BAOBAB

PLANT US & WE'LL GROW

Document: Architectural Requirements

Project: Group Chat For Linphone (Agile DO-178)

Date: October 28, 2015

Patience Mtsweni, Lerato Molokomme, Tsepo Ntsaba, Mpedi Mello, Lutfiyya Razak, Ephiphania Munava

Github repository:
https://github.com/TheBaobabTeam/linphone-android

Member details

Ephiphania Munava	10624610
Lutfiyya Razak	10198408
Lerato Molokomme	11197961
Mpedi Mello	11210754
Patience Mtsweni	11116774
Tsepo Ntsaba	10668544

Contents

1 Introduction

The purpose of this project is to extend Linphone's Instant Messaging (IM) implementation on Android platforms to include group chat and to implement other minor improvements to Linphones IM capabilities and user interface.

2 Scope

We are required to develop the following functionality for the Linphone project:

- Group chat (Invite additional members to a chat, all members receive chats)

- Secure group chat (AES256)

- Creation and deletion of groups

- Voice record and send over IM

- Improve the messaging user interface

    - Spacing between words are terrible
    - Make text bigger
    - Blocks indents required to better specify who said what
    - Presence indication to show a remote user is typing
    - User picture portraits

3 Access and Integration Channel

3.1 Access Channels

3.1.1 Human access channels

The system will be accessible by human users through the following channels:

- Android devices

3.1.2 System access channels

Linphone provides a library called Liblinphone which is a cross-platform SDK for SIP communication and media processing. It's an easy to use API, that enables other systems such as iOS, Blacberry, Windows phone, Linux, Mac and windows to have access to the system.

3.2 Integration channels

The system will be able to access the following for chat purposes:

- Phonebook contacts

- Pictures

- Sound

4 Architecture Requirements

4.1 Architectural Scope

- Platform independent

  – The system is accessible on the website however the group chat functionality will be accessible only for an android mobile application.

- Distributed databases

  – We are making use of the Linphone database to store the messages.

- Security

  – Confidentiality: Encrypt messages, pictures and voice when users are chatting.

- Backup

  – The system should have an automated data backup.

  – The system should have alternative platforms(i.e. smart phones, etc.).

- Store uploaded content (i.e. voice notes, images)

  – The system should allow the user to be able to upload an image should they wish to.

4.2 Quality Requirements

4.2.1 Usability

Usability is an important quality requirement needed by the system. Usability takes first priority over other quality requirements. We divide usability into this :

- **Learnability**
  The interface should be designed in such a way that it is easy to use and learn. If the user is a first time user then the system should not be hard to understand. The interface should not be complex for the user, it should include basic functions and fewer things to learn and remember.

- **Error Handling**
  The system should be able to handle errors made by users. Correct, prevent or even detect them. If there is no group name entered it will let the user know before it moves to the next screen. Users will be allowed to perform certain operations based on their user level. Some functions will be deactivated for certain users. For example a member cannot removed other members from the group, only the administrator can do that. Such functionality will be hidden from the user preventing errors.

- **Acceptability**
  Acceptability is a subjective phenomenon to which many factors contribute, it measures the extent to which users like the system. The system should be created in such a way that the users are comfortable with and they actually like it, this can be achieved by a technique called Usability Testing where we have users that test our system and get feedback from them by giving them questionnaires which will require answers based on the system they interacted with.

- **Effectiveness**
  The system should be effective in such a way that it does the right tasks, completing activities and achieving the required goals. It should allow users to exchange messages in a group and send voice recording.

4.2.2 Security

Security is an important issue in the group chat. A Secure system focuses on three main aspects, namely: to detect attacks, resist attacks and recover from attacks.
We will, however on the Linphone group chat focus on the resisting attacks. The messages that are exchanged in the group chat will be encrypted before being sent and minutes to discourage bot attacks, this might, however, have an impact on usability.

4.2.3 Availability

The system must to be available when needed by the users and by the members of a group chat for communicating

4.2.4 Reliability

The system should be reliable in such a way that it will be easy to implement or make changes on it should any errors or failures occur, the system should be able to handle them and recover easily.

5 Architectural Tactics or Strategies

5.1 Maintainability

In order to improve maintainability and reduce the skills requirements for the team who needs to maintain the system, the software architecture of the system has been split into submodules, which allows for flexibility and easy maintainability because a dedicated team will be in charge of a specific submodule. This allows for automatic updating, if the submodules are update, the system will automatically upgrade the source code for future downloads.

5.2 Reliability and Availability

Linphone is hosted by Belldone communications, thus it is hosted privately even though it is open source, thus ensuring that it will always be available and reliable.

5.3 Security

There are a number of tactics we are using to ensure security. The first is authentication, a user has to register and be authenticated to chat using Linphone. Further security measures we implemented are that we are using the ZRTP protocol to encrypt voice calls. We are also using the ORT protocol to encrypt the instant messages and to ensure no third parties can eaves drop on a conversation. We are also using the Socialist Millionaire Protocol to detect man in the middle attacks and eaves dropping.

5.4 Testability

Continuous Integration and Development is a agile tactic that we are using for ensuring that we test the code efficiently. We also use unit tests to test the code based on the service contracts.

5.5 Usability

The user interface has been separated from the rest of the application. Since the user interface is expected to change frequently both during the development and after development, maintaining the user interface code separately will localize changes to it. Usability can only be achieved through rigorous testing and modification, satisfying the HCI components such as error handling

and proper feedback when a user clicks on a button etc. Using templates has also been used to enforce a consistent user interface and maintainability.

5.6 Intergrability

Since Linphone has a modular design, meaning it has been subdivided into smaller parts called modules or submodules it is easy to integrate new submodules and functionalities.

Since we are also using continuous integration, which means merging several developers code in to the mainline several times a day, to solve integration problems.

6 Architectural Components

6.1 LibLinphone

Liblinphone is a high level library integrating all the SIP video calls feature into a single easy to use API. Usually telecommunications is made of two things: media (transport of voice or video, encoding and decoding), and signalling (routing calls, ringing, accepting a call). Liblinphone aims at combining the two things together and doing most things automatically. This makes it easier to the programmer to implement video calls in any application, without being an expert in VoIP and telecommunications. Liblinphone is an open source library based on Mediastreamer2 for voice/video streaming, and belle-sip for SIP signalling.

6.2 Belle-SIP

Belle-sip is a C object oriented SIP Stack.

7 Technologies

- For version control, we are using github. It has our git repository, which has a branch for each feature or bug. The branches are only merged with the main trunk once it passes the unit test and travis-ci for th continuous integration

- IDE we are using is Eclipse

- We are using travis-ci for builds. It is used for building and performing continuous integration or Continuous deployment. If the build fails then we do not merge the new feature or bug fix with the main trunk until it can build with Travis which is linked with our git repository.

- We are using Trello for our scrum board, where there is our product backlog with all our user stories to be implemented, it has the sprint backlog for each sprint, it has a list of completed user stories and those that still need to be completed.

- The group chat user interface is coded in java (Android) and for unit test, Android Testing Library is used

- The group chat back-end functionality is coded in C and for unit testing, CUnit is used.

8 References

- Linphone information, Avaiable from: https://www.linphone.org [Accessed: 28 August 2015]