Lost & Found Person

Atharv Sharma
IT department. Acropolis Institute of Technology and Research RGPV
Indore, India
atharvsharma231126@acropolis.in

Dali Vaishnav
IT department. Acropolis Institute of Technology and Research RGPV
Indore, India
dalivaishnav220436@acropolis.in

Harshita Sultanpure
IT department. Acropolis Institute of Technology and Research RGPV Indore, India
harshitasultanpure221199@acropolis.in

Khushi Agrawal
IT department. Acropolis Institute of Technology and Research RGPV
Indore, India
khushiagrawal220285@acropolis.in

Abstract —
The traditional methods for locating missing persons and recovering lost items are often slow, fragmented, and lack a unified, real-time mechanism, leading to low success rates. This project introduces the Lost & Found Person, an innovative, AI-powered digital platform designed to address these critical challenges. The system provides a centralized, secure, and accessible ecosystem for users to report incidents, while leveraging Artificial Intelligence (AI) for smart matching, specifically utilizing facial recognition for missing persons and object detection for lost items. Furthermore, the platform integrates geolocation services (Google Maps API) and push notification systems (Firebase/Twilio) to issue instant, location-based alerts to nearby users and verified authorities upon detecting a potential match, thereby significantly enhancing response speed and promoting community participation in recovery efforts. The objective is to transform the search process into a smart, community-driven, and highly efficient operation. Additionally, the system now integrates a real-time camera-based face identification module that uses the device camera to continuously scan individuals, detect faces using computer vision, match them with the missing-person database, and automatically generate location-based alerts when a positive match is found.

Keywords—
Artificial Intelligence, Facial Recognition, Object Detection, Geolocation Services, Real-Time Alerts, Community Reporting, Missing Person, Lost Item Tracker, Firebase, Real-Time Face Detection, Camera-Based Identification, Live Alerts.

I. INTRODUCTION

In today's fast-paced and densely populated society, the incidence of people going missing (such as children or the elderly with memory disorders) and the frequency of lost personal belongings are alarmingly high. Existing mechanisms like manual FIRs, public announcements, and social media posts are scattered and lack the unified, intelligent, and real-time ecosystem necessary for rapid identification and recovery.

A. Overview

The Lost & Found Person project aims to fill this critical void by offering a centralized, secure, and accessible platform. Users can register incidents by providing descriptive details, uploading photographs, and sharing last-known locations. Key features of the system include:

AI-Powered Matching: Utilizing machine learning algorithms, specifically facial recognition for missing persons and object classification for lost items, to automatically compare new data against existing entries.

- Community-Driven Support: Empowering the general public to submit sightings and contribute to recovery efforts, building a strong support system.
- Real-Time Geo-Tracking and Alerts: Integrationwiththe Google Maps API for spatial tagging and tracking, and Firebase/Twilio for instantly sending alerts to nearby users and verified authorities when a probable match is detected.
- Authority Verification: Providing an admin dashboard for law enforcement and NGOs to validate cases, update statuses, and maintain the integrity of the system.
- Scalability: The modular design allows implementation across diverse environments, including schools, transport hubs, malls, and smart city networks.

B. Purpose

The primary purpose of this system is to streamline the process of reporting and matching lost and found items. By automating critical tasks like item categorization, keyword-based matching, and location tracking, the platform enhances operational efficiency and user satisfaction. Transparency is a cornerstone of the system, with every reported item undergoing admin verification to ensure the accuracy and legitimacy of the data.

Furthermore, the system aims to foster a collaborative environment where users and authorities work together effectively to recover lost belongings. By integrating detailed item descriptions and advanced matching algorithms, the platform minimizes the chances of errors and miscommunication. The system also seeks to establish a secure, trustworthy platform where users feel confident sharing information, knowing that their data is handled responsibly and ethically.

II. LITERATURE SURVEY

The current state of locating missing persons and lost items is inefficient due to a severe lack of centralized reporting platforms and reliance on slow, manual search methods. This traditional approach suffers from the absence of smart matching capabilities , failing to utilize intelligent techniques like facial recognition or object detection, which leads to lower recovery rates. Furthermore, there is limited public participation and no real-time alert mechanism to instantly notify nearby users or authorities when a case is reported.

To solve these critical issues, this project proposes a unified digital platform. The key innovation is the Integration of AI for Smart Matching , employing facial recognition for missing individuals and object classification for lost items.

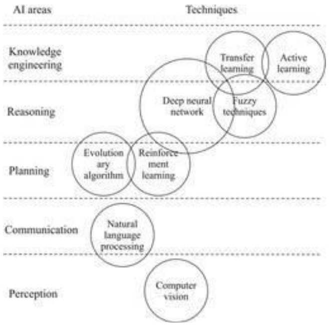

Fig: article

III. MODULES DESCRIPTION

There are several modules in this application which are:

A. *User Registration and Login*

This module is the entry point for system users, enabling them to register and securely log in. Registration requires essential details like name, email, and a strong password. For enhanced security, the system incorporates password encryption and optional two-factor authentication. Once logged in, users can report lost items, search for matches, and monitor updates on their reports. Features like "Forgot Password" and profile management ensure a smooth user experience.

B. *Lost Item/Person Reporting*

This module is for public users to report an incident. Users can submit detailed forms for both Missing Persons and Lost Items by providing:

- Descriptive details (name, age, type, color, unique features).
- Uploading photographs for AI processing.
- Sharing the last-known location. All submitted reports are queued for storage and initial processing.

C. *Data Storage and AI Processing*

This core module handles the system's intelligence and data management. Securely stores report data and multimedia (images) using databases like MongoDB and PostgreSQL. It Utilizes machine learning algorithms (TensorFlow, OpenCV) to automatically compare data. For missing persons, Facial recognition algorithms analyze uploaded photos to detect potential matches. for lost items, Object classification models are used to identify similarities between reports.

D. *Geolocation and Real-Time Alert System*

This critical module ensures rapid response by integrating location intelligence and communication services. The Integration with the Google Maps API allows for spatial tagging of reports and visual tracking of last-seen locations and When the system detects a probable match, it instantly sends alerts to nearby users and verified authorities using Firebase Cloud Messaging (push notifications) or Twilio API (SMS alerts).

E. *Admin Verification and Case Management*

This module is accessible to law enforcement, NGOs, or campus authorities. They are responsible for verification, case management, and closure. Reviewing and validating the authenticity and accuracy of user-submitted reports. Updating case statuses, managing the follow-through process, and generating performance analytics and Marking resolved incidents as closed once the person or item is successfully recovered.

F. *Public Sightings and Tips*

This module empowers the community by allowing any registered user to submit new sightings, tips, or potential found-item reports that may match an active case. This contribution helps in real-time tracking and recovery efforts.

G. *Reporting and Analytics*

This module provides admins with an overview of the system's performance. Metrics such as the number of reports submitted, successful matches, and flagged entries are displayed in dashboards. This data aids in understanding user behavior, improving the system, and identifying areas that need attention.

IV. METHODOLOGY

The development of the Lost & Found Person follows a systematic and structured approach based on the Waterfall Model of software development. This methodology ensures each phase is thoroughly completed before progressing to the next, promoting clarity, organization, and consistency in the project lifecycle.

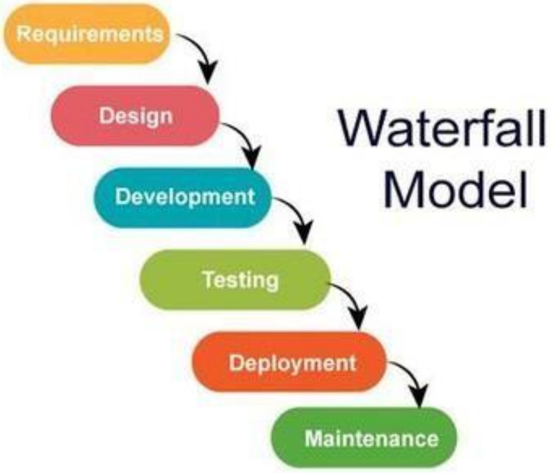

Figure 1. Waterfall Model

B. *TechnicalTools*

The development of the Lost & Found Person involved a comprehensive stack of technical tools, categorized based on their functionality. These tools ensured seamless design, development, testing, and deployment of the system.

1. Frontend Development Tools

The frontend of the system was built to ensure a user-friendly and responsive interface, allowing users to easily report and search for items.

- HTML5: Used to define the structure and layout of the web pages. It supports modern semantic elements that improve accessibility and SEO.
- CSS3: Utilized for designing the visual aspect of the website, including layouts, color schemes, and typography. CSS media queries ensured a responsive design compatible with various devices.
- JavaScript: Incorporated for dynamic interactivity, such as form validations, modal popups, and real-time updates on the webpage.
- Bootstrap: A popular CSS framework that accelerated the development of responsive, mobile-first designs by providing pre-designed components like grids, navigation bars, and buttons.

2. Backend Development Tools

The backend was developed to handle business logic, data processing, and communication with the database.

- Java Spring Boot: A lightweight Java-based

framework that simplifies the development of RESTful web services. It provided robust support for creating APIs to manage data exchange between the frontend and the database
- Apache Maven: Used for managing dependencies and automating the build process for the backend project.
- REST APIs: Developed using Spring Boot, these APIs enabled secure communication between the client- side application and the server.

3. Database Management Tools

The system required a reliable and scalable database to store user reports, found item records, and matching results.

- MySQL: A relational database management system used for storing and managing structured data. It was chosen for its scalability and support for complex queries.
- phpMyAdmin: Provided a web-based interface to manage the MySQL database efficiently during development and testing phases.
- JDBC (Java Database Connectivity): Ensured seamless interaction between the backend and the database.

4. Deployment Tools

To make the system accessible to users, deployment was carried out using cloud-based platforms.

- Heroku: A cloud platform used for deploying and hosting the web application. It simplified the deployment process with built-in tools for managing servers and resources.
- AWS (Amazon Web Services): Considered for advanced deployment scenarios, offering services like EC2 (Elastic Compute Cloud) for hosting and RDS (Relational Database Service) for managing the database.
- Docker: Used to containerize the application, ensuring consistency between development and production environments.

5. Testing Tools

Thorough testing was conducted to ensure the functionality, reliability, and security of the system.

- JUnit: A widely used framework for unit testing the backend logic and ensuring the correctness of API endpoints.
- Selenium: Employed for automated testing of the frontend, ensuring that all UI components functioned as expected across various browsers and devices.
- Postman: Used for manual testing of APIs to verify data exchange between the frontend and backend.
- JMeter: Applied for performance testing, ensuring that the system could handle concurrent user requests without degradation.

6. Design and Prototyping Tools

These tools were utilized during the planning and design phases to create visual representations of the system and its architecture.

- Figma: A collaborative tool used to design user interfaces and create clickable prototypes, allowing for early feedback and adjustments.
- Lucidchart: Assisted in creating flowcharts, system diagrams, and data flow models to represent the architecture and processes of the system.
- Balsamiq: Used for creating wireframes and low-fidelity prototypes of web pages.

7. Version Control and Collaboration Tools

Efficient collaboration and code management were critical for the success of the project.

- Git: Enabled version control, allowing developers to track changes, revert to previous versions, and manage code efficiently.
- GitHub: Served as a centralized repository for collaborative development and project management, ensuring that team members could work simultaneously without conflicts.

8. Additional Tools

To enhance the system's functionality, additional tools were integrated.

- Google Maps API: Incorporated to visualize item recovery locations and provide users with directions to authorities or designated areas.
- SendGrid: Used for sending email notifications to users about updates on their reports or potential matches.
- OpenAI GPT Models (optional): Explored for implementing AI-driven features such as intelligent suggestions for item matching based on descriptions.

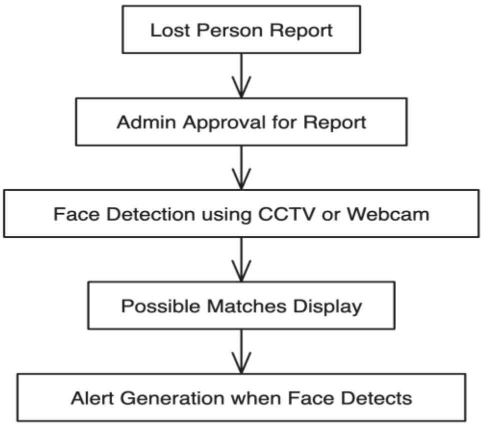
Lost Person Report
Admin Approval for Report
Face Detection using CCTV or Webcam
Possible Matches Display
Alert Generation when Face Detects

Figure 2. process of algorithm

The visual representation depicts the process of matching a lost item with a found item in the context of a Lost & Found Person. The matching algorithm relies on key attributes of items, such as their type, description, and other distinguishing features, to identify potential matches between lost and found items.

1. **Lost Person Submission**
   A user reports a *lost person* by providing essential details such as name, age, physical description, last-known location, and a reference image. The submitted report is stored in the database and remains **pending** until reviewed by the admin.

2. **Admin Approval for Report**
   The admin reviews the submitted lost-person report to ensure accuracy and authenticity. Once approved, the report becomes **active** and is added to the system's missing-person database. Only approved reports are used for AI-based matching.

3. **Face Detection Using CCTV or Webcam**
   The system supports real-time face detection using:
   - Webcam (via WebRTC in browser)
   - CCTV/IP camera streams (if integrated)

   The camera feed is processed frame-by-frame using OpenCV, and facial embeddings are generated using DeepFace/FaceNet. These embeddings are then compared with all approved lost-person profiles stored in the database.

4. **Matching Criteria**
   The system matches detected faces based on multiple parameters:
   - Facial embeddings similarity score
   - Confidence threshold
   - Feature extraction using AI (eyes, nose, chin, etc.)
   - Additional attributes such as gender/age estimation (if used)
   - If the similarity score exceeds the configured threshold, the system considers it a potential match.

5. Possible Matches Display

When a match is found, the system displays a Possible Match Result, including:

- Detected face image
- Matched lost-person profile
- Similarity/confidence score
- Time of detection
- Source of detection (Webcam/CCTV)

This helps the user or admin verify the accuracy of the match.

6. Alert Generation When Face is Detected

If a live face matches an approved lost-person record, the system automatically generates an alert that includes:

- Matched person details
- Detected image
- Detected time
- Location (if available)
- Confidence score

Alerts are sent to:

- The admin dashboard
- The reporter/user (via notification system, if enabled).

VII. FUTURE SCOPE

The *Lost & Found Person* web-based system has strong potential for further expansion and real-world deployment. As technology evolves, several enhancements can be integrated to extend the system's usability, performance, and reach.

1. **Advanced Web-Based Camera Identification**
   - Improve the real-time face recognition accuracy through browser-based WebRTC camera access.
   - Add **multi-face scanning** to identify multiple individuals at once through a webcam or laptop camera.
   - Extend functionality to allow **CCTV/IP camera streaming directly into the web system** for automated monitoring.
   - Implement a **browser-based offline detection model** using TensorFlow.js for limited offline capability.
2. **Enhanced AI Matching**
   - Upgrade AI models to more efficient architectures for facial recognition (e.g., FaceNet, ArcFace) within a web environment.
   - Use **Natural Language Processing (NLP)** for more accurate matching of item descriptions.
   - Integrate **semantic search** for lost item reports using AI embeddings.

3. **Enhanced Real-Time Web Notifications**
   - Improve alert delivery using **web push notifications**, browser pop-ups, and sound alerts.
   - Add **geofencing**, where the system notifies nearby users when a missing person is detected within a certain radius.

4. Multi-Language Support
   Expanding accessibility by integrating multi-language capabilities, allowing users from diverse linguistic backgrounds to interact with the system seamlessly.

5. Community and Institutional Collaboration
   Partnering with local governments, schools, universities, and transportation agencies to expand the adoption of the system.
   Enabling organizations to integrate the system into their operations for better lost-and-found management.

6. Scalability for Global Usage
   Scaling the system's architecture to accommodate larger databases and global operations, supporting users from multiple regions.

VIII. REFERENCES

1.**Google Maps API Documentation**

https://developers.google.com/maps/documentat on

2. **OpenCV Documentation**

https://docs.opencv.org/master/

3. **Firebase Cloud Messaging (FCM) Docs**

https://firebase.google.com/docs/cloud-messaging

4. **Twilio Messaging API**

https://www.twilio.com/docs/messaging

5. **Face Recognition Using Deep Learning: A Review** – IEEE Xplore

https://ieeexplore.ieee.org/document/9449828

6. **Kaggle – Missing Person Data Sets**

https://www.kaggle.com/datasets/iamsouravbanerjee/missing-persons-dataset

7. **TensorFlow Documentation**

https://www.tensorflow.org/

8.**Pandas Documentation (Data Preprocessing)**

https://pandas.pydata.org/docs/

9. **MongoDB Official Docs**

https://www.mongodb.com/docs/

10. **Real-Life Missing Persons Statistics – NCRB India**

https://ncrb.gov.in/en/crime-india