Unit testen

Unittesten is een methode om software[modulen](https://nl.wikipedia.org/wiki/Module) of stukjes [broncode](https://nl.wikipedia.org/wiki/Broncode) (units) afzonderlijk te [testen](https://nl.wikipedia.org/wiki/Testen_(software)). Bij unittesten zal voor iedere unit een of meerdere tests ontwikkeld worden. Hierbij worden dan verschillende [testcases](https://nl.wikipedia.org/wiki/Testcases) doorlopen. In het ideale geval zijn alle testcases onafhankelijk van andere tests. Eventueel worden hiertoe zogenaamde [mockobjecten](https://nl.wikipedia.org/wiki/Mockobject" \o "Mockobject) gemaakt om de unittests gescheiden uit te kunnen voeren.

## Mockobject

Een mockobject is een (software)object speciaal gemaakt om de eigenschappen en gedragingen te simuleren van een of meerdere objecten tijdens een [softwaretest](https://nl.wikipedia.org/wiki/Software), zoals een [unittest](https://nl.wikipedia.org/wiki/Unittesten). Een mockobject is een [testtool](https://nl.wikipedia.org/wiki/Testtool).

# Doelstelling

Het doel van unittesten is om functionele units onafhankelijk van elkaar te kunnen testen op correcte werking. Deze gedachte vloeit voort uit [modulaire](https://nl.wikipedia.org/wiki/Modulair) [softwareontwikkeling](https://nl.wikipedia.org/wiki/Softwareontwikkeling).

**Unittest tegenover gebruikerstest**

Unit testen is typisch een taak voor de [softwareontwikkelaar](https://nl.wikipedia.org/wiki/Softwareontwikkelaar) of testteam. Dit in tegenstelling tot [eindgebruikertests](https://nl.wikipedia.org/wiki/Gebruiker" \o "Gebruiker). Ook het doel is anders.

* Met unit testen controleert de ontwikkelaar of delen van de software goed werken of blijven werken gegeven bepaalde invoer (correct of foutief).
* Met gebruikerstesten controleert men in samenwerking met de gebruiker (of alleen de gebruiker) of de software reageert op de manier zoals de gebruiker verwacht dat hij zou reageren.

Wanneer voordelen?

Waar uiten zich de voordelen van unit testen?

**Bij het aanpassen in de software**

Unit testen is gebaseerd op het gebruik van testcases. Deze worden voorafgaand aan de test opgesteld, meestal zelfs tijdens het ontwikkelen van de software. De verschillende testcases vormen op deze manier een omschrijving van een goede werking van de unit. Wanneer in een later stadium aanpassingen of correcties uitgevoerd moeten worden op de unit, is betrekkelijk eenvoudig na te gaan voor de ontwikkelaar of de wijzigingen de unit in het gedrag niet hebben beïnvloed. De test die dan wordt uitgevoerd noemt men [Regressietest](https://nl.wikipedia.org/wiki/Regressietest).

**Bij het integreren van softwaremodules**

Unittesten gebeurt op afzonderlijke units, meestal door de ontwikkelaar. Als de ontwikkelaar tevreden is wordt de unit opgeleverd. De volgende stap is het testen van de geïntegreerde units in de [Integratietests](https://nl.wikipedia.org/wiki/Integratietest).

**Bij het up-to-date houden van documentatie**

Documentatie bijhouden van software is een van de lastigste en vervelendste klusjes. Meestal wordt dit niet goed gedaan, wat resulteert in grote verschillen tussen de documentatie en de software. Unittests en testcase omschrijvingen zou men kunnen beschouwen als "live" documentatie. Iedere wijziging (groot of klein, ontwerpwijziging of bugfix) zal getest moeten worden. Wanneer men dit doet op basis van unittesten, dan zijn de testcases de meest recente beschrijvingen van de correcte werking van een unit en vormen zij dus de meest recente documentatie van de software.

Beperkingen

Een unittest is beperkt tot de unit zelf. Dit betekent dat niet goed getest is hoe goed het systeem werkt wanneer de units geïntegreerd met elkaar zijn. Andere manieren van testen zullen deze tekortkoming van unittesten moeten compenseren. Daarbij geldt, net als bij iedere manier van testen, dat het nooit mogelijk is om voor alle mogelijk scenario's testcases op te stellen en hierop te testen.

### Bronnen en meer informatie

<https://nl.wikipedia.org/wiki/Integratietest>

<https://nl.wikipedia.org/wiki/Systeemtest>

<https://geteasyqa.com/qa/best-test-case-templates-examples/>

<https://nl.wikipedia.org/wiki/Testcase>