LOST AND FOUND MANAGEMENT SYSTEM

Subhendu Jena
Department of Computer Science and Engineering GIFT Autonomous,
Bhubaneswar, Odisha, India

Kumar Sambit Rout
Department of Computer Science and Engineering GIFT
Autonomous, Bhubaneswar, Odisha, India

Mr. Shubhendu Sekhar Sahoo
Department of Computer Science and Engineering
GIFT Autonomous, Bhubaneswar, Odisha, India

ABSTRACT

The Lost and Found Management System is a web-based application developed to provide an organized, secure, and efficient platform for managing lost and found items in educational institutions, workplaces, public places, and residential communities. Traditional lost-and-found processes often rely on manual registers, paper notices, or verbal communication, which can lead to delays, missing records, and inefficient item recovery.

The proposed system uses web technologies such as HTML, CSS, JavaScript, PHP, and MySQL to create a centralized digital platform where users can report lost items, upload found item details, search records, and manage their accounts securely. The system includes user authentication, item listing management, search and filtering mechanisms, and administrative controls for record monitoring and management.

The primary objective of this project is to simplify the recovery process by maintaining accurate records and providing quick access to information related to lost and found belongings. The application reduces manual workload, improves transparency, enhances user convenience, and increases the probability of successful item recovery. The system also provides a responsive interface that supportaccessibility across multiple devices.

Keywords: Lost and Found System, Web Application, PHP, MySQL, Authentication, Record Management, Item Tracking, Database System.

1. INTRODUCTION

finders, and the recovery process may become time-consuming.

With the growth of web technologies and digital information systems, automated solutions can significantly improve the efficiency of lost-and-found management. A web-based platform provides centralized storage, secure access, faster searching, and improved communication among users.

The Lost and Found Management System aims to develop an online platform where users can register, report lost items, upload found item details, search available records, and manage their profiles securely. The system uses PHP and MySQL for backend processing and database management, while HTML, CSS, and JavaScript provide an interactive user interface.

The project focuses on improving transparency, reducing manual effort, maintaining accurate records, and providing a user-friendly environment for efficient lost item recovery.

2. OBJECTIVES OF THE PROJECT

Lost personal belongings are common problems in schools, colleges, offices, malls, railway stations, airports, and public gathering places. Items such as mobile phones, wallets, identity cards, books, keys, bags, and important documents are frequently misplaced by users. Recovering these items can become difficult due to the absence of a proper centralized reporting and tracking mechanism.

Traditional lost and found systems mainly depend on manual methods such as paper notices, announcement boards, physical registers, or verbal communication. These approaches are often slow, disorganized, and prone to errors. Important records may be misplaced, communication gaps may occur between owners and

The major objectives of the Lost and Found Management System are:

1. To develop a web-based platform for reporting lost and found items.
2. To provide secure user registration and authentication functionality.
3. To maintain a centralized database of all reported items.
4. To allow users to post detailed information about lost and found belongings.
5. To provide search and filtering mechanisms for efficient item tracking.
6. To reduce manual workload involved in traditional lost-and-found processes.
7. To improve communication between item owners and finders.
8. To provide administrative control for managing records and users.
9. To improve transparency, accessibility, and usability of the system.

3. LITERATURE SURVEY

Several studies and web-based management systems have been developed to improve record management, user authentication, and online information retrieval.

Web-Based Management Systems

Web-based systems are widely used in educational institutions, hospitals, organizations, and businesses for maintaining records and automating manual operations. These systems improve accessibility, data storage, and communication efficiency

Database-Driven Applications

Database management systems such as MySQL are commonly used for storing and retrieving structured information efficiently. Relational databases provide secure storage, query optimization, and fast access to user records and item information.

Authentication and Security Systems

Modern web applications implement authentication mechanisms such as user registration, login validation, session management, and access control to secure user information and protect sensitive records.

Search and Filtering Systems

Many digital platforms use search and filtering techniques to improve information retrieval. Keyword-based searching, category filtering, and location-based searching help users find relevant records quickly and accurately.

Existing Challenges

Although digital management systems provide better efficiency, challenges still exist regarding data accuracy, duplicate entries, security vulnerabilities, scalability, and user-generated incorrect information. Proper validation, secure authentication, and structured database design are essential to overcome these issues.

EXISTING SYSTEM

Traditional lost-and-found systems have been used for many years in educational institutions, offices, public transport stations, shopping centers, and community organizations. Most of these systems depend on manual processes such as written complaints, notice boards, verbal reporting, or paper registers.

Although these methods are widely used, they suffer from several limitations.

One major issue is the lack of a centralized database. Lost and found records are often stored manually, making retrieval difficult and inefficient. Users may struggle to locate information quickly, and administrators may face challenges in maintaining accurate records.

Another limitation is delayed communication between owners and finders. In many situations, found items remain unclaimed because there is no proper platform connecting both parties.

Manual systems also increase administrative workload. Managing physical records, maintaining reports, and tracking item status require considerable time and effort.

Security and transparency are additional concerns. Paper-based records may be lost, modified, or damaged, affecting reliability and accountability.

These limitations highlight the need for a secure, organized, and digital solution capable of managing lost and found activities efficiently.

4. PROPOSED SYSTEM

The proposed Lost and Found Management System is a web-based application designed to provide an efficient and centralized platform for handling lost and found records.

The system includes three major modules:

1. **User Module**
2. **Lost and Found Listing Module**
3. **Admin Module**

During registration, users create accounts by providing personal details such as username, email address, contact number, and password. User authentication is performed through secure login mechanisms.

After successful login, users can report lost items or found items by entering details such as item name, category, description, date, and location information.

The search functionality enables users to locate relevant records using keywords, categories, locations, and item descriptions.

The Admin Module manages overall system operations, monitors user activity, verifies records, and controls sensitive actions such as record deletion or modification.

The proposed system improves efficiency, transparency, accessibility, and record organization while reducing the limitations associated with manual lost-and-found management methods.

5. SYSTEM REQUIREMENTS

The Lost and Found Management System requires appropriate hardware and software components to ensure smooth execution, secure data management, and efficient user interaction. The system is designed as a web-based application and can operate on standard computer systems with internet connectivity.

Hardware Requirements

Component	Specification
Processor	Intel i3 or above
RAM	Minimum 4 GB
Storage	50 GB Hard Disk or above
System Type	64-bit Computer
Internet	Required for web access and database connectivity

Software Requirements

Software	Requirement
Operating System	Windows 10/11 or Linux
Frontend Technologies	HTML, CSS, JavaScript
Backend Language	PHP
Database	MySQL
Web Server	XAMPP / WAMP Server
IDE / Editor	Visual Studio Code / Notepad++
Browser	Google Chrome, Mozilla Firefox, Microsoft Edge

The system architecture is divided into **frontend, backend, and database layers** to provide efficient communication between users and the server.

The **frontend layer** is developed using HTML, CSS, and JavaScript. HTML is used for creating the webpage structure, CSS is used for styling and responsive design, while JavaScript is used to implement interactive functionality and client-side validation.

The **backend layer** uses PHP as the server-side scripting language. PHP handles user requests, processes form submissions, manages authentication, communicates with the database, and performs business logic operations.

The **database layer** uses MySQL for secure storage and retrieval of user information, lost item records, found item records, login credentials, and administrative data. The relational database structure ensures efficient data management and query processing.

- **Software Requirements**
- **Software** • **Requirement**

libraries to perform voter authentication. During voter registration, facial images are captured through a webcam and stored securely in the database. During login or voting, the system captures a live image of the voter and compares it with previously stored facial data using machine learning algorithms. If the facial features match successfully, the voter is authenticated and allowed to proceed with voting. This process significantly reduces unauthorized access and duplicate voting attempts.

The database stores voter records, candidate details, facial recognition data, election schedules, voting status, and election results. MySQL is used as the database management system because of its reliability, scalability, and secure data handling capabilities. Each voter record is associated with a unique voter ID to maintain consistency and avoid duplication.

• **Operating System**	• **Windows 10/11 or Linux**
• **Programming Language**	• **Python**
• **IDE**	• **VS Code / PyCharm / Jupyter Notebook**
• **Database**	• **MySQL / SQLite**
• **Libraries**	• **Pandas, NumPy, Matplo**
• **Framework**	
•	

6. SYSTEM ARCHITECTURE

The System Architecture of the Lost and Found Management System represents the overall structure and interaction between users, application modules, server-side processing, and the database. The system follows a Client–Server Architecture, where users interact with the application through a web browser, while the server processes requests and communicates with the database.

The Admin Module plays a vital role in managing the overall election process. Administrators can verify voter registrations, manage candidate information, create elections, monitor voting activities, and generate election reports. The admin module also helps identify suspicious activities and maintain transparency throughout the election process.

The Candidate Module allows candidates to upload their personal details, political party information, election symbols, and campaign-related data. These details are displayed to voters during the election process. Candidates can also monitor election announcements and updates through the system interface.

The Voting Module is responsible for secure vote casting and vote management. After successful facial authentication, voters are allowed to access the voting page where they can select their preferred candidate. Once the vote is submitted, the system stores the voting information securely in the database and immediately updates the voter's status to prevent multiple voting attempts.

Overall, the system architecture of the Advanced AI Facing Voting System provides a secure, intelligent, scalable, and transparent framework for conducting digital elections using Artificial Intelligence and biometric authentication technologies.

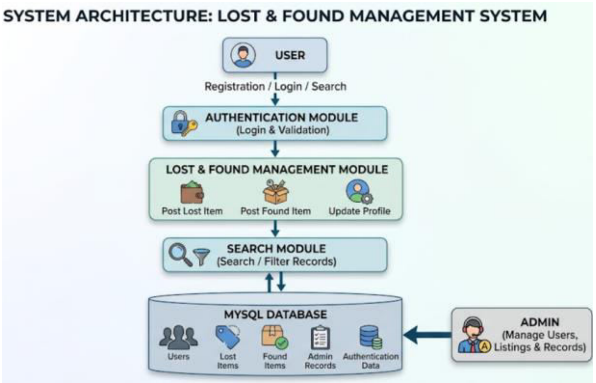

Fig 1: System Architecture Diagram

7. DATA FLOW DIAGRAM

The **Data Flow Diagram (DFD)** represents the movement of data within the **Lost and Found Management System**. It illustrates how information flows between users, application modules, processing units, and the database. The DFD helps in understanding how data is collected, processed, stored, and retrieved by the system.

The major entities involved in the system are:

1. **User**
2. **Admin**
3. **Authentication Module**
4. **Lost & Found Management Module**
5. **Search Module**
6. **Database**

The data flow process begins with **user registration**. Users provide personal information such as username, email address, phone number, and password through the registration interface. The system validates the entered details and stores them securely in the database.

After registration, users can log in using their credentials. The **Authentication Module** verifies login information against stored database records. If authentication succeeds, the user gains access to the system.

Once authenticated, users can report **lost items** or **found items** by entering details such as item name, category, description, location, and date information. The submitted records are processed and stored in the database.

The **Search Module** allows users to search available listings using keywords, categories, locations, and descriptions. Search requests are processed by the backend server, which retrieves matching results from the database and displays them to users.

The **Admin Module** interacts with all components of the system. Administrators can monitor users, manage item listings, update records, remove invalid entries, and maintain system integrity.

All user activities and record information are stored within the **MySQL database**, ensuring centralized data management and secure information handling.

The Data Flow Diagram demonstrates how data moves through different modules of the system while maintaining secure communication and centralized storage. It helps developers understand system behavior, optimize data processing, and improve operational efficiency.

The DFD also highlights the interaction between the **frontend interface, backend processing, and database layer**, ensuring accurate record handling and efficient lost-and-found management operations.

8. DATABASE DESIGN

The **Database Design** of the Lost and Found Management System plays a crucial role in storing, organizing, and managing all system-related data efficiently. A well-structured database ensures data consistency, faster retrieval, and secure handling of user and item records.

The system uses **MySQL** as the relational database management system (RDBMS). The database is designed using multiple tables that are logically connected through primary and foreign keys. This relational structure helps maintain integrity and reduces data redundancy.

8.1 USERS TABLE

This table stores all user-related information required for registration and login authentication.

Field Name	Data Type	Description
id	INT (Primary Key, Auto Increment)	Unique user ID
username	VARCHAR(50)	User login name
password	VARCHAR(255)	Encrypted password
email	VARCHAR(100)	User email address
phone	VARCHAR(15)	Contact number
created_at	TIMESTAMP	Account creation time

Purpose:
To manage user registration, login authentication, and profile information.

8.2 LOST ITEMS TABLE

This table stores information about items reported as lost by users.

Field Name	Data Type	Description
id	INT (Primary Key, Auto Increment)	Unique item ID

Field Name	Data Type	Description
user_id	INT (Foreign Key)	Reference to user
item_name	VARCHAR(100)	Name of lost item
description	TEXT	Item details
lost_location	VARCHAR(100)	Location where item was lost
lost_date	DATE	Date of loss
status	VARCHAR(20)	Pending / Found / Claimed

Purpose:
To store and track all lost item records submitted by users.

9.3 FOUND ITEMS TABLE

This table stores information about items found by users.

Field Name	Data Type	Description
id	INT (Primary Key, Auto Increment)	Unique found item ID
user_id	INT (Foreign Key)	Reference to user
item_name	VARCHAR(100)	Name of found item
description	TEXT	Item details
found_location	VARCHAR(100)	Location where item was found
found_date	DATE	Date of finding item
status	VARCHAR(20)	Available / Returned

Purpose:
To maintain records of found items for matching and recovery.

9.4 ADMIN TABLE

This table stores administrator login credentials and system control information.

Field Name	Data Type	Description
admin_id	INT (Primary Key)	Unique admin ID
username	VARCHAR(50)	Admin username
password	VARCHAR(255)	Admin password

Purpose:
To manage system administration and maintain security control.

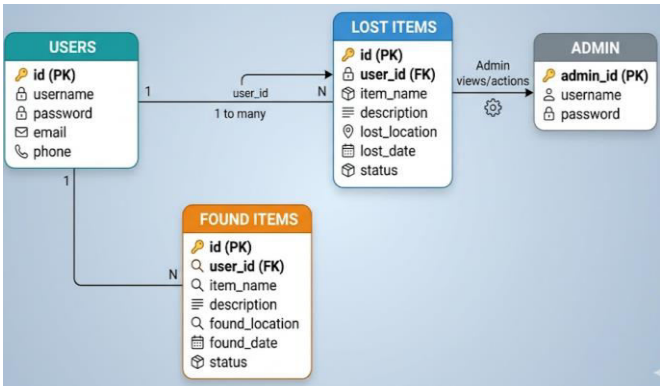

Fig 3: Database Structure

9. MODULE DESCRIPTION

The **Lost and Found Management System** is divided into multiple functional modules. Each module is designed to perform a specific task to ensure smooth operation, better organization, and efficient management of lost and found records. The modular approach also improves system scalability and maintainability.

9.1 USER REGISTRATION MODULE

The User Registration Module allows new users to create an account in the system.

Functions:

- Accepts user details such as username, email, phone number, and password
- Validates input data to ensure correctness
- Stores user information securely in the database
- Prevents duplicate registrations using unique email/username

Purpose:
To provide a secure entry point for users to access system features.

9.2 USER AUTHENTICATION MODULE

This module handles login and session management for users.

Functions:

- Verifies username and password during login
- Uses session management to maintain user login state
- Provides logout functionality
- Restricts unauthorized access to system features

Purpose:

To ensure only valid users can access the system securely.

9.3 LOST ITEM MANAGEMENT MODULE

This module allows users to report items they have lost.

Functions:

- Users can submit lost item details
- Captures item name, description, location, and date
- Stores data in the Lost Items table
- Allows users to update or view their submissions

Purpose:

To maintain a structured record of all lost items for easy tracking and recovery.

9.4 FOUND ITEM MANAGEMENT MODULE

This module manages items found by users.

Functions:

- Allows users to report found items
- Collects item details such as description and location
- Stores records in the Found Items table
- Enables matching with lost item records

Purpose:

To help return found items to their rightful owners efficiently.

9.5 SEARCH AND FILTER MODULE

This module helps users search for lost or found items.

Functions:

- Keyword-based search (item name, description)
- Filtering by category, location, and date
- Displays matching results from database
- Improves fast retrieval of records

Purpose:

To enhance accessibility and speed of item recovery.

9.6 ADMIN MANAGEMENT MODULE

This module provides full control to the administrator.

Functions:

- Manages user accounts
- Approves or removes item listings
- Monitors system activity
- Deletes false or invalid entries
- Generates system reports

Purpose:

To ensure system integrity, security, and proper administration.

9.7 DATABASE MANAGEMENT MODULE

This module handles all backend database operations.

Functions:

- Stores user, lost, and found item data
- Maintains relationships between tables
- Ensures data consistency and integrity
- Performs CRUD operations (Create, Read, Update, Delete)

Purpose:

To provide secure and organized data storage for the entire system.

10. IMPLEMENTATION

The implementation of the **Lost and Found Management System** is carried out using a combination of frontend, backend, and database technologies. The system follows a client–server architecture where the client side handles user interaction and the server side manages business logic and database operations.

The frontend is developed using **HTML, CSS, and JavaScript**, while the backend is implemented using **PHP**. The database management is handled using **MySQL**, which stores all user details and lost/found item records securely.

11.1 FRONTEND IMPLEMENTATION

The frontend of the system is responsible for providing an interactive and user-friendly interface.

Technologies Used:

- HTML (structure of web pages)
- CSS (design and styling)
- JavaScript (client-side validation and interactivity)

Key Features:

- Responsive homepage design
- User registration and login forms
- Lost item submission form
- Found item submission form
- Search and listing pages

Functionality:

- HTML creates the layout of each page
- CSS improves visual appearance and responsiveness

- JavaScript validates user inputs before submission

11.2 BACKEND IMPLEMENTATION

The backend is developed using **PHP**, which handles all server-side operations.

Responsibilities of PHP:

- Processing user requests
- Handling form submissions
- User authentication (login/register)
- Connecting frontend with MySQL database
- Managing session control

Workflow:

1. User submits data through frontend forms
2. PHP receives and processes the request
3. Data is validated and sanitized
4. PHP interacts with MySQL database
5. Results are returned to the user interface

11.3 DATABASE IMPLEMENTATION

The system uses **MySQL** for storing and managing all data.

Database Operations:

- Insert (adding new user/item records)
- Select (retrieving lost/found items)
- Update (modifying user or item details)
- Delete (removing invalid records by admin)

Tables Used:

- Users Table
- Lost Items Table
- Found Items Table
- Admin Table

Features:

- Primary keys for unique identification
- Foreign key relationships for data consistency
- Secure password storage
- Efficient query execution

11.4 AUTHENTICATION IMPLEMENTATION

Authentication ensures secure access to the system.

Steps:

- User enters username and password
- PHP validates credentials with database records
- If valid → user is logged in
- If invalid → error message is displayed

Security Measures:

- Session management using PHP sessions
- Password protection (hashed storage recommended)
- Prevention of unauthorized access

11.5 SEARCH FUNCTIONALITY IMPLEMENTATION

The search module is implemented using PHP and MySQL queries.

Features:

- Keyword-based search
- Filter by item name, location, or category
- SQL queries used for retrieving matched records

Example Logic:

- User enters search keyword
- PHP generates SQL query
- MySQL returns matching results
- Results displayed on UI

11.6 ADMIN PANEL IMPLEMENTATION

The admin panel is designed for system control and monitoring.

Functions:

- Manage user accounts
- Approve or delete lost/found posts
- Monitor system activity
- Maintain database integrity

Access Control:

- Only authorized admin can access dashboard
- Separate login credentials for admin

11. RESULTS AND DISCUSSION

The Lost and Found Management System was successfully implemented and tested to evaluate its functionality, usability, security, and overall performance. The system provides a centralized web-based platform that enables users to report lost items, register found items, search records, and communicate efficiently through a structured database-driven interface.

11.1 Results of System Implementation

After development, the system was deployed in a local server environment using PHP and MySQL. The application was tested with multiple user roles such as normal users and administrators. The following outcomes were observed:

- Users were able to successfully register and create accounts.
- Login and authentication system worked correctly with proper session handling.
- Lost item submission module allowed users to enter item details including name, description, location, and date.
- Found item module successfully stored and displayed reported items.
- Search functionality returned relevant results based on keywords and filters.
- Admin panel successfully managed user accounts and item records.
- Database operations such as insert, update, and retrieval were executed without errors.

Overall, the system performed all core operations as expected, confirming that the implementation objectives were achieved.

11.2 Functional Performance

The system was evaluated based on different functional modules. Each module performed efficiently under normal usage conditions.

Key Functional Outcomes:

- User Registration: Successfully stores user details in the database.
- Authentication: Secure login and logout with session control.
- Item Reporting: Users can post lost/found item details accurately.
- Data Retrieval: Fast and accurate display of stored records.
- Admin Control: Proper management of user-generated content.

The system showed consistent performance without data mismatch or major runtime errors.

11.3 Usability Results

The interface of the system was designed to be simple and user-friendly. During testing, users with basic computer knowledge were able to operate the system without difficulty.

Usability Observations:

- Navigation between pages is smooth and intuitive.
- Forms are simple and require minimal user input effort.
- Search functionality is easy to understand.
- Layout is clean and responsive for different screen sizes.

The feedback indicates that the system is suitable for real-world users in educational institutions and public environments.

12. CONCLUSION

The Lost and Found Management System was successfully designed and implemented as a web-based application to replace traditional manual methods of handling lost and found records. The system provides an efficient, centralized, and user-friendly platform where users can report lost items, register found items, and search for possible matches with ease.

The project effectively demonstrates the use of web technologies such as HTML, CSS, JavaScript, PHP, and MySQL for building a complete client–server application. It also highlights the importance of database management and secure user authentication in real-world applications.

From the implementation and testing results, it can be concluded that the system achieves its primary objectives, including:

- Reducing manual effort in lost and found management
- Improving the speed of item recovery
- Providing a centralized database for better record keeping
- Enhancing communication between users
- Ensuring basic security and controlled access

13. REFERENCE

[1] Avraham Leff, James T. Rayfield, “Web Application Development using the Model/View/Controller Design Pattern,” IBM T.J. Watson Research Center.

[2] G. Coulouris, J. Dollimore, T. Kindberg, *Distributed Systems: Concepts and Design*, Addison-Wesley, 2001.

[3] T. Converse, J. Park, C. Morgan, *PHP5 and MySQL Bible*, Wiley Publishing, 2004.

[4] PHP Documentation, https://www.php.net/manual/en/intro-whatis.php

[5] MySQL Documentation, https://dev.mysql.com/doc/

[6] MDN Web Docs – HTML, CSS, JavaScript, https://developer.mozilla.org/

[7] A. S. Tanenbaum, M. Van Steen, *Distributed Systems: Principles and Paradigms*.

[8] Research papers on web-based Lost and Found Management Systems and database-driven applications.

[9] W3Schools Web Development Tutorials, https://www.w3schools.com/