

# Aufgabenblatt 1

**Projektnummer:** swp10-7

**Projekttitle:** jQuery Plugin - Erweiterte Autovervollständigung auf der Basis von SPARQL Endpunkten

**Abgabe:** 12.04.2010

**Verantwortlicher:** Frederik Baumgardt

# Risikoanalyse

## Zielsetzung

Jedes umfangreiche und komplexe Projekt wird zwangsläufig auf Probleme stoßen. Je nachdem wie gravierend die Probleme sind und wie effektiv darauf reagiert wird, können sie vollständig abgefangen werden oder es treten Verzögerungen und Qualitätsminderungen verschiedener Ausprägung im Projekt auf, bis hin zu einem kompletten Fehlschlag.

Um dem Vorzubeugen ist eine genaue Analyse der potentiellen Problemstellen und ihrer Zusammenhänge nötig. Darauf aufbauend lassen sich Strategien zur Lösung und Vermeidung der Probleme entwickeln.

Für unser Projekt sind die folgenden Risiken besonders präsent. Sie lassen sich zusammenfassen in organisatorische (1 - 5), personelle (5 - 8) und technische (9 - 10) Risiken.

*Anmerkung: Die Gefahr des Datenverlusts ist der Gruppe bewusst, wird allerdings wegen des geforderten Einsatz von SVN weniger gravierend als die u.g. Risiken eingeschätzt.*

### 1. unpräzises Auftragsverständnis

Ursache: ungenaue Vorgaben und unterschiedliche Interpretationen derselben; Missverständnisse zwischen Team und Auftraggeber

Folge: je später im Projektverlauf, desto schwerer zu korrigierende Inkompatibilität zu Prozessen und Systemen des Kunden; durch Korrekturen variierende Vorgaben und nicht zielorientierter Zeitaufwand

Maßnahmen: Absprache eines projekt-spezifischen Vokabulars; technisch versierte Vertreter der Kundenseite in Planung einbeziehen; Abnahme von Milestones

### 2. ungenügende Programmierkenntnisse

Ursache: geringe Programmiererfahrung und Kenntnis der einzusetzenden Schnittstellen

Folge: fehlerhafte Modellierung; fehlerhafte, umständliche und ineffiziente Implementierungen; eigene Implementierung von Funktionalität die im Framework bereits vorhanden

Maßnahmen: sorgfältige und anwendungsorientierte Recherche (z.B. Tutorials); Kommunikation mit den Entwicklern der Frameworks; Arbeitsteilung und Spezialisierung bei der Recherche

### 3. fehlerhafte Aufwandsschätzung

Ursache: Zeitbedarf und Schwierigkeit der Teilaufgaben werden falsch eingeschätzt

Folge: Verschiebung des Zeitplans oder unzureichende Qualität der Arbeitsabschnitte; Verfehlung von Milestones; Überlastung und Demotivation der Teammitglieder

Maßnahmen: realistische Zieldefinition; besondere Sorgfalt bei Planung und Erstellung des Pflichtenhefts; regelmäßige Prüfung des tatsächlichen Zeitbedarfs und ggf. Korrekturen im Pflichtenheft

#### **4. falsche Prioritäten**

Ursache: Perfektionismus oder fehlender Überblick führen zu unverhältnismäßigem Zeitaufwand bei kleineren oder weniger wichtigen Teilaufgaben; Aufgaben werden in der falschen Reihenfolge erledigt

Folge: weniger Zeit für wichtige Aufgaben; Verzögerung abhängiger Arbeiten

Maßnahmen: eindeutige Definition und Zuteilung von Aufgaben; regelmäßige Rücksprache mit dem Verantwortlichen der Teilaufgabe; Zeitmanagement im Pflichtenheft festhalten

#### **5. mangelhafte Kommunikation**

Ursache: zu wenig, zu unstrukturierte und zu ungenaue Kommunikation zwischen den Teammitgliedern

Folge: Teammitglieder folgen falschen Anweisungen oder wissen nicht was andere gerade machen; Aufgaben werden nicht, fehlerhaft oder redundant erledigt; Teilarbeiten sind untereinander oder zu externen Systemen inkompatibel

Maßnahmen: regelmäßige Treffen und Skype-Konferenzen; Einsatz des Projektmanagementsystems Redmine; Teammitglieder tragen Aufgaben eigenständig ein, anschliessend Abnahme durch Verantwortlichen der Teilaufgabe

#### **6. Konflikte**

Ursache: unterschiedliche Auffassungen zu Organisations- oder Implementierungsfragen; ungeklärte Kompetenzfragen; zwischenmenschliche Probleme durch unsoziales Verhalten

Folge: Teammitglieder erledigen Aufgaben nicht nach Absprache; Arbeitsteilung wird erschwert; Motivation der Teammitglieder sinkt

Maßnahmen: präzise Formulierung von Rechten und Pflichten; frühstmögliche Klärung von Meinungsverschiedenheiten; Mediation bei persistenten Konflikten; notfalls Richtspruch durch Projektleiter

#### **7. Motivationsprobleme**

Ursache: fehlendes Interesse; Überforderung; Misserfolge; unfaire Arbeitsteilung; externe Probleme

Folge: sinkende Leistungsfähigkeit einzelner Teammitglieder; Verzögerung oder verminderte Qualität von Teilergebnissen; Demotivation kann auf weitere Mitglieder übergreifen

Maßnahmen: Evaluation der Stärken, Schwächen und Interessen von Teammitgliedern; faire und interessensgerechte Aufteilung der Arbeiten; flexible Verteilung des Aufwands einzelner Teammitglieder in den Teilaufgaben (nicht jeder muss an jeder Teilaufgabe mitarbeiten); Treffen zum gemeinsamen Arbeiten; Verantwortliche der Teilaufgaben und Projektleiter dienen als Ansprechpartner bei Unzufriedenheit

#### **8. Personalausfälle**

Ursache: Belastung durch andere Kurse oder Arbeit; familiäre und gesundheitliche Probleme; Kurs- oder Studiumsabbruch

Folge: erhöhte zeitliche Belastung der übrigen Teammitglieder; Aufwand zur Einarbeitung in bisherige Arbeit des Teammitglieds; unerledigte Teilaufgaben

Maßnahmen: Schaffung von Redundanzen (stellvertretende Verantwortliche, gemeinsame Recherche, Pair-Programming); frühzeitige Bekanntgabe von geplanten Ausfallterminen in Redmine; Umverteilung zusätzlicher Arbeit möglichst gleichmäßig, aber effizient (wer kann evtl. schnell und ohne Vorbereitung die Aufgaben erledigen), durch Verantwortliche der Teilbereiche; kontinuierliche Dokumentation der eigenen Arbeit durch jedes Mitglied mit Tickets, Wiki-Einträgen und kommentierten Code

## 9. unentdeckte Fehler

Ursache: mit Skriptsprachen treten häufiger (als bei Entwicklung in Compilersprachen) unauffällige Fehler auf, die nur in seltenen Szenarien sichtbare Wirkung entfalten; semantische Fehler die bei entwicklungsinternen Prüfungen übersehen wurden

Folgen: erheblicher Mehraufwand bei der Behebung, durch späte Entdeckung; schwer nachvollziehbare Probleme bei abhängigen Arbeitsschritten (u.a. wg. steigender Komplexität des Gesamtsystems mit Projektfortschritt); bei Entdeckung durch Kunden (bzw. Betreuung) folgen Vertragsstrafen (bzw. Notenabzug)

Maßnahmen: sorgfältig aufgebaute Testverfahren; in den Entwicklungsprozess integriertes Test-Rahmenwerk; Zusammenarbeit zwischen den Verantwortlichen für Testing, Modelling und Implementierung

## 10. inkompatible Schnittstellen

Ursache: ungenaue Recherche; ambivalente oder fehlerhafte Modellierung; Bugs in den Frameworks

Folge: Nachkorrekturen werden notwendig; Fehlersuche dauert bei Bugs in Frameworks sehr lange; Probleme können evtl. nur durch Andere behoben werden

Maßnahmen: besonderes Augenmerk auf Recherche und Modellierung; teamintern strukturiertes Bug-Reporting; bei persistenten Bugs Zusammenarbeit mit Verantwortlichen für Modellierung und Implementierung; Identifizierung von Ansprechpartnern für externe Softwarekomponenten und direkte Kontaktaufnahme bei auftretenden Bugs

# Rollenverteilung

Rolle	Person
Projektleitung	Frederik Baumgardt
Projektadministration	Marina Mitjagin
Technischer Assistent	Clemens Hoffmann
Recherche	Thomas Schöne
Testing	Konrad Baumheier
Modellierung	Frederik Baumgardt
Implementierung	Marcus Nitzschke
Dokumentation & Qualitätssicherung	Marina Mitjagin

# Website

<http://pcai042.informatik.uni-leipzig.de/~swp10-7/>