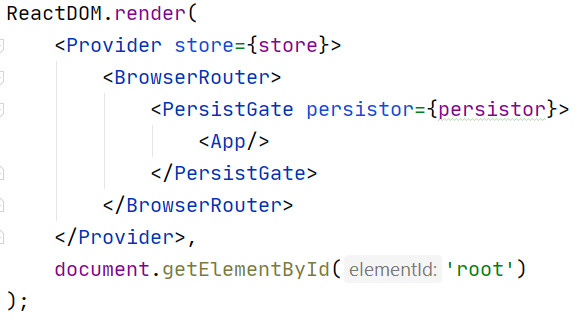
**III. React applicatie**

De gebruikte versie van React is 16.13.1. Npm of yarn kan gebruikt worden om de benodigde packages te installeren.

De gebruikte applicatie code om de tests uit te voeren is te vinden op GitHub[18]

De entry point van de React applicatie is index.js. Hier wordt de methode ReactDOM.render() opgeroepen met als belangrijkste argument het rootcomponent App.jsx. Het doel van ReactDOM.render() is het weergeven van de HTML code in het HTML element dat aan de methode is meegegeven. Ook staat er een Provider met de store voor het beschikbaar maken van de Redux store aan de applicatie, in de geneste componenten binnen App.jsx kan dan via de methode connect() toegang verkregen worden tot de store.

Tevens vinden we hier nog de BrowserRouter en de PersistGate. BrowserRouter zorgt ervoor dat de UI in sync blijft met de weergegeven URL, dit hebben we nodig want anders zou het in onze SPA het probleem geven dat gebruikers niet terug kunnen naar de vorige pagina. De PersistGate zorgt ervoor dat eerst veranderingen in state of properties worden gedaan en dat daarna pas elementen worden weergegeven.



De pagina’s en componenten in deze applicatie zijn opgedeeld en gestructureerd in 3 types.

* **Pages**: dit zijn pagina’s die we gebruiken in de applicatie
* **Shared**: In de shared folder zijn de componenten opgenomen die gedeeld en hergebruikt kunnen worden doorheen de volledige applicatie.
* **Redux**: hierin staat code betreft het definiëren van en reageren op acties van de gebruiker en het wijzigen en doorgeven van state en properties.

**Pages**

Pages zijn componenten die gelinkt zijn aan een bepaalde URL. We volgen hier dezelfde structuur als in Blazor alleen is de locatie van het declareren van de routes en de componenten verschillend. De route wordt gedeclareerd in een bovenliggend component en niet in het component zelf. In ons geval in “MainLayout".

Afbeelding met tafel, computer

Automatisch gegenereerde beschrijving

Via de detailpage wordt de parameter dan als volgt ingevuld en wordt er daarna een call gedaan naar de database om het JSON object op te halen.

Afbeelding met schermafbeelding

Automatisch gegenereerde beschrijving

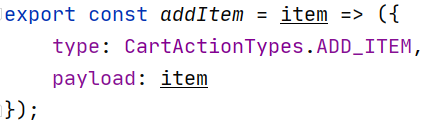
Ook in de React applicatie gebruiken we in de pages herbruikbare componenten uit de shared map om dubbele code te vermijden en het overzicht in de code te bewaren.

**Shared**

In de shared map vinden we net zoals in Blazor dezelfde herbruikbare componenten die we in de hele app kunnen gebruiken.

**Redux**

De map Redux bestaat niet in Blazor, maar doet in feite hetzelfde als de Models, Services en CartStateProvider. Dat komt omdat React is ontworpen om zich alleen bezig te huiden met het visualiseren van de DOM. Redux is een library waar React mee kan samenwerken. Redux is zo ontworpen dat de focus ligt op gebruiker, de acties die een gebruiker doet en de veranderingen die deze acties teweeg brengen. In de map Redux wordt de volledige state beheerd, ook worden hier gebruikersacties gedefinieerd en de veranderingen (payloads) die deze acties teweeg brengen.



Gebruikersacties zijn de enige bron van informatie voor de store. De store bevat de state, beheert de toegang tot de state en geeft de mogelijkheid om de state te updateten via de methode dispatch().

Afbeelding met schermafbeelding, vogel

Automatisch gegenereerde beschrijving

Met reducers kun je bepalen wat er precies verandert in de state wanneer een gebruiker een bepaalde actie uitvoert. Hieronder kunt u een voorbeeld vinden van de cartReducer.

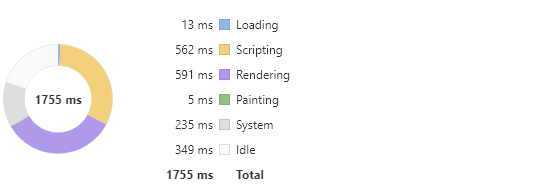
Afbeelding met schermafbeelding, vogel

Automatisch gegenereerde beschrijving

**Performantie testen**

Net zoals bij de Blazor applicatie definiëren we dezelfde acties.

* Home pagina:
  + Laden van de pagina met 460 producten



* + - DOMContentLoaded Event: 177,6ms
    - First Paint: 180,9ms
    - First Contentful Paint: 180,9ms
    - First Meaningful Paint: 1433,7ms
    - Onload Event: 178,2ms
  + Toevoegen aan winkelmand

Afbeelding met schermafbeelding

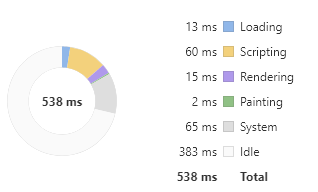
Automatisch gegenereerde beschrijving

* + Openen van product detailpagina door te klikken op product titel

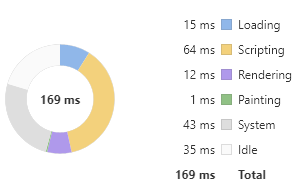
Afbeelding met schermafbeelding

Automatisch gegenereerde beschrijving

* Product detailpagina
  + Laden van de pagina



* + - DOMContentLoaded Event: 167,6ms
    - First Paint: 170,7ms
    - First Contentful Paint: 170,7ms
    - First Meaningful Paint: 246,5ms
    - Onload Event: 168,2ms
* Shopping cart pagina
  + Laden van de pagina met 100 producten

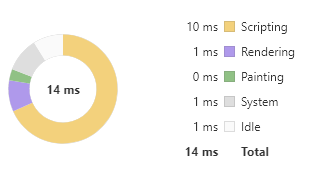


* + - DOMContentLoaded Event: 155,7ms
    - First Paint: 162,2ms
    - First Contentful Paint: 162,2ms
    - First Meaningful Paint: 162,2ms
    - Onload Event: 167,8ms
  + Verwijderen van producten uit de winkelmand

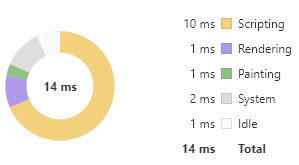
Afbeelding met tekening

Automatisch gegenereerde beschrijving

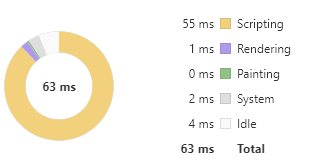
* + Aanpassen van de aantallen van een bepaald product in de winkelmand
    - Vermeerderen per stap van 1

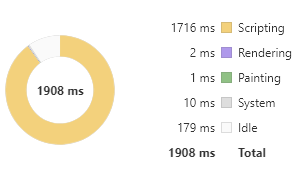


* + - Verminderen per stap van 1



* + - Vermeerderen door tekst input per stap van 1000,100.000, 1.000.000





Afbeelding met tekening

Automatisch gegenereerde beschrijving

* + - Verminderen door tekst input per stap van 1000,100.000, 1.000.000

