



HOGESCHOOL ROTTERDAM / CMI

Project 2 – Game development

INFPRJ01-2



Aantal studiepunten: 4

Modulebeheerder: Stelian Paraschiv



Modulebeschrijving

Modulenaam:	Project 2 - Game development																					
Modulecode:	INFPRJ01-2																					
Aantal studiepunten en studiebelastinguren:	<p>Dit studieonderdeel levert je vier (4) studiepunten op, hetgeen overeenkomt met een studielast van 112 uren.</p> <p>De aanbevolen verdeling van deze 112 uren over de collegeweken is als volgt:</p> <table><tr><td><u>Begeleide colleges:</u></td><td></td><td></td></tr><tr><td>Kick-off:</td><td>2 * 50 minuten</td><td>2 uur</td></tr><tr><td>Gedurende 8 weken projectles:</td><td>8 * 180 minuten</td><td>24 uur</td></tr><tr><td>Presentatie product:</td><td>1 * 50 minuten</td><td>1 uur</td></tr><tr><td><u>Onbegeleide uren per week:</u></td><td></td><td></td></tr><tr><td>Tijd werken aan project incl. literatuurstudie</td><td></td><td>85 uur</td></tr><tr><td>Totaal</td><td></td><td>112 uur</td></tr></table>	<u>Begeleide colleges:</u>			Kick-off:	2 * 50 minuten	2 uur	Gedurende 8 weken projectles:	8 * 180 minuten	24 uur	Presentatie product:	1 * 50 minuten	1 uur	<u>Onbegeleide uren per week:</u>			Tijd werken aan project incl. literatuurstudie		85 uur	Totaal		112 uur
<u>Begeleide colleges:</u>																						
Kick-off:	2 * 50 minuten	2 uur																				
Gedurende 8 weken projectles:	8 * 180 minuten	24 uur																				
Presentatie product:	1 * 50 minuten	1 uur																				
<u>Onbegeleide uren per week:</u>																						
Tijd werken aan project incl. literatuurstudie		85 uur																				
Totaal		112 uur																				
Vereiste voorkennis:	Voor deze module is geen voorkennis vereist																					
Werkvorm:	Projectonderwijs (groepswork)																					
Toetsing:	Toetsing vindt plaats op basis van opgeleverd beroepsproduct en toetsing van projectproces																					
Leermiddelen:	Franken, M (2013), <i>Scrum voor Dummies</i> . (1e druk), Amsterdam: Pearson Education Benelux B.V., ISBN 978-90-430-2403-7																					
Draagt bij aan competentie:	<p><i>Algemene beroepsvaardigheden, zie bijlage 2 modulewijzer INFSKL01-1:</i></p> <ul style="list-style-type: none">▪ Ondersteunen en samenwerken▪ Analyseren en interpreteren▪ Organiseren en uitvoeren																					
Leerdoelen:	<ul style="list-style-type: none">▪ Je bent je bewust van je sterke en zwakke kanten binnen samenwerking en voelt je verantwoordelijk voor de uitvoering van jouw taken▪ Je bent in staat feedback op adequate wijze te geven en kunt feedback ook adequaat ontvangen▪ Je kunt in een gesprek of in een vergadering kort en zakelijk informatie uitwisselen▪ Je bent in staat om de basiskennis die nodig is voor dit project (deze studie) je eigen te maken en zo nodig aan te vullen met (zelf)studie op gebied van recente ontwikkelingen▪ Je kent de regels en procedures die in een samenwerkingscontract met je projectgroep zijn opgesteld▪ Je kunt een nieuwe epic introduceren bij de PO en deze operationaliseren▪ Je bent in staat user stories te schrijven volgens de (Scrum) richtlijnen▪ Je kunt planningpoker toepassen om tot reële schattingen van taken te komen▪ Je bent in staat een spel te programmeren in een object georiënteerde taal																					
Inhoud:	Je leert in groepsverband werken (proces) en een projectopdracht voor een opdrachtgever realiseren (product). In een projectgroep realiseer je een game die relatie heeft met het Havenbedrijf Rotterdam “Port of Rotterdam”. Aan bod komen allerlei aspecten van het ontwikkel-/projectmethodiek Scrum.																					
Opmerkingen:	Aanwezigheid is verplicht. Groepen worden door studenten zelf samengesteld en tijdens de eerste les bij de tutor bekend gemaakt. 4 à 5 studenten per groep.																					
Modulebeheerder:	Stelian Paraschiv																					
Datum:	17 november 2014																					



1. Algemene omschrijving

1.1 Inleiding

Wie heeft er nu nog nooit een spel gespeeld? Bij het onderwerp van de projectopdracht kun je goed gebruik maken van je ervaring als gamer. Misschien heb je dit altijd al willen doen, of ben je gewoon nieuwsgierig naar hoe dit in de praktijk gedaan wordt. Het grote verschil is, dat je nu zelf het spel maakt!

Bij de hbo-opleiding Informatica ga je vanaf dag één leren hoe je op een professionele manier een project opzet en uitvoert om een gezamenlijk beroepsproduct op te leveren. Daarnaast bieden we je de benodigde theorie en skills die daarvoor vereist zijn, flankerend aan. We geven je steeds feedback over hetgeen je doet, maakt en laat zien en daarvan zal je juist weer veel leren.

In deze module leer je het maken van een beroepsproduct in de vorm van een game door gebruik te maken van een object-georiënteerd (OO) programmeren in de taal Java. De casus van deze onderwijsperiode staat centraal bij het uitvoeren van dit project.

Het beheersen van OO programmeren is een van de belangrijkste vaardigheden die je tijdens je studie opdoet. Het overgrote gedeelte van de software die vandaag de dag beschikbaar is, wordt op deze manier gemaakt. Van bijvoorbeeld een simpele website, tot de *apps* die je op je telefoon hebt of de interactieve televisie thuis. Het aanleren van het object-georiënteerde gedachtepatroon, geeft je dus de mogelijkheden om vrijwel alle denkbare software te kunnen schrijven.

1.2 Relatie met andere onderwijseenheden

De skills in een project zijn onlosmakelijk verbonden met het project zelf. Denk daarbij aan *samenwerken, leiding geven, overleggen, elkaar feedback geven en mondeling en schriftelijk communiceren*. De flankerende modules die je naast het project krijgt, hebben allemaal te maken met de kennis en vaardigheden die je in het project nodig hebt. Het onderwerp van het project is daarbij leidend. Bovendien trainen we je om een professionele beroepshouding te laten zien.

Wat je tijdens de flankerende modules leert, pas je direct toe tijdens dit project. Hier kun je zien of je voldoende achtergrondkennis hebt of dat je je extra moet voorbereiden om deze module te kunnen volgen. De flankerende modules zijn:

- **INFANL01-2:** Analyse 2 is de module die je helpt bij het vertalen van problemen uit de reële wereld naar programmeerbare software. Deze module analyse is dus nauw verbonden aan dit project. Je leert hier de werkende code te documenteren in de analysemodule
- **INFDEV01-2:** Development 2 is de module waar je oefent met de techniek voor het uitvoeren van dit project. In dit vak is het niet de bedoeling dat je alles voorgekauwd krijgt, maar dat je het gereedschap aangereikt krijgt om je project zelf aan te pakken
- **INFSKL01-1 (OP2):** Skills is een module die het gehele jaar naast de projecten loopt en waarin je dit kwartaal kennis maakt met vergaderen, notuleren, planningpoker, retrospective, user stories en the daily Scrum

1.3 Leermiddelen

Verplicht:

- Franken, M. (2013), *Scrum voor Dummies* (1^e druk), Amsterdam: Pearson Education Benelux B.V., ISBN 978-90-430-2403-7
- Trello: <https://trello.com> (project Scrum)
- Box.com: <https://box.com> (project dossier)

Facultatieve bronnen:

- <http://www.stackoverflow.com>
- <http://ninjamock.com>
- <http://www.portofrotterdam.com>



- Kölling, M. (2009). *Introduction to Programming with Greenfoot: Object-Oriented Programming in Java with Games and Simulations*, Amsterdam: Pearson Education Benelux B.V., ISBN 978-01-360-3753-8
- Greenfoot download: <http://www.greenfoot.org> en introductie: <http://www.greenfoot.org/doc>

2. Programma en inhoud

2.1 Case

De haven van Rotterdam is een fantastische plek om te werken en biedt jongeren een goed toekomstperspectief. Er zijn ook kansen voor mensen die een baan zoeken of willen veranderen van werk. De komende jaren komen er ondanks de economische situatie flink wat banen bij. Bovendien is voldoende aanbod van goed gekwalificeerd personeel een cruciale succesfactor voor de haven.

Jaarlijks zijn er zo'n 2.200 vacatures in de haven. Dit komt doordat mensen een baan vinden buiten de haven, met pensioen gaan of doordat bedrijven uitbreiden. Wanneer Maasvlakte 2 in 2015 echt in gebruik is, komen daar tot het jaar 2030 in totaal nog zo'n 6.000 banen bij. Voor deze banen zoeken de bedrijven mensen met een technische, maritieme of logistieke opleiding.

In de haven is werk voor iedereen. Bedrijven zijn vooral op zoek naar: kraandrijvers, expediteurs, IT-experts, logistiek managers, scheepsplanners, carrierchauffeurs, vorkheftruckchauffeurs, loodsen, sleepbootkapiteins en technisch specialisten zoals: loodgieters, elektriciens, onderhoudsmonteurs, procesoperators of lassers. Er zijn ook banen voor financieel controllers, marketingadviseurs, salesmanagers, personeels-adviseurs, accountants, koks en secretaresses.

In de ideale haven van 2030 sluiten vacatures van bedrijven en het aanbod van personeel optimaal op elkaar aan. Mensen worden beroepsgericht (technisch, maritiem, logistiek) opgeleid. Jongeren willen graag in de haven werken onder meer omdat werkgevers moderne arbeidsvoorwaarden bieden. Daarin staan flexibiliteit (het nieuwe werken, uitwisseling personeel tussen bedrijven) centraal. Het Havenbedrijf Rotterdam "[Port of Rotterdam](#)" wendt zijn invloed, middelen en kennis aan om deze ideale haven-arbeidsmarkt te realiseren. Natuurlijk zorgen bedrijven zelf voor het werven en selecteren van de juiste mensen. En zorgen scholen voor de juiste opleidingen. Maar omdat het voor het succes van de haven van levensbelang is, heeft het Havenbedrijf Rotterdam samen met andere partijen een programma opgezet om jongeren te interesseren voor de haven. Zoals havenlessen voor alle niveaus en leeftijden, om de aanwas van technische studenten te vergroten. En door overleg te organiseren tussen werkgevers, werknemers, overheden en onderwijsinstellingen probeert het Havenbedrijf knelpunten op te lossen en kansen te benutten op de havenarbeidsmarkt (korte en lange termijn).

Het Havenbedrijf Rotterdam vraagt jullie om jongeren te enthousiasmeren, door ze deze banen in de vorm van een spel aan te bieden. De basis van het spel is een grote kaart van Rotterdam. Op diverse plekken dient werkgelegenheid zich aan. De banen zijn als het ware kleine mini-games in een groter geheel. Je spel bevat vier verschillende banen, waarvoor wij de eisen voor twee mini-games hebben beschreven (zie bijlage 3). Je gaat zelf een voorstel indienen voor twee of drie zelf bedachte mini-games met je projectgroep en deze uitwerken.

2.2 Opdracht

Maak een digitale *serious game* aan de hand van de casus, waarin je jongeren stimuleert om de haven als werkveld te zien door ze kennis te laten maken met de verschillende beroepen van de haven. Dit moet ontwikkeld worden in de interactieve Java- ontwikkelomgeving Greenfoot. De game moet vóór het oplevermoment online komen te staan. Maak hiervoor ook een korte trailer ter promotie van je spel (maximaal een paar minuten).

In bijlage 2 staat de visie op het product verder beschreven.



2.3 Mijlpaalproducten

In de onderstaande tabel staan de op te leveren mijlpaalproducten per week. Hiervoor geldt: lever ze uiterlijk 24 uur vóór aanvang van de betreffende les in. Het inleveren gebeurt middels de systemen Trello (project Scrum) en Box.com (project dossier). Eindpresentatie (incl. trailer) +/- 15 minuten.

Week	Sprint	Activiteit	Mijlpaalproduct PO (cijfer)	Mijlpaalproduct Tutor (inleveren ter afvinken)	Individuele bijdrage Tutor (cijfer)
1	1	Kick-off			
2		Sprint review	<ul style="list-style-type: none"> - Product backlog - User stories mini-game 1 en 2 - Epics mini-game 3 en 4 - Sprint backlog (voorstel Sprint 2) 	<ul style="list-style-type: none"> - Samenwerkingscontract - Vergaderlijst met roulerende voorzitter en notulist - Planningpoker - Scrum board 	<ul style="list-style-type: none"> - Ieder projectlid heeft een idee ingebracht voor mini-games (eigen schetsen laten zien) - Ieder projectlid heeft individueel leerdoelen geformuleerd voor dit project
3	2	Toelichting verwachtingen sprint, afstemming met PO en Tutor			
4		Sprint review	<ul style="list-style-type: none"> - User stories mini-game 3 en 4 - Sprint backlog (voorstel Sprint 3) - Shippable product 	<ul style="list-style-type: none"> - Agenda en notulen van vergadering - Retrospective - Planningpoker - Scrum board - Burndown chart sprint 2 	<ul style="list-style-type: none"> - Ieder projectlid kan zelf aangeven wat hij/zij aan zelfstudie voor dit project heeft gedaan met welk resultaat
5	3	Toelichting verwachtingen sprint, afstemming met PO en Tutor			
6		Sprint review	<ul style="list-style-type: none"> - Sprint backlog (voorstel Sprint 4) - Shippable product 	<ul style="list-style-type: none"> - Agenda en notulen van vergadering - Retrospective - Planningpoker - Scrum board - Burndown chart sprint 3 	<ul style="list-style-type: none"> - Ieder projectlid toont aan welk concreet gedrag/actie voortvloeit uit afspraken eerdere retrospective (uit sprint 2)
7	4	Toelichting verwachtingen sprint, afstemming met PO en Tutor			
8		Opleveren gehele project	<ul style="list-style-type: none"> - Product backlog - Eindproduct (code* & URL) - Eindpresentatie (incl. trailer) 	<ul style="list-style-type: none"> - Agenda en notulen van vergadering - Retrospective - Planningpoker - Scrum board - Burndown chart sprint 4 	<ul style="list-style-type: none"> - Ieder projectlid kan reflecteren op individuele leerdoelen (uit sprint 1)
9		Inloop voor feedback beoordeling			

* Telt mee voor eindcijfer, zie criteria bijlage 3

Scrum planning richtlijnen

In project 2 ga je verder met het toepassen van Scrum. Na project 1 heb je een basis gelegd en nu ga je verder met het opdoen van ervaring. Bij sprint review 1 laat het team alle user stories van de eerste twee mini-games en het menu zien aan de product owner. Daarnaast werkt het team twee zelf verzonden epics uit (bijvoorbeeld in storyboards) voor mini-game drie en vier. Bij sprint review 2 laat het team de user stories zien voor de twee zelf bedachte mini-games. Elke sprint backlog wordt naar eigen inzicht samengesteld en wordt definitief gemaakt na afstemming en overeenstemming met PO.



3. Toetsing en beoordeling

3.1 Voorwaarden

Om de punten voor het team per sprint te ontvangen, moeten de onderstaande mijlpaalproducten voldoende zijn. Ook moet worden voldaan aan de aanwezigheidsplicht en moeten de (tussen)producten op tijd zijn ingeleverd.

Sprint	Onderdelen afgevinkt door tutor	Doel
1	Samenwerkingscontract, vergaderlijst met roulerende voorzitter en notulist, planningpoker en Scrum board	Afvinken (moet voldoende zijn)
2	Agenda en notulen van vergadering, retrospective, planningpoker, Scrum board en burndown chart sprint 2	Afvinken (moet voldoende zijn)
3	Agenda en notulen van vergadering, retrospective, planningpoker, Scrum board en burndown chart sprint 3	Afvinken (moet voldoende zijn)
4	Agenda en notulen van vergadering, retrospective, planningpoker, Scrum board, burndown chart sprint 4 en eindpresentatie	Afvinken (moet voldoende zijn)
Elke sprint	Aanwezigheid (mede afgevinkt door PO)	Afvinken (moet voldoende zijn)
Elke sprint	Op tijd inleveren producten (mede afgevinkt door PO)	Afvinken (moet op tijd zijn)

3.2 Eindcijfer

Het project wordt beoordeeld op een combinatie van groepsprestatie en individuele inzet. Het individuele cijfer telt voor 60% en het teamcijfer voor 40% mee.

Voor elke sprint zijn punten te behalen. De toekenning van deze punten wordt tijdens de reviews bepaald, waarbij de tutor de individuele bijdrage beoordeeld en de product owner het groepsresultaat. Hieronder staat de verdeling van de punten per sprint en de bijbehorende onderdelen.

Sprint	Onderdelen team (40%), beoordeeld door product owner	Aantal punten
1	Product backlog, User stories mini-game 1 en 2, Sprint backlog en Epics mini-game 3 en 4	0 of 1
2	User stories mini-game 3 en 4, Sprint backlog en shippable product	0 of 2
3	Sprint backlog en shippable product	0 of 2
4	Eindproduct en eindpresentatie (mede beoordeeld door tutor)	0 of 2

Sprint	Onderdelen team (40%), beoordeeld door development docent	Aantal punten
4	Eindproduct code*	0, 1, 2 of 3

* Wordt beoordeeld door development docent, zie criteria bijlage 3

Sprint	Onderdelen individueel (60%), beoordeeld door tutor	Aantal punten
1	Idee voor mini-games en individueel leerdoelen	0 of 2
2	Zelfstudie voor dit project en hier het resultaat van	0 of 2
3	Concreet gedrag/actie voortvloeiend uit afspraken eerdere retrospective	0 of 3
4	Reflecteren op individuele leerdoelen	0 of 2

3.3 Formule

Om het eindcijfer te kunnen bepalen wordt de volgende formule gebruikt:

$$\text{eindcijfer} = \frac{(4 \times \text{punten team}) + (6 \times (\text{punten individueel} + 1))}{10}$$



Voorbeeld *voldoende* (geslaagd):

$$eindcijfer = \frac{(4 \times 8) + (6 \times (3 + 1))}{10} = \frac{56}{10} = 5,6$$

Voorbeeld *onvoldoende* (herkansen):

$$eindcijfer = \frac{(4 \times 9) + (6 \times (2 + 1))}{10} = \frac{54}{10} = 5,4$$

3.4 Herkansing

Bij een onvoldoende voor INFPRJ01-2 dient het project in zijn geheel te worden herkanst in de volgende onderwijsperiode. Dit houdt in dat er een vergelijkbare opdracht opnieuw in een samengestelde projectgroep gemaakt moet worden (OP-beheerder deelt in), zonder begeleiding en zonder ondersteunend onderwijs. Alleen eindproduct(en) worden beoordeeld.



Bijlage 1: Toetsmatrijs

De toetsmatrijs geeft de relatie weer tussen de de leerdoelen en de opdrachten en criteria van een toets.

	Leerdoelen	Verwijzing naar producten / criteria
1.	Je bent je bewust van je sterke en zwakke kanten binnen samenwerking en voelt je verantwoordelijk voor de uitvoering van jouw taken	Samenwerkingscontract, retrospective en Scrum board
2.	Je bent in staat feedback op adequate wijze te geven en kunt feedback ook adequaat ontvangen	Retrospective en planningpoker
3.	Je kunt in een gesprek of in een vergadering kort en zakelijk informatie uitwisselen	Retrospectie, sprint review en planningpoker
4.	Je bent in staat om de basiskennis die nodig is voor dit project (deze studie) je eigen te maken en zo nodig aan te vullen met (zelf)studie op gebied van recente ontwikkelingen	Shippable product en eindproduct (code)
5.	Je kent de regels en procedures die in een samenwerkingscontract met je projectgroep zijn opgesteld	Samenwerkingscontract
6.	Je kunt een nieuwe epic introduceren bij de PO en deze operationaliseren	Epics mini-games (3 en 4)
7.	Je bent in staat user stories te schrijven volgens de (Scrum) richtlijnen	Product backlog, sprint backlog en user stories mini-games
8.	Je kunt planningpoker toepassen om tot reële schattingen van taken te komen	Planningpoker en burndown chart
9.	Je bent in staat een spel te programmeren in een object georiënteerde taal	Shippable product en eindproduct (code)



Bijlage 2: Visie op het product

Het doel van het spel is het stimuleren van jongeren om de haven als werkveld te zien door ze kennis te laten maken met de beroepen van de haven. Deze visie wordt gewaarborgd door de product owner (PO). Daarnaast zijn de twee epics hieronder leidraad voor het sturen van de inhoud van een groot gedeelte van het spel. De doelgroep van het spel zijn jongeren/jong volwassenen (14-23 jaar).

Epics (of features)

Voor het spel zijn een aantal onderdelen gegeven. Daarnaast mag je zelf twee *epics* (mini-games) introduceren. Eventueel mag je nog een vijfde epic in overleg met je PO bespreken. De epics van de opdrachtgever zijn hieronder beschreven. Ga zelf op zoek naar bronnen om jezelf in te lezen in de werkzaamheden van de beroepen...

Epic: menu

De basis van het spel is een high-level view waar de haven gevisualiseerd wordt. Dit vormt het startpunt voor de mini-games. Ondanks het mini-game concept is het geheel wel een geïntegreerd spel. Alle mini-games zijn speelbaar, d.w.z. het spel is pas 'uitgespeeld' als alle mini-games succesvol zijn doorgelopen. Het spelverloop kan op twee manieren uitgewerkt worden, de volgorde waarin je de mini-games speelt kan willekeurig zijn of als verhaallijn waarbij de mini-games de loop van het spel beïnvloeden.

Epic: mini-game controlecentrum

In deze mini-game neemt de speler de rol aan van coördinator voor de instroom en doorstroom van schepen in de haven. De speler ziet de monding van de haven. Nieuwe schepen komen de haven binnenvaren. Aan schepen wordt een loods toegekend die de schepen begeleid naar een ligplaats. Nadat schepen gelost hebben en/of geladen zijn kunnen ze de haven uitgeloozd worden. De nieuwe schepen komen steeds iets sneller aan bij de haven. De uitdaging voor de speler is het onder toenemende druk coördineren van de scheepvaart.

Epic: mini-game vracht overslaan

In deze mini-game maakt de speler kennis met de beroepen binnen vracht overslag (bijvoorbeeld kraanmachinist). De aanvoer van vracht wordt gedaan door schepen, de afvoer van vracht wordt gedaan door transportvoertuigen zoals trein of vrachtwagen. De containers op een boot hebben echter niet altijd de optimale volgorde hiervoor. De (bestemming van) transportvoertuig bepaalt de aanvraag, dus wat er van de schepen wordt gehaald. Bij het verplaatsen van containers moet rekening worden gehouden met de balans van het schip. Containers kunnen op het schip ook niet eindeloos opgestapeld worden (maximale hoogte). Containers hebben ook verschillende afmetingen. Er is een computertegenstander die tegelijkertijd ook de vracht van een schip wegwerkt, je doel is om de computertegenstander bij te houden. Wie als eerste het schip leeg krijgt, heeft gewonnen. De uitdaging zit in het speelbaar houden van het spel door de moeilijkheid (snelheid/complexiteit) te beïnvloeden. Zo krijgt de speler een competitief gevoel om van de computertegenstander te winnen.

Tip! De speelbaarheid van het spel wordt bepaald door de speelbalans. Test dit met gebruikers!

Epic: mini-game 3,4

Door de groep zelf in te vullen. Je werkt twee epics uit in user stories. Deze worden ter goedkeuring aangeboden in het tweede overleg met de PO. De PO kan hierbij aanwijzingen geven. Ga zelf op zoek naar documentatie over de casus en gebruik dit als inspiratie voor je nieuwe mini-games.



Bijlage 3: Beoordelen van code in het project (door development docent)

Beoordeling worden per onderdeel (kolommen) opbouwend beoordeeld (3 impliceert ook 2 en 1). Alleen 'eigen code' wordt beoordeeld dat binnen het Greenfoot project(scenario) valt. Kopiëren van code van derde is toegestaan, maar geef in commentaar aan waar deze code begint en eindigt. Er worden vier minigames gecontroleerd op onderstaande criteria (controle of code 'op niveau' is). Deze moeten allen voldoen.

	Code algemeen	OO – klassenniveau	OO – Functieniveau	OO - toepassing
1	Commentaar per functie en class Correct gebruik van controle structuren Correct leesbaar (o.a. goed ingesprongen, 1 statement per regel)	Begrijpelijke benaming Encapsulation toegepast Modifiers uniform gebruikt op variabelen	Begrijpelijke benaming Overloading toegepast Non-static functies toegepast Zowel void als non-void functies toegepast (naast die bij encapsulation)	Constructor gebruikt voor instantiatie objecten
2	Urenverdeling dev via readme bestand (wie/wat en hoeveel uur) Correct gebruik van loops Juiste datatypen gebruikt	Abstracte objecten waar mogelijk Geen overbodig gebruik van de modifier public	Method overriding gebruikt Static functies toegepast waar toepasselijk	Ook constructor gebruikt met argumenten Static variabelen toegepast
3	Alle code wordt aangeroepen (geen 'dode' code) Zeer weinig redundante (duplicate) code Commentaar per functie en class conform javadoc	Interfaces gebruikt waar dat kan	Public functies beperkt gebruikt (bewust 'exposure', dus niet <i>standaard</i> public gebruikt)	

0 = onvoldoende of niet aanwezig of te laat ingeleverd