



Stichting NIOC

Stichting NIOC en de NIOC kennisbank

Stichting NIOC (www.nioc.nl) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website www.nioc.nl ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2025, gehouden op donderdag 27 maart 2025 jl. en georganiseerd door Hogeschool Windesheim). Bij elkaar zo'n 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats op 18 maart 2027 in Arnhem en wordt georganiseerd door HAN University of Applied Sciences.

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga kennisbank@nioc.nl.

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.

Softwarekwaliteit

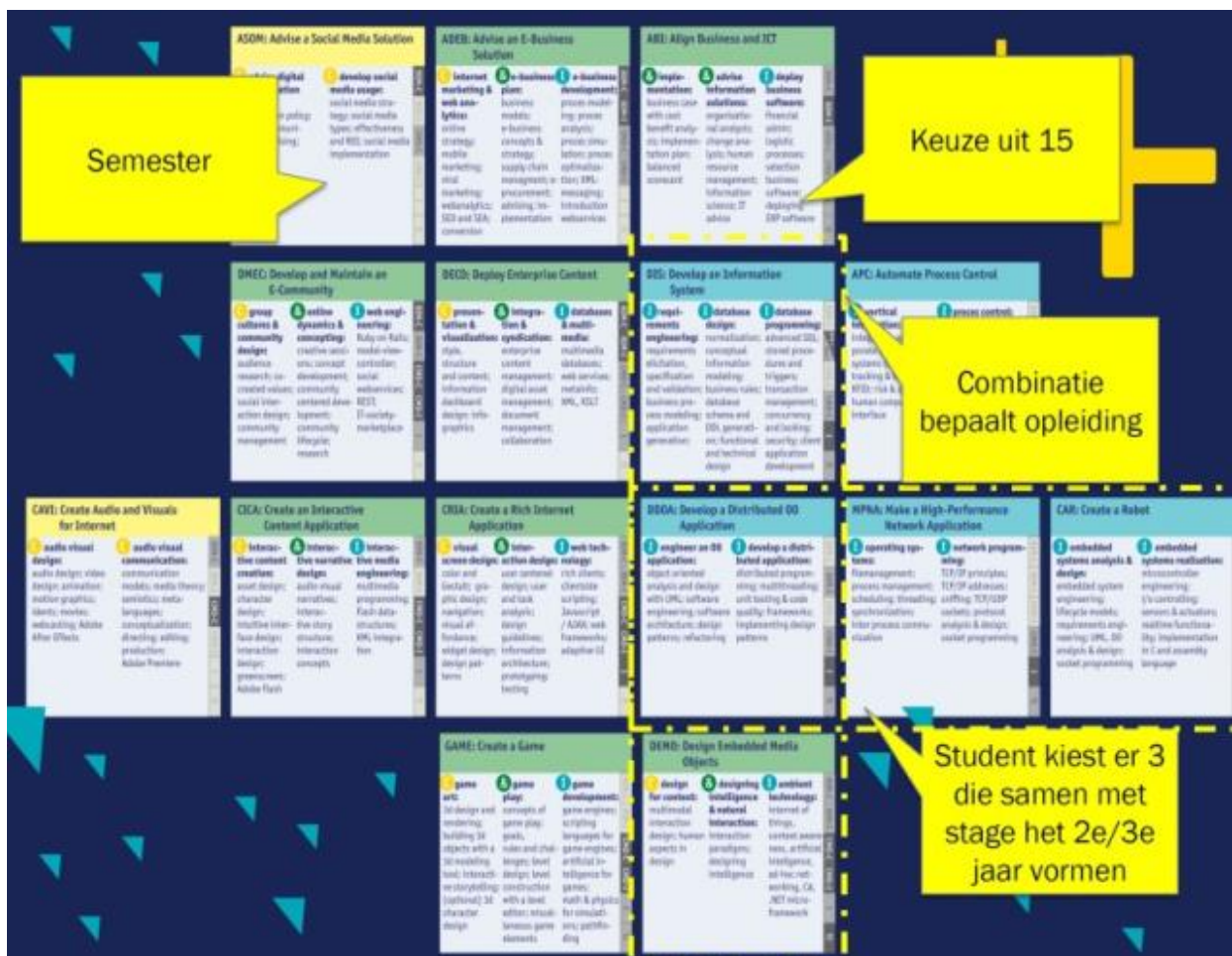
Softwarekwaliteit in een ontwikkelstraat: tools, processen, standaarden en richtlijnen – ervaringen na 8 jaar DDOA

Door: Rody Middelkoop.

Kernwoorden: Agile, Scrum, TDD, Confluence, Jira, Jenkins, Sonar, Open Source.

In deze presentatie toonde Middelkoop de aanpak en werkwijze binnen het onderwijssemester DDOA (Develop a Distributed OO Application). Het doel is om studenten te laten werken aan kwalitatief goede software voor externe opdrachtgevers. In de presentatie werden tools, standaarden, processen en richtlijnen gedemonstreerd en werd er gezamenlijk stilgestaan bij de keuzes van de ontwikkelstraat.

Vragen die aan de orde kwamen hadden te maken met organisatie en didactiek: afdwingen in kaders en vrijheid voor studenten; daarbij komen teamrollen en toetsing aan de orde. Voor de inhoud van het semester is de kernvraag: wat is kwaliteit (van software) en wat is goed genoeg als eindresultaten voor studenten in ‘leren en produceren’? Daarbij is ook de vraag: Hoe komen we aan opdrachten? Naast een ‘docentenblik’ kwamen ook enkele studenten aan bod die hun indruk gaven van het werken in een ontwikkelstraat.



Figuur 1. Semestermodel in ICA-opleidingen.

DDOA-Semesterstructuur.

De beroepstaak voor het DDOA-semester is: 'Ontwikkel een gedistribueerde OO-applicatie'
Deze beroepstaak richt zich op alle aspecten van een software engineeringsproject: analyseren, ontwerpen, bouwen, testen, implementeren. Figuur 2 toont de structuur en globale inhoud.

Course 1 (7,5 ECTS)	Course 2 (7,5 ECTS)
EOA - Engineer an OO Application: Object oriented analysis and design with UML; software engineering; software architecture; design patterns; refactoring.	DDA - Develop a Distributed Application: Develop a distributed application: distributed programming; multithreading; unit testing & code quality; frameworks; implementing design patterns.
Course 1 Inhoud: - OO, UML (class- and sequence-diagrams), Requirements, Design Patterns, Use Cases, Architectuur; - Applying UML and Patterns (Craig Larman); - Powerdesigner, Astah, Enterprise Architect.	Course 2 inhoud: -Parallel computing, Clean Code, Java, Threading, RMI, JavaEE, Big-O, Ant, JUnit / TDD, Frameworks; -Java Enterprise in a Nutshell (Jim Farley); -Eclipse, IntelliJ, NetBeans, SpringSource ToolSuite, Maven, Ant, SoapUI, GlassFish, Tomcat, Spring Roo, MySQL;
Project (15 ECTS): 10 weken in groepen van 5 studenten Aanpak: Agile, Scrum Inhoud: Gastcolleges en workshops van IT-bedrijven: InfoSupport, Luminis, VXCompany Projecten: Externe projectopdrachten voor IT-bedrijven: Avisi, First8, InfoSupport, CGI/Logica, Luminis, Sogyo, TASS, Twijnstra&Gudde, VXCompany	

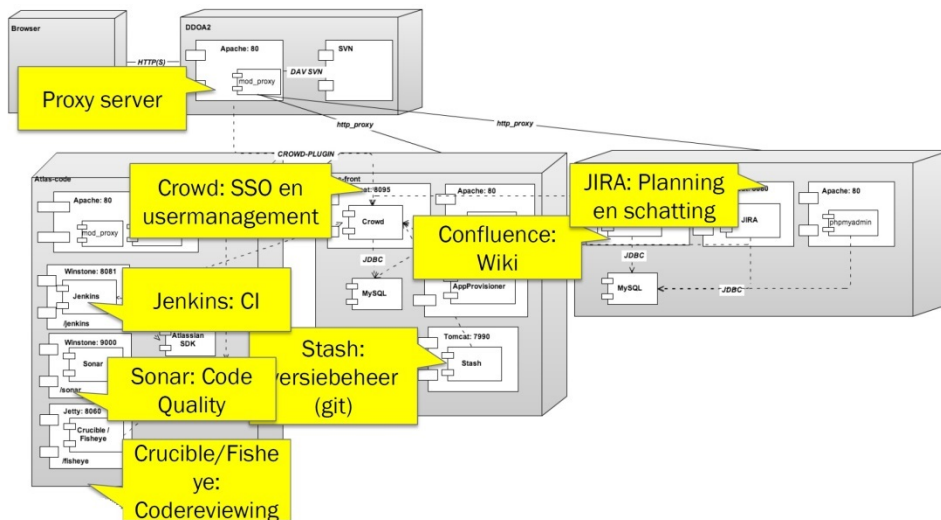
Figuur 2. Structuur en inhoudelijke aspecten van het semester DDOA.

Projecten

De inhoud van de projecten wordt ontwikkeld "van zaadje tot karbonaadje" door: **eerst overleggen** met de klant over de eisen en deze eisen vastleggen, vaak visueel; door nadenken over de structuur van je software; door ná het ontwerpen pas te gaan programmeren. Het einddoel is een werkende applicatie, vaak voor een echte klant uit het bedrijfsleven. Figuur 3 toont enkele projectvoorbeelden. Voor de softwareontwikkelstraat hanteert het DDOA-semester a.k.a. 'Software factory' standaarden en templates. Scrum wordt gebruikt voor het proces en de toegepaste tools zijn Atlassian Suite (Confluence, JIRA, Stash, Crowd), Jenkins en Sonar.

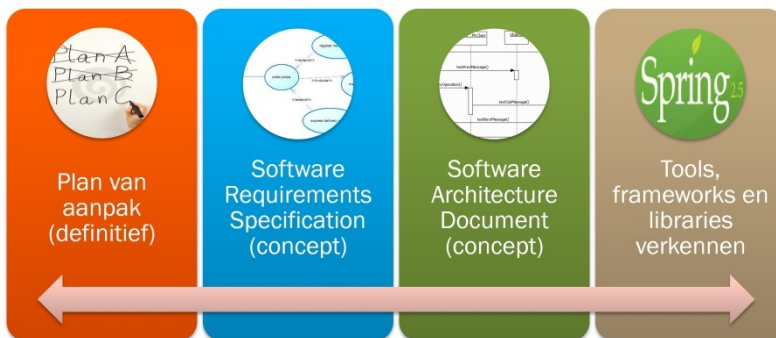
Project	Inhoud	Technologie	Bedrijf
IQ Cabin	Ontwikkel een Android-implementatie van het IQ Cabin-spel.	Android.	AVISI
Atlassian Provisioner	Ontwikkel product waarmee de ontwikkelstraat beschikbaar is voor nieuwe groepen.	Java, webservices (REST/SOAP).	HAN
Profielgenerator	Ontwikkel een applicatie waarmee CVs kunnen worden gegenereerd.	JavaScript, Spring Roo, AtlassianCrowd.	Luminis

Figuur 3. Voorbeeldprojecten in semester DDOA.



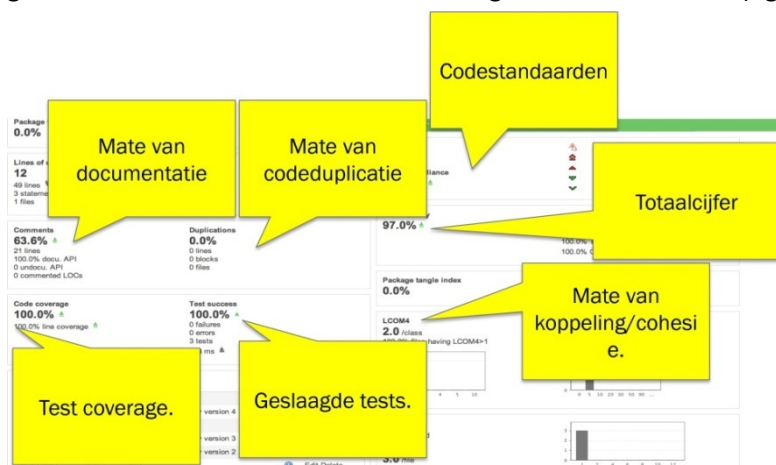
Figuur 4. Tooloverzicht en deployment.

Standaarden worden gehanteerd voor documentatie en codering (figuur 4). De documentatiestandaarden zijn geleend van Unified Process (RUP op maat), er zijn formats voor Plan van aanpak (Software Development Plan + Vision), Software Requirements Specification en het Software Architecture Document (figuur 5).



Figuur 5. Documentatiestandaarden RUP op maat.

Voor de codingstandaarden is er een vaste set voor DDOA, afwijkingen zijn bespreekbaar mits gemotiveerd en ze worden automatisch gecheckt door Sonar (figuur 6).



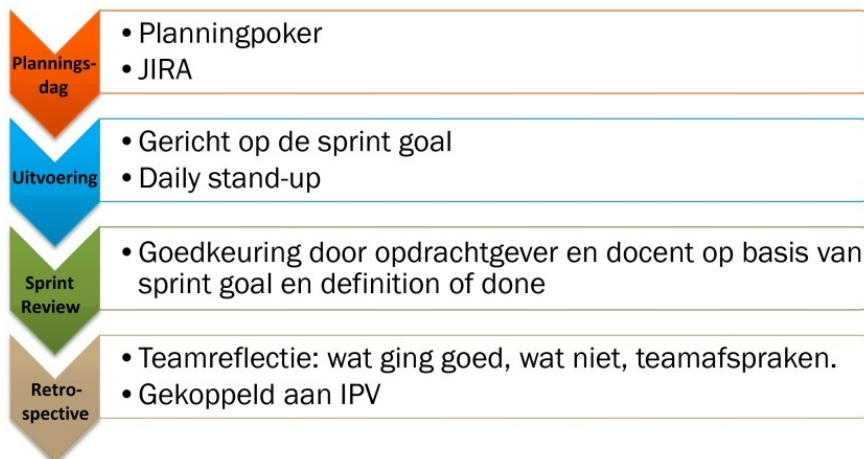
Figuur 6. Check op codingstandaarden met Sonar.

Alle technische documentatie wordt geschreven en bewaard in de tool Confluence. Het faciliteert een Wiki, Versiebeheer, Usertracking, geïntegreerde UML en UIMock. De studenten hebben schrijfrechten op hun eigen wiki en leesrechten in alle andere spaces.

Agile development

De software ontwikkeling is Agile met de specifieke kenmerken: “Early automated testing” per unit; “Incremental design” en geen groot ontwerp vooraf; “daily deployment” om elke dag te bouwen en op te leveren. “Customer involvement” eist veel contact met de klant en door “Continuous integration” wordt op de achtergrond de kwaliteit van ieders werk gecontroleerd. Als laatste is incremental planning van belang, omdat die regelmatig kan wijzigen.

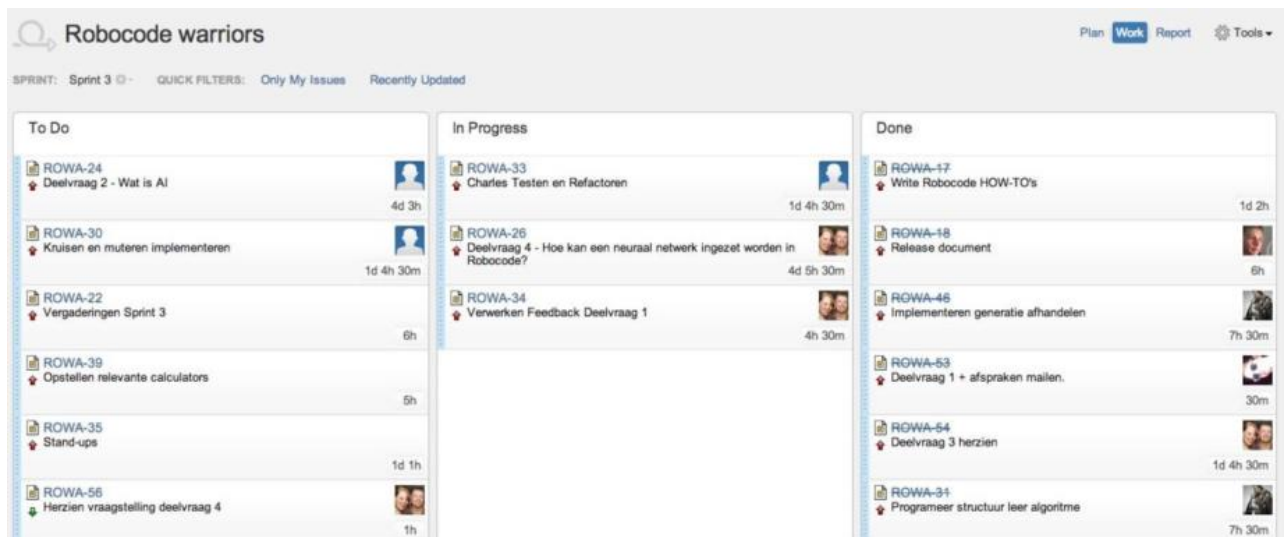
Het ontwikkelproces hanteert: “Scrum-planning” in: 1 week pre-game (‘warming up’); 4 sprints van 2 weken. Elke sprint begint met een planning dag met planningpoker, verwerkt de resultaten in JIRA; start de uitvoering en hanteert een Sprint Review en Retrospective (figuur 7). Tot slot is er 1 week afsluiting (‘cooling down’) gepland.



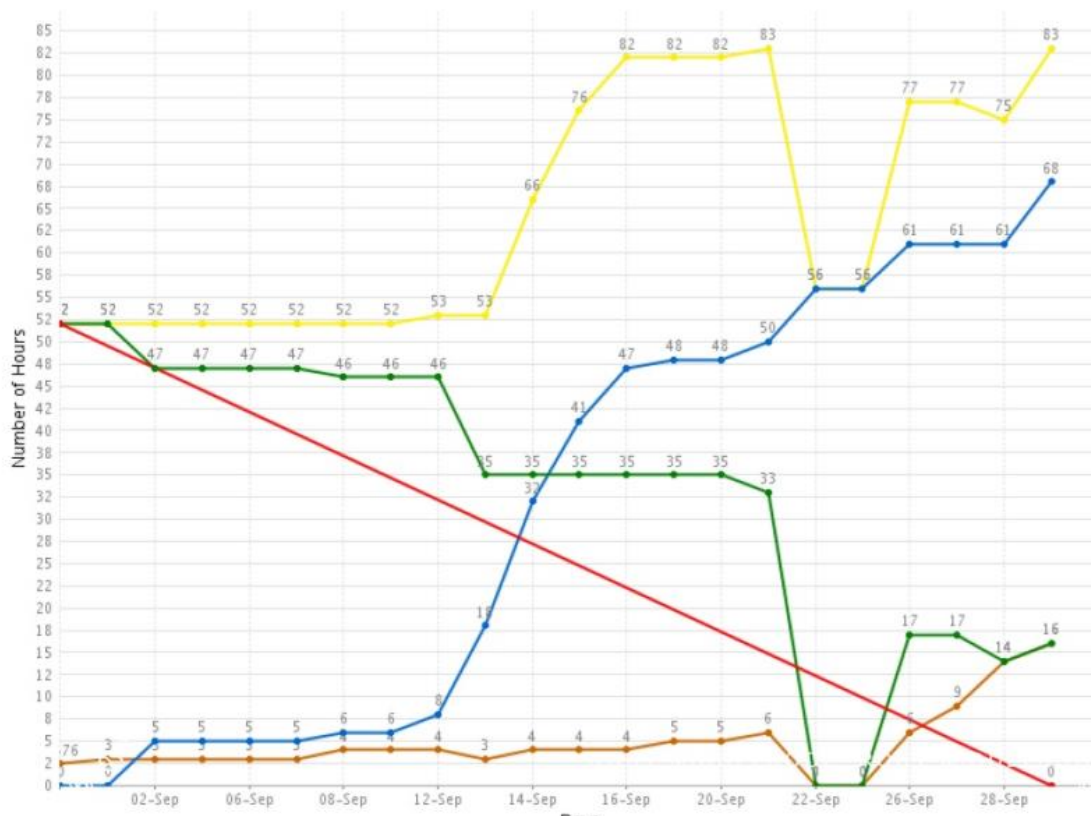
Figuur 7. Proces stappen in Sprint.

Name	Project Lead
Beheer	DDOA Administrator
DDOA Avisi LogAnalyzer	Sjors Palm
DDOA Avisi OpenSpaces	Harm Berntsen
DDOA Example Project	DDOA Administrator
DDOA First8 TennisCMS	Gerben Tijdeman
DDOA ICA GameAPI	Edward Raak
DDOA InfoSupport Endeavour Project Assistant	Gerard Verbeek
DDOA InfoSupport Keytool UI	Maik Diepenbroek
DDOA Logica OV Crowdsourcing	Nati Yonatan Abbai
DDOA Luminis QR Geocaching	Tom Niemantsverdriet
DDOA RID Planning Roostering Systeem	Dennis Bameveld

Figuur 8. Jira Projects.



Figuur 9. Daily Standup (ToDo, InProgress, Done) met Jira Greenhopper.



Figuur 10. Jira voorbeeld: gewerkte uren in periode.

De Daily Stand-Up vindt plaats op een vaste plek, vast tijdstip en duurt maximaal 15 minuten (mag dus korter). De scrummaster is in control met de (standaardvragen): Wat heb je gisteren gedaan? Wanneer is je werk klaar? Wat ga je vandaag doen? Wat blokkeert ('impediment') het succesvol doen van je taak? (figuur 9 en figuur 10).

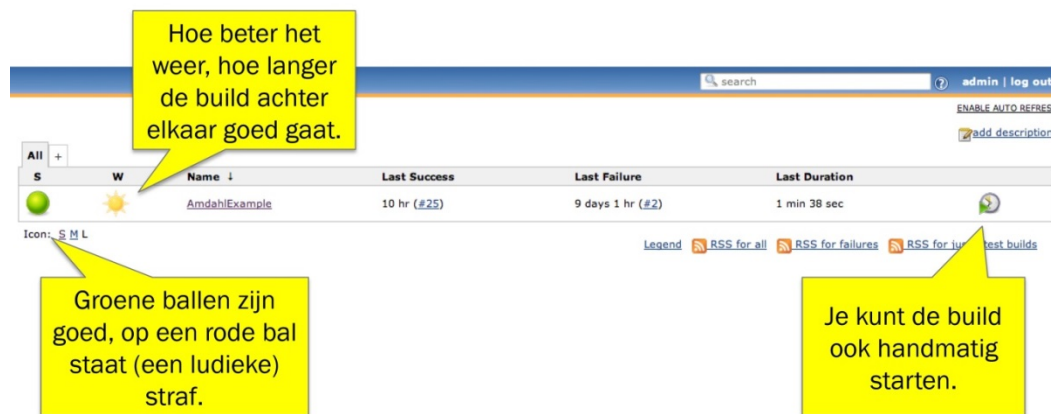
De Sprint Review checkt de sprint-goal met de opdrachtgever/product owner door demonstratie van de features, door een dwarsdoorsnede van je architectuur te maken en de definition-of-done te controleren (figuur 11). De studenten reviewen elkaars werk en vragen op initiatief van de student ook reviews aan bij de docent.

Per Use Case	Alle code compileert en is ingecheckt, voorzien van zinvol commentaar.
	De publieke interface van elke klasse is voorzien van Javadoc.
	Code is geïndenteerd en geformatteerd op uniforme wijze.
	Code voldoet aan door DDOA vastgestelde coderichtlijnen (te checken met Sonar, Checkstyle of PMD).
	Alle unittests kunnen uitgevoerd worden.
	Aantoonbaar 100% Testcoverage voor alle classes die getest kunnen worden.
Per Sprint	Product backlog is up to date.
	Sprint backlog is up to date.
	Use Case beschrijvingen zijn bijgewerkt in het SRS.
	Package-, class-, sequence-, en deployment-diagrammen zijn bijgewerkt in het SAD
	Ontwerpbeslissingen zijn vastgelegd: het gebruik van koppeling, cohesie en/of andere design patterns.
	De burndown eindigt op 0.
	Alle taken (issues, bugs, tasks, etc.) zijn afgesloten of uitgesteld tot de volgende sprint.
	Er is een installatiehandleiding voor de installatie van de software.
	Er is zoveel mogelijk gebruikersdocumentatie beschikbaar in de applicatie; rest digitaal en/of op papier.
	Falende nightly en continuous builds zijn binnen een dag op orde.
	De sprint is geaccepteerd door de productowner/ opdrachtgever(s), de sprint goal is gehaald.

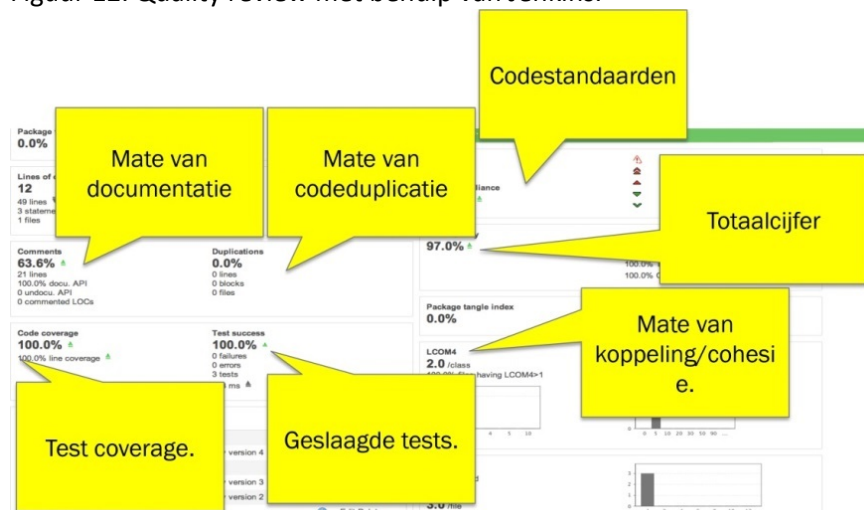
Figuur 11. De 'definitie' van de 'Definition of Done'.

Quality review

De controle op de kwaliteit van het ontwikkelde systeem gebeurt handmatig door het team, handmatig door docenten en/of de opdrachtgever, gevraagd en ongevraagd. Quality review maakt gebruik van automatisch controles en analyses door de tools Jenkins (figuur 12) en Sonar (figuur 13).

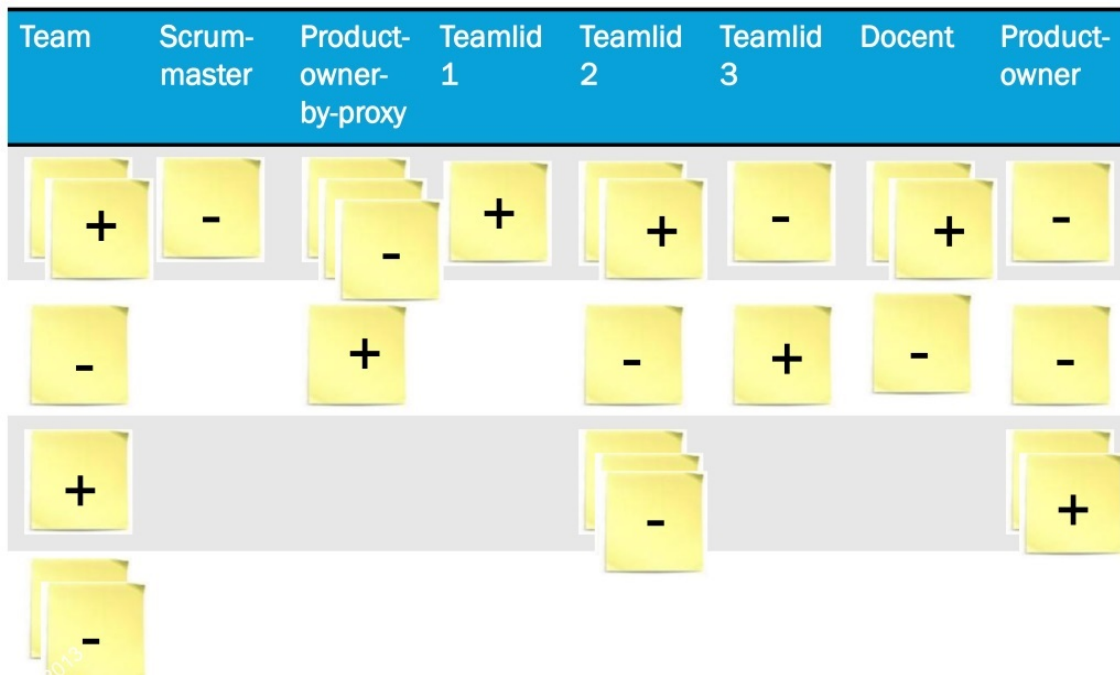


Figuur 12. Quality review met behulp van Jenkins.



Figuur 13. Quality review met behulp van Sonar.

In de Retrospective wordt feedback geleverd door alle betrokkenen op elkaar. Door per teamlid en voor de verschillende rollen de positieve en negatieve bevindingen te noteren ontstaat een overzicht van de mogelijke verbeteringen in het product, proces en project (figuur 14).



Figuur 14. Retrospectieve: feedback van en voor alle betrokkenen in de Sprint.

Toetsing

De toetsing van de studenten vindt individueel plaats, door drie docenten op basis van het portfolio voor de productbeoordeling en een assessment voor de competentiebeoordeling (figuur 15). De producten in het portfolio worden beoordeeld en met cijfers gewaardeerd. De competenties worden in het assessment finaal beoordeeld, mede op basis van de activiteiten die de student tijdens het project heeft verricht. Daarbij moet de student per projectrol voldoen aan verschillende eisen binnen eenzelfde competentie (figuur 16).

Producten	Competenties
- Requirements	- Analyseren
- Functioneel Ontwerp	- Ontwerpen
- Technisch Ontwerp	- Realiseren/Testen
- Software	- Samenwerken
- Retrospectives	- Communiceren
- Requirements/Ontwerpen	- Planmatig werken
- Plan van Aanpak	

Figuur 15. Te toetsen producten en competenties.

Rol/Competentie	Samenwerken	Communiceren	Planmatig Werken
Scrum Master	X		X
Product Owner		X	
Team Member	X	X	X

Figuur 16. Competenties en rollen in DDOA-project.

Evaluatie door studenten

De studenten vinden het lastig om docent en opdrachtgever beiden tevreden te houden (wanneer is het klaar?). Het is motiverend om voor echte opdrachtgevers te werken en een product op te leveren dat bruikbaar is. Ze ervaren een goede begeleiding door docenten en bedrijven. Het agile werken bevalt goed en wordt toegepast in andere onderwijseenheden zoals stage en afstuderen. De aangereikte en deels 'afgedwongen' structuur geeft focus en bevordert actieve deelname (beperkt 'lui' zijn). De tools zijn nuttig maar hebben hoge leercurve.

Nadere informatie over het DDOA semester is te vinden op: <http://wiki.icaprojecten.nl>

Wilt u reageren op deze presentatie? Neem dan contact op met:

Rody Middelkoop; docent; Hogeschool van Arnhem en Nijmegen.

rody.middelkoop@han.nl

Twitter: @rodmidde

W <http://nl.linkedin.com/in/rodym/>

W <http://www.slideshare.net/rodym>