NeuroMemorize

Submitted in partial fulfilment of the requirements

for the degree of

Bachelor of Engineering

by

|  |  |
| --- | --- |
| Gunjan Jhanwar | 60003160024 |
| Srushti Pathak | 60003160044 |
| Aziz Presswala | 60003160045 |

Project Guide:

Prof. Neha Katre



(Information Technology)

Dwarkadas J. Sanghvi College of Engineering

University of Mumbai

2019-20



CERTIFICATE

This is to certify that the project entitled “**NeuroMemorize**” is a bonafide work of

Gunjan Jhanwar (60003160024)

Srushti Pathak (60003160044)

Aziz Presswala (60003160045)

Submitted to the University of Mumbai in partial fulfilment of the requirement for the award of the degree of “**Bachelor of Engineering**” in “**Information Technology**”.

Prof. Neha Katre

Guide

Dr. Vinaya Sawant Dr. Hari Vasudevan

Head of Department Principal



Project Report Approval for B. E.

This project report entitled ***NeuroMemorize*** by ***Gunjan Jhanwar, Srushti Pathak, Aziz Presswala*** is approved for the degree of Information Technology.

Examiners 1. --------------------------------------------

2. ---------------------------------------------

Date:

Place:



Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Gunjan Jhanwar (60003160024) -----------------------------------

Srushti Pathak (60003160044) -----------------------------------

Aziz Presswala (60003160045) -----------------------------------

Date:

**Acknowledgement**

We would like to take this opportunity to thank our guide, Prof. Neha Katre for inspiring us and guiding us throughout this project. She imparted us with valuable guidance and timely support, without which we wouldn’t have come this far.

We would like to convey our heartfelt gratitude to our Principal, Dr. Hari Vasudevan and our Head of Department, Dr. Vinaya Sawant for all their support and encouragement.

We would like to wholeheartedly thank the entire staff of IT Department for their cooperation and help. We would like to thank the management of Dwarkadas J. Sanghvi College of Engineering for providing us with the infrastructure and facilities to complete this project.

Lastly we would like to thank all our loved ones for all their moral support and belief in us.

Gunjan Jhanwar, Srushti Pathak, Aziz Presswala

Dwarkadas J. Sanghvi College of Engineering

**Abstract**

Alzheimer’s disease is an irreversible, progressive brain disorder that slowly destroys memory and thinking skills and, eventually, the ability to carry out the simplest tasks [10]. Caretakers are very often expected to focus their attention on tackling the Alzheimer's patient’s distress or discomfort. Research has shown that methods of technology help to lower the anxiety and agitation levels of Alzheimer’s patients. The goal is therefore to provide a comprehensive and easy-to-use application that will help patients with Alzheimer's disease. Thus, the authors have implemented an application for Alzheimer’s patients that uses techniques such as Artificial Intelligence based Face Recognition, GPS based live location tracking, photo sharing and effective reminders that assist Alzheimer's patients maintain routine and remain attached and involved with their loved ones.

**List of Figures**

Figure 1: Task Network Diagram………………………………………………………………...23

Figure 2: Data Model…..………………………………………………………………………....28

Figure 3: Level 0 Data Flow Diagram…………………...………………………………….……28

Figure 4: Level 1 Data Flow Diagram………………………….…………………………….…..29

Figure 5: Level 2 Data Flow Diagram…………………………………………………….……...29

Figure 6: Use Case Diagram…………………………………………...……………….………...30

Figure 7: Activity Diagram…….…………………………………………………….…………...36

Figure 8: Deployment Diagram.…….…………………………………………….……………...37

Figure 9: System Architecture…………………………………………………….……………...38

Figure 10: Working of System……………………………………………………………….…...40

Figure 11: Login……………………………………………………………………….………....42

Figure 12: Users …………………………………………………………………….…………....42

Figure 13: Caretaker Details …………………………………………………..………………....42

Figure 14: Patient Details…………………………………..………………………………….....43

Figure 15: Caretaker Login………………………………………………………….…………....43

Figure 16: Caretaker Functions …………………………………………..……………………....43

Figure 17: Reminders pushed by Caretaker……………………………………………………....43

Figure 18: Location Tracking………………...…………………………..……………………....43

Figure 19: Updates Caretaker…………………………………………….……………………....43

Figure 20: Memory Sharing ……………………………………………...……………………....43

Figure 21: Setting Reminders………..…………………………………...……………………....44

Figure 22: Face Training (person List)…………………………………...……………………....44

Figure 23: Add person……………………………………………...……….…………………....44

Figure 24: Patient Receiving Updates………………………………...……….………………....44

Figure 25: Patient Receiving Reminders………………………………...…….………………....44

Figure 26: Call/Message Tab…………………………………………...………...……………....45

Figure 27: Patient Profile……………………………………………...…………….…………....45

Figure 28: Face Recognition……………………………………………...……………………....45

Figure 29: Final Recognized Image…………………………………...……………….………....49

Figure 30: Tracking……………………………………………...…………………………….....49

**List of Tables**

Table 1: Project Schedule………………………………………………………………………...24

Table 2: COCOMO Values ………………………………………….…………………………...26

Table 3: Risk Management Mitigation Planning…………….…………………………………...27

Table 4: Use case - Register/Login…………………………….………………………………....31

Table 5: Use case - Information……….…………………………….…………………………....32

Table 6: Use case - Reminders…………………………………………………………………....32

Table 7: Use case - Real time Face Recognition……………………………….………………....33

Table 8: Use case - Location Tracker ………………………………………….………………...34

Table 9: Use case - Check Connection with Wi-fi or Battery Status……………………………...34

Table 10: Data Structures for Azure Face API…………………………………………………....40

Table 11: Test Cases……………………………………………………………………………...47

Table of Contents

Page number

1. [**Introduction** 13](#_Toc39093319)

[1.1 Motivation / Objective 13](#_Toc39093320)

[1.2 Major Challenges 13](#_Toc39093321)

[1.3 Report Overview 14](#_Toc39093322)

1. [**Literature Review** 15](#_Toc39093323)

[2.1 Existing Work 15](#_Toc39093324)

[2.1.1 Literature Related to Existing Systems 15](#_Toc39093325)

[2.1.2 Literature Related to Methodology 16](#_Toc39093334)

[2.1.3 Literature Related to Algorithms 18](#_Toc39093335)

[2.1.4 Literature Related to Technology / Tools / Frameworks 19](#_Toc39093336)

[2.2 Observations on Existing Work 20](#_Toc39093338)

1. [**Our Methodology** 22](#_Toc39093339)

[3.1 Problem Definition 22](#_Toc39093340)

[3.2 Scope 22](#_Toc39093341)

1. [**Project Management** 25](#_Toc39093345)

[4.1 Project Schedule 25](#_Toc39093346)

[4.1.1 Task Network Diagram 25](#_Toc39093347)

[4.1.2 Timeline Chart 26](#_Toc39093348)

[4.2 Feasibility Study 26](#_Toc39093349)

[4.2.1 Technical Feasibility 26](#_Toc39093350)

[4.2.2 Operational Feasibility 27](#_Toc39093351)

[4.2.3 Economic Feasibility 27](#_Toc39093352)

[4.3 Project Resources 27](#_Toc39093353)

[4.3.1 Hardware Requirements 27](#_Toc39093354)

[4.3.2 Software Requirements 27](#_Toc39093355)

[4.3.3 Operating Requirements 27](#_Toc39093356)

[4.4 Project Estimation 28](#_Toc39093357)

[4.4.1 COCOMO Estimation Model 28](#_Toc39093358)

[4.4.2 Function Point Analysis 29](#_Toc39093359)

[4.5 Risk Management Mitigation Planning 29](#_Toc39093360)

1. [**System Design** 30](#_Toc39093361)

[5.1 Design Diagrams 30](#_Toc39093362)

[5.1.1 Data Model 30](#_Toc39093363)

[5.1.2 DFD 30](#_Toc39093364)

[5.1.3 UML Diagrams 32](#_Toc39093365)

[5.2 Proposed System Architecture 38](#_Toc39093366)

[5.3 Proposed Tools and Technology 39](#_Toc39093367)

1. [**Implementation** 41](#_Toc39093368)

[6.1 Working of System 41](#_Toc39093369)

[6.2 Algorithms and Tools used 41](#_Toc39093370)

[6.3 Interface Design 44](#_Toc39093378)

1. [**Testing and Results** 48](#_Toc39093379)

[7.1 Test Plan 48](#_Toc39093380)

[7.2 Test Cases 49](#_Toc39093381)

[7.3 Testing Methods Used 50](#_Toc39093382)

[7.4 Experimental Results and Conclusions 50](#_Toc39093383)

1. [**Conclusion** 52](#_Toc39093384)
2. [**Future Scope** 53](#_Toc39093385)

[**References** 54](#_Toc39093386)

**Chapter 1**

# **Introduction**

## **1.1 Motivation / Objective**

Alzheimer’s disease is an irreversible disease of the brain that affects a person’s memory, thinking, and other abilities. It is progressive, meaning symptoms get worse and more functions are lost the longer an individual has Alzheimer’s. As the disease progresses, Alzheimer's symptoms become even more prominent and may include problems with processing new information, trouble concentrating, difficulty recalling the names of objects and important dates, losing things and being unable to locate them, poor judgment, detachment from social activities and change of mood / personality [12].

Worldwide, at least 50 million people are believed to be living with Alzheimer's disease or some form of Dementia. In India, more than 4 million people have some form of Dementia. By 2050, nearly 131.5 million people worldwide could be living with the disease, unless scientists develop new approaches to prevent or cure it [1].

As mentioned earlier, Alzheimer's is a medical condition, with symptoms gradually worsening over a number of years. In its early stages, memory loss is mild but individuals lose the ability to carry on a conversation and adapt to their environment with late-stage Alzheimer's. Over the years, research and advisory initiatives from around the world have created a new market for software developers and professionals in the health care sector. More use of the technology by older adults has also brought about a new age. Therefore, technology can be used for both people suffering from Alzheimer’s and their loved ones who care for them.

So in order to help the Alzheimer’s patients, an application is proposed that uses latest technologies such as Artificial Intelligence based Face Recognition. The application aims to help people with Alzheimer relieve fear, develop rhythm, improve their quality of life, and easily connect with their family and friends.

## **1.2 Major Challenges**

Those who develop dementia are faced with a range of complex challenges caused by symptoms, such as memory loss and confusion, which can result from the condition. With advances in medicine and technology the number of older adults and life expectancy is increasing. From problems with spatial orientation to difficulties with social interaction, the consequences of dementia can be broad and encompass many aspects of daily life. But these are by no means too great to overcome. There are many ways that those living with dementia can work through the problems that they come up against, with the support of family members, friends and caretakers, allowing them to live happy, fulfilled lives.

Since the software would be largely used by people with a non-technological background, a simple, convenient-to-use interface for the device is developed. This system can be effectively in a number of use-cases, and the user does not feel lost when using the system. It is considered when planning the front end, the fact that most of the users will be technologically illiterate.

## **1.3 Report Overview**

The rest of the report is organized as follows. Chapter 2 covers the Literature Survey with detailed discussion on analysis of Existing Systems, different methodologies used for Face Recognition as well as the Algorithms that are currently used for Face Recognition and their comparative study. Chapter 3 outlines the problem definition, our approach of developing the proposed system, scope and features of the system. The Project Schedule, Feasibility Study, Resources used, Project Estimation and Risk Management & Mitigation planning are presented in Chapter 4. Chapter 5 provides in detail description of system design through ER (Entity Relationship) Model, DFD (Data Flow Diagram), UML diagrams, System Architecture and Proposed tools. Chapter 6 explains complete working of our system along with individual modules through algorithms/tools used and interface design. Chapter 7 provides the Test cases, Test plan of the project, Testing parameters along with experimental results followed by conclusions and references. Chapter 8 and Chapter 9 consists of Conclusion and Future Scope of the application respectively. Chapter 10 mentions the details of the papers published regarding the system.

**Chapter 2**

# **Literature Review**

## **2.1 Existing Work**

### **2.1.1 Literature Related to Existing Systems**

Currently very few applications are available for Alzheimer’s patients and their caretakers. They have made it easier for patients, particularly in earlier stages, to communicate and pass the time. They are discussed in brief below:

**1. MindMate**

The MindMate app has daily activities and games designed to improve brain health, and can be customized for your loved one depending on the level of thinking impairment. The app has mental exercises and even physical exercises, and tracks progress. It includes nutrition advice and clips of classic movies, music, and TV from decades past.

## **2. Lumosity Mobile**

## The Lumosity application features a combination of more than 25 cognitive games aimed at "exercising" the brain. The games test memory and attention by getting the user interested in specific cognitive and neuropsychological activities.

## **3. It's Done**

## It's Done is an application that provides a checklist for life's everyday critical tasks such as locking doors, feeding pets, taking medication, and turning off the stove. The It's Done application is suitable for those with short-term memory loss due to brain disorder, Alzheimer’s or other medical conditions.

## **4. Timeless**

## Timeless is a simple, easy-to-use application for Alzheimer's patients to recall things, stay connected and interact with friends and family, and recognize people using Face Recognition technology based on Artificial Intelligence.

**5. Carezone**

Carezone is a helpful tool for managing all medical care, useful for someone in the early stages of dementia and caregivers during the later stages. The app easily organizes lists of medications, sets reminders of when to take them, orders refills, and keeps track of doctor appointments

### **2.1.2 Literature Related to Methodology**

Alzheimer’s can be frightening and overwhelming - for both caretakers and people living with the disease. However, new technologies can help ease anxiety, establish routine and improve the quality of life for everyone involved. Here is a list of the technologies for caretakers and those living with Alzheimer’s today:-

1. **Photo Albums -** Photo albums hold pictures of loved ones, each with a short description of the photo along with the date on which the photo was captured. This can be a great way to remember loved ones or fun times in the past and help patients to ease anxiety & agitation.
2. **Reminders** - Reminders can help keep assets and loved ones safe when the caregiver can't. Reminders can be recorded / set on the computer and played back loud at the appropriate time. For example, a caregiver may record a video message that advises a person to take a prescription at the right time.
3. **Location Tracking - As the disease progresses,** wandering becomes a problematic behavior, which can occur due to a variety of reasons. Location tracking is the answer. They include GPS devices, so the patient can easily be located and avoids getting lost. These devices can also be programmed to alert caretakers if the patient goes outside a specific area.
4. **Picture-based Phonebook -** They are reprogrammed with important phone numbers, so that the patient need not remember them or have them written down. It features actual pictures of loved ones along with large, easy-to-use buttons corresponding to each person.
5. **Music** - Music helps healing, so any system designed to play favorite songs for someone with Alzheimer’s can be genuinely beneficial. After 20 minutes of listening to music, people with Alzheimer’s in one study saw an immediate, measurable increase in happiness, eye contact, and talkativeness, and a decrease in fatigue.
6. **Face Recognition** - As discussed earlier, Alzheimer’s patients tend to forget their family, friends and loved ones. Therefore, Artificial Intelligence based Face Recognition can be used in order to help them recall their loved ones.

### **2.1.3 Literature Related to Algorithms**

One of the significant features of the application is the Artificial Intelligence based Face Recognition. In order to implement this technology in our application we did extensive research and survey of the Services/API’s available which can be easily integrated with the application. All these Services/API’s are described below:-

1. **Microsoft Azure Face API**

The Azure Cognitive Services Face technology offers algorithms for the identification, classification, and interpretation of human faces in images. In several different software scenarios the capacity to interpret information about the human face is essential. Examples of these scenarios are security, natural user interface, analysis and management of image content, mobile apps and robotics. The Identify API is used to identify a face detected against a human database. This function could be useful in image processing applications for automated labelling of images. [9]

The advantage is that it has a wider customer base, which alone will improve a product's availability through market demand and a response from manufacturers.

1. **OpenCV**

OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. OpenCV has a modular structure, which means that the package includes several shared or static libraries. The following modules are available [10]:-

* **core** - a compact module defining basic data structures, including the dense multi-dimensional array Mat and basic functions used by all other modules.
* **imgproc** - an image processing module that includes linear and non-linear image filtering, geometrical image transformations (resize, affine and perspective warping, generic table-based remapping), color space conversion, histograms, and so on.
* **video** - a video analysis module that includes motion estimation, background subtraction, and object tracking algorithms.
* **calib3d** - basic multiple-view geometry algorithms, single and stereo camera calibration, object pose estimation, stereo correspondence algorithms, and elements of 3D reconstruction.
* **features2d** - salient feature detectors, descriptors, and descriptor matchers.
* **objdetect** - detection of objects and instances of the predefined classes (for example, faces, eyes, mugs, people, cars, and so on).
* **highgui** - an easy-to-use interface to video capturing, image and video codecs, as well as simple UI capabilities.

1. **Amazon Rekognition**

Amazon Rekognition makes it easy to add image and video analysis to your applications using proven, highly scalable, deep learning technology that requires no machine learning expertise to use. With Amazon Rekognition, you can identify objects, people, text, scenes, and activities in images and videos, as well as detect any inappropriate content. Amazon Rekognition also provides highly accurate Face analysis and Face search capabilities that you can use to detect, analyze, and compare faces for a wide variety of user verification, people counting, and public safety use cases.

With Amazon Rekognition Custom Labels, you can identify the objects and scenes in images that are specific to your business needs. For example, you can build a model to classify specific machine parts on your assembly line or to detect unhealthy plants. Amazon Rekognition Custom Labels takes care of the heavy lifting of model development for you, so no machine learning experience is required. You simply need to supply images of objects or scenes you want to identify, and the service handles the rest. [11]

After exhaustive literature survey of all the existing methodologies that are used for Face Recognition, we found Microsoft Azure Face API the most suitable for our application considering its high accuracy & low latency time. Therefore, the application uses Microsoft Azure Face API for face recognition.

### **2.1.4 Literature Related to Technology / Tools / Frameworks**

The applications that are currently available in the market for Alzheimer’s patients are either Paid or only available for iOS users. Considering the fact that a large part of Indian mobile market is dominated by Android users, therefore, the application is designed & developed in **Android** so that the application can reach a wider audience.

Some of the primary tools and API’s used to develop the application are discussed in brief below:

## **Android Studio -** is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is a replacement for the Eclipse Android Development Tools (ADT) as the primary IDE for native Android application development.

1. **Firebase Realtime Database** - Firebase provides a real-time database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud.

#### **Firebase Auth** - Firebase Auth is a service that can authenticate users using only client-side code. It supports [social login providers](https://en.wikipedia.org/wiki/Social_login) Facebook, GitHub, Twitter and Google (and [Google Play Games](https://en.wikipedia.org/wiki/Google_Play_Games)). Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase.

1. **Microsoft Azure Face API** - The Azure Cognitive Services Face service provides algorithms that are used to detect, recognize, and analyze human faces in images. The ability to process human face information is important in many different software scenarios. Example scenarios are security, natural user interface, image content analysis and management, mobile apps, and robotics. The Identify API is used to identify a detected face against a database of people. This feature might be useful for automatic image tagging in photo management software. [9]
2. **Google Maps API -** With the Maps SDK for Android, maps based on Google Maps data can be added to the application. The API automatically handles access to Google Maps servers, data downloading, map display, and response to map gestures. API calls can also be used to add markers, polygons, and overlays to a basic map, and to change the user's view of a particular map area. These objects provide additional information for map locations, and allow user interaction with the map.
3. **Firebase Storage -** Cloud Storage is built for app developers who need to store and serve user-generated content, such as photos or videos. Cloud Storage for Firebase is a powerful, simple, and cost-effective object storage service built for Google scale. These SDKs can be used to store images, audio, video, or other user-generated content.

## **2.2 Observations on Existing Work**

After exhaustive survey of existing systems, it is conclude that these systems lack in offering some of the key features. Some of the key observations are listed below:

**1. Existing Systems**

Currently only one app is available for Alzheimer’s patients that provides customer with the features which we are aiming to provide but it is still in development & is only available for iOS platform.

**2. More Features**

There is a need of an application that provides complete assistance to an Alzheimer’s patient which current apps fail to offer.

**3. Availability in Android**

Currently only one app is available for Alzheimer’s patients that provides all the features which we are aiming to provide but this app will only available for iOS platform. Since Indian mobile market is dominated by Android, apps need to be available in Android platform for wider reach.

**4. Ease of Use**

Since most of the Alzheimer’s patients are above the age group of 50 and above, who are not comfortable with using & adopting new technologies, app is designed to provide more accessibility & ease of use.

**Chapter 3**

# **Our Methodology**

## **3.1 Problem Definition**

Alzheimer’s disease is an irreversible, progressive brain disorder that slowly destroys memory and thinking skills and, eventually, the ability to carry out the simplest tasks. As the disease progresses, patients start to lose track of time and have difficulty in managing their daily activities and have difficulty in recognizing close people. From problems with difficulties with social interaction, the consequences of dementia can be broad and encompass many aspects of daily life. But these are by no means too great to overcome. There are many ways that those living with dementia can work through the problems that they come up against, with the support of family members, friends and caretakers, allowing them to live happy, fulfilled lives. [2]

The use of the brain games and other computer-based stimulation therapy has given new light to healthcare professionals, caregivers, and the patients themselves. Research and therapy programs from around the world have created a new market for healthcare professionals and software developers. The increased use of the Internet by older adults has also spawned a new age. Previously, older adults and people with disabilities often struggled to cope with the demand integration of IT into their daily lives. For many older adults complex menus, small fonts, and the fear of an unwanted feedback from a device or the Internet caused many to avoid these technologies all together. Hence, we provide a complete application which can help the Mild and Moderate Stage Alzheimer’s patients to remember their relatives, family and friends.[2]

## **3.2 Scope**

Our application mainly focuses on patients which are having Mild and Moderate stages of Alzheimer’s disease. The stages of the disease are explained below:

## **Mild Alzheimer's disease (early stage)**

In the early stage of Alzheimer's, a person may function independently. He or she may still drive, work and be part of social activities. Despite this, the person may feel as if he or she is having memory lapses, such as forgetting familiar words or the location of everyday objects.

## **Moderate Alzheimer's disease (middle stage)**

Moderate Alzheimer's is typically the longest stage and can last for many years. As the disease progresses, the person with Alzheimer's will require a greater level of care.  
During the moderate stage of Alzheimer’s, the dementia symptoms are more pronounced. A person may have greater difficulty performing tasks, such as paying bills, but they may still remember significant details about their life.

You may notice the person with Alzheimer's confusing words, getting frustrated or angry, or acting in unexpected ways, such as refusing to bathe. Damage to nerve cells in the brain can make it difficult to express thoughts and perform routine tasks.

## **Severe Alzheimer's disease (late stage)**

In the final stage of this disease, dementia symptoms are severe. Individuals lose the ability to respond to their environment, to carry on a conversation and, eventually, to control movement. They may still say words or phrases, but communicating pain becomes difficult. As memory and cognitive skills continue to worsen, significant personality changes may take place and individuals need extensive help with daily activities.

**3.2.1 Assumptions and Constraints**

* Usage of technology by Alzheimer patients is a challenging task. Previously, older adults and people with disabilities often struggled to cope with the integration of technology into their daily lives. But repititional practice can help them use it very effectively.
* This application is developed only for Mild and Moderate stage patients because they can perform their daily routine task and remember few things.
* For Severe stage patients, even the medicines don’t work and there is complete memory loss. Technology can help a little but does not work effectively.

**3.3 Our Approach to build NeuroMemorize**

**3.3.1 Features of Proposed System**

Following are the key features of the system:

* Inviting friends and family to send an initial set of photos that the Face recognition system can use to learn & identify them.
* The app will tell who the person is and how they are related to the patient using Real-Time Face recognition.
* A simple reminder screen that lists daily tasks.
* Live location tracking of the patient’s phone & notify the caregiver if the patient’s device is out of range.
* Notify the family member/caretaker if the patient’s phone is not connected to Wi-Fi or the battery is low.
* A picture-based phone book which enables the patient to call or text a person.
* A “Me” page shows the patient name, photo, phone number, email and address.
* If somebody is mild in the disease, and with the support of the caregiver, it is possible that if the app is simple enough that they can learn to use it through repetitional practice.

**Chapter 4**

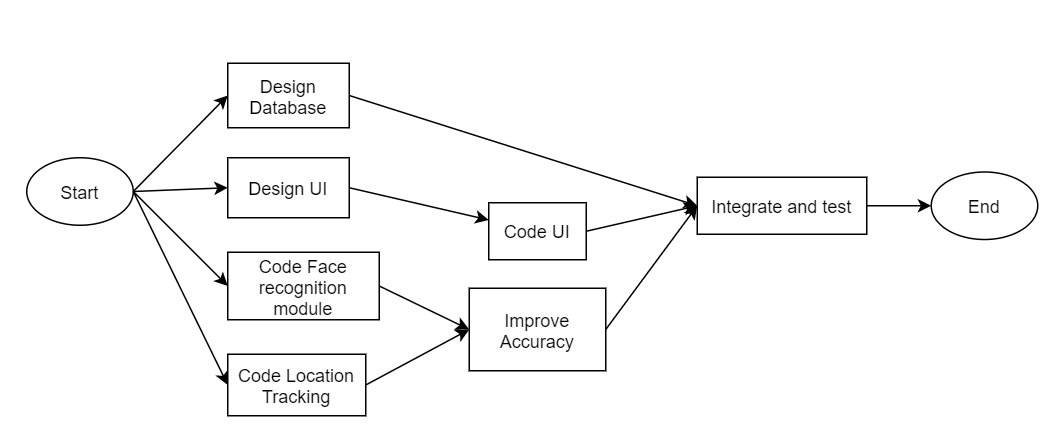
# **Project Management**

## **4.1 Project Schedule**

### **4.1.1 Task Network Diagram**

The task network diagram starts with creation of these modules:

1. UI Design – This module includes authentication of users by logging them in to the application. Firebase is used for authentication purposes. The caretaker will enter details for the patient and submit. Patient will scan the QR code and they both will get linked.
2. Face Recognition and Location Tracker module – These modules need high accuracy and hence more time is given in its creation.
3. Group images by people module – This module requires face recognition from the images and then group images by people for patient to easily get the details.



**Figure 1: Task Network Diagram**

### **4.1.2 Timeline Chart**

The plan for the development of the project is given as follows:

Table 1: Project Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr.no. | Task Name | Duration | Start Date | End Date |
| 1 | Existing System Study | 15 days | 10/01/19 | 25/01/19 |
| 2 | Software Requirement | 7 days | 27/01/19 | 02/02/19 |
| 3 | Research and Literature Review | 15 days | 03/02/19 | 18/02/19 |
| 4 | Feasibility Study | 7 days | 19/02/19 | 26/02/19 |
| 5 | Planning and Design | 5 days | 27/02/19 | 04/03/19 |
| 6 | Login for users | 5 days | 05/03/19 | 09/03/19 |
| 7 | Authentication and Linking | 5 days | 10/03/19 | 15/03/19 |
| 8 | Face Recognition module | 20 days | 16/03/19 | 05/04/19 |
| 9 | Location Tracking Module | 15 days | 01/07/19 | 16/07/19 |
| 10 | Reminders module | 4 days | 17/07/19 | 20/07/19 |
| 11 | Notification module | 5 days | 24/07/19 | 30/07/19 |
| 12 | Phone book and messaging module | 4 days | 06/08/19 | 10/08/19 |
| 13 | Improving accuracy of recognizer | 10 days | 15/08/19 | 25/08/19 |
| 14 | UI implementation | 5 days | 30/08/19 | 04/09/19 |
| 15 | Integration of modules | 35 days | 16/09/19 | 21/10/19 |
| 16 | Preparing Test Cases | 5 days | 10/01/20 | 15/01/20 |
| 17 | Unit Testing | 10 days | 20/01/20 | 30/01/20 |
| 18 | Formulating Changes | 10 days | 01/02/20 | 10/02/20 |
| 19 | Testing on Patient | 30 days | 11/02/20 | 12/03/20 |
| 20 | Documentation | 7 days | 13/03/20 | 20/03/20 |
| 21 | Final Project Closure | 3 days | 21/03/20 | 24/03/20 |
| 22 | Presentation | 3 days | 07/04/20 | 10/04/20 |

## **4.2 Feasibility Study**

### **4.2.1 Technical Feasibility**

Technical Feasibility is analysis or research into the practicality of a proposed plan or method, including whether it can be delivered with the available technology, techniques, skills and resources (human and financial). A technical feasibility assessment will therefore focus on 'can it be done'.

Technical feasibility in our project is:

* This system is an Android Application.
* Will run in 2GB RAM or above
* Android version should be higher than Lollipop

### **4.2.2 Operational Feasibility**

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfils the requirements as identified during the development of the project.

Operational feasibility in our project is:

* The system uses Real-time Face Recognition hence reducing the time taken for processing the images.
* It also includes in-built location tracking feature which can track the patient and update it to caretaker after every 10 seconds.

### **4.2.3 Economic Feasibility**

Analysis of a project’s costs and revenues in an effort to determine whether or not it is logical and possible to complete is called Economic Feasibility.

Economic feasibility in our project is:

* Our system can run in any device of moderate processing power which makes it cost-effective (zero-cost and freely available for download)

## **4.3 Project Resources**

### **4.3.1 Hardware Requirements**

Device configuration:

* Minimum 4GB RAM
* 32GB secondary storage

### **4.3.2 Software Requirements**

* Android Operating System (OS) higher than Lollipop.

### **4.3.3 Operating Requirements**

* Constant Internet Connection
* GPS permission
* Contacts permission

## **4.4 Project Estimation**

### **4.4.1 COCOMO Estimation Model**

Basic COCOMO Model is taken into consideration to estimate the development time of our project. Evaluation the parameters like the size of code, team size, developer experience, working environment, innovation and deadline it can be said that organic development model is being followed as per COCOMO model.

**Estimation Technique**

The COCOMO cost estimation model is used by thousands of software project managers, and is based on a Study of hundreds of software projects. It is applied to three classes of software projects: Organic, Semi-detached and Embedded systems. Unlike other cost estimation models, COCOMO model is an open model, so all the details are published including:

* The underlying cost estimation equations.
* Every assumption made in the model (e.g. “the project will enjoy good management")
* Every definition (e.g. “the precise definition of the Product Design phase of a project)
* The costs included in an estimate are explicitly Stated (e.g. project managers are included, secretaries aren't).

Table 2: COCOMO Values

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Software Project** | **ab** | **bb** | **cb** | **db** |
| **Organic** | 2.4 | 1.05 | 2.5 | 0.38 |
| **Semi detached** | 3.0 | 1.12 | 2.5 | 0.35 |
| **Embedded** | 3.6 | 1.20 | 2.5 | 0.32 |

**Estimate for Effort, Cost and Duration**

We estimate our project to be of 12 KLOC (kilo lines of code) 1.0

Having the following details as COCOMO model’s guidelines, we get:

**Effort** = 2.4 \* (12)1.05 = 33 person-months

**Development Time** = 2.5 \* (33)0.38 = 9 months

**People** = Effort/ Development time = 33/9 = 3.667 = 3 persons (approx.)

Thus, from COCOMO model we can estimate that the project can be completed in 9 months as per our project management schedule.

### **4.4.2 Function Point Analysis**

For function point analysis, we have following values for our project:

User input =50

User output = 70

User enquires = 10

User files = 10

External interfaces = 9

Also, we have Complexity Adjustment Factor (CAF) and all weighing factors as average.

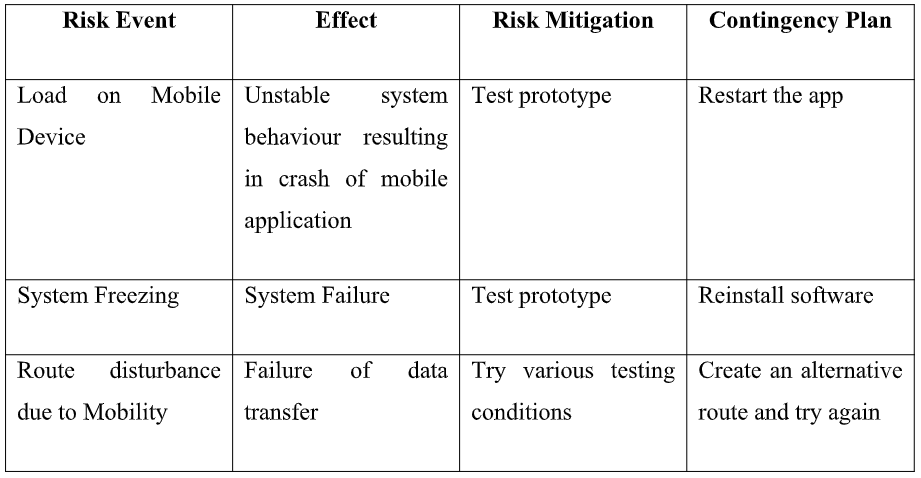
So we can calculate Functional points as Unadjusted Functional Point (UAF) = 50\*4 +70\*5 +10\*4+ 10\*10 + 9\*7 = 753

Complexity Adjustment Factor (CAF) = 0.65 + (0.01 \* 42) = 1.07

Functional points = 753 \* 1.07 = 805.71

## **4.5 Risk Management Mitigation Planning**

**Table 3: Risk Management Mitigation Planning**

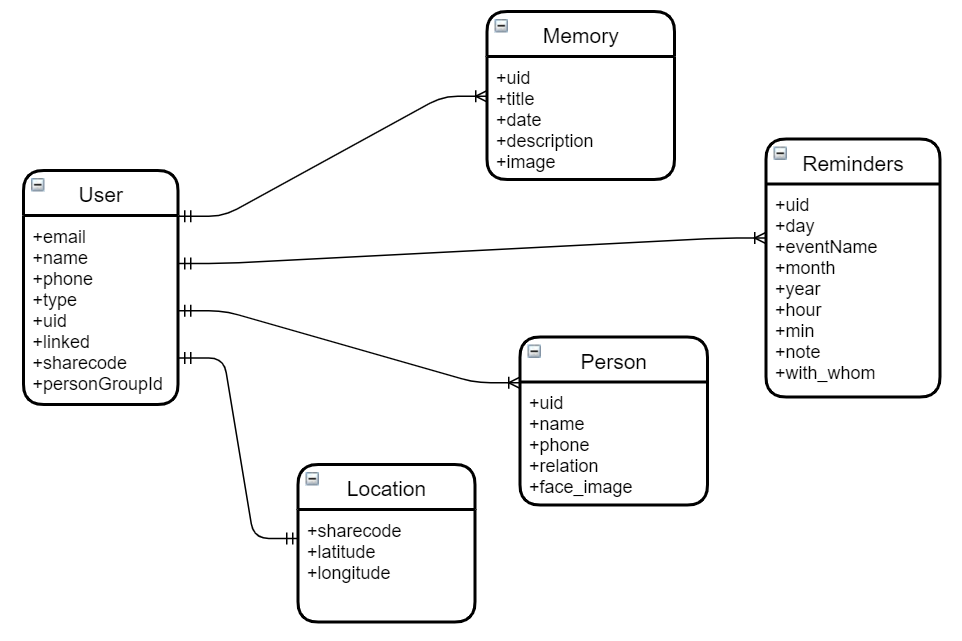


**Chapter 5**

# **System Design**

## **5.1 Design Diagrams**

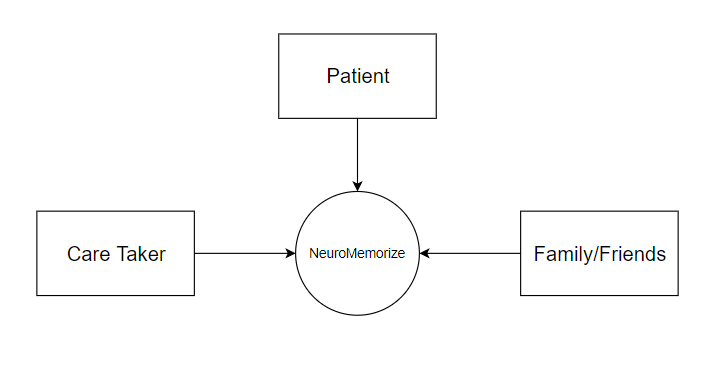
### **5.1.1 Data Model**



**Figure 2: Data Model**

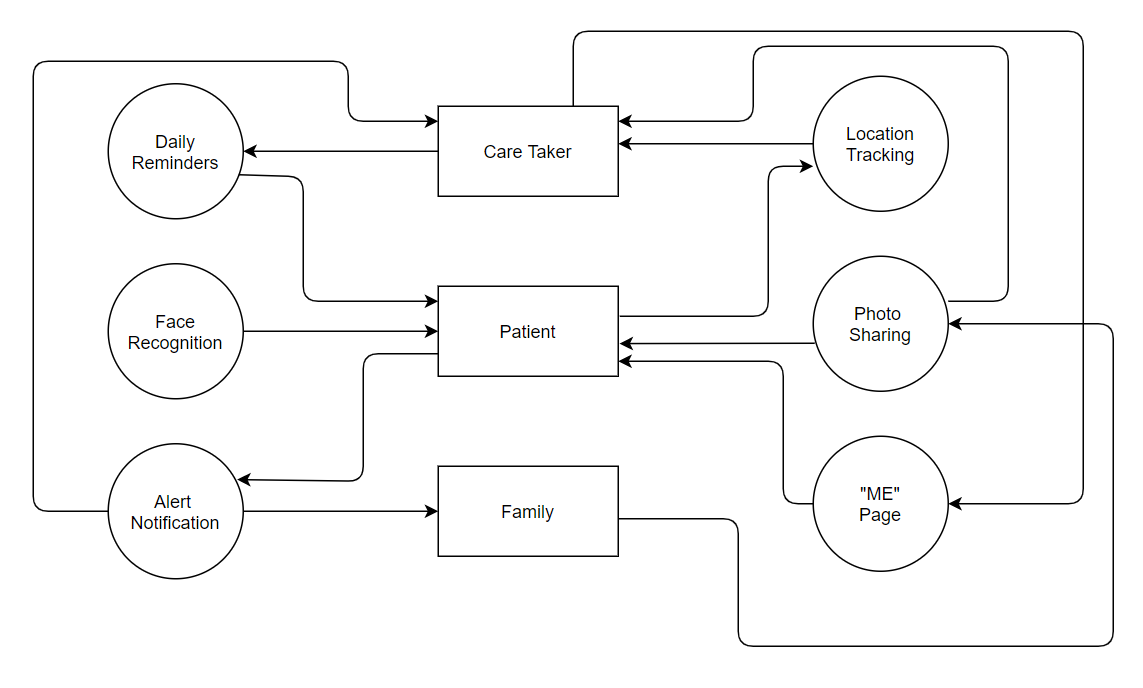
### **5.1.2 DFD**

**Level 0**

****

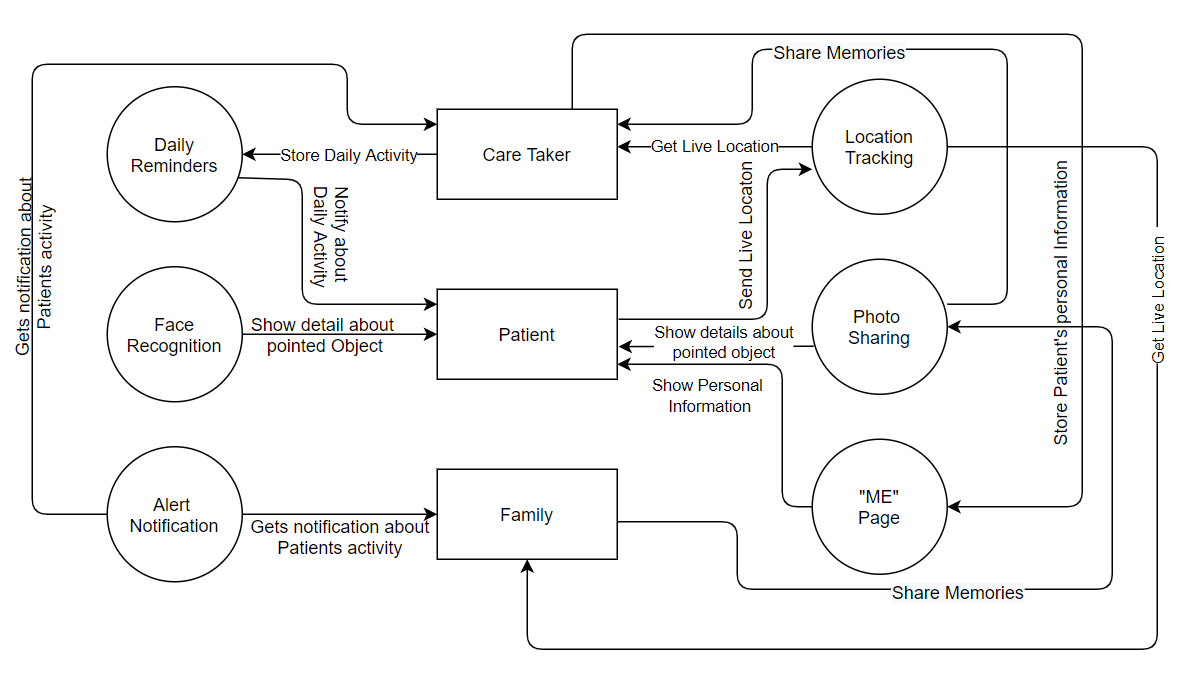
**Figure 3: Level 0 Data Flow Diagram**

**Level 1**



**Figure 4: Level 1 Data Flow Diagram**

**Level 2**



**Figure 5: Level 2 Data Flow Diagram**

The data flow diagram of our system is as shown above. There are three entities namely, Caretaker, Patient, and Family. There is a server where all the information of the entities will be stored. There are 6 functions which are shown in the diagram. They are – Location update, Information, Notification, Reminder Screen, Real time face recognition, and profile page.

1) Caretaker – The caretaker can get the live location of the patient and retrieve information about patient. They can also store daily activities of the patient which will be stored in the server.

2) Patient – The patient can be tracked by the caretaker by sending live location to the caretaker. He/she gets notified about their daily activities. They can recognize people by just pointing camera to the person and can get details about the person like name and relation with the patient. Their personal information will get stored in the “Me” page in the application.

3) Family – They can track the patient’s live location and gets notified about the patient’s activity.

### **5.1.3 UML Diagrams**

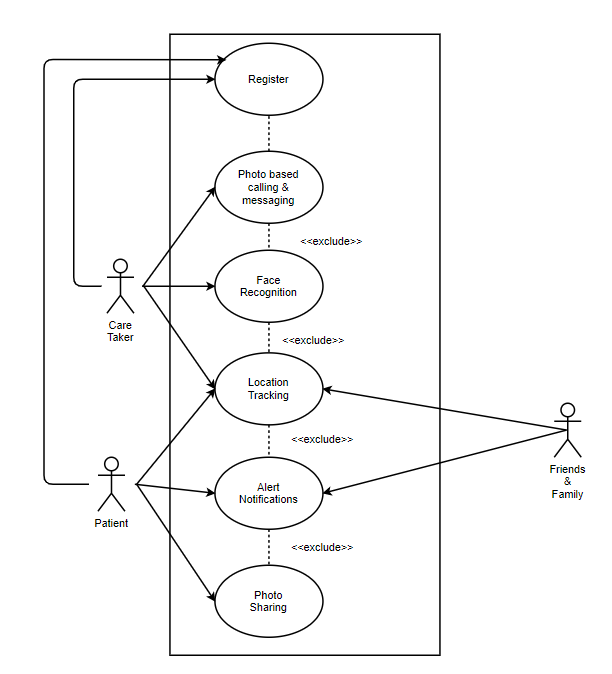
**5.1.3.1 Use Case Diagram**

The use case diagram for the application is given as shown in fig. The users of the system are: caretaker, family and friends and the patient.

**1) Caretaker -** The caretaker logins into the app and enters information about the patient. The caretaker can set reminders for the patient such as daily routine tasks. He/She can also take images of the patient’s related things like friends, family, objects, things of daily purpose use, etc.

**2) Family and Friends -** Family and friends of the patient logs in to the app and also can set reminders for the patient like taking medicines and other daily routine tasks. They can share their photos with the patient for them to recognize their closed ones. They will receive a notification if the patient’s phone is not connected to Wifi or the battery is low. They can take further decision according to this.

**3) Patient -** Patient logs in to the app and receives alarms and reminders which are set by the caretaker and closed ones. The patient takes Real-time images of their closed ones and their things which will be recognized and the details of the captured image will be displayed to the patient. If he/she gets lost or goes out of safe zone set by the caretaker then a notification will go to the caretaker as well as family members. The patient will be tracked wherever he/she goes.



**Figure 6: Use Case Diagram**

The use case diagram for all the use cases are:

**Table 4: Use Case – Register/Login**

|  |  |
| --- | --- |
| Use case ID | Register/Login on App |
| XRef |  |
| Trigger | User clicks on Register/Login Button |
| Precondition | User is on main screen of the app |
| Basic Path | 1. On button click, user is taken to respective screen and prompts user to fill the details |
| Alternative Paths | There are no alternative paths. To start the app, user must login in the app. |
| Postcondition | Taken to main screen of app. |
| Exception Paths | The user may close the app at any time |
| Other |  |

**Table 5: Use Case – Information**

|  |  |
| --- | --- |
| Use case ID | Enter Information |
| XRef |  |
| Trigger | User has completed the signup process. |
| Precondition | User is on main screen of the app and is new user |
| Basic Path | 1. Complete the signup process and then the caretaker is taken to the information page. |
| Alternative Paths | There are no alternative paths. It is mandatory to fill the basic information of the patient. |
| Postcondition | Taken to the home page of the application. |
| Exception Paths | The user may close the app at any time |
| Other |  |

**Table 6: Use Case – Reminders**

|  |  |
| --- | --- |
| Use case ID | Daily Activity Reminder |
| XRef |  |
| Trigger | Caretaker has uploaded some activity which Patient has to complete. |
| Precondition | User and caretaker have completed login process and are connected. |
| Basic Path | 1. Caretaker is on activity upload screen and uploads the daily task of patient. 2. Patient gets notification about performing the activity at a particular time. 3. After receiving notification patient can either snooze or stop the notification of activity. |
| Alternative Paths | There are no alternative paths. |
| Postcondition | Daily task are uploaded. |
| Exception Paths | The user may close the app at any time |
| Other |  |

**Table 7: Use Case – Real time face recognition**

|  |  |
| --- | --- |
| Use case ID | Real time face recognition. |
| XRef |  |
| Trigger | Patient can recognize any object or person by pointing camera at it. |
| Precondition | Patient have login in the app.  The pointing object or person should be present in database. |
| Basic Path | 1. Caretaker/Patient has already uploaded the information about the person or object. 2. When patient wants to recognize about the person or object patient will point the camera to that thing and can get information. |
| Alternative Paths | There are no alternative paths. |
| Postcondition | Patient will get information about the pointed object or person. If the pointed object or person isn’t present in database patient can add it. |
| Exception Paths | The user may close the app at any time |
| Other |  |

**Table 8: Use Case – Location Tracker**

|  |  |
| --- | --- |
| Use case ID | Location Tracker |
| XRef |  |
| Trigger | Caretaker wants to know where the Patient is. |
| Precondition | Patient and caretaker have login and are in sink.  Patients GPS is on and the application is running in the background. |
| Basic Path | 1. Caretaker is on location tracker page and can track patient’s location. |
| Alternative Paths | There are no alternative paths. |
| Postcondition | Patient is been tracked. |
| Exception Paths | The user may close the app at any time |
| Other |  |

**Table 9: Use Case – Check Connection with Wi-Fi or battery status**

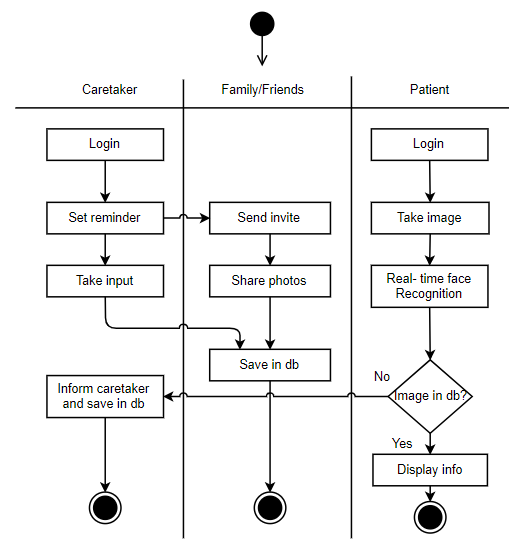
|  |  |
| --- | --- |
| Use case ID | Reminder to caretaker if patients cell’s battery is low/not connected to Wi-Fi |
| XRef |  |
| Trigger | Patient’s cell’s battery is low or it isn’t connected to internet. |
| Precondition | Patient and caretaker have login and are in sink.  Patients GPS is on and the application is running in the background. |
| Basic Path | 1. Caretaker is on location tracker page and can track patient’s location. |
| Alternative Paths | There are no alternative paths. |
| Postcondition | Patient is been tracked. |
| Exception Paths | The user may close the app at any time |
| Other |  |

**5.1.3.2 Activity Diagram**

The diagram below shows the activity diagram of the application. The flow of the diagram is shown as three swim lanes i.e. Caretaker, Family/ friends, and Patient.

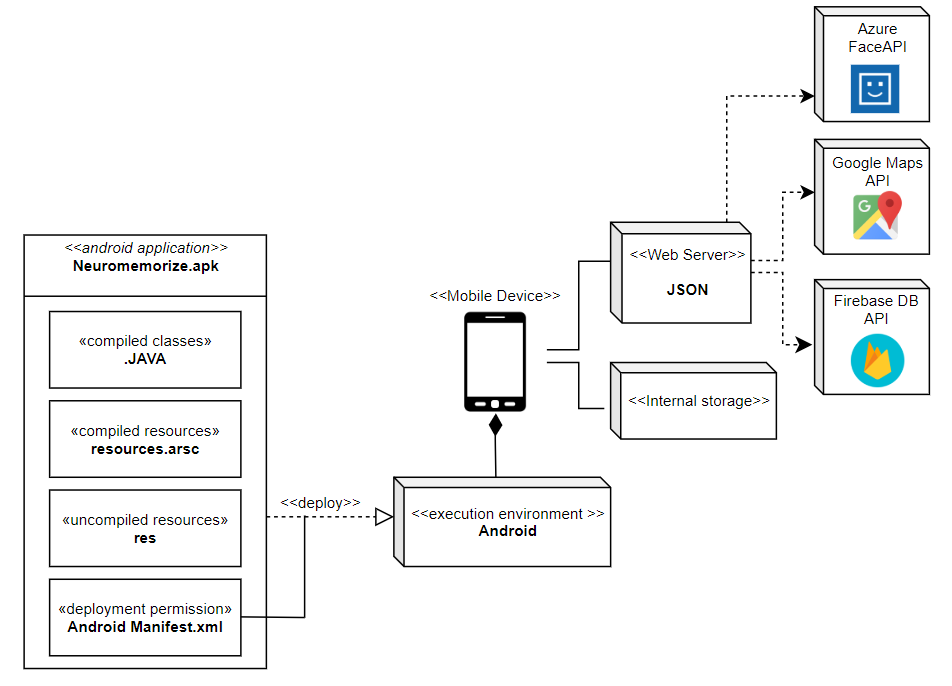
The users of the system logs in to the app.

* The Caretaker can set reminders for the patient to take daily medicines and perform daily routine tasks. He / She takes input for reminders and also in the form of photos of relatives of the patient and their things and save in database.
* Family/Friends of the patient can send invites and share their photos for the patient to remember. Those pictures also get saved in database.
* Patient logs in to the application. He / She takes real-time images which is recognized by the face recognition algorithm. If the image exists in the local database then display information about the corresponding image. If not, then inform caretaker and save image in database.



**Figure 7: Activity Diagram**

**5.1.3.3 Deployment Diagram**



**Figure 8: Deployment Diagram**

**The deployment diagram consists of the Android application apk, Android execution environment, Mobile Device, Web Server and Internal Storage.**

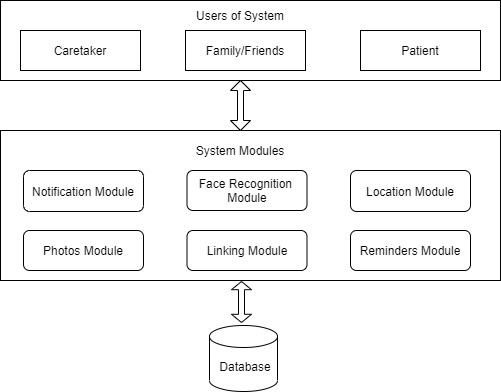
**Android applications** are written in Java. Android SDK tools compile and package the code along with any required data and resource files into Android application apk file. The apk file represents one Android application to be deployed to the Android-enabled mobile devices.

The [manifest](https://www.uml-diagrams.org/deployment-diagrams.html#deployment-spec) (deployment specification) file AndroidManifest.xml describes application requirements, such as the minimum version of Android required and any supported hardware configurations, and it also declares all components in the application.

The application can access Internal Storage of the mobile device to store data of the users. The application also uses Firebase Database for storing data on the cloud and Google Maps API to track the location of the Patient.

## **5.2 Proposed System Architecture**

Our application follows 3-tier architecture. Users of the system are: Caretaker, Patient, Family and friends. System APIs like notification handler (for sending notifications while patient is out of safe zone or not connected to Wi-Fi or battery is low), Tracking and Location handler (for tracking patients current location), Multimedia Handler (for taking input as images and storing in local database), Database Handler (handle local database) and Visualization handler (for viewing recognized images of patient’s closed ones). Face and object recognition comes in the next layer. The images get stored in the local database i.e. the mobile phone of the patient for them to recognize people. The server contains information about the patient which is entered by the caretaker.



**Figure 9: System Architecture**

## **5.3 Proposed Tools and Technology**

1. **Firebase Realtime Database** - Firebase provides a real-time database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud.
2. **Firebase Auth** - Firebase Auth is a service that can authenticate users using only client-side code. It supports [social login providers](https://en.wikipedia.org/wiki/Social_login) Facebook, GitHub, Twitter and Google (and [Google Play Games](https://en.wikipedia.org/wiki/Google_Play_Games)). Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase.
3. **Microsoft Azure Face API** - The Azure Cognitive Services Face service provides algorithms that are used to detect, recognize, and analyze human faces in images. The ability to process human face information is important in many different software scenarios. Example scenarios are security, natural user interface, image content analysis and management, mobile apps, and robotics. The Identify API is used to identify a detected face against a database of people. This feature might be useful for automatic image tagging in photo management software. [9]

**5.3.1 Hardware Resources**

* Minimum 2GB RAM

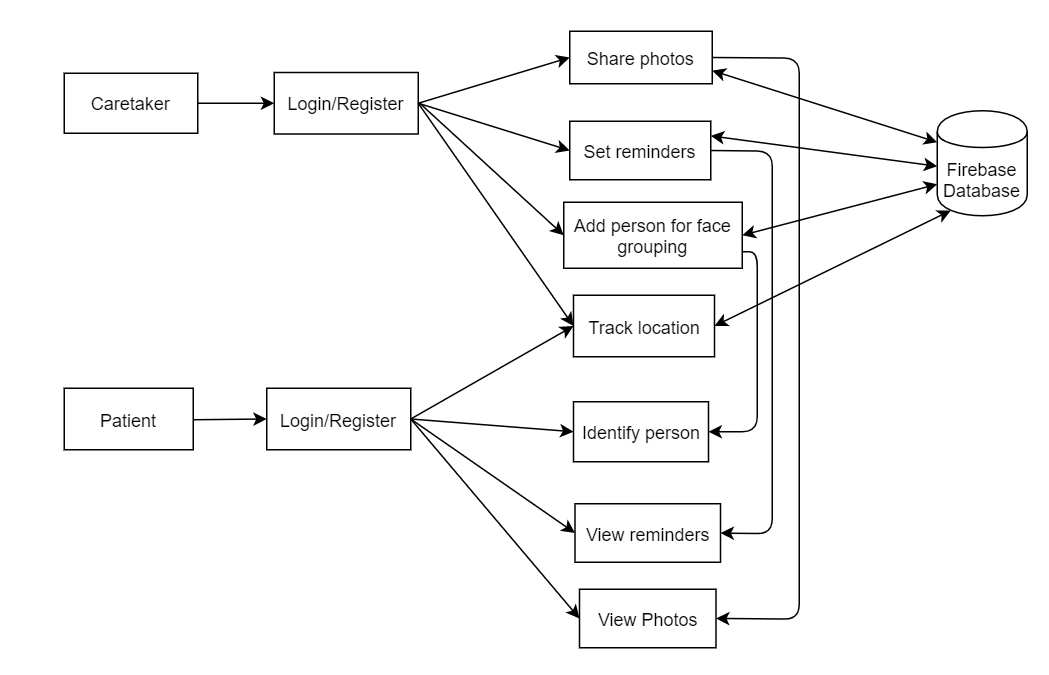
**5.3.2 Software Resources**

* Executable file installment allowed
* Installment uninterruptedness

**Chapter 6**

# **Implementation**

## **6.1 Working of System**



**Figure 10: Working of system**

The following are the steps for working of the system:

* The caretaker logins to the app and register with his/her details. A QR code will be generated which will be scanned by the patient. This will link caretaker and the patient.
* After logging in, the patient will see 4 tabs namely - Face recognition, Location Tracker, near ones, and Profile.
* The caretaker can see his/her profile, reminders, location of patient, update tab for any update in any details of patient.
* Family can login and view patient details and set reminders for patient. They can track the patient’s location, group photos by people and also view they profile.

## **6.2 Algorithms and Tools used**

The following are the algorithms and tools which are going to be used in this project are:

1. **Firebase Realtime Database** - Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase's cloud.
2. **Firebase Auth** - Firebase Auth is a service that can authenticate users using only client-side code. It supports [social login providers](https://en.wikipedia.org/wiki/Social_login) Facebook, GitHub, Twitter and Google (and [Google Play Games](https://en.wikipedia.org/wiki/Google_Play_Games)). Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase.
3. **Android Studio** - Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. We will be using Android Studio for developing and integrating modules of application.
4. **Microsoft Azure Face API** - The Azure Cognitive Services Face service provides algorithms that are used to detect, recognize, and analyze human faces in images. The ability to process human face information is important in many different software scenarios. Example scenarios are security, natural user interface, image content analysis and management, mobile apps, and robotics. The Identify API is used to identify a detected face against a database of people. This feature might be useful for automatic image tagging in photo management software.

Broadly, recognition describes the work of comparing two different faces to determine if they're similar or belong to the same person. [9]

**Recognition related Data Structures:**

The recognition operations use mainly the following data structures. These objects are stored in the cloud and can be referenced by their ID strings. ID strings are always unique within a subscription. Name fields can be duplicated.

**Table 10: Data structures for Azure Face API**

|  |  |
| --- | --- |
| Name | Description |
| DetectedFace | This single face representation is retrieved by the [face detection](https://docs.microsoft.com/en-us/azure/cognitive-services/face/face-api-how-to-topics/howtodetectfacesinimage) operation. Its ID expires 24 hours after it's created. |
| PersistedFace | When DetectedFace objects are added to a group, such as FaceList or Person, they become PersistedFace objects. They can be [retrieved](https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f3039524c) at any time and don't expire. |
| FaceList or LargeFaceList | This data structure is an assorted list of PersistedFace objects. A FaceList has a unique ID, a name string, and optionally a user data string. |
| [Person](https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f3039523c) | This data structure is a list of PersistedFace objects that belong to the same person. It has a unique ID, a name string, and optionally a user data string. |
| PersonGroup or LargePersonGroup | This data structure is an assorted list of Person objects. It has a unique ID, a name string, and optionally a user data string. A PersonGroup must be trained before it can be used in recognition operations. |

**Recognition Operation:**

The [Identify](https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395239) operation takes one or several face IDs from DetectedFace or PersistedFace and a PersonGroup and returns a list of Person objects that each face might belong to. Returned Person objects are wrapped as Candidate objects, which have a prediction confidence value. [9]

## **Step 1: Authorize the API call**

Every call to the Face API requires a subscription key. This key can be either passed through a query string parameter or specified in the request header.

## **Step 2: Create the PersonGroup**

Each person has several faces registered. The faces must be detected from the images.

### **Step 2.1: Define people for the PersonGroup**

A person is a basic unit of identify. A person can have one or more known faces registered. A PersonGroup is a collection of people. Each person is defined within a particular PersonGroup. Identification is done against a PersonGroup. The task is to create a PersonGroup, and then create the people in it.

### **Step 2.2: Detect faces and register them to the correct person**

Detection is done by sending a "POST" web request to the [Face - Detect](https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395236) API with the image file in the HTTP request body. When you use the client library, face detection is done through one of the Detect..Async methods of the FaceClient class.

## **Step 3: Train the PersonGroup**

The PersonGroup must be trained before an identification can be performed by using it. The PersonGroup must be retrained after you add or remove any person or if you edit a person's registered face. The training is done by the [PersonGroup – Train](https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395249) API.

## **Step 4: Identify a face against a defined PersonGroup**

When the Face service performs identifications, it computes the similarity of a test face among all the faces within a group. It returns the most comparable persons for the testing face. This process is done through the [Face - Identify](https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395239) API or the IdentifyAsync method of the client library.

The testing face must be detected by using the previous steps. Then the face ID is passed to the identification API as a second argument. Multiple face IDs can be identified at once. The result contains all the identified results. By default, the identification process returns only one person that matches the test face best.

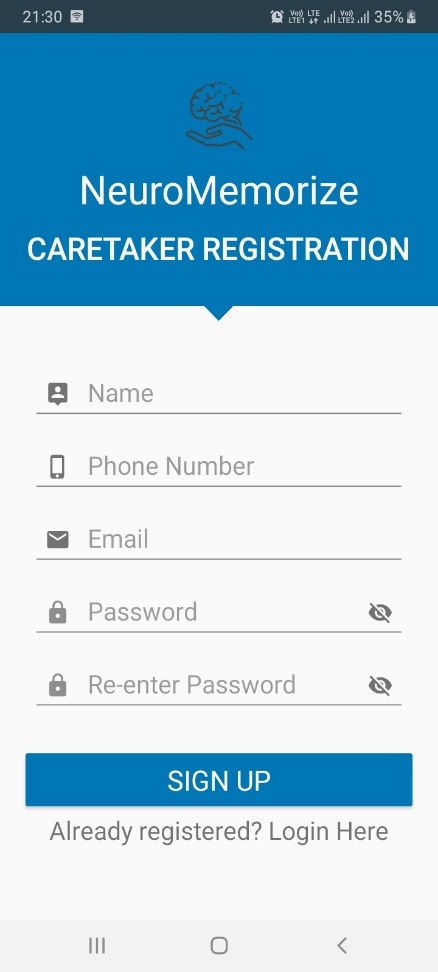
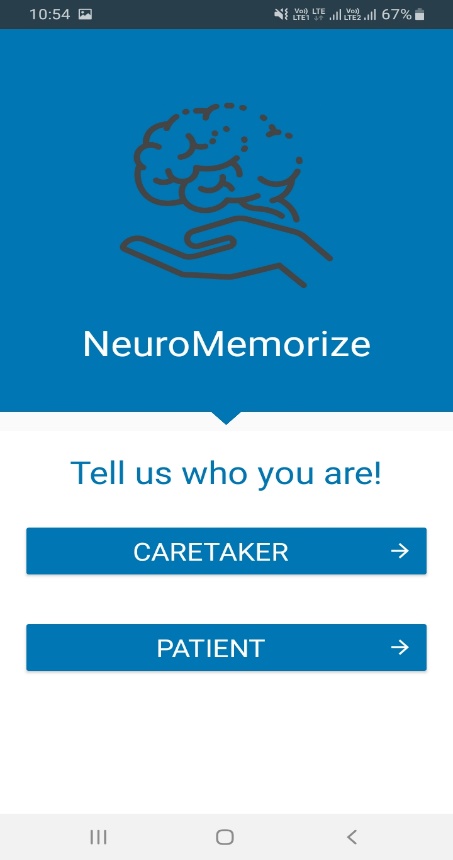
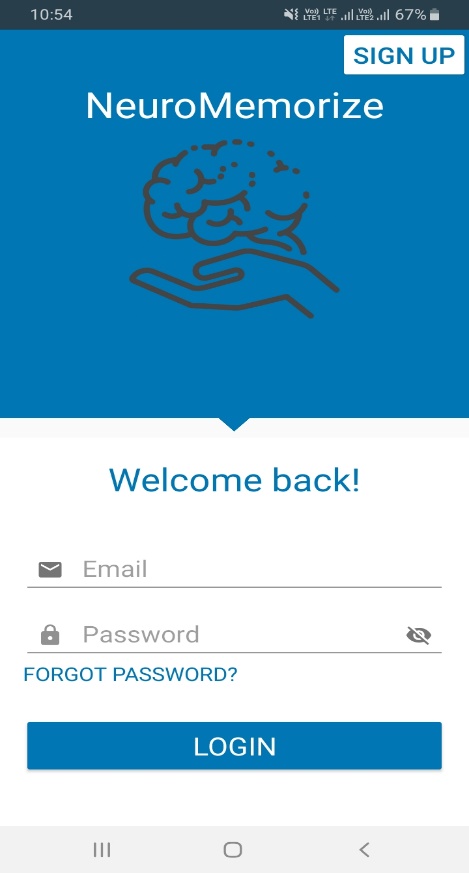
## **Step 5: Request for large scale**

A PersonGroup can hold up to 10,000 persons based on the previous design limitation.

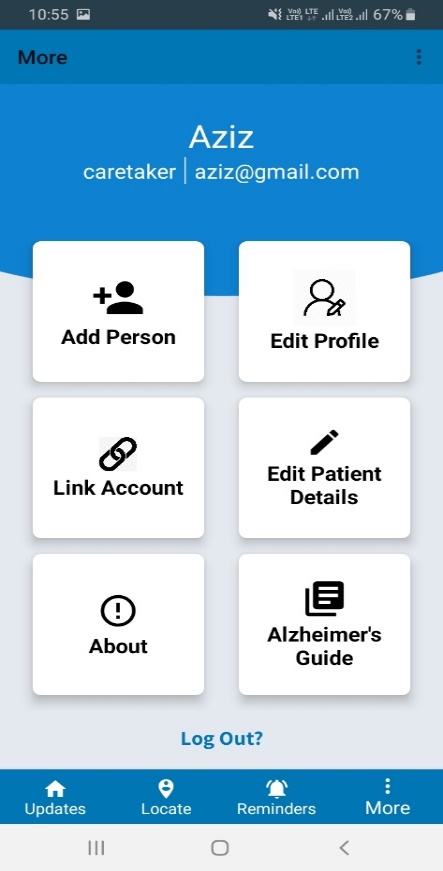
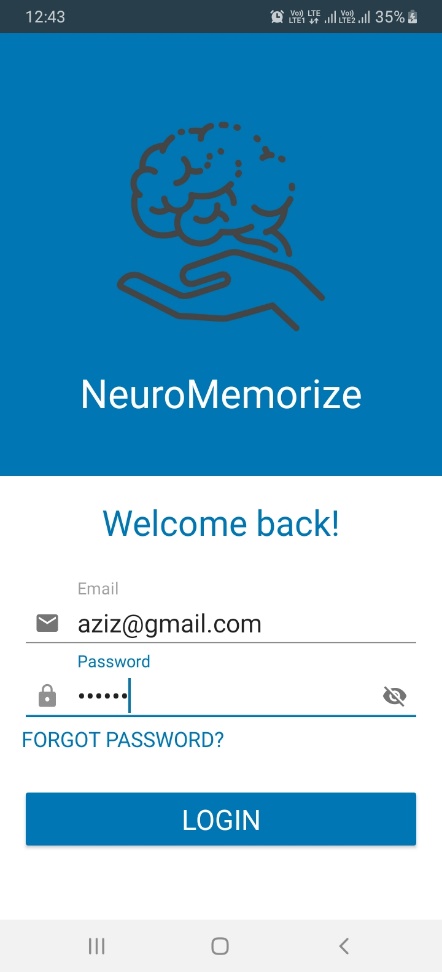
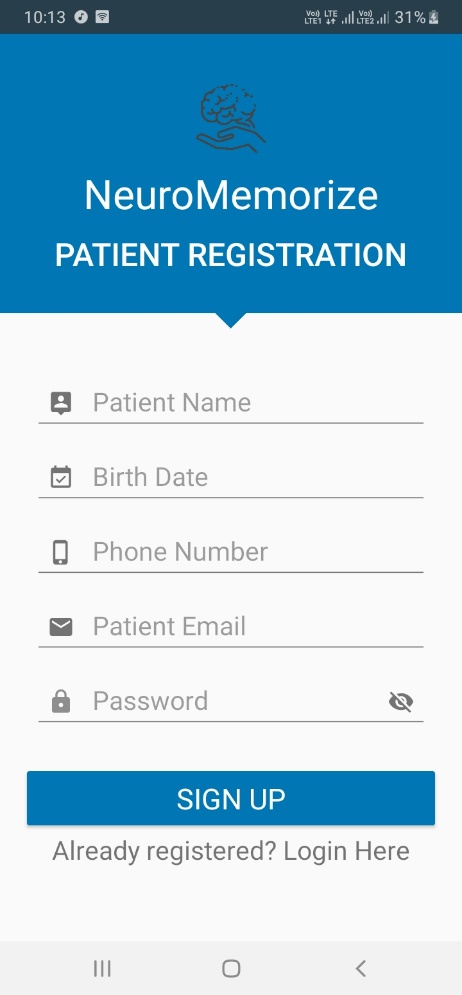
## 

## **6.3 Interface Design**

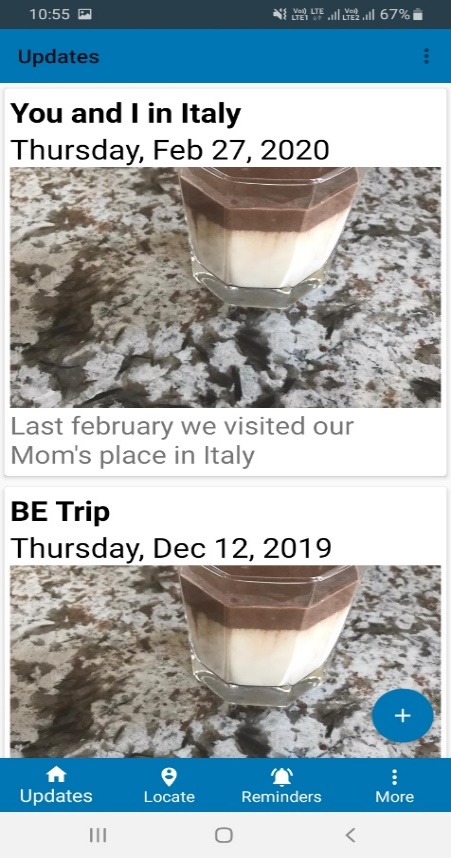
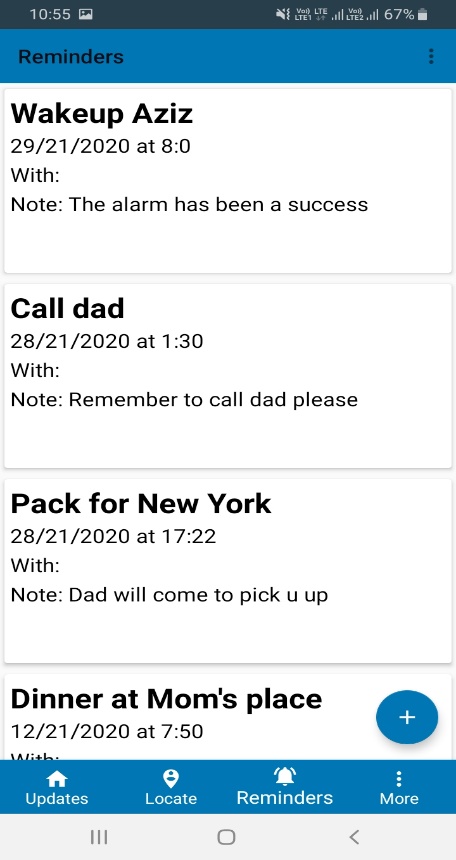
The interface design of the application is shown in the figures below:



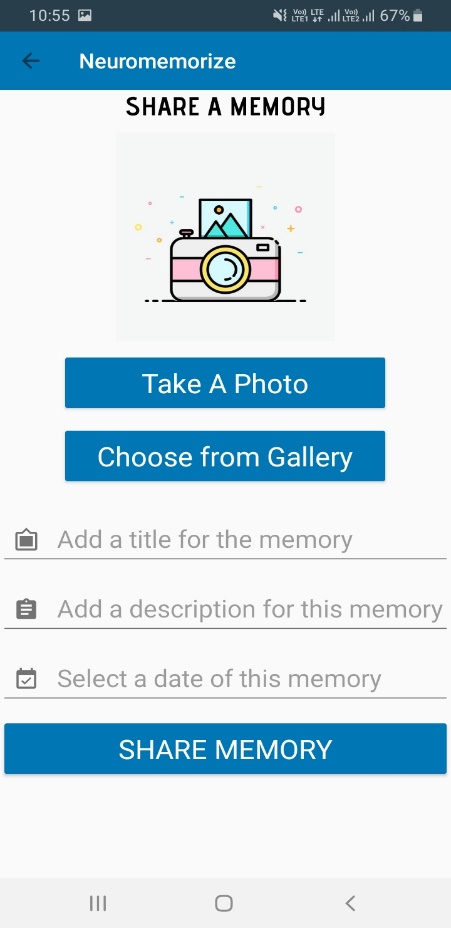
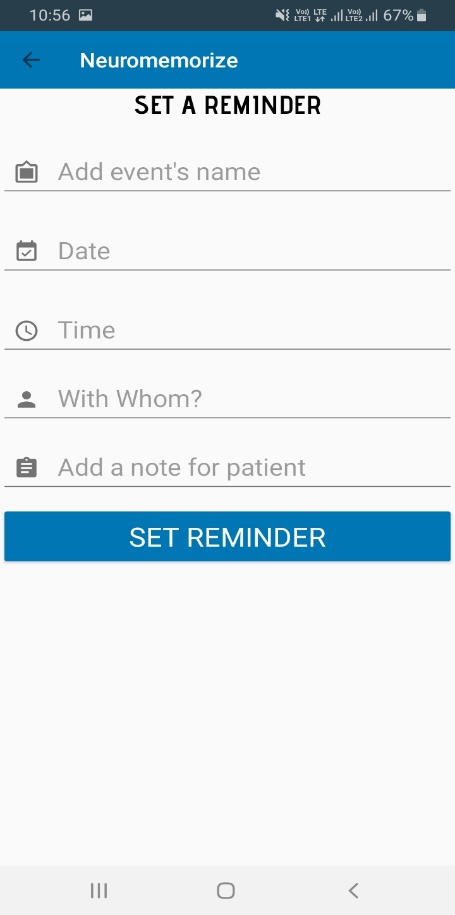
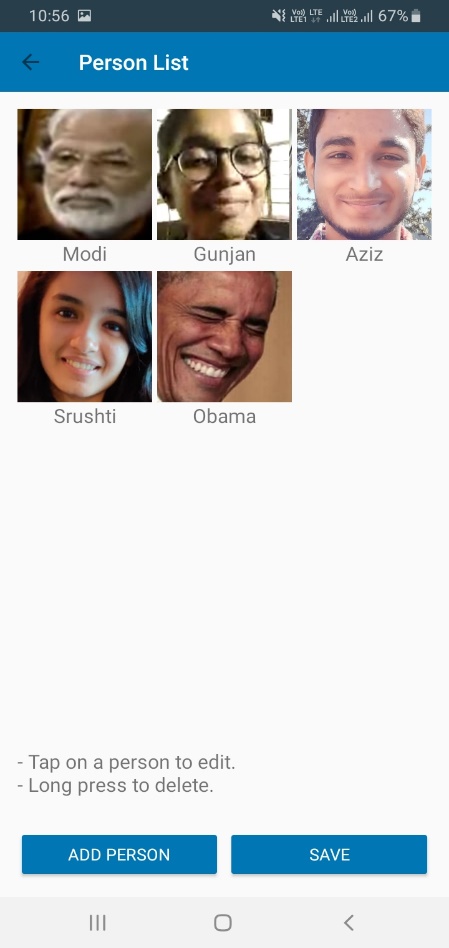
# **Figure 11: Login Figure 12: Users Figure 13: Caretaker Details**



**Figure 14: Patient Details Figure 15: Caretaker Login Figure 16: Caretaker Functions**

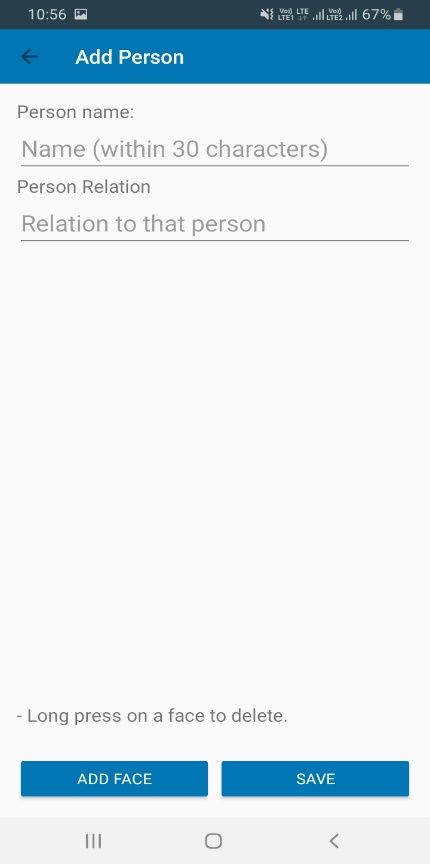
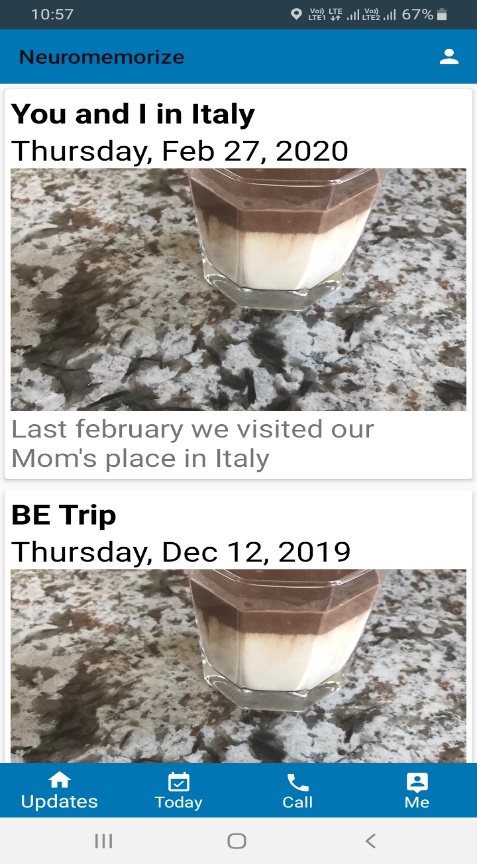
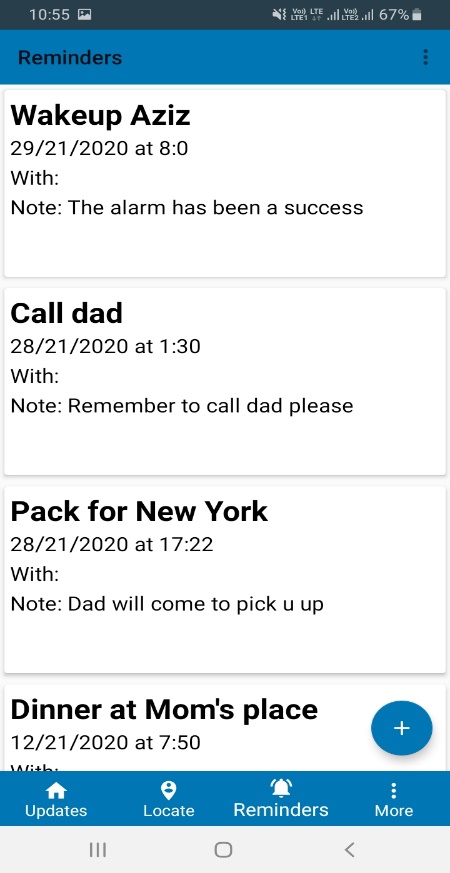


**Figure 17: Reminders pushed by Figure 18: Location Tracking Figure 19: Updates pushed by Caretaker Caretaker**

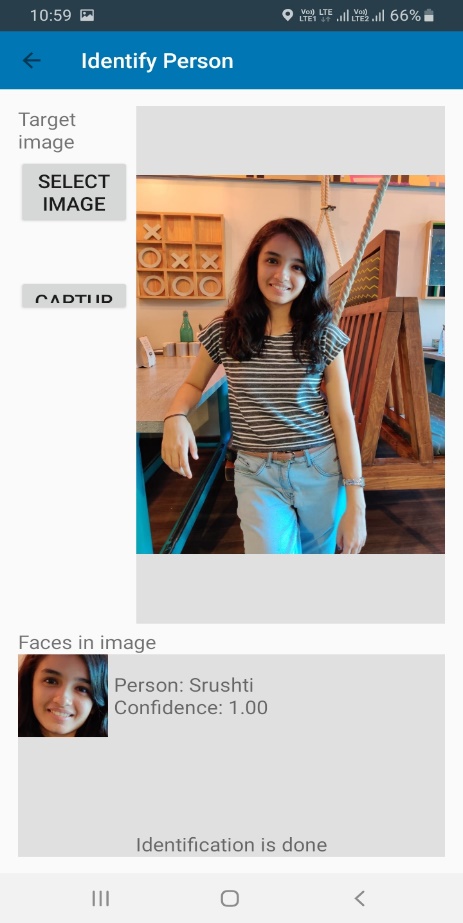
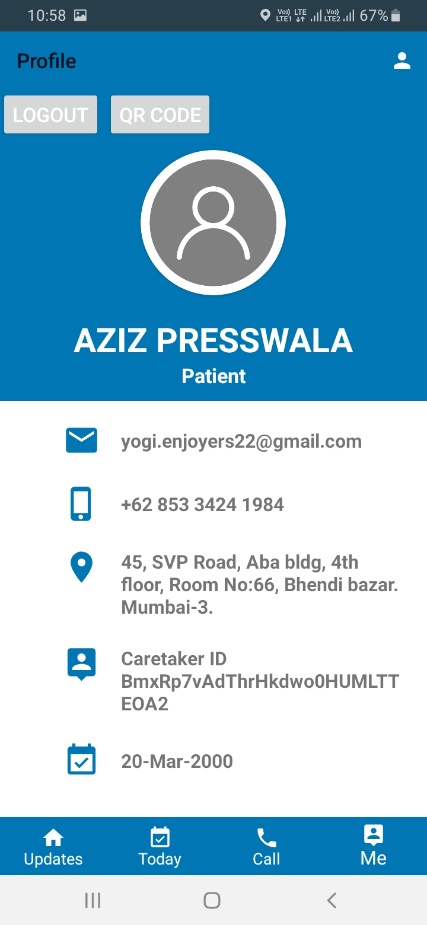
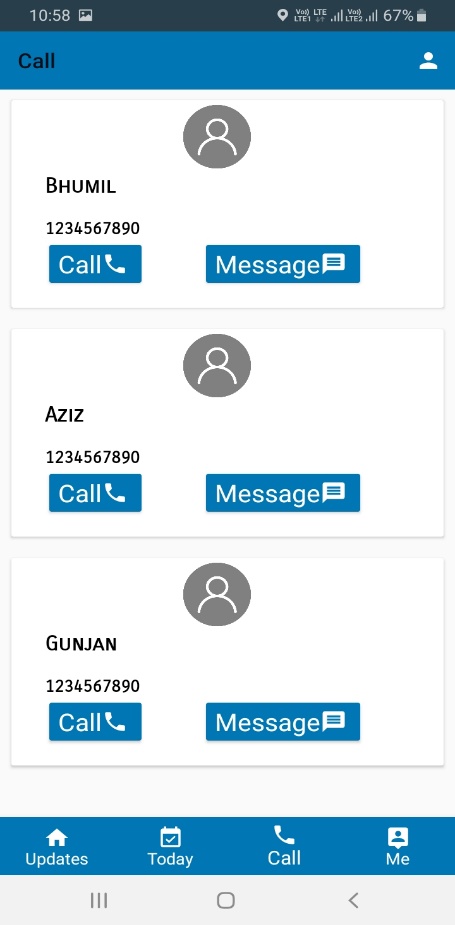
**Figure 20: Memory sharing Figure 21: Setting reminders Figure 22: Face training (Person**

**List)**

**Figure 23: Add person Figure 24: Patient receiving updates Figure 25: Patient receiving**

**Reminders**



**Figure 26: Call/Message Tab Figure 27: Patient Profile Figure 28: Face recognition**

**Chapter 7**

# **Testing and Results**

## **7.1 Test Plan**

A test plan is a general document for the entire project that defines the scope, approach to be taken, and the schedule of testing as well as identifies the test items for the entire testing process and the personnel responsible for the different activities of testing.

The test planning can be done well before the actual testing commences and can be done in paralleled with the design and coding phase. The inputs for test plan are:

1. Project Plan

2. Requirement Document

3. System Design Document

The project plan is needed to make sure that the test plan can be consistent with the overall plan for the project and testing schedule matches that of the project plan.

The requirement documents and the design documents are the basic documents for selecting the test unit and deciding the approaches to be used during testing. A test plan should contain the following:

1. Test Unit Specifications

2. Features to be tested.

3. Approach for testing.

**Test Unit**

Set of one or more modules, together with associated data, that are from single program and that are the object of testing.

**Features to be tested**

1. Ability to access the system at all times.

2. User should be able to upload image of any format.

3. The uploaded image should be successfully recognized by the program.

4. The model should produce accurate image.

5. The system should be able to give results on the screen in least time as possible.

**Approaches for testing**

1. Unit testing

2. Integration Testing

## **7.2 Test Cases**

The application was tested against the following test cases which gave the following results:

**Table 11: Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Case Description | Expected Result | Actual Result | Pass/Fail |
| 1 | Linking of Caretaker and Patient via QR code | The scanner scans the QR code for linking | Caretaker successfully links with the patient by scanning the QR code. | Pass. |
| 2 | Real time face recognition of the people related to the patient | Face training and recognition of people related to the patient by Microsoft Azure API. | Successfully recognizing faces with confidence value more than 70% in all trails. | Pass. |
| 3 | Location Tacking of the patient by the caretaker | Real time location of the patient should be viewed by the caretaker using Google Maps API. | Location of the patient gets updated every 10 seconds to get accurate location of the patient. | Pass. |
| 4 | Setting daily reminders by the caretaker for patient to remember daily routine. | Reminders should notify the patient on time and should not miss any of them | Successful implementation of setting daily reminders | Pass. |
| 5 | Caretaker uploading picture memories on patient’s home screen | Images should be uploaded without compression and with clear image | 100% accuracy. Successful uploading of memories | Pass. |
| 6 | Picture based phonebook for calling and messaging | Calling and messaging should happen with one click. | 100% accuracy.  Efficient calling and messaging. | Pass. |

## **7.3 Testing Methods Used**

System testing is a critical phase implementation. Testing of the system involves hardware devise and debugging of the computer programs and testing information processing procedures. Testing can be done with text data, which attempts to simulate all possible conditions that may arise during processing. The testing methods adopted during the testing of the system are:

1. **Unit testing**

Unit testing is a method by which individual units of source code, sets of one or more program modules together with associated control data, usage procedures, and operating procedures, are tested to determine if they are fit for use. In our tool, we considered each module as one unit and tested these units with help of test cases and test plan developed. Unit testing was carried out on each module and on every function within the module. Output of each unit was assessed for accuracy and if found incorrect, appropriate corrections were made.

1. **Integration testing**

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. The modules of our tool were integrated together in order to verify that they provide the required functionalities appropriately. The various modules were tested together to check for their accuracy and compatibility.

1. **System Testing**

System testing of software is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. In this testing, we tested the system as whole to ensure that it provides the appropriate output as stated in the requirements. Overall performance of the system was also tested simultaneously.

1. **Validation Testing**

Validation testing is the assurance that a product, service, or system meets the needs of the customer and other identified stakeholders. Here, we gave the system various possible inputs that the user might give to the system and tested if it provides correct and expected outputs. In case of any deviation from the expected output, corrective action was taken.

## **7.4 Experimental Results and Conclusions**

The recognition accuracy of the system was good. The patients were able to successfully identify their closed ones. From the figure given below, it is evident that the application performs close to what was expected. The primary goal of the system was to carry out efficient face recognition of the family members and closed ones along with location tracking of the patient which could be seen by the caretaker in real-time. The test case was real time face recognition of any person which the patient has to recognize. The images of that person has to be trained by the caretaker and stored in the server. These images will be retrieved on the patient’s side which in turn will help the patient recognize the person by just pointing the camera to the person. The application successfully tracks the patient and the location updates every 10 seconds. The other features like daily reminders, calling and messaging, updates, notifying caretaker if the patient’s phone is not connected to the internet or battery is low, and lastly uploading memories on the patient’s side by the caretaker. All these features are tested which gives accurate results as required.

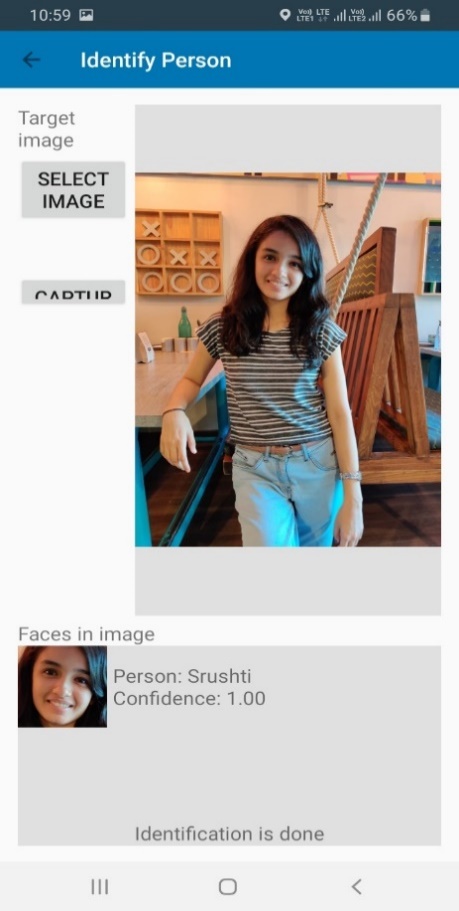


Figure 29: Final Recognized Image Figure 30: Tracking

**Chapter 8**

# **Conclusion**

Having studied various existing systems, their proposed approach and methodologies, algorithms they have used and their results. After their detailed analysis, a comparative study between them has been generated, understanding the efficiency and accuracy of algorithms used, their merits and demerits.

After having the analysis, the algorithms that will be used to implement the application is also decided. Microsoft Azure API was used as our Face recognition module.

Further the tools and technologies required to implement the system and various risk mitigation strategies has been discussed.

Lastly, the system using the above decided techniques and algorithms has also been developed.

Microsoft Azure API has been used to train images and recognize them and Google Maps API for the location tracking feature.

When tested with other applications, this application comes out as the first application with all the features which will improve the cognitive skills of the Alzheimer’s patients and help them remember their loved ones and daily routine. The application can be deployed as an application on minimal system and comes as zero cost.

**Chapter 9**

# **Future Scope**

In the course of developing the application, there were various roadblocks and setbacks but worked hard to overcome them in order to integrate and develop some of the best possible features to assist Alzheimer’s patients and their caretakers.

The application is still in early beta stage and it is aimed to add more and more features in the near future. Constant efforts are being made to enhance the overall user experience of the application so that it can be used effortlessly by the patients as well as the caretakers.

Some of the key features that can be added are - Voice based reminders, wherein the patient will be notified of a particular reminder by a built-in assistant similar to that of Google Assistant or Siri. It is also planned to integrate a Preliminary Quiz which the patient can take in order to know the severity of the disease. An Activity Tracker is also on the list, which would track the daily activities of the patient and notify the caretaker on timely basis.

With many features already implemented and much more on the cards, the application aims to ease the life of patients as well as their caretakers. With more and more people adopting new technologies in their day to day life, the future of this application looks bright.

# 

# **References**

1. Christina Yamagata, Marc Kowtko, Jean F. Coppola, Shannon Joyce, “Mobile App Development and Usability Research to Help Dementia and Alzheimer Patients”, IEEE, May 2013.
2. “Adoption and Use of a Mobile Health Application in Older Adults for Cognitive Stimulation”, Research and development department, Paris, 2016.
3. Alzheimer’s disease: Facts & Figures, available: www.brightfocus.org/alzheimers/article/ alzheimers-disease-facts-figures.
4. Timeless App, available: https://www.timeless.care/
5. 9 Great Apps for People with Dementia or Alzheimer’s, available: seniordirectory.com/articles/info/9-great-apps-for-people-with-dementia-or-alzheimers.
6. Efficient Face Recognition Algorithms and Techniques, available: www.rankred.com/face-recognition-algorithms-techniques.
7. Treatments, available: www.alz.org/alzheimers-dementia/treatments.
8. The structure of the MTCNN model, available: medium.com/@reina.wang/face-detection-neural-network-structure.
9. Microsoft Azure Face API, available: https://docs.microsoft.com/en-us/azure/cognitive-services/face/overview
10. OpenCV API, available: https://docs.opencv.org/2.4/modules/refman.html
11. Amazon Rekognition, available: https://aws.amazon.com/rekognition/