|  |
| --- |
| Elmer Lukas, Heidt Christina, Treichler Delia  19. Dezember 2011 |

|  |
| --- |
| Studienarbeit |
| Projektretrospektive |
| [Geben Sie den Untertitel des Dokuments ein] |

****

# Dokumentinformationen

## Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Änderung | Autor |
| 19.12.2011 | 1.0 | Erste Version des Dokuments | dtreichl |

## Inhaltsverzeichnis

[1 Dokumentinformationen 1](#_Toc312069778)

[1.1 Änderungsgeschichte 1](#_Toc312069779)

[1.2 Inhaltsverzeichnis 1](#_Toc312069780)

[2 Vorgehen und Technologien 2](#_Toc312069781)

[3 Persönliche Berichte 3](#_Toc312069782)

[3.1 Lukas Elmer 3](#_Toc312069783)

[3.2 Christina Heidt 3](#_Toc312069784)

[3.3 Delia Treichler 3](#_Toc312069785)

[4 Aufwandanalyse 3](#_Toc312069786)

# Vorgehen und Technologien

Zu Beginn von Project Flip 2.0 wurde eine grobe Projektplanung im Redmine (Projekplanungstool und Ticketingsystem) gemacht, in der die wichtigsten Aspekte beschrieben wurden und alle Kalenderdaten festgehalten wurden.

Während des Projektes wurde dann stark agil nach Scrum gearbeitet. Die 14 Wochen Dauer des Projektes wurde in 7 Sprints zerlegt, in denen man sich auf gewisse Punkte konzentrierte.

Zu Beginn jedes Sprints wurde die Planung für den Sprint durchgeführt, was sich als riesigen Vorteil erwies, denn so konnten die Anforderungen viel genauer aufgefasst werden.

War ein Feature für einen Sprint geplant, aber noch nicht fertig, wurde es in den nächsten Sprint verschoben, sodass der Abgabetermin eines Sprints immer eingehalten werden konnte.

Da Usability für das Projekt ein sehr wichtiger Punkt war, wurde benutzerorientiert entwickelt. So wurde vor dem Programmieren der Lösung viel Zeit in Personas, Szenarien und einen Papierprototypen investiert und viele neue Lösungen ausprobiert, was sehr interessant war.

Um die Codequalität zu gewährleisten und die Applikation einfacher weiterzuentwickeln, wurde sie in verschiedene kleine Teilprojekte aufgeteilt. Diese wurden dann auch mit Unit Tests stark überdeckt, zwischen 80% und 95%. Dies erwies sich als grossen Vorteil, da so schnell Probleme erkannt wurden, wenn etwas nicht wie gewünscht funktionierte.

Da die Microsoft Welt für alle Studenten noch unbekannt war, waren die Schätzungen nicht immer einfach. Das war ein weiterer Grund, um Scrum einzusetzen, denn so konnte der Umfang bei jedem Sprint angepasst werden und alle Mitglieder wurden dazu gezwungen, bei Problemen externe Hilfe zu suchen.

Die .NET / WPF Technologie ist zweifellos sehr gross, mächtig und interessant. Dies hat sich während des Projektes aber nicht immer nur positiv auf die Arbeit ausgewirkt, denn für gewisse Probleme wurde viel Zeit investiert, um ein kurzen und einfachen Befehl zu finden. WPF war auch in Bezug auf Performance nicht immer einfach.

Leider war der Zeitrahmen der Arbeit beschränkt und es konnte nicht alles so implementiert werden, wie wir es gerne gewollt hätten und es uns beim Papierprototyp vorgestellt hatten. Zum Beispiel hatten wir keine Zeit mehr, die Gesten zu implementieren, und auch für die Animationen stand nicht so viel Zeit zur Verfügung, wie wir es gerne gehabt hätten.

Sehr schön war das Gefühl, ein starkes, motiviertes Team mit viel Wissen hinter sich zu haben. So konnten wir während der vollen Projektdauer bei Unklarheiten um Rat fragen, ohne viel Zeit zu verlieren.

Ganz speziell unterstützt haben uns Markus Stolze (IFS, HSR), Christian Moser und Marco Balzarini (Zühlke Engineering AG). An dieser Stelle einen ganz herzlichen Dank an diese Personen.

# Persönliche Berichte

## Lukas Elmer

Ein sehr interessanter Aspekt an Scrum ist für mich, dass das Produkt zwar noch nicht fertig entwickelt ist, jedoch ein voll funktionsfähiger, stabiler und getesteter Prototyp entwickelt ist. Diesen könnte man bereits so benutzen, wie er jetzt ist, es ist einfach noch nicht die ganze Funktionalität vorhanden.

In einem nächsten Projekt würden wir wieder zweiwöchentliche Sprints ansetzen, also etwa 32h pro Person pro Sprint. Wenn also zu 100% an einem Projekt gearbeitet wird, können wöchentliche Sprints durchgeführt werden.

TODO: More

## Christina Heidt

TODO

## Delia Treichler

TODO

# Aufwandanalyse

Das Projekt war lief über ein Semester von 14 Wochen. Für das Modul werden 8 ETCS pro Student vergeben, pro Punkt wird mit einem Aufwand von etwa 30 Stunden gerechnet. Also standen uns Total etwa 3 \* 8 \* 30 = 720 Stunden zur Verfügung.

Natürlich mussten wir damit rechnen, dass mit einer neuen Technologie der Zeitrahmen ein wenig gesprengt werden muss. Deshalb wurde zu den 7 Sprints ein zusätzlicher Sprint 0 eingeführt, in dem wir uns bereits ein wenig in die neue Technologie einarbeiten konnten.

Der investierte Aufwand war aber kontinuierlich immer etwa gleich hoch und konstant etwas über der vorgegebenen Zeit. Nachfolgend eine kurze Übersicht über die einzelnen Sprints:

|  |  |
| --- | --- |
| Sprint | Aufwand in Stunden |
| 0 | 19,00 |
| 1 | 126,50 |
| 2 | 111,00 |
| 3 | 103,50 |
| 4 | 119,75 |
| 5 | 120,25 |
| 6 | 127,25 |
| 7 | TODO: Anpassen! |
| Total | TODO: Anpassen! |

TODO: More, Charts, …

Dok: Aufwand dokumentiert (nachgeführte Arbeitsliste pro Person) und sinnvoll analysiert (PieCart)