

Inhaltsverzeichnis

1	Einführung	3
1.1	Grundlegendes	3
1.1.1	Betreuender Professor	3
1.1.2	Studenten	3
1.2	Motivation	4
1.3	Ideen zur Projektarbeit	4
1.3.1	Tippspiel	4
1.3.2	Videosoftware	4
1.4	Zielsetzung	5
2	Konzepte und Aufbau	6
2.1	Aufbau	6
2.1.1	Beschreibung des Gesamtsystems	6
2.1.2	das Kundenmodul im Detail	9
2.1.3	das Versandmodul im Detail	9
2.1.4	das Verwaltungsmodul im Detail	10
2.2	Einsatzbereich	10
2.3	verschiedene Versionen	11
3	Technologien	12
3.1	Datenbank	12
3.2	Versionsverwaltung	14
3.2.1	Concurrent Versions System - CVS	15
3.2.2	Subversion	15
3.3	Entwicklungsumgebung	15
3.3.1	JBuilder	15
3.3.2	Netbeans	15
3.3.3	Eclipse	15
3.4	grafischen Benutzerschnittstellen in Java	15
3.4.1	Abstract Window Toolkit - AWT	15
3.4.2	Swing	15

INHALTSVERZEICHNIS

3.4.3	Standard Widget Toolkit - SWT	15
3.5	Java-Web-Anwendungen	15
3.5.1	Java Server Faces - JSF	15
3.5.2	Struts	15
3.6	Persistenzschichten in Java	15
3.6.1	Java Data Objects	15
3.6.2	Hibernate	15
4	Implementierung	17
4.1	Versionsverwaltung mit Subversion	17
4.2	Entwicklungsumgebung mit Eclipse	17
4.3	grafischen Benutzerschnittstellen mit SWT	17
4.4	Java-Web-Anwendungen mit Struts	17
4.5	Persistenzschichten mit Hibernate	17
5	Zusammenfassung	18
A	Protokoll vom 11. Mai 2004	19
	Literaturverzeichnis	21

Kapitel 1

Einführung

1.1 Grundlegendes

1.1.1 Betreuender Professor

Hochschule Harz
Prof. Dr. Sigurd Günther
Friedrichstr. 57- 59
38855 Wernigerode
sguenther@hs-harz.de

1.1.2 Studenten

Remo Griesch	Stefan Forstner	Philipp Schneider
Strasse	Strasse der Jugend 22	Kastanienring 16
Ort	04880 Dommitsch	04316 Leipzig
Romeodied@gmx.de	fossiosi@web.de	provirent@phil-schneider.de

1.2 Motivation

Im Rahmen des Studiums an der Fachhochschule Harz in Wernigerode muß jeder Student des Studiengangs Kommunikationsinformatik eine Projektarbeit abgeben. Dies bedeutet, daß der Student eine Aufgabe (meist Programmieraufgabe) alleine oder in einem kleinen Team bewältigen muss. Die Professoren der Hochschule bieten dabei viele interessante Projektarbeiten an, sind jedoch offen für eigene Vorschläge der Studenten.

Da schon in den Teamprojekten¹ *Labmin*² und *German Team Sony Aibo*³ eine interessante Aufgabe von den Studenten gelöst wurde, sollte das dort erlernte Wissen vertieft und weiter ausgebaut werden.

1.3 Ideen zur Projektarbeit

1.3.1 Tippspiel

Die erste Idee dieser Projektarbeit war die Umsetzung eines Tippspiels in Java, passend zu den damaligen Fussball-Europameisterschaft in Portugal. Diese Idee wurde im JavaMagazin⁴ in mehreren Ausgaben aufgegriffen und verschiedene Ansatzmöglichkeiten diskutiert. Die Idee unseres Tippspiel war dabei eine Webanwendung mit Datenbankbindung. Nutzer dieses Systems sollten sich in verschiedenen Tippgemeinschaften, mit je einem Tippgemeinschaftsverwalter, zusammen tun und gemeinsam die EM 2004 tippen. Das Tippspiel sollte jedoch nicht nur auf die EM 2004 zugeschnitten sein, sondern auch für andere Fußballereignisse tauglich sein. Zusätzlich kam von unserer Seite die Idee, eine Webanwendung zur Verwaltung der Bundesligaergebnisse. Ein Tippspiel System sollte dann auf diese Daten zurückgreifen und so ein Bundesligatippspiel darstellen können.

Dieser Gedanke wurde jedoch aus verschiedenen Gründen verworfen. Zum einen war es nicht unsere Idee, sondern die des Javamagazin's und zum anderen wussten wir nicht sofort was bei diesem System alles zu realisieren war. Auch war uns die Funktionsweise dieses Tippspiels nicht sofort klar.

1.3.2 Videosoftware

Da jeder von uns schon einmal ein Video in einer Videothek ausgeliehen, kam uns der Gedanke einer Onlinevideothek. Solche Videotheken gab es mittlerweile schon

¹ Auch das Teamprojekt ist Bestandteil des Studiums. Beim Teamprojekt müssen mehrere Studenten (7-15) gemeinsam eine Programmieraufgabe umsetzen.

² <http://labmin.de.vu>

³ <http://www.der-baer.com/projects.htm>

⁴ [Frotscher 2004a, b, c]

wie bspw Amango⁵, Netleih⁶, Invdeo⁷ und Verleihshop⁸. Bei genauer Betrachtung dieser Onlinevideotheken, fragten wir uns wie solch eine Videothek technisch funktioniert. Da wir gerade auf der Suche nach einem idealen Projekt waren, hatten wir damit eins gefunden.

Es sollte versucht werden eine Online-Videothek mit entsprechenden Modulen zu realisieren.

1.4 Zielsetzung

Zielsetzung dieses Projektes ist dabei Erfahrung mit verschiedenen neuen Technologien zu sammeln und selbständig an einem Projekt zu arbeiten. Sowohl die eigene Gedanken, Ideen, Planung und auch Realisierung dieses Projektes sollten uns auf eine spätere Eigenverantwortung im Berufsleben vorbereiten. Das Projekt sollte dabei keine vollständige und fehlerfreie Implementierung darstellen. Uns war bewußt, dass wir nur einen einfachen Prototypen einzelner Module realisieren könnten.

⁵ <http://www.amango.de>
⁶ <http://www.netleih.de>
⁷ <http://www.invdeo.de/>
⁸ <http://www.verleihshop.de>

Kapitel 2

Konzepte und Aufbau

2.1 Aufbau

2.1.1 Beschreibung des Gesamtsystems

Bei einer klassischen Videothek besucht der Kunde das Ladengeschäft der Videothek und stöbert dabei nach Videos, die er gerne an diesen Abend schauen möchte. Dabei muss die Videothek eine möglichst große Ladenfläche besitzen um die Videos dem Kunden zu präsentieren. Nachdem der Kunde sich für ein Video entschieden hat, nimmt er entweder die leere Verpackung oder ein Plastikschild mit einer Nummer zum Verleihschalter der Videothek. Nachdem der Kunde seine Kundenkarte vorzeigt und durch sein Passwort oder seine Unterschrift verifiziert wurde, sucht der Mitarbeiter anhand einer Nummer in der Leerverpackung oder des Plastikschilds das entsprechende Video heraus, markiert dieses Video im System und gibt es dem Kunden. Dies ist der klassische Ablauf in einer Videothek.

Bei einer Online-Videothek kann der Kunde, durch den Versand der Videos, keine Videos für den gleichen Abend ausleihen. Er ist gezwungen, sich einige Tage vorher für ein oder mehrere Videos zu entscheiden. Der Ablauf unterscheidet sich von einer klassischen Videothek. Der Kunden „besucht“ die Webseite der Videothek und sucht im Angebot nach Filmen die er sich ausleihen möchte. Nachdem die Verfügbarkeit überprüft wurde, legt er Videos in seinem Warenkorb ab. Durch Eingabe seines Benutzernamens und das zugehörige Passwort wird der Kunde verifiziert. In dem Lager der Videothek nimmt ein Mitarbeiter die Bestellung über einen Monitor oder eine ausgedruckte Liste entgegen und bearbeitet die Bestellung. Dabei sucht dieser die Videos für den Kunden heraus, nimmt die Videos in das System auf und versendet die Videos zu dem Kunden per Post. Der Kunde erhält seine gewünschten Videos, kann diese sich anschauen und schickt diese nach einer bestimmten Zeit an die Videothek zurück.

Das hier gewünschte System soll dabei eine komplette Videothek ersetzen. Die

Online-Videothek benötigt nur noch ein Lager für die zu verleihenden Videos und wenige Mitarbeiter für den Versand der Videos und die Verwaltung der Videothek. Die Software wird dabei wie in Abbildung 2.1 auf der nächsten Seite zu sehen ist, von zwei verschiedenen Personengruppen benutzt, dem Kunden und dem Mitarbeiter. Der Kunde kann die in der Abbildung dargestellten Aktionen ausführen, wie bspw. betrachten und bestellen von Videos. Der Mitarbeiter kann dabei das System verwalten und Bestellungen der Kunden bearbeiten.

Nach einiger Überlegung wurde festgestellt, dass die Software aus drei anstatt zwei Modulen bestehen muss, wie in Abbildung 2.2 auf Seite 9 zu sehen ist. Bei den Mitarbeitern der Online-Videothek muss in Verwaltung und Versand/Lagen unterschieden werden, da diese unterschiedlichen Aufgaben von unterschiedlichen Mitarbeitern bearbeitet werden. Das **Kundenmodul** ist die Internetpräsenz der Videothek und repräsentiert das Unternehmen nach aussen. Auf dieser dynamischen Webseite kann der Kunde die vorhandenen Videos durchstöbern und detaillierte Informationen zu den Videos erhalten. Nach erfolgreicher Anmeldung im System kann der Kunde die Verfügbarkeit des jeweiligen Videos kontrollieren und auf Wunsch Videos ausleihen. In einem zusätzlichen Menüpunkt kann er seine bestellten Videos betrachten und sich ggf. Rechnungen ausdrucken. Das **Versandmodul** stellt die benötigte Software für das Lager und den Versand zur Verfügung. Mit deren Hilfe kann ein Mitarbeiter der Online-Videothek Videos für den Versand vorbereiten. D.h. der Mitarbeiter bekommt eine Liste mit Bestellungen von Kunden (elektronisch oder auf Papier) und arbeitet diese ab. Damit der Mitarbeiter nicht jedesmal die Kundennummer und Nummern der Videos eintippen muss, wird seine Arbeit durch Barcodes und Barcodescanner unterstützt. Mit dessen Hilfe markiert er Videos für einen bestimmten Kunden und eine bestimmte Bestellung und versendet diese. Dem System teilt der Mitarbeiter dadurch mit, dass bestimmte Videos nicht mehr verfügbar sind und von einem bestimmten Kunden ausgeliehen wurde. Rechnungen, Versandetiketten und eventuelle Lieferscheine werden dabei automatisch mit Hilfe eines Druckers erstellt. Zusätzlich bietet das Versandmodul die Möglichkeit, zurückgekommene Videos der Kunden wieder in das System aufzunehmen. Somit wurde das Video wieder vom Kunden zurückgegeben und es kann im System als vorhanden/ausleihbar markiert werden oder gleich an den nächsten Kunden weitergeschickt werden. Das **Verwaltungsmodul** hilft den Mitarbeitern in der Verwaltung bei der Organisation der Online-Videothek. Es können Kundendaten und Rechnungen betrachtet und ggf. gedruckt werden. Das Videosortiment kann bearbeitet und inhaltliche Änderungen (z.B. Sonderangebote) an dem Kundenmodul vorgenommen werden. Weiterhin besteht die Möglichkeit ausführliche Reports & Statistiken zu erstellen und zu betrachten.

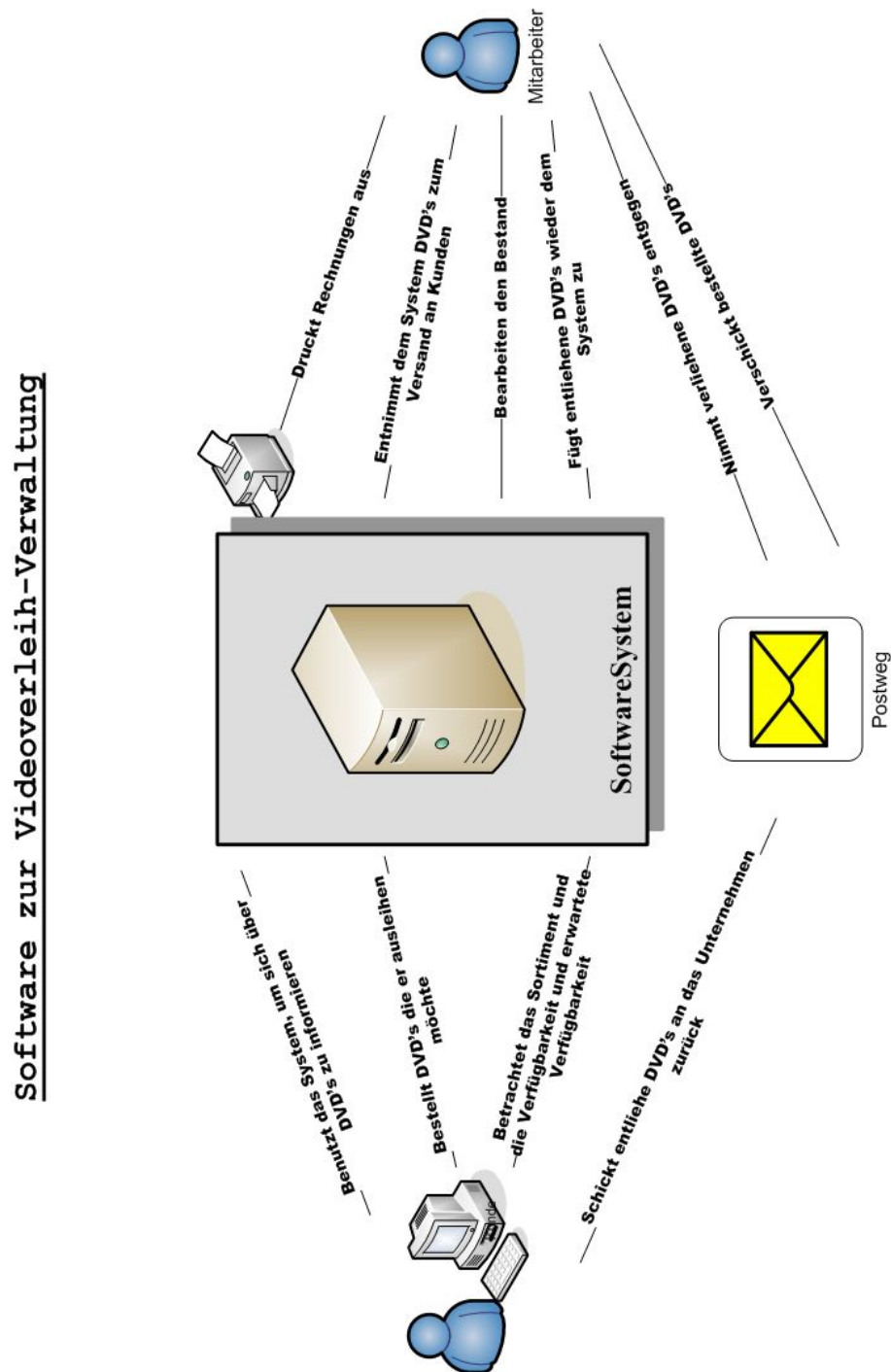


Abbildung 2.1: Erste Gedanken zu der Videosoftware

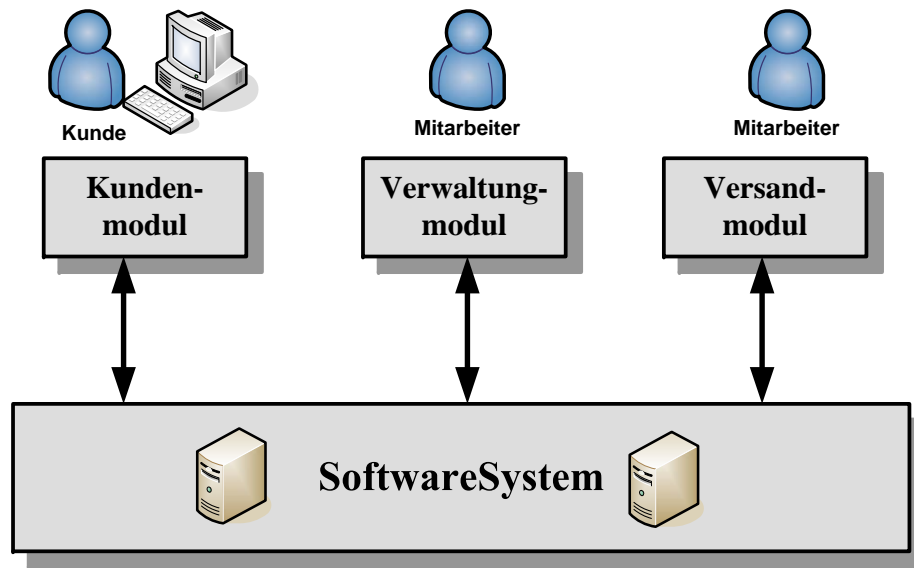


Abbildung 2.2: Die drei Module der Software

2.1.2 das Kundenmodul im Detail

Das Kundenmodul ist eine Webseite die der Nutzer über das Internet erreichen kann.

- anmeldung - informationen zu den einzelnen Videos (beschreibung) - ausführliche Suche (auch intelligente) - verfügbarkeit (anzahl und wann das nächste video wieder da ist) - betrachtung der VORBESTELLUNGEN (bzw. Abarbeitungsliste)
- betrachtung der bestellhistorie - Rechnungen/Mahnungen - allgemeine Einstellungen

Die Kundensoftware ist eine Webapplikation. Auf dieser „Webseite“ können sich sowohl Interessenten als auch Kunden über Videos informieren. Nach Registrierung kann man sich auch den Status von einzelnen Videos anschauen und verleihbare ausleihen. Neben Statistiken stehen dem Kunden auch ein Navigationspunkt mit seinen bisherigen Bestellungen und dem aktuellen Stand seiner Wunschliste zur Verfügung.

2.1.3 das Versandmodul im Detail

- Übersicht über offenen Bestellungen, - erstellung einer bestellliste für einen mitarbeiter - erstellung lieferschein/rechnung, etiketten - videos aus dem system nehmen (versand) - videos in das system nehmen (rückversand) - barcode als kundennummer, rechnungsnummer, und auf jedem Video - lagerplatz der videos (regal, reihe)

2.2 Einsatzbereich

eventuell automatisch - Ablauf einer Bestellung

Ein Mitarbeiter der Videothek hat dann die Möglichkeit über eine eigene Software die gewünschten Videos zum Versand vorzubereiten. Dabei wird er so weit vom System unterstützt, dass er nur noch die fehlenden Roboterarme ersetzen muss. Mit Hilfe eines Barcodescanners kann der Mitarbeiter, auf dem bereits fertig ausgedrucktem Bestellschein, die Kundennummer und die gewünschten Videos erfassen. Der Barcodescanner erleichtert dabei nur die Eingabe von Kundennummern, Rechnungsnummern und Artikelnummern. Die Aufgaben des Mitarbeiters für eine Bestellung können wie folgt zusammengefasst werden:

- Bestellschein vom Drucker entnehmen
- Bestellnummer vom Bestellschein dem System mitteilen
- gewünschte Artikel aus dem Sortiment nehmen
- Artikelnummern einzeln dem System mitteilen *dadurch wird dem System mitgeteilt, daß diese Artikel verliehen sind*
- Artikel in einen Versandumschlag packen und ggf. Versandaufkleber vom Bestellschein entnehmen

2.1.4 das Verwaltungsmodul im Detail

- Supporttelefon - bestellungen betrachten, verändern - kundendaten betrachten - neue Filme ins System aufnehmen - neue Videos(dvds, vhs) ins system aufnehmen - preise ändern - sonderaktionen hinzufügen - statistiken (auslastung) betrachten,exportieren und drucken

Die Verwaltungssoftware wird von Mitarbeitern der Videothek bedient. Mit Hilfe dieser ist es möglich das Video-Sortiment zu bearbeiten und zu vervollständigen. Weiterhin ist es möglich mit der Verwaltungssoftware Rechnungen von Kunden, Bestellungen oder ähnliches zu bearbeiten. Zusätzlich steht eine ausführliche Statistik zu einzelnen Kunden, einzelnen Videos und Genre zur Verfügung.

2.2 Einsatzbereich

Diese Software ist sowohl für kleine als auch für grosse Unternehmen gedacht. Dabei ist es unwichtig, ob es sich um eine reine OnlineVideothek oder um eine richtige Videothek, die jetzt auch per Versand irhe Videos verleihen möchte, handelt. Durch weitere Module kann die Software so erweitert werden, dass die Software auch für eine richtige Videothek geeignet ist.

2.3 verschiedene Versionen

Die einfache Version des System wird von uns implementiert. Diese Version soll die wichtigsten Bestandteile des Systems enthalten, jedoch nicht zu komplex sein. Es soll außerdem möglich sein, diese Version durch Updates mit zu erweitern, sodaß eine komplexes System entstehen kann.

Diese einfache Version beinhaltet die Kundensoftware(Webanwendung) und die Lagersoftware in einfacher Form. Dies bedeutet für die Lagersoftware, dass keine grafische Oberfläche vorhanden ist. Als Ausgabemedium dient die Konsole, die man mit einem Dos Fenster vergleichen kann.

Eine Verwaltungssoftware ist nicht geplant. Diese wird jedoch zu einem späteren Zeitpunkt auch als Consolenanwendung realisiert. Die einfache Version ist als Testversion gedacht, bei der die Daten bereits in der Datenbank vorhanden sind, somit ist eine Verwaltungssoftware erstmal überflüssig.

- Kundenmodul
 -
- Verwaltungsmodul
 - grafische Benutzeroberfläche die unter Windows und Linux verfügbar ist
 - neue DVD's hinzufügen bzw. bearbeiten & löschen
 - ausführliche Statistiken
 -
- Versandmodul
 - grafische Benutzeroberfläche die unter Windows und Linux verfügbar ist
 - automatischer Ausdruck der Versandlabels und Rechnungen

Kapitel 3

Technologien

In diesem Abschnitt werden kurz die zu verwendeten Technologien verwendet.

3.1 Datenbank

Zum Einsatz soll eine OpenSource Datenbank kommen. Gedanken an eine kommerzielle Datenbank kam aus Gründen der Lizenzkosten nicht auf.

Zu Auswahl standen mehrere OpenSource Datenbanken. MySql ¹, SAP DB ², HSQL DB ³ und Firebird ⁴.

- **MySql**

- + sehr verbreitet
- + einige Erfahrung
- + gut Dokumentiert & große Community
- +
 - schlechtes Lizenzmodell
 - zu bekannt
 - keine Trigger
 - meist nur im privat bzw. klein Unternehmer Einsatz

- **SAP DB**

-
- ¹ <http://dev.mysql.com/downloads/mysql/4.0.html>
 - ² <http://dev.mysql.com/downloads/maxdb/7.5.00.html>
 - ³ <http://hsqldb.sourceforge.net>
 - ⁴ <http://firebird.sourceforge.net>

3.1 Datenbank

- +
- + Datenbank seit mehreren Jahren bei SAP im Einsatz
- schlechte Skalierbarkeit, da der Datenbank Speicherbereich im Vorfeld festgelegt werden muss
- schlechte Erfahrung

• HSQLDB

- + reine JavaDatenbank
- + sehr klein
- + kann als reine Speicher Datenbank verwendet werden (Daten nur im Arbeitsspeicher)
- + kann als Applikations Datenbank verwendet werden (nur eine Applikation benutzt die Datenbank)
- nicht für große Applikationen geeignet
-

• Firebird

- + geringe Erfahrung durch Studium
- + sehr klein
- + gute grafische Tools
- + Original Sourcen kommen von Borland
- + Interbase Datenbank seit mehreren Jahren im Professionelle einsatz
- schlechtes Lizenzmodell
-

Wir haben uns für die Firebird Datenbank entschieden, da es keine wirkliche Konkurrenz im Open Source Bereich gibt.

HSQL scheidet schon aus, weil es nicht für grosse Datenmengen geeignet ist. Bei der SAP DB muss der benötigte Speicherplatz der Datenbank vorher bekannt sein, was bei unserem Projekt nicht der Fall ist. MYSQL unterstützt keine Triggers und ist zu bekannt, d.h. MySql kann und sollte jeder Informatiker kennen und benutzt haben.

Firebird ist für uns relativ neu und die Erfahrungen die wir in der Vorlesung „Datenmanagment 2“ bekommen haben, war sehr positiv. Da diese Datenbank ursprünglich von Borland kommt, ist diese Datenbank auch nicht so neu, wie viele

3.2 Versionsverwaltung

Denken.

Es soll aber schon am Anfang des Projektes bedacht werden, dass die Datenbank zu einem späteren Zeitpunkt eventuell mit einer professionellen Datenbank⁵ ausgetauscht werden könnte. Deswegen muss schon am Anfang eine hohe Abstraktionsebene vorhanden sein, so dass eventuelle Datenbankspezifische Elemente (Klassen) sehr einfach ausgetauscht werden können.

3.2 Versionsverwaltung

Eine Versionsverwaltung

<http://better-scm.berlios.de/comparison/comparison.html> [Wikipedia 2005, Versionsverwaltung]

⁵ z.B. DB2 von IBM

3.3 Entwicklungsumgebung

3.2.1 Concurrent Versions System - CVS

3.2.2 Subversion

3.3 Entwicklungsumgebung

3.3.1 JBuilder

3.3.2 Netbeans

3.3.3 Eclipse

3.4 grafischen Benutzerschnittstellen in Java

3.4.1 Abstract Window Toolkit - AWT

3.4.2 Swing

3.4.3 Standard Widget Toolkit - SWT

3.5 Java-Web-Anwendungen

3.5.1 Java Server Faces - JSF

3.5.2 Struts

3.6 Persistenzschichten in Java

3.6.1 Java Data Objects

3.6.2 Hibernate

Datenbank - erste Ideen

- Kunden

- kundenid *Integer Autoincrement Primary Key*
- name *VARCHAR(200)*
- vorname *VARCHAR(200)*
- strasse *VARCHAR(200)*

- benutzer

- benutzerid *Integer Autoincrement Primary Key*
- dvds
 - dvdid *Integer Autoincrement Primary Key*
- genre
 - genreid *Integer Autoincrement Primary Key*
- artikel
 - artikelid *Integer Autoincrement Primary Key*
- verleih
 - verleihid *Integer Autoincrement Primary Key*
- preis
 - preisid *Integer Autoincrement Primary Key*

Prototyp

Bevor versuchen ein fertiges Produkt zu realisieren und daran vermutlich scheitern werden, haben wir beschlossen einen einfachen Prototyp zu programmieren. Dieser soll die wichtigsten Merkmale besitzen und zu Demonstrationszwecken dienen. Jedoch soll es auch möglich sein, diesen Prototypen zu einem fertigen Produkt fertig zu entwickeln. Der Prototyp soll also nicht quick & dirty programmiert werden.

Der Prototyp soll hauptsächlich die Kundenseite implementieren. D.h. er soll eine Webanwendung bereitstellen, bei der der Kunde bzw. Interessent sich registrieren kann und die Videothek benutzen kann. Dies bedeutet er kann sich die vorhanden Videos anschauen (Informationen zu diesen), kann sich die Verfügbarkeit anschauen, Video ausleihen, Rechnungen ansehen bzw. ausdrucken und eine Liste mit all seinen bisherigen Bestellungen anschauen.

Das Modul für die Verwaltung der DVD's ist in diesem Prototypen noch nicht vorgesehen.

Das Modul für das Versenden und Empfangen ist nur in einfacher Variante vorgesehen. Der Mitarbeiter der Onlinevideothek bekommt eine kleine Anwendung auf Konsolenbasis ohne grafische Oberfläche.

Kapitel 4

Implementierung

4.1 Versionsverwaltung mit Subversion

¹

<http://www.phil-schneider.de> [Frotscher 2004c] Verweis auf eine Literaturquelle *hervorgehoben*

Fett

Typewriter

4.2 Entwicklungsumgebung mit Eclipse

4.3 grafischen Benutzerschnittstellen mit SWT

4.4 Java-Web-Anwendungen mit Struts

4.5 Persistenzschichten mit Hibernate

¹ So wird eine Fussnote gemacht

Kapitel 5

Zusammenfassung

Könnte von weiteren Studenten weitergeführt werden, da das gesamte Projekt in einzelne Module geteilt werden kann.

- suchmodul apache lucence - barcodemodul - lagerverteilung der Videos und kürzeste Wegstreckenberechnung - reports und statistik erstellung mit eclipse birt
- GUI (swt) tool für das Lager mit touchscreen

Anhang A

Protokoll vom 11. Mai 2004

Drei Frameworks stehen zur Auswahl

- Apache Cocoon
- Apache Struts
- Apache Tapestry

Jeder erstellt eine einfache (Web)Anwendung mit Hilfe eines dieser Frameworks folgende Komponenten sollen/muessen enthalten sein:

- einfache LoginSeite (über Datenbank)
- Liste aller Videos in Datenbank anzeigen
- EingabeMaske für neues Labor
- Validierung der Eingabedaten
- dynamische Navigation
- eventuell ein Bild für den Status der einzelnen Bilder (dynamisches Bild??)

Für diese BeispielAnwendung sollen möglichst viele Elemente des jeweiligen Framework verwendet werden. Wichtig ist dabei der Umgang und die Bedienbarkeit des Systems.

Wiederverwendbarkeit einzelner Module.

Design und Logik Trennung vorhanden? Kann das Design einfach/schnell ausgetauscht werden.

Es geht dabei nicht um ein 100Design spielt keine wichtige Rolle, es sollte jedoch beachtet werden, dass dieses später vom Kunden ausgetauscht werden möchte.

-
- Struts: Stefan
 - Cocoon: Philipp
 - Tapestry: Remo
-
- Namen für das Projekt finden
 - Link mit Beispiel Webseiten rumschicken

Literaturverzeichnis

Frotscher 2004a

FROTSCHER, Thilo: Der Ball ist rund. In: *Javamagazin* 04 (2004), S. 95

Frotscher 2004b

FROTSCHER, Thilo: Das naechste Spiel ist immer das schwerste. In: *Javamagazin* 05 (2004), S. 89

Frotscher 2004c

FROTSCHER, Thilo: Das Runde muss ins Eckige. In: *Javamagazin* 06 (2004), S. 60

Wikipedia 2005

WIKIPEDIA: *Wikipedia - Enzyklopädie*. Version: 2005. <http://de.wikipedia.org>. – Online Ressource