1. Halaman Pengesahan
2. Halaman Persembahan dan Motto

**HALAMAN PERSEMBAHAN**

Kupersembahkan karya ini untuk orang-orang hebat dibawah ini :

Mamakku Samilah yang selalu memberikan cinta dan kasih sayang pada diriku melebihi apapun yang beliau tau.

Bapakku Ngatiman yang selalu berkerja keras dengan segala kemampuanya untuk menghidupi keluarga dan membiayai diriku sampai lulus menjadi sarjana.

Kedua Adikku Sila dan Vellia yang selalu ada dihatiku.

Kerabat dan saudara yang tak henti-hentinya menanyakan kepada diriku, kapan saya wisuda?

**MOTTO**

Mentari terbit saat pagi dan terbenam diwaktu sore. Hujan turun setelah mendung dan berhenti sebelum pelangi. Ketahuilah bawa semua hal mempunyai waktu dan ritme nya tersendiri, begitu pula dengan dirimu. Maka jangan pernah membandingkan dirimu dengan siapapun atau apapun. (Penulis)

Jangan menepi, jangan berhenti, tetap terkini, tetap terkini (Penulis)

KATA PENGANTAR

Puji syukur atas kehadirat Tuhan Yang Maha Esa, berkat kehendaknya penulis dapat menyelesaikan skripsi yang berjudul “APLIKASI AGENDA BERBASIS ANDROID DENGAN FITUR PUSH NOTIFICATION DAN REMINDER”. Sehingga penulis dapat memenuhi salah satu syarat untuk lulus dari program studi S1 Teknik Informatika STMIK AKAKOM YOGYAKARTA.

Penulisan skripsi ini dapat terselesaikan atas bantuan dari berbagai pihak. Maka dengan ini penulis ingin menyapaikan ucapan terimaksih kepada :

1. Bapak Totok Suprawoto selaku ketua umum STMIK AKAKOM YOGYAKARTA
2. Ibu Dini Fakta Sari selaku ketua jurusan Teknik Informatika STMIK AKAKOM YOGYAKARTA
3. Ibu Indra Yatini selaku dosen pembimbing skripsi yang senantiasa membimbing dan memberikan arahan kepada penulis dalam mengegerjakan skripsi ini hingga selesai.
4. Ibu L.N Hananingrum selaku dosen narasumber skripsi yang senantiasa mengkritisi dan memberi masukkan kepada penulis.
5. Bapak Tri Prabawa selaku dosen narasumber yang senantiasa mengkritisi dan memberi masukkan kepada penulis.
6. Teman-Teman Sekelasku yang selalu mensupport dan memberikan semangat pada penulis untuk terus berusaha.

Semoga skirpsi ini dapat memberikan manfaat khususnya kepada Akademisi STMIK AKAKOM dan UKM Informatika dan Komputer selaku organisasi yang menjadi objek dalam penelitian ini.

Yogyakarta, xx xx xxxx

Resta Bayu Setiawan

----------------backup-------------

* + 1. **Implementasi Halaman Login Pengguna**

Pada saat pertama kali aplikasi dijalankan, maka tampilan utama yang akan dilihat oleh pengguna adalah halaman login. Adapun kode program nya seperti berikut :

|  |
| --- |
| public class SigninActivity extends AppCompatActivity implements View.OnClickListener {  FirebaseAuth mAuth;  EditText editTextEmail, editTextPassword;  ProgressBar progressBar;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.signin\_page);  mAuth = FirebaseAuth.getInstance();  editTextEmail = (EditText) findViewById(R.id.editTextEmail);  editTextPassword = (EditText) findViewById(R.id.editTextPassword);  progressBar = (ProgressBar) findViewById(R.id.progressbar);  findViewById(R.id.textViewSignup).setOnClickListener(this);  findViewById(R.id.buttonLogin).setOnClickListener(this);  }  private void userLogin() {  String email = editTextEmail.getText().toString().trim();  String password = editTextPassword.getText().toString().trim();  if (email.isEmpty()) {  editTextEmail.setError("Masukkan Email !");  editTextEmail.requestFocus();  return;  }  if (!Patterns.EMAIL\_ADDRESS.matcher(email).matches()) {  editTextEmail.setError("Masukkan Email yang Benar !");  editTextEmail.requestFocus();  return;  }  if (password.isEmpty()) {  editTextPassword.setError("Masukkan Password !");  editTextPassword.requestFocus();  return;  }  if (password.length()<6) {  editTextPassword.setError("Minimal harus 6 karakter");  editTextPassword.requestFocus();  return;  }  progressBar.setVisibility(View.VISIBLE);  mAuth.signInWithEmailAndPassword(email, password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  @Override  public void onComplete(@NonNull Task<AuthResult> task) {  progressBar.setVisibility(View.GONE);  if(task.isSuccessful()) {  finish();  Intent intent = new Intent(SigninActivity.this, MainActivity.class);  intent.addFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP);  startActivity(intent);  } else {  Toast.makeText(getApplicationContext(),task.getException().getMessage(), Toast.LENGTH\_SHORT).show();  }  }  });  }  @Override  protected void onStart() {  super.onStart();  if(mAuth.getCurrentUser() !=null) {  finish();  startActivity(new Intent(this,MainActivity.class));  }  }  @Override  public void onClick(View view) {  switch (view.getId()) {  case R.id.buttonLogin:  userLogin();  break;  case R.id.textViewSignup:  finish();  startActivity(new Intent(this, SignupActivity.class));  break;  }  }  } |

* + 1. **Implementasi Halaman Registrasi Pengguna**

Halaman registrasi pengguna akan tampil saat user yang belum mempunyai akun kemudian ingin registrasi/mendaftar. Adapun kode program nya seperti berikut :

|  |
| --- |
| public class SignupActivity extends AppCompatActivity implements View.OnClickListener {  ProgressBar progressBar;  EditText editTextEmail, editTextPassword;  private FirebaseAuth mAuth;  @Override  protected void onCreate (Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.signup\_page);  editTextEmail = (EditText) findViewById(R.id.editTextEmail);  editTextPassword = (EditText) findViewById(R.id.editTextPassword);  progressBar = (ProgressBar) findViewById(R.id.progressbar);  mAuth = FirebaseAuth.getInstance();  findViewById(R.id.buttonSignUp).setOnClickListener(this);  findViewById(R.id.textViewLogin).setOnClickListener(this);  }  private void registerUser() {  String email = editTextEmail.getText().toString().trim();  String password = editTextPassword.getText().toString().trim();  if (email.isEmpty()) {  editTextEmail.setError("Masukkan Email !");  editTextEmail.requestFocus();  return;  }  if (!Patterns.EMAIL\_ADDRESS.matcher(email).matches()) {  editTextEmail.setError("Masukkan Email yang Benar !");  editTextEmail.requestFocus();  return;  }  if (password.isEmpty()) {  editTextPassword.setError("Masukkan Password !");  editTextPassword.requestFocus();  return;  }  if (password.length()<6) {  editTextPassword.setError("Minimal harus 6 karakter");  editTextPassword.requestFocus();  return;  }  progressBar.setVisibility(View.VISIBLE);  mAuth.createUserWithEmailAndPassword(email,password).addOnCompleteListener(new OnCompleteListener<AuthResult>() {  @Override  public void onComplete(@NonNull Task<AuthResult> task) {  progressBar.setVisibility(View.GONE);  if(task.isSuccessful()) {  Toast.makeText(getApplicationContext(),"Registrasi Berhasil !",Toast.LENGTH\_SHORT).show();  /\* finish();  startActivity(new Intent(SignupActivity.this, MainActivity.class));  Intent intent = new Intent(SignupActivity.this, MainActivity.class);  intent.addFlags(Intent.FLAG\_ACTIVITY\_CLEAR\_TOP);  startActivity(intent); \*/  } else {  if(task.getException()instanceof FirebaseAuthUserCollisionException) {  Toast.makeText(getApplicationContext(), "Email ini sudah terdaftar !", Toast.LENGTH\_SHORT).show();  } else {  Toast.makeText(getApplicationContext(), task.getException().getMessage(), Toast.LENGTH\_SHORT).show();  }  }  }  });  }  @Override  public void onClick (View view) {  switch (view.getId()) {  case R.id.buttonSignUp:  registerUser();  break;  case R.id.textViewLogin:  finish();  startActivity(new Intent(this, SigninActivity.class));  break;  }  }  } |

* + 1. **Implementasi Halaman Utama Aplikasi**

Halaman utama akan tampil saat user yang yang sudah mempunyai akun kemudian melakukan login. Adapun kode program nya seperti berikut :

|  |
| --- |
| public class MainActivity extends AppCompatActivity {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*FIREBASE HELPER START\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  public class FirebaseHelper {  DatabaseReference db;  ArrayList<Classified> classifieds = new ArrayList<>();  ListView mListView;  Context c;  ProgressBar progressBar;  /\*  let's receive a reference to our FirebaseDatabase  \*/  public FirebaseHelper(DatabaseReference db, Context context, ListView mListView) {  this.db = db;  this.c = context;  this.mListView = mListView;  this.retrieve();  progressBar = (ProgressBar) findViewById(R.id.progressbar);  progressBar.setVisibility(View.VISIBLE);  }  /\*  Retrieve and Return them clean data in an arraylist so that they just bind it to ListView.  \*/  public ArrayList<Classified> retrieve() {  db.child("classified").addValueEventListener(new ValueEventListener() {  @Override  public void onDataChange(DataSnapshot dataSnapshot) {  classifieds.clear();  if (dataSnapshot.exists() && dataSnapshot.getChildrenCount() > 0) {  for (DataSnapshot ds : dataSnapshot.getChildren()) {  //Now get Classified Objects and populate our arraylist.  Classified classified = ds.getValue(Classified.class);  classifieds.add(classified);  }  adapter = new CustomAdapter(c, classifieds);  mListView.setAdapter(adapter);  new Handler().post(new Runnable() {  @Override  public void run() {  mListView.smoothScrollToPosition(classifieds.size());  }  });  }  progressBar.setVisibility(View.GONE);  }  @Override  public void onCancelled(DatabaseError databaseError) {  Log.d("mTAG", databaseError.getMessage());  Toast.makeText(c, "ERROR " + databaseError.getMessage(), Toast.LENGTH\_LONG).show();  }  });  return classifieds;  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CUSTOM ADAPTER START\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  class CustomAdapter extends BaseAdapter {  Context c;  ArrayList<Classified> classifieds;  public CustomAdapter(Context c, ArrayList<Classified> classifieds) {  this.c = c;  this.classifieds = classifieds;  }  @Override  public int getCount() {  return classifieds.size();  }  @Override  public Object getItem(int position) {  return classifieds.get(getCount() - position - 1);  }  @Override  public long getItemId(int position) {  return position;  }  @Override  public View getView(int position, View convertView, ViewGroup parent) {  if (convertView == null) {  convertView = LayoutInflater.from(c).inflate(R.layout.model, parent, false);  }  TextView nameTextView = convertView.findViewById(R.id.nameTextView);  TextView quoteTextView = convertView.findViewById(R.id.quoteTextView);  TextView descriptionTextView = convertView.findViewById(R.id.descriptionTextView);  final Classified s = (Classified) this.getItem(position);  nameTextView.setText(s.getNama());  quoteTextView.setText(s.getDeskripsi());  descriptionTextView.setText(s.getTanggal());  //ONITECLICK  convertView.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View v) {  // Toast.makeText(c, s.getNama(), Toast.LENGTH\_SHORT).show();  String[] classifieds = {  s.getNama(),  s.getTanggal(),  s.getWaktu(),  s.getTempat(),  s.getDeskripsi(),  };  openDetailActivity(classifieds);  }  });  return convertView;  }  //OPEN DETAIL ACTIVITY  private void openDetailActivity(String[] data)  {  Intent i = new Intent(MainActivity.this, DetailActivity.class);  i.putExtra("NAMA\_KEY",data[0]);  i.putExtra("TANGGAL\_KEY",data[1]);  i.putExtra("WAKTU\_KEY",data[2]);  i.putExtra("TEMPAT\_KEY",data[3]);  i.putExtra("DESKRIPSI\_KEY",data[4]);  startActivity(i);  }  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*TOOLBAR MENU\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  @Override  public boolean onCreateOptionsMenu(Menu menu) {  MenuInflater inflater = getMenuInflater();  inflater.inflate(R.menu.menu, menu);  return true;  }  @Override  public boolean onOptionsItemSelected(MenuItem item) {  switch (item.getItemId()) {  case R.id.menuLogout:  FirebaseAuth.getInstance().signOut();  finish();  startActivity(new Intent(this, SigninActivity.class));  break;  }  return true;  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*EXIT DIALOG\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  @Override  public void onBackPressed() {  new AlertDialog.Builder(this, R.style.AlertDialogStyle)  .setTitle("Apakah anda yakin ingin keluar ?")  .setPositiveButton("Ya", new DialogInterface.OnClickListener()  {  @Override  public void onClick(DialogInterface dialog, int which) {  finish();  }  })  .setNegativeButton("Tidak", null)  .show();  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*MAIN ACTIVITY CONTINUATION\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  //instance fields  DatabaseReference db;  FirebaseHelper helper;  CustomAdapter adapter;  ListView mListView;  FirebaseAuth mAuth;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_main);  //nyobo  mAuth = FirebaseAuth.getInstance();  Toolbar toolbar = findViewById(R.id.toolbar);  setSupportActionBar(toolbar);  toolbar.setTitleTextColor(getResources().getColor(android.R.color.white));  FirebaseMessaging.getInstance().subscribeToTopic("news");  mListView = (ListView) findViewById(R.id.myListView);  //initialize firebase database  db = FirebaseDatabase.getInstance().getReference();  helper = new FirebaseHelper(db, this, mListView);  }  } |

* + 1. **Implementasi Halaman Detail Acara**

Halaman detail acara merupakan halaman yang menampilkan detail/rincian acara secara lengkap meliputi nama acara, waktu, tanggal, deskripsi/ketereangan, dan tempat diselenggarakan. Adapun kode program nya seperti berikut :

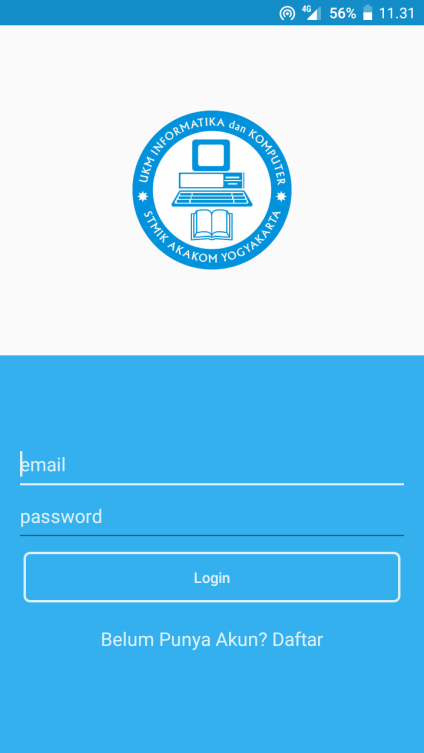
|  |
| --- |
| public class DetailActivity extends AppCompatActivity {  TextView namaTxt,tanggalTxt,waktuTxt,tempatTxt,deskripsiTxt;  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.activity\_detail);  namaTxt = (TextView) findViewById(R.id.namaDetailTextView);  tanggalTxt= (TextView) findViewById(R.id.tanggalDetailTextView);  waktuTxt= (TextView) findViewById(R.id.waktuDetailTextView);  tempatTxt= (TextView) findViewById(R.id.tempatDetailTextView);  deskripsiTxt = (TextView) findViewById(R.id.deskripsiDetailTextView);  //GET INTENT  Intent i=this.getIntent();  //RECEIVE DATA  String nama=i.getExtras().getString("NAMA\_KEY");  String tanggal=i.getExtras().getString("TANGGAL\_KEY");  String waktu=i.getExtras().getString("WAKTU\_KEY");  String tempat=i.getExtras().getString("TEMPAT\_KEY");  String deskripsi=i.getExtras().getString("DESKRIPSI\_KEY");  //BIND DATA  namaTxt.setText(nama);  tanggalTxt.setText(tanggal);  waktuTxt.setText(waktu);  tempatTxt.setText(tempat);  deskripsiTxt.setText(deskripsi);  }  } |

* 1. **Pembahasan Sistem**

Pada bagian ini akan membahas mengenai hasil dari implementasi sistem yang telah di uji coba dengan dijalankan di perangkat smartphone android dengan keadaan yang sebenarnya.

* + 1. **Halaman Login**

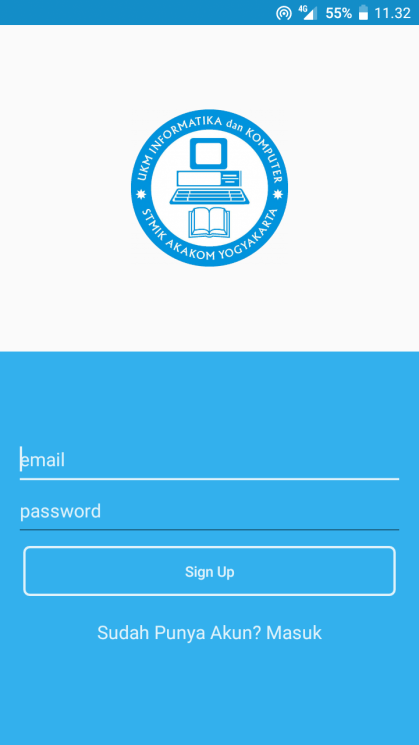
Berikut ini merupakan tampilan halaman login yang telah dijalankan di perangkat smartphone android. Apabila user sudah mempunyai akun maka user dapat melakukan login dengan memasukkan email dan password miliknya. Apabila user belum mempunyai akun dan ingin mendaftar maka user dapat memilih textfield pada tampilan paling bawah. Kemudian user akan diarahkan ke halaman regsitrasi.



Gambar 4.1. Tampilan Halaman Login User

* + 1. **Halaman Registrasi**

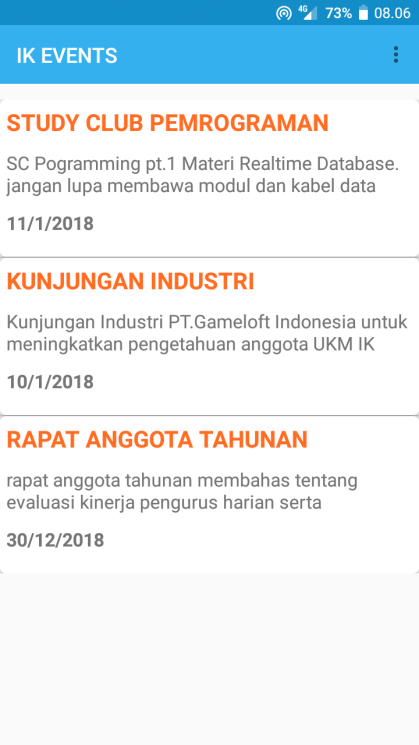
Berikut ini merupakan tampilan halaman registrasi yang telah dijalankan di perangkat smartphone android. Apabila user belum mempunyai akun maka user dapat melakukan registrasi dengan memasukkan email dan password. Setelah registrasi berhasil maka user dapat memilih textfield pada tampilan paling bawah untuk melakukan login.



Gambar 4.2. Tampilan Halaman Registrasi User

* + 1. **Halaman Utama**

User yang telah berhasil login maka akan tampil halaman utama seperti pada gambar dibawah. Dimana pada halaman utama ini terdapat daftar agenda kegiatan yang akan diselanggarakan oleh UKM IK berupa nama kegiatan, deskripsi, dan tanggal dilaksanakannya. User dapat mengklik daftar agenda yang ada untuk melihat informasi detail acara.



Gambar 4.3. Tampilan Halaman Utama

* + 1. **Halaman Detail Acara**

Saat user membuka daftar acara yang ada sebelumnya, maka akan tampil halaman detail acara seperti pada gambar dibawah ini. Halaman detail acara memuat informasi lengkap acara seperti nama acara, tanggal, waktu, tempat, dan deskripsi acara.



Gambar 4.4. Tampilan Halaman Detail Acara

# **BAB V**

# **PENUTUP**

* 1. **Kesimpulan**

fkdfjdkfdjkfdfjdkfjdkfdjfjdsf

* 1. **Saran**

dkjfkdslfjkdjfskdjfkdjfkdlfjsdklfj