

Dokumentation
zur Studienarbeit im Fach
Praktikum Softwareengineering

- Entwicklung einer Lernsoftware -
"myMemo"

von Susanne Kießling
Dozent: Prof. Dr. Martin Thost

19. Juni 2013

Inhaltsverzeichnis

1	Anforderungen des Kunden	4
1.1	Systemvoraussetzungen	4
1.2	Zielgruppe	4
2	Lastenheft	5
2.1	Zielbestimmung	5
2.2	Produkteinsatz	5
2.3	Produktfunktionen	5
2.4	Produktdaten	6
2.5	Produktleistungen	6
2.6	Qualitätsanforderungen	6
2.7	Ergänzungen	6
3	Aufwandskalkulation	7
3.1	Function-Point-Analyse	7
3.2	7
4	Use Case Diagramme	8
4.1	Spieler verwalten - \LF10\,\LF11\,\LF12\	8
4.2	Anzahl der Spieler wählen - \LF20\	9
4.3	Thema wählen - \LF30\	10
4.4	Spielfeldgröße wählen - \LF40\	11
4.5	Spiel starten - \LF50\	12
4.6	Highscore - \LF60\,\LF61\	13
4.7	Vokabeltraining - \LF70\	14
4.8	Audiodaten abspielen - \LF80\	15
5	Use Case Beschreibungen	16
5.1	16
5.2	16
6	Projektplan	17
6.1	17

6.2	17
7 Klassendiagramm (vorläufig)	18
7.1	18
7.2	18
8 Sequenzdiagramme	19
8.1	19
8.2	19
9 Klassendiagramm (final)	20
9.1	20
9.2	20
10 Implementierung	21
10.1	21
10.2	21

1 Anforderungen des Kunden

Die Diakonie Hochfranken möchte das Angebot an Lernsoftware in ihren Kindergärten und Kindertagesstätten erweitern. Es soll eine Lernsoftware entwickelt werden, die Gedächtnistraining und die Erweiterung des Wortschatzes spielerisch umsetzt.

1.1 Systemvoraussetzungen

Aktuell verfügt der Kunde über Einzelplatz-PC's, die in den Jahren 2005 bis 2010 angeschafft wurden. Als Betriebssystem kommt Linux und Windows zum Einsatz.

1.2 Zielgruppe

Die Besucher der Einrichtung im Alter von fünf bis zehn Jahren bilden die Zielgruppe der Software.

2 Lastenheft

2.1 Zielbestimmung

Mit der Lernsoftware wird die Möglichkeit geschaffen, das pädagogisch wertvolle Prinzip des klassischen Memory-Spiel auf eine digitale Plattform zu übertragen. Zusätzlich zum Gedächtnistraining trägt die Vokabelfunktion zur Erweiterung des Wortschatzes bei. Die Führung einer Highscore ermöglicht den Vergleich der erreichten Punkte und steigert die Motivation der Benutzer, eine bessere Platzierung zu erreichen. Das erhöht wiederum den Lerneffekt. Neben dem Spielen an sich wird der Umgang mit dem Computer spielerisch erlernt. Vor Spielbeginn sind Attribute festzulegen und Meldungen der Software zu beachten. Das schult zusätzlich Logik. Der 2-Player-Modus unterstützt außerdem die Kommunikation mit anderen Spielern.

2.2 Produkteinsatz

Die Lernsoftware kommt in den Kindertagesstätten der Diakonie Hochfranken zum Einsatz. Anwender der Software sind Besucher der Kindertagesstätte im Alter von fünf bis zehn Jahren.

2.3 Produktfunktionen

\LF10\	Spieler neu erstellen
\LF11\	Spieler laden
\LF12\	Spielerdaten ändern
\LF20\	Anzahl der Spieler wählen
\LF30\	Thema wählen
\LF40\	Spielfeldgröße wählen
\LF50\	Spiel starten
\LF60\	Highscore anzeigen
\LF61\	Urkunde drucken

\LF70\	Vokabeltraining
\LF80\	Audiodaten abspielen

2.4 Produktdaten

\LD10\	Spielerdaten
\LD20\	Highscoredaten

2.5 Produktleistungen

2.6 Qualitätsanforderungen

Funktionalität:	gut
Zuverlässigkeit:	sehr gut
Benutzbarkeit:	gut
Effizienz:	normal
Änderbarkeit:	normal
Portierbarkeit:	sehr gut
Spasfaktor:	sehr gut

2.7 Ergänzungen

Die Umsetzung der Software erfolgt in der Programmiersprache Java. Da der Kunde Linux und Windows als Betriebssystem einsetzt stellt dies die notwendige Portierbarkeit sicher.

3 Aufwandskalkulation

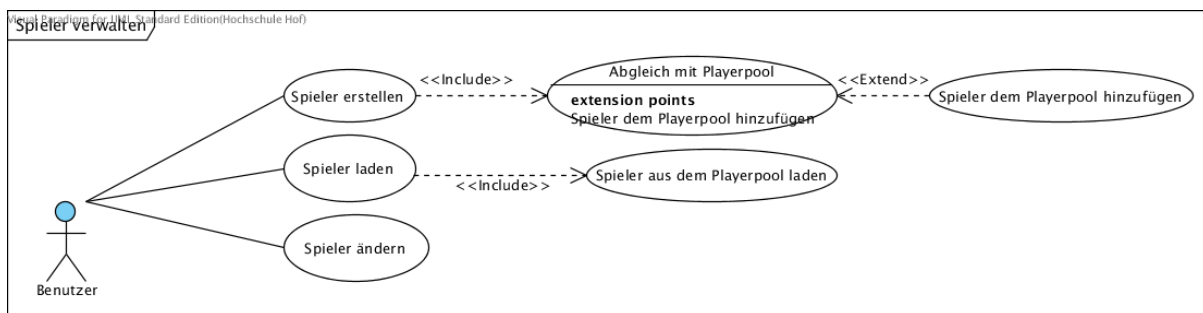
3.1 Function-Point-Analyse

3.2

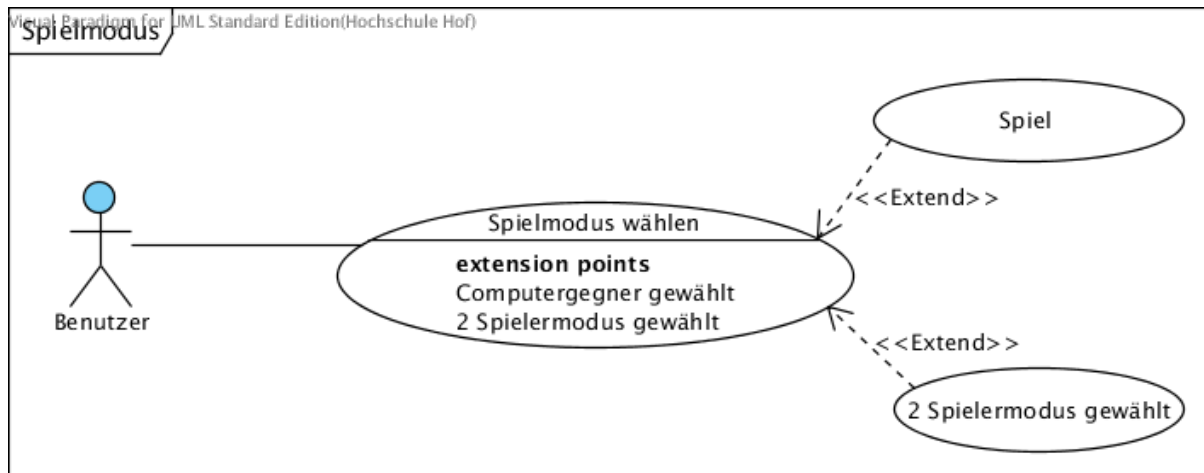
4 Use Case Diagramme

Folgend wird die Software als Use Case Diagramme dargestellt. Use Case Diagramme stellen das System von außen dar.

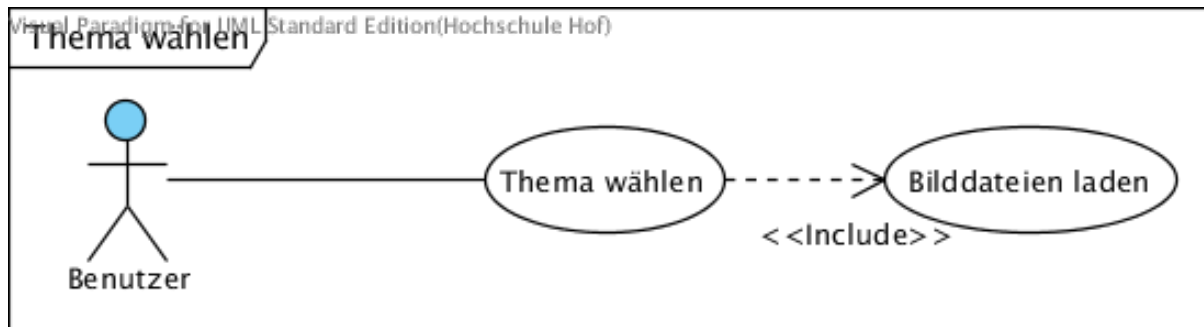
4.1 Spieler verwalten - \LF10\,\LF11\,\LF12\



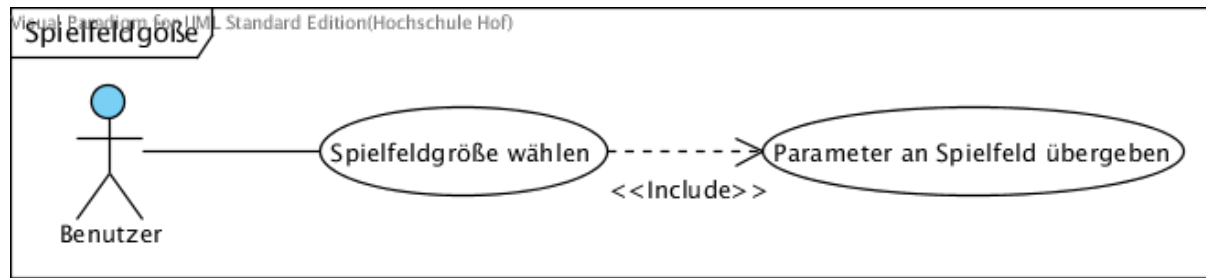
4.2 Anzahl der Spieler wählen - \LF20\



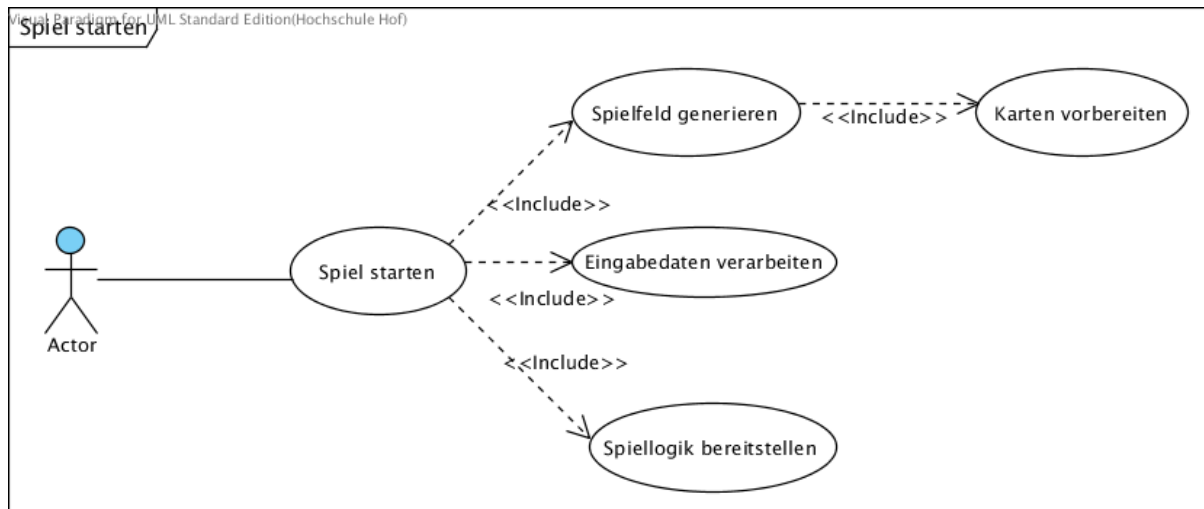
4.3 Thema wählen - \LF30\



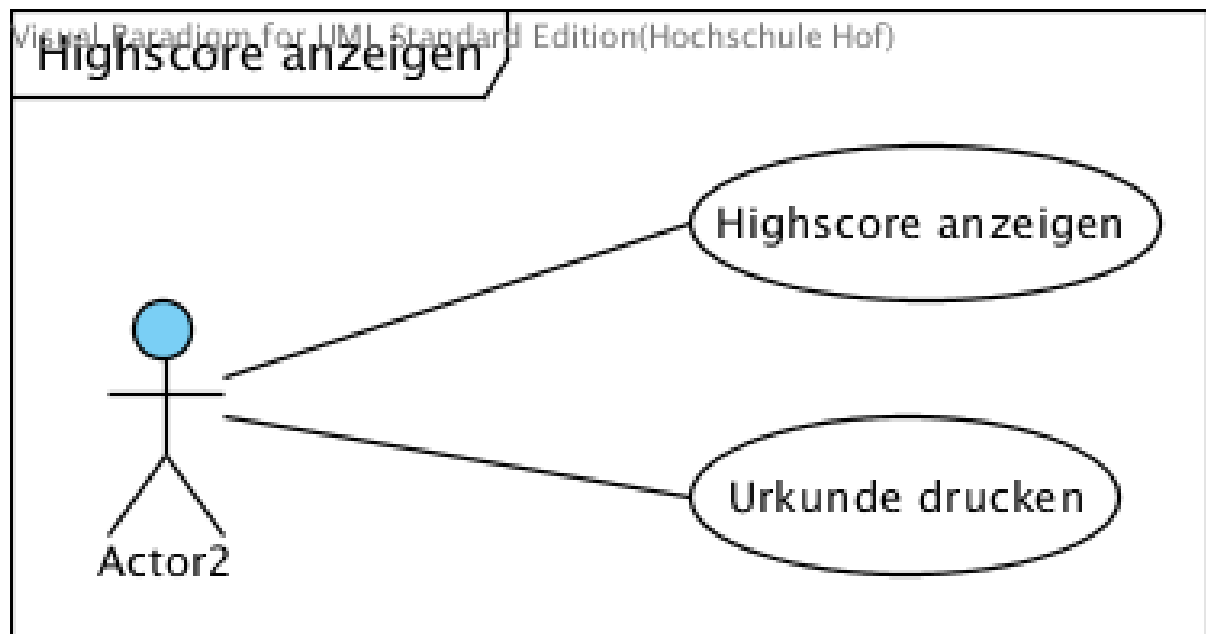
4.4 Spielfeldgröße wählen - \LF40\



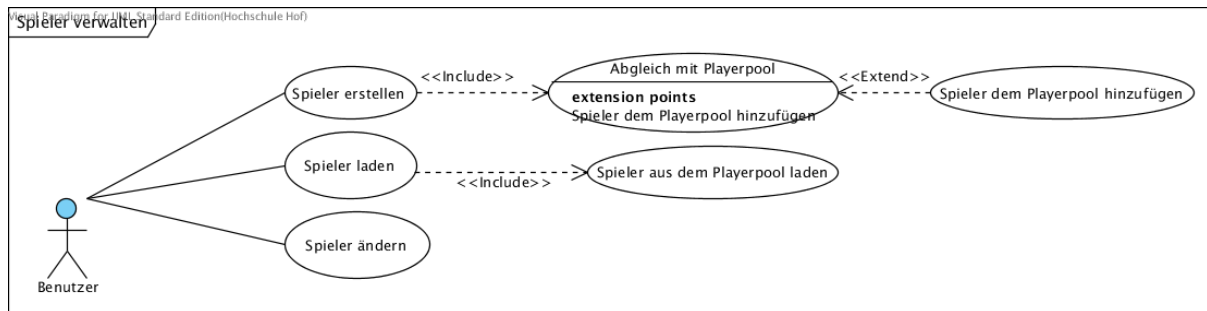
4.5 Spiel starten - \LF50\



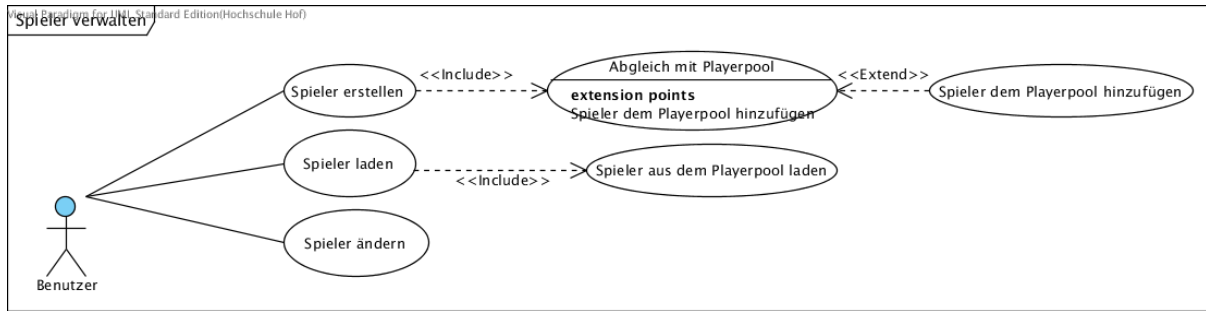
4.6 Highscore - \LF60\,\LF61\



4.7 Vokabeltraining - \LF70\



4.8 Audiodaten abspielen - \LF80\



5 Use Case Beschreibungen

5.1

5.2

6 Projektplan

6.1

6.2

7 Klassendiagramm (vorläufig)

7.1

7.2

8 Sequenzdiagramme

8.1

8.2

9 Klassendiagramm (final)

9.1

9.2

10 Implementierung

10.1

10.2

