IoT - Based Safety Gadget for Child Safety Monitoring and Notification

# A PROJECT REPORT

***Submitted by***

|  |  |
| --- | --- |
| **CHARULATHA N**  **(TEAM LEADER)** | **[710719106017]** |
| **APARNA BHUVANESVARI L**  **(TEAM MEMBER)** | **[710719106006]** |
| **ABINEETHA M** | **[710719106004]** |
| **(TEAM MEMBER)**  **ASWINI G**  **(TEAM MEMBER)**   |  |  | | --- | --- | |  | **[710719106013]** |   **TEAM ID : PNT2022TMID31414** | **[710719106013]** |

# 

**ABSTRACT**

This paper is mainly streamed towards child safety solutions by developing a gadget which can be tracked via its GPS locations and also a panic button on gadget is provided to alert the parent via GSM module calling for help. Parental android app is developed to manage and track the device anytime. Smart gadget device is always connected to parental phone which can receive and make phone calls and also receive SMS on gadget via GSM module, also a wireless technology is implemented on device which is useful to bound the device within a region of monitoring range, if device is moving out of monitoring range then an alert will be triggered on binding gadget, This helps you keep a virtual eye on child. Health monitoring system on gadget checking for parameters like heart beat/pulse rate and temperature is included which can be monitored on parental app.

Gadget also monitors whether it is plugged on hand or not using contact switch and alert the parent as soon as it is unplugged.



# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER** | **TITLE** | **PAGE NO** |
|  | **ABSTRACT** | **i** |
|  | **LIST OF FIGURES** | **ii** |
|  | **LIST OF TABLES** | **iii** |
|  | **LIST OF ABBREVIATIONS** | **iv** |
| **1.** | **INTRODUCTION** | **1** |
|  | 1.1 PROJECT OVERVIEW | 1 |
|  | 1.2 PURPOSE | 1 |
| **2.** | **LITERATURE SURVEY** | **2** |
|  | 2.1 EXISTING PROBLEM | 2 |
|  | 2.2 REFERENCES | 3 |
|  | 2.3 PROBLEM STATEMENT DEFINITION | 3 |
| **3.** | **IDEATION & PROPOSED SOLUTION** | **4** |
|  | 3.1 EMPATHY MAP CANVAS | 4 |
|  | 3.2 IDEATION & BRAINSTORMING | 5 |
|  | 3.3 PROPOSED SOLUTION | 7 |
|  | 3.4 PROBLEM SOLUTION FIT | 8 |
| **4.** | **REQUIREMENT ANALYSIS** | **9** |
|  | 4.1 FUNCTIONAL REQUIREMENTS | 9 |
|  | 4.2 NON-FUNCTIONAL REQUIREMENTS | 9 |
| **5.** | **PROJECT DESIGN** | **10** |
|  | 5.1 DATA FLOW DIAGRAM | 11 |
|  | 5.2 SOLUTION & TECHNICAL  ARCHITECTURE | 11 |
|  | 5.3 USER STORIES | 13 |
| **6.** | **PROJECT PLANNING & SCHEDULING** | **14** |
|  | 6.1 SPRINT PLANNING & ESTIMATION | 14 |
|  | 6.2 SPRINT DELIVERY SCHEDULE | 15 |

|  |  |  |
| --- | --- | --- |
|  | 6.3 REPORTS FROM JIRA | 16 |
| **7.** | **CODING & SOLUTIONING** | **17** |
|  | 7.1 FEATURE 1 | 17 |
|  | 7.2 FEATURE 2 | 19 |
|  | 7.3 DATABASE SCHEMA | 22 |
| **8.** | **TESTING** | **23** |
|  | 8.1 TEST CASES | 23 |
|  | 8.2 USER ACCEPTANCE TESTING | 24 |
| **9.** | **RESULTS** | **25** |
|  | 9.1 PERFORMANCE METRICS | 25 |
| **10.** | **ADVANTAGES & DISADVANTAGES** | **29** |
| **11.** | **CONCLUSION** | **30** |
| **12.** | **FUTURE SCOPE** | **31** |
| **13.** | **APPENDIX** | **32** |
|  | SOURCE CODE | 32 |
|  | GITHUB & PROJECT DEMO LINK | 32 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURE NO** | **TITLE** | **PAGE NO.** |
| 3.1.1 | EMPATHY MAP | 4 |
| 3.2.1 | IDEATION & BRAINSTORMING | 5 |
| 3.2.2 | IDEATION PRIORITIZATION | 6 |
| 3.4.1 | PROBLEM SOLUTION FIT | 8 |
| 5.1.1 | DATA FLOW DIAGRAM | 10 |
| 5.2.1 | SOLUTION ARCHITECTURE | 10 |
| 5.2.2 | TECHNICAL ARCHITECTURE | 11 |
| 5.3.1 | USER STORIES | 13 |
| 6.3.1 | ROADMAP | 16 |
| 6.3.2 | BACKLOG | 16 |
| 8.1.1 | TEST CASES | 23 |
| 9.1.1 | USER REGISTRATION | 25 |
| 9.1.1 | VERIFICATION MAIL | 25 |
| 9.2.1 | USER LOGIN | 26 |
| 9.2.2 | USER DETAILS | 26 |
| 9.3.1 | ADDING GEOFENCE | 27 |
| 9.3.2 | ALERT NOTIFACTION | 28 |



# LIST OF TABLES

|  |  |  |
| --- | --- | --- |
| **TABLE NO.** | **TITLE** | **PAGE NO.** |
| 3.3.1 | PROPOSED SOLUTION | 4 |
| 4.1.1 | FUNCTIONAL REQUIREMENTS | 9 |
| 4.2.1 | NON-FUNCTIONAL REQUIREMENTS | 9 |
| 5.2.3 | COMPONENETS & TECHNOLOGIES | 11 |
| 5.2.4 | APPLICATION CHARACTERISITICS | 12 |
| 6.1.1 | SPRINT PLANNING & ESTIMATION | 14 |
| 6.2.1 | SPRINT DELIVERY SCHEDULING | 15 |
| 8.2.1 | DEFECT ANALYSIS | 24 |
| 8.2.2 | TEST CASE ANALYSIS | 24 |



**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| IoT | : INTERNET OF THINGS |
| GPS | : GLOBAL POSITIONING SYSTEM |
| FR | : FUNCTIONAL REQUIREMENTS |
| NFR | : NON-FUNCTIONAL REQUIREMENTS |
| DFD | : DATA FLOW DIAGRAM |
| SQL | : STRUCTURED QUERY LANGUAGES |
| STT | : SECURITY TRANSACTIONS TAX |
| DB | : DATABASE |
| UAT | : USER ACCEPTANCE TESTING |
| WIFI | : WIRELESS FIDELITY |

# CHAPTER 1 INTRODUCTION

* 1. **Project Overview**

A tracker that helps parents track a child's location so that the child does not get intodangerous situations.

The inspiration for this wearable comes mainly from the ever-increasing need of safety for small children in present times because there may be a chances of child lost in the major crowded areas.

This main script mainly focuses on the key features of missing child can be helped by the individuals present around the child and plays an important role in the child's safety until reunite the parent to that location.

# Purpose

Now a day’s Parents have more responsibility than older about their children's. Because Crimes rates are increasing day by day in our country, Crimes such as Child Amusement, Rapes, Murders, Illegal Relationship to avoid these kinds of crimes parents must watch their children every step. Eventually mobile phones cause major allegations on our society. Many teens must be noticed by their own parents; it is our duty. But sometimes children are arguing with their parents for watching their steps, to overcome these issues, we need to watch them through online.

# CHAPTER 2 LITERATURE SURVEY

* 1. **Existing Problem**

Authors: M Nandini Priyanka, S Murugan, K. N. H. Srinivas, T. D. S. Sarveswararao, E. Kusuma Kumari. Title: Smart IoT Device for Child Safety and Tracking. Published in: 2019 IEEE. The system is developed using Link-It ONE board programmed in embedded C and interfaced with temperature, heartbeat, touch sensors and also GPS, GSM & digital camera modules. The novelty of the work is that the system automatically alerts the parent/caretaker by sending SMS, whenimmediate attention is required for the child during emergency.

Merits: The parameters such as touch, temperature & heartbeat of the child are used for parametric analysis and results are plotted for the same.

**Demerits:** To implement the IoT device this ensures the complete solution for child safety problems.

Authors: Akash Moodbidri, Hamid Shahnasser Title: Child safety wearable device. Published in: 2017 IEEE. The purpose of this device is to help the parents to locate their children with ease. At the moment there are many wearables’ in the market which helps to track the daily activity of children and also helps to find the child using Wi-Fi and Bluetoothservices present on the device.

Merits: This wearable over other wearable is that it can be used in any phone and it is not necessary that an expensive smartphone is required and doesn’t want to be very tech savvyindividual to operate.

**Demerits**: As, this device’s battery gives short life-time.

Authors: Aditi Gupta, Vibhor Harit. Published in: 2016 IEEE. Title: Child Safety & Tracking Management System by using GPS. This paper proposed a model for child safety through smart phones that provides theoption to track the location of their children as well as in case ofemergency children is able to send a quick message and its current location via Short Message services.

Authors: Dheeraj Sunehera, Pottabhatini Laxmi Priya. Title: Children Location Monitoring on Google Maps Using GPS and GSM. Published in: 2016 IEEE. This paper provides an Android based solution for the parents to track their children in real time. Different devices relate to a single device through channels of internet. The concerned device is connected to server via internet. The device can be used by parents to track their children in real time or for women safety. The proposed solution takes the location services provided by GSM module. It allows the parents to get their child’s current-location via SMS. Merits: A child tracking system using android terminal and hoc networks.

**Demerits:** This device cannot be used in rural areas.

# REFERENCES

M Nandini Priyanka, S Murugan, K. N. H. Srinivas, T. D. S. Sarveswararao, E. Kusuma Kumari, ‘Smart IoT Device for Child Safety and Tracking’ International Journal of Innovative Technology and Exploring Engineering, Volume 8, Issue 8, June 2019. Akash Moodbidri, Hamid Shahnasser (Jan. 2017) ‘Child safety wearable device’, International Journal for Research in Applied Science & Engineering Technology, Vol. 6 Issue 2, pp. 438-444.Aditi Gupta, Vibhor Harit, ‘Child Safety & Tracking Management System by using GPS, GeoFencing & Android Application: An Analysis,’ 2016 Second International Conference on Computational Intelligence & Communication Technology.

# PROBLEM STATEMENT DEFINITION

Child tracker helps the parents in continuously monitoring the child's location. They can simply leave their children in school or parks and create a geofence around the particular location. By continuously checking the child's location notifications will be generated if the child crosses the geofence. Notifications will be sent according to the child's location to their parents or caretakers.



# CHAPTER 3

**IDEATION & PROPOSED SOLUTION**

# 3.1 EMPATHY MAP CANVAS

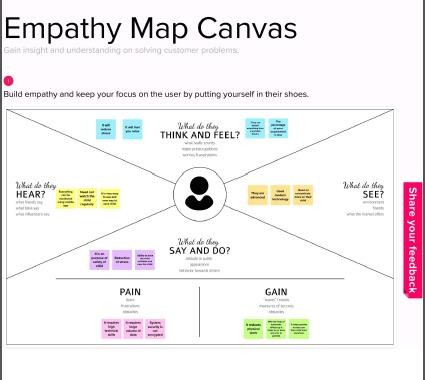


Fig 3.1.1 Empathy Map

This project was created to help parents keep track of their children's whereabouts. Children are more readily influenced by their peers these days, and they may be duped or abducted by strangers. This method may be developed to track a child's current position. After a specific period, the Web application on the device will update the location of the kid to the application. By pushing the distinct button that has been introduced, parents may even take action if their kid is unstable or in an inappropriate area.

WFPS, a WIFI positioning system that doesn't connect to the internet but connects to Wi-Fi access points, will be used to track the child's whereabouts



# IDEATION PRIORITIZATION

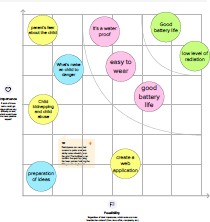
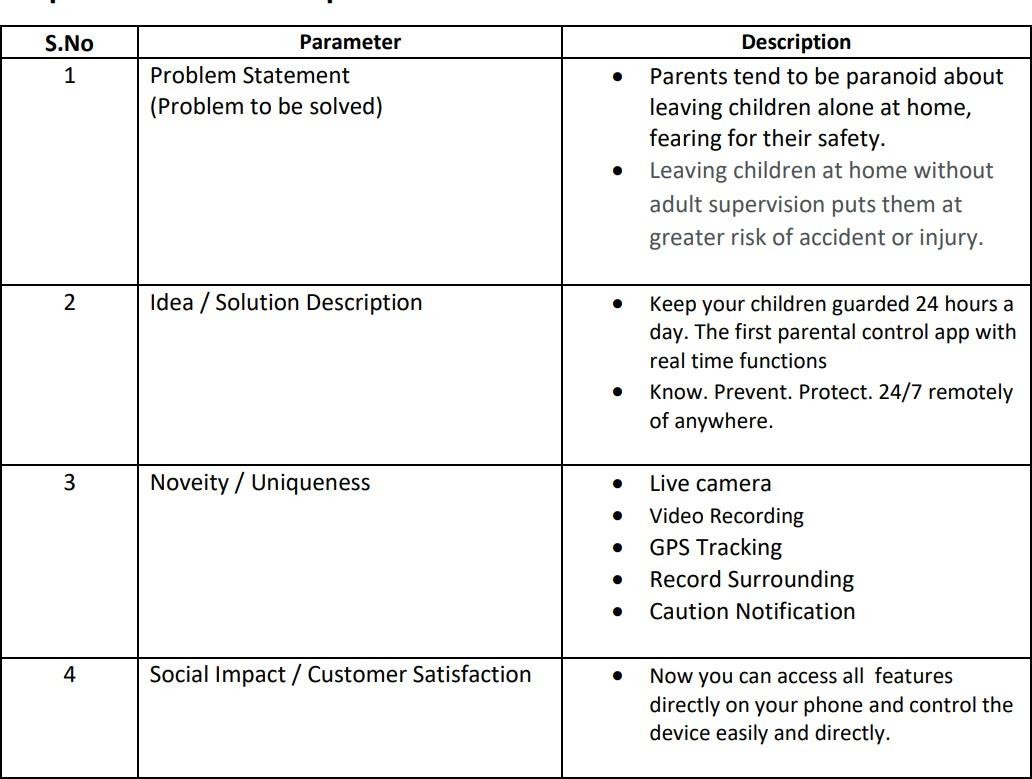


Fig 3.2.1 Ideation Prioritization

# PROPOSED SOLUTION



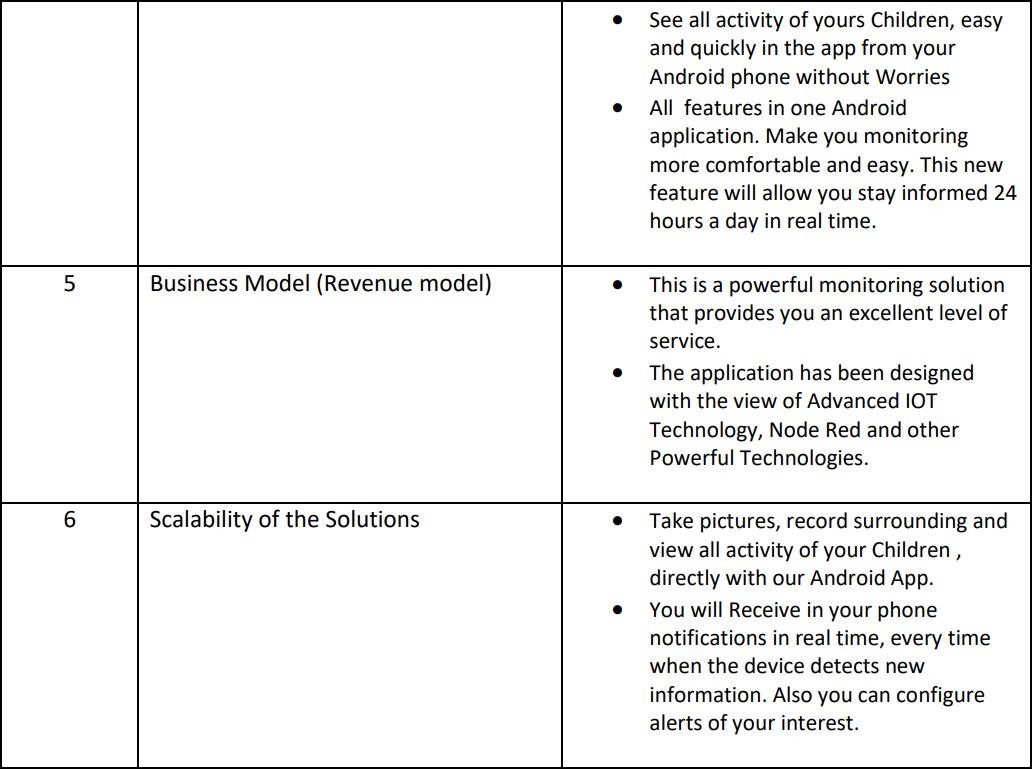


Table 3.3.1 Proposed Solution

# PROBLEM SOLUTION FIT

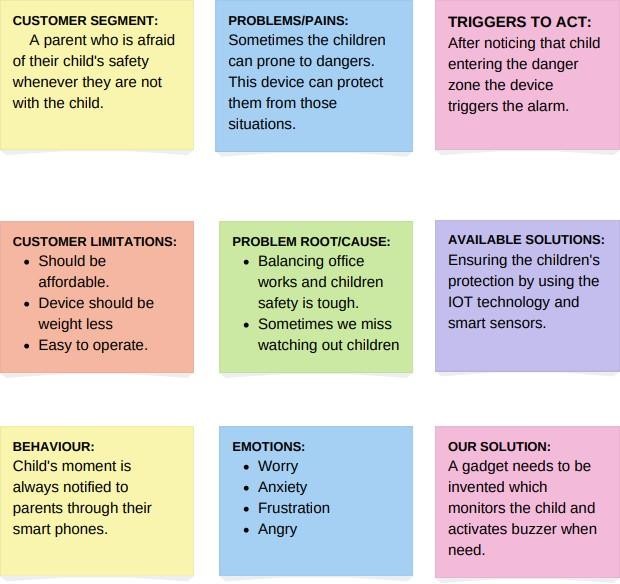


Fig 3.4.1 Problem Solution Fit

# CHAPTER 4 REQUIREMENT ANALYSIS

* 1. **FUNCTIONAL REQUIREMENT**

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional**  **Requirement(Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Registration | Registration through FormRegistration through Gmail |
| FR-2 | User Confirmation | Confirmation via Email  Confirmation via OTP |
| FR-3 | User Notification | Notification send to Mobile Number  Notification send through message/ call |
| FR-4 | User Location Check | Check through Account |

Table 4.1.1 Functional Requirements

# NON-FUNCTIONAL REQUIREMENTS

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional**  **Requirement** | **Description** |
| NFR-1 | **Usability** | Allow parents to track their child’s location and also monitor them. |
| NFR-2 | **Security** | Creates a secure environment for children to monitor around |
| NFR-3 | **Reliability** | Increased reliability towards  technology |
| NFR-4 | **Performance** | High performance in terms of simple usage and  security |
| NFR-5 | **Availability** | Backup power supply |
| NFR-6 | **Scalability** | Increase in scalability |

Table 4.2.1 Non-Functional Requirements



# CHAPTER 5 PROJECT DESIGN

* 1. **DATA FLOW DIAGRAMS**

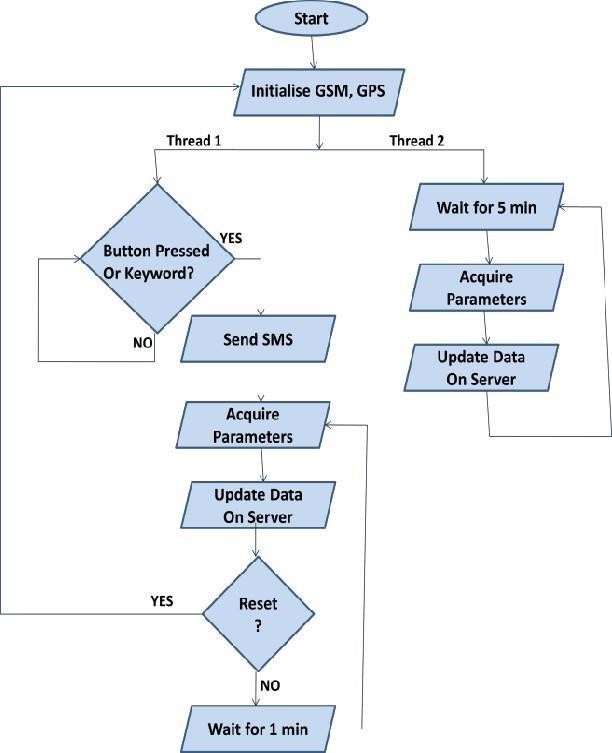


Fig 5.1.1 Data Flow Diagrams

# Solution & Technical Architecture

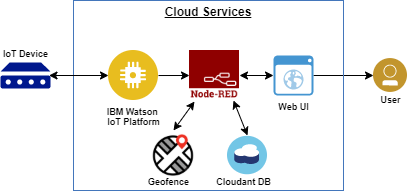


Fig 5.2.2 Technical Architecture

# USER STORIES

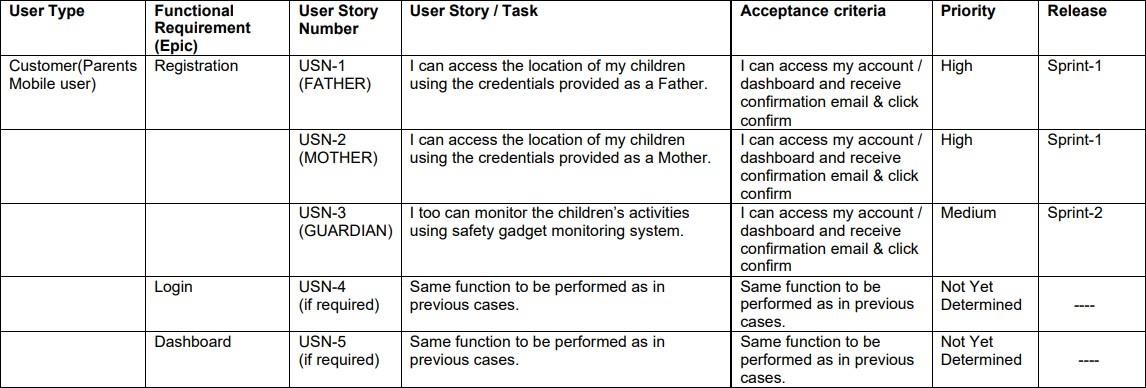
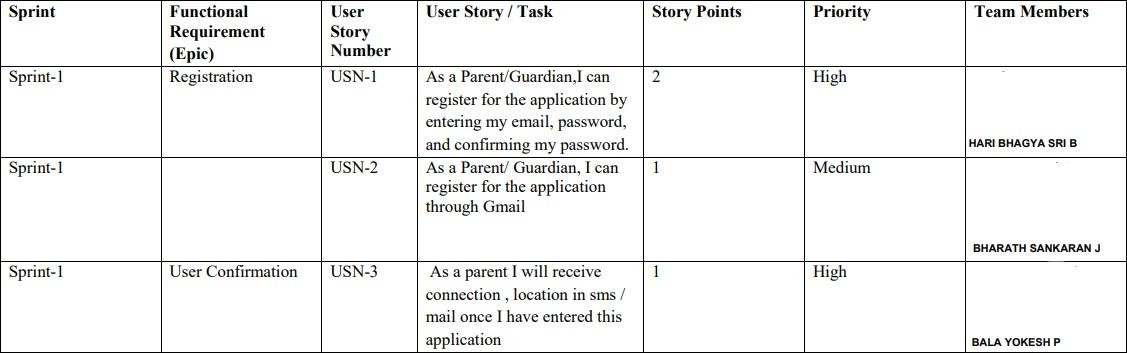


Fig 5.3.1 User Stories

# CHAPTER 6

**PROJECT PLANNING & SCHEDULING**

# SPRINT PLANNING & ESTIMATION



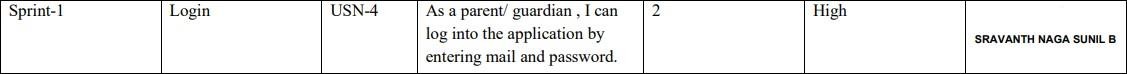


Table 6.1.1 Sprint Planning & Estimation

# SPRINT DELIVERY SCHEDULE

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sprint** | **Functional**  **Requirement (Epic)** | **User Story Number** | **User Story / Task** | **Story Points** | **Priority** |
| Customer (Mobile user) | Registration | USN-1 | As a user, I can register for the application by entering my email,  password, and confirming my password. | 20 | High |
|  |  | USN-2 | As a user, I will receive  on confirmation emailfor ce I have registered  the application | 20 | High |
|  |  | USN-3 | As a user, I can register  for the application through Facebook | 5 | Low |
|  |  | USN-4 | As a user, I can register for the application  through Gmail | 10 | Medium |
|  | Login | USN-5 | As a user, I can log into the application by entering email & password | 20 | High |
| **Dashboard** | | | | | |
| Customer (Web user) | Login |  | When I enter I can view the working of  applications, scan and monitor the operationsand check if all the users are  authorized | 10 | Medium |
| Customer Care Executiv e | Login |  | Maintaining and accessing the database containing the locations are secure  accu andand update  rate  constantly | 20 | High |
| Administra tor | Login |  | As a user I can register for the application by entering my correct credentials | 20 | High |

Table 6.2.1 Sprint Planning & Estimation

# REPORTS FROM JIRA ROADMAP

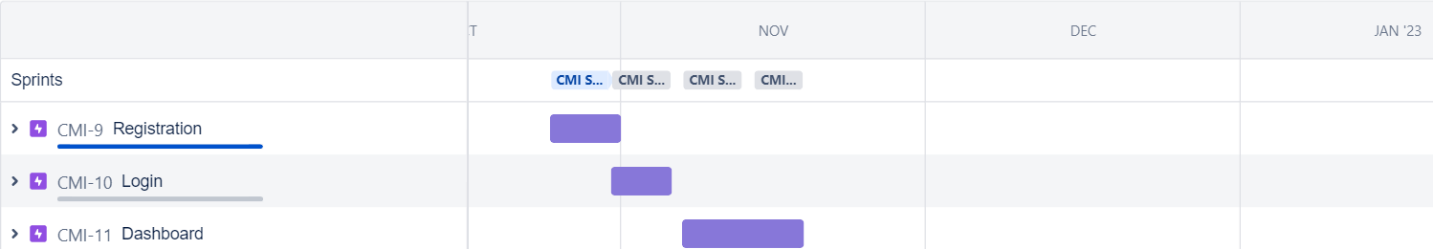


Fig 6.3.1 Road Map

# BACKLOG

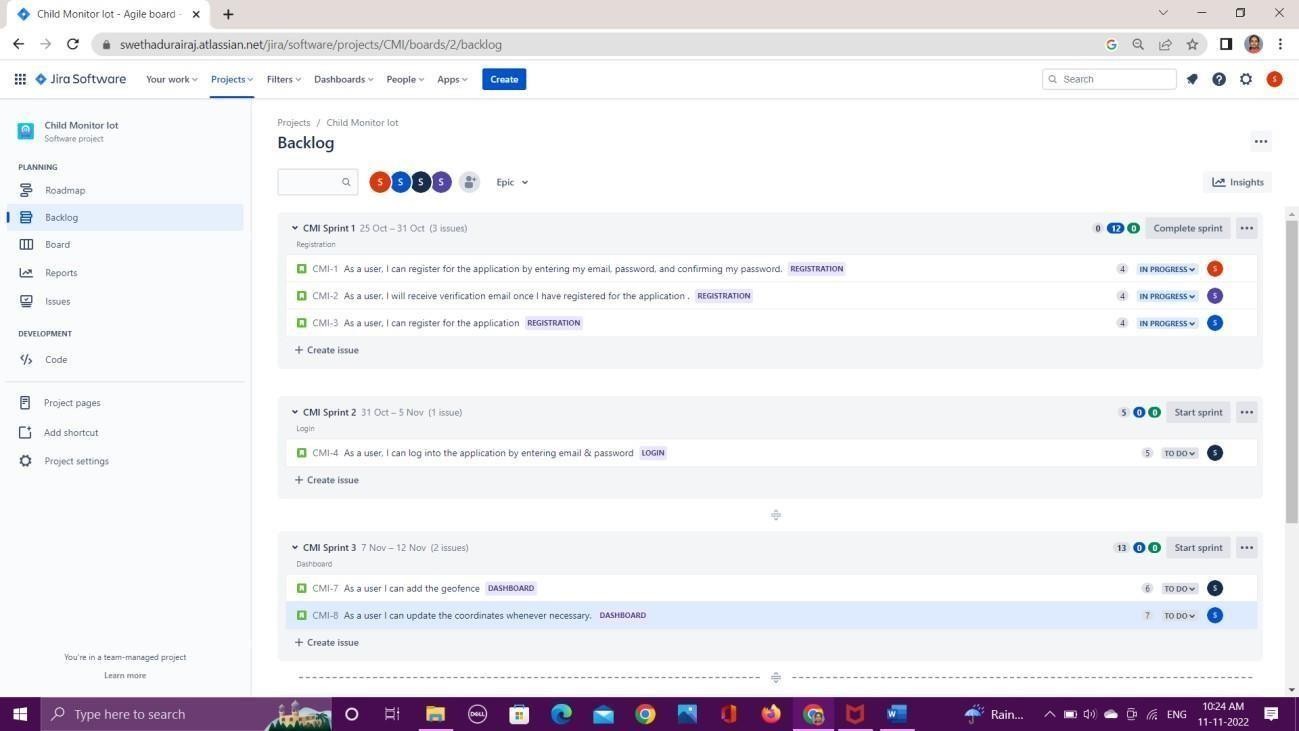


Fig 6.3.2 Backlog

# CHAPTER 7

**CODING & SOLUTIONING**

# FEATURE 1 (ADDING GEOFENCE)

Geofence is like a round wall covering the given location. So parents can use them to mark the locationwhere their children are going.

package com.example.geofence; import android.app.PendingIntent; import android.content.Context;

import android.content.ContextWrapper;

import android.content.Intent;import android.widget.Toast; import com.google.android.gms.common.api.ApiException; import com.google.android.gms.location.Geofence;

import com.google.android.gms.location.GeofenceStatusCodes; import com.google.android.gms.location.GeofencingRequest; import com.google.android.gms.maps.model.LatLng;

public class GeofenceHelper extends ContextWrapper {

private static ﬁnal String *TAG* = "GeofenceHelper";PendingIntent pendingIntent;

public GeofenceHelper(Context base) { super(base);

}

public GeofencingRequest getGeofencingRequest(Geofence geofence) {return new GeofencingRequest.Builder()

.addGeofence(geofence)

.setInitialTrigger(GeofencingRequest.*INITIAL\_TRIGGER\_ENTER*)

.build();

}

public Geofence getGeofence(String ID, LatLng latLng, ﬂoat radius,int transitionTypes) {

return new Geofence.Builder()

.setCircularRegion(latLng.latitude, latLng.longitude,

radius)

.setRequestId(ID)

.setTransitionTypes(transitionTypes)

.setLoiteringDelay(5000)

.setExpirationDuration(Geofence.*NEVER\_EXPIRE*)

.build();

}

public PendingIntent getPendingIntent() { if (pendingIntent != null) {

return pendingIntent;

}

Intent intent = new Intent(this, GeofenceBroadcastReceiver.class);

pendingIntent = PendingIntent.getBroadcast(this, 2607, intent, PendingIntent.FLAG\_IMMUTABLE);

return pendingIntent;

}

public String getErrorString(Exception e) { if (e instanceof ApiException) {

ApiException apiException = (ApiException) e; switch (apiException.getStatusCode()) {

case GeofenceStatusCodes

. GEOFENCE\_NOT\_AVAILABLE: return "GEOFENCE\_NOT\_AVAILABLE";

case GeofenceStatusCodes

GEOFENCE\_NOT\_AVAILABLE:

return "GEOFENCE\_NOT\_AVAILABLE";

case GeofenceStatusCodes

.GEOFENCE\_TOO\_MANY\_GEOFENCES: return "GEOFENCE\_TOO\_MANY\_GEOFENCES";

case GeofenceStatusCodes

.GEOFENCE\_TOO\_MANY\_PENDING\_INTENTS: return "GEOFENCE\_TOO\_MANY\_PENDING\_INTENTS";}}

# FEATURE 2 (ALERT NOTIFICATION)

Once geofence is added, when the child enters the geofence a notification will be sent

When the child leaves the geofence a notification will be sent.

package com.example.geofence;

import android.content.BroadcastReceiver;import android.content.Context;

import android.content.Intent; import android.location.Location;import android.os.CountDownTimer;import android.util.Log;

import android.widget.Toast;

import com.google.android.gms.location.Geofence; import com.google.android.gms.location.GeofencingEvent import java.util.List;

import android.os.Handler;

public class GeofenceBroadcastReceiver extends BroadcastReceiver {

private static ﬁnal String *TAG* = "GeofenceBroadcastReceiv";

@Override

public void onReceive(Context context, Intent intent) {

// *TODO: This method is called when the*

*BroadcastReceiver* is receiving

// an Intent broadcast

//.

/\*Toast.makeText(context, "GEOFENCE\_ENTERED", Toast.LENGTH\_SHORT).show();

ﬁnal Toast mToastToShow;

int toastDurationInMilliSeconds = 1200000;

mToastToShow = Toast.makeText(context, "GEOFENCE\_EXITED", Toast.LENGTH\_LONG);

// Set the countdown to display the toast CountDownTimer toastCountDown; toastCountDown = new

CountDownTimer(toastDurationInMilliSeconds, 100000) { public void onTick(long millisUntilFinished) {

mToastToShow.show();

}

public void onFinish() { mToastToShow.cancel();

}

};

// Show the toast and starts the countdown mToastToShow.show(); toastCountDown.start();\*/

NotiﬁcationHelper notiﬁcationHelper = new NotiﬁcationHelper(context); notificationHelper.sendHighPriorityNotification("GEOFENCE\_TRANSITION\_EN TER",

"", MapsActivity.class);

GeofencingEvent geofencingEvent = GeofencingEvent.fromIntent(intent); If (geofencingEvent.hasError())

Log.d(TAG, "onReceive: Error receiving geofence event..."); return;

}

List<Geofence> geofenceList = geofencingEvent.getTriggeringGeofences();

for (Geofence geofence: geofenceList) {

Log.d(TAG, "onReceive: " + geofence.getRequestId());

}

// Location location = geofencingEvent.getTriggeringLocation(); int transitionType = geofencingEvent.getGeofenceTransition();

switch (transitionType) {

case Geofence.GEOFENCE\_TRANSITION\_ENTER:

notificationHelper.sendHighPriorityNotification("Entered the Location", "", MapsActivity.class);

break;

case Geofence.GEOFENCE\_TRANSITION\_EXIT: notificationHelper.sendHighPriorityNotification("Exited the Location ", "", MapsActivity.class);

break;} } }

# DATABASE SCHEMA

We assume that only one child can leave the set maximum distance at a time. The beacons take 20 seconds to update the previous location data, hence we assume the notification trigger has a 20-40 seconds’ lag in updating the right location. We assume that Wi-Fi is readily available since the backend server is located in the cloud and then touse the mobile devices’ location services.

|  |  |  |  |
| --- | --- | --- | --- |
| Moto Xplay Mobile Device | 1 | Communication Hardware | OS:Android Qualcomn Snapdragon 615 Octa-core Memory:32GB |
| Google Asus Table | 1 | Communication Hardware | OS:Android Quad-core 1.2GHZ  Cortex-A9 Bluetooth 3.0 Memory:1GB |
| Sony Xperia D5803 | 1 | Communication  Hardware | OS:Android Qualcomm MSM8974AC  snapdragon RAM:2GB  Memory:16GB Bluetooth 4.0 |
| Cloud Storage Amazon EC2 | 1 | Communication  Hardware | OS:Ubuntu Memory:1GB RAM:2GB |

# CHAPTER 8

**TESTING**

# TEST CASES

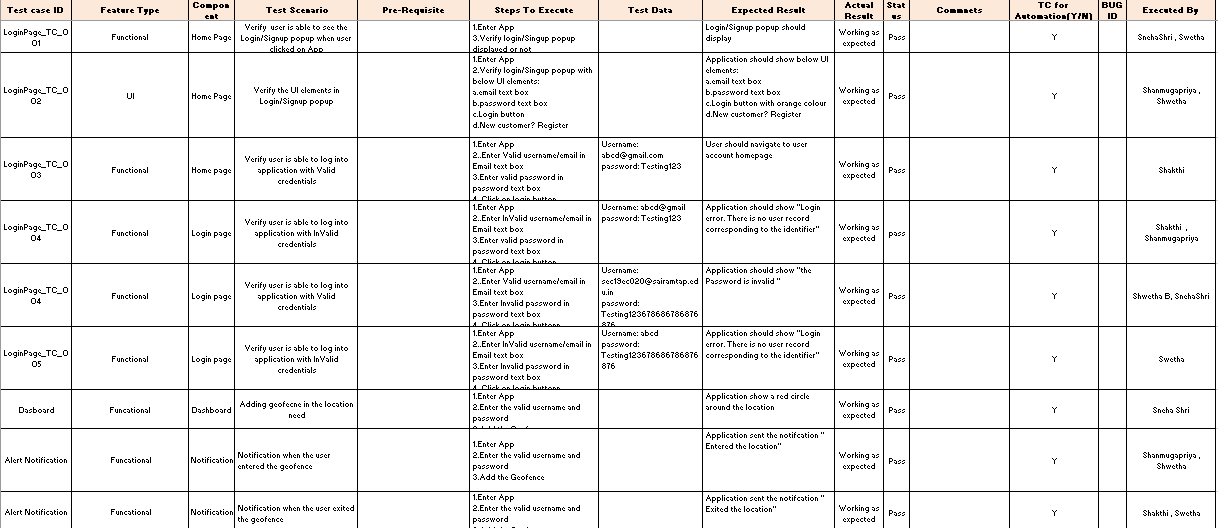


Fig 8.1.1 Test Cases

# USER ACCEPTANCE TESTING

* + 1. **DEFECT ANALYSIS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Resolution** | **Severity 1** | **Severity 2** | **Severity 3** | **Severity 4** | **Subtotal** |
| By Design | 11 | 4 | 2 | 2 | 19 |
| Duplicate | 1 | 1 | 2 | 0 | 4 |
| External | 2 | 3 | 0 | 1 | 6 |
| Fixed | 10 | 2 | 3 | 20 | 35 |
| Not Reproduced | 0 | 0 | 2 | 0 | 2 |
| Skipped | 0 | 0 | 2 | 1 | 3 |
| Won't Fix | 0 | 5 | 2 | 1 | 8 |
| Totals | 24 | 15 | 13 | 25 | 77 |

Table 8.2.1 Defect Analysis

# TEST CASE ANALYSIS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Section** | **TotalCases** | **Not Tested** | **Fail** | **Pass** |
| Print Engine | 5 | 0 | 1 | 4 |
| Client Application | 47 | 0 | 2 | 45 |
| Security | 3 | 0 | 0 | 3 |
| OutsourceShipping | 2 | 0 | 0 | 2 |
| Exception Reporting | 11 | 0 | 2 | 9 |
| Final Report Output | 5 | 0 | 0 | 5 |
| Version Control | 3 | 0 | 1 | 2 |

Table 8.2.2 Test Case Analysis

# CHAPTER 9

**RESULTS**

# PERFORMANCE METRICS

* + 1. **USER REGISTRATION:**

User gets registered to the app using their mail and create their password. On the user is registered a verification mail will be sent to the user mail id. The user needs to verify the account. All user details are stored in the firebase and verification mail is sent by firebaseauthentication.

# Registration Page:

Fig 9.1.1 User Registration

# USER LOGIN

User with their registered mail and password will login to the account. As the details are stored in firebase, when invalid mail or password is entered a message say invalid mail or password occur

# Login page:

Fig 9.2.1 User login

# User Details

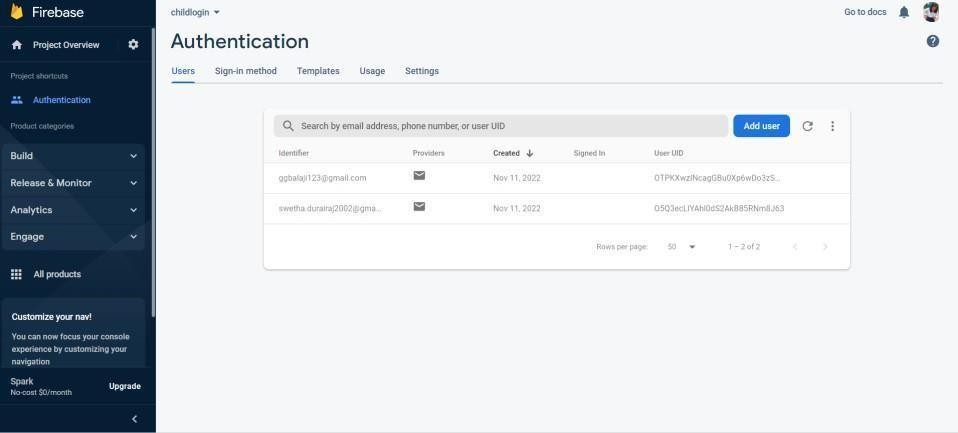


Fig 9.2.2 User Details

# ADDING GEOFENCE AND ALERT NOTIFICATION

User can add geofence in the location where they want to add or where their child is going play so they can monitor the child location. Once the child enters the geofence alert notification says entered the location will be displayed. When the child leaves the geofence alert notification says exited the location will have displayed.

# Geofence

Fig 9.3.1 Adding Geofence

# NOTIFICATION

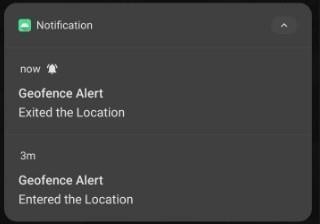


Fig 9.3.2 Alert Notification

# CHAPTER 10 ADVANTAGES & DISADVANTAGES

**ADVANTAGES:**

* Simple and easy to use
* Parents can feel secure because if the child leave the desired location and immediately anotification will be sent.
* Geofence can be added easily.
* Accurate real-time data.
* Efficient use of resources.
* Accountability and Safety.
* Process automation

# DISADVANTAGES:

* Multiple geofence can be a problem.
* Maintenance can be time-consuming.
* Pushback due to privacy concerns.
* Battery and data draining.
* Lack of formal policies.

# CHAPTER 11 CONCLUSION

This research demonstrates Smart IoT device for child safety and tracking, to help the parents to locate and monitor their children. Through this device, the parent can track and monitor their child with just a simple app. It is not possible to always stay beside children asmost of the parents need to go for work. With this project, parents can track the location of their children and get alerts whenever the child out of the geofence. It becomes easy for parents to look after their child while working. This device is efficient to use. Thus, by keeping in mind the advantages and applications we are developing a child monitoring device.In order to avoid kidnapping cases, the child monitoring system is needed.

# CHAPTER 12 FUTURE SCOPE

The future work would be to further develop and implement the safety wearable device so thatit could be watch or sown into a fabric that could be worn, using synthetic fibers. When a violation of child safety is identified, a certain sensor in the child module will emit a signal, which is the main function of the suggested child tracking system. These sensors and WFPS will send this signal to the microcontroller, which will then send it to the transmitter, which will then send it to the parent module. The decision will be made by the parent module, and the violation handling procedure will begin. The kid tracking system's functionality necessitates hardware between the child and parent models, which comprises a drive circuit for the sensors' activation.

# CHAPTER 13 APPENDIX

GitHub link

https://github.com/IBM-EPBL/IBM-Project-39504-1660452006