# JSON

JSON staat voor Javascript object notation. JSON is een array/object gebaseerde datatype dat vooral wordt uitgelezen met Javascript. Het kan gedaan worden met andere talen maar het is het meest praktisch in Javascript. JSON is een veelgebruikt bestandstype in browsers en word veel gebruikt door kleine bedrijven die makkelijk data willen verwerken.

## Dataverwisseling JSON

JSON data wordt meestal verwisseld door het mee te geven aan de Javascript en de Javascript kan het dan uitlezen door om te zetten in Javascript formaat. De Javascript kan het dan laten zien op de pagina in kwestie. JSON is een heel handige taal omdat het makkelijk gemanipuleerd kan worden met Javascript. Daarom gaat de dataverwisseling meestal met veel gemak.

## Welke browsers ondersteunen JSON?

Alle moderne browsers ondersteunen native JSON-codering/ decodering (Internet Explorer 8+, Firefox 3.1+, Safari 4+ en Chrome 3+).

## Welke Programmeertalen begrijpt JSON?

De meeste technologieën die werken met de gegevensoverdracht tussen de systemen ondersteunen het JSON-formaat.

HTML, javascript, C, C++, C#, Perl, Java, Python, Php etc.

# CSV

CSV staat voor “comma separated values”. CSV is een data bestand type dat werkt met commas om data verdeling en kolommen in de data aan te geven. Het leest data uit van links naar rechts dus als je bijvoorbeeld dit hebt: Bob,,Smith,,,,,,,,Bob@example.com. Het word uitgelezen van links naar rechts. De extra comma(s) zijn verdeling zodat de uitlezer weet hoeveel ruimte zit tussen data. Dit is CSV een heel simpel data type meestal gebruikt in Excel maar kan ook gebruikt worden in notepad of normale tekst editors. Het uiteindelijke doel van CSV is om simpel uitgelezen te worden op bijvoorbeeld een webpagina.

In conclusie CSV is een simpele data base taal die wel gebruikt kan worden voor simpele taken zoals uitgewerkte Excel data op een pagina laten zien. Dat is ongeveer waar het op neer komt.

## Bestandsuitwisseling CSV:

CSV werkt heel veel met kolommen. Zoals al eerder gezegd is het verdeeld met komma`s. Maar de kolommen worden simpelweg verdeeld met een enter. Als het uitgelezen wordt door bijvoorbeeld JS word het getoond in kolommen. De JS kan dit gewoon manipuleren hoe hij dat wilt en laten zien hoe hij dat wilt.

Voor CSV kan je een simpele XMLHTTPRequest doen om de data op te vragen/in te laden. Dit past toe bij JS. In conclusie het is een simpele bestandsuitwisseling die met een XMLrequest gedaan kan worden.

Ondersteuning CSV:   
Omdat CSV bijna gewoon een simpel tekstbestand is kan het uitgelezen worden door bijna elke programmeertaal. Behalve misschien basic maar verder kan het uitgelezen worden door bijvoorbeeld: Java, javascript, python, C en PHP.

Omdat het zo enorm simpel is kan het heel gemakkelijk uitgelezen worden. Vandaar de grote ondersteuning.

## Keuze programmeertaal CSV:

Over het algemeen de beste keuze voor data uitlezen is PHP. Maar door de simpele natuur van CSV is het eigenlijk compleet aan de ontwikkelaar. PHP is veiliger maar JS is makkelijker voor mensen met minder backend ervaring. Javascript is ook makkelijker te manipuleren.

Het kan ook gedaan worden met Java of python maar dit is niet ideaal voor de browser, wel voor applicaties. Dus voor de browser zou ik kiezen voor Javascript maar voor applicaties Python.

# XML

## Documentatie XML:

XML staat voor Extensible Markup Language. De taal lijkt heel erg op HTML. Het verschil is dat ze allebei een ander doel hebben. XML is gemaakt om gegevens te transporteren en op te slaan, en HTML is ontworpen om te laten zien hoe de gegevens er uit zien.

Als je bezig gaat met XML moet je al enige kennis hebben van HTML en Javascript.

XML is niet heel bijzonder. Het is gewoon tekst. Software die platte tekst aankan, kan ook overweg met XML. XML-bewuste apps kunnen de XML-tags verwerken. De functie van een tag hangt af van de app.

Met XML bouw je je eigen tags uit. XML-taal heeft vooraf geen gedefinieerde tags. De tags die gebruikt worden in bijvoorbeeld HTML zijn wel vooraf gedefinieerd. Dat betekend dat je in HTML alleen de tags kan gebruiken die zijn gedefinieerd in de HTML-standaard (zoals <p>, <H1>.). Met XML is dit niet zo,

XML zelf doet niks. Alles in een XML bestand is gewoon pure informatie.

## Bestandsuitwisseling XML:

XML lijkt dus heel erg op HTML. Met XML werk je voornamelijk met tags. Deze tags bedenk je zelf. In het volgende voorbeeld kan je zien hoe een XML bestand werkt.

<CATALOG>

<CD>

<TITLE>Hide your heart</TITLE>

<ARTIST>Bonnie Tyler</ARTIST>

<COUNTRY>UK</COUNTRY>

<COMPANY>CBS Records</COMPANY>

<PRICE>9.90</PRICE>

<YEAR>1988</YEAR>

</CD>

</CATALOG>

Dit XML script staat pure informatie over de cd. Het script zelf doet helemaal niks

## Ondersteuning XML:

XML is dus gewoon tekst die heel makkelijk te begrijpen is voor mens en machine. Het is een globaal datatype. En dus te gebruiken met elke progameertaal.

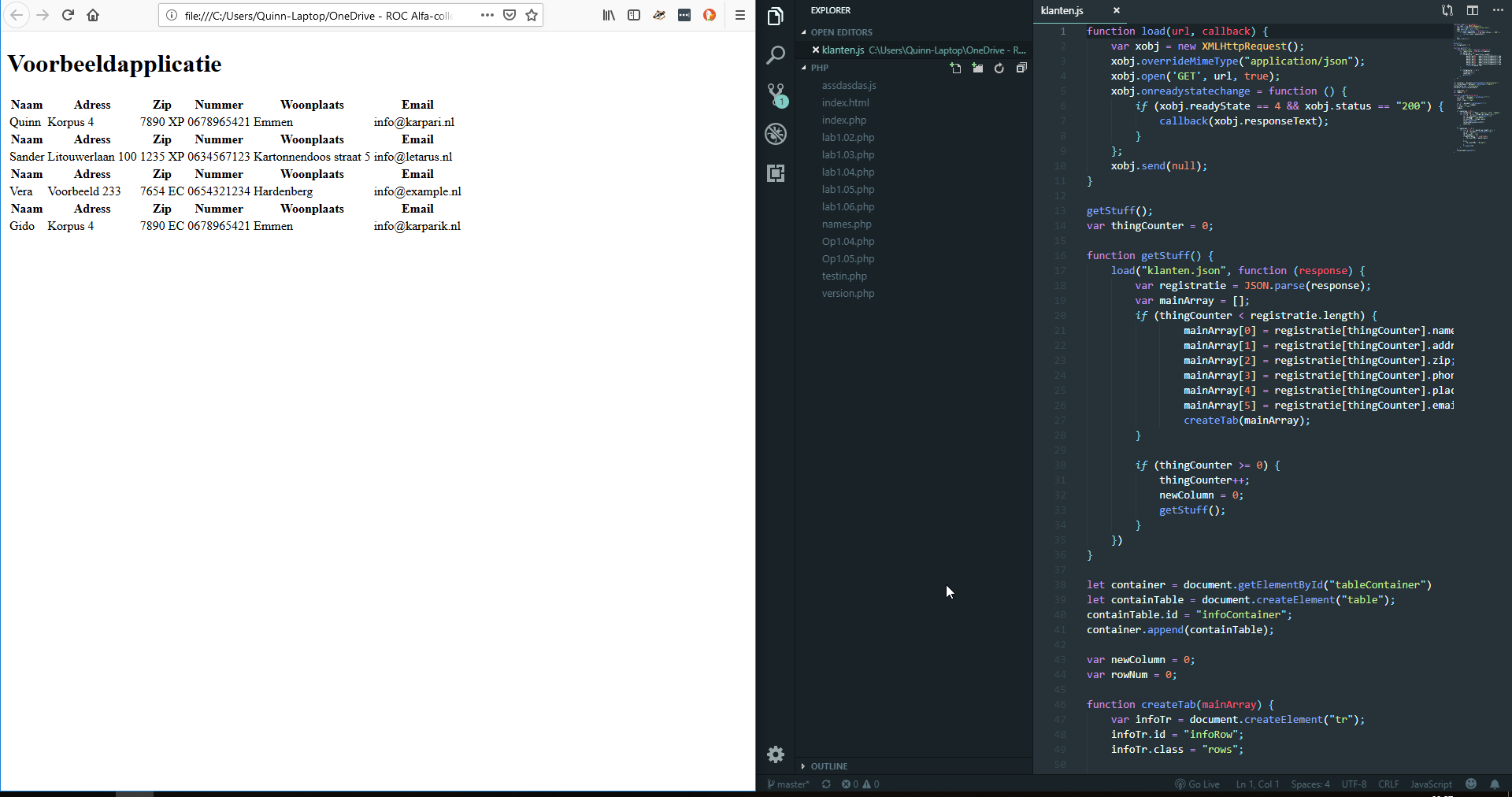
# Verantwoording

Wij hebben als groepje besloten dat het makkelijkst te gebruiken data type XML is. De paar redenen hiervoor zijn is dat het een globale data taal is, en het als makkelijkst te gebruiken is. Je kan het heel simpel manipuleren in de XML dus het data verwerken in de Javascript is heel makkelijk. Dit is een van de grootste voordelen van XML en waarom wij vinden uit als onze vindingen dat XML de makkelijkste bruikbare taal is.

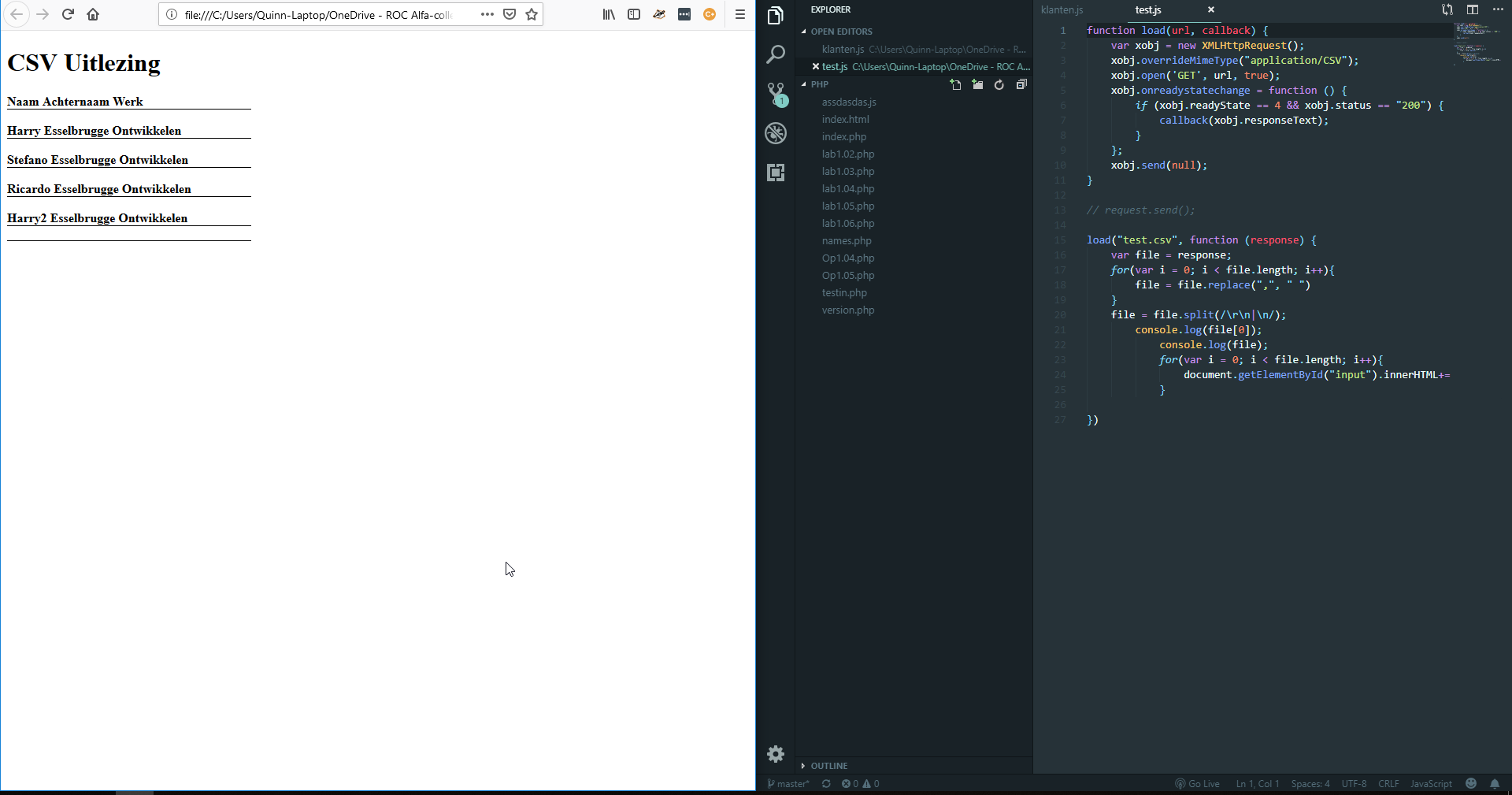
In conclusie wij kozen voor XML omdat het makkelijk is en heel user friendly. Vooral voor beginners is het heel makkelijk en leuk te gebruiken. Dus XML is voor ons het makkelijkst om gegevens tussen externen uit te wisselen met externen.

## Voorbeeldapplicaties

**JSON**



**CSV**



**XML**

