**Pflichtenheft**

**Kurs:** TIF20

**Gruppe/Team:** Böhler Anton, Cosovan Andrei, Dietrich Alexander, Hügle Christian, Schüler David, Broer Jan (Gruppe4)

Inhalt

[1. Einleitung 1](#_Toc72502094)

[2. Technische Lösung 2](#_Toc72502095)

[2.1 Konzeption 2](#_Toc72502096)

[2.2 Entwicklung 2](#_Toc72502097)

[2.3 Projektmanagement 2](#_Toc72502098)

[3. Beschreibung der Anforderungen bzw. Komponenten 2](#_Toc72502099)

[4. Bereitstellung der Mockups und der Webanwendung 4](#_Toc72502100)

[4.1 Auslieferung 4](#_Toc72502101)

[5. Qualitätsmanagement und Tests 5](#_Toc72502102)

[5.1 Anforderungsprüfung 5](#_Toc72502103)

[5.2 Intuitivitäts- und Funktionalitätsprüfung 5](#_Toc72502104)

[5.3 Sicherheit 5](#_Toc72502105)

[6. Projektplanung 5](#_Toc72502106)

[6.1 Zeitschätzung / Meilensteine 5](#_Toc72502107)

[6.1.1 Planungsphase 5](#_Toc72502108)

[6.1.2 Durchführungsphase 5](#_Toc72502109)

[6.1.3 Prüfungsphase 6](#_Toc72502110)

[6.1.4 Übergabephase 6](#_Toc72502111)

[6.2 Rollen 6](#_Toc72502112)

[7. Anhang 6](#_Toc72502113)

[7.1 Hinweise 6](#_Toc72502114)

# Einleitung

Ausgangssituation:

Das Sekretariat des SZI der DHBW-Lörrach ist damit beauftragt, alle Vorlesungen und Klausuren der ihnen zugehörigen Kurse zu planen. Diese Planung beginnt spätestens drei Monate vor Beginn der Theoriephase der Kurse und ist spätestens einen Monat vor Vorlesungsbeginn abgeschlossen. Bislang erfolgt die gesamte Planung in einem Tischkalender, in dem die Termine mit Bleistift eingetragen werden. Die Kommunikation mit den Dozierenden erfolgt per E-Mail und/oder Telefon.

Ziel:

Um die Vorlesungsplanung in Zukunft zu erleichtern, soll dieser Prozess in eine Webanwendung übertragen werden. Dafür sollen Wireframes und Mockups einer Webanwendung entwickelt und im besten Fall ein Prototyp der Anwendung bereitgestellt werden.

# Technische Lösung

## 2.1 Konzeption

Wireframes werden durch das Tool Excalidraw erstellt. Durch die Verfügbarkeit über den Web-Browser können die Wireframes gemeinsam besprochen und erstellt werden. Ebenfalls ist so gewährleistet, dass die Wireframes von allen Beteiligten, unabhängig von ihrem gewählten System, gelesen und bearbeitet werden können.

In den Wireframes werden die einzelnen Webseiten bzw. Ansichten mit Beispieldaten dargestellt. Pfeile und Beschreibungen sollen die Funktionalität und Abläufe verdeutlichen. Dadurch kann man sich später bei der Entwicklung an diesen orientieren und zusätzlich bedeutende Edge-Cases erkennen. Die deutliche Unterteilung der Webseiten-Abschnitte ermöglicht zusätzlich, dass man sich den einzelnen Abschnitten in der Entwicklung, unabhängig voneinander, widmen kann.

Zusätzlich soll durch das Mockup-Tool „Figma“ veranschaulicht werden, wie die finale Webanwendung designtechnisch aussehen soll. Diese stellt jedoch lediglich unsere Konzeption zum Design dar und wird für die Umsetzung des ersten Web-Prototyps, in den Aspekten des Designs, nicht eins zu eins umgesetzt.

## 2.2 Entwicklung

Für die Entwicklung des Tools haben wir uns für einen klassischen Webentwicklungsansatz entschieden. Dies bedeutet, dass ein Front- und Backend entwickelt werden soll, welche zusammen die Anwendung bilden. Dabei wird NodeJS als Backendlösung verwendet.

Der Server wird Anfragen von den Benutzern, wie Webseiten-Aufrufe, Daten-Anfragen (Termine) und Änderungs-Anfragen annehmen und bearbeiten. Dazu wird das Web-Application-Framework Express verwendet. Zusätzlich soll der Server die Termine, Kurse und Benutzer (Admin, Dozent & Student) verwalten. Bei der Datenspeicherung wird keine Datenbank verwendet, sondern lediglich strukturiert abgespeicherte JSON-Dateien, um den Umfang des Projekts gering zu halten.

Für die Darstellung des Frontends werden für jede Seite eine HTML-, CSS- und JavaScript-Datei vom Backend ausgeliefert. Diese werden strukturiert in Ordnern abgespeichert. Im Frontend werden keine Frameworks verwendet. Um die Redundanz im Code geringer zu halten, soll es generische Style-Sheets geben, welche für alle Webseiten verwendet werden.

Dieser Ansatz der Webentwicklung wurde gewählt, da das nötige Knowhow schon vorhanden ist und deshalb eine schnellere Umsetzung zu erwarten ist. Für die Verwaltung des Codes und die Zusammenarbeit wird ein Github-Repository verwendet.

## 2.3 Projektmanagement

Als Projektmanagement-Tool wird Trello verwendet. In diesem werden alle Aufgaben und Termine der einzelnen Beteiligten verwaltet. Trello wurde gewählt, da das Tool sehr übersichtlich ist und sich intuitiv bedienen lässt. Für die Erstellung des Projektstrukturplans wird yEd Graph Editor von yWorks eingesetzt, da dieses auch schon in anderen Projekten zum Einsatz kam und sich als geeignet erwiesen hat. Für die Darstellung des Projektes in einem Zeit-/Ressourcenplan wurde sich für ein GANTT-Diagramm entschieden, dieses wird in der Software ProjectLibre erstellt. MS-Office 365 (Word, Excel) wird für die Erstellung und Bearbeitung des Projektsteckbriefs und des Risikoregisters verwendet.

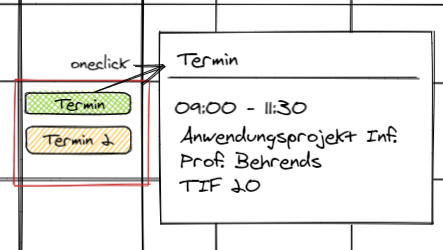
# Beschreibung der Anforderungen bzw. Komponenten

**MUST-HAVEs**

* Vorlesungszeitraum (Theoriephase, Quartal) zur Planung festlegen (z.B. 29.03.2021-18.06.2021)
* Kurs angeben oder auswählen für den geplant werden soll



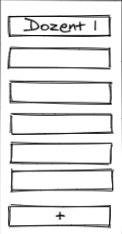
* Zu planende Vorlesungen mit Namen und Stundenzahl eingeben, ggf. mit Klausur
* Termine eintragen, bearbeiten und löschen (nur innerhalb des Zeitraums/Quartals)
* Termine erscheinen in einer Wochen- und/oder Monats- und/oder Kalenderansicht
* Termine haben mindestens eine Start- und Endzeit, einen Vorlesungstitel und den Namen eines Dozierenden



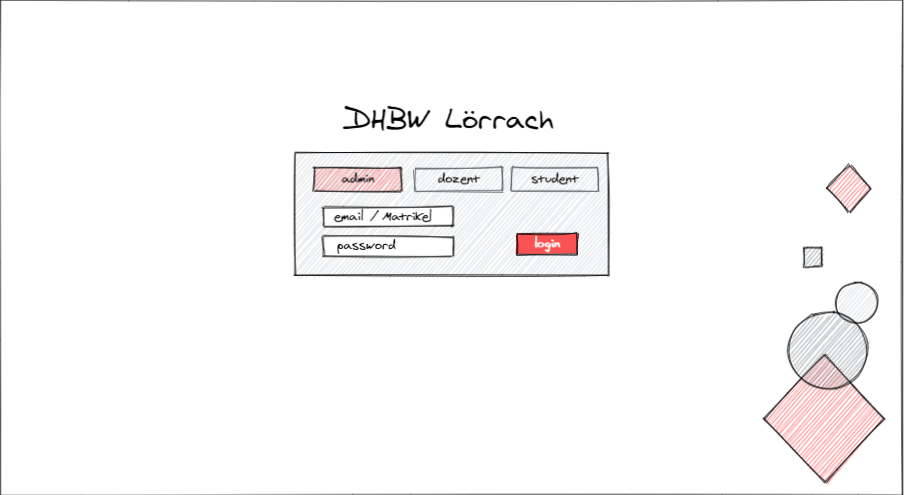
* Vorlesungen beginnen nicht vor 08:00 Uhr und enden spätestens um 17:00 Uhr
* Mindestens eine Stunde Mittagspause zwischen 11:30 und 14:00 Uhr (Mensazeiten?)
* Es ist bei den Terminen erkennbar, ob die Anzahl der Termine die Vorlesungsstunden einer Vorlesung ergibt („Sind alle Termine eingeplant?“)
* Es ist bei den Terminen ersichtlich, ob sie bestätigt wurden oder noch nicht („Kann der Vorlesungsplan veröffentlicht werden?“)

**NICE-TO-HAVEs**

* Planung für mehrere Kurse
* auch Dozierende könnte das Tool nutzen, z.B. für „Wunschtermine“
* Alle Vorlesungen eines Dozierenden über mehrere Kurse hinweg anzeigen (Quartal, Historie, usw.)



* Optimale Vorlesungszeiten sind 9:00 - 12:15 Uhr und 13:00 - 16:15 Uhr
* Anmeldung mit Benutzername und Passwort



* Prozesse zur Benachrichtigung, Erinnerung und Bestätigung von Terminen
* Auswahl von Räumen für Vorlesungen (insbesondere Labore, Rechnerräume)
* Dozierendenmanagement (z.B. bevorzugte Vorlesungen, Termine, usw.)
* Export als PDF zum Drucken bzw. per E-Mail verschicken
* Usability und Vereinfachungen: Vorlesungen beginnen/enden immer XX:00, XX:15, XX:30, XX:45, Autovervollständigung, Suche, usw.

# Bereitstellung der Mockups und der Webanwendung

## 4.1 Auslieferung

Als Ergebnis wird eine .zip-Datei geliefert. In dieser ist das Github-Repository sowie ein Link zu diesem enthalten. Zudem gibt es einen Link zu den Mockups von „Figma“, welche im Laufe der Entwicklung genutzt wurden. In dem Repository sind neben der Anwendung (Backend & Frontend), welche es einem ermöglicht das fertige Produkt lokal auszuführen oder es auf einem Server zu deployen, zusätzlich eine Dokumentation/Kurzanleitung vorhanden. Diese enthält eine Anleitung mit Bildern und Beschreibungen, der einzelnen Abschnitte der verschiedenen Webseiten.

# Qualitätsmanagement und Tests

Alle Tests werden in der hier dargestellten Reihenfolge durchgeführt. Sollten Qualität, bzw. Sicherheitsmängel auftreten, werden diese nach Entdeckung sofort vor dem nächsten Schritt behoben. Dies stört zwar die Parallelität der Arbeitspakete, ist aber notwendig, damit nicht auf Basis eines Fehlers weiterentwickelt wird.

## 5.1 Anforderungsprüfung

Nach der Erstellung eines ersten Prototyps, wird dieser auf die Anforderungen aus dem Pflichtenheft geprüft. Diese Überprüfung wird firmenintern in einem Meeting vorgenommen. Bei der Prüfung soll sichergestellt werden, dass alle Wünsche des Kunden berücksichtigt werden. Dabei soll das Pflichtenheft als Checkliste dienen, wobei die Must-haves natürlich oberste Priorität haben.

## 5.2 Intuitivitäts- und Funktionalitätsprüfung

Sobald die Anforderungsprüfung abgeschlossen ist, soll eine „Intuitivitätsprüfung“ und gleichzeitig ein Funktionstest durchgeführt werden. Dabei wird dem Sekretariat des SZI ein Prototyp mit Anleitung bereitgestellt. Dieser soll von den Sekretärinnen getestet werden. Nach dem Test soll ein Feedback zur Benutzung des Tools und Verständlichkeit der Anleitung eingeholt werden. Mit diesem Feedback kann die Anwendung und Anleitung gegebenenfalls optimiert werden. Dieser Test kann bei Bedarf mehrfach durchgeführt werden, um die Anwendung so angenehm wie möglich zu gestalten.

## 5.3 Sicherheit

Im letzten Test soll die Sicherheit des Tools geprüft werden. Dies soll durch die externe Firma „Backdoor-Seekers“ übernommen werden. Diese sind auf Angriffe öffentlicher Einrichtungen spezialisiert. Die Firma wird beauftragt, einen „Pentest“ an dem Tool durchzuführen. Dabei wird das Tool, wie im Produktivbetrieb, aus dem Internet erreichbar sein. Dieser Test wird lediglich mit Dummy-Daten durchgeführt.

In einem späteren Meeting mit der Firma werden diese Verbesserungsvorschläge und mögliche Fehlerquellen aufgezeigt, welche danach gegebenenfalls verbessert werden.

# Projektplanung

## 6.1 Zeitschätzung / Meilensteine

Das Projekt soll in vier Phasen unterteilt werden. Jede dieser Phasen baut aufeinander auf. Es können also keine Arbeitspakete aus späteren Phasen vorgezogen werden.

## 6.1.1 Planungsphase

In der ersten Phase soll das Tool geplant werden, dafür sollen Wireframes gemeinsam in Meetings erstellt werden. Ebenfalls sollen Arbeitspakete festgelegt werden und auf die verschiedenen Rollen und Personen, je nach Qualifikation, aufgeteilt werden. Die Planungsphase soll allen Beteiligten die Problemstellung und geplante Lösung klar machen, wodurch es später zu weniger Komplikationen kommt. Geschätzte Dauer ca. 9 Manntage.

## 6.1.2 Durchführungsphase

In Phase Zwei wird das Tool entwickelt. Dies beinhaltet das Front- und Backend, sowie die Verbindung der beiden Komponenten. Das Frontend und Backend wird anhand des Plans der Planungsphase parallel entwickelt.

Jedes Teammitglied bekommt mindestens ein Arbeitspaket des Frontend zugeteilt. Grundlage hierfür ist der Projektstrukturplan. Das Backend wird exklusiv von einer Teilgruppe implementiert. Nach der Durchführung eines Arbeitspakets soll eine kurze Besprechung erfolgen, um das Feedback des Teams einzuholen. Geschätzte Dauer ca. 16 Manntage.

## 6.1.3 Prüfungsphase

In der dritten Phase wird die Qualität überprüft, sowie das Tool selbst getestet. Wie diese Tests durchgeführt werden sollen, wird im Abschnitt „Qualitätsmanagement und Tests“ beschrieben. Geschätzte Dauer ca. 7 Manntage.

## 6.1.4 Übergabephase

In der letzten Phase wird das Ergebnis, wie in 4.1 „Auslieferung“ beschrieben, an das SZI der DHBW übergeben. Geschätzte Dauer ca. 3 Manntage.

## 6.2 Rollen

|  |  |
| --- | --- |
| **Mitarbeiter** | **Rolle** |
| Broer Jan | Projektmanagement / Developer |
| Böhler Anton | Projektmanagement / (Backend-) Developer |
| Cosovan Andrei | Projektmanagement / Developer |
| Dietrich Alexander | Head of Design / Projektmanagement / Developer |
| Hügle Christian | Projektmanagement / Developer |
| Schüler David | Projektmanager / Developer |

# Anhang

## 7.1 Hinweise

Das Tool wird ausschließlich auf Windows 10 mit dem Firefox Version 87.0 Chrome Version 90.0 entwickelt. Kompatibilität mit Safari und dem Internet Explorer sind nicht vorgesehen.