****

THE UNIVERSITY OF BUEA

P.O Box 63,Buea

Buea,South West Region

Cameroon

**REPUBLIC OF CAMEROON**

PEACE-WORK-FATHERLAND

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**COMPUTER ENGINEERING**

**CEF440: INTRENET PROGRAMMING AND MOBILE PROGRAMMING**

**REQUIREMENT ANALYSIS FOR THE DESIGN AND IMPLEMENTATION OF A MOBILE BASED ARCHIVAL AND RETRIEVAL OF MISSING OBJECTS APPLICATION USING IMAGE MATCHING**

**MAY/2024**

**COURSE INSTRUCTOR**

**Dr. NKEMENI VALERY**

PRESENTED BY: **By Group 21**

|  |  |  |
| --- | --- | --- |
| **S\N** | **NAME** | **MATRICULE** |
| 1 | AKENGNI KEANLI EMMANUEL | FE21A132 |
| 2 | CHOMOH SYLVIA FONCHENALLAH | FE21A162 |
| 3 | MBI AYAMBA DIANNA | FE21A230 |
| 4 | NGAUEN NDJOMOU LOICE VANELLE | FE21A258 |
| 5 | NGOBA STEVE JONES NTONSG | FE21A259 |

Contents

[INTRODUCTION 2](#_Toc166509595)

[1.1 PROBLEM STATEMENT 2](#_Toc166509596)

[1.2 PROJECT GOALS 2](#_Toc166509597)

[1.3 PR0JECT SCOPE 3](#_Toc166509598)

[1.4 Methodology 3](#_Toc166509599)

[2. Requirement Prioritization 3](#_Toc166509600)

[2.1 Must Have (Critical for Launch): 4](#_Toc166509601)

[2.2 Should Have (Important but Not Critical for Initial Launch): 5](#_Toc166509602)

[2.3. Could Have (Desirable Features): 6](#_Toc166509603)

[2.4. Won’t Have (Lowest Priority): 7](#_Toc166509604)

[3. System Requirements Specification 8](#_Toc166509605)

[3.1. Functional Requirement 9](#_Toc166509606)

[3.2 Non Functional Requirement 10](#_Toc166509607)

[3.3 Hardware Requirements: 11](#_Toc166509608)

[3.4 Software Requirements: 11](#_Toc166509609)

[4. Technical requirement: 12](#_Toc166509610)

[5. Regulatory compliance: 15](#_Toc166509611)

[6. Feasible Requirements: 17](#_Toc166509612)

[6.1 Potentially Feasible Requirements: 17](#_Toc166509613)

[6.2 Not Feasible Requirements: 18](#_Toc166509614)

[7. Analyzing market competition: 18](#_Toc166509615)

[8. CONCLUSION 19](#_Toc166509616)

[9. REFERENCES 20](#_Toc166509617)

# INTRODUCTION

The proposed application aims to provide a solution for the common problem of lost or misplaced items by utilizing mobile technology and advanced image recognition algorithms. By allowing users to upload images of lost items and search for matches against a central database, the application will offer a modern and efficient alternative to traditional search methods.

# 1.1 PROBLEM STATEMENT

In a world where the loss of personal belongings and valuables is a common occurrence, there is a significant need for a reliable and efficient way to assist individuals in recovering their lost items. Traditional methods of reporting and searching for lost items are often time-consuming, labor-intensive, and have a low success rate. The lack of a centralized, accessible, and technologically advanced system for reporting and retrieving lost items exacerbates the problem.

# 1.2 PROJECT GOALS

* To create an easy-to-use mobile platform that allows users to upload images of lost or found objects.
* To use image matching technology to automate the comparison of newly uploaded images against existing entries in a robust database.
* To enable users to communicate securely and coordinate the return of objects.

# 1.3 PR0JECT SCOPE

This report explores the development and implementation of a Mobile-Based Archival and Retrieval of Missing Objects Application using Image Matching technology. It delves into the functionalities of the proposed application, examining how image matching algorithms can be leveraged to efficiently locate lost items. Additionally, it investigates potential use cases across various sectors, such as transportation, hospitality, and personal belongings. Furthermore, the report assesses the effectiveness and feasibility of the proposed solution in comparison to traditional methods of object retrieval.

# 1.4 Methodology

The requirement analysis was conducted through a series of steps:

* Stakeholder interviews to gather insights and expectations.
* User surveys and questionnaires to identify common use cases and desired features.
* Market research to analyze existing solutions and identify gaps.
* Technical feasibility studies to assess the capabilities of current image matching technologies.

# 2. Requirement Prioritization

Requirement prioritization is a critical process in the development of any software project. It involves evaluating and ranking each requirement based on its importance to the project's objectives, stakeholders' needs, and the overall value it adds to the product. This report focuses on the prioritization of requirements for the proposed mobile application designed to help users find missing objects using image matching technology.

Prioritization Methodology:

For this project, we will use the MoSCoW method of prioritization, which categorizes requirements into four groups:

* Must have (M): Essential for the project's success and form the minimum usable subset.
* Should have (S): Important but not necessary for launch; can be included in future releases.
* Could have (C): Desirable but not necessary and could improve user experience or customer satisfaction if included.
* Won’t have (W): Least critical, lowest payback items, or not appropriate at this time.

# Must Have (Critical for Launch):

1. **User Authentication**:

* Secure login/logout functionality.
* Password recovery and account verification mechanisms.

1. **Image Upload and Database Management:**

* Basic image upload functionality with metadata (description, location, timestamp).
* A centralized database infrastructure for storing and retrieving images efficiently.

1. **Core Image Matching Functionality:**

* Robust image recognition and matching algorithms.
* Accurate and fast search results for matching missing objects.

1. **Real-Time Notifications:**

* Immediate alerts for users when potential matches are found.
* Push notification service integration.

1. **Data Privacy Compliance:**

* Adherence to GDPR and other relevant data protection regulations.
* Secure handling and storage of user data.

# Should Have (Important but Not Critical for Initial Launch):

1. **Profile Management:**

* Ability for users to manage their profiles and posted items.
* Options for users to edit and delete their posts.

1. **Advanced Search Filters:**

* Additional search parameters such as date, location, and category.
* Enhanced sorting and filtering capabilities for search results.

1. **User Feedback Mechanism:**

* Features for users to report success stories or provide app feedback.
* Rating system for community engagement and trust-building.

1. **Social Media Integration:**

* Sharing of posts to social platforms to increase reach.

Social login options for ease of access

# 2.3. Could Have (Desirable Features):

1. **User Community Features:**

* Forums or chat functions for user interaction.
* Gamification elements to encourage community participation.

1. **Multilingual Support:**

* Translation features to cater to non-English speaking users.
* Localization of content based on the user's geographic location.

# 2.4. Won’t Have (Lowest Priority):

1. **Augmented Reality (AR) Integration:**

* AR features for visualizing lost items in the environment.
* Due to budget and time constraints, this feature will not be included in the initial release.

1. **Offline Access:**

* Ability to use certain features of the app without an internet connection.
* Considered a luxury feature that can be explored in future updates.

The prioritization of requirements ensures that the development team can focus on delivering the most valuable features first, aligning with the project's strategic goals and constraints. This prioritized list will guide the development process and help stakeholders understand the project's focus areas for the initial launch and future updates.

# System Requirements Specification

The System Requirements Specification (SRS) document outlines the system requirements for the mobile application designed to archive and retrieve missing objects using image matching technology. This includes functional nonfunctional and both software and hardware requirements needed to support the functionality of the application.

**Purpose**:

The purpose of this SRS is to provide a detailed overview of the functional and non-functional requirements that will guide the development and ensure the application meets the needs of its users and stakeholders.

**Scope**:

The application will allow users to upload images of missing objects, search for objects using image matching, and receive notifications for potential matches. It will support both iOS and Android platforms.

# 3.1. Functional Requirement

Functional requirements describe what the system should do. They detail the behaviors and functions the application must support. These include:

* + 1. **User Account Management:**
* The system shall allow users to create, edit, and delete their accounts.
* The system shall provide password recovery and account verification functionality.
  + 1. **Image Processing:**
* The system shall allow users to upload images with associated metadata.
* The system shall perform image matching using a predefined algorithm.
  + 1. **Search and Retrieval:**
* The system shall provide a search function for users to find objects.
* The system shall display search results with potential object matches.
  + 1. **Notifications:**
* The system shall send real-time notifications to users when potential matches are found.
* The system shall allow users to customize notification settings.
  + 1. **Data Management and Privacy:**
* The system shall comply with data protection regulations.
* The system shall ensure secure data transmission and storage.

# 3.2 Non Functional Requirement

Non-functional requirements specify how the system performs certain tasks and under what constraints. And then include:

**Performance:**

* The system shall return search results within an acceptable time frame.
* The system shall handle a high number of concurrent users without degradation of performance.

**Usability:**

* The system shall offer an intuitive and user-friendly interface.
* The system shall provide help documentation for user support.

**Reliability:**

* The system shall have an uptime of 99%.
* The system shall provide accurate image matching results.

**Security:**

* The system shall implement industry-standard encryption for data transmission.
* The system shall have measures to prevent unauthorized access and data breaches.

**Scalability:**

* The system shall be designed to easily accommodate an increasing number of users.
* The system shall allow for future expansion of features and functionalities.

# Hardware Requirements:

**Mobile Devices:**

* The system shall be compatible with iOS devices running iOS 13.0 or later.
* The system shall be compatible with Android devices running Android 8.0 (Oreo) or later.

**Server:**

* The system shall be hosted on a server with a minimum of 8 GB RAM and quad-core CPU.
* The system shall ensure data redundancy and backup solutions are in place.

# Software Requirements:

1. **Operating Systems:**

* The system shall support the latest two major versions of iOS and Android operating systems.

**Development Frameworks:**

* The system shall be developed using cross-platform frameworks such as React Native or Flutter for mobile development.
* The system shall use a reliable database management system like PostgreSQL or MongoDB.

1. **APIs and Libraries:**

* The system shall integrate with push notification services like Firebase Cloud Messaging (FCM) or Apple Push Notification service (APNs).
* The system shall utilize image processing libraries or APIs that support image matching functionalities.

The system requirements outlined in this SRS are intended to provide a comprehensive understanding of the application's specifications. These requirements will drive the development process and ensure that the end product aligns with user expectations and project objectives.

# 4. Technical requirement:

Technical requirements for a mobile-based archival and retrieval system for missing objects using images encompass the specific technologies, frameworks, and components needed to develop and deploy the system effectively. Here's a detailed explanation of key technical requirements:

1. **Mobile App Development:**

* Cross-Platform Compatibility: Develop the mobile app for multiple platforms, such as iOS and Android, to reach a wider audience. Consider using frameworks like React Native or Flutter to build cross-platform apps with a single codebase.
* User Interface Design: Design intuitive and user-friendly interfaces that facilitate easy reporting of missing objects, searching for matches, and receiving notifications. Implement responsive design principles to ensure compatibility with various screen sizes and resolutions.
* Image Capture and Processing: Integrate functionalities for capturing images using the device's camera and processing them to enhance quality and reduce noise. Implement features like cropping, rotating, and adjusting image brightness and contrast.
* Offline Support: Implement offline support to allow users to report missing objects and search for matches even when they're not connected to the internet. Use local storage to cache data and synchronize it with the server when connectivity is restored.

1. **Backend Development:**

* Server Infrastructure: Set up a robust server infrastructure to handle image processing, storage, and matching algorithms. Use cloud services like AWS, Google Cloud, or Azure for scalability and reliability.
* API Development: Develop RESTful APIs to enable communication between the mobile app and the server. Define endpoints for user authentication, object reporting, search queries, and notifications.
* Image Matching Algorithm: Implement image recognition and matching algorithms to compare reported missing objects with found objects. Consider using computer vision libraries like OpenCV for feature extraction and matching.
* Database Management: Use a relational or NoSQL database to store user data, reported objects, matching results, and other relevant information. Ensure data integrity, security, and scalability by optimizing database design and indexing.

1. **Integration of Third-Party Services:**

* Authentication Services: Integrate authentication services like Firebase Authentication or OAuth to enable secure user authentication and authorization.
* Cloud Storage: Use cloud storage services like Amazon S3 or Google Cloud Storage to store uploaded images securely and cost-effectively.
* Push Notifications: Integrate push notification services like Firebase Cloud Messaging (FCM) or Apple Push Notification Service (APNs) to send timely notifications to users about potential matches for their reported missing objects.

1. **Security and Privacy:**

* Data Encryption: Encrypt sensitive data, including user information and uploaded images, both in transit and at rest, using industry-standard encryption algorithms.
* Secure Communication: Use HTTPS protocol for communication between the mobile app and the server to ensure data integrity and prevent eavesdropping.
* Access Control: Implement access control mechanisms to restrict unauthorized access to sensitive features and data. Use role-based access control (RBAC) or permissions-based access control to define user roles and privileges.

1. **Testing and Quality Assurance:**

* **Unit Testing**: Write unit tests to verify the functionality of individual components and ensure code quality and reliability.
* **Integration Testing**: Conduct integration testing to validate the interactions between different modules and components of the system.
* **User Acceptance Testing**: Involve real users in user acceptance testing to gather feedback on the usability, performance, and functionality of the system.

# 5. Regulatory compliance:

Regulatory compliance for a mobile-based archival and retrieval system for missing objects using images involves adhering to relevant laws, regulations, and industry standards to ensure the protection of user data, privacy, and rights. Here's a detailed explanation of key aspects of regulatory compliance:

1. **Data Protection Regulations:**

* **General Data Protection Regulation (GDPR)**: If the system operates in the European Union or processes data of EU citizens, it must comply with GDPR requirements. This includes obtaining user consent for data processing, providing transparency about data practices, and implementing measures to ensure data security and privacy.
* **California Consumer Privacy Act (CCPA)**: If the system serves users in California or collects personal information of California residents, it must comply with CCPA requirements. This includes providing users with the right to know what personal information is collected and how it's used, as well as the right to opt-out of data sharing or sale.
* **Other Data Protection Laws**: Depending on the jurisdiction, there may be other data protection laws and regulations that the system must comply with, such as the Personal Information Protection and Electronic Documents Act (PIPEDA) in Canada or the Data Protection Act in the United Kingdom.

1. **Privacy Policy:**

* **Transparency**: The system should have a clear and comprehensive privacy policy that outlines how user data is collected, used, stored, and shared. It should inform users about their rights regarding their personal information and provide instructions for exercising those rights.
* **Accessibility**: The privacy policy should be easily accessible to users, such as through a link in the mobile app's settings or on the app store listing page. It should be written in clear and understandable language, avoiding technical jargon.

1. **Security Measures:**

* **Data Encryption**: User data, including personal information and uploaded images, should be encrypted during transmission and storage to prevent unauthorized access or interception.
* **Access Control**: Implement access controls and authentication mechanisms to restrict access to sensitive data and features. Only authorized personnel should have access to user data, and access should be logged and monitored.
* **Data** **Breach** **Response** Plan: Develop and maintain a data breach response plan to address security incidents effectively. This includes procedures for detecting, containing, and mitigating data breaches, as well as notifying affected users and regulatory authorities as required by law.

1. **Compliance Documentation:**

* **Records of Processing Activities**: Maintain records of data processing activities, including the purposes of processing, categories of data subjects and personal data, and security measures implemented.
* **Consent Management**: Document user consent for data processing activities, such as user registration and opt-in for receiving notifications. Keep records of consent forms or checkboxes to demonstrate compliance with consent requirements.

1. **Audit and Oversight:**

* **Internal Audits**: Conduct regular internal audits to assess compliance with data protection regulations and identify areas for improvement. Review policies, procedures, and security measures to ensure they align with regulatory requirements.
* **External Oversight**: Cooperate with external auditors, regulators, or data protection authorities who may conduct audits or investigations to verify **compliance with applicable laws and regulations.**

6. Feasility

# 6.1. Feasible Requirements:

* **User Registration and Authentication:**

Implementing user registration and authentication is a standard practice in mobile applications and can be achieved using existing authentication frameworks or services.

* **Object Archival:**

Allowing users to capture and store images of objects along with metadata such as name, description, and location is technically achievable using mobile device capabilities and database storage.

* **Search and Retrieval:**

Implementing search functionality to retrieve archived objects based on various criteria is achievable using database querying and filtering mechanisms.

* **User Feedback and Rating:**

Allowing users to provide feedback and ratings on matched objects and retrieval experiences can enhance user engagement and satisfaction. This functionality can be implemented using user interaction features and database storage.

# 6.1 Potentially Feasible Requirements:

* **Image Matching:**

Implementing image matching algorithms to compare uploaded images with archived objects may require advanced computer vision techniques and significant computational resources. The feasibility depends on the availability of suitable algorithms and the ability to optimize them for mobile devices.

# 6.2 Not Feasible Requirements:

* **Real-Time Image Matching:**

Implementing real-time image matching, where matches are instantly identified as the user captures an image, may be technically challenging and resource-intensive for mobile devices. The processing power required for real-time matching may exceed the capabilities of typical smartphones.

* **100% Accuracy in Matching:**

Achieving 100% accuracy in image matching is unrealistic due to variations in lighting, angles, and object appearances. While algorithms can strive for high accuracy, it's essential to set realistic expectations and focus on optimizing matching performance within acceptable tolerances.

* **Unlimited Scalability without Infrastructure Costs:**

Ensuring unlimited scalability without considering infrastructure costs is impractical. Scalability solutions such as cloud hosting may incur additional expenses as the user base and data volume grow. It's crucial to balance scalability with cost-effectiveness.

# Analyzing market competition:

Analyzing market competition is crucial for understanding the landscape in which the Mobile-Based Archival and Retrieval of Missing Objects Application using Image Matching would operate. Here's an analysis of the market competition:

* **Market Landscape:**
  + Existing Solutions: Evaluate existing mobile applications and services that offer similar functionalities, such as object archiving, retrieval, and image recognition.
  + Competitors: Identify direct competitors that provide comparable features or target similar user segments.
  + Market Trends: Analyze market trends, user preferences, and emerging technologies related to image recognition and mobile applications in the object management space.
* **Competitor Analysis:**
* Features and Functionality: Compare the features and functionality offered by competitors, including object archiving methods, search capabilities, image matching accuracy, and user experience.
* User Base: Assess the size and demographics of the competitor's user base to understand their market penetration and user engagement.
* Pricing Model: Analyze the pricing model adopted by competitors, such as subscription-based, freemium, or one-time purchase, to determine competitive pricing strategies.
* **Strengths and Weaknesses:**
* Identify the strengths and weaknesses of competitors in terms of technology, user experience, customer support, and brand reputation.
* **Differentiation Strategy:**
* Unique Selling Proposition (USP): Define the unique value proposition of the Mobile-Based Archival and Retrieval of Missing Objects Application using Image Matching compared to competitors. This could include superior image matching accuracy, user-friendly interface, or innovative features.
* Target Audience: Identify specific user segments or niche markets that may be underserved by existing solutions and tailor the application's features and marketing strategy accordingly.
* Innovation: Focus on innovation and continuous improvement to stay ahead of competitors, whether through the integration of cutting-edge image recognition algorithms, seamless user experience, or novel functionalities.
* **Market Entry Barriers:**
  + Technology Barrier: Evaluate the technological challenges involved in implementing image matching algorithms and ensuring reliable performance on mobile devices.
  + Brand Recognition: Building brand awareness and trust among users may pose challenges in a competitive market dominated by established players.
  + Regulatory Compliance: Ensure compliance with data privacy regulations and industry standards, which may pose legal and operational challenges.

# CONCLUSION

In conclusion, the thorough analysis of requirements outlined in the report underscores the meticulous planning and strategic approach to developing the proposed mobile application for archival and retrieval of missing objects. By prioritizing features based on their criticality and categorizing them according to the MoSCoW method, the project ensures that essential functionalities are prioritized for the initial launch, while also considering future enhancements. The detailed specification of system requirements, both functional and non-functional, along with hardware and software needs, provides a clear roadmap for development and deployment. Moreover, the emphasis on regulatory compliance highlights the commitment to safeguarding user privacy and data security. Through meticulous attention to these requirements, the project aims to deliver a robust, user-friendly, and compliant solution that effectively addresses the challenge of lost item retrieval in a modern, technologically advanced manner.

# REFERENCES

1. M. Ghazal, F. Haneefa, S. Ali, Y. Alkhalil and E. Rashed, "Mobile-Based Archival and Retrieval of Missing Objects Using Image Matching," 2015 3rd International Conference on Future Internet of Things and Cloud, Rome, Italy, 2015,
2. [https://www.researchgate.net/figure/An-overview-of-image-hashing-consistingof-two-primary-stages-1-an-offline-stage\_fig1\_355764637](https://www.researchgate.net/figure/An-overview-of-image-hashing-consisting-of-two-primary-stages-1-an-offline-stage_fig1_355764637) (Accessed, 24 April 2024)
3. [https://www.repo.unihannover.de/bitstream/handle/123456789/14634/Feature \_detection\_and\_description.pdf?sequence=1](https://www.repo.unihannover.de/bitstream/handle/123456789/14634/Feature_detection_and_description.pdf?sequence=1) (Accessed, 24 April 2024)
4. [https://www.researchgate.net/figure/An-overview-of-image-hashing-consistingof-two-primary-stages-1-an-offline-stage\_fig1\_355764637](https://www.researchgate.net/figure/An-overview-of-image-hashing-consisting-of-two-primary-stages-1-an-offline-stage_fig1_355764637)  (Accessed, 24 April 2024)
5. <https://ilost.co/> (Accessed on, April 27)
6. <https://joinfoundhero.com/> (Accessed on, April 27)
7. <https://www.findmylost.it/> (Accessed on, April 27)