

Risikoanalyse

Wie in anderen Bereichen des Lebens und der Wirtschaft ist auch die Entwicklung von Software mit Risiken verbunden. Um ein späteres Scheitern des Projektes zu verhindern ist es wichtig sich schon im Vorfeld Gedanken über diese Risiken zu machen und Strategien zu entwickeln um sie zu verhindern.

Risiko: Zeitmanagement

Problem: Aufgrund fehlender Erfahrung oder falschen Einschätzungen des Projektaufwands, aber auch durch anderweitige Verpflichtungen (Vorlesungen) kann es besonders im späteren Verlauf des Projektes (Implementierungsphase) aufgrund von Termindruck zu Zeitmangel kommen.

Strategie: Durch vorrausschauende Planung und rechtzeitiges Bearbeiten der Aufgaben kann diesem Risiko zum Teil entgegen gewirkt werden. Auch das Schaffen von Zeitpuffern kann bei Terminänderungen helfen.

Risiko: Aufgabenverteilung

Erklärung: Da nicht allen die Stärken und Schwächen eines jeden Gruppenmitgliedes bekannt sind, aber auch aufgrund mangelnder Selbsteinschätzung, kann es zu Fehlbesetzungen kommen, weil Ressourcen verschenkt werden.

Strategie: Offener Umgang miteinander und eine objektive Selbsteinschätzung helfen dabei, die benötigten Rollen optimal zu besetzen. Sollten sich im Verlaufe des Projektes dennoch Probleme herauskristallisieren, muss über einen möglichen Rollentausch nachgedacht werden.

Risiko: Personalausfall

Erklärung: Aufgrund von Krankheiten kann es immer vorkommen, dass ein Teammitglied kurzfristig über längere Zeit teilweise oder sogar vollständig ausfällt.

Strategie: Durch eine schnelle Bekanntgabe des Betroffenen kann eine Neuverteilung seiner Aufgaben diskutiert werden. Wichtige Gruppenrollen sollte daher doppelt belegt werden um Ausfällen zu kritischen Zeitpunkten optimal entgegenwirken zu können.

Risiko: Überforderung

Erklärung: Durch anfänglich mangelndes Wissen und dem damit verbundenem Lernaufwand, kann es durch den gegebenen Termindruck zu Überforderung einzelner Gruppenmitglieder kommen. Aber auch private Probleme können dies zur Folge haben. Der damit eingehende Motivationsverlust kann das ganze Projekt gefährden.

Strategie: Durch gegenseitiges Lehren (Tutoring) kann Wissen besser verteilt werden um die Gruppe auf eine gemeinsame Ebene zu bringen. Gegenseitiges Helfen bei Problemen kann ebenfalls Motivationsverlust vermindern.

Risiko: Mangelnde Kommunikation zwischen Auftraggeber und Projektgruppe

Erklärung: Eine unzureichende oder zu allgemein gefasste Auftragsbeschreibung kann zu Missverständnissen führen. Diese Fehlinterpretationen können im schlimmsten Fall zu einem falschen Endergebnis führen, welches den eigentlichen Anforderungen nichtmehr entspricht.

Strategie: Wichtige Anforderungen sollten am Anfang des Projekts ausführlich diskutiert werden. Mit Hilfe von Milestones, kann der Auftraggeber über den Verlauf der Arbeit informiert werden.

Risiko: Änderung der Anforderungen

Erklärung: Durch (z.B. wirtschaftliche) Umstände kann es zu einer Änderung der Anforderungen kommen. Diese können je nach Zeitpunkt einen Großteil des Projekts unnütz machen.

Strategie: Bei plötzlich auftretenden Änderungen muss im Notfall die Arbeit am Projekt eingestellt werden, damit keine Ressourcen verschwendet werden und um die neuen Anforderungen ausführlich besprechen zu können.

Risiko: Unstimmigkeiten zwischen Gruppenmitgliedern

Erklärung: Aufgrund mangelnder Kommunikation zwischen einzelnen Gruppenmitgliedern kann es zu Fehlinterpretationen der Arbeit anderer kommen, aber auch die Einarbeitung in diese wird erschwert. Dies kann zu Demotivation und einem schädlichen Gruppenklima führen. Aber auch unterschiedliche Lösungsansätze können wegen unzureichender Kommunikation zu Problemen führen.

Strategie: Gemeinsam vereinbarte Termintreffen können zum Finden einer gemeinsamen Lösung genutzt werden. Weiter hilft Pair Programming zum besseren „Kennenlernen“ und optimierten Arbeiten. Im Ernstfall hat sich die Gruppe an das Wort des Projektleiters zu halten, da dieser die Verantwortung für das Projekt trägt.

Risiko: Datenverlust

Erklärung: Aufgrund von Hard-, Softwareversagen oder auch menschlichem Versagen kann es zu Datenverlust kommen.

Strategie: Durch Backups kann dem entgegengewirkt werden (Mercurial).

Risiko: Systemausfall

Erklärung: Es besteht immer die Gefahr auf einen temporären Ausfall der örtlichen Infrastruktur (Strom, Internet u.Ä.), wodurch Teile der Gruppe behindert werden und es zu Datenverlusten kommen kann.

Strategie: Mercurial speichert die Fortschritte lokal auf dem Rechner, wodurch ein unabhängiges Arbeiten gewährleistet ist und nie das gesamte Projekt verloren geht.

Risiko: Planungs- und Entwicklungsfehler

Erklärung: Fehlentscheidungen in der Planungsphase können im weiteren Verlauf des Projekts (speziell in der Implementierungsphase) zu schwerwiegenden Problemen führen, so dass Teile oder sogar das gesamte Projekt neu geschrieben werden muss.

Strategie: Genaue und vorrausschauende Planung helfen dabei um den späteren Korrekturbedarf zu minimieren oder mögliche Probleme gar nicht erst auftreten zu lassen.

Risiko: Dokumentation

Erklärung: Mangelhafte und unzureichende Dokumentation kann die Korrektur von Fehlern behindern und verlängert die Einarbeitungszeit in den Quellcode. Aber auch die spätere Wartung und Wiederverwendbarkeit wird somit faktisch verhindert oder zumindest stark erschwert.

Strategie: Ein gemeinsam ausgearbeitetes Qualitätssicherungskonzept hilft dabei um ein Grundlegendes Gerüst für die Dokumentation zu schaffen. Auch sollten alle getroffenen Entscheidungen sorgfältig dokumentiert werden um getroffene Entscheidungen später nachvollziehbar zu machen.

Gruppenübersicht/Rollenverteilung

Rolle	Name
Projektleiter	Oliver Schurig
Technischer Assistent	Manuel Schönberner
Verantwortlicher für Recherche	Marc Wolff
Verantwortlicher für Modellierung	Christopher Themann
Verantwortlicher für Tests	Marc Wolff
Verantwortlicher für Implementierung	Daniel Abitz
Verantwortlicher für Qualitätssicherung und Dokumentation	Oliver Schurig