

REPUBLIC OF CAMEROON REPUBLIC DU CAMEROUN

Peace-Work-Fatherland Paix-Travail-Patrie

University of Buea Universite de Buea

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF COMPUTER ENGINEERING**

**CEF440: Internet Programming and Mobile Programming**

Design and Implementation of Mobile-Based Archival and Retrieval of Missing Objects Application using Image Matching

**Academic year 2023/2024**

SUPERVISED BY

**Dr Nkemeni Valery**

**PRESENTED BY MEMBERS OF GROUP 9**

|  |  |
| --- | --- |
| **MATRICLE** | **NAME** |
| FE21A128 | AHOUMO TEMATEU ROXANE PHILIPPINE |
| FE21A152 | BOUCHUKE BABILA DANIEL |
| FE21A166 | DERRICK MBUNBO FORCHA |
| FE21A325 | TIANI PEKINS EBIKA |
| FE21A303 | SAMUEL OSOH |

**ACKNOWLEDMENTS**

**A**cknowledging the assistances of DR NKEMENI VALERY the course instructor and an elderly for his marvelous work and knowledge mounted upon me through this years of studies, my team and some close friends for their guidance and care. May the good LORD continues to guide and protect you all .Thanks

TABLE OF CONTENT

[INTRODUCTION 4](#_Toc21078)

[1. OBJECTIVES 4](#_Toc21079)

[structure of the report 4](#_Toc21081)

[2. UI UX DESIGN 4](#_Toc21083)

1. [INTRODUCTION AND WIREFRAMING **Error! Bookmark not defined.**](#_Toc21084)
2. [UI ELEMENTS AND COLOUR CHOICE **Error! Bookmark not defined.**](#_Toc21085)
3. [VISUAL DESIGN: **Error! Bookmark not defined.**](#_Toc21086)

[3. IMPLEMENTATION 10](#_Toc21088)

1. [TECHNOLOGY USED **Error! Bookmark not defined.**](#_Toc21089)

B. DEVELOPMENT ENVIRONMENT [**Error! Bookmark not defined.**](#_Toc21090)

[C. CODE STRUCTURE **Error! Bookmark not defined.**](#_Toc21091)

[4. CONCLUSION 13](#_Toc21092)

Design and Implementation of Mobile-Based Archival and Retrieval of Missing Objects Application using Image Matching

# **INTRODUCTION**

In an increasingly digital and interconnected world, the challenge of locating and managing missing objects ranging from personal belongings to institutional assets, has become more complex and pressing. Traditional methods of retrieval, often reliant on manual searches and limited databases, are time-consuming and prone to errors. To address these limitations, we propose a novel system that leverages advanced image matching algorithms for the efficient retrieval and archival of missing objects.

This system integrates cutting-edge image recognition technology with a comprehensive and scalable database to provide a robust solution for identifying, tracking, and archiving lost items. By converting visual information into actionable data, our approach transforms how missing objects are handled, ensuring quicker recovery and systematic archival.

The design process follows a structured approach, including various diagrams to model the system comprehensively:

1. **Context Diagram:** Establishes the system foundation, detaling data flow and interactions between the user, the application, and backend services.
2. **Use Case Diagram:** Based on the context , it details the functionalities and interactions between the user and the application.
3. **Sequence Diagram:** Breaks down specific use cases into sequences of actions and messages between components.
4. **Class Diagram:** Defines the system's internal building blocks, their attributes, and methods.
5. **Deployment Diagram:** Illustrates the physical layout of the system components and their interactions. (can be created later).

These models provide a clear and concise blueprint for developing a reliable, efficient, and user-friendly application that integrates seamlessly with various external services to deliver real-time information to drivers.

# 1. OBJECTIVE

The aim of this exercise is to provide a user-friendly interface that will permit users to smoothly load their missing items while waiting for a favourite response. Talking about friendly user interface we mean a UI that will be easy for the user to understand and interpret the various icons, buttons and their signification.

## Structure of the work

The report is divided into 2 major parts. Below is the structure of the report

* UI Design
* Introduction and Wireframing
* UI elements and colour choice
* Virtual design
* Implementation
* Technologies Used
* Development Environment
* Code Structure

# 2. UI UX DESIGN

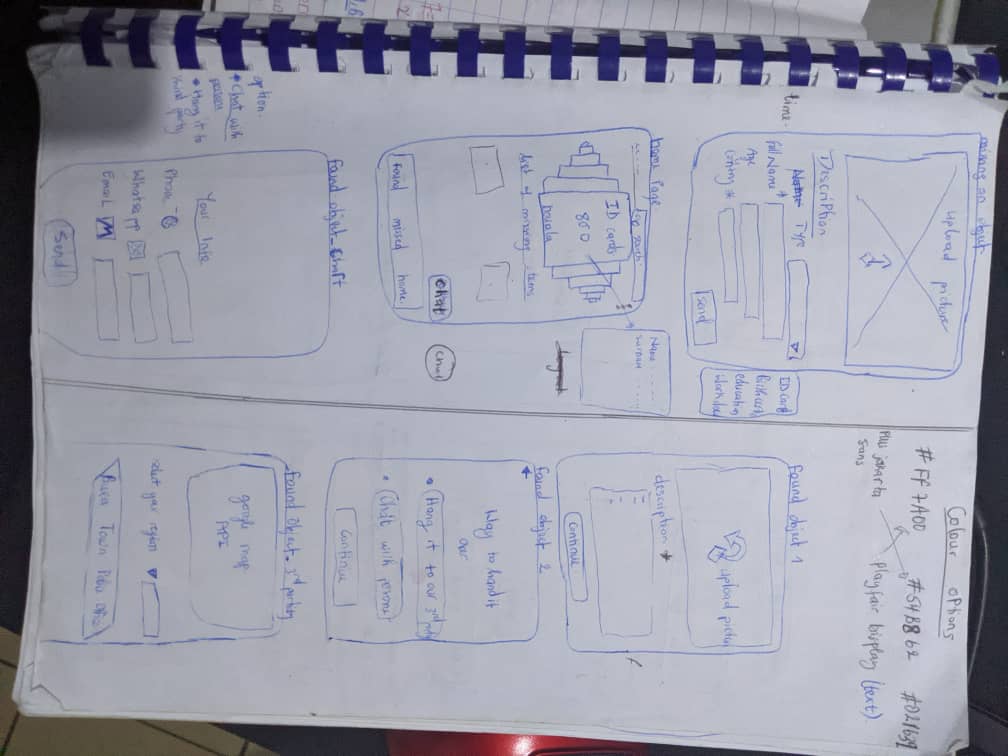
# Introduction and Wireframing

Following our previous tasks, Requirement gathering and requirement analysis (which are actually UX Design phases), we came out to understand the real need of our users, there by picturing out their expectations and understanding the way they want the interface to look like. From this work, we came out with a propose wireframing which was validated by the users. After this, we went directly into designing the UI Elements and the Visual design.

Thus, **a Wireframing** is a simple visual guide that focus on the layout, structure, and basic functionality of the UI without delving into design details like colours, graphics, or fonts. In shorts, it represents only the skeletal frame work of the UI interface.

**The layout of the wireframe** is such a way that it shows how the different elements such as headers, footers, navigation bars, content areas, and interactive elements are arranged. In this report, we have provided the **Low-Fidelity Wireframes.**

**DIAGRAMATICAL REPRESENTATION OF WIREFRAM**



BENEFITS OF WIREFRAMING

* **Provides clarity and focus:** Wireframes permit the entire team analysis (developers, designers tester, that is all the stakeholders) to focus on the essential aspect of the interface without being distracted by the visual design or UI.
* **Design testing tool**: a wireframe can actually be seen as a testing tool in design since it can easily be modified based on the feedback gotten from other stakeholders.
* **It Selves as a good design foundation:** They provide a solid foundation for more detailed design work, ensuring that the final design is based on a well-thought-out structure and user flow.

## UI ELEMENTS and COLOR CHOICE

Directly after the wireframing, we jump into creating the UI element of the app.

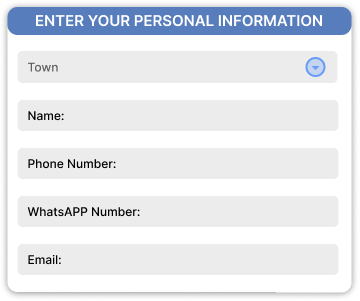
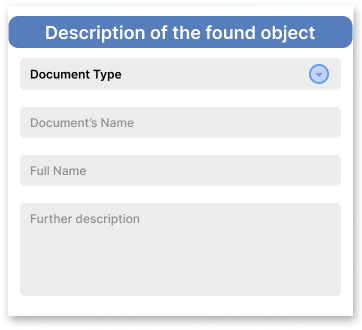
A UI element is a component or object within a graphical user interface that users interact with to perform actions or access information. UI elements are the building blocks of an application's interface and are designed to be intuitive and responsive, enhancing the overall user experience.

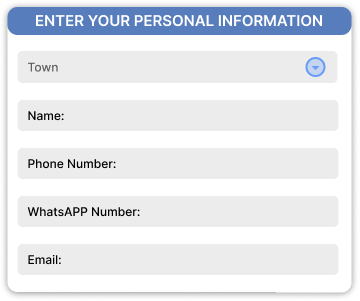
The UI element will be divided into several parts such as

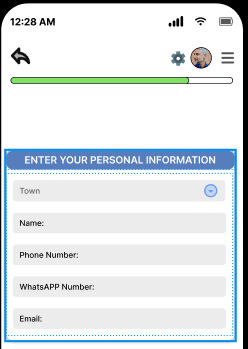
* **BUTTONS**
* Purpose: The main purpose of a Button is to trigger an action that might redirect the user into another page.
* Colour choice: as seen below, the colour of the buttons where all filled with white (#fff) and where given a dropped down shadow as border of black (#000) and an opacity of 20%. This is to permit all users including users suffering of color blindness such as **Deuteranopia (which is the inability to distinguish between green and red), Tritanopia (Absence of blue cone cells, leading to difficulty distinguishing between blue and yellow) e.t.c.**
* Example: the ‘submit’ button permits you to send the information of the items to the database and also redirect the user to the confirmation page.



* **Text Fields, Input Boxes and forms:**
* Purpose: permits the user to enter certain information base on his state, personal information and item in hand or missing.
* Colour choice: as primary colour, we choose the **medium-to-white blue** (#587DBD).
* Example: the **‘enter your personal information’** form permits the founder to enter his personal information such as his town, name, phone number and email.

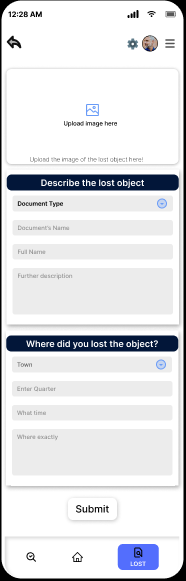


* **Drop down menus**
* **Purpose: to permit the user to select an option from a list of options.**
* **Colour choice: it is also in blue colour, but this time around with the interior shaded in black with 50% opacity.**
* **Example: on the town input boxes, the drop down purposes Douala, Yaoundé, limbe, Buea, Bafoussam e.t.c.**
* **Progression Bar**
* **Purpose:** it permits the users to evaluate the amount of time and steps needed to perform a particular task. This gives the illusion to users that the waiting time is lesser than expected.
* **Colour choice: a greenish colour was selected here due to it significant. Green colour usually reflect nature and peace of mind, thus it is use here to make the user feel well and at home when looking for his property and also to make the finder feel like he is doing the right thing by declaring a missed document.**
* **Example: on the ‘Found Page’, the user is able to see his progression as he enters all the necessary information requested.**

****

* **Icons**
* **Purpose:** provide a visual symbol representing the actions, object or concept.
* **Source:** all the icons are gotten from a plugging called **iconify.**
* **Examples**

## VISUAL DESIGN

****

# 3. IMPLEMENTATION

## Technology Used

In this part of the report, we are going to talk about all the technology we are going to use on the project starting from the frontend, to the backend, to the image algorithm, ending with the database.

The term Technology here refers to the various tools, frameworks, programming languages, libraries, and environments employed by developers to write, test, deploy, and maintain software applications. We are going to elaborate more on the technology used on each parts of the project.

1. **Frontend**

* **Programming language**

The programming used at the level of the frontend is **JAVASCRIPT** reason being that javascript is

1. Universal and versatile: javascripts can be used for both the client-side and the server-side development. With frameworks such as react native as frontend and node.js as backend, developers can work seamlessly across the entire project using a single language. Also, javascript provide cross platform compatibilities, meaning ‘the same code base can be used for web application and at the same time for mobile application.’
2. Javascript is rich in ecosystem and library. JavaScript has one of the largest and most active developer communities, which ensures robust support, continuous updates, and a wealth of resources for learning and troubleshooting needed by the image matching algorithm.
3. Javascript also provide scalability and performance by allowing many activities to be carried on at the same time.

* **Framework**

The framework used in this project is **REACT NATIVE.**

React native is good here because it provides a framework for native application. In this our case, we are to build but an ANDROID application.

1. **Backend and Database**

* **Programming language**
* **Framework**

We used expo go to be able to follow the implementation steps by steps

SUPERBASED: superbase supports many backend languages and at the same time it provides the database which will be instantly connected to the backend

## Development environment

This part talk about tools and setup used in the project. This will be further discussed below

* **TOOLS**

FRONTEND TOOLS

* **Figma**: which was used to export the various icons and pictures used in the UI Design.
* **Virtual studio code, VSCODE**: this is the development environment used to code the frontend of the project. It is here that we imported all the modules provided by react native community, and where we also import all the library necessary for JavaScript.

BACKEND TOOLS

* **NODE JS**: This is where we installed and use expo go in our computer in other to be able to connect it to the EXPO app in our phone.
* **EXPO**: we installed this tool from PLAYSTORE into our android apps. It permits us to run and test the code as we are coding it. It is connected to the backend (node.js) and the backend is connected to the frontend (VS code).

DATABASE TECHNOLOGY AND SERVICES

* We are going to us a **SUPERBASE** which is actually a **software as a sevice** (SaaS).
* **SETUP**

FRONTEND

* Figma: figma was used online. Below is the link to our figma design

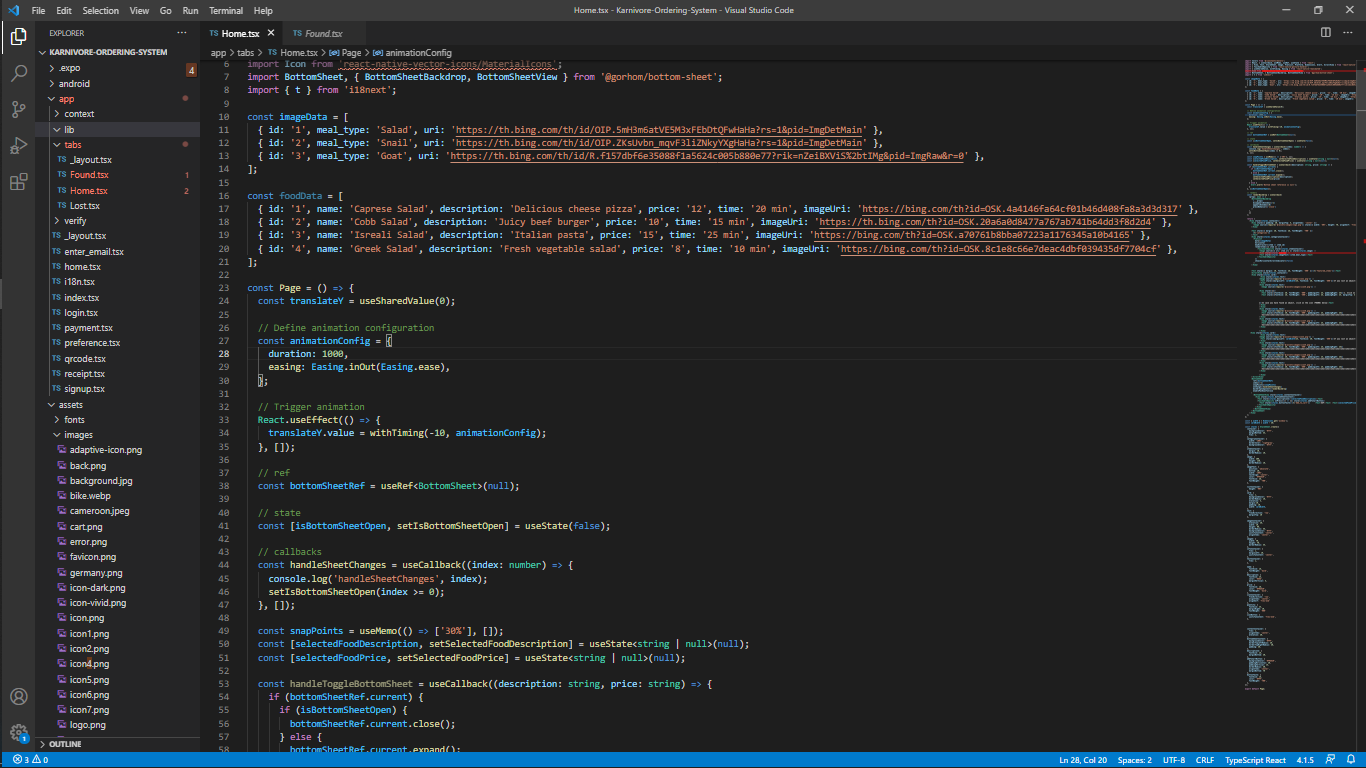
<https://www.figma.com/design/ZbvF7DZY4aXXE6JYWI3mwB/laure?node-id=0-1&t=JDXUnKMcwucVgegh-1>

* VS CODE: to install VS Code, you just need to download it online and click on the on the setup to download.
* Then install the node package by typing ‘npm install’.

BACKEND AND DATABASE

* NODEJS: go to your browser and search ‘nodejs install’, then click on the setup to install
* EXPO PACKAGE: go to the terminal run ‘npm install -g react-native-cli’, then run ‘npm i -g yarn’. Then run ‘’yarn expo go’’ to start the server.

## Code structure



The code was structured into several folder as seen above.

# SUMMARY

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| PART | | PROGRAMMING LANGUAGE | FRAMEWORK | TOOLS | USE |
| Frontend | | JavaScript | React native | VS Code | Contains all the package for react native and exp.  This is where all the codes for the UI elements are written |
| Backend | | JavaScript | Node.js | Node.js  Expo go | Nodejs is where all the packages are being downloaded  It is link to the frontend |
| Database | | SQL | Superbase | Superbase  Expo | Keeps all the images and associate each image to a finder information. |
| UI DESIGN | // | | // | FIGMA | It is use to design the UI element and visual design |

# 4. CONCLUSION

The implementation of a system for the retrieval and archival of missing objects using image matching algorithms relies on a robust combination of advanced technologies. From image recognition and database management to backend and frontend frameworks, each component plays a crucial role in ensuring the system's efficiency, scalability, and user-friendliness. By leveraging these technologies, the system can effectively manage and retrieve missing objects, providing a valuable solution for various applications.