

# Design & Build a single page web app

Based on a user story

1 Quote Api

2 Rijksmuseum Api

3 Frisbee api

→ 4 Barcode scanner on  
Products to scan

5 diagnose illness online

6 text to speech voice patterns  
for children bedtime stories

① Pick a user story

② Load data from the Api

Breakdown graph

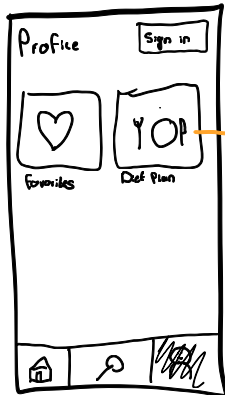
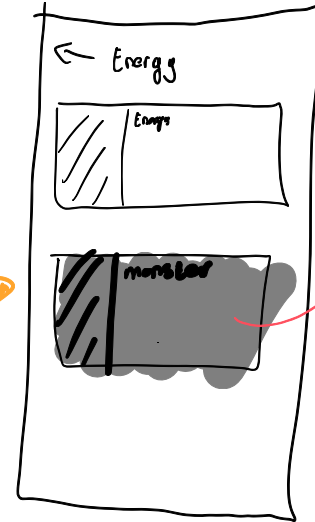
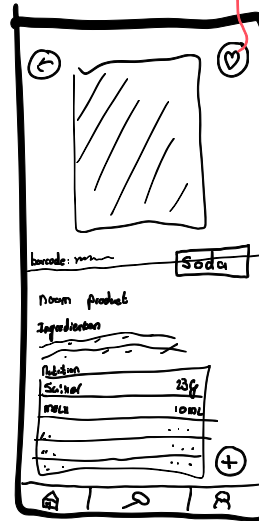
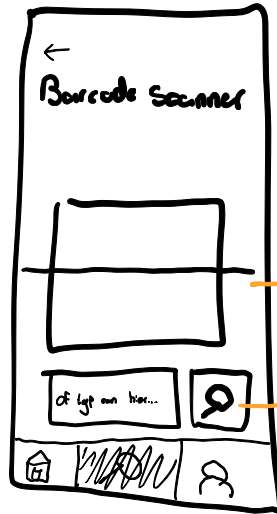
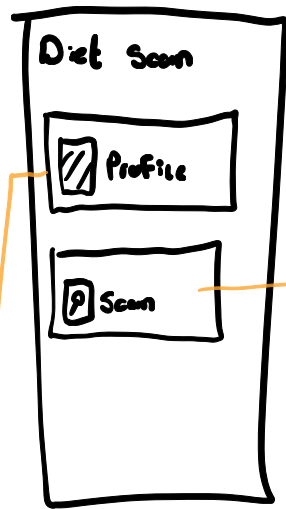


<https://nl.openfoodfacts.org/discover>

<https://nl.openfoodfacts.org/data>

<https://openfoodfacts.github.io/api-documentation/>

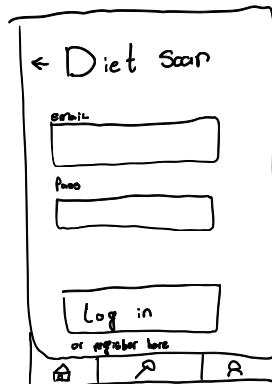
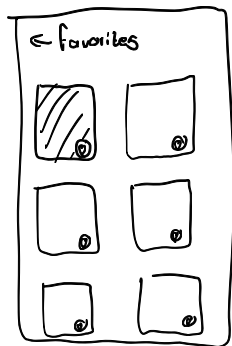
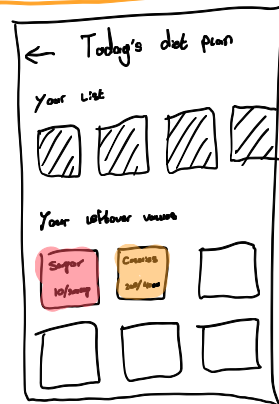
# Barcode scanner



logged in?

yes

no





Ja, je kunt een Node.js-applicatie maken voor gebruikersregistratie en -inloggen zonder gebruik te maken van Express.js. In plaats daarvan kun je de ingebouwde http-module van Node.js gebruiken om een server op te zetten en te werken met de request en response objecten.

Het opzetten van een back-end server met de http-module van Node.js vereist echter meer code en configuratie dan wanneer je een back-end framework zoals Express gebruikt. Je moet ook handmatig alle routing en middleware functies maken.

Als je wilt doorgaan zonder gebruik van Express, zijn hier enkele stappen die je kunt volgen:

1. Importeer de http-module van Node.js.
2. Gebruik de `createServer()`-methode van de http-module om een server te maken.
3. Luister naar inkomende verzoeken met behulp van de `server.listen()` -methode.
4. Schrijf handlers voor de gewenste HTTP-methodes zoals GET, POST, PUT, DELETE.
5. Werk met request en response objecten in de handlers om de juiste reacties te retourneren.
6. Om een database te gebruiken, installeer en importeer een database driver zoals 'mysql' of 'mongodb' en gebruik het om verbinding te maken met de database en de nodige gegevens te halen.
7. Gebruik hashing algoritmes zoals bcrypt of scrypt om wachtwoorden van gebruikers te beveiligen.

Dit zijn slechts enkele basisstappen om u op weg te helpen bij het bouwen van een Node.js-applicatie voor gebruikersregistratie en -inloggen zonder het gebruik van Express.js. Het kan echter een complex proces zijn ■

Op de eerste dag ben ik bezig geweest met het algemene idee voor de barcode scanner uit te werken. Ik bedacht me dat naarmate ik verder kom, ik de applicatie uitbreidt maar vermoed dat ik het alsnog iets te lastig heb gemaakt voor mezelf.

Ik liep vandaag vast op verschillende dingen. Het eerste wat ik probeerde te doen was het uitschrijven van de fetch naar een json bestand zodat als een fetch al is gedaan op een specifieke query. Dat dit niet opnieuw hoeft. Dit heb ik nog niet compleet geïmplementeerd maar het uitschrijven lukte met 'fs'. Enkel liep ik vast omdat nodemon constant mijn server opnieuw opstartte. Na wat strugglen bleek het te gaan om een change die natuurlijk plaatsvindt omdat ik een file aanmaak. daardoor herstart nodemon de server. Dit kan je verhelpen door een nodemon.json bestand aan te maken en daarin dingen te ignoreren.

```
1  {
2    "ignore": [
3      "*.json"
4    ]
5  }
6
```

Na een tijd kwam ik erachter dat ik geen node-js applicatie mocht bouwen. Hier was ik dus mee bezig dus moet ik aardig wat opnieuw doen. Ik sla het op voor later.





