

## **Smart solutions for Railways**

**By :**

**PL.Ambiga(Team Leader)**

**S.Asmitha**

**R.Abitha**

**A.Eugin jebaraj**

**E.S.Sugirtha Angel**

## **Project report**

### **1. INTRODUCTION**

1.1 Project Overview

1.2 Purpose

### **2. LITERATURE SURVEY**

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

### **3. IDEATION & PROPOSED SOLUTION**

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

### **4. REQUIREMENT ANALYSIS**

4.1 Functional requirement

4.2 Non-Functional requirements

### **5. PROJECT DESIGN**

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

### **6. PROJECT PLANNING & SCHEDULING**

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

### **7. CODING & SOLUTIONING (Explain the features added in the project along with code)**

7.1 Feature

1 7.2 Feature

2 7.3 Database Schema (if Applicable)

### **8. TESTING**

8.1 Test Cases

8.2 User Acceptance Testing

### **9. RESULTS**

9.1 Performance Metrics

### **10. ADVANTAGES & DISADVANTAGES**

### **11. CONCLUSION**

### **12. FUTURE SCOPE**

### **13. APPENDIX**

# **1.INTRODUCTION:**

## **1.1Project overview:**

Smart solutions for railway is to manage Indian railways, which is the largest railway network in Asia and additionally it is the world's second largest network operated underneath a single management, this paper deals with the problem in medical during emergency. For example, if a person has a sudden heart attack or any emergency situation they can contact doctor in the railway by using IOT system attached in each compartment, which helps in reducing death rate in railway system

## **1.2 Purpose:**

- The IOT system has an app in it , which helps us to contact doctor immediately during emergency situation
- The call option in the app, will help us to contact the doctor and we can explain the emergency situation
- Incase of some other illness like headache, cold etc, we can send the message to the doctor
- By using this app, we can easily manage the medical emergencies in railway system and can give better health safety.

## 2) LITERATURE SURVEY:

### 2.1 & 2.2 Existing problem and References:

1)

**Paper Title** : A Smart Railway Track Fault Detection Using IOT

**Authors** : Vatsala Sharma,Kamal Nayanam,Himani.

**Publication** : IOSRJEN, August 2021, Volume:11, Issue:08

**Methodology** : Using Raspberry pi, ,Ultrasonic Sensor, ,IOT.

#### **Abstract :**

The proposed system introduces technology, to prevent railway accident. An Ultrasonic sensor is installed that monitors the track and provides status to the Raspberry pi controller. If any crack is detected it immediately sends the cloud crunch to the user's mobile phone. The project uses a stepper motor assembly in which the ultrasonic sensor is attached and the ultrasonic sensor moves to the side of the track and in case of error anywhere the trackreceives and provides information to the railway department. If there is any fault detected on the track, the ultrasonic sensor attached to the system will send out ultrasonic waves and indicates the distance which is less than the preset distance and indicates a fault in track and the fault information to monitor screen of the railway department. This project is very helpful in preventing accidents and provides information before the railway department about the cracks so they can get information about the cracks and can be fixed soon.

#### **Components :**

It consists of Power supply, Stepper Motor, Raspberry Pi, Ultrasonic Sensor,Buzzer and LCD display and the output is given to Cloud server.

#### **Advantages :**

Accidents occurring in Railway transportation system cost a large number of lives ,So this system helps us to prevent accidents and giving information about faults or cracks in advance to railway authorities. So that they can fix them and accidents cases becomes less.

This project is cost effective. By using more techniques they can be modified and developed according to their applications

#### **Disadvantages :**

The ultrasonic Sensor has limited range. Temperature Fluctuation affects the speed of an ultrasonic sensors pulse or sound waves, Even though the target has not likely moved or shifted, it will appear that the target is closer due to the increase in temperature, thus we don't get accurate results.

2)

**Paper Title** : Automation of railway gate Using IOT

**Authors** : M. Abinaya , Vidya , Thenmozhi

**Publication** : IJERT (ISSN: 2278-0181)

**Methodology** : Using Raspberry pi, IR Sensor, IOT.

**Abstract** :

The railway system in India and other countries is the most commonly used transportation mode and it is also a one of the low cost transportation mode. At present, country like India is having world's largest railway network in the world. There are thousands of rails running on track every day. In railway system, it is impossible to stop some of the critical situations or emergencies which are arising during the running of train. The RFID reader reads the ID of train, information related to train means date, time, train RFID number are sent to central server through internet. Central server receives the information from pi, immediately it will fetch all phone numbers registered to train based on RFID number and send the notification message to all phone numbers. An IR proximity sensor detects the presence of train/object without any physical contact. The proposed system consists of 2 LEDs- red and yellow. When the train arrives red LED will glow and when the train departs yellow LED will glow.

**Components** :

It consists of IR sensors, actuators , LED Raspberry pi and RFID devices

**Advantages** :

Automatic gate control system using IoT is an effective and advanced method to reduce the occurrence of railway accidents. This system provides the lot of benefits to the road users and railway management

**Disadvantages** :

Data transfer speed is slow in Infra red and to establish a entire network initialization

3)

**Paper Title :** Design and Simulation of a PLC and IoT-based Railway Level Crossing Gate Control and Track Monitoring System using LOGO

**Authors :** Muhibul Haque, Bhuyan, Sheikh Md. Mamunur Rahman, Md. Tofayel Tarek

**Publication :** (IOSR-JEEE) Volume 17, Issue 2 (Mar. – Apr. 2022))

**Methodology :** Using LOGO software with the PLC, IR sensors, GUI & IOT.

**Abstract :**

In this paper, the main concern is to design a PLC and an IoT-based automatic interlocking system to protect trains from accidents or clashes and ensure the safety and security of the passengers. Therefore a model has been designed using LOGO software with the PLC as the main heart of the control system. The ladder logic program for PLC was developed using LOGO software on a personal computer and then downloaded into the PLC. For communication between the PLC and computer, an RS485 serial port was used. To detect the presence of a train on the track, ultrasonic and radio frequency (RF) transducers and infrared (IR) sensors were used as both the transmitters and receivers. The system was simulated using LOGO software. In the system, we have incorporated Internet of Things (IoT) and Visual Basic Programming software to connect it with the mobile Apps and create Graphic User Interface (GUI) respectively. When the train passes out from the level crossing gate then the end side's second sensor gets HIGH and gives a signal to the PLC by which the PLC Sends the signal to open up the barriers and then the signals come in their normal positions

**Components :**

Power supply 5V DC (LM7805), power supply 24V DC (LM7824), PLC, Buzzer, DC motor, OLED Display, ultrasonic sensor, RF Tx-Rx, IR Tx-Rx, Mobile apps

**Advantages:**

This method would automate our system, reduce the manpower requirements, the manual operation would be replaced, and hence it would be cost-effective as well.

**Disadvantages :**

These efforts are still under exploration stage, especially in Bangladesh and have not been practically implemented so far. cost is high.

4)

**Paper Title** : Analysis of smart railway gate control

**Authors** : Lakshmi , Maria Viji

**Publication** : (SSRG - IJCSE) - Volume 6 Issue 2

**Methodology** : Using Arduino , IR Sensor, IOT.

**Abstract** :

The main aim of this paper represents a model for ARGC (Automatic railway gate control) using Arduino UNO R3. In the existing model has a serious disadvantage that is it cannot only detect train but also other objects (like animals, human) and it doesn't deal with the trespassers who are already into the gate. The proposed model overcomes the inaccurate identifying of objects and also manages the trespassers who are already into the gate. The ultrasonic sensor placed at the front of the train will sense for the obstacle on the track. At the beginning, the IR sensor1 senses for the detection of arriving train and the output of sensor goes HIGH, that is when the sensor1 senses the train .The output of the ultrasonic sensor goes HIGH when the obstacle is detected, the train will be stopped and the message of obstacle will be sent to the nearby railway station so that it can be cleared.

**Components** :

Ultrasonic Sensor, IR sensors , Motor driver L293d , Arduino UNO R3, and IR LEDs.

**Advantages** :

Smart automatic railway gate control system provides a best and the safest way of transportation for not only passengers inside the train but also for the trespassers in and around.

The main advantage of this system is its cost effectiveness as it can be used in all countries independent of what area it can be.

**Disadvantages** :

The Arduino board is a delicate device , so it has to be handled carefully and IR sensor range is limited.

5)

**Paper Title** : IOT Based Fault Control in Railway Track System

**Authors** :Bagyalakshmi.N , Oviya .N

**Publication** : (IJAREEIE) Volume 8, Special Issue 1, March 2019

**Methodology** : Using Arduino ATmega328P Microcontroller

**Abstract** :

This project concern to a process for monitoring the condition of train tracks and more specifically has the object of the identification of defects When the supply is given to the device, the DC motor gets start through relay driver circuit. Two IR sensors are fixed in front of the train which is used to find out the crack on the track. Each sensor will produce the signal related to the position with the rail. If the track position is normal both the sensor gives the constant sensed output. If any sensor misses their output condition to fail then there is defect on that side. Ultrasonic sensor is used to detect the obstacle in the track if any obstacle detected means the Arduino automatically transmits the message to nearby station via IOT using GPRS MODEM

**Components** : IR sensors , GPS , and LCD

**Advantages** :

This project presents the implementation results of the railway track crack control system using simple components

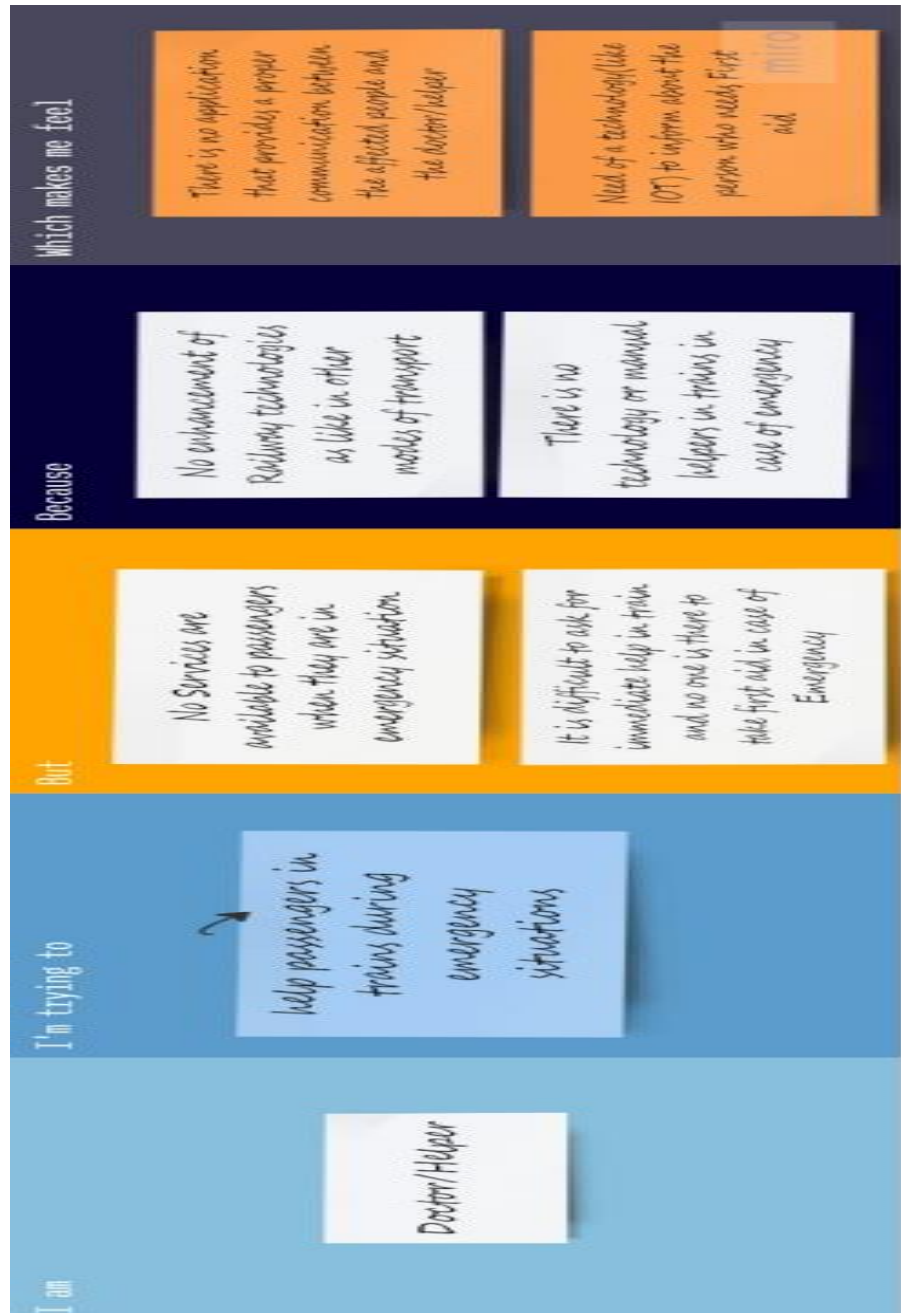
This is cost effective railway track crack detection system with greater accuracy and High speed information transferring after crack detection.

**Disadvantages:**

Arduino board have to be handled carefully.

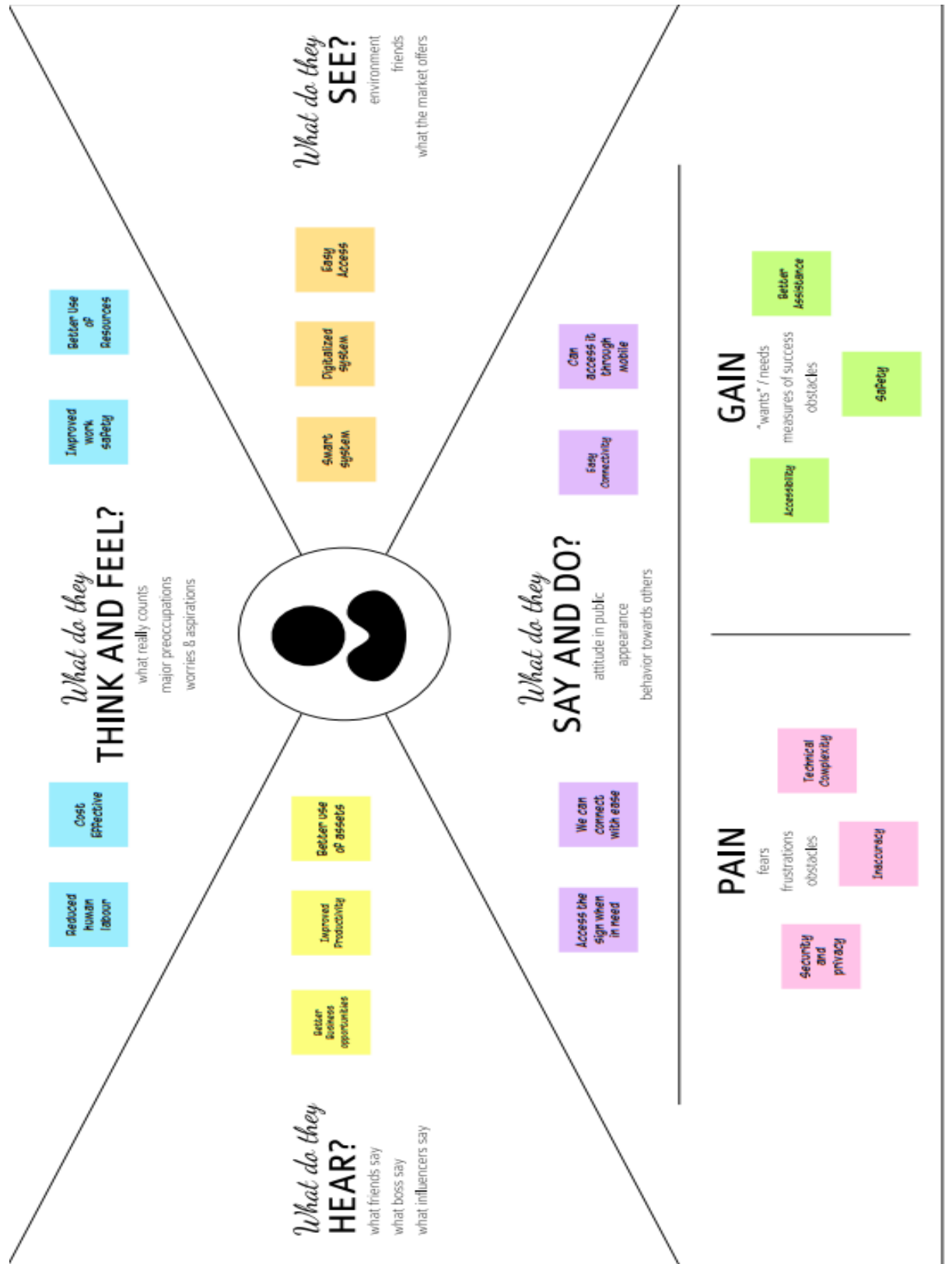


## 2.3) Problem Statement Definiton:



### 3) Ideation & Proposed Solution:

#### 3.1 Empathy Map Canvas:



### 3.2 Ideation & Brainstorming:

**Brainstorm & idea prioritization**

Use this template in your next brainstorming session to get your team to generate ideas, evaluate them, and select the most promising ones to develop.

**1. Brainstorm ideas**

**2. Evaluate ideas**

**3. Select ideas to develop**

**4. Develop ideas**

**5. Implement ideas**

**6. Review ideas**

**7. Share ideas**

**8. Iterate**

**9. Refine**

**10. Finalize**

**11. Launch**

**12. Monitor**

**13. Evaluate**

**14. Iterate**

**15. Refine**

**16. Finalize**

**17. Launch**

**18. Monitor**

**19. Evaluate**

**20. Iterate**

**21. Refine**

**22. Finalize**

**23. Launch**

**24. Monitor**

**25. Evaluate**

**26. Iterate**

**27. Refine**

**28. Finalize**

**29. Launch**

**30. Monitor**

**31. Evaluate**

**32. Iterate**

**33. Refine**

**34. Finalize**

**35. Launch**

**36. Monitor**

**37. Evaluate**

**38. Iterate**

**39. Refine**

**40. Finalize**

**41. Launch**

**42. Monitor**

**43. Evaluate**

**44. Iterate**

**45. Refine**

**46. Finalize**

**47. Launch**

**48. Monitor**

**49. Evaluate**

**50. Iterate**

**51. Refine**

**52. Finalize**

**53. Launch**

**54. Monitor**

**55. Evaluate**

**56. Iterate**

**57. Refine**

**58. Finalize**

**59. Launch**

**60. Monitor**

**61. Evaluate**

**62. Iterate**

**63. Refine**

**64. Finalize**

**65. Launch**

**66. Monitor**

**67. Evaluate**

**68. Iterate**

**69. Refine**

**70. Finalize**

**71. Launch**

**72. Monitor**

**73. Evaluate**

**74. Iterate**

**75. Refine**

**76. Finalize**

**77. Launch**

**78. Monitor**

**79. Evaluate**

**80. Iterate**

**81. Refine**

**82. Finalize**

**83. Launch**

**84. Monitor**

**85. Evaluate**

**86. Iterate**

**87. Refine**

**88. Finalize**

**89. Launch**

**90. Monitor**

**91. Evaluate**

**92. Iterate**

**93. Refine**

**94. Finalize**

**95. Launch**

**96. Monitor**

**97. Evaluate**

**98. Iterate**

**99. Refine**

**100. Finalize**

**101. Launch**

**102. Monitor**

**103. Evaluate**

**104. Iterate**

**105. Refine**

**106. Finalize**

**107. Launch**

**108. Monitor**

**109. Evaluate**

**110. Iterate**

**111. Refine**

**112. Finalize**

**113. Launch**

**114. Monitor**

**115. Evaluate**

**116. Iterate**

**117. Refine**

**118. Finalize**

**119. Launch**

**120. Monitor**

**121. Evaluate**

**122. Iterate**

**123. Refine**

**124. Finalize**

**125. Launch**

**126. Monitor**

**127. Evaluate**

**128. Iterate**

**129. Refine**

**130. Finalize**

**131. Launch**

**132. Monitor**

**133. Evaluate**

**134. Iterate**

**135. Refine**

**136. Finalize**

**137. Launch**

**138. Monitor**

**139. Evaluate**

**140. Iterate**

**141. Refine**

**142. Finalize**

**143. Launch**

**144. Monitor**

**145. Evaluate**

**146. Iterate**

**147. Refine**

**148. Finalize**

**149. Launch**

**150. Monitor**

**151. Evaluate**

**152. Iterate**

**153. Refine**

**154. Finalize**

**155. Launch**

**156. Monitor**

**157. Evaluate**

**158. Iterate**

**159. Refine**

**160. Finalize**

**161. Launch**

**162. Monitor**

**163. Evaluate**

**164. Iterate**

**165. Refine**

**166. Finalize**

**167. Launch**

**168. Monitor**

**169. Evaluate**

**170. Iterate**

**171. Refine**

**172. Finalize**

**173. Launch**

**174. Monitor**

**175. Evaluate**

**176. Iterate**

**177. Refine**

**178. Finalize**

**179. Launch**

**180. Monitor**

**181. Evaluate**

**182. Iterate**

**183. Refine**

**184. Finalize**

**185. Launch**

**186. Monitor**

**187. Evaluate**

**188. Iterate**

**189. Refine**

**190. Finalize**

**191. Launch**

**192. Monitor**

**193. Evaluate**

**194. Iterate**

**195. Refine**

**196. Finalize**

**197. Launch**

**198. Monitor**

**199. Evaluate**

**200. Iterate**

**201. Refine**

**202. Finalize**

**203. Launch**

**204. Monitor**

**205. Evaluate**

**206. Iterate**

**207. Refine**

**208. Finalize**

**209. Launch**

**210. Monitor**

**211. Evaluate**

**212. Iterate**

**213. Refine**

**214. Finalize**

**215. Launch**

**216. Monitor**

**217. Evaluate**

**218. Iterate**

**219. Refine**

**220. Finalize**

**221. Launch**

**222. Monitor**

**223. Evaluate**

**224. Iterate**

**225. Refine**

**226. Finalize**

**227. Launch**

**228. Monitor**

**229. Evaluate**

**230. Iterate**

**231. Refine**

**232. Finalize**

**233. Launch**

**234. Monitor**

**235. Evaluate**

**236. Iterate**

**237. Refine**

**238. Finalize**

**239. Launch**

**240. Monitor**

**241. Evaluate**

**242. Iterate**

**243. Refine**

**244. Finalize**

**245. Launch**

**246. Monitor**

**247. Evaluate**

**248. Iterate**

**249. Refine**

**250. Finalize**

**251. Launch**

**252. Monitor**

**253. Evaluate**

**254. Iterate**

**255. Refine**

**256. Finalize**

**257. Launch**

**258. Monitor**

**259. Evaluate**

**260. Iterate**

**261. Refine**

**262. Finalize**

**263. Launch**

**264. Monitor**

**265. Evaluate**

**266. Iterate**

**267. Refine**

**268. Finalize**

**269**

### 3.3) Proposed Solution :

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Railway passengers need <b>hospitability</b> in the case of any <b>emergency</b>
2.	Idea / Solution description	<ul style="list-style-type: none"> <li>➤ Smart railway using <b>iot</b> (hospitability) , by connecting passenger using smart <b>iot</b> system by <b>tab</b> device</li> <li>➤ In case of emergency, the affected person can <b>touch</b> the <b>tab</b> emergency button</li> <li>➤ That will alert the <b>helper</b> in a train, who is also <b>connected</b> with alarm</li> <li>➤ Helper is <b>assigned</b> for emergency needs, who knows first aid and have all <b>emergency</b> kit</li> </ul>
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> <li>➤ Using <b>iot</b> ,which is the main for this system, power source for the helper in the form of <b>tab</b> device</li> <li>➤ <b>Ultrasonic sensor</b> which performs the function of alerting the helper , that the passenger needs immediate help</li> <li>➤ <b>Bolt-iot cloud</b> which is software website we have to create</li> <li>➤ <b>Adruino- uno</b> which is used in this system for easy to use programmable open source microcontroller board</li> </ul>
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> <li>➤ This system will help the people who suffer in <b>heart attacks</b> and other main problems</li> <li>➤ This system will help for <b>hospitability</b> and other needs also</li> <li>➤ This will create a <b>awareness</b> to people and when they came to</li> </ul>

		know they will <b>feel safe</b> to travel in train
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> <li>❖ <b>Adruino</b> is flexible , <b>low cost</b> and easy to programmable</li> <li>❖ <b>Ultrasonic sensor</b> can be used in <b>dark</b> environment , it will not affect by colour or <b>transparency</b> of objects</li> </ul>
6.	Scalability of the Solution	➤ <b>95%</b> , it can reduce emergency needs of passenger, if they need <b>immediate</b> need helper will be there <b>24/7 to help</b>

### 3.4) Problem Solution Fit:

<p><b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span></p> <p>Who is your customer? i.e. working parents of 0-5 y.o. kids</p> <p>A passenger who needs medical help during emergency situation</p>	<p><b>6. CUSTOMER CONSTRAINTS</b></p> <p>What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.</p> <p>Not all people have iot enabled devices and know how to operate it</p>	<p><b>5. AVAILABLE SOLUTIONS</b> <span>AS</span></p> <p>Which solutions are available to the customers when they face the or need to get the job done? What have they tried in the past? What pros &amp;</p> <p>They can contact the doctor/helper by using iot enabled devices</p>
--	---	---

<p><b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span></p> <p>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one serious different</p> <p>If a person in the train facing health issue, it will be difficult for the passenger to communicate with the doctor/helper</p>	<p><b>9. PROBLEM ROOT CAUSE</b> <span>RC</span></p> <p>What is the real reason that this problem exists? What is the back story behind the need to do this job?</p> <p>Unavailability of devices to use in the train in order to seek help at emergency situation</p>	<p><b>7. BEHAVIOUR</b> <span>BE</span></p> <p>What does your customer do to address the problem and not the i.e. directly related: find the right solar panel installer, calculate usage and</p> <p>They inform their issues to the railway transport system committee</p>
---	---	--

<p><b>3. TRIGGERS</b> <span>TR</span></p> <p>Since technologies are emerging well, they feel why don't they implement such technology in railway systems.</p> <p><b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span></p> <p>Whenever their loved ones.. facing emergency situation, they feel angry if they can't able to help them., even though we are having enormous technologies</p>	<p><b>10. YOUR SOLUTION</b> <span>SL</span></p> <p>If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.</p> <p>We can fix an in built iot enabled devices, which will make a communication between a passenger seeking help in train and the doctor/helper</p>	<p><b>8. CHANNELS of BEHAVIOUR</b> <span>CH</span></p> <p><b>8.1 ONLINE</b> What kind of actions do customers take online? Extract online channels from #7</p> <p>They address their issues through the feedback portal or try to contact nearby station for help</p> <p><b>8.2 OFFLINE</b> What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.</p> <p>Passenger call out for help to ticket checker, or other doctor available in the train at that moment</p>
--	---	---

## 4) Requirement Analysis :

### 4.1) Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Message to Doctor	Message through <b>SMS</b> Message through <b>Call over Internet</b>
FR-2	User inputs	An IOT device which has various options ,thus it helps the user to communicate better.
FR-3	Web Application	It is done by using IBM IOT platform and with the help of NODE red app.
FR-4	IOT device	We can make use of tabs in the compartment of the train.

### 4.2) Non-functional Requirements:

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

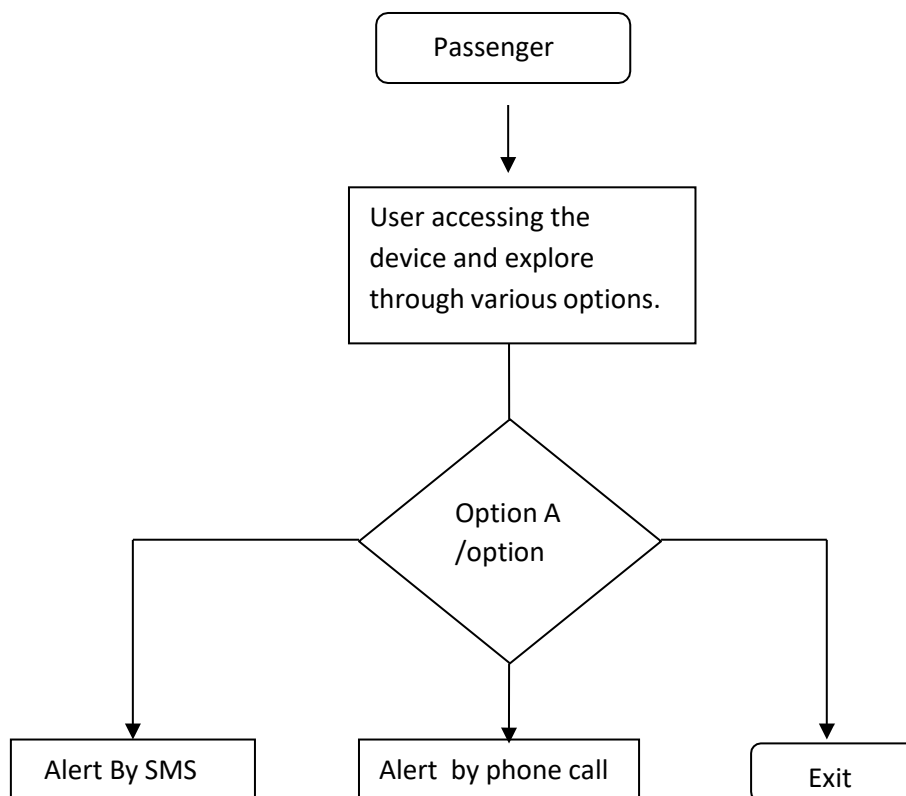
FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	In Train,We are having an in – built device,which will be <b>easy to use</b> by the passengers.
NFR-2	<b>Security</b>	For this process it doesn't require any personal details. Hence, <b>high</b> Security.
NFR-3	<b>Reliability</b>	Officials of the Railway will install this and create awareness ,thus it is <b>more</b> reliable.
NFR-4	<b>Performance</b>	This is a new technology,which has <b>high</b> performance ,since we are using an IOT device.
NFR-5	<b>Availability</b>	It is fixed in the compartments of the train and hence it is available <b>easily</b> .
NFR-6	<b>Scalability</b>	If it's usage is more,we can increase the no of in – built IOT device,thus it is scalable.

## 5) Project Design:

### 5.1) Data Flow Diagrams:

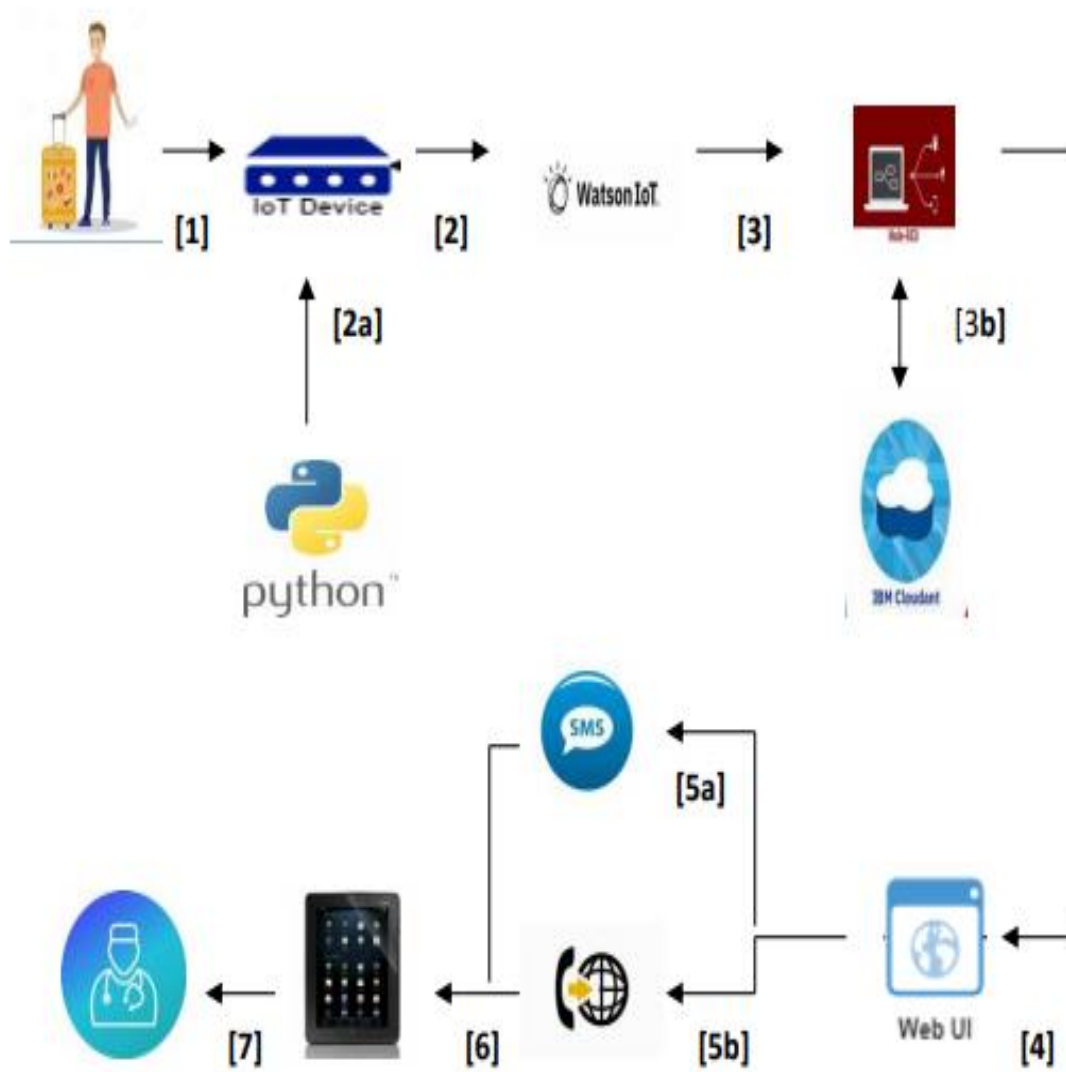
1. User gives input to the IOT device
2. Python code is used to build the application
3. In order to create this application ,we make use of this IBM IOT platform ,NODE app.
4. A web link is create, which send the user's alert message
5. Alert message via SMS
  - a) Alert message via Call over Internet
6. These messages will be received by an another iot device at the other end of the use
7. The doctor access these messages via tab or laptop,

### DFD:





## 5.2) Solution and technical architecture:



### 5.3 ) User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Accessing the Web using an IOT device	USN-1	As a user, I can use the Application by clicking the screen on the Tab	I can access the Tab	High	Sprint-1
		USN-2	As a user, If I don't know how to use it, I can ask help from Ticket checker in the train or from other passenger.	By the in-built IOT device	Medium	Sprint-2
Customer (Web user)	User Option	USN-1	Alert Message via SMS	By the Web	Medium	Sprint-3
		USN-2	Alert Message via Call over Internet	By the Web	High	Sprint-4

6) Project Planning & Scheduling:

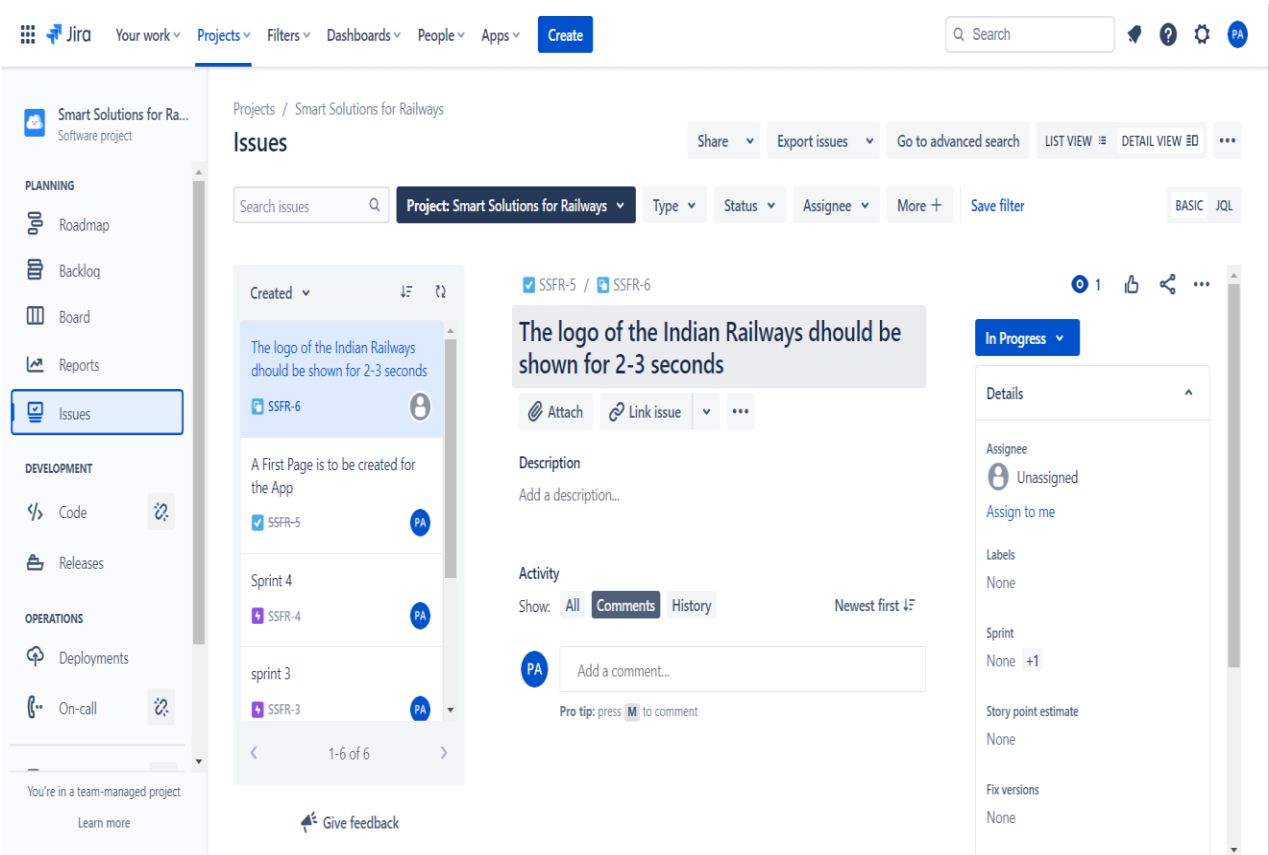
6.1) Sprint planning and estimation:

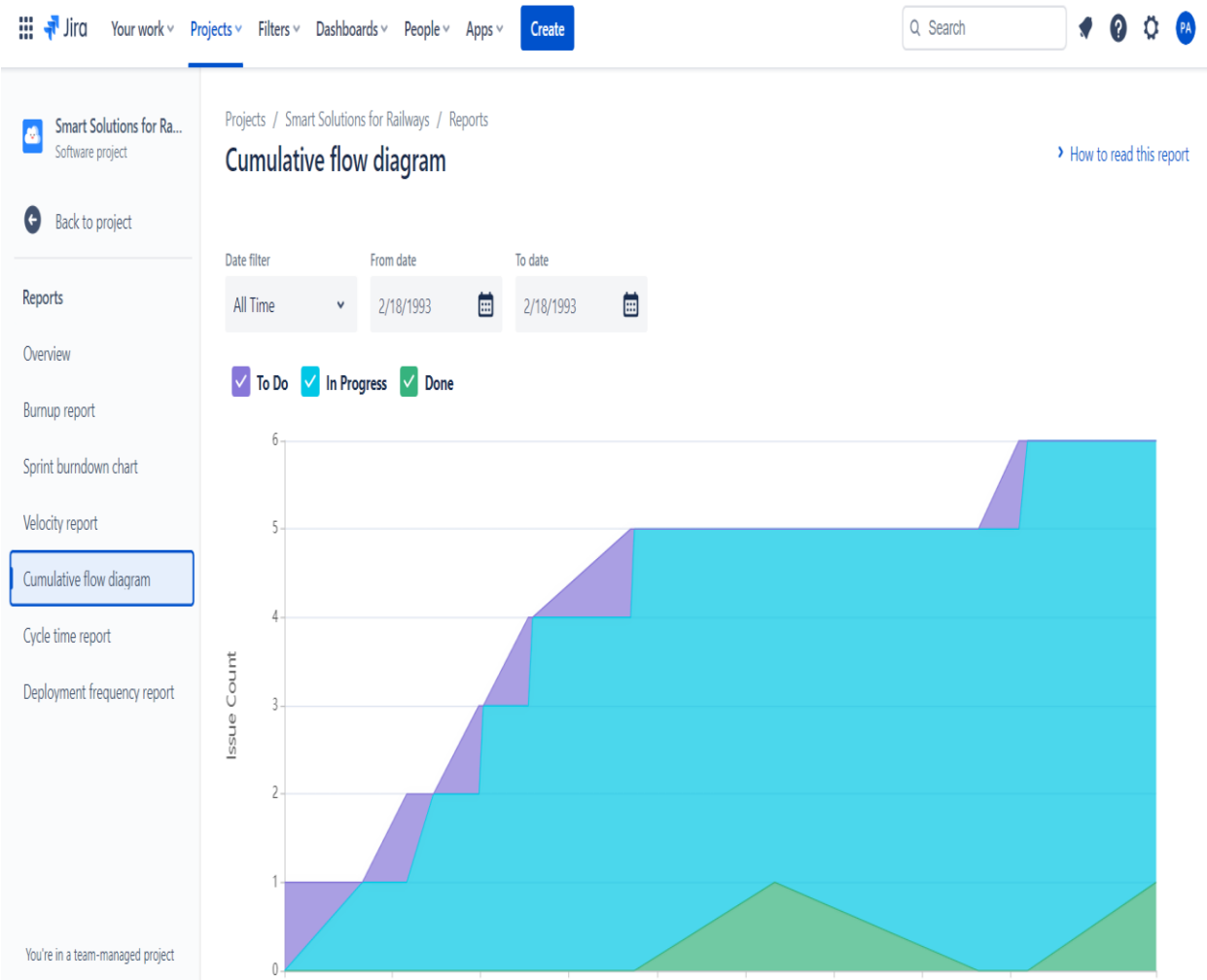
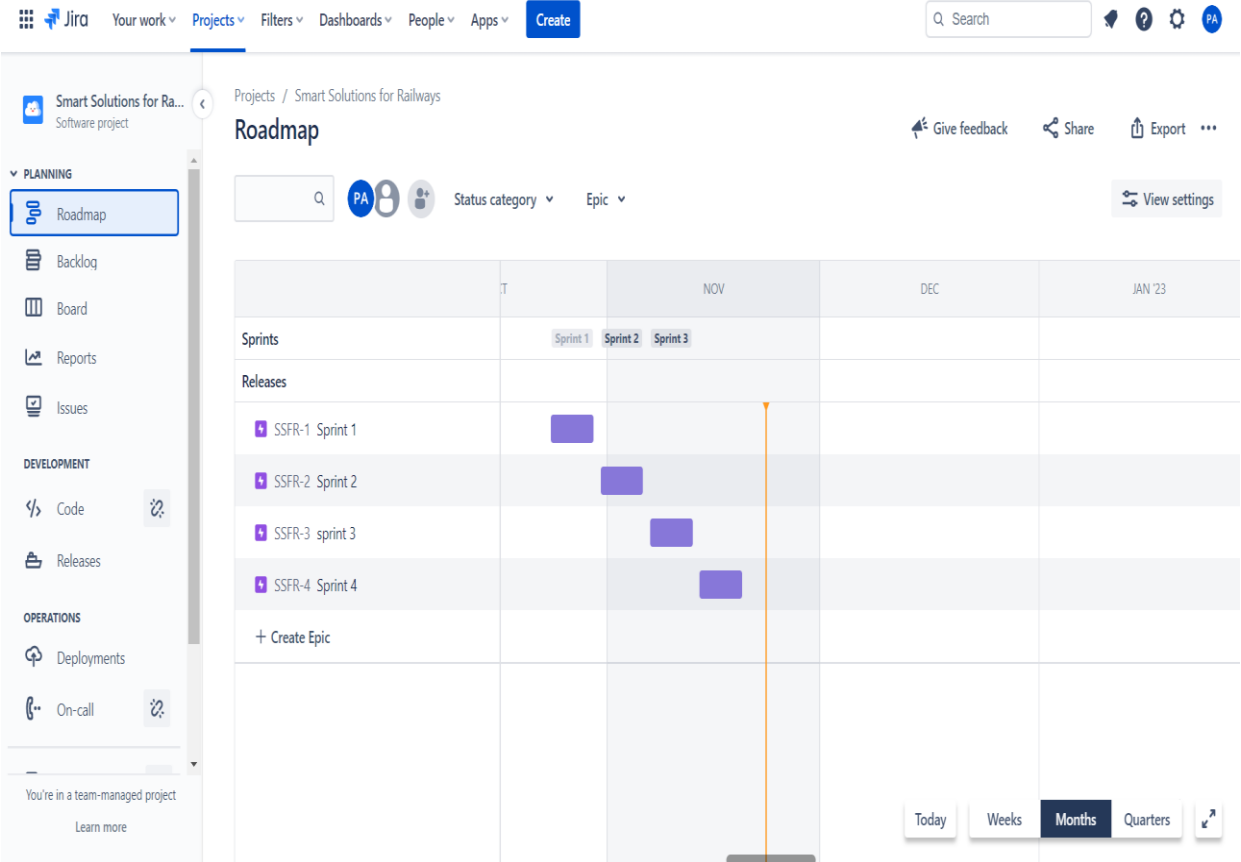
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	10	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	-	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	-	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	-	19 Nov 2022

6.2) Sprint delivery and schedule:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-1	As a user, I don't need to enter any credentials and login because it is for public use. (health safety)	1	Low	Ambiga,asmitha, abitha,Eugin,Sugirtha
	Dashboard		A home page will be available which helps to enter to next page	1	Medium	Ambiga,asmitha, abitha,Eugin,Sugirtha
Sprint-2	Help Screen		After the Home page ,it will direct you the next page which contains 3 options.	2	Medium	Ambiga,asmitha, abitha,Eugin,Sugirtha
Sprint-3	1 st option		User can send Alert message by SMS	2	Medium	Ambiga,asmitha, abitha,Eugin,Sugirtha
Sprint-4	2 <sup>ND</sup> OPTION AND 3 <sup>RD</sup> OPTION		User can send Alert message by Call and Exit option	2	High	Ambiga,asmitha, abitha,Eugin,Sugirtha

6.3) Reports from JIRA :





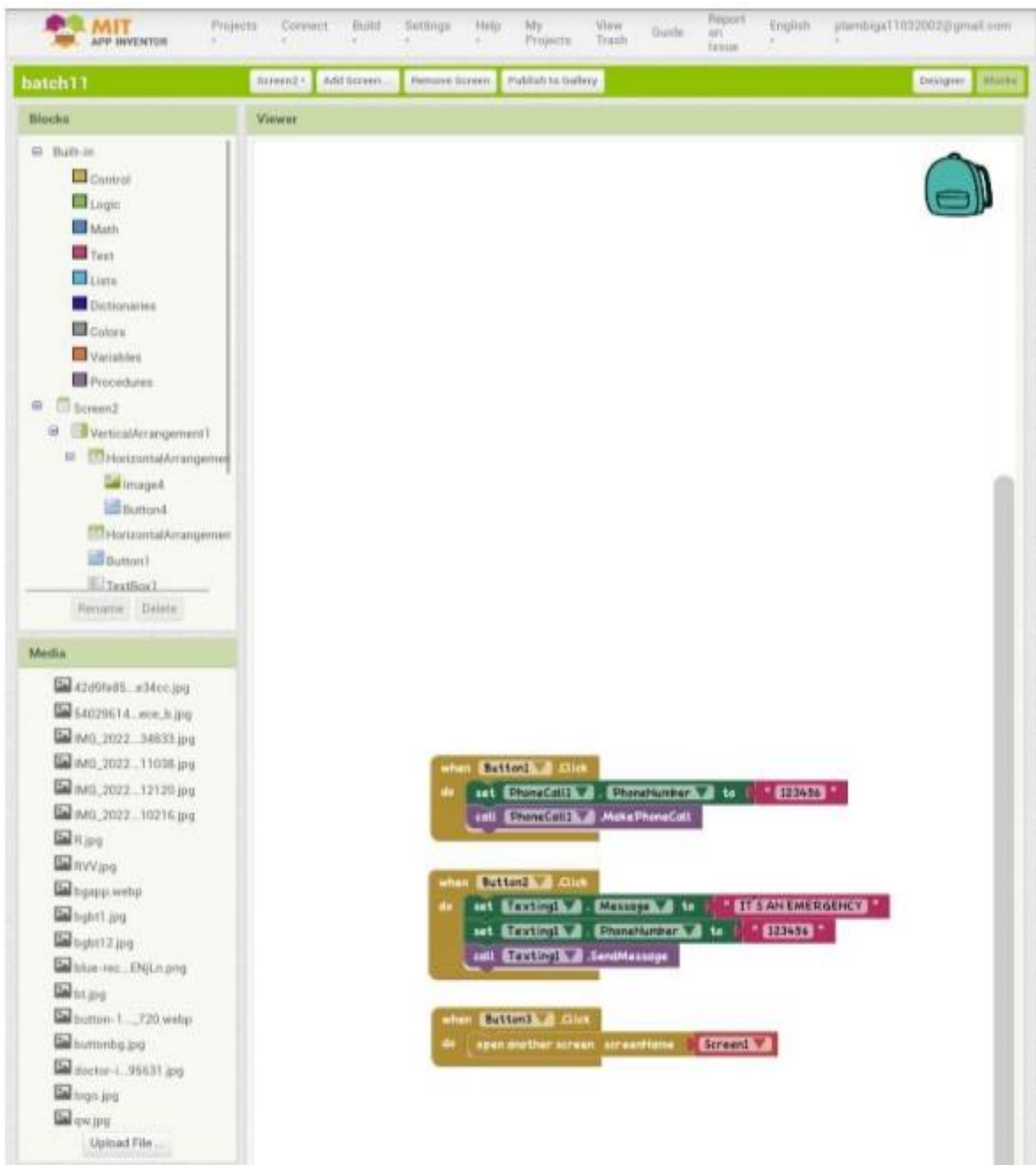
## 7) Coding & Solutioning:

### 7.1) Feature 1 :

As an User, in order to contact doctor ,I can use the “Call option” available in the App



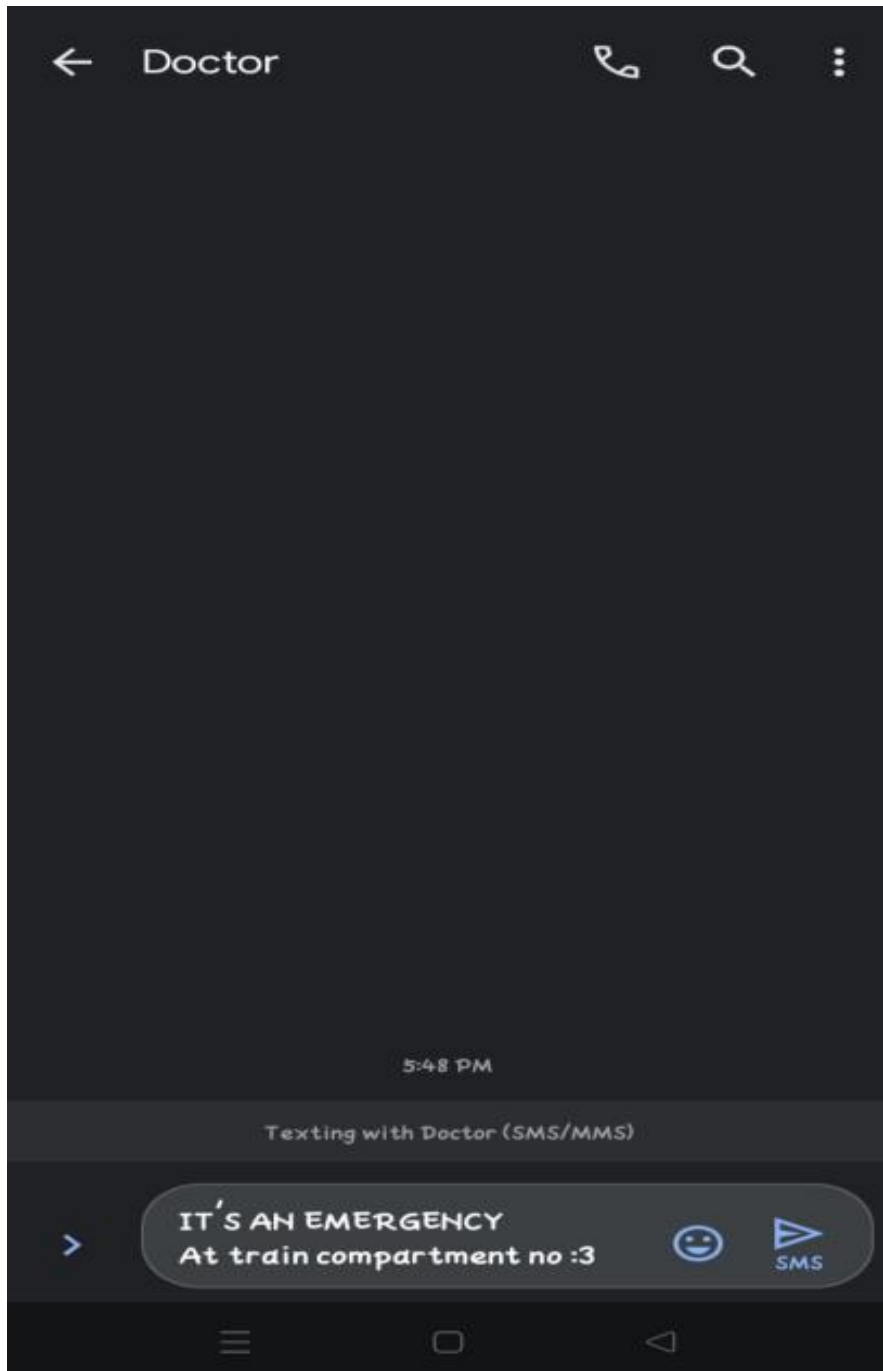
The Call will be directed to the doctor’s number which will be stored in the back end block.



## 7.2) Feature 2:

As an User, in order to contact doctor ,I can also use the “SEND SMS” option available in the App

People having headaches, cold and minor issues can use this option to communicate with doctor



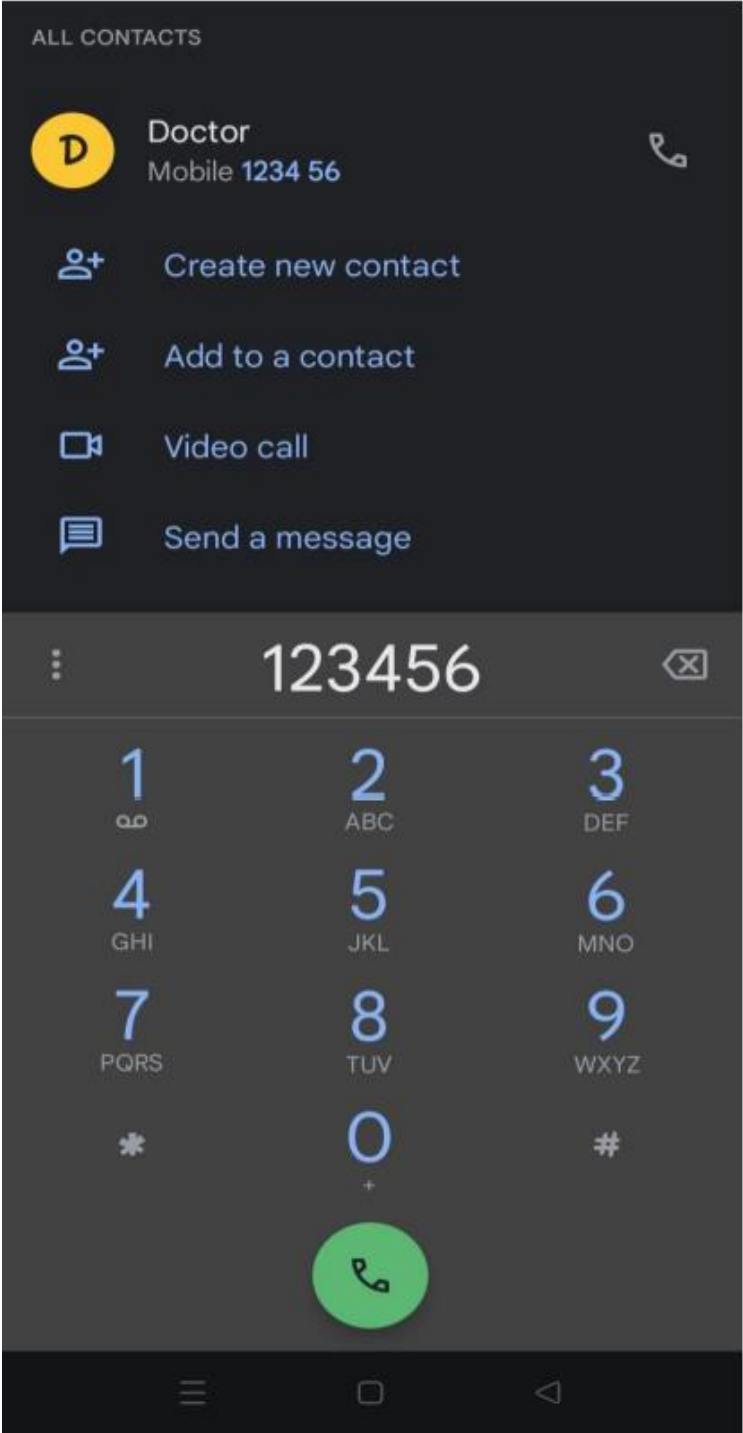
## **8) Testing:**

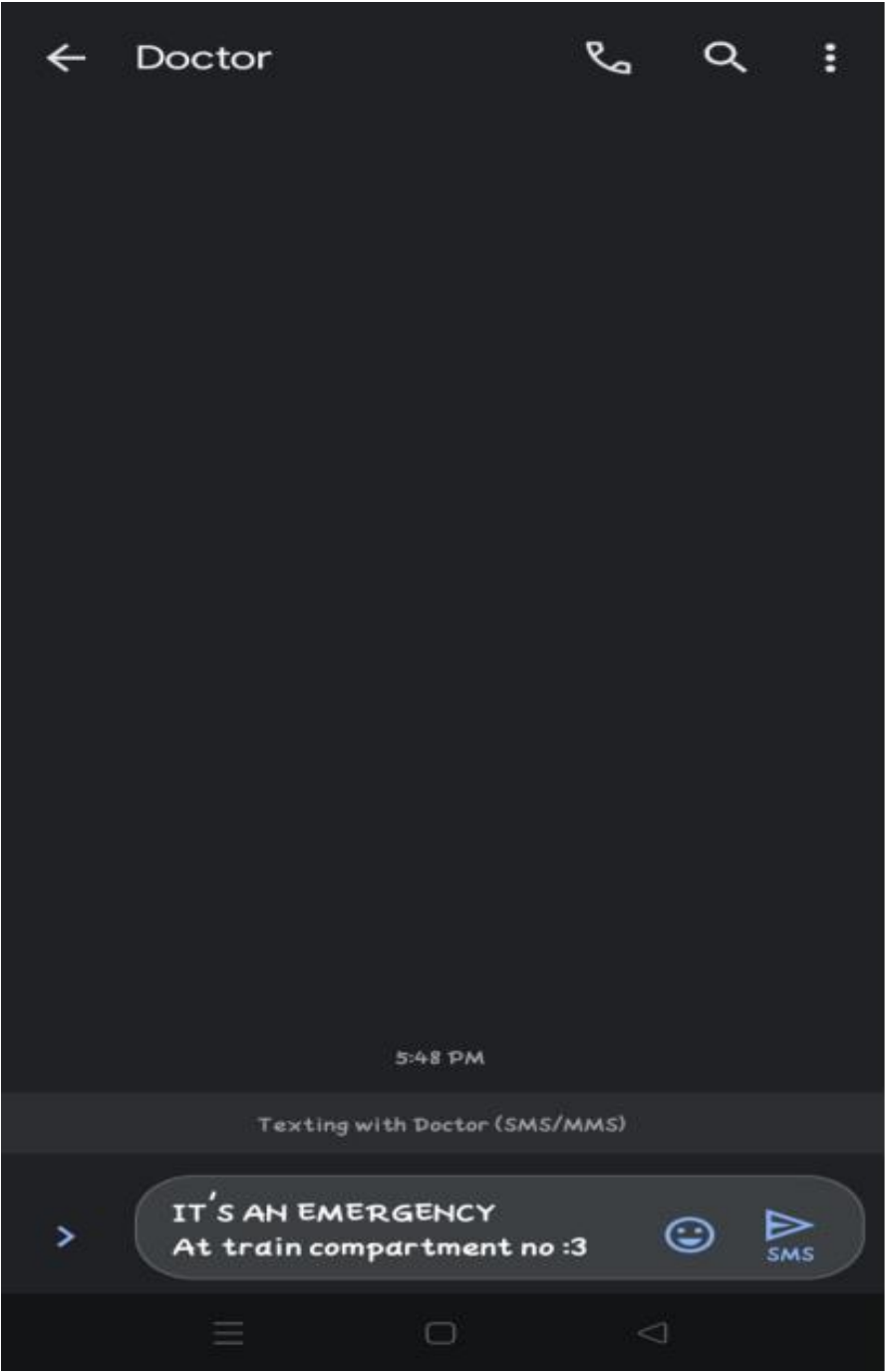
### **8.1) Test Cases & User Acceptance Testing:**

The App has been tested with the help of MIT app inventor









## **9) Results:**

### **9.1) Performance Metrics:**

The call and sending SMS to the doctor is working fast as we expected and result can be viewed by using MIT APP Inventor. We are minimizing the time required to communicate the doctor by storing the contact number of the doctor and making it easy for the user. The app is readily available in the IOT device which makes it easy to access and complexity is reduced.

## **10) Advantages and disadvantages:**

### **Advantages:**

- ❖ People can easily access it since it is available in the compartment already.
- ❖ In emergency situation, there is no need of waiting because communication is made easier.
- ❖ Generally ,People feel safe after using it.
- ❖ The no of people who will be saved by this system is more than the normal system used in the current railways.
- ❖ More Specialized doctors are allotted for this job at each train.
- ❖ In this project , calls and messages both are included
  - a) In case of high emergency call is used
  - b) In case of minor issues, messages are used

### **Disadvantage:**

- ❖ Children might play with this device, which cause unwanted requests to the doctor.
- ❖ While using for first time ,People may find it difficult
- ❖ Aged people might find it hard to use this device on their own.
- ❖ Implementation is costlier.

## 11) Conclusion:

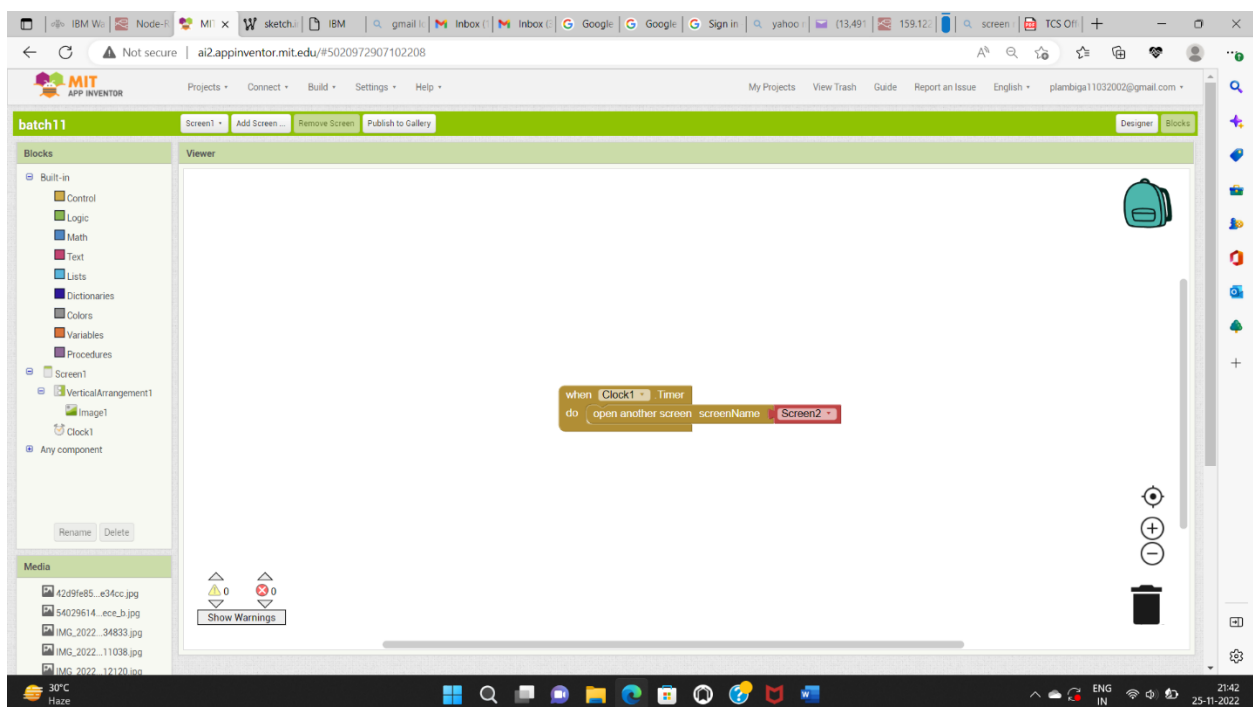
In today's Railway System ,Safety on health is questionable. Due to this problem, people may find this less safe to travel. To overcome this problem, we have included IOT based system which can help people to solve this problem and increase health Safety. It minimizes the problem that occur by waiting for the doctor to come from the nearest hospital. It also minimizes the time required to contact the doctor.

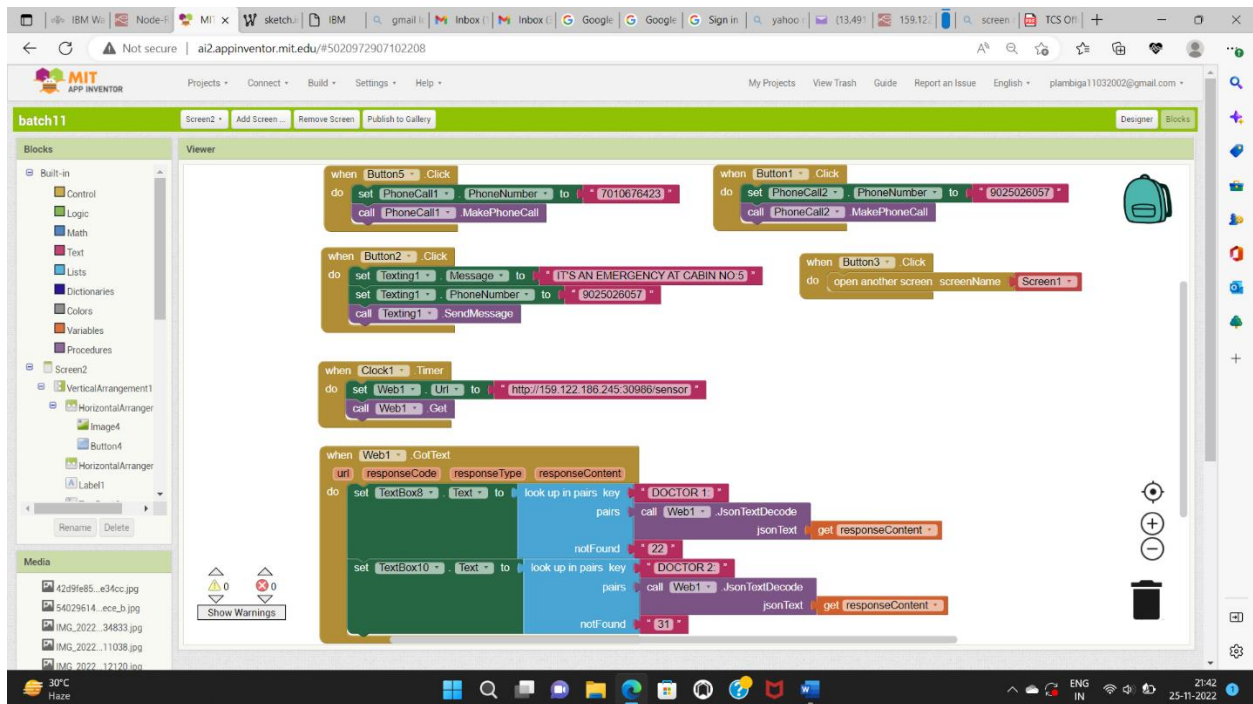
## 12) Future scope:

Further, we can make a new option on which we can able to video call the doctor and the availability of the doctor will be updated on that app screen based on cloud services.

In the future ,we can make a system, which also have some additional features on the area other than this health Safety.

## 13) Appendix: Source Code:





**GitHub link & Project Demo Link:**

**[IBM-EPBL/IBM-Project-30454-1660146909: Smart Solutions For Railways \(github.com\)](#)**