



Bilkent University

Department of Computer Engineering

Senior Design Project

Project short-name: HandsGiving

Analysis Report

Fırat Yönak, Ege Hakan Karaağaç, Oğuzhan Dere, Mehmet Tolga Tomris, Atakan Arslan

Supervisor: Özgür Ulusoy

Jury Members: Çiğdem Gündüz Demir, Can Alkan

Analysis Report

November 21, 2020

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Contents

Introduction	4
Proposed System	4
Overview	4
Functional Requirements	5
2.2.1 System functionality	5
2.2.2 User functionality	6
2.2.3 Storage of the Application Data	7
2.2.4 Server-Client Communication	7
Non-functional Requirements	8
2.3.1 Usability & Accessibility	8
2.3.2 Reliability of the System	8
2.3.3 Reliability in between the Users	9
2.3.4 Privacy and Security of the User Information	9
2.3.5 Efficiency	9
2.3.6 Extensibility	9
2.3.8 Recovery of the Data	10
2.3.9 Legality	10
2.3.10 Performance	10
Pseudo Requirements	11
System Models	12
Scenarios	12
Use-Case Model	16
Object and Class Model	17
2.5.3.1 Client Class Diagram	17
2.5.3.2 Server Side Class Diagram	18
2.5.4 Dynamic Models	19
2.5.5 User Interface	24
2.5.5.1 Sign In	24

2.5.5.2 Profile	25
2.5.5.3 Home Page	26
2.5.5.4 Social Page	27
2.5.5.5 Video Chat	28
2.5.5.6 Help Request	29
Other Analysis Elements	30
Consideration of Various Factors in Engineering Design	30
Risks and Alternatives	32
Project Plan	33
Ensuring Proper Teamwork	39
Ethics and Professional Responsibilities	39
Planning for New Knowledge and Learning Strategies	40
References	41

Analysis Report

Project Short-Name: HandsGiving

1 Introduction

With the increase in the number of people in need in the world and the extension of human life, solidarity has become more important. People started to need each other more both materially and spiritually. However, the distance between people and their inability to communicate easily with each other made it more difficult to cooperate. HandsGiving is an application intended to be a solution to all these problems faced by people. It will provide a platform for people in need of help to have their voices heard. Those people who are in need will have the opportunity to express their needs to volunteers through the application.

The application will also be designed to reduce the negative effects of the COVID-19 virus on people's social life[1]. As part of COVID-19 measures, elderly people had to be restricted from going out to the streets[2]. Both the elderly who were left alone in their homes due to these restrictions and those who already had fewer people around became more in need of social activities and their needs became more difficult to meet. Thanks to the video chat feature that HandsGiving will offer, the elderly will have the chance to meet with different volunteers and meet their social needs.

2 Proposed System

2.1 Overview

HandsGiving will be an Android application that is designed for hand givers and for the people who need a hand of any type. To be able to reach more people who need a hand, the application is going to be free and will work not only on the latest android version but on lower android versions as well. More details on which android versions are covered are mentioned in further sections. The application will also define a help-request system which will be the key point that binds the person in need and the person who can help together. It is important to note that the helping activity

should not be considered in a single scope that is finance. Financial support will also be possible in this application, nevertheless, they are not the only ones.

To make the reader have a better understanding of the application, it is thought by the team that it might be useful to explain some of the most important requirements in plain English sentences also, right along with upcoming sections. The HandsGiving application will be a free application that will be available on Google Play which is the main application store for Android phones. The application will require an account for the ones who are willing to use it. The users will be able to create and answer help requests. These help requests will be filterable and will sometimes even include unique information such as location information when it is needed. The users will have the option to validate their accounts for less occurrence of fraudulence in the application (it is important to note the users that the application will not have any responsibilities over any fraudulent actions.). The help requests will also have an overview system for the matter of safety. Both the request creator and its answerer will be able to rate each other and write a few sentences of comment after a request is completed either in success or failure. These two factors come under the term, an ‘overview’. The users will be able to view each other’s reviews and past completed requests on their own pages. TCP will be used for the communication between the Android client and the server for the reasons which are also mentioned in further sections.

All the details on the application’s requirements will be discussed in the next 4 sections. Later on, system models will be shown as visuals.

2.2 Functional Requirements

2.2.1 System functionality

- A login and register system for the application database should exist for any possible fraud action.
- The system should have a list of help requests with filter parameters such as social/financial to help benevolent users.
- The system should include the personal information of a user in order to make it easier for other users to decide whether they want to help or not.

- The system should be available on the application store of the according platform for free.
- The system should be able to show the previous helping activities of a user in that user's profile.
- The system should be able to maintain and view the reviews acquired by a user.

2.2.2 User functionality

- Users should be able to login/logout or register through the application.
- Users should be able to create their own profiles and publish their personal information.
- Users should have the option to hide their personal information which is not required to show.
- Users should be able to create help requests either socially or financially.
- Users should be able to help any user who has a specific help request. Help requests can be found from either a profile or the help request list which is also mentioned in the system functionality.
- Users should be able to accept or decline a help request. They do not have to accept every response to a help request.
- Users should be able to download the application from the application store.
- After a request is accepted and completed, both sides of the requests will be able to give a review which includes a rate out of 10 and a short comment consisting of a few sentences (i.e. maximum of 300 characters, this value can be increased depending on the users' demand over time.). Users should be able to view the reviews of users in their profiles.
- Users should be able to list their own requests and cancel them if they are not active or removable.

2.2.3 Storage of the Application Data

- The application data should be stored in a relational database with efficient relations of entities, for easier maintainability of the data.
- The property data of each help request should be kept on the server. The property data includes information such as the content of the request, request creator user ID, and sometimes even the request location coordinate information.
- Each user information with each user-id should be kept in the server, same as the help requests.
- Other important data that will be present in user profiles such as review data should also be kept separately with IDs in the mentioned relational database.

2.2.4 Server-Client Communication

- The communication protocol between the server and clients must be safe since the private information of users will be asked to create a safer environment for the users.
- The communication protocol must be data leak-proof since any incorrect or missing data may lead to bigger problems.

2.3 Non-functional Requirements

2.3.1 Usability & Accessibility

- The application should be available on an application store since the user community of the program should be able to contact each other by downloading the application over an application store.
- The application should have an option that allows an easier interface for elders. This option can be used by users who are not old also.
- The easier interface that is implemented for elder users should be tested with a sample of 10 to 25 elders and optimized if necessary. Both interfaces should be capable of doing the exact same things.
- The easier interface should have bigger text boxes which are easier to touch and activate or have a zoom-in capability.
- The request creation section should be clear, and it should be on the main page of the application.
- The application should notify the user using android's notification layer whenever necessary. There should be an option to choose when to get a notification from the application if the operating system of the user's device does not have an option to handle such cases.

2.3.2 Reliability of the System

- The application should not mix users during the financial or social support of a benevolent. Tests for this requirement must be made.
- The application must have an error handling system that determines the reason for any error and notifies the user accordingly.
- The server of the application should be capable of working as expected in the scope of the maximum number of users.
- The maximum number of users should be fixed to 10000 at first, the number can be increased if the number of users gets close to the maximum amount.

2.3.3 Reliability in between the Users

- A phone number verification system should exist in the system for the purpose of lowering the number of imposter users in the application. The review system also strengthens the reliability between the users.

2.3.4 Privacy and Security of the User Information

- The application should not use the GPS information of a user when it is not necessary. The GPS information should be kept private and protected for any possible abuse. GPS information should not be used without consent.
- The server must be secure enough for any possible threat by a client.

2.3.5 Efficiency

- The application should be able to run smoothly in older generation smartphones since the user community of the application will be including financially suffering people.
- The application's total file size should not take more than 100 MB in order to make it easier to store and access. Redundant files incurred by events such as application updates should be removed immediately.
- The application should be able to detect users who have not been using the application for a long time period and delete the accounts within their knowledge for a better server capacity efficiency, in case of any unexpected user number increase. This can be done by sending a mail to their mail address or by sending a notification message to their phone number if it is registered in the system. The users should be able to stop the deletion progress.

2.3.6 Extensibility

- The object documentation of the application should be systematic and open to any upgrades such as new functional/non-functional requirements or user interface changes.
- The resource codes of the application should be well-commented and easy to understand for newer employees and for the sake of the application.

2.3.7 Maintainability

- The application should have its modules correctly separated, such that any update or change in the application should not interfere with any other non-related module.
- The server should be built such that it should be possible to increase the maximum user number without changing the whole system.

2.3.8 Recovery of the Data

- The application should have a recovery system such that any client data should not be lost when the servers are down. Any unsavable data should be notified to the corresponding user and the server should be recovered back to its latest save state.

2.3.9 Legality

- The application should notify the user in the process of registration that the application does not take any responsibility for any illegal event performed by the users and on any illegal events' consequences on any user.
- The application should not share any private user information with third parties

2.3.10 Performance

- The creation of a request should not take more than 5 seconds when there are no existing errors. If it takes more than 5 seconds, the user must be notified with a reason.
- The starting process of the application should not take more than 10 seconds. Since impatience of users may lead to losing them.

2.4 Pseudo Requirements

- The programming paradigm of the application will be object-oriented programming.
- The integrated development environment for the team will be android studio which is built by IntelliJ Idea[3].
- Git will be used to be able to work as a team properly. It is possible to authenticate an Android Studio project with GitHub.
- Java or Kotlin will be used for the implementation of the application[3].
- Since the application will be used by the unwealthy also, the minimum android version required for the application will be as low as possible. For now, it is decided as “4.4 KitKat (2013)”. However, the decided version is subject to change[4].
- Object classes are, again, subject to change depending on the implementation direction.
- The source code will be consisting of modules that are separate from each other for easier implementation. The modular system will also be more maintainable since it makes the source code more systematic
- The modules will be checked partially before combining them for easier implementation and to handle errors better.
- Licenses of external frameworks (if any external framework exists) will be taken before using them in the project if it is necessary.
- The expenses of the application such as hosting costs will be covered by the team itself. If the application gets near the maximum user number, ads will be added to the application for the purpose of covering the growing expenses.

2.5 System Models

2.5.1 Scenarios

Scenario 1 Sign Up

Actors: Needy Person, Old Person, Benevolent

Entry Conditions: The user opens the app.

Exit Conditions: The user is navigated to the main menu.

OR

The user closes the app.

The flow of Events:

1. The user clicks the “Sign Up” button on the homepage.
2. The registration page is opened.
3. The user enters the required information.
4. The user clicks the “Sign Up” button.
5. An authentication mail is sent to the provided email address.
6. The user enters the code given in the mail.
7. The user clicks the “Verify the Code” button.
8. The system verifies the code and creates the desired account.

Scenario 2 Sign In

Actors: Needy Person, Old Person, Benevolent

Entry Conditions: The user opens the app.

Exit Conditions: The user closes the app.

The flow of Events:

1. The user enters the required information.
2. The user clicks the “Sign In” button on the homepage.
3. The entered information is checked in the database.
4. The user is navigated to the main menu.

Scenario 3 Edit Settings

Actors: Needy Person, Old Person, Benevolent

Entry Conditions: The user clicks the “Settings” button on the homepage.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. If the feature which the user wants to change can be changed directly on the page, the user changes the feature by selecting one of the given options.
2. Else if the feature change is required pop up page, the user clicks the feature button.

3. If a pop-up page is opened, the user selects one of the options.

Scenario 4 Edit Profile

Actors: Needy Person, Old Person, Benevolent

Entry Conditions: The user clicks the “Profile” button on the homepage.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. If the feature which user wants to change is required to enter information, the user enters information to edit.
2. Else if the user can change the feature by selecting an option from the menu.

Scenario 5 Give Feedback

Actors: Needy Person, Old Person, Benevolent

Entry Conditions:

1. At least one help process which the user is involved in needs to be done.
2. The user clicks the “Feedback” button on the homepage.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. The user selects one of the completed help processes on the page.
2. The user enters a rating.
3. The user writes comments about the other side of the help.
4. The user clicks the “Save Feedback” button.

Scenario 6 Respond Assigned Help Request

Actors: Benevolent

Entry Conditions:

The user clicks the “Assigned Help Requests” button on the homepage.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. The user selects one of the assigned requests.
2. The user sees what the help is about and who creates the request.
3. The user accepts or denies the help request.

Scenario 7 Choose a Help to Meet Needings

Actors: Benevolent

Entry Conditions:

The user clicks the “Help Requests” button on the homepage.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. The user selects one of the assigned requests.
2. The user sees what the help is about and who creates the request.
3. The user accepts or closes the request information pop up.

Scenario 8 Respond to a Video Call

Actors: Benevolent

Entry Conditions: The user clicks the “Video Call” button on the homepage.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. The user selects one of the video call requests.
2. The user sees what the creator of the call wants to talk about.
3. The user takes the call or closes the request information pop up.

Scenario 9 Add Friend

Actors: Old Person

Entry Conditions: The user clicks the “Add Friend” button on the social platform page.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. The user selects one of the people who had a video call with the user before from the menu.
2. The user clicks the “Add Friend” button.

Scenario 10 Answer a Video Call

Actors: Old Person

Entry Conditions: The user clicks the “Video Call Requests” button on the social platform page.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. The user selects one of the video call requests.

2. The user sees what the creator of the call wants to talk about.
3. The user takes the call or closes the request information pop up.

Scenario 11 Create a Video Call Request

Actors: Old Person

Entry Conditions: The user clicks the “Video Call” button on the social platform page.

Exit Conditions: The user clicks the “Create” button.

The flow of Events:

1. The user selects one of the general subjects from the menu.
2. The user specifies what he or she wants to talk about.
3. The user selects gender preferences for the users who can see the requests.

Scenario 12 Add Location

Actors: Old Person, Needy Person

Entry Conditions: The user clicks the “Add Location” button on the help page.

Exit Conditions: The user clicks the “Save” button.

The flow of Events:

1. The user clicks the okay button to provide access for the application to GPS.
2. The user clicks the add or the delete button.

Scenario 13 Get Help

Actors: Old Person, Needy Person

Entry Conditions: The user clicks the “Help” button on the homepage.

Exit Conditions: The user clicks the “Back” button.

OR

The user closes the app.

The flow of Events:

1. The user clicks either the “Monetary Help” or the “Physical Help” button.
2. If the user chooses the monetary help option, the user specifies what s/he needs by writing.
3. If the user selects the physical help option, s/he needs to choose help from the menu or create a new request and specify what the user needs.

2.5.2 Use-Case Model

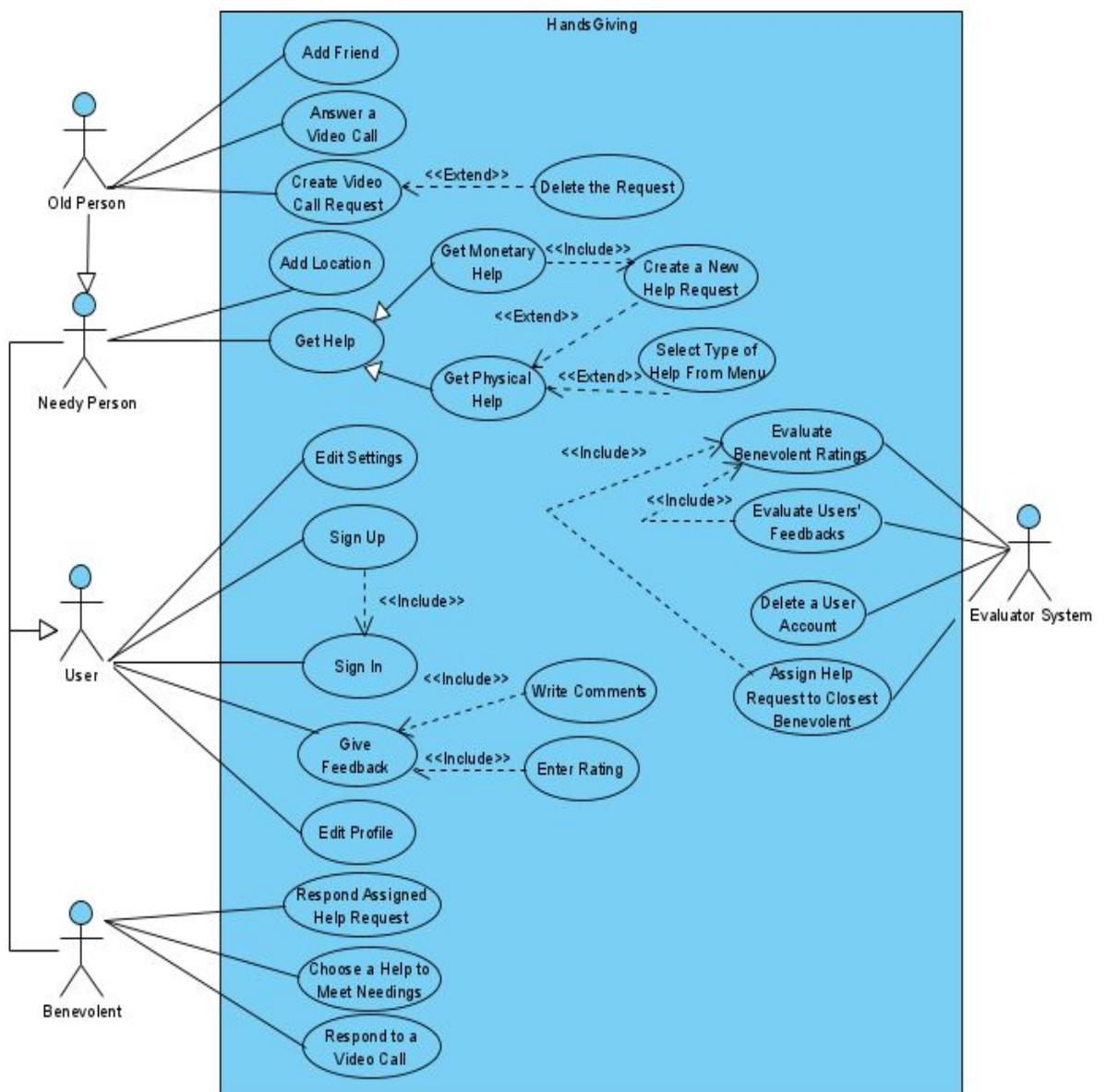


Figure 1: Use Case diagram of HandsGiving

2.5.3 Object and Class Model

2.5.3.1 Client Class Diagram

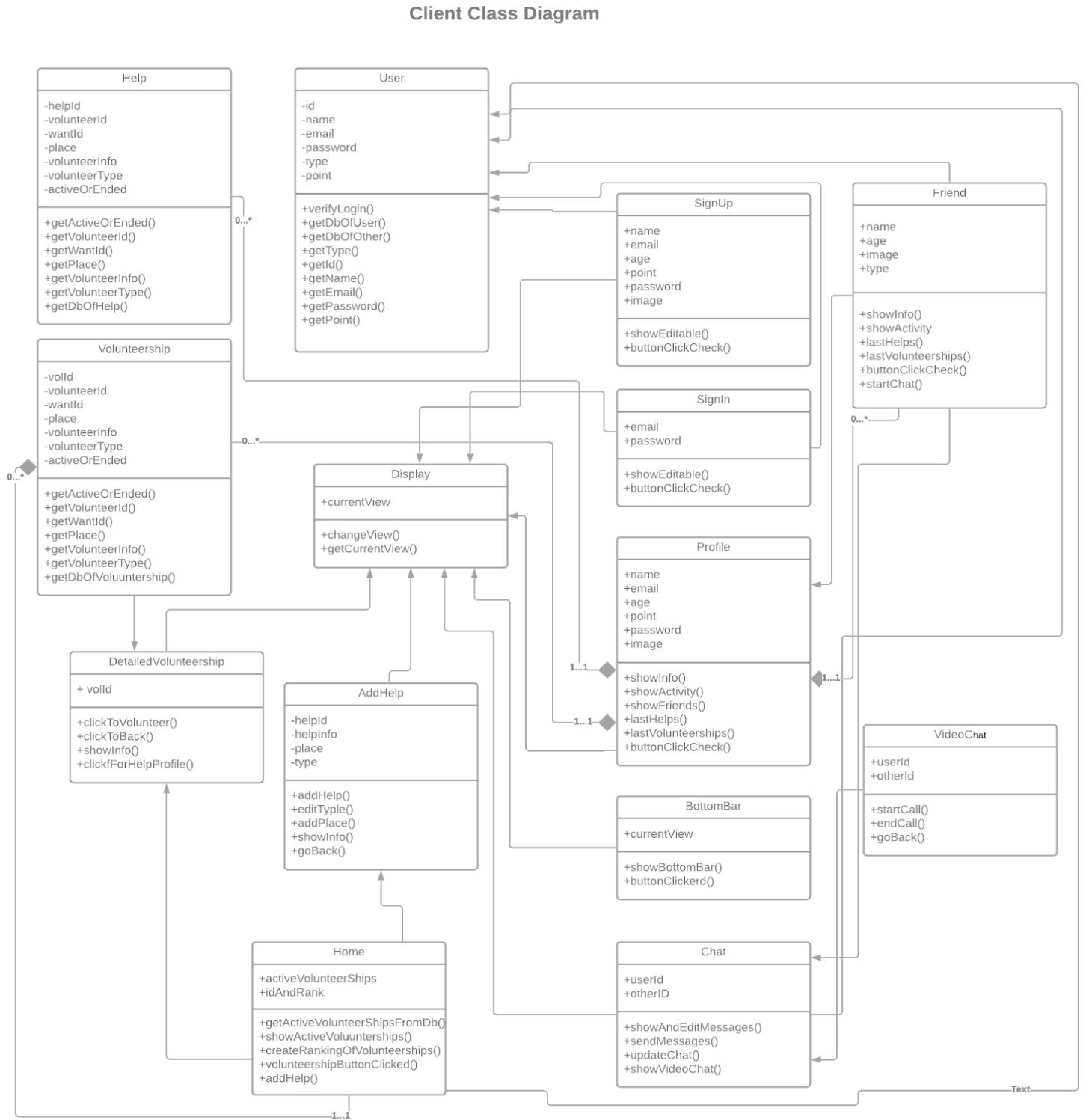


Figure 2: Client side class diagram of HandsGiving

2.5.3.2 Server Side Class Diagram

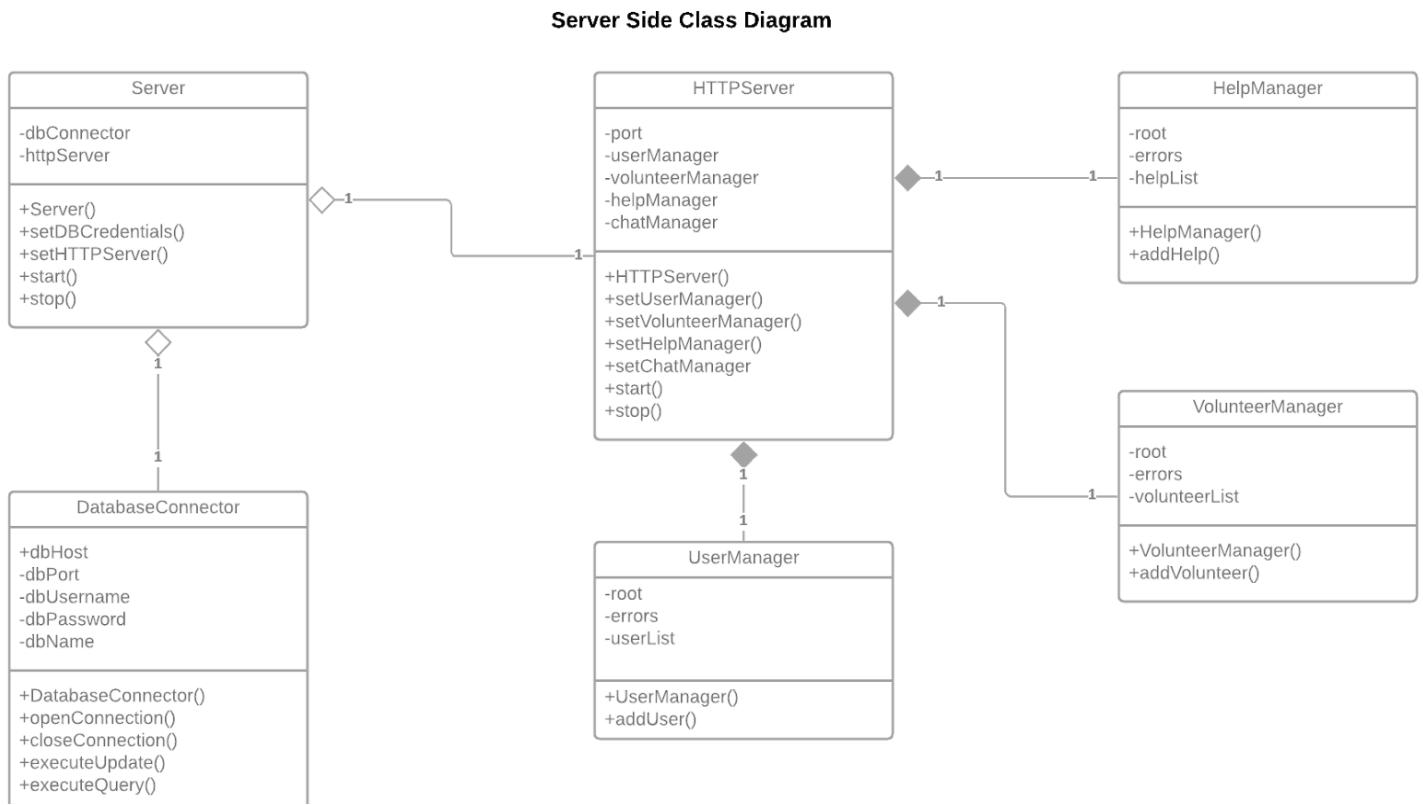


Figure 3: Server side class diagram of HandsGiving

2.5.4 Dynamic Models

There is no final point for the diagram since the user can exit the app any time by using designated application closing action of their phones.

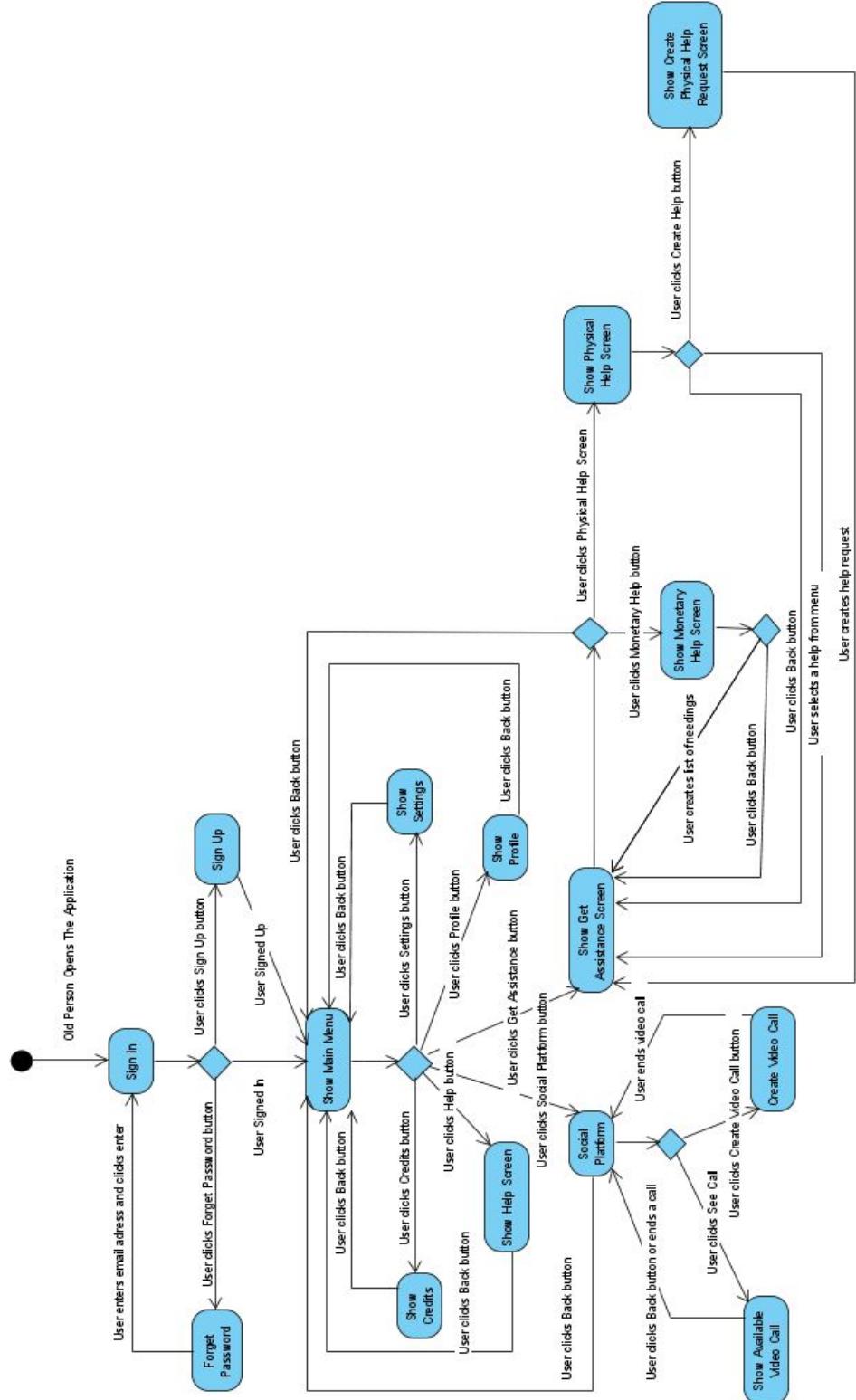


Figure 4: Activity diagram of old person side of application

Start Video Chat

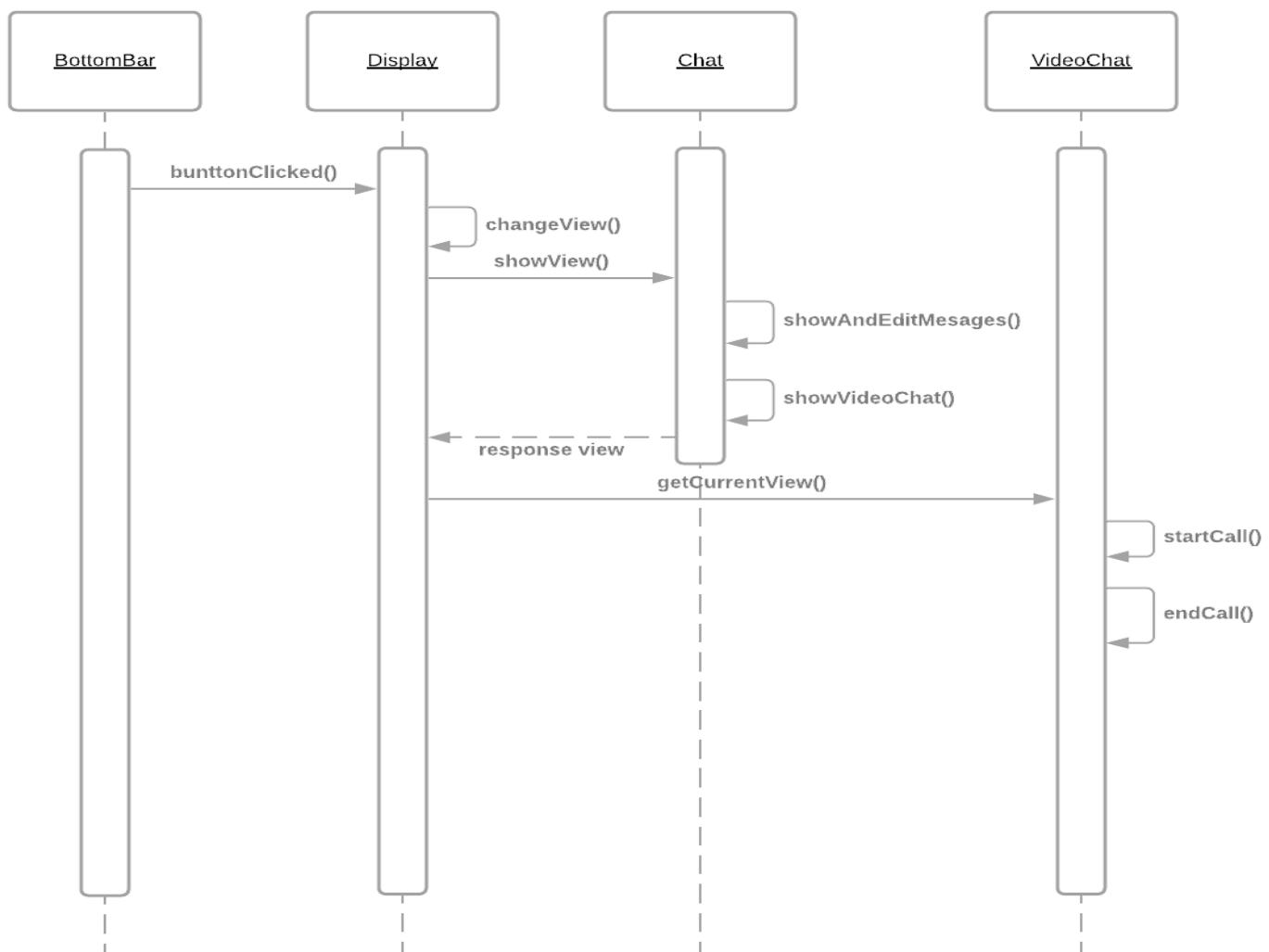


Figure 5: Sequential Diagram of starting video chat

Volunteer for a Help

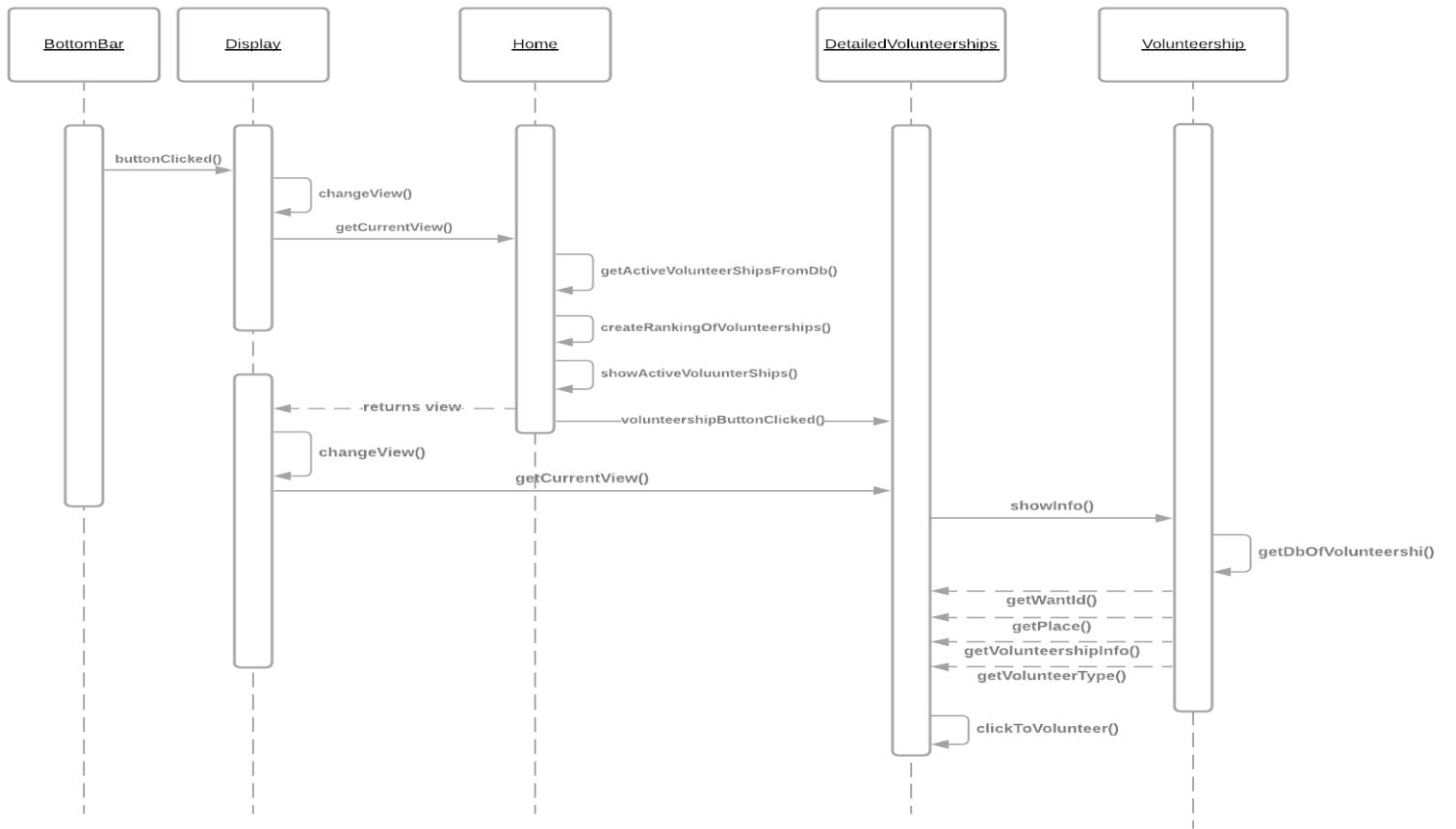


Figure 6: Sequential diagram of being volunteer for a help request

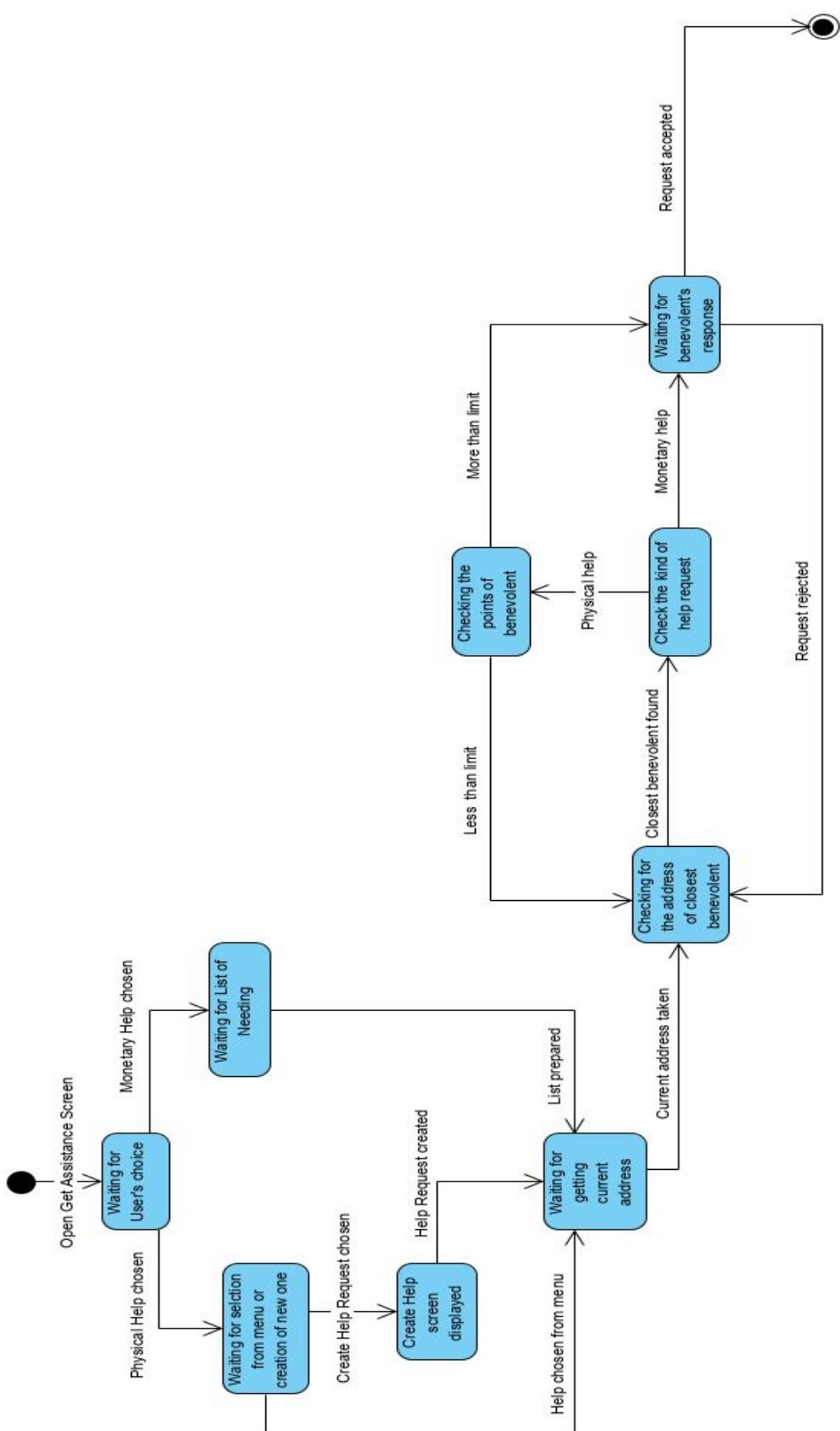


Figure 7: State diagram of help process

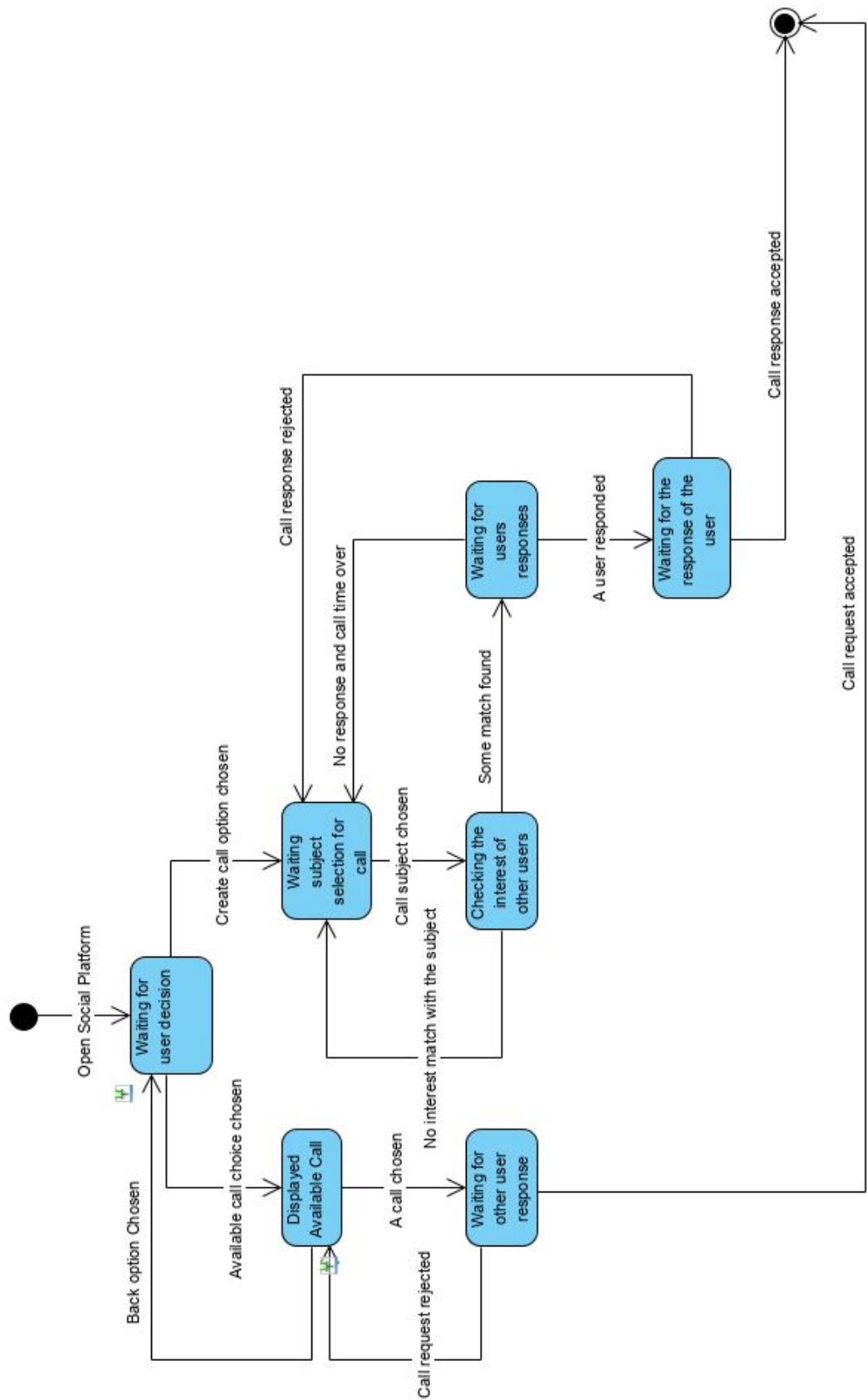


Figure 8: State diagram of video chat process

2.5.5 User Interface

2.5.5.1 Sign In

The page where you sign in or go to the registration page. You can also use google sign in.

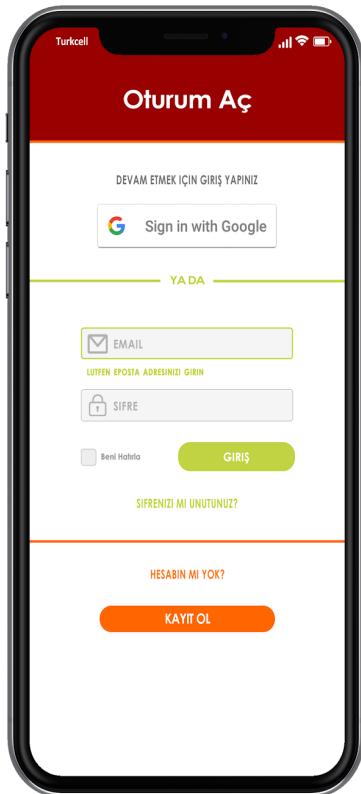


Figure 9: Mockup of Sign In page

2.5.5.2 Profile

The page where you can see your previous volunteerships and edit your profile information.

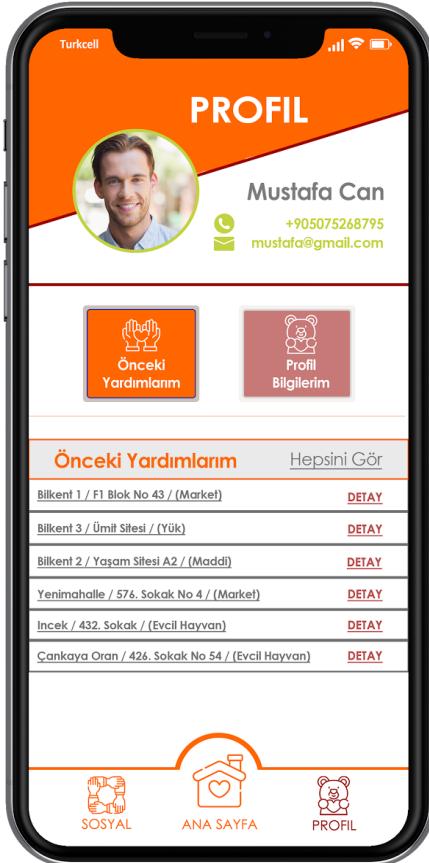


Figure 10: Mockup of profile page of a benevolent

2.5.5.3 Home Page

The page where you give feedback about your volunteership and see the requested and unrequested helps.



Figure 11: Mockup of feedback and help requests page

2.5.5.4 Social Page

The page where you can chat with other people, start voice calls, and meet new people.



Figure 12: Mockup of social platform page

2.5.5.5 Video Chat

The video chat screen where two old people or an old person and a benevolent person talks with each other.

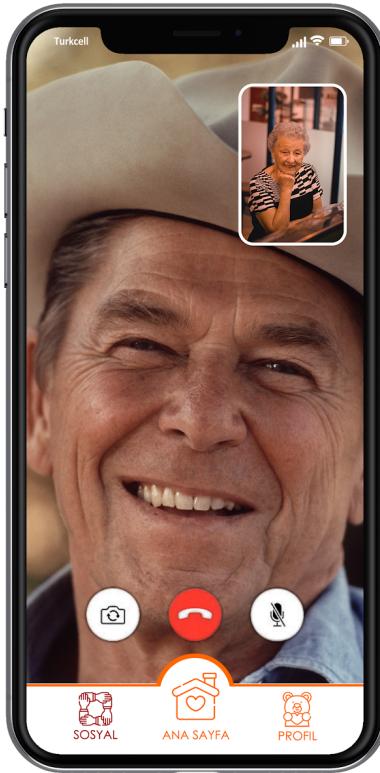


Figure 13: Mockup of video chat screen

2.5.5.6 Help Request

Monetary help request page. You first tell the reason for your request and the amount you need and press the request button page.



Figure 14: Mockup of monetary help request page

3 Other Analysis Elements

3.1 Consideration of Various Factors in Engineering Design

During the research process, there are several factors that may have an impact on the results and development process of the HandsGiving. These factors will be explained in the below sections.

3.1.1 Public Health

A video chat feature will be developed in HandsGiving to facilitate the socialization of the elderly. In this way, it will be aimed to keep the elderly healthy psychologically. Apart from this, people who have difficulties in meeting their food and other basic needs will be helped to make them healthier.

3.1.2 Public Safety

Private information such as user credentials and location will be kept confidential and will not be shared by any third-party apps. In cases where it is necessary to share the credentials between users, they will be shared with the permission of the users.

3.1.3 Public Welfare

It is aimed to increase the welfare of society by creating a platform that will provide easier access to people in need of help.

3.1.4 Global Factors

HandsGiving will not have an impact on global factors because it will be an application specifically created for our country.

3.1.5 Cultural Factors

A cultural interaction will be created by enabling people from different cultures and different regions to help each other.

3.1.6 Social Factors

HandsGiving will create social awareness by making it easier to reach people who need help. A wider audience will be reached to meet people's needs more quickly.

3.1.7 Environmental Factors

HandsGiving will not have an impact on environmental factors because it will be a completely online application.

3.1.8 Economic Factors

HandsGiving will not be for profit in any way, neither for the volunteers using the application nor for our project group. People who want to use the application just need to register for free. Volunteers who want to help needy people can make certain payments for this purpose.

Table 1: Factors that can affect analysis and design.

	Effect Level	Effect
Public Health	9	Mental and physical health
Public Safety	9	Keeping private information confidential
Public Welfare	10	Raise the whole society above a certain level of welfare
Global Factors	0	None
Cultural Factors	5	Bringing different cultures together while helping each other
Social Factors	7	Raising social awareness about people in need of assistance
Environmental Factors	0	None

Economic Factors	6	Having enough economic strength to meet the needs of people in need
------------------	---	---

3.2 Risks and Alternatives

HandsGiving aims to provide a platform to bring volunteers and people who are in need of help together. To do this, there is a need of detecting the location of both volunteers and needy people. The application requires the location of needy people to assign a volunteer located near to them. If the related locations are not detected precisely, volunteers may be assigned to a faraway and irrelevant location. In such cases, the application will need to provide a new system to keep the locations of the users. In this new system, users will provide their address information by filling in options such as district, neighborhood, apartment, and flat number while registering for the application. In this way, the application will use this information while assigning a volunteer to a needy person.

The second risk is that some people may abuse the application by requesting help although they actually do not need any. To prevent such cases, the application will ask for the Turkish ID Number of the users in order to verify their accounts. In this way, if any abuse happens, users will report it to the application. The application will detect the person involved in the case and an investigation will be started by the administration unit.

Table 2: Risks

	Likelihood	Effect on the project	B Plan Summary
Incorrect user location detections	5	Risk of assigning volunteers to far away and irrelevant locations	Enabling users to enter their addresses directly by filling in the options
Abusing the purpose of application	7	Risk of helping to malicious people	Asking Turkish ID Number to the users

3.3 Project Plan

Table 3: List of work packages

WP#	Work package title	Leader	Members involved
WP1	Analysis	Atakan Arslan	Everybody
WP2	Design	Mehmet Tolga Tomris	Everybody
WP3	Development 1	Oğuzhan Dere	Everybody
WP4	Development 2	Fırat Yönak	Everybody
WP5	Testing	Ege Hakan Karaağaç	Everybody

Table 4: Project Work Plan - WP1

WP 1: Analysis			
Start date: 16 September 2020 End date: 21 November 2020			
Leader:	<i>Atakan Arslan</i>	Members involved:	<i>Mehmet Tolga Tomris</i> <i>Oğuzhan Dere</i> <i>Firat Yönak</i> <i>Ege Hakan Karaağac</i>
Objectives: <i>The purpose of this work package is to explore the requirements and other factors that should be taken into account when designing the project.</i>			
Tasks: <p>Task 1.1 Requirement Meetings: <i>Analyzing and determining all requirements and specifications of the project.</i></p> <p>Task 1.2 Specification Report: <i>Writing a report about the specifications and requirements of the project.</i></p> <p>Task 1.3 Analysis Report: <i>Writing a report about the use cases, dynamic modes, and development structure of the project.</i></p>			
Deliverables <p>D1.1: <i>Specification Report</i></p> <p>D1.2: <i>Analysis Report</i></p>			

Table 5: Project Work Plan - WP2

WP 2: Design			
Start date: 12 October 2020 End date: 8 February 2021			
Leader:	<i>Mehmet Tolga Tomris</i>	Members involved:	<i>Atakan Arslan</i> <i>Oğuzhan Dere</i> <i>Firat Yönak</i> <i>Ege Hakan Karaağac</i>
Objectives: <i>The purpose of this study module is to explore the requirements and other factors that should be taken into account when designing the project. This part consists of high and low-level designs of the project.</i>			
Tasks: <p>Task 2.1 Evaluation of the Analysis: Evaluating the analysis part for the design.</p> <p>Task 2.2 High-Level Design: Defining project design goals and dividing the project into subsystems.</p> <p>Task 2.3 Low-Level Design: Defining design principles to be used in the project.</p>			
Deliverables <p>D2.1: <i>High-Level Design Report</i></p> <p>D2.2: <i>Low-Level Design Report</i></p>			

Table 6: Project Work Plan - WP3

WP 3: Development 1			
Start date: 12 October 2020 End date: 27 December 2020			
Leader:	<i>Oğuzhan Dere</i>	Members involved:	<i>Atakan Arslan</i> <i>Mehmet Tolga Tomris</i> <i>Firat Yönak</i> <i>Ege Hakan Karaağac</i>
Objectives: <i>The purpose of this work package is to create the basic structure of the project to be prepared for the presentation and demo.</i>			
Tasks: <p>Task 3.1 Creation of Database: <i>Creating the database that will be used in the project.</i></p> <p>Task 3.2 Implementation of User Interface: <i>Implementing the user interface of HandsGiving</i></p> <p>Task 3.3 Implementation of Some Features: <i>Implementing some features for the first demo.</i></p>			
Deliverables <p>D3.1: <i>First Presentation and Demo</i></p>			

Table 7: Project Work Plan - WP4

WP 4: Development 2			
Start date: 8 February 2021 End date: 26 April 2021			
Leader:	<i>Firat Yönak</i>	Members involved:	<i>Atakan Arslan</i> <i>Mehmet Tolga Tomris</i> <i>Oğuzhan Dere</i> <i>Ege Hakan Karaağac</i>
Objectives: <i>The purpose of this work package is to create the advanced structure of the project to be prepared for the final presentation and demo.</i>			
Tasks: <p>Task 4.1 GPS Location System: <i>Implementing GPS location system.</i></p> <p>Task 4.2 Server and Database: <i>Creating the server and database according to the specifications of the project.</i></p> <p>Task 4.3 Implementation of All Other Features: <i>Implementing all features and finalizing the project for the final demo.</i></p>			
Deliverables <p>D4.1: <i>GPS Location System</i></p> <p>D4.2: <i>Server and Database</i></p>			

Table 8: Project Work Plan - WP5

WP 5: Testing			
Start date: 21 November 2021 End date: 7 May 2021			
Leader:	<i>Ege Hakan Karaağaç</i>	Members involved:	<i>Atakan Arslan</i> <i>Mehmet Tolga Tomris</i> <i>Oğuzhan Dere</i> <i>Fırat Yönak</i>
Objectives: <i>The purpose of this work package is to test the implemented parts of the project. This work package is the last one, the development and testing of the project will finish.</i>			
Tasks: Task 5.1 GPS Location Testing: <i>Testing GPS accuracy.</i> Task 5.2 Server and Database Testing: <i>Testing the flow of information between application and server and database.</i> Task 5.3 Application Testing: <i>Testing all features of the project.</i>			
Deliverables D5.1: <i>HandsGiving Application</i>			

Gantt Chart for the Project Schedule

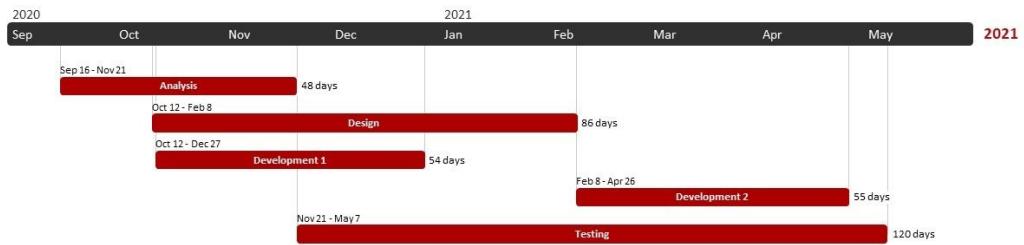


Figure: Gantt Chart for the Project Schedule

3.4 Ensuring Proper Teamwork

To encourage everyone to participate in leadership, we divide the project into work packages and make everyone the leader of one package. Each member of the team is assigned to all work packages so that they have an equal workload. In this way, each project member will be aware of all processes of the project. All team members will be in contact with each other at every moment of the project in order to complete the tasks simultaneously. There is a dedicated GitHub repository to maintain the project's code base and progress. All team members have access to the repository. This repository will be shared with the supervisor and jury members of the project if they wish.

3.5 Ethics and Professional Responsibilities

The application will have some confidential information about users, but this information will not be shared with anyone without the user's permission. In addition, the locations of the users will not be shared with anyone too. The application will ask users to obtain their permission before sharing any information with another user. The

application will provide a terms and conditions agreement that the user needs to read and accept while registering to HandsGiving. Apart from these, to prevent any copyright issues, our team will prefer to use open-source products in order to develop the application.

3.6 Planning for New Knowledge and Learning Strategies

Our main learning strategies are self-learning and researching. Our group will watch video tutorials and read online articles and discussions to gain new knowledge. Our group will also follow the latest academic publications in order to improve the application. In the project, GPS will be used to determine the locations of the users. Therefore, our group will have to learn how to use and integrate GPS into the application. To implement the video chat feature, our group also needs to gain deeper knowledge about the subject. To do all this, our group members will utilize their self-learning skills and do research using accurate and related sources.

4 References

- [1] Li, Sijia, et al. “*The Impact of COVID-19 Epidemic Declaration on Psychological Consequences: A Study on Active Weibo Users.*” MDPI, Multidisciplinary Digital Publishing Institute, 19 Mar. 2020.[Online]. Available: www.mdpi.com/1660-4601/17/6/2032/htm. [Accessed: 10- Oct2020].
- [2] Hürriyet.” *65 Yaş Üstü Ne Zaman Sokşa Çıkacak?* ”.[Online]. Available: www.hurriyet.com.tr/galeri-65-yas-ustu-ne-zaman-soksha-cikacak-41503410/2. [Accessed: 11- Oct2020].
- [3] Wikipedia. ” *Android Studio* ”.[Online].Available: https://en.wikipedia.org/wiki/Android_Studio#:~:text=Android%20Studio%20support%20all%20the,platform%20version.%22%20External%20projects%20backport .[Accessed: 15- Nov2020].
- [4] Wikipedia. ” *Android version history* ”.[Online].Available: https://en.wikipedia.org/wiki/Android_version_history. [Accessed: 15- Nov2020].