

# Software–Projekt 2 2014 (RE SWP)

VAK 03-BA-901.02

## Anforderungsspezifikation

Schibboleth



Patrick Hollatz	phollatz@tzi.de	2596537
Tobias Dellert	tode@tzi.de	2936941
Tim Ellhoff	tellhoff@tzi.de	2520913
Daniel Pupat	dpupat@tzi.de	2703053
Olga Miloevich	halfelv@tzi.de	2586817
Tim Wiechers	tim3@tzi.de	2925222

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung (Patrick)</b>	<b>4</b>
1.1	Zweck	4
1.2	Rahmen	5
1.3	Definitionen, Akronyme und Abkürzungen	5
1.4	Referenzen	5
1.5	Übersicht über das Dokument	5
<b>2</b>	<b>Allgemeine Beschreibung</b>	<b>6</b>
2.1	Ergebnisse der Ist-Analyse (Tim E.)	6
2.1.1	Erstes Kundengespräch vom 08.05.2014	6
2.1.2	Auswertung des Kundengesprächs	7
2.1.3	Analyse eines ähnlichem Systems (Daniel Pupat)	7
2.2	Produktperspektive (Tim W. und Tim E.)	18
2.2.1	Systemschnittstellen	18
2.2.2	Benutzerschnittstelle	18
2.2.3	Hardwareschnittstellen	18
2.2.4	Softwareschnittstellen	18
2.2.5	Kommunikationsschnittstellen	18
2.2.6	Speicherbeschränkung	18
2.2.7	Operationen (Betriebsmodi)	19
2.2.8	Möglichkeiten der lokalen Anpassung	19
2.3	Anwendungsfälle (Daniel)	19
2.4	Charakteristika der Benutzer (Tim W.)	20
2.5	Einschränkungen (Tim E.)	21
2.5.1	Technische Rahmenbedingungen	23
2.5.2	Gesetzliche Rahmenbedingungen	24
2.5.3	Sicherheitskritische Aspekte	24
2.6	Annahmen und Abhängigkeiten (Tim E.)	24
2.7	Ausblick (Tim E.)	24
<b>3</b>	<b>Detaillierte Beschreibung</b>	<b>25</b>
3.1	Datenmodell (Tobias)	25
3.1.1	Datenmodellbeschreibung	25
3.2	Anwendungsfälle (Tobias und Daniel)	26
3.3	Aktionen (Tim W.)	33
3.3.1	App.Abmelden()	33
3.3.2	App.Anmelden(nutzername, passwort)	34
3.3.3	App.Einstellungen ändern(neuerNutzername, neuesPasswort)	34
3.3.4	App.Einstellungen aufrufen()	34
3.3.5	App.Endergebnis bestätigen()	34
3.3.6	App.Frage beantworten(antwort)	35

3.3.7	App.Gegner herausfordern(nutzer)	35
3.3.8	App.Herausforderung annehmen(herausforderung)	35
3.3.9	App.Joker nutzen(art)	35
3.3.10	App.Neues Spiel starten(modus)	36
3.3.11	App.Offline-Modus starten()	36
3.3.12	App.Online-Modus starten()	36
3.3.13	App.Rangliste aufrufen()	36
3.3.14	App.Registrieren(nutzername, passwort)	37
3.3.15	Website.Anmelden(nutzername, passwort)	37
3.3.16	Website.Abmelden()	37
3.3.17	Website.Frage bearbeiten(id, frage, antworten, kategorie)	37
3.3.18	Website.Frage hinzufügen(frage, antworten, kategorie)	38
3.3.19	Website.Frage löschen(id)	38
3.3.20	Website.Fragenliste aufrufen()	38
3.3.21	Website.Nutzerliste aufrufen()	38
3.3.22	Website.Nutzer löschen(id)	39
3.3.23	Website.Nutzernamen ändern()	39
3.4	Softwaresystemattribute Tim E.)	39

## Version und Änderungsgeschichte

Version	Datum	Änderungen
1.0	21.05.2014	Analyse ähnlicher Systeme.
1.1	22.05.2014	Anwendungsfälle
1.2	27.05.2014	Ist-Analyse
1.3	28.05.2014	Produktperspektiven
1.4	29.05.2014	Einschränkungen
1.5	30.05.2014	Annahmen und Abhängigkeiten
1.6	30.05.2014	Ausblick
1.7	30.05.2014	Charakteristika der Benutzer
1.8	31.05.2014	Softwaresystemattribute

**Wichtiger Hinweis:** Diese Anforderungsspezifikation wurde zu einem Teil aus verschiedenen Dokumententeilen der Anforderungsspezifikation unserer Gruppenmitglieder aus dem Wintersemester 2013/14 erstellt (Gruppe *IT-REVOLUTION*). Einige Teile wurden komplett übernommen, andere überarbeitet bzw. angepasst.

Diese Vereinbarung haben wir in der Kick-Off-Veranstaltung für RE SWP 2014 mit dem Veranstalter Dr. Karsten Hölscher getroffen.

## 1 Einleitung (Patrick)

### 1.1 Zweck

Zweck dieser Anforderungsspezifikation ist es, die Funktionalität und für unser Produkt bestehende Rahmenbedingungen zu spezifizieren. Die Rahmenbedingungen sind in dem Kapitel 2.5 Einschränkungen detailliert beschrieben. Sinn dieser Spezifikation ist es, dass jeder Entwickler, der an dieser Software beteiligt ist, genau weiß wie die Software gestaltet werden, und was für Funktionen sie enthalten soll.

Dieses Dokument dient zusätzlich als Vertragsgegenstand, weil es alle mit dem Kunden besprochenen Leistungen schriftlich festhält, und gemeinsam mit dem Angebot auch alle Punkte betreffend der Kosten angibt. Bei Nicht-Erbringung der Leistung treten gesetzliche Regelungen in Kraft, wie z.B. eine Kostenreduktion oder die Nachlieferung der fehlenden bzw. nicht funktionsfähigen Bestandteile.

## 1.2 Rahmen

Es soll eine Quiz-Applikation für Android Geräte entwickelt werden, welche da Spielen gegen andere Menschen ermöglicht. Die interne Umsetzung dieser Funktionen obliegt unserer Entscheidung, allerdings soll die Software für nicht Entwickler intuitiv benutzbar sein.

Zum Produktumfang gehört neben der Software ein Handbuch, das die Verwendung der Software so erklärt, dass jeder diese verstehen kann. Außerdem wird eine Installationsanleitung für die Software mitgeliefert.

## 1.3 Definitionen, Akronyme und Abkürzungen

## 1.4 Referenzen

## 1.5 Übersicht über das Dokument

Das Dokument beinhaltet im ersten Abschnitt eine Einleitung, die den Zweck des Dokuments beschreibt, den Rahmen des zu entwickelnden Produkts, sowie eine Kurzbeschreibung der auftauchenden Begrifflichkeiten, Referenzen und zuletzt diesen Abschnitt, der eine Übersicht über das Dokument liefert.

Der zweite Abschnitt beinhaltet eine allgemeine Beschreibung der Software, deren Inhalt folgender ist: Ist-Analyse, die Beschreibungen der Produktperspektive, Anwendungsfälle, Spielelemente, Regeln, Persona, Einschränkungen, Annahmen und Abhängigkeiten, Ausblick.

Der dritte Abschnitt ist eine detaillierte Beschreibung, welche aus einem Datenmodell, einer detaillierten Beschreibung der Anwendungsfälle, den dazugehörigen Aktionen, einer Beschreibung der Entwurfseinschränkungen, der Softwaresystemattribute sowie weitere Anforderungen besteht.

Im vierten und letzten Abschnitt befindet sich ein Anhang, der Dokumente enthält, die zur Entwicklung dieses Dokuments dienlich gewesen sind.

## 2 Allgemeine Beschreibung

### 2.1 Ergebnisse der Ist-Analyse (Tim E.)

Das Rektorat der Universität Bremen möchte aus Eigeninteresse und auch als Werbung Menschen und insbesondere Studieninteressierte, aber auch Studierende und Mitarbeitern die Uni in drei Bereichen näher bringen: Campusleben, Studium und Forschung. Dabei soll insbesondere der Bezug zum Studienort hergestellt werden.

Um dies zu erreichen, möchte die Universität gerne eine Quiz-App veröffentlichen und bereitstellen, die von den oben stehenden Interessengruppen gespielt werden kann und Fragen aus den drei Kernbereichen zur Verfügung stellt.

Bisher gibt es zwar eine Campus App der Universität, die aber andere Ziele verfolgt und in erster Linie nur für Studierende gedacht ist. Eine spezielle Quiz-App, zugeschnitten auf die Belange der Uni, gibt es allerdings noch nicht.

Unser Ziel ist es, genau eine solche App zu entwickeln, die benutzerfreundlich, leicht zu bedienen ist und durch ihr ansprechendes GUI-Design auch noch Spaß machen soll. Darüber hinaus wird es eine Administrationswebseite geben, auf der grundlegende Einstellungen konfiguriert werden können und auch eine redaktionelle Bearbeitung der Fragen und Antworten stattfinden kann.

Dabei werden die Fragen aus den drei Kategorien von der Pressestelle der Universität zur Verfügung gestellt.

Um die genauen Anforderungen und Kundenwünsche an das System über die vom Veranstalter des Software-Projekts 2 erhobenen Mindestanforderungen hinaus zu erfassen, haben wie ein Kundengespräch durchgeführt.

#### 2.1.1 Erstes Kundengespräch vom 08.05.2014

Am Donnerstag, den 08.05.2014, um 10:00 Uhr begann unser erstes Kundengespräch mit der Vertreterin des Kunden, Frau Sprindt von der Pressestelle der Universität. In den Tagen zuvor hat die Gruppe Ideen zu einem Fragenkatalog gesammelt, der dann am Vortag der Besprechung fertiggestellt wurde. Er beinhaltet eine Auflistung aller Mindestanforderungen, zu denen Unklarheiten bezüglich des Realisierungsvorgangs notiert wurden.

Das offizielle Gespräch begann pünktlich um 10:00 Uhr. Leider gab es keine wirkliche Ordnung bei den gestellten Fragen, was es etwas schwer machte, die einzelnen Vorgänge strukturiert zu durchdenken.

Grundsätzlich soll sich die App an die bekannte Quiz-App *Quizduell* anlehnen bzw. als eine Art Vorbild genommen werden. Genauer dazu findet man im Abschnitt [2.1.3](#), wo die Analyse eines bestehenden Systems vorgenommen wird.

### 2.1.2 Auswertung des Kundengesprächs

Es haben sich im Kundengespräch jedoch keine großen Überraschungen oder unerwartete Kundenwünsche oder -änderungen ergeben, die über die Mindestanforderungen besonders hinaus gehen. Einige Besonderheiten sollen aber im Folgenden näher beschrieben werden.

Man soll die App auch ohne Registrierung spielen können, allerdings nur lokal auf einem Gerät. Man hat dann nicht die Möglichkeit, später woanders weiterzuspielen. Das soll nur funktionieren, wenn man sich als Nutzer mit Benutzernamen und Passwort registriert hat, wobei es eine Benutzernamenkontrolle geben soll, die unpassende Namen verbietet. Auch soll der Admin jederzeit unpassende Benutzer über den Admin-Zugang löschen oder ändern können.

Es soll bis zu 10 parallel laufende Spiele geben können, wobei dieser Wert vom Admin editiert werden kann. Ebenso sollen die Spiele zeitversetzt gespielt werden können. Drei Fragen pro Runde sind vorgesehen, wobei ein Zeitlimit von 10-30 Sekunden eingeplant ist (die tatsächliche Antwortzeit sollte man in einem Praxistest genauer ermitteln).

Es wurde besonderen Wert darauf gelegt, dass in der GUI-Gestaltung das Corporate Design der Universität Bremen mit einbezogen wird, sodass z.B. durch Einfügen des Uni-Logos ein Wiedererkennungswert und ein direkter Bezug zur Uni bei den Nutzern hergestellt wird. Auch ein Verweis auf die Universität-Webseite sowie ggf. eine Social-Media-Anbindung sind erwünscht, wenn auch eher optionaler Art.

Die Spieler sollen geduzt werden und möglichst gender-neutral, sofern sinnvoll und möglich, angesprochen werden.

Auch die Einbindung von wohlklingenden und passenden Sounds soll den Spaß am Spiel wecken.

Des Weiteren sollen im Spiel selbst zu jeder Frage vier Antwortmöglichkeiten gegeben werden und Joker in Form eines 50:50-Jokers oder einer Zeitverlängerung eingebaut werden, die man sich jedoch erst erspielen muss und nicht von Beginn an zur Verfügung stehen.

Der Schwierigkeitsgrad steigt nicht mit jeder Frage, sondern die Fragen werden nach Kategorie, die die Spieler zu Beginn bestimmen können, zufällig ausgewählt.

Wenn ein Spieler aufgibt bzw. das laufende Spiel abbricht, soll er in Form eines Punktabzugs bestraft werden.

Es soll allerdings keine Freundes- oder Rivalenlisten geben.

### 2.1.3 Analyse eines ähnlichem Systems (Daniel Pupat)

Als System haben wir uns die App *Quizduell* ausgesucht. Die App ist im App-Store kostenlos zu erhalten. Wir benutzen hierzu nicht die Premium-Version, da die kostenlose unserem System am nächsten kommt. Wir haben nicht die Rechte eines Administrators, weshalb sich die Analyse nur auf die App bezieht. Die App kann nur verwendet werden, wenn man eine bestehende Internetverbindung hat.



Abbildung 1: Ladebildschirm

Dies ist der Ladebildschirm, sobald die App gestartet wird.





Abbildung 2: Startbildschirm

Hier ist der Startbildschirm, nachdem man eingeloggt ist. Oben hat man die Funktionen Aktualisieren, Statistiken und Einstellungen. Beim Aktualisieren wird die Seite aktualisiert.



Abbildung 3: Statistiken

Hier kann man die Rating Liste aller Nutzer in Deutschland im Bezug auf der Punktzahl und der Anzahl der geschrieben und angenommenen Fragen sehen. Für die eigene Statistik wird eine Premium-Version benötigt.



Abbildung 4: Einstellungen

Unter Einstellungen kann man seine Kontoeinstellungen bearbeiten, wo man den Benutzernamen und Passwort ändern kann. Für Avatar gestalten und Farbschema ändern benötigt man die Premium-Version, welche man unter PREMIUM kaufen erwerben kann. Bei Fragen schreiben, kann man eigene Fragen schreiben, die evtl. dann übernommen werden. Bei Hilfe werden bestimmte Fragen im Hinblick auf die Benutzung beantwortet.



Abbildung 5: Startbildschirm

Weitere Funktionen beim Startbildschirm sind Neues Spiel starten oder ein bereits angefangenes Spiel weiterspielen.



Abbildung 6: Bildschirm wenn man ein neues Spiel starten möchte

Wenn man ein neues Spiel starten möchte, kann man einen neuen Spieler suchen, unter Facebook Freunden suchen, falls man unter Facebook eingeloggt ist, oder gegen einen beliebigen Spieler spielen, der ebenfalls einen gesucht hat.



Abbildung 7: Bildschirm beim neuen oder laufenden Spiel

Hier wird der bisherige Spielstand angezeigt. Insgesamt gibt es 6 Fragen, welche jeder beantworten muss. Die Spieler erhalten jeweils die gleichen Fragen. Es werden immer 2 Fragen abwechselnd gestellt, wobei der Spieler einmal eine Kategorie auswählen darf und einmal die Fragen der Kategorie des Gegners beantworten muss. Die Spieler haben 48 Stunden Zeit, die Fragen zu beantworten, bis das Spiel automatisch mit einer Niederlage des Spielers, der nicht geantwortet hat, endet.

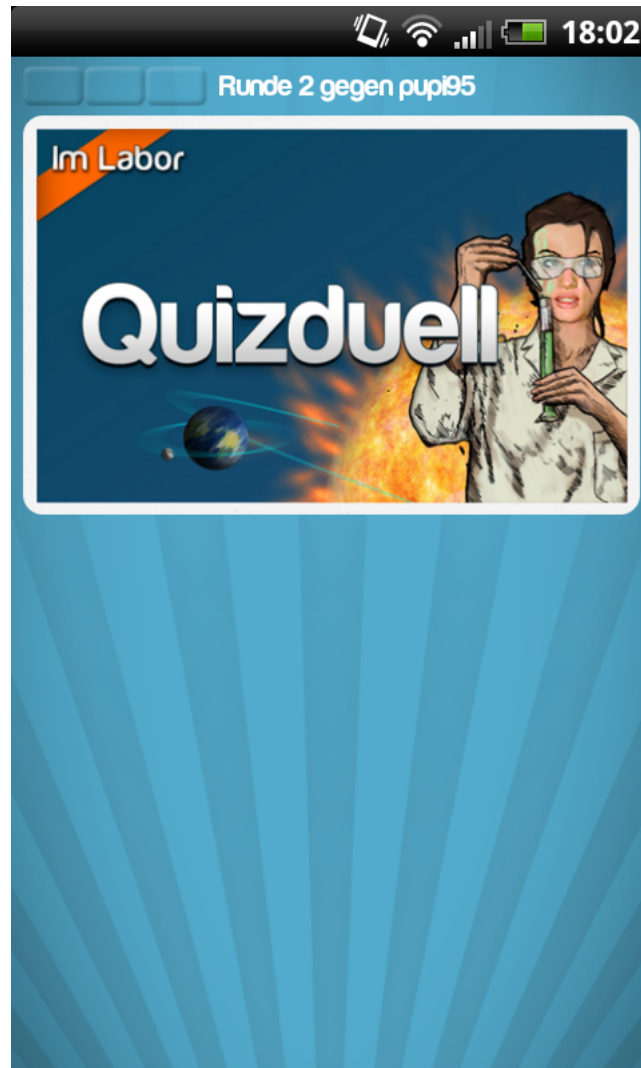


Abbildung 8: Beim Starten der Fragen

Beim Starten einer Frage erscheint erst nur eine Karte mit der Kategorie, bei einer Berührung wird die Karte umgedreht.

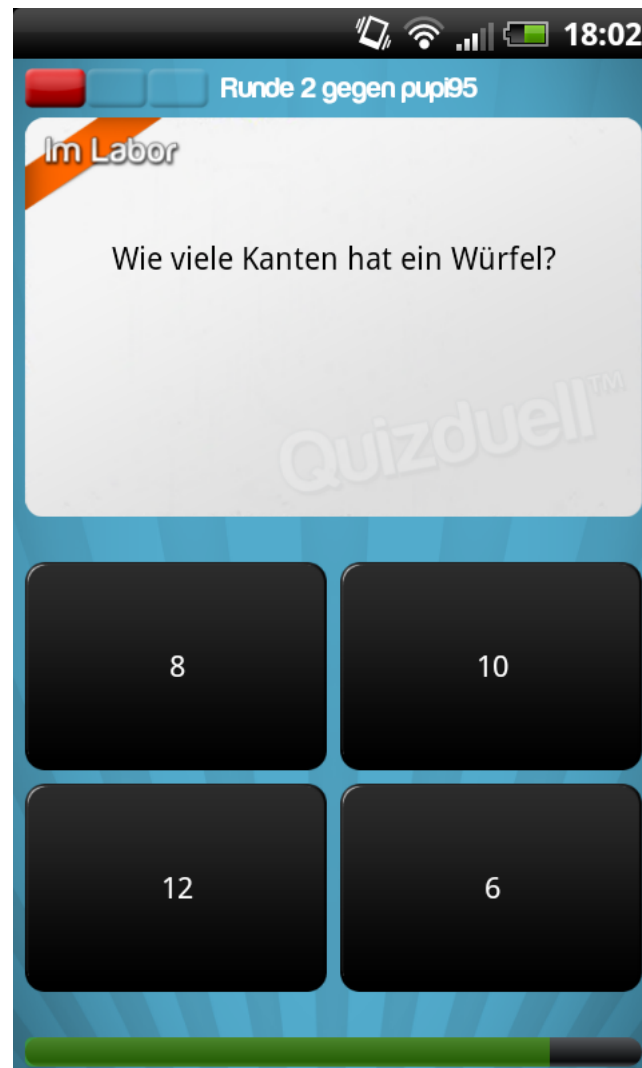


Abbildung 9: Anzeigen der Frage

Nun wird die Frage angezeigt, man hat 20 Sekunden Zeit, um die Frage zu beantworten, sonst wird diese als falsch gewertet.



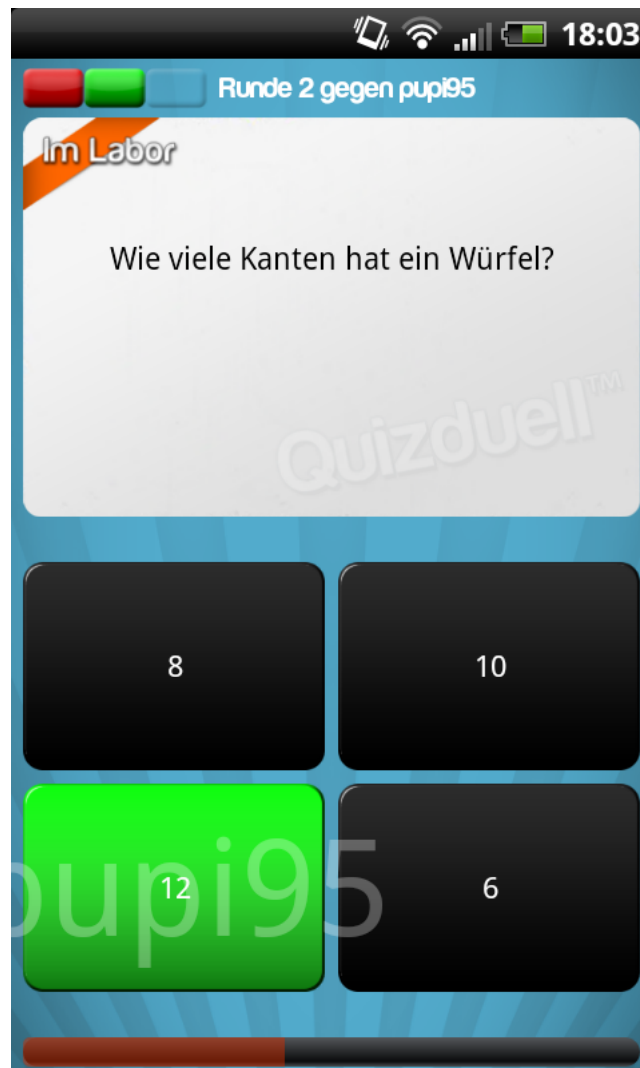


Abbildung 10: Reaktion bei richtiger Antwort

Bei einer richtigen Antwort wird diese mit Grün unterlegt und zeitgleich wird der Name des Gegners auf seiner Antwort angezeigt, wenn er diese bereits beantwortet hat. Bei einer falschen wird die Antwort mit rot unterlegt.

**Fazit:**

Die *Quizduell* App ist ein gutes System, mit welcher man unseres später vergleichen kann. Es hat viele Funktion, wie gezeigt, welche unser System ebenfalls haben muss. Es gibt allerdings auch Unterschiede. In unserer App wird es nur 3 Fragen pro Runde geben und es werden keine eigenen Fragen geschrieben, außerdem wird es keine Premium-Version geben. Zudem werden wir auch Offline-Funktion einbauen, in welcher der Spieler alleine spielen kann.

## 2.2 Produktperspektive (Tim W. und Tim E.)

### 2.2.1 Systemschnittstellen

Als Grundlage dient ein neu aufzusetzendes oder bestehendes System, auf dem die Software installiert werden kann. Als externe Schnittstelle muss dieses mit dem Internet verbunden sein. Der Server der Quizapp kann an einem PC über HTTP mit einem HTML5-unterstützenden Browser angesprochen werden. Mobiler Zugriff erfolgt auf Android-Geräten zudem mit einer eigenen App.

### 2.2.2 Benutzerschnittstelle

Die GUI der Website weist je nachdem, ob sich ein normaler Nutzer oder ein Admin anmeldet, unterschiedliche Funktionalitäten auf.

### 2.2.3 Hardwareschnittstellen

Die HTML-Website ist plattform-unabhängig nutzbar. Smartphones mit kompatibler Android-Version (welche?) können die Quizapp nutzen.

### 2.2.4 Softwareschnittstellen

Name	Version	Hersteller	Quelle
Java Runtime	6 Update 37	Oracle	<a href="http://java.com">http://java.com</a>
JUnit			
libGDX			
Android			

### 2.2.5 Kommunikationsschnittstellen

Das System muss mit einer öffentlichen IP-Adresse und Domain von außen ansprechbar sein. Die Bandbreite sollte groß genug sein, um mehreren Nutzern parallel den Zugang zur Software zu gewähren, wobei jeder Nutzer einzeln nur eine sehr geringe Bandbreite benötigt.

### 2.2.6 Speicherbeschränkung

Die App wird voraussichtlich bis zu 30 MB Speicherplatz beziehen. (Arbeitsspeicher?)

### 2.2.7 Operationen (Betriebsmodi)

Wir unterscheiden zwischen dem normalen Betriebsmodus und einem Wartungsmodus, in dem z.B. Backups aufgespielt werden können. Während des Wartungsmodus ist kein öffentlicher Zugriff vorgesehen. Im normalen Betriebsmodus wird unterschieden zwischen Admins, registrierten Benutzern und Gästen. Jeder hat das Recht zu spielen, wobei registrierte Nutzer Vorteile wie Statistiken etc. genießen. Der Admin kann das System über die Website verwalten, z.B. Nutzernamen ändern. In der App ist kein Admin-Modus vorhanden.

### 2.2.8 Möglichkeiten der lokalen Anpassung

Es handelt sich bei dem System um ein lauffähiges Gesamtpaket. Es muss keine Datenbank o.Ä. eingerichtet werden. Lediglich eine IP-Adresse muss eingerichtet werden. Somit ist keine lokale Anpassung nötig.

## 2.3 Anwendungsfälle (Daniel)

- **1. App starten**  
Die App wird gestartet.
- **2. Offline-Modus starten**  
Die App wird im Offline-Modus gestartet. Keine Anmeldung und Internet Verbindung notwendig.
- **3. Benutzer registrieren**  
Benutzer registriert sich mit Benutzernamen und Passwort
- **4. Benutzer anmelden**  
Der Benutzer meldet sich mit Benutzernamen und Passwort an. Die Daten werden gespeichert, sodass kein nochmaliges anmelden nötig ist.
- **5. Benutzer abmelden**  
Benutzer wird abgemeldet und muss sich beim nächsten Start wieder anmelden.
- **6. Spielen(Offline)**  
Der Akteur spielt Offline.
- **7. Spielen(Online)**  
Es erscheint eine Auswahl zwischen Neues Spiel starten und es werden alle offenen Spiele angezeigt
- **8. Neues Spiel**  
Es erscheint eine Liste aller Spieler die Online sind.
- **9. Gegner herausfordern**  
Aus der Liste kann ein Spieler angeklickt und herausgefordert werden.

- **10. Offenes Spiel spielen**  
Eine neue Spielrunde wird gestartet, die erste Frage erscheint.
- **11. Einstellungen ändern**  
Unter Einstellungen kann der Benutzername und das Passwort geändert werden
- **12. Website wird aufgerufen**  
Website wird angezeigt.
- **13. Admin anmelden**  
Benutzer meldet sich als Admin an. Die Website ist nur für Administratoren verwendbar.
- **14. Admin abmelden**  
Benutzer meldet sich als Admin ab.
- **15. Frage hinzufügen**  
Eine neue Frage wird hinzugefügt
- **16. Frage bearbeiten**  
Eine vorhandene Frage wird bearbeitet
- **17. Frage löschen**  
Eine vorhandene Frage wird gelöscht
- **18. Frageliste importieren**  
Eine Liste an Fragen wird importiert
- **19. Frageliste exportieren**  
Eine Liste an Fragen wird importiert
- **20. User löschen**  
Bestimmter User wird gelöscht
- **21. Passwort ändern**  
User will sein Passwort ändern

## 2.4 Charakteristika der Benutzer (Tim W.)

Name:	Armin Admin
Benutzertyp:	Administrator
Tätigkeit	Redakteur
Wohnort	Uniallee, Bremen
Alter	42
Motivation	System aufrecht erhalten
Aufgaben	Verwaltung und Systemwartung

<b>Name:</b>	<b>Paul Neuling</b>
Benutzertyp:	Gelegenheitsspieler
Tätigkeit	Student
Wohnort	Uniallee, Bremen
Alter	19
Motivation	Etwas über die Uni lernen und Spaß haben

<b>Name:</b>	<b>Klaus Zocker</b>
Benutzertyp:	Häufiger Spieler
Tätigkeit	Student
Wohnort	Uniallee, Bremen
Alter	23
Motivation	Will die Rangliste anführen

<b>Name:</b>	<b>Sabine Ohnenetz</b>
Benutzertyp:	Gelegenheitsspielerin
Tätigkeit	Student
Wohnort	Uniallee, Bremen
Alter	19
Motivation	Offline etwas spielen

### Armin Admin

Text...

### Paul Neuling

Text...

### Klaus Zocker

Text...

### Sabine Ohnenetz

Text...

## 2.5 Einschränkungen (Tim E.)

Im Folgenden listen wir die Mindestanforderungen des Produkts auf, die nur mithilfe eines Logins bei StudIP einzusehen sind<sup>1</sup>.

### Server:

- SpielerInnen können sich beim Server registrieren
- verwaltet eine Liste aller SpielerInnen
- verwaltet die Fragekategorien, die Fragen und die Antworten sowie Belohnungen

<sup>1</sup>[https://elearning.uni-bremen.de/scm.php?cid=2b323f34b16a84e8dce31dcdfc0be6ad&show\\_scm=4c88951a202b2543c96de2c8a476d471](https://elearning.uni-bremen.de/scm.php?cid=2b323f34b16a84e8dce31dcdfc0be6ad&show_scm=4c88951a202b2543c96de2c8a476d471)

- SpielerInnen können sich beim Server als gerade spielbereit setzen und wieder entfernen
- auf Anfrage wird aus der Liste der gerade spielbereiten SpielerInnen zufällig eine konkrete GegnerIn ermittelt
- verwaltet laufende Spiele und archiviert diese
  - TeilnehmerInnen
  - aktueller Punktestand
  - gestellte Fragen
  - gegebene Antworten
- verwaltet die Konfiguration der Spiele
  - Anzahl Fragerunden
  - Anzahl Fragen pro Runde
  - Zeitspanne zur Auswahl einer Antwort
  - Punkte pro richtiger Antwort
  - Punkte für gewonnenes und unentschiedenes Spiel
- realisiert ein Belohnungssystem für SpielerInnen
  - Erwerb eines Titels bei bestimmten Punktzahlen
  - Joker (z.B. Zeitverlängerung oder Anzeige der richtigen Antwort auf eine Frage)
- verwaltet eine Rangliste aller SpielerInnen
- ein Spiel wird komplett vom Server geleitet, d.h. die Fragen und möglichen Antworten werden an die jeweiligen Clients gesendet und die korrekte Antwort wird vom Server geprüft (dies beinhaltet auch evtl. Spielregeln wie z.B. der Einsatz von Jokern oder das Einhalten der Zeitvorgaben)
- Administrationszugang über eine Webseite
  - redaktionelle Bearbeitung der Fragen und Antworten
    - \* Kategorien hinzufügen, Zuordnung der Fragen zu Kategorien ändern
    - \* Fragen und Antworten bearbeiten
    - \* Fragen und Antworten hinzufügen
    - \* Fragen und Antworten löschen
  - Import/Export der Fragen und Antworten in/aus geeignetem Format
  - Konfiguration der Spiele
    - \* Anzahl Fragerunden
    - \* Anzahl Fragen pro Runde

- \* Zeitspanne zur Auswahl einer Antwort
- \* Punkte pro richtiger Antwort
- \* Punkte für gewonnenes und unentschiedenes Spiel

**Client:**

- lauffähig auf Android-Systemen
- unterstützt Registrierung und Anmeldung beim Server
- neues Spiel starten
- zufällige GegnerIn (siehe Server)
- Auswahl einer GegnerIn aus Liste der spielbereiten SpielerInnen
- Eingabe einer bekannten GegnerIn
- eine Spielanfrage annehmen, d.h. an einem Spiel teilnehmen
- Anzeige einer Frage mit möglichen Antworten
- Auswahl einer Antwort in vorgegebener Zeitspanne
- Anzeige einer Rangliste

### 2.5.1 Technische Rahmenbedingungen

- Softwareergonomische Prinzipien werden umgesetzt
- Der Server soll unter Linux, Windows und MacOS laufen (als Referenz gelten die Rechner in den Praktikumsräumen im MZH)
- Es muss eine relationale Datenbank für die serverseitige Persistenz benutzt werden
- Persistenz-Frameworks sind erlaubt (z.B. JPA)
- Verwendung leichtgewichtiger DBMS (z.B. Derby, SQLite) ohne echte Serverinstallation ist vorgeschrieben
- Verwendung und Abgabe eines Build-/Installationsskriptes, damit die Anwendung einfach installiert und aus den Quellen gebaut werden kann. Alle notwendigen Installations- und Konfigurationsschritte sind dokumentiert
- Eventuell benutzte Fremdbibliotheken dürfen für den Einsatz in Forschung und Lehre keine Beschränkungen (Geld, Benutzung, ...) aufweisen
- Quelltext in Deutsch oder Englisch dokumentiert. Gleiches gilt für Variablen- und Klassennamen. Alle anderen Dokumente in Deutsch Die Implementierungssprache ist Java 6 oder höher (weitere zulässige Sprachen in geringem Umfang sind: HTML, XML und JavaScript)

### 2.5.2 Gesetzliche Rahmenbedingungen

Hier gilt in erster Linie das deutsche Recht. Um genau zu sein, kommen hier das Datenschutzgesetz<sup>2</sup> und das Urheberrecht<sup>3</sup> zum Tragen. Verfahrensrechtliche Vorkehrungen, um die Datensicherheit<sup>4</sup> zu gewährleisten, sind von dem Kunden zu treffen. Der Kunde wird von der Software hinsichtlich der technischen Vorkehrungen insofern unterstützt, dass der Zugriff auf die Daten und der Zugang auf das Administrationstool passwortgeschützt ist. Hinsichtlich des Urheberrechts ist besonders auf die Regelung für Computerprogramme<sup>5</sup> zu achten.

### 2.5.3 Sicherheitskritische Aspekte

Um das deutsche Datenschutzgesetz einzuhalten, muss der Kunde weitere Maßnahmen treffen. Diese sind nicht entscheidend für die Entwicklung der Software und liegen in der Verantwortung des Kunden.

## 2.6 Annahmen und Abhängigkeiten (Tim E.)

Bis zur Auslieferung der Software wird sich der Kunde nicht ändern. Die Anforderungsspezifikation dient als eine Art Vertrag mit dem Kunden. Deshalb ist davon auszugehen, dass nach der Abgabe der Anforderungsspezifikation keine zusätzlichen Anforderungen hinzukommen. Abgabetermine haben Deadlines und sind somit strikt einzuhalten.

Des Weiteren wird davon ausgegangen, dass sich die Nutzer der Software zwar mit dem System eingehend auseinandersetzen. Es wird jedoch auch für den ungeübten Nutzer leicht möglich sein, dieses zu verwenden. Der jeweilige Nutzer sollte zumindest schon mal mit einem Computer bzw. Smartphone und dessen Bedienung vertraut sein.

## 2.7 Ausblick (Tim E.)

Große Änderungen sind nach Auslieferung des Systems zwar nicht zu erwarten, aber es können sich dennoch immer wieder mit der Zeit Anpassungen ergeben, die zu diesem Zeitpunkt allerdings noch nicht feststehen.

---

<sup>2</sup>[http://www.gesetze-im-internet.de/bdsg\\_1990/](http://www.gesetze-im-internet.de/bdsg_1990/)

<sup>3</sup><http://www.gesetze-im-internet.de/urhg/>

<sup>4</sup>[http://www.gesetze-im-internet.de/bdsg\\_1990/\\_\\_9.html](http://www.gesetze-im-internet.de/bdsg_1990/__9.html)

<sup>5</sup><http://www.gesetze-im-internet.de/urhg/BJNR012730965.htm#BJNR012730965BJNG004201377>



## 3 Detaillierte Beschreibung

### 3.1 Datenmodell (Tobias)

#### 3.1.1 Datenmodellbeschreibung

Im Folgenden werden die einzelnen Klassen ergänzend beschrieben:

**Nutzer:** Diese Einheit steht für den Appnutzung, die über verschiedene Aktionen mit dem Serversystem in Verbindung steht. Die Einheit Nutzer hat sowohl alle Fragen als Stringliste, um auch offline spielen zu können, als auch den Usernamen gespeichert. Der Nutzer kann sich über das System einloggen, ausloggen oder auch erst noch registrieren. Sobald sich der Nutzer angemeldet oder registriert hat, prüft das System automatisch, ob die App auf dem neusten Stand ist. Der Nutzer kann sowohl offline, als auch online über das System Spielen.

**Server:** Die Klasse stellt die bereitgestellten Server und Datenbanken dar, indem alle notwendigen Daten wie auch Quizfragen und Nutzerlisten gespeichert sind. Diese Einheit steht abgesehen vom Offline-Spiel mit allen Aktionen der App oder Webseite in Verbindung.

**Admin:** Der Admin verwaltet das System oder die Server über eine bereitgestellte Webseite in der sich einloggt oder wieder ausloggt. Er bearbeitet/löscht oder fügt Fragen hinzu, oder benennt gewisse Nutzer mit einem unangemessenen Nutzernamen um. Er kann einen Nutzer auch löschen.

**Bearbeiten/löschen/hinzufügen:** Diese Einheit stellt die Aktivitäten des Admin am Server dar. Der Admin kann eine Datei mit den neuen Fragen importieren und alles andere damit ersetzen oder einzelne löschen oder bearbeiten.

**Update:** Das Update prüft mithilfe des Boolean `isUpToDate`, ob der eingeloggte Nutzer auf dem neusten Stand ist. Dieser Boolean bekommt seinen Wert wiederum mithilfe eines Vergleiches des letzten Einloggdatus des Nutzers.

**Einloggen/Ausloggen/Registrieren:** Der Einlogg- und der Registriervorgang ist sehr ähnlich, weshalb diese zusammengefasst wurden. Über diesen Vorgang wird die Gültigkeit des eingegebenen Passworts über den Nutzernamen verifiziert. Sollte der Nutzer nach dem Verlassen der App nicht abgemeldet haben, so bleibt der Nutzer auch angemeldet. Daher wird in dieser Klasse nach dem Starten des OnlineModus der Boolean `alreadyLoggedIn` abgefragt.

**Spielen.** Der Nutzer kann sowohl online, als auch offline spielen. Diese Einheit hat die/den Spieler als Objekte und die Punkte der jeweiligen Spieler gespeichert. Es weiß auch, welcher Spieler am Zug ist. Am Ende eines Spiels wird der Gewinner in einem Boolean eingetragen

### 3.2 Anwendungsfälle (Tobias und Daniel)

1	<b>App starten</b>
Akteure	Paul Neuling, Klaus Zocker, Sabine Ohnenetz
Ziel	Der Akteur möchte die App starten
Vorbedingungen	App bereits heruntergeladen und installiert
Regulärer Ablauf	1. Der Akteur startet die App, indem er auf den Start-button drückt 2. Die App startet und zeigt die Startseite
Varianten	keine
Nachbedingungen	Die App ist gestartet und der Benutzer kann diese nun verwenden
Fehler-/Ausnahmefälle	keine

Abbildung 11: Startseite



<b>2</b>	<b>Offline-Modus starten</b>
<b>Akteure</b>	Paul Neuling, Sabine Ohnenetz
<b>Ziel</b>	Der Akteur möchte ohne Internet Verbindung und Anmeldung spielen
<b>Vorbedingungen</b>	App wurde bereits gestartet
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf Jetzt Spielen
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Der Akteur kann nun ohne Anmeldung spielen
<b>Fehler-/Ausnahmefälle</b>	keine

<b>3</b>	<b>Benutzer registrieren</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker
<b>Ziel</b>	Der Akteur möchte sich registrieren um im Internet gegen andere zu spielen
<b>Vorbedingungen</b>	App wurde bereits gestartet
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf registrieren 2 Der Akteur gibt seinen Benutzernamen und Passwort an 3 Der Akteur drückt auf registrieren
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Der Akteur ist nun registriert und kann sich anmelden
<b>Fehler-/Ausnahmefälle</b>	1. Es ist keine Internet-Verbindung vorhanden 2. Benutzername ist bereits vergeben 3. Doppelte Passworteingabe stimmt nicht überein

<b>4</b>	<b>Benutzer anmelden</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker
<b>Ziel</b>	Der Akteur möchte sich anmelden und dann online spielen
<b>Vorbedingungen</b>	App wurde bereits gestartet
<b>Regulärer Ablauf</b>	1. Der Akteur gibt seinen Benutzernamen und sein Passwort ein 2 Der Akteur drückt auf anmelden
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Der Akteur ist nun angemeldet und kann Online-Spielen
<b>Fehler-/Ausnahmefälle</b>	1. Es ist keine Internet-Verbindung vorhanden 2. Benutzername existiert nicht 3. Passwort ist falsch

<b>5</b>	<b>Benutzer abmelden</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker
<b>Ziel</b>	Der Akteur möchte sich abmelden
<b>Vorbedingungen</b>	Der Akteur war bereits angemeldet
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf abmelden
<b>Varianten</b>	1. Der Akteur ist bereits angemeldet, hat aber keine Internet-Verbindung und wird automatisch abgemeldet
<b>Nachbedingungen</b>	Der Akteur ist nun abgemeldet und ist wieder auf dem Startbildschirm
<b>Fehler-/Ausnahmefälle</b>	keine

<b>6</b>	<b>Spielen(Offline)</b>
<b>Akteure</b>	Sabine Ohnenetz
<b>Ziel</b>	Der Akteur möchte spielen
<b>Vorbedingungen</b>	Der Akteur hat den Anwendungsfall 2 ausgeführt
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf spielen
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Die Spielrunde wird gestartet
<b>Fehler-/Ausnahmefälle</b>	keine

<b>7</b>	<b>Spielen(Online)</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker
<b>Ziel</b>	Der Akteur möchte spielen
<b>Vorbedingungen</b>	Der Akteur ist bereits Online angemeldet
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf Spielen
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Es erscheint eine Auswahl zwischen Neues Spiel starten und es werden alle offenen Spiele angezeigt
<b>Fehler-/Ausnahmefälle</b>	keine

<b>8</b>	<b>Neues Spiel</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker
<b>Ziel</b>	Der Akteur möchte ein neues Spiel starten
<b>Vorbedingungen</b>	Der Akteur hat bereits Anwendungsfall 7 ausgeführt
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf neues Spiel
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Es wird eine Liste aller Spieler die Online sind angezeigt
<b>Fehler-/Ausnahmefälle</b>	Leere Liste, wenn keine Spieler online sind

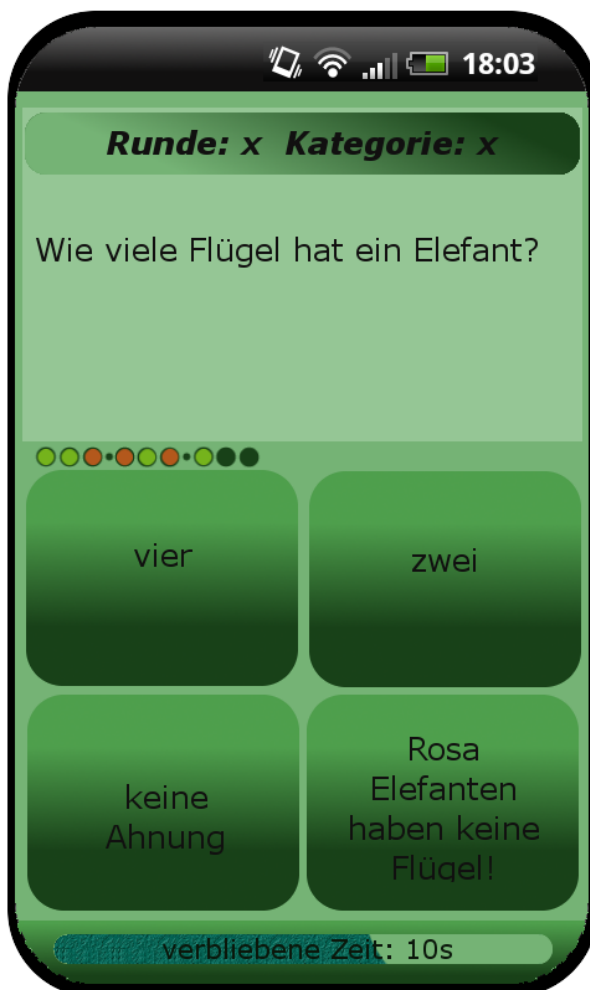
Abbildung 12: Liste aller Spieler die Online sind



<b>9</b>	<b>Gegner herausfordern</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker
<b>Ziel</b>	Der Akteur möchte einen anderen Spieler herausfordern
<b>Vorbedingungen</b>	Der Akteur hat bereits ein neues Spiel gestartet und eine Liste ist erschien(Anwendungsfall 8)
<b>Regulärer Ablauf</b>	1. Der Akteur drückt in der Liste auf einen Spieler und fordert ihn heraus
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Es wird eine Anfrage an den Spieler gesendet und der Akteur kann bereits die erste Fragerunde beginnen
<b>Fehler-/Ausnahmefälle</b>	Leere Liste, wenn keine Spieler online sind

<b>10</b>	<b>Offenes Spiel spielen</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker, Sabine Ohnenetz
<b>Ziel</b>	Der Akteur möchte anfangen die Fragen zu beantworten
<b>Vorbedingungen</b>	Der Akteur hat noch ein Spiel offen
<b>Regulärer Ablauf</b>	<ol style="list-style-type: none"> <li>1. Die erste Frage erscheint</li> <li>2. Der Spieler hat 20sek Zeit die Frage zu beantworten</li> <li>3. Nachdem der Spieler 3 Fragen weiter ist als der Gegner muss er warten</li> </ol>
<b>Varianten</b>	Im Offline-Modus kann der Spieler so viele Fragen beantworten wie er will
<b>Nachbedingungen</b>	Der Akteur hat alle Fragen beantwortet und wartet nun auf den Gegner
<b>Fehler-/Ausnahmefälle</b>	keine

Abbildung 13: Frage



<b>11</b>	<b>Einstellungen ändern</b>
<b>Akteure</b>	Paul Neuling, Klaus Zocker
<b>Ziel</b>	Der Akteur möchte seine Einstellungen ändern
<b>Vorbedingungen</b>	Der Akteur ist bereits angemeldet
<b>Regulärer Ablauf</b>	1. Der Akteur gibt seinen neuen Nutzernamen an 2. Der Akteur gibt sein neues Passwort an
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Das Passwort und der Nutzername wurden geändert
<b>Fehler-/Ausnahmefälle</b>	Nutzername ist bereits vergeben

<b>12</b>	<b>Website wird aufgerufen</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur will die Website aufrufen
<b>Vorbedingungen</b>	keine
<b>Regulärer Ablauf</b>	1. Der Akteur ruft die Website auf, indem er die URL angibt
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Die Website wird angezeigt
<b>Fehler-/Ausnahmefälle</b>	Server nicht erreichbar

<b>13</b>	<b>Admin anmelden</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte sich anmelden
<b>Vorbedingungen</b>	Website ist aufgerufen
<b>Regulärer Ablauf</b>	1. Der Akteur gibt seinen Nutzernamen und Passwort an 2. Der Akteur drückt auf anmelden
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Der Akteur ist angemeldet
<b>Fehler-/Ausnahmefälle</b>	Falscher Nutzername oder Passwort

<b>14</b>	<b>Admin abmelden</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte sich abmelden
<b>Vorbedingungen</b>	Der Akteur möchte sich abmelden
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf abmelden
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Der Akteur ist abgemeldet
<b>Fehler-/Ausnahmefälle</b>	keine

<b>15</b>	<b>Frage hinzufügen</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte eine Frage hinzufügen
<b>Vorbedingungen</b>	Der Akteur ist angemeldet
<b>Regulärer Ablauf</b>	1. Der Akteur gibt die Frage und die Antworten an 2. Der Akteur drückt auf speichern
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Die Frage wurde gespeichert
<b>Fehler-/Ausnahmefälle</b>	Frage, bzw. Antworten unvollständig ausgefüllt

<b>16</b>	<b>Frage bearbeiten</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte eine Frage bearbeiten
<b>Vorbedingungen</b>	Es existieren bereits Fragen
<b>Regulärer Ablauf</b>	1. Der Akteur gibt die veränderte Frage und die Antworten an 2. Der Akteur drückt auf speichern
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Die alte Frage wurde gelöscht und die neue gespeichert
<b>Fehler-/Ausnahmefälle</b>	Frage, bzw. Antworten unvollständig ausgefüllt

<b>17</b>	<b>Frage löschen</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte eine Frage löschen
<b>Vorbedingungen</b>	Es existieren bereits Fragen
<b>Regulärer Ablauf</b>	1. Der Akteur wählt die Frage aus 2. Der Akteur drückt auf löschen
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Die Frage wurde gelöscht
<b>Fehler-/Ausnahmefälle</b>	keine

<b>18</b>	<b>Frageliste importieren</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte eine Liste von Fragen importieren
<b>Vorbedingungen</b>	Es existiert bereits eine Liste von Fragen
<b>Regulärer Ablauf</b>	1. Der Akteur wählt die Datei aus 2. Der Akteur drückt auf importieren
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Die Fragen wurden importiert
<b>Fehler-/Ausnahmefälle</b>	Falsches Dateiformat



<b>19</b>	<b>Frageliste exportieren</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte eine Liste von Fragen exportieren
<b>Vorbedingungen</b>	Es existieren bereits Fragen
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf Datei exportieren 2. Der Akteur wählt den Speicherort aus
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Die Fragen wurden exportiert
<b>Fehler-/Ausnahmefälle</b>	keine

<b>20</b>	<b>User löschen</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte einen User löschen
<b>Vorbedingungen</b>	Der User existiert
<b>Regulärer Ablauf</b>	1. Der Akteur drückt auf den entsprechenden User 2. Der Akteur drückt auf löschen
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Der User wurde gelöscht
<b>Fehler-/Ausnahmefälle</b>	keine

<b>21</b>	<b>Passwort ändern</b>
<b>Akteure</b>	Armin Admin
<b>Ziel</b>	Der Akteur möchte sein Passwort ändern
<b>Vorbedingungen</b>	Der Akteur hat Einstellungen aufgerufen
<b>Regulärer Ablauf</b>	1. Der Akteur gibt sein neues und altes Passwort an 2. Der Akteur drückt auf Passwort ändern
<b>Varianten</b>	keine
<b>Nachbedingungen</b>	Das Passwort wurde geändert
<b>Fehler-/Ausnahmefälle</b>	altes Passwort wurde falsch eingegeben

### 3.3 Aktionen (Tim W.)

#### 3.3.1 App.Abmelden()

- Der Nutzer meldet sich per Button ab.
- **Vorbedingung:** Der Nutzer befindet sich im Online-Modus.
- **Nachbedingung:** Der Nutzer kann zwischen Online- und Offline-Modus wählen.  
Beim nächsten Start wird der Nutzer nicht automatisch eingeloggt.
- **max. Reaktionszeit:** 2 sek.

### 3.3.2 App.Anmelden(nutzername, password)

- Der Nutzer gibt seine Daten, mit denen er sich registriert hat, in zwei Eingabefelder ein und bestätigt die Eingabe mit einem Buttondruck.
- **Vorbedingung:** Der Nutzer ist registriert und möchte den Onlinemodus starten.
- **Nachbedingung:** Der Nutzer ist eingeloggt und die Daten sind für den nächsten Start gespeichert.
- **Fehlerbedingung:** Falls Nutzername oder Passwort nicht korrekt sind, wird der Nutzer aufgefordert, die Daten erneut einzugeben.
- **max. Reaktionszeit:** 5 sek.

### 3.3.3 App.Einstellungen ändern(neuerNutzername, neuesPasswort)

- Der Nutzer kann im Einstellungsmenü einen neuen Nutzernamen und/oder ein neues Passwort bestimmen. Das neue Passwort muss durch eine zweite Eingabe in ein weiteres Eingabefeld bestätigt werden. Die Eingaben werden mit einem Buttondruck gespeichert.
- **Vorbedingung:** Der Nutzer befindet sich im Einstellungsmenü.
- **Nachbedingung:** Der Nutzer kehrt ins Menü zurück. Die Änderungen sind im Clienten und auf dem Server gespeichert.
- **max. Reaktionszeit:** 3 sek.

### 3.3.4 App.Einstellungen aufrufen()

- Der Nutzer kann im Online-Modus per Button die Einstellungen aufrufen, um Nutzernamen oder Passwort zu ändern.
- **Vorbedingung:** Der Nutzer befindet sich im Online-Modus.
- **Nachbedingung:** Der Nutzer befindet sich im Einstellungsmenü.
- **max. Reaktionszeit:** 1 sek.

### 3.3.5 App.Endergebnis bestätigen()

- Nach einer Partie werden den Nutzern das Endresultat angezeigt. Um zum Menü zurückzukehren, drücken sie einen Button zur Bestätigung.
- **Vorbedingung:** Eine Runde wurde beendet.
- **Nachbedingung:** Der Nutzer kehrt ins Menü zurück.
- **max. Reaktionszeit:** 1 sek.

### 3.3.6 App.Frage beantworten(antwort)

- Der Nutzer beantwortet eine offene Frage, in dem er auf eine der vier Antwortmöglichkeiten drückt.
- **Vorbedingung:** Eine Runde wurde gestartet.
- **Nachbedingung:** Das System wertet die Antwort aus. Den Nutzern wird mitgeteilt, ob die Antwort stimmt. Die nächste Runde wird eingeleitet.
- **max. Reaktionszeit:** 1 sek.

### 3.3.7 App.Gegner herausfordern(nutzer)

- Der Nutzer fordert einen Nutzer heraus, indem er auf einen der Namen in der Liste drückt.
- **Vorbedingung:** Der Nutzer startet ein Online-Match.
- **Nachbedingung:** Die Herausforderung wurde versandt.
- **Fehlerbedingung:** Falls die Übertragung fehlschlägt, wird dem Nutzer dies mitgeteilt.
- **max. Reaktionszeit:** 3 sek.

### 3.3.8 App.Herausforderung annehmen(herausforderung)

- Der Nutzer lässt sich über einen Button offene Herausforderungen anzeigen und kann diese mit einem weiteren Druck annehmen.
- **Vorbedingung:** Der Nutzer ist im Online-Modus.
- **Nachbedingung:** Die Herausforderung wurde angenommen und das Spiel wird gestartet.
- **Fehlerbedingung:** Falls die Übertragung fehlschlägt, wird dem Nutzer dies mitgeteilt.
- **max. Reaktionszeit:** 5 sek.

### 3.3.9 App.Joker nutzen(art)

- Der Nutzer nutzt einen Button, um während einer Fragerunde einen Joker zu nutzen.
- **Vorbedingung:** Eine Runde wurde gestartet. Der Nutzer hat sich den Joker in einem früheren Spiel Joker erspielt.
- **Nachbedingung:** Je nach Art des Jokers wird die Zeit verlängert oder zwei Antworten werden entnommen.

- **Fehlerbedingung:** Falls der Spieler bereits einen Joker gespielt hat, wird ihm mitgeteilt, dass er nicht noch einen spielen kann.
- **max. Reaktionszeit:** 1 sek.

#### 3.3.10 App.Neues Spiel starten(modus)

- Der Nutzer startet eine neue Partie per Button.
- **Vorbedingung:** Der Nutzer befindet sich im Offline- oder Online-Modus.
- **Nachbedingung:** Falls der Nutzer sich im Offline-Modus befindet, wird eine Einzelspieler Partie gestartet. Ist er online, wird ihm eine Liste von anderen Spielern angezeigt.
- **Fehlerbedingung:** Falls keine weiteren Spieler online sind, wird dem Nutzer dies mitgeteilt und er kann es später noch einmal versuchen.
- **max. Reaktionszeit:** 3 sek.

#### 3.3.11 App.Offline-Modus starten()

- Der Nutzer wählt den Offline-Modus über einen Button aus.
- **Vorbedingung:** Die App wurde gestartet.
- **Nachbedingung:** Der Nutzer ist im Offline-Modus.
- **max. Reaktionszeit:** 2 sek.

#### 3.3.12 App.Online-Modus starten()

- Der Nutzer wählt den Online-Modus über einen Button aus.
- **Vorbedingung:** Die App wurde gestartet.
- **Nachbedingung:** Der Nutzer wird aufgefordert sich einzuloggen.
- **Fehlerbedingung:** Falls der Server nicht angesprochen werden kann, wird dem Nutzer dies durch eine Fehlermeldung mitgeteilt.
- **max. Reaktionszeit:** 1 sek.

#### 3.3.13 App.Rangliste aufrufen()

- Der Nutzer kann im Online-Modus per Button die Rangliste aufrufen.
- **Vorbedingung:** Der Nutzer befindet sich im Online-Modus.
- **Nachbedingung:** Der Nutzer befindet sich im Ranglistenmenü.
- **max. Reaktionszeit:** 3 sek.

#### 3.3.14 App.Registrieren(nutzername, password)

- Der Nutzer gibt seine Daten, mit denen er sich registrieren möchte, in zwei Eingabefelder ein und bestätigt die Eingabe per Button.
- **Vorbedingung:** Die App ist gestartet.
- **Nachbedingung:** Der Nutzer ist registriert.
- **Fehlerbedingung:** Falls Nutzername oder Passwort nicht den Anforderungen entsprechen, wird der Nutzer an den entsprechenden Feldern darauf hingewiesen.
- **max. Reaktionszeit:** 5 sek.

#### 3.3.15 Website.Anmelden(nutzername, password)

- Der Admin gibt seine Daten, mit denen er sich einloggen möchte, in zwei Eingabefelder ein und bestätigt die Eingabe per Button.
- **Vorbedingung:** Der Nutzer ist als Admin registriert.
- **Nachbedingung:** Der Admin ist eingeloggt.
- **Fehlerbedingung:** Falls Nutzername oder Passwort falsch sind, kann der Nutzer es nochmal probieren.
- **max. Reaktionszeit:** 3 sek.

#### 3.3.16 Website.Abmelden()

- Der Admin kann sich über das Menü abmelden.
- **Vorbedingung:** Der Admin ist eingeloggt.
- **Nachbedingung:** Der Admin ist ausgeloggt.
- **max. Reaktionszeit:** 2 sek.

#### 3.3.17 Website.Frage bearbeiten(id, frage, antworten, kategorie)

- Die Zellen der Fragenliste sind bearbeitbar. Der Admin kann an den nötigen Stellen einfach Änderungen vornehmen und mit einem Button bestätigen.
- **Vorbedingung:** Der Admin hat die Fragenliste aufgerufen.
- **Nachbedingung:** Die Änderungen wurden in der Datenbank eingetragen. Die Änderungen werden nun in der Liste angezeigt.
- **max. Reaktionszeit:** 5 sek.

**3.3.18 Website.Frage hinzufügen(frage, antworten, kategorie)**

- Der Admin kann über einen Button ein Formular zum Hinzufügen einer neuen Frage aufrufen. Frage, Antworten und Kategorie werden in Eingabefelder eingetragen und per Button abgespeichert.
- **Vorbedingung:** Der Admin hat die Fragenliste aufgerufen.
- **Nachbedingung:** Die Frage wurde in der Datenbank eingetragen. Die Frage wird nun in der Liste angezeigt.
- **Fehlerbedingung:** Falls etwas schief läuft, kann der Admin es nochmal probieren, die Daten im Formular gehen nicht verloren.
- **max. Reaktionszeit:** 5 sek.

**3.3.19 Website.Frage löschen(id)**

- Die Zeilen der Fragenliste können markiert werden. Mehrere markierte Fragen können mit einem Buttondruck gelöscht werden, wobei dies mit einem weiteren Klick bestätigt oder abgebrochen werden kann.
- **Vorbedingung:** Der Admin hat die Fragenliste aufgerufen.
- **Nachbedingung:** Die markierten Fragen wurden gelöscht. Die Änderungen werden nun in der Liste angezeigt.
- **Fehlerbedingung:** Falls etwas schief läuft, kann der Admin es nochmal probieren, die markierten Fragen bleiben weiterhin markiert.
- **max. Reaktionszeit:** 5 sek.

**3.3.20 Website.Fragenliste aufrufen()**

- Der Admin kann über das Menü die Liste aller Fragen einsehen.
- **Vorbedingung:** Der Admin ist eingeloggt.
- **Nachbedingung:** Dem Admin wird die Fragenliste angezeigt.
- **max. Reaktionszeit:** 3 sek.

**3.3.21 Website.Nutzerliste aufrufen()**

- Der Admin kann über das Menü die Liste registrierter Nutzer einsehen.
- **Vorbedingung:** Der Admin ist eingeloggt.
- **Nachbedingung:** Dem Admin wird die Nutzerliste angezeigt.
- **max. Reaktionszeit:** 3 sek.

### 3.3.22 Website.Nutzer löschen(id)

- Die Zeilen der Nutzerliste können markiert werden. Mehrere markierte Nutzer können mit einem Buttondruck gelöscht werden, wobei dies mit einem weiteren Klick bestätigt oder abgebrochen werden kann.
- **Vorbedingung:** Der Admin hat die Nutzerliste aufgerufen.
- **Nachbedingung:** Die markierten Nutzer wurden gelöscht. Die Änderungen werden nun in der Liste angezeigt.
- **Fehlerbedingung:** Falls etwas schief läuft, kann der Admin es nochmal probieren, die markierten Nutzer bleiben weiterhin markiert.
- **max. Reaktionszeit:** 5 sek.

### 3.3.23 Website.Nutzernamen ändern()

- Der Admin kann jeden Nutzernamen in der Liste bearbeiten und die Änderungen mit einem Button abspeichern.
- **Vorbedingung:** Der Admin hat die Nutzerliste aufgerufen.
- **Nachbedingung:** Die Nutzernamen werden in der Datenbank geändert. Die Nutzerliste ist weiterhin aufgerufen.
- **Fehlerbedingung:** Falls etwas schief läuft, kann der Admin es nochmal probieren, die aktuellen Änderungen bleiben in den Eingabefeldern erhalten.
- **max. Reaktionszeit:** 5 sek.

## 3.4 Softwaresystemattribute Tim E.)

Hier werden die sogenannten „nichtfunktionalen Anforderungen“ spezifiziert. Dazu gehören beispielsweise:

- **Performanz:** Beim Entwickeln unserer Software wird Anspruch auf eine fließende Arbeitsweise gelegt, was bedeutet, dass Interaktionen mit dem System möglichst schnell zu bewältigen sind. Deshalb ist darauf zu achten, Berechnungsprozesse so wenig komplex wie möglich zu halten. Die Implementierung eines Ladezeichens bei aufkommenden Verbindungsschwierigkeiten ist selbstverständlich vorgesehen.
- **Zuverlässigkeit (Korrektheit, Robustheit, Ausfallsicherheit):** Die zu entwickelnde Software, insbesondere die Androidapplikation soll auf vielen verschiedenen Geräten funktionieren, weshalb trotz sorgfältig geplanten Testdurchläufe keine absolute Zuverlässigkeit garantiert werden kann.
- **Verfügbarkeit:** Die Webapplikation soll wie gewohnt einfach über einen beliebigen Browser zugreifbar sein und die Androidapplikation soll auf jedem Gerät laufen, dass mit Android läuft und natürlich internetfähig ist.

- Sicherheit: Sicherheit bezüglich der persönlichen Daten wird zum Teil durch den passwortgeschützten Login gewährleistet, wobei jeder Nutzer ein individuelles Passwort besitzt. Auf Profildaten haben nur der bestimmte eingeloggte Nutzer und der Admin Zugriff. Zusätzlich werden vor dem Versenden von Daten, diese via SSL verschlüsselt, was die Datensicherheit in unserem System garantiert.
- Wartbarkeit: Um eine gute Wartbarkeit zu gewährleisten, soll der Programmcode in Komponenten aufgeteilt werden, dessen Zuständigkeit klar definiert sein muss. Zusätzlich zu einer übersichtliche Programmstruktur, soll der Code sorgfältig und umfassend kommentiert werden.
- Portabilität: Da die Androidapplikation auf jedem androidfähigem Gerät laufen können soll, ist eine hohe Portabilität bereits gegeben. Durch Tests auf den verschiedensten Endgeräten kann die Applikation auf diesen absolut funktionstüchtig laufen.