**Dokumentation einer beispielhaften Projektarbeit**

**im Rahmen der Abschlussprüfung zum Fachinformatiker/-in für Anwendungsentwicklung**

„Erstellung eines Vokabeltrainers“

**Bearbeitungszeitraum:**

[02.03.2020] bis [23.03.2020]

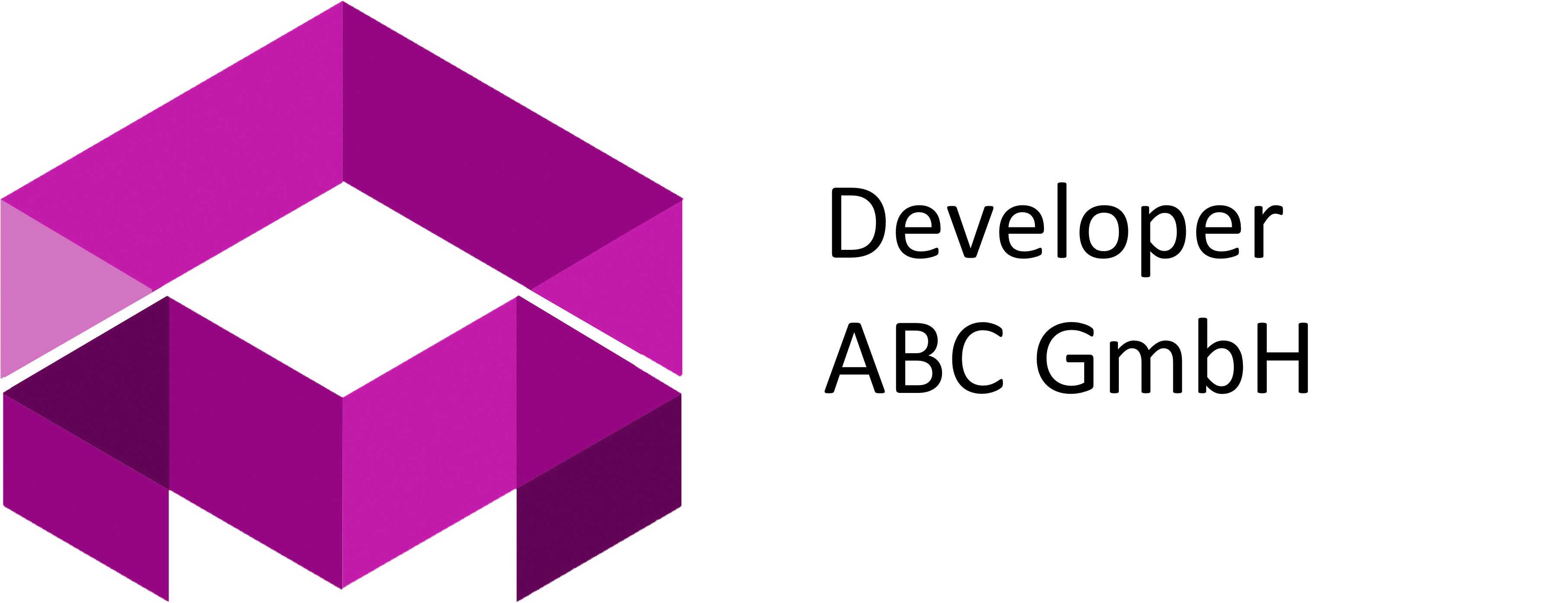
**Prüfungsteilnehmer/-in**: Bernhardt, Domenik

Trockenbrodt, Tobias

Sowa, Falco

Schneider, Korina

Weitere Angaben zu den Prüfungsteilnehmern entfallen.



**Ausbildungsbetrieb**: Developer ABC GmbH

Ada Lovelace Straße 42

12345 Berlin

**Projektverantwortliche**: Frau Krebs

**Prüfungsausschuss**: FIAN XX

**Inhalt**

[1 Einleitung 1](#_Toc34839206)

[1.1 Projektumfeld 1](#_Toc34839207)

[1.2 Projektziel (Soll-Konzept) 1](#_Toc34839208)

[1.3 Projektbegründung 1](#_Toc34839209)

[1.4 Projektschnittstellen 1](#_Toc34839210)

[2 Projektplanung 2](#_Toc34839211)

[2.1 Projektphasen 2](#_Toc34839212)

[2.2 Projektressourcen 2](#_Toc34839213)

[2.3 Kostenplanung/-kalkulation 4](#_Toc34839214)

[2.3.1 Make-or-Buy-Entscheidung 4](#_Toc34839215)

[2.3.2 Kostenplanung Entwicklungskosten 4](#_Toc34839216)

[2.3.3 Kostenplanung Folgekosten 4](#_Toc34839217)

[2.3.4 Amortisierung 5](#_Toc34839218)

[3 Projektentwurf 6](#_Toc34839219)

[3.1 Ist-Analyse 6](#_Toc34839220)

[3.2 Design/Entwurf des Programms 6](#_Toc34839221)

[4 Projektdurchführung 7](#_Toc34839222)

[4.1 Implementierung/Realisierung 7](#_Toc34839223)

[4.2 Tests/Qualitätssicherungsmaßnahmen 7](#_Toc34839224)

[4.3 Kundenübergabe 7](#_Toc34839225)

[5 Reflektion 8](#_Toc34839226)

[5.1 Soll-Ist-Vergleich 8](#_Toc34839227)

[5.2 Rückblick & Fazit 9](#_Toc34839228)

[5.3 Ausblick 9](#_Toc34839229)

[6 Quellen 10](#_Toc34839230)

[7 Anhang 10](#_Toc34839231)

[Anhang 1: Pflichtenheft/Lastenheft 10](#_Toc34839232)

[Anhang 2: Benutzerhandbuch 10](#_Toc34839233)

[Anhang 3: Quelltextauszüge 11](#_Toc34839234)

[Anhang 4: Angebot für die Volkshochschule 11](#_Toc34839235)

[Anhang 5: Annahmebestätigung des Angebots 11](#_Toc34839236)

[Anhang 6: [Weitere Anhänge] 11](#_Toc34839237)

## 1 Einleitung

### 1.1 Projektumfeld

Das Projekt findet im Rahmen des AS-Unterrichts im zweiten Ausbildungsjahr statt. Es wird von einem Team von 4 Entwicklern bearbeitet. Die Volkshochschule ist in diesem Projekt der fiktive Auftraggeber. Als Kundenkontakt und Ansprechpartner ist Frau Krebs zu nennen. Frau Krebs übernimmt auch die Projektabnahme.

### 1.2 Projektziel (Soll-Konzept)

Für ein Volkshochschulkurs soll zu Schulungszwecken mit einem Vokabeltrainer gearbeitet werden. Dieser Vokabeltrainer soll im Rahmen der Projektarbeit erstellt werden. Die grafische Benutzeroberfläche des Vokabeltrainers soll Vokabeln und deren mögliche Übersetzungen anzeigen. Dabei werden vier verschiedene Übersetzungen angezeigt, wobei nur eine Übersetzung, die jeweils Richtige ist. Der Nutzer der Anwendung erhält ein direktes Feedback, welche der Übersetzungen die Richtige ist, sobald er eine Übersetzung ausgewählt hat. Die Vokabeln für den Vokabeltrainer sind in einer XML-Datei gespeichert.

### 1.3 Projektbegründung

Derzeitig werden die Fragen in Papierform von jedem Teilnehmer beantwortet und eine Aufsicht korrigiert die Antworten der Teilnehmer im Anschluss. Die Auswertung der Vokabeln und deren Übersetzung kostet unnötige Zeit. Auch würde eine Anwendung den Ausdruck auf Papier überflüssig machen und erspart so die Druckkosten. Da die Volkshochschule seit 2015 einen Computerraum für die Teilnehmer zur Verfügung stellt, entstehen keine weiteren Kosten für die Bereitstellung der Hardware.

### 1.4 Projektschnittstellen

Nutzer der Anwendung sind ausschließlich Teilnehmer der Volkshochschule. Dabei bleibt der Benutzer anonym. Es werden keine Daten des Benutzers erfasst oder gespeichert. Zudem soll die Anwendung mit einer XML-Datei arbeiten, wobei lediglich lesend auf die Daten zugegriffen wird.

## 2 Projektplanung

### 2.1 Projektphasen

|  |  |
| --- | --- |
| **Projektphasen** | **Soll-Zeit in h** |
| **Analysephase:** |  |
| Ist-Analyse und Definition der Anforderungen im Kundengespräch | 0,5 |
| Ressourcenplanung | 1,0 |
| Wirtschaftlichkeitsanalyse | 1,0 |
| **Entwurfsphase:** |  |
| Entwurf GUI | 1,0 |
| Entwurf Unittests | 1,0 |
| Entwurf Klassenlogik (UML) | 2,0 |
| **Implementierungsphase:** |  |
| Erstellung der GUI | 2,5 |
| Erstellung der Komponenten | 6,0 |
| Erstellung der Unittests | 3,0 |
| **Qualitätsmanagementphase:** |  |
| Durchführung und Auswertung Unittests | 3,0 |
| Durchführung und Auswertung Blackboxtest | 2,0 |
| Debugging | 2,0 |
| **Dokumentationsphase:** |  |
| Erstellung der Projektdokumentation (PD) | 7,0 |
| Korrekturlesen der PD | 1,0 |
|  |  |
| **Gesamt** | **33,0** |

Bei unserem Vorgehen haben wir uns grob am sogenannten Wasserfallmodell orientiert.

Dieses sieht vor, dass es klar abgegrenzte Phasen gibt, die jeweils über vordefinierte Start- und Endpunkte mit eindeutigen Ergebnissen verfügen.

[IHK: detaillierte Zeitplanung mit Gliederung in Haupt- und Teilaufgaben. Beschreibung/Begründung des gewählten Vorgehensmodells]

### 2.2 Projektressourcen

#### 2.2.1 Personelle Ressourcen

Das Projekt wird mit 4 Auszubildenden der Developer ABC GmbH realisiert.

Namentlich Domenik Bernhardt, Tobias Trockenbrodt, Falco Sowa und Korina Schneider.

Da Herr Bernhardt krankheitsbedingt nur an 4 der 6 Arbeitseinheiten von 90 Minuten teilnehmen konnte, beläuft sich die tatsächliche Gesamtarbeitszeit in unserem Entwicklerteam entgegen der Projektvorgabe lediglich auf 1.980 Minuten bzw. 33 Stunden, statt der angesetzten 36 Stunden.

Die folgende Tabelle (Abbildung xxx) zeigt die geplante Aufgabenverteilung und die daraus resultierenden Personalkosten je Vorgang.



**GesamtkostenT – Gesamtkosten für die ausgeführte Tätigkeit**

**GesamtkostenP – Gesamtkosten für das eingesetzte Personal**

#### 2.2.2 Software-Ressourcen

Zur Umsetzung des Projekts, sowie der Erstellung der Dokumentation wird folgende Software eingesetzt:

1. Betriebssystem Microsoft Windows 10

2. Microsoft Visual Studio 2019 Community Edition

3. Microsoft Word

4. Microsoft Excel

5. GitHub

6. Dia

Die Kosten für die eingesetzten Softwareressourcen werden hier nicht berücksichtigt, da es sich entweder um freie oder bereits vorhandene firmenweit eingesetzte Software handelt und daher keine zusätzlichen Kosten für deren Beschaffung entstanden.(sind)

#### 2.2.3 Hardware-Ressourcen

Die Hardwareressourcen umfassten im Wesentlichen 4 Hardwarearbeitsplätze der Developer ABC GmbH.

Die Kosten für Hardware und sonstige Büroausstattung wurden im Projekt fix mit 500,00 € pro Entwickler, d. h. insgesamt 2000,00 €, angesetzt.

Da sich trotz des kurzzeitigen Ausfalls eines Entwicklers an den bereitgestellten Ressourcen keine Änderungen ergaben, übernehmen wir diesen Fixbetrag in unsere Kostenaufstellung.

|  |  |
| --- | --- |
| **Ressourcenplanung Büro/Ausstattung (fix)** | |
| **GesamtkostenBuA in €** | **2.000,00** |

**GesamtkostenBuA – Gesamtkosten für die eingesetzte Hardware und Büroausstattung**

### 2.3 Kostenplanung/-kalkulation

### 2.3.1 Make-or-Buy-Entscheidung

Da die VHS an uns herangetreten ist, um ein Programm, das ihren Anforderungen entspricht, zu entwickeln, ist die Entscheidung offensichtlich bereits gefallen.

Entweder verfügt unser Kunde nicht über die notwendigen personellen und infrastrukturellen Ressourcen, um das gewünschte Programm selbst zu entwickeln oder es hat sich in seiner Kalkulation herausgestellt, dass es zeit-/kostengünstiger ist, dieses von der Developer ABC GmbH entwickeln zu lassen.

### \*(ggf. Rücksprache mit Frau Krebs)

### 2.3.2 Kostenplanung Entwicklungskosten

Aus den unter Punkt 2.2.1 und 2.2.3 ermittelten Gesamtkosten für das eingesetzte Personal, sowie die verwendete Hardware und Büroausstattung, ergeben sich für das Projekt Gesamtkosten in Höhe von 2.495,00 Euro.



**GesamtkostenProjektEntwicklung – Gesamtkosten für die Projektentwicklung**

### 2.3.3 Kostenplanung Folgekosten

Da bereits bei der Entwicklung unseres Vokabeltrainers besonders seine Usability und seine Wartbarkeit im Fokus standen, werden sich die Folgekosten für Wartungen und Schulungen auf ein Minimum belaufen.

Aufgrund des angesetzten Stundensatzes pro Entwickler in Höhe von 15,00 Euro und des Material- und Bürokostensatzes von ca. 14 Euro pro Stunde gehen wir jährlich von Zusatzkosten in Höhe von etwa 174,00 Euro aus.

Dies umfasst ca. 6 Arbeitsstunden, die in die Wartung und Optimierung verschiedener Komponenten und/oder in die Schulung der VHS-Mitarbeiter fließen können.

Da das Programm für Menschen ohne körperliche oder geistige Einschränkungen keine erkennbaren Hürden bietet, halten wir Einführungsschulungen generell für redundant.

Sollten seitens des Kundens größere Veränderungen/Anpassungen gewünscht werden, können die jährlichen Kosten entsprechend variieren.

### 2.3.4 Amortisierung

Um zu ermitteln ab welchem Zeitpunkt sich die Entwicklung und Implementation unserer Anwendung für unseren Kunden „rechnet“, also amortisiert hat, betrachten wir die monatlichen Kosten, die nach dem bisher genutzten Vorgehen entstehen und stellen sie den Entwicklungs- und Folgekosten des Vokabeltrainers gegenüber.

Zunächst einmal spielen hier die materiellen Kosten des Ausdruckens von Übungsblättern für die Kunden der VHS eine Rolle. Die Druckkosten pro Blatt belaufen sich auf 0,01 Euro.

Bei ca. 10 Seiten pro Kursteilnehmer und ca. 100 Teilnehmern verschiedener Sprachkurse pro Monat zahlt die VHS annähernd 10,00 Euro monatlich für Ausdrucke.

Hinzu kommen die personellen Kosten für die manuelle Erstellung und Auswertung der Übungsblätter. Bei einem durchschnittlichen Stundenlohn von 20,00 Euro pro Dozent und etwa 12 Arbeitsstunden für die Bearbeitung der Blätter entstehen monatlich Kosten in Höhe von (20,00 €/h x12h) +10€ = 250 €.

Die Kosten unseres Projekts belaufen sich im ersten Jahr auf 2495,00 Euro für die Entwicklung und 174,00 EURO für eventuelle Wartungsarbeiten.

Der Break-Even-Punkt, also der Zeitpunkt ab dem die bisherige Vorgehensweise und die Gesamtkosten der Vokabeltrainereinführung identisch sind, ist bereits nach 0,88 Jahren bzw. 10 Monaten 20 Tagen 6 Stunden 43 Minuten und 12 Sekunden erreicht. (vgl. Abbildung xxx)

In den Monaten nach dem Break-Even wäre das bisherige Vorgehen teurer als der Einsatz unseres Programms.

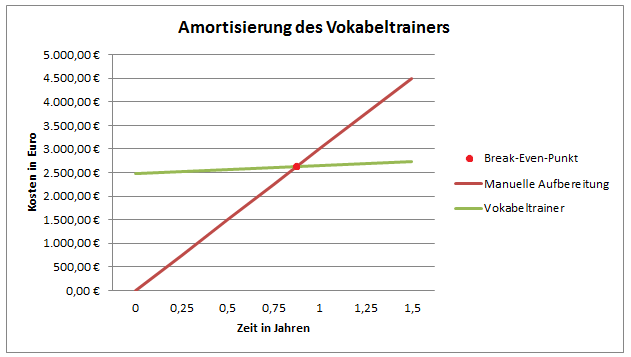


Abbildung 1: Amortisierung des Vokabeltrainers und Break-Even-Punkt

## 3 Projektentwurf

### 3.1 Ist-Analyse

Aktuell verwendet unser Kunde (VHS) zur Überprüfung des Wissensstandes seiner Sprachkursteilnehmer Übungsblätter in Papierform, die manuell erstellt, kontrolliert und ausgedruckt werden müssen. D.h. ein Dozent der VHS muss sich für jeden seiner Kurse Fragen und Antwortmöglichkeiten überlegen, diese in einem geeigneten Programm aufarbeiten und ausdrucken. Die Ausdrucke muss er/sie an seine/ihre Teilnehmer verteilen, sie nach der Übung wieder einsammeln, kontrollieren und wieder ausgeben.

Da dies neben Kosten für Papier und Ausdrucke vor allem auch personellen Aufwand generiert, wünscht sich unser Kunde eine programmatische Lösung für diese Vorgänge.

Diese ist im Übrigen auch im Hinblick auf nachhaltiges Wirtschaften zu bevorzugen, da weniger Papier und Druckertinte verbraucht wird. Zudem sind Fehler bei der Erstellung der Vokabeltests leichter vermeidbar, solange die Eingangsdatei (XML) korrekt angelegt wurde.

[IHK: Wie ist die bisherige Situation (z.B. bestehende Programme, Wünsche der Mitarbeiter)? Was gilt es zu erstellen/verbessern?]

### 3.2 Design/Entwurf des Programms

[IHK:

Beschreibung des Programms, Ziel der Entwicklung

Bei unserem Programm handelt es sich um eine eigenständige, ereignisorientierte Windows-GUI-Anwendung.

Eine Anbindung an existierende Programme ist nicht vorgesehen.

Das Programm soll auch ungeschulten Nutzern mit wenig Erfahrung im Umgang mit PC-Anwendungen eine möglichst intuitive Oberfläche bieten, die sie nicht überfordert.

Die Anwendung muss auf Rechnern mit dem Betriebssystem Windows ab 7 (oder höher) lauffähig sein. Es bot sich daher an als Programmiersprache C# zu wählen, da die für die Ausführung benötigte .NET API auf Windowsrechnern standardmäßig vorinstalliert ist und somit keine zusätzlichen Programme installiert und gewartet werden müssen.

//ToDo

- Funktionen des Programms: z.B. mit Use-Case-/Aktivitätsdiagramm, EPK

* **Darstellung der eingesetzten Datenstrukturen** (z.B. XML, CSV o.ä.)
* **Geschäftslogik**: z.B. Komponenten-/Klassen-/Sequenz-/Datenflussdiagramm, Architekturplanung, EPK
* Benutzerschnittstelle: z.B. GUI, Webinterface, Entwurf/Gestaltung der Oberflächen/Masken, Corporate Identity, Einbindung in andere Anwendungen
* Um der Corporate Identity des Kunden entgegenzukommen, haben wir sein Logo in die Anwendung eingebunden.
* Qualitätsmerkmale: z.B. Anforderungen hinsichtlich Performance, Usability, Effizienz etc. (ISO 9126)
* Qualitätssicherung: Testszenarien, Benutzer-/Entwicklertests, Code-Reviews, statische Codeanalyse]

## 4 Projektdurchführung

### 4.1 Implementierung/Realisierung

[IHK: Programmierung (interessante Funktionen zeigen, Quelltextbeispiele); Screenshots der Oberfläche]

### 4.2 Tests/Qualitätssicherungsmaßnahmen

[IHK: z.B. Test-Reports, Unit-Tests, Code-Reviews]

### 4.3 Kundenübergabe

Die Übergabe an den Kunden erfolgt auf Kundenwunsch digital. Der Kunde plant das Programm in Augenschein zu nehmen und wird uns kontaktieren, falls Rückfragen entstehen.

Wir betrachten das Projekt mit dem Zeitpunkt der Übergabe als abgeschlossen.

Nutzerschulungen im Rahmen der Übergabe nicht vorgesehen.

Da wir die Lauffähigkeit des Programms unter Realbedingungen getestet haben, sind (unter Windows 10 Home) keine Probleme zu erwarten.

Sollte es dennoch zu Herausforderungen auf Seiten des Kunden kommen, werden wir den notwendigen Support in den ersten 3 Monaten nach der Übergabe kostenfrei durchführen.

[IHK: Form der Übergabe, Nutzungsschulungen, Lauffähigkeit]

## 5 Reflektion

### 5.1 Soll-Ist-Vergleich

Wie in der Projektplanung (xxx) beschrieben, wurde ein funktionaler (hopefully) Vokabeltrainer erstellt und für die Nutzung durch den Kunden freigegeben.

Die Umstellung der analogen Vokabelkontrollen in digitale konnte somit erfolgreich (hopefully) abgeschlossen werden.



Größere Abweichungen gab es in der Zeitplanung vor allem in Bezug auf Entwicklung und Implementierung, die - ähnlich wie die Ausarbeitung der Projektdokumentation - mehr Zeit in Anspruch nahmen als wir antizipiert hatten. Glücklicherweise verliefen das Debugging und die Unit- und Blackboxtests schneller als in der Planungsphase angenommen, weshalb die Gesamtbearbeitungszeit dennoch eingehalten werden konnte.

### ***5.2 Rückblick & Fazit***

//ToDo

Bei der Bearbeitung des Projekts …

Rückblickend betrachtet funktionierte die Teamarbeit …

Absprachen wurden (meist) eingehalten

An den Kosten änderte sich nichts, da keine zusätzlichen Anschaffungen nötig waren und auftretende Herausforderungen innerhalb des Teams – ohne Zuhilfenahme anderer Ansprechpartner – bewältigt werden konnten.

### ***5.3 Ausblick***

Erweiterungsmöglichkeiten sehen wir insbesondere in Hinblick auf die Feedbackfunktionen des Programms. Aktuell erhält der Anwender lediglich einen groben Überlick darüber wie gut er die abgefragten Vokabeln beherrscht, in der Zukunft könnte nach Beendigung der Lerneinheit eine Liste abgerufen werden, die alle Vokabeln nach Lernstatus sortiert anzeigt.

Damit wäre der Nutzer in der Lage seine Lernbestrebungen auf die Wörter zu fokussieren, die ihm bisher noch Schwierigkeiten bereiten.

Desweiteren wäre über eine Sprachausgabe nachzudenken, die jedoch eine gewisse Vorarbeit auf Seiten der Volkshochschule (Erstellung/Erwerb Sprachdateien) voraussetzen würde.

Außerdem bestünde die Möglichkeit das Programm bezüglich seiner Barrierefreiheit zu optimieren. Das heißt, dass auch Menschen mit körperlichen Einschränkungen in die Lage versetzt würden den Vokabeltrainer uneingeschränkt zu nutzen.

Um die Erweiterbarkeit des Programms zu gewährleisten, haben wir Wert auf Modularität gelegt und bereits beim Designentwurf darauf geachtet ein Menüfenster vor dem eigentlichen Vokabeltest einzufügen.

Über dieses können nachträglich ggf. Optionen eingefügt werden, die die vorgenannten Funktionalitäten unterstützen.

[IHK: z.B. Wie ist das Projekt rückblickend zu bewerten?

Begründung von Änderungen zum Projektantrag;

Soll-/Ist-Vergleich: Wurde das Ziel erreicht? Wurden die Kosten/Zeiten eingehalten?

Ausblick: Erweiterungsmöglichkeiten, Anschlussprojekte, Akzeptanz der Benutzer; Lessons Learned, kritische Bewertung, persönliches Fazit: Was lief gut? Was lief nicht gut?]

## 6 Quellen

* Leitfaden zur IHK-Abschlussprüfung Fachinformatik/-in für Anwendungsentwicklung, IHK Berlin, Stand 2013
* Macke, S.: <https://fachinformatiker-anwendungsentwicklung.net/inhalte-der-projektdokumentation/>, Letzter Aufruf: 17.02.2019

[Beispiel einer Quellenangabe für ein Buch:

* Kecher, C.: UML 2.0 Das umfassende Handbuch. Galileo Computing, Bonn 2005, Seite 76 ff]

## 7 Anhang

### Anhang 1: Pflichtenheft/Lastenheft

Auf die Erstellung eines Lasten- und Pflichtenheftes wird in dem Beispielprojekt verzichtet. Dieser Abschnitt ist im Sinne der Projektdokumentation einer Abschlussprüfung für den Beruf Fachinformatiker/-in Anwendungsentwicklung nur der Vollständigkeit halber aufgeführt.

### Anhang 2: Benutzerhandbuch

Dieser Abschnitt im Sinne der Projektdokumentation einer Abschlussprüfung für den Beruf Fachinformatiker/-in Anwendungsentwicklung ist für das Beispielprojekt nicht relevant und wird nur der Vollständigkeit halber aufgeführt.

### Anhang 3: Quelltextauszüge

### Anhang 4: Angebot für die Volkshochschule

[Dieser Abschnitt entfällt.]

### Anhang 5: Annahmebestätigung des Angebots

Dieser Abschnitt im Sinne der Projektdokumentation einer Abschlussprüfung für den Beruf Fachinformatiker/-in Anwendungsentwicklung ist für das Beispielprojekt nicht relevant und wird nur der Vollständigkeit halber aufgeführt.

### Anhang 6: [Weitere Anhänge]

[IHK: Lasten-/Pflichtenheft; Datenbankentwurf; UML-Diagramme, EPKs, Flusspläne, PAPs, Entwürfe/Screenshots der Oberflächen, Dokumentation (Entwickler/Benutzer), Glossar, Quelltexte]