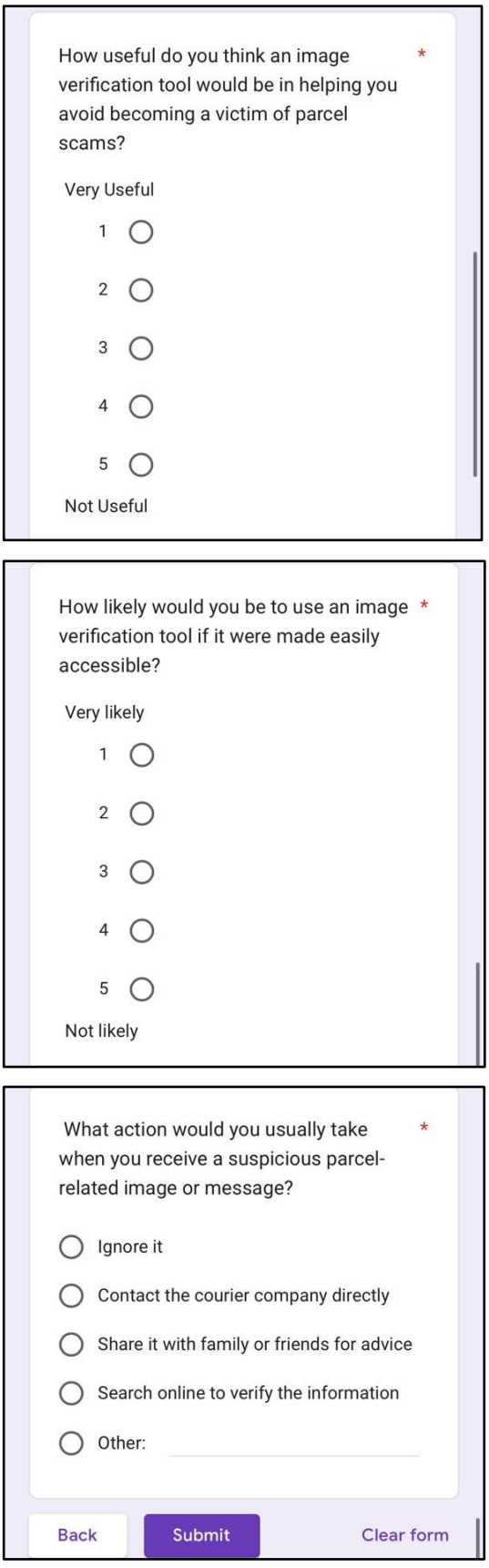
- Yes
- No


How important do you think it is to have access to tools that can verify if parcel-related images have been reused or manipulated? *

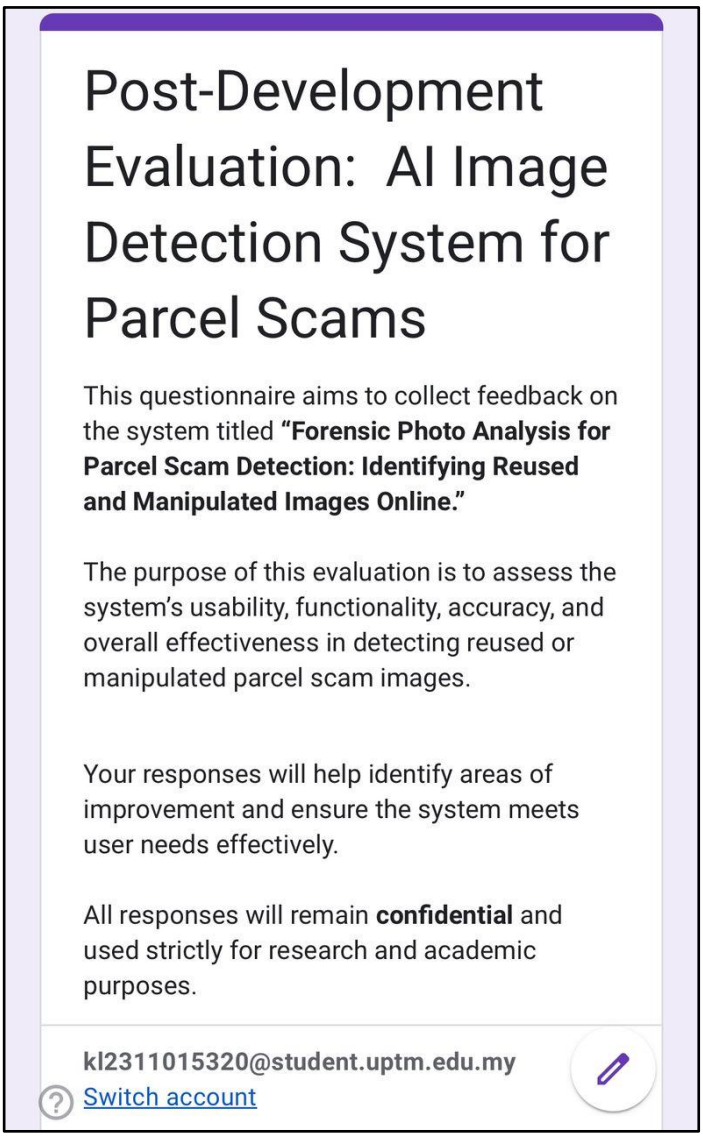
Very important

- 1
- 2
- 3
- 4
- 5

Not important


How useful do you think an image verification tool would be in helping you avoid becoming a victim of parcel scams? *
Very Useful
1
2
3
4
5
Not Useful
How likely would you be to use an image verification tool if it were made easily accessible? *
Very likely
1
2
3
4
5
Not likely
What action would you usually take when you receive a suspicious parcel-related image or message? *
Ignore it
Contact the courier company directly
Share it with family or friends for advice
Search online to verify the information
Other:
Back
Submit
Clear form

II. Questionnaire Post-Development



Post-Development Evaluation: AI Image Detection System for Parcel Scams

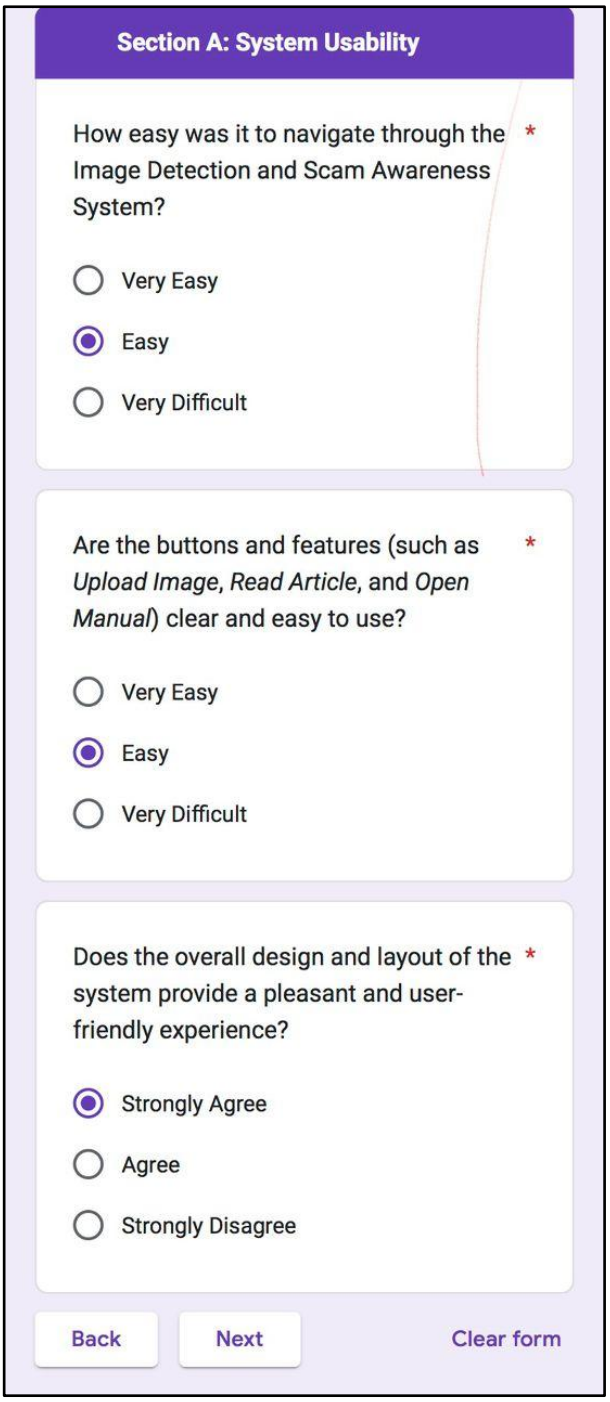
This questionnaire aims to collect feedback on the system titled **"Forensic Photo Analysis for Parcel Scam Detection: Identifying Reused and Manipulated Images Online."**

The purpose of this evaluation is to assess the system's usability, functionality, accuracy, and overall effectiveness in detecting reused or manipulated parcel scam images.

Your responses will help identify areas of improvement and ensure the system meets user needs effectively.

All responses will remain **confidential** and used strictly for research and academic purposes.

kl2311015320@student.uptm.edu.my
Switch account



Section A: System Usability

How easy was it to navigate through the Image Detection and Scam Awareness System? *

- ○ Very Easy
- ◉ Easy
- ○ Very Difficult

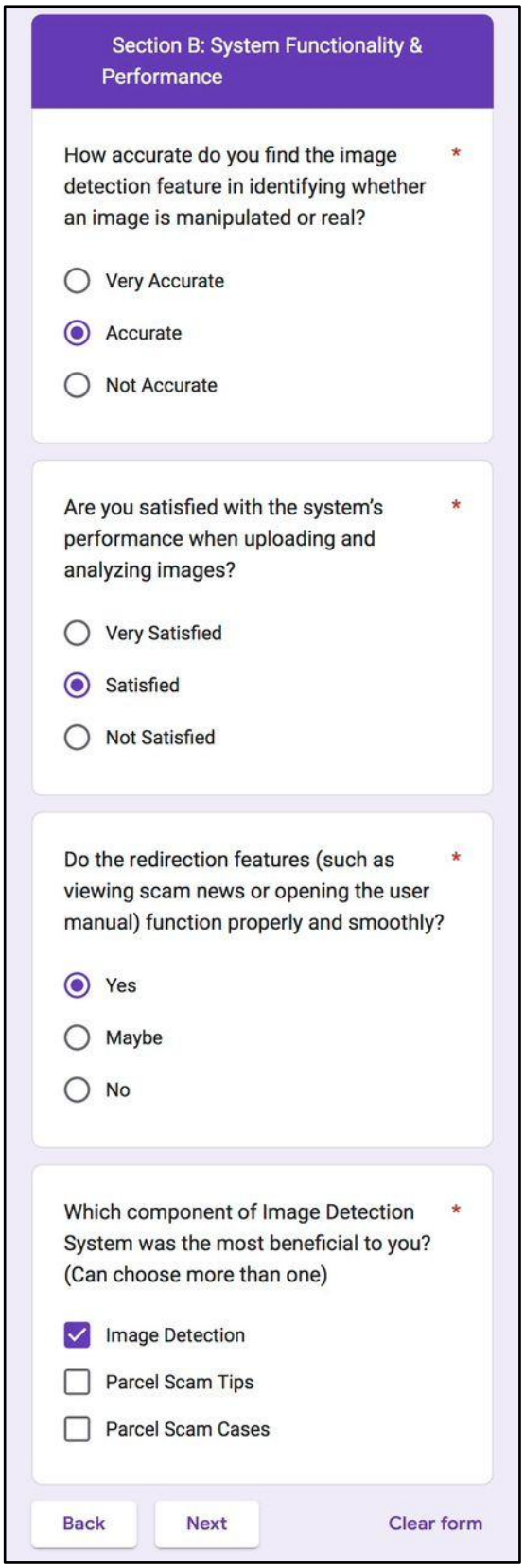
Are the buttons and features (such as *Upload Image*, *Read Article*, and *Open Manual*) clear and easy to use? *

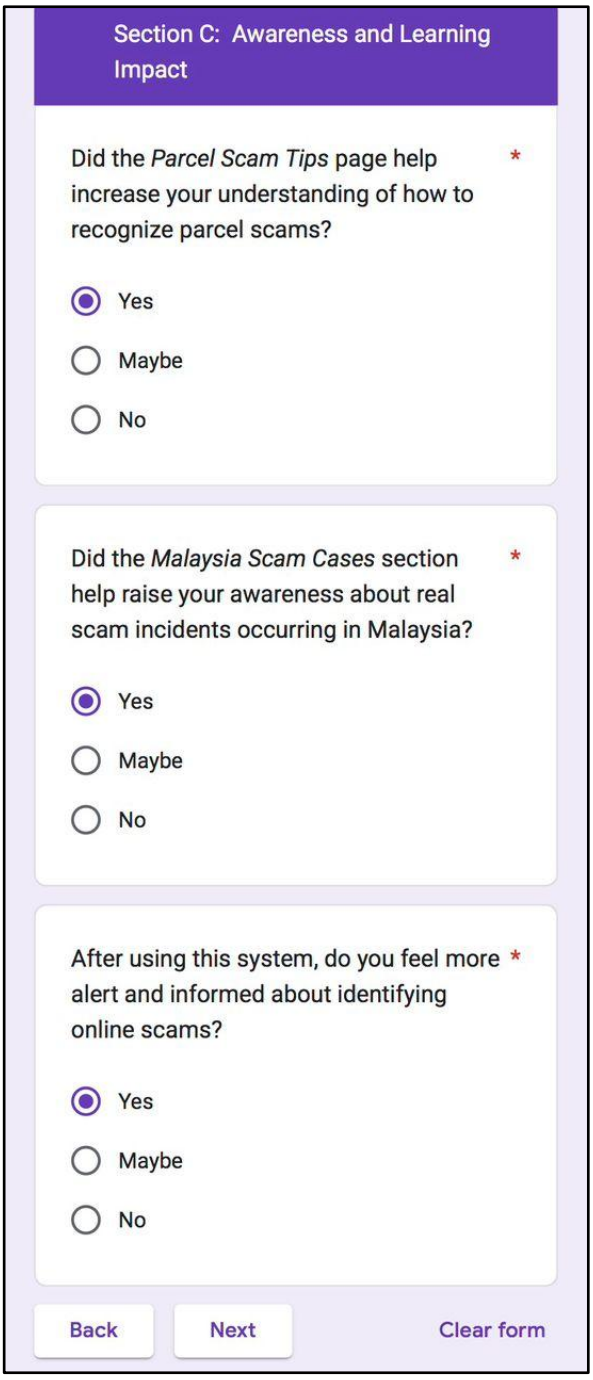
- ○ Very Easy
- ◉ Easy
- ○ Very Difficult

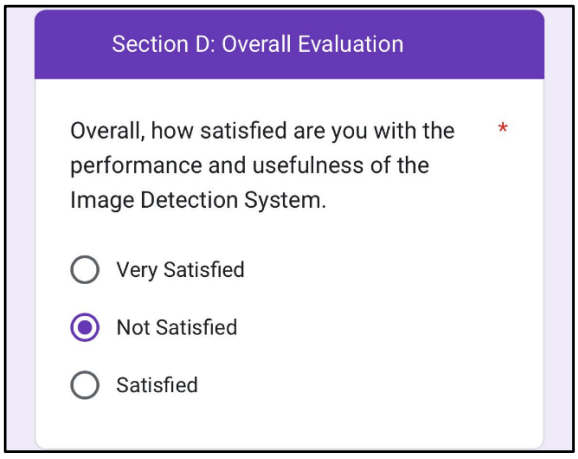
Does the overall design and layout of the system provide a pleasant and user-friendly experience? *

- ◉ Strongly Agree
- ○ Agree
- ○ Strongly Disagree

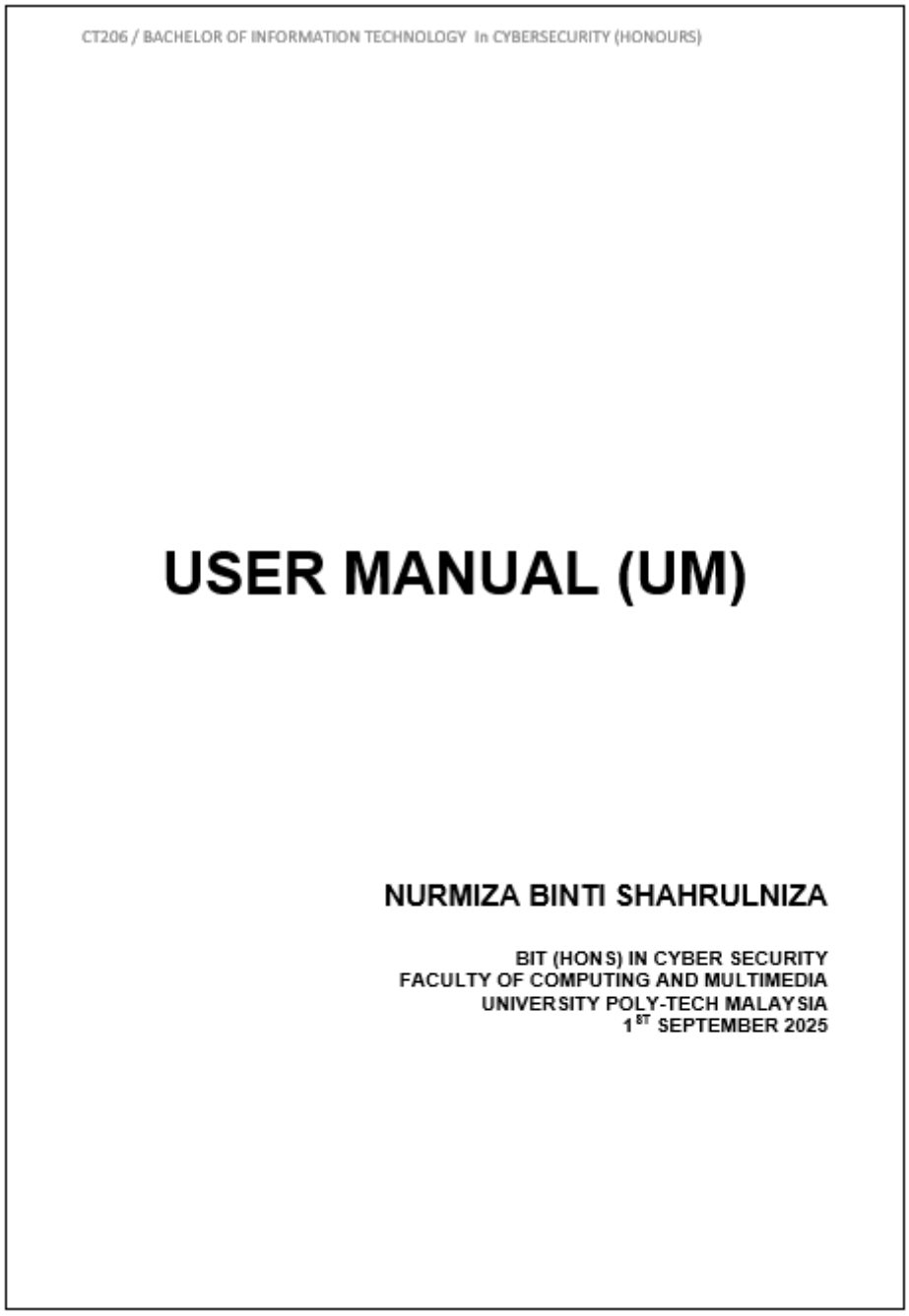
Back Next Clear form


Section B: System Functionality & Performance
How accurate do you find the image detection feature in identifying whether an image is manipulated or real? *
Very Accurate
Accurate
Not Accurate
Are you satisfied with the system's performance when uploading and analyzing images? *
Very Satisfied
Satisfied
Not Satisfied
Do the redirection features (such as viewing scam news or opening the user manual) function properly and smoothly? *
Yes
Maybe
No
Which component of Image Detection System was the most beneficial to you? (Can choose more than one) *
Image Detection
Parcel Scam Tips
Parcel Scam Cases
Back
Next
Clear form


Section C: Awareness and Learning Impact
Did the *Parcel Scam Tips* page help increase your understanding of how to recognize parcel scams? *
Yes
Maybe
No
Did the *Malaysia Scam Cases* section help raise your awareness about real scam incidents occurring in Malaysia? *
Yes
Maybe
No
After using this system, do you feel more alert and informed about identifying online scams? *
Yes
Maybe
No
Back
Next
Clear form


Section D: Overall Evaluation
Overall, how satisfied are you with the performance and usefulness of the Image Detection System. *
Very Satisfied
Not Satisfied
Satisfied

APPENDIX C User Manual (UM)


CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

USER MANUAL (UM)

NURMIZA BINTI SHAHRULNIZA

BIT (HONS) IN CYBER SECURITY
FACULTY OF COMPUTING AND MULTIMEDIA
UNIVERSITY POLY-TECH MALAYSIA
1ST SEPTEMBER 2025

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

Table of Contents

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

1. Homepage – Image Detection Page

What you see:

The main Image Detection page with an upload button.

Instructions:

- Click **"Upload an image to analyze"**.
- Select any parcel-related image (receipt, tracking screenshot, suspicious delivery photo).
- The system will process the image and display the result.

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

2. Image Result – AI Generated
What you see:

A red-highlighted box showing **AI-Generated** result with confidence score and likely AI model.

Instructions:

- Review the **confidence score** and **probability** to understand how likely the image is AI-generated.
- Read the **Detailed Analysis** to see what AI artifacts were detected.
- Click **"Analyze Another Image"** to check another picture.

3. Image Result – Likely Real
What you see:
A green-highlighted box showing **Likely Real** result.

Instructions:

- View the **realness indicators** such as natural lighting, noise, texture.
- Check the **likely source** (Real Photo).
- Click **"Analyze Another Image"** if you want to verify more images.

4. Parcel Scam Tips Page
What you see:
Posters explaining parcel scam detection tips.

Instructions:

- Scroll through each **poster** to learn signs of fake parcel images.
- Use these tips to identify suspicious tracking or delivery images.

5. Malaysia Parcel Scam Cases Page
What you see:
Real reported cases shown as article cards.

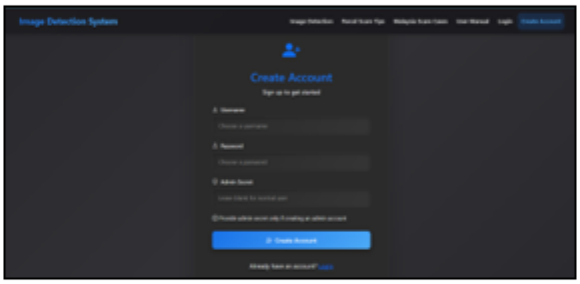
Instructions:

- Click **"Read Article"** under any card to open the scam story.
- Use these cases to understand how scams happen in Malaysia.

6. User Manual Page
What you see:
A button labeled **Open Manual**.

Instructions:

- Click **"Open Manual"** to download the full system user manual in PDF format.



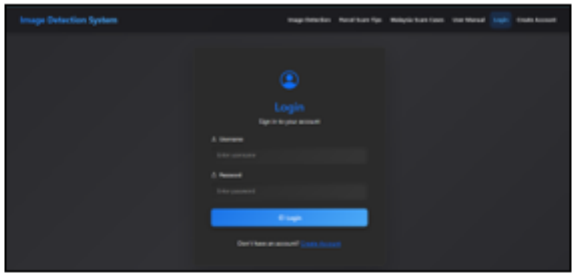
7. Create Account Page

What you see:
User registration form with username, password, and admin secret.

Instructions:

- Fill in **Username** and **Password**.
- Leave **Admin Secret** empty if you are a normal user.
- If you are an admin, enter the secret code.
- Click **Create Account** to register.

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)



8. Login Page
What you see:
Login form.

Instructions:

- Enter your **username** and **password**.
- Click **Login** to access your account.

9. Admin Dashboard
What you see:
Admin homepage with counters and expandable management menus.

Instructions:
Admins can:

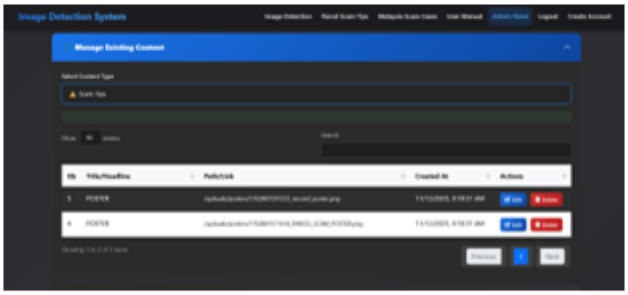
- **Upload new content** (manuals, posters, scam cases)
- **Manage existing content** (edit/delete)
- **View image detection history**
- **Monitor all user uploads**

Click any section to expand and manage the data.

10. Admin – Upload New Content
What you see:
Form for uploading new manuals, scam tips, or scam cases.

Instructions:

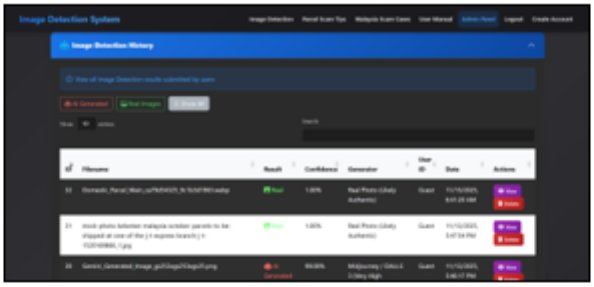
- Choose **Content Type** (PDF, poster, scam case).
- Enter a **title**.
- Click **Choose File** and select the file.
- Click **Upload Content** to publish it.



11. Admin – Manage Existing Content
What you see:
A list of uploaded content with Edit/Delete options.

Instructions:

- Use the dropdown to select **content type** (Scam Tips, Scam Cases, Manuals).
- Click **Edit** to modify title or file.
- Click **Delete** to remove content.
- Use search bar to find specific items.



12. Admin – Image Detection History
What you see:
A table of all scanned images with results.

Instructions:

- Filter results using **AI Generated**, **Real Images**, or **Show All**.
- Click **View** to open the image result in detail.
- Click **Delete** to remove an entry.
- Search for filenames using the search bar.

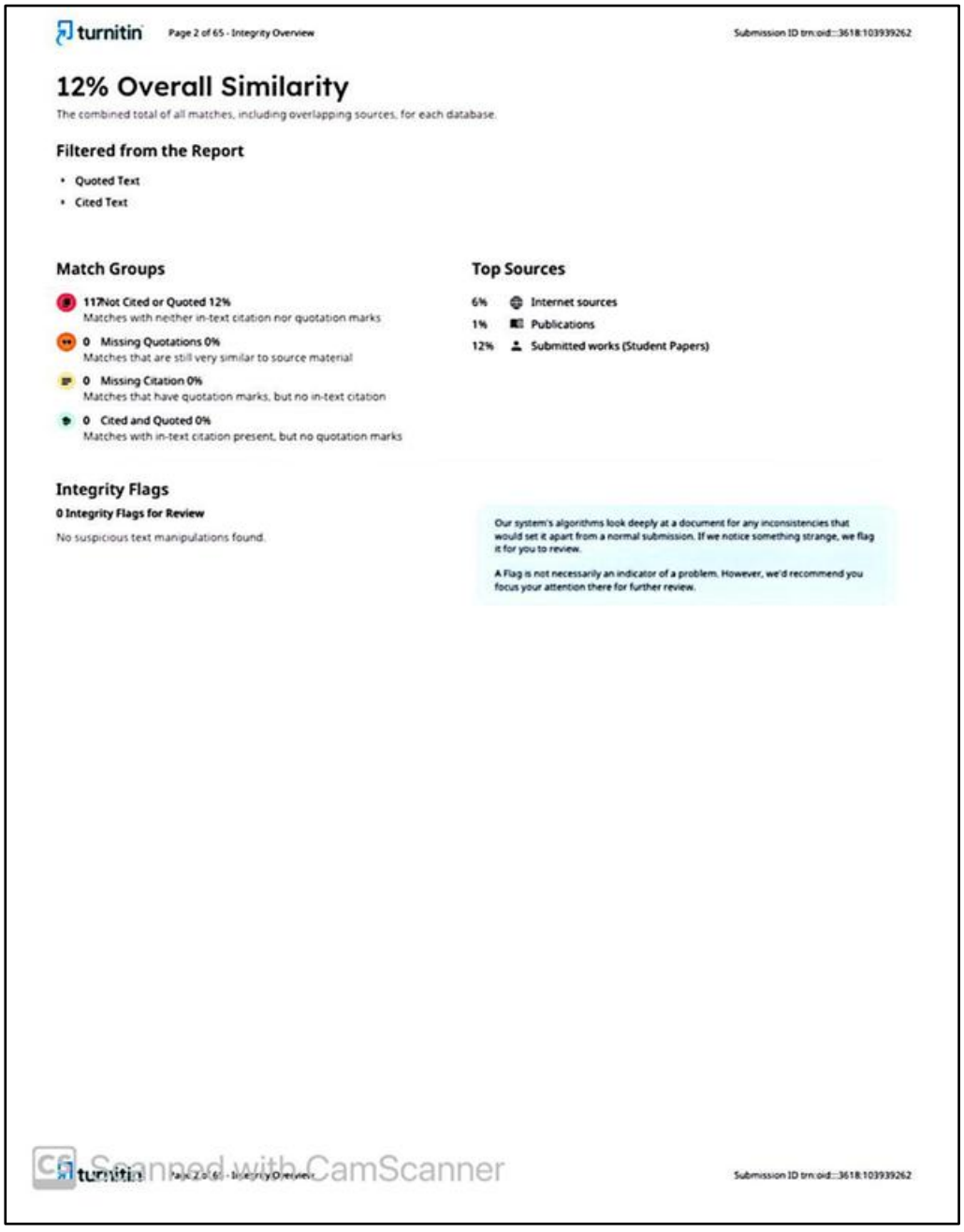
13. Admin – User Upload Monitoring
What you see:
A list of all uploaded images by users.

Instructions:

- View all files uploaded by users with file path and upload date.
- Use this section to track user activity.
- Search specific uploads using the search bar.

APPENDIX D – TURNITIN RESULT

I. Information Security Project 1 (FYP4074)



turnitin Page 2 of 65 - Integrity Overview Submission ID trn:oid:::3618:103939262

12% Overall Similarity

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

Filtered from the Report

- Quoted Text
- Cited Text

Match Groups

- 117 Not Cited or Quoted 12%
  Matches with neither in-text citation nor quotation marks
- 0 Missing Quotations 0%
  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

6% Internet sources
1% Publications
12% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

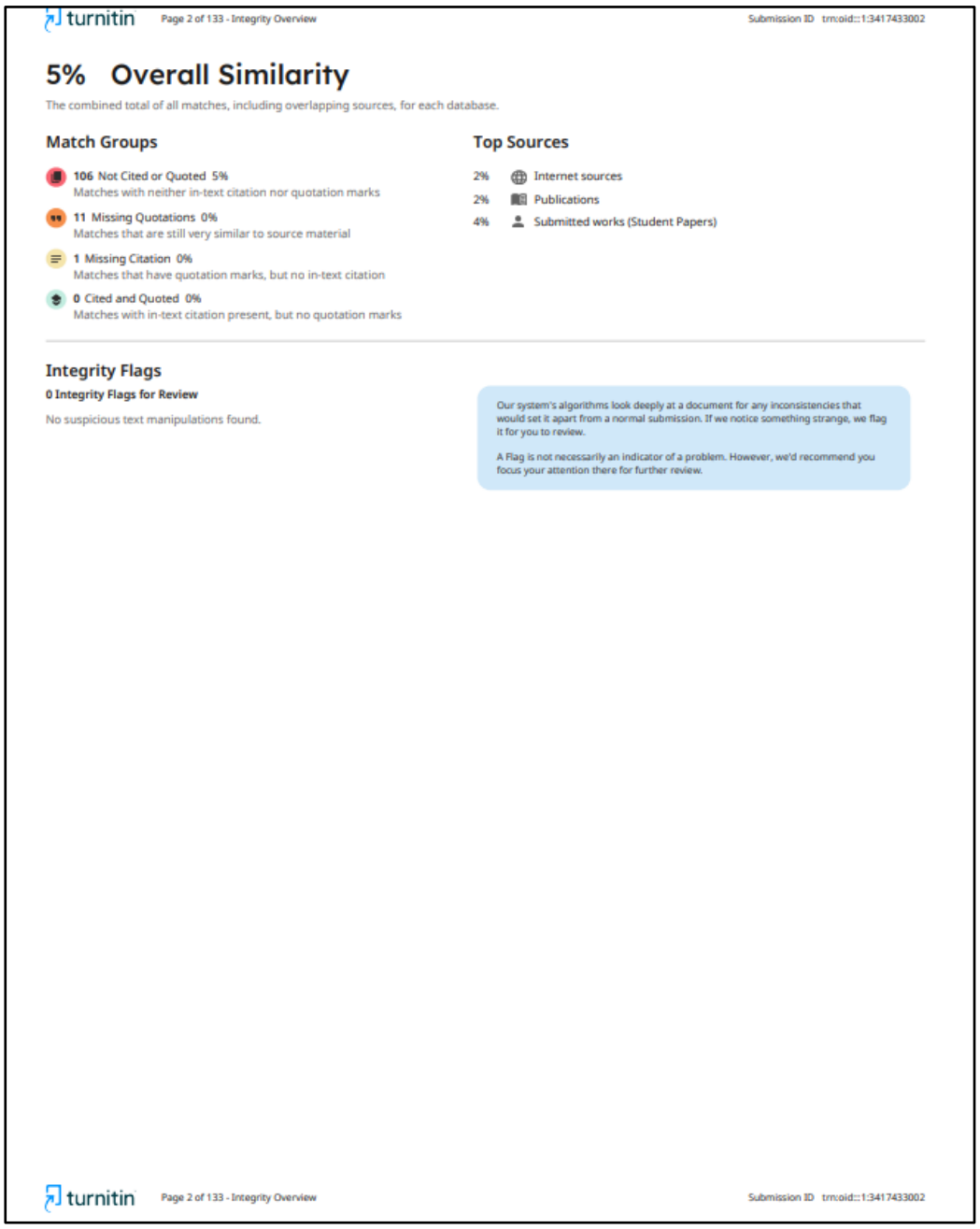
No suspicious text manipulations found.

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

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

turnitin Page 2 of 65 - Integrity Overview Submission ID trn:oid:::3618:103939262

II. Information Security Project 2 (FYP4085)



turnitin Page 2 of 133 - Integrity Overview Submission ID trn:oid:::1:3417433002

5% Overall Similarity

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

Match Groups

- **106** Not Cited or Quoted 5%
  Matches with neither in-text citation nor quotation marks
- **11** Missing Quotations 0%
  Matches that are still very similar to source material
- **1** 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

- 2% Internet sources
- 2% Publications
- 4% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

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

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

turnitin Page 2 of 133 - Integrity Overview Submission ID trn:oid:::1:3417433002

APPENDIX E – LOG BOOKS

I. Information Security Project 1 (FYP4074)

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

UNIVERSITI POLY-TECH MALAYSIA
UPTM

FACULTY OF COMPUTING & MULTIMEDIA (FCOM)

INFORMATION SECURITY PROJECT O1

FYP4074

LOG BOOK

STUDENT'S NAME : NURMIZA BINTI SHAHRULNIZA

ID NO. : AM2311015320

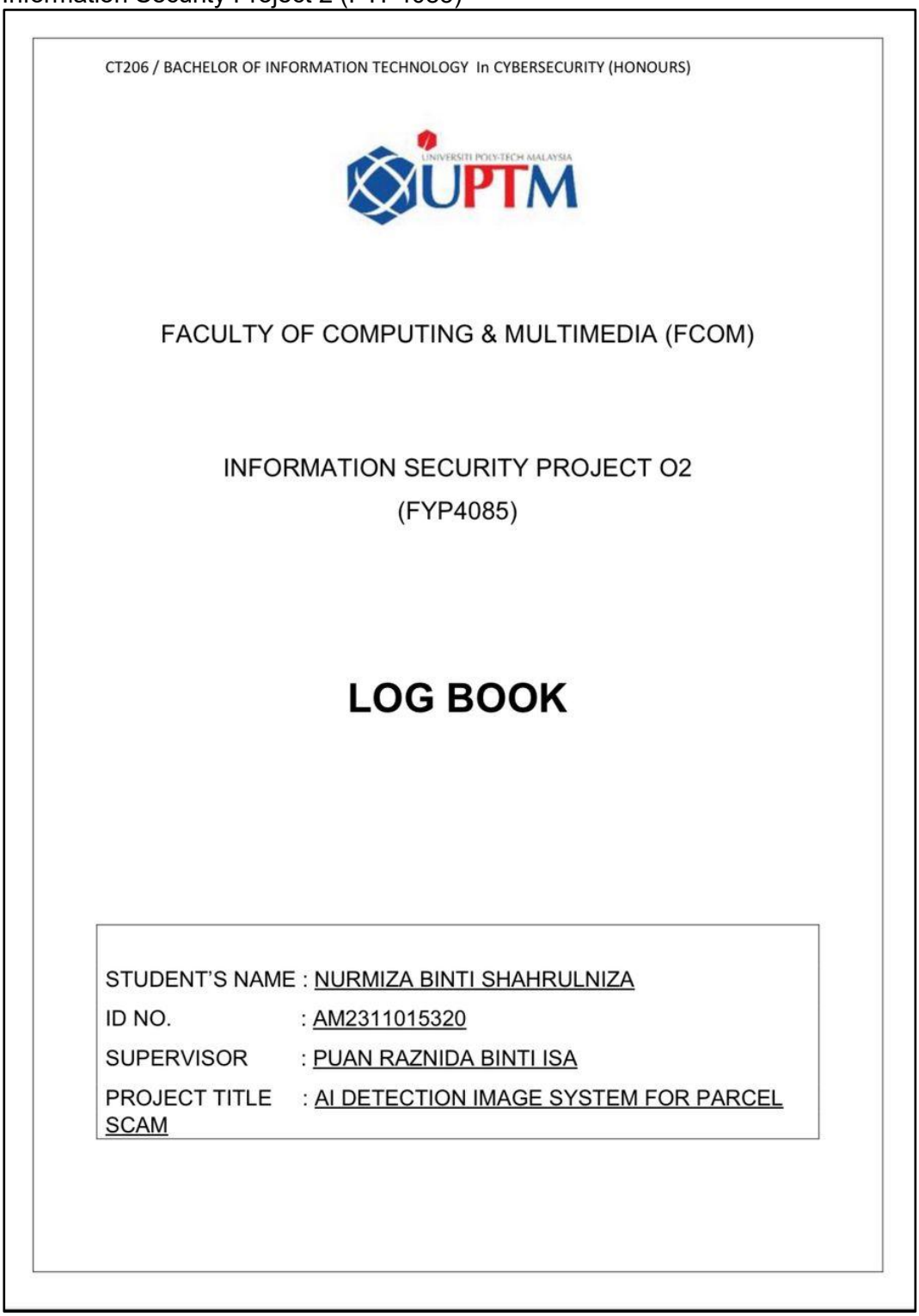
SUPERVISOR : PUAN RAZNIDA BINTI ISA

PROJECT TITLE : AI IMAGE DETECTION SYSTEM FOR PARCEL SCAM

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

Week		Agenda	Next Agenda	Signature (Supervisor / Coordinator)
19/5/2025	1	Initial discussion with supervisor. Confirmation of project title and background research scope.	Prepare draft of problem statement, objectives, and title justification.	Raznida Isa
26/5/2025	2	Discussed draft problem statement, objective, and scope. Supervisor gave feedback on narrowing scope.	Refine and finalize problem statement and project objectives for proposal submission.	Raznida Isa
2/6/2025	3	Submitted updated proposal with revised problem statement and scope. Introduced methodology (Agile).	Prepare questionnaire and identify suitable interview target.	Raznida Isa
9/6/2025	4	Discussed survey structure and interview questions. Supervisor approved direction.	Distribute Google Form to respondents and schedule interview session.	Raznida Isa
16/6/2025	5	Collected initial survey responses. Conducted virtual client interview.	Analyse responses and begin writing Chapter 2: Literature Review and Investigation.	Raznida Isa
23/6/2025	6	Completed questionnaire analysis. Supervisor reviewed Chapter 2 draft.	Revise Chapter 2 and start development of system diagrams (use case, flowchart).	Raznida Isa
30/6/2025	7	Presentation of proposed project to supervisor and examiner (FYP1 evaluation).	Submission of final report, logbook, and presentation slides.	Raznida Isa

II. Information Security Project 2 (FYP4085)



CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

UNIVERSITI POLY-TECH MALAYSIA
UPTM

FACULTY OF COMPUTING & MULTIMEDIA (FCOM)

INFORMATION SECURITY PROJECT O2
(FYP4085)

LOG BOOK

STUDENT'S NAME : NURMIZA BINTI SHAHRULNIZA
ID NO. : AM2311015320
SUPERVISOR : PUAN RAZNIDA BINTI ISA
PROJECT TITLE : AI DETECTION IMAGE SYSTEM FOR PARCEL SCAM

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

Week		Agenda	Next Agenda	Signature (Supervisor / Coordinator)
4/8/2025	1	Start system development planning; finalize functional modules, database structure, and coding tasks.	Begin designing system flow and interface structure (wireframes).	
11/8/2025	2	Create wireframes for user pages (Image Detection, Scam Tips, Scam Cases, User Manual) and admin pages.	Start backend setup for Flask framework and system routing.	
18/8/2025	3	Begin backend development: configure Flask environment, initialize server, and set up database connection (MySQL).	Develop authentication modules (register, login, JWT handling).	
25/8/2025	4	Complete user authentication system using JWT; implement secure password hashing and role-based access control.	Start developing admin features (content upload modules).	
1/9/2025	5	Implement admin upload modules: manuals, scam tips, and scam case images. Integrate secure file validation and sanitization.	Develop Image Detection module using SightEngine API.	
8/9/2025	6	Integrate AI Image Detection API; handle image uploads and detection result storage.	Develop UI for detection results (AI-generated or Likely Real).	
22/9/2025	7	Continue system integration; refine user interface for detection, scam tips, scam cases, and user manual pages.	Start admin monitoring dashboard for detection logs and user uploads.	
29/9/2025	8	Implement Admin Panel Monitoring features (view user upload logs, detection results). Enhance UI consistency across user and admin sections.	Prepare materials to begin writing Chapter 7 (Implementation).	

CT206 / BACHELOR OF INFORMATION TECHNOLOGY In CYBERSECURITY (HONOURS)

6/10/2025	9	Write Chapter 7 (Implementation) including execution platform, tools, interfaces, and security elements.	Begin Chapter 8 (Testing & Evaluation).	
13/10/2025	10	Conduct system testing (functional testing, security testing, interface testing). Collect test results and screenshots.	Complete Chapter 8 documentation and proceed to Chapter 9.	
20/10/2025	11	Write Chapter 9 (Project Management), including Gantt Chart, tools, and budget.	Complete Chapter 10 (Conclusion and Future Work).	
27/10/2025	12	Finalize system; perform final debugging; create UAT Google Form for feedback.	Prepare for final report compilation and presentation materials.	
3/11/2025	13	Finalize report formatting; prepare and rehearse presentation; conduct demo with supervisor and client.	Submit final report and prepare for formal FYP presentation.	
10/11/2025	14	Final report submission and FYP presentation.	(End of FYP2)	

REFERENCES

AARP. (2023). *How to use reverse image search to spot fake photos.* https://www.aarp.org/money/scams-fraud/reverse-image-search-for-fake-photos/

Agilemania. (n.d.). *Functional vs. nonfunctional requirements.* https://agilemania.com/functional-vs-nonfunctional-requirements

Amazon Web Services. (n.d.). *What is SDLC?* https://aws.amazon.com/what-is/sdlc/

American Society for Quality. (2025). *What is a flowchart?* https://asq.org/quality-resources/flowchart

Asana. (n.d.). *What is agile methodology?* https://asana.com/resources/agile-methodology

Atlassian. (n.d.). *Agile project management.* https://www.atlassian.com/agile

Bandhari, P. (2021). *Questionnaire.* Scribbr. https://www.scribbr.com/methodology/questionnaire/

Blog.tineye.com. (n.d.). *How to use TinEye search.* https://blog.tineye.com/how-to-use-tineye-search/

Boddis, L. (2024). *Fraud prevention: Combating the rise of deepfake & image manipulation.* Forbes. https://www.forbes.com/councils/forbesbusinessdevelopmentcouncil/2024/11/18/fraud-prevention-combating-the-rise-of-deepfake--image-manipulation/

Britannica. (n.d.). *Scam.* https://www.britannica.com/dictionary/scam

Cacoo Staff. (2021). *How a UML use case diagram can benefit any process.* Nulab. https://nulab.com/learn/software-development/how-a-uml-use-case-diagram-can-benefit-any-process/

Creatus. (n.d.). *Acer Aspire 3 A315-24P Ryzen 3 7320U Laptop.* https://www.creatus.com.bd/acer-aspire-3-a315-24p-ryzen-3-7320u-laptop

Dalitso Kuphanga. (2024). *Questionnaires in research: Their role, advantages, and main aspects.* ResearchGate. https://www.researchgate.net/publication/378868278_Questionnaires_in_Research_Their_Role_Advantages_and_Main_Aspects

Enago Life Sciences. (2024). *From detection to prevention: Countering image fraud.* https://lifesciences.enago.com/blogs/from-detection-to-prevention-countering-image-fraud

Feedzai. (2024). *Deepfake fraud: A threat to banks and consumers.* https://www.feedzai.com/blog/deepfake-fraud/

FreeBSD Foundation. (n.d.). *How to use VS Code on FreeBSD.* https://freebsdfoundation.org/resource/how-to-use-vs-code-on-freebsd/

GeeksforGeeks. (n.d.). *Functional vs non-functional requirements.* https://www.geeksforgeeks.org/software-engineering/functional-vs-non-functional-requirements/

HDFC Bank. (2023). *What is parcel scam & its type.* https://www.hdfcbank.com/personal/resources/learning-centre/vigil-aunty/what-parcel-scams

Hecker, J., & Kalpokas, N. (2024). *Advantages of research interviews.* ATLAS.ti. https://atlasti.com/guides/interview-analysis-guide/advantages-of-research-interviews

Hostinger. (n.d.). *What is VS Code?* https://www.hostinger.com/tutorials/what-is-vs-code

Hughes, M. (2019, August 28). *Developer forks GIMP image editor over naughty name.* The Next Web. https://thenextweb.com/news/developer-forks-gimp-image-editor-over-naughty-name

IBM. (2024). *What is BPMN?* https://www.ibm.com/think/topics/bpmn

Ironhack. (n.d.). *Functional vs. non-functional requirements: Understanding the core differences.* https://www.ironhack.com/gb/blog/functional-vs-non-functional-requirements-understanding-the-core-differences-and

Laoyan, S. (2025, February 20). *What is agile methodology?* Asana. https://asana.com/resources/agile-methodology

Loudbench. (n.d.). *Python.* https://loudbench.com/python/

Lucidchart. (n.d.). *What is agile methodology?* https://lucid.co/blog/what-is-agile-methodology

MDPI. (2023). *A survey on digital image forgery detection.* Applied Sciences, 13(19), 10980. https://www.mdpi.com/2076-3417/13/19/10980

Maverick-OS. (2023). *Image manipulation: We can tell when you're faking it.* https://www.maverick-os.com/news-events/news/image-manipulation-we-can-tell-when-youre-faking-it/

Medium. (n.d.). *FotoForensics: The best free OSINT image tool.* https://medium.com/@samuel.i.steers/fotoforensics-the-best-free-osint-image-tool-0c98b9c45a58

PMCID. (2020). *Interpol review of imaging and video 2016–2019.* National Library of Medicine. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7770461/

RBL Bank. (2023). *Parcel scams: How to spot and avoid them.* https://www.rblbank.com/blog/banking/safe-banking/parcel-scams-how-to-spot-and-avoid-them

Research Methods Community. (2021, April 26). *Collecting data with interviews.* Sage. https://researchmethodscommunity.sagepub.com/blog/collecting-data-with-interviews

SearchCIO. (2022). *Business process modeling notation (BPMN).* TechTarget. https://www.techtarget.com/searchcio/definition/Business-Process-Modeling-Notation

SearchFacts. (n.d.). *Yandex Webmaster Tools.* https://searchfacts.com/yandex-webmaster-tools/

Teradata. (n.d.). *What is Python?* https://www.teradata.com/glossary/what-is-python

The Bureau of Investigative Journalism. (2024). *What is a deepfake and what are the different types?* https://www.thebureauinvestigates.com/stories/2024-03-07/what-is-a-deepfake-and-what-are-the-different-types

The Star. (2024). *Parcel scam costs Kota Tinggi woman RM238,000.* https://www.thestar.com.my/news/nation/2024/11/14/parcel-scam-in-costs-kota-tinggi-woman-rm238000

The Star. (2025). *Report: Online shoppers hit by increasing number of parcel scams.* https://www.thestar.com.my/tech/tech-news/2025/06/12/report-online-shoppers-hit-by-increasing-number-of-parcel-scams-more-than-3500-cases-in-q1-2025

Tungsten Automation. (2024). *Push back against fraud with AI photo analysis.* https://www.tungstenautomation.com/learn/blog/push-back-against-fraud-with-ai-photo-analysis

University of Edinburgh. (2024). *Literature review.* Institute for Academic Development. https://institute-academic-development.ed.ac.uk/study-hub/learning-resources/literature-review

Visual Paradigm. (n.d.). *What is use case diagram?* https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/

Vocabulary.com. (n.d.). *Parcel and scam definitions.* https://www.vocabulary.com/lists/220683

Wrike. (n.d.). *What is agile methodology in project management?* https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/

Yahoo News. (2024). *Malaysians scammed using parcel scams.* https://malaysia.news.yahoo.com/paying-nothing-malaysians-scammed-using-230000479.html

Itexus. (2024). *The implementation phase in SDLC: A comprehensive guide*. Itexus – Custom Software / Apps Development Company. https://itexus.com/the-implementation-phase-in-sdlc-a-comprehensive-guide/

Knapp, M. (2023). *Acer Aspire 3 (A315-24P) review*. PCMag. https://www.pcmag.com/reviews/acer-aspire-3-a315-24p

Muchmore, M. (2025). *Microsoft Windows 11 preview*. PCMag. https://www.pcmag.com/reviews/microsoft-windows-11

Gurnov, A. (2024). *What is a project schedule in project management?* Wrike. https://www.wrike.com/project-management-guide/faq/what-is-a-project-schedule-in-project-management/

Malsam, W. (2025). *Articles by William Malsam*. ProjectManager. https://www.projectmanager.com/author/williammalsam

Schwartz, B. (2025). *Importance of a work breakdown structure in project management: Benefits of WBS*. ProjectManager. https://www.projectmanager.com/blog/importance-of-a-work-breakdown-structure-in-project-management-benefits-of-wbs

Bernama. (2025). *Retailers urged to adopt AI tools to reduce scam cases*. https://bernama.com/en/news.php?id=2457140

Papasavva, A., Lundrigan, S., Lowther, E., Johnson, S., Mariconti, E., Markovska, A., & Tuptuk, N. (2025). Applications of AI-based models for online fraud detection and analysis. *Crime Science, 14*(1). https://doi.org/10.1186/s40163-025-00248-8

Arachchilage, N. A. G., & Love, S. (2014). Security awareness of computer users: A literature review. *International Journal of Computer Applications, 88*(9).

Bayar, B., & Stamm, M. C. (2016). A deep learning approach to universal image manipulation detection. *ACM IH&MMSec*.

Johnson, M. K., & Farid, H. (2005). Exposing digital forgeries by detecting inconsistencies in lighting. *IEEE Workshop on Multimedia Signal Processing*.

Popescu, A. C., & Farid, H. (2005). Exposing digital forgeries by detecting duplicated image regions. *Dartmouth College Technical Report*.

Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). FaceForensics++: Learning to detect manipulated facial images. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Shadiqe, J. (2024). Two lose RM394,000 in separate cases of contraband parcel scam. *New Straits Times*. https://www.nst.com.my/news/crime-courts/2024/04/1040739/two-lose-rm394000-separate-cases-contraband-parcel-scam

Sheng, S., Holbrook, M., Kumaraguru, P., Cranor, L. F., & Downs, J. (2010). Who falls for phish? *CHI 2010 Proceedings.*

TodayOnline. (2024). *Teenager in Malaysia loses S$89,000 to fake parcel scam syndicate*. https://www.todayonline.com/world/teenager-malaysia-loses-s89000-fake-parcel-scam-syndicate-which-also-tried-demand-ransom-his-family-2279386

Verdoliva, L. (2020). Media forensics and deepfakes: An overview. *IEEE Journal of Selected Topics in Signal Processing, 14*(5), 910–932.

Zalani, A. (2025). Paying for nothing: How Malaysians are scammed using parcels they never ordered. *Malay Mail*. https://www.malaymail.com/news/malaysia/2025/06/17/paying-for-nothing-how-malaysians-are-scammed-using-parcels-they-never-ordered/179959

Wrike. (2023). *The benefits & advantages of Agile*. https://www.wrike.com/agile-guide/benefits-of-agile/