

Design and Development of an application for immediate local assistance

TROUBLESHOOT

A THESIS

Submitted by

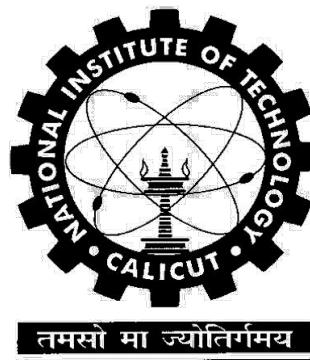
PREETI KUMARI - M170528CA

In partial fulfilment for the award of the Degree of

MASTER OF COMPUTER APPLICATIONS (MCA)

Under the guidance of

Dr VINOD PATHARI



तमसो मा ज्योतिर्गमय

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

National Institute of Technology Calicut NIT Campus PO, Calicut

Kerala, India - 673601

JUNE 2020

ACKNOWLEDGEMENT

I take this opportunity to express a deep sense of gratitude and indebtedness towards my guide Dr. Vinod Pathari, for providing excellent guidance, encouragement, and support throughout the project work. His priceless and meticulous supervision at every phase of work inspired me in innumerable ways.

I express my sincere thanks and gratitude to my additional mentor Mr. Ravi Sharma for guiding and helping me during the whole project. Her guidance helped me a lot in comprehending the things in a better way.

Lastly I will like to thanks family and friends specially Arathi and Manish for their support and motivations. Their help and supports are invaluable.

Date: 25th June 2020

Name: Preeti Kumari

Reg. No: M170528CA

DECLARATION

I hereby declare that the report entitled “*Design and Development of an application for immediate local assistance (TroubleShoot)*” submitted by me, for the partial fulfilment of the degree of Master of Computer Applications is a record of the work carried out by me under the supervision of **Dr. Vinod Pathari**.

I further declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature :

Place : Calicut

Date : 25/06/2020

Name : Preeti Kumari

Reg. No : M170528CA

CERTIFICATE

This is to certify that the report entitled "*Design and Development of an application for immediate local assistance (TroubleShoot)*" submitted by Ms. Preeti Kumari(M170528CA) to the National Institute of Technology Calicut towards partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications(MCA) is a bonafide record of the work carried out by her under my supervision and guidance.

Dr.Vinod Pathari
Department of Computer Science and Engineering.
NIT Calicut

Place : Calicut

Date : 25/06/2020

Signature of Head of the Department

Table Of Contents

1 Introduction	1
1.1.Purpose	1
1.2.Document Conventions	1
1.3.Product Scope	1
1.4.Future Scope	1
1.5.Definition	2
2.Overall Description	2
2.1.Product Perspective	2
2.2.Product Functions	2
2.3.User Classes and Characteristics	3
2.4.Operating Environment	3
2.5.Design and Implementation Constraints	3
2.6.User Documentation	3
2.7.Assumptions and Dependencies	3
2.8.Design Diagram	4
2.8.1.Class Diagram	4
2.8.2.Activity Diagram	4
2.8.3.Use - Case Diagram	5
2.8.4. ER-Diagram	5
2.8.5.Sequence Diagram	6
3.External Interface Requirements	6
3.1.Functional Requirements	6
3.2. Interfaces	8
3.2.1 FRONTEND Interface (Android Studio)	8
3.2.2 BACKEND Interface (FIREBASE)	12
3.3.Hardware Interfaces	12
3.4.Software Interfaces	12
4.System Features	13
4.1.Feeds	13
4.2.Ask Question	13

4.3.Answer	13
4.4.Notifications	13
4.5.Emergency	13
4.6.Read	13
4.7.Settings	13
5.Other Nonfunctional Requirements	14
5.1.Performance Requirements	14
5.2.Design Requirements	14
5.3.Operating Constraint	14
5.4.Security Requirements	14
6. CONCLUSION	15
7. REFERENCES	16

ABSTRACT

This report details the design and implementation of a software solution "Troubleshoot" intended to provide immediate support to individuals from other users in the same geographic location. It is very common for us to require assistance when we are in a new location or even stranded in a particular area. "Troubleshoot" helps such individuals to pose their queries and get quick assistance from the immediate neighbourhood. Detailed requirements, design, and implementation details of the above software is provided in this report.

LIST OF DESIGN DIAGRAMS

Figure Number	Figure	Page Number
1	Class Diagram	10
2	Flow Diagram	11
3	Use-Case Diagram	12
4	ER-Diagram	12
5	Sequence Diagram	13

1.Introduction

Troubleshoot is a network on which people can help each other, share their problems, send and receive messages, get different solutions etc. In order to gain an overview about the report, firstly, the purpose and scope of this document will be given, and then an overall description of Troubleshoot system is followed. In addition to these, system features such as selecting range; emergency help etc. are described deeply. After mentioning about the introduction of the software system, the specific requirements will be addressed for it. In the final part, functional and non-functional requirements will be addressed.

1.1.Purpose

The report is needed to evolve as the development of the software product processes. The purpose of this document is to give a complete description about how Troubleshoot network system can be developed. This document is to provide information about what the software product is to do to customers and establish an agreement between customers and suppliers and also become helpful for development. In addition to these, it provides a basis for validation and verification. The issues which are basically addressed are functionality such as posting problems, answering questions, external interfaces, performance, attributes and the design constraints of the system.

1.2.Document Conventions

Helper: Any user who is going to answer the question.

Seeker: Any User who is going to raise the question.

Authority: At the time of emergency the people connected to certain organisation (like: police, hospital, fire brigade) are responsible to give immediate response to the seeker.

1.3.Product Scope

The name of the software product is Troubleshoot. Troubleshoot is a helping network that helps people with their problem. The aim of Troubleshoot is to provide help to the users from the other user around them. The users of Troubleshoot can share their problems, get emergency help in an instance; answer and solve others problem, get connected with specialist, chatting with them for further help. People will be notified if anyone around them needs any type of help.

1.4.Future Scope

As we progress with this application we are thinking to add a chatting system so that if a seeker wants to get more information from the helper then he/she can request for further communication and if the helper accepts to communicate they can communicate via text

messages. If any user wants to disconnect they can immediately close the chat. And every person will use this application and its IOS version will be built.

1.5.Definition

- i) Range- From the seeker coordinates (latitude/longitude) to the radius ‘d’ given by seeker, all the available users will be the range for the feed.
- ii) Feed/Post- Seeker will select the range and they will post the issue/problem with image and its description.
- iii) Delete- The post/feed will be deleted by individual user but stories will automatically removed in 24 hours.

2.Overall Description

In this section, background information about what type of requirements the system should have will be provided briefly.

2.1.Product Perspective

Troubleshoot is an independent and world-wide helping portal. Every person can use it online without a fee. The Troubleshoot is not a part of a larger system, it is an independent system. People from different regions of the world can connect and exchange information and help each other. In order to control the contents of the problems, answers and comments done by the other people, Troubleshoot has also a control mechanism. People can deliver their complaints about any part of the Troubleshoot to the “Troubleshoot Administrators”. Then, “Troubleshoot Administrators” might take appropriate actions according to the situation.

2.2.Product Functions

After creating an account and starting to use the Troubleshoot, first thing he or she will make is choosing the area of interest like various fields (science, politics, environment etc.). The user will search the field according to its name and add it as a field of interest. After the completion of choice submission the user will be provided with the questions related to their field of interest that are asked by different users of the community. Any user can share his/her question or answer like whatever he is thinking, whatever he wants to know. . Other user can see his/her question and will be able to answer to it. Furthermore, a user can search for a question also which can be already asked by some other user. Also a user with the question can approve the answers given by another user. Seeker can send request for personal chat with the helper for further question. Seeker will also have an emergency network for urgent help which will be connected to authorize personal.

2.3.User Classes and Characteristics

Troubleshoot does not require any specific computer knowledge to use it except the developers and administrators of it. Standard users are thought to be from any age, any gender and from any nationality who can use just android application. On the other hand, administrators and potential developers need a high level of expertise to understand android development.

2.4.Operating Environment

Troubleshoot is an android application which can be used on any android system. Android device with *Jelly Bean! Android 4.3 or above* is required. RAM with 1GB or more. Google play services should be installed in the user's device. And GPS should be on.

2.5.Design and Implementation Constraints

Being a helping network android application, the software should ensure the safety of information given by the user and provide some privacy settings options to the user. Firstly, Troubleshoot provides people the right to choose the category of people who will be able to view their shared information. Some users may not desire the access of some people to their shared questions and information. If this is the case, users can set their privacy settings to prevent some people's access to their information. Secondly, Troubleshoot cannot sell the private information of users to someone else. However, if the user permits, an application can access to some information of the user.

2.6.User Documentation

There is no need for user documentation. All the functionality are in sequential way and every event is designed as simple as possible with proper notation. This application is so simple that a user who shares some common knowledge of any android application can use this Troubleshoot application.

2.7.Assumptions and Dependencies

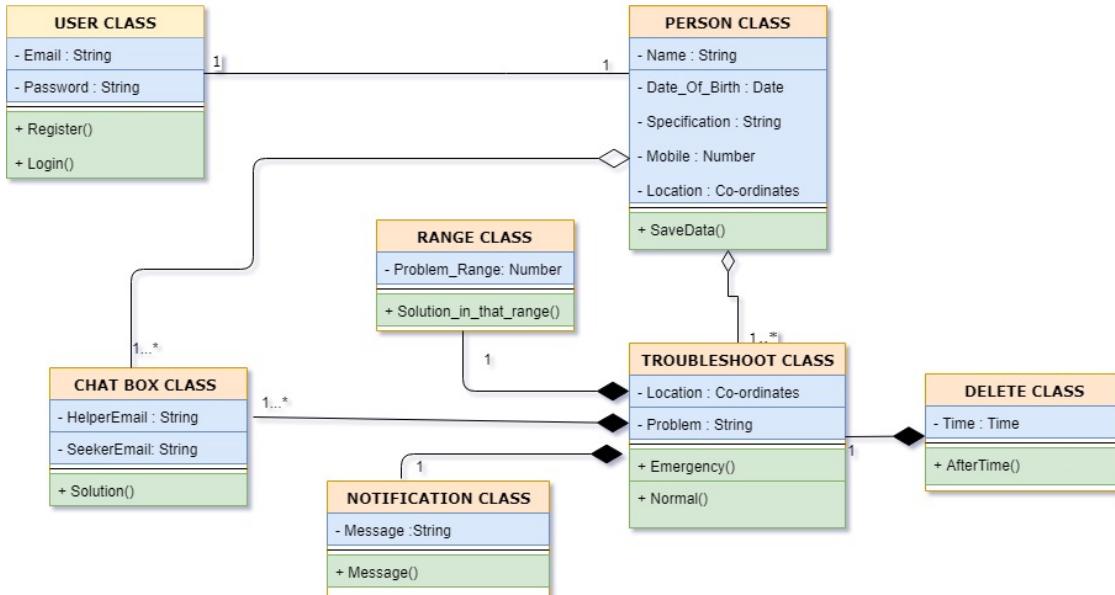
In this section, all normal and alternative flow of events are organized with the assumption that users or administrators have successfully reached their homepage by logging into the system. This assumption is made in order to describe specifications of the sub-features with better focusing.

2.8.Design Diagram

2.8.1.Class Diagram

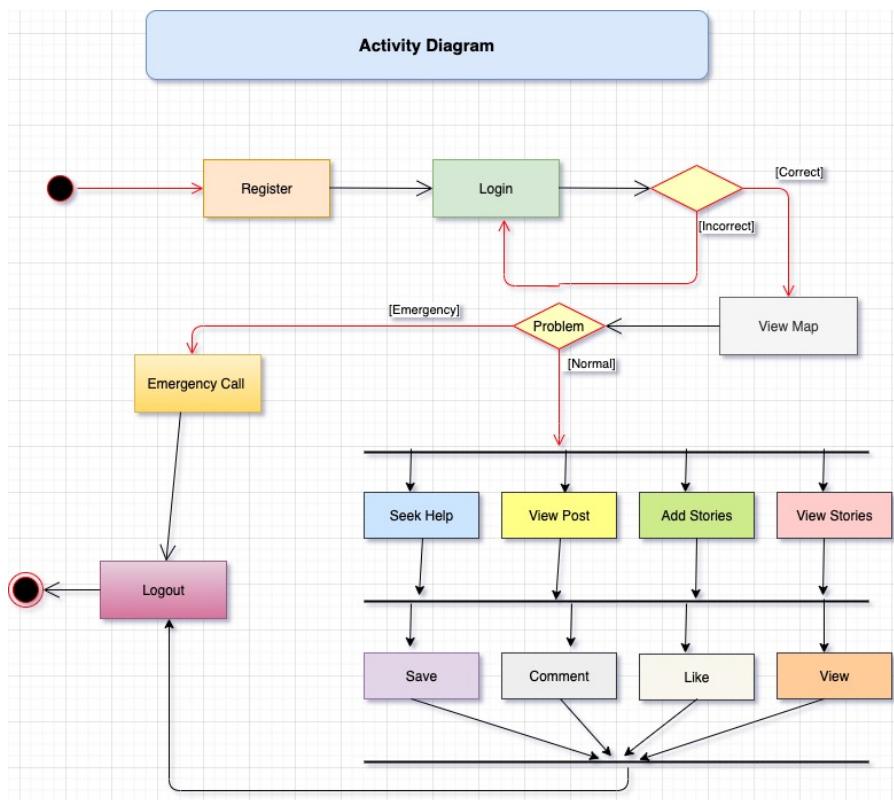
Figure No- 1

TROUBLESHOOT



2.8.2.Activity Diagram

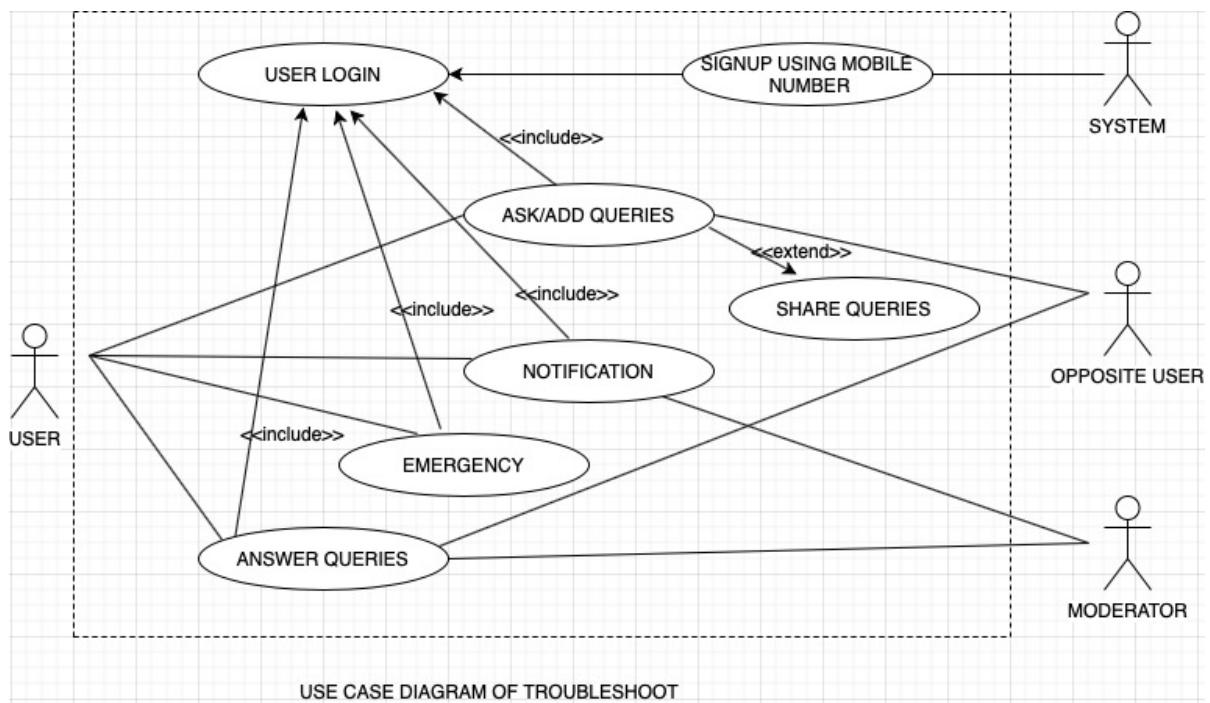
Figure No- 2



2.8.3. Use - Case Diagram

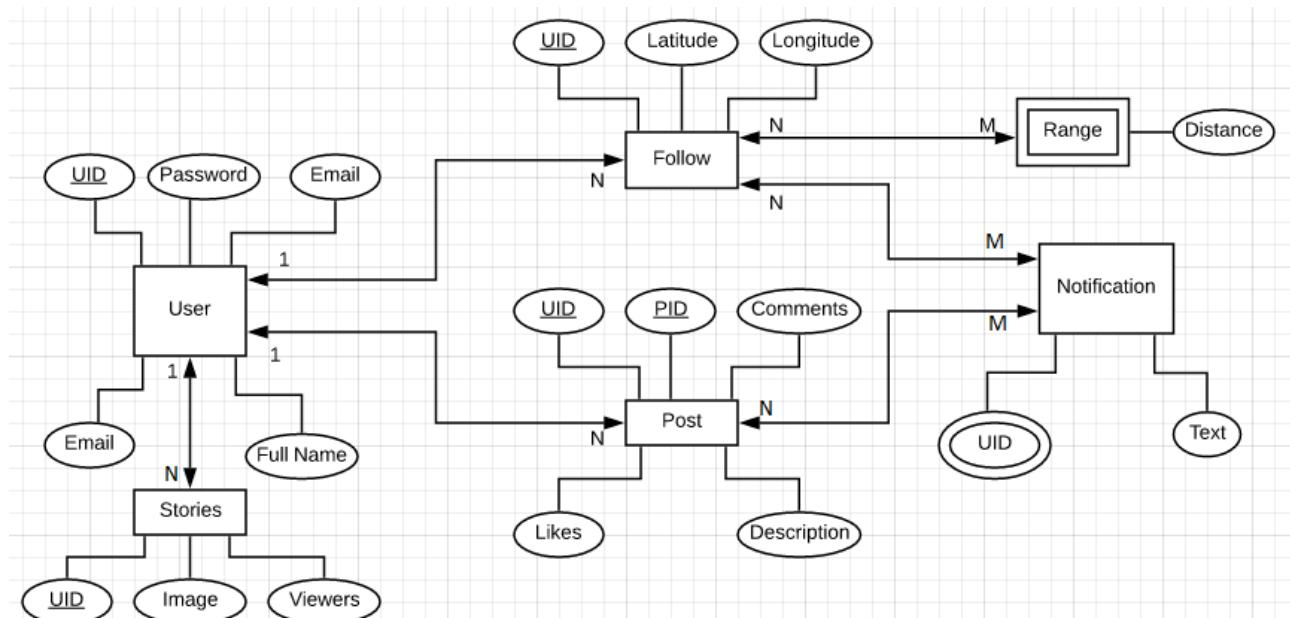
Here we have 4 actors in use-case diagram. User can add/ask queries, make emergency call, answer queries and even share and save queries. Whereas system authenticates the users and opposite user can ask/add queries and answer them where moderator helps to notify users notification and check on answer queries.

Figure No- 3



2.8.4. ER-Diagram

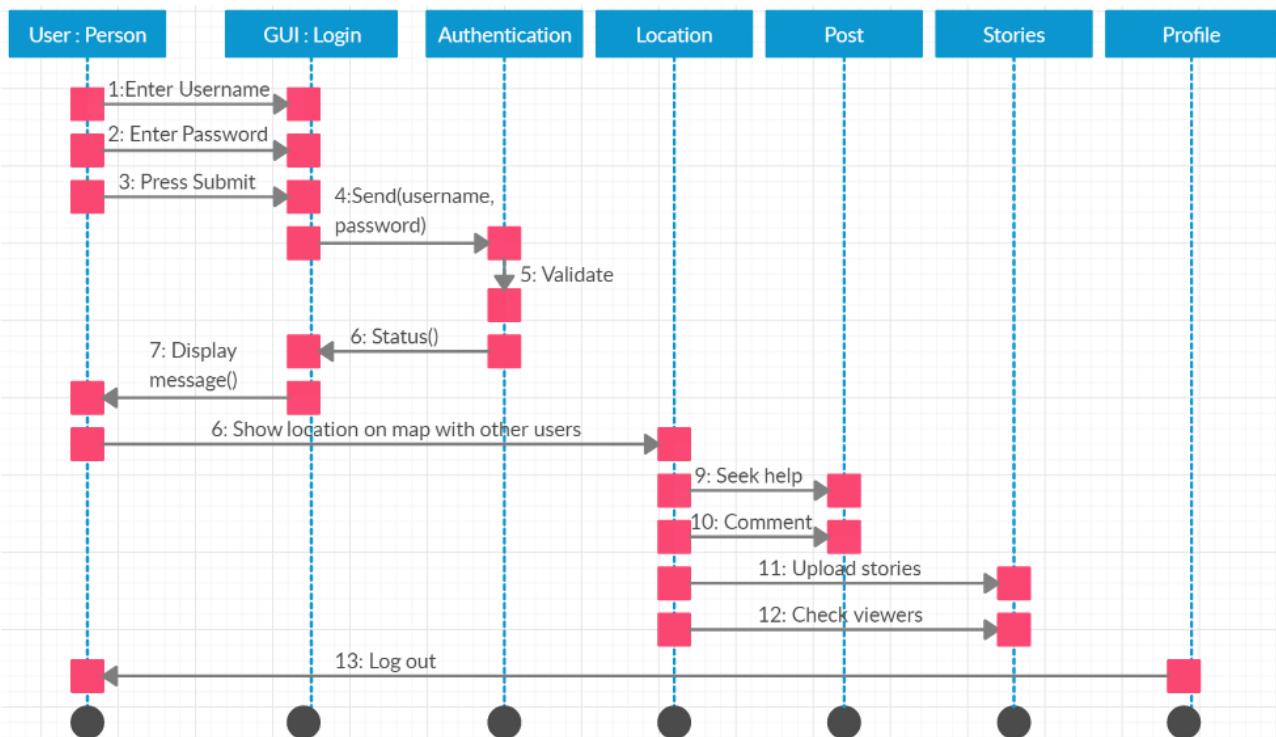
Figure No- 4



2.8.5.Sequence Diagram

Here, user will login to GUI if its authentication id and password are correct. Then we will go to the home page where user can see all available users. If it is incorrect, then it will again go to login page. After authentication user can post stories, feeds and check the profile. Users can comment on feed of others. User can logout which will then be directed to the login page.

Figure No- 5



3.External Interface Requirements

3.1.Functional Requirements

The requirements for the software system are as follows:

Functional requirement 1:

- Description: Login
- Input: Email and password are entered
- Processing: verifying the email
- Output: Login Successful

Functional requirement 2:

- Description: Register
- Input: Email, password, name, specialization and number are entered
- Processing: Verifying the email and phone number
- Output: Register Successful

Functional requirement 3:

- Description: Questioning phase
- Input: Questions asked must have a brief description
- Processing: verifying question and removing any restricted content
- Output: Question displays

Functional requirement 4:

- Description: Answer
- Input: Answering the question
- Processing: verifying the answer and removing the explicit content
- Output: answer submitted

Functional requirement 5:

- Description: Search
- Input: Search the specific topic or question
- Processing: searching the input in database
- Output: results shown

Functional requirement 6:

- Description: comment
- Input: Commenting the answer
- Processing: verifying the comment and removing the explicit content
- Output: comment submitted

Functional requirement 7:

- Description: Profile
- Input: Uploading the profile picture and writing bio

- Processing: Uploading the pic
- Output: Profile Updated

Functional requirement 8:

- Description: emergency
- Input: Specify the emergency authority
- Processing: Getting connected with specific authority
- Output: Receiving call or acknowledgment for the authorised personal

3.2. Interfaces

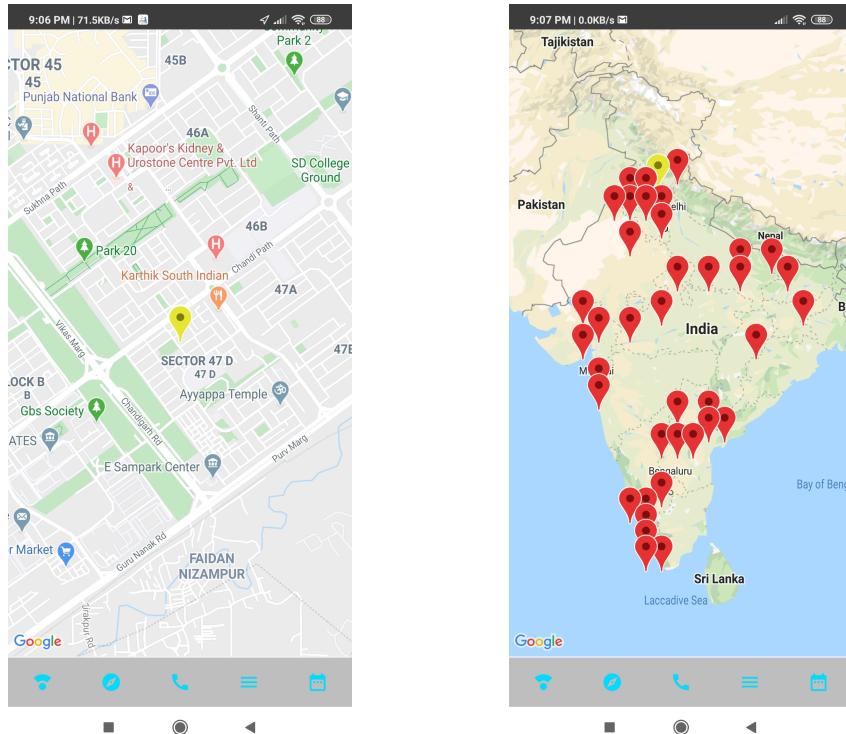
The customer user interface should be intuitive, such that all new users are able to complete their goals without any assistance.

3.2.1 FRONTEND Interface (Android Studio)

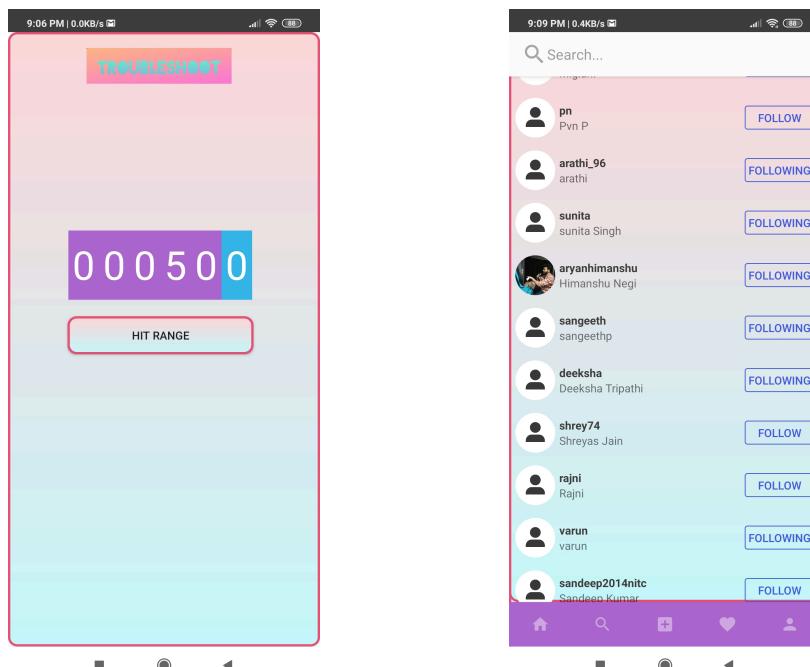
I) Here user will login or register for authentication.



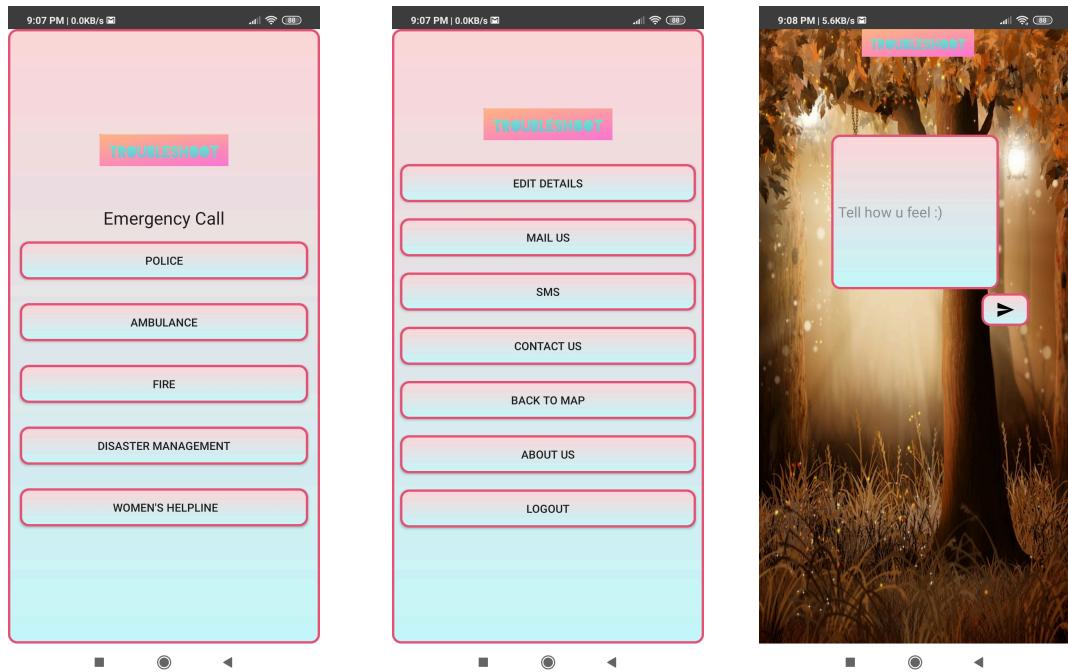
II) After login we will direct to the home/Map page when your current location will be shown in yellow and available users in red.



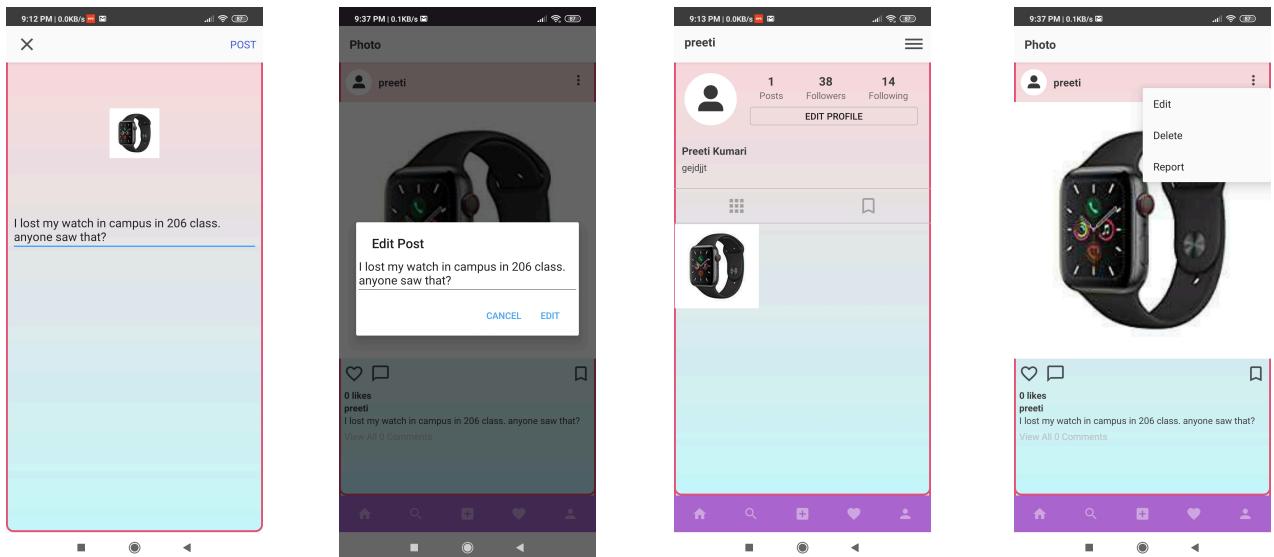
III) When we set the range we will direct to the main page where we post. In search, we can see all the users in range as 'following' and others those are out of range are as 'follow'.



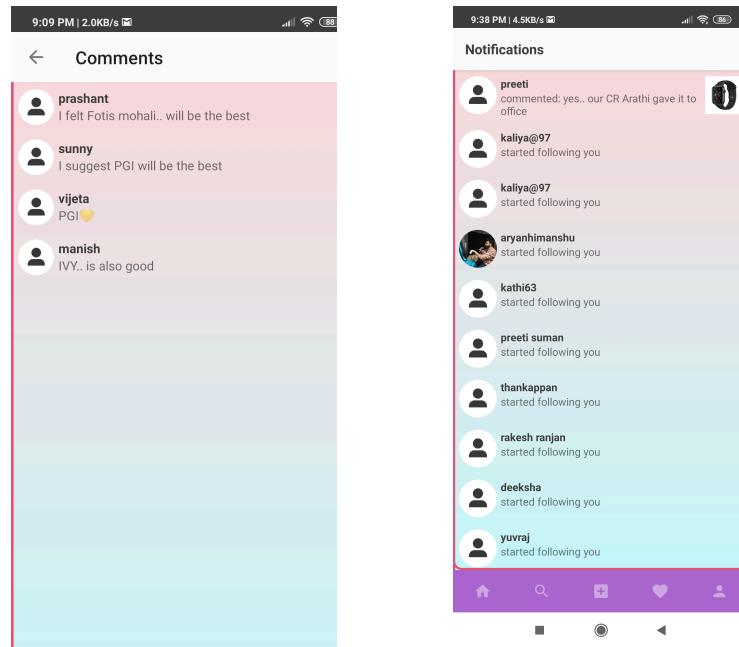
IV) For Emergency we can call directly from this page and we have option page for contact us via sms/call/email.



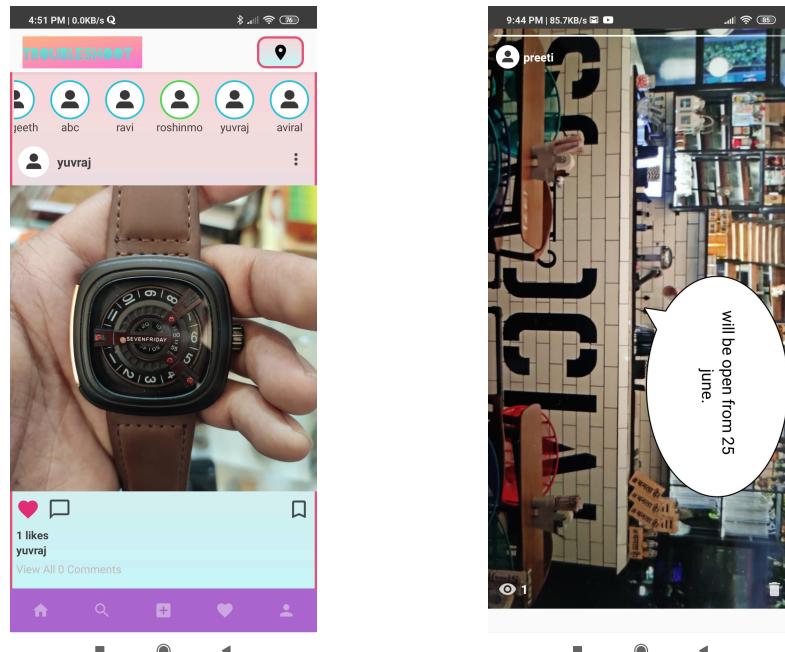
V) Seeker will post the problem for help or Suggestion with image and description. Later on user can edit or delete the post according to the need. User can edit his profile and add his profile picture. User can see his saved post.



VI) Helper users can add their suggestion or solution on comment box. And seeker will get the notifications.



VII) Users can add their stories for 24 hours where they can see viewers and have feature to delete anytime they want. The seen stories will be in green and others will be in blue.



3.2.2 BACKEND Interface (FIREBASE)

Here we are using Firebase in the backend. It is Google Cloud where we are storing our data. As its a Cloud so we are secure from the side of load of data and we are using JSON way for storing our data. JSON stores data in tree structure. Storing data in tree structure saves a lot of space and easy in searching and maintaining. In Firebase for security we are using SHA, as it change the password of users to hash map code and every user get one auto generated ID by which all the trees will be linked. Here is the table of the trees I have used in my application with details.

#	Tree Name	Details
I	Login Users	Authenticated users
II	Comment	List of comments in a post with respective users name
III	Follow	Latitude and longitude of every users
IV	Likes	List of all the likes on the post
V	Notification	If anyone comment and like post, follows will show notification
VI	Post	Description and details for the post
VII	Saves	Detail of post saved
VIII	Story	Detail of the Story Feed
IX	User Available	Users in the given range List
X	Uploads Images	Stores only Profile Image
XI	Story Images	Stores only Story Image
XII	Post Images	Stores only Post Image

3.3.Hardware Interfaces

This application works according to the location of the user so GPS is needed. Internet connectivity is mandatory.

3.4.Software Interfaces

The software interfaces are specific to the target helping network systems. XML and firebase gives frontend and backend software Interface.

4.System Features

In this section, all normal and alternative flow of events are organized with the assumption that users or administrators successfully reach their homepage by logging into the system. This assumption is made in order to describe specifications of the sub-features.

4.1.Feed

This feature will be used by seeker to select the range and they will post the issue/problem with image and its description and helper to give suggestion/solution.

4.2.Ask Question

There will be a help button where user will be directed to a new page where user can enter the question. If that given question by the user is present in our database it will show previous responses otherwise it will raise a new question with specific details in a generalized way and range will be set for the question.

4.3.Answer

Answer menu will provide user to answer to the various question available on the news feed for answering.

4.4.Notifications

The notification pane of the android interface will inform the user about the emergency activities of the user as well as the normal helps they want.

4.5.Emergency

If a seeker wants any type of emergency help that user can use the emergency help line. That will inform the responsible authority as well as the users nearby.

4.6.Read

All the questions where the current user is fitting in the range and also related to the concerned field of user interests. A search menu to search available users by username.

4.7.Settings

If user wants any kind of change in their account he/she will mail the authority with the particular request.

5.Other Nonfunctional Requirements

5.1.Performance Requirements

System shall be available from all over the world at all times. Being a helping network, any interruption in the sharing chain will cause people to give up on Troubleshoot; therefore it is essential that the system shall be available at all times. System shall not be affected from the large number of active users in the system become active. Being a worldwide network, assuming that half of the registered users are reaching to the website is a legitimate and necessary requirement. Its performance can degrade on system failure or no proper internet connectivity.

5.2.Design Requirements

Design of the system shall arrange the content size as compatible for android devices. Since Troubleshoot is based on sharing with other users, design of the system expects high level of mobile access.

5.3.Operating Constraint

- Android smartphone with sufficient amount of RAM.
- Proper internet connectivity should be there.

5.4.Security Requirements

As the account of each user is protected by a user id and password, any unauthenticated access to the account is not allowed. Also some privacy policies are also applicable to the user.

6. CONCLUSION

The software solution explained in this report has been devised to provide assistance to individuals from people who are near-by. While general purpose support sites and forums are available, there are not many applications focusing on providing local support. Thus the "Troubleshoot" software is a novel solution and has been developed using latest technologies. The interfaces are also intuitive to assist easy adoption.

We hope to popularise this android application and to work on the iOS version next.

7. REFERENCES

1. "How To Create XML in android studio ... - Developers." 29 Mar. 2018, <https://developer.android.com/guide/topics/ui/declaring-layout>. Accessed 18 May. 2020.
2. "How To build Google Map in android studio ... - Android - Google Maps." 29 Mar. 2018, https://www.tutorialspoint.com/android/android_google_maps.htm. Accessed 20 May. 2020.
3. Ramez Elmasri, Shamkant B. Navathe: Fundamentals of Database Systems sixth edition - 2016. Accessed 25 May. 2020. Pages[359-555, 841-848, 915-930]
4. "Java Android coding ... -Android Studio ."Java CompleteReference by Herbert Schildt , 9th Edition. Accessed 3 June. 2020. Pages[455-485, 751-755]
5. [https://www.tutorialspoint.com.firebaseio/index.htm](https://www.tutorialspoint.com/firebase/index.htm). Accessed 15 June. 2020.
6. "UML Use Case Diagram Tutorial - YouTube." 8 Feb. 2018, <https://www.youtube.com/watch?v=zid-MVo7M-E>. Accessed 16 June. 2020.
7. "Entity-Relationship Diagram Tutorial | ER Diagram ... - YouTube." 27 Nov. 2013, <https://www.youtube.com/watch?v=SDsJG-a4WAI>. Accessed 17 June. 2020.
8. Android Installation link-“<https://drive.google.com/file/d/1HXPrsbxM6Z8fP-hh8IX6b08-zsM6VEWR/view?usp=sharing>”. Accessed 14 June. 2020.