



**-Systemspezifikationen: Gesamtsystemspezifikation (Pflichtenheft)-
ATUIN - ATUIN the unique innovation network**

Version: 1.0

Projektbezeichnung	Problemlösungsnetzwerk	
Projektleiter	Denis Brumann	
Verantwortlich	Robert Kandzia	
Erstellt am	13.05.2007	
Zuletzt geändert	13.05.2007	
Bearbeitungszustand	x	in Bearbeitung
		vorgelegt
		fertig gestellt
Dokumentablage	CVS: pcai042.informatik.uni-leipzig.de/tr-07-2	
V-Modell-XT Version	Version 1.2.0	

Weitere Produktinformationen

Mitwirkend	Denis Brumann [nicht beteiligt] [nicht beteiligt]	Projektleiter Projektmanager Anwender
Erzeugung	Initial	

Änderungsverzeichnis

Änderung			Geänderte Kapitel	Beschreibung der Änderung	Autor	Zustand
Nr.	Datum	Version				
1		1.1	Alle	Initiale Produkterstellung		

Prüfverzeichnis

Die folgende Tabelle zeigt einen Überblick über alle Prüfungen – sowohl Eigenprüfungen wie auch Prüfungen durch eigenständige Qualitätssicherung – des vorliegenden Dokumentes.

Datum	Geprüfte Version	Anmerkungen	Prüfer	Neuer Produktzustand

Inhaltsverzeichnis

1. Einleitung
 - Zielbestimmung
2. Ausgangssituation und Zielsetzung
3. Funktionale Anforderungen
 - Produktleistungen
 - Produktdaten
4. Nicht-funktionale Anforderungen
 - Performanceanforderungen
 - Qualitätsanforderungen
5. Lebenszyklus und Gesamtsystemarchitektur
6. Lieferumfang
 - Abnahmekriterien
7. Glossar
8. Literaturverzeichnis
9. Abbildungsverzeichnis

1 Einleitung

Die Gesamtsystemspezifikation (Pflichtenheft) ist das Pendant zu dem Auftraggeberprodukt Anforderungen (Lastenheft) auf Auftragnehmerseite. Sie wird vom Auftragnehmer in Zusammenarbeit mit dem Auftraggeber erstellt und stellt das zentrale Ausgangsdokument der Systemerstellung dar.

Wesentliche Inhalte der Gesamtsystemspezifikation sind die funktionalen und nicht-funktionalen Anforderungen an das zu entwickelnde Gesamtsystem. Die Anforderungen werden aus den Anforderungen (Lastenheft) übernommen und geeignet aufbereitet. Eine erste Grobarchitektur des Systems wird entwickelt und in einer Schnittstellenübersicht beschrieben. Das zu entwickelnde System sowie weitere zu entwickelnde Unterstützungssystem werden identifiziert und den Anforderungen zugeordnet. Zusätzliche Anforderungen an die Logistik werden in Zusammenarbeit mit dem Logistikverantwortlichen erarbeitet. Abnahmekriterien und Lieferumfang für das fertige Gesamtsystem werden aus den Anforderungen (Lastenheft) übernommen und konkretisiert. Um sicher zu stellen, dass alle Anforderungen berücksichtigt sind, wird eine Anforderungsverfolgung, sowohl hin zu den Anforderungen (Lastenheft) als auch zu System und Unterstützungssystemen, durchgeführt.

Zur Erstellung der Gesamtsystemspezifikation (Pflichtenheft) sind Kenntnisse aus unterschiedlichen Disziplinen wie Systementwicklung, Systemsicherheit, Ergonomie und Logistik notwendig, die üblicherweise nicht von einer Person abgedeckt werden können. Da Anforderungen den Kern der Spezifikation darstellen, fällt dem Anforderungsanalytiker (AN) die verantwortliche Rolle für die Erstellung der Gesamtsystemspezifikation zu. Für die inhaltliche Ausarbeitung benötigt er jedoch intensive Unterstützung durch Experten der verschiedenen Disziplinen.

Zu jedem in der Gesamtsystemspezifikation identifizierten System, Unterstützungssystem und Segment werden die entsprechenden Produkte wie Spezifikation und Architektur erstellt. Anforderungen an die Logistik werden in der Spezifikation logistische Unterstützung weiter verfolgt.

Zielbestimmung

Die Anwendung soll dem Benutzer das Einstellen, Editieren und Löschen sowie Verknüpfen mit internen sowie externen Inhalten – genauer Problemstellungen – mittels einer internetbasierten Applikation ermöglichen. Die Bedienung erfolgt für alle Benutzer einfach und komfortabel über einen Webbrowser.

Die Wahl des Formats zur Datenspeicherung sowie des Datenaustauschs innerhalb und außerhalb des Systems, soll eine maschinenbasierte Klassifikation und Organisation der gesammelten Informationen ermöglichen und dem Nutzer diese in einer anwenderfreundlichen Form aufbereiten und darstellen. Die Grundlage bilden dabei RSS 1.0 und RDF.

Das System folgt einer stark kollaborativ orientierten Philosophie. Durch die Abgabe und Speicherung von Kommentaren und Lösungsvorschlägen soll eine Diskussion der beteiligten Nutzer im iterativen Prozess der Problemlösung ermöglicht werden.

Jeder Benutzer legt seine Kennung durch eigenständiges Registrieren selbst fest, für nicht registrierte Nutzer ist ein Gastzugriff vorgesehen, der jederzeit in eine registrierte Kennung verwandelt werden kann. Mit der gewählten Kennung kann sich der Benutzer jeder Zeit am System anmelden.

Alle Nutzer der Anwendung können unabhängig von ihrer Rolle:

- Problem/Lösungsvorschlag einsenden
 - editieren
 - löschen
- Kommentar einsenden
 - löschen
- sich Anmelden/Abmelden
- Volltextsuche betreiben oder
- anhand von Metainformationen suchen

Jeder Artikel ist öffentlich einsehbar. Die Verwaltung der vorhandenen Artikel erfolgt über die Benutzergemeinschaft sowie durch Administratoren, die über uneingeschränkte Rechte bei der Bearbeitung verfügen. Administratoren können sowohl einzelne Benutzeraccounts als auch Rollen, z.B. Gäste im Zugriff auf die Artikel beschränken oder ganz ausschließen.

Um Spam zu vermeiden, muss beim Erstellen eines Benutzeraccounts ein Captcha-Riddle gelöst werden.

Einzelne Artikel werden von der Benutzergemeinde verwaltet, d.h. ein anfangs unstrukturiertes, meist knapp formuliertes Problem wird über die Zeit in einer Vielzahl von Iterationen spezifiziert, so dass im Laufe der Zeit ein wachsender Verfeinerungsgrad erreicht wird. Im System werden Kommentare und Lösungsvorschläge unterschieden. Lösungsvorschläge gehören dabei ebenfalls der Klasse Problem an, bilden also wiederum eigene Artikelseiten mit jeweils einem Thema bzw. Problem im System.

Es ist möglich RSS-Feeds aus Fremdsystemen als Lösungsvorschlag/Problem zu integrieren. Neben starker interner Vernetzung, entsteht so eine Vernetzung auch nach außen.

Das System soll, die über die Zeit gewonnen Themen ordnen, bewerten, das Durchsuchen vereinfachen, sowie deren Verknüpfung mit in Zusammenhang stehenden internen und externen Inhalten über standardisierte Schnittstellen ermöglichen. Ziel ist die Schaffung eines Systems zur Unterstützung fortschreitender Wissensbildungsprozesse.

Die Entwicklung des Systems soll, nach erfolgreichem Abschluss des Projekts, wenn möglich als OpenSource Projekt weitergeführt werden.

2 Ausgangssituation und Zielsetzung

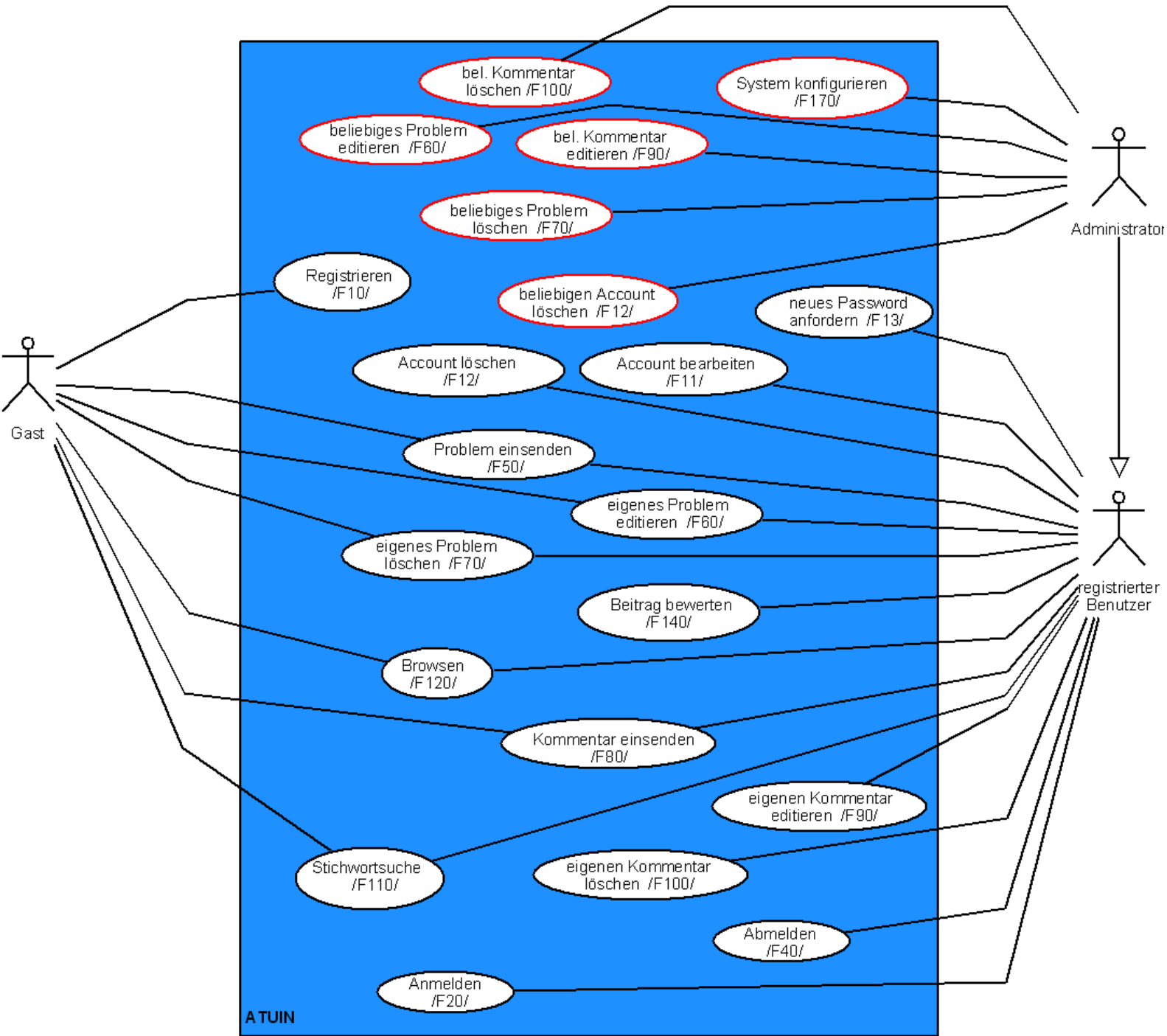
Mittels des Systems kann sich eine Benutzergemeinde über das Internet der Lösung von formulierten Problemstellungen widmen. Es wird größter Wert auf Zusammenarbeit gelegt. Benutzer sollen in einem kollaborativen Prozess verschiedenartige Problemstellungen detaillieren, verknüpfen und somit ein Informations-Netzwerk aufbauen auf Grundlage offener Internet-Standards. Zum Einsatz sollen dabei RSS 1.0 und RDF kommen. Das System übernimmt dabei die Aufbereitung, Speicherung und Darstellung der gewonnenen Inhalte und Metadaten sowie deren Verknüpfungen.

Dabei erfolgt keine Festlegung auf einen speziellen Themenbereich. Das System arbeitet gegenüber den konkreten Inhalten neutral. Die Wahl der Problemstellung treffen die Anwender. Anfangs erfolgt eine Konzentration auf textbasierte Inhalte, die aber in späteren Entwicklungsstufen auf andere Typen, wie z.B. Audio- oder Videoinhalte und deren Metainformationen erweitert werden können.

Durch Internationalisierung und Erweiterbarkeit soll langfristig ein weltweiter Einsatz des Systems möglich sein, anfangs ist als einzige Verkehrssprache Deutsch festgelegt.

Die Verwaltung erfolgt weitestgehend durch die Benutzer selbst mittels eines auf starker Zusammenarbeit basierenden Prozesses. Zusätzlich sind zur Verwaltung und Verbesserung des Systems Administratoren vorgesehen, die wie normale Benutzer auch Artikel publizieren. Administratoren achten auf Einhaltung der vorgegebenen Rahmenbedingungen und besitzen umfangreiche Rechte und Möglichkeiten der Nutzer- und Themenverwaltung. Außerdem sind sie für die technische Wartung und Fehlerbehandlung im System verantwortlich.

3 Funktionale Anforderungen



Produktleistungen

Nutzerverwaltung:

/F10/ (/LF10/)

Anwendungsfall:	Erstellen eines Benutzeraccounts (Registrieren)
Ziel:	Anlegen eines neuen Benutzeraccounts
Kategorie:	primär
Vorbedingung:	-
Nachbedingung Erfolg:	Benutzeraccount mit den eingegebenen Daten ist erstellt
Nachbedingung Fehlschlag:	Benutzeraccount ist nicht erstellt
Akteure:	beliebiger Nutzer
Auslösendes Ereignis:	-
Beschreibung:	<ol style="list-style-type: none">1. Benutzer trägt die geforderten Daten in ein Formular ein<ol style="list-style-type: none">1.1 Captcha-Riddle um Spam zu vermeiden2. Ausgefülltes Formular wird abgesendet3. Es erfolgt eine Rückmeldung über Erfolg / Misserfolg

/F11/

Anwendungsfall:	Benutzeraccount bearbeiten
Ziel:	Änderung der Accountdaten
Kategorie:	primär
Vorbedingung:	Account existiert und Benutzer hat Zugriff
Nachbedingung Erfolg:	Daten geändert
Nachbedingung Fehlschlag:	Daten unverändert
Akteure:	angemeldeter Benutzer
Auslösendes Ereignis:	-
Beschreibung:	<ol style="list-style-type: none">1. neue Daten in Formular eintragen2. Formular absenden3. Änderung der Daten oder Fehler<ol style="list-style-type: none">4.1 Wenn erfolgreich geändert: Account anzeigen4.2 Sonst: Fehlermeldung anzeigen

/F12/

Anwendungsfall:	Benutzeraccount löschen
Ziel:	Account aus dem System löschen
Kategorie:	primär
Vorbedingung:	Account existiert und Benutzer hat Zugriff
Nachbedingung Erfolg:	Account gelöscht
Nachbedingung Fehlschlag:	Account nicht gelöscht
Akteure:	angemeldeter Benutzer
Auslösendes Ereignis:	-
Beschreibung:	<ol style="list-style-type: none">1. Sicherheitsabfrage<ol style="list-style-type: none">2.1 Wenn bestätigt: Account löschen2.2 Sonst: Abbruch, Account anzeigen3 Account löschen oder Fehler<ol style="list-style-type: none">4.1 Wenn erfolgreich gelöscht: Startseite anzeigen4.2 Sonst: Fehlermeldung anzeigen

/F13/

Anwendungsfall: neues Benutzerpasswort generieren
Ziel: neues Passwort generieren, falls das alte vergessen wurde
Kategorie: primär
Vorbedingung: existierender Account und bekannte Email-Adresse
Nachbedingung Erfolg: Bestätigungsmail an die Email-Adresse des Nutzers
Nachbedingung Fehlschlag: Hinweis, dass keine Mail zugestellt wurde
Akteure: registrierter, aber nicht angemeldeter Benutzer
Auslösendes Ereignis: -
Beschreibung:
1.1 Eingabe der Email-Adresse in ein Formular
1.2 es erfolgt eine Weiterleitung auf die Sarteite
3. Email mit neuem Passwort versenden

/F20/(LF20/)

Anwendungsfall: manuelle Anmeldung am System
Ziel: Benutzer wird gegenüber dem System authentifiziert
Kategorie: primär
Vorbedingung: Benutzer ist registriert
Nachbedingung Erfolg: Benutzer ist am System angemeldet
Nachbedingung Fehlschlag: Benutzer ist nicht angemeldet
Akteure: registrierter Benutzer
Auslösendes Ereignis: -
Beschreibung:
1. Benutzer trägt Accountnamen und Passwort in Formular ein
2. Daten werden an den Server gesendet
3. Es erfolgt eine Bestätigung der Anmeldung bzw. eine Fehlermeldung

/F30/

Anwendungsfall: automatische Anmeldung am System
Ziel: Benutzer wird gegenüber dem System authentifiziert
Kategorie: primär
Vorbedingung: Benutzer ist registriert, automatische Anmeldung ist aktiviert
Nachbedingung Erfolg: Benutzer ist am System angemeldet
Nachbedingung Fehlschlag: Benutzer ist nicht angemeldet
Akteure: Benutzerseitiges Programm (Internet-Browser)
Auslösendes Ereignis: URI des Systems wird aufgerufen
Beschreibung:
1. Benutzer greift im Browser auf URI des Systems zu
2. Cookie wird ausgelesen
3. Benutzer wird angemeldet

/F40/ (LF30/)

Anwendungsfall: Abmeldung vom System
Ziel: Benutzer vom System abmelden
Kategorie: primär
Vorbedingung: Benutzer ist am System angemeldet
Nachbedingung Erfolg: Benutzer ist abgemeldet
Nachbedingung Fehlschlag: Benutzer ist nicht abgemeldet
Akteure: angemeldeter Benutzer
Auslösendes Ereignis: -
Beschreibung:
1. Benutzer abmelden
2. Bestätigungsmeldung

Interaktion:

/F50/ (/LF40/)

Anwendungsfall: Problem erstellen
Ziel: Problem im System einstellen
Kategorie: primär
Vorbedingung: -
Nachbedingung Erfolg: Problem ist erstellt
Nachbedingung Fehlschlag: Problem ist nicht erstellt
Akteure: Benutzer
Auslösendes Ereignis: -
Beschreibung:

1. Problem Daten in Formular eintragen
2. Formular absenden
3. Stichwortsuche (**/F110/**) anhand der Problembeschreibung
- 4.1 Wenn passendes Problem gefunden: Abbruch, dieses Problem anzeigen
- 4.2 Sonst: Problem wird erstellt oder Fehler
- 5.1 Wenn erfolgreich erstellt: Problem anzeigen
- 5.2 Sonst Fehlermeldung anzeigen
- 6 Wenn Problem als Lösung erstellt: im Feed des (gelösten) Problems eintragen

/F60/ (/LF40/)

Anwendungsfall: Problem bearbeiten
Ziel: existierendes Problem bearbeiten
Kategorie: primär
Vorbedingung: Problem existiert, Benutzer hat es selbst erstellt
Nachbedingung Erfolg: Problem ist geändert
Nachbedingung Fehlschlag: Problem ist nicht geändert
Akteure: registrierter Benutzer
Auslösendes Ereignis: -
Beschreibung:

1. Problem auswählen
2. neue Problem Daten in Formular eintragen
3. Formular absenden
4. Problem wird geändert oder Fehler
5. Problem anzeigen

/F70/ (/LF40/)

Anwendungsfall: Problem löschen
Ziel: existierendes Problem aus System löschen
Kategorie: primär
Vorbedingung: Problem existiert, Benutzer hat es selbst erstellt
Nachbedingung Erfolg: Problem ist gelöscht
Nachbedingung Fehlschlag: Problem ist nicht gelöscht
Akteure: Benutzer
Auslösendes Ereignis: -
Beschreibung:

1. Problem auswählen
2. Sicherheitsabfrage
- 3.1 Wenn bestätigt: Problem löschen
- 3.2 Sonst Abbruch: Problem anzeigen
- 4 Problem löschen oder Fehler
- 5.1 Wenn gelöscht: Bestätigung anzeigen
- 5.2 Sonst Fehlermeldung ausgeben, Problem anzeigen

/F80/ (/LF60/)

Anwendungsfall: Kommentar erstellen
Ziel: Kommentar an ein Problem anhängen
Kategorie: primär
Vorbedingung: Problem existiert
Nachbedingung Erfolg: Kommentar ist erstellt
Nachbedingung Fehlschlag: Kommentar ist nicht erstellt
Akteure: Benutzer
Auslösendes Ereignis: -
Beschreibung:
1. Problem auswählen
2. Kommentar in Formular eintragen
3. Formular absenden
4. Kommentar wird erstellt oder Fehler
5.1 Wenn erstellt: Problem und Kommentare anzeigen
5.2 Sonst: Fehlermeldung anzeigen

/F90/ (/LF60/)

Anwendungsfall: Kommentar bearbeiten
Ziel: Kommentartext ändern
Kategorie: primär
Vorbedingung: Kommentar existiert und ist in aktueller Session erstellt
Nachbedingung Erfolg: Kommentartext ist geändert
Nachbedingung Fehlschlag: Kommentartext nicht geändert
Akteure: Benutzer
Auslösendes Ereignis: -
Beschreibung:
1. Kommentar auswählen
2. neuen Text in Formular eintragen
3. Formular absenden
4. Kommentar wird geändert oder Fehler
5.1 Wenn geändert: Problem und Kommentare
5.2 Sonst: Fehlermeldung anzeigen

/F100/ (/LF60/)

Anwendungsfall: Kommentar löschen
Ziel: Kommentar aus dem System löschen
Kategorie: primär
Vorbedingung: Kommentar existiert
Nachbedingung Erfolg: Kommentar ist gelöscht
Nachbedingung Fehlschlag: Kommentar nicht gelöscht
Akteure: Benutzer
Auslösendes Ereignis: -
Beschreibung:
1. Kommentar auswählen
2. Sicherheitsabfrage
3.1 Wenn bestätigt: löschen
3.2 Sonst: Abbruch, Kommentar anzeigen
4. Kommentar wird gelöscht oder Fehler
5.1 Wenn gelöscht: Problem und Kommentare
5.2 Sonst: Fehlermeldung anzeigen

Navigation und Strukturierung:

/F110/ (/LF70/)

Anwendungsfall:	Problemstellungen nach Stichwörtern durchsuchen
Ziel:	alle Probleme finden, die eingegebene Stichwörter enthalten
Kategorie:	primär
Vorbedingung:	-
Nachbedingung Erfolg:	gefundene Probleme werden angezeigt
Nachbedingung Fehlschlag:	Suche kann nicht durchgeführt werden
Akteure:	Benutzer
Auslösendes Ereignis:	-
Beschreibung:	<ol style="list-style-type: none">1. Stichwörter angeben2. alle Problembeschreibungen werden durchsucht oder Fehler3.1 Wenn Suche durchgeführt: gefundene Probleme anzeigen3.2 Sonst: Fehlermeldung anzeigen

/F120/ (/LF80/)

Anwendungsfall:	Browsen im System
Ziel:	manuelle Traversierung der Beiträge
Kategorie:	primär
Vorbedingung:	-
Nachbedingung Erfolg:	gewählter Beitrag wird angezeigt
Nachbedingung Fehlschlag:	gewählter Beitrag kann nicht angezeigt werden
Akteure:	Benutzer
Auslösendes Ereignis:	-
Beschreibung:	<ol style="list-style-type: none">1. Beitrag wählen durch 'anklicken' oder direkt über URI2. zugehörige Datei wird geladen und geparkt3.1 Wenn erfolgreich geladen: Beitrag anzeigen3.2 Sonst: Fehlermeldung anzeigen

/F130/

Anwendungsfall:	Logfile verwalten
Ziel:	Änderungen und Nutzung des Systems protokollieren
Kategorie:	primär
Vorbedingung:	-
Nachbedingung Erfolg:	Änderung wird im Logfile vermerkt
Nachbedingung Fehlschlag:	Änderung nicht vermerkt
Akteure:	-
Auslösendes Ereignis:	beliebige Änderung am System (außer am Logfile) oder Fehler
Beschreibung:	<ol style="list-style-type: none">1. Änderung in Protokolleintrag parsen2.1 Wenn aktuelles Logfile zulässige Größe nicht überschreitet: auswählen2.2 Sonst: neues Logfile anlegen: auswählen3. Eintrag an Inhalt des gewählten Logfiles anfügen

/F140/ (/LF120/)

Anwendungsfall:	Beiträge evaluieren
Ziel:	Bewertung eines Beitrages
Kategorie:	primär
Vorbedingung:	Beitrag existiert
Nachbedingung Erfolg:	Bewertung des Beitrags hinzugefügt
Nachbedingung Fehlschlag:	Bewertung nicht vorgenommen

Akteure: angemeldeter Benutzer
Auslösendes Ereignis: -
Beschreibung:
1. Beitrag auswählen
2. Bewertung auswählen
3. Bewertung wird gesetzt oder Fehler
4.1 Wenn erfolgreich bewertet: Beitrag anzeigen
4.2 Sonst: Fehlermeldung anzeigen

/F150/ (/LF120/)

Anwendungsfall: automatische Evaluierung
Ziel: Bewertung der Aktualität eines Beitrages
Kategorie: primär
Vorbedingung: Beitrag existiert
Nachbedingung Erfolg: Bewertung des Beitrags hinzugefügt
Nachbedingung Fehlschlag: Bewertung nicht vorgenommen
Akteure: -
Auslösendes Ereignis: URI des Beitrags wird aufgerufen
Beschreibung:
1. Zugriff auf den Beitrag
2. Zugriffszähler wird erhöht

/F160/

Anwendungsfall: Captcha-Riddle
Ziel: Vermeidung von Spam-Accounts
Kategorie: primär
Vorbedingung: Benutzer führt /F10/ aus
Nachbedingung Erfolg: Misserfolg einer automatisierter Registrierung
Nachbedingung Fehlschlag: Erfolg einer automatisierter Registrierung
Akteure: beliebiger (nicht registrierter) Nutzer
Auslösendes Ereignis: /F10/
Beschreibung:
1. Anzeigen eines Captcha-Riddles (z.B. einfache Rechenaufgabe)
2. Lösen des Riddles durch den Akteur
2.1 Wenn richtig: Registrierung am System wird gewährt
2.2 Sonst: Anzeigen eines neuen Riddles

/F170/

Anwendungsfall: System konfigurieren
Ziel: Anpassung der Einstellungen des Systems
Kategorie: primär
Vorbedingung: -
Nachbedingung Erfolg: Einstellungen des Systems geändert
Nachbedingung Fehlschlag: Einstellungen unverändert
Akteure: Administrator
Auslösendes Ereignis: -
Beschreibung:
1. Einstellungen ändern
2. Einstellungen speichern
3.1 Wenn erfolgreich gespeichert: System mit neuen Einstellungen anzeigen
3.2 Sonst: Fehlermeldung anzeigen

Die Funktionen **/F111/**, **/F60/**, **/F70/**, **/F90/**, **/F100/** stehen dem Administrator ohne Einschränkungen zur Verfügung.

Produktdaten

/D10/

Die einzelnen Seiten liegen als XML-Dateien (RDF-Format) vor (max. Größe 1MB)

/D20/

Kommentare sind auf 255 Zeichen beschränkt, werden als ASCII-Text gespeichert (Markups)

/D30/

Logfiles, z.B. zum Auswerten der Zugriffe auf einzelne Seiten (max. Größe 1MB)

/D40/

Benutzerdatensätze liegen als verschlüsselte Dateien vor (max. Größe 20kB)

4 Nicht-Funktionale Anforderungen

- Administrationsbereich:
 - einfach zu bedienen
 - einfach einzurichten
- soll auf einem PDA darstellbar sein
- das Erscheinungsbild soll veränderbar sein (CSS-Dateien)
- modulare Architektur des Systems
- zukünftiges PlugIn-System (Tagging, Migration einzelner Themen(-bereiche) in neues System)

Performanceanforderungen

/L10/

Das Laden der Seite soll weniger als 2 Sekunden dauern (je nach Netzlast und Verbindung kann dies jedoch variieren).

/L20/

Die Funktionen **/F50/** - **/F70/** sollten in weniger als 3 Sekunden (je nach Netzlast und Verbindung kann dies jedoch variieren) aufrufbar und ausführbar sein.

/L30/

Die Funktion **/F110/** sollte maximal 20 Sekunden dauern.

/L40/

Der Zugriff auf History und Logfiles sollte maximal 20 Sekunden dauern

/L50/

Die Session soll durch einen permanenten Cookie auch mehrere Tage aktiv sein.

/L60/

Es können maximal 1000 Benutzer gleichzeitig die Plattform nutzen.

Qualitätsanforderungen

Produktqualität	sehr gut	gut	normal	nicht relevant
Funktionalität Angemessenheit Richtigkeit Interoperabilität Ordnungsgemäßheit Sicherheit			X X X X X	
Zuverlässigkeit Reife Fehlertoleranz Wiederherstellbarkeit		X	X X	
Benutzbarkeit Verständlichkeit Modifizierbarkeit Stabilität Prüfbarkeit	X X X	X		
Effizienz Zeitverhalten Verbrauchsverhalten			X X	
Änderbarkeit Analysierbarkeit Modifizierbarkeit Stabilität Prüfbarkeit Übertragbarkeit	X X X	X X		
Anpassbarkeit Installierbarkeit Konformität Austauschbarkeit	X X	X		

5 Lebenszyklus und Gesamtsystemarchitektur

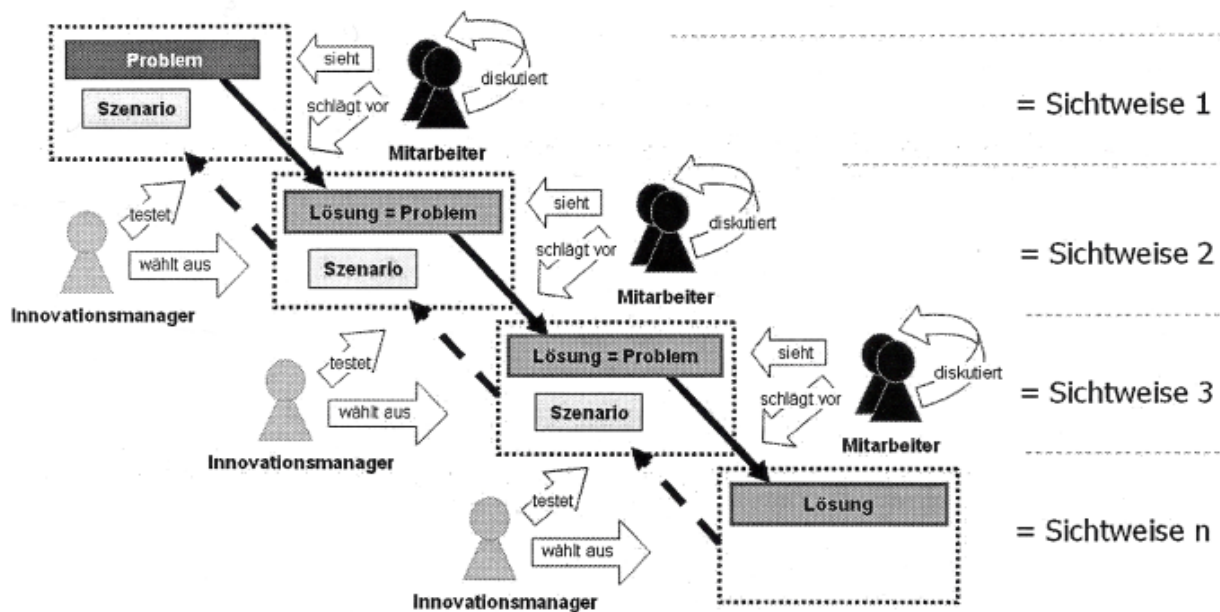
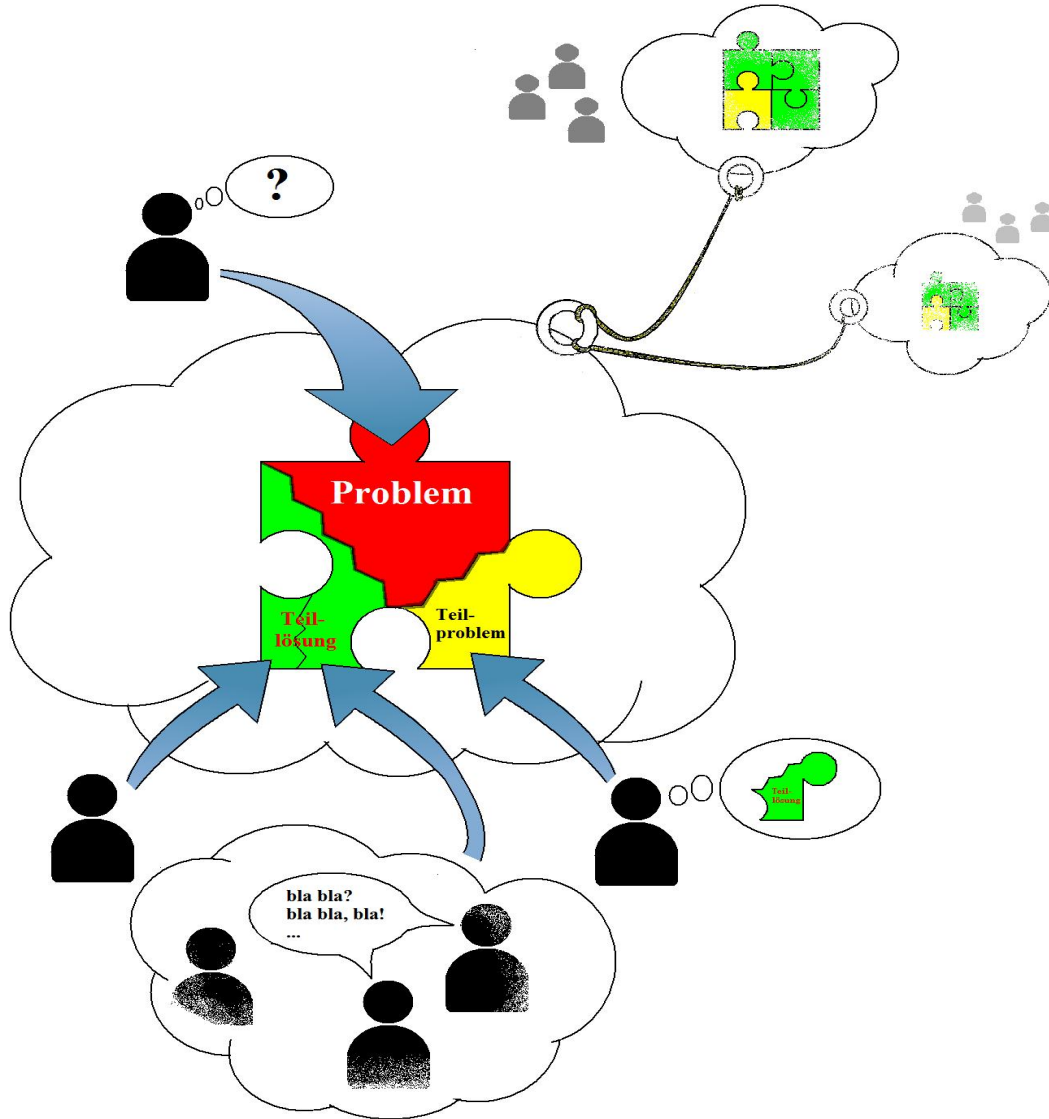


Abbildung 2: Problem-Lösungsansatz

6 Lieferumfang

Im Lieferumfang des Produktes ist ein Installationspaket enthalten. Des Weiteren wird eine umfassende Dokumentation zur Verfügung stehen, ebenso ein Leistungsnachweis aller am Projekt Beteiligten.

Abnahmekriterien

Um abgenommen zu werden, muss das Projekt termingerecht geliefert werden, die geforderten Funktionalitäten beinhalten und alle Beteiligten einen entsprechenden Leistungsnachweis vorweisen.

Ebenso muss das Produkt mindestens 2 (zwei) Anwendern zum Testen vorgelegt werden, welche nicht zum Projektteam gehören. Deren Bewertung dient u.a. mit als Abnahmekriterium – je nach positiver oder negativer Produktbewertung.

Entsprechende Anpassungen zur Lauffähigkeit in der Referenzumgebung gehören zum Leistungsumfang.

Sollte einer dieser Punkte nicht erfüllt (oder erfüllbar) sein, so wird das Produkt nicht abgenommen.

7 Glossar

Abkürzung	Erklärung
Account	Mit dem Begriff Account wird die passwortgeschützte Identität eines Benutzers bezeichnet. Der Account ist der Überbegriff für alle vom Nutzer eingegebenen Daten, mit denen er sich am System authentifiziert.
Administrator	Als Administrator wird ein Benutzer mit erweiterten Rechten im System bezeichnet. Diese Rechte sind im Bezug auf Wartung und Verwaltung eines Softwaresystems stark erweitert. So kann ein Administrator Benutzer anlegen/löschen/bearbeiten. Er darf auch Beiträge bearbeiten, welche nicht von ihm angelegt wurden. Das Konto eines Administrators sollte ausnahmslos nur für administrative (d.h. verwaltungstechnische und pflegerische) Tätigkeiten verwendet werden. Da der Administrator weitreichende Rechte in einem Softwaresystem hat, sollte sein Konto mit einem gut gesicherten Passwort versehen werden.
Akteur	Akteur wird als Oberbegriff im Lasten- und Pflichtenheft verwendet, um die an einzelnen Funktionen beteiligten Rollen aufzuzeigen.
Benutzer	Der Benutzer ist der Standardnutzer in einem Softwaresystem. Er kann Beiträge verfassen und auch auf diese Antworten. Der Benutzer registriert sich in der Regel in einem Softwaresystem. Dies erfolgt durch die Angabe eines gewünschten Nutzernamens, unter dem er dann Beiträge verfassen kann. Durch die Angabe eines Namens, wird es auch anderen Benutzern ermöglicht, mit dem Benutzer in Kontakt zu treten um weitere Informationen auszutauschen.
Benutzerdatensatz	Der Benutzerdatensatz umfasst den Log-In-Namen, das Passwort, die E-Mail-Adresse und weitere persönliche Daten, wie zum Beispiel weitergehende Kontaktinformationen oder Informationen zur Person. Diese Daten sind verschlüsselt abgelegt.
Captcha-Riddle	Captchas (<i>Completely Automated Public Turing test to tell Computers and Humans Apart</i>) werden verwendet, um zu entscheiden, ob das Gegenüber ein Mensch oder eine Maschine ist. Sie sollen im Internet sicherstellen, dass nur Menschen und keine programmierten Bots

	bestimmte Dienste nutzen können. Dabei wird eine Aufgabe (Riddle) gestellt, die für den Menschen einfach, für eine Maschine jedoch schwer zu lösen ist.
Content Syndication	Unter Content-Syndication versteht man die Mehrfachvererbung von Inhalten. In der Regel werden Daten übertragen, die dann von der Benutzerseite wieder aufbereitet werden können. Die zur Übertragung genutzten Sprachen sind üblicherweise XML und RSS. Das Übertragen von kompletten Artikeln wird auch als Artikel-Marketing bezeichnet.
Cookie	Ein Cookie ist eine kleine Datei, die von einem webbasierten Softwaresystem an den Browser des Nutzers gesendet wird. Die in der Datei abgelegten Informationen dienen in der Hauptsache dazu Besucher und Benutzer wiederzuerkennen. Diese Daten können zum Beispiel Informationen enthalten, wie der Benutzer sich das Softwaresystem eingerichtet hat (Layout, Sprache, etc.)
CSS	CSS (Cascading Style Sheets) werden vor allem zusammen mit HTML eingesetzt um festzulegen wie ein besonders ausgezeichneter Inhalt dargestellt werden soll. Neben diversen Fähigkeiten im Farb- und Schriftbereich bietet CSS die Möglichkeit, Elemente frei zu positionieren oder Hintergrundbilder festzulegen.
Gast	Ein Gast beschreibt einen Nutzer des Softwaresystems, der keine Anmeldedaten hinterlegt hat. Der Gast kann Antworten schreiben, seine Texte editieren und auch löschen. Die Zuordnung der Daten zu dem jeweiligen Gast erfolgt durch einen alphanumerischen Code. Es liegen keine weiteren Daten vom Gast vor, so dass er keine Möglichkeit hat über Änderungen informiert zu werden. Es sei denn, er hat seine E-Mail-Adresse mit angegeben.
HTML	Hypertext Markup Language (HTML) ist eine Auszeichnungssprache, die zur Darstellung von Text, Bildern und Hyperlinks in Dokumenten verwendet wird. HTML wurde aus SGML abgeleitet. HTML leidet unter der Eigenschaft, dass die einzelnen Browser bei der Interpretation von HTML nicht ganz dem Standard folgen. Somit ist es für einen Webentwickler unentbehrlich die von ihm entwickelten Dokumente in verschiedenen Browsern zu testen. Die aktuelle Version ist 4.01 und wurde vom W3C entworfen. HTML soll durch XHTML ersetzt werden.
Log (Logfile)	In einem Logfile legt das Softwaresystem Metainformationen zu seiner Bedienung ab. In einer solchen Datei werden in der Regel Informationen gespeichert, die Veränderungen in den einzelnen Texten aufzeigen, um diese auch eventuell wieder rückgängig machen zu können. Logfiles werden auch genutzt um eventuell auftretende Fehler zu dokumentieren.
OWL	OWL (Web Ontology Language) wurde von der W3C spezifiziert um eine Wissensrepräsentation in einer Beschreibungssprache zu ermöglichen. OWL ist Bestandteil des semantischen Web. Die technische Grundlage von OWL ist RDF, besitzt aber eine weitaus größere Mächtigkeit als RDFS bezüglich der erzeugbaren Ausdrücke. OWL erweitert das Konzept von RDFS um eine Beschreibungslogik, um die Defizite von RDFS zu beseitigen.
PHP	PHP ist eine an der Syntax von C/C++ angelehnte Skriptsprache. Die Bedeutung von PHP := PHP Hypertext Preprocessor (rekursives

	Bachronym). Mittels PHP lassen sich dynamische Webseiten erstellen, was fortschrittliche Webanwendungen ermöglichte wie z.B. CMS, Groupware und Internetforen.
Posting	Ein Posting beschreibt im allgemeinen einen Artikel. Hier wird der Begriff Posting für eine durch einen Benutzer/Gast gestellte/geschriebene Frage/Antwort verwendet.
RDF	RDF (Resource Description Framework) bezeichnet eine Sprache, mit der es möglich ist, Metadaten im Internet bereitzustellen. Ziel ist es die Daten in maschinenlesbarer Form darzustellen um damit eine automatisierbare Verarbeitung zu erreichen. RDF ist eine W3C Empfehlung.
RDFS	RDFS (Resource Description Framework Schema), ebenfalls eine W3C Empfehlung. RDFS legt ausschließlich die Syntax und das Vokabular fest, in der die Daten ausgetauscht werden sollen. Die grundlegende Idee hinter RDFS ist das mengentheoretische Klassenmodell. Die Klassen und die Eigenschaften werden bei RDFS getrennt modelliert.
Rolle	Die Rolle eines Akteurs beschreibt dessen Status im System und die damit verbundenen Rechte, mögliche Rollen sind (->) Administrator, registrierter Benutzer, (->) Gast. Durch eine Authentifizierung (i.A. mittels Passwort) kann ein Akteur seine Rolle und damit seine Rechte im System ändern.
RSS	RSS (Really Simple Syndication) ist ein elektronisches Nachrichtenformat, das dem Nutzer ermöglicht, die Inhalte einer Webseite – oder Teile davon – als sogenannte RSS-Feeds zu abonnieren oder in andere Webseiten zu integrieren. Im Rahmen dieses Projektes wird die RSS 1.0 Spezifikation verwendet. (http://web.resource.org/rss/1.0/spec)
Session	Unter einer Session (deutsch: Sitzung) versteht man eine stehende Verbindung zwischen Client (Benutzer) und Server, wobei die Session mit dem Anmelden am System beginnt und mit dem Abmelden endet.
W3C	W3C ist die Abkürzung für das World Wide Web Consortium. Die W3C wurde im Jahre 1994 gegründet, mit dem Ziel, Techniken für das World Wide Web zu standardisieren. Das W3C ist keine staatlich anerkannte Organisation, so dass sie nicht in der Lage ist, Industriestandards wie z.B. die ISO festzulegen. Im allgemeinen gibt das W3C lediglich Empfehlungen heraus, die aber so sehr beachtet werden, so dass das W3C viele so genannte de-facto Standards geprägt hat, wie z.B. XML.
XML	XML ist eine Abkürzung für eXtensible Markup Language. Entwickelt wurde XML von W3C, ein Web Konsortium, welchem große Firmen angehören (z.B. IBM). XML ist allerdings kein Standard sondern lediglich eine Empfehlung. XML ist eine Metasprache, d.h. sie beschreibt eine Sprache, mit der es möglich ist, weitere Sprachen zu beschreiben. Die Metasprache XML bildet eine Untermenge von SGML, wobei große Teile aus SGML nicht in XML übernommen worden sind.
Zend Framework	Ein Framework entwickelt von der Israelischen Firma Zend Technologies Ltd., die hauptverantwortlich für die Entwicklung der Skriptsprache PHP ist.

8 Literaturverzeichnis

- <http://wikipedia.org>
- <http://www.w3.org/>
- <http://www.php.net>
- <http://v-modell-xt.de>
- vorlesungsbegleitende Skripte zur Vorlesung Softwaretechnik SS07
- Rechercheberichte zur Aufgabenserie 3 (zu finden im internen Arbeitswiki auf: <http://stinfwww.informatik.uni-leipzig.de/~mam05dqx>)

9 Abbildungsverzeichnis

- Abb. 1: Skizze Problemlösungsnetzwerk (erstellt von Michael Leßnau)
- Abb. 2: Problem-Lösungsansatz (aus: Projektskizze: LeiKo)
- Abb. 3: Seitenhierarchie (erstellt von Martin Meißner)
- Abb. 4: externe Verlinkung (erstellt von Michael Leßnau)