Testplan

Ontdekstation 013

Fontys

Voorjaar 2024

Inleiding

**Om dit langlopende project overzichtelijk te houden hebben wij dit testplan aangemaakt, waarin wij updaten wat voor tests wij hebben toegevoegd aan het project en hoe ze werken.**

Wat gaan we testen

In dit semester gaan wij een variatie aan toevoegingen en veranderingen aanbrengen aan het project. Om de integriteit van de codebase zo goed mogelijk te behouden is het belangrijk om de code op verschillende punten te testen.

# End-to-end testing

Een van de punten waarop we het project gaan testen is e2e. Hiermee simuleren we automatisch de stappen die een gebruiker ook zou doorlopen op de site. Het voordeel van e2e tests automatiseren is dat het sneller is dan menselijke testers. Het grootste nadeel zijn zogenaamde edge cases. Dit situaties die buiten de test ruimte vallen. Een mens zou het handmatig nog kunnen controleren, maar een machine doet alleen maar wat het moet doen, waardoor edge cases ook niet snel aan het licht komen.

# Component testing

Component testing komt neer op het testen van individuele ‘componenten’ van een applicatie. Dit kan bijvoorbeeld een zoekfunctie zijn, of een winkelwagen.  
Wij willen graag component testing uitvoeren om er zeker van te zijn dat onze componenten goed functioneren.  
Een voordeel van component testing tegenover e2e is dat er per component tests worden ontwikkeld, waardoor we componenten met meer kans op edge cases beter kunnen controleren. Terwijl e2e over grote lijnen test, kunnen we met component testing onze resources beter verdelen over de kwetsbaardere onderdelen van de applicatie terwijl de robuuste delen het minimum krijgen.

# Unit testing

Unit tests zijn een mindere focus van ons, maar ook belangrijk. Zeker voor gevoelige onderdelen van de applicatie is het belangrijk om verzekerd te zijn van de veiligheid van de applicatie en de gegevens die je bewaart.  
Net zoals met component testing kunnen we met unit testing nog een laagje dieper in gaan op edge cases door zwakke plekken beter aan te pakken.

Hoe gaan we testen?

# Cypress

**Voor E2E en component tests gaan we gebruik maken van een frontend library genaamd Cypress. Deze tool kan acties van gebruikers automatiseren en heeft een aantal handige features om te controleren of dat alles klopt.**

E2e gaan we testen door cypress een volledig proces door te laten lopen, zoals inloggen, uitloggen of een nieuw meetstation toevoegen.  
component tests gaan we uitvoeren door enkele pagina’s of delen van pagina’s (components) te automatiseren, en in meer detail te kijken naar de resultaten. Hiermee willen we vooral controleren of dat de juiste errors worden gebruikt en of dat alle onderdelen van een component werken naar behoren.

Component testing

**Dit stuk zal alles omvatten wat te maken heeft met onze component tests. Hierin zullen we de doelen, tests en resultaten weergeven.**

# Gebruikersregistratie

Het registratie component is van groot belang voor de site, omdat het kostbare privégegevens betreft en toegang geeft tot een groot deel van de functionaliteit van de applicatie. We vinden het belangrijk om dit dus niet alleen op e2e niveau te testen, maar ook op component niveau.

## Eisen

De eisen van de registratie tests zijn als volgt:

* Correcte registratie zou een succesvol bericht moeten geven
* Incorrecte registratie moet toepasselijke error codes weergeven
* Onverwachte errors moeten op een gebruiksvriendelijke wijze worden afgehandeld

## De tests

Op basis van de testeisen hebben we de volgende tests opgesteld:

* Empty field test
* Should validate first name length
* Should validate last name length
* Should only allow alphanumeric characters in first name
* Should check email validity
* Should check password requirements
* Should check password and confirm match
* Should allow valid registration
* Should give unexpected error message