

PFLICHTENHEFT

IM STUDIENGANG INFORMATIONSTECHNIK
der
DUALEN HOCHSCHULE BADEN-WÜRTTEMBERG
STANDORT STUTTGART
SOFTWARE-ENGINEERING

Bewertungssystem

von:

STANISLAV SOKOL ,LOUIS STEINKAMP, DOMINIK ZIPPERLE

Datum: 12. November 2014

Inhaltsverzeichnis

| | |
|--|----|
| 1 Zielbestimmung | 4 |
| 1.1 Musskriterien | 4 |
| 1.2 Wunschkriterien | 4 |
| 1.3 Abgrenzungskriterien | 4 |
| 2 Produkteinsatz | 5 |
| 2.1 Anwendungsbereiche | 5 |
| 2.2 Zielgruppen | 5 |
| 2.3 Betriebsbedingungen | 5 |
| 3 Produktübersicht | 6 |
| 3.1 Use Case Analyse | 6 |
| 3.2 Use Cases in grafischer Darstellung | 8 |
| 4 Produktfunktionen | 9 |
| 5 Produktdaten | 13 |
| 6 Produktleistungen | 15 |
| 7 Qualitätsanforderungen | 16 |
| 8 Benutzeroberfläche | 18 |
| 9 Nichtfunktionale Anforderungen | 19 |
| 10 Technische Produktumgebung | 20 |
| 10.1 Software | 20 |
| 10.2 Hardware | 20 |
| 10.3 Orgware | 20 |
| 10.4 Produkt-Schnittstellen | 20 |
| 11 Spezielle Anforderungen an die Entwicklungs-Umgebung | 21 |
| 11.1 Software | 21 |
| 11.2 Hardware | 21 |
| 11.3 Orgware | 21 |
| 11.4 Entwicklungs-Schnittstellen | 21 |
| 12 Gliederung in Teilprodukte | 22 |
| 13 Ergänzungen | 23 |
| A Begriffsdefinitionen | 23 |
| B Abkürzungen | 24 |
| C Modelle | 25 |
| C.1 Geschäftsprozesse | 25 |
| C.2 Data Flow Diagramm | 27 |
| C.3 Structured Analysis | 28 |
| D Qualitätsmerkmale | 30 |

| | | |
|----------|----------------------------|----|
| E | Aufwandsabschätzung | 31 |
|----------|----------------------------|----|

1 Zielbestimmung

1.1 Musskriterien

- Verschiedene Benutzergruppen
- Benutzer anlegen
- Rechtehierarchien
- Studenten können ihre Noten in dem System einsehen.
- Notenkonvertierungsprofile erstellen und verwalten
- Dozenten weisen einer Veranstaltung Bewertungsschema zu
- Ein Template kann zu einer Veranstaltung erstellt werden.
- Veranstaltungen des Typs (Gruppenarbeit, Einzelarbeit) können erstellt werden.
- Die Festlegung eines Grundscore (Gruppe, Einzeln) lässt sich in zwei skalierbare Bereiche beliebig teilen (H_2)
- Score Refining Faktoren können in einem Template frei eingestellt werden (Items) (R_2S)
- Skalierbarkeit des Bewertungssystems von Bewertung der Gruppe zu Einzelperson und umgekehrt.
- Zuweisung von beliebig gruppierten Studenten zu einer
- Veranstaltung durch Dozent oder Prüfer.

1.2 Wunschkriterien

- Alle Zwischenbewertungen die zur Note führen werden dem Studenten angezeigt.
- Profilverwaltung um standardisierte Bewertungen erzeugen zu können.

1.3 Abgrenzungskriterien

- Bewertungssystem bewertet nur die Veranstaltungen die einer Art bestanden/nicht bestanden Logik unterliegen. Anschließend erfolgt eine konfigurierbare Punkte-Notenkonvertierung.
- Jede erstellte Veranstaltung beinhaltet genau ein Bewertungsschema und führt somit zu einer Note. Die Kombinierbarkeit von verschiedenen Einzelnoten ist nicht möglich.

2 Produkteinsatz

2.1 Anwendungsbereiche

- Planung einer Veranstaltung mit anstehenden Prüfungsleistungen für Lehrveranstaltungsplaner
- Errechnung einer Note durch den Prüfer anhand festgelegter Kriterien.
- Einsicht der Noten durch Studenten, sowie die Leistungsübersicht für die Dozenten und Prüfer.

2.2 Zielgruppen

- Administrator (Verwalter des Gesamtsystems)
- Dozent
- Prüfer
- Studenten

2.3 Betriebsbedingungen

- Das System soll nur einmal angelegt werden und autark funktionieren.
- Klar definierte Benutzergruppen und Rechte
- Datenbankschnittstelle für die Benutzerverwaltung (mögliche Anbindung an Dualis bei vorhandener API)
- Datenbank-Umgebung für die Speicherung der Daten
- Archivierung der Daten (Datensicherheit wegen Einsehbarkeit der Daten)
- Zugriff von außen sowohl als Lesen, als auch als Edit möglich.
- Die Anwendung soll parallelen Zugriff von mehreren Benutzern zulassen.
- Der Administrator kann die Datenbank sichern und die Programmdaten ändern. Er hat keine Lese-/Schreibrechte auf die Datenbankinhalte.

3 Produktübersicht

Das Produkt wird durch folgende Use Cases beschrieben. Alle Use Cases beziehen sich auf das Verhalten des Systems aus der Sicht eines außenstehenden Anwenders. Die Use Cases beschreiben die Hauptfunktionen, welche ein Benutzer des Systems wahrnehmen kann. Die Anforderung an die Funktionalität entsteht aus der Auswertung der Akteur/Produkt- und Akteur/Akteur-Beziehungen im Kapitel 4.

3.1 Use Case Analyse

| | |
|-------------------------------------|--|
| Use Case ID | 1 |
| Elementarer Geschäftsprozess | Veranstaltung planen |
| Ziel des Use Cases | Darstellung des Prozesses einer Veranstaltungsplanung |
| Umgebende Systemgrenze | Veranstaltungskonfigurator |
| Vorbedingung | Dozent ist angemeldet |
| Nachbedingung Erfolg | Veranstaltung ist angelegt |
| Beteiligte Nutzer: | Dozent |
| Auslösendes Ereignis: | Dozent möchte Veranstaltung planen |
| Use Case ID | 2 |
| Elementarer Geschäftsprozess | Template erstellen |
| Ziel des Use Cases | Templates für die Veranstaltungen können angelegt werden |
| Umgebende Systemgrenze | Veranstaltungskonfigurator |
| Vorbedingung | Dozent ist angemeldet |
| Nachbedingung Erfolg | Template ist erstellt |
| Beteiligte Nutzer: | Dozent |
| Auslösendes Ereignis: | Dozent möchte Template erstellen |
| Use Case ID | 3 |
| Elementarer Geschäftsprozess | Bewerten |
| Ziel des Use Cases | Prüfer bewertet |
| Umgebende Systemgrenze | Bewertung |
| Vorbedingung | Veranstaltung existiert |
| Nachbedingung Erfolg | Studenten/Gruppen existieren Teilbewertung liegt vor Bewertung ist abgeschlossen |
| Beteiligte Nutzer: | Prüfer |
| Auslösendes Ereignis: | Prüfer will Gruppe/Student bewerten |

| | |
|-------------------------------------|--|
| Use Case ID | 4 |
| Elementarer Geschäftsprozess | Einsehen |
| Ziel des Use Cases | Studenten sehen ihre Noten ein |
| Umgebende Systemgrenze | Bewertung |
| Vorbedingung | Bewertung liegt bereits vor |
| Nachbedingung Erfolg | Student sieht die Note |
| Beteiligte Nutzer: | Student |
| Auslösendes Ereignis: | Student will seine Note sehen |
| | |
| Use Case ID | 5 |
| Elementarer Geschäftsprozess | Teilnehmer anlegen |
| Ziel des Use Cases | Prüfer oder Dozent konfigurieren die Teilnehmer einer Veranstaltung |
| Umgebende Systemgrenze | Planung Prüfung |
| Vorbedingung | Veranstaltung ist konfiguriert |
| Nachbedingung Erfolg | Veranstaltung ist freigegeben |
| Beteiligte Nutzer: | Prüfer, Dozent |
| Auslösendes Ereignis: | Studenten sollen einer Veranstaltung hinzugefügt werden |

3.2 Use Cases in grafischer Darstellung

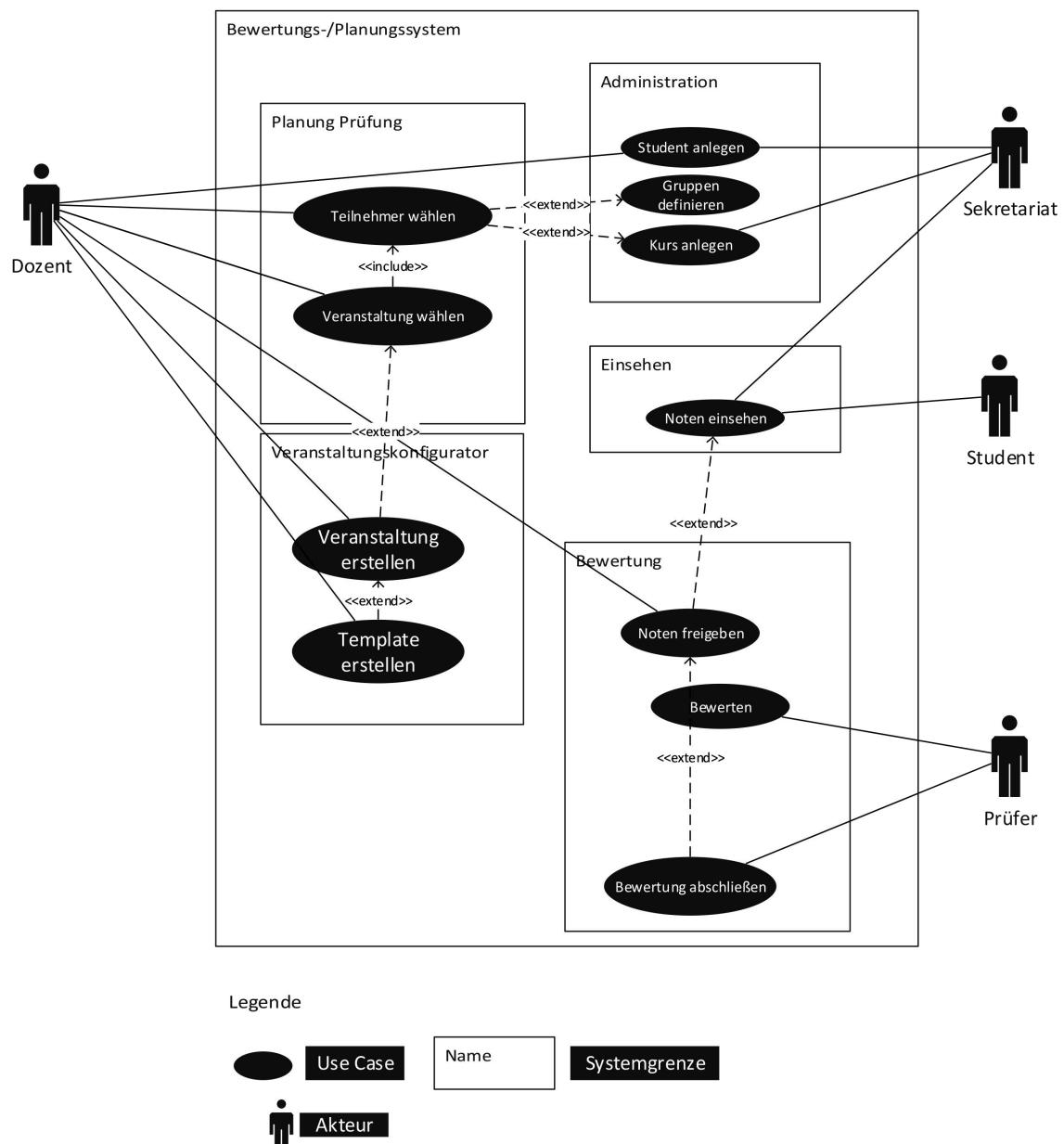


Abbildung 1: Darstellung des Gesamtsystems anhand der ausgearbeiteten Use-Cases

4 Produktfunktionen

Im folgenden sind die Geschäftsprozesse der Anwendung dokumentiert.

Diese Produktfunktionen werden im Laufe der Entwicklung angepasst und in ihrem Umfang erweitert. Die Geschäftsprozesse finden sich als Diagramme im Anhang (Abb. 3, 4, 5)

/F10/

| | |
|-----------------------------|---|
| Geschäftsprozess | Bewertungsschema festlegen |
| Kategorie | primär |
| Vorbedingung | 1. Dozent ist angemeldet |
| Nachbedingung Erfolg | Ein Bewertungsschema steht zur Verfügung Prüfer erfährt welchen Kurs (bzw. Student/en) er zu prüfen hat. |
| Akteure | Dozent |
| Auslösendes Ereignis | Dozent legt ein Bewertungsschema an |
| Beschreibung | 1. Umrechnungsformel festlegen 2. Parameter anpassen |

/F20/

| | |
|---------------------------------|---|
| Geschäftsprozess | Template erstellen |
| Kategorie | primär |
| Vorbedingung | Dozent ist angemeldet ein Bewertungsschema liegt vor |
| Nachbedingung Erfolg | Template ist zur Veranstaltungsplanung freigegeben |
| Nachbedingung Fehlschlag | |
| Akteure | Dozent |
| Auslösendes Ereignis | Dozent legt ein Template an Bei Veranstaltungserstellung wurde kein Template gefunden Bei Veranstaltungserstellung wird ein neues Template erstellt |
| Beschreibung | 1. Die Art der Prüfung wird spezifiziert 2. Items hinzufügen 3. Bewertungsschema hinzufügen |
| Erweiterung | 1. Templates refactoring |
| Alternativen | 1. Neues Bewertungsschema erstellen anlegen |

/F30/

| | |
|---------------------------------|---|
| Geschäftsprozess | Veranstaltung planen |
| Kategorie | primär |
| Vorbedingung | Dozent ist angemeldet ein Template liegt vor Bewertungsschema liegt vor Prüfer liegen vor |
| Nachbedingung Erfolg | Eine Veranstaltung ist angelegt Prüfer erfährt welchen Kurs (bzw. Student/en) er zu prüfen hat. |
| Nachbedingung Fehlschlag | |
| Akteure | Dozent |
| Auslösendes Ereignis | Dozent legt eine Veranstaltung an |
| Beschreibung | 1. Dozent konfiguriert das Template 2. Dozent befüllt(prüft) die Parameter des Bewertungsschemas 3. Dozent kann Prüfer einer Veranstaltung zuteilen zuteilen 4. Dozent kann Studenten/Gruppen der Veranstaltung zuweisen |
| Erweiterung | |
| Alternativen | 1. Neues Template anlegen 2. Neue Studenten anlegen 3. Neue Prüfer Anlegen 4. Neues Bewertungsschema anlegen |

/F40/

| | |
|-----------------------------|---|
| Geschäftsprozess | Bewerten |
| Kategorie | primär |
| Vorbedingung | 1. Dozent/Prüfer ist angemeldet 2. Veranstaltung ist angelegt 3. Gruppen/Student ist einer Veranstaltung zugewiesen 4. Prüfer ist einer Veranstaltung zugewiesen |
| Nachbedingung Erfolg | Ein Student/Gruppe wurde bewertet Teilbewertung/vollständige Bewertung ist abgeschlossen |
| Akteure | Dozent Prüfer |
| Auslösendes Ereignis | Prüfer will bewerten |
| Beschreibung | 1. Prüfer wählt Veranstaltung aus Prüfer wählt Student/Gruppe zur Bewertung aus Bewertungen/Teilbewertungen eintragen Bewertung abschließen |

/F50/

| | |
|---------------------------------|--|
| Geschäftsprozess | Bewertung abschließen |
| Kategorie | primär |
| Vorbedingung | Bewertungen einer Veranstaltung in allen Items liegen vor Bewertungen für einen Student/Gruppe sind komplett Prüfer ist angemeldet |
| Nachbedingung Erfolg | Dozent erfährt, dass die Endnote der Veranstaltung vorliegt Prüfer verliert das Recht zu editieren, nur noch sehen Nur Dozent ist berechtigt zu ändern |
| Nachbedingung Fehlschlag | Die Bewertung steht weiter aus Bestätigung für den Abschluss trotz der unvollständigen Daten wird erfragt |
| Akteure | Dozent Prüfer |
| Auslösendes Ereignis | Prüfer schließt die Bewertung ab |
| Alternativen | 1. Bewertung vervollständigen |

/F60/

| | |
|---------------------------------|---|
| Geschäftsprozess | Kurs anlegen |
| Kategorie | primär |
| Vorbedingung | 1. Dozent ist angemeldet Studenten liegen vor |
| Nachbedingung Erfolg | Kurs ist angelegt |
| Nachbedingung Fehlschlag | - |
| Akteure | Dozent |
| Auslösendes Ereignis | Dozent legt einen Kurs an |
| Beschreibung | 1. Kurs erstellen 2. Studenten hinzufügen 3. Kurs erstellen |
| Alternativen | 1. Studenten anlegen |

/F70/

| | |
|---------------------------------|---|
| Geschäftsprozess | Studenten anlegen |
| Kategorie | primär |
| Vorbedingung | 1. Dozent ist angemeldet |
| Nachbedingung Erfolg | Ein Student ist angelegt Student kann einem Kurs/Gruppe hinzugefügt werden |
| Nachbedingung Fehlschlag | - |
| Akteure | Dozent |
| Auslösendes Ereignis | Dozent legt Studenten an |
| Beschreibung | 1. Studentenregisterkarte erstellen 2. Daten befüllen |

/F80/

| | |
|---------------------------------|---|
| Geschäftsprozess | Bewertung einsehen |
| Kategorie | primär |
| Vorbedingung | 1. Student ist angemeldet 2. Bewertung liegt vor |
| Nachbedingung Erfolg | 1. Student sieht seine Noten |
| Nachbedingung Fehlschlag | Benachrichtigung |
| Akteure | Student |
| Auslösendes Ereignis | Student will seine Noten einsehen |
| Beschreibung | 1. Student meldet sich an 2. Student sieht seine Noten |

5 Produktdaten

Die Größe der einzelnen Datenpunkte richtet sich variabel nach der letztendlichen Größe des Anwendungssystems. Die praktische Obergrenze wird durch die Hardware bzw. Datenbank festgelegt.

Die unten aufgeführten Produktdaten in der Notation /Dxx/ beschreiben die Gesamtheit der Daten, welche die Anwendung benötigt oder erstellt. Das vorläufige Datenbankmodell ist in der Abbildung 2 dargestellt, erst in dem ER-Modell werden die Trennungen in die Tabellen vorgenommen.

/D10/ Benutzer {Benutzer_id, Benutzertyp(Rechte) , Personendaten, Veranstaltungen} (Max. 100000)

/D20/ Veranstaltung {Veranstaltungs_id, Bewertungsschema(S₂G), Bewertungstemplate, Prüfer, Studenten, Score, Note, Status_Bewertung Beschreibung} (Max. 100000)

/D30/ Datenarchiv (Alle Daten)

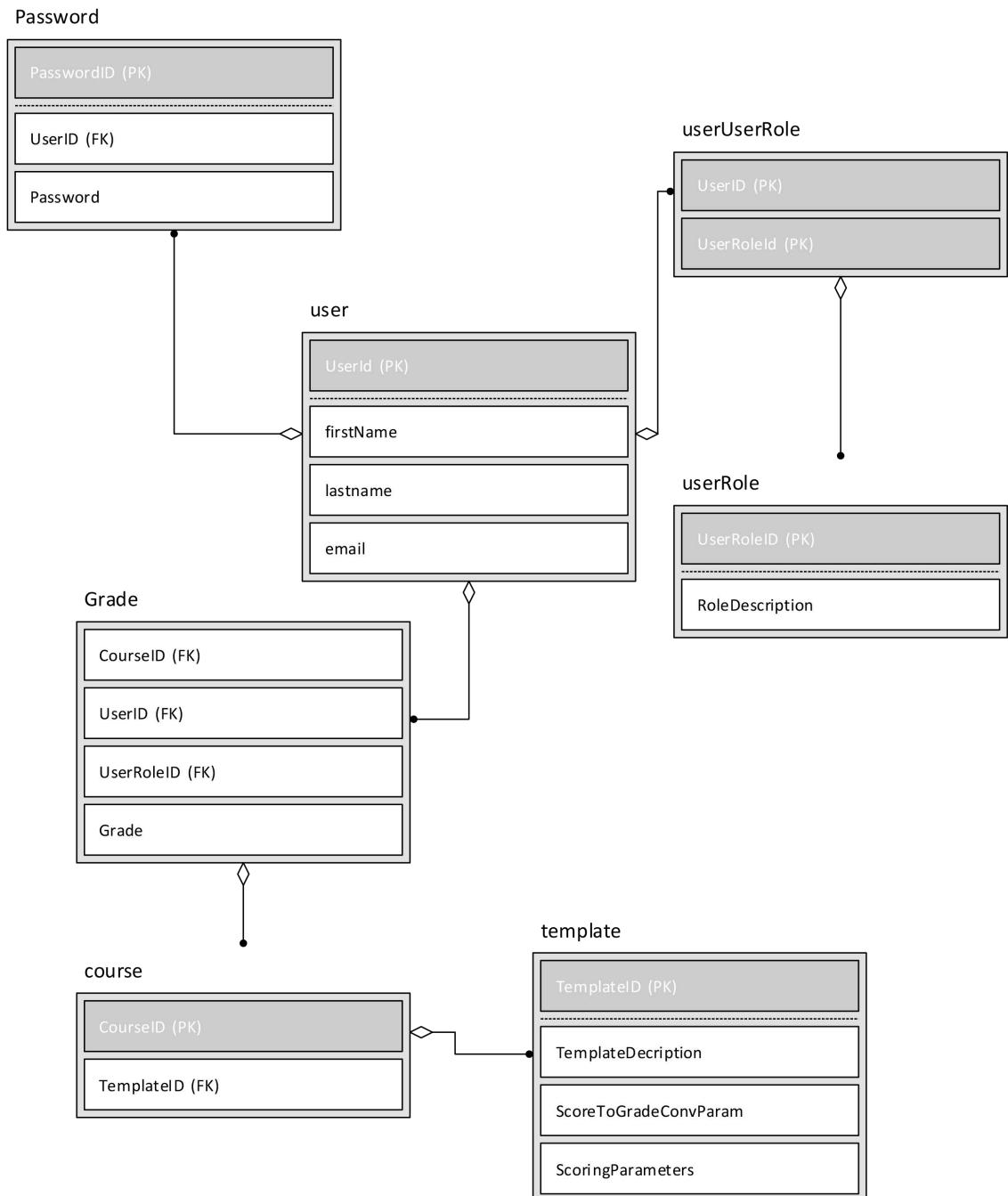


Abbildung 2: Entity Relation Modell in IDEF1X Notation

6 Produktleistungen

- /L10/ Die Antwortzeiten der Anwendung sollen so gering wie möglich gehalten werden.
- /L20/ Die steigenden Benutzerzahlen die zur gleichen Zeit auf das System zugreifen, sollen keine signifikanten Einflüsse auf die Antwortzeiten des Systems haben.
- /L30/ Benutzerfreundliche UX soll implementiert werden.
- /L40/ Die Anwendung besitzt eine eigene Datenbank zur Speicherung und Verwaltung der Anwendungsdaten. Die Ausgliederung einzelner Teilmodule, wie z.B. die Benutzerverwaltung ist möglich (API muss von Kunden bereitgestellt werden).
- /L50/ Die Abfragen und die Einsicht der Daten von Außerhalb (WWW) ist entsprechend der Benutzerberechtigung möglich. (wird im Prototypen zu einem späteren Zeitpunkt realisiert)
- /L60/ Datenbeständigkeit soll garantiert werden.

7 Qualitätsanforderungen

Tabelle 1: Qualitätsmerkmale nach DIN ISO 9126 – siehe Anhang D in T3-4

| Produktqualität | sehr gut | gut | normal | nicht relevant |
|-----------------------|----------|-----|--------|----------------|
| Funktionalität | | | | |
| Angemessenheit | | | ✓ | |
| Richtigkeit | | ✓ | | |
| Interoperabilität | | | ✓ | |
| Ordnungsmäßigkeit | | | ✓ | |
| Sicherheit | | ✓ | | |
| Zuverlässigkeit | | | | |
| Reife | | | ✓ | |
| Fehlertoleranz | | ✓ | | |
| Wiederherstellbarkeit | ✓ | | | |
| Benutzbarkeit | | | | |
| Verständlichkeit | | | ✓ | |
| Erlernbarkeit | | | | ✓ |
| Bedienbarkeit | | | ✓ | |
| Effizienz | | | | |
| Zeitverhalten | | | ✓ | |
| Verbrauchsverhalten | | | ✓ | |
| Änderbarkeit | | | | |
| Analysierbarkeit | | | ✓ | |
| Modifizierbarkeit | | | ✓ | |
| Stabilität | | | ✓ | |
| Prüfbarkeit | | | ✓ | |
| Übertragbarkeit | | | | |
| Anpassbarkeit | | | ✓ | |
| Installierbarkeit | | | ✓ | |
| Konformität | | | ✓ | |
| Austauschbarkeit | | | | ✓ |

Richtigkeit

Da das System mit persönlichen und sensiblen Daten arbeitet, sollte dieses eine hohe Richtigkeit haben. Die Tests auf die Richtigkeit und die Robustheit des Systems sind mit dem Entwicklungsprozess verflochten und basieren auf den vorgestellten Benutzerszenarien (Agile Development).

Sicherheit

Die Sicherheitsaspekte der einzelnen Module werden im Lauf der Entwicklung integriert. Im Prototyp wird für das Anmeldeverfahren per Passwort der Sicherheitsstandard PKCS#5 (PBKDF2, RFC2898) verwendet und die Sichten der einzelnen Bereiche der Anwendung strikt an die Benutzerberechtigung gebunden. Für die Umsetzung der Anwendung soll eine gesicherte Verbindung über HTTPS (SSH, RS2818) bereit gestellt werden.

Fehlertoleranz

Bei der Berechnung der Noten sollten so wenige Fehler wie möglich auftreten

Wiederherstellbarkeit Die Daten des Systems müssen entsprechend der jeweiligen Anforderung der Hochschule über einen längeren Zeitraum einsehbar und wiederherstellbar sein. Da alle benutzerbezogenen Daten in der Datenbank gespeichert und verwaltet werden bezieht sich die Wiederherstellbarkeit auf der MYSQL-Standard. Die Entwicklung der Backupstrategie folgt im weiteren Verlauf der Entwicklung.

8 Benutzeroberfläche

Die Benutzeroberfläche entspricht den modernen Ansprüchen einer Web-Anwendungen und wird mit entsprechend Google-Designrichtlinien erstellt. Die intuitive Bedienung soll das Leitmotiv der Entwicklung werden.

Die genauere Spezifikation der grafischen Benutzeroberfläche wird zu einem späteren Zeitpunkt des Projektes definiert.

Tabelle 2: Benutzergruppen und Rechteverteilung

| Benutzergruppe | Lesen | Schreiben | Ändern | Systemanpassungen |
|----------------|-------|-----------|--------|-------------------|
| Administrator | ✓ | ✓ | ✓ | ✓ |
| Verwaltung | ✓ | ✓ | ✓ | |
| Prüfer | ✓ | ✓ | | |
| Student | ✓ | | | |

9 Nichtfunktionale Anforderungen

- /Q10/** Das System muss die Rollen, wie in der Tabelle 2 dargestellt, unterscheiden und die Rechteverteilung sicherstellen können.
- /Q20/** Für die im System gespeicherten Daten muss gewährleistet werden, dass deren Sicherung zuverlässig passiert.

10 Technische Produktumgebung

Das System läuft auf einem Server, der die Anfragen der webbasierten Clients entgegennehmen kann.

Die GUI für den Web-Browser wird erst im nächsten Entwicklungsstadium bereitgestellt.

10.1 Software

Server-Betriebssystem: UNIX

Clientsysteme: Browser

Web-Server: Jetty (bei Bedarf)

10.2 Hardware

Sind für dieses Projekt nicht relevant. (prinzipiell ein leistungsstarker Server)

Für den Prototypen ist nur eine eingeschränkte Funktionalität vorgesehen.

10.3 Orgware

Zunächst ist das System nur durch freigegebene Benutzer erreichbar.

10.4 Produkt-Schnittstellen

Web-Schnittstelle, Serverschnittstellen

11 Spezielle Anforderungen an die Entwicklungs-Umgebung

11.1 Software

Git Versionsverwaltung, Issue Tracking System - YouTrack, IDE für JAVA Development, Database Development Tools

11.2 Hardware

Laptop, Standalone PC (Server, Client)

11.3 Orgware

XP (extreme programming und agile development)

11.4 Entwicklungs-Schnittstellen

WEB-Schnittstelle

12 Gliederung in Teilprodukte

Der Prototyp besteht aus einem Veranstaltungskonfigurator, einem Bewertungssystem und einem Modul zum Einsehen der Bewertung.

Veranstaltungskonfigurator beschränkt sich hierbei auf einen minimalen Funktionsumfang um die mögliche Arbeitsweise deutlich zu machen.

Konfigurieren bedeutet in diesem Zusammenhang die Notenumrechnung mit Parametern zu belegen und zwischen unterschiedlichen Bewertungsschemata auswählen zu können.

Des Weiteren lassen sich zu einer Veranstaltung einzeln gewichtete Bewertungskriterien hinzufügen.

Das Bewertungssystem beinhaltet eine vom Dozent vordefinierte Maske, in die der Prüfer die Ergebnisse der Veranstaltung eintragen kann. Anhand der Ergebnisse wird dann eine Note für diese Veranstaltung errechnet. Über ein Modul zum Einsehen der Bewertung, können zum einen Dozent und Prüfer die eingegebenen Ergebnisse überprüfen, zum anderen die Studenten ihr Abschneiden bei der jeweiligen Veranstaltung ansehen.

13 Ergänzungen

A Begriffsdefinitionen

| | |
|---|--|
| Benutzergruppen | Mögliche Benutzer werden mit verschiedenen Rechten und Bedienszenarien versehen. |
| Student | Eine Benutzergruppe mit stark eingeschränkten Zugriffssrechten. Kann nur die vom Dozent freigegebenen Informationen sehen. |
| Dozent | Eine Benutzergruppe mit umfassenden Rechten. Kann Templates, Veranstaltungen und Benutzergruppen erstellen und verwalten. |
| Prüfer | Ein Benutzer mit Rechten zur Dateneingabe. Kann nichts an einer Veranstaltung ändern. Muss sich an die vom Dozent vorgegeben Muster halten. |
| Sekretariat | Benutzergruppe, welche die Rechte zur Datenverwaltung besitzt (kein Löschrecht). |
| Administrator | Eine Benutzergruppe mit der Berechtigung zur systemnahen Anpassungen. |
| Veranstaltung | Template + Bewertungsschema + angepasste Items + Studenten/Gruppen + Kurs + Prüfer . Eine Veranstaltung umfasst die obengenannte Bausteine und wird vom Dozenten freigegeben. |
| Template | Ein Template enthält eine beliebige Anzahl von Items und evtl. ein oder mehrere zugewiesene Bewertungsschemata. Ein Template darf leer sein. Somit werden komplett neue Veranstaltungen planbar. |
| Item | Ein entweder fest oder variabel angelegtes Bewertungskriterium. Diese Kriterien können für eine Veranstaltung angepasst werden. |
| Notenkonvertierungsprofil (S₂G) | Dies ist eine Umrechnungsformel, die sich aus der Skala, wie der Score in eine Note umgerechnet werden soll, bestimmt. Das konkrete Profil kann für eine Veranstaltung durch Eingangsparameter angepasst werden. |
| Score | Punkte die eine Basis für die Notenbildung darstellen. |
| Grundscore (H₂) | Zweistufige holistische Bewertung. Diese zwei Bereiche sind frei einstellbar. |
| Score Refining (S₂R) | Ein Schema zur Aufbereitung des Grundscore durch Faktoren (refining). Die Faktoren werden vom Dozenten vordefiniert. |

B Abkürzungen

- API** Application Programming Interface. Externe Programmschnittstelle, hier für Anbindung an externe Datenbank.
- S₂G* Punkte zu Note (score to grade) Konvertierung anhand festgelegter Formel und Parameter.
- R₂S* Wertung zu Punkten (rate to score). Punkte werden anhand manuell eingegebener Rates errechnet.
- H₂* Hollistisches Bewertungsmodul. Zwei Bereiche (frei konfigurierbar). Ausgang ist eine Score, welche weiter in *R₂S*.

C Modelle

C.1 Geschäftsprozesse

Aus den Use Cases in Abbildung 1 ergeben sich die folgende Geschäftsprozesse. Die Diagramme beschreiben systeminterne Abläufe. Die im folgenden benutzte Nomenklatur entspricht der EPK (Ereignisgesteuerte Prozesskette) Syntax.

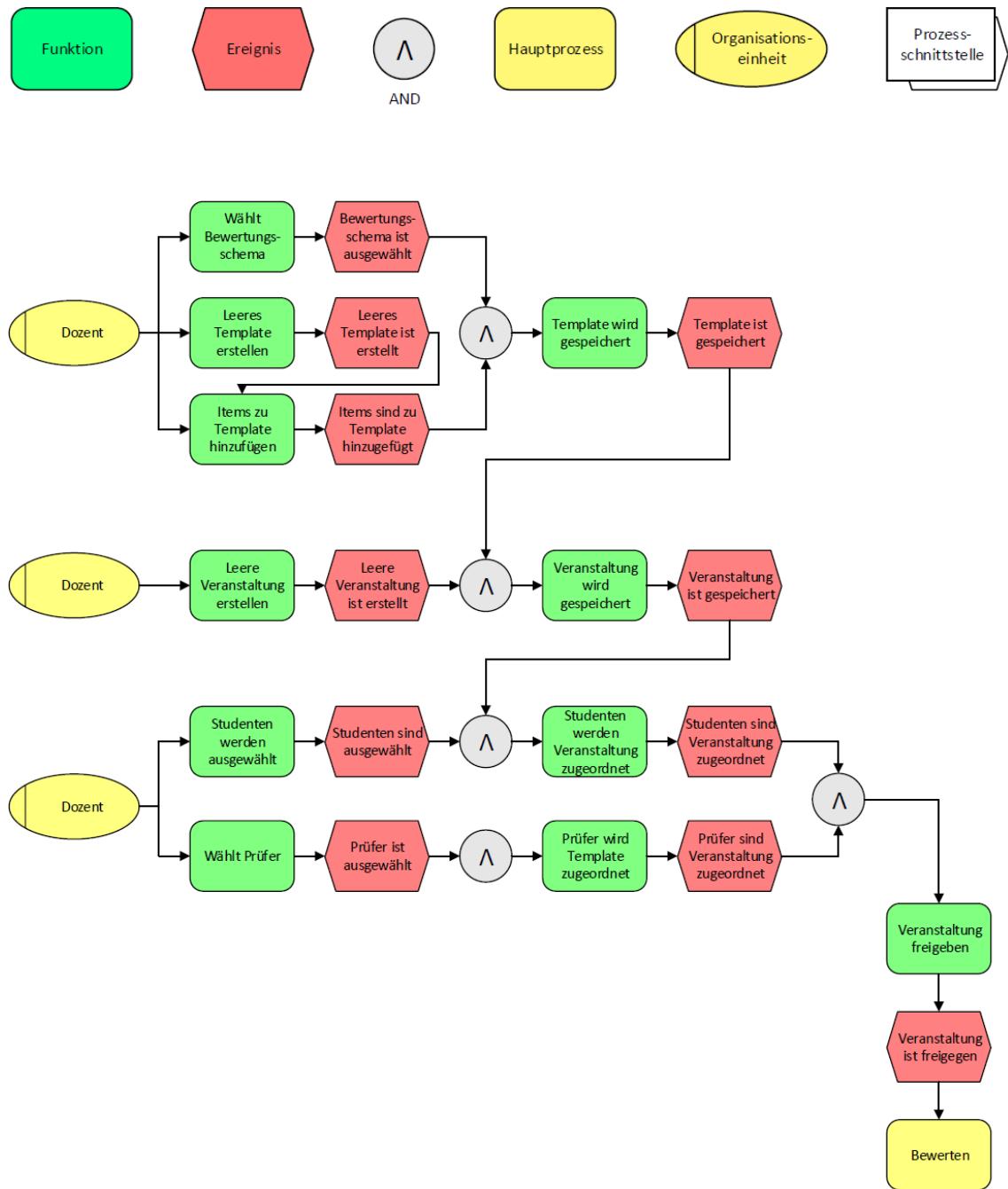


Abbildung 3: Prozessdiagramm für den Geschäftsprozess Veranstaltungserstellung

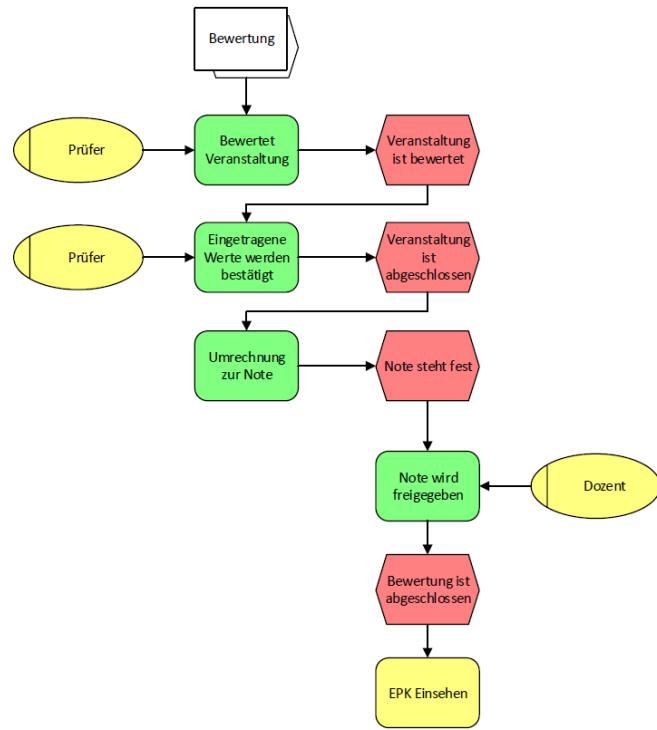


Abbildung 4: Ablaufdiagramm für den Geschäftsprozess Bewertung

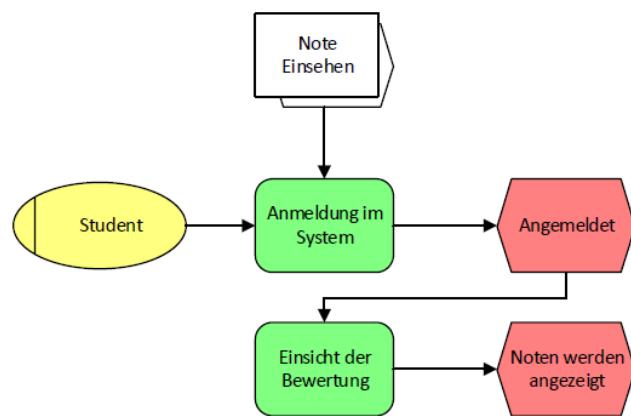


Abbildung 5: Ablaufdiagramm für den Geschäftsprozess Einsehen der Noten

C.2 Data Flow Diagramm

Die Abbildung 6 stellt den groben Datenfluss für die gesamte Anwendung dar:

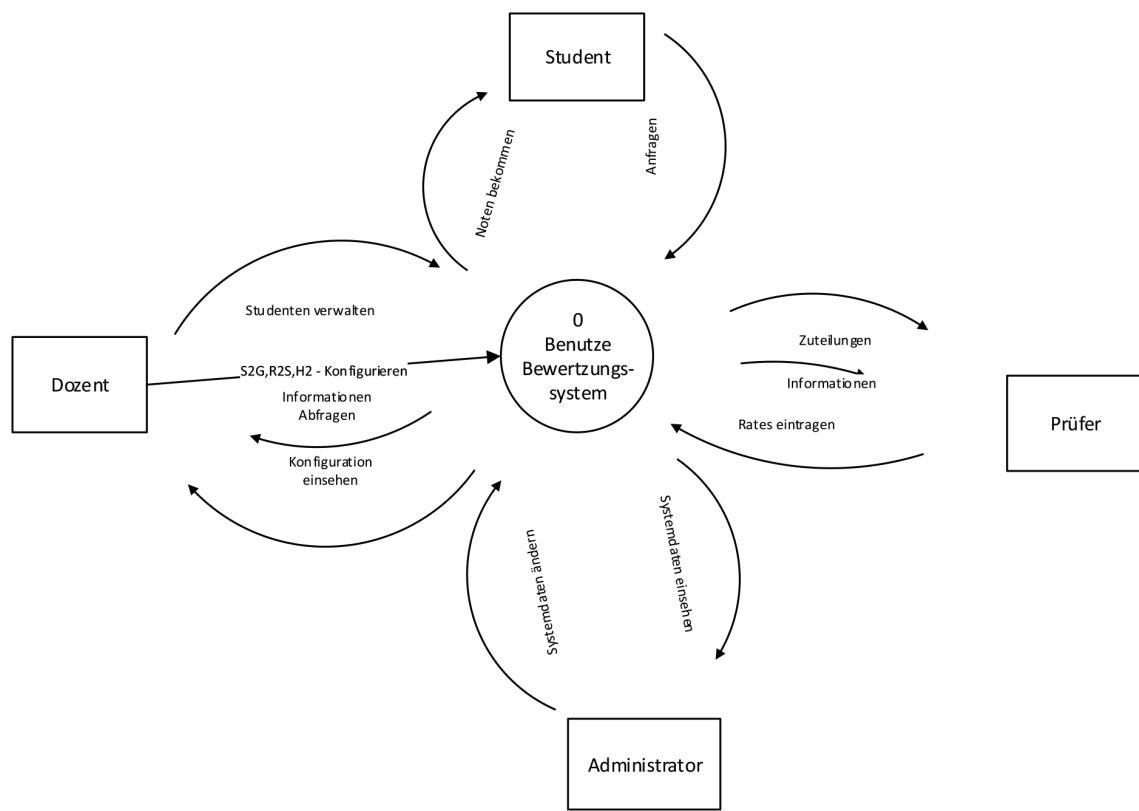


Abbildung 6: Kontextdiagramm der Software

C.3 Structured Analysis

Ausgehend von Abbildung 6 kann die Struktur des Datenflusses folgendermaßen spezifiziert werden. Dabei werden die Datenbewegungen innerhalb der verschiedenen Softwareschichten analysiert. In der Abbildung 7 wird der Login-Prozess abgebildet.

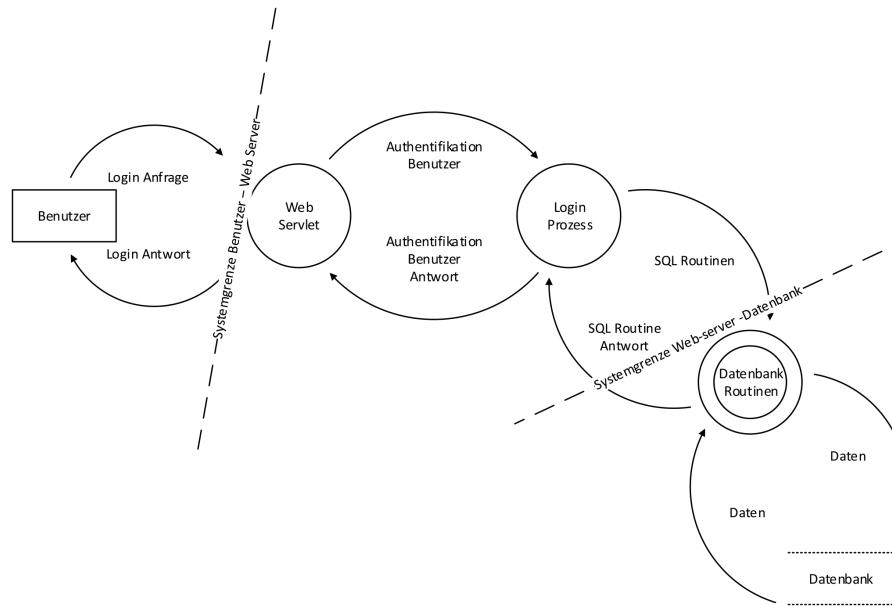


Abbildung 7: Datenfluss bei einem Login Prozess

In der Abbildung 8 ist der Datenfluss bei der Konfigurierung des Templates, eines Kurses oder einer Veranstaltung dargestellt.

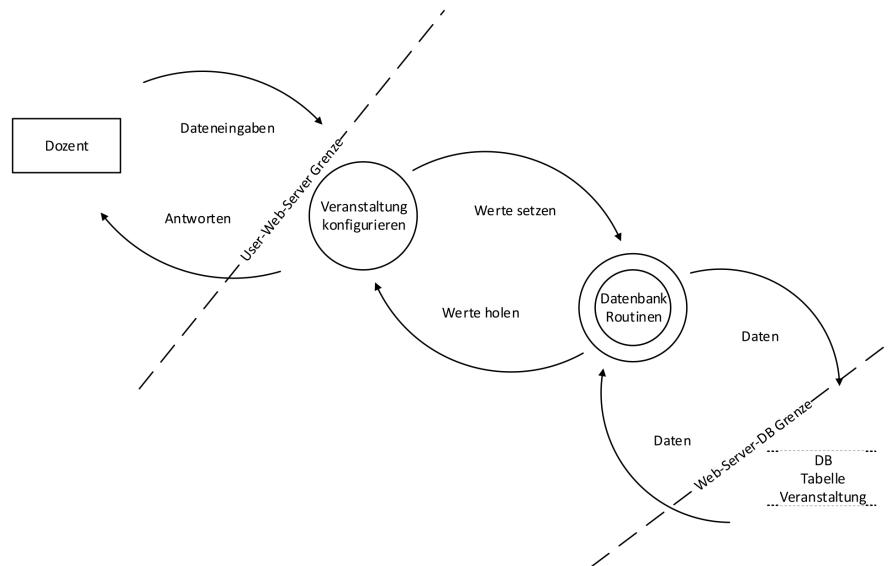


Abbildung 8: Datenfluss bei einem Konfigurierung-Prozess

In der Abbildung 9 ist der Datenfluss bei einem Bewertungsvorgang abgebildet.

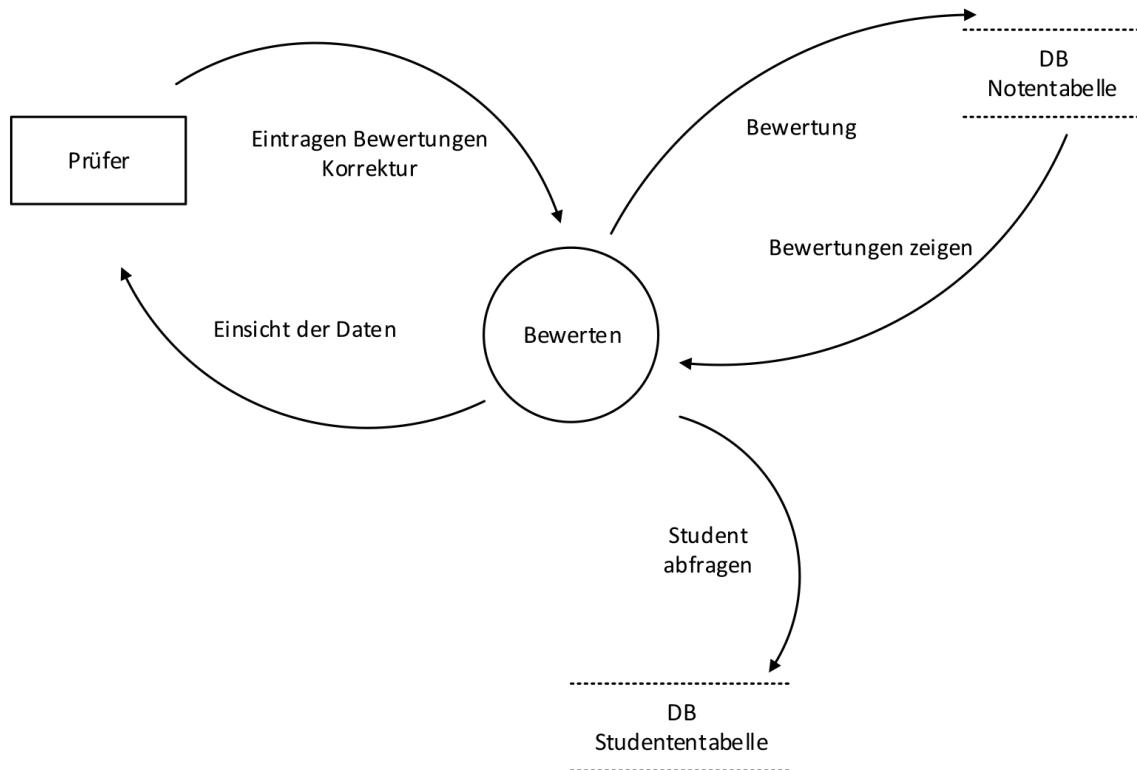


Abbildung 9: Datenfluss bei einem Bewertungsvorgang

D Qualitätsmerkmale

E Aufwandsabschätzung

| # | Kategorie | Klasse | Anzahl | Gewichtung | Beitrag |
|--|---------------|---------|--------|------------|-----------|
| 1 | Eingabedaten | einfach | 3 | x 3 = | 9 |
| | | mittel | | x 4 = | 0 |
| | | komplex | | x 6 = | 0 |
| 2 | Abfragen | einfach | 4 | x 3 = | 12 |
| | | mittel | | x 4 = | 0 |
| | | komplex | | x 6 = | 0 |
| 3 | Ausgaben | einfach | 4 | x 4 = | 16 |
| | | mittel | | x 5 = | 0 |
| | | komplex | | x 7 = | 0 |
| 4 | Datenbestände | einfach | 5 | x 7 = | 35 |
| | | mittel | | x 10 = | 0 |
| | | komplex | | x 15 = | 0 |
| 5 | Referenzdaten | einfach | 3 | x 5 = | 15 |
| | | mittel | | x 7 = | 0 |
| | | komplex | | x 10 = | 0 |
| uFP unbewertete Function Points | | | | = | 87 |

| Einflussfaktor mit Skala | | < | Gewichtung |
|--------------------------|---|------|-------------|
| 0 | Grundstock | | 70 |
| 1 | Verflechtung mit anderen Anwendungssystemen | 5 | 2,0 |
| 2 | Dezentrale Daten, dezentrale Verarbeitung | 5 | 2,0 |
| 3 | Transaktionsrate | 5 | 2,0 |
| Verarbeitungslogik | 4a Rechenoperationen | 10 | 5,0 |
| | 4b Kontrollverfahren | 5 | 2,0 |
| | 4c Ausnahmeregelungen | 10 | 4,0 |
| | 4d Logik | 5 | 3,0 |
| | 5 Wiederverwendbarkeit | 5 | 3,0 |
| | 6 Datenbestandskonvertierungen | 5 | 2,0 |
| | 7 Anpassbarkeit | 5 | 3,0 |
| E | Einflusskoeffizient | 0,01 | x 98 = 0,98 |

| | | | | | | |
|-----------|----------------------------------|----------|---|------------|---|--------------|
| FP | Bewertete Function Points | E | x | uFP | = | 85,26 |
|-----------|----------------------------------|----------|---|------------|---|--------------|

| | | | | |
|-----------|--|---|-------------|-----------------|
| PM | Umrechnung anhand der IBM-Tabelle | = | 7,12 | <i>P-Monate</i> |
|-----------|--|---|-------------|-----------------|

| | | | | | |
|--------------|---|---------------|---|-------------|-----------------|
| Dauer | Umrechnung nach Boehm's Formel für ... | Dialog | = | 4,97 | <i>K-Monate</i> |
|--------------|---|---------------|---|-------------|-----------------|

S = Stapelsysteme ; D = Dialogsysteme ; E = Echtzeitsysteme

| | | | | |
|--------------|---|---|-------------|--------------------|
| Größe | Umrechnung durch Division von PM durch Dauer | = | 1,43 | <i>Mitarbeiter</i> |
|--------------|---|---|-------------|--------------------|