Whozzit?: A School Asset Recovery and Lost Item Reporting Application

In Final Fulfillment of the Academic Requirements for the Course CC106 - Application Development and Emerging Technologies

Madalmingan, Arcelie M.
Flores, Princess Allyn T.
Oncinces, Janmer

Mr. John Patrick Eleria
December 16, 2025

Table of Contents

Executive Summary

Schools often face problems in handling lost and found items. Students lose their ID cards, wallets, gadgets, and other belongings, but there is no fast and organized way to report, track, or claim these items. This results in delays, unclaimed items, and extra work for school staff.

Our proposed solution is a Lost and Found Mobile Application that helps the school manage all lost and found cases in one system. The app allows users to report lost items, report found items, upload photos, search items, match items, and claim them online. It also includes an admin panel where authorized staff can review reports, match items with claims, and update statuses.

The target users of this application are students, teachers, non-teaching staff, and school administrators. Students can report and claim items quickly, while staff can manage the system efficiently through the admin dashboard.

This application is expected to bring several benefits to the school community, such as faster reporting, easier tracking, reduced manual workload, and better organization of lost and found items. It increases transparency, improves communication, and helps return lost belongings to their owners more efficiently. Overall, the app enhances safety, order, and convenience within the school.

Mobile Application Description

Purpose and Objectives

The **Whozzit? Lost and Found Mobile Application** is designed to address the common problem of lost personal items within St. John Paul II College of Davao. Students often misplace valuable items such as ID cards, wallets, gadgets, or notebooks, and school staff face difficulty managing lost-and-found items efficiently.

Objectives:

- Provide a centralized digital platform to report and claim lost items.
- Reduce time spent searching for lost belongings.
- Allow administrators to efficiently manage lost and found items, including verification, approval, and matching.
- Ensure user security through account-based authentication and role-based access (admin and student).
- Enable image-based reporting to clearly identify items.

Key Features

User Features:

- User Registration & Login: Students can create accounts with their school email and upload a profile picture.
- Report Lost Item: Users can report lost items with details such as description, photo, and category.
- Report Found Item: Users can report found items with details such as description, photo, location, and category.
- Item Search: Provides keyword-based searching for both lost and found entries.
- Claim Items: Users can request to claim items they found or lost.
- Status Tracking: Users can monitor the progress of their reports (e.g., Pending, Verified, Claimed).
- Profile Management: Allows users to update their personal information and view their reporting history

Admin Features:

- Admin Login: Admins have a separate login with role verification.
- Admin Dashboard: Allows authorized personnel to oversee all reports.
- Item Verification: Admins may validate reports and ensure accuracy of submissions.
- Item Management – Admins can update item statuses, match lost items with found items, and mark items as claimed.

Unique Tools:

- ImgBB Integration – Users can upload item images via gallery or camera, and the images are stored online.
- Role-Based Access – Ensures that only authorized users can access admin functionalities.
- Edge-to-Edge Splash Screen – Provides a polished app launch experience.

System Features

- Centralized Database: Real-time storage and retrieval using Firebase Firestore.
- Image Upload System: Supports secure photo uploads via the ImgBB API.
- Modular Design: Fragment-based navigation for scalable and maintainable structure.

Technology Used

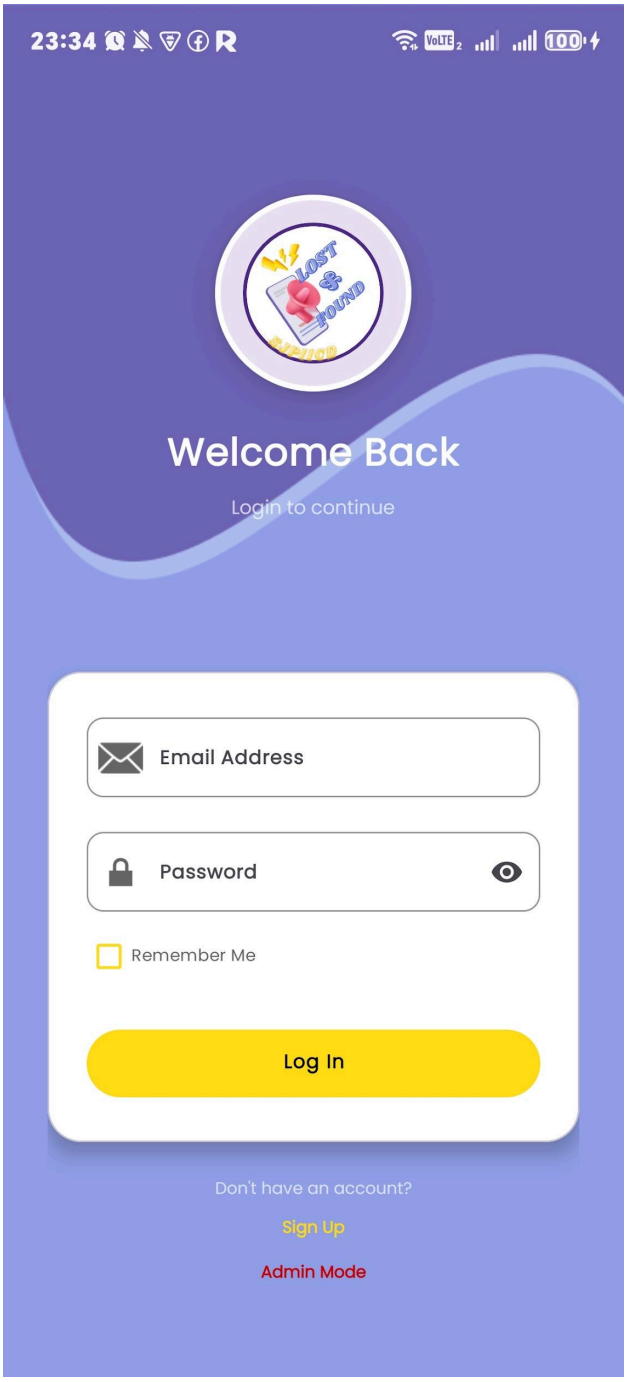
- Programming Language: Kotlin
- Platform: Android Studio (Native Android App)
- Backend & Database: Firebase Firestore for real-time data, Firebase Authentication for user management
- Image Hosting and Upload: ImgBB API for hosting uploaded images
- Prototype and Wireframe Tool: Figma

User Interface (UI)

Student Interface

	This interface is the first screen displayed when the application is launched. It presents the official logo and name of the system, establishing visual identity and branding.

Figure 1. Splash Screen



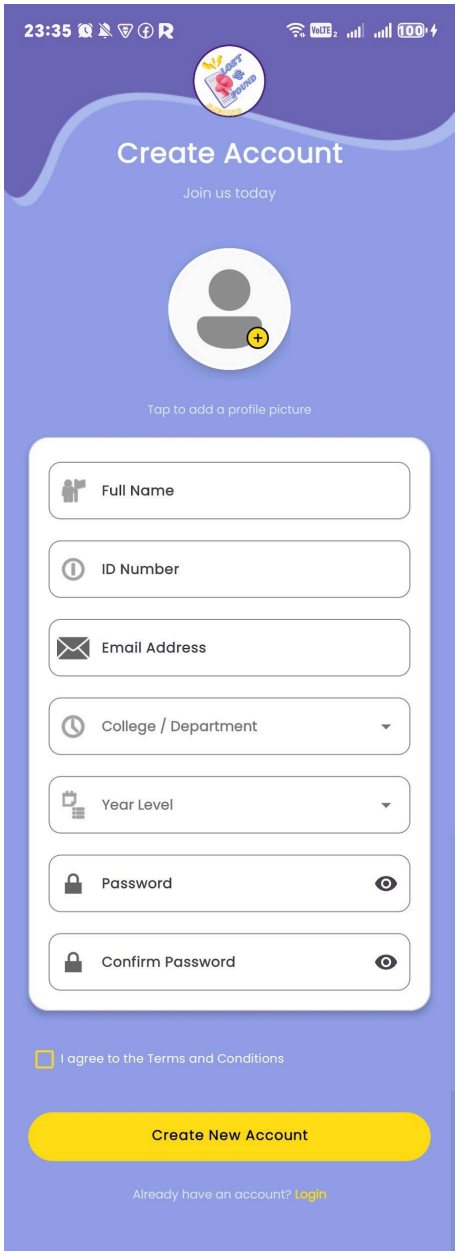
Key Features:

- App logo displayed at the top in a circular card with elevation.
- Welcome message: "Welcome Back" and "Login to continue."
- Input fields for email and password, styled with Material Design TextInputLayouts.
- "Remember Me" checkbox to save login credentials.
- Login button that triggers Firebase authentication with progress indicator.
- Links for "Sign Up" (redirects to registration screen) and "Admin Mode" (opens a login dialog for administrators).

User Flow:

1. User enters email and password.
2. Optional: select "Remember Me"
3. Click "Log In" to authenticate.
4. If login is successful, student or staff is redirected to their dashboard.
5. Clicking "Admin Mode" opens a dialog for admin login with role verification via Firebase.

Figure 2. Login Screen



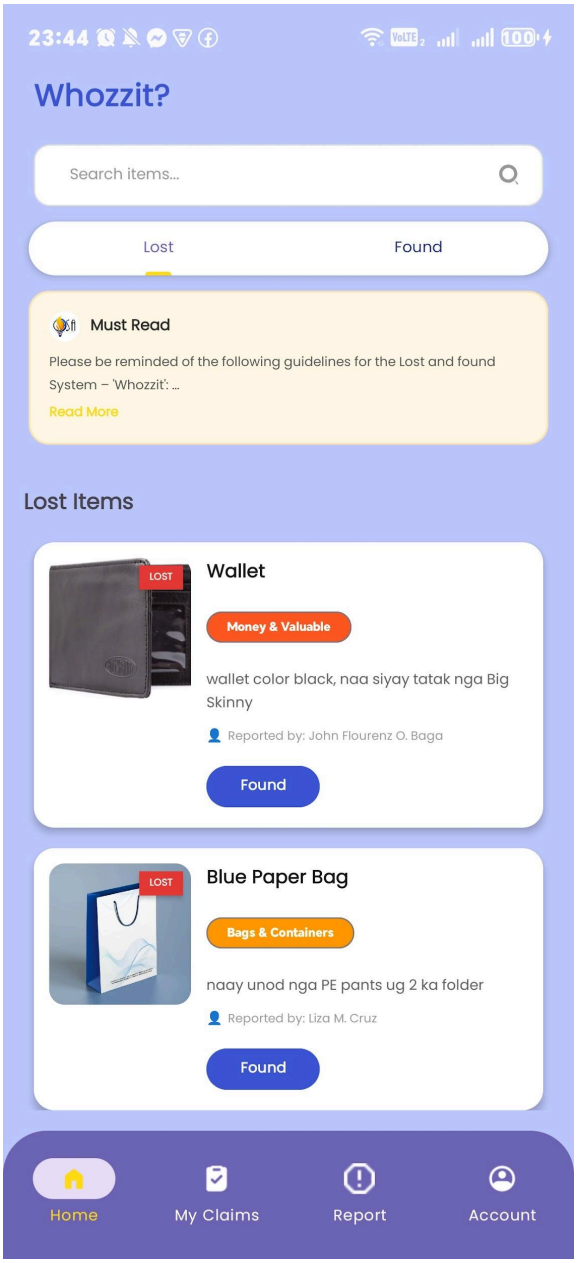
Key Features:

- Profile picture upload (camera or gallery).
- Input fields for Full Name, ID Number, Email, Department, Year Level, Password, Confirm Password.
- Terms and Conditions checkbox.
- "Create New Account" button for registration.
- Link to Login screen for existing users.

User Flow:

1. User fills in all required fields and selects a profile picture.
2. Agree to Terms and Conditions.
3. Click "Create New Account" to register.
4. If successful, user is redirected to the Login screen.

Figure 3. Register Screen



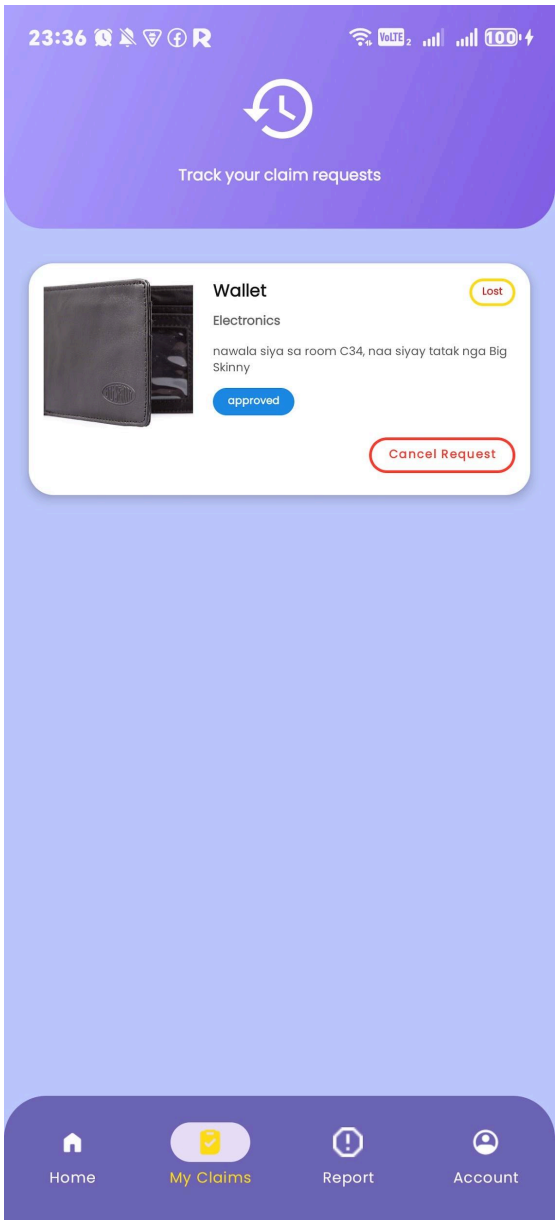
Key Features:

- Displays lists of Lost Items and Found Items
- Search and filter items by name, category, or description
- Buttons to report a lost item or report a found item
- Access to claim found items
- Navigation bar for quick access to other sections

User Flow:

1. Student logs in to the application.
2. The dashboard is displayed as the home screen.
3. Student browses lost or found items.
4. Student may:
   Report a lost item
   Report a found item
   Claim a found item
5. Student navigates to other features using the bottom navigation bar.

Figure 4. Student Dashboard



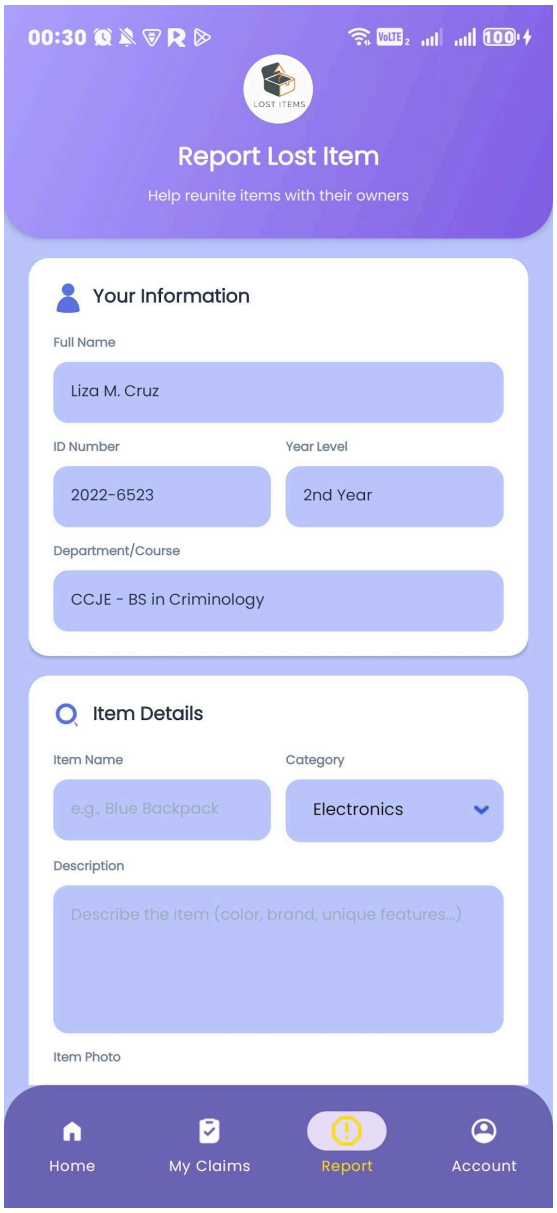
Key Features

- List of claimed items submitted by the student
- Displays item name, category, and claim status
- Shows whether a claim is pending, approved, or rejected
- Organized for easy tracking and reference.

User Flow

1. Student opens the My Claims screen from the dashboard.
2. The system retrieves all claims made by the student.
3. Student views the details and current status of each claim.
4. Updates are reflected automatically once reviewed by the administrator.

Figure 5. Claim Items Screen



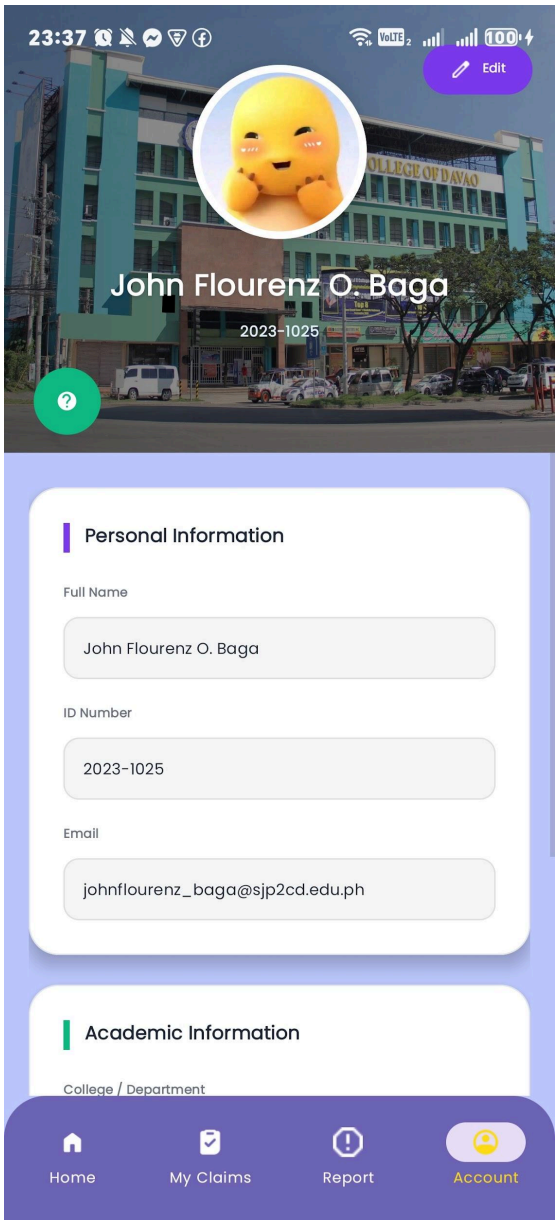
Key Features:

- Item Type selector (Lost or Found)
- Input fields for Item Name, Category, and Description
- Location field (last seen or found location)
- Image upload for better item identification
- Automatic association of the report with the student's account

User Flow:

1. Student opens the Item Report Screen. Selects whether the item is Lost or Found.
2. Enters the required item details and uploads an image.
3. Submits the report.
4. The report is saved and becomes available for matching and review.

Figure 6. Report Screen



Key Features

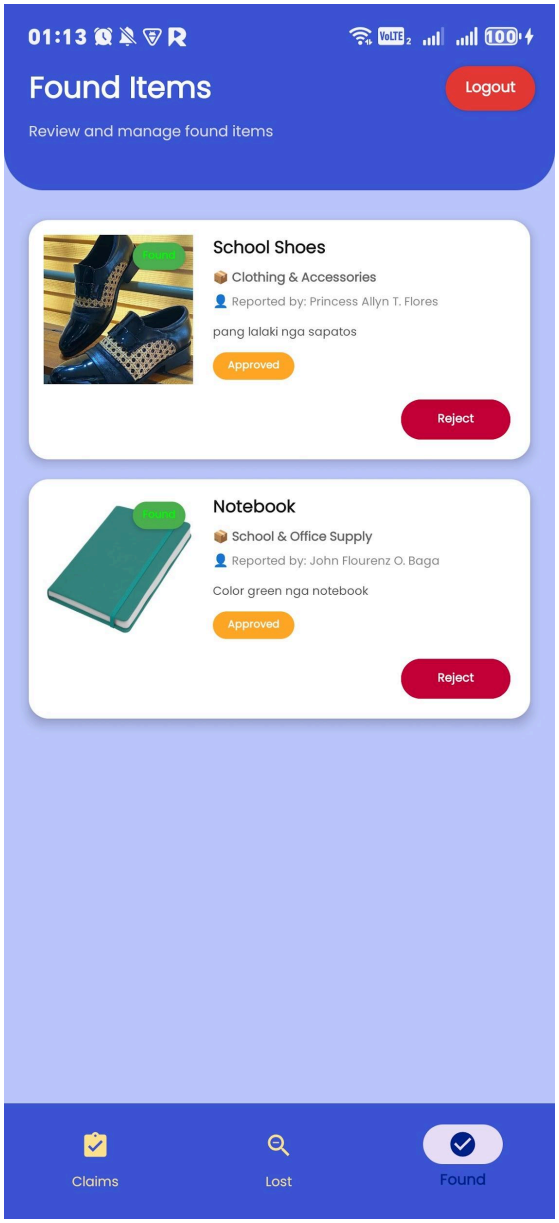
- Displays **student profile information**
- Option to update personal details
- View registered ID number, department, and year level
- Logout button for secure sign-out

User Flow

1. Student opens the Account screen from the dashboard.
2. Profile information is displayed. Student updates details if needed.
3. Student may log out of the application.

Figure 7. Profile Screen

Admin Interface



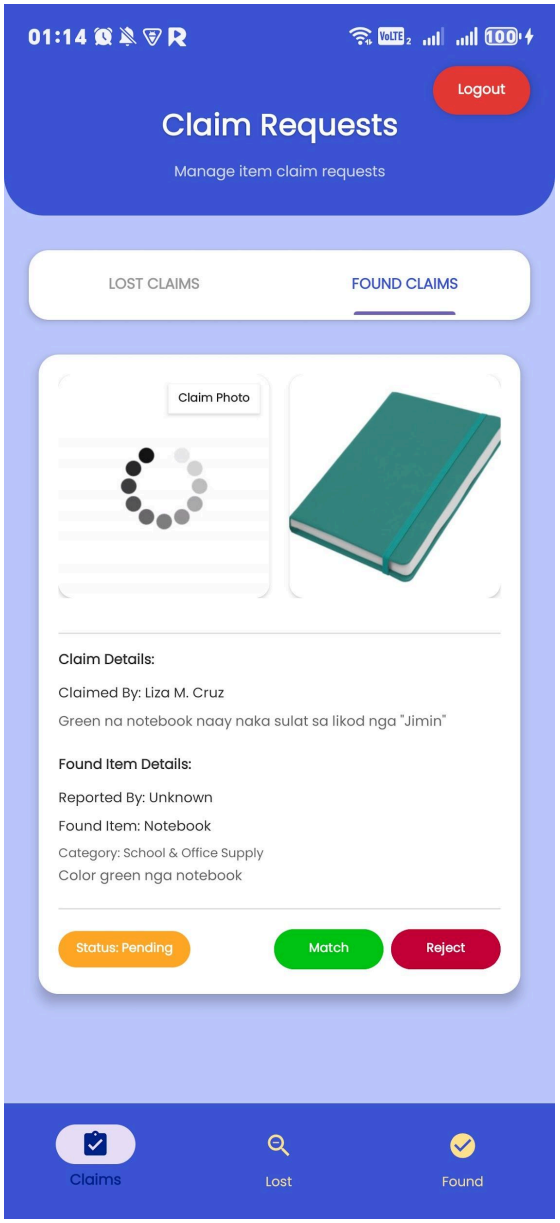
Key Features

- Overview of reported lost and found items
- Access to pending, approved, and matched items
- Navigation to claim verification and item management
- Summary of system activities

User Flow

1. Admin logs in to the system.
2. Admin is redirected to the Dashboard.
3. Admin selects a section to manage (reports of lost and found or claims).
4. Admin performs review or approval actions.

Figure 9. Admin Lost and Found Reports



Key Features

- Displays claimant information
- View uploaded claim photos
- Compare claim details with reported item
- Match or reject claims

User Flow

5. Admin opens the Claims section.
6. Admin reviews claim details and evidence.
7. Admin verifies ownership.
8. Admin matches or rejects the claim.
9. After matching item and students get it personally, admins can set it as claimed.

Figure 10. Admin Admin Claim Verification Screen

Technical Description

- **Authentication:** Users register using school emails (@sjp2cd.edu.ph) and passwords. Firebase Auth handles login and registration securely.
- **Data Management:** Firestore stores user profiles, lost/found reports, and claim records. Each document contains fields like description, date, location, status, imageUrl, and role.
- **Role Verification:** Admin access is restricted using a role field in Firestore. Non-admin users attempting admin login are denied.
- **Image Upload:** Users upload item photos from camera or gallery. Images are converted to Base64 and uploaded to ImgBB. URL is saved in Firestore.
- **Claims Workflow**: Users request claims for items. Admin verifies, approves, or rejects. Firestore status fields update in real time.
- **UI Interaction**: Bottom navigation allows smooth switching between tabs. Alerts and Toasts provide feedback.

Expected Benefits

- **For Students**: Quickly report lost/found items, track status, and retrieve personal belongings with minimal hassle.
- **For Administrators**: Centralized management of lost/found items, reduce manual paperwork, and maintain accurate records.
- **For the School**: Promotes accountability, reduces lost property complaints, reduces manual tracking, and improves campus operations.

SWOT Analysis

Strengths	Weaknesses	Opportunities	Threats
Centralized lost & found system	Requires internet connection	Expand to multiple schools	Users might misuse claims
Role-based access controL	Dependent on users uploading correct info	Add notification system for matched items	Data security risks if Firebase rules misconfigured
Real-time updates via Firestore	Limited to school email accounts	Implement AI for image matching	Server/API downtime may prevent image uploads
Image-based reporting	Image upload may fail on poor connectivity	Add analytics for lost/found trends	Unauthorized access if credentials leaked
User-friendly UI	Admin access restricted to trained personnel		Misuse or false reports

Table 1. E-SWOT Analysis

The Lost and Found App takes advantage of its strengths by providing a centralized, intuitive platform for managing lost and found items within the school. With real-time updates via Firebase and role-based access for administrators, the app ensures that items are reported, tracked, and claimed efficiently and securely. The team can leverage these strengths by promoting the app among students and ensuring that administrators are trained to manage records effectively.

To address weaknesses, the team should ensure reliable internet connectivity and educate users on providing accurate item details and clear images. Potential threats, such as misuse of claims or security vulnerabilities, can be minimized through strict authentication, Firebase security rules, and regular monitoring. Meanwhile, opportunities like expanding the app to other campuses, adding push notifications for matched items, and tracking trends through analytics can enhance its impact. By effectively using strengths and opportunities while mitigating

weaknesses and threats, the app can improve the efficiency and reliability of lost-and-found management in the school.

Value Proposition Canvas

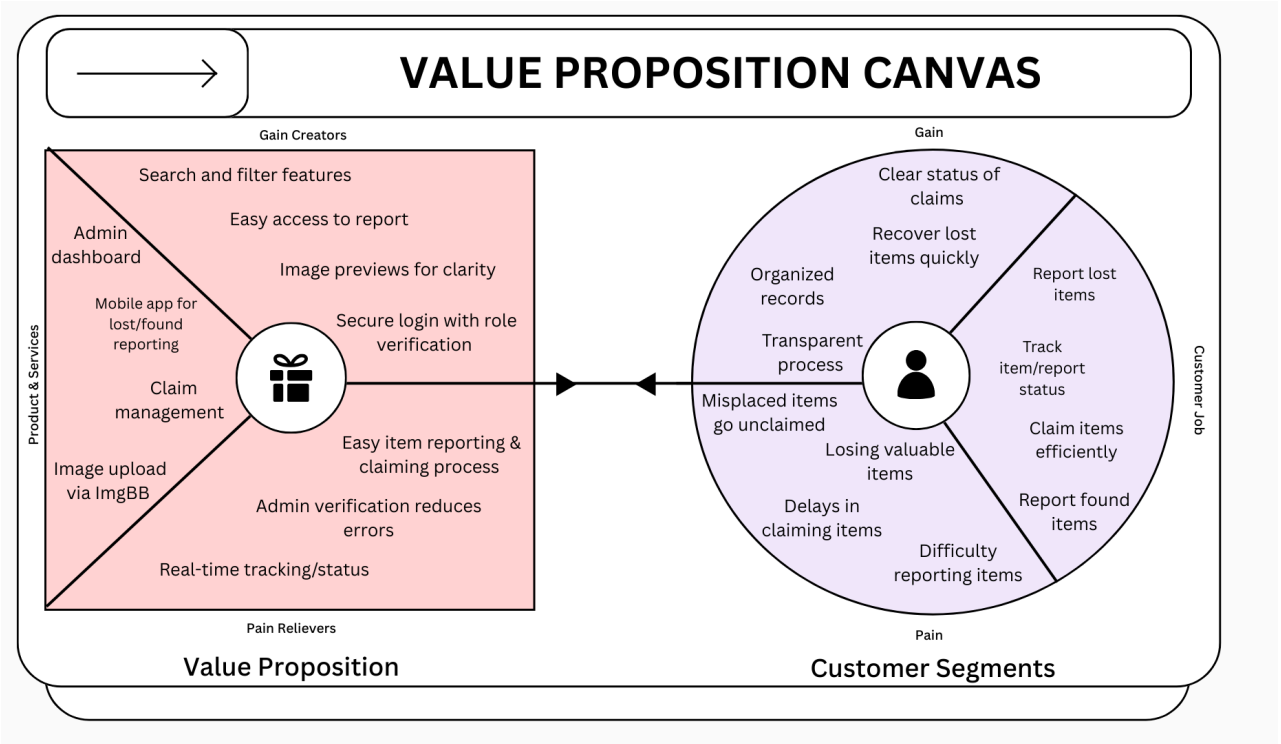


Figure 11. Value Proposition Canvas

The primary users of the Whozzit? app are students who need a fast and organized way to report lost or found items and claim belongings, while school administrators serve as secondary users managing, verifying, and matching these reports. Students often face problems such as losing valuable items like IDs, gadgets, or wallets, encountering delays in claiming items, and dealing with unverified or disorganized reports. These challenges make it difficult to track and recover lost property efficiently. The Whozzit? app provides value by offering a centralized digital platform that allows students to report and claim items with images, descriptions, and categories. Real-time tracking and status updates ensure transparency, while admins can verify, approve, and match items accurately to reduce errors. Additional features such as search and filter functions, notifications for matched items, access to report history, image previews, and secure role-based login enhance usability and convenience. Overall, the app creates a streamlined and reliable system for quickly recovering lost items and keeping the school community informed and organized.

Market Study

Target Market

The primary users of the Whozzit? app are students of St. John Paul II College of Davao who frequently misplace personal items such as ID cards, gadgets, wallets, or notebooks. These students value convenience, speed, and transparency in recovering lost belongings. Secondary users include teachers and non-teaching staff, who may report found items or assist in the claims process. Lastly, school administrators form another user group responsible for managing, verifying, and matching reported items. All users require a secure, reliable, and user-friendly system that allows quick reporting, easy tracking of lost and found items, and seamless communication regarding claims.

Market Description and Potential Challenges

Currently, the school does not have a fully digital or centralized system for lost and found items. Existing practices rely heavily on manual logbooks, bulletin boards, or verbal reports, which often lead to delays, misplacement of items, and unclaimed belongings. Potential challenges for the adoption of the Whozzit? app include resistance from users unfamiliar with mobile applications, technical issues such as poor internet connectivity, and ensuring consistent and accurate data entry from students and staff. Additionally, implementing the system may require initial training and awareness campaigns to encourage user adoption.

Competitors and Competitive Advantage

While there are general lost and found or asset management applications available on the market, very few are tailored specifically for school environments. Existing apps may lack features such as role-based access, student verification, real-time claim tracking, and integrated image upload. The Whozzit? app offers a unique competitive advantage by providing a centralized, secure platform with customized workflows for students and administrators, image-based reporting for clear identification, and real-time updates via Firebase Firestore. Its focus on

school-specific needs, user-friendly interface, and claim verification process make it more suitable and practical for the academic setting compared to generic apps.

Feasibility Study

Technical Feasibility

The Whozzit? app is technically feasible because the development team has the necessary skills and tools to build it. The project uses Kotlin for Android development, Android Studio as the integrated development environment, and Firebase for real-time database and authentication services. Image hosting is handled via ImgBB, ensuring smooth integration with the app. The team is proficient in mobile app programming, database management, and API integration, which ensures that both development and maintenance are manageable with the available resources.

Operational Feasibility

Operationally, the app is designed to be user-friendly and intuitive, allowing students, staff, and administrators to adopt it quickly. Features like simple reporting forms, image uploads, bottom navigation, and real-time status tracking make it easy for users to perform their tasks. Training and guidance are minimal since the app mirrors common mobile interfaces that users are already familiar with. Status updates, and visual cues ensure smooth daily operation without disrupting school routines.

Economic Feasibility

The project is economically feasible because it leverages free or low-cost tools such as Firebase's free tier for authentication and database, and ImgBB's free API for image hosting. Hardware requirements are minimal, as most students and staff already own smartphones. The app reduces costs associated with manual tracking, paperwork, and lost item management, making it cost-effective for the school in the long run.

Schedule Feasibility

The app development is schedule-feasible. With careful planning, modular development using fragments and reusable components allows the team to complete the project within a reasonable timeframe. The use of Firebase accelerates backend setup, while pre-designed UI prototypes from Figma reduce design iterations. This structured approach ensures the app can be developed, tested, and deployed within the academic semester.

Project Implementation Plan

Planning and Requirements Gathering

The team identifies the problem, defines objectives, and gathers requirements from potential users such as students, teachers, and administrators. This phase establishes app functionalities, user roles, and technical constraints to ensure the project meets its goals.

Design and Prototyping

Using tools like Figma, the team creates wireframes and UI prototypes for each screen. This phase focuses on user experience (UX) and interface design, ensuring the app is intuitive and aligns with user expectations.

App Development

The app is built in Android Studio using Kotlin. Backend services are implemented with Firebase Firestore for real-time data management and Firebase Authentication for secure login. Features like image uploads are integrated via ImgBB API, and modular fragment-based architecture is used for maintainable code.

Testing and Debugging

Functional, usability, and performance testing are conducted to identify and fix bugs. User testing ensures the app works as intended for both students and administrators, and that features like claim submission, item reporting, and real-time updates function smoothly.

Deployment (School Use or Demo)

For deployment, the application launched as a demo version intended for academic presentation. This allows the team to showcase its features and functionality without full-scale implementation, providing a practical demonstration of the system’s capabilities to users and evaluators

Feedback and Improvement

After deployment, feedback is gathered from our teacher to identify areas for enhancement. Updates are then implemented to improve user experience, performance, and feature reliability, ensuring the app meets academic requirements and demonstrates its intended functionality.

Timeline

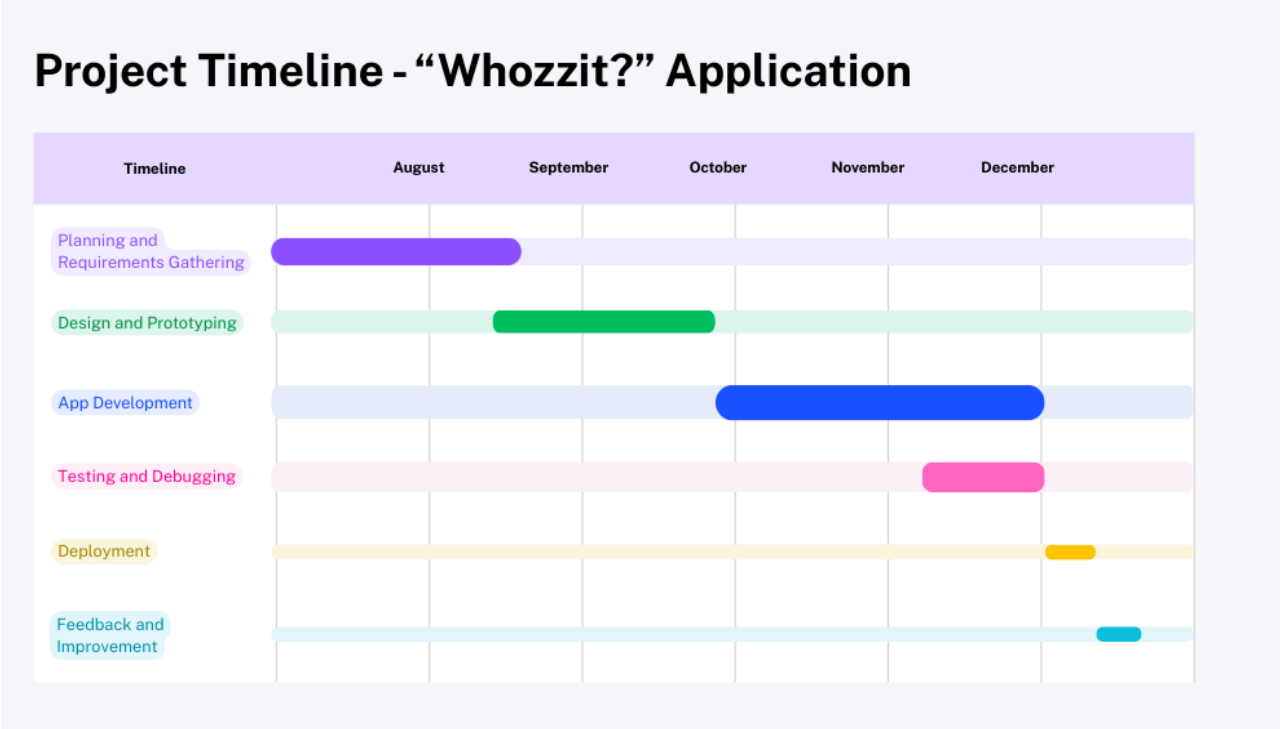


Figure 12. Gantt Chart

Team Roles and Responsibilities

Janmer J. Oncines

TESTER

Arcelie M. Madalmingan

TESTER

Princess Allyn T. Flores

PROGRAMMER/DESIGNER/RESEARCHER

Cost-Benefit Analysis

Estimated Costs

- Development Tools and Resources: Android Studio, Kotlin libraries, Firebase services.
- Hosting and Maintenance: Firebase Firestore and Authentication, ImgBB API usage.
- Training/Promotion (if applicable): Orienting students and staff on app usage.
- Time and Manpower: Labor equivalent of development and testing hours.

Expected Benefits

- Time Savings and Efficiency Improvements: Reduces manual tracking of lost and found items.
- Easier Data Management and Communication: Centralized database with real-time updates.
- Positive User Experience: Simplifies reporting, claiming, and tracking items for students, staff, and admin.
- Potential Long-Term School Benefits: Supports digitalization and sustainability initiatives.

Category	**Description**	**Estimated Value (₱)**
Development Costs	Tools, hosting, testing	₱10,000
Labor Equivalent	Team effort (100 hours)	₱20,000
Total Cost		₱30,000
Expected Benefits	Improved operations, time saved	₱50,000
Net Benefit	Benefits – Costs	₱20,000

The project requires development tools (Android Studio, Firebase, ImgBB API) and labor for coding, testing, and debugging. These costs are estimated at

₱30,000. The expected benefits include significant time savings for students and staff, improved organization, and digital record management, valued at ₱50,000. Therefore, the net benefit is ₱20,000, demonstrating that the project is economically viable and beneficial for the school community.

References