INDIRA GANDHI ENGINEERING COLLEGE

SAGAR (M.P.)

A

REPORT ON MINOR PROJECT

"SECURE VISION"

SESSION 2025

Submitted to

Rajiv Gandhi Proudyogiki Vishwavidyalaya,

Bhopal(M.P.)

In partial fulfilment of the degree

Of

Bachelor of Technology in

Information Technology

GuidedBy:
Mrs.Pragati Rajpoot
Mrs.Poonam Vinode
Department of Information Technology
I.G.E.C Sagar(M.P)

SubmittedTo:
Prof.R.S.S.Rawat
Department of Information Technology
I.G.E.C Sagar(M.P.)

SUBMITTED BY:-

MANASVI SAKALLEY 0601IT221035

PRAGATI GUPTA 0601IT221044

INDIRA GANDHI ENGINEERING COLLEGE

SAGAR (M.P.)

CERTIFICATE

This is to Certify that Pragati Gupta, Manasvi Sakalley, B.Tech.6thSemester,Information Technology Engineering have completed report on Minor.

Projectof"SECUREVISION" towards the partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Information Technology of Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P.) for the session 2025.
The work presented in this report has been carried out by them under our guidance and supervisions.

Co-GuidedBy:

Mrs.Pragati Rajpoot

Mrs. Poonam Vinode

(Faculty)

Department of Information Technology,

I.G.E.C Sagar(M.P.)

SubmittedTo:

Prof.R.S.S Rawat

(Headof Department)

Department of InformationTechnology,

I.G.E.C Sagar(M.P.)

ACKNOWLEDGEMENT

It is with great reverence that we express our gratitude to our guide "Mrs. Pragati Rajpoot" and "Mrs.Poonam Vinode" Department of Information Technology Indira Gandhi Engineering College for their precious guidance and help in this project work. The credit for the successful completion of this project goes to their keen interest timing guidance and valuable suggestion otherwise our endeavor would have been futile.

We owe to regard to "Prof. R.S.S Rawat", Head of Department of Information Technology Engineering for his persistent encouragement and blessing which were bestowed upon us.

We owe to our sincere thanks to honourable Principal "DrAnuragTrivedi",for his kind support which he rendered us in the envisagement for great success of our project.

INDIRA GANDHI ENGINEERING COLLEGE SAGAR(M.P.)

DECLARATION

We hereby declare that the following Project on "SECURE VISION" is an authentic work done by us. We undertake the project as a part of the course curriculum of Bachelor of Technology inInformation Technology of Indira Gandhi Engineering College, Sagar (M.P.) affiliated to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P.).

MANASVI SAKALLEY 0601IT221035

PRAGATI GUPTA 0601IT221044

ABSTRACT

SECURE VISION is an innovative web-based application designed to enhance student authentication and document management in educational environments. The platform leverages modern **web development** technologies—**HTML, CSS, JavaScript, React**, and **Python**—alongside key **cyber security principles** to ensure a secure and user-friendly experience.

The core functionality of Secure Vision revolves around a **biometric face** integrated with an institutional database. This mechanism verifies student identity during login or sign-up by matching live camera input with stored facial data, providing an added layer of security over traditional methods, allowing automatic extraction and validation of student information.

In addition to authentication, Secure Vision features a streamlined user interface with options like a **Free Library**, **Document Upload**, **Whiteboard**, and **Secure File Inputs**, accessible via a dynamic **hamburger menu**. Emphasis is placed on protecting data through secure transmission protocols, input validation, and authentication flows built using **Python back-end logic**.

By combining **responsive design** with **robust security practices**, Secure Vision demonstrates how modern web technologies can be harnessed to create smart, secure, and accessible platforms tailored for academic institutions.

INDEX

CONTENT	PAGENO
1.Introduction	7
2.LiteratureSurvey	8
• 2.1 Frontend	8
• 2.2 Backend	10
3. E-R Diagram and Dataflow diagram	12
4. Project Instances	15
5. Scope of future work	18
6. Result of the work	20
7. Conclusion.	21
8.Reference	23

INTRODUCTION

In an increasingly digital world, protecting personal information and documents has become more important than ever. **Secure Vision** is a web-based application developed to address this need by offering a secure platform for individuals to store and manage their personal documents. The website allows users to **log in or sign up using face recognition**, providing a more secure and user-friendly alternative to traditional authentication methods. Built using modern **web development technologies**—including **HTML, CSS, JavaScript, React**, and **Python**—Secure Vision focuses on both functionality and security. The platform incorporates **cyber security principles** to ensure that all user data and documents are stored safely and accessed only by verified individuals.

Once authenticated, users can upload, view, and manage their documents within a private and encrypted environment. The intuitive interface, enhanced by responsive design and a clean layout, includes features like a **Free Library**, **Document Upload**, **Secure File Inputs**, accessible via a compact **hamburger menu**. Secure Vision demonstrates how powerful web technologies and facial recognition can be combined to create a personalized, secure, and easy-to-use document management system for individuals concerned about their digital privacy.

Secure Vision stands as a practical demonstration of how web development, cybersecurity, and biometric technology can be combined to create an individual-focused platform that protects digital identity and personal data in a user-friendly environment.

LITERATURE SURVEY

Frontend Files

1.index.html – Landing Page

Purpose:

This is the **homepage** or **first screen** of the Secure Vision website. It introduces users to the platform gatey.

Key Features:

- **Animated Loading Screen:** A security-themed loading animation with a rotating shield icon and scanning effect, visually reinforcing the security focus of the platform.
- **Modern UI/UX Design:**
  - Gradient background (dark blue tones) for a professional cybersecurity look.
  - Responsive typography and animated tagline ("Your Trusted Document & Identity Security System").
- **Action Buttons:**
  - **Register** – directs new users to the registration page.
  - **Login** – directs returning users to the login page.
- **Security Highlights:**
  - Section displaying key features: Biometric Authentication, Document Encryption, and Identity Protection using icons and short descriptions.
- **Particle Animation:** Background with floating particles and dynamic canvas lines for visual appeal.
- **Tech Stack Used:** HTML, CSS, JavaScript, Bootstrap, Font Awesome.

2..register.html – Registration Page

Purpose:

This page allows **new users to create an account** by submitting their name and capturing a **facial image .**

Key Features:

- **Username Input:** Users type their name to create a unique identity.
- **Webcam Integration:**
  - Uses getUserMedia API to access the webcam.
  - Captures a facial image which is converted to a base64 string using <canvas>.
- **Face Capture Workflow:**
  - Users click **"Capture Face"** to store an image.
  - Submit the registration form using the **"Register"** button.
- **User Feedback:**
  - Dynamic message area displays capture success, errors, or registration status.
- **Security Design:**
  - Blurred glass effect on the container with shadow and rounded edges.
- **Tech Stack Used:** HTML, CSS, JavaScript (WebRTC for webcam), Font Awesome.

3. login.html – Login Page

Purpose:

Allows **existing users to log in** by entering their username and **recapturing their face** for verification.

Key Features:

- **Username Input:** User enters the same name used during registration.
- **Live Face Capture:**
  - Uses webcam and <canvas> to capture an image.
  - The image is sent to the backend along with the username for verification.
- **Face Matching:**
  - Backend compares captured face with stored data using face recognition (Django + Python).
- **Visual Feedback:**
  - Confirmation of successful login or error messages.
- **Redirection:**
  - On success, the user is redirected to the **dashboard**.
- **Design:**
  - Gradient background with centered login card and interactive buttons.
- **Tech Stack Used:** HTML, CSS, Bootstrap, JavaScript, WebRTC.

4. dashboard.html – User Dashboard

Purpose:

After successful login, users land on the **dashboard** where they can upload, view, and manage their

Key Features:

- **Secure File Upload:**
  - Form allows users to upload important documents.
  - Files are associated with the authenticated user and saved securely.
- **View Uploaded Files:**
  - Users see a list of their uploaded files, with links to view/download and timestamps.
- **Delete Option:**
  - Each document card has a delete button for removing documents.
- **Profile Sidebar:**
  - Hamburger menu with "Profile" and "Logout" options.
  - Clicking "Profile" opens a sliding sidebar showing the user's profile image and username.
- **Stylish UI:**
  - Glassmorphism design with blur effects and glowing highlights.
  - Responsive layout using Flexbox for document display.
- **Security Practices:**
  - CSRF protection on forms.
  - Backend handles document filtering to show only logged-in user's files.
- **Tech Stack Used:** HTML, CSS, JavaScript, Django template tags.

Backend Files

1. urls.py – URL Routing File

Purpose: Maps browser URLs to their corresponding views (functions that handle requests).

Key Features:

- Connects routes like /register/, /login/, /dashboard/, /logout/ to their respective functions in views.py.
- Uses Django's built-in static files configuration to serve uploaded media.
- Includes an admin route (/admin/) and a delete document route (/delete/<int:doc_id>/).

2. views.py – View Functions / Backend Logic

Purpose: Handles request processing, user interaction, and communication between frontend, models.

Key Functions:

- **home(request)**
  Renders the landing page (index.html).
- **register(request)**
  Handles registration:
  - Accepts username and face image from webcam (base64 format).
  - Stores the user and facial image in the database (User + UserImage).
  - Returns a JSON response to confirm registration or error.
- **login(request)**
  Handles facial recognition login:
  - Matches the captured face image with the stored one using the face_recognition library.
  - If match is successful, logs in the user using Django's auth_login.
  - Redirects to /dashboard/ on success.
- **dashboard(request)**
  Shows the user dashboard:
  - Displays all uploaded documents by the logged-in user.
  - Allows uploading new files using DocumentForm.
- **logout_user(request)**
  Logs out the current user and redirects to the homepage.
- **delete_document(request, doc_id)**
  Allows a user to delete one of their uploaded documents.

3. apps.py – Application Configuration

Purpose: Registers the Django app.

Details:

- App name is set as myapp.
- This name is used in INSTALLED_APPS in Django settings.

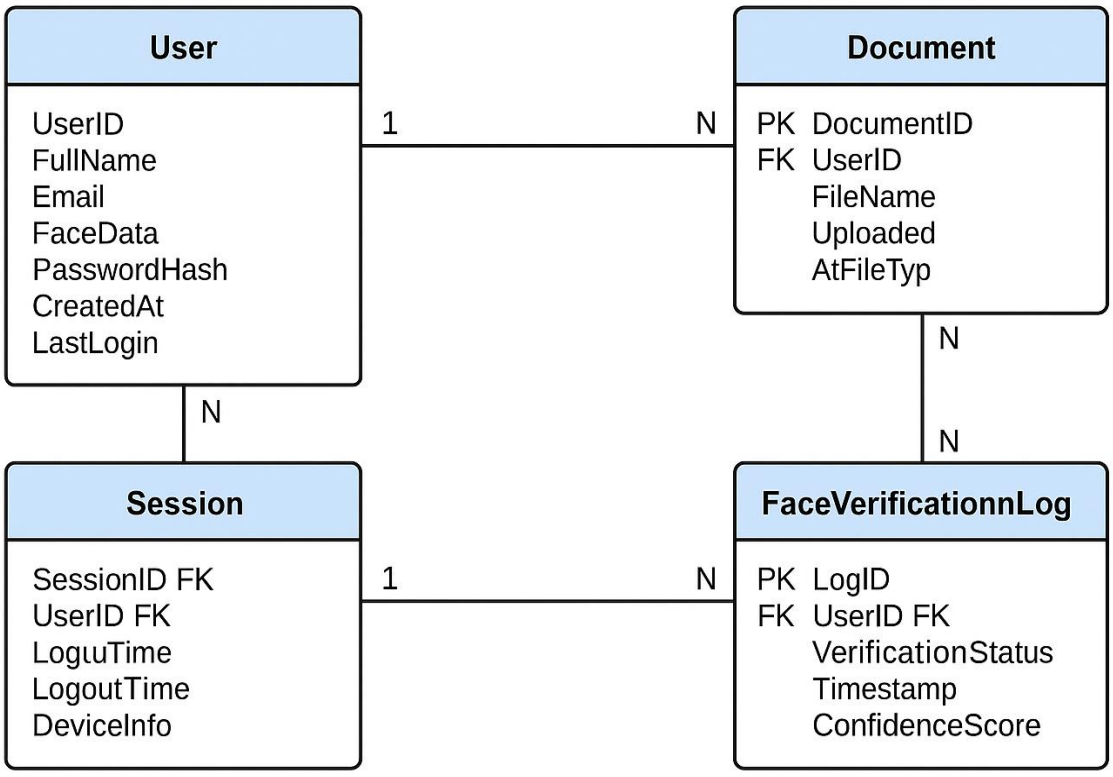
4. models.py – Database Models

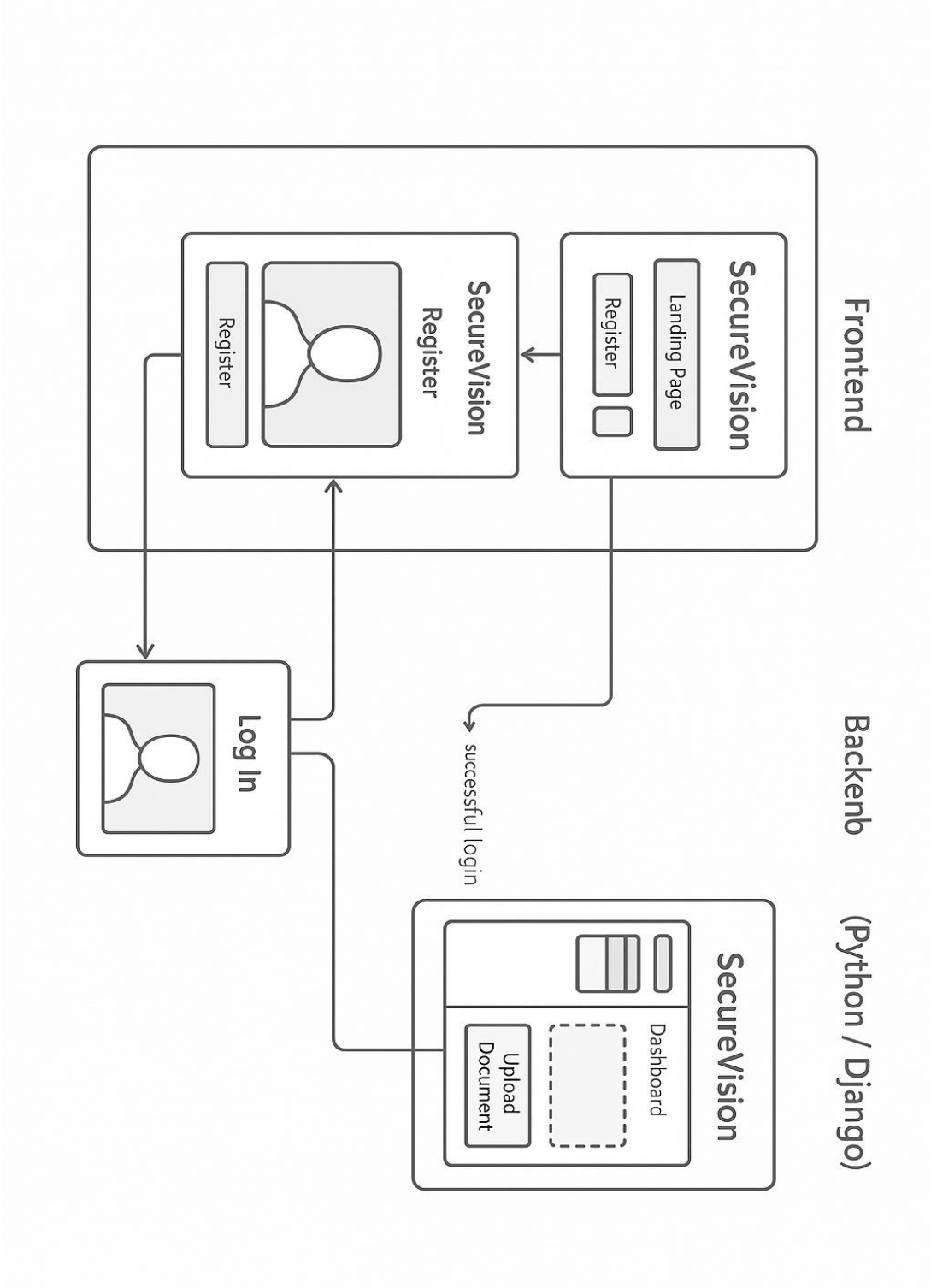
Purpose: Defines the structure of your database tables.

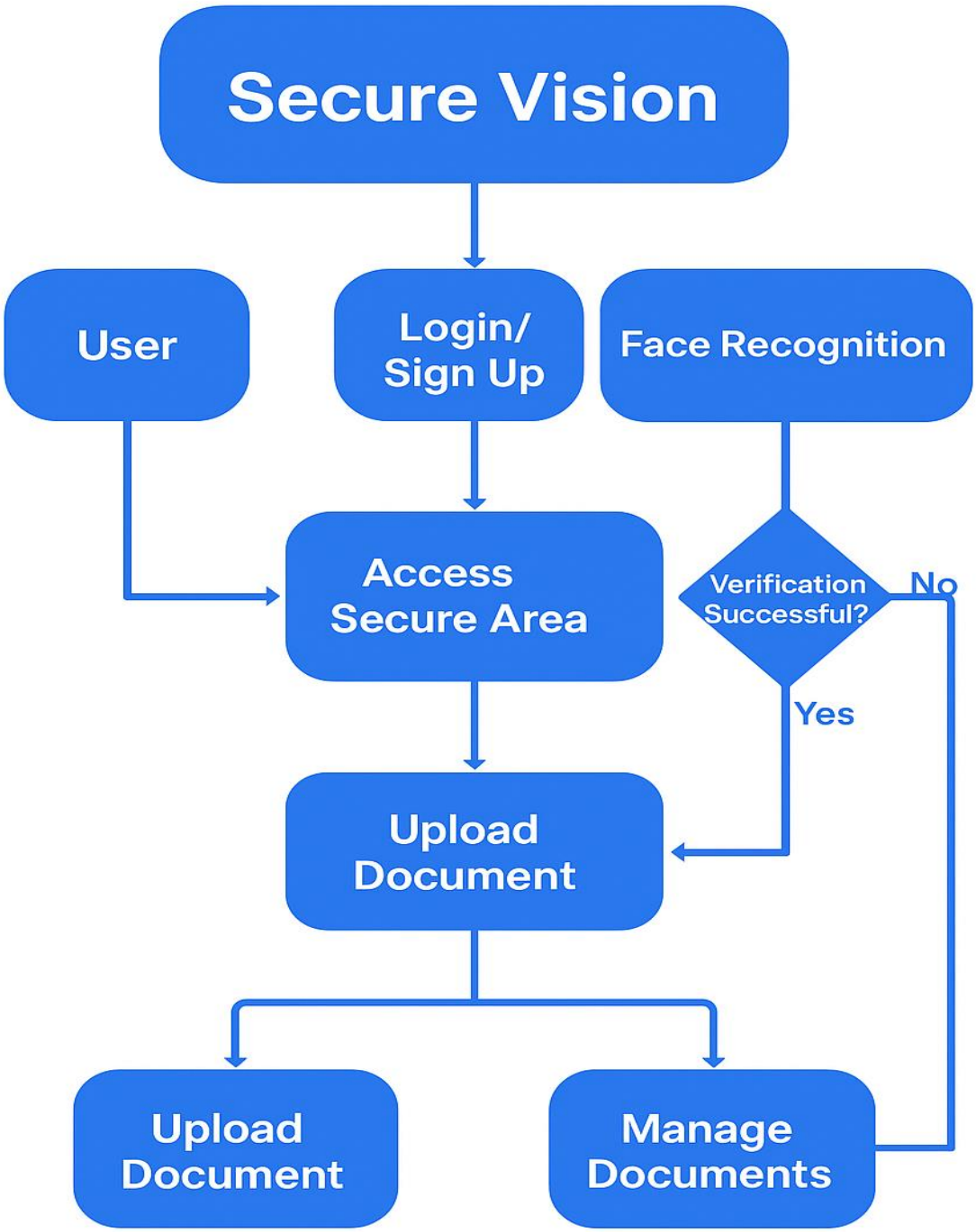
Models:

- **UserImage**
  - Stores facial image associated with a user (ForeignKey to User).
  - Image is saved under the user_faces/ directory.
- **Document**
  - Stores files uploaded by the user (ForeignKey to User).
  - Records the file and upload timestamp.
  - Stored in the documents/ directory.
- **Profile (inside Document class)**
  Likely a misplaced class – this should be moved outside Document. It defines a user profile image for potential display on the dashboard sidebar.

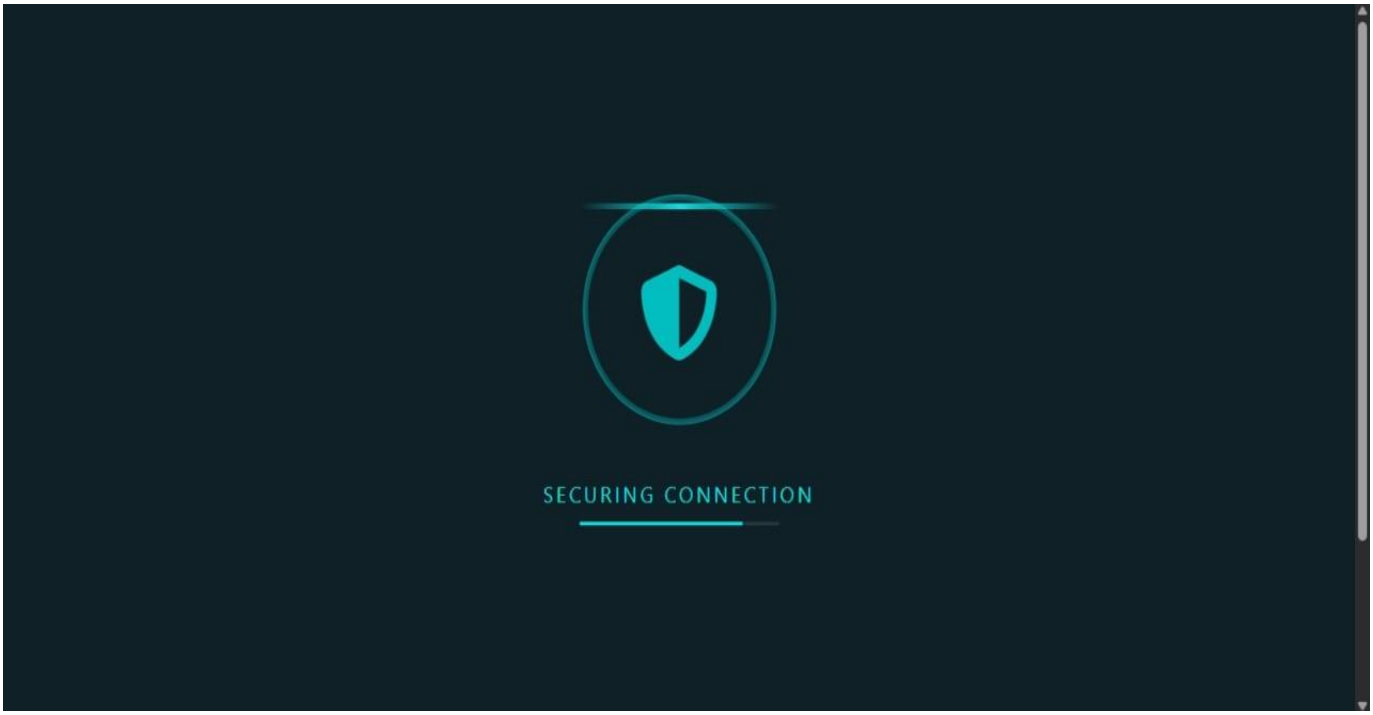
E-R DIAGRAM AND WORKFLOW DIAGRAM

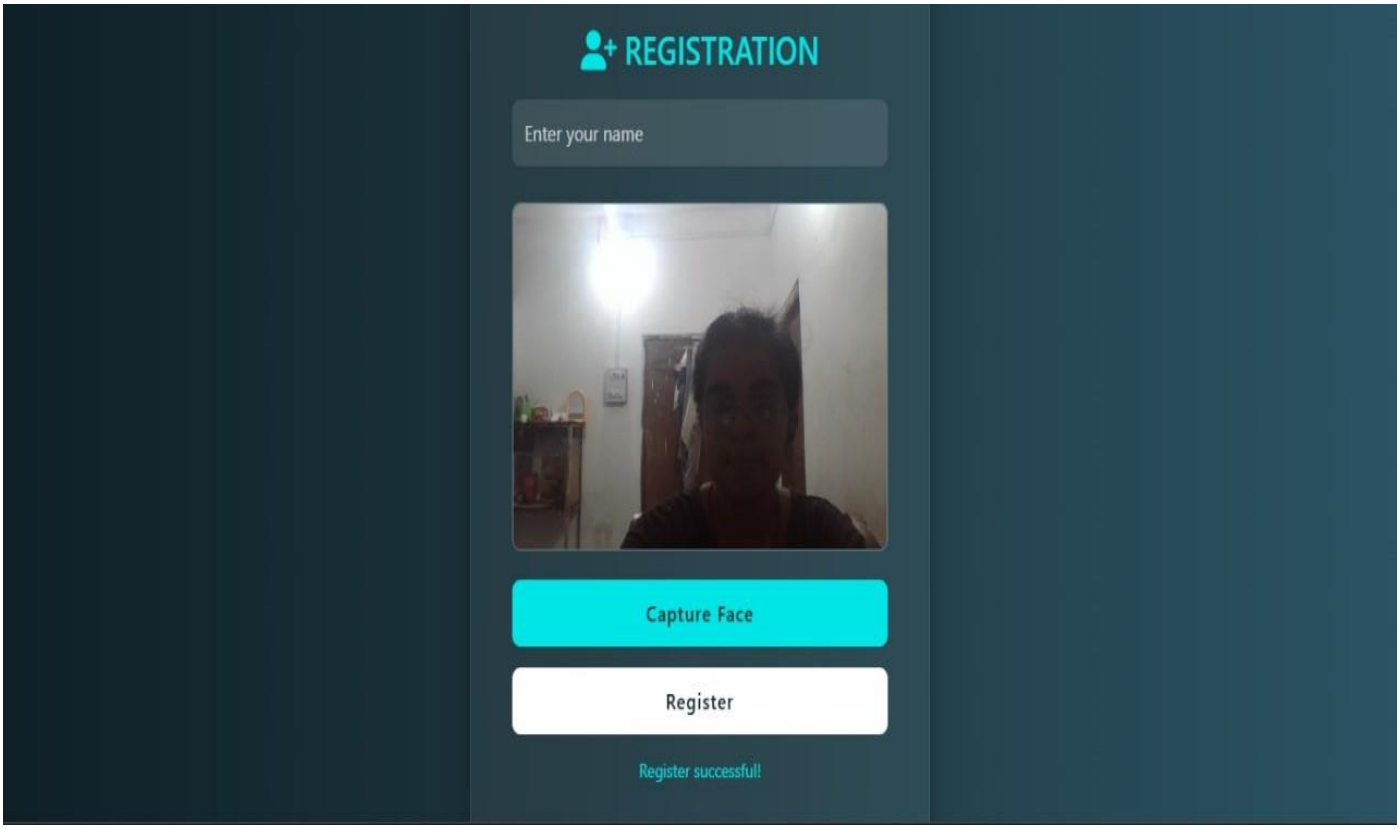


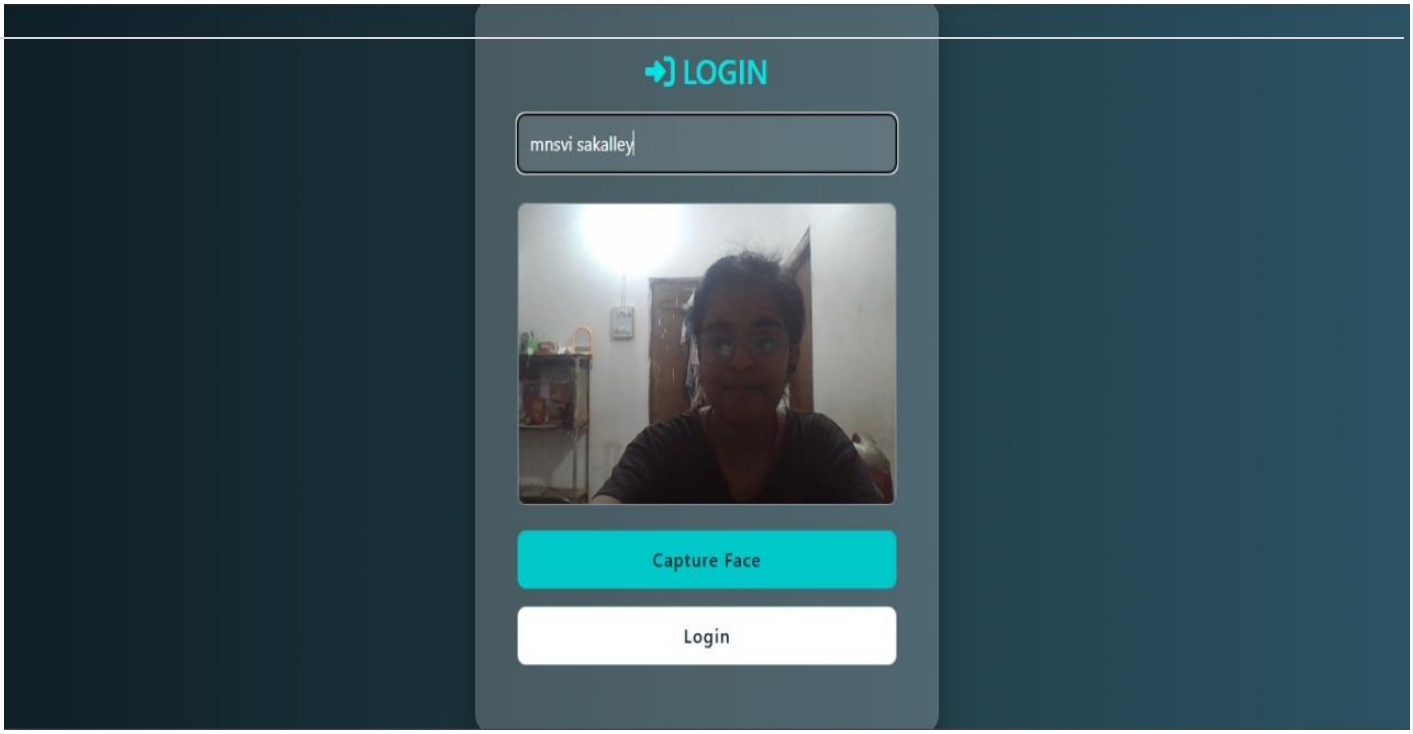

Frontend
SecureVision
Landing Page
Register
SecureVision
Register
Register
Backenb
(Python / Django)
successful login
Log In
SecureVision
Dashboard
Upload Document

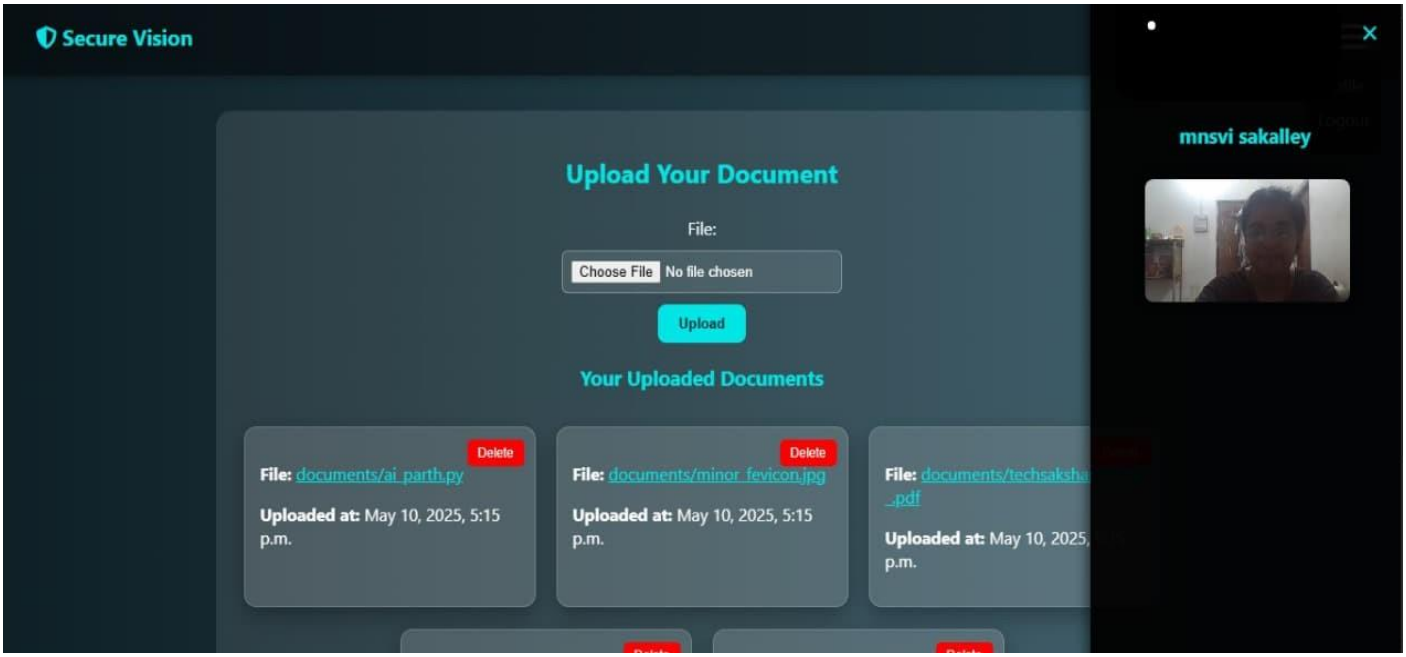

Secure Vision
User
Login/
Sign Up
Face Recognition
Access
Secure Area
Verification
Successful?
No
Yes
Upload
Document
Upload
Document
Manage
Documents

PROJECT INSTANCES




REGISTRATION
Enter your name
Capture Face
Register
Register successful!


LOGIN
mnsvi sakalley
Capture Face
Login


Secure Vision
Upload Your Document
File:
Choose File
No file chosen
Upload
Your Uploaded Documents
Delete
File: documents/ai_parth.py
Uploaded at: May 10, 2025, 5:15 p.m.
Delete
File: documents/minor_fevicon.jpg
Uploaded at: May 10, 2025, 5:15 p.m.
File: documents/techsaksha
.pdf
Uploaded at: May 10, 2025,
p.m.
mnsvi sakalley

SCOPE OF FUTURE WORK

Secure Vision currently provides a secure and personalized platform for face recognition-based login document storage, there areas where the system can be expanded and improved in the future. These enhancements will further increase usability, performance, and security for individual users:

1. Multi-Device Synchronization

- Implement real-time synchronization across desktop, tablet, and mobile devices.
- Ensure sessions are secure and uniquely tied to the user's biometric identity.
- Allow simultaneous document access and updates from multiple platforms.

2. Mobile Application Integration

- Develop native apps for Android and iOS using frameworks like React Native or Flutter.
- Utilize mobile hardware for face/fingerprint authentication via the device's secure APIs.
- Push notifications for login alerts, document uploads, or suspicious activity.

3. Advanced Biometric Features

- Add support for **voice recognition** as an alternative login method.
- Enable **fingerprint login** for devices that support biometric hardware.
- Introduce multi-factor authentication (MFA) by combining face + fingerprint/voice.

4. AI-Based Threat Detection

- Integrate AI/ML algorithms to detect abnormal behavior patterns (e.g., login from new locations or devices).
- Automatically trigger additional verification steps for suspicious sessions.
- Log anomalies and notify users in real time.

5. Encrypted Cloud Backup

- Offer optional cloud storage for documents with end-to-end encryption.
- Use services like AWS S3, Google Drive API, or custom cloud infrastructure.
- Allow users to schedule automatic backups or sync important folders.

6. User Activity Logs

- Maintain a detailed activity history including:
    - Login/logout timestamps
    - Document uploads/downloads
    - Any failed face recognition attempts
- Allow users to view and download these logs from their dashboard.

7. Integration with Digital ID Systems

- Link Secure Vision with official digital ID platforms like Aadhaar (India), DigiLocker, or government e-ID systems.
- Allow identity verification for uploading official documents like PAN card, driver's license, etc.
- Improve document authenticity for legal, banking, or educational use.

8. End-to-End Encryption and Zero-Knowledge Architecture

- Enhance security by ensuring only the user can decrypt stored files.
- Platform will not retain any decryption keys (zero-knowledge policy).
- Use robust encryption standards like AES-256 and secure key management techniques

RESULTS OF THE WORK

1. **Successfully Implemented Face-Based Registration and Login**
    - Users can register by entering a username and capturing their face via webcam.
    - Login is performed by recapturing the face and verifying it against stored data.
    - This biometric method ensures high security and eliminates the need for passwords.

2. **Developed a Clean and Interactive Frontend**
    - The UI was created using **HTML, CSS, and JavaScript** for a smooth and modern look.
    - The landing page clearly offers options to **Register** or **Login**, making it user-friendly.
    - Smooth transitions and styling improve user engagement and clarity.

3. **Integrated Webcam Access for Real-Time Face Capture**
    - Implemented webcam functionality using **WebRTC APIs and JavaScript**
    - Enables the user to capture their face directly in the browser during both registration and login.

4. **Backend Powered by Python and Django**
    - Handled core functions like user management, image matching, and session control.
    - Django was used to structure the application securely and efficiently.

5. **Secure Dashboard for Document Management**
    - After login, users are redirected to their personal dashboard.
    - They can **upload, view, and manage important files or documents**.
    - Uploaded files are securely stored and linked only to the authenticated user.

6. **Maintained Strong Data Privacy and Access Control**
    - Documents are only accessible after successful facial verification.
    - Prevents unauthorized users from accessing private files.

7. **Tested for Functional Accuracy and User Experience**
    - The full flow from registration to dashboard access was tested and confirmed to be seamless.
    - All major features performed as expected under different test scenarios.

CONCLUSION OF THE SECURE VISION

1. Secure and Modern Authentication System

- Secure Vision eliminates traditional password-based login and replaces it with **facial recognition authentication**.
- This biometric method increases the overall security of the platform by ensuring that only the legitimate user can access their account.
- Face recognition reduces the chances of brute force attacks, phishing, and credential leaks.

2. User-Friendly Registration and Login Workflow

- Users can easily **register** by entering a username and capturing a facial image using their webcam.
- For login, the same process is repeated — the system verifies the face against stored data.
- This simple yet effective flow enhances **both user convenience and access control.**

3. Interactive Frontend Design

- Developed with **HTML, CSS, and JavaScript**, the frontend ensures a smooth, responsive, and accessible user interface.
- The **landing page** offers two clear options: Register (for new users) and Login (for returning users).
- The **dashboard** is designed for easy navigation and offers essential features like document upload and viewing.

4. Reliable and Scalable Backend Architecture

- Built using **Python and Django**, the backend handles the core logic of face verification, session management, and secure data handling.
- Django provides a secure foundation, with built-in support for authentication, database operations, and server-side processing.
- This architecture makes the system robust and scalable for future enhancements.

5. Document Management System

- After successful login, users access a **personal dashboard** where they can upload important files and documents.
- Documents are securely stored and are accessible only after successful face verification.
- This turns Secure Vision into a **personal, digital locker** designed to protect sensitive data.

6. High Potential for Real-World Use

- Designed for **individual use**, Secure Vision provides a practical solution for people who want a secure digital space.
- Its real-world applications include storing personal IDs, certificates, legal documents, and more.
- The platform demonstrates how cybersecurity and biometrics can be combined to serve everyday needs.

7. Future-Ready and Scalable Design

- The modular structure makes it easy to integrate **future features** like mobile apps, cloud backups, or multi-factor authentication.
- Its focus on **privacy, user identity, and data protection** makes it a forward-thinking and relevant solution in today's digital landscape.

REFERENCE

https:/chatgpt.com

https://opencv.org

https://www.djangoproject.com

https://www.friends2support.org/

https://github.com/ageitgey/face_recognition

https://webrtc.org