



# PFLICHTENHEFT

htw-quiz

Exposee

Webapplikation für ein Quizspiel für Studierende der htw im Bereich Informatik

Marius Backes und Tobias Meiser

# 0.Inhalt

0.	Inhalt.....	1
1.	Zielbestimmung.....	2
1.1.	Musskriterien .....	2
1.2.	Wunschkriterien .....	3
1.3.	Abgrenzungskriterien .....	3
2.	Produkteinsatz.....	4
2.1.	Anwendungsbereiche.....	4
2.2.	Zielgruppen.....	4
2.3.	Betriebsbedingungen .....	4
3.	Produktübersicht.....	5
3.1.	Software .....	5
3.2.	Hardware .....	5
3.3.	Orgware.....	5
4.	Produktfunktionen .....	6
4.1.	Benutzerfunktionen.....	6
4.2.	Spielfunktionen .....	6
5.	Produktdaten.....	8
6.	Produktleistungen .....	9
7.	Qualitätsanforderungen .....	10
8.	Benutzeroberfläche.....	11
9.	Nichtfunktionale Anforderungen .....	12
10.	Technische Produktumgebung.....	12
10.1.	Software .....	12
10.2.	Hardware .....	12
10.3.	Orgware.....	12
10.4.	Produktschnittstellen .....	12
11.	Testszenarien und Testfälle.....	13

# 1. Zielbestimmung

Das zu erstellende Produkt ist eine Webapplikation für ein Quizspiel und soll für die Studierenden und Professoren der htw saar entwickelt werden. Hierbei wird in erster Linie auf das eigentliche Spiel und die angemeldeten Benutzer, sowohl Studierende als auch Professoren, eingegangen.

## 1.1. Musskriterien

- Nutzer
  - Der Benutzer kann sich beim System registrieren
  - Der Benutzer kann sich sowohl im System anmelden als auch abmelden
  - Dem Nutzer ist es möglich seine persönlichen Daten jederzeit zu ändern und den Bedürfnissen anzupassen
- Professoren
  - Professoren können Spiele anlegen
  - Professoren können die Fragen für ein Spiel erstellen, bearbeiten und löschen
  - Professoren können ein Spiel starten und anschließend können Studierende dem Spiel beitreten
  - Professoren können sich die Nutzer-Scores zu einzelnen Spielen anschauen
  - Professoren können beim Anlegen eines Spiels eine Start- und Endzeit des Quiz auswählen. Das Spiel kann nur in dieser Zeit von Studenten genutzt werden.
- Studierende
  - Der Benutzer hat die Möglichkeit Fragekataloge anzulegen
  - Für einen Fragekatalog können beliebig viele Fragen erstellt werden
  - Beim Spielen erzielte Resultate können vom Spieler eingesehen werden
  - Es kann eine Highscore aller Spieler eingesehen werden
  - Der Benutzer hat Möglichkeit ein Spiel alleine zu starten, ohne ein interaktives Spiel, welches vom Professor gestartet wurde
- Spiel
  - Für das Spiel wird ein Fragekatalog ausgewählt, in dem sich eine gewisse Anzahl an Fragen befindet
  - Beim Spielen selbst, gibt es verschiedene Möglichkeiten eine Antwort zu wählen
    - Eingabe der Antwort
    - Auswählen einer richtigen Antwort aus 4 Möglichkeiten
    - Auswählen von mehreren richtigen Antworten, wobei der Nutzer nicht weiß wie viele davon richtig sind.
  - Jede Frage hat ein bestimmtes Zeitlimit, welches bei Erstellen der Frage gewählt werden kann.
  - Beim Anlegen der Fragen, ist bei Auswahlfragen immer die erste eingegebene Antwort die korrekte Antwort. Beim Spielen werden die Antwortmöglichkeiten in zufälliger Reihenfolge angezeigt.

## 1.2. Wunschkriterien

- App
  - Das Spiel wird mit Hilfe eines Plugins für die mobilen Plattformen Android und iOS zur Verfügung gestellt.
- Studierende
  - Studierende können Freundeslisten erstellen
  - Studierende können ihren Fragenkatalog anderen Nutzern freigeben und diesen dann gemeinsam bearbeiten oder für Quizrunden nutzen.
- Professoren
  - Professoren können sich spielübergreifend die Scores der Studenten anschauen, für alle Spiele die sie selbst gestartet haben.
- Fragen
  - Die Fragen sollen auch im HTML-Format gestellt werden können.

## 1.3. Abgrenzungskriterien

→ Noch sind keine Abgrenzungskriterien vorhanden

## 2. Produkteinsatz

Im Folgenden werden die Bereiche, in welchen das Produkt nach Fertigstellung eingesetzt werden soll näher erläutert

### 2.1. Anwendungsbereiche

Der Anwendungsbereich zielt darauf ab, dass entweder der Professor ein Spiel starten und Studierende dem Spiel interaktiv beitreten können, oder dass Studierende ein Spiel für sich alleine starten um ihr gelerntes Wissen der Vorlesung auf den Prüfstand zu stellen.

### 2.2. Zielgruppen

Die Zielgruppe an welche sich das Spiel in erster Linie wendet sind die Studierenden, denn diese können ihr Wissen mit Hilfe dieses Quiz überprüfen oder verbessern, um sich auf eine Klausur vorzubereiten. Des Weiteren gehören auch die Professoren der HTW der Zielgruppe an. Diese können sich mit der App einen Überblick über den aktuellen Kenntnisstand der Studenten verschaffen.

### 2.3. Betriebsbedingungen

Das System verfügt im Hintergrund über eine Datenbank, in der die einzelnen Fragekataloge, oder auch Benutzerdaten gespeichert werden.

Die Datenbank soll zu jeder Uhrzeit erreichbar sein. Weiterhin wird auf den sensiblen Umgang mit Benutzerdaten geachtet, so ist es unbedingt notwendig die Anmeldedaten eines Benutzers verschlüsselt zu speichern.

Um für Ausfallsicherheit zu sorgen, soll in regelmäßigen Abständen eine Kopie der Datenbank gemacht werden.

## 3. Produktübersicht

Da es sich bei dem Produkt um eine Webapplikation handelt, kann es Betriebssystemunabhängig betrieben werden. Weiterhin ist es auch möglich das Produkt von mobilen Geräten, wie Smartphone oder Tablet zu nutzen.

### 3.1. Software

- Client
  - Es steht ein Browser mit grafischer Oberfläche zur Verfügung
- Server
  - Die Backendumgebung wird mit node.js realisiert
  - Als Datenbank kommt MySQL zum Einsatz

### 3.2. Hardware

- Client
  - Der Client muss über ein internetfähiges Gerät verfügen
- Server
  - Der Server benötigt eine dauerhafte Internetverbindung
  - Es muss ausreichend Rechenleistung und Festplattenspeicher zur Verfügung stehen

### 3.3. Orgware

Eine dauerhafte Internetanbindung muss unbedingt gewährleistet sein

## 4. Produktfunktionen

Im Folgenden werden die verschiedenen Funktionen, die das Produkt bereitstellen soll näher erläutert.

### 4.1. Benutzerfunktionen

/F010/ Registrieren: Ein beliebiger Nutzer hat die Möglichkeit sich über die Weboberfläche einen Account anzulegen. Hierfür sind verschiedene Angaben notwendig:

- Gewünschte Kennung
  - Gewünschter Benutzername
  - Gewünschter Passwort
- Vorname
- Name
- E-Mail-Adresse

Die Registrierung gilt als erfolgreich, wenn der Nutzer eine Mail als Bestätigung erhält und sich anschließend im System anmelden kann.

/F020/ Anmelden: Nach der Registrierung, kann sich der Nutzer mit Hilfe seiner E-Mail-Adresse und seinem festgelegten Passwort im System anmelden. Nach erfolgreicher Prüfung der Daten erfolgt seine Weiterleitung.

/F030/ Abmelden: Der Nutzer kann sich wieder vom System abmelden

/F040/ Passwort ändern: Ein angemeldeter Benutzer hat im System eine Option sein momentanes Passwort zu aktualisieren.

/F050/ Benutzerinformationen anzeigen: Ein Benutzer kann die von ihm angegebenen Informationen anzeigen.

/F060/ Benutzerinformationen ändern: Der Nutzer kann seine Daten nach der Anmeldung auch ändern.

### 4.2. Spielfunktionen

/F110/ Fragenkatalog anlegen: Nutzer können sich eigene Fragenkataloge anlegen. Dazu wird ein Name für den Katalog gewählt, beispielsweise „Rechnerarchitektur“

/F120/ Fragenkatalog bearbeiten: Nutzer können den Katalog bearbeiten. (Nutzer können neue Fragen hinzufügen, vorhandene bearbeiten und löschen.)

/F130/ Fragenkatalog löschen: Nutzer können ihren Katalog löschen.

/F140/ Quiz starten: Nutzer können einen Fragenkatalog spielen und Professoren können Spiele starten.

/F150/ An Spiel teilnehmen: Studenten können an Spielen teilnehmen.

/F160/ Ergebnisse ansehen: Alle am Spiel Beteiligten können sich die Endergebnisse ansehen.

/F170/ Fragenkatalog teilen: Der Ersteller eines Fragenkatalogs kann dieses für andere Studenten freigeben, sodass diese den Katalog auch spielen oder sogar bearbeiten können.



## 5.Produktdaten

/D010/ Benutzerdaten: Alle Informationen zu einem Benutzer.

- BenutzerID (eindeutig)
- Kennung
  - Benutzername
  - E-Mail (eindeutig)
  - Vorname
  - Name
  - Passwort (verschlüsselt)
- Sonstige Daten
  - Registrierungsdatum
  - Datum des letzten Logins
  - Absolvierte Spiele
  - Erreichte Punkte
  - Status (Admin oder Benutzer)

/D020/ Fragedaten: Alle Informationen zu einer Frage sind hier enthalten.

- FrageID (eindeutig)
- Fragetext
- Art der Frage
- Antwortmöglichkeiten (optional, nur bei Multiple Choice Fragen benötigt)
- Korrekte Antwort/en

/D030/ Spieldaten: Alle Daten, welche für ein gesamtes Spiel benötigt werden

- FragenkatalogsID oder SpielID (eindeutig)
- Katalogs- bzw. Spielbeschreibung (z.B. Programmierung 1 / Metasprachen Test 2)
- FragenIDs
- BenutzerID
- Erreichte Punkte

/D040/ Highscore Daten: Alle Daten, die in der Highscore Liste zusammengeführt werden

- SpielID (eindeutig)
- Benutzernamen der Spieler
- Punkte der Spieler

## 6. Produktleistungen

/L100/ Zeitlimit: Für eine Frage kann beim Erstellen ein Schwierigkeitsgrad eingestellt werden und je nach Schwierigkeitsgrad hat der Spieler eine gewisse Anzahl an Sekunden um die Frage zu beantworten

/L200/ Antworteingabe: Hat der Nutzer falsche Sonderzeichen oder ähnlich in einer Antwort gewählt, so wird im sofort eine Fehlermeldung angezeigt.

/L300/ Absenden von Antworten: Hat ein Nutzer eine Antwort von Hand eingegeben, so wird der Button zum Absenden der Antwort erst dann aktiviert, wenn der Nutzer auch eine Antwort eingegeben hat.

## 7. Qualitätsanforderungen

In der nachfolgenden Tabelle werden die Qualitätsanforderungen, die an die Webapplikation gestellt werden weiter eingegrenzt.

	SEHR WICHTIG	WICHTIG	WENIGER WICHTIG	UNWICHTIG
ROBUSTHEIT			X	
ZUVERLÄSSIGKEIT		X		
KORREKTHEIT	X			
BENUTZERFREUNDLICHKEIT	X			
EFFIZIENZ		X		
ERWEITERBARKEIT		X		
KOMPATIBILITÄT	X			

- **Robustheit:** Robustheit bedeutet, dass das System gegen Eingriffe Dritter gewappnet ist.
- **Zuverlässigkeit:** Unter Zuverlässigkeit wird die ständige Erreichbarkeit des Systems verstanden.
- **Korrektheit:** Der fehlerfreie Ablauf der Programmfunktionen muss stets gewährleistet sein.
- **Benutzerfreundlichkeit:** Die Anwendung sollte intuitiv benutzbar sein.
- **Effizienz:** Es soll kein unnötig großer Ressourcenaufwand entstehen.
- **Erweiterbarkeit:** Die Software soll so entwickelt werden, dass sie zu jedem Zeitpunkt leicht erweiterbar und anpassbar ist.
- **Kompatibilität:** Die Anwendung soll auf sämtlichen Endgeräten und gängigen Browsern lauffähig sein.

## 8. Benutzeroberfläche

Die Benutzeroberfläche der Anwendung wird mit vue.js realisiert. Beim ersten Start der Webapplikation wird dem Nutzer eine Eingabemaske zur Registrierung angezeigt. Nach erfolgreichem Anmelden des Nutzers wird der Hauptbildschirm der Anwendung angezeigt. Dieser Screen wird ab dem Anmelden auch immer beim Start der Anwendung angezeigt werden, sofern der Nutzer sich nicht ausloggt. In diesem Screen ist es dem Nutzer möglich neue Fragenkataloge zu erstellen, vorhandene Kataloge zu bearbeiten oder Spielrunden zu Starten. Alle bereits verfügbaren Kataloge werden hier angezeigt und als Button dargestellt.

Sollte der Nutzer eine Quizrunde spielen, so wird ihm die jeweilige Frage, die verbleibende Zeit und im Falle einer Multiple-Choice Frage auch die Antwortmöglichkeiten angezeigt. Die Antwortmöglichkeiten werden wie die Fragenkataloge als anwählbare Buttons realisiert, sodass der Nutzer diese direkt auswählen kann. Sollte die Frage keine Multiple-Choice sondern eine Schätzfrage oder Frage nach einer konkreten Antwort sein, so wird dem Nutzer ein Textfeld angezeigt und sobald dieses Feld angewählt wird, wird die Bildschirmtastatur zur Eingabe der Antwort aufgerufen.

Nach Abschluss einer Frage wird dem Nutzer Feedback zur jeweiligen Frage gegeben. So wird die vom Nutzer gegebene Antwort sowie die korrekte Antwort und die erhaltenen Punkte angezeigt.

Des Weiteren wird in dem Hauptbildschirm ein Menü angeboten, über welches beispielsweise zu den Benutzerinformationen und den Anwendungseinstellungen navigiert werden kann.

## 9. Nichtfunktionale Anforderungen

Als nichtfunktionale Anforderungen wurde vorgegeben eine Webapplikation zu entwickeln. Diese Webapplikation soll im Frontend vue.js und am Backend node.js verwenden. Da es im Browser läuft, ist es weitgehend Plattformunabhängig. Lediglich ein grafischer Browser muss zur Verfügung stehen, da im System auch grafische Anzeigen vorhanden sind.

Alle Daten der angemeldeten Benutzer werden verschlüsselt und sind nicht einsehbar für Dritte. Nur der jeweilige Benutzer hat die Möglichkeit seine eigenen Benutzerdaten zu ändern.

## 10. Technische Produktumgebung

### 10.1. Software

- **Client**
  - Webbrowser
- **Server**
  - Node.js Https Server
  - MySQL Datenbank

### 10.2. Hardware

- **Client**
  - Der Client muss über ein internetfähiges Gerät verfügen
- **Server**
  - Der Server benötigt eine dauerhafte Internetverbindung
  - Es muss ausreichend Rechenleistung und Festplattenspeicher zur Verfügung stehen

### 10.3. Orgware

Eine dauerhafte Internetanbindung muss unbedingt gewährleistet sein

### 10.4. Produktschnittstellen

Es sind keine Schnittstellen zu anderen Produkten geplant.

# 11. Testszenarien und Testfälle

Für alle vorhandenen Produktfunktionen werden auch entsprechende Testfälle erstellt. Mit Hilfe dieser Tests werden die verschiedenen Produktfunktionen automatisiert überprüft.

/T010/ Registrieren: Der Nutzer Max Mustermann registriert sich mit dem Benutzernamen „max\_mustermann“, der E-Mail-Adresse „[max@mustermann.de](mailto:max@mustermann.de)“ und dem Passwort „MusterM@nnP@ssW0rd!“

/T020/ Anmelden: Max Mustermann meldet sich mit den vorher angelegten Benutzerdaten beim System an.

/T030/ Abmelden: Max Mustermann loggt sich aus dem System aus.

/T040/ Passwort ändern: Max Mustermann aktualisiert sein Passwort. Neues Passwort wird: „m@XNeuesP@ssw0rd!“

/T050/ Fragekatalog anlegen: Max Mustermann legt einen Fragekatalog mit der Bezeichnung „Testkatalog“ an.

/T060/ Fragekatalog bearbeiten: Der erstellte Fragekatalog wird von Max Mustermann bearbeitet. Hierfür werden 2 neue Fragen zum Katalog hinzugefügt.

/T070/ Fragekatalog löschen: Es wird ein Fragekatalog mit dem Titel „Löschender Katalog“ angelegt und anschließend wieder entfernt.

/T080/ Quiz starten: Der erstellte Fragenkatalog wird von Max Mustermann gespielt.

/T090/ Ergebnisse ansehen: Die erzielten Resultate können von Max Mustermann begutachtet werden.