

**Titel**

**STUDIENARBEIT**

des Studiengangs Angewandte Informatik  
an der Dualen Hochschule Baden-Württemberg Karlsruhe  
von

**Mehmet Ali Incekara & Tom Wolske**

Abgabedatum 12. Januar 2017

<b>Bearbeitungszeitraum</b>	12 Wochen
<b>Matrikelnummer</b>	12345678 & 1156973
<b>Kurs</b>	TINF14B2
<b>Gutachter der Studienakademie</b>	Prof. Dr. Kay Berkling

# Erklärung

Gemäß §5 (2) der „Studien- und Prüfungsordnung DHBW Technik“ vom 18. Mai 2009 erkläre ich hiermit,

1. dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.
2. dass die Übernahme von Zitaten und Gedankengut anderer Autoren gekennzeichnet wurde.
3. dass die eingereichte elektronische Fassung exakt mit der schriftlichen übereinstimmt.
4. dass ich die Projektarbeit keiner externen Prüfung vorgelegt habe.

Karlsruhe, den 12. Januar 2017

Ort, Datum

Tom Wolske

Karlsruhe, den 12. Januar 2017

Ort, Datum

Mehmet Ali Incekara

# Inhaltsverzeichnis

<b>Abkürzungsverzeichnis</b>	<b>IV</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	2
1.2 Aufbau der Arbeit . . . . .	2
1.3 Ziel der Arbeit . . . . .	2
<b>2 NoRPG</b>	<b>3</b>
2.1 Gedanken . . . . .	4
2.2 Story . . . . .	5
2.3 Abgrenzung zu MOOC . . . . .	5
<b>3 Software Requirements Specification</b>	<b>6</b>
3.1 Einführung . . . . .	6
3.1.1 Zweck . . . . .	6
3.1.2 Umfang . . . . .	6
3.2 Allgemeine Beschreibung . . . . .	7
3.2.1 Produktperspektive . . . . .	7
3.2.2 Produktfunktionen . . . . .	8
3.2.3 Benutzermerkmale . . . . .	8
3.2.4 Einschränkungen . . . . .	9
3.2.5 Annahmen und Abhängigkeiten . . . . .	10
3.2.6 Aufteilung der Anforderungen . . . . .	10
3.3 Spezifische Anforderungen . . . . .	10
3.3.1 Externe Schnittstellen . . . . .	10
3.3.2 Funktionale Anforderungen . . . . .	15
3.3.3 Performanz Anforderungen . . . . .	19
3.3.4 Datenbank Anforderungen . . . . .	19
3.3.5 Entwurfsbeschränkungen . . . . .	19
3.3.6 Benutzerfreundlichkeit . . . . .	19
3.3.7 Zuverlässigkeit . . . . .	19
3.3.8 Verfügbarkeit . . . . .	20
3.3.9 Sicherheit . . . . .	20

3.3.10 Wartbarkeit . . . . .	20
3.3.11 Portabilität . . . . .	20
<b>4 Technische Grundlagen</b>	<b>21</b>
4.1 Unity3D . . . . .	21
4.2 Visual Studio . . . . .	22
4.3 C Sharp . . . . .	22
4.4 SQL . . . . .	22
<b>5 Umsetzung</b>	<b>23</b>
5.1 App . . . . .	23
5.2 Datenbank auf dem Handy . . . . .	23
5.3 Datenbank auf dem Server . . . . .	23
<b>6 Fazit und Ausblick</b>	<b>24</b>
6.1 Fazit . . . . .	24
6.2 Ausblick . . . . .	24
<b>Literaturverzeichnis</b>	<b>VII</b>
<b>Anhang</b>	<b>VIII</b>

# **Abkürzungsverzeichnis**

# Abbildungsverzeichnis

1	Login-Screen Mockup . . . . .	11
2	Register-Screen Mockup . . . . .	12
3	Register-Screen Mockup . . . . .	12
4	HUD Mockup . . . . .	13
5	HUD offenes Menü Mockup . . . . .	13
6	Fortschritt-Screen Mockup . . . . .	14
7	Overall Use Case Diagramm . . . . .	15
8	Create Character Activity UML . . . . .	16

## Listings

# 1 Einleitung

Über Spiele zu spieleplattform und darüber auf den lernende Effekt oder über lernen auf multimediales lernen zu einer Plattform zum lernen bestehend aus einem spiel

Spiele, jeder kennt sie und spielt regelmäßig welche. Sie sind ein fester bestandteil unserer Kultur und das schon seit tausenden Jahren. Die ersten Gesellschaftsspiele wurden noch im Sand mit Stöcken oder Steinen gespielt. Eines der frühesten Spiele wird auch heutzutage noch gespielt, dabei handelt es sich um Mühle, ein Spiel welches die Ägypter vor bereits 3000 Jahren gespielt haben.(<http://www.gesellschaftsspiele.de/geschichte-brettspiele/>) Spiele haben sich seit dem jedoch weiterentwickelt und dienen heutzutage nicht nur zum munteren Zeitvertreib.

Ob als Brett, Karten oder Glückspiel, Spiele sind überall zu finden und jeder kann sie spielen. Seit 1972 entwickeln sich darüber hinaus weitere Spiele, Videospiele. Sie nutzen die immer größer werdende Rechenleistung von Computern aus, um uns immer realistischer aussehender Spiele zu liefern. Um den Überblick über die Vielzahl an Videospielen zu behalten, haben sich in den letzten Jahren verschiedene Plattformen etabliert, die versuchen dem Nutzer das zu bieten, was sie suchen. Dabei bieten diese viele verschiedene Arten von Spielen an, die einen beim Spielen die Zeit vergessen lassen. Allerdings können Spiele uns nicht nur die Zeit vergessen lassen und für heitere Stunden sorgen, sie können uns auch Wissen vermitteln. Sei es durch eine Geschichte die sich real abgespielt hat, wie der erste Weltkrieg, oder anderes. Dieses Wissen wird unterbewusst an den Nutzer vermittelt, ohne das er aktiv versucht dieses zu lernen.

Für diesen Zweig hat sich eine eigene Branche entwickelt, welche sich mit Lernspielen befasst und versucht uns, über Videospiele, dieses Wissen zu vermitteln. Viele dieser Spiele nutzen bekannte Figuren, welche die Kinder aus dem Fernseh kennen, um dieses Wissen zu vermitteln.

Diese Spiele werden hauptsächlich in den Schulen eingesetzt um den Kindern Wissen spielerisch zu vermitteln. Jedoch profitiert nicht jedes Kind von diesem Vorteil. Sei es, weil die Schule keine Computer hat, oder weil das Kind nicht zur Schule gehen kann. Für diesen Zweck wurde die Plattform Hone entwickelt, mit der Kinder, die nicht zur Schule gehen können, die Möglichkeit haben, Wissen zu erlangen.



## **1.1 Motivation**

Bei Hone handelt es sich um eine Spieleplattform auf der sich Kinder, bevorzugt aus Regionen in denen Bildung mangelhaft ist, anmelden können. Auf dieser Plattform haben Sie dann eine Ansicht Ihrer, durch Spiele gelernte Kompetenzen, bzw. können sich die Kinder dort neue Spiele herunterladen, um weitere Kompetenzen zu erwerben. Das Aussehen und die Bedienung dieser Plattform ist allerdings nicht reizvoll für Kinder gestaltet.

Deshalb soll eine Spieleapp entwickelt werden, in der die Kinder auf spielerischer Weise Fortschritte machen. Dabei werden die Inhalte von Hone nicht übernommen und die App funktioniert unabhängig davon. Durch die Umsetzung als App gelingt es darüber hinaus den Kindern eine Offlineplattform zu geben, welche sie unabhängig von der Internetverbindung nutzen können.

## **1.2 Aufbau der Arbeit**

Die Arbeit beginnt mit der grundlegenden Idee hinter der App. Darauf folgend wird die Spezifikation durchgeführt und dargestellt. Anschließend wird auf die technischen Grundlagen eingegangen, in denen die verwendeten Technologien und auf die umzusetzenden Ziele vorgestellt werden, damit verstanden werden kann was getan wurde. Darauf aufbauend wird die Implementierung der zuvor erklärten Ziele erläutert und es wird auf schwierige Stellen in der Umsetzung eingegangen. Abgeschlossen wird diese Arbeit mit einem Fazit und einem kurzen Ausblick, in dem der weitere Werdegang des Projektes geschildert wird.

## **1.3 Ziel der Arbeit**

Die Arbeit hat als Ziel eine App zu gestalten, welche es Kindern spielerisch ermöglicht, Lernfortschritte zu erzielen. Die App wird dabei für Smartphones mit dem Betriebssystem Android optimiert.

Das Ziel dieser Arbeit ist es dabei eine Dokumentation über die Vorgehensweise zu liefern, sowie eine Dokumentation, mit deren Hilfe andere arbeiten können.

## 2 NoRPG

Bei NoRPG handelt es sich um ein Spiel, welches an ein Role Player Game (RPG) erinnern soll. Bei RPGs spielt der Spieler einen Charakter in einer fiktiven Welt. Dabei gibt es eine Story die während des Spielens erlebt wird. Damit diese Geschichte vorran geht, muss der Spieler verschiedene Quests, dabei handelt es sich um verschiedenste Aufgaben, erledigen, um einen Fortschritt im Spiel zu erlangen. Darüber hinaus sammelt der Spieler Objekte in der Welt die er im Spiel nutzen kann. Ein Beispiel für solche Spiele ist das Spiel The Witcher 3 welches auf diesen Prinzipien aufbaut.

Darüber hinaus gibt es noch sogenannte MMORPGs, Massive Multiplayer Online Role Player Game. Dabei Gibt es wie bei RPGs eine Story und Quests allerdings kann man auch auf andere Spieler treffen und mit diesen zusammen spielen. Das wohl berühmteste MMORPG ist dabei World of Warcraft vom entwickler Blizzard. Hier kann der Spieler zwischen verschiedenen Klassen einen Charakter auswählen, von Menschenähnlichen bis zu skurilen Gestalten. Jede dieser Klassen hat verschiedene Fähigkeiten welche sich auf den Spielverlauf auswirken. Darüber hinaus gibt es in MMORPGs verschiedene Events an denen die Spieler gemeinsam versuchen eine Quest zu erfüllen. Dieser Multiplayermodus grenzt die MMORPGs von den RPG ab.

NoRPG hat keine Multiplayer möglichkeiten und ist deswegen nur ein RPG. Allerdings muss der Spieler keine Quests erfüllen wie sie aus anderen Spielen bekannt sind. Der Nutzer muss andere Spiele spielen, damit die Geschichte im Spiel vorran geht. Deshalb wurde sich für den Namen NoRPG entschieden, da es sich bei dem Spiel um ein RPG handelt, aber nicht alle klassischen Eigenschaften eines RPGs besitzt.

NoRPG ist ein Spiel welches versucht Bildung für jeden erreichbar zu machen. Dieses Ziel ist dabei in den Global Goals definiert. Dabei handelt es sich um 17 Ziele welche bis 2030 Umgesetzt werden sollen, um das Leben für alle Menschen auf der Welt zu verbessern. 2015 haben 193 Weltführer diese Unterzeichnet und begonnen diese Umzusetzen. Dabei sind diese Ziele umfangreich und reichen von einem besseren Umgang mit den uns zur Verfügung stehenden Ressourcen bis hin zu qualitativ hochwertiger Bildung für jeden und kostenlos.

NoRPG unterstützt dabei das Ziel Hochwertige Bildung für jeden Zugänglich zu machen, da die App für jeden frei zugänglich ist. Dieses Ziel hat weitere Unterziele, wobei nun kurz auf die für NoRPG relevanten Unterziele eingegangen wird.

- Bis 2030 sicherstellen, dass alle Mädchen und Jungen gleichberechtigt eine kostenlose und hochwertige Grund- und Sekundarschulbildung abschließen, die zu brauchbaren und effektiven Lernergebnissen führt.

Dieses Unterziel wird in NoRPG dahingehend unterstützt, dass die Common Core State Standards implementiert. Dies sind standardisierte Standards für Unterrichtsfächer und beschreiben den zu erlernenden Inhalt für Kinder in den verschiedenen Klassen. Da NoRPG für alle kostenfrei zugänglich ist und Mädchen und Jungen gleichberechtigt sind werden auch diese zwei Aspekte des Unterziels unterstützt. Die Standards werden dabei von den zur Verfügung gestellten Spielen erfüllt.

- Aufbau und Weiterentwicklung von Bildungseinrichtungen, die kinder- und behindertengerecht und geschlechtsspezifisch sind und für alle eine sichere, gewaltfreie, integrative und effektive Lernumgebung bieten

Da NoRPG keine Bildungseinrichtung ist wird dieses Unterziel nur bedingt erfüllt. NoRPG bietet Kindern jedoch eine sichere, gewaltfreie, integrative und effektive Lernumgebung, wodurch dieses Ziel zum Teil erfüllt wird. Darüber hinaus ist diese Umgebung für Kinder ausgelegt und somit kindergerecht. Geschlechtsspezifisch ist das Spiel nur dahingehend, dass die Kinder zu begin einen Charaktere in ihrem Geschlecht und Alter auswählen können.

<http://www.globalgoals.org/de/global-goals/quality-education/>

## 2.1 Gedanken

Dieses Ziel der Global Goals in der App wird dabei durch Gamification umgesetzt. Bei Gamification oder Gamifizierung handelt es sich um das nutzen von Spielespezifischen Eigenschaften, welche außerhalb von Spielen verwendet werden. Ein Beispiel dafür ist zum Beispiel PayBack mit Ihren PayBack Punkten. Dabei sammeln die Nutzer bei jedem Einkauf Punkte. Diese Punkte können sie dann durch Prämien eintauschen. In Videospielen sammeln die Spieler zum Beispiel Münzen um diese anschließend einzutauschen.

Gamifikation setzt darüber hinaus auch weitere erfolgreiche Prinzipien aus Videospielen um, damit die Nutzer Spaß daran haben, das Produkt zu nutzen.

GAMIFICATION – was ist gamification wieso wird das gemacht, hier rein passt nicht bei SRS

## **2.2 Story**

In NoRPG spielt der Spieler einen Charaktere der zu Beginn der Geschichte in einem Dorf wohnt. Von diesem können die Einwohner durch Portale in 5 andere Welten gehen, in denen andere Menschen leben und andere Lebensbedingungen herrschen. Eines Tages wurde dieses Dorf von einem Bösewicht heimgesucht und dieser hat die komplette Welt, in der der Spieler lebt, farblos gemacht. Zusammen mit den Farben wurden die Emotionen aus dem Dorf genommen. Nun hat sich der Hauptcharaktere das Ziel gesetzt, die Farben und die Emotionen der Einwohner zurück zu bringen, damit wieder alles wie vor dem Angriff ist. Dazu muss der Spieler in die verschiedenen Welten gehen und verschiedene Aufgaben erfüllen.

In Welt 1, einer Waltwelt, muss der Spieler verschiedene Edelsteine sammeln und Aufgaben erfüllen, um die ersten Farben zurück zu erlangen. Sobald dies geschehen ist kommt die Farbe Grün in das Dorf des Spielers zurück. Nun muss der Spieler in die anderen Welten gehen und die Farbe zurück holen. Über dem Dorf wird sein aktueller Fortschritt als Regenbogen dargestellt, welcher umso bunter und voller wird, umso mehr Edelsteine gefunden und Aufgaben abgeschlossen wurden. Sobald der Spieler alle Edelsteine gefunden und alle Aufgaben abgeschlossen hat, hat er die Welt gerettet und es kommt eine Endsequenz.

## **2.3 Abgrenzung zu MOOC**

NoRPG ist allerdings kein Massive Open Online Course (MOOC) sondern baut nur auf einem auf. Die Daten dieses MOOCs werden in NoRPG zur Verfügung gestellt, damit die Kinder einen Anreiz haben, diesen zu nutzen. Der eigentliche MOOC ist dabei Hone. Dabei handelt es sich um eine Lernplattform, welche es für Kinder ermöglicht, grundlegendes Verständnis, welches in der Grundschule vermittelt wird, zu erlangen. Dabei stellt Hone den Kindern eine Liste von Spielen bereit. Durch spielen dieser Spiele lernen die Kinder die Dinge die sie auch in der Grundschule lernen würden. Dabei sind diese Lerninhalte in den CCSS definiert und aufgliedert.

## 3 Software Requirements Specification

Das Software Requirements Specification, kurz SRS, ist ein veröffentlichter Standard zur Spezifikation einer Software. Der Inhalt eines SRS ist vom Institute of Electrical and Electronics Engineers im Standard IEEE 830-1998 festgehalten.

Die Struktur des SRS entspricht der Struktur, die im Standard beschrieben wird. Einige Kapitel des SRS werden nicht auftauchen und nicht erwähnt, da sie keine Relevanz für NoRPG haben oder an einer anderen Stelle in diesem Dokument haben.

### 3.1 Einführung

Das erste Kapitel des SRS enthält eine Beschreibung und eine Übersicht über alles, was im SRS enthalten ist.

#### 3.1.1 Zweck

Das SRS beschreibt den kompletten Projektumfang und die Anforderungen an die Software NoRPG. Es illustriert den Zweck und die vollständige Erklärung für die Entwicklung der Software. Dabei werden Systemeinschränkungen, Schnittstellen und Interaktionen mit externen Schnittstellen thematisiert<sup>1</sup>.

Die Zielgruppe des SRS sind zunächst alle, die in irgendeiner Verbindung mit NoRPG stehen oder jene, die Interesse an der Umsetzung haben. Zudem dient die Spezifikation zur Kommunikation zwischen den Stakeholdern und den Entwicklern.

#### 3.1.2 Umfang

Dieses SRS handelt von der in Kapitel 2 beschriebenen Software NoRPG.

---

<sup>1</sup> vgl. Tripp [1](1998) Seite 3

## 3.2 Allgemeine Beschreibung

Im zweiten Kapitel des SRS werden die allgemeinen Faktoren, die das Produkt und seine Anforderungen betreffen, beschrieben. Dieses Kapitel bietet einen Überblick über die Systemfunktionalitäten und stellt verschiedene Arten von Stakeholdern und deren Interaktion mit dem System vor. Dieses Kapitel behandelt nicht die spezifischen Anforderungen sondern stellt den Hintergrund für diese Anforderungen dar.

### 3.2.1 Produktperspektive

Bei NoRPG handelt es sich zunächst um ein selbstständiges abgeschlossenes Projekt, denn es bildet den Anfang für die Entwicklung eines ganzen Bildungssystems. Daher ist es besonders wichtig, das Produkt in unterschiedlichen Perspektiven mit verwandten und geplanten Produkten zu betrachten.

Es wird zwischen System-, Benutzer-, Hardware- und Softwareschnittstellen unterschieden. Neben den hier vier genannten gibt es weitere Schnittstellentypen, welche jedoch nicht für dieses Projekt in Betracht gezogen werden.

Jede Systemschnittstelle beschreibt und identifiziert die Funktionalität der Software, um ... to accomplish the system requirement and the interface description to match the system. Dieses Projekt hat in diesem Stadium nicht vor, mit anderen Systemen zu interagieren. Jedoch ist dies nicht hundertprozentig abstreitbar und wird deswegen hier erwähnt.

Die Benutzerschnittstellen beschreiben alle Schnittstellen zwischen NoRPG und seinen Nutzern. Nutzer bzw. Spieler können mit NoRPG nur über die Benutzeroberfläche, also über die Graphical User Interface GUI der App, interagieren. Dazu zählen alle GUI Elemente von der App.

Jede Schnittstelle zwischen NoRPG und Hardwarekomponenten des Systems werden als Hardwareschnittstellen bezeichnet. Dazu gehören auch Konfigurationsmerkmale. Das Smartphone und seine Komponenten ist größte Hardwareschnittstelle. Zum Smartphone gehört das Touchscreen, die Lautsprecher oder der WiFi-Adapter.

Softwareschnittstellen bilden den Übergang zwischen unterschiedlichen Programmen und ermöglichen dadurch den Datenaustausch zwischen diesen Programmen. Die bekannteste Softwareschnittstelle sind API's. API ist eine Programmierschnittstelle. NoRPG wird Schnittstellen mit Android, Google Play Store, Datenbank.

### 3.2.2 Produktfunktionen

In diesem Unterkapitel werden die wichtigsten Funktionen von NoRPG zusammengefasst.

Hauptzweck und Hauptfunktion von NoRPG: Lernspiele zum Herunterladen anzeigen. Diese Spiele sind in einer festgelegten Reihenfolge verfügbar. Die Reihenfolge und der Grund dafür wird im nächsten Kapitel beschrieben

Spieler kann einmalig bei der Registrierung seinen eigenen Charakter erstellen

Nach erfolgreicher Anmeldung kann er seinen Charakter durch die Spielwelt steuern, mit Elementen im Spiel interagieren.

- Die Interaktion mit NPC's starten eine Unterhaltung, wodurch der Spieler mehr über das Spiel erfahren kann.
- Die Interaktion mit einer Verschlussenen Truhe zeigt die Lernspiele, die gespielt werden müssen um die Truhe öffnen zu können.
- Die Interaktion mit einer freigespielten Truhe, ermöglicht dem Spieler die Truhe zu öffnen.
- Die Interaktion mit Collectables, die durch das Suchen/Reisen gefunden werden können, sammelt diese.
- Die Interaktion mit der Umwelt (Tiere, Bäume, etc.) startet keine Aktion.

Menü mit mehreren Funktionen: Neben kleineren Funktionen wie Spieloptionen gibt es die Möglichkeit den Lernfortschritt zu betrachten, etc.

### 3.2.3 Benutzermerkmale

In diesem Projekt wird zwischen zwei Benutzergruppen unterschieden.

Die erste Benutzergruppe sind die Spieler, die User bzw. die Kinder. Grundsätzlich richtet sich NoRPG an Kinder, die keine Möglichkeit haben eine Schule zu besuchen. Jedoch werden keine Benutzergruppen für diese App ausgeschlossen. Egal ob jung oder alt sowie männlich oder weiblich. Der Spieler sollte Erfahrung mit der Verwendung eines Smartphones, insbesondere eines Android-Systems, haben. Dazu zählt die Bedienung der Android-Oberfläche und die des Google Play Stores, da die. Zudem sollte der Spieler die englische Sprache verstehen können, da NoRPG zunächst nur in Englisch erscheinen wird. Jedoch wird das Spiel in jedem Land verfügbar sein, denn bei den vermittelten Standards handelt es sich um international geltende.

Bei der Implementierung ist es wichtig, dass die englischen Texte einfach zu lesen und zu verstehen sind. Denn bei den meisten Benutzern handelt es sich um Kinder zwischen der ersten und fünften Klasse. Zum Voranschreiten ist es notwendig, dass der Spieler den Inhalt von NoRPG versteht. Aus diesem Grund ist eine einfache Sprache ohne Fachjargon notwendig.

Die zweite Benutzergruppe sind die Administratoren. Administratoren wollen Daten analysieren und daraus Aktionen ableiten. Welche Daten wie gespeichert werden, werden in den spezifischen Anforderungen genauer definiert.

### 3.2.4 Einschränkungen

Bei den Einschränkungen wird zwischen Einschränkungen der Entwickler und Einschränkungen der Spieler unterschieden.

Entwickler müssen sich an regulatorische Richtlinien, wie beispielsweise die Datenschutzerklärung von Google oder das IT-Sicherheitsgesetz, halten.

Da NoRPG sich an Kinder in bildungsfernen Ländern richtet, darf NoRPG keine hohen Hardwareressourcen anfordern. Als Maßstab wird das Smartphone Samsung Galaxy S4 genommen. Das Samsung Galaxy S4 kostet 250<sup>2</sup> und hat einen Quadcore?? mit 1,6GHz und 2GB RAM sowie 16GB internen Speicher. Es ist die Android Version 5.0 Lollipop standardmäßig von Samsung installiert.

Schnittstellen anbieten, wenn eine Webapplikation für Administratoren entwickelt wird um Daten besser auszuwerten. Oder Schnittstellen für Spielentwickler. (Bestätigen, dass das Kind den Kurs erfolgreich abgeschlossen hat)

Verfügbar- und Zuverlässigkeit: Spiel muss Offline spiel und -startbar sein. Das Spiel muss sicherstellen, dass ein Kurs vollständig absolviert wird, damit keine verfälschten Daten gesendet werden.

Daten der Spieler werden anonymisiert gespeichert um anschließend Analysen zu machen, um rauszufinden in welchen Ländern, Stadtteilen oder Gegenden die meisten Kinder aktiv sind. Daher sollten nur relevante und notwendige Informationen gespeichert werden um diese dann analysieren zu können.

Internet zum Downloaden von Spielen, von NoRPG, installieren von Updates, Synchronisieren, Registrieren, Einloggen, Ausloggen (da Synchronisiert wird)

Genügend Speicherplatz, RAM und andere Ressourcen

---

<sup>2</sup> Stand 04.01.2017 Quelle: sdaqdsa



### **3.2.5 Annahmen und Abhängigkeiten**

Eine Annahme von NoRPG ist, dass es immer auf Smartphones, die genügend Leistung haben, verwendet wird. Wenn das Telefon nicht über genügend Hardwareressourcen für die Anwendung verfügt, kann es Szenarien geben, in denen die Anwendung nicht wie beabsichtigt oder überhaupt nicht funktioniert.

Eine weitere Annahme ist, dass das Smartphone und dessen Hardware sowie Software funktionieren. Das Smartphone muss sich mit dem Internet verbinden können, wenn der Benutzer sich anmelden möchte oder Lernspiele herunterladen will. Neben einer funktionierenden Internetverbindung sollten andere Hardwareelemente wie die Lautsprecher oder der Touchscreen funktionieren. Das Smartphone muss eine gültige Android Version mit einem Google Konto besitzen.

### **3.2.6 Aufteilung der Anforderungen**

In dem Fall, dass das Projekt verzögert wird, gibt es einige Anforderungen, die auf die nächste Version der Anwendung übertragen werden könnten.

## **3.3 Spezifische Anforderungen**

Das letzte Kapitel des SRS dient dazu alle Softwareanforderungen detailliert zu beschreiben. Dies ermöglicht es Designern ein System zu entwickeln, welches allen Anforderungen entspricht, und Testern das System ausreichend zu testen.

Das letzten Kapitel des SRS enthält ausführlich die Anforderungsspezifikation und eine Beschreibung der unterschiedlichen Systemschnittstellen. Es werden verschiedene Spezifikationstechniken verwendet, um die Anforderungen für unterschiedliche Zielgruppen genau festzulegen.

### **3.3.1 Externe Schnittstellen**

Dieser Abschnitt ist die detaillierte Beschreibung aller Ein- und Ausgänge von NoRPG. Die Beschreibung ergänzt und vervollständigt die Schnittstellenbeschreibung von Kapitel 2.2.1.

## Benutzerschnittstellen

Die Benutzerschnittstellen, die user interfaces, sind der Punkt, an dem der Benutzer mit der Software interagiert. Zur Beschreibung der Benutzerschnittstellen werden logische Eigenschaften sowie Aspekte zur Optimierung formuliert.

Einem Benutzer, der NoRPG zum ersten Mal startet oder der nicht angemeldet ist, wird der Login-Screen präsentiert. Auf dem Login-Screen hat der Benutzer die Möglichkeit sich mit seinem Benutzernamen und seinem Passwort anzumelden oder sich, falls noch nicht geschehen, bei NoRPG zu registrieren. Das Smartphone muss Quer gehalten werden, da alle Elemente des Bildschirms vertikal angeordnet sind. Diese Eigenschaft trifft auch auf alle anderen Benutzerschnittstellen zu. Das Layout des Login-Screens ist ein Border-Pane, in dem die Bestandteile in einer einzigen Spalte angeordnet sind. Zur Optimierung der Nutzung werden kurze Fehlermeldungen ausgegeben, wenn der Benutzer falsche Login-Daten eingibt.

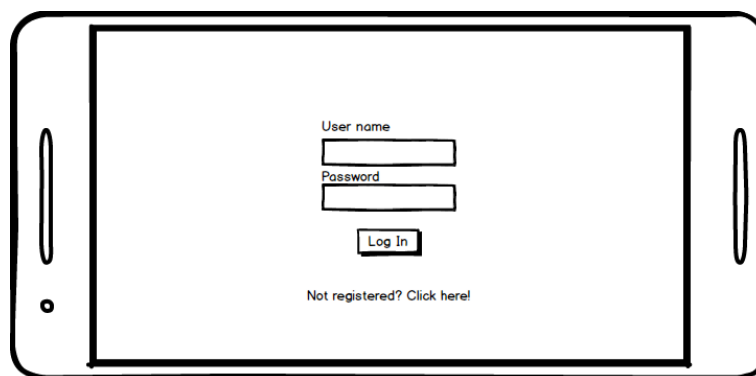


Abbildung 1: Login-Screen Mockup

Falls sich der Benutzer bei NoRPG registrieren möchte, hat er die Möglichkeit dies in der App zu machen. Dazu klickt der Benutzer im Login-Screen auf den Register-Button. Anschließend öffnet sich der Register-Screen. Die Elemente sind im Tabellen Layout angeordnet, wodurch der Benutzer weiß, welche Daten in welches Feld eingetragen werden müssen. Die Registrierung ist notwendig, damit der Spielstand, somit der Fortschritt in einer Relation mit dem Benutzer steht. Zur weiteren Optimierung werden kurze Fehlermeldungen ausgegeben, damit der Benutzer weiß, in welchem Feld ein Fehler ist.

Abbildung 2: Register-Screen Mockup

Nach der Registrierung kann der Spieler einmalig seinen Charakter für den angelegten Account erstellen. Dazu bestimmt der Benutzer den Namen, das Geschlecht und das Aussehen des Charakters.

Abbildung 3: Register-Screen Mockup

NoRPG startet, nachdem alles geladen wurde und der Benutzer angemeldet ist. Das Spiele-Screen besteht aus der Spielewelt (Grafik) und dem Head-Up Display, kurz HUD. Das HUD ist eine Methode, mit der Informationen visuell als Teil der Benutzeroberfläche eines Spiels vermittelt werden. Während die Informationen, die auf dem HUD angezeigt werden, stark vom Spiel abhängen, gibt es viele Eigenschaften, die Spieler über viele Spiele erkennen. Die meisten von ihnen sind statisch auf dem Bildschirm, so dass sie während des Spiels sichtbar bleiben.

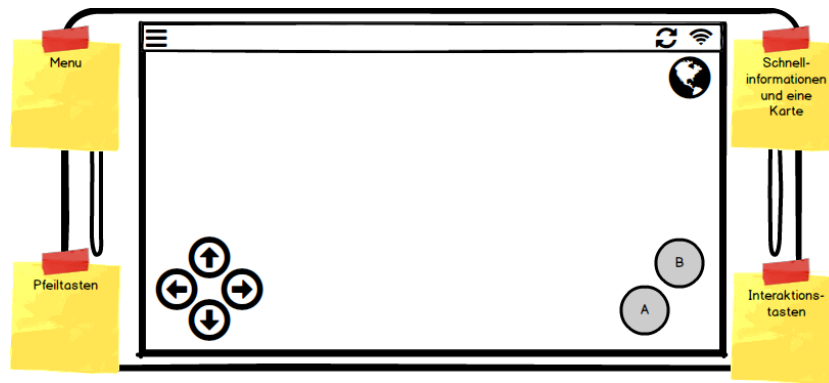


Abbildung 4: HUD Mockup

Das Mockup 3 enthält alle direkt sichtbaren HUD Elemente, die während des Spieles aktiv sind. Die Elemente sind an die Ecken gebunden, so befinden sich beispielsweise die Pfeiltasten zur Bewegung des Charakters in der linken unteren Ecke des Bildschirms (siehe Graifk). Es sind so wenig Elemente wie möglich auf dem Bildschirm angeordnet und die verwendeten Symbole sind aus anderen bekannten Spielen und Konsolen übernommen und sind quasi ein Standard. Durch diese bekannte Anordnung der Elemente kann der User Informationen schneller verstehen und schneller reagieren.

Das Menü, welches sich in der oberen linken Ecke befindet, kann geöffnet werden. Dadurch wird das laufende Spiel pausiert und es werden weitere Optionen bzw. Interaktionen mit dem Spiel möglich. Diese HUD Elemente werden nur dann sichtbar, wenn der Spieler das Menü öffnet. Dadurch rückt das Spiel und die anderen Elemente in den Hintergrund. Das bedeutet nicht, dass die Elemente ausgeblendet werden, sondern dass der Benutzer diese Elemente nicht benutzen kann solange das Menü offen ist.

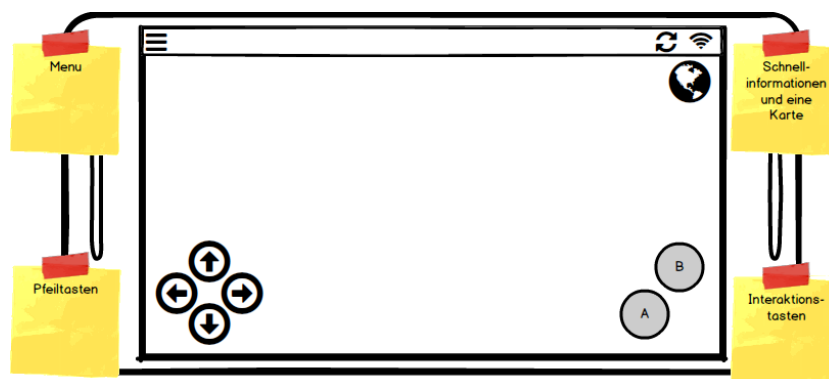


Abbildung 5: HUD offenes Menü Mockup

Einige der Menü-Elemente öffnen wiederum einen anderen Screen. Diese werden dann über das aktuelle Spiel geöffnet. Das Spiel befindet sich im Hintergrund und kann nicht gesehen bzw. angeklickt werden. Der neu geöffnete Screen muss erst geschlossen werden um das Spiel fortsetzen zu können. Ein Beispiel dafür ist der Fortschritt-Screen. Hier kann der Benutzer seinen Lern- bzw. Spielfortschritt betrachten.

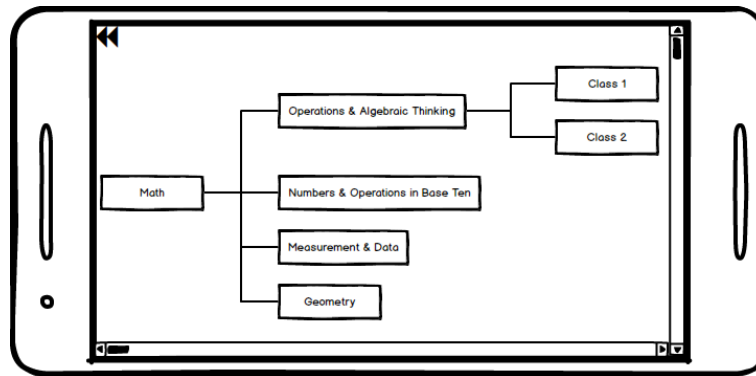


Abbildung 6: Fortschritt-Screen Mockup

Die Auflistung aller Screens würde den Rahmen dieser Arbeit überschreiten. Daher befinden sich die Mockups für die restlichen Screens im Anhang.

## Hardwareschnittstellen

Die Hardwarechnittstellen spezifiziert die logischen Eigenschaften jeder Schnittstelle zwischen NoRPG und den Hardwarekomponenten des Systems.

NoRPG besitzt keine direkten Hardwarechnittstellen: Verbunden mit Lautsprechern für Soundausgaben, Touchscreen für Eingaben/Interaktionen, WLAN für die Verbindung mit dem Server/Synchronisierung, Verbindung zum Datenbank Server

## Softwareschnittstellen

This should specify the use of other required software products and interfaces with other application systems

Data mangement system, Android OS

### 3.3.2 Funktionale Anforderungen

Use Cases dokumentieren Funktionalitäten eines Systems auf Basis von einfachen Modellen. In einem Use Case wird das nach außen sichtbare Verhalten eines Systems aus der Sicht der Nutzer beschrieben. Ein Nutzer kann hierbei eine Person, eine Rolle oder ein anderes System sein. Dieser Nutzer tritt als Akteur mit dem System in Interaktion, um ein bestimmtes Ziel zu erreichen.

Use Cases verwenden Activity UML um anzuzeigen, wie der Benutzer vorgehen muss. Ein Aktivitätsdiagramm ist ein Verhaltensdiagramm der Unified Modeling Language (UML) und stellt die Vernetzung von elementaren Aktionen und deren Verbindungen mit Kontroll- und Datenflüssen grafisch dar.

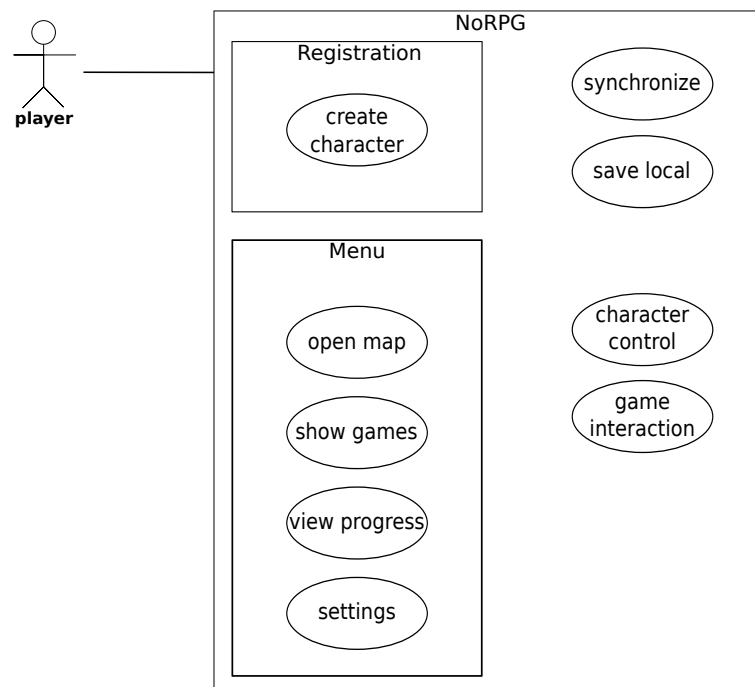


Abbildung 7: Overall Use Case Diagramm

Das abgebildete System stellt die zu entwickelnde App für die User dar. Die App stellt die Graphische Oberfläche und somit die beschriebenen Benutzerschnittstellen dar. Es sind nur die Funktionalitäten enthalten, die der Benutzer ausführen kann, also jene die über die Benutzerschnittstellen angesprochen werden können. Use Cases wie Login oder Registrierung sind im Overall Use Case Diagramm nicht enthalten, da diese im Vergleich zu anderen Use Cases primitiv sind.

## Create character

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer seinen Charakter erstellen möchte. Dieser Use Case wird pro Account genau einmal nach der Registrierung ausgeführt und zählt noch zum Prozess der Registrierung.

Nach erfolgreicher Registrierung kann der Spieler seinen Charakter erstellen. Der User kann seinem Charakter einen Namen geben, das Geschlecht auswählen und anschließend das Aussehen bestimmen. Anschließend wird dem User der Hinweis angezeigt, dass es sich um eine einmalige Aktion handelt. Nachdem diese bestätigt wurde, wird der Charakter erstellt und gespeichert.

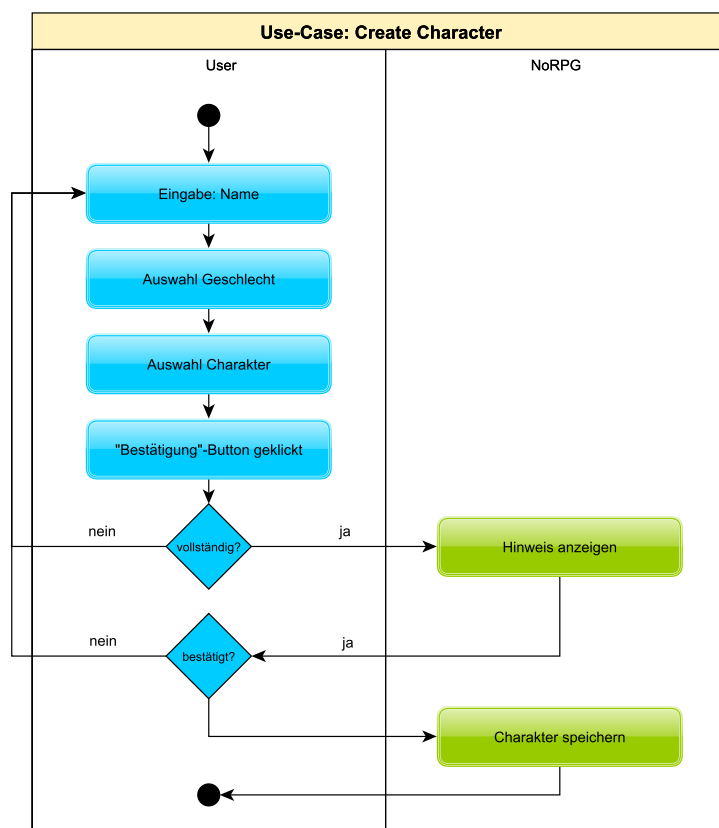


Abbildung 8: Create Character Activity UML

Bevor jedoch dieser Use Case ausgeführt werden kann, muss die Registrierung vollständig und erfolgreich abgeschlossen werden. Die Registrierung ist erfolgreich, wenn alle benötigte Daten eingetragen wurden und der Account noch nicht existiert. Für den gesamten Registrierungsprozess wird eine aktive Internetverbindung benötigt.

Nach erfolgreicher Erstellung des Charakters, wird dieser in die Datenbank gespeichert und der User kann sich nun anmelden und in die Rolle seines Charakters schlüpfen.

## **Open map**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer die Karte öffnet. Die Karte dient zur Orientierung der Welt und beinhaltet Symbole etc. um herauszufinden was so ist

Ereignisablauf: Benutzer öffnet Menü und klickt auf "Map"...

Vorbedingungen: Menü offen, Benutzer befindet sich nicht in einer NPC Interaktion

Nachbedingungen: Eine Karte von der aktuellen Welt wird geöffnet

## **Show games**

Dieser Use Case beschreibt den Anwendungsfall: Liste der gespielten und heruntergeladenen Spiele wird angezeigt. Zuordnung zu den Standards. Aus NoRPG das Spiel starten können.

Ereignisablauf: Benutzer öffnet Menü und klickt auf "Games"...

Vorbedingungen: Menü offen, Benutzer befindet sich nicht in einer NPC interaktion

Nachbedingungen: Eine Liste wird angezeigt

## **View progress**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

Ereignisablauf

Vorbedingungen

Nachbedingungen

## **Settings**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

Ereignisablauf

Vorbedingungen

Nachbedingungen



## **Synchronize**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

Ereignisablauf

Vorbedingungen

Nachbedingungen

## **Save local**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

Ereignisablauf

Vorbedingungen

Nachbedingungen

## **Character control**

Dieser Use Case beschreibt den Anwendungsfall: Benutzer interaktionen, wie Bewegen oder Bestätigen.

Ereignisablauf

Vorbedingungen: Spieler befindet sich im Spiel (nicht loading screen und menü ist geschlossen)

Nachbedingungen: Charakter bewegt sich, bestätigt oder lehnt ab

## **game interaction**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer sich in einer Interaktion mit einem NPC befindet. NPC bedeutet Non-Player Charakter und stellt die programmierten Charaktere dar (Unterhaltungen mit NPC, Storytelling)

Ereignisablauf

Vorbedingungen: Spieler befindet sich im Spiel (nicht loading screen und menü ist geschlossen)

Nachbedingungen: Charakter bewegt sich, bestätigt oder lehnt ab

### **3.3.3 Performanz Anforderungen**

This subsection should specify both the static and the dynamic numerical requirements placed on the software or on human interaction with the software as a whole. Static numerical requirements may include the following

The number of terminals to be supported, The number of simultaneous users to be supported, Amount and type of information to be handled

### **3.3.4 Datenbank Anforderungen**

This should specify the logical requirements for any information that is to be placed into a database. This may include the following:

Types of information used by various functions

Frequency of use

Accessing capabilities

Data entities and their relationships

Integrity constraints

Data retention requirements.

### **3.3.5 Entwurfsbeschränkungen**

This should specify design constraints that can be imposed by other standards, hardware limitations, etc.

Standards compliance: This subsection should specify the requirements derived from existing standards or regulations. They may include the following:

Report format, Data naming, Accounting procedures and Audit tracing.

### **3.3.6 Benutzerfreundlichkeit**

### **3.3.7 Zuverlässigkeit**

This should specify the factors required to establish the required reliability of the software system at time of delivery.

### **3.3.8 Verfügbarkeit**

This should specify the factors required to guarantee a defined availability level for the entire system such as checkpoint, recovery, and restart.

### **3.3.9 Sicherheit**

This should specify the factors that protect the software from accidental or malicious access, use, modification, destruction, or disclosure. Specific requirements in this area could include the need to

- Utilize certain cryptographic techniques;
- Keep specific log or history data sets;
- Assign certain functions to different modules;
- Restrict communications between some areas of the program;
- Check data integrity for critical variables.

### **3.3.10 Wartbarkeit**

This should specify attributes of software that relate to the ease of maintenance of the software itself. There may be some requirement for certain modularity, interfaces, complexity, etc. Requirements should not be placed here just because they are thought to be good design practices.

### **3.3.11 Portabilität**

This should specify attributes of software that relate to the ease of porting the software to other host machines and/or operating systems. This may include the following:

- Percentage of components with host-dependent code;
- Percentage of code that is host dependent;
- Use of a proven portable language;
- Use of a particular compiler or language subset;
- Use of a particular operating system.

## 4 Technische Grundlagen

Nachstehend werden die technischen Grundlagen erläutert und es wird kurz auf diese eingegangen. Dabei wird mit den Entwicklungsumgebungen begonnen.

### 4.1 Unity3D

Unity3D ist eine Entwicklungs- und Laufzeitumgebung, die viel in der Spieleindustrie verwendet wird, aber auch in anderen Industrien immer mehr zum Einsatz kommt. Mit diesem Programm können 3D Anwendungen erstellt werden, aber auch 2D oder 2,5D Anwendungen sind möglich. Diese Anwendungen können dank Unity3D plattformübergreifend entwickelt werden. Die Entwicklungsumgebung ist dabei an einen 3D-Editor angelegt. Das User Interface (UI) besteht dabei aus verschiedenen Fenstern, um verschiedene Einstellungen der Szenen in Unity3D zu bearbeiten. Eines dieser Standardfenster ist das „Scene“ Fenster, das in der Mitte von Abbildung ?? zu sehen ist. In diesem Fenster ist immer die aktuelle Szene dargestellt. Außerdem können Nutzer mit den Objekten aus der Szene interagieren und sie verändern, um sie beispielsweise in der Szene neu platzieren. Sobald ein Objekt in der Szene ausgewählt wurde, öffnen sich in dem Fenster „Inspector“ weitere Einstellmöglichkeiten. Diese Einstellmöglichkeiten variieren je nach gewähltem Objekt. Hier können den Objekten zusätzliche Eigenschaften zugewiesen werden, um so ihr Verhalten nochmals zu ändern. Dabei können die visuellen als auch die physischen Eigenschaften der Objekte verändert werden.

Durch das Markieren in dem „Hierarchy“ Fenster werden Objekte zusätzlich ausgewählt und es können Eigenschaften zugewiesen werden. Dort werden alle Objekte aus der Szene in einer Hierarchie angezeigt. Dabei sind die Objekte auf unterschiedlichen Ebenen dargestellt, um zu zeigen, welches Objekt mit anderen Objekten verknüpft ist. Das Fenster „Project“ dient dazu, alle Dateien, die in dem Projekt vorhanden sind, anzuzeigen. In diesem Fenster können Ordner, sowie andere Dateien erstellt werden. Oben, in der Mitte der Benutzeroberfläche, befinden sich zudem jeweils ein Start, Pause und Vorlauf Button. Diese Buttons dienen dazu, das Programm im Vorfeld zu testen

und in der Entwicklungsumgebung zu rendern. Der Code wird von Unity3D JustIn-Time (JIT) kompiliert, und anschließend auf Mono oder dem Microsoft .NET Framework ausgeführt. Der Code steht in sogenannten Skripten, die in C, UnitySkript (ähnlich JavaScript) oder Boo geschrieben sind.

Für den Fall, dass Fehler während des Kompilierens oder während der Laufzeit auftreten, gibt es ein „Console“ Fenster. In diesem Fenster können aktuelle Fehlermeldungen ausgegeben werden. Zudem werden hier auch gezielte Meldungen, die in den Skripten programmiert wurden, eingeblendet. Um diesem Fall entgegenzuwirken, gibt es in Unity Tests, die die Szenen auf ihre Korrektheit testen. Die Integrationstests simulieren eine Szene. So können die verschiedenen Objekte auf ihre Eigenschaften geprüft werden. Bei den zuvor in Kapitel 2.1 genannten Prefabs handelt es sich um vorgefertigte Objekte, die in einer Szene verwendet werden können. Dabei sind dies meist Objekte, die mehrmals in einem Projekt verwendet werden können. Bei Skripten handelt es sich wiederum um die Logik, die ein Objekt hat. Kann ein Objekt beispielsweise seine Farbe ändern, wenn ein Nutzer mit diesem Objekt interagiert, steht die Logik dafür in einem Skript. Um diese Logik zu bearbeiten und anzupassen, wird eine geeignete Entwicklungsumgebung benötigt. Bei der Installation von Unity3D ist eine Version von Microsoft Visual Studio enthalten, die zum Bearbeiten von Skripten dient.

## **4.2 Visual Studio**

Wie kann es genutzt werden und wozu? Was ist das? Alternativen?

## **4.3 C Sharp**

Was ist das? Vergleich zu C? Unterschiede? Vergleich zu Java? Unterschiede ? Vor/-Nachteile? Allgemeiner Aufbau einer Klasse?

## **4.4 SQL**

→ Für die Datenbankverbindung innerhalb der App wichtig, was ist das? Wozu wird es benötigt?

## **5 Umsetzung**

Nachdem im vorherigen Kapitel die technischen Grundlagen erläutert wurden, wird jetzt auf die Umsetzung eingegangen. Dabei wird diese in drei Teile unterteilt

### **5.1 App**

### **5.2 Datenbank auf dem Handy**

### **5.3 Datenbank auf dem Server**

## **6 Fazit und Ausblick**

### **6.1 Fazit**

### **6.2 Ausblick**

Webfrontend ...

Eigene Spiele zur Überprüfung, ob der Spieler den Standard erfüllt hat...

weitere Klassen und weiter Fächer, dementsprechend auch neue Spielwelten

Ranking/Multiplayer/...

# Literaturverzeichnis

- [1] Leonard L. Tripp et al. *IEEE Recommended Practice for Software Requirements Specifications*. Techn. Ber. ISBN: 0-7381-0448-5. The Institute of Electrical und Electronics Engineers, 1998.
- [2] Anita Rudolf Siegbert A. Warwitz. *Vom Sinn des Spielens: Reflexionen und Spielideen*. 4. Auflage. Schneider Hohengehren, 2016. ISBN: 978-3834016645.



# Anhang

## Anhang A