A picture containing shape

Description automatically generated

**Voordelen van PWA’s voor administratiesoftware**

**AON1**

Emir Ozdemir

Fabio Ruffolo

Jonathan Vuurstaek

Maxim De Cuyper

Victor Tuerlinckx

Projectomschrijving

De opdracht van het researchproject is het bouwen van een applicatie om het proces van de stage-administratie te stroomlijnen. De administratie wordt hoofdzakelijk gedaan met Google Forms en daarnaast doet Gmail dienst als communicatiemiddel. Momenteel bestaat het takenpakket uit heel wat manueel werk door stageverantwoordelijke Marijke Willems. Het doel is een webapplicatie te maken die beide diensten integreert en bijgevolg op één plaats samenbrengt om het administratieproces te versnellen.

Het onderzoek van dit project behandelt PWA’s of *Progressive Web Applications*. Hierbij wordt verduidelijkt wat PWA’s zijn, hoe deze werken, waarin ze verschillen van gewone webapplicaties en welke voordelen ze bieden. De antwoorden op deze deelvragen worden verkregen aan de hand van een literatuurstudie. Vervolgens wordt de link met de projectcase gelegd in de hoofdvraag: “Op welke manier biedt een *Progressive Web Application* voordelen voor de stage-administratiesoftware?”

Voor de uitwerking van deze case wordt er gebruikgemaakt van Google Sheets en Google Mail aan de hand van API’s. De frontend wordt gebouwd met HTML, CSS, JavaScript en Vue. Java en meer bepaald Springboot wordt gebruikt in de backend.

Inhoudsopgave

[Projectomschrijving ii](#_Toc99910738)

[Inhoudsopgave iii](#_Toc99910739)

[Lijst van gebruikte figuren iv](#_Toc99910740)

[Lijst van gebruikte tabellen v](#_Toc99910741)

[Lijst van gebruikte afkortingen vi](#_Toc99910742)

[1 Onderzoeksvraag en hypothese 1](#_Toc99910743)

[2 Onderzoeksmethode 2](#_Toc99910744)

[1 Literatuurstudie 3](#_Toc99910745)

[1.1 Wat is een PWA? 3](#_Toc99910746)

[1.2 Hoe werkt een PWA? 4](#_Toc99910747)

[1.4 Wat zijn de verschillen tussen PWA’s en gewone applicaties? 4](#_Toc99910748)

[1.5 Wat zijn de voor- en nadelen van PWA’s? 5](#_Toc99910749)

[1.6 Wat zijn de voordelen van PWA’s voor de stage-applicatie? 6](#_Toc99910750)

[2 Uitvoering 7](#_Toc99910751)

[2.1 Hoofdstuk 7](#_Toc99910752)

[2.1.1 Hoofdstuk 7](#_Toc99910753)

[2.2 Hoofdstuk 7](#_Toc99910754)

[3 Conclusie 8](#_Toc99910755)

[Bibliografie 9](#_Toc99910756)

[Bijlagen 10](#_Toc99910757)

Lijst van gebruikte figuren

Lijst van gebruikte tabellen

Lijst van gebruikte afkortingen

|  |  |
| --- | --- |
| PWA | *Progressive web app* |
| PDF | *Portable Document Format* |

# Onderzoeksvraag en hypothese START

Als onderwerp voor het researchproject is er gekozen voor het bouwen van een stagebeheerapplicatie. Dit programma heeft als doel de verwerking van stageopdrachten te vereenvoudigen en te centraliseren.

In het oude systeem dient een bedrijf een stagevoorstel in aan de hand van een lijvig Google-formulier. Zodra deze info correct in de achterliggende lijst terechtkomt, stuurt de stageverantwoordelijke een bevestigingsmail naar het corresponderende bedrijf en wijst vervolgens een of meerdere reviewers toe aan de betreffende STformopdracht. Ook de toewijzing gebeurt via een mail met daarin de opgave als PDF. Na het beoordelen van de opdracht wordt deze teruggestuurd naar de stageverantwoordelijke die op basis van de bijgevoegde feedback een van de volgende acties onderneemt: de opdracht dient bijgewerkt te worden op bepaalde aspecten en het bedrijf wordt hierover via mail ingelicht, de opdracht is in orde en het bedrijf ontvangt hierover een goedkeuringsmail of de opdracht voldoet niet en het bedrijf wordt op de hoogte gesteld van de afkeuring. Indien het bedrijf de opgave aanpast, kan deze door de stageverantwoordelijke worden aangepast in de lijst en belandt deze, al dan niet, opnieuw in de reviewerfase.

Het gehele proces uit de voorgaande alinea omvat veel werk met wisselen tussen verschillende applicaties en manuele arbeid om info aan te passen. De taak van dit project bestaat erin het proces samen te brengen in één applicatie die alle functionaliteiten omvat. Hierbij wordt er ook nog gevraagd om onderzoek te doen naar PWA’s of voluit Progressive Web Applications. Tijdens dit onderzoek wordt er een antwoord gezocht op de hoofdvraag van dit project: Welke voordelen kunnen PWA’s bieden voor de stagebeheersoftware? De deelvragen die hierbij gesteld en beantwoord worden, zijn de volgende:

* Wat is een PWA?
* Hoe werkt een PWA?
* Wat zijn de verschillen tussen PWA’s en gewone applicaties?
* Wat zijn de voor- en nadelen van PWA’s?
* Wat zijn de voordelen van PWA’s voor de stage-applicatie?

Goed opgesteld!!

# Onderzoeksmethode

Voor de uitwerking van deze paper is er gekozen voor een combinatie van literatuurstudie en praktische demo’s.

Op deze manier worden de PWA’s eerst onder de microscoop gelegd en nabij bestudeerd om ze nadien in de praktijk toe te passen. Het doel van de literatuurstudie bestaat erin de deelvragen van dit onderzoek te beantwoorden en een globale kennis op te doen over PWA’s.

Met deze verworven inzichten worden dan enkele praktische demo’s opgezet om te testen hoe bruikbaar *Progressive Web Applications* zijn voor de stagesoftware. Het is echter niet de bedoeling dat deze demo’s nadien ook effectief worden opgenomen in de applicatie. Zij dienen louter als *proof of concept* en bevestiging van de vooraf gestelde hypothese.

# Literatuurstudie

## Wat is een PWA?

Het web is een geweldig platform, het is overal te vinden op toestellen zoals: smartphones, laptops en zelfs smartwatches. Het gebruiksgerichte securitymodel en het feit dat het niet geïmplementeerd of gecontroleerd wordt door één bedrijf maakt het web een uniek platform voor Grdial! applicaties op te ontwikkelen. Samen met de bereikbaarheid, de mogelijkheid om te zoeken en te delen met wie en waar dan ook. StinfEen website is ook altijd up-to-date, het kan ook iedereen, waar en wanneer bereiken met een en dezelfde applicatie.

Platform specifieke applicaties staan bekend als zeer veelzijdig en betrouwbaar te zijn. Ze werken netwerkonafhankelijk. Ze starten in hun eigen standalone ervaring. Ze kunnen lokale bestanden lezen en schrijven, hardware benaderen die zijn aangesloten via USB, serieel of bluetooth, en zelfs communiceren met gegevens die zijn opgeslagen op uw toestel, zoals contactpersonen. In deze applicaties kunt u dingen doen als STinf foto's maken, op het startscherm zien welke nummers er worden afgespeeld of het afspelen van nummers regelen terwijl u in een andere app bent. Platform specifieke applicaties voelen aan als onderdeel van het apparaat waarop ze draaien.

*Progressive Web Apps* worden gebouwd en verbeterd met moderne Apis SPom verbeterde eigenschappen, betrouwbaarheid en installeerbaarheid te bieden terwijl ze iedereen, overal en op elk apparaat bereiken met een enkele codebase.

*PWA’s* zijn zo ontwikkeld dat ze: GRintcapabel, betrouwbaar en installeerbaar zijn. Deze drie pillaren vormt ze GRowin een ervaring Stinf dat GRrefvoelt zoals een platform specifieke applicatie. Zin klopt niet + te STinf!

*Progressive Web Apps* zijn eigenlijk gewoon webapplicaties dat GRrefdoor gebruik te maken van *progressive enhancement* nieuwe mogelijkheden beschikbaar maakt Grow voor moderne browsers. [1]

## Hoe werkt een PWA?

*Progressive Web Apps* (PWA’s) maken gebruik van verschillende technologieën om de eindgebruiker het gevoel van een *native* applicatie aan te bieden. Een van de eerste stappen om een webapplicatie om te vormen naar een *Progressive Web App*, is het voorzien van een manifest-bestand.

Een manifest-bestand is een JSON-bestand dat de browser vertelt Stinfhoe de webapplicatie zich moet gedragen na installatie op een mobiel apparaat van een gebruiker. Hoewel dit bestand enkel algemene informatie van een webapplicatie bevat, speelt het echter een belangrijke rol bij het laten lijken en aanvoelen van de webapplicatie als een *native* applicatie.

Een typisch manifest-bestand bevat informatie over de naam van de app, de pictogrammen die deze moet gebruiken, de URL waarnaar gesurft moet worden wanneer de app opstart en nog veel meer. Zodra aan een bepaald criterium is voldaan, zullen de meeste browsers de bezoekers automatisch voorstellen om de applicatie te installeren.

Het belangrijkste element voor de werking van PWA’s zijn de *service workers*. In essentie is een *service worker* een JavaScriptbestand dat wordt toegevoegd aan je code-base. Het bevindt zich tussen de server en de browser en voegt een nieuwe laag toe om applicatie functies na te bootsen. Een *service worker* voert zijn functies uit zonder nood te hebben aan een geopende webpagina of interactie met de gebruiker. Dit maakt applicatiewaardige functies mogelijk zoals mobiele meldingen, het vastleggen van gebruikersacties terwijl ze offline zijn en ze afleveren terwijl ze online zijn, toegang tot de interne schijf, contactlijst, camera van het apparaat, enzovoort…

Tot slot is de compatibiliteit met de browsers een ander belangrijk onderdeel voor de werking van *Progressive Web Apps*. *Service workers* zijn het brandpunt van PWA’s, maar niet alle browsers ondersteunen deze. Safari bijvoorbeeld blijft vaak achter en wordt dikwijls gezien als de *Internet Explorer* van PWA’s. Een gebrek aan ondersteuning voor specifieke functies belemmert het gebruik van *Progressive Web Applications* echter niet omdat het websites zijn. Bijgevolg zullen ze nog steeds werken in alle browsers, hoewel dit soms zonder de volledige set van functies zal zijn. [2], [3]

## Wat zijn de verschillen tussen PWA’s en gewone applicaties?

Naast *Progressive Web Apps* zijn er nog andere soorten applicaties, zoals bijvoorbeeld *native app*, *hybrid app* en een *responsive* website.

Een *native* applicatie wordt ontworpen voor een specifiek besturingssysteem en is bijgevolg geschreven in een specifieke programmeertaal voor dat besturingssysteem.

Een concreet verschil tussen een *progressive* en een *native* applicatie is dat bij de laatstgenoemde verschillende, compatibele versies voor elk besturingssysteem moeten worden gemaakt. Een *progressive* applicatie wordt één keer geschreven aan de hand van gemakkelijk toegankelijke *frameworks* en is beschikbaar voor alle verschillende besturingssystemen.

In tegenstelling tot een *native* applicatie is het voor de gebruiker mogelijk om *Progressive Web Apps* te installeren op bijvoorbeeld hun bureaublad. Deze mogelijkheid wordt in het algemeen positief ervaren door de eindgebruiker omdat het tal van voordelen met zich meebrengt zoals toegankelijkheid en snelheid. [4]

## Wat zijn de voor- en nadelen van PWA’s?

*Progressive Web Apps* hebben verschillende voordelen voor zowel de developers als de eindgebruiker. Zo ervaart de eindgebruiker een snellere service die gerealiseerd wordt door het feit dat *Progressive Web Apps* data kunnen opslaan in het cachegeheugen. Dit geheugen is een plaats waar data tijdelijk wordt bewaard en die veel sneller toegankelijker is in vergelijking met andere opslagplaatsen.

Een tweede voordeel van *Progressive Web Apps* is hun toegankelijkheid. Ze zijn verkrijgbaar op bijna alle apparaten en na een simpele installatie blijft de applicatie ten alle tijden beschikbaar op het bureaublad. Door middel van een simpele dubbelklik op het logo opent de applicatie zich en kan de gebruiker ermee aan de slag gaan.

Het onderhoud van *Progressive Web Apps* gebeurt automatisch. Er wordt geen verzoek naar de eindgebruiker verstuurd. Indien er een nieuwe versie van de applicatie beschikbaar is, kan de developer de vernieuwde versie live zetten op de server. De applicatie staat hiermee in contact en gaat automatisch de update installeren op een manier waarop de eindgebruiker er zo min mogelijk last van ondervindt. Dankzij deze technologie maakt de eindgebruiker ten alle tijden gebruik van de beste functionaliteiten en de nieuwste services.

PWA’s bevatten niet enkel voordelen voor de eindgebruiker, maar ook voor de developer. Over het algemeen zijn de ontwikkelingskosten van *Progressive Web Apps* lager dan die van een *native* webapplicatie. Voor niet-*Progressive Web Apps* dienen er echter 3 verschillende applicaties ontwikkeld te worden, namelijk: een website, een iOS-app en een Android-App. Hiervoor betaal je drie keer de ontwikkelingskosten en de hostkosten.

Daarnaast zijn *Progressive Web Apps* ook gemakkelijker te distribueren. Hoewel reguliere applicaties nood hebben aan Google Play of de App Store, kunnen *Progressive Web Apps* eenvoudig online gevonden worden en eenvoudig worden toegevoegd aan het bureaublad zonder tussenkomst van een medium.

Er zijn ook echter enkele nadelen. Zo zijn verschillende functionaliteiten hier niet toegankelijk, maarwel bij een *native* applicatie. Enkele voorbeelden hiervan zijn Bluetooth, GPS SPen camera’s. Bovendien ondersteunen niet alle mobiele browsers *Progressive Web Apps*. De functionaliteiten die door PWA’s niet gebruikt kunnen worden, zijn inmiddels standaard ingeburgerd bij de eindgebruiker. Hierdoor dient een weloverwogen keuze gemaakt te worden afhankelijk van het doel van de applicatie. [4]

stop

## Wat zijn de voordelen van PWA’s voor de stage-applicatie?

Zoals eerder besproken zijn er veel voordelen die een *Progressive Web App* kan bieden. Maar welke voordelen zijn nu echt van toepassing op de stagesoftware?

Het eerste voordeel dat van toepassing is op de stagesoftware is dat de klant de software niet moet installeren op het apparaat waarop hij het wil gebruiken. De klant kan simpelweg met de gegeven URL de Web App overal installeren en gebruiken wanneer hij maar wil.

Het tweede grote voordeel is de gebruiksvriendelijkheid van een PWA. Een PWA kan als snelkoppeling worden toegevoegd op het startscherm van het apparaat van de gebruiker waardoor deze makkelijk en snel opgestart kan worden.

Het derde voordeel dat zeer belangrijk is voor de stagesoftware is dat een PWA offline gebruikt kan worden. De gebruiker kan hierdoor altijd gebruik maken van de applicatie met de data die opgeslagen is. Wanneer dat de gebruiker weer online komt, zal al de data up-to-date gebracht worden. In het geval van de stagesoftware betekent dit dat de wijzigingen die aangebracht werden, doorgevoerd zullen worden naar de database, nieuwe aanvragen en mails binnen zullen komen en mails die in de wachtrij gezet werden verzonden zullen worden. De klant heeft aangegeven dat hij dit belangrijk vindt.

Ten slotte wordt bij een PWA de pagina van een applicatie in één keer volledig ingeladen en zijn de grootte van *requests* kleiner. Dit zorgt voor een enorme performancewinst. Hierdoor moet de gebruiker veel minder lang wachten en dat vergroot de gebruiksvriendelijkheid. [5], [6]

# Uitvoering

## Hoofdstuk

### Hoofdstuk

## Hoofdstuk

# Conclusie

Bibliografie

|  |  |
| --- | --- |
| [1] | „What are Progressive Web Apps?,” [Online]. Available: https://web.dev/what-are-pwas/. |
| [2] | „Progressive web apps (PWAs) | MDN,” [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/Progressive\_web\_apps/Introduction. |
| [3] | „Introduction to Progressive Web App Architectures | Google Developers,” [Online]. Available: https://developers.google.com/web/ilt/pwa/introduction-to-progressive-web-app-architectures. |
| [4] | „Progressive Web Apps,” [Online]. Available: https://web.dev/progressive-web-apps/. |
| [5] | „PWA (progressive web apps): de voordelen en nadelen op een rij,” [Online]. Available: https://www.d-tt.nl/artikelen/pwa-progressive-web-apps-voordelen-nadelen. |
| [6] | „Wat is een PWA en is dit iets voor mijn organisatie?,” 01 10 2021. [Online]. Available: https://www.frankwatching.com/archive/2020/03/10/pwa-progressive-web-app-voordelen-nadelen/. |

Bijlagen