



Mobile Programming

Catatan Kuliah #11

Alauddin Maulana Hirzan, M. Kom

0607069401

Otomatisasi Sistem Build (Gradle)



Otomatisasi Sistem Build (Gradle)

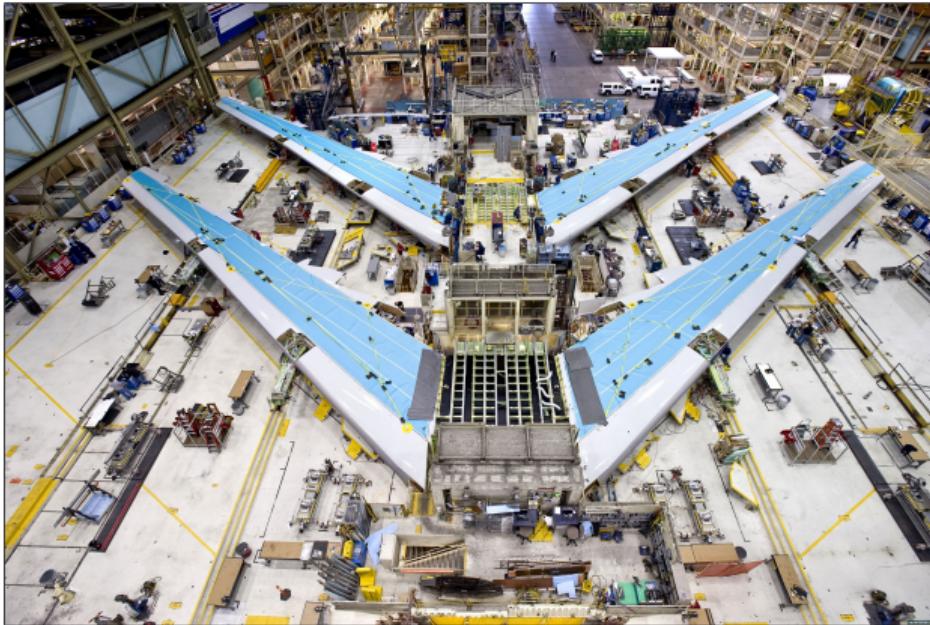
Apa itu Sistem Build (Gradle)

Bayangkan **engineer** memiliki proyek besar yang harus diselesaikan, seperti membuat model pesawat terbang. Agar lebih mudah, **engineer** membagi tugas menjadi bagian-bagian yang lebih kecil, seperti merakit sayap, memasang baling-baling, dan mengecat badan pesawat.

Maka diperlukanlah sebuah **petugas** yang memastikan tugas-tugas yang sudah terpecah tadi terpantau sukses atau tidaknya. Sehingga pesawat yang dibangun tadi tidak akan memiliki masalah yang fatal ketika akan diterbangkan.

Otomatisasi Sistem Build (Gradle)

Proses Pembangunan Pesawat





Otomatisasi Sistem Build (Gradle)

Apa itu Sistem Build (Gradle)

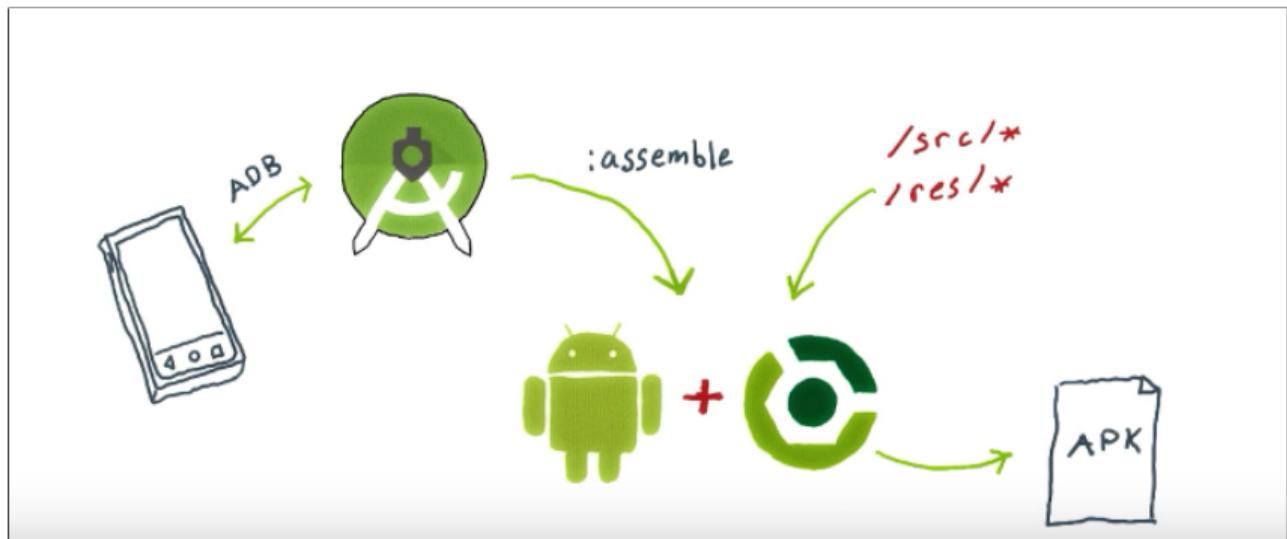
Konsep ini berlaku dengan pembuatan aplikasi **Android** yang di mana terdapat kompleksitas tinggi untuk bisa membuat aplikasi yang dapat berjalan di perangkat mobile.

Dalam dunia pemrograman, Gradle adalah sebuah sistem build. Gradle membantu **men-gelola** dan **mengatur** berbagai bagian proyek perangkat lunak, seperti **perpustakaan**, **file kode**, dan **sumber daya**. Gradle memastikan bahwa semuanya **disatukan dengan benar** dan **dalam urutan yang benar**, seperti merakit bagian-bagian yang berbeda dari sebuah model pesawat terbang.

Gradle juga menangani pengelolaan ketergantungan.

Otomatisasi Sistem Build (Gradle)

Cara Kerja Sederhana Gradle





Otomatisasi Sistem Build (Gradle)

Apa itu Sistem Build (Gradle)

Tugas Gradle:

- ▶ **Otomatisasi Pembuatan:** Gradle mengotomatiskan proses pembuatan aplikasi Android, menangani tugas-tugas seperti mengkompilasi kode sumber, mengemas sumber daya, dan menghasilkan file APK (paket aplikasi Android).
- ▶ **Manajemen Ketergantungan:** Gradle menyederhanakan manajemen ketergantungan dengan mengizinkan pengembang untuk mendeklarasikan ketergantungan pada pustaka atau modul eksternal.

Otomatisasi Sistem Build (Gradle)

Apa itu Sistem Build (Gradle)

- ▶ **Kustomisasi dan Konfigurasi:** Gradle menyediakan sistem build yang fleksibel dan dapat disesuaikan. Pengembang dapat menentukan konfigurasi build kustom, menerapkan plugin, dan menentukan varian build untuk menyesuaikan proses build dengan kebutuhan spesifik mereka.
- ▶ **Pembangunan Bertingkat:** Gradle mendukung pembangunan bertahap, yang berarti hanya membangun ulang bagian yang diperlukan dari proyek ketika ada perubahan.
- ▶ **Dukungan Proyek Multi-modul:** Gradle memungkinkan pengelolaan proyek multi-modul, di mana aplikasi Android dibagi menjadi beberapa modul atau sub-proyek.



Otomatisasi Sistem Build (Gradle)

Apa itu Sistem Build (Gradle)

- ▶ **Integrasi dan Pengujian Berkelanjutan:** Gradle terintegrasi dengan baik dengan sistem integrasi berkelanjutan (CI) dan memungkinkan pengujian yang sedang berjalan sebagai bagian dari proses build.
- ▶ **Ekstensibilitas dan Ekosistem Plugin:** Gradle sangat mudah dikembangkan dan mendukung ekosistem plugin yang luas.

Gradle bertindak sebagai tulang punggung proses build di Android Studio, menyediakan otomatisasi build, manajemen ketergantungan, opsi kustomisasi, dan dukungan untuk proyek multi-modul dan pengujian.



Otomatisasi Sistem Build (Gradle)

Apa itu Sistem Build (Gradle)

Untuk bisa membuat aplikasi Android, Gradle memiliki proses seperti berikut:

- ▶ **Inisialisasi Proyek:** Ketika membuat proyek berbasis Gradle baru, Gradle menginisialisasi struktur proyek dengan direktori dan berkas yang diperlukan, termasuk berkas **build.gradle** di tingkat proyek.
- ▶ **Membangun Konfigurasi Skrip:** File **build.gradle** berisi konfigurasi untuk proses build. Ini mendefinisikan pengaturan, repositori, ketergantungan, dan tugas yang diperlukan untuk membangun proyek.
- ▶ **Resolusi Ketergantungan:** Gradle menyelesaikan ketergantungan proyek yang dideklarasikan di dalam berkas **build.gradle**. Ini mengambil dependensi yang diperlukan dari repositori jarak jauh atau sumber lokal dan mengunduhnya ke cache Gradle lokal.

Otomatisasi Sistem Build (Gradle)

Apa itu Sistem Build (Gradle)

- ▶ **Eksekusi Tugas:** Gradle beroperasi berdasarkan prinsip tugas. Tugas mewakili tindakan spesifik yang harus dilakukan sebagai bagian dari proses pembuatan, seperti mengkompilasi kode, mengemas sumber daya, menjalankan pengujian, atau menghasilkan file APK
- ▶ **Pembuatan Bertingkat:** Gradle melakukan pembangunan inkremental dengan menganalisis perubahan dalam kode sumber, sumber daya, dan ketergantungan proyek.
- ▶ **Plugin Aplikasi:** Gradle memungkinkan aplikasi plugin untuk memperpanjang proses build dan menyediakan fungsionalitas tambahan.



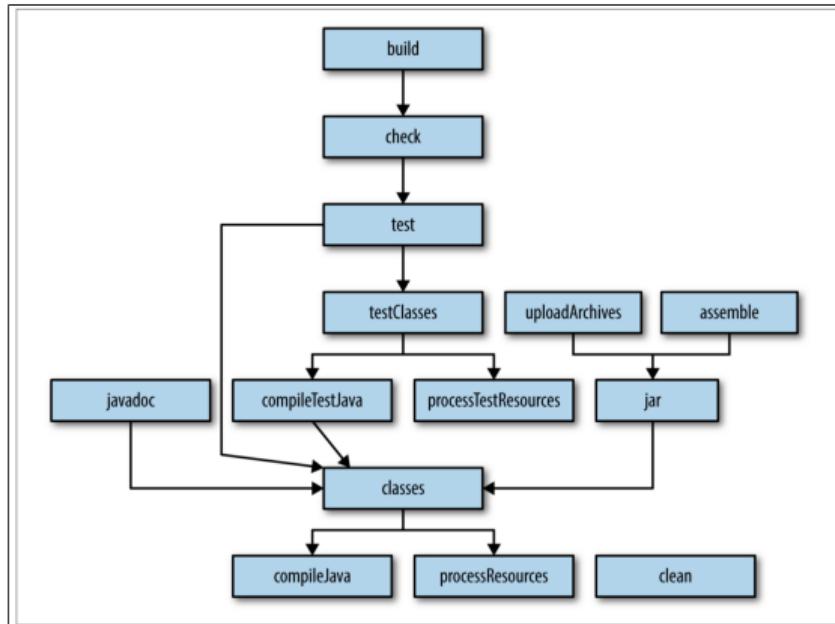
Otomatisasi Sistem Build (Gradle)

Apa itu Sistem Build (Gradle)

- ▶ **Kustomisasi dan Varian:** Gradle memungkinkan kustomisasi proses build melalui berbagai opsi konfigurasi. Hal ini memungkinkan pembuatan varian build untuk menghasilkan versi aplikasi yang berbeda.
- ▶ **Build Caching:** Gradle menyediakan build caching untuk mempercepat build berikutnya. Ini menyimpan output dari tugas yang telah selesai, memungkinkannya untuk digunakan kembali ketika input dan konfigurasi tidak berubah.
- ▶ **Membangun Keluaran:** Pada akhir proses build, Gradle menghasilkan keluaran build yang diinginkan, seperti berkas APK, laporan pengujian, atau artefak lain yang ditentukan dalam berkas build.gradle.

Otomatisasi Sistem Build (Gradle)

Task Splitting pada Gradle





Otomatisasi Sistem Build (Gradle)

Struktur Gradle

Karena projek Android yang dibuat melalui Studio adalah Projek Gradle, maka memiliki Struktur berupa:

- ▶ **Pengaturan:** File settings.gradle tingkat atas mendefinisikan pengaturan untuk build Gradle.
- ▶ **Modul:** Setiap proyek dapat terdiri dari satu atau beberapa modul, yang merupakan unit yang dapat dibangun secara independen.
- ▶ **Build Scripts:** Setiap modul memiliki file build.gradle sendiri yang berisi konfigurasi untuk proses build modul tersebut.
- ▶ **Plugin:** Skrip build dapat menerapkan plugin untuk memperluas fungsionalitas Gradle.



Otomatisasi Sistem Build (Gradle)

Struktur Gradle

- ▶ **Ketergantungan:** Gradle mengelola dependensi proyek, yang merupakan pustaka atau modul eksternal yang dibutuhkan oleh proyek.
- ▶ **Repositori:** Gradle menggunakan repositori untuk mengambil dependensi dan plugin.

Informasi

Gradle menempatkan semua konfigurasinya tepat di bawah projek Android sehingga mudah untuk dicari. Selain itu, Gradle sangat vital untuk aplikasi Android sehingga tetap digunakan meski dibangun menggunakan bahasa pemrograman lain



Otomatisasi Sistem Build (Gradle)

Directory pada Gradle

- ⚡ Gradle Scripts
 - ⚡ build.gradle (Project: Praktikum1MP)
 - ⚡ build.gradle (Module :app)
- 📄 proguard-rules.pro (ProGuard Rules for ":app")
- 📊 gradle.properties (Project Properties)
- 📊 gradle-wrapper.properties (Gradle Version)
- 📊 local.properties (SDK Location)
- ⚡ settings.gradle (Project Settings)



Otomatisasi Sistem Build (Gradle)

Manajemen Ketergantungan Gradle

Untuk bisa menarik ketergantungan yang dibutuhkan aplikasi (Contoh: fungsi tambahan aplikasi), maka programmer dapat menggunakan fitur Gradle untuk secara otomatis mengunduh file-file yang dibutuhkan.

Dalam pengaturan ketergantungan ini, harus dilalui dengan beberapa tahapan. Untuk memastikan tidak ada kerusakan di dalam **Dependency Tree** projek Android.

Otomatisasi Sistem Build (Gradle)

Manajemen Ketergantungan Gradle

- ▶ **Deklarasi Ketergantungan:** Di dalam file build.gradle sebuah modul, ketergantungan dideklarasikan dengan menggunakan blok dependencies.
- ▶ **Notasi Ketergantungan:** Ketergantungan ditentukan menggunakan notasi ketergantungan yang mencakup grup, modul, dan versi library atau modul yang diandalkan.
- ▶ **Resolusi Ketergantungan:** Gradle menyelesaikan ketergantungan dengan mengambilnya dari repositori jarak jauh yang ditentukan dalam berkas build.gradle.
- ▶ **Manajemen Repozitori:** Gradle mendukung berbagai jenis repositori, seperti Maven Central, JCenter, dan repositori khusus.



Otomatisasi Sistem Build (Gradle)

Manajemen Ketergantungan Gradle

- ▶ **Transitivitas Ketergantungan:** Gradle secara otomatis menyelesaikan ketergantungan transitif, yaitu ketergantungan yang dibutuhkan oleh ketergantungan lain.
- ▶ **Resolusi Konflik Ketergantungan:** Ketika beberapa dependensi dengan versi berbeda dideklarasikan atau ditarik secara transitif
- ▶ **Cache Ketergantungan:** Cache Gradle menyelesaikan ketergantungan dalam direktori cache lokal.
- ▶ **Pembaruan Ketergantungan Dinamis:** Gradle mendukung pembaruan ketergantungan dinamis melalui penggunaan kualifikasi versi atau rentang versi dinamis.
- ▶ **Wawasan Ketergantungan:** Gradle menyediakan alat dan perintah untuk mendapatkan wawasan tentang ketergantungan proyek.

Otomatisasi Sistem Build (Gradle)

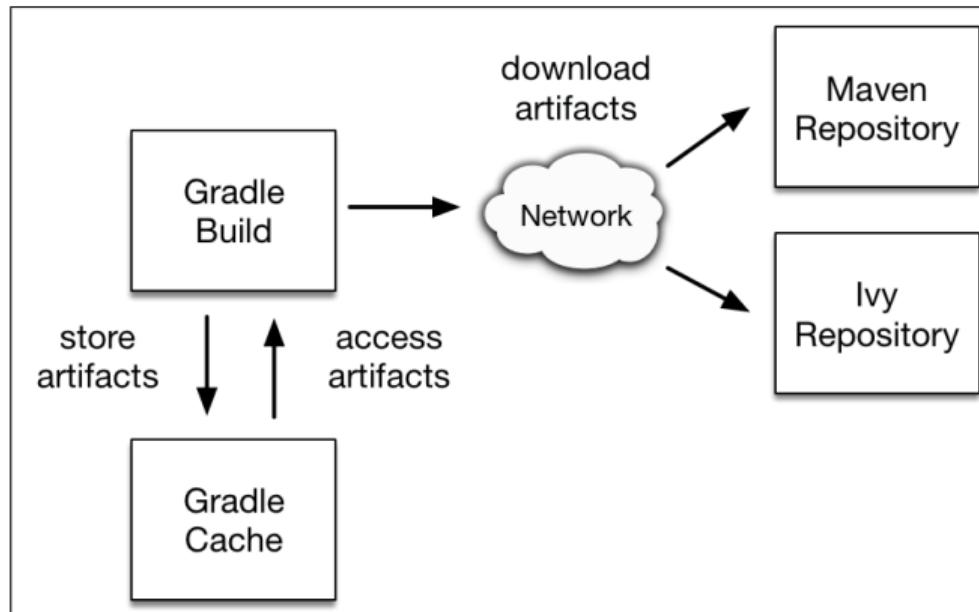
Deklarasi Ketergantungan pada Gradle

```
22 dependencies {  
23     implementation project(":module_a")  
24     implementation project(":module_b")  
25     implementation deps.app_compat_v7  
26     implementation deps.support_v4  
27     implementation deps.design  
28     implementation deps.recyclerview_v7  
29     implementation deps.kotlin_stdlib  
30 }
```

Dengan menuliskan dependency yang dibutuhkan di dalam **tag dependency** di file **build.gradle**, Gradle akan secara otomatis mengambil dependency tersebut

Otomatisasi Sistem Build (Gradle)

Sinkronisasi Ketergantungan pada Gradle





Otomatisasi Sistem Build (Gradle)

Konflik Ketergantungan pada Gradle

```
An issue was found when checking AAR metadata:  
  
1. Dependency 'com.google.android.material:material:1.10.0-alpha01' requires libraries and applications that  
depend on it to compile against codename "UpsideDownCake" of the  
Android APIs.  
  
:app is currently compiled against android-33.  
  
Recommended action: Use a different version of dependency 'com.google.android.material:material:1.10.0-alpha01',  
or set compileSdkPreview to "UpsideDownCake" in your build.gradle  
file if you intend to experiment with that preview SDK.
```

Error ini diakibatkan oleh **dependency** yang tidak sesuai dengan versi SDK yang digunakan projek



Otomatisasi Sistem Build (Gradle)

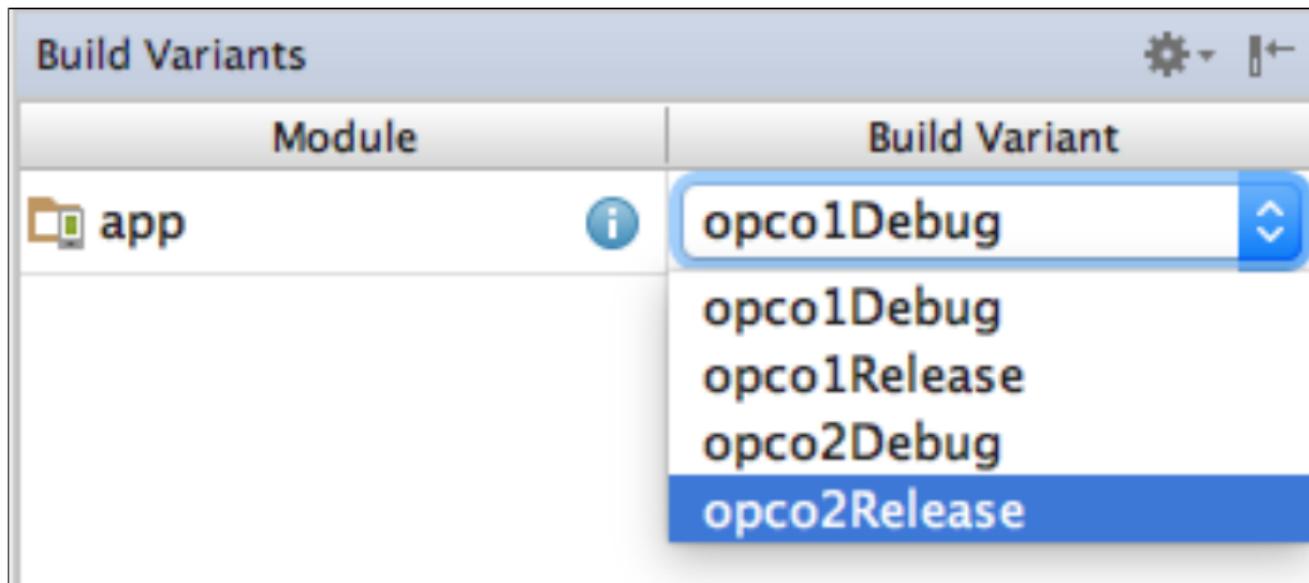
Kustomisasi Varian dan Rasa Aplikasi

Gradle dapat membuat berbagai macam variasi aplikasi hanya dengan menggunakan satu projek. Contoh sederhana dari kasus ini adalah aplikasi gratis dengan iklan dan aplikasi berbayar (premium). Semua dibangun hanya dengan satu projek.

- ▶ Membangun Varian: Gradle memungkinkan untuk membuat varian build yang berbeda untuk aplikasi Android.
- ▶ Rasa Produk: Product flavors memungkinkan untuk membuat variasi khusus aplikasi yang menargetkan konfigurasi atau audiens tertentu. Misalnya, membuat rasa untuk versi gratis dan berbayar, atau rasa untuk bahasa atau wilayah yang berbeda.

Otomatisasi Sistem Build (Gradle)

Pemilihan Variasi pada Gradle



THANK YOU

YOU