PES UNIVERSITY, BANGALORE

Department of Computer Science and Engineering

Software Requirements Specification (SRS) for University Lost and Found Management System

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a comprehensive overview of the University-Level Lost and Found Management System. This document outlines the objectives, features, functionalities, and constraints of the system. It serves as a guide for the development team, stakeholders, and any parties involved in the project.

1.2 Intended Audience and Reading Suggestions

This document is intended for the following audiences:

- Development Team: Developers, designers, and testers who will be involved in the design, development, and testing of the University-Level Lost and Found Management System.

- Stakeholders: University administrators, department heads, and other stakeholders with an interest in the successful implementation and operation of the system.

- Quality Assurance Teams: Those responsible for ensuring that the system conforms to the specified requirements and quality standards.

- Users: Students, faculty, staff, and other members of the university community who will interact with the system.

- Administrators: University personnel responsible for managing the system, handling reported items, and resolving matches.

Reading Suggestions:

- Developers should focus on the technical details and functional specifications of the system.

- Stakeholders should review the document to gain a clear understanding of the system's scope, objectives, and features.

- Users should refer to sections that describe user roles, functionalities, and guidelines for interacting with the system.

- Administrators should pay particular attention to sections that pertain to system administration, item management, and resolution processes.

1.3 Product Scope

The University-Level Lost and Found Management System is a specialized web-based application designed to simplify and enhance the process of reporting lost items, finding lost items, and facilitating communication between the university community members. The system allows users, including students, faculty, and staff, to report lost or found items, search for items, and receive notifications about potential matches within the university campus.

1.4 References

There are no specific external references for this university-level lost and found management system. However, the following internal references and documents may be relevant during the development and operation of the system:

- University policies and regulations related to lost and found items.

- Campus security guidelines and procedures.

- Any internal standards or guidelines for data privacy and information security within the university.

These internal references provide essential context and guidelines for developing and operating the University-Level Lost and Found Management System within the university's specific environment and policies.

2. Overall Description

2.1 Product Perspective

The Lost and Found Management System is a standalone software application designed to streamline the process of reporting and retrieving lost and found items. It operates independently and does not rely on external systems. However, it may integrate with external

services for user authentication (e.g., OAuth with social media platforms) and email notifications.

2.2 Product Functions

The primary functions of the Lost and Found Management System include:

- User registration and authentication.

- Reporting lost items with detailed information.

- Reporting found items with descriptions and locations.

- Searching for lost and found items based on various criteria.

- Matching lost and found items and facilitating contact between owners and finders.

- Admin panel for user and item management.

2.3 User Classes and Characteristics

The system supports the following user classes:

1. Registered Users: Individuals who have created an account and can report, search for, and manage lost and found items.

2. Guest Users: Individuals who can perform basic searches for lost and found items without registering.

3. Admins: Administrators responsible for overseeing the system, moderating user accounts, and managing reported items.

2.4 User Characteristics:

- Registered users may vary in age, technical expertise, and familiarity with the system.

- Guest users are typically looking to search for specific lost or found items.

- Admins have privileged access to the system's admin panel for moderation and resolution of items.

2.5 Operating Environment

The Lost and Found Management System is designed to operate in the following environments:

- Web-based: The system is accessible via standard web browsers on desktop and mobile devices.

- Cross-platform: It is compatible with various operating systems, including Windows, macOS, and mobile OS (iOS, Android).

2.6 Design and Implementation Constraints

The system design and implementation are subject to the following constraints:

-Technological Stack: The system will be developed using specific programming languages, frameworks, and libraries, as determined during the development phase.

- Database: The choice of a relational database system will influence data storage and retrieval.

- Data Security: Compliance with data protection laws and standards is mandatory, leading to specific security constraints.

- Scalability: The system should be designed with scalability in mind to accommodate future growth in user and item databases.

2.7 Assumptions and Dependencies

The development and operation of the Lost and Found Management System are based on several assumptions and dependencies:

- User Authentication: The system assumes that users will provide accurate and verifiable information during registration.

- Internet Connectivity: Users and admins are assumed to have a stable internet connection to access the system.

- External Services: If integrated, the system depends on external services (e.g., email servers, third-party authentication providers) to function correctly.

- Legal Compliance: The system assumes that users and administrators will adhere to legal requirements related to lost and found items and data privacy.

- Maintenance: Ongoing system maintenance, bug fixes, and updates are assumed to be carried out regularly to ensure its optimal operation.

These assumptions and dependencies are critical for the successful deployment and use of the Lost and Found Management System.

3. External Interface Requirements:

We are planning to develop this project using technologies that allow you to create a dynamic and interactive web application.

- **HTML and CSS are used to create the structure and style of the web pages.**
- **PHP and MySQL are used to create and manage the database, and to generate the dynamic content of the web pages.**
- **JavaScript and jQuery are used to add interactivity to the web pages, such as allowing users to click on buttons and submit forms.**
- **Ajax requests are used to communicate with the server without reloading the web page, which makes the web application more responsive.**
- **The Bootstrap framework and icons are used to create a consistent and modern look and feel for the web application.**
- **The NiceAdmin template is used to provide a starting point for the design of the web application.**

3.1. User Interface:

In this project we will develop a website. This website will have the following features which will make it user-friendly.
The homepage of our website will have clear and concise labels for the links to the lost and found page, about the website page, contact us page, and post and item page. The labels will

be easy to understand and will accurately reflect the content of the linked pages, we will also provide a button labeled login which on clicking will lead to the login page which different users can use to login through the website.All of these labels will be present at the header of the website.

The post an item page on your website should have a form that allows users to enter the details of the lost object and upload images. The form will consist of the following sections:

Category: This section should allow users to mention the category of the lost object.
Founder name: This section should allow users to enter their name.

- **Title: This section should allow users to enter a title for the lost item. The title should be a brief description of the item, such as "Lost iPhone" or "Lost black wallet".**
- **Found contact: This section should allow users to enter their contact information, such as their phone number.**
- **Description: This section should allow users to enter a detailed description of the lost item.**

We will include appropriate validation to check the contents filled in the form.

Client-side validation: Client-side validation is performed by the web browser before the form is submitted to the server. This type of validation can be used to check for basic things like required fields and valid email addresses.

Server-side validation: Server-side validation is performed by the server after the form is submitted.We use a scripting language like PHP or Python to check the form data after it is submitted. For example, you could use PHP to check that the username exists in the database or not.

The lost and found items page will be used to display the contents filled in the form.This page will be used by the user to view the lost and found items.

3.2. Software Interface:

The website would need to be connected to a database to store the data for the lost and found items. Here we will make use of the MySql database.

Application programming interface protocols

The website would use the following application programming interface (API) protocols:

- **MySQL API: The website would use the MySQL API to communicate with the MySQL database.**
- **PHP Mailer API: The website would use the PHP Mailer API to send email notifications.**

The website would need to be connected to the following software components:

- **XAMPP: XAMPP is a software stack that includes a web server, a database server, and a programming language interpreter. The website would use the Apache web server to serve its files to users, the MySQL database server to store its data, and the PHP programming language interpreter to process its code.**
- **HTML: HTML is a markup language that is used to create the structure and content of the website's pages.**
- **PHP: PHP is a programming language that is used to develop the website's functionality.**
- **MySQL Database: MySQL Database is a relational database management system (RDBMS) that is used to store the website's data.**
- **jQuery: jQuery is a JavaScript library that simplifies the process of adding interactivity to web pages.**
- **Ajax Requests: Ajax requests are used to communicate with the server without reloading the web page. This allows the website to be more responsive and user-friendly.**

Data items or messages coming into the system:

- **User registration information (name, email address, password)**
- **Lost item information (category, title, description, images)**
- **Found item information (category, title, description, contact information)**

3.3. Communication Interface:

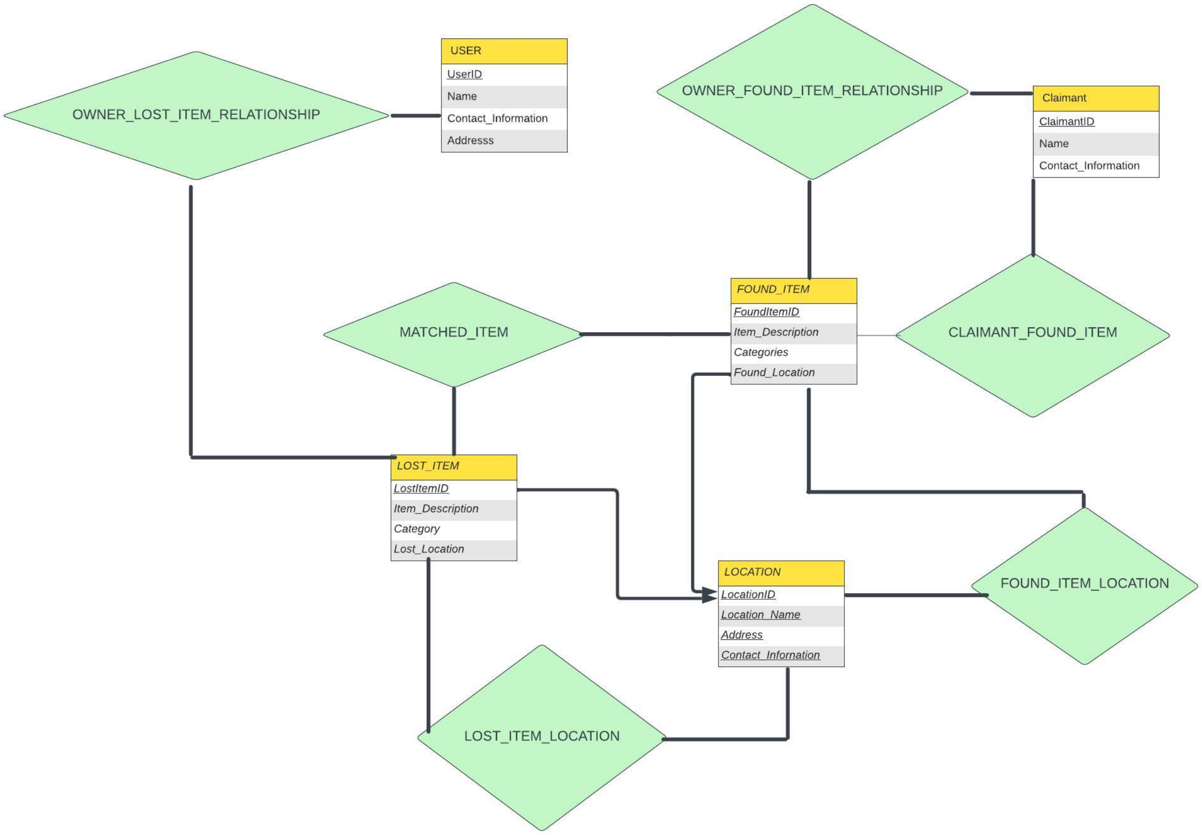
The contact information page on your website can be used to contact the admin user in a number of ways:

- **Email: The user can send an email to the email address provided on the contact page.**
- **Telephone: The user can call the telephone number provided on the contact page.**
- **Mobile phone: The user can call the mobile phone number provided on the contact page.**
- **Walk-in: The user can visit the office address provided on the contact page.**
- **Contact form: The user can fill out the contact form on the contact page and submit it.**

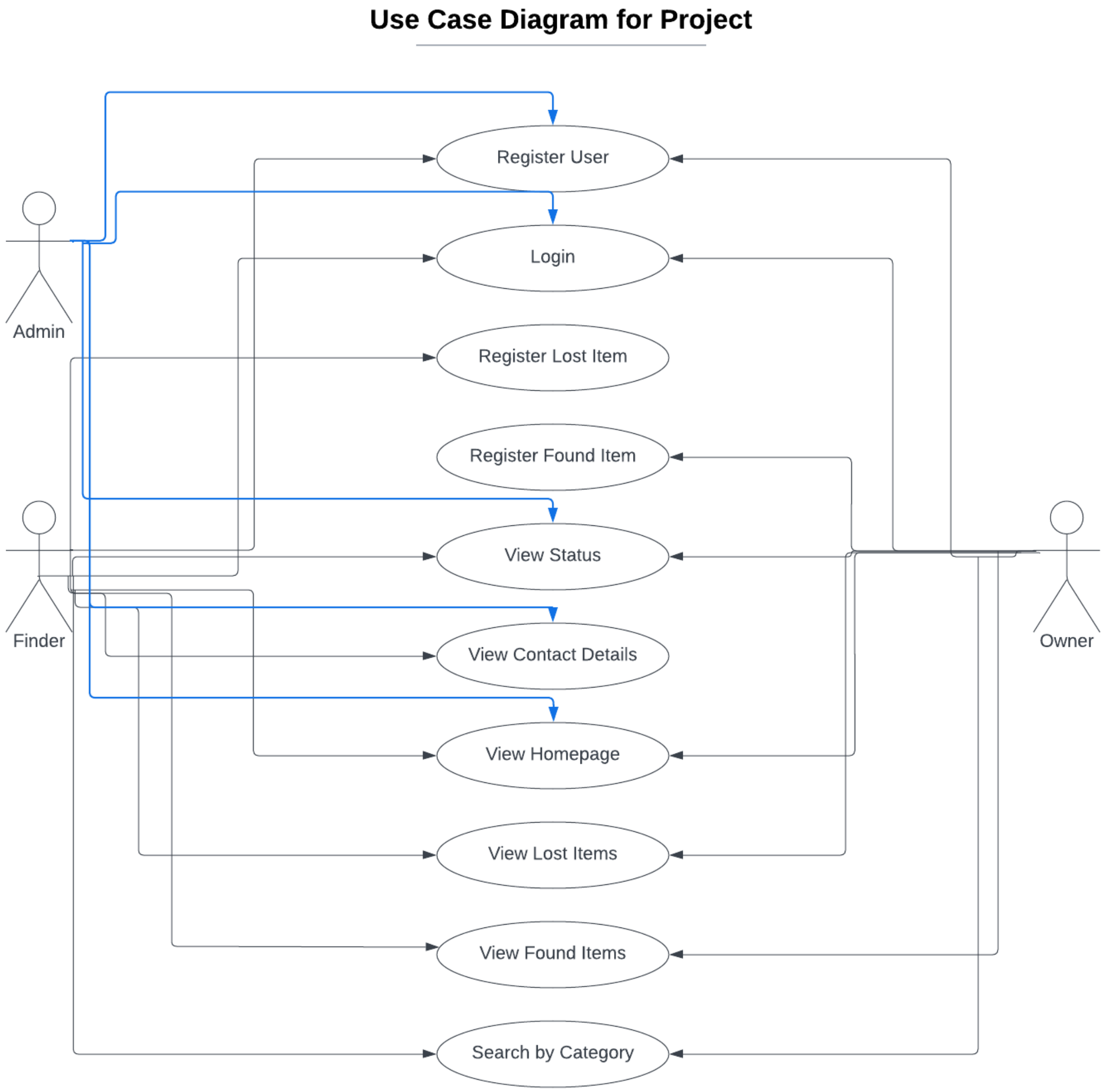
The contact form is a convenient way for users to contact the admin user. The form should collect the user's name, email address, contact number, and message. The admin user can then respond to the user's message using the contact information provided in the form.

4. Analysis Model

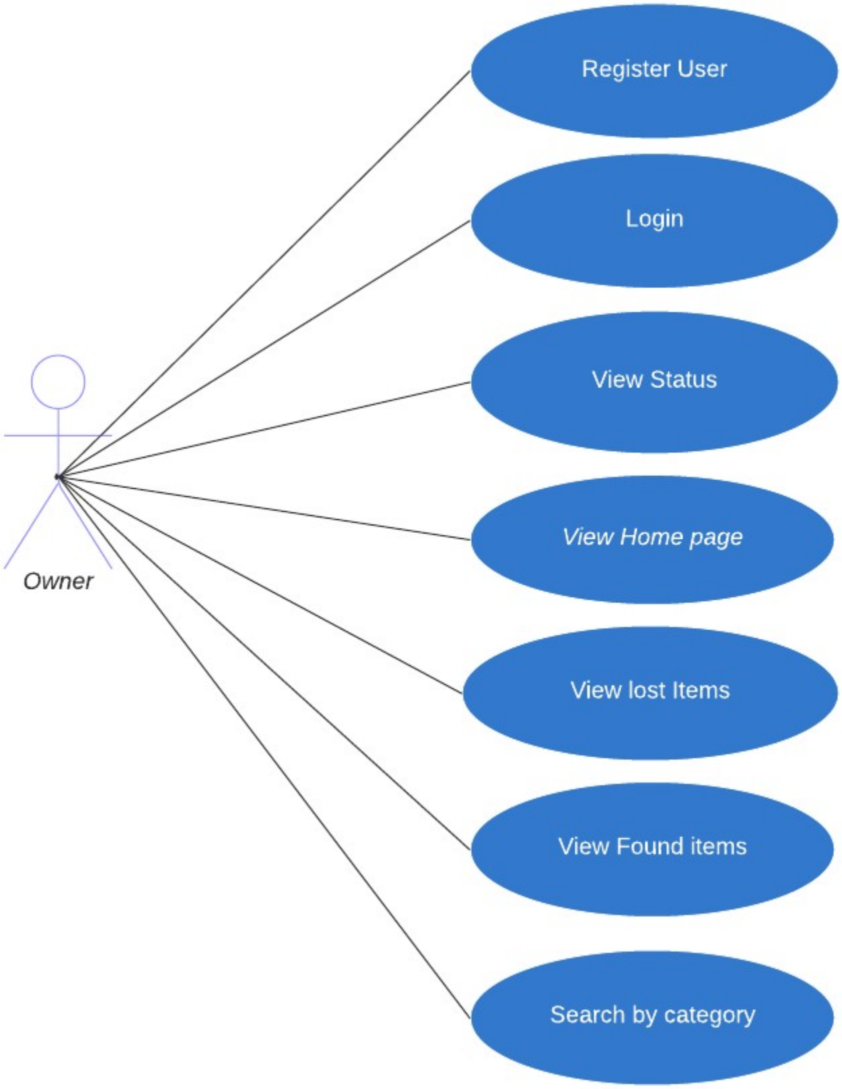
ER DIAGRAM



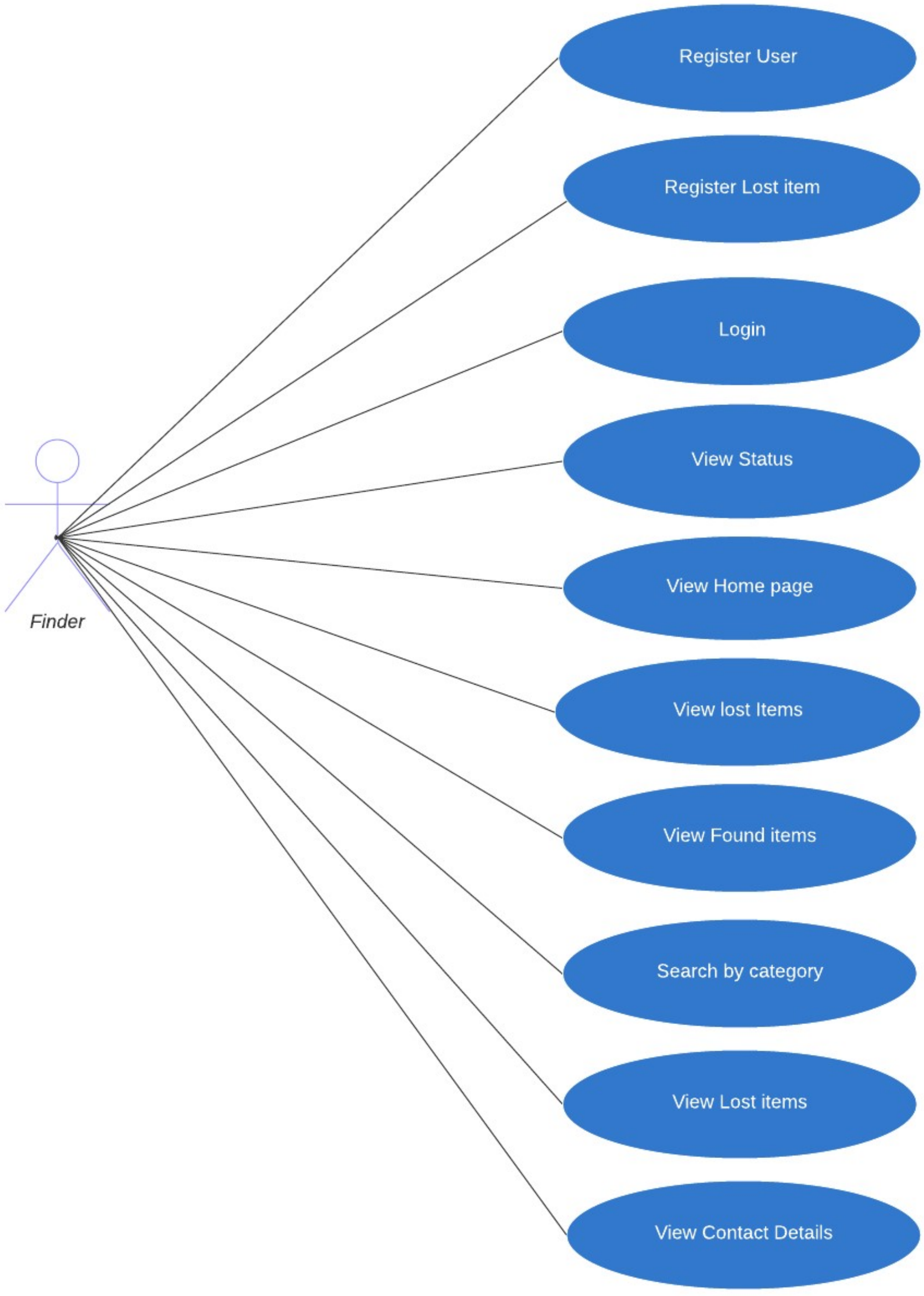
Use Case Diagram for Project:



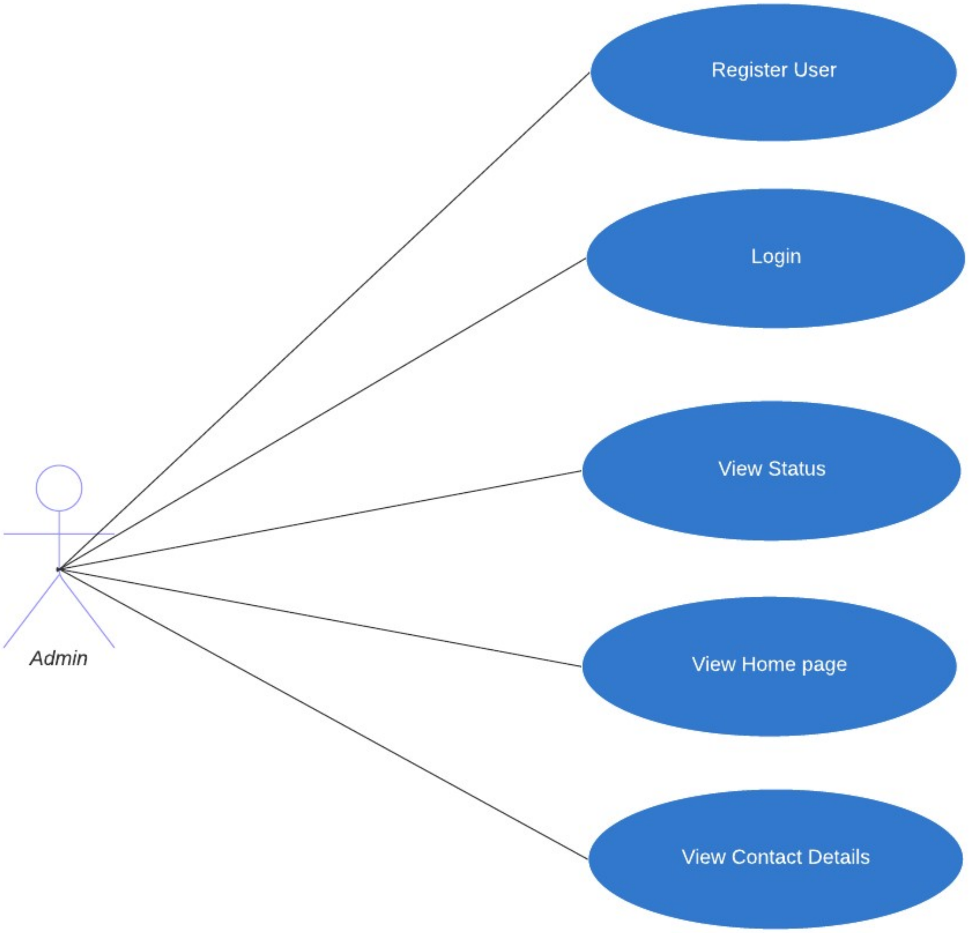
Use case diagram for Owner



Use case diagram for Finder



Use case diagram for Admin



5. System Features

System Feature-1: User login

Description:
User login allows registered users to login to their accounts securely. This feature is of high priority since it ensures user access control and data privacy.

Stimulus/Response sequences:
-User enters his/her registered email and password
-System validates the entered credentials
-If correct, system logs in to the user's account

Functional Requirements:
-REQ-1: The owner/finder can login with the system if he/she is registered with the system
-REQ-2: The owner/finder is able to enter registered email address
-REQ-3: The owner/finder is able to enter a password
-REQ-4: If owner/finder has given valid email address and password, they are able to press the login button
-REQ5: User is able to login into the system

System Feature-2: Report found item

Description:
Users are able to report a found item on the system. This is of high priority found items need to be returned and their status updated accordingly.

Stimulus/Response sequences:
-User enters details of a found item
-System updates the status of the lost item to 'found'

Functional requirements:
-REQ-1: The user is able to report his/her lost item
-REQ-2: The user is able to enter the date on which item was found
-REQ-3: The user is able to enter location where the item was found
-REQ-4: The user is able to enter category of found item
-REQ-5: After entering the details, user is able to press the submit button

System Feature-3: Report lost item

Description:
User is able to report a lost item on the system

Stimulus/Response sequences:
-User enters the details of the lost item
-User clicks on the ‘submit’ button
-System records the lost item into the database

Functional requirements:
-REQ-1: The owner is able to report his/her lost item
-REQ-2: The owner is able to enter date on which his/her item was lost
-REQ-3: The owner is able to enter the category of his/her lost item
-REQ-4: The is able to enter a brief description of his/her lost item
-REQ-5: After entering the details, owner is able to press the submit button

System Feature-4: New user registration

Description:
New user registration allows an owner/finder to register himself/herself on the lost and found management system.

Stimulus/Response sequences:
-User enters his/her details first name and last name
-User enters a valid email address
-User sets a password
-System creates a new account for the user

Functional Requirements:
-REQ-1: The owner or finder can register himself/herself onto the system
-REQ-2: Users are able to enter their first and last name
-REQ-3: The user is able to enter a valid email address
-REQ-4: The user is able to set a password

System feature-5: View status

Functional requirements:
-REQ-1: The owner/finder can check the status of the lost item by entering the reference number of that particular item
-REQ-2: The admin is able to check the performance of the system
-REQ-3: Admin is able to block the user, who is trolling, posting abusive content, etc

System feature-6: View homepage

Functional requirements:

-REQ-1: The lost items highlights are displayed on the main page
-REQ-2: The user can search the lost item from the search bar
-REQ-3: The user is able to report a lost item
-REQ-4: The user is able to report a found item

6. Other Non-functional Requirements

6.1 Performance Requirements:

The system should be able to handle up to 100 lost and found item reports daily and with a response time of under 5 seconds per request.

6.2 Safety Requirements:

The system should not disclose any sensitive information and should comply with all applicable safety regulations within the university.

6.3 Security Requirements:

- **The system should implement authentication.**
- **All sensitive data in the system's database should be encrypted using strong encryption algorithms.**
- **Secure data transmission provided by using HTTPS for communication between client and server.**
- **Security patches and updates should be added regularly to the system.**

6.4 Software Quality Attributes:

The system should emphasize user-friendliness, comprehensibility and ease of maintenance.

6.5 Compatibility Requirements:

- **The system should be compatible with the latest versions of web browsers.**
- **It should also be compatible with mobile devices (such as smartphones and tablets)**

6.6 Logging and Monitoring Requirements:

The system should maintain logs of user interactions, events and errors.

6.7 Business Rules:

- **Found items can be categorized only by the admins.**
- **Only members of the university (students and faculty) can report lost items.**
- **When reporting lost items, reporters must provide a detai led description of the item.**
- **User consent - users using the Lost and Found system for reporting and claiming lost items should abide by the system's policy.**

Appendix A: Glossary

Term	**Definition**
SRS (System Requirements Specification)	Document is to provide a comprehensive overview of the University-Level Lost and Found Management System
REQ (Requirements)	Requirements in an SRS (Software Requirements Specification) are descriptions of what the software system must do and how it must perform.
OAuth	OAuth is an open standard for access delegation, commonly used as a way for internet users to grant websites or applications access to their information on other websites but without giving them the passwords.
PHP Mailer API	The PHPMailer API is a PHP library that provides a simple and flexible way to send emails from a PHP script.
MySql API	The MySQL API is a set of functions and classes that developers can use to interact with MySQL databases.
Security Patches	A security patch is a software update that fixes a

	security vulnerability. Security vulnerabilities are weaknesses in software that can be exploited by attackers to gain access to systems or data.

Appendix B:

Field Layout:

Field	Length	Data Type	Description	Is Mandatory
Item ID	5	Alphanumeric		Y
Item Name	20	String		Y
Category	8	String	The category to which the item belongs	Y
Color	8	String	Color of the lost item	Y
Size	25	Alphanumeric	Size of the lost item	Y
User Name	60	String		Y
Brand	4	String	Reject Reason code in case mandate is rejected	N
Item Description	50	String	Additional details about item appearance	

Lost location	25	String	Location where the item was lost	Y
Finder’s Name	20	String	Name of the finder	Y
Owner’s Name	20	String	Name of person claiming ownership	Y
Contact Information	10	Numeric	Contact information of the owner	Y

Appendix C: Requirement Traceability Matrix

Sl. No	Requirement ID	Brief Description of Requirement	Architecture Reference	Design Reference	Code File Reference	Test Case ID	System Test Case ID
1.	REQ-01	Users can report lost items	UseCase Diagram	Use Case Specification	Implementation File	TC-01	STC-01
2.	REQ-02	Users can report found items.	UseCase Diagram	Use Case Specification	Implementation File	TC-02	STC-02
3.	REQ-03	Record lost item details: description, category, and location.	Entity-Relationship Diagram	Entity-Relationship Diagram	Data Model File	TC-03	STC-03
4.	REQ-04	Record found item details: description, category, and location.	Entity-Relationship Diagram	Entity-Relationship Diagram	Data Model File	TC-04	STC-04
5.	REQ-05	Users can claim ownership of found items.	UseCase Diagram	Use Case Specification	Implementation File	TC-05	STC-05

6.	REQ-06	Associate found items with claimants.	Use Case Diagram	Use Case Specification	Implementation File	TC-06	STC-06
7.	REQ-07	Track the location where items were lost.	Entity-Relationship Diagram	Entity-Relationship Diagram	Data Model File	TC-07	STC-07
8.	REQ-08	Track the location where items were found.	Entity-Relationship Diagram	Entity-Relationship Diagram	Data Model File	TC-08	STC-08
9.	REQ-09	Users can search for lost items.	Use Case Diagram	Use Case Specification	Implementation File	TC-09	STC-09
10.	REQ-10	Users can search for found items.	Use Case Diagram	Use Case Specification	Implementation File	TC-10	STC-10