

Arbeitsplan

Inhaltsverzeichnis

1. Projektvision	2
2. Voraussetzungen	2
3. Designübersicht und Funktionalität.....	3
3.1 Designvorstellungen.....	3
3.2 Funktionale Anforderungen	3
3.2.1 Muss-Ziele	3
3.2.2 Kann-Ziele.....	4
4. Arbeitspakete	4
4.1 Muss-Ziele.....	4
4.2 Kann-Ziele.....	5
5. Vorprojekt.....	5
6. Glossar.....	6

1. Projektvision

Das Produkt soll am Ende des Projekts dafür genutzt werden, eine Online-Plattform mit englischsprachigen Präsentationen im pptx³⁰-Format zu füllen. Dazu werden Informationen zu wichtigen Themen, z.B. Städte oder Mineralien, aus Quellen wie Wikipedia, DBpedia³³ o.ä. über ein Java-Programm gesammelt und anschließend für die einzelnen Folien aufbereitet. Eine API⁹ soll es ermöglichen, daraus die Präsentationen anzufertigen und abzuspeichern. Eine GUI⁵ ist für dieses Programm keine Notwendigkeit, jedoch eine sinnvolle Erweiterung, sollte die Zeit im Rahmen des Projekts dafür reichen. Wichtig ist außerdem eine umfangreiche Dokumentation inner- und außerhalb des Quelltextes, da das Programm später eventuell fortgeführt und erweitert werden soll und die Möglichkeit besteht, dass es Schülern zur Verfügung gestellt wird. Der Kunde hat außerdem festgelegt, dass alle genutzten Quellen kostenlos verfügbar sein müssen, damit die Präsentationen später auch kostenlos veröffentlicht werden können.

2. Voraussetzungen

Die folgenden Voraussetzungen werden vom Kunden an die Applikation gestellt:

- dem Benutzer muss es möglich sein ein Thema, beispielweise Stadt oder Mineral, und eine Instanz²⁴ dieses Themas, wie „Leipzig“, einzugeben
- es muss eine Automatische Extraktion von Informationen aus Datenbanken, ähnlich DBpedia oder Wikipedia, erfolgen
 - eine Netzwerk-/Internetanbindung zum Erreichen der Datenbanken ist notwendig
- die Informationen müssen Foliengerecht aufbereitet werden
- die bearbeiteten Informationen müssen formatiert in eine pptx-Präsentation eingefügt werden
 - eine Open Source API für Java zur Erstellung und Bearbeitung von pptx-Files ist erforderlich
- Quellenangaben: als Notiz zu jeder Folie und als abschließende Quellenfolie vorhanden
- Präsentationen müssen und werden auf der Plattform noch bearbeitet und nachgebessert, dementsprechend ist Fehlerfreiheit keine realistische Anforderung

3. Designübersicht und Funktionalität

3.1 Designvorstellungen

Das Programm soll über ein CLI³¹ funktionieren, Kenntnisse zur Bedienung sollten vorhanden sein. Der Benutzer wählt eines der zur Verfügung stehenden Themen mit einer konkreten Instanz (Beispieleingabe: „City: Leipzig“). Dann lädt das Programm von verschiedenen Quellen aus dem Internet Daten herunter. Dabei wird Information, die gezielt durch festgelegte, themenrelevante Suchbegriffe ausgewählt wird, ergänzt durch solche, die von einem eigenen Algorithmus³² aus der Datenmenge extrahiert wurde. Anschließend werden die Informationen in einer Präsentation inhaltlich sinnvoll angeordnet und den vereinbarten Standards³⁶ entsprechend formatiert. Diese Präsentation wird folglich im .pptx-Format ausgegeben, in eine .zip-Datei³⁷ verpackt, und am Ausführungsort des Programms gespeichert. Das Ergebnis steht dem Benutzer folglich zur Verfügung.

In einem Programmdurchlauf wird genau eine Präsentation erstellt, aber eine mögliche Erweiterung dessen wurde als Kann-Ziel festgehalten.

Da das Programm erst einmal einem kleinen Kreis von Benutzern im Fachbereich Informatik zur Verfügung steht, ist im Projekt selbst weder eine Benutzeroberfläche noch der Zugriff auf Parameter (wie gewählte Suchbegriffe zu einem Thema, Formatierung der Präsentation) vorgesehen.

Instanzen für Präsentationen sind, falls existent, frei wählbar, aber das Programm gibt eine Warnung aus, falls es zur gewählten Instanz nicht genug Informationen gibt, um eine relevante Präsentation zu erstellen.

3.2 Funktionale Anforderungen

3.2.1 Muss-Ziele

/FA 10/

Der Benutzer kann zu vorgegebenen Themen Präsentationen für mindestens 10 Instanzen erstellen lassen.

/FA 20/

Informationen zu Instanzen werden aus Onlinequellen extrahiert.

/FA 30/

Die Ausgabe ist eine Präsentation im .pptx-Format, welche in einer .zip-Datei verpackt ist.

/FA 40/

Die Präsentationen genügen, mit kleinen Abweichungen, formell dem mit dem Kunden vereinbarten Standard³⁶.

3.2.2 Kann-Ziele

/FA 10 op/

Das Programm lässt sich über eine grafische Benutzeroberfläche bedienen.

/FA 20 op/

Der Benutzer kann in einem Programmdurchlauf mehrere Präsentationen erzeugen lassen.

/FA 30 op/

Der Benutzer kann zu einem gewählten Thema Präsentationen zu zufälligen Instanzen erzeugen lassen.

/FA 40 op/

Der Speicherort für die Ausgabe ist vom Benutzer selbst bestimmbar.

/FA 50 op/

Der Benutzer hat Formatierungsoptionen zur individuellen Gestaltung zur Verfügung

/FA 60 op/

Der Benutzer kann Suchbegriffe zu einem Thema festlegen.

4. Arbeitspakete

4.1 Muss-Ziele

1. Vorprojekt - 15%

Das Vorprojekt stellt einen wichtigen ersten Schritt - das Proof-Of-Concept³⁴ - dar. Das Einlesen von Daten und deren Weg auf eine Folie bilden das Fundament des Projekts, auf dem die weiteren Arbeiten aufbauen.

2. Füllen erster Folien zu einem Thema - 20%

Ganze Folien können mit den richtigen Informationen zu Instanzen des Themas gefüllt werden. Die Informationen auf den Folien sind angemessen aufbereitet und zusammengehörig. Eine solche Folie sollte auch problemlos in einer Präsentation Verwendung finden können.

3. Erstellen erster Präsentationen zu einem Thema - 10%

Durch genügend inhaltlich verschiedene Folien, ist es möglich eine ganze Präsentation für Instanzen zu erstellen.

4. Abschluss der Arbeiten am ersten Thema - 20%

Der Informationsgehalt der Folien, also der Präsentation, wird auf ein angemessenes Niveau gebracht. Außerdem finden entscheidende Tests statt, sodass Bugs³⁵ gefunden und Optimierungsspielräume sichtbar werden. Die Präsentationserstellung für das erste Thema funktioniert wie gewünscht.

5. Erweitern der Themenauswahl - 15%

Aus den Erfahrungen des ersten Themas schließend, kann die Anzahl an Themen effektiv und realistisch erhöht werden. Hierbei muss jedoch auf jedes Thema speziell eingegangen werden, was für großen Aufwand sorgen kann.

6. Finale Tests, Optimierung, Bug-Fixing und Dokumentation - 20%

Die finale Phase des Projekts dient dem Durchführen von Tests, sowie der Umsetzung sich daraus ergebender Optimierungs- und Bug-Fixing-Vorhaben. Des Weiteren soll in dieser Phase die Dokumentation für die Abgabe finalisiert werden. Sollte diese Phase früh genug eintreten, ist es möglich sich auch Kann-Zielen zuzuwenden.

4.2 Kann-Ziele

- Eigene Benutzeroberfläche - 10%
- Erstellung mehrerer Präsentationen in einem Durchlauf - 10%
- Random-Funktion für Instanzauswahl - 7%
- Speicherortwahl durch Benutzer - 3%
- Erweitern des Informationsgehalts der Präsentationen - 10%
- Weiteres Vergrößern der Themenauswahl - 15%
- Formatierungsoptionen für die Präsentation - 10%
- Optionale Benutzerintegration in die Präsentationsgenerierung - 25%

5. Vorprojekt

Ziel des Vorprojekts ist es, eine Information zu einem festgelegten Stichwort (z.B. Einwohnerzahl) zu einer festgelegten Instanz (z.B. Leipzig) aus einer Onlinequelle abzurufen, auf eine Folie zu schreiben und diese in einer (aus nur einer Folie bestehenden) Präsentation im .pptx-Format auszugeben und als .zip-Datei zu speichern.

Dazu muss zunächst das Programm modelliert werden. Das Programm zerfällt dabei in zwei Bereiche: zum einen muss Information online gesucht, heruntergeladen und gegebenenfalls aufbereitet werden; zum anderen muss diese Information dann sinnvoll formatiert auf eine Folie geschrieben und die Präsentation gespeichert werden. Besonders zu beachten ist im ersten Teil die Struktur der Onlinequelle: Datenbanken, Fließtexte und formatierte Texte müssen unterschiedlich gehandhabt und nach Stichworten oder seitenspezifischen Strukturen durchsucht werden.

6. Glossar

Python¹

Eine universelle, üblicherweise interpretierte höhere Programmiersprache.

Java²

Eine objektorientierte Programmiersprache.

Plattform³

Bezeichnet in der Informatik eine einheitliche Grundlage, auf der Programme ausgeführt. und entwickelt werden können.

Jython⁴

Eine reine Java-Implementierung der Programmiersprache Python und ermöglicht somit die Ausführung von Python-Programmen auf jeder Java-Plattform.

GUI⁵

(engl. graphical user interface) - Eine grafische Benutzeroberfläche.

Webanwendung⁶

Die Datenverarbeitung und –auswertung findet hauptsächlich auf einem entfernten Webserver statt.

Desktopanwendung⁷

Wird lokal auf dem Rechner des Benutzers installiert und dort ausgeführt.

Präsentationssoftware⁸

Programm, das die Erarbeitung und Präsentation eines Vortrages oder Referates unterstützt.

API⁹

(engl. application programming interface) - Ein Programmteil, mit dessen Hilfe sich andere Programme an das Softwaresystem anbinden können.

HTML¹⁰

(engl. Hypertext Markup Language) - Textbasierte Auszeichnungssprache zur Strukturierung digitaler Inhalte wie Texte, Hyperlinks, Bilder und anderen.

Office Open XML¹¹

Von Microsoft entwickelte Dateiformate zur Speicherung von Bürodokumenten auf XML-Basis.

Library/Bibliothek¹²

Sammlung von Unterprogrammen/-Routinen; nicht eigenständig lauffähig.

Parsen/Parser¹³

Programm zur Zerlegung und Umwandlung einer Eingabe in ein bestimmtes Format.

String¹⁴

Folge von Zeichen aus einem definierten Zeichensatz; in Java ein Datentyp.

XML¹⁵

(engl. Extensible Markup Language) - Auszeichnungssprache zur Darstellung hierarchisch strukturierter Daten in Form von Textdateien.

Automatic summarization¹⁶

Automatisierte Zusammenfassung von Texten.

Extraction¹⁷

Wichtige Thesen aus einem Text werden extrahiert, ohne dabei die Sätze oder Wörter zu verändern.

Abstraction¹⁸

Umschreibung von wichtigen Kernpunkten („paraphrasing“).

NLP¹⁹

(engl. natural language processing) - Automatisierte Textverarbeitung und/oder -generierung.

JSON²⁰

(engl. JavaScript Object Notation) - Kompaktes Datenformat in einer einfach lesbaren Textform zum Zweck des Datenaustauschs zwischen Anwendungen.

URL²¹

(engl. Uniform Resource Locator) - Lokalisiert eine Ressource, z.B.: eine Webseite.

String Operatoren²²

Funktionen zur Manipulation von Zeichenketten.

Framework²³

Programmiergerüst, das in der Softwaretechnik verwendet wird.

(Präsentations-) Instanz²⁴

Subjekt einer spezifischen Präsentation, z.B. „Leipzig“.

Datenbank²⁵

Strukturierte Sammlung von Information; Daten sind systematisch katalogisiert und können gezielt abgerufen werden.

SPARQL²⁶

(engl. SPARQL Protocol and RDF Query Language) - Graphenbasierte Abfragesprache für RDF.

RDF²⁷

(engl. Resource Description Framework) - Bezeichnet eine technische Herangehensweise im Internet zur Formulierung logischer Aussagen über beliebige Dinge (Ressourcen).

Hyperlink²⁸

Querverweis, der funktional einen Sprung zu einem anderen elektronischen Dokument oder an eine andere Stelle innerhalb eines Dokuments ermöglicht.

Apache Jena²⁹

Java-Bibliothek zur Arbeit mit unter Anderem RDF und SPARQL.

pptx-Format³⁰

Dateiformat von Microsoft PowerPoint.

CLI³¹

(engl. command-line interface) - Eingabebereich zur Steuerung einer Software über Commands.

Algorithmus³²

Ein Algorithmus ist eine eindeutige Handlungsvorschrift zur Lösung eines Problems oder einer Klasse von Problemen.

DBpedia³³

Über Spenden finanziertes Gruppenprojekt, das strukturierte Informationen aus Wikipedia online zugänglich machen soll.

Proof-Of-Concept³⁴

Ein Meilenstein im Projektmanagement, der die prinzipielle Durchführbarkeit belegt.

Bug³⁵

Ein Fehler im Programm.

Mit dem Kunden vereinbarter Standard (formell)³⁶

Siehe Dokument Präsentationsstandard.

.zip-Datei³⁷

Format verlustfrei komprimierter Daten.