Bronnen end to end

<https://code-maze.com/selenium-aspnet-core-ui-tests/>

* Stappen om een test op te stellen met selenium.

<https://github.com/CodeMazeBlog/testing-aspnetcore-mvc/tree/ui-testing-selenium>

* Broncode die bij de stappen hoort.

<https://www.lambdatest.com/learning-hub/webdriver>

* Uitleg die verduidelijkt wat selenium is en hoe je dit kan gebruiken in je code.

<https://www.browserstack.com/guide/wait-commands-in-selenium-webdriver?utm_source=google&utm_medium=cpc&utm_platform=paidads&utm_content=602353912717&utm_campaign=Search-DSA-NB-Europe&utm_campaigncode=Guide-Pages+9040086&utm_term=+&gclid=Cj0KCQjwxYOiBhC9ARIsANiEIfZBJWIcq9t7xA6Y5Qvytk6cMZBNVzhdNmmc9_1wLHh5yPgfbjyKHyMaAtrOEALw_wcB>

* Bepaalde wait functies die een specifiek onderdeel zijn van de testen. In deze bron staat welke verschillende functies er zijn en hoe deze correct te gebruiken.

Daarnaast heb ik ook nog gebruikt gemaakt van de presentaties die gegevens zijn in de lessen Agile en Testing om de basisprincipes te leren gebruiken.

Voor en nadelen

## Voordelen

* De volledige applicatie testen in gebruiksomstandigheden.
* De UI live kunnen bekijken hoe dit er voor de gebruiker uitziet wanneer deze gebruikt word.
* Niet werkende delen van het programma worden zo snel duidelijk alvorens deze in de praktijk gereleased worden.

## Nadelen

* End to end testen duurt zeer lang in vergelijking met unit testing.
* Een test voorbereiden (analyse) en het schrijven van de test zelf neemt veel tijd in beslag.