

Introduction to Mobile App Development

Introduction to Mobile App Development

(A quick-reference guide for beginners)

1. What Is Mobile App Development?

- **Definition:** The process of creating software applications that run on mobile devices (smartphones, tablets, wearables).
 - **Primary Platforms:**
 - **iOS** – Apple devices (iPhone, iPad, Apple Watch).
 - **Android** – Devices from Google, Samsung, Huawei, etc.
 - **Delivery Models:**
 - **Native apps** – Built with platform-specific languages & SDKs.
 - **Cross-platform apps** – Single codebase runs on multiple OSes.
 - **Hybrid apps** – Web technologies wrapped in a native container.
-

2. Core Concepts & Terminology

Term	Meaning
SDK (Software Development Kit)	Tools, libraries, and documentation needed to build apps for a specific platform.
IDE (Integrated Development Environment)	Application that combines code editor, debugger, emulator, etc. (e.g., Xcode, Android Studio).
API (Application Programming Interface)	Set of functions that let an app interact with OS features or external services.
UI/UX	User Interface (visual layout) and User Experience (interaction flow).
Responsive Design	UI that adapts to different screen sizes and orientations.
App Store Optimization (ASO)	Techniques to improve an app's visibility in app marketplaces.

3. Development Approaches

Introduction to Mobile App Development

Approach	Languages / Frameworks	Pros	Cons
Native	• iOS: Swift, Objective-C • Android: Kotlin, Java	• Best performance • Full access to device features • Platform-specific UI guidelines	• Separate codebases ? higher cost • Longer development time
Cross-Platform	• Flutter (Dart) • React Native (JavaScript/TypeScript) • Xamarin (C#)	• One codebase for iOS & Android • Faster time-to-market • Large community support	• May need native bridges for some features • Slight performance overhead
Hybrid	• Ionic (HTML/CSS/JS) • Cordova/PhoneGap	• Leverages web development skills • Quick prototyping	• Heavier runtime • Limited native performance

4. Typical Development Workflow

1. Idea & Market Research

- Identify target users, problem solved, competitor analysis.

2. Requirement Specification

- Functional specs (features, screens).
- Non-functional specs (performance, security, offline support).

3. Design

- **Wireframes** – low-fidelity layout sketches.
- **Mockups / Prototypes** – high-fidelity UI using tools like Figma, Sketch, Adobe XD.

4. Set Up Development Environment

- Install SDK, IDE, emulator/simulator.
- Configure version control (Git).

Introduction to Mobile App Development

5. Implementation

- Build UI components.
- Integrate business logic, APIs, databases.
- Apply platform-specific guidelines (Human Interface Guidelines for iOS, Material Design for Android).

6. Testing

- **Unit tests** – individual functions/classes.
- **UI/Integration tests** – automated UI flows (Espresso, XCUITest).