

Softwareprojekt SS 2015

Dozentin: Grit Behrens
<mailto:grit.behrens@fh-bielefeld.de>

Kurspasswort: SWPrSS2015

Inhalt

1. Allgemeines zum Fach
2. Organisatorisches
 1. Termine,
 2. Bewertungen
3. Literaturangaben
4. Themenbeschreibungen

SW-Projekt – allgemein

- Aufsetzen auf dem Kenntnisstand aus Softwareengineering, OOP, PM, SWPr-Management und Useability
- Ein gemeinsames Aufgabenrahmenwerk, das individuell von der Projektgruppe gestaltet wird
- Projektgruppengröße: 6-11 Personen
- Programmiersprache: Java
- Jeder programmiert, hat aber zusätzlich unterschiedliche Rollen (Projektleiter, QS-Leiter, Doku-Organisator, Methodik-Experte, Werkzeug-Experte, ...)
- Entwicklung ist dokumentenzentriert (Sie dürfen mit Kanonen auf Spatzen schießen), zentrales Endergebnis: dokumentiertes Projekt
- Zentrales fachliches Ziel: möglichst selbständig ein SW-Projekt von der Anforderungsanalyse bis zur Abnahme durchziehen
- Lernziel: Projektarbeit mit Projektorganisation und Zusammenarbeit erfahren

Organisatorisches: Termine

Teilgruppe 1: Raum C1 Di 10:45 – 14:00 Uhr

Teilgruppe 2: Raum C3 Di 10:45 -- 14:00 Uhr

- Zusätzlich zu den Präsenzterminen wöchentlich 4 SWS sind selbständige Arbeiten im Umfang von 6 SWS nötig, die in wöchentlichen Stundenzetteln und wachsendem Teamprojektbericht ausgewiesen werden müssen.



Softwareprojekt					
Kenn-Nr.	Work-load	Credits	Studien-semester	Häufigkeit des Angebots	Dauer
4.1	150 h	5 cps	4. Sem.	Jedes SoSe	1 Semester
1	Lehrveranstaltungen Praktikum 4 SWS	Kontakt-zeit 4 SWS/60 h		Selbst-studium 90 h	geplante Grup-pengröße Praktikum 15

Organisatorisches: Termine

Termin	Inhalt	Bewertung
07.04.2015	Projektvergabe; Kick-off und Brainstorming, Rollenvergabe im Team, Spielregeln , Risikoanalyse , proaktives Risikomanagement, Werkzeugeinsatz, CodeRules, Vorgehensmodell	Moderation / keine Note
14.04.2015	Vorstellung Grobkonzept mit mind. Architekturübersicht, erste Version Lastenheft, Arbeitsteilung im Team, Schnittstellen zw. ProjektMA, grober Zeitplan	Moderation / keine Note
21.04.2015	Feinkonzept: Anforderungsanalyse , mit useCases, Sequenzdiagrammen, Datenbankdesign, Klassendiagramm und genauer Arbeitsplan mit Meilensteinplanung - Vertrag Lastenheft	Moderation/keine Note
28.04.2015	Abnahme Konzept (Präsentation des Teams unter Einbezug aller Mitglieder, incl. Lastenheft, aller UML-Diagramme und Teamarbeitsplan) (Abgabe alles als Hefter mit Dokumenten, Vorstellung zur Note als Präsentation mit Beamer)	1. Teilnote (Teamnote)
05.05. 2015	Projektfortschritt oder Meilenstein je nach Projektplanung	Evtl. 2. Teilnote nach Milestones
12.05.2015	Projektfortschritt oder Meilenstein je nach Projektplanung	Evtl. 2. Teilnote nach Milestones
19.05.2015	Projektfortschritt oder Meilenstein je nach Projektplanung	Evtl. 2. Teilnote nach Milestones
26.05.2015	Termin fällt aus, ggf. eigenständige Teamarbeit ohne Dozentin	Fällt aus
02.06. 2015	Projektfortschritt oder Meilenstein je nach Projektplanung	Evtl. 3. Teilnote nach Milestones
09.06.2015	Projektfortschritt oder Meilenstein je nach Projektplanung	Evtl. 3. Teilnote nach Milestones
16.06.2015	Übergabe Projektbericht (aktualisiertes Konzept, Teamarbeitsbericht, SW-Testbericht, SW-Dokumentation)	4. Teilnote (Teamnote)
23.06.2015	SW-Übergabe Teilgruppe 1 (PPT-Präsentation mit SW-Demo, Präsentation der SW-Funktionalität nach Lastenheft, Abgabe: Benutzerhandbuch, ggf. aktualisierter Projektbericht)	5. Note (Teamnote)
30.06.2015	SW-Übergabe Teilgruppe 2 in der Woche danach	5. Note (Teamnote)

Allgemeines zur Bewertung

- Jeweils 4h kompakte Arbeitszeit
- Anwesenheit und projektrelevante Aktivität wird vorausgesetzt und bewertet
- Projektplanung orientiert an individuellen Meilensteinen ist Aufgabe des Projekts
- Die Meilensteinergebnisse werden präsentiert (Funktionalität, konzeptuelle Inhalte (Designpattern, UML-Diagramme), SW-Implementierungen, SW-Doku, JUnit-Tests, SW-Qualität, Arbeitsanteile der Teammitglieder anhand ihrer Rollen im Team.
- In den individuellen Anteil der Gesamtbewertung geht das Arbeitsverhalten im Laufe des Projekts maßgeblich ein
- Wöchentliche Stundenzettel sind Pflichtabgaben
- Praktikumsleiter steht als Berater zur Aufgabenstellung in der Zeit zur Verfügung
- Praktikumsleiter unterstützt bei technischen Fragestellungen (die grundsätzlich vom Team zu lösen sind)
- Praktikumsleiter diskutiert mit Projektmitarbeitern ihre geschafften und geplanten Arbeiten unabhängig von den persönlichen Implementierungsmeilensteinen (Teil der Bewertung)

Bewertung

Praktikum:

- Prüfungsform „Projektabgabe“ im Sinne der Prüfungsordnung
 - Anwesenheitspflicht im Praktikum ($> 75\%$)
 - Wiederholbarkeit gegeben (aber: immer alles wieder komplett neu, da Teamarbeit)

Noten:

- Die Gesamtnote für das Modul errechnet sich als arithmetisches Mittel aller fünf Teilnoten:
 - Konzept
 - Implementierungsmeilenstein 1
 - Implementierungsmeilenstein 2
 - Projektbericht
 - SW-Übergabe
- In die Teamnoten geht zusätzlich zur Teamleistung nach Ermessen des Dozenten ein individueller Anteil mit ein der das jeweilige Arbeitsverhalten des Studierenden innerhalb des Teams einschätzt. Pro Termin und Studi gibt es individuelle Aktivitätspunkte.

Werkzeuge

- Zur Softwareentwicklung und für konzeptuelle Arbeiten wird ausschließlich Eclipse mit Plugins eingesetzt. Für UML-Diagramme wird das Umlet-Plugin genutzt, kein anderes Tool auch kein Powerdesigner.
 - es gibt professionellere Werkzeuge als Umlet, die aber nicht generell frei verfügbar sind (jedes Unternehmen kocht hier seinen eigenen „Werkzeugbrei“ zusammen)
 - Eigene Fähigkeiten in der Umsetzung sollen geschult werden, UML soll nach Standard eingesetzt werden
 - PowerDesigner verhindert Lernfortschritt durch zu viel Automatismus bei der Codegenerierung, UML-Standard wird nicht eingehalten
- Zur SW-Versionsverwaltung kann wahlweise GIT oder Subversion eingesetzt werden
- Zur Verwaltung, Organisation und zum Protokollieren der Teamarbeiten kann eine beliebige Software zur Unterstützung eingesetzt werden, auch manuelles Erstellen der Dokumente ist möglich.

Bewertungskriterien für Projektberichte

1. Inhaltliche Vollständigkeit und Themenbezug:

- Einführung mit Projektbeschreibung
- Vorstellung der Gruppenmitglieder und Ihre Rollen und Aufgaben im Team
- Konzeptionelle Arbeiten: z.B. mit Auswahl der UseCases und SW-Übersichts-Architektur, Klassendiagramm, Sequenzdiagramm alle textuell beschrieben
- Code-Ausschnitte zu aussagekräftigen Details
- Ausblick, Fazit
- Installationshinweise und Benutzerhandbuch zur Software
- Anlagen(Teamarbeitsplan mit Teamarbeitsbericht, SW-Design mit kompl. Use Case Diagrammen, kompl. Klassendiagrammen, ggf. ER-Diagramm, SW-Testbericht, SW-Dokumentation) Klassendiagramm, ER- Diagramm und SW-Dokumentation ggf. in elektronischer Form)

2. Schreibweise

- Verständliche Ausdrucksweise,
- durchgängiger Stil, immer gleiche Abkürzungen und Bezeichnungen
- Grammatikalisch und orthografisch korrekt

Bewertungskriterien für Präsentationen

1. Inhaltliche Vollständigkeit und Themenbezug:

- Projektbeschreibung
- Vorstellung der Gruppenmitglieder und Ihre Rollen und Aufgaben im Team
- Konzeptionelle Arbeiten
- SW-Architektur
- Code-Implementierungen mit aussagekräftigen Details
- Demonstration der SW
- Ausblick, Fazit

2. Übersichtlichkeit und Leserlichkeit der Folien

- Schriftgröße
- Passende und hilfreiche Grafiken
- Orientierungshilfen während des Vortrages

3. Vortragsweise

- Verständlichkeit (Lautstärke, Wortwahl, didaktische Methodik)
- Spannung im Vortrag, Aufmerksamkeitsgewinn beim Publikum
- Gleichmäßige und passende Aufteilung der Vortragsteile in der Gruppe : jeder kommt zu Wort

Inhalte der SW-Übergabe-Prüfungen

- 1. SW-Übergabe mit Präsentation und Demo der SW auf Server der FHBI**
- 2. Zeigen des Quellcodes anhand SW-Doku und Quellen auschnittsweise**
- 3. Bedien- und Funktionstest durch den Prüfer**
- 4. Prüfungsgespräch mit Dozentem in der Gruppe zum Thema Projektdurchführung**

Literatur für dieses Fach

- 1) M.Hitz, G.Kappel, E. Kapsamer, W.Reschitzergger „UML@Work – Objektorientierte Modellierung mit UML 2“, dpunkt.verlag, Heidelberg 2009, *ISBN 3-89864-261-5*
- 2) Bernd Oestereich, Analyse und Design mit UML 2.1, 8. Auflage, Oldenbourg, München
- 3) Stephan Kleuker „Grundkurs Software-Engineering mit UML – Der pragmatische Weg zu erfolgreichen Softwareprojekten“, Vieweg+Teubner Studium, Wiesbaden 2009, *ISBN 978-3-8348-0391-7*
- 4) ...

Themenbeschreibung 1: Lernplattform

Aufgabe:

Sie sollen eine serverbasierte Software einer Lernplattform erstellen.

Anforderungen des Kunden:

- Der Server soll einen Lernkatalog zu unterschiedlichen Themen bereit halten können. Beispielhaft sollen die Prüfungsfragenkataloge aus „Einführung ins Berufsfeld“ (mind. die ersten drei Veranstaltungen) eingepflegt werden.
- Ein Lernkatalog besteht aus Fragen zu einem Themengebiet. Unterschiedliche Fragenarten sollen dabei möglich sein. Die Fragen sollen auch Multimediainhalte enthalten können. (Sound/Bild), beliebige eigene Beispiele einpflegen.
- Benutzer sollen sich auf dem Server anmelden können, um online über einen Client ein Themengebiet zu lernen. Das System soll dabei die Fragen des Lerngebietes stellen und den Wissenstand des Benutzers auf Grund seiner Antworten protokollieren.
- Der Fragenkatalog soll zu Fragegebieten Links zu Ressourcen enthalten, die die abgefragten Themengebiete vertiefen und dem lernenden entsprechend seines Kenntnisstandes empfohlen werden.
- Es soll eine online-Lernmöglichkeit geben, in der der komplette Fragenkatalog eines Themas auf dem Client geladen wird.
- Es soll eine Möglichkeit geben den Fragenkatalog eines Themengebietes zum Lernen auf Karteikarten auszudrucken.
- Es soll die Möglichkeit geben, einen Kurs neu zu erstellen und neue Fragen zu editieren.
- I18N: es soll möglich sein, Fragen mit internationalen Zeichen zu verwalten (griechische, römische, arabische . . . Schriftzeichen)
- Der Quelltext soll auf Englisch kommentiert sein.
- Der Quelltext soll eine Qualität haben, die es erlaubt ohne sich schämen zu müssen nach Abschluss des Projektes das Programm als sourceforge-Projekt öffentlich zu machen und weiter zu pflegen. Diesbezügliche Coderules sollen definiert und eingehalten werden.
- Das Lernspiel soll Spaß machen und motivierend auf die Studis wirken.

Themenbeschreibung 2: Lehrdeputatsplanung

Aufgabe:

Sie sollen eine Software zur Lehrdeputatsplanung erstellen.

Anforderungen des Kunden:

- Die Software soll dem Studiengangsleiter die Möglichkeit bieten, Lehrdeputatsplanungen für Studiengänge zu unterstützen.
- Der Benutzer soll sich anmelden können.
- Der Benutzer soll einen Studiengang anlegen, einsehen und den Studienverlauf editieren können. Beispielhaft sollen die aktuellen Studienverläufe der beiden Informatikstudiengänge Bachelor und Master eingepflegt werden.
- Die Software soll die Möglichkeit bieten, Studienverlaufspläne als Text im pdf-Format auszugeben und als verständliche Grafik auszugeben und auszudrucken.
- Die Software soll die Möglichkeit bieten, Gruppengrößen zu editieren.
- Die Software soll die Möglichkeit bieten, Dozenten einzupflegen und zu verwalten und die jeweiligen Lehrdeputate zu planen, für vergangene Semester zu speichern, Reduktionen aufzunehmen und für kommende Semester neu zu berechnen.
- Eine Mehrfachbesetzung der Dozenten in mehreren Studiengängen ist zu beachten.
- Die Zwischenstände in der Deputatsplanungen sollen als pdf ausgedruckt werden können.
- Zur Planungsunterstützung soll es möglich sein, für Zwischenstände Mails an alle oder ausgewählte Dozenten zu generieren, um deren Einverständnis oder Änderungswünsche in der Rückantwort aufnehmen zu können.
- Die Software soll eine sehr gut bedienbare, übersichtliche und aussagekräftige grafische Oberfläche besitzen.
- Der Quelltext soll auf Englisch kommentiert sein.
- Der Quelltext soll eine Qualität haben, die es erlaubt ohne sich schämen zu müssen nach Abschluss des Projektes das Programm als sourceforge-Projekt öffentlich zu machen und weiter zu pflegen. Diesbezügliche Coderegeln sollen definiert und eingehalten werden.

Projektname: Lernplattform/Lehrdeputatsplanung (TL Student X)

Termin : Di 10:45-14:00 Uhr

Teilgruppe 1:

/acc. svn:

/acc. git:

Bewertete Leistung	Termin	Note + Bemerkungen zur Abnahme
Konzept	28.04.	
Implementierungsmeilenstein 1		
Implementierungsmeilenstein 2		
Projektbericht	16.06.	
Softwareübergabe	23.06.	
Gesamtnote		

Projektname: Lernplattform/Lehrdeputatsplanung (TL Student X)

Termin : Di 10:45-14:00 Uhr

Teilgruppe 2:

/acc. svn:

/acc. git:

Bewertete Leistung	Termin	Note + Bemerkungen zur Abnahme
Konzept	28.04.	
Implementierungsmeilenstein 1		
Implementierungsmeilenstein 2		
Projektbericht	16.06.	
Softwareübergabe	30.06.	
Gesamtnote		