**AI GENERATED IMAGE DETECTION & HARM**

**PREVENTION SYSTEM**

Higher National Diploma in Software

Engineering 24.2F

Emerging Technology

VITHANA D K W COHNDSE242F-032

RODRIGO P H V V COHNDSE242F-053

FERNANDO M M S COHNDSE242F-054

WANIGASOORIYA J J M K N COHNDSE242F-062



School of Computing and Engineering

National Institute of Business Management

Colombo-07

Project title: AI GENERATED IMAGE DETECTION & HARM PREVENTION SYSTEM

Student index and name:

COHNDSE242F-032 VITHANA D K W

COHNDSE242F-053 RODRIGO P H V V

COHNDSE242F-054 FERNANDO M M S

COHNDSE242F-062 WANIGASOORIYA J J M K N

Name of the program: Higher National Diploma in Software Engineering 24.2F

Institution: School of Computing and Engineering National Institute of Business Management

Date: 10/27/2025

Contents

[Chapter 1: Introduction 4](#_Toc212375533)

[1.1 Introduction Application 4](#_Toc212375534)

[1.2 Problem Definition 4](#_Toc212375535)

[1.3 Proposed Solution 5](#_Toc212375536)

[1.4 Chapter Summary 5](#_Toc212375537)

[Chapter 2: Methodology 6](#_Toc212375538)

[2. 1 Introduction 6](#_Toc212375539)

[2.2 Software Process Model 6](#_Toc212375540)

[2.3 Technologies Used 6](#_Toc212375541)

[2.4 Chapter Summary 7](#_Toc212375542)

[Chapter 3: Analysis 8](#_Toc212375543)

[3.1 Introduction 8](#_Toc212375544)

[3.2 UML Diagram 8](#_Toc212375545)

[3.3 Chapter Summary 22](#_Toc212375546)

[Chapter 4: Implementation and Finalized Design 23](#_Toc212375547)

[4.1 Introduction 23](#_Toc212375548)

[4.2 System Implementation 23](#_Toc212375549)

[4.3 Finalized Design 23](#_Toc212375550)

[4.4 User Interfaces 24](#_Toc212375551)

[4.5 Challenges Encountered 29](#_Toc212375552)

[4.6 Chapter Summery 29](#_Toc212375553)

[Chapter 5: Business Model 30](#_Toc212375554)

[5.1 Introduction 30](#_Toc212375555)

[5.2 Business Model Canvas 30](#_Toc212375556)

[5.3 Future Business Expansion 30](#_Toc212375557)

[5.4 Chapter Summery 31](#_Toc212375558)

[Chapter 6: References 32](#_Toc212375554)

# Chapter 1: Introduction

## 1.1 Introduction Application

Generative artificial intelligence (AI) has grown so quickly that it has changed the way digital information is made and shared. With the use of tools like Midjourney, Stable Diffusion, DALLE, users can now create incredibly realistic synthetic images with a level of ease never before possible. These technologies have enormous possibilities for design, entertainment, education and innovation. But these advantages come with significant hazards and difficulties as well.

Malicious uses of AI-generated photos include disseminating false information, creating false evidence. Posing as people and swaying public opinion. Such abuse jeopardizes individual reputations, presents more general risks to societal stability, politics and journalism, and creates false evidence. These issues are exacerbated by the lack of easily available and trustworthy verification methods for regular users, which leaves the general public open to fraud.

## 1.2 Problem Definition

With the explosive advancement of generative AI technologies, it has become increasingly easy to create highly realistic fake images. Tools like Midjourney, DALL·E, and Stable Diffusion allow users to generate synthetic content that is often indistinguishable from real photography. While these innovations have powerful creative applications, they also present significant ethical and societal challenges.

One of the most pressing concerns is the misuse of these technologies to produce deceptive or harmful content. AI-generated images can be used to defame individuals, impersonate public figures, fabricate evidence, and disseminate misinformation. These capabilities pose threats to personal reputations, political stability, journalistic integrity, and public trust in visual media.

Unfortunately, there is currently no accessible, widely used solution that allows non-expert users to verify whether an image is authentic or generated by AI. While some academic and corporate tools exist, they are often not user-friendly, lack transparency, or are inaccessible to the public.

## 1.3 Proposed Solution

Our proposed solution consists of a web application with a strong backend that integrates a pre-trained AI model capable of analyzing images and predicting whether they are AI-generated or real. The model will rely on advanced deepfake, or GAN-detection techniques trained on large datasets, potentially using frameworks and weights hosted on platforms such as Hugging Face.

**Key Features**

* **Image Upload & Scan**

Users can drag and drop or upload images from their device. Once uploaded, the system will process the image and return a result indicating whether it is likely real or AI-generated.

* **Detection Result & Confidence Score**

The backend will return to a prediction along with a confidence percentage (e.g., 92% AI-generated). This helps users assess the likelihood of manipulation.

* **Report Harmful Images**

Users can flag harmful or deceptive images, optionally providing context. Reports are logged in the database and can be reviewed by moderators or system administrators.

* **Scan History**

Logged-in users will have access to their personal scan history, allowing them to track past checks and view previous results.

* **Educational** **Resources**

To support digital literacy, the site will include links to articles, guides, and video explainers from trusted institutions such as the Electronic Frontier Foundation, MIT Media Lab, and journalism verification tools.

## 1.4 Chapter Summary

In this chapter we discussed the idea behind our project and explained why we decided to build this system. And discusses the problems with the growing popularity of AI generated imagery and offers a solution. Also listed the goals we want to achieve and gave a basic idea of how the solution will work. Solution describes a specialized online application that empowers people by offering picture verification, a reporting system and instructional materials.

# Chapter 2: Methodology

## 2. 1 Introduction

This chapter explains how we planned and completed our project. The methodical process used to create the web application for AI generated picture identification and the process is intended to guarantee the development of a reliable, expandable and user focused solution. And describe the tools and development models we used and describe how we gathered the information needed to understand what features the system should have.

**Data Collection Methods**

* We performed interviews with journalists, social media users, and educators who often interact with digital content in order to comprehend user difficulties and skepticism.
* We sent a poll to a larger online community to get input on current verification technologies and user requirements.
* We spoke with experts in digital forensics and AI researchers to obtain technical and operational insights.
* We assessed current image verification platforms and scholarly detection techniques to assess the advantages and disadvantages of competing strategies.
* Utilizing Public Datasets: The primary data will come from sizable, publicly accessible datasets on websites such as Hugging Face and Kaggle, which comprise thousands of labeled "real" and "AI-generated" images.
* Diverse Generator Sources: To guarantee that the model performs well in general, the AI-generated portion of the data will contain images produced by a range of models, such as GANs (like StyleGAN), Diffusion Models (like Stable Diffusion, DALL·E), and other architectures.

## 2.2 Software Process Model

Agile Development Model was applied, and it divides work into manageable chunks, each of which is completed and tested before going on to the next. This allowed us to get feedback early and make changes if needed. We held regular meetings with our group to check progress and solve any problem quickly. This made the project more organized and easier to manage.

## 2.3 Technologies Used

* Frontend: ReactJS / HTML-CSS-JS
* Backend: Python (Flask) or Node.js
* Database: Firestore / Cloudinary
* AI Model: Pre-trained DeepFake-V2detection model (from Hugging Face)

## 2.4 Chapter Summary

In this chapter, we explained how we planned, developed and tested our web application. The method is supported by rigorous testing strategies for both software and AI components, an Agile Process Model for flexible and iterative development and a strong Data Collection strategy to train genes. We used simple tools and an easy-to-follow development method. This helped us create a system that is functional and user friendly.

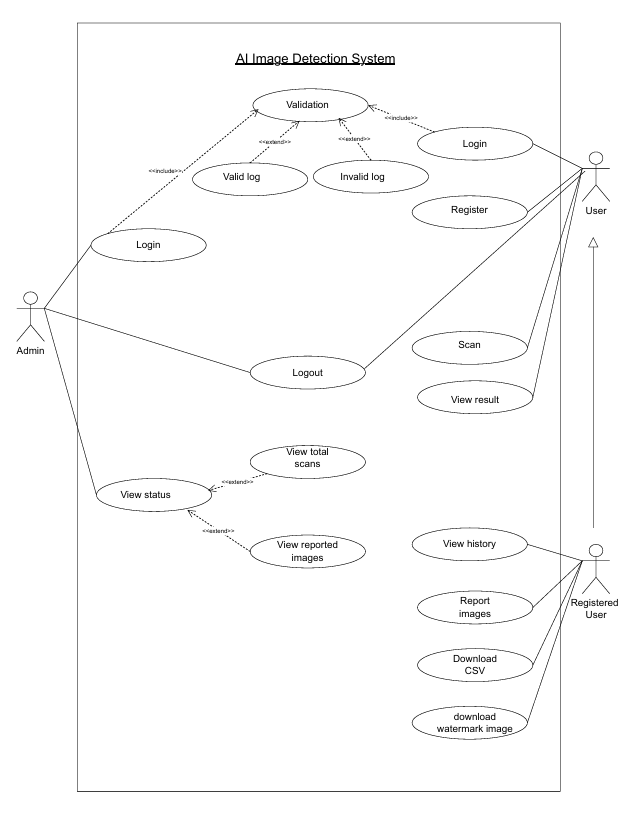
# Chapter 3: Analysis

## 3.1 Introduction

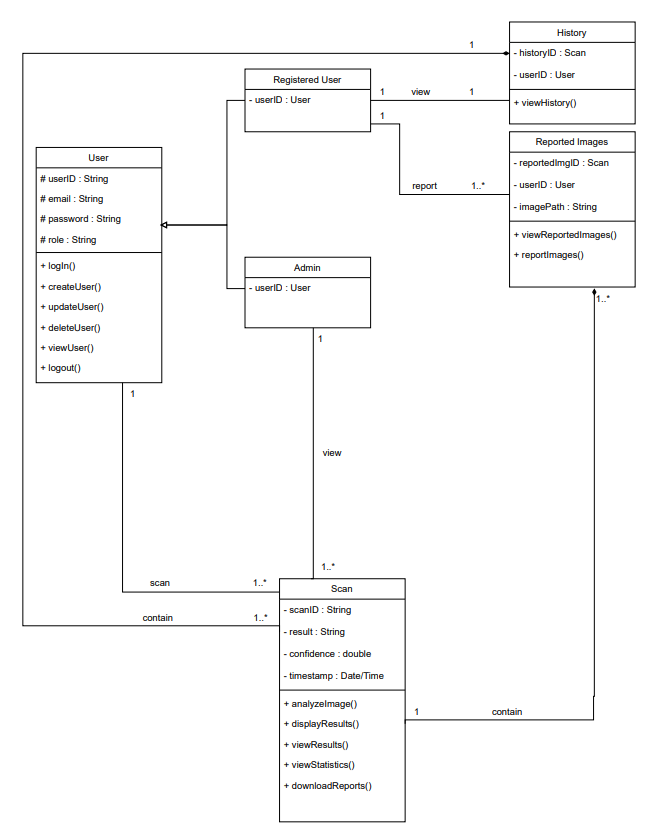
The method of problem analysis was discussed in this part. UML diagrams are covered. The use case diagram for this application is detailed in the UML Diagrams section. It also discusses sequence diagrams (each use case), class diagrams and the application's ER Diagram.

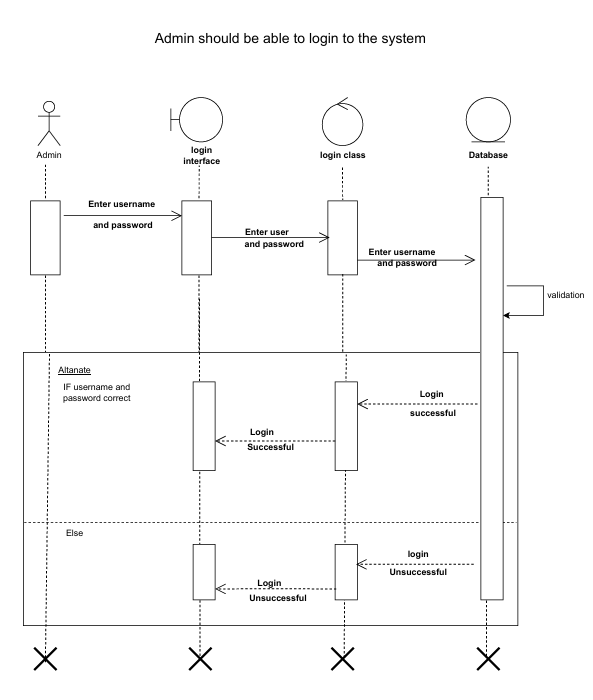
## 3.2 UML Diagram

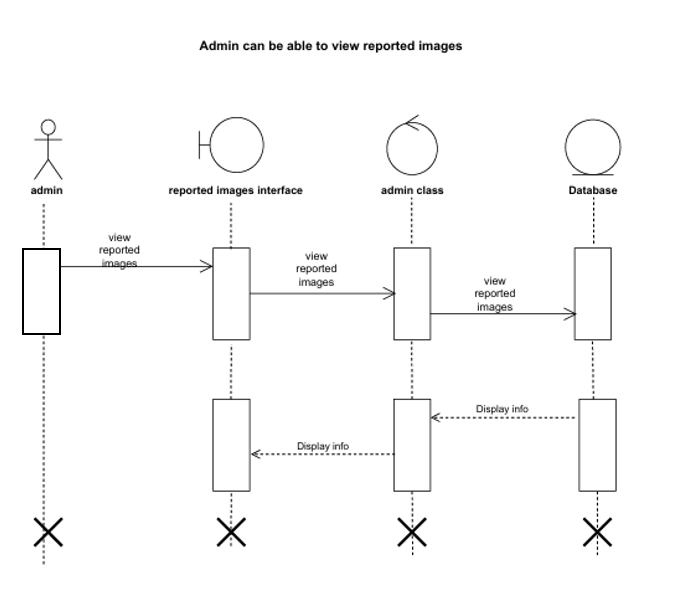
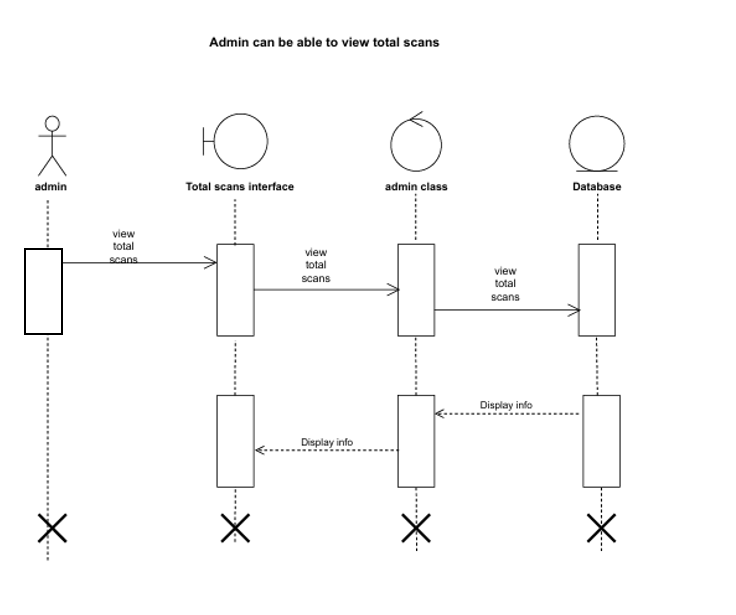
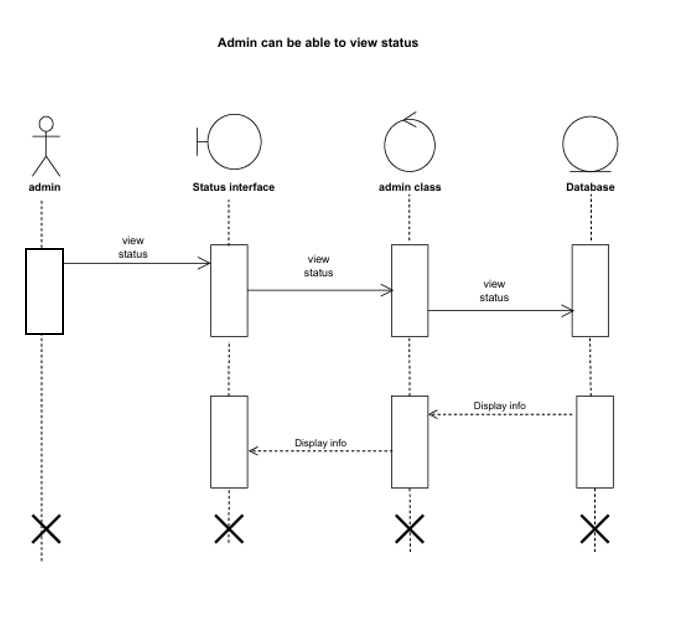
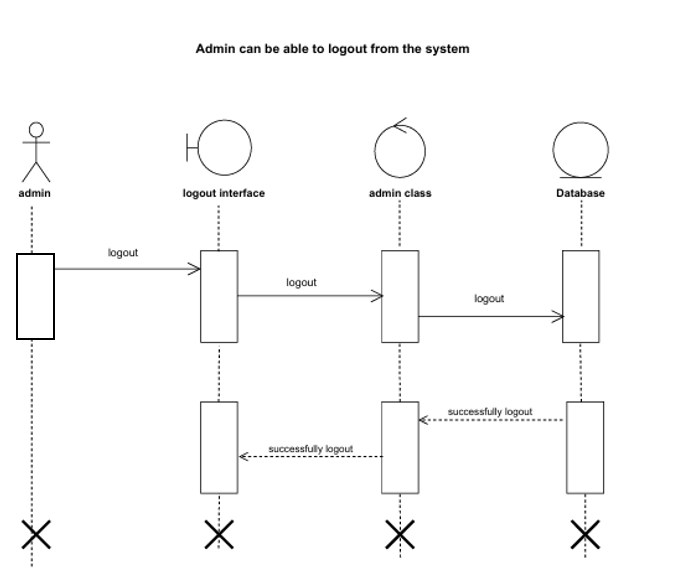
Use Case Diagram

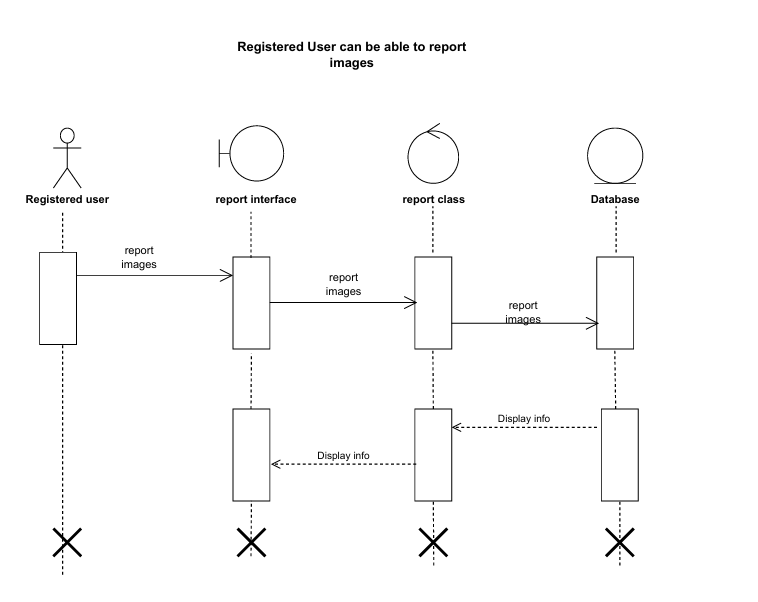
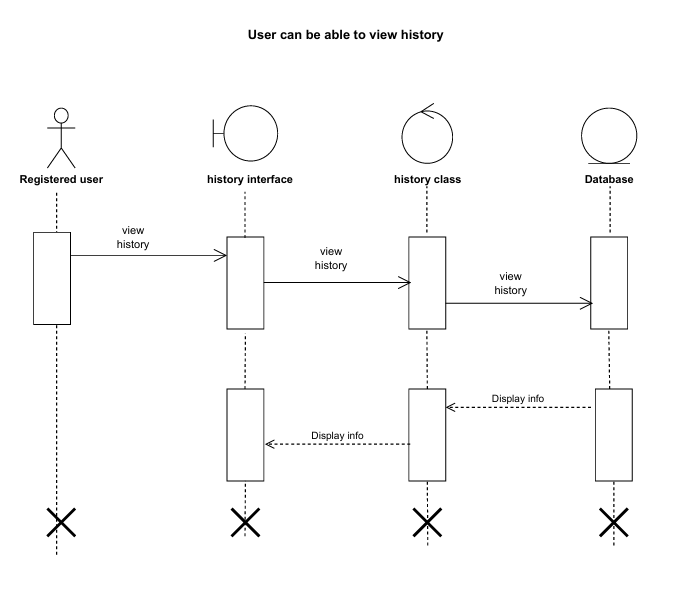
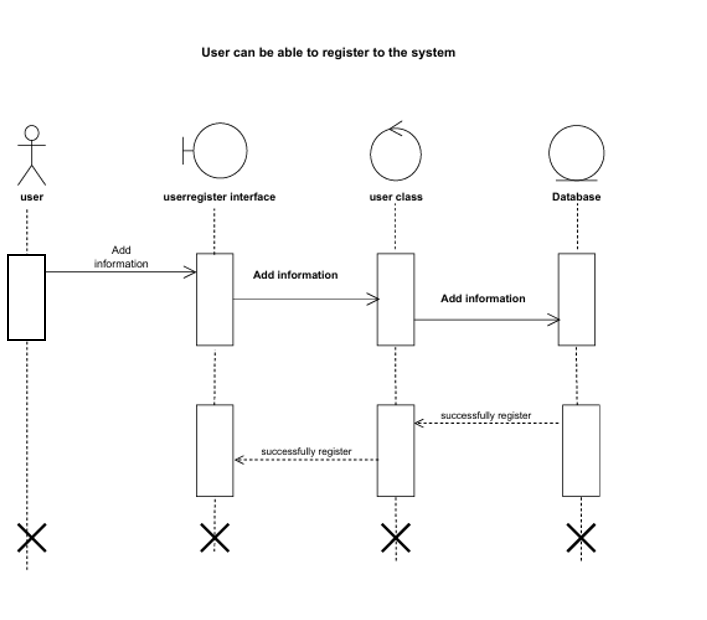
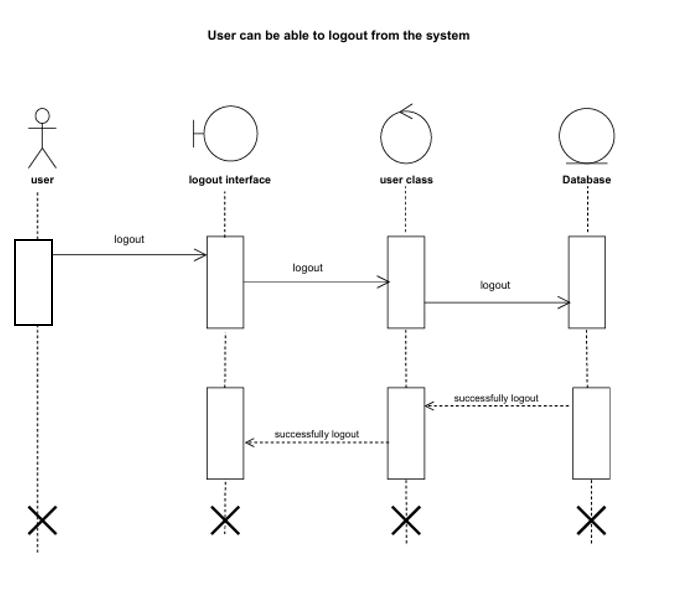
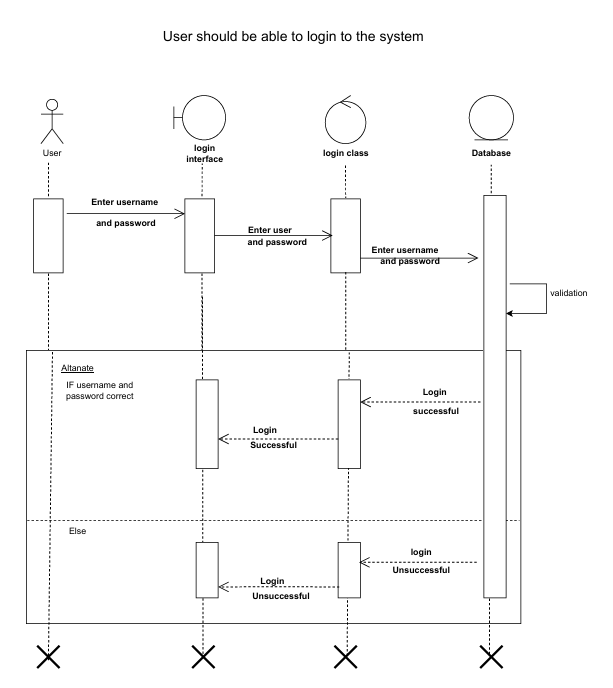


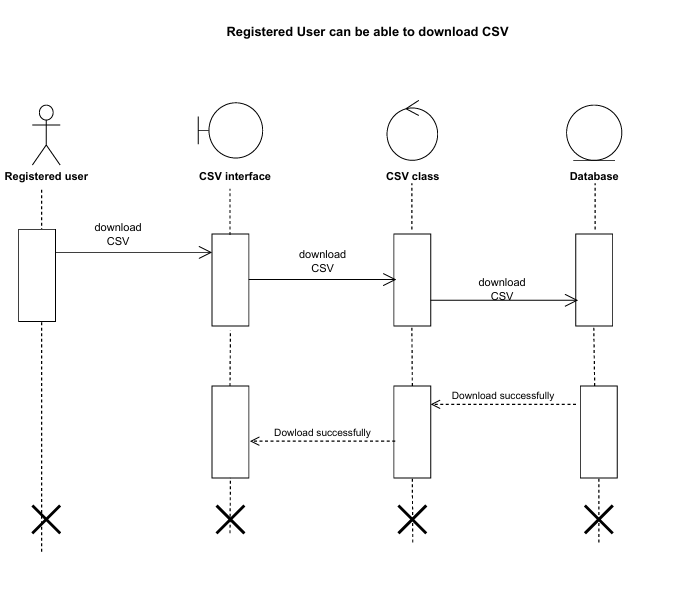
Class Diagram



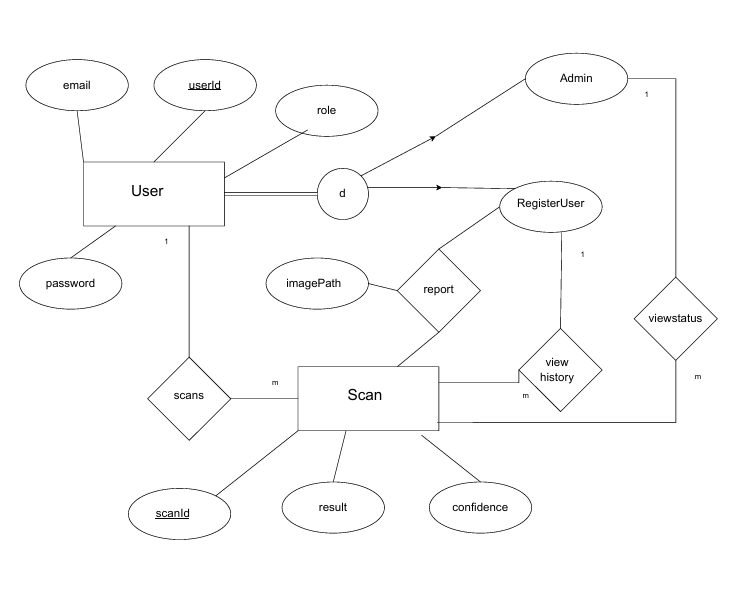
Sequence Diagram







ER Diagram



## 3.3 Chapter Summary

Using key diagrams that illustrate the system's behavior, structure, and data relationships, this chapter offers a thorough picture of the system design. The Use Case Diagram shows how the system's actors and Users—interact with its features, which include handling service data, uploading photos, and managing user profiles. Through connections like include and extend, these interactions specify the essential characteristics and dependencies.

# Chapter 4: Implementation and Finalized Design

## 4.1 Introduction

This chapter explains how the proposed system, DeepFakeShield, was practically implemented and transformed from a conceptual design into a working application. It covers the frontend, backend, and database integration processes, the technologies used during development, and the finalized user interface design of the system.

## 4.2 System Implementation

The implementation stage focused on converting the analyzed models and diagrams into functional code. The system was developed using React.js for the front end, Flask (Python) for the backend, and Firebase as the database and authentication service. Each team member worked on separate modules, ensuring efficient parallel development through version control using GitHub.

The frontend was designed to provide a simple and modern interface, allowing users to upload an image and receive real-time detection results. The backend handles image processing and communicates with an external AI model API that analyzes whether an image is AI-generated or real.

The Firebase integration was used for user management, storing scan history for registered users, and maintaining secure login sessions. The application also includes an admin panel where the administrator can view reported images, check statistics, and generate simple analytics.

## 4.3 Finalized Design

The finalized system includes the following key interfaces:

* **Home Page:** Allows any user to upload up to three images for free without signing in.
* **Authentication Page:** Provides login and registration for users using Firebase authentication.
* **Dashboard:** Displays the detection result, allows reporting of images, and shows scan history.
* **Admin Page:** Displays reported images, total scan statistics, and basic analytics.
* **Browser Extension:** Provides quick scanning capability directly from a user’s browser.

The color theme and layout were selected to maintain a professional, consistent, and user-friendly appearance. The final UI emphasizes simplicity, clarity, and responsiveness.

## 4.4 User Interfaces

**Web App**

A screenshot of a computer

AI-generated content may be incorrect.

Figure 1 - Home Page

A screenshot of a login form

AI-generated content may be incorrect.

Figure 2 - Log In Page

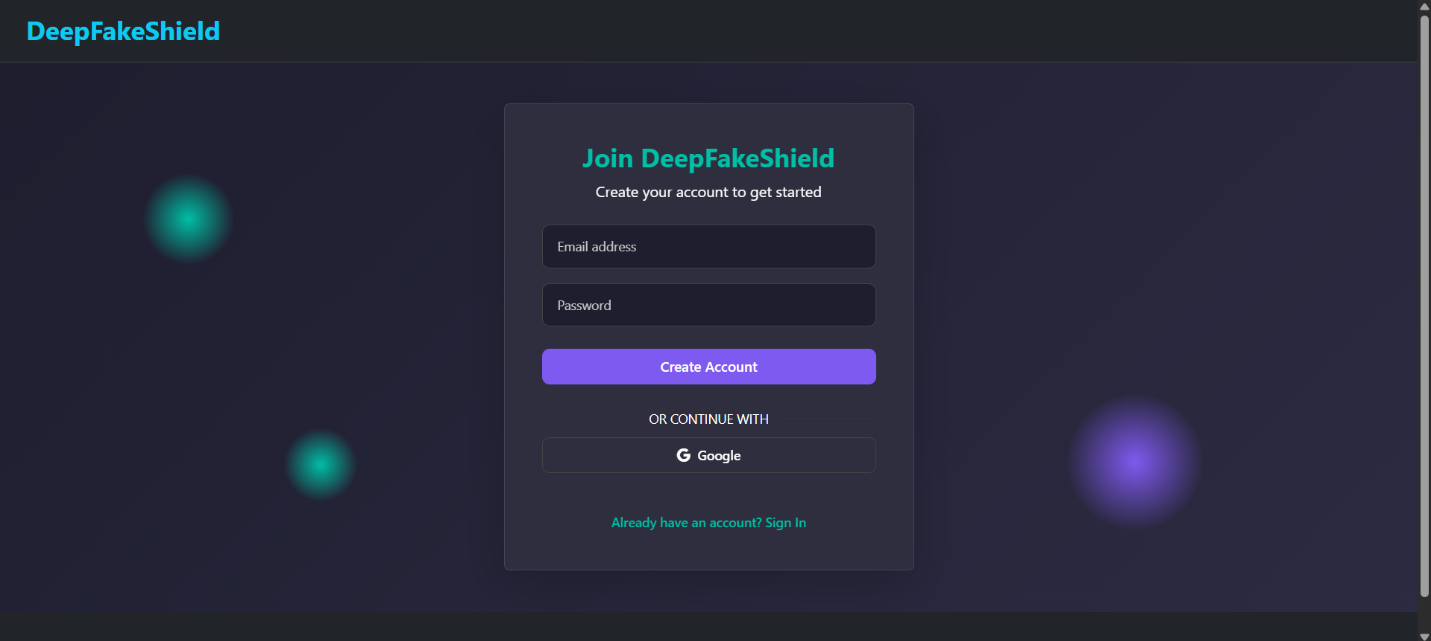


Figure 3 - Sign Up Page

A screenshot of a computer

AI-generated content may be incorrect.

Figure 4 - Dashboard

A black screen with white text

AI-generated content may be incorrect.

Figure 5 - Footer

A screenshot of a computer

AI-generated content may be incorrect.

Figure 6 – History

A screenshot of a computer

AI-generated content may be incorrect.

Figure 7 - Admin Dashboard

**Extension**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Mobile App**

**A screenshot of a sign in

AI-generated content may be incorrect.A screenshot of a login form

AI-generated content may be incorrect.**

Figure 8 - Sign In

Figure 9 - Sign Up

**A screenshot of a phone

AI-generated content may be incorrect.A screenshot of a cellphone

AI-generated content may be incorrect.**

Figure 10 - Dashboard

Figure 11 - Scan

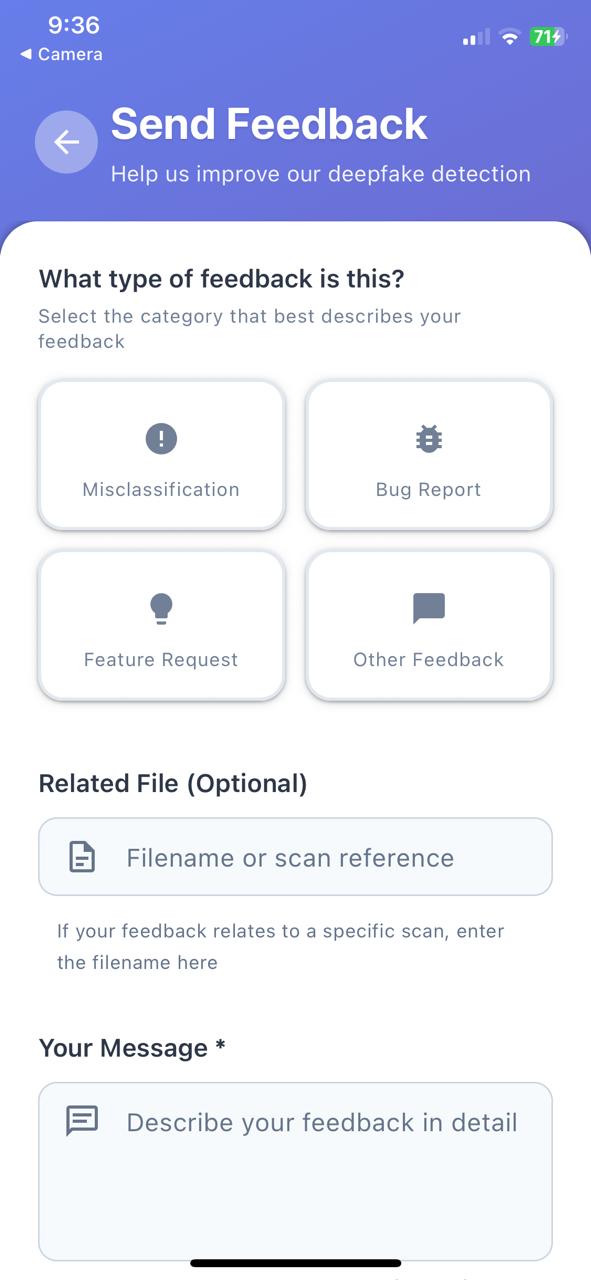
****

Figure 12 - Report/Feedback

## 4.5 Challenges Encountered

During the implementation phase, the main challenge was the accuracy of the AI model, as pre-trained models sometimes produced incorrect classifications. Additionally, hosting the backend required several adjustments because of large dependency sizes and free-tier limitations.  
These issues were managed by optimizing requests and switching between free cloud platforms for deployment and testing.

## 4.6 Chapter Summery

In summary, the implementation phase successfully transformed the proposed concept into a functional and user-interactive system. All main features were developed as planned, achieving the objectives set at the beginning of the project.

# Chapter 5: Business Model

## 5.1 Introduction

This chapter discusses the business model behind DeepFakeShield, explaining how the system can operate sustainably and provide value to users and organizations.

## 5.2 Business Model Canvas



## 5.3 Future Business Expansion

In the future, DeepFakeShield could expand by integrating premium features such as bulk image scanning, real-time detection APIs for social media platforms, and partnerships with cybersecurity organizations. Educational campaigns could also be introduced to spread awareness about the impact of AI-generated media.

## 5.4 Chapter Summery

This chapter outlined how DeepFakeShield can evolve into a sustainable and socially beneficial business model. By focusing on accuracy, transparency, and user trust, the system has the potential to grow into a valuable online safety tool used by individuals and institutions worldwide.

# Chapter 6: References

1. Croitoru, F-A., Hiji, A-I., Hondru, V., Ristea, N. C., Irofti, P., Popescu, M., Ionescu, R. T., Khan, F. S., & Shahbaz Khan, M. (2024). Deepfake Media Generation and Detection in the Generative AI Era: A Survey and Outlook.– This paper provides an up-to-date survey of both deepfake generation and detection across images, video and audio.

– Good for your report’s background section and for showing why your detector is needed.

2. Liu, P., Tao, Q., & Zhou, J. T. (2024). Evolving from Single‐modal to Multi‐modal Facial DeepfakeDetection:ASurvey.

– Focuses on how detection methods have moved from single-modality (just image) to multi-modality(audio+video+image).– Useful if you mention future expansions (e.g., detecting fake videos or audio as well).

3. Luan, T. (2024). A Survey on Deepfake Detection Technologies. (Preprint).

– A broad survey of technical methods (physical features, CNNs, transformers etc.). ResearchGate

– Good citation for the methods section in your report.

4. “Deepfakes Generation and Detection: A Short Survey”. (2023). PMC.

– Another overview of generation vs detection of deepfakes. PubMed

– Good supporting citation for the problem statement of your project.

5. Khan, S. A., Dang-Nguyen, D.-T., & others. (2024). Deepfake Detection: A Comparative Analysis.

– Compares different architectures (supervised vs self-supervised) for fake image/video detection.

– Useful if you justify your choice of model architecture versus alternatives.

6. (Optional but good for the latest tech) Exploring Self-Supervised Vision Transformers for DeepfakeDetection.(2024).

– Talks about ViT based detection with limited data, relevant since your system uses vision-model architectures