Bilkent University

Department of Computer Engineering

Senior Design Project

Project short-name: Charin

Final Report

Abdulkhaligli Mastan, Ahmadov Fuad, Mammadov Ismayil, Sadigova Shabnam

Supervisor: Prof. H. Altay Güvenir
Jury Members: Prof. Özcan Öztürk and Prof. Özgür Ulusoy

May 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 CS492.

Contents

1. Introduction

On different continents of the world conditions of different societies, groups of people or even animals might be in poor conditions. Benevolent people help those who are in need or in a bad condition. There are respectable foundations which try to provide this service to people who are willing to help others, such as Bill and Melinda Gates Foundation, William and Flora Hewlett Foundation, United Nations Foundation, Rockefeller Foundation. These foundations try to find more suitable profiles to help and they are getting funds from people who want to do charity. Donating or in other words helping those who are in need can be a tiring process and would need a more organized, transparent way. Also, sometimes it is really hard to reach foundations to donate money, or some people prefer to know where exactly their money goes and most foundations do not provide such information. We came up with the idea of Charin when we were thinking about such problems and realized that we can provide help for such foundations to gather more money and reach more people and also make it easy for people who want to help others. Furthermore, most of the foundations only accept money as donation; however, there are people who want to donate clothes, books, food and other necessities to live. We came up with the idea of Charin which allows people to seek help, to provide help for those who need it, bringing people together for charity purposes. Charin is a mobile application which is applicable for both Android and IOS devices. By using this application, people will be able to donate money easily, contact with people who are helping others and who need help, contact with foundations easily, create events for helping the ones in need, create groups to reach out to people easier. We understand that trusting third parties can be hard for some people, and our application is meant to solve that problem since we are not using any third parties to manage the transactions. However, we realize that it is possible that some would try to use others' good intentions. We want our users to be sure that their help will reach the right people. For that we are going to work with organizations and foundations in order to provide money transactions. Users do not need to donate money for specific people, they can also donate money to foundations and they will arrange where to spend the money. Charin allows only authenticated users to create events for money donations, and the process of creating an authenticated account is done by our

customer services. Organizations will be able to create authenticated accounts by calling or emailing to our customer services, and we will provide them the account. Although unauthenticated users will not be able to create events to collect money donations, they will be able to open other types of events such as social events for gathering and collecting help for someone and delivering the needs to them. Both authenticated and unauthenticated users are able to create groups and become a member of groups created by others. Users are able to follow each other to be able to stay in touch and they can contact each other through the chat feature of our application. Users can post on their timeline and see the posts of users they are following on their home page. There is a discovery feature where users will be able to discover events, groups created nearby, and find users by using their username.

2. Requirements Details

2.1 Functional Requirements

2.1.1 Unauthenticated Users

- Users are able to register to the platform by email.
- Users need to specify a username, password, first name, surname, birthdate, gender, and location when they are signing up to the application.
- Users have their profile which will contain information of the user including followers, followings, events, groups and posts of the user.
- The users are able to modify their user information such as first name, surname, gender, location, birthdate.
- The user is not able to change his/her username and email.
- The user is able to change his/her password.
- Users are able to create events by entering needed information about the event.
- Users are able to enroll in the event.
- Users are able to create groups.
- Users are able to add members to groups if they are admin of the group.

- The user is able to assign other members as an admin to the group if the user is an admin of that group.
- The user is able to delete the group if he/she is the admin.
- Users are able to share posts in the group.
- Users are able to send messages in group chat.
- The user is able to ban users from the group if he/she is the admin.
- The user is able to ban users from group chat if he/she is the admin.
- Users are able to report other users if they are abusing them in the app.
- Users are able to follow other users.
- Users are able to unfollow users that he/she followed earlier.
- Users are able to see events that users he/she follows shared.
- Users are able to discover nearby the most enrolled events.
- Users are able to send private messages to the other users.
- Users are able to view other users' profiles.
- Users are able to search for an event, a group, a user.
- The user is able to leave a group when he/she wishes to.

2.1.2 Authenticated Users

- An authenticated user needs to contact the customer services in order to get an account.
- An authenticated user is able to do everything that an unauthenticated user can do.
- An authenticated user is able to create an event for money donations unlike an unauthenticated user.

2.1.3 Donations

- The user is able to donate on the events which require money donations.

2.2 Non-functional Requirements

2.2.1 Usability

- The application is designed in a way that has user friendly design so that people from every age are able to use the application.
- The system is designed simple enough to be understood by everyone.
- The application is free to use.

2.2.2 Security

- The system is designed secure in order not to disobey security rules. None of the users are able to reach restricted information about another user.

2.2.3 Reliability

- As long as the mobile device is connected to the internet, the application is available to use for the users.

2.2.4 Extensibility

- The application is designed in a way that allows the designers to add more features or update some features in the future as desired.

3. Final Architecture and Design Details

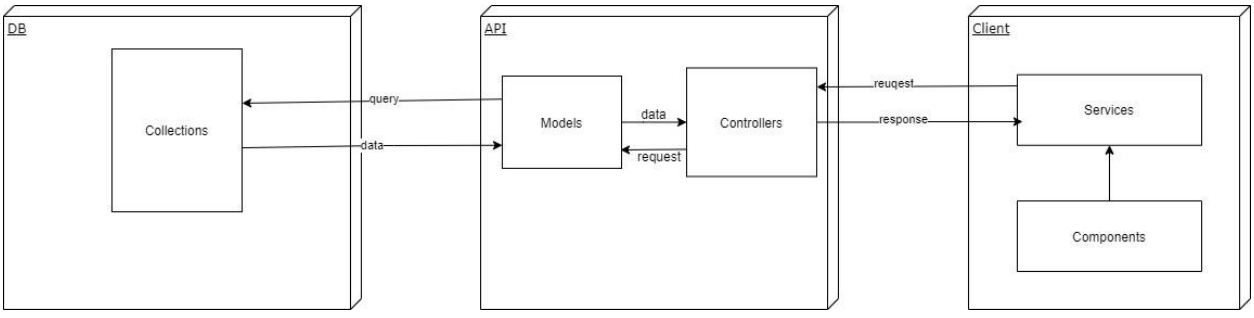

Figure 1. System Architecture

The diagram in Figure 1 illustrates the final architecture of Charin. There are three main parts constituting the system: DB, API, and Client. DB (database) is responsible for data management.

The information related to the events, posts and users is kept here. On the other hand, API handles the functional requirements of the system. API includes two parts: models and controllers. Here, “models” interact with the DB -using custom queries- to retrieve data. For example, by using models, API can retrieve the event for a particular organization and show information of this event to users. On the other hand, controllers receive requests from the client and return the appropriate response by fetching data via models. To the right part of the diagrams the “client”, which is responsible for sending requests and displaying the responses. This part of the system allows users to interact with Charin.

3.1 Controllers

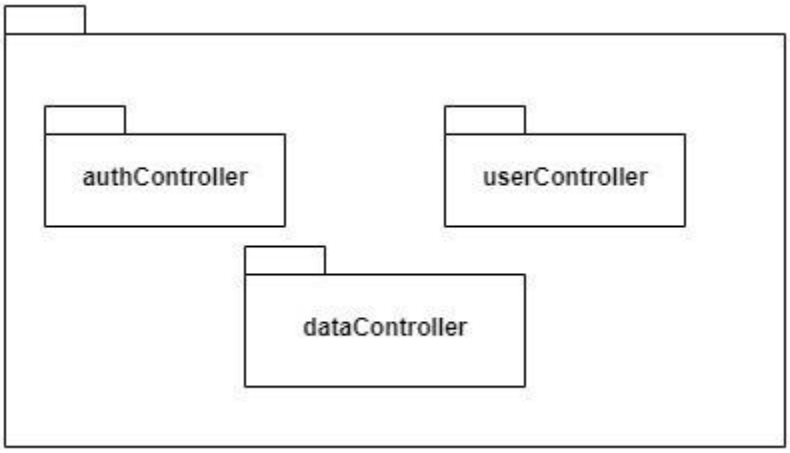

Figure 2. Controllers Diagram.

Controllers handle all the requests that come from the client-side. The requests come as a JSON file and the appropriate response is also sent back to the client-side as a JSON file. In our system, there are 3 main controllers handling various different requests. The data controller is responsible for handling the CRUD operations that are done on the event collections of the database. Some example usages of the event controller are: The controller fetches events according to the preferences of the user that were sent from the client-side, inserts events to the database. The authController handles requests such as registration, logging in/out. This controller is also responsible for verifying and validating users. The userController is responsible for fetching/updating user information, handling social interactions between users such as friend requests, searching for users and fetching the participated events of users.

3.2 Models

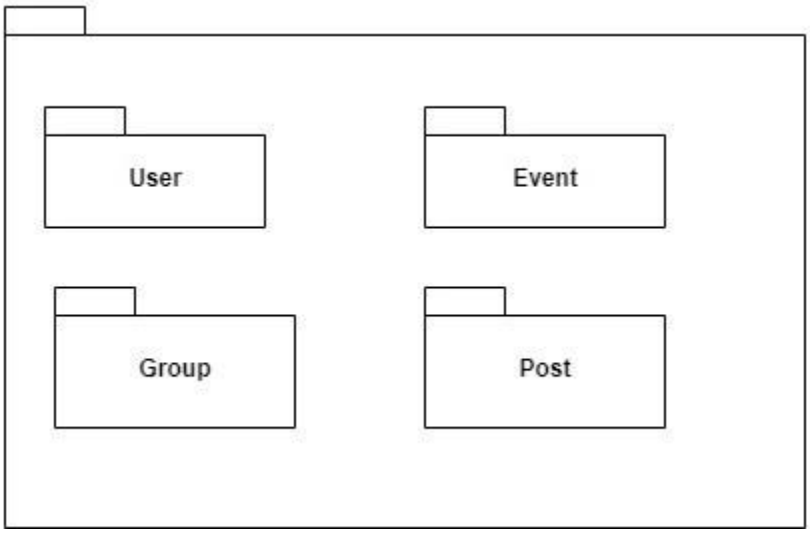

Figure 3. Models Diagram.

Models handle the data flow across DB and controllers. In other words, the controller is not permitted to access DB directly; it has to use “models” to access and modify the data in DB. As a result, models are split into four: User, Event, Group and Post. Each of these parts handles the DB operations on their corresponding parts. For instance, a “user” insertion to the DB is handled by the user subcomponent. Similarly, an event retrieval from the DB is the task of the event model.

3.3 Services

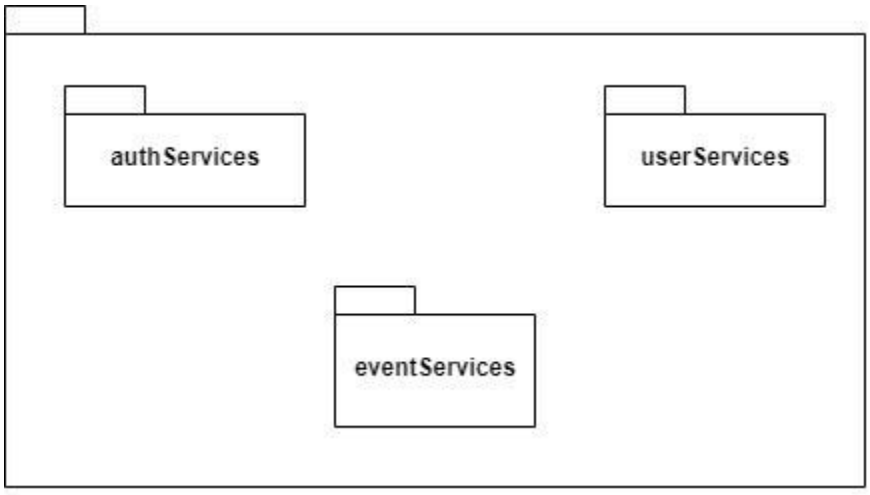

Figure 4. Services Diagram.

Services are responsible for sending requests to the appropriate endpoints of the API. As a result, there are three parts of this component: authServices, userServices, and eventServices. authServices, as its name would suggest, handles the requests that are related to authorization. Some example requests, here, can be login, sign up and sign out requests. On the other hand, userServices is responsible for sending the requests regarding “user” functionalities of the system (i.e. social aspects). For instance, the requests for accepting friends, accessing profiles of users, etc. are some of the requests that are handled here. Similarly, the last one, eventServices, deals with the event-related requests.

3.4 Components

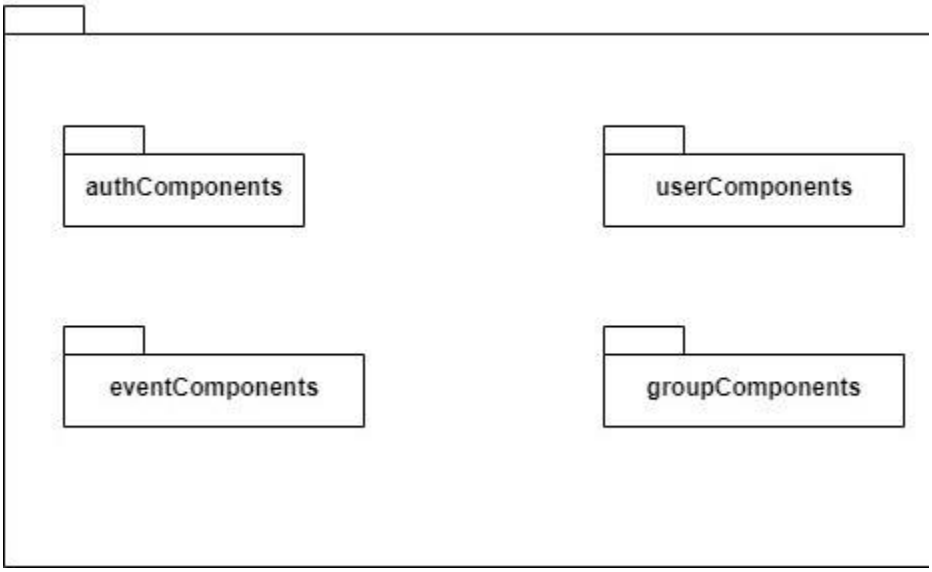

Figure 5. Components Diagram.

The components package resides within the client side of the project. These components are used in different screens of the project. Firstly, authComponents consist of a login system which user logins his/her email and username, then the system generates a token that redirects the user to the main page of our application. Besides, userComponents is mainly for displaying the profile of a user. GroupComponents displays a group that the user selected to join. Users will be able to display posts and events shared by group users. EventComponenets is for displaying information about events a button for donation, like, comment will appear.

4. Development/Implementation Details

Charin is developed for both Android and IOS devices and to be able to provide this, we have used React Native. React.js is used for the frontend of the application alongside with its features. We have used Node.js for the backend of the application. Express and mongoose are used to be able to connect the application to the database. We are using MongoDB as a database for our project. Mongo DB provides a NoSQL database which is placed on AWS cloud services.

5. Testing Details

The application was tested during the implementation process. We used both test-driven-development (TDD) and unit testing. In test-driven-development (TDD), we started with writing simple test cases before implementing the features. After finishing the test cases, we wrote codes good enough to pass the green line of the test cases. TDD was the first step of the testing and implementation process of the project. After implementing the features, we used unit testing to test the implemented parts. After finishing all steps of the testing and implementation, we refactored the code. Refactoring included some minor changes and user interface design of the project. Because we implemented some parts of the project separately, we had to do some extra tests in the process of merging the parts.

6. Maintenance Plan and Details

The application needs to be available for the users all the time. To be able to provide all time access to the users we have used AWS cloud services. However, we are using a free trial for now and we have to change the payment plan when the number of the users using our application increases. For the maintenance of our application, we will switch our free trial accounts and set a payment method. We will also keep track of the performance of the application and interfere when an error is found. There might be problems which we would not be aware of and for such cases, we are providing an email address so that users can contact us and report the problem

they are facing. For now, all of our group mates will have access to that email address and whoever is available at the time of error report will try to fix the problem.

7. Other Project Elements

7.1 Consideration of Various Factors

When this project was started, one of the initial aims was making the system available to everyone regardless of location, age, gender, nationality, income. To be able to make the system available to everyone, we are using MongoDB as a database which is using AWS cloud services. Using cloud services allows us to provide availability to the application. For that reason, we did not use a local database. Another consideration was not discriminating against people by their nationality and for that sense we are not asking people to provide information about their nationality. We have designed our project to be used by people from every age and regardless of their gender and income. Considering that Charin is an application where people are donating necessities for the ones in need, some people might think they need to spend too much money if they use our application. However, we have designed the application in a way that, regardless of their income, everyone can use the application and support others. There are various ways to help someone other than donating money for them. In our application users can help others by sharing the event on their timeline or groups, by creating events to help someone find help faster, or donating their books, clothes which they do not need anymore.

Another factor we consider was accessibility of the system. We wanted to prepare a system that will be accessible anywhere, anytime for the users. We chose to implement it as a mobile application for that reason. As we all know, people carry their phones with themselves wherever they go. However, carrying a laptop is harder and, in that sense, desktop applications would not be accessible everywhere. Using cloud services also affects accessibility of the system in a positive way.

7.1.1 Public Health

Charin is designed in a way that does not affect public health and is not affected by public health.

7.1.2 Public Safety

In our application all the information is kept secret unless the user chooses to appeal to them. Information about user credentials are kept in the database and they are only available for the user himself/herself. In the process of donating money, we do not keep the card information for safety reasons. For these reasons, Charin does not affect public safety.

7.1.3 Public Welfare

Charin is an application which is private to its users and will be used by their own choice. In that sense, Charin is not affected by public welfare.

7.1.4 Global Factors

Global factors which might affect our system can be situations where people are not able to establish public events where they gather to help others. This can happen in case of pandemic, epidemic, natural disasters. Also, economic crises can affect the usage of our application.

7.1.5 Cultural Factors

Charin is an application where people have a chance of meeting with people from different countries, nations thus people will be exposed to various cultures. This might affect our application; however, cultures can be affected by Charin too. As mentioned before, people meeting with new cultures will expand their thoughts and actions. Charin can be welcomed in some cultures but refused in others due to cultural differences. We are expecting Charin not to be successful in some countries where it is not common to donate to help others as much as it will be in other countries. Lack of helpful organizations, foundations in a culture will also affect the usage of Charin in that area. As above mentioned, Charin can also introduce the culture of doing charity to some cultures as they will get exposed to it.

7.1.6 Social Factors

Charin is designed in a way that it can be used by every social group. As stated above, people from different age, background, social groups will be able to use the application equally. To be able to provide this, we have made the application with a user-friendly interface so that it can be used by everyone without facing any hardship.

7.1.7 Environmental Factors

Charin is using the location of the user in order to provide him/her with clearer data about events happening nearby and groups created in those areas. Even though Charin is embracing the whole world, because of some environmental factor we might not be able to get the location of the user using API. These environmental factors can be high mountains where there is no signal, or heavy fogs which can cause loss of signal on the mobile phone.

7.1.8 Economic Factors

Economic factors affect our project in a way that because of the low budget we cannot expand our project more right now. Which will be handled by time.

7.2 Ethics and Professional Responsibilities

During the design process of this application ethical and professional responsibilities were considered in every step. We have focused on a design which is user-centered. We have implemented our application as a cross platform application meaning that it works both on IOS and Android devices. Our ethical responsibility is securing the user data so that everyone can use our application without having fear of security of the system. We have tried to decrease the data loss to the maximum extends. It is our duty to provide a system which is reliable. To provide reliability to our project we have decided to have two kinds of users which are authorized and unauthorized users. We wanted to minimize the chances of any forgery in requiring donations, so that users would not donate to fake events. For that we have been able to create events based on money donation only for authorized users. Authorized users need to register to the system by calling or emailing the customer services of Charin. In this way only organizations and foundations will be able to have authorized accounts by providing a proof as asked and they will be able to gather donations.

7.3 Judgements and Impacts to Various Contexts

7.3.1 Global Context

As we mentioned above, epidemics and pandemics can affect our application in a way that people will not be able to gather and have events. However, we have taken precautions for such situations such as our application does not only require gatherings, people can help others from far away. They can donate money, send other donations by cargo etc. Another global factor which can affect our system is economic crises and our solution for that is the application does not require everyone to make donations, there are other ways which allows users to help others such as raising awareness.

7.3.2 Economic Context

While implementing the design we have used free platforms since we did not have enough budget to spend. However, these free platforms are not totally free. They are providing limited usage for free and after finishing the free trial, we have to pay in order to get these services. In that sense, we will face economic problems shortly. We have decided to solve the economic issues by getting advertisements in the application. In this case, we will earn enough money to spend on the maintenance of the system.

7.3.3 Environmental Context

Unfortunately, we are not able to offer a solution for environmental factors which can affect our system. Since we are using Google API to extract the location of the user, it is more of Google services responsibility to expand their coverage.

7.3.4 Societal Context

Our application provides suggestions to the users which are close to their interest areas. In that case every user will be able to find events and groups which are in the scope of their interests and they will meet with people with similar interests. The application Charin grants its users to follow other users and contact them via chat feature.

7.4 Teamwork and Peer Contribution

Abdulkhaligli Mastan:

- Contributed in Low-Level Design Report.
- Implemented chat feature of the application.

Ahmedov Fuad:

- Contributed in Low-Level Design Report.
- Contributed in the Final Report.
- Implemented Group and Post feature of the application with frontend and backend.
- Implemented Group and Post model.
- Implemented Image upload system.
- Researched Hyperledger Composer for Blockchain Technology.

Mammadov Ismayil:

- Contributed in Low-Level Design Report.
- Contributed in the Final Report.
- Implemented Event, Discovery and Activity features of the application with frontend and backend.
- Implemented Event and Donation model.
- Implemented Payment for donation.
- Researched Hyperledger Fabric for Blockchain Technology.

Sadigova Shabnam:

- Contributed in Low-Level Design Report.
- Prepared the Final Report.
- Implemented Login and Sign up features of the application with frontend and backend.

- Implemented User model and Post model for user posts on profile page.
- Implemented User Profile and features related to that with frontend and backend.
- Implemented user posts for the profile page.
- Redesigned the screens existing on the application.

7.5 Project Plan Observed and Objectives Met

After overall consideration of the process of design and implementation of the project, it can be said that we have met with our objectives. Although there were some changes in the architectural design of the project, we believe that we have met with the promises of our project. Due to some worldwide issues, we have had problems in meeting some deadlines which had been set for completing some parts. But, even with a few days of delay we have been able to complete the tasks we were responsible for.

7.6 New Knowledge Acquired and Learning Strategies Used

Some of our group mates were already familiar with Node.js, React.js and Mongo DB. The ones who did not have experience with these learned them for implementing the application.

Our project was designed based on Blockchain technology to provide transparency in the payment and none of the members of our group had knowledge or experience with the technology. To be able to implement the technology into our application, we learned Blockchain. However, after learning the Hyperledger Fabric for Blockchain technology and considering our application we decided that this technology is not a good fit for us. The logic of the Blockchain technology is to create business networks between different peers and these peers share information ledgers among themselves. This ledger updated if more than half of the peers accepted the information. In our system there is one donor and one organization who receives the payment. In this case it is meaningless to use blockchain technology for transparency. So, we decided to move forward and not use Blockchain in Charin.

We have used Stripe for the money donation feature of the application. Stripe was a new technology for us to learn and use. We are using it to be able to get donations from credit cards provided by users.

Another new technology which we have learned and used in this project is Machine Learning. We are using Machine Learning to discover features of the application.

To be able to learn the new technologies we used documentations and books available on the web. We also watched tutorials in order to understand the technologies better and to be able to apply them in our design. “EDx” and “Udemy” was used as a platform for finding tutorials. Also, the most effective learning strategy for us is starting the implementation of the technology with simple examples, and after understanding it, applying our knowledge to the implementation of our project.

8. Conclusion and Future Work

Charin is an application which was aiming to ease the process of helping people in need, to provide clear information to donors, and being able to reach out to more people by cooperating with organization and foundations. We believe that we have reached the goals we had set before starting this project. Even though we have made changes in the architectural design of the, they were made to make the application more logical. In conclusion, we are satisfied by the work done. We have designed Charin in a way that we would be able to add more features in future. Our initial purpose for the future is keeping the application working and attracting users into using it. After that we are planning to release new features according to the needs of the users.

9. User Manual

9.1 Login Page

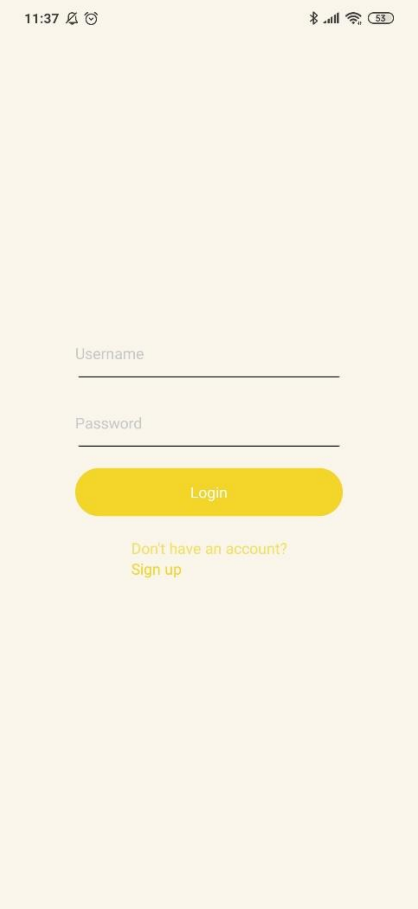

Figure 6. Login Page.

This is the first page that the user sees when he/she opens the application. If the user already has an account, he/she can enter by using his/her username and password and pressing on the Login button. If the user does not have an account yet, he/she can press on “Sign up” text and the user will be directed to the Sign-Up page.

9.2 Sign Up Page

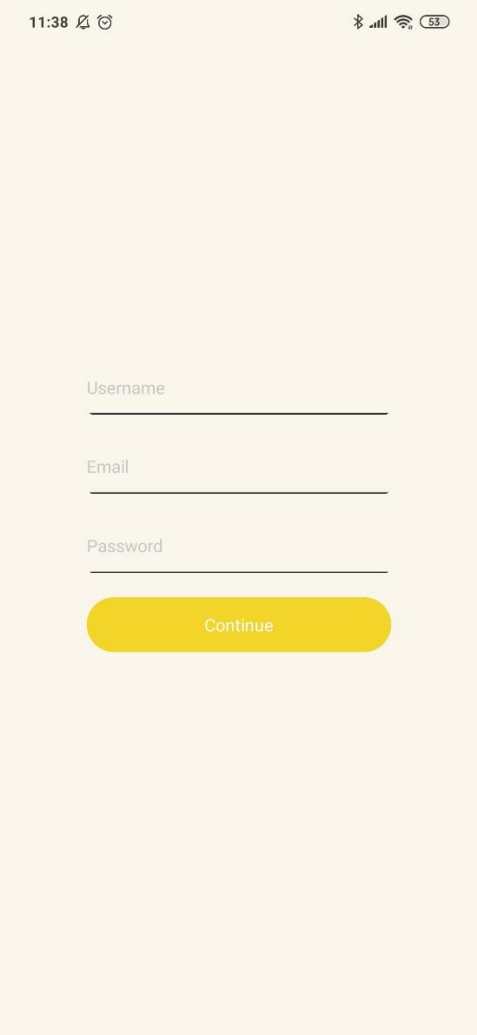

Figure 7. Sign Up Page.

When the user clicks on the “Sign up” text in the Login page, he/she is directed to this page. In this page, the user is asked to specify a username, email and password and then press on the “Continue” button. However, there are some restrictions for the required fields. Empty inputs are not accepted. The username should be all lower-case letters, numbers are accepted but only “-” and “_” are accepted as symbols. User should remember the username he/she chooses since he/she will need it for signing in to the application. The password should be at least 8 characters and it should include numbers, letters, uppercase letters. Users are expected to enter a valid email address. Otherwise the signing up process will not be proceeded.

9.3 Complete Sign Up Page

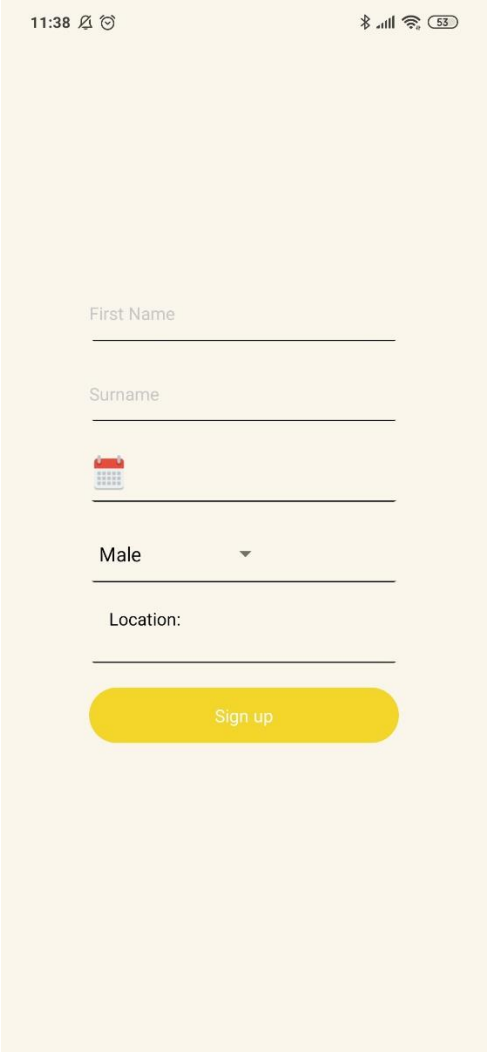

Figure 8. Complete Sign Up Page.

After pressing the “Continue” button on the Sign-Up page, the user will be directed to this page to complete the steps of the signing up to the application. All the fields in this page are required in order to complete the process. The first name and the surname of the user cannot include any number or symbol. By clicking to the date picker, the user can pick his/her birthdate. The application provides three options for gender and they are “Male”, “Female”, “Prefer Not to Specify”. Users are required to choose one of the options for gender. The location of the user is taken automatically when the user clicks on the location input area. After completing these steps,

the user will press on the “Sign up” button and he/she will be directed to the Login page to be able to sign in to his/her account.

9.4 User Profile Page

Figure 9. Profile Page.

This is the profile page for the user. Profile page displays the first name and surname of the user and location of the user. Here we have displayed the count of followers, followings, events, groups and posts the user has. When the user clicks on the “Followers”, he/she is directed to another page where the list of followers is displayed. When he/she clicks on the “Following”, the user is directed to the page where the users followed by that user are displayed. When the “Events” clicked, the list of events the user is attending is displayed. When the “Groups” is pressed, the user is directed to a page where he/she can see the list of groups he/she is a member of. User posts are displayed on the Profile page. When the user clicks on the three dots button,

a pop-up menu offers 2 options to the user. Those options are Edit Profile and Logout. When the “Logout” is selected, the session ends and the user is directed to the Login page. When the “Edit Profile” is selected, the user will be directed to a page to update his/her information.

9.5 Edit User Information Page

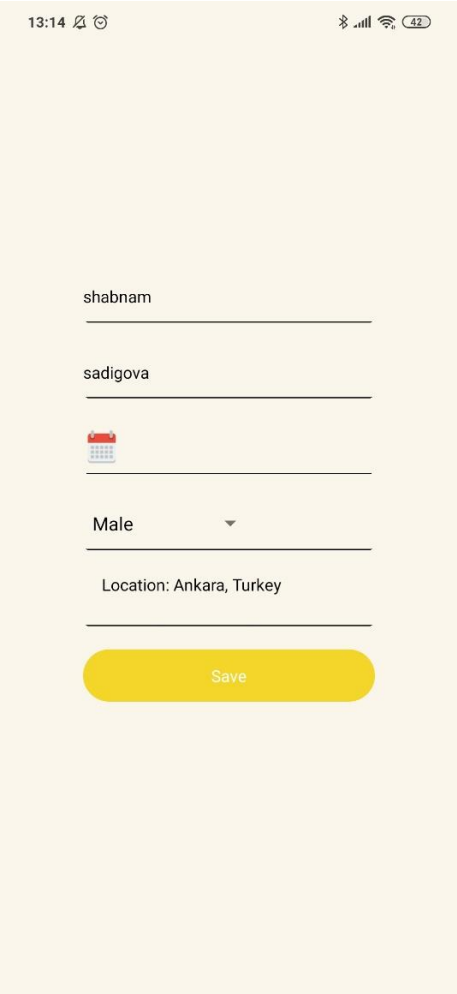

Figure 10. Edit Profile Page.

This is the Edit Profile page where the user is able to edit his/her information. After updating the wished fields, the user will click on the “Save” button and the information will be updated and the user will be directed back to the Profile page.

9.6 Other User Profile Page

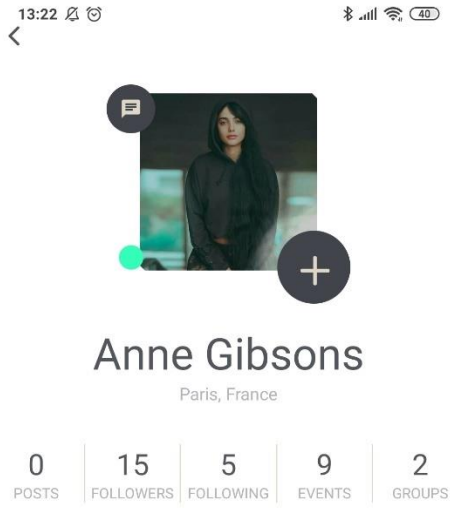

Figure 11. Other User's Profile Page.

This is how a user sees another user's profile when he/she is not following him/her. As we can see from the figure, there is a plus icon. When it is clicked, a follow request is sent to that user. There is a message button at the left top of the profile. By clicking that button, the user is able to start a conversation with that user.

9.7 Other User Profile Page Following

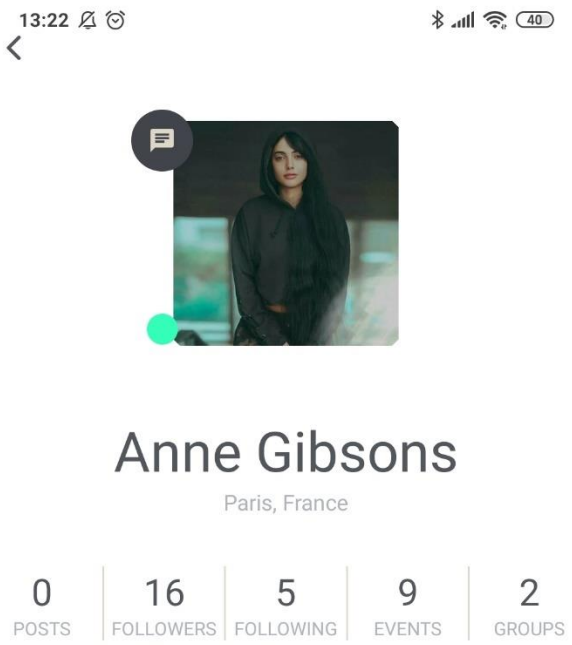

Figure 12. Other User's Profile Page After Following.

This is how another user's profile looks like after she/he has accepted the follow request.

9.8 Home Screen

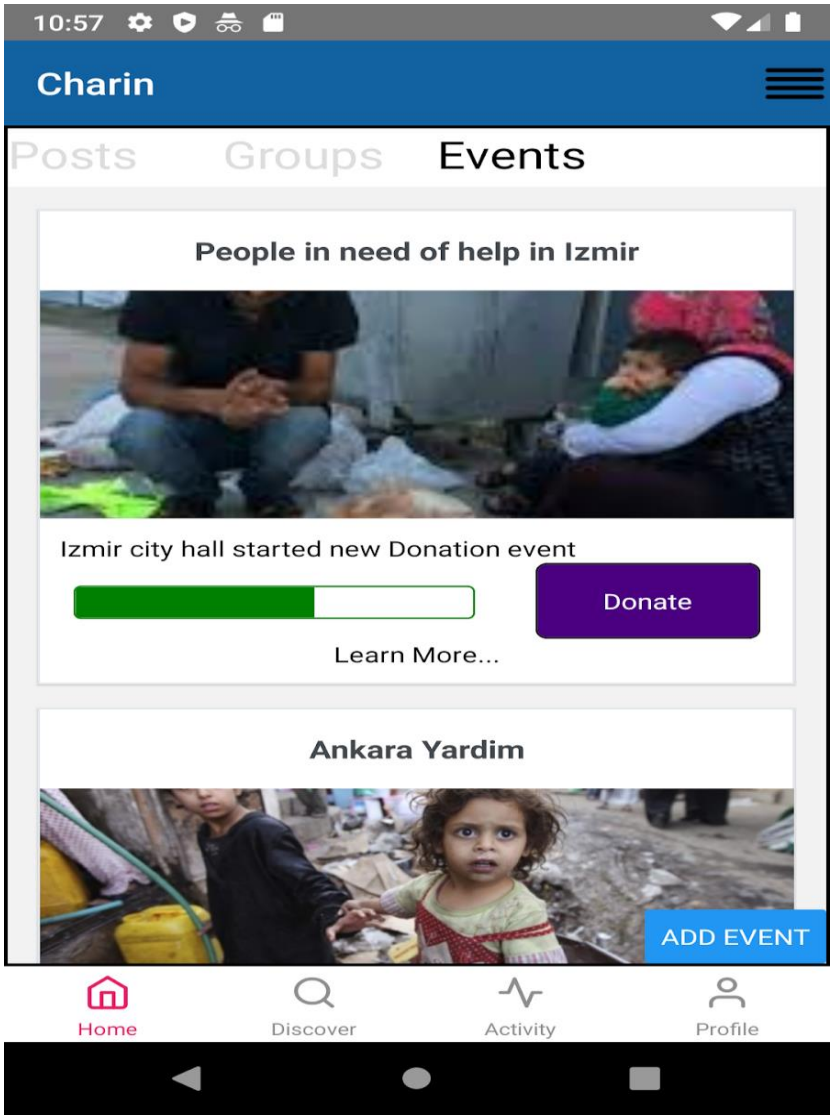

Figure 13. Home Screen.

After successful login, User encounters this screen. If a user does not donate any event, the home screen will be empty. In the home screen, there will be events only donated by the user. Users can see progress of the donation for the specific event. User can click the Donate button and after that, he/she can enter his/her card number to donate. For organizations, there will be an add event button to add events for donation and/or participation. For users, this button will be created only for participation-purpose events.

9.9 Input Card Information Screen

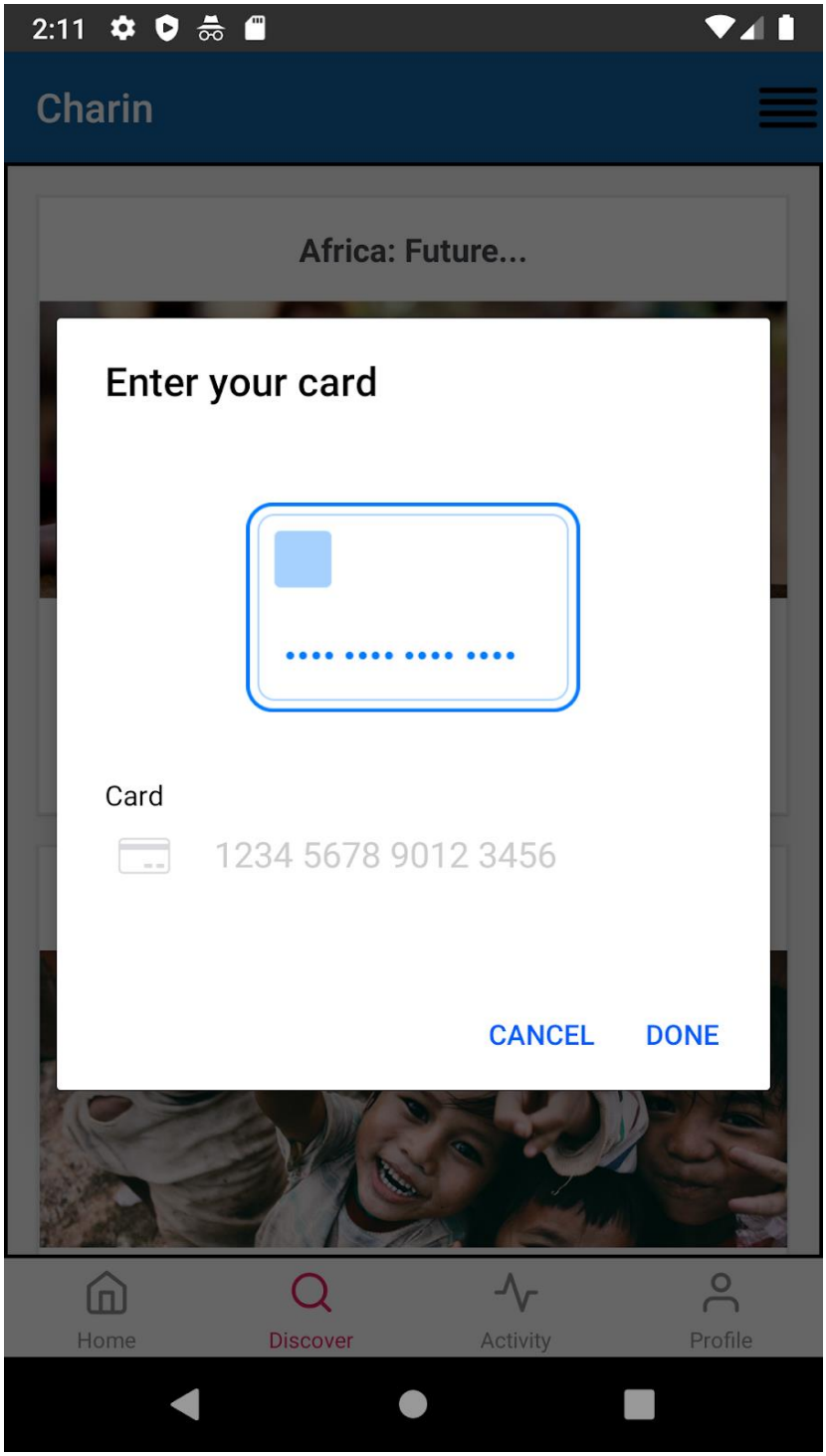

Figure 14. Input Card Information Page.

From this screen, users can enter card info and donate to specific events.

9.10 Discovery Screen

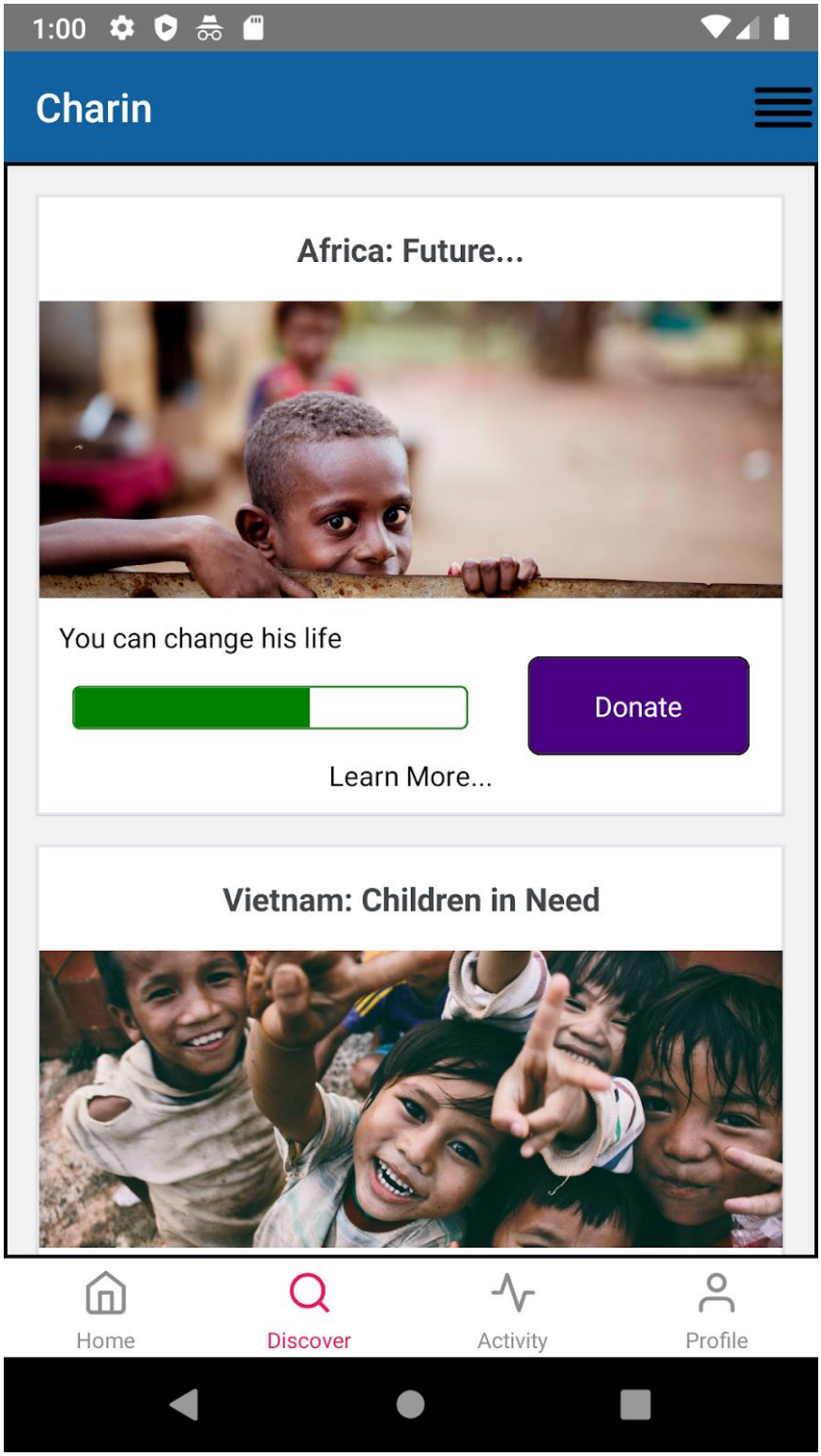

Figure 15. Discovery Page.

In this screen, users will discover different events. Firstly, these events will be given default. Afterwards, according to the user's interaction with different events, the discovery page will use this data to generate appropriate events for the user.

9.11 Add group Screen

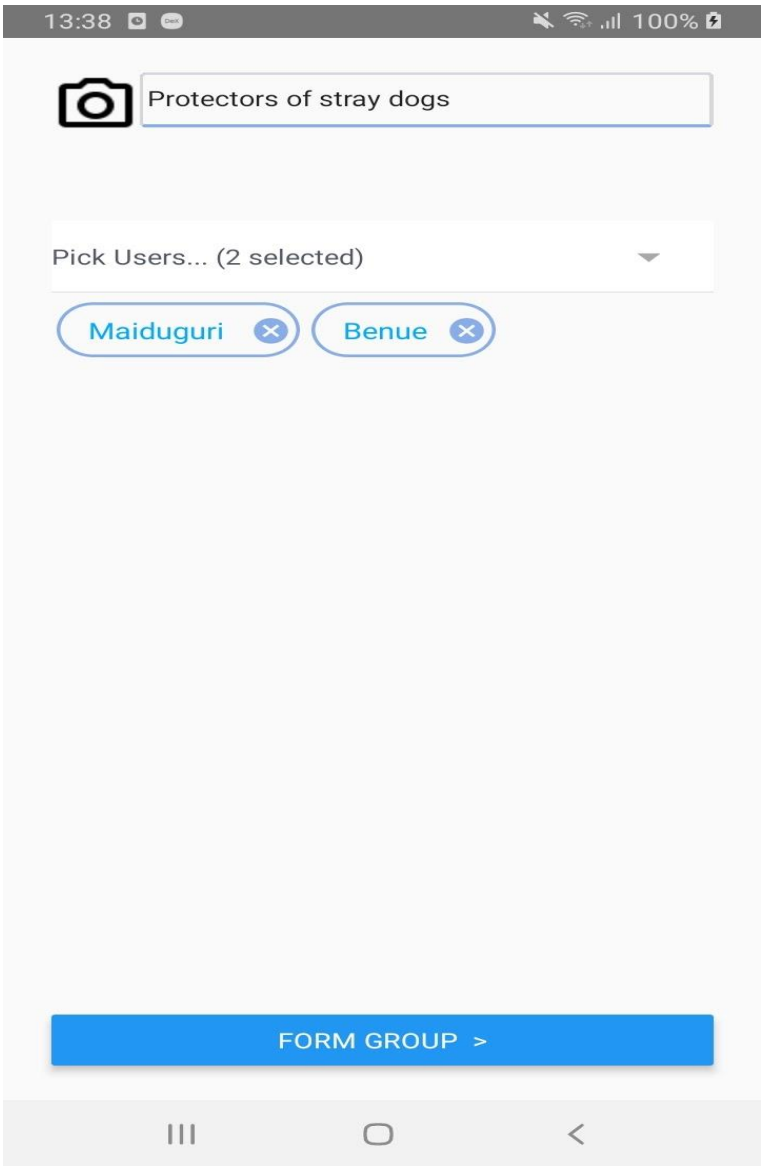

Figure 16. Add Group Page.

In this screen, a user will be able to create a group with specific users from his followers by his/her choice. Group will have its own name, photo.

9.12 Share a post screen

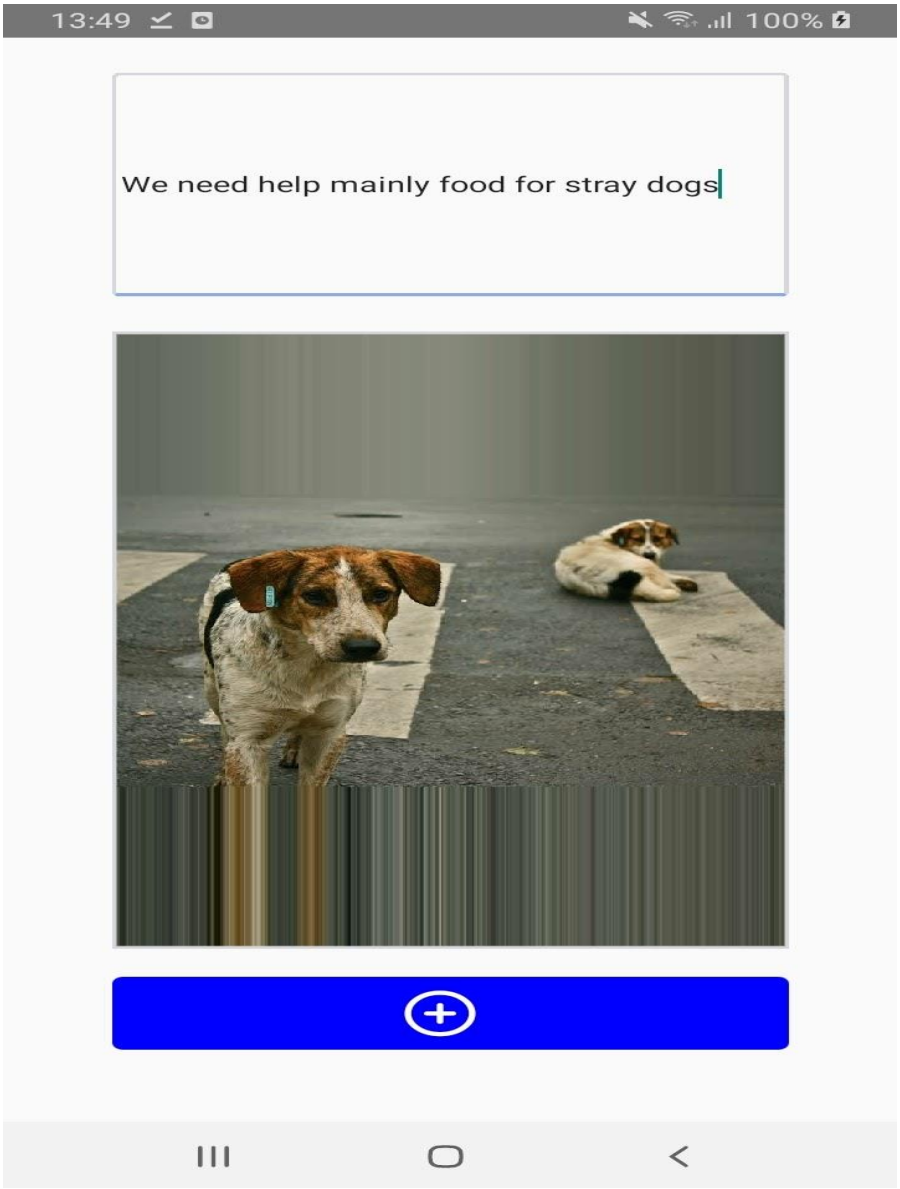

Figure 17. Share a Post Page.

This page demonstrates a post sharing screen, which a user will be able to share post with its description and image. User will be able to share a post on group or with his friends from home feed by his/her choice.

9.13 Group Page

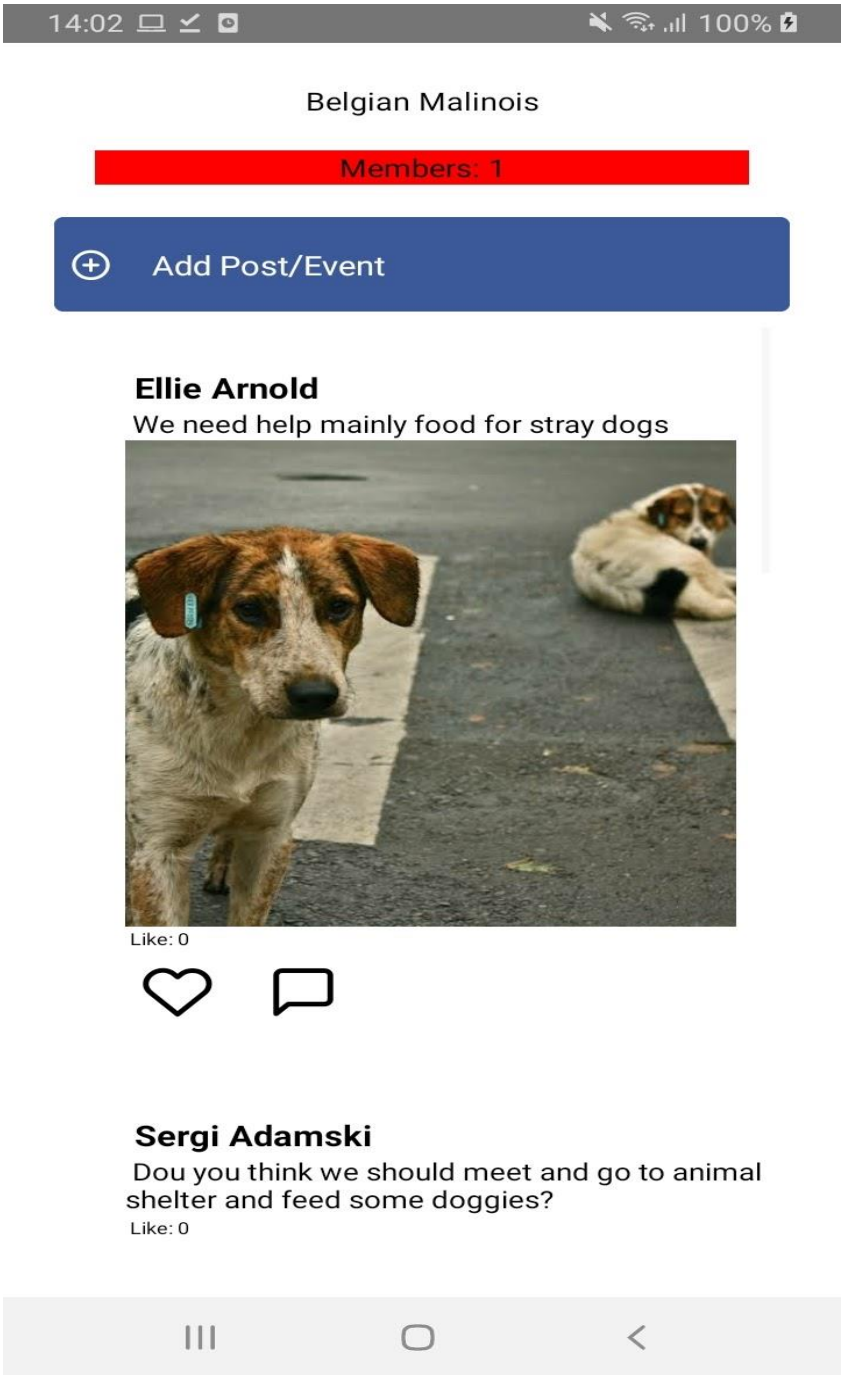

Figure 18. Group Page.

In this screen group's main page is demonstrated and users will be able to share posts and direct events to this group. In this page users will be able to communicate by groups own chat.

9.14 Chat Page

Figure 19. Chat Screen.

When user enter chat, system automatically send “Welcome” message for every user. We use 12-hour clock system as you see near of name of message sender. There are 2 buttons for users which are “Send” and “Send Location”. When new user joins chat system automatically recalls “A new user has joined”.

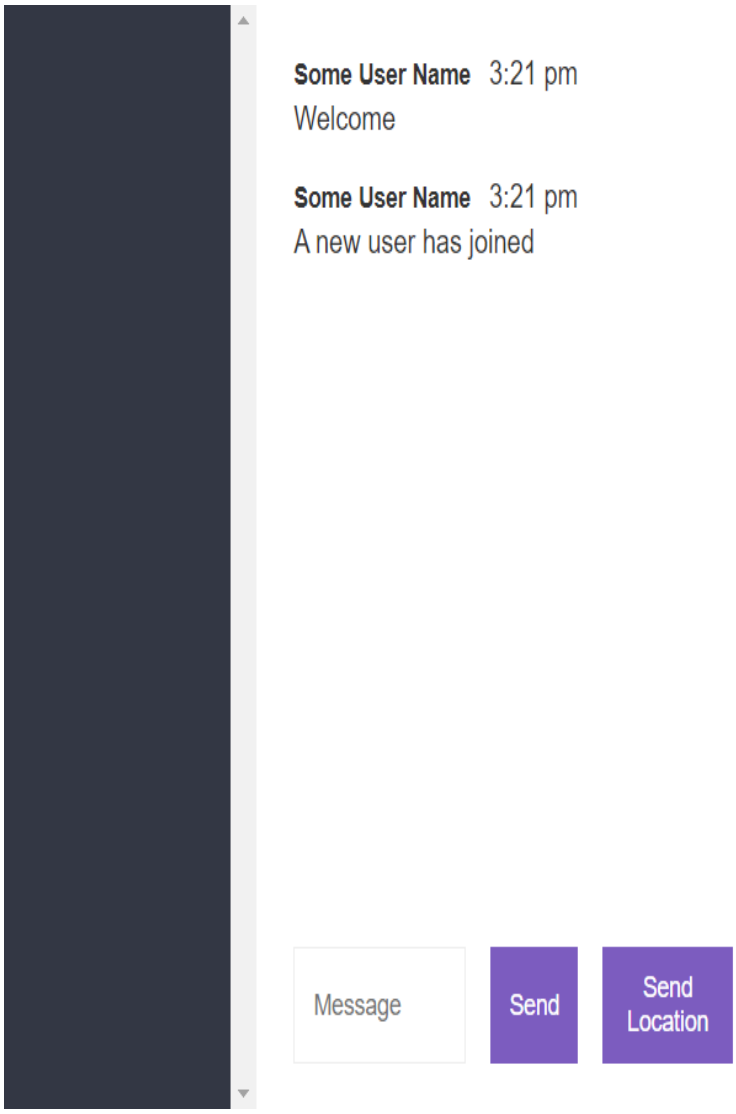

Figure 20. Chat Screen.

When user wants to send his/her location he simply taps Send Location button and after that location send with the name of “My Current Location”

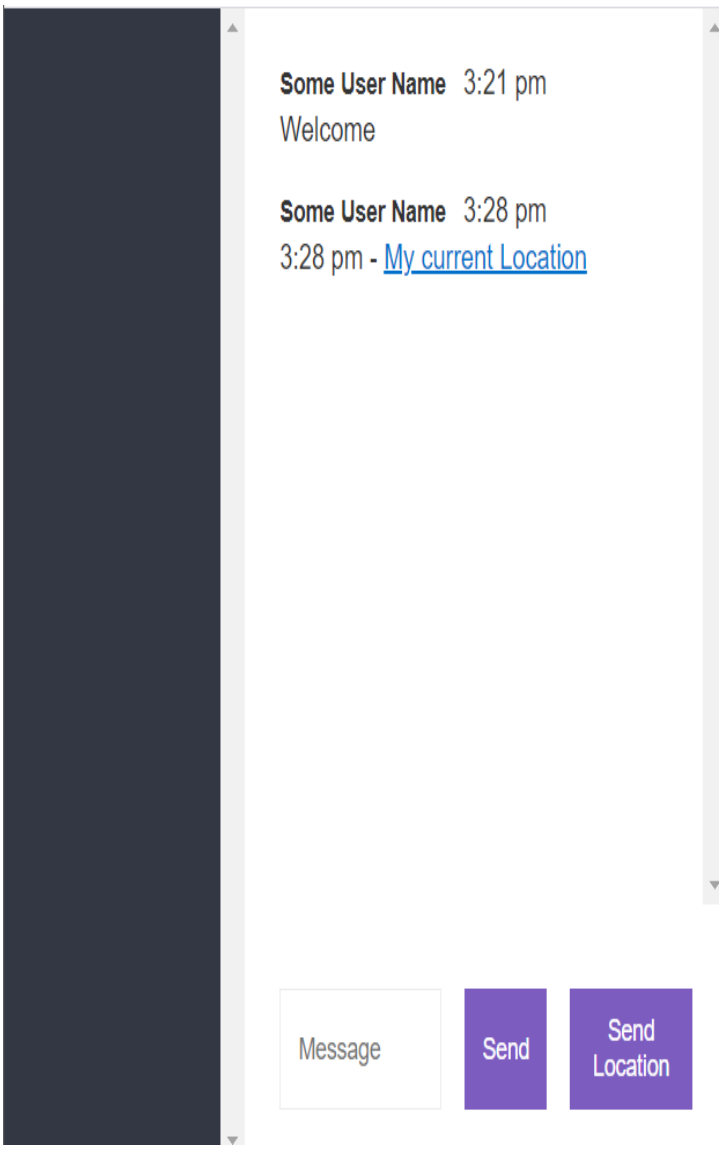

Figure 21. Chat Screen.

When receiver tap to current location, browser automatically open google maps and show user's friend location and his coordinate.

Figure 22. Chat Screen Map.

10. References

[1] Iansiti Marco and Karim R. Lakhani. “The truth about blockchain” Harvard Business Review. https://enterprisersproject.com/sites/default/files/the_truth_about_blockchain.pdf. [Accessed: Oct 25, 2019]