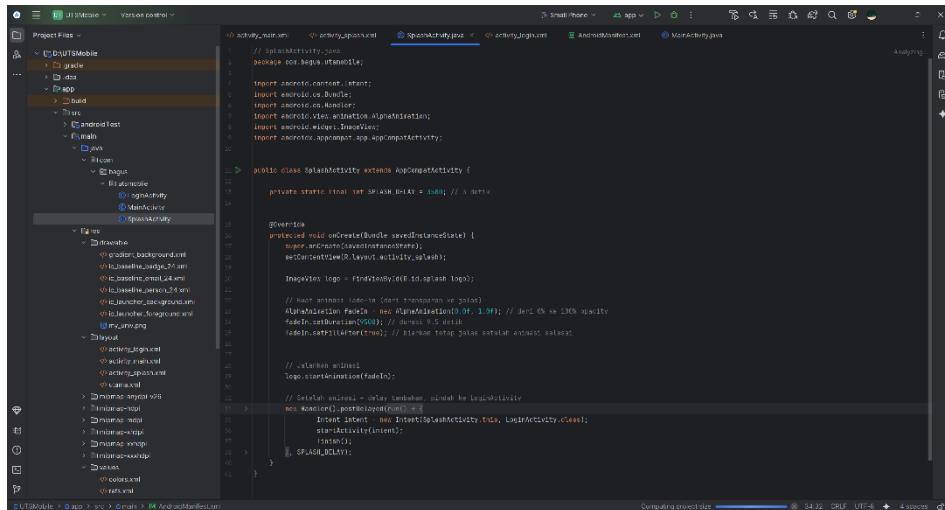


Laporan UTS
Pemograman Mobile



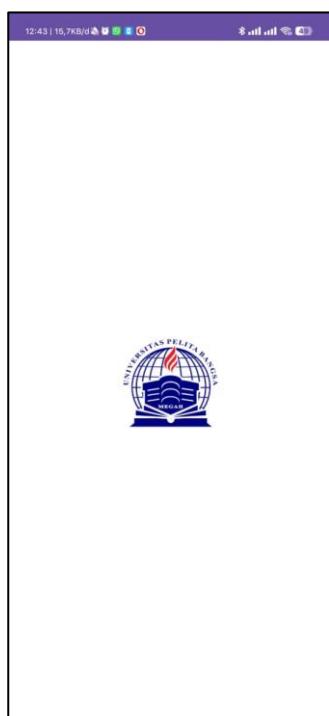
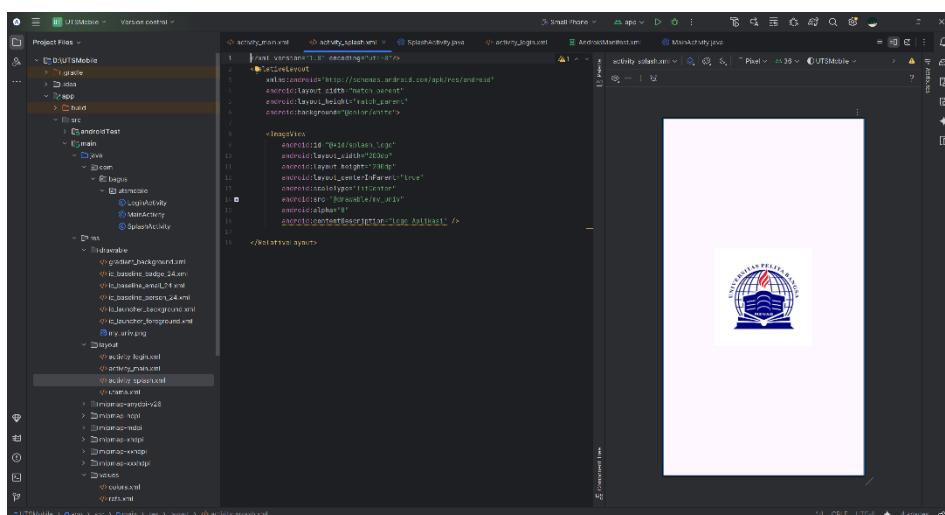
Nama : Bagus Sulistyo
NIM : 312410011
Jurusan : Teknik Informatika
Matkul : Pemograman Mobile
Dosen : Yogi Yulianto, M.Kom.

1. Program & Tampilan Splash



```
activity_splash.xml    activity_splash.java    SplashActivity.java    activity_login.xml    AndroidManifest.xml
```

```
1 // splashActivity.java
2 package com.splash.utsmobile;
3
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.view.View;
7 import android.widget.ImageView;
8 import androidx.appcompat.app.AppCompatActivity;
9 import androidx.core.app.ActivityCompat;
10 import androidx.core.app.NotificationCompat;
11
12 public class SplashActivity extends AppCompatActivity {
13
14     private static final int SPLASH_WAIT = 3000; // 3 seconds
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(R.layout.activity_splash);
20
21         ImageView logo = findViewById(R.id.splash_logo);
22
23         // Set splash logo
24         logo.setImageResource(R.drawable.splash_logo);
25
26         // Set splash intent
27         Intent intent = new Intent(SplashActivity.this, LoginActivity.class);
28         intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);
29
30         ActivityCompat.startActivity(this, intent, null);
31     }
32 }
```



Program SplashActivity dan Activity_splash,
Beserta Tampilan Run program

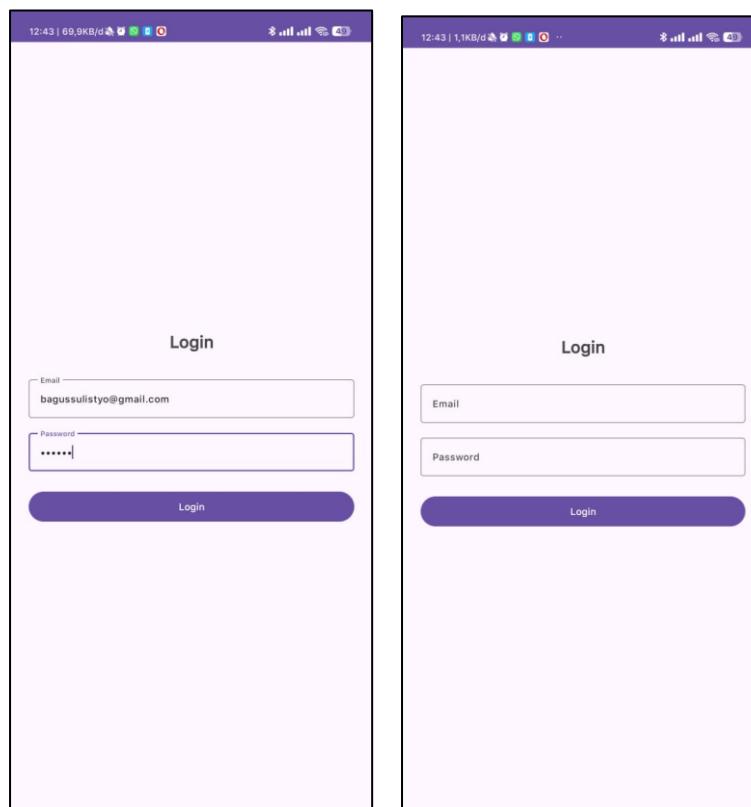
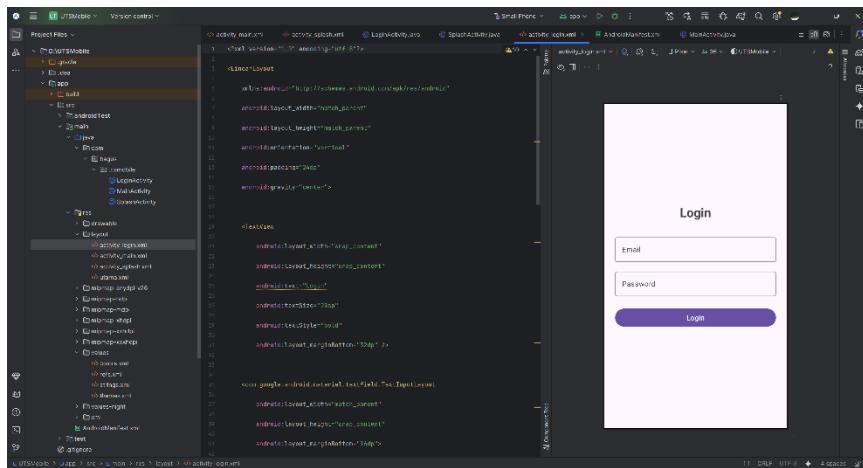
2. Program & Tampilan Login



```
activity_main.xml    activity_login.xml    LoginActivity.java    splashActivity.java    activity_main.xml    AndroidManifest.xml    ManifestOverview

Project Files -> <Android> -> <app> -> <src> -> <com> -> <MyApp> -> <LoginActivity> -> LoginActivity.java

1 // LogInActivity.java
2 package com.example.myapplication;
3
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.Button;
7 import android.widget.EditText;
8 import android.widget.Toast;
9 import androidx.appcompat.app.AppCompatActivity;
10 import com.google.android.material.textfield.TextInputEditText;
11
12 public class LoginActivity extends AppCompatActivity {
13
14     EditText editTextEmail;
15     EditText editTextPassword;
16     Button buttonLogin;
17
18     @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21         setContentView(R.layout.activity_login);
22
23         editTextEmail = findViewById(R.id.editTextEmail);
24         editTextPassword = findViewById(R.id.editTextPassword);
25         buttonLogin = findViewById(R.id.buttonLogin);
26
27         buttonLogin.setOnClickListener(new View.OnClickListener() {
28             @Override
29             public void onClick(View v) {
30                 String email = editTextEmail.getText().toString();
31                 String password = editTextPassword.getText().toString();
32
33                 if (email.isEmpty() || password.isEmpty()) {
34                     Toast.makeText(LoginActivity.this, "Both fields must not be empty!", Toast.LENGTH_SHORT).show();
35                 } else {
36                     // Intent intent = new Intent(LoginActivity.this, MainActivity.class);
37                     // startActivity(intent);
38                     // finish();
39                 }
40             }
41         });
42     }
43 }
```



Program LoginActivity & Activity_Login, Beserta Tampilan Run Program

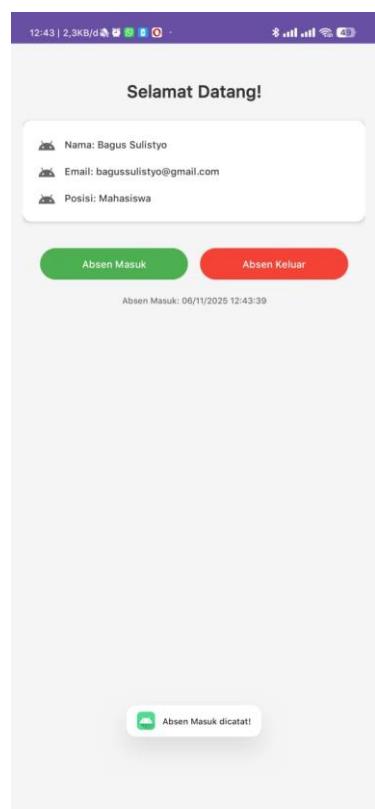
3. Program & Tampilan Dashboard

```
1 package com.example.absensi;
2
3 import android.os.Bundle;
4 import android.view.LayoutInflater;
5 import android.view.View;
6 import android.view.ViewGroup;
7 import android.widget.TextView;
8
9 import androidx.annotation.NonNull;
10 import androidx.annotation.Nullable;
11 import androidx.fragment.app.Fragment;
12 import androidx.recyclerview.widget.LinearLayoutManager;
13 import androidx.recyclerview.widget.RecyclerView;
14
15 import com.google.android.gms.tasks.OnCompleteListener;
16 import com.google.android.gms.tasks.Task;
17 import com.google.firebase.firestore.QuerySnapshot;
18
19 import java.util.ArrayList;
20 import java.util.List;
21
22 public class HomeFragment extends Fragment {
23     RecyclerView recyclerView;
24     AdapterAbsensi adapterAbsensi;
25     List<DataAbsensi> list;
26
27     @Nullable
28     @Override
29     public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
30         View view = inflater.inflate(R.layout.fragment_home, container, false);
31
32         recyclerView = view.findViewById(R.id.recycleView);
33         recyclerView.setHasFixedSize(true);
34         recyclerView.setLayoutManager(new LinearLayoutManager(getActivity()));
35
36         list = new ArrayList<>();
37
38         loadAbsensi();
39
40         return view;
41     }
42
43     private void loadAbsensi() {
44         FirebaseFirestore db = FirebaseFirestore.getInstance();
45         db.collection("absensi").get().addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
46             @Override
47             public void onComplete(@NonNull Task<QuerySnapshot> task) {
48                 if (task.isSuccessful()) {
49                     QuerySnapshot querySnapshot = task.getResult();
50                     for (DocumentSnapshot document : querySnapshot.getDocuments()) {
51                         DataAbsensi dataAbsensi = document.toObject(DataAbsensi.class);
52                         list.add(dataAbsensi);
53                     }
54
55                     adapterAbsensi = new AdapterAbsensi(list);
56                     recyclerView.setAdapter(adapterAbsensi);
57                 }
58             }
59         });
60     }
61
62     public class AdapterAbsensi extends RecyclerView.Adapter<AdapterAbsensi.MyViewHolder> {
63         List<DataAbsensi> list;
64
65         public AdapterAbsensi(List<DataAbsensi> list) {
66             this.list = list;
67         }
68
69         @NonNull
70         @Override
71         public AdapterAbsensi.MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
72             View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.item_absensi, parent, false);
73             MyViewHolder myViewHolder = new MyViewHolder(view);
74             return myViewHolder;
75         }
76
77         @Override
78         public void onBindViewHolder(@NonNull AdapterAbsensi.MyViewHolder holder, int position) {
79             DataAbsensi dataAbsensi = list.get(position);
80             holder.tglAbsensi.setText(dataAbsensi.getTanggal());
81             holder.namaAbsen.setText(dataAbsensi.getNama());
82             holder.jenisAbsen.setText(dataAbsensi.getJenisAbsen());
83             holder.statusAbsen.setText(dataAbsensi.getStatus());
84         }
85
86         @Override
87         public int getItemCount() {
88             return list.size();
89         }
90     }
91
92     public class MyViewHolder extends RecyclerView.ViewHolder {
93         TextView tglAbsensi;
94         TextView namaAbsen;
95         TextView jenisAbsen;
96         TextView statusAbsen;
97
98         public MyViewHolder(@NonNull View itemView) {
99             super(itemView);
100            tglAbsensi = itemView.findViewById(R.id.tglAbsensi);
101            namaAbsen = itemView.findViewById(R.id.namaAbsen);
102            jenisAbsen = itemView.findViewById(R.id.jenisAbsen);
103            statusAbsen = itemView.findViewById(R.id.statusAbsen);
104        }
105    }
106}
```

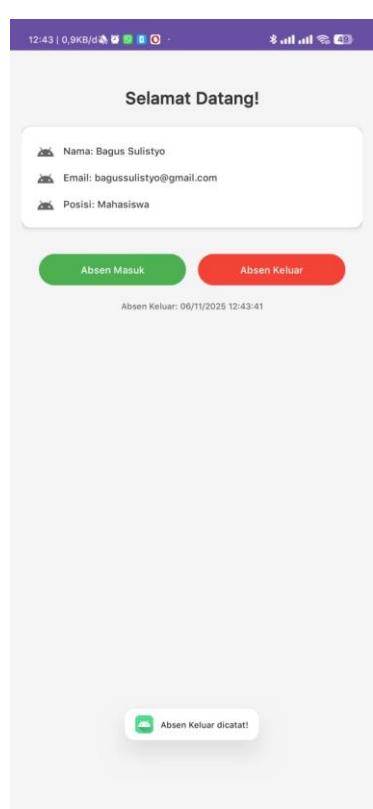
Sebelum Absen Masuk



Setelah Absen Masuk



Setelah Absen Keluar



4. Kesimpulan

- **Implementasi Program Mobile :** Laporan ini mendokumentasikan implementasi program aplikasi mobile, mencakup beberapa tahap utama, yaitu :
 1. **Tampilan Splash**
 2. **Tampilan Login**
 3. **Program & Tampilan Dashboard**
- **Fungsional Utama :** Program yang dikembangkan berfokus pada **Absensi** yang di akses melalui dashboard. Tampilan Dashboard menunjukkan perubahan status sebelum absensi masuk, setelah masuk dan setelah absensi keluar.
- **Dokumentasi Kode:** Setiap bagian implementasi (Splash, Login, Dashboard) didokumentasikan dengan menyertakan cuplikan **kode program** (dalam mode gelap/dark theme) dan **tampilan antarmuka** (UI) dari aplikasi yang dihasilkan.