Bilkent University

Department of Computer Engineering

Senior Design Project

Project short-name: Charin

Analysis Report

Abdulkhaligli Mastan, Ahmadov Fuad, Mammadov Ismayil, Sadigova Shabnam

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

Nov 11, 2019

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 CS49

Table of Contents

1. Introduction.

Blockchain is becoming famous technology nowadays. It was introduced in October 2008. First application of blockchain was bitcoin [1]. Blockchain technology allows to monitor transaction of money according to hash codes of the transacted money. It gives possibility of transparency at the payments. If we consider different social events that contains claims for donations to different people or places, transparency becomes crucial thing. Nowadays there are a lot of social awareness acts which people generally share these acts and give comments about these acts. However, in donation cases, there are a lot of frauds which you cannot know in which account your money combined. In Charin application, developers tried to solve this problem for society. The purpose of Charin application is to use Blockchain technology in social charity application. With the help of this application people can share the event and get support from their followers. In this application, people become famous with the help of their donations.

2. Current Systems

Nowadays there are many applications for people who want to help others. There are two kinds of charity applications, first type is charity organizers that establishes required environment, and arranges a meeting between who wants to help with the ones that need help. Second type is donation applications that collects money from donors and give collected money to organization or person. Problem with this kind of donation systems that donors cannot trace their money, and make sure that money has reached to desired destination.

3.1 TDP

TDP is a social awareness project in Bilkent University [2]. TDP has 19 projects that aims to help to different groups of people which needs support economically or psychologically. TDP also supports students to socialize in the act of benevolence [3].

- This community is only supported at universities.
- They don't’ have mobile app to reach out to people.
- They help different groups of people across the country.

3.2 Globalgiving.com

Globalgiving.com collects different projects from different spots of the world, and users can donate to that projects, these projects mostly involve helping people who seeks support.

- There are many different projects around the world, user can choose to help which one she/he wants.
- User cannot trace money that is donated. And cannot make sure money reached to target.

3.3 Alice.si

Alice.si is a company that focuses on applying blockchain on finance [5]. They also offer blockchain system in donations.

- Alice.si allows users to donate and track their payment.
- Alice.si only allows donation of money.
- Alice.si is not a charity app, it only guarantees transparency in payments.
- Alice.si is not social media.

3.4 GiveTrack

Give track is charity application that contains different events which need donation. Philanthropists can track their donations from the website.

- Give track does not have social media side.

3. Proposed System

3.5 Overview

Charin is a social platform that gives chance to people that want to support others in both psychological and economic way. Charin gives chance to philanthropists to donate money and participate in events.

Idea of Charin comes from the fact that there are many people who are in need of better conditions in Turkey. In Turkey there are many help centers and foundation but, none of them is transparent enough. Besides, with Charin users will be able to participate in events, and observe.

Charin aims to support those people who needs support both in economic and psychological way. Users can donate money in our system and they will be able to trace their payment. Besides, they will be able to participate in an event by visiting the specified location.

Charin will be different from other charity foundations, and the most crucial aspect is transparency which donors will be able to donate and even trace the donation and see where it is spent. Transparency will be guaranteed by the usage of the blockchain. On the other hand, event report will be shared to demonstrate type of the support and quantity of the given products, food, and etc. is given.

Ranking system will be implemented in the system to increase support frequency. Besides, users will be able to share events or their ranks with their friends on other social media.

3.6 Functional Requirements

3.6.1 Social Platform

- Users should be able to register to the platform by email, twitter, instagram. And verify themselves by email verification.
- Users will have their profile which will contain information of the user including rank of the user.
- The user should be able to modify their user information.
- Users should be able to share their profile on other social media.
- Users should be able to create events by entering needed information about the event.
- Users should be able to enroll to the event.
- Users will be able to create groups.
- Users will be able to add members to groups if they are admin of the group.
- The user will be able to assign other members as an admin to the group if the user is an admin in that group.
- The user will be able to delete the group if he/she is the admin.
- Users will be able to share posts in the group.

- Users will be able to send message in group chat.
- The user will be able to ban users from group if he/she is the admin.
- The user will be able to ban users from group chat if he/she is the admin.
- Users will be able to block other users.
- Users will be able to report other users if they are abusing them in the app.
- Users should be able to add events to favourite events.
- Users should be able to follow other users.
- Users should be able to unfollow users that he/she followed earlier.
- Users should be able to see events that users he/she follows shared.
- Users should be able to see discover nearby the most enrolled events.
- Users should be able to send private messages to other users.
- Users should be able to see description, place on the map and reports of the events.
- Users should be able to send membership request to private groups.
- Users should be able to view other users' profiles.
- Users should be able to search for an event, a group, a user.
- The user should be able to leave a group when he/she wishes to.

3.6.2 Donation System

- Users should be able to donate to specific events with specific products.
- Users should be able to trace the donation, and see that donation reached target.

3.7 Nonfunctional Requirements

3.7.1 Usability

- The application should have simple enough and user-friendly interface that users will be able to use application features easily.
- The application should be free on Google Play market for Android devices and on App Store for Apple devices.

3.7.2 Security

- The system should be secure enough since it is going to hold data about users. Essential data about user will be encrypted and then stored into system.

3.7.3 Reliability

- The application should always work as long as device is connected to internet.

3.7.4 Extensibility

- The system should allow developers to update and add new features.

3.7.5 Transparency

- The system allows to make a donation; therefore, it should be transparent to let users to trace their payment.

3.8 System Models

3.8.1 Scenarios

Scenario 1:

Scenario Name: CreateNewAccount

Participating actor instances: Normal User

Main flow of events:

1. The user downloads the application from Google Play Store and opens it.
2. The user chooses sign up with email account option.
3. The user enters a username.
4. The user enters his/her email address.
5. The user enters a password and confirms the password.
6. The user clicks on “Sign Up” button.
7. The system checks if the email address already exists in database.
8. The system checks if the username is taken by other user.
9. The system confirms that information entered by the user is valid.
10. The system creates a new account.

11. The user is directed to the Home Page.

Scenario 2:

Scenario Name: CreateNewAccountUsingFacebookCredentials

Participating actor instances: Normal User

Main flow of events:

1. The user downloads the application from Google Play Store and opens it.
2. The user chooses sign up with Facebook account option.
3. The system navigates the user to Facebook Login page.
4. The user enters his/her password.
5. The user clicks on “Sign Up Using Facebook” button.
6. The system checks if the email address already exists in the database.
7. The system creates a new account.
8. The user is directed to the Home Page.

Scenario 3:

Scenario Name: CreateNewAccountUsingInstagramCredentials

Participating actor instances: Normal User

Main flow of events:

1. The user downloads the application from Google Play Store and opens it.
2. The user chooses sign up with Instagram account option.
3. The system navigates the user to Instagram Login page.
4. The user enters his/her password.
5. The user clicks on “Sign Up Using Instagram” button.
6. The system checks if the email address already exists in the database.
7. The system creates a new account.
8. The user is directed to the Home Page.

Scenario 4:

Scenario Name: CreateNewAccountUsingGooglelCredentials

Participating actor instances: Normal User

Main flow of events:

1. The user downloads the application from Google Play Store and opens it.
2. The user chooses sign up with Google account option.
3. The system navigates the user to Google Login page.
4. The user enters his/her password.
5. The user clicks on “Sign Up Using Google” button.
6. The system checks if the email address already exists in the database.
7. The system creates a new account.
8. The user is directed to the Home Page.

Scenario 5:

Scenario Name: LoginAtCharin

Participating actor instances: Normal User

Main flow of events:

1. The user opens the application and chooses sign in option.
2. The user enters his/her email address.
3. The user enters his/her password.
4. The user clicks on the Login button.
5. The system checks and confirms that credentials entered by the user are correct.
6. The user is directed to the Home Page.

Scenario 6:

Scenario Name: ChangeSettings

Participating actor instances: Normal User

Main flow of events:

1. After the user logs in to the system, he/she goes to his/her profile.
2. The user clicks on Settings button on his/her profile page.
3. The user makes changes on settings page.
4. The user confirms the changes by clicking on “Ok” button at the top of the page.
5. The system detects the changes and saves them in the database.

Scenario 7:

Scenario Name: CreateAnEvent

Participating actor instances: Normal User

Main flow of events:

1. The user logs in into the system.
2. The user clicks on the Add Button on the screen.
3. The user goes to adding event option on the current screen.
4. The user fills out the necessary information about the event he/she wants to create.
5. The user clicks on Add button.
6. The system checks the information entered by the user and adds the event.

Scenario 8:

Scenario Name: CreateAGroup

Participating actor instances: Normal User

Main flow of events:

7. The user logs in into the system.
8. The user clicks on the Add Button on the screen.
9. The user goes to adding group option on the current screen.
10. The user fills out the necessary information about the group he/she wants to create.
11. The user clicks on Add button.
12. The system checks the information entered by the user and adds the group.
13. The group is created and the user is signed as an admin.

Scenario 9:

Scenario Name: FollowingUsers

Participating actor instances: Normal User

Main flow of events:

1. The user logs in into the system.
2. The user goes to Discover page by clicking on the Discover button at the bottom of the screen.
3. The user types the name of the user on the search field and clicks on Search button.

4. The system checks for the username entered and gives suggestions to the user.
5. The user chooses one of the options and clicks on the user's name.
6. The user is directed to other user's profile.
7. The user clicks on Follow button.
8. The system updates the information accordingly.

Scenario 10:

Scenario Name: SendingPrivateMessage

Participating actor instances: Normal User

Main flow of events:

1. After the user logs in to the system, he/she tap to message button.
2. The user chooses receiver of the message from search.
3. User type in text section.
4. After typing, the user clicks on the Send button.

Scenario 11:

Scenario Name: JoiningToAPublicGroup

Participating actor instances: Normal User

Main flow of events:

1. After the user logs into the system, he/she clicks to the Discover button.
2. User can search group by name.
3. The system checks for the information entered and gives suggestions.
4. The user chooses among suggested groups and clicks on the group name.
5. The system directs the user to the group page.
6. The user clicks on the Join button.
7. The system adds the user as a member of the group and adds the group to the user's profile.

Scenario 12:

Scenario Name: JoiningToAPrivateGroup

Participating actor instances: Normal User

Main flow of events:

1. After the user logs in to the system, he/she clicks to the Discover button.
2. User can search group
3. User enter group page
4. User press send request button

Scenario 13:

Scenario Name: JoiningToGroupChat

Participating actor instances: Normal User

Main flow of events:

1. After the user logs in to the system, he/she tap to + button
2. User can search group
3. User enter group page
4. User press message button
5. User type message on text field
6. User push send button

Scenario 14:

Scenario Name: FollowAnEvent

Participating actor instances: Normal User

Main flow of events:

1. After the user logs in to the system, he/she tap to discovery button
2. User search for Event
3. User enters event page
4. User press favourite button

Scenario 15:

Scenario Name: UnfollowAUser

Participating actor instances: Normal User

Main flow of events:

1. After the user logs in to the system, he/she tap to profiles button or search for person
2. User press following button

3. User enters follower list page
4. User press unfollow

Scenario 16:

Scenario Name: LeaveAGroup

Participating actor instances: Normal User

Main flow of events:

1. After the user logs in to the system, he/she tap to profiles button or search for group
2. User enters group page
3. User press leave button
4. User see system message

Scenario 17:

Scenario Name: DisplayMap

Participating actor instances: Normal User

Main flow of events:

1. User opens up the application.
2. And press discover button which pops up the map and nearby events.

Scenario 18:

Scenario Name: Donate

Participating actor instances: Normal User

Main flow of events:

1. User opens up the application
2. Selects the event that he/she wants to donate from the main page or in the discover page (some charin events)
3. After opening event page, user will be able to donate to that event by pressing donate button.
4. After pressing donate button, user should specify needs that she/he is going to send, and how much he/she is going to send.

5. After specifying type of needs and amount of needs, user will have two choices which are delivering himself, or delivering with courier with some extra cost. User will choose one of them.

Scenario 19:

Scenario Name: BanAUserFromTheGroup

Participating actor instances: Admin

Main flow of events:

1. User opens up the application.
2. Navigates to the groups page from searching its name in discover page or selecting group from his/her own profile.
3. Opens up group settings from the button in the group page.
4. Admins will be able to ban a user from group from the button that is located in group settings.
5. Banned users' posts will not be shared in the group page.

Scenario 20:

Scenario Name: DeleteTheGroup

Participating actor instances: Admin

Main flow of events:

1. User opens up the application.
2. Navigates to the groups page from searching its name in discover page or selecting group from his/her own profile.
3. Opens up group settings from the button in the group page.
4. If the user is admin to the group, user will be able to delete the group
5. That group will be deleted from the system, and will not show up in search.

Scenario 21:

Scenario Name: AssignAUserAsAdmin

Participating actor instances: Admin

Main flow of events:

1. User opens up the application.
2. Navigates to the groups page from searching its name in discover page or selecting group from his/her own profile.
3. Opens up group settings from the button in the group page.
4. If the user is admin to the group, user will be able to assign new admin to the from group settings page.
5. That post will be deleted from the group page and will not be seen furthermore.

Scenario 22:

Scenario Name: ShareAPostOnAGroup

Participating actor instances: Normal User

Main flow of events:

1. User opens up the application.
2. Navigates to the groups page from searching its name in discover page or selecting group from his/her own profile.
3. Wser will be able to share a post in the group chat from the add button the group page.
4. That post will be added to the group page and other members of the group will be able to examine it.

Scenario 23:

Scenario Name: DeleteAPostOnTheGroup

Participating actor instances: Admin

Main flow of events:

1. User opens up the application.
2. Navigates to the groups page from searching its name in discover page or selecting group from his/her own profile.
3. If the user is admin to the group, user will be able to delete a post from the group chat from the group chat settings.
4. That post will be deleted from the group page and will not be seen furthermore.

Scenario 24:

Scenario Name: BanAMemberFromGroupChat

Participating actor instances: Admin

Main flow of events:

5. User opens up the application.
6. Navigates to the groups page from searching its name in discover page or selecting group from his/her own profile.
7. If the user is admin to the group, user will be able to ban a user from the group chat from the group chat settings.
8. That user will not be able to message in the chat for the specific period.

Scenario :25

Scenario Name: Block User

Participating actor instances: Normal User

Main flow of events:

1. User opens up the app.
2. Navigates to the user's page that he/she wants to block.
3. User can block any other user that can message him/her by the button in the user profile.
4. After blocking a user, that user's messages will not delivered to the user. And post will not be available to that user to see.

3.8.2 Use Case Model

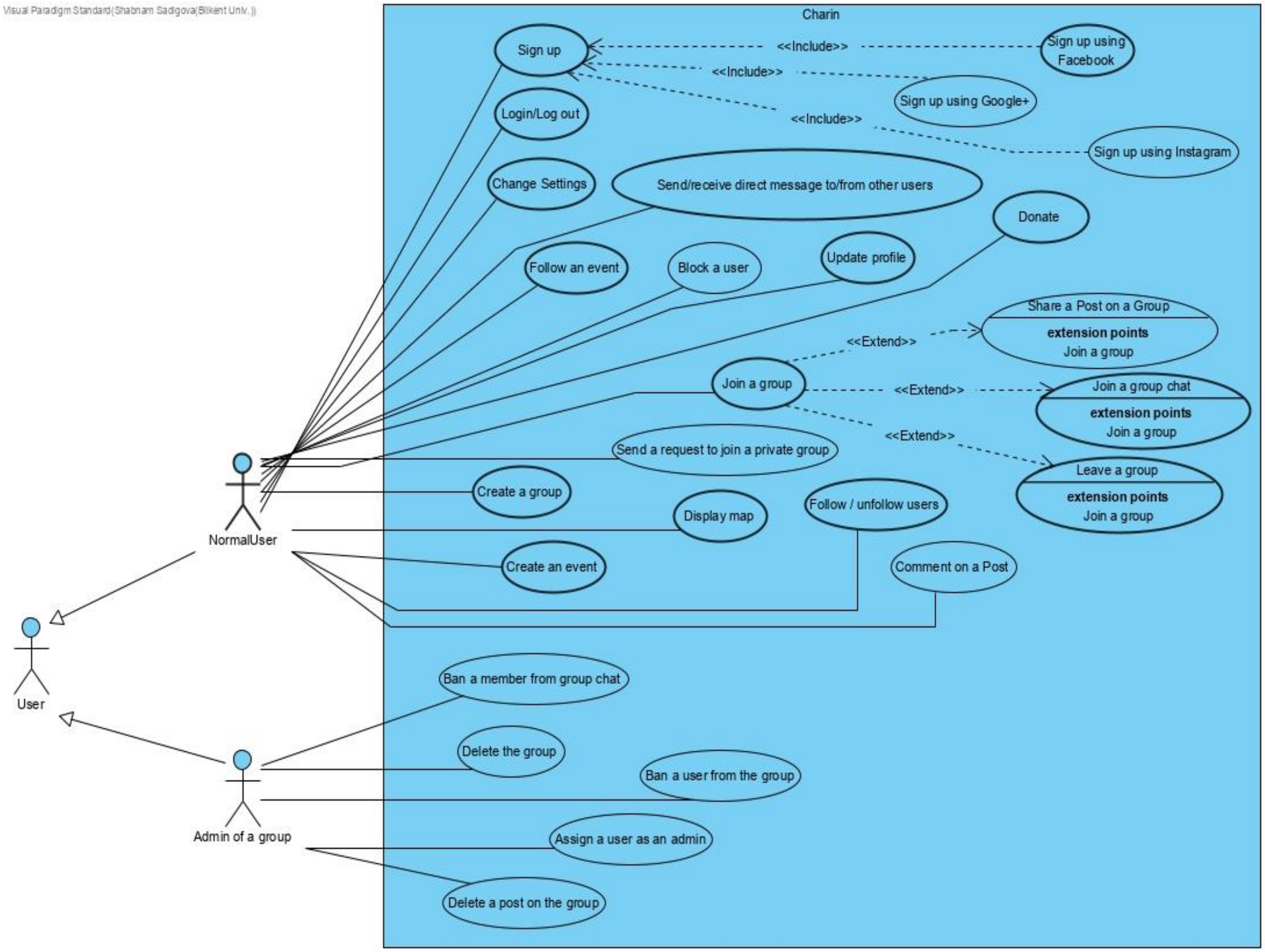


Figure 1. Use Case Diagram.

3.8.2.1 Login Use Case

Use Case Name: Login

Actor: Mezi

Entry Condition:

- User is on Login Screen.

Exit Condition:

- User is on HomePage screen.
- Authentication Failed and user is on Login screen.

● User change to SignUp screen.

Main flow of events:

1. User choose sign in method
2. App checks if the account exists.
3. App redirect user to HomePage.

Alternative flow of events:

1. Such account does not exist.
    a. App detects that account does not exist and inform user
    b. App redirect user to Login page
    c. App notifies user about the wrong entries.
2. User taps “Sign In With Facebook” button
    a. App checks for the user’s Facebook credentials.
    b. App redirects user to HomePage
3. User taps “Sign In With Google” button
    a. App checks for the user’s Google credentials.
    b. App redirects user to HomePage.

3.8.2.2 Create Account Use Case

Use Case Name: Create Account

Actor: Mezi

Entry Condition:

● User is on Login Screen.

Exit Condition:

● User is on HomePage screen.

● User change to Login screen

Main flow of events:

1. User clicks “Sign Up” button
2. App redirects user to Sign Up page.
3. User enters preferred username
4. User enters email address.

5. User enters password twice.

6. User clicks “Register” button.

7. App checks if the username is available.

8. App checks if there is a user with the given email address.

9. App creates a new account for the user.

10. App redirects user to HomePage.

Alternative flow of events:

1. User inputs are inaccurate or missing.

   a. App empties the inaccurate forms and redirect the SignUp page.

   b. App notifies the user about the inaccurate forms properly.

   c. User enters the inaccurate information again and step 8-10 are repeated

3.8.2.3 Change Settings Use Case

Use Case Name: Change Settings

Actor: Mezi

Entry Condition:

- User is logged in system and on profile screen.

Exit Condition:

- User changes to the home screen.
- User changes to the discover screen.
- User changes to the adding event/group screen.
- User returns back to profile screen.

Main flow of events:

1. User opens up settings screen from profile screen from the button.
2. Activates and deactivates options that he wants.
3. In the moment of changing settings, those changes apply to app.

3.8.2.4 Block user Use Case

Use Case Name: Block user

Actor: Mezi

Entry Condition:

- User opened app, and logged in the system.

Exit Condition:

- User changes to the home screen.
- User changes to the discover screen.
- User changes to the adding event/group screen.
- User changes to his own profile screen.

Main flow of events:

1. User searches name of the user that he wants to block.
2. Presses the button on the top right corner which shows up options about users.
3. He chooses block user option and blocks user which he prevents this user from messaging him and observing his posts.

Alternative flow of events:

1. User is already blocked
    a. System pops up message that user is already blocked
2. No such user to block
    a. System does not find that user with the message that no such user exists.

3.8.2.5 Update Profile Use Case

Use Case Name: Update Profile

Actor: Mezi

Entry Condition:

- User opened app, and logged in the system.

Exit Condition:

- User changes to the home screen.
- User changes to the discover screen.
- User changes to the adding event/group screen.

Main flow of events:

4. User presses the edit profile button to change information indicated in the User profile screen.

5. Selects the slot that he wants to change and fills the new information.
6. Presses the confirm button to save changes.

Alternative flow of events:

3. User is not connected to the internet.
    a. Should try it again when he is connected to the internet.

3.8.3 Object and Class Model

Class Diagram:

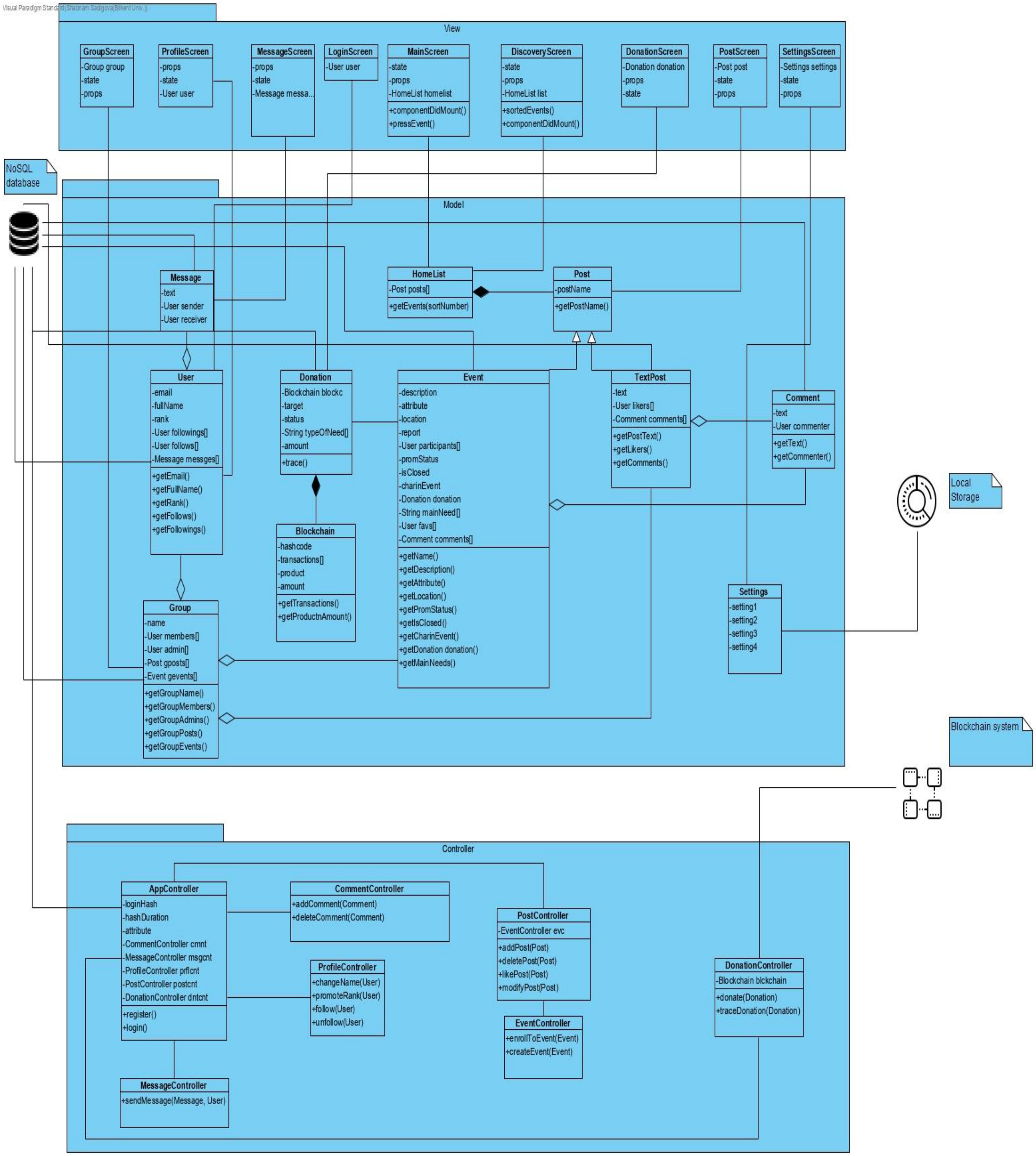


Figure 2. Class Diagram.

Object Diagram:

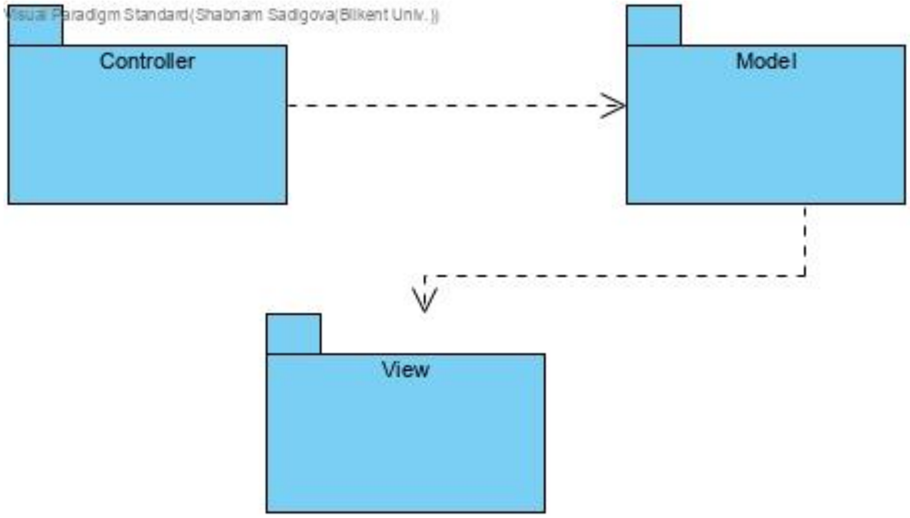


Figure 3. Object Diagram.

3.8.4 Dynamic Model

Activity Diagram:

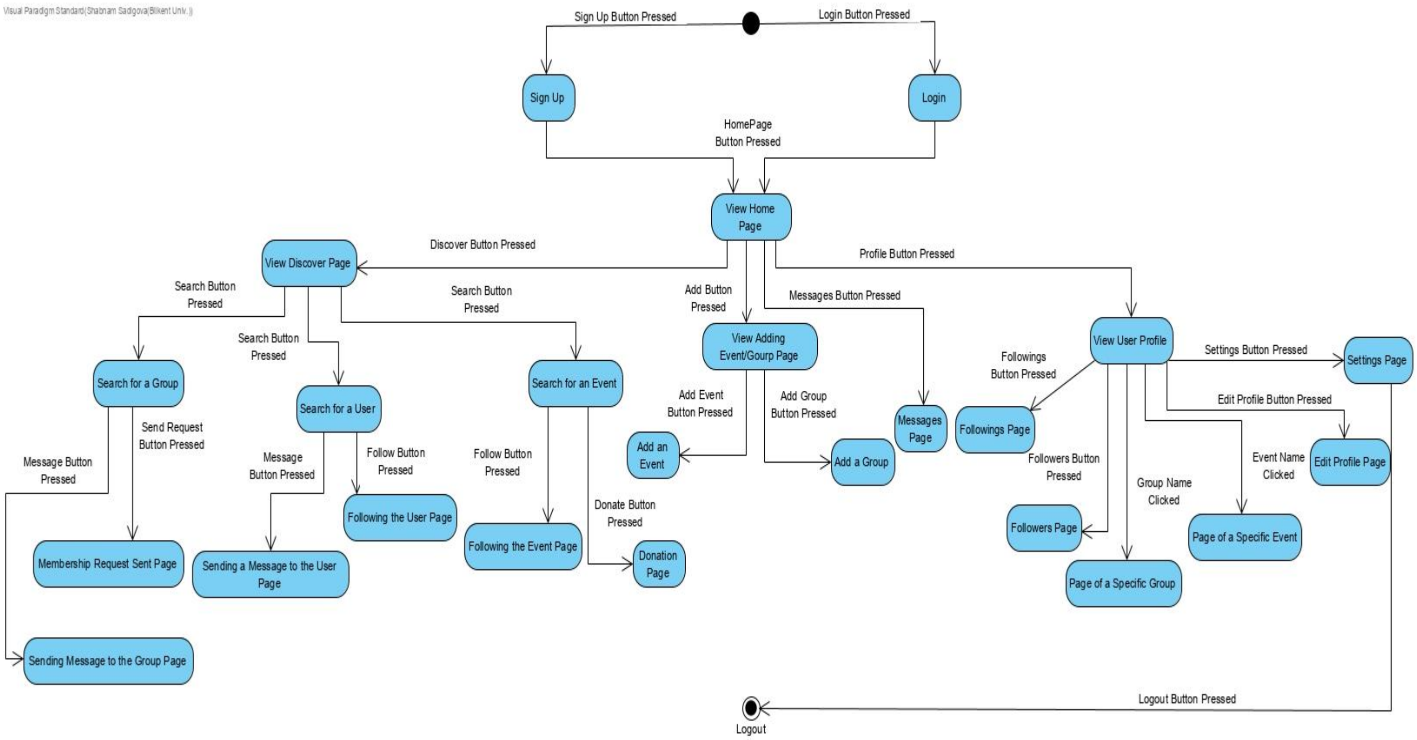


Figure 4. Activity Diagram.

Login Sequence Diagram:

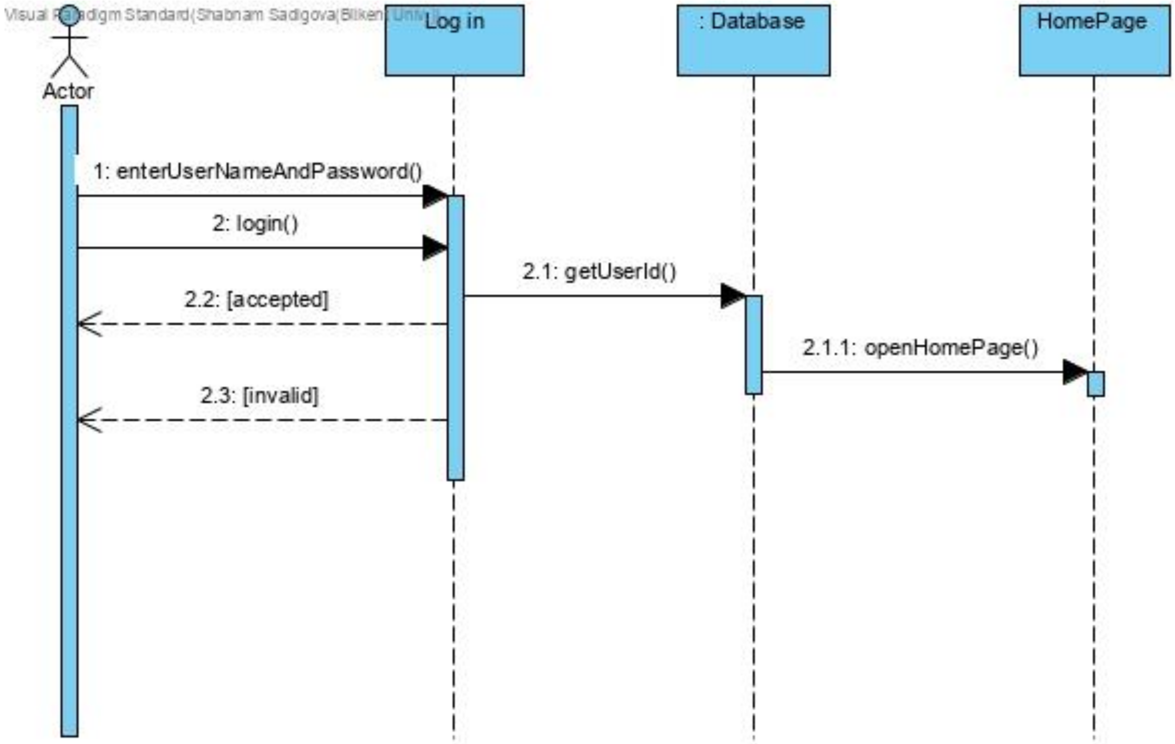


Figure 5. Login Sequence Diagram.

Sign-up Sequence Diagram:

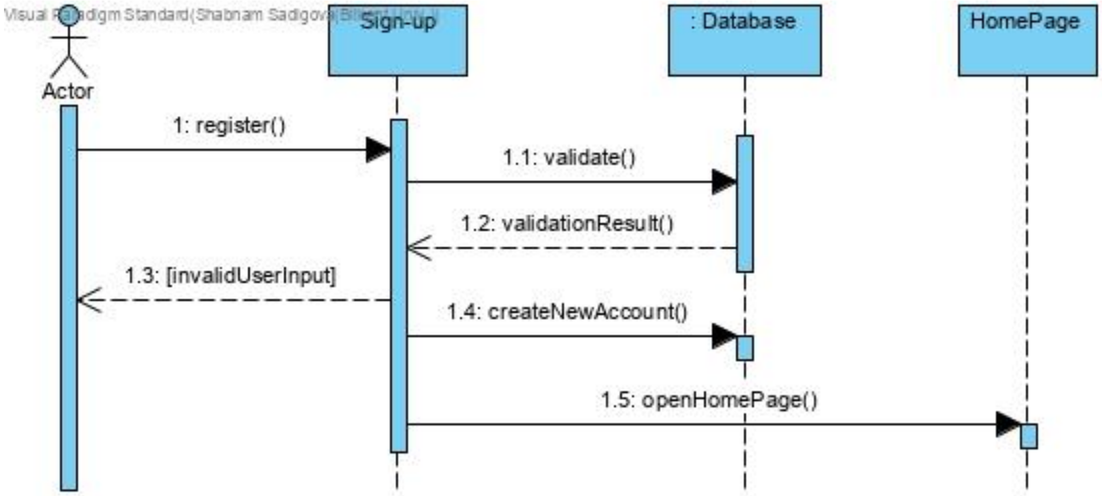


Figure 6. Sign-up Sequence Diagram.

Search Events Sequence Diagram:

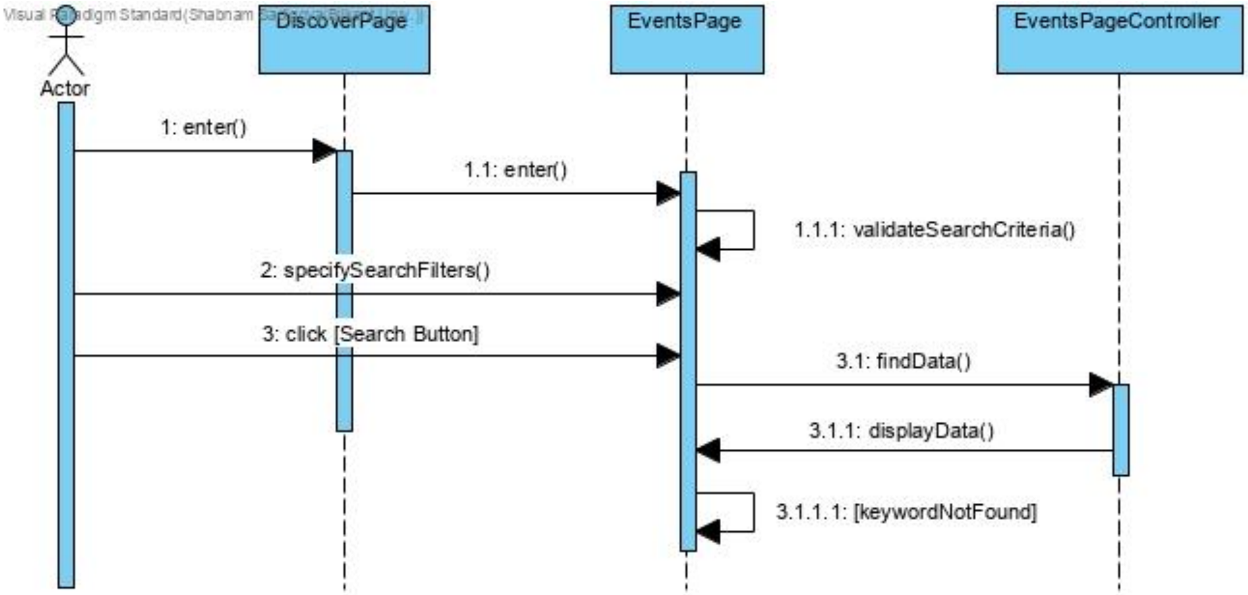

Visual Paradigm Standard(Shabnam Sadigova(Bilkent Univ.))
DiscoverPage
EventsPage
EventsPageController
Actor
1: enter()
1.1: enter()
1.1.1: validateSearchCriteria()
2: specifySearchFilters()
3: click [Search Button]
3.1: findData()
3.1.1: displayData()
3.1.1.1: [keywordNotFound]

Figure 7. Search Event Sequence Diagram.

Search Groups Sequence Diagram:

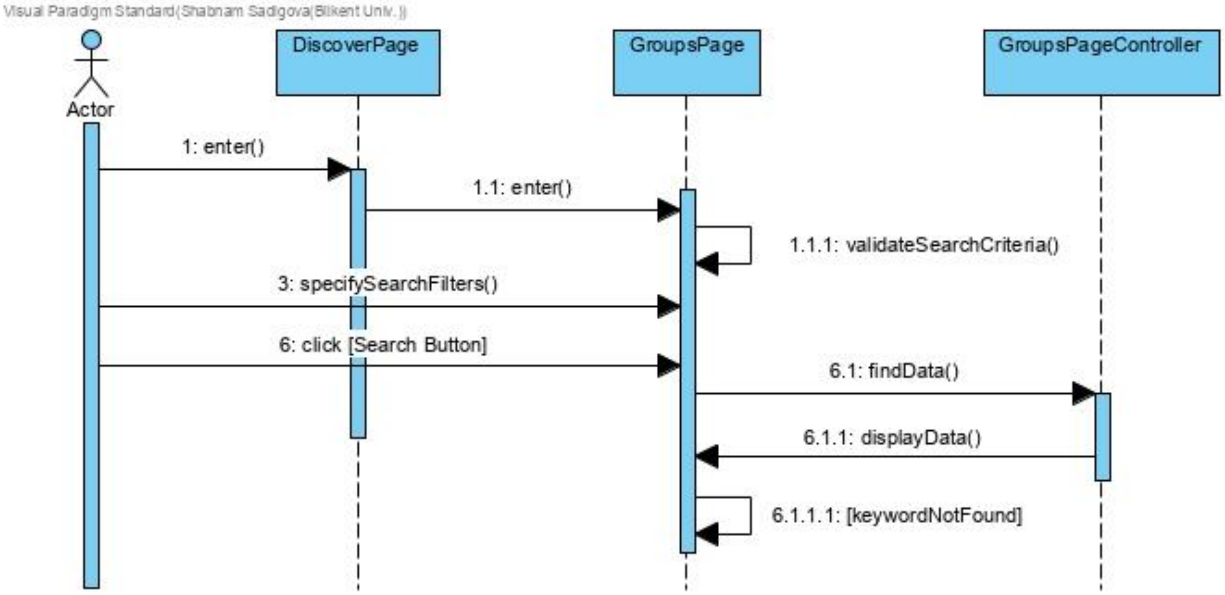

Visual Paradigm Standard(Shabnam Sadigova(Bilkent Univ.))
DiscoverPage
GroupsPage
GroupsPageController
Actor
1: enter()
1.1: enter()
1.1.1: validateSearchCriteria()
3: specifySearchFilters()
6: click [Search Button]
6.1: findData()
6.1.1: displayData()
6.1.1.1: [keywordNotFound]

Figure 8. Search Group Sequence Diagram.

Search User Sequence Diagram:

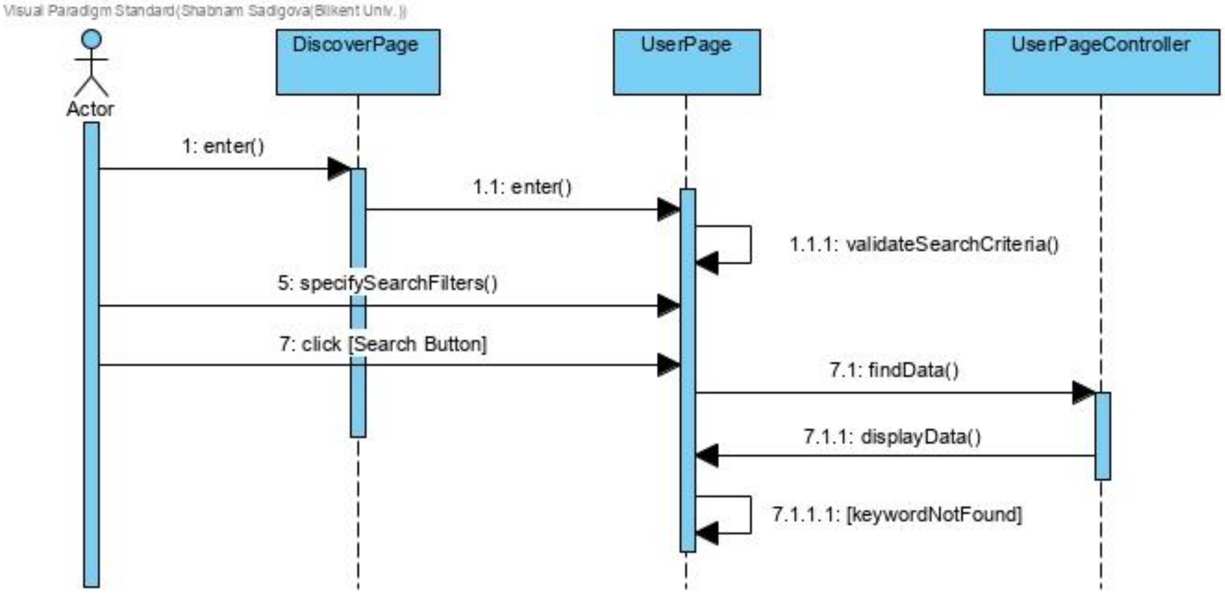

Visual Paradigm Standard(Shabnam Sadigova(Bilkent Univ.))
Actor
DiscoverPage
UserPage
UserPageController
1: enter()
1.1: enter()
1.1.1: validateSearchCriteria()
5: specifySearchFilters()
7: click [Search Button]
7.1: findData()
7.1.1: displayData()
7.1.1.1: [keywordNotFound]

Figure 9. Search User Sequence Diagram.

3.8.5 User Interface - Navigational Paths and Screen Mock-ups

3.8.5.1 Login Screen

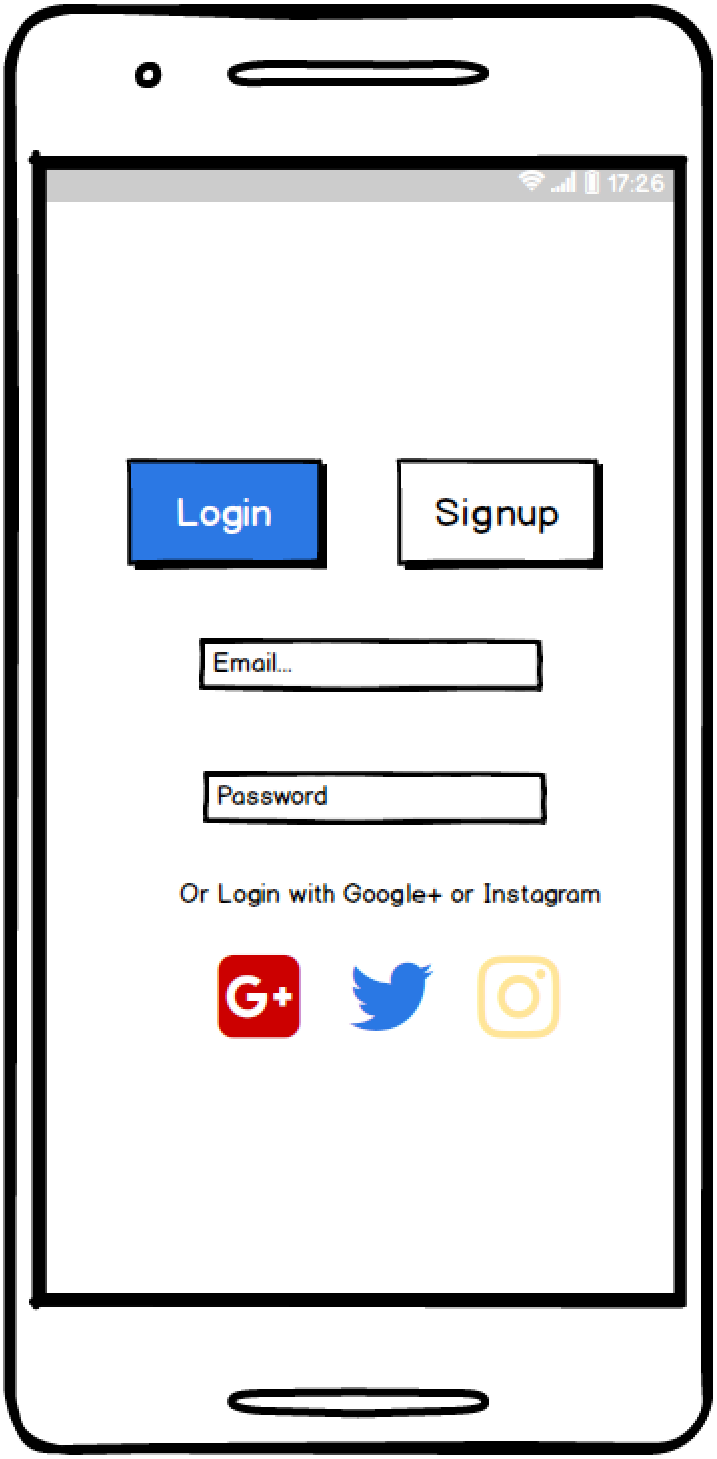


Figure 10. Login Screen.

Users can login to the app with their registered emails or their G+ or social accounts.

3.8.5.2 Register Screen

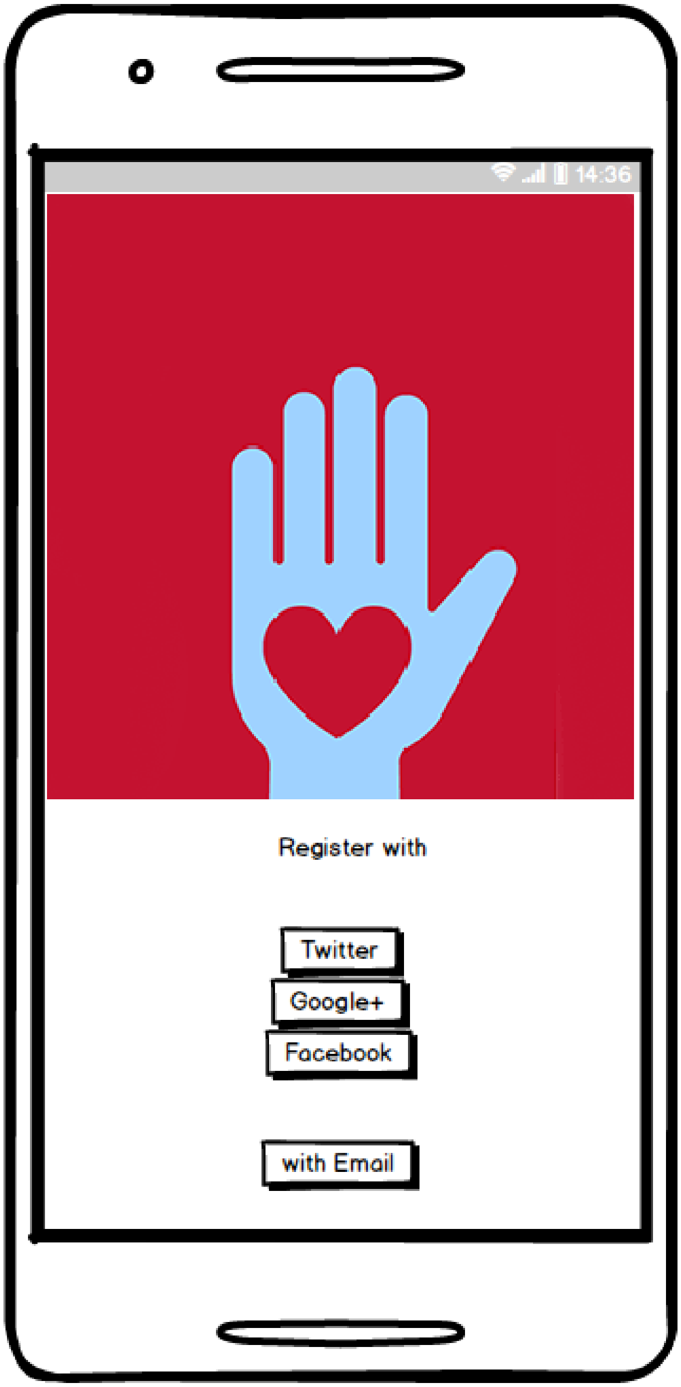


Figure 11. Sign-up Screen.

This is register page for users that uses our app for the first time. It offers options like Twitter, Facebook, Google+ which makes registration process much easier. Besides, app also offers registration with ordinary email.

3.8.5.3 Sign with mail

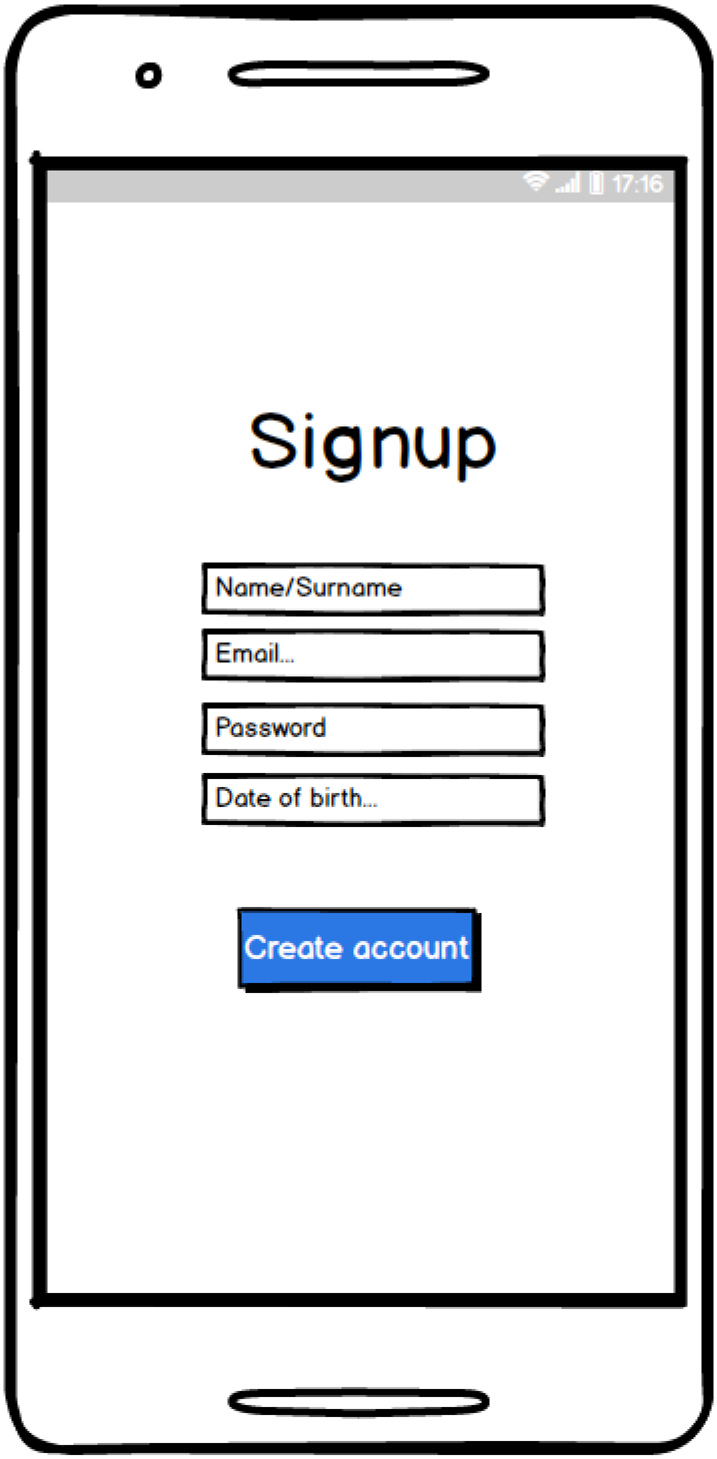
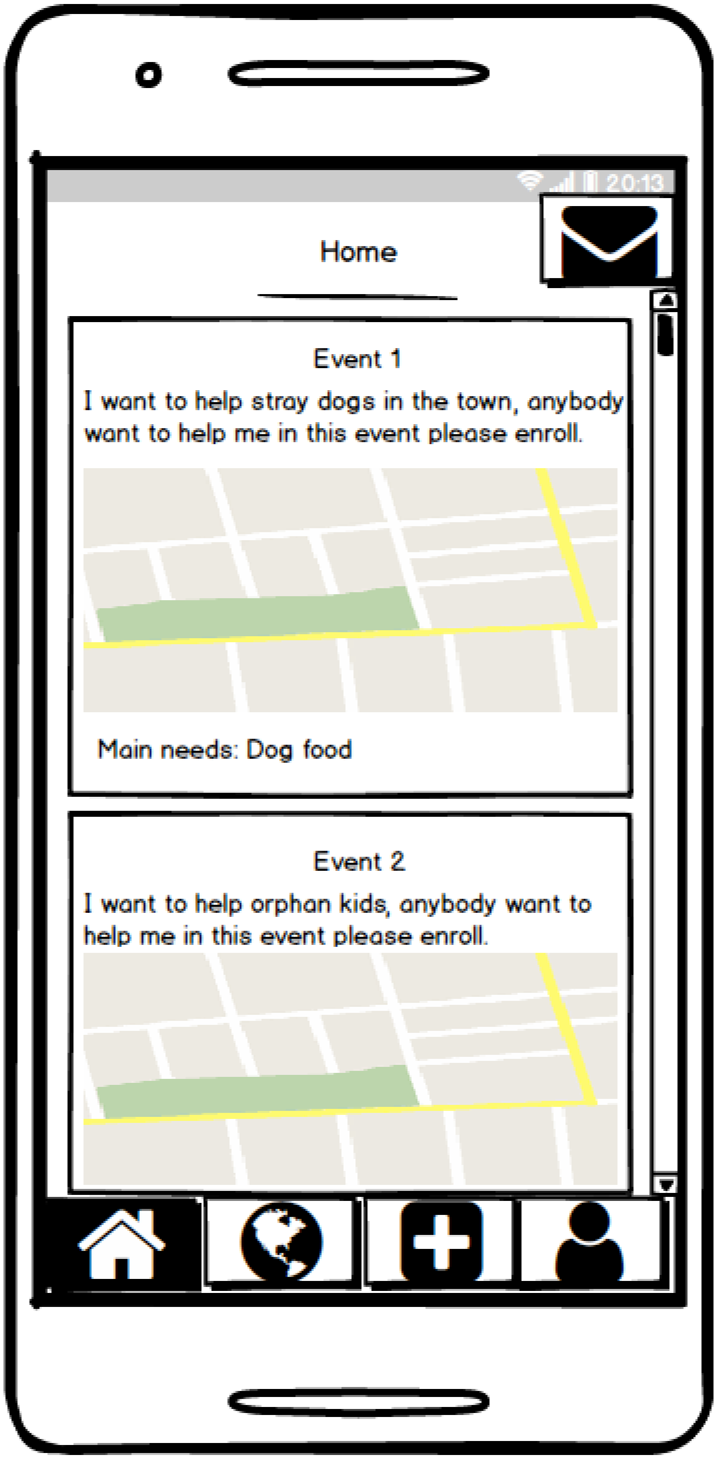


Figure 12. Sign-up with email Screen

In register screen, if users do not have social accounts or G+, they can register with their emails.

3.8.5.4 Main Page



Event 1

I want to help stray dogs in the town, anybody want to help me in this event please enroll.

Main needs: Dog food

Event 2

I want to help orphan kids, anybody want to help me in this event please enroll.

Figure 13. Main Page Screen

The following image is the main screen in our app. It displays events those are shared by followed users. Each event is briefly stated in rectangles, it informs user with crucial info such as name, description, location, and main need. User can choose in which he/she wants to participate in. Users will be able to see the messages that he/she received from top-right corner.

3.8.5.5 User Profile

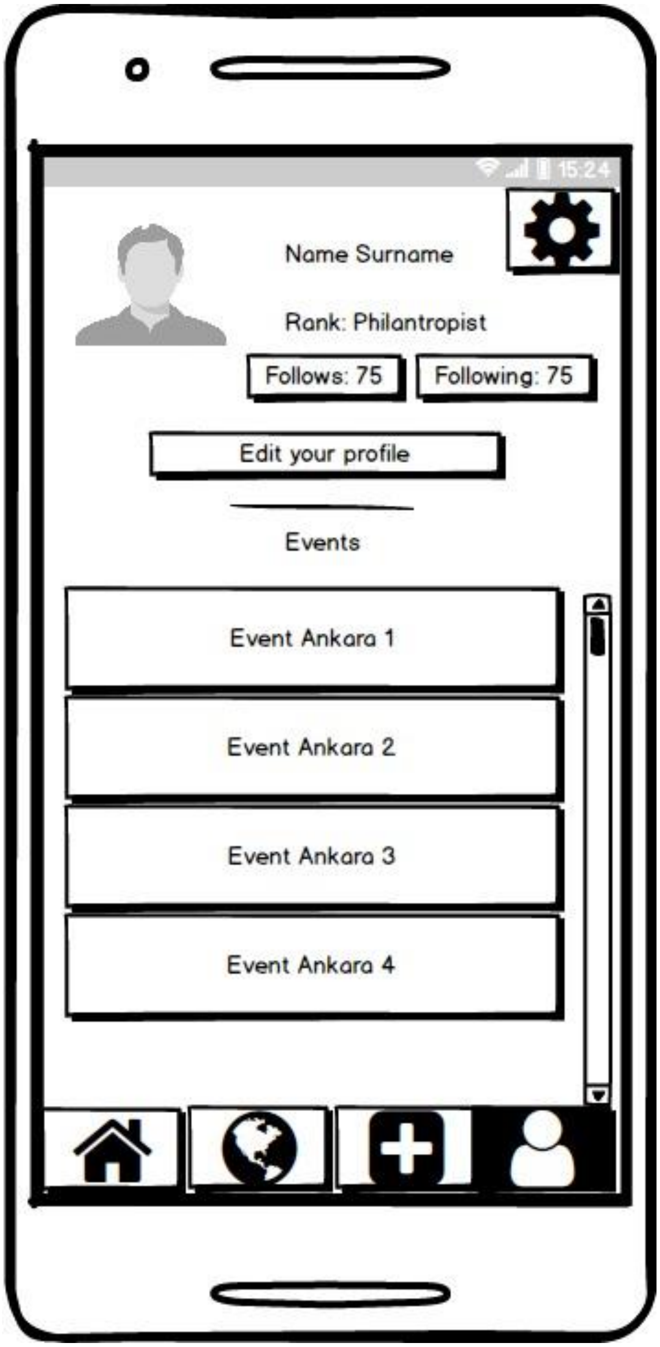
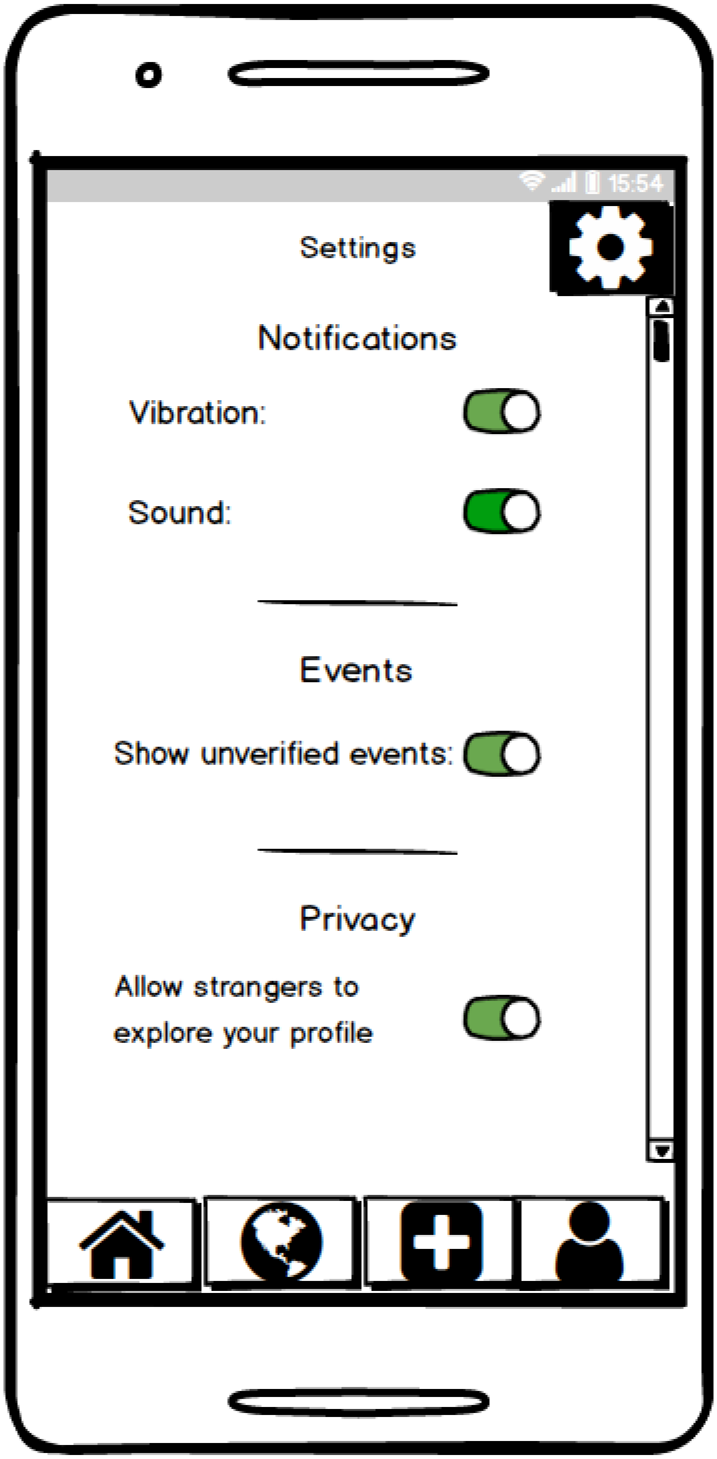


Figure 14. Profile Screen

Users can examine their profile and edit their information in the app. User profile page will let user to learn count of his/her followings, followers and events that he participated in. Since we have rank system in app, user will also be able to display his rank.

3.8.5.6 Settings page



Settings

Notifications

Vibration:

Sound:

Events

Show unverified events:

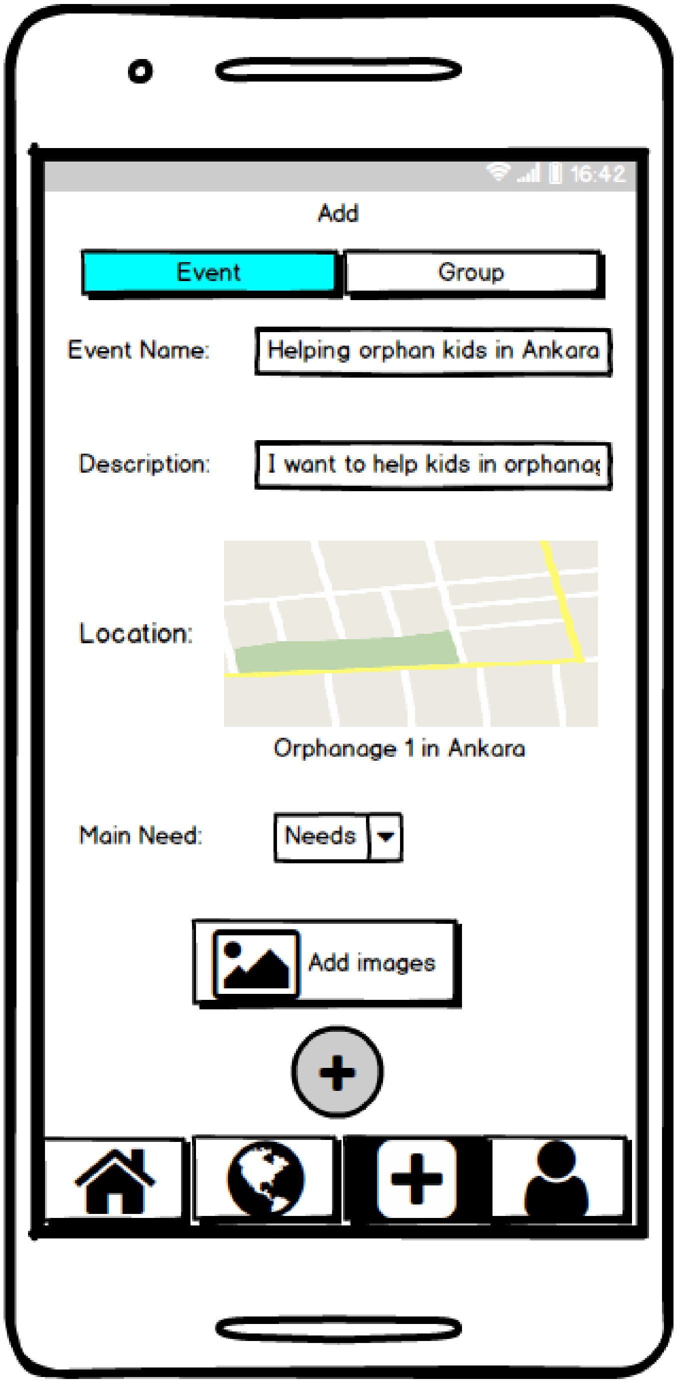
Privacy

Allow strangers to explore your profile

Figure 15. Settings Screen

Following page is settings page which let users to make some changes from the default settings.

3.8.5.7 Adding Event



Event Name: Helping orphan kids in Ankara

Description: I want to help kids in orphanag

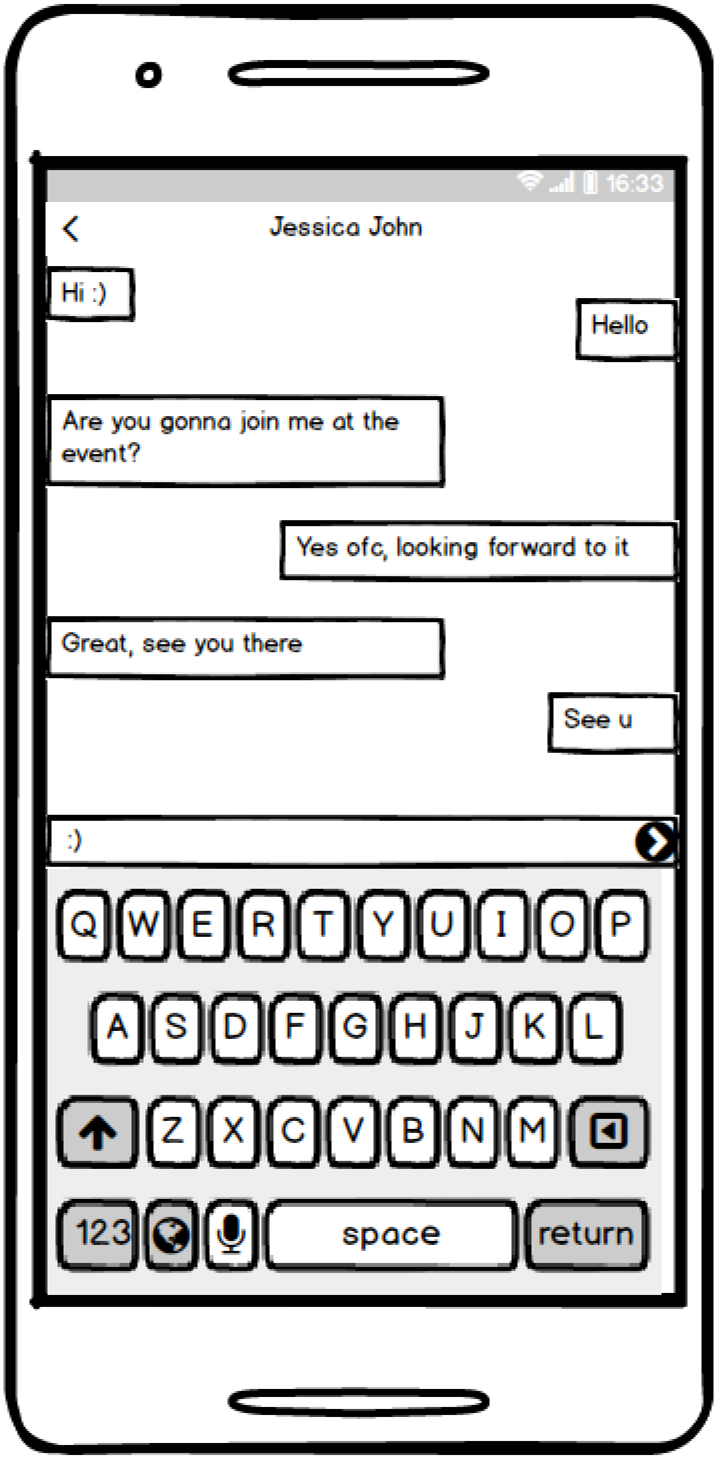
Location: Orphanage 1 in Ankara

Main Need: Needs

Figure 16. Adding Event Screen

The following page demonstrates adding event page. By this page users will be able to add event by filling specific slots. User needs to specify events name, description, location, and the main need. User also will be able to add photos of the event to visualize status.

3.8.5.8 Private Messaging



Hi :)

Hello

Are you gonna join me at the event?

Yes ofc, looking forward to it

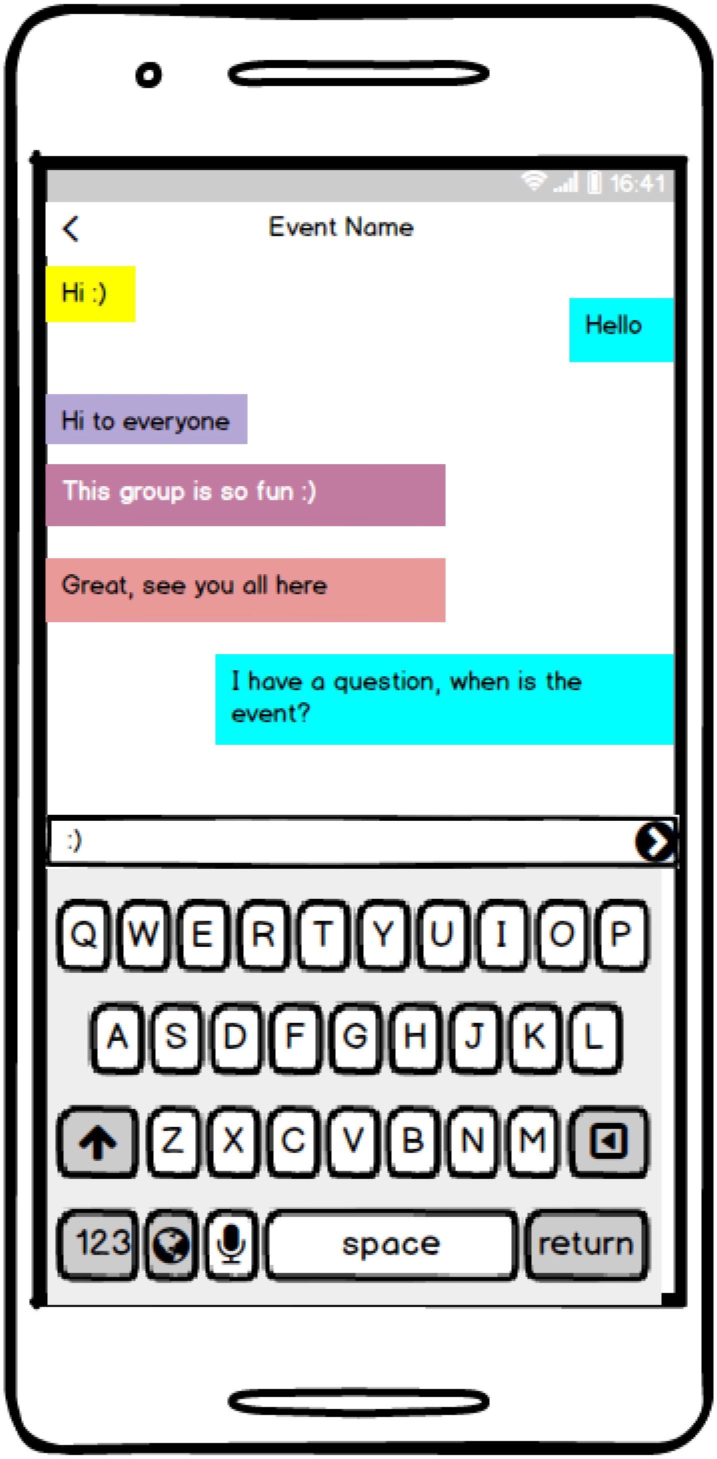
Great, see you there

See u

:)

Figure 17. Private Messaging Screen

3.8.5.9 Group Messaging



16:41

Event Name

Hi :)

Hello

Hi to everyone

This group is so fun :)

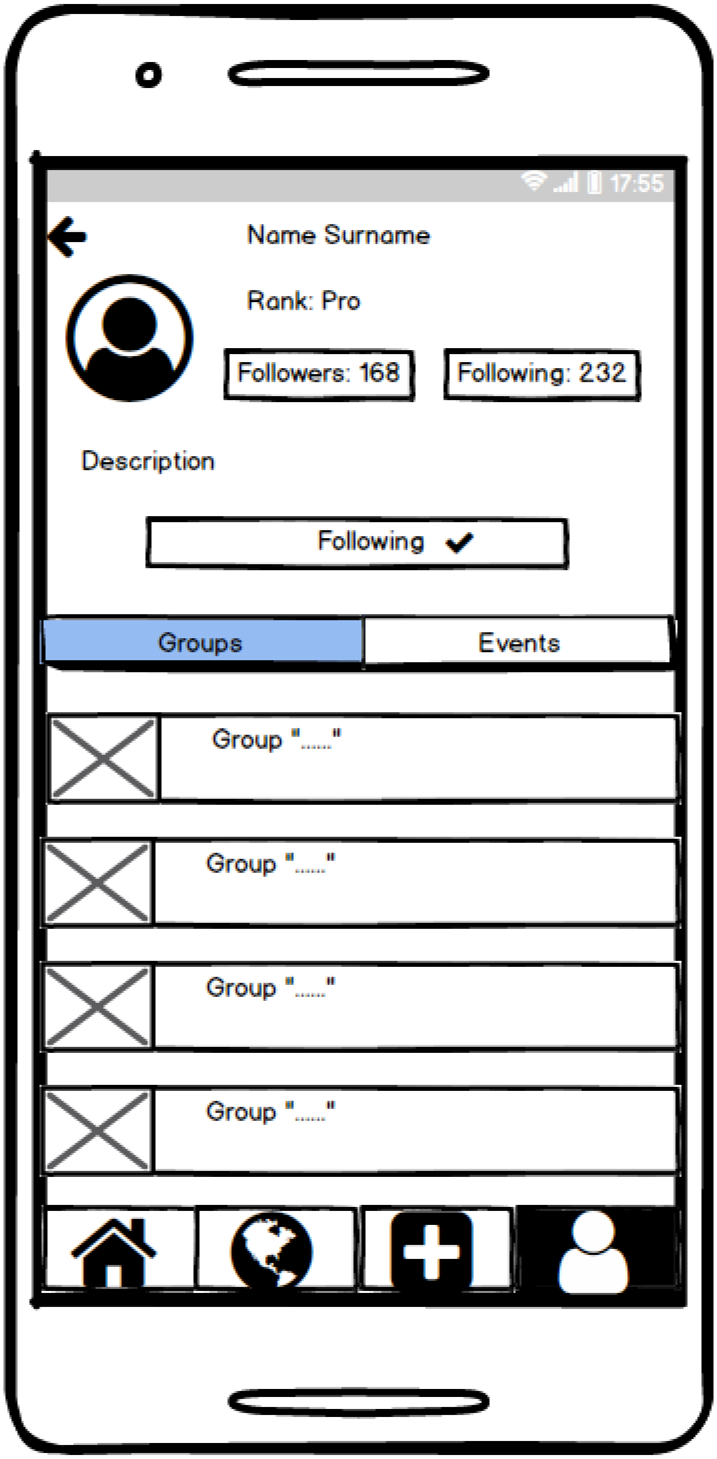
Great, see you all here

I have a question, when is the event?

:)

Figure 18. Group Messaging Screen

3.8.5.10 View Followed Account



17:55

Name Surname

Rank: Pro

Followers: 168

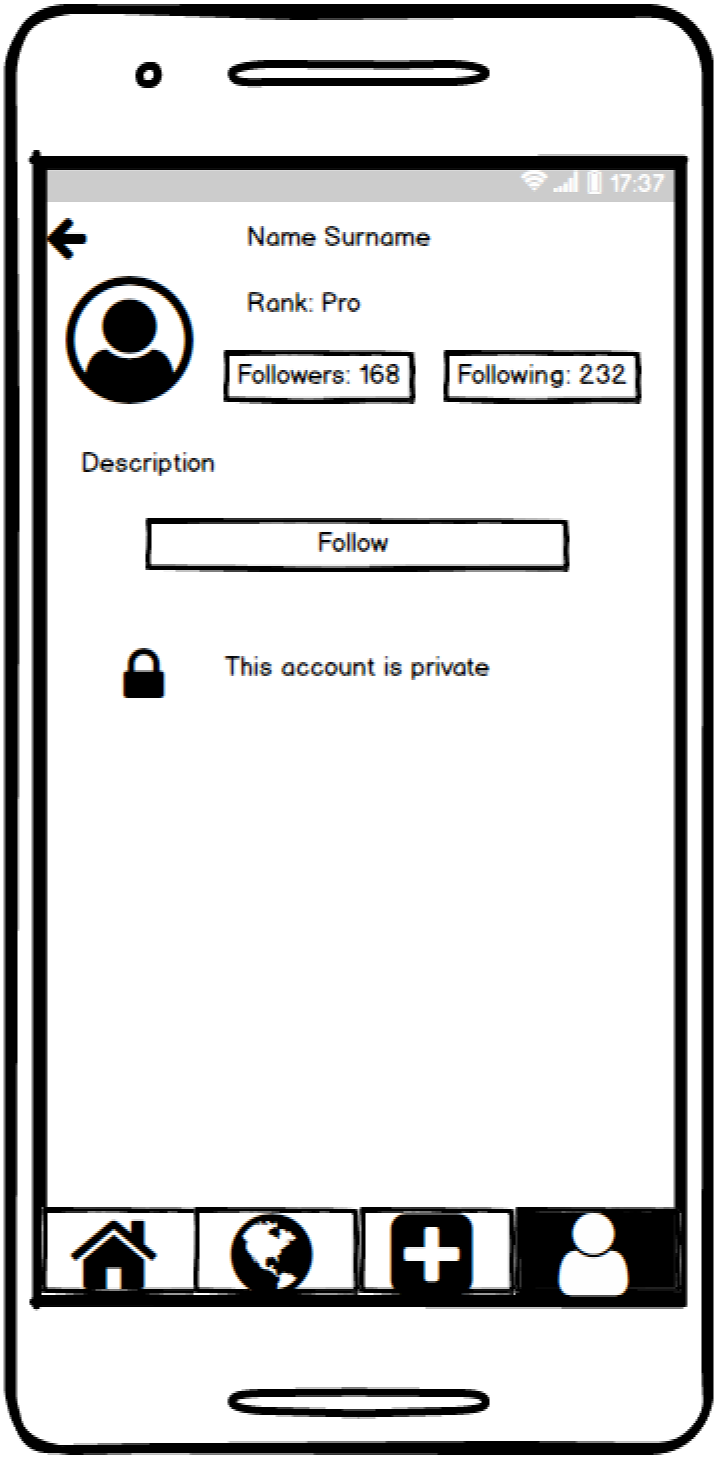
Following: 232

Description

Following ✓

Figure 19. Viewing Followed Account Screen

3.8.5.11 View Private Account



Name Surname

Rank: Pro

Followers: 168

Following: 232

Description

This account is private

Figure 20. Viewing Private Account Screen

3.8.5.12 View Public Account

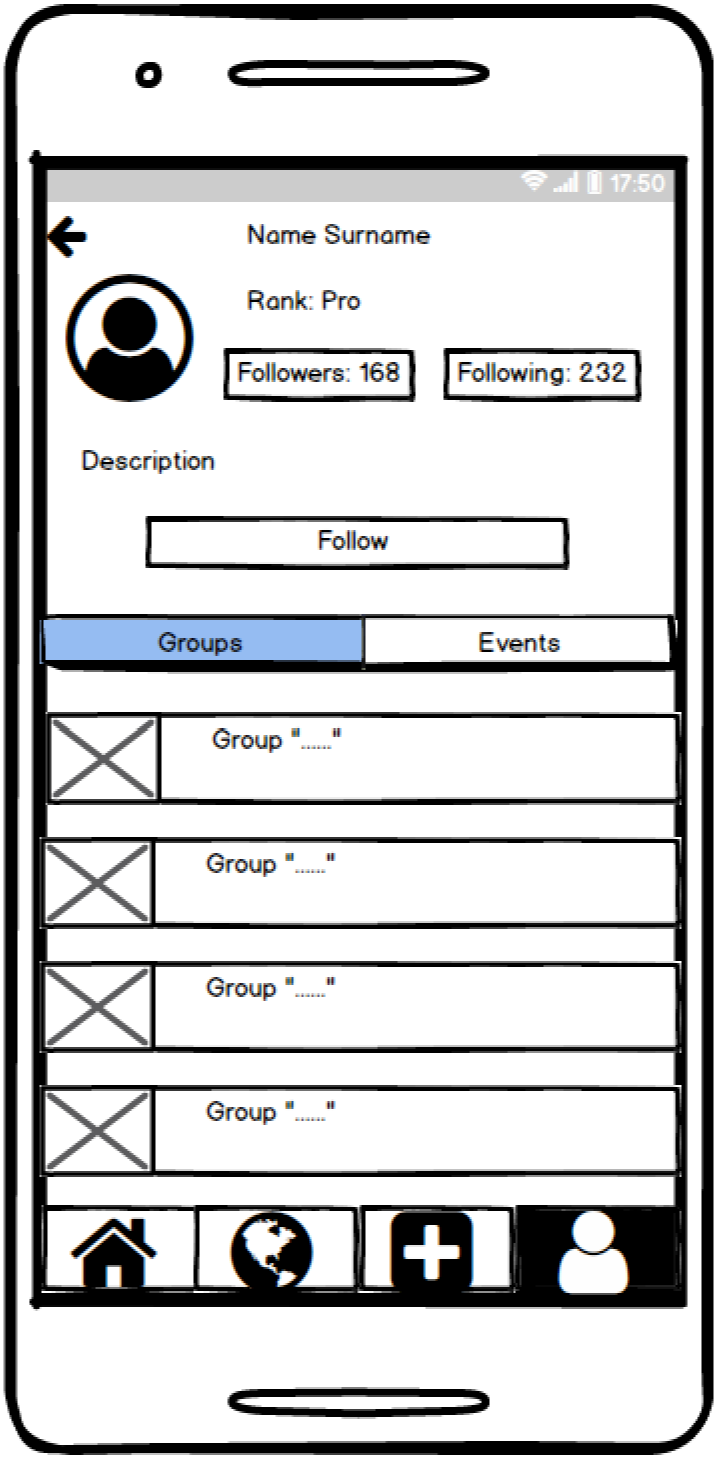


Figure 21. Viewing Public Account Screen

4. Other Analysis Elements

4.1 Consideration of Various Factors

4.2 Risks and Alternatives

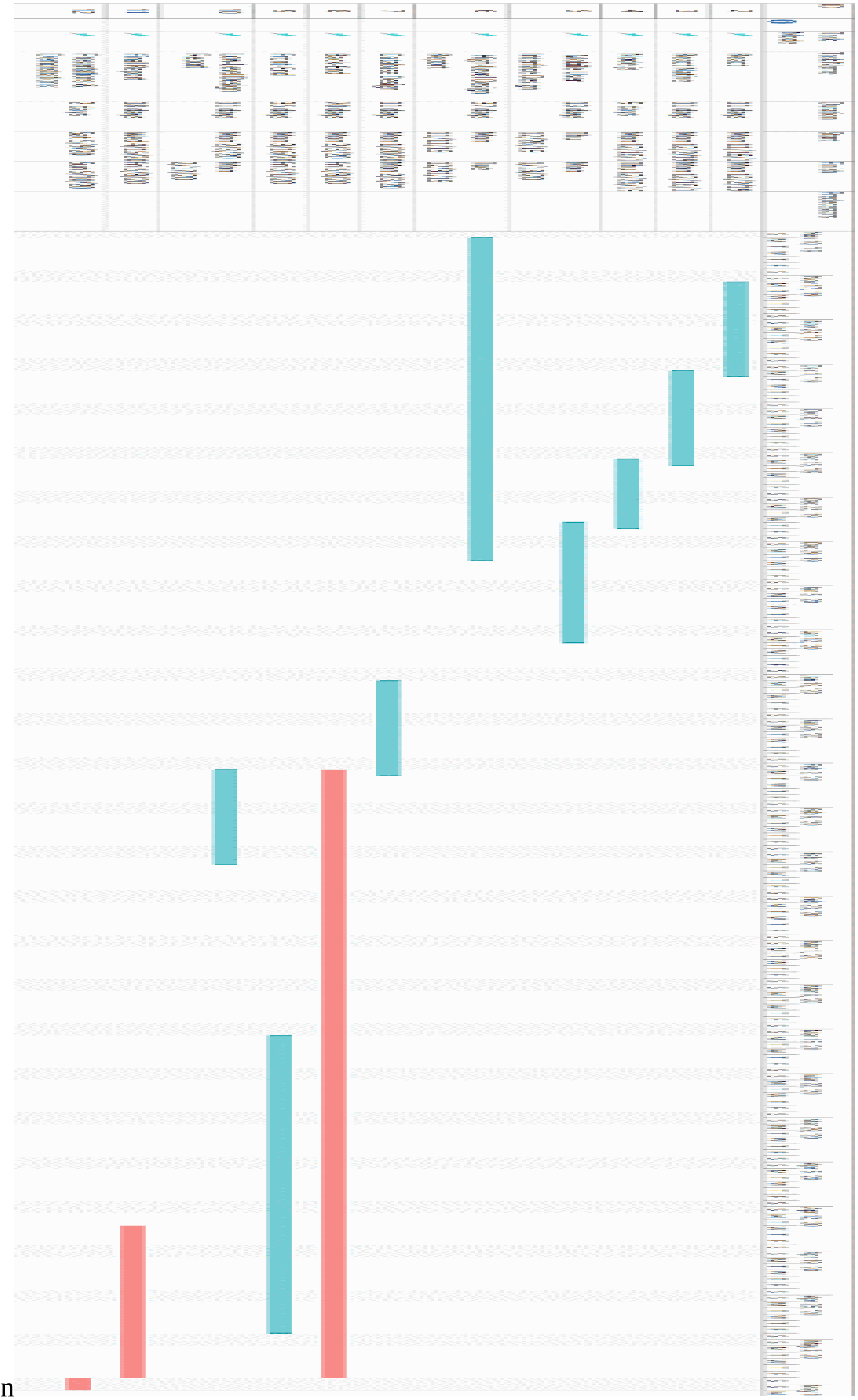
Performance risk, the risk that the project will fail to produce results consistent with project specifications.

To eliminate performance risk our team members will work hard. The weak points of our team is blockchain. Because it is new technology and in university there is no course for that. But there are a lot of sources on the internet using these sources we will improve our knowledge about blockchain technology our team members are enthusiastic about it.

Schedule risk, the risk that activities will take longer than expected. Slippages in schedule typically increase costs and delay the receipt of project benefits, with a possible loss of competitive advantage.

For eliminating this risk, we use Gantt chart. We will obey this chart. In the semester break our team members will stay in Ankara and we will arrange daily meeting and try to develop our app.

Project Plan



4.3 Ensuring Proper Team-work

We will work together and try to improve our skills in every aspect. But of course, some of team members are good at some parts. Shabnam and Ismayil are good at front-end, Mastan and Fuad are good in back - end. Fuad also try to improve himself in blockchain technology.On next phase we will design Team work using Microsoft project.

4.4 Ethics and Professional Responsibilities

Our team try to be cross functional. Every team member are good at in one field and when we unite we fill our gap. We keep ourselves up to date and constantly research for improving our app and using new programming language. Our team leader change time to time. It depends on fields which we are good or pro. Our main goal is to produce good app which serves people who need help and unite them. Honesty is one of the most fundamental ethical values. We try to be honest in workload and our source code belongs to us.

4.5 New Knowledge and Learning Strategies

In our summer internship every team member learned some new technology. SQL database, Node.js , React and React Native are examples. Blockchain is the new technology and we are noob at it. We will use Retrieval Practice and Elaboration strategies for learning. In retrieval practice method when we will learn new things we try to practice it for clear understanding. Using Elaboration technique we will explain and describe ideas with many details. No Sql database are new technology for us. “EDx” and “Udemy” has many courses for learning No sql and Blockchain technology. “EDx” is free all of the team members can use with their account but Udemy courses are not free. We use one account for learning. We schedule time and try to improve our skills. In addition, our instructor also will help us. We arrange daily meeting with our professor and he helps us to make our app good.

5. 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]

[2] Tdp.bilkent.edu.tr. (2019). [online] Available at: https://www.tdp.bilkent.edu.tr/ [Accessed 25 Oct. 2019].

[3] Observation of Elshad Mahmudov(Member of K1L0 project of TDP) [Observed. 25 March 2018].

[4] https://www.globalgiving.org/ (2019). [online] Available at: https://www.globalgiving.org/ [Accessed 26 Oct. 2019].

[5] https://www.alice.si/ (2019). [online] Available at: https://www.alice.si/ [Accessed 26 Oct. 2019].