**1 Einleitung**

**1.1 Projektumfeld**

**Ausbildungsbetrieb:** Die Berliner Synchron GmbH (BSG) ist ein mittelständisches, internationaltätiges Unternehmen, dessen Kerngebiet die deutsche Synchronisation englischsprachiger Filme und Serien ist. 1949 als erstes Synchronunternehmen Deutschlands gegründet, hält die BSG nach wie vor eine Spitzenstellung innerhalb der Synchronbranche. Seit 2016 ist das Unternehmen Teil der S&L

Medien Gruppe GmbH, die es sich zum Ziel gesetzt hat, ein allumfassendes Unterhaltungsmarketing zu bieten. Neben den rund 60 festen Mitarbeitern, greift die BSG auf einen Pool aus über 3 000 Synchronschauspielern, Autoren, Regisseuren, Cuttern und Takern zurück.

Die interne IT-Abteilung des Unternehmens beschäftigt derzeit eine Vollzeitkraft und einen Auszubildenden. Unterstützt wird diese durch die IT-Abteilung der S&L Medien Gruppe sowie durch einen technischen Dienstleister vor Ort. Das Aufgabenfeld der IT-Abteilung umfasst den Aufbau und die Wartung des firmeninternen Netzwerks, die Administration der eigenen Server, die Entwicklung und Pflege der in der Firma eingesetzten Systeme sowie die alltägliche Benutzerunterstützung.

**Auftraggeber:** Der Auftraggeber ist in diesem Projekt der Ausbildungsbetrieb – die BSG.

**Projektbeschreibung:** Derzeit befindet sich eine neue Software – Copper in der Testphase. Diese wird von einem externen Dienstleister in Zusammenarbeit mit der BSG programmiert.

Mit Copper sollen alle Tätigkeiten von der Projekterstellung über die Projektumsetzung bis hin zur Projektauswertung gesteuert und alle betrieblichen Prozesse über ein Webinterface zentralisiert dargestellt werden. Eine zusätzliche Webanwendung soll für die neue Software entwickelt werden. Diese Webanwendung soll ein Eingabeformular bereitstellen und die vom Benutzer eingegebenen Daten in eine MySQL-Datenbank speichern.

**1.2 Projektziel**

**Projekthintergrund:** Die Synchronisation von Filmen und Serien wird jeweils als ein Projekt bezeichnet. Die Akte eines Kinofilms bzw. die Episoden einer Serie bilden dabei einzelne Teilprojekte.

Um diese Teilprojekte in deutscher Sprache aufzunehmen, erhält die Berliner Synchron GmbH von seinen Kunden, meist ausländische Produktionsstudios, sowohl das zu synchronisierende Video- als auch diverse Audiomaterialien. Dabei handelt es sich um unterschiedliche Audiotypen wie Originalton, Musik, Effekte und andere, von denen es wiederum unterschiedliche Versionen gibt. Bevor das gelieferte Material in den Produktionsprozess integriert werden kann, muss dieses analysiert werden. Dabei wird es auf Fehler überprüft und die Qualität des angelieferten Materials wird bewertet. Diese Bewertung kann dazu führen, dass das Audiomaterial überarbeitet oder sogar erneut vom Kunden zugesandt werden muss. Bisher wird das Ergebnis dieser Prüfung formfrei, meist in Word Dateien, geschrieben. Danach werden diese in den entsprechenden Projektordnern gespeichert und die betreffenden Projektbeteiligten per E-Mail informiert.

**Ziel des Projekts:** Ziel dieses Projektes ist die Entwicklung einer einfachen und leicht bedienbaren Webanwendung zur Erfassung, Darstellung und zentralen Speicherung der Daten der Materialeingangsprüfung und deren direkter Zuordnung zu einem Teilprojekt. Alle Mitarbeiter sollen nach einer kurzen Einführung in der Lage sein, ein solches Formular auszufüllen.

**1.3 Projektbegründung**

**Nutzen des Projekts:** Durch ein standardisiertes Formular mit zum Großteil vorgegebenen Werten über Dropdown-Listen soll die Erstellung eines solchen Meldeberichtes erleichtert und damit beschleunigt werden. Dieses Projekt dient zudem einen weiteren wichtigen Aspekt: dem Controlling. Neben der Kostenerfassung für den Materialeingangsbericht durch Copper, lässt die zentralisierte Erfassung der Daten und deren direkter Zuordnung zu den Projekten und Kunden zu, eine übersichtliche und gefilterte Darstellung aller Materialeingangsberichte aus den gewonnen Daten zu generieren. Damit lassen sich Rückschlüsse auf anfallende Kosten für zukünftige Projekte ziehen. Das Erstellen eines solchen Berichts ist allerdings nicht Teil dieses Projekts und wird erst in einem Nachfolgeprojekt entwickelt.

Die Erstellung und Speicherung der Daten eines Materialeingangs stellt lediglich die Grundlage dar.

**Motivation:** Der Auftraggeber ist daran interessiert, ein allumfassendes System für die Projektabwicklung zu haben, welches alle Projektschritte dokumentiert und darstellt. Dazu wurde die Software Copper in Auftrag gegeben, welche durch einen externen Dienstleister entwickelt und nach erfolgreicher Einführung von der eigenen IT-Abteilung erweitert werden soll.

**1.4 Projektschnittstellen**

Copper ist eine Webapplikation, die mit der Programmiersprache Ruby (Version: 2.3.1) und dem Framework Rails (Version 4.2.7.1) erstellt wurde. Da diese Anwendung später auf einen firmeneigenen Server ausgeführt und hauptsächlich im lokalen Netzwerk eingesetzt wird, soll diese vor allem über den in der Berliner Synchron eingesetzten Webbrowser – Google Chrome ausführbar sein. Zur Erstellung der Webseite wird das Framework Bootstrap (Version 3.3) verwendet.

Die browserbasierten Nutzereingaben werden dann mittels http von einem Apache-Webserver entgegen genommen und an die Rails-Applikation übermittelt. Dort werden die Anfragen bearbeitet und beantwortet.

Die Daten werden mittels eines in Rails integrierten MySQL-Connectors (mysql2, Version 0.3.17) in eine lokale relationale MySQL-Datenbank gespeichert und abgerufen.

Projekttitel

Projektuntertitel

*2 Projektplanung*

**1.5 Projektabgrenzung**

**Was dieses Projekt nicht bietet:** Eine automatisierte Auswertung sowie sonstige Zusammenfassungen außerhalb eines Teilprojektes werden nicht vorgenommen. Dieses Projekt stellt lediglich die Grundlage dar.

Zudem wird die Software während des Projektes noch nicht in Copper integriert. (Siehe Abweichung vom Projektantrag).

**2 Projektplanung**

Der Projektplanung ging ein Meeting zwischen der Geschäftsführung, einer Mitarbeiterin, welche zum Großteil Materialeingangsberichte erfasst und der IT-Abteilung voraus. Nachdem die Geschäftsführung sich nach einer Zusammenfassung der Materialeingangsprüfungsberichte für einen bestimmten Kunden erkundigte bzw. nach der Möglichkeit einer solchen, wurde schnell klar, dass dies auf Basis der aktuellen Arbeitsweise nicht gewährleistet werden kann. Für eine übersichtliche, gefilterte Darstellung solcher Berichte würde eine gewisse Standardisierung und zentrale Speicherung der Daten benötigt.

Schnell kam man dann auf die Idee, dies gleich mit der Einführung der neuen Software – Copper und meiner Projektarbeit zu verknüpfen. Nach Absprache mit dem Entwickler von Copper wurde dann der Projektantrag bei der IHK eingereicht.

**2.1 Projektphasen**

Von der Industrie- und Handelskammer (IHK) wurden für die Bearbeitung des Projektes 70 Arbeitsstunden vorgegeben. Daher wurde das Projekt in ein Teilprojekt gewandelt. Nach der positiven Rückmeldung auf mein Projektantrag wurde das Projekt in den Tagesablauf meiner betrieblichen Arbeit integriert und mit ca. 30 Wochenstunden bearbeitet.

**2.2 Zeitplanung**

Tabelle 1 zeigt eine grobe Zeitplanung für die jeweils bevorstehenden Projektphasen. Eine detaillierte Zusammenfassung findet sich im Anhang

|  |  |
| --- | --- |
| **Projektphase** | **Geplante Zeit** |
| Analyse | 6 |
| Entwurf | 14 |
| Implementierung | 37 |
| Abnahme | 1 |
| Einführung | 1 |
| Dokumentation | 11 |

**2.3 Abweichungen vom Projektantrag**

Bei der Erstellung des Projektantrags wurde geplant, das Projekt auf einer bereits vorhandenen Testumgebung aufzusetzen und in Absprache mit dem Entwickler von Copper anzufertigen. Da der Entwickler von Copper während der Projektarbeit weder erreichbar, noch ein Zugriff auf die Testumgebung vorhanden war, musste das Projekt mit den zur Verfügung stehenden Daten zur Zeit der Erstellung des Projektantrags nachgebaut werden. Somit ergaben sich erhebliche Änderungen im Vergleich zum Projektantrag. Ferner konnte aufgrund der am Ende fehlenden Zeit - durch die vorhergehenden, nicht eingeplanten Entwicklungsarbeiten - keine automatisierten Tests des entwickelnden Materialeingangsberichts mehr durchgeführt werden. Dies hätte das zusätzliche Aufsetzen einer Testumgebung bedurft, welche am Ende nicht einmal den tatsächlichen Produktionseinsatz der Anwendung sichergestellt hätten.

Daher wurde in Absprache mit dem Ausbilder beschlossen lediglich ein Testprotokoll zu erstellen. Weiterhin wurde auf die Erstellung einer Entwicklerdokumentation verzichtet.

**2.3 Ressourcenplanung**

Alle benötigten Ressourcen zur Durchführung des Projekts sind bereits im Unternehmen vorhanden bzw. können online heruntergeladen werden. Eine Auflistung befindet sich im Anhang A

**2.4 Entwicklungsprozess**

Die Entwicklung des Projekts erfolgt nach Rücksprache mit der verantwortlichen Mitarbeiterin für Qualitätsprüfung im Unternehmen. Ansonsten wird das Projekt weitestgehend nach dem Wasserfallmodell in Eigenregie erstellt.

**3. Analysephase**

Die erste Analysephase wurde bereits zum Zeitpunkt der Antragstellung des IHK-Projektes durchgeführt. Dort wurde bereits mit dem externen Entwickler gesprochen und auf der Grundlage eines SQL-Scripts ein vorläufiges Datenbankmodell für das Projekt erstellt und eingegrenzt.

**3.1 Ist-Analyse**

In der Berliner Synchron GmbH wird sowohl das eingehende als auch das ausgehende Audio- und Videomaterial analysiert und auf Fehler überprüft. Bei der Analyse werden neben möglichen Fehlern auch Meta-Daten wie das Materialeingangsdatum, Audioformat oder die Version gesammelt. Zudem wird entschieden, ob das Material nochmal überarbeitet oder gar vom Kunden neu übermittelt werden muss, bevor es in die Produktion geht. Eine solche Prüfung wird bisher formfrei, meist in Word-Dateien geschrieben. Danach werden die für das Projekt verantwortlichen Mitarbeiter per Email über die Fertigstellung und auf mögliche Mängel hingewiesen. Bei einem solchen Bericht gibt es viele Informationen, welche nur einen begrenzt möglichen Datensatz aufweisen. Diese würden über ein Dropdown-Menü dargestellt und per einfachen Mausklick ausgewählt werden können. Dies würde den individuellen Schreibaufwand der entsprechenden Mitarbeiter erheblich senken, was wiederum zu einer Zeitersparnis sowie zu einer Fehlerreduzierung und damit auch zu einer Kostensenkung führt.

Weiterhin werden dann die Berichte in den entsprechenden Netzwerkordnern der einzelnen Teilprojekte gespeichert. Das Erhalten eines Überblicks über die Materialeingänge und deren Fehler für ein Teilprojekt ist somit äußerst schwierig. Will man einen solchen Überblick über ein komplettes Projekt oder gar für einen Kunden erhalten, ist dies nahezu unmöglich. Dies erschwert die Kostenberechnung und das Ziehen von Rückschlüssen für künftige Projekte.

**3.2 Wirtschaftlichkeitsanalyse**

Die Wirtschaftlichkeit des neuen standardisierten Berichts zur Qualitätssicherung steht außer Frage. Die derzeitige Darstellung und Bearbeitung ist unübersichtlich und ineffizient. Die Vielzahl der Erstellung von Berichten würde sich in jedem Fall mittelfristig gesehen, refinanzieren. Mit der Inbetriebnahme der neuen Software sollen zudem alle übrigen Projektprozesse dargestellt und deren Kosten ermittelt werden. Warum also die Erstellung eines solchen Berichts ausschließen? Das würde wiederum das Controlling erschweren und auch für diese Kostenstelle einen zusätzlichen Zeit- und damit Kostenaufwand bedeuten.

**3.2.1 „Make or Buy“**

Da die Synchronbranche sehr speziell ist und es keine standardisierte Arbeitsweise oder Projektsoftware für diesen Bereich gibt, entschied sich die BSG einen externen Dienstleister mit der Entwicklung einer Software, zugeschnitten auf die Bedürfnisse des Unternehmens, zu engagieren. Von Anfang an war klar, dass dieses Projekt, selbst nach Einführung, von der unternehmenseigenen IT-Abteilung stetig weiterentwickelt werden soll und muss. Da die neue Software noch in den Händen des Entwicklers ist, stellt sich nur die Frage, ob dieser den neuen Bericht erstellt oder der firmeneigene Mitarbeiter/Auszubildende. Aus Kostengesichtspunkten ist der Auszubildende für das Unternehmen am geeignetsten.

**3.2.2 Projektkosten**

Die Kosten zur Durchführung des Projekts ergeben sich aus den Gehältern der beteiligten Mitarbeiter, die für die Mitarbeiter aufzuwendenden Sozialabgaben, sowie die Gemeinkosten sonstiger Ressourcen(Footnote). Sonstige Ressourcen wie Internet und Strom werden dabei vernachlässigt, da diese Kosten ohnehin anfallen. Bei 20 Arbeitstagen monatlich erhält der Auszubildende 884€ brutto. Der Arbeitgeberanteil zur Sozialversicherung beträgt dabei 229,53€. Für den Personalaufwand der übrigen Mitarbeiter wird mit einer Pauschale von 30€ / Stunde gerechnet.

Stundensatz Auszubildenden

8 h/Tag · 20 Tage/Monat = 160 h/Monat (1)

(884,00 € + 229,53 €)/Monat

160 h/Monat = 1113,53 €/Monat

160 h/Monat = 6,96 €/h (2)

Es ergibt sich demnach ein Stundensatz von 6,96 €. Die Durchführungszeit des Projekts beträgt 70 Stunden. Für die Nutzung der Ressourcen wird ein pauschaler Stundensatz von 10€ herangezogen. Nachfolgender Tabelle lässt sich eine detaillierte Kostenübersicht entnehmen.

Vorgang Zeit Kosten pro Stunde Gesamtkosten

Entwicklungskosten 70 6.96 388,08

Fachgespräch 1 30 30

Meeting QM 2 30 60

Benutzerschulung 0,5 4 x 30 60

Ressourcen 70 10 700

Kosten 1238,08

**3.2.3 Amortisationsdauer**

Geht man von einer Zeitersparnis bei der Erstellung von 10 Minuten pro Bericht aus, bei zwei Berichten pro Tag, ergibt sich eine durchschnittliche monatliche Zeitersparnis von 400 Minuten über 6,6 Stunden. Bei der veranschlagten Mitarbeiterpauschale ergibt sich eine monatliche Kostenersparnis von 200€. Damit wäre dieses Projekt nach knapp über 6 Monaten amortisiert.

Die tatsächliche Amortisationsdauer des Projektes ist zum jetzigen Zeitpunkt noch nicht absehbar, da es nur den ersten Teil des Projekts zur Qualitätssicherung darstellt. Dies stellt wiederum auch nur einen Teil des viel größeren Projektes – Copper dar.

**3.3 Nutzwertanalyse**

Neben der Zeitersparnis und Minimierung der Fehleranfälligkeit liegt der größte Nutzen des Projekts in der Standardisierung der Berichte, sowie der späteren Integration in Copper. Auf dieser Grundlage lassen sich die tatsächlich aufgetreten Projektkosten noch genauer kalkulieren und es ist mögliche Rückschlüsse auf Kosten zukünftiger Projekte zu ziehen.

**3.4 Qualitätsanforderungen**

Die Webanwendung soll intuitiv und leicht bedienbar für die im Tonbereich qualifizierten Mitarbeiter sein. Schlüsseldatensätze für spätere Auswertungen sollen auswählbar sein. Zudem muss es für etwaige Fehler Kommentarfelder für genauere Erläuterungen geben. Die Berichte müssen sich eindeutig unterscheiden und einem Teilprojekt zugeordnet sein. Es muss möglich sein, sich die erstellten Berichte anzusehen und diese gegebenenfalls zu editieren. Zudem muss es eine Übersicht über die erstellten Berichte eines Teilprojekts geben. Die komplette Ansicht ist über Google-Chrome skalierbar.

**3.5 Lastenheft**

Nachfolgend werden die wichtigsten Anforderungen für das Projekt dargestellt.

Anforderungen an die Benutzeroberfläche

LB10: Die Berichte müssen online über den Chrome-Webbrowser erreichbar sein.

LB20: Die Ansicht muss skalierbar sein.

LB30: Die Bedienung und Formularfelder müssen intuitiv und selbsterklärend sein.

LB40: Sich wiederholende standardisierte Datensätze müssen auswählbar sein.

LB50: Es muss möglich sein zu diversen Datensätze Kommentare zu schreiben.

LB60: Erstellte Berichte müssen editierbar sein.

LB70: Für die erstellten Berichte muss es eine Anzeige geben.

LB80: Es muss eine Übersicht aller Berichte eines Teilprojekts geben.

Funktionale Anforderungen

LF10: Die Berichte müssen eindeutig voneinander unterschieden werden können.

LF20: Die Navigation zu den Berichten muss vorhanden sein.

Sonstige Anforderungen

LS10: Die Anwendung muss anhand der vorhandenen Information in Copper integrierbar sein.

LS20: Das Projekt muss für das Anschlussprojekt erweiterbar sein.

**3.6 Zwischenstand**

Tabelle zeigt den Zwischenstand nach der Analysenphase

|  |  |  |  |
| --- | --- | --- | --- |
| **Vorgang** | **Geplant** | **Tatsächlich** | **Differenz** |
| Durchführung einer Ist-Analyse | 2 | 3 | +1 |
| Wirtschaftlichkeitsanalyse | 1 | 1 | 0 |
| Erstellung eines Lastenhefts | 3 | 3 | 0 |
| **Gesamt** | **6** | **6** | **+1** |

**4 Entwurfsphase**

Die meisten, jedoch nicht alle, Entscheidungen über verwendete Technologien stehen bereits mit Projektbeginn fest. Denn sie sind von der sich in der Entwicklung befindenden Software – Copper abhängig. In der Entwurfsphase ist es das Ziel, anhand der zur Verfügung stehenden Informationen ein Designplan für die Anwendung zur späteren Integration in Copper zu entwickeln.

**4.1 Zielplattform**

Die Ruby on Rails Webanwendung soll unter dem Apple-Betriebssystem OS X laufen. Die Daten werden eine lokal installierte MySQL-Datenbank gespeichert. Die Anbindung zur Datenbank erfolgt über den mysql2 gem, einer Ruby Bibliothek. Zum lokalen Aufrufen und Testen der Applikation über den Chrome-Browser wird der Thin-Webserver verwendet. Die Realisierung der skalierbaren Webanwendung wird über die Auszeichnungssprachen Haml, CSS und dem CSS-Framework Bootstrap (Version 3.3) sichergestellt.

**4.2 Architekturdesign**

Als Architekturdesign wird das in Rails üblicherweise eingesetzte Model View Controller (MVC) – Pattern genutzt. Dabei repräsentieren die sich im Model befindlichen Klassen die Datenbanktabellen. Die interne Kommunikation zwischen den Modell-Klassen, der Datenhaltungsschicht, und der MySQL-Datenbank wird dann vom mysql2 gem übernommen. Die browserbasierten Darstellungen und Formulare, die GUI-Schicht, werden in der View repräsentiert. Die Kommunikation und Logik zwischen View und dem Model übernimmt der Controller. Die ankommenden HTTP-Anfragen und Parameter werden über entsprechende Routen an den Controller weitergeleitet, welcher dann anhand dieser die sogenannten Create Read Update Delete (CRUD) – Operationen durchführt.

**4.3 Entwurf der Benutzungsoberfläche**

Da die Anwendung primär von ausgebildetem Fachpersonal in der Tonbranche verwendet wird, können auch Fachbegriffe oder Abkürzungen verwendet werden. Die Bewertung des Materials sollte jedoch für jedermann verständlich sein. Die Skalierbarkeit der Anwendung wird durch Bootstrap realisiert. Da die Anwendung von internem Personal genutzt und in ein anderes Projekt eingepflegt werden soll, wird auf ein ausgeklügeltes, repräsentatives Design verzichtet. Für Eingabefelder mit begrenzten Werten werden Dropdown-Menüs erzeugt. Zur eindeutigen Kennzeichnung jeder Eingabe werden Label verwendet. Die Datumseingabe erfolgt mithilfe eines Datumswählers. Layout Siehe Anhang

**4.4 Daten- und Datenbankmodell**

Während der Vorbereitung des Projektantrags ließ mir der externe Entwickler einen kleinen Ausschnitt der Datenbank in Form eines MySQL-Skripts zukommen. Auf dieser Grundlage muss der bestehende Ausschnitt der Datenbank mit den entsprechenden Namenskonventionen erweitert werden. Dabei werden die benötigten Tabellen über Rails migriert. Es werden Tabellen für alle auszufüllenden Felder mit begrenzten Datensätzen erstellt. Eine Tabelle für den Materialeingangsbericht sammelt die eingegebenen Daten und ordnet diese der Tabelle des Teilprojekts zu, indem sie die entsprechenden Datensätze der anderen Tabellen referenziert. Für jede Tabelle muss in Rails eine Klasse angelegt werden. Dort müssen die Eingaben dann validiert werden, bevor sie tatsächlich in die Datenbank geschrieben werden.

**4.5 Fachkonzept**

Für die Umsetzung der Logik muss jeweils ein Controller für ein Projekt und ein angelegt werden, um die grundlegende Struktur von Copper zu simulieren. Dabei beinhalten diese nur die grundlegendsten Methoden, welche es mir gestatten sollen, mich über den Browser zu meinen Views zu navigieren und den zu erstellenden Controller für die Berichte anzusprechen. Auf diesen werden dann die CRUD-Operationen definiert.

**4.6 Maßnahmen zur Qualitätssicherung**

Während der Bearbeitung des Projekts wurde die Funktionalität der Anwendung kontinuierlich getestet. Dabei wurden die Links ausgeführt, die Routen getestet und nach Eingaben die Datenbank auf die Richtigkeit der Werte geprüft. In Zusammenarbeitet mit der QM wird ein für die Abnahme erforderliches Testprotokoll erstellt.

**4.7 Pflichtenheft**

Nachfolgend werden die wichtigsten vereinbarten Pflichten des Projekts aufgelistet, um eine Standardisierung der Eingangsberichte, die Integrierbarkeit in Copper sowie die Erweiterbarkeit für die Erstellung späterer Auswertungen zu gewährleisten.

Pflichten zur Benutzungsoberfläche

PB10: Es muss eine Webanwendung programmiert und ständig mit Chrome getestet werden.

PB20: LS20: Es muss Bootstrap verwendet und die Skalierbarkeit ständig über den Webbrowser getestet werden.

PB30: Jede Eingabe muss ein Label erhalten. Die Namensbezeichnung erfolgt in Absprache mit der zuständigen Mitarbeiterin für Qualitätssicherung

PB40: Es muss Dropdown-Menüs und Checkboxen geben.

PB50: Es muss Textfelder und Textareas geben. Diese müssen optisch zu den Dropdown-Menüs und Checkboxen zugeordnet werden

PB60: Es muss eine initiale tabellarische Anzeige der Berichte nach Auswahl des Teilprojekts geben. Dort müssen der Audiotyp, die Version, die Bewertung, sowie Links zum Anzeigen, Bearbeiten und Löschen erscheinen.

PB70: Es muss ein Webformular mit den o.g. Konventionen geben.

PB80: Es muss einen Link zum Bearbeiten geben, welcher das unter PB70 definierte Formular mit den entsprechenden Datensätzen lädt.

PB90: Es gibt einen Link zur Anzeige, welches das unter PB70 definierte Formular mit deaktivierten Eingabefelder lädt.

Funktionale Pflichten

PF10: Berichte mit gleichem Teilprojekt, Audiotyp und Audioformat müssen eine unterschiedliche Versionsnummer haben.

PF20: Über die Auswahl der Teilprojekte muss die Auswahl eines jeden zugehörigen Berichts erfolgen.

Sonstige Anforderungen

PS10: Alle bekannten Informationen über Copper vom Betriebssystem bis zu den Versionen der Frameworks und Bibliotheken werden verwendet.

PS20: Die Daten werden in eine relationale MySQL-Datenbank gespeichert.

**4.8 Zwischenstand**

|  |  |  |  |
| --- | --- | --- | --- |
| **Vorgang** | **Geplant** | **Tatsächlich** | **Differenz** |
| Entwurf der Projektumgebung | 1h | 1h | 0 |
| Entwurf der Benutzeroberflächen | 3h | 4h | +1 |
| Entwurf der erweiterten Datenbank | 4h | 4h | 0 |
| Erstellung eines Testprotokolls | 2h | 2h | 0 |
| Erstellung des Pflichtenhefts | 4h | 4h | 0 |
| **Gesamt** | **14h** | **15h** | **+1** |

**5 Implementierungsphase**

Bevor mit der eigentlichen Programmierung und Implementierung begonnen werden kann, muss direkt am Anfang eine dazugehörige Entwicklungsumgebung aufgesetzt werden. Dies war zum Zeitpunkt des Projektantrags nicht bekannt.

Während der gesamten Dauer das Projekts wurde regelmäßig über die Versionsverwaltungssoftware Github gesichert.

**5.1 Implementierung der Projektumgebung**

Nach einem Backup des bisherigen Systems auf einem MacBook Pro, werden Ruby, Rails und alle weiteren benötigten Frameworks und Bibliotheken mit den bekannten Informationen eingebunden und deren Funktionstüchtigkeit getestet. Der MySQL-Server, Ruby Version Manager (rvm), über den Ruby installiert wird, sowie das Datenbank-Management System Sequel Pro werden mit Hilfe des OS X Paket Managers Homebrew installiert. Über den Bash

rails new app

wird ein neues Projekt erstellt. Als Entwicklungsumbegung (IDE) wird Rubymine verwendet.

**5.2 Implementierung des Webservers und des Routings**

Als Web Server wird Thin verwendet, welcher leicht als gem über Ruby installiert werden kann und keiner weiteren Konfiguration bedarf. In der Projektdatei config/routes.rb werden die Routen zum Controller mit den entsprechenden Operationen definiert.

**5.3 Implementierung der Datenbank und Datenbankmodelle**

**5.3.1** **Implementierung der Datenbank**

Nach der Erstellung eines Projekts wird über den simplen Befehl rake db:create

die Datenbank erstellt werden und danach das vom externen Entwickler erhaltene MySQL-Skript importiert. Migrationen für die neuen Tabellen werden in Rails geschrieben und über rake db:migrate der Datenbank hinzugefügt. Das hat den Vorteil, dass man jederzeit seine selbst erstellten Migrationen zurücksetzen, diese editieren und der Datenbank wieder hinzufügen kann. Über ein erstelltes MySQL-Skript werden dann die Daten für die Tabellen der Dropdown Menüs importiert.

**5.3.2 Implementierung der Modelle**

Für jede Tabelle der Datenbank wird eine Klasse erstellt. Dies geschieht über den Befehl rails generate model [Tabellenname]. Dabei ist darauf zu achten, dass man sich an den Namenskonventionen von Rails orientiert. Rails kann so die entsprechenden Modelle der jeweiligen Tabelle automatisch zuordnen und die Datenbankabfragen generieren. Danach werden in den Model-Klassen die Relationen der Klassen nach dem Vorbild der Tabellen untereinander definiert und Operationen zur Gültigkeitsprüfung der an die Tabelle zu übergebenen Daten geschrieben.

**5.4 Implementierung des Fachkonzepts**

Über den Befehl rails generate controller [Controllername] wird eine neue Klasse für die entsprechenden Controller erstellt. Entsprechend der Namenskonventionen werden durch Rails zusätzliche Hilfsdateien, sowie einen Pfad zur View des Controller angelegt. Anhand der Methodennamen erfolgt das Routing zu dessen Operationen. Des weiteren wird anhand der Rails Namenskonvention auch die entsprechende Ansicht festgelegt. Hier werden die entsprechenden CRUD – Operationen implementiert.

**5.5 Implementierung der Benutzeroberfläche**

Für die Ansicht der Projekte und Teilprojekte werden nur die nötigsten Dateien und Links für die Weiterleitung zum Materialeingangsbericht implementiert. Die Ansichten für den Bericht werden mit der HTML – Abstraktionssprache HAML geschrieben. Über die Index-Datei des Berichts werden alle zu dem Teilprojekt aufgelisteten gespeicherten Berichte angezeigt und zu den entsprechenden Ansichten (Anzeigen, Bearbeiten, Löschen) verlinkt. Zudem wird ein Link zur Erstellung eines neuen Berichts generiert. Für das Anzeigen, Bearbeiten und Erstellen wird jeweils das gleiche „Partial“ – Formular mit einem Parameter für das Aktivieren bzw. Deaktivieren geladen. Die Datenauswahl in den Dropdown-Menüs werden über Rails geladen.

**5.6 Zwischenstand**

Die nachfolgende Tabelle zeigt den Zwischenstand nach der Implementierungsphase

|  |  |  |  |
| --- | --- | --- | --- |
| **Implementierungsphase** | **Geplant** | **Tatsächlich** | **Differenz** |
| Implementierung der Projektumgebung | 6h | 7h | +1 |
| Implementierung der Datenbank und Migrationen | 8h | 6h | -2 |
| Implementierung der Models | 6h | 6h | 0 |
| Implementierung der Logik für die Darstellung und Datenhaltung | 8h | 7h | -1 |
| Implementierung der Views | 9h | 10h | +1 |
| **Gesamt** | **37h** | **38h** | **-1** |

**6. Qualitätssicherung und Abnahme**

**6.1 Qualitätssicherung und Abnahme**

Die Abnahme erfolgt durch die zuständige Mitarbeiterin für Qualitätssicherung (QM) sowie den Ausbilder. Dabei werden erneut die Daten aus dem Testprotokoll eingegeben. Die Aus- und Eingabe sowie das äußere Design wird durch die QM-Verantwortliche geprüft, wohingegen die technischen Spezifikationen für Copper durch den Ausbilder geprüft werden.

**6.2 Zwischenstand**

Tabelle zeigt den Zwischenstand nach der Abnahmephase.

|  |  |  |  |
| --- | --- | --- | --- |
| **Vorgang** | **Geplant** | **Tatsächlich** | **Differenz** |
| Qualtiätssicherung und Abnahme | 1 h | 1 h |  |
| **Gesamt** | **1 h** | **1 h** | **0** |

**7. Einführungsphase**

Die tatsächliche Inbetriebnahme der Anwendung kann derzeit noch nicht erfolgen, da dieses Projekt abhängig von der Integration in Copper ist.

**7.1 Einführung bei den Mitarbeitern**

Den Sounddesignern wird das Projekt demonstriert und die Bedienung erklärt.

|  |  |  |  |
| --- | --- | --- | --- |
| **Vorgang** | **Geplant** | **Tatsächlich** | **Differenz** |
| Einführung bei den Mitarbeitern | 1 h | 1 h |  |
| **Gesamt** | **1 h** | **1 h** | **0** |

**8 Dokumentation**

Diese Projektdokumentation wurde im Rahmen der IHK - Abschlussprüfung zum IT-Fachinformatiker für Anwendungsentwicklung mit LATEX erstellt.

**8.1 Benutzerdokumentation**

Eine vorläufige Benutzerdokumentation mit entsprechenden Hinweisen für die Bedienung wurde mit Microsoft-Word geschrieben und in einen für Handbücher vorgesehenen Netzwerkordner im PDF-Format gespeichert.

**8.3 Zwischenstand**

|  |  |  |  |
| --- | --- | --- | --- |
| **Erstellung der Dokumentation** |  | **Tatsächlich** | **Differenz** |
| Erstellung der Projektdokumentation | 10h | 12h | +2 |
| Erstellung der Benutzerdokumentation | 1h | 1h | 0 |
| **Gesamt** | **11h** | 13h | +2 |

**9 Fazit**

**9.1 Soll-/Ist-Vergleich**

Auch wenn ich mir den Projektablauf und das Ergebnis zum Zeitpunkt der Antragsstellung gänzlich anders vorgestellt habe, sind die Kollegen mit dem neuen Formular sehr zufrieden und würden es gerne schon in Betrieb nehmen.

Die Zeitplanung weicht, insgesamt betrachtet, geringfügig ab. Betrachtet man jedoch die einzelnen Teile, fällt auf, dass es nicht in Gänze möglich war vorher zu sehen wie groß der tatsächliche Aufwand für die einzelnen Teile ist. Tabelle **100000** zeigt einen Vergleich der veranschlagten sowie tatsächlich aufgewendeten Zeit.

|  |  |  |  |
| --- | --- | --- | --- |
| Projektphase | **Geplant** | **Tatsächlich** | **Differenz** |
| **Analysephase** | **6 h** | **3 h** | **+1 h** |
| **Entwurfsphase** | **14 h** | **15 h** | **+1 h** |
| **Implementierungsphase** | **37 h** | **36 h** | **-1 h** |
| **Qualtiätssicherung und Abnahme** | **1 h** | **1 h** | **0** **h** |
| **Einführungsphase** | **1 h** | **1 h** | **0 h** |
| **Erstellung der Dokumentation** | **11 h** | **11 h** | **+2 h** |
| **Gesamt** | **70 h** | **73 h** | **+3 h** |

**9.2 Gewonnene Erkenntnisse**

Das Projekt in kompletter Eigenregie zu erstellen, kostete zwar einerseits viel Kraft, aber hat mich persönlich weitergebracht. Jeder Schritt in der Erstellung des Projekts machte aus mir einen besseren Entwickler, auch wenn es mir vor Augen führte wieviel Sachen es noch über in Ruby, Rails und die komplette Webprogrammierung es noch zu lernen und zu verinnerlichen gilt, da dies wirklich, ausgenommen der Projektdokumentation, eine Sache ist, die mir Freude bereitet und für die ich gerne morgens zur Arbeit gehe.

**9.3 Ausblick**

Wenn man für die Anwendung noch ein paar Sicherheitsrichtlinien hinzufügt, die Erstellung von Projekten und Teilprojekten auf ähnlicher Weise wie den Audiobericht generiert, noch ein wenig mehr Zeit dem Testen widmet, könnte diese Anwendung tatsächlich in den Betrieb gehen. Da dies jedoch einen zusätzlichen Mehraufwand für die Firma bedeutet, muss dies noch von der Geschäftsführung entschieden werden. Dabei spielt vor allem der tatsächliche Zeitpunkt der Implementierung von Copper eine Rolle.

|  |  |  |  |
| --- | --- | --- | --- |
| **Vorgang** | **Geplant** | **Tatsächlich** | **Differenz** |
| **Analysephase** | **6 h** | **3 h** | **+1 h** |
| Durchführung einer Ist-Analyse | 2 h | 3 h | +1 h |
| Wirtschaftlichkeitsanalyse und Amortisationsrechnung | 1 h | 1 h |  |
| Erstellung eines Lastenhefts | 3 h | 3 h |  |
| **Entwurfsphase** | **14 h** | **15 h** | **+1 h** |
| Entwurf der Projektumgebung | 1 h | 1 h |  |
| Entwurf der Benutzeroberflächen | 3 h | 4 h | +1 h |
| Entwurf der erweiterten Datenbank | 4 h | 4 h |  |
| Erstellung eines Testprotokolls | 2 h | 2 h |  |
| Erstellung des Pflichtenhefts | 4 h | 4 h |  |
| **Implementierungsphase** | **37 h** | **36 h** | **-1 h** |
| Implementierung der Projektumgebung | 6 h | 7 h | +1 h |
| Implementierung der Datenbank und Migrationen | 8 h | 6 h | -2 h |
| Implementierung der Models | 6 h | 6 h |  |
| Implementierung der Logik für die Darstellung und Datenhaltung | 8 h | 7 h | -1 h |
| Implementierung der Views | 9 h | 10 h | +1 h |
| **Qualtiätssicherung und Abnahme** | **1 h** | **1 h** | **0** **h** |
| Qualtiätssicherung und Abnahme | 1 h | 1 h |  |
| **Einführungsphase** | **1 h** | **1 h** | **0 h** |
| Einführungsphase | 1 h | 1 h |  |
| **Erstellung der Dokumentation** | **11 h** | **11 h** | **+2 h** |
| Erstellung der Projektdokumentation | 10 h | 12 h | +2 h |
| Erstellung der Benutzerdokumentation | 1 h | 1 h |  |
| **Gesamt** | **70 h** | **73 h** | **+3 h** |

**Benutzerdokumentation**

**[Screenshot index]**

**Navierung**

* Über die zum dem Bericht zugehörigen Links können bestehende Berichte angesehen, bearbeitet oder gelöscht werden.
* Über den Link „Neuer Bericht“ wird ein neuer Bericht erstellt.
* Über den Link zurück, gelangt man wieder auf die Seite Hauptseite der Serie oder des Aktes.

**[Screenshot index]**

**Beschreibung der Formularfelder**

* Durch einfaches Anklicken der Formularfelder wählt ihr den Audiotyp, das Audioformat, die Version usw.
* In den Kommentarfeldern neben den Buttons könnt ihr zusätzliche Informationen zu Fehlerbeschreibung wie die Stelle im Timecode eintragen.
* Das Datum für den Erhalt des Materials kann sowohl durch hinein klicken gewählt, als auch durch manuelles eintippen, eingetragen werden.
* In den Feldern für Vollständigkeit und Synchronität wählt ihr durch einfaches Klicken aus, ob das Audiomaterial vollständig bzw. synchron zum Video ist. Standardmäßig ist der Haken nicht gesetzt. D.h. wenn ihr den Bericht speichert, wird das Audiomaterial automatisch als asynchron bzw. unvollständig gewertet.

**Zu Beachten:**

* Es ist nicht möglich den gleichen Bericht für eine Serie / einen Akt zweimal abzulegen. Sollte dies dennoch nötig sein, müsst ihr eine neue Version wählen. Einen entsprechenden Hinweis dazu, könnt ihr im Eingabefeld für den externen Namen der Version hinterlegen.
* Das Datum für den Materialeingang darf nicht in der Zukunft oder gelöscht werden. Wählt ihr kein Datum aus, wird automatisch das Datum der Erstellung gespeichert.

**Sollte in einem Dropdown-Menü ein benötigtes Auswahlfeld fehlen, müsst ihr euch bei der IT-Abteilung melden. Dort kann der gewünschte Datensatz in das Formular eingepflegt werden.**