# **Welk ontwikkelraamwerk is geschikt voor het ontwikkelen van de mobiele applicatie rekening houdend met de vooraf gedefinieerde randvoorwaarden?**

# **Raamwerken**

Mobiele ontwikkeling is tegenwoordig vaak onmisbaar bij het aantrekken van gebruikers of bij het implementeren van de nieuwste functionaliteiten of mogelijkheden. Onder mobiele ontwikkeling wordt vaak aan mobiele telefoons gedacht, echter gaat mobiele ontwikkeling verder dan mobiele telefoons. Hierbij kunnen ook bijvoorbeeld smartwatches, smart armbanden, tablets en convertables vallen. De scope van dit onderzoek zal zich beperken tot mobiele telefoons en tablets die draaien op de twee grootste besturingssystemen Android en iOS.

Voor het ontwikkelen van mobiele applicaties zijn er verschillende raamwerken of technologieën beschikbaar.

Ieder van deze technologieën geeft zijn voor en nadelen. Voor het bepalen van het raamwerk of technologie zijn de mogelijkheden onderverdeeld in 4 categorieën.

### **Native app ontwikkeling**

Ten eerste is het mogelijk om “Native” applicaties te ontwikkelen. Hiermee worden de applicaties bedoeld die zijn ontwikkeld met de software die de makers van de twee besturingssystemen hebben meegeleverd voor de ontwikkeling van de applicaties.

Voordelen:

* Native applicaties zijn geoptimaliseerd en bieden een beter prestatie en gebruikerservaring
* Kunnen gebruik maken van alle native API’s en functies van het besturingssysteem
* Geen dataverbinding nodig

Nadelen:

* Het ontwikkelen vergt expertise van verschillende programmeertalen en ontwikkelomgevingen om voor beide platformen te ontwikkelen.
* Onderhouden van meerdere codebases is complex en tijdrovend.

Applicaties die native ontwikkeld zijn:

* Whatsapp
* Spotify
* Pokemon Go
* Waze

### **Cross platform native applicaties**

Cross platform applicaties zijn ontworpen om op verschillende besturingssystemen en apparaten te werken met een enkele codebase. Verder worden native elementen gebruikt om de user interface op te bouwen en om een zoveel mogelijke native ervaring te geven.

Voordelen:

* Cross-platform applicaties kunnen worden ontwikkeld met een enkele codebase.
* Kunnen op verschillende platforms worden ingezet.

Nadelen:

* Prestatie van cross-platform apps komt niet overeen met die van native applicaties.

Applicaties die crossplatform zijn ontwikkeld:

* Facebook (iOS, Android) - React Native
* Microsoft Teams - React Native
* Outlook - React Native
* Tesla - React Native
* PS app - React Native
* Discord Bloomberg - React Native
* BMW app - Flutter
* Toyota app - Flutter
* Google pay - Flutter
* Nubank - Flutter

### **Hybride mobiele applicaties**

Hybride applicaties zijn zoals crossplatform applicaties geschreven met een code base die werkt op verschillende besturingssystemen en apparaten. Het voornaamste verschil tussen hybride en crossplatform is de opbouw van de gebruikersinterface. Waarbij de gebruikersinterface van hybride apps is geschreven met web technologieën.

Voordelen:

* Hybride apps kunnen op meerdere platforms worden ingezet.
* Native functies en API’s kunnen middels plugins worden benaderd.

Nadelen:

* Prestatie van Hybride apps komt niet overeen met die van native applicaties.

Applicaties die hybride zijn:

* Bobcat - Ionic
* Southwest airlines - Ionic

<https://flutter.dev/showcase>

<https://reactnative.dev/showcase>

<https://stackshare.io/apache-cordova>

<https://ionic.io/customers>

### **Progressive web apps (PWA)**

“Progressive web apps” zijn webapplicaties ontwikkeld om te optimaal te kunnen worden gebruikt op mobiele apparatuur. De applicatie wordt altijd geopend door een webbrowser zonder dat daarvoor installatie nodig is. Hierbij worden geen afzonderlijke versies voor de verschillende besturingssystemen gebruikt.

Voordelen:

* Kunnen rechtstreeks vanuit het web worden geopend waardoor geen installatie nodig is.
* Kunnen op verschillende platforms worden gebruikt.

Nadelen:

* Hebben beperkte toegang tot native API’s en functies.
* Prestatie komt niet overeen met zowel native, hybride als crossplatform applicaties.

## **Overwegingen**

Voor het bepalen van een geschikt framework voor het ontwikkelen van een mobiele applicatie zijn er een aantal factoren die moeten worden overwogen:

* Functionaliteit: De gewenste functionaliteit speelt een belangrijke rol bij de keuze van het ontwikkelplatform. Bepaalde functies zijn alleen beschikbaar bij een native oplossing waarbij andere vormen afvallen.
* Kosten/Tijd: Het ontwikkelen en onderhouden van een applicatie voor verschillende besturingssystemen kost veel tijd en geld. Hierbij moet een overweging worden gemaakt wat belangrijker is.
* Gebruikerservaring: De gebruikerservaring is een belangrijke factor voor het succes van een applicatie. Native applicaties kunnen de beste gebruikerservaring bieden, terwijl native en hybride applicaties wellicht concessies moeten maken op verschillende aspecten.