ChatConnect -
A Real-Time Chat And Communication App

Surya S

8F486A332F96B4B99C640BE808C16E3C

Swadhi K

D07D54CE365C10EB0C84A695595E1628

Ganga Raj R

B80DC7C10054E5D982440097D878E03A

Darshan Jebaraj D

6000B0D29588CC4B6A94CBBE33BA0401

DESCRIPTION

ChatConnect is an interactive communication platform designed for seamless messaging, built with Android’s Compose UI toolkit. It enables real-time text messaging, making it easy for users to connect and communicate efficiently.

- **Compose UI Integration**: ChatConnect showcases Compose's modern, declarative approach to UI design, allowing smooth, responsive, and adaptive user interfaces.
- **Real-Time Messaging**: With efficient state management, messages appear instantly as they are sent and received, enhancing the immediacy and fluidity of conversations.
- **File Sharing**: ChatConnect supports sharing various file types, including images, ZIP files, PDFs, and more, making it a versatile tool for sharing multimedia and documents during chats.
- **Firebase Integration**: Utilizing Firebase, ChatConnect provides a robust backend for message storage and retrieval, enabling real-time updates in chat windows.
- **User-Friendly Navigation**: Through composable functions, the app demonstrates seamless navigation, enhancing usability and guiding users intuitively through chat threads.

ChatConnect serves as a foundational project for developers looking to understand and implement chat functionalities in Android, leveraging the power of Compose for a modern and streamlined UI experience.

ChatActivity.kt:

```
package com.chat.chatapp

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.EditText
import android.widget.ImageView
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.database.*

class ChatActivity : AppCompatActivity() {
    private lateinit var chatRecyclerView: RecyclerView
    private lateinit var messageBox: EditText
    private lateinit var sendButton: ImageView
    private lateinit var messageAdapter: MessageAdapter
    private lateinit var messageList: ArrayList<Message>
    private lateinit var mDbRef: DatabaseReference

    var receiverRoom: String? = null
    var senderRoom: String? = null
```

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_chat)

    val name = intent.getStringExtra("name")
    val receiverUid = intent.getStringExtra("uid")

    val senderUid = FirebaseAuth.getInstance().currentUser?.uid
    mDbRef = FirebaseDatabase.getInstance().getReference()
    senderRoom = receiverUid + senderUid
    receiverRoom = senderUid + receiverUid

        supportActionBar?.title = name

    chatRecyclerView = findViewById(R.id.chatRecyclerView)
    messageBox = findViewById(R.id.messageBox)
    sendButton = findViewById(R.id.sentButton)
    messageList = ArrayList()
    messageAdapter = MessageAdapter(this, messageList)

    chatRecyclerView.layoutManager = LinearLayoutManager(this)
    chatRecyclerView.adapter = messageAdapter
```

```
  //        logic for adding data or chats to recycler view
        mDbRef.child("chats").child(senderRoom!!).child("messages")
            .addValueEventListener(object : ValueEventListener {
                override fun onDataChange(snapshot: DataSnapshot) {
                    messageList.clear()
                    for (postSnapshot in snapshot.children) {
                        val message = postSnapshot.getValue(Message::class.java)
                        messageList.add(message!!)
                    }
                    messageAdapter.notifyDataSetChanged()
                }
                override fun onCancelled(error: DatabaseError) {
                }
            })
 //         adding the message to database
        sendButton.setOnClickListener {
            val message = messageBox.text.toString()
            val messageObject = Message(message, senderUid)
            mDbRef.child("chats").child(senderRoom!!).child("messages").push()
                .setValue(messageObject).addOnSuccessListener {
                    mDbRef.child("chats").child(receiverRoom!!).child("messages").push()
                        .setValue(messageObject) }
            messageBox.setText("")
        } } }
```

LoginActivity.kt

```
package com.chat.chatapp

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import com.google.firebase.auth.FirebaseAuth

class LoginActivity : AppCompatActivity() {

    private lateinit var edtEmail: EditText
    private lateinit var edtPasswoord: EditText
    private lateinit var btnLogin: Button
    private lateinit var btnSignUp: Button
    private lateinit var mAuth : FirebaseAuth


    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_login)
mAuth = FirebaseAuth.getInstance()

        edtEmail = findViewById(R.id.edt_email)
        edtPasswoord = findViewById(R.id.edt_Password)
        btnLogin = findViewById(R.id.btn_Login)
        btnSignUp = findViewById(R.id.btn_SignUp)
        btnSignUp.setOnClickListener {
            startActivity(Intent(this, SignupActivity::class.java))
        }

        btnLogin.setOnClickListener {
            val email = edtEmail.text.toString()
            val password = edtPasswoord.text.toString()
            login(email,password)
        }

        val user = FirebaseAuth.getInstance().getCurrentUser()
        if (user != null)
        {
            val intent = Intent(this, MainActivity::class.java)
            startActivity(intent)
            finish()
        }
    }
```

```
  private fun login(email: String, password: String) {

        mAuth.signInWithEmailAndPassword(email, password)
            .addOnCompleteListener(this) { task ->
                if (task.isSuccessful) {
                    // Sign in success, update UI with the signed-in user's information
                    val intent = Intent(this,MainActivity::class.java)
                    finish()
                    startActivity(intent)
                } else {
                    // If sign in fails, display a message to the user.
                    Toast.makeText(this,"user not exist", Toast.LENGTH_LONG).show()
                }
            }
    }
}
```

MainActivity.kt

```
package com.chat.chatapp

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
```

```
 import android.view.Menu
import android.view.MenuItem
import androidx.recyclerview.widget.LinearLayoutManager
import androidx.recyclerview.widget.RecyclerView
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.database.*

class MainActivity : AppCompatActivity() {
    private lateinit var userRecyclerView: RecyclerView
    private lateinit var userList: ArrayList<User>
    private lateinit var adapter: UserAdapter
    private lateinit var mAuth: FirebaseAuth
    private lateinit var mDbRef: DatabaseReference

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        mAuth = FirebaseAuth.getInstance()
        mDbRef = FirebaseDatabase.getInstance().getReference()

        userList = ArrayList()
        adapter = UserAdapter(this, userList)
userRecyclerView = findViewById(R.id.userRecyclerView)
        userRecyclerView.layoutManager = LinearLayoutManager(this)
        userRecyclerView.adapter = adapter

        mDbRef.child("user").addValueEventListener(object : ValueEventListener {
            override fun onDataChange(snapshot: DataSnapshot) {

                userList.clear()

                for (postSnapshot in snapshot.children) {
                    val currentUser = postSnapshot.getValue(User::class.java)

                    if (mAuth.currentUser?.uid != currentUser?.uid) {
                        userList.add(currentUser!!)
                    }
                }
                adapter.notifyDataSetChanged()
            }
            override fun onCancelled(error: DatabaseError) {
            }
        })
    }
```

```
 override fun onCreateOptionsMenu(menu: Menu?): Boolean {
        menuInflater.inflate(R.menu.menu, menu)
        return super.onCreateOptionsMenu(menu)
    }
    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        if (item.itemId == R.id.logout) {
            mAuth.signOut()
            val intent = Intent(this, LoginActivity::class.java)
            finish()
            startActivity(intent)
            return true
        }
        return true
     }
}
```

Message.kt

```
package com.chat.chatapp

class Message {
    var message:String? = null
    var senderId: String? = null
constructor(){}
    constructor(message:String?,senderId:String?){
        this.message = message
        this.senderId = senderId
    }
}
```

MessageAdapter.kt

```
package com.chat.chatapp

import android.content.Context
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView
import com.google.firebase.auth.FirebaseAuth

class MessageAdapter(val context: Context, val messageList: ArrayList<Message>) :
    RecyclerView.Adapter<RecyclerView.ViewHolder>() {
    val ITEM_RECEIVE = 1
    val ITEM_SENT = 2
```

```
override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): RecyclerView.ViewHolder {
        if (viewType == 1) {
//            inflate receive
            val view: View = LayoutInflater.from(context).inflate(R.layout.receive, parent, false)
            return ReceiveViewHolder(view)
        } else {
//            inflate sent
            val view: View = LayoutInflater.from(context).inflate(R.layout.sent, parent, false)
            return SentViewHolder(view)
        }
    }
 override fun onBindViewHolder(holder: RecyclerView.ViewHolder, position: Int) {
        val currentMessage = messageList[position]

        if (holder.javaClass == SentViewHolder::class.java) {
            //stuff for sent view holder
            val viewHolder = holder as SentViewHolder
            holder.sentMessage.text = currentMessage.message
        } else {
            //stuff for receive view holder
            val viewHolder = holder as ReceiveViewHolder
            holder.receiveMessage.text = currentMessage.message
        }
    }
```

```
override fun getItemViewType(position: Int): Int {
        val currentMessage = messageList[position]

        if (FirebaseAuth.getInstance().currentUser?.uid.equals(currentMessage.senderId)) {
            return ITEM_SENT
        } else {
            return ITEM_RECEIVE
        }
    }



    override fun getItemCount(): Int {
        return messageList.size
    }

    class SentViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val sentMessage = itemView.findViewById<TextView>(R.id.txt_sent_message)
    }

    class ReceiveViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val receiveMessage = itemView.findViewById<TextView>(R.id.txt_receive_message)
    }
}
```

SignupActivity.kt

```
package com.chat.chatapp

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.database.DatabaseReference
import com.google.firebase.database.FirebaseDatabase

class SignupActivity : AppCompatActivity() {
    private lateinit var edtEmail: EditText
    private lateinit var edtPasswoord: EditText
    private lateinit var edtName: EditText
    private lateinit var btnSignUp: Button
    private lateinit var mAuth: FirebaseAuth
    private lateinit var mDbRef: DatabaseReference
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_signup)
```

```
mAuth = FirebaseAuth.getInstance()

        edtName = findViewById(R.id.edt_Name)
        edtEmail = findViewById(R.id.edt_email)
        edtPasswoord = findViewById(R.id.edt_Password)
        btnSignUp = findViewById(R.id.btn_SignUp)

        btnSignUp.setOnClickListener {
            val name = edtName.text.toString()
            val email = edtEmail.text.toString()
            val password = edtPasswoord.text.toString()
            signUp(name,email,password)
        }
    }

private fun signUp(name: String, email: String, password: String) {
        mAuth.createUserWithEmailAndPassword(email, password)
            .addOnCompleteListener(this) { task ->
                if (task.isSuccessful) {
                    addUserToDatabase(name, email, mAuth.currentUser?.uid!!)
                    // Sign in success, update UI with the signed-in user's information
                    val intent = Intent(this,MainActivity::class.java)
                    finish()
                    startActivity(intent)
```

```
 } else {
                    // If sign in fails, display a message to the user.
                    Toast.makeText(this, "error", Toast.LENGTH_LONG).show()
                }
            }
    }
    private fun addUserToDatabase(name: String, email: String, uid: String?) {
        mDbRef = FirebaseDatabase.getInstance().getReference()
        mDbRef.child("user").child(uid!!).setValue(User(name,email,uid))
    }
}
```

SplashActivity.kt

```
package com.chat.chatapp

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.os.Handler
import androidx.appcompat.app.AppCompatDelegate
```

```
class SplashActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_splash)

        AppCompatDelegate.setDefaultNightMode(AppCompatDelegate.MODE_NIGHT_NO)

        Handler().postDelayed({
            startActivity(Intent(this, LoginActivity::class.java))
            finish()
        }, 2000)
    }
}
```

User.kt

```
package com.chat.chatapp

class User {
    var name:String?=null
    var email:String?=null
    var uid:String?=null

    constructor(){}
```

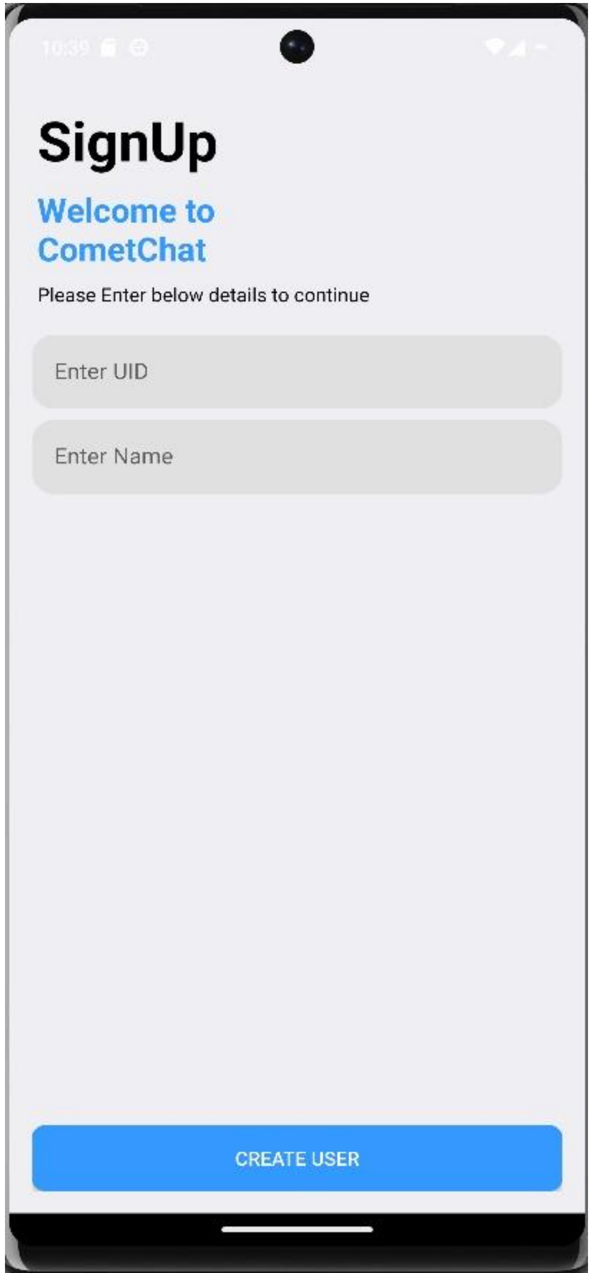
```
 constructor(name:String?,email:String?,uid:String?){
        this.name = name
        this.email = email
        this.uid = uid
    }
}
```

UserAdapter.kt

```
package com.chat.chatapp

import android.content.Context
import android.content.Intent
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.TextView
import androidx.recyclerview.widget.RecyclerView
import com.google.firebase.auth.FirebaseAuth

class UserAdapter(val context: Context, val userList: ArrayList<User>) :
    RecyclerView.Adapter<UserAdapter.UserViewHolder>() {
```

```
    val view: View = LayoutInflater.from(context).inflate(R.layout.user_layout, parent, false)
        return UserViewHolder(view)
    }

    override fun onBindViewHolder(holder: UserViewHolder, position: Int) {
        val currentUser = userList[position]
        holder.textName.text = currentUser.name
        holder.itemView.setOnClickListener {
            val intent = Intent(context, ChatActivity::class.java)
            intent.putExtra("name",currentUser.name)
            intent.putExtra("uid",currentUser.uid)
            context.startActivity(intent)
        }
    }
    override fun getItemCount(): Int {
        return userList.size
    }
 class UserViewHolder(itemView: View) : RecyclerView.ViewHolder(itemView) {
        val textName = itemView.findViewById<TextView>(R.id.txtName)
    }
}
```

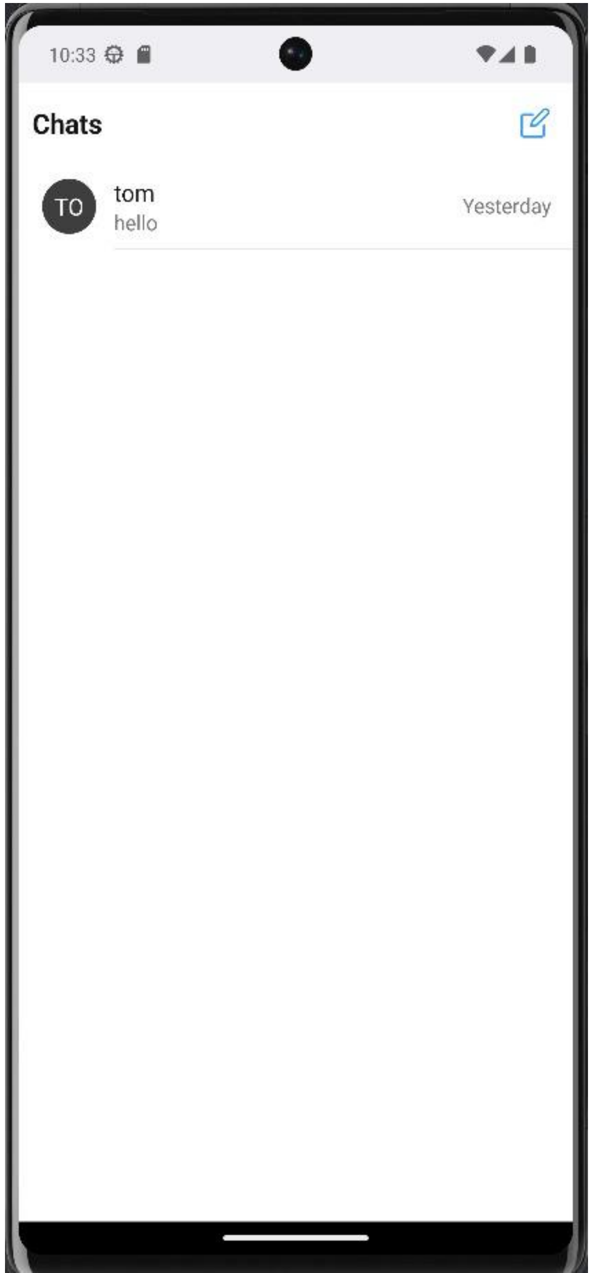
Output:



SignUp

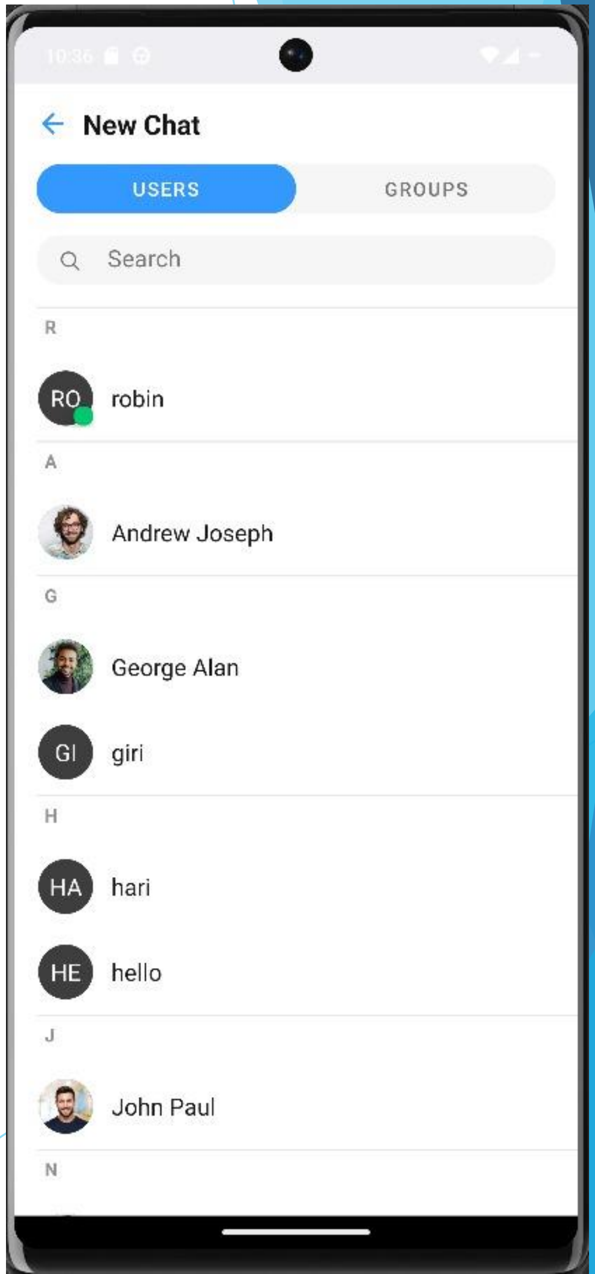
Welcome to CometChat

Please Enter below details to continue



Chats

TO	tom hello	Yesterday



New Chat

R

RO robin

A

Andrew Joseph

G

George Alan

GI giri

H

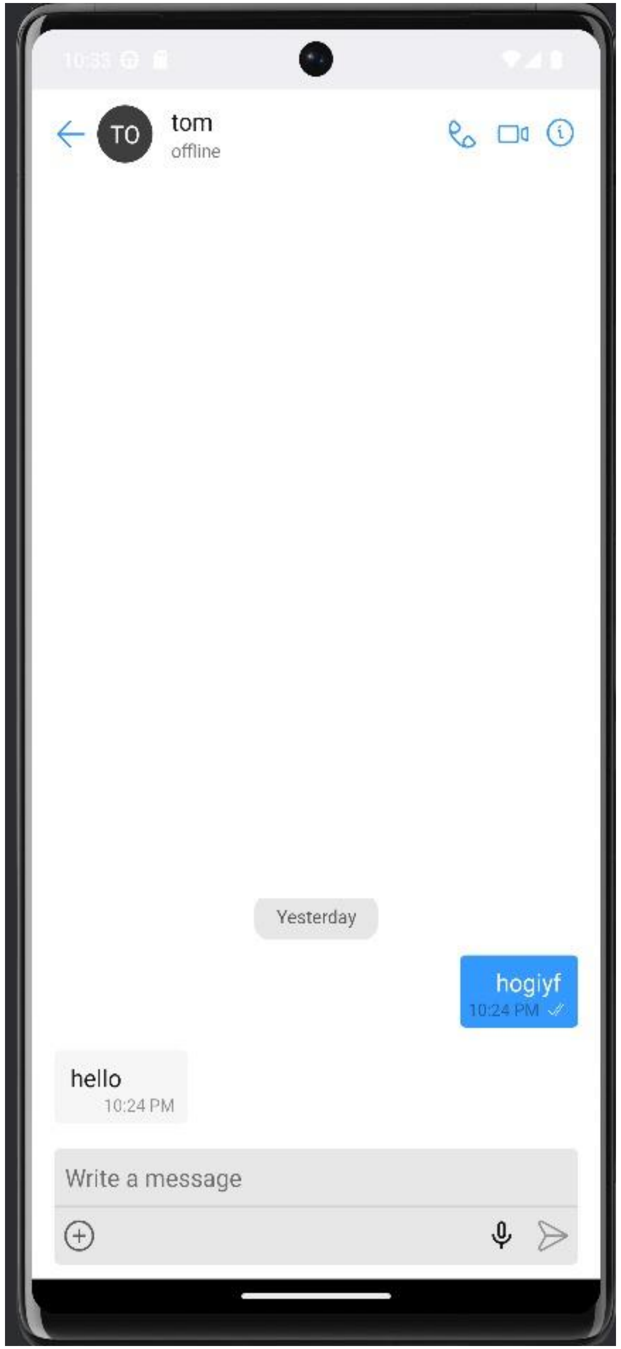
HA hari

HE hello

J

John Paul

N



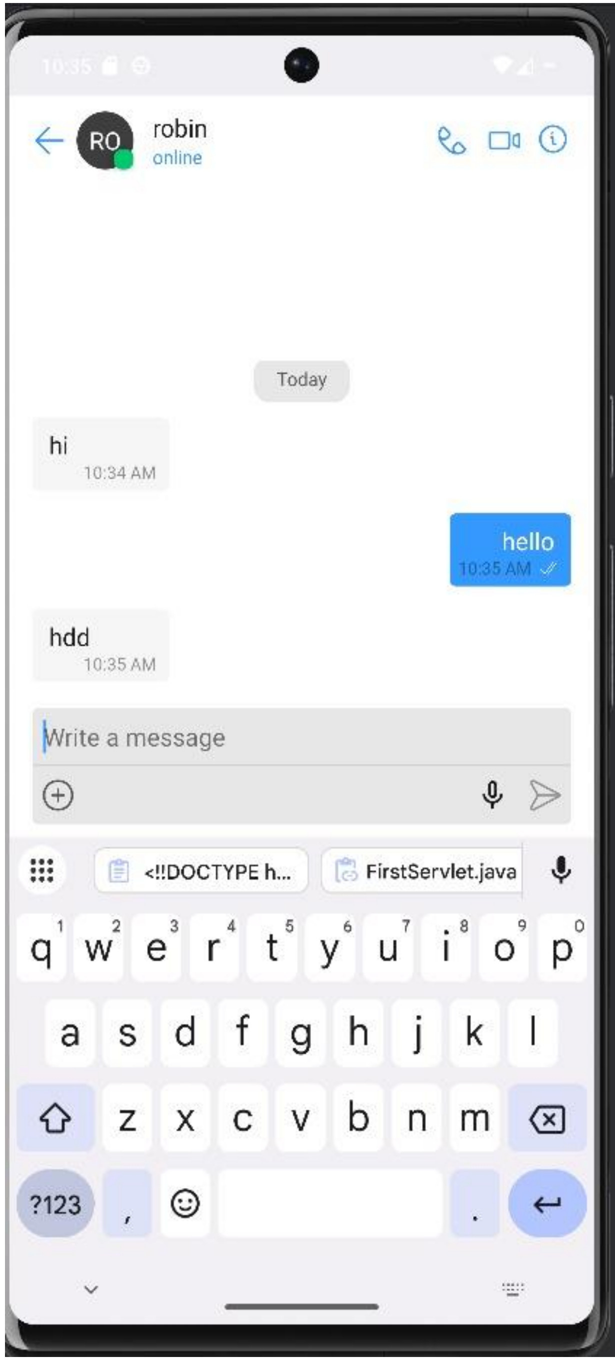
tom
offline

Yesterday

hogiyf
10:24 PM

hello
10:24 PM

Write a message



robin
online

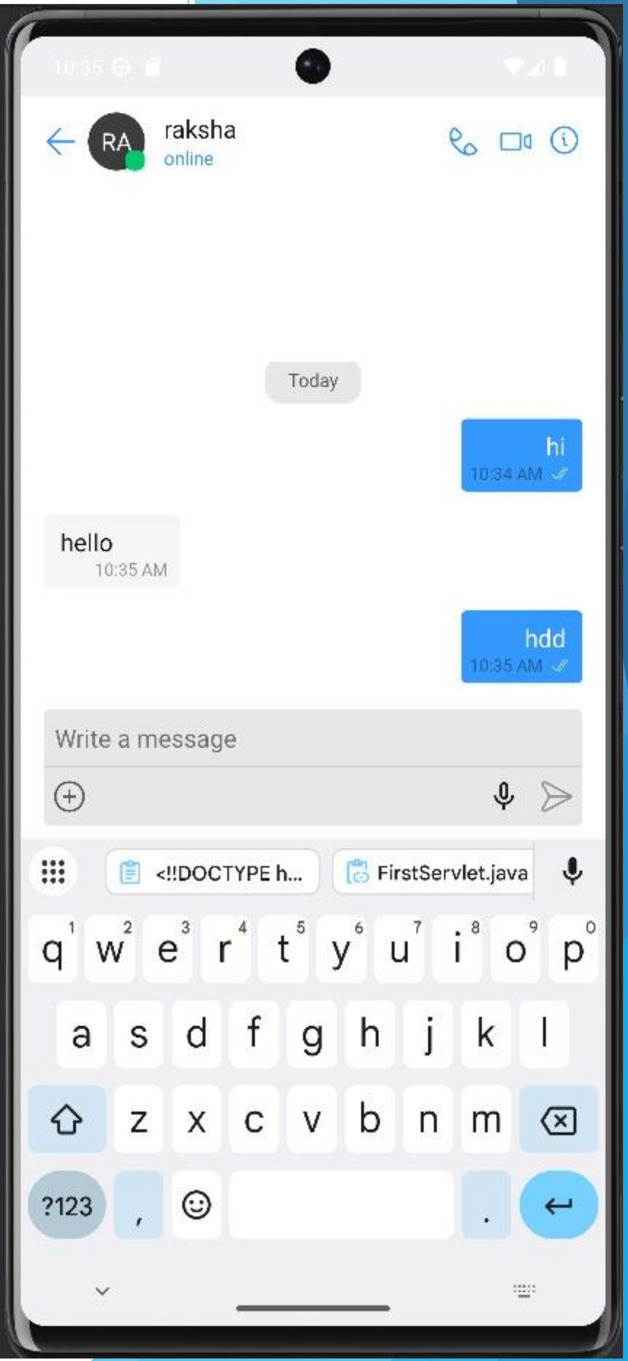
Today

hi
10:34 AM

hello
10:35 AM

hdd
10:35 AM

Write a message



raksha
online

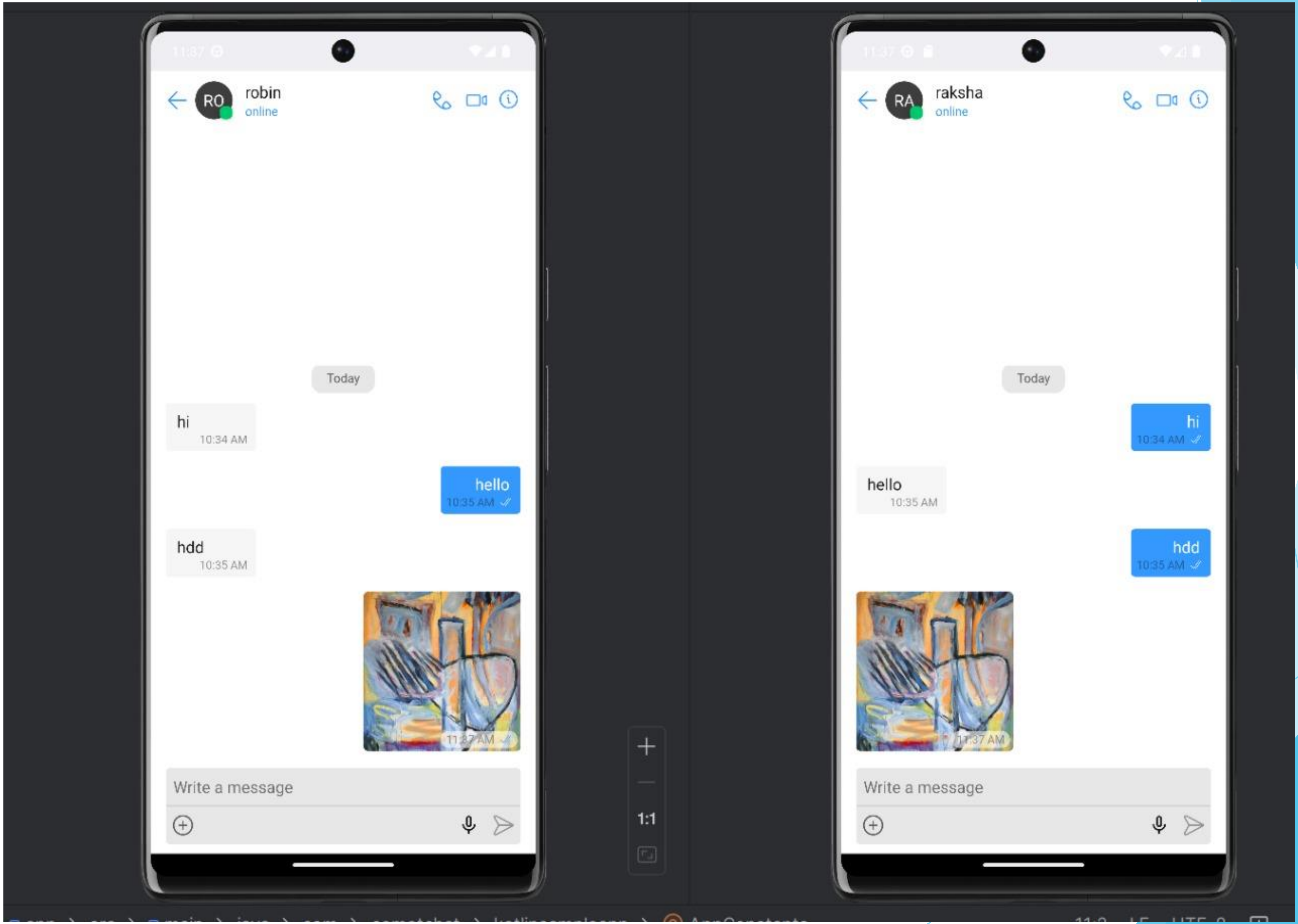
Today

hi
10:34 AM

hello
10:35 AM

hdd
10:35 AM

Write a message



hi
10:34 AM

hello
10:35 AM

hdd
10:35 AM

11:37 AM

hi
10:34 AM

hello
10:35 AM

hdd
10:35 AM

11:37 AM