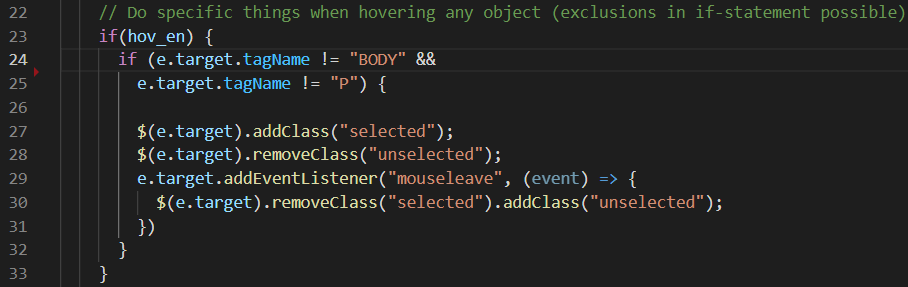
Tracking user mouse data

# Git

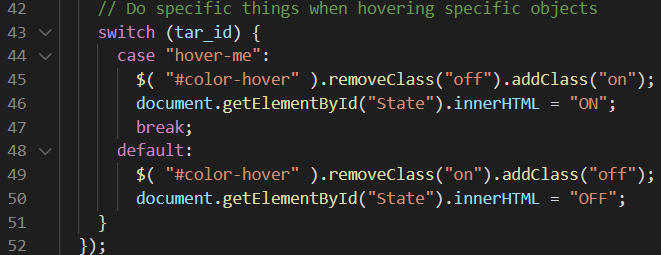
<https://github.com/TommyTytgat/Stage>

# Onderzoek

## Gegevens muis

* Onderzoeken hoe we in JavaScript de muislocatie (coordinaten) van de gebruiker kunnen ophalen  
  Dit is super ‘spammy’ dus heb ik dit voorlopig in commentaar gezet, maar e.pageX, e.clientX, e.pageY en e.clientY kunnen gebruikt worden om de coördinaten van de cursor zijn locatie weer te geven.  
  Page <-> client:  
  De page-coördinaten geven de coördinaten tot het begin van de DOM (als DOM uit het zichtbaar scherm loopt, zijn coördinaten groter dan de dimensies van het scherm mogelijk)  
  De client-coördinaten geven de coördinaten in het zichtbaar scherm, dus deze hebben een vaste min-max op elke pagina (afhankelijk van schermgrootte neem ik aan)
  + Hoe gaan we dit opslaan?  
    2D-Tensor? [[x1, y1][ x2, y2]… [xN, yN]  
    Misschien herschalen naar procentuele waarden?  
    Timestamp belangrijk?
  + Hoe vaak gaan we de muislocatie moeten opslaan?  
    TODO: Berekening grootte van opgeslagen array per minuut/uur/dag?
  + Scaling van scherm? → Iemand met een ultra-wide t.o.v. een gewoon 13” laptop?  
    Opnieuw: schaling kan procentueel => makkelijker voor universaliteit en misschien later een stap uitgespaard voor het AI-deel
  + Kunnen we enkel de locatie van de muis bekomen als de persoon op een website zit? Of kunnen we de muis ook volgen als de gebruiker in de browser zit te bewegen (vorige/volgende/zoekbalk)?  
    Ik heb geen manier gevonden om user input te volgen buiten de DOM van de huidige webpagina. We kunnen wel detecteren of men de DOM verlaat met de muis, maar niet of men de zoekbalk/navigatie/tabbladen selecteert.  
    TODO: Misschien via extensie?
* Aan de hand van de muislocatie de onderliggende HTML elementen opvragen

Deze code staat in voor elke target (object) dat de muis over zweeft een class toe te kennen en daarna ook terug te verwijderen en een andere toekennen als ‘default’. Let wel dat geen enkel object in mijn DOM al een van deze classes heeft, deze worden dynamisch toegekend, zo moeten we geen events aanmaken voor iets dat nog nooit geselecteerd was.

Deze code is specifieker, maar komt op hetzelfde neer. Switch statement voor optimaal prutserijen te vermijden (zoals if-lus na if-lus).

## Extensie

Research doen rond het bouwen van een Chrome/Safari/Firefox extensie.

* Is het mogelijk een extensie te bouwen voor alle browsers?
  + Gaan we moeten kiezen voor Chromium / Safari / Firefox?

Normaal niet, er zijn wel een aantal dingen die we rekening mee moeten houden:  
<https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions/Build_a_cross_browser_extension>

* Mogelijk om de gegevens van een muis op te vragen in een extensie? GDPR-wise?

Er zijn een aantal musts die voldaan moeten worden, meer info op deze link:  
<https://www.cookiebot.com/en/gdpr-cookies/>

* Extensie uitrollen in de store van een browser?
  + Gratis? Betalend? Andere zaken waar we rekening mee moeten houden?

In bovenstaande mozilla-link staan meer details. Chrome ($5) en Safari vragen een registration-fee als men extensions wil publishen.  
Ook moet elke store van elke browser apart beheerd worden.

* Is het mogelijk een overlay te tonen op een website (zoals de ALS-Liga zelf heeft op hun website).