SOLL-KONZEPT

Rel. Userstory ID: US009

Version: V002

Inhaltsverzeichnis

[Versionsgeschichte 0](#_Toc439592851)

[Ausarbeitungsteil 1](#_Toc439592852)

# Versionsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Änderungsbeschreibung** | **Bearbeiter** |
| 001 | 03.02.16 | Dokumenterstellung | Jolitz, K.; |
| 002 | 03.02.16 | Inhalte eingefügt | Serfling, T. |
| 003 | 03.02.16 | Weitere Inhalte eingefügt | Jolitz, K.; |
|  |  |  |  |
|  |  |  |  |

# Vorgehensmodell

## Generelles

Viele Elemente von Scrum sind nicht vollständig benutzbar aufgrund der Projektvorgaben, daher benutzen wir eine abgewandelte Form von Scrum.

Zunächst haben wir keinen Product Owner und Scrum Master, sondern einen traditionellen Projektleiter.

Die User Stories (Arbeitspakete) werden gemeinsam in der Gruppe besprochen und erstellt. Übertragen werden diese auf die Website „Scrumdesk“, welche viele Funktionen und Erleichterungen für die Projektplanung bereitstellt.

Eine weitere Abweichung betrifft die Flexibilität. Sprints können in unserem Projekt verschiedene Längen haben und die Abgabe von User Stories ist streng zeitlich vorgegeben und kann nicht flexibel aufgeschoben oder verändert werden.

In den 4 Sprints ergeben sich nicht immer Prototypen der Software, sondern fest definierte Ergebnisse des Gesamtprojektes (Projektdokumentation, Prototyp, Abgabefassung, Präsentation).

Zur Synchronisation unserer Dokumente verwenden wir das Versionierungstool Git, mit Server Speicher von GitHub.

## Zeitplanung

siehe Anlage: „mp\_anwen\_scrumiverse.pdf“

## Projektstrukturplan

Siehe Anlage: „xyz“

# Lösungsansatz

## Generelles

Unter der Betrachtung der Konkurrenz verfolgen wir den Ansatz, ein mit von Scrum gestützte Online-Projektplanung zu erstellen. Dabei möchten wir eine ähnliche Plattform schaffen. Funktionen und Umsetzungen, welche wir für umständlich oder nicht intuitiv halten wollen wir verbessern und um mit den bereits existierenden Plattformen zu konkurrieren werden wir auch Funktionen, welche wir für sinnvoll erachten, für unsere Plattform adaptieren. Das Vorgehensmodell von Scrum soll besser und original getreu umgesetzt werden. Hierfür wollen wir die haptische Umsetzung in eine Online-Version bündeln und zentral erreichbar machen. Da uns auf den Konkurrenz-Plattformen schnell aufgefallen ist, dass ein Rechtemanagement nur schlecht oder gar nicht vorhanden ist, wollen wir dieses für den Benutzer sinnvoll umzusetzen.

## Systemarchitektur

Die Architektur der Website „Scrumiverse“ ist in Frontend und Backend aufgeteilt. Die Basis des Backends bildet Java, für das Controlling kommt das SpringMVC Framework, für die Persistenz das Hibernate ORM-Framework zum Einsatz.  
SpringMVC arbeitet mit einem Dispatcher Servlet, welches Serveranfragen an einen passenden Controller weiterleitet. Dieser mappt ein Model welches dann an die View weitergegeben wird.

Hibernate ist ein O/R-Framework, d.h. es verbindet das objektorientierte Modell von Java mit dem relationalen Modell von klassischen Datenbanken.  
In der Entwicklung des Frontends werden hauptsächlich JQuery, JSTL, JavaScript und Bootstrap verwendet, auf welche hier jedoch nicht weiter eingegangen wird.  
In der Persistenzschicht existieren Interfaces für User (UserDAO), Rollen (RoleDAO), Projekte (ProjektDAO), Sprints (SprintDAO) und User Stories (UserStoryDAO). Zusätzlich dazu gibt es Taskobjekte.

Ähnlich zu den DAOs in der Persistenzschicht arbeitet das Backend durch SpringMVC auch mit passenden Controllern: User-Controller, Rollen-Controller, Projekt-Controller, Sprint-Controller und User Story-Controller.

## Datenmodell

Siehe Datenmodell

## Vor- und Nachteile