**GSO3 Studieplanner**

Auteur: Nico Kuijpers

Datum: 30 jan 2017

**Opmerking mbt voorjaar 2017:**

In het voorjaar van 2017 vindt de NEXT week plaats in week 1. In de planning hieronder wordt gesproken over lessen. Voor voorjaar 2017 geldt dat les 1 in week 2 valt, les 2 in week 3, etc.

**Beroepsproducten**

Het eindcijfer voor GSO3 wordt bepaald op basis van 4 beroepsproducten:

1. Domeinmodel inclusief aanpassing URS en sequence diagrammen.
2. Unittesten en Specificeren.
3. AEX Banner.
4. GSO3 Applicatie (Killer App).

**Les 1 (2 x 2 uur)**

Onderwerpen: organisatie GSO3, requirements, use cases, Visual Paradigm.

Werken aan 1e beroepsproduct:

* Bestudeer het URS (zie URS-internetbankieren.docx) en pas dit aan.
* Bestudeer de video-tutorial StudentAdminClassDiagram.mp4
* Stel het klassendiagram op van het domein op in Visual Paradigm.

**Les 2 (2 x 2 uur)**

Onderwerpen: domeinmodel, navigatie, gedrag en compositie

Werken aan 1e beroepsproduct:

* Breid het klassendiagram uit met navigatie, methoden, constructoren en compositie.

**Individueel** werken aan casus Tour de Fance om te **oefenen** met het opstellen van een klassendiagram. Het is van belang dat je het klassendiagram helemaal hebt uitgewerkt voor aanvang van de les in week 3.

Voorbereiding voor de les van week 3:

* Bestudeer de video-tutorial StudentAdminSequenceDiagram.mp4.

**Les 3 (2 x 2 uur)**

Onderwerp: sequence diagram

Werken aan 1e beroepsproduct:

* Bestudeer de video-tutorial BestellingenClassSequenceDiagram.mp4.
* Stel twee sequence diagrammen op in Visual Paradigm, waaronder in ieder geval het overmaken van geld.

**Individueel** werken aan casus Tour de France om te **oefenen** met het opstellen van een een sequence diagram. Het is van belang dat je het sequence diagram helemaal hebt uitgewerkt voordat je de voorbeeld-uitwerking bestudeert.

**Inleveren voor einde van deze week**: URS, Visual Paradigm project en word document met toelichting op het domeinmodel (1e beroepsproduct).

**Les 4 (2 x 2 uur)**

Onderwerpen: interface, specificeren, JUnit4, schrijven van unittesten

Werken aan 2e beroepsproduct:

* Bestudeer de video-tutorial TreasureChestUnittest.mp4
* Bestudeer de powerpoint presentatie GSO3 W04 2 unittesten.pptx
* Installeer NetBeans Plugin JaCoCo (zie S3\_JaCoCoPluginNetBeans.docx)
* Schrijf unittests voor de klassen Time en TimeSpan en verwijder de fouten uit de gegeven klassen Time en TimeSpan.
* Maak een unittest voor de klasse TimeSpan2 en implementeer deze klasse.
* Maak een begin met het specificeren, testen en implementeren van de klassen Contact en Appointment.
* Vraag tussendoor feedback aan de docent.

**Les 5 (2 x 2 uur)**

Werken aan 2e beroepsproduct:

* Werk verder aan het specificeren, testen en implementeren van de klassen Contact en Appointment.
* Vraag de docent om een beoordeling voor dit beroepsproduct.

**Inleveren voor einde van deze week**: NetBeans project met volledige uitwerking van de opdracht unittesten. Let erop dat de code coverage 100% is! (2e beroepsproduct)

**Les 6 (2 x 2 uur)**

Onderwerp: timers

Werken aan 3e beroepsproduct:

* Realiseer de AEX banner stand-alone applicatie op basis van het gegeven klassendiagram in de opdrachtomschrijving.

Beoordeling 2e beroepsproduct:

* Laat de docent zien dat alle unittests slagen en geef een toelichting.

**Les 7 (2 x 2 uur)**

Onderwerp: RMI, Client Pull

Werken aan 3e beroepsproduct:

* Bestudeer de video-tutorial StudentAdminComponentDiagram.mp4
* Bestudeer de powerpoint presentatie GSO3 W07 2 Component.pptx
* Maak een klassendiagram en een componentendiagram voor de gedistribueerde variant van de AEX banner volgens het Client Pull principe.
* Realiseer de gedistribueerde variant van de AEX banner volgens het Client Pull principe.
* Demonstreer de gedistribueerde applicatie aan de docent.

**Les 8 (2 x 2 uur)**

Onderwerpen: RMI vervolg, Server Push

Werken aan 3e beroepsproduct:

* Maak een klassendiagram en een componentendiagram voor de gedistribueerde variant van de AEX banner volgens het Server Push principe.
* Realiseer de gedistribueerde variant van de AEX banner volgens het Server Push principe.
* Demonstreer de gedistribueerde applicatie aan de docent.

**Inleveren voor einde van deze week**: Visual Paradigm project en NetBeans project met de volledige uitwerking van de opdracht AEX Banner (3e beroepsproduct)

**Les 9 (2 x 2 uur)**

Onderwerp: Introductie opdracht GSO3 Killer App. Dit is een individuele opdracht.

User Requirements Specifications (URS) voor GSO3 Killer App:

1. Maak een beschrijving van je killer app en neem deze op in je URS.
2. Stel een URS op aan de hand van het template document URS-template.docx. Voor een voorbeeld, zie URS-internetbankieren.docx. Zie ook de slides van Les 1.
3. Lever het URS in via Canvas en bespreek het document met de docent.

Beoordeling 3e beroepsproduct:

* Geef een demonstratie van de AEX banner applicatie volgens het pull en het push principe en geef een toelichting.

**Inleveren voor einde van deze week**: URS voor GSO3 Killer App.

**Les 10 (2 x 2 uur)**

Onderwerpen: componenten (herhaling), sessies

Oefenopdracht componenten: casus ShooterGame of casus WhiteBoard.

User Requirements Specifications (URS) voor GSO3 Killer App:

1. Je zult via Canvas het verzoek ontvangen om twee peer-beoordelingen uit te voeren. Voer deze beoordelingen uit.
2. Pas je eigen URS aan op basis van de twee peer-beoordelingen die je zult ontvangen van de andere studenten en lever het URS opnieuw in via Canvas.

Software Architectuur Document (SAD) voor GSO3 Killer App:

1. Stel een klassendiagram op van het domein van jouw applicatie in visual paradigm.
2. Stel een componentendiagram op in visual paradigm.
3. Bedenk hoe je sessies kunt realiseren voor jouw applicatie: hoe zorg je ervoor dat een persoon niet ongeoorloofd een actie uit kan voeren waartoe hij niet gerechtigd is?

**Les 11 (2 x 2 uur)**

Onderwerp: Software Architectuur Document (SAD)

Software Architectuur Document (SAD) voor GSO3 Killer App:

Maak een architectuur document op basis van het template document ArchitectuurDocumentTemplate.docx.

1. Werk het hoofdstuk ‘Domeinmodel’ uit.
2. Werk het hoofdstuk ‘Opdeling in componenten’ uit. Geef aan welke objecten (van welke klasse) er actief zijn binnen welke component. Zie de video StudentAdminComponentDiagram.mp4.
3. Werk het hoofdstuk ‘Communicatie tussen componenten’ uit. Stel een of meerdere (communicatie) klassediagrammen op in Visual Paradigm zodat de RMI-techniek tot uitdrukking komt in het model. In het geval van RMI volgens het server-push principe moet je gebruik maken van de klasse RemotePublisher en de interfaces IRemotePublisherForListener en IRemotePropertyListener (zie FontysPublisherHandleiding.docx). Geef in het klassediagram aan hoe je gebruik maakt van de klasse UnicastRemoteObject en de interfaces Remote en Serializable.
4. Zorg ervoor dat in het communicatie-klassediagram duidelijk wordt door welke klassen de interfaces van het componentdiagram gerealiseerd worden en aan welke interfaces de verschillende klassen behoefte hebben.
5. Communicatie tussen componenten komt neer op (remote) aanroep van methoden. De methoden waar een component behoefte aan heeft worden in de vorm van een 1 of meer interfaces gepresenteerd. Definieer de methoden (inclusief parameters en return waarden) voor alle interfaces in de klassediagrammen.
6. Vraag feedback aan medestudenten en de docent tijdens deze of de volgende les.

Beoordeling URS:

Docent geeft GO/NOGO voor de Killer App op basis van het aangepaste URS. Indien nodig zal de docent requirments aanpassen of extra requirements eisen.

**Les 12 (2 x 2 uur)**

Onderwerp: Observerpatroon (herhaling)

Werk verder aan het Software Architectuur Document (SAD) voor GSO3 Killer App:

1. Maak een gedetailleerd ontwerp per component (klassediagram, sequence diagrammen).
2. Werk het hoofdstuk ‘Persistentie per component’ uit.
3. Werk het hoofdstuk ‘Detailed design per component’ uit.
4. Werk het hoofdstuk ‘Realisatie niet-functionele eisen’ uit.
5. Maak een deployment diagram in visual paradigm en werk het hoofdstuk ‘Deployment’ uit.
6. Werk het hoofdstuk ‘Specificatie van interfaces’ uit. Defineer voor elke methode de naam, parameters (inclusief type) en returnwaarden, de preconditie, een beschrijving van de methode en mogelijke excepties. Uiteindelijk moet elke interface worden gespecificeerd met behulp van javadoc.
7. Schrijf de inleiding.
8. Laat je SAD reviewen door een medestudent.
9. Lever het SAD in via Canvas en bespreek het document met de docent.

**Inleveren voor einde van deze week**: SAD voor GSO3 Killer App.

**Les 13 (2 x 2 uur)**

Onderwerp: Testen - introductie

Realiseer je GSO3 Killer App (Les 13 – Les 16):

1. Gebruik GitLab voor versiebeheer.
2. Zorg dat je voor alle functionaliteit javadoc en unittests op orde hebt.
3. Gebruik JaCoCo om unittest coverage aan te tonen (zie S3\_JaCoCoPluginNetBeans.docx voor beschrijving installatie en gebruik JaCoCo).
4. Gebruik SonarQube om een indruk te krijgen van de kwaliteit van je code.
5. Stel een systeemtestplan met testgevallen op aan de hand van de templates Template TESTPLAN.docx (les 14) en Template TESTGEVALLEN.docx (les 15).
6. Pas het architectuur document aan indien nodig.
7. Voer de systeemtest uit en stel een testrapport op.
8. Voeg alle op te leveren producten samen in een .zip of .rar bestand en lever dit in in Canvas (les 16).

Beoordeling SAD:

Deze week en volgende week worden beoordelingsgesprekken gepland voor het SAD. De docent plant 15 minuten per student en geeft feedback en GO/NOGO voor realisatie van de Killer App op basis van het ingeleverde SAD. Deze gesprekken vinden plaats tijdens de ingeroosterde lesuren.

**Les 14 (2 x 2 uur)**

Onderwerp: Testen – systeemtestplan

Realiseer je GSO3 Killer App (Les 13 – Les 16):

Zie Les 13.

Beoordeling SAD:

Vorige week en deze week worden beoordelingsgesprekken gepland voor het SAD. De docent plant 15 minuten per student en geeft feedback en GO/NOGO voor realisatie van de Killer App op basis van het ingeleverde SAD. Deze gesprekken vinden plaats tijdens de ingeroosterde lesuren.

**Les 15 (2 x 2 uur)**

Onderwerp: Testen – testontwerp

Realiseer je GSO3 Killer App (Les 13 – Les 16):

Zie Les 13.

Voortgang realisatie GSO3 Killer App:

Deze week en volgende week worden gesprekken gepland om de voortgang van de realisatie van de GSO3 Killer App te bespreken. De docent plant 15 minuten per student en geeft feedback. Deze gesprekken vinden plaats tijdens de ingeroosterde lesuren.

**Les 16 (2 x 2 uur)**

Onderwerp: Testen - systeemtestplan

Realiseer je GSO3 Killer App (Les 13 – Les 16):

Zie Les 13.

Voortgang realisatie GSO3 Killer App:

Vorige week en deze week worden gesprekken gepland om de voortgang van de realisatie van de GSO3 Killer App te bespreken. De docent plant 15 minuten per student en geeft feedback. Deze gesprekken vinden plaats tijdens de ingeroosterde lesuren.

**Inleveren voor einde van deze week**: GSO3 Killer App.

**Les 17 (2 x 2 uur)**

Deze week worden de beoordelingsgesprekken voor de GSO3 Killer App gepland. De docent plant 15 minuten per student.

**Les 18 (2 x 2 uur)**

Reparatiemogelijkheid voor de GSO3 Killer App. In overleg met de docent.