## Inleiding

In deze sectie van het onderzoek zal er gekeken welke functies het web aanbiedt en dus ook door PWA’s.

Dit onderzoek werd gevoerd met behulp van de website whatwebcando.today en caniuse.com. Als er meer informatie nodig was over de web API’s werd deze gevonden op developers.google.com of op developer.mozilla.org.

Er werd gekeken op welke platformen bepaalde functies wel en niet werkten. De volgende platformen werden onderzocht

* Microsoft edge versie 80 op Windows 10
* Mozilla firefox versie 73.0 op Windows 10
* Google chrome versie 79.0 op Windows 10
* Safari (desktop), kan niet geïnstalleerd worden op windows, alle gegevens voor safari komen van caniuse.com.
* Google chrome versie 80 op Android 10 op een OnePlus 6
* Safari (mobiel) versie 13 op IOS 13 op een Iphone SE

Het onderzoek werd opgesplitst in verschillende delen:

* Media
* Verbinding
* Toestel kenmerken
* Native gedrag
* Besturingssysteem
* Input
* User experience
* Locatie en positionering
* Scherm en output

## Onderzoek

**Media**

Video met audio – Om video en audio vast te leggen met een webapplicatie wordt er gemaakt van de Media Capture API.

Voor het starten van opnemen van een fragment zal het systeem toegang vragen tot de camera en de microfoon.

Via het window object kan er toegang verkregen worden tot het navigator object. Dit is een object dat informatie en methode bevat over de huidige applicatie.

Een van deze (read-only) properties is ‘mediaDevices’, dit geeft een object terug dat toegang heeft tot alle verbonden geconnecteerde media input toestellen. Dit kunnen camera’s, microfoons of apparatuur om het scherm te delen zijn.

Op dit object kan volgende methode uitgevoerd worden: ‘getUserMedia(constraints)’.

Dit zal een promise teruggeven dat ‘resolves’ naar een MediaStream die uitgegeven kan worden op een canvas. Als de user geen toegang geeft of er is geen camera aanwezig zal er een error gegooid worden.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Ja |

Foto’s (met controlls) – Er is een API voorzien voor het nemen van foto’s binnen de browser. Dit heeft het voordeel dat het gedrag van de camera kan aangepast worden in de code. Instellingen zoals witbalans en shutterspeed kunnen aangepast worden.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Nee | Ja | Nee | Ja | Nee |

Deze beperkte beschikbaarheid geldt enkel voor het nemen van een foto zonder de website te verlaten. Het is op alle platformen mogelijk om automatisch de app te verlaten en een foto te nemen in de native camera app, deze foto wordt dan automatisch gebruikt in de website.

<https://developers.google.com/web/fundamentals/media/capturing-images>

Geluid opname – de mediarecorder API die meerdere browsers aanbieden is een manier om eenvoudig geluidsfragmenten op te nemen en te importeren in een webapplicatie. Dit wordt op een vergelijkbare manier gedaan als bij het opnemen van video. Opnieuw wordt het ‘userMedia’ object gebruikt.

Helaas is de ondersteuning voor deze web-API beperkt. In de toekomst zal deze functie waarschijnlijk ook beschikbaar worden op Apple toestellen. Dit wordt in de volgende versie van safari (safari 14) voor desktop verwacht. Voor safari voor IOS bestaat deze functie al maar is het nog een experimentele functie die de gebruiker zelf moet activeren.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Nee | Ja | Nee |

Gelukkig is er een alternatief voorzien met HTML5 tags. Dit is een methode die voor alle platformen zal werken maar niet op dezelfde manier.

<input type="file" accept="audio/\*" capture>

Er wordt gebruik gemaakt van een gewoon input veld waar de gebruiker een bestand kan uploaden. Door het accept attribuut wordt duidelijk gemaakt dat enkel audio fragmenten geüpload kunnen worden. Het capture attribuut zorgt ervoor, indien mogelijk, dat de gebruiker een audiofragment kan opnemen in de default geluidsopname app, dit fragment wordt dan automatisch geïmporteerd in de webapplicatie. Dit is enkel mogelijk op mobiele toestellen en dus niet in desktopbrowsers.

<https://developers.google.com/web/fundamentals/media/recording-audio>

Real-time communicatie/ stream – ‘real-time communication in the web’ of WebRTC is een verzameling van API’s die het verzenden en ontvangen van real-time video en audio mogelijk maakt, zonder afhankelijk te zijn van een gecentraliseerde server. Deze server is echter wel nodig om een connectie tot stand te brengen. Eens deze connectie er is, is er een peer-to-peer verbinding.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Ja |

**Verbinding**

Bleutooth – Met de Web Bluetooth API kan er vanuit de browser verbinding gemaakt worden met bluetooth toestellen. Dit kan handig zijn voor applicaties die een bluetooth toestel ondersteunen zoals hartslag meters.

Er kan enkel gebruik gemaakt worden van de API als de website een SSL-certificaat heeft.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | Nee | Ja | Nee |

USB – Verkopers van toestellen met usb kunnen nu gebruik maken van de WebUSB API. Bij het verbinden van een USB-toestel kan er automatisch een website geopend worden waarmee het toestel kan interageren.

Dit kan een interessant zijn voor toestellen die een eenmalige set-up nodig hebben. Met deze technologie kan vermeden worden dat er overbodige software moet geïnstalleerd worden op het toestel van de gebruiker.

Dit is echter enkel mogelijk met een beperkt aantal browsers en er moet een HTTPS-verbinding zijn.

<https://developers.google.com/web/updates/2016/03/access-usb-devices-on-the-web>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | Nee | Ja | Nee |

NFC – Near field communication of NFC is een technologie om een kleine hoeveelheid informatie uit te wisselen over een kleine afstand (max 20cm). NFC wordt gebruikt om draadloze betalingen uit te voeren moet een betaalkaart of met een gsm.

Bron 20cm: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.392.7860&rep=rep1&type=pdf>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Nee | Nee | Nee | Nee | Nee |

Dit is een functie met veel mogelijkheden die helaas niet beschikbaar is voor webapplicaties.

Er bestaat echter wel een API om gebruik te kunnen maken van NFC, maar dit is een experimentele functie. Dit betekent dat de eindgebruiker dit nog niet kan gebruiken.

**Toestel kenmerken**

Netwerkinformatie – De Network information API voorziet informatie over het type netwerkverbinding de gebruiker momenteel bezit. Deze informatie bevat het connectietype (2g, 3g, 4g) en wat de maximale downloadsnelheid is van deze verbinding.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | Nee | Ja | Nee |

Online status – dit is een eenvoudige eigenschap die kan opgeroepen worden op het navigator object. Deze eigenschap is een boolean die true zal zijn als de gebruiker een connectie heeft met het internet. Deze informatie kan interessant zijn bij het ontwikkelen van een PWA met offline functionaliteiten.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Ja |

Vibratie motor – De Vibration API zorgt ervoor dat de vibratiemotor kan aangesproken worden vanuit de een webapplicatie. Dit kan voor een meer native gevoel zorgen. Niet alle toestellen beschikken over een vibratiemotor, deze functie mag dus enkel als ondersteuning van iets anders dienen.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Nee | Ja | Nee |

Batterij status – Aan de hand van de Battery Status API kan er informatie over de batterij van het toestel verkregen worden.

Volgende informatie kan verkregen worden

* Aan het opladen
* Batterijpercentage
* (Bij opladen) tijd tot volladen
* (Bij niet opladen) tijd tot batterij leeg

Aan de hand van deze API kunnen er ook evenement uitgevoerd worden op basis van een actie op de batterij. Er kan bijvoorbeeld een functie uitgevoerd worden als de gebruiker zijn toestel met een energiebron verbindt.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | Nee | Ja | Nee |

Toestel geheugen – de Device Memory API geeft informatie over het RAM-geheugen van het toestel van de gebruiker. Dit kan interessant zijn voor het laden van een eventuele lichtere versie van een website.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | Nee | Ja | Nee |

**Native gedrag**

Lokale notificaties – lokale notificaties zijn beschikbaar via de Notifciations API. Lokale notificaties zijn notificaties die geen internet of server nodig hebben. Deze kunnen gepland worden bij het laden van de website. Ze worden dus lokaal geactiveerd.

Binnen lokale notificaties kan er ook onderscheid gemaakt worden tussen twee soorten

* Persistent notifications – Dit zij notificaties waarbij het tabblad niet meer geopend moet zijn om een notificatie te ontvangen. Om dit te verkregen moet er een actieve service worker geregistreerd zijn.
* Non-persistent notifications – bij deze notificaties moet het tabblad wel nog geopend zijn, Hier is geen service worker voor nodig.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Nee |

Push messages – Push notificaties maken ook gebruik van de Notifications API. Het verschi met lokale notificaties is dat deze notificaties geactiveerd worden vanop een server. Als de admin wil dat er notificatie verzonden wordt, kan hij vanop afstand, een notificatie versturen naar alle gebruikers.

Om notificaties te ontvangen van een server moet er ook nog gebruik gemaakt worden van de Push API.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Nee | Ja | Nee |

Home screen installatie – Door het toevoegen van een web app manifest kan je de browser duidelijk maken hoe een applicatie er moet uitzien als het toegevoegd wordt aan het home screen.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Nee | Ja | Ja |

Voorgrond detectie – De Page Visibility Detection API voorziet een methode waarmee een applicatie weet als het in de voorgrond gebruikt wordt of niet.

Dit kan een handige functie zijn voor meerdere redenen. Het laden van resource-intensieve animaties is nodig als de gebruiker op dat moment niet op de pagina is.

Ook wordt het gebruikt om adverteerders te garanderen dat gebruikers een bepaalde tijd effectief naar de advertentie kijken op een website.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | ja |

Toestemmingen (permissions) – Om gebruik te maken van hardware functies van een toestel is vaak, om privacy redenen, de toestemming van de gebruiker nodig. Hiervoor is de permissions API ontwikkeld. Er kan toestemming gevraagd worden op een gelijkaardige manier voor verschillende functies.

Functies waarvoor toestemming gevraagd kan worden:

* Locatie
* Notificaties
* Push-notificaties
* Midi (musical instrument digital interface)
* klembord
* Camera
* Microfoon
* Achtergrond synchronisatie
* Licht sensor
* versnellingsmeter
* Gyroscoop
* magneet sensor
* betalingen

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | nee | Ja | nee |

**Besturingssysteem**

Offline opslag – er zijn verschillende technologieën om data offline op te slaan.

* Web storage – De meeste eenvoudige manier om data op te slaan. Er kunnen key-value paren opgeslagen worden in het localstorage of in het sessionStorage.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | ja | Ja | ja |

* indexedDB – een API voor het opslaan van grote hoeveelheden gestructureerde data op het toestel van de eindgebruiker. De data kan snel gelezen worden omdat er indexen gebruikt worden.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | ja | Ja | ja |

* cacheAPI – gespecialiseerd in het opslaan van request/ request paren. Dit is heel handig in samenwerking met een serviceworker. API-calls kunnen opgeslagen worden voor offline gebruik.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | ja | Ja | nee |

* storage API – data die is opgeslagen in een van vorige methoden kan eenvoudig verwijderd worden door de browser. Met de storage API kan data, met de gebruikers toegang, opgeslagen worden op het systeem voor een langere periode

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | ja | Ja | ja |

Bestanden toegang – toegang tot het bestandssysteem, de gebruiker kan bestanden uploaden met de file API.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | nee | Ja | nee |

Contacten – De contacten die opgeslagen staan op het systeem van de gebruiker kunnen geïmporteerd worden in een webapplicatie met de Contacts API.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Nee | Nee | nee | Beperkt \* | nee |

\* Op het moment van schrijven is dit een nieuwe en experimentele functie die enkel werkt op   
Android 10. Verdere ondersteuning is nog onbekend.

Sms – met de SMS receiver API kan er gekeken worden naar inkomende sms’en. Dit kan handig zijn bij het ontvangen van een verificatiecode.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Nee | Nee | nee | Beperkt \* | nee |

\*Op het moment van schrijven is dit een nieuwe en experimentele functie die enkel werkt op Android 10. Verdere ondersteuning is nog onbekend.

Taak planning – De task scheduler API kan er voor zorgen dat taken zoals alarmen, herinneringen en gelijkaardige taken kunnen ingepland worden in het systeem. Deze API is slechts een voorstel en heeft dus nog geen ondersteuning.

**Input**

Touch gebaren – HTML5 voegt aan de reeds bestaande input toestellen nu ook touch-controlls toe. Dit is belangrijk om een applicatie intuïtief te laten werken. Het is logisch dat Safari op desktop dit niet ondersteund aangezien safari enkel kan gedownload worden op Mac toestellen en geen enkel Mac toestel een touchscreen heeft.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Nee | Ja | Ja |

Klembord toegang – De Clipboard API geeft een ontwikkelaar de mogelijkheid om te interageren met het klemboard. Er kunnen zowel items van het klembord gelezen worden als dat er items kunnen geschreven worden naar het klembord.

Er worden ook methode voorzien voor het reageren op de actie waarbij een gebruiker zelf iets kopieert of plakt.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Ja |

**User experience**

Offline gebruik – Door het gebruik van serviceworkers kan een website offline gebruikt worden. Deze website moet eerste bezocht worden als de gebruiker online. De geladen pagina’s en andere items zoals foto’s kunnen opgeslagen worden voor offline gebruik.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Ja |

Achtergrond synchronisatie – Een actie kan van start gaan als de gebruiker een trage verbinding heeft of als hij offline. Achtergrond synchronisatie zal ervoor zorgen dat deze actie uitgevoerd wordt vanaf er een stabiele internetconnectie is, zelfs al is de applicatie reeds gesloten.

<https://developers.google.com/web/updates/2015/12/background-sync>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| ja | Nee | Ja | Nee | Ja | Nee |

Inter-app communicatie – de Web Share API samen met de Web share Target API zorgen ervoor dat links van websites kunnen geopend worden in native applicaties.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Nee | Nee | Nee | Ja | ja |

Betalingen – Aan de hand van de payment request API kan heel snel een betaling uitgevoerd worden. Bij apple toestellen zal dit gebeuren via apple pay. Bij andere browsers en toestellen is dit aan de hand van google pay of met een betaalkaart.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | Ja | Ja | ja |

Credentials – De credential management API levert methodes voor het ophalen en opslaan van de credentials van een gebruiker. Op deze manier kan de gebruiker eenvoudiger en veiliger aanmelden op een platform.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | Nee | Ja | Nee |

**Locatie en positionering**

Geolocatie – De Geolocation API zorgt ervoor dat een applicatie toegang heeft tot de locatie van een toestel. Dat wordt gedaan op basis van de gps-sensor of op basis van het netwerk.

De API voorziet niet enkel de locatie maar ook methodes die de applicatie informeren als de locatie veranderd.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Ja |

Geofencing – Geofencing is een technologie waarbij er een geografische zone wordt ingesteld. Als de gebruiker in deze zone komt wordt er automatisch een actie uitgevoerd.

Er was een voorstel om deze API uit te werken voor het web, maar op het moment van schrijven heeft geen enkele browser dit geïmplementeerd. Dit is functie die enkel beschikbaar is voor native IOS en Android-applicaties.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Nee | Nee | Nee | Nee | Nee |

Toestel oriëntatie – De Device Orientation API levert methodes voor het detecteren van de oriëntatie van het toestel. Er zijn drie eigenschappen die de oriëntatie bepalen.

* Alpha – Dit is de richting naar waar het toestel gericht is.
* Beta – Dit is het aantal graden dat het toestel voorwaarts of achterwaarts gekanteld is.
* Gamma – Dit is het aantal graden dat het toestel naar links of naar rechts gekanteld is.

De toestel oriëntatie kan, net zoals bij native applicaties, gebruikt worden als een input methode voor applicaties. Dit wordt vaak gebruikt bij games.

Deze API levert ook methodes die website in een bepaalde oriëntatie forceren.

Niet elk toestel heeft deze sensoren. Als ze aanwezig zijn worden ze ondersteund in volgende browsers. Deze API is vooral gericht naar mobiele toestellen.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Nee | Ja | Ja |

Toestel beweging – De Generic Sensor API is een verzameling van APi’s voor het gebruiken van verschillende sensoren van het toestel.

* deviceMotionEvent – levert informatie over de snelheid waarmee een toestel zijn oriëntatie en positie veranderd.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | onbekend | Ja | ja |

* accelerometer – levert information over de snelheid waarbij het toestel zich beweegt in een ruimte. De API levert zowel X, Y als Z coördinaten.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | onbekend | Ja | Nee |

* gyroscope – Levert informatie over de snelheid waarmee een toestel aan het roteren is. Om toegang te krijgen tot deze gegevens moet er eerst toestemming gevraagd worden aan de hand van de Permissions API.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | onbekend | Ja | Nee |

* Magnetometer – Meet het magnetische veld rond het toestel. Ook hier moet er eerst toestemming gevraagd worden om de sensor te mogen gebruiken.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Nee | Ja | onbekend | Nee | Nee |

* LinearAccelerationSensor – levert informatie over de snelheid waarbij een toestel beweegt, maar dit zonder de impact van de zwaartekracht.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | Ja | onbekend | Ja | Nee |

Nabijheid sensoren – geeft informatie over de afstand tussen het toestel en een object. Deze sensor wordt gebruikt om het scherm uit te zetten als een persoon aan het bellen is.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Nee | Ja | Nee | Nee | Nee | nee |

**Scherm en output**

Virtual and augmented reality – De webXR Device API zorgt voor een interface voor het connecteren van een virtual reality toestel met de browser. De sensoren van het toestel kunnen gebruikt worden om het canvas element te laden in het VR-toestel.

De oudere webVR API levert gelijkaardige methodes maar is beter ondersteund.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja (webXR) | Ja (webVR) | Ja (webVR & webXR) | Nee | Ja (webVR & webXR) | Nee |

Fullscreen – de Fullscreen API geeft een aantal methodes die ervoor zorgen dat de browser elementen verwijderd worden. Dit laat een website meer aanvoelen als een native applicatie.

Er worden twee methodes voorzien, een voor in fullscreen mode te gaan en een om deze te verlaten.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Ja | Ja | Ja | Ja | Ipad – ja Iphone – nee |

Wake lock – Veel toestellen gaan het scherm dimmen na een bepaalde tijd van inactiviteit. Dit kan onhandig zijn voor bepaalde applicaties zoals een gps. Met de Wake Lock API kan het automatisch dimmen van het scherm tegengegaan worden.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Edge | Firefox | Chrome | Safari | Android (chrome) | IOS (safari) |
| Ja | Nee | ja | Onbekend | Ja | Nee |

## Conclusie

### Browsers

We kunnen concluderen dat Google verder staat met zijn producten dan Apple. Zowel op de desktop-browser als op de mobiele browser worden meer functies ondersteund.

De nieuwste versie van Microsoft Edge ondersteunt verrassend veel functies. Dit komt omdat deze laatste versie (v80.0) gebaseerd is op de open source browser chromium die ontwikkeld is door google. Ook Google Chrome is op chromium gebaseerd.

Het is ook opvallende dat het bij Apple niet het besturingssysteem is die deze functionaliteiten tegen houdt maar de browser safari. Als Google Chrome gedownload wordt op een Apple toestel zijn deze functionaliteiten wel beschikbaar voor dit toestel.

### Consistentie

Het is opvallend hoeveel functies beschikbaar zijn voor het web. Slechts een beperkt aantal native applicaties zouden niet met web-technologieën gemaakt kunnen worden.

Het probleem is echter consistentie, het voordeel van het web is dat je één codebase hebt en dat deze applicatie op verschillende soorten toestellen werkt. Dit is voor een basis applicatie het geval, maar als er specifieke functies gebruikt moeten worden wordt het moeilijker. Verschillende systemen verwachten verschillende web-API’s. Veel van de API’s bestaan en kunnen gebruikt worden zullen ook niet op alle browsers werken.

Deze technologieën kunnen dus gebruikt worden om de functionaliteit van een applicatie te ondersteunen. Maar een applicatie zou niet mogen afhangen van een van de functionaliteiten want niet alle gebruikers zullen de toepassing kunnen gebruiken.

### Mobiele besturingssystemen

Zoals verwacht kunnen toestellen die Android gebruiken als besturingssysteem meer genieten van de functionaliteiten die het web te bieden heeft.

De grootste limitatie die IOS heeft in vergelijking met Android is het niet ondersteunen van push-notificaties. Dit is een functie die belangrijk is om gebruiker om betrokken te houden met een platform. Push notificaties zijn voor e-commerce platformen vaak een heel belangrijke tool binnen hun marketingplan  
<https://fivepints.com/PDF/5pp_Why_Are_Push_Notifications_So_Important.pdf>

Achtergrondsynchronisatie is een functie die de gebruikers ervaring kan verhogen. Deze functie die mogelijk gemaakt wordt door service workers is ook niet beschikbaar op een IOS-toestel.

Apple laat PWA’s toe om slechts 50MB aan data offline op te slaan, als deze applicatie voor een bepaalde tijd niet gebruikt wordt zal al deze data verwijderd worden om ruimte te op het toestel te besparen. Bij Android is dit 6% van de beschikbare opslagruimte en zullen de opgeslagen gegevens niet automatisch verwijderd worden.

Ook voor de functie om een web-applicatie te installeren is de ervaring op een Android toestel beter. Als een web-applicatie voldoet aan de normen om geïnstalleerd te worden zal Google Chrome op Android de gebruiker voorstellen om deze applicatie toe te voegen aan het startscherm. Deze functionaliteit is ook beschikbaar voor Safari op een Iphone maar hier moet de gebruiker zelf actie ondernemen en naar de instellingen van de website gaan om de knap ‘zet op beginscherm’ te vinden.

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

De integratie van de ingebouwde slimme assistent is beperkter. Als een applicatie geïnstalleerd is op een Android toestel, kan Google assistant deze openen aan de hand van een stemcommando. Siri heeft deze functionaliteit niet.

<https://aureatelabs.com/pwa/ios-getting-into-pwa-space-faster-than-ever/>

Het is verrassend dat Apple niet meer ondersteuning biedt voor PWA’s. Tijdens de voorstellening van de eerste IPhone (2007) had Steve Jobs het over het integreren van het web met de nieuwe Iphone. Tijdens zijn keynote had hij volgende uitspraken over het web:

‘you can write apps that look and behave exactly like apps on the phone’

‘Just put them on an internet server, They’re easy to update’.

<https://medium.com/@firt/progressive-web-apps-on-ios-are-here-d00430dee3a7>