

# TH06 Team 12

## Plan van Aanpak

### Datum

4 december 2015

### Auteurs

Christiaan VAN DEN BERG 1660475

Aydin BIBER 1666849

Martijn VAN DIJK 1660713

Chiel DOUWES 1666311

### Docenten

Wouter VAN OOIJEN

Joost SCHALKEN

Marten WENSINK

Jan ZUURBIER



# Inhoudsopgave

<b>Inhoudsopgave</b>	<b>2</b>
<b>1 Inleiding</b>	<b>3</b>
<b>2 Producten</b>	<b>4</b>
2.1 Inleiding . . . . .	4
2.2 Inleiding . . . . .	4
<b>3 MoSCoW</b>	<b>6</b>
<b>4 Onderzoek</b>	<b>9</b>
4.1 Read/Write met RTOS . . . . .	9
4.2 Wasprogrammas . . . . .	9
4.3 Snelheid van de UI . . . . .	9
<b>5 Kwaliteitseisen</b>	<b>10</b>
<b>6 Methode van kwaliteitsbewaking</b>	<b>11</b>
6.1 Kwaliteit van de code . . . . .	11
6.2 Gebruik van standaarden . . . . .	12
6.3 Technische storingsen . . . . .	12
<b>7 Projectorganisatie</b>	<b>13</b>
<b>8 Projectactiviteiten</b>	<b>14</b>
8.1 Mijlpalen en Planning . . . . .	14
<b>9 Risico's</b>	<b>17</b>
9.1 Uival van de hardware . . . . .	17
<b>Bibliografie</b>	<b>18</b>

# 1 Inleiding

Er is door de Hogeschool Utrecht opdracht gegeven voor de ontwikkeling van een wasmachine die via een netwerkverbinding te bedienen is. Steeds meer apparaten hebben een internetverbinding, en zijn verbonden met andere apparaten. Blaauboer, 2014

De opdracht betreft het ontwikkelen van een prototype om in een kleine groep testers te evalueren wat de meerwaarde is van een wasmachine met internetverbinding is.

## 2 Producten

### 2.1 INLEIDING

Hier worden alle op te leveren producten behandeld die van toepassing zijn tot het project.

- Gebruikershandleiding wasmachine
- Gebruikershandleiding webinterface
- Teamcontract
- Github Repository
- Interview Opdrachtgever
- Plan van Aanpak
- Requirements Document
- Requirements Architecture
- Solution Architecture
- Technisch Verslag
- Eindproduct (software + hardware)

### 2.2 INLEIDING

In dit hoofdstuk worden alle op te leveren functionaliteiten behandeld die van toepassing zijn tot het project. Deze producten variëren van software en hardware tot documentatie en gemaakte notulen. Van deze producten wordt een prioriteiten lijst gemaakt en een lijst aan kwaliteitseisen waaraan deze documenten moeten voldoen. Tevens wordt besproken hoe de kwaliteit van deze producten zal worden bewaakt.

## 2.2. INLEIDING

---

Om er voor te zorgen dat de op te leveren producten goed leesbaar zijn worden er een aantal eisen gesteld aan deze producten.

- Documentatie mag geen 3 spelfouten bevatten op een enkele pagina. In dien dit wel het geval is wordt het document niet opgeleverd verklaard.
- De webinterface moet voldoen aan de door de MoSCoW beschreven eisen.

### 3 MoSCoW

In dit onderdeel bespreken wij de lijst van functionaliteiten die tijdens het interview met de klant zijn achterhaald. Wij maken een lijst van deze prioriteiten aan de hand van de MoSCoW methode. Deze methode houdt in dat functionaliteiten een letter krijgen toegewezen die aanduidt wat hun prioriteit is binnen het systeem. De aanduiding is als volgt:

- M Must Deze eis is verplicht voor een goede afronding van het project.
- S Should Deze eis moet er in komen maar het project komt niet in gevaar wanneer dit niet mogelijk is.
- C Could Deze eis zou kunnen worden toegevoegd indien de Must en Should eisen behaald zijn.
- W Would Indien er tijd extra over is kunnen deze eisen worden geïmplementeerd.

Functionaliteit	Beschrijving	Prioriteit
Starten & Stoppen machine	De machine mag er tot een half uur over doen om het wasprogramma te starten. Stoppen moet vrijwel direct zijn	M
Aanpassen temperatuur	De gebruiker moet de temperatuur van een wasprogramma moeten kunnen aanpassen aan de hand van beschikbare opties	M
Wasprogramma's	Er moeten standaard 3 wasprogramma's aanwezig zijn: Witte was, Fijne was en Bonte was	M
Toevoegen wasprogramma's	Iemand die in dienst is bij de klant kan indien gewenst een nieuw wasprogramma maken en toevoegen aan de lijst van beschikbare wasprogramma's	M

### 3. MOSCOW

---

Inhoud wasprogramma's	Wasprogramma's bevatten de volgende gegevens: Duur, Temperatuur, Voorkeurstemperatuur en Centrifugesnelheid	M
Updaten wasprogramma's	Het updaten van de lijst van wasprogramma's moet automatisch gaan	M
Inloggen webinterface	De gebruiker logt in op de webinterface door middel van een pincode	M
Inplannen wastaken	Wastaken moeten kunnen worden ingepland (wasprogramma uitvoeren naar X aantal uren)	M
Logbestanden	Er moeten logs worden bijgehouden van wat de gebruiker heeft gedaan	M
Inhoud logbestanden	De logbestanden moeten de volgende informatie tonen: Uren motor gedraaid, waterverbruik en stroomverbruik	M
Crashbeveiliging	Het systeem moet beveiligd zijn tegen crashes	M
Acties stroomuitval	Wanneer na een stroomuitval de stroom weer terug is moet het systeem automatisch verder gaan met het wasprogramma	M
Accepteren updates	De gebruiker moet per update aangeven of hij/zij deze update wilt ontvangen	S
Aanpassen pincode	De pincode moet door de gebruiker aanpasbaar zijn	S
Herstelcode pincode	Er moet een herstelcode beschikbaar zijn die bij het systeem wordt geleverd om de huidige pincode op te vragen	S
Kiezen pincode	De gebruiker kiest zelf een pincode bij het voor het eerst opstarten van de webapplicatie	S

### 3. MOSCOW

---

Opties	stroom- uitval	De gebruiker kan via een optie in de we- binterface kiezen of de machine na een stroomuitval automatisch verder gaat of dat het water wordt weggepompt	S
Meldingen	webinterface	De huidige tijd en temperatuur van het systeem moet worden getoond aan de gebruiker	S



## 4 Onderzoek

Om dit project te kunnen maken is er onderzoek naar diversen onderwerpen nodig. Zonder dit onderzoek mist er veel informatie die er later in het project nodig zijn om problemen te verhelpen of het project sneller te laten lopen. Hiervoor zijn de volgende onderwerpen die wij gaan onderzoeken.

### 4.1 READ/WRITE MET RTOS

Om bij te houden waar het systeem was na stroomuitval is het nodig om de informatie op te slaan in een log bestand, maar omdat het maken en schrijven in een bestand een actie van het OS is, kan dit het rest van het systeem in een sleep zetten. Om te onderzoeken of dit een probleem wordt met deadlines van het realtime systeem gaan wij onderzoeken hoe lang het duurt om een log file te maken, en dan meerdere keren het bestand openen, schrijven en daarna sluiten om te kijken hoe lang het duurt. Als deze periode niet de deadlines overschrijdt dan kunnen wij dit gebruiken in het systeem.

### 4.2 WASPROGRAMMAS

Om te zorgen dat er door specialisten wasprogrammas kunnen worden toegevoegd, moeten wij ons meer verdiepen in hoe de wasprogrammas in elkaar zitten en hoe de wasmachine precies werkt, dit wordt vooral veel opzoeken en zelf met een werkende wasmachine uittesten om te kijken hoe de programmas in elkaar zitten.

### 4.3 SNELHEID VAN DE UI

Wij moeten onderzoeken hoe snel het systeem kan reageren op een actie in de UI, zoals de wasmachine stoppen, starten, etc. dit kunnen wij doen door simpelweg te kijken naar het systeem nadat wij een bepaalde optie selecteren.

## 5 Kwaliteitseisen

Om er voor te zorgen dat de op te leveren producten goed leesbaar zijn worden er een aantal eisen gesteld aan deze producten.

- Documentatie mag geen 3 spelfouten bevatten op een enkele pagina. In dien dit wel het geval is wordt het document niet opgeleverd verklaard.
- De webinterface moet voldoen aan de door de MoSCoW beschreven eisen.

## 6 Methode van kwaliteitsbewaking

Om de kwaliteit van het eindproduct te verzekeren zullen er een aantal maatregelen worden genomen:

### 6.1 KWALITEIT VAN DE CODE

#### GEAUTOMATIZEERDE TESTS

Er zullen geautomatiseerde tests gemaakt worden om er zeker van te zijn dat de code in werkende staat is. Deze tests zullen automatisch verifiëren of de code de bedoelde functionaliteit heeft, en als dit niet zo is dan zullen de tests dit aangeven.

#### CODE REVIEWS

Er zullen code reviews gedaan worden door de verschillende leden van het team om te verzekeren dat de geschreven code aan alle standaarden voldoet. Op deze manier kunnen wij er allemaal zeker van zijn dat de code goed leesbaar is, en het door iedereen begrepen en aangepast kan worden in het geval van bijvoorbeeld een bug.

#### CODE REVIEWS VAN DOCENTEN

Omdat wij er zelf niet zeker van kunnen zijn dat onze code reviews goed uitgevoerd worden, zullen wij waar mogelijk onze code door docenten laten overzien. Door het gebruikmaken van een (meer ervaren) derde partij om de code na te kijken kunnen wij zeker zijn dat onze code geen onopgemerkte problemen heeft, en de kwaliteit van de code zal in het algemeen verbeterd worden.

## 6. METHODE VAN KWALITEITSBEWAKING

---

### 6.2 GEBRUIK VAN STANDAARDEN

#### L<sup>A</sup>T<sub>E</sub>X

Wij gebruiken L<sup>A</sup>T<sub>E</sub>X voor alle documenten rondom het project. De reden hiervoor is om de layout van de documenten te standardiseren, en het makkelijker te maken om veranderingen in de documenten in git te kunnen synchronizeren. Zo kunnen wij effectiever samenwerken, hoeven we minder tijd te besteden aan het formatteren en opstellen van de documentatie en wordt de kwaliteit van de documenten verbeterd.

#### CODE FORMATTING

Wij zullen standaarden opstellen met betrekking tot hoe de code geformatteerd moet worden, en deze standaarden zullen in de code review op gebracht worden.

### 6.3 TECHNISCHE STORINGEN

#### MANKEMENTEN AAN HARDWARE

##### **Raspberry Pi**

Bij uitval van de Raspberry Pi hebben meerdere teamleden een vervanging, en zullen deze gebruikt worden in plaats van de originele pi. Als de vervangende pi's van de teamleden niet beschikbaar zijn om een bepaalde reden, zullen wij ons inzetten om een nieuwe pi te regelen.

## 7 Projectorganisatie

(Verantwoordelijkheden teamleden) Voor het notuleren van vergaderingen en klantgesprekken is Aydin Biber aangewezen. Hij zal zijn notities op github zetten, waarna hij een uitgewerkte notule zal maken en deze ook op github plaatst. Martijn van Dijk is verantwoordelijk voor het maken van de Agenda's en het voorzitten van de vergaderingen. Agenda's moeten voor een vergadering beschikbaar zijn.

# 8 Projectactiviteiten

## 8.1 MIJLPALEN EN PLANNING

### MIJLPALEN

Hieronder vind u de planning voor het gehele project. De genoemde data zijn de deadlines voor oplevering en reflecteren niet de werkelijke inleverdatum.

### PLANNING

[illegible][illegible]

P2												0.0	0.0
P2												0.0	0.0
P2												0.0	0.0
P2												0.0	0.0
P2												0.0	0.0
P2												0.0	0.0
	Totaal	30.7	11.6	30.7	10.6	28.7	9.6	25.2	6.6	0.0	0.0	115.3	38.4



Teamcontract	11-11-2015 (Week 1)
Github Repo (met bijbehorende mappen)	11-11-2015 (Week 1)
Interview Opdrachtgever	30-11-2015 (Week 3)
Plan van Aanpak	02-12-2015 (Week 4)
Requirements Document (MoSCoW)	01-12-2015 (Week 4)
Requirements Architecture	09-12-2015 (Week 5)
Solution Architecture	16-12-2015 (Week 6)
Technisch Verslag	(Week 7 / Projectweek 1)
Eindproduct (software + hardware)	22-01-2016

## 9 Risico's

Er zijn verschillende risico's die het project in gevaar kunnen brengen. Hier worden die risico's genoemd. Zie de Kwaliteitsbewaking voor wat de gepaste reactie is op deze risico's.

### 9.1 UIVAL VAN DE HARDWARE

Het kan gebeuren dat de hardware niet goed werkt, of compleet uitvalt. Dit vormt een risico voor het project omdat het team dan niet goed de code kan testen.

# Bibliografie

Blaauboer, R. (2014, oktober 7). Wonen tussen 100 sensoren: het internet of things start bij jou thuis. <http://www.frankwatching.com/archive/2014/10/07/wonen-tussen-100-sensoren-het-internet-of-things-start-bij-jou-thuis/>.