

## 1 Inleiding

Het bedrijf Educom wil een leerplatform Odinido ontwikkelen waarmee docenten kennistoetsen kunnen aanmaken en die laten uitvoeren door studenten. Het platform dient beschikbaar te komen op de meest gangbare devices.

Educom vraagt aan ICA-studenten om een analyse en ontwerp te maken voor Odinido. Hieronder wordt de volledige functionaliteit van Odinido beschreven.

Er zijn een paar randvoorwaarden aan deze opdracht verbonden: je moet in groepjes van twee studenten een objectgeoriënteerde analyse en ontwerp in UML maken volgens de richtlijnen gegeven in Larman, en je moet dit ontwerp richten op een implementatie (van het serverside deel) in Java. Tevens bouw je (een deel van) een Java-applicatie voor het beantwoorden van de vragen van een kennistoets.

## 2 Beschrijving van het leerplatform Odinido

### ***Kennistoets aanmaken***

Een docent kan een kennistoets aanmaken die bestaat uit meerkeuzevragen, juist/onjuist-vragen en kort-antwoord-vragen. In de bijlage vind je een voorbeeld van een kennistoets die uit 5 vragen bestaat waarvan vier meerkeuzevragen en een kort-antwoord-vraag (vraag 3). Bij een kort-antwoord-vraag kan worden vastgelegd welke antwoorden correct zijn.

### ***Registreren***

Voordat een docent een kennistoets kan aanmaken dient hij zich te registreren bij Odinido door een e-mailadres (dat wordt als gebruikersnaam gehanteerd), voornaam en achternaam en wachtwoord op te geven. Verder dient de docent aan te geven bij welke school/organisatie hij werkzaam is.

Er zijn twee typen accounts: Basis en Premium. Bij een Basis account krijgt een docent een lokaal met een lokaalnummer toegewezen waarin hij een kennistoets kan uitvoeren. Bij een Premium account kan een docent tot acht verschillende lokalen aanmaken en (parallel) gebruiken.

Er kunnen maximaal 50 studenten deelnemen aan een sessie, d.w.z. de uitvoering van een kennistoets, in een lokaal.

Odinido Basis is gratis. Odinido Premium kost €49,99 per jaar. Deze betaling (de eerste keer bij registratie) kan via een online betaalservice of m.b.v. een creditcard. In dat laatste geval zal er bij de verwerking van de betaling gebruik worden gemaakt van een betrouwbare creditcard validator.

Na de registratie heeft de docent natuurlijk nog de mogelijkheid om zijn profielgegevens, en dus ook het accounttype, aan te passen.

### ***Uitvoeren kennistoets***

De docent kiest een kennistoets en start deze in (een van) zijn lokaal(en). Studenten kunnen zich vervolgens aanmelden bij het lokaal. Daarbij dienen ze een studentnaam op te geven.

De student beantwoordt de vragen een voor een. Daarbij moet het mogelijk zijn om op een eenvoudige manier door de vragen te navigeren en terug te gaan naar een eerder bekeken vraag om alsnog een antwoord te geven of het eerder gegeven antwoord aan te passen. Als de student alle vragen heeft beantwoord, dan geeft hij aan dat hij klaar is.

Een student verdient punten per goed beantwoorde vraag. Verder krijgt de student een bonus op basis van de totale tijd die nodig was om de kennistoets te doorlopen. Hoe sneller, hoe meer bonuspunten er verdiend worden. Als de tijd van de kennistoets verstreken is en/of alle studenten klaar zijn met het beantwoorden van de vragen, dan kan de docent m.b.v. Odinido een overzicht, met per deelnemer de (totaal)score, genereren.

### ***Puntentelling***

Hierboven is de puntentelling bij een kennistoets (globaal) beschreven. Hoeveel punten je precies krijgt voor een goed beantwoorde vraag en hoeveel bonuspunten op basis van de tijd moet nog nader gespecificeerd worden. Bij Educom bestaat de wens om de puntentelling flexibel te maken, d.w.z. er moet makkelijk overgestapt kunnen worden op een andere systematiek.

### **Teamgevecht**

Docenten met een Premium account kunnen een teamgevecht starten. Bij een teamgevecht wordt een bestaande kennistoets door teams gespeeld. De docent stelt een tijd in die gebruikt kan worden voor het doorlopen van de kennistoets.

Per team meldt één persoon, de aanvoerder, zich aan. De aangemelde aanvoerders krijgen automatisch een teamnummer toegewezen. Vervolgens kunnen de vragen worden beantwoord. De groep die de meeste vragen goed beantwoordt wint. Bij een gelijk aantal punten geeft de bestede tijd de doorslag. Om de spanning op te voeren wordt de voortgang van elk van de teams op een scorebord getoond.

### **Toekomstplannen**

De eerste versie van Odinido is gericht op de Nederlandse markt. In de toekomst wil Educom ook versies in andere talen (bijv. Engels, Duits, Frans en Spaans) op de markt brengen.

## **3. Opdracht**

Je wordt gevraagd om de analyse en het ontwerp voor het volledige systeem uit te voeren. Voer de onderstaande stappen uit. Beschrijf de resultaten van deze stappen in een SRS (Software Requirements Specification) en een SDD (Software Design Description), inclusief relevante toelichting bij diagrammen en onderbouwing van de belangrijkste keuzes. Templates voor SRS en SDD vind je op OnderwijsOnline.

1. Stel een use case model op dat bestaat uit een use case diagram en de use case beschrijvingen in brief format.
2. Stel een lijst op met de overige functionele eisen en de niet-functionele eisen. Nummer deze en deel ze in volgens FURPS+. Zorg er voor dat je minimaal de hints voor niet-functionele eisen uit de tekst allemaal SMART hebt opgenomen.

### **Iteratie 1: Uitvoeren quiz**

3. Beschrijf de use case(s) die hoort (horen) bij het uitvoeren van een kennistoets in fully-dressed format. Gebruik eventueel activity diagrams om de flow van ingewikkelde use cases weer te geven.
4. Maak een domeinmodel dat in ieder geval alles bevat dat relevant is voor het uitvoeren van de kennistoets.
5. Werk de use case(s) en bijbehorende systeemoperaties uit met behulp van sequence diagrams via system sequence diagrams. Het uitvoeren van de kennisquiz moet minimaal tot de uitgewerkte diagrammen horen.
6. Stel een design class diagram op voor deze iteratie.
7. Realiseer een console-applicatie in Java waarmee een student de vragen van een kennistoets beantwoordt. Voor deze applicatie mag je ervan uitgaan (in tegenstelling tot wat er onder het kopje **Uitvoeren kennistoets** wordt aangegeven) dat de vragen altijd van voor naar achter worden doorlopen.

*Opmerking: De data voor deze applicatie mag (voorlopig) gemoockt worden. Je kunt hiervoor zelf een class maken die een methode bevat die voorgedefinieerde data retourneert.*

### **Iteratie 2: Registreren en aanmaken kennistoets, uitvoeren teamgevecht**

8. Beschrijf de use cases die horen bij het registreren en aanmaken van een kennistoets en het uitvoeren van een teamgevecht in fully-dressed format.  
Gebruik eventueel activity diagrams om de flow van ingewikkelde use cases weer te geven.
9. Breid indien nodig het domeinmodel uit.
10. Werk de use cases, die je bij stap 8 in fully dressed format hebt beschreven, en de bijbehorende systeemoperaties uit met behulp van sequence diagrams via system sequence diagrams (en eventueel de daarbij behorende operation contracts).

11. Stel indien van toepassing een statemachine diagram op voor objecten die in verschillende toestanden kunnen verkeren.
12. Stel het definitieve design class diagram op.
13. Beschrijf aan de hand van een aantal goed gekozen voorbeelden hoe je OO ontwerpprincipes (denk met name aan GRASP en geschikte design patterns) hebt toegepast.

Het eindproduct (SRS, SDD en de code bij stap 7) dient via iSAS te worden ingeleverd. Zie het toetsrooster voor de inleverdeadline.

**Enkele opmerkingen en tips:**

- Mochten er onduidelijkheden zijn over de kennisquiz, neem dan contact op met de opdrachtgever. De docent zal deze rol op zich nemen.
- Vraag tussendoor feedback op de diagrammen en modellen die je opstelt.

## Bijlage: Voorbeeld kennistoets

Toetscode: ODI-23644129

Toetstitel: Requirements

<b>Vraag 1:</b> Welke van de begrippen needs, features en requirements horen bij het solution domain?
<b>Alternatieven:</b>
A. needs, features en requirements
B. needs en requirements
✓ C. features en requirements
D. alleen requirements

<b>Vraag 2:</b> Wat wordt bedoeld met 'elicit needs from stakeholders'?
<b>Alternatieven:</b>
✓ A. Het boven tafel krijgen van de needs van belanghebbenden.
B. Het documenteren van de needs van belanghebbenden.
C. Het valideren van de needs van belanghebbenden.
D. Het vertalen van de needs van belanghebbenden naar software features.

<b>Vraag 3:</b> Welke term wordt gebruikt voor 'the ability to describe and follow the life of a requirement, in both forwards and backwards direction'?
<b>Correcte antwoorden:</b> traceability, tracebility, traceability, tracebility

<b>Vraag 4:</b> In een project wordt gewerkt volgens een iteratieve en incrementele aanpak. Aan het begin van het project heb je de requirements opgesteld. Op welke momenten in het project kunnen de requirements gewijzigd worden?
<b>Alternatieven:</b>
A. Op elk gewenst moment.
✓ B. Na elke iteratie.
C. Niet meer, de requirements liggen vast.

<b>Vraag 5:</b> In welke categorie (volgens FURPS) valt de volgende eis: R 1. Het systeem moet 75% van de webpagina's binnen 2 seconden tonen. Voor de overige pagina's mag het nooit langer dan 5 seconden duren.
<b>Alternatieven:</b>
A. Functionality
B. Usability
C. Reliability
✓ D. Performance
E. Supportability