**MISRIMAL NAVAJEE MUNOTH JAIN ENGINEERING COLLEGE**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**IBM PROJECT REPORT**

**TOPIC -IOT Based Safety Gadget for Child Safety Monitoring & Notification**

**TEAM ID:**PNT2022TMID37374

**TEAM MEMBERS:**

PRATHAM V JAIN(TL)

BHARANI THARAN B

NATHAN V

ROHITH P

**Introduction:**

**Project Overview:**

Parents need not have a smart mobile. Set of keywords are used to gain information from the kit. A portable device which will have a pressure switch. As soon as an assailant is about to attack the person or when the person senses any insecurity from a stranger, he/she can then put pressure on the device by squeezing or compressing it. Instantly the pressure sensor senses this pressure and a conventional SMS, with the victim’s location will be sent to their parents/guardian cell phone numbers stored in the device while purchasing it, followed by a call. If the call is unanswered for a prolonged time, a call will be redirected to the police and the same message will be sent. Additionally, if the person crosses some area which is usually not accessed by the person then a message with the real-time location is sent to the parent/guardian's phone via conventional SMS.

**Project Purpose:**

It makes parents to easily monitor their children in real time just like staying beside them as well as focusing on their own career without any manual intervention.

**Literature Survey:**

**Existing Problem:**

The disadvantage of this project are,

* 1. The child could not produce the exact alert command during a panic condition.
  2. The command produced may not match with the previously stored command.
  3. This project requires manual intervention.

**References:**

**Problem Statement Definition:**

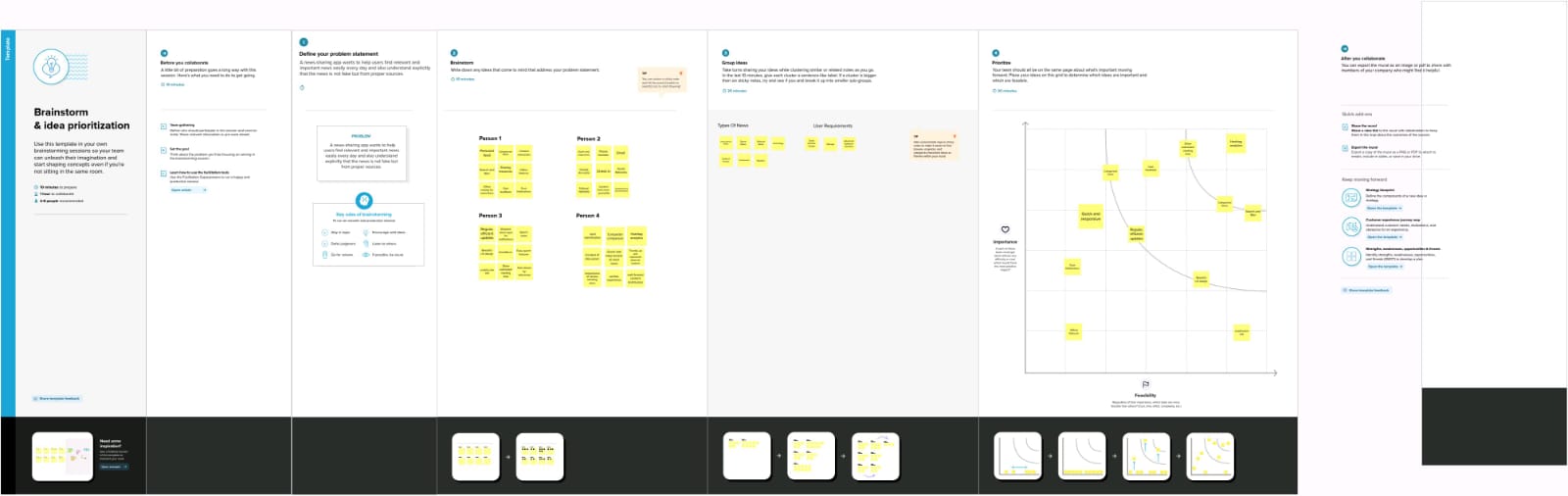
More families are now spending time on work and social duties, hence away from their children. This causes increased concerns towards their safety and whereabouts, and has made keeping a track of their activities quite challenging. Crimes against children are increasing Year on Year. According to a study, roughly 60,000 children go missing in India every year [1]. There is an assumption that every 10 minutes, a child goes missing. Mumbai and Delhi have the highest rate when compared to other metro cities. With the lack of availability of affordable child monitoring systems it is hard to monitor the whereabouts of Children. Safety of children is very critical since children cannot protect themselves. A momentary lack in parental supervision should be combated with an appropriate IT solution in context. Therefore, it is necessary for the proposed system to alert the parents when the child walks too far away and/or outside the “circle of safety” when they are away. In case of an emergency, or in a situation of panic, the child must be able to communicate with their parents. This can be done via live transmission of audio from the device with the child, to the parent’s device.

**Ideation and Proposed Solution:**

**Empathy Map Canvas:**



**Ideation & Brainstorming:**

****

**Proposed solution:**

|  |  |  |
| --- | --- | --- |
| **S.NO.** | **PARAMETER** | **DESCRIPTION** |
| 1. | PROBLEM STATEMENT (PROBLEM TO BE SOLVED) | Parents are often worried about their children when they are out of sight, The aim of this project is to help parents to monitor their children’s location and to see whether their child is safe or not. This system provides a tracking solution for the parent to keep tracking their child's location outdoors by using GPS as it allows them to determine the exact location of the child. |
| 2. | IDEA / SOLUTION DESCRIPTION | It has always been a troublesome process for the parents to look after their children with their busy schedules, so this system sends a notification message to parents and stores the data of the child’s movement and geospace periodically. At times the notification may not hear or be received to the parents, We aim to develop and provide a good interface that would give a tremendous output. The technology used here is PYTHON IDLE and CLOUD for storing data. |
| 3. | NOVELTY / UNIQUENESS | This project is basically for the parents who cannot balance their children and work at the same time and also for nonworking parents. The uniqueness of our project is about geofencing, high noise alert, and location monitoring. |
| 4. | SOCIAL IMPACT / CUSTOMER SATISFACTION | The parents will have the satisfaction that their child is safe and not involved in any critical situation even in their absence. |
| 5. | BUSINESS MODEL (REVENUE MODEL) | The business model is in such a way that everyone can afford it.  It is very cost-efficient.  We are cutting the cost in external components.  There is no need of buying any external components instead they can use their mobile phones to track. |
| 6. | SCALABILITY OF THE SOLUTION | Child safety monitoring is a guardian angel for the parents who can have the exact location of their child which helps to protect the child from any critical situations.  So we resolve the problems like low noise notification, high standard geofencing and since we store data in the cloud it can be retrieved when needed. |

**Problem solution fit:**

**Requirement analysis:**

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Registration | Registration through Form  Registration through Gmail and phone number |
| FR-2 | User Confirmation | Confirmation via Email  Confirmation via OTP |
| FR-3 | Create Profile | Getting Personal Detail with area ,location, and pincode |
| FR-4 | Profile Confirmation | Confirmation via Email  Confirmation via Message |
| FR-5 | Geo-Fencing | Plotting the marks of the location using altitude and longitude |

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | Monitoring and Tracking the child’s Location |
| NFR-2 | **Security** | Sending notification message to the parents and caretakers through messages. |
| NFR-3 | **Reliability** | Provide accurate Location of the child |
| NFR-4 | **Performance** | Updating the child’s location minute by minute |
| NFR-5 | **Availability** | Available 24/7 |
| NFR-6 | **Scalability** | No problem will be faced in the server during increased demand |

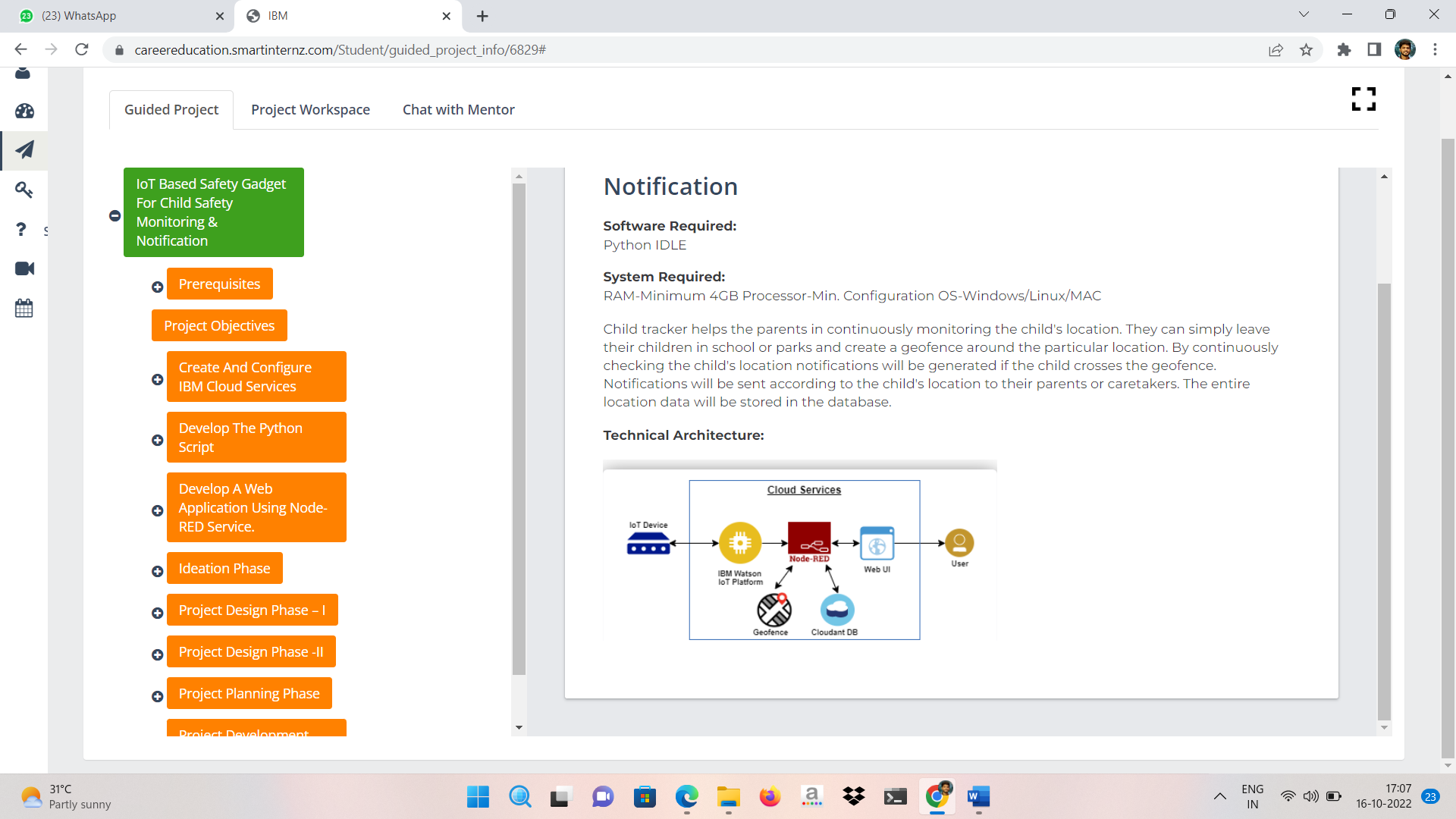
**Project Design phase:**

Data flow diagram:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the rightamount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data isstored.

TECHNOLOGY ARCHITECTURE**:**

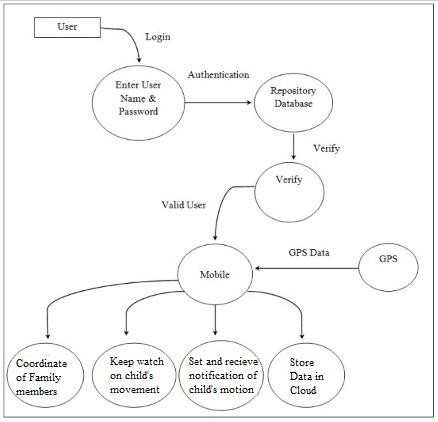
* Technology architecture deals with the deployment of application components on technology components



|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
| 1. | User Interface | Web UI | HTML, CSS |
| 2. | Application Logic-1 | Logic for a process in the application | Python |
| 3. | Application Logic-2 | Logic for a process in the application | IBM Watson STT service |
| 4. | Application Logic-3 | Logic for a process in the application | IBM Watson Assistant |
| 5. | Database | Data Type, Configurations etc. | MySQL |
| 6. | Cloud Database | Database Service on Cloud | IBM DB2, IBM Cloudant etc. |
| 7. | File Storage | File storage requirements | IBM Block Storage or Other  Storage Service or Local Filesystem |
| 8. | External API-1 | Purpose of External API used in the application | Flask |
| 9. | External API-2 | Purpose of External API used in the application | Flask |
| 10. | Machine Learning Model | Purpose of Machine Learning Model | Supervised learning |
| 11. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud Local Server Configuration:  Cloud Server Configuration : | Local, IBM cloud |

TABLE-2: APPLICATION CHARACTERISTICS:

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | List the open-source frameworks used | Python web frameworks |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | e.g. SHA-256, Encryptions, IAM  Controls, OWASP etc. |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | Technology used |
| 4. | Availability | Justify the availability of application  (e.g. use of load balancers, distributed servers etc.) | Technology used |
| 5. | Performance | Design consideration for the performance of the application (number of requests per sec, use of  Cache, use of CDN’s) etc. | Technology used |



**Project planning and scheduling:**

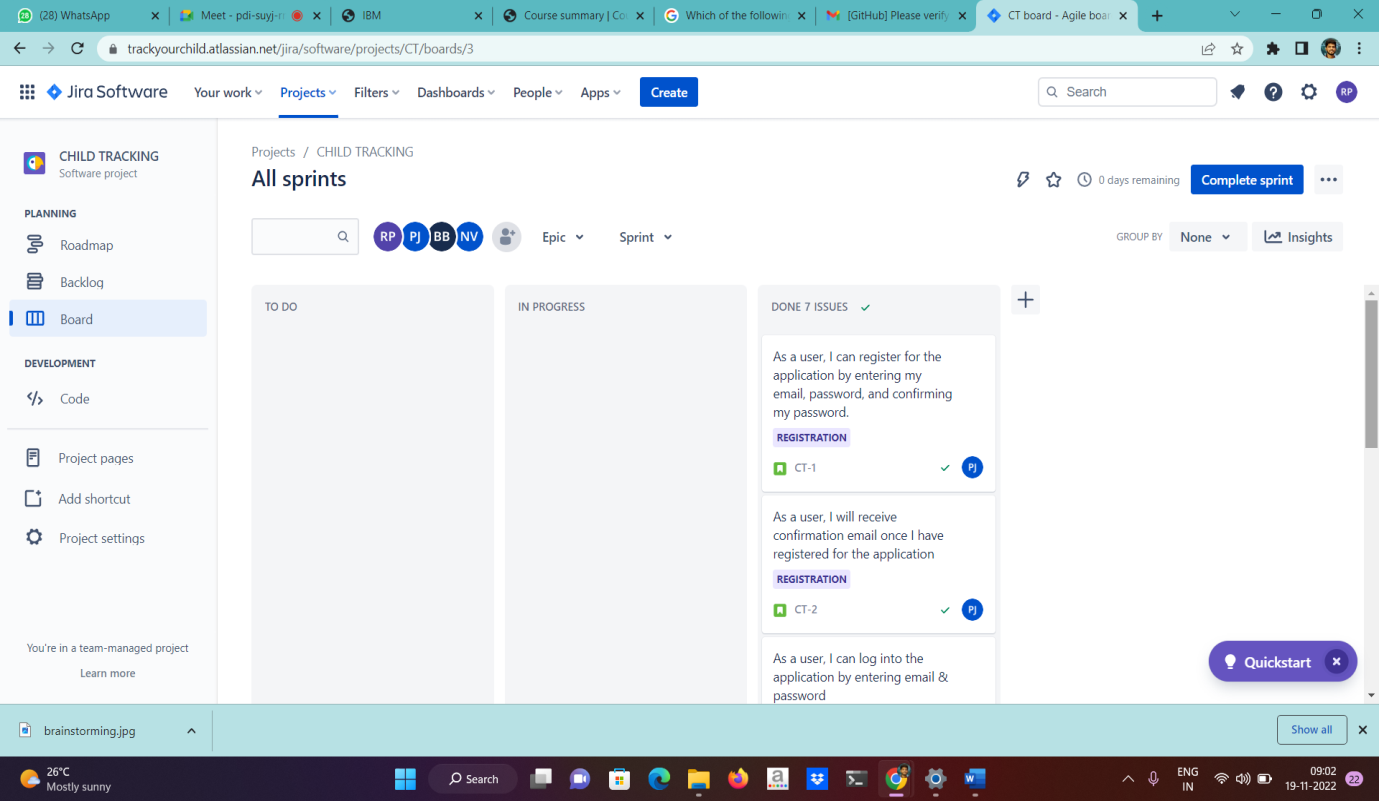
**Product Backlog, Sprint Schedule, and Estimation (4 Marks)**

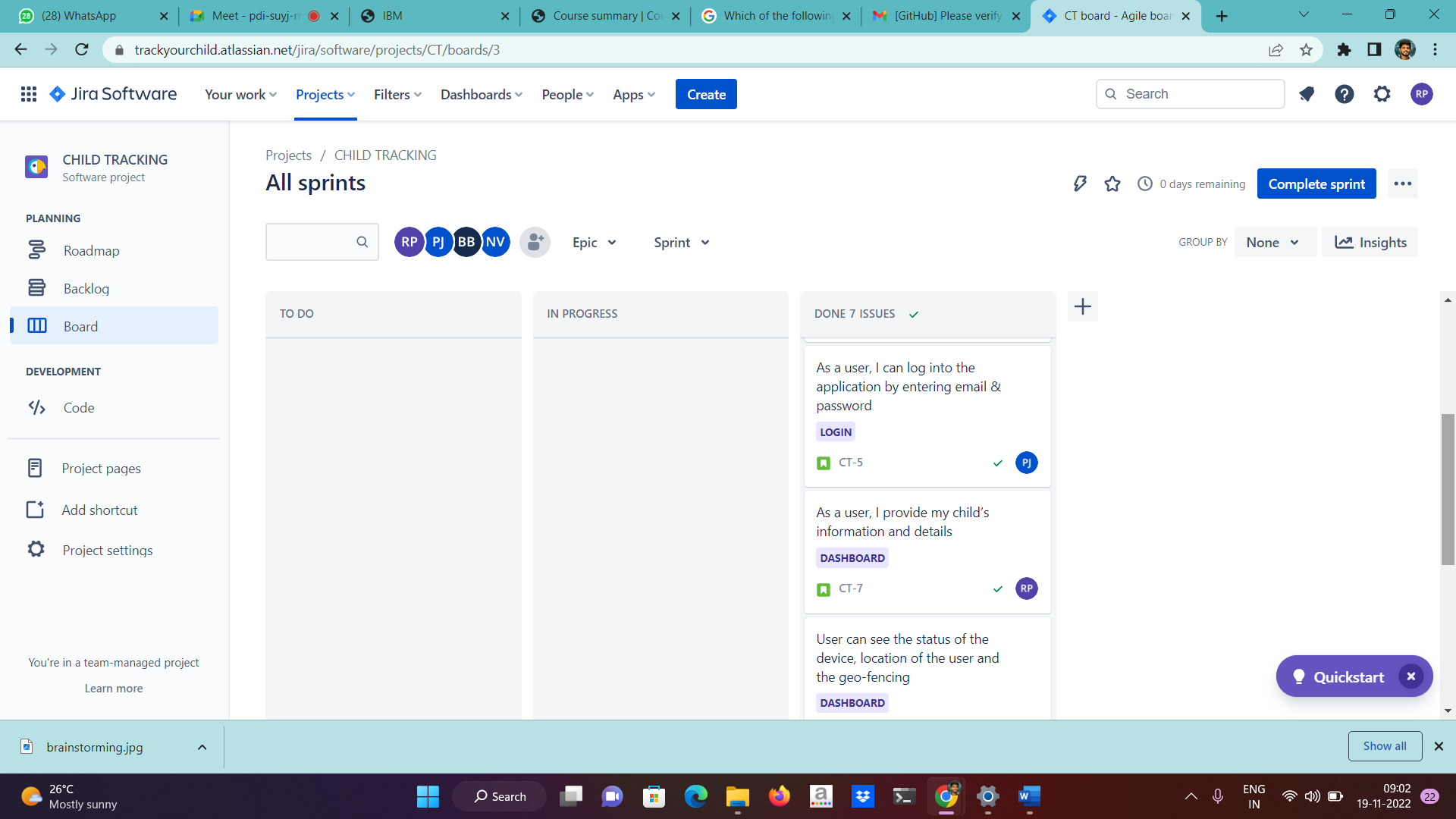
| **Sprint** | **Functional Requirement (Epic)** | **User Story Number** | **User Story / Task** | **Story Points** | **Priority** | **Team Members** |
| --- | --- | --- | --- | --- | --- | --- |
| Sprint-1 | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | 2 | High | Pratham |
| Sprint-1 |  | USN-2 | As a user, I will receive confirmation email once I have registered for the application | 1 | High | Pratham |
| Sprint-1 | Login | USN-3 | As a user, I can log into the application by entering email & password | 2 | High | Pratham |
| Sprint-2 | Dashboard | USN-4 | As a user, I provide my child’s information and details | 3 | High | Rohith |
| Sprint-2 |  | USN-5 | User can see the status of the device, location of the user and the geo-fencing | 2 | High | Rohith |
| Sprint-3 |  | USN-1 | As an administrator, I update the child location | 5 | Medium | Bharani |
| Sprint-4 |  | USN-1 | As a user ,after monitoring the child location if I have any doubts I can view the customer care phone number for additional clarification | 5 | Medium | Nathan |

**Project Tracker, Velocity & Burndown Chart: (4 Marks)**

| **Sprint** | **Total Story Points** | **Duration** | **Sprint Start Date** | **Sprint End Date (Planned)** | **Story Points Completed (as on Planned End Date)** | **Sprint Release Date (Actual)** |
| --- | --- | --- | --- | --- | --- | --- |
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 20 | 29 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 20 | 05 Nov 2022 |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 20 | 12 Nov 2022 |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 20 | 19 Nov 2022 |

**REPORTS FROM JIRA:**





**CODING &SOLUTIONING :**

**7.1 Feature 1**

In this system we uses the Node-Red to create a web Application for finding the location of the Child.The Parent has to create the username and password for the child.If the Parent enters the childsInformations on the Node-Red App ,The Geofence will search for the location and shows on the Screen.

**Coding :**

# HTML:

<!DOCTYPEhtml>

<htmllang="en">

<head>

<metacharset="UTF-8">

<metahttp-equiv="X-UA-Compatible"content="IE=edge">

<metaname="viewport"content="width=device-width,initial-scale=1.0">

<linkrel="stylesheet"href="/css/login.css">

<title>SignUp</title>

<script>

if (window.location.hostname !== "localhost") {if(location.protocol!=="https:"){

location.replace(

`https:${location.href.substring(location.protocol.length

)}`

)

}

}

</script>

<scriptsrc="./localforage.js"></script>

</head>

<body>

<divclass="wrapper">

<divclass="loginContainer">

<span>LogintoContinue</span>

<divclass="traditionalLoginContainer">

<formclass="signupForm"action="/"method="post">

<inputtype="text"name="firstName"placeholder="FirstName"id="firstName">

<inputtype="text"name="lastName"placeholder="LastName"id="lastName">

<inputtype="text"name="username"placeholder="UserName"id="username">

<inputtype="email"name="email"placeholder="Email"id="email">

<inputtype="password"name="password"placeholder="Password"id="password">

<inputclass="loginButton"type="submit"value="SignUp">

</form>

</div>

<divclass="loginWithFireContainer">

<button type="button" class="fire" title="Login with SAFETY" id="fire">Login withSAFETY</button>

</div>

<aclass="hyperLink"href="/login">AlreadyhaveanAccount?Login↗</a>

</div>

</div>

<script>

//NecessaryforFireOAuthtoFunction

const fireBroadcastingChannel = new BroadcastChannel('fireOAuthChannel');fireBroadcastingChannel.addEventListener('message',asyncevent=>{

letdata =event.data

/\*\*

* @typedef{Object<string,any>}Data
* @property{boolean}success-Whethertheloginwassuccessful
* @property{string}token-Thedata returnedfromthelogini.e.FireToken

\*/

//data.tokenisthemessagesentfromthefireOAuthChannelafterverification

//data.successisabooleanthatindicateswhethertheverificationwassuccessful

//data.tokenisthefire token

//What todowiththeFireToken?

// \* Fire Token is an unique token which uniquely identifies the user who authorized your login attemptwithFire

//\* YoucanusethistokenONLYONCEasitwillbedestroyedafterthefirstuse

//1.SendthefiretokentotheFireServertoverifythe user

//-Youcandothatclientsidedorserversided

//-YouneedtosendaPOSTRequesttotheFireServerwiththefiretoken

// attheURL:http://localhost:3003/api/tokens/verify

//-The FireServerwillverifythefiretokenandreturnaresponse

// - If the verification was successful - CODE (200), the Fire Server will return a response with theuser'sdata

// - If the verification was unsuccessful - CODE (400) or CODE (401), the Fire Server will return aresponsewithanerror'message'

//-YoucanusethedatareturnedfromtheFireServertocreate a newuserinyour database

// This example will send the token to Fire Servers and console.log the responseconsole.log("%c"+`FireToken:${data.token}`,`color:#f1c40f;font-weight:bold;`);

const response = await fetch('https://fire.adaptable.app/api/tokens/verify', {method:'POST',

headers:{

'Content-Type':'application/json'

},

body: JSON.stringify({token:data.token

})

})

//get theresponse

constresponseData=awaitresponse.json()

// console.log the responseconsole.log(responseData)

awaitlocalforage.setItem('userData',{...responseData,isFire:true})

//AddingtheuserdatatotheuserDatabase

let database = await localforage.getItem("userDatabase")if(database ==null) {

database=[]

}

database.push(responseData)

awaitlocalforage.setItem("userDatabase",database)

// redirect to the home pagewindow.location.href

function popupwindow(url, title, w, h) {varleft=(screen.width/2)-(w/2);

vartop=(screen.height/2)-(h/2);

return window.open(url, title, 'toolbar=no, location=no, directories=no, status=no, menubar=no,scrollbars=no,resizable=no,copyhistory=no,width='+w+',height='+h+',top='+top+',left='+left);

}

document.getElementById("fire").addEventListener("click", function() {popupwindow("/fireoauth.html","Fire OAuth",450,600)

})

</script>

<script>

// this.Website's Scripts / App Logicdocument.querySelector(".signupForm").addEventListener("submit",async(e)=>{

e.preventDefault()

let firstName = document.getElementById("firstName").valueletlastName = document.getElementById("lastName").valueletusername=document.getElementById("username").valueletemail=document.getElementById("email").value

let password = document.getElementById("password").valueletprofilePic=`https://avatars.dicebear.com/api/adventurer

neutral/${firstName}${lastName}.svg?backgroundColor=variant03`

let data = { firstName, lastName, username, email, password, profilePic }awaitlocalforage.setItem("userData",data)

let database = await localforage.getItem("userDatabase")if(database ==null) {

database=[]

}

database.push(data)

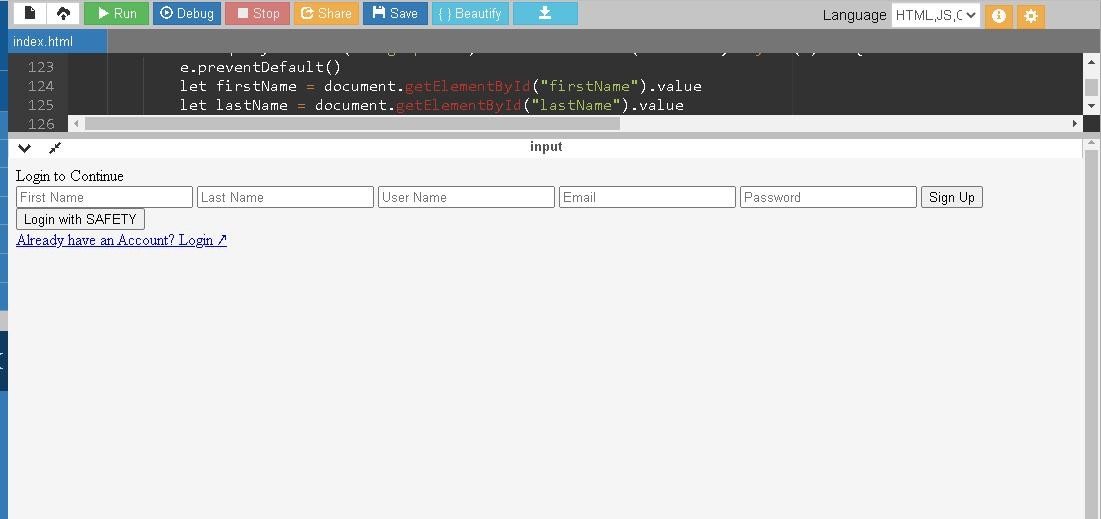
await localforage.setItem("userDatabase", database)window.location.href= "/"

})

</script>

</body>

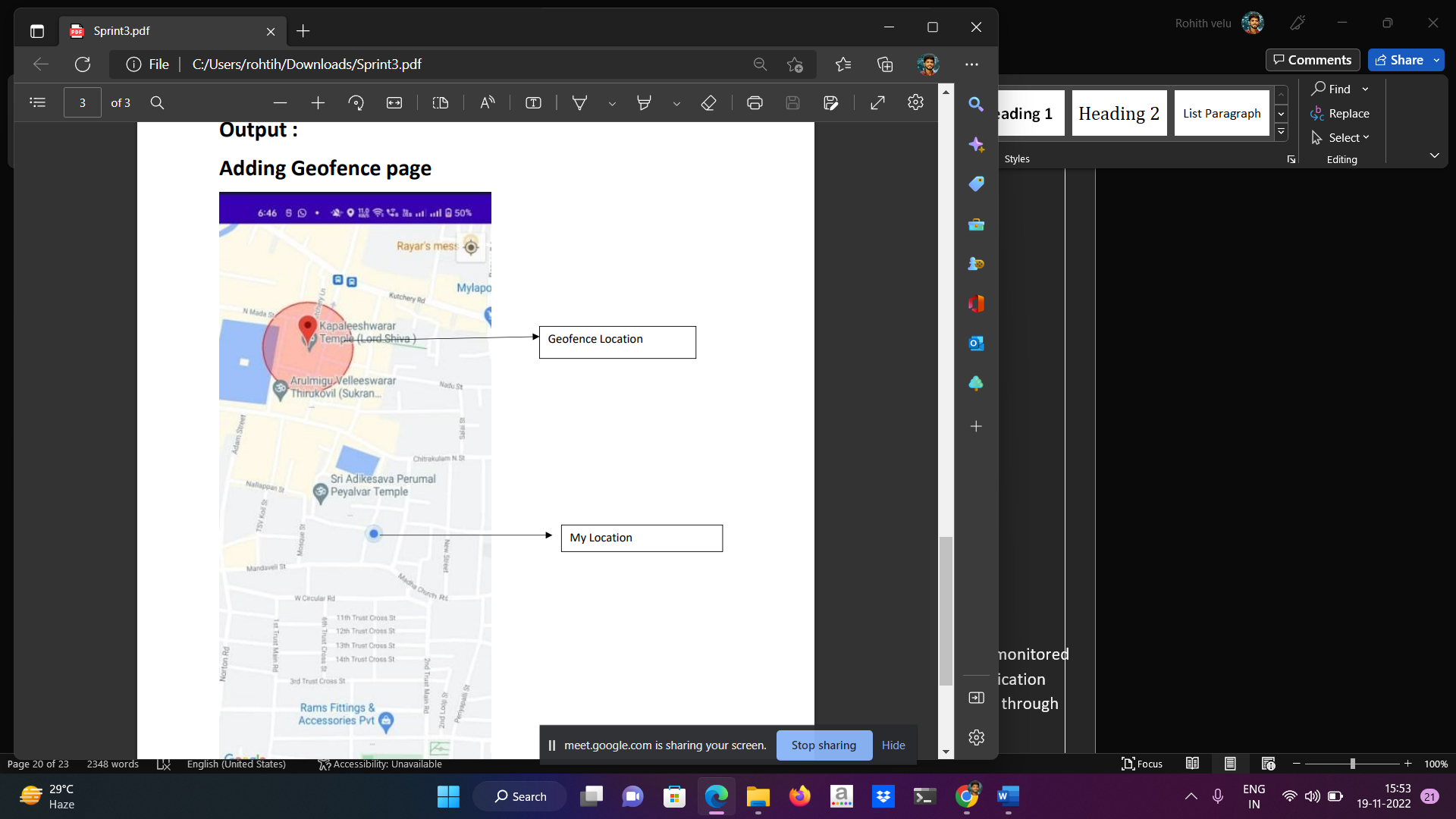
</html>

****

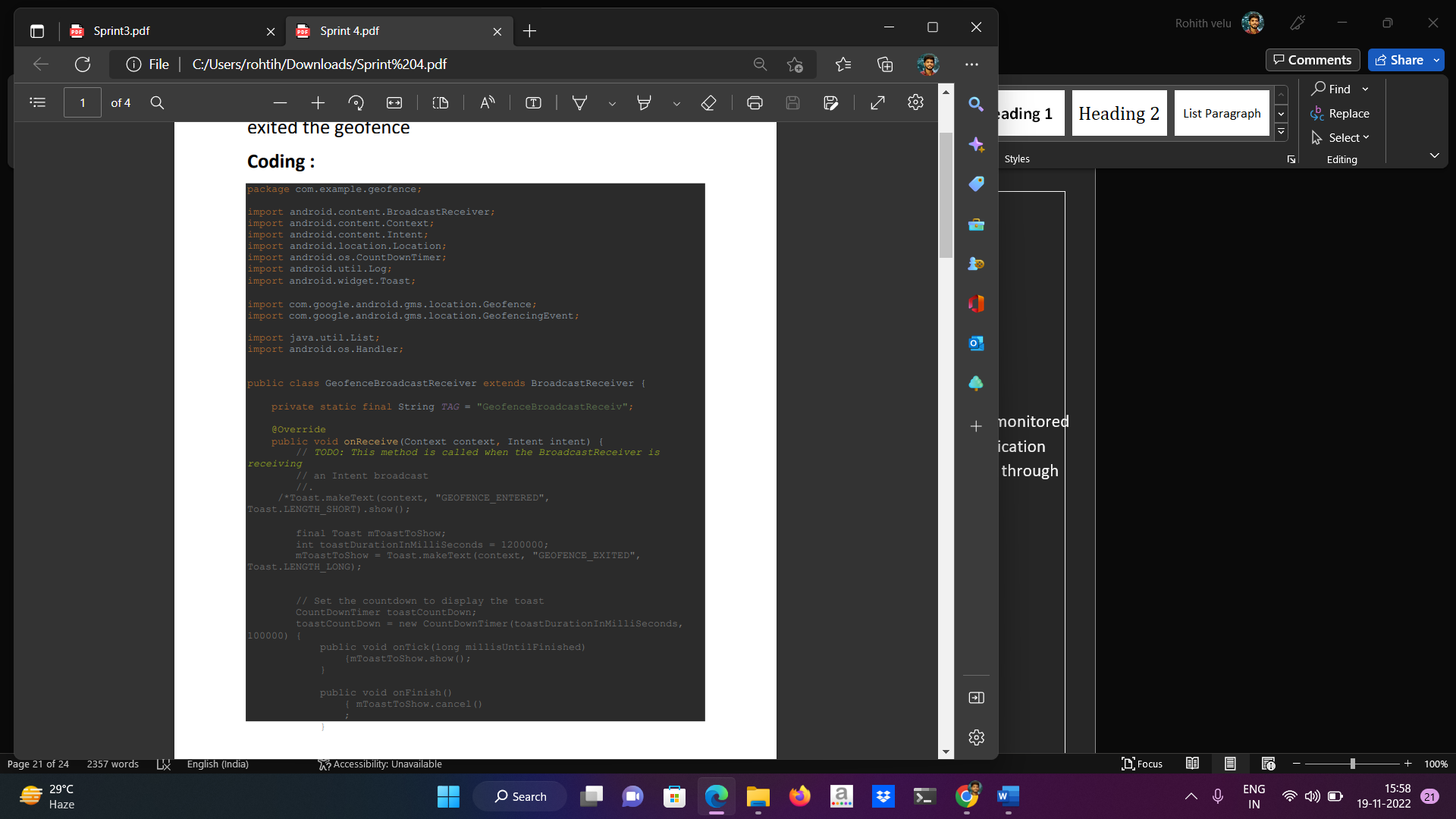
**Adding the geofence and changing the geofence when needed:**

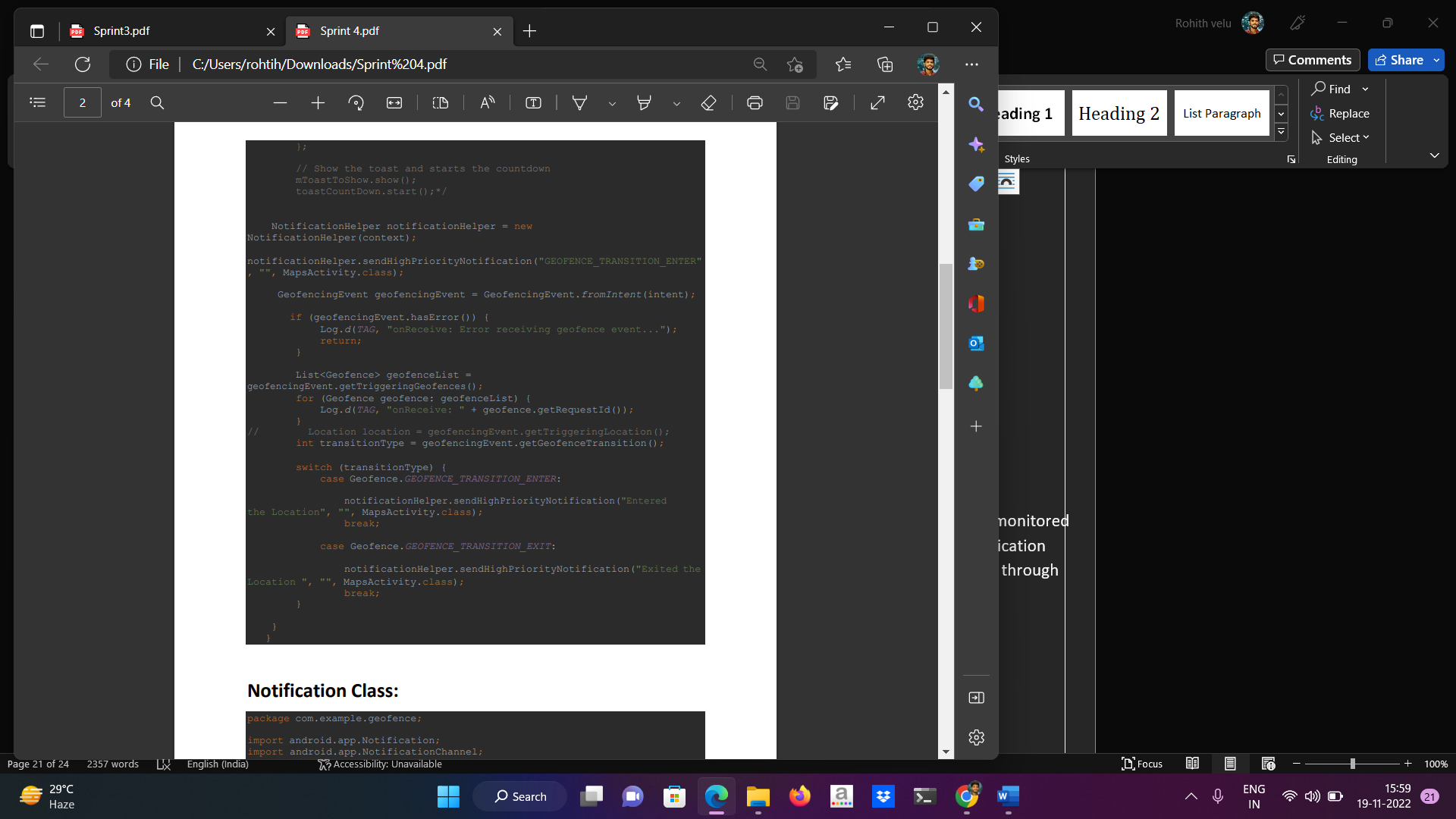
package com.example.geofence; import android.app.PendingIntent; import android.content.Context; import android.content.ContextWrapper; import android.content.Intent; import android.widget.Toast; import com.google.android.gms.common.api.ApiException; import com.google.android.gms.location.Geofence; import 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"; PendingIntentpendingIntent; public GeofenceHelper(Context base) {super(base); } public GeofencingRequestgetGeofencingRequest(Geofence geofence) {return new GeofencingRequest.Builder() .addGeofence(geofence) .setInitialTrigger(GeofencingRequest.INITIAL\_TRIGGER\_ENTER) .build(); } public Geofence getGeofence(String ID, LatLnglatLng, 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 PendingIntentgetPendingIntent() {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 instanceofApiException) { ApiExceptionapiException = (ApiException) e; switch (apiException.getStatusCode()) { 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"; } } return e.getLocalizedMessage()

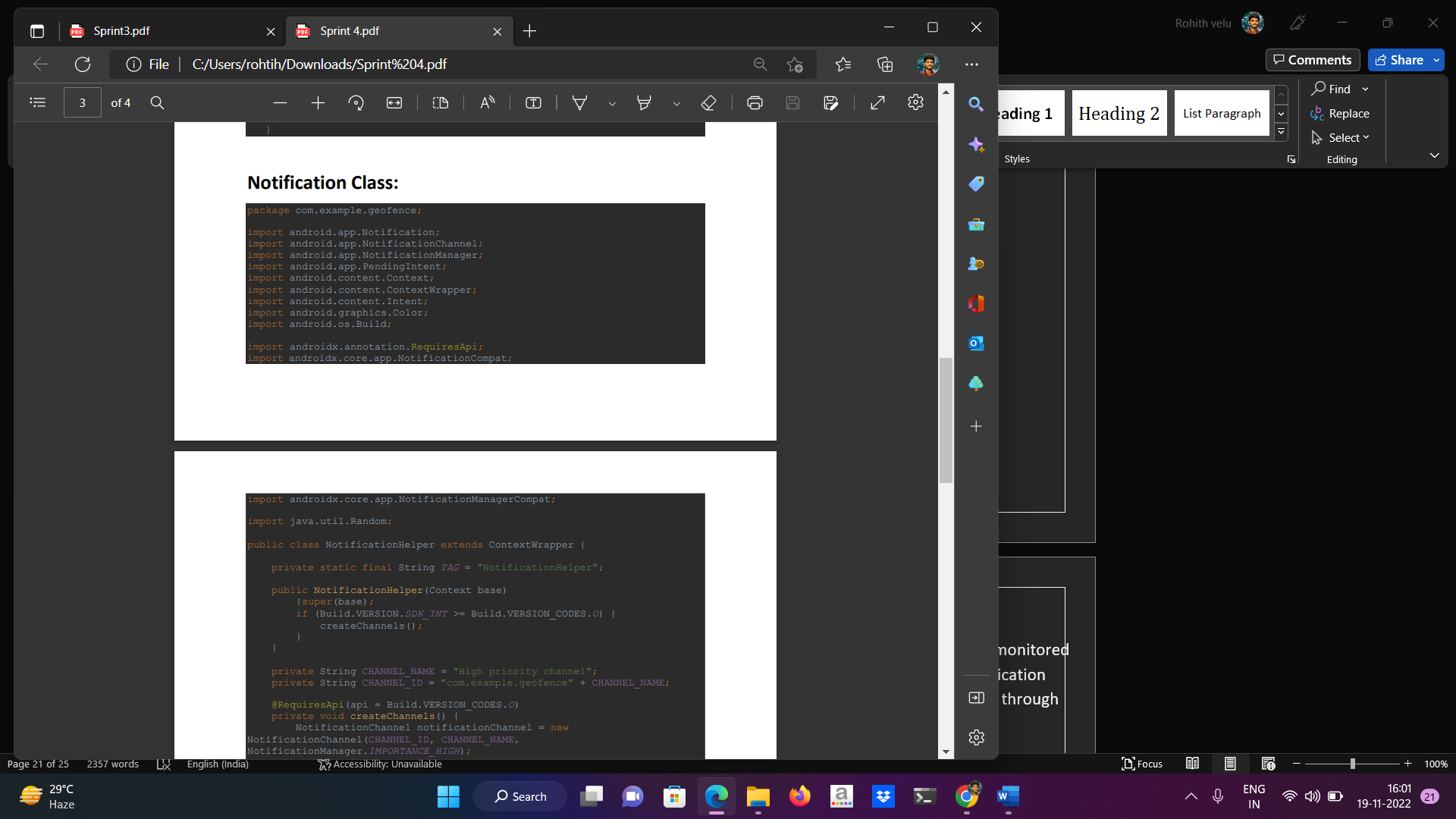
**OUTPUT:**

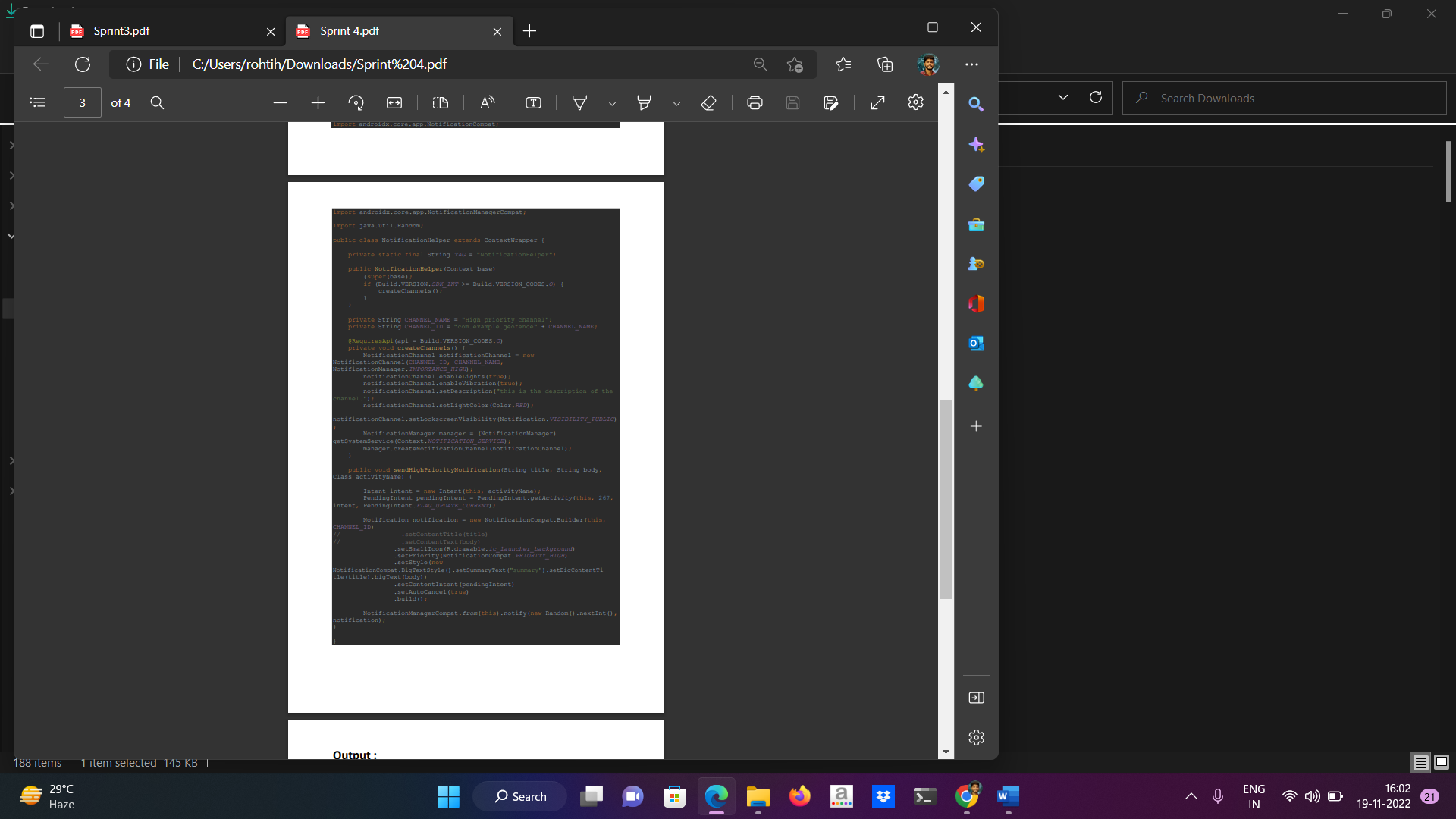


**Send alert notification when entered and exited geofence:**

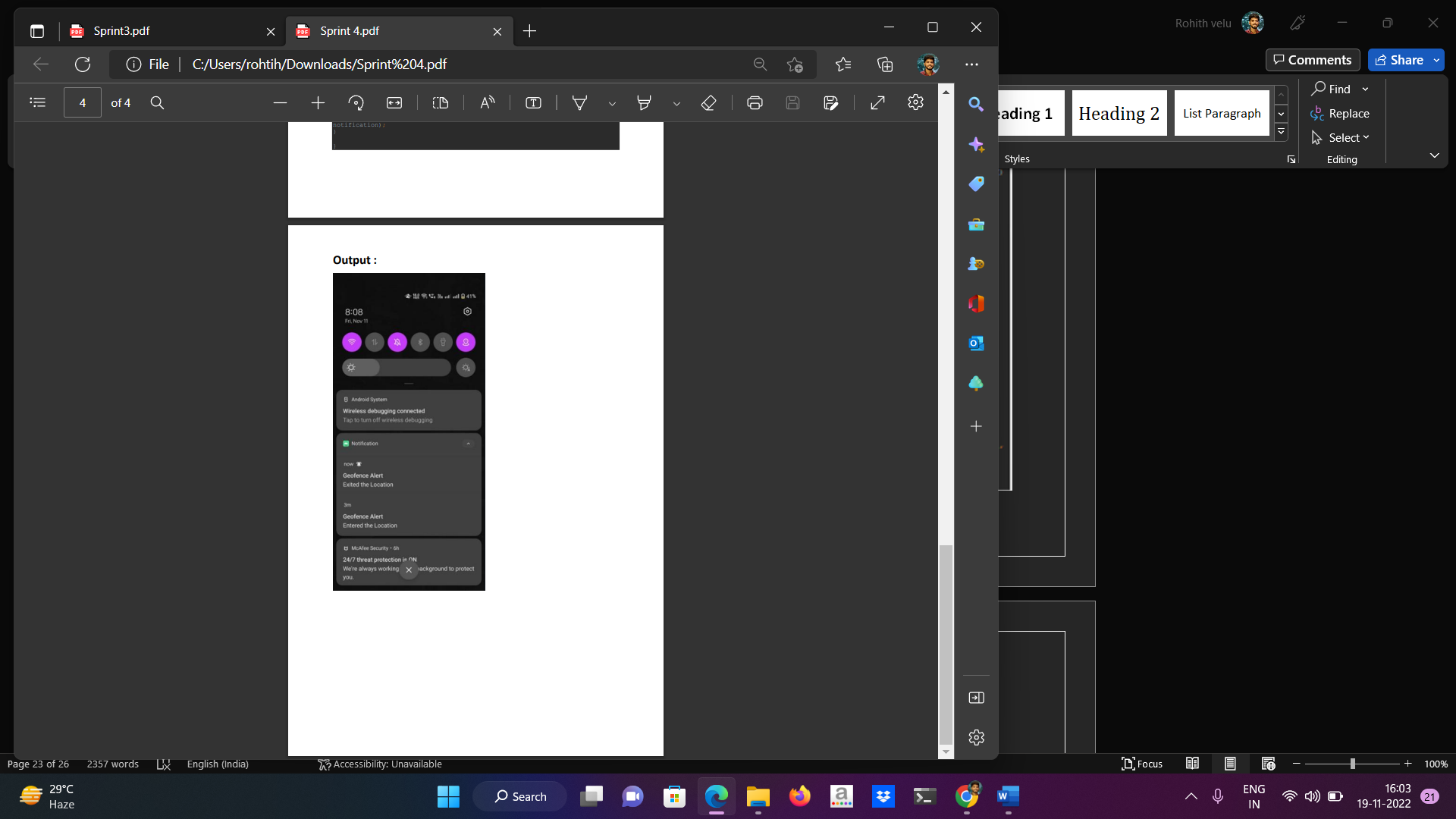








**Output:**



**7.2 Feature 2**

This system also consists of Wifi Module used to implement IOT and send all the monitored parameters to the cloud for android app monitoring on the parental device. Notification System also used in child Device to send the Alert message to the parental device through SMS.

**Advantages and Disadvantages :**

**Advantages:**

Heart-beats, temperature is monitored and the values areupdated to cloud continuously for parent app monitoring.Boundary monitoring system is implemented on safety gadgetwith the help of BEACON technology, as soon as the safetygadget moves far away from the binding gadget an alert isprovided to parent on binding gadget. the system is used tomonitor the health parameters and also used for locationtracking during necessary situations in safety concern

Disadvantages:

a. The child could not produce the exact alert command during a panic condition.

b. The command produced may not match with the previously stored command.

c. This project requires manual intervention.

**Conclusion:**

Throughout the research, it is clearly explained theIoT concept, child safety issues and the need of using child security system. Some previous studies have been included for designing the IoT-based child security smart band. It assists parents to monitor their children remotely.In case situations happen, notifications will be sent toparents so that actions can be taken. Through this, childsafety can be ensured and crime rate will be reduced. However, the proposed device is not robust enough and does not contain sufficient functions to operates like a mobile phone. Hence, the future enchantments will be adding more features, software, applications, hardware to make the proposed system capable of working more intelligently, meanwhile guarantee the safety of children

**Future Scope:**

This research demonstrates Smart IoT device for child safety and tracking helping the parents to locate and monitor their children. If any abnormal values are read by the sensorthen an SMS is sent to the parents mobile and an MMS indicating an image captured by the serial camera is also sent. The future scope of the work is to implement the IoT device which ensures the complete solution for child safety problems.

**References :**

* A. Jatti, M. Kannan, R. M. Alisha, P. Vijayalakshmi and S. Sinha, "Design and development of an IOT based wearable device for the safety and security of women and girl children," 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, 2016, pp. 1108-1112.
* David Hanes, Gonzalo, Patrick Grosetete, Robert, Barton, Jerome Henry “IoT Fundamental and Networking Technologies, Protocols”.