

**LAPORAN TUGAS BESAR
PEMROGRAMAN MOBILE
APLIKASI PENGINGAT TUGAS**

DISUSUN OLEH :

MUHAMMAD SUGIANTO

1809075023



**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS MULAWARMAN**

SAMARINDA

2021

BAB I

PENDAHULUAN

A. Latar Belakang

Pada umumnya kebanyakan orang membuat sebuah daftar list masih dibuat dengan cara manual, yaitu mencatatkan daftar tugas yang akan kita lakukan pada sebuah kertas. Mengapa penting membuat to do list? sesuatu yang besar biasanya harus dimulai dari kebiasaan kecil yang terlihat sederhana namun berdampak cukup banyak jika dilakukan dengan baik. Salah satu contohnya adalah kebiasaan membuat to do list sebelum bekerja. Secara teori mungkin kita sudah tahu dan pernah melakukan hal ini, namun karena merasa sibuk dan terburu kadang kita lupa dan tidak disiplin dalam melakukannya. Selain itu anggapan bahwa hal ini tidak terlalu penting menjadi faktor pendukung untuk tidak membuat to do list tersebut. Namun jika kita melihat manfaat yang didapat dengan membuat to do list yang baik, mungkin kita bisa mempertimbangkan kembali untuk lebih rajin dan sadar akan pentingnya membuat to do list. Manfaat utama dari membuat to do list adalah membantu kita berpikir secara bertahap sesuai urutan yang sudah dibuat. Kita harus ingat jika hal rumit akan lebih mudah dikerjakan jika kita membagi kerumitan tersebut menjadi beberapa hal yang lebih sederhana dan memprosesnya secara bertahap. Dengan pengurutan dan pembagian ini maka otak dapat mengerjakan tugas yang banyak dan rumit tadi secara bertahap namun tetap pada jalur yang benar sesuai tujuan. Pengecekan waktu juga dilakukan secara manual sehingga menjadi kurang efektif terutama untuk tugas tugas yang cukup kompleks. Cara lain adalah dengan menggunakan perangkat lunak pada PC. Namun cara ini masih mempunyai kekurangan karna kita harus tetap berada pada PC tempat kita menyimpan task list tersebut untuk mengeceknya. Oleh karena itu dibuatlah sebuah aplikasi yang dapat menyediakan kemudahan pengelolaan daftar list / to-do-list seperti kemudahan menuliskan pada sebuah kertas yang dapat di akses lewat smartphone posel genggam kita.

B. Rumusan Masalah Manfaat

- a. Memaparkan pembuatan sistem aplikasi Android yang bisa berjalan di sistem operasi Android.
- b. Versi Android yang digunakan yaitu Android versi 3.0 (HoneyBomb) sampai Android 11.1(R).
- c. Aplikasi berjalan dalam kondisi offline maupun online

C. Tujuan Pembuatan Aplikasi Android

Tujuan dari penelitian ini adalah untuk merancang dan membuat aplikasi pengingat tugas/to do list dapat berjalan pada sistem operasi Android sehingga memberikan kemudahan bagi pemakainya.

D. Manfaat

- a. Mempermudah user dalam mengakses aplikasi karena menggunakan system operasi android
- b. Membantu user dalam membuat daftar list pengingat.

BAB II

LANDASAN TEORI

A. List View

ListView adalah salah satu widget yang digunakan untuk menampilkan data atau nilai dalam bentuk daftar/list, nilai atau data yang ditampilkan pada ListView tersebut didapat dari sebuah Array atau Database yang sudah ditentukan.

Jadi apa itu ListView? Sesungguhnya ListView adalah sebuah widget yang terdapat pada komponen UI (user interface) dari sebuah aplikasi. ListView adalah widget yang dipakai untuk menampilkan nilai-nilai dan data dalam bentuk list atau daftar. Definisi ini sesuai dengan nama ListView itu sendiri, yang terdiri dari 2 kata: list dan view. List berarti daftar, dan view berarti tampilan.

Item-item data yang ditampilkan dalam ListView biasanya dapat tersusun dalam urutan berjajar ke samping atau ke bawah. Intinya, tampilan ListView selalu dapat dengan mudah di-scroll oleh user aplikasi. Data-data pada ListView didapat dari sebuah query database atau array yang telah ditetapkan sebagai sebuah data model.

Untuk bisa bekerja dengan baik, ListView punya sebuah adapter. Adapter adalah sebuah class yang berfungsi untuk mengatur resource view, model dan susunan item di ListView. Resource view mengatur item yang terlihat dan tersembunyi di ListView dan adapter hampir selalu mempunyai jumlah resource view yang tetap, tidak berubah, sesuai dengan item-item yang muncul di layar aplikasi.

ListView masih cukup banyak digunakan oleh developer aplikasi, terutama jika aplikasi yang dibuat masih sederhana. Para app developer pemula juga sering kali menjadikan tugas membuat ListView Android Studio sebagai salah satu sistem belajar. Namun selain ListView, terdapat pula mekanisme penampikan data dari array dan database lainnya yang bernama RecyclerView yang lebih banyak digunakan oleh developer app profesional kekinian.

Akan tetapi, pemahaman dan kemampuan ListView adalah dasarnya yang harus Anda kuasai. Jika Anda tidak mengerti bagaimana cara membuat ListView, Anda bisa lebih sulit saat belajar menguasai cara membuat RecyclerView.

B. Floating Action Button

Floating Action Button (FAB) adalah salah satu komponen dari Material Desain. FAB merupakan sebuah tombol yang berbentuk lingkaran dan ditampilkan seperti melayang pada aplikasi android.

C. SQL Database

Database SQLite adalah solusi penyimpanan yang baik jika anda memiliki data terstruktur yang perlu diakses dan disimpan secara persisten serta sering ditelusuri dan diubah. Anda juga bisa menggunakan SQLite sebagai media penyimpanan utama untuk data aplikasi atau pengguna, atau anda juga bisa menggunakannya untuk proses caching serta menyediakan data yang diambil dari cloud.

Jika anda bisa menyatakan data berupa baris dan kolom, pertimbangkan untuk memakai database SQLite. Jika anda menggunakan database SQLite, yang dinyatakan sebagai objek SQLiteDatabase adalah semua interaksi dengan database adalah melalui instance dari kelas SQLiteOpenHelper yang akan mengeksekusi permintaan dan pengelolaan database. Aplikasi anda hanya boleh berinteraksi dengan SQLiteOpenHelper, yang akan kita bahas bersama-sama dibawah ini.

Singkatnya SQLite Database memiliki metode untuk membuat, menghapus, menjalankan perintah SQL, dan melakukan tugas manajemen database umum lainnya. seperti perintah CRUD (Create, Read, Update, Delete) data pada sebuah aplikasi catatan sederhana.

BAB III

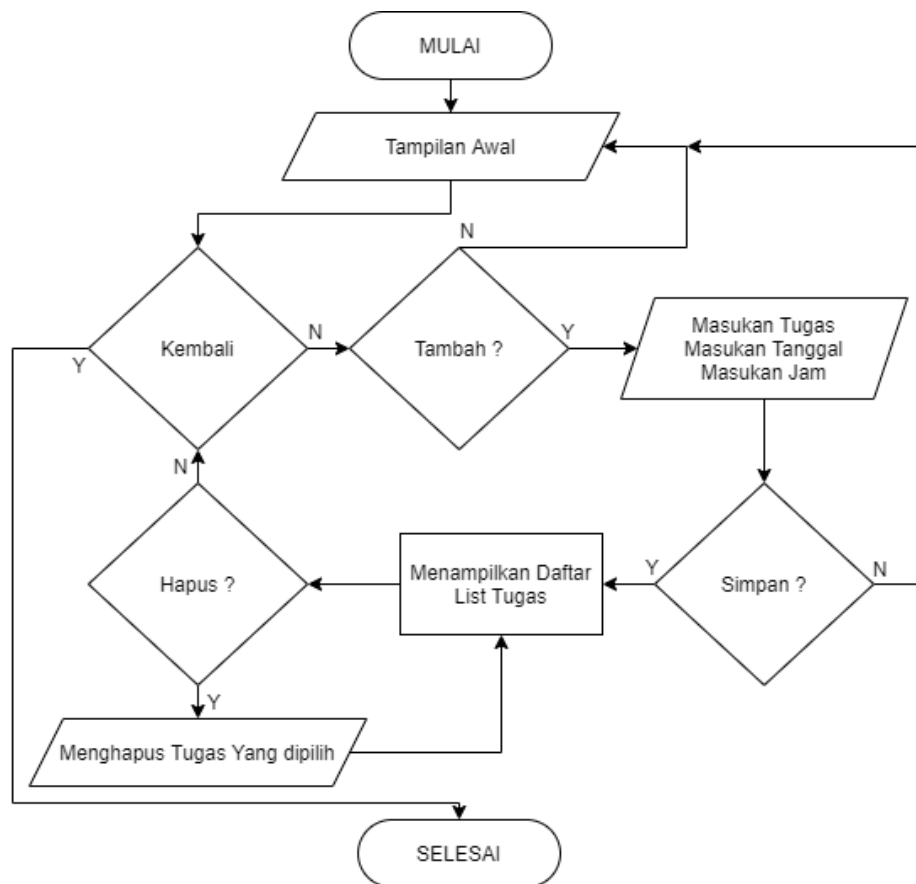
PERANCANGAN APLIKASI

A. Perancangan Sistem

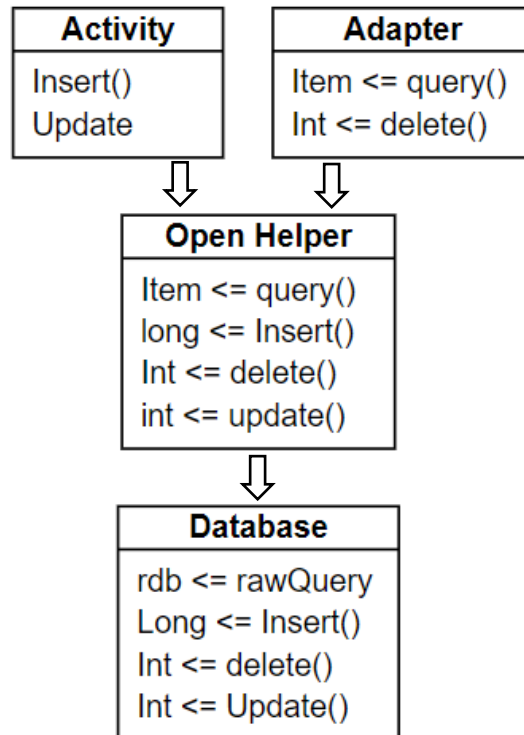
- Spesifikasi Minimum

Aplikasi Pengingat Tugas ini dapat berjalan dengan spek minimum Android versi 3.0 (HoneyBomb)

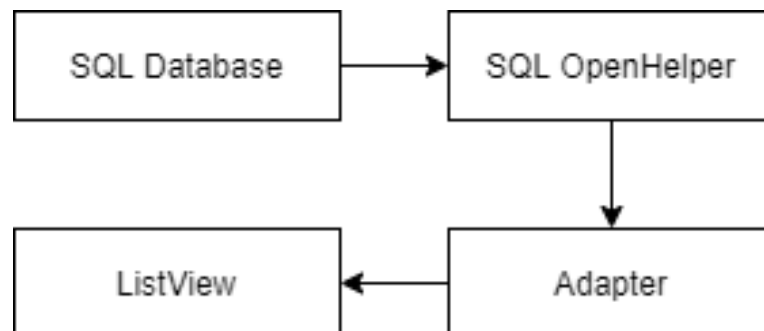
- Flowchart



- Class Diagram



- Entyty Diagram



B. Perancangan UI/UX

- Component

Component yang digunakan dalam aplikasi ini ada listview, floating button, text view

- Layout

Layout yang digunakan dalam aplikasi ini relative layout, dan linear layout

C. Coding

- activity_main.xml

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

    <ListView
        android:id="@+id/itemsList"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="6dp"
        android:layout_marginBottom="50dp"
        android:divider="@null"
        android:dividerHeight="0dp"
        android:paddingLeft="6dp"
        android:paddingRight="6dp"/>

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fab"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="15dp"
        android:backgroundTint="@color/purple_500"
        android:clickable="true"
        android:focusable="true"
        android:src="@drawable/ic_tambah"
        tools:ignore="UnusedAttribute" />

</RelativeLayout>
```

- daftar_todo.xml


```

<?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:ignore="ContentDescription,
UselessParent,UseCompoundDrawables,HardcodedText">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp">

        <ImageView
            android:id="@+id/delete"
            android:layout_width="36dp"
            android:layout_height="36dp"
            android:layout_alignParentEnd="true"
            android:layout_centerVertical="true"
            android:layout_marginEnd="16dp"
            android:background="@drawable/ic_hapus" />

        <LinearLayout
            android:id="@+id/titleRow"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            tools:ignore="UseCompoundDrawables">

                <ImageView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="16dp"
                    android:layout_marginEnd="16dp"
                    android:layout_marginBottom="8dp"
                    android:background="@drawable/ic_judul"
                    app:tint="@android:color/holo_blue_dark" />

                <TextView
                    android:id="@+id/title"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="16dp"
                    android:layout_marginBottom="8dp"
                    android:text="Judul"
                    android:textAllCaps="false"
                    android:textSize="18sp" />
            </LinearLayout>

            <LinearLayout
                android:id="@+id/dateRow"

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/titleRow"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="16dp"
            android:layout_marginBottom="8dp"
            android:background="@drawable/ic_tanggal" />

        <TextView
            android:id="@+id/dateTitle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="4dp"
            android:layout_marginBottom="8dp"
            android:text="Tanggal"
            android:textAllCaps="true"
            android:textSize="12sp" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/dateRow"
        android:orientation="horizontal"
        tools:ignore="UseCompoundDrawables">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginEnd="16dp"
            android:layout_marginBottom="16dp"
            android:background="@drawable/ic_waktu"
            tools:ignore="ContentDescription" />

        <TextView
            android:id="@+id/timeTitle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="4dp"
            android:layout_marginBottom="16dp"
            android:text="Waktu"
            android:textAllCaps="true"
            android:textSize="12sp" />

    </LinearLayout>
</RelativeLayout>
</RelativeLayout>

```

- custom_dialog_todo.xml

```
<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"
android:divider="@null"
android:dividerPadding="0dp"
android:orientation="vertical"
android:padding="10dp"
app:layout_constraintCircleRadius="8dp"
tools:ignore="UseCompoundDrawables,
ContentDescription,HardcodedText">
```

```
<LinearLayout
```

```
android:id="@+id/titleLayout"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal">
```

```
<ImageView
```

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_margin="16dp"
android:src="@drawable/ic_judul" />
```

```
<EditText
```

```
android:id="@+id/edit_title"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_marginTop="4dp"
android:hint="Tugas"
android:inputType="text"
tools:ignore="Autofill" />
```

```
</LinearLayout>
```

```
<LinearLayout
```

```
android:id="@+id/dateLayout"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_below="@+id/titleLayout"
android:orientation="horizontal">
```

```
<ImageView
```

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_margin="16dp"
android:background="@drawable/ic_tanggal"/>
```

```
<TextView
```

```
android:id="@+id/date"
android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:text="Tanggal"
        android:layout_marginTop="16dp"
        android:layout_marginBottom="16dp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/dateLayout"
        android:orientation="horizontal">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_margin="16dp"
            android:background="@drawable/ic_waktu" />

        <TextView
            android:id="@+id/time"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Waktu"
            android:layout_marginTop="16dp"
            android:layout_marginBottom="16dp" />
    </LinearLayout>
</RelativeLayout>

```

- DatabaseHelper

- `package` com.example.todolist;

```

import android.annotation.SuppressLint;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

import java.util.ArrayList;

public class DatabaseHelper extends SQLiteOpenHelper {

    private static final String TAG = "DatabaseHelper";

    private static final String TABLE_NAME = "ToDo_Table";
    private static final String COL1 = "ID";
    private static final String COL2 = "Name";
    private static final String COL3 = "Date";
    private static final String COL4 = "Time";

    public DatabaseHelper(Context context) {
        super(context, TABLE_NAME, null, 1);
    }

```

```

    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTable = "CREATE TABLE " + TABLE_NAME +
            "(" + COL1 + " integer primary key, "
            + COL2 + " TEXT, " + COL3 + " DATE, "
            + COL4 + " TIME" + ")";
        Log.d(TAG, "Creating table " + createTable);
        db.execSQL(createTable);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion,
        int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }

    //Memasukkan data ke database
    public boolean insertData(String item, String date,
        String time) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL2, item);
        contentValues.put(COL3, date);
        contentValues.put(COL4, time);
        Log.d(TAG, "insertData: Inserting " + item + " to "
            + TABLE_NAME);
        long result = db.insert(TABLE_NAME, null,
            contentValues);
        db.close();
        return result != -1;
    }

    //Menghapus data dari database
    void deleteData(int id) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_NAME, COL1 + "=" + id, null);
    }

    //Memuat semua data ke Listview
    public ArrayList<ModelData> getAllData() {
        ArrayList<ModelData> arrayList = new ArrayList<>();
        SQLiteDatabase db = this.getReadableDatabase();
        String query = "SELECT * FROM " + TABLE_NAME;
        @SuppressWarnings("Recycle")
        Cursor cursor = db.rawQuery(query, null);

        while (cursor.moveToNext()) {
            int id = cursor.getInt(0);
            String title = cursor.getString(1);
            String date = cursor.getString(2);
            String time = cursor.getString(3);

```

```

        ModelData modelData = new ModelData(id, title,
date, time);
        arrayList.add(modelData);
    }
    db.close();
    return arrayList;
}
}

```

- ItemAdapter

```

package com.example.todolist;

import android.annotation.SuppressLint;
import android.content.Context;
import android.view.Gravity;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import java.util.ArrayList;

public class ItemAdapter extends BaseAdapter {

    private Context context;
    private ArrayList<ModelData> arrayList;

    public ItemAdapter(Context context, ArrayList<ModelData>
arrayList) {
        super();
        this.context = context;
        this.arrayList = arrayList;
    }

    @Override
    public int getCount() {
        return this.arrayList.size();
    }

    @Override
    public Object getItem(int position) {
        return arrayList.get(position);
    }

    @Override
    public long getItemId(int position) {
        return position;
    }

    @SuppressLint({"ViewHolder", "InflateParams"})

```

```

        @Override
        public View getView(int position, View convertView,
final ViewGroup parent) {
            LayoutInflater inflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
            assert inflater != null;
            convertView =
inflater.inflate(R.layout.daftar_todo, null);
            TextView titleTextView =
convertView.findViewById(R.id.title);
            TextView dateTextView =
convertView.findViewById(R.id.dateTitle);
            TextView timeTextView =
convertView.findViewById(R.id.timeTitle);
            final ImageView delImageView =
convertView.findViewById(R.id.delete);
            delImageView.setTag(position);

            //Menghapus tugas dari database saat icon hapus di
klik
            delImageView.setOnClickListener(new
View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    final int pos = (int) v.getTag();
                    deleteItem(pos);
                }
            });

            ModelData modelData = arrayList.get(position);
            titleTextView.setText(modelData.getTitle());
            dateTextView.setText(modelData.getDate());
            timeTextView.setText(modelData.getTime());
            return convertView;
        }

        //Menghapus tugas dari Listview
        private void deleteItem(int position) {
            deleteItemFromDb(arrayList.get(position).getId());
            arrayList.remove(position);
            notifyDataSetChanged();
        }

        //Menghapus tugas dari database
        private void deleteItemFromDb(int id) {
            DatabaseHelper databaseHelper = new
DatabaseHelper(context);
            try {
                databaseHelper.deleteData(id);
                toastMsg("Tugas di hapus");
            } catch (Exception e) {
                e.printStackTrace();
                toastMsg("Oppss.. ada kesalahan saat
menghapus");
            }
        }

```

```

    }
}

//Metode pesan toast
private void toastMsg(String msg) {
    Toast t = Toast.makeText(context, msg,
Toast.LENGTH_SHORT);
    t.setGravity(Gravity.CENTER,0,0);
    t.show();
}
}

```

- MainActivity

```

package com.example.todolist

import android.annotation.SuppressLint
import android.app.*
import android.app.DatePickerDialog.OnDateSetListener
import android.app.TimePickerDialog.OnTimeSetListener
import android.content.Context
import android.content.Intent
import android.os.Bundle
import android.util.Log
import android.view.Gravity
import android.view.animation.AlphaAnimation
import android.widget.*
import androidx.appcompat.app.AlertDialog
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.NotificationCompat
import com.google.android.material.floatingactionbutton.FloatingActionButton
import java.text.DateFormatSymbols
import java.text.SimpleDateFormat
import java.util.*

class MainActivity : AppCompatActivity() {
    private var dbHelper: DatabaseHelper? = null
    private var itemsListView: ListView? = null
    private var fab: FloatingActionButton? = null
    private val buttonClick = AlphaAnimation(1f, 0.3f)
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        dbHelper = DatabaseHelper(this)
        fab = findViewById(R.id.fab)
        itemsListView = findViewById(R.id.itemsList)
        populateListView()
        onFabClick()
        hideFab()
    }
}

```



```

//Mengatur notifikasi
private fun scheduleNotification(
    notification: Notification,
    delay: Long
) {
    val notificationIntent = Intent(this,
Notifikasi::class.java)

    notificationIntent.putExtra(Notifikasi.NOTIFICATION_ID, 1)
    notificationIntent.putExtra(Notifikasi.NOTIFICATION,
notification)
    val pendingIntent = PendingIntent.getBroadcast(
        this, 0,
        notificationIntent,
        PendingIntent.FLAG_UPDATE_CURRENT
    )
    val alarmManager = LayoutInflater.context
        .getSystemService(Context.ALARM_SERVICE) as
AlarmManager
    alarmManager?.set(AlarmManager.RTC_WAKEUP, delay,
pendingIntent)
}

private fun getNotification(content: String):
Notification {

    //Saat notifikasi di klik di arahkan ke MainActivity
    val intent = Intent(this, MainActivity::class.java)
    intent.flags = Intent.FLAG_ACTIVITY_NEW_TASK or
Intent.FLAG_ACTIVITY_CLEAR_TASK
    val pendingIntent = PendingIntent.getActivity(this,
0, intent, 0)
    val builder = NotificationCompat.Builder(
        LayoutInflater.context,
        default_notification_channel_id
    )
    builder.setContentTitle("Pengingat")
    builder.setContentText(content)
    builder.setContentIntent(pendingIntent)
    builder.setAutoCancel(true)
    builder.setSmallIcon(R.drawable.ic_judul)
    builder.setDefaults(Notification.DEFAULT_LIGHTS or
Notification.DEFAULT_SOUND)
    builder.setChannelId(NOTIFICATION_CHANNEL_ID)
    builder.priority = NotificationCompat.PRIORITY_HIGH
    return builder.build()
}

//Memasukkan data ke database
private fun insertDataToDb(
    title: String,
    date: String,
    time: String
) {

```

```

        val insertData = databaseHelper!!.insertData(title,
date, time)
        if (insertData) {
            try {
                populateListView()
                toastMsg("Tugas di tambahkan")
            } catch (e: Exception) {
                e.printStackTrace()
            }
        } else toastMsg("Opps.. terjadi kesalahan saat
menyimpan!")
    }

    //Mengambil seluruh data dari database ke listview
    private fun populateListView() {
        try {
            val items = databaseHelper!!.allData
            val itemsAdopter = ItemAdapter(this, items)
            itemsListView!!.adapter = itemsAdopter
            itemsAdopter.notifyDataSetChanged()
        } catch (e: Exception) {
            e.printStackTrace()
        }
    }

    //Menyembunyikan tombol floating tambah saat listview di
scroll
    private fun hideFab() {
        itemsListView!!.setOnScrollListener(object :
AbsListView.OnScrollListener {
            override fun onScrollStateChanged(
                view: AbsListView,
                scrollState: Int
            ) {
                if (scrollState ==
AbsListView.OnScrollListener.SCROLL_STATE_IDLE) {
                    fab!!.show()
                } else {
                    fab!!.hide()
                }
            }
        })

        override fun onScroll(
            view: AbsListView,
            firstVisibleItem: Int,
            visibleItemCount: Int,
            totalItemCount: Int
        ) {
        }
    })
}

    private fun onFabClick() {
        try {

```

```

        fab!!.setOnClickListener { v ->
            v.startAnimation(buttonClick)
            showAddDialog()
        }
    } catch (e: Exception) {
        e.printStackTrace()
    }
}

//Implementasi klik dari tombol tambah
@SuppressLint("SimpleDateFormat")
private fun showAddDialog() {
    val dialogBuilder =
        AlertDialog.Builder(layoutInflater.context)
    val inflater = this.layoutInflater
    @SuppressLint("InflateParams") val dialogView =
        inflater.inflate(R.layout.custom_dialog_todo,
null)
    dialogBuilder.setView(dialogView)
    val judul =
dialogView.findViewById<EditText>(R.id.edit_title)
    val tanggal =
dialogView.findViewById<TextView>(R.id.date)
    val waktu =
dialogView.findViewById<TextView>(R.id.time)
    val date = System.currentTimeMillis()
    val dateSdf = SimpleDateFormat("d MMMM")
    val dateString = dateSdf.format(date)
    tanggal.text = dateString
    val timeSdf = SimpleDateFormat("hh : mm a")
    val timeString = timeSdf.format(date)
    waktu.text = timeString
    val cal = Calendar.getInstance()
    cal.timeInMillis = System.currentTimeMillis()

    //Set tanggal
    tanggal.setOnClickListener {
        val datePickerDialog =
            DatePickerDialog(
                layoutInflater.context,
                OnDateSetListener { view, year,
monthOfYear, dayOfMonth ->
                    val newMonth = getMonth(monthOfYear
+ 1)
                    tanggal.text = "$dayOfMonth
$newMonth"
                    cal[Calendar.YEAR] = year
                    cal[Calendar.MONTH] = monthOfYear
                    cal[Calendar.DAY_OF_MONTH] =
dayOfMonth
                },
                cal[Calendar.YEAR],
                cal[Calendar.MONTH],
                cal[Calendar.DAY_OF_MONTH]
            )
    }
}

```

```

        )
        datePickerDialog.show()
        datePickerDialog.datePicker.minDate = date
    }

    //Set waktu
    waktu.setOnClickListener {
        val timePickerDialog =
            TimePickerDialog(
                LayoutInflater.context,
                OnTimeSetListener { view, hourOfDay,
minute ->
                var hourOfDay = hourOfDay
                val time: String
                @SuppressWarnings("DefaultLocale") val
minTime =
                String.format("%02d", minute)
                if (hourOfDay >= 0 && hourOfDay <
12) {
                    time = "$hourOfDay : $minTime
AM"

                } else {
                    if (hourOfDay != 12) {
                        hourOfDay = hourOfDay - 12
                    }
                    time = "$hourOfDay : $minTime
PM"

                }
                waktu.text = time
                cal[Calendar.HOUR] = hourOfDay
                cal[Calendar.MINUTE] = minute
                cal[Calendar.SECOND] = 0
                Log.d(
                    TAG,
                    "onTimeSet: Time has been set
successfully"
                )
            },
            cal[Calendar.HOUR],
            cal[Calendar.MINUTE],
            false
        )
        timePickerDialog.show()
    }
    dialogBuilder.setTitle("Buat tugas baru")
    dialogBuilder.setPositiveButton(
        "Tambah"
    ) { dialog, whichButton ->
        val title = judul.text.toString()
        val date = tanggal.text.toString()
        val time = waktu.text.toString()
        if (title.length != 0) {
            try {
                insertDataToDb(title, date, time)
            }
        }
    }
}

```

```

scheduleNotification(getNotification(title),
cal.timeInMillis)
    } catch (e: Exception) {
        e.printStackTrace()
    }
    } else {
        toastMsg("Oops, Gak bisa kosong tugas nya.")
    }
}
dialogBuilder.setNegativeButton(
    "Cancel"
) { dialog, whichButton -> dialog.cancel() }
val b = dialogBuilder.create()
b.show()
}

//Metode pesan toast
private fun toastMsg(msg: String) {
    val t = Toast.makeText(this, msg,
Toast.LENGTH_SHORT)
    t.setGravity(Gravity.CENTER, 0, 0)
    t.show()
}

//Mengkonversi bulan dari huruf menjadi angka
private fun getMonth(month: Int): String {
    return DateFormatSymbols().months[month - 1]
}

companion object {
    const val NOTIFICATION_CHANNEL_ID = "10001"
    private const val default_notification_channel_id =
"default"
    private const val TAG = "MainActivity"
}
}

```

- ModelData

```

package com.example.todolist;

public class ModelData {

    int id;
    private String title;
    private String date;
    private String time;

    ModelData(int id, String title, String date, String
time) {
        this.id = id;
        this.title = title;
        this.date = date;
    }
}

```

```

        this.time = time;
    }

    int getId() {
        return id;
    }

    String getTitle() {
        return title;
    }

    String getDate() {
        return date;
    }

    String getTime() {
        return time;
    }
}

```

- Notifikasi

```

package com.example.todolist;

import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;

public class Notifikasi extends BroadcastReceiver {

    public static String NOTIFICATION_ID = "notification-id";
    public static String NOTIFICATION = "notification";

    public void onReceive(Context context, Intent intent) {
        NotificationManager notificationManager =
            (NotificationManager)
            context.getSystemService(Context.NOTIFICATION_SERVICE);
        Notification notification =
            intent.getParcelableExtra(NOTIFICATION);
        if (android.os.Build.VERSION.SDK_INT >=
            android.os.Build.VERSION_CODES.O) {
            int importance =
                NotificationManager.IMPORTANCE_HIGH;
            NotificationChannel notificationChannel = new
                NotificationChannel(MainActivity.NOTIFICATION_CHANNEL_ID,
                    "Pengingat", importance);
            assert notificationManager != null;

            notificationManager.createNotificationChannel(notificationCh

```

```

        annel);
    }
    int id = intent.getIntExtra(NOTIFICATION_ID, 0);
    if (notificationManager != null) {
        notificationManager.notify(id, notification);
    }
}
}

```

- AndroidManifest

```

<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.todolist">

    <uses-permission
android:name="android.permission.VIBRATE" />
    <uses-permission
android:name="com.google.android.c2dm.permission.RECEIVE" />
    <uses-permission
android:name="android.permission.WAKE_LOCK" />
    <uses-permission
android:name="android.permission.RECEIVE_BOOT_COMPLETED" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Todolist">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action
android:name="android.intent.action.MAIN" />

                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <receiver android:name=".Notifikasi"/>
    </application>

</manifest>

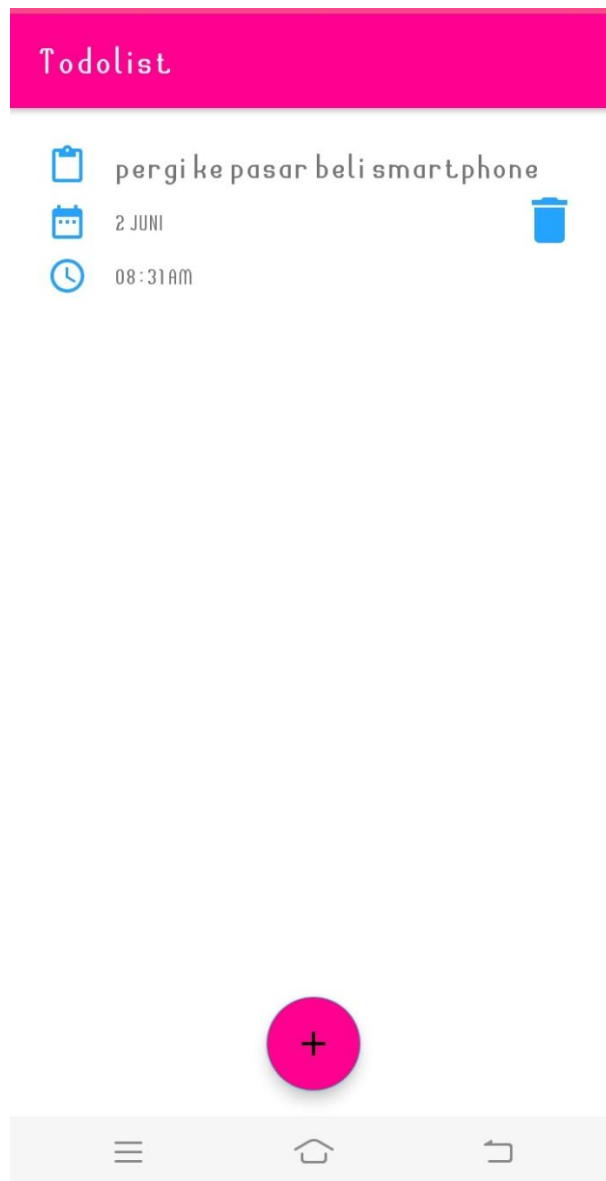
```

D. Testing



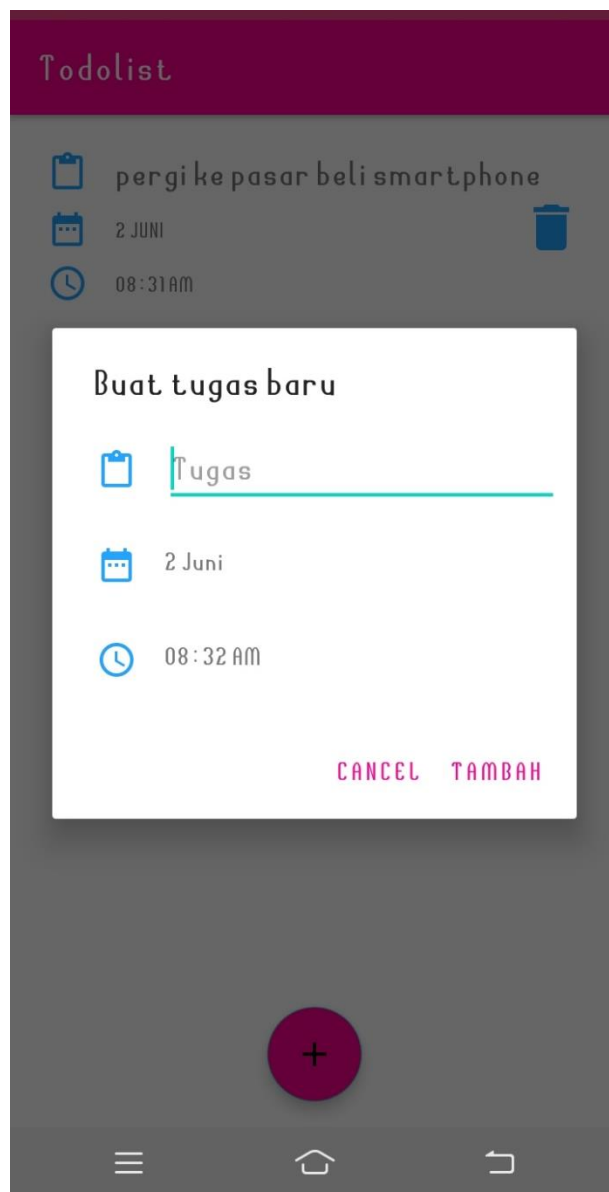
Gambar 1.1 Tampilan Awal Aplikasi

Tampilan awal aplikasi, Pada bar atas telah di beri judul aplikasi yang bernama todolist, disini masih belum menampilkan daftar tugas karena memang belum ada tugas yang diinput. Tampilan yang sederhana dengan memanfaatkan list view dan float button, disini float button nya berada di tengah. dengan fungsi untuk menambahkan tugas.



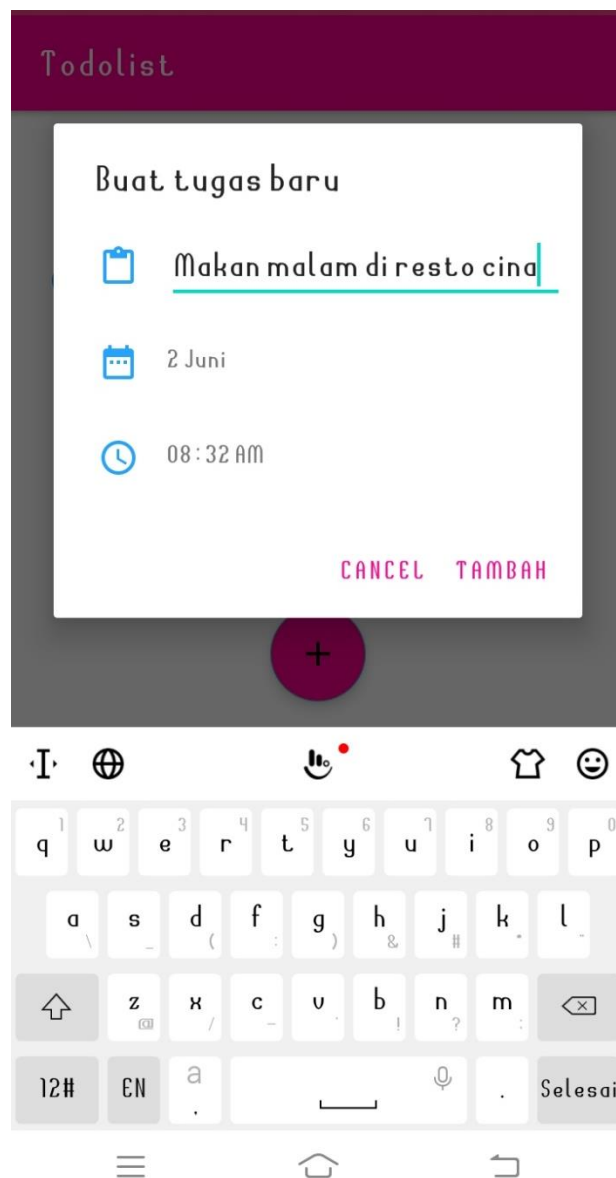
Gambar 1.2 Tampilan List View Saat Sudah Ada Tugas

Tampilan list view saat sudah ada tugas, disini tugas yang sebelum nya diinput akan tertampil, tampilan akan berupa list sesuai dengan yang kita pakai adalah list view. Pada tampilan list ini di sertakan tgl dan jam yang telah kita input.



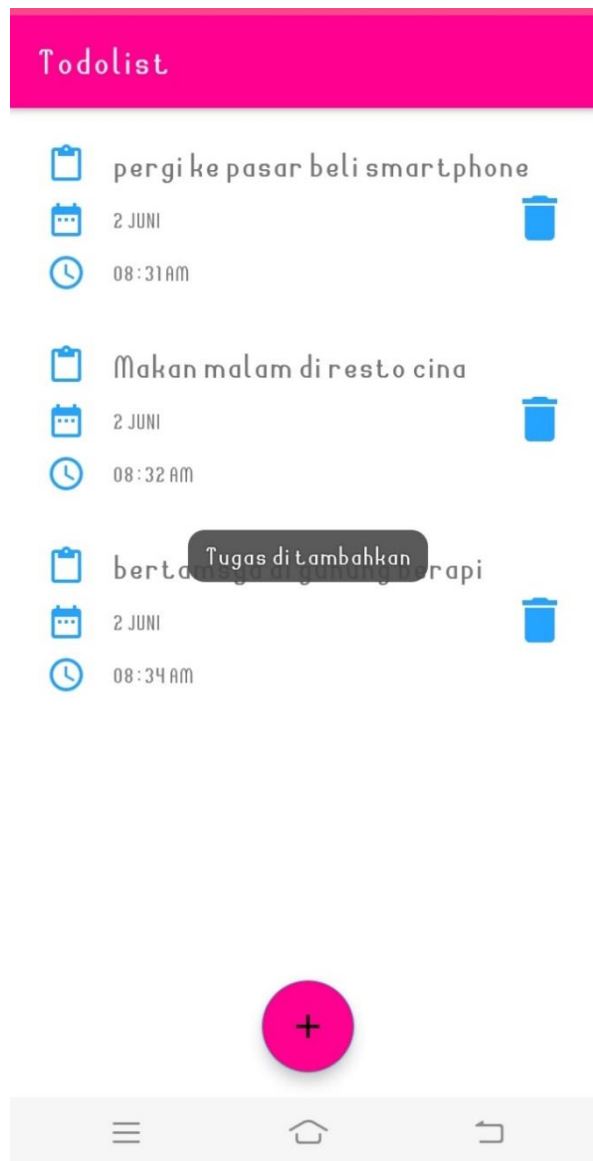
Gambar 1.3 Tampilan Layout Input Tugas

Tampilan layout input tugas, Tugas, disini kita akan menginput tugas apa yang akan kita input serta tanggal dan jam nya.



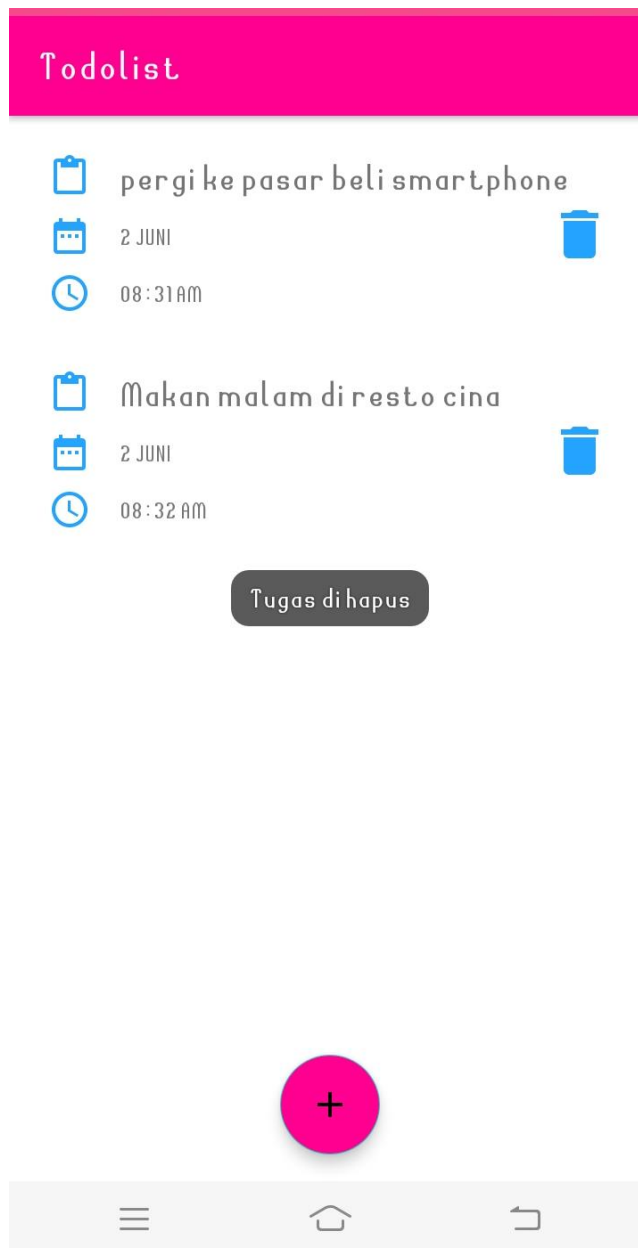
Gambar 1.4 Tampilan Saat Menginput Tugas

Tampilan saat menginput tugas, disini kita memasukan data tugas yang nanti nya akan di simpan didatabase yang akan tertampil di tampilan list view, seperti contoh di atas memasukan tugas “Makan malam di resto cina”. lalu kita akan memilih action apakah kita akan tambah tugas ataupun membatalkan nya, jika memilih action tambah maka tugas akan tersimpan dan nanti nya akan tertampil namun jika cancel maka akan kembali ke menu tampilan list view.



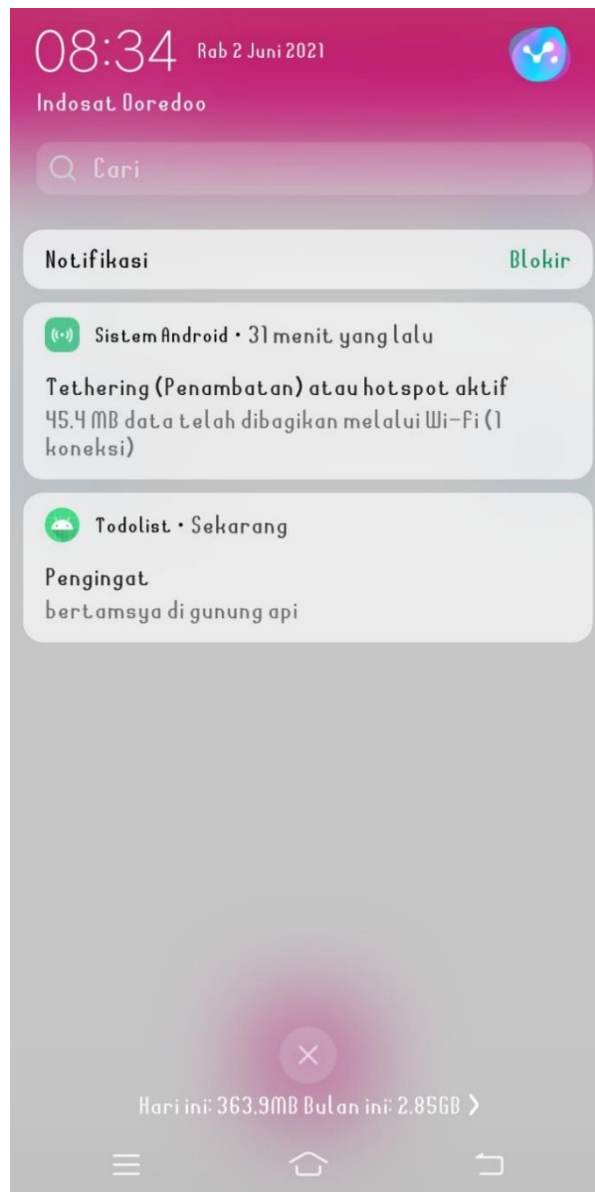
Gambar 1.5 Tampilan Layout Setelah Input Tugas

Tampilan layout setelah input tugas, disini akan ada toast yang memberi tau bahwa tugas telah di tambahkan dan sudah tertampil di list view.



Gambar 1.6 Tampilan Layout Setelah Hapus Tugas

Tampilan layout hapus tugas, disini kita akan menghapus tugas dan akan ada toast yang memberi tau bahwa tugas telah di hapus dan tugas sudah tidak ada lagi tampilan di list view.



Gambar 1.7 Tampilan Notification

Tampilan notification saat tugas sudah berada pada waktu yang sudah di input maka akan ada notifikasi yang muncul bahwa tugas harus di selesaikan.

E. Debuging

Penambahan Notifikasi, yang awal nya tidak ada tambahkan kode pada android manifest.

namun sebelum nya sudah buat file kotlin notification

```
<activity android:name=".MainActivity">
  <intent-filter>
    <action android:name="android.intent.action.MAIN" />

    <category android:name="android.intent.category.LAUNCHER" />
  </intent-filter>
</activity>
<receiver android:name=".Notifikasi"/>
```

selanjut nya ada penambahan untuk mengurutan tugas namun karena tidak berhasil maka tidak jadi di tambahkan.

Ada juga penambahan menu namun aplikasi crack sehingga tidak jadi di tambahkan.

BAB IV

KESIMPULAN DAN SARAN

A. Kesimpulan

- c. Aplikasi pengingat tugas/to do list dapat berjalan pada sistem operasi Android. yang mana spesifikasi minimum nya yaitu Android versi 3.0 (HoneyBomb) sampai Android 11.1(R). aplikasi berjalan tanpa menggunakan koneksi internet (Offline). di arapkan aplikasi dapat mempermudah user dalam mengakses aplikasi karena menggunakan system operasi android dan membantu user dalam membuat daftar list pengingat.

B. Saran

Aplikasi perlu adanya pengembangan lebih lanjut karena masih sangat sederhana. Tampilan UI nya juga butuh sentuhan dari warna hingga permodelan nya.

LAMPIRAN

A. Referensi

Modul praktikum 4

Modul Praktikum 5

Modul Praktikum 10

<https://appkey.id/pembuatan-aplikasi/aplikasi-android/listview-android-studio/>

<https://medium.com/easyread/konsep-dasar-database-sqlite-pada-android-e9445558f494>

<https://www.gwnbs.com/2020/06/aplikasi-pengingat-tugas-to-do-reminder-android.html>

