Курсов проект

Програмиране на мобилни устройства

Разработване на мобилно чат приложение за Android устройства

Изготвила:

## Христиана Христова

ФКСТ, КСИ, 43 група  
Фак. № 121220041



Технически университет София

Съдържание

Въведение……………………………………………………………………………………….стр3

Анализ на съществуващи разработки................................................стр3

Проектиране.........................................................................................стр5

Софтуерна реализация........................................................................стр7

Потребителско ръководство.............................................................стр32

Заключение.........................................................................................стр35

Литература..........................................................................................стр35

**Въведение**

В съвременния свят комуникацията се превърна в неразделна част от ежедневието ни. С възхода на технологиите и интернет хората имат достъп до различни начини да поддържат връзка със своите приятели, семейство и колеги. Хората непрекъснато търсят нови и по-ефективни начини за комуникация помежду си. Ето защо приложение за чат, улеснява хората да поддържат връзка.

Те се нуждаят от просто, ефективно и лесно за потребителя приложение за чат, което същевременно да поддържа всички необходими функционалности.

Приложението позволява на потребителите да се регистрират със своя имейл или акаунт в Google и да изпращат съобщения до други потребители или да участват в групови чатове. Тази функция улеснява потребителите да поддържат връзка със своите приятели, семейство и колеги, независимо къде се намират.

То също така включва възможност за качване на профилна снимка, така че потребителите да могат да персонализират своя акаунт.

**Анализ на съществуващи разработки**

Приложението е направено за устройства, работещи с операционната система Android, което води до много конкуренти, които могат да бъдат намерени в Google Play магазина.

На пазара има много приложения за чат, но малко са тези, които комбинират простотата и функционалността на приложението. Много съществуващи приложения за чат са претрупани с ненужни функции или са твърде сложни за обикновения потребител.

1. WhatsApp

Предимства:

* Огромна потребителска база: над 2 милиарда месечни активни потребители по целия свят
* Криптиране от край до край за повишена поверителност и сигурност
* Позволява гласови и видео разговори, както и текстови съобщения
* Предлага се на множество платформи, включително Android, iOS, и настолен компютър

Недостатъци:

* Няма поддръжка за групови видео разговори с повече от 8 участници
* Ограничени възможности за персонализиране на профили и фонове за чат.
* Няма вградена функция за преводач за съобщения на различни езици

1. Telegram

Предимства:

* Силен фокус върху поверителността и сигурността, с незадължително криптиране от край до край и самоунищожаващи се съобщения
* Поддържа групови чатове с до 200 000 членове
* Вграден - в ботове за различни задачи, като актуализации на новини и прогнози за времето
* Предлага се на множество платформи, включително Android, iOS и десктоп.

Недостатъци:

* По-малка потребителска база в сравнение с WhatsApp и други популярни приложения за чат
* По-малко интуитивен потребителски интерфейс, с някои функции, “скрити” в менютата
* Ограничени възможности за персонализиране за профили и фонове за чат

1. Signal:

Предимства:

* Силен фокус върху поверителността и сигурността, с криптиране от край до край по подразбиране.
* Софтуер с отворен код за повишено сътрудничество на общността
* Поддържа гласови и видео разговори, както и текстови съобщения
* Предлага се на множество платформи, включително Android, iOS и десктоп

Недостатъци:

* По-малка потребителска база в сравнение с WhatsApp и други популярни приложения за чат.
* Няма поддръжка за групови видео разговори
* Ограничени опции за персонализиране за профили и фонове за чат.

В заключение, тези три приложения имат своите силни и слаби страни и зависи от потребителите да решат кое най-добре отговаря на техните нужди.

**Проектиране**

Потребителски профил:

Чат приложението е предназначено за потребители, които искат лесен и ефективен начин да поддържат връзка със своите приятели, семейство и колеги. Приложението позволява на потребителите да създадат акаунт, като използват имейла си или акаунта си в Google, като от тях ще се изисква да предоставят някои основни лични данни като потребителско име и по избор снимка. Потребителите също ще могат да персонализират своя профил, като добавят биография, статус и променят снимката на своя профил, когато пожелаят.

Целевата група са потребители над 13 години, като няма горна граница за възрастта.

Целевата група са потребители с българска националност, тъй като интерфейса на приложението е на български език.

Функционалност:

Реализирани са две възможности за регистрация: чрез имейл или чрез Google акаунт.

След регистрация, потребителя се пренасочва до страницата за влизане в профила.

След като потребителите влязат в приложението, те ще имат достъп до редица функционалности.

Потребителите могат да се присъединяват към групови чатове, както и да комуникират с отделни потребители, да изпращат текстови съобщения и да качват и споделят снимки и файлове.

Приложението също така включва известия (push notifications), които ще предупреждават потребителите, когато получат ново съобщение.

Потребителите ще могат да преглеждат историята на чатовете си и да изтриват съобщения.

Навигация:

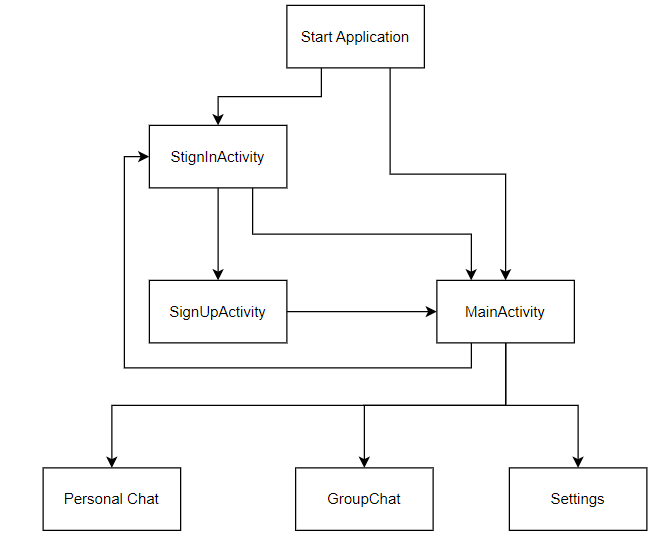
В случай, че потребителят не е влязъл в профила си, ще бъде отворен екранът за вход.

При отваряне на приложението, потребителите ще бъдат отведени до началния екран, който ще покаже тяхната история на чатове и списък с контакти. От тук потребителите могат да навигират до други страници като менюто с настройки, групови чатове и индивидуални чатове. Менюто с настройки включва опции за персонализиране на информацията в профила.

Данни:

Приложението използва предимно данни като информация за потребителски акаунт, хронология на чатове и качени файлове. За гарантиране сигурността и поверителността на потребителите, се използва аутентикацията на Firebase и услуги за база данни, за съхраняване и управление на тези данни.

Блокова схема



**Софтуерна реализация**

За създаването на приложението, е използвана комбинация от език за програмиране Java, среда за разработка на Android Studio и Firebase (authentication, database, and storage).

Също така е важно да се отбележи, че приложението е направено да поддържа устройства, работещи с Android 4.1 “Jelly Bean” и нагоре, което означава, че към днешна дата приблизително 100% от устройствата, работещи на Android операционната система, ще могат да го използват.

Java



Java е обектно ориентиран език за програмиране, разработен от Sun Microsystems като част от Java платформата.

Изходният код, написан на Java, не се компилира до машинен код за определен микропроцесор, а се компилира до междинен език – така нареченият байткод (bitecode). Байт кодът не се предава за директно изпълнение от процесора, а се изпълнява от негов аналог – виртуален процесор, наречен Java Virtual Machine (JVM). Подобна реализация има своите предимства и недостатъци.

Сред главните предимства са лесна преносимост между различните платформи (софтуерни или хардуерни), Garbage collector, контрол на правата на потребителя на ниво виртуална машина, висока степен на сигурност.

Android Studio



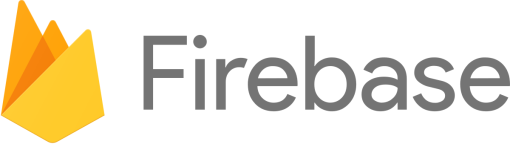
Android Studio е официалната интегрирана среда за разработка (IDE) за операционната система Android на Google, изградена върху софтуера JetBrains Intellia IDEA и проектирана специално за разработка под Android.

Характеристики предоставени в текущата версия, свързани с проекта:

* Gradle-базирана поддръжка за компилация
* Lint инструменти за улавяне на производителност, използваемост, съвместимост на версиите и други проблеми.
* Помощници, базирани на шаблони, за създаване на често срещани дизайни и компоненти на Android
* Вградена поддръжка за Google Cloud Platform, позволяваща интеграция с Firebase Cloud Messaging (по-ранни Google Cloud Messaging') и Google App Engine.
* Виртуално Android устройство (емулатор) за стартиране и отстраняване на грешки в приложения в Android studio.

Средата използва езикът за програмиране Java за осъществяване на автоматизирани дейности и XML файлове за описание на външния вид на потребителския си интерфейс.

Firebase

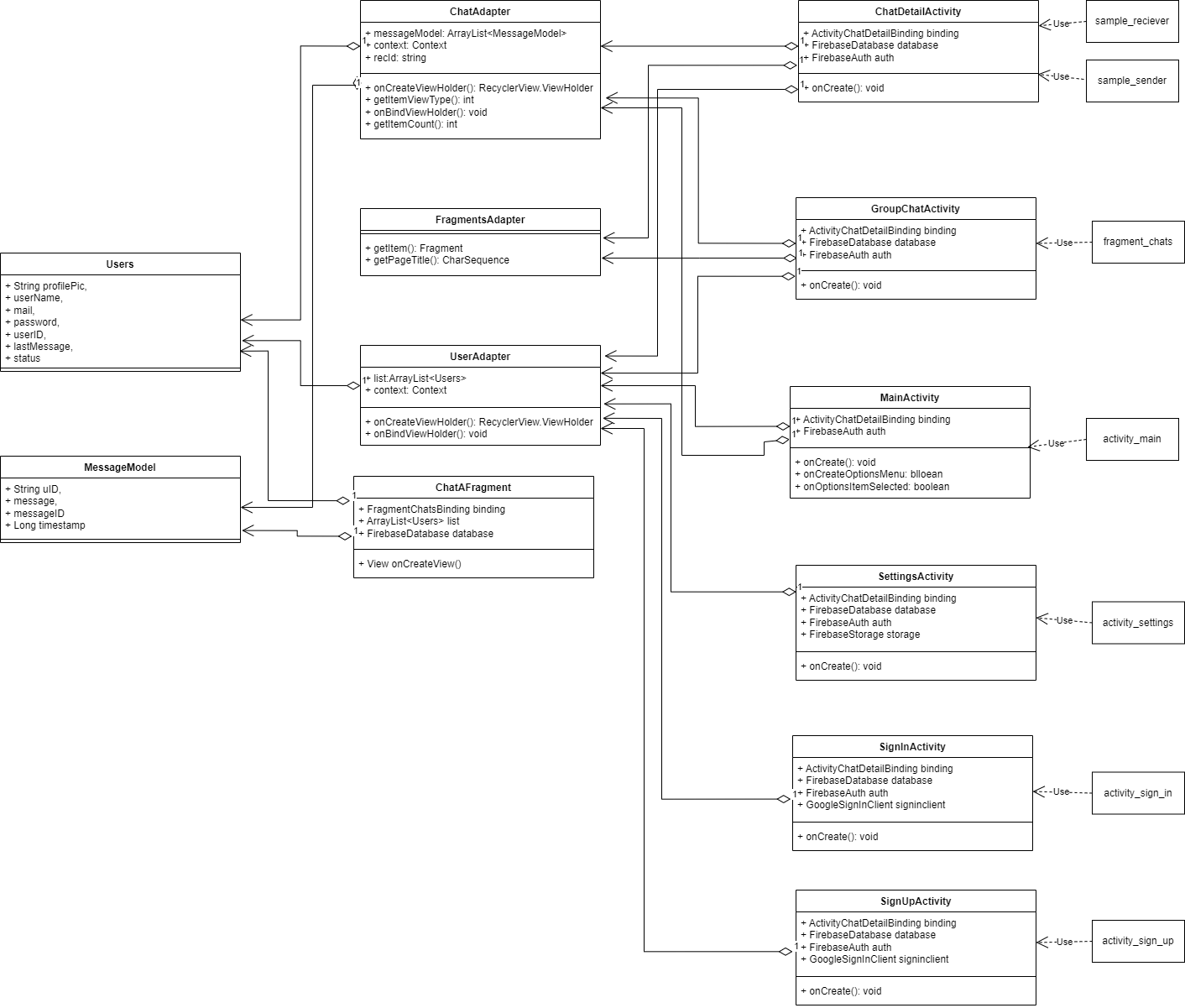


Firebase е мобилната платформа на Google, която ви помага бързо да разработвате висококачествени приложения Android studio. Firebase е набор от сървърни облачни изчислителни услуги и платформи за разработка на приложения. Тя хоства бази данни, услуги, удостоверяване и интеграция за различни приложения, включително Android, ios, Javascript, Node.js, Java, Unity, PHP и C++.

Други външни ресурси (dependencies):

1. CircularImageView: <https://github.com/hdodenhof/CircleImageView>
2. Picasso: <https://github.com/square/picasso>
3. Retrofit: <https://square.github.io/retrofit/>

Клас диаграма



Описание на класовете

1. ChatAdapter

public class ChatAdapter extends RecyclerView.Adapter {

ArrayList<MessageModel> messageModels;

Context context;

String recId;

int SENDER\_VIEW\_TYPE = 1;

int RECEIVER\_VIEW\_TYPE = 2;

@Override

public RecyclerView.ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {…..}

@Override

public int getItemViewType(int position) {…..}

@Override

public void onBindViewHolder(@NonNull RecyclerView.ViewHolder holder, int position) {…}

@Override

public int getItemCount() {…….}

public class ReceiverViewHolder extends RecyclerView.ViewHolder {

TextView receiverMsg, receiverTime;

public ReceiverViewHolder(@NonNull View itemView) {

super(itemView);

receiverMsg = itemView.findViewById(R.id.receiverText);

receiverTime = itemView.findViewById(R.id.receiverTime);

}

}

public class SenderViewHolder extends RecyclerView.ViewHolder {

TextView senderMsg, senderTime;

public SenderViewHolder(@NonNull View itemView) {

super(itemView);

senderMsg = itemView.findViewById(R.id.senderText);

senderTime = itemView.findViewById(R.id.senderTime);

}

}

}

Класът се използва за визуализиране на съобщенията под формата на лист, като определя подателя и получателя

1. FragmentsAdapter

public class FragmentsAdapter extends FragmentPagerAdapter {

public FragmentsAdapter(@NonNull FragmentManager fm) {

super(fm);

}

@NonNull

@Override

public Fragment getItem(int position) {

switch (position){

case 0: return new ChatsFragment();

case 1: return new StatusFragment();

case 2: return new CallsFragment();

default: return new ChatsFragment();

}

}

@Override

public CharSequence getPageTitle(int position) {.....}

Класът се използва за намиране на кой фрагмент да се изобрази в MainActivity (ChatsFragment, StatusFragment, CallsFragment)

1. UsersAdapter

public class UsersAdapter extends RecyclerView.Adapter<UsersAdapter.ViewHolder> {

ArrayList<Users> list;

Context context;

@NonNull

@Override

public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {…..}

@Override

public void onBindViewHolder(@NonNull ViewHolder holder, int position) {…..}

public class ViewHolder extends RecyclerView.ViewHolder {

ImageView image;

TextView userName, lastMessage;

public ViewHolder(@NonNull View itemView) {

super(itemView);

image = itemView.findViewById(R.id.profile\_image);

userName = itemView.findViewById(R.id.userNameList);

lastMessage = itemView.findViewById(R.id.lastMessage);

}

}

Класът наследява RecyclerView и се изплзва за визуализация на списъка от потребители в MainActivity. Използва и връзка с базата, с цел визуализиране на последното съобщение.

1. ChatsFragment

public class ChatsFragment extends Fragment {

public ChatsFragment() {

// Required empty public constructor

}

FragmentChatsBinding binding;

ArrayList<Users> list = new ArrayList<>();

FirebaseDatabase database;

@Override

public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {

// Inflate the layout for this fragment

binding = FragmentChatsBinding.inflate(inflater, container, false);

database = FirebaseDatabase.getInstance();

UsersAdapter adapter = new UsersAdapter(list, getContext());

binding.chatRecyclerView.setAdapter(adapter);

LinearLayoutManager layoutManager = new LinearLayoutManager(getContext());

binding.chatRecyclerView.setLayoutManager(layoutManager);

database.getReference().child("Users").addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot snapshot) {......}

Класът се използва за визуализация на списъка от потребители, а които текущия потребител е разговарял

1. Users - модел клас

public class Users {

String profilePic, userName, mail, password, userID, lastMessage, status;

// constuctors, getters, setters

}

1. MessageModel модел клас

public class MessageModel {

String uID, message, messageID;

Long timestamp;

// constuctors, getters, setters

}

1. public interface ApiInterface {

@Headers({"Authorization:" + SERVER\_KEY, "Content-Type:" + CONTENT\_TYPE})

@POST("fcm/send")

Call<PushNotification> sendNotification(@Body PushNotification pushNotification);

1. ApiUtils

public class ApiUtils {

public static Retrofit retrofit = null;

public static ApiInterface getClients(){

if(retrofit == null){

retrofit = new Retrofit.Builder()

.baseUrl(BASE\_URL).addConverterFactory(GsonConverterFactory.create())

.build();

}

return retrofit.create(ApiInterface.class);

}

}

1. MessageClass

public class MessageClass extends FirebaseMessagingService {

@Override

public void onMessageReceived(@NonNull RemoteMessage message) {

super.onMessageReceived(message);

}

}

1. public class NotificationData {

private String title, body; }

1. PushNotification

public class PushNotification {

private NotificationData notification;

private String to;

}

1. ValuesClass

public class ValuesClass {

public static final String BASE\_URL = "https://fcm.googleapis.com";

public static final String SERVER\_KEY = "key=...";

public static final String CONTENT\_TYPE = "application/json";

public static final String TO = "/topics/All";

}

Activities:

1. ChatDetailActivity

public class ChatDetailActivity extends AppCompatActivity {

ActivityChatDetailBinding binding;

FirebaseDatabase database;

FirebaseAuth auth;

// Push notification vars:

private String title, messagePush;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

binding = ActivityChatDetailBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

getSupportActionBar().hide();

database = FirebaseDatabase.getInstance();

auth = FirebaseAuth.getInstance();

final String senderId = auth.getUid();

String receiverId = getIntent().getStringExtra("userId");

String userName = getIntent().getStringExtra("userName");

String profilePic = getIntent().getStringExtra("profilePic");

binding.userName.setText(userName);

Picasso.get().load(profilePic).placeholder(R.drawable.avatar).into(binding.profileImage);

binding.backArrow.setOnClickListener(new View.OnClickListener() {…}

// fetch the messages to display them

database.getReference().child("chats")

.child(senderRoom)

.addValueEventListener(new ValueEventListener() {…….}

binding.send.setOnClickListener(new View.OnClickListener() {…….}

}

private void msg(){……}

private void sendNoti(){…….}

Класът представлява страницата на личните съобщения и се използва за визуализация, изпращане и приемане на съобщения с друг потребител. Позволява и изтриване на съобщения и изпраща нотификации при изпращане на съобщения. Свързва се с базата в реално време и модула за аутентикация на FireBase

1. GroupChatActivity

public class GroupChatActivity extends AppCompatActivity {

ActivityGroupChatBinding binding;

@Override

protected void onCreate(Bundle savedInstanceState) {

database.getReference().child("Group Chat")

.addValueEventListener(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot snapshot) {

messageModels.clear();

for(DataSnapshot dataSnapshot : snapshot.getChildren()){

MessageModel model = dataSnapshot.getValue(MessageModel.class);

messageModels.add(model);

}

adapter.notifyDataSetChanged();

}

@Override

public void onCancelled(@NonNull DatabaseError error) {

}

});

binding.send.setOnClickListener(new View.OnClickListener() {…}

Има същата функционалност като ChatDetailActivity , но позволява разговори между повече потребители

1. MainActivity

public class MainActivity extends AppCompatActivity {

ActivityMainBinding binding;

FirebaseAuth mAuth;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

binding = ActivityMainBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

mAuth = FirebaseAuth.getInstance();

binding.viewPager.setAdapter(new FragmentsAdapter(getSupportFragmentManager()));

binding.tabLayout.setupWithViewPager(binding.viewPager);

}

@Override

public boolean onCreateOptionsMenu(Menu menu){

MenuInflater inflater = getMenuInflater();

inflater.inflate(R.menu.menu, menu);

return super.onCreateOptionsMenu(menu);

}

@Override

public boolean onOptionsItemSelected(@NonNull MenuItem item) {

switch (item.getItemId()){

case R.id.settings:

//Toast.makeText(this, "Setting is clicked", Toast.LENGTH\_SHORT).show();

Intent intent2 = new Intent(MainActivity.this, SettingsActivity.class);

startActivity(intent2);

break;

case R.id.groupChat:

//Toast.makeText(this, "GroupChat is clicked", Toast.LENGTH\_SHORT).show();

Intent intent1 = new Intent(MainActivity.this, GroupChatActivity.class);

startActivity(intent1);

break;

case R.id.logout:

//Toast.makeText(this, "Logout is clicked", Toast.LENGTH\_SHORT).show();

mAuth.signOut();

Intent intent = new Intent(MainActivity.this, SignInActivity.class);

startActivity(intent);

break;

}

return super.onOptionsItemSelected(item);

}

}

Класът е главната страница на приложението. В него се инициализират менюто и фрагментите

1. SettingsActivity

public class SettingsActivity extends AppCompatActivity {

ActivitySettingsBinding binding;

FirebaseAuth auth;

FirebaseDatabase database;

FirebaseStorage storage;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

binding = ActivitySettingsBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

getSupportActionBar().hide();

storage = FirebaseStorage.getInstance();

auth = FirebaseAuth.getInstance();

database = FirebaseDatabase.getInstance();

binding.backArrow.setOnClickListener(new View.OnClickListener() {…}

binding.saveButton.setOnClickListener(new View.OnClickListener() {….}

database.getReference().child("Users").child(FirebaseAuth.getInstance().getUid())

.addListenerForSingleValueEvent(new ValueEventListener() {

@Override

public void onDataChange(@NonNull DataSnapshot snapshot) {

Users users = snapshot.getValue(Users.class);

Picasso.get().load(users.getProfilePic())

.placeholder(R.drawable.avatar)

.into(binding.profileImage);

binding.etStatus.setText(users.getStatus());

binding.txtUsername.setText(users.getUserName());

}

@Override

public void onCancelled(@NonNull DatabaseError error) {

}

});

binding.plus.setOnClickListener(new View.OnClickListener() {…}

@Override

protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {

if(data.getData() != null){

Uri sFile = data.getData();

binding.profileImage.setImageURI(sFile);

final StorageReference reference = storage.getReference().child("profile\_pic")

.child(FirebaseAuth.getInstance().getUid());

reference.putFile(sFile).addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {

@Override

public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {

reference.getDownloadUrl().addOnSuccessListener(new OnSuccessListener<Uri>() {

@Override

public void onSuccess(Uri uri) {

database.getReference().child("Users").child(FirebaseAuth.getInstance().getUid())

.child("profilePic").setValue(uri.toString());

}

Класът служи за визуализиране на настройките на потребителя. Там потребителя може да сменя името си и да добавя профилна снимка от галерията на устройството.

1. SignInActivity

public class SignInActivity extends AppCompatActivity {

ActivitySignInBinding binding;

ProgressDialog progressDialog;

FirebaseAuth mAuth;

FirebaseDatabase firebaseDatabase;

BeginSignInRequest signInRequest;

GoogleSignInClient mGoogleSignInClient;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

binding = ActivitySignInBinding.inflate(getLayoutInflater());

setContentView(binding.getRoot());

getSupportActionBar().hide();

mAuth = FirebaseAuth.getInstance();

firebaseDatabase = FirebaseDatabase.getInstance();

progressDialog = new ProgressDialog(SignInActivity.this);

progressDialog.setTitle("Влизане");

progressDialog.setMessage("Моля изчакайте\nВалидиране");

GoogleSignInOptions gso = new GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT\_SIGN\_IN)

.requestIdToken(getString(R.string.default\_web\_client\_id))

.requestEmail()

.build();

mGoogleSignInClient = GoogleSignIn.getClient(this, gso);

binding.btnSignIn.setOnClickListener(new View.OnClickListener() {….}

binding.txtClickSignUp.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(SignInActivity.this, SignUpActivity.class);

startActivity(intent);

}

});

binding.btnGoogle.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

signIn();

}

});

}

@Override

public void onActivityResult(int requestCode, int resultCode, Intent data){….}

private void firebaseAuthWithGoogle(String idToken){…..}

Класът реализира една от основните функционалности на системата - влизане в профила на потребителя. Потребителят може да влезе с имейл и парола или да използва Google акаунт.

1. SignUpActivity

public class SignUpActivity extends AppCompatActivity {

ActivitySignUpBinding binding;

private FirebaseAuth mAuth;

FirebaseDatabase database;

ProgressDialog progressDialog;

@Override

protected void onCreate(Bundle savedInstanceState) {

binding.btnSignUp.setOnClickListener(new View.OnClickListener() {…}

Класът реализира регистрация на нов клиент, като приема имейл, потребителско име и парола

XML флайлове, описващи дизайна на страници или отделни елементи

1. sample\_show\_user

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:orientation="horizontal">

<de.hdodenhof.circleimageview.CircleImageView

android:layout\_width="96dp"

android:layout\_height="96dp"

android:id="@+id/profile\_image"

android:layout\_marginLeft="5dp"

android:padding="5dp"

android:src="@drawable/avatar3"

app:civ\_border\_color="#FF000000"

app:civ\_border\_width="2dp"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:gravity="center"

android:orientation="vertical">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/userNameList"

android:paddingTop="5dp"

android:paddingBottom="5dp"

android:text="John John"

android:textSize="18sp"

android:textStyle="bold"

android:textColor="@color/black"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/lastMessage"

android:paddingTop="2dp"

android:paddingBottom="2dp"

android:text="Last Message"/>

</LinearLayout>

</LinearLayout>

1. sample\_sender

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

xmlns:app="http://schemas.android.com/apk/res-auto">

<androidx.constraintlayout.widget.ConstraintLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentRight="true"

android:layout\_marginStart="5dp"

android:layout\_marginLeft="5dp"

android:layout\_marginTop="5dp"

android:layout\_marginEnd="7dp"

android:layout\_marginRight="7dp"

android:background="@drawable/bg\_sender"

android:maxWidth="310dp"

android:maxHeight="400dp"

android:minHeight="30dp">

<androidx.constraintlayout.widget.Guideline

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/guideline2"

android:orientation="vertical"

app:layout\_constraintGuide\_end="16dp"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/senderText"

android:layout\_marginStart="8dp"

android:layout\_marginLeft="8dp"

android:layout\_marginTop="4dp"

android:layout\_marginBottom="3dp"

android:inputType="textLongMessage|textMultiLine"

android:maxWidth="270dp"

android:minWidth="50dp"

android:text="sender text"

app:layout\_constraintBottom\_toTopOf="@id/senderTime"

app:layout\_constraintEnd\_toStartOf="@id/guideline2"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="0.0"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/senderTime"

android:layout\_marginEnd="4dp"

android:layout\_marginRight="4dp"

android:layout\_marginBottom="4dp"

android:text="12:45 PM"

android:textSize="10sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toStartOf="@id/guideline2"/>

</androidx.constraintlayout.widget.ConstraintLayout>

</RelativeLayout>

1. sample\_reciever

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:layout\_marginStart="5dp"

android:layout\_marginLeft="5dp"

android:layout\_marginTop="4dp"

android:layout\_marginRight="7dp"

android:background="@drawable/bg\_reciever"

android:maxWidth="310dp"

android:maxHeight="400dp"

android:minHeight="30dp">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/receiverText"

android:layout\_margin="8dp"

android:inputType="textLongMessage|textMultiLine"

android:maxWidth="270dp"

android:minWidth="50dp"

android:text="TextView"

app:layout\_constraintBottom\_toTopOf="@id/receiverTime"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="@id/guideline2"

app:layout\_constraintTop\_toTopOf="parent"

app:layout\_constraintVertical\_bias="1.0"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/receiverTime"

android:layout\_marginEnd="4dp"

android:layout\_marginRight="4dp"

android:layout\_marginBottom="3dp"

android:text="12:56 PM"

android:textSize="10sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

/>

<androidx.constraintlayout.widget.Guideline

android:id="@+id/guideline2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="vertical"

app:layout\_constraintGuide\_begin="16dp" />

</androidx.constraintlayout.widget.ConstraintLayout>

1. fragment\_status

<?xml version="1.0" encoding="utf-8"?>

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".Fragments.StatusFragment">

<!-- TODO: Update blank fragment layout -->

<TextView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:text="Status Fragment\nСтатус"

android:gravity="center"

android:textSize="35dp"

android:textColor="@color/black"

android:textStyle="bold" />

</FrameLayout>

1. fragment\_chats

<?xml version="1.0" encoding="utf-8"?>

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".Fragments.ChatsFragment"

android:background="@drawable/w2">

<!-- TODO: Update blank fragment layout -->

<androidx.recyclerview.widget.RecyclerView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:id="@+id/chatRecyclerView"/>

</FrameLayout>

1. fragment\_call

<?xml version="1.0" encoding="utf-8"?>

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".Fragments.CallsFragment">

<!-- TODO: Update blank fragment layout -->

<TextView

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:text="Calls Fragment\nОбаждания"

android:gravity="center"

android:textSize="35dp"

android:textColor="@color/black"

android:textStyle="bold"/>

</FrameLayout>

1. activity\_sign\_up

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".SignUpActivity"

android:orientation="vertical"

android:background="@drawable/w1"

android:gravity="center"

>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="30dp"

android:layout\_marginLeft="30dp"

android:layout\_marginEnd="30dp"

android:layout\_marginRight="30dp"

android:gravity="center\_horizontal"

android:orientation="vertical">

<ImageView

android:layout\_width="100dp"

android:layout\_height="100dp"

android:id="@+id/imageView"

android:src="@drawable/ic\_whatsapp"/>

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/txtUsername"

android:layout\_marginTop="15dp"

android:layout\_marginBottom="10dp"

android:ems="10"

android:hint="Потребителско име"

android:inputType="textPersonName"

android:padding="9dp"

android:textColor="@color/lightGrey"

android:textColorHint="@color/chatbackground" />

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:id="@+id/txtEmail"

android:ems="10"

android:hint="Имейл"

android:inputType="textEmailAddress"

android:padding="9dp"

android:textColor="@color/lightGrey"

android:textColorHint="@color/chatbackground" />

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="10dp"

android:id="@+id/txtPassword"

android:ems="10"

android:hint="Парола"

android:inputType="textPassword"

android:padding="9dp"

android:textColor="@color/lightGrey"

android:textColorHint="@color/chatbackground" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/txtAlreadyHaveAccount"

android:layout\_gravity="right"

android:text="Вече имате акаунт?"

android:textColor="@color/white"

android:gravity="right"

android:padding="5dp"

android:textStyle="bold"

/>

<Button

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:id="@+id/btnSignUp"

android:layout\_marginTop="10dp"

android:padding="15dp"

android:text="Регистрация"

android:textAllCaps="false"

app:backgroundTint="@color/colorPrimary"

app:cornerRadius="8dp" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_marginTop="25dp"

android:orientation="horizontal" >

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/btnGoogle"

android:layout\_marginEnd="4dp"

android:layout\_marginRight="4dp"

android:layout\_weight="1"

android:drawableLeft="@drawable/ic\_google"

android:padding="15dp"

android:text="Google"

android:textAllCaps="false"

android:textColor="@color/black"

app:backgroundTint="@color/white"

app:cornerRadius="8dp" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/btnFacebook"

android:layout\_marginStart="4dp"

android:layout\_marginLeft="4dp"

android:layout\_weight="1"

android:drawableLeft="@drawable/ic\_facebook"

android:padding="15dp"

android:text="Facebook"

android:textAllCaps="false"

android:textColor="@color/white"

app:backgroundTint="@color/fb\_color"

app:cornerRadius="8dp" />

</LinearLayout>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/txtSignUpPhone"

android:layout\_marginTop="15dp"

android:text="Регистрация с номер"

android:textColor="@color/white"

android:textSize="19dp" />

</LinearLayout>

</LinearLayout>

1. activity\_sign\_in

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".SignInActivity"

android:orientation="vertical"

android:gravity="center"

android:background="@drawable/w1"

>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="30dp"

android:layout\_marginEnd="30dp"

android:layout\_marginLeft="30dp"

android:layout\_marginRight="30dp"

android:gravity="center\_horizontal"

android:orientation="vertical">

<ImageView

android:layout\_width="100dp"

android:layout\_height="100dp"

android:src="@drawable/ic\_whatsapp"

android:id="@+id/imageView" />

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/txtEmail"

android:layout\_marginBottom="10dp"

android:layout\_marginTop="10dp"

android:ems="10"

android:hint="Имейл"

android:inputType="textEmailAddress"

android:padding="9dp"

android:textColor="@color/white"

android:textColorHint="@color/chatbackground" />

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/txtPassword"

android:ems="10"

android:hint="Парола"

android:inputType="textPassword"

android:padding="9dp"

android:textColor="@color/white"

android:textColorHint="@color/chatbackground" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/txtClickSignUp"

android:text="Регистрация"

android:textColor="@color/white"

android:layout\_gravity="right"

android:gravity="right"

android:padding="5dp"

android:textStyle="bold"

android:textSize="16sp" />

<Button

android:layout\_width="150dp"

android:layout\_height="wrap\_content"

android:id="@+id/btnSignIn"

android:layout\_marginTop="10dp"

android:padding="15dp"

android:text="Вход"

android:textAllCaps="false"

app:backgroundTint="@color/colorPrimary"

app:cornerRadius="8dp" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_marginTop="25dp"

android:orientation="horizontal" >

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/btnGoogle"

android:layout\_marginEnd="4dp"

android:layout\_marginRight="4dp"

android:layout\_weight="1"

android:drawableLeft="@drawable/ic\_google"

android:padding="15dp"

android:text="Google"

android:textAllCaps="false"

android:textColor="@color/black"

app:backgroundTint="@color/white"

app:cornerRadius="8dp" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/btnFacebook"

android:layout\_marginStart="4dp"

android:layout\_marginLeft="4dp"

android:layout\_weight="1"

android:drawableLeft="@drawable/ic\_facebook"

android:padding="15dp"

android:text="Facebook"

android:textAllCaps="false"

android:textColor="@color/white"

app:backgroundTint="@color/fb\_color"

app:cornerRadius="8dp" />

</LinearLayout>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/txtSignUpPhone"

android:layout\_marginTop="15dp"

android:text="Регистрация с номер"

android:textColor="@color/white"

android:textSize="20dp" />

</LinearLayout>

</LinearLayout>

1. activity\_settings

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".SettingsActivity"

android:background="@drawable/w1"

>

<ImageView

android:layout\_width="30dp"

android:layout\_height="30dp"

android:id="@+id/backArrow"

android:layout\_marginStart="10dp"

android:layout\_marginLeft="10dp"

android:layout\_marginTop="10dp"

app:srcCompat="@drawable/leftarrowwhite"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/linear"

android:layout\_below="@id/backArrow"

android:layout\_marginStart="20dp"

android:layout\_marginLeft="20dp"

android:layout\_marginRight="20dp"

android:layout\_marginEnd="20dp"

android:gravity="center"

android:orientation="vertical">

<de.hdodenhof.circleimageview.CircleImageView

android:layout\_width="120dp"

android:layout\_height="120dp"

android:id="@+id/profile\_image"

android:layout\_marginStart="10dp"

android:layout\_marginLeft="10dp"

android:layout\_marginTop="10dp"

android:src="@drawable/avatar"

app:civ\_border\_color="#FF000000"

app:civ\_border\_width="2dp"/>

<ImageView

android:layout\_width="30dp"

android:layout\_height="30dp"

android:id="@+id/plus"

android:layout\_marginStart="30dp"

android:layout\_marginLeft="30dp"

android:layout\_marginTop="-25dp"

app:srcCompat="@drawable/plus"/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_marginTop="10dp"

android:orientation="horizontal">

<TextView

android:layout\_width="100dp"

android:layout\_height="wrap\_content"

android:padding="5dp"

android:text="Име:"

android:textColor="@color/white"

android:textSize="16sp"

android:textStyle="bold"/>

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/txtUsername"

android:hint="Въведете потребителско име"

android:inputType="textPersonName"

android:textColor="@color/white"

android:textColorHint="#BFBFBF"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_marginTop="10dp"

android:orientation="horizontal">

<TextView

android:layout\_width="100dp"

android:layout\_height="wrap\_content"

android:padding="5dp"

android:text="Описание:"

android:textColor="@color/white"

android:textSize="16sp"

android:textStyle="bold"/>

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/etStatus"

android:hint="Въведете описание"

android:inputType="textPersonName"

android:textColor="@color/white"

android:textColorHint="#BFBFBF"/>

</LinearLayout>

<androidx.appcompat.widget.AppCompatButton

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="right"

android:layout\_marginTop="10dp"

android:layout\_marginBottom="5dp"

android:background="@drawable/btn\_bg"

android:padding="5dp"

android:text="Запази"

android:textAllCaps="false"

android:textColor="@color/colorPrimary"

android:id="@+id/saveButton"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:background="@drawable/linear\_bg"

android:layout\_marginTop="10dp"

android:orientation="vertical"

android:padding="40dp"

>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/textView2"

android:layout\_marginStart="50dp"

android:layout\_marginLeft="50dp"

android:drawableLeft="@drawable/privacy"

android:drawablePadding="21dp"

android:padding="10dp"

android:text="Поверителност"

android:textColorHint="@color/black"

android:textSize="18sp"

android:textStyle="bold"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/textView3"

android:layout\_marginStart="50dp"

android:layout\_marginLeft="50dp"

android:drawableLeft="@drawable/about"

android:drawablePadding="26.5dp"

android:padding="10dp"

android:text="Относно"

android:textColorHint="@color/black"

android:textSize="18sp"

android:textStyle="bold"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/textView4"

android:layout\_marginStart="50dp"

android:layout\_marginLeft="50dp"

android:drawableLeft="@drawable/invite"

android:drawablePadding="25dp"

android:padding="10dp"

android:text="Покани Приятел"

android:textColorHint="@color/black"

android:textSize="18sp"

android:textStyle="bold"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/textView5"

android:layout\_marginStart="50dp"

android:layout\_marginLeft="50dp"

android:drawableLeft="@drawable/notification"

android:drawablePadding="20dp"

android:padding="10dp"

android:text="Нотификации"

android:textColorHint="@color/black"

android:textSize="18sp"

android:textStyle="bold"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:id="@+id/textView6"

android:layout\_marginStart="50dp"

android:layout\_marginLeft="50dp"

android:drawableLeft="@drawable/help"

android:drawablePadding="26.5dp"

android:padding="10dp"

android:text="Помощ"

android:textColorHint="@color/black"

android:textSize="18sp"

android:textStyle="bold"/>

</LinearLayout>

</RelativeLayout>

1. activity\_main

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity"

android:orientation="vertical">

<com.google.android.material.tabs.TabLayout

android:layout\_width="match\_parent"

android:layout\_height="58dp"

android:background="@color/colorPrimary"

android:elevation="12dp"

app:tabIndicator="@color/white"

app:tabSelectedTextColor="@color/white"

app:tabMaxWidth="0dp"

app:tabGravity="fill"

app:tabMode="fixed"

app:tabTextColor="@color/white"

android:id="@+id/tabLayout"/>

<androidx.viewpager.widget.ViewPager

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

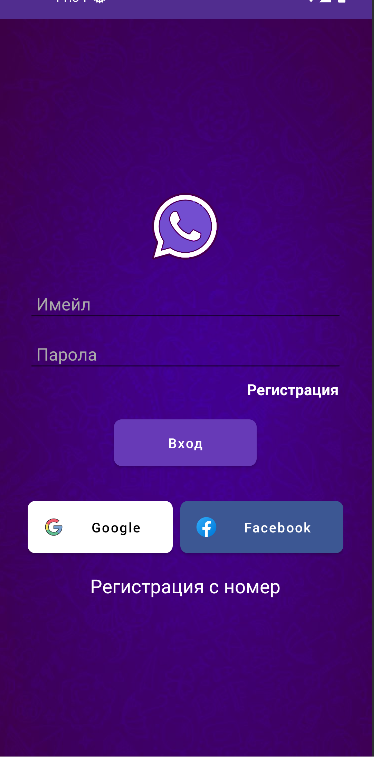
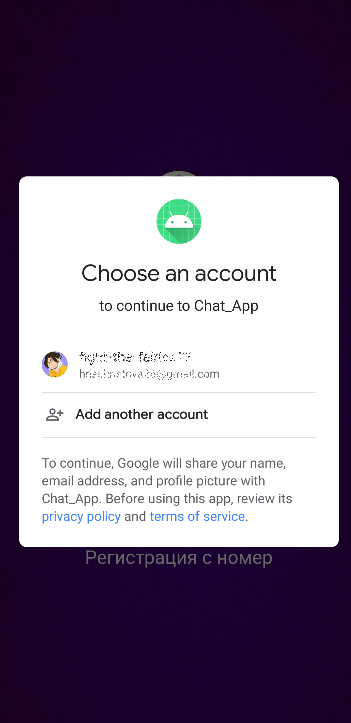
android:id="@+id/viewPager"/>

</LinearLayout>

**Потребителско ръководство**

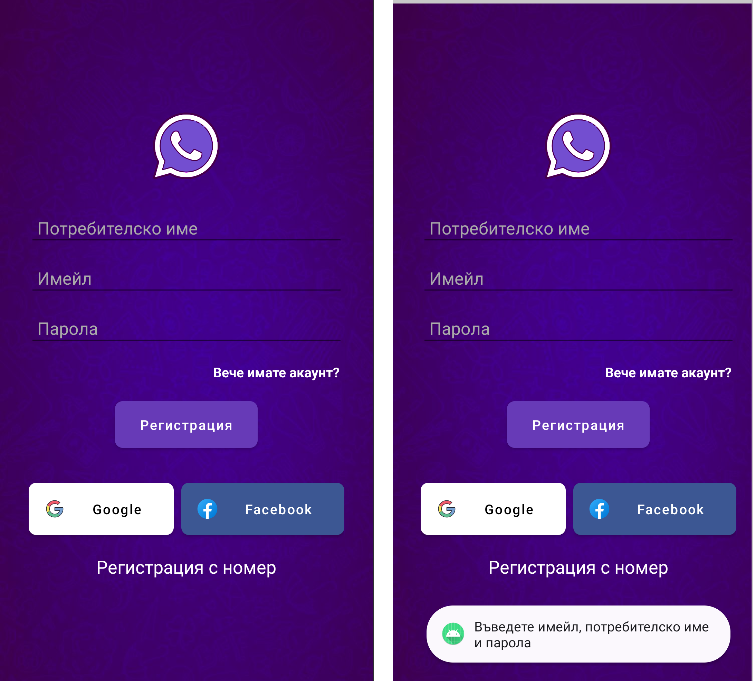
1. При първо влизане на потребител, се отваря страницата за влизане, където потребителят трябва да въведе своя имейл и парола. Има и избор да влезе е Google акаунт.

SinInActivity:

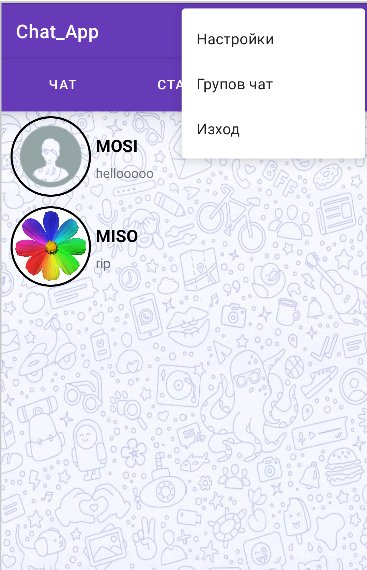
 

1. Ако потребителя няма акаунт, то той може да си създаде като кликне на бутона “Регистрация”. Като при некоректно зададени или липсващи полета, се подава съобщение за грешка

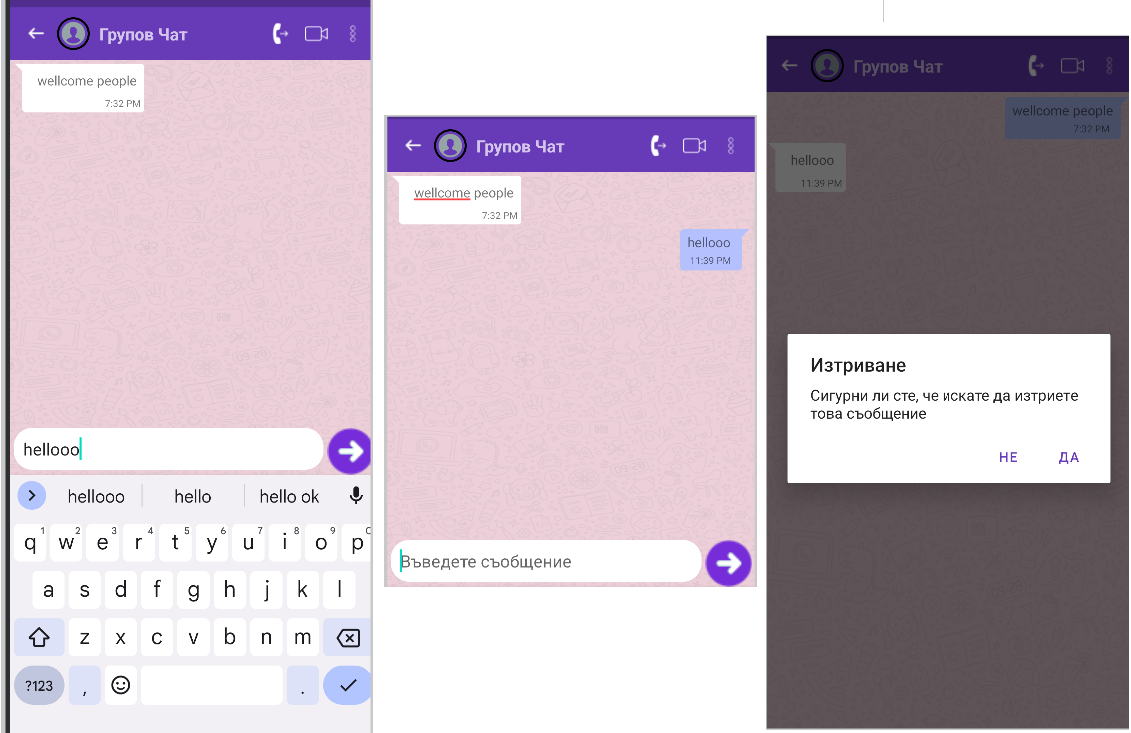
SignUpActivity:



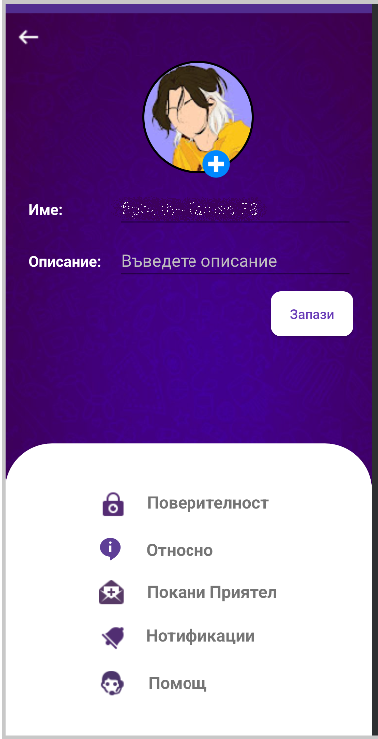
1. При успешно влизане, приложението прехвърля потребителя към главната страница (MainActivity), където потребителя може да види свойве чатове и меню с групов чат, настройките и опцията за излизане от профила



1. След отваряне на всяки индивидуален чат, потребителя може д изпраща и изтрива съобщения. Аналогично е и при груповия чат.



1. От менюто могат да се изберат настойките и това ще отведе потребителя към съответната страница, където потребителя може да сменя потребителското си име, описанието и снимката си



**Заключение**

На пазара има много приложения за чат, но малко са тези, които комбинират простотата и функционалността на приложението. Много съществуващи приложения за чат са претрупани с ненужни функции или са твърде сложни за обикновения потребител.

Текущата реализация на приложението има имплементирани основни функционалности за едно чат приложение. В реализацията, обаче, не са имплементирани всички функционалности, необходими за цялостно чат приложение. Тези функционалности предстои да бъдат реализирани в бъдеще.

Макар че има неизпълнени функционалности, приложението е с интуитивен и лесен за използване интерфейс, което го прави потенциален конкурент на много приложения, пуснати на пазара.

**Литература**

1. Android доkументация

<https://developer.android.com/docs>

1. FireBase доkументация

<https://firebase.google.com/docs/cloud-messaging/android/client#java>

<https://firebase.google.com/docs/auth/android/google-signin>

<https://firebase.google.com/docs/cloud-messaging/android/client>

1. Retrofir доkументация

<https://square.github.io/retrofit/>

<https://www.digitalocean.com/community/tutorials/retrofit-android-example-tutorial>

1. Google Cloud доkументация

<https://console.cloud.google.com/apis/api/googlecloudmessaging.googleapis.com/quotas?project=chat-app-5d081>

<https://console.cloud.google.com/getting-started>