رکھوالا “Rakhwala” (Child Security System)

BS(SE) Batch: 2017-2020

Submitted By:-

1. Muhammad Bilal Qamar(35985)
2. Mohammad Umer (36682)
3. Adeel Arshad (36498)

Supervised By

Dr Mansoor Ibrahim

Faculty of Engineering Science and Technology

IQRA University, Karachi

Abstract

It is a natural fact that parents keep worrying about their children whenever they are out of home especially if they are not an adult . When the child is out of home he/she cannot be traced by parents which is the biggest source of tension and worry for the parents and eventually the whole family. Not only is the child untraceable, he/she cannot be saved if he/she is injured, kidnapped, or Is in any emergency. Families are not yet combined on a single platform by which they can keep themselves in contact in any emergency scenario

Contents

Table of Contents

Chapter 1: **INTRODUCTION**

1.1 Introduction

Rakhwala is a child and in turn of a family security app which ensures parents about their child security specially . not only having superiority in child security as well as in child health monitoring. Rakhwala bears the potential to join a family over a single platform in emergency situations.

Uwatch is the first solution in Pakistan which is providing child security for the first time . It was first initiated by jazz but later on when it was not so popular then ufone came up with their name as **Uwatch** by purchasing this from jazz.

1.2 Project Objective

The main purpose of our app is to serve tensed parents and family members whenever their child/loved ones are out of home and they are very worried about them. This platform reduces their tension by providing them current location, and health updates for child and for family members within a single group. All of these qualities are available within a range of single touch (app).

1.3 Structure of the report

The report is organized as follows:

- Chapter 2: Literature review
- Chapter 3: System design
- Chapter 4: General Architecture
- Chapter 5: Methodology
- Chapter 6: Web Interface
- Chapter 7: Mobile Interface
- Chapter 8: Results and Analysis
- Chapter 9: Conclusion

Chapter 2: **TECHNOLOGY BACKGROUND**

It is a natural fact that parents keep worrying about their children whenever they are out of home especially if they are not an adult. When a child is out of home he/she cannot be traced by parents which is the biggest source of tension and worry for the parents and eventually the whole family. Not only is the child untraceable, he/she cannot be saved if he/she is injured, kidnapped, or Is in any emergency. Families are not yet combined on a single platform by which they can keep themselves in contact in any emergency scenario. No Health monitoring is provided by the current market solution U-Watch it just traces the location of the child .

Following technologies, which we are using to develop this project:

2.1 Mobile App :

React-Native :-

React Native is a JavaScript framework for writing real, natively rendering mobile applications for iOS and Android. It's based on React, Facebook's JavaScript library for building user interfaces, but instead of targeting the browser, it targets mobile platforms. In other words: web developers can now write mobile applications that look and feel truly "native," all from the comfort of a JavaScript library that we already know and love. Plus, because most of the code you write can be shared between platforms, React Native makes it easy to simultaneously develop for both Android and iOS.

Similar to React for the Web, React Native applications are written using a mixture of JavaScript and XML-esque markup, known as JSX. Then, under the hood, the React Native "bridge" invokes the native rendering APIs in Objective-C (for iOS) or Java (for Android). Thus, your application will render using real mobile UI components, *not* webviews, and will look and feel like any other mobile application. React Native also exposes JavaScript interfaces for platform APIs, so your React Native apps can access platform features like the phone camera, or the user's location.

React Native currently supports both iOS and Android, and has the potential to expand to future platforms as well. In this book, we'll cover both iOS and Android. The vast majority of the code we write will be cross-platform. And yes: you can really use React Native to build production-ready mobile applications! Some anecdota: Facebook, Palantir, and TaskRabbit are already using it in production for user-facing applications.

Reactive Native: The pros

React Native has its standard rendering API to render applications. Compared with other cross-platform development frameworks like Cordova and Ionic, React Native stands above the crowd. The framework is easy to work with, providing the developer with a useful set of developer tools, and meaningful troubleshooting messages. This makes React Native a robust framework that improves the developer experience. Several performance issues arise when mobile applications are rendered using webviews rendering. React Native transforms your markup, filling the markup with real and native UI elements. As far as performance is concerned, since React Native works from the main UI branch, the applications will usually not face any performance issues. Because of its iteration speed, it allows developers to share code knowledge and efficiently utilize their resources

React Native: The cons

The use of the React Native is highly dependent on the requirements of your team, and whether or not it fits with your overall development process. The project is still young, meaning that it has not reached the level of maturity that developers would render it a universal framework for developing mobile applications. Because of its age, the documentation still has room for improvement. Developers tend to face many difficulties if the documentation is updated continuously. The best features and practices of React Native are still being tested. Again, its constant evolution as a framework inhibits its standing as a dedicated framework. The debugging process of React Native can become rather complicated since it adds another layer to your project. The point where the React Native and host platform intersects is particularly tricky to debug.

2.2 Server Side

2.1 Node.JS

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent. Node.js library is very fast in code execution.

Asynchronous and Event Driven :− All APIs of Node.js library are asynchronous, that is, non-blocking. It essentially means a Node.js based server never waits for an API to return data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.

Single Threaded but Highly Scalable :− Node.js uses a single threaded model with event looping. Event mechanism helps the server to respond in a non-blocking way and makes the server highly scalable as opposed to traditional servers which create limited threads to handle requests. Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.

Node.js applications never buffer any data. These applications simply output the data in chunks.

Following are the areas where Node.js is proving itself as a perfect technology partner.

- I/O bound Applications
- Data Streaming Applications
- Data Intensive Real-time Applications (DIRT)
- JSON APIs based Applications
- Single Page Applications

2.3 Data Base

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

3.1 MySQL

Structured Query Language (SQL) is a standard computer language for relational database management and data manipulation. SQL is used to query, insert, update and modify data. Most relational databases support SQL, which is an added benefit for database administrators (DBAs), as they are often required to support databases across several different platforms.
Categories
Data Definition Language (DDL) :
It is used to define the structure of the database. e.g; CREATE TABLE, ADD COLUMN, DROP COLUMN and so on.
Clauses:
The SELECT statement is the omnipresent part of the statements to

perform queries. It is further divided into clauses, which include SELECT, FROM, WHERE and ORDER BY.

Data Manipulation Language (DML):
Commands which are used to manipulate the data in a database, like add, update or delete data. It comes under the Data Manipulation Language (DML). DML consists of statements like SELECT, INSERT, DELETE, and UPDATE along with BEGIN TRANSACTION, SAVEPOINT, COMMIT and ROLLBACK as some of the control statements.

Data Control Language (DCL):
The GRANT and REVOKE statements are two main statements used to assign and revoke database rights and permissions. These commands make up the Data Control Language (DCL).

Uses of SQL

SQL is used in health care (cancer registries) business (inventories, trends analysis), and education. It even has applications in the defense industry

MySQL databases are relational :

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and “pointers” between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

Nowadays, we use relational database management systems (RDBMS) to store and manage huge volumes of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as Foreign Keys.

MySQL Server works in client/server or embedded systems.

The MySQL Database Software is a client/server system that consists of a multithreaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

2.4 IOT

Internet of Things (IoT) :

The internet of things (IoT) is a computing concept that describes the idea of everyday physical objects being connected to the internet and being able to identify themselves to other devices. The term is closely identified with RFID as the method of communication, although it also may include other sensor technologies, wireless technologies or QR codes.

OR

“The Internet of Things (IoT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.”

The IoT is significant because an object that can represent itself digitally becomes something greater than the object by itself. No longer does the object relate just to its user, but it is now connected to surrounding objects and database data. When many objects act in unison, they are known as having "ambient intelligence."

Internet of Things covers the broad categories of devices that are connected to the internet. These applications have spread to both the customer and industrial domains. In the area of safety and customer experience, IoT is witnessing exponential demands.

The Internet of Things extends internet connectivity beyond traditional devices like desktop and laptop computers, smartphones and tablets to a diverse range of devices and everyday things that utilize embedded technology to communicate and interact with the external environment, all via the Internet.

Every second over 127 new devices will be connected to the internet, according to David Evan’s calculations (former researcher of CISCO). This gives an idea about the constantly increasing reach of IoT.

Over 90% automobiles by 2020 will be IoT enabled as per PWC estimates. Moreover, according to the data found by Statista, there were around 23.14

billion devices in 2018 and the number will reach to 26.66 billion by 2019

IoT Examples :

Examples of objects that can fall into the scope of Internet of Things include connected security systems, thermostats, cars, electronic appliances, lights in household and commercial environments, alarm clocks, speaker systems, vending machines and more.

Exp :

A lightbulb that can be switched on using a smartphone app is an IoT device, as is a motion sensor or a smart thermostat in your office or a connected streetlight. An IoT device could be as fluffy as a child's toy or as serious as a driverless truck. Some larger objects may themselves be filled with many smaller IoT components, such as a jet engine that's now filled with thousands of sensors collecting and transmitting data back to make sure it is operating efficiently. At an even bigger scale, smart cities projects are filling entire regions with sensors to help us understand and control the environment.

The birth of IoT :

The term Internet of Things is 16 years old. But the actual idea of connected devices had been around longer, at least since the 70s. Back then, the idea was often called “embedded internet” or “pervasive computing”. But the actual term “Internet of Things" was coined by Kevin Ashton in 1999 during his work at Procter & Gamble. Ashton, who was working in supply chain optimization, wanted to attract senior management’s attention to a new exciting technology called RFID. Because the internet was the hottest new trend in 1999 and because it somehow made sense, he called his presentation “Internet of Things".

IoT takes off :

The concept of IoT started to gain some popularity in the summer of 2010. Information leaked that Google’s StreetView service had not only made 360 degree pictures but had also stored tons of data of people’s Wifi networks. People were debating whether this was the start of a new Google strategy to not only index the internet but also index the physical world.

The same year, the Chinese government announced it would make the Internet of Things a strategic priority in their Five-Year-Plan.

Detail of IOT :

The Internet of Things (IoT) has not been around for very long. However, there have been visions of machines communicating with one another since the early 1800s. Machines have been providing direct communications since the telegraph (the first landline) was developed in the 1830s and 1840s. Described as “wireless telegraphy,” the first radio voice transmission took place on June 3, 1900, providing another necessary component for developing the Internet of Things. The development of computers began in the 1950s.

The Internet, itself a significant component of the IoT, started out as part of DARPA (Defense Advanced Research Projects Agency) in 1962, and evolved into ARPANET in 1969. In the 1980s, commercial service providers began supporting public use of ARPANET, allowing it to evolve into our modern Internet. Global Positioning Satellites (GPS) became a reality in early 1993, with the Department of Defense providing a stable, highly functional system of 24 satellites. This was quickly followed by privately owned, commercial satellites being placed in orbit. Satellites and landlines provide basic communications for much of the IoT.

One additional and important component in developing a functional IoT was IPV6’s remarkably intelligent decision to increase address space.

By the year 2013, the Internet of Things had evolved into a system using multiple technologies, ranging from the Internet to wireless communication and from micro-electromechanical systems (MEMS) to embedded systems. The traditional fields of automation (including the automation of buildings and homes), wireless sensor networks, GPS, control systems, and others, all support the IoT.

National Strategies :

The Internet of Things offers many opportunities to grow the economy and improve quality of life. Just as the public sector was instrumental in enabling the development and deployment of the Internet, it must play a similar role to ensure the success of the Internet of Things. Therefore, national governments should create comprehensive national strategies for the Internet of Things to ensure that the technology develops cohesively and rapidly, that consumers and businesses do not face barriers to adoption, and that both the private and public sector take full advantage of the coming wave of smart devices.

The Future of IoT :

As far as the reach of the Internet of Things, there are more than 12 billion devices that can currently connect to the Internet, and researchers at IDC

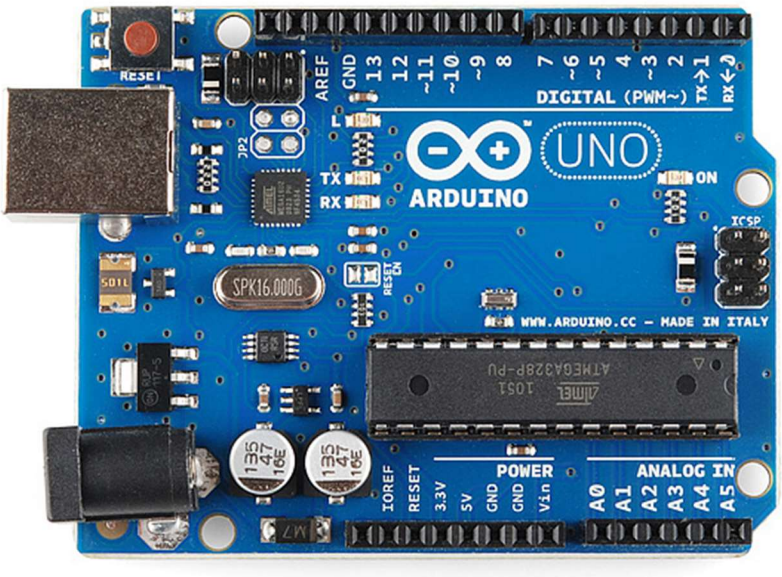
estimate that by 2020 there will be 26 times more connected things than people.According to Gartner, consumer applications will drive the number of connected things, while enterprise will account for most of the revenue. IoT adoption is growing, with manufacturing and

utilities estimated to have the largest installed base of Things by 2020.

2.5 **IOT - Arduino :**

Arduino is a user based community that is indulged in the design, development and sales of open-source software, open source hardware, and microcontroller-based kits that are used in the manufacturing of interactive objects and digital devices that have an ability of sensing and controlling physical devices.

The project is formed on the basis of microcontroller board designs that have been developed by a number of vendors through the deployment of different microcontrollers. An integrated development environment is provided by the Arduino project for the development of these microcontrollers. This environment is based on Arduino project's programming language that supports both C as well as C++.



Chapter 3: REQUIREMENTS & METHODOLOGY

3.1 Introduction

Rakhwala is a child and in turn a family security app which ensures parents about their child security specially . not only having superiority in child security as well as in child health monitoring. Rakhwala bears the potential to join a family over single platform in emergency situations. We are using the state of the art languages and tools such as react native,node js/python,IOT(Arduino) .

3.2 Proposed Solution:-

The main purpose of our app is to serve tensed parents and family members whenever their child/loved ones are out of home and they are very worried about them. This platform reduces their tension by providing them current location, and health updates for children and for family members within a single group. All of these qualities are available within a range of single touch (app)
We are using popular frameworks and languages such as (React Native, Node.js, Arduino MySQL, Html,, Css, JavaScript.
We will host our server and on heroku as cloud hosting and our server will be hosted on a domain like (www.rakhwala-herokuapp.com).

3.4 Functional Requirement Of Rakhwala:

- Users should be able to register in the system.
- User should login to the system.
- System should send the real time location of the child to the server.
- System should send the temperature and send the pulse rate of the child to the server.
- Application should show real time location and health status in the form of pulse rate and temperature by clicking an status button.
- User should be able to create a family circle group and add any family member by sending a joining request.
- Every group should show joined members real time location with their image on google map.
- Location should be auto updated in the database when users move from one location to another.
- If any of the group members are in panic,there is an alarm button which will send the push notification to all of the group members that the member (ABC-) is in panic.

3.5 Type of Roles (USERs):

- Parent / User
- child
- Admin

3.6 Non-Functional Requirements Of Rakhwala

- Users shall receive notification of profile changes via preferred communication method of record when profile information is modified.
- Passwords shall never be viewable at the point of entry or at any other time.
- The system for child security purpose should be accessible for only the peoples who has IOT device.
- Unless the system is non-operational, the system shall present a user with notification informing them that the system is unavailable.
- At least 20 percent of the processor capacity and storage space available to the system.
- People with no training shall be able to use the system.
- The application development process must have a regression test procedure that allows complete re-testing.
- The system shall not be shut down for maintenance more than once in a 24-hour period.

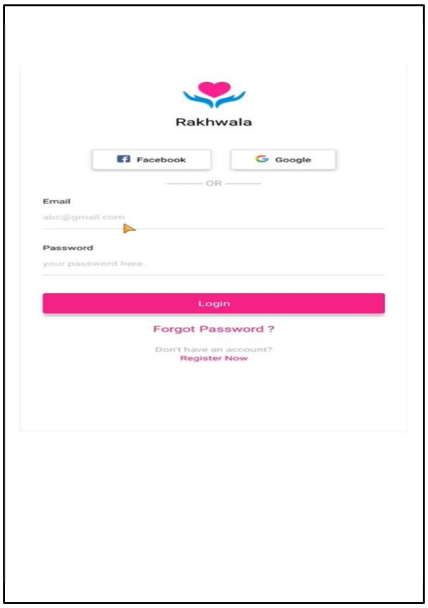
Chapter 4: PROJECT PLANNING & INITIAL DESIGN.

4.1 Proposed Timeline:

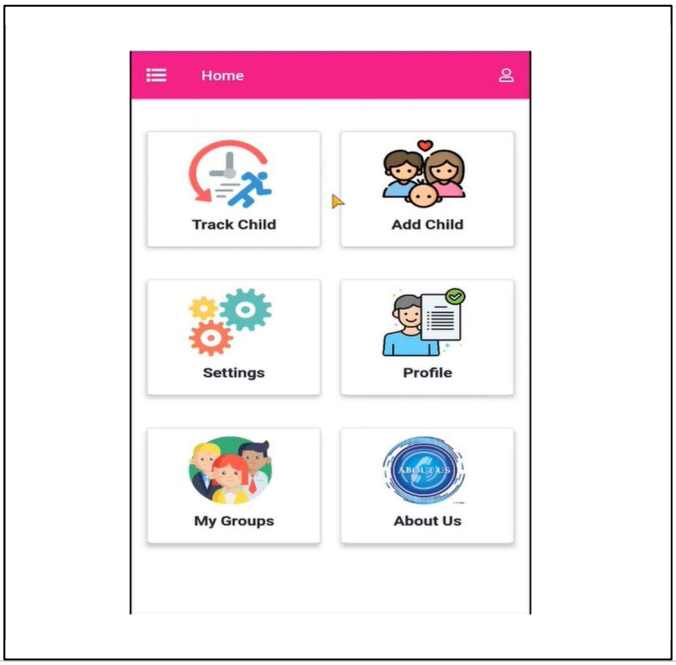
The following task were distributed on the basis of per week and with description.

WEEK	TASK	DESCRIPTION
1	Develop Database Schema Design	develop Database Schema (all-tables) related to each activity described such as 1. Child Parent registration, 2. Family group creation 3. Insert update delete operations.
2	Develop Rest APIS	Too develop Rest apis for communication our app with database
3	Start Front-End	Designing Front-end screens
4	Front-End	Develop remaining front-end screens
5	Set-up GPS Modules	Setup gps module . take location in the form of coordinates(latitude & amp; longitude) and send it to the server.
6	Setup temperature and pulse rate sensors	Setup temperature and pulse rate sensors take reading from pins and send these readings to the server.
7	Merge Arduino based code	Merge all the Arduino based code and test it accordingly. Successfully tested arduino based code.
8	Integrate arduino	Integrate arduino with android app.

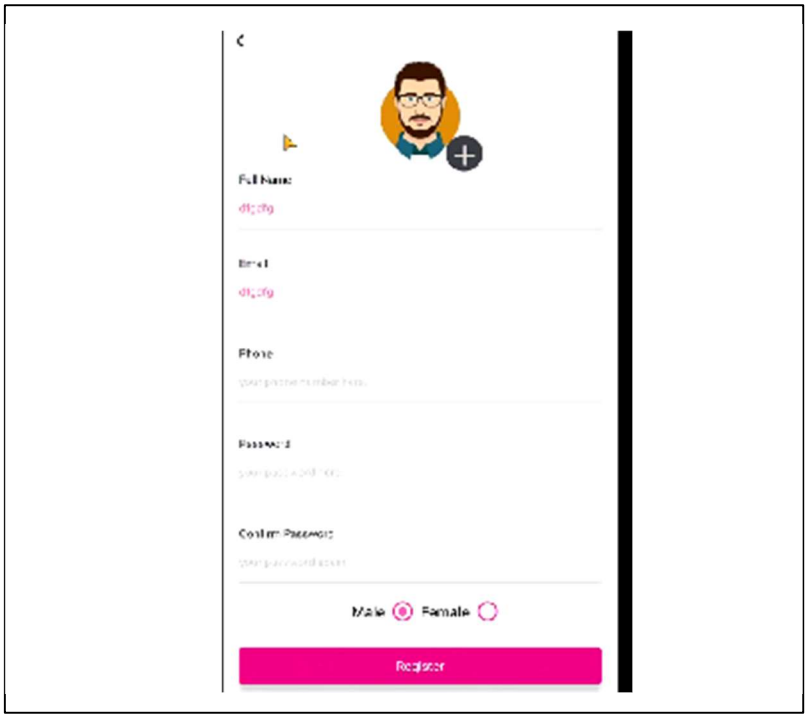
Login:



Initial Dashboard:



Initial Manage Profile Page:



Challenges faced during planning and designing:

The main challenge that we face of this project is to provide relief to the family and parents in a situation where there is no other tool to trace down where the family member, child is and how well he/she is feeling.

Overcome Challenges:

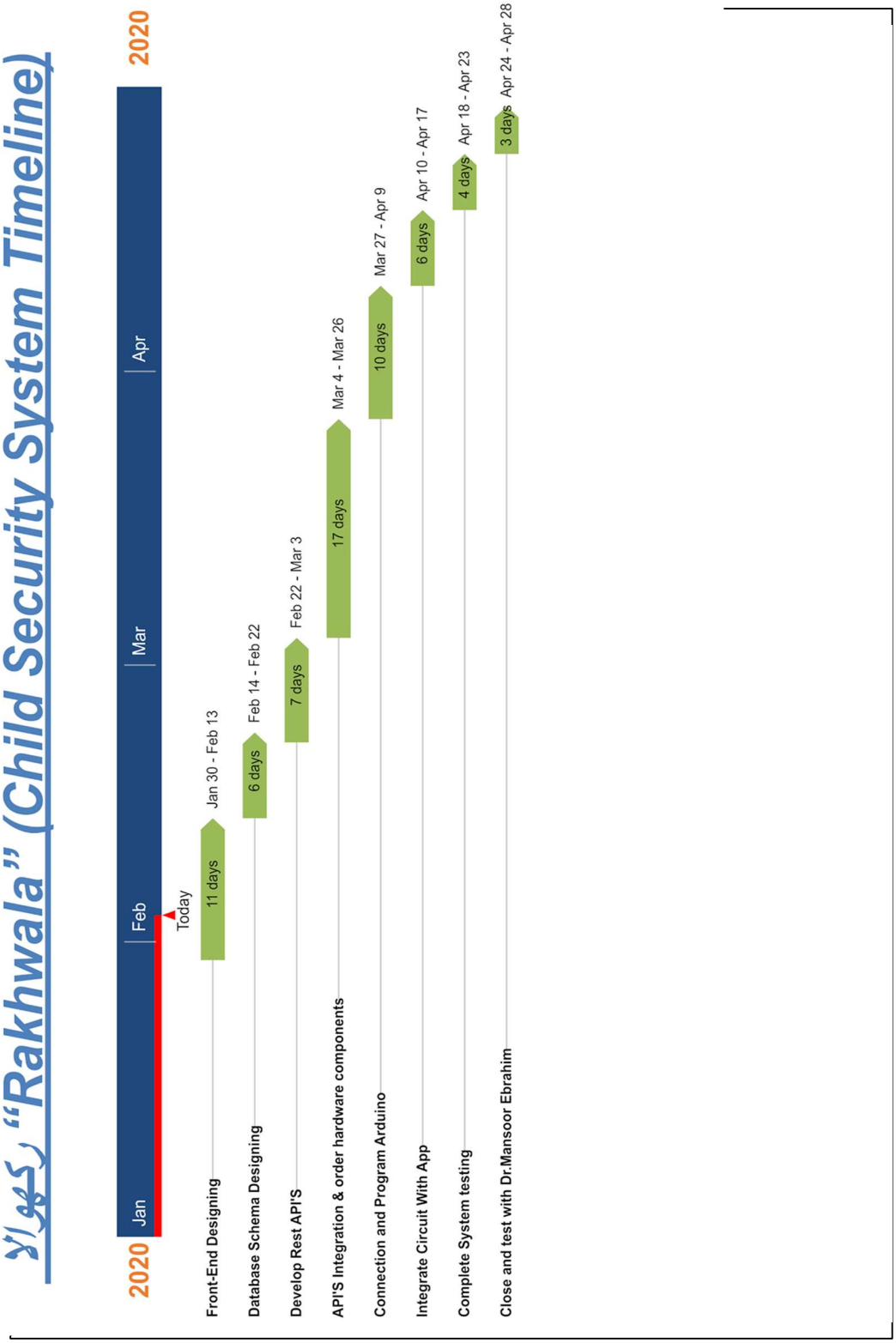
Water resistance, covering childrens of all age, provide Health Monitoring, avoiding call option , Lcd displays the info of the person to whom we are calling which can alert the kidnapper so it is eliminated in our solution,

4.2 Analysis Chart task Distribution and Coordination:

The challenges were resolved by the following coordination in %

Tasks Affected (Delayed/Errors)	**Mohammad Umer (36682)**	**Bilal Qamar (35985)**	**Adeel Arshad (36498)**
Initial Database Modification	40%	40%	20%
Frontend Design Delayed (Website & Mobile)	50%	40%	10%
Delayed Backend Development (Initial Website Portal)	50%	50%	0%
Errors and Bugs Resolution (Initial website portal)	40%	40%	20%
Error in Mobile Design	50%	50%	0%

4.3 initial Gantt Chart:

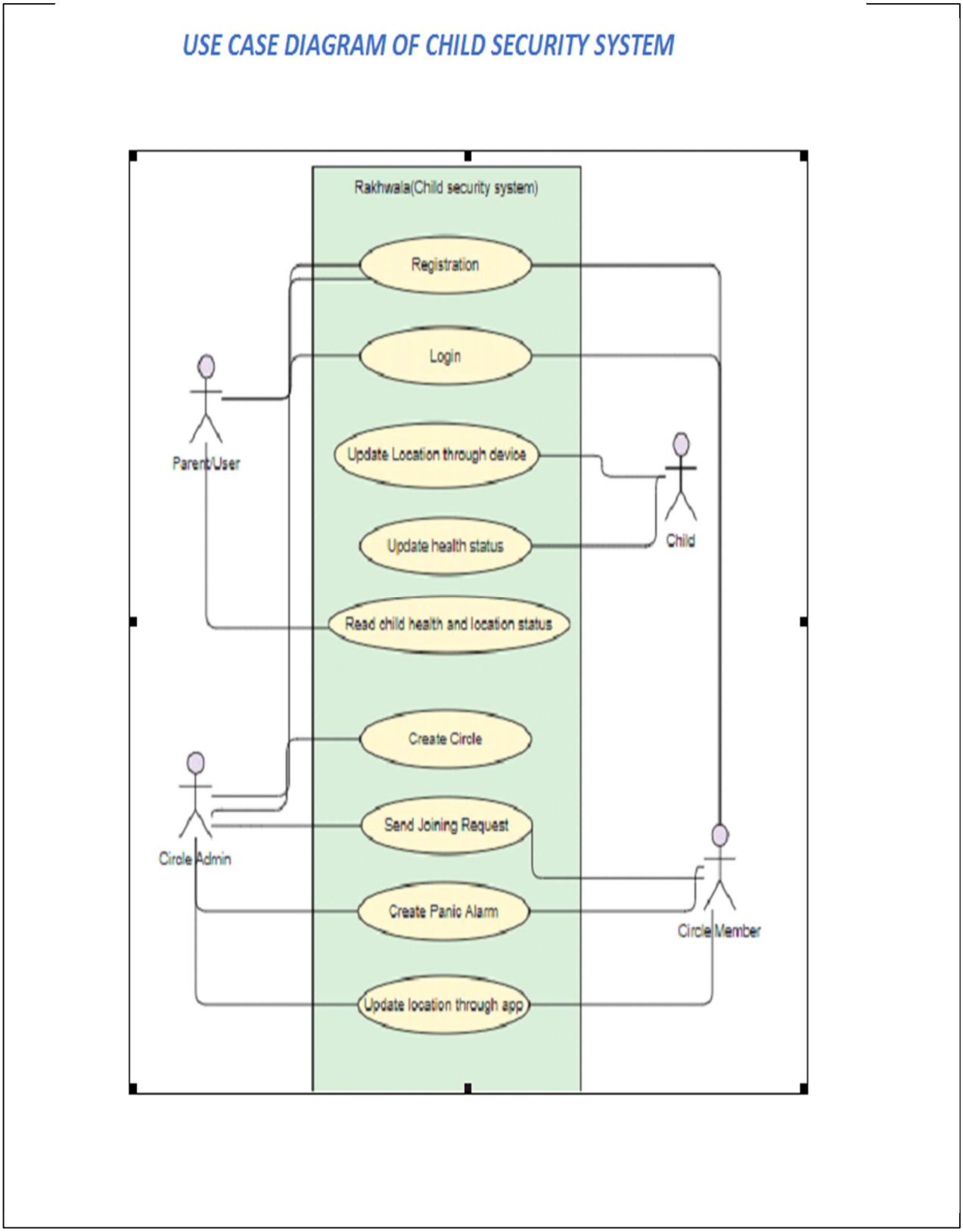


4.4 Modified Things In Gantt Chart :

	Database	Web & Mobile Frontend	Web Backend	Mobile Backend	Testing
Start	1/27	2/3	2/16	2/29	1/27
Duration	6.1	11.9	12.3	18.3	52

- Database development and implementation took place on 1/27/2020 and completed in its proposed time.
- Frontend Development took place on the proposed date and ended at the proposed deadline
- Web backend should complete in the proposed time
- Mobile app should complete in the proposed time
- Testing took place from the starting till the end to ensure quality.

4.5 Initial Use case Diagram:



4.6 Narratives:

1.1 User Login

Use Case Name:	User Login	
ID:	Rakhwala.UC.1.1.1	
Actors Involved:	Student, Facilitator & Administrator	
Brief Description	User admin or parents will login to the system to activate further events. Or access app features	
Pre-Conditions	User must be Registered to the system.	
Post-Conditions	User Activity Screen	
Normal Flow of Events:	**Actor Action**	**System Response**
	1. Provide email and password. 2. Click on login button.	1. System should validate email and password. 2. Dashboard will appear.

1.2: User Register

Use Case Name:	User Registration.	
ID:	Rakhwala.UC.1.1.2	
Actors Involved:	User	
Brief Description	User will register to the app by providing neccessory information like name, email, password	
Pre-Conditions	Must not be register to the app before.And should be a new user	
Post-Conditions	Will be register to the app and allowed to take benefit from system	
Normal Flow of Events:	**Actor Action**	**System Response**
	Provide email, name password,child name, image, and click on the register button.	System should validate all the data and do register successfully

1.3:Update Location

Use Case Name:	Update Location	
ID:	Rakhwala.UC.1.1.3	
Actors Involved:	Child, or user	
Brief Description	Whenever a child moves, his real time location(coordinates:{latitude & longitude}) will update in the database through an IOT device(Watch).	
Pre-Conditions	Child or user must have an IOT device on hand.	
Post-Conditions	Update coordinates after particular interval of time	
Normal Flow of Events:	**Actor Action**	**System Response**
	Provide latitude, name longitude,child name, image, through device	Get latitude and longitude through device and update it on live server

1.4:Read child location

Use Case Name:	Post Updates / Request Other Users.	
ID:	Rakhwala.UC.1.1.4	
Actors Involved:	Parents(or may be a family member).	
Brief Description	Fetch location of the child from the database and display it on map in application.	
Pre-Conditions	To request other users the Featured Sport Facility must be available.	
Post-Conditions	Check status about the child location.	
Normal Flow of Events:	**Actor Action**	**System Response**
	Do request for location	After some interval of time application would display the child real time location

1.5: Update Health Status

Use Case Name:	Update Health Status	
ID:	Rakhwala.UC.1.1.5	
Actors Involved:	Child	
Brief Description	Pulse rate sensor and Temperature sensor will get the health status of the child and send it to the server through IOT device. users.	
Pre-Conditions	Child or user must have an IOT device and sensors must works properly.	
Post-Conditions	Update status in database after particular interval of time.	
Normal Flow of Events:	**Actor Action**	**System Response**
	Provide temperature and pulse rate through sensors.	Send it to database

1.6: Read Health Status

Use Case Name:	Read Health Status	
ID:	Rakhwala.UC.1.1.6	
Actors Involved:	Parents or a family member	
Brief Description	Fetch temperature and pulse rate of the child from the database and display it in application.	
Pre-Conditions	Devices and sensors must be active and update status continuously after a particular period of time.	
Post-Conditions	Check status about the child's health..	
Normal Flow of Events:	**Actor Action**	**System Response**
	Request for health status	After some interval of time application would display the child updated status in the form of pulse rate and temperature

1.7: Create Family Circle

Use Case Name:	Create Family Circle	
ID:	Rakhwala.F2.1.1	
Actors Involved:	Admin, User	
Brief Description	Admin/User will create a family circle (group) by providing a circle name and image and send joining requests to other family members.	
Pre-Conditions	Admin/User must be logged into the application.	
Post-Conditions	Admin must provide a circle name and image in order to create new one.	
Normal Flow of Events:	**Actor Action**	**System Response**
	Provide circle name, image and click on create button	Circle will have been successfully created

1.8 : Create Panic Alarm

Use Case Name:	Create Panic Alarm	
ID:	Rakhwala.F2.1.2	
Actors Involved:	Admin or circle member	
Brief Description	If the circle member is in panic condition then there is a button called panic alarm. By pressing it ,push notification will send to every group member that member is in panic..	
Pre-Conditions	Send panic notification to other group members	
Post-Conditions	Admin must provide a circle name and image in order to create a new one.	
Normal Flow of Events:	**Actor Action**	**System Response**
	Press Alarm button	all-members get notification, also member location will be marked in the form of red circle and display it on map

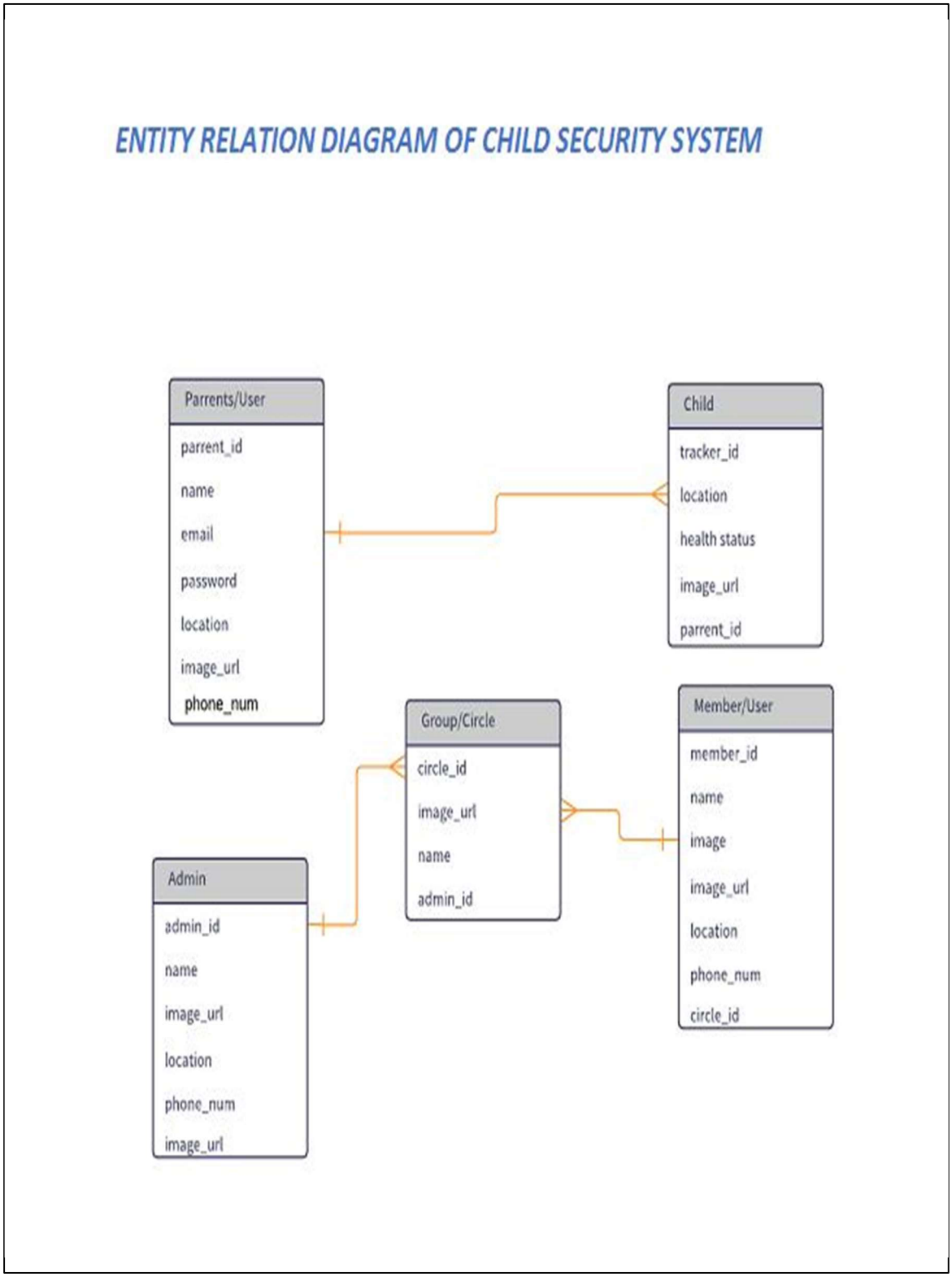
1.9 : Read Circle Members Location

Use Case Name:	Read Circle Members Location	
ID:	Rakhwala.F2.1.3	
Actors Involved:	Circle member(user)	
Brief Description	Fetch real time location and image of the members from the database and display it on map in application	
Pre-Conditions	User must be member of that particular circle and gps and internet of mobile should be on	
Post-Conditions	Update the real time location of every member in the database whenever they can move .	
Normal Flow of Events:	**Actor Action**	**System Response**
	● Open particular circle ● check where members are present through maps	● Display members on map ● Members location will display in the form of marker with picture

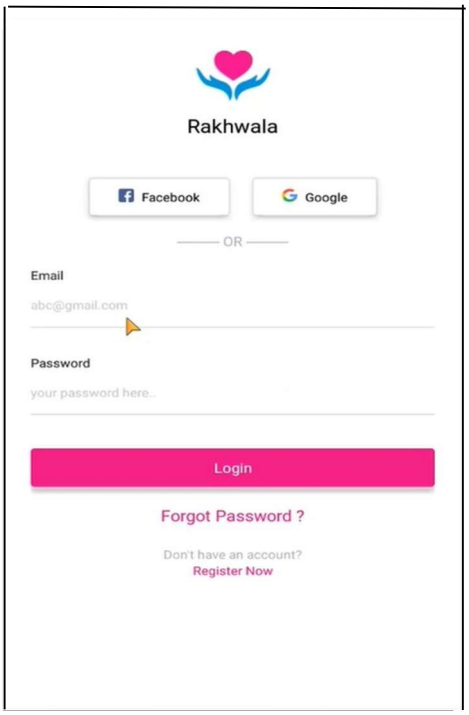
2.0 : Read Circle Members Location

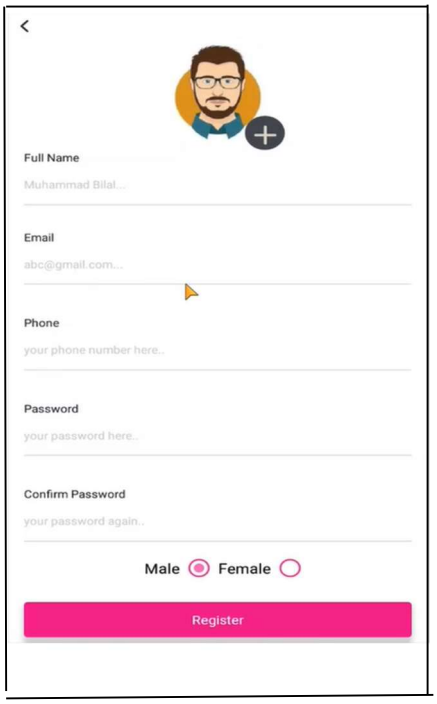
Use Case Name:	Read Circle Members Location	
ID:	Rakhwala.F2.1.3	
Actors Involved:	Admin or circle member	
Brief Description	If the circle member is in panic condition then there is an button called panic alarm. By pressing it ,push notification will send to every group member that member is in panic..	
Pre-Conditions	Send panic notification to other group members	
Post-Conditions	Admin must provide circle name and image in order to create new one.	
Normal Flow of Events:	**Actor Action**	**System Response**
	Press Alarm button	all-members get notification, also member location will be marked in the form of red circle and display it on map

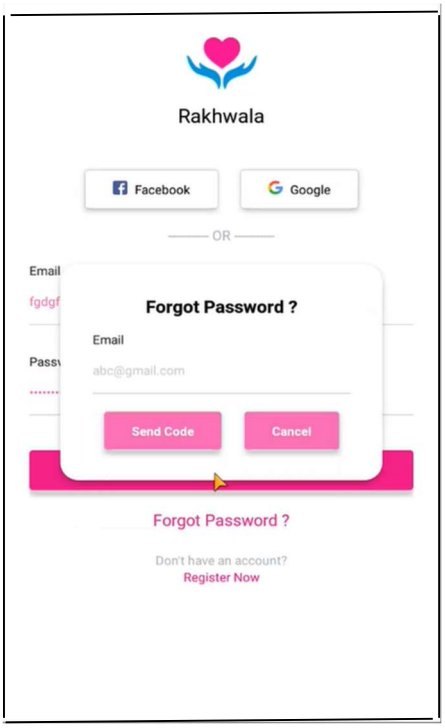
Entity Relationship Diagram Initial

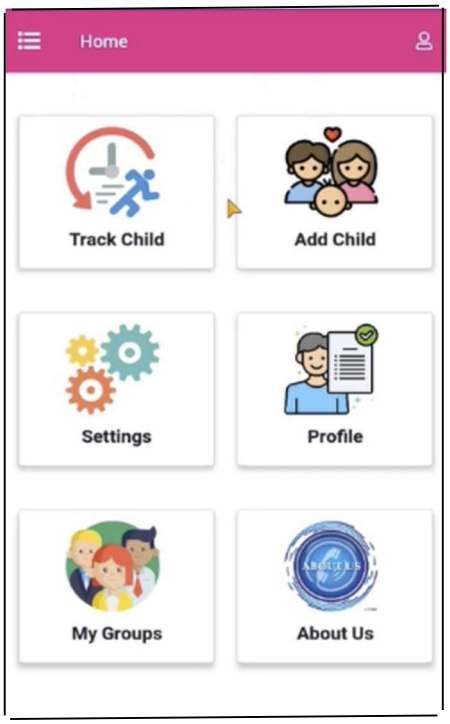


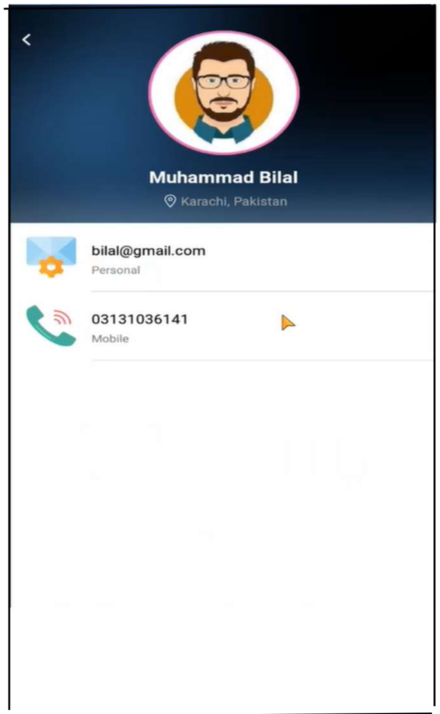
Chapter 5: PROJECT PLANNING & INITIAL DESIGN

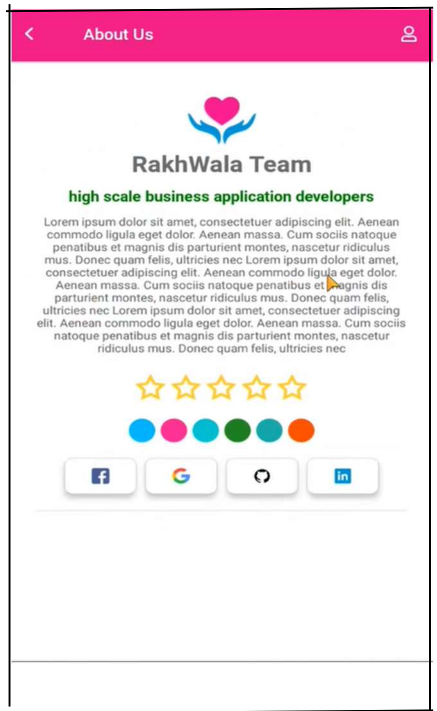









Muhammad Bilal
Karachi, Pakistan
bilal@gmail.com
Personal
03131036141
Mobile


About Us
RakhWala Team
high scale business application developers
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec