

Abschlussbericht

Team: 4 - iKita

Daniel Walder:: 1518406
Lucas Markovic: 1519597
Kerstin Klabisch: 1518671
André Potocnik: 1518553
Fabio Valentini: 1018782

Proseminargruppe: 1

Datum: 19.06.2017

1 Analyse des Projektablaufs

Dass dieses Projekt sehr aufwändig sein wird, wurde uns zu Beginn des Proseminars mitgeteilt. Doch der tatsächliche Aufwand war sehr viel höher als erwartet. Es wurde gesagt wir sollen mit etwa 10 Stunden pro Woche kalkulieren. Nicht alle Teammitglieder machten eine genaue Zeiterfassung, aus den vorhandenen Daten kann aber geschlossen werden, dass im Durchschnitt 10h (teilweise mehr) pro Woche in das Proseminar investiert wurden.

Bis zum Meilenstein "Increment 1" waren wir vor dem Zeitplan, da uns die Osterferien zu Gute kamen. In der Mitte des Zeitplans wurden manche Teammitglieder zurück geworfen wegen der Komplexität mancher Views und Klassen. Es wurde versucht so gut wie möglich den Zeitplan aufzuholen. Leider sind deswegen einige Sonderfeatures vom letzten Increment auf der Strecke geblieben. Doch alle Pflichtfeatures und die wichtigsten Sonderfeatures sind implementiert. Gegen Ende wurde das Testen intensiviert und es tauchten immer wieder Bugs auf, die so unauffällig waren, dass sie gar nicht bemerkt wurden, aber doch signifikant waren.

Etwa 20% sind dem Entwurf des Systems zuzuschreiben. Im Laufe des fortgeschrittenen Projekts wurden aber immer wieder kleine Änderungen vorgenommen, die wir wegen unserer Unerfahrenheit nicht berücksichtigt haben.

2 Analyse des implementierten Systems

Das grobe Konzept hat an sich sehr gut gepasst. An die ursprüngliche Idee, wie das Projekt sein soll, wurde sich prinzipiell bis zum Ende gehalten.

Natürlich wurden fortlaufend Anpassungen vorgenommen. Änderungen und Klassen-Properties waren nicht unüblich, da Eigenschaften nicht berücksichtigt wurden oder Datentypen nicht ideal gewählt erschienen.

Zum Teil wurden die Views an die vorhandenen Technologie-Komponenten angepasst. In einem Fall wurde an einer Idee gebastelt und nach Tagen an Arbeit wurde festgestellt, dass dieser Weg eine Sackgasse ist und die Idee wurde wieder verworfen (Child Calendar). Was sehr zeitraubend sein kann.

Welches Kriterium nicht unterschätzt werden darf, ist ein uneiniges Team. Bei einigen wenigen Entscheidungen war sich das Team uneins und dies konnte zu hitzigen Diskussionen ausarten, was ebenfalls Zeit und Nerven kostet.

Nützlich könnte ein Mensch sein, der nicht farbenblind und ein Auge für Ästhetik hat.

Alle geplanten Funktionen konnten leider nicht umgesetzt werden, wie z.B. die Picture Gallery oder das Terminal.

3 Ursachenanalyse

Eklatante Diskrepanzen wurden nicht wirklich festgestellt. Wenn junge und unerfahrene Studenten ihr erstes großes Projekt im Team bewältigen, sind Fehler in der Architektur und Ideen, die nicht so umgesetzt werden können, ganz natürlich.

Eine stabile Basis und durchdachte Software-Architektur erleichtert das weitere Arbeiten ungemein. Doch erreicht man diese Eigenschaften nur mit viel Erfahrung.

Wichtig ist eine Person die den Überblick behält und sich um Koordination und Organisation bemüht. Vermutlich wäre es sinnvoll dieser Person mehr Zeit für solche Arbeiten einzuräumen.

Für das intensive Testen und die anschließende Fehlerbehebung sollte ebenfalls um einiges mehr an Zeit eingeplant werden.

4 Erfahrungen mit der Entwicklungsumgebung

In unserem Team wurde Eclipse und IntelliJ als IDE verwendet. Mit den "üblichen" Verdächtigen kann ansich nichts falsch gemacht werden. Das Internet bietet sehr viele Hilfestellungen für alle möglichen Probleme an.

Allerdings hatte ein IntelliJ-Nutzer das Problem, das die xhtml-Dateien kein Syntax-Highlighting bot, dieses Problem konnte bis zum Schluss trotz intensiver Fehlersuche, nicht behoben werden.

Des Weiteren kann es bei Eclipse vorkommen, dass neu hinzugefügte Dependencies nicht automatisch heruntergeladen werden. Durch manuelle Installationen oder durch "cleanen" der fehlerhaften Abhängigkeiten kann das Problem aber umgangen werden.

5 Feedback zur Proseminar-Organisation

Positiv ist zu bemerken, dass ein Projekt mit tatsächlichem Hintergrund als sehr sinnvoll erscheint. Da sowieso vorhandene Kapazitäten genutzt werden und mehr Motivation dem Projekt entgegengebracht wird. Einfacher erwiesen sich dadurch manche Design Entscheidungen, da man sich besser in wirkliche Endbenutzer reinversetzen konnte.

Vorteilhaft wäre es, wenn das Codereview erst gegen Ende stattfindet, wenn das Projekt abgabebereit ist. Wenn es vorher stattfindet sind viele Kritiken nicht wirklich hilfreich, da man sich noch mitten in der Entwicklung befindet und die meisten Probleme davon eh bekannt sind. Wobei auch klar ist, dass die Zeit zu knapp ist und dies schwer zu realisieren ist.

Negativ aufgefallen ist die mangelhafte Dokumentation von JSF und Hibernate. Viele Probleme ließen sich nur durch "ausprobieren" beheben - oder durch Hilfe von Seiten wie StackOverflow.

6 Bekannte Bugs

Liste von Bugs, welche bis zum Abgabezeitpunkt nicht behoben werden konnte:

- **Primefaces "schedule" Kinder Kalender**
manchmal wird ein Monat beim zurückgehen übersprungen.
- **Datenbank / Hibernate**
Es fliegt eine Exception der Datenbank mit der Meldung "multiple representations of the same entity"
M:N - Beziehungen in der Datenbanken funktionieren nicht fehlerfrei. Wird eine Person mehrmals zb. als Bezugsperson bei einem Kind hinterlegt, wird diese Exception geworfen.
- **Primefaces "schedule" Pädagogen Kalender**
In der Tagesansicht kann es sein, dass nicht immer alles Kinder aufgelistet werden. Dieses Problem tritt nur auf wenn das Projekt mit Vagrant gestartet wird, mit spring-boot klappt es einwandfrei.