calgon

Joost Wagensveld 1664713 Zehna van den Berg 1662506 Jessy Visch 1661709 Koen de Groot 1638079

Thema Opdracht 6 Technische Informatica

calgon | Nijenoord 1

2016

# Inhoudsopgave

[Inhoudsopgave 2](#_Toc440532061)

[1. Inleiding 4](#_Toc440532062)

[2. De wasmachine 5](#_Toc440532063)

[2.1 Algemeen 5](#_Toc440532064)

[2.1.1 De opdracht 5](#_Toc440532065)

[2.1.2 Bediening 5](#_Toc440532066)

[2.1.3 Communicatie van de web browser en wasmachine 5](#_Toc440532067)

[2.1.5 Uitvoering nodige stappen wasmachine 6](#_Toc440532068)

[2.1.6 Log systeem 6](#_Toc440532069)

[2.2 Screenshots van de webinterface 6](#_Toc440532070)

[3. Ontwikkelomgeving 7](#_Toc440532071)

[4. klassendiagram 8](#_Toc440532072)

[5. Toelichting klassendiagram 9](#_Toc440532073)

[6. Beschrijving verantwoordelijkheid klassen 9](#_Toc440532074)

[Wasprogrammacontroller 9](#_Toc440532075)

[SensorHandler 9](#_Toc440532076)

[UpdatingSensor 9](#_Toc440532077)

[WaterniveauSensor 9](#_Toc440532078)

[TemperatuurSensor 9](#_Toc440532079)

[DeurvergrendelSensor 9](#_Toc440532080)

[WMStatusSensor 9](#_Toc440532081)

[Sensor 9](#_Toc440532082)

[SensorListener 9](#_Toc440532083)

[StatusWeergaveController 9](#_Toc440532084)

[NoodstopController 10](#_Toc440532085)

[UART 10](#_Toc440532086)

[Wasprogramma 10](#_Toc440532087)

[Fase 10](#_Toc440532088)

[LogController 10](#_Toc440532089)

[ActivityLogItem 10](#_Toc440532090)

[SystemLogItem 10](#_Toc440532091)

[MessageBuffer 10](#_Toc440532092)

[MessageBroadcaster 10](#_Toc440532093)

[Websocket 10](#_Toc440532094)

[WebsocketListener 10](#_Toc440532095)

[MessageQueue 11](#_Toc440532096)

[7. Testen 12](#_Toc440532097)

[8. Conclusie 13](#_Toc440532098)

[9. Evaluatie 13](#_Toc440532099)

[10. Suggesties en aanbevelingen 14](#_Toc440532100)

[11. Bijlagen 15](#_Toc440532101)

# 1. Inleiding

We zullen het hier gaan hebben over alles dat van belang was voor het ontwikkelen van de over internet bestuurbare wasmachine. We zullen het hebben over de test die we hebben gedaan om tot bepaalde beslissingen te komen. Ook zullen we uitgebreid praten hoe de uiteindelijke bediening en werking van de wasmachine is geworden en de keuzes die we hier in gemaakt hebben. We hopen hier door een duidelijk beeld te geven over het proces dat wij als groep hebben doorlopen.

# 2. De wasmachine

## 2.1 Algemeen

in het volgende hoofdstuk zal er gekeken worden naar de eigenschapen en onderdelen van de wasmachine.

### 2.1.1 De opdracht

De opdracht die wij van swurl industries hebben gekregen. Was het ontwikkelen van software en een wasmachine.

Deze wasmachine moest op een voor de gebruiker eenvoudige manier bestuurbaar zijn over het internet. Hier voor moest er gebruik gemaakt worden van een web browser. Vanaf deze browser moest het vervolgens mogelijk zijn om een was programma te selecteren. Deze wasprogramma’s werden mee geleverd op de wasmachine. Wel moest het mogelijk zijn dat de gebruiker de tempratuur aan kon passen naar wat wenselijk was voor de was die zij wilde draaien. De gebruiker zou ook aan moeten kunnen geven dat de was om een bepaalde tijd moest starten.

De laatste Unique wens die swurl industries had voor deze wasmachine was dat er logs bijgehouden zou worden. Van deze logs zouden er 2 versies bestaan. Hier later meer over. Verder moet de wasmachine natuurlijk alles kunnen dat een gewonen machine kan.

### 2.1.2 Bediening

De bediening van de was machine word gedaan via het internet. Hier voor is er met HTML, CSS en javascript een webpagina gemaakt. Deze pagina voorzien van een gebruiksvriendelijke vormgeving en makkelijk te gebruiken bediening, zorgt er voor dat de gebruiker op eenvoud wijze het gewenste wasprogramma kan starten. Wij hebben qua vormgeving gekozen voor groten knoppen en niet overdreven veel informatie in een scherm. Dit vooral met het oog op tablet en smartphones.

### 2.1.3 Communicatie van de web browser en wasmachine

Gezien het mogelijk is om vanaf het web opdrachten te geven aan de wasmachine is het natuurlijk ook mogelijk om de wasmachine informatie te laten sturen naar het web. Zo is het dus mogelijk om op de zelfde plek als waar de gebruiker het wasprogramma heeft gestart te laten zien waar de machine mee bezig. Maar ook informatie als de huidige water tempratuur. De tijd die hij nog nodig verwacht te hebben.

### 2.1.5 Uitvoering nodige stappen wasmachine

### 2.1.6 Log systeem

Het log systeem bestaat uit 2 verschilden logs die bijgehouden worden.

Een die alleen maar de hoofd activiteiten van de wasmachine bijhoud. Deze is bedoeld voor de gebruiker om bijvoorbeeld terug te kunnen kijken wanneer welke was gedraaid was en hoe lang dat heeft geduurd.

Het anderen log dat bijgehouden zou worden is dat van het systeem. In dit log worden dus alle stappen die de wasmachine doorloopt bijgehouden stap voor stap word er dus vertelt wat hoe laat gestart is en hoe lang dit geduurd heeft. Dit log was niet bedoeld voor de gebruiker maar voor onderhoud/reparatie van het apparaat. Zo zou het dankzij dit log mogelijk moeten zijn dat een reparateur goed kan zien dat het verwarmingselement niet goed meer werkt omdat het verwarmen van het water steeds langer duurt.

### 

# 2.2 Screenshots van de webinterface

# 3. Ontwikkelomgeving

Een ontwikkelomgeving is alles van de computersoftware en de hulpmiddelen die de ontwikkelaar ondersteunen bij het ontwikkelen van de software.

De aspecten van een ontwikkelomgeving hangen af van de taken die moeten worden uitgevoerd. Er kan een onderscheidt gemaakt worden tussen het ontwerpen, programmeren en het testen van de software.

#### Ontwerpen

Bij het ontwerpen zal de omgeving vooral bestaan uit modelleringsgereedschappen en programma’s waarmee je modellen kan maken en aanpassen.

De modelleringsgereedschappen bieden ondersteuning in het maken van bijvoorbeeld een UML, klassendiagram en dergelijke. Deze worden ontworpen met behulp van het modellering programma software ideas modeler.

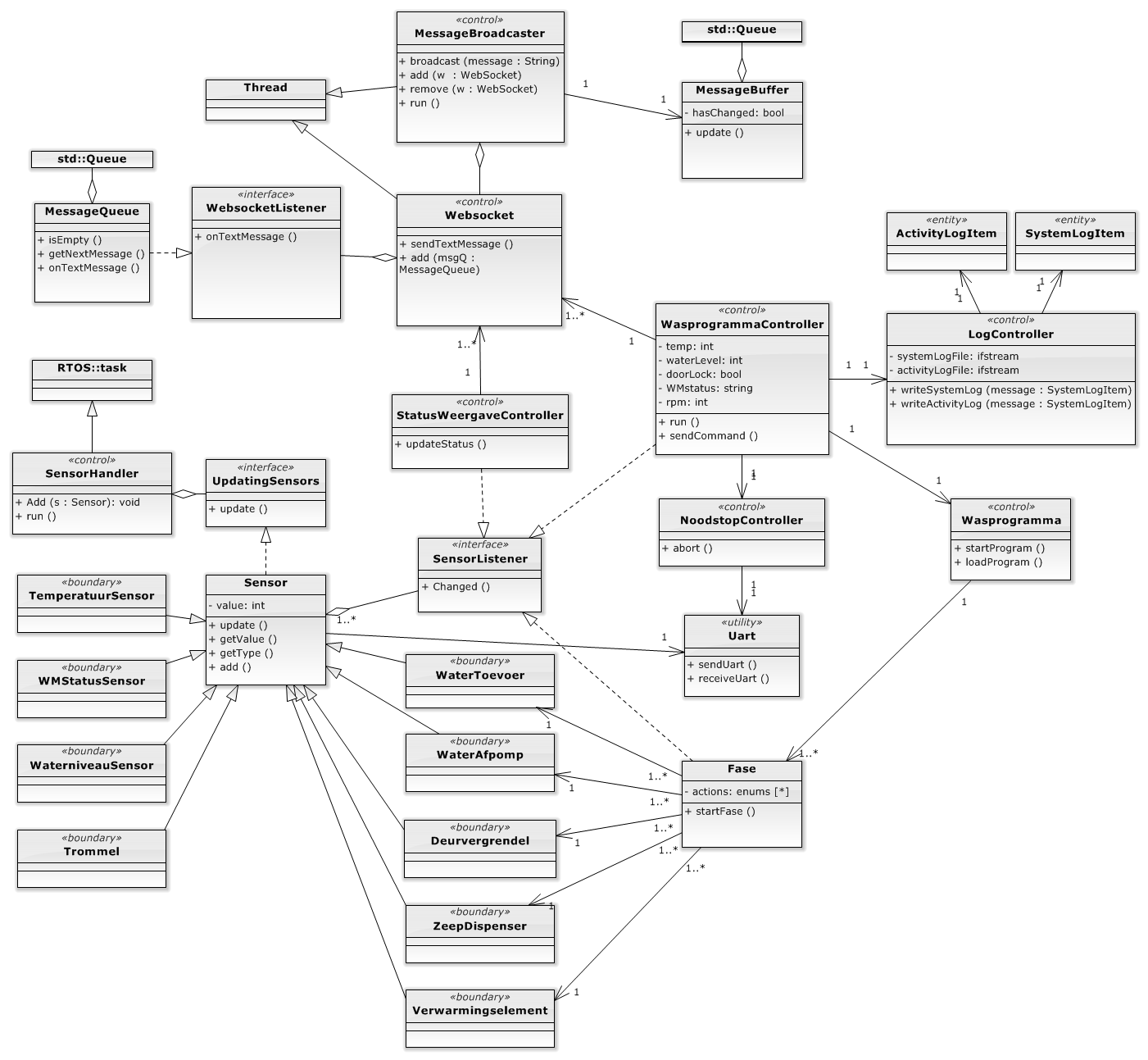
#### Programmeren

#### Testen

Bij het testen wordt de ontwikkelde software getest. Voldoet de test aan de eisen dan wordt deze versie opgeslagen. Is dit niet het geval dan wordt de code aangepast en weer getest. Dit wordt herhaald totdat het geteste item naar behoren werkt.

Ook wordt de code door meerdere teamleden bekeken en getest. Dit om er voor te zorgen dat de code niet alleen goed werkt maar ook volgens de afspraken is opgemaakt.

# 4. klassendiagram



# 5. Toelichting klassendiagram

# 6. Beschrijving verantwoordelijkheid klassen

### Wasprogrammacontroller

De klasse wasprogrammacontroller is verantwoordelijk voor het uitvoeren van het wasprogramma. Tevens is deze controller verantwoordelijk voor de communicatie met de websocket.

### SensorHandler

Deze klasse is verantwoordelijk voor het periodiek aan roepen van de update functie van alle child klassen van de interface UpdatingSensor. Ook is deze klasse verantwoordelijk voor het bijhouden van alle boundary objecten die hij moet aanroepen.

### UpdatingSensor

UpdatingSensor is de superklasse van alle sensor boundary klassen.

### WaterniveauSensor

Deze klasse is verantwoordelijk voor het pollen van de waterniveau sensor op aangeven van de SensorHandler.

### TemperatuurSensor

Deze klasse is verantwoordelijk voor het pollen van de temperatuur sensor op aangeven van de SensorHandler.

### DeurvergrendelSensor

Deze klasse is verantwoordelijk voor het pollen van de status van de deurvergrendeling. Tevens kan er via deze klasse de deur worden vergrendelt of ontgrendelt.

### WMStatusSensor

Deze klasse is verantwoordelijk voor het pollen van de status van de wasmachine op aangeven van de SensorHandler

### Sensor

Deze klasse is de superklasse van alle sensoren.

### SensorListener

Deze klasse is de interface die bijhoud of de sensoren nieuwe waarden hebben.

### StatusWeergaveController

Deze klasse zorgt er voor dat de juiste informatie door gegeven wordt aan de web interface zodat de informatie die hier word weergegeven up-to-date blijft.

### NoodstopController

Deze Klasse is verantwoordelijk voor het afhandelen van de noodstopprocedure. Dit houdt in dat het wasprogramma onderbroken word, het water uit de trommel wordt gepompt en de deur word ontgrendelt.

### UART

Deze klasse is verantwoordelijk voor het regelen van de communicatie tussen de sensor klassen en de wasmachine zijn UART interface.

### Wasprogramma

Houd bij welke fases er gedaan moeten worden betreffende het was programma dat gestart is en zorgt dat deze op de juiste moment gestart word.

### Fase

Word opgesteld en aan geroepen door het wasprogramma. Deze klassen weet wat er stap voor stap gedaan moet worden voor elke fase die binnen het wasprogramma nodig zijn.

### LogController

Zorgt dat de juiste data weg geschreven word naar de betreffende logs. Deze klassen krijgt de benodigde data binnen van ActivityLogItem en/of SystemLogItem

### ActivityLogItem

Krijgt binnen welke activiteit gedaan is en geeft dit door aan LogController zodat het weg geschreven kan worden naar het log.

### SystemLogItem

Zorgt er voor dat elke stap die het systeem door loopt op de juiste manier weg gezet kan worden in het systeem log door de LogController.

### MessageBuffer

Dit is een wacht rij met alle berichten die van af het RTOS bij de massageBroadcaster binnen komen.

### MessageBroadcaster

Roept tegen de klassen die hier naar luisteren dat er en wat voor bericht binnen is gekomen.

### Websocket

Schrijft naar de pool waar de RTOS opdrachten neer gezet worden zodat deze door de RTOS uitgelezen en uit gevoerd kunnen worden

### WebsocketListener

Zorgt er voor dat het verstuurde bericht van de websocket goed neer gezet word in de massageQueue

### MessageQueue

Een lijst met alle opdrachten (berichten) die naar RTOS door gegeven moeten worden om uitgevoerd te kunnen worden.

# 7. Testen

# 8. Conclusie

# 9. Evaluatie

# 10. Suggesties en aanbevelingen

# 11. Bijlagen