Technische Dokumentation:

"Übersichten zum Lehrplan des ITI-Bereichs am Heinrich-Hertz-Berufskolleg"

|  |  |
| --- | --- |
| Teamnummer: | 3 |
|  |  |
| Temamitglieder: | Kenji Kokubo (Projektleiter) |
|  | Monika Schepan (Stellv. Projektleiter) |
|  | Felix Ehrenpfort |
|  | Christopher Schmitz |
|  |  |
| Ausbildungsberuf: | Fachinformatiker Anwendungsentwicklung |
|  |  |
| Datum der Abgabe: | 08.05.2017 |

# Ausgangssituation

## Umfeld

?

## Einleitung

Im Rahmen des Oberstufenprojekts des Heinrich-Hertz-Berufskollegs sollen Fertigkeiten in Projektmanagement und Teamarbeit getestet und gefördert werden. In Übereinstimmung mit unseren bisherigen Arbeitsaufträgen, gilt es nun eine Desktop-Applikation in JavaFX zu entwickeln. Der Verlauf des Projektes ist maßgeblich an die IHK-Abschlussprüfung angelehnt. Dazu gehört der Entwurf eines Pflichtenheftes unter Zunahme eines gegebenen Lastenheftes. Zudem ist das Erstellen eines Projektplans vorgesehen, welches die Arbeitsschritte im Projektverlauf wiederspiegeln soll. Es wird erwartet, dass der Projektplan bei unerwarteten Ereignissen, die die vorbestimmte Projektzeit beeinflussen, situationsgemäß angepasst bzw. aktualisiert wird. Anders als bei der IHK Projektdurchführung, wird das Projekt in Gruppen geführt. Dies bedeutet, dass unter den einzelnen Gruppenmitgliedern Teamleiter und Stellvertreter bestimmt werden müssen. Dem Projektplan muss daher auch ein Ressourcenplan beigelegt werden, sofern der Projektplan diesen Aspekt nicht bereits berücksichtigt. [Verweis auf Anhnge]

## Ausgangslage (Ist-Zustand)

Das Heinrich-Hertz-Berufskolleg verwendet zum Speichern von Informationen zu einzelnen Lehrgängen einen Web-Tool namens „Didaktischer Wizard“. In diesem werden Informationen zu einzelnen Ausbildungsberufen den Lehrkörpern zur Verfügung gestellt. Die Applikation bietet seinen Nutzern unter Angabe des Ausbildungsberufs und -jahres eine entsprechende Jahresübersicht. Diese ist interaktiv, das heißt, dass sogenannte Lernsituationen ausgewählt werden können. Es folgt die Darstellung einer neuen Übersicht mit detaillierten Informationen zu dieser Auswahl. Derzeit können diese einzelnen Ansichten, i.e. Jahresübersicht und Detailansicht einer gewählten Lernsituation als PDF generiert und gespeichert werden. Diese Funktion ist darauf beschränkt.

Die Aufgabe, die uns nun gestellt ist, ist das Generieren der Jahresübersicht und Detailansicht aller Lernsituation eines Ausbildungsberufes und -jahres in einzelnes PDF. Hierzu soll eine stand-alone Desktop-Applikation in Java entwickelt werden, die jedoch eine Fremdbeziehung zur Datenbank des Didaktischen Wizards haben soll.

## Projektziel (Soll-Zustand)

Ziel des Kunden ist eine Applikation die es ihr/ihm, sowie Auszubildende und Ausbilder ermöglichen Informationen zu einem Ausbildungsberufsjahr automatisch zu sammeln und diese in einem PDF zu erhalten.

Zusätzlich soll es einer ausgewählten Nutzergruppe möglich sein, „Templates“ zu verwalten. Mit „Template“ ist eine Konfiguration gemeint, die den Inhalt des besagten PDFs filtert. Ein Template kann so konfiguriert werden, dass es aus allen Detailübersichten nahezu alle bis keine Punkte entfernt. Sollte dem Nutzer (noch) kein Template zur Verfügung stehen, wird ein vordefiniertes Standardtemplate angeboten, dass ein Mindestmaß an Informationen bei der PDF Generierung lässt. Templates sollen in eine eigene Datenbank abgelegt werden.

Der Zugriff auf die Applikation soll durch eine eigene Benutzerverwaltung gesteuert werden. Auch wird erwartet, eine eigene Datenbank zum Speicher von Nutzerprofilen anzulegen.

## Kundenanforderung

Der Kunde erwartet nach Projektdurchlauf eine ausführbare JAR-Datei der gewünschten Applikation. Zudem sind die Skripte für die Datenbank der Benutzerverwaltung mit abzugeben. Als Schulprojekt sind zudem Softwarequalitäts- und Sicherheitsstandards einzuhalten.

Autorisierung und Authentifizierung

* Speichern von Benutzerprofilen auf der Datenbank
* Nutzerzugriff auf Applikation ist nur möglich sofern dieser sich mit einem gültigen Benutzernamen und Passwort angemeldet hat.
* Benutzerprofile sind nur durch ein vorbestimmtes Administrationsprofil änderbar.
* Gemäß dem festgelegten Autorisationslevel eines angemeldeten Nutzers werden Kontrollelemente ein- oder ausgeblendet. Untergestellten Nutzern darf es nicht möglich sein, Funktionen ihren übergeordneten Autorisationslevel auszuführen.
* Templates sind ihren Urhebern fest zugeordnet. Nur sein Urheber kann seine Templates nutzen und verwalten.

Templates

* Der Nutzer mit dem entsprechenden Autorisationslevel sollen über eine Auswahl an Konfigurationspunkten ein Template erstellen können. Unter Angabe eines Template-Namens soll dieses auch über mehrere Logins hinweg für seinen Urheber verfügbar sein.

## Änderungen zum Projektantrag

Arbeitsausfall, sowie eine fehleingeschätzte Recherchedauer machten einen Abstrich in den Kundenanforderungen unumgänglich. Es wurde entschieden, dass das Generieren der Jahresübersicht aus dem Projektplan gestrichen wird.

## Abgrenzung zu Fremdleistungen

Beim Gegenstand dieses Projektes handelt es sich im Wesentlichen um eine stand-alone Anwendung. Mit der Einschränkung, dass sie Fremddaten der „Didaktischen Wizard“-Datenbank beziehen muss, um PDFs zu generieren.

## Projektumfeld

* + 1. Projektinstanzen/Personalplanung

Anders als in einer IHK Projektdurchführung fallen in diesem Projekt Instanzen für die Projektgenehmigung und Qualitätskontrolle weg. Die übrigen Instanzen sind wie folgt:

|  |  |
| --- | --- |
| Bezeichnung | Name |
| Auftraggeber | Frau Dresen, Frau Brockmann-Wächter, Herr Ganser |
| Projektleitung | Kenji Kokubo |
| Ansprechpartner bezgl. Projektorganisation | Frau Dresen, Frau Brockmann-Wächter, Herr Ganser |
| Unterstützung auf fachlicher Ebene | Frau Dresen, Frau Brockmann-Wächter, Herr Ganser |

* + 1. Software und Hardware

In den Räumlichkeiten des Heinrich-Hertz-Berufskollegs stehen PCs mit folgenden technischen Eigenschaften für die Entwicklung bereit.

|  |  |
| --- | --- |
| Hardware | Bezeichnung |
| Prozessor |  |
| Arbeitsspeicher |  |
| Betriebsysteme  Textverarbeitung, Diagrammerstellung |  |
| Entwicklungsumgebung |  |
| Apache Server |  |
| SQL Server |  |
| GitHub Repository |  |

* + 1. Sprachen, Spezifikation und Standards

|  |  |  |
| --- | --- | --- |
| Bezeichnung | Version | Erläuterung |
| Java Development Kit (JDK) | 6 | Software Development Kit für Java |
| Scene Builder | 2.0 | Graphical Interface for FXML-Erstellung |
| MySQL | 5.7 | Relationales Datenbankverwaltunssystem |

Folgende Richtlinien müssen bei der Entwicklung berücksichtigt warden

|  |  |
| --- | --- |
| Kategorie | Richtllinie |
| Lesbarkeit | Einhaltung der Java Konventionen |
| Code und Kommentare in englischer Sprache |
| Inline Kommentare |
| Dokumentation mittels Javadoc | Beschreibung von Konstruktoren / Methoden (inkl. Parameter und Rückgabewert) |
| Beschreibung von wichtigen Attributen |
| Strukturierung | Organisation des Codes in Paketen |
| Strikte Schichtentrennung |
| Verwendung von Interfaces |
| Vermeidung unnötiger Importe |

* + 1. Applikationsumfeld

Nach Angabe des Kunden, wird die Applikation auf einem lokalen Rechner installiert. Die Datenbank läuft ebenfalls unter einem lokalen Server.

* + 1. Benötigte Software für Anwender

Der Anwender benötigt den Zugriff auf einen Datenbankserver auf dem die Datenbank der Benutzer-, sowie der Template-Verwaltung liegt.

## Schnittstellen

* + 1. Personelle Schnittstellen

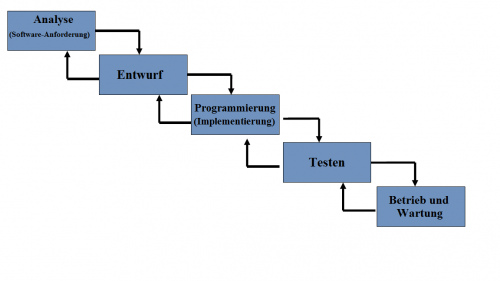
Über den Verlauf des Projektes stehen den einzelnen Teams die Lehrkörper Frau Dresen, Frau Brockman-Wächter und Herr Ganser für technische als auch projektorganisatorische Fragen zur Verfügung.

* + 1. Applikationsschnittstellen

Der volle Funktionsumfang der Applikation ist von Fremddaten der Datenbank des „Didaktischen Wizards“ abhängig. Darüber hinaus ist die Verbindung zur eigenen Datenbank mandatorisch. Sollte der Zugriff auf diese Datenbank nicht möglich sein, kann die Applikation nicht operieren. [Framework, JDBC]

# Projektplanung

## 2.1 Vorgehensmodell

Das festgelegte Vorgehensmodell für dieses Projekt ist das erweiterte Wasserfallmodell. Dieses erlaubt gegenüber seiner klassischen Form Rücksprünge in die jeweilig vorausgehende Phase. Sie besteht aus den folgenden Phasen:

|  |
| --- |
| Analyse |
| * Machbarkeitstudie * Erste Zeitschätzung * Kundenmanagement * Erstellung des Pflichtenheftes * Aufstellung eines Projektplans |

|  |
| --- |
| Entwurf |
| * Konzeption   + Entwurf der Klassendiagramme und des Datenbankschemas   + Technische Planung (Entscheid über die Wahl der Methodik/ Fremdsoftware) |

|  |
| --- |
| Implementierung |
| * Implementierung der Software * Ggf. adjustieren des Projektplans und der Ressourcen * Uni-Tests implementieren |

|  |
| --- |
| Testen |
| * Testfälle des Pflichtenheftes durchführen * Unit-Test ausführen |

|  |
| --- |
| Betrieb und Wartung |
| * Installationsinstruktionen ausführen * Zuhilfenahme der technischen Dokumentation für Wartungen |

Dabei sind Dokumente wie das Erstellen der Projektdokumentation und des Benutzerhandbuchs keiner spezifischen Phase zugeordnet. Sie werden projektbegleitend erstellt, wobei Handbücher noch vor der „Betrieb und Wartung“-Phase fertig gestellt werden müssen. Sie sind für die Installation und die Inbetriebnahme unablässig.

## 2.2 Zeit- und Ablaufplanung

Für die Projektdurchführung sind 25,5 Stunden die Woche angesetzt. Das machen folglich 5,1 Stunden pro Werktag. Für unser vier-köpfiges Team rechnete es sich damit auf 102 Stunden insgesamt.

[Wie lang war die initiierungspahse?]

|  |  |  |
| --- | --- | --- |
| Phase | Geplante Zeit | Prüfungsrelevante Zeit |
| Analyse |  |  |
| Entwurf |  |  |
| Raelisierung |  |  |
| Testen |  |  |
| Betrieb und Wartung |  |  |

## 2.3 Kostenplanung und Nutzwertdarstellung

[k.A.???]

## 2.4 Qualitätssicherung

|  |  |
| --- | --- |
| Qualitätsaspekt | Maßnahme |
| Funktionalität | Starker Einbezug dieses Dokumentes in der Projektdurchführung soll die Vollständigkeit und die Richtigkeit der zu implementierenden Funktionen gewährleisten. Vordefinierte Testfälle werden die Ordnungsmäßigkeit dieser Funktionen sichern. Darüber hinaus werden Unit-Tests bereitgestellt diese Funktionen programmatisch prüft. |
| Zuverlässigkeit | Die Anwendung soll tolerant gegenüber Eingabefehlern des Nutzers sein. Dies wird insbesondere durch Prüfungen in der GUI, aber auch durch ausgeprägtes Exception-Handling realisiert werden.  Kritische Operationen wie das endgültige Löschen erstellter Template müssen durch eine Sicherheitsabfrage bestätigt werden. Somit werden ungewollte Nutzeraktion vermieden |
| Benutzbarkeit | Das Programm ist auf seine Kernfunktion begrenzt. Der schlichte Aufbau der Nutzerschnittstellen soll die Bedienung ohne Zuhilfenahme des Handbuches ermöglichen. |
| Effizienz | Aufgrund der Größe dieses Programms und seiner Daten, sind Verbrauchs- und Zeitverhalten vernachlässigbar. |
| Änderbarkeit | Durch die Wahl eines 3 Schichten Modells mit strenger Schichtentrennung soll die Austauschbarkeit von einzelnen Modulen vereinfacht werden. Lose Kopplungen und hohe Kohäsion der Klassen vereinfachen zudem Prüfungen. |
| Übertragbarkeit | Als Java-Anwendung ist das Programm auf jeder Java Virtual Machine lauffähig. Beigefügte Datenbankskripte können auf neuen Umgebungen die benötigen Datenbanken erstellen. |

# Projektdurchführung

## 3.1 Methodik

Das gesamte Projekt wurde gemäß dem erweiterten Wasserfallmodell durchgeführt (vgl.[2.1 „Vorgehensmodell](#_2.1_Vorgehensmodell)“). Als vorgeschriebenes Vorgehensmodel kam eine Auswahl nicht in Betracht.

## 3.2 Initiierung

Das Projekt wurde von Lehrern des Heinrich-Hertz-Berufskolleg im Rahmen des Oberstufenprojektes initiiert. Damit wurden Projektvorarbeiten den einzelnen Teams gänzlich abgenommen

## 3.3 Analyse

Für die Analysephase wurde uns initial der Arbeitsauftrag mitsamt Gruppeneinteilung und Lastenheft übergeben. Zu Beginn fanden wir uns in unserer Gruppe zusammen und wählten einen Teamleiter und eine Stellvertretung. Die Wahl fiel auf Kenji Kokubo und Monika Schepan als Stellvertreterin.

Unter Zuhilfenahme des Lastenheftes wurde eine erste Zeiteinschätzung vorgenommen. Daraufhin war aufgetragen, dass – der Norm entsprechend – ein Pflichtenheft zum Lastenheft erstellt wird. In ihr wurde zusätzlich eine Machbarkeitsstudie zur Umsetzung als Web Applikation geführt. (vgl. 6.6 Lastenheft)

Zum Schluss dieser Phase wurde eine Projektplanung angefertigt. In ihr wurden absehbare Arbeitspakete den einzelnen Zeiten und Ressourcen zugewiesen. Man beachte, dass dieser Plan auf Zeit und nicht auf Aufwand basiert. Erst nach Abgabe des Pflichtenheftes und der Zeit- und Ressourcenplanung war der Übergang in die Entwurfsphase (vgl. [2.1 Vorgehensmodell](#_2.1_Vorgehensmodell_1)) möglich

## 3.4 Entwurfsphase

## 3.5 Implementierungsphase

## 3.6 Testen

## 3.7 Phasenübergreifende Tätigkeiten

## 3.7.1 Kundenmanagement

Nach dem gewählten Wasserfallmodell sind Rücksprachen nur noch bis spätestens in der Entwurfsphase möglich. Diesem Schema entsprechend, orientierte sich unser Handeln. Absehbare Fragen oder Unklarheiten wurden in der Analysephase abgesprochen. Dennoch kam es vor, dass noch Fragen in der Entwurfsphase aufkamen, die nicht zu einem früheren Zeitpunkt nicht gänzlich beachtet werden konnten.

## 3.7.2 Qualitätsmanagement

Das Qualitätsmanagement wurde gemäß den Qualitätssicherungsangaben in Pflichtenheft bzw. in Kapitel [2.4 „Qualitätssicherung](#_2.4_Qualitätssicherung)“ geführt.

## 3.7.3 Dokumentation

Die Projektdokumentation ist das einzige Dokument, welches über alle Phasen des Vorgehensmodells hinweg entwickelt wird.

# Projektergebnisse

## 4.1 Phasenbegleitende Dokumentation

Im Folgenden werden die angefertigten Dokumente mit den Phasen in Verbindung gesetzt in denen sie ausgefertigt wurden.

|  |  |
| --- | --- |
| Phase | Dokumente |
| Analyse | Pflichtenheft(Machbarkeitsstudie, Testfälle), Projektplan |
| Entwurf | Klassen- und Datenbanktabellendiagramm |
| Implementierung | - |
| Testen | Testprotokoll |
| Betrieb und Wartung | - |

## 4.2 Soll-Ist-Vergleich

## 4.2.1 Projektziele

## 4.2.2 Zeitplanung

## 4.2.3 Kostenplanung

## 4.3 Nutzwertbetrachtung

## 4.3.1 Qualitative Betrachtung

## 4.3.2 Quantitative Betrachtung

# Projektabschluss

## 5.1 Abnahme und Übergabe

## 5.2 Persönliche Reflexion

## Kenji Kokubo (Projektleiter)

Dieses Projekt war durchaus interessant für mich, da ich mich gegenüber des damaligen Unterstufenprojektes nun in der Rolle des Projektleiter wiederfand. Ich durfte erleben, wie ….

## Monika Schepan (Stellvertretende Projektleiterin)

## Felix Ehrenpfort

## Christopher Schmitz

## 5.3 Ausblick

# Anhang

## 6.1 Abkürzungsverzeichnis

## 6.2 Glossar

## 6.3 Projektplan

## 6.4 Zeitlicher Soll-Ist-Vergleich

## 6.5 Wirtschaftlichkeit

## 6.6 Lastenheft

## 6.7 Pflichtenheft (Auszug)

# Mock-up

|  |  |
| --- | --- |
| Picture 5  Legende für die Abbildungen | |
| C:\Users\user\Pictures\Berseria\1e5007f9-593f-4c01-93e0-a9a595fb7758.jpg | Loginfenster (für alle Benutzer gleich)  Anmeldung durch Eingabe des Benutzernamens und des Passwortes [(**PF2020**)](#bookmark). |
| Picture 9 | Hauptfenster für den Gastzugang  Der Gast kann Ausbildungsberuf und Ausbildungsjahr Auswählen. Für Detailansicht wird das Standardtemplate verwendet  ([PF1000](#PF1000), [PF1200](#PF1200)).  Nach der Auswahl kann die PDF-Datei erstellt werden ([PF1500](#PF1500)). |
| Picture 6 | Hauptfenster für die Lehrer  Lehrer können zwischen vorhanden Templates für die Detailansicht auswählen, ein neues Template generieren oder löschen.([**PF2000**](#PF2000)) |
| Z:\LF06\Oberstufenprojekt\Gruppe_3\Template_Generierung.jpg | Template-Generierung  Der Benutzer kann aus der Detailliste auswählen, aus welchen Details ein neues Template erstellt werden soll. |
| Z:\LF06\Oberstufenprojekt\Gruppe_3\Admin_Hauptfenster.jpg | Hauptfenster für den Admin  Administratoren können zusätzlich ein Fenster zur Benutzerverwaltung aufrufen |
| Picture 7 | Fenster für die Benutzerverwaltung  Administratoren können Benutzerkonten anlegen oder einen Benutzer aus der Liste der vorhandenen Benutzer auswählen und löschen. |
| Picture 8 | Template/Benutzer löschen  Ein Template bzw. ein Benutzer kann gelöscht werden. Das Löschen muss bestätigt werden. |

[…]

# Qualitätszielbestimmung

## Funktionalität

PQ0600 Starker Einbezug dieses Dokumentes in der Projektdurchführung soll die Vollständigkeit und die Richtigkeit der zu implementierenden Funktionen gewährleisten. Vordefinierte Testfälle werden die Ordnungsmäßigkeit dieser Funktionen sichern. Darüber hinaus werden Unit-Tests bereitgestellt diese Funktionen programmatisch prüft.

## Zuverlässigkeit

PQ0610 Die Anwendung soll tolerant gegenüber Eingabefehlern des Nutzers sein. Dies wird insbesondere durch Prüfungen in der GUI, aber auch durch ausgeprägtes Exception-Handling realisiert werden.

PQ0620 Kritische Operationen wie das endgültige Löschen erstellter Template müssen durch eine Sicherheitsabfrage bestätigt werden. Somit werden ungewollte Nutzeraktion vermieden.

## Benutzbarkeit

PQ0630 Das Programm ist auf seine Kernfunktion begrenzt. Der schlichte Aufbau der Nutzerschnittstellen soll die Bedienung ohne Zuhilfenahme des Handbuches ermöglichen.

## Effizienz

PQ0640 Aufgrund der Größe dieses Programms und seiner Daten, sind Verbrauchs- und Zeitverhalten vernachlässigbar.

## Änderbarkeit

PQ0650 Durch die Wahl eines 3 Schichten Modells mit strenger Schichtentrennung soll die Austauschbarkeit von einzelnen Modulen vereinfacht werden. Lose Kopplungen und hohe Kohäsion der Klassen vereinfachen zudem Prüfungen.

## Übertragbarkeit

PQ0660 Als Java-Anwendung ist das Programm auf jeder Java Virtual Machine lauffähig. Beigefügte Datenbankskripte können auf neuen Umgebungen die benötigen Datenbanken erstellen.

# Globale Testfälle

Die hier genannten Testfälle beziehen sich auf die Produktfunktionen (Prefix: PF). Entsprechende Zahlen hinter den Präfixen markieren die Zugehörigkeit eines Testfalls zu seiner Funktion. Ein Beispiel hierfür ist die PF1200 <-> PT1200-Zugehörigkeit.

**PT1000** Abgleich der aufgelisteten Ausbildungsberufe und –jahre mit den Einträgen in der Datenbank.

**PT1100** Prüfen, dass keine nicht-freigegebene Lernsituation der Datenbank entnommen wird.

**PT1200** Standardtemplates als auch selbst erzeugte Templates auf Pflichtelemente in PF1200 prüfen. Ausnahmslos jedes muss vorhanden sein.

**PT1300** Überprüfung von selbst erzeugten Template-Konfigurationen und Einträgen in der Datenbank bei Speicherung, sowie beim Laden und Editieren.

**PT1310** Überprüft in PF1300

**PT1400** Templates die als Gast generiert werden, müssen mit dem Standardkonfigurationstemplate übereinstimmen. Das Gast-Konto muss exklusiv dieses Template für das PDF generieren verwenden.

**PT1500** Überprüfung, ob gespeicherte PDFs auch im gewünschten Verzeichnis abgelegt wurden.

**PT2000** Abgleich der für den Benutzer sichtbaren und erlaubten Templates mit den angezeigten. Nur der Urheber seines Templates darf es nutzen oder löschen.

**PT2010** Überprüfung der Verfügbarkeit eines Standardtemplates in allen Bildungsgängen

**PT2020** Überprüfung von falschen und validen Login Daten bei der Anmeldung.

# Entwicklungs-Umgebung

## Software

* Betriebssystem Windows 7 oder höher
* Java SDK 8 and Java JRE 8
* Lokale XAMPP Installation
* Microsoft Office
* Microsoft Visio
* Astah Community
* Draw.io (<https://www.draw.io/>)
* Projektmanagementtool – nach Absprache
* MS Project 2013
* Git Versionierungssoftware

[…]

## 6.8 Klassendiagramm

## 6.9 Relationales Schema der Datenbank