
Praktikumsanmeldesoftware UniKit

Inhaltsverzeichnis

- [1. Projekt-Teilnehmer](#)
 - [2. Projektbeschreibung](#)
 - [3. Projektplan](#)
 - [4. Vorgehensmodell](#)
 - [5. Prototyp](#)
 - [6. Statusbericht](#)
 - [6.1. Gesamtprojektzustand](#)
 - [6.2. Auslieferungsgegenstände](#)
 - [6.3. Qualitätsdarstellung: Probleme](#)
 - [6.4. Qualitätsdarstellung: Risiken](#)
-

1. Projekt-Teilnehmer

| Name | TelNr | E-Mail | Spezielle Vorkenntnisse | Funktion |
|-----------------|-------------------|--------------------------------|-------------------------|---|
| Thomas Bednorz | 0175 / 896 54 28 | thomas.bednorz@haw-hamburg.de | | - Chefdesigner (fachlich) - Backend-Entwickler |
| Andreas Berks | 0176 / 808 16 437 | andreas.berks@haw-hamburg.de | | - Projektleiter - Infrastrukturverantwortlicher - Frontend-Designer |
| Jonas Johannsen | 0176 / 477 85 997 | jonas.johannsen@haw-hamburg.de | | - Chefdesigner (technisch) - Backend-Entwickler |
| Jana Wengenroth | 0176 / 832 02 135 | jana.wengenroth@haw-hamburg.de | | - Qualitätsbeauftragte - Backend-Entwicklerin |

- **Andreas** ist Projektleiter, Infrastrukturverantwortlicher und Frontend-Designer. Als **Projektleiter** übernimmt er die Organisation des Projektes und verteilt Aufgaben an die einzelnen Teammitglieder. Als **Infrastrukturverantwortlicher** ist er dafür zuständig, dass sowohl Datenbank (inkl. Anbindung), Versionsverwaltung als auch Server ordnungsgemäß funktionieren. Er arbeitet zurzeit auch in der Entwicklung des Backends mit. Der Grund dafür ist bei den Problemen näher erläutert. In Phase 2 wird er das **Frontend** der Anwendung designen.
- **Jana** ist Qualitätsbeauftragte und außerdem Backend-Entwicklerin. Als **Qualitätsbeauftragte** ist sie dafür zuständig, dass sich alle an die Code-Konventionen halten. In Zukunft wird sie auch verstärkt dafür sorgen, dass die Software ausreichend durch die Entwickler getestet wird. Außerdem hält sie zurzeit die Dokumentation des Projektes auf dem Laufenden. In Phase 2 wird sie an dem **Backend** der Anwendung entwickeln.

- **Thomas** ist Chefdesigner (fachlich) und einer unserer Backend-Entwickler. Als **fachlicher Chefdesigner** organisiert er die konzeptionelle Entwicklung des Projektes. Er hat im Laufe unseres ersten Entwicklungszyklus die **Logik der Anwendung** für Phase 1 (siehe Projektbeschreibung) und ein rudimentäres Frontend für diese entwickelt.
- **Jonas** ist Chefdesigner (technisch) und einer unserer Backend-Entwickler. Als **technischer Chefdesigner** organisiert er die technische Entwicklung des Projektes (z.B. Programmarchitektur). Er ist für den **Algorithmus** der Praktikumsgruppenbelegung nach Phase 2 (siehe Projektbeschreibung) zuständig.

2. Projektbeschreibung

UniKit ermöglicht es den Studenten sich mit einem Team bestehend aus zwei bis 10 Studenten für ein Praktikum anmelden zu können. Dies erfolgt in zwei Phasen. In Phase Eins melden sich die Studenten für das Praktikum an. So wissen wir für den späteren Algorithmus schon einmal, ob noch weitere Praktikumsgruppen benötigt werden und können die Uni dann dahingehend informieren. Jetzt sind bereits alle Studenten für das Praktikum angemeldet. In Phase Zwei haben die Studenten dann die Möglichkeit sich in einem Team zusammen zu finden. Am Ende der Phase Zwei springt dann unser Algorithmus an und versucht möglichst alle Teams konfliktfrei auf die Praktikumsgruppen zu verteilen. Ist ein Student keinem Team beigetreten, so macht das nichts und er wird dennoch einer Praktikumsgruppe zugeteilt. Als Ergebnis liefert UniKit dann einen Belegungsplan für alle Praktika und den dazugehörigen Praktikumsgruppen.

3. Projektplan

- 1) Einrichtung der Systemumgebung
- 2) Planung (und Einarbeitung)
- 3) Entwicklung von Phase 1
- 4) Entwicklung von Phase 2 (Einzelanmeldungen)
- 5) Entwicklung Phase 2 (Gruppenanmeldungen)
- 6) Entwicklung des Verteilungsalgorithmus
- 7) Qualitätssicherung und Projektabschluss
- 8) Präsentation der Ergebnisse

| August | September | Oktober | November | Dezember | Januar |
|---------------------|---------------------|---------------------|---------------------|---------------------|--------------------|
| 1 Sa | 1 Di | 1 Do | 1 So | 1 Di | 1 Fr |
| 2 So | 2 Mi | 2 Fr | 2 Mo ⁴⁵ | 2 Mi | 2 Sa |
| 3 Mo ³² | 3 Do | 3 Sa | 3 Di | 3 Do | 3 So |
| 4 Di | 4 Fr | 4 So | 4 Mi | 4 Fr | 4 Mo ¹ |
| 5 Mi | 5 Sa | 5 Mo ⁴¹ | 5 Do | 5 Sa | 5 Di |
| 6 Do | 6 So | 6 Di | 6 Fr | 6 So | 6 Mi |
| 7 Fr | 7 Mo ³⁷ | 7 Mi | 7 Sa | 7 Mo ⁵⁰ | 7 Do |
| 8 Sa | 8 Di | 8 Do | 8 So | 8 Di | 8 Fr |
| 9 So | 9 Mi | 9 Fr | 9 Mo ⁴⁶ | 9 Mi | 9 Sa |
| 10 Mo ³³ | 10 Do | 10 Sa | 10 Di | 10 Do | 10 So |
| 11 Di | 11 Fr | 11 So | 11 Mi | 11 Fr | 11 Mo ² |
| 12 Mi | 12 Sa | 12 Mo ⁴² | 12 Do | 12 Sa | 12 Di |
| 13 Do | 13 So | 13 Di | 13 Fr | 13 So | 13 Mi |
| 14 Fr | 14 Mo ³⁸ | 14 Mi | 14 Sa | 14 Mo ⁵¹ | 14 Do |
| 15 Sa | 15 Di | 15 Do | 15 So | 15 Di | 15 Fr |
| 16 So | 16 Mi | 16 Fr | 16 Mo ⁴⁷ | 16 Mi | 16 Sa |
| 17 Mo ³⁴ | 17 Do | 17 Sa | 17 Di | 17 Do | 17 So |
| 18 Di | 18 Fr | 18 So | 18 Mi | 18 Fr | 18 Mo ³ |
| 19 Mi | 19 Sa | 19 Mo ⁴³ | 19 Do | 19 Sa | 19 Di |
| 20 Do | 20 So | 20 Di | 20 Fr | 20 So | 20 Mi |
| 21 Fr | 21 Mo ³⁹ | 21 Mi | 21 Sa | 21 Mo ⁵² | 21 Do |
| 22 Sa | 22 Di | 22 Do | 22 So | 22 Di | 22 Fr |
| 23 So | 23 Mi | 23 Fr | 23 Mo ⁴⁸ | 23 Mi | 23 Sa |
| 24 Mo ³⁵ | 24 Do | 24 Sa | 24 Di | 24 Do | 24 So |
| 25 Di | 25 Fr | 25 So | 25 Mi | 25 Fr | 25 Mo ⁴ |
| 26 Mi | 26 Sa | 26 Mo ⁴⁴ | 26 Do | 26 Sa | 26 Di |
| 27 Do | 27 So | 27 Di | 27 Fr | 27 So | 27 Mi |
| 28 Fr | 28 Mo ⁴⁰ | 28 Mi | 28 Sa | 28 Mo ⁵³ | 28 Do |
| 29 Sa | 29 Di | 29 Do | 29 So | 29 Di | 29 Fr |
| 30 So | 30 Mi | 30 Fr | 30 Mo ⁴⁹ | 30 Mi | 30 Sa |
| 31 Mo ³⁶ | | 31 Sa | | 31 Do | 31 So |

4. Vorgehensmodell

Das Vorgehen zur Projektumsetzung ist iterativ angelegt mit Elementen, die sich an „Scrum“ anlehnen. Es gibt regelmäßige wöchentliche Treffen, in denen der bisherige Stand, das weitere Vorgehen und Schwierigkeiten besprochen werden. In jedem Treffen werden die nächsten Aufgaben erfasst und auf die Teammitglieder verteilt.

Die gesetzten Meilensteine sind so angelegt, dass immer ein lauffähiges Teilergebnis erreicht

wird, das durch die darauffolgenden Meilensteine erweitert werden soll. Es gibt viele kleinere Zwischenschritte, die zum Erreichen des Meilensteins notwendig sind. Die als nächstes folgenden Schritte werden aus dem aktuellen Stand und der zeitlichen Notwendigkeit ihrer Erfüllung ermittelt, an die Teammitglieder verteilt und mit einer Deadline versehen.

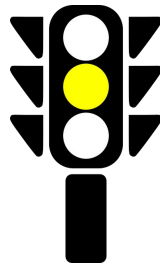
5. Prototyp

- Ein vollständiger, jedoch auf **statischen** Webseiten basierender **Prototyp** ist unter folgender Adresse erreichbar: <http://h2448947.stratoserver.net/prototype/>
 - Außerdem steht ein **Download** unter folgender Adresse bereit: <https://github.com/HAW-BAI4-SE2/UniKit/raw/master/exports/prototype.zip>
- Die **erste Beta-Version** unserer Anwendung (Ergebnis der Entwicklung von Phase 1) ist voraussichtlich ab den 20.10.2015 um 8:00 Uhr unter folgender Adresse erreichbar: <http://h2448947.stratoserver.net:9000/>
 - Außerdem steht bei Veröffentlichung ein **Download** unter folgender Adresse bereit: <https://github.com/HAW-BAI4-SE2/UniKit/raw/master/exports/unikit-0.8.2.zip>
- Zur **Verwaltung der Datenbank** haben wir ein eigenständiges Modul geschrieben, welches unter folgender Adresse zum **Download** bereitsteht: <https://github.com/HAW-BAI4-SE2/UniKit/raw/master/exports/haw-hamburg-0.3.2.jar>
 - Weiterhin steht unter folgender Adresse eine JavaDoc-Dokumentation bereit: <http://haw-bai4-se2.github.io/UniKit/haw-hamburg/current/javadoc/index.html>

6. Statusbericht

- **Projektname:** UniKit
- **Aktuelle Version:** 0.8.1 (vom 18.10.2015)
- **Nächste Version:** 0.8.2-beta (voraussichtlich am 19.10.2015)

6.1. Gesamtprojektzustand



6.2. Auslieferungsgegenstände

- **Veröffentlichungen** (siehe Kapitel Prototyp)
 - **Statischer Prototyp:**
Fertiggestellt
 - **Datenbankverwaltung (Version 0.3.2):**
Fertiggestellt und getestet, jedoch nicht vollständig dokumentiert!
Außerdem sind die Methoden für Phase 2 noch nicht implementiert!
 - **Erste Beta-Version der Software (Version 0.8.2-beta):**
Vorraussichtlich am 19.10.2015 fertiggestellt, jedoch noch nicht vollständig getestet und dokumentiert!
- **Datenbankmodelle**
 - **ER-Diagramm für interne Daten** (Anhänge/unikit_database_diagram.pdf):
Fertiggestellt
 - **ER-Diagramm für externe Testdaten** (Anhänge/import_database_diagram.pdf):
Fertiggestellt
- **Datenbankskripte**
 - **Datenbankskripte für interne Daten** (Anhänge/unikit_database_scripts.zip):
Fertiggestellt
 - **Datenbankskripte für externe Daten** (Anhänge/import_database_scripts.zip):
Fertiggestellt
- **Komponentendiagramm** (Anhänge/component_diagram.pdf):
Nicht auf dem neusten Stand! Interne Repräsentation in unserer Software weicht inzwischen stark von dem Diagramm ab!

6.3. Qualitätsdarstellung: Probleme

Wir haben derzeit noch einige Anlaufschwierigkeiten. Die Versionsverwaltung mit „GitHub“ bietet zwar sehr schöne Möglichkeiten, wird zu Weilen aber auch recht unübersichtlich. Wir wollten GitHub mit seinem Ticketsystem zur primären Kommunikation außerhalb unserer Teammeetings nutzen, was sich jedoch auf Grund der Masse an Kommentaren als sehr schwierig darstellt. In Zukunft werden wir verstärkt auf Telefonkonferenzen zurückgreifen, sollten wichtige Dinge zu besprechen sein.

Wir haben die Teilaufgaben in einzelne „Branches“ aufgeteilt. Beim Versuch, die Teillösungen im „Master-Branch“ zusammen zu führen, gab es allerdings diverse Konflikte: Die Datenbank hat sich auf Grund ungenügender Planung in den letzten Tagen nochmal stark verändert, was zur Auswirkung hatte, dass diverse Änderungen in den Schnittstellen gemacht werden mussten und entsprechende Verzögerungen eingetreten sind.

Obwohl wir uns wöchentlich treffen, bei den Treffen ein Protokoll führen, planen und Aufgaben verteilen, gestaltet sich die Organisation und Kommunikation des Teams deutlich schwerer als erwartet und wir werden in Zukunft noch strukturierter vorgehen müssen. Es war zwar auch ein Problem für uns, dass Christopher Wolter kurzfristig das Projekt verlassen musste und wir dadurch umstrukturieren mussten, mit der aktuellen Vorgehensweise haben wir es aber bisher nicht geschafft, auf Problemsituationen im Allgemeinen sinnvoll und strukturiert zu reagieren.

Das Team ist aber motiviert und es gibt bis auf die noch nicht ausgereifte Strukturierung, keine zwischenmenschlichen Verwerfungen. Das hier noch Verbesserungsbedarf besteht ist auch allgemeiner Konsens.

6.4. Qualitätsdarstellung: Risiken

Auf Grund von unzureichenden Tests, ist es möglich, dass sich Fehler in unseren Programmcode eingeschlichen haben. Um diesem Risiko entgegen zu wirken haben wir bereits einen großen Teil der Software überarbeitet und auf diesen neuen Modulen Tests während der Entwicklung durchgeführt (durch umfangreiche Debug-Ausgaben). Dennoch werden wir von nun an verstärkt Unit-Tests einsetzen um die Konsistenz der Anwendung auch nach der Entwicklung aufrecht zu erhalten bzw. zu gewährleisten.

Durch unvollständige JavaDoc-Dokumentation kann es außerdem zur Verständnisschwierigkeiten in der Implementation der Software kommen. Wir werden diese Dokumentation in den kommenden Tagen nachrüsten.