

Anforderungsspezifikation

Projekt: SprichWort

Referenznummer: 143376-LLP-1-2008-1-SI-KA2-KA2MP

Verantwortlicher Partner: IICM, TU Graz

Leiter: Univ. Doz. Dr. DI Denis Helic

1	Softwareentwicklungsprozess	4
2	Anforderungsanalyse.....	5
2.1	Zielsetzung des Systems.....	5
3	Allgemeine Anforderungsanalyse	6
3.1	Technische Ziele des Gesamtsystems	6
3.2	Funktionale Anforderungen an das Gesamtsystem.....	6
3.2.1	Online Verfügbarkeit.....	6
3.2.2	Link Zugang zu den Teilbereichen.....	6
3.2.3	Link Zugang zu Anleitungen und Hilfe-Dateien.....	6
3.2.4	Suchmechanismus	7
3.2.5	Benutzerverwaltung	7
3.2.6	Backup von allen Daten	7
3.3	Nicht funktionale Anforderungen an das Gesamtsystem.....	7
3.3.1	Web Anwendung.....	7
3.3.2	Bedienung mit einem Standard Web Browser	7
3.3.3	Verfügbarkeit und Nachhaltigkeit.....	7
3.3.4	Zuverlässigkeit	8
3.3.5	Benutzerfreundlichkeit	8
3.3.6	Open Source	8
4	Anforderungsanalyse – SprichWort Datenbank.....	9
4.1	Zielsetzung der SprichWort-Datenbank.....	9
4.2	Datenbeschreibungsmodel	9
4.3	Technische Ergänzungen zum Datenbeschreibungsmodel	10
4.4	Funktionale Anforderungen	11
4.4.1	Erstellen der Beiträge	11
4.4.2	Editieren der Beiträge.....	11
4.4.3	Vorlagen	11
4.4.4	Strukturieren der Beiträge und Verlinkungen	11
4.4.5	Versionskontrolle	12
4.4.6	Datenzugangskontrolle.....	12
5	Anforderungsanalyse – SprichWort-Übungen	13
5.1	Zielsetzung der SprichWort-Übungen	13
5.2	Zielgruppen und Phasen	13
5.3	Rollen und Anwendungsfälle des Systems	14
5.4	Funktionale Anforderungen	16
5.4.1	Aufgabentypologie	16
5.4.1.1	Hotspot.....	16
	Sprichwort kennzeichnen	16

	Fehler kennzeichnen.....	16
	Fehler korrigieren.....	17
	Spruchwort ergänzen.....	17
5.4.1.2	Multiple Choice Fragen	17
	Bedeutung von Sprichwort bestimmen	17
	Spruchwort durch Bedeutung bestimmen	17
	Spruchwort ergänzen.....	18
	Anderssprachige Entsprechung	18
5.4.1.3	Lückentext (Buchstabe, Wort, Satz)	18
5.4.1.4	Drag and Drop (Zuordnung)	18
	Spruchwortsalat.....	18
	Zuordnung – Objekt auf Ziel (Ziel frei platzierbar).....	19
	Zuordnung – Objekt auf Ziel (Ziel in Raster)	19
5.4.1.5	Datenbank-Aufgaben	20
5.4.1.6	Textproduktion.....	20
5.4.1.7	Alphanumerische Zuordnung.....	20
5.4.1.8	Interaktive Spiele - Memoryspiel.....	20
5.4.2	Unterscheidung zwischen Übung und Test.....	21
5.4.3	Parametrisierung und Randomisierung	21
5.4.4	Prozesse.....	22
	5.4.4.1 Test / Übung und Aufgaben Erzeugung.....	22
	5.4.4.2 Test / Übung und Aufgaben Durchführung	22
	5.4.4.3 Test / Übung und Aufgaben Evaluierung	23
5.4.5	Ergebnisse betrachten.....	23
5.4.6	Überblick der funktionalen Anforderungen	23
6	Anforderungsanalyse SprichWort-Community.....	25
6.1	Zielsetzung der SprichWort-Community	25
6.2	Funktionale Anforderungen	25
	6.2.1 Diskussionsforum.....	25
	6.2.2 Annotationen und Kommentare	25
	6.2.3 Tagging Funktionalität	25
	6.2.4 Erstellen der Inhalte	25
	6.2.5 Datenaustausch.....	25

1 Softwareentwicklungsprozess

Der allgemeine Softwareentwicklungsprozess für das Projekt wird als Rapid-Prototyping bezeichnet. Dieser Prozess wird durch die folgenden Phasen gekennzeichnet:

1. Anforderungsanalyse mit der Anforderungsspezifikation als Resultat. Die Anforderungsspezifikation dient als die Grundlage für die Phase des Software Designs.
2. Software Design Phase in der die Entscheidung über die technische Infrastruktur des Gesamtsystems getroffen wird. Aufgrund der Gegebenheiten der ausgewählten Struktur werden die fehlenden funktionalen bzw. nicht funktionalen Anforderungen in der Form eines ersten Prototypes umgesetzt.
3. Umsetzung Phase die in mehreren Iterationsschritten verläuft wobei bei jedem Iterationsschritt eine weitere Approximation an die Anforderungen des Systems umgesetzt wird.

Im Allgemeinen ist für das Rapid Prototyping Modell anzumerken dass die Analyse bzw. die Design Phase sehr rasch zu erledigen sind. Man geht davon aus dass die vielen Prototypen die im Laufe des Entwicklungsprozesses entstehen zum besseren Verständnis der Anforderungen beisteuern und somit im Endeffekt zu einem System führen, das den Anforderungen besser entspricht.

Von den Endbenutzern wird eine große Flexibilität erwartet da die ersten Iterationsschritte nicht die Funktionalität des Endproduktes anbieten und die Benutzer öfters aber dafür in kleineren Schritten ihre Wünsche gegenüber dem Entwicklungsteam äußern müssen.

Auf der anderen Seite wird ähnliche Flexibilität von dem Entwicklungsteam erwartet, das dann in regelmäßigen Abständen neue Funktionen bzw. neue nicht funktionale Anforderungen der Endbenutzer rasch umsetzen müssen.

Diese Vorgehensweise hat sich aber insofern in der Praxis bewährt durch viel so genannte Web 2.0 Anwendungen die ständig an die Bedürfnisse der Endbenutzer angepasst werden.

Diesen Eigenschaften des Entwicklungsprozesses folgen stellt dieses Dokument nur eine erste Analyse der Anforderungen der Benutzer dar und daher eine Basis für das Software Design Dokument bzw. der erste System Prototyp.

2 Anforderungsanalyse

2.1 Zielsetzung des Systems

Das Hauptziel des Projektes ist es, einen Mangel an SprichWort-Lernmaterialien für die beteiligten Sprachen zu beheben und den Bedarf an mehrsprachigen IKT-unterstützten und frei zugänglichen SprichWort-Ressourcen zu decken.

Für die Erreichung dieses Zieles soll eine Internet-basierte Plattform entwickelt mit 3 Bereichen: SprichWort Datenbank, SprichWort-Übungen und SprichWort-Community.

In Details soll die Internet-basierte Plattform daher die folgenden Ziele erfüllen:

1. Die Plattform soll eine allgemein zugängliche **Onlineanwendung**, basierend auf den projekteigenen, multifunktionalen und multilingualen Inhalten darstellen.
2. Diese Plattform soll Datenbank, Übungsteil und Community Bereich beinhalten.
3. Sie soll die linguistische Beschreibung und die didaktische Aufbereitung der Sprichwörter unterstützen.
4. Die Plattform soll der effektiven Zusammenarbeit der Mitglieder des Projektkonsortiums sowohl während der Projektdauer als auch nachhaltig mit der Perspektive einer dauerhaften Expertenplattform für Korpuslinguisten, Lexikographen und Didaktiker dienen.
5. Die Plattform soll der permanenten Nutzung anderer online zugänglicher und FS-didaktisch relevanter Ressourcen (vorrangig Sprachkorpora) dienen.
6. Es soll ein **adaptives, flexibles und mehrsprachiges Lern- und Lehrmaterial** entwickelt werden, welches den Mangel an vergleichbaren mehrsprachigen Materialien mit den beteiligten Sprachen DE, SL, SK, HU deckt und somit die Grundlage für die notwendige und fachgerechte Einbeziehung von Sprichwörtern als besonders ausgeprägte Träger kultureller Informationen in das FS-Lernen schafft.

Des Weiteren soll die Plattform die folgenden Zielgruppen europaweit ansprechen:

1. FS-Lernende der Sprachen DE, SL, SK, HU auf den Niveaus B1 bis C2
2. FS-Lehrer der beteiligten Sprachen;
3. FS-Didaktiker für die beteiligten und für andere FS;
4. Experten im Bereich E-Lernmaterialentwicklung;
5. FS-Lehrer-Ausbilder.

Dieses Dokument beschreibt die technische Analyse des Systems. Diese Analyse dient der Erfassung technischer Anforderung an das System und stellt somit die Grundlage für die Phasen des Software Designs sowie der Software Entwicklung.

Das Dokument ist unterteilt in vier Hauptkapitel. Das erste Hauptkapitel analysiert die allgemeinen Anforderungen an das System; das zweite Hauptkapitel befasst sich mit den Anforderungen an die Datenbankkomponente; das dritte Hauptkapitel analysiert die Anforderung an den didaktischen Teil des Systems; das vierte Kapitel befasst sich mit Anforderungen an den Community Bereich. Desweiteren beinhaltet das erste Hauptkapitel die allgemeinen nicht funktionalen Anforderungen die für alle Teilbereiche gelten. Deswegen wird in den anderen Hauptkapiteln nur auf die funktionalen Anforderungen des jeweiligen Teilbereichs eingegangen.

3 Allgemeine Anforderungsanalyse

3.1 Technische Ziele des Gesamtsystems

Das System ist eine frei zugängliche Internet-Plattform. Der Zugang zum System und darin enthaltenen Inhalten erfolgt über das Web mittels eines Standard Web Browsers. Um das System zu benutzen sollen die Benutzer keine weiteren Anwendungen bzw. Browser-Plugins installieren.

Die Plattform bietet auf der Startseite den Zugang zu den 3 Teilbereichen. Der Zugang wird durch normale Links auf der Startseite zur Verfügung gestellt.

Die Startseite bietet außerdem weiterführende Informationen zum Projekt selbst sowie Anleitungen und Hilfe-Dateien zum Verwenden des Systems und dessen Teilbereiche.

Die Plattform bietet eine übersichtliche Navigationsstruktur erstellt nach den aktuellen Web-Usability Standards. Außerdem sollen die Inhalte im System durchsuchbar sein. Der Suchmechanismus bietet daher eine weitere Alternative zum Auffinden der relevanten Informationen im System.

Das System unterstützt desweiteren eine fein abgestimmte Benutzerverwaltung wobei die Benutzer den verschiedenen Benutzergruppen hinzugefügt werden können. Jede Benutzergruppe erhält fein abgestimmte Zugangsrechte für das System und die darin enthaltenen Inhalte. Somit ist den Benutzer einer bestimmtem Gruppe möglich die Inhalte zu erstellen und editieren (zum Beispiel der „Editoren“ Gruppe) ; eine weitere Gruppe soll die Inhalte verwenden können (zum Beispiel die Gruppe der Studenten kann die Übungen ausarbeiten); eine weitere Gruppe kann die Inhalte nur betrachten (zum Beispiel die Gruppe der anonymen Benutzer).

Der Community Bereich soll offen für alle sein, also auch für die anonymen Benutzer. In diesem Bereich soll es jedem Benutzer möglich sein die Inhalte zu erstellen, zu kommentieren, oder zu modifizieren.

3.2 Funktionale Anforderungen an das Gesamtsystem

3.2.1 Online Verfügbarkeit

Das System soll über das Internet bzw. über das Web erreichbar sein. Technisch bedeutet dies dass die Plattform einen eigenen Domain-Name erhält.

3.2.2 Link Zugang zu den Teilbereichen

Der Zugang zu den Teilbereichen erfolgt über normale Web Navigation. Dies bedeutet dass die Teilbereiche der Plattform eindeutige URLs haben müssen sodass eine Verlinkung im System möglich ist.

3.2.3 Link Zugang zu Anleitungen und Hilfe-Dateien

Anleitungen und Hilfe-Dateien die das Verwenden des Systems erleichtern sollen bekommen auch eindeutige Adressen.

3.2.4 Suchmechanismus

Ein moderner Suchmechanismus ist ein Bestandteil des Systems. Die Suche soll über alle Inhalte im System möglich sein. Verschiedene Arten von Suche sollen unterstützt werden: Volltext-Suche, die im Text von Beiträgen nach den Suchwörtern sucht. Titel-Suche, die sich auf die Suche in den Wörtern aus dem Titel der Beiträge beschränkt. Autoren-Suche, die nach dem Ersteller der Beiträge sucht.

Desweiteren soll der Suchmechanismus Möglichkeiten bieten logische Verknüpfungen zwischen verschiedenen Suchfeldern zu erstellen.

3.2.5 Benutzerverwaltung

In Details soll das System die folgenden Benutzergruppen unterstützen:

1. Administratoren, die das System technisch administrieren. Die Administration befasst sich mit Aufgaben wie Gestaltung des Look-and-Feels der Seite, Erstellung von Navigationsstrukturen im System, Konfiguration von Suchmechanismen oder Verwaltung der Benutzergruppen
2. Editoren, die die Projekt-bezogenen Inhalte erstellen und verwalten. Diese Gruppe soll sowohl die Linguisten die die Datenbank Inhalte erstellen sowie die Didaktiker die die Übungen erfassen beinhalten
3. Studenten, die die Projekt-bezogenen Inhalte verwenden um Wissen zu erlangen oder zu vertiefen indem sie zum Beispiel die Übungen und Tests lösen.
4. Anonyme Benutzergruppe, die alle Inhalte betrachten können.

3.2.6 Backup von allen Daten

Tägliches Backup von allen Projekt-relevanten Daten soll durchgeführt werden.

3.3 Nicht funktionale Anforderungen an das Gesamtsystem

3.3.1 Web Anwendung

Das System stellt eine Standard Web Anwendung. Die Anwendungslogik liegt dabei auf einem Web Server.

3.3.2 Bedienung mit einem Standard Web Browser

Um das System zu benutzen sollen die Benutzer keine weiteren Anwendungen bzw. Browser-Plugins installieren.

3.3.3 Verfügbarkeit und Nachhaltigkeit

Das System soll hochverfügbar sein. Nach dem Ende des Projektes soll die Plattform weiterhin erreichbar sein.

3.3.4 Zuverlässigkeit

Das System soll eine hohe Testzuverlässigkeit aufweisen. Eine inkonsistente Punktevergabe kann den Lernerfolg und die Akzeptanz der Kandidaten mindern.

Ebenso muss das System an sich eine hohe Zuverlässigkeit bieten, da ein Absturz während einer Prüfung nicht von Vorteil ist.

3.3.5 Benutzerfreundlichkeit

Die einfache Benutzbarkeit hat die höchste Priorität unter den Anforderungen. Instruktoren müssen in der Lage sein, ohne großes technisches Hintergrundwissen, Aufgaben, Übungen und Tests zu erstellen, durchzuführen und zu bewerten. Dabei steht vor allem die Erzeugung im Vordergrund, da eine große Menge an Aufgaben erzeugt werden soll.

3.3.6 Open Source

Eine weitere wichtige Anforderung ist, dass das System auf Open Source Software aufbaut.

4 Anforderungsanalyse – SprichWort Datenbank

4.1 Zielsetzung der SprichWort-Datenbank

Die SprichWort-Datenbank stellt eine multidimensionale hypertextuell verlinkte Matrix. Diese Matrix basiert auf dem im Rahmen des Projektes erstellten Beschreibungsmodell die zwei wesentlichen Merkmale der Datenbank ermöglicht: flexible und einfache Zugriffsmöglichkeiten und die Verlinkung mit externen Daten und Ressourcen auf dem Gebiet korpusbasierter Sprachbearbeitung.

Desweiteren ermöglicht die SprichWort-Datenbank eine einfache Möglichkeit zur Erstellung und Modifikation von Sprichwörtern.

4.2 Datenbeschreibungsmodell

Basierend auf den theoretischen Grundlagen für die Beschreibung der Gesamtmatrix in so genannten Datenblättern sind folgende Felder für die Datenbank festgesetzt:

1. Sprichwort

Beinhaltet die Grundform des Sprichwortes und kann mit den Belegen verlinkt werden.

2. Äquivalente in anderen Sprachen

Beinhaltet die Links zu den Äquivalenten in anderen Sprachen.

3. Komponenten

Beinhaltet die Komponenten des Sprichwortes. Jede Komponente ist mit einer Übersichtsseite verlinkt. Diese Übersichtsseite beinhaltet die Liste aller Sprichwörter wo diese Komponente erscheint.

4. Bedeutung(en)

Beschreibt näher die Bedeutung(en) des Sprichwortes. Dieses Feld kann Links zu wichtigen Konzepten beinhalten. Die Summe aller solchen Links bildet dann eine Ontologie der Konzepte. Desweiteren beinhaltet dieses Feld Links zu den Belegen.

5. Gebrauchsbesonderheit(en)

Erklärt näher die Besonderheiten des Gebrauchs des Sprichwortes. Ähnlich wie bei den Bedeutungen kann dieses Feld die Links zu Konzepten sowie die Links zu den Belegen beinhalten.

6. Varianten

Varianten des Sprichwortes. Dieses Feld kann Links zu den Belegen beinhalten.

7. Formvarianten

Formvarianten des Sprichwortes. Dieses Feld kann Links zu den Belegen beinhalten.

8. Ersetzung von Komponente

Beschreibt formell die Möglichkeiten zur Ersetzung einiger Komponenten des Sprichwortes. Beinhaltet Links zu den Belegen.

9. Variantenkomponenten

Beinhaltet die Komponenten der Variante des Sprichwortes. Jede Komponente ist mit einer Übersichtsseite verlinkt. Diese Übersichtsseite beinhaltet die Liste aller Sprichwörter wo diese Komponente erscheint.

10. Typische Verwendung im Text

Erklärt die Verwendung im Text und kann mit Links zu den Belegen versehen werden.

11. Belege

Belege zu dem Sprichwort. Beinhaltet Back-Links zu den verweisenden Feldern des Sprichwortes.

12. Weitere Belegsuche im Korpus

Beinhaltet die Anleitung für weitere Suche im jeweiligen Korpus mit externen Links zu den Korpus-Seiten.

4.3 Technische Ergänzungen zum Datenbeschreibungsmodell

Außer den oben beschriebenen Feldern sind folgende Datenbeschreibung bzw. Datendarstellung Merkmale zu berücksichtigen:

13. Inhaltsverzeichnis

Jeder Sprichwort Beitrag soll mit einem Inhaltsverzeichnis versehen werden. Das Inhaltsverzeichnis bietet einfache Möglichkeit zum schnellen Springen (durch Linkverfolgung) zu den für den Benutzer relevanten Feldern des Sprichwortes.

14. Kommentare

Es soll den Autoren des Sprichwortes ermöglicht werden überall in der Sprichwortbeschreibung Kommentare zu den einzelnen Angaben zu schreiben. Um die Übersichtlichkeit des Beitrages zu gewährleisten sollen die Kommentare als Tool-Tips angezeigt werden.

15. Einblendung und Ausblendung einzelner Felder

Da die Beschreibungen der einzelnen Sprichwörter lang werden könnten soll die Übersichtlichkeit der Darstellung durch die Möglichkeit des Ausblendens und Einblendens der einzelnen Felder erhalten werden.

16. Editieren der Felder

Es soll die Möglichkeit gegeben werden dass die Editoren die Felder einzeln bearbeiten können. Dadurch wird die Usability des Datenerfassungsmoduls gesteigert.

17. Automatische Verlinkung

Das System soll die Möglichkeit einer automatischen Verlinkung zwischen zB. Äquivalenten oder Sprichwörter und Konzepten ermöglichen.

18. Links zu Übungen

Jedes Sprichwort das didaktisch vorbereitet wird soll mit dem jeweiligen Sprichwort Beitrag verlinkt werden.

4.4 Funktionale Anforderungen

Außer der oben genannten Anforderungen an das Datenbeschreibungsmodell hat die Analyse folgende funktionalen Anforderungen identifiziert.

4.4.1 Erstellen der Beiträge

Erstellen eines neuen Beitrages soll durch das System unterstützt werden. Das System fragt den Benutzer nach dem Namen des neuen Beitrages. Das System ist zuständig für die Datenintegrität und Datenkonsistenz und führt alle benötigten Tests durch um festzustellen ob der neue Beitrag mit dem ausgewählten Namen erstellt werden kann.

4.4.2 Editieren der Beiträge

Editieren der Beiträge erfolgt im Web Browser mittels eines „What you see is what you get“ Editors.

Als Alternative bietet sich die Möglichkeit des Verwendens eines einfachen im System integrierten Text-Editors. Dabei soll auf die direkte Eingabe vom HTML Code verzichtet werden. Es soll möglich sein einfache Textformatierungen durchzuführen wie zB. Texte kursiv, fett, unterstrichen darzustellen, Überschriften zu definieren, Liste oder Tabellen einzufügen, usw.

Die Syntax für die direkte Text Eingabe soll einfach wie möglich gehalten werden auf jeden Fall so dass keine Programmierkenntnisse zur Dateneingabe notwendig sind.

4.4.3 Vorlagen

Das System soll für jede Sprache Vorlagen anbieten die das Eingeben von Daten vereinfachen. So soll es möglich sein eine Vorlage aus einem Pool von Vorlagen auszuwählen und dann einfach an den entsprechenden Stellen des Datenbeschreibungsmodells eigenen Daten einzufügen.

4.4.4 Strukturieren der Beiträge und Verlinkungen

Die Beiträge sollen dreistufig strukturiert werden. Jede Stufe entspricht dem aktuellen Bearbeitungsstand eines Sprichwortes:

1. Sprichwort in Arbeit
2. Sprichwort fertig für interne Zwecke
3. Sprichwort fertig.

Die letzte Stufe soll dann frei zugänglich sein. Eine weitere Strukturierung erfolgt über die im Projekt beteiligten Sprachen sowie die oben erwähnten Übersichtseiten (Konzepte, Komponenten, usw.)

Diese Übersichtsseiten für die Stufen werden automatisch erstellt. Man muss nur bei dem jeweiligen Sprichwort einen Link auf die entsprechende Übersichtsseite setzen.

4.4.5 Versionskontrolle

Das System soll eine Versionskontrolle unterstützen die es ermöglicht eine frühere Version des Beitrages wiederherzustellen.

4.4.6 Datenzugangskontrolle

Für die Stufe 1 und 2, also Sprichwörter in Arbeit bzw. Sprichwörter die fertig sind für die internen Zwecke soll es eine strikte Datenzugangskontrolle geben. Diese Zugangskontrolle kann auf der Ebene der einzelnen Sprichwörter festgelegt werden.

5 Anforderungsanalyse – SprichWort-Übungen

5.1 Zielsetzung der SprichWort-Übungen

Ziel dieses Teilabschnittes des SprichWort Projekts ist es, ein webbasiertes Übungs- und Testsystem mit formativen und summativen Charakter zu erstellen. Dieses muss sowohl standardisierte objektive als auch nicht standardisierte subjektive Fragen verarbeiten und verwalten können.

Didaktiker oder andere Instruktoren können durch dieses System, Übungs- oder Testsequenzen sowie deren beinhalteten Fragen erstellen und verwalten. Im Vordergrund steht hierbei die einfache Verwendung des Systems, ohne allzu großen technischen Hintergrund. Dabei wird gefordert, dass die Plattform eine einfache Syntax und einen Editor für diese Aufgaben zur Verfügung stellt.

Für die Prüfungsdurchführung werden die Aufgabenstellungen, den Kandidaten (Schüler oder Studenten) zur Verfügung gestellt. Dazu müssen sich die Testpersonen auf der Plattform einloggen und im Anschluss die Prüfungen oder Selbstüberprüfungen über einen Webbrowser durchführen. Bei den Prüfungen wird zwischen Übungen und Tests unterschieden. Ein Test ermöglicht nur einen Lösungsversuch pro Frage und muss in einer bestimmten Zeitvorgabe gelöst werden. Bei den Übungen existiert diese Restriktion nicht.

Die Evaluierung der Fragen kann entweder direkt nach jedem Lösungsversuch, durch das System, oder im Anschluss einer Prüfung, durch den Instruktor, stattfinden. Der Prüfer kann die ausständigen Antworten, die sich aus Textaufgaben ergeben, über eine spezielle Schnittstelle bewerten.

Das System bietet ebenso verschiedene Ansichten auf die Prüfungsergebnisse. Zum einen können Prüfer die gesamten Assessments betrachten und zum anderen können Kandidaten deren eigene Prüfungen durchsehen. Zusätzlich können Testpersonen die subjektiven Antworten anderer Studenten betrachten und evaluieren. Zusammenfassend für die Ergebnisse gibt es Selbstbewertungsbogen, die vom Studenten und auch teilweise vom Instruktor ausgefüllt werden.

5.2 Zielgruppen und Phasen

Die Zielgruppen des Systems bauen auf verschiedenen Schwierigkeitsstufen auf. Diese können in folgende Gruppen gegliedert werden:

- **Stufen B1-B2**
- **Stufen C1-C2**
- **Germanistikstudenten**

Für Germanistikstudenten sind nicht nur Fragen aus den Stufen B und C bestimmt, sondern auch linguistische Fragen, in denen die Klassifikation von Sprichwörtern und deren etymologischer Ursprung Thematik ist.

Die Übungen der Stufen B und C werden in folgende Phasen unterteilt:

- 1-2. **Sprichwörter erkennen und entschlüsseln.** Bei dieser Phase sollen Sprichwörter möglichst in der Originalform erlernt werden. Dazu ist es wichtig eine Verknüpfung zur Datenbank zu erstellen.
3. **Sprichwörter festigen.** Der Fokus liegt hier auf der Vielfältigkeit der Fragen. Die Übungsformen sollen somit verschiedene Lerntypen ansprechen. Zu den klassischen Aufgaben wie Multiple Choice oder Zuordnungsübungen sollen auch spielerische interaktive Aufgaben gestellt werden.
4. **Sprichwörter verwenden.** Diese Phase umfasst die metakommunikative Einbettung der Sprichwörter in Texten sowie die subjektive Beantwortung von Textaufgaben.

Im Anschluss zu den formativen Übungen erfolgt das Testen der Sprichwörter. Der genaue Inhalt dieser Tests wurde noch nicht festgelegt.

5.3 Rollen und Anwendungsfälle des Systems

Die Rollen des Systems bestehen aus einem Instruktor und einem Kandidaten.

Instruktor – Der Instruktor beinhaltet mehrere kleine Subrollen und stellt den Lehrenden dar. Diese Subrollen setzen sich wiederum aus einem Item Autor, einem Item Verwalter, einem Testkonstrukteur, einem Proktor und einem Bewerter zusammen. Dadurch, dass eine Person all diese Aufgaben bewältigen muss, können diese auf die Rolle des Instructors zusammengefasst werden. Die Aufgaben eines Instructors definieren sich wie folgt:

- Erstellen, Modifizieren und Löschen von Items
- Erstellen, Modifizieren und Löschen von Tests und Übungen
- Prüfungen den Kandidaten zur Verfügung stellen (damit verbunden eine Benutzer- und Gruppenverwaltung)
- Lernende durch einen Lernprozess dirigieren.
- subjektive Antworten bewerten
- Kandidaten Kommentare zu anderen Ergebnissen kommentieren
- Testergebnisse evaluieren
- Selbstbewertungsbogen der Studenten teilweise befüllen

Eine wichtige Anforderung dieser Rolle ist die einfache Verwaltung von Prüfungen und Prüfungsfragen.

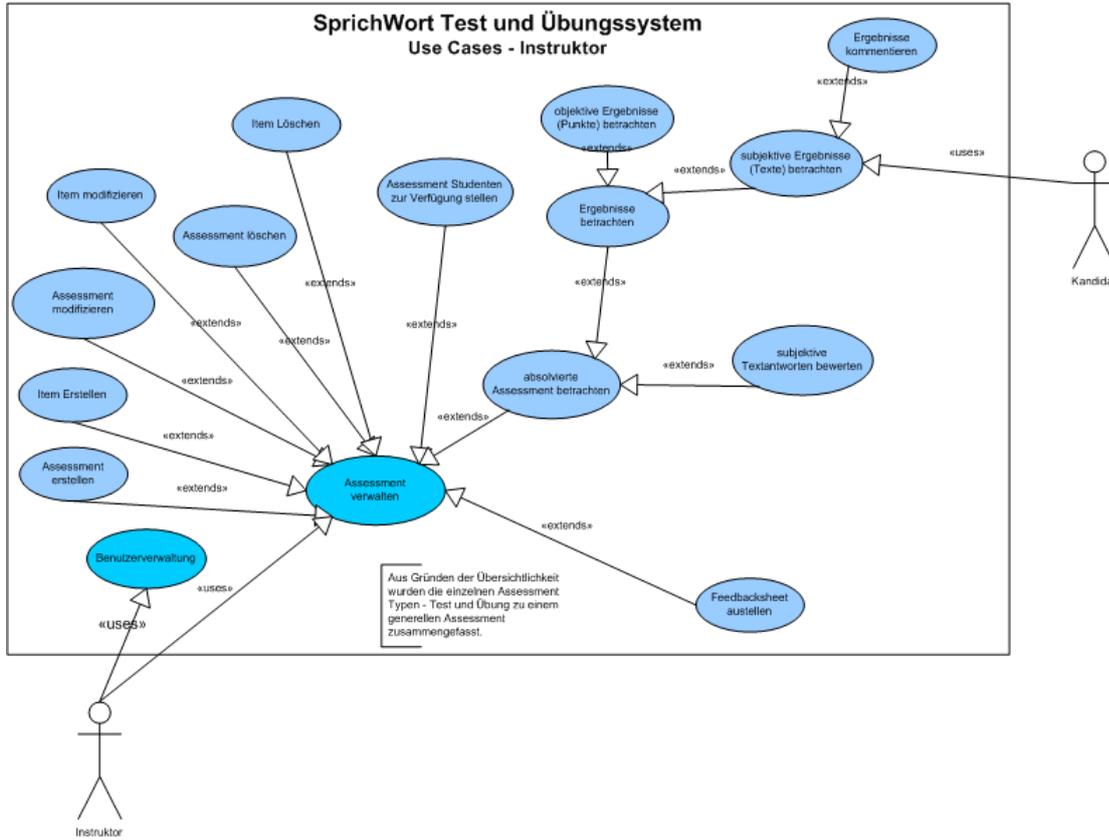
Kandidat – Der Kandidat setzt sich aus den verschiedenen Zielgruppen des Systems zusammen und bildet den Lernenden Part der beiden Rollen. Folgende Aufgaben und Bedürfnisse bilden seinen Charakter:

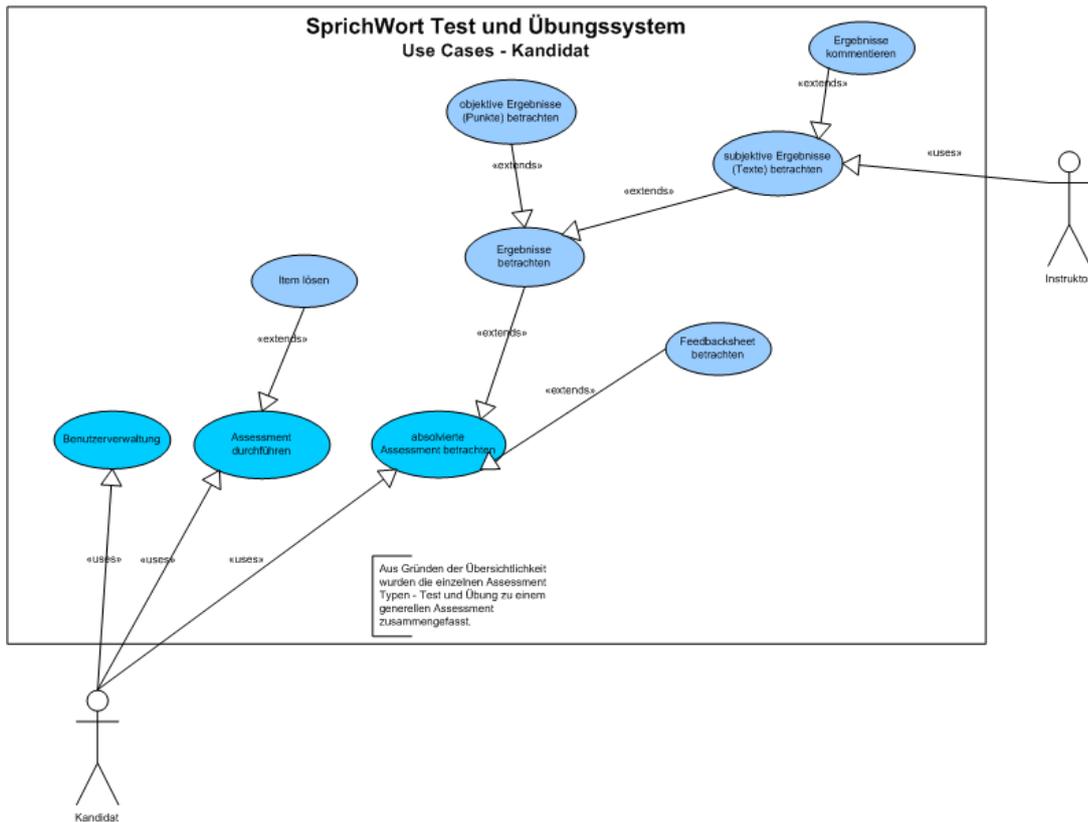
- eigene Benutzerverwaltung (registrieren, einloggen, verwalten)
- Prüfungen durchführen
- Prüfungsfragen lösen
- die Ergebnisse der absolvierten Prüfungen betrachten

- Kommentierung der Textproduktionen anderer Kandidaten
- Selbstbewertungsbogen ausfüllen
- Selbstbewertungsbogen betrachten

Eine grundlegende Anforderung dieser Rolle ist eine einfache Navigation durch verschiedene Prüfungsaufgaben.

Die nachfolgenden Diagramme stellen die verschiedenen Anwendungsfälle des Systems dar.





5.4 Funktionale Anforderungen

5.4.1 Aufgabentypologie

Das Projekt fordert eine Fülle an verschiedenen Aufgabentypen. Diese Typenformate können in Hotspot, Multiple Choice Fragen, Lückentext, Korrektur, Zuordnungsaufgaben und Textproduktion unterteilt werden. Ebenso sind Aufgaben wie zum Beispiel ein Memory oder auch hybride Formen, wie Hotspot Korrektur oder Hotspot Multiple Choice gefordert. Die nachfolgende Auflistung gibt einen Überblick über die detaillierten Anforderungen der Aufgabentypologie.

5.4.1.1 Hotspot

Sprichwort kennzeichnen

Ablauf: Ein oder mehrere Sprichwörter müssen in einem Text erkannt werden. Dazu klickt der Kandidat auf die Textpassagen. Werden alle Sprichwörter erkannt, gibt es eine volle Punktzahl, ansonsten eine prozentuelle Aufteilung.

Zuordnung zum Aufgabenbedarf: [1.]

Fehler kennzeichnen

Ablauf: Ein oder mehrere Fehler werden in einem Text dargestellt. Der Kandidat muss diese Fehler markieren. Werden alle Fehler gefunden, erreicht der Kandidat die volle Punktzahl, ansonsten eine prozentuelle Aufteilung.

Zuordnung zum Aufgabenbedarf: [5.]

Fehler korrigieren

Ablauf: Ein oder mehrere Fehler werden in einem Text dargestellt. Der Kandidat muss diese Fehler markieren und korrigieren. Werden alle Fehler gefunden und ausgebessert, erreicht der Kandidat die volle Punktzahl, ansonsten eine prozentuelle Aufteilung.

Zuordnung zum Aufgabenbedarf: [6.]

Anmerkung: Diese Aufgabe ist eine hybride Form. Sie erfordert eine Hotspot und Korrektur Möglichkeit.

Sprichwort ergänzen

Ablauf: Mehrere Fehler oder Lücken werden in einem Text dargestellt. Der Kandidat muss diese Fehler oder Lücken markieren und durch ein Multiple Choice Menü korrigieren oder ergänzen. Werden alle Fehler oder Lücken gefunden und ausgebessert, erreicht der Kandidat die volle Punktzahl, ansonsten eine prozentuelle Aufteilung.

Zuordnung zum Aufgabenbedarf: [8.]

Anmerkung: Diese Aufgabe ist eine hybride Form. Sie erfordert eine Hotspot und Multiple Choice Möglichkeit und wird nur verwendet wenn mehrere Fehler oder Lücken in einem Text vorhanden sind.

5.4.1.2 Multiple Choice Fragen

Bedeutung von Sprichwort bestimmen

Ablauf: Ein Sprichwort wird angezeigt. Der Kandidat muss die Bedeutung aus einer Multiple Choice Liste auswählen. Wird die richtige Antwort gewählt so gibt es die vollen Punkte, ansonsten werden keine Punkte vergeben.

Zuordnung zum Aufgabenbedarf: [2.]

Sprichwort durch Bedeutung bestimmen

Ablauf: Die Bedeutung eines Sprichwortes wird angezeigt. Der Kandidat muss das dazugehörige Sprichwort aus einer Multiple Choice Liste auswählen. Wird die richtige Antwort gewählt, so gibt es die vollen Punkte, ansonsten werden keine Punkte vergeben.

Zuordnung zum Aufgabenbedarf: [12.]

Spruchwort ergänzen

Ablauf: Ein Text mit einer Lücke wird angezeigt. Der Kandidat muss den passenden semantischen Teil aus einer Multiple Choice Liste auswählen. Wird die richtige Antwort gewählt so gibt es die vollen Punkte, ansonsten werden keine Punkte vergeben.

Zuordnung zum Aufgabenbedarf: [8.]

Anderssprachige Entsprechung

Ablauf: Ein Sprichwort wird angezeigt. Der Kandidat muss die passende anderssprachige Entsprechung aus einer Multiple Choice Liste auswählen. Wird die richtige Antwort gewählt so gibt es die vollen Punkte, ansonsten werden keine Punkte vergeben.

Zuordnung zum Aufgabenbedarf: [10.][9.]

5.4.1.3 Lückentext (Buchstabe, Wort, Satz)

Ablauf: Eine oder mehrere Lücken werden in einem Text dargestellt. Der Kandidat muss diese markieren und ausbessern. Werden alle Lücken ausgebessert, gibt es eine volle Punktzahl, ansonsten eine prozentuelle Aufteilung.

Zuordnung zum Aufgabenbedarf: [3.] [12.] [8.]

Anmerkung: Diese Aufgabe ist eine hybride Form. Sie erfordert eine Hotspot, Lückentext und Korrektur Möglichkeit.

5.4.1.4 Drag and Drop (Zuordnung)

Spruchwortsalat

Ablauf: Es wird ein Feld mit vermischten semantischen Teilen eines oder mehrerer Sprichwörter angezeigt. Ein Kandidat muss diese Anordnung entwirren und in dafür vorgesehenen Felder ziehen um einen korrekten Satz zu bilden. Die Punktevergabe erfolgt nach gelösten Sätzen.

Zuordnung zum Aufgabenbedarf: [4.]

Ein mögliches Beispiel:

Übung/Aufgabe: Sprichwortsalat

Jeder	was	macht nicht	ist nicht	Ausnahmen	Geld allein
	jeder	soll	Es	Der Zweck	glänzt
vor seinen eigenen		alles		heiligt	kehren

bestätigen	die Regel Gold	die Mittel	Tür	glücklich
1.....				
2.....				
3.....				
4.....				
5.....				

Zuordnung – Objekt auf Ziel (Ziel frei platzierbar)

Ablauf: Es wird eine Liste mit Objekten (Texte, Bilder) angezeigt. Der Kandidat muss diese Elemente auf ein Zielobjekt ziehen. Werden alle Objekte richtig zugeordnet, gibt es eine volle Punktzahl, ansonsten eine prozentuelle Aufteilung.

Anmerkung: Das Ziel ist bei dieser Aufgabe frei platzierbar.

Ein mögliches Beispiel:

Übung/Aufgabe: Zuordnungsübung

- A. guter Rat teuer
- B. Geld allein macht nicht glücklich.

1. Die pure Freude Petra Haltmayr hat diesen Blitz-Sieg verdient. Endlich haben sich all die Trainingsstrapazen ausgezahlt, noch dazu in barer Münze. Für ihren eineinhalb Minuten-"Flug" mit Tempo 120 über die Piste am Lake Louise gab es immerhin über 90 000 Mark. Doch Viel mehr dürfte für Petra Haltmayr zählen, Asse wie Isolde Kostner und Renate Götschl hinter sich gelassen zu haben.

2. Jeder Besucher muss erst einmal an der Palastwache vorbei, bevor ihm der Eintritt gewährt wird. Durch das große Tor hindurch und schon steht der Neuankömmling mitten auf dem Bazar. Um sich herum ein Rund mit Zelten und Stuben, dazwischen orientalisch gekleidete Mädchen und Jungen. Jetzt ist - zur Schreibstube, zu den Baumeistern oder lieber in die Schneiderei?

Zuordnung – Objekt auf Ziel (Ziel in Raster)

Ablauf: Es werden eine Liste mit Objekten (Texte, Bilder) und ein Raster mit Zielen angezeigt. Der Kandidat muss diese Elemente auf ein Zielobjekt ziehen. Werden alle Objekte richtig zugeordnet, gibt es eine volle Punktzahl, ansonsten eine prozentuelle Aufteilung.

Zuordnung zum Aufgabenbedarf: [10.] [11.] [12.]

Anmerkung: Zu diesem Typen zählen Aufgaben wie die Zuordnung von anderssprachigen Entsprechungen auf Sprichwörter, Bilder auf Sprichwörter, Sprichwörter auf Bilder, Sprichwörter auf Bedeutungen und Hörtexte auf Sprichwörter.

5.4.1.5 Datenbank-Aufgaben

Ablauf: Ein Kandidat sucht bestimmte Informationen aus der SprichWort Datenbank. Diese Informationen verwendet er um die Aufgabe zu lösen.

Anmerkung: Diese Aufgabe kann auf allen Aufgabentypen basieren.

5.4.1.6 Textproduktion

Ablauf: Es wird eine offene Frage angezeigt. Der Kandidat muss dazu einen Text produzieren. Diese Frage kann nicht automatisch vom System beantwortet werden und muss somit an einen Instruktor weitergeleitet werden. Die Bewertung erfolgt im Anschluss nach Ermessen des Instructors.

Zuordnung zum Aufgabenbedarf: [12.] [14.] [15.]

Anmerkung: Es werden bei der Durchführung Tipps und Hinweise gegeben werden.

5.4.1.7 Alphanumerische Zuordnung

Ablauf: Ein Kandidat muss einen semantischen Teil oder ein anderes Objekt einem weiteren Teil / Objekt zuordnen. Dies erfolgt durch die Eingabe verschiedener Indizes.

Zuordnung zum Aufgabenbedarf: [7.]

Ein mögliches Beispiel:

Übung 8: Bilden Sie Sprichwörter, indem Sie die richtigen Satzteile kombinieren.

- | | |
|----------------------|--------------------------|
| 1. Ohne Fleiß, | a) nicht weit vom Stamm. |
| 2. Wo ein Wille ist, | b) Gold im Munde. |
| 3. Übung macht | c) sind tief. |
| 4. Der Apfel fällt | d) kein Preis. |
| 5. Der Klügere | e) ist auch ein Weg. |
| 6. Morgenstunde hat | f) gibt nach. |
| 7. Stille Wasser | g) den Meister. |

1 :	2 :	3 :	4 :	5 :	6 :	7 :
-----	-----	-----	-----	-----	-----	-----

5.4.1.8 Interaktive Spiele - Memoryspiel

Ablauf: Es wird ein Memoryfeld angezeigt. Der Kandidat muss dieses lösen. Punkte werden bei dieser speziellen Übungsaufgabe keine verteilt.

5.4.2 Unterscheidung zwischen Übung und Test

Das System unterstützt sowohl formative Prüfungen wie Self- oder Group Assessments als auch summative Prüfungen wie Tests.

Formative Prüfungen können auch als Übung bezeichnet werden. Sie dienen dazu Informationen und Richtungsweisungen für den laufenden Lernprozess zu liefern. Ein Kandidat kann über ein Menü je nach Phase und Zielgruppe verschiedene Übungen auswählen.

Die Übungen bestehen aus den Instanzen der oben genannten Aufgabentypen. Die Aufgaben werden je nach Typus vom System oder von einem Instruktor bewertet. Somit sind diese Übungen nicht nur Selbstbewertungen sondern besitzen auch einen Charakter der Fremdbewertung. Um einen weiteren Lerneffekt durch Kommunikation in die Übung zu bringen, ist es möglich, dass bereits produzierte Texte von Textaufgaben durch andere Kandidaten kommentiert werden können. Diese Kommentare erzeugen somit eine Gruppenbewertung.

Bei der Durchführung der Übungen ist es möglich Fragen mehrmals zu beantworten. Die Punkte werden zwar gespeichert, haben aber keinen Einfluss auf die Endnote.

Übungen beinhalten neben Aufgaben, die Punkte erzeugen, auch Aufgaben, die rein zur Interaktion dienen. Solche Aufgaben sind zum Beispiel ein Memoryspiel.

Summative Prüfungen werden auch als Tests bezeichnet. Sie dienen zur Feststellung von erlerntem Wissen. Der Instruktor weist Kandidatengruppen eine Prüfung zu. Der Kandidat wählt über ein Menü den zugewiesenen Test aus und führt ihn durch.

Die Tests bestehen aus den oben genannten Aufgabentypen mit der Ausnahme von Aufgaben die keine Punkte liefern. Ansonsten besitzt ein Test den gleichen Ablauf wie eine Übung mit der Restriktion, dass die Aufgaben nur einmal gelöst werden dürfen.

Die Punkte werden ebenso im System gespeichert. Nach der erfolgten Bewertung können die Ergebnisse vom Kandidaten betrachtet werden.

5.4.3 Parametrisierung und Randomisierung

Um Aufgaben interessanter zu gestalten unterstützt das System Parametrisierung und Randomisierung der Aufgaben. So können Templates erstellt werden, die über spezielle Parameter konfigurierbar sind. Wird eine Instanz dieses Templates erstellt wird eine Konfiguration zufällig gewählt und angezeigt.

Bei einer Hotspot Aufgabe werden zum Beispiel drei zuvor konfigurierte Hotspots jeweils zufällig initialisiert.

1. Aufruf:

Übung/Aufgabe: Hotspot – Fehler suchen

Die Wasserballer schafften mit psychologischer Hilfe die Olympia-Qualifikation. Handball-Bundestrainer Heiner Brand gilt als Fan der Sportpsychologie. Als seine Mannschaft

Europameister wurde, ließ er vor jedem Spiel im Besprechungszimmer Sprüche wie diesen aufhängen: „Wenn es einem Glauben gibt, der Wolkenkratzer versetzen kann, dann ist es der Gedanke an die eigene Stärke.“

2. Aufruf:

Übung/Aufgabe: Hotspot – Fehler suche

Die Wasserballer schafften mit psychologischer Hilfe die Olympia-Qualifikation. Handball-Bundestrainer Heiner Brand gilt als Fan der Sportpsychologie. Als seine Mannschaft Europameister wurde, ließ er vor jedem Spiel im Besprechungszimmer Sprüche wie diesen aufhängen: „Wenn es einen Gedanken gibt, der Berge versetzen kann, dann ist es der Gedanke an die eigene Stärke.“

5.4.4 Prozesse

Das Projekt beinhaltet vier Hauptprozesse. Diese Prozesse umfassen die Erzeugung, Durchführung und Evaluierung von Aufgaben und Test / Übungen, sowie die Betrachtung derer Ergebnisse. Die nachfolgenden Abschnitte geben einen Überblick über den Ablauf und die Anforderungen der jeweiligen Projekte. Auf das Usermanagement wird hier keine Rücksicht genommen, da es als generelle Anforderungen bereits erwähnt wurde.

5.4.4.1 Test / Übung und Aufgaben Erzeugung

Bei der Erzeugung von Tests und Übungen kann aus den oben genannten Aufgabentypen gewählt werden. Für diese Erstellung ist ein Autorenwerkzeug vorgesehen. Dieses Werkzeug ermöglicht eine schnelle Erstellung über eine einfache Syntax oder ein UI.

Für eine Aufgabe müssen verschiedene Parameter definiert werden. Die wichtigsten Elemente hierfür sind der Aufgabentyp, die Frage oder der Stimulus, die Instruktion, die Antwortmöglichkeiten, wenn nötig die Distraktoren und die richtige Lösung. Zusätzlich können auch Punkte und Hinweise definiert werden. Ebenso wird die Parametrisierung und Randomisierung berücksichtigt. Der Instruktor kann der erstellten Aufgabe auch eine spezielle Phase zuweisen. Diese dient zur Orientierung des Kandidaten.

Nachdem die Aufgabe erstellt wurde, kann diese einem Test oder einer Übung zugewiesen werden und dadurch Sequenzen erstellt werden. Bei Tests stellt das System die Möglichkeit einer Zeitkonfiguration zur Verfügung. Damit kann definiert werden, wie lange ein Kandidat eine Prüfung lösen darf. Im Anschluss werden die erstellten Übungen in ein, dem Kandidaten zur Verfügung stehendes, Menü eingetragen.

Nach der Erstellung dieser Daten werden diese vom System gespeichert. Der Instruktor kann auch nachträglich Änderungen vornehmen. Daraus entsteht das Bedürfnis einer Datenbank für Aufgaben, Tests und Übungen, welche zudem auch eine Revisionsverwaltung besitzt.

5.4.4.2 Test / Übung und Aufgaben Durchführung

Die Durchführung von Übungen und Tests unterscheidet sich in einigen Punkten. Bei Übungen kann der Kandidat selbstständig aus einer Übungsliste auswählen. Der Kandidat kann während der Durchführung, Hinweise und die richtige Lösung anzeigen. Wird die

Funktion „Lösen“ gewählt, so werden alle richtig gelösten Teile in einer bestimmten Farbe angezeigt. Die falschen oder noch ausstehenden Antworten, bei denen eine Lösung angezeigt werden musste, werden mit einer anderen Farbe markiert. Bei den Lösungsversuchen werden die Punkte mitdokumentiert. Ein Kandidat kann mehrmals eine Aufgabe lösen und selbst den Pfad durch die Übungssequenz wählen.

Bei Tests hingegen muss der Instruktor einem Kandidaten (über Gruppen) einen Test zuordnen. Der Kandidat kann im Anschluss aus einem Testmenü auswählen und die Durchführung beginnen. Bei dieser Prüfungsart werden keine Hilfestellungen angezeigt. Zusätzlich wird dem Kandidaten eine verbleibende Zeit angezeigt, in welcher er die Aufgaben lösen muss. Nach Ablauf dieser Zeit dürfen keine Lösungsversuche durchgeführt werden.

Beide Prüfungstypen beinhalten die gleiche Navigation. Ein Kandidat kann frei zwischen allen Aufgaben wählen, welche er lösen möchte. Bei Tests besteht die Restriktion, dass eine Aufgabe nur einmal gelöst werden darf.

5.4.4.3 Test / Übung und Aufgaben Evaluierung

Die Evaluierung erfolgt entweder direkt nach der Beantwortung der Frage durch das System, oder indirekt durch einen Instruktor. Die Art der Bewertung hängt vom Aufgabentyp ab. Textproduktionen können nur subjektiv durch einen Instruktor bewertet werden. Das System unterstützt somit die Vergabe von Feedback in Form von Punkten oder Kommentaren.

Bei der subjektiven Evaluierung stellt das System dem Instruktor eine einfache Schnittstelle zur Bewertung zur Verfügung. Ebenso gibt es eine Schnittstelle für andere Kandidaten um die Kommentierung von Texten ihrer Kollegen vorzunehmen.

Der Kandidat kann auch Selbstbewertungen mit Hilfe eines Selbstbewertungsbogens durchführen. In diese Selbstbewertungsbögen fließen auch Kommentare von Instruktoren.

5.4.5 Ergebnisse betrachten

Das System bietet dem Kandidaten die Möglichkeit seine Ergebnisse anzusehen. Dazu wird eine einfache Schnittstelle zur Verfügung gestellt. Über eine andere Schnittstelle kann er die Kommentare seiner Kollegen zu seinen Texten betrachten. Ebenso hat der Kandidat Zugriff auf die Selbstbewertungsbogen.

Der Instruktor kann über das System die Ergebnisse aller durchgeführten Übungen und Tests betrachten.

5.4.6 Überblick der funktionalen Anforderungen

Generelle Anforderungen

- Übungen und Tests: Das System soll Übungen wie auch Tests unterstützen.
- Aufgabentypologie: Das System soll eine große Anzahl an Aufgaben / Fragen zur Verfügung stellen.

- Parametrisierung / Randomisierung: Das System soll einen Mechanismus zur Parametrisierung und Randomisierung von Aufgaben besitzen.
- Authentifizierung / Userverwaltung: Das System soll einen Mechanismus für die Authentifizierung und Userverwaltung zur Verfügung stellen.
- Zugriffskontrolle / Gruppenverwaltung: Das System soll den Zugriff von Unbefugten auf Tests oder Übungen verhindern.
- Mediasupport: Das System soll multimediale Inhalte unterstützen.
- Sprache: Die Menüpunkte sollen in unterschiedlichen Sprachen darstellbar sein.

Test / Übung und Aufgaben Erzeugung

- Erzeugen: Das System soll eine einfache Syntax oder einen Editor zur Erzeugung von Test/Übungen und Aufgaben bieten.
- Test und Übungssequenzen: Das System soll eine einfache Syntax oder einen Editor zur Erzeugung von Test oder Übungssequenzen bieten.
- Zeit: Der Instruktor soll über das System einfache Zeitkonfigurationen für Tests vornehmen können.
- Punkte: Das System soll ermöglichen Punkte bei Aufgaben zu definieren.
- Phasen: Das System soll die Möglichkeit bieten, verschiedene Phasen zu definieren und dieser in Übung anzuzeigen.
- Revisionen: Das System muss ein Revisionshandling von Aufgaben unterstützen.

Test / Übung und Aufgaben Durchführung

- Zuweisen: Der Instruktor soll über das System verschiedenen Kandidatengruppen Tests zuweisen können.
- Auswahl: Das System soll ein Menü zur Verfügung stellen, von dem Kandidat die Übung oder den Test auswählen kann.
- Navigation: Das System soll eine Navigation zwischen Aufgaben eines Tests oder einer Übung ermöglichen.
- Zeit: Der Kandidat soll durch das System mit einer zuvor definierten Zeitvorgabe bei der Durchführung beschränkt werden.
- Hinweise: Das System soll bei Übungen definierte Hinweise anzeigen.
- Lösungen: Bei Übungen soll das System die Option bieten Lösungen anzuzeigen.

Test / Übung und Aufgaben Evaluierung

- Evaluierung: Aufgaben sollen entweder vom System oder von einem Instruktor bewertet werden können (abhängig von deren Typus).
- Evaluierung durch Instruktor: Das System soll ein einfaches Interface bieten um eine Bewertung vorzunehmen
- Gruppenbewertung / Kommentarfunktion: Das System soll anderen Kandidaten ermöglichen die Textproduktionen anderer Kollegen zu kommentieren.

Ergebnisse betrachten

- Sicht des Kandidaten: Das System soll ein benutzerfreundliches Interface bieten um die eigene Test / Übungsergebnisse zu betrachten.
- Sicht des Instructors: Das System soll ein benutzerfreundliches Interface bieten um die Tests / Übungsergebnisse aller Kandidaten zu betrachten.
- Selbstbewertungsbogen: Das System soll die Möglichkeit eines Selbstbewertungsbogens bieten.

6 Anforderungsanalyse SprichWort-Community

6.1 Zielsetzung der SprichWort-Community

Die SprichWort-Community verbindet die ersten zwei Module und bietet die Funktionen zur Förderung einer Community an. Dabei werden servicefähige Funktionen wie Diskussionsforen, Annotationen, Kommentare, Tagging, Verlinkung, Erstellen der Inhalte, Datenaustausch unterstützt.

Das Hauptziel dieser Aspekte ist es, eine Plattform für europaweite sprach- und sprachdidaktische Community zum Austauschen und Auffinden von Ideen, Konzepten und Inhalten in diesem Bereich zu schaffen.

6.2 Funktionale Anforderungen

6.2.1 Diskussionsforum

Das System soll eine Möglichkeit zur strukturierten Diskussion für beliebige, darunter auch anonyme, Benutzer zur Verfügung stellen. Da die Anforderungen an ein Web-basiertes Diskussionsforum mittlerweile allgemein bekannt sind wird in diesem Dokument nicht näher darauf eingegangen.

6.2.2 Annotationen und Kommentare

Es muss jedem Benutzer ermöglicht werden jeden Beitrag im System zu kommentieren bzw. zu annotieren.

6.2.3 Tagging Funktionalität

Jeder Beitrag im System soll (extern) getaggt werden können. Dies entspricht der Anforderung dass jeder Beitrag im System eine eindeutige URL Adresse erhalten soll.

6.2.4 Erstellen der Inhalte

Jeder Benutzer soll in einem speziellen Bereich des Systems die Möglichkeit haben eigene Beiträge zu erstellen und zu verwalten. Dabei gelten dieselben Anforderungen für das Erstellen bzw. Editieren von Beiträgen wie bei den Editoren bzw. Mitglieder des Projektteams.

6.2.5 Datenaustausch

Es soll möglich sein die Inhalte leicht mit anderen Systemen bzw. Communities auszutauschen. Dabei soll das System in der Lage sein die Inhalte in verschiedenen Formaten wie PDF, HTML oder auch XML per Knopfdruck zu exportieren.