

Risikoanalyse für das Softwareprojekt AGB

Allgemeine Präventionsmaßnahmen

Es wurde ein fester wöchentlicher Termin, bei dem sich die ganze Gruppe trifft, festgelegt. Bei diesem sollen neben dem Plan für die kommende Woche auch die individuellen Arbeiten und dabei auftretende Probleme der vergangenen Woche offen ausgesprochen werden. Dadurch wird sichergestellt, dass sich die Gruppe auf einem Wissens- und Informationsstand befindet. Im Zuge dessen wurden allgemeine Verhaltensregeln (insbesondere für Konfliktfälle) vereinbart. Dazu zählen:

- das offene Ansprechen von Problemen und Streitigkeiten fachlicher und zwischenmenschlicher Natur, wobei Kritik sachlich geübt werden soll und Kritikfähigkeit von jedem Teammitglied erwartet wird
- fachliche Meinungsverschiedenheiten, die trotz Bemühens nicht im Konsens geklärt werden konnten, sollen möglichst durch Mehrheitsentscheidungen entschieden werden, wobei im Zweifelsfall der Projektleiter das letzte Wort hat
- bei der Vielzahl von Problemen soll sich in der Gruppe gegenseitig geholfen werden

1. Risikofaktoren

In der Gruppe wurden folgende Risikofaktoren als die zehn wichtigsten festgehalten:

1.1. Überforderung

Da es für alle das erste umfangreiche Softwareprojekt ist, kann es sein dass die Rollenzuteilung Einzelne überfordert. Gruppenmitglieder können außerdem überfordert werden, da die Komplexität des Projektes noch nicht klar eingeschätzt werden kann. Eine gerechte Aufteilung der Aufgaben, die erforderlich ist, um Konflikte und Stress zu vermeiden, kann nicht immer umgesetzt werden.

Gegenmaßnahmen:

- Austausch über persönliche Stärken und Schwächen und dementsprechende Rollenzuteilung in den ersten Treffen
- frühzeitige und gemeinsame Planung der Aufgabenverteilung
- gegenseitige Hilfestellungen

1.2. Aufgaben- und Aufwandsunterschätzung

Wegen geringer Erfahrung und Kenntnisse in der Erstellung umfangreicher Softwareprojekte ist es möglich, dass Aufgaben für zu einfach gehalten und dadurch mit zu geringem Aufwand abgeschätzt werden. Dies kann zu zeitlichen Engpässen führen.

Gegenmaßnahmen:

- Festlegung kurzzeitiger (interner) Ziele um Schätzungen zu verfeinern
- Einigung auf interne Deadlines vor den Abgabeterminen
- frühzeitige Planung der einzelnen Aufgaben

1.3. Übersehen bzw. Missverständnis von Produktanforderungen

Es ist möglich das Wünsche des Kunden übersehen, Anforderungen falsch verstanden werden und somit nicht das gewünschte Produkt entwickelt wird.

Gegenmaßnahmen:

- Kundengespräche gewissenhaft führen, um daraus ein möglichst vollständiges Lasten/Pflichtenheft zu erstellen
- Checklisten
- möglicherweise spätere Nachfragen beim Kunden
- gegenseitige Kontrolle in der Gruppe

1.4. zu lange Einarbeitungszeit in die Frameworks/Toolkits

Da die zu verwendenden Toolkits und Frameworks (Prefuse, Borderflow) dem gesamten Team unbekannt sind, ist es schwer einzuschätzen, wie viel Zeit für die Einarbeitung benötigt wird.

Gegenmaßnahmen:

- Aufteilung der Einarbeitung und gegenseitige Schulung
- Generierung von Paket- und Klassendiagrammen aus dem Code der Frameworks, für eine bessere Übersicht
- Erstellung von Java-Doc

1.5. falsche Prioritätensetzung

Durch Über- bzw. Untererfüllung von (Teil-)Aufgaben oder Anforderungen kann es zu Verzögerungen und Fehlentwicklungen kommen.

Gegenmaßnahmen:

- Absprache und Festlegung mit dem Auftraggeber
- insbesondere Herausarbeiten der Qualitätsanforderungen (Funktionalität, Zuverlässigkeit, Effizienz, usw.)
- Kontrolle durch die Gruppe

1.6. Konflikte in der Gruppe

In der Gruppe kann es zu fachlichen aber auch persönlichen Konflikten kommen. Mangelhafte Kommunikation kann außerdem zu Missverständnissen führen.

Gegenmaßnahmen:

- die allgemeinen Präventionsmaßnahmen
- ferner können technische Möglichkeiten (Skype) genutzt werden, um Absprachen außerhalb regulärer Treffen zu ermöglichen
- sollten Konflikte in der Gruppe nicht gelöst werden können, so wird Rücksprache mit dem Tutor/Betreuer gehalten

1.7. falsche und ungeeignete Werkzeuge

Da diese Gruppe noch nie zusammen gearbeitet hat, werden verschiedene Tools für gleiche Zwecke (bspw. IDEs) verwendet. Da außerdem neue Tools verwendet werden müssen, kann es durch Einarbeitungszeiten in diese zu Verzögerungen und Problemen kommen.

Gegenmaßnahmen:

- Auswahl und Einigung auf einheitliche Werkzeuge
- gegenseitige Einführungen und Hilfestellungen

1.8. mangelnde Motivation und Frustration

Durch Überforderung oder Missverständnisse Einzelner kann es zu Frustration kommen. Außerdem mindert ein Mangel an Motivation die Produktivität und beeinflusst somit die Arbeit der ganzen Gruppe negativ.

Gegenmaßnahmen:

- Rücksichtnahme auf die Fähigkeiten und Wünsche Einzelner (notfalls Umstrukturierung)
- Hilfestellungen
- Klare Zielstellungen und (Selbst-)Reflexion:
Aufzeigen, was bereits alles erreicht wurde und was noch zu schaffen ist
- Persönliche Motive / Gruppenmotive finden („Wofür machen ich / wir das eigentlich), z.B.:
 - Erfolgreiches Bestehen des Moduls
 - Bewältigung des ersten großen Softwareprojekts
 - die Chance, viel dazu lernen zu können, sowohl fachlich als auch Teamfähigkeit und die Zusammenarbeit betreffend

1.9. technischer Ausfall

Es könnte vorkommen, dass durch Hardwareschäden Gruppenmitglieder an der Arbeit gehindert werden und Datenverluste das Team zurückwerfen.

Gegenmaßnahmen:

- konsequente Nutzung von Subversion
- back-ups
- notfalls Ausweichen auf Computerpools

1.10. personeller Ausfall

Durch Krankheit oder andere Gründe können Gruppenmitglieder ausfallen.

Gegenmaßnahmen:

- Übernahme der Aufgaben durch andere Gruppenmitglieder
- offener und gut kommentierter Quellcode
- regelmäßiges Informieren der Gruppe über die eigene Arbeit

2. Gruppenaufteilung

Martin Türpe	Projektleiter
Klaus Lyko	Technischer Assistent
Anastasiya Chyhir	Verantwortliche für Recherche
Moritz Kähler	Verantwortlicher für Modellierung
Renata Wong	Verantwortliche für Tests
Marcello Scrobanita	Verantwortlicher für Implementierung
Victor Christen	Verantwortlicher für Qualitätssicherung und Dokumentation