**Bachelor toegepaste informatica**

Logboek – Software Testing

Vak : Software Testing

Docent : Serneels Frank

Jimmy Miels

R0295105

Odisee VZW (Odisee - Campus Brussel)

Stormstraat 2,

Brussel B-1000 Belgie

Version 1.0.0

27/09/2021

Inhoudstafel

[1. Cursus + bijkomende bronnen 3](#_Toc85642562)

[1.1 Udemy cursus 3](#_Toc85642563)

[1.2 Bijkomende bronnen 3](#_Toc85642564)

[2. Logboek 4](#_Toc85642565)

[2.1 20/09/2021 – 27/09/2021 ( 7 uur totaal) 4](#_Toc85642566)

[2.2 28/09/2021 – 04/10/2021 ( 5 uur totaal) 5](#_Toc85642567)

[2.3 05/10/2021 – 11/10/2021 ( 6 uur totaal) 7](#_Toc85642568)

[2.3 12/10/2021 – 18/10/2021 ( 8.30 uur totaal) 9](#_Toc85642569)

[2.4 19/10/2021 – 26/10/2021 ( uur totaal) 13](#_Toc85642570)

# Cursus + bijkomende bronnen

1.1 Udemy cursus

<https://www.udemy.com/course/complete-guide-to-unit-testing-in-net-core-nunit-xunit/>

1.2 Bijkomende bronnen

<https://docs.microsoft.com/en-us/dotnet/core/testing/unit-testing-with-dotnet-test>

<https://docs.google.com/document/d/1MrtJNc7Bb0ULezicfqjXDyuTozQB4jC-/edit>

Lock, A. (2018). *Asp.net Core in Action*. Manning.

Price, M. J. (2019). *C# 8.0 and .NET Core 3.0 - Modern Cross-Platform Development* (4de ed.). Van Haren Publishing.

*Comparing xUnit.net to other frameworks*. (z.d.). XUnit.Net. Geraadpleegd op 20 oktober 2021, van <https://xunit.net/docs/comparisons>

Mathurin, C. (2019, 30 maart). *NET Core 2: Why xUnit and not NUnit or MSTest*. DEV Community. Geraadpleegd op 20 oktober 2021, van <https://dev.to/hatsrumandcode/net-core-2-why-xunit-and-not-nunit-or-mstest--aei>

# Logboek

2.1 20/09/2021 – 27/09/2021 ( 7 uur totaal)

* 4 uur les
* 3 uur eigen tijd

Zoektocht naar gepaste testomgeving en bijbehorende cursus.

Cursus is goedgekeurd door docent.

Sotware Testing studiewijzer van pagina 4 tot en met pagina 12 gelezen.

**Eerste deel van eigen gekozen cursus doorgenomen.** Test omgeving ingesteld en nagegaan of alle versie up to date zijn.

Graphical user interface, application

Description automatically generated

2.2 28/09/2021 – 04/10/2021 ( 5 uur totaal)

**Tweede deel van online cursus gedaan. (Section 2 : MSTest Testing)**

Eerste Unit test uitgevoerd in Visual Studio C# .NET 5 – Mstest ( standaard meegeleverde Unit testing framework)

Link naar github waar de Testen van de online cursus zullen bijgehouden worden :

<https://github.com/M-JM/FirstUnitTest>

Graphical user interface, text, application

Description automatically generated

**Derde deel van de cursus gedaan ( Section 3 : NUnit Testing)**

Graphical user interface, text, application

Description automatically generated

Extra testen gemaakt om dieper kennis te vergaren van Nunit

Text

Description automatically generated

2.3 05/10/2021 – 11/10/2021 ( 6 uur totaal)

**Vierde deel van de cursus begonnen ( Section 4 : NUnit Testing – advanced concepts)**

Unit Testen zitten maken met collections and meerdere asserts.

MSDN documentatie doorgenomen om eens alle helpers van Assert te bekijken die ter beschikking gesteld zijn

<https://docs.microsoft.com/en-us/dotnet/api/nunit.framework.assert?view=xamarin-ios-sdk-12>

Graphical user interface, text, application, email

Description automatically generated

**Vierde deel van de cursus gedaan ( Section 4 : NUnit Testing – advanced concepts)**

Vierde deel afgewerkt. Unit Testen zitten schrijven voor Expections messages en objecten vergelijken.

Twee opdrachten volbracht en oplossingen vergeleken en commentaren in testproject voorzien.

Graphical user interface, application

Description automatically generated

**Hoofdstuk over Testing uit het boek “ASP.NET CORE in Action” – Andrew Lock (uitgever Manning) zitten doornemen. (XUNIT)**

2.3 12/10/2021 – 18/10/2021 ( 8.30 uur totaal)

**Gastcollege : Introductie tot Software Testing (Jürgen Meheus van M²Q)**

Gastcollege gevolgd (14:00 – 16:15).

Todo: Slides nogmaals bekijken want heel veel informatie was meegegeven.

**Vijfde deel van de cursus begonnen ( Section 5: MOQ with Unit Testing)**

Unit testen zitten maken voor een nieuwe Class die een interface implementeer.

Sinds de E-cursus een focus legt op Unit Testen en niet Integration Testen is het noodzakelijk om een valse implementatie (leeg object) te maken i.p.v. de effectieve Implementatie van de interface methods te gebruiken.

Voorbeelden uitgewerkt met eigen Fake implementatie & via een Mock library (Moq)

Graphical user interface, text, application

Description automatically generated

**Eigen implentatie van een Mock dependency:**

Unit Test vanuit BankAccountNUnitTest.Class:



Description automatically generated

Fake implementatie van Logbook Class vanuit LogBook.Class (eigen commentaar voorzien in code)

Text

Description automatically generated

**Moq implementatie van een Mock dependency:**

Unit Test vanuit BankAccountNUnitTest.Class:

Text

Description automatically generated

**Vijfde deel van de cursus verder gezet ( Section 5: MOQ with Unit Testing)**

Unit testen zitten maken om verdere functionaliteiten van de Moq library uit te testen.

**Text

Description automatically generated**

**A screenshot of a computer

Description automatically generated with medium confidence**

Graphical user interface, text, application, email

Description automatically generated

**Vijfde deel van de cursus gedaan ( Section 5 : MOQ with Unit Testing)**

Vijfde hoofdstuk afgewerkt, meerdere unit testen gemaakt om verschillende mogelijke returns van methods uit te testen.

Text

Description automatically generated

Graphical user interface, application

Description automatically generated

**Hoofdstuk over Testing uit het boek “ASP.NET CORE in Action” – Andrew Lock (uitgever Manning) afgewerkt ter voorbereiding van Section 6 van E-course (met uitzondering van de opdracht). (XUNIT)**

**Slides M²Q gastcollege nogmaals op mijn gemak bekijken**

2.4 19/10/2021 – 26/10/2021 ( uur totaal)

**Gastcollege : BDD ( Gregory Roosenboom)**

Gastcollege gevolgd (14:00 – 16:30).

Niet aanwezig in de Teams sessie sinds ik geen netwerk verbinding kon maken op campus. Via Samy zijn Teams mee kunnen volgen.

**Zesde deel van de cursus begonnen ( Section 6: XUnit)**

Test project met XUnit opgezet met alle nodige NugetPackages. Bestanden van de NUnit test project overgebracht en de nodige naamgeving veranderd. Test klassen – product en Calculator aangepast naar de syntax van Xunit.

Text

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

**Extra : NUnit vs XUnit**

Documentatie doorgenomen over de verschillen tussen de syntax van XUnit vs MSUnit en NUnit. Verder op het internet gezocht naar verdere verschillen tussen de twee test frameworks. Conclusie is dat XUnit minder verbose is en wendbaarder als NUnit dus dev. Hebben hier duidelijk een voorkeur voor.