|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| * Mengimplementasikan Fragment pada Android * Membuat Movie catalod dengan retofit dan API | 90 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

**1.** Memahami bahwa fragment adalah bagian dari UI, biasanya bertempat pada activity yang memungkinkan untuk mendesign activity yang modular. Singkatnya fragment merupakan sub activity utama dan tidak dapat bekerja tanpa adanya activity utama (lengket).

**2.** Memahami dan dapat menambahkan liblary untuk keperluan design pada android studio yaitu pada file→Project Structure→Depencies→Add Depencies yang Liblary Depencies lalu tinggal input keywords nya dan search untuk versi dan jenis lalu akan ditambahkan pada android studio

**3.** Dapat menambahkan icon vector untuk keperluan UI Aplikasi pada Drawable vector asset

**4.** Memahami cara menambahkan fragmen pada UI program Yaitu :

* + Pada activity main xml menambahkan FrameLayout yg berisi id, layout untuk width, height, margin bottom dan constraint untuk left to left of dan top to top of
  + Membuat resource liblary baru dengan nama menu, lalu menambahakan resource file baru dengan format menu. Kemudian diisi dengan tag item yang berisi data icon untuk ditambahkan ke layar, yaitu android id, icon dan title
  + Lalu pada activity main xml menambahkan tag untuk liblary yang sudah ditambahkan sebelumnya dan mengatur ukuran serta letaknya terutama layout constraint yang bottom. Menu diatur bottom, jadi icon nya akan tampil sebagai navbar pada bagian bawah app
  + Tidak lupa diberi id pada tag liblary nya agar bisa diatur
  + Pada package utama, klik kanan dan tambahkan fragmen blank masing2 untuk 4 icon dan diberi nama.
  + Untuk mengatur tampilan fragmen, pada tag TextView bisa diatur posisi text, size, dan layout width juga height nya
  + Untuk mengatur onClick nya apabila icon di klik maka program akan mengalihkan kepada fragmen yang sudah di buat yaitu pada main activitu java di class utama tambahkan impliments untuk bottom nav dan OnnavItemSelectedListener ini berguna layaknya event OnClick
  + Pada function onCreate tambahkan method untuk dijalanakan apabila app pertama kali dibuka yaitu loadFragmennt(new loaFragment)); juga panggil BottomNavigationView menggunakan findViewById(R.id.navigation) kemudian navigationView.setOnNavigationItemSelectedListener(this) untuk menjalanakan nya
  + Diluar onCreate klik oncrete nya loadfragment lalu dihapus
  + Lalu pada method onNavigationItemSelected tambahkan switch untuk menu icon yang di klik home, dahsboard, notifiction, account dengan menggunakan R.id.fr.home/account/notif/dashboard yang akan mengrahkan ke fragment = new Account/notif/dashboard/homeFragment();
  + Lalu loadFragment(fragment); dan oncreate method nya dan pada method nya tambahkan if (fragment != null/kosong) maka akan menjalanakan getSupportFragmentManager() FragmentManager lalu .replace)R.id.fragment\_container, fragment) dan .commit();
  + Kemudian return true lalu diluar if return false
  + Fungsi daripada syntax if yaitu untuk menetapkan kondisi apabila fragment yang dipanggil tidak ada ataupun kosong maka akan menjalankan script tersebut untuk meminta keterangan

**5.** Memahami bahwa liblary retrofit merupakan liblary khusus android dimana berfungsi untuk menghandle yang berkaitan dengan koneksi data dari android ke internet, contohnya request data ke server atau dengan kata lain Retrofit adalah sebuah library yang digunakan untuk mempermudah proses pertukaran data antara aplikasi android dengan server melalui REST API.

**6.** Dapat menambahkan liblary dengan sytax pada android studio melalui gradle script

**7.** Memahami dan dapat menerapkan penggunaan retrofit untuk membuild aplikasi (Movie Catalog) lalu menggunakan API untuk menyambungkan dengan database nya :

* Menambahkan liblary secara syntax melalui gradle script
* Menambahkan new resource folder pada res untuk UI main menu app
* Menambahkan icon search untuk nantinya akan digunakan sebagai search section
* Pada main\_menu xml tambahkan icon search juga atur tampilannya
* Menambahkan new activity untuk DetailMovie baik java dan xml nya.
* Pada xml nya bisa diatur untuk layout2 tampilan
* Sedangkan pada activity\_main.xml menambahkan recycler widget dan diberi id guna menampilkan konten movie nantinya
* Membuat resource file baru pada res-layout untuk mengatur tampilan pict, nama dan deskripsi dari movie nya menggunaakan tag imageView dan TextView dengan pengaturan RelativeLayout (tidak lupa diberi id karena berguna untuk proses coding program nantinya)
* Pada activity detail movie xml tambahkan atribut image, dan 3 text view yang masing2 untuk judul, deskripsi movie, isi deskripsi movie dan atur layout nya. Page ini berfungsi untuk menampilkan informasi film dari thumbnail yang di klik pada main\_menu (tidak lupa tiap elemen diatur id dan posisi layout nya)
* Pada mainActivity.java buat onCreateOptionMenu kemudian sebelum return getMenuInflater().inflate(R.menu.main\_menu, menu) agar fungsi search dan icon nya dapat berjalan dan tampil pada aplikasi
* Membuat akun padaa website theMoviedb.org lalu create API untuk aplikasi CatalogMovie dan mengambil code Api\_Key nya.
* Buka documentasi API nya dan get data misal untuk pemcarian film populer dengan menggunakan Api\_Key tadi
* Script data json nya di convert terlebih dahulu ke java lalu di copy bagian sebelum result untuk diletakkan pad new java activity diberi nama response. Sementara untuk script setelah result di copy dan diletakkan dalam activity yang diberi nama result.
* Kemudian buat package baru diberi nama model dan meletakkan activity response dan result pada package tersebut agar lebih rapi
* Setelah itu membuat folder package config dan menambahkan new java activity untuk meletakkan script Server
* Server diisi dengan URL\_BASE dan API\_ENDPOINT yang berisi URL\_BASE
* Membuat package lagi dengan nama rest untuk meletakkan API, add new activity java dengan nama ApiClient. Lalu diisi dengan deklarasi String BASE\_URL = ServerConfig.API\_ENDPOINT; dan deklarasi Retrofit bernilai Null;
* Membuat function static Retrofit getClient() untuk mendapatkan data diis dengan if retrofit == null maka akan create baru dengan builder lalu di convert dari addConverterFactory(GsonConverterFActory)
* Setelah itu membuat new activity untuk ApiInterface diisi dengan @GET jumlahmovie/movie{category} untuk mengambil data movie berdasarkan kategori pada link API yang direquest sebelumnya. Lalu call Response untuk menggunakan script dari data yang sudah di convert sebelumnya untuk akses path category, query api\_key, languange dan page
* Masih di dalam package Adapter, membuat adpater untuk API diisi dengan context dan Listresult lalu di generate
* Extends recyclerView.Adapter kemudian tinggal implement dan create-create syntax nya.
* Pada public MovieAdapter.MyViewHolder deklarasikan View view; dan LayoutInflater inflater nya kemudian menggunakan view untuk memanggil item\_view, parent
* Create new MovieAdapter.ViewHolder berdasarkan dari variable view tadi karena datanya sudah di ambil lalu return viewHolder;
* Sebelum mengisi function MyViewHolder, terlebih dahulu membuat function getItemCourt() untuk menampilkan resultList.size()
* DI function MyViewHolder deklarasikan gambar dan TextView untuk menampilkan poster film dan informasinya nantinya. Menggunakan itemView.findViewById
* Pada onBindViewHolder setText, get position dan getTitle untuk tvTitle dan getOverview untuk tvDescription
* Untuk function getItemCOunt isi dengan return returnList.size();
* Tambahkan liblary Glide pada glider, kemudian pada function onBindViewHolder tambahkan Glide.with(context).load link image poster dan menggunakan getPosterPath untuk mengambil datanya lalu dikirim untuk ditampilkan pada into(holder.imgPoster);
* Pada android manifest tambahkan syntax untuk permission akses internet menggunakan tag <uses-permission>

**8.** Kemudian untuk mengatur search bar agar dapat digunakan untuk mencari movie melalui judulnya, yaitu dengan :

* Pada apiinterface tambahkan @GET lagi dengan format seperti sebelumya namun dengan function yang berbeda yaitu menggunakan getQuery()
* Beralih ke mainActivity.java pada function onCreteOptionsMenu gunakan SearchView untuk akses kepada file menu dengan memanfaatkan getACtionView()
* Atur juga searchView untuk setOnQueryListeners untuk membuat event baru agar searchBar dapat digunakan dan akan otomatis membentuk 2 function onQueryTextSubmit dan onQueryTextChange
* Pada onQueryTextChange isi dengan if(newText.length() > 1) maka akan menjalakan ApiInterdace untuk getClient() dan create Apiinterface.class
* Lalu menggunakan Call<Response> untuk memanggil data getQuery pada Activity ApiInterface tidak lupa Call.enqueue (new Callback<Response>() Sehingga akan terbentuk 2 function onResponse dan onFailure
* Isi onResponse untuk memanggil response dari retrofit2 guna mendapatkan data List<Result> dengan resonse.body().getResults();
* Terakhir tambahkan adapter dan setAdapter pada recyclerView

**9.** Mengatur detail movie untuk menampilkan informasi movie apabila di klik :

* Pada activity detailMovie.java deklarasikan String untuk title,overview, image lalu ImageView untuk imgDetail guna menampilkan gambar, TextView untuk tvTitle dan tvDetail dan Result result untuk variable hasil
* Gunakan findViewById pada onCreate untuk menyambungkan variable2 yg sudah di deklarasikan dengan atribut yang sudah dibuat pada xml
* Lalu mengambilnya menggunakan result dan getOriginalTitle untuk title, getOverview untuk overview dan getPosterPath untuk image
* Kemudian setText untuk title dan tvDetail berisi overview
* Gunakan result untuk getIntent().getPacelableExtra(EXTRA\_MOVIE\_ untuk mendapatkan info movie nya
* Lalu tambahkan glide untuk mengambil gambar dari data nya, link database sumber dimuat dalam .load() sedangkan variable penyimpan gambar disimpan pada .into(imgDetail)
* Tambahkan if (getSupportActionBar() != null) untuk menampilkan button back gan gunakan .setTitle(title) sebagai pemicu nya
* Diluar if set menggunakan setDIsplayHomeAsUpEnabled(true); agar button nya tampil
* Diluar function onCreate tambahkan function onSupportNavigateUp diisi dengan finish() untuk menjalankan button nya
* Pada activity MovieAdapter di function event onClick tambahkan Intent untuk mengambil contex perintah Detail.Movie dan result untuk menampilkan nya
* Kemudian gunakan result untuk setOriginalTitle dan getOriginalTitle gunanya adalah untuk data Judul Movie
* Untuk overview adalah setOverview dan getOverview, sedangkan untuk gambar adalah setPosterPath dan getPosterPath
* Lalu gunakan intent untuk data EXTRA\_MOVIE dan parent.getContext().startActivity(intent) untuk set intent nya
* Isi return viewHolder;
* Tidak lupa untuk menyambungkan data dari variable maka menggunakan itemView.findViewByID tinggal digunakan pada masing-masing variable
* Berlaku Juga untuk RelativeLayout, tidak lupa terlebih dahulu diberi id nya

**Saya Belum Mengerti**

1.

2.

3.