Dit document beschrijft het solution architecture van de interface en de software van de nieuwste generatie wasmachines van Swirl Industries B.V.

Solution Architecture

*Easy interface, easy life.*

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

Team 11

Versie 0.1

15-12-2015

Inhoud

[1. Inleiding 2](#_Toc437641075)

[2. Use Case diagram 3](#_Toc437641076)

[2.1. Het diagram 3](#_Toc437641077)

[2.2. Use case beschrijving 3](#_Toc437641078)

[2.2.1. Programma uitvoeren 3](#_Toc437641079)

[2.2.2. Wasprogramma stoppen 3](#_Toc437641080)

[2.2.3. Update uitvoeren 4](#_Toc437641081)

[2.2.4. Activiteiten logs weergeven 4](#_Toc437641082)

[2.2.5. System logs weergeven 4](#_Toc437641083)

[3. Activity Diagram 5](#_Toc437641084)

[4. Constraints diagram 6](#_Toc437641085)

[Appendices 7](#_Toc437641086)

# Inleiding

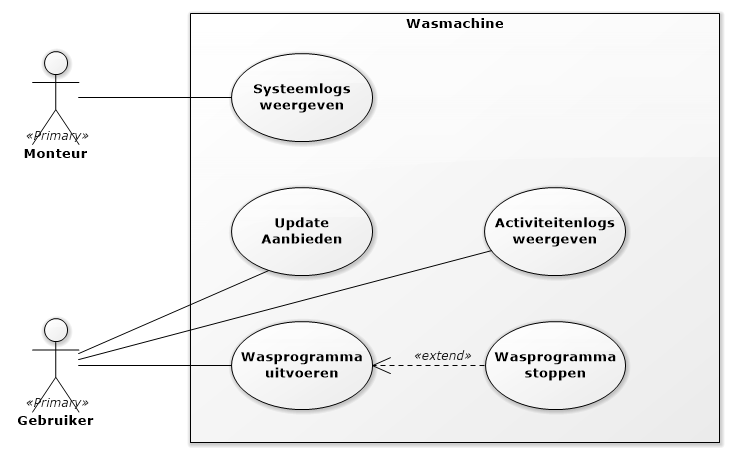
In dit document kunt u alle diagrammen terug vinden die behoren tot het requirements architecture. Een requirements architecture helpt bij het vinden van de antwoorden op de vraag: “Wat moet de software van het systeem doen?” Ook wordt vastgelegd hoe goed het systeem moet functioneren. Daarvoor hebben we het constraints model opgenomen in hoofdstuk 4.

Verder zal de requirements architecture als input dienen voor het maken van de solution architecture.

# Klassendiagram

Dit hoofdstuk beschrijft het klassen diagram. In de eerste paragraaf wordt het diagram getoond. Daarop volgend worden in de tweede paragraaf worden een aantal beschreven.

## Het diagram



## Beschrijving klassen

Dit paragraaf is voornamelijk een opsomming van de verschillende klassen die gebruikt zullen worden. Ook zal er een uitleg zijn van de klassen en bepaalde relaties tussen deze klassen.

### //klasse 1

### //klasse 2

### //klasse 3

### //klasse 4

### //klasse rest

# Concurrency Diagram

In dit hoofdstuk is het Concurrency diagram te zien. Dit spreekt redelijk voor zich.