

**LAPORAN PRAKTIKUM
PEMROGRAMAN MOBILE
MODUL 5**



Mengambil Data dari Internet

Oleh:

Muhammad Ammarin Ihsan NIM. 2010817210002

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS TEKNIK
UNIVERSITAS LAMBUNG MANGKURAT
MEI 2022**

LEMBAR PENGESAHAN
LAPORAN PRAKTIKUM PEMROGRAMAN MOBILE
MODUL 5

Laporan Praktikum Pemrograman Mobile Modul 5: Mengambil Data dari Internet ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman Mobile. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Muhammad Ammarin Ihsan
NIM : 2010817210002

Menyetujui,
Asisten Praktikum

Mengetahui,
Dosen Penanggung Jawab Praktikum

Rezi Rahdianor
NIM. 1810817210019

Andreyan Rizky Baskara, S.Kom., M.Kom.
NIP. 199307032019031011

DAFTAR ISI

LEMBAR PENGESAHAN	2
DAFTAR ISI	3
DAFTAR GAMBAR.....	4
SOAL.....	5
HASIL Pengerjaan.....	5
Daftar Sourcecode:	5
Daftar XML:	6
Source Code.....	6
Hasil Gambar	13
TAUTAN GIT	14

DAFTAR GAMBAR

Gambar 1. Daftar Sourcecode	5
Gambar 2. Daftar XML	6
Gambar 3. Tampilan Awal	13
Gambar 4. Tampilan ketika mengklik item dan masuk ke menu detail	14

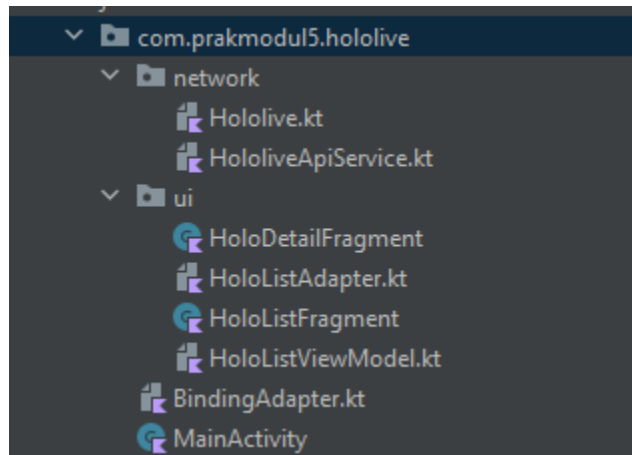
SOAL

Buatlah sebuah aplikasi Android sederhana dengan spesifikasi sebagai berikut:

1. Daftar Public API yang dapat digunakan dapat dilihat pada link berikut: <https://github.com/public-apis/public-apis> (dapat juga mengambil diluar dari link tersebut).
2. Pada saat dijalankan, aplikasi akan terhubung dengan Internet untuk menarik data dari Public API tersebut.
3. Gunakan library tambahan yaitu Retrofit untuk mempermudah proses koneksi internet.
4. Gunakan library tambahan yaitu Moshi untuk mempermudah proses data JSON.
5. Data tersebut kemudian ditampilkan dalam bentuk RecyclerView.
6. Masing-masing data di RecyclerView tersebut dapat diklik untuk menampilkan detailnya.
7. Gunakan LiveData dan ViewModel untuk mempertahankan state dari aplikasi pada saat Configuration Changes.
8. Saat pengguna merotasi tampilan handphone dari Portrait menjadi Landscape maka tampilan data yang sudah ada tidak boleh hilang.

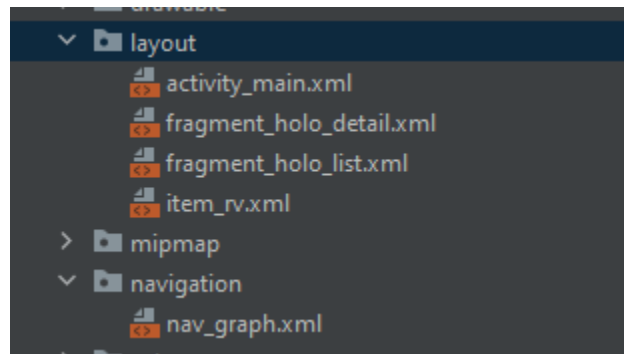
HASIL Pengerjaan

Daftar Sourcecode:



Gambar 1. Daftar Sourcecode

Daftar XML:



Gambar 2. Daftar XML

Source Code

Class MainActivity.kt

```
package com.prakmodul5.hololive

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.navigation.NavController
import androidx.navigation.fragment.NavHostFragment
import androidx.navigation.ui.NavigationUI

class MainActivity : AppCompatActivity() {
    private lateinit var navController: NavController

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val navHostFragment =
            supportFragmentManager.findFragmentById(R.id.nav_host_fragment) as
            NavHostFragment
        navController = navHostFragment.navController
        NavigationUI.setupActionBarWithNavController(this, navController)
    }
}
```

Class ini merupakan activity yang menset navigasi.

File BindingAdapter.kt

```
package com.prakmodul5.hololive

import android.view.View
import android.widget.ImageView
import androidx.core.net.toUri
import androidx.databinding.BindingAdapter
import androidx.recyclerview.widget.RecyclerView
import coil.load
import com.prakmodul5.hololive.network.ChannelsItem
```

```

import com.prakmodul5.hololive.ui.HoloListAdapter
import com.prakmodul5.hololive.ui.HololiveApiStatus

@BindingAdapter("listData")
fun bindRecyclerView(recyclerView: RecyclerView, data: List<ChannelsItem>?) {
    val adapter = recyclerView.adapter as HoloListAdapter
    adapter.submitList(data)
}

@BindingAdapter("imageUrl")
fun bindImage(imgView: ImageView, imgUrl: String?) {
    imgUrl?.let {
        val imgUri = imgUrl.toUri().buildUpon().scheme("https").build()
        imgView.load(imgUri) {
            placeholder(R.drawable.loading_animation)
            error(R.drawable.ic_broken_image)
        }
    }
}

@BindingAdapter("hololiveApiStatus")
fun bindStatus(statusImageView: ImageView, status: HololiveApiStatus?) {
    when (status) {
        HololiveApiStatus.LOADING -> {
            statusImageView.visibility = View.VISIBLE
            statusImageView.setImageResource(R.drawable.loading_animation)
        }
        HololiveApiStatus.ERROR -> {
            statusImageView.visibility = View.VISIBLE
            statusImageView.setImageResource(R.drawable.ic_connection_error)
        }
        HololiveApiStatus.DONE -> {
            statusImageView.visibility = View.GONE
        }
    }
}

```

Class ini merupakan class binding yang fungsinya adalah untuk digunakan pada file xml seperti memanggil status, memanggil data, dan mengkonversikan string url gambar menjadi gambar.

Class HololiveApiService.kt

```

package com.prakmodul5.hololive.network

import com.squareup.moshi.Moshi
import com.squareup.moshi.kotlin.reflect.KotlinJsonAdapterFactory
import retrofit2.Retrofit
import retrofit2.converter.moshi.MoshiConverterFactory
import retrofit2.http.GET

private const val BASE_URL = "https://api.holotools.app/v1/"

private val moshi = Moshi.Builder()
    .add(KotlinJsonAdapterFactory())
    .build()

private val retrofit = Retrofit.Builder()
    .addConverterFactory(MoshiConverterFactory.create(moshi))

```

```

        .baseUrl(BASE_URL)
        .build()

interface HoloApiService {
    @GET("channels")
    suspend fun getData(): Hololive
}

object HoloApi {
    val retrofitService: HoloApiService by lazy {
        retrofit.create(HoloApiService::class.java)
    }
}

```

Class ini adalah class yang mengatur koneksi ke API dengan menggunakan retrofit dan moshi untuk menangani file JSON dan nantinya akan disimpan pada data class.

Data Class Hololive.kt

```

package com.prakmodul5.hololive.network

data class Hololive(
    val channels: List<ChannelsItem>? = null,
)

data class ChannelsItem(
    val name: String? = null,
    val description: String? = null,
    val photo: String? = null,
    val id: Int? = null
)

```

File ini berfungsi untuk menyimpan data JSON.

Class HoloListViewModel.kt

```

package com.prakmodul5.hololive.ui

import android.util.Log
import androidx.lifecycle.LiveData
import androidx.lifecycle.MutableLiveData
import androidx.lifecycle.ViewModel
import androidx.lifecycle.viewModelScope
import com.prakmodul5.hololive.network.ChannelsItem
import com.prakmodul5.hololive.network.HoloApi
import kotlinx.coroutines.launch

enum class HololiveApiStatus { LOADING, ERROR, DONE }
class HoloListViewModel : ViewModel() {
    private val _status = MutableLiveData<HololiveApiStatus>()
    val status: LiveData<HololiveApiStatus> = _status

    private val _hololives = MutableLiveData<List<ChannelsItem>?>()
    val hololives: MutableLiveData<List<ChannelsItem>?> = _hololives

    private val _hololive = MutableLiveData<ChannelsItem>()
    val hololive: LiveData<ChannelsItem> = _hololive
}

```



```

init {
    getHololiveData()
}

fun getHololiveData() {
    viewModelScope.launch {
        _status.value = HololiveApiStatus.LOADING
        try {
            _hololives.value = HoloApi.retrofitService.getData().channels
            _status.value = HololiveApiStatus.DONE
        } catch (e: Exception) {
            _status.value = HololiveApiStatus.ERROR
            _hololives.value = listOf()
            e.message?.let { Log.i("Pesan Error", it) }
        }
    }
}

fun onHololiveItemClicked(holo: ChannelsItem) {
    _hololive.value = holo
}
}

```

Class ini merupakan class Viewmodel yang berfungsi untuk memasukkan data dari data class menjadi live data dan menyimpannya.

Class HoloListFragment.kt

```

package com.prakmodul5.hololive.ui

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.appcompat.app.AppCompatActivity
import androidx.fragment.app.Fragment
import androidx.fragment.app.activityViewModels
import androidx.navigation.fragment.findNavController
import androidx.recyclerview.widget.LinearLayoutManager
import com.google.android.material.divider.MaterialDividerItemDecoration
import com.prakmodul5.hololive.R
import com.prakmodul5.hololive.databinding.FragmentHoloListBinding

class HoloListFragment : Fragment() {
    private val viewModel: HoloListViewModel by activityViewModels()

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        val binding = FragmentHoloListBinding.inflate(inflater)
        viewModel.getHololiveData()
        binding.lifecycleOwner = viewLifecycleOwner
        binding.viewModel = viewModel
        binding.rv.adapter = HoloListAdapter(HololiveListener { holo ->
            viewModel.onHololiveItemClicked(holo)
            findNavController()
                .navigate(R.id.action_holoListFragment_to_holoDetailFragment)
        })
    }
}

```

```

    })

    (activity as AppCompatActivity).supportActionBar?.title = "Data Vtuber
Hololive"
    binding.rv.addItemDecoration(
        MaterialDividerItemDecoration(
            requireContext(),
            LinearLayoutManager.VERTICAL
        )
    )

    return binding.root
}
}

```

Class ini merupakan class fragment yang berfungsi untuk menampilkan data pada halaman utama nantinya.

Class HoloListAdapter.kt

```

package com.prakmodul5.hololive.ui

import android.view.LayoutInflater
import android.view.ViewGroup
import androidx.recyclerview.widget.DiffUtil
import androidx.recyclerview.widget.ListAdapter
import androidx.recyclerview.widget.RecyclerView
import com.prakmodul5.hololive.databinding.ItemRvBinding
import com.prakmodul5.hololive.network.ChannelsItem

class HoloListAdapter(private val clickListener: HololiveListener) :
    ListAdapter<ChannelsItem, HoloListAdapter.HoloViewHolder>(DiffCallback) {
    class HoloViewHolder(
        var binding: ItemRvBinding
    ) : RecyclerView.ViewHolder(binding.root) {
        fun bind(clickListener: HololiveListener, hololive: ChannelsItem) {
            binding.holoItem = hololive
            binding.clickListener = clickListener
            binding.executePendingBindings()
        }
    }

    companion object DiffCallback : DiffUtil.ItemCallback<ChannelsItem>() {
        override fun areItemsTheSame(oldItem: ChannelsItem, newItem:
ChannelsItem): Boolean {
            return oldItem.id == newItem.id
        }

        override fun areContentsTheSame(oldItem: ChannelsItem, newItem:
ChannelsItem): Boolean {
            return oldItem.name == newItem.name
        }
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType: Int):
HoloViewHolder {
        val inflater = LayoutInflater.from(parent.context)
        return HoloViewHolder(

```

```

        ItemRvBinding.inflate(layoutInflater, parent, false)
    )
}

override fun onBindViewHolder(holder: HoloViewHolder, position: Int) {
    val holo = getItem(position)
    holder.bind(clickListener, holo)
}

}

class HololiveListener(val clickListener: (holo: ChannelsItem) -> Unit) {
    fun onClick(holo: ChannelsItem) = clickListener(holo)
}

```

Class ini merupakan class adapter untuk recycler view pada class HoloListFragment yang berfungsi untuk membuat recycler view.

Class HoloDetailFragment.kt

```

package com.prakmodul5.hololive.ui

import android.os.Bundle
import android.view.LayoutInflater
import android.view.MenuItem
import android.view.View
import android.view.ViewGroup
import androidx.appcompat.app.AppCompatActivity
import androidx.fragment.app.Fragment
import androidx.fragment.app.activityViewModels
import androidx.navigation.fragment.findNavController
import com.prakmodul5.hololive.R
import com.prakmodul5.hololive.databinding.FragmentHoloDetailBinding

class HoloDetailFragment : Fragment() {
    private val viewModel: HoloListViewModel by activityViewModels()

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        val binding = FragmentHoloDetailBinding.inflate(inflater)
        binding.lifecycleOwner = this
        binding.viewModel = viewModel

        (activity as AppCompatActivity).supportActionBar?.title =
        viewModel.hololive.value?.name
        return binding.root
    }

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setHasOptionsMenu(true)
    }

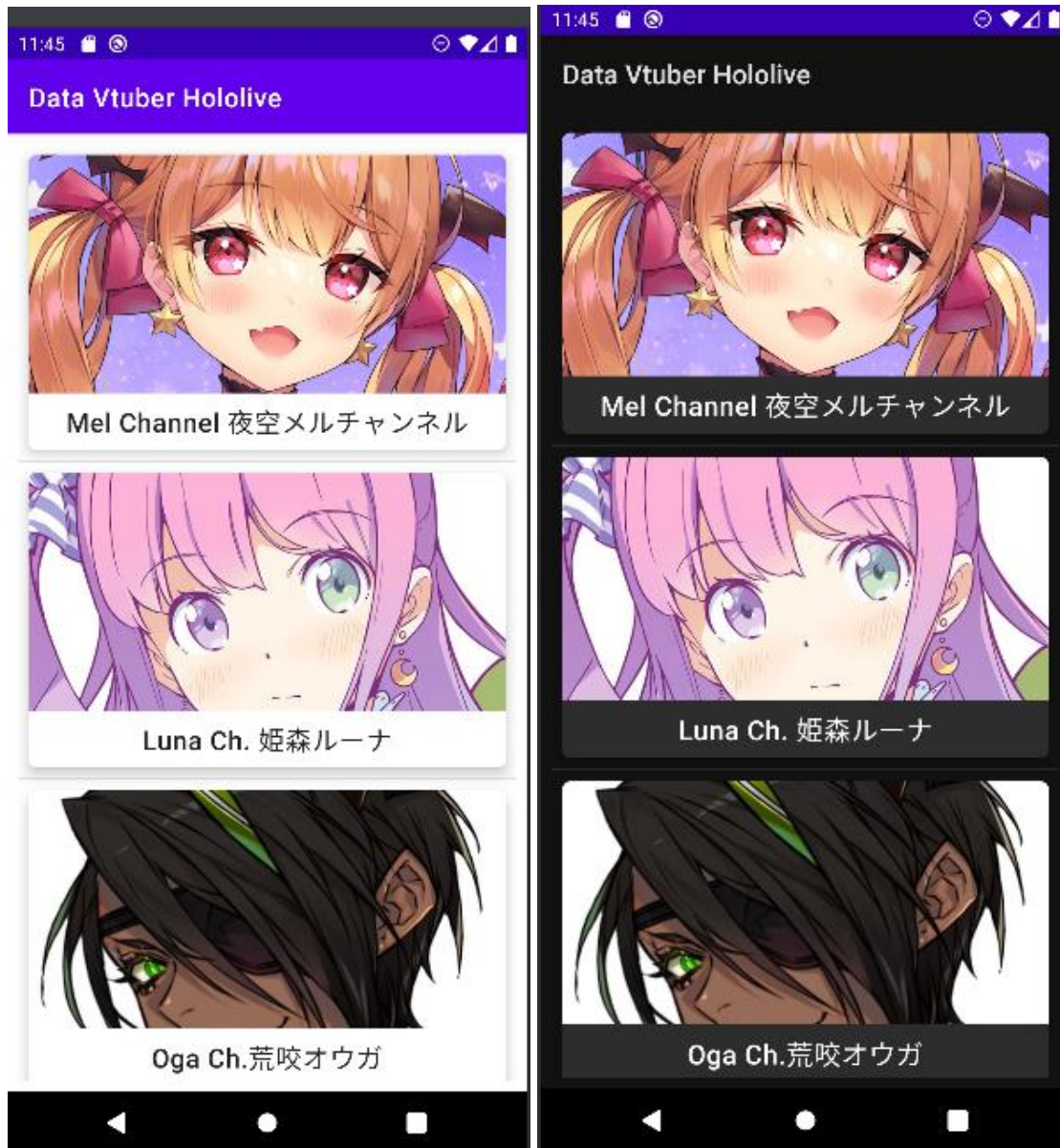
    override fun onOptionsItemSelected(item: MenuItem): Boolean {
        when (item.itemId) {
            android.R.id.home ->

```

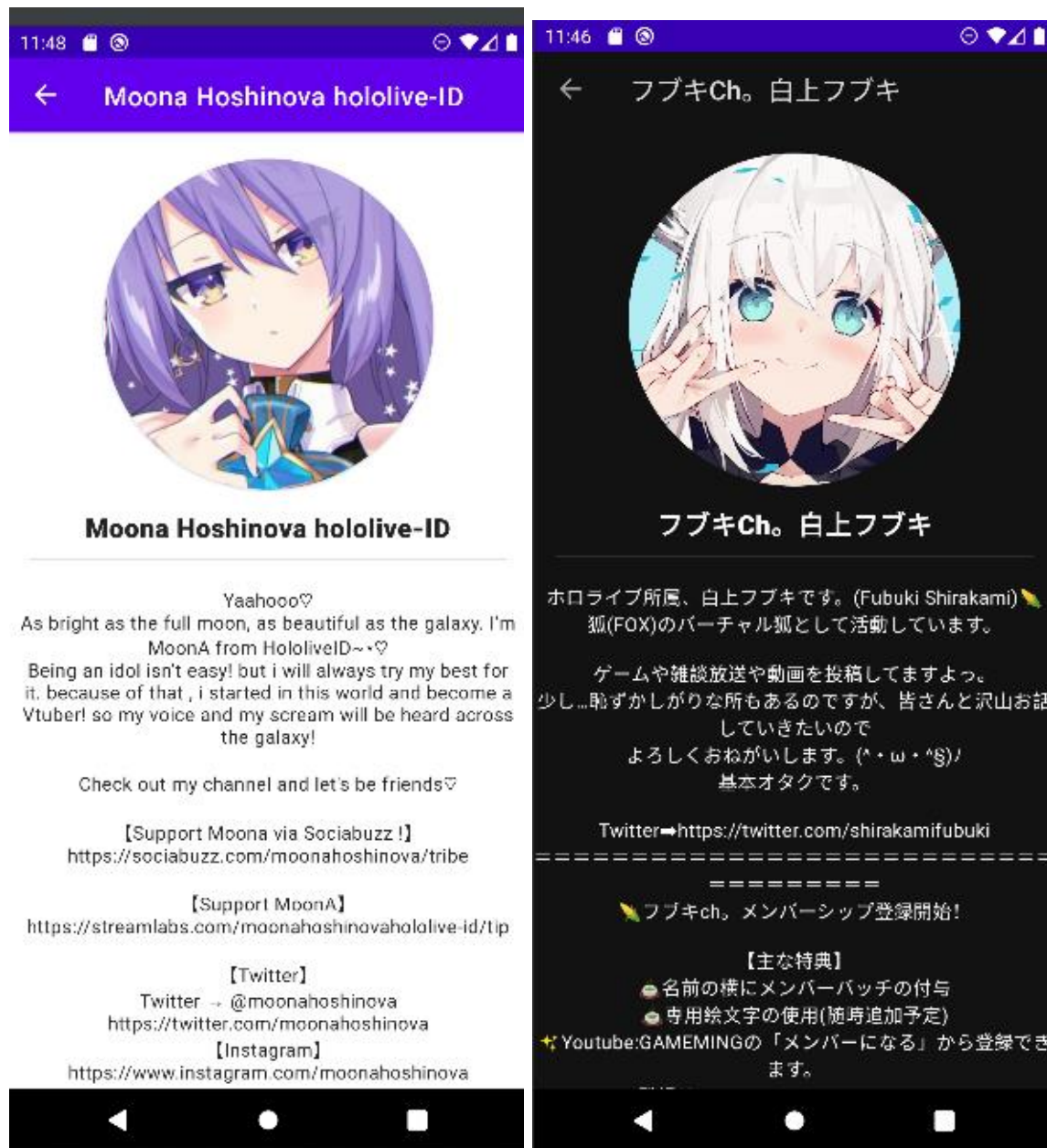
```
findNavController().navigate(R.id.action_holoDetailFragment_to_holoListFragment
)
    }
    return true
}
}
```

Class ini merupakan class fragment yang menampilkan detail ketika salah satu item pada recycler view di class HoloListFragment diclick.

Hasil Gambar



Gambar 3. Tampilan Awal



Gambar 4. Tampilan ketika mengklik item dan masuk ke menu detail

TAUTAN GIT

<https://github.com/FollowID/praktikummobile2/>