**FEST Research Group**

**Final Year Project Weekly Report Log**

*(To be completed by Student)*

|  |  |  |
| --- | --- | --- |
| **S.No** | **Student Name:** | **Student ID:** |
|  | Muhammad Umar | 36682 |
|  | Muhammad Bilal Qamar | 35985 |
|  | Adeel Arshsd | 36498 |
|  |  |  |

**Project Title: Rakhwala (child security system)**

Dr Mansoor Ebrahim

**Supervisor:**

**Report Week: Date:**

**Planned Work**

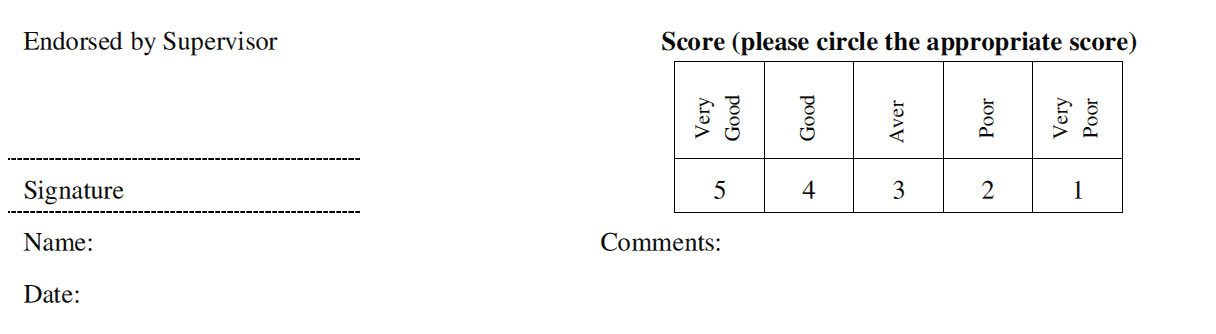
**Achievements**

Successfully develop remaining front-end screens.

Develop remaining front-end screens

**FEST Research Group**

**Proposed work**



Proposal for setup temperature and pulse rate sensors

**Chapter 2 ;**

**Chapter Title :** Technology Background

**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](https://www.webopedia.com/TERM/D/desktop.html) and [laptop computers](https://www.webopedia.com/TERM/L/laptop_computer.html), [smartphones](https://www.webopedia.com/TERM/S/smartphone.html) and [tablets](https://www.webopedia.com/TERM/T/tablet_PC.html) 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](https://www.politico.com/agenda/story/2015/06/internet-of-things-growth-challenges-000098) (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.

Ex :

[A lightbulb](https://www.zdnet.com/article/building-my-own-internet-of-things-ambient-experience-one-step-at-a-time/) that can be switched on using a smartphone app is an IoT device, as is a motion sensor or a [smart thermostat](https://www.zdnet.com/article/johnson-controls-cortana-powered-thermostat-is-up-for-preorder-in-march/) in your office or a connected streetlight. An IoT device could be as fluffy as [a child's toy](https://www.zdnet.com/article/fbi-to-parents-beware-your-kids-smart-toy-could-be-a-security-risk/) or as serious as [a driverless truck](https://www.zdnet.com/article/driverless-trucks-are-coming-but-for-now-adoption-is-in-the-slow-lane/). 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](https://www.zdnet.com/article/las-vegas-announces-smart-city-plans-with-cisco/) 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](http://twitter.com/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.

**REACT NATIVE**

React Native is a JavaScript framework for mobile apps based on the React JS platform. It’s used in building popular apps such as Pinterest, Tesla, and Bloomberg News. Some of its features include ease of use and code reusability. In this article, let’s take a look at some of the pros and cons to decide if this framework is right for your work and your app development.

After React JS, [React Native](https://facebook.github.io/react-native/) is here. The robust JavaScript framework is used for creating native mobile applications for iOS and Android.

Based on the React JS platform, the only difference is that it focuses on building mobile applications rather than user interfaces for the browser. To put things into perspective, web developers can now focus on writing code that is “native” by every definition of the word. Both Android and iOS developers can feel comfortable writing code with React Native.

A genuinely diverse framework, React Native combines many different languages. On the surface, it is written in a mix of JavaScript and JSX, a JS version of XML. Below, you have the React Native “bridge” that combines Objective C and Java for iOS and Android, respectively. Your application, therefore, renders like a real application rather than a browser app. Filled with APIs, the React Native platform provides you multiple options to integrate features with your application.

**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.

**Summary :**

All in all, we believe that React Native is a framework that allows web developers the convenience of building efficient and robust mobile applications with just their existing JavaScript knowledge.

**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.

**OR**

Node.js is a server-side environment that allows [Node developer](https://www.upwork.com/hire/nodejs-developers/)s to build servers and network applications with JavaScript for the first time. This means entire sites can be run on a unified JavaScript stack—both the client-side software, and the server-side software.Technically, it’s a development platform, not a framework—with [numerous frameworks that run on top of it](https://www.upwork.com/hiring/development/15-node-js-frameworks-to-know/). It’s also a runtime environment, a component of a development platform that allows developers to test a software program while it’s running—simulating how the program will behave once it’s been executed, so any bugs or errors can be tracked and fixed. Node.js runs a programming construct called an “event loop,” which waits for client requests then sends them to the server or database. It’s best to think of Node.js as less of a full-fledged server, and more of a foundation upon which you can easily add libraries and workhorse connectors like HTTP, SSL, and TCP to quickly set up a working dynamic web server—all with only a few lines of JavaScript.

**Pros of Node JS** :

Using Node.js for backend .

* code sharing and reuse
* speed and performance
* easy knowledge sharing within a team
* a huge number of free tools

**STRUCTURAL QUERY LANGUAGE (SQL)**

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.

**Problems**

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. No Health monitoring is provided by the current market solution U-Watch it just traces the location of the child .

U watch 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 .

**Historical Background :**

Ufone’s UWatch is primarily aimed at keeping the kids connected with parents when they are away, and that too without the need of smartphones.With the UWatch app, the parent is always in control. From live monitoring to setting up safe zones, the app provides several ways to customize the UWatch experience for your family.

**Main Features of UWatch**

* Real time tracking
* Voice calling (Parent’s to kids)
* Fast Dial Emergency SOS Calling (Kids to Parents)
* Safe Zones / Geo Fence
* Safe list calling
* Tracking History Playback
* Remote Turn off/Turn On Function
* Battery time of up to 2 days
* Water Resistant
* Time, Date and Alarm.

**References**

Qian, Z., Jiang, W. and Tsui, K. -L. (2006) ‘Churn detection via customer profile modeling’, International Journal of Production Research, Vol. 44, No. 14, pp. 2913–2933.

Aydin, S. and Özer, G. (2006) ‘How switching costs affect subscriber loyalty in the Turkish mobile phone market: An exploratory study’, Journal of Targeting, Measurement and Analysis for Marketing, Vol. 14, No. 2, pp. 141–155.

Ultsch, A. (2002) ‘Emergent self-organising feature maps used for prediction and prevention of churn in mobile phone markets’, Journal of Targeting, Measurement and Analysis for Marketing, Vol. 10, No. 4, pp. 314–324.

**Chapter 3 ;**

**Chapter Title :** Requirement Methodology

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) .

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](http://www.rakhwala-herokuapp.com/)).

**Scope of project :**

The scope 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.

**Our solution will address the following weaknesses of above mentioned solutions.**

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, Manually press button in emergency situation we are eliminating this due to children of all ages small children will misuse the button.

**We will use the following techniques to achieve improvements mentioned above.**

**Front-End :-**

* React-Native(CLI/Expo)

**Back-End:-**

* NodeJs/Python

**Databases:-**

* SQL

**Apis:-**

* Google Maps api
* Directions api
* Search Places api

**IOT :-**

* Arduino Interface
* Arduino boards

**Sensors:-**

* Temperature
* Pulse rate

**Expectd Outcomes**

By this application and IOT based watch, Parent can monitor their children and another special feature called family circle where all family members are present and check the location of other family members. And if any of the member is in panic condition there is a button called send panic alarm by pressing it push notification is sent to all members which are active, logged in and part of the respective group

**Key Benefits**

Parents can trace current location and health status of their child every second.

Family members can trace current location and health of their loved ones every moment.

Childs are safe from incidents such as kidnapping, lost and many other incidents.