1. **PLANNING PHASE**
   1. **Project Overview**

The Lost and Found Management App is a digital solution specifically designed to streamline the process of managing lost and found items within educational institutions. The system allows students, faculty, and staff to report lost items, submit found items, and claim belongings efficiently through an intuitive platform. By integrating digital record-keeping and automated notifications, the app enhances transparency, minimizes administrative workload, and significantly improves the likelihood of lost items being returned to their rightful owners. This solution eliminates the inefficiencies of traditional manual lost and found methods, replacing them with a modern, structured approach

**Stakeholders:**

* **Primary Users**:
  + **Students**: The primary users who will report lost items, check for found items, and claim belongings they lost. They will also be responsible for tracking their claims and responding to notifications from the system.
  + **Faculty Members**: Faculty members who might also use the app to report lost items, help identify and verify found items and provide confirmation on claims made by students or other faculty.
* **Secondary Users:**
  + **Administrators/IT Support Staff**: Responsible for maintaining the system, managing user data, verifying claims, and ensuring the application operates smoothly.
  1. **Objectives**

The primary objectives of the Lost and Found App include:

* **Efficient Lost Item Management**: Provide a platform for students and faculty to report lost items and find items that others have found, improving campus-wide lost and found operations.
* **Real-Time Claim Tracking:** Enable users to claim lost items and receive notifications about claim statuses, ensuring timely action for both lost and found items.
* **Data-Driven Insights**: Offer administrators and faculty insights into lost and found trends through analytics and usage reports.
* **Streamlined Claim Verification Process**: Allow faculty or administrators to verify claimed items efficiently, minimizing the chances of incorrect claims.

Secondary objectives include:

* Reducing the need for manual lost-and-found management in physical spaces.
* Providing users with a secure platform where they can safely report or claim items without relying on public bulletin boards or physical submissions.
* Enhancing user engagement by offering an intuitive interface that makes it easy for users to interact with the system.
* Providing students with a reliable, self-service platform that encourages them to resolve lost item issues on their own.
  1. **Scope**

The Lost and Found app aims to provide a user-friendly platform for reporting, searching, and claiming lost items. This section outlines the key features that are within the scope of the app's development and those that are outside the app’s intended functionality.

* **In-Scope**:
  + Mobile app (iOS & Android) for reporting, searching, and claiming lost items.
  + Admin panel for verification and approval of claims.
  + Secure user authentication.
  + Notifications about item status updates.
  + Search filters for sorting items by category, time, and location.
* **Out-of-Scope**:
  + GPS tracking of lost items
  + Integration with third-party security systems
  + Direct physical handling of lost and found items (the app will only facilitate reporting and communication)
  + Delivery or courier service for returning items
  1. **Assumptions**

The development of the Lost and Found app is based on several key assumptions that guide its functionality, user experience, and overall design. These assumptions help create a clear framework for addressing the needs of users while ensuring that the app remains efficient, intuitive, and effective in fulfilling its purpose. The app is designed with the idea that users will actively engage with it to report lost or found items, claim them, and maintain a transparent process.

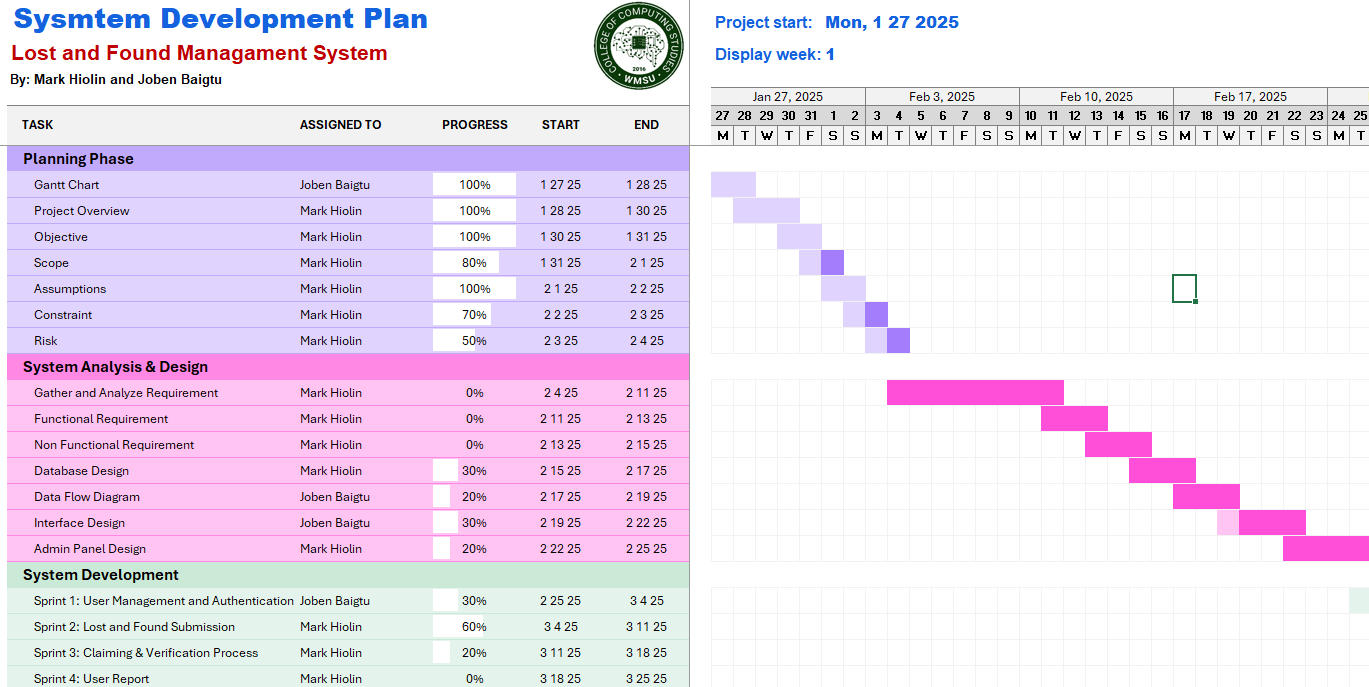
* **Mobile Device Availability:** All students and faculty will have access to smartphones with internet access to download and use the app.
* **Security & Privacy**: The app will comply with data protection policies to ensure that personal data, particularly regarding lost and found items, is handled securely.
* **Faculty Training**: Faculty members will be trained on how to use the admin panel to manage items, verify claims, and update the system.
* **Consistency of Lost Items**: The app will be used consistently to report lost items. Users will be encouraged to use the platform to report items, and faculty will ensure proper handling on the verification side.
  1. **Constraints**

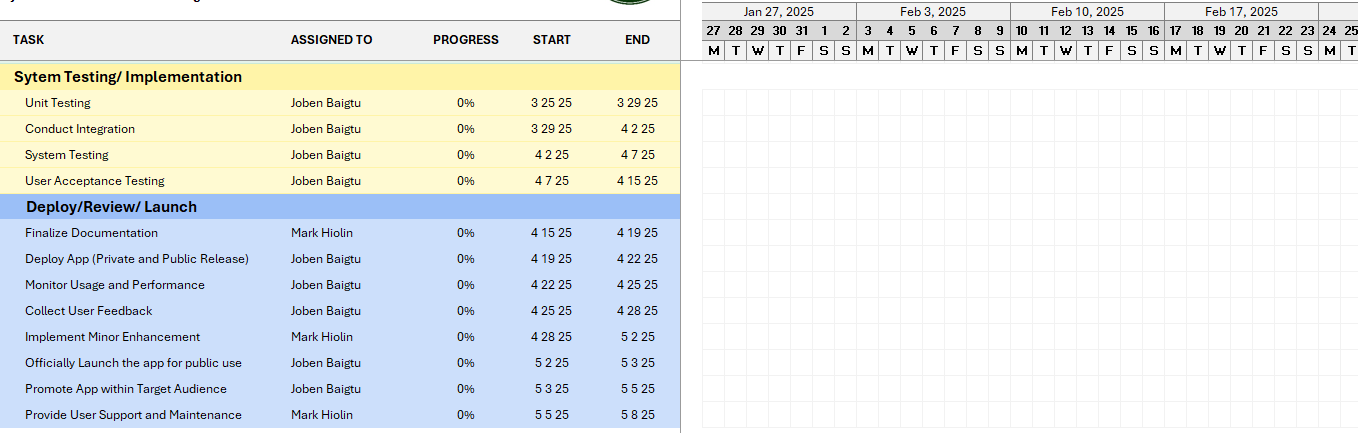
The development of the Lost and Found app comes with several constraints that will shape its features, performance, and overall implementation.

* The app will need to support both Android and iOS but may be initially optimized for one platform before expanding to others.
* Limited resources for development and maintenance.
* The app will not be responsible for physically storing or handling the items. It will only track them digitally.
* No third-party integrations for payment gateways or external tracking systems unless required by the institution later.
* The app should be optimized for both low-end and high-end mobile devices.
* Budget constraints may limit the complexity of features.
* The app must comply with the university's IT and security policies, especially related to data storage and access permissions.
  1. **Risks**

While the Lost and Found app is designed to improve the process of reporting, searching, and claiming lost items, there are several potential risks that may affect its effectiveness and security.

* Potential misuse of the platform for false reporting.
* Low adoption rates due to lack of awareness or resistance to change. Students and faculty may not fully utilize the app, leading to inefficiency.
* System downtime or technical issues affecting operations. Possible bugs, downtime, or performance issues that could affect usability.
* Potential risks of unauthorized access or data breaches.
  1. **Gantt Chart**





1. **SYSMTEMS ANALYSIS**

The Lost and Found Management System is designed to assist individuals in reporting and locating lost and found items efficiently. This app provides a platform for users to report lost and found belongings, search for items, and receive notifications when a match is found. The system ensures that users can easily interact with it through a simple, intuitive interface. The purpose of this system is to facilitate a more organized and convenient method of managing lost items, ultimately helping users reconnect with their belongings in a timely and secure manner. This analysis outlines the functional and non-functional requirements necessary to develop a robust, user-friendly, and secure app that meets the needs of users, administrators, and system stakeholders.

* 1. **Functional Requirements**
* User Registration & Authentication: Users can create accounts and log in securely. Basic authentication will be used, and options for social media logins may also be integrated for user convenience.
* Lost Item Reporting: Users can report lost items by providing details such as item description, category, location, and image. Reports will be stored in the database for future reference and searching.
* Found Item Reporting: Users can report found items in a similar manner to lost item reports, including item details and images.
* Search Functionality: Users can search for lost or found items based on item category, description, location, and other filters.
* Notifications & Alerts: The app will notify users when a match is found between a lost item and a reported found item. It will also send updates when the status of an item is updated (e.g., item claimed, returned, etc.).
* Item Status Management: Users can update the status of their items (e.g., marking a lost item as found or a found item as returned).
* Admin Panel: Admins can manage the system by reviewing and approving/rejecting item reports, monitoring user interactions, and resolving disputes when necessary.
  1. **Non-Functional Requirements**
* **Usability**: The app should have a simple, intuitive interface, ensuring users can quickly understand how to use it, especially on mobile devices.
* **Performance**: The system should handle a high volume of reports and user interactions without performance degradation, offering fast load times and seamless interactions.
* **Security**: User data must be stored securely, with sensitive information encrypted. Secure authentication methods should be used, including password hashing.
* **Scalability**: The system should scale to handle an increasing number of users and reports over time, ensuring that it remains responsive even with growing usage.
* **Availability**: The app should have a high availability rate, with minimal downtime, targeting a 99% uptime. Data should be backed up regularly to prevent loss of information.
* **Compatibility**: The system should be compatible across multiple platforms, including both Android and iOS devices, providing a consistent experience on various screen sizes and resolutions.
  1. **Systems Architecture**

The Lost and Found App is a mobile developed using Adalo, a no-code platform that allows rapid app development with a visual interface. This document describes the system architecture of the application, detailing its structure, components, and interactions.

* **Architecture Overview**

This architecture ensures scalability, maintainability, and efficient data management.

* + The Lost and Found App follows a Three-Tier Architecture, which consists of the following layers:
  + Presentation Layer (Frontend/UI)
  + Business Logic Layer (Backend Processing)
  + Data Layer (Database/Storage)
* **Component of the System**
  + - **Presentation Layer (FrontEnd/UI)**
      * The user interface is built using Adalo’s drag-and-drop editor, which allows seamless interaction with the app.
      * Users can submit lost or found items, search for reports, and communicate with item owners or finders.
      * The app is accessible via mobile (Android & iOS).
    - **Business Logic Layer (Backend Processing)**
      * The backend is managed within Adalo’s workflow automation.
      * This includes:
        + User authentication (login/signup via email, Google, or social media).
        + Conditional logic for matching lost and found items based on location, description, or category.
        + Push notifications for updates on found items.
        + API integration (if needed) for external services like Google Maps.
    - **Data Layer (Database/Storage)**
      * The database is managed by Adalo’s built-in NoSQL database.
      * Data types include:
        + User Profiles: Name, contact details, and preferences.
        + Lost Items: Item name, description, location, date lost and contact details.
        + Found Items: Item details, location, finder’s contact information.
        + Messages & Notifications: Communication between users.
      * The database supports CRUD (Create, Read, Update, Delete) operations for managing lost and found items efficiently.
  1. **Requirements Gathering and Analysis**

This outlines the comprehensive process for gathering and analyzing requirements for the Lost and Found Management System (L&FMS) built using Adalo. The goal is to ensure the application meets the needs of all stakeholders and operates efficiently.

**Scope**

This document outlines the comprehensive process for gathering and analyzing requirements for the Lost and Found Management System (L&FMS) built using Adalo. The goal is to ensure the application meets the needs of all stakeholders and operates efficiently.

**Definitions, Acronyms, and Abbreviations:**

* Adalo: A no-code platform for building mobile and web apps.
* LFMS: Lost and Found Management System.
* UI: User Interface.
* UAT: User Acceptance Testing.

**User Characteristics**

* **Students:** Regular users who will report lost items and check for found items.
* **Staff:** Users who will report found items and assist with claims.
* **Administrators:** Users who will manage the system, verify claims, and generate reports.
  1. **Detailed Requirements Specifications**

**User Registration and Login**

* **Description**: Users can create accounts, log in, and manage their profiles to access the lost and found system.
* **Acceptance Criteria:** Users must provide valid credentials to log in, and the system must securely store and retrieve user data. Only registered users can report, claim, or manage lost and found items.
* **User Stories:**

1. As a student, I want to create an account so that I can report and track my lost and found items.
2. As a finder, I want to log in so that I can report items I have found and manage claim requests.
3. As an administrator, I want to log in to monitor user activities and manage the lost and found database.

**Lost Item Reporting System**

* **Description**: Users can report lost items, track their status, and update their reports. The system ensures that all reported lost items are recorded, searchable, and can be marked as found or deleted when necessary.
* **Acceptance Criteria**: Users must provide details about the lost item when reporting it. The system must store the report and allow users to update, delete, or mark the item as found. A history of all reported lost items must be maintained for reference..
* **User Stories:**

1. As a user, I want to report a lost item so that I can seek assistance in finding it.
2. As a user, I want to mark my lost item as found if I recover it myself to keep the records accurate.
3. As a user, I want to delete a lost item report if it is no longer relevant.
4. As a user, I want to view the history of my lost item reports to track past records.
5. As an administrator, I want to manage and monitor reported lost items to ensure proper handling.

**Found Item Reporting System**

* **Description**: Users can report items they have found, either by selecting a lost item from the system or by filing a new found item report. The system ensures proper tracking and management of found items, preventing finders from claiming the items themselves.
* **Acceptance Criteria:** Users must provide details about the found item when reporting it. If an item is reported as found from the lost item list, it will be moved to the original owner’s "My Claimable Items" list. Found items reported by users will be stored in their "My Found Items" list, and by default, the status will be "No Claimant Yet."
* **User Stories:**

1. As a finder, I want to report a found item so that its owner can claim it.
2. As a finder, I want all found items I report to be listed under "My Found Items" for reference.
3. As a finder, I want the system to prevent me from claiming an item I reported as found to ensure fairness.
4. As a user, I want my lost item to appear under "My Claimable Items" if someone reports it as found.
5. As an administrator, I want to monitor all found item reports to ensure proper handling.

**Claiming Process**

* **Description**: Users who have lost an item can request to claim it once it has been reported as found. The system ensures that only eligible users can request a claim and that finders cannot claim their own reported found items. The claim request process involves verification by the finder, who can either approve, deny, or turn over the item to the administrator for further handling.
* **Acceptance Criteria:** A user must submit a claim request with supporting details before proceeding. Once a claim request is made, the item is no longer visible in the public "Found Items" list but remains in the "My Found Items" list of the finder with a status update to "Claim request by [User]." The finder then decides how to handle the request..
* **User Stories:**

1. As a user, I want to submit a claim request for my lost item to recover it.
2. As a user, I want to provide proof of ownership when requesting a claim to verify my ownership.
3. As a finder, I want to receive claim requests and review them before approving or denying the request.
4. As a finder, I want the ability to approve a claim and fill in the claimant’s details to confirm the handover.
5. As a finder, I want to deny a claim if the proof provided is insufficient, making the item available for other claimants
6. As a finder, I want to turn over an item to an administrator if I cannot determine the rightful owner.
7. As an administrator, I want to handle turned-over items and verify claim requests to ensure proper item retrieval.

**Reporting a Finder**

* **Description**: Users can report a finder if misconduct is observed during the item return process. This ensures fair handling of lost and found items and prevents misuse of the system. Once a finder is reported, the item is removed from the public "Found Items" list and transferred to the administrator for further action. The administrator can issue a warning, suspend the finder, or mark the issue as resolved.
* **Acceptance Criteria:** A user can only report a finder if an item has been marked as found and a claim request is in progress. The system must ensure that the claimant has requested a claim before allowing them to report a finder. Once reported, the administrator takes action based on the severity of the issue.
* **User Stories:**

1. As a claimant, I want to report a finder if they engage in misconduct during the claim process.
2. As a claimant, I can only report a finder if I have already submitted a claim request.
3. As a system, I want to remove a reported finder’s item from the public "Found Items" list until the issue is resolved.
4. As an administrator, I want to review reported finders and decide on the appropriate action.
5. As an administrator, I want to issue a warning that limits the finder’s ability to report lost or found items.
6. As an administrator, I want to suspend a finder’s account if they have repeated offenses, restricting their access to the system.
7. As an administrator, I want to mark the issue as resolved, release the item, and document the claimant’s details for record-keeping.

**Messaging Functionality**

* **Description**: Users can communicate with a finder to facilitate the claim process for a found item. Messaging is only enabled after a claim request has been submitted. Alternatively, users can initiate a direct conversation through the chat module by searching for the finder’s name.
* **Acceptance Criteria:** A user can only message a finder if a claim request has been submitted. The chat module should also allow users to search for and start a conversation with a finder directly. The system must ensure that only authorized users can send messages related to lost and found items.
* **User Stories:**

1. As a user, I want to message a finder after submitting a claim request to coordinate the retrieval of my lost item.
2. As a user, I want to search for a finder’s name in the chat module and start a conversation directly.
3. As a finder, I want to receive messages from claimants to discuss the verification process.
4. As a system, I want to enable messaging only when a claim request is submitted to prevent spam.
   1. **Database Design**

The database design for this system follows a relational structure where different entities (tables/collections) are linked together using relationships. The primary goal is to streamline the reporting, tracking, and claiming process of lost and found items while keeping records accurate and accessible.

Key Component of the Database Design

* **Users Table** - Stores user information, including their name, contact details, and authentication credentials.

Purpose: To manage user profiles and enable authentication.

Relationships: Connected to lost items, found items, and claims.

* **Categories Table** - Defines different types of lost and found items, such as Electronics, Bags, Keys, and Documents.

Purpose: To classify items for better search and filtering.

Relationships: Linked to both lost and found items.

* **Lost Items Table** - Stores details of items reported as lost, including descriptions, images, and the date/location where they were lost.

Purpose: To allow users to list and track their lost items.

Relationships: Connected to users, categories, and claims.

* **Found Items Table** - Stores details of items reported as found, with information about their discovery.

Purpose: To let users post found items and help reconnect them with owners.

Relationships: Linked to users, categories, and claims.

* **Claims Table** - Manages item claim requests, ensuring that users provide valid proof before retrieving their lost items.

Purpose: To verify ownership and prevent false claims.

Relationships: Connected to lost and found items, users, and admins.

* **Messages Table** - Facilitates communication between users, allowing them to coordinate item returns.

Purpose: To improve interaction between finders and owners.

Relationships: Links users in conversations.

**How the Database Supports the System**

✅ Data Integrity → Ensures that only valid users can post and claim items.

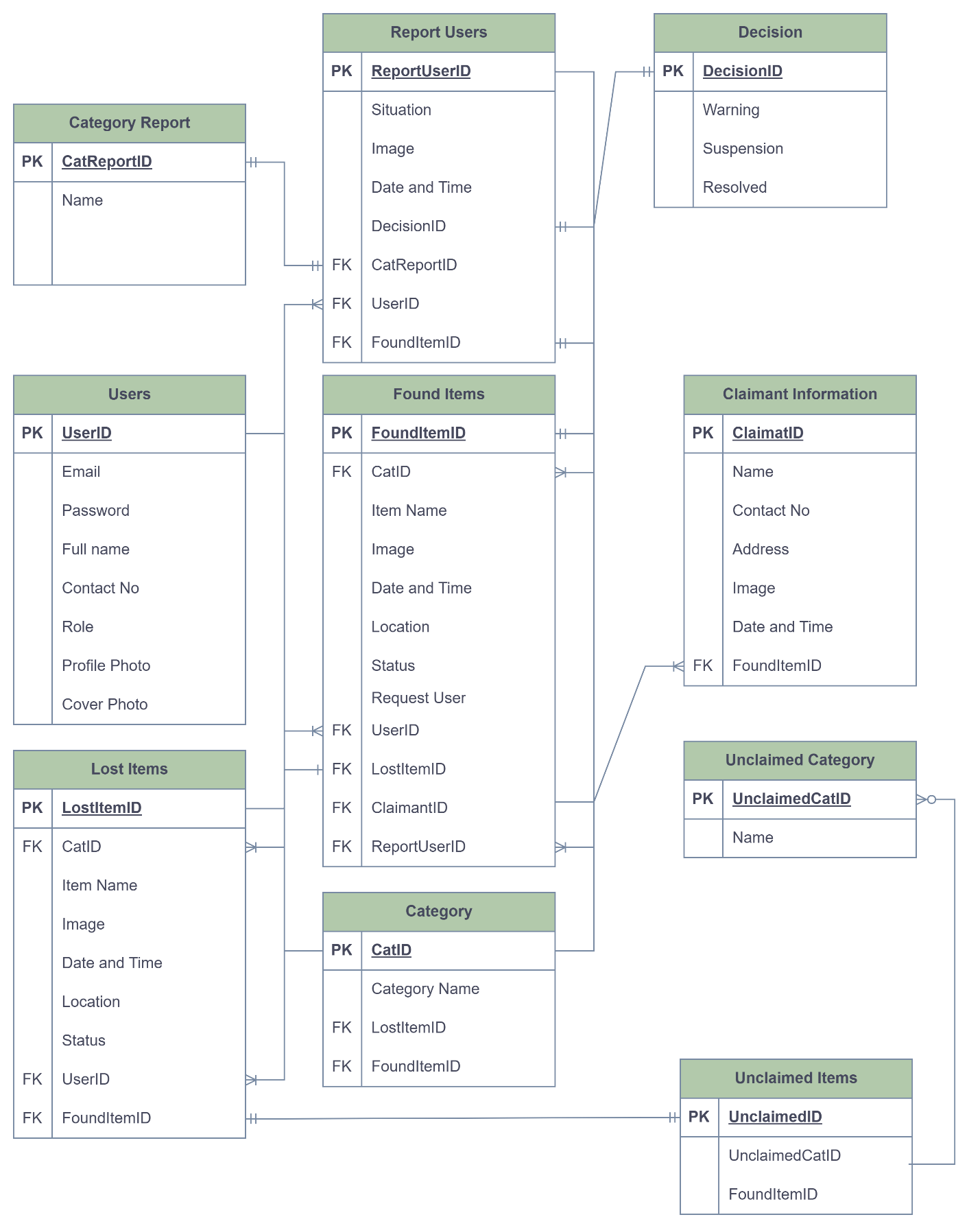
✅ Efficient Searching → Categorized and searchable lost and found listings.

✅ Claim Verification → Prevents fraudulent claims using proof submission.

✅ User Interaction → Optional messaging feature for seamless coordination.

✅ Scalability → Can support a growing number of users and reports.

This well-structured database design ensures that the Lost and Found App operates smoothly, providing a reliable and efficient solution for reconnecting lost items with their rightful owners.

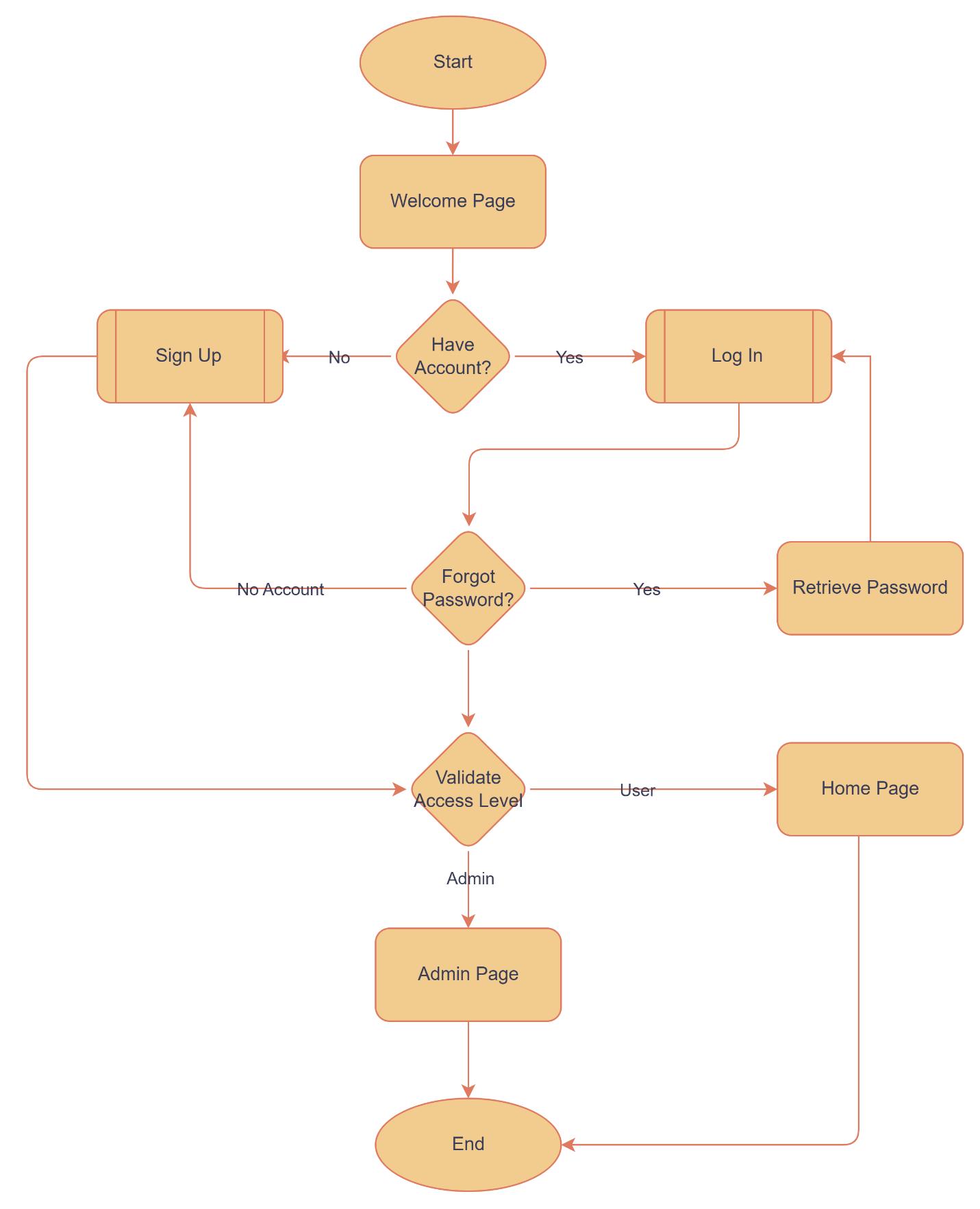


* 1. **Data Flow Diagram**

A Data Flow Diagram (DFD) represents how data moves through the Lost and Found App, showing the interaction between users, processes, and the database. It helps in understanding the system’s workflow, inputs, outputs, and data storage.

* **Process Flow: User Authentication**

The process begins with the Welcome Page, where the user is greeted with a friendly message and presented with two options: "Login" for existing users or "Sign Up" for new users.

* + - * 1. The apps starts in the Welcome Page
        2. Account Checks – The user is asked: “Do you have an account?”

If No, proceed to Sign Up

If Yes, proceed to Login

* + - * 1. Sign Up Process (For New User) – User provides necessary details(e.g. name, email, password). System stores the user information in the database.

Login Process (For Existing Users). User enters email and password

If correct, proceed to validate access level

If incorrect, prompt Forget Password.

Forget Password (If Selected) – If the user selects Forget Password, they are redirected to retrieve password. The system verifies the identity an allows the user to reset the password

Access Level Validation – The system checks if the user is an Admin or a Regular user.

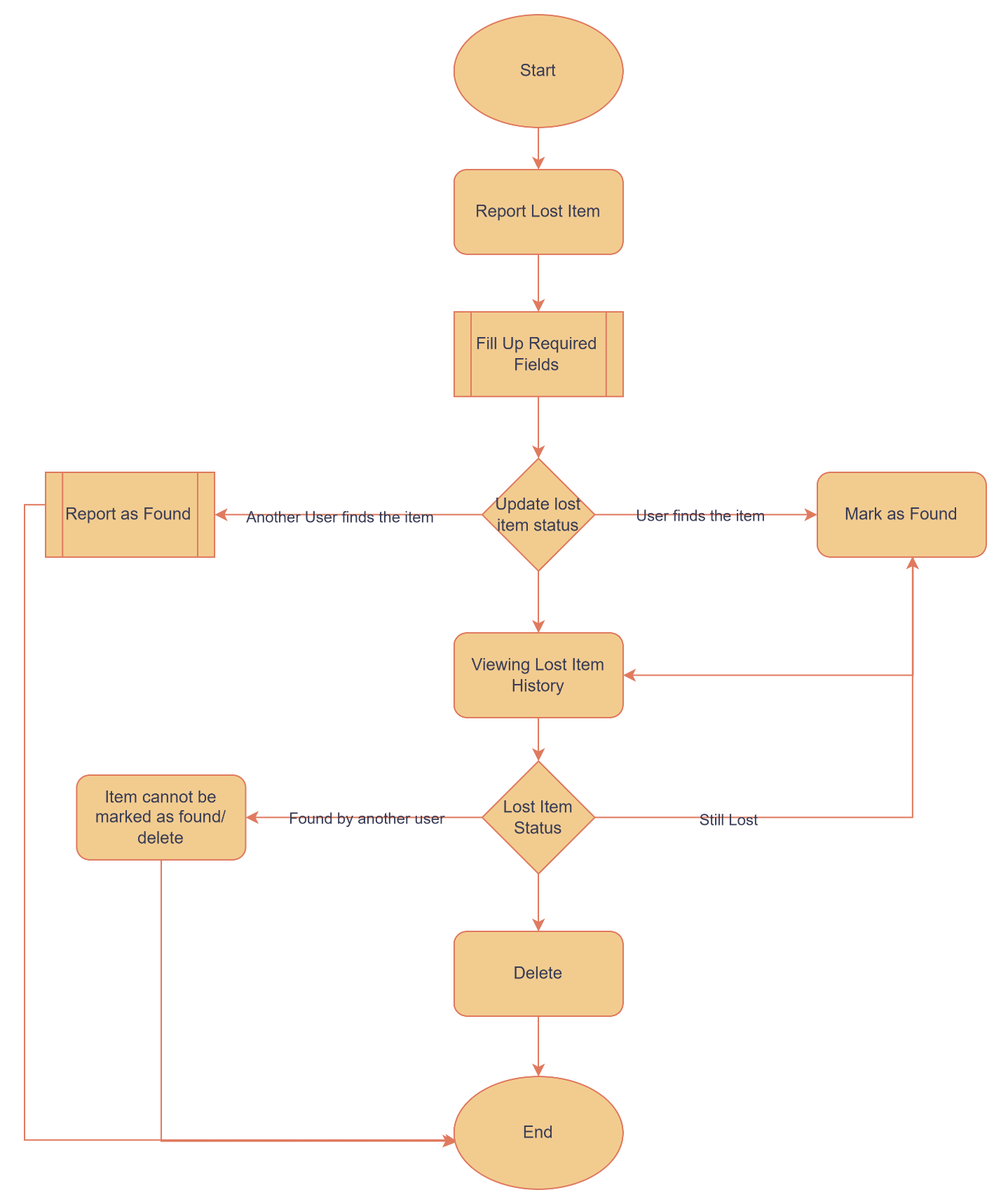
If User, redirect to Home Page

If Admin, redirect to admin page

The process ends after the user successfully logs in to their respective page.

* + - * **Process Flow: Lost Item Reporting**

Help users efficiently report, track, and manage lost items. It provides a structured process for filing lost item reports, updating their status when found, and maintaining a history of reports. Users can mark their own lost items as found or delete reports when applicable, while restrictions are in place to ensure data accuracy when another user reports an item as found. This system improves lost-and-found management by offering a transparent and organized approach to handling lost belongings.



Filing a Lost Item Report – Th user logs into the app. The user navigates the to the “Report Lost Item” section. The user fills out a form with item details.

Updating a lost item status – The user checks the Lost Item List to track their lost item.

Two possible ways the item can be marked as found.

User finds the item: the user select the lost item and mark it as found

Another user finds the item: The finder reports it as “Found”, updating the item’s status automatically

Deleting a Lost item report – A user can delete their own lost item report if:

The item is still reported as lost.

The user manually marked it as found

If another user reported the item as found, deletion is no longer allowed.

Viewing Lost Item History – All lost item reports are stored in History

If the item is still lost, the user can mark as found or delete the report.

If the item is already reported as found by another user. The user cannot mark it as found and the user cannot delete the report.

1. System enforcement rules – Only the original filer can update or delete their report. Once an item is mark as found by another user, further changes are restricted.
2. History maintains a record of all lost item even after resolution.
3. **Process Flow: Found Item Reporting**

The Found Item Reporting Process provides users with two methods to report items they have found. Users can either select a lost item from the Reported Lost Items List and mark it as found, or they can manually file a new found item report. When an item is reported as found, it is added to the finder's "My Found Items" List and, if applicable, moved to the original owner's "My Claimable Items" List. The system ensures proper tracking by setting the item's status to "No Claimant Yet" until a claim request is made. This process helps streamline the management of lost and found items, ensuring that rightful owners are notified and can retrieve their belongings efficiently.

1. **A diagram of a diagram

   AI-generated content may be incorrect.**Choosing how to report a found item – The user has 2 options to report a found item.
2. Option 1: Reporting item from the lost list.

The user navigates to the reported lost item list.

User selects an item that matches the found item

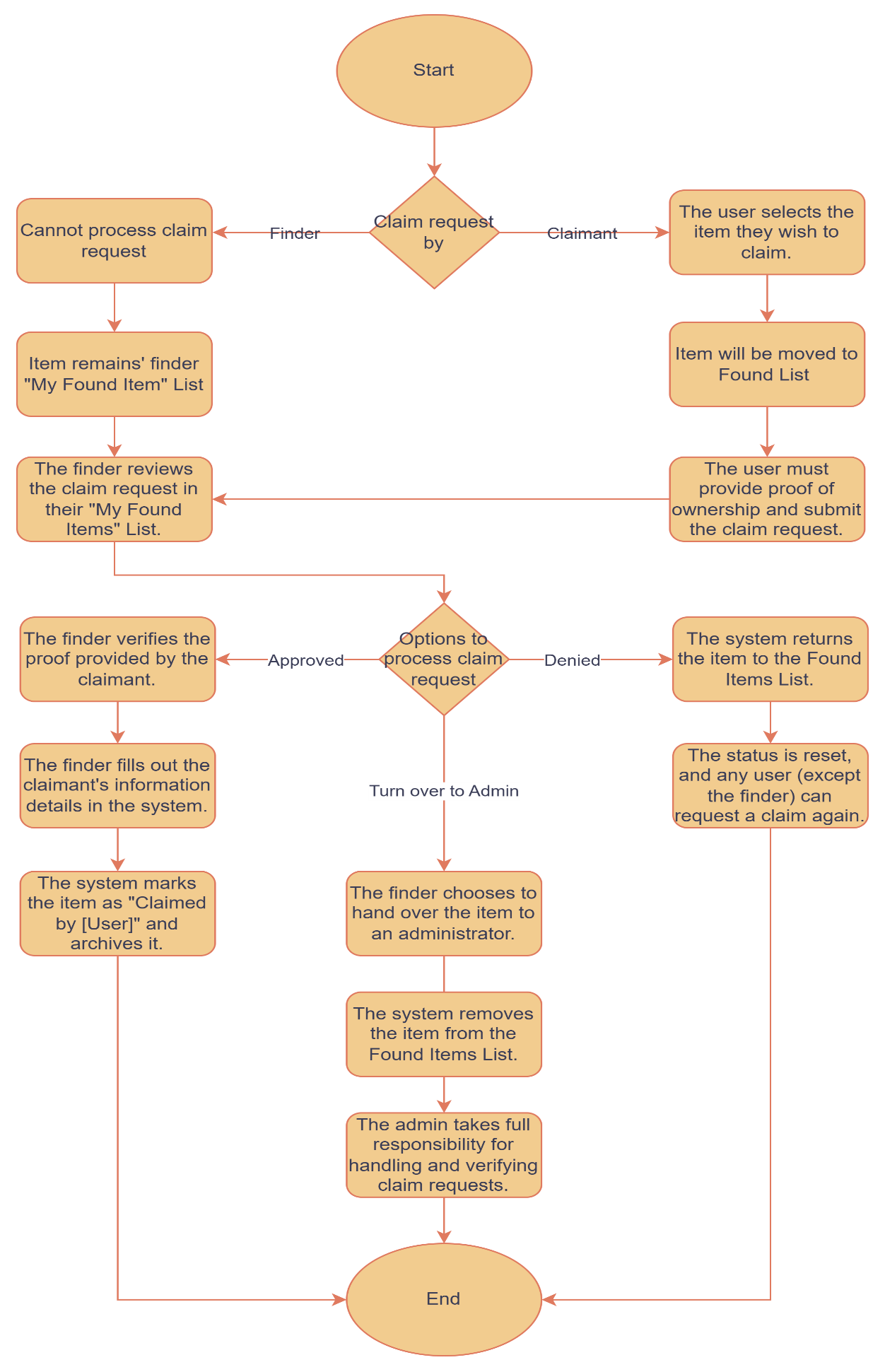
The app update the item’s status to “found” and notifies original filer.

The item is added to the user’s “My Found Items” list

By Default, the status of the item is set to “No Claimant Yet”

1. Filing a new found item – The user goes to the “Report Found Item” section
2. The user fills out the required details.
3. The app saves the report and add the item to the “My Found Item” List.
4. By Default, the item’s status is set to “No Claimant Yet”
5. **Process Flow: Claim Request**

The Claim Request Process allows users to request ownership of found items while ensuring proper verification and accountability. When an item is found, any user—except the finder—can submit a claim request by providing proof of ownership. Once a claim is submitted, the item is removed from the public Found Items List but remains in the finder's "My Found Items" List with a status update.



Step-by-Step Process for Claiming a Found Item

* + Requesting a Claim for a Found Item
  + Any user (except the finder) can request a claim for an item listed in the Found Items List.
* The user selects the item they wish to claim.
* The system updates the item's status to "Claim Request by [User]".
* The item is removed from the public Found Items List but remains in the finder's "My Found Items" List with the updated status.
* The user must provide proof of ownership and submit the claim request.

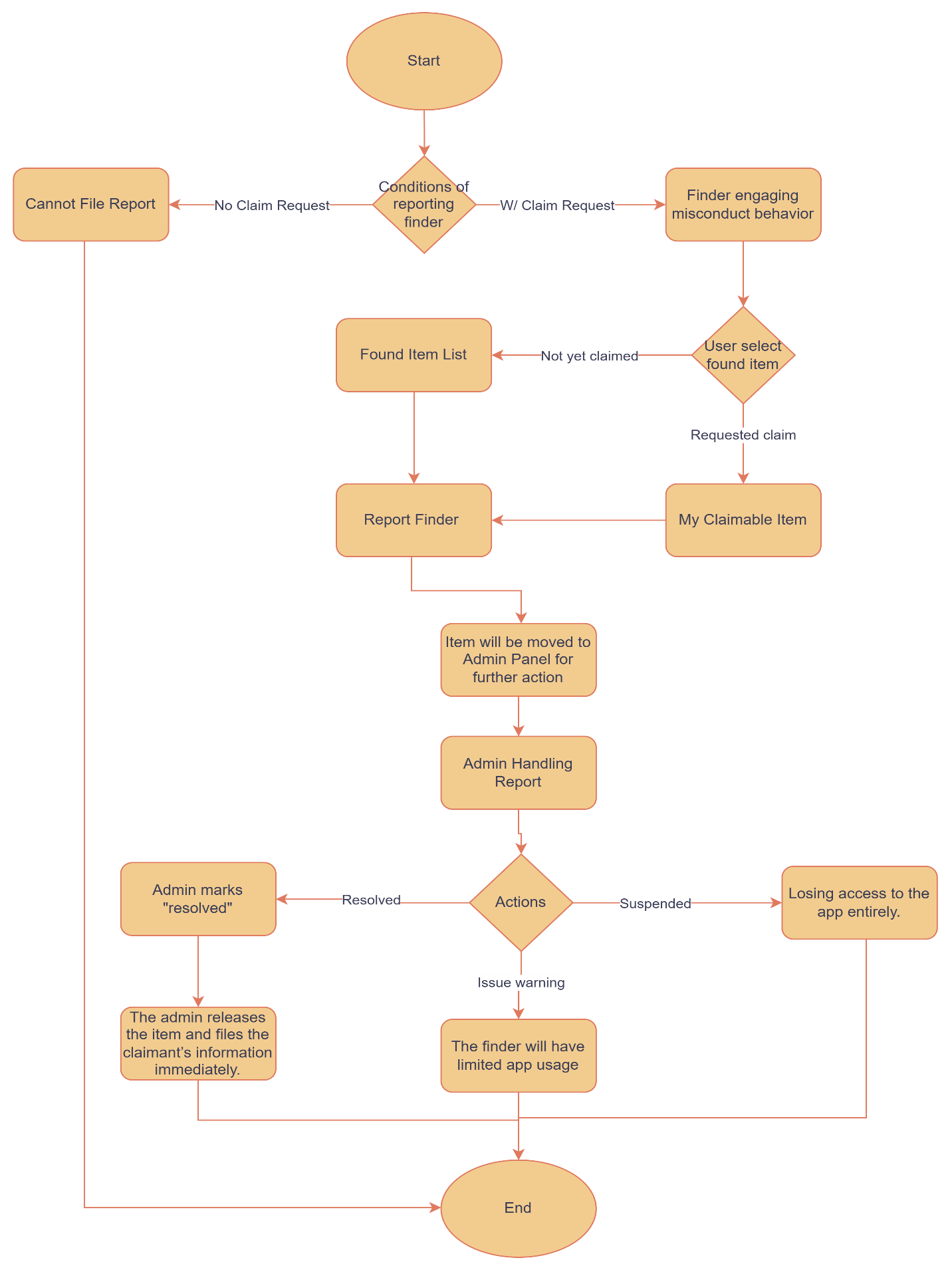
1. Finder Handles the Claim Request

* The finder reviews the claim request in their "My Found Items" List.
* The finder has three options to process the request:
* Option 1: Approve the Claim Request
  + The finder verifies the proof provided by the claimant.
  + If approved, the finder fills out the claimant's information details in the system.
  + The system marks the item as "Claimed by [User]" and archives it.
* Option 2: Deny the Claim Request
  + If the finder rejects the claim, the system returns the item to the Found Items List.
  + The status is reset, and any user (except the finder) can request a claim again.
* Option 3: Turn Over the Item to the Admin
  + The finder chooses to hand over the item to an administrator.
  + The system removes the item from the Found Items List.
  + The admin takes full responsibility for handling and verifying claim requests.

1. Finalizing the Claim Process

* If the item is approved and claimed, it is removed from all active lists.
* If denied, it is made available again for new claim requests.
* If turned over to the admin, the admin oversees the claim process from that point forward.

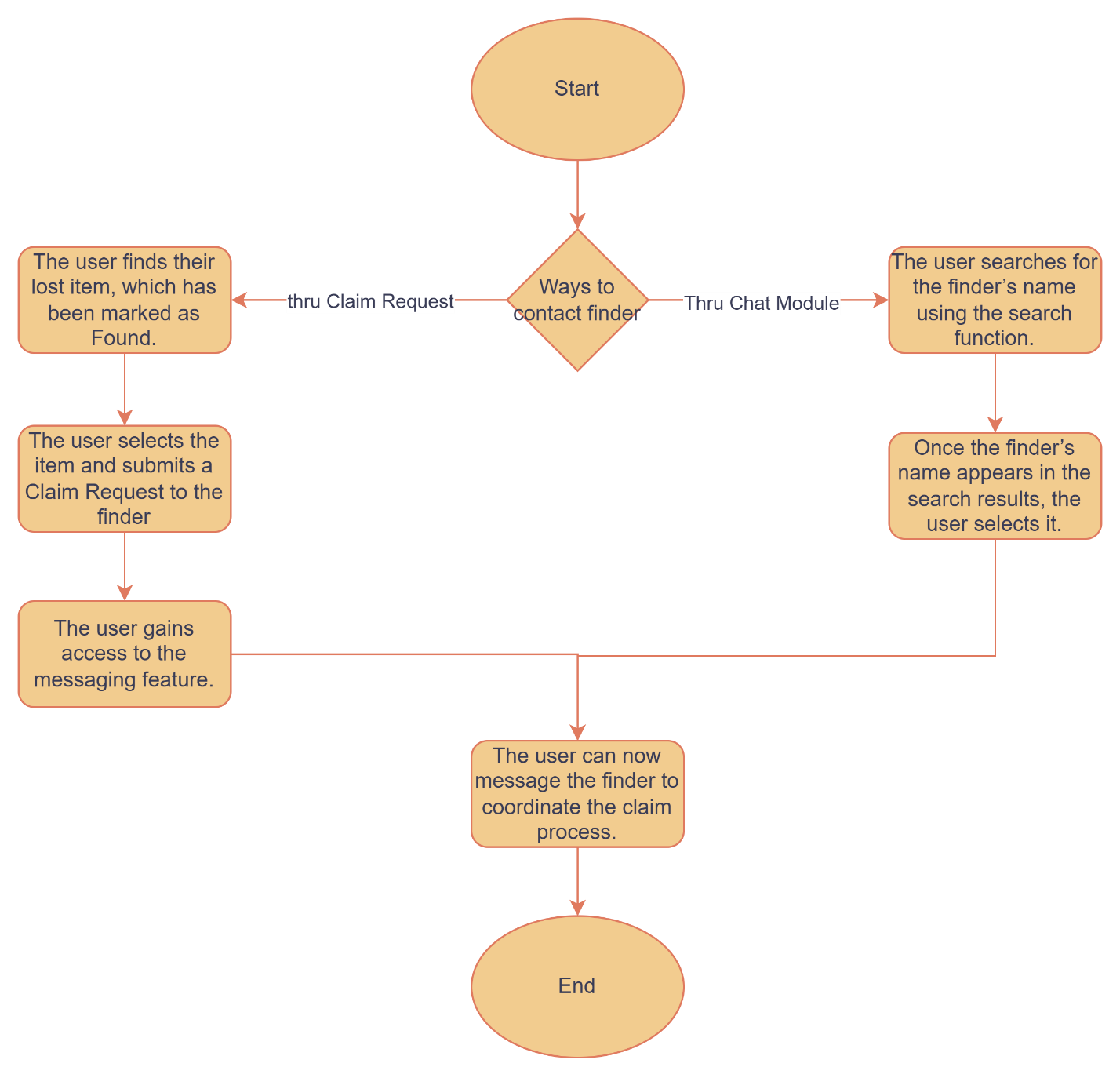
1. **Process Flow: Reporting a Finder**

****Allows users to report a finder if they engage in misconduct after finding an item, such as refusing to process claims or ignoring requests. A report can only be filed if the item is already marked as found and a claim request has been submitted without any action from the finder. Once reported, the item is removed from the public Found Items List and forwarded to the admin for review. The admin can issue a warning (restricting app usage), suspend the finder (revoking access), or resolve the issue by releasing the item and recording the claimant’s details. This process ensures fair handling of lost-and-found transactions while preventing misuse.

Step-by-Step Process for Reporting a Finder

1. Conditions for Reporting a Finder
2. A user can report a finder only if:
   * The item has been found, and the finder is engaging in misconduct (e.g., refusing to process claims, unfair handling).
   * The user has requested a claim for the item, but the finder has not taken any action on the request.
   * The claimant cannot report the finder if they have not submitted a claim request.
3. Filing a report against a Finder
   * The user selects the found item from either:
     + The "Found Items" List (if the user has not yet claimed it).
     + The "My Claimable Items" List (if the user has already requested a claim).
   * The user taps "Report Finder" and provides details about the misconduct.
   * Once submitted, the system:
     + Removes the item from the public Found Items List.
     + Moves the item to the admin panel for further action.
4. Admin Handling the Report
5. The admin reviews the case and has three options for action against the finder:
   * Option 1: Issue a Warning
     + The finder receives an official warning.
     + The finder will have limited app usage, including restrictions on:
     + Filing lost or found items.
     + Processing new claims.
   * Option 2: Suspend the Finder
     + The finder is suspended, losing access to the app entirely.
   * Option 3: Mark as Resolved & Release the Item
     + If the issue is resolved, the admin marks the item as "Resolved".
     + The admin releases the item and files the claimant’s information immediately.

* **Process Flow: Chat Functionality**

****The Messaging Process enables communication between users and finders regarding lost and found items. A user can message a finder in two ways: first, by submitting a claim request for a found item, which then grants access to message the finder directly; and second, by using the Chat Module, where the user can search for the finder’s name and start a conversation. This process ensures seamless communication, allowing users to coordinate claims efficiently and retrieve their lost items.

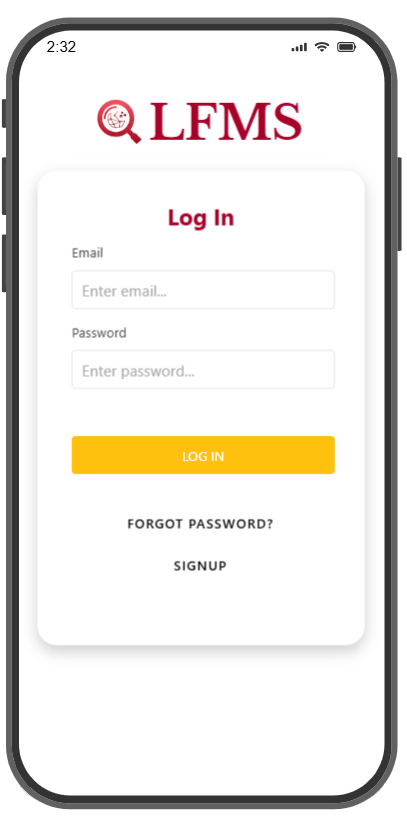
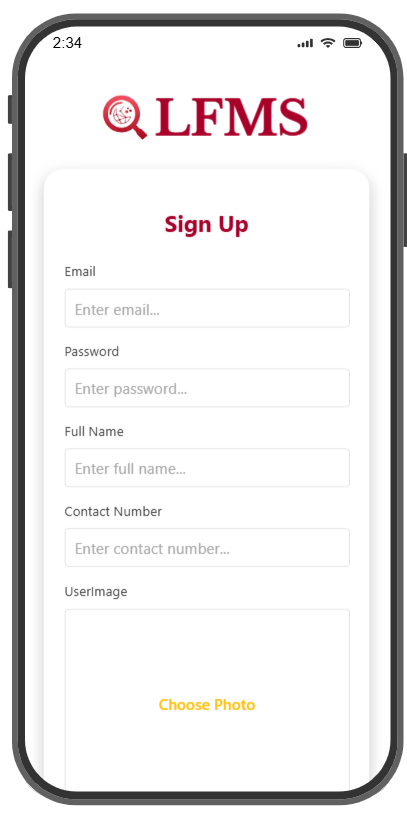
* 1. Ways to contact a Finder.

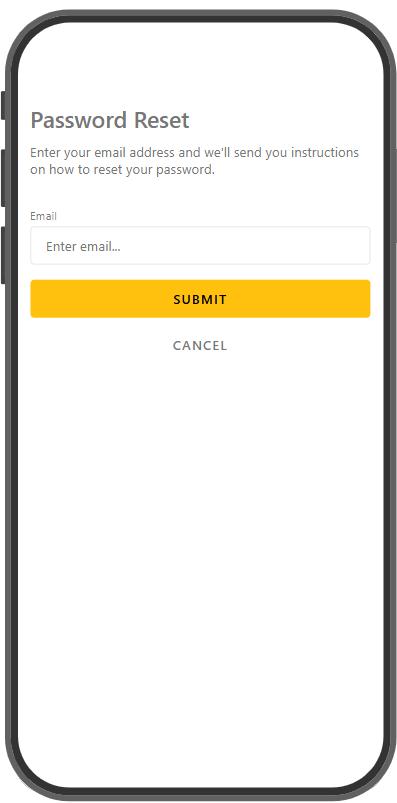
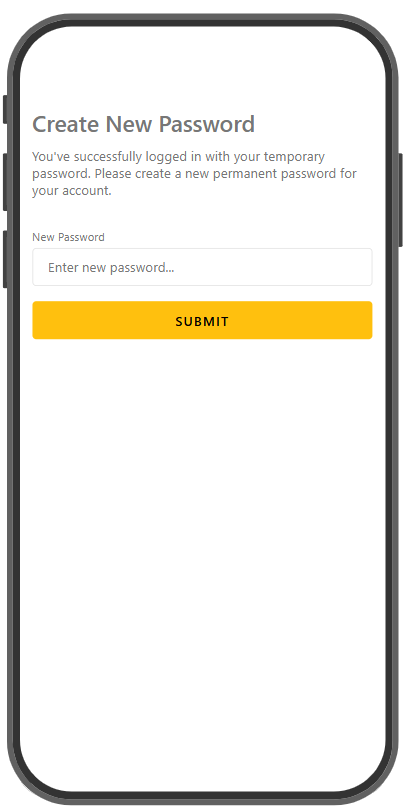
There are 2 ways for users

Message or contact a finder

* 1. Option 1: Messaging thru a claim request
     + The user navigates to the Reported Lost Item List
     + The user find their lost item, which has been reported as found.
     + The user selects the the item and submit a claim request to the finder
       - Once the claim request is successfully submitted, the user gain access to the messaging feature
       - The user can now message the finder to coordinate the claim process.
  2. Option 2: Message through the chat module
     + The user naviates to the Chat module.
     + The user searches for the finder’s name using the search function.
     + Once the finder’s name appears in the search result, the selects it.
     + The user can now star a direct conversation with the finder.
  3. **Interface Design**

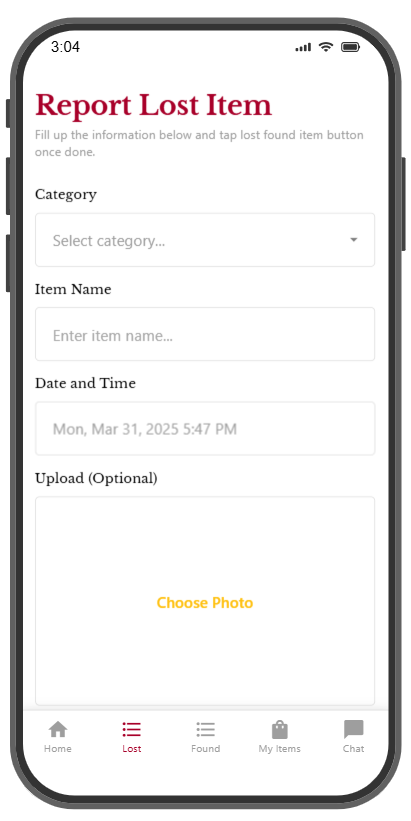
**User Registration and Login**

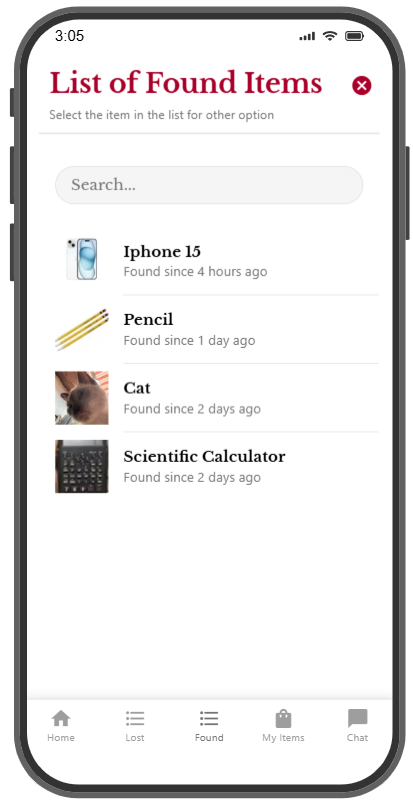
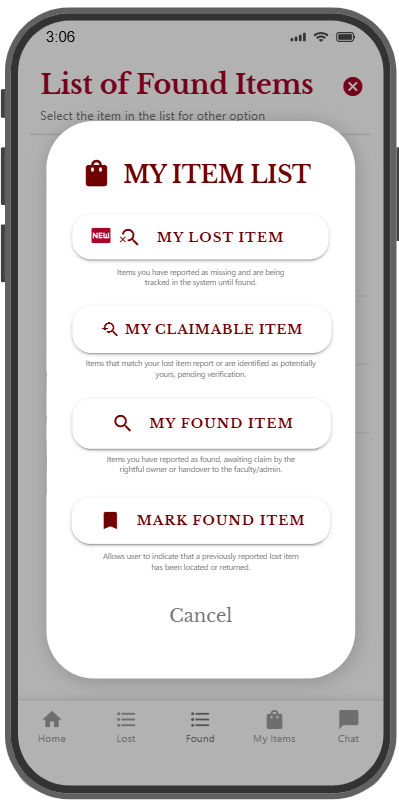
****

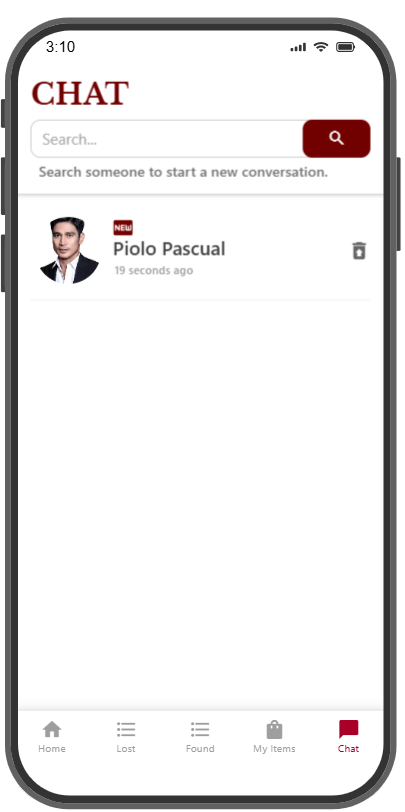
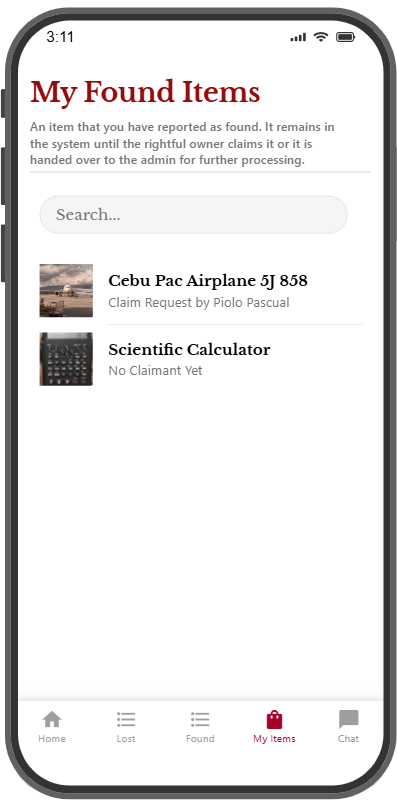
****

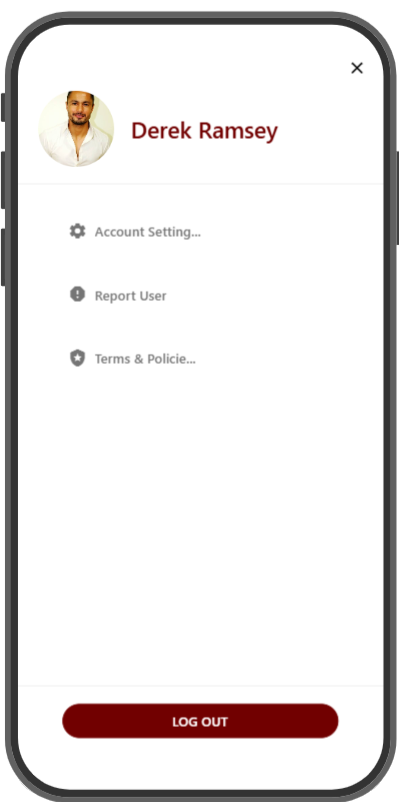
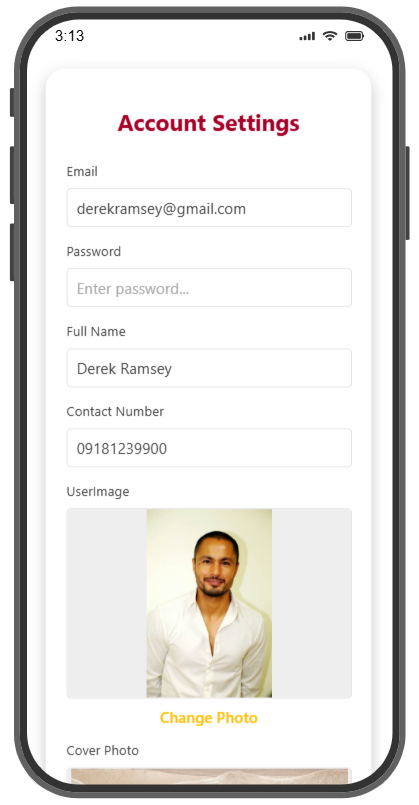
**A screenshot of a cell phone

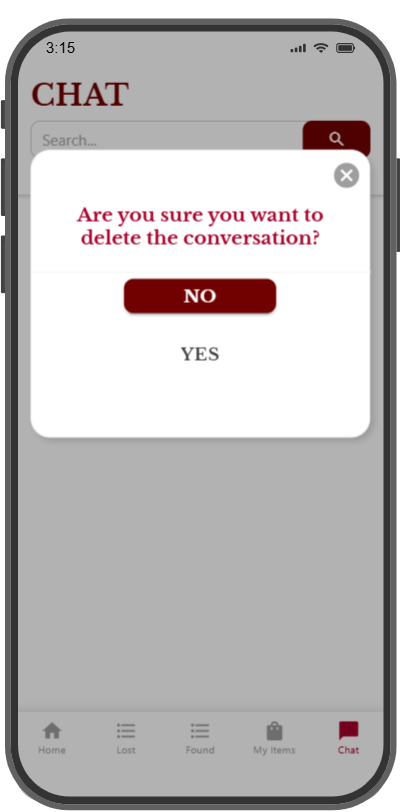
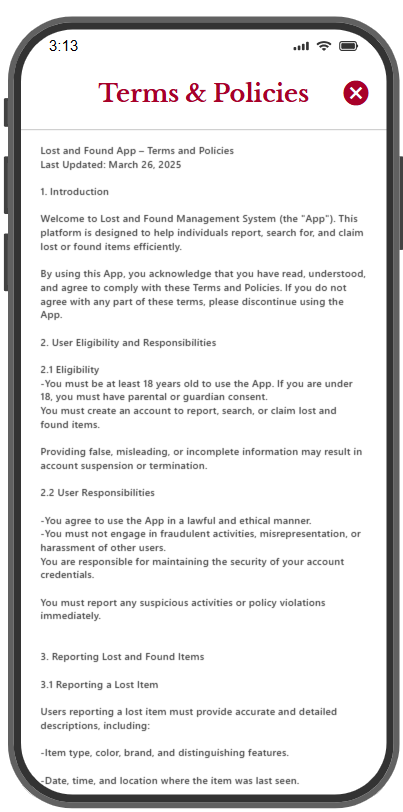
AI-generated content may be incorrect.Dashboard/ Reporting/Listing/Items**

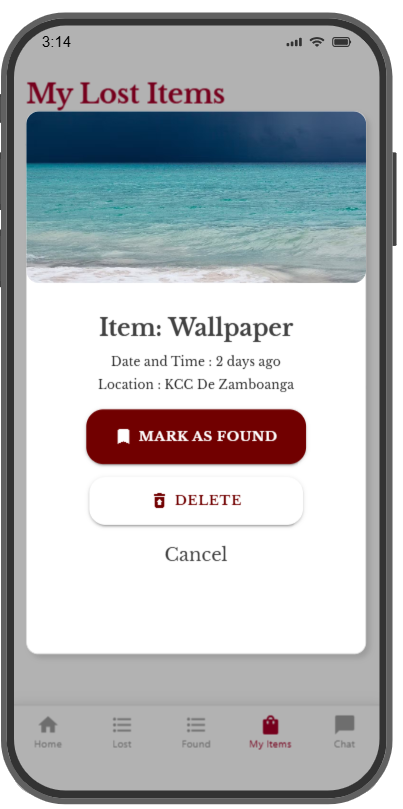
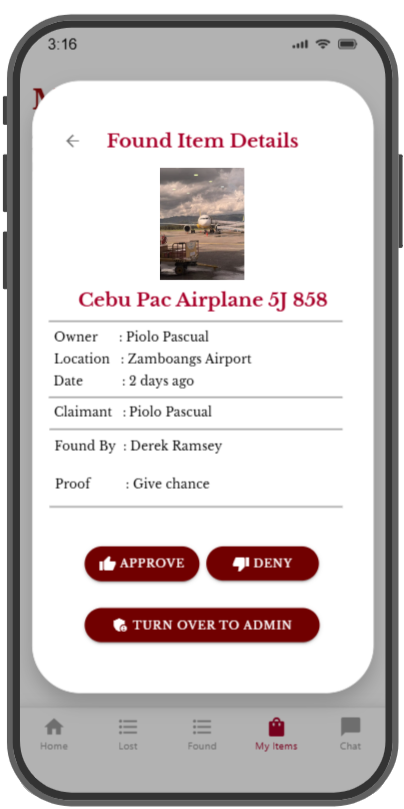
****

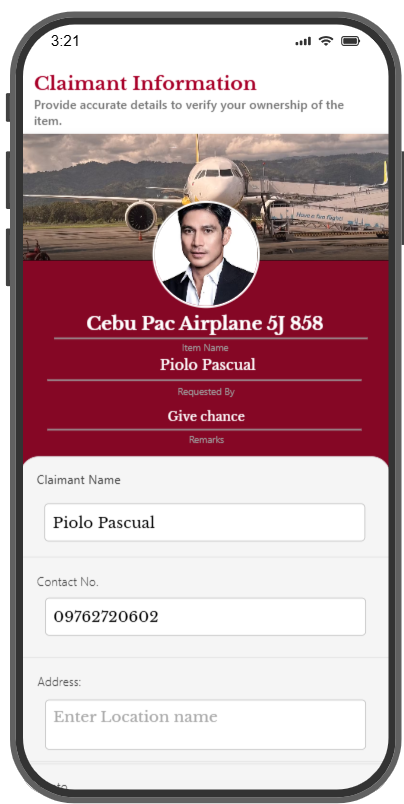
****

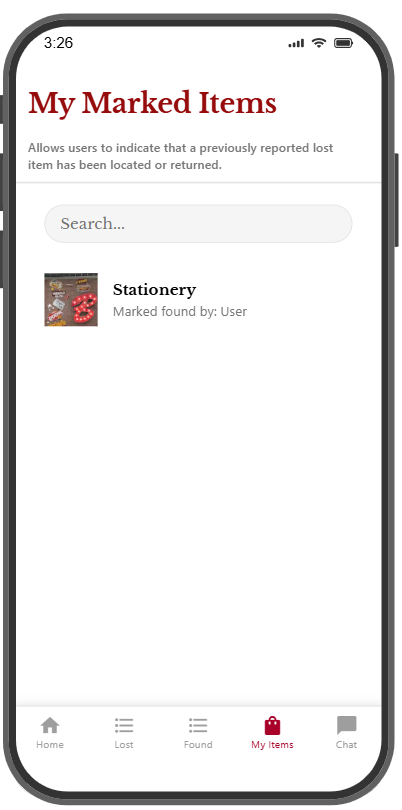
****

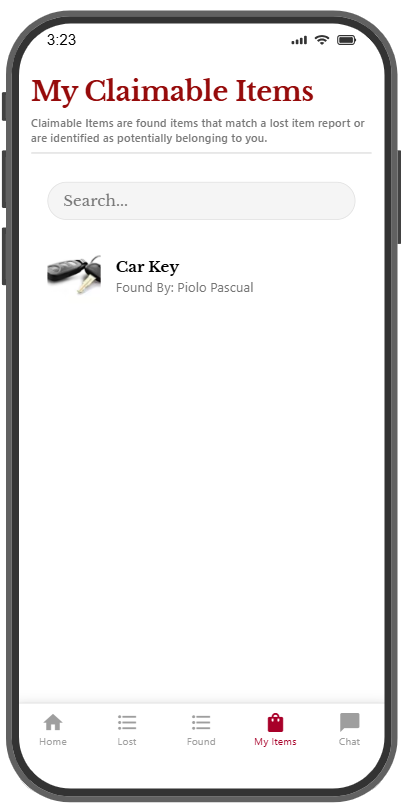
****

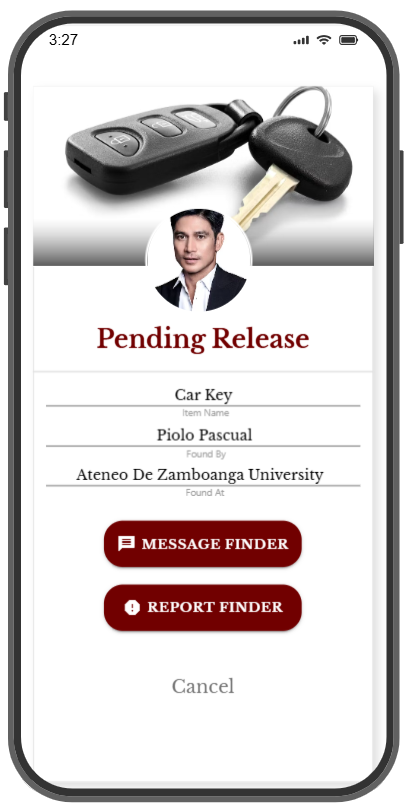
****

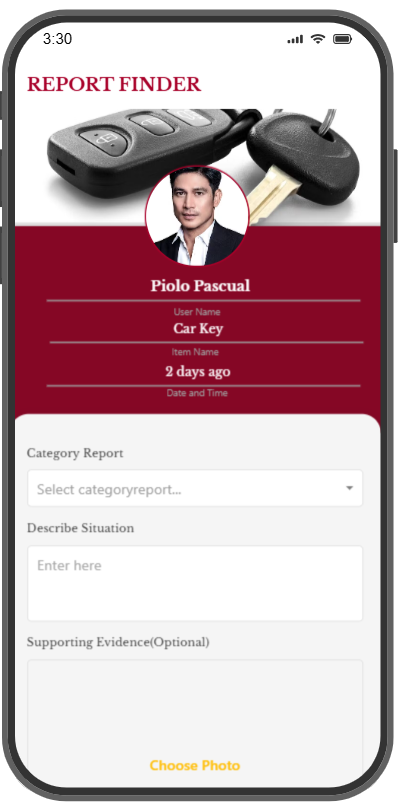
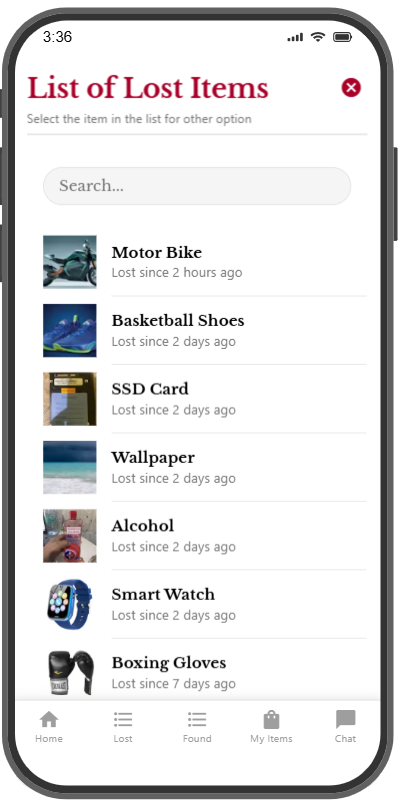
****

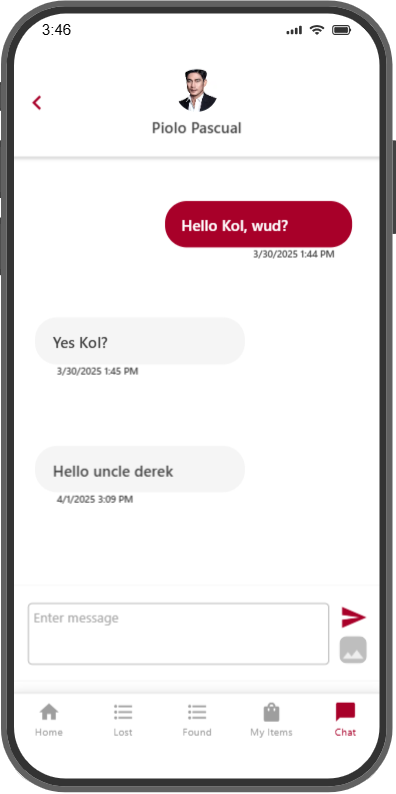
****

****

****

****

****

****

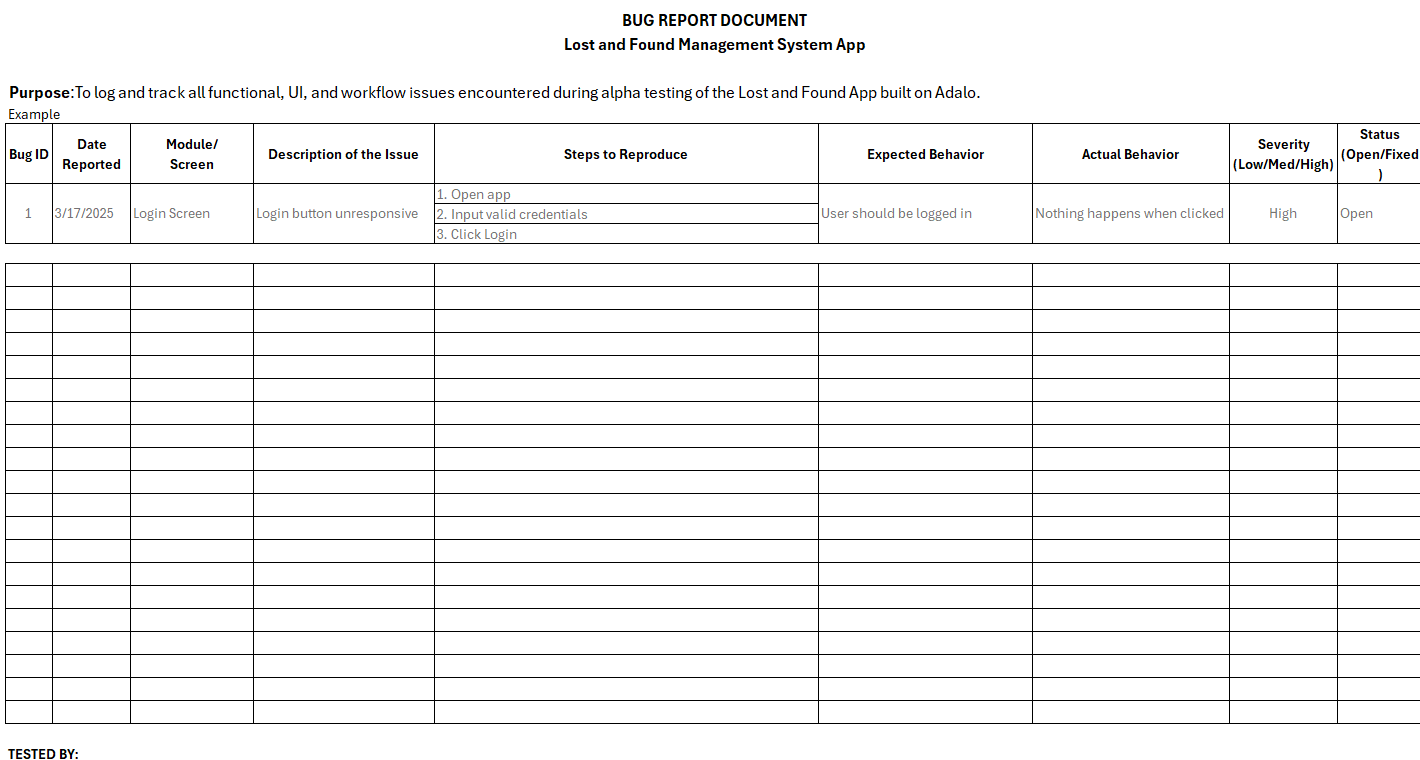
1. **SYSTEM TESTING**

**Alpha Testing**

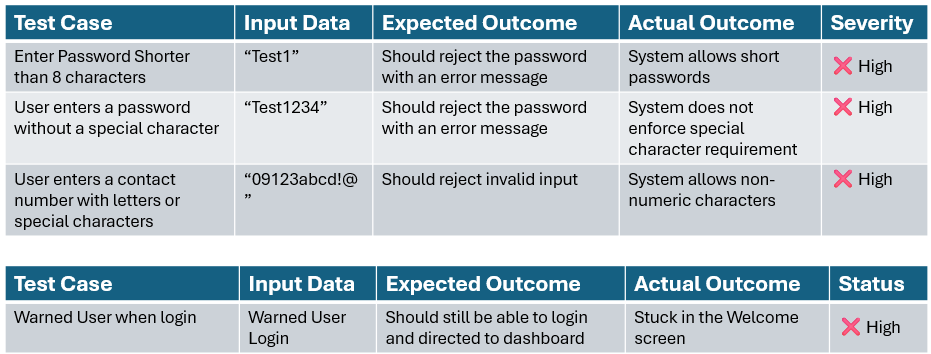
* 1. **Objectives**
* To identify functionality issues, bugs, and usability problems.
* To verify that core features work as expected before moving to beta testing or public release.
* To gather user feedback for further improvements.
  1. **Pre-Testing**
* Selecting testers (2 users ComSci Students, 1 programmer, 1 MISTO head department, 3 employees )
* Setting up test accounts with various roles (regular user, finder, admin, warning, suspended, deactivated).
* Prepare test data (lost/found items, claims, reports).
* Provide Checklist and Bug Report

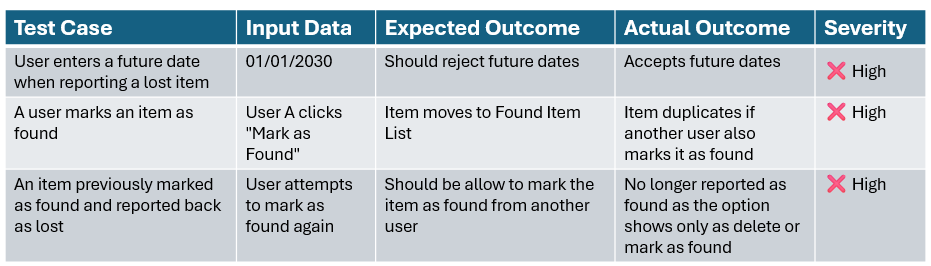
A group of check boxes

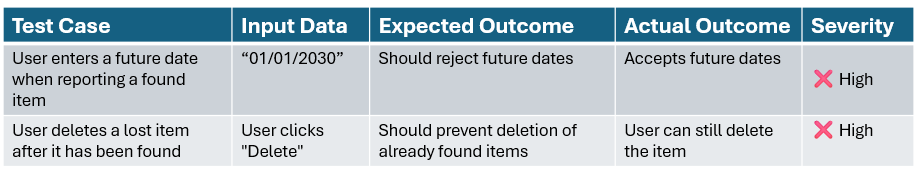
AI-generated content may be incorrect.

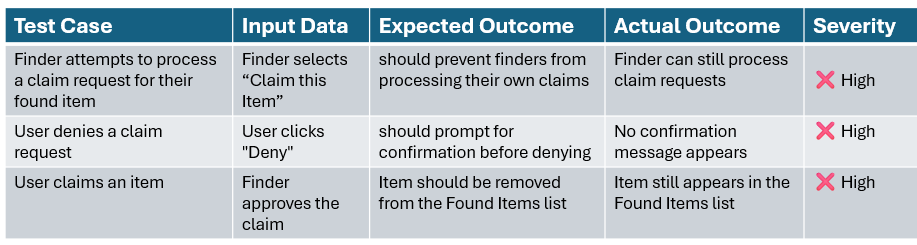


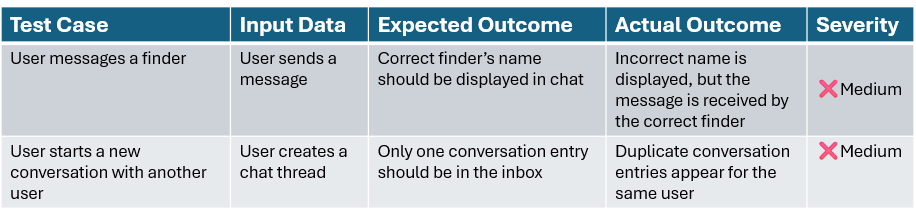
* 1. **During Testing**
* Testers follow the Alpha Testing Checklist for each module.
* Evaluating system performance and usability.
* Log issues in the Bug Report Document.
* Collect feedback on the user interface and experience.
  1. **Post Testing**
* Reviewing all reported bugs and feedback.
* Categorizing issues into critical, major, and minor.
* Prioritizing fixes and planning updates.
  1. **Test Case Result**

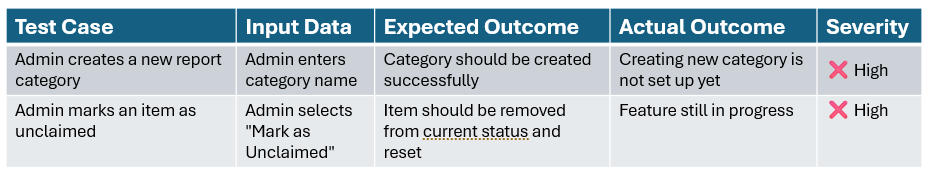












* 1. **Recommendations**
* Implement robust validation for user registration fields.
* Restrict marking an item as found after it has already been processed.
* Ensure that item claims update the Found Items list correctly.
* Add confirmation prompts where necessary to prevent errors.
* Improve chat functionality to prevent duplicate conversations and incorrect user display.
* Complete category management and admin functionalities to enhance system control.
  1. **Performance Observation**
* System experiences delays when handling item status updates.
* Multiple users marking the same item as found leads to duplication and inconsistency.
* Claim processing lacks clear flow and restrictions.
  1. **User Experience**
* Lack of proper validation messages leads to user confusion.
* Missing confirmation prompts can result in unintended actions.
* Inconsistencies in chat functionality disrupt communication between users.