



Projekt: adit Projektplan

Oliver Dias odiaslal@hsr.ch, Fabian Hauser fhauser@hsr.ch, Murièle Trentini mtrentin@hsr.ch,
Nico Vinzens nvinzens@hsr.ch, Michael Wieland mwieland@hsr.ch

Änderungsgeschichte

Datum	Version	Änderung	Autor
23.02.2017	1.0	Erstellen des Dokuments	Projektteam
24.02.2017	1.1	Allgemeine Ergänzungen	Projektteam
27.02.2017	1.2	Meilensteine	tre
01.03.2017	1.3	Allgemeine Ergänzungen	hau, vin
02.03.2017	1.4	Allgemeine Ergänzungen	wie, vin, tre, hau
03.03.2017	1.5	Allgemeine Ergänzungen	tre
04.03.2017	2.0	Berechnung verfügbare Zeit pro MS	tre
09.03.2017	2.1	Spezifizierung gitflow	wie, vin
10.03.2017	2.2	Ergänzungen Testplan	wie
20.03.2017	2.3	Gantt-Diagramm für Issues einsetzen	hau
27.05.2017	2.4	Verantwortlichkeiten, weitere Ergänzungen und Überarbeitung	dia
30.05.2017	2.5	Änderung der Teamnamen→Back-/Frontend	tre
31.05.2017	2.6	Aktualisieren der Arbeitspakete und Gantt Chart	dia

Inhalt

Änderungsgeschichte	2
Inhalt.....	3
1. Einführung	5
1.1 Zweck	5
1.2 Gültigkeitsbereich	5
1.3 Referenzen	5
1.3.1 Quellenverzeichnis	5
2. Projekt Übersicht	6
2.1 Zweck und Ziel	6
2.2 Lieferumfang.....	6
2.3 Annahmen und Einschränkungen	6
3. Projektorganisation	7
3.1 Organisationsstruktur	7
3.1.1 Verantwortlichkeiten.....	8
3.2 Externe Schnittstellen	8
4. Management Abläufe	9
4.1 Kostenvoranschlag.....	9
4.2 Zeitliche Planung.....	9
4.2.1 Phasen / Iterationen.....	9
4.2.2 Meilensteine.....	10
4.3 Besprechungen	12
4.3.1 Sprint Review.....	12
5. Risikomanagement	13
5.1 Risiken	13
5.2 Umgang mit Risiken	13
6. Arbeitspakete	14
6.1 Arbeitspakete in MS1: Projektplan	14
6.2 Arbeitspakete in MS2: Anforderungen	15
6.3 Arbeitspakete in MS3: End of Elaboration.....	15
6.4 Arbeitspakete in MS4: Grundfunktionalität	16
6.5 Arbeitspakete in MS5: Feature Freeze	16
6.6 Arbeitspakete in MS6: Usability/Testing	17
6.7 Arbeitspakete in MS7: End of Construction	18
6.8 Arbeitspakete in MS8: Präsentation/Demo.....	19
6.9 Arbeitspakete in MS9: Abgabe	19
6.10 Workflow	20

6.11 Gantt-Diagram / Arbeitspakete	21
7. Infrastruktur	22
7.1 Entwicklungswerkzeuge.....	22
7.2 Frameworks	23
7.3 Weiteres.....	23
8. Qualitätsmassnahmen.....	24
8.1 Dokumentation	24
8.2 Projektmanagement	24
8.3 Entwicklung.....	24
8.3.1 Git Workflow	25
8.3.2 Code Reviews	26
8.3.3 Code Style Guidelines	26
8.3.4 Testing	26

1. Einführung

1.1 Zweck

Der vorliegende Projektplan beschreibt die geforderten Arbeitsschritte die im Laufe des Engineering Projekts durchzuführen sind. Es soll helfen, Klarheit über das angestrebte Ziel sowie die Anforderungen zu schaffen. Des Weiteren gibt es eine Übersicht über die Projektorganisation.

1.2 Gültigkeitsbereich

Der Gültigkeitsbereich beschränkt sich auf die Projektdauer des Modul Engineering Projekt FS17. Das Dokument wird HSR Intern verwendet.

1.3 Referenzen

Beschreibung	Name
Projektantrag	Projektantrag.pdf
Protokolle	5_Protokolle
Risikomanagement	TechnischeRisiken.pdf EingetreteneRisiken.pdf
Arbeitspakete und Zeiterfassung	Redmine (siehe Abschnitt 6 für Login)
Github Repositories	https://project.qo.is/projects/engineering-projekt/repository
Continuous Integration	https://travis-ci.org/
Testplan	Testplan.pdf

1.3.1 Quellenverzeichnis

Angular. (2. März 2017). *Angular2*. Von <https://angular.io> abgerufen

Apache. (2. März 2017). *Apache Maven*. Von <http://maven.apache.org/> abgerufen

Bitovi. (02. März 2017). *Embedded JavaScript*. Von <http://www.embeddedjs.com/> abgerufen

Docker. (24. Februar 2017). *Docker*. Von <https://www.docker.com/> abgerufen

Eclipse. (23. Februar 2017). *Eclipse Jetty*. Von <http://www.eclipse.org/jetty/> abgerufen

Eclipse. (23. Februar 2017). *Eclipse Neon*. Von <http://www.eclipse.org/neon/> abgerufen

Geosoft. (2. März 2017). *geosoft.no*. Von <http://geosoft.no/development/javastyle.html> abgerufen

Github. (23. Februar 2017). *Github*. Von <https://github.com/> abgerufen

Hibernate. (23. Februar 2017). *Hibernate*. Von <http://hibernate.org/> abgerufen

Jasmine. (2. März 2017). *Jasmine*. Von <https://jasmine.github.io/> abgerufen

Java Spark. (23. Februar 2017). *Java Spark*. Von <http://sparkjava.com/> abgerufen

Jetbrains. (2. März 2017). *jetbrains.com*. Von <https://www.jetbrains.com/webstorm/> abgerufen

JUnit. (2. März 2017). *JUnit*. Von <http://junit.org/junit4/> abgerufen

karma. (02. März 2017). *Karma*. Von <https://karma-runner.github.io/1.0/index.html> abgerufen

Kreeftmeijer, J. (2. März 2017). *JeffKreeftmeijer*. Von <http://jeffkreeftmeijer.com/2010/why-arent-you-using-git-flow/> abgerufen

Microsoft. (2. März 2017). *Microsoft Office*. Von <https://products.office.com/de-ch/home> abgerufen

myBalsamiq. (23. März 2017). *myBalsamiq*. Von <https://www.mybalsamiq.com/> abgerufen

PalantirTech. (02. März 2017). *TSLint*. Von <https://palantir.github.io/tslint/> abgerufen

PostgreSQL. (23. Februar 2017). *PostgreSQL*. Von <https://www.postgresql.org/> abgerufen

Redmine. (23. Februar 2017). *Redmine*. Von <http://redmine.com/> abgerufen

sass-lint. (02. März 2017). *sass Lint*. Von <https://www.npmjs.com/package/sass-lint> abgerufen

Spark. (2. März 2017). *SparkJava*. Von <http://sparkjava.com/> abgerufen

Travis CI. (23. Februar 2017). *Travis CI*. Von <https://travis-ci.org/> abgerufen

University of Maryland. (2. März 2017). *Findbugs*. Von <http://findbugs.sourceforge.net/> abgerufen

webpack. (02. März 2017). *Webpack*. Von <https://webpack.github.io/> abgerufen

2. Projekt Übersicht

Das Ziel unserer Dienstleistung ist es, eine Webapplikation für Inserate an der HSR zu erstellen. Über das Portal sollen damit Bücher, Jobs und WG Zimmer auf eine einfache Art und Weise ausgeschrieben werden können. Die Dienstleistung richtet sich ausschliesslich an Studenten, Dozenten und Mitarbeiter der HSR (HSR Login notwendig).

Die Webapplikation bietet eine REST API, die von einem Browser aufgerufen wird. Die Daten werden ausschliesslich serverseitig in einer relationalen Datenbank gehalten.

2.1 Zweck und Ziel

Ziel des Engineering Projekt ist es, die Methoden des Software Engineerings zu vertiefen und dabei ein praktisches Produkt zu schaffen. Wir möchten den HSR eine neue Plattform bieten, die es Studierenden erlaubt, Inserate einfach zu veröffentlichen.

Im Laufe des Projektes wollen wir uns mit neuartigen Technologien beschäftigen und die Herausforderungen bei der Arbeit in einem Projektteam näher kennenlernen. Am Ende des Projekts soll das geleistete reflektiert und die richtigen Schlüsse gezogen werden.

Im Speziellen werden wir uns in die beiden Frameworks Angular 2 und Spark einarbeiten sowie einen umfänglichen Build-, Deploy-Prozess inkl. Continuous Integration umsetzen. Ebenfalls gehört der Einsatz eines DBMS und das Betreiben der Server-Infrastruktur zu unseren Aufgaben.

Der spannendste Teil wird aber sein, ein selbstgewähltes Projekt von Anfang bis Ende durchführen zu können.

2.2 Lieferumfang

Am Ende des Projekts soll eine mobile-first Website geliefert werden, deren Daten in einem Webserver gehalten werden.

Der Lieferumfang beinhaltet folgende Dokumente:

Software & softwarespezifische Dokumente	Projektmanagement
Sourcecode	Projektplan
Domainmodell	Zeiterfassung /-auswertung
Use Case Diagramme	Anforderungsspezifikation
Statediagramme	Domainanalyse
SSD	Architekturplan
Testprotokolle	Meetingprotokolle
Installationsanleitung	Schlusspräsentation

2.3 Annahmen und Einschränkungen

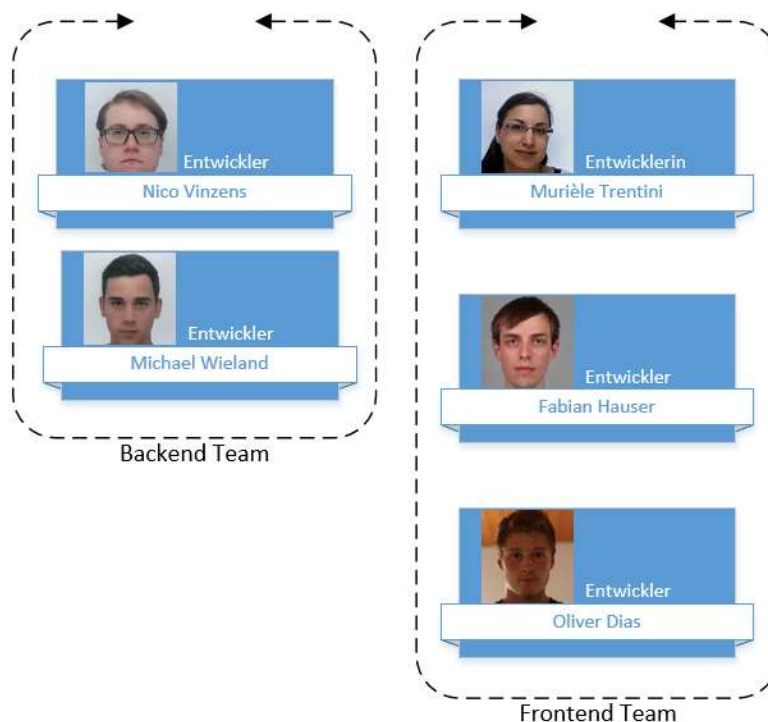
Die Projektdauer ist auf Ende des Semesters, am 02.06.2017 terminiert. Das Projekt sollte im Rahmen des geforderten Zeitaufwands von 600 Arbeitsstunden, bei 5 Projektmitglieder realisierbar sein. Das Hauptaugenmerk liegt bei der professionellen Anwendung von Software Engineering Methodik.

3. Projektorganisation

Das Projektteam ist als flache Organisationsstruktur organisiert. Alle Teammitglieder verfügen über die gleichen Rechte für strategische Entscheide. Um den organisatorischen Aufwand zu erleichtern wurde Fabian Hauser als primäre Ansprechperson für das Team auserkoren. Er wird als Projektleiter fungieren und das Team fortlaufend über Neuigkeiten informieren.

3.1 Organisationsstruktur

Das Team wird in zwei Teams aufgeteilt, welche sich um die zwei Kernbereiche kümmern. Das Frontend-Team kümmert sich um die Erstellung der Webapplikation und den Aufruf der REST API. Zusätzlich fallen hier noch Arbeiten bezüglich Interface Design an. Das Backend-Team kümmert sich um das Backend mit der dazugehörigen Server-Schnittstelle, sowie die Datenhaltung. Zudem kümmert sich das Frontend-Team um den automatischen Build- und Deploy-Prozess der Applikation.



Anmerkung: In Texten werden die Teammitglieder z.T. mit den ersten 3 Buchstaben des Nachnamens referenziert.

3.1.1 Verantwortlichkeiten

Neben den erwähnten Zuständigkeiten haben alle Projektmitarbeiter die Aufgabe, ihre Arbeit zu dokumentieren und die Dokumente stets aktuell zu halten. Des Weiteren gibt es Überschneidungen in den Verantwortlichkeitsbereichen innerhalb der Teams.

Name	Zuständigkeit
Nico Vinzens	<ul style="list-style-type: none">- JWT/Authentifizierung- Bereitstellen der REST API serverseitig- Umsetzung Berechtigungen
Michael Wieland	<ul style="list-style-type: none">- Persistierung der Daten/Aufsetzen Postgres DB- Bereitstellen der REST API serverseitig
Murièle Trentini	<ul style="list-style-type: none">- Inserate Übersicht/Inserate erstellen- Userinserate/Profil bearbeiten- Supervisorpanel- Testing
Fabian Hauser	<ul style="list-style-type: none">- Continuous Integration/Deployment- Authentifizierung/Login
Oli Dias	<ul style="list-style-type: none">- Inseratsuche- Adminpanel- REST API Definition

3.2 Externe Schnittstellen

Name	E-Mail	Zuständigkeit
Dr. Farhad Mehta	fmehta@hsr.ch	Betreuer
Fabian Hauser	fhauser@hsr.ch	Ansprechpartner

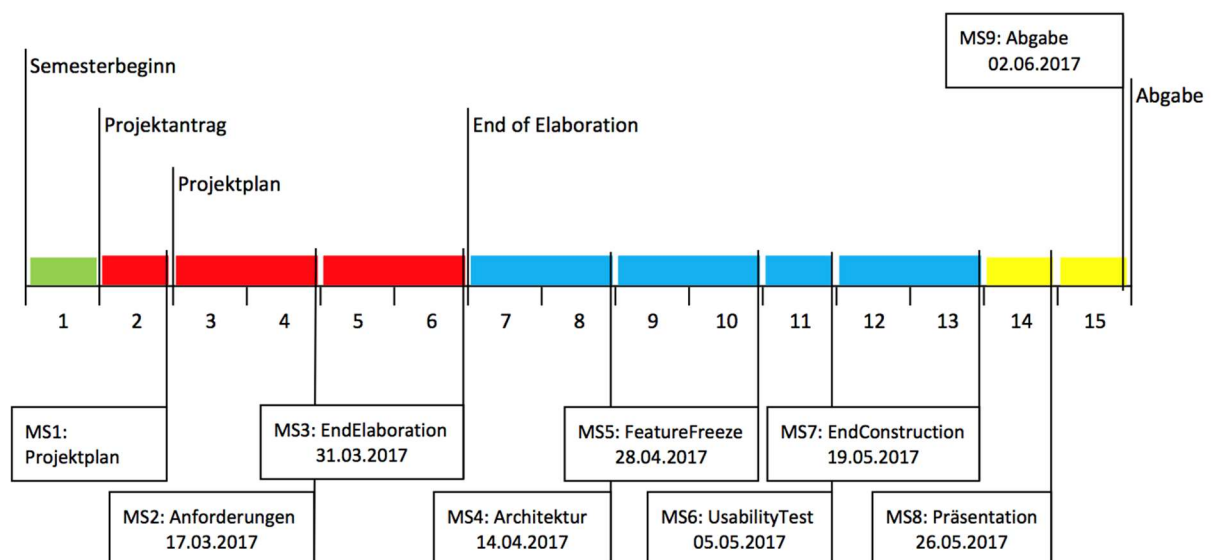
4. Management Abläufe

4.1 Kostenvoranschlag

Das Projekt läuft während 15 Wochen. Insgesamt stehen 600 Arbeitsstunden zur Verfügung. Diese Zeit wird für das Schreiben von Code, Erstellen von Dokumentationen, Testing, Reserve aufgrund von Risiken etc. verwendet.

Projektdauer	15 Wochen
Projektmitglieder	5 Personen
Arbeitsstunden pro Woche und Person	8h
Eingeplantes Risiko	49h bzw. 3.25h/Woche
Arbeitsstunden gesamt	600h
Projektstart	Montag, 20. Februar 2016
Projektende	Freitag, 2. Juni 2016

4.2 Zeitliche Planung



4.2.1 Phasen / Iterationen

4.2.1.1 Inception

Die Inception Phase dauert eine Woche und dient dem Erstellen des Projektantrags, eines Entwurfs des Projektplans, sowie der Recherche von zu verwendenden Technologien.

4.2.1.2 Elaboration

Die Elaboration Phase dauert 5 Wochen und beinhaltet 3 Iterationen. Am Ende der Elaboration sollen alle offenen Fragen geklärt sein. D.h., Die Requirements und Werkzeuge sind definiert, die Architektur steht und es existieren funktionierende Prototypen, sowie ein GUI-Entwurf.

4.2.1.3 Construction

Die Construction Phase dauert 7 Wochen mit 4 Iterationen. Während dieser Zeit wird der grösste Teil des Codes geschrieben. Gegen Ende der Construction Phase werden die Features eingefroren und

unsere Applikation wird ausgiebig getestet. Am Ende soll unser Projekt soweit fertiggestellt sein, dass nur noch der Projektplan und die Präsentation ausstehen.

4.2.1.4 Transition

Die Transition Phase dauert 2 Wochen mit 2 Iterationen. Diese Zeit wird verwendet um den Projektplan fertigzustellen und die Präsentation vorzubereiten und durchzuführen.

4.2.2 Meilensteine

4.2.2.1 MS1: Projektplan

Datum	03.03.17	
Beschreibung	Projektplan in Rohfassung mit Phasen, Iterationen, Meilensteinen mit Datum	
Dokumente	Projektplan	
Verfügbare Zeit	2 Woche: $8h * 5 * 2 = 80h$ abzüglich Risiko von $2*3.25h$	<u>73.5h</u>

4.2.2.2 MS2: Anforderungen

Datum	17.03.17	
Beschreibung	Funktionale Anforderungen: Use Cases in 'brief' Format Nicht-Funktionale Anforderungen	
Dokumente	Use Case Diagramm Domain-Modell Zustandsdiagramm Instate	
Verfügbare Zeit	2 Woche: $8h * 5 * 2 = 80h$ abzüglich Risiko von $2*3.25h$	<u>73.5h</u>

4.2.2.3 MS3: End of Elaboration mit Prototyp

Datum	31.03.17	
Beschreibung	Architekturprototyp	
Dokumente	WireFrame der Website Testplanung und -protokolle vorbereiten Architektur/Schichten Diagramm Backlog erfasst und Gantt Chart	
Verfügbare Zeit	2 Woche: $8h * 5 * 2 = 80h$ abzüglich Risiko von $2*3.25h$	<u>73.5h</u>

4.2.2.4 MS4: Grundfunktionalität Architektur/Design

Datum	14.04.17	
Beschreibung	Frontend/Backend Kommunikation soll grundlegend funktionieren	
Dokumente	Keine	
Verfügbare Zeit	2 Woche: $8h * 5 * 2 = 80h$ abzüglich Risiko von $2*3.25h$	<u>73.5h</u>

4.2.2.5 MS5: Feature Freeze

Datum	28.04.17	
Beschreibung	Alle notwendigen Features sind erfasst, es werden keine neuen Features mehr angefangen	
Dokumente	Keine	
Verfügbare Zeit	2 Woche: $8h * 5 * 2 = 80h$ abzüglich Risiko von $2*3.25h$	<u>73.5h</u>

4.2.2.6 MS6: Usability/Testing

Datum	05.05.17	
Beschreibung	Website wird ausführlich getestet und der Code wird anhand der Testresultate ergänzt / angepasst	
Dokumente	Testspezifikation Testprotokolle Testresultate	
Verfügbare Zeit	1 Woche: $8h * 5 = 40h$ abzüglich Risiko von 3.25h	<u>36.75h</u>

4.2.2.7 MS7: End of Construction

Datum	19.05.17	
Beschreibung	Website erfüllt Requirements, die in der Elaboration definiert wurden.	
Dokumente	Deployment-Diagramm Daten-Modell Metriken (vom Code)	
Verfügbare Zeit	2 Woche: $8h * 5 * 2 = 80h$ abzüglich Risiko von $2 * 3.25h$	<u>73.5h</u>

4.2.2.8 MS8: Präsentation / Demo

Datum	26.05.2017	
Beschreibung	Erstellung einer PowerPoint-Präsentation	
Dokumente	PowerPoint-Präsentation	
Verfügbare Zeit	1 Woche: $8h * 5 = 40h$ abzüglich Risiko von 3.25h	<u>36.75h</u>

4.2.2.9 MS9: Abgabe

Datum	02.06.2017	
Beschreibung	Fertigstellen des Projektplans, Abschliessende Arbeiten	
Dokumente	Projektplan	
Verfügbare Zeit	1 Woche: $8h * 5 = 40h$ abzüglich Risiko von 3.25h	<u>36.75h</u>

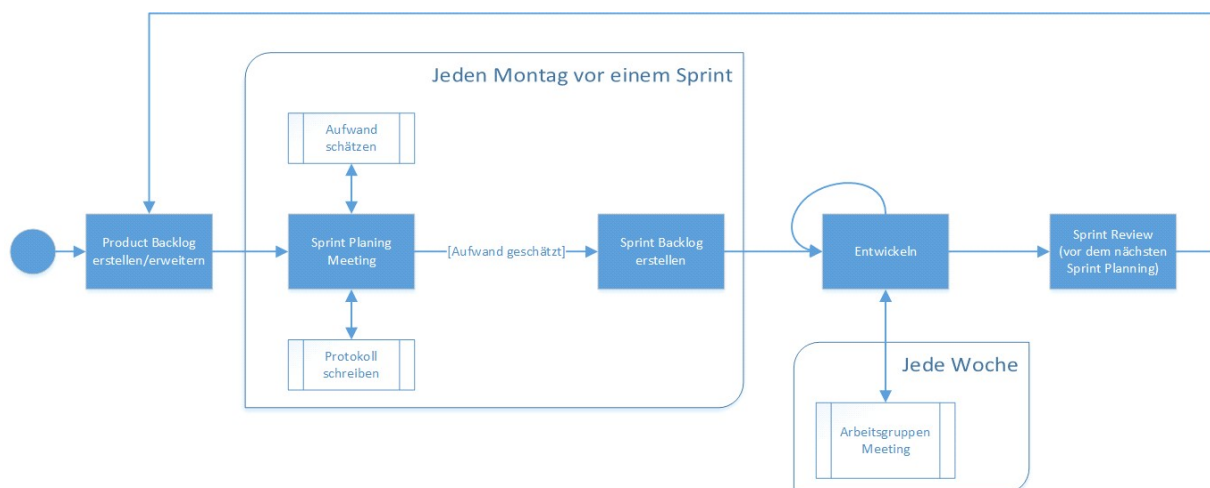
4.3 Besprechungen

Die Teams treffen sich jeweils montags 10.00- 12.00 Uhr, um allfällige Probleme zu klären und sicherzustellen, dass der Sprint gemäss Planung abgeschlossen wird.

Am Ende jedes Sprints trifft sich das ganze Projektteam am Montag 10.00 Uhr-12.00 Uhr zum Sprint-Review und zur Planung des nächsten Sprints.

Die Reviews mit dem Betreuer werden jeweils mittwochs um 15:15-16:00 durchgeführt.

Folgendes Flowchart visualisiert den Ablauf eines Sprints:



4.3.1 Sprint Review

Beim Sprint Review wird das Vorgehen noch einmal reflektiert und im Team besprochen.

Zeitauswertung:

Neben der Analyse der geleisteten Arbeit wird ebenfalls die Zeitauswertung analysiert. Dabei prüfen wir, ob alle Teammitglieder ungefähr den gleichen Aufwand investiert haben und ob wir als Team, die geschätzten Zeit einhalten konnten. Die Resultate werden im Protokoll erfasst.

Risikoauswertung:

Zusätzlich zur Zeitauswertung wird überprüft, ob die Punkte aus der Risikoplanung im letzten Sprint aufgetreten sind. Da wir den gewichteten Schaden aus dem Risikomanagement bereits bei der Zeitschätzung der Issues berücksichtigen, sollte das Auftreten eines Risikos einen verkraftbaren Impact auf das Zeitmanagement haben. Ist der Schaden jedoch grösser als erwartet, wird die zusätzliche Zeit in der Datei [TechnischeRisiken.xlsx](#) notiert und der nächste Sprint entsprechend geplant.

5. Risikomanagement

5.1 Risiken

Die Risikoplanung ist in der externen Datei [TechnischeRisiken.xlsx](#) zu finden. Durch das agile Projektvorgehen kann beim Eintreten eines Zwischenfalls das weitere Vorgehen besprochen und die Projektrichtung entsprechend angepasst werden.

5.2 Umgang mit Risiken

Die Risiken werden so gut wie möglich, bereits im Vorhinein, durch eine umfängliche Projektplanung minimiert. Dazu wurden alle Projektmitglieder über die vorhandenen Risiken informiert und sensibilisiert.

Für den Fall, dass trotzdem ein Problem auftritt, wurden die beschriebenen Risiken in der Zeitplanung berücksichtigt. In jedem Arbeitspaket wird daher ein Zeitpolster eingerechnet. Dieses Polster beläuft sich auf 3.25h pro Woche. Die Reserven sind somit gleichmässig über alle Iterationen verteilt. Sollten unerwartet viele Probleme auftreten, wird die Projektplanung neu evaluiert und das weitere Vorgehen im Team und mit dem Betreuer besprochen. Die Risiken werden in den jeweiligen Sprint Reviews besprochen und entschieden, ob in Zukunft etwas anders gemacht werden muss oder ob im gleichen Stile weiter gemacht werden kann.

6. Arbeitspakete

Alle Arbeitspakete sind im Redmine genau spezifiziert und den Projektmitgliedern zugewiesen.

URL	https://project.go.is/projects/engineering-projekt
Username	"guest"
Passwort	"Guest%12345"

Jedes Arbeitspaket wird im Redmine erfasst, geschätzt, priorisiert und in einen Sprint eingeplant. Einige Pakete werden im Laufe des Projektes noch genauer spezifiziert. Die aufgewendete Zeit wird einer Aktivitätsgruppe zugeteilt. Dies erleichtert die Zeitauswertung pro Gruppe.

Nr	Aktivitätsgruppe	Beschreibung
1.	Project Management	Projektplanung, Administrative Aufgaben
2.	Development	Entwicklung der Applikationen
3.	Testing	Review und Testing der Features
4.	Application Design	Technologie Stack und Architekturfragen und Diagramme
5.	Infrastructure / Deployment	Server Infrastruktur und CI, Versionierung, Code Style

Im Folgenden werden die zum Ende des Projekts definierten Arbeitspakete aufgelistet. Alle Issues wurden entweder erledigt (Closed) oder abgelehnt (Rejected). Anzumerken ist, dass die geschätzte Zeit nicht mit derjenigen vom Beginn des Projekts übereinstimmt, da Issues z.T. von einem Sprint in den nächsten verschoben werden mussten (und somit auch die schon darauf geloggte Zeit).

6.1 Arbeitspakete in MS1: Projektplan

MS1: Projektplan 12 Estimated time: 73.25 Spent time: 77.08 Collapse all/Expand all									
<input type="checkbox"/>	49	Issue Collection	Closed	Recherche	Normal		03/17/2017		0
<input type="checkbox"/>	52	Task	Closed	► Recherche SoftwareStack Server	Normal	Backend	03/03/2017	10.00	10.50
<input type="checkbox"/>	55	Task	Closed	Grundkonfiguration Virtueller Server	Normal	Backend	03/03/2017	1.00	1.00
<input type="checkbox"/>	50	Task	Rejected	Recherche Xamarin	Normal	Frontend		5.00	5.50
<input type="checkbox"/>	47	Task	Closed	Projektplan erstellen	Normal	Projektarbeit		40.00	44.50
<input type="checkbox"/>	60	Task	Closed	► Risikoplanung erstellen	Normal	Projektarbeit		2.00	2.75
<input type="checkbox"/>	46	Task	Closed	Projektantrag erstellen	Normal	Projektarbeit	02/24/2017	5.00	5.00
<input type="checkbox"/>	51	Task	Closed	Redmine konfigurieren	Normal	Projektarbeit	03/03/2017	5.00	3.75
<input type="checkbox"/>	58	Task	Closed	Issues / Arbeitspakete definieren	Normal	Projektarbeit	03/03/2017	3.00	1.75
<input type="checkbox"/>	57	Task	Closed	Dokumentationen organisieren	Normal	Projektarbeit	03/03/2017	1.00	1.00
<input type="checkbox"/>	54	Task	Closed	Projekt-Homepage erstellen	Normal	Projektarbeit	03/03/2017	0.75	0.75
<input type="checkbox"/>	48	Task	Closed	Virtuellen Server Beantragen	Normal	Projektarbeit	03/03/2017	0.50	0.58

6.2 Arbeitspakete in MS2: Anforderungen

MS2: Anforderungen 16 Estimated time: 72.50 Spent time: 81.15								
<input type="checkbox"/>	73	Task	Closed	Entwicklungsprozess festlegen	Normal	Backend	4.00	0
<input type="checkbox"/>	102	Task	Closed	► Gitflow anschauen und Integration in Eclipse	Normal	Backend		1.90
<input type="checkbox"/>	103	Task	Closed	► Checkstyle	Normal	Backend		1.25
<input type="checkbox"/>	104	Task	Closed	► Unit und Integration Tests planen	Normal	Backend		1.00
<input type="checkbox"/>	53	Task	Closed	Recherche Server: Server Grundstruktur erstellen	Normal	Backend	03/03/2017	8.00
<input type="checkbox"/>	70	Task	Closed	Client Projektgrundstruktur erstellen	Normal	Frontend		7.00
<input type="checkbox"/>	72	Task	Closed	Entwicklungsprozess festlegen	Normal	Frontend		4.00
<input type="checkbox"/>	112	Feature	Closed	Umsetzung Wireframe-Vorlage: Start-Seite	Normal	Frontend		2.00
<input type="checkbox"/>	113	Feature	Closed	Assets Packaging für Weppack	Normal	Frontend		0.50
<input type="checkbox"/>	56	Task	Closed	Recherche Client: Technologien	Normal	Frontend	03/17/2017	16.00
<input type="checkbox"/>	61	Task	Closed	Domainmodell erstellen	Normal	Projektarbeit		8.00
<input type="checkbox"/>	67	Task	Closed	Issue Backlog und Gantt Diagramm	Normal	Projektarbeit		8.00
<input type="checkbox"/>	63	Task	Closed	Use Case Diagramme	High	Projektarbeit		6.00
<input type="checkbox"/>	64	Task	Closed	Nichtfunktionale Anforderungen definieren	Normal	Projektarbeit		4.00
<input type="checkbox"/>	106	Task	Closed	Contracts zu Domainmodell erstellen	Normal	Projektarbeit		3.00
<input type="checkbox"/>	65	Task	Closed	Zustandsdiagramm	High	Projektarbeit		2.00

6.3 Arbeitspakete in MS3: End of Elaboration

MS3: Ende Elaboration mit Prototype 19 Estimated time: 92.50 Spent time: 90.25								
<input type="checkbox"/>	88	Feature	Closed	ORM-Anbindung Datenbank	Normal	Backend		8.00
<input type="checkbox"/>	130	Task	Closed	Testdatenbank aufsetzen und Integration Tests Erweitern	Normal	Backend		6.00
<input type="checkbox"/>	69	Task	Closed	REST API Spezifizieren	Normal	Backend		4.00
<input type="checkbox"/>	128	Feature	Closed	Exception Handling	Normal	Backend		3.00
<input type="checkbox"/>	86	Feature	Closed	Datenmodell Umsetzen	Normal	Backend		2.50
<input type="checkbox"/>	78	Task	Closed	Postgres Server einrichten	Normal	Backend		2.00
<input type="checkbox"/>	127	Feature	Closed	JWT / KeyStore	Normal	Backend		2.00
<input type="checkbox"/>	131	Task	Closed	Datenbankpasswort sicher ablegen	Normal	Backend		1.00
<input type="checkbox"/>	62	Task	Closed	Wireframes Erstellen	Normal	Frontend		8.00
<input type="checkbox"/>	76	Task	Closed	Travis / CI Einrichten	Normal	Frontend		8.00
<input type="checkbox"/>	84	Feature	Closed	REST-API spezifizieren	Normal	Frontend		8.00
<input type="checkbox"/>	74	Task	Closed	Build Prozess	Normal	Frontend		6.00
<input type="checkbox"/>	138	Feature	Closed	Inserate CRUD	Normal	Frontend		6.00
<input type="checkbox"/>	83	Feature	Closed	Angular Datenmodelle umsetzen	Normal	Frontend		4.00
<input type="checkbox"/>	115	Task	Closed	Client Schichtendiagramm erstellen	High	Frontend		3.00
<input type="checkbox"/>	116	Feature	Closed	Auflistung Inserate	Normal	Frontend		2.00
<input type="checkbox"/>	137	Feature	Closed	Inserat Details anzeigen	Normal	Frontend		1.00
<input type="checkbox"/>	68	Task	Closed	Architekturplan erstellen	Normal	Projektarbeit		10.00
<input type="checkbox"/>	66	Task	Closed	Test- und Reviewprozeduren definieren	Normal	Projektarbeit		8.00

6.4 Arbeitspakete in MS4: Grundfunktionalität

MS4: Grundfunktionalität Architektur / Design 15 Estimated time: 58.75 Spent time: 57.58								
<input type="checkbox"/>	75	Task	Closed	Build Prozess	Normal	Backend	14.00	13.00
<input type="checkbox"/>	111	Task	Closed	Sonarqube anbinden	Low	Backend	4.00	5.50
<input type="checkbox"/>	153	Feature	Closed	Statuscode Testing für Inserat / Tag / Category	Normal	Backend	3.00	2.50
<input type="checkbox"/>	147	Feature	Closed	Enum Typen einführen	Normal	Backend	1.50	1.25
<input type="checkbox"/>	151	Feature	Closed	Gson: PK als Long serialisieren	Normal	Backend	0.75	1.25
<input type="checkbox"/>	149	Feature	Closed	Integrationstests: Jetty nur einmal starten	Normal	Backend	0.50	0.50
<input type="checkbox"/>	150	Feature	Closed	PK Sequenzen sauber herauszählen	Normal	Backend	0.50	1.83
<input type="checkbox"/>	152	Feature	Closed	Hibernate Connection Pool umstellen	Normal	Backend	0.50	0.33
<input type="checkbox"/>	80	Feature	Rejected	Grundlayout erstellen	Normal	Frontend	11.00	0
<input type="checkbox"/>	79	Task	Closed	nginx einrichten	Normal	Frontend	4.00	4.00
<input type="checkbox"/>	148	Feature	Closed	Verbesserung Inserate Create (Issue 138)	Normal	Frontend	2.00	2.00
<input type="checkbox"/>	141	Feature	Closed	Create Inserat (Server <> Client)	High	Projektarbeit	10.00	12.25
<input type="checkbox"/>	139	Feature	Closed	Error Codes definieren und mappen	Normal	Projektarbeit	4.00	11.83
<input type="checkbox"/>	145	Feature	Closed	Dokumente nachführen nach Review	Normal	Projektarbeit	2.00	0.25
<input type="checkbox"/>	144	Feature	Closed	Prototyp Demo vorbereiten	Normal	Projektarbeit	04/04/2017 1.00	1.08

6.5 Arbeitspakete in MS5: Feature Freeze

MS5: Feature Freeze 20 Estimated time: 56.00 Spent time: 74.80 Collapse all/Expand all								
<input type="checkbox"/>	109	Feature	Rejected	Biete / Suche Funktion hinzufügen (Optional)	Normal		0.00	0
<input type="checkbox"/>	185	Feature	Closed	Passwortmanagement	Normal	Backend	4.00	6.25
<input type="checkbox"/>	176	Feature	Closed	Subscription API bereitstellen	Normal	Backend	1.00	1.00
<input type="checkbox"/>	186	Feature	Closed	Message API erstellen	Normal	Backend	1.00	0.92
<input type="checkbox"/>	189	Feature	Closed	Update, Delete und GetAll Methoden überdenken	Normal	Backend	0.75	0.75
<input type="checkbox"/>	165	Feature	Closed	Categories API bereitstellen	Normal	Backend	0.50	0.25
<input type="checkbox"/>	166	Feature	Closed	Role API bereitstellen	Normal	Backend	0.50	1.25
<input type="checkbox"/>	167	Feature	Closed	Permission API bereitstellen	Normal	Backend	0.50	2.08
<input type="checkbox"/>	168	Feature	Closed	User API bereitstellen	Normal	Backend	0.50	2.30
<input type="checkbox"/>	187	Feature	Closed	Advertisement API erweitern	Normal	Backend	0.50	0.50
<input type="checkbox"/>	82	Feature	Closed	Listenansicht / Filter erstellen	High	Frontend	19.00	23.25
<input type="checkbox"/>	143	Feature	Closed	CRUD Inserate (API)	Normal	Frontend	12.00	17.25
<input type="checkbox"/>	129	Feature	Closed	Media Queries -> Desktop/Mobile	Normal	Frontend	5.00	4.25
<input type="checkbox"/>	195	Bug	Closed	Diverse BugFixes	Normal	Frontend	3.00	2.50
<input type="checkbox"/>	157	Feature	Closed	Registration (Create User)	Normal	Frontend	2.00	5.75
<input type="checkbox"/>	164	Feature	Closed	Supervisorpanel erstellen	Normal	Frontend	2.00	3.00
<input type="checkbox"/>	136	Feature	Closed	i18n	Normal	Frontend	1.00	2.00
<input type="checkbox"/>	172	Feature	Closed	Inseratliste filtern	Normal	Frontend	1.00	0.25
<input type="checkbox"/>	156	Task	Closed	Architekturdiagramm mit SearchProposal anpassen	Normal	Frontend	0.25	0.50
<input type="checkbox"/>	154	Feature	Closed	REST API nachführen	Normal	Projektarbeit	1.50	0.75

6.6 Arbeitspakete in MS6: Usability/Testing

<div> <div>MS6: Usability / Testing</div> <div>23</div> <div>Estimated time: 44.50 Spent time: 49.25</div> </div>								
<input type="checkbox"/>	94	Feature	Closed	Speedtests	Normal	Backend	6.00	2.50
<input type="checkbox"/>	159	Feature	Closed	JWT Token nach Login liefern	Normal	Backend	4.00	10.08
<input type="checkbox"/>	193	Feature	Closed	REST API: Trailing Slash	Normal	Backend	2.50	1.75
<input type="checkbox"/>	192	Feature	Closed	Update Methoden anpassen	Normal	Backend	2.00	3.25
<input type="checkbox"/>	214	Bug	Closed	Authorization Bugs	High	Backend	1.00	0.75
<input type="checkbox"/>	215	Feature	Closed	Authentication Refactoring	Normal	Backend	1.00	1.25
<input type="checkbox"/>	205	Feature	Closed	Nicht gebrauchte Tags löschen	Normal	Backend	0.75	1.50
<input type="checkbox"/>	197	Feature	Closed	CORS Header	Normal	Backend	0.50	0.33
<input type="checkbox"/>	206	Feature	Closed	Advertisementsuche mit OR	Normal	Backend	0.50	0.33
<input type="checkbox"/>	207	Task	Closed	GSON Klassen im Model/Architekturplan nachtragen	Normal	Backend	0.50	0.75
<input type="checkbox"/>	110	Task	Rejected	Coverity anbinden	Low	Backend	0.00	0
<input type="checkbox"/>	198	Feature	Closed	Unit Tests Authentication Services	Normal	Frontend	6.00	2.00
<input type="checkbox"/>	142	Feature	Closed	Tests nachführen	Normal	Frontend	4.00	7.50
<input type="checkbox"/>	162	Feature	Closed	Userprofil bearbeiten	Normal	Frontend	3.00	4.00
<input type="checkbox"/>	199	Feature	Closed	Unit Tests Supervisorpanel	Normal	Frontend	2.00	3.00
<input type="checkbox"/>	200	Feature	Closed	Unit Tests AdvertisementList	Normal	Frontend	2.00	3.00
<input type="checkbox"/>	202	Feature	Closed	Unit Tests AdvertisementInfo	Normal	Frontend	2.00	2.00
<input type="checkbox"/>	204	Feature	Closed	Unit Tests UseradvertisementComponent	Normal	Frontend	2.00	1.75
<input type="checkbox"/>	217	Feature	Closed	Unit Tests Userprofile	Normal	Frontend	2.00	1.00
<input type="checkbox"/>	218	Feature	Closed	Unit Tests Advertisement Service	Normal	Frontend	2.00	2.00
<input type="checkbox"/>	213	Bug	Closed	Verbesserung ManageCategories Component	Normal	Frontend	0.50	0.25
<input type="checkbox"/>	212	Task	Closed	Einheitliche Strings auf English	Normal	Frontend	0.25	0.25
<input type="checkbox"/>	201	Feature	Rejected	Unit Tests Services	Normal	Frontend	0.00	0

6.7 Arbeitspakete in MS7: End of Construction

MS7: End of Construction 38 Estimated time: 162.00 Spent time: 78.50								
<input type="checkbox"/>	118	Feature	Rejected	Abonnieren von Kategorien	Normal		24.00	0
<input type="checkbox"/>	184	Feature	Closed	Umsetzung Permission	Normal	Backend	10.00	10.50
<input type="checkbox"/>	178	Feature	Rejected	Media Uploader API	Normal	Backend	6.00	3.00
<input type="checkbox"/>	107	Feature	Rejected	Passwort reset (Mailversand / Framework)	Normal	Backend	4.00	0
<input type="checkbox"/>	188	Feature	Rejected	Advertisement expired CRON-Job	Normal	Backend	4.00	0
<input type="checkbox"/>	228	Feature	Closed	Sonarqube Report aufarbeiten	Normal	Backend	3.50	3.00
<input type="checkbox"/>	173	Feature	Rejected	Hibernate paging	Normal	Backend	3.00	0
<input type="checkbox"/>	177	Feature	Rejected	Cron-Job für Subscriptions	Normal	Backend	3.00	0
<input type="checkbox"/>	216	Feature	Closed	JWT Token in der DB ablegen	Normal	Backend	3.00	2.25
<input type="checkbox"/>	223	Feature	Closed	Testcoverage erhöhen	Normal	Backend	3.00	3.00
<input type="checkbox"/>	181	Feature	Rejected	UserLog nachführen und API bereitstellen	Normal	Backend	2.00	0
<input type="checkbox"/>	155	Feature	Rejected	Fehlermeldungen nicht nach aussen propagieren	Normal	Backend	1.00	0
<input type="checkbox"/>	170	Feature	Rejected	SMTP Server aufsetzen	Normal	Backend	1.00	0
<input type="checkbox"/>	174	Feature	Rejected	Messaging / Conversation Screen implementieren	Normal	Frontend	12.00	0
<input type="checkbox"/>	158	Feature	Closed	User Login	High	Frontend	9.00	8.00
<input type="checkbox"/>	92	Feature	Rejected	e2e Tests	Low	Frontend	8.00	10.00
<input type="checkbox"/>	179	Feature	Rejected	Media-Upload	Normal	Frontend	8.00	0
<input type="checkbox"/>	194	Task	Closed	Bereinigen Arbeitsumgebung	High	Frontend	7.00	6.75
<input type="checkbox"/>	225	Feature	Closed	UI Redesign	Normal	Frontend	6.50	5.50
<input type="checkbox"/>	175	Feature	Rejected	Subscription auf Kategorien	Normal	Frontend	4.00	0
<input type="checkbox"/>	190	Feature	Closed	Supervisorpanel erstellen - manage advertisements (Issue 164)	Normal	Frontend	4.00	2.50
<input type="checkbox"/>	196	Feature	Closed	Unit Tests Search	High	Frontend	3.50	2.50
<input type="checkbox"/>	169	Feature	Rejected	Passwort reset	Normal	Frontend	3.00	0
<input type="checkbox"/>	211	Task	Closed	Leermeldung bei fehlenden Suchergebnissen anzeigen	Normal	Frontend	3.00	4.00
<input type="checkbox"/>	171	Feature	Rejected	Inserat reaktivieren	Normal	Frontend	2.00	0
<input type="checkbox"/>	180	Feature	Rejected	Userlog implementieren	Normal	Frontend	2.00	0
<input type="checkbox"/>	203	Feature	Closed	Unit Tests Register Component	Normal	Frontend	2.00	0.75
<input type="checkbox"/>	209	Task	Closed	Deployment Diagramm in Architekturplan einfügen	Normal	Frontend	2.00	1.50
<input type="checkbox"/>	219	Feature	Closed	Unit Tests Login Component	Normal	Frontend	2.00	1.00
<input type="checkbox"/>	220	Feature	Closed	Unit Tests Category Service	Normal	Frontend	2.00	0.75
<input type="checkbox"/>	221	Feature	Closed	Unit Tests Tag Service	Normal	Frontend	2.00	1.50
<input type="checkbox"/>	222	Feature	Closed	Unit Tests User Service	Normal	Frontend	2.00	0.50
<input type="checkbox"/>	210	Bug	Closed	Nicht nur Expired Inserate in Suche anzeigen	Normal	Frontend	1.00	1.00
<input type="checkbox"/>	226	Feature	Closed	Unit Tests AppComonent etc	Normal	Frontend	1.00	0.75
<input type="checkbox"/>	191	Bug	Closed	Menu Desktop Version	Normal	Frontend	0.50	0.75
<input type="checkbox"/>	133	Task	Closed	Systemtests für Prototyp durchführen	Normal	Projektarbeit	4.00	3.00
<input type="checkbox"/>	182	Feature	Closed	Permission definieren	Normal	Projektarbeit	2.00	4.00
<input type="checkbox"/>	229	Task	Closed	Systemtests erweitern	Normal	Projektarbeit	2.00	2.00

6.8 Arbeitspakete in MS8: Präsentation/Demo

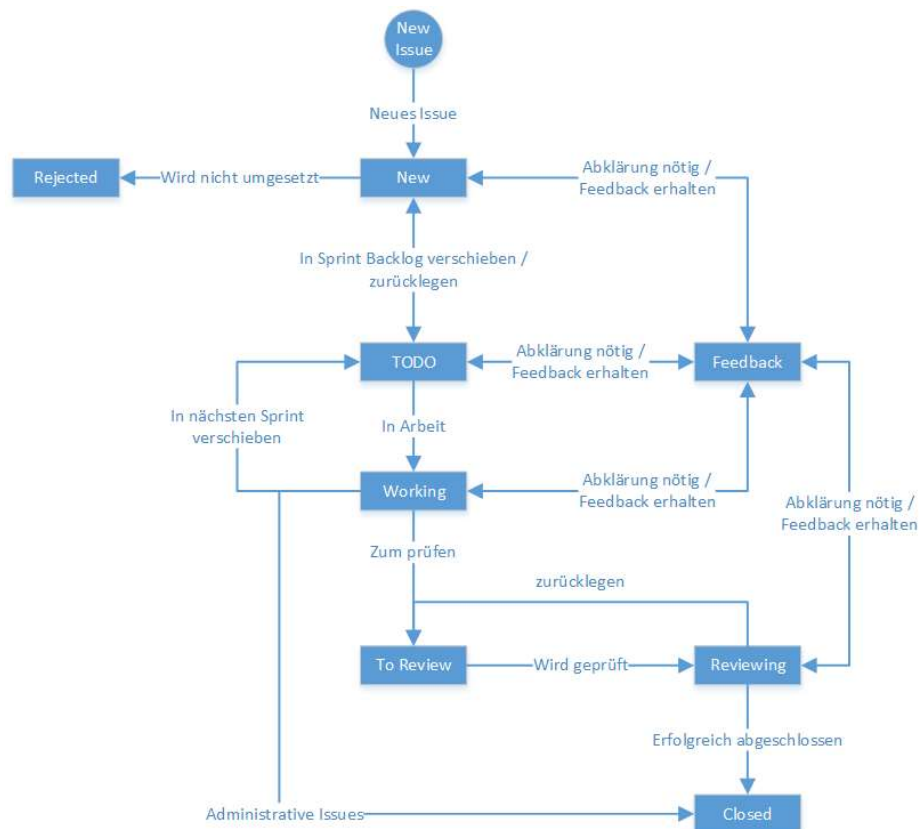
MS8: Präsentation / Demo 23 Estimated time: 54.00 Spent time: 56.25								
<input type="checkbox"/>	135	Task	Closed	Usability Tests durchführen	Normal	Backend	3.00	5.75
<input type="checkbox"/>	245	Bug	Closed	JWT Token Handling	Normal	Backend	1.50	0.50
<input type="checkbox"/>	230	Feature	Closed	Testuser mit Rollen erstellen	Normal	Backend	1.00	0.50
<input type="checkbox"/>	247	Bug	Closed	Supervisor Rolle braucht edit_categories Permission	Normal	Backend	0.50	0.25
<input type="checkbox"/>	163	Feature	Closed	Adminpanel erstellen	High	Frontend	10.00	10.00
<input type="checkbox"/>	146	Task	Closed	Widget für Error, Success und Help Messages	Normal	Frontend	8.50	9.75
<input type="checkbox"/>	160	Feature	Closed	Anbindung User an bestehenden Code	Normal	Frontend	3.50	3.75
<input type="checkbox"/>	183	Feature	Closed	Umsetzung Permission	Normal	Frontend	2.50	3.00
<input type="checkbox"/>	224	Task	Closed	Anpassung Architektur Dokument Schichtendiagramm	Normal	Frontend	2.00	2.25
<input type="checkbox"/>	227	Bug	Closed	Nach zweifachem Suchen funktioniert die Suche nicht	Normal	Frontend	1.00	0
<input type="checkbox"/>	243	Bug	Closed	Entfernen von E-Mail-Adressfeld bei Benutzerprofil	Normal	Frontend	1.00	1.00
<input type="checkbox"/>	244	Bug	Closed	Keine Eingabe von neuem Kategoriennamen möglich	Normal	Frontend	0.50	0.50
<input type="checkbox"/>	249	Bug	Closed	Tag wird doppelt in Suche angezeigt	High	Frontend		1.00
<input type="checkbox"/>	105	Task	Rejected	Zeitauswertung	Normal	Projektarbeit	5.00	4.75
<input type="checkbox"/>	96	Task	Closed	Grundversion Präsentation	Normal	Projektarbeit	3.00	3.00
<input type="checkbox"/>	241	Feature	Closed	Installationsanleitung erstellen	Normal	Projektarbeit	2.50	2.50
<input type="checkbox"/>	234	Feature	Closed	Präsentation Michi	Normal	Projektarbeit	2.00	1.50
<input type="checkbox"/>	237	Feature	Closed	Projektplan Updaten	Normal	Projektarbeit	2.00	2.50
<input type="checkbox"/>	238	Feature	Closed	Anforderungsspezifikation updaten	Normal	Projektarbeit	1.50	1.75
<input type="checkbox"/>	208	Task	Closed	Anpassen der Position der "nichtfunktionalen" Anforderungen	Normal	Projektarbeit	1.00	0.50
<input type="checkbox"/>	231	Feature	Closed	Logo updaten und Favicon hinzufügen	Normal	Projektarbeit	1.00	0.75
<input type="checkbox"/>	240	Feature	Closed	REST-API mit Token updaten	Normal	Projektarbeit	1.00	0.75
<input type="checkbox"/>	246	Feature	Rejected	Fehlende Präsentations Punkte?	High	Projektarbeit	05/26/2017	0

6.9 Arbeitspakete in MS9: Abgabe

MS9: Abgabe 16 Estimated time: 82.00 Spent time: 32.00								
<input type="checkbox"/>	253	Feature	Closed	Testplan reviewen	Normal	Backend	1.00	0.50
<input type="checkbox"/>	248	Bug	Closed	erstes Inserat eines Users im Nirvana	Normal	Frontend	2.00	2.50
<input type="checkbox"/>	100	Task	Closed	Projektplan updaten	Normal	Projektarbeit	32.00	2.75
<input type="checkbox"/>	98	Task	Closed	Projektplan Abschliessen	Normal	Projektarbeit	16.00	0.50
<input type="checkbox"/>	126	Task	Closed	Präsentation Vorbereiten / Üben	Normal	Projektarbeit	8.00	6.25
<input type="checkbox"/>	97	Task	Closed	Demo vorbereiten	Normal	Projektarbeit	6.00	3.00
<input type="checkbox"/>	99	Task	Closed	Template Zeitauswertung erstellen	Normal	Projektarbeit	4.00	0
<input type="checkbox"/>	232	Task	Closed	Präsentation Muriele	Normal	Projektarbeit	2.00	2.00
<input type="checkbox"/>	233	Feature	Closed	Präsentation Nico	Normal	Projektarbeit	2.00	4.00
<input type="checkbox"/>	235	Task	Closed	Präsentation Fabian	Normal	Projektarbeit	2.00	1.25
<input type="checkbox"/>	236	Task	Closed	Präsentation Oli	Normal	Projektarbeit	2.00	4.00
<input type="checkbox"/>	242	Feature	Closed	Codestatistik erstellen	Normal	Projektarbeit	1.50	1.50
<input type="checkbox"/>	239	Feature	Closed	Architekturdokument updaten	Normal	Projektarbeit	1.00	0.75
<input type="checkbox"/>	250	Feature	Closed	Präsentation aktualisieren	Normal	Projektarbeit	1.00	1.00
<input type="checkbox"/>	252	Task	Closed	Rückblick und Erkenntnisse schreiben	Normal	Projektarbeit	1.00	1.00
<input type="checkbox"/>	251	Task	Closed	Gantt Diagramm in Projektplan aktualisieren	Normal	Projektarbeit	0.50	1.00

6.10 Workflow

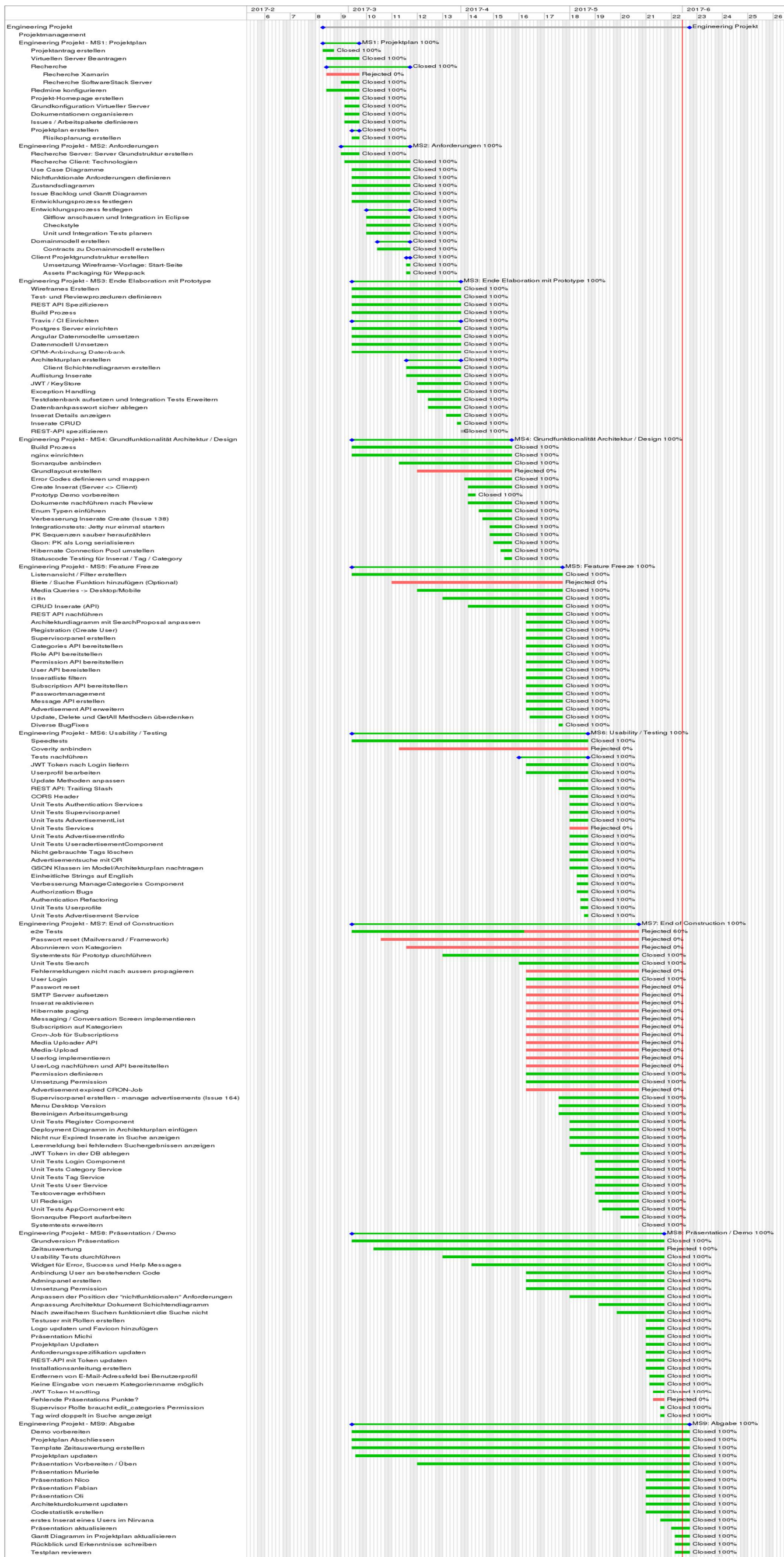
Jedes Arbeitspaket durchläuft einen festgelegten Workflow der in folgendem Flowchart ersichtlich ist:



Zur Veranschaulichung wird der Ablauf im Normalfall genauer beschrieben:

Ablauf	Status	Beschreibung
1.	New	Das Arbeitspaket wird erfasst, der benötigte Aufwand geschätzt und priorisiert
2.	TODO	Das priorisierte Arbeitspaket wird in den Sprint eingeplant
3.	Working	An dem Arbeitspaket wird aktuell gearbeitet
4.	To Review	Das Arbeitspaket ist vom Entwickler abgeschlossen und muss nun von einer weiteren Person überprüft werden
5.	Reviewing	Ein weiterer Entwickler hat das Arbeitspaket übernommen und prüft ob alle qualitätssichernden Massnahmen umgesetzt sind.
6.	Closed	Das Arbeitspaket ist erfolgreich abgeschlossen

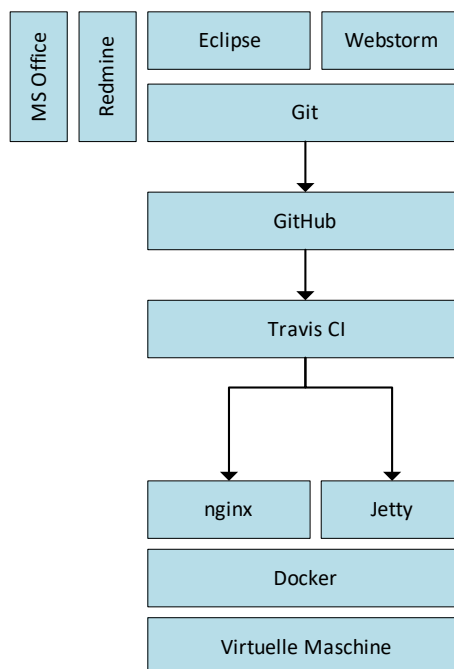
6.11 Gantt-Diagramm / Arbeitspakete



7. Infrastruktur

7.1 Entwicklungswerkzeuge

- VM für Server-Hosting
- REST Schnittstelle für Kommunikation mit der App
- Entwicklungssysteme: Webstorm¹, Eclipse Neon²
- Redmine³ als Projektmanagement-Tool
- Git/Github⁴ für die Versionskontrolle
- Travis⁵ für Continuous-Integration
- Jetty⁶ als App-Server
- PostgreSQL⁷ als Datenbank
- Docker⁸ für Deployment und einheitliches Testing
- Microsoft Office⁹ für das Erstellen der Dokumente und der Präsentation
- myBalsamiq¹⁰ für die Erstellung der Wireframes



¹ <https://www.jetbrains.com/webstorm/>

² <http://www.eclipse.org/neon/>

³ <http://redmine.com/>

⁴ <https://github.com/>

⁵ <https://travis-ci.org/>

⁶ <http://www.eclipse.org/jetty/>

⁷ <https://www.postgresql.org/>

⁸ <https://www.docker.com/>

⁹ <https://products.office.com/de-ch/home>

¹⁰ <https://www.mybalsamiq.com/>

7.2 Frameworks

Beide

- JWT¹¹ für die Authentisierung

Frontend

- Angular¹² für das Web-Frontend
- Jasmine¹³ für Frontend-Testing
- Karma¹⁴ als Test-Runner
- Webpack¹⁵ als Buildtool
- Embedded JavaScript¹⁶ als templating language

Backend

- Maven¹⁷ als Buildtool
- Java Spark¹⁸ für das Definieren der REST-Schnittstelle
- Hibernate¹⁹ als OR-Mapper
- JUnit²⁰ für Backend-Testing
- Checkstyle für Coding Guidelines
- Findbugs für statische Code Analyse

7.3 Weiteres

- Virtueller Server von DigitalOcean²¹

¹¹ <https://jwt.io/>

¹² <https://angular.io>

¹³ <https://jasmine.github.io/>

¹⁴ <https://karma-runner.github.io/1.0/index.html> (karma, 2017)

¹⁵ <https://webpack.github.io/>

¹⁶ <http://www.embeddedjs.com/> (Bitovi, 2017)

¹⁷ <http://maven.apache.org/>

¹⁸ <http://sparkjava.com/>

¹⁹ <http://hibernate.org/>

²⁰ <http://junit.org/junit4/>

²¹ <https://www.digitalocean.com/>

8. Qualitätsmassnahmen

Um einen hohen Qualitätsstandard sicherzustellen wurden folgende Massnahmen getroffen:

Massnahme	Zeitraum	Ziel
Arbeitsgruppen-Meeting	Wöchentlich	Synchronisation innerhalb der Arbeitsgruppe. Aushelfen bei Problemen damit keine Zeit verschwendet wird.
Sprint Review	Montags, gemäss Abschnitt 4.2 (Zeitliche Planung)	Austausch der beiden Arbeitsgruppen über Projektstand und Herausforderungen.
Sprint Planning	Montags, gemäss Abschnitt 4.2 (Zeitliche Planung)	Planung des nächsten Sprints. Verschieben der höchst priorisierten Arbeitspakete in den Sprint Backlog.
Code Reviews	Fortlaufend	Sicherstellung der Code Qualität und Einhaltung der Code Guidelines. Code Reviews werden jeweils vor dem mergen des Feature-Branche vorgenommen.
Automatisierte Tests	Fortlaufend	Sicherstellung, dass der Code auch noch nach einer Feature Implementation noch funktioniert

8.1 Dokumentation

Sämtliche Dokumente werden in ein eigenes Git-Repository (separat vom Code) eingchecked und deren Qualität wird durch Peer-Reviews sichergestellt.

8.2 Projektmanagement

Als Projektmanagement-Tool wird Redmine eingesetzt (siehe Abschnitt 6). Darin werden alle Arbeitspakete verwaltet und Aufgaben den Teammitgliedern zugeteilt. Jedes Teammitglied ist aufgefordert, die Zeit wahrheitsgemäss zu erfassen, was eine aufschlussreiche Reflexion der geleisteten Arbeit erlaubt.

8.3 Entwicklung

Der Source Code wird in einem privaten Git-Repository auf Github ²²eingchecked und dessen Qualität wird durch verschiedene Plugins und durch Peer-Reviews sichergestellt.

Plugins:

- FindBugs²³ (Java)
- CheckStyle²⁴ (Java)
- TSLint²⁵ (TypeScript)
- sass Lint²⁶ (SCSS)
- Codelyzer²⁷ (Angular Style Guide)

²² <https://github.com/>

²³ <http://findbugs.sourceforge.net/>

²⁴ <http://checkstyle.sourceforge.net/>

²⁵ <https://palantir.github.io/tslint/> (PalantirTech, 2017)

²⁶ <https://www.npmjs.com/package/sass-lint> (sass-lint, 2017)

²⁷ <http://codelyzer.com/>

8.3.1 Git Workflow

Der Git Workflow wird mit dem gitflow²⁸ Plugin durchgesetzt. Die Git-Repositories werden über Github abgeglichen. Commit Messages müssen aussagekräftig und auch für die anderen Teammitglieder verständlich formuliert werden (siehe auch Best Practices GitBook²⁹). Es gilt das Prinzip «commit early, commit often».

Vorgehen bei gitflow:

1. Ein neuer Feature-Branch wird erstellt. Der Name des Branches hat folgende Form: <Issue Nr. des Features>_<Kurze Beschreibung des Features>
2. Bei der Arbeit am Feature wird fortlaufend in den Feature-Branch committed. Nach Abschluss des Features wird das zuständige Issue auf den ToReview-Status gesetzt und mit einer Notiz versehen, die den Feature-Branch-Namen beinhaltet. Ebenfalls muss der Feature Branch in das Remote Repository gepusht werden.
3. Der Feature-Branch wird von einem Teammitglied überprüft und (falls der Code i.O. ist), vom überprüfenden Teammitglied in den Development-Branch merged. Der Development Branch wird bei jedem neuen Commit auf den Development Server deployed.
4. Am Ende des Sprints findet ein Release gemäss gitflow statt (siehe Grafik).
 - a. Das Release wird im Release-Branch vorbereitet
 - b. Der Release-Branch wird in den Master gemerged. Es gibt einen neuen Tag, welcher auf den produktiven Server deployed wird.

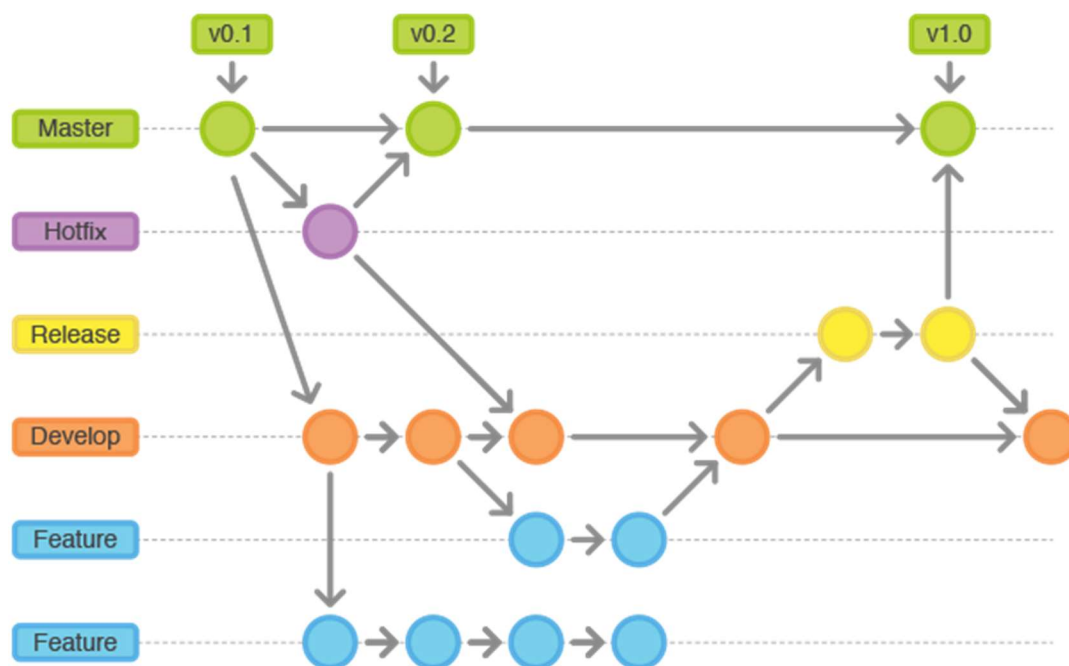


Abbildung 1: gitflow³⁰

²⁸ <http://jeffkreeftmeijer.com/2010/why-arent-you-using-git-flow/>

²⁹ <https://git-scm.com/book/en/v2/Distributed-Git-Contributing-to-a-Project>

³⁰ <https://blog.seibert-media.net/blog/2014/03/31/git-workflows-der-gitflow-workflow-teil-1/>

8.3.2 Code Reviews

Sämtlicher verfasster Code wird von mindestens einer Person aus dem gleichen Team überprüft, bevor er in Redmine in den «Closed» Zustand verschoben wird und der Feature-Branch in den develop-Branch gemerged wird. Ein spezielles Augenmerk haben folgende Punkte:

- Ist die Testabdeckung genügend?
- Ist die Architektur eingehalten?

8.3.3 Code Style Guidelines

Die Coding Guidelines werden im Buildprozess automatisch überprüft. Im Backend-Team wird ein angepasster Google Java Style Guide eingesetzt. Das Frontend-Team verwendet den Angular Styleguide.

8.3.4 Testing

Der gesamte Testplan ist im Dokument [Testing\Testplan.docx](#) beschrieben.