

## Eindtoets React (5 EC)

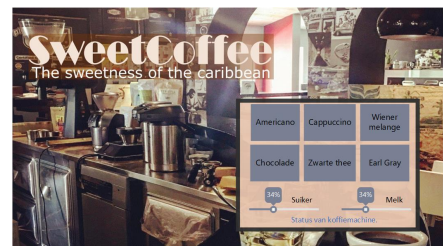
Deze leerlijn bestaat uit de volgende cursussen:

- Inleiding tot ReactJS (2 EC)
- Programmeren met ReactJS (3 EC)

### Omschrijving eindopdracht

De doelstelling van deze eindopdracht is het aantonen dat je de verworven inzichten en vaardigheden kan inzetten om een user interface te maken die door middel van React werkt.

De eindtoets bestaat uit twee use cases: **Selecteren van drank** en **Tonen foutstatus**. De use cases bestaan beide uit een hoofdsценario en alternatieve scenario's. De alternatieve scenario's zijn alternatieve paden die de software kan doorlopen. In sommige scenario's is er een preconditionie; dat wil zeggen dat er een conditie voorafgaand aan het scenario 'waar' moet zijn om het scenario mogelijk te maken. Als er geen preconditionie is vermeld dan is dat scenario altijd mogelijk. Daarnaast vind je in de opdrachtmap grafisch materiaal dat moet worden gebruikt.



### Doel van de opdracht en context.

De SweetCoffee Company is een fabrikant van koffiemachines die graag de kantoren koffiemarkt wil gaan betreden. Voor dat doel is er een koffieautomaat ontwikkeld die zelfstandig koffie zet. De bedoeling is dat deze automaat voorzien wordt van een 10,1" scherm (22,45 cm breed en 12,45 cm hoog, 1280 x 800 pixels). Door het team van IT-architecten binnen SweetCoffee is besloten dat de machinebesturing en bediening wordt gescheiden in twee computers. De besturingscomputer is niet beschikbaar en daarom wordt er een mock (vervangend object) beschikbaar gesteld.

### Voorwaarden eindopdracht

- De opdracht is geïmplementeerd in React waarbij er voor een van de user interface elementen gebruik mag worden gemaakt van een (1) component uit de npm bibliotheek.
- De opdracht maakt gebruik van de mock ('SweetCoffeeMock.js' in de opdrachtmap) en is gemaakt met javascript (ES6). Het gedrag van de mock mag worden aangepast, de interface niet. Gebruik de use cases om het gedrag te bepalen.
- Het resultaat van de opdracht is te gebruiken op de laatste versie van Node.js en is geschreven in JSX en javascript. Chrome is de browser waarin het resultaat van de opdracht wordt bekeken.
- Het resultaat van de opdracht is als project te importeren in Eclipse of Netbeans.

Voor een aantal van de criteria in de use cases is het belangrijk te weten hoe je tot een implementatie bent gekomen, de rationale van je keuzes. Op het moment dat je een keuze maakt (waarom doe ik dit zo?) voor een specifieke implementatie, leg dit dan vast in een verantwoordingsdocument.

Voor de testen is de tip om eerst de testen te maken en daarna pas de code. Op die manier denk je na over de code en heb je testen voor je code.

### Oplevering

- Een importeerbare zipbestand (max. 50 MB) in Netbeans of Eclipse als project. Gebruik hiervoor de exporteer functionaliteit uit het menu.

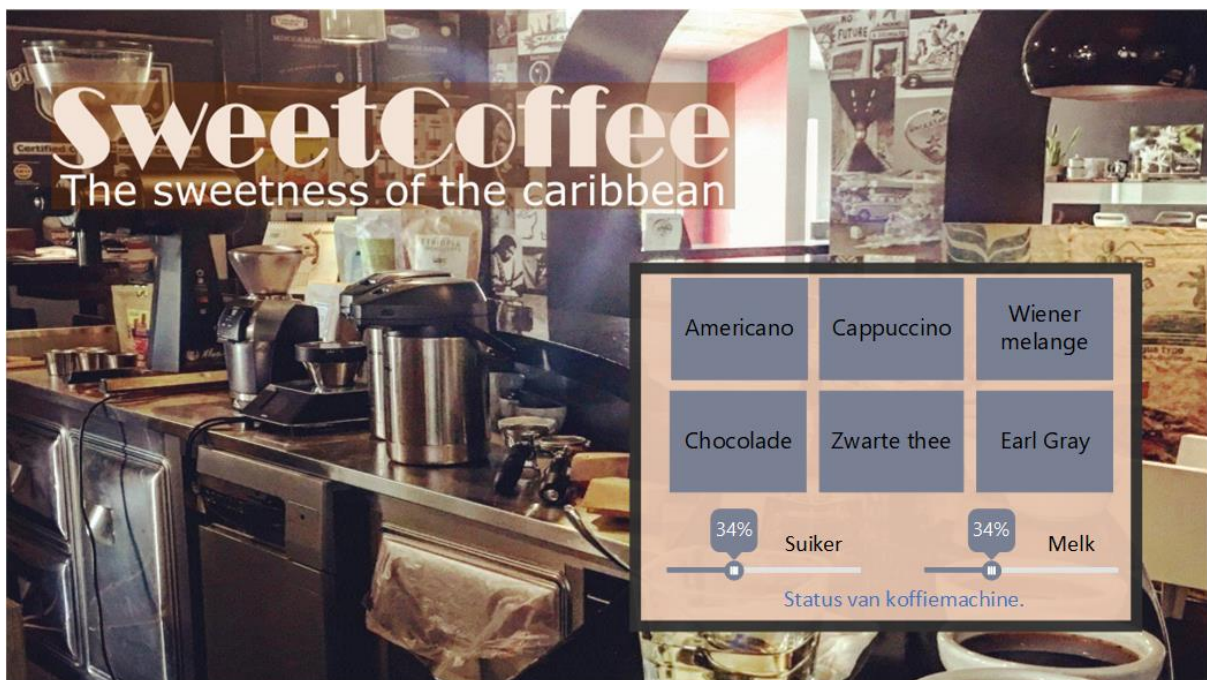
- Maak in de root van het project een map 'rationale' met daarin het verantwoordingsdocument in PDF of DOCX formaat. Het npm commando voor het te installeren component is in het verantwoordingsdocument ook opgenomen met de onderbouwing waarom deze geschikt is voor het gebruik in je applicatie.
- Maak in de root van het project een map 'testresultaat' met daarin het document (TXT) met de output van de testen die gemaakt zijn en uitgevoerd zijn.

## Use case 1: Selecteren van drank

### Doel van de use case 1

Door het kiezen van een drank wordt het mogelijk voor de gebruiker om te kiezen tussen de verschillende soorten koffie, thee en chocolademelk die de machine kan maken.

### Schermontwerp



### Hoofdscenario

1. De klant drukt op een van de knoppen die een drankje maken.
2. Alle user interface elementen worden gedisabled (onmogelijk gemaakt om te gebruiken).
3. Statusveld wordt "Machine maakt <gekozen drank>", gekozen drank wordt vervangen met de naam van de gekozen drank.
4. De machine maakt de gekozen drank.
5. Status geeft weer "Klaar voor keuze".
6. Alle user interface elementen worden enabled (weer 'aangezet').

Opmerking: Als de besturingscomputer van de koffiemachine bezig is met het maken van het drankje en er wordt een tweede opdracht tot het maken van een drankje gegeven dan resulteert dat in een "interne status fout" code 2 Error event (zie use case 2, hoofdscenario).

**Alternatief scenario 1**

1. Gebruiker stelt niveau van suiker of melk in.
2. (Naar hoofdscenario) De klant drukt op een van de knoppen die een drankje maken.

**Alternatief scenario 2**

Preconditie: Melk of suiker is op.

1. Sliders van melk of suiker staan op 0 en zijn disabled.
2. Knop Cappuccino is disabled.
3. (Naar hoofdscenario) De klant drukt op een van de knoppen die een drankje maken.

Opmerking: Kan gelijktijdig voorkomen met scenario 3.

**Alternatief scenario 3**

Preconditie: Chocolate is op.

1. Knoppen van Chocolate en Wiener melange zijn disabled.
2. (Naar hoofdscenario) De klant drukt op een van de knoppen die een drankje maken.

Opmerking: Kan gelijktijdig voorkomen met scenario 2.

## Use case 2: Tonen foutstatus

### Doel van de use case 2

Het doel van dit scenario is om als er een onoverkomelijke fout ontstaat in de besturingscomputer van de koffiemachine, het kiezen van dranken (use case 1) niet meer mogelijk wordt gemaakt.

### Schermontwerp



### Hoofdscenario

1. De besturingscomputer verstuurd het event "error".
2. Alle user interface elementen worden disabled.
3. Een foutmelding overlay wordt getoont.
4. De storingscode wordt vervangen door een gebruiksmelding.
- 5.

Code	Omschrijving	Gebruiksmelding
0	Geen storing	
1	Geen water	Geen waterdruk
2	Interne status fout	Machine is kapot
3	Temperatuur te laag	Machine is kapot

### **Alternatief scenario 1**

Preconditie: Na een storing wordt de storingscode 0 (geen storing) opgestuurd.

1. Foutmelding overlay wordt verwijderd.
2. Alle user interface elementen worden enabled.
3. Use case 1: Selecteren van drank wordt gestart.

## Beoordelingscriteria

Voor ieder criterium kun je een cijfer krijgen tussen de 1 en 10, afhankelijk van het niveau van de uitwerking van de eindopdracht. Een 8 wordt gezien als zeer goed. Een cijfer boven de 8 wordt alleen gegeven wanneer er boven het gevraagde niveau opgeleverd wordt.

#	Beoordeling leerlijn React	Weging	Score
1	De student kan schone React code schrijven door o.a. principes toe te passen die clean code dicteren.	20%	
2	De student toont aan componenten en properties op de juiste manier toe te kunnen passen.	30%	
3	De student kan React componenten op de juiste manier voorzien van interactie door middel van het toepassen van de daarbij horende React principes.	30%	
4	De student kan aan de hand van het Data Context Interaction principe zorg dragen voor ontkoppeling tussen React en de applicatieserver.	10%	
5	De student kan test software toepassen om React code te testen.	10%	
	<b>Totaal</b>	<b>100%</b>	

### Wijzigingen ten aanzien van versie 180921:

1. Verduidelijking onderdeel 'Doel van de opdracht en context'.