



Abschlussprüfung Winter 2024/2025

Fachinformatiker Anwendungsentwicklung
Dokumentation zur betrieblichen Projektarbeit

Entwicklung eines Webdiensts

zur Suche und Verwaltung von Praktikumsstellen

Prüfling:

Duc Binh Pham

Adresse: Kammerstraße 54, 47057 Duisburg

E-Mail: binhpham199x@gmail.com, Telefon: +49 1575 8181482

Ausbildungsbetrieb:

Cuno-Berufskolleg I

Viktoriastraße 2

58095 Hagen

Schlagwortverzeichnis	4
Abkürzungsverzeichnis	5
1. Einleitung	6
1.1 Projektbeschreibung	6
1.2 Projektziel	6
1.3 Projektumfeld	6
1.4 Projektabgrenzung	6
1.5 Aufgabenstellung	7
2. Projektplanung	7
2.1 Analysephase	7
2.1.1 Ist-Analyse	7
2.1.2 Soll-Konzept	8
2.1.3 Wirtschaftlichkeitsanalyse	8
2.1.3.1 Kosten	8
2.1.3.2 Nutzen	8
2.1.3.3 Kosten Nutzen Vergleich	8
2.2 Definitionsphase	8
2.2.1 Pflichtenheft (Beschreibung von Umsetzung der Anwendung)	8
2.2.2 Wahl des Entwicklungsmodells	8
2.2.3 Zeitplan	9
2.2.4 Meilensteinplan	10
2.2.5 Ressourcenplan	10
2.2.6 Meilensteintrendanalyse	11
2.2.7 Qualitätssicherungsplan	12
2.2.8 Testfallkatalog	12
3. Projektdurchführung	12
3.1 Technologien	12
3.2 Entwurfsphase	13
3.2.1 Anwendungsfälle	13
3.2.2 Aktivitätsdiagramm	13
3.2.3 Klassendiagramm	13
3.2.4 Datenbankmodell - ERD	13
3.3 Implementierungsphase:	13
3.4 Testen der Anwendung	13
3.5 Containerisierung der Anwendung	13

Prozessorientierter Projektbericht

4.	Projektabschluss.....	13
4.1	Projektergebnis	13
4.2	Benutzerdokumentation (API-Dokumentation)	13
5.	Projektreflection	13
5.1	Soll-Ist-Vergleich	14
5.2	Projekterfolg	14
	Eidesstattliche Erklärung	15
	Anhang	16
	A.1 Detaillierter Zeitplan	16
	A.2 Anwendungsfalldiagramm.....	17
	A.3 Aktivitätsdiagramm.....	17
	A.4 Klassendiagramm	17
	A.5 Entity-Relationship Model	17
	A.6 Testfallkatalog.....	17
	A.7 Kommentierter Quellcode	17
	A.8 Test.....	17
	A.9 Benutzerdokumentation (API-Dokumentation).....	17
	A.10 Freigabeerklärung.....	17
	A.11 Abnahmeprotokoll	17

Schlagwortverzeichnis

TypeScript: Eine von Microsoft entwickelte Programmiersprache, die die statische Typisierung von JavaScript ermöglicht und verbesserte Typsicherheit, besseres Code-Refactoring und erhöhte Produktivität bietet.

Env: Umgebungsvariablen dienen dazu, sensible Informationen wie API-Schlüssel oder Zugangsdaten außerhalb des Codes zu speichern und sicher abzurufen. Sie werden verwendet, um die Sicherheit von Anwendungen zu gewährleisten und die Portabilität von Anwendungen zwischen Umgebungen zu erleichtern.

// weitere Schlagwörter

Abkürzungsverzeichnis

// Abkürzungen

1. Einleitung

Die folgende Projektbericht beschreibt den Ablauf eines internen Projekts, welches als Abschlussprojekt für die Ausbildung zum Fachinformatiker für Anwendungsentwicklung absolviert wurde. Der Ausbildungsbetrieb ist Cuno-Berufskolleg I, Viktoriastraße 2, 58095 Hagen. Es ist wichtig zu beachten, dass die Entwicklung des Frontends separat geschieht.

1.1 Projektbeschreibung

Im Rahmen dieses Projekts wird ein Webdienst entwickelt, der den Prozess der Suche und Verwaltung von Praktikumsstellen effizienter gestaltet. Durch die zentrale Bereitstellung von Praktikumsanzeigen sowie die Möglichkeit für Lernende und Betriebe, diese direkt zu verwalten und zu bearbeiten, soll der Aufwand bei der Vermittlung von Praktika reduziert und die Übersichtlichkeit erhöht werden. Die Lösung unterstützt eine einfache und benutzerfreundliche Praktikumsverwaltung und erleichtert die Organisation und den Austausch von Informationen zwischen Betrieben und Lernenden erheblich.

1.2 Projektziel

Das Ziel dieses Projekts ist die Entwicklung eines digitalen Webdienstes, der den Prozess der Suche und Verwaltung von Praktikumsstellen zentralisiert und vereinfacht. Die Plattform soll sowohl Lernenden als auch Betrieben eine Möglichkeit bieten, miteinander in Kontakt zu treten. Lernende können Praktikumsanzeigen durchsuchen, speichern und sich direkt darauf bewerben, während Betriebe ihre Stellenangebote eigenständig erstellen, verwalten und aktualisieren können. Dadurch wird der bisher unübersichtliche und ineffiziente Prozess digitalisiert und optimiert.

Darüber hinaus wird die Plattform durch eine Administratorrolle ergänzt, die es ermöglicht, die Praktikumsanzeigen zu moderieren, zu aktualisieren und bei Bedarf zu löschen. Durch diese zentrale Lösung wird die Organisation der Praktikumsvermittlung deutlich verbessert, wodurch sowohl Lernende als auch Betriebe von einer schnelleren und effizienteren Abwicklung profitieren.

1.3 Projektumfeld

Das Projekt wird im Rahmen der Ausbildung zum Fachinformatiker für Anwendungsentwicklung am Berufskolleg durchgeführt. Der Betrieb, in dem dieses Projekt implementiert wird, ist die IT-Abteilung des Berufskollegs, die für die Einführung und Betreuung der IT-Lösungen zuständig ist. Der Auftraggeber ist die Schulleitung des Berufskollegs, die den Bedarf an einer digitalen Praktikumsbörse erkannt hat, um den Prozess der Praktikumsvermittlung zwischen Betrieben und Lernenden zu vereinfachen.

Der Betrieb selbst verfolgt das Ziel, die Digitalisierung im Bereich der Praktikumsverwaltung voranzutreiben und eine effiziente Plattform bereitzustellen, die den Beteiligten den Zugriff auf relevante Praktikumsinformationen ermöglicht. Das Projekt soll innerhalb des Berufskollegs dienen, kann aber potenziell in anderen Schulen oder Bildungseinrichtungen ebenfalls Anwendung finden.

1.4 Projektabgrenzung

Im Rahmen dieses Projekts wird ein eigenständiger Webdienst entwickelt, der ausschließlich zur Suche und Verwaltung von Praktikumsstellen dient. Die Kommunikation erfolgt ausschließlich mit einer PostgreSQL-Datenbank, welche als einzige Datenquelle

verwendet wird. Externe Schnittstellen oder Verbindungen zu anderen Systemen sind nicht vorgesehen, da der Fokus auf einer internen Implementierung liegt.

Der Webdienst ist nicht dafür ausgelegt, eine eigene Benutzerschnittstelle bereitzustellen. Die Entwicklung von Anwendungen, die diesen Dienst nutzen, sowie die Integration des Dienstes in bestehende Applikationen gehören nicht zum Umfang dieses Projekts.

1.5 Aufgabenstellung

Meine Aufgabe besteht darin, ein Headless-System für die Suche und Verwaltung von Praktikumsstellen zu entwickeln. Im Fokus steht die Bereitstellung einer Programmierschnittstelle (API), die es ermöglicht, Praktikumsinformationen anzulegen, in der Datenbank zu speichern, zu aktualisieren und bei Bedarf zu löschen. Darüber hinaus soll das System eine Funktion zur Suche von Praktikumsstellen bieten, die von Lernenden markiert werden können. Praktikumsinformationen sollen außerdem jederzeit über die API abrufbar sein, um eine flexible und effiziente Nutzung der Daten zu gewährleisten.

Die Aufgaben umfassen im Wesentlichen folgende Bereiche:

- **Erstellung und Umsetzung eines Datenbankkonzepts:** Alle relevanten Benutzer- und Praktikumsdaten werden in einer relationalen Datenbank gespeichert und verwaltet.
- **Entwicklung von API-Endpunkten:** Bereitstellung von Endpunkten für Authentifizierung, Suche und Verwaltung von Praktikumsstellen.
- **Implementierung zentraler Funktionalitäten:** Umsetzung aller erforderlichen Funktionen für die Nutzung und Verwaltung der Praktikumsbörse.
- **Dokumentation der REST-API:** Erstellung einer API-Dokumentation nach dem OpenAPI-Standard, um die Schnittstellen verständlich und transparent zu gestalten.

Der Schwerpunkt des Projekts liegt auf der objektorientierten Entwicklung, der Gestaltung und Nutzung relationaler Datenbanken, dem objektrelationalen Mapping sowie dem API-Design.

2. Projektplanung

2.1 Analysephase

In dieser Phase wurde eine Ist-Analyse durchgeführt, um die vorhandenen Herausforderungen und Bedürfnisse zu verstehen.

2.1.1 Ist-Analyse

Derzeit gibt es am Berufskolleg keine digitale Plattform, die eine zentrale Verwaltung und Suche von Praktikumsstellen ermöglicht. Unternehmen veröffentlichen ihre Angebote über verschiedene Kanäle wie Broschüren, eigene Webseiten oder externe Jobbörsen. Diese dezentrale Verteilung der Informationen führt zu einer unübersichtlichen und ineffizienten Handhabung des gesamten Prozesses. Für Lernende gestaltet sich die Suche nach passenden Praktikumsstellen dadurch schwierig, da keine zentrale Anlaufstelle existiert, die alle relevanten Informationen bündelt.

Auf Seiten der Betriebe besteht keine standardisierte Möglichkeit, Praktikumsangebote strukturiert zu veröffentlichen oder zu aktualisieren, was den Aufwand für die Verwaltung

und Verbreitung erhöht. Ebenso fehlen Funktionalitäten, die eine direkte Interaktion zwischen Lernenden und Betrieben erleichtern würden.

Die aktuellen Kommunikationswege und Abläufe bringen dabei zwei wesentliche Herausforderungen mit sich:

- **Unübersichtlichkeit und hoher Aufwand:** Die dezentrale Verteilung der Praktikumsangebote über verschiedene Kanäle macht es für Lernende schwer, passende Stellen zu finden, und erhöht den Aufwand für Betriebe, ihre Anzeigen aktuell zu halten.
- **Ineffizienz und Kommunikationsprobleme:** Der fehlende zentrale Zugang erschwert den Austausch zwischen Lernenden und Betrieben, was den Vermittlungsprozess zeitaufwendig und anfällig für Missverständnisse macht.

Diese Herausforderungen verdeutlichen, dass eine zentrale, digitale Plattform erforderlich ist, um die Verwaltung und Suche von Praktikumsstellen zu vereinfachen. Eine solche Lösung könnte nicht nur den Vermittlungsprozess effizienter gestalten, sondern auch die Kommunikation zwischen Lernenden und Betrieben verbessern sowie den Verwaltungsaufwand für beide Seiten erheblich reduzieren.

2.1.2 Soll-Konzept

Funktionale Anforderungen:

Nichtfunktionale Anforderungen

2.1.3 Wirtschaftlichkeitsanalyse

2.1.3.1 Kosten

2.1.3.2 Nutzen

2.1.3.3 Kosten Nutzen Vergleich

2.2 Definitionsphase

2.2.1 Pflichtenheft (Beschreibung von Umsetzung der Anwendung)

Der folgende Auszug aus dem Pflichtenheft beschreibt die geplante Umsetzung der Anforderungen, die im Soll-Konzept definiert wurden:

2.2.2 Wahl des Entwicklungsmodells

Die Wahl eines geeigneten Entwicklungsmodells war entscheidend, um einen strukturierten und effizienten Prozess für dieses Projekt sicherzustellen. Aufgrund des überschaubaren Projektumfangs und der bereits klar definierten Anforderungen wurde das erweiterte Wasserfallmodell (siehe Abbildung 2) als optimale Methode ausgewählt.

Dieses Modell überzeugt durch seine klare Struktur und hohe Planungssicherheit. Da die Anforderungen bereits vor Beginn des Projekts festgelegt wurden, konnten potenzielle Risiken und Herausforderungen frühzeitig identifiziert und berücksichtigt werden. Die klar

definierte Abfolge der Projektphasen erlaubt zudem eine rückwirkende Fehlerkorrektur sowie den gezielten Rücksprung in vorherige Phasen bei Bedarf.

Da das Projekt in Einzelarbeit umgesetzt wurde, war der Einsatz agiler Vorgehensmodelle nicht erforderlich, da deren Vorteile insbesondere bei größeren Projekten mit mehreren Teammitgliedern zum Tragen kommen.

Zusammenfassend bietet das erweiterte Wasserfallmodell eine solide und strukturierte Basis für eine erfolgreiche Projektdurchführung. Die klare Definition der Anforderungen und die starke Fokussierung auf Planungssicherheit gewährleisten einen effizienten und zielgerichteten Entwicklungsprozess.

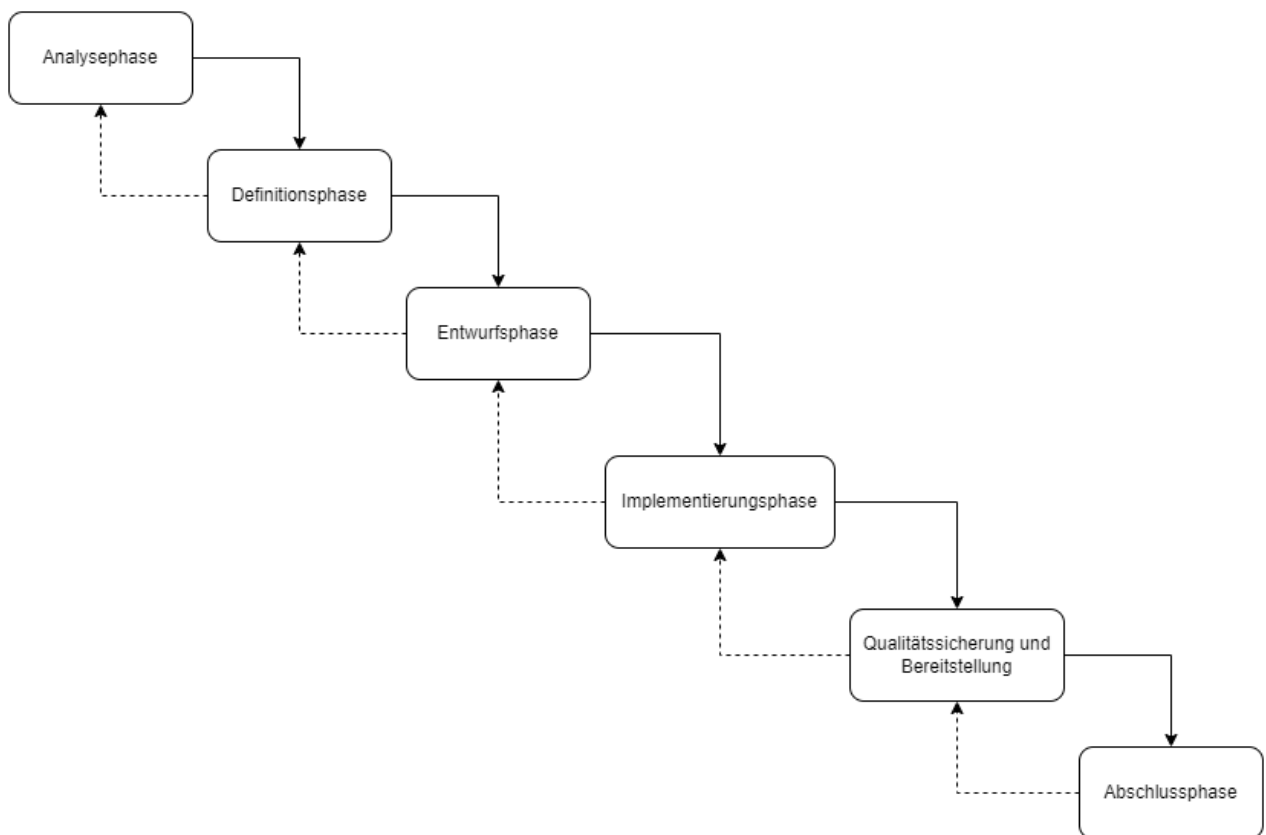


Abbildung 1 Erweitertes Wasserfallmodell

2.2.3 Zeitplan

Durch die Erstellung des Zeitplans ist ein Aufwand von 80 Stunden für die Realisierung des Projekts ersichtlich. Im Folgenden ist die grobe Zeitplanung abgebildet während sich die detaillierte im Anhang befindet ([siehe Anhang A.1](#)).

Arbeitspaket	Geplanter Aufwand (Stunden)
Projektinitialisierung	2
Analysephase	3
Definitionsphase	12
Entwurfsphase	4
Implementierungsphase	38

Qualitätssicherung und Bereitstellung	7
Abschlussphase	12
Puffer	2
Gesamt	80

Tabelle 1 Grobe Zeitplanung

2.2.4 Meilensteinplan

Beschreibung und eine Tabelle für den Meilensteinplan

Nr.	Meilenstein	Plan-Termin	Dokumente
1	Projektinitialisierung	2 Stunden	
2	Analysephase abgeschlossen	5 Stunden	Ist-Analyse Soll-Konzept Wirtschaftlichkeitsbetrachtung
3	Definitionsphase abgeschlossen	17 Stunden	Pflichtenheft Erweitertes Wasserfallmodell Zeitplan Meilensteinplan Ressourcenplan Meilensteinanalyse Qualitätssicherungsplan Testfallkatalog
4	Entwurfsphase abgeschlossen	21 Stunden	Anwendungsfalldiagramm Aktivitätsdiagramm Klassendiagramm Entity-Relationship Model
5	Implementierung abgeschlossen	59 Stunden	Kommentierter Quellcode
6	Testen und Containerisierung abgeschlossen	66 Stunden	
7	Abschlussphase abgeschlossen	78 Stunden	Benutzerdokumentation Soll-Ist-Vergleich

Tabelle 2 Meilensteinplan

2.2.5 Ressourcenplan

Für die erfolgreiche Umsetzung des Projekts steht eine Vielzahl an Ressourcen bereit. Ein Großteil dieser Ressourcen ist bereits vorhanden und verursacht daher keine zusätzlichen Kosten. Dazu zählen unter anderem die erforderliche Hardware, Software und Entwicklungsumgebung.

Die einzige spezifizierte Ressource, die zusätzlich eingeplant wird, ist die Arbeitszeit des internen Projektmanagers und Entwicklers, die mit insgesamt 80 Stunden veranschlagt ist.

Prozessorientierter Projektbericht

Diese Ressourcen sind ideal auf die Anforderungen des Projekts abgestimmt und gewährleisten eine reibungslose sowie effiziente Umsetzung.

Eine ausführliche Auflistung der verfügbaren Ressourcen findet sich nachfolgend:

Mitarbeiter

- Projektmanager/Entwickler - Planung und Umsetzung (Intern)
 - Gesamtarbeitszeit: 80 Stunden

Hardware

- AMD Ryzen 7 4700U @ 2GHz mit Radeon Graphics (Prozessor)
- 16,0 GB (Arbeitsspeicher)
- Intel SSD NVMe | 476 GiB (Festplatte)

Software

- Windows 11 (Betriebssystem)
- Visual Studio Code (Entwicklungsumgebung)
- Git (Version Control)
- Microsoft Word (Projektdokumentation)
- Draw.io (Diagramme / Graphen)
- Docker Desktop (Containerisierung)

2.2.6 Meilensteintrendanalyse

Die Meilensteintrendanalyse ([siehe Abbildung 2](#) unten) wird regelmäßig durchgeführt, um den Fortschritt des Projekts zu überwachen und Trends in Bezug auf die Erreichung der Meilensteine ([siehe Tabelle 2](#)) zu analysieren. Hierbei werden die geplanten Termine mit den tatsächlichen Fertigstellungsdaten verglichen, um Abweichungen wie Verzögerungen oder Beschleunigungen zu identifizieren.

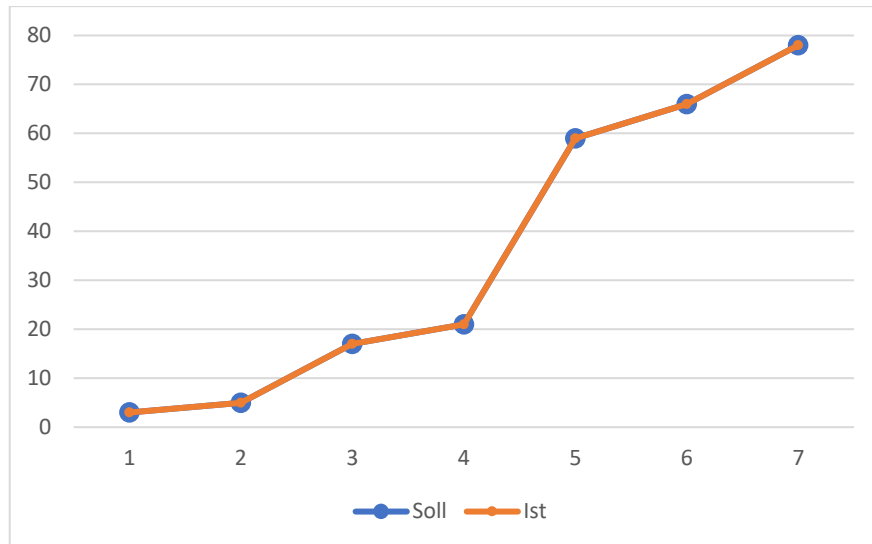


Abbildung 2 Meilensteintrendanalyse

2.2.7 Qualitätssicherungsplan

Um sicherzustellen, dass das Projekt die höchsten Qualitätsstandards erreicht und die festgelegten Ziele erfolgreich umgesetzt werden, wurde ein Qualitätssicherungsplan entwickelt. Dieser Plan umfasst zentrale Maßnahmen wie die Erstellung eines Pflichtenhefts, die Definition von Meilensteinen, die Entwicklung eines Testfallkatalogs und die Durchführung einer Wirtschaftlichkeitsanalyse.

Der Qualitätssicherungsplan umfasst die folgenden Maßnahmen:

- Erstellung eines Pflichtenhefts
- Erstellung eines Meilensteinplans
- Erstellung eines Testfallkatalogs
- Durchführung der Meilensteintrendanalyse
- Durchführen von einem Soll-Ist-Vergleich
- Durchführen einer Wirtschaftlichkeitsanalyse

Durch kontinuierliche Qualitätsmaßnahmen wird sichergestellt, dass das Projekt erfolgreich abgeschlossen wird. Eine gründliche Planung und konsequente Umsetzung der genannten Maßnahmen sind entscheidend für einen reibungslosen Projektverlauf und einen erfolgreichen Abschluss.

2.2.8 Testfallkatalog

Im Anhang

3. Projektdurchführung

3.1 Technologien

Technologien-Stack und Begründung für jede Technologie

3.2 Entwurfsphase

Erstellung von Diagrammen

3.2.1 Anwendungsfälle

Im Anhang

3.2.2 Aktivitätsdiagramm

Im Anhang

3.2.3 Klassendiagramm

Im Anhang

3.2.4 Datenbankmodell - ERD

Im Anhang

3.3 Implementierungsphase:

Die Umsetzungen werden hier ausführlich beschrieben.

Beschreibung der Implementierung von JWT-Authentifizierung, API-Endpunkten, Funktionalitäten

3.4 Testen der Anwendung

Beschreibung von Testen

(Ggf. Screenshot von Testen und im Anhang hinzufügen)

3.5 Containerisierung der Anwendung

Beschreibung von Containerisierung

4. Projektabschluss

4.1 Projektergebnis

Die Entwicklung der Anwendung wurde erfolgreich abgeschlossen, womit das Projektziel erreicht wurde.

4.2 Benutzerdokumentation (API-Dokumentation)

Screenshots im Anhang

5. Projektreflektion

Durch die umfangreiche Planung gab es beim Absolvieren des Projekts keinerlei Unklarheiten. Durch die sorgfältige Analyse der Anforderungen und die Festlegung eines detaillierten Zeitplans war es möglich, sich auf die Umsetzung der Aufgaben zu konzentrieren und potenzielle Hindernisse frühzeitig zu identifizieren.

Die Verwendung bewährter Technologien und Entwicklungspraktiken sowie die Einhaltung von Best Practices trugen ebenfalls zur Effizienz und Qualität des Projekts bei.

5.1 Soll-Ist-Vergleich

Beschreibung für den Vergleich und mit einer Tabelle wie folgende:

Arbeitspaket	Geplanter Aufwand (Stunden)	Tatsächlicher Aufwand (Stunden)
Projektinitialisierung	2	2
Analysephase	3	3
Definitionsphase	12	12
Entwurfsphase	4	4
Implementierungsphase	38	38
Qualitätssicherung und Bereitstellung	7	7
Abschlussphase	12	12
Puffer	2	-
Gesamt	80	78

5.2 Projekterfolg

Der Projekterfolg ist an mehreren Kriterien bewertbar. Mithilfe des Qualitätssicherungsplans ist ersichtlich, dass das Projekt mit angemessener Qualität fertiggestellt wurde. An dem Soll-Ist-Vergleich ist zu sehen, dass der Zeitplan erfolgreich eingehalten wurde. Während die Wirtschaftlichkeitsanalyse als Beweis für den Finanziellen Erfolg bietet.

Eidesstattliche Erklärung

Anhang

A.1 Detaillierter Zeitplan

Arbeitspaket	Geplanter (Stunden)	Aufwand
Projektinitialisierung	2	
Projektbesprechung	2	
Analysephase	3	
Durchführung der Ist-Analyse	1	
Erstellung eines Soll-Konzepts	1	
Erstellung einer Wirtschaftlichkeitsbetrachtung	1	
Definitionsphase	12	
Erstellung des Pflichtenhefts	3	
Abnahme des Pflichtenhefts	1	
Definition des Zeitplans	2	
Definition des Erweitertes Wasserfallmodells	1	
Definition des Ressourcenplans	1	
Definition des Meilensteinplans	1	
Definition der Meilensteintrendanalyse	1	
Definition des Qualitätssicherungsplans	1	
Definition des Testfallkatalogs	1	
Entwurfsphase	4	
Erstellung vom UML-Use-Case-Diagramm	1	
Erstellung vom UML-Aktivitätsdiagramm	1	
Erstellung vom UML-Klassendiagramm	1	
Erstellung vom ER-Diagramm	1	
Implementierungsphase	38	
Implementierung der REST-API Endpunkte	6	
Implementierung der Datenbankschema	3	
Implementierung der Authentifizierung	5	
Implementierung der administrativen Funktionalität	8	
Implementierung der Funktionalität für betriebliche Benutzer	8	
Implementierung der Funktionalität für schulische Benutzer	8	
Qualitätssicherung und Bereitstellung	7	
Testen	3	
Containerisierung von der Anwendung	4	
Abschlussphase	12	
Erstellen einer API-Dokumentation	1	
Projektabschluss durch den Auftraggeber	1	
Durchführung Soll-Ist-Vergleich	1	
Verfassung des prozessorientierten Projektberichts	9	
Puffer	2	
Gesamt	80	

A.2 Anwendungsfalldiagramm

A.3 Aktivitätsdiagramm

A.4 Klassendiagramm

A.5 Entity-Relationship Model

A.6 Testfallkatalog

A.7 Kommentierter Quellcode

A.8 Testen

A.9 Benutzerdokumentation (API-Dokumentation)

A.10 Abnahmeprotokoll

A.11 Freigabeerklärung