Lastenheft Quiz Webanwendung

# Zielbestimmung

Die zu entwickelnde Software soll das individuelle Wissen der Studenten in Form von einem Quiz auf verschiedene Arten Testen.

## Produkteinsatz

Das Quiz wird als Webanwendung realisiert und soll nach Fertigstellung in jedem gängigen Browser ausführbar sein. Weiterhin, sollen die Ansichten für verschiedene Bildschirmgrößen angepasst werden (responsive Design).

## Vergleichbare Anwendungen

### Kahoot:

Kahoot ist ebenfalls eine Webanwendung um selbst Quizze zu erstellen, welche dann beispielsweise im Unterricht eingesetzt werden können, um diesen interaktiver zu gestalten und den Lernfortschritt der Klasse zu überprüfen.

## Aufbau des Dokuments

Im Folgenden werden der aktuelle Stand, sowie der Soll-Zustand näher erläutert. Weiterhin werden funktionale und nicht-funktionale Anforderungen geklärt.

# Ist-Zustand

Studenten erstellen sich zur Prüfungsvorbereitung gerne Fragenkataloge, welche sie abarbeiten können um den Stoff zu verinnerlichen und zu überprüfen wie gut sie diesen bereits beherrschen. Diese Kataloge werden oft in der Form von Textdokumenten erstellt und werden von den Studierenden auch gern untereinander ausgetauscht.

Professoren können bereits Webanwendungen wie Kahoot nutzen um ihre Vorlesung interaktiver zu gestalten und gleichzeitig die Studenten zum Lernen motivieren und ihren aktuellen Kenntnisstand zu überprüfen. Jedoch lassen sich mit Kahoot nur Multiple-Choice Fragen erstellen und die Konfigurationsmöglichkeiten für die einzelnen Fragen sind gering.

# Soll-Zustand

Die Webanwendung soll dazu dienen, dass Studenten und Professoren ihre Fragenkataloge mit der Webanwendung erstellen können. Professoren können in den Vorlesungen Quizrunden starten, an denen jeder Student einmal teilnehmen darf, um den Kenntnisstand der Studenten zu überprüfen.

Die Studenten sollen ihre Kataloge auch mit anderen Studenten teilen können und können anhand der Highscore-Liste sehen, wie gut sie im Vergleich zu anderen abschneiden.

# Funktionale Anforderungen

* Benutzerverwaltung
* Erstellen eigener Kataloge, beispielsweise für jede Vorlesung ein eigener Fragenkatalog
* 3 Fragenkategorien:
  + Schätzfragen
  + Multiple-Choice-Fragen
  + Eingabe der Antwort durch den Nutzer
* Erstellen eigener Fragen für einen Katalog
  + Auswahl Fragekategorie
  + Auswahl der Zeit um Frage zu beantworten
  + Bei Multiple-Choice müssen Antwortmöglichkeiten gegeben werden
  + Angabe korrekte Antwort
* Veröffentlichen / Teilen eines Fragenkatalogs
* Challenge-Runde, an der jeder Nutzer nur einmal teilnehmen darf
* „Highscore“-Listen
* Freundeslisten

# Nicht-funktionale Anforderungen

* Das Frontend der Webanwendung muss mit Vue.js umgesetzt werden
* Das Backend der Webanwendung muss mit Node.js umgesetzt werden