2QUEUEOOL4U

OCTOBER 31, 2017

PROJECT DEMONSTRATIONS

Part 0: Metadata

Project name: NextText

Team name: 2QueueOOL4U

Team members:

James Mulvenna/100965629

Yue Zhang/100980408

Devon Plouffe/100715712

Part 1: Final Status Report

At the current stage in the development process, the NextText android application is fully functional. Although the weather, and location sub-systems remain incomplete, stemming from the proposed functionalities in the beginning phase of the development, 2QueueOOL4U was able to implement, in essence, most functional/non-functional properties as they concern each event trigger individually. For instance, the functional and non-functional properties proposed include:

Functional Properties:

1. The system should have the ability to save user information
2. The system should be able to import users contacts
3. The system should be able to access conditions
4. The system should list queued messages
5. The system should be able to add new queued messages
6. The system should be able to modify existing queued messages
7. The system should respect user privacy and authorization

Non-functional Properties:

1.1 User information will have the ability to be modified

2.1 Native contacts, Facebook contacts and E-mails contacts will be able to be imported

3.1 Time based services will be used, location based services (API) will be used, weather information (API) will be relayed incrementally

4.1 A list of stored messages will be visual and accessible to the user

5.1 User will have the ability to choose their contact and condition in an "add" interface

7.1 The application will respect the permissions the user has set

As a team, we were able to confidently save user information with the ability to be modified, make conditions accessible using time based services, list the stored messages to provide an accessible user experience, give the user the ability to add and modify messages through a friendly interface, and do all this by respecting the user's privacy authorizations. However, where we fell short was on the importation of contacts from platforms such as Gmail, Facebook, and native contacts. As well as, implementing our outstanding event triggers including location, and weather. These sub-systems were dropped due to lack of outstanding development time, and developers. As it stands, our team consists of 3 members, while only 2 put in consistent work on the application as seen in the dev_logs. Where most teams consisted of 4 members, we managed to compile an application following alongside our proposed functionalities, with half the resources. This did affect our developmental progression immensely, which can be clearly seen in the dev_logs. Following this reason, some sub-systems were also dropped by reason of our team prioritizing debugging/testing time as this is a major component of Software Engineering that is often under looked. Considering this, we spent the majority of the interval between D3, and D4 surveying user experience, as well as virtually trying to break our app. By doing this, we can confidently say NextText is commercially ready. Finally, the last major complication we over-came was a complete platform change. At the beginning of development, we envisioned creating this application on iOS. We quickly realized that there were many restrictions set by Apple regarding message automation. However, we initially overcame these restrictions by utilizing 3rd party APIs such as Twilio, and MessageGrid to send messages on behalf of our clients. Soon after, we were banned from these platforms by cause of spam, and with the option to contact Twilio/MessageGrid user-support, or switch to Androids' message automation friendly platform, we clearly decided to go with the latter, as it complemented the existing development time we had, and our functional/non-functional properties. With these reasons in mind, NextText still administers scheduling messages such as SMS, or E-mail based on future times set by the user. As far as the backend system technicalities go, 2QueueOOL4U managed user data utilizing SQL transactions, and performed data transfer by taking advantage of a request/handler style where applicable. Regarding the frontend system technicalities, interfaces were utilized as necessary, providing a commercial ready, encapsulated user experience.

The weekly development logs can be accessed with valid credentials at:
https://github.com/mryuezhang/NextText/tree/master/dev_logs

The promotional video can be found at (If there's a copyright issue with the audio, please contact us):

https://youtu.be/uZZ-qhnb1OU

Part 2: Demo Description

Since the last demonstration (D2), 2QueueOOL4U has allocated time to support data transfer (sending messages) by setting requesters client side, and handlers server side. We have focused attention on providing Gmail services, making message objects fully modifiable, and sustaining exceptional user experiences. Finally, we have spent endless time exhausting our applications functionalities to make sure it is as commercially ready as possible. We will be presenting these additional functions listed above, as well as our typical proposed user scenario regarding an ordinary client we might provide service to. This entails using the application to add a new Email, by setting the content as well as the future time of dispatch. If time permits we will repeat this step, however interchange Email for SMS, to show that message dispatch is fully exploitable. Finally, we will demonstrate features previously demonstrated as it is necessary in a basic user scenario. Withal, during demonstration data transfer pauses, we will discuss the technical limitations we overcame throughout development.