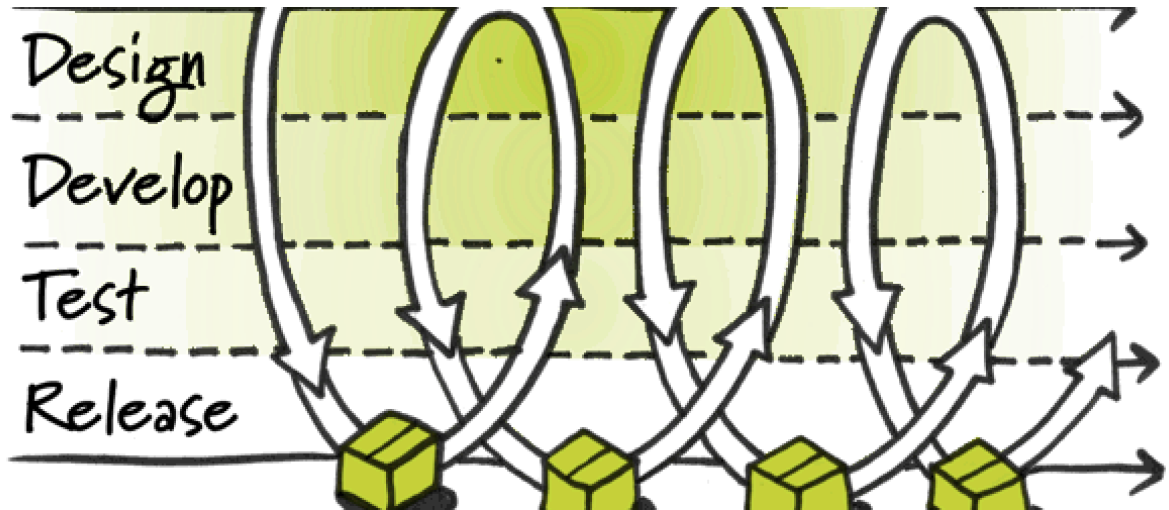


Testing Jaar 2

Opdracht 3: Testtooling



Frank Martens
500745737

Gijs Post
500745748

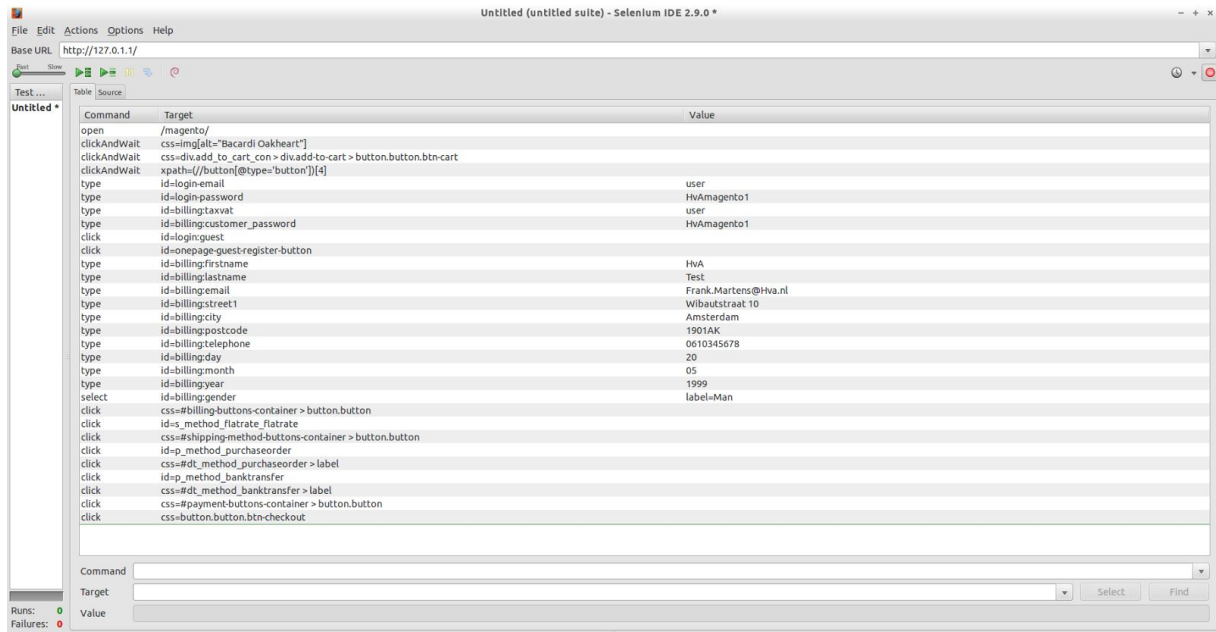
IS-205

Deel 1 - UI-testen met Selenium

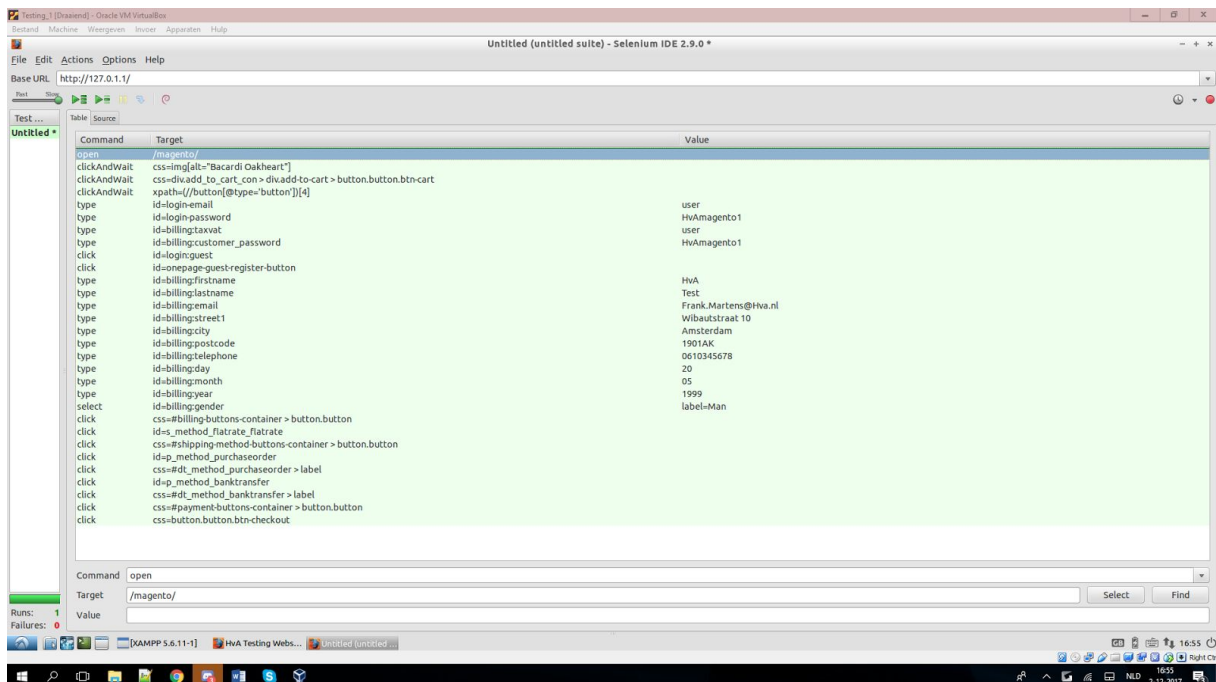
Deel 1 - Opdracht 1

Screenshot Deel A: Ga op de home page staan en neem op dat je een product koopt.

Hieronder staat een screenshot van Selenium:

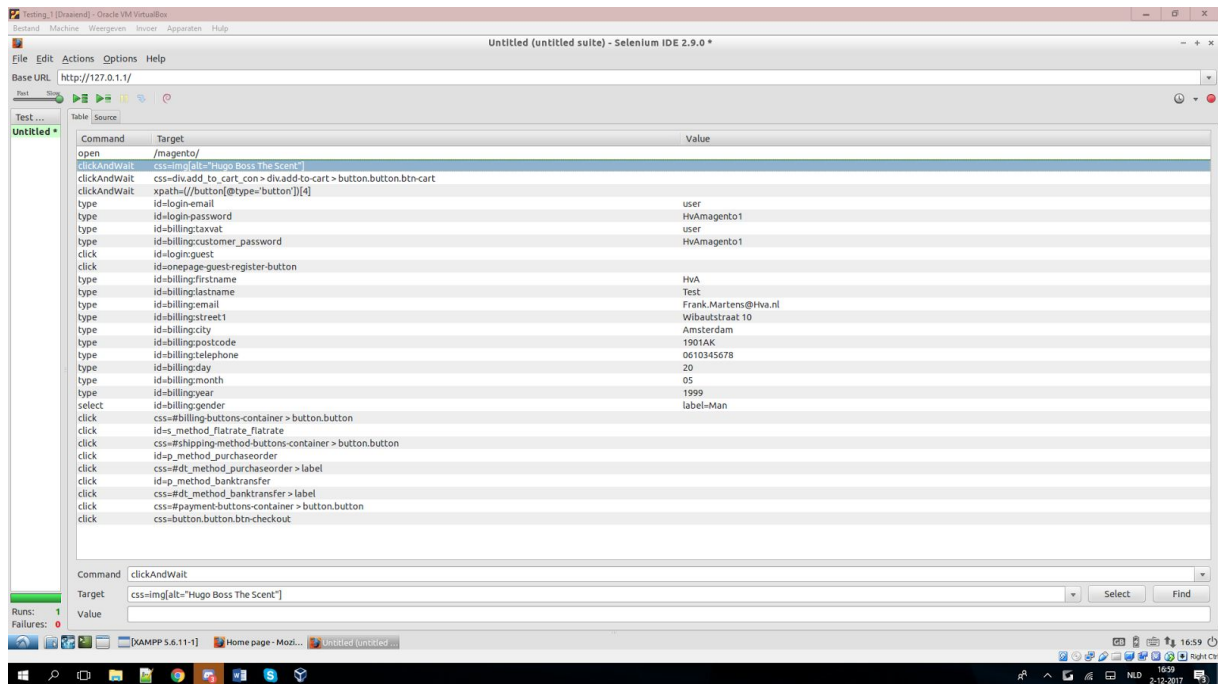


Screenshot Deel B: Zet de slider op “slow” en run het testgeval. Na het runnen van het testgeval ziet Selenium er als volgt uit:



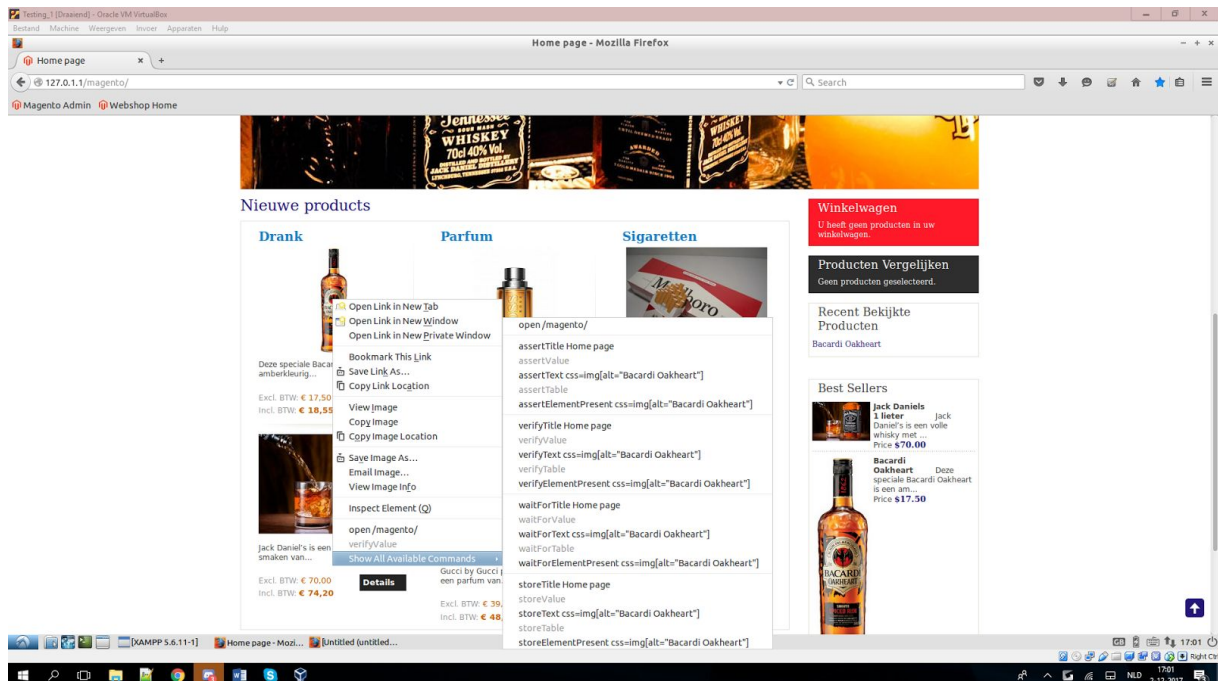
Screenshot Deel C: Pas het script zo aan dat je een ander product kunt kopen: Klik op het product en voor bij Target een ander product in.

Ik heb het product aangepast door “css=img[alt=”Bacardi Oakheart”]” aan te passen naar “css=img[alt=”Hugo Boss The Scent”]”, zoals te zien is op de volgende screenshot:



Deel D: Bekijken van de handleiding voor Record-and-play. We hebben de handleiding doorgelezen.

Deel E: Het bekijken van de Available Commands. We hebben de commands bekeken en gevonden, zoals te zien op onderstaande screenshot:



Deel 1 - Opdracht 2

Test 1: SEM-01 van Frank Martens

Screenshot van het resultaat:

Command	Target	Value
open	/magento/	
clickAndWait	link=Login	
type	id=pass	HvAmagento1
type	id=email	user
type	id=email	ongeldigemailadres
type	id=pass	testpassword
click	id=send2	
waitForText	id=advise-validate-email-email	Geef a.u.b. een geldig e-mailadres op. Bijvoorbeeld: iemand@domein.nl.

De Source is te vinden via: <https://pastebin.com/TNWbcAup>

Test 2: SEM-02 van Frank Martens

Screenshot van het resultaat:

Command	Target	Value
open	/magento/drunk/jack-daniels-1-liter	
click	css=button.button.btn-cart	
verifyText	id=advise-validate-one-required-by-name-options_1_2	Selecteer a.u.b. één van de opties.

De source is te vinden via: <https://pastebin.com/bpWPGgNu>

Test 3: SEM-03 van Frank Martens

Screenshot van het resultaat:

Command	Target	Value
open	/magento/drunk/jack-daniels-1-liter	
clickAndWait	link=Contact	
type	id=name	Frank
type	id=email	Frank@ditdomeinbestaatnietkjfdkf.nl
type	id=comment	Test Bericht
clickAndWait	css=div.buttons-set > button.button	
verifyText	css=li > span	Er is geen mailbox gevonden op het opgegeven emailadres

De source is te vinden via: <https://pastebin.com/jwJHxEWL>

Test 4: SEM-02 van Gijs Post

Screenshot van het resultaat:

Command	Target	Value
open	/magento/contacts/index/	
type	id=search	Jack Daniels
clickAndWait	css=button.button	
verifyText	css=img[alt="Jack Daniels 1 liter"]	

De source is te vinden via: <https://pastebin.com/T4GkesX5>

Test 5: SEM-04 van Gijs Post

Screenshot van het resultaat:

Command	Target	Value
open	/magento/catalogsearch/result/?q=Jack+Daniels	
clickAndWait	link=Login	
type	id=pass	HvAmagento1
type	id=email	user
clickAndWait	css=button.button.right	
type	id=taxvat	user
type	id=password	HvAmagento1
type	id=firstname	Frank
type	id=lastname	Martens
type	id=email_address	frank.martens@hva.nl
type	id=day	20
type	id=month	05
type	id=year	1999
type	id=password	test123
type	id=confirmation	test123
clickAndWait	css=div.buttons-set > button.button	
clickAndWait	link=Logout	
clickAndWait	link=Login	
type	id=pass	HvAmagento1
type	id=email	frank.martens@hva.nl
type	id=pass	test123
clickAndWait	id=send2	
clickAndWait	link=Mijn account	
verifyText	css=div.box-content > p	Frank Martens frank.martens@hva.nl Wachtwoord wijzigen

De source is te vinden via: <https://pastebin.com/1Q8Mk5t7>

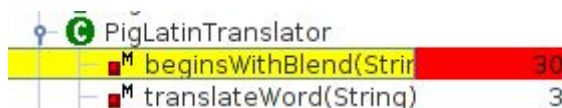
Deel 1 - Opdracht 3

Drie mogelijke gevolgen wanneer het ontwikkelteam de software heeft verbeterd:

- De link naar een webpagina kan veranderd zijn. Daardoor kan een bepaalde pagina mogelijk niet gevonden worden;
- Bepaalde attributen, zoals een id, class of alt kunnen veranderd zijn waardoor deze niet meer gelijk zijn aan de attribuutnamen in de test;
- Er kunnen onverwachte fouten optreden doordat er dubbel gegevens bekend zijn. Bijvoorbeeld bij het aanmaken van een account: Het emailadres bestaat dan al waardoor de test niet verder kan worden getest.

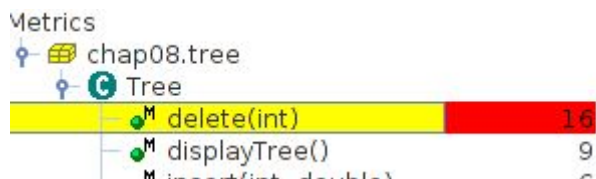
Deel 2 - Software metriecken

1. De methode met de grootste complexiteit.



Dit is de methode 'beginsWithBlend()' in de class 'PigLatinTranslator'. Dit komt doordat een bepaalde variabele vele keren vergeleken wordt, en zo dus veel verschillende mogelijke paden in de code maakt.

2. De methode met de grootste complexiteit in ADS.



Dit is de methode 'delete()' in de class 'Tree'. Deze methode is beter onderhoudbaar omdat deze een kleinere complexiteit heeft dan die uit de LewisCode.

3. Drie andere soorten metriecken.

- Executable statements. Hier wordt eigenlijk alleen gekeken naar het aantal statements dat het programma uit kan voeren. Hoe meer statements het programma heeft, hoe complexer het is. Dit werkt beter dan LOC (Lines of Code), omdat het niet op de programmeer-stijl let.
- RFC: Response for Class. Dit is het aantal mogelijke methodes dat uitgevoerd kan worden op een andere class dan waar de methode zelf staat. Dit kan gebruikt worden om de connectie tussen verschillende classes te weergeven, en hoe intensief deze connectie is.
- WMC: Weighted Methods per Class. Dit is eigenlijk de McCabe complexiteiten uit een hele class bij elkaar opgeteld. Zo kan je dus verschillende classes met elkaar vergelijken.

4. Leg uit hoe een tester in een agile ontwikkeltraject metriecken kan gebruiken om te bepalen wat hij/zij wel of niet gaat testen.

Het ligt aan de prioriteit van het testen. Een tester werkt natuurlijk altijd van hoogste naar laagste prioriteit. Hiervoor kunnen de metriecken gebruikt worden, om die prioriteit in de vorm van complexiteit te bepalen. Een stukje code dat volgens de metriecken een hele hoge complexiteit heeft, zal dus hoog aan de prioriteitenlijst staan van de tester.

Deel 3 – Agile software testing

1) Leg uit hoe je in een scrum project test-tooling kan inzetten om daarmee elke sprint structureel bij te dragen aan de kwaliteit van het shippable product.

Een tester moet bij scrum elke sprint het shippable product testen. Maar een tester is niet altijd alleen maar een tester. Vaak is het zo dat een programmeur uit het team de taak op zich neemt om het te testen. In dit geval kan de tester een programma zoals Refactorit gebruiken om na development snel en eenvoudig te zien hoe zijn code eruit ziet qua complexiteit. Zo kan hij rekening houden met deze metriecken als hij in de volgende sprint weer verder gaat aan deze code.

2) Beargumenteer in max. 50 woorden welke van de besproken agile aanpakken, pair programming of test driven development, het best zorgt voor software met de minste bugs.

TDD. Want na elke iteratie wordt er getest, en zo zorg je dat de code aan de eisen blijft voldoen. Bij Pair Programming kunnen beide developers de eisen niet goed nakomen, en kunnen ze de context (buitenom waar ze mee bezig zijn) van het programma vergeten, terwijl TDD hier rekening mee houdt.

3) Beschrijf wat belangrijk is om in een agile omgeving, tijdens een scrum project, te organiseren zodat de 4 waarden van het agile manifesto worden toegepast.

1. **Individuals and interactions:** Ieder teamlid een functie geven met een duidelijk doel dat aanvult op de rest van het team.
2. **Working software:** Elke sprint moet een Potentially Shippable Product opgeleverd worden.
3. **Customer collaboration:** Er moet een speciale rol binnen het scrum team komen die de brug is tussen het development team en de klant: De Product Owner.
4. **Responding to change:** De taken die gedaan moeten worden, moeten makkelijk onderverdeeld worden in Backlog items, die verschoven kunnen worden van prioriteit wanneer nodig.