

## 7. Risiken

In diesem Kapitel werden die potenziellen Risiken dargelegt, die während des Entwicklungsprozesses auftreten oder von projektspezifischer Natur sind. Die Ergebnisse des PoC liefern Anhaltspunkte darüber, welche Risiken in welchem Umfang auftreten können und ob diese minimiert werden können.

### Projektspezifische Risiken

#### Netzwerkprobleme

Ein verteiltes System setzt ein zuverlässiges Netzwerk voraus. Ansonsten kann die Anwendung nur unvollständig oder überhaupt nicht ausgeführt werden.

Umgang: Siehe PoC

#### Nutzungsrechte von Daten eines Fremdanbieters

Bei der Nutzung von APIs auf fremde Rezeptquellen müssen die Nutzungsrechte berücksichtigt werden. Eine Verletzung dieser Rechte kann zu rechtlichen Konsequenzen führen.

*Umgang:* Wenn ein Betreiber die Nutzung und Manipulation seiner Daten zustimmt (siehe PoC), gilt das Risiko schon als beseitigt. Ansonsten muss geklärt werden, ob die Community eigene Rezepte in das System einpflegen kann, was zum momentanen Zeitpunkt nicht zum Leistungsumfang des Systems gehört.

#### Datenanreicherung

Es könnten Probleme bei der Verarbeitung der Rezepte auftreten, wie die inkorrekte Umwandlung der Einheiten.

*Umgang:* Siehe PoC

### **Einbindung von APIs**

Ein hohes Risiko besteht, wenn Daten von Fremdanbietern zwar kostenlos oder gegen Gebühr verwendet werden können, diese jedoch nicht die nötigen Schnittstellen für eine Anbindung in das System anbieten.

### **Gebrauchsuntaugliche UI**

Der Benutzer muss das Gefühl haben, dass er Interaktionen mit dem System effektiv, effizient und zufriedenstellend erledigen kann. Ein System, welches zwar alle funktionalen Anforderungen erfüllt, aber eine mangelhafte Usability aufweist, ist für den Benutzer unbrauchbar.

Man muss dazu anmerken, dass dieser Aspekt nicht zwangsläufig den Entwicklungsprozess behindern muss.

## **Projektinterne Risiken**

### **Schlechtes Zeitmanagement**

Erfordert eine Entwicklungsphase unerwartet mehr Zeit, müssen Wege gefunden werden, die ein Überschreiten der finalen Deadline verhindern. Erfordert die Fertigstellung eines Meilensteins unerwartet mehr Zeit, muss man damit rechnen, dass in nachfolgenden Meilensteinen ungenügend Zeit und Arbeit investiert wird. Im schlimmsten Falle droht das Projekt zu scheitern.

*Umgang:* Der Entwicklungsprozess muss sorgfältig geplant. Dazu gehört das Erstellen eines Projektplans mit den wichtigsten Meilensteinen sowie die Einplanung von Pufferzeiten.

### **Ausfall eines Teammitgliedes**

Durch den Ausfall eines Teammitglieds während des Entwicklungsprozesses, kann es zu erheblichen Verzögerungen bei der Entwicklung kommen. Die Folgen wären ein zeitlicher Rückschlag bis zur Fertigstellung der Software.

*Umgang:* Da bereits bei der zeitlichen Knappheit gewisse Pufferzeiten und Arbeitsschritte geplant sind, können diese Zeiten für die Erreichung des Projekterfolges genutzt werden. Der Fokus sollte dann nur noch auf den Kernelementen des Systems liegen.

### **Unerfahrenheit im Bereich der App-Entwicklung**

Mindestens einer der Projektteilnehmer benötigt wahrscheinlich etwas mehr Zeit, um sich in die App-Entwicklung einzuarbeiten, was Zeitkosten mit sich bringt. Wird zu viel Zeit in die Aneignung und die Entwicklung des Systems investiert, so wirkt sich das negativ auf den Projekterfolg aus.

*Umgang:* Zeit für Einarbeitung in das Thema Appentwicklung muss eingeplant werden. Die Entwicklung der Proof Of Concepts prüft dieses Risiko.