



Software Engineering Übung 7

Sprint 3 und Abschlusspräsentation des Projektes

1 Informationen

1.1 Daten

- Ausgabe Di 3.12.2013
- Abgabe So 15.12.2013 bis 23:59 Uhr
- Vorträge am Di 17.12.2013 um 10:00 Uhr
- Besprechung am Di 17.12.2013 um 11:15 Uhr

1.2 Formales

Die Lösungen für Teilaufgaben 7.1-3 sollen als PDF Datei mit dem Namen **Ex 7 [TeamName].pdf** abgegeben werden, wobei **[TeamName]** der Name des Teams ist. Die PDF Datei sollte ausserdem ebenfalls Ihren Namen beinhalten.

Die Folien für die Vorträge sollen als PPT(X) Datei mit dem Namen **Ex 7 [TeamName].ppt(x)** abgegeben werden, wobei **[TeamName]** der Name des Teams ist.

Bitte senden Sie diese zwei Dateien als Archiv (ZIP File) via OLAT – benennen Sie das Archiv anhand der oben erwähnten Konventionen.

Alle Teilaufgaben sind als Gruppenarbeit in Ihrer Übungsgruppe zu bearbeiten. Jedes Gruppenmitglied muss über alle Teile der Lösungen Auskunft geben können. Verspätete Abgaben werden korrigiert, aber nicht bewertet.

2 Überblick und Ziele

In Übung 6 haben Sie die Sprint-Auftragsliste (*Sprint Backlog*) für Sprint 3 erstellt, inklusive der zu implementierenden Benutzergeschichten (*User Stories*) und deren Prioritäten. Dann haben Sie die Benutzergeschichten für Sprint 3 auf einzelne Aufgaben bzw. Arbeitseinheiten (*Tasks*) heruntergebrochen und diese den einzelnen Gruppenmitgliedern zugeteilt.

Diese Übung besteht nun aus dem Ausführen der Aufgabenbeschreibung für Sprint 3, und resultiert sowohl in einer lauffähigen, getesteten, bereitgestellten (deployed) und fertiggestellten Applikation, als auch in der Abschlusspräsentation Ihres Endprojektes.

Deshalb werden Sie die meiste Zeit dieser Übung für das Implementieren der von Ihnen beschriebenen Aufgaben und für das Testen verwenden. Nachdem die Implementierung gegen Ende des Sprints vollendet ist, werden Sie die Abschlusspräsentation Ihres Projektes durchführen.

3 Aufgabenstellung

Alle Teilaufgaben basieren auf der Anforderungsspezifikation einer Applikation zur Visualisierung von Abstimmungen, welche Sie in Übung 2 erstellt haben, sowie auf den zugehörigen Informationen in der Aufgabenstellung von Übung 2, und der Besprechung mit dem Auftraggeber am 8.10. Wenn Sie Unklarheiten haben, dann benutzen Sie das dazu eingerichtete Online Forum in OLAT oder wenden sich per E-Mail an ihren Auftraggeber.

Wie schon früher erwähnt, verwenden wir in unserem Übungsprojekt einen hybriden Entwicklungsprozess. Die vergleichsweise ausführliche Dokumentation von Anforderungen und Architektur am Beginn der Entwicklung ist eher typisch für ein ergebnisorientiertes Phasenmodell. Die Realisierung in Inkrementen fester Dauer dagegen entspricht einer agilen Vorgehensweise und lehnt sich an Scrum an. Wir verwenden jedoch wesentliche Elemente von Scrum (z.B. Scrum-Master, Daily Scrum) nicht, weil diese sich im gegebenen Kontext (viele Übungsgruppen, begrenzte Arbeitskapazitäten von Teams und Betreuern) nicht sinnvoll realisieren lassen.

Wir haben diese hybride Vorgehensweise bewusst gewählt, um den im Rahmen eines Übungsprojekts erzielbaren Lernerfolg zu optimieren.

Teilaufgabe 7.1: Das Ziel von Sprint 3 (1 Punkt)

Basierend auf den bisher erhobenen Anforderungen und dem vom Produkteigner (gespielt von den Kursassistenten) erhaltenen Input am 3. Dezember soll Ihr Team gemeinsam das Ziel des Sprints festlegen. Beschreiben Sie kurz, was der Sprint zu erreichen versucht. Beispiel: „Implementiert grundlegende Einkaufswagen-Funktionalität mit dem Hinzufügen, Entfernen und Aktualisieren von Stückzahlen.“ Das Sprint-Ziel kann verwendet werden, um Aussenstehende kurz und prägnant über den Sprint zu berichten. Es gibt ständig Interesseneigner (*Stakeholders*), die wissen wollen, woran das Team gerade arbeitet, aber keine Details über die einzelnen Benutzergeschichten in der Produkt-Auftragsliste wissen müssen. Der Erfolg des Sprints wird später während der Sprint Review Sitzung anhand des Sprint-Ziels beurteilt (17.12.2013), und nicht nur anhand der einzelnen Benutzergeschichten.

Die Abgabe dieser Teilaufgabe besteht aus einem Satz, welcher das Ziel des Sprints beschreibt.

Teilaufgabe 7.2: Durchführung von Sprint 3 (8 Punkte)

Das Hauptziel dieser Aufgabe besteht aus der Vollendung und Auslieferung des ersten Release Ihrer Applikation: am Ende von Sprint 3 haben Sie ein funktionierendes, mit Google App Engine bereitgestelltes (deployed) Pilotsystem, welches für die geplanten Endbenutzer einen Mehrwert liefert. Der Schwerpunkt dieses Sprints liegt daher einerseits auf der Vervollständigung der Implementierung Ihrer kritischen Benutzergeschichten sowie auf Test und Fehlerbehebung.

Nach der Planung in Übung 6 sollte jedes Teammitglied ungefähr dieselbe Auslastung haben. Falls Sie nach dem Erhalt des Feedbacks am 3. Dezember das Gefühl haben, die Aufgaben neu verteilen oder anpassen zu müssen, dürfen Sie dies tun. Vergessen Sie nicht, die Änderungen zu dokumentieren und bei der Abgabe dieser Übung ebenfalls mitzuliefern.

Teilen Sie die Arbeit entsprechend Ihrer aktualisierten Aufgabenverteilungsliste auf die Teammitglieder auf. Führen Sie die Ihnen zugewiesenen Aufgaben aus, indem Sie entsprechende Software entwerfen, codieren und gemäss Teilaufgabe 7.3 testen. Erzeugen Sie am Ende des Sprints ein Release, welches mit Google App Engine bereitgestellt (deployed) ist. Nehmen Sie eine Selbsteinschätzung vor, wie viel Sie in den drei Sprints erreicht haben und geben Sie Ihrem Release entsprechend eine Nummer n mit $0.7 \leq n \leq 1.0$.

Die Abgabe dieser Teilaufgabe besteht aus dem funktionierenden, gemäss Teilaufgabe 7.3 getesteten Release Ihrer Applikation, welche Ihr Team dem Produkteigner (Kursassistenten) am 17. Dezember präsentiert.

Hinweise:

- Die Arbeiten der Teilaufgaben 7.2, 7.3a und 7.3b sollten *zeitlich verzahnt* und *nicht* sequenziell nacheinander ausgeführt werden.
- Planen Sie die Implementierung so, dass in Ihrem Team genügend Arbeitskapazität für den Systemtest sowie die Suche und Behebung von Defekten (Teilaufgabe 7.3 c) verbleibt.
- Ihr Team wird *nicht* mit Punktabzügen bestraft, wenn noch unvollständige Benutzer geschichten bestehen, solange diese nicht als kritisch oder wichtig eingestuft wurden. Falls dies der Fall wäre (dass unvollständige kritische oder wichtige Benutzer geschichten existieren), müssen Sie eine Liste mit allen unvollständigen Benutzer geschichten einreichen. Zusätzlich sollten Sie uns erklären, wieso sie nicht fertiggestellt wurden. Falls Sie kein lauffähiges, getestetes Pilotsystem vorweisen können, ziehen wir Ihnen Punkte ab.
- Bitte stellen Sie sicher, dass die URL zu Ihrer bereitgestellten Applikation im abgegebenen PDF enthalten ist.

Teilaufgabe 7.3: Testen (7 Punkte)

a) Komponententest (3 Punkte)

Schreiben Sie einen Satz von JUnit Tests, welche alle Methoden überdecken, die Sie in diesem Sprint neu geschrieben haben, d.h. mindestens ein Test pro Methode. Falls einige der Methoden Ausnahmen (exceptions) werfen, decken Sie auch diese mit Tests ab. Wir empfehlen Ihnen, den Code (vgl. Teilaufgabe 7.2) und die zugehörigen JUnit Tests parallel zu entwickeln. Die Entwicklung einer Methode oder Klasse ist erst abgeschlossen, wenn die zugehörigen JUnit Tests keine Fehler anzeigen. Falls Sie Software ändern, die Sie in Sprint 1 oder Sprint 2 erstellt haben, müssen Sie die zugehörigen, in Sprint 1 oder Sprint 2 erstellten JUnit Tests anpassen. Führen Sie nicht nur die neuen und geänderten JUnit Tests aus, sondern *sämtliche* bestehenden JUnit Tests. Mit diesem Regressionstest prüfen Sie, ob die Erweiterung Ihres Systems mit neuer Software möglicherweise Fehler in der bestehenden Software verursacht.

b) Testvorschrift für den Systemtest (2 Punkte)

Erweitern Sie die Testvorschrift, die Sie in Übung 6 für den manuellen Systemtest von Sprint 2 erstellt haben, um Testfälle für die in Sprint 3 implementierte Software. Erneut geht es primär darum, mit diesem Test das Verhalten Ihres Systems und die Benutzerschnittstelle zu testen. Orientieren Sie sich bei der Auswahl der Testfälle an den Abnahmekriterien, die Sie zu den für Sprint 3 ausgewählten Benutzer geschichten geschrieben haben. Falls Sie Software ändern, die Sie in Sprint 1 oder Sprint 2 erstellt haben, müssen Sie die zugehörigen Systemtestfälle anpassen.

c) Durchführung des Systemtests (2 Punkte)

Führen Sie den Systemtest mit der in Teilaufgabe 7.3b erstellten Testvorschrift durch und dokumentieren Sie das Resultat. Es sind *alle* Testfälle, d.h. die bisherigen Testfälle aus Sprint 1 und Sprint 2 und die in Teilaufgabe 7.3b neu erstellten bzw. geänderten Testfälle zu testen. Wenn der Test Fehler findet, suchen Sie die fehlerverursachenden Defekte und beheben diese. Danach ist der Systemtest zu wiederholen. Wenn Ihnen im Rahmen der für den Sprint verfügbaren Arbeitskapazität keine Zeit mehr für die Defektsuche und -behebung bleibt, erstellen Sie für jeden nicht behobenen Fehler einen Fehlerbericht (bug report). Dieser besteht aus folgenden Feldern: Fehler-ID, Kurzbeschreibung des Fehlers in wenigen Worten, Nummer des Testfalls, bei dem der Fehler auftritt, Verfasser, Priorität und Aufwand. Die Felder für Priorität und Aufwand bleiben vorerst leer.

Die Abgabe von Teilaufgabe 7.3 besteht aus den von Ihnen geschriebenen JUnit Tests, der Systemtestvorschrift mit den eingetragenen Befunden der letzten Durchführung des Systemtests sowie gegebenenfalls den Fehlerberichten für alle am Ende des Sprints nicht behobenen Fehler.

Teilaufgabe 7.4: Abschlusspräsentation des Projektes (4 Punkte)

Nachdem Sie die Implementierung (d.h. Codierung, Integration, Test und Bereitstellung (deployment)) für Sprint 3 beendet haben, ist es jetzt Zeit, Ihre Abschlusspräsentation vorzubereiten.

Am Dienstag, 17. Dezember, bekommen Sie als Team maximal 3 Minuten Zeit Ihr Produkt zu präsentieren. Dabei können Sie dem Professor, den Kursassistenten und TAs (die Produkteigner sind) sowie Ihren Mitstudierenden, einen kurzen Überblick über Ihre Ergebnisse geben. In der nachfolgenden Übungsbesprechung werden Sie Ihr System im Detail vorstellen können..

Wenn Sie Ihre Abschlusspräsentation vorbereiten, sollten Sie das Folgende beachten:

- zeigen Sie Ihr Produkt und seine Haupteigenschaften;
- präsentieren Sie so, dass alle Interesseneigner (Stakeholder), auch diejenigen, die Ihr Projekt nicht kennen, ein gutes Verständnis bekommen;
- konzentrieren Sie sich auf die Haupterfolge Ihres Teams;
- meiden Sie technische Implementierungsdetails;
- die Präsentation wird nur von einem Gruppenmitglied in einer professionellen Weise vorgestellt – d.h. Sie müssen vorher üben;
- den Vortragenden können Sie selber wählen.

Regeln betreffend den Präsentationsfolien:

- Sie können Ihre Folien als PPT oder PPTX einreichen;
- keine anderen Formate als PPT oder PPTX werden akzeptiert;
- die Anzahl der Folien können Sie selber bestimmen, solange die Zeit nicht überschritten wird;
- Animationen sind erlaubt;
- alle Präsentationen werden auf einem MacBook mit Microsoft PowerPoint für den Mac 2011, Version 14.1.3 laufen.

Bitte beachten Sie, dass wir den Zeitplan strikt durchsetzen werden: wenn Sie nach drei Minuten nicht fertig sind, werden wir Ihre Präsentation abbrechen. Die Präsentationen finden am 17.12. zwischen 10:00 und 11:00 statt.

Die Abgabe dieser Teilaufgabe besteht aus den PPT(X) Folien für die Abschlusspräsentation.

Ein paar Tipps: Wie Sie sich für die Diskussion am 17. Dezember vorbereiten

In der Diskussion am 17. Dezember werden Sie den Produkteigner (Kursassistenten) treffen, welche/r beurteilt wird, welche Positionen (Benutzergeschichten, Arbeitseinheiten) als abgeschlossen betrachtet werden. Jedes Team bekommt 15 Minuten Besprechungszeit.

Sie werden exakt *12 Minuten* Zeit haben, um Ihr Produkt zu demonstrieren (falls Sie überziehen, werden Sie unterbrochen/abgewürgt). Planen Sie etwa sechs Minuten für Ihre Demonstration und sechs Minuten für die Beantwortung der Fragen, welche der Produkteigner Ihnen stellen wird. Nutzen Sie diese Zeit, um Ihr System zu erklären und zu zeigen, welche Benutzergeschichten aus der Sprint-Auftragsliste implementiert sind. Es wird dringend empfohlen, dass Sie Ihre Demonstration im Vorfeld üben.

Der Produkteigner wird Ihnen Fragen zu Ihren Entwurfs- und Implementierungsentscheidungen stellen. Diese Fragen werden an einzelne Teammitglieder gerichtet. Sie sollten daher nicht nur den von Ihnen selbst geschriebenen Code kennen, sondern auch über das Gesamtdesign Ihres Pilotsystems Bescheid wissen. Zusätzlich sollten Sie sowohl JUnit Tests durchlaufen lassen, als auch einige manuelle Tests demonstrieren.

Nach der Demonstration und den Fragen, werden Sie *3 Minuten* Zeit für eine kurze Projekt-Retrospektive: was lief gut, was lief falsch, welche Lektionen haben Sie gelernt.