**Icon

Description automatically generated**

**CIS 634 Object-Oriented Software Engr**

**Group 11**

**Secure Chat App Test Specification**

**2836088 – Abdul Rehman**

**2836430 – Tejam Reddy**

**2828837 - Varun**

**2837055 - Sravan**

**1.0 Introduction**

Secure chat app Android chat app, which is a chat application used for personal and office purposes. A friend or office colleague can share any kind of private data through this application. It will help to chat with friends and office colleagues. Mainly, it will help people who want to hide their conversation. This app allows users to chat securely and create groups for chatting to keep privacy in the group. Privacy is the top priority of this application. Languages that have been used are Java and XML, developed in Android. Android studio is the tool that is used for this chat application development.

**1.1 Goals and objectives**

The main goal of a secure chat app is a secure messaging app featuring end-to-end encryption as well as the ability to encrypt your messages so no one can read them. You can safely text a friend or family member your personal details like banking info, phone numbers, health information, and much more

* Sign in or Sign up using your phone number
* One to One chat
* Encrypt and decrypt messages
* Preventing screenshots during chatting
* Firebase Real-Time Chat Integration
* Group chat
* Status Updating Feature
* Beautiful Material Design
* Sweet and Clean animations
* Image Sharing

The testing process for Secure Chat App has a number of goals. The software will be thoroughly tested for coding bugs and logic errors. In addition to testing for bugs, Secure Chat App will be tested to ensure that it is of the utmost quality. It will be expected exhibit the following qualities: well-executed app, high production values, easy to use interface with a complete privacy in android.

Testing of the Secure Chat App is the divided into multiple steps. All steps will be performed one by one. Unit testing, Integration testing, UI testing, validation testing and alpha testing will be performed.

**1.2 Statement of scope**

Secure chat app will be tested on a number of levels, beginning with unit testing (using white box testing methods), integration testing (using black box testing methods), validation testing, and ending with alpha testing with all project members. A number of design principles will be validated during the testing process. The interface should be easy to use. Data should read and write flawlessly from the database/firebase.

The app should not be crashing while in use. Real time data processing should be implemented. Users should be able to use all the functions of the app with an ease of interface. Account creation is automatic so there should be no problem from app to produce errors. Encryption and decryption are the main functions of this app which should be working all the time and user should be able to use them as per need.

The secure chat app is an Android chat app, focused on individuals who have trouble following safely texting with loved ones and want to hide their conversation. It is developed in a Java and XML environment, we will be testing the following option available on the application.

* Dashboard
* Chat
* Profile
* Contacts
* Settings
* Create group
* Group chat
* Encrypt message
* Decrypt message

**2.0 Test Plan**

Different steps are involved in the testing of the secure chat app. There are multiple methods available for the testing of software. The best method of testing is that which captures a maximum number of errors at the time of testing. By using a combination of multiple testing we will be testing our app. A short description already provided as a secure chat app will be tested on a number of levels, beginning with unit testing (using white box testing methods), integration testing (using black box testing methods), validation testing, and ending with alpha testing with all project members. A number of design principles will be validated during the testing process. The interface should be easy to use. Data should be read and written flawlessly from the database/firebase.

**2.1 Software to be tested**

Secure chat app is being tested with different components. Details are given below:

* Interface components to be tested
* Registration activity layout
* Login layout
* Firebase Read/Write
* Input handlers
* Adding friend
* Start chat
* Encrypt message
* Decrypt message
* Make group
* Backend functionality to be tested
* Object handlers
* Layout parser
* Identifiers
* Database handler
* Data loaders
* Data encryption handlers
* Error handlers
* Functionality

To test the functionality of the application unit testing will be performed. Luckily, android studio supports the unit testing while using app in the emulator and show all the activities, errors and crashes at runtime.

* Application launching

Application launching needs connection to firebase over internet. It will be tested.

* Integration Test

As the secure chat app is following a modular design, top-down and bottom-up integration will occur. The app will be integrated incrementally, to control the number of bugs that need to be fixed at any given time. The app will be integrated in the following order: layout handling, input handling, firebase handling, and logic handling. Tests will be conducted in black box testing.

* Errors

Errors that occurred during the testing will be documented and will be resolved by the development team after testing.

* Input’s information (Importing the data)

Importing data from firebase or entering the new data will be tested. Incorrect inputs will be provided and it will be tested to if it gets passed or not as example we can use special characters for registration numbers.

* Security

Security testing will be performed to ensure the credibility of the app. All possible loopholes will be tested.

* Field Test

Combined app components will be tested as a whole. To maintain maximum control over the testing criteria, all data files will be made specifically for testing purposes. The chat function will be tested to ensure proper communication between the interface and the database. Testing will be done in black box mode.

* Performance

The performance of the secure chat app will be tested by the firebase response. It is used to make as fast as real-time working.

* Stress Test

Stress testing will be performed to measure the level of the application as a good or worse response. By giving more than specified inputs and using the app in disorder trying to create the group before adding friends.

* Application Interface

Alpha testing will be performed by the project members and they will be able to submit interface responses with any kind of suggestions as concerning the application.

**2.3 Testing tools and environment**

*Testing tool and Environment*

Android studio and an emulator or external android phone will be used as testing tools and environments.

*Specialized hardware*

No specialized hardware is required for this testing. External android phones can be used for testing, but it is not compulsory.

*Document*

Microsoft Excel will be used to document the test results with Id, test case name, tester details, error/bug/mistake, and description.

*Resources*

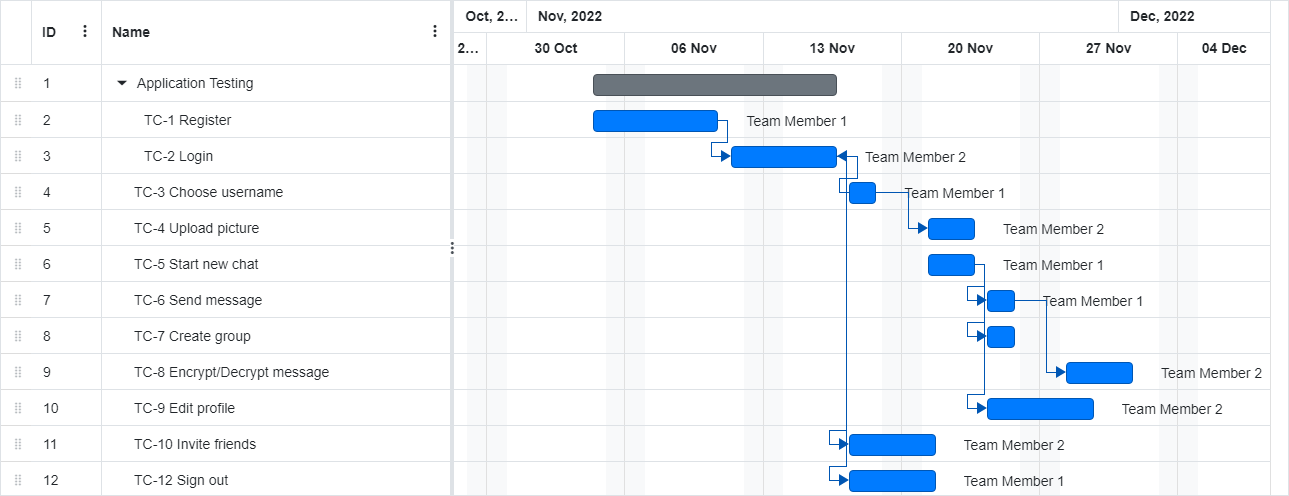
No special resources are required beyond those already needed for development.

*Staffing*

Project members will be testing the application.

**2.4 Test schedule**

A detailed schedule for testing is described.



**3.0 Test Cases**

This section enumerates a complete list of test cases for the software.

|  |  |
| --- | --- |
| ID | **TC-1 Register** |
| Test Input | Phone number |
| Expected Output | Verification message should be received |
| Description | To register on the app phone number will be entered and a verification from firebase will be sent to ensure security. |

|  |  |
| --- | --- |
| ID | **TC-2 Login** |
| Test Input | Phone number |
| Expected Output | Verification message should be received |
| Description | To log in to the app phone number will be entered and verification from firebase will be sent to ensure security. |

|  |  |
| --- | --- |
| ID | **TC-3 Choose username** |
| Test Input | Enter the string for the username |
| Expected Output | User name should be assigned |
| Description | To choose a username the app string will be entered and data will be saved into firebase. |

|  |  |
| --- | --- |
| ID | **TC-4 Upload picture** |
| Test Input | Click on the button to upload a profile picture |
| Expected Output | Picture should be uploaded |
| Description | To upload a user profile picture gallery should be appeared for choosing pictures. |

|  |  |
| --- | --- |
| ID | **TC-5 Start new chat** |
| Test Input | Click on the button and select contact |
| Expected Output | Contacts should be appeared to start the chat |
| Description | To start chatting on the app clicking on the button will lead to new activity where a contact needed to be selected and starts a chat. |

|  |  |
| --- | --- |
| ID | **TC-6 Send message** |
| Test Input | Click on the button, select contact, Enter a character send a message |
| Expected Output | User should be able to send a message. |
| Description | To send a message from the app clicking on the button will lead to sending a message. |

|  |  |
| --- | --- |
| ID | **TC-7 Create group** |
| Test Input | Click on the button and select contact |
| Expected Output | Contacts should be appeared to start the chat |
| Description | To create a group on the app clicking on the button will lead to new activity where a contact needed to be selected and create group. |

|  |  |
| --- | --- |
| ID | **TC-8 Encrypt/Decrypt message** |
| Test Input | Click on the button |
| Expected Output | Message should be encrypted/decrypt with key. |
| Description | To encrypt/decrypt on the app clicking on the button will lead to function where contact needs a key to encrypt/decrypt messages. |

|  |  |
| --- | --- |
| ID | **TC-9 Edit profile** |
| Test Input | Click on the button, Enter a new username, and Enter new status. |
| Expected Output | Username or status should be updated. |
| Description | To update the profile on the app clicking on the button will lead to a function where the app needs input to update the status. |

|  |  |
| --- | --- |
| ID | **TC-10 Invite friends** |
| Test Input | Click on the button. |
| Expected Output | Contacts should be appeared to share links. |
| Description | To share the app, clicking on the button will lead to a layout where the app link can be shared. |

|  |  |
| --- | --- |
| ID | **TC-11 Share .apk** |
| Test Input | Click on the button. |
| Expected Output | Contacts should appear to .apk file. |
| Description | To share the .apk file, clicking on the button will lead to layout where .apk can be shared. |

|  |  |
| --- | --- |
| ID | **TC-12 Sign out** |
| Test Input | Click on the button |
| Expected Output | User should be sign out and login layout should be appeared. |
| Description | To sign out from the app clicking on the button will lead to login activity. |