# Eerste PWA

Voor sommige webapplicaties is het een grote meerwaarde om geïnstalleerd te kunnen worden en offline bruikbaar te zijn op het toestel van de gebruiker. Eenvoudige applicaties die geen gebruik maken van backend verzoeken kunnen op deze manier volledig functioneel zijn zonder internetconnectie.

## De applicatie

Darts is een spel dat gespeeld wordt drie pijltjes per persoon en een dartbord. De plaats waar het pijltje land op het bord bepaald de score van de worp, dit kan variëren van 0 tot 60. De speler die het eerst een score van 501 bij elkaar gooit is de winnaar. Deze score’s optellen en bijhouden wordt al snel ingewikkeld. Darts is een spel dat vaak gespeeld wordt in een bar, spelers zullen dus vaak zo snel mogelijk een tel applicatie gebruiken. Veel mensen zullen het als een voordeel ervaring dat de applicatie niet eerst gedownload moet worden

## Analyse

Er werden user stories opgesteld zodat de requirements van de applicatie duidelijk werden. Requirements opstellen aan de hand van user stories is een manier waarbij de requirements geformuleerd worden vanuit een nood van de eindgebruiker. Op deze manier wordt er steeds ontworpen in functie van de eindgebruiker

Een user story volgt volgend format: als een <type gebruiker> wil ik <een doel> zo dat <een reden>

<http://athena.ecs.csus.edu/~buckley/CSc131_files/Advantages%20of%20User%20Stories%20for%20Requirements.pdf>

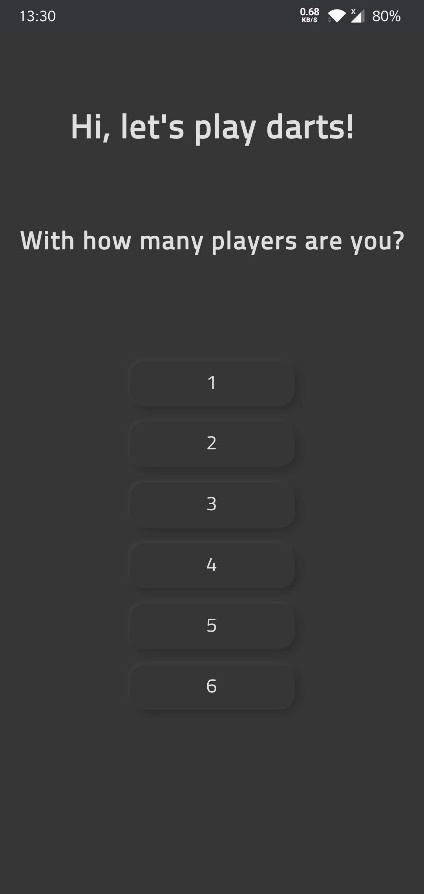
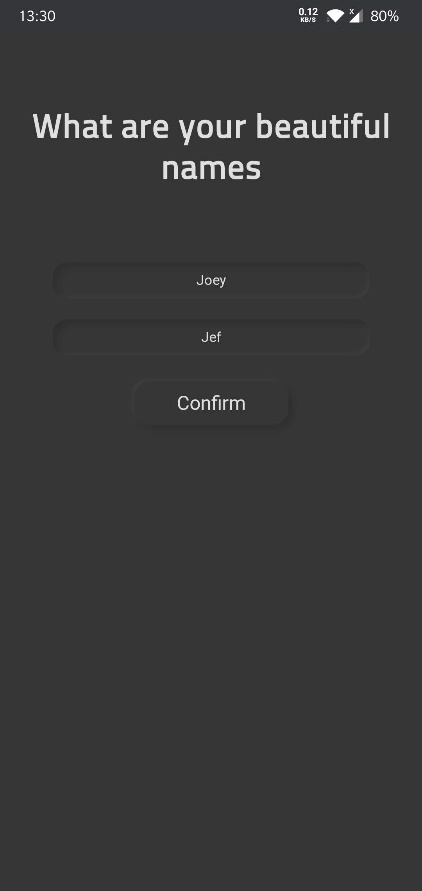
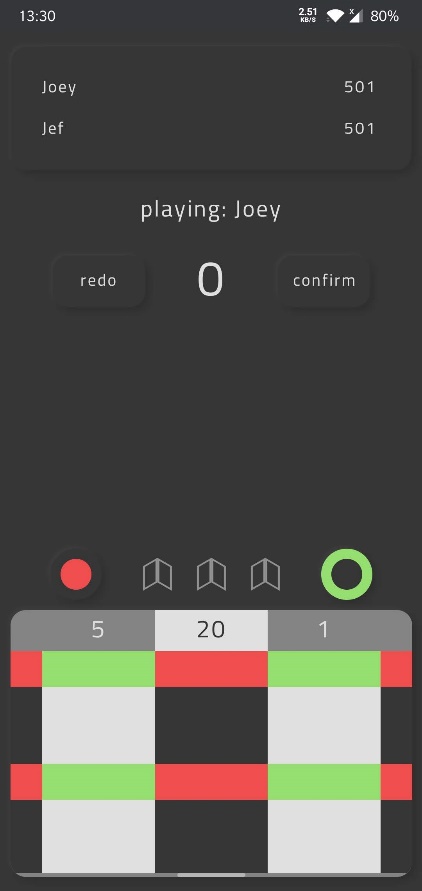
voor deze applicatie warden volgende user stories opgesteld

* Als een speler wil ik de score van een spel kunnen bijhouden zonder een applicatie te downloaden.
* Als een speler wil ik de score van een dartspel op mijn smartphone kunnen bijhouden zodat ik altijd en overal kan spelen.
* Als een speler wil ik de score van een dartspel bijhouden zonder een internet verbinding zodat ik niet afhankelijk ben van een eventuele wifi of netwerk verbinding.
* Als een groep vrienden willen we een dartspel kunnen starten en de score kunnen bijhouden voor een variabel aantal personen zodat we snel kunnen spelen.

## Implementatie

De applicatie werd eerst ontworpen aan de hand van figma, dit is een tool die gebruikt wordt om designs en prototypes te maken. De applicatie zal drie schermen hebben: een die het aantal speler vraagt, een die de spelersnamen vraagt en een scherm waar het spel kan bijgehouden worden.

Voor de ontwikkeling van de applicatie werd de javascript library react gekozen.

## A2HS

**HTTPS**

Een van de drie vereisten om een applicatie te kunnen toevoegen aan het startscherm is dat er een https connectie moet zijn. Dit werd bekomen door de webiste te hosten op Netlify (<https://www.netlify.com/>). Dit is een online service die gratis hosting voor statische websites aanbiedt en hier kan ook een SSL certificaat toegevoegd worden zodat er een HTTPS connectie tot stand komt.

**App manifest**

manifest.json:

{

  "short\_name": "Dart 501",

  "name": "Dart 501",

  "icons": [

    {

      "src": "/icons/android-icon-36x36.png",

      "sizes": "36x36",

      "type": "image/png",

      "density": "0.75"

    },

    {

      "src": "/icons/android-icon-48x48.png",

      "sizes": "48x48",

      "type": "image/png",

      "density": "1.0"

    },

    {

      "src": "/icons/android-icon-72x72.png",

      "sizes": "72x72",

      "type": "image/png",

      "density": "1.5"

    },

    {

      "src": "/icons/android-icon-96x96.png",

      "sizes": "96x96",

      "type": "image/png",

      "density": "2.0"

    },

    {

      "src": "/icons/android-icon-144x144.png",

      "sizes": "144x144",

      "type": "image/png",

      "density": "3.0"

    },

    {

      "src": "/icons/android-icon-192x192.png",

      "sizes": "192x192",

      "type": "image/png",

      "density": "4.0"

    },

    {

      "src": "logo512.png",

      "type": "image/png",

      "sizes": "512x512"

    }

  ],

  "start\_url": ".",

  "display": "standalone",

  "theme\_color": "#95df71",

  "background\_color": "#363636"

}

de naam en de verkorte naam (naam die op startscherm getoond zal worden) worden als eerste vastgelegd. Deze zijn gelijk omdat de volledige naam al kort genoeg is. Het maximum aantal karakters voor ‘short\_name’ is 12.

Vervolgens worden alle iconen voor de verschillende platformen bepaald. Dit deel van het manifest en de bestanden waarnaar het refereert kunnen gegenereerd worden met een online tool (<https://appicon.co/>)

Vervolgens wordt het gedrag en het uitzicht van de applicatie. Er wordt ingesteld dat de applicatie start op het scherm waar het aantal spelers geselecteerd kan worden, er wordt ook bepaald dat de applicatie het volledige scherm in beslag moet nemen.

Ook het kleurenschema wordt bepaald, deze kleuren worden gebruikt bij het startscherm en bepalen ook de kleur van de statusbalk van het toestel.

Er moet een link gemaakt worden van de html bestanden naar dit manifest bestand. Dit gebeurd binnen de head tags van de index.html file

  <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />

**Service worker**

Als een react applicatie gecreëert wordt met het ‘create-react-app) commando wordt er standaard een basis service worker aangemaakt. Standaard wordt deze niet gebruikt, om dit te veranderen moet de regsiter methode van de service worker aangeroepen worden in het index.js bestand.

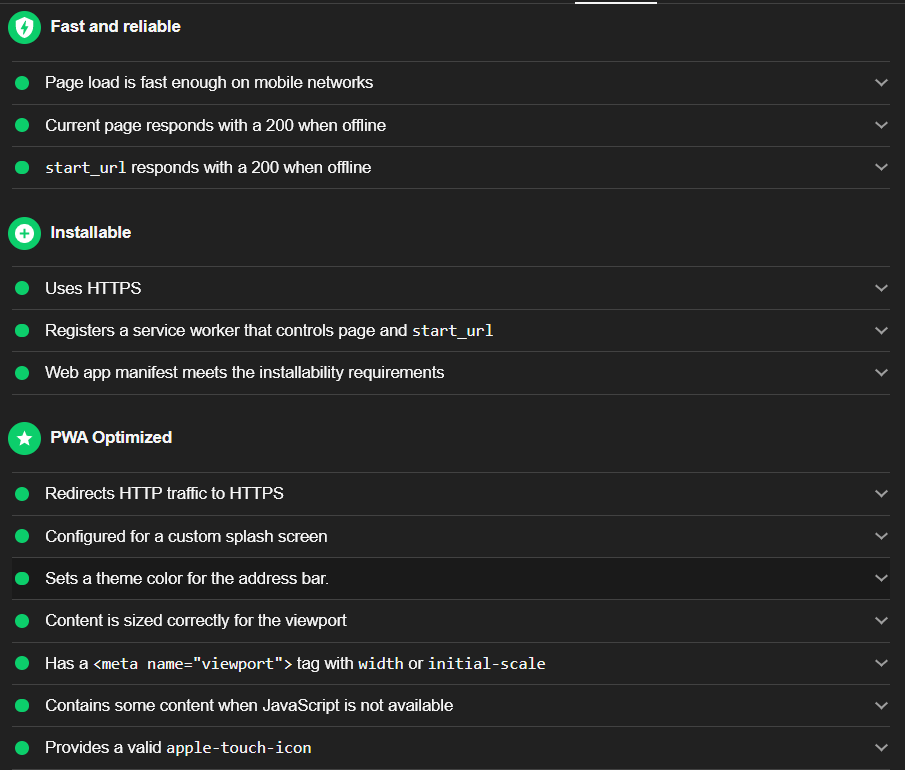
serviceWorker.register();

Deze service worker zorgt ervoor dat alle statische bestanden offline beschikbaar gemaakt worden en dat de gebruiker op een Android toestel de vraag krijgt om deze applicatie aan het startscherm toe te voegen.

Voor deze applicatie zijn biedt deze service worker genoeg functionaliteit.

**Controle**

Aan de hand van een lighthouse audit kan er gecontroleerd worden als alles werkt zoals verwacht.



Onder de sectie ‘installable’ kunnen we zien dat de applicatie inderdaad een HTTPS connectie heeft, dat er een service worker is en dat er een manifest aanwezig is.