



LinkedGeoData.org

— Risikoanalyse —

Von: Peter Pfahl (Projektleiter)

Abgabe: 11. Januar 2012

Gruppe: SWP12-11



Die Durchführung eines Softwareprojektes bringt viele Risiken mit sich. Um gegen die wichtigsten gerüstet zu sein, ist es von großer Wichtigkeit, diese schon im Vorfeld zu kennen und ihnen mit entsprechenden Maßnahmen entgegen zu wirken. Im Folgenden sind die zehn größten Risiken unseres Projektes mit entsprechenden Gegenmaßnahmen aufgelistet.

1. Zeitmanagement

Problem: Durch ungünstiges Zeitmanagement und falsche zeitliche Einschätzungen von Aufgaben kann es zu Stress wegen Zeitdruck und daraus resultierend zu Fehlern sowie negativen Auswirkungen auf die Motivation der Teammitglieder kommen.

Lösung: In erster Linie ist es sicherlich gut, die Deadlines für Aufgaben bis zu einer Woche früher anzusetzen, als für die Planung tatsächlich erforderlich ist. Hierdurch lösen auch Verzögerungen nicht zwangsläufig Stress aus und es entsteht ein gewisser „Zeitpuffer“. Außerdem hilft eine aufmerksame Beobachtung der wöchentlichen Aufwandserfassung, um mit diesen Erfahrungen eine optimale Aufgabenverteilung mit entsprechend realistischen Zeiträumen zu planen.

2. Ausfall von Teammitgliedern

Problem: Wie jedes Projekt, an dem Menschen arbeiten, beinhaltet auch unser Softwareprojekt das Risiko, dass ein Teammitglied zeitweise oder sogar gänzlich ausfällt. Mögliche Gründe hierfür sind von zusätzlichen universitären Belastungen (Prüfungsstress, Nachklausuren) über leichte und schwerere Krankheitsfälle bis hin zu Unfällen oder Bruch mit der Gruppe weit gefächert. Der Ausfall eines Teammitgliedes ist aufgrund der ihm übertragenen Aufgaben- und Verantwortungsbereiche eine starke Belastung für das Projekt.

Lösung: Dieses Risiko möchten wir durch unsere regelmäßigen Treffen und eine gute Dokumentation vermindern, da es dann im Ernstfall einfacher ist, die Arbeit eines Teammitgliedes nachzuvollziehen. Auch die entsprechende Ehrlichkeit der Teammitglieder untereinander, verbunden mit schneller und direkter Benachrichtigung, hilft dem ganzen Team und Projekt, wenn wirklich mal etwas derartiges auftritt. So kann bei einem Ausfall gemeinsam und effizient die Arbeit aufgeteilt werden.



3. Kommunikationsprobleme

Problem: Auch die Kommunikation ist ein Risiko, wenn mehrere Personen in einem Team arbeiten. So kann es passieren, dass einzelne Personen wichtige Informationen zu spät oder gar nicht bekommen, nicht an wichtigen Diskussionen teilnehmen (können) oder auch mal klassisch „aneinander vorbei reden“. Auch die temporäre Unerreichbarkeit kann unter Umständen zu Abstimmungsproblemen führen.

Lösung: Wie schon bei Punkt 2 sollen uns auch hier die regelmäßigen Projekttreffen helfen, die Kommunikation möglichst effektiv zu gestalten und Probleme sofort aus dem Weg zu räumen. Unterstützt wird dies zusätzlich durch einen eigenen Chat-Raum, in dem die Teammitglieder auch außerhalb der Veranstaltungen und Projekttreffen gut erreichbar sind.

4. Veränderte Zielsetzung

Problem: Ein weiteres Problem kann sich durch Änderungen der Projekt-Anforderungen ergeben, die bei Softwareprojekten nicht selten sind. Die Umsetzung solcher Anpassungen wird umso schwieriger, je später sie geäußert werden. Ab einem bestimmten Zeitpunkt der Entwicklung kann es sogar unmöglich sein, sie den Vorstellungen entsprechend umzusetzen.

Lösung: Die wichtigste Gegenmaßnahme stellt ein gut vorbereitetes Kundengespräch dar, in dem die Anforderungen so weit wie möglich konkretisiert werden sollten. Eine gute Recherche unterstützt dies zusätzlich. Außerdem ist ein regelmäßiger Kontakt zum Auftraggeber von großer Wichtigkeit, um eventuelle Diskrepanzen zwischen dem Entwicklungsstand und den Vorstellungen des Kunden so früh wie möglich zu erkennen und ihnen gegen zu wirken.

5. Fachliche Probleme / Fehler

Problem: Da das Team mit all seinen Verantwortungsbereichen zum ersten Mal ein solches Softwareprojekt bearbeitet, ist auch das Risiko von fachlichen Problemen und Fehlern nicht von der Hand zu weisen. Dies kann sich durch Denkblockaden und Fehlentscheidungen ebenso wie durch fehlerhafte Programmierung äußern.

Lösung: Am besten begegnet man diesem Risiko mit einer guten und ausführlichen Recherche. Die dadurch erlangte Sicherheit und fachliche Kompetenz vermeidet die meisten größten Fehler, sodass die Relevanz dieses Risikos sinkt. Des Weiteren ermöglicht die Hilfsbereitschaft der anderen Teammitglieder eine objektivere Betrachtung von Problemen und damit eine effektivere Lösung im Team.



6. Ungünstige Rollen- und Aufgabenverteilung

Problem: Die eben angesprochenen fachlichen Fehler betreffen natürlich auch den Projektleiter, der beispielsweise Aufgaben an die falschen Teammitglieder verteilt. Dies hat leider meist große Auswirkung auf den Fortschritt und den Erfolg des Projektes, weshalb sie hier gesondert genannt wird. Auch die generelle Rollenverteilung kann ungünstig gewählt worden sein, sodass die persönlichen Stärken der Teammitglieder nicht optimal genutzt werden können.

Lösung: Zur Vermeidung dieses Risikos ist eine ehrliche und gute Selbsteinschätzung eines jeden Teammitgliedes von größter Bedeutung. Auch die Kommunikation dieser Einschätzung gegenüber den anderen Teammitgliedern muss ehrlich und direkt erfolgen, damit stets klar ist, was für jeden einzelnen machbar ist und was nicht. Auch die Erfahrungsberichte der Gruppen früherer Semester kann bei der richtigen Einschätzung von Arbeitsaufwand und -verteilung helfen.

7. Schlechte Dokumentation

Problem: Auch eine mangelhafte Dokumentation kann zu Problemen führen, da sie beim Verständnis und der Nachvollziehbarkeit der einzelnen Projektschritte eine große Rolle spielt. Sie wird besonders beim Ausfall eines Teammitgliedes und bei weiteren Arbeit am Projekt durch andere Entwickler dringend benötigt und darf deshalb nicht vernachlässigt werden.

Lösung: Durch eine festgelegte, gute Dokumentationsstrategie, die auch regelmäßig überprüft wird, kann dieses Risiko ausreichend stark minimiert werden. Am wichtigsten ist das Augenmerk auf eine kontinuierliche Dokumentation, damit sich keine Fehler einschleichen.

8. Datenverlust

Problem: Da während des gesamten Projektes natürlich vorwiegend an digitalen Dokumenten gearbeitet wird, stellt auch der Datenverlust ein allgegenwärtiges Risiko dar. Der Verlust der Projektdaten oder Fragmenten dieser wäre ein enormes Risiko für den planmäßigen Erfolg des Projektes.

Lösung: Um auch dieses Risiko weitestgehend zu minimieren, werden unsere Projektdaten in einem redundanten System verwaltet und durch regelmäßige Backups gesichert. Ein Verlust aller Projektdaten wird hierdurch nahezu unmöglich, da jedes Teammitglied alle Daten vorliegen hat.



9. Fehlerhafte Abgabe / Endsoftware

Problem: Trotz aller Arbeit und konkreten Spezifikation der Anforderungen kann es passieren, dass die abgegebene Software am Ende gravierende Fehler enthält. Dieses Risiko gefährdet die Qualität der Software erheblich. Auch ungewollte „Side-effects“ könnten auftreten.

Lösung: Zur Vermeidung von groben Fehlern in der Software ist ein intensives Testen im Rahmen der Qualitätssicherung unabdingbar. Hierbei können schon vor Auslieferung Fehler erkannt, zurück verfolgt und beseitigt werden. Durch die Erstellung von Prototypen kann dies schon in frühen Entwicklungsstadien durchgeführt werden, womit die Anzahl und Auswirkung solcher Fehler im Endprodukt stark sinkt.

10. Technische Probleme

Problem: Da die Planung, Entwicklung und Implementierung von Softwareprojekten sowie die Kommunikation der Teammitglieder überwiegend computergestützt läuft, stellt auch die verwendete Technik ein gewisses Ausfallrisiko dar. Ohne die technischen Hilfsmittel wird der Projektprozess erheblich behindert und führt über aufwändige Umwege.

Lösung: Am meisten hilft hierbei die Erfahrung der Teammitglieder, insbesondere des technischen Assistenten, um die sorgfältige Auswahl und Pflege der benötigten Hardware und einen störungsfreien Betrieb der gesamten Technik sicherzustellen. Sollte es doch einmal zu Ausfällen kommen, ist für ausreichend Ersatzhardware gesorgt, sodass ein zügiger Austausch das effektive Arbeiten schnell wieder ermöglicht.

Rollenverteilung

Aufgabenbereich	Name
Projektbetreuung	Konrad Höffner
Projektleitung	Peter Pfahl
Recherche	Simon Vetter
Testing	Dennis Konrad
Technischer Assistent	Martin Grohmann
Modellierung	Felix Rauchfuß
Implementierung	Fabian Henze
Dokumentation und Qualitätssicherung	Lydia Lotzmann