**Icon

Description automatically generated**

**CIS 634 Object-Oriented Software Engr**

**Group 11**

**Secure Chat App SDS**

**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. **Goals and objectives**

The main goal of 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

**1.2 Statement of scope**

This document is intended to give a detailed technical description of the Secure Chat App project. It does not, however, explain the secure chat app standard itself, or the rationale behind the implementation or standard.

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, The Main Features of the application.

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

**1.3 Software context**

The software is placed in a business or product line context. Strategic issues relevant to context are discussed. The intent is for the reader to understand the 'picture'.

This application is based on privacy and secure chatting among friends, family and office colleagues. Different kind of functionalities have been implemented to make sure the high ratio success of this chat app. Functions like encryption and decryption, Group chatting with security and easy use of application are the important factors of this developed application.

**1.4 Major constraints**

* All database maintenance will be handled by the users and it is available when the user needs.
* Data is secure enough that no user can see the personal information of the other user.
* Data must be restored in any case loss of data.

**2.0 Data design**

This app uses firebase database presented by the Google for the convenience of the users. It is a real-time database that provides the services for the developers to initiate projects for testing and then it charges on increment of the 50+ users.

We are using android system to implement this service of secure chat app. The data structures is described below.

**2.1 Data structures**

Data structured that are available to major portions of the architecture are described.

**2.2 Database description**

An android application is developed with the privacy of data, messages and provide full security using real-time database. This app uses firebase database presented by the Google for the convenience of the users. It is a real-time database that provides the services for the developers to initiate projects for testing and then it charges on increment of the 50+ users. Every person can use this application as per need. This application provide some special functions like data encryption and decryption with using different algorithms. Group chat option is available to chat in group securely.

**3.0 Architectural and component-level design**

A description of the software architecture components is presented.

Profile

Create Group

Encrypt/Decrypt algorithm

Send/Receive message

Contacts

Settings

**3.1 Architecture diagrams**

Diagram

Description automatically generated

**3.2 Description for Components**

A description of major software components contained within the architecture is presented.

**3.2.1 Component 1 description**

**3.2.1.1 Interface description**

**Input**

Button click only required. No special input for encryption of the message. But for the decryption user has to enter key to decrypt message.

**Output**

User will send encrypted message and will be able to decrypt the received message after entering the key.

**Exceptions**

The message is encrypted or decrypted.

**3.2.3.2 Static models**

**Class diagrams**

Decrypt

Encrypt

-DecryptAlgo()

-EncryptAlgo()

**composite structure diagram**

**3.2.3.3 Dynamic models**

**Activity diagrams**

Sender

Reciever

**sequential diagrams**

User request Application Database

Response

**3.3 External Interface Description**

No external design is needed. Only application designs are implementable and shared.

**4.0 User interface design**

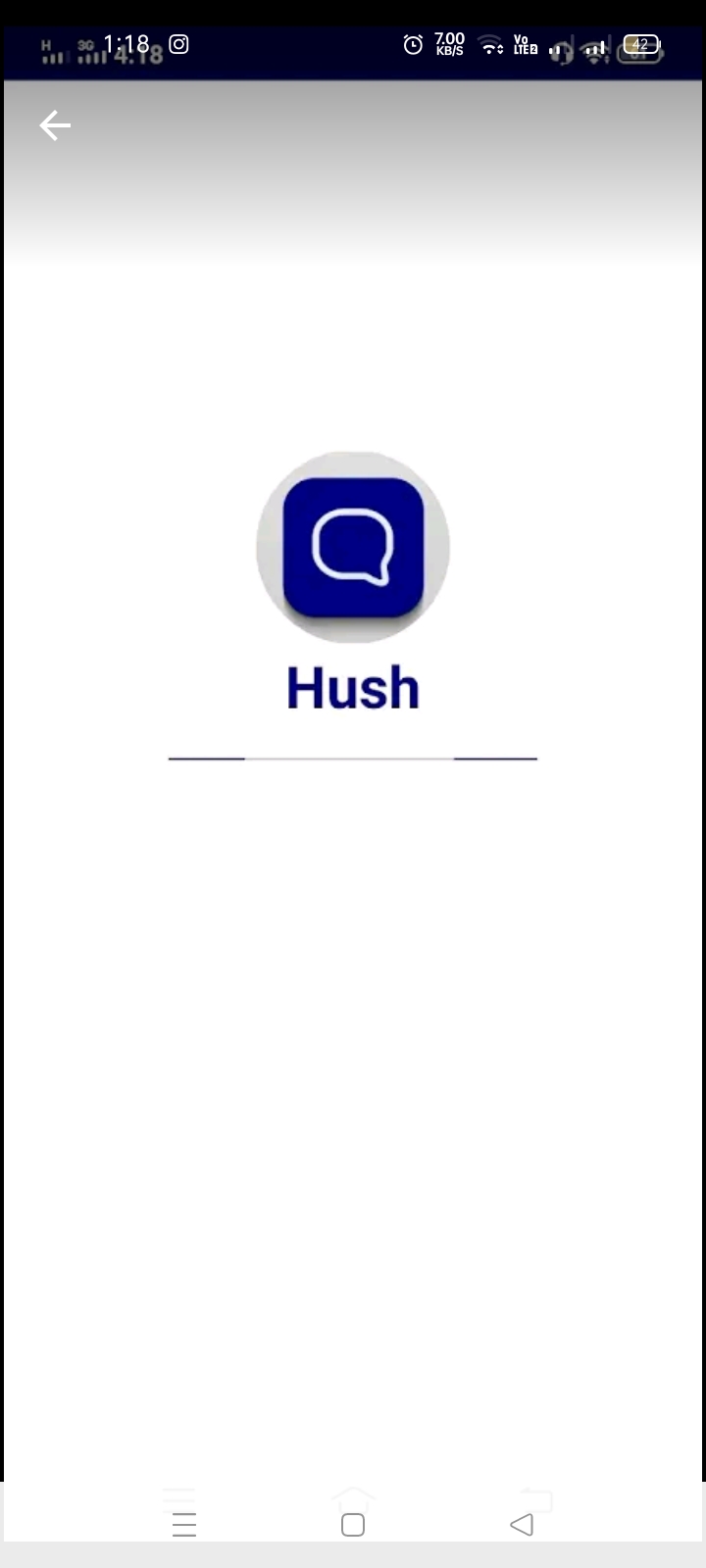
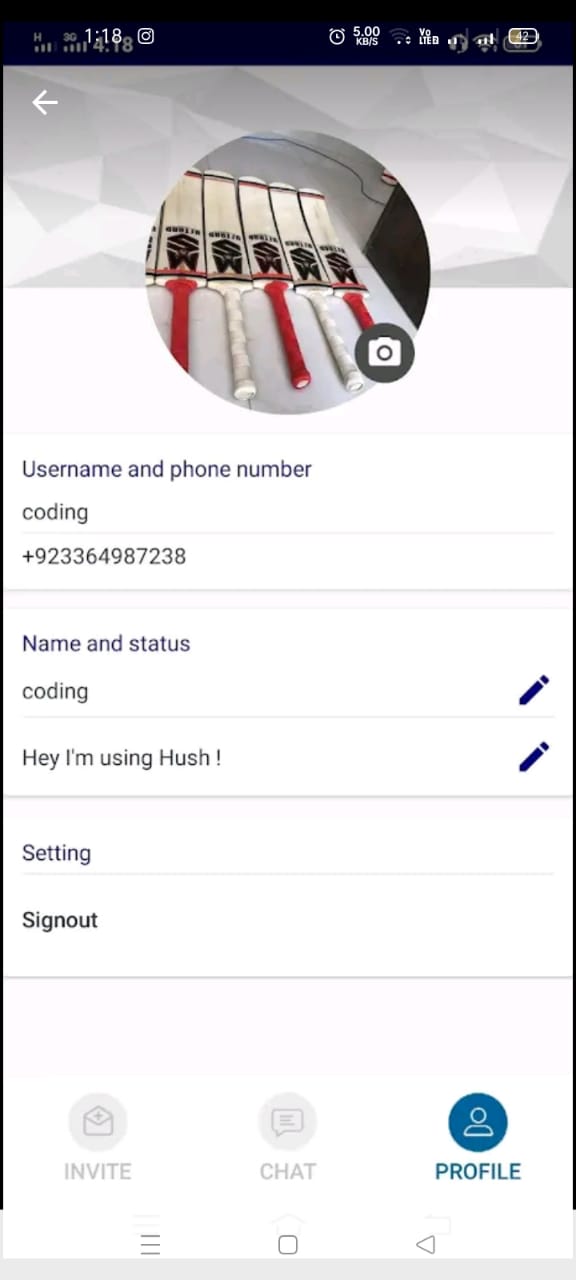
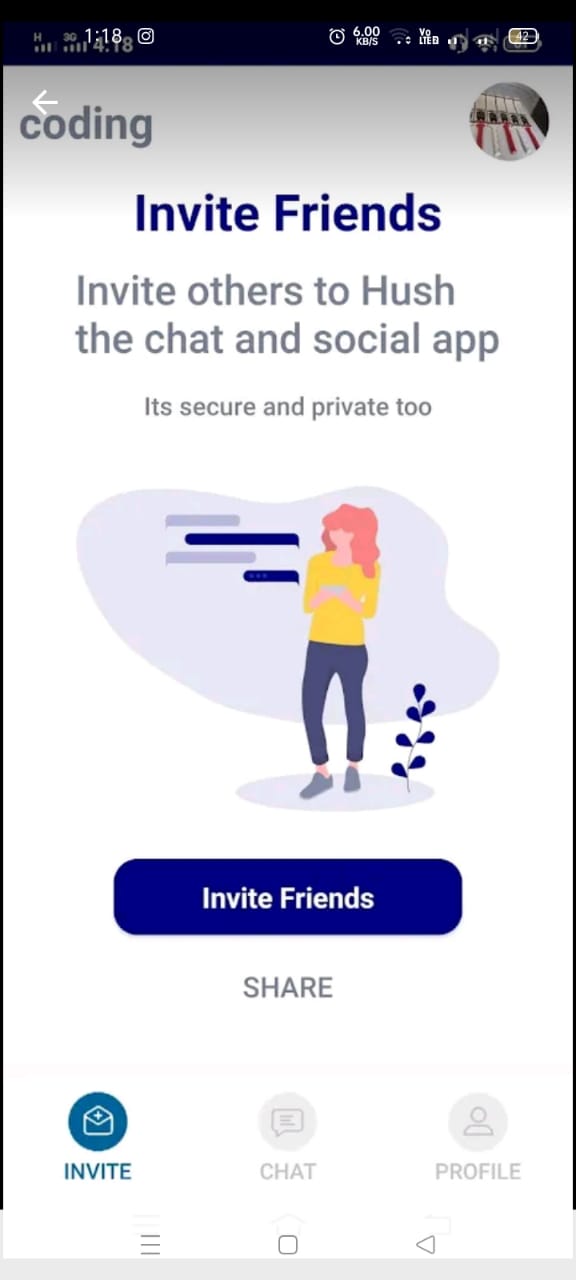
A description of the user interface design of the software is presented.

**4.1 Description of the user interface**

A description of user interface including screen images or prototype is presented.

Graphical user interface, application

Description automatically generated



**5.0 Restrictions, Limitations, and Constraints**

Special issues which impact the specification, design, or implementation of the software are noted here.

**5.0.1 Restrictions**

* Unauthorized users do not use this system.
* To use this system users will be registered first.

**5.0.2 Limitations**

* Android based application
* Only firebase implemented specifications are usable
* Do not support cross platform working

**5.0.5 Constraints**

* All database maintenance will be handled by the users and it is available when the user needs.
* Data is secure enough that no user can see the personal information of the other user.
* Data must be restored in any case loss of data.
* The Internet should be available to use this system.
* All the records of the User stored in Database.

**6.0 Appendices**

Presents information that supplements the design specification.

**6.1 Requirements traceability matrix**

Currently, Not applicable until the completion of development of the project.

**6.2 Implementation issues**

At the start there were some issues with the android studio but those were resolved with updating new Gradle files and updating android studio. We have already defined the roles of the team members and negotiated issues in every scheduled meeting for the development of the project. Currently, there is no issue while implementing of the project.