



Bilkent University

Department of Computer Engineering

Senior Design Project

Project short-name: Bloodhub.

Project Specifications Report

Mustafa Culban, İzel Gürbüz, Erdem Karaosmanoğlu, Mehmet Orçun Yalçın

Supervisor: Uğur Güdükbay

Jury Members: İbrahim Körpeoğlu, Özgür Ulusoy

Progress Report
Oct 9, 2016

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

1	Introduction	3
1.2	Constraints	4
1.1.1	Sustainability	4
1.1.2	Economic Constraints	4
1.1.3	Security Constraints	5
1.1.4	Implementation Constraints	5
1.3	Professional and Ethical Issues	5
2	Requirements	6
2.1	Functional Requirements	6
2.2	Non-Functional Requirements	7
3	Works Cited	9

1 Introduction

Humankind has become the leader race of earth and got top of food chain with indisputable helps of their intelligence and technology related to it. Technology is a controversial issue because while it gives numerous blessings to humanity, it also took a lot from them. Violence, battles, fights, diseases have always been in history of humankind but with developing technology their harms have significantly increased.

Traffic accidents, surgeries, gun battles cause thousands of wounding every day. Naturally, a considerable amount of people need blood donation. Ironically, these problems are solved with technology again. Technology can take an important place in solutions to blood donating problems. With social networking voluntary blood donors and people who need blood can meet under a social media platform.

With invention of Facebook, Internet started to use its potential. People started to interact others while connected to Internet. After Facebook, “social network” has become a phenomenon. Different websites and applications followed this trend. All of these social platforms created a new power which is called as “social media”. Today, even governments forced into consider social media. Social platforms on Internet are used by millions of people, so people use it for their urgent needs too. During emergency situations users announce need for blood for their relatives or friends. With this way, announcement is heard by many people and it makes easy to find blood. Making an application just for this job can save lives.

1.1 Description

BloodHub aims to bring many easiness to blood donation process. There will be two types of users. First one is the users who want to donate their blood. They can see available places to donate their blood from interactive map of application. They will be notified when there is a need for blood. Application will send an notification alert to their phone. Users from the near of place where there is a need for blood will be alerted. Their location will be taken with their phone’s location services. Second type

of users are people who look for blood for their friends, relatives or themselves. They can interact with volunteers by using BloodHub.

For the patients who are unconscious and unable to use app, there will be tutelage system. Prioritized assigned people can use patient's account to search for volunteer. They can send notification to users who located nearby them or directly communicate them by nearby users' communication information.

We aim to supplying blood to answer all expectations. There are many humanitarian people to donate their blood for the ones that they never met in their entire lives. We want to make processes easier for these generous people.

1.2 Constraints

1.1.1 Sustainability

- Bloodhub will be updatable from Google Play Store if there is an update for the user.
- There will be scalability about storage and the server capability if number of user goes higher.

1.1.2 Economic Constraints

- Bloodhub will be non-commercial android application.
- Server and storage cost will be supplied by the program developers and in the future it is thought to all the bills will be shared among the charities.

1.1.3 Security Constraints

- There will work a person check algorithm which use Turkish Citizen Number (TCK) *or* Foreign Identity No and birthdate for checking from the Government Database. This authorization check takes up to 3 seconds.
- Confidential information will not be shared with 3rd parties.
- Only the last 3 digits of the TCK *or* Foreign Identity No will be stored.
- User's credentials will be encrypted with MD5 One-way Encryption Protocol.
- Turkish Citizen Number and Foreign Identity No will be checked by SOAP (Simple Object Access Protocol). Security will be established between SOAP client and government web service.

1.1.4 Implementation Constraints

- Users will be notified with updates via Android Push Notification (GCM ^[2] (Google Cloud Messaging))
- This is where you provide the details of the results of your analysis work.
- Java will be used on development of front-end with Android Studio.
- PHP5 will be used on back-end.
- Back-end of the program will supply API^[2] (Application Programming Interface) to the front-end of the program.
- Back-end will be up all the time.
- There will be an option for Foreigners to use system with their Foreigner Identity No instead of TCK.

1.3 Professional and Ethical Issues

Firstly, our program will be collecting some fragile information from its users. This data of course includes their name, surname, blood type, their address, telephone numbers, emails and their last 3 digits of Identity Number. Information

will be recorded for program to be used properly. We are going to supply End User License Agreement and Privacy Policy to give reliability to our users. We will collect data as less as possible, furthermore we will be granted by user for some data , location of the device i.e. . We won't be able to access directly their secluded data. Also the passwords, usernames, emails stored on our database may or may not be encrypted with one-way encryption protocol (on 1.1.3 explained in detail) for to secure data access from us to their privy data.

Secondly, there will be user who don't want to share his/her phone number with the system but the phone number will be necessary in places where there is a low bandwidth of internet which might delay the communication between system and users. So on rural areas *or* low bandwidth internet supplied areas (2G i.e.) phone number will be a necessity for user to supply.

Finally, we, as Bloodhub., will be abide by the Code of Ethics which was epitomized by National Society of Professional Engineers ^[3].

In addition to these, for security purposes user will be using his/her own data when he/she signed-in with his/her Facebook, Gmail, Twitter etc. account or he/she can login with his/her native account that is stored on our server. On the security part of logging-in OAuth 2.0 Framework ^[4] will be used. This framework is preventing security flaws which is outlined in RFC-6819 ^[5].

2 Requirements

2.1 Functional Requirements

Users should be able to create their unique account using their own real name, citizenship number, blood group information, e-mail address, date of birth and telephone number. Besides that users should be able to add their home address optionally.

- Users should be able to confirm user agreement before entering application.
- Users should have their own unique password and username so that users can enter application using them via "Login" page.
- Users who want to donate blood should be able to provide some prior condition such as optimal weight, have or not to have tattoo etc.
- Users should be able to provide information about selected friend or relatives who has same blood group with them for emergency blood donation.
- The system should provide information about close blood donation tents, blood donation camps, hospitals to users who want to donate blood using their location information. Besides that, the system should be able to show these health centers in Google Map.
- The system should send notifications about close blood donor to users who make a request for specific blood group if users' blood group match with close donor's blood group.
- The system should send notifications to users in order to inform blood donation campaigns. Users should be able to shut off these notifications.

2.2 Non-Functional Requirements

As usability, our application should be used by anyone regardless of their age or knowledge about software systems.

Our application should provide simple, clean and easy user interfaces. Users must not have any problems with donating blood or requesting blood donation. We can say that it will be user friendly and stable.

- As reliability, our application should continuously collect real-time data about blood donation tents and blood donation camps.
- Users should not be able to change any information related to other users. Application should require an account to login

- As portability, our application should be able to be used in various hardware and software platforms
- As efficiency, our application should be quick to match requests and donations.
- The response time of the application should be less than 100 milliseconds.
- The creating request of blood donation should take less than 10 second.
- As extensibility, our application should be ready to new features and functionalities.
- Our application should be both extensible frontend and backend therefore new donation's functionalities can be added easily.
- As supportability, our application should require Internet connection
- Our application should support Android and iOS.

3 Works Cited

- [1] "Documentation", Firebase.google.com, 2017. [Online]. Available: <https://firebase.google.com/docs/cloud-messaging/> [Accessed: October 8, 2017].
- [2] "Application programming interface", Wikipedia.com, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Application_programming_interface [Accessed: October 8, 2017].
- [3] "Code of Ethics | National Society of Professional Engineers", Nspe.org, 2017. [Online]. Available: <https://www.nspe.org/resources/ethics/code-ethics>. [Accessed: October 8, 2017].
- [4] "OAuth 2.0", oauth.net, 2017. [Online]. Available: <https://oauth.net/2/> [Accessed: October 8, 2017].
- [5] "RFC 6819 - OAuth 2.0 Threat Model and Security Consideration", rfc-base.org, 2017. [Online]. Available: <http://www.rfc-base.org/rfc-6819.html> [Accessed: October 8, 2017].