

Case Study OOAD: Provo

1 Inleiding

Het bedrijf Educom wil een leerplatform Provo 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 AIM-studenten om een analyse en ontwerp te maken voor Provo. Hieronder wordt de volledige functionaliteit van Provo beschreven.

De volgende randvoorwaarden gelden voor deze opdracht: 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 Provo

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 6 vragen bestaat waarvan vier meerkeuzevragen, een kort-antwoord-vraag (vraag 4) en een juist/onjuist-vraag (vraag 6). Bij een kort-antwoord-vraag dient te worden vastgelegd welke antwoorden correct zijn.

Registreren

Voordat een docent een kennistoets kan aanmaken dient hij zich te registreren bij Provo 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.

Provo Basis is gratis. Provo Premium kost €49,99 per jaar. Deze betaling (de eerste keer bij registratie) kan via een betalingsprovider (bijv. Mollie, Buckaroo, ...).

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 loka(a)l(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. Provo 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.

Toekomstplannen

De eerste versie van Provo 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 de overige functionele eisen (d.w.z. aanvullend op de use cases) en de niet-functionele eisen op, genummerd en ingedeeld volgens FURPS+. In de tekst staan een paar hints voor niet-functionele eisen. Deze hints moet je nog wel wat SMARTer maken.
3. Beschrijf de use cases die te maken hebben met het registreren en het starten, maken en afronden van de kennistoets in fully-dressed format.
Gebruik daar waar zinvol activity diagrams om de flow van ingewikkelde use cases weer te geven.
4. Maak een domeinmodel dat alles bevat dat relevant is voor het uitvoeren van de kennistoets.
Ook de antwoorden die een speler geeft op de vragen van de kennistoets dienen in het domeinmodel te worden opgenomen.
5. Werk de belangrijkste systeemoperaties van de use case(s), die je bij stap 3 hebt uitgeschreven in fully dressed format, uit met behulp van sequence diagrams.
Opmerking: Stel voor use cases met meer dan één systeemoperatie eerst een system sequence diagram op en maak vervolgens per systeemoperatie een sequence diagram.
6. Stel een design class diagram op.
7. Pas het Strategy Pattern toe voor het flexibiliseren van de puntentelling. Laat in zowel het design class diagram als ook in het betreffende sequence diagram zien hoe het Strategy Pattern wordt toegepast.
Mogelijk is er nog een ander GoF design pattern dat voor de hand ligt om te gebruiken. Denk daar eens goed over na en werk het, indien van toepassing, uit in het SDD.
8. Beschrijf aan de hand van een aantal goed gekozen voorbeelden, die zijn te zien in het design class diagrams en/of sequence diagrams, hoe je OO ontwerpprincipes (denk met name aan SOLID, GRASP en GoF design patterns) hebt toegepast.
9. 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) gemockt worden. Je kunt hiervoor zelf een class maken die een methode bevat die voorgedefinieerde data retourneert.

Enkele opmerkingen en tips:

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

Inleveren

In week 4 of 5 kun je de eerste versie van het SRS uploaden naar iSAS. Je krijgt dan feedback van de docent in de vorm van een ingevulde rubric met toelichting.

Ook het eindproduct (SRS, SDD en de Java-code) dient via iSAS te worden ingeleverd. Zie het toetsrooster in iSAS voor de inleverdeadline.

Bijlage: Voorbeeldtoets

1. Welke van de begrippen needs, features en requirements horen bij het solution domain?

- ☐ A needs, features en requirements
 - ☐ B needs en requirements
 - ☒ C features en requirements
 - ☐ D alleen requirements
-

2. Wat wordt bedoeld met 'elicit needs from stakeholders'?

- ☒ 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
-

3. Wat leg je vast m.b.v. use cases?

- ☐ A needs
 - ☐ B features
 - ☒ C functional requirements
 - ☐ D non-functional requirements
-

4. Welke term wordt gebruikt voor 'the ability to describe and follow the life of a requirement, in both forwards and backwards direction'?

☒ traceability ☒ traceability ☒ traceability ☒ traceability

5. 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?

- ☐ A Op elk gewenst moment
 - ☒ B Na elke iteratie
 - ☐ C Niet meer, de requirements liggen vast
-

6. De onderstaande eis R1 valt in de categorie Usability.

R1: 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.

☒ False