

Opdrachtnemers

Naam, Studentnummer, Klas	Mike ten Klooster, 226751, DEV3
Cohort	2016-2017, DEV3
Crebonummer	25187
Opleiding naam	Applicatie- en mediaontwikkelaar
Onderwijsinstelling	Friesland College, Heerenveen

Document

Datum publicatie	17-06-2020
Documentversie	1.4

1. INHOUDSOPGAVE

1.	Inhoudsopgave.....	2
2.	Wat is voor mij Front-end	3
3.	Welke Front-End Frameworks ken ik	3
4.	Welke Front-end Frameworks heb ik gebruikt.....	4
4.1	Metro	4
4.2	Kube	5
4.3	UIKit.....	6
4.4	Ionic.....	8
5.	Welke methodieken heb ik geleerd	10
5.1	HTML / CSS	10
5.2	Javascript.....	10
6.	Waarom heb ik voor een API gekozen	10
7.	Project specificaties	11
7.1	Inleiding.....	11
7.2	Samenvatting	11
7.3	Link	11
7.4	Software	12
7.5	Eisen	12
8.	Schetsen	13
9.	Gebruikte documenten en Bronnen	14

2. WAT IS VOOR MIJ FRONT-END

Front-end is de uitstraling van de pagina, hoe die eruit ziet, hoe soepel de pagina verloopt naar andere pagina's en dat de pagina's dynamisch zijn en er dus zo goed uit zien op zoveel mogelijk formaten.

3. WELKE FRONT-END FRAMEWORKS KEN IK

Voor dit keuzedeel kende ik alleen maar Bootstrap en had ik gehoord van Angular en Ionic.

Maar dat is te weinig, ik heb me dus verdiept in de verschillende soorten front-end frameworks die er zijn. Ik heb verschillende sites vergeleken en gekeken welke in de meeste sites voorkwamen.

Uit het onderzoek kwamen een paar frameworks als meest gebruikt naar voren zoals Kube, Uikit en Ionic. Maar ik wou ook ondervinden waarom bijvoorbeeld ééntje slechter wordt beoordeeld, daarom besloot ik om ook één van de slechtere frameworks uit te proberen, dit werd Metro.

URL's naar de front-end frameworks:

Kube: <http://kube7.imperavi.com>

Uikit: <https://getuikit.com/>

Ionic: <https://ionicframework.com/>

Metro: <https://metroui.org.ua/index.html>

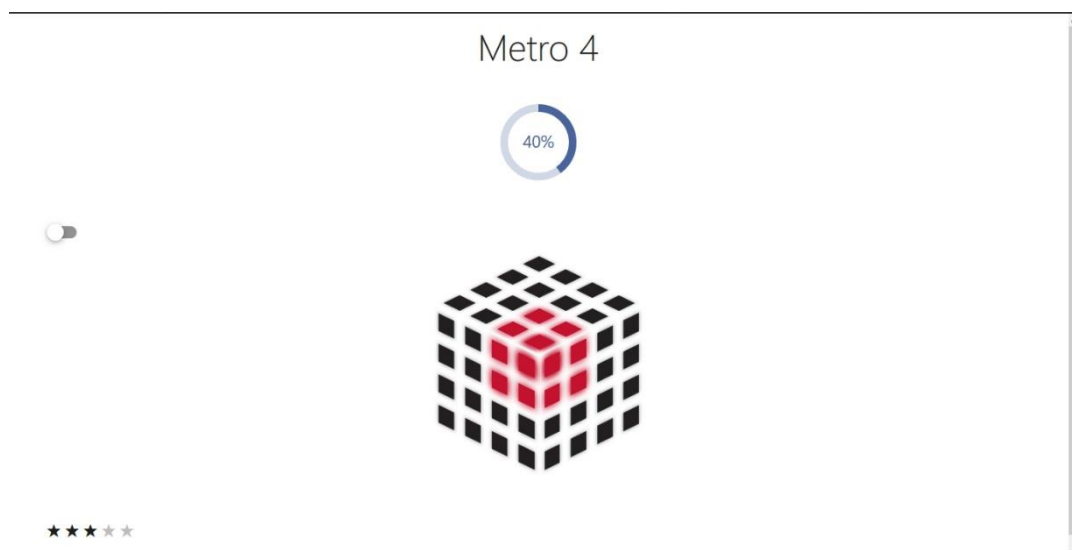
4. WELKE FRONT-END FRAMEWORKS HEB IK GEBRUIKT

Uit het onderzoek dat ik heb gedaan bleek dat het ene front-end framework betere documentatie had dan de ander, ik besloot dus om één te behandelen met slechtere documentatie (Metro) en 3 met betere documentatie (Kube, Ulkit, Ionic).

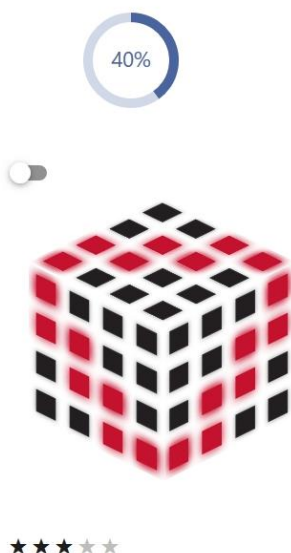
4.1 Metro

Ik besloot te beginnen met Metro, aangezien de documentatie daarvan het kleinst / onduidelijkst was. Hierbij probeerde ik de site dynamisch te maken en dat alles in het midden gecentreerd was. Dit was niet gelukt aangezien Metro geen align element heeft, het heeft alleen een grid met columns. Wat inhoudt dat je wel de column kan centreren maar niet de elementen die erin zitten.

Desktop view



Metro 4



Mobile view

4.2 Kube

Als tweede ging ik bezig met Kube hierbinnen wou ik een form maken en kijken hoe de pagina update op verschillende groottes, en of hij dynamisch blijft. Dit lukte vrij gemakkelijk aangezien het framework dynamische klassen levert.

Desktop view

Hello, world!

First Name	Last Name	Points
Goku	Son	9001
Gohan	Son	5301

Show ▾

LOGIN DATA

Name

Country

☐ Check me

☐ Radio me

Hello, world!

FIRST NAME	Goku
LAST NAME	Son
POINTS	9001
FIRST NAME	Gohan
LAST NAME	Son
POINTS	5301

Show ▾

LOGIN DATA

Name

Country

Mobile view

4.3 Ulkit

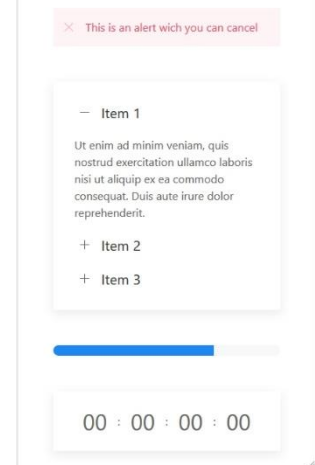
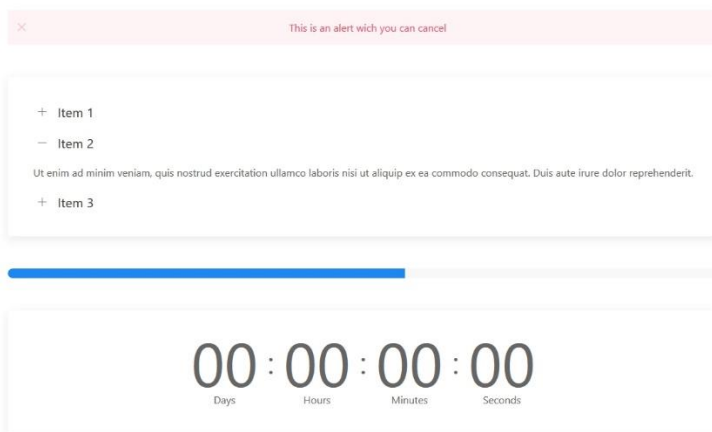
Voor de derde ging ik met Ulkit aan de slag, en kwam ik erachter dat Ulkit een grote en duidelijke documentatie heeft, dus besloot ik om hier een kleine uitdaging van te maken en het zelfde te doen van wat niet was gelukt met Metro, dit lukte snel en vrij gemakkelijk en was er zelf positief van verrast.

Toen besloot ik dat ik de moeilijkheidsgraat omhoog zou gooien en het front-end te verbinden met een API (hier heb ik zelf een API voor gebouwd) en er ook een login aan vast te knopen. Na veel onderzoek is dat ook gelukt en ik was blij met het resultaat.

Desktop view #1

|

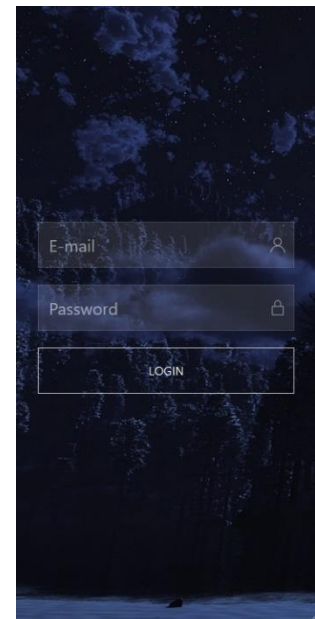
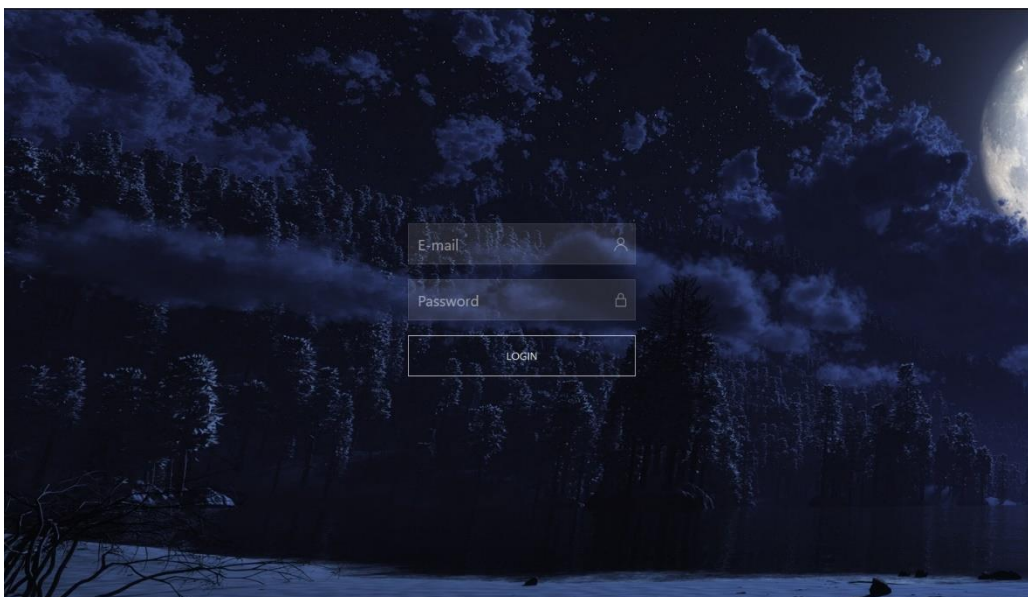
Mobile view #1



Desktop view #2 Login

|

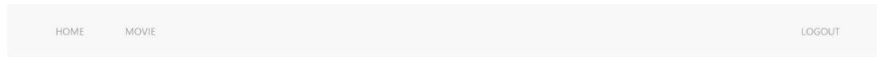
Mobile view #2 Login



Desktop view #2 Home

|

Mobile view #2 Home



Welcome back

16:11:12:18

Tue Hours Minutes Seconds



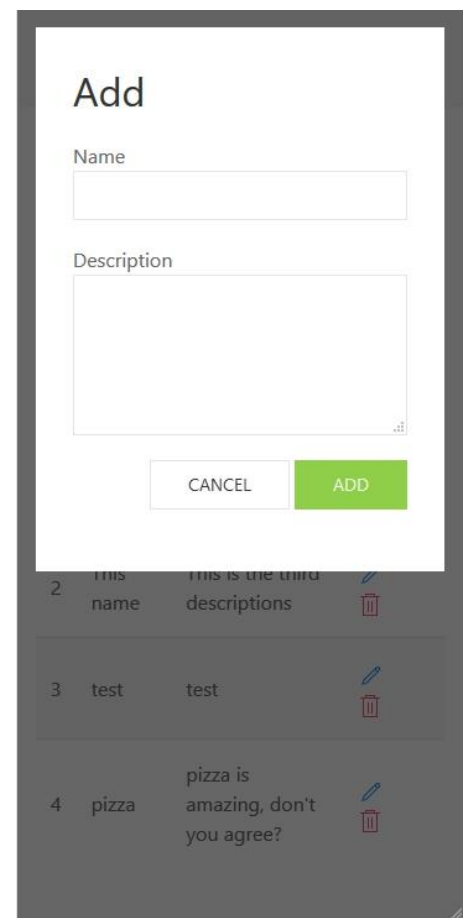
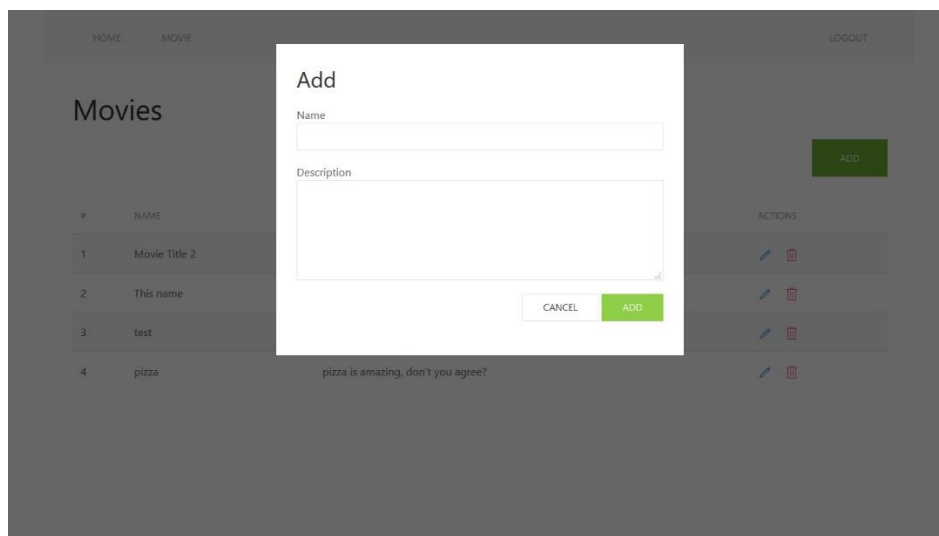
Welcome back

16 : 11 : 33 : 52

Desktop view #2 Movie Add

|

Mobile view #2 Movie Add



4.4 Ionic

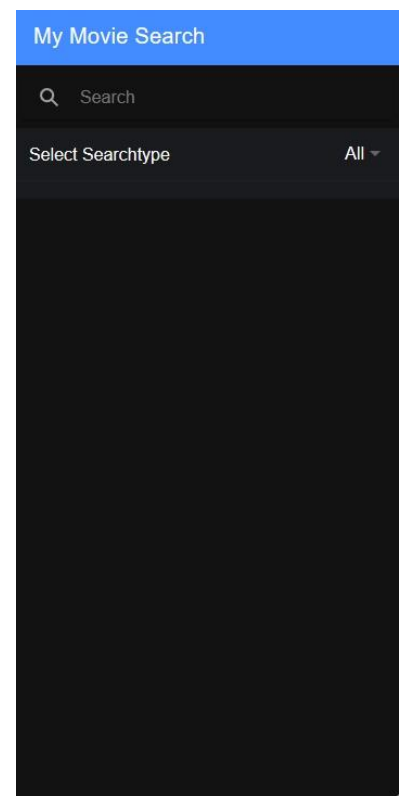
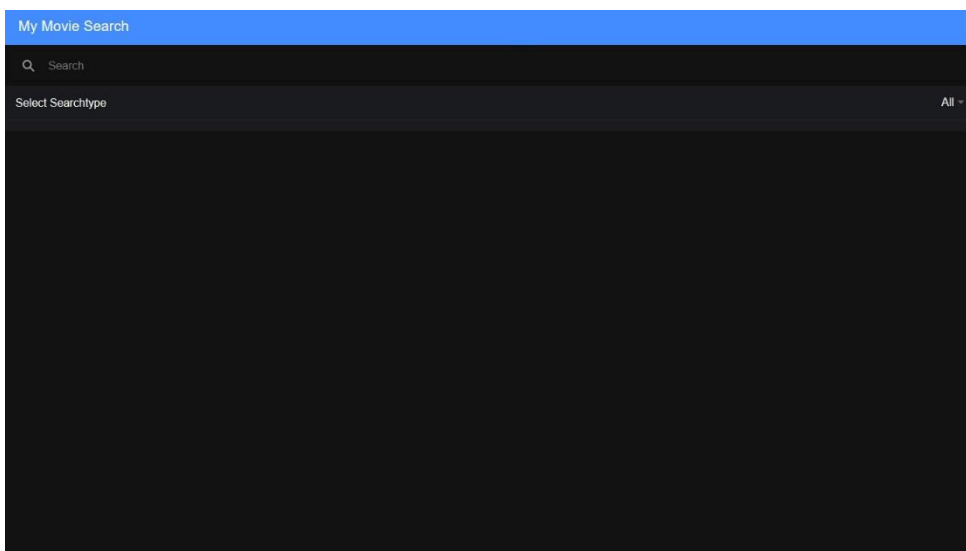
Als vierde ging ik aan de slag met Ionic, Ionic is groter en complexer dan de voorgaande frameworks en je kan er dus ook meer mee. Met deze insteek besloot ik om een Ionic project te maken die ook werkt met een API, als eerste heb ik er dus één programma gemaakt met de OMDb API en daarna één met mijn eigen API.

Na het vergelijken van de twee projecten was het overduidelijk dat ik niet een grote applicatie kon maken met mijn API wat tot de beslissing heeft geleid om mijn eindproject met de OMDb API te doen. Het enige nadeel van de OMDb API is dat er dubbele data in de database zit en dat er zo veel resultaten zijn dat je een specifieke query moet opgeven omdat je anders te veel resultaten terug krijgt.

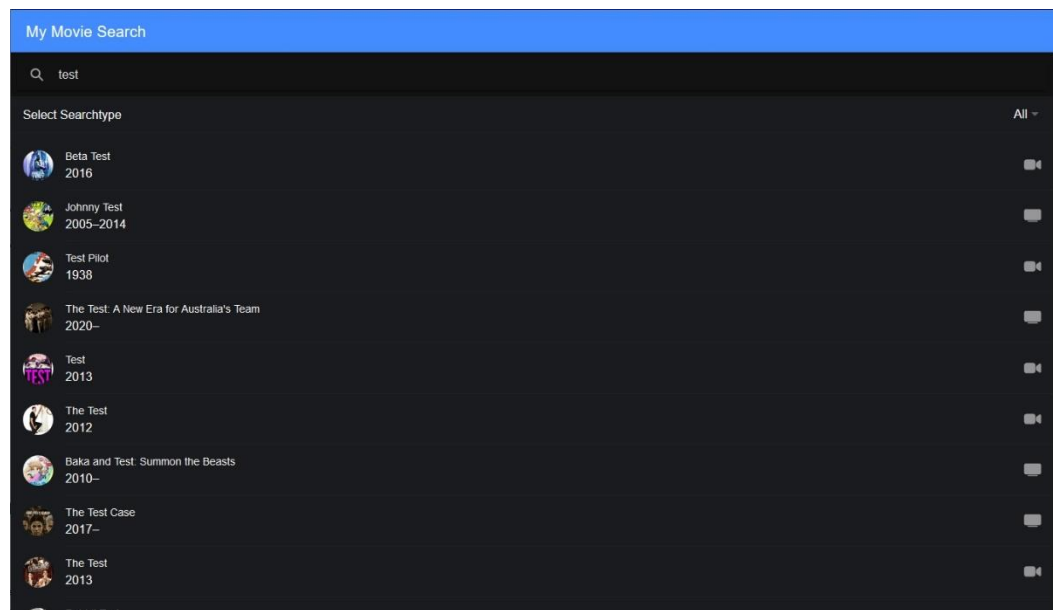
Desktop view #1

/

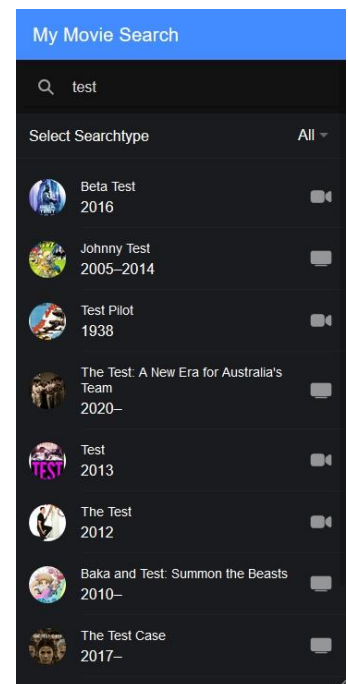
Mobile view #1



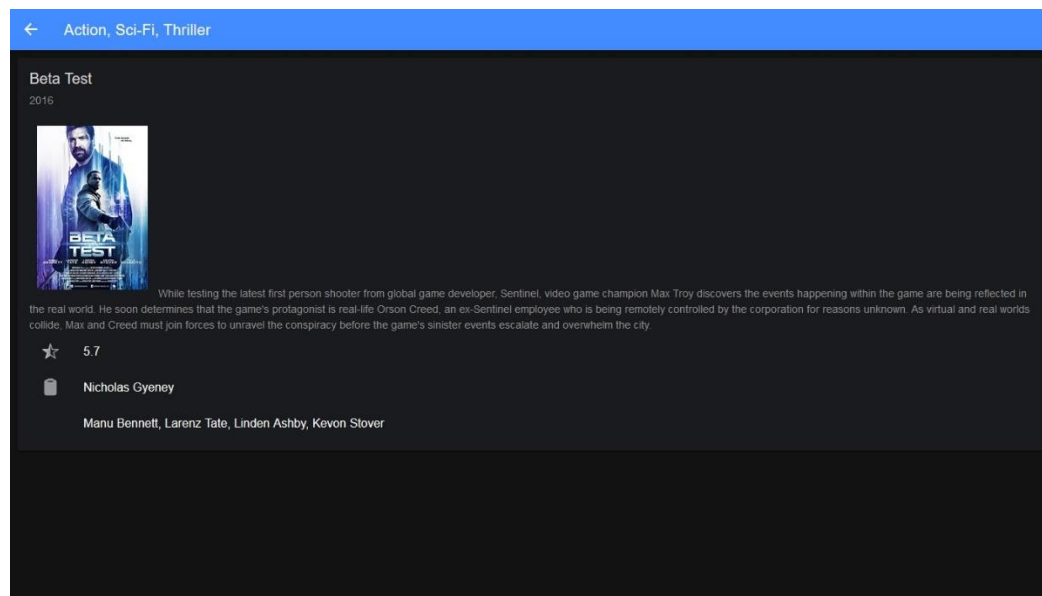
Desktop view #2 Home



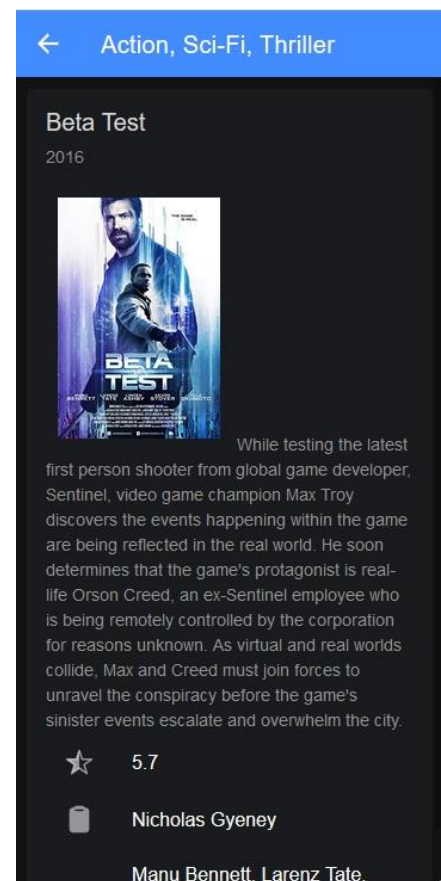
Mobile view #2 Home



Desktop view #2 Detail



Mobile view #2 Detail



5. WELKE METHODIEKEN HEB IK GELEERD

5.1 HTML / CSS

Binnen HTML / CSS heb ik geleerd de methodieken van de verschillende frameworks te gebruiken, hoe de opbouw van het document loopt en hoe de klassen van het framework worden stapelt om het gewenste resultaat te krijgen.

5.2 Javascript

Front-end heeft mij geforceerd om javascript te gaan gebruiken, zelf had ik weinig ervaring en kennis binnen javascript. Eén van de grootste dingen die ik heb geleerd binnen javascript is Fetch, Fetch is een manier van het manipuleren van de requests. Ik heb Fetch ontdekt aangezien ik een manier zocht om requests te sturen en op te vangen van de API.

Iets anders is het gebruik maken van DOM elementen zoals hoe je bijvoorbeeld door data heen kan mappen en de data ervan aan een DOM element kan meegeven.

Het andere is de Sessionstorage gebruiken, dit was simpeler dan gedacht en kan je makkelijk controleren, Sessionstorage wordt gebruikt om gegevens tijdelijk op te slaan.

6. WAAROM HEB IK VOOR EEN API GEKOZEN

Wanneer ik over een API (Application Programming Interface) spreek doel ik op een Rest API, maar wat is een Rest API eigenlijk, een Rest API is a connectie tussen de database en de site (client). Dus door een request naar de API te sturen kan je de data ophalen die in de database zit.

Zelf had ik nog nooit gewerkt met API's en wist niet waar ze voor waren en wat ze deden, ik heb er onderzoek naar gedaan en besloot om er ook binnen een project er gebruik van te maken. Dat heeft er tot geleid dat ik er zelf één heb ik gemaakt om te kijken hoe een API in elkaar steekt.

Maar mijn eigen API heeft natuurlijk niet genoeg data om een leuke / grote site te maken. Daarom besloot ik om een grotere complexere API te gebruiken, na lang onderzoek besloot ik om gebruik te gaan maken van de OMDb API aangezien deze heel groot is en daarnaast ook nog eens duidelijk documentatie heeft. Het enige nadeel daarvan is dat er dubbele data in voorkomt en dat je specifieke zoekopdracht moet doen omdat je anders te veel resultaten terug krijgt en die dus niet kan ophalen.

7. PROJECT SPECIFICATIES

7.1 Inleiding

Dit project is mijn Eindproject voor het keuzedeel Front-end development.

De doelgroep is alle leeftijdsklassen

7.2 Samenvatting

Het project is het maken van een filmsite waar je een grote verscheidenheid hebt aan films, en waar je op basis van een titel een film kan vinden en de details ervan kan bekijken.

Op deze manier kan iedereen die op de site komt films browsen en informatie opdoen voor als ze de film willen kijken of niet.

7.3 Link

Github	https://github.com/Miketenklooster/api_frontend
Movie Radar site	https://miketenklooster.github.io

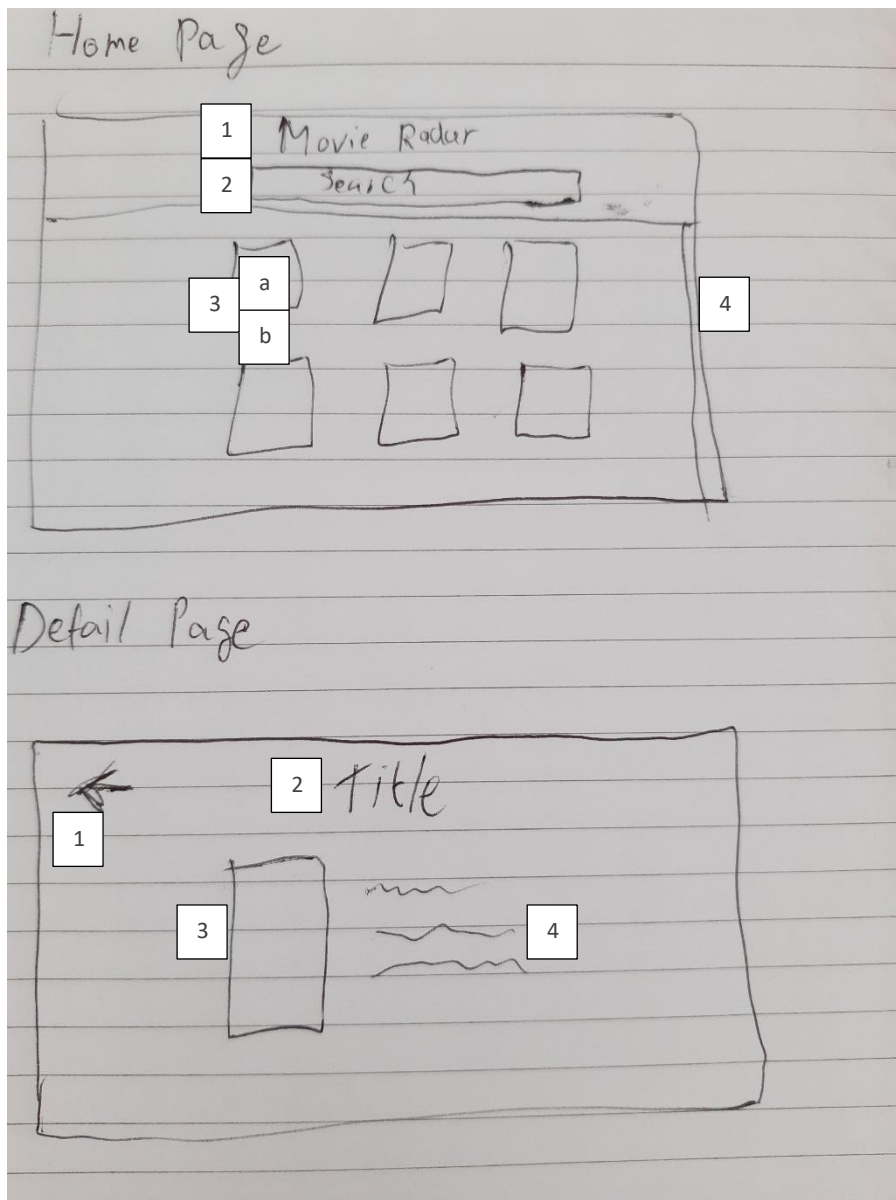
7.4 Software

Microsoft Teams	Chat applicatie die we gebruiken voor onderlinge communicatie.
Github	Voor versie beheer, kan je iedereen zijn input bijhouden.
PHP 7.4	PHP versie die ik ga gebruiken.
PHPStorm 2019.3.2	Code editor die ik ga gebruiken.
UIKit 3.5.3	Front-end Framework die ik ga gebruiken
Microsoft Word	Voor het maken van documentatie.

7.5 Eisen

- Een Front-end Framework
- Dynamisch op alle formaten
- Het systeem bestaat uit:
 - Hoofdpagina
 - Detailpagina
- SEO (Search Engine Optimisation).

8. SCHETSEN



Home pagina

1. Pagina titel
2. Searchbar
3. Resultaten
 - a. Poster
 - b. Titel
4. Scrollbar

Detail pagina

1. Terug naar vorige pagina
2. Pagina titel
3. Film poster
4. Film beschrijving

9. GEBRUIKTE DOCUMENTEN EN BRONNEN

Google translate: <https://www.google.com/search?client=firefox-b-d&q=google+translate>

W3Schools: <https://www.w3schools.com/>

Fetch: https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch

Feedback

- Collin Franckena
- Nick Glazenburg
- Timo Hogeveen