

**Titel**

**STUDIENARBEIT**

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

**Mehmet Ali Incekara & Tom Wolske**

Abgabedatum 8. November 2016

<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 8. November 2016

Ort, Datum

\_\_\_\_\_  
Tom Wolske

Karlsruhe, den 8. November 2016

Ort, Datum

\_\_\_\_\_  
Mehmet Ali Incekara

# Inhaltsverzeichnis

o

r brzungsverzeichnisIVsection\*.3

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Aufbau der Arbeit . . . . .	2
1.3	Ziel der Arbeit . . . . .	2
<b>2</b>	<b>Software Requirements Specification</b>	<b>3</b>
2.1	Allgemeine Beschreibung . . . . .	3
2.1.1	Benutzermerkmale . . . . .	3
2.1.2	Beschrungen . . . . .	4
2.2	Funktionalitn . . . . .	4
2.2.1	Login . . . . .	4
2.2.2	Create character . . . . .	5
2.2.3	Player interaction . . . . .	6
2.2.4	NPC interaction . . . . .	6
2.2.5	Choose games . . . . .	7
2.2.6	Open map . . . . .	7
2.2.7	Show games . . . . .	8
2.2.8	Show colors . . . . .	8
2.2.9	View progress . . . . .	8
2.2.10	Settings . . . . .	9
2.2.11	Synchronize . . . . .	9
2.2.12	Save local . . . . .	9
2.3	Benutzbarkeit . . . . .	10
2.3.1	Anlernzeit . . . . .	10
2.3.2	Hardware Anforderungen . . . . .	10
2.3.3	Software Anforderungen . . . . .	10
2.4	Zuverligkeit . . . . .	10
2.4.1	gbarkeit10subsection.2.4.1	
2.4.2	Performanz . . . . .	10

2.5	Schnittstellen . . . . .	11
2.5.1	Benutzer Schnittstellen . . . . .	11
2.5.2	Hardware Schnittstellen . . . . .	11
2.5.3	Software Schnittstellen . . . . .	11
<b>3</b>	<b>NoRPG</b>	<b>12</b>
3.1	Gedanken . . . . .	12
3.2	Story . . . . .	12
3.3	Abgrenzung zu MOOC . . . . .	12
	<b>Anhang</b>	<b>VII</b>

# **Abkürzungsverzeichnis**

# Abbildungsverzeichnis

1	Overall Use Case Diagram . . . . .	4
---	------------------------------------	---

## 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 sind ein Bestandteil unserer Kultur schon seit tausenden Jahren. Das erste Spiel soll das Königliches Spiel von Ur gewesen sein, welches bereits 2600 vor Christus existierte. Spiele haben sich seit dem jedoch weiterentwickelt und dienen heutzutage 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 Videospiele 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 vermittelt 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. 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 die Spiele, gelernten Kompetenzen, bzw. können



sich die Kinder dort neue Spiele herunterladen, um weitere Kompetenzen zu erwerben. Diese Plattform ist allerdings nicht reizvoll für Kinder gestaltet und soll deshalb attraktiver für Kinder werden. Das wird durch die Entwicklung eines Spieles umgesetzt, in dem die Kinder ihre Fortschritte einsehen können und dabei spielen können. Durch die Umsetzung als App gelingt es zusätzlich den Kindern eine Offlineplattform zu geben, welche sie unabhängig von der Internetverbindung nutzen können.

## 1.2 Aufbau der Arbeit

Die Arbeit beginnt mit den Grundlegenden Themen die für das verständniss dieser Arbeit von nöten sind und um die verbindung der Altanwendung aufzuzeigen. In diesen Grundlagen wird auf die verwendeten Technologien und auf die umzusetzenden Ziele eingegangen, damit verstanden werden kann was getan wurde. Darauf aufbauend wird die Implementierung der zuvor erklärten Ziele erläutert und es wird auf schwirige 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 die Onlineplattform **Hone** für Kinder interessanter zu gestalten. Dabei wird dies durch eine App realisiert. Diese baut auf der Altanwendung auf und benutzt Schnittstellen zu dieser, um an die benötigten Daten zu gelangen. 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 Software Requirements Specification

Das Software Requirements Specification, kurz SRS, ist ein veröffentlichter Standard zur Spezifikation von Anforderungen einer Software. Das SRS beschreibt den Projektumfang und dient zur Kommunikation zwischen Stakeholders und Entwickler. Stakeholders können im SRS ihre Anforderungen für die Software beschreiben und die Entwickler können aus dem SRS Informationen für die Implementierung gewinnen.

Das SRS hat eine feste Struktur, jedoch muss die Spezifikation einer Software nicht alle Punkte beinhalten. Die folgenden Kapitel stellen die Kapitel des SRS dar.

### 2.1 Allgemeine Beschreibung

NoRPG ist eine spielbasierte Lernspielplattform für Android. NoRPG selbst ist wie ein Rollenspiel aufgebaut, jedoch stellt es kein Rollenspiel dar. In NoRPG werden unterschiedliche Lernspiele zum herunterladen angeboten.

NoRPG unterstützt die Global Goals, 4. Ziel Quality Education. Paar Worte dazu verlieren.

Weitere Informationen in Kapitel 4 über die Gedanken für die Story, Design etc.

#### 2.1.1 Benutzermerkmale

NoRPG richtet sich größtenteils an Kinder, denen die Möglichkeit in die Schule zu gehen nicht ermöglicht werden. Jedoch werden keine anderen Benutzergruppen ausgeschlossen.

Die Benutzer sollten Erfahrungen mit der Verwendung eines Smartphones, insbesondere mit Android-Systemen haben. Dazu zählt die Bedienung der Android-Oberfläche (Starten einer App, Herunterladen von Updates, Verwendung des Google Play Stores).

Dazu sollten die Benutzer englische Texte lesen können, für die Registrierung, Anmeldung und durchführen von NPC-Kommunikation zum verstehen des ablaufes der Plattform.

### 2.1.2 Beschränkungen

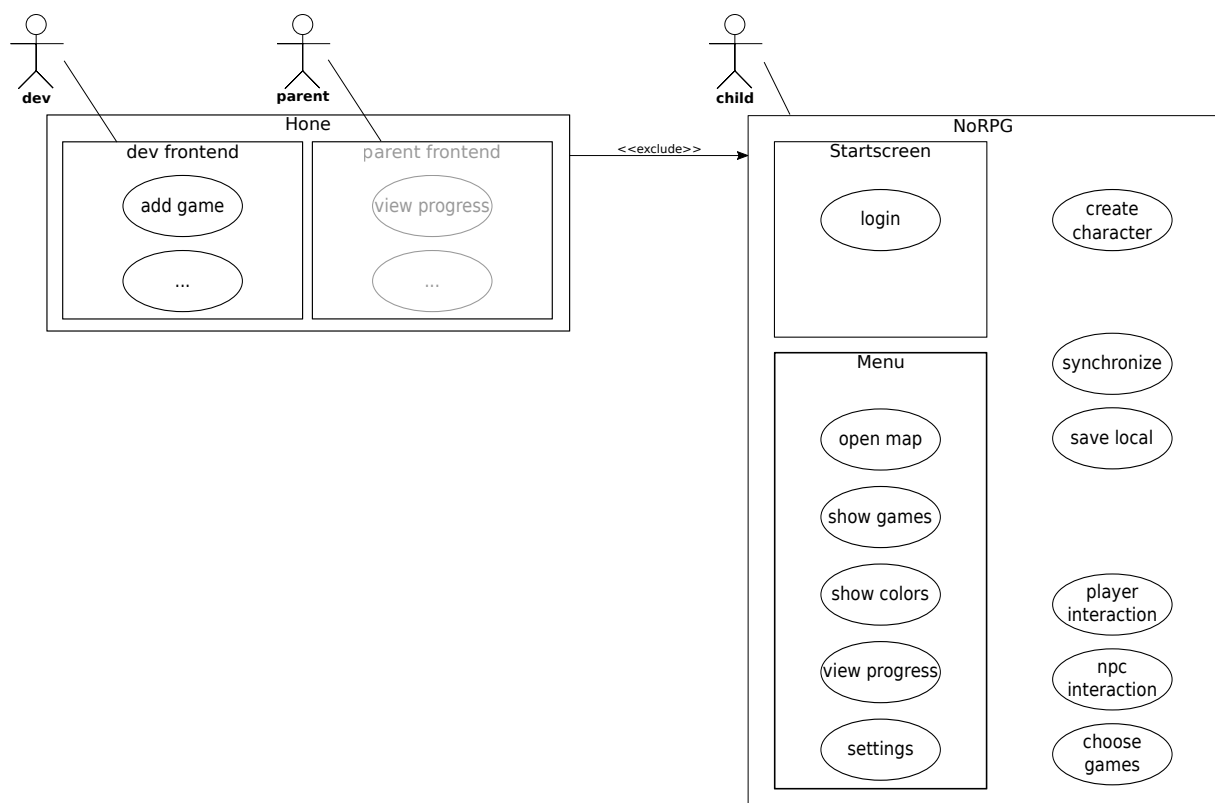
Zum Herunterladen von NoRPG und den angebotenen Lernspielen wird eine aktive Internetverbindung vorausgesetzt.

Zum Spielen wird keine durchgehende Internetverbindung benötigt.

Spiel zunächst nur in Englisch und Deutsch verfügbar. Jedoch bleibt es erweiterbar.

## 2.2 Funktionalitäten

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.



## Abbildung 1: Overall Use Case Diagramm

In der Grafik sind 2 Systeme zu sehen. Links das vorhandene System Hone welches ein Frontend für die Entwickler und für die Kinder darstellt. Die Kinder sollen nicht mehr über Hone die Spiele herunterladen sondern nur noch die App NoRPG verwenden. Die Ansicht wird jedoch weiterhin genutzt und soll den Eltern der Kinder die Möglichkeit geben, den Fortschritt des Kindes nachzuschauen. Hone soll von den Rollen Entwickler und Eltern entwickelt werden.

Das rechte System NoRPG stellt die zu entwickelnde App dar. Diese dient als Frontend für das Kind. Es gibt viele Use Cases. Die Use Cases werden in unterschiedliche Gruppen zusammengefasst. Die nächsten Unterkapitel sind die einzelnen Gruppierungen.

### 2.2.1 Login

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer sich bei NoRPG anmelden möchte. Eine Anmeldung ist notwendig um NoRPG zu starten.

#### Ereignisablauf

Eingeben von Benutzername und Passwort. Klicke auf Login.

Alternativer Ablauf: Abbruch

#### Vorbedingungen

Benutzer ist registriert, Während der Anmeldung ist eine Internetverbindung vorhanden, Kombination von Benutzername und Passwort existiert

#### Nachbedingungen

Benutzer angemeldet, kann online oder offline weiterspielen

### 2.2.2 Create character

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer seinen Charakter erstellen möchte.

## **Ereignisablauf**

Wenn der Benutzer sich zum ersten mal anmeldet hat er die Möglichkeit seinen Charakter zu erstellen. Dafür wählt der Benutzer sich zunächst sein Geschlecht aus und wählt anschließend den passenden Charakter.

Zum Abschluss vergibt der Benutzer seinen Charakter einen Namen.

## **Vorbedingungen**

Der Account meldet sich das erste mal in der App an.

## **Nachbedingungen**

Nach der Erstellung beginnt das Spiel und der Charakter ist gespeichert. Bei erneuter Anmeldung muss der Benutzer nicht erneut einen Charakter erstellen.

## **2.2.3 Player interaction**

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

## **Ereignisablauf**

Klickt auf Pfeiltasten, Charakter bewegt sich in diese Richtung

Klickt auf A, Charakter bestätigt

Klickt auf B, Charakter lehnt ab

(Bild Mockup)

## **Vorbedingungen**

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

## **Nachbedingungen**

Charakter bewegt sich, bestätigt oder lehnt ab

## **2.2.4 NPC 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**

Ereignisablauf etc.

### **Vorbedingungen**

Ingame, nicht loading screen oder menü offen

### **Nachbedingungen**

Unterhaltung findet statt, etc.

## **2.2.5 Choose games**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer ein spiel zum downloaden auswählt

### **Ereignisablauf**

Der Benutzer kann sich (wenn vorhanden) zwischen mehrere Spielen auswählen um den Kurs abzuschließen.

### **Vorbedingungen**

Internetverbindung, darf die SPiele nach dem Standard spielen

### **Nachbedingungen**

Weiterleitung auf Google Play Store

## **2.2.6 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

## **2.2.7 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

### **2.2.8 Show colors**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

**Ereignisablauf**

**Vorbedingungen**

**Nachbedingungen**

### **2.2.9 View progress**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

**Ereignisablauf**

**Vorbedingungen**

**Nachbedingungen**

### **2.2.10 Settings**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

**Ereignisablauf**

**Vorbedingungen**

**Nachbedingungen**

### **2.2.11 Synchronize**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer



**Ereignisablauf**

**Vorbedingungen**

**Nachbedingungen**

## **2.2.12 Save local**

Dieser Use Case beschreibt den Anwendungsfall, dass der Benutzer

**Ereignisablauf**

**Vorbedingungen**

**Nachbedingungen**

## **2.3 Benutzbarkeit**

hier steht noch ein bisschen text und so, das und das

### **2.3.1 Anlernzeit**

asdfasf

### **2.3.2 Hardware Anforderungen**

keine Ahnung

### **2.3.3 Software Anforderungen**

Android 4.4 mind

## **2.4 Zuverlässigkeit**

### **2.4.1 Verfügbarkeit**

Offlien verfügbar etc. etc.

### **2.4.2 Performanz**

Response Time

## **2.5 Schnittstellen**

hgierasoujdas asdjkoiiiasjd asdhjukasdjn asm,dhasjkd

### **2.5.1 Benutzer Schnittstellen**

MockUps etc.

### **2.5.2 Hardware Schnittstellen**

Server? Datenbank?

### **2.5.3 Software Schnittstellen**

Zu Hone

## **3 NoRPG**

asdasd

### **3.1 Gedanken**

asdasd

### **3.2 Story**

asdasd

### **3.3 Abgrenzung zu MOOC**

Was ist ein MOOC. Wieso ist unser kein klassischen MOOC.

# Anhang

## Anhang A