

**University Institute of Information Technology,**

**PMAS-Arid Agriculture University,**

**Rawalpindi Pakistan**

**Family GPS Tracker**

***By***

**Jawad Ahmed 18-Arid-3020**

**Hassaan Ahmed 18-Arid-3016**

***Supervisor*Ms.Farkhanda**

**Bachelor of Science in Software Engineering (2018-2022)**

**The candidate confirms that the work submitted is their own and appropriate  
 credit has been given where reference has been made to the work of others**.

**DECLARATION**

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software documentation and accompanied the report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Jawad Ahmed Hassaan Ahmed

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

**CERTIFICATE OF APPROVAL**

It is to certify that the final year project of BS (SE) “**Family GPS Tracker**” was developed by “**Jawad Ahmed (18-arid-3020)”** and “**Hassaan Ahmed (18-arid-3016)”** under the supervision of “**Ms. Farkhanda Qamar**” and that in their opinion; it is fully adequate, in scope and quality for the degree of Bachelor of Science in Software Engineering.

---------------------------------------

**Ms. Farkhanda Qamar**

**Supervisor**

---------------------------------------

**Jannat Khatoon**

**External Examiner**

---------------------------------------

**Prof. Dr. Yaser Hafeez**

**Director UIIT**

**Acknowledgement**

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Ms. Farkhanda Qamar” for personal supervision, advice, valuable guidance, and completion of this project. We are deeply indebted to him/them for encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

**Table of Contents**

Chapter [1: Introduction 1](#_Toc109342267)

* 1. [Brief 1](#_Toc109342268)
  2. [Relevance to Course Modules 2](#_Toc109342269)
  3. [Project Background 2](#_Toc109342270)
  4. [Literature Review 3](#_Toc109342271)
  5. [Analysis from Literature Review 4](#_Toc109342272)
  6. [Methodology and Software Lifecycle for This Project 4](#_Toc109342273)

[1.6.1. Rationale behind Selected Methodology 4](#_Toc109342274)

[Chapter 2: Problem Definition 5](#_Toc109342275)

[2.1. Purpose: 5](#_Toc109342276)

[2.2. Product Functions: 5](#_Toc109342278)

[2.3. Proposed Architecture: 6](#_Toc109342279)

[2.4. Project Deliverables: 7](#_Toc109342280)

[2.4.1 Projects Deliverables: 7](#_Toc109342281)

[2.4.2 Development Requirements: 7](#_Toc109342282)

[2.5. Operating Environment: 8](#_Toc109342283)

[Chapter 3: Requirement Analysis 9](#_Toc109342284)

[3.1 Functional Requirements 9](#_Toc109342285)

[3.2. Non-Functional Requirements 9](#_Toc109342286)

[3.3. Use Case Model 10](#_Toc109342287)

[3.3.1 Use Case Diagram: 11](#_Toc109342288)

[3.3.2 Actors Description: 12](#_Toc109342289)

[3.3.3 Use Case Description: 12](#_Toc109342290)

[Chapter 4: Design and Architecture 16](#_Toc109342291)

[4.1. System Architecture 16](#_Toc109342292)

[4.2. System Design 16](#_Toc109342293)

[4.2.1 UML Structural Diagrams 16](#_Toc109342294)

[4.2.1.1 Class Diagram 17](#_Toc109342295)

[4.2.1.2 ERD Diagram 18](#_Toc109342296)

[4.2.1.3 Activity Diagram 19](#_Toc109342297)

[4.2.1.4 Component Diagram 20](#_Toc109342298)

[Chapter 5: Implementation 22](#_Toc109342299)

[5.1 User Interface 22](#_Toc109342300)

[Chapter 6: Testing and Evaluation 30](#_Toc109342301)

[6.1 Sign Up Module : 30](#_Toc109342302)

[6.2 Login Module 31](#_Toc109342303)

[6.3 Location History Module 31](#_Toc109342304)

[6.4 Test Case for live location 32](#_Toc109342305)

[6.5 Test Case for Notification Module 33](#_Toc109342306)

[6.6 Test case for Geo fencing Module 33](#_Toc109342307)

[**Chapter 7: Conclusion and Future Work** 34](#_Toc109342308)

[7.1 Conclusion 34](#_Toc109342309)

[7.2 Future Work 34](#_Toc109342310)

# List of Figures

[Figure 3.1 : Use Case Diagram 11](#_Toc109349810)

[Figure 4.1 : System Architecture Diagram 18](#_Toc109349811)

[Figure 4.2 : Class Diagram 28](#_Toc109349812)

[Figure 4.3 : ERD Diagram 18](#_Toc109349811)

[Figure 4.4 : Activity Diagram 28](#_Toc109349812)

[Figure 4.5 : Component Diagram 18](#_Toc109349811)

[Figure 5.1 : Sign Up Screen 28](#_Toc109349812)

[Figure 5.2 : Profile Screen 18](#_Toc109349811)

[Figure 5.3 : Decide User Screen 28](#_Toc109349812)

[Figure 5.4 : Location History Screen 18](#_Toc109349811)

[Figure 5.5 : Code Generation Screen 28](#_Toc109349812)

[Figure 5.6 : Sign In Screen 18](#_Toc109349811)

[Figure 5.7 : Connect Device Screen 28](#_Toc109349812)

[Figure 5.8 : Home Screen 28](#_Toc109349812)

# List of Tables

[Table 3.1 : Authentication 33](#_Toc109550568)

[Table 3.2 : Link Device 33](#_Toc109550568)

[Table 3.3 : Set Geofence 33](#_Toc109550568)

[Table 6.1 : Sign Up Module 33](#_Toc109550568)

[Table 6.2 : Login Module 33](#_Toc109550568)

[Table 6.3 : Location History Module 33](#_Toc109550568)

[Table 6.4 : Live Location Module 33](#_Toc109550568)

[Table 6.5 : Notification Module 33](#_Toc109550568)

[Table 6.6 : Geofencing Module 33](#_Toc109550568)

# Chapter 1 : Introduction

In this chapter we will discuss over view of our whole project, its brief introduction, and how it is relevant to the courses which we have studied during our degree. We will also discuss project literature review and its analysis and methodology that we will use in project.

# Brief

The cases of missing children continue to rise which has been highly alarming to parents and relatives. Apart from the daily rise in children/women abduction or missing cases, we as a family member feel worried whenever our loved ones are away from us therefore, we need to stay connected to our family members or have information about the whereabouts of our family members and without the use of tracking technology, it is extremely difficult for us to keep a watch on our loved ones while we are away from them.

In fact it is a centuries-old problem that children go missing but now we have internet and with the use of it, we can provide a virtual solution to this problem by developing an app which will not only make sure the safety of our loved ones but also help us keep track of them .This application will facilitate family members monitoring by providing Gps location tracking feature .Many family locating applications have been introduced in the past but there were some flaws in them .To redress those flaws, we develop an advanced and efficient Family members monitoring system using latest technologies. The platform we are going to target is android so it will be an android application and as far as the users are concerned, this app is going to be useful for anyone in the family. Parents can use it to keep a watch on their children and in the same way, teenagers can use it to stay connected with their old parents. Since it is a network-based application, Wi-Fi and Gps are one of the requirements of our application to operate. Ensured of security will be one of our main concern for which we will be providing two layered security through QR code scanning and password verification. Provision of accurate locations and pushing notifications on time to make the product reliable will also be a matter of importance for us. In conclusion, this app provides peace of mind to parents, is a reassurance for children/women and assures the safety of family members.

# Relevance to Course Modules

Almost everything which we are using in our project “Family Gps Tracker” is totally relevant to our course materials.

* The android app which we are developing in our project is relevant to the subject “Modern Programming Language".
* The backend service which will help our application communicate with database server ,

Is going to be developed in C# language which is relevant to the subject “Visual Programming”.

* The data of the users which will be stored in Database and status will be shown on screen, is relevant to the subject Database Management System.

# Project Background

In this modern era, it is quite difficult for parents to be with children all the time as both of the parents have to work to fulfil their children’s needs .For this reason, they fail to keep a watch on their children .In the same way, an eldest son/daughter who has to take care of his old parents or younger siblings, needs to stay connected with his/hers loved ones or have information about their whereabouts in order to keep them safe. Therefore, we aim to provide a reliable and secured android app which lets you track your family members even if you are away from them. It ensures that your loved ones are safe and don’t go to any dangerous places. The proposed application will facilitate family member’s monitoring by providing GPS location tracking feature which includes live tracking, logging location history, draw safe zones and many more. Family linking would be the initial step to start supervision and would be done with simple steps to make the process convenient for user and tracking will be triggered after receiving a proper permission for it. Once the tracking is triggered, user will be able to draw safe zones and receive notification when family members leave or enter that zone. In short, this will make sure you are linked with your family even if you are busy.

# Literature Review

There are already applications that provide GPS location tracking feature but there are some flaws in it and they lack some essential features which should be present in every tracking system. There have been many issues reported, in the previously introduced tracking systems. Some of which are poor device linking, no option for immediately notifying your family members in case of emergency. We will try to correct all these issues and provide all those necessary functionalities by developing a new tracking system. Some of the existing systems having some shortcomings are discussed in detail down below:

**Find My Kids:**

This is a GPS location tracker app that is designed for child safety and parental control. This can be done discreetly and track your child without them having to even know. You can connect a GPS watch or install the app to track your child’s phone and location instantly.

**Short Comings:**

It has no option to view or download an entire location history of your family members.

**Life 360 Family Locator:**

Family Locator lets you stay linked to your family members during the day and will allow you to share status via GPS. You can install the Life360 Family Locator app on your phone, and invite your family to join also so you can all stay connected.

**Short Comings:**

It has no option for immediately notifying your family members in case of emergency.

**Beacon Family Locator:**

This app is also a free GPS tracking, missing person and personal safety app.

**Short Comings:**

It doesn’t have a feature like smart alert in which you receive real time smart alerts when your loved ones arrive or leave home, school or any places you set.

# Analysis from Literature Review

The goal is to make a secured and less complex system of tracking. So as per the goal we enabled futuristic system that will help family members stay linked all the time. Here the app is designed in a way which anyone can use it to monitor their family members. This app is a solution to all those issues faced in the existing systems. It provides all those essential feature that should be available in a tracking system as well as some new features. Although an app is never perfect but we will try to upgrade our app with the passage of time.

# Methodology and Software Lifecycle for This Project

We use **agile methodology**. The proposed system is going to follow the agile development method because this method assists in responding to the unpredictability of constructing software. This method offers a light framework and focus on rapid delivery of the software. Moreover, agile method facilitates us to create and respond to change in an uncertain and turbulent situation. Our focus is to collaborate with stakeholders during project to ensure the product quality which is impossible without using agile methodology.

* + 1. Rationale behind Selected Methodology
* Creativity and innovation
* Lower costs
* Improved quality
* Customer satisfaction
* Focus on users
* Early and predictable delivery
* Predictable costs and schedule
* Allows for change

# Chapter 2 : Problem Definition

## 2.1. Purpose:

## The main motive behind this project is to facilitate family member's monitoring and reduce children/women abduction or missing cases by providing a system that will help parents or any person, who wants to supervise his/her family members, keep track of their loved ones. Speaking of modern era, where both parents have to work to fulfil their children's needs, parents can’t be with their children’s all the time and fail to keep a watch on them. In the same way, an eldest son/daughter who has to take care of his old parents or younger siblings needs to stay connected with his/hers loved ones or have information about their whereabouts in order to keep them safe. Therefore, we aim to provide a reliable and secured android app which lets you track your family members even if you are away from them. It ensures that your loved ones are safe and don’t go to any dangerous places.

# . Product Functions:

The product functions are **Register:** In registration the users provide their credentials and those credentials get verified and stored into database. Once the user is registered with the system, user is able to **Login:** The user provide his/hers account credential after that they can use the app features. **Device Linking :** It is one of the product's special functionalities in which family member has to generate a 5-digit code and the family manager has to insert the password to get their devices linked. After the linking their devices, the family manager will be able to track his/hers family members location. The family manager can start or stop the supervision of his/hers family members. **Live Tracking:** In live tracking, this feature lets user know where his family members are on real-time basis by showing their current location on google map. **Location History:** This features keeps the user informed about their family member’s movements throughout the day by logging their activities after every 1 minute. **Create Safe Zones:** it lets the user create a specific geographical boundary or set a virtual fence for their family members on a google map. User will immediately be notified whenever his family members breach the geo-fence. **Smart Alerts:** With this feature, user can receive real time smart alerts when your loved ones arrive home, school or any places you set. **Notification Log:** This module provides the family manager with a notification history log where it logs all the notifications received by the family manager **Panic Button:** In emergency situations, your family members can press the panic button to notify you immediately

* register
* login
* Link Device
* Draw Safe Zone
* Live Tracking
* Location History
* Smart Alerts
* Notification Log
* Panic Button

## 2.3. Proposed Architecture:

MVVM Architecture has been used in our project.

It has three layers:

1. Model
2. View
3. View Model

* **Model:**

It represents the data and the business logic of the Android Application. It consists of the business logic - local and remote data source, model classes, and repository.

* **View:**

It consists of the UI Code (Activity, Fragment), XML. It sends the user action to the View Model but does **not** get the response back directly. To get the response, it has to subscribe to the observables which View Model exposes to it.

* **View Model:**

It is a bridge between the View and Model (business logic). It does not have any clue which View has a reference to this view model. It basically interacts with the Model and exposes the observable that can be observed by the View.

## 2.4. Project Deliverables:

Following are the deliverables and development requirements:

## 2.4.1 Projects Deliverables:

In this project the deliverables will consist of the input as equipment (i.e., hardware components and software components) then there will be a process (i.e., development phases in which development will be completed) applied on the input and then as a result there will be an output (Project being completed “Family GPS Tracker”). In this case product deliverables are the completed parts or modules of the project. Input will be software-based components. The project is divided into in different modules and each module is major milestone in the project.

## 2.4.2 Development Requirements:

The major requirements for the development process are listed below:

**Google Map Services:**

Since, our applications has to track location and display on the google map. This API is one of the main requirements of development and is a dependency for most of the modules. In other words, our app can’t function without this library.

**Geofencing API :**

This API lets you place virtual fences around a specific location that measures when someone with a mobile phone enters or exit the digital boundary in the physical world. This API is a dependency for draw safe zones module.

**Places Library:**

The Places API lets you search for place information on google. With this api, You can search for areas of interest, things to do, or notable locations in Google Maps. This library is a dependency for the draw safe zones module as user have to select a place before drawing safe zones.

## 2.5. Operating Environment:

Operating environment for the Family GPS Tracker is as listed below:

**Operating System**: Android Operating system

**Database**: SQL Server

**Platform:** Android studio in which android app will be developed using kotlin language and

Rest web services which will handle all the interactions between our app and the database,

will be developed on Visual Studio using c#.

**2.6. Assumptions and Dependencies:**

**Assumptions:**

The application developed from this technique is more efficient than others. This software gives accurate result in real time for location tracking.

**Dependencies:**

This software is dependent on internet connectivity.

Application is also dependent on phone GPS.

# Chapter 3 : Requirement Analysis

In this chapter we will define all the requirements of proposed system that include functional and non-functional requirements. We will also discuss about use cases of the system and see how our system will respond to various use cases.

# Functional Requirements

For our system to work and facilitate the user number of functional requirements have been are needed. These functional requirements also be presumed as interface requirements as they are all but interface.

* The system will allow the user to register himself.
* The system will authenticate the user through login functionality.
* The system will display the available supervisions on family manager’s account.
* The system will allow the users to link their devices.
* The system will allow the user to track location live.
* The system will allow the user to view location history.
* The system will allow the user to create safe zones for his/hers family members.
* The system will send notifications to its users.
* The system will provide its user with notification history log.

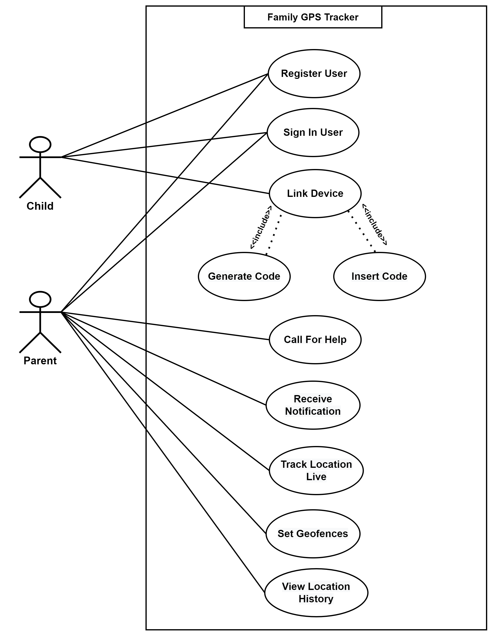
# Non-Functional Requirements

* The system should be scalable so that is capable of handling the concurrent requests from different users.
* The system should provide confidentiality for user data.
* The system should be stable and reliable enough to handle the exceptions.
* The system should be available for 24/7 of the time to handle the concurrent request of the users.
* The system should permit only authorize users to ensure its security.
* The system should be efficient enough to handle the concurrent request from the user.
* Interface and the system itself should be user friendly so that the customer will feel it easy to use.
* The system will authenticate the user by verifying the credentials to database.

# Use Case Model

In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. Following are the use cases of the Family GPS Tracker.

## 3.3.1 Use Case Diagram:



**Figure 3.1 : Use Case Diagram**

## 3.3.2 Actors Description:

**We have three types of actors.**

**Parent:**

Parent is the primary actor who is mainly interacting with the app as he is the one supervising all of his family members. He can monitor his loved ones by tracking them live or viewing their location history .He is also able to set safe zones for them and receive notification when they leave or arrive to that particular zone.

**Child:**

Child is person who is going to be supervised by the family manager .He/She could be anyone in the family but normally family member is a child or a sibling.

**Google Map Service:**

It is actually a third party service which enables our application to display google maps.

**Geofencing Service:**

It is also a third party service which allows the user to create virtual boundaries on places which he considers are safe for his family members.

## 3.3.3 Use Case Description:

**Table 3.1 : Authentication**

|  |  |
| --- | --- |
| **Use Case ID:** | ID-01 |
| **UseCase Name:** | Authentication |
| **Actors:** | Parent, Child |
| **Description:** | User will provide its credentials and get authenticated by the system through database. |
| **Trigger:** | When user clicks on login app |
| **Preconditions:** | Username and password must be provided by the users. |
| **Post conditions:** | User will login successfully. |
| **Normal Flow:** | Credentials will be entered by users  Credentials will be authenticated by the system  User will be logged in successfully. |
| **Alternative Flows:** | If user is not already registered, then user will first sign up  User will again be authenticated through database |
| **Exceptions:** | If user will be not authenticated, error message will be appeared. |
| **Special Requirements:** | None |
| **Assumptions:** | None |
| **Notes and Issues:** | User will have only 3 attempts for login. After 3 attempts user will have to wait for 30 seconds to retry. |

**Table 3.2 : Link Device**

|  |  |
| --- | --- |
| **Use Case ID:** | ID-02 |
| **UseCase Name:** | Link Device |
| **Actors:** | Parent , Child |
| **Description:** | Child generates pairing code which is only valid for 1 hour and then parent uses that code . |
| **Trigger:** | When Parent wants to supervise a child. |
| **Preconditions:** | Both Parent and child should be authenticating. |
| **Post conditions:** | Both of their devices will get linked. |
| **Normal Flow:** | Child generates the Pairing Code and Parent uses that code to make a connect with child’s device. |
| **Alternative Flows:** | None |
| **Exceptions:** | Parent can supervise only 5 family members. |
| **Special Requirements:** | None |
| **Assumptions:** | Android phone will be connected to the internet and have its GPS on. |
| **Notes and Issues:** | If the Parent want to the link the devices then Child must generate a pairing code. |

**Table 3.3 : Set Geofences**

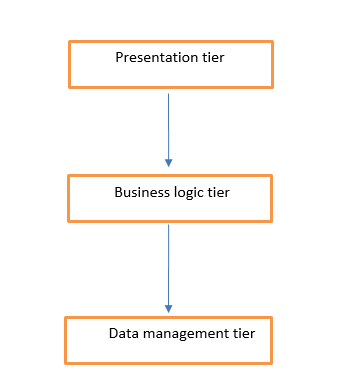
|  |  |
| --- | --- |
| **Use Case ID:** | ID-03 |
| **UseCase Name:** | Set Geofences |
| **Actors:** | Parent , Geofencing Service |
| **Description:** | The parent puts virtual fences on the places which he considers are safe for his family using Geofencing api and receives notification when his/hers family members leave or arrive at that particular zone. |
| **Trigger:** | When the Parent wants to set Geofences for their family members. |
| **Preconditions:** | Parent has to select a place on which he wants to put virtual fences.  Parent has to set the circle radius for that zone. |
| **Post conditions:** | Parent will be able to start receiving notifications whenever his/hers family members enter or breach that fence. |
| **Normal Flow:** | The Parent selects a place by searching it.  Parent sets the circle radius for that zone.  Parent start receiving notifications whenever his/hers family member arrive or leave that zone. |
| **Alternative Flows:** | None |
| **Exceptions:** | None |
| **Special Requirements:** | None |
| **Assumptions:** | None |

# Chapter 4 : Design and Architecture

In this chapter we will discuss the design and architecture of our system.

## 4.1. System Architecture

As system design varies from system to system, therefore user need to have the architecture view of the whole system.



**Figure 4.1 : System Architecture**

## 4.2. System Design

Systems design is the process of defining elements of a system like components, modules, architecture and their interfaces and data for a system based on the specified requirements. The purpose of the System Design process is to provide sufficient detailed data and information about the system. Following is the system design of the family gps tracker.

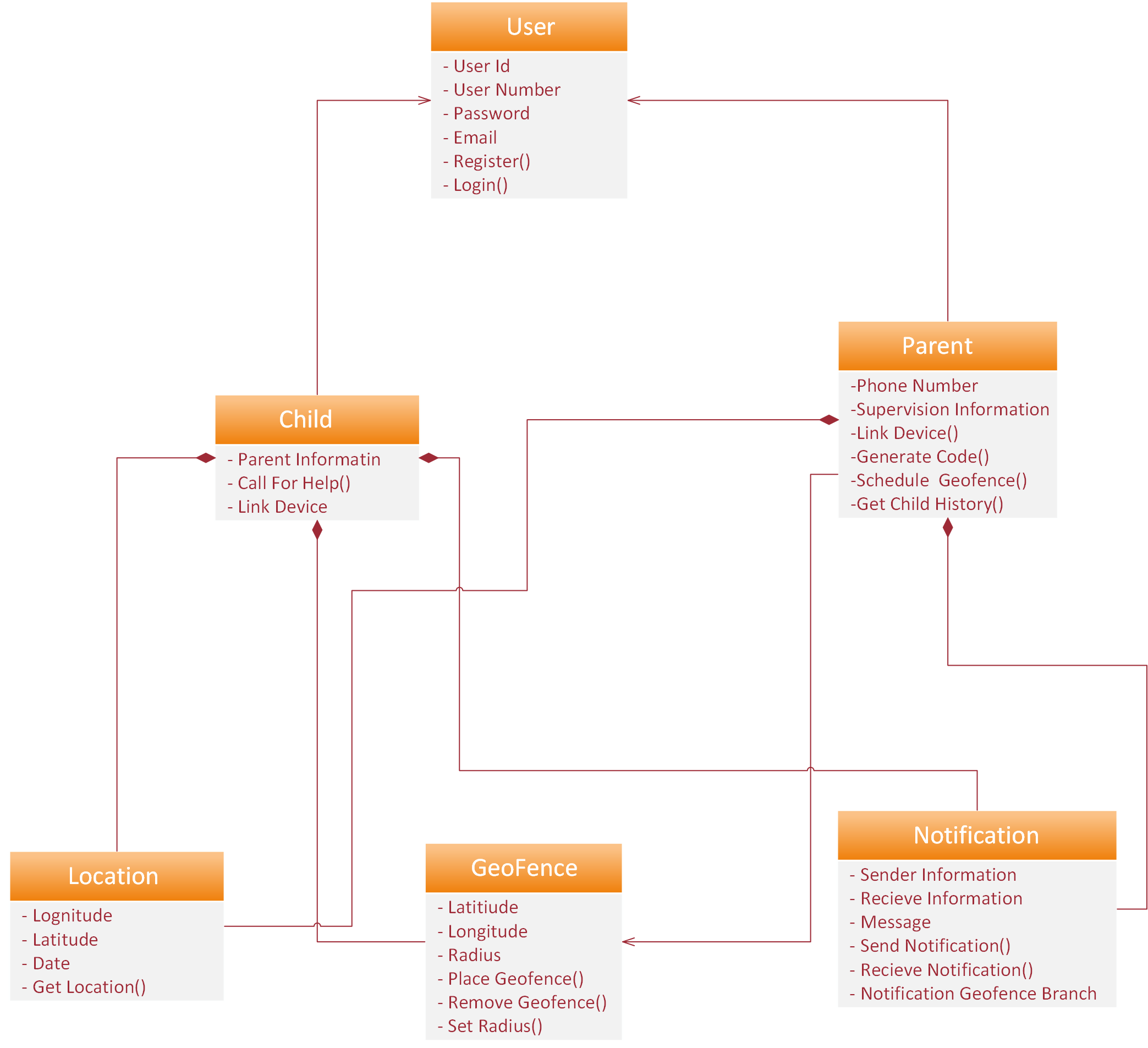
## 4.2.1 UML Structural Diagrams

Following are the UML structural diagrams of our system:

## 4.2.1.1 Class Diagram

The class diagram consists six classes which consisting of User class, Parent class, Child class, Location class, Notification class and Geo-fence class is shown in Figure 4.1. The User class will be inherited by Parent class and Child class as they both are the user using the device. The Parent class and Child class will both using Location class in which Parent depends on the Location class to get the location history of their child and the Child class is using the Location class to record their current location. Besides, the Parent class will also use the Geo-fence class to set the Geo-fence to monitor the child and the Geo-fence class is then passing the Geo-fence information set by the parent to the Child app by using the Child class to add that information into Geo-fence API called from the Child app. The Parent class and child class will both using notification class as child will send notifications to parent and on the other hand parent will receive notifications from the particular child.

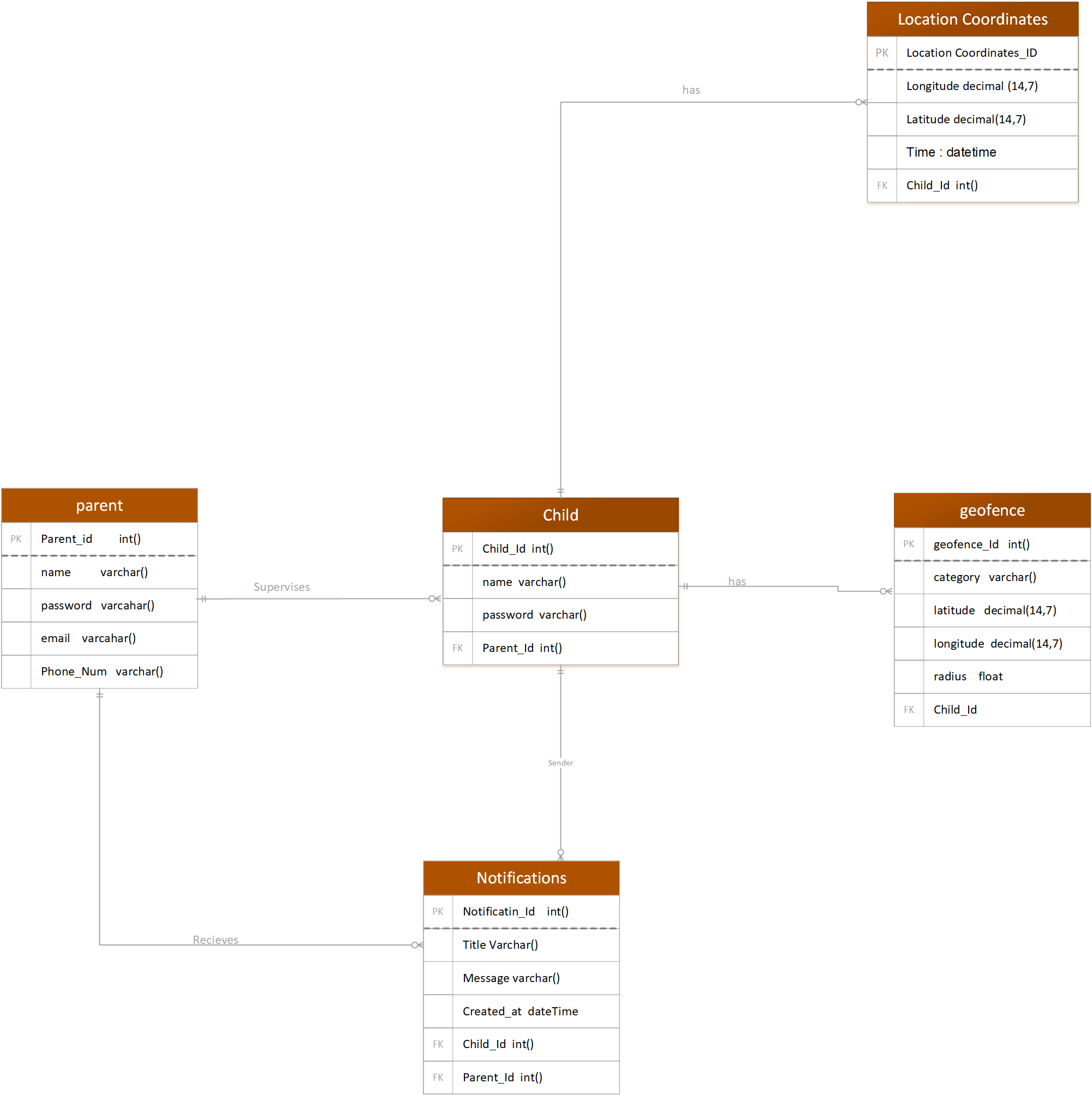
.



**Figure 4.2 : Class Diagram**

## 4.2.1.2 ERD Diagram

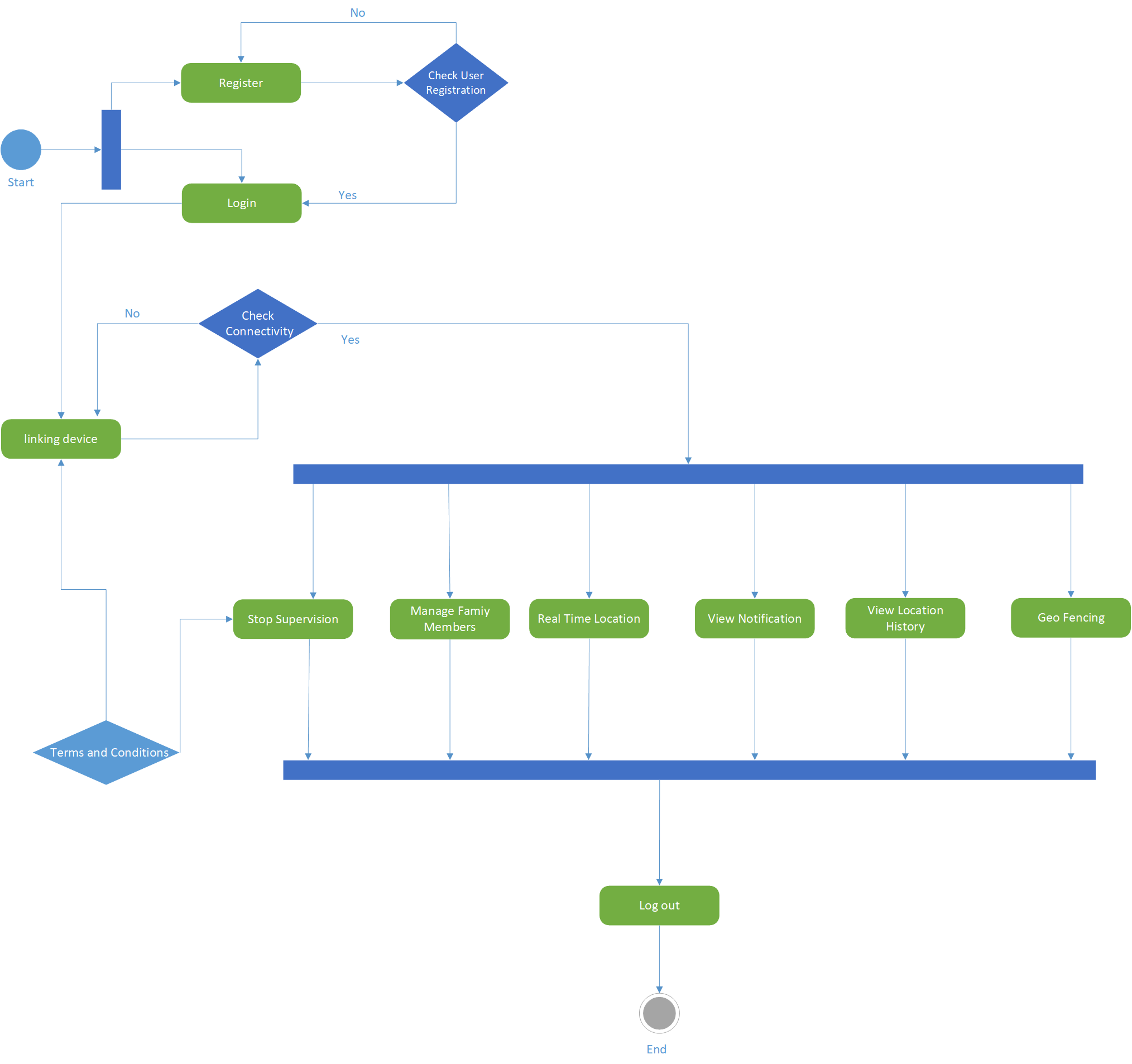
The ERD diagram that consist of 5 tables, which is Parent, Child, Location Coordinates, Notifications and Geo fence table is shown in Figure. The relationship between parent object and child object is One to Many relationship as the parent can have more than one child.



**Figure 4.3 : ERD Diagram**

## 4.2.1.3 Activity Diagram

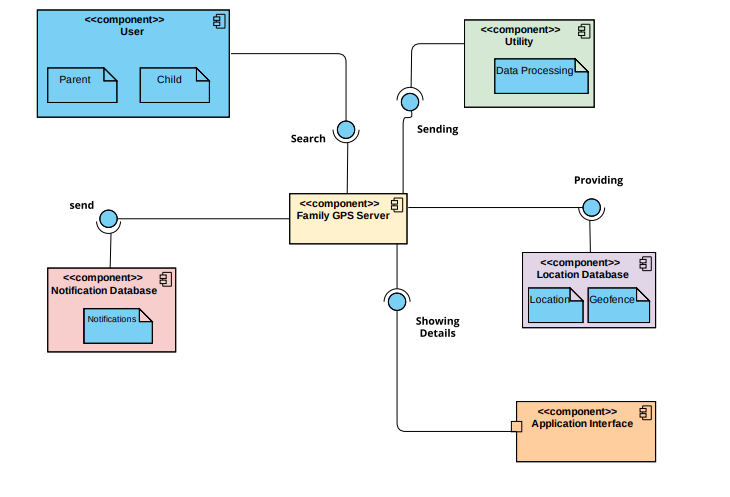
Activity diagrams are intended to model both computational and organizational processes. Activity diagrams show the overall flow of control. In this system, activity diagram is used to show the flow of activities for both users and administrators as shown in Figure. These activities include sign up and login, which allow them to perform other activities according to their role.



**Figure 4.4 : Activity Diagram**

## 4.2.1.4 Component Diagram

Component diagrams are used to model physical aspects of a system. Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.



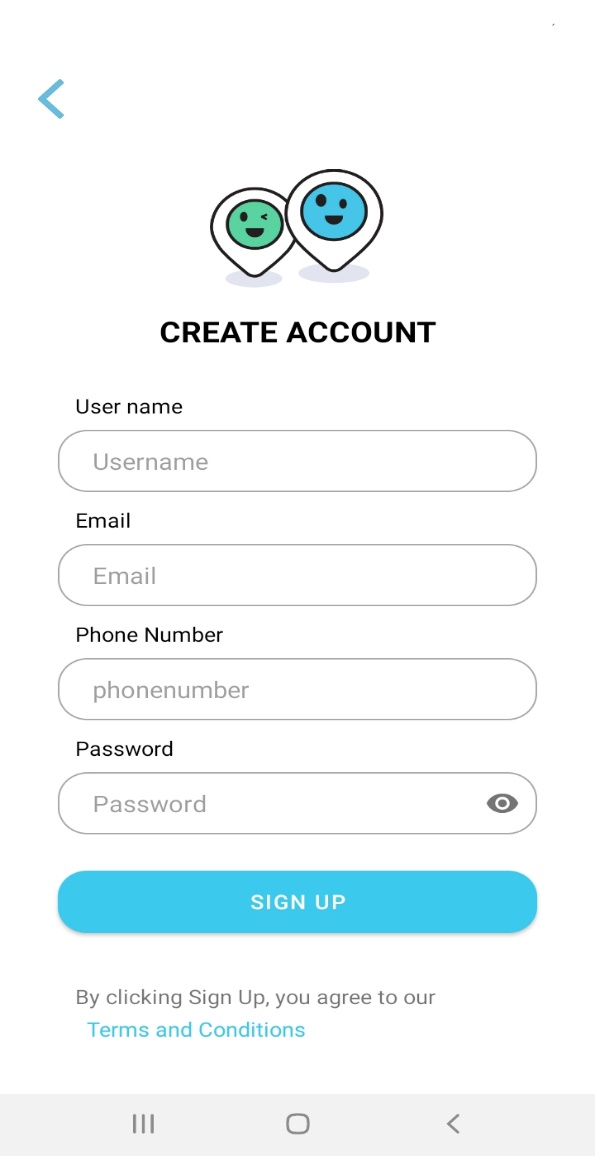
**Figure 4.5 : Component Diagram**

# Chapter 5 : Implementation

This chapter will discuss implementation details supported by UML diagrams (if applicable). You will not put your source code here. Any of the following sections may be included based on your project.

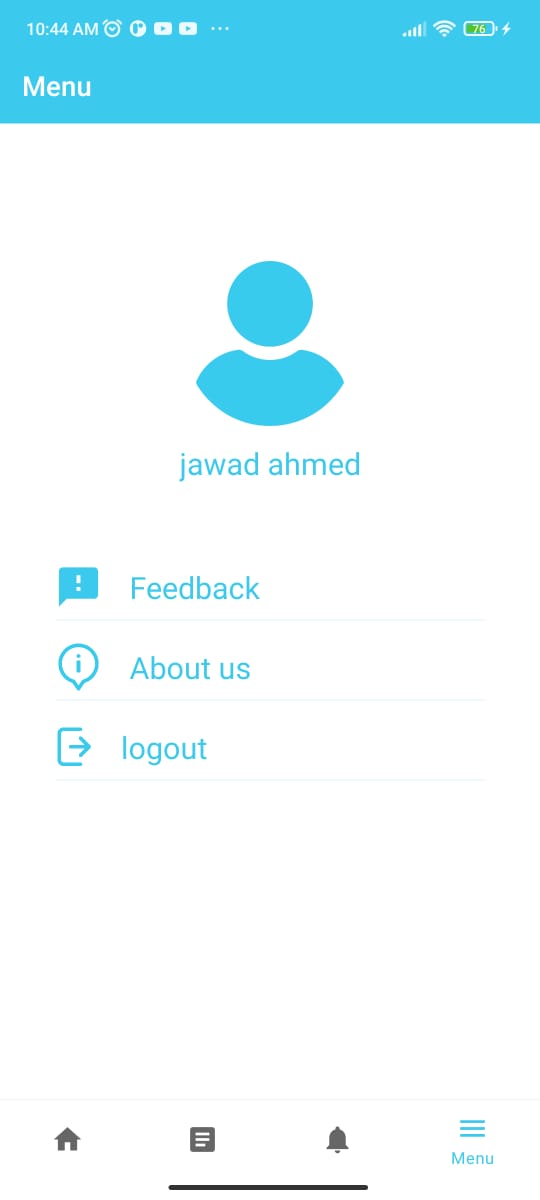
## 5.1 User Interface

**Sign-up**



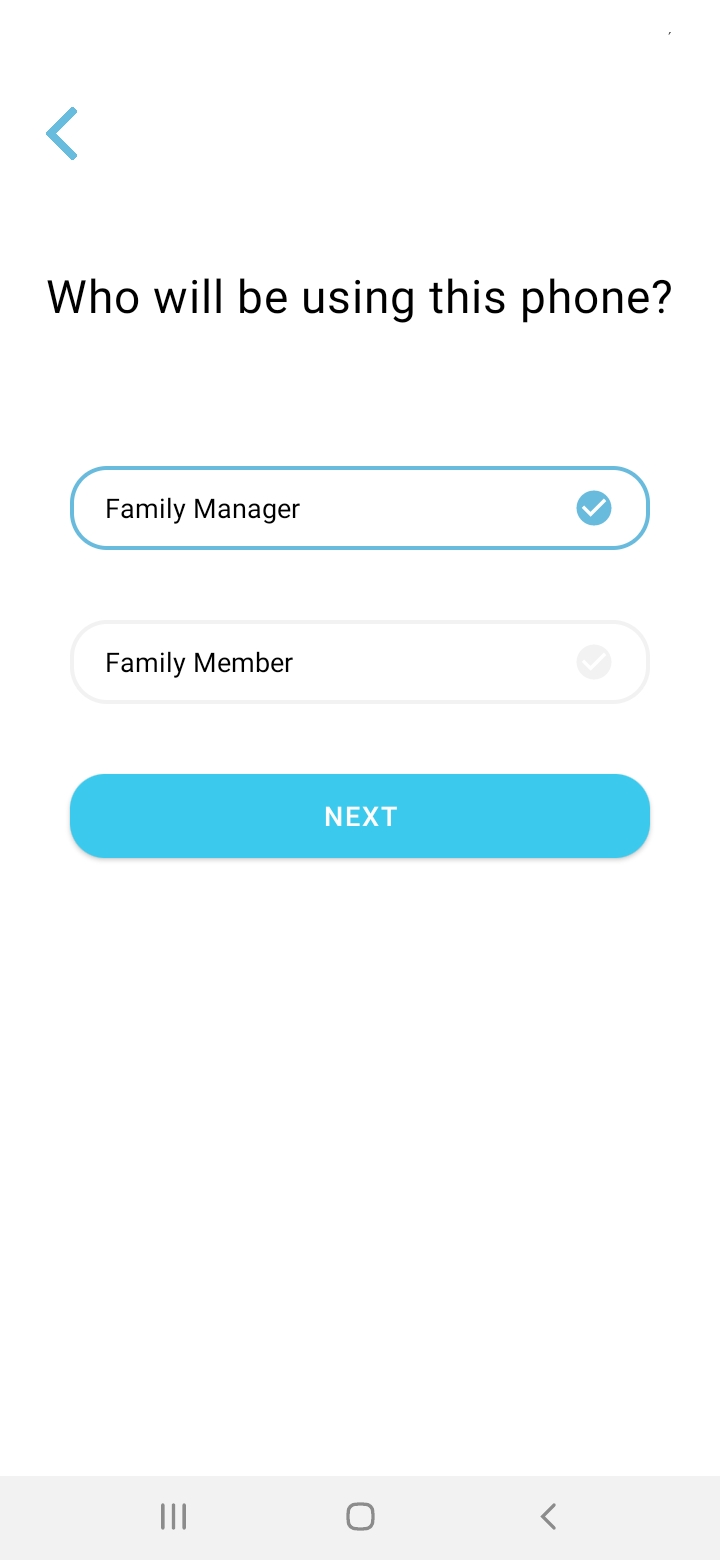
**Figure 5.1 : Sign Up Screen**

**Profile Screen**



**Figure 5.2: Profile Screen**

**Decide User Screen**

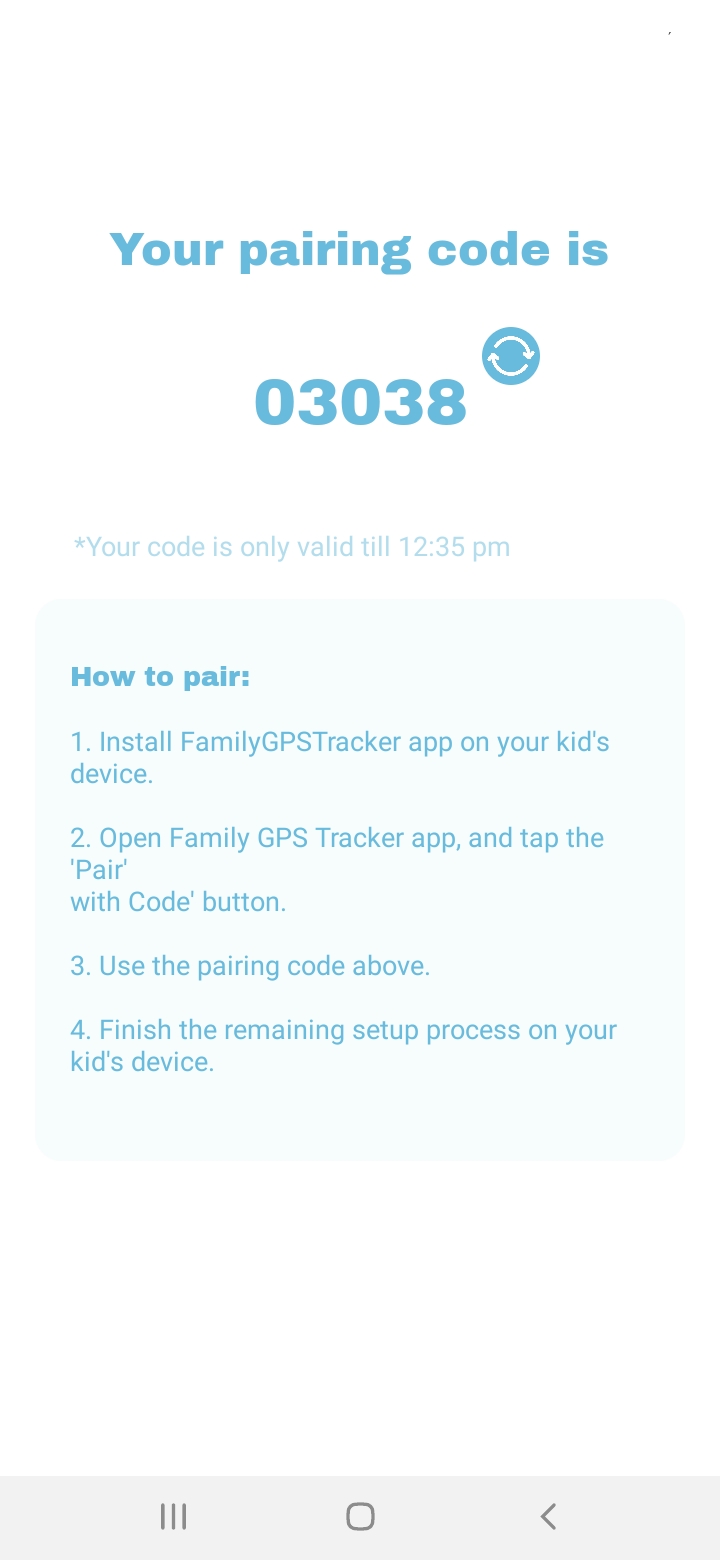


**Figure 5.1.3 : Decide User Screen**

**Location History Screen**

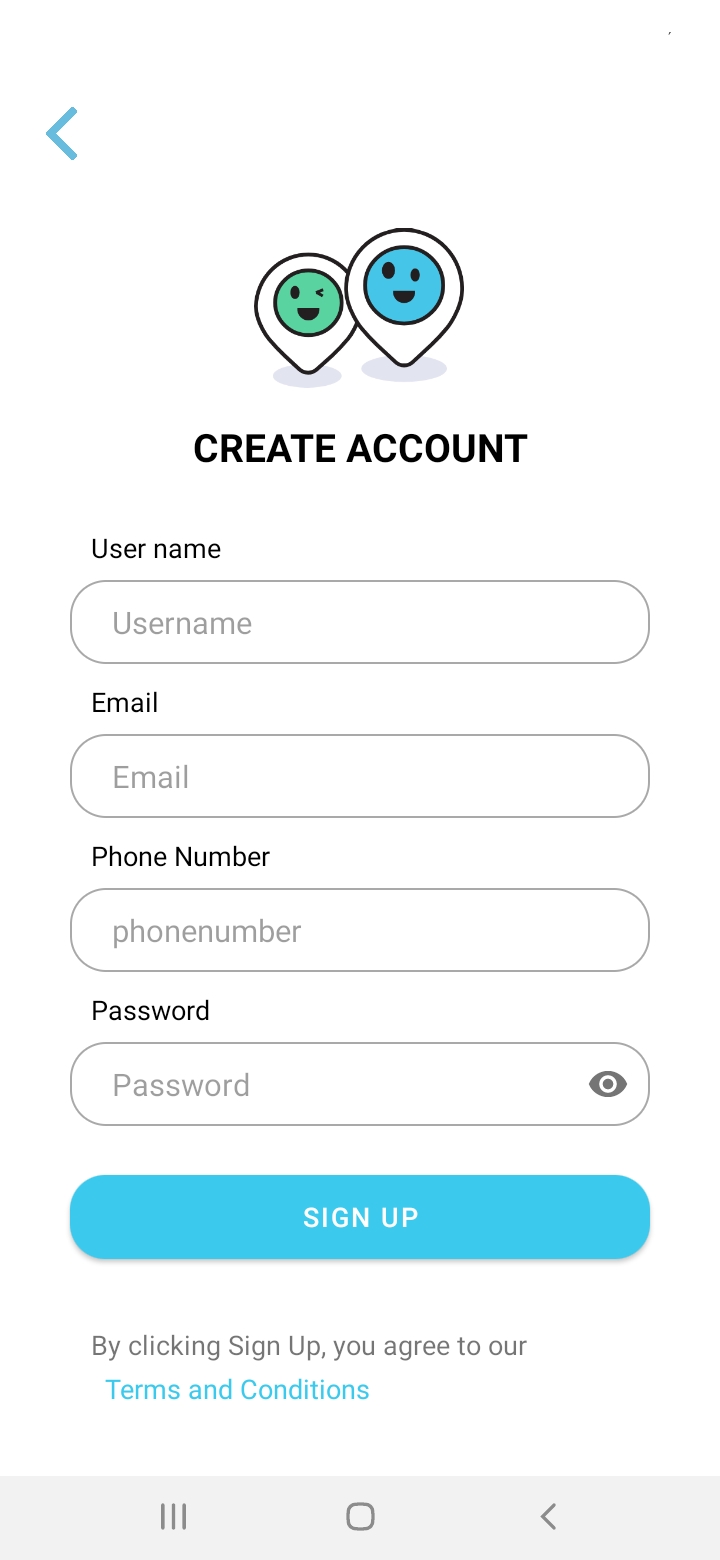
**Figure 5.4 : Location History Screen**

**Code Generation Screen**



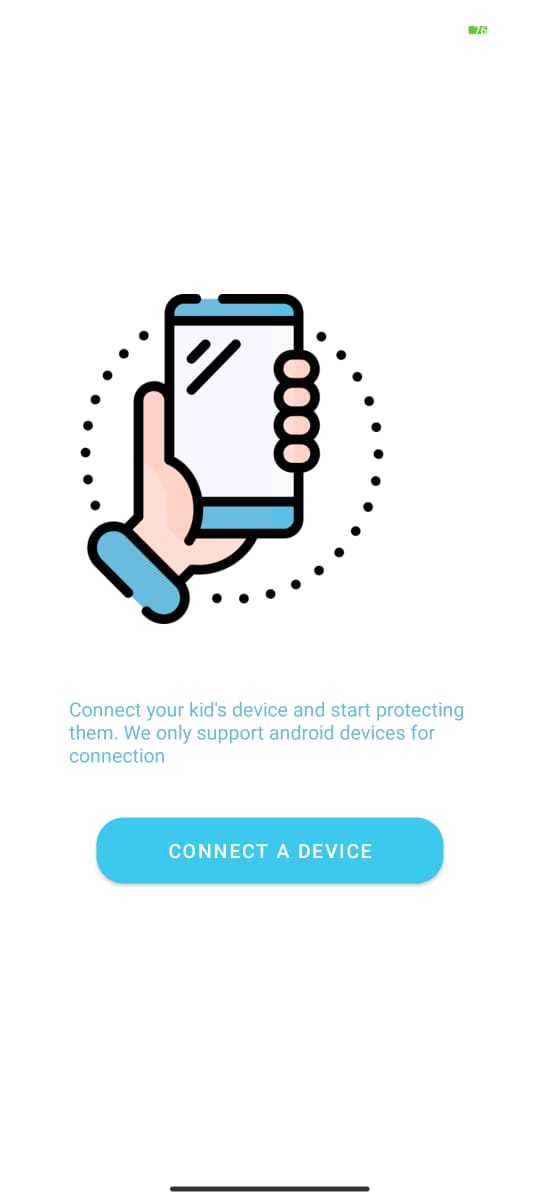
**Figure 5.5 : Code Generation Screen**

**Sign In Screen**



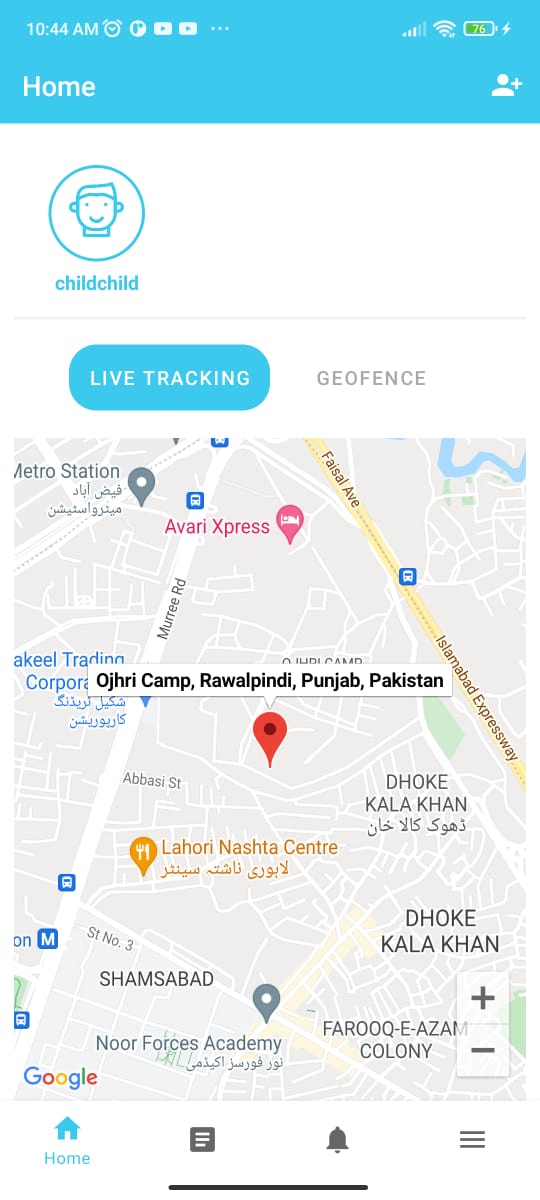
**Figure 5.6 : Sign In Screen**

**Device Connecting Screen**

****

**Figure 5.7 : Connect Device Screen**

**Home Screen**

****

**Figure 5.7 : Home Screen**

# Chapter 6 : Testing and Evaluation

## 6.1 Sign Up Module :

**Table 6.1 : Sign up Module**

|  |  |
| --- | --- |
| Tested By | Hassaan Ahmed |
| Test type | Unit testing |
| Test case number | 01 |
| Test case name | Sign up |
| Test case description | This test case will check sign up form details |
| **Procedural Steps** | **Procedural Steps** |
| 1 | User will fill sign up form |
| 2 | All information must be entered correct |
| 3 | User account will be created after storing the credentials on database server. |

## 6.2 Login Module

**Table 6.2 : Login Module**

|  |  |
| --- | --- |
| Tested By | Jawad Ahmed |
| Test type | Unit testing |
| Test case number | 02 |
| Test case name | Login |
| Test case description | This test case will check login info |
| **Procedural Steps** | **Procedural Steps** |
| 1 | User will enter email and password |
| 2 | On clicking login button user will be logged in |
| 3 | If not able to login, then first have to create account |

## 6.3 Location History Module

**Table 6.3 : Location History Module**

|  |  |
| --- | --- |
| Tested By | Jawad Ahmed |
| Test type | Unit testing |
| Test case number | 3 |
| Test case name | Location history |
| Test case description | This test case will check location history of family member |
| **Procedural Steps** | **Procedural Steps** |
| 1 | Linking both devices |
| 2 | Select a particular family member whom you want to track on real time basis. |
| 3 | Go to location history activity page to check for existing records |

## 6.4 Test Case for live location

**Table 6.4 : Live Location Module**

|  |  |
| --- | --- |
| Tested By | Hassaan Ahmed |
| Test type | Unit testing |
| Test case number | 4 |
| Test case name | Live location |
| Test case description | This test case will check live location of child |
| **Procedural Steps** | **Procedural Steps** |
| 1 | Linking both devices |
| **2** | Select a particular family member whom you want to track on real time basis. |
| **3** | After that family manager will get current live location of selected family member on Google map on real time basis |

## 6.5 Test Case for Notification Module :

**Table 6.5 : Notification Module**

|  |  |
| --- | --- |
| Tested By | Jawad Ahmed |
| Test type | Unit testing |
| Test case number | 5 |
| Test case name | Notification log |
| Test case description | This test will check notification history log |
| **Procedural Steps** | **Procedural Steps** |
| 1 | Go to notification log |
| 2 | family manager can see all the entire notifications |

## 6.6 Test case for Geofencing Module :

**Table 6.6 : Geofencing Module**

|  |  |
| --- | --- |
| Tested By | Jawad Ahmed |
| Test type | Unit testing |
| Test case number | 6 |
| Test case name | Geo fencing |
| Test case description | This test case will check geofences created by family manager |
| **Procedural Steps** | **Procedural Steps** |
| 1 | Create a virtual boundary on Google map |
| 2 | Whenever a family member breaches this virtual boundary, the family manager will be notified immediately |

## Chapter 7 : Conclusion and Future Work

# Conclusion

This project is basically an android tracking application for family members so that they are connected with each other. It has two main users family supervisor and child. The main aim of this application is to facilitate family members by monitoring their children and to reduce children/women abduction or missing cases. It helps to ensure that your loved ones are safe. It has many features like live location, create safe zone, geo fencing, panic button and smart alerts. Ensured of security is our main concern for which we will be providing two layered security through QR code scanning and password verification. It will overcome all the problems that were present in existing tracking applications. Our application is very easy to use and all its features are free of cost. In short this app provides peace of mind to parents and assures the safety of family members.

# Future Work

We are planning to do following advancements in our project in future.

* Implement search functionality
* Implement the dependency inject principle.
* Implement recommend screen.
* Improvements performance.
* Control and optimize of project budget.
* Improvements in the designing and performance of hardware as well as software.