



# PROJEKTPLAN

## SCRUMIVERSE



*“The whole universe of scrum projects”*

*Lasse Jacobs  
Joshua Ward  
Kevin Jolitz*

*Toni Serfling  
Kevin Wessler*



# Projektplan

Rel. Userstory ID: US010

Version: V002

## INHALTSVERZEICHNIS

<b>1</b>	<b>VERSIONSGESCHICHTE</b>	<b>2</b>
<b>2</b>	<b>PROJEKTBESCHREIBUNG SCRUM MANAGER</b>	<b>2</b>
<b>3</b>	<b>BEGRÜNDUNG DER PROJEKTWAHL</b>	<b>3</b>
<b>4</b>	<b>IST-ANALYSE „VORGEHENSMODELL SCRUM“</b>	<b>3</b>
4.1	ERKLÄRUNG „VORGEHENSMODELL SCRUM“	3
4.2	DARSTELLUNG DES ARBEITSPROZESSES	3
4.3	UMSETZUNGEN DES ARBEITSPROZESSES	4
4.4	PROBLEME DER UMSETZUNGEN	5
4.5	FAZIT	5
<b>5</b>	<b>SOLL-KONZEPT: VORGEHENSMODELL</b>	<b>6</b>
5.1	GENERELLES	6
5.2	PROJEKTSTRUKTURPLAN	6
5.3	ZEITPLANUNG	7
<b>6</b>	<b>SOLL-KONZEPT: LÖSUNGSANSATZ</b>	<b>7</b>
6.1	GENERELLES	7
6.2	SYSTEMARCHITEKTUR	8
6.3	DATENMODELL	8
<b>7</b>	<b>ANHANG</b>	<b>9</b>
7.1	PROJEKTSTRUKTURPLAN	9
7.2	GANTT DIAGRAMME	11
7.3	DATENMODELL	15
7.4	SYSTEMARCHITEKTUR	18
7.5	PAPIERPROTOTYP	19

## 1 VERSIONSGESCHICHTE

Version	Datum	Änderungsbeschreibung	Bearbeiter
001	03.02.16	Dokumenterstellung, Zusammenfassung	Jolitz, K.;
002	04.02.16	Bearbeitung einiger Kapitel	Ward, J.;
003	05.02.16	Finalisierung	Jolitz, K.;

## 2 PROJEKTBSCHREIBUNG SCRUM MANAGER

Ziel dieses Projektes ist die Entwicklung einer „Scrum-Projektmanagement-Software“. Die Website-Anwendung soll die onlinebasierte Zusammenarbeit von Projektteams mit dem Vorgehensmodell „Scrum“ ermöglichen und fördern.

Durch den Auftraggeber sind folgende Technologien vorgegeben:

- **Entwicklung**
  - Spring MVC, Java
  - JSP / JSTL
  - JavaScript
- **Planung und Realisierung**
  - Webplattform [Scrumdesk](#)
  - Versionsverwaltung mit Git mit einem Repository zur Verfügung gestellt von [GitHub](#)

Um unser Projektziel zu erreichen hat sich das Projektteam auf folgenden Funktionsumfang geeinigt:

- Ein umfangreiches Accountmanagement, welches Rechteverwaltung innerhalb von Projekten ermöglicht.
- Funktionen zur Unterstützung des Projektmanagements
  - Detailgetreue Abbildung des Scrum-Modells
- Eine angemessene grafische Aufbereitung, welches jederzeit eine Auswertung des aktuellen Projektstandes ermöglicht.
  - Dashboard mit Ereignisreport
- Echtzeit Informationsverteilung durch E-Mail notifier und einem Nachrichtensystem zwischen Projektteilnehmern.

Folgende Elemente werden eventuell realisiert:

- Rating System
- Kapazitätenplanung der Benutzer durch den Projektleiter.
- Anbindung an externe Versionsverwaltung Software wie beispielsweise Github oder andere Git Repositories.

Der zeitliche Rahmen für die Umsetzung beginnt am 04.01.2016 0:00 und endet am 25.04.2016 um 23:59 Uhr, mit der Abgabe der folgenden Elemente:

- Kommentierter und archivierter (zip) Quellcode der Webplattform
- Projektdokumentation
- Projektpräsentation

### 3 BEGRÜNDUNG DER PROJEKTWAHL

Moderne Softwareentwicklung wird immer häufiger auf Basis von agilen Vorgehensmodellen durchgeführt. Für diese agilen Ansätze existieren auch schon verschiedene Softwareprojekte, welche die Arbeit unterstützen. Jedoch kosten diese viel Geld oder weisen einige Mängel in der Bedienung oder Rechteverwaltung auf (Siehe Ist-Analyse).

Durch eigene gute Erfahrungen im Team mit dem agilen Ansatz „Scrum“ besteht im Team ein hohes Interesse, eine gute Plattform für private oder eventuell sogar berufliche Projekte zu planen und umzusetzen.

### 4 IST-ANALYSE „VORGEHENSMODELL SCRUM“

Im Folgenden wird das Vorgehensmodell Scrum näher betrachtet. Nach einer kurzen Erklärung wird die Darstellung des Arbeitsprozesses und dessen Umsetzung beleuchtet, sowie die Probleme der bereits bestehenden Umsetzungen. Abschließend wird daraus ein Fazit gezogen, in dem auf Verbesserungsmöglichkeiten eingegangen wird.

#### 4.1 ERKLÄRUNG „VORGEHENSMODELL SCRUM“

Scrum ist ein Vorgehensmodell im Bereich Projektmanagement, welches ursprünglich in der Softwareentwicklung entwickelt und dort zuerst eingesetzt wurde. Es ist ein agiler Ansatz der auf iterativer Entwicklung aufbaut und den Kunden (bei Scrum Product Owner) aktiv in die Entwicklung mit einbezieht.

#### 4.2 DARSTELLUNG DES ARBEITSPROZESSES

Scrum besteht aus einem zyklischen Aufbau um Softwareentwicklung zu unterstützen, sogenannten Sprints.

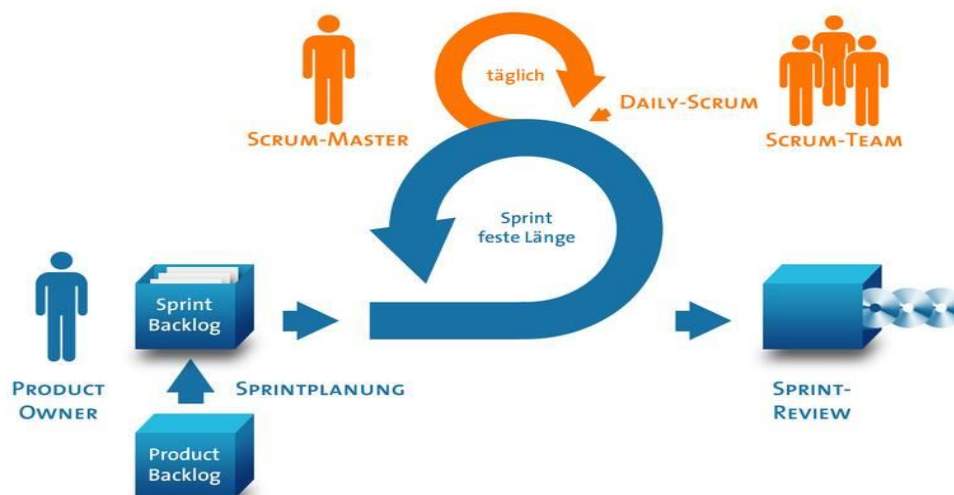


Abbildung 1: Scrum-Arbeitsprozess  
<https://www.3m5.de/scrum/>

Ein Sprint ist ein zeitlicher Rahmen (meist 2-4 Wochen) in welchem eine neue Iteration des Produkts entwickelt wird.

Während des Sprints gibt es tägliche Meetings (Daily Scrum), bei denen der Scrum Master und das Team ihren Fortschritt, Probleme und Erfolge kurz erläutern. Am Ende eines Sprints wird dem Product Owner diese Iteration vorgestellt und mit diesem der



nächste Sprint geplant. Damit wird der Kunde, wie oben bereits erwähnt, aktiv mit in den Entwicklungsprozess einbezogen. Dies erlaubt eine im Vergleich zu nicht-agilen Vorgehensmodellen sehr flexible Entwicklung, in der schnell auf Änderungswünsche reagiert werden kann.

Außerdem wird am Ende jedes Sprints eine Sprint-Retrospektive durchgeführt, in dem alle Beteiligten des Projekts Feedback zum vergangenen Sprint geben können.

Dokumentiert wird Scrum mithilfe von zwei zentralen Dokumenten: Der Produkt-Backlog (ähnlich dem Lasten-/Pflichtenheft aus traditionellen Vorgehensmodellen) bestimmt die Spezifikationen des fertigen Produkts, ist jedoch jederzeit anpassbar und zu verbessern. Der Sprint-Backlog nimmt einzelne Teile dieses Product-Backlogs und erstellt Arbeitspakete für den nächsten Sprint, welche wiederum in kleinere Tasks aufgeteilt werden können, welche zum Beispiel in Form von User Stories formuliert werden können.

Eine User Story ist ein Anwendungsfall des Produkts, quasi eine Antwort auf die Frage „Was möchte der User mit dem Produkt machen können?“. Ebenfalls besitzt Scrum drei Schlüsselrollen im Entwicklungsprozess, zunächst den bereits erwähnten Product Owner, welcher der Auftraggeber des Produkts ist, Backlog-Items priorisiert und für den wirtschaftlichen Erfolg ebendieses Produktes zuständig ist.

Der Scrum Master ist der Leiter des Scrums und dafür verantwortlich dass der Scrum gelingt. Er arbeitet mit dem Entwicklungsteam zusammen, welches für die Lieferung der im Product-/ Sprint Backlog aufgelisteten Features zuständig ist, gehört aber meist nicht dazu.

#### 4.3 UMSETZUNGEN DES ARBEITSPROZESSES

In heutigen Arbeitsumgebungen existieren viele Ansätze Scrum als Projektplanungstool einzusetzen. Dabei lässt sich zwischen dem „Offline Ansatz“ und dem „Online-Ansatz“ unterscheiden. Der Offline Ansatz besteht in seiner Urform aus Karteikarten, welche die User Stories repräsentieren. Diese werden entsprechend auf einem Whiteboard oder einer der Bürowände angeordnet, je nach Zuständigkeit und aktiven Sprint. Zumeist wird dieser Ansatz auch durch digitale Dokumente gestützt, ist aber eher in der eben beschriebenen Papierform zu finden, um dynamisch Dinge verschieben zu können und die Kommunikation und Zusammenarbeit innerhalb eines Büros zu fördern.

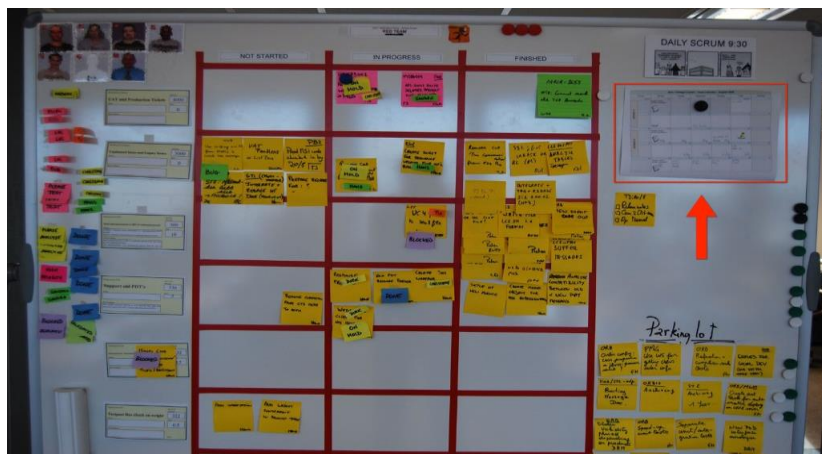


Abbildung 2 Bsp.: Offline-Scrumboard

[http://www.xqa.com.ar/visualmanagement/wp-content/gallery/general-pictures/board\\_with\\_calendar.jpg](http://www.xqa.com.ar/visualmanagement/wp-content/gallery/general-pictures/board_with_calendar.jpg)

Die andere Art der Scrum Projektplanung sind digitale Webplattformen, welche durch ihren Aufbau und Workflow versuchen die Grundprinzipien von Scrum umzusetzen und den haptischen und agilen Ansatz der Papier- und Stiftvariante zu imitieren.

Zusätzlich bieten sie weitere Funktionen, wie z.B. die Möglichkeit mehrere Projekte gleichzeitig zu verwalten, Ressourcen einzuteilen oder wichtige grafische Elemente von Scrum (z.B. BurnDown-Chart) automatisch generieren zu lassen.

Auch die Kommunikation wird versucht zu übertragen und bei größeren Projektgruppen zu vereinfachen. Dazu existieren verschiedene Kommentarfunktionen oder Nachrichtensysteme auf den Plattformen.

#### 4.4 PROBLEME DER UMSETZUNGEN

Trotz der vielen unterstützenden Funktionen der Webanwendungen gibt es viele Probleme, die wir durch die Analyse von Konkurrenzprodukten gefunden haben (für detaillierte Informationen siehe 2\_Konkurrenz\_Analyse.pdf). Allen voran gibt es bei vielen der Plattformen eine große Einstiegshürde durch preisliche Barrieren, die sich meist anhand der Elemente, der Teilnehmeranzahl oder nach einem festen Abo Modell berechnen. Sollte man sich dann für eine der Plattformen entschieden haben, gibt es dann meist größere Mängel was die Übersicht über Funktionen und Informationen angeht. Intuitiv sind dadurch die Oberflächen selten und der Scrum Arbeitsprozess wird entweder abgeändert oder schlichtweg mangelhaft umgesetzt.

Dazu kommt dann ein meist starres oder schlichtweg kaputtes Rechtemanagement. Beispielsweise auf der Plattform ScrumDesk, an denen Benutzer User Stories und Tasks selbständig erstellen, löschen und ändern können. Ändern kann man dies nicht, Teilnehmer können alles oder nichts.

Wenn die Rechte nicht schon zu Chaos führen, könnte dies durch die teilweise sehr langsame Synchronisierung kommen. Auf vielen der Plattformen ist es unklar, wann und wie eine Synchronisierung erfolgt und zudem dauert es häufig zu lange.

Der letzte Punkt betrifft die Datensicherheit und die Zusammenarbeit mit anderen Systemen wie Git oder SVN zur Versionskontrolle. Datensicherheit ist meist nicht gegeben, da keines der von uns analysierten Tools eine Export Funktion bereitstellte. Dadurch muss man sich auf die Sicherungen des Anbieters verlassen. Ebenso ist die Anbindung an andere Systeme nur selten möglich und wenn, nur in den kostenpflichtigen Anwendungen.

Viele dieser Nachteile hat die klassische offline Variante nicht. Dafür ist ihr gravierendster Nachteil die kaum koordinierbare Arbeit mit vielen Teilnehmern. Wenn die Leute nicht im gemeinsamen Großraumbüro sitzen wird es enorm schwierig zu kommunizieren und aktuelle Fotos oder Office Dokumente mit dem aktuellen Sprint umher zu schicken und zwischen den Teilnehmern zu synchronisieren. Dazu kann es zu einem Tunnelblick bei der Papiervariante kommen, da man häufig den Fokus auf einen Sprint legt und so das große ganze eventuell im Papierchaos aus den Augen verliert.

#### 4.5 FAZIT

Aus diesen von uns erarbeiteten Mängeln lassen sich folgende Kernprobleme ableiten, die wir verbessern und überarbeiten wollen:

- Kostenloses Geschäftsmodell anbieten
- Funktionen und Informationen übersichtlich darstellen, auch für mehrere Projekte.

- Saubere und intuitive Umsetzung des Scrum Arbeitsprozesses
  - Versuchen den klassischen/haptischen Ansatz in die digitale Welt zu übertragen
- Konfigurierbares und detailliertes Rechtemanagement
- Den Benutzern vielfältige Kommunikations- und Kommentarfunktionen bieten
- Versuchen die Synchronisationszeit zu verringern.

## 5 SOLL-KONZEPT: VORGEHENSMODELL

### 5.1 GENERELLES

In der Gruppe haben wir uns dazu entschieden das agile Vorgehensmodell Scrum zu verwenden. Jedoch viele Elemente von Scrum sind nicht vollständig benutzbar aufgrund der Projektvorgaben, daher benutzen wir eine abgewandelte Form. 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 zusätzliche Abweichung betrifft die Abhängigkeit zwischen den User Stories, welche durchaus gegeben ist bei unserem Projekt.

Eine weitere Abweichung betrifft die Flexibilität. Durch die externen Rahmenbedingungen müssen Sprints in unserem Projekt verschiedene Längen haben und die Abgabe von User Stories ist streng zeitlich vorgegeben und ist weitaus weniger flexibel als im Standard Scrum. Die Kommunikationsinstrumente von Scrum verwenden wir gar nicht oder nur in leicht abgewandelter Form. Aus dem Daily Scrum haben wir ein weekly Scrum gemacht, welches wir am Ende der Woche in Textform niederschreiben. Aus diesem Dokument und der allgemeinen Zeitverschreibung wird dann ein Ablauf und Fazit für das Projekt ermittelt.

Auf Sprint Reviews verzichten wir, genauso wie die Vorgabe dass Sprints feste Iterationen des Produkts ergeben müssen. In den Sprints ergeben sich nicht immer Prototypen der Software, sondern fest definierte Ergebnisse des Gesamtprojektes (Projektdokumentation, Prototyp, Abgabefassung, Präsentation).

Regelungen, die Toolsbaseline und die Tagging Richtlinien für unsere Commits auf der Seite GitHub, sind in einem externen Dokument festgehalten, dem sogenannten „Teilnehmer Handbuch“ (**siehe xyz**).

### 5.2 PROJEKTSTRUKTURPLAN

In unserem Projektstrukturplan haben wir uns auf die Epics, welche wir in „Scrumdesk“ festgelegt haben, beschränkt. Diese beschreiben die Features, welche wir in unserem Projekt umsetzen wollen. Der Grund für die Beschränkung wäre ein Verlust der Übersicht, durch die hohe Anzahl von User Stories.

Im Projektstrukturplan werden wir eine Aufteilung nach den verschiedenen Sprintphasen vornehmen. Diese Phasen werden in der Umsetzung gestaffelt (**siehe Fehler! Verweisquelle konnte nicht gefunden werden.: AStruktur 2**). So wird im ersten Sprint die Projektplanung und die Planung eines Prototypens vorgenommen (**siehe Fehler! Verweisquelle konnte nicht gefunden werden.: AStruktur 3**). In der zweiten Sprintphase wird die Implementierung der Kernfunktionen der Software vorgenommen (**siehe Fehler! Verweisquelle konnte nicht gefunden werden.: AStruktur 4**). Die dritte Sprintphase wird zur Finalisierung und Erweiterung der Funktionen/des Funktionsumfanges genutzt (**siehe Fehler! Verweisquelle konnte nicht gefunden werden.:**



**AStruktur 5).** Im dem Sprint wird die Entwicklung abgeschlossen. Der letzte Sprint wird sich mit der Erstellung der Präsentation und Dokumentation beschäftigen (*siehe Fehler! Verweisquelle konnte nicht gefunden werden.: AStruktur 6*). Mit diesem Sprint wird das Projekt „Scrumiverse“ beendet.

### 5.3 ZEITPLANUNG

Die im Projektstrukturplan erwähnten Sprintphasen haben verschiedene Prioritäten und Längen. Da die Planung eine sehr hohe Priorität in unserem Projekt hat, wurde der erste Sprint mit 5 Wochen etwas länger angesetzt (siehe **Gantt Diagramme: AGantt1 – Sprint 1**).

Für den zweiten Sprint haben wir wie im klassischen Scrum haben hierfür 4 Wochen bzw. 1 Monat vorgesehen (*siehe Gantt Diagramme: AGantt2 – Sprint 2*).

Im dritten Sprint haben wir für die ersten drei Wochen für die Finalisierung der Webseite geplant, in dem wir nach dem zweiten Sprint noch fehlende Features implementieren. Der Meilenstein des dritten Sprint ist erst nach 4 Wochen erreicht, sodass wir noch 1 Woche Zeit haben um fehlerhafte Teile der Software zu beheben, noch unfertige User-Stories zu bearbeiten oder weitere Features hinzuzufügen (siehe **Gantt Diagramme: AGantt3 – Sprint3**).

Nachdem wir im zweiten und dritten Sprint die Implementation abgeschlossen haben, werden wir im vierten Sprint die erste Woche nutzen um die erwähnte Präsentation zu erstellen und die Dokumentation aus Sprint 1 zu finalisieren. Ebenso wie in Sprint 3, ist der Meilenstein ein paar Tage nach Ende des geplanten Sprints angesetzt, um Verzögerungen oder Unerledigtes zu kompensieren (siehe **Gantt Diagramme: AGantt4 – Sprint 4**).

Nachdem die Sprints erstellt wurden, hat sich das Team zusammengesetzt, User-Stories gemeinsam erarbeitet und anschließend auf Sprint 2 und 3 aufgeteilt. Im Zuge dessen waren die Teammitglieder, nachdem das Team in Backend und Frontend eingeteilt wurde, selbstständig dafür verantwortlich sich User-Stories zuzuweisen und in dieser Tasks zu erstellen. Dabei wurde auch auf die Zeiteinteilung eines Teammitglieds während der Sprints geachtet, da zu den Task auch die Zeit geschätzt werden musste.

## 6 SOLL-KONZEPT: LÖSUNGSANSATZ

### 6.1 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. Um mit den bereits existierenden Plattformen zu konkurrieren werden wir auch Funktionen adaptieren, welche wir für unsere Plattform als sinnvoll erachten.

Das Vorgehensmodell von Scrum soll besser und originalgetreu umgesetzt werden. Hierfür wollen wir die haptische Umsetzung in eine Online-Version bündeln und zentral erreichbar machen. Auf Konkurrenz-Plattformen ist uns schnell aufgefallen, dass ein Rechtemanagement nur schlecht oder gar nicht vorhanden ist. Dies wollen wir für unsere Benutzer sinnvoller umzusetzen.

Ein weiterer Aspekt der originalgetreuen Umsetzung betrifft das Weglassen einiger zusätzlicher Funktionen, wie das Verwalten von Releases und Kanban. Aufgrund der



zeitlichen Knappheit werden auch wenige Kernfunktionen nicht umgesetzt, wie z.B. eine Retrospektive.

## 6.2 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. Die Systemarchitektur ist durch SpringMVC streng vorgegeben. Außerdem wird durch das MVC pattern das Backend klar von dem Frontend getrennt.

Hibernate ist ein O/R-Framework, d.h. es verbindet das objektorientierte Modell von Java mit dem relationalen Modell von klassischen Datenbanken. Hibernate erspart das Schreiben eigener Datenbankzugriffe und somit auch wertvolle Zeit.

In der Entwicklung des Frontends werden hauptsächlich JQuery, JSTL, JavaScript, Bootstrap, Ajax und Highchart verwendet, auf welche hier jedoch nicht weiter eingegangen wird. Durch JSTL wird viel Glue Code in der View Ebene gespart. Zur Vereinfachung und Standardisierung der Benutzeroberfläche verwenden wir Bootstrap. Des Weiteren liefert Bootstrap auch neue Kontrollelemente. Highchart erleichtert die Darstellung von Diagrammen.

Da Bootstrap und Highchart teilweise auf JavaScript basiert, werden dessen Elemente auf älteren Browsern nicht Unterstützt.

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.

Für eine graphische Abbildung der Systemarchitektur: siehe **Systemarchitektur, ASystem 1**

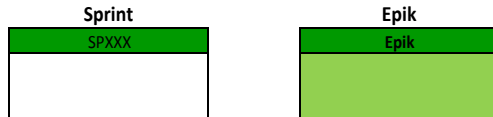
## 6.3 DATENMODELL

Bei der Erstellung des Datenmodells haben wir uns stark an dem Basis-Vorgehensmodell von Scrum orientiert (siehe **Datenmodell: Business Domain**). In Anbetracht unserer Umsetzung haben wir dann noch zusätzlich Website relevante Datenmodelle abgebildet (siehe **Datenmodell: Message und Ereignis Modelle** und **Datenmodell: Benutzerdaten**).

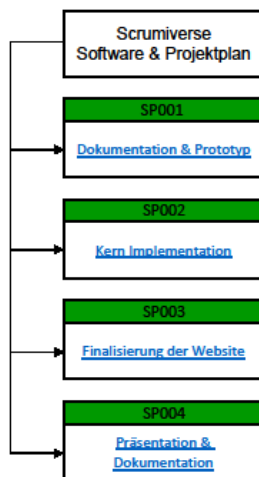
## 7 ANHANG

### 7.1 PROJEKTSTRUKTURPLAN

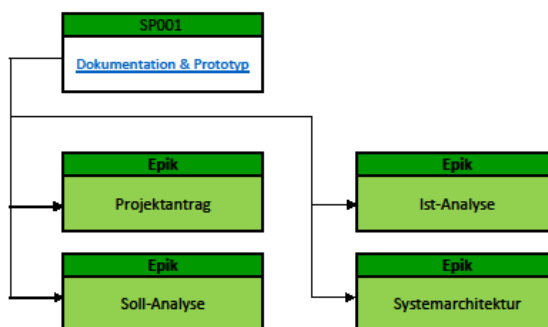
#### Erklärung



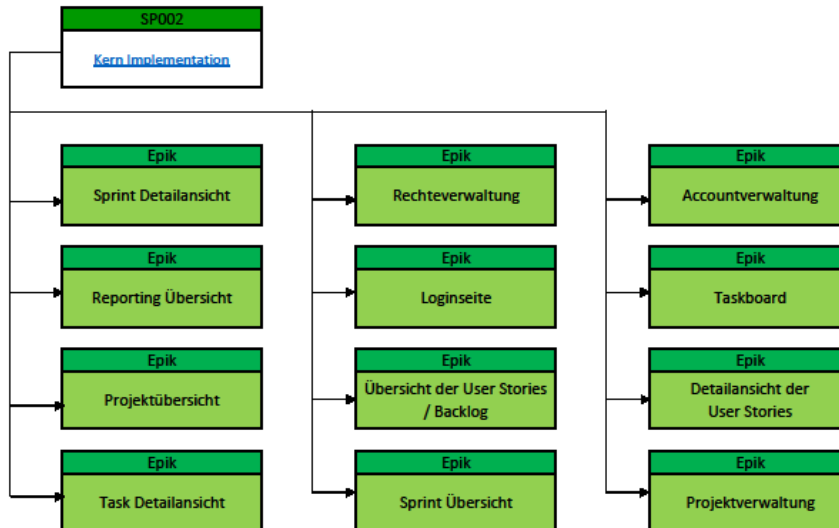
AStruktur 1 – Erklärung



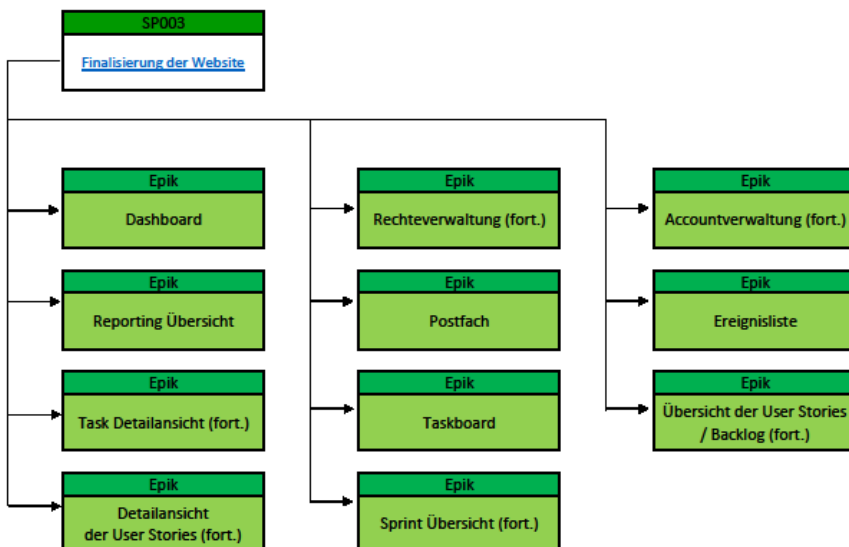
AStruktur 2 - Sprint Übersicht



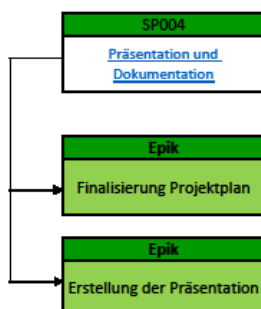
AStruktur 3 - Sprint 1



AStruktur 4 - Sprint 2



AStruktur 5 - Sprint 3

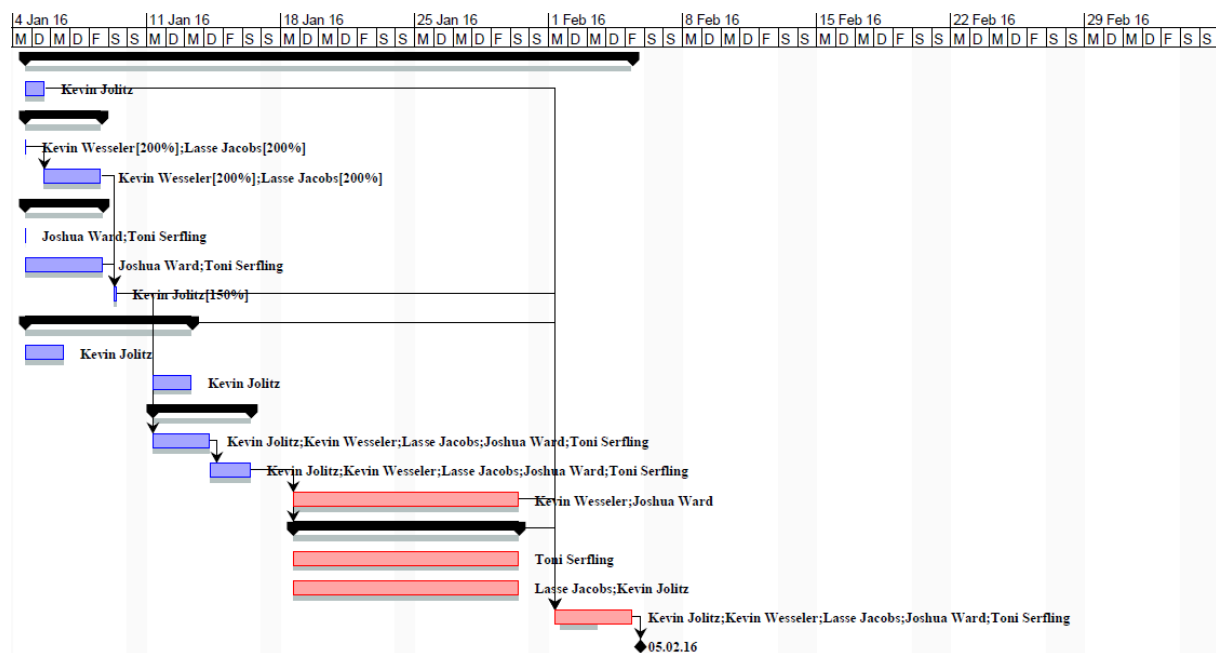


AStruktur 6 - Sprint 4



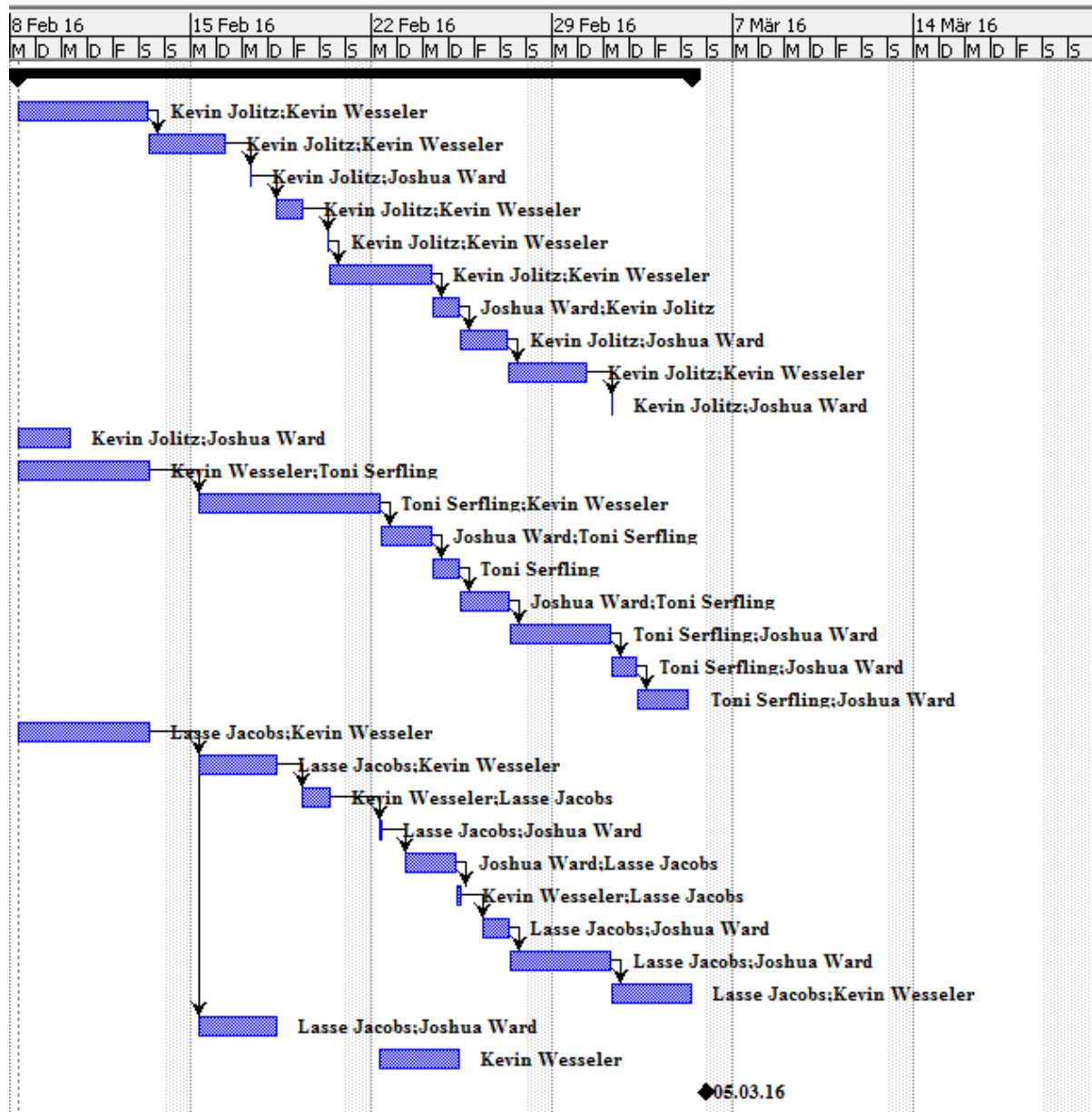
## 7.2 GANTT DIAGRAMME

		Name	Dauer	Aufwand	Start	Ende
1		SP001 Papierprototyp & Projektplan		249 Stunden	04.01.16 16:00	05.02.16 09:30
2		US001 Projektantrag ausformulieren	2 Tage	2 Stunden	04.01.16 16:00	05.01.16 17:00
3		US002 Analyse von Konkurrenzprodukten	5 Tage	18 Stunden	04.01.16 16:00	08.01.16 16:30
4		T001 Ermitteln der Konkurrenzsoftware	2 Tage	4 Stunden	04.01.16 16:00	04.01.16 17:00
5		T002 Stärken und Schwächenanalyse der Konkurrenz	3 Tage	14 Stunden	05.01.16 16:00	08.01.16 16:30
6		US003 Analyse des Scrumvorgangsmodells	5 Tage	12 Stunden	04.01.16 16:00	08.01.16 17:00
7		T003 Interviews führen	1 Tag	2 Stunden	04.01.16 16:00	04.01.16 17:00
8		T004 Analyse und Darstellung des Workflows	4 Tage	10 Stunden	04.01.16 16:00	08.01.16 17:00
9		US004 Schreiben des Kapitels "Ist Analyse"	1 Tag	4 Stunden	09.01.16 08:00	09.01.16 10:40
10		US005 Formulieren des gewählten Vorgangsmodells	6 Tage	8 Stunden	04.01.16 16:00	13.01.16 09:00
11		T005 Erfassen der Toolsbaseline	3 Tage	3 Stunden	04.01.16 16:00	06.01.16 17:00
12		T006 Formatvereinbarungen ausarbeiten / vereinbaren	2 Tage	5 Stunden	11.01.16 08:00	13.01.16 09:00
13		US006 Programmrelevante User Stories ermitteln	6 Tage	75 Stunden	11.01.16 08:00	16.01.16 10:30
14		T007 Ermitteln der User Stories	3 Tage	35 Stunden	11.01.16 08:00	14.01.16 08:30
15		T008 Spezifizieren der User Stories	3 Tage	40 Stunden	14.01.16 08:30	16.01.16 10:30
16		US007 Papierprototyp erstellen	12 Tage	32 Stunden	18.01.16 16:00	30.01.16 11:00
17		US008 Architektur erstellen	12 Tage	48 Stunden	18.01.16 16:00	30.01.16 11:00
18		T009 Erstellen des Datenmodells	12 Tage	16 Stunden	18.01.16 16:00	30.01.16 11:00
19		T010 Erstellen des Systementwurfs	12 Tage	32 Stunden	18.01.16 16:00	30.01.16 11:00
20		US009 Zusammenführen der Ergebnisse ggf. Korrektur	6 Tage	50 Stunden	01.02.16 08:00	05.02.16 09:30
21		Abgabe Projektplan		0 Stunden	05.02.16 18:30	05.02.16 09:30



AGantt 1 - Sprint 1

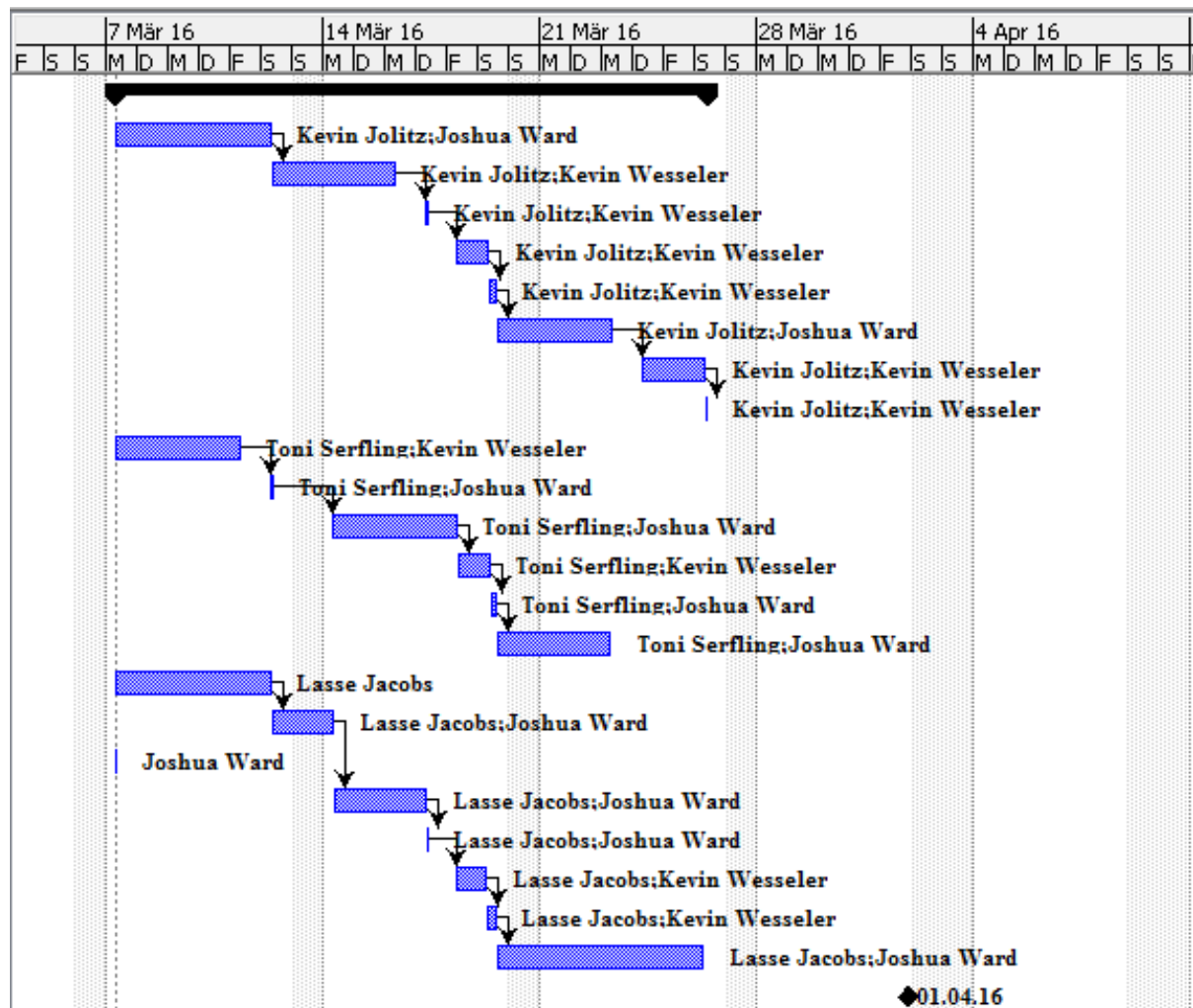
Name	Dauer	Aufwand	Start	Ende
<b>SP002 Kern Implementation</b>		<b>164,5 Stun...</b>	<b>08.02.16 08:00</b>	<b>05.03.16 10:30</b>
US010 Der Benutzer will ein Account erstellen können.	5 Tage	10 Stunden	08.02.16 08:00	13.02.16 10:00
US014 Der Benutzer möchte sich einloggen können.	3 Tage	3,5 Stunden	13.02.16 10:00	16.02.16 09:00
US018 Der Projektadmin will Rollen verteilen können	1 Tag	2 Stunden	17.02.16 08:00	17.02.16 09:00
US025 Der Product Owner will eine Übersicht über alle erstellten Sprints	2 Tage	3 Stunden	18.02.16 08:00	19.02.16 09:00
US028 Der Product Owner will alle zu einem Sprint hinzugefügten User Stories anzeigen lassen.	1 Tag	3,5 Stunden	20.02.16 08:00	20.02.16 10:00
US029 Der Projektteilnehmer will eine neue Task erstellen können.	5 Tage	6,5 Stunden	20.02.16 10:00	24.02.16 09:00
US030 Der Projektteilnehmer möchte detaillierte Information zum Task in einem Popup sehen	2 Tage	4 Stunden	24.02.16 09:00	25.02.16 10:30
US031 Die Projektteilnehmer wollen Tasks verwalten können (RUD)	3 Tage	7 Stunden	25.02.16 10:30	27.02.16 08:30
US032 Der Projektteilnehmer will eine Übersicht seiner Tasks zu einer User Story (Task Board)	5 Tage	8,5 Stunden	27.02.16 08:30	01.03.16 09:00
US033 Der Projektteilnehmer will in der Detailansicht des Tasks diesen in verschiedene Phasen (TO DO, IN PROGRESS, DONE) zuordnen können	1 Tag	2 Stunden	02.03.16 08:00	02.03.16 09:00
US040 Der Projektteilnehmer will einer Task TAGS zu teilen können	3 Tage	4 Stunden	08.02.16 08:00	10.02.16 09:00
US011 Der Benutzer will ein Scrum-Projekt erstellen können	5 Tage	8,5 Stunden	08.02.16 08:00	13.02.16 11:00
US013 Der Product Owner will Sprints erstellen können.	8 Tage	9,5 Stunden	15.02.16 08:00	22.02.16 09:00
US016 Der Projektadmin will andere Benutzer einladen können	3 Tage	7 Stunden	22.02.16 09:00	24.02.16 09:00
US017 Der Projektadmin will vordefinierte Rollen benutzen können.	2 Tage	3 Stunden	24.02.16 09:00	25.02.16 10:30
US019 Der Benutzer will ein Projekt auswählen können und anschließend auf das Dashboard weitergeleitet werden	3 Tage	6 Stunden	25.02.16 10:30	27.02.16 09:30
US034 Der Projektadmin will andere Benutzer ausladen können.	4 Tage	5 Stunden	27.02.16 09:30	02.03.16 08:30
US035 Der Projektadmin will das Projekt umbenennen können.	2 Tage	2 Stunden	02.03.16 08:30	03.03.16 08:30
US036 Der Projektadmin will ein Scrum-Projekt löschen können	3 Tage	3 Stunden	03.03.16 08:30	05.03.16 08:30
US012 Der Product Owner will eine User Story erstellen können.	6 Tage	8,5 Stunden	08.02.16 08:00	13.02.16 11:00
US015 Der Benutzer will eine Übersicht über alle seine zugehörigen Projekte.	4 Tage	6 Stunden	15.02.16 08:00	18.02.16 09:00
US020 Der Projektteilnehmer will eine tabellarische Übersicht über alle User Stories	2 Tage	6 Stunden	19.02.16 08:00	20.02.16 11:00
US021 Der Product Owner will User Stories verwalten können (RUD)	1 Tag	4 Stunden	22.02.16 08:00	22.02.16 11:00
US022 Der Projektteilnehmer möchte detaillierte Information zur User Story in einem Popup sehen	3 Tage	4 Stunden	23.02.16 08:00	25.02.16 08:30
US023 Der Projektteilnehmer will eine Post-It Übersicht über alle User Stories.	1 Tag	5 Stunden	25.02.16 08:30	25.02.16 12:30
US024 Der Product Owner will in der Detailansicht User Stories Sprints zuweisen können.	2 Tage	4 Stunden	26.02.16 08:00	27.02.16 09:30
US026 Der Product Owner will Sprints verwalten können (RUD)	6 Tage	6 Stunden	27.02.16 09:30	02.03.16 08:30
US037 Der Projektteilnehmer möchte ein Burn down/up chart über die verschiedenen Sprints haben.	4 Tage	8 Stunden	02.03.16 08:30	05.03.16 10:30
US039 Der Projektteilnehmer will eine History über alle Änderungen einer User Story	4 Tage	7 Stunden	15.02.16 08:00	18.02.16 09:00
US027 Der Product Owner will eine ausklappbare, tabellarische User Story Übersicht um diese den Sprint zuzuordnen	4 Tage	8 Stunden	22.02.16 08:00	25.02.16 10:30
Sprint Ende		0 Stunden	05.03.16 23:59	05.03.16 11:00



AGantt 2 - Sprint 2

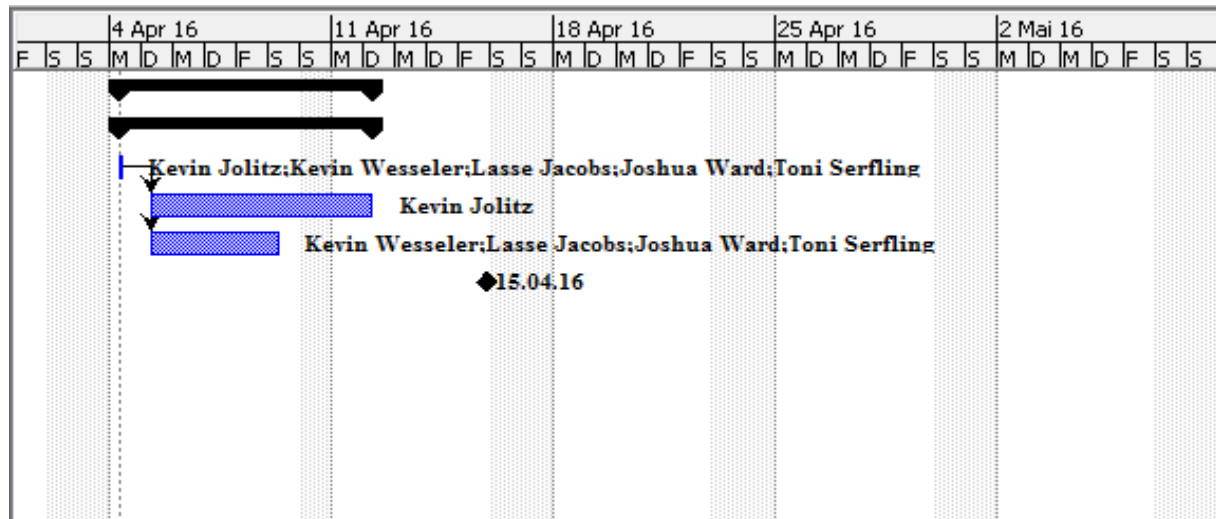


Name	Dauer	Aufwand	Start	Ende
<b>SP003 Finalisierung der Website</b>		129,25 Stund...	<b>07.03.16 08:00</b>	<b>26.03.16 11:00</b>
US038 Der Projektteilnehmer will eine History über alle Changes einer Task	6 Tage	10 Stunden	07.03.16 08:00	12.03.16 09:00
US042 Der Product Owner will Kategorien erstellen und editieren können (Farbe)	5 Tage	12 Stunden	12.03.16 09:00	16.03.16 09:30
US043 Der Product Owner will User Stories in eine (vordefinierte) Kategorie einordnen können.	1 Tag	4,5 Stunden	17.03.16 08:00	17.03.16 11:00
US050 Der Projektadmin will Rollen definieren können.	2 Tage	6,5 Stunden	18.03.16 08:00	19.03.16 10:00
US049 Der Projektadmin will Rechte an selbstdefinierte Rollen verteilen können	1 Tag	6,5 Stunden	19.03.16 10:00	19.03.16 16:00
US057 Der Projektteilnehmer will seine Zeitverschreibung über den Task loggen können. (In welcher Zeit welche Änderung)	5 Tage	4,75 Stunden	19.03.16 16:00	23.03.16 09:00
US059 Die Projektteilnehmer wollen eine Übersicht über die einzelnen Zeiten (Geplant, Verbraucht, Verbleibend) aller Teilnehmer einer Task	3 Tage	4,75 Stunden	24.03.16 08:00	26.03.16 10:00
US060 Die Projektteilnehmer wollen in den Tasks deren zeitlichen Aufwand schätzen können.	1 Tag	1,75 Stunden	26.03.16 10:00	26.03.16 11:00
US041 Der Benutzer möchte Accounteinstellungen vornehmen können	5 Tage	9 Stunden	07.03.16 08:00	11.03.16 09:00
US044 Der Benutzer möchte einstellen können, ob er über Ereignisse per E-Mail informiert wird.	1 Tag	4 Stunden	12.03.16 08:00	12.03.16 11:00
US051 Der Projektteilnehmer möchte eine Nachricht schreiben, lesen und löschen können	5 Tage	13 Stunden	14.03.16 08:00	18.03.16 09:00
US054 Der Projektteilnehmer möchte innerhalb der detaillierten User Story Ansicht dazugehörige Tasks sehen (Auflistung)	2 Tage	3,75 Stunden	18.03.16 09:00	19.03.16 10:30
US055 Der Projektteilnehmer will User Stories filtern können.	1 Tag	4,75 Stunden	19.03.16 10:30	19.03.16 15:30
US058 Der Projektteilnehmer will Tasks filtern können.	5 Tage	4,75 Stunden	19.03.16 15:30	23.03.16 08:30
US045 Der Projektteilnehmer möchte alternativ Ereignisbenachrichtigungen per E-Mail erhalten.	6 Tage	6 Stunden	07.03.16 08:00	12.03.16 09:00
US046 Der Projektteilnehmer möchte über Änderungen (Changes und Kommentare) an seiner User Story / Task informiert werden	3 Tage	5 Stunden	12.03.16 09:00	14.03.16 10:00
US047 Der Scrummaster möchte über Änderung der Aufwandsschätzung einer Task des aktuellen Sprint informiert werden	1 Tag	1 Stunde	07.03.16 08:00	07.03.16 09:00
US048 Der Scrummaster möchte über Änderung einer User Story / Task des aktuellen Sprint informiert werden	4 Tage	6,75 Stunden	14.03.16 10:00	17.03.16 10:00
US052 Der Projektteilnehmer möchte Ereignisse als gelesen markieren können.	1 Tag	2 Stunden	17.03.16 10:00	17.03.16 11:00
US053 Der Projektteilnehmer möchte in einer visuellen Darstellung die Sprints in eine Gegenüberstellung von geplanter/ tatsächlicher Zeit	2 Tage	2,75 Stunden	18.03.16 08:00	19.03.16 08:30
US056 Der Projektteilnehmer möchte per Drag n Drop einen Task in verschiedene Phasen einteilen können.	1 Tag	6,75 Stunden	19.03.16 08:30	19.03.16 15:30
US061 Der Projektteilnehmer will nach Projektauswahl ein Dashboard mit Elementen aus anderen Boards	8 Tage	9 Stunden	19.03.16 15:30	26.03.16 08:30
Sprint Ende		0 Stunden	01.04.16 22:59	01.04.16 09:00

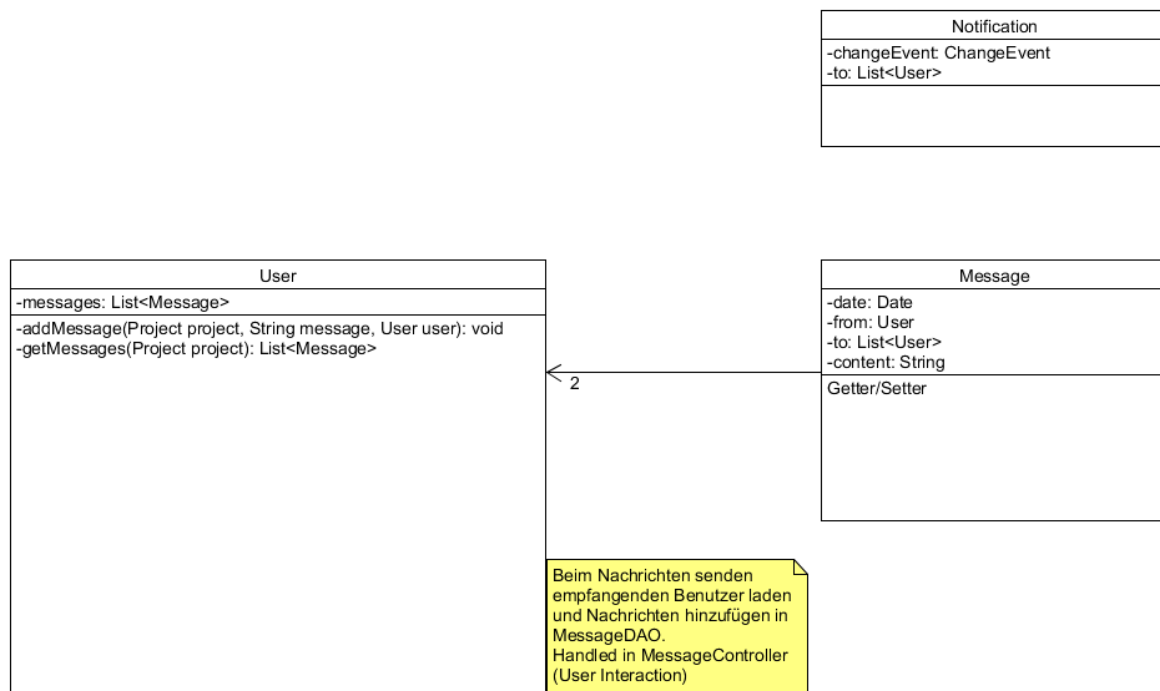


AGantt 3 - Sprint 3

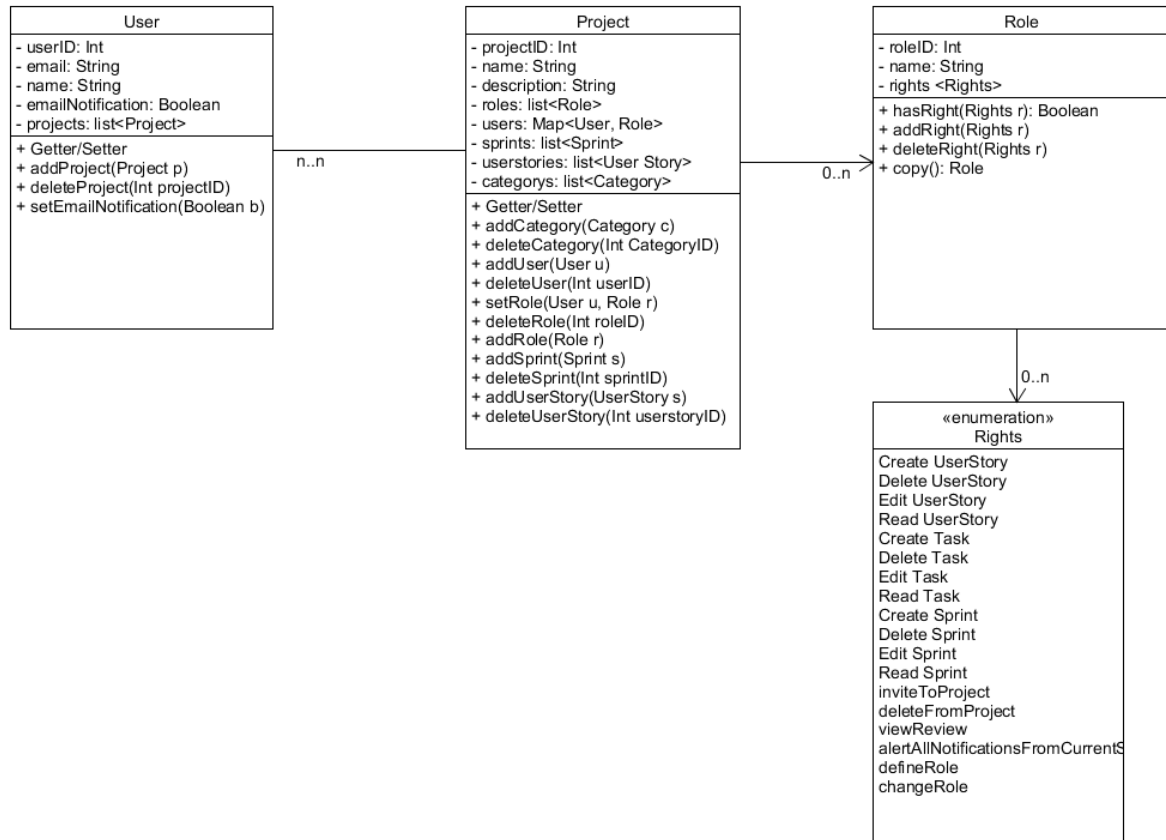
Name	Dauer	Aufwand	Start	Ende
SP004 Präsentation & Dokumentation		68 Stunden	04.04.16 08:00	12.04.16 08:30
US062 Projektdokumentation fertigstellen		28 Stunden	04.04.16 08:00	12.04.16 08:30
Teambesprechung Projektfazit		15 Stunden	04.04.16 08:00	04.04.16 11:00
Projektdokumentation finalisieren		13 Stunden	05.04.16 08:00	12.04.16 08:30
US063 Präsentation erstellen		40 Stunden	05.04.16 08:00	09.04.16 09:30
Sprint Ende		0 Stunden	15.04.16 22:59	15.04.16 09:00



## 7.3 DATENMODELL

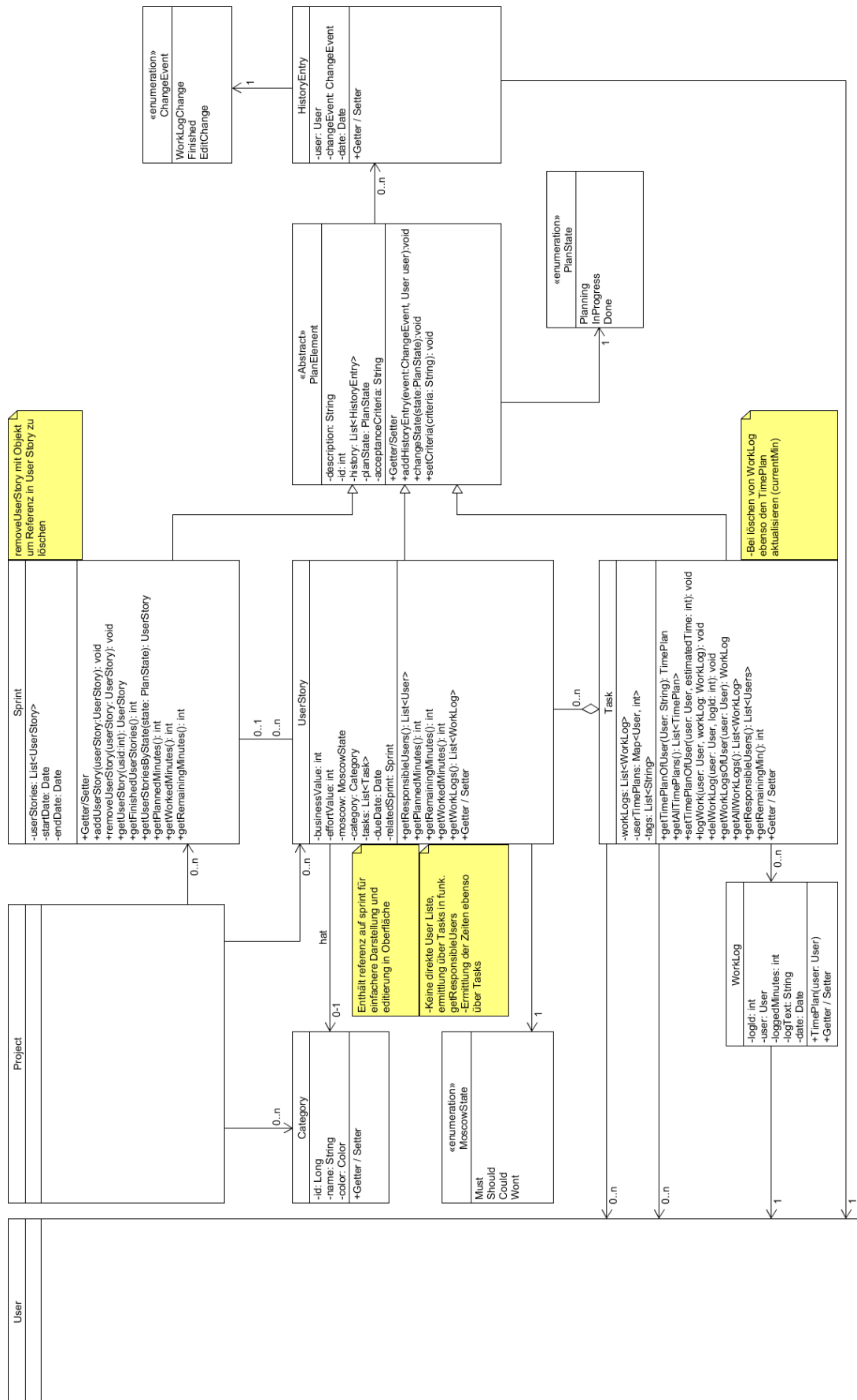


AData 1 - Message und Ereignis Modelle

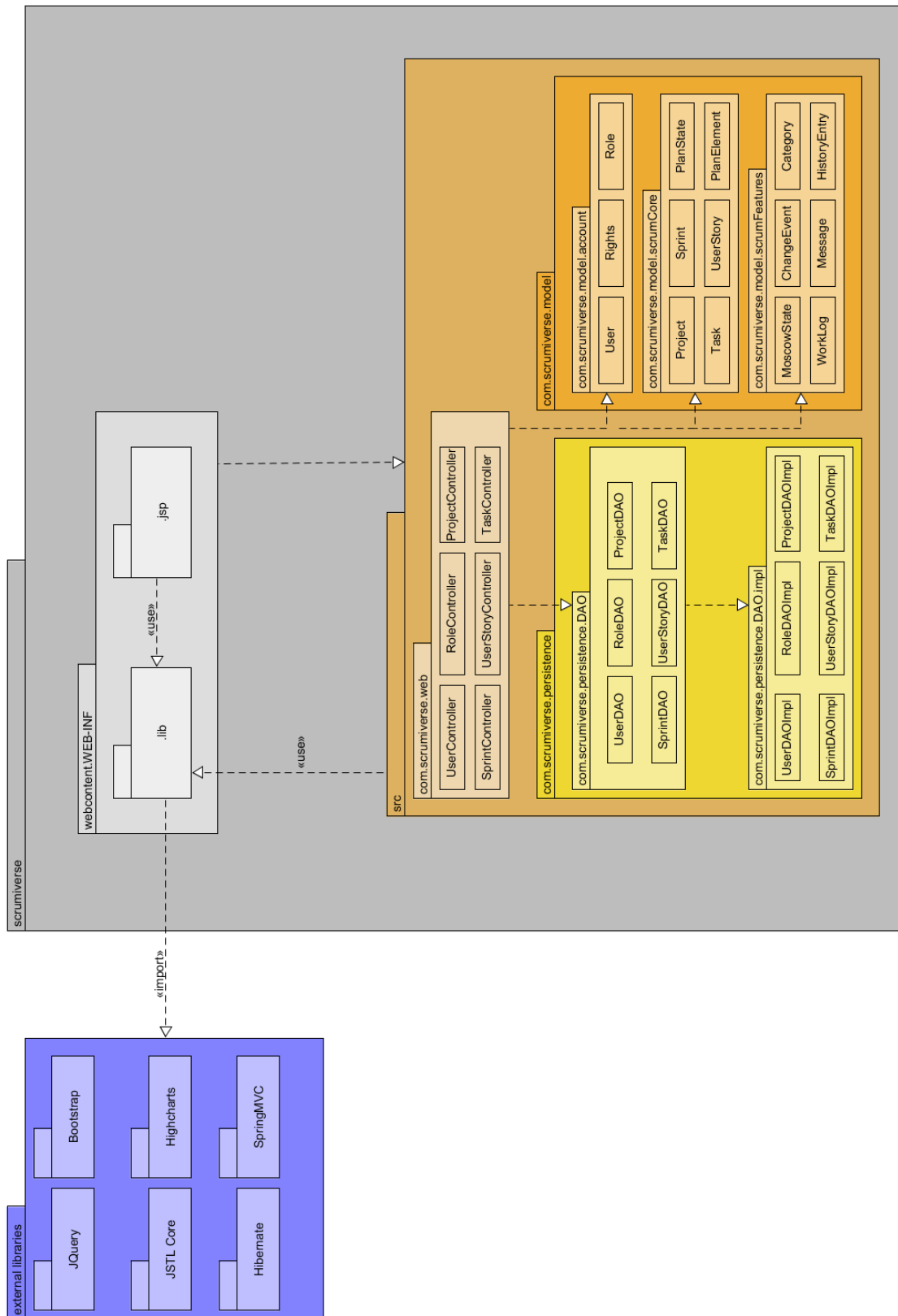


AData 2 - Benutzerdaten





## 7.4 SYSTEMARCHITEKTUR



ASystem 1 - Overview

## 7.5 PAPIERPROTOTYP

ID ## - USERSTORY - HEADER	TIME	MEMBERS	CATEGORY SPRINT	TIME STATS MOSCOW VALUE RISK	EFFORT	
ID 001 - USERSTORY	TIME	6+	CATEGORY SPRINT	## / ## / ## M 100 40	40	
ID 001 - USERSTORY	TIME	6+	CATEGORY SPRINT	## / ## / ## M 100 40	40	⚙
ID 001 - USERSTORY	TIME	6+	CATEGORY SPRINT	## / ## / ## M 100 40	40	>

ID 001 - USERSTORY	TIME	6+	CATEGORY SPRINT	## / ## / ## M 100 40	40	⚙
ID 001 - USERSTORY	TIME	6+	CATEGORY SPRINT	## / ## / ## M 100 40	40	>

SAND UHR BEST ICON EVER - ZEIGT DUE DATE ALS SAND

SAND UHR BEST ICON EVER - ZEIGT DUE DATE ALS SAND

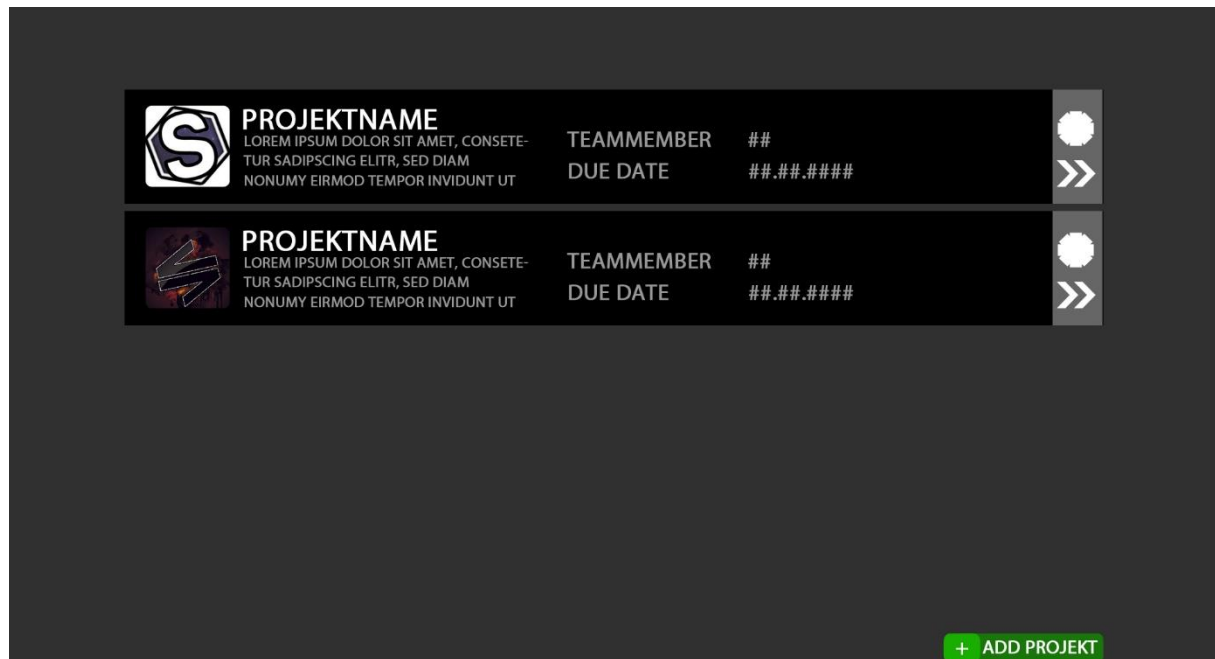
+ ADD STORY ↗

AProto 1 - Userstory Übersicht

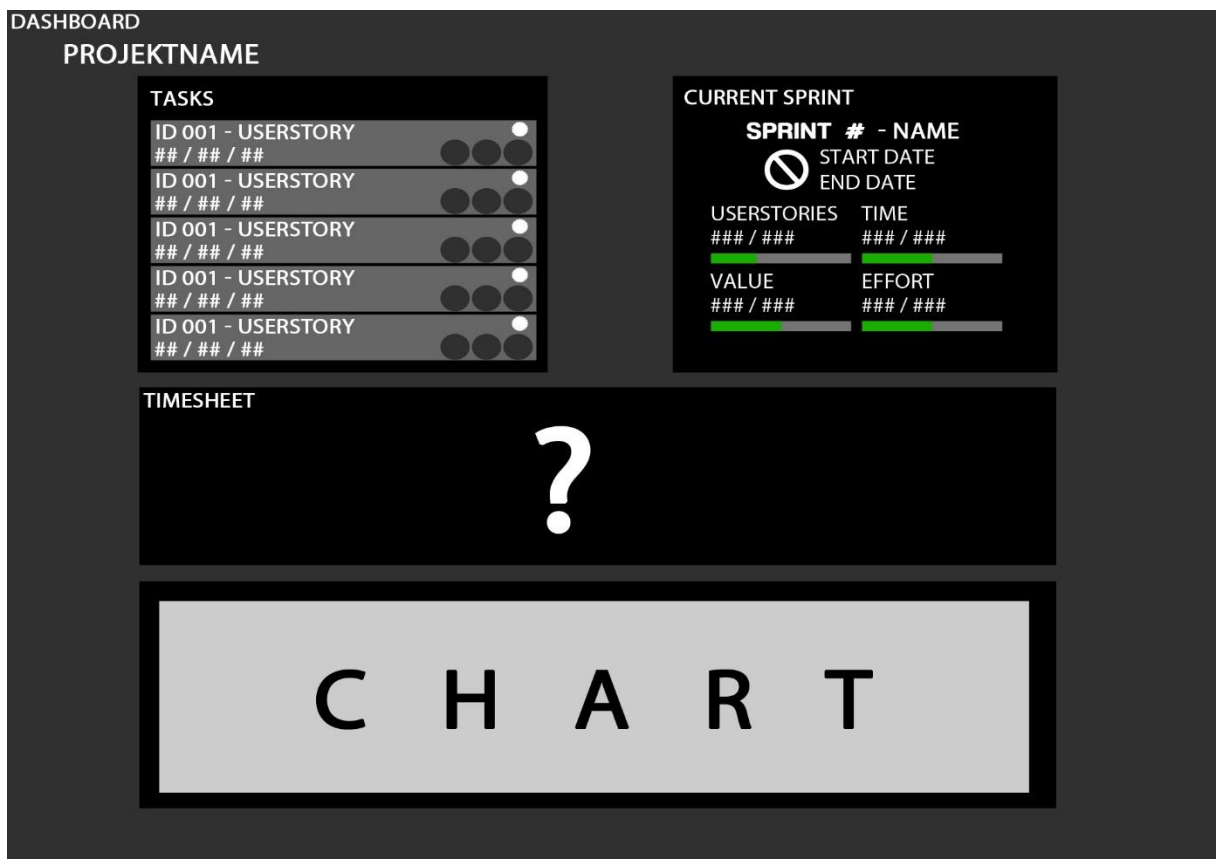


AProto 2 - Menüführung





AProto 3 - Projektwahl



AProto 4 - User Stories

## RECHTEVERWALTUNG

HEADING

MEMBERROLE  
MEMBERROLE  
MEMBERROLE

RECHTENNAME

RECHT ☐

RECHT ☐

RECHT ☐

RECHT ☐

RECHT ☐

RECHT ☐

SAVE

+ ADD ROLE

AProto 5 - Rechteverwaltung

PROJEKTNAME

LOREM IPSUM DOLOR SIT AMET, CONSETETUR SADIPSCING ELITR, SED DIAM NONUMY  
EIRMOD TEMPOR INVIDUNT UT LABORE ET DOLORE MAGNA ALIQUYAM ERAT, SED DIAM  
VOLUPTUA. AT VERO EOS ET ACCUSAM ET

KLICK = FORMULAR

BENUTZERNAME

EVTL DATA: ## ☐

ROLLE

BENUTZERNAME

EVTL DATA: ## ☐

ROLLE

BENUTZERNAME

EVTL DATA: ## ☐

ROLLE

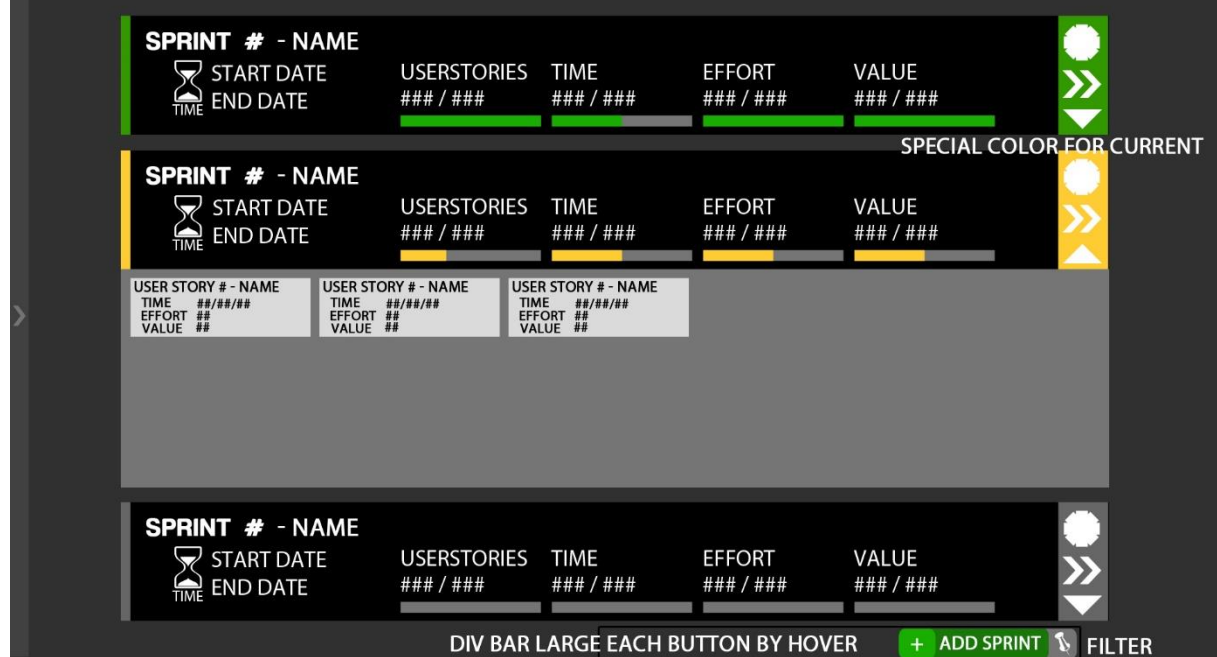
SAVE

+ ADD MEMBER

AProto 6 - Projektverwaltung

21

## SPRINT ÜBERSICHT



AProto 7 - Sprint Übersicht



AProto 8 - Sprint Übersicht 2

## SPRINT ÜBERICHT

**SPRINT # - NAME**

**SPRINTNAME**

DATA: ### ← FORM ZUM EDITIEREN  
ALLE VALUES IN FORMS / DROPDOWNS

SAVE CANCEL

DIV BAR LARGE EACH BUTTON BY HOVER

+ ADD SPRINT B FILTER

AProto 9 - Editierungsformulare

	TODO	IN PROGRESS	DONE
ID 001 - USERSTORY M 100 40 TIME ### / ## / ## 40 6+	ID 002 - TASK ## / ## / ## 3+ ID 003 - TASK ## / ## / ## 3+	ID 001 - TASK ## / ## / ## 3+	ID 001 - TASK ## / ## / ## 3+

+ ADD TASK

AProto 10 - Task Übersicht