

## Quality-checks opdracht Software- en Gameproject

- De opdracht heeft een **innovatief** karakter / maatschappelijke relevantie.
- Opdrachtgever omschrijft de **functionele opzet**; de opdrachtgever legt de gewenste functionaliteit vóór aanvang op hoofdlijnen vast.
- Er is ook een product-owner met **vereiste domein kennis bij de opdrachtgever**.
- Er is bij voorkeur ook ruimte voor **creativiteit en inbreng** van de studenten.
- Er moet een **duidelijk scope** van de opdracht zijn én het is primair **een software development project**.
- Het project omvat een **op zichzelf te ontwikkelen** en vastomlijnde module. Het is prima als de module moet aansluiten op andere modules, maar het format van de interface moet vastliggen.
- De opdracht omvat een traject bestaande uit **software ontwerp, bouw en testen**.
- Opdrachtgever kiest één target platform en/of een platform onafhankelijk omgeving.
- Indien een project voortbouwt op een bestaande software base, moet er **vóór aanvang** van het project een stable release zijn met bijbehorende release notes, known issues; een geïnstalleerd en draaiende omgeving. Major refactoring in bestaande software elders ontwikkeld is in principe niet toegestaan.
- De opdrachtgever omschrijft het **projectplan** ruim voor het begin van het project in de template ontwikkeld door het Projectbureau.
- Studenten werken als **zelfstandig** ontwikkelteam of game studio en doen zelf de projectorganisatie. Het is **geen stage** noch zijn ze onderdeel van uw ontwikkelteam.
- Het is een software project met nadruk op engineering en discipline. Er moet worden gewerkt volgens **SCRUM (én niet Scrum in name only: SINO)**. Na elke sprint (twee weken) een demo aan de opdrachtgever. De opdrachtgever geeft hierbij feedback en er is een discussie over prioriteiten.
- Studenten maken **technische keuzes** binnen de randvoorwaarden van de opdrachtgever en zijn verantwoordelijk voor software ontwerp, bouw en testen.
- Studenten zijn verantwoordelijk voor planning en versiebeheer, de opdrachtgever heeft geen directe toegang tot hun backlog of versiebeheer systeem.
- UU regelt werkplekken en een standaard ontwikkelomgevingen, inclusief een server en versiebeheer. Opdracht specifiek benodigheden dienen door opdrachtgever te worden geleverd; b.v. specifieke software assets van de Unity asset-store.
- Een **begeleider** van de UU coacht en beoordeelt de studenten op resultaat, proces, rapportage, communicatievaardigheden en technisch skills. **NB:** De begeleider doet geen uitvoerende werkzaamheden in het project.
- De opdrachtgever woont de tussen- en eindpresentaties bij, heeft regelmatig contact met de begeleider over de voortgang en vult aan het eind een evaluatie in. De evaluatie van de opdrachtgever is kwalitatief, de begeleider en het Projectbureau bepalen de cijfers van de studenten.
- Studenten werken op locatie op de UU in kantoor tijden; teamgrootte 9-11 studenten, **15ECTS** per student.
- Voor het aanleren van een **academische werkwijze** volgen de studenten ook hoorcolleges, houden ze presentaties en schrijven een verslag, waarin veel aandacht is voor het onderbouwen van gemaakte keuzes.
- Opdrachten zijn op best-effort basis (zie contract).
- In de laatste twee sprints wordt er geen (grote) nieuwe functionaliteit gebouwd maar bugs opgelost en eindrapportages gemaakt.