Testrapport

Door: Tim IJntema (1682222) en Arco Gelderblom(1679531)

Dit testrapport omvat de testen die zijn uitgevoerd voor het drankjes mix systeem. Het testrapport is gebaseerd op het testplan. Helaas was het echter niet mogelijk om alle testen die eerder in het testplan stonden uit te voeren. Dit kwam doordat gedurende de weken het project veranderde en sommige aannames van eerst niet meer klopten. Hieronder volgt een lijst van de testen die wij niet meer hebben kunnen uitvoeren per niveau.

* Unit niveau
  + Temperatuurregeling
  + Kleurregeling
* Subsysteem niveau
  + Heatertest
  + Volumetest
  + Valvetest
  + Colortest
* Systeemniveau
  + Temperatuurtest
  + Kleurtest

Er zijn verschillende redenen waarom deze testen niet uitgevoerd kunnen worden. In het geval van de temperatuur en kleur is dit vrij eenvoudig. Er is toegegeven dat deze sensoren niet werkten of niet werkten naar behoren. Hierdoor werd het onmogelijk dit te testen. Verder was er eerst gezegd dat de valve en het volume (de pomp) geregeld werden met een bepaalde voltagewaarde. De subsysteem testen die hierbij horen kunnen we niet uitvoeren want daar werd ook een voltagewaarde verwacht terwijl de latere implementatie ervoor zorgden dat wij deze moesten besturen met een boolean waarde. Bij de test op unit niveau was dit niet gespecificeerd en kunnen wij deze test dus wel uitvoeren. Verder is de volumetest op systeemniveau wel mogelijk.

Hierdoor blijven er 2 testen over die wij wel kunnen uitvoeren. Op unit niveau de test genaamd volumeregeling en op systeemniveau de volumetest. Hieronder zullen de rapporten van deze testen staan.

# Volumeregeling

Black box unit level test

Test verantwoordelijke: Arco Gelderblom

## Doel

Controleren of de pompen correcte waarden krijgen wanneer er gevraagd word om water te pompen.

## Beschrijving

Wanneer er gevraagd word aan het systeem om water in de beker te pompen moet de waterpomp aan gaan en de watervalve uit. Als dit gebeurt is de test correct uitgevoerd.

Deze test word uitgevoerd zonder hardware verbindingen. Hij geld voor beide de limonade en de water pomp. Dit is doordat de 2 pompen door dezelfde functie aangezet worden.

## Methode

Voor deze test is er geen verbinding met de hardware. Het niet gebruiken van de hardware word gegarandeert door het op “False” zetten van de “gui.sharedVariables.usingHardware” variabele.

Doctest binnen de code op een specifieke unit. Binnen de file controller.py is er een functie genaamd ‘pumpTestIfOn’. Deze controleert of de waterpomp de juiste waarde krijgt wanneer er om water gevraagd word. Dit doet hij door aan te geven dat er water moet komen. Vervolgens voert het de functie uit die normaal uitgevoerd word elke x aantal tijd. Deze functie zou, als er om water of limonade gevraagd word, de water pomp aan moeten zetten en de valve uit. Vervolgens word er gechecked of de pomp inderdaad ook aan is gegaan.

Om de test uit te voerden, open command line tool, navigeer naar de locatie van de simulator folder via het “cd” commando en start de controller.py met het commando “python controller.py”. Zorg wel dat je python 3 hebt. In de command line tool zou de test uitkomst moeten zien (zelfde als bij het kopje resultaten).

## Resultaten

Het resultaat wat wij verkregen was altijd het juiste gebaseerd op de verwachte resultaten. Dus True (1) wanneer de pomp aan hoort te zijn en een False (0) wanneer de pomp uit hoort te zijn. Hier word automatisch voor getest door Doctest. De uitkomst die zichtbaar word in je commandline tool wanneer je de test runt staat hier onder.

C:\Users\Arco\Documents\practicum-atp\arcoentim\simulator>python controller.py

TestResults(failed=0, attempted=3)

Zoals zichtbaar is in de output na het afspelen van de code, zijn alle tests correct uitgevoerd.

## Conclusie

Op basis van de resultaten kunnen wij concluderen dat de water pomp de juiste waarden krijgt.

# Volumetest

Black box system level test

Test verantwoordelijke: Tim IJntema

## Doel

Controleren of de verwachte hoeveelheid vloeistof geleverd wordt.

## Beschrijving

De gebruiker stelt een hoeveelheid in en ontvangt die hoeveelheid in de beker. In de simulator word dit zichtbaar gemaakt door bij het kopje “Level sensor: … ml”. Deze test word uitgevoerd op de hardware.

## Methode

De test wordt uitgevoerd door de waarden in te stellen op de hardware en vervolgens het programma zijn werk te laten doen. Er wordt consequent 10 mL ingesteld en nadat de applicatie klaar is met runnen wordt de hoeveelheid mL afgelezen uit de GUI van de simulator. Deze test word 5 keer herhaald. Een gemiddelde word genomen van de afgelezen waardes.

Om de test uit te voeren sluit je de hardware aan op de computer, je navigeert met “cd <pad>” naar de locatie van de simulator in een command line tool, en start de main.py met het commando “python main.py”. Voor deze test is het belangrijk om hardware support aan te zetten door de variabele “gui.sharedVariables.usingHardware” op True te zetten. Bij het starten is het mogelijk dat er een STOI error komt. Wanneer dit gebeurt, voer opnieuw het start command uit. Blijf dit doen totdat de de code start.

Wanneer het system gestart is zet je een kopje in de lemonator. Op het schermpje zou de tekst moeten veranderen van “Put cup in machine” naar:

“Start(C)

W(A): 0

L(B): 0

Change L”

Hierin staat L voor limonade en W voor water. Als je deze tekst ziet zit het systeem te wachten op een aantal ML voor limonade. Druk nu op 5 op het keypad om 5 ml limonade toe te voegen. Druk vervolgens op A en dan op 5. Nu zou er het volgende moeten staan:

“Start(C)

W(A): 5

L(B): 5

Change W”

Druk nu op C en het drankje word gemaakt. Als het systeem niet lijkt te reageren op keypad acties, moet je de keypad knop ingedrukt houden tot het schermpje leeggehaald word en nieuwe letters typt. Als het schermpje raar gaat doen (gekke tekens laat zien), reset dan het systeem en start opnieuw.

## Resultaten

De resultaten van de metingen zijn ondergebracht in de onderstaande tabel.

|  |  |  |
| --- | --- | --- |
|  | mL ingesteld | mL gemeten |
| Test 1 | 10 | 13 |
| Test 2 | 10 | 15 |
| Test 3 | 10 | 12 |
| Test 4 | 10 | 14 |
| Test 5 | 10 | 17 |

Het gemiddelde van de gemeten waarden is:

(13+15+12+11+17) / 5 = 13,6 mL

## Conclusie

De gemeten hoeveelheid 13,6 mL is meer dan de ingestelde hoeveelheid. Er bestaat een afwijking van 3,6. Interessant is dat geen enkel moment de werkelijke hoeveelheid ingestelde mL is uitgegeven door de lemonator. Dit kan mogelijk komen door de uitlees snelheid of reactiesnelheid van het systeem. Daarnaast is de sensor precizie een probleem in combinatie met water. Wat we proberen te meten is water wat langzaam aan dichter bij de sensor komt. Dit water zal soms iets plonsen of uitdijen. Hierdoor kan het zijn dat de sensor bij een keer lezen een verkeerde water krijgt.