activity\_login.xml

<?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"

android:padding="5pt"

android:orientation="vertical"

tools:context=".LoginActivity">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:textSize="14pt"

android:text="Login"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:textSize="10pt"

android:text="E-mail:"/>

<EditText

android:id="@+id/edtEmail"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:textSize="10pt"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:textSize="10pt"

android:text="Password:"/>

<EditText

android:id="@+id/edtPassword"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:textSize="10pt"/>

<Button

android:id="@+id/btnLogin"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:textSize="10pt"

android:text="Login"/>

</LinearLayout>

activity\_home.xml

<?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"

android:padding="5pt"

android:orientation="vertical"

tools:context=".HomeActivity">

<TextView

android:id="@+id/tvLogged"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="10pt"

android:text="Welcome back, "/>

<ScrollView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/rvInbox"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"/>

</LinearLayout>

</ScrollView>

<Button

android:id="@+id/btnCompose"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10pt"

android:textSize="10pt"

android:text="Compose New Email"/>

</LinearLayout>

activity\_home.xml

<?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"

android:padding="5pt"

android:orientation="vertical"

tools:context=".HomeActivity">

<TextView

android:id="@+id/tvLogged"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="10pt"

android:text="Welcome back, "/>

<ScrollView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<androidx.recyclerview.widget.RecyclerView

android:id="@+id/rvInbox"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"/>

</LinearLayout>

</ScrollView>

<Button

android:id="@+id/btnCompose"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10pt"

android:textSize="10pt"

android:text="Compose New Email"/>

</LinearLayout>

activity\_detail.xml

<?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"

android:padding="5pt"

android:orientation="vertical"

tools:context=".DetailActivity">

<TextView

android:id="@+id/tvSubject"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textStyle="bold"

android:textSize="14pt"/>

<TextView

android:id="@+id/tvSender"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:textSize="10pt"

android:text="From: "/>

<TextView

android:id="@+id/tvDateTime"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="10pt"/>

<TextView

android:id="@+id/tvMessage"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="10pt"/>

<Button

android:id="@+id/btnReply"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="15pt"

android:textSize="10pt"

android:text="Reply"/>

<Button

android:id="@+id/btnBack"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:textSize="10pt"

android:text="Back"/>

</LinearLayout>

activity\_compose.xml

<?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"

android:padding="5pt"

android:orientation="vertical"

tools:context=".ComposeActivity">

<TextView

android:id="@+id/tvSender"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="10pt"

android:text=" From: "/>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:orientation="horizontal">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="10pt"

android:text=" To: "/>

<EditText

android:id="@+id/edtRecipient"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"/>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="5pt"

android:orientation="horizontal">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="10pt"

android:text=" Subject: "/>

<EditText

android:id="@+id/edtSubject"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"/>

</LinearLayout>

<EditText

android:id="@+id/edtMessage"

android:inputType="textMultiLine"

android:layout\_width="match\_parent"

android:layout\_height="80pt"

android:layout\_marginTop="5pt"

android:layout\_marginLeft="42pt"

android:textSize="10pt"

android:hint="Compose E-mail"/>

<Button

android:id="@+id/btnSend"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10pt"

android:textSize="10pt"

android:text="Send E-mail"/>

<Button

android:id="@+id/btnCancel"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="10pt"

android:text="Cancel"/>

</LinearLayout>

inbox\_view.xml

<?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"

android:orientation="vertical">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="5pt"

android:orientation="horizontal">

<TextView

android:id="@+id/tvIcon"

android:layout\_width="30pt"

android:layout\_height="30pt"

android:textAlignment="center"

android:textColor="#FFFF"

android:textSize="20pt"/>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="2"

android:layout\_marginLeft="5pt"

android:orientation="vertical">

<TextView

android:id="@+id/tvSender"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="7pt"/>

<TextView

android:id="@+id/tvSubject"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="7pt"/>

<TextView

android:id="@+id/tvMessage"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="7pt"/>

</LinearLayout>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:orientation="vertical">

<TextView

android:id="@+id/tvDateTime"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center"

android:textSize="7pt"/>

<ImageView

android:id="@+id/ivStar"

android:layout\_width="20pt"

android:layout\_height="15pt"

android:layout\_gravity="center"

android:layout\_marginTop="3.5pt"/>

</LinearLayout>

</LinearLayout>

</LinearLayout>

options\_menu.xml

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

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

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

<item

android:id="@+id/menuLogout"

android:title="Logout"/>

<item

android:id="@+id/menuStarred"

android:title="Starred"/>

<item

android:id="@+id/menuAll"

android:title="All E-mail"/>

<item

android:id="@+id/menuTrash"

android:title="Trash"/>

<item

android:id="@+id/menuDeleteAll"

android:title="Delete All E-mail"/>

</menu>

context\_menu.xml

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

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

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

<item

android:id="@+id/menuDelete"

android:title="Delete E-mail"/>

</menu>

LoginActivity.java

public class LoginActivity extends AppCompatActivity {

EditText edtEmail, edtPassword;

Button btnLogin;

Intent page;

ArrayList<User> users;

int idxUser;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_login);

edtEmail= findViewById(R.id.edtEmail);

edtPassword= findViewById(R.id.edtPassword);

btnLogin= findViewById(R.id.btnLogin);

users= new ArrayList<>();

getExtrasData();

btnLogin.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String email= edtEmail.getText().toString();

String password= edtPassword.getText().toString();

if (!email.isEmpty() && !password.isEmpty()) {

if (isEmail(email) && isPassword(email, password)) {

if (!users.isEmpty()) {

boolean same= false;

for (int i= 0; i<users.size(); i++) {

if (users.get(i).getEmail().equals(email)) {

same= true;

idxUser= i;

break;

}

}

if (!same) {

users.add(new User(email));

idxUser= users.size()-1;

}

} else {

users.add(new User(email));

idxUser= users.size()-1;

}

page= new Intent(LoginActivity.this, HomeActivity.class);

page.putExtra("users", users);

page.putExtra("idxUser", idxUser);

startActivity(page);

} else {

Toast.makeText(LoginActivity.this, "Wrong e-mail or password!", Toast.LENGTH\_SHORT).show();

}

} else {

Toast.makeText(LoginActivity.this, "Field must not empty!", Toast.LENGTH\_SHORT).show();

}

}

});

}

private boolean isEmail(String email) {

String depanAt= "", belakangAt= "", belakangDot= "";

if (email.contains("@")) {

depanAt= email.substring(0, email.indexOf("@"));

belakangAt= email.substring(email.indexOf("@")+1);

if (!depanAt.isEmpty() && !belakangAt.isEmpty()) {

if (belakangAt.contains(".")) {

belakangDot= belakangAt.substring(belakangAt.indexOf(".")+1);

}

if (!belakangDot.isEmpty()) {

return true;

}

}

}

return false;

}

private boolean isPassword(String email, String password) {

String check= email.substring(0, 3)+email.length();

if (check.equals(password)) return true;

return false;

}

private void getExtrasData() {

page= getIntent();

if (page.hasExtra("users")) {

if (page.getSerializableExtra("users") != null) {

ArrayList<User> extra= (ArrayList<User>) page.getSerializableExtra("users");

users.addAll(extra);

}

} else {

Toast.makeText(this, "No data", Toast.LENGTH\_SHORT).show();

}

}

}

HomeActivity.java

public class HomeActivity extends AppCompatActivity {

TextView tvLogged;

RecyclerView rvInbox;

Button btnCompose;

InboxAdapter adapter;

Intent page;

ArrayList<User> users;

ArrayList<Email> displayInbox;

int idxUser;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_home);

tvLogged= findViewById(R.id.tvLogged);

rvInbox= findViewById(R.id.rvInbox);

btnCompose= findViewById(R.id.btnCompose);

users= new ArrayList<>();

displayInbox= new ArrayList<>();

getExtrasData();

displayInbox.addAll(users.get(idxUser).getAllInbox());

rvInbox.setHasFixedSize(true);

rvInbox.setLayoutManager(new LinearLayoutManager(this));

adapter= new InboxAdapter(displayInbox);

rvInbox.setAdapter(adapter);

adapter.setOnItemClickCallback(new InboxAdapter.OnItemClickCallback() {

@Override

public void onItemClicked(int position) {

if (!users.get(idxUser).getAllInbox().isEmpty() && !displayInbox.get(position).isDeleted()) {

page= new Intent(HomeActivity.this, DetailActivity.class);

page.putExtra("users", users);

page.putExtra("idxUser", idxUser);

page.putExtra("idxMail", position);

startActivity(page);

}

}

});

adapter.setOnItemLongClickCallback(new InboxAdapter.OnItemLongClickCallback() {

@Override

public void onItemLongClicked(View view, int position) {

if (!users.get(idxUser).getAllInbox().isEmpty() && !displayInbox.get(position).isDeleted()) {

onCreatePopUpMenu(view, position);

}

}

});

adapter.setOnImageViewClickCallback(new InboxAdapter.OnImageViewClickCallback() {

@Override

public void onStarClicked(int position) {

if (!users.get(idxUser).getAllInbox().isEmpty() && !displayInbox.get(position).isDeleted()) {

if (users.get(idxUser).getAllInbox().get(position).isStarred()) {

users.get(idxUser).getAllInbox().get(position).setStarred(false);

} else {

users.get(idxUser).getAllInbox().get(position).setStarred(true);

}

adapter.notifyDataSetChanged();

}

}

});

btnCompose.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

page= new Intent(HomeActivity.this, ComposeActivity.class);

page.putExtra("users", users);

page.putExtra("idxUser", idxUser);

startActivity(page);

}

});

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.options\_menu, menu);

return super.onCreateOptionsMenu(menu);

}

@Override

public boolean onOptionsItemSelected(@NonNull MenuItem item) {

switch (item.getItemId()) {

case R.id.menuLogout:

page= new Intent(this, LoginActivity.class);

page.putExtra("users", users);

startActivity(page);

break;

case R.id.menuStarred:

users.get(idxUser).getStarredInbox().clear();

for (Email mail : users.get(idxUser).getAllInbox()) {

if (mail.isStarred()) {

users.get(idxUser).setStarredInbox(mail);

}

}

displayInbox.clear();

displayInbox.addAll(users.get(idxUser).getStarredInbox());

adapter.notifyDataSetChanged();

break;

case R.id.menuAll:

displayInbox.clear();

displayInbox.addAll(users.get(idxUser).getAllInbox());

adapter.notifyDataSetChanged();

break;

case R.id.menuTrash:

displayInbox.clear();

displayInbox.addAll(users.get(idxUser).getTrash());

adapter.notifyDataSetChanged();

break;

case R.id.menuDeleteAll:

if (!users.get(idxUser).getAllInbox().isEmpty()) {

for (Email mail : users.get(idxUser).getAllInbox()) {

mail.setDeleted(true);

users.get(idxUser).setTrash(mail);

}

users.get(idxUser).getAllInbox().clear();

displayInbox.clear();

displayInbox.addAll(users.get(idxUser).getAllInbox());

adapter.notifyDataSetChanged();

}

break;

};

return super.onOptionsItemSelected(item);

}

private void onCreatePopUpMenu(View view, final int position) {

PopupMenu menu= new PopupMenu(this, view);

menu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {

@Override

public boolean onMenuItemClick(MenuItem menuItem) {

if (menuItem.getItemId() == R.id.menuDelete) {

users.get(idxUser).getAllInbox().get(position).setDeleted(true);

users.get(idxUser).setTrash(users.get(idxUser).getAllInbox().get(position));

users.get(idxUser).getAllInbox().remove(position);

displayInbox.clear();

displayInbox.addAll(users.get(idxUser).getAllInbox());

adapter.notifyDataSetChanged();

}

return false;

}

});

menu.getMenuInflater().inflate(R.menu.context\_menu, menu.getMenu());

menu.show();

}

private void getExtrasData() {

page= getIntent();

if (page.hasExtra("users") || page.hasExtra("idxUser")) {

if (page.getSerializableExtra("users") != null) {

users.addAll((ArrayList<User>) page.getSerializableExtra("users"));

}

if (page.getIntExtra("idxUser", 0) >= 0) {

idxUser= page.getIntExtra("idxUser", 0);

tvLogged.setText(tvLogged.getText()+users.get(idxUser).getEmail()+"!");

}

} else {

Toast.makeText(this, "No data", Toast.LENGTH\_SHORT).show();

}

}

}

DetailActivity.java

public class DetailActivity extends AppCompatActivity {

TextView tvSubject, tvSender, tvDateTime, tvMessage;

Button btnReply, btnBack;

Intent page;

ArrayList<User> users;

int idxUser, idxMail;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_detail);

tvSubject= findViewById(R.id.tvSubject);

tvSender= findViewById(R.id.tvSender);

tvDateTime= findViewById(R.id.tvDateTime);

tvMessage= findViewById(R.id.tvMessage);

btnReply= findViewById(R.id.btnReply);

btnBack= findViewById(R.id.btnBack);

users= new ArrayList<>();

getExtrasData();

btnReply.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

page= new Intent(DetailActivity.this, ComposeActivity.class);

page.putExtra("users", users);

page.putExtra("idxUser", idxUser);

page.putExtra("idxMail", idxMail);

page.putExtra("reply", true);

startActivity(page);

}

});

btnBack.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

finish();

}

});

}

private void getExtrasData() {

page= getIntent();

if (page.hasExtra("users") || page.hasExtra("idxUser") || page.hasExtra("idxMail")) {

if (page.getSerializableExtra("users") != null) {

users.addAll((ArrayList<User>) page.getSerializableExtra("users"));

}

if (page.getIntExtra("idxUser", 0) >= 0) {

idxUser= page.getIntExtra("idxUser", 0);

}

if (page.getIntExtra("idxMail", 0) >= 0) {

idxMail= page.getIntExtra("idxMail",0);

if (!users.get(idxUser).getAllInbox().isEmpty()) {

tvSubject.setText(users.get(idxUser).getAllInbox().get(idxMail).getSubject());

tvSender.setText(tvSender.getText()+users.get(idxUser).getAllInbox().get(idxMail).getSender().getEmail());

tvMessage.setText(users.get(idxUser).getAllInbox().get(idxMail).getMessage());

tvDateTime.setText(users.get(idxUser).getAllInbox().get(idxMail).getDateTime());

}

}

} else {

Toast.makeText(this, "No data", Toast.LENGTH\_SHORT).show();

}

}

}

ComposActivity.java

public class ComposeActivity extends AppCompatActivity {

TextView tvSender;

EditText edtRecipient, edtSubject, edtMessage;

Button btnSend, btnCancel;

Intent page;

ArrayList<User> users;

int idxUser, idxMail;

boolean reply;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_compose);

tvSender= findViewById(R.id.tvSender);

edtRecipient= findViewById(R.id.edtRecipient);

edtSubject= findViewById(R.id.edtSubject);

edtMessage= findViewById(R.id.edtMessage);

btnSend= findViewById(R.id.btnSend);

btnCancel= findViewById(R.id.btnCancel);

users= new ArrayList<>();

getExtrasData();

btnSend.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String recipient= edtRecipient.getText().toString();

String subject= edtSubject.getText().toString();

String message= edtMessage.getText().toString();

if (!recipient.isEmpty() && !subject.isEmpty() && !message.isEmpty()) {

if (recipient.contains("-")) {

String[] splits= recipient.split("-");

for (String split : splits) {

sendEmail(split, subject, message);

}

} else {

sendEmail(recipient, subject, message);

}

page= new Intent(ComposeActivity.this, HomeActivity.class);

page.putExtra("users", users);

page.putExtra("idxUser", idxUser);

startActivity(page);

} else {

Toast.makeText(ComposeActivity.this, "Field must not empty!", Toast.LENGTH\_SHORT).show();

}

}

});

btnCancel.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

finish();

}

});

}

private void getExtrasData() {

page= getIntent();

if (page.hasExtra("users") || page.hasExtra("idxUser") || page.hasExtra("idxMail") || page.hasExtra("reply")) {

if (page.getSerializableExtra("users") != null) {

users.addAll((ArrayList<User>) page.getSerializableExtra("users"));

}

if (page.getIntExtra("idxUser", 0) >= 0) {

idxUser= page.getIntExtra("idxUser", 0);

tvSender.setText(tvSender.getText()+users.get(idxUser).getEmail());

}

if (page.getIntExtra("idxMail", 0) >= 0) {

idxMail= page.getIntExtra("idxMail",0);

}

if (page.getBooleanExtra("reply", false)) {

reply= page.getBooleanExtra("reply", false);

if (reply) {

edtRecipient.setText(users.get(idxUser).getAllInbox().get(idxMail).getSender().getEmail());

edtSubject.setText("RE: "+users.get(idxUser).getAllInbox().get(idxMail).getSubject());

}

}

} else {

Toast.makeText(this, "No data", Toast.LENGTH\_SHORT).show();

}

}

private void sendEmail(String recipient, String subject, String message) {

int idx= -1;

for (int i= 0; i<users.size(); i++) {

if (users.get(i).getEmail().equals(recipient)) {

idx= i;

break;

}

}

if (idx >= 0) {

users.get(idx).setAllInbox(new Email(users.get(idxUser), subject, message));

} else {

User newUser= new User(recipient);

newUser.setAllInbox(new Email(users.get(idxUser), subject, message));

users.add(newUser);

}

}

}

InboxAdapter.java

public class InboxAdapter extends RecyclerView.Adapter<InboxAdapter.InboxViewHolder> {

private ArrayList<Email> inbox;

private OnItemClickCallback onItemClickCallback;

private OnItemLongClickCallback onItemLongClickCallback;

private OnImageViewClickCallback onImageViewClickCallback;

public InboxAdapter(ArrayList<Email> inbox) {

this.inbox = inbox;

}

public void setOnItemClickCallback(OnItemClickCallback onItemClickCallback) {

this.onItemClickCallback = onItemClickCallback;

}

public void setOnItemLongClickCallback(OnItemLongClickCallback onItemLongClickCallback) {

this.onItemLongClickCallback = onItemLongClickCallback;

}

public void setOnImageViewClickCallback(OnImageViewClickCallback onImageViewClickCallback) {

this.onImageViewClickCallback = onImageViewClickCallback;

}

@NonNull

@Override

public InboxViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {

View view= LayoutInflater.from(parent.getContext()).inflate(R.layout.inbox\_view, parent, false);

return new InboxViewHolder(view);

}

@Override

public void onBindViewHolder(@NonNull final InboxViewHolder holder, int position) {

final Email mail= this.inbox.get(position);

holder.tvIcon.setText(mail.getSender().getEmail().toUpperCase().charAt(0)+"");

holder.tvSender.setText(mail.getSender().getEmail());

holder.tvSubject.setText(mail.getSubject());

holder.tvMessage.setHint(mail.getMessage());

holder.tvDateTime.setText(mail.getDate());

if (mail.isStarred()) holder.ivStar.setImageResource(R.drawable.starred);

else holder.ivStar.setImageResource(R.drawable.unstarred);

holder.ivStar.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

if (onImageViewClickCallback != null) {

onImageViewClickCallback.onStarClicked(holder.getAdapterPosition());

}

}

});

holder.itemView.setOnLongClickListener(new View.OnLongClickListener() {

@Override

public boolean onLongClick(View view) {

if (onItemLongClickCallback != null) {

onItemLongClickCallback.onItemLongClicked(view, holder.getAdapterPosition());

}

return false;

}

});

holder.itemView.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

if (onItemClickCallback != null) {

onItemClickCallback.onItemClicked(holder.getAdapterPosition());

}

}

});

}

@Override

public int getItemCount() {

return this.inbox.size();

}

public class InboxViewHolder extends RecyclerView.ViewHolder {

TextView tvIcon, tvSender, tvSubject, tvMessage, tvDateTime;

ImageView ivStar;

public InboxViewHolder(@NonNull View itemView) {

super(itemView);

this.tvIcon= itemView.findViewById(R.id.tvIcon);

this.tvSender= itemView.findViewById(R.id.tvSender);

this.tvSubject= itemView.findViewById(R.id.tvSubject);

this.tvMessage= itemView.findViewById(R.id.tvMessage);

this.tvDateTime= itemView.findViewById(R.id.tvDateTime);

this.ivStar= itemView.findViewById(R.id.ivStar);

tvIcon.setBackground(this.drawBorder());

}

private GradientDrawable drawBorder() {

GradientDrawable gd= new GradientDrawable();

gd.setColor(Color.GREEN);

gd.setCornerRadius(7);

gd.setStroke(5, Color.BLACK);

return gd;

}

}

public interface OnItemClickCallback { void onItemClicked(int position); }

public interface OnItemLongClickCallback { void onItemLongClicked(View view, int position); }

public interface OnImageViewClickCallback { void onStarClicked(int position); }

}

User.java

public class User implements Serializable {

private String email;

private ArrayList<Email> allInbox;

private ArrayList<Email> starredInbox;

private ArrayList<Email> trash;

public User(String email) {

this.email= email;

this.allInbox= new ArrayList<>();

this.starredInbox= new ArrayList<>();

this.trash= new ArrayList<>();

}

@Override

public String toString() {

return this.email;

}

public String getEmail() { return this.email; }

public ArrayList<Email> getAllInbox() { return this.allInbox; }

public void setAllInbox(Email newEmail) { this.allInbox.add(newEmail); }

public ArrayList<Email> getStarredInbox() { return starredInbox; }

public void setStarredInbox(Email newStarred) { this.starredInbox.add(newStarred); }

public ArrayList<Email> getTrash() { return this.trash; }

public void setTrash(Email newTrash) { this.trash.add(newTrash); }

}

Email.java

public class Email implements Serializable {

private User sender;

private String subject, message, date, time;

private boolean starred, deleted;

public Email(User sender, String subject, String message) {

this.sender= sender;

this.subject= subject;

this.message= message;

this.date= getDateFormat();

this.time= getTimeFormat();

this.starred= false;

this.deleted= false;

}

@Override

public String toString() {

return this.sender.getEmail()+"-"+this.subject;

}

private String getDateFormat() {

Date date= Calendar.getInstance().getTime();

DateFormat dateFormat= new SimpleDateFormat("MMM dd");

return dateFormat.format(date);

}

private String getTimeFormat() {

Date date= Calendar.getInstance().getTime();

DateFormat dateFormat= new SimpleDateFormat("hh:mm");

return dateFormat.format(date);

}

public User getSender() { return this.sender; }

public String getSubject() { return this.subject; }

public String getMessage() { return this.message; }

public String getDate() { return this.date; }

public String getDateTime() {return this.date+", "+this.time; }

public boolean isStarred() { return this.starred; }

public void setStarred(boolean starred) { this.starred = starred; }

public boolean isDeleted() { return deleted; }

public void setDeleted(boolean deleted) { this.deleted = deleted; }

}