

“Project Structure dan Komponen Android Studio”

Ikrima Amanda



Pengenalan Android Studio

Android Studio

- IDE (Integrated Development Environment) berbasis IntelliJ IDEA yang digunakan khusus untuk membuat aplikasi android
- Bahasa pemrograman => Java dan Kotlin
- Rekomendasi Google => Kotlin
- Fitur-fitur yang meningkatkan produktivitas saat mem-build aplikasi android :
 - Sistem build berbasis Gradle
 - Emulator yang cepat dan kaya fitur
 - Lingkungan pengembangan yang terpadu untuk mengembangkan aplikasi android
 - Template design dan code untuk fitur aplikasi umum (login, register, dll)
 - Integrasi dengan github dan Firebase mudah
 - Alat pengujian lengkap (Unit test dan End-to-end test)

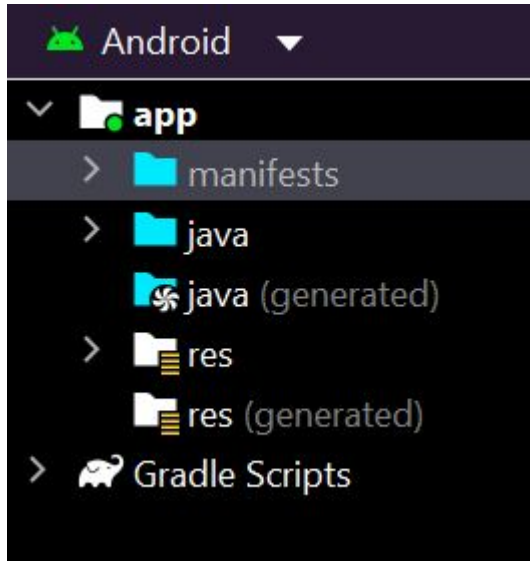
Gradle

- Sebuah program yang berfungsi untuk melakukan build secara otomatis.
- Sering disebut sebagai build-tool.
- Build => aktivitas melakukan compile dan packaging.
- Umumnya bisa compile file menggunakan command ini => `javac namafile.java`
- Jika banyak file yang di-compile sekaligus dan banyak library yg dibutuhkan, maka perlu bantuan build tools seperti Gradle.

Project Structure

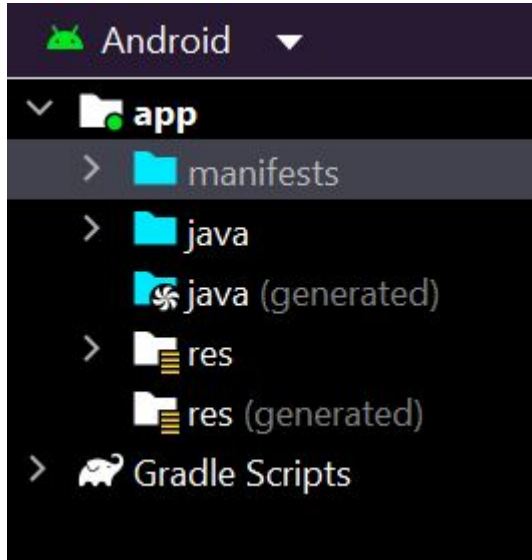
Java, Kotlin, XML, .gradle files

Project Structure (App)



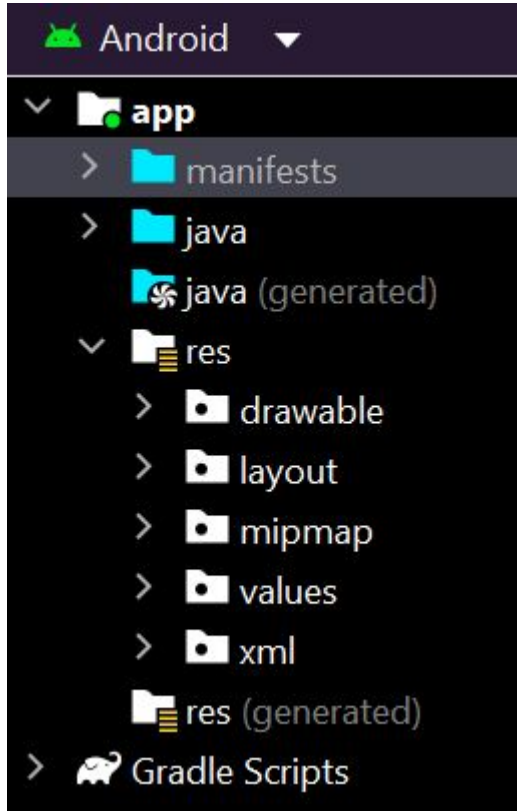
- manifest
 - Fungsi => menyimpan file Manifest aplikasi Android Anda. File Manifest adalah file XML yang berisi informasi penting tentang aplikasi Anda, seperti nama paket, aktivitas utama, izin yang diperlukan, dll.
 - File Manifest memberikan informasi dasar ke sistem operasi Android tentang bagaimana aplikasi harus berinteraksi dengan perangkat dan lingkungan sekitarnya.
- java
 - Fungsi => menyimpan source code java/kotlin dari aplikasi android yang akan kita buat.
 - Folder ini adalah tempat utama menulis kode Java untuk membangun berbagai komponen aplikasi, seperti activity, fragment, utility, helper, pengolahan data, dan sebagainya sesuai kebutuhan.

Project Structure (App)



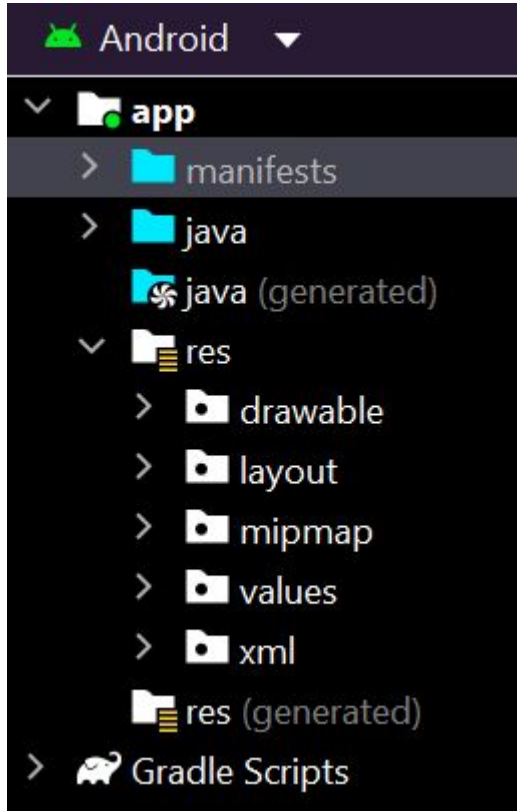
- res (resources)
 - Fungsi => menyimpan berbagai sumber daya yang digunakan oleh aplikasi android.
 - Sumber daya ini termasuk file XML, gambar, layout (tata letak), nilai-nilai string, style, icon, dan berbagai elemen lain yang digunakan dalam tampilan dan fungsionalitas aplikasi.

Project Structure (App)



- Beberapa folder di dalam folder res:
 - drawable => menyimpan berbagai gambar dan ikon yang digunakan dalam aplikasi. Ini termasuk file PNG, JPEG, SVG, dan berbagai format gambar lainnya.
 - layout => menyimpan file XML yang mendefinisikan tata letak antarmuka pengguna aplikasi. File-file ini digunakan untuk mengatur posisi dan penampilan elemen UI seperti tombol, teks, gambar, dan lain-lain.
 - mipmap => menyimpan ikon aplikasi dengan resolusi yang berbeda. Biasanya, Anda akan menempatkan ikon aplikasi yang digunakan untuk peluncuran aplikasi di sini.

Project Structure (App)



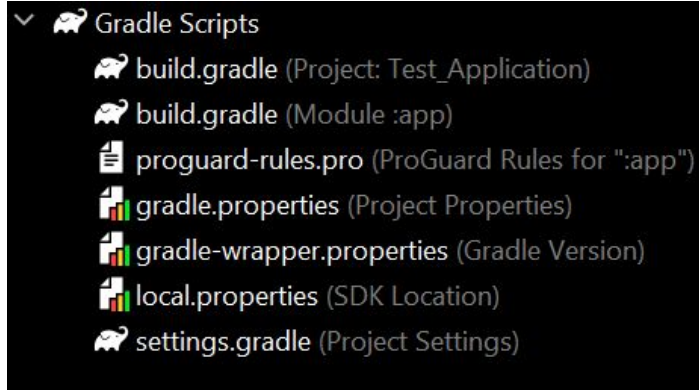
- Beberapa folder di dalam folder res:
 - values => berisi file XML yang mengandung nilai-nilai seperti string, warna, dan tema yang digunakan dalam aplikasi.
 - xml => menyimpan berbagai file XML yang digunakan dalam pengembangan aplikasi Android untuk menyimpan file XML yang tidak termasuk dalam folder "res" yang lain seperti file XML untuk konfigurasi, data, dan file-file lainnya yang tidak berhubungan dengan sumber daya UI.

Project Structure (Gradle Scripts)

- Menyimpan berbagai script terkait Gradle yang digunakan untuk mengelola proyek Android.

File penting untuk build dan konfigurasi:

- build.gradle(Project): berisi konfigurasi untuk proyek secara keseluruhan. Di file ini dapat menentukan dependensi yang dibutuhkan, dan plugin-plugin yang akan diterapkan pada proyek.
- build.gradle(Module) : setiap modul dalam proyek android memiliki file build.gradle sendiri. File ini berisi konfigurasi untuk modul tertentu, seperti mengatur build type (debug atau release), library yang digunakan, version code dan version name, dll.



ViewGroup and View

(Button, etc)

View dan ViewGroup

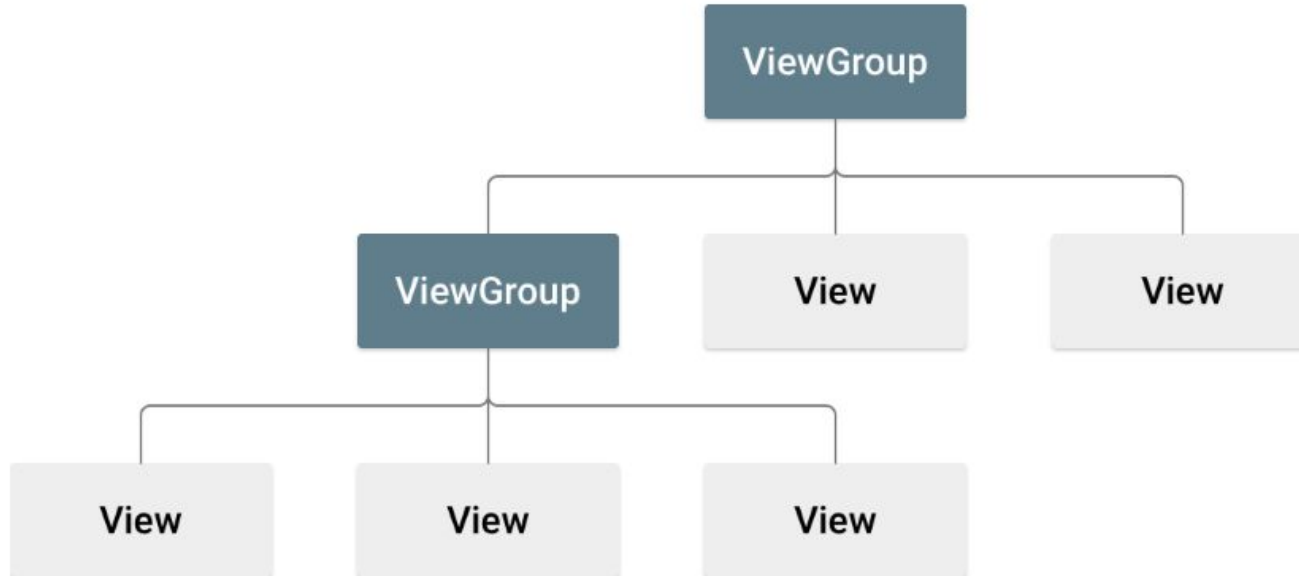
- View

- komponen dasar yang membentuk elemen antarmuka pengguna dalam aplikasi Android.
- elemen terkecil yang dapat dilihat atau berinteraksi dengan pengguna.
- Contoh => TextView (teks), ImageView (gambar), Button (tombol), dan lain-lain.

- ViewGroup

- turunan dari kelas View yang memiliki kemampuan untuk mengelola tata letak dan tampilan beberapa View lainnya.
- Fungsi => sebagai wadah (container) untuk mengatur penempatan dan hubungan antara elemen-elemen View dalam tampilan UI.
- Contoh => LinearLayout, ConstraintLayout, RelativeLayout, dan FrameLayout.

View dan ViewGroup



Source : <https://developer.android.com/guide/topics/ui/declaring-layout?hl=id>

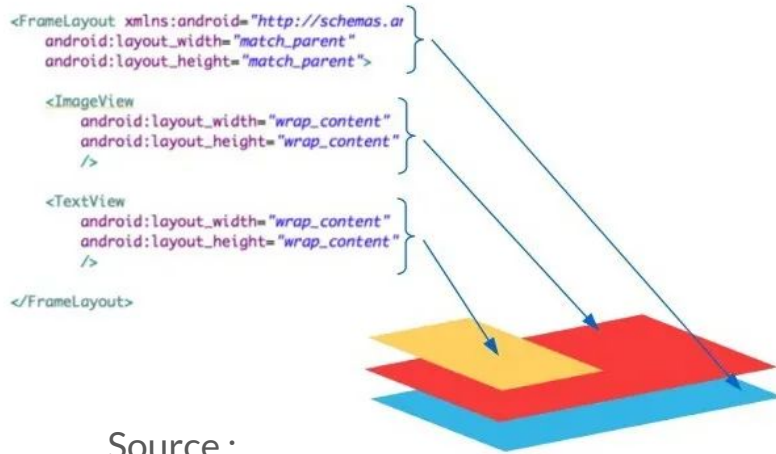
Some Views

- TextView => menampilkan teks kepada pengguna dalam aplikasi Android.
- EditText => memungkinkan pengguna memasukkan dan mengedit teks dalam aplikasi Android.
- ImageView => menampilkan gambar dalam aplikasi.
- Button => memicu tindakan atau proses tertentu dalam aplikasi.
- ProgressBar => menampilkan status proses yang sedang berjalan dalam aplikasi.

Some ViewGroups

- `FrameLayout`
- `LinearLayout`
- `RelativeLayout`
- `ConstraintLayout`
- `WebView`

FrameLayout

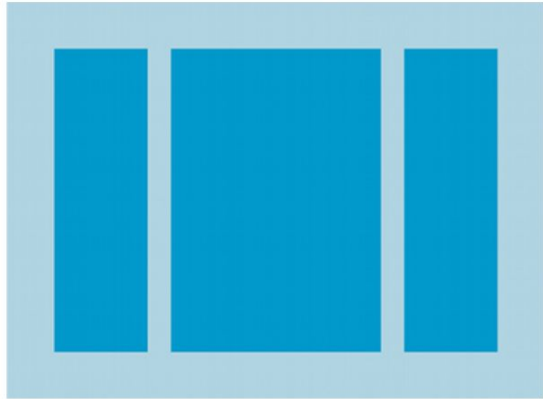


Source :

<https://medium.com/flutter-community/flutter-for-android-developers-how-to-design-frame-layout-in-flutter-93a19fc7e7a6>

Digunakan untuk menempatkan dan mengatur tata letak elemen-elemen UI dalam bentuk tumpukan (stack). Dalam FrameLayout, setiap elemen ditumpuk satu di atas yang lain, dengan elemen terakhir ditambahkan menjadi yang paling atas (paling depan).

LinearLayout



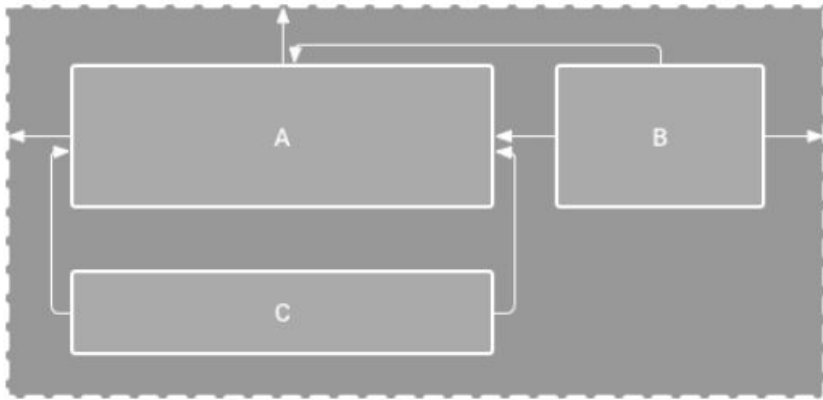
Digunakan untuk mengatur tata letak elemen-elemen UI secara linier atau berurutan. Dalam LinearLayout, elemen-elemen anak disusun dalam satu baris atau satu kolom secara berurutan.

RelativeLayout



Digunakan untuk mengatur tata letak elemen-elemen UI berdasarkan hubungan relatif antara elemen-elemen tersebut. Dalam RelativeLayout, elemen-elemen anak ditempatkan secara relatif terhadap elemen-elemen lainnya.

ConstraintLayout



salah satu jenis layout di Android yang kuat dan fleksibel, yang digunakan untuk mengatur tata letak elemen-elemen UI dengan menggunakan kendali constraints (batasan). ConstraintLayout memungkinkan untuk menentukan hubungan antara elemen-elemen anak secara visual.

WebView

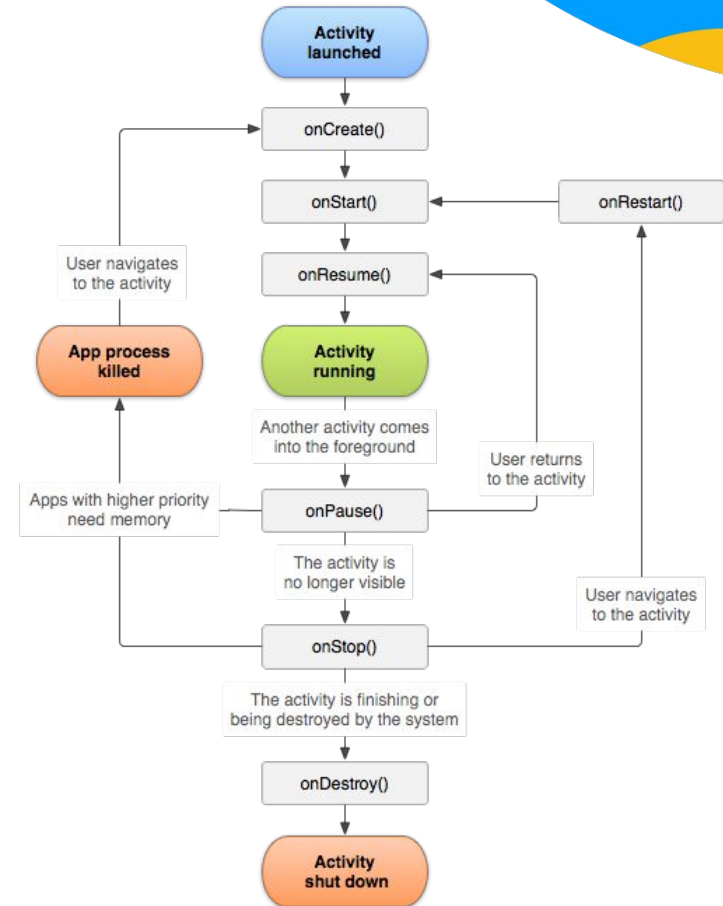
komponen antarmuka pengguna (UI) pada Android yang digunakan untuk menampilkan konten web di dalam aplikasi. WebView memungkinkan Anda untuk menampilkan halaman web, mengakses URL, menampilkan konten HTML, dan berinteraksi dengan konten web secara langsung melalui aplikasi Android.

Activity and Fragment

Lifecycle

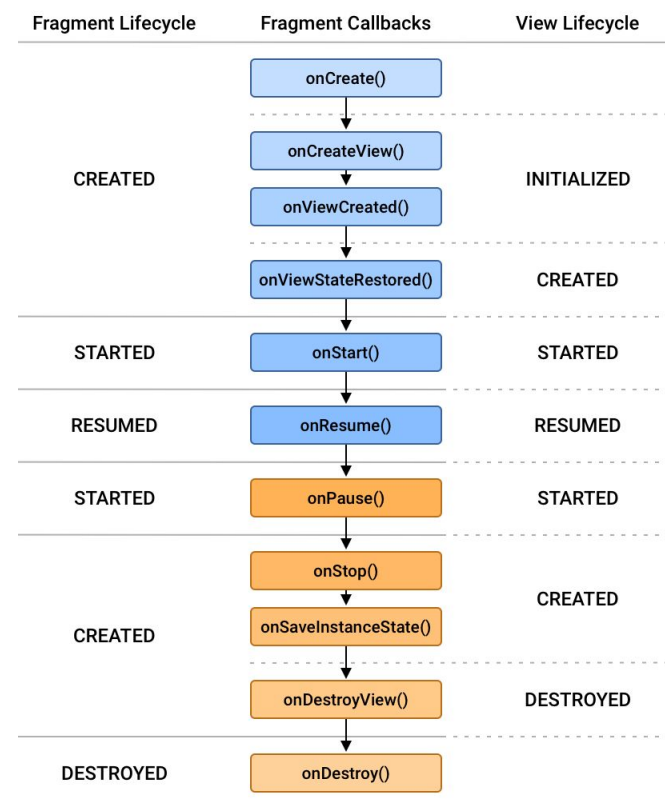
Activity Lifecycle

- Memahami siklus hidup activity sangat penting dalam mengelola keadaan dan menjaga integritas data saat activity berinteraksi dengan pengguna dan sistem Android.



Fragment Lifecycle

- Siklus hidup fragment pada Android mirip dengan siklus hidup activity, tetapi memiliki beberapa perbedaan karena fragment tidak mandiri dan bergantung pada activity. Fragment adalah komponen UI yang dapat digunakan kembali dan dapat diintegrasikan ke dalam activity untuk mengatur antarmuka pengguna.



Intent - Activity and Fragment

Intent

- komponen penting dalam pengembangan aplikasi Android yang digunakan untuk:
 - memulai aktivitas baru
 - memulai layanan
 - mengirim dan menerima pesan antar-komponen.

Intent => activity to activity

- Harus menyediakan 2 activity
- Sediakan kondisi untuk melakukan intent antar activity (Misal aksi menekan button)
- Beri command pada activity A:

```
binding.btnTest.setOnClickListener { it: View!  
    val i = Intent(packageContext: this, MainActivity::class.java)  
    startActivity(i)  
    finish()  
}
```

Intent => activity to fragment

- Sediakan 2 activity dan 1 fragment
- Sediakan kondisi untuk melakukan intent dari activity ke fragment (Misal aksi menekan button)
- Beri command:

- pada activity A:

```
binding.btnTest.setOnClickListener { it: View!  
    val i = Intent(packageContext: this, OneMainActivity::class.java)  
    i.putExtra(OneMainActivity.DATA_ONE_MAIN_ACTIVITY, OneMainActivity.DATA_ONE_FRAGMENT)  
    startActivity(i)  
    finish()  
}
```

- pada activity B:

```
private fun parcelableData() {  
    data = intent.getStringExtra(DATA_ONE_MAIN_ACTIVITY).toString()  
  
    if (data == DATA_ONE_FRAGMENT) {  
        supportFragmentManager.beginTransaction().replace(R.id.fl_container,  
            OneBlankFragment()).setTransition(FragmentTransaction.TRANSIT_FRAGMENT_OPEN).commit()  
    }  
}
```

- pada fragment:

- do anything, just template or show something.

Intent => fragment to activity

- Sediakan 2 activity dan 1 fragment
- Sediakan kondisi untuk melakukan intent dari fragment ke activity (Misal aksi menekan button)
- Beri command pada fragment:

```
binding.btnMoveToActivity.setOnClickListener { it: View!  
    val i = Intent(requireContext(), OneMainActivity::class.java)  
    startActivity(i)  
    activity?.finish()  
}
```

Intent => fragment to fragment

- LIVE CODING

How to Install Android Studio

Source...

- https://www.youtube.com/watch?v=mZUJ3EdriUo&ab_channel=DavarPedia

How to added GitHub to Android Studio

Siapkan...

- Akun GitHub
- Download Git => <https://git-scm.com/downloads>
- Ikuti tahapan seperti link =>

https://www.youtube.com/watch?v=GhfJTOu3_SE&list=PLR65H9gdaMhMr6Lu51aVL3ebEtkT0p9Qc&index=6&ab_channel=PracticalCoding

Siapkan...

- Akun GitHub
- Download Git => <https://git-scm.com/downloads>
- Install Git dengan next saja (pilihan default)
- Buka File pada tab di Android Studio untuk mengecek apakah path git nya sudah sesuai : File > Settings > Version Control > Git > See Path to Git executable
- Jika belum sesuai, cari secara manual tempat file git.exe yang sudah diinstal sebelumnya.
- Buat Repository di GitHub
- Buka tab VCS di Android Studio, click Enable Version Control Integration... > pilih Git > click OK
- Buka terminal untuk menginisialisasi git => `git init`
- Lanjut tambahkan repository => `git remote add origin link_masing-masing`
- Cek => `git remote -v`
- Lanjut menambahkan perubahan yang dilakukan di local => `git add .`
- Commit perubahan => `git commit -m "pesan..."`
- Push perubahan => `git push origin nama_branch`

Referensi

- <https://developer.android.com/guide/topics/ui/declaring-layout?hl=id>
- <https://developer.android.com/develop/ui/views/layout/declaring-layout>
- <https://developer.android.com/topic/libraries/view-binding?hl=id>
- <https://www.geeksforgeeks.org/difference-between-view-and-viewgroup-in-android/>
- <https://developer.android.com/guide/fragments/lifecycle>
- <https://developer.android.com/guide/components/activities/activity-lifecycle?hl=id>
- <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/creating-and-deleting-branches-within-your-repository>

Tugas

Tampilan Login dan Show Data

- Terdapat form untuk menginput data pengguna (Nama, Email, Jurusan, Semester)
- Terdapat validasi apakah inputan sudah sesuai atau tidak (seharusnya input email tetapi input yang lain)
- Terdapat pengecekan apakah semua form sudah terisi
- Dikumpulkan dalam bentuk **link github**
- Deadline pengumpulan tugas => **6 Juli 2023 09.00 WIB**
- **Link pengumpulan tugas** => <https://forms.gle/zmWTcXAB2jpMHYSL6>

Stuck?

talk with ur best friend...
chatgpt, google, stack over flow, medium

The Most Important Thing...

<https://www.instagram.com/reel/CtoupCfADAv/?igshid=Y2lzZGU1MTFhOQ%3D%3D>

“Practice doesn’t make perfect. Perfect practice makes perfect”

- Vince Lombardi

Terima kasih!

telegram : @ikrimaamandaa

LinkedIn :

<https://www.linkedin.com/in/ikrimaamanda/>

