

# **LAMPIRAN**

## **Lampiran 1**

Cara memakai aplikasi iPresence:

1. Install apk aplikasi pada handphone
2. Pilih tombol daftar apabila belum memiliki akun, apabila sudah memiliki akun, masukkan email dan password yang diminta dan klik tombol masuk.
3. Pada halaman utama, untuk melakukan presensi atau check-in pengguna harus mendaftar praktikum terlebih dahulu yaitu dengan mengklik tombol 'daftar praktikum'
4. Apabila pengguna telah mendaftar, pengguna dapat melakukan presensi
5. Apabila pengguna presensi di mata praktikum yang tidak diambil, maka presensi akan gagal dan akan muncul notifikasi kepada pengguna
6. Untuk melihat jadwal praktikum yang diambil, pengguna dapat melihat jadwal dengan mengklik tombol 'jadwal praktikum'
7. Untuk melihat presensi praktikum, pengguna dapat melihatnya dengan mengklik tombol presensi praktikum
8. Untuk keluar dari aplikasi, pengguna dapat mengklik tombol 'keluar'

## Lampiran 2

### Script Mesosfer

```
public class MesosferApp extends Application {
    @Override
    public void onCreate() {
        super.onCreate();

        Mesosfer.setPushNotification(true);

        // initialize Mesosfer SDK
        Mesosfer.initialize(this, "Bei00uy4o", "YSyWUTrJYM0mowVn5uBZaIxK3pSV0iGH");

        // initialize Cubeacon SDK
        Cubeacon.initialize(this);
    }
}
```

### Script Register

```
public void handleRegister(View view) {
    // get all value from input
    email = textEmail.getText().toString();
    password = textPassword.getText().toString();
    firstname = textFirstname.getText().toString();
    lastname = textLastname.getText().toString();
    nim = textNIM.getText().toString();

    // validating input values
    if (!isInputValid()) {
        // return if there is an invalid input
        return;
    }

    registerUser();
}
```

## Script Login

```
public void handleLogin(View view) {
    String username = textUsername.getText().toString();
    String password = textPassword.getText().toString();

    if (TextUtils.isEmpty(username)) {
        Toast.makeText(this, "Alamat email belum diisi", Toast.LENGTH_LONG).show();
        textUsername.requestFocus();
        return;
    }

    if (TextUtils.isEmpty(password)) {
        Toast.makeText(this, "Password belum diisi", Toast.LENGTH_LONG).show();
        textPassword.requestFocus();
        return;
    }

    loading.setMessage("Logging in...");
    loading.show();
    MesosferUser.loginAsync(username, password, (user, e) -> {
        loading.dismiss();
        if (e != null) {
            AlertDialog.Builder builder = new AlertDialog.Builder(LoginActivity.this);
            builder.setTitle("Login Failed");
            builder.setMessage(
                "Maaf, email/password salah"
            );
            dialog = builder.show();
            return;
        }

        Toast.makeText(LoginActivity.this, "User logged in...", Toast.LENGTH_SHORT).show();
        Intent intent = new Intent(LoginActivity.this, MainUtamaActivity.class);
        startActivity(intent);
        finish();
    });
}
```

## Script Main Utama

```
private void panggil() {
    final MesosferUser user = MesosferUser.getCurrentUser();
    user.fetchAsync((mesosferUser, e) → {
        // hide progress dialog loading
        loading.dismiss();
    });

    String namaawal = user.getFirstName();
    String namaakhir = user.getLastName();
    String nama = namaawal+" "+namaakhir;
    String nim = user.getDataString("nim");

    setContentView(R.layout.activity_main_utama);
    TextView textView = (TextView) findViewById(R.id.text_view_id);
    textView.setText(nama);

    TextView textView3 = (TextView) findViewById(R.id.text_nik);
    textView3.setText(nim);
}
```

## Script Daftar

```
public void daftar() {
    MesosferQuery<MesosferData> daftar = MesosferData.getQuery("DataPraktikum");

    daftar.findAsync((list, e) → {
        loading.dismiss();
        DaftarActivity.this.data = list;
        mapDataList.clear();
        for (MesosferData ini : list){
            waktupraktikum = ini.getDataString("praktikum");
            nama = ini.getDataString("namapraktikum");

            Map<String, String> map = new HashMap<>();
            map.put("id", nama);
            map.put("data", "Teknik Elektro UMY");
            mapDataList.add(map);
        }
        runOnUiThread(() → { adapter.notifyDataSetChanged(); });
    });
}
```

```

@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
    final MesosferData indah = this.data.get(position);
    Intent intent = new Intent(DaftarActivity.this, DaftarPraktikumActivity.class);
    intent.putExtra("BEACONS", indah.getDataString("praktikum"));
    startActivity(intent);
    recreate();
}

```

```

public void jadwalpraktikum(){

    MesosferQuery<MesosferData> DataPraktikum = MesosferData.getQuery(inidia);
    DataPraktikum.findAsync((list, e) → {
        if (e != null) {
            // handle the exception
            return;
        }

        DaftarPraktikumActivity.this.dian = list;
        data.clear();

        for (MesosferData jadwalpraktikum : list ){

            hari = jadwalpraktikum.getDataString("hari");
            jam = jadwalpraktikum.getDataString("jam");
            kodepraktikum = jadwalpraktikum.getDataString("kodepraktikum");

            Map<String, String> map = new HashMap<>();

            map.put("title", hari);
            map.put("subtitle", jam);
            data.add(map);
        }
        runOnUiThread(() → { adapter.notifyDataSetChanged(); });
    });
}

```

```
else if (jumlah>=3){  
    alertDialogBuilder  
        .setMessage("Daftar Praktikum Gagal")  
        .setCancelable(false)  
        .setPositiveButton("OK", (dialog, id) → {  
            Intent intent = new Intent(SimpanJadwalActivity.this, MainUtamaActivity.class);  
            startActivity(intent);  
            finishAffinity();  
        });  
    // membuat alert dialog dari builder  
    AlertDialog alertDialog = alertDialogBuilder.create();  
    // menampilkan alert dialog  
    alertDialog.show();  
}  
  
user.setData("jumlah", String.valueOf(jumlah+1));  
user.saveAsync((e) → { //loading.dismiss(); });  
}
```

## Script Presensi

```
@Override
public void didRangeBeaconsInRegion(final List<CBBeacon> beacons, CBRegion region) {
    this.beaconsmes = beacons;

    Map<String, String> map;
    String title, subtitle;

    // clear data
    data.clear();

    for (CBBeacon beacon : beacons) {

        //memanggil bucket data praktikum
        final MesosferQuery<MesosferData> DataPraktikumQuery = MesosferData.getQuery("DataPraktikum");

        //persamaan yang diambil yaitu major dan minor
        DataPraktikumQuery.whereEqualTo("major", String.valueOf(beacon.getMajor()));
        DataPraktikumQuery.whereEqualTo("minor", String.valueOf(beacon.getMinor()));

        //memanggil
        DataPraktikumQuery.findAsync((list, e) -> {
            if (e != null) {
                return;
            }

            RangingActivity.this.listdata = list;
            for (MesosferData satu : list) {
                kodepraktikum = satu.getDataString("kodepraktikum");
                namapraktikum = satu.getDataString("namapraktikum");
            }
        });

        if (beacon.getMajor() == 1 && beacon.getMinor() == 284) {
            MesosferData DataPraktikum = MesosferData.createWithObjectId("g0frlwcRQt");
            name = DataPraktikum.getDataString("namapraktikum");
            ruang = DataPraktikum.getDataString("ruangan");
        } else if (beacon.getMajor() == 2 && beacon.getMinor() == 285) {
            MesosferData DataPraktikum = MesosferData.createWithObjectId("352ze9NkBN");
            name = DataPraktikum.getDataString("namapraktikum");
            ruang = DataPraktikum.getDataString("ruangan");
        } else if (beacon.getMajor() == 3 && beacon.getMinor() == 286) {
            MesosferData DataPraktikum = MesosferData.createWithObjectId("wvfrNAVvNR");
            name = DataPraktikum.getDataString("namapraktikum");
            ruang = DataPraktikum.getDataString("ruangan");
        }

        //menampilkan
        map = new HashMap<>();
        map.put("title", name);
        map.put("subtitle", "Teknik Elektro UMY");
        data.add(map);
    }

    runOnUiThread() -> {
        adapter.notifyDataSetChanged();
        if (getSupportActionBar() != null) {
            getSupportActionBar().setSubtitle("Praktikum yang terdeteksi: " + beacons.size());
        }
    }
};
}
```