

## **Polteq professionals Daniel Berendsen en Werner Oving namen deel aan de Leergang Future Proof Testing in Groningen en schreven een verslag over de module Java/C# voor testers.**

Op de eerste trainingsdag werden de Future Proof Testers opgedeeld in twee groepen: zes collega's gaan Java volgen en vijf C#.

Tijdens deze eerste dag leerden we werken met de IDE (Eclipse), we leerden projecten, packages en classes (met een mainmethode) aanmaken en we schreven onze eerste code met variabelen en diverse datatypes. Een van de trainers hamerde erop om de bal het werk te laten doen en niet zelf te gaan rennen, in andere woorden: zorg dat je IDE het werk doet door code suggestion and completion. Natuurlijk waren de meesten hardleers en waren 9 van de 10 errors te wijten aan typefoutjes.

Tijdens de training waren we veel zelf aan het werk, het was een prettige afwisseling van luisteren en direct toepassen. Aan het einde van de training kregen we een opdracht mee, we moesten zelf een game bouwen.

Iedereen heeft flink lopen stoeien om GuessTheWordGame correct draaiende te krijgen, het was interessant om de verschillende uitwerkingen en methodes te zien. Tijdens de eerste helft van de tweede dag gingen we verder met objectgeoriënteerd programmeren. Dat classes blauwdrukken zijn en objecten uitwerkingen van die blauwdrukken werd er flink ingestampt door de trainers. Na het avondeten ging trainer Marthijs los met demo's. Inheritance, polymorphism, encapsulation abstraction werden behandeld. Ons hoofd tolt nog steeds van alle mogelijkheden.

In een andere ruimte werd een tweede groep onder kundige leiding van David Baak en Robby Wiegman meegenomen in de wondere wereld van C#. Sommigen hadden al wat ervaring, anderen kenden al Java en wilden nu eens C# bekijken.

We leerden in deze training de basisvaardigheden voor het schrijven van C#, het benutten van de ontwikkelomgeving, in dit geval Visual Studio, we maakten kennis met NUnit als testframework en uiteindelijk werd er wat dieper ingegaan op objectgeoriënteerd programmeren (OOP).

We begonnen met het gebruik van Visual Studio, het gebruik maken van de ingebouwde handigheidjes, het opzetten van een project, aanmaken van classes en het draaien van je programma.

Vervolgens werden de Variabelen en Datatypes uitgelegd, gevolgd door een oefening. We

kunnen nu variabelen declareren van verschillende datatypen en weergeven in een console.

Om deze variabelen te kunnen gebruiken, leerden we vervolgens methoden te maken. Dit is “where the magic happens”. Je kan in methoden allerlei logica schrijven, andere methoden aanroepen, waarden teruggeven en nog veel meer.

Het was een erg goede verhouding tussen theorie op de beamer en het maken van oefeningen om de theorie direct toe te passen, al dan niet ondersteund door collega’s David of Robby. De oefeningen werden vervolgens klassikaal besproken en uitgelegd.

Na een aantal hoofdstukken kunnen we al aardig uit de voeten met C#. We kunnen nu o.a. if/else logica gebruiken, for en while loops maken, datacollecties gebruiken en exceptions catchen.

Omdat we nu eenmaal testers zijn en we graag testen willen schrijven in C#, kwam ook NUnit uitgebreid aan bod. Dit is een unit-testing framework waarmee we testen kunnen draaien in Visual Studio. Nu kunnen we testen of onze methodes kloppen met de verwachte uitkomst.



Als huiswerk kregen we een opdracht mee waarin we een GuessTheWordGame programma schrijven. Het was ook bij de C# klas leuk om bij de bespreking te zien dat er veel verschillende manieren zijn om (ongeveer) dezelfde uitkomst te krijgen.

Op dag twee leerden we verder hoe we onze code netjes kunnen maken door te refactoren en hoe je effectief en efficiënt kunt programmeren door met objecten te werken en instanties aan te roepen in plaats van jezelf telkens te herhalen.

Het laatste deel van de avond ging over encapsulation, abstraction, inheritance en polymorphism. Gelukkig werd dit ook weer goed uitgelegd met duidelijke voorbeelden en oefeningen, want dit begon toch wel een beetje ingewikkeld te worden.

Al met al hebben we veel geleerd op deze training. Het helpt ons om onze opdrachten nog beter uit te voeren dan voorheen!

**Daniel Berendsen en Werner Oving**