

# Nakijkmodel PO Object georiënteerd Python – Tekst RPG

Totaal punten mogelijk: 88 (voor 2-tallen: 100)	Beoordeling (punten/totaal*10) ...
----------------------------------------------------	------------------------------------

Opdracht	Kwaliteit van uitwerking met puntentelling	Weging dit onderdeel
----------	--------------------------------------------	----------------------

Code layout en verslaglegging	0	1	2	3	4	
1. Code layout, consistentie en commentaar	Code is rommelig. Variabelen hebben onlogische namen. Inspringen gebeurt niet consistent. Geen verduidelijkend commentaar	Er is enige orde en consistentie, maar deze laat nog behoorlijk te wensen over.	Code is leesbaarder gemaakt met commentaar en/of logische naamgevingen. Het kost een buitenstaander nog redelijke moeite om de code goed te begrijpen	Code is over het algemeen netjes. Soms zou extra commentaar wenselijk zijn of is de naamgeving nog wat onduidelijk/onlogisch, maar de code is goed te gebruiken door een buitenstaander.	Code layout is netjes en consistent. Variabelen hebben logische namen. De code is voorzien van genoeg commentaar waar dat nodig is. Iemand anders kan zo met de code aan de slag.	2
2. Verslaglegging – Toelichting op de uitbreidingen / wijzigingen	Ontbreekt	Minimale beschrijving van de wijzigingen  Slechte layout	Op hoofdlijnen is beschreven wat de toevoegingen en wijzigingen zijn.  OF Layout laat te wensen over	Bovendien is het effect van de wijzigingen op de gameplay besproken OF Op hoofdlijnen is besproken hoe deze wijzigingen technisch zijn gerealiseerd in de code	←, maar nu “en”  Tevens nette layout	2
3. Verslaglegging – Samenwerking <i>(alleen voor 2-tallen)</i>	Ontbreekt	Er is genoemd wat ieders aandeel is geweest, zonder verdere toelichting	Er is een beknopte beschrijving van ieders aanpak en aandeel	Uitgebreide beschrijving van de aanpak en ieders bijdrage. Uitleg over welke strategie er is gebruikt om efficiënt samen te werken en de code samen te voegen. OO principes zijn hier genoemd	← tevens is hier expliciet gebruik gemaakt van OO principes en dit inzichtelijk gemaakt. Bijvoorbeeld door beschrijvingen van classes en hun functies vooraf op te stellen en bij te voegen.	3

Implementatie en uitbreidingen	0	1	2	3	4	
4. Basiscode uit filmpjes Wt ( <i>beoordeling per groepslid individueel</i> )  <a href="#">Dit zijn data van de tussendeadlines</a>	Filmpjes niet of veel te laat gemaakt (na laatste tussendeadline)	Alle filmpjes gemaakt en gedemonstreerd dat je het zelf hebt gedaan voor tussendeadline 4	Alle filmpjes gemaakt en gedemonstreerd dat je het zelf hebt gedaan voor tussendeadline 3	Alle filmpjes gemaakt en gedemonstreerd dat je het zelf hebt gedaan voor tussendeadline 2	Alle filmpjes gemaakt en gedemonstreerd dat je het zelf hebt gedaan voor tussendeadline 1	3
5. Object Oriëntatie principes toegepast bij de uitbreidingen	Geen OO principes toegepast	Enkele basale OO principes die rechtstreeks doorwerken op de OO onderdelen uit de filmpjes	Tevens enkele echte eigen OO toevoegingen zoals nieuwe (sub)classes en/of functies	Substantiële OO principes gebruikt, maar niet alles of niet overal zinvol.	Het volledige spectrum aan OO principes goed en <u>zinvol</u> toegepast. Nieuwe (sub)classes, functies, default-values, overriding, etc.	3
6. Juiste keuzes gemaakt bij implementatie van de eigen toevoegingen	Geen structuur. Onhandige / onlogische keuzes	Veel van de geïmplementeerde toevoegingen hadden logischer op een andere manier gemaakt kunnen worden. Denk aan subclasses of juist niet, aparte functies voor aparte acties / functionaliteit, etc.	Over het algemeen zinnige keuzes gemaakt voor de indeling van de code. Maar op een aantal vlakken zijn duidelijke verbeteringen aan te wijzen	Op de meeste plekken is voor de juiste constructies gekozen om de gekozen functionaliteit te implementeren. Zinnige (sub)classes, logische verdeling van functionaliteiten over functies. Logische constructors, etc.	De eigen uitbreidingen zijn op alle vlakken op de juiste manier geïmplementeerd. Er is voor de beste manier gekozen om elke nieuwe functionaliteit in het geheel op te nemen	3
7. Complexiteit van de uitbreidingen	(Bijna) niets gedaan	Enkele eenvoudige uitbreidingen	Meerdere uitbreidingen, van redelijk niveau.	Een aantal substantiële uitbreidingen die de game naar een hoger plan tillen.	Zeer complexe uitbreidingen die hele nieuwe dimensies aan de basisgame toevoegen. VWO+ niveau	8
8. Creativiteit / Originaliteit	Geen	Eigen dingen, maar volledig in de lijn van verwachting	Een eigen stempel op de game, maar nog erg herkenbaar en vaker gedaan.	Originele ideeën, consistent doorgevoerd en echt een eigen "smoel" aan de game gegeven	Zeer hoge mate van originaliteit en creativiteit.	1