

JeMiNa

Bibliothekverwaltung – System der FH Darmstadt

Versionsverlauf			
Wer	Wann	Was	Version
Nadine Thybusch Jens Giehl Mike Weis	10.11.2012	Erster Entwurf	1.0
Mike Weis	21.11.2012	Überarbeitet	1.1
JeMiNa	14.01.2013	Abnahme	

1. Einleitung

Für die FH Darmstadt wird eine neue Bibliothek-Verwaltungs-Software erstellt und eingeführt. Notwendig wurde das durch die steigende Zahl von Studenten und Mitarbeiter der FH die sich in der Bibliothek Bücher, Zeitschriften, CD und weitere Medien ausleihen können. Es gibt ein stetig steigendes Angebot an Medien welche ausgeliehen werden können. Medien welche im Zusammenhang mit aktuellen Themen stehen oder neu auf dem Markt erscheinen müssen schnellstmöglich den Nutzer zur Verfügung gestellt werden. Durch ein Vorschlagwesen das von bestimmten Benutzergruppen genutzt werden kann soll die Aktualität und der fachliche Bestand der Medien weiter verbessert werden.

Durch die hohe Frequentierung der Bibliothek ist eine effektive Kontrolle der entliehenen Medien notwendig. Das soll durch ein Mahnsystem welches Bestandteil der Software ist realisiert werden.

Der Nutzer hat die Möglichkeit Medien direkt vor Ort in der Bibliothek zu nutzen oder auszuleihen. Um sicher zu stellen, dass bei einem Besuch vor Ort die Medien vorhanden sind soll es einen Bestand an Medien geben welcher nicht ausgeliehen, sondern nur vor Ort genutzt werden kann.

Der Nutzer hat die Möglichkeit Medien Online zu Reservieren und unter Einhaltung einer Ausgabefrist seine reservierten Medien in der Bibliothek abzuholen.

Um für die Nutzer eine maximale Bedienerfreundlichkeit zu erreichen, soll das System Online erreichbar sein. Jeder kann auch ohne Anmeldung das System nutzen um einen Überblick über die angebotene Medien zu erhalten, jedoch kann er nichts Reservieren. Dadurch soll eine hohe Frequentierung der Bibliothek erreicht werden um neue Kunden zu gewinnen.

Der Nutzer kann Medien nur Ausleihen wenn er sich am System angemeldet hat. Dazu ist eine Registrierung notwendig. Registrieren können sich alle Hochschulangehörigen welche im Besitz einer ID-Karte sind.

Um den Administrativen Aufwand für den Zugang zum System gering zu halten wird sich der Nutzer über seine ID-Karte welche er bei der Studentenverwaltung bzw. Personalverwaltung erhalten hat Registrieren. Nur Daten welche sich auf die Rolle des Nutzers im System beziehen und welche für das Mahn- und Ausleihverfahren relevant sind werden gespeichert. Damit wird auch der Datenschutz berücksichtigt und gewährleistet.

Da das Onlinesystem auf einer Internet/Intranet Anwendung beruht bedarf es keiner Installation auf dem Kunden Computer. Der Kunde kann Medien bequem Suchen und Vorbestellen. Mitarbeiter die Medien pflegen haben einen erweiterten Zugang zum System. Sie sind in der Lage Medien anzulegen zu ändern und zu löschen. Außerdem können Sie Bestellungen der Kunden Ausführen und auf das Mahnsystem zugreifen.

Das bisherige System kann diese hohen Anforderungen nicht mehr standhaltend und ist permanent überlastet, wodurch es zu lange Wartezeiten bei den Bestellungen und Ausgabe kommt. Außerdem ist die Pflege des Artikelbestandes sehr aufwendig und schwierig. Zum jetzigen Zeitpunkt ist das nur mit hohem personellen Aufwand zu realisieren.

Stand heute ist, dass die Medien mit einem Karteikartensystem verwaltet werden. Dasselbe vorgeht wird für die Nutzer der Bibliothek angewendet. Das Mahnverfahren wird manuell durchgeführt. Eine Möglichkeit der Bibliothek Artikel vorzuschlagen gibt es nicht.

2. Funktionale und nicht funktionale Ziele:

2.1. Welche funktionalen Ziele wird das System nicht unterstützen.

- 2.1.1. Es wird keine Nutzerverwaltung geben. Die Nutzer der Software müssen sich mit ihre ID-Karte am System registrieren. Die ID-Karte hat jeder Mitarbeiter/Inn und Student/Inn der Hochschule.

2.2. Welche funktionalen Ziele wird das System unterstützen.

- 2.2.1. Medien Online über eine Weboberfläche suchen.
- 2.2.2. Medien Online über eine Weboberfläche reservieren.
- 2.2.3. Medien Online über eine Weboberfläche verwalten.
- 2.2.4. Mahnverfahren um ausstehende Medien zurück zu fordern.
- 2.2.5. Ein Vorschlagsystems um der Bibliothek neue Medien anzubieten.
- 2.2.6. Nutzer registrieren/authentifizieren sich über ihre ID-Karte.
- 2.2.7. Rollenverwaltung welche Funktionen des Systems für den Nutzer beschränkt oder erweitert.
- 2.2.8. ...

2.3. Welche nicht funktionalen Ziele wird das System unterstützen.

- 2.3.1. Durch einfache Bedienung wird der Zeitaufwand zur Pflege und Verwaltung der Artikel verbessert.
- 2.3.2. Durch Rollenbasierte Anmeldung wird die Funktionsvielfalt eingeschränkt indem die Grafische Oberfläche angepasst wird.
- 2.3.3. Einfache Bedienung des Systems.
- 2.3.4. Schnelles suchen und reservieren.
- 2.3.5. Aktuelle Daten zum Bestand der Medien.
- 2.3.6. Sicherheit der Nutzerdaten.

3. Anforderung

- 3.1. Suchen
 - 3.1.1. Artikel suchen (UC_D),(UC_UML)
 - 3.1.2. Artikel reservierbar
- 3.2. Verleihen
 - 3.2.1. Artikel reservieren (UC_D) ,(UC_UML)
 - 3.2.2. Artikel abholen (UC_D) ,(UC_UML)
 - 3.2.3. Artikel zurückgeben (UC_D) ,(UC_UML)
- 3.3. Bestandsverwaltung
 - 3.3.1. Artikel hinzufügen (UC_D) ,(UC_UML)
 - 3.3.2. Artikel ändern (UC_D) ,(UC_UML)
 - 3.3.3. Artikel löschen (UC_D) ,(UC_UML)
- 3.4. Vorschlagsystem
 - 3.4.1. Artikel Vorschlagen (UC_D) ,(UC_UML)
- 3.5. Nutzerverwaltung
 - 3.5.1. Nutzer Registrieren (UC_D) ,(UC_UML)
 - 3.5.2. Nutzer Anmelden (ZD_UML)
- 3.6. Abfragen für externe Systeme -> Studenten.-und Personalverwaltung
 - 3.6.1. Welche Medien sind von Nutzer ausgeliehen
(Exmatrikulation, Kündigung)
- 3.7. Synchronisieren
 - 3.7.1. Löschen von markierten Medien in der DB
 - 3.7.2. Versenden von Mahnungen
 - 3.7.3. Versenden von Erinnerungen zu rechtzeitigen Rückgabe
 - 3.7.4. Nicht rechtzeitig abgeholte Reservierungen in der DB freigeben
- 3.8. Ausblicke
 - 3.8.1. Barrierefrei. (Version 2)
 - 3.8.2. persönliche Programmeinstellungen speichern. (Version 2)
 - 3.8.3.
- 3.9. ...

4. Stackholder

Stackholder	Personen	Interesse
Mitarbeiter	Mitarbeiter der Bibliothek	<ul style="list-style-type: none"> - Mediendaten pflegen - Kundendaten pflegen
Nutzer	Dozenten Studenten Mitarbeiter	Der Bibliothek Medien im Vorschlagssystem bereitzustellen, welche sie als interessant und notwendig erachten.
Auftraggeber	Leitung	Ein an den Anforderungen der Bibliothek funktionierendes Software System bereitzustellen.
IT Abteilung	Personen welche das System IT-technisch Verwalten	<ul style="list-style-type: none"> - kontinuierliche Sicherung der persistenten Datenbestände - Zugriff Sicherung auf Daten - Zusicherung der Datenintegrität - Verfügbarkeit der Daten und Software - Wartung - Update
Nachbarsystem	Studentenverwaltung Mitarbeiterverwaltung	<ul style="list-style-type: none"> - Stellen Schnittstelle zum Prüfen der ID-Karte zur Verfügung. - Nutzen Schnittstelle zur Bibliotheksoftware um zu prüfen ob Personen, welche die FH verlassen, Medien entliehen haben.
Entwickler	Personen welche die Bibliotheksoftware programmieren	Entwickeln und programmieren alle Anforderungen an die Software.
Auftragnehmer	Personen welche das Projekt durchführen werden.	Dem Auftraggeber ein System <ul style="list-style-type: none"> - welches den Anforderungen des Kunden entspricht - innerhalb der Kosten - in der vorgegebenen Zeit zu liefern.

5. Glossar

Bezeichnung	Beschreibung
ID Karte	Ausweis mit dem man sich u.a. bei der Bibliothek authentifizieren kann. Dieser Ausweis besitzt jede Person die an der Hochschule Darmstadt angemeldet ist.
Weboberfläche	Eine Grafische Benutzeroberflächen, die in einem Internetbrowser läuft.
Online	Eine Anwendung die im Internet oder Intranet ausgeführt wird.
Intranet	Eine Anwendung die nur innerhalb einer Organisation verfügbar ist.
Rolle	Rechte welche der Nutzer im System hat. Funktionen die der Nutzer im System ausführen darf.
UC	Anforderung wurde in USECASE festgehalten.

USECASE

ID	Artikel suchen						
Beschreibung	Suchen eines Artikels in Datenbank mit Hilfe einer grafischen Benutzereingabe.						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.1.1						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Suchen						
Abhängigkeit	keine						
<i>Primary Actor</i>	Kunde						
<i>Further Actors</i>	Mitarbeiter						
<i>Stakeh & Inter</i>	Kunde: will seine Reservierung einfach/schnell ausführen Mitarbeiter: will Zentrale Verwaltung/Zeitersparnis						
<i>Success Guar</i>	Artikel gefunden						
<i>Minimal Guar.</i>	Artikel nicht vorhanden						
<i>Trigger</i>	Kunde sucht Artikel						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. System sucht Artikel 3. System gibt Informationen aus 						
<i>Alternative Course</i>	2a. Artikel kann nicht gefunden werden 2a1. System gibt Meldung aus: Grund für Abbruch 2a2. Abbruch						

ID	Artikel reservieren						
Beschreibung	Der gefundene Artikel soll reserviert werden.						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.2.1						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Verleihen						
Abhängigkeit	keine						
<i>Primary Actor</i>	Kunde						
<i>Further Actors</i>							
<i>Stakeh & Inter</i>	Kunde möchte sicher gehen das er den Artikel erhält.						
<i>Success Guar</i>	Artikel wird für Kunde reserviert						
<i>Minimal Guar.</i>	Kunde erhält Rückmeldung						
<i>Trigger</i>	Button „Artikel Reservieren“ gedrückt						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. System prüft ob Artikel ausgeliehen werden kann 3. System reserviert Artikel für Kunde 4. System gibt Informationen zur Abholung aus 						
<i>Alternative Course</i>	<p>2a. Artikel kann nicht reserviert werden</p> <p>2a1. System gibt Informationen (Reservierung nicht möglich) aus.</p> <p>2a2. Abbruch</p>						

ID	Artikel abholen						
Beschreibung	Artikel kann innerhalb der Reservierungsdauer abgeholt werden.						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.2.2						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Verleihen						
Abhängigkeit	keine						
<i>Primary Actor</i>	Mitarbeiter						
<i>Further Actors</i>	Kunde						
<i>Stakeh & Inter</i>	Kunde: möchte reservierten Artikel abholen Mitarbeiter: sicherstellen das Artikel nur an berechnigte Nutzer ausgeben werden.						
<i>Success Guar</i>	Kunde erhält reservierten Artikel						
<i>Minimal Guar.</i>	Kunde erhält Rückmeldung						
<i>Trigger</i>	Kunde authentifiziert sich mit ID-Karte gegenüber Mitarbeiter						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. Mitarbeiter ruft Kunde im System auf 3. System zeigt vorliegende Reservierungen 4. Mitarbeiter belastet Konto des Kunden 5. Mitarbeiter gibt Artikel aus 6. Reservierung wird gelöscht 7. System gibt Informationen aus 						
<i>Alternative Course</i>	3a. keine Reservierungen vorhanden 3a1. System gibt Informationen (Zeitüberschreitung der Abholung) 3a2. Abbruch						

ID	Artikel zurückgeben						
Beschreibung	Artikel wird innerhalb der Rückgabefrist zurückgegeben.						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.2.3						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Verleihen						
Abhängigkeit	keine						
<i>Primary Actor</i>	Kunde						
<i>Further Actors</i>	Mitarbeiter						
<i>Stakeh & Inter</i>	Kunde: möchte den zurückgegebenen Artikel von seinem Konto entlastet wissen. Mitarbeiter: will dass der Artikel wieder verliehen werden kann.						
<i>Success Guar</i>	Konto wird entlastet						
<i>Minimal Guar.</i>							
<i>Trigger</i>	Kunde authentifiziert sich mit ID-Karte gegenüber Mitarbeiter						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. Mitarbeiter ruft Kunde im System auf 3. Mitarbeiter entlastet Artikel vom Konto des Kunden 4. System prüft Rückgabefrist 5. System entlastet Konto 6. System gibt Artikel zum ausleihen frei 7. System gibt Informationen aus 						
<i>Alternative Course</i>	<ol style="list-style-type: none"> 4a. Rückgabefristen ist abgelaufen 4a1. System druckt Rechnung (Nachgebühr) 						

ID	Artikel hinzufügen						
Beschreibung	Artikel der Bibliothek hinzufügen						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.3.1						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Bestandsverwaltung						
Abhängigkeit	Eingabedaten prüfen						
<i>Primary Actor</i>	Mitarbeiter						
<i>Further Actors</i>							
<i>Stakeh & Inter</i>	Mitarbeiter will das Artikel verliehen werden können						
<i>Success Guar</i>	Artikel wird erfasst						
<i>Minimal Guar.</i>	Meldung ausgeben						
<i>Trigger</i>	Mitarbeiter öffnet neues Eingabeformular						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. Mitarbeiter gibt Daten ein 3. Mitarbeiter speichert Daten 4. System speichert Datensatz 5. System gibt Informationen aus 						
<i>Alternative Course</i>	<p>3a. System prüft Daten</p> <p>3a1. System erkennt das Daten nicht vollständig sind.</p> <p>3a2. System gibt Informationen über fehlende/falsche Daten aus.</p> <p>3a3. Rückkehr nach 2</p>						

ID	Artikel ändern						
Beschreibung	Änderungen an Artikeln können notwendig sein wenn Daten beim Anlegen falsch eingegeben wurden						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.3.2						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Bestandsverwaltung						
Abhängigkeit	Eingabedaten prüfen						
<i>Primary Actor</i>	Mitarbeiter						
<i>Further Actors</i>							
<i>Stakeh & Inter</i>	Mitarbeiter will das Artikel aktuell sind						
<i>Success Guar</i>	Datensatz wird geändert						
<i>Minimal Guar.</i>	Meldung ausgeben						
<i>Trigger</i>	Mitarbeiter sucht Artikel						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. Mitarbeiter gibt Daten ein 3. Mitarbeiter speichert Daten 4. System speichert Datensatz 5. System gibt Informationen aus 						
<i>Alternative Course</i>	<ol style="list-style-type: none"> 3a. System prüft Daten 3a1. System erkennt das Daten nicht vollständig sind. 3a2. System gibt Informationen über fehlende/falsche Daten aus. 3a3. Rückkehr nach 2 						

ID	Artikel löschen						
Beschreibung	Löschen von Artikeln wird ausgeführt wenn - Artikel nicht mehr zurückgeben werden können. - Inhalte nicht mehr aktuell sind - Artikel defekt sind						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.3.3						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Bestandsverwaltung						
Abhängigkeit	keine						
<i>Primary Actor</i>	Mitarbeiter						
<i>Further Actors</i>							
<i>Stakeh & Inter</i>	Mitarbeiter will das Artikel aktuell sind						
<i>Success Guar</i>	Datensatz wird gelöscht						
<i>Minimal Guar.</i>	Meldung ausgeben						
<i>Trigger</i>	Mitarbeiter sucht Artikel						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. Mitarbeiter löscht den Artikel 3. System markiert Datensatz zum löschen 4. System gibt Informationen aus 						
<i>Alternative Course</i>	<p>2a. System prüft ob Artikel gelöscht werden kann (hat Referenzen)</p> <p>2a1. Artikel kann nicht gelöscht werden</p> <p>2a2. System gibt Abbruchmeldung aus</p> <p>2a3. Abbruch</p>						

ID	Nutzer registrieren						
Beschreibung	Student / Dozenten im System registrieren						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.5.1						
Datum	10.12.2012	Version	1.0	Priorität	10	gelöscht	nein
Status	implementiert						
Kategorie	Nutzerverwaltung						
Abhängigkeit	Eingabedaten prüfen						
<i>Primary Actor</i>	Student						
<i>Further Actors</i>							
<i>Stakeh & Inter</i>	Studenten müssen im System registriert sein um es zu nutzen						
<i>Success Guar</i>	Student ist registriert und damit Kunde						
<i>Minimal Guar.</i>	Meldung ausgeben						
<i>Trigger</i>	Student authentifiziert sich mit ID Karte gegenüber Mitarbeiter						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. Mitarbeiter gibt Daten in Registrierungsformular ein 3. Mitarbeiter speichert Daten 4. System speichert Datensatz 5. System gibt Informationen aus 						
<i>Alternative Course</i>	<ol style="list-style-type: none"> 1a. System prüft Rolle <ol style="list-style-type: none"> 1a1. Rolle ist nicht Berechtig 1a2. System gibt Abbruchmeldung aus 1a3. Abbruch 3a. System prüft Daten <ol style="list-style-type: none"> 3a1. System erkennt das Daten nicht vollständig sind. 3a2. System gibt Informationen über fehlende/falsche Daten aus. 3a3. Rückkehr nach 2 						

Dieser UseCase wurde gestrichen, da die Benutzerverwaltung über ein bestehendes Fremdsystem (LDAP) realisiert wurde.

ID	Artikel vorschlagen						
Beschreibung	Bestimmte Nutzer können Artikel vorschlagen. Diese Nutzer sind geschult um Artikel professionell zu dokumentieren. Sie sollen die Mitarbeiter unterstützen um die Inhaltliche Qualität der Artikel in der Bibliothek zu sichern.						
Bemerkung	Bezieht sich auf Anforderung Punkt 3.4.1						
Datum	10.12.2012	Version	1.0	Priorität	10	Gelöscht	nein
Status	implementiert						
Kategorie	Vorschlagsystem						
Abhängigkeit	Eingabedaten prüfen						
<i>Primary Actor</i>	Nutzer						
<i>Further Actors</i>	Mitarbeiter						
<i>Stakeh & Inter</i>	Nutzer: Vorschlag in die Bibliothek einreichen Mitarbeiter: Inhaltliche Qualität der Artikel erhalten und verbessern.						
<i>Success Guar</i>	Vorschlag wurde an die Mitarbeiter weitergeleitet						
<i>Minimal Guar.</i>	Meldung ausgeben						
<i>Trigger</i>	Nutzer ruft Vorschlagsformular auf						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. Nutzer gibt Daten in Vorschlagsformular ein 3. Nutzer speichert Daten 4. System speichert Datensatz 5. System gibt Informationen aus 						
<i>Alternative Course</i>	<ol style="list-style-type: none"> 1a. System prüft Rolle <ol style="list-style-type: none"> 1a1. Rolle ist nicht berechtigt 1a2. System gibt Abbruchmeldung aus 1a3. Abbruch 3a. System prüft Daten <ol style="list-style-type: none"> 3a1. System erkennt das Daten nicht vollständig sind. 3a2. System gibt Informationen über fehlende/falsche Daten aus. 3a3. Rückkehr nach 2 						

ID	Eingabedaten prüfen						
Beschreibung	Prüft alle Daten welche in Formularen eingegeben werden						
Bemerkung	<include>						
Datum	10.12.2012	Version	1.0	Priorität	10	Gelöscht	nein
Status	implementiert						
Kategorie	Validierung						
Abhängigkeit	keine						
<i>Primary Actor</i>	Mitarbeiter						
<i>Further Actors</i>							
<i>Stakeh & Inter</i>	Daten müssen korrekt ans System übergeben werden						
<i>Success Guar</i>	Formulardaten sind korrekt						
<i>Minimal Guar.</i>	Meldung ausgeben						
<i>Trigger</i>	Daten werden gespeichert						
<i>Basic Course</i>	<ol style="list-style-type: none"> 1. Siehe Trigger 2. System prüft Daten 3. System speichert Daten 						
<i>Alternative Course</i>	2a. Daten sind nicht korrekt 1a1. Abbruch						

Allgemeine Erklärungen zur Umsetzung des Projekts:

1. Alle Anforderungen mit „(UC_D)“ wurden als UseCase beschrieben.
2. Alle Anforderungen mit „(UC_UML)“ wurden als UseCase gezeichnet.
3. Alle Anforderungen mit „(ZD_UML)“ wurden als Zustanddia. gezeichnet.
4. Wir haben uns für deutsche und englische Bezeichnungen entschieden.