# Aantal personen: 2-3

# Inleiding

Swoppr biedt wijken de mogelijkheid om voor een jaarlijkse prijs gebruik te maken van huis- en tuinmateriaal. Op een centrale plaats in jullie wijk kan u beurtelings via onze website materiaal lenen. Wij bieden jullie de kans om materiaal te gebruiken zonder noodzaak om het daadwerkelijk te kopen. Ook bieden wij jullie de mogelijkheid om via een aanvraag materiaal uit te lenen dat je voor een langere tijd niet nodig hebt.

# Vereisten

## Front End

De front van de applicatie bestaat uit verschillende delen.

Niet ingelogd

* Pagina die de werking van de applicatie beschrijft. Maak dit visueel, zorg dat dit niet een gewone tekstpagina is
* Registratie van inwoners die gebruik willen maken van de applicatie

Ingelogd als uitlener

* Welkom pagina met een summier maar duidelijk overzicht van de meest recent toegevoegde uit te lenen materialen. Ook het materiaal die momenteel uitgeleend wordt, is zichtbaar
* Pagina waar je de uit te lenen goederen kan vinden op basis van uitgebreide filters
* Overzicht van de goederen die jij uitgeleend hebt
* In detailpagina periode kiezen (dagen + uren) om materialen uit te lenen
* Aanvragen om nieuw materiaal toe te voegen (eigen materiaal)
* Overzicht van alle uitleners op de map waarbij de gebruiker kan zien welk materieel er momenteel uitgeleend wordt

Administrator

* Een aantal statistieken

## Back End

* Alle data wordt opgeslagen en is voorzien van een timestamp
  + Belangrijkste data is de periode waarin het verhuurd wordt
* Registratie via Facebook of Google
* Realtime informatie over de beschikbaarheid van de uitgeleende materialen
* Een administrator kan na inloggen een aantal statistieken opvragen: wie leent het meeste zaken uit, welke materiaal of type wordt het meest uitgeleend, gemiddelde periodes
* Realtime communicatie met beheerder om breuken of verdachte zaken te melden

# Uitwerking

## Te gebruiken technologieën

|  |  |
| --- | --- |
| **Front End**  8pt van de 20pt | **Back End**\*  7pt van de 20pt |
| * CSS Preprocessor, * taskRunnner, * front -end testing, * LocalStorage, * Maps, * gebruik van framework naar keuze (angular). | * volledig asynchroon in javascript – node.js, * persistentie in een NoSQL database * (mongodb), * sockets voor multi-user integratie(socket-io), * autorisatie, authenticatie, * minstens één integratie en één unittest(jade), * deployment op cloud naar keuze(azure, * heroku), * gebruik van framework naar keuze (express).   \* Default zaken, zoals gezien tijdens de lessen, staan tussen haakjes maar mogen aangepast worden. |

Noot:

* Wie geen Front End volgt kan gebruik maken van een standaard HTML, Bootstrap of Express lay-out.

Eventueel kan een student, die enkel Front End volgt, deze front-end aanmaken als project.

* Wie geen Back End volgt kan alle data stockeren in een lokale JSON file.

Eventueel kan een student, die enkel Back End volgt, deze backend aanmaken als project

## Vereisten

* Programmeer in het Engels
* Javascript volgens de regels van de kunst (jshint als evaluatie tool)
* CSS volgens de regels van de kunst (csslint als evaluatie tool)
* Conventies:
  + Gebruik de juiste css naamgevingen.
  + Gebruik duidelijke code.
* • Performantie:
  + minifying,
  + concat
  + performante selectoren,
  + performante javascript,
  + foutcontrole op kritische onderdelen,
  + zoveel mogelijk DRY en abstract (= vermijden van herhalingen)
* Mooie, acceptabele (UI) en gebruiksvriendelijke(UX) userinterface voor zowel mobile of desktop gebruik.
* Gedocumenteerd:
  + Waar nodig, als verduidelijking, in de code
  + Aanvullend document (min. 2 bladzijden) met de nodige paswoorden, online URL’s, werkverdeling, schatting van werkuren, grootste moeilijkheden(2) en successen(2), eventueel architectuur en database schema, eventuele referenties en persoonlijke conclusie
* Project beschikbaar op een source control systeem naar keuze.
* Indien datum: ten laatste afgewerkt in de week vóór het examen. Tussentijdse evaluaties zijn mogelijk.