# Projektmanagement

## Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Änderung | Autor |
| 23.02.2012 | 1.0 | Erste Version des Dokuments | DT |
| 24.02.2012 | 1.1 | Review | CH |

## Projektplan

Das Management des Projektes Video Wall geschieht im Redmine (https://redmine.elmermx.ch/).

### Releases

Da das Projekt nach dem Vorgehensmodell Scrum (siehe Unterkapitel I.5 Vorgehensmodell) durchgeführt wird, gibt es nur zu Projektende einen Release.

Während des Projekts wird immer am Ende eines Sprints ein voll lauffähiger Prototyp vorliegen.

Folgender Release ist vorgesehen:

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Typ | Beschreibung | Datum |
| 1.0 | Finalversion | Fertige Version, Bugs gefixt | 15.06.2012 |

### Milestones

Ein Milestone ereignet sich jeweils am Ende eines Sprints. Die Sprints wie auch die Milestones sind im Redmine dokumentiert.

Auf der Website https://redmine.elmermx.ch/ steht der detaillierte Projektplan zur Verfügung. Die detaillierte Planung der jeweiligen Sprints erfolgt schrittweise nach dem Vorgehensmodell Scrum (siehe Unterkapitel I.5 Vorgehensmodell).

### Zeitplan und Zeiterfassung

Die einzelnen Arbeitspakete (Tickets) sind den jeweiligen Sprints zugeordnet. Das Projekt ist in vierzehn Sprints unterteilt. Das Ende eines Sprints entspricht jeweils einem Milestone.

Die komplette Zeitplanung und die Zeiterfassung werden auf dem Redmine-Server durchgeführt. Für jedes Arbeitspaket wird der Zeitaufwand geschätzt und ein Ticket erstellt. Diese Tickets werden den jeweiligen Sprints zugeordnet. Wurde an einem Ticket gearbeitet, wird die dafür aufgewendete Zeit auf das Ticket gebucht. Die Erfassung der Zeit für die jeweilig bearbeiteten Tickets geschieht sofort nach Abschluss der Arbeiten. Somit ist die Zeiterfassung stets aktuell.

Die Reportfunktion bietet einen Überblick über den geplanten und tatsächlichen Zeitaufwand vermitteln. Zudem ist es möglich, den Arbeitsaufwand mittels einer cvs-Datei zu exportieren und z.B. in Excel anschaulich darzustellen.

### Aufwandschätzung

Die Aufwandschätzung ergibt sich durch den geschätzten Aufwand pro Ticket im Redmine.

## Projektorganisation

### Team und Verantwortlichkeiten

#### Lukas Elmer (Abk. LE)

Kenntnisse in: Ruby on Rails, PHP, Python / Django, Typo3, Wordpress, Java,   
XHTML, JavaScript, C++, C#, Ubuntu Server

Rolle/Verantwortlichkeiten: Architektur, Serverunterhalt von Redmine, Konfigurations-  
management

Mailadresse : lelmer@hsr.ch

Skype Adresse: lukas.elmer

#### Christina Heidt (Abk. CH)

Kenntnisse in: Java, HTML/CSS, C++, C#, Photoshop

Rolle/Verantwortlichkeiten: Grafisches Design, Risikomanagement, Anforderungen,   
Sitzungsprotokollierung

Mailadresse: cheidt@hsr.ch

Skype Adresse: christina\_heidt

#### Delia Treichler (Abk. DT)

Kenntnisse in: Java, HTML/CSS, C++, C#

Rolle/Verantwortlichkeiten: Überwachung und Erstellung Projektplan, Teamsitzungen,   
Usability-Tests

Mailadresse: dtreichler@hsr.ch

Skype Adresse: de-lia

## Risiken

Das Risikomanagement befindet sich im Anhang (siehe TODO).

## Vorgehensmodell

Für das Projekt Video Wall wird hauptsächlich der Ansatz von Scrum verfolgt, weil dieses Vorgehensmodell auf die Eigenorganisation der einzelnen Teammitglieder ausgerichtet und äusserst produktiv ist, da Overhead so weit wie möglich reduziert wird.

Die nachfolgende Tabelle zeigt auf, welche Elemente (Rollen, Meetings und Artefakte) von Scrum wie gehandhabt werden.

|  |  |
| --- | --- |
| Scrum-Element | Umsetzung |
| Rollen |  |
| Product Owner, Scrum Master | Diese Rollen können personalbedingt nicht besetzt werden. Die Aufgaben des Product Owners und des Scrum Masters werden vom Entwicklungsteam übernommen. |
| Meetings |  |
| Sprint Planung | Die Planung des darauf folgenden Sprints wird zu Ende des gegenwärtigen Sprints durchgeführt. |
| Daily Scrum | Das Meeting wird jeweils dienstags und freitags durchgeführt (es wird kein Protokoll geführt). |
| Sprint Review | Das Review Meeting findet jeweils am letzten Tag des aktuellen Sprints statt. |
| Artefakte |  |
| Product Backlog | Die Anforderungen an das Produkt sind als Tickets im Redmine erfasst. Die Schätzung des Aufwands geschieht für jede Anforderung auf dem entsprechenden Ticket nach dem Modell Planning Poker. |
| Sprint Backlog | Die für einen Sprint geplanten Aufgaben existieren im Redmine als Ticket, welche dem jeweiligen Sprint zugeordnet sind. Der Restaufwand, der für eine einzelne Aufgabe noch benötigt wird, ist über die Differenz der geschätzten und bisher gebuchten Zeit ersichtlich. Gleichzeitig dient das Redmine als Ersatz für das Taskboard. |
| Burndown-Charts | Das Gantt-Diagramm, welches sich im Redmine anzeigen lässt, erübrigt einen Burndown-Chart. |
| Impediment Backlog | Im Redmine Wiki besteht eine Seite zur Eintragung von Hindernissen und Problemen des Projektes. |
| Releaseplan | Die Tickets im Redmine bieten einen Überblick über den Zeitplan und die Termine/Meetings. Die erwartete Anzahl Sprints ist unter Roadmap ersichtlich. Für die Behandlung von Risiken siehe Unterkapitel I.4 Risiken. |

Das Verfassen formaler Dokumente sowie die Erstellung eines Architekturprototyps sind dem Vorgehensmodell RUP entnommen.