|  |
| --- |
| **Reflexionsbericht** |
| FST Projekt „Eventalizer“ Team 5 |
|  |
| Verbundstudium, Master of Science  Fortgeschrittene Softwaretechnologien  SS 2012 |
|  |
| **Matthias Beer, Alexander Benölken, Martin Garrels, Felix Schulze Mönking, Felix Wessel, Patrick Wiebeler** |
| **11.06.2012** |
| **Dozent: Prof. Dr. Mario Winter** |

1. Inhaltsverzeichnis

[I Inhaltsverzeichnis I](#_Toc327004568)

[II Abkürzungsverzeichnis I](#_Toc327004569)

[III Abbildungsverzeichnis II](#_Toc327004570)

[IV Tabellenverzeichnis II](#_Toc327004571)

[1 Überschrift 1 1](#_Toc327004572)

[1.1 Überschrift 2 1](#_Toc327004573)

[1.1.1 Überschrift 3 1](#_Toc327004574)

[2 Überschrift 1 2](#_Toc327004575)

[2.1 Projektplanung 2](#_Toc327004576)

[2.1.1 Eine Tabelle 2](#_Toc327004577)

[2.1.2 Eine Grafik 2](#_Toc327004578)

[3 Standardtabelle 3](#_Toc327004579)

[V Literaturverzeichnis IV](#_Toc327004580)

[VI Anhang 22](#_Toc327004581)

[VII Eigenständigkeitserklärung IV](#_Toc327004582)

1. Abkürzungsverzeichnis

|  |  |
| --- | --- |
| Abkürzung | Erläuterung / Definition |
| BPMN | **Business Process Model and Notation** |
| CRUD | **Create Read Update Delete** |
| CSS | **Cascading Style Sheets** |
| DAO | **Data Access Object** |
| EPK | **Ereignisgesteuerte Prozesskette** |
| GUI | **Graphical User Interface** |
| HTTP | **Hypertext Transfer Protocol** |
| IDE | **Integrated Development Environment** |
| JSP | **Java Server Pages** |
| MSDNAA | **Microsoft Developer Network Academic Alliance** |
| RUP | **Rational Unified Process** |
| UML | **Unified Modeling Language** |

1. Abbildungsverzeichnis

[Abbildung 1 Logische Architekur des Literatur-Informationssystems 2](#_Toc327004584)

1. Tabellenverzeichnis

[Tabelle 1 Eine Beispieltabelle 2](#_Toc327004586)

[Tabelle 2 Informationssystem 3](#_Toc327004587)

# Einleitung

Der vorliegende Reflexionsbericht entstand im Rahmen des berufsbegleitenden Masterstudiums im Verbundstudium der Fachhochschulen Köln und Dortmund. Ziel dieses Berichts ist die Reflexion des eigenen Verhaltens, Vorgehens und der Erfahrungen, die während des seminaristischen Softwareentwicklungsprojekts im Modul “Fortgeschrittene Softwaretechnologien” gewonnen wurden.

Inhaltlich richtet sich der Bericht an Leser mit Grundlagenkenntnissen im Bereich der Wirtschaftsinformatik und allgemeiner IT-Technik, insbesondere der Softwareentwicklung und des Projektmanagements. Es werden gewisse Fachtermini und Zusammenhänge vorausgesetzt. Die Verwendung von Anglizismen ist bei dieser Thematik unumgänglich.

Fokus dieses Berichts liegt zunächst auf eine Darstellung und Erläuterung der wesentlichen Entscheidungen, die zu Beginn des Projekts getroffen wurden und die das weitere Vorgehen im Projekt definiert haben. Die Analyse und Reflexion dieser Entscheidungen und die Ergebnisse der einzelnen Artefakte erfolgt im anschließenden Kapitel. Nach der Analyse erfolgt zum Abschluss des Berichts eine zusammenfassende und übergreifende Reflexion des Projekts, sowie eine Darstellung der weiteren Entwicklungsschritte und Potentiale der Plattform.

# Projektinitialisierung

## Projektidee

Zu Beginn des Projektes musste eine Projektidee ausgearbeitet werden. Mittels eines kurzen Brainstormings wurden drei Projektideen kurz schematisiert und beschrieben:

● Plattform zum Werkzeugverleih

Auf dieser Internetplattform sollten Werkzeuge ver- und gemietet werden können.

● Plattform zur Event-Organisation

Auf dieser Internetplattform sollten durch Benutzer (Gruppen-)Events eingestellt werden können, an denen andere Benutzer wiederum teilnehmen können.

● Browserspiel

Es sollte ein Mehrspieler-Spiel, welches im Browser läuft, entwickelt werden.

Die Idee, ein Browserspiel im Projekt zu entwickeln, wurde schnell verworfen, da weder das Know-how für Spieleprogrammierung noch die Zeit für eine - wenn auch nur prototypische - Implementierung vorhanden gewesen wäre. Die Idee, eine Plattform zur Event-Organisation zu entwickeln, auf der Benutzer Events einstellen und an Events anderer Benutzer teilnehmen können, fand bei allen Projektmitglieder schnell Gefallen. Diese Idee hätte eine Praxisrelevanz gehabt und stellte einen sinnvollen Anwendungsbereich da. Ferner würde dieses Projekt auch eine prototypische Implementierung zulassen. Daher erfolgte zeitnah die Festlegung auf diese Projektidee. Die Basis hierfür bildete jedoch eine noch sehr rudimentäre Vorstellung, bei welcher noch genügend Interpretationsspielraum in vielen Themenbereichen vorhanden war.

## Aufgabenverteilung und –durchführung

Zu Beginn des Projektes wurde diskutiert, wie die Aufgabenverteilung aussehen sollte. Im Rahmen dieser Diskussion wurde gemeinschaftlich entschieden, dass es keine auf die Projektmitglieder aufgeteilten Aufgabenbereiche geben soll und es Ziel des Projektes sein sollte, dass jeder möglichst in alle Bereiche des Softwareentwicklungsprozesses beteiligt ist. Um dem gerecht zu werden sollten die Aufgaben gleichmäßig aufgeteilt werden und jedes Teammitglied sollte in möglichst jedem Aufgabenbereich der Softwareentwicklung, d.h. angefangen bei der Anforderungsermittlung, über die Softwarespezifizierung, die Architekturkonzeption, den Entwurf, bis hin zu der Implementierung und der Qualitätssicherung einmal tätig werden.

## Entscheidungskompetenz

Durch das Fehlen eines Auftraggebers wurde zu Beginn des Projektes auf einen Projektleiter, der die entsprechende Kommunikation zwischen Auftraggeber und der Projektgruppe übernommen hätte, verzichtet. Innerhalb der Projektgruppe wurden keine Hierarchieeben aufgebaut, so dass alle Projektmitglieder in der Entscheidungsfindung gleichberechtigt waren. Im Rahmen der Umsetzung einzelner Artefakte war der jeweilige Bearbeiter in seiner Entscheidung frei. Sofern eine Entscheidung Auswirkungen auf andere Artefakte hat, so wurde diese Entscheidungen im Jour fixe vorgestellt. Grundsätzliche und für den weiteren Projektverlauf betreffende Entscheidungen wurden grundsätzlich im Jour fixe vom gesamten Team mehrheitlich entschieden.

## Jour fixe

Beim Kick-off des Projektes wurde ein Jour fixe, d.h. ein fester, wöchentlicher Termin für das Treffen des Projektteams festgelegt. Der Mittwoch wurde als Tag ausgewählt, an dem aktuelle Themen besprochen, Ergebnisse vorgestellt, Entscheidungen getroffen und weitere Aufgaben verteilt wurden. Neben dem Mittwoch wurde festgelegt, sich bei Bedarf auch freitags zu treffen. Für Projektmeetings wurde ein Büroraum mit Beamer, Flipchart, etc. reserviert. Die Dauer der Treffen bewegte sich, in Abhängigkeit von der Anzahl der Themen bzw. Entscheidungen oder ob innerhalb dieser Treffen Ergebnisse vorgestellt, ausgearbeitet und finalisiert wurden, zwischen 30 Minuten und drei Stunden. Jedes Meeting wurde dabei protokolliert, um auch fehlenden Mitgliedern die Möglichkeit zu bieten, die besprochenen Themen, die Ergebnisse, sowie die Entscheidungen nachverfolgen zu können.

## Infrastruktur

### Dateiaustauschplattform

Arbeitsergebnisse sollten von Beginn an zeitgemäß über einen zentralen Server verwaltet und verteilt werden. Von besonderer Bedeutung war, dass alle Projektmitglieder jederzeit Zugriff auf alle Dokumente der anderen Projektmitglieder haben sollen. Es wurden das kostenlose Quellcode-Repository GitHub[[1]](#footnote-2), Google Docs[[2]](#footnote-3) sowie der Cloudspeicherdienst Dopbox[[3]](#footnote-4) zur Auswahl gestellt.

Über den verwendeten Dienstleister sollten Office Dokumente und UML-Diagramme, aber auch Quellcode und beliebige weitere Dokumenttypen ausgetauscht werden. Zudem sollte gerade für Quellcode-Dateien die Synchronisierung kontrolliert erfolgen, also nicht automatisiert nach einem Speichervorgang. Bei genauerer Betrachtung der Anbieter konnte die Auswahl folgendermaßen vorgenommen werden:

Google Docs bietet einen gut funktionierenden Online-Editor für Office-Dokumente, an dem mehrere Bediener gleichzeitig arbeiten können. Problematisch ist der Im-/Export von Dokumenten insbesondere in Bezug auf die Formatierung. Zudem müssen alle Projektmitglieder für jede Datei, die sie lesen oder bearbeiten können sollen, explizit über einen erforderlichen Google-Account für diese berechtigt werden. Alternativ hätte die Datei für anonymes lesen und bearbeiten freigegeben und der entsprechende Dokumentenlink verteilt werden müssen. Ein Anonymes bearbeiten ist aus Datenschutzgründen und Schutz vor Vandalismus nicht gewünscht gewesen. Das Verteilen von Dokumentenlinks erschwert die Übersicht und das Handling der zahlreichen Dokumente und Artefakte.

Ein Dropbox-Verzeichnis bietet hinsichtlich des Hinzufügen und Ändern von Dateien den größten Komfort. Es gibt keine Einschränkung bezüglich des Dateiformats und die Synchronisierung mit dem Online-Speicher erfolgt unmittelbar mit dem Speichern der Datei im lokalen Dateisystem. Dies stellt sich bei der Bearbeitung von Quellcode, bei dem mehrere Dateien nur parallel (z.B. eine fertige Komponente) eingestellt werden sollen oder Änderungen verworfen werden sollen, als Nachteil heraus.

GitHub ist ein Webhoster für ein originäres Repository für Softwareentwicklungsprojekte. Die Verwaltung beliebiger Dateien erfolgt in einem so genannten Branch. Neue Dateien müssen manuell hinzugefügt werden und Änderungen manuell bestätigt werden. Als lokales GUI-basiertes Synchronisationswerkzeug wird zur Unterstützung TortoiseGit[[4]](#footnote-5) verwendet. Damit können Änderungskonflikte in Text-Dateien leicht zusammengeführt werden. Sofern innerhalb der Textdatei die Änderungen in verschiedenen Bereichen vorgenommen wurden, erfolgt eine automatische Zusammenführung. Nur bei Konflikten in gleichen Bereichen in der Textdatei muss ein manueller Eingriff erfolgen, die Änderungen werden dabei von TortoiseGit besonders hervorgehoben. Bei binären Dateien steht eine solche Funktion nicht zur Verfügung. Durch das Repository wird in diesem Fall die Synchronisierung der Änderung in das Online-Verzeichnis unterbunden. Dieser Konflikt muss durch einen manuellen Eingriff lokal aufgelöst werden.

Für die Verwaltung der Arbeitsergebnisse wurde das Tool GitHub ausgewählt. Dieses Tool kombiniert die Unterstützung beliebiger Dateiformate mit der vollen Benutzerkontrolle bei Synchronisationsvorgängen. Der Dropbox wurde aufgrund der eingeschränkten Nutzbarkeit für die Speicherung von Quellcode (gezieltes Publizieren eines Quellcode-Stand vs. automatischer Upload nach dem Speichern) abgelehnt. Bei paralleler Bearbeitung von Quellcode hätten durch versehentliche Speichervorgänge unnötig viele Synchronisierungskonflikte auftreten können. Google Docs wurde auf Grund der nur eingeschränkten Unterstützung von wenigen Dateitypen ausgeschlossen.

### Modellierungswerkzeug

Um UML-Modellierungs- und Quellcodeartefakte synchron zu entwickeln und homogen zu speichern, wurde zu Beginn des Projektes ein einheitliches Modellierungswerkzeug ausgewählt. Mit diesem Werkzeug sollte ein automatisierter Abgleich zwischen den Artefakten stattfinden können, um zum einen Fehler schneller erkennen zu können und zum anderen die Arbeit zu vereinfachen. Als UML-Modellierungswerkzeug wurde die kostenfreie Open-Source-Software Modelio aufgrund der guten Integration in das ausgewählte Softwareentwicklungswerkzeug und der intuitiven Bedienung gewählt. Der ausführliche Auswahlprozess ist in dem angehängten Dokument „Auswahl des UML-Modellierungswerkzeug“[[5]](#footnote-6) beschrieben.

Die Oberflächenmodellierung sollte zunächst möglichst einfach erfolgen, so dass zu diesem Zweck Microsoft PowerPoint[[6]](#footnote-7) verwendet werden soll. Dadurch soll die Einarbeitung in ein spezielles MockUp- oder Oberflächenmodellierungs-Werkzeug eingespart werden.

### Softwareentwicklungswerkzeug

Die Entwicklung der Software soll in Java erfolgen, als IDE soll Eclipse[[7]](#footnote-8) verwendet werden. Dies ist die einzige Plattform/Entwicklungsumgebung, bei der im Projektteam bereits Vorwissen vorhanden war. Weiterhin wurde auch die Roo Shell benutzt. Diese kann sowohl nativ als eigenständige Kommandoshell als auch per Plug-In in die Eclipse-IDE importiert werden.[[8]](#footnote-9)

## Vorgehensmodell

Für das Softwareprojekt sollte ein weit verbreitetes und etabliertes Vorgehensmodell verwendet werden. Bei der Auswahl möglicher Vorgehensmodelle wurde auf eine Auswahl etablierter Modelle aus dem Studium zurückgegriffen: Wasserfallmodell, V-Modell, Scrum oder der Rational Unified Process (RUP).

Das sequenzielle Wasserfallmodell sollte in diesem Projekt nicht verwenden. Aufgrund des sehr strikten Projektablaufs fehlt die notwendige Flexibilität, zumal die Anforderungen und der Umfang zu Beginn des Projekts noch nicht vollständig bekannt waren. Das Wasserfallmodell ist besser für Projekte mit klar definierten Anforderungen verwendbar, weil ein Rücksprung in eine vorherige Phase nicht bzw. nur eingeschränkt möglich ist.[[9]](#footnote-10)

Aufgrund der klaren Strukturen, der dadurch guten Planbarkeit und der umfangreichen Spezifikation der Artefakte ist das V-Modell bzw. V-Modell XT als Vorgehensmodell für dieses Projekt besser geeignet. Jedoch erfolgen bei diesem flexiblen Vorgehensmodell intensive Zusammenarbeit und Abstimmungen mit dem - im Projektseminar nicht existierenden - Auftraggeber. Ebenso fehlte im Projekt die entsprechende Ausschreibung und Anforderungsspezifikation durch den Auftraggeber, welches häufig bei dem Vorgehen nach dem V‑Modell XT bereits vorliegt bzw. in der Analysephase berücksichtigt wird.[[10]](#footnote-11) Aus diesen beiden Gründen wurde sich ebenfalls vom V-Modell bzw. V-Modell XT als Vorgehensmodell für dieses Projekt distanziert.

Ein agiles Vorgehensmodell erschien als deutlich geeigneter. Es wurden die beiden Vorgehensmodelle Scrum und RUP näher betrachtet:

Ein Vorgehen nach Scrum ist äußert flexibel auf Änderungen von Anforderungen durch entsprechende Dokumentation und Priorisierung im Produkt-Backlog. Ebenfalls ist Scrum sehr gut für kleine Entwicklungsteams geeignet und durch die kurzen Sprints können sehr schnell erste Ergebnisse vorgestellt werden.[[11]](#footnote-12) Nachteil dieses Vorgehensmodell und damit Ausschlusskriterium für die Verwendung in diesem Projekt ist, dass de facto ein Projektmitglied vollständig als Scrum-Master ausgelastet ist und die Planung und Koordination übernimmt.[[12]](#footnote-13) Ferner war es während der Sprints nicht möglich die täglichen Statusmeeting (Daily Scrums) zeitlich einzurichten. Scrum benötigt ein kleines Projektteam was nahezu 100% der zur Verfügung stehenden Zeit für das Projekt aufwenden kann. Abweichungen eines reinen Vorgehens nach Scrum sind dabei verpönt und werden als Scrumbut verschmäht.[[13]](#footnote-14)

Der Rational Unified Process ist in Prozessphasen gegliedert, die iterativ durchlaufen werden. Aufeinanderfolgende Phasen können parallel bearbeitet werden. Darüber hinaus wird ein Großteil der Spezifikationen noch vor der Umsetzung vorgenommen. Für jede Projektphase sind bestimmte Artefakte definiert, in denen die Arbeitsergebnisse festgehalten werden. Bei diesem agilen Vorgehen erfolgen die Spezifikation und die Umsetzung inkrementell. Dies erfordert viel Erfahrung der Projektmitglieder im Bereich der Teamarbeit miteinander.[[14]](#footnote-15) RUP erschien aus dieser Sicht ideal, da durch dieses Vorgehensmodell alle Projektmitglieder den gesamten Softwareentwicklungsprozess erfahren können. Gleichzeitig bietet RUP die notwendige Flexibilität auf Änderungen der Anforderungen zu reagieren. Ferner erschien der hohe Formalisierungsgrad für ein Studienprojekt, bei dem die Arbeitsergebnisse dokumentiert werden sollen, ideal.

## Projektplanung

Wie in Kapitel 2.3 „Entscheidungskompetenz“, Seite 3, bereits erwähnt, wurde zu Beginn festgelegt, dass es innerhalb dieses Projektseminars keinen Projektleiter geben sollte bzw. dass diese Rolle nicht durch ein einzelnes Projektmitglied besetzt werden sollte. Die zugehörigen unterstützenden Aktivitäten und Aufgaben sollten durch ein Teammitglied zusätzlich zu den „normalen” Projektaufgaben bzw. Softwareentwicklungsaktivitäten bearbeitet und mit erledigt werden.

Die grobe Projektplanung sollte dabei in Microsoft Projekt[[15]](#footnote-16) erfolgen und das Vorgehensmodell des Rational Unified Process berücksichtigen. Zur Arbeitserleichterung sollte die Feinplanung, d.h. die Steuerung und Dokumentation aller Aufgaben bzw. Aufgabenpakete, inklusive Bearbeiter und Erledigungszeitpunkt in einer auf Microsoft Excel[[16]](#footnote-17) basierende Aufgabenliste erfolgen.

## RUP-Artefakte

Zu Beginn des Projektes wurde die für den Rational Unified Process relevanten Artefakte identifiziert und hinsichtlich der Umsetzung innerhalb des Projektseminars bewertet.[[17]](#footnote-18)

### Visions-Dokument

Das Visions-Dokument soll zum einen das Resultat der Analyse des Problemfeldes sein und zum anderen die Anforderungen der Stakeholder, Kunden oder Endanwender enthalten. Da diese beiden Aspekte sowohl durch das Projekt-Exposé, als auch durch das Lastenheft abgedeckt werden, wurde im Team entschieden, dieses Artefakt nicht zu erstellen bzw. nicht weiter zu berücksichtigen.

### Risikoanalyse

Die Risikoanalyse wird zur Identifikation und Bewertung der Projektrisiken eingesetzt, um präventiv Maßnahmen zur Risikovermeidung ergreifen oder das Risiko steuern zu können. Ein Risiko wird dabei nach der Eintrittswahrscheinlichkeit und der Auswirkung auf das Projekt, in Bezug auf die Faktoren Zeit, Ressourcen und Inhalt/Umfang bewertet. Um diesem auch sehr praxisrelevanten und -nahen Aspekt zu berücksichtigen, einigte man sich darauf, diese Artefakte zu erstellen, um die identifizierten Risiken, deren Auswirkungen sowie die möglichen präventiven Maßnahmen auch für die Projektplanung zu berücksichtigen.

### Grobe Projektplanung

Einstimmig wurde im Team entschieden, dass das die grobe Projektplanung als Artefakt unverzichtbar ist und daher erstellt werden soll. Die grobe sowie auch feine Projektplanung ist separat in den Kapiteln 2.7 „Projektplanung“, Seite 8, und 3.5 „Umsetzung der Projektplanung“, Seite 10, beschrieben, daher wird an dieser Stelle nicht weiter auf diese eingegangen.

### Kosten-/Nutzenabschätzung

Ohne eine Kosten-/Nutzenabschätzung ist es fast unmöglich Finanzinvestoren zur Unterstützung des Projekts mit Venture Capital zu gewinnen.[[18]](#footnote-19) Aus diesem Grund und zur Überprüfung, ob und unter welchen Rahmenbedingungen ein wirtschaftlicher Betrieb von der Plattform Eventalizer möglich ist bzw. gewesen wäre, wurde im Team festgelegt, dass eine Kosten-/Nutzenabschätzung erfolgen soll.

### Anwendungsfalldiagramm

Das Anwendungsfalldiagramm, welches auch als Use Case Modell bezeichnet wird, sollte, einheitlich beschlossen, definitiv umgesetzt werden. Es wurde aber nicht als eigenes Artefakt erstellt, sondern in das Lastenheft unter dem Kapitel “Funktionale Anforderungen” integriert.

### Definition des Projektziels und Abgrenzung

### Überblick über Problembereich und Anforderungen

### Stakeholder-Übersicht

### Szenarien

### Überblick über die zu erbringenden Leistungen

### Begriffslexikon/Glossar

### Lastenheft

### Pflichtenheft

### Anwendungsarchitektur

### Designmodell

# Projektdurchführung

Umsetzung der Projektidee

Umsetzung der Aufgabenverteilung und –durchführung

## Umsetzung der Entscheidungskompetenz

## Umsetzung der Jour fixe

## Umsetzung der Infrastruktur

## Umsetzung des Vorgehensmodell

## Umsetzung der Projektplanung

## Umsetzung der RUP-Artefakte

### Visionsdokument

### Risikoabschätzung

### Kosten-/Nutzenabschätzung

### Anwendungsfalldiagramm

### Szenarien

### Begriffslexikon/Glossar

### Lastenheft

#### Produktfunktionen

#### Domänenklassendiagramm

#### Abnahmekriterien

### Pflichtenheft

#### Produktfunktionen

#### Entitätsklassendiagramm

#### Aktivitätsdiagramm

#### Abnahmekriterien

### Anwendungsarchitektur

### Design Modell

# Projektabschluss

## Zusammenfassung

## Ausblick

# Überschrift 1

## Projektplanung

### Eine Tabelle

| Szenario | Ausführungszeit in Sekunden | | Faktor |
| --- | --- | --- | --- |
|  | FileMaker | PostgreSQL |  |
|  |  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |  |
|  |  |  |
|  |  |  |

Tabelle 1 Eine Beispieltabelle[[19]](#footnote-20)

### Eine Grafik

|  |
| --- |
| architektur.png  Abbildung 1 Logische Architekur des Literatur-Informationssystems[[20]](#footnote-21) |

Hinweis: Jede Grafik besteht aus einer unsichtbaren Tabelle:

1. Zeile das Bild
2. Zeile die Beschriftung

# Standardtabelle

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Informations-empfänger | Art der Information | Verantwort-licher | Informations-zeitpunkt | Kommunikations-kanal |
| Projektgruppe | Weiterleitung von wichtigen Informationen von externen Quellen | Projektleitung | Je nach Bedarf | Schriftlich per Mail |
| Projektgruppe | Organisatorische Planungen | Projektleitung | Je nach Bedarf | Schriftlich per Mail |
| Projektcoach | Statusmeldungen | Projektleitung | Wöchentlich | Schriftlich in der eLearning-Plattform Ilias |
| Auftraggeber | Mitteilung der Arbeitsergebnisse (z. B. Detailkonzept) | Projektleitung | Zu den entsprechenden Meilensteinen | Schriftlich per Mail |
| Auftraggeber | Organisatorische Planungen | Projektleitung | Je nach Bedarf | Schriftlich per Mail |
| Projektleitung | Anfragen, Probleme, etc. | Projektcoach, Projektgruppe, Auftraggeber | Je nach Bedarf | Schriftlich per Mail oder mündlich |

Tabelle 2 Informationssystem

1. Literaturverzeichnis

**Schatten, A.; Biffel, S. et al. (2010):**  
Schatten, A.; Biffel, S.; Demolsky, M.; Gotischa-Franta, E.; Östreicher, Th.; Winkler, D.: Best Practice Software-Engineering. 1. Aufl., Heidelberg 2010.

1. URL-Verzeichnis

**[Kaak, J. (2012):]**Strategische Partner für den Start, http://www.mittelstandswiki.de/wissen/Venture\_Capital, Überprüft am 28.05.2012.

**o.V. (2012):**  
SpringSource Tool Suite - The Best Development Tool for Enterprise Java, http://www.springsource.com/developer/sts; Überprüft am 07.06.2012.

**Schwaber, K. (2012):**  
ScrumButs and Modifying Scrum, http://www.scrum.org/scrumbut, Überprüft am 04.06.2012.

1. Anhang

[A 1. Projektmanagement A-Fehler! Textmarke nicht definiert.](#_Toc225785803)

[A 1.1. Projektstrukturplan A-**Fehler! Textmarke nicht definiert.**](#_Toc225785804)

[A 1.2. Arbeitspakete A-**Fehler! Textmarke nicht definiert.**](#_Toc225785805)

[A 1.3. Risikoanalyse A-**Fehler! Textmarke nicht definiert.**](#_Toc225785806)

[A 1.4. Balkendiagram A-**Fehler! Textmarke nicht definiert.**](#_Toc225785807)

[A 1.5. Netzplan A-**Fehler! Textmarke nicht definiert.**](#_Toc225785808)

[A 1.6. Statusberichte A-**Fehler! Textmarke nicht definiert.**](#_Toc225785809)

[A 1.6.1. 07. Februar 2009 A-**Fehler! Textmarke nicht definiert.**](#_Toc225785810)

[A 1.6.2. 14. Februar 2009 A-**Fehler! Textmarke nicht definiert.**](#_Toc225785811)

[A 1.6.3. 21. Februar 2009 A-**Fehler! Textmarke nicht definiert.**](#_Toc225785812)

[A 1.6.4. 28. Februar 2008 A-**Fehler! Textmarke nicht definiert.**](#_Toc225785813)

[A 1.6.5. 10. März 2009 A-**Fehler! Textmarke nicht definiert.**](#_Toc225785814)

[A 1.6.6. 18. März 2009 A-**Fehler! Textmarke nicht definiert.**](#_Toc225785815)

[A 1.6.7. 25. März 2009 A-**Fehler! Textmarke nicht definiert.**](#_Toc225785816)

[A 2. Protokolle A-Fehler! Textmarke nicht definiert.](#_Toc225785817)

[A 2.1. 02.02.2009 – Interner Kick-Off 2.Phase A-**Fehler! Textmarke nicht definiert.**](#_Toc225785818)

[A 2.2. 04.02.2009 – Präsentation 1.Phase und Ausblick 2.Phase A-**Fehler! Textmarke nicht definiert.**](#_Toc225785819)

[A 2.3. 05.02.2009 – 1. vorort Test A-**Fehler! Textmarke nicht definiert.**](#_Toc225785820)

[A 2.4. 05.02.2009 – Kick-Off 2.Phase A-**Fehler! Textmarke nicht definiert.**](#_Toc225785821)

[A 2.5. 12.02.2009 – Meeting im ISFH A-**Fehler! Textmarke nicht definiert.**](#_Toc225785822)

[A 2.6. 19.02.2009 – Statusmeeting / Testkonzept A-**Fehler! Textmarke nicht definiert.**](#_Toc225785823)

[A 2.7. 26.02.2009 – Testtag & Statusmeeting A-**Fehler! Textmarke nicht definiert.**](#_Toc225785824)

[A 2.8. 04.03.2009 – Installation der Testversion A-**Fehler! Textmarke nicht definiert.**](#_Toc225785825)

[A 2.9. 12.03.2009 – Statusmeeting A-**Fehler! Textmarke nicht definiert.**](#_Toc225785826)

[A 2.10. 12.03.2009 - Testinstallation im ISFH A-**Fehler! Textmarke nicht definiert.**](#_Toc225785827)

[A 2.11. 17.03.2009 – Erste Testversion mit PostgreSQL A-**Fehler! Textmarke nicht definiert.**](#_Toc225785828)

[A 2.12. 19.03.2009 – Installation der finalen Version A-**Fehler! Textmarke nicht definiert.**](#_Toc225785829)

[A 2.13. 23.03.2009 – Statusmeeting A-**Fehler! Textmarke nicht definiert.**](#_Toc225785830)

[A 2.14. 23.03.2009 – Einführungspräsentation im ISFH A-**Fehler! Textmarke nicht definiert.**](#_Toc225785831)

[A 3. Wichtige Korrespondenz A-Fehler! Textmarke nicht definiert.](#_Toc225785832)

[A 3.1. Benutzerrechte A-**Fehler! Textmarke nicht definiert.**](#_Toc225785833)

[A 3.2. Lastenheft A-**Fehler! Textmarke nicht definiert.**](#_Toc225785834)

[A 3.3. Migrationsvorbereitung A-**Fehler! Textmarke nicht definiert.**](#_Toc225785835)

[A 3.4. Test-Server A-**Fehler! Textmarke nicht definiert.**](#_Toc225785836)

[A 3.5. Klärung von Anforderungen A-**Fehler! Textmarke nicht definiert.**](#_Toc225785837)

[A 3.6. Test der Datenbanken A-**Fehler! Textmarke nicht definiert.**](#_Toc225785838)

[A 3.7. Test nach Änderung der Datenbank A-**Fehler! Textmarke nicht definiert.**](#_Toc225785839)

[A 3.8. URL der Anwendung A-**Fehler! Textmarke nicht definiert.**](#_Toc225785840)

[A 3.9. Installation Backup A-**Fehler! Textmarke nicht definiert.**](#_Toc225785841)

[A 4. Präsentationen A-Fehler! Textmarke nicht definiert.](#_Toc225785842)

[A 4.1. 04.02.2009 - Zwischenpräsentation A-**Fehler! Textmarke nicht definiert.**](#_Toc225785843)

[A 4.2. 23.03.2009 - Einführung im ISFH A-**Fehler! Textmarke nicht definiert.**](#_Toc225785844)

[A 4.3. 26.03.2009 – Projektabschlusspräsentation im ISFH A-**Fehler! Textmarke nicht definiert.**](#_Toc225785845)

[A 5. Arbeitsergebnisse A-Fehler! Textmarke nicht definiert.](#_Toc225785846)

[A 5.1. Lastenheft A-**Fehler! Textmarke nicht definiert.**](#_Toc225785847)

[A 5.2. Konzept zur Qualitätssicherung A-**Fehler! Textmarke nicht definiert.**](#_Toc225785848)

[A 5.3. Benutzer Handbuch A-**Fehler! Textmarke nicht definiert.**](#_Toc225785849)

[A 5.4. Technisches Handbuch A-**Fehler! Textmarke nicht definiert.**](#_Toc225785850)

1. Überschrift Anahng 1
   1. Anhang 2

Mehr als zwei Ebenen gibt es im Anhang nicht!

1. Anhang 2
2. Eigenständigkeitserklärung

“Hiermit versichern wir an Eides statt, dass diese Arbeit selbständig und ohne unzulässige fremde Hilfe erstellt worden ist, keine anderen Quellen und Hilfsmittel als die angegebenen benutzt und die Stellen der Arbeit, die anderen Werken dem Wortlaut oder dem Sinn nach entnommen sind, in jedem einzelnen Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht haben. Das gleiche gilt auch für eingefügte Zeichnungen, Karten Skizzen und Darstellungen.”

Münster, den 11.06.2012.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Matthias Beer | Alexander Benölken | Martin Garrels |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Felix Schulze Mönking | Felix Wessel | Patrick Wiebeler |

1. Siehe https://github.com. [↑](#footnote-ref-2)
2. Siehe http://docs.google.com/?hl=de. [↑](#footnote-ref-3)
3. Siehe https://www.dropbox.com. [↑](#footnote-ref-4)
4. Siehe http://code.google.com/p/tortoisegit/. [↑](#footnote-ref-5)
5. Siehe Artefakte auf CD „Auswahl Modellierungswerkzeug.pdf“. [↑](#footnote-ref-6)
6. Siehe http://office.microsoft.com/de-de/powerpoint/. [↑](#footnote-ref-7)
7. Siehe www.eclipse.org/. [↑](#footnote-ref-8)
8. Vgl. o.V. (2012). [↑](#footnote-ref-9)
9. Vgl. Schatten, A.; Biffl, S. et al., Seite 48f. [↑](#footnote-ref-10)
10. Vgl. Schatten, A.; Biffl, S. et al., Seite 68. [↑](#footnote-ref-11)
11. Vgl. Schatten, A.; Biffl, S. et al., Seite 64f. [↑](#footnote-ref-12)
12. Dieses Aufgabenteilung sollte in diesem Projekt vermeiden werden, siehe Kap. 2.3 „Entscheidungskompetenz“, Seite 3. [↑](#footnote-ref-13)
13. Vgl. Schwaber, K. (2012). [↑](#footnote-ref-14)
14. Vgl. Schatten, A.; Biffl, S. et al., Seite 58ff. [↑](#footnote-ref-15)
15. Siehe http://www.microsoft.com/project/de/de/default.aspx. [↑](#footnote-ref-16)
16. Siehe http://office.microsoft.com/de-de/excel/. [↑](#footnote-ref-17)
17. Siehe dazu auch Besprechungsprotokoll vom 21.03.2012, Top 3. [↑](#footnote-ref-18)
18. Vgl. Kaack, J. (2012). [↑](#footnote-ref-19)
19. Vgl. Deutschkämer et al. (2008), Seite A-37. [↑](#footnote-ref-20)
20. Vgl. Microsoft Corp. (2008). [↑](#footnote-ref-21)