

## Risiken

Zu Beginn des Projektes galt es, mögliche Risiken im Projektverlauf zu identifizieren. Diese Risiken können sowohl technischer als auch persönlicher Natur sein. Während es recht kompliziert ist, die persönlichen Risiken im Vorfeld zu beseitigen sollen die technischen Risiken durch Proofs of Concept eliminiert werden.

### Persönliche Risiken

Da die beiden Projektteilnehmer seit 5 Semestern häufig zusammen Projekte bearbeiten war das finden von typischen persönlichen Problemen (auch solchen, die speziell in dieser Konstellation auftreten) eher einfach. Hinzu kommen noch einige Risiken nicht-technischer Natur, deren Auftreten sich nur schwer oder gar nicht beeinflussen lassen.

#### **Zeitmangel**

Zeitmangel ist in den meisten Projekten eines der Hauptrisiken. In diesem Projekt herrschen zusätzlich verschärfte Bedingungen vor, das das Wintersemester typischerweise kürzer ist, als das Sommersemester. Außerdem verlief die Ideenfindung zu Beginn des Projektes nicht so reibungslos wie gewünscht.

Ob tatsächlich Zeitmangel auftritt, lässt sich leider erst zu einem späteren Zeitpunkt im Projekt feststellen. Um hier entgegen zu wirken müssen unter Umständen einige, kleinere Features fallen gelassen um am Ende des Projektzeitraumes ein MVP (Minimum viable product) abliefern zu können.

#### **Krankheit**

Gerade zu Beginn der Winterzeit ist statistisch eine Erkrankung der Mitglieder des Projektteams wahrscheinlicher, deshalb muss auch hiermit gerechnet werden. Obwohl jedoch eine generelle Erkrankung eines Mitglied des Team wahrscheinlicher ist, ist es unwahrscheinlich, dass diese Erkrankung das Mitglied komplett unproduktiv macht (also z.B. durch eine sehr schwere Erkrankung) sondern lediglich eine Minderung der Produktivität zur Folge hat. Es wird damit gerechnet, dass der Teampartner diese geminderte Produktivität für einen gewissen Zeitraum ausgleichen kann. Sollte dies nicht möglich sein, muss auch hier über das streichen von Features nachgedacht werden.

#### **Räumliche Trennung der Gruppenmitglieder über Weihnachten/Neujahr mindert Produktivität**

Dies ist ein Risiko, dessen Auftrittswahrscheinlichkeit bei annähernd 100% liegt. Da die beiden Gruppenmitglieder die Weihnachtszeit in ihrer Heimat verbringen werden und diese etwa 300km auseinander liegen ist ein persönliches Treffen in dieser Zeit nicht ohne weiteres möglich.

Hier ist anzumerken, dass beide Gruppenmitglieder bereits Erfahrung mit dem Arbeiten von zu Hause aus und dem besprechen von Projekten über Kommunikationsanwendungen haben. Außerdem kann das Projekt so organisiert werden, dass möglichst viele Aufgaben in

die Weihnachtszeit fallen, die ein vergleichsweise geringes Maß an Kommunikation fordern.

### **Zu hohe Konzentration auf Details, so dass das Funktionieren der Anwendung an sich an Fokus verliert**

Dieses Risiko konnte aus vergangenen Projekten, die die beiden Teammitglieder bereits zusammen absolviert haben extrahiert werden.

Beide Teammitglieder haben sich in der Vergangenheit als recht perfektionistisch herausgestellt. So fiel es ihnen schwer eher unwichtige Funktionen in einem ausreichenden Zustand zu belassen.

Dieses Risiko soll hier durch regelmäßige gegenseitige Kontrolle des Fortschritts und Investigation der Gründe für zu langsamen Fortschritt minimiert werden.

### **Vernachlässigung der Dokumentation durch Tatendrang**

Auch dieses Risiko wurde in vergangenen Projekten deutlich. Beide Teammitglieder haben mehr Spaß daran, tatsächlich zu Programmieren als eine Dokumentation zu schreiben.

Auch dieses Risiko kann nur durch gegenseitige Kontrolle und das setzen von festen Zeiträumen, in denen die Dokumentation weiter voran gebracht werden soll, minimiert werden.

### **Es steht keine Hardware zum Testen der Anwendung zur Verfügung**

Dieses Risiko kann in seiner Auftrittswahrscheinlichkeit als relativ gering bewertet werden.

Beide Gruppenmitglieder besitzen ein aktuelles Android-Handy und einen vServer.

Außerdem besteht Zugang zu einem root-Server. Damit sollte das Testen mit tatsächlicher Hardware kein Problem darstellen.

In dem unwahrscheinlichen Fall, dass all diese Möglichkeiten zum Testen nicht zur Verfügung stehen ist das kann mit Emulatoren und Simulatoren getestet werden. Ein zu einem gewissen Grad lauffähiges und fehlerfreies Endprodukt ist damit garantiert.

- Zeitmangel wegen Erlernen der Clientseitigen Programmiersprache
- Es wird zu viel Zeit mit Design verbracht

## **Technische Risiken**

Wie oben bereits erwähnt ist es möglich, die technischen Risiken bereits im Vorfeld zu einem gewissen Grad zu eliminieren bzw. auf ihre Auftrittswahrscheinlichkeit zu testen. Eine genauere Erläuterung der technischen Risiken und die Fail/Exit-Kriterien werden in den Proofs of Concept dargelegt.

Der Vollständigkeit halber sind sie hier nur kurz aufgelistet.

- Wir kommen mit der Clientseitigen Programmiersprache nicht zurecht
- Es wird zu viel Zeit mit Design verbracht
- Board-System kann unter Android nicht realisiert werden
- Notifications über Google Cloud Messaging können nicht realisiert werden