

**A
Project Report
on
Project Title
Developed By
*Krishna Kumar [2405112120086]
Koushiki [2405112120085]
Makwana Riya [2405112120096]*
as
Partial Fulfillment of Semester III of
Master of Computer Applications /
Master of Science in information Technology
for A.Y. 2025 - 2026
Under the Guidance of
Prof. Faruk Abdulla
Submitted To
Department of MCA/MScIT
Faculty of IT & Computer Science
PARUL University**





CERTIFICATE

This is to certify that **Mr. Krishna Kumar [2405112120086]** , **Ms. Koushiki, [2405112120085]** and **Ms. Makwana Riya, [2405112120096]** students of Master of Computer Applications and/or Master of Science in Information Technology has satisfactorily completed the Minor Project on **“Project Title”** at **Faculty of IT & Computer Science, Parul University** as partial fulfillment of MCA and/or M.Sc. (IT) Semester III.

Seat No. _____ Date _____ of _____ Submission:

Internal Guide

Head of Department

Dean - FITCS

Faculty of IT & Computer Science
PARUL University
Vadodara

1. About Department of MCA & M.Sc. IT

PARUL University

Parul University is a legitimate university established under Gujarat Private University Act 2009, after legislation passed by the Government of Gujarat on 26th March 2015 giving University status to Parul Group of Institutes functioning under the aegis of Parul Arogya Seva Mandal Trust.

Faculty of IT & Computer Science

Faculty of IT and Computer Science, Parul University has materialized as one of the prime IT education providers at global level. Various departments under Faculty of IT and Computer Science strive in preparing IT-industry ready professionals by means of various skill development courses, vocational courses, co-curricular & extra-curricular activities, industry visits and expert lectures.

MCA Department

The Department of Master of Computer Application and Master of Science in Information Technology at Parul University emphasizes on building professionals in the domain of computer applications by providing necessary environment by means of facilitating suitable blend of technical and non-technical learning experience. The department cultivates students in various curricular, co-curricular and extra-curricular activities in order to produce future system analysts, system designers, system programmers, application programmers, testing professionals, system managers, project managers, researchers and other leading positions in systems/IT department.

The departments offers various subjects from diversified technical/non-technical areas such as – core IT domain, management, communication skills, mathematics & logic building and rich pool of elective subjects.

The department of MCA and M.Sc. (IT) focuses on project-based learning, and hence students are motivated to work on tiny hands-on projects in practical oriented subjects to get better exposure. Moreover, throughout their MCA studies, students are required to work on around 3 mini/major projects in individual/team to get enough confidence on software-development and thereby become industry-ready.

INDEX

No.	Description	Page No.
1.	About Department of MCA	1
2.	Project Profile <ul style="list-style-type: none"> 2.1 Project Definition 2.2 Project Description 2.3 Existing System / Work Environment 2.4 Problem Statements 2.5 Need for New System 2.6 Proposed System & Features 2.7 Scope 2.8 Outcomes 2.9 Tools & Technology used 2.10 Project Plan 	
3.	Requirement Analysis <ul style="list-style-type: none"> 3.1 Feasibility Study 3.2 Users of the System 3.3 Modules 3.4 Process Model 3.5 Hardware & Software Requirements 3.6 Use Cases 3.7 Use Case Diagram 	
4.	Design <ul style="list-style-type: none"> 4.1 Use Case Scenarios 4.2 Diagrams <ul style="list-style-type: none"> 4.2.1 UML / DFD 4.2.2 Entity Relationship Diagram 4.3 Data Dictionary 	
5.	Implementation <ul style="list-style-type: none"> 5.1 Form Layouts 5.2 Report Layouts 5.3 Coding Convention (Business Logic) 	
6.	Testing <ul style="list-style-type: none"> 6.1 Test Strategy 6.2 Test Cases 	
7.	Future Enhancement	
8.	Bibliography	

2. Project Profile

2.1 Project Definition

The Lost & Found Management System is a secure, web-based application designed to record, track, and facilitate the return of lost and found items through transparent claims and verification procedures.

2.2 Project Description

The application allows users to report items they have lost or found, including photos, category, location, and detailed descriptions. It features a matching algorithm that identifies potential matches between lost and found reports based on key attributes. Authorized personnel verify claims, maintain custody records, and oversee the process until resolution.

2.3 Existing System / Work Environment

Current manual methods such as noticeboards, security logbooks, and informal messaging are inefficient, prone to errors, and lack transparency.

2.4 Problem Statements

- Ensuring accurate matches between lost and found reports
- Verifying ownership claims securely
- Maintaining a chain of custody for found items
- Timely notification of matches and updates to stakeholders

2.5 Need for New System

A centralized platform with automated matching, verification, and real-time notifications increases efficiency, transparency, and recovery rates.

2.6 Proposed Features

- Lost/Found submission forms with photos
- Automated match suggestions
- Evidence-based claims with file uploads
- Role-based access control (user, verifier, admin)
- Search, filter, and report generation
- Email/SMS notifications

2.7 Scope

In Scope: Web application, MySQL storage, multi-department support, authentication. **Out of Scope:** Mobile apps, AI-based image recognition in Phase-1.

2.8 Outcomes

- Higher recovery rates
- Reduced time to reunite items
- Improved transparency and trust in the process

2.9 Tools & Technology

- **Backend:** Django (Python)
- **Frontend:** HTML5, CSS3, Bootstrap, JavaScript
- **Database:** MySQL
- **Deployment:** Nginx, Ubuntu Server
- **Security:** HTTPS, CSRF protection, password hashing

2.10 Project Plan

A 10-week schedule from requirements gathering to deployment, including design, development, testing, and documentation.

3. Requirement Analysis

3.1 Feasibility Study

- **Technical:** All components are supported by mature, open-source tools.
- **Economic:** Low cost due to free tools; minimal server expenses.
- **Operational:** Easy to use for all roles.
- **Schedule:** Fits within academic term timelines.

3.2 Users

- Public users (students/staff)
- Verifiers (authorized department staff)
- Administrators

3.3 Modules

1. Authentication & Profiles
2. Lost Item Management
3. Found Item Management & Custody
4. Matching Engine
5. Claims & Verification
6. Notifications
7. Search & Reports

8. Admin Management
9. Audit Logging

3.4 Process Model

Agile methodology with iterative development and frequent user feedback.

3.5 Hardware & Software Requirements

- Server: 2 vCPU, 4GB RAM, Ubuntu 22.04
- Developer: 8GB RAM, Python 3.12+, MySQL 8.x

3.6 Use Cases

- UC-01 Register/Login
 - UC-02 Report Lost Item
 - UC-03 Report Found Item
 - UC-04 View/Filter Items
 - UC-05 Auto-Match Suggestions
 - UC-06 Raise Claim
 - UC-07 Verify Claim
 - UC-08 Custody Handover
 - UC-09 Notifications
 - UC-10 Generate Reports
-

4. Design

4.1 Use Case Scenarios

Scenario A: Lost Phone – User reports, system suggests matches, claim raised, verified, and item returned. **Scenario B: Found Backpack** – Staff submits report, match found, owner claims, verified, item released.

4.2 ERD Entities

User, Profile, Item, ItemPhoto, Claim, Verification, CustodyLog, Category, Location, Department, Notification, AuditLog.

5. Implementation

5.1 Key Screens

- Login/Signup Page
- Report Lost/Found Item Form
- Item Details Page
- Raise Claim Form
- Verification Console for staff
- Admin Console for master data management

5.2 Reports

- Open Items by Category
- Ageing Report
- Recovery Rate Statistics
- Claims Funnel

5.3 Coding Conventions

- Python code adheres to PEP8
 - Bootstrap-based responsive UI
 - Secure password storage using PBKDF2/Argon2
 - CSRF and input validation
-

6. Testing

6.1 Strategy

- Unit testing for models and services
- Integration testing for end-to-end workflows
- Security testing for vulnerabilities
- UAT with representative users

6.2 Sample Test Cases

1. TC-01: User Registration – Valid details create an account
 2. TC-02: Report Lost Item – Valid submission stored in database
 3. TC-03: Report Found Item – Creates custody log entry
 4. TC-04: Match Suggestion – Displays potential matches
 5. TC-05: Claim Submission – Status set to Pending
-

7. Future Enhancement

- AI-assisted photo matching
- QR code-based custody management
- Native mobile apps
- Geo-mapping of loss hotspots
- Multi-institution deployment

8. Bibliography

1. Django Software Foundation. *Django Documentation*. Available at: <https://docs.djangoproject.com/>
2. Oracle Corporation. *MySQL Reference Manual*. Available at: <https://dev.mysql.com/doc/>
3. OWASP Foundation. *OWASP Top 10 Application Security Risks*. Available at: <https://owasp.org/www-project-top-ten/>
4. Bootstrap Team. *Bootstrap 5 Documentation*. Available at: <https://getbootstrap.com/>
5. Fowler, M. *Patterns of Enterprise Application Architecture*. Addison-Wesley.
6. Martin, R.C. *Clean Code: A Handbook of Agile Software Craftsmanship*. Prentice Hall.