



Bilkent University

Department of Computer Engineering

Senior Design Project

Quex

Project Specifications

Barış Ardiç 21401578

Emir Acımiş 21201233

Mert Kara 21400976

Umutcan Aşutlu 21301093

Atakan Özdemir 21301134

Supervisor: Çigdem Gündüz Demir

Jury Members: Hamdi Dibeklioğlu, Fazlı Can

Innovation Expert: Bora Güngören

Sep 9, 2017

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.

Table of Contents

1.Introduction	2
1.1.Description	2
1.2 Constraints	3
1.2.1 Social & Ethical Constraints	3
1.2.2 Implementation Constraints	4
1.2.3 Sustainability Constraints	4
1.3 Professional and Ethical Issues	4
2.Requirements	6
2.1 Functional Requirements	6
2.2 Non-Functional Requirements	7
2.2.1 Extensibility	7
2.2.3 Usability	7
2.2.4 Availability	7
2.2.5 Reliability	8
2.2.6 Portability	8
2.2.7 Scalability	8

1. Introduction

The problem of finding an expert has always been a problem that several communities tried to come up with a solution. However, in order to validate expertise such platforms could not stay generic. Q&A sites like Stackoverflow[1] are successful because they are as specific as they can get, than validation is not as necessary as before. Our approach to this problem is different, to provide experts in broad and generic matters we have a base point of validation by background and in time users of our program will cross validate each other. The purpose of the program is to provide experts in a social and mobile platform while using locality.

1.1. Description

Quex is a mobile social media application which aims to find its users trustworthy individuals to help them with the problems encountered in their daily routine. The program provides a platform that connects the user to another user with required skills to try and solve any problem that can occur such as computer hardware or software issues, daily decision-making, travelling or dining.

The basic principle of the program assumes that there exists a user with a problem, the user notifies the program of that problem and gets assigned to an another user of the program who is close by if that user is considered an expert on the topic based on user background, in-app ratings and user history with similar problems. After matching the user's Quex then allows these users to start a conversation in order to come up with a solution.

We are trying to solve the problem of “expert finding,” therefore, there is a noticeable amount of features regarding expertise validation. The program is designed to fuse certain methods in order to determine which user would be the most appropriate and helpful for which problem. Quex considers the feedback from users and the user’s preferences in their profiles. The program uses the GPS data of its users to consider how close the potential expert is to the user with the problem. The closest possible expert is generally the best match. However, there also exists situations where the proximity of the expert is irrelevant, so the user can ignore the generic matching method and push the notification directly to main dashboard under a certain

category where the problem can be seen by users without matching. This use cases that benefit from locality can present themselves in a variety of situations.

Consider that you are in a university campus and you are having trouble with your computer. A simple notification to Quex can quickly refer you to an engineering student whom may be able to help you instantly. In completely different setting imagine that you are standing in front of the movie theater and trying to decide which movie to see, notifying the program will potentially result in a match with another user who just come out of a movie in that theater. The locality (user GPS data) aspect of the program provides the user with exclusive information that cannot be found on the internet such as how to connect to Bilkent Vpn for the first use case example or information about the sound system of the movie theater for the second example.

1.2 Constraints

1.2.1 Social & Ethical Constraints

- The users must have a way to report other users if they are abusing the programs chat features.
- Quex should minimize and not encourage misuse of the program by auto censoring the messages.
- The program should state that user's GPS data will be used for certain features.
- The development team and all of our products will comply to Code of Ethics by the National Society of Professional Engineers [2].
- A user profile can only be controlled/edited by its unique user account.
- User location information is private and only usable for certain system features.
- Quex will disable users from moving the conversation to a third party platform.

1.2.2 Implementation Constraints

- To follow the contribution on the project we use the GitHub platform.

- The application will be developed for the most contemporary version of Android at the time of implementation.
- The application UI will be designed with usability and functionality in mind.
- The application should use minimal amount of data while its working in background in order to not drain battery.
- The application will use Google Maps API for location information.
- The application implementation will be done in multiple phases, at each phase the user base to test the application will get bigger.

1.2.3 Sustainability Constraints

- The user profiles on expertise will be updated using feedback from other users.
- The application features may be tweaked according to user feedback after the beta version is distributed.,
- The application will be free to install and there will not be any micropayments.

1.3 Professional and Ethical Issues

There are several professional and ethical issues that must be dealt with about Quex, that could be broadly listed as validating the correctness of an answer(help), avoiding the misuse of program, and privacy.

Validation of the correctness of an answer is significant both professionally and ethically. Quex's main objective is to provide help to its users from experts, so the reliability of helps is professionally a must, as the program would be useless if the answers are not reliable. In addition, there is an ethical responsibility to avoid any harm that users could get, as they could possibly do whatever the "expert" suggests, without questioning. Unfortunately, there is no feasible solution to validate every help. Therefore, we will rely on and promote the good intentions of the Quex community. The users will be shown the user license agreements when they sign up for the program. This agreement will include the community rules, which will emphasize the importance of giving correct answers to problems people face and that any false-informing "help" could result in the banning of the user permanently. The agreement will

also mention that users should not practice every help without questioning, and also that although Quex tries to connect people who are in need to experts in a short amount of time, it is especially designed to solve simpler problems, and that it could mostly be used just to have an idea in more serious contexts, such as getting medical help. Although we believe that the community will use the application responsibly and good-intentioned, we are aware that it is not possible to ensure that every community member would behave so. Therefore, we will also use a reactive approach with reporting mechanism.

The program could be misused in a wide variety of ways. One of the most serious examples is harassment. We are ethically obliged to avoid any possible harassment, and to do so, we proactively avoid users seeing other's personal information, including full names and location. Users will only be able to see the expertise and comments about people who they will possibly get help. Depending on the solution, and of course with user's own consent, they will be able to share their location, in order to meet up to solve the problem. However, they will always be able to reverse this action if they change their mind, and their location will be hidden again. In addition to this anonymity, we would again rely on the reporting facility to ban the possible harassers permanently.

Finally, we may add a machine learning or keyword search technology to avoid any kind of harassment speech beforehand. Another most possible misuse issue could be advertising and plagiarism. To avoid the misuse of advertising, Quex will censor any sharing of phone number. In addition, reporting facility is expected to work effectively, as advertising is against the logic of using the application. However, avoiding plagiarism issue is more difficult, since Quex is also expected to be used for getting help for studying and learning, and it may be hard to make the distinction. Although we are relying on reporting mechanisms and user's responsibility for now, we are aware of the issue and thinking on possible solutions.

Privacy is another important issue. Quex needs access to the location of user and some personal information for validation purposes. Users will be informed about the privacy policy of Quex, and these accesses would only be possible with user's consent. This information will not be shared with third-parties, and they will be held in the most secure way possible.

2. Requirements

2.1 Functional Requirements

- Users should be able to login to the program via Facebook or Google account. If these are undesirable, user can create an original account.
- Program will ask user to create a profile including areas of knowledge and proficiency.
- In case a password is lost, it should be possible to recover it via e-mail verification.
- The program should be able to update user profiles.
- Users should be able to send problems to the program with or without a certain fixed category.
- To match with the closest expert, program uses the GPS data of the users. This data is taken from the Google Maps API.
- User can accept or decline a match. If the match is accepted by the user with the problem, a notification will be sent to the expert.
- If the match is accepted by both parties, the program will provide for a real time conversation via text messaging.
- The message traffic between users is monitored by the program in order to block certain behavior that is considered inappropriate or irrelevant. For example, it should not be possible to send a phone number or slang words.
- In case a need arises for users to meet, location data can of a user can be sent to another voluntarily.
- User will be asked to rate the expert after a case is closed. It is possible to up vote or report a user for their behavior.
- Program must have a dashboard to cover the cases which the location is not of importance or there are no matches. It is possible to send cases directly to the dashboard.
- The program must have fixed expertise categories in order to provide better matching between users.

2.2 Non-Functional Requirements

2.2.1 Extensibility

- Quex will work better as the user base gets larger.
- Quex's side menu system will allow new extensions, these extensions can be suggested by users or determined by the development team.

2.2.2 Supportability

- Quex will be easily adapted to the new technologies or features via new modules. Object oriented design principles will allow for these type of expansions.
- New updates will be regularly provided to make the application up to date and innovator.

2.2.3 Usability

- Quex will be easy to use and easy to understand its working mechanism.
- Quex will have user friendly interface.
- Sign in and sign up functions will not be complicated, user can login via their Gmail or Facebook accounts. The application gets minimal information from the user.
- Quex will provide helps and tips in itself to inform users about the application and its features.
- The application is not needed a long time to perform operations and tasks.

2.2.4 Availability

- Quex server will always be available for the users.
- Quex will be a mobile application which can be accessed very easily.
- The application will be free to use, users can download and use it without paying money.

2.2.5 Reliability

- Quex will be a bug and error free application.
- Quex will detect any undesired substrings with its black-list mechanism including bank account numbers to prevent use case of making profit by the application.
- Secured login system and reputation point system will increase the reliability of the users.
- Personal data of the users will be preserved well.

2.2.6 Portability

- Quex will be made for Android system but it can be ported to different operating systems like iOS, Windows with a little effort.

2.2.7 Scalability

- Quex will support large number of users.
- Capacity of the Quex will be increased according to number of users.

3. References

[1]“Stack Overflow - Where Developers Learn, Share, & Build Careers,” *Stack Overflow - Where Developers Learn, Share, & Build Careers*. [Online]. Available: <https://stackoverflow.com/>. [Accessed: 09-Oct-2017].

[2] "Code of Ethics | National Society of Professional Engineers", Nspe.org, 2016.

[Online]. Available: <https://www.nspe.org/resources/ethics/code-ethics>.

[Accessed: 09- Oct- 2017].