

Containing



Gemaakt door:

Joshua Bergsma

Remco de Bruin

Melinda de Roo

Arjen Pander

Jeffrey Harders

Yme van der Graaf

Inhoudsopgave

1. Inleiding	3
2. IEEE	4
3. Testverslag voorbeeld	6

1. Inleiding

In het 2^{de} jaar van de informatica opleiding aan de NHL, krijgen wij als studenten een grote opdracht genaamd containing. De opdracht is aanzienlijk groter dan wat we gewend zijn, dus bijna alles moet daarom anders geregeld en gepland worden. Naast taakverdeling, grotere groepen en meer programmeer werk, is er ook een grotere kans op fouten, omdat veel code los van elkaar wordt geschreven, en dan later samen veel problemen op kan leveren. Om dit tegen te gaan is ons aangeraden veel meer met testen te werken. Dit doen wij onder andere met hulp van ATOS, en doormiddel van JUnit.

2. IEEE

Functies:

- Uitladen XML.
- Controller/server toet de gegevens van de XML real time.
- Controller/server stuurt XML door naar Simulator/client.
- Simulator/client ontvangt van de Controller/server.
- Simulator/client moet de gegevens van de Controller/server simuleren.
- Kranen moeten containers van schepen aftillen.
- AGV's moeten deze containers verplaatsen.
- Containers moeten van AGV's af worden gehaald en op voertuigen worden geplaatst.
- Containers moeten van AGV's af worden gehaald en op het opslag terrein worden geplaatst.
- AGV's moeten met container langzamer rijden dan zonder container.
- Er moeten 100 AGV's tegelijk kunnen werken.
- Er moet op een mobiel apparaat een snapshot van de data kunnen worden bekeken.
- De AGV's moeten de kortste/snelste route kiezen.

Wat te testen:

- Uitladen XML.
- Controller/server toet de gegevens van de XML real time.
- Controller/server stuurt XML door naar Simulator/client.
- Simulator/client ontvangt van de Controller/server.
- Kranen moeten containers van schepen aftillen.
- AGV's moeten deze containers verplaatsen.
- Containers moeten van AGV's af worden gehaald en op voertuigen worden geplaatst.
- Containers moeten van AGV's af worden gehaald en op het opslag terrein worden geplaatst.
- Er moeten 100 AGV's tegelijk kunnen werken.
- Er moet op een mobiel apparaat een snapshot van de data kunnen worden bekeken.

Wat niet te testen:

- *Simulator/client moet de gegevens van de Controller/server simuleren:*
Als dit niet werkt, zou er niks gebeuren, als het wel werkt, zou alles moeten doen wat het hoort te doen. Dit is niet echt een vorm van testen, aangezien je in 1 oog opslag ziet of het werkt of niet.
- *AGV's moeten met container langzamer rijden dan zonder container:*
Ook hier kun je niet goed testen of het werkt, je moet het gewoon zien aangezien we niet een snelheidsmeter willen inbouwen.
- *De AGV's moeten de kortste/snelste route kiezen:*
Wij schrijven deze code op zo'n manier dat hij alleen de beste route kan kiezen, wat zou betekenen dat hij niks doet als hij die route niet kiest, wat ook weer betekent dat je het niet echt test.

Aanpak:

Zijn er specifieke tools of software die gebruikt moeten worden?

- Java.
- JUnit.

Hebben deze programma's extra training nodig?

- Nee.

Welke variabelen worden verzameld?

- Snelheid van de voertuigen.
- Het aantal containers per platform.

Waar worden deze variabelen verzameld?

- In de Controller/server, sommigen doorgestuurd naar de Simulator/client en/of de Management Interface.

Hoeveel verschillende configuraties worden er getest?

- 7, namelijk door invoering van de 7 XML bestanden.

Hardware:

- De software moet werken op de computer van ieder groepslid van de projectgroep.

Software:

- Moet in Java worden geschreven en getest worden met JUnit.

Hoe worden negatieve of onlogische resultaten verwerkt?

- Er wordt voor gezorgd dat deze met exceptions worden opgevangen en aangegeven zodat wij die kunnen repareren.

Wanneer is het goed:

In onze ogen is alles voldoende, als wij de 7 aangeleverde XML bestanden probleemloos kunnen inladen en simuleren. Dit betekent ook dat alle programma's goed samenwerken, een acceptabele snelheid (geen lag) hebben en de Management Interface de statistieken toont.

3. Testverslag voorbeeld

Punten	Status (Open/Clos e)	Omschrijving (bij Open probleem) (bij Close eventueel opmerking)	Datum (datum van Close of wanneer dit punt in de planning staat)	Prioriteit (hoog, middel, laag)
Uitladen XML	OPEN	Wordt nog verbeterd.	07-12-13	HOOG
Controler toont de gegevens van de XML real time	OPEN	Wordt nog niet uitgebreid getoond.	04-01-14	MIDDEL
Controler/se rver stuurt XML door naar Simulator	OPEN	Wordt nog niet bruikbaar doorgestuurd.	07-12-13	HOOG
Simulator onvangt van Controller	OPEN	Wordt wel ontvangen, maar nog niet begrepen.	07-12-13	HOOG
Kranen moeten containers van schepen aftillen	OPEN	Kranen werken nog niet.	04-01-14	MIDDEL
AGV's moeten deze containers verplaatsen	OPEN	AGV's rijden, maar krijgen nog geen containers.	04-01-14	LAAG