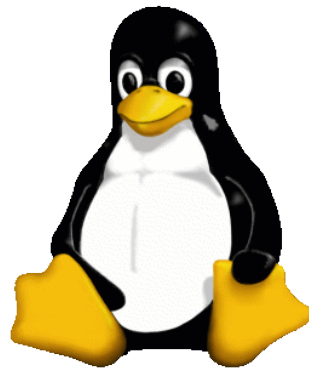


**Fiktive Projektentwicklung eines Systems zur
Verwaltung von Studienergebnissen
von Studenten der NTA FH Isny**



Projektarbeit Software-Development

NTA FH Isny
12. Info

Jue Li, Matthias Müller, Richard Peters, David Mayr

Inhaltsverzeichnis

| | | |
|-------|---|----|
| 1 | Aufgabenstellung..... | 3 |
| 2 | Planungsphase..... | 4 |
| 2.1 | Lastenheft..... | 4 |
| 2.1.1 | Zielbestimmung..... | 4 |
| 2.1.2 | Produkteinsatz..... | 4 |
| 2.1.3 | Produktfunktionen..... | 4 |
| 2.1.4 | Produktdaten..... | 4 |
| 2.1.5 | Produktleistungen..... | 5 |
| 2.1.6 | Qualitätsanforderung..... | 5 |
| 2.1.7 | Ergänzung..... | 5 |
| 2.2 | Aufwandsschätzung..... | 5 |
| 3 | Definitionsphase (OOA)..... | 7 |
| 3.1 | Pflichtenheft..... | 7 |
| 3.1.1 | Aufgaben..... | 7 |
| 3.1.2 | Adressaten..... | 7 |
| 3.1.3 | Inhalt / Produktfunktionen..... | 7 |
| 3.1.4 | Produktdaten..... | 8 |
| 3.1.5 | Produktumgebung..... | 8 |
| 3.2 | Objektorientierte Analyse..... | 9 |
| 3.2.1 | Use-Case Diagramm..... | 9 |
| 3.2.2 | Klassen-Diagramm..... | 13 |
| 3.2.3 | Sequenz-Diagramme..... | 14 |
| 4 | Entwurfs-/Designphase (OOD)..... | 17 |
| 4.1 | Klassendiagramm mit Assoziationen / Kardinalitäten..... | 17 |
| 5 | Implementierungsphase (Prototyp)..... | 18 |
| 5.1 | Anmeldedialog..... | 18 |
| 5.2 | Hauptfenster..... | 18 |
| 5.3 | Erläuterungen zum Hauptfenster..... | 20 |

1 Aufgabenstellung

Eine Hochschule möchte in Ihrem Prüfungsamt die Verwaltung der Studienergebnisse ihrer Studenten automatisieren.

Jeder Student wird bei der Immatrikulation einem **Fachbereich** der Hochschule zugeordnet. Fachbereiche bieten **Studiengänge** an. Innerhalb eines Studiengangs kann ein Student **Module** wählen.

Für die einzelnen Module sind Prüfungsergebnisse der einzelnen Studenten zu erfassen, zu speichern und sichtbar zu machen.

Hierbei sind die folgenden Phasen einer objektorientierten Entwicklung zu durchlaufen:

- Planungsphase
- Definitionsphase
- Entwurfsphase
- Implementierungsphase

Die Implementierungsphase beschränkt sich auf die Realisierung eines Prototyps der Benutzerschnittstelle.

2 Planungsphase

2.1 Lastenheft

2.1.1 Zielbestimmung

Automatisieren der Verwaltung von Studienergebnissen für eine Hochschule.

2.1.2 Produkteinsatz

Zielgruppe: Verwaltungsangestellte und Dozenten einer Hochschule.

2.1.3 Produktfunktionen

LF10: **Fachbereiche** anlegen, ändern und löschen
LF20: **Studiengänge** anlegen, ändern und löschen
LF30: **Module** anlegen, ändern und löschen
LF40: **Studenten** erfassen, ändern und löschen
LF50: **Ergebnisse** erfassen, ändern, Liste drucken und auf Zeugnis drucken

2.1.4 Produktdaten

LD10: **Fachbereich**
 Bezeichnung
 Dekan
LD20: **Studiengang**
 Bezeichnung
 Fachbereich ->
LD30: **Modul**
 Bezeichnung
 Studiengang ->
 Dozent
LD40: **Student**
 Name
 Matrikelnummer
 Adresse
 Geburtsdatum
 Studiengang ->
LD50: **Ergebnis**
 Note
 Bemerkung
 Modul ->
 Student ->

2.1.5 Produktleistungen

LL10: Matrikelnummer 6-stellig
LL20: Sicherheits- / Passwortabfrage für Zugriff zu System

Mit dem Produkt sollen Studenten von einer zuständigen Person erfasst und modifiziert werden können. Die Dozenten sollen die erbrachten Leistungen ihrer Studenten für die jeweiligen Module in das System eingeben können.

Dem Prüfungsamt soll die Arbeit dadurch leichter gemacht werden, dass das Software-Produkt automatisch Zeugnisse und Notenlisten zum Aushang oder veröffentlichen per Internet generieren kann.

2.1.6 Qualitätsanforderung

- Funktionalität: normal
- Zuverlässigkeit: hoch
- Datensicherheit: sehr hoch
- Benutzbarkeit: normal
- Änderbarkeit: normal
- Portabilität: normal

2.1.7 Ergänzung

Hauptteil der Entwicklungskosten: Personalkosten

Da LINUX bei der Entwicklung benutzt wurde fallen keine weiteren Kosten an!

2.2 Aufwandsschätzung

Wir führen die Aufwandsschätzung mit Hilfe der Function-Point-Methode durch.

Function-Points:

| Kategorie | Anzahl | Klass. | Gewichtung | Zeilensumme |
|---------------|--------|--------|------------|-------------|
| Einagbe | 8 | leicht | x3 | 24 |
| Abfrage | 2 | leicht | x3 | 6 |
| Ausgabe | 1 | leicht | x4 | 4 |
| Datenbestände | 1 | leicht | x7 | 7 |
| Referenzdaten | - | - | - | - |
| Summe | | | E1 | 41 |

Einflussfaktoren:

| | |
|---|-----------|
| Verflechtung mit anderen Anwendungssystemen | = 1 |
| Dezentrale Daten, dezentrale Verarbeitung | = 1 |
| Transaktionsrate | = 1 |
| <u>Verarbeitungslogik</u> | |
| Rechenoperationen | = 2 |
| Kontrollverfahren | = 1 |
| Ausnahmeregelungen | = 1 |
| Logik | = 0 |
| Wiederverwendbarkeit | = 0 |
| Datenbestandskonvertierung | = 0 |
| Anpassbarkeit | = 1 |
| | ===== |
| Summe: | E2 8 |

Faktor Einflussbewertung: $E3 = 8/100 + 0,7 = 0,78$

Bewertete Funktion Points: = $E1 * E3 = 31,98 \text{ FP}$

In Mannmonate sind das ungefähr: $= 1,5 \text{ MM}$

3 Definitionsphase (OOA)

3.1 Pflichtenheft

3.1.1 Aufgaben

Verwaltung der Daten über Studenten und Ergebnisse.

3.1.2 Adressaten

Auftraggeber: Hochschule
Auftragnehmer: Studenten

3.1.3 Inhalt / Produktfunktionen

Die Software, im weiteren SW bezeichnet, soll folgende Merkmale besitzen:

- **Fachbereich anlegen**
Bezeichnung, Dekan und angebotene Studiengänge werden mit dieser Funktion angelegt.
- **Fachbereich ändern**
Diese Funktion ändert die Daten des Fachbereichs.
- **Fachbereich löschen**
Die Funktion löscht einen Fachbereich.

- **Studiengang anlegen**
Bezeichnung und enthaltene Module werden mit dieser Funktion angelegt.
- **Studiengang ändern**
Diese Funktion ändert die Daten des Studiengangs.
- **Studiengang löschen**
Die Funktion löscht einen Studiengang.

- **Modul anlegen**
Bezeichnung und Dozent werden mit dieser Funktion angelegt.
- **Modul ändern**
Diese Funktion ändert die Daten des Moduls.
- **Modul löschen**
Die Funktion löscht ein Modul.

- **Studenten anlegen**
Name, Anschrift, Geburtsdatum, Matrikelnummer, Studiengang und Module werden mit dieser Funktion erfasst.
- **Studenten ändern**
Diese Funktion ändert die Daten eines Studenten.
- **Studenten löschen**

Die Funktion löscht die Daten eines Studenten.

- **Ergebnisse erfassen/eintragen**
Mit dieser Funktion werden die Ergebnisdaten (z.B. Note) erfasst
- **Ergebnisse ändern**
Diese Funktion ändert die Ergebnisdaten.
- **Ergebnisse veröffentlichen**
Diese Funktion generiert eine HTML-Seite mit definierten Ergebnisdaten.
- **Ergebnisse auf Zeugnissen drucken**
Diese Funktion druckt Zeugnisse.

3.1.4 Produktdaten

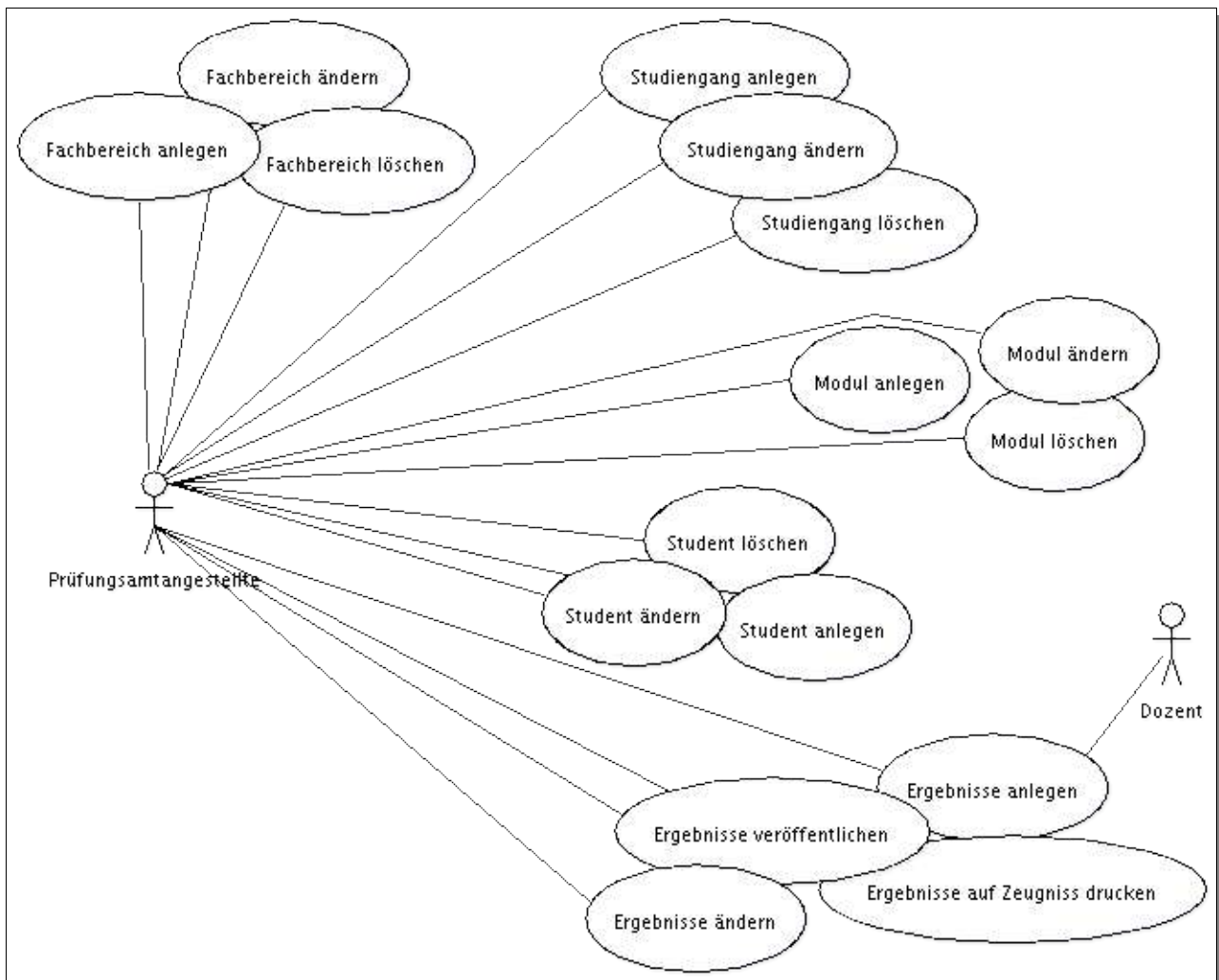
| | |
|-------|--|
| PD10: | Fachbereich Bezeichnung Dekan Studiengänge |
| PD20: | Studiengang Bezeichnung Module |
| PD30: | Modul Bezeichnung Dozent |
| PD40: | Student Name Matrikelnummer Adresse Geburtsdatum Studiengang |
| PD50: | Ergebnis Note Modul Student |

3.1.5 Produktumgebung

Das Produkt soll auf einem Arbeitsplatz mit grafischer Oberfläche laufen. Als stabile Basis soll das freie Betriebssystem Linux verwendet werden.

3.2 Objektorientierte Analyse

3.2.1 Use-Case Diagramm



Spezifikation:

Anlegen eines Fachbereichs

| | |
|---------------------------|---|
| Ziel: | Speicherung der Daten eines Fachbereichs |
| Vorbedingung: | Fachbereich noch nicht vorhanden |
| Nachbedingung Erfolg: | sämtliche Fachbereichsdaten werden gespeichert |
| Nachbedingung Misserfolg: | Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert |
| Akteure: | Angestellte der Verwaltung |
| auslösendes Ereignis: | Anbieten eines neuen Fachbereichs / Ersterfassung |
| Beschreibung: | 1. Eingabe der Daten 2. Speicherung der Daten |

Ändern eines Fachbereichs

| | |
|---------------------------|---|
| Ziel: | Ändern der Daten eines Fachbereichs |
| Vorbedingung: | Fachbereich vorhanden |
| Nachbedingung Erfolg: | sämtliche Fachbereichsdaten werden geändert und gespeichert |
| Nachbedingung Misserfolg: | Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert |
| Akteure: | Angestellte der Verwaltung |
| auslösendes Ereignis: | Daten eines Fachbereichs haben sich geändert |
| Beschreibung: | <ol style="list-style-type: none">1. Auswahl des Fachbereichs2. Ausgabe der bisherigen Daten3. Eingabe der neuen Daten4. Speicherung der Daten |

Löschen eines Fachbereichs

| | |
|---------------------------|--|
| Ziel: | Daten eines Fachbereichs löschen |
| Vorbedingung: | Fachbereich vorhanden |
| Nachbedingung Erfolg: | sämtliche Fachbereichsdaten werden gelöscht |
| Nachbedingung Misserfolg: | Ausgabe einer Fehlermeldung, Daten werden nicht gelöscht |
| Akteure: | Angestellte der Verwaltung |
| auslösendes Ereignis: | Fachbereich wurde aufgelöst |
| Beschreibung: | <ol style="list-style-type: none">1. Auswahl des Fachbereichs2. Entfernen der Daten des Fachbereichs aus dem Datenbestand |

Anlegen eines Studiengangs

| | |
|---------------------------|---|
| Ziel: | Speicherung der Daten eines Studiengangs |
| Vorbedingung: | Studiengang noch nicht vorhanden |
| Nachbedingung Erfolg: | sämtliche Studiengangsdaten werden gespeichert |
| Nachbedingung Misserfolg: | Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert |
| Akteure: | Angestellte der Verwaltung |
| auslösendes Ereignis: | Anbieten eines neuen Studiengangs / Ersterfassung |
| Beschreibung: | <ol style="list-style-type: none">1. Eingabe der Daten2. Speicherung der Daten |

Ändern eines Studiengangs

| | |
|---------------------------|---|
| Ziel: | Ändern der Daten eines Studiengangs |
| Vorbedingung: | Studiengang vorhanden |
| Nachbedingung Erfolg: | sämtliche Studiengangssdaten werden geändert und gespeichert |
| Nachbedingung Misserfolg: | Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert |
| Akteure: | Angestellte der Verwaltung |
| auslösendes Ereignis: | Daten eines Studiengangs haben sich geändert |
| Beschreibung: | <ol style="list-style-type: none">1. Auswahl des Studiengangs2. Ausgabe der bisherigen Daten3. Eingabe der neuen Daten4. Speicherung der Daten |

Löschen eines Studiengangs

Ziel: Daten eines Studiengangs löschen
Vorbedingung: Studiengang vorhanden
Nachbedingung Erfolg: sämtliche Studiengangssdaten werden gelöscht
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gelöscht
Akteure: Angestellte der Verwaltung
auslösendes Ereignis: Studiengang wurde aufgelöst
Beschreibung: 1. Auswahl des Studiengangs
2. Entfernen der Daten des Studiengangs aus dem Datenbestand

Anlegen eines Moduls

Ziel: Speicherung der Daten eines Moduls
Vorbedingung: Modul noch nicht vorhanden
Nachbedingung Erfolg: sämtliche Moduldaten werden gespeichert
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert
Akteure: Angestellte der Verwaltung
auslösendes Ereignis: Anbieten eines neuen Moduls / Ersterfassung
Beschreibung: 1. Eingabe der Daten
2. Speicherung der Daten

Ändern eines Moduls

Ziel: Ändern der Daten eines Moduls
Vorbedingung: Modul vorhanden
Nachbedingung Erfolg: sämtliche Moduldaten werden geändert und gespeichert
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert
Akteure: Angestellte der Verwaltung
auslösendes Ereignis: Daten eines Moduls haben sich geändert
Beschreibung: 1. Auswahl des Moduls
2. Ausgabe der bisherigen Daten
3. Eingabe der neuen Daten
4. Speicherung der Daten

Löschen eines Moduls

Ziel: Daten eines Moduls löschen
Vorbedingung: Modul vorhanden
Nachbedingung Erfolg: sämtliche Moduldaten werden gelöscht
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gelöscht
Akteure: Angestellte der Verwaltung
auslösendes Ereignis: Modul wurde aufgelöst
Beschreibung: 1. Auswahl des Moduls
2. Entfernen der Daten des Moduls aus dem Datenbestand

Erfassen von Studenten

Ziel: Speicherung der Daten von Studenten
Vorbedingung: Student noch nicht vorhanden
Nachbedingung Erfolg: sämtliche Studentendaten werden gespeichert

Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert
Akteure: Angestellte der Verwaltung
auslösendes Ereignis: Immatrikulation eines Studenten
Beschreibung: 1. Eingabe der Daten
2. Speicherung der Daten

Ändern von Studenten

Ziel: Ändern der Daten von Studenten
Vorbedingung: Student vorhanden
Nachbedingung Erfolg: sämtliche Studentendaten werden geändert und gespeichert
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert
Akteure: Angestellte der Verwaltung
auslösendes Ereignis: Daten eines Students haben sich geändert
Beschreibung: 1. Auswahl des Students
2. Ausgabe der bisherigen Daten
3. Eingabe der neuen Daten
4. Speicherung der Daten

Löschen von Studenten

Ziel: Daten von Studenten löschen
Vorbedingung: Student vorhanden
Nachbedingung Erfolg: sämtliche Studentendaten werden gelöscht
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gelöscht
Akteure: Angestellte der Verwaltung
auslösendes Ereignis: Student wurde exmatrikuliert
Beschreibung: 1. Auswahl des Students
2. Entfernen der Daten des Students aus dem Datenbestand

Anlegen von Ergebnissen

Ziel: Speicherung von Studienergebnissen
Vorbedingung: Studienergebnis noch nicht vorhanden
Nachbedingung Erfolg: das Ergebnis wird gespeichert
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert
Akteure: Angestellte der Verwaltung & Dozent des jew. Moduls
auslösendes Ereignis: fertiggestellte Korrektur eines Leistungsnachweises
Beschreibung: 1. Eingabe der Daten
2. Speicherung der Daten

Ändern von Ergebnissen

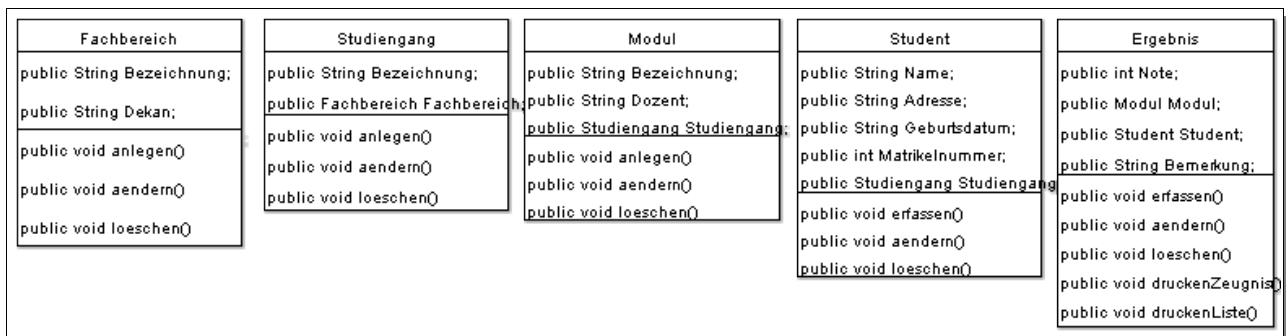
Ziel: Ändern von Ergebnissen
Vorbedingung: Ergebnis vorhanden
Nachbedingung Erfolg: Ergebnis wird geändert und gespeichert
Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gespeichert
Akteure: Angestellte der Verwaltung & Dozent des jew. Moduls
auslösendes Ereignis: Daten eines Students haben sich geändert
Beschreibung: 1. Auswahl des Ergebnisses
2. Eingabe des neuen Ergebnisses

3. Speicherung der Daten

Löschen von *Ergebnissen*

Ziel: Ergebnis löschen
 Vorbedingung: Ergebnis vorhanden
 Nachbedingung Erfolg: Ergebnis aus Datenbestand entfernen
 Nachbedingung Misserfolg: Ausgabe einer Fehlermeldung, Daten werden nicht gelöscht
 Akteure: Angestellte der Verwaltung
 auslösendes Ereignis: Ergebnis ungültig
 Beschreibung: 1. Auswahl des Ergebnisses
 2. Entfernen der Daten des Ergebnisses aus dem Datenbestand

3.2.2 Klassen-Diagramm



Attribut-Spezifikation:

Klasse Fachbereich:

Bezeichnung String {mandatory, key}
 Dekan String {mandatory}
 Studiengänge Studiengang {mandatory} -> Studiengänge in diesem Fachb.

Klasse Studiengang:

Bezeichnung String {mandatory, key}
 Module Modul {mandatory} -> Module in diesem Studiengang

Klasse Modul:

Bezeichnung String {mandatory, key}
 Dozent String {mandatory}

Klasse Student:

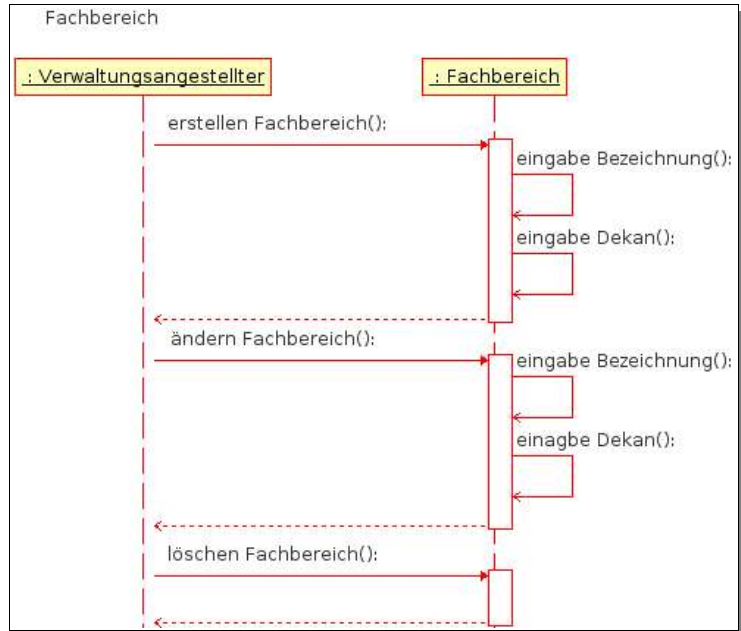
Name String {mandatory}
 Matrikelnummer Int {mandatory, key}
 Adresse String {mandatory}
 Geburtsdatum Date {mandatory}
 Studiengang Studiengang {mandatory} -> gewählter Studiengang des Student

Klasse Ergebnis:

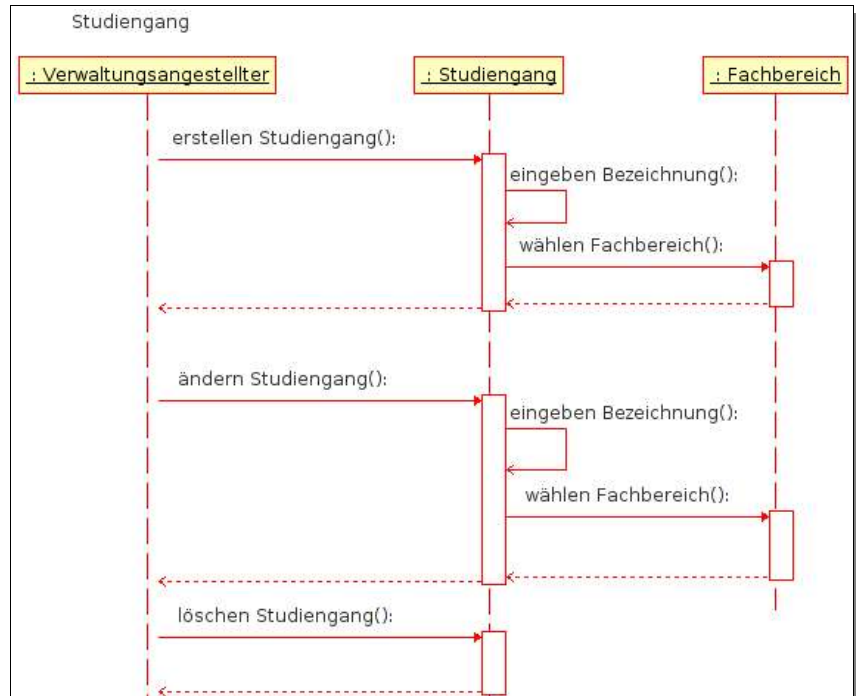
Note Float {mandatory}
 Modul Modul {mandatory} -> Modul in dem es ein Ergebnis gibt
 Student Student {mandatory} -> Student der das Ergebnis erreicht hat

3.2.3 Sequenz-Diagramme

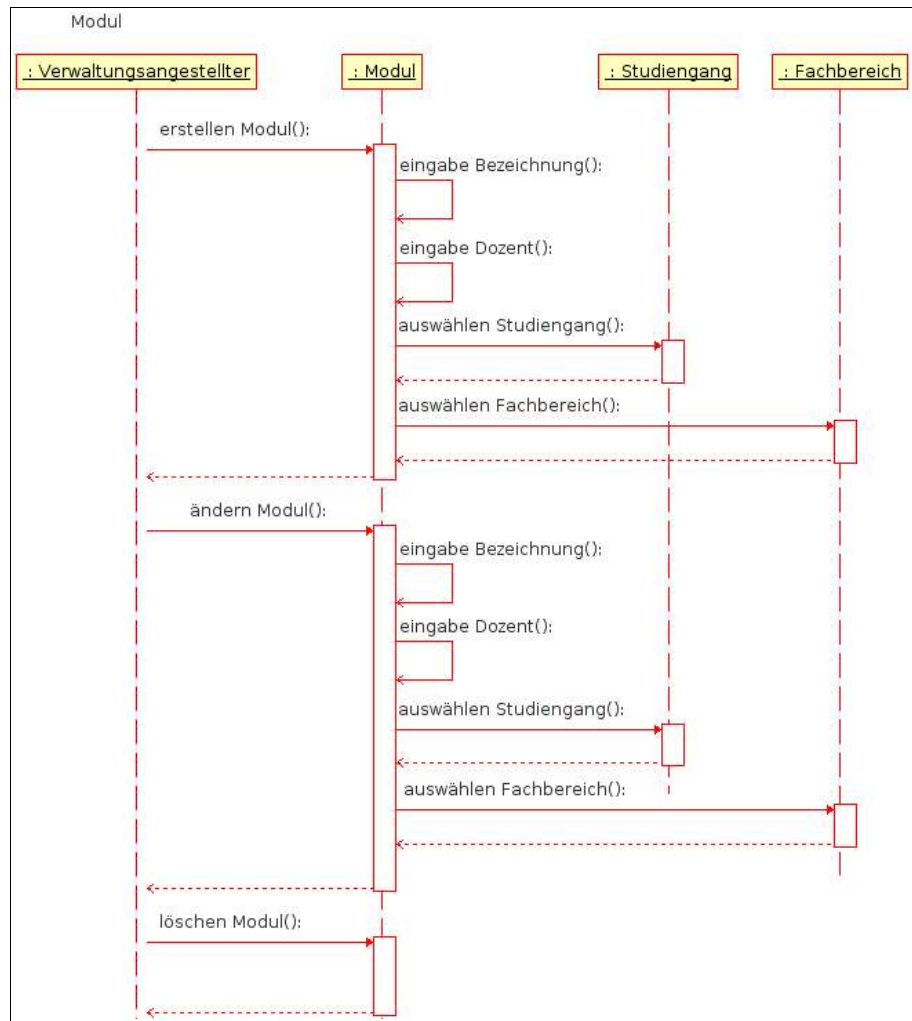
Fachbereich



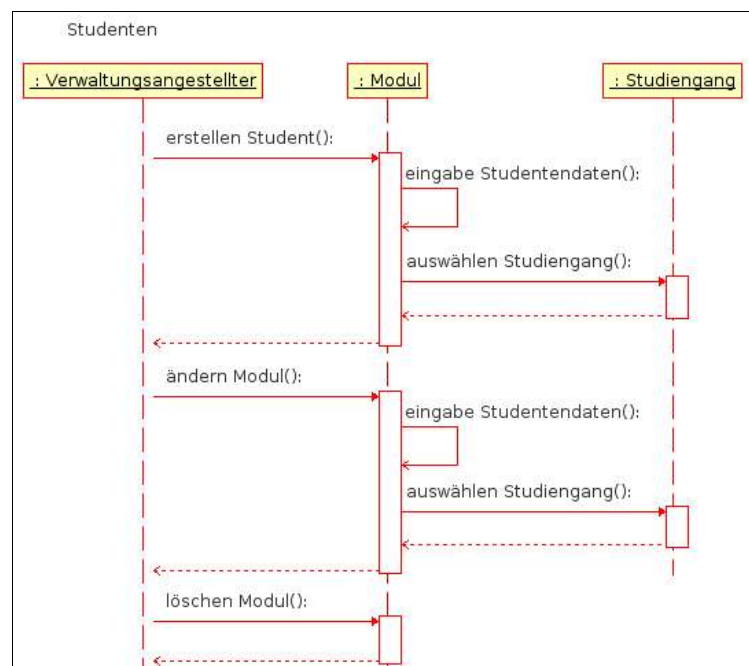
Studiengang



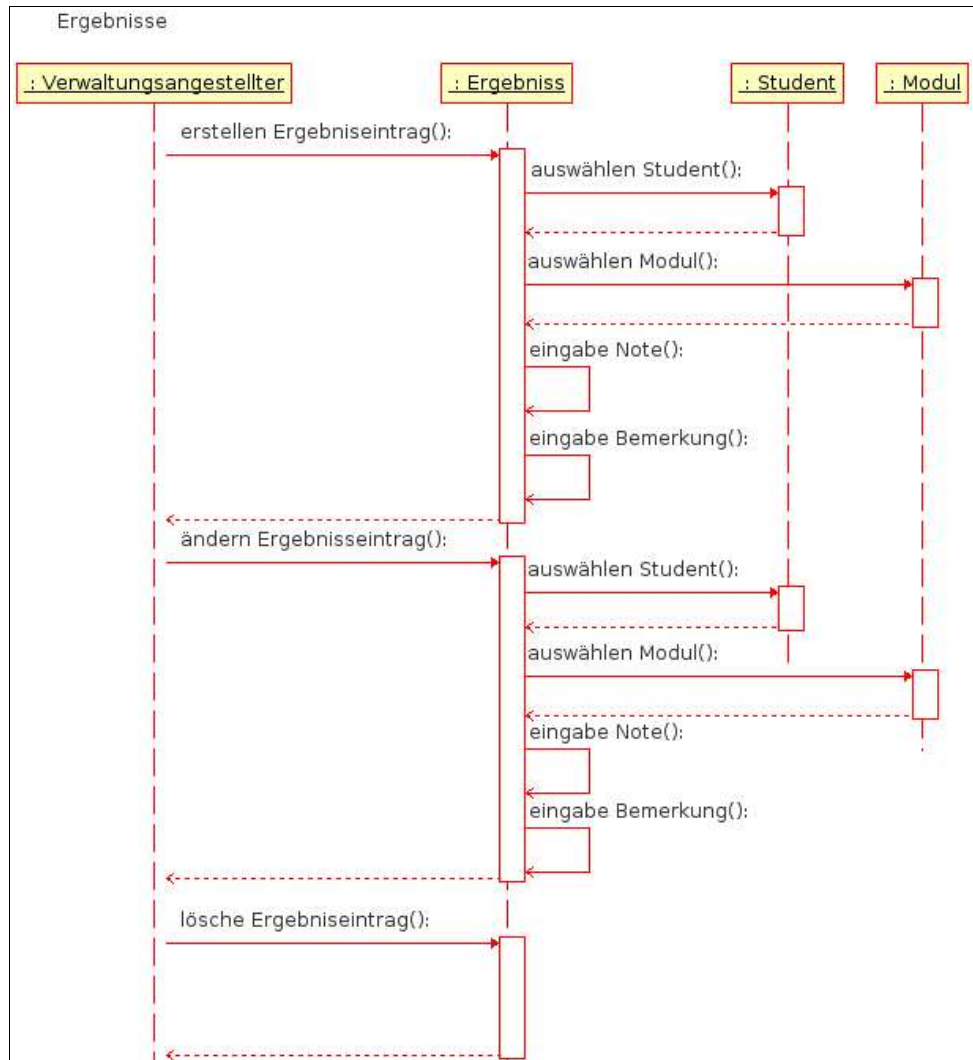
Modul



Student

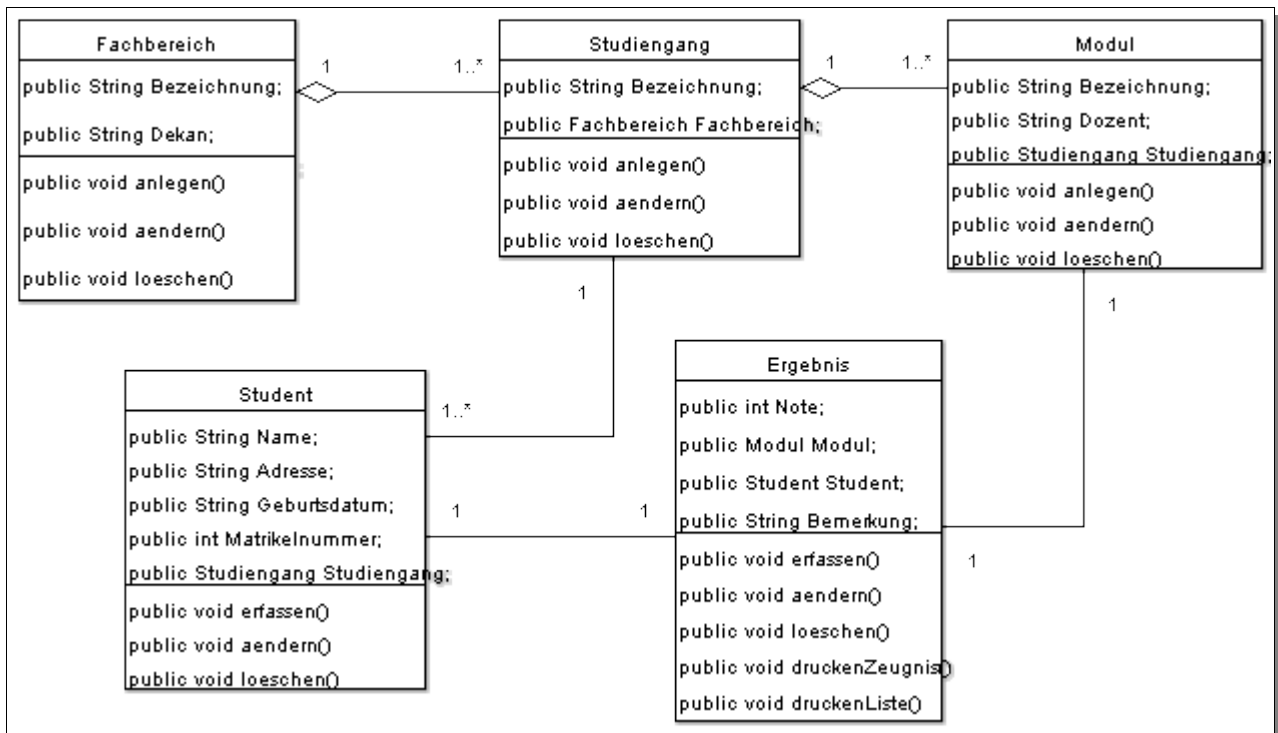


Ergebnis



4 Entwurfs-/Designphase (OOD)

4.1 Klassendiagramm mit Assoziationen / Kardinalitäten



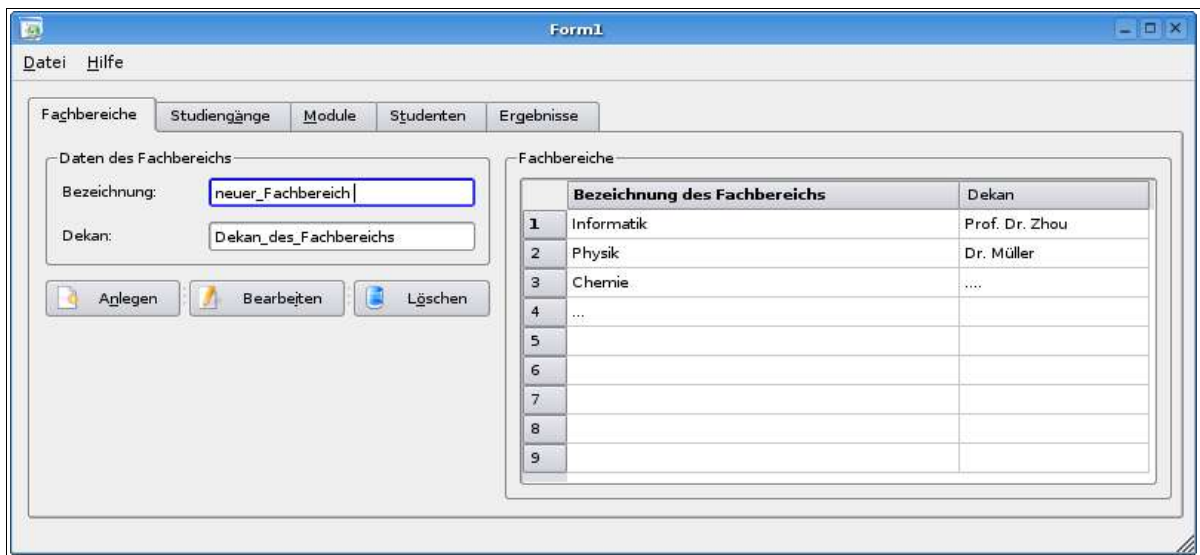
- Ein Fachbereich besteht aus einem oder mehreren Studiengängen.
- Ein Studiengang besteht aus einem oder mehreren Modulen.
- Mehrere Studenten sind einem Studiengang zugeordnet.
- Jedes Ergebniss (mit u.a. "Note" als Attribut) ist genau einem Studenten und einem Modul zugeordnet.
- Über das Ergebniss-Attribut "Bemerkung" kann vermerkt werden, ob es sich um eine wiederholte Prüfung (z.B. "2. Wiederholung") handelt.

5 Implementierungsphase (Prototyp)

5.1 Anmeldedialog



5.2 Hauptfenster



Form1

Datei Hilfe

Fachbereiche Studiengänge **Module** Studenten Ergebnisse

Daten des Moduls

Bezeichnung:

Dozent:

Studiengang:

Anlegen Bearbeiten Löschen

Module

| | Bezeichnung des Moduls | Dozent | Studiengang |
|---|---------------------------|----------------|-------------|
| 1 | Bussysteme und Interfaces | Hr. B. Gerum | Informatik |
| 2 | Computergraphik | Prof. Dr. Zhou | Informatik |
| 3 | Software-Development | Hr. H. Sorber | Informatik |
| 4 | ... | ... | ... |
| 5 | ... | ... | ... |
| 6 | | | |
| 7 | | | |
| 8 | | | |
| 9 | | | |

Form1 <2>

Datei Hilfe

Fachbereiche Studiengänge **Module** **Studenten** Ergebnisse

Daten des Studenten

Matrikelnummer:

Studiengang:

Name, Vorname:

Strasse, HausNr.:

PLZ, Wohnort:

Geburtsdatum:

Anlegen Bearbeiten Löschen

Studenten

| | MatrikelNr. | Studiengang | Name | Adresse | Geburtsdatum |
|---|-------------|-------------|--------|---------|--------------|
| 1 | 01232323 | Informatik | Müller | Isny | 01.01.1970 |
| 2 | 03248532 | Informatik | Huber | Kempton | 01.01.1980 |
| 3 | ... | ... | ... | ... | ... |
| 4 | ... | ... | ... | ... | ... |
| 5 | | | | | |
| 6 | | | | | |
| 7 | | | | | |
| 8 | | | | | |

Form1

Datei Hilfe

Fachbereiche Studiengänge **Module** **Studenten** **Ergebnisse**

Daten des Ergebnisses

Student/MatrikelNr.:

Modul:

Note:

Bemerkung:

Anlegen Bearbeiten Löschen

Ergebnisse

| | Note | Student | Modul | Bemerkung |
|---|------|----------|-------------------|-----------------|
| 1 | 1,0 | Torvalds | Betriebssysteme-2 | - |
| 2 | 1,7 | Müller | Computergraphik | 1. Wiederholung |
| 3 | ... | ... | ... | ... |
| 4 | ... | ... | ... | ... |
| 5 | | | | |
| 6 | | | | |
| 7 | | | | |

Notenlisten drucken Zeugnisse drucken

5.3 Erläuterungen zum Hauptfenster

Die Oberfläche des Hauptfensters gliedert sich auf jedem der fünf Hauptfunktions-Bereiche (Karteireiter) grob in zwei Teile:

links das jeweilige Eingabeformular und

rechts die im jeweiligen Bereich vorhandenen Datensätze

In diesem Prototyp wurden anstelle der angedachten Datenbankbindung nur einfache Tabellen zur Visualisierung der Idee verwendet – damit ist aber nur mit Hilfe der obigen Bilder evtl. noch unklar wie Daten bearbeitet und gelöscht werden, deshalb folgende Erklärung:

Wenn im fertigen Produkt eine Zeile (=Datensatz) im rechten Bereich (Tabelle mit vorhandenen Datensätzen) mit der Maus angeklickt (=markiert) wird, werden die Attribute des angeklickten Datensatzes automatisch in die Eingabefelder im linken Bereich übernommen.

Dort könnten nun die Werte des Datensatzes geändert, und durch den Knopf "Bearbeiten" wieder in die rechte Datentabelle (=spätere DB) übernommen werden (Änderungen speichern).

Durch den "Löschen"-Knopf könnte der zuvor rechts ausgewählter Datensatz (der dann auch in den Eingabefeldern erscheint) aus der Datenbank gelöscht werden.