

Projekt: wt4u  
  
Projektplan

Andreoli Dario (dandreol@hsr.ch)

Schiepek Richard (rschiepe@hsr.ch)

Zahner Tobias (tzahner@hsr.ch)

# Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Änderung | Autor |
| 20.02.14 | 0.1 | Inititaldokument | dandreol  rschiepe  tzahner |
| 28.02.14 | 0.2 | Zeitplanung angepasst | rschiepe |
| 01.03.14 | 0.3 | Qualitätsmassnahmen  C# Conventions  Arbeitspakete | rschiepe |
| 01.03.14 | 0.4 | Einzelne Verbesserungen  Risikomanagement | dandreol |
| 03.03.14 | 0.5 | Vorabgabe Version | dandreol  rschiepe  tzahner |
| 05.03.14 | 0.6 | Tickets angepasst (Tipps per Mail von L.Bläser) | rschiepe |
| 05.03.14 | 0.7 | Neubeurteilung Risikomanagement | dandreol |
| 05.03.14 | 1.0 | Rechtschreibung und Grammatik. Kleine Anpassungen | dandreol  rschiepe |
| 06.03.14 | 1.1 | Anpassungen gemäss Review Projektplan | dandreol |
| 13.03.14 | 1.2 | Entfernen von zwei optionalen Meilensteinen gemäss Review 06.03.14 | rschiepe |
| 18.04.14 | 1.3 | Neubeurteilung der Risiken nach Iteration Elaboration E1 | dandreol |
| 01.04.14 | 1.4 | Anpassungen beim Testing | tzahner |
| 02.04.14 | 1.5 | Neubeurteilung der Risiken | dandreol |
| 07.04.14 | 1.6 | Anpassung Planung aufgrund Verschiebung Review Architektur/Design | dandreol |
| 18.04.14 | 1.7 | Neubeurteilung Risiken nach Construction C1 | dandreol |
| 01.05.14 | 1.8 | Neubeurteilung Risiken nach Construction C2 | dandreol |
| 19.05.14 | 1.9 | Aktuelle Version der Tickets eingefügt. Coding-Style-Guidelines erweitert um Gross- Kleinschreibung. | rschiepe |
| 27.05.14 | 2.0 | Neubeurteilung der Risiken nach Construction C3 und Transition T1 aus Excel eingefügt | dandreol |

# Inhalt

[Änderungsgeschichte 2](#_Toc388972855)

[Inhalt 3](#_Toc388972856)

[1. Einführung 5](#_Toc388972857)

[1.1 Zweck 5](#_Toc388972858)

[1.2 Gültigkeitsbereich 5](#_Toc388972859)

[1.3 Referenzen 5](#_Toc388972860)

[1.3.1 Glossar 5](#_Toc388972861)

[1.3.2 Literaturverzeichnis 5](#_Toc388972862)

[2. Projekt Übersicht 6](#_Toc388972863)

[2.1 Zweck und Ziel 6](#_Toc388972864)

[2.2 Lieferumfang 6](#_Toc388972865)

[2.3 Annahmen und Einschränkungen 6](#_Toc388972866)

[3. Projektorganisation 7](#_Toc388972867)

[3.1 Organisationsstruktur 7](#_Toc388972868)

[3.2 Verantwortlichkeiten 7](#_Toc388972869)

[3.3 Externe Schnittstellen 7](#_Toc388972870)

[3.4 Organigramm 7](#_Toc388972871)

[4. Management Abläufe 8](#_Toc388972872)

[4.1 Kostenvoranschlag 8](#_Toc388972873)

[4.2 Zeitliche Planung 8](#_Toc388972874)

[4.2.1 Phasen / Iterationen 8](#_Toc388972875)

[4.2.2 Meilensteine 9](#_Toc388972876)

[4.3 Besprechungen 9](#_Toc388972877)

[5. Risikomanagement 10](#_Toc388972878)

[5.1 Risiken 10](#_Toc388972879)

[5.1.1 Neubeurteilungen 11](#_Toc388972880)

[5.2 Umgang mit Risiken 13](#_Toc388972881)

[6. Arbeitspakete 13](#_Toc388972882)

[7. Infrastruktur 20](#_Toc388972883)

[7.1 Server-Architektur 20](#_Toc388972884)

[7.2 Entwicklungsumgebung 20](#_Toc388972885)

[7.3 Dokumentverwaltung 20](#_Toc388972886)

[8. Qualitätsmassnahmen 21](#_Toc388972887)

[8.1 Dokumentation 21](#_Toc388972888)

[8.2 Projektmanagement 21](#_Toc388972889)

[8.3 Entwicklung 21](#_Toc388972890)

[8.3.1 Vorgehen 21](#_Toc388972891)

[8.3.2 Unit Testing 21](#_Toc388972892)

[8.3.3 Code Reviews 21](#_Toc388972893)

[8.3.4 Code Style Guidelines 21](#_Toc388972894)

[8.4 Testen 23](#_Toc388972895)

[8.4.1 Unit Tests 23](#_Toc388972896)

[8.4.2 System Tests 23](#_Toc388972897)

[8.4.3 Usability Tests 23](#_Toc388972898)

[8.4.4 Web Browser 23](#_Toc388972899)

[8.4.5 Bug Tracking 24](#_Toc388972900)

# Einführung

## Zweck

Der Projektplan ist das wichtigste Planungsdokument unseres Projektes Work Time 4 You (wt4u), welches eine Webanwendung für die Verwaltung von Projektarbeitszeiten zum Ziel hat. Alle phasen- und iterationsübergreifenden Ziele, Meilensteine und Organisationsstrukturen werden darin festgelegt.

## Gültigkeitsbereich

Dieses Dokument hat für das ganze Projekt Gültigkeit und soll laufend aktualisiert werden.

## Referenzen

### Glossar

|  |  |
| --- | --- |
| Begriff | Beschreibung |
| MSDN | Das **Microsoft Developer Network (MSDN)** ist ein Informations- und Software-Angebot für Programmierer und Software-Architekten sowie Designer, die sich mit [Microsoft](http://de.wikipedia.org/wiki/Microsoft" \o "Microsoft)-Produkten und -Technologien beschäftigen |
| Visual Studio | **Visual Studio** ist eine von dem Unternehmen [Microsoft](http://de.wikipedia.org/wiki/Microsoft" \o "Microsoft) angebotene, integrierte Entwicklungsumgebung für verschiedene [Hochsprachen](http://de.wikipedia.org/wiki/H%C3%B6here_Programmiersprache" \o "Höhere Programmiersprache). Die aktuelle Version 12.0 („Visual Studio 2013“) unterstützt [Visual Basic .NET](http://de.wikipedia.org/wiki/Visual_Basic_.NET" \o "Visual Basic .NET), [C](http://de.wikipedia.org/wiki/C_%28Programmiersprache%29), [C++](http://de.wikipedia.org/wiki/C%2B%2B), [C++/CLI](http://de.wikipedia.org/wiki/C%2B%2B/CLI), [C++/CX](http://de.wikipedia.org/wiki/C%2B%2B/CX), [C#](http://de.wikipedia.org/wiki/C-Sharp), [F#](http://de.wikipedia.org/wiki/F-Sharp), [SQL Server](http://de.wikipedia.org/wiki/Microsoft_SQL_Server), [TypeScript](http://de.wikipedia.org/wiki/TypeScript) und [Python](http://de.wikipedia.org/wiki/Python_%28Programmiersprache%29), sowie [HTML](http://de.wikipedia.org/wiki/Hypertext_Markup_Language" \o "Hypertext Markup Language), [JavaScript](http://de.wikipedia.org/wiki/JavaScript) und [CSS](http://de.wikipedia.org/wiki/Cascading_Style_Sheets) für die Entwicklung von [Windows Store](http://de.wikipedia.org/wiki/Windows_Store" \o "Windows Store)-Apps. |
| C# | Ist eine vom [Softwarehersteller](http://de.wikipedia.org/wiki/Software" \o "Software) [Microsoft](http://de.wikipedia.org/wiki/Microsoft) im Rahmen seiner [.NET](http://de.wikipedia.org/wiki/.NET" \o ".NET)-Strategie entwickelte [Programmiersprache](http://de.wikipedia.org/wiki/Programmiersprache" \o "Programmiersprache). |
| Redmine | **Redmine** ist ein webbasiertes Projektmanagement-Tool. Es kann für Benutzer- und Projektverwaltung, Diskussionsforen, [Wikis](http://de.wikipedia.org/wiki/Wiki" \o "Wiki), zur Ticketverwaltung oder Dokumentenablage genutzt werden. Weltweit wird Redmine bei mehreren großen Projekten verwendet, unter anderem bei dem [Issue-Tracking-System](http://de.wikipedia.org/wiki/Issue-Tracking-System" \o "Issue-Tracking-System) der Programmiersprache [Ruby](http://de.wikipedia.org/wiki/Ruby_%28Programmiersprache%29" \o "Ruby (Programmiersprache)) oder dem Entwicklerportal von [TYPO3](http://de.wikipedia.org/wiki/TYPO3" \o "TYPO3). |

### Literaturverzeichnis

*Microsoft C# Coding Conventions*. (kein Datum). Abgerufen am 3. März 2014 von http://msdn.microsoft.com/en-us/library/ff926074.aspx

*Tag-Names für Kommentare*. (kein Datum). Abgerufen am 3. März 2014 von http://msdn.microsoft.com/en-us/library/5ast78ax.aspx

# Projekt Übersicht

“wt4u“ ist eine Webapplikation, mit welcher Arbeitszeiten und Projekte verwaltet werden können. Das Stundenerfassungstool soll für Mitarbeiter einfach zu benutzen sein und dem Projektleiter umfangreiche Auswertungen ermöglichen. Die Software soll mittels ASP.NET realisiert werden. Die zugehörigen Daten sollen auf einem Microsoft SQL Server gehalten werden.

Um die Anwendung einfach benutzbar zu machen, stehen unterschiedliche Rollen zur Verfügung. Dabei wird unter Projektleiter und Projektmitarbeiter unterschieden. Jeder Benutzer loggt sich mit einem persönlichen Account ein. Zusätzlich zur Projektverwaltung für Projektleiter soll der Arbeitgeber eine Möglichkeit zur Auswertung der gesamten Arbeitszeit seiner Mitarbeiter erhalten. Dieser Zugang bietet zusätzliche Möglichkeiten im Vergleich mit dem Zugang eines Projektleiters.

## Zweck und Ziel

Unser primäres Ziel ist natürlich der erfolgreiche Abschluss des Moduls “Software-Engineering 2 –Projekt“. Dies wollen wir erreichen indem wir unser Wissen des Stoffes der Module SE1 und SE2 anhand eines konkreten Projektes erweitern und festigen. Dabei sollen die verschiedenen Phasen eines Projektes durchlaufen und dabei auftretende Probleme gelöst werden.

Da sich alle Teammitglieder für Webentwicklung interessieren und erste Erfahrungen mit den für dieses Projekt notwendigen Technologien in den Modulen „Internettechnologien“, „.Net Technologien“ und „Datenbanksysteme 1“ sammeln konnten, wollen wir mit diesem Projekt weitere Erfahrungen sammeln.

Am Ende soll eine web-basierte Zeiterfassungssoftware, welche auch für grosse Projektgruppen einfach zu bedienen und sehr übersichtlich ist, als Ergebnis vorliegen.

## Lieferumfang

Die Software besteht aus 3 Ansichten:

* Eine Arbeitgeber-Ansicht, welche die Arbeitsstunden der einzelnen Mitarbeiter anzeigt und verwaltet. Diese wird benötigt um am Ende des Monats den Lohn für die einzelnen Mitarbeiter zu berechnen.
* Eine Projektleiter-Ansicht, welche Auswertungen und Statistiken zu seinem Projekt beinhaltet.
* Eine Mitarbeiter-Ansicht, mit welcher er einchecken kann, wenn er zu arbeiten beginnt und seine Arbeitszeit bei Bedarf einem Projekt zuweisen kann.

## Annahmen und Einschränkungen

Jedes Teammitglied soll ca. 9 Stunden pro Woche für das Projekt aufwenden. Insgesamt fallen somit rund 400 Stunden Arbeit an. Es wird von einer Projektdauer von 15 Wochen ausgegangen, obwohl die offizielle Semesterdauer nur 14 Wochen ist. Für dieses Projekt wird aber die dazwischenliegende Woche Ferien ebenfalls genutzt. Das Projekt muss bis am 30. Mai 2014 abgegeben werden.

# Projektorganisation

Unser Team besteht aus drei gleichgestellten Mitgliedern. Aus organisatorischen Gründen übernimmt Richard Schiepek die Rolle des Projektleiters. Jedes Teammitglied soll sich in allen Aufgabengebieten einbringen. Prof. Dr. Luc Bläser ist Betreuer des Projektes und steht dem Projektteam bei Fragen zur Seite und ist für die Bewertung der Reviews sowie des Gesamtprojektes verantwortlich.

Das Team trifft sich wöchentlich zu einer Teamsitzung, an der über das weitere Vorgehen, Pendenzen, Probleme und die Aufgabenteilung diskutiert wird. Dies findet jeweils am Donnerstag von 08:10 Uhr bis 9:10 Uhr statt.

## Organisationsstruktur

    
Dario Andreoli Richard Schiepek Tobias Zahner

## Verantwortlichkeiten

|  |  |  |
| --- | --- | --- |
| Name | E-Mail | Verantwortlichkeiten |
| Dario Andreoli | [dandreol@hsr.ch](mailto:dandreol@hsr.ch) | Arbeiten in allen Aufgabengebieten |
| Richard Schiepek | [rschiepe@hsr.ch](mailto:rschiepe@hsr.ch) | Arbeiten in allen Aufgabengebieten |
| Tobias Zahner | [tzahner@hsr.ch](mailto:tzahner@hsr.ch) | Arbeiten in allen Aufgabengebieten |

## Externe Schnittstellen

|  |  |  |
| --- | --- | --- |
| Name | E-Mail | Rolle |
| Prof. Dr. Luc Bläser | [luc.blaeser@hsr.ch](mailto:luc.blaeser@hsr.ch) | Projektbetreuer |

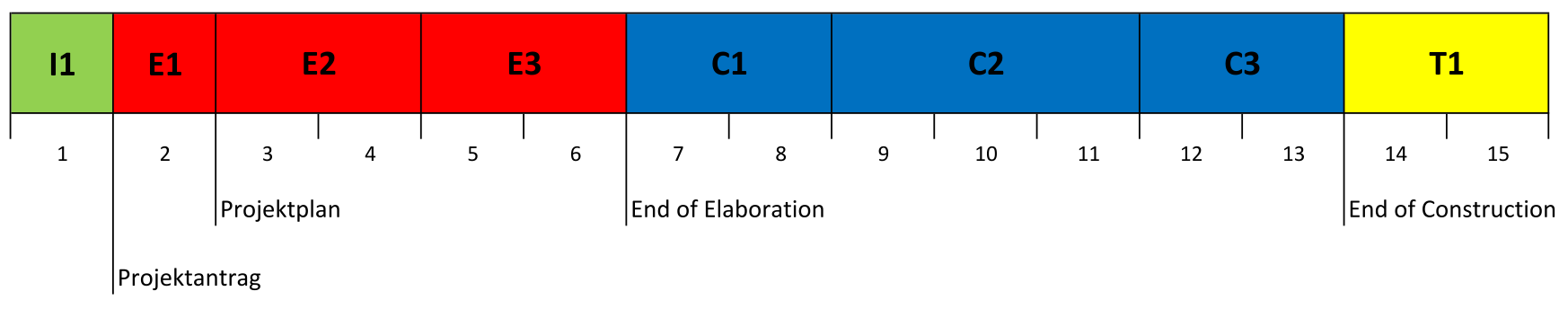
## Organigramm

# Management Abläufe

## Kostenvoranschlag

Das Projekt muss spätestens am 30. Mai 2014 abgeschlossen sein. Dazu stehen pro Woche ca. 9 Stunden zur Verfügung.

## Zeitliche Planung



### Phasen / Iterationen

|  |  |  |  |
| --- | --- | --- | --- |
| Iteration | Beschreibung | Ende | Dauer (in Wochen) |
| Inception I1 | Projektantrag und Vision erstellen. Beginn mit der Arbeit am Projektplan. | SW01 | 1 |
| Elaboration E1 | Projektplan erstellen | SW02 | 1 |
| Elaboration E2 | Vorbereiten und Konfiguration des Servers, Domainanalyse und Anforderungsspezifikation, Einarbeitung in ASP.NET, Architekturprototyp eingerichtet. | SW04 | 2 |
| Elaboration E3 | Einchecken- und Auschecken der Mitarbeiter grob implementiert, simples UI erstellen, Externes UI-Design | SW06 | 2 |
| Construction C1 | Ein- und Auschecken der Mitarbeiter vollständig implementiert (inkl. Login). Oberfläche grob designt und programmiert. | SW08 | 2 |
| Construction C2 | Views von Projektleiter und Arbeitgeber implementieren.  Umsetzung der Statistiken für Projektleiter und Arbeitgeber implementieren. | SW11 | 3 |
| Construction C3 | Optimierungen, Fehlerbehebung, Tests, evtl. optionale Arbeitspakete | SW13 | 2 |
| Transition T1 | Dokumentation, Präsentation, Abgabe | SW15 | 2 |

### Meilensteine

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Meilenstein | Name | Beschreibung | Produkte | Woche |
| MS1 | Projektplanung | Der Projektplan ist fertiggestellt, er beinhaltet u.a. Mitarbeitende und Arbeitspakete. | Projektplan | SW02 |
| MS2 | Anforderungsspezifikation und Domainanalyse | Use Cases und Domain Model sind fertiggestellt. | Anforderungsspezifikation, Domainanalyse | SW04 |
| MS3 | Ende Elaboration | Architekturprototyp fertiggestellt inkl. Tests. | Architekturprototyp | SW07 |
| MS4 | Architektur / Design | Erste lauffähige Version von wt4u vorhanden. Wichtigste Funktionen implementiert. | Architekturdesign, Architekturdokumentation | SW11 |
| MS5 | Schlusspräsentation / Abgabe | Präsentation durchgeführt. Sämtliche Dokumente wurden dem Betreuer übergeben. | Präsentation, Dokumentation,  Software-Paket | SW15 |

## Besprechungen

Vorgesehen ist ein Treffen aller Teammitglieder jeden Donnerstagvormittag. Dieses wird genutzt um das weitere Vorgehen, Pendenzen, Probleme und die Aufgabenteilung zu besprechen. Für alle Besprechungen wird ein Protokoll verfasst. Für das führen des Protokolls wird jeweils vor der Sitzung ein Teammitglied bestimmt.

|  |  |
| --- | --- |
| Sitzung | Donnerstag 08:10 Uhr (Dauer ca. 1h) |
| Teilnehmer | Dario Andreoli, Richard Schiepek, Tobias Zahner |
| Ort | HSR Rapperswil, 1.256a |

Bei Bedarf können zusätzliche Meetings einberufen werden.

# Risikomanagement

## Risiken

Ausschnitt aus dem Dokument „TechnischeRisiken.xlsx“:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Nr** | **Titel** | **Beschreibung** | **max. Schaden [h]** | **Eintrittswahrscheinlichkeit** | **Gewichteter Schaden** | **Vorbeugung** | **Verhalten beim Eintreten** |
| R1 | Datenbank | Kein Teammitglied hat schon einmal eine Datenbank auf einem Microsoft Server aufgesetzt. Auch beim Herstellen der Verbindung aus dem Programm können Probleme auftreten. | 10 | 40% | 4 | Man informiert sich mit diversen Tutorials aus dem Internet und bei Mitstudenten. | Man nimmt zuerst das Internet und die Mitstudenten zu Hilfe. Falls das nicht hilft, wendet man sich an den Betreuer. |
| R2 | Webserver | ASP.NET muss auf dem Server zum Laufen gebracht werden. Dazu wird wohl ein Service gebraucht. | 4 | 30% | 1.2 | Man informiert sich mit diversen Tutorials aus dem Internet. | Man nimmt zuerst das Internet zu Hilfe. Falls das nicht hilft, wendet man sich an den Betreuer. |
| R3 | ASP.NET | Es ist keine Erfahrung mit ASP.NET im Team vorhanden. Probleme mit der Architektur und Funktionsweise können auftreten. | 40 | 80% | 32 | Einlesen und diverse Tutorials durcharbeiten. | Google, Teammitglieder und Betreuer fragen. |
| R4 | Falsche Aufwand-Schätzung | Die Arbeitspakete wurden falsch eingeschätzt. | 30 | 90% | 27 | Detailliertes Planen und Abschätzen. | Mehraufwand betreiben, falls es soweit kommt. |
| R5 | Versionierung / Unit Testing | Keines der Teammitglieder konnte bereits Erfahrungen mit Visual Studio Online sammeln. Entsprechend können Probleme damit auftreten. | 50 | 80% | 40 | Frühzeitige Nutzung von Visual Studio Online. Studieren der Anleitungen. | Man nimmt zuerst das Internet zu Hilfe. Falls das nicht hilft, wendet man sich an den Betreuer. |
| R6 | Security | Da Benutzerverwaltung für Applikation notwendig, müssen Passwörter etc. verschlüsselt abgelegt werden. Auch hier ist noch keine Erfahrung im Team vorhanden. | 30 | 60% | 18 | Einarbeitung in Materie, Benutzung von vorhandenen Lösungen | Implementierung ohne Verschlüsselung |
| R7 | Diagramme Auswertung | Diagramme mittels HTML, CSS, JS wurde noch von keinem Teammitglied erstellt. Dies benötigt ebenfalls Einarbeitungszeit und Probleme können auftreten. | 30 | 70% | 21 | Verwendung von bereits vorhandenen Bibliotheken (z.B.Chart.JS: http://www.chartjs.org/) | Verzicht auf grafische Auswertung |
| R8 | Falsches Risiko-Management | Nicht erwartete Probleme können auftreten und somit das Projekt verzögern. | 30 | 50% | 15 | Alle möglichen Fehlerquellen des Projekts bereits zu Beginn erkennen und vorbeugen. | Mehraufwand betreiben, falls es soweit kommt. |
| **Summe** |  |  | **224** |  | **158.2** |  |  |

### Neubeurteilungen

|  |  |  |
| --- | --- | --- |
| **Inception I1** |  |  |
| Datum: | 24.02.14 |  |
| Risiko: | - |  |
| Beschreibung | - |  |
| Eintrittswahrscheinlichkeit neu: | - |  |

|  |  |  |
| --- | --- | --- |
| **Elaboration E1** |  |  |
| Datum: | 03.03.14 |  |
| Risiko: | R2 | Webserver |
| Beschreibung: | Da Webserver (IIS) installiert, ist dieses Risiko beseitigt | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |
| Risiko: | R3 | ASP.NET |
| Beschreibung: | ASP.NET Tutorial MVC-5 von allen Teammitgliedern durchgeführt. | |
| Eintrittswahrscheinlichkeit neu: | 50% |  |

|  |  |  |
| --- | --- | --- |
| **Elaboration E2** |  |  |
| Datum: | 18.03.14 |  |
| Risiko: | R1 | Datenbank |
| Beschreibung: | Datenbank wurde angelegt und Zugriff funktioniert, dieses Risiko ist somit beseitigt | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |
| Risiko: | R3 | ASP.NET |
| Beschreibung: | Erster Prototyp wurde eingerichtet und funktioniert | |
| Eintrittswahrscheinlichkeit neu: | 40% |  |
| Risiko: | R5 | Versionierung / Unit Testing |
| Beschreibung: | Visual Studio Online wurde eingerichtet und die Versionierung funktioniert | |
| Eintrittswahrscheinlichkeit neu: | 40% |  |

|  |  |  |
| --- | --- | --- |
| **Elaboration E3** |  |  |
| Datum: | 02.04.14 |  |
| Risiko: | R3 | ASP.NET |
| Beschreibung: | Prototyp wurde erstellt (Ein-/ Auschecken, Employee erstellen, auf Projekt buchen, Roles, etc.) | |
| Eintrittswahrscheinlichkeit neu: | 10% |  |
| Risiko: | R4 | Falsche Aufwand-Schätzung |
| Beschreibung: | Erste Erfahrungen zeigen, dass Schätzungen nicht komplett falsch sind | |
| Eintrittswahrscheinlichkeit neu: | 50% |  |
| Risiko: | R5 | Versionierung / Unit Testing |
| Beschreibung: | Unit Testing eingerichtet, erste Unit Tests mit Prototyp lauffähig | |
| Eintrittswahrscheinlichkeit neu: | 5% |  |
| Risiko: | R6 | Security |
| Beschreibung: | Durch Nutzung von MVC User Identity ist vieles bereits implementiert, Probleme jedoch immer noch möglich | |
| Eintrittswahrscheinlichkeit neu: | 15% |  |

|  |  |  |
| --- | --- | --- |
| **Construction C1** |  |  |
| Datum: | 18.04.14 |  |
| Risiko: | R3 | ASP.NET |
| Beschreibung: | Weitere Erfahrungen mit ASP.NET gesammelt. Probleme sind aber immer möglich. | |
| Eintrittswahrscheinlichkeit neu: | 5% |  |
| Risiko: | R4 | Falsche Aufwand-Schätzung |
| Beschreibung: | Weitere Erfahrungen zeigen, dass Schätzungen i.O. sind | |
| Eintrittswahrscheinlichkeit neu: | 10% |  |
| Risiko: | R5 | Versionierung / Unit Testing |
| Beschreibung: | Unit Testing, Versionierung funktioniert ohne Probleme | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |

|  |  |  |
| --- | --- | --- |
| **Construction C2** |  |  |
| Datum: | 01.05.14 |  |
| Risiko: | R4 | Falsche Aufwand-Schätzung |
| Beschreibung: | Weitere Erfahrungen mit ASP.NET gesammelt. Probleme sind aber immer möglich. | |
| Eintrittswahrscheinlichkeit neu: | 5% |  |
| Risiko: | R6 | Security |
| Beschreibung: | Erfahrungen mit ASP.NET gesammelt und ohne Probleme verwendet. Sollte SSL eingesetzt werden, könnten noch Probleme auftreten | |
| Eintrittswahrscheinlichkeit neu: | 5% |  |
| Risiko: | R8 | Falsches Risiko-Management |
| Beschreibung: | Da alle benötigten Komponente bereits verwendet wurden, sind mit keinen weiteren Risiken zu rechnen | |
| Eintrittswahrscheinlichkeit neu: | 5% |  |

|  |  |  |
| --- | --- | --- |
| **Construction C3** |  |  |
| Datum: | 15.05.14 |  |
| Risiko: | R6 | Security |
| Beschreibung: | Mit keinen weiteren Probleme mehr zu rechnen. | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |
| Risiko: | R7 | Diagramme Auswertung |
| Beschreibung: | Erste Diagramme mit Chart.JS erstellt. Framework nun bekannt. | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |
| Risiko: | R8 | Falsches Risiko-Management |
| Beschreibung: | Mit keinen weiteren Risiken mehr zu rechnen. | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |

|  |  |  |
| --- | --- | --- |
| **Transition T1** |  |  |
| Datum: | 27.05.14 |  |
| Risiko: | R3 | ASP.NET |
| Beschreibung: | Projekt wurde erfolgreich abgeschlossen. | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |
| Risiko: | R4 | Falsche Aufwand-Schätzung |
| Beschreibung: | Projekt wurde erfolgreich abgeschlossen. | |
| Eintrittswahrscheinlichkeit neu: | 0% |  |

## Umgang mit Risiken

Die Arbeitspakete wurden Bereits sehr grosszügig geplant, so dass ein Puffer für allfällige Probleme bereits vorhanden ist. Um Risiken wie Client/Server, DB, ASP.NET-Technologie vorzubeugen, wird früh ein funktionierender Prototyp erstellt.

Nach jeder Iteration werden die Risiken neu bewertet und im Dokument „TechnischeRisiken.xlsx“ in einem separaten Tabellenblatt geführt.

# Arbeitspakete

Die Arbeitspakete unseres Projektes werden im Tool Redmine geführt. Dies ermöglicht eine einfache Zeiterfassung der geleisteten Arbeiten. Für den Zugriff auf Redmine können folgende Informationen genutzt werden:

|  |  |
| --- | --- |
| Gast Login | |
| Server-Link | <http://sinv-56077.edu.hsr.ch/redmine> |
| Benutzername | guest |
| Passwort | 12345678 |

| **Thema (Planungsstand Projekt Anfang)** | **Geplant(h)** | **Thema(Planungsstand Projekt Ende)** | **Geplant(h)** | **Von** | **Bis** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| **Inception I1** |  |  |  |  |  |
| Projektantrag stellen | 3 | Wochenmeeting SW01 | 3 | 20.02.2014 | 20.02.2014 |
| In Projektplan Templates Einarbeiten | 1.5 | In Projektplan Templates Einarbeiten | 1.5 | 20.02.2014 | 20.02.2014 |
| Wochenmeeting SW01 | 3 | Projektantrag stellen | 3 | 20.02.2014 | 21.02.2014 |
| **Total** | **7.5** | **Total** | **7.5** |  |  |
|  |  |  |  |  |  |
| **Elaboration E1** |  |  |  |  |  |
| Projektplan erstellen | 40 | Iterationsassessment Elaboration 1 | 0.5 | 06.03.2014 | 06.03.2014 |
| SQL-Server installieren | 1 | Vorbereitung Review Projektplan | 3 | 05.03.2014 | 06.03.2014 |
| Redmine installieren und konfigurieren | 1 | Review Projektplan | 2.25 | 06.03.2014 | 06.03.2014 |
| Webserver installieren und konfigurieren | 0.5 | Wochenmeeting SW02 | 3 | 20.02.2014 | 20.02.2014 |
| Asp.net MVC-5 Tutorial | 12 | ASP.NET MVC-5 Tutorial | 12 | 19.02.2014 | 30.03.2014 |
| Wochenmeeting SW02 | 3 | Webserver installieren und konfigurieren | 0.5 | 27.02.2014 | 27.02.2014 |
| Review Projektplan | 2.25 | Redmine installieren und konfigurieren | 1 | 27.02.2014 | 27.02.2014 |
| Vorbereitung Review Projektplan | 3 | SQL-Server installieren | 1 | 27.02.2014 | 27.02.2014 |
|  |  | Projektplan erstellen | 40 | 20.02.2014 | 06.03.2014 |
| **Total** | **62.75** | **Total** | **63.25** |  |  |
|  |  |  |  |  |  |
| **Elaboration E2** |  |  |  |  |  |
| UseCase im brief-Format | 3 | Erstellung Dokument Domainanalyse | 3 | 07.03.2014 | 13.03.2014 |
| UseCase Diagramm | 2 | Iterationsplan Elaboration 2 | 0.5 | 06.03.2014 | 06.03.2014 |
| Nicht funktionale Anforderungen | 2 | Iterationsassessment Elaboration 2 | 0.5 | 14.03.2014 | 14.03.2014 |
| Klassendiagramm | 4 | Dokument Anforderungsspezifikation | 3 | 06.03.2014 | 13.03.2014 |
| System Sequenz Diagramm | 2 | Vorbereitung Review Anforderung und Analyse | 3 | 19.03.2014 | 20.03.2014 |
| Wichtigste Operation Contracts | 3 | Review Anforderung und Analyse | 2.25 | 20.03.2014 | 20.03.2014 |
| Asp.NET WebProject auf Visual Studio online erstellen und mit Teammitgliedern verbinden | 4 | Wochenmeeting SW04 | 3 | 13.03.2014 | 13.03.2014 |
| wt4u mit DB verbinden | 3 | Wochenmeeting SW03 | 3 | 06.03.2014 | 06.03.2014 |
| wt4u mit Webserver verbinden | 3 | Simple DB-Abfragen testen | 4 | 06.03.2014 | 13.03.2014 |
| UseCase Full-Dressed-Format | 6 | Architekturprototyp mittels einfachsten Operationen testen | 8 | 06.03.2014 | 13.03.2014 |
| Architekturprototyp mittels einfachsten Operationen testen | 8 | UseCase Fully-Dressed-Format | 6 | 06.03.2014 | 13.03.2014 |
| Simple DB-Abfragen testen | 4 | wt4u mit Webserver verbinden | 3 | 06.03.2014 | 13.03.2014 |
| Wochenmeeting SW03 | 3 | wt4u mit DB verbinden | 3 | 06.03.2014 | 14.03.2014 |
| Wochenmeeting SW04 | 3 | ASP.NET WebProject auf Visual Studio online erstellen und mit Teammitgliedern verbinden | 4 | 06.03.2014 | 13.03.2014 |
| Review Anforderung und Analyse | 2.25 | Wichtigste Operation Contracts | 3 | 07.03.2014 | 14.03.2014 |
| Vorbereitung Review Anforderung und Analyse | 3 | System Sequenz Diagramm | 2 | 07.03.2014 | 14.03.2014 |
|  |  | Klassendiagramm | 4 | 07.03.2014 | 14.03.2014 |
|  |  | Nicht funktionale Anforderungen | 2 | 07.03.2014 | 14.03.2014 |
|  |  | UseCase Diagramm | 2 | 07.03.2014 | 14.03.2014 |
|  |  | UseCase im brief-Format | 3 | 07.03.2014 | 14.03.2014 |
| **Total** | **55.25** | **Total** | **62.25** |  |  |
|  |  |  |  |  |  |
| **Elaboration E3** |  |  |  |  |  |
| Architekturdiagramm | 6 | Testing Dokument | 1 | 01.04.2014 | 03.04.2014 |
| Externes UI-Design | 3 | Authorization implementieren | 5 | 24.03.2014 | 28.03.2014 |
| User/Login implementieren | 9 | Projekt buchen MVC Simpel implementieren | 5 | 27.03.2014 | 28.03.2014 |
| Simples UI erstellen (JS,HTML,CSS) um Resultate darzustellen | 16 | Erste Simple Unit Tests implementiert | 4 | 27.03.2014 | 03.04.2014 |
| Einchecken/Auschecken Mitarbeiter grob implementieren | 10 | Erstellung Software Architektur Dokument Entwurf | 7 | 21.03.2014 | 28.03.2014 |
| Wochenmeeting SW05 | 3 | Klassen erstellen, DB Tabellen generieren | 3 | 20.03.2014 | 21.03.2014 |
| Wochenmeeting SW06 | 3 | Iterationsplan Elaboration 3 | 0.5 | 17.03.2014 | 17.03.2014 |
| Review End of Elaboration | 2.25 | Iterationsassessment Elaboration 3 | 0.5 | 28.03.2014 | 28.03.2014 |
| Vorbereitung Review End of Elaboration | 3 | Systemtest Architekturprototyp | 3 | 17.03.2014 | 27.03.2014 |
| Usability Test Paper Prototyping | 1 | Vorbereitung Review End of Elaboration | 3 | 02.04.2014 | 03.04.2014 |
| Systemtest Architekturprototyp | 3 | Review End of Elaboration | 2.25 | 03.04.2014 | 03.04.2014 |
|  |  | Wochenmeeting SW06 | 3 | 27.03.2014 | 27.03.2014 |
|  |  | Wochenmeeting SW05 | 3 | 20.03.2014 | 20.03.2014 |
|  |  | Einchecken/Auschecken MVC Simpel implementieren | 10 | 17.03.2014 | 27.03.2014 |
|  |  | Login MVC Simpel implementieren | 16 | 20.03.2014 | 27.03.2014 |
|  |  | Projektdokumentation / Tickets Elaboration 3 | 5 | 17.03.2014 | 29.03.2014 |
|  |  | Externes UI-Design (nur Hauptansichten) | 3 | 17.03.2014 | 27.03.2014 |
|  |  | N-Tier Diagramm & Package Diagramm | 6 | 17.03.2014 | 27.03.2014 |
| **Total** | **59.25** | **Total** | **80.25** |  |  |
|  |  |  |  |  |  |
| **Construction C1** |  |  |  |  |  |
| Ein/Auschecken Mitarbeiter vollstÃ¤ndig implementieren | 16 | Automatische Builds erstellen (Visual Studio Online) | 1 | 04.04.2014 | 10.04.2014 |
| View Ein/Auschecken | 16 | Anpassung Planung (Projektplan, Redmine) | 0.5 | 04.04.2014 | 11.04.2014 |
| View Login | 8 | Neue BusinessLayer implementieren/Methoden auslagern | 5 | 03.04.2014 | 11.04.2014 |
| Wochenmeeting SW07 | 3 | Systemtests gemäss Testphasen + Protokoll | 3 | 31.03.2014 | 11.04.2014 |
| Wochenmeeting SW08 | 3 | Unit Tests gemäss Testphasen + Protokoll | 10 | 31.03.2014 | 11.04.2014 |
| Review Architektur und Design | 2.25 | Simples View/Controller Home | 5 | 31.03.2014 | 11.04.2014 |
| Vorbereitung Review Architektur und Design | 3 | Iterationsplan Construction 1 | 0.5 | 31.03.2014 | 31.03.2014 |
|  |  | Iterationsassessment Construction 1 | 0.5 | 11.04.2014 | 11.04.2014 |
|  |  | Wochenmeeting SW08 | 3 | 10.04.2014 | 10.04.2014 |
|  |  | Wochenmeeting SW07 | 3 | 03.04.2014 | 03.04.2014 |
|  |  | Simples View/Controller Project | 4 | 31.03.2014 | 10.04.2014 |
|  |  | Simples View/Controller WorkingSession | 4 | 31.03.2014 | 11.04.2014 |
|  |  | Simples View/Controller Account | 5 | 31.03.2014 | 10.04.2014 |
| **Total** | **51.25** | **Total** | **44.5** |  |  |
|  |  |  |  |  |  |
| **Construction C2** |  |  |  |  |  |
| Membern Rechte hinzufuegen | 6 | Account Tests | 5 | 01.05.2014 | 08.05.2014 |
| View Projektleiter erstellen (zusaetzliche Ansicht) | 20 | AccountTests | 4 | 01.05.2014 | 08.05.2014 |
| View Arbeitgeber erstellen (zusaetzliche Ansichten) | 16 | Systemtests + Protokoll C2 | 3 | 30.04.2014 | 01.05.2014 |
| Statistik Arbeitszeiten Mitarbeiter implementieren | 20 | isActive Filter bei All Employee Table (Javascript) | 7 | 01.05.2014 | 08.05.2014 |
| Wochenmeeting SW09 | 3 | Detail-View User | 6 | 01.05.2014 | 08.05.2014 |
| Wochenmeeting SW10 | 3 | Neue Status Seite Home/Status | 2 | 29.04.2014 | 01.05.2014 |
| Wochenmeeting SW11 | 3 | Entwicklung eines GUI | 12 | 28.04.2014 | 02.05.2014 |
|  |  | Code Refactoring gemäss CodeReview C2 | 1 | 25.04.2014 | 01.05.2014 |
|  |  | Ausblenden Start/End Project/Break auf Home-View | 0.75 | 24.04.2014 | 27.04.2014 |
|  |  | Delete von Project | 3 | 24.04.2014 | 24.04.2014 |
|  |  | Edit/Detail View von Project | 4 | 24.04.2014 | 01.05.2014 |
|  |  | zusätzliches Feld isClosed in Project | 1 | 24.04.2014 | 01.05.2014 |
|  |  | Transaction Neue Klasse (Alle Transactions aus Controller Auslagern) | 0.5 | 23.04.2014 | 01.05.2014 |
|  |  | Kommentar am Ende des ProjectBooking (inkl. Pflichtfeld) | 1 | 22.04.2014 | 22.04.2014 |
|  |  | Erstellung Code Review Dokument | 2 | 22.04.2014 | 02.05.2014 |
|  |  | Test Edit/Delete Alle Klassen + Protokoll | 3 | 21.04.2014 | 27.04.2014 |
|  |  | Navigation inkl. Berechtigungen | 2 | 14.04.2014 | 02.05.2014 |
|  |  | Edit/Delete WorkingSession / Break / ProjectBookingTime inkl.Views | 7 | 18.04.2014 | 24.04.2014 |
|  |  | Roles/Rechte Account | 7 | 14.04.2014 | 01.05.2014 |
|  |  | Roles/Rechte WorkingSession | 7 | 14.04.2014 | 01.05.2014 |
|  |  | Erstellung Software Architektur Dokument | 12 | 18.04.2014 | 01.05.2014 |
|  |  | Iterationsplan Construction 2 | 0.5 | 14.04.2014 | 14.04.2014 |
|  |  | Iterationsassessment Construction 2 | 0.5 | 02.05.2014 | 02.05.2014 |
|  |  | Vorbereitung Review Architektur und Design | 3 | 01.05.2014 | 01.05.2014 |
|  |  | Review Architektur und Design | 2.25 | 01.05.2014 | 01.05.2014 |
|  |  | Wochenmeeting SW11 | 3 | 01.05.2014 | 01.05.2014 |
|  |  | Wochenmeeting SW10 | 3 | 24.04.2014 | 24.04.2014 |
|  |  | Wochenmeeting SW09 | 3 | 17.04.2014 | 17.04.2014 |
|  |  | Roles/Rechte Project | 7 | 14.04.2014 | 01.05.2014 |
|  |  | JS-Tabelle/View Account | 12 | 24.04.2014 | 01.05.2014 |
|  |  | JS-Tabelle/View WorkingSession | 12 | 17.04.2014 | 01.05.2014 |
|  |  | JS-Tabelle/View Projects | 12 | 17.04.2014 | 01.05.2014 |
| **Total** | **71** | **Total** | **148.5** |  |  |
|  |  |  |  |  |  |
| **Construction C3** |  |  |  |  |  |
| Statistik Projektzeiten DB-Abfrage implementieren | 10 | CSV Datei exportieren | 5 | 01.05.2014 | 15.05.2014 |
| Grafische Darstellung Arbeitszeiten Mitarbeiter implementieren (optional) | 0 | Diagramme Account (optional) | 0 | 08.05.2014 | 16.05.2014 |
| Grafische Darstellung Projektauswertungen implementieren (optional) | 0 | Aktuelle Uhrzeit immer sichtbar | 3 | 05.05.2014 | 16.05.2014 |
| Umfangreiche Tests der Software | 12 | Iterationsplan Construction 3 | 1 | 05.05.2014 | 05.05.2014 |
| Wochenmeeting SW12 | 3 | Iterationsassessment Construction 3 | 0.5 | 16.05.2014 | 16.05.2014 |
| Wochenmeeting SW13 | 3 | Usability Test Programm + Protokoll | 2 | 05.05.2014 | 16.05.2014 |
| Tabellarische Darstellung Projektauswertung implementieren | 10 | Auswertunen Account | 7 | 01.05.2014 | 08.05.2014 |
| Tabellarische Darstellung Arbeitszeiten Mitarbeiter implementieren | 10 | Auswertungen WorkingSession | 7 | 01.05.2014 | 08.05.2014 |
| Usability Test Programm | 1 | Wochenmeeting SW13 | 3 | 15.05.2014 | 15.05.2014 |
|  |  | Wochenmeeting SW12 | 3 | 08.05.2014 | 08.05.2014 |
|  |  | Systemtests + Protokoll | 2 | 08.05.2014 | 16.05.2014 |
|  |  | Diagramme WorkingSession (optional) | 0 | 08.05.2014 | 16.05.2014 |
|  |  | Diagramme Project (Optional) | 0 | 08.05.2014 | 16.05.2014 |
|  |  | Auswertungen Projekt | 7 | 01.05.2014 | 09.05.2014 |
| **Total** | **49** | **Total** | **40.5** |  |  |
|  |  |  |  |  |  |
| **Transaction T1** |  |  |  |  |  |
| Praesentation vorbereiten | 21 | Präsentation halten | 0.5 | 28.05.2014 | 28.05.2014 |
| Projekt ausfuehrbar bereitstellen | 15 | Alle Tabellen mit Bootstrap Pagination | 3 | 15.05.2014 | 15.05.2014 |
| Wochenmeeting SW14 | 3 | Dokumentationsübersicht HTML erstellen mit allen Links | 2 | 15.05.2014 | 20.05.2014 |
| Wochenmeeting SW15 | 3 | Projektplan kontrollieren auf allfällige Aktualisierungen, Tickets aktualisieren | 1 | 22.05.2014 | 28.05.2014 |
|  |  | Exceptionhandler implementieren | 4 | 15.05.2014 | 22.05.2014 |
|  |  | Testphase 4 durchführen + Protokoll | 1 | 15.05.2014 | 22.05.2014 |
|  |  | CodeReview C3 Dokument nachführen + Realisierung | 3 | 15.05.2014 | 22.05.2014 |
|  |  | Abschlussbericht | 3 | 19.05.2014 | 30.05.2014 |
|  |  | Iterationsplan Transition | 0.5 | 19.05.2014 | 19.05.2014 |
|  |  | Iterationsassessment Transition | 0.5 | 30.05.2014 | 30.05.2014 |
|  |  | Wochenmeeting SW14 | 3 | 22.05.2014 | 22.05.2014 |
|  |  | Projekt ausfuehrbar bereitstellen | 5 | 22.05.2014 | 29.05.2014 |
|  |  | Praesentation vorbereiten | 21 | 22.05.2014 | 29.05.2014 |
|  |  | Dokumentation kontrollieren, Rechtschreibung > Deployen | 2 | 19.05.2014 | 28.05.2014 |
| **Total** | **42** | **Total** | **49.5** |  |  |
|  |  |  |  |  |  |
| **Projekt wt4u Total** | **398** | **Projekt wt4u Total** | **496.25** |  |  |

# Infrastruktur

## Server-Architektur

Unser Projekt läuft auf einer virtuellen Maschine mit Microsoft-Server 2008 R2, welche von der HSR zur Verfügung gestellt wird. Darauf ist das Projektplaungstool Redmine installiert. Redmine wird genutzt, um die Arbeitspakete zu verwalten und um Statistiken und Arbeitszeitauswertung zu erhalten. Als zentrale Datenbank wird ein MS-SQL Server 2012 genutzt, welcher ebenfalls auf dem virtuellen Server installiert ist. Als Webserver für unser Produkt, wie auch für Redmine wird Microsoft Internet Information Services (IIS) verwendet.

Die Spezifikationen des Servers:

|  |  |
| --- | --- |
| Betriebssystem | Microsoft Windows Server 2008 R2 |
| Lebenszyklus virtuelle Maschine | 31.08.2014 |
| Hostname (FQDN) | Sinv-56077.edu.hsr.ch |
| CPU | 1x 2.27 GHz |
| RAM | 1 GB |
| Disk Space | 40 GB |
| LAN Adapter | 1 VMWare Flexible NIC (VMXNET3) |
| Video Adapter | VMWare SVGA II |

## Entwicklungsumgebung

Alle Teammitglieder arbeiten mit ihren persönlichen Notebooks, daher kann sich die Spezifikation dieser unterscheiden. Folgende Software wird allerdings bei allen verwendet:

* Microsoft Visual Studio Ultimate 2013
* Microsoft .NET Framework 4.5
* Projektplanung mittels Redmine
* Versionskontrolle mittels Visual Studio Online (Team Foundation Version Control)

## Dokumentverwaltung

Um die Projektdokumente für alle Teammitglieder verfügbar zu halten, wird Microsoft OneDrive verwendet. OneDrive ermöglicht das gleichzeitige Bearbeiten von Dokumenten durch verschiedene Mitglieder des Projektteams. Zudem bietet es eine automatische Versionierung (bei jeder Speicherung).

Gründe für den Einsatz von OneDrive.

* Einfache Installation
* Zentrale Ablage von überall zugreifbar
* Gleichzeitiges Bearbeiten durch mehrere Mitglieder
* Versionierung der verschiedenen Dokumente sichergestellt

# Qualitätsmassnahmen

## Dokumentation

Als Vorlage der erstellten Dokumente dienen die Templates, welche zur Verfügung gestellt wurden. Alle Projektbezogenen Dokumente werden über OneDrive allen Teammitgliedern zur Verfügung gestellt. Die Dokumente sollen laufend zu aktualisiert werden.

## Projektmanagement

Für das Projektmanagement wird das Tool Redmine eingesetzt. Redmine ist auf unserem virtuellen Server installiert.

|  |  |
| --- | --- |
| Gast Login | |
| Server-Link | <http://sinv-56077.edu.hsr.ch/redmine> |
| Benutzername | guest |
| Passwort | 12345678 |

## Entwicklung

### Vorgehen

Der Source Code befindet sich auf Microsoft Visual Studio Online. Die Versionskontrolle erfolgt nach dem Prinzip von Team Foundation Version Control. Nur Quellcodedateien mit lauffähigem Code werden in das zentrale Repository eingecheckt. Falls einmal falscher Code eingecheckt wurde, ist es durch Team Foundation Version Control möglich eine lauffähige ältere Version zurückzuholen.

### Unit Testing

Während der Entwicklung werden wir auf das Unit-Test Framework in Visual Studio Ultimate 2013 oder Nant zurückgreifen.

Den Zeitaufwand für das Schreiben der Tests ist jeweils in die Arbeitspakete der Implementierungen miteinbegriffen.

### Code Reviews

Auf Wunsch eines Teammitglieds können Code Reviews für einzelne, selbst entwickelte Codebausteine durchgeführt werden. Dieser Review findet jeweils anlässlich unserem wöchentlichem Teammeeting statt. Aufgrund der kurzen Projektdauer und verfügbaren Zeit, soll darauf geachtet werden, dass Reviews hauptsächlich für kritische Softwarekomponenten durchgeführt werden.

### Code Style Guidelines

Wir verwenden die offiziellen Microsoft C# Coding Conventions.

Link: <http://msdn.microsoft.com/en-us/library/ff926074.aspx>

Deshalb werden im Visual Studio Ultimate 2013 alle Einstellungen so belassen, wie sie bei der Installation standardmässig eingestellt sind.

* Code und Kommentare werden ausschliesslich in englischer Sprache verfasst.
* Bei jeder erstellten Klasse wird ein Kommentar verfasst, der denn Sinn der Klasse beschreibt.
* Methoden werden nur dann mit einem Kommentar versehen, wenn aus dem Methodennamen nicht ersichtlich ist, was diese Methode implementiert hat.
* Für alle weiteren Kommentare für Klassen / Methoden sollten die empfohlen Tag-Names auf der MSDN-Seite verwendet werden.   
  Link: <http://msdn.microsoft.com/en-us/library/5ast78ax.aspx>
* Für die Namespace Bezeichnungen werden die vorgefertigten Namen des MVC-5 Frameworks verwendet.
* Zusammengesetzte Wörter werden im CamelCase-Format geschrieben.
* Bei Variablen wird der Anfangsbuchstabe klein geschrieben.
* Bei Klassen und Methoden wird der Anfangsbuchstabe gross geschrieben.
* Variablen-, Klassen-, und Methodennamen sollen einen aussagekräftigen Namen haben. Bezeichnungen wie int a = 5; gibt es nicht (ausser bei Laufvariablen in for-Schlaufen).
* Bei klassischen Klassenvariablen mit gettern und settern wird mit Properties gearbeitet.
* Underscore in Varibelnamen vermeiden.
* Visibility von Variabeln, Klassen, etc. immer angeben.

namespace Application

{

// this class does something...

public class Class

{

private static int Pi = 3.14;

private int ExampleVariable;

// Calculates the sum of two summands

public int CalculateSum(int summand\_a, int summand\_b)

{

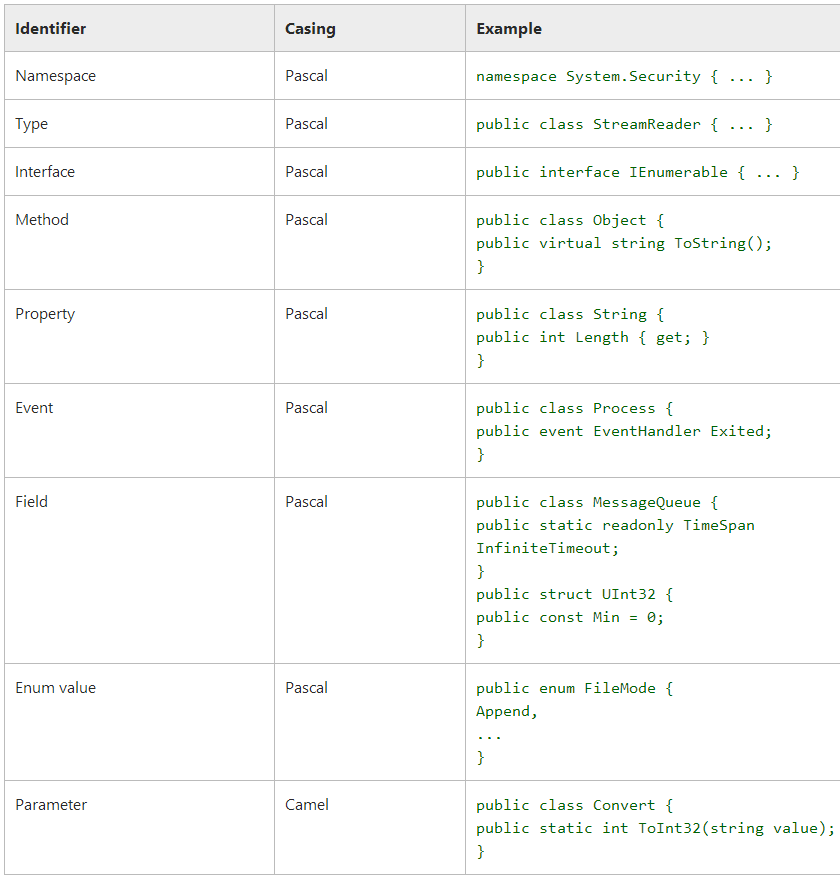
return summand\_a + summand\_b;

}

}

}

Hier nochmals eine Übersicht zur Gross- und Kleinschreibung



## Testen

### Unit Tests

Der grösste Teil der Tests unseres Projektes wird aus Unit Tests bestehen. Die Unit Tests werden erst lokal programmiert, dann lokal getestet. Das automatisierte Testing wird mit Nant oder VisualStudio Online geschehen, je nachdem welche Applikation sich als besser erweist.

### System Tests

Ende des Milestones 3 werden ausführliche Systemtests implementiert.

### Usability Tests

In der Phase in der das Paper Prototyping des externen Designs durchgeführt wird, werden wir innerhalb der Gruppe schon kleine Usability Tests auf Papier durchführen.

Ende der Constructionphase werden wir Usability-Tests mit echten Usern durchführen um in Erfahrung zu bringen ob unser Programm benutzerfreundlich, gut bedienbar und effizient ist.

### Web Browser

Da wir eine Web-Applikation erstellen, wollen wir sicherstellen, dass unsere Applikation auf sämtlichen aktuellen Browsern läuft. Dazu werden wir diverse Browser unter die Lupe nehmen.

### Bug Tracking

Werden gröbere Bugs währen der Entwicklung festgestellt, erfassen wir in unserem Projektmanagementtool Redmine neue Tickets. In diesen Tickets wird festgehalten welcher Bug wie viel Zeit zur Behebung braucht.