Entwickler: Daniel Dobras, Fernando Pfennig

dobras@hm.edu, pfennig@hm.edu

Software Engineering I

Auftraggeber: Stephan Schiffner, Konrad Schmid

Wintersemester 2016/17

Ablieferungsdatum: 20.01.2017

Version 1.0

InstaLearnApp  
Pflichtenheft







**Inhaltsverzeichnis**

[I. Konzeption der InstaLearnApp 1](#_Toc471844651)

[1. Allgemeine Informationen zum zu entwickelnden Produkt 1](#_Toc471844652)

[a. Kurzbeschreibung des Produkts 1](#_Toc471844653)

[b. Anforderungen an ein Massenprodukt 1](#_Toc471844654)

[2. Visionen und Ziele 1](#_Toc471844655)

[a. Vision 1](#_Toc471844656)

[b. Ziele 1](#_Toc471844657)

[aa. Meilenstein I 1](#_Toc471844658)

[(1) Statusbericht 1](#_Toc471844659)

[(2) Lastenheft 2](#_Toc471844660)

[(3) Lauffähiger Prototyp 2](#_Toc471844661)

[bb. Meilenstein II 2](#_Toc471844662)

[(1) Statusbericht 2](#_Toc471844663)

[(2) Pflichtenheft 2](#_Toc471844664)

[(3) Lauffähige Web-Applikation 2](#_Toc471844665)

[3. Stakeholder 2](#_Toc471844666)

[4. Anforderungen an unser zu entwickelndes System 3](#_Toc471844667)

[a. Zwingende Mindestanforderungen – Version 1.0 3](#_Toc471844668)

[b. Weitergehende Implementierungsmöglichkeiten – Version 2.0 3](#_Toc471844669)

[c. Grenzen des Systems 4](#_Toc471844670)

[5. Qualitätsanforderungen an das zu entwickelnde System 4](#_Toc471844671)

[a. Funktionalität 4](#_Toc471844672)

[b. Zuverlässigkeit 4](#_Toc471844673)

[c. Benutzbarkeit 5](#_Toc471844674)

[d. Änderbarkeit 5](#_Toc471844675)

[e. Übertragbarkeit 5](#_Toc471844676)

[6. Zusätzliche selbst auferlegte Anforderungen an die Applikation (unique selling point) 5](#_Toc471844677)

[7. Grafische Übersicht 6](#_Toc471844678)

[a. Geschäftsprozessbeschreibung 6](#_Toc471844679)

[aa. Geschäftsprozesslandkarte 6](#_Toc471844680)

[bb. Textuelle Beschreibung 6](#_Toc471844681)

[b. Anwendungsfälle 7](#_Toc471844682)

[aa. Anwendungsfalldiagramm 7](#_Toc471844683)

[bb. Tabellarische Beschreibung der Anwendungsfälle 7](#_Toc471844684)

[(1) UC01 Schüler registrieren 7](#_Toc471844685)

[(2) UC02 Lehrer registrieren 9](#_Toc471844686)

[(3) UC03 Login 9](#_Toc471844687)

[(4) UC11 Frage ändern 10](#_Toc471844688)

[(5) UCXX Frage erstellen 11](#_Toc471844689)

[(6) UCXX Spiel spielen 12](#_Toc471844690)

[(7) UCXX Spielauswertungen anzeigen 13](#_Toc471844691)

[c. Fachklassendiagramm 13](#_Toc471844692)

[II. Glossar 14](#_Toc471844693)

[III. Ansprechpartner für Rückfragen 14](#_Toc471844694)

# I. Konzeption der InstaLearnApp

Zunächst sind die grundsätzlichen Rahmenbedingungen der Lernapplikation zu definieren.

## 1. Allgemeine Informationen zum zu entwickelnden Produkt

*bearbeitet von Daniel Dobras*

### a. Kurzbeschreibung des Produkts

Es ist eine Softwarelösung zu entwerfen und erarbeiten, die Schüler der Unterstufe des Gymnasiums beim Lernen unterstützt (fortan Lernapp). Die Lernapp wird im Auftrag von Herrn Schiffner und Herr Schmidt konzipiert und entwickelt. Vom Auftraggeber spezifizierte Anforderungen sind daher in erster Linie zu berücksichtigen und damit zu implementieren. Die Grundidee ist zumindest eine im Webbrowser zu verwendende Anwendung zu entwickeln, die auf Grund der geringen Altersstufe der Lernenden Benutzerfreundlichkeit und die damit verbundene Einfachheit der Bedienung als einer der wesentlichsten Aspekte vorsieht. Lehrer hingegen dürfen weitere Fragen nach Belieben hinzufügen oder bearbeiten. Darüber hinaus haben Lehrer die Möglichkeit auf die Spielergebnisse der Lernenden zuzugreifen und diese somit einzusehen.

### b. Anforderungen an ein Massenprodukt

Problematisch könnte jedoch vorliegend die Zielgruppe der Lernapp werden. Diese ist zahlenmäßig nicht begrenzt, womit im Rahmen des Projekts überdies die Anforderungen an ein massetaugliches Produkt zu berücksichtigen sind. Insbesondere soll hierbei abermals die Benutzerfreundlichkeit, sowie die Einfachheit der Nutzung besonders bei der Entwicklung der Softwarelösung berücksichtigt werden. Weiterführend ist eine einfache sprachliche Formulierung von Nöten, sodass eine möglichst große Gruppe von Lernenden die Nutzung der Applikation ermöglicht wird.

## 2. Visionen und Ziele

### a. Vision

*bearbeitet von Fernando Pfennig*

Ziel der MyLearnigApp ist es, den Schülern eine App bereitzustellen, mit der sie spielend lernen können und ihnen eine positive Einstellung gegenüber des Lernvorgangs vermittelt. Des Weiteren sollen die Lehrer bzw. Eltern in der Lage sein, Fragen der Schüler zu verwalten und so kontrollieren, ob der Schüler erfolgreich mit der App lernt.

### b. Ziele

*bearbeitet von Daniel Dobras*

Zur zielsicheren und pünktlichen Ablieferung des Projekts sind zwei wesentliche Meilensteine zu konkretisieren.

#### aa. Meilenstein I

Zum Abschluss des Meilenstein I sind bis zum 02.12.2016 folgende Artefakte abzuliefern. Sämtliche Dokumente sind im PDF-Format abzugeben.

##### (1) Statusbericht

Es ist zum Abschluss des Meilenstein I ein Statusbericht beizulegen. Dieser umfasst folgenden Inhalt:

* Auflistung bereits erfüllter Aufgaben, derzeitiger Stand des Projekts, sowie Funktionalitätserläuterung des Prototyps[[1]](#footnote-1)
* Projektplan für die Restlaufzeit des Projekts; umfasst Aktivitäten, Zuordnung zu den Teammitgliedern, Zeiträume (Aktivitäten sind in wöchentliche Abschnitte aufzu-gliedern)
* Risikoanalyse und dazugehörige Gegenmaßnahmen
* Aufwandsnachweis (tageweise Aufstellung der Arbeiten der einzelnen Mitarbeiter)

##### (2) Lastenheft

Über den Statusbericht hinaus ist ein Lastenheft anzufertigen und dem Auftraggeber im PDF-Format zu übermitteln.

##### (3) Lauffähiger Prototyp

Schließlich soll ein lauffähiger Prototyp mit abgeliefert werden, damit sich der Auftraggeber vom Fortschritt des Produkts überzeugen kann.

#### bb. Meilenstein II

Zum Abschluss des Meilensteins II, der letztlich auch die Deadline für dieses Projekt setzt, sind bis zum 20.01.2017 folgende Dokumente, sowie die lauffähige Web-Applikation in GIT abzugeben. Sämtliche Dokumente sind im PDF-Format abzugeben.

##### (1) Statusbericht

Es ist zum Abschluss des Meilenstein II ebenfalls ein Statusbericht beizulegen. Dieser umfasst folgenden Inhalt:

* Kurze Zusammenfassung und Überblick über das Projekt
* Ggfls. Hinweise auf Einschränkungen bei der Nutzung/ Hinweise zur Benutzung
* Aufwandsnachweis für das gesamte Projekt
* Rückblickende Analyse (sog. „Post-Mortem“)

- Was lief gut? Was lief schlecht?

- Verbesserungsvorschläge für zukünftige Projekte

- Feedback an den Auftraggeber bezüglich der Semesteraufgabe

##### (2) Pflichtenheft

Überdies ist ein Pflichtenheft zu führen. Dieses muss mindestens drei UML-Diagramme verschiedenen Typs enthalten. Darüber hinaus sind sämtliche implementierten Funktionen zu spezifizieren. Aus dieser Spezifikation muss hervorgehen welche Funktionen realisiert wurden und welche nicht.

##### (3) Lauffähige Web-Applikation

Die lauffähige Web-Applikation muss folgende Kriterien erfüllen:

* Zwingende Anforderungen (Version 1.0) müssen implementiert sein
* Zur Erreichung einer sehr guten Note sind darüber hinaus mindestens zwei Funktionen der Version 2.0 zu implementieren
* Die Web-Applikation ist im Rahmen einer kurzen Vorführung während des Praktikums anderen Teams vorzustellen und zu erläutern. Hierbei sind sämtliche Teammitglieder zur Anwesenheit verpflichtet
* Darüber hinaus ist eine Erklärung auszudrucken und ausgefüllt, sowie unterschrieben am letzten Vorlesungstermin abzugeben (Termin: 13.01.2017). Das PDF-Formular für die Erklärung wird vor dem Termin in Moodle bereitgestellt sein.

## 3. Stakeholder

*bearbeitet von Daniel Dobras*

**Auftraggeber:** Selbst verständlich ist der Auftraggeber als Interessenhalter besonders zu berücksichtigen. Von ihm geforderte Funktionen sind zu implementieren. Weitergehende Anweisungen oder Hinweise sind zu beachten und entsprechend Folge zu leisten.

**Lernende:** Weitergehend sind jedoch auch die Zielgruppe der Lernapp – die Lernenden selbst – zu beachten. Dadurch, dass die Anwendung für die Unterstufe des Gymnasiums gedacht ist, gilt es den Komplexitätsgrad möglichst gering zu halten.

**Lehrer:** Lehrer müssen die Applikation intuitiv anwenden können und darüber hinaus ebenfalls von der Einfachheit überzeugt werden, da eine Verwendung für deren Schüler andernfalls ausscheidet.

**Gesetzliche Vertreter:** Schließlich müssen ggfls. gesetzliche Vertreter der Schüler vom Nutzen der Anwendung überzeugt sein. Eltern sehen es nicht gerne, wenn ihre Kinder den ganzen Tag vor dem Computer sitzen. Hier gilt es die Applikation derart zu gestalten – insbesondere sachlich zu halten – dass auch Eltern von dem Nutzen überzeugt sind und ihr Kind sogar dazu anhalten die Anwendung als Ergänzung zu ihrem Lernprogramm zu verwenden.

4. Anforderungen an unser zu entwickelndes System

*bearbeitet von Daniel Dobras*

Folgende Anforderungen sind an die Anwendung zu stellen:

### a. Zwingende Mindestanforderungen – Version 1.0

Folgende Mindestanforderungen sind im Rahmen der InstaLearnApp zu implementieren:

|  |  |
| --- | --- |
| **Allgemeine Anforderungen an das System:** | **Implementierungsfortschritt** |
| Login über Dropdown-Liste (ohne Passwortabfrage) | Client:  Implementiert über Passwortabfrage |
| Unterscheidung zwischen Lerner/Lehrer bei der Nutzung der Applikation | implementiert |

|  |  |
| --- | --- |
| **Spezifisch für Lernende:** | **Implentierungsfortschritt** |
| Lerner können sich selbst registrieren | implementiert |
| Lerner können Spiele spielen | implementiert |
| Lerner bekommen ihr Spielergebnis angezeigt | implementiert |

|  |  |
| --- | --- |
| **Spezifisch für Lehrer:** | **Implementierungsfortschritt** |
| Lehrer können Fragen verwalten (anzeigen, hinzufügen, ändern) | Client:  Lehrer können sich Fragen anzeigen lassen und Fragen hinzufügen; Fragen bearbeiten nicht fertig implementiert und wird daher auch nicht angezeigt |
| Lehrer können Auswertungen über alle Spiele nach folgenden Kriterien durchführen:   * Alle Spiele anzeigen, inkl. Angabe von Spieler, Ergebnis, durchschnittliche Beantwortungszeit; Sortierung nach Ergebnis, durchschnittliche Beantwortungszeit * Alle Spiele pro Spieler anzeigen, inkl. Angabe von Ergebnis, durchschnittliche Beantwortungszeit | Nicht implementiert |

### b. Weitergehende Implementierungsmöglichkeiten – Version 2.0

*bearbeitet von Daniel Dobras*

Im Rahmen der Version 2.0 ist in der Applikation lediglich das Login-Verfahren mit Benutzername und Passwort umgesetzt werden. Anderweitige Optionen sind bislang nicht implementiert.

Folgende weiterführende Implementierungsoptionen bestehen:

|  |
| --- |
| Allgemeine weiterführende Implementierungsmöglichkeiten: |
| Login-Verfahren mit Name und Passwort |
| Scoreboard über alle Spieler |

|  |
| --- |
| Spezifisch für Lernende: |
| Lerner können optional das aktuelle Spiel speichern und später weiterspielen |
| Implementierung eines Progression-Systems |
| Implementierung eines Features, welches auf die Hardware Funktionen eines Mobilgeräts zugreift (z.B. Kamera, GPS, …) |

|  |
| --- |
| Spezifisch für Lehrer: |
| Fragen-Kategorien-Verwaltung (anzeigen, hinzufügen, ändern) |
| Löschen von Fragen und Kategorien (Löschen nur dann möglich, sofern es noch keine zugehörigen Spiele gibt, andernfalls nur deaktivieren) |
| Benutzerverwaltung (anzeigen, hinzufügen, ändern) |
| Weitere Auswertung über Spiele/Spieler |

### c. Grenzen des Systems

*bearbeitet von Daniel Dobras*

|  |
| --- |
| **Speziell nicht geforderte Implementierungen:** |
| Lehrer müssen das Spiel nicht spielen können (strikte Trennung zwischen Lernenden und Lehrer) |
| Lehrer haben nicht die Möglichkeit, sich selbst zu registrieren |
| Lernende selbst haben nicht die Option weitere Fragen hinzufügen bzw. generell zu verwalten |
| Keine Austauschfunktion zwischen Lernenden und Lehrer innerhalb der Anwendung (wie Chatfunktion o.ä.) |
| Keine Cloudspeicherung der Fragen, lediglich lokale Speicherung |
| Keine spezifische grafische Nutzungsoberfläche |
| Keine Hilfestellungsfunktion bei Fragen |

## 5. **Qualitätsanforderungen an das zu entwickelnde System**

*bearbeitet von Fernando Pfennig*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Sehr wichtig | Wichtig | Weniger wichtig | Nicht relevant |
| Funktionalität |  |  |  |  |
| Zuverlässigkeit |  |  |  |  |
| Benutzbarkeit |  |  |  |  |
| Änderbarkeit |  |  |  |  |
| Übertragbarkeit |  |  |  |  |

### a. Funktionalität

An nächster Stelle liegt die Stabilität der Applikation. Damit der Schüler möglichst frustfrei lernen kann, ist es ebenfalls wichtig, dass die App nicht abstürzt und die Performance gut ist. Damit möglichst viele Schüler mit diesem Anwenderprogramm arbeiten bzw. lernen können, sollen die Systemanforderungen so gering wie möglich gehalten werden., um eine hohe Anzahl an Benutzern zu erreichen.

Hinzu kommen alle Funktionen der Version 1.0, die funktionsfähig implementiert werden und ebenfalls zwei Anwendungsfälle der Version 2.0.

### b. Zuverlässigkeit

Außerdem sollen bei der Verwendung der App keine schwerwiegenden Bugs auftreten, wie z.B. das Löschen der Highscores, obwohl eine Frage gelöscht werden sollte. Dazu gehört auch ein gut ausgebildeter Support, der leicht erreichbar ist und so die Bugs schnell beheben kann. Dieser Punkt ist sehr wichtig um frustfreies Lernen zu ermöglichen und dem Lerner viel Spaß am Spiel zu vermitteln. Außerdem soll jeder Lerner dieselben Funktionen haben und jeder Lehrer ebenfalls.

### c. Benutzbarkeit

Durch ein möglichst einfach gehaltenes User Interface soll die InstaLearnApp in der Lage sein, von Kindern und Erwachsenen benutzt zu werden. So werden alle Buttons eindeutig bezeichnet, sodass der Lehrer bzw. Erwachsene sich nicht zu lange mit der Bedienung der App auseinandersetzen müssen. So wird auch die Verwendung durch jüngere Schüler einfacher, da es keine komplizierten Methoden gibt, sondern einfach nur „Spielen“.

### d. Änderbarkeit

Dieser Punkt ist weniger wichtig, da alle relevanten Anwendungsfälle implementiert werden und somit das Programm nicht sehr flexibel sein muss.

### e. Übertragbarkeit

Im Moment ist die Applikation nur auf einem Gerät zu verwenden, d.h. die Fragen und Benutzer werden lokal gespeichert.

## 6. Zusätzliche selbst auferlegte Anforderungen an die Applikation (unique selling point)

*Bearbeitet von Fernando Pfennig*

Das Entwicklerteam möchte dem Auftraggeber über seine genauen spezifizierten Anforderungen hinaus noch eine weitere Funktion zur Verfügung stellen, aus denen klar wird, warum unser Softwareentwicklungsteam der Konkurrenz weit voraus ist.

Der Lerner soll für die Verwendung der InstaLearnApp belohnt werden. Dies erreichen wir durch die Einführung einer Erfahrungsleiste in das Spiel. Nach jedem beendeten Spieldurchgang erhält der Lerner eine Anzahl an Erfahrungspunkten, die je nach Anzahl richtig beantworteter Fragen variiert. Je mehr Fragen er in sich einem Durchgang stellt, desto höher ist die maximale Menge an Erfahrungspunkten, die er bekommen kann.

Sobald der Lerner ein Level aufsteigt, bekommt er z.B. die Möglichkeit, in der App einen Teil der UI farblich zu verändern. Je höher das Level desto „exklusiver“ die Farbe, so erhält man erst nach vielen Level die schwarze Farbe, des Weiteren erhöht sich auch die Menge an Erfahrungspunkten, die man benötigt, um Level aufzusteigen.

**Beispiel für eine Progression-Leiste:**



**Farben zum Freischalten – am Beispiel der App Highrise:**



## 7. Grafische Übersicht

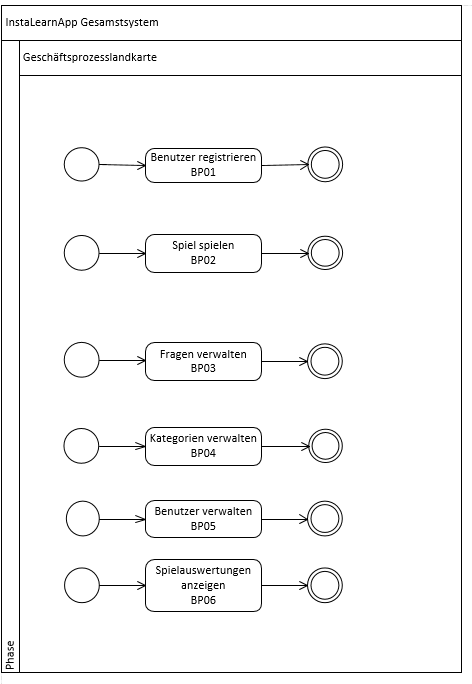
Zur Visualisierung der Projektanforderungen sind dem Lastenheft grafische Übersichten beigefügt worden.

## a. Geschäftsprozessbeschreibung

*bearbeitet von Daniel Dobras*

### aa. Geschäftsprozesslandkarte

*bearbeitet von Daniel Dobras*



### bb. Textuelle Beschreibung

Geschäftsprozesse werden hier zur Erleichterung des Verständnisses textuell erläutert.

**BP01 Benutzer registrieren**

Im Rahmen des BP01 sollen Benutzer registriert werden. Nutzer können sich als Lehrer oder als Lernende im System registrieren, sofern sie vorher bereits noch nicht registriert sind. Die einzelne Ausgestaltung wird im Rahmen der use cases näher definiert.

**BP02 Spiel spielen**

BP02 definiert sämtliche Funktionen, die im Zusammenhang zum Spiel spielen erforderlich sind. Es wird das Spiel gestartet, optional lässt sich in Version 2.0 das Spiel speichern und schließlich wird nach Beantwortung sämtlicher Fragen dem Benutzer das Spielergebnis angezeigt. Der Benutzer kommt nach seiner Wahl die gewünschte Menge an Fragen und kann diese im Rahmen von Multiple Choice Fragen oder offenen Fragen beantworten.

**BP03 Fragen verwalten**

BP03 bietet dem Lehrer verschiedene Optionen, um Fragen zu verwalten. Diese beinhalten das Anzeigen von Fragen, das Hinzufügen von Fragen, die Änderung von Fragen, sowie das Löschen von Fragen.

**BP04 Kategorien verwalten**

In Version 2.0 können Lehrer über die Fragenverwaltung hinaus auch noch die Kategorien der Fragen verwalten. Beispielsweise lassen sich neue Kategorien hinzufügen oder bestehende Kategorien bearbeiten.

**BP05 Benutzer verwalten**

Im Rahmen des Geschäftsprozesses „Benutzer verwalten“ haben Lehrer die Möglichkeit einzelne Benutzer anzuzeigen, hinzuzufügen oder deren Daten zu ändern.

**BP06 Spielauswertungen anzeigen**

Lehrer können die gesamten Spielergebnisse aller Benutzer einsehen oder sich die Spielergebnisse eines einzelnen Schülers anzeigen lassen. In Version 2.0 können sich Schüler auf einem Scoreboard miteinander vergleichen.

## b. Anwendungsfälle

*bearbeitet von Fernando Pfennig*

### aa. Anwendungsfalldiagramm



### bb. Tabellarische Beschreibung der Anwendungsfälle

Zur Erleichterung des Verständnisses sind hier ausgewählte use cases tabellarisch beschrieben.

#### (1) UC01 Schüler registrieren

*bearbeitet von Daniel Dobras*

|  |  |
| --- | --- |
| UC01 Schüler registrieren | |
| Kurzbeschreibung | Der Lernende startet die Applikation und möchte das Spiel spielen. Bevor er das Spiel jedoch starten kann, muss er sich zuvor einloggen. Einloggen können sich jedoch nur Personen, die bereits im System registriert sind. Daher muss sich der Schüler vor Benutzung der Applikation mit einem Benutzernamen und Passwort registrieren. |
| Hauptkunde | Lernende |
| Auslöser | Lernender klickt auf die Schaltfläche „Neu bei InstaLearn“ und wird somit auf die Registrierungsseite weitergeleitet |
| Vorbedingung | keine |
| Ergebnis (normal) | Der Schüler wird mit ID, Namen und als Schüler in der Datenbank „User“ angelegt. |

|  |  |
| --- | --- |
| Normalablauf | |
| Ergebnis | 1. Lernender wird in die Datenbank eingefügt und hat sich somit im System registriert  2. Lernender wird über erfolgreiche Registrierung informiert |
| Ablauf | 1. Start der Applikation führt zur Login-Seite  2. Klicken auf die Schaltfläche „Neu bei InstaLearn“  3. Lernender muss seinen Benutzernamen und bei Version 2.0 überdies sein Passwort eingeben  4. Klicken auf die Schaltfläche „Registrieren“  5. Benutzerdaten werden an Datenbank übermittelt und diese gibt eine Erfolgsmeldung an den Client zurück  6. Lernender wird über erfolgreiche Registrierung informiert  7. Rückführung zur Login-Seite |

|  |  |
| --- | --- |
| Alternativablauf – Lernender hat nicht alle Registrierungsfelder ausgefüllt | |
| Ergebnis | 1. Lernender wird in die Datenbank eingefügt und hat sich somit im System registriert  2. Lernender wird über erfolgreiche Registrierung informiert |
| Ablauf | 1. – 3. Wie im Normalfall  4. Klicken auf Schaltfläche „Registrieren“ ohne vorheriges ausfüllen der erforderlichen Felder für Benutzername und ggfls. Passwort  5. Lernender wird darüber informiert, dass nicht sämtliche erforderlichen Felder ausgefüllt worden sind  6. Lernender füllt weiter die erforderlichen Registrierungsfelder aus  7. Sofern sämtliche Felder ordnungsgemäß ausgefüllt sind weiteres Vorgehen wie Schritt 5 bis Schritt 7 des Normalfalls, ansonsten sind Schritt 4 bis Schritt 6 des Alternativablaufs zu wiederholen. |

|  |  |
| --- | --- |
| Alternativablauf – Benutzername vergeben | |
| Ergebnis | 1. Lernender wird in die Datenbank eingefügt und hat sich somit im System registriert  2. Lernender wird über erfolgreiche Registrierung informiert |
| Ablauf | 1.-4. Wie Normalfall  5. Benutzerdaten werden an die Datenbank übermittelt und es stellt sich heraus, dass der Benutzername bereits vergeben ist  6. Server sendet eine Fehlermeldung an den Client, womit dieser weiß, dass kein weiterer Nutzer mit diesem Benutzernamen registriert werden kann  7. Lernender wird darüber informiert und muss einen freien Benutzernamen wählen  8. Klicken auf die Schaltfläche „Registrieren“  9. Benutzerdaten werden erneut übermittelt  10. Wenn Benutzername noch nicht vergeben ist erhält der Client eine Erfolgsmeldung, ansonsten Schritt 6 bis Schritt 9 wiederholen  11. Lernender wird über erfolgreiche Registrierung informiert  12. Rückführung zur Login-Seite |

#### (2) UC02 Lehrer registrieren

*bearbeitet von Daniel Dobras*

|  |  |
| --- | --- |
| UC02 Lehrer registrieren | |
| Kurzbeschreibung | Lehrer möchte sich im System der InstaLearnApp registrieren, um auf administrative Funktionen zugreifen zu können. Um missbräuchlichen Anmeldungen vorzubeugen, lassen sich Lehrer nur direkt von den Mitarbeitern der InstaLearnApp in das System aufnehmen. Lehrer müssen daher eine E-Mail an den Service der InstaLearn GbR senden, um von diesen überprüft und entsprechend freigeschaltet zu werden |
| Hauptkunde | Lehrer |
| Auslöser | E-Mail an die Service-Abteilung von InstaLearn |
| Vorbedingung | Konkrete Nachweise für die Eigenschaft als Lehrer und entsprechende Zusicherung, um entsprechendes missbräuchliches Verhalten möglichst auszuschließen |
| Ergebnis (Normal) | Lehrer wird im System mit seinem Benutzernamen und seiner Eigenschaft als Lehrer registriert. |

|  |  |
| --- | --- |
| Normalablauf | |
| Ergebnis | 1. Lehrer wird im System als Lehrer mit Benutzernamen und ggfls. Passwort angelegt  2. Benachrichtigung des Lehrers per E-Mail, dass die Registrierung erfolgreich war |
| Ablauf | 1. Lehrer sendet E-Mail mit relevanten Nachweisen seiner Eigenschaft als Lehrer und seinem Wunsch als Lehrer registriert zu werden, um auf administrative Funktionen zugreifen zu können  2. Bearbeitung und Überprüfung durch die Service-Abteilung  3. Manuelle Registrierung des Lehrers in die Datenbank durch die Service-Abteilung  4. Bestätigung über erfolgreiche Registrierung via E-Mail an den Lehrer |

#### (3) UC03 Login

*bearbeitet von Daniel Dobras*

|  |  |
| --- | --- |
| UCXX Login | |
| Kurzbeschreibung | Der Benutzer, Lernender oder Lehrer möchte sich mit seinen Nutzerinformationen in das System einloggen, um auf die Funktionen zugreifen zu können |
| Hauptkunde | Lernende, Lehrer |
| Auslöser | Klicken auf die Schaltfläche „Einloggen“ |
| Vorbedingung | Benutzer ist bereits im System registriert |
| Ergebnis(Normal) | Benutzer ist im System eingeloggt und kriegt entsprechend seinem Benutzerstatus als Lernender oder Lehrer entsprechende Optionen angezeigt |

|  |  |
| --- | --- |
| Normalablauf | |
| Ergebnis | 1. Benutzer wird im System eingeloggt  2. Benutzer kriegt entsprechend seinem Benutzerstatus verschiedene Optionen angezeigt |
| Ablauf | 1. Eingeben der erforderlichen Login-Felder  2. Einloggen in das System über Schaltfläche „Einloggen“  3. Überprüfung, ob Nutzer im System registriert ist  4. Positive Rückmeldung  5. Anzeigen der entsprechenden Funktionen gemäß des Benutzerstatus |

|  |  |
| --- | --- |
| Alternativablauf – Login-Felder nicht ausgefüllt | |
| Ergebnis | 1. Benutzer wird im System eingeloggt  2. Benutzer kriegt entsprechend seinem Benutzerstatus verschiedene Optionen angezeigt |
| Ablauf | 1. Klicken auf die Schaltfläche „Einloggen“ ohne die erforderlichen Login-Felder ausgefüllt zu haben  2. Rückmeldung an den Benutzer, dass erforderliche Felder ausgefüllt sein müssen  3. wie Normalablauf ab Schritt 2 |

|  |  |
| --- | --- |
| Fehlerfall – Benutzer nicht registriert | |
| Ergebnis | Benutzer wird nicht im System eingeloggt und verbleibt auf der Login-Seite |
| Ablauf | 1. bis 3. Wie Normalfall  4. Negative Rückmeldung  5. Benutzer wird nicht in das System eingeloggt und verbleibt auf der Login-Seite |

#### (4) UC11 Frage ändern

*bearbeitet von Daniel Dobras*

|  |  |
| --- | --- |
| UCXX Frage ändern | |
| Kurzbeschreibung | Ein Lehrer möchte eine bereits existierende Frage ändern. |
| Hauptkunde | Lehrer |
| Auslöser | Klick auf die Schaltfläche „Frage ändern“ |
| Vorbedingung | Lehrer ist im System eingeloggt und hat auf die Schaltfläche „Frage verwalten“ geklickt |
| Ergebnis(Normal) | Bestehende Frage wird durch die überarbeitete Version ersetzt. |

|  |  |
| --- | --- |
| Normalablauf | |
| Ergebnis | Frageänderung wird in der Datenbank abgespeichert und es erfolgt eine Rückmeldung an den Lehrer, dass Änderung erfolgreich war |
| Ablauf | 1. Klicken auf die Schaltfläche „Fragen verwalten“  2. Klicken auf die Schaltfläche „Frage ändern“  3. Auswahl der konkreten Frage  4. Bearbeitung der einzelnen Felder durch den Lehrer  5. Klicken auf die Schaltfläche „Änderungen speichern“ |

|  |  |
| --- | --- |
| Fehlerfall – Abbrechen des Änderungsvorgangs | |
| Ergebnis | 1. Änderungen werden nicht übernommen  2. Lehrer wird auf die Lehrer-Startseite zurückgeleitet |
| Ablauf | Während des gesamten Prozesses könnte der Lehrer den Vorgang abbrechen:  1. bis 3. Wie Normalfall  4. Klicken auf die Schaltfläche „Abbrechen“  5. Änderungen werden NICHT gespeichert  6. Lehrer wird auf die Verwaltungsseite zurückgeführt |

#### (5) UCXX Frage erstellen

*bearbeitet von Fernando Pfennig*

|  |  |
| --- | --- |
| UCXX Frage erstellen | |
| Kurzbeschreibung | Der Lehrer möchte eine neue Frage erstellen |
| Hauptkunde | Lehrer |
| Auslöser | Klicken auf die Schaltfläche „Frage erstellen“ |
| Vorbedingung | Lehrer ist eingeloggt |
| Ergebnis(Normal) | Eine neue offene oder Multiple-Choice-Frage wird in der Datenbank unter einer bestimmten Kategorie gespeichert. |

|  |  |
| --- | --- |
| Alternativablauf 1 – Erstellung einer offenen Frage | |
| Ergebnis | 1. Die neue offene Frage wird in die Datenbank eingefügt und kann somit für zukünftige Spiele verwendet werden  2. Der Lehrer wird über die erfolgreiche Erstellung der Frage informiert |
| Ablauf | 1. Lehrer klickt auf „Fragen verwalten“  2. Lehrer klickt auf „Frage erstellen“  3. Lehrer klickt auf „Offene Frage erstellen“  4. Lehrer sucht eine Kategorie aus  5. Lehrer gibt eine Frage ein  6. Lehrer gibt die richtige Antwort auf die Frage an  7. Lehrer klickt auf „Offene Frage erstellen!“ |

|  |  |
| --- | --- |
| Alternativablauf 2 – Erstellung einer Multiple-Choice-Frage | |
| Ergebnis | 1. Die neue Multiple-Choice-Frage wird in die Datenbank eingefügt und kann somit für zukünftige Spiele verwendet werden  2. Der Lehrer wird über die erfolgreiche Erstellung der Frage informiert |
| Ablauf | 1. Lehrer klickt auf „Fragen verwalten“  2. Lehrer klickt auf „Frage erstellen“  3. Lehrer klickt auf „Multiple-Choice-Frage erstellen“  4. Lehrer sucht eine Kategorie aus  5. Lehrer gibt eine Frage ein  6. Lehrer gibt vier mögliche Antworten an  7. Lehrer gibt die richtige Antwort auf die Frage an  8. Lehrer klickt auf „Multiple-Choice-Frage erstellen!“ |

|  |  |
| --- | --- |
| Fehlerfall – Ein oder mehrere der benötigten Felder nicht ausgefüllt | |
| Ergebnis | 1. Frage wird nicht erstellt  2. Lehrer wird aufgefordert, alle Felder auszufüllen  3. Lehrer bleibt auf der Seite |
| Ablauf | 1. bis 3. wie in den Alternativabläufen  4. Lehrer füllt ein oder mehrere Felder nicht aus  5. Lehrer wird darüber informiert, dass nicht alle Felder ausgefüllt worden sind  6. Lehrer bleibt auf derselben Seite |

#### (6) UCXX Spiel spielen

*bearbeitet von Fernando Pfennig*

|  |  |
| --- | --- |
| UCXX Spiel spielen | |
| Kurzbeschreibung | Der Lerner möchte das Spiel spielen |
| Hauptkunde | Lernende |
| Auslöser | Klicken auf die Schaltfläche „Spiel starten!“ |
| Vorbedingung | Lernender ist eingeloggt |
| Ergebnis(Normal) | Lernender bekommt die Anzahl der Fragen aus einer Kategorie nacheinander angezeigt und wird nach deren Beantwortung auf den Ergebnisbildschirm weitergeleitet |

|  |  |
| --- | --- |
| Normalablauf | |
| Ergebnis | Lernender beantwortet eine davor festgelegte Anzahl an Fragen aus einer Kategorie und wird anschließend auf den Ergebnisbildschirm weitergeleitet |
| Ablauf | 1. Lernender bekommt die erste Frage angezeigt  2. Bei einer Multiple Choice Frage sucht der Lernende seine Antwort aus  3. Bei einer offenen Frage schreibt der Lernende seine Antwort in das leere Feld  4. Der Lernende drückt auf die Schaltfläche „Antworten“  5. Dem Lernenden wird angezeigt, ob die Frage richtig beantwortet wurde  6. Die nächste Frage wird angezeigt  1 bis 6: Wird bis zur letzten Frage wiederholt  7. Nach Beantwortung der letzten Frage wird das Spielergebnis mit der dazugehörigen Note gezeigt. Des Weiteren wird der Benutzername, die Kategorie, die Anzahl der Fragen, die Anzahl der richtig beantworteten Fragen und deren relativer Anteil angezeigt |

|  |  |
| --- | --- |
| Fehlerfall – Nicht genügend Fragen stehen zur Verfügung | |
| Ergebnis | Benutzer wird nicht zu dem Spiel weiter geleitet |
| Ablauf | 1. Lernenden wird gezeigt, es stehen nicht genügend Fragen zur Verfügung, um das Spiel mit der Anzahl an Fragen zu spielen  2. Lernenden wird hingewiesen, sich an den Administrator zu wenden  3. Weiterleitung zur Startseite |

|  |  |
| --- | --- |
| Fehlerfall – Abbruch des Spiels | |
| Ergebnis | 1. Spiel wird durch Beenden der Applikation abgebrochen  2. Lernender muss wieder von vorne anfangen  3. Bisherige Spielresultate gehen verloren |
| Ablauf | 1. Lernender beendet die Applikation bevor er die letzte Frage beantwortet hat  2. Erneuter Login des Lernenden |

#### (7) UCXX Spielauswertungen anzeigen

*bearbeitet von Fernando Pfennig*

|  |  |
| --- | --- |
| Normalablauf | |
| Ergebnis | 1. Die vorhandenen Spielauswertungen werden angezeigt |
| Ablauf | 1. Lehrer klickt auf „Spielauswertungen anzeigen!“  2. Es werden alle verfügbaren Spielergebnisse angezeigt  3. Lehrer klickt auf „Ergebnisse auswerten!“ |

|  |  |
| --- | --- |
| Alternativablauf – Keine Spielergebnisse in der Datenbank | |
| Ergebnis | Der Lehrer wird darauf hingewiesen, dass es keine Spielergebnisse in der Datenbank gibt |
| Ablauf | 1. Lehrer klickt auf „Spielauswertungen anzeigen!“  2. Lehrer wird angezeigt, es gäbe keine vorhandenen Spiel-ergebnisse  3. Lehrer bleibt auf derselben Seite |

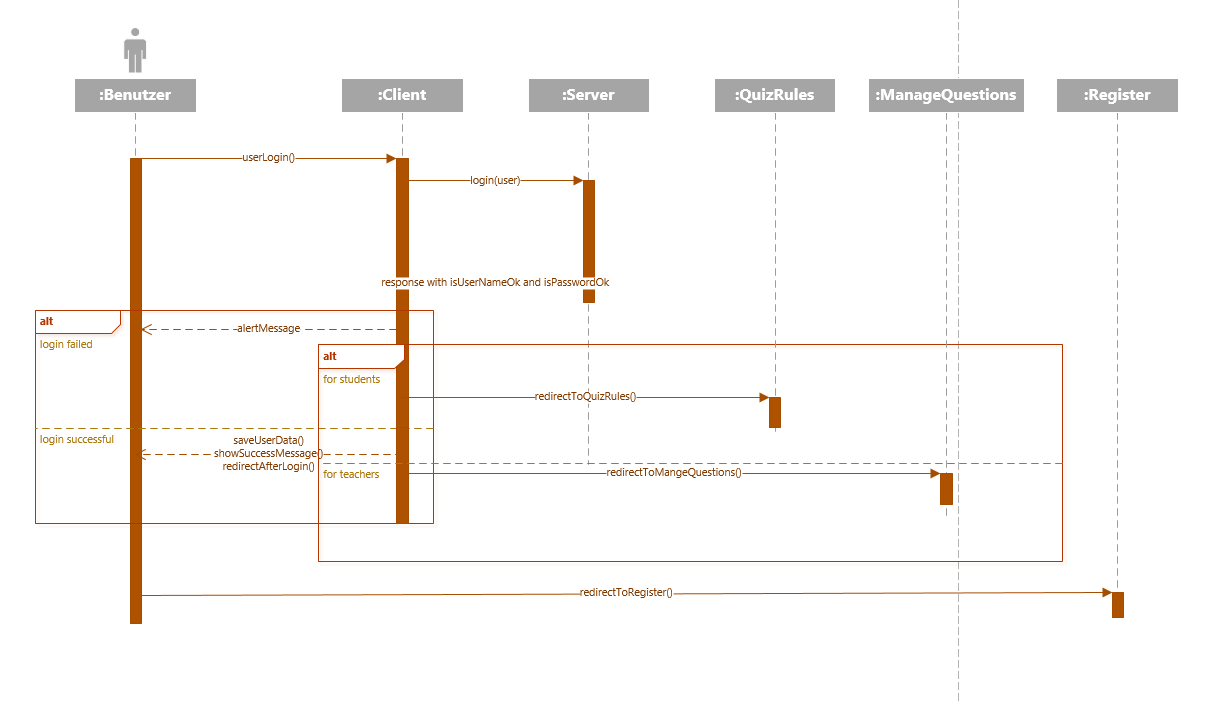
## c. Fachklassendiagramm

*bearbeitet von Fernando Pfennig*



## d. Sequenzdiagramm für den Benutzerlogin

*bearbeitet von Daniel Dobras*

Im Rahmen des Benutzerlogins arbeiten viele Klassen zusammen. Um die Beziehungen untereinander leichter erkennbar zu machen, soll dieses Sequenzdiagramm grafisch eine Hilfestellung bieten. Für eine erleichterte Lesbarkeit empfiehlt es sich hier die Zoomfunktion zu verwenden.

Zu sehen ist hier der Vorgang, der sich abspielt, wenn sich ein Benutzer im System mit seinem Benutzernamen und Passwort anmelden möchte (userLogin( )). Zunächst wird vom Server überprüft, ob ein Benutzer mit dem angegebenen Namen bereits im System registriert ist und ob das angegebene Passwort mit demjenigen des registrierten Benutzers übereinstimmen. Je nachdem wird eine Fehlermeldung an den Benutzer zurückgesendet oder der Benutzer wird im System eingeloggt. Anschließend wird der Benutzer gemäß seinem Benutzerstatus zu einer weiteren Seite weitergeleitet. Schüler werden direkt zu den Spielregeln geführt, wohingegen Lehrer zu der Übersicht gelangen, an der sie verschiedene Optionen für die Fragenverwaltung angezeigt bekommen.

Gemäß dem Falle, dass der Benutzer noch nicht im System registriert ist, kann er im Login-Bildschirm auf eine Schaltfläche klicken, die ihn zur Registrierungsseite weiterleitet.

# II. Glossar

*bearbeitet von Daniel Dobras*

|  |  |
| --- | --- |
| Geschäftsprozess/  business process (Abk. BP) | Ein Geschäftsprozess ist eine Menge logisch verknüpfter Einzeltätigkeiten (Auf-gaben, Aktivitäten), die ausgeführt werden, um ein bestimmtes geschäftliches oder betriebliches Ziel zu erreichen. |
| Anwendungsfall/  use case (Abk. UC) | Ein Anwendungsfall bündelt alle möglichen Szenarien, die eintreten können, wenn ein Akteur versucht, mit Hilfe des betrachteten Systems ein bestimmtes fachliches Ziel zu erreichen. |
| Human Resource (Abk. HR) | Als Human Resources bezeichnet man die Resourcen eines Unternehmens in Bezug auf das Wissen, die Fähigkeiten und die Motivation der Mitarbeiter. |
| Meilenstein | Als Meilenstein sind im Rahmen dieses Projekts besondere Zeitpunkte zu denen gewisse Funktionen bereits zwingend implementiert sein müssen. |
| Special feature / unique selling point | Das special feature meint eine vom Team selbst auferlegte Idee, um sich von etwaigen Konkurrenten absetzen zu können. |

# III. Ansprechpartner für Rückfragen

**Auftraggeber:**

|  |  |
| --- | --- |
| Stephan Schiffner | stephan.schiffner@hm.edu |
| Konrad Schmid | Ifw14113@cs.hm.edu |

**Entwickler:**

|  |  |
| --- | --- |
| Daniel Dobras | dobras@hm.edu |
| Fernando Pfennig | pfennig@hm.edu |

1. Siehe hierfür unter cc. Lauffähiger Prototyp [↑](#footnote-ref-1)