PUCIT
Punjab University College of Information Technology

LOST2FIND App

Group Leader:

BSEF20A049   Laiba Liaquat

Group Member:

BSEF20A020   Noor Fatima

Supervised By:

Sir Muhammad Nadeem

First Deliverable

Version 1.0

TABLE OF CONTENTS

1 Introduction

In our daily life people face many problems related to finding their lost valuables. The application is intended for the person who has lost his valuables and the one who has found those valuables. Both just have to visit the app and just have to make a profile. The profile will consist of different required fields such as name, address, location, contact number and email id and these profiles will be visible to all the visitors of portal. On other hand everybody can see the ads of lost goods of users & can easily contact them through the contact number and email provided by users.

The main idea of our project is to provide service to people who are struggling with finding their lost valuables problems by building android based application. In this project, we are willing to make a system based on the concept of user and second user. User do effort to return someone's valuable and he can also make some money if there is reward included in the deal. In this system there will be one person will be doing efforts like an employee and he will be called **valuable returner** and the other person will use his service to get his valuables back and that person will be named as **user**. Only a single android application interfaces will be used for both parties just. A database will be used to save communication history between both parties for security purpose.

This deliverable is all about planning and scheduling of our project. So it contains the following artifacts:

a. Project Feasibility Report
b. Project Scope
c. Project Costing
d. Critical Path Method Analysis (CPM Analysis)
e. Introduction to team members & their skills set
f. Tools and Technologies
g. Vision Document
h. Risk List

1.1 Project/Product Feasibility Report

Feasibility of a project is to assess practicality of that project. Our project feasibility report will completely describe the all the possible concerns of feasibility of project with all aspects. Develop small components and then combine them.

There are many types of feasibilities:

- Technical
- Operational
- Economic
- Schedule
- Specification
- Information
- Motivational
- Legal and Ethical

1.1.1 Technical Feasibility

- **Technology Stack and Resources**:
  The project will use Flutter for the mobile application development, HTML, and CSS for the user interface design, and Firebase for data storage. A team of experienced developers with knowledge of these technologies will work on the project.
- **Challenges:**
  Integration of location-based services: Ensuring the precise location of lost items and facilitating user interaction based on geographical proximity.
  Database Security: Implementing robust security measures to protect user data and ensure user privacy.

1.1.2 Operational Feasibility

- **User Interaction:**
  Users can create profiles, post lost item advertisements, and edit their listings. A chat or messaging system is in place to facilitate communication between users who have lost and found items.
- **User Management:**
  The admin panel will have the authority to manage user accounts and moderate content. Content moderation is necessary to prevent misuse of the platform, ensuring it remains reliable and trustworthy.

1.1.3 Economic Feasibility

The project is economically feasible, as it requires an android smartphone that is available to everyone now days. There is no need of expensive hardware for this project, so its development and operation both are feasible economically.

- **The Cost-Benefit Analysis:**
  The project involves development costs, server maintenance, and operational expenses. Revenue will be generated through premium features (e.g., promoted listings) and advertisements. Costs have been estimated, and potential revenue has been projected based on user adoption rates.

1.1.4 Schedule Feasibility

- **Special Project Timeline:**
  The project is divided into development, testing, and deployment phases. Special attention will be paid in order to complete the project on time. All the project activities have been analyzed and timed, considering real world scenario. There will be no delay in project schedule.

1.1.5 Specification Feasibility

- **Our User Experience:**

The project specifications are designed to provide an intuitive and user-friendly experience. Key features include a search and filter system, chat functionality, and an attractive user interface. We'll ensure that the app meets quality standards.

1.1.6 Information Feasibility

Our application will gain the user satisfaction as we are very much clear about the user requirements. We can add new functionalities with the passage of time, according to demand of users. Each team member is supposed to do the work within given deadline.

- **Data Sources:**

Data sources include user profiles (name, contact information) and lost item listings (description, location, photos). The admin panel will manage this data for security and privacy.

- **Data Privacy:**

Security measures will be implemented to protect user information. Personal data will be encrypted and stored securely.

- **Data Availability:**

The project ensures that sufficient information is available for users to post lost items and facilitate item recovery.

1.1.7 Motivational Feasibility

. The gathered information is sufficient, reliable and meaningful to develop this application with all of its functionalities.

- **Alignment with Motivation:**

Lost2Find app aligns with the motivation to help people retrieve their lost valuables. The project's core mission is to connect individuals and foster a sense of community and assistance.

- **Benefits:**

The project offers both societal and individual benefits. Users will be able to recover lost items, reducing personal distress and inconvenience.

1.1.8 Legal & Ethical Feasibility

- **Issue Handling:**

A plan is in place to address potential legal or ethical issues, such as disputes between users or misuse of the platform. An efficient moderation system will be implemented to ensure adherence to ethical guidelines.

1.2 Project Scope:

There is no dedicated app to find your lost items in Pakistan, so it has very scope to all over the Pakistan because the people have never seen that type of service before, it will be new to everyone. Before the app, the people use to go to the locations where he is guessing that there he could find his lost item and if he fails to find that than it will be considered lost there will be no chance of finding it back.

1. The Lost2Find project aims to address the common issue of lost and found items by providing a platform that connects people who have lost their valuables with those who have found them. It offers a user-friendly mobile application and website to facilitate the process of reporting lost items, searching for found items,

2. Create a user-friendly mobile Application Lost2Find. Provide a platform for users to report lost items with detailed descriptions, location information, and rewards (if applicable). Enable users who have found lost items to post information about the items they found. Implement a communication system that allows users to connect with each other and arrange for the return of lost items. Enhance the efficiency of reuniting lost items with their owners and promote a sense of community and helpfulness.

1.3 Project Costing

Cost is one of the most important factors for any project. There is always a tradeoff present between cost, quality and time. We are using Function Point Analysis for cost estimation. In this process, costs are analyzed using mathematical formulas linking costs or inputs with metrics to produce an estimated output.

1.3.1 Function Point Analysis

We will use a technique in which we will estimate the cost of our project with the help of functionalities of our project or application. With directly measured techniques, we will measure our functionalities indirectly. For this technique, there are five major things which are:

- External Input
- External Output
- External Inquiries
- External Interface Files
- Internal Logical Files

First calculate functional counts

EQ (External Inquiries):
1. Search for Lost Items:
         -This feature allows users to search for lost items based on keywords, categories, or location.

2. View Item Details:
   - Users can inquire about the details of a specific lost item to learn more about it.

3. Filtering Lost and Found Items:
   - Users can apply filters to narrow down their search results, such as filtering by date, item category, or proximity.

4. View Item Matches:

     - After searching for a lost item, users can inquire about potential matches, i.e., items found by others that match their description.

5. Sort Search Results:

     - Users can sort search results based on criteria like relevance, date found, or distance.

6. Get Directions to Found Item:

     - If a user finds a match for their lost item, they can inquire about directions to the location where the item was found.

7. Contact the Finder:

     - Users can inquire about the contact information of the person who found their lost item.

8. View Item History:

     - Users can inquire about the history of a lost item, including when and where it was found, and any interactions related to it.

9. Request Additional Information:

     - Users can inquire about additional information related to a found item, such as its condition or any other details.

10. Receive Alerts for Matches:

     - The app sends notifications or alerts to users when it identifies potential matches for their lost items.

EI (External Input):

1. Report lost item:

     -This is a fundamental EI feature where users input information about a lost item.
     They provide details such as a description, category, and location of the lost item.
     - The system receives this input and processes it to record the lost item in the database.

2. Add Item Photo:

     - Users can upload photos of the lost item to include with their report.
     - The app receives the photo as external input, and it associates it with the lost item's record.

3. Reward Offer:

     - Users have the option to offer a reward for the return of their lost item. This is an input where they specify the reward amount and terms.
  - The system takes this input and stores it with the lost item report.

4. User Registration:

     - When users register on the app, they provide personal information, such as their name, email, and contact details.

     - This external input helps create user profiles in the system.

5. Filter Lost and Found Items:

     - Users can input filter criteria, such as date, category, or location, to refine their search for lost or found items.

     - The system takes these inputs and retrieves matching records from the database.

6. Sort Search Results:

     - Users can select how they want to sort their search results, choosing options like relevance, date found, or distance.

     - The app processes this user input to reorganize the search results accordingly.

7. Contact the Finder:

     - When a user wants to get in touch with the person who found their lost item, they can send a message or inquire about the item.

     - This input is handled by the system to facilitate communication between the parties.

8. Request Additional Information:

     - Users can request more details about a found item, such as its condition or any additional notes provided by the finder.

     - The system takes these input requests and provides the relevant information to the user.

9. Report Inappropriate Content:

     - If users encounter inappropriate or spam content, they can report it to the app administrators.

     - The system processes this input to review and potentially remove offensive content.

10. Send Notifications:

     - The app can send various notifications to users, such as alerts about potential matches for their lost items.

     - User interactions with these notifications, like opening or dismissing them, are considered external inputs.

11. User Feedback and Support Requests:

     - Users can submit feedback or support requests through the app, providing details about their issues or suggestions.

12. In-App Messaging:

     - When users send and receive messages within the app to coordinate the return of lost items, these messages are considered external inputs.

EO (External Output):

1. Match Found Item Notifications:

   - When a potential match for a user's lost item is found, the app sends an external output notification to the user, informing them of the discovery.

2. Search Results Display:

   - After a user conducts a search for lost items, the app displays the results, presenting relevant information about found items.
   - The app generates this external output to inform the user of potential matches.

3. Item Details Display:

   - When a user clicks on a specific lost or found item in the app, the app displays detailed information about the item, including its description, location, and any associated photos.
   - This is an external output since the app provides information to the user based on their input (selection).

4. Item History View:

   - Users can inquire about the history of a lost item, and the app displays a history log that includes when and where the item was found, and any related interactions.

5. Sorted Search Results:

   - After a user selects a sorting criterion (e.g., relevance, date found, distance) the app reorganizes and displays the search results accordingly.

6. Contact Information Display:

   - When a user requests contact information for the person who found their lost item, the app displays the relevant contact details.

7. Additional Information Presentation:

   - When users request more details about a found item, such as its condition or notes from the finder, the app displays this information.

8. User Registration Confirmation:

   - After a user successfully registers, the app provides a confirmation message or notification to acknowledge the registration process.

9. User Feedback Confirmation:

   - When users submit feedback or support requests, the app sends confirmation messages acknowledging the receipt of the user's input.

10. In-App Messaging Display:

   - When users exchange messages within the app for coordination in returning lost items, the app displays these messages to facilitate communication.

---

11. Alerts on App Updates:

      - The app may send notifications to users regarding app updates, new features, or important announcements.

12. Reporting Outcome Notifications:

      - After a user reports inappropriate content or behavior, the app provides notifications or messages indicating the outcome of the report, such as content removal.

ILF (Internal Logical File):

1. User Profiles:

      - The app stores user profiles containing information like usernames, contact details, and settings. This data allows users to manage their accounts and preferences.

2. Lost Item Records:

      - The app maintains a database of lost items reported by users. These records include information such as item descriptions, categories, locations, dates of loss, and photos. This data is essential for matching lost items with found ones.

3. Found Item Records:

      - Similar to lost item records, the app manages a database of found items. These records contain descriptions, categories, locations, and dates of discovery, photos, and any associated notes from the finders.

4. Reward Listings:

      - If users offer rewards for returning lost items, the app keeps a record of these reward listings. This includes details about the reward amount, terms, and which lost item it's associated with.

5. Messaging Conversations:

      - The app maintains records of messaging conversations between users. These records include message content, timestamps, and details about the participants. Storing this data enables communication between users within the app.

6. Search Result Listings:

      - The app generates and temporarily stores search result listings based on user queries. These listings include references to lost and found items that match the search criteria.

7. Alerts and Notification Records:

      - The app records alerts and notifications sent to users. This includes information about the type of notification, content, recipient, and timestamps for managing and displaying notifications.

8. Feedback and Support Requests:

- Users can submit feedback, support requests, or reports regarding inappropriate content. The app maintains records of these requests, including user-provided information and system responses.

9. Item History Logs:
- For each lost and found item, the app logs a history of interactions and events related to that item. This may include timestamps, actions taken (e.g., reported, matched, returned), and user details.

10. Admin and Moderation Records:
- If the app employs administrators or moderators to oversee content and user behavior, internal records related to these roles and their activities are maintained. This helps manage app security and user behavior.

EIF (External Interface File):
1. External Location Data:
  - If our app relies on external sources for location data, such as maps, GPS, or geocoding services, this data would be considered an ELF. It may include geographical coordinates, place names, addresses, and maps.

2. Public Transportation Schedules:
  - If app helps users locate lost items on public transportation, it may need to interface with external public transportation data sources to provide schedules, routes, and service information.

3. Crowdsourced Lost and Found Databases:
  - If app connects to external lost and found databases that are maintained by other organizations or communities, these databases with their lost and found item information would be considered ELFs.

4. Legal and Regulatory Databases:
  - To ensure your app complies with local laws and regulations related to lost and find items, you might need to interface with external legal or regulatory databases that provide guidance and information.

5. Social Media Platforms:
  - If app integrates with social media platforms for sharing lost and found posts or user authentication, the user profiles and social media posts managed by these platforms would be ELFs.

6. Payment Gateway Services:
 - If app facilitates reward payments for returned lost items through external payment gateways like PayPal, the payment data and transactions handled by these gateways would be considered ELFs.

7. Image Recognition Libraries:

- If app uses external image recognition libraries, such as TensorFlow, to help identify and categorize images, the image recognition databases and services provided by these libraries would be ELFs.

8. Language Translation Services:
- If app offers multilingual support and uses external language translation services like Google Translate, the translation services and their data would be considered ELFs.

9. Local Event Calendars
- If app integrates with external event calendars to help users stay informed about local events that may be relevant to lost items or activities in their area, the event listings would be ELFs.

These External Logical Files play a crucial role in providing additional data and services to enhance the functionality and user experience of lost2find app.

Parameter	COUNT	SIMPLE	AVERAGE	**COMPLEX**	MULTIPLY	EQUAL
EQ	10	3	4	**6**	10*6	60
EI	12	4	5	**7**	12*7	84
EO	12	3	4	**6**	12*6	72
ILF	10	7	10	**15**	10*15	150
EIF	9	5	7	**10**	9*10	90
					TOTAL=	456

Unadjusted Functional Point=456

- The scores (ranging from 0 to 5) for these characteristics are then summed, based on the following formula, to arrive at the value adjustment factor (VAF) where ci is the score for general system characteristic i.

$$VAF = 0.65 + 0.01 \sum_{i=1}^{14} c_i$$

1. Data communication: The ability of the system to transmit and exchange data between different users rating: 3
2. Distributed data processing: The system's capability to process data across multiple servers of different admin database. Rating: 3
3. Performance: The speed and responsiveness of the system in handling user requests and processing operations. Rating: 3

4. Heavily used configuration: The system's ability to handle a large number of concurrent lost things data and heavy traffic loads. Rating: 2
5. Transaction Rate: The system's capacity to handle a high volume of transactions efficiently. Rating: 3
6. Online data entry: The amount of online data entry required by user Rating: 3
7. End-user efficiency: The system usability and efficiency in facilitating tasks and workflows for end users. Rating: 3
8. Online update: how often data is modified or updated in real time Rating: 4
9. Complex processing. Our system handle all the complexities Rating: 5
10. Reusability: our app have capacity to reuse component and code Rating: 4
11. Installation ease: we should launch our system to worldwide. Rating: 3
12. Operational ease: The ease of managing and operating the system Rating: 3
13. Multiple sites: The ability to support and manage the system across multiple locations or sites. Rating: 3
14. Facilitate change: System is flexible and adaptable Rating: 3

Now sum all these 14 characteristics rating = 3+3+3+2+3+3+3+4+5+4+3+3+3+3=45
Now calculate Value adjustment factor VAF =0.65+0.01(45)

VAF=1.1
Adjusted functional point=VAF*UFP
AFP=1.1*456
AFP=501.6

Calculation in Months:

FP estimated = Count Total * VAF = 456 * 1.1 = 501.6 ~ = 501

Labor Rate = Rs. 10,000/Month **(454.5/day)**

Average Productivity = 10 FP/Month **(0.45 FP/day)**

Cost per Function Point =Labor Rate/Productivity Parameter = 10,000/10 = 1,000 Rs/FP

Total Project Cost = FP estimated * Cost per FP = 501*1,000 = Rs 501,000

Total Estimated Effort = Estimated FP/Productivity = 501/10 = 50.1 PM ~=50 PM

1.4 CPM

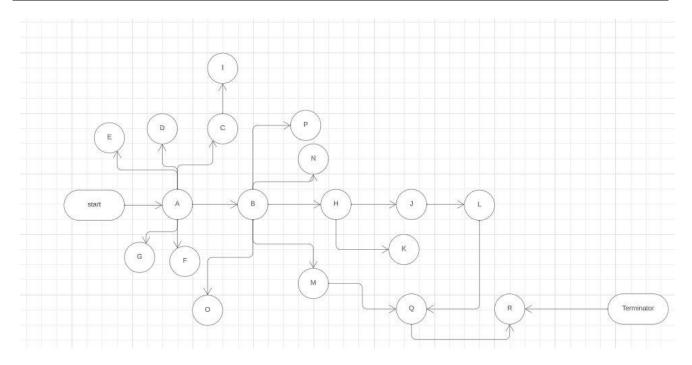| ACTIVITY	ACTIVITY NAME	IMMEDIATE PREDECESSOR	DURATION IN DAYS
A	Project Initiation	None	5
B	Feasibility Study	A	9
C	Requirement Gathering	A	18
D	Market Research and Analysis	A	17
E	Budgeting and Financial Planning	A	13
F	Legal and Regulatory Compliance	A	18
G	Vendor Selection and Procurement	A	23
H	App Design and UX Planning	B	27
I	Marketing and Promotion Planning	C	25
J	Testing and Quality Assurance	H	18
K	Security and Privacy Assessment	H	22
L	User Support and Helpdesk Setup	J	17
M	Launch Preparation	B	25
N	User Training and Education	B	10
O	Monitoring and Analytics Setup	B	15
P	Documentation and Knowledge Transfer	B	8
Q	Ongoing Maintenance and Operations	L,M	67
R	User Feedback and Improvement Process	Q	14

Activity ID	Duration	ES	EF	LS	LF	TS	FS
A	5	0	5	0	5	0	0
B	9	5	14	5	14	0	0
C	18	5	23	5	23	0	0
D	17	5	22	5	22	0	0
E	13	5	18	5	18	0	0
F	18	5	23	5	23	0	0
G	23	5	28	5	28	0	0
H	27	32	59	28	55	0	0
I	25	41	66	45	70	0	0
J	18	59	77	55	73	0	0
K	22	59	81	55	77	0	0
L	17	95	112	91	108	0	0
M	25	14	39	15	40	0	0
N	10	14	24	15	25	0	0
O	15	14	29	15	30	0	0
P	8	14	22	15	23	0	0
Q	67	39	106	40	107	0	0
R	14	106	120	107	121	0	0

The critical path is:

A→B→H→J→L→Q→R

The critical time is:

5+9+27+18+17+67+14 = 157 days

1.5 Introduction to Team member and their skill set

Name	Roll No	Skill sets	Description
Laiba Liaquat	BSEF20A049	➢ Dart ➢ Flutter ➢ Php ➢ Laravel ➢ Firebase	She is good in Programming (e.g., Dart, Flutter, PHP, Laravel), mobile app development, web development, database management, version control. She is also good in designing and documentation skills.
Noor Fatima	BSEF20A020	➢ Dart ➢ Flutter ➢ Php ➢ Laravel ➢ Firebase	She is good in designing and documentation skills. She is also good in Programming (e.g., Dart, Flutter, PHP, Laravel) and web development.

1.6 Tools and Technology with reasoning

Following tools and technologies we will use for accomplishing our project.

Front-end Development:
1. **Programming Languages:**
   - **Dart**:
     Dart is selected as the primary programming language for developing your Android application using the Flutter framework. The reason is that Dart is the official language for Flutter, providing a seamless development experience. It allows for efficient code execution, making it suitable for mobile app development.
     Reasoning:
     Dart is specifically designed for creating high-performance mobile apps. It compiles to native code, reducing runtime overhead and improving app responsiveness.
2. **Development Environment:**
   - **Flutter Framework**:
     Flutter is chosen for its cross-platform capabilities. With a single codebase, it allows you to develop for both Android and iOS, saving time and effort.
     Reasoning:

Flutter's approach is cost-effective, enabling a faster time-to-market while ensuring a consistent user experience across different platforms.

3. **Database:**

- **Firebase/Firestore**:

    Firestore is ideal for project because of its real-time database features. This ensures that users can quickly interact with the app and receive timely updates.

    Reasoning:

    Firestore provides seamless synchronization, making it perfect for a lost-and-found application where users need to communicate about found items in real-time.

4. **Backend Development:**

- **PHP**:

    PHP is known for server-side processing, making it suitable for managing user accounts, ad data, and interactions.

    Reasoning:

    PHP offers excellent support for processing form data and handling server-side operations. It's cost-effective and well-suited for your needs.

5. **Development Tools:**

- **Android Studio**:

    Android Studio is the official IDE for Android app development. It offers an array of development tools, debugging features, and Android-specific support.

- **Visual Studio Code (VS Code)**:

    VS Code is a versatile code editor suitable for writing and editing Dart code for your Flutter app. It's known for its extensions and developer community.

    Reasoning:

    Android Studio and VS Code provide a comprehensive development environment with robust debugging capabilities and a broad community, enhancing your development workflow.

6. **Security and Communication:**

- **HTTPS and APIs**:

    Implementing HTTPS ensures secure communication between the app and the server. APIs facilitate data exchange and interactions between the frontend and backend.

    Reasoning:

    Security is paramount, especially for an application handling user data. HTTPS and APIs are essential for protecting user information and ensuring reliable communication between users, their ads, and lost items.

- **GitHub:**
  This allows developers to collaborate, as they can download a new version of the application, make changes and upload the newest revision. Developers can see these new changes, download them and contribute.
- **Microsoft Office:**
  We will use it for documentation purpose.

1.7 Vision Document

This document is to set out the high-level vision for the project "LOST2FIND".

Vision Document: Lost2find

1. Vision

Empowering Individuals to Reclaim Their Lost Valuables with Ease and Confidence.

2. Project Description

Lost2find is a revolutionary mobile application and web portal designed to address the widespread issue of lost valuables. It aims to connect individuals who have lost their possessions with those who have found them. Our vision is to create a user-friendly platform that enables people to post details about their lost items and effortlessly reunite with their cherished belongings.

3. Objectives

1. **Solve the Lost Valuables Problem:** Provide a dedicated and efficient platform for individuals to find their lost possessions promptly.
2. **User-Friendly Experience:** Develop a seamless and intuitive mobile application and web portal to cater to users of all backgrounds.
3. **Facilitate Return of Valuables:** Create a community-driven network that encourages honest individuals to return found valuables.
4. **Enhance Security and Trust:** Implement robust security measures to protect user data and build trust within the Lost2find community.
5. **Revenue Generation:** Explore sustainable revenue models to support the continued growth and development of the platform.

4. Scope

- **Mobile Application:** Lost2find will be accessible through an Android and iOS application.
- **Web Portal:** A web portal will complement the mobile application, facilitating interaction through various devices.
- **Database:** A database will store user profiles, lost and found item details, communication history, and advertisements.

5. Target Audience

Our intended audience is anyone, but it will help the society living people more because they can easily find their lost valuables by just posting and ad as soon as possible and also for those peoples who are looking to return these lost valuables.

6. Unique Selling Proposition (USP)

Lost2find's USP lies in its commitment to connecting individuals, fostering trust, and providing a simple, secure, and community-driven platform for finding lost valuables.

7. Future Enhancements

As the Lost2find community grows, we envision incorporating features like real-time chat, advanced search options, and expanded geographical coverage.

1.7.1 Problem Statement:

1. **Inefficient Traditional Methods:**
   Current approaches for recovering lost items, such as physical posters or word-of-mouth inquiries, are inefficient and lack a wide-reaching scope.

2. **Lack of a Centralized Platform:**
   There is no centralized platform that effectively connects individuals who have lost their valuables with those who have found them.

3. **Privacy and Trust Concerns:**
   Users may be hesitant to share personal information or details about their lost items on public platforms due to privacy and trust concerns.

4. **Limited Communication Channels:**
   The absence of dedicated communication channels between the person who lost the item and the finder makes the recovery process cumbersome.

5. **Inadequate Tracking:**
   Traditional methods offer limited means of tracking the status of the lost item or the efforts made for its return.

1.7.2 Successful Solutions:

1. **User-Friendly Mobile Application and Web Portal:**
   - Develop an intuitive and easy-to-use mobile application and web portal that caters to users of all technical backgrounds. A well-designed interface ensures that users can efficiently post information about their lost valuables and search for found items.

2. **Centralized Lost-and-Found Platform:**
   - Create a centralized platform that connects individuals who have lost their valuables with those who have found them. This platform allows users to report lost items and search for found ones, consolidating all related activities in one place.

3. **Privacy and Data Security Measures:**
   - Implement robust data security measures to address privacy concerns. Ensure that user data is securely handled, and obtain any necessary certifications to instill trust in the platform.

4. **Efficient Communication Channels:**
   - Enhance communication channels within the application to facilitate interaction between those who have lost items and those who have found them. Provide secure messaging and calling features to enable users to connect and arrange item returns.

5. **Advanced Tracking Capabilities:**
   - Improve item tracking features to keep users informed about their lost items' status. Users should receive real-time updates, such as when their item was last seen or if there have been retrieval attempts.

1.7.3 Positioning Statement:

Unlike other lost-and-find solutions, Lost2Find offers a user-friendly interface, advanced tracking capabilities, and robust data security, providing a one-stop hub for users to report their losses and recover their items efficiently. Our commitment to user trust and efficient communication channels sets us apart, making Lost2Find the go-to solution for reuniting people with their cherished belongings."

1.7.4 User Environment:

Lost2Find offers a user environment that caters to "valuable losers" seeking to recover lost items, "valuable finders" who have discovered misplaced belongings and wish to return them, "General Users" interested in assisting or staying informed, and an "Admin and Support Team" responsible for platform management and user assistance. With its user-friendly interface, the application serves as a collaborative hub, promoting trust and cooperation among users while addressing the common issue of lost valuables.

1.7.5 Assumptions and Dependencies:

- The system will be Up-to-date
- We will rely on Google API's for getting location services
- It will have good quality internet access
- Enough space/memory in order to perform well.
- Mobile devices should have proper operating system.

Device should be internet capable.

1.7.6 Project Features:

Lost2Find project features include:

- **User Profiles:** Users can create profiles with essential details, such as name, address, contact information, and a description of the lost or found items.
- **Lost Item Listings:** Users can post detailed descriptions of lost items, including appearance, name, location, and the circumstances surrounding their loss.
- **Found Item Listings:** Valuable Finders can post advertisements about items they've found and wish to return.
- **Search Functionality:** The application offers robust search capabilities, allowing users to find lost or found items that match their criteria.
- **Communication Channel:** Users can communicate with each other via provided contact information, such as phone numbers and email addresses, to coordinate the return of items.
- **User Reviews:** A rating and review system enables users to share their experiences, enhancing the platform's trustworthiness.
- **Privacy and Data Security:** Stringent data security measures are in place to protect user information.

- **User Support:** An admin and support team is available to assist users and manage the platform effectively.
- **Revenue Generation:** The platform incorporates features for revenue generation, such as advertising, premium services, and partnerships.
- **User-Friendly Interface:** The application is designed with an intuitive and easy-to-navigate user interface, ensuring accessibility for a wide range of users.

1.8 Risk List

Project risk management is useful and necessary as it helps with the early identification of problems. After their identifications, we can take actions to overcome these risks. Our project has following risks that has written in the decreasing order of their priority.

Risk ID: RSK001

- **Risk:** Technical Challenges
- **Impact:** Delays in project completion, increased development costs.
- **Likelihood:** Moderate
- **Mitigation:** Engage a skilled development team, conduct thorough testing, and maintain open communication to address challenges promptly.

Risk ID: RSK002

- **Risk:** User Adoption
- **Impact:** Low user engagement, potential revenue shortfalls.
- **Likelihood:** Moderate
- **Mitigation:** Implement a user-friendly design, offer incentives for early adopters, and conduct marketing campaigns to boost user adoption.

Risk ID: RSK003

- **Risk:** Privacy and Data Security
- **Impact:** Loss of trust, legal implications, reduced user base.
- **Likelihood:** Low
- **Mitigation:** Implement robust data security measures, obtain necessary certifications, and be transparent about data handling practices.

Risk ID: RSK004

- **Risk:** Competitive Landscape
- **Impact:** Reduced market share and revenue.
- **Likelihood:** Moderate
- **Mitigation:** Continuously monitor the competition, enhance platform features, and maintain a unique value proposition.

Risk ID: RSK005

- **Risk:** Technical Outages
- **Impact:** Negative user experiences, loss of engagement.
- **Likelihood:** Low
- **Mitigation:** Implement robust server infrastructure, maintain backups, and provide users with real-time updates during outages.

Risk ID: RSK006

- **Risk:** Legal and Compliance
- **Impact:** Legal action, financial penalties.
- **Likelihood:** Low

- **Mitigation:** Consult legal experts to ensure compliance with all relevant laws and regulations, and establish clear user terms and conditions.

Risk ID: RSK007

- **Risk:** Revenue Generation.
- **Impact:** Financial difficulties and resource constraints.
- **Likelihood:** Moderate
- **Mitigation:** Diversify revenue streams, seek strategic partnerships, and adapt the business model as needed.

Risk ID: RSK008

- **Risk:** User Trust.
- **Impact:** Reduced user engagement, negative word of mouth.
- **Likelihood:** Moderate
- **Mitigation:** Ensure effective communication, handle lost item cases with transparency and diligence.

1 Introduction

1.1 Systems Specifications

The following are the clauses that must be included while describing the system specifications.

Introduction

Due to its platform being simple to use and affordable digital devices for immediate cell phones, tablets, laptops, desktops, and now even the smart televisions and other computing devices, apps have grown to be the most often used application software. In this project, we are willing to make a system based on the concept of User and Employee and User and User. User do effort to return someone's valuable and he can also make some money if there is reward included in the deal. In this system there will be one person will be doing efforts like an employee and he will be called Valuable Returner and the other person will use his service to get his valuables back and that person will be named as User. Only a single android application interfaces will be used for both parties just. A database will be used to save communication history between both parties for security purpose

Purpose

Lost2Find as its name suggests, is the idea of connecting the users on one social platform to help the users to find their lost valuables. It is also difficult for the user to find his lost valuable on social media platforms that we use today. This platform will completely avoid the risk of any scammer because no scammer can get anything after posting ad of anything but no one will contact him and at the end his ad will be turned down. Lost2Find will allow the user to register themselves with the basic information about their addresses and their email and phone number, later they can login to the app through their username and password.

Organizational Chart

As this project is for learning purposes as a student's final year project, not for any organization, therefore there is no organizational chart.

Scope of the System

There is no dedicated app to find your lost items in Pakistan, so it has very scope to all over the Pakistan because the people have never seen that type of service before, it will be new to everyone. Before the app, the people use to go to the locations where he is guessing that there he could find his lost item and if he fails to find that than it will be considered lost there will be no chance of finding it back. It will provide the following benefits: -

1. To facilitate people in a better way.
2. Develop a new application for service.
3. Fast and time saving

Summary of Requirements (Initial Requirements)

1. Mobile Application: The project will involve the development of a mobile application using Flutter, catering to both Android and iOS platforms.

2. User Profiles: Users will create profiles with essential details such as name, contact information, and location.

3. Lost Item Listings: Users can post detailed descriptions of lost items, including name, appearance, location, and circumstances of loss.

4. Found Item Notifications: Users who find lost items can contact the owners through the app to facilitate their return.

5. Category-Based Search: Items will be categorized for easy search, helping users quickly identify lost items they are looking for.

6. Reward System: Owners can offer rewards to encourage the return of their valuable items.

7. Real-Time Communication: The app will enable real-time communication between users to coordinate the return of items.

8. Risk Management: A risk list will be maintained to identify potential project risks and mitigation actions.

9. Data Security: User data, lost item records, and interactions will be securely managed, ensuring user privacy.

10. Agile Methodology: Agile will be adopted as the development methodology to support iterative development and adaptability to evolving user needs.

11. Version Control: Git and GitHub will be used for version control, enabling collaborative development and code management.

12. Cloud Hosting: Cloud hosting services like AWS or GCP will be considered for scalability and reliability.

13. Communication Tools: Communication and collaboration tools like Slack or Microsoft Teams will be utilized for effective team communication and project management.

The system should have following functionalities:

- Sign Up
- Login
- Add new valuable user.
- Remove a valuable user
- Update a valuable user record
- Update a user's record
- Remove users
- Show details of a valuable user
- Generate list of valuable users according to specific disease
- Generate list of users of a valuable user
- Update some record.
- Filter FAQ
- Generate user list of specific valuable user
- Generate monthly valuable users record
- Give rating to valuable user
- Generate monthly users report
- Rate App and give feedback
- Logout

There are following three possible users of this system

- Admin
- Individuals Who Lost Valuables:
- Users Who Found Lost Items

1. Operations performed by non-registered user:

1.1 Sign up:

Non registered users can only register for their account.

2. Operations Performed by User & Admin:

2.1 Login:

Both administrators and users will log in to the application using their respective credentials to access their accounts and perform various actions.

2.2 Logout:

Both administrators and users will log out from the application to protect their accounts.

2.3 Search and Filtering:

Both groups can search for lost and found advertisements and apply filters to refine their search results. This helps users find specific items, and administrators can use this feature to locate specific ads for review.

2.4 Contacting Users:

Both users can view the profiles details of each other and both administrators and users can initiate contact with other users via the platform's messaging system. Users contact each other regarding lost and found items, while administrators may communicate with users for various reasons, including ad approval or dispute resolution.

2.5 Notification Management:

Both administrators and users can manage their notification preferences. Users may choose to receive notifications about found items, while administrators can set their preferences for system notifications.

2.6 System Updates:

Both administrators and users benefit from regular system updates and improvements to ensure the platform operates efficiently and securely.

2.7 Change password:

Users can change their password to a new password

2.8 Feedback:

User shall rate the app and give feedback

3. Operations Performed by Admin:

3.1 Login:

Admin can login in to the app by providing his username and password. Users need to sign up first, then they can login.

3.2 Logout**:**

The Admin who are logged in the app can logout from the app by clicking the logout icon or logout button.

3.3 Approve/reject/block new users**:**
    The admin is able approve or reject the user from entering application.
3.4 Profile Management:
    Admin can update picture, records and name on the profile.
3.5 Feedback:
    Admin can give feedback of the app.
3.6 View list of all Users**:**
    Admin can view the list of all the Users.
3.7 View details of any Users**:**
     He can view the details of Users.

1.2 Identifying External Entities

- Admin
- User
- Valuable User

1.3 Capture "shall" Statements and the external entities (Actors)

Sr #	Para #	External Entity	Initial Requirement
1	1.1	Non-registered user	Non-registered user **"shall"** request for registration (sign-up) if not already registered.
2	2.1	Valuable User	Valuable User "**shall**" login to his account.
3	2.2	Valuable User	Valuable User **"shall"** logout of account.
4	2.8	User	User **"shall"** rate the app and give feedback.
5	3.1	Admin	Admin **"shall"** login to her account.
6	3.1	Admin	Admin **"shall"** login to his account.
7	3.2	Admin	Admin **"shall"** logout of her account.
8	3.3	Admin	Admin **"shall"** approve / reject new user.
9	3.4	Admin	Admin **"shall"** block the user.
10	3.5	Admin	Admin **"shall"** rate the app and give feedback.
11	3.7	Admin	Admin **"shall"** view details of every USER.
12	3.6	Admin	Admin **"shall"** view list of USER

1.4   Allocate Requirements

Sr #	Para #	Initial Requirement	Use Case Names
1	1.1	Non-registered user **"shall"** request for registration (sign-up) if not already registered.	UC_user_registration
2	2.1	Valuable user **shall**" login to his account.	UC_Login
3	2.2	Valuable user **"shall"** logout of account.	UC_Logout
4	2.4	Valuable user **"shall"** view user details.	UC_View_profile
5	2.4	User **"shall"** view valuable user's details.	UC_View_user_profile
6	2.8	User or Valuable user **"shall"** shall rate the app and give feedback.	UC_Rating_and_give feedback
7	3.1	Admin **"shall"** login to her account.	UC_Admin_Login
8	3.1	Admin **"shall"** login to his account.	UC_Admin_Login
9	3.2	Admin **"shall"** logout of her account.	UC_Logout
10	3.3	Admin **"shall"** approve / reject new user.	UC_Admin_manage_user
11	3.3	Admin **"shall"** block the user.	UC_Admin_manage_User
12	3.5	Admin **"shall"** rate the app and give feedback.	UC_Rating_and_give Feedback
13	3.6	Admin **"shall"** view list of Valuable user.	UC_View_List
14	3.7	Admin **"shall"** view details of every Valuable user.	UC_View_profile

1.5   Prioritize Requirements

Sr #	Para #	Rank	Initial Requirement	Use Case_ID	Use Case Names
1	1.1	High	Non-registered user **"shall"** request for registration (sign-up) if not already registered.	UC_1	UC_user_registration
2	2.1	High	VALUABLE USER "**shall**" login to his account.	UC_2	UC _Login
3	2.2	High	VALUABLE USER **"shall"** logout of account.	UC_15	UC_Logout
4	2.4	Medium	VALUABLE USER **"shall"** view user profile details.	UC_9	UC_View_user_profile
5	2.4	High	User **"shall"** view valuable user details.	UC_9	UC_View_user _profile
6	2.8	High	User or valuable user **"shall"** shall rate the app and give feedback.	UC_14	UC_Rating_and_give feedback
7	3.1	High	Admin **"shall"** login to her account.	UC_10	UC_Admin_Login
8	3.2	High	Admin **"shall"** logout from his account.	UC_15	UC_Admin_Logout
9	3.3	High	Admin **"shall"** approve / reject new user.	UC_12	UC_Admin_Manage_User
10	3.3	High	Admin **"shall"** block the user.	UC_12	UC_Admin_Manage_User

---

11	3.5	Low	Admin **"shall"** rate the app and give feedback.	UC_14	UC_Rating andFeedback
12	3.6	High	Admin **"shall"** view list of VALUABLE USER	UC_13	UC_View_List
13	3.7	High	Admin **"shall"** view details of every VALUABLE USER.	UC_9	UC_View_user_profile

1.6 Requirements Trace-ability Matrix

The requirements trace-ability matrix is a table used to trace project life cycle activities and work products to the project requirements. The matrix establishes a thread that traces requirements from identification through implementation.

Sr #	Para #	Initial Requirement	Use Case Names	Build	Category
1	1.1	Non-registered user **"shall"** request for registration (sign-up) if not already registered.	UC_user_registration	P2	Business
2	2.1	VALUABLE USER "**shall**" login to his account.	UC_Login	P2	Business
3	2.2	VALUABLE USER **"shall"** logout of account.	UC_Logout	P2	Business
4	2.4	VALUABLE USER **"shall"** view user details.	UC_View_user_profile	P2	Business
5	2.4	User **"shall"** view valuable user details.	UC_View_user_profile	P3	Business
6	2.8	User or valuable user **"shall"** shall rate the app and give feedback.	UC_Rating_and_give feedback	P3	Business
7	3.1	Admin **"shall"** login to her account.	UC_Admin_Login	P2	Business
8	3.1	Admin **"shall"** login to his account.	UC_Admin_Login	P2	Business
9	3.2	Admin can logout from his account.	UC_Admin_Logout	P2	Business
10	3.3	Admin **"shall"** approve / reject new user.	UC_Admin_Manage_user	P3	Business
11	3.4	Admin **"shall"** block the user.	UC_Admin_Manage_User	P2	Business
12	3.5	Admin **"shall"** rate the app and give feedback.	UC_rating_and_give Feedback	P2	Business
13	3.6	Admin **"shall"** view list of VALUABLE USER.	UC_View_List	P2	Business
14	3.7	Admin **"shall"** view details of every VALUABLE USER.	UC_View_profile	P2	Business

1.9 Use Case Diagram: