MIDDLE EAST TECHNICAL UNIVERSTIY

DEPARTMENT OF COMPUTER ENGINEERING

CENG 491 Senior Design Project I - II

2016-2017

Abdulkadir Dalga

Hasan Ali Duran

Yasin Berk Gültekin

Tuğca Eker

Table of Contents

1. INTRODUCTION 4
2. SYSTEM REQUIREMENT 4
2.a. Problem Definition 4
2.b. System Requirements and Overview 6
2.c. Project Process Plan and Assumptions 6
2.d. Use Case Diagrams and Explanations 7
3. SYSTEM DESIGN 10
3.a. Module Structure 10
3.b. Source Code Structure 12
3.c. Component Diagram 14
3.d. Deployment Diagram 15
4. Test 15
4.a. Test Plan and Test Scenarios 15
4.b. Testing Code Structure 23
4.c. Test Results 24
5.References 25
6. Appendices 25
6.a. Definitions, acronyms, and abbreviations 25

Table of Figures

Figure 1 - Use Case Diagrsam 7
Figure 2 - Use Case Diagram 9
Figure 3 - Component Diagram 14
Figure 4 - Deployment Diagram 15

1. INTRODUCTION

This document is basic complete voucher of Search & Rescue mobile and web application that includes three main parts. These parts are system requirements, system design and testing.

Our Project Team Structure:

1. Ebru Aydın Göl- Project Advisor
2. Barış Nasır- Project Assistant
3. Yasin Berk Gültekin- Developer
4. Abdulkadir Dalga – Developer
5. Tuğca Eker- Developer
6. Hasan Ali Duran – Developer

2. SYSTEM REQUIREMENT

2.a. Problem Definition

In this project, a system that will be used in Search & Rescue operations will be designed and its software components will be implemented. Main components of the system are:

- Mission Planning and Coordination Center
- Rescue Team Member Computers
- Rescue Team Member Smart Glasses
- Unmanned Reconnaissance Vehicle(Quadcopter)

S&R System is used for planning and executing missions that are aimed to find and rescue people who are lost or injured in the field. It consists of some subsystems which are used in mission planning, tracking and execution of the rescue missions. A rescue mission is planned in computer at Mission Planning and Coordination Center. During planning phase, last known position and estimated position of lost person, digital maps of field, rescue team's information etc.. are used as an input. Mission will be planned in Planning & Coordination Center by using a computer having internet access and after planning has been completed,

necessary information will be loaded to devices that is going to be used by rescue team. When mission starts, everyone in the rescue team will be able to see their planned route on the mission computers and on the smart glasses. Also, they will be able to see their position and everyone else's position on their devices. Information on the smart glasses will be presented using augmented reality technology. Information on the computers and smart phones will be presented in two ways; by augmented reality technology over device's camera view and by placing information on digital maps. 5 If needed, rescue team will be able to use quadcopter for reconnaissance and the video taken by camera of quadcopter will be transferred to smart glass and/or mission computers. Everyone in the rescue team will be able to contact to each other by text messages or by talking using VoIP technology. They will also be able to transfer their device's video streams to each other's when they have been asked. Everyone in the rescue team will be wearing pulse sensors. Everyone's health information will be displayed on mission computers and planning center by using these pulse sensors' information. This information will also be shared among team members' computers. Also, the actions (walking, sitting, running, etc..) of members during the mission will be shown on the mission computers. During the mission, execution of the rescue operation can be monitored by the mission center. All the positions of rescue members, health states and their actions will be shown on the Planning Software using Geographical Information System. If needed, a member will be requested to transfer his video image to the center and watched from there. If a member finds the lost person from a distance, he will be able to measure the distance with laser range finder and compute his geographic coordinate. Then he is going to be able to mark his position and inform the other members and planning center. When the lost person is found during the mission, rescue team will put a pulse sensor to the person and person's health status will also be shown in all mission computers and in mission planning center up to rescue operation completed.

2.b. System Requirements and Overview

Project Search&Rescue is web and mobile application. Search&Rescue has two main parts. These parts are Mission Planning and Coordination Center and Rescue Team Member Mission Devices. For Mission Planning and Coordination Center there will be a web application which provides creating, tracking, and coordinating the missions. For Rescue Team Member Mission Devices, there will be an android application that provides tracking other rescue team members, communicating with each other's and coordination center and getting important data from external sensors such as pulse and muscle motions sensor. Since the project has an augmented reality part, also the camera is used for getting the real-world data to be augmented.

2.c. Project Process Plan and Assumptions

We plan to finish this project by June 2017. We divided our schedule to two main parts according to school semesters. In the first semester, we worked to create Mission Planning and Coordination Center which includes back-end and front-end and main part of the Android application that communicates Mission Planning and Coordination Center. We created object and database models. In the second semester, we plan to apply video stream and VoIP to android application to communicate with members and center. Also, Augmented Reality will be added on Android application. We held weekly meetings with team members to discuss current situation of the project. We held weekly meetings with our assistant to report our progress. Also, once in every week we held meetings with our supervisor to obtain solutions for our problems in the project.

Also, Aselsan wants us digital image transfer over Smart Glass to computer and server. They want filtering mechanism on augmented reality according to the emergency and secure

conditions of rescue members. We will add these functionalities on Search&Rescue application.

2.d. Use Case Diagrams and Explanations

We have two actors which are Coordinator and Team Member. Their use cases are mentioned below.

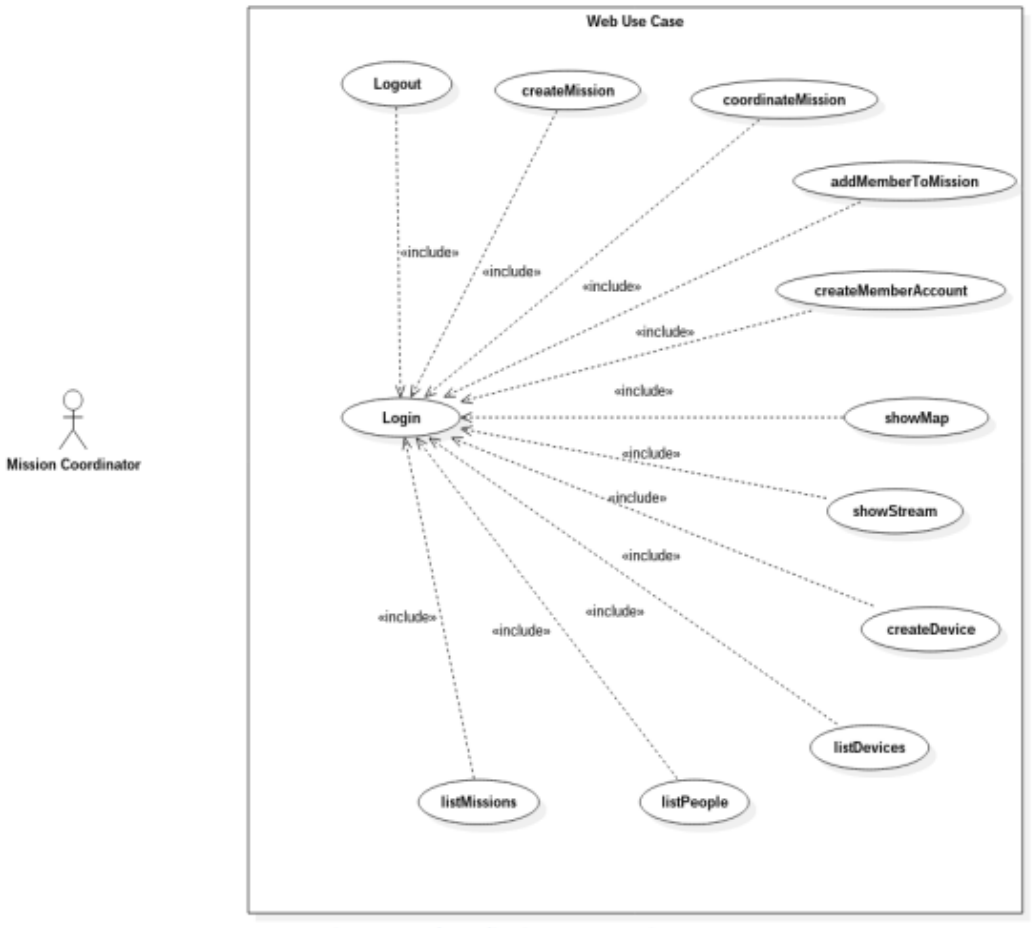

Figure 1 - Use Case Diagram

When actor is Mission Coordinator:

- Login: Coordinator will be logged in the system before executing any operation
- Create Mission: It creates a search and rescue operation

● Coordinate Mission: It coordinates the both mission and team members.
● AddMemberToMission: Coordinator can add team member to current mission
● createMemberAccount: Coordinator can create a new team member model on database.
● showMap: Coordinator can get all team member locations and show them on real search area map. Figure 1: Web Application Use Case Diagram 9
● ShowStream: Coordinator can play current video streams on web site
● createDevice: Coordinator can create a new device model on database.
● ListDevices: Coordinator can list all devices to use them in the mission
● listPeople: Coordinator can list all persons to select mission members
● listMissions: Coordinator can list all missions and mission details for current mission
● Logout: Coordinator can logout

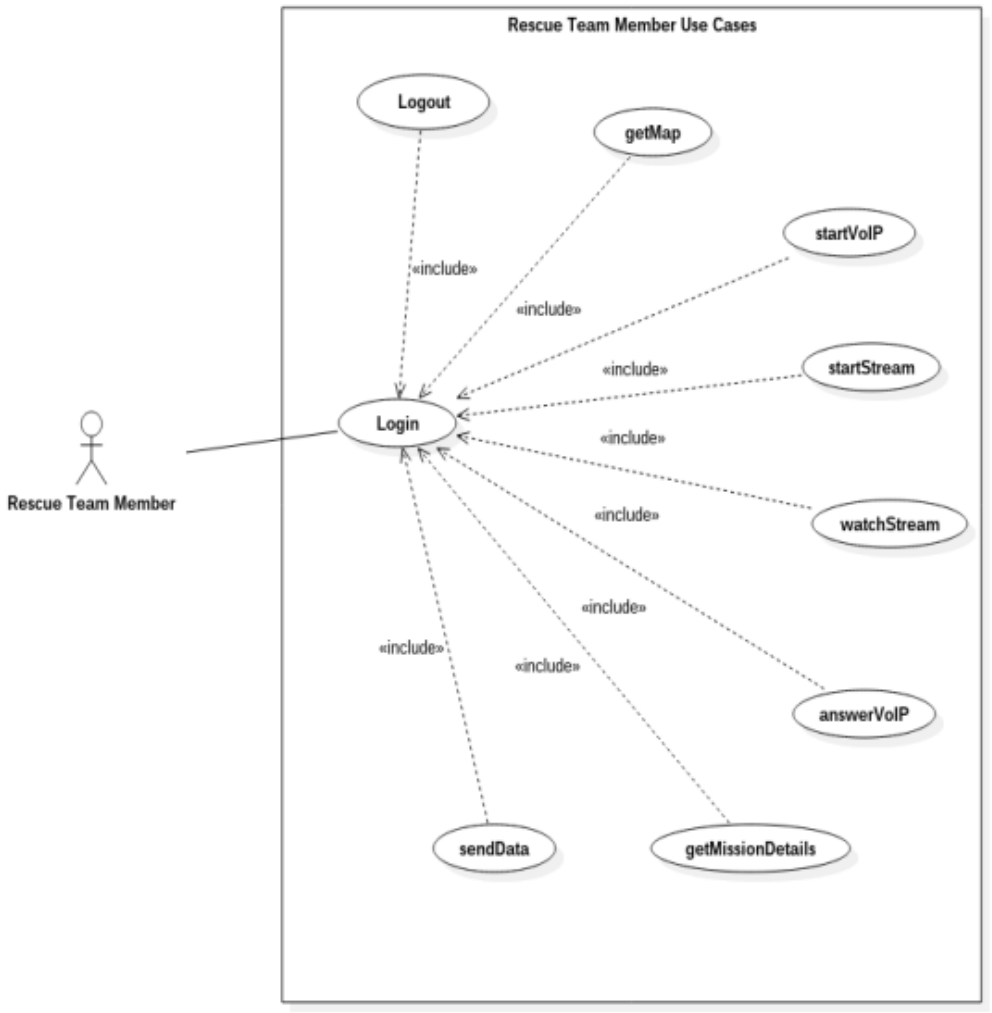

Figure 2 - Use Case Diagram

When actor is Team Member:

● Login: Team members will be logged in the system by using their android devices

● Logout: Team members can logout

● getMap: Team members can see other team member's positions on the map view

● startVoIP: Any team member can start audio conversation

● startStream: Any team member can start living video stream

● watchStream: When someone is streaming any team, member can watch this stream

● answerVoIP: When there is a started conservation user can join

● getMissionDetails: Any team member can see mission details

● sendData: Team members can send their information's to center

3. SYSTEM DESIGN

3.a. Module Structure

Our System has five main subsystems. These main systems are database server, backend server, mission coordination center (web application), sensors and team member android application.

A database server is a computer program that provides database services to other computer programs or to computers, as defined by the client–server model. Database server needs a database management system therefore database server system has one component that is MySQL as management system. MySQL is an open source relational database management system. It is used for arranging storing data and retrieving data from stored place.

A backend server system is responsible for communication between database and web. It has two main components first component is rest server. Rest is the underlying architectural principle of the web. The system about the web is the fact that clients (browsers) and servers can interact in complex ways without the client knowing anything beforehand about the server and the resources it hosts. The second main component of the backend server is that hibernate. Hibernate ORM (Hibernate in short) is an object-relational mapping tool for the Java programming language. It provides a framework for mapping an object-oriented domain model to a relational database. Hibernate solves object-relational impedance

mismatch problems by replacing direct, persistent database accesses with high-level object handling functions. Hibernates primary feature is mapping from Java classes to database tables, and mapping from Java data types to SQL data types. Hibernate also provides data query and retrieval facilities. It generates SQL calls and relieves the developer from the manual handling and object conversion of the result set. Hibernate provides an SQL inspired language called Hibernate Query Language (HQL) that allows SQL-like queries to be written against Hibernate's data objects. Criteria Queries are provided as an object-oriented alternative to HQL. Criteria Query is used to modify the objects and provide the restriction for the objects. HQL (Hibernate Query Language) is the object-oriented version of SQL. It generates database independent queries so that there is no need to write database-specific queries. Without this capability, changing the database would require individual SQL queries to be changed as well, leading to maintenance issues. Under the lights of explanations above the rest server and Hibernate communication is provided by HQL.

Third main system of project is that mission coordination center. This mission coordination center make communication to backend server over rest calls. These calls mean basically sending request to rest server and is taking response from server and database association. This system has three main components these components are authentication, GIS, and AngularJS. These three part connects each other over GUI. The first component is authentication. Authentication is a process that ensures and confirms a user's identity. Authentication works as pre-evolution step of rest server. Authentication begins when a user tries to access information. First, the user must prove his access rights and identity. When logging into a computer, users commonly enter usernames and passwords for authentication purposes. This login combination, which must be assigned to each user, authenticates access. Second component of the mission coordination center is that geographic information system. A geographic information

system (GIS) is a computer system for capturing, storing, checking, and displaying data related to positions on Earth's surface. GIS can show many kinds of data on one map. This enables people to see more easily, analyze, and understand patterns and relationships. The last component of the system is that AngularJS. AngularJS is a structural framework for dynamic web apps. It lets you use HTML as your template language and lets you extend HTML's syntax to express your application's components clearly and succinctly. AngularJS's data binding and dependency injection eliminate much of the code you would otherwise have to write. And it all happens within the browser, making it an ideal partner with any server technology.

Fourth main part of the project is sensors. These sensors provide user and server to data. Sensors includes Myo Armband, SmartGlass and Heart Pulse sensor. These sensors create communication to android devices over Bluetooth.

The last and main part of the project is Android application. This application has Geographic information system and Authentication again. It also includes live stream and voice over ip (VoIP). Live stream component connected with camera VoIP component connected with microphone. The system also uses GPS component.

3.b. Source Code Structure

Our implementations are under 5 main repository.

- https://gitlab.ceng.metu.edu.tr/iTech/Web_App_Interface

Link above includes our web application that is used mission coordination center.

- https://gitlab.ceng.metu.edu.tr/iTech/JerseyTest

Link above includes rest server of the project. That also contains hibernate and database codes.

- https://gitlab.ceng.metu.edu.tr/iTech/multiRTC

Link above includes video stream part of the project.

- https://gitlab.ceng.metu.edu.tr/iTech/AndroidRestClient

Link above includes android application of our project. This also contains augmented reality part, VoIP part, GPS , interface and sensor communications parts.

- https://gitlab.ceng.metu.edu.tr/iTech/ReconApp

Link above includes implementation of smart glass.

3.c. Component Diagram

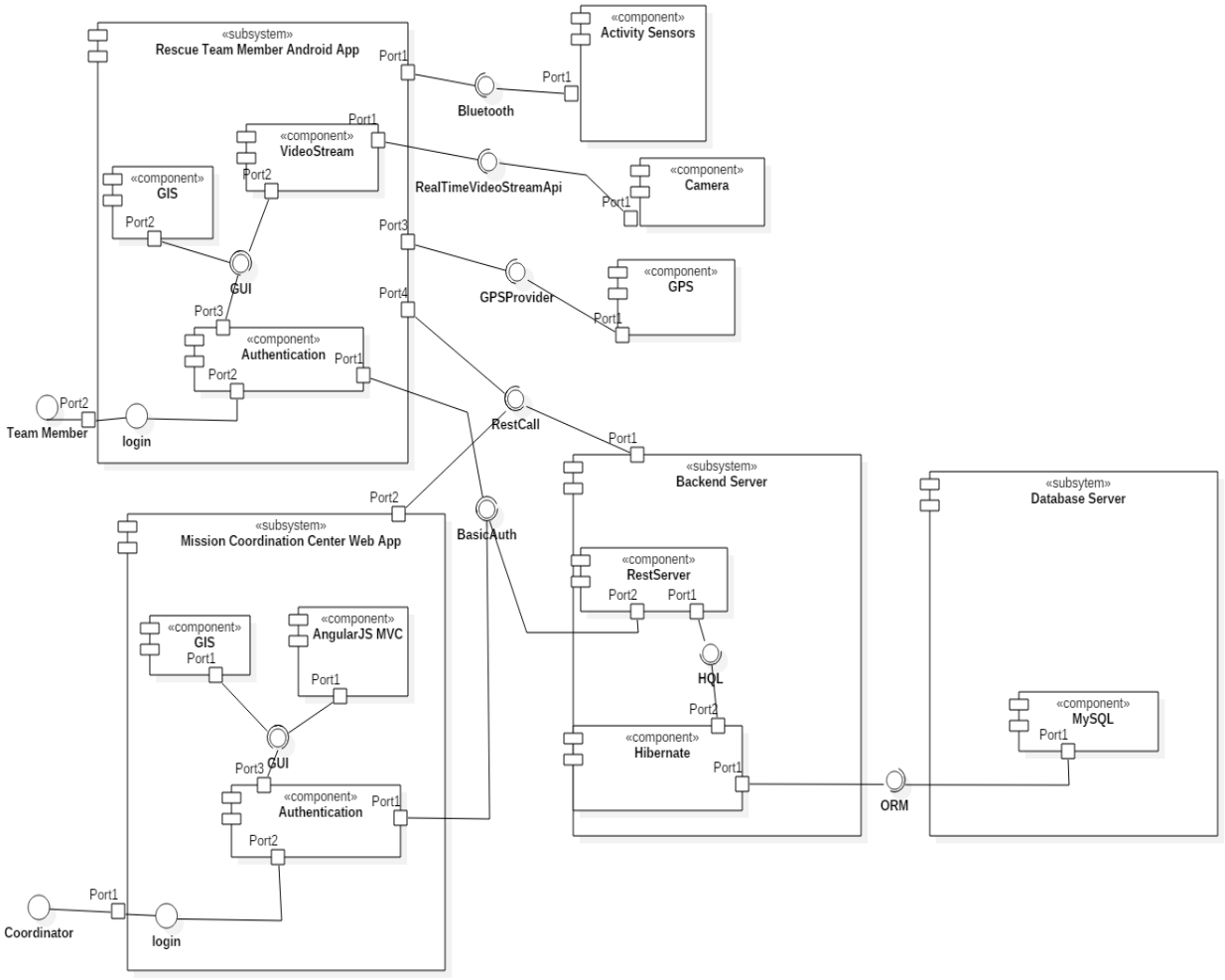

Figure 3 - Component Diagram

3.d. Deployment Diagram

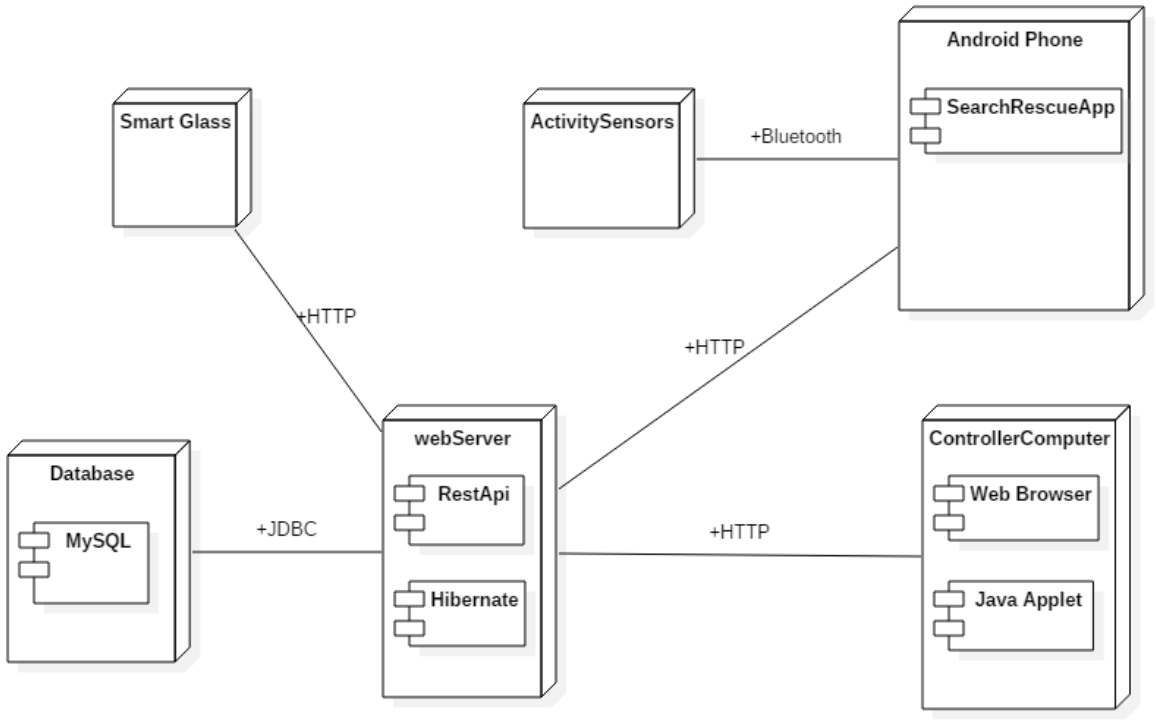

Figure 4 - Deployment Diagram

4. Test

4.a. Test Plan and Test Scenarios

The purpose of this section is to provide the test cases of Search&Rescue project. It defines the objective scenario and expected outcomes for each test cases. This section also describes the specific items to be tested, and general test approach is described with the pass/fail criteria, and test deliverables and test items.

First, the test plan must test all components of the project. Therefore, our test plan is based on testing each component of

project and functions of them. Our project has 5 main components. These components are Database, Rest Server, Android application, Web application and sensors. Some of components are work as a group so if we test functions of master component that uses slave component it also means we have tested slave component. For example, Rest Server uses Database continuously, when any rest all activate rest server it connects database and get data from database or post data on it. As a result, our test plan based on testing main components and testing main functions of these components by comparison result that supposed to be.

Test Scenario	**Login**
Test Case ID	TC1
Inputs	Username and Password
Expected Outcomes	Given username and password authenticate the system if system confirms it then system allow user to use main application.
Environmental Needs	Server and Database must properly run
Included Test Cases	-

Test Case Scenario	**Create Mission**
Test Case ID	TC2
Inputs	Mission details info
Expected Outcomes	System take mission details from user and create new mission and post it on database.
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Coordinate Mission**
Test Case ID	TC3
Inputs	Mission Detail info
Expected Outcomes	System take mission details from user and update mission according to the new data and post it on database.
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Add Member to Mission**
Test Case ID	TC4
Inputs	Team member information and mission id
Expected Outcomes	System take team member details from user and put member to existing mission that reached by id and post it on database.
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Create Member Account**
Test Case ID	TC5
Inputs	Team member information
Expected Outcomes	System take team member details from user and create new team member according to user data and post it on database
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Add Device to Mission**
Test Case ID	TC6
Inputs	Device information and mission id
Expected Outcomes	System take device details from user and put member to existing mission that reached by id and post it on database.
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Showing map**
Test Case ID	TC7
Inputs	-
Expected Outcomes	System must show all team members and mission area on GIS map
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Create Device**
Test Case ID	TC8
Inputs	Device information's
Expected Outcomes	System take device details from user and create new device according to data and post it on database
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**List Devices**
Test Case ID	TC9
Inputs	-
Expected Outcomes	System should show all devices and their attributes these are already posted on database
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**List Missions**
Test Case ID	TC10
Inputs	-
Expected Outcomes	System should show all missions that are already posted on database
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Logout**
Test Case ID	TC11
Inputs	-
Expected Outcomes	User cannot be able to call any authorized rest call
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Start VoIP**
Test Case ID	TC12
Inputs	Team member Id
Expected Outcomes	User should call another team member who are in same mission and can communicate him/her by voice over ip.
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Start Stream**
Test Case ID	TC13
Inputs	Camera view of user
Expected Outcomes	Team member should send their camera view as a byte array to server
Environmental Needs	Server and Database must properly run also team members device camera should properly run.
Included Test Cases	TC1

Test Case Scenario	**Join Stream**
Test Case ID	TC14
Inputs	Stream Id
Expected Outcomes	User should join and watch stream that are already started by some other user according to their data
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Answer VoIP**
Test Case ID	TC15
Inputs	-
Expected Outcomes	The user should answer call by another team member and transfer their voice both way
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Get Mission Details**
Test Case ID	TC16
Inputs	Mission Id
Expected Outcomes	User should reach all information on database about his/her mission.
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Case Scenario	**Send Data**
Test Case ID	TC17
Inputs	Data (Gps location, sensor data etc.)
Expected Outcomes	User should send data to server and data should post on database
Environmental Needs	Server and Database must properly run
Included Test Cases	TC1

Test Scenario	**Wrong Login**
Test Case ID	TC18
Inputs	Wrong Username and Password
Expected Outcomes	Given username and password authenticate the system system does not confirms it then system show warning " your username or password is wrong ".
Environmental Needs	Server and Database must properly run
Included Test Cases	-

Test Scenario	**Calling Offline Member over VOIP**
Test Case ID	TC19
Inputs	UserID
Expected Outcomes	System should inform actor about the online-offline status of the called member.
Environmental Needs	Server and Database must properly run, Called person must be offline
Included Test Cases	TC1

Test Scenario	**Sending Media from SmartGlass to Mission Control**
Test Case ID	TC20
Inputs	Photo or Video
Expected Outcomes	Smartglass should trigger mobile phone and send media over phone to REST API. GUI should show alarm about that.
Environmental Needs	Server and Database must properly run, Smart Glass should be connected to Mobile Phone
Included Test Cases	-

Test Scenario	**Area Guidance on SmartGlass**
Test Case ID	TC22
Inputs	Area coordinates
Expected Outcomes	Smartglass should trigger mobile phone and retrieve data over phone from REST API. View should reflect moves of head.
Environmental Needs	Server and Database must properly run, areas should be defined by Mission Control Center
Included Test Cases	-

Test Scenario	**Adding area or point to mission over GUI**
Test Case ID	TC23
Inputs	Coordinate or Coordinate List
Expected Outcomes	Selected and posted polygons/points should be saved to database and can be retrieved again over REST.
Environmental Needs	Server and Database must properly run, areas should be defined by Mission Control Center
Included Test Cases	TC1

4.b. Testing Code Structure

Our test codes are stored on GitLab. These test codes under one repository its link is below:

https://gitlab.ceng.metu.edu.tr/iTech/JerseyTest/tree/master/src/test/java/org/itech/searchrescue

4.c. Test Results

Test Case Identifier	Result
TC1	Pass
TC2	Pass
TC3	Pass
TC4	Pass
TC5	Pass
TC6	Pass
TC7	Pass
TC8	Pass
TC9	Pass
TC10	Pass
TC11	Pass
TC12	Pass
TC13	Pass
TC14	Pass
TC15	Pass
TC16	Pass
TC17	Pass
TC18	Pass
TC19	Pass
TC20	Pass
TC21	Pass
TC22	Pass
TC23	Pass

5.References

https://jersey.java.net/

http://hibernate.org/orm/

https://dev.mysql.com/doc/refman/8.0/en/

https://www.jetbrains.com/idea/

https://developer.android.com/studio/index.html

https://en.wikipedia.org/wiki/Voice_over_IP

https://en.wikipedia.org/wiki/Livestream

https://www.reconinstruments.com/products/jet/

https://www.myo.com/

http://www.scosche.com/rhythm-plus-heart-rate-monitor-armband

https://angularjs.org/

http://getbootstrap.com/

6. Appendices

6.a. Definitions, acronyms, and abbreviations

SRS	Software Requirements Specification
S&R	Search And Rescue
AR	Augmented Reality
GPS	Global Positioning System
METU	Middle East technical University
GUI	Graphical User Interface
IDE	Integrated Development Environment
Android SDK	Software Development Kit which is officially released for Android
Android Studio	Official IDE designed for Android
UML	Unified Modeling Language
Use Case Diagram	Diagram of interactions of users with the system
Class Diagram	Diagram that describes the structure of a system by showing its classes, attributes of these classes and method
GIS	Geographical Information System