

# Mobile Programming

## Pertemuan 04

Alauddin Maulana Hirzan, S.Kom., M.Kom.  
NIDN. 0607069401

Fakultas Teknologi Informasi dan Komunikasi, Universitas Semarang



- 1 Lingkungan Pengembangan Aplikasi
- 2 Lingkungan Pengembangan Android
- 3 Lingkungan Pengembangan iOS
- 4 Lingkungan Pengembangan Lintas Platform

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

Arti daripada **Lingkungan Pengembangan** cukup jelas jika diartikan secara langsung. Namun secara terperinci **Lingkungan Pengembangan** membahas mengenai apa-apa saja (alat, sumber daya, konfigurasi) yang digunakan oleh **pengembang** atau **programmer** untuk membuat, menguji, dan memperbaiki aplikasi.

Namun tidak berhenti di situ saja, perangkat lunak maupun perangkat keras yang digunakan juga termasuk sebagai bagian dari **Lingkungan Pengembangan**.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

Secara sederhana **Lingkungan Pengembangan** memiliki komponen utama sebagai berikut:

- 1 Perangkat keras
- 2 Sistem Operasi (OS)
- 3 Lingkungan Pengembangan Terpadu (IDE)
- 4 Bahasa dan Kerangka Kerja Pemrograman
- 5 Sistem Kontrol Versi (VCS)
- 6 Alat Pembangun
- 7 Sistem Manajemen Basis Data (DBMS)
- 8 Alat Pengujian
- 9 Alat Dokumentasi
- 10 Alat Kontainerisasi dan Virtualisasi
- 11 Alat Kolaborasi dan Komunikasi
- 12 Alat Keamanan

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 1. Perangkat keras

Komponen ini termasuk komponen fisik sistem komputer yang digunakan untuk pengembangan, seperti prosesor, memori (RAM), penyimpanan (hard drive atau SSD), dan periferal seperti monitor, keyboard, dan mouse. Perangkat keras harus cukup mumpuni untuk mendukung tugas-tugas pengembangan secara efisien.

#### Informasi

Spesifikasi perangkat keras merupakan rahasia umum untuk bisa menjalankan aplikasi, apalagi membuat software yang sejenis.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 2. Sistem Operasi

Pengembang biasanya bekerja di komputer yang menjalankan sistem operasi seperti Windows, macOS, atau Linux. Pilihan OS tergantung pada preferensi pribadi, persyaratan aplikasi, dan kompatibilitas dengan alat pengembangan.

#### Informasi

Sistem Operasi seperti macOS lebih mudah untuk membuat aplikasi untuk iOS karena ketersediaan perangkat lunak.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 3. Integrated Development Environment (IDE)

IDE adalah aplikasi perangkat lunak yang menyediakan fasilitas komprehensif bagi programmer untuk pengembangan perangkat lunak. IDE biasanya mencakup editor kode, kompiler atau interpreter, alat otomatisasi pembuatan, debugger, dan fitur lainnya untuk merampingkan proses pengembangan. Contoh IDE yang populer antara lain Visual Studio Code, IntelliJ IDEA, dan Eclipse.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 4. Bahasa Pemrograman dan Framework

Bergantung pada kebutuhan proyek dan preferensi pengembang, berbagai bahasa pemrograman dan kerangka kerja dapat digunakan. Misalnya, jika mengembangkan aplikasi web, pengembang dapat menggunakan bahasa seperti JavaScript, Python, atau Java bersama dengan kerangka kerja seperti React.js, Django, atau Spring Boot.



# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 5. Sistem Kontrol Versi (Version Control System)

Perangkat lunak VCS, seperti Git, memungkinkan pengembang untuk melacak perubahan pada file kode sumber dari waktu ke waktu. Perangkat lunak ini memungkinkan beberapa pengembang untuk berkolaborasi dalam proyek yang sama, mengelola berbagai versi basis kode, dan menggabungkan perubahan dengan mulus.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 6. Alat Pembangunan

Alat bantu build mengotomatiskan proses kompilasi kode sumber menjadi program atau pustaka yang dapat dieksekusi. Contohnya adalah Apache Maven untuk proyek-proyek Java, Gradle untuk aplikasi Android, dan web-pack untuk pengembangan web.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 7. Database Management System

Untuk aplikasi yang membutuhkan penyimpanan data, pengembang menggunakan perangkat lunak DBMS seperti MySQL, PostgreSQL, MongoDB, atau SQLite. Sistem-sistem ini mengelola penyimpanan, pengambilan, dan manipulasi data secara efisien.

### 8. Alat Pengujian

Alat pengujian membantu pengembang mengotomatiskan proses pengujian, melakukan pengujian unit, pengujian integrasi, dan pengujian sistem. Contohnya adalah JUnit untuk Java, Jest untuk JavaScript, dan PyTest untuk Python.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 9. Alat Dokumentasi

Dokumentasi yang tepat sangat penting untuk memelihara dan memahami basis kode. Alat bantu dokumentasi seperti Javadoc, Sphinx, atau Doxygen membantu pengembang menghasilkan dokumentasi yang konsisten dan mudah dibaca dari komentar kode sumber.

### 10. Kontainerisasi dan Virtualisasi

Alat kontainerisasi seperti Docker dan platform virtualisasi seperti VirtualBox atau VMware memungkinkan pengembang untuk membuat lingkungan yang terisolasi untuk pengujian dan penerapan. Alat-alat ini memfasilitasi konsistensi di berbagai lingkungan dan menyederhanakan proses penerapan.

# Lingkungan Pengembangan Aplikasi

## Apa itu Lingkungan Pengembangan?

### 11. Kolaborasi

Untuk memfasilitasi kolaborasi di antara anggota tim, pengembang sering kali menggunakan alat komunikasi seperti Slack, Microsoft Teams, atau Discord, serta platform manajemen proyek seperti Jira, Trello, atau Asana.

### 12. Alat Keamanan

Pengembang menggunakan berbagai alat dan praktik keamanan untuk mengidentifikasi dan mengurangi kerentanan dalam aplikasi mereka, seperti alat analisis kode statis, pemindai kerentanan, dan pustaka enkripsi.

- 1 Lingkungan Pengembangan Aplikasi
- 2 Lingkungan Pengembangan Android**
- 3 Lingkungan Pengembangan iOS
- 4 Lingkungan Pengembangan Lintas Platform

# Lingkungan Pengembangan Android

## Apa itu Android Studio?

Android Studio adalah lingkungan pengembangan terintegrasi (IDE) resmi untuk pengembangan aplikasi Android, yang disediakan oleh Google. Ini menawarkan serangkaian alat dan fitur komprehensif yang dirancang khusus untuk membangun aplikasi Android.

Android Studio dibangun di atas platform IntelliJ IDEA dan dirancang untuk menyederhanakan seluruh proses pengembangan, mulai dari pengkodean awal hingga pengujian dan penerapan.

# Lingkungan Pengembangan Android

## Apa itu Android Studio?

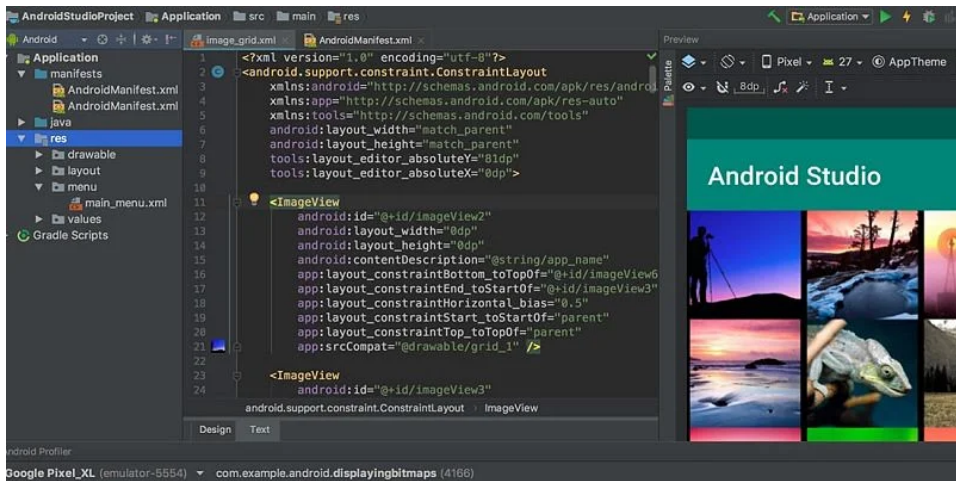
Fitur yang disediakan oleh Android Studio:

- 1 Build aplikasi berbasis Gradle
- 2 **Refactoring** khusus Android dan perbaikan cepat
- 3 Alat bantu **Lint** untuk mengetahui performa, kegunaan, kompatibilitas versi, dan masalah lainnya
- 4 Editor tata letak yang kaya yang memungkinkan pengguna untuk menarik dan melepas komponen UI
- 5 Dukungan untuk membangun aplikasi Android Wear
- 6 Perangkat Virtual Android (Emulator)



# Lingkungan Pengembangan Android

## Apa itu Android Studio?



# Lingkungan Pengembangan Android

## Android Software Development Kit

Android **Software Development Kit (SDK)** merupakan kebutuhan dasar dari Android Studio untuk membuat aplikasi berbasis Android. Tanpa ada SDK ini, aplikasi dari bahasa apapun tidak dapat dijadikan sebagai aplikasi Android.

Android SDK (Software Development Kit) adalah kumpulan alat perangkat lunak, pustaka, dan sumber daya yang disediakan oleh Google untuk memungkinkan pengembang membuat aplikasi untuk platform Android.

# Lingkungan Pengembangan Android

## Android Software Development Kit

Android SDK berisikan:

- **API Android:** Android SDK menyediakan serangkaian API (Antarmuka Pemrograman Aplikasi) yang kaya yang dapat digunakan oleh pengembang untuk mengakses berbagai fitur dan fungsi sistem operasi Android.
- **Alat Pengembangan:** Android SDK mencakup serangkaian alat pengembangan untuk memfasilitasi pembuatan dan pengelolaan aplikasi.
- **Dokumentasi dan Sampel:** Android SDK mencakup dokumentasi yang komprehensif dan contoh kode untuk membantu para pengembang memahami fitur-fitur platform dan praktik terbaik.

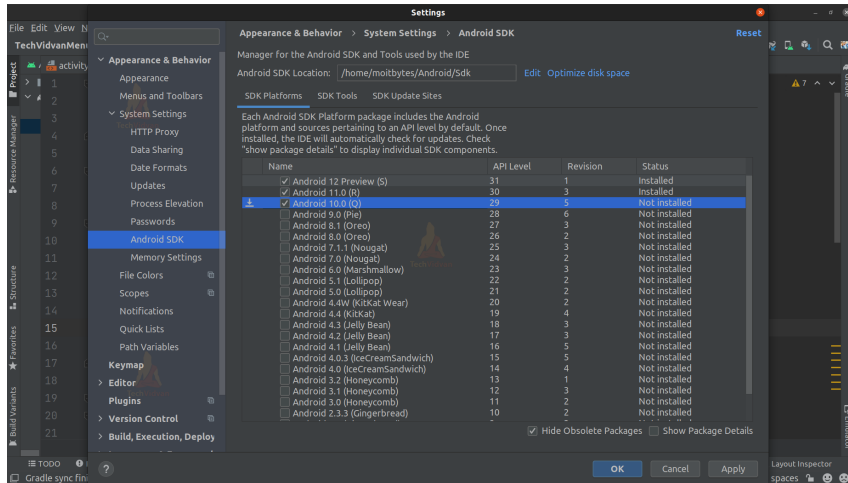
# Lingkungan Pengembangan Android

## Android Software Development Kit

- **Alat Pembangunan:** SDK Android menyertakan alat bantu pembuatan seperti Gradle dan Android Asset Packaging Tool (AAPT) untuk mengkompilasi kode sumber, mengemas sumber daya, dan membuat APK (file Paket Android) untuk distribusi.
- **Alat-alat Platform:** Alat-alat platform adalah utilitas penting untuk pengembangan Android, termasuk ADB, Fastboot, dan Systrace.
- **Sistem Image dan Skin Emulator:** Android SDK menyediakan gambar sistem dan skin emulator yang dapat digunakan oleh pengembang untuk membuat perangkat Android virtual untuk menguji aplikasi mereka

# Lingkungan Pengembangan Android

## Android Software Development Kit



- 1 Lingkungan Pengembangan Aplikasi
- 2 Lingkungan Pengembangan Android
- 3 Lingkungan Pengembangan iOS**
- 4 Lingkungan Pengembangan Lintas Platform

# Lingkungan Pengembangan iOS

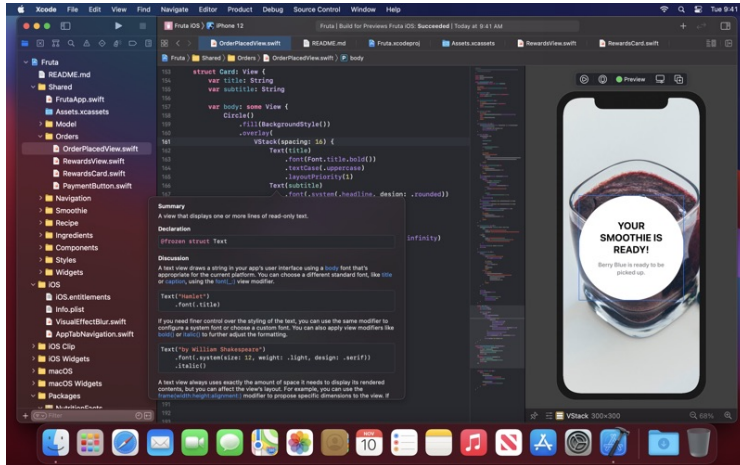
## Apa itu XCode?

Xcode adalah lingkungan pengembangan terintegrasi (IDE) resmi untuk pengembangan aplikasi iOS, macOS, watchOS, dan tvOS, yang disediakan oleh Apple Inc. Xcode menawarkan seperangkat alat dan sumber daya yang komprehensif untuk membangun, menguji, dan men-debug aplikasi di seluruh ekosistem Apple. Xcode secara luas dianggap sebagai salah satu IDE yang paling kuat dan kaya fitur untuk pengembangan platform Apple.

Dapat diakses secara gratis via App Store Apple. Namun untuk mengembangkan aplikasi iOS di XCode memerlukan sumber tambahan yang harus diinstall.

# Lingkungan Pengembangan iOS

## Apa itu XCode?





# Lingkungan Pengembangan iOS

## Apa itu XCode?

Komponen utama yang dimiliki oleh XCode:

- 1 **Interface Builder** : alat desain antarmuka pengguna grafis (GUI) yang terintegrasi ke dalam Xcode untuk membuat antarmuka pengguna untuk aplikasi iOS dan macOS.
- 2 **Simulator iOS** : alat bawaan dalam Xcode yang mengemulasi perilaku perangkat iOS pada komputer Mac
- 3 **Swift** : bahasa pemrograman yang kuat dan modern yang dikembangkan oleh Apple secara khusus untuk pengembangan aplikasi iOS, macOS, watchOS, dan tvOS.

# Lingkungan Pengembangan iOS

## Perbedaan iOS dan Android

- **Bahasa:** Xcode → Swift untuk pengembangan iOS, sedangkan Android Studio → Java atau Kotlin untuk pengembangan Android.
- **Desain Antarmuka:** Interface Builder Xcode menyediakan antarmuka visual untuk mendesain antarmuka pengguna, sedangkan Android Studio menawarkan editor tata letak dengan desain berbasis XML.
- **Penerapan:** Xcode membutuhkan komputer Mac untuk pengembangan dan distribusi aplikasi iOS, sedangkan Android Studio dapat berjalan di Windows, macOS, atau Linux.
- **Integrasi Ekosistem:** Xcode terintegrasi secara mulus dengan ekosistem Apple, sedangkan Android Studio terintegrasi dengan layanan Google seperti Firebase dan Google Play Console.

- 1 Lingkungan Pengembangan Aplikasi
- 2 Lingkungan Pengembangan Android
- 3 Lingkungan Pengembangan iOS
- 4 Lingkungan Pengembangan Lintas Platform**

# Lingkungan Pengembangan Lintas Platform

## Apa itu Pengembangan Lintas Platform?

Jika pengembang tidak ingin berada di satu sisi, maka dapat memanfaatkan Lingkungan Pengembangan **Cross Platform** untuk dapat menghasilkan aplikasi Android dan iOS.

Pengembangan lintas platform mengacu pada proses pembuatan aplikasi perangkat lunak yang dapat berjalan di berbagai sistem operasi atau platform dengan modifikasi minimal. Pendekatan ini memungkinkan pengembang untuk menulis kode sekali dan menerapkannya di berbagai platform, seperti iOS, Android, dan web, sehingga mengurangi waktu dan upaya pengembangan.

# Lingkungan Pengembangan Lintas Platform

## Framework Lintas Platform

Terdapat framework di luar Java, Kotlin, maupun Swift yang dapat digunakan sebagai framework aplikasi lintas platform seperti:

- **React Native** adalah kerangka kerja sumber terbuka yang dikembangkan oleh Facebook untuk membangun aplikasi mobile native menggunakan JavaScript dan React. Framework ini memungkinkan pengembang untuk menulis kode menggunakan satu basis kode yang dikompilasi menjadi kode native untuk platform iOS dan Android.

# Lingkungan Pengembangan Lintas Platform

## Framework Lintas Platform

- **Xamarin** adalah kerangka kerja pengembangan lintas platform yang diakuisisi oleh Microsoft, yang memungkinkan pengembang untuk membangun aplikasi seluler asli menggunakan C# dan kerangka kerja .NET.
- **Flutter** adalah toolkit UI sumber terbuka yang dikembangkan oleh Google untuk membangun aplikasi yang dikompilasi secara native untuk seluler, web, dan desktop dari satu basis kode.

# Lingkungan Pengembangan Lintas Platform

## Kelebihan Model Lintas Platform

- **Penggunaan Kembali Kode:** Kerangka kerja pengembangan lintas platform memungkinkan pengembang untuk menulis kode sekali dan menerapkannya di berbagai platform
- **Efisiensi Biaya:** Dengan menggunakan satu basis kode, organisasi dapat menghemat biaya
- **Waktu ke Market Lebih Cepat:** Pengembangan lintas platform memungkinkan iterasi dan penerapan aplikasi yang lebih cepat
- **Akses ke Fitur Asli:** Banyak kerangka kerja lintas platform yang menyediakan akses ke API dan fitur asli

# Lingkungan Pengembangan Lintas Platform

## Kekurangan Model Lintas Platform

- **Kinerja:** Aplikasi lintas platform dapat mengalami masalah kinerja dibandingkan dengan aplikasi asli
- **Pengalaman Native yang terbatas:** Pengalaman pengguna yang benar-benar asli dapat menjadi tantangan dalam pengembangan lintas platform
- **Keterbatasan Kerangka Kerja:** Setiap framework lintas platform memiliki batasan dan kendala tersendiri
- **Penguncian Vendor:** Beberapa kerangka kerja pengembangan lintas platform dapat mengikat pengembang ke platform atau ekosistem tertentu



*Terima Kasih*