Bilkent University

Department of Computer Engineering

Senior Design Project

Project short-name: HandsGiving

High-Level Design Report

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

Supervisor: Özgür Ulusoy

Jury Members: Çiğdem Gündüz Demir, Can Alkan

High-Level Design

December 27, 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

High-Level Design 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 for social activities and their needs became more difficult to meet. Thanks to the video chat feature that HandsGiving will offer, people will gain the chance to meet with different volunteers and meet their social needs. HandsGiving is going to be a free application, which has the sole purpose of connecting people of any age with each other for various reasons. The application feeds on the mutual benefits of the communal life, which leads to a win-win situation rather than what most applications were created for. It is also worth mentioning that the scope of this application is even more than social concerns.

1.1 Purpose of the system

HandsGiving is a mobile app, where people of various status and age can help and connect with each other. The application is planned to be highly related to the dependency of the society between its individuals. Therefore, the mentioned dependency will be playing an important role in the realization of the system's purpose. The system will be keeping the requests created by users from around the world, filtering them before presenting them to users who are willing to help. The way the program reshapes and keeps the data of the users will be one of the most important concerns of the program's purpose, and thus, main functionality.

1.2 Design goals

Our design goals aim for both simplicity and a fancy user interface. The data manipulation has got to be as error-proof as possible while the system speed is not affected badly. The reason behind the aim for simplicity is simply the wide range of the expected users. For this specific reason, the application must be easy to use and easy to understand as much as possible. Here are some goals which are worth mentioning related to the system:

1.2.1 User Related

- 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.

1.2.2 System Related

- 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 related 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.

1.2.3 Usability & Accessibility Related

- 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 that 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.

1.2.4 Reliability Related

- 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.
- 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.

1.2.5 Privacy and Security Related

- 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 from any possible abuse. GPS information should not be used without consent.
- The server must be secure enough for any possible threat by a client.

1.2.6 Efficiency Related

- 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.

1.2.7 Extensibility Related

- 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.

1.2.8 Maintainability Related

- 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.

1.2.9 Legality Related

- 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

1.2.10 Performance Related

- 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.

1.3 Definitions, acronyms, and abbreviations

App: Stands for ‘application’. In this report’s context, an application is a smartphone program that is designed in various programming languages for various purposes. Such programs are reachable through ‘application stores’ of the corresponding smartphone.

HandsGiving: The name of the application. There were various other ideas such as HelpOthers or Help.me, yet the team chose this one.

Request: A request is a demand of someone for asking help for the things which cannot afford or do.

Social Platform: The social platform is a part of the application which provides opportunity to the users to socialize with other users.

Video Chat: It is a communication way in the social platform of the application. By using this feature the users can talk and see each other at the same time.

Feedback: It is the evaluation process of the help process by the involved users. With this feature, the users can rate each other and comment about other users. The application uses these evaluations to enable the benevolents to participate in physical help and also the significant amount of negative feedback may result in the deletion of the user account.

Client: A user of the application. Then, the client side of HandsGiving is the interface in which a benevolent person can respond to help requests, take video calls and a needy person and an old person can create help requests and use the social platform .

Server: The back-end of the application. This is where information about the requests, feedback and accounts are kept.

Account: The personal data of a user such as completed help processes, their username, password and location information is the contents of their account. An account can be created and be logged into inside the HangsGiving app.

Needy: It is the word to describe the users who use the application to get help for the things which they cannot afford or do.

Benevolent: It is the word to describe the users who use the application to help needy people.

Correction Process: This is the process which is used to be sure the email address provided by the users actually belongs to them.

1.4 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, yet, it is not the only one.

The application's front end will be just as important as the back end since the user interface is a substantial parameter on the preference of this application among other applications. We want to make sure that the front end's usability is age-dependent in order to extend the user profile of the application. 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 fraudulency 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. Further sections will be focusing mainly on the architectural properties of the system.

2. Proposed software architecture

2.1 Overview

Understanding the software architecture and analyzing the subsystems of the project HandsGiving will give information about the relations of the components and subsystems. The post sections of this part of the report will give detailed information about the technologies that will be used in the project and the main and subsystems of the architecture.

2.2 Subsystem decomposition

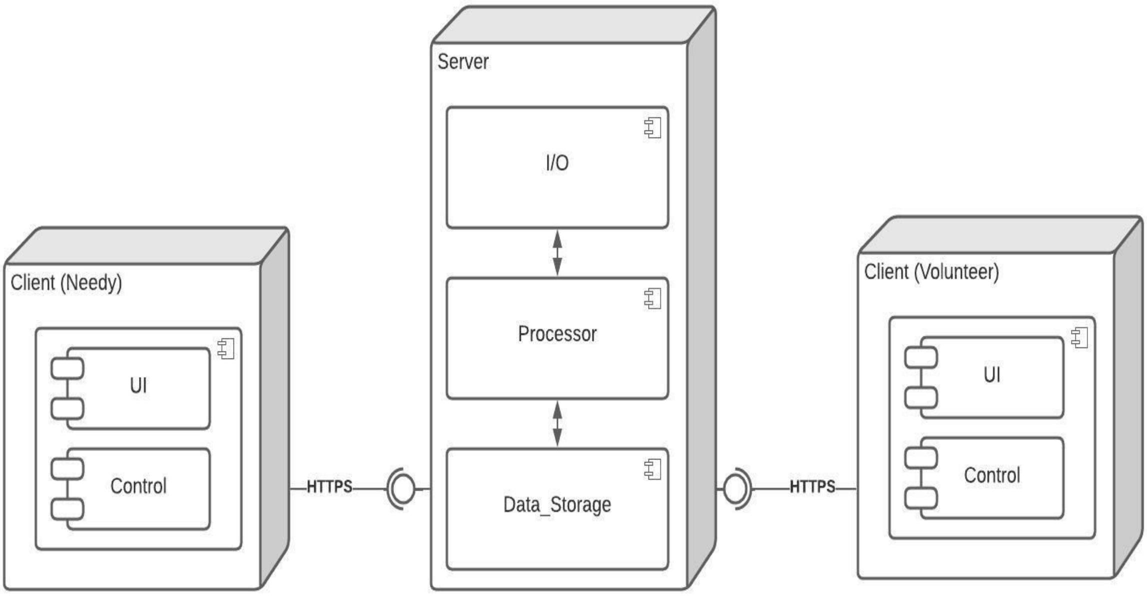
The whole system is decomposed into two main architecture models: client & server. The client architecture will be implemented into smartphones (Android OS) and the server-side will be developed with Spring and MySql Database.

There will be two client-side architectures, first for needy people and the second for volunteers. Both of the client-side architectures will be divided into two subsystems which are named: Control unit and UI unit. The UI side will be the part where the client interacts with the user interface of the project and the control subsystem of the client-side manages the requests of the client and the responses of the server, stores the local variables, and manages the client-side functions and methods. When a change occurs on the client-side and the user submits any request, the control unit manipulates the data and sends a request to the server-side. After getting the request from the client-side, the server stores the information to the database and sends a response to the control-unit. After getting the response, the control unit changes the UI-side accordingly to the response.

The server-side architecture will be divided into three main subsystems: I/O unit & Processor unit & Data_Storage unit. The I/O unit will handle the requests that come from the client-side and manipulate the requests to send the request to the processor unit. It also sends a response to the client-side after the request is processed in the Processor unit and stored in the database. The Processor unit handles the user's information, authentication, and requests. According to the request it sends a request to the Data_Storage unit to store the data in the database.

2.3 Hardware/software mapping

The Client-side of the system will be separated into two parts. The first client-side will be for volunteers which will operate on Android smartphones with GPS for location tracking and a camera for video calls. The second client-side will be for the people who seek help. This side will also operate on Android smartphones with GPS and a camera. Needy people who are on the second client-side can create a help request, chat, and start a video call with other people. Volunteers who are on the first client-side can accept the help requests of people with needs and chat with other people. The server side will be developed with Spring. For the database, MySQL technology will be used to manage the database. Clients will send requests to the server-side with HTTPS protocol[3] and the interactions between two sides of the server will be managed by the REST API in Spring[4].



2.4 Persistent data management

For HandsGiving application, persistent data includes users private information, help information for each user who is at least one time a part of the help process, feedbacks from users which will be private for each user and each one of them can only see their feedbacks for other users, last calls information and location information of the users. Users’ information will be kept on the server as long as the user deletes his or her account from HandsGiving. Feedbacks will be taken from each user who is a part of the help process and will be stayed on the server. These taken feedbacks will be used to evaluate the users and kept for each user until the user deletes the account. Help and call information will be kept from their creation to two sides of the processes delete their account and these data will be available for users after the end of the process.

In the HandsGiving application, persistent data will be kept with the MySQL database. The private personal information will be used for authentication and some services of the application like help and call activities. This information will not be shared with any third parties, just will be used in the application. Other data like feedback and help information will be kept to inform the people when they want to access these data.

The server of the application will be built by using java. In order to save some data locally, the HandsGiving will benefit from SQLite[5]. The purpose of the local storage is to provide fast access for users to some information like profile. For the database system, the application will benefit from MySQL which is a free and open-source relational database management system emphasizing extensibility and technical standards compliance[6].

2.5 Access control and security

Each user has to create an account when they open the application. To register the HandsGiving, they need to provide required information like name, surname, and mail address. After the submission, the system sends a verification mail to the provided email address by the user. When the user clicks the verification button, the registration process will be done. There is a time limitation for the button. If the button is not clicked in 30 minutes, the email automatically will be disabled. This is the correction process of the application. Even if the user clicks after this time, the registering process will not be completed. In the application, there are three types of users who are old, needy, and benevolent. Each of them has access to different services of the application. A user cannot have accounted for more than one type of user account. It will be checked by the system. Each mail account must be different for each user. An email account cannot be used to register more than once. To enter the system, users need to provide their registered email addresses and passwords. If the provided email address and the password match with a database object, the user will enter his or her account. During the authentication process, JWT (JSON network token) will be used to secure the transmitting information[7].

The private information of the user will not be shared with third parties. Only their names will be included in a help process or a video chat when they will be a part of it. Needy users' locations will be shared with benevolent only when they request help. Otherwise, this data also will be unavailable for others. Both email addresses and passwords will not be shared and just be used for the authentication process. When the users delete their accounts, their information will be permanently deleted from the database.

2.6 Global software control

Event-driven software control is used to manage applications by interactions of users. Each one of the user actions like creating a help request, giving feedback, or taking a video call are events that cause a request from the client-side to the server. In the server, requests are handled and there is a need to return required information, they will be provided to the user. Some provided information will not be available for users. For instance, when a user enters an email address and a password, if both information is correct, an authentication token will be delivered to the client-side; however, it will be used by the application and opens the main menu screen for the users.

2.7 Boundary conditions

2.7.1 Initialization

HandsGiving will be an Android-based application that’s why the users need to have an Android device to download and use the application. This kind of device is not important; therefore, they can use the application on a smartphone or tablet. When a user launches the HandsGiving app for the first time, s/he needs to create an account if they have not had one already. To create an account the user needs to provide the required information. S/he can write their name, surname, and email address to the provided text boxes, and also password should be entered twice the time to the relevant field. These two passwords need to match each other. Otherwise, the registration process fails and the user receives a message saying that the password is wrong.

If there is an account that is registered on the system, the user can log in by entering the email and password. Provided data needs to match with a registered combination in the database. If there will not be a match, the user cannot access the app and receives a message saying that your email address or your password is wrong. The users need to have a stable internet connection to both install and launch the application.

2.7.2 Termination

The termination process is not complicated in the HandsGiving. The users can exit from the application whenever they want by closing the app with android provided buttons or the provided log out button on the main screen.

2.7.3 Failure

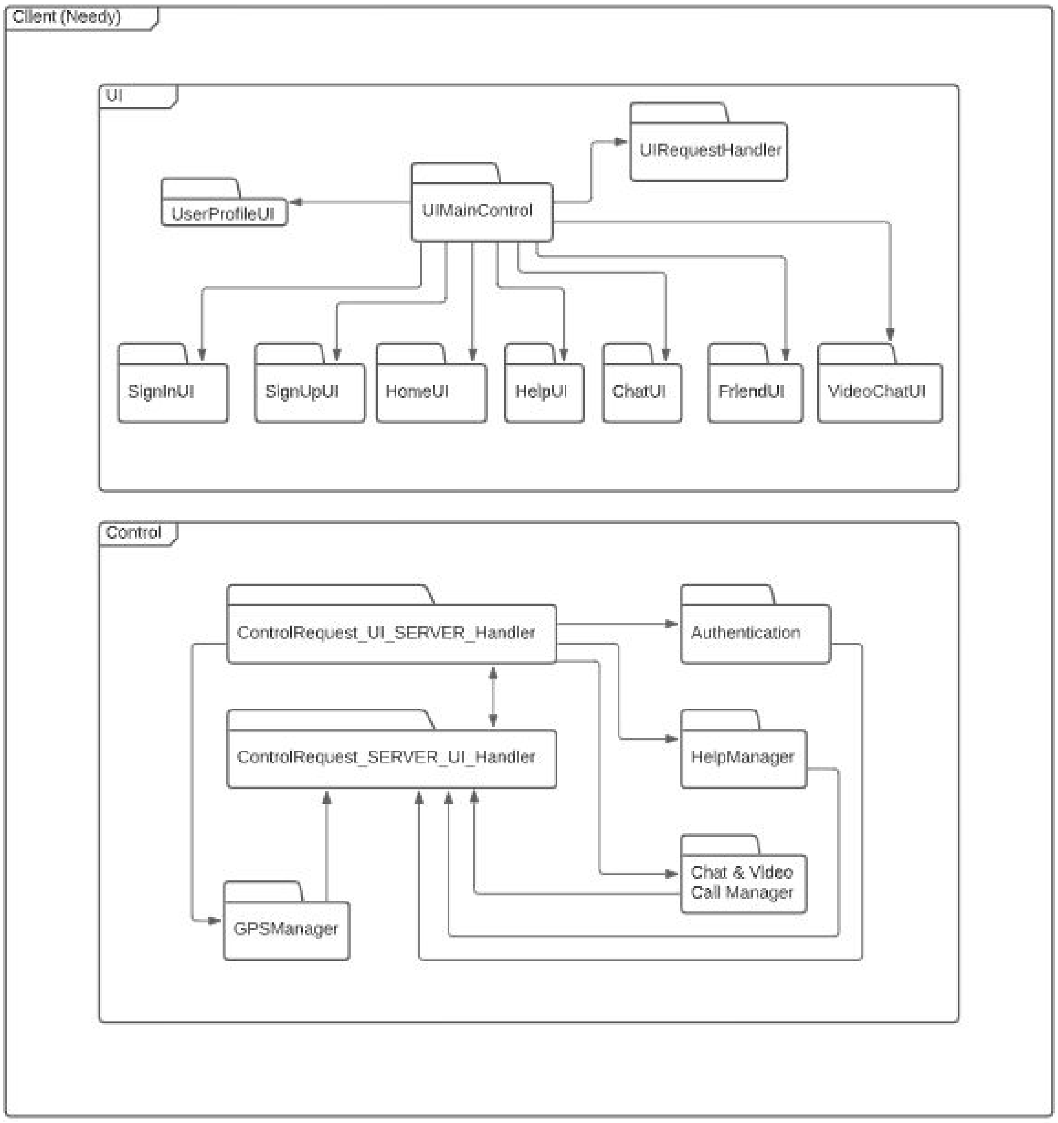
When the HandsGiving fails, the user should close the app and reopen it. They need to log in again since the app is closed. No data loss will happen since all of the completed processes are recorded into the database; however, if the application fails while the user is taking action. This uncompleted action will not be recorded that’s why after the user reopens the app, the action needs to be done again.

A stable internet connection is important while using the app since the internet provides the connection between the server and the installed app. If the user device loses the internet, HandsGiving may fail.

3. Subsystem services

The HandsGiving Project has 2 subsystems for the client-side and 1 for the Server-side. Client-side subsystems are separated into two: Client (Needy) and Client (Volunteer) and the Server-side will manage both client subsystem sides.

3.1 Client (Needy)



This client-side for needy people will work on smartphones with Android operating systems. The UI part of this client-side will manage the view of the app and send requests to the control part according to the events the user enters. The Control side will manage and manipulate the requests that come from the UI and the server-side to send appropriate responses.

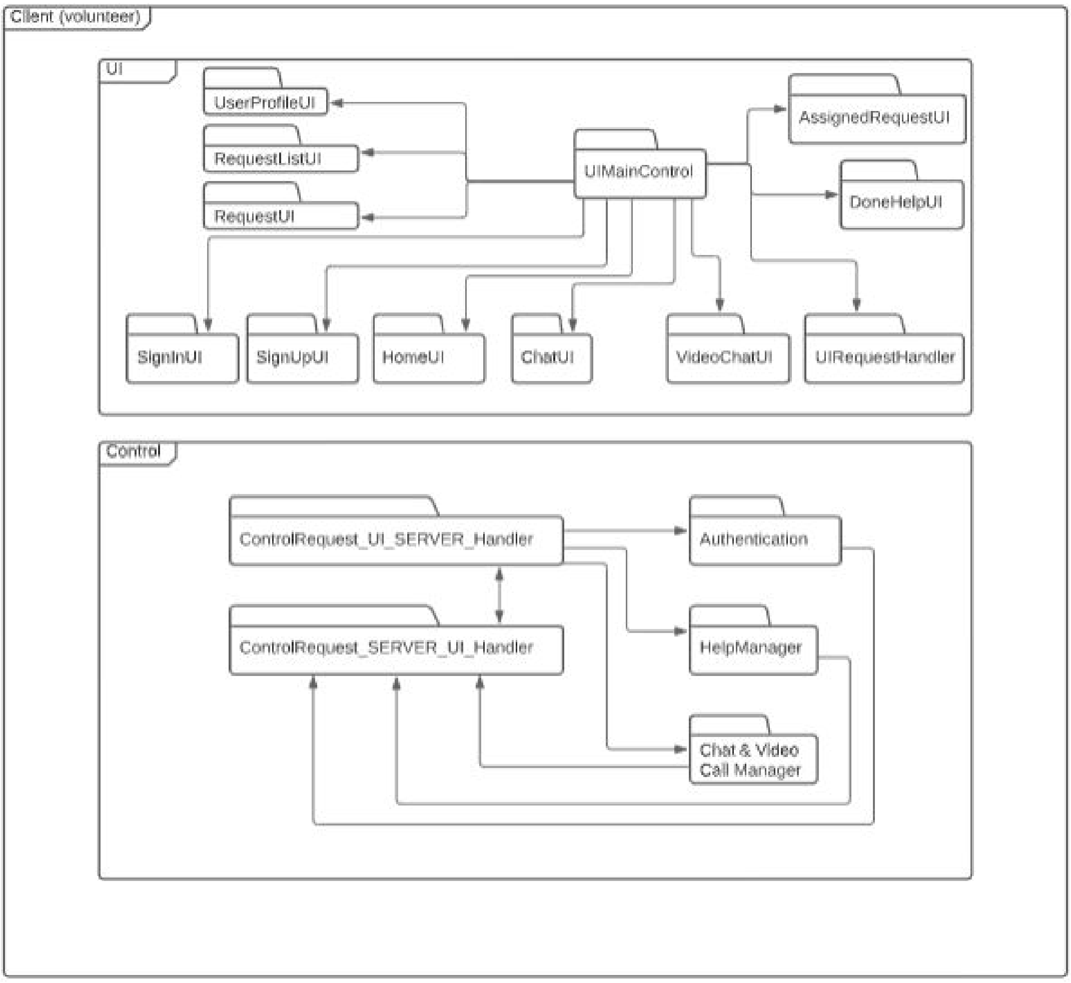
3.1.1 UI

UIMainControl: The main control part of the UI which controls all the views. Sends requests to UIRequestHandler that come from the views parts and changes view according to the response.
UIRequestHandler: Sends requests to control part and manipulates the responses to send to UIMainControl.
SignInUI: Authentication view for sign in.
SignUpUI: Authentication view for sign up.
HomeUI: View of the home page of the needy person, can add a help request in this view.
HelpUI: View of Help Requests of the needy people.
ChatUI: View of chat and video call operations of the client.
FriendUI: View of all of the friends the user has.
VideoChatUI: View of video chat among the users.
UserProfileUI: View of the profile of users. Shows the properties of a user account.

3.1.2 Control

ControlRequest_UI_Server_Handler: Handles the requests that come from the UI part and manages basic requests and sends the information to ControlRequest_Server_UI_Handler and for complicated requests, it communicates with the appropriate control part of the subsystem.
ControlRequest_Server_UI_Handler: The last manipulation of responses that came from the server or from control parts of the project to create serviceable responses to the UI part.
Authentication: Authentication for safe connection with the server.
HelpManager: Manages the help request of the needy person.
GPSManager: Manages the current position of the needy person.
Chat & Video Call Manager: Managing objects of the project where chat and video call operations are handled.

3.2 Client (Volunteer)



3.2.1 UI

UIMainControl: The main control part of the UI which controls all the views. Sends requests to UIRequestHandler that come from the views parts and changes view according to the response.

UIRequestHandler: Sends responses of benevolent to control part and manipulates the responses to send to UIMainControl.

SignInUI: Authentication view for sign in.

SignUpUI: Authentication view for sign up.

HomeUI: View of the home page of the needy person, can add a help request in this view.

DoneHelpUI: View of the completed help process of the user.

ChatUI: View of chat and video call operations of the client.

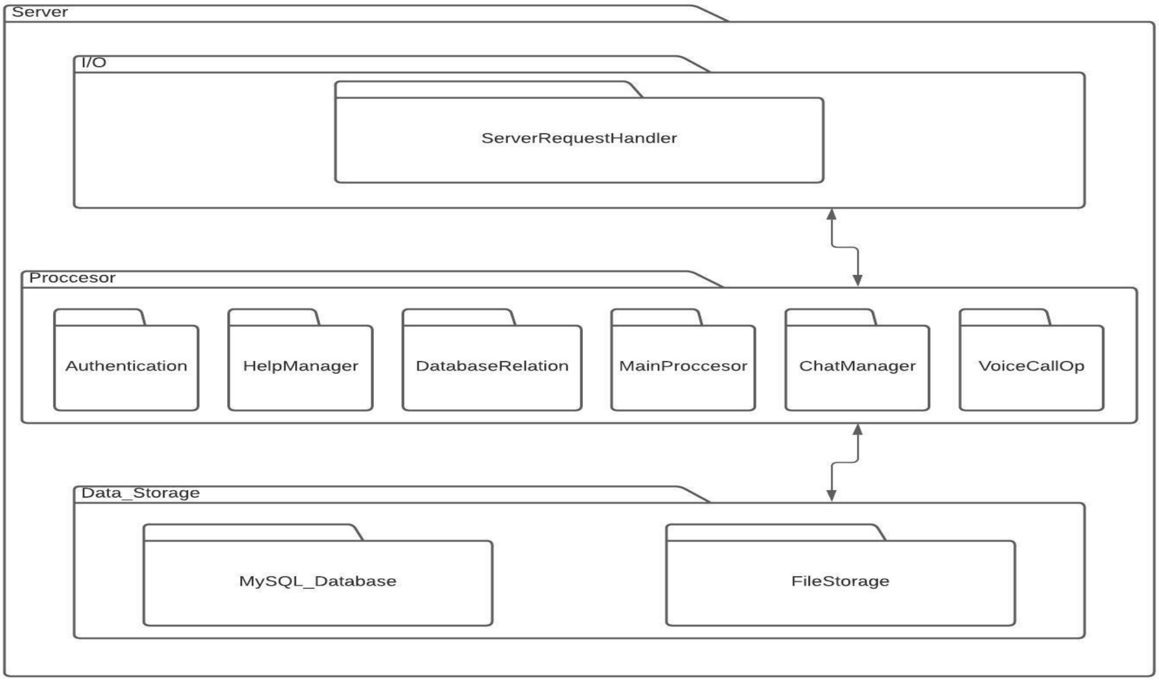
VideoChatUI: View of video chat among the users.
UserProfileUI: View of the profile of users. Shows the properties of a user account.
RequestListUI: View of the unaccepted help requests of needy people.
RequestUI: View of a specific request which includes information about the request.
AssignedRequestUI: View of the assigned requests to the user.

3.2.2 Control

ControlRequest_UI_Server_Handler: Handles the responses that come from the UI part and manages and sends the information to ControlRequest_Server_UI_Handler.
ControlRequest_Server_UI_Handler: The last manipulation of responses that came from the server or from control parts of the project to create serviceable responses to the UI part.
Authentication: Authentication for safe connection with the server.
HelpManager: Manages the assigned help requests and the responses of the user. .
Chat & Video Call Manager: Managing objects of the project where chat and video call operations are handled.

3.3 Server

The server structure is divided into three main layers: the I / O layer, the processor layer, and the storage layer. The I / O layer manages communication between clients and the server. The processor layer manages all the calculations and decisions made on request come from the I / O layer. It also acts as a bridge between the I / O layer and the storage layer.



3.3.1 I/O Layer

This is the subsystem that is the server's gateway to the outside world.

Request Handler: Provides a REST API to receive client requests and deliver them to the processor layer to get responses to requests.

3.3.2 Processor Layer

Authentication: This module authenticates the information of the users while they are accessing the server.

HelpManager: This module processes help requests of the users.

DatabaseRelation: This module manages database query mechanisms.

Main Processor: This module gets the request from the I / O layer and organizes the processor layer to provide the required data to the clients.

ChatManager: This module provides a chat feature to the users.

VoiceCallOp: This module carries out the voice call functionality of the program.

3.3.3 Storage Layer

The storage layer deals with the file and database storage of the server.

Database: MySQL database is used for storing the information of all users in the related tables.

File Storage: File storage is used for storing the data in files. The data is kept as pieces of information inside folders in an organized way.

4. Consideration of Various Factors in Engineering Design

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

4.1 Public Health

Video chat features will be improved in HandsGiving to make it easier for the elderly to socialize. In this way, it will be aimed to keep the elderly psychologically healthy. Besides, people who have difficulties in meeting their food and other basic needs will be helped to be healthier.

4.2 Public Safety

Private information such as user credentials and location will be kept confidential and will not be shared by any third-party applications. In cases where identification information needs to be shared among users, it will be shared with the users' permission

4.3 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 assistance.

4.4 Global Factors

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

4.5 Cultural Factors

Cultural interaction will be created by making it possible for people from different cultures and different regions to help each other.

4.6 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 faster.

4.7 Environmental Factors

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

4.8 Economic Factors

HandsGiving will not make any profit for the volunteers using the application nor for our project group. People who want to use the application only need to register for free. Volunteers who want to help people in need can make specific 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

5. Teamwork Details

5.1 Contributing and functioning effectively on the team

Because a project has a specific end, it is important that every person contributing to the effort knows the desired result. By understanding the desired outcome, a project member can make better individual decisions and reduce confusion and rework. Moreover, each person is an essential piece in the overall project structure. Everyone should know their role and the roles of others. Each group member should communicate with each other by asking questions and giving updates about the parts that they are responsible for. Besides, every group member should adapt themselves to the changes that may possibly occur in different phases of the project. Therefore, they can maximize their contribution to the project. Apart from these, tasks in the project should be divided in a way that everyone can function effectively. This can be achieved by dividing the tasks according to the areas in which project members' knowledge is strong.

5.2 Helping creating a collaborative and inclusive environment

In order to have a collaborative and inclusive environment in the project, all team members should know the strengths and weaknesses of their members and act according to them. They should try to compensate for the weaknesses of their team members so that the flow of the project is not disturbed. Each group member should have a chance to be open-minded about their ideas and their behaviors to take a unique approach to each situation with a different perspective. That will affect the overall performance of the team positively.

5.3 Taking lead role and sharing leadership on the team

To encourage everyone to participate in leadership, we break the project down 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 the same workload. Thus, every project member will be aware of all the processes of the project. All team members will be in contact with each other at every moment of the project to complete tasks simultaneously. There is a dedicated GitHub repository to maintain the project's code base and progress. All team members have access to the archive. This archive will be shared with the project advisor and jury members if they wish.

6. 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: 15- Dec2020].

[2] Hürriyet." *65 Yaş Üstü Ne Zaman Sokağa Çıkacak?".*[Online]. Available: www.hurriyet.com.tr/galeri-65-yas-ustu-ne-zaman-sokaga-cikacak-41503410/2. [Accessed: 15- Dec2020].

[3] CloudFlare." *What is HTTPS?*".[Online]. Available: https://www.cloudflare.com/learning/ssl/what-is-https/. [Accessed: 16- Dec2020].

[4] Spring. " *Building REST Services with Spring*".[Online]. Available: https://spring.io/guides/tutorials/rest/. [Accessed: 17- Dec2020].

[5] MySQL. " *Why MySQL*". [Online]. Available: https://www.mysql.com/why-mysql/. [Accessed: 19- Dec2020].

[6] developers. " *Save data using SQLite*".[Online]. Available: https://developer.android.com/training/data-storage/sqlite. [Accessed: 19- Dec2020].

[7] JWT. " *What is JSON Web Token*".[Online]. Available: https://jwt.io/introduction. [Accessed: 20- Dec2020].