

# **Verwandtschaftsbeziehungen Projektdokumentation zum**

## **Softwareentwicklungsprojekt (Entwicklerdokumentation)**

Lehrveranstaltung „Software Engineering I / II“

Entwickler: Sebastian Mischke, Julian Fuchs, Anja Handrianz

Auftraggeber: Prof. Dr.-Ing. habil. Hartmut Fritzsche

Bachelorstudiengang Medieninformatik

Hochschule für Technik und Wirtschaft Dresden

June 17, 2016

### **Zusammenfassung**

Die Applikation bietet dem Nutzer die Möglichkeit Informationen über Personen sowie deren Verwandtschaftsbeziehungen über eine Eingabemaske einzutragen und abzuspeichern. Zudem können die eingetragenen Daten grafisch ausgegeben und als Bilder exportiert werden. Es soll jedem Menschen ermöglichen, einen persönlichen Überblick über seine Familie und deren Beziehungen zu bekommen. Ebenso soll es die Arbeit von Personen im Bereich der Ahnenforschung erleichtern.

# Contents

<b>1</b>	<b>Einleitung</b>	<b>3</b>
<b>2</b>	<b>Projektmanagement</b>	<b>3</b>
2.1	Vorgegebener Zeitablauf . . . . .	3
2.2	Ressourcenplanung und Organisation . . . . .	3
2.3	Werkzeugunterstützung . . . . .	4
2.3.1	Managementwerkzeuge . . . . .	4
2.3.2	Softwareentwicklungswerkzeuge . . . . .	4
<b>3</b>	<b>Pflichtenheft</b>	<b>4</b>
<b>4</b>	<b>Anforderungsanalyse und Entwurf</b>	<b>4</b>
4.1	Anwendungsfallanalyse . . . . .	4
4.2	Problembereichsanalyse . . . . .	6
4.3	Stand der Wissenschaft und Technik . . . . .	7
4.4	Entwurf der Systemarchitektur . . . . .	7
4.5	Entwurf der Benutzeroberfläche . . . . .	8
4.6	Entwurf der Funktionalität/Interaktionsmodell . . . . .	9
4.7	Datenverwaltung / Datenbankentwurf . . . . .	9
<b>5</b>	<b>Implementation</b>	<b>9</b>
5.1	Der Build-Prozess . . . . .	9
5.2	API-Dokumentation . . . . .	9
5.3	Teststrategien und -werkzeuge . . . . .	9
5.4	Testfallspezifikation . . . . .	9
5.5	Testdurchführung und Testergebnisse . . . . .	9
<b>6</b>	<b>Anwenderdokumentation</b>	<b>10</b>
<b>7</b>	<b>Projektbewertung aus Entwicklersicht</b>	<b>10</b>
	<b>References</b>	<b>10</b>

# 1 Einleitung

Ziel des Projektes ist es ein Programm zu entwickeln und eine Dokumentation darüber zu verfassen. Anhand dieser Dokumentation soll für den Leser verständlich werden wie die Software funktioniert und welche Funktionalitäten mit dem Programm möglich sind. Das vorhandene Programm heißt Verwandtschaftsbeziehungs-Generator (VBZ-Generator).

## 2 Projektmanagement

### 2.1 Vorgegebener Zeitablauf

Vorlage Pflichtenheft	12.05.2016
Bearbeitung Analyse/Entwurf:	
Klassendiagramme, Komponentendiagramme, Paketstruktur; Verteilungsdiagramme, ... Festlegung der Rollenverteilung	
Vorlage Projektdokumentation (Zwischenstand)	02.06.2016
Durchführung der Implementierung User Interface Prototyping	
Gruppenkolloquium	14.06.2016
Implementation, Testfallspezifikation, Testung (JUnit) Bereitstellung API (Javadoc)	
Abgabe der Projektdokumentation	21.06.2016
Vorbereitung Präsentation	
Praesentation	23.06.2016

### 2.2 Ressourcenplanung und Organisation

Das Projekt teilt sich zwischen 3 Personen auf. Die Zusammenarbeit aller 3 Beteiligten ist dabei erforderlich.

Unsere Aufteilung sieht die Analyse, Implementierung und Dokumentation vor. Zwei Personen (Anja Handrianz und Julian Fuchs) beschäftigen sich mit der gestalterischen und funktionalen Ausarbeitung, dem Pflichtenheft sowie der Entwicklerdokumentation. Eine Person (Sebastian Mischke) implementiert das Programm, was die funktionale Umsetzung der Benutzeroberfläche, die Auswahl einer technischen Herangehensweise und die Integration gestalterischer Vorarbeiten beinhaltet.

## **2.3 Werkzeugunterstützung**

### **2.3.1 Managementwerkzeuge**

Für das Projektmanagement wird die Subversion-Versionverwalten SVN in Topcased genutzt. Zur Erstellung der Dokumentation wird Lyx verwendet.

### **2.3.2 Softwareentwicklungswerkzeuge**

Zur Implementierung werden folgende Werkzeuge verwendet:

- Eclipse
- SVN in Topcased
- Java-Editor
- Sublime
- Php MyAdmin

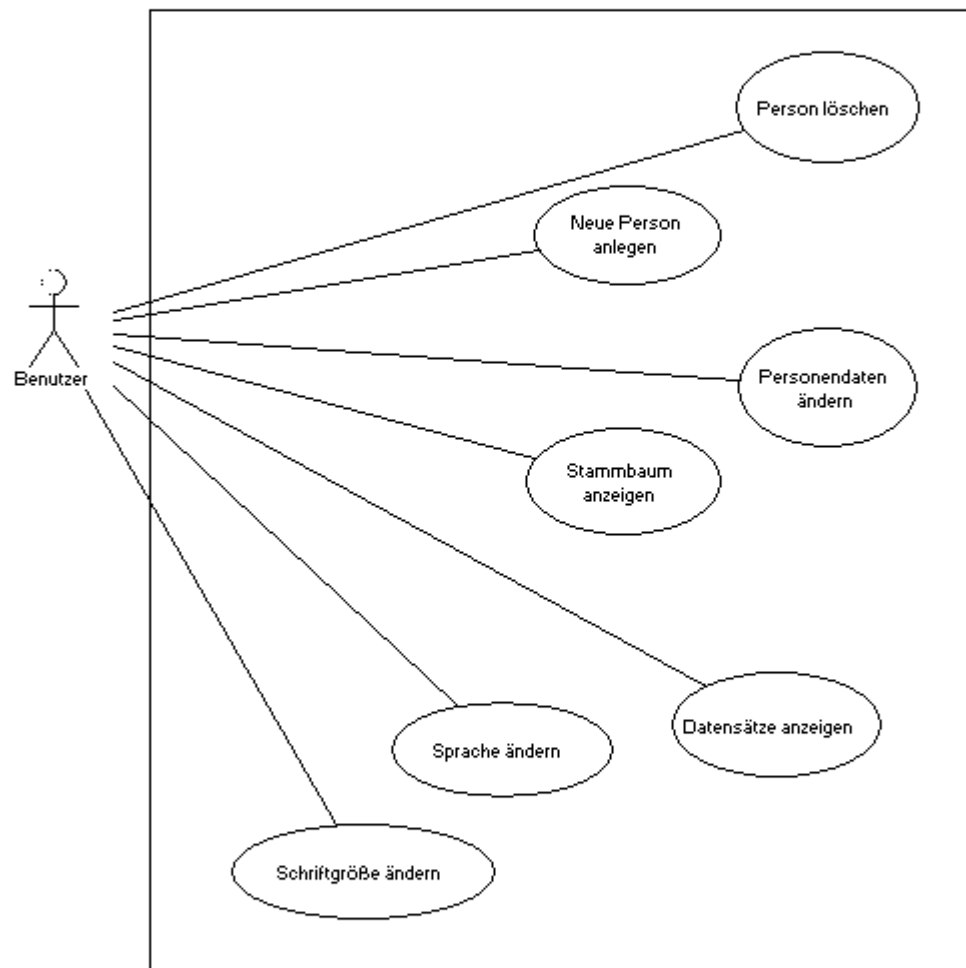
## **3 Pflichtenheft**

Das Pflichtenheft liegt in einem gesonderten Dokument vor.

## **4 Anforderungsanalyse und Entwurf**

### **4.1 Anwendungsfallanalyse**

Die Anwendungsoberfläche strukturiert sich in drei Teile: die Menüleiste, den Editor - mit seinen Benutzerelementen und der Bearbeitungsoberfläche - und die Anzeige - mit deren Anzeigeelementen und Grafiken. In der Menüleiste sind alle Optionen für das Ändern, Speichern und Laden der Anzeige realisiert.



### Use-Case-Diagramm

Dem Anwendungsnutzer steht frei die Schriftgröße und Sprache an seine Bedürfnisse anzupassen. Er hat die Möglichkeit einen Stammbaum anzulegen, zu bearbeiten oder zu löschen, indem er Personen hinzufügt, löscht oder Personendaten ändert. Der Nutzer erhält eine Ausgabe in Form eines Graphen oder kann sich die Datensätze eines Stammbaums als Tabelle ausgeben lassen.

Anwendungsbeispiel:

1. Start des Programms (siehe 6. Anwenderdokumentation)
2. Anwählen des Einstellungen-Icons
3. Festlegen der Sprache und der Textgröße
4. Mit dem Button “Übernehmen” bestätigen
5. Anwählen des Datenbank-Icons
6. Eine vorhandenen Stammbaum auswählen und mit dem Button “laden” bestätigen
7. Anwählen des Personen-Icons
8. Über die Eingabemaske eine Person zum Stammbaum hinzufügen und mit dem Button “Übernehmen” bestätigen
9. Anwählen des Tabellen-Icons
10. Stammbaum über den Button “speichern” unter einem beliebigen Dateinamen abspeichern
11. Beenden des Programms

## 4.2 Problembereichsanalyse

Unser Programm besteht aus einem JFrame, welcher 5 JPanels beinhaltet (Einstellungen, Graph, Person-Editor, Tabelle, Stammbaum-Archiv). Um einen Stammbaum anzeigen zu lassen, muss der Zugriff und das Auslesen aus einer Datenbank möglich sein. Dafür wurde bei dem Freehoster XY ein Account angelegt und anschließend die Datenbank erzeugt. Nun muss eine Funktion zum Anlegen eines Stammbaums implementiert werden. Für jeden angelegten Stammbaum wird eine Tabelle erzeugt, die den Namen des Stammbaums trägt. Diese Tabelle kann nun mit Personendaten gefüllt werden (Nachname, Vorname, Geburtsdatum, Geschlecht, Vater, Mutter, Ehepartner). Hierfür wird eine Klasse “Person” geschrieben. In dem JPanel “Personen-Editor” befindet sich eine Eingabemaske in der man die Personendaten einträgt. Nun kann der Stammbaum als Graph oder Tabelle ausgegeben werden, die dem Inhalt der angelegten Tabelle in der Datenbank entsprechen. *Um als Graph anzuzeigen*

wird... *WIE WIRD DAS UMGESETZT?*. Unter dem JPanel „Stammbaum-Archiv“ kann man einen bereits angelegten Stammbaum aus einer Übersicht auswählen, um diesen zu löschen, anzuzeigen oder zu editieren. Dafür ist eine weitere Klasse nötig, welche ... *verwaltet (PersonList)???*. In dem JPanel „Einstellungen“ kann das Programm der Individuellen Bedürfnisse des Nutzers angepasst werden. Es sind drei Einstellungsgrößen für die Schriftgröße wählbar und die Sprache kann zwischen Deutsch und Englisch umgestellt werden.

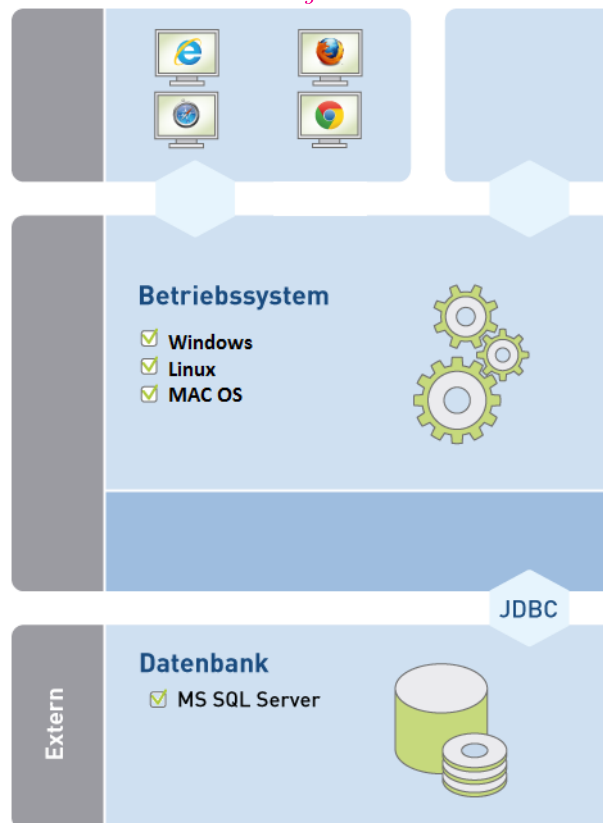
### 4.3 Stand der Wissenschaft und Technik

Die Nachnutzung des Programms

*Überlegungen zur Abwägung zwischen Nachnutzung und Eigenentwicklung. Dieser Punkt kann recht umfangreich sein, wenn vorhandene Lösungen erst analysiert werden müssen (z.B. CMS\_Projekt).*

### 4.4 Entwurf der Systemarchitektur

*Grobarchitektur, z.B. Darstellung eines Schichtenmodells. Komponentenbildung. Nachnutzung von Bibliotheken/Frameworks. Verteilungsdiagramm(e) im Falle von Client-Server-Anwendungen.*



## 4.5 Entwurf der Benutzeroberfläche



## 4.6 Entwurf der Funktionalität/Interaktionsmodell

*Beschreibung der logischen Struktur der Anwendung, Klassendiagramme, Zustandsdiagramme, Schnittstellenspezifikationen (Interfaces!) und verwendete bzw. selbst definierte Protokolle.*

## 4.7 Datenverwaltung / Datenbankentwurf

*Beschreibung der persistenten Datenspeicherung. Angabe von ER-Diagrammen und des OR-Mappings.*

Für eine persistente Datenhaltung werden die Stammbäume in einer MS SQL Datenbank gespeichert.



## 5 Implementation

*Abgrenzung von Geschäftsklassen und Fachklassen. Erstellung bzw. Generierung der Quellcodes.*

*Implementierung von Algorithmen.*

### 5.1 Der Build-Prozess

*Beschreibung der Erstellung der auszuliefernden Anwendung. Enthält auch eine Installationsanleitung.*

### 5.2 API-Dokumentation

*Beschreibung der Erstellung der API-Dokumentation (Javadoc).*

### 5.3 Teststrategien und -werkzeuge

*Hier wird beschrieben, ob und wenn ja welche Testwerkzeuge eingesetzt werden (z.B. JUnit).*

### 5.4 Testfallspezifikation

*Beschreibung der JUnit-Testfälle. Insbesondere sollen Überprüfungen mit Assert-Prädikaten dargestellt werden.*

### 5.5 Testdurchführung und Testergebnisse

*Beschreibung der Ergebnisse der mit JUnit spezifizierten und ausgeführten Testfälle. Zusätzlich ist ein Testbericht zu erstellen.*

## 6 Anwenderdokumentation

Admin:

Kompilierung unter Windows/Linux/Mac: Zuerst muss sichergestellt werden, dass das Java SDK auf den Computer installiert ist und auch lauffähig ist. Zur Kompilierung ist die Umgebungsvariable zu setzen (im selben Pfad wo auch Java SDK installiert ist). Als nächstes muss die Eingabeaufforderung z.B. mit dem Befehl „cmd“ geöffnet werden. Im Folgenden muss man in dem Pfad navigieren wo sich die Projektdateien befinden. Dort ist es notwendig zunächst alle Klassen zu löschen. (Befehl: `del /S *.class`) Nun müssen die Klassen neu erstellt werden damit Änderungen übernommen werden. (Befehl: `javac ....`) Jetzt kann das Projekt ausgeführt werden. (Befehl: `java ....`) WIE AUF RECHNER INSTALLIEREN FÜR NUTZER?

Nutzer:

Auf desktop das Programm „VBZ-Generator“ mit einem Doppelklick starten. Das Programm wird ausgeführt. Eine Anmeldung ist nicht notwendig.

## 7 Projektbewertung aus Entwicklersicht

Am Ende des Projektes ist es uns erfolgreich gelungen ein Programm zur Generierung von einem oder mehrerer Stammbäume zu erstellen. Dabei konnten wir alle uns aufgefallenen lösen.

Zusätzlich ist die Umstellung der Sprache und Schriftgröße sowie das Anlegen eines Hilfemenüs gelungen. WEITERE SOLL MUSS KANN KRITERIEN AUFLISTEN??

Wenn mehr Zeit zur Verfügung stünde, könnten zusätzliche Funktionalitäten wie z.B. eine Nutzerverwaltung mit Login und das Teilen von Stammbäumen mit anderen Nutzern, implementiert werden.

Mit entsprechendem Budget könnte man die SQL-Datenbank bei einem „??nicht free hoster??“ anlegen.

Die Gestaltung der Entwicklungsumgebung (Pflichtenheft, Dokumentation) hat uns geholfen an Dinge zu denken die uns sonst nicht aufgefallen wären ??????.

## References