SDC Flutter Mobile Dev

With Edd Gachira

What is a mobile app?

A mobile app is a software application developed specifically for use on small wireless computing devices, such as smartphones and tablets rather than desktop or laptop computers.

Mobile application development

Mobile application development is the process of creating software applications that run on mobile devices, and a typical mobile application utilizes a network connection to work with remote computing resources.

The software application can be preinstalled on the device, downloaded from a mobile app store or accessed through a mobile web browser.

Mobile apps are the most popular way for people and businesses to connect to the internet.

Many independent application development teams choose to build their apps for Android first. Why? The vast majority—around 70%—of smartphones run Android, and the Google Play Store has fewer restrictions than the Apple App Store.

Depending on the intended use case and target audience for the mobile application you are developing, you might have other considerations.

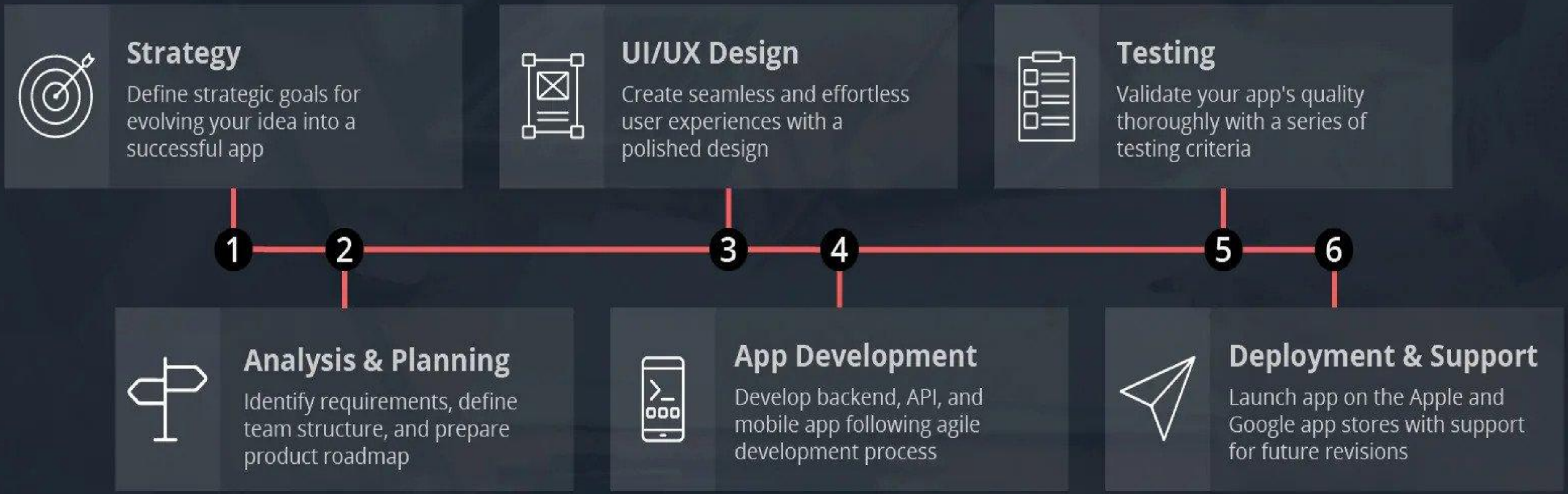
Examples;

- students - cross-platform
- customer - most use iPhones - IOS is the way to go

What Mobile Developers Do

A mobile app developer is a software engineer who works primarily with mobile devices, **creating, programming, testing, and maintaining apps and mobile platforms across devices like smartphones, tablets and laptops**.

MOBILE APP DEVELOPMENT PROCESS



1

Strategy

Define strategic goals for evolving your idea into a successful app

2

Analysis & Planning

Identify requirements, define team structure, and prepare product roadmap

3

UI/UX Design

Create seamless and effortless user experiences with a polished design

4

App Development

Develop backend, API, and mobile app following agile development process

5

Testing

Validate your app's quality thoroughly with a series of testing criteria

6

Deployment & Support

Launch app on the Apple and Google app stores with support for future revisions

INVONTO

Mobile development process

1. **Strategy** - The first step is to come up with a strategy of how your business can improve through a mobile app. How will it affect your employees, customers, business partners?

   Some things to think about and do.

   a. App's purpose
   b. App's audience
   c. Research the competition
   d. App's goals and objectives

2. **Analysis and Planning** - At this stage, your app idea starts taking shape and turns into an actual project. Analysis and planning begins with defining use cases and capturing detailed functional requirements.

Then prepare a product road map. Part of the planning phase includes identifying the skills needed for your app development initiative. For example IOS and Android use different stacks, if you want to develop for both then your team should include IOS and Android developers or Cross-Platform developers using React Native or Flutter.

3. **UI/UX Design** - The purpose for this is to deliver seamless and effortless user experiences with a polished look. The success of your app will depend on how well users adopt and benefit from its features.

- excellent user experiences, making your app interactive, intuitive and user-friendly
- polished UI designs will help with early adoption
- intuitive user experiences will keep your users engaged

4. **App Development** - Planning remains an integral part of this phase in the mobile app development process. Before actual coding/programming, you will have to;

- Define the technical architecture
- Pick a technology stack
- Define the development milestones

A typical mobile app project is made up of three integral parts;

- Backend/Server Technology - This includes the database and server-side objects needed to support mobile app functionality.
- API -An Application Programming Interface (API) is a method of communication between the app and the back-end server/database.
- Mobile App (Front-end) - This is the app the end-user installs and interacts with on their mobile devices. In most cases, mobile apps consist of interactive user experiences that rely on real-time data and require network connectivity between the app's front-end and back-end

5. **Application Testing** - Performing thorough Quality Assurance testing during the mobile app development process makes applications stable, usable and secure. Your app should undergo the testing methods below to deliver a quality solution;

- **User Experience Testing** - A critical part is to confirm that the final implementation matches the user experience created by the app design team. Visuals, workflows and interactivity are what will give end users a first hand impression of your app.
- **Functional Testing** - The accuracy of your mobile app functionality is essential to its success. This phase allows you to establish that nearly every possible behaviour has been proven to work. (tested by many users to cover as many potential testing conditions as possible)
- **Performance Testing** - There are many quantitative criterias available to measure your app's performance i.e:
    - How well is your app responding to user's requests?
    - How fast is your app's screens loading?
    - Is your app draining the phone battery or causing memory leaks?
    - Does your app leverage network bandwidth efficiently?
    - Is the size of your app bigger than it should be?
- **Security Testing** - Security is of the utmost concern for enterprise mobile apps. Any potential vulnerability can lead to a hack. Many companies hire outside agencies to perform thorough security testing on their applications. However, your QA and development teams can take a few simple measures to help secure your app.
- **Device & Platform Testing** - Your mobile app should be tested on multiple mobile devices to ensure smooth working for your app for all users.

6. **Deployment and Support** - To deploy your mobile app you must submit it to the appropriate app store, Apple App Store for IOS apps and Google Play for Android apps. You will need a developer account for each of these before launching your mobile app. Some of the requirement needs include;

- App title
- Short description
- Category
- Keywords
- Launch icon
- Screenshots
- Banner graphic
- Promotional video

Mobile Development Approaches

There are 4 major mobile development approaches when building mobile applications

- **Native mobile apps** - There are written in the platform's programming language and frameworks and run directly on the operating system of the device such as Android and IOS. Native apps provide the best performance and user experience.
- **Cross-platform mobile apps** - Can be written in a range of programming languages & frameworks and then compiled into a native application that runs on the device's operating system. (simpler apps that don't require native device features)
- **Hybrid mobile apps** - built with standard web technologies including JS, CSS & HTML5 then bundled as app installation packages.
- **Progressive web apps** - provide an alternative approach to traditional mobile app development that side-steps app store delivery and app installations. They are web applications that use browser capabilities including working offline running background processes, e.t.c

Introduction To Frameworks & Cross-platform Development

Let’s say you need to do mobile application development for both the Android and iOS operating system. What is the best software development approach?

You could develop two native applications (creating a mobile application that is tailored and dedicated to a specified platform like iOS, or Android). Taking advantage of native APIs and OS-specific programming languages can help you build a powerful app.

Another option is to go cross-platform. Cross-platform native mobile applications can be written in a variety of different programming languages and frameworks, but they are compiled into a native application running directly on the operating system of the device.

Pros of frameworks:

- Single code base for multiple platforms
- Easy to build and maintain your app
- 70–90% reusable code
- Shorter development time
- Exposure to a larger number of users and Updates synchronization

Examples include:

- Xamarin Built with #C and .Net, Xamarin allows developers to create cross-platform applications for Android, iOS, tvOS, macOS, and Windows.
- React Native. Unveiled by Facebook in 2015, React Native has reigned in the cross-platform app development market ever since. The framework is based on React — a JavaScript library for building highly responsive user interfaces.
- Apache Cordova. Apache Cordova’s tech stack features HTML5, CSS3, and JavaScript. The mobile app development framework provides access to a smartphone’s built-in accelerometer, file storage, GPS, contact data, media, and notifications.
- Flutter. Flutter is Google’s UI toolkit that allows developers to create natively compiled applications for mobile devices, web browsers, and PCs using the same codebase.

What Is Flutter?

Flutter is Google’s UI toolkit that allows developers to create natively compiled applications for mobile devices, web browsers, and PCs using the same codebase. Flutter is based on Dart — a relatively new programming language that shares many features with Swift and Kotlin. With Flutter, you can design applications that render fast and adapt to platform-specific UX logic.

Flutter helps developers build high-performance, scalable applications with attractive and functional user interfaces for Android or IOS.

Flutter relies on a library of pre-made widgets that make it simple for developers to build and launch mobile applications quickly.

Flutter’s extensive widget library also gives developers ready access to pre-made functionality, reducing overall coding time. And reduced coding time leads to reduced development costs.

Flutter’s popularity has skyrocketed for very good reason. Flutter offers developers a wide range of benefits that speed development and lead to more user-friendly mobile applications.

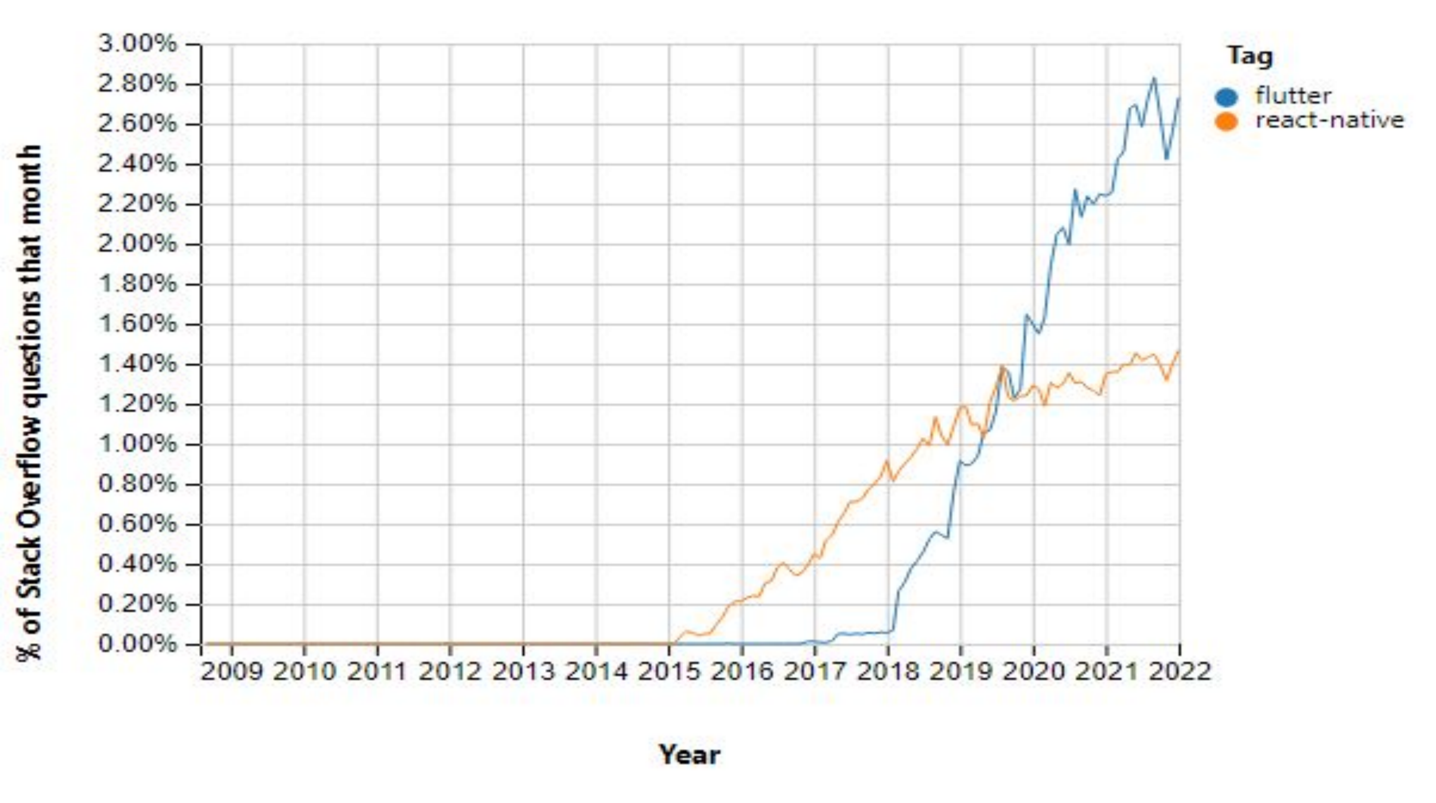
Any developer that plans to launch a mobile app knows how important it is to have versions for both Android and iOS. As of September 2021, Android maintains a firm hold on the mobile market with a 72% market share. But iOS also has a large group of users, along with nearly a 27% market share.

Focusing on building a single native application ignores a large part of the market and limits the potential future success of the application. But developing separate native applications can be time-consuming and costly.

Flutter makes cross-platform development simple. Rather than creating separate code for each platform, developers can take advantage of Flutter’s single codebase.

But Flutter’s suitability for cross-platform development goes beyond code portability. Unlike other cross-platform frameworks such as React Native and Xamarin, Flutter-built user interfaces (UI) are also platform-agnostic because Flutter’s Skia rendering engine does not require any platform-specific UI components.

One of the most important features of Flutter for reducing development time is its stateful hot reload capability. Hot reload, which relies on JIT compilation, speeds the implementation of changes to the code.



Popular Companies Using Flutter & Their Experience

https://flutter.dev/showcase

BMW, Google Pay, eBay, Google Stadia, Tencent, Toyota

Our Learning Process

Dart basics -

Flutter Widgets -

Third-Party Plugins -

Flutter State -

Flutter Navigation -

Local Storage and Persistence -

Native Components -