# LAPORAN TUGAS BESAR

**PEMROGRAMAN MOBILE**

**APLIKASI PEMBUAT RESEP MINUMAN**

**DISUSUN OLEH :**

**NANDA HENDRWAN**

**1809075044**



**PROGRAM STUDI TEKNIK ELEKTRO**

**FAKULTAS TEKNIK**

**UNIVERSITAS MULAWARMAN**

**SAMARINDA**

**2021**

**BAB I**

**PENDAHULUAN**

* 1. **Latar Belakang**

Seperti yang diketahui dewasa ini banyak sekali peranan dan fungsi Teknologi Informasi dalam segala bidang. Fungsi dan peranan Teknologi *mobile* sebagai contoh, tidak hanya digunakan sebagai alat komunikasi saja, tetapi juga sebagai alat untuk memudahkan pengguna dalam kehidupan sehari hari. Para pelaku bisnis pun mulai memanfaatkan kemajuan teknologi saat ini untuk keperluan manajerial perusahaan mereka. Salah satu pelaku bisnis yang ada adalah pelaku bisnis *restaurant*. Mengingat persaingan bisnis *restaurant* saat ini sangatlah ketat, untuk itu dibutuhkan manajerial strategi yang tepat guna. Oleh karena itu pemilik *restaurant* ataupun para pelaku bisnis dalam restaurant dapat memanfaatkan adanya peranan teknologi untuk membantu mempercepat proses pengelolaan. Pemesanan menu merupakan aspek yang sangat berpengaruh terhadap ekspektasi pengunjung terhadap suatu restaurant, untuk selanjutnya disebut sebagai proses penjualan. Ketika penjualanmengalami peningkatan pendukung yang sama pentingnya adalahketersediaan bahan makanan atau minuman pada *restaurant*. Dalam pengelolaan menu makanan dan minuman terdapat penentubahan makanan yang digunakan atau sering disebut sebagai resep menu.Melalui resep menu tersebut akan didapatkan kebutuhan bahan makanan atauminuman per-porsinya. Dengan menggunakan otomatisasi pengurangan bahan sesuai gramasinya melalui proses penjualan menu makanan, tentunya dapat dianalisir ketika bahan makanan dan minuman mengalami kekurangan.

* 1. **Rumusan Masalah**

1. Bagaimana cara membangun aplikasi manajemen inventory bahan dan penjuakan pada multi-unit restoran.
2. Bagaimana dampak yang dirasakan setelah adanya penelitian pembuatan aplikasi pembuat resep minuman.
3. Bagaimana cara mengintergrasikan aplikasi pembuatan resep minuman
   1. **Tujuan Pembuatan Aplikasi Android**

Mengetahui cara pembutan resep minuman dengan menggunakan sebuah aplikasi android studio dan mempermudah nya

* 1. **Manfaat**

1. Mempermudah user dalam mengakses aplikasi karena menggunakan system operasi android
2. Membantu user dalam membuat daftar list pengingat.

**BAB II**

**LANDASAN TEORI**

1. 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 penampilan data dari array dan database lainnya yang bernama RecyclerView yang lebih banyak digunakan oleh developer app profesiona 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.

1. 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.

1. 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.

Singktnya 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**

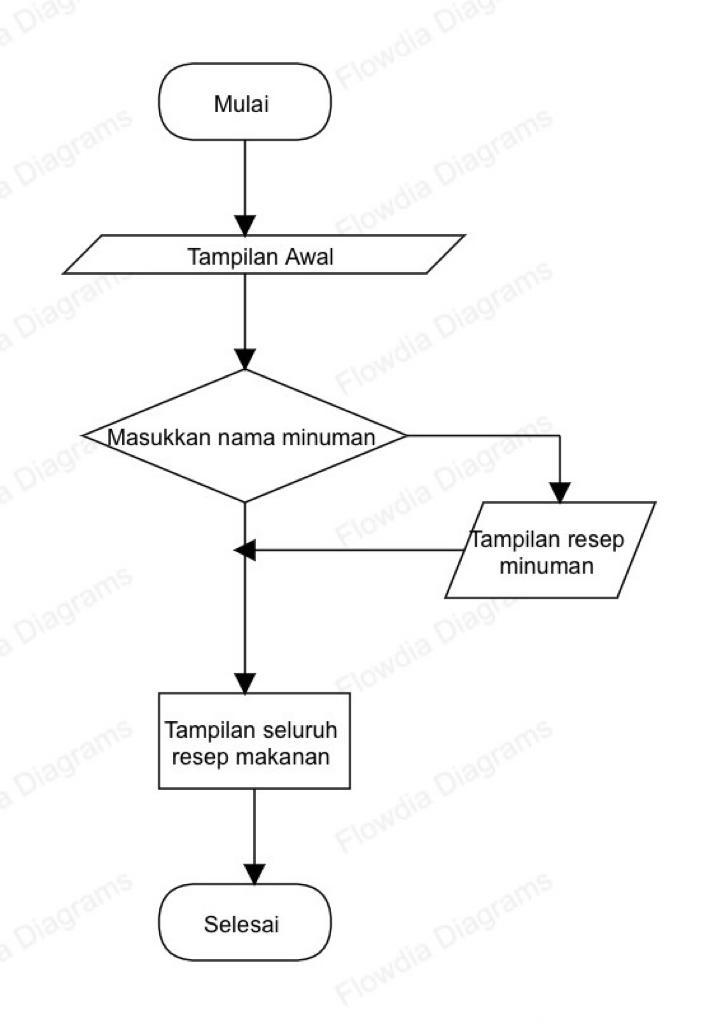
**PERANCANGAN APLIKASI**

* 1. **Perancangan Sistem**
* Spesifikasi Minimum

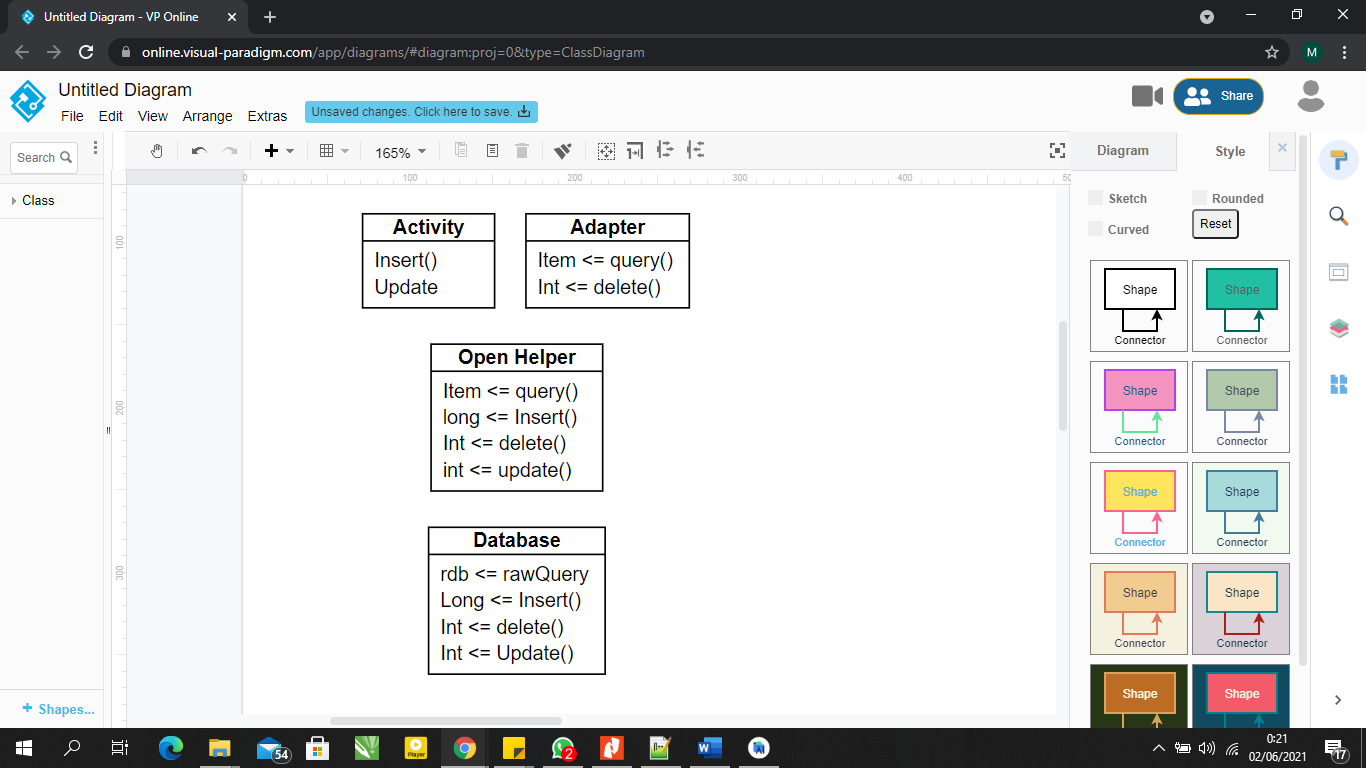
Aplikasi Pembuat resep minuman

Lolipop 5.0

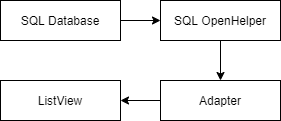
* Flowchart



* Class Diagram



* Entyty Diagram



* 1. **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

* 1. **Coding**
* AndroidManifest.xml
* <?xml version="1.0" encoding="utf-8"?>  
  <manifest xmlns:android="http://schemas.android.com/apk/res/android"  
   xmlns:tools="http://schemas.android.com/tools"  
   package="com.azhar.drinkrecipe">  
    
   <uses-permission android:name="android.permission.INTERNET" />  
    
   <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/AppTheme"  
   android:usesCleartextTraffic="true"  
   tools:targetApi="q">  
   <activity android:name=".activities.RecipeDrinkActivity" />  
   <activity android:name=".activities.ListDrinkActivity" />  
   <activity android:name=".activities.CategoriesActivity">  
   <intent-filter>  
   <action android:name="android.intent.action.MAIN" />  
    
   <category android:name="android.intent.category.LAUNCHER" />  
   </intent-filter>  
   </activity>  
   </application>  
    
  </manifest>
* CategoriesActivity
* package com.azhar.drinkrecipe.activities  
    
  import android.app.ProgressDialog  
  import android.os.Bundle  
  import android.widget.Toast  
  import androidx.appcompat.app.AppCompatActivity  
  import androidx.recyclerview.widget.LinearLayoutManager  
  import com.androidnetworking.AndroidNetworking  
  import com.androidnetworking.common.Priority  
  import com.androidnetworking.error.ANError  
  import com.androidnetworking.interfaces.JSONObjectRequestListener  
  import com.azhar.drinkrecipe.adapter.CategoriesAdapter  
  import com.azhar.drinkrecipe.databinding.ActivityCategoriesBinding  
  import com.azhar.drinkrecipe.model.ModelCategories  
  import com.azhar.drinkrecipe.networking.ApiEndpoint  
  import org.json.JSONException  
  import org.json.JSONObject  
  import java.util.\*  
    
  class CategoriesActivity : AppCompatActivity() {  
    
   private val binding: ActivityCategoriesBinding {  
   ActivityCategoriesBinding.inflate(layoutInflater)  
   }  
    
   var modelCategories: MutableList<ModelCategories> = ArrayList()  
   var categoriesAdapter: CategoriesAdapter? = null  
   var progressDialog: ProgressDialog? = null  
    
   override fun onCreate(savedInstanceState: Bundle?) {  
   super.onCreate(savedInstanceState)  
   setContentView(binding.*root*)  
    
   progressDialog = ProgressDialog(this)  
   progressDialog?.setTitle("Mohon Tunggu...")  
   progressDialog?.setCancelable(false)  
   progressDialog?.setMessage("Sedang menampilkan kategori")  
    
   categoriesAdapter = CategoriesAdapter(modelCategories, this@CategoriesActivity)  
   binding.rvCategories.*layoutManager* = LinearLayoutManager(this)  
   binding.rvCategories.*adapter* = categoriesAdapter  
   binding.rvCategories.setHasFixedSize(true)  
    
   //get data categories  
   getCategories()  
   }  
    
   private fun getCategories() {  
   progressDialog?.show()  
   AndroidNetworking.get(ApiEndpoint.BASEURL + ApiEndpoint.URL\_CATEGORIES)  
   .setPriority(Priority.*MEDIUM*)  
   .build()  
   .getAsJSONObject(object : JSONObjectRequestListener {  
   override fun onResponse(response: JSONObject) {  
   progressDialog?.dismiss()  
   try {  
   val jsonArray = response.getJSONArray("drinks")  
   for (i in 0 *until* jsonArray.length()) {  
   val jsonObject = jsonArray.getJSONObject(i)  
   val dataModel = ModelCategories()  
   dataModel.strCategory = jsonObject.getString("strCategory")  
   modelCategories.add(dataModel)  
   }  
   categoriesAdapter?.notifyDataSetChanged()  
   } catch (e: JSONException) {  
   Toast.makeText(this@CategoriesActivity,  
   "Oops, gagal menampilkan kategori.", Toast.*LENGTH\_SHORT*).show()  
   }  
   }  
    
   override fun onError(anError: ANError) {  
   progressDialog?.dismiss()  
   Toast.makeText(this@CategoriesActivity,  
   "Oops! Sepertinya ada masalah dengan koneksi internet kamu.", Toast.*LENGTH\_SHORT*).show()  
   }  
   })  
   }  
    
  }
* ListDrinkActivity
* package com.azhar.drinkrecipe.activities  
    
  import android.app.ProgressDialog  
  import android.os.Bundle  
  import android.view.MenuItem  
  import android.widget.Toast  
  import androidx.appcompat.app.AppCompatActivity  
  import androidx.recyclerview.widget.StaggeredGridLayoutManager  
  import com.androidnetworking.AndroidNetworking  
  import com.androidnetworking.common.Priority  
  import com.androidnetworking.error.ANError  
  import com.androidnetworking.interfaces.JSONObjectRequestListener  
  import com.azhar.drinkrecipe.R  
  import com.azhar.drinkrecipe.adapter.DrinkAdapter  
  import com.azhar.drinkrecipe.databinding.ActivityListDrinkBinding  
  import com.azhar.drinkrecipe.model.ModelCategories  
  import com.azhar.drinkrecipe.model.ModelDrink  
  import com.azhar.drinkrecipe.networking.ApiEndpoint  
  import org.json.JSONException  
  import org.json.JSONObject  
  import java.util.\*  
    
  class ListDrinkActivity : AppCompatActivity() {  
    
   private val binding: ActivityListDrinkBinding {  
   ActivityListDrinkBinding.inflate(layoutInflater)  
   }  
    
   companion object {  
   const val LIST\_DRINK = "CATEGORIES"  
   }  
    
   var modelDrink: MutableList<ModelDrink> = ArrayList()  
   var modelCategories: ModelCategories? = null  
   var drinkAdapter: DrinkAdapter? = null  
   var progressDialog: ProgressDialog? = null  
   var strCategory: String? = null  
    
   override fun onCreate(savedInstanceState: Bundle?) {  
   super.onCreate(savedInstanceState)  
   setContentView(binding.*root*)  
    
   progressDialog = ProgressDialog(this)  
   progressDialog?.setTitle("Mohon Tunggu...")  
   progressDialog?.setCancelable(false)  
   progressDialog?.setMessage("Sedang menampilkan minuman")  
    
   setSupportActionBar(binding.toolbar)  
   *assert*(*supportActionBar* != null)  
   *supportActionBar*?.setDisplayHomeAsUpEnabled(true)  
    
   //get data intent  
   modelCategories = *intent*.getSerializableExtra(LIST\_DRINK) as ModelCategories  
   if (modelCategories != null) {  
   strCategory = modelCategories?.strCategory  
    
   binding.tvCategories.*text* = strCategory  
    
   drinkAdapter = DrinkAdapter(modelDrink, this@ListDrinkActivity)  
   binding.rvListDrink.*layoutManager* = StaggeredGridLayoutManager(3, StaggeredGridLayoutManager.*VERTICAL*)  
   binding.rvListDrink.*adapter* = drinkAdapter  
   binding.rvListDrink.setHasFixedSize(true)  
    
   //get data drink  
   getListDrink(strCategory)  
   }  
   }  
    
   private fun getListDrink(strCategory: String?) {  
   progressDialog?.show()  
   AndroidNetworking.get(ApiEndpoint.BASEURL + ApiEndpoint.URL\_FILTER)  
   .addPathParameter("strCategory", strCategory)  
   .setPriority(Priority.*MEDIUM*)  
   .build()  
   .getAsJSONObject(object : JSONObjectRequestListener {  
   override fun onResponse(response: JSONObject) {  
   progressDialog?.dismiss()  
   try {  
   val jsonArray = response.getJSONArray("drinks")  
   for (i in 0 *until* jsonArray.length()) {  
   val jsonObject = jsonArray.getJSONObject(i)  
   val dataModel = ModelDrink()  
   dataModel.idDrink = jsonObject.getString("idDrink")  
   dataModel.strDrink = jsonObject.getString("strDrink")  
   dataModel.strDrinkThumb = jsonObject.getString("strDrinkThumb")  
   modelDrink.add(dataModel)  
   }  
   drinkAdapter?.notifyDataSetChanged()  
   } catch (e: JSONException) {  
   Toast.makeText(this@ListDrinkActivity,  
   "Oops, gagal menampilkan minuman.", Toast.*LENGTH\_SHORT*).show()  
   }  
   }  
    
   override fun onError(anError: ANError) {  
   progressDialog?.dismiss()  
   Toast.makeText(this@ListDrinkActivity,  
   "Oops! Sepertinya ada masalah dengan koneksi internet kamu.", Toast.*LENGTH\_SHORT*).show()  
   }  
   })  
   }  
    
   override fun onOptionsItemSelected(item: MenuItem): Boolean {  
   if (item.*itemId* == R.id.*home*) {  
   finish()  
   return true  
   }  
   return super.onOptionsItemSelected(item)  
   }  
    
  }
* RecipeDrinkActivity
* package com.azhar.drinkrecipe.activities  
    
  import android.app.ProgressDialog  
  import android.os.Bundle  
  import android.view.MenuItem  
  import android.widget.Toast  
  import androidx.appcompat.app.AppCompatActivity  
  import com.androidnetworking.AndroidNetworking  
  import com.androidnetworking.common.Priority  
  import com.androidnetworking.error.ANError  
  import com.androidnetworking.interfaces.JSONObjectRequestListener  
  import com.azhar.drinkrecipe.R  
  import com.azhar.drinkrecipe.databinding.ActivityRecipeDrinkBinding  
  import com.azhar.drinkrecipe.model.ModelDrink  
  import com.azhar.drinkrecipe.networking.ApiEndpoint  
  import com.bumptech.glide.Glide  
  import com.bumptech.glide.load.engine.DiskCacheStrategy  
  import org.json.JSONException  
  import org.json.JSONObject  
    
  class RecipeDrinkActivity : AppCompatActivity() {  
    
   private val binding: ActivityRecipeDrinkBinding {  
   ActivityRecipeDrinkBinding.inflate(layoutInflater)  
   }  
    
   companion object {  
   const val DRINK\_RECIPE = "RECIPE"  
   }  
    
   var modelDrink: ModelDrink? = null  
   var progressDialog: ProgressDialog? = null  
   var idDrink: String? = null  
   var strDrink: String? = null  
   var strCategory: String? = null  
   var strAlcoholic: String? = null  
   var strGlass: String? = null  
   var strInstructions: String? = null  
   var strDrinkThumb: String? = null  
    
   override fun onCreate(savedInstanceState: Bundle?) {  
   super.onCreate(savedInstanceState)  
   setContentView(binding.*root*)  
    
   progressDialog = ProgressDialog(this)  
   progressDialog?.setTitle("Mohon Tunggu...")  
   progressDialog?.setCancelable(false)  
   progressDialog?.setMessage("Sedang menampilkan resep")  
    
   setSupportActionBar(binding.toolbar)  
   *assert*(*supportActionBar* != null)  
   *supportActionBar*?.setDisplayHomeAsUpEnabled(true)  
    
   //get data intent  
   modelDrink = *intent*.getSerializableExtra(DRINK\_RECIPE) as ModelDrink  
   if (modelDrink != null) {  
   idDrink = modelDrink?.idDrink  
    
   //get data recipe  
   getRecipeDrink(idDrink)  
   }  
   }  
    
   private fun getRecipeDrink(idDrink: String?) {  
   progressDialog?.show()  
   AndroidNetworking.get(ApiEndpoint.BASEURL + ApiEndpoint.URL\_RECIPE)  
   .addPathParameter("idDrink", idDrink)  
   .setPriority(Priority.*MEDIUM*)  
   .build()  
   .getAsJSONObject(object : JSONObjectRequestListener {  
   override fun onResponse(response: JSONObject) {  
   progressDialog?.dismiss()  
   try {  
   val jsonArray = response.getJSONArray("drinks")  
   for (i in 0 *until* jsonArray.length()) {  
   val jsonObject = jsonArray.getJSONObject(i)  
    
   //get json  
   strDrink = jsonObject.getString("strDrink")  
   strCategory = jsonObject.getString("strCategory")  
   strAlcoholic = jsonObject.getString("strAlcoholic")  
   strGlass = jsonObject.getString("strGlass")  
   strInstructions = jsonObject.getString("strInstructions")  
   strDrinkThumb = jsonObject.getString("strDrinkThumb")  
    
   //set data  
   binding.tvTitle.*text* = strDrink  
   binding.tvSubtitle.*text* = strCategory  
   binding.tvAlcoholic.*text* = strAlcoholic  
   binding.tvGlass.*text* = strGlass  
   binding.tvInstructions.*text* = strInstructions  
    
   Glide.with(this@RecipeDrinkActivity)  
   .load(strDrinkThumb)  
   .diskCacheStrategy(DiskCacheStrategy.*ALL*)  
   .fitCenter()  
   .into(binding.imageDrink)  
   }  
    
   //set ingredient & measure  
   var x = 0  
   while (x < 15) {  
   if (x = 0) {  
   x++  
   }  
   binding.tvIngredients.append("\u2022 " + jsonArray.getJSONObject(0)  
   .getString("strIngredient" + x)  
   .*replace*(oldValue = "null", newValue = "-") + "\n");  
   binding.tvMeasure.append(jsonArray.getJSONObject(0)  
   .getString("strMeasure" + x)  
   .*replace*(oldValue = "null", newValue = "-") + "\n");  
   }  
   } catch (e: JSONException) {  
   Toast.makeText(this@RecipeDrinkActivity,  
   "Oops, gagal menampilkan resep.", Toast.*LENGTH\_SHORT*).show()  
   }  
   }  
    
   override fun onError(anError: ANError) {  
   progressDialog?.dismiss()  
   Toast.makeText(this@RecipeDrinkActivity,  
   "Oops! Sepertinya ada masalah dengan koneksi internet kamu.", Toast.*LENGTH\_SHORT*).show()  
   }  
   })  
   }  
    
   override fun onOptionsItemSelected(item: MenuItem): Boolean {  
   if (item.*itemId* == R.id.*home*) {  
   finish()  
   return true  
   }  
   return super.onOptionsItemSelected(item)  
   }  
    
  }
* CategoriesAdapter
* package com.azhar.drinkrecipe.adapter  
    
  import android.content.Context  
  import android.content.Intent  
  import android.view.LayoutInflater  
  import android.view.ViewGroup  
  import androidx.recyclerview.widget.RecyclerView  
  import com.azhar.drinkrecipe.activities.ListDrinkActivity  
  import com.azhar.drinkrecipe.databinding.ListItemCategoriesBinding  
  import com.azhar.drinkrecipe.model.ModelCategories  
    
  class CategoriesAdapter(private val modelCategories: List<ModelCategories>, private val context: Context) :  
   RecyclerView.Adapter<CategoriesAdapter.ViewHolder>() {  
    
   override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): ViewHolder = ViewHolder(  
   ListItemCategoriesBinding.inflate(LayoutInflater.from(parent.context), parent, false)  
   )  
    
   override fun onBindViewHolder(holder: ViewHolder, position: Int) {  
   with(holder.itemView) **{** val model = modelCategories[position]  
    
   /\*Random randomColor = new Random();  
   int currentColor = Color.argb(255, randomColor.nextInt(256), randomColor.nextInt(256), randomColor.nextInt(256));  
   holder.cvCategories.setCardBackgroundColor(currentColor);\*/  
    
   holder.binding.tvCategories.text = model.strCategory  
    
   root.setOnClickListener **{** val intent = Intent(context, ListDrinkActivity::class.java)  
   intent.putExtra(ListDrinkActivity.LIST\_DRINK, modelCategories[position])  
   context.startActivity(intent)  
   **}  
   }** }  
    
   override fun getItemCount(): Int = modelCategories.size  
    
   class ViewHolder(val binding: ListItemCategoriesBinding)RecyclerView.ViewHolder(binding.root)

}

* DrinkAdapter
* package com.azhar.drinkrecipe.adapter  
    
  import android.content.Context  
  import android.content.Intent  
  import android.view.LayoutInflater  
  import android.view.ViewGroup  
  import androidx.recyclerview.widget.RecyclerView  
  import com.azhar.drinkrecipe.activities.RecipeDrinkActivity  
  import com.azhar.drinkrecipe.databinding.ListItemDrinkBinding  
  import com.azhar.drinkrecipe.model.ModelDrink  
  import com.bumptech.glide.Glide  
  import com.bumptech.glide.load.engine.DiskCacheStrategy  
    
    
  class DrinkAdapter(private val modelDrink: List<ModelDrink>, private val context: Context) :  
   RecyclerView.Adapter<DrinkAdapter.ViewHolder>() {  
    
   override fun onCreateViewHolder(parent: ViewGroup, viewType: Int): ViewHolder = ViewHolder(  
   ListItemDrinkBinding.inflate(LayoutInflater.from(parent.context), parent, false)  
   )  
    
   override fun onBindViewHolder(holder: ViewHolder, position: Int) {  
   with(holder.itemView) **{** val model = modelDrink[position]  
    
   Glide.with(context)  
   .load(model.strDrinkThumb)  
   .diskCacheStrategy(DiskCacheStrategy.ALL)  
   .fitCenter()  
   .into(holder.binding.imageDrink)  
    
   holder.binding.tvNameDrink.text = model.strDrink  
    
   root.setOnClickListener **{** val intent = Intent(context, RecipeDrinkActivity::class.java)  
   intent.putExtra(RecipeDrinkActivity.DRINK\_RECIPE, modelDrink[position])  
   context.startActivity(intent)  
   **}  
   }** }  
    
   override fun getItemCount(): Int = modelDrink.size  
    
   class ViewHolder(val binding: ListItemDrinkBinding) : RecyclerView.ViewHolder(binding.root)  
    
  }
* Networking
* package com.azhar.drinkrecipe.networking  
    
  object ApiEndpoint {  
   var BASEURL = "https://www.thecocktaildb.com/api/json/v1/1/"  
   var URL\_CATEGORIES = "list.php?c=list"  
   var URL\_FILTER = "filter.php?c={strCategory}"  
   var URL\_RECIPE = "lookup.php?i={idDrink}"  
  }
* ModelCategories

Package com.azhar.drinkrecipe.model

Import java.io.Serializable

Class ModelCategories : Serializable {

Var strCategory : String? = null

* ModelDrink

Package com.azhar.drinkrecipe.model

Import java.io.Serializable

class ModelDrink : Serializable {

var idDrink: String? = null

var strDrink: String? = null

var strDrinkThumb: String? = mull

}

* AndroidManifest

<?xml version="1.0"encoding="utf8"?>

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

xmlns:tools=<http://schemas.android.com/tools>

package="com.azhar.drinkrecipe">

<uses-permissionandroid:name="android.permission.INTERNET

<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/AppTheme"

android:usesCleartextTraffic="true"

tools:targetApi="q">

<activity android:name="com.azhar.drinkrecipe.activities.RecipeDrinkActivity"/>

<activity android:name="com.azhar.drinkrecipe.activities.ListDrinkActivity"/>

<activityandroid:name="com.azhar.drinkrecipe.activities.CategoriesActivity">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<categoryandroid:name="android.intent.category.LAUNCHER"/>

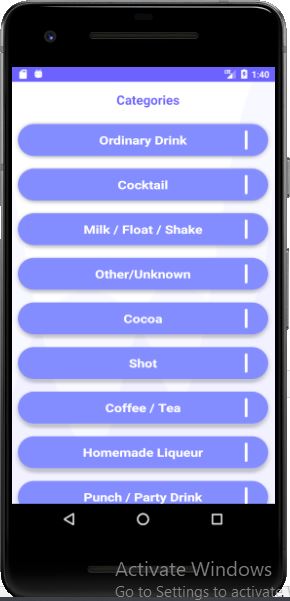
</intent-filter>

</activity>

</application>

</manifest>

* 1. **Testing**

****

**Keterangan**

Dimana disini dia tidak dapat membuka semua resep makanan tersebut karena sebuah kesalahan coding dan jaringan sehingga tidak dapat berjalan nya sebuehan program tersebut

* 1. **Debuging**

Penambahan dmna .replace (oldValue dan newValue)

binding.tvIngredients.append("\u2022 " + jsonArray.getJSONObject(0)  
 .getString("strIngredient" + x)  
 .*replace*(oldValue = "null", newValue = "-") + "\n");  
 binding.tvMeasure.append(jsonArray.getJSONObject(0)  
 .getString("strMeasure" + x)  
 .*replace*(oldValue = "null", newValue = "-") + "\n");  
}

jika tidak ditambah sebuah replace maka coding akan menjadi eror

root.setOnClickListener **{** val intent = Intent(*context*, RecipeDrinkActivity::class.*java*)  
 intent.putExtra(RecipeDrinkActivity.DRINK\_RECIPE, modelDrink[position])  
 *context*.startActivity(intent)

dan dsini awal coding nya cuman root. Maka coding tersebut tidak akan berjalan maka saya di menambah root.view

*rootView*.setOnClickListener **{** val intent = Intent(*context*, RecipeDrinkActivity::class.*java*)  
 intent.putExtra(RecipeDrinkActivity.DRINK\_RECIPE, modelDrink[position])  
 *context*.startActivity(intent)

tampilan nya seperti ini menambakan sebuah coding rootView.

**BAB IV**

**KESIMPULAN DAN SARAN**

* 1. **Kesimpulan**

1. Aplikasi pengingat tugas/to do list dapat berjalan pada sistem operasi Android. yang mana spefikasi minimum nya yaitu Android versi lollipop 5.0 aplikasi berjalan harus menggunakan koneksi internet (Online). di arapkan aplikasi dapat mempermudah user dalam mengakses aplikasi karena menggunakan system operasi android dan membantu user dalam membuat daftar list pengingat.
   1. **Saran**

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

**LAMPIRAN**

* 1. **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>