Mobile development onderzoek

[Inleiding 3](#_Toc82653218)

[Onderzoeksvragen 4](#_Toc82653219)

[SPA vs PWA 5](#_Toc82653220)

[SPA (Single Page Application) 5](#_Toc82653221)

[PWA (Progressive Web App) 5](#_Toc82653222)

[Service workers 5](#_Toc82653223)

[Web App Manifest 5](#_Toc82653224)

[HTTPS 6](#_Toc82653225)

[Overeenkomsten 6](#_Toc82653226)

[Verschillen 6](#_Toc82653227)

[Snelheid 6](#_Toc82653228)

[Kosten 7](#_Toc82653229)

[Wat zijn de verschillen tussen native mobile apps, webapplicatie en hybride applicaties? 8](#_Toc82653230)

[Bronnen 10](#_Toc82653231)

# Inleiding

In dit document beschrijven we welke onderzoeken we hebben uitgevoerd binnen het groepsproject Mobile Development.

# Onderzoeksvragen

Welke opties zijn er voor het maken van de solut website?

Wat zijn de verschillen tussen native mobile apps, web application en hybrid applications?

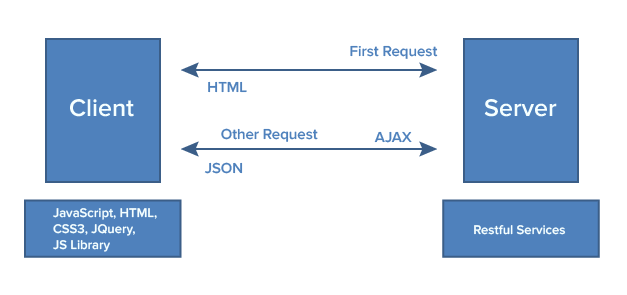
# SPA vs PWA

## SPA (Single Page Application)

SPA is een website waarvan de huidige pagina dynamisch wordt bijgewerkt in plaats van volledig te worden gedownload van een server. Met andere woorden, alle benodigde applicatiecode voor de webapplicatie (HTML, CSS, JavaScript) hoeft maar een keer te worden geladen.

Terwijl de gebruiker door de webtoepassing navigeert, worden alle inhoud en elementen die moeten worden bijgewerkt, opgehaald en opnieuw weergegeven zonder dat de gebruiker de browser opnieuw hoeft te laden.

Dit bespaart de traditionele ‘’rondreis’ tussen de browser en de server, waardoor snellere interacties en een betere gebruikerstevredenheid mogelijk is



Er zijn veel bekende SPA’s die vrijwel iedereen gebruikt:

* Facebook
* Gmail
* Linkedin
* Netflix
* Google Maps

## PWA (Progressive Web App)

In tegenstelling tot een SPA is een PWA meer een set richtlijnen en checklists dan een specifieke architectuur. PWA’s zijn bekend en wijken af van de SPA’s door verschillende kenmerken, zoals:

### Service workers

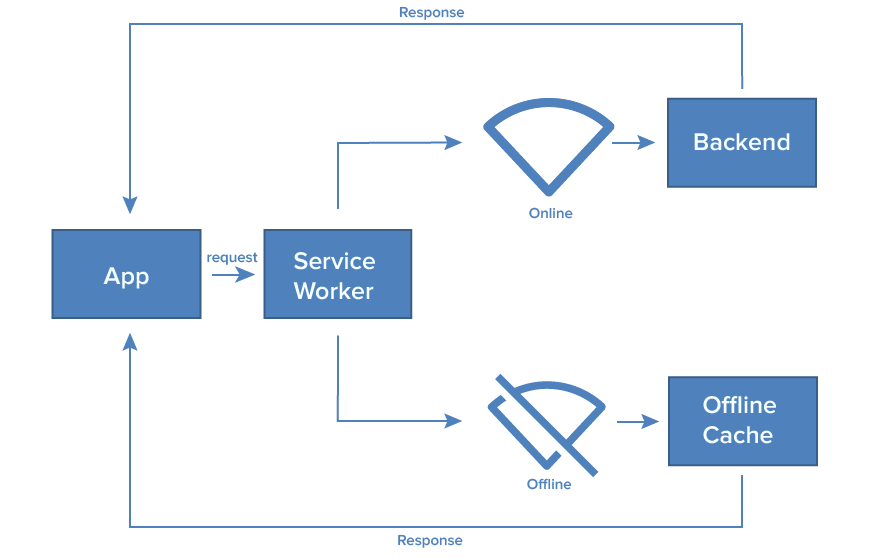
Service workers vormen de technische basis voor veel van de prominente functies van PWA namelijk offline beschikbaarheid en pushmeldingen.

### Web App Manifest

Een JSON-bestand dat de benodigde metadata voor de PWA bevat. Web app manifesten kunnen automatisch worden gegenereerd met behulp van de PWA-manifestgenerator.

### HTTPS

PWA’s worden altijd geleverd vanuit een veilige bron.



Service workers werken als een proxy tussen de browser en het netwerk en zijn verantwoordelijk voor het cachen van de website assets en het onderscheppen van netwerk verzoeken wat leidt tot de offline beschikbaarheid van de website en verschillende andere functies, waaronder pushmeldingen.

In de achtergrond helpen service workers ook enorm mee. Naast het aanbieden van offline mogelijkheden, zijn ze ook verantwoordelijk voor veel van de zware berekeningen, wat de belangrijkste reden is waarom de meeste PWA-websites zo efficiënt en snel ladend zijn.

## Overeenkomsten

Beide zijn het nieuwe en revolutionaire technologieën. Daarom bieden deze twee architecturen de gebruiker een app-achtige, boeiende ervaring die enorm verschilt van de traditionele websites.

Op eerste aanzicht lijken ze veel op elkaar, mar de onderliggende processen die al deze ervaringen mogelijk maken, zijn echter niet hetzelfde.

## Verschillen

Vaak wordt gezegd dat een PWA bijna altijd een SPA is, wat onjuist is. In een PWA wordt de inhoud van de website niet per se in de browser weergegeven. Ook mag een PWA door de server gegenereerde pagina’s opvragen.

### Snelheid

In termen van snelheid lijken beide architecturen op elkaar, omdat ze beide geavanceerde technologie zijn. Echter gaat de overhand uit naar PWA.

Dit komt omdat hoewel zowel SPA’s en PWA’s JavaScript gebruiken voor hun ontwikkelingen kunnen PWA zijn service workers gebruiken om delen van het werk weer te geven. Oftewel veel van de JavaScript overhead die SPA’s toevoegen kunnen PWA’s overlaten aan de service workers waardoor de JavaScript ‘impact’ aanzienlijk worde verminderd.

### Kosten

Op het gebied van kosten verliest PWA van SPA, omdat PWA beschikt over de allernieuwste technologie en als bijgevolg een meerprijs met zich mee heeft. De ontwikkeling van PWA’s is wel vergelijkbaar met SPA’s.

De kosten van PWA’s en SPA’s kunnen variëren:

PWA’s: $2000 - $20.000

SPA’s: $1500 - $12.000

Er zijn veel factoren die van invloed zijn op de bouwkosten, zoals het tarief van de ontwikkelaar, de complexiteit van de klus, de voorkeuren en aanpassingen van de product owner. Deze moeten allemaal in overweging genomen worden bij het maken van een kostenplaatje.

# Wat zijn de verschillen tussen native mobile apps, webapplicatie en hybride applicaties?

**Native mobile applicatie**

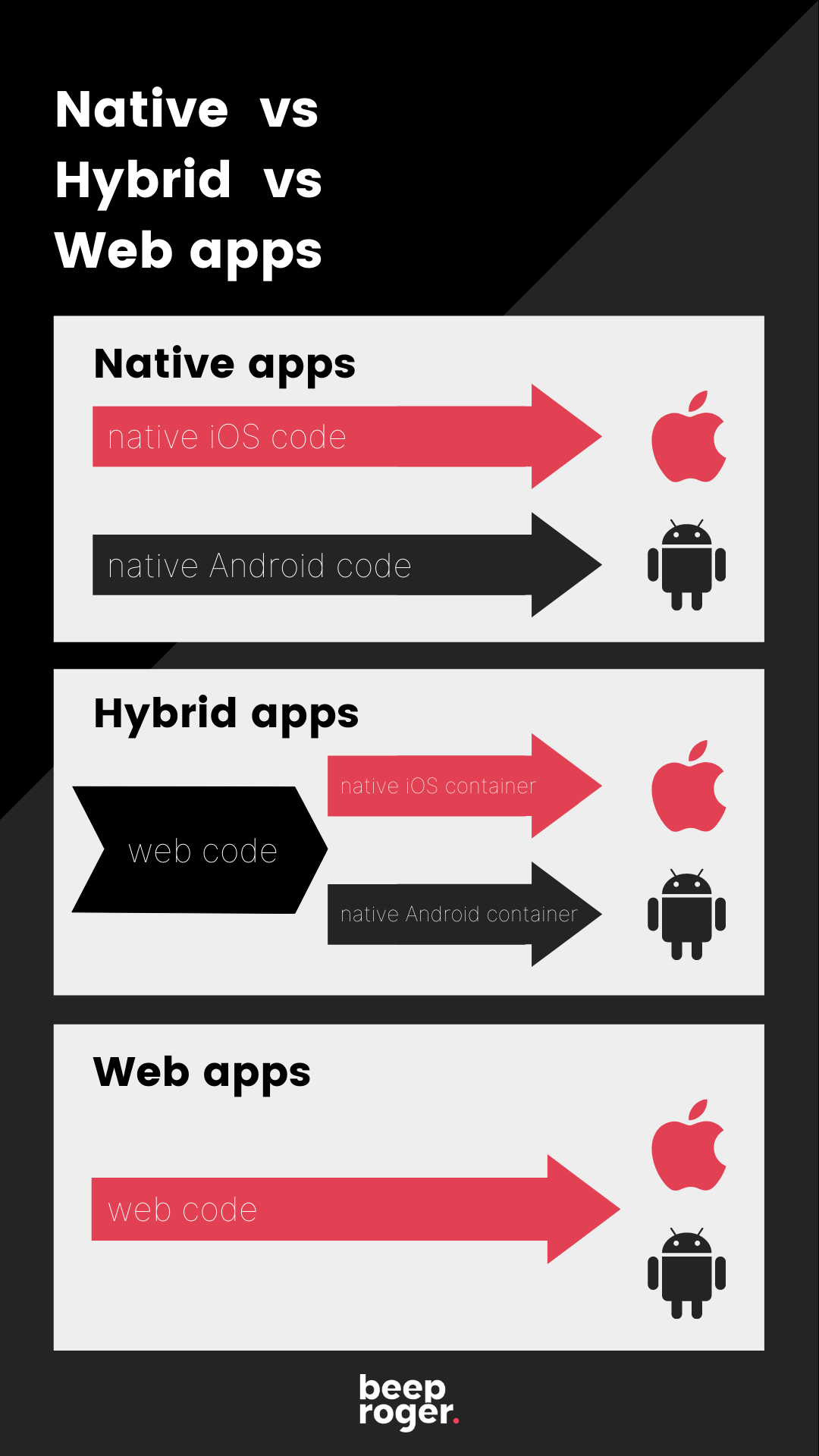
Een native mobile app wordt voor elk besturingssysteem apart gebouwd, een app voor iOS moet dus apart gebouwd worden dan een app voor Android. Een native app moet je van de app store downloaden anders kan je er niet bij. Een native app maken vanaf niks geeft je de meeste flexibiliteit vergeleken met alle andere ontwikkelmethodes. Je kan alles maken als je maar weet hoe. Dit duurt wel het langst van alle ontwikkelmethodes. Wanneer een app gebouwd is moet deze worden gereviewed door google play of de apple store voordat deze gelanceerd wordt.

**Webapplicatie**

Een webapplicatie hoef je niet van de app store te downloaden maar kun je bereiken door een URL in te voeren in je webbrowser. Webapplicaties geven mobiele gebruikers een betere ervaring zonder alle extra kosten van een Native mobile applicatie. Dit is een goede oplossing als je geen push notificaties en offlinesupport nodig hebt.

**Hybride applicatie**

Een hybride applicatie is een combinatie van de twee. Deze wordt als een webapplicatie ontworpen maar kan worden gedownload via de app store. Dit wordt een keer gecodeerd en kan worden gedeployed op meerdere soorten platformen. Dit zal wel een klein beetje slomer zijn dan een native app maar is niet heel erg merkbaar met de internetsnelheden van vandaag.



# Bronnen

Griffith, C. (2021, 9 september). *What is Hybrid App Development?* Geraadpleegd op 16 september 2021, van <https://ionic.io/resources/articles/what-is-hybrid-app-development>

Saltis, S. (2021, 22 juni). *StackPath*. Geraadpleegd op 16 september 2021, van <https://www.coredna.com/blogs/progressive-web-app>

*Native apps vs Web apps vs Hybride apps - wat is de beste keuze?* (2021, 28 juli). Geraadpleegd op 16 september 2021, van <https://beeproger.com/blog/native-apps-vs-web-apps-vs-hybride-apps-is-beste-keuze/>