Dokumentation

Wahlprojekt Wintersemester 2022/23

Hochschule RheinMain

**Digitaler Ausbildungsbegleiter für IT-Berufe**

Erstellen einer Online-Lernportal für IT-Auszubildende.

Die Ausbilder können auf der Plattform die Prüfungsinhalte bereitstellen. 

|  |  |
| --- | --- |
| Prüfer | Björn Fischer, Prof. Dr. Eva-Maria Iwer |
| Projektteilnehmer | Adrian Füller  Aischa Kabir  Dominik Braungardt  Leon Schmerr  Marvin Heyne  Mehindi Ibrahimova  Samet Kale  Yassine Chakir |
| Durchführungszeitraum: | 11.10.2022 bis 03.02.2023 |

Inhaltsverzeichnis

[1. Einleitung 3](#_Toc126327749)

[1.1 Motivation 3](#_Toc126327750)

[1.2 Aufgabenstellung 3](#_Toc126327751)

[1.3 Projektziele 4](#_Toc126327752)

[1.4 Stakeholder 4](#_Toc126327753)

[2. Anforderungen 5](#_Toc126327754)

[2.1 Funktionale Anforderungen 7](#_Toc126327755)

[2.1.1 Personas 7](#_Toc126327756)

[2.1.2 Szenarien 8](#_Toc126327757)

[2.2 Nicht-Funktionale Anforderungen 14](#_Toc126327758)

[3. Projektplanung 16](#_Toc126327759)

[3.1 Teamplanung 16](#_Toc126327760)

[3.2 Projektablaufplan 17](#_Toc126327761)

[3.2.1 Meilensteine 17](#_Toc126327762)

[3.2.2 Zeitplan und Kommunikationsstruktur 19](#_Toc126327763)

[4. Kontextabgrenzung 19](#_Toc126327764)

[5. Bausteinsicht 20](#_Toc126327765)

[5.1 Gesamtsystem 20](#_Toc126327766)

[5.2 Ebene 1 22](#_Toc126327767)

[5.3 Ebene 2 22](#_Toc126327768)

[5.4 Systemkomponente 23](#_Toc126327769)

[5.4.1 Web Interface 23](#_Toc126327770)

[5.4.2 Frontend 23](#_Toc126327771)

[5.4.3 Backend 23](#_Toc126327772)

[5.4.4 Server 24](#_Toc126327773)

[5.4.5 Datenbank 24](#_Toc126327774)

[5.4.6 PDF-Generator 24](#_Toc126327775)

[5.4.7 Datei-Management 24](#_Toc126327776)

[6. Laufzeitsicht 26](#_Toc126327777)

[6.1 Dateihandling 27](#_Toc126327778)

[6.1.1 Dateidownload 27](#_Toc126327779)

[6.1.2 Dateiupload 28](#_Toc126327780)

[6.2 Datenbankanbindung 28](#_Toc126327781)

[6.3 Suche 28](#_Toc126327782)

[6.4 PDF-Generierung 29](#_Toc126327783)

[6.5 Rucksack-Filter 29](#_Toc126327784)

[6.6 Startseite 29](#_Toc126327785)

[6.7 Authentifizierung 30](#_Toc126327786)

[6.7.1 Session 30](#_Toc126327787)

[6.7.2 Login / Registrierung 30](#_Toc126327788)

[7. Verteilungssicht 31](#_Toc126327789)

[8. Konzepte 32](#_Toc126327790)

[8.1 Konzeption der Webseite mit Balsamiq Wireframes 32](#_Toc126327791)

[8.2 Profilseite (Frontend) + PW ändern Feature 32](#_Toc126327792)

[8.3 Dashboard / Fortschritts-Seite 32](#_Toc126327793)

[8.4 Fehlermeldungen/Statusmeldungen 33](#_Toc126327794)

[8.5 Datenbank 33](#_Toc126327795)

[8.6 Programmierung 35](#_Toc126327796)

[8.6.1 Experten- Azubiansicht 35](#_Toc126327797)

[8.6.2 Thema 36](#_Toc126327798)

[8.6.3 Rucksack 37](#_Toc126327799)

[8.6.4 Inhalte erstellen 38](#_Toc126327800)

[8.7 Webanwendung starten 38](#_Toc126327801)

[9. Architekturentscheidungen 39](#_Toc126327802)

[10. Manuelles Testing 41](#_Toc126327803)

[11. Fazit und Ausblick 47](#_Toc126327804)

[12. Anhang 49](#_Toc126327805)

[12.1 Glossar 49](#_Toc126327806)

[12.2 Abbildungsliste 50](#_Toc126327807)

[12.3 Tabellenliste 51](#_Toc126327808)

# Einleitung

Das Wahlprojekt „Digitaler Ausbildungsbegleiter für IT-Berufe“ wird an der Hochschule RheinMain durchgeführt und von Studierenden der Studiengänge Angewandte Informatik und Informatik – Technische Systeme entwickelt. Die Projektdokumentation stellt die Planungsphase, die Implementierung und die Umsetzung des Systems dar.

## Motivation

Es sind viele diverse LMS für unterschiedliche Zielgruppen auf dem Markt verfügbar, aber es gibt kein LMS, welches speziell für IT-Auszubildende angedacht ist. Es fehlen Funktionalitäten, wie das eigenständige Hochladen von Unterrichtsmaterialien oder die Möglichkeit Empfehlungen für andere Lern-Webseiten abzugeben. Viele Plattformen bieten lediglich die Möglichkeit an, vorhandene Kurse zu abonnieren und zu absolvieren. Diese Kurse verfügen aber meist nicht über zugeschnittenen Inhalte, die für die IT-Ausbildung notwendig sind. Auch eine individuelle Zusammenstellung aus mehreren Materialien oder Inhalten ist nicht möglich, wodurch zusätzlicher Aufwand betrieben werden muss, um beispielsweise Übungsklausuren zu erstellen.

Ein solches System kann dazu beitragen, Papier- und Druckkosten zu sparen, da Informationsquellen und Übungsmaterialien digital bereitgestellt werden können. Außerdem kann ein LMS von überall genutzt werden, sodass damit auch Home-Office möglich ist. Die Experten sparen Zeit, da sie mit wenigen Mausklicks beispielsweise Prüfungsaufgaben generieren oder eine Kollektion an Inhalten für eine Gruppe von Azubis bereitstellen können.

## Aufgabenstellung

Die Aufgabenstellung des Projektes beinhaltet die Entwicklung eines Lernmanagementsystems (LMS) für die IT-Auszubildenden und IT-Ausbilder der Hochschule RheinMain. Dieses LMS soll in Form einer Webseite dargestellt werden, sodass die Nutzer über eine webbasierte Benutzeroberfläche die Anwendung nutzen können. Es soll fachbezogene Materialien, wie beispielsweise Prüfungsaufgaben oder Foliensätze von Lehrer\*innen in Form von Dateien (PDFs, PNGs, …) oder Hyperlinks organisieren und verwalten, indem es diese Daten in einer Datenbank und Dateien auf einem Server speichert. Diese Materialien sollen nach der Speicherung von den Expert\*innen in Inhalten mit dazugehörigen Themen untergliedert werden können, sodass die Auszubildenden danach selbst suchen können oder die Inhalte ihnen zur Bearbeitung zugewiesen werden.

IT-Auszubildende können sich auf dem System anmelden und einloggen, nach Themen suchen und auf Übungseinheiten zugreifen. Sie können zusammengestellte Übungseinheiten bearbeiten, ihren Lernfortschritt dokumentieren und das System zur Selbsteinschätzung ihrer Lernfortschritte nutzen.

IT-Ausbilder\*innen haben Zugriff auf das System und erhalten eine Übersicht über die Lehrenden und Lehrlinge. Sie können den Lernfortschritt der Lehrlinge verfolgen und passenden Themen erstellen. Sie können ihre Skripte, Übungen, Video-Links und weitere inhaltsbezogene Themen in dem System bereitstellen, die für die IT-Auszubildende zugänglich sind.

## Projektziele

Das Ziel des Projekts ist es, ein Online-Lernportal zu entwickeln, um die IT-Auszubildenden bei der Ausbildung zu unterstützen. Dafür sollen Informationsquellen und Prüfungsaufgaben organisiert werden können. Sie sollen auf einem Server hinterlegt, bearbeitet und für die IT-Azubis zur Verfügung gestellt werden. Die Bereitstellung soll es unter anderem ermöglichen, Übungseinheiten zusammenzustellen in Form einer Kollektion oder der bereits genannten PDFs. Außerdem soll der Lernfortschritt der IT-Azubis dokumentiert und nachverfolgt werden können.

## Stakeholder

Abbildung 1: Stakeholder

Zu den Stakeholdern gehören die Ausbilder, die Auszubildenden sowie das Personal, das für die Durchführung und die Umsetzung des Projekts verantwortlich sind. Zur Veranschaulichung der Einteilung wurde eine Abbildung vorbereitet.

Die folgende Tabelle gibt einen Überblick über die wichtigsten Stakeholder des Projekts eines Lernportals für IT-Azubis in Zusammenarbeit mit der Hochschule RheinMain und ihre Erwartungshaltungen bezüglich der Architektur und Dokumentation des Systems. Diese Tabelle dient dazu, die Anforderungen und Erwartungen der Stakeholder zu verstehen und sicherzustellen, dass das Projekt ihre Bedürfnisse erfüllt.

|  |  |  |
| --- | --- | --- |
| **Stakeholder** | **Rolle** | **Erwartungshaltung** |
| Ausbilder | Experte, primäre Zielgruppe | Ausbilder sind für das Pflegen des Lernsystems zuständig. Die Plattform soll eine Möglichkeit anbieten, den Lernfortschritt der Auszubildenden zu verfolgen, um gegebenenfalls Unterstützung anzubieten. |
| Auszubildende | Hauptnutzer | IT-Azubis sind Stakeholder und können von Entscheidungen und Aktionen des Projekts betroffen sein. Ihre Meinungen und Anforderungen sollten in die Entscheidungsfindung und die Umsetzung des Projekts einbezogen werden. |
| Personal | Systemunterstützung | Das Personal setzt seine Fähigkeiten und Kenntnisse in die Umsetzung und Entwicklung des Projektes ein. |

Tabelle 1: Stakeholder

# Anforderungen

In diesem Abschnitt werden die Anforderungen definiert, dazu gibt es drei Anwendungsfalldiagramme, die eine Übersicht der im Abschnitt 1.3 definierten Ziele zeigen.

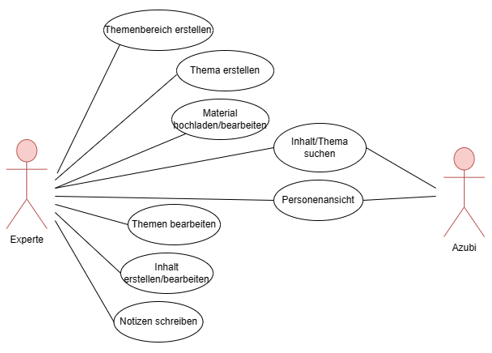


Abbildung 2: Use-Case Diagramm für die Organisation von Informationen und Prüfungen

Die Abbildung 1.1 zeigt, wie die Anwendungsfälle zu dem 1.3.a zu realisieren sind. In dem Diagramm ist es zu sehen, dass die Anwender der abgebildeten Funktionen Experte und Azubi sind. Die wichtige Funktionalitäten „Thema erstellen“, „Inhalt erstellen“ werden im Abschnitt 2.1 genauer erläutert.

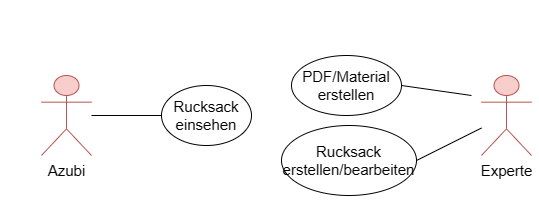


Abbildung 3: Use-Case Diagramm für die Zusammenstellung von Übungseinheiten

In dem Use-Case Diagramm 2 werden die sogenannten Rucksäcke abgebildet, die zur Zusammenstellung der Übungseinheiten dienen.

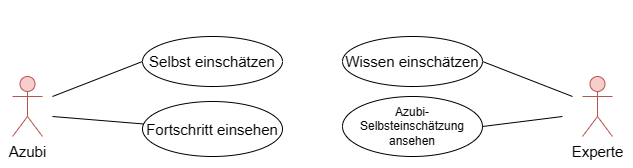


Abbildung 4: Use-Case Diagramm für die Dokumentation des Lernfortschritts

In der Abbildung 1.3 wird dargestellt, wie die Lernfortschritte im System dokumentiert werden.

Die Anwendungsfälle, die alle Zwecke des Systems abdecken, werden in dem folgenden Abschnitt erläutert, die beinhalten jeweils 2 Akteure, die mit dem System interagieren.

## Funktionale Anforderungen

Um die Bedürfnisse von Akteuren und deren Lösungen klarzustellen, werden die Personas verwendet.

### Personas

Table

Description automatically generated with medium confidence

Abbildung 5: Persona Experte

**AK1**: **Experte:** Der Experte erstellt einen Rucksack für mehrere oder einen Azubi. Dieser Rucksack wird auf die Bedürfnisse des Azubis angepasst, befüllt mit verschiedenen Themen/Materialien und zum Schluss erstellt. Hier werden Problemgebiete des Azubis mit dem Rucksack verdeutlicht. Ist der Azubi mit dem Themenbereich Programmierung überfordert und hängt mit dem Stoff hinterher, kann der Experte auf diese Lücke reagieren und diesen Rucksack so gestalten, dass der Azubi diese Lücke schließen kann. Um den Fortschritt des Azubis zu dokumentieren und ihm bewusst zu machen, wie gut er in einem Thema ist, schätzt AK1 ihn in einem Punkten-Skala zwischen 0-100.

Table

Description automatically generated with medium confidence

Abbildung 6: Persona Azubi

**AK2**: **Azubi:** Der Azubi löst die von dem Experten erstellten Rucksack. Da der Azubi in dem Rucksack mit dem Themengebiet Programmierung Schwierigkeiten hatte, kann er diese anhand des Rucksackes kompensieren und den Stoff im Unterricht aufholen oder die gestellten Aufgaben lösen. Wenn der Azubi das Gefühl hat, mit diesem Rucksack überfordert zu sein, kann er den Experten für einen neuen Rucksack anfragen, der genauer auf die Probleme des Azubis eingeht. Anschließend hat AK2 die Möglichkeit, sein Wissen in jeweiligen Themen einzuschätzen und die Einschätzung vom AK1 anzusehen.

Die nächsten Szenarien (SZ) sollen die Anwendung verdeutlichen und die Abläufe werden anschließend mit einem Aktivitätsdiagramm dargestellt.

### Szenarien

**SZ1**: **Rucksack erstellen:**

Der Experte Markus erstellt auf Wunsch des Azubis Thorsten einen neuen Rucksack. Dazu wählt Markus zunächst einen passenden Namen für den zu erstellenden Rucksack aus, Programmierung mit C++. Bestätigt seine Auswahl „Rucksack Erstellen“ und der Rucksack wurde mit dem Namen angelegt. Der erstellte Rucksack ist zunächst leer und kann nun „bearbeitet“ werden. In dieser Bearbeitungsphase kann der Experte nach dem Auswählen, den erstellten Rucksack umbenennen und bearbeiten. Bei der Phase der Bearbeitung kann Markus nun aus den Inhalten, die schon bestehen, Inhalte dem Rucksack hinzufügen. Passend zum Namen des Rucksackes kann Markus nun Inhalte und Materialien, die in SZ2 und SZ3 erstellt sind, zu Programmierung mit C++ einfügen, die dann der Azubi Thorsten bearbeiten kann.

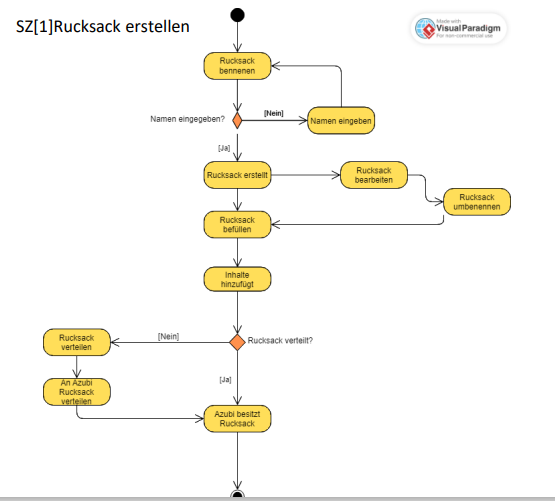


Abbildung 7: Aktivitätsdiagramm für das Erstellen eines Rucksack

**SZ2: Thema erstellen:**

Da der Experte Markus für Thorsten erstellen möchte, braucht er dazu Themen zu erstellen. Zuerst erstellt er auf der Themen Seite ein Themenbereich, in dem er spezialisiert ist. Dafür gibt er einen Namen wie „Programmierung C++“ für den Themenbereich, den er erstellen möchte. Daraufhin bekommt er die Meldung, dass das neue Überthema hinzugefügt wurde und er kann auf der Themenbereich-Liste und im Kuchendiagramm das neu erstellte Thema sehen. Nun möchte er zu diesem Überthema ein Unterthema hinzufügen. Dazu gibt er den Titel „Einführung in C++“ im dafür vorgesehenen Feld ein. Nach der Bestätigung reagiert das System mit einer erfolgreichen Meldung und das Unterthema ist zu sehen.

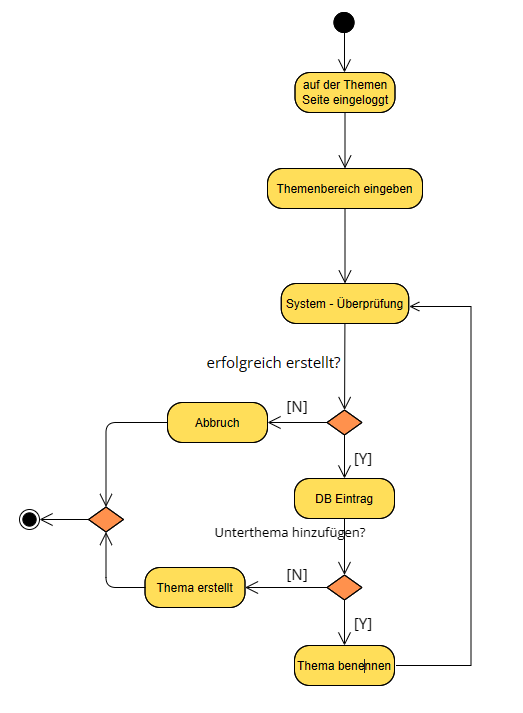


Abbildung 8: Aktivitätsdiagramm für das Erstellen von Themen

**SZ3: Inhalt erstellen**

Die Themen (SZ2) sind erstellt, es gibt nur einen Schritt zur Erstellung des Rucksackes, nämlich die Inhalte. Der Experte Marcus kann den Inhalt erstellen, indem er dem Inhalt einen Namen gibt. In diesem Fall nennt er den Inhalt Programmieren in C++. Danach wählt er die zugehörigen Materialien und Themen aus der Liste aus. Er kann auch ein paar nützliche Informationen zu dem neu erstellten Inhalt schreiben. Nachdem er die oben beschriebenen Felder ausgefüllt hat, speichert er den Inhalt und er sieht den neuen Inhalt auf der Liste.

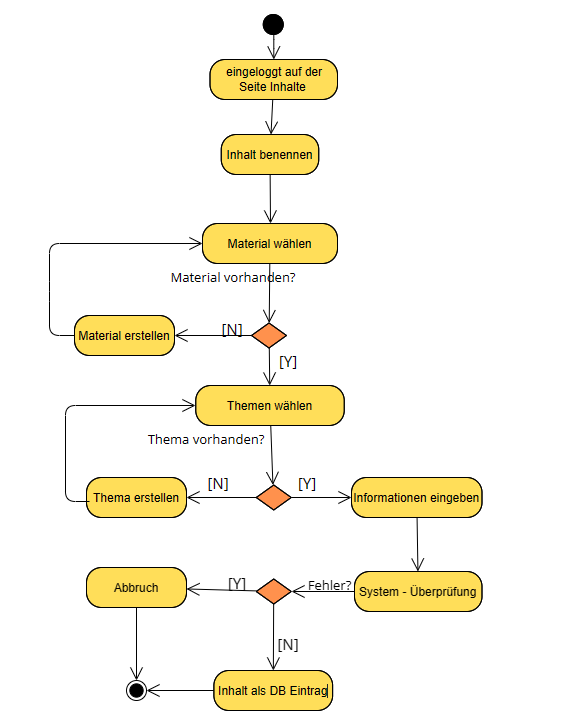


Abbildung 9: Aktivitätsdiagramm für die Erstellung von Inhalten

**SZ4: Zugeteilten Rucksack einsehen und bearbeiten**

Der Azubi Thorsten hatte den Experten Markus dazu gebeten, einen Rucksack zu erstellen, der auf seine Fragen und Problemstellungen basierend angelegt wurde. Thorsten hat Schwierigkeiten in Programmierung mit C++ und wünscht sich mithilfe des Rucksackes seine Defizite zu kompensieren. Thorsten kann stets auf seine zugewiesenen Rucksäcke zugreifen und diese dann anschließend bearbeiten. Öffnet Thorsten nun seinen persönlichen zugewiesenen Rucksack, kann er mit dem Rucksack arbeiten. Dieser beinhaltet Lernstoffe und Übungen zu Programmierung mit C++. Thorsten hat nun seinen Rucksack geöffnet und sieht die Inhalte wie Lernstoffe und Übungen. Hier kann Thorsten auch nach der gewünschten Form filtern und ausschließlich nur mit Videos arbeiten oder aber auch Übungsaufgaben vom Experten anzeigen lassen und bearbeiten. Während der Bearbeitung der Inhalte, kann Thorsten auch eine PDF generieren lassen, damit seine Inhalte stets gesichert sind und in einer ausdruckbaren Form abgespeichert werden kann.

Diagram

Description automatically generated

Abbildung 10: Aktivitätsdiagramm für das Einsehen und Bearbeiten eines Rucksacks

**SZ5 Wissen einschätzen**

Nachdem der Experte Marcus den Rucksack erstellt hat, kann der Azubi Thorsten weiterlernen. Nun möchte er den Azubi Thorsten in jeweiligen Themen mit einer Skala zwischen 1 und 100 bewerten. Dafür wählt er aus der Azubi-Liste den Namen von dem Azubi Thorsten aus und gibt bei dem Thema Programmierung C++ die Bewertung 90 ab. Das bedeutet, der Rucksack SZ1 hat Thorsten geholfen, sein Wissen zu erweitern.

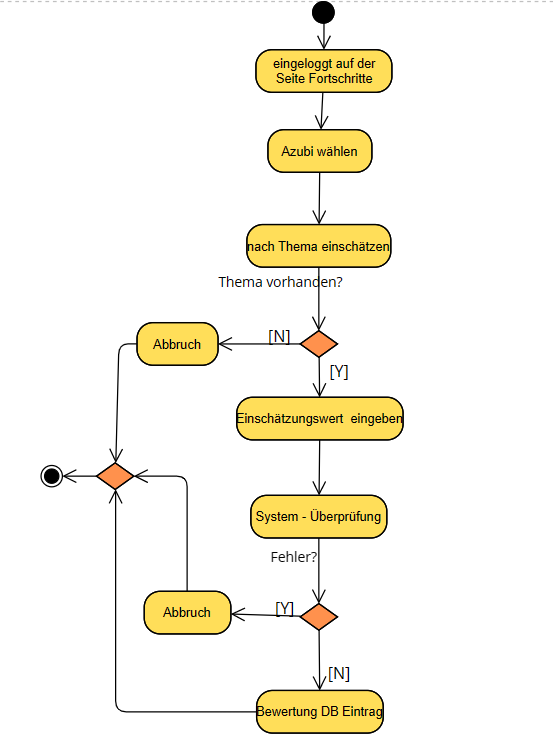


Abbildung 11: Aktivitätsdiagramm für die Selbsteinschätzung

**SZ6 Wissen einschätzen**

Die Vorgabe für das Festhalten eines Lernfortschrittes wurde durch ein Thema und Rucksack-basiertes Einschätzungs-System entwickelt.

* Azubis können sich pro Rucksack und pro Thema selbst einschätzen mit einer Prozent-Angabe zwischen 0 und 100.
* Experten sehen im Dashboard für jeden Azubi die jeweilige Selbsteinschätzung und können dazu noch eine Expertenseitige Schätzung (auch 0-100%) abgeben.

## Nicht-Funktionale Anforderungen

|  |  |
| --- | --- |
| Eigenschaften | Beschreibung |
| Verfügbarkeit | Zugänglich für Benutzer Experten und IT-Auszubildenden sein. Hierfür ist eine Internetverbindung notwendig. |
| Übertragbarkeit | Das System soll mindestens auf den Computer oder Laptop sowie verschiedenen Browser für die Anwender verfügbar sein, um den Zugang der Benutzer zu gewährleisten. |
| Skalierbarkeit | Das System muss gleichzeitige Nutzung für mehrere Benutzer gewähren und das Hinzufügen der Inhalte der jeweiligen Themenbereiche unterstützen. |
| Effizienz | Eine einfache Bedienung und das Organisieren der Inhalte und Daten gewährleisten, um eine effiziente Nutzung zu ermöglichen. |
| Funktionalität | Das System muss die Funktionalität Überblick auf die Themen, Einblick der Übungseinheiten sowie Informationsquellen für die Azubis aufweisen und die Selbsteinschätzung für die Azubis und den Experten zugänglich sein. Das Hochladen verschiedener Medienformate unterstützen. |
| Zuverlässigkeit | Das System soll stabil arbeiten und bei Datenverlusten oder bei nicht korrektem Hochladen der Dateien eine Fehlermeldung anzeigen. |
| Benutzerfreundlichkeit | Die Lernplattform muss einen einfachen Überblick für Experten und Azubis sein. Die Prüfungsinhalte und Informationsquellen müssen schnell auffindbar sein. Das System soll auf der Webseite eine Navigationssystem, Suchfeld für Experten sowie eine übersichtliche Benutzeropferfläche haben. |
| Sicherheit | Die personenbezogenen Daten und die Materialien müssen geschützt abgelagert werden. Eine sichere Übertragung der Inhalte und Daten anbieten. |
| Leistung | Das System muss einen funktionsfähigen Navigationssystem oder einen Zugang für die Experten und Azubis ermöglichen. Zudem eine schnelle Themensuche für Experten haben und die Themen verwalten können. Das System muss Informationsquellen und Lerninhalte organisieren sowie verwalten können, damit IT-Azubis diese einsehen können. Das muss die Möglichkeit anbieten, dass die Benutzer die Lernfortschritte dokumentieren und verfolgen können. |
| Zugänglichkeit | Der Zugang muss mit einer Anmeldung erfolgen und muss ortsunabhängig nutzbar sein. |
| Kompatibilität | Das System muss für die üblichen Browserarten kompatibel sein. |

Tabelle 2: Nicht-Funktionale Anforderungen

# Projektplanung

## Teamplanung

|  |  |  |
| --- | --- | --- |
| **Feature** | **Verantwortliche** | **Beschreibung** |
| Seitenstruktur | Adrian | Aufbau und Anordnung der Seiten |
| Home / Startseite | Adrian | Willkommensseite mit Verlauf der Suche |
| Sidebar | Marvin, Samet | Anzeige der Unterseiten auf allen Seiten |
| Header | Yassine | Suchmöglichkeit auf allen eingeloggten Seiten |
| Suche | Adrian, Leon, Mehindi | Suche nach Themen und Inhalten (Lernstoffe und Übungen) über Suchbar. Ausgabe von Suchergebnissen |
| Themen | Dominik, Yassine, Leon | Darstellung der Themen in interaktivem Kuchendiagramm  Möglichkeit für CRUD der Themen |
| Leute | Mehindi, Aischa, Yassine | Auflistung der Experten und Azubis |
| Rucksäcke | Leon, Adrian | Kollektionen von Inhalten mit Lernfortschritt und Möglichkeit von CRUD |
| PDF-Erstellung | Adrian, Yassine | Erstellung von PDFs (zu neuen Materialien) aus einzelnen Inhalten (PNGs) bzw. lokalen PNGs |
| Material | Dominik, Yassine | CRUD von Material |
| Inhalte | Dominik, Yassine | CRUD von Inhalten |
| Fortschritte | Marvin | Selbsteinschätung und Experteneinschätzung von Azubis |
| Profil | Marvin | Profilansicht und Bearbeitung |
| Session (Login/Register) | Adrian, Yassine, Samet | Möglichkeit eines Logins/Registeriung über Session |

Tabelle 3: Teamplanung

## Projektablaufplan

### Meilensteine

Meilensteine sind wichtige Etappen im Projektablauf, die das Fortschreiten des Projekts zu markieren und dessen erfolgreiche Umsetzung sicherstellen sollen. Die Ziele wurden folgendermaßen dargestellt und beschrieben:

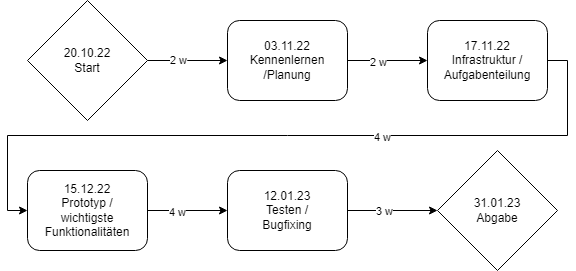


Abbildung 12: Meilensteine / Roadmap

|  |  |  |
| --- | --- | --- |
| Phasen | Datum | Beschreibung |
| Beginn | 03.11.2022 | * Zu diesem Zeitpunkt beginnt offiziell das Projekt * Gemeinsam werden die ersten Schritte geplant. * Die Projektziele werden festgelegt. |
| Vorbereitung | 17.11.2022 | * In diesem Meilenstein lernen sich alle Teammitglieder kennen * Die geplante Vorgehensweise wird diskutiert und eine Infrastruktur aufgebaut. * Das Personal wird in die jeweiligen Teams aufgeteilt. * Die notwendigen Infrastrukturen wie Server, Datenbanken und Tools werden festgelegt. * Die die Aufgaben innerhalb der Teams Backend, Frontend und Datenbankenteam verteilt. |
| Realisierung | 15.12.2022 | * Das erste Release wird veröffentlicht. * Weitere Planungen werden für das Endprodukt bestimmt. * Der Prototyp wird dem Prüfer vorgestellt. |
| Testverfahren | 12.01.2023 | * In diesem Meilenstein wird der erste Prototyp des Lernsystem vorgestellt * Die wichtigsten Funktionalitäten wie Themenseiten, Übungsseiten und die Bereitstellung von Inhalten gezeigt. * Die letzten Fehler und Bugfixes besprochen. |
| Abgabe | 31.01.2022 | * Das Lernsystem weitgehend fertiggestellt und die Fehler gehoben. * Der letzte Meilenstein, in dem das Projekt abgeschlossen und abgegeben wird. |

Tabelle 4: Beschreibung der Meilensteine

### Zeitplan und Kommunikationsstruktur

Während des Projekts wurde Jira als Tool verwendet, um die Erarbeitung der Tasks der Teammitglieder im Überblick zu behalten und Unterstützung anzubieten, um gegeben falls Unterstützung anzubieten. Wöchentliche Projektbesprechungen wurden durchgeführt, um den Fortschritt des Projekts zu besprechen und eventuelle Herausforderungen zu lösen. Diese Meetings wurden über verschiedene Kommunikationskanäle wie E-Mail, die Gitlab-Seite der Hochschule, Discord und eine WhatsApp-Gruppe abgehalten.

Wichtige Entscheidungen und Aktivitäten wurden auf Gitlab und Jira festgehalten. Zusätzlich wurden interne Teambesprechungen über Discord durchgeführt, um spezifische Probleme oder Herausforderungen innerhalb des Teams zu besprechen und zu lösen.

Die Dokumentation des Projekts wurde sowohl auf Gitlab als auch lokal gespeichert, damit sie jederzeit leicht zugänglich ist.

# Kontextabgrenzung

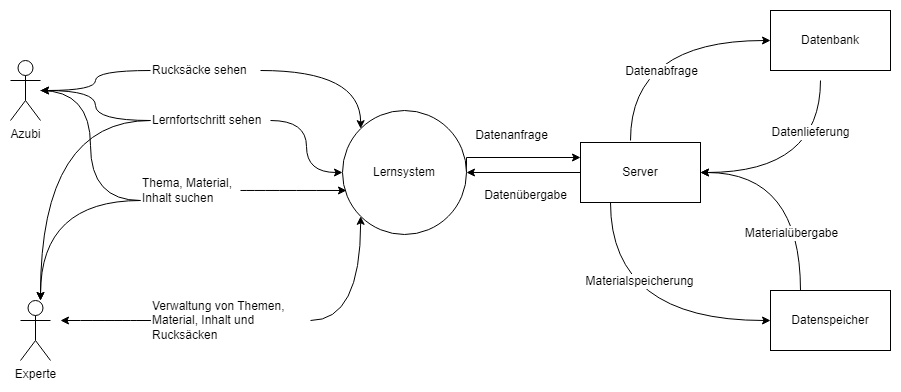


Abbildung 13: Kontextdiagramm

|  |  |  |  |
| --- | --- | --- | --- |
| Kommunikations-beziehung | Beschreibung | Eingabe | Ausgabe |
| Azubis | Der Azubi nutzt das System, um die Themen, Rucksäcke einzusehen sowie die Lernschritte zu verfolgen. | * Selbsteinschätzung * Suche | * Themen * Rucksäcke * Suchergebnisse |
| Experte | Der Experte pflegt die Informationsquellen und Übungseinheiten in das System ein. Zudem verfügt er die Rechte die Themen, Inhalte, Rucksäcke zu erstellen, bearbeiten und löschen. Auch kann er die Lernfortschritte der Azubis sehen. | * Themen * Material, Inhalt * Azubi-Einschätzung * Rucksack * PDF-Erstellung | * Themen * Material, Inhalt * Einschätzung * Rucksack |
| Server | Verbindungsaufbau zwischen System und Anwender | * Dateien | * Dateien |
| Datenbank | Verbindung zu Datenbank zum System | * Datenanfrage * Datenmanipulation | * Abruf der Daten |

Tabelle 5: Beschreibung des Kontextdiagramms

# Bausteinsicht

## Gesamtsystem

Der folgende Abschnitt beschreibt die Zerlegung des Systems. In Komponenten, Schnittstellen und die verwendeten Pakete. Die untenstehende Darstellung zeigt einen minimalen Einblick in die Systemstruktur. Die Weitere Informationen und die Verantwortungen der jeweiligen Blackbox-Sicht sind in den Abschnitt Ebene vorzufinden. Im Systemablauf wird ein detaillierterer Ablauf beschrieben.

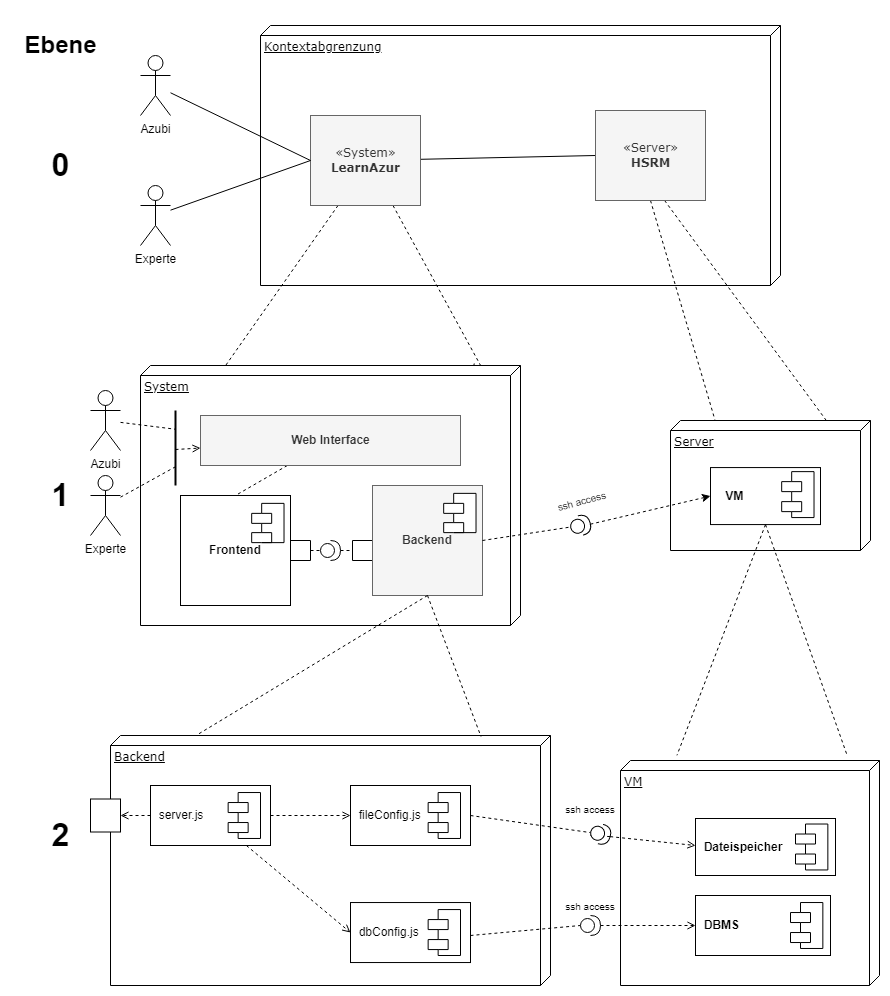


Abbildung 14: Bausteinsicht

## Ebene 1

|  |  |
| --- | --- |
| Komponente o. Subsysteme | Verantwortung |
| Web Interface | Sorgt für die Verbindung zwischen Benutzer und das System, die über eine Benutzeroberfläche sichtbar ist. |
| Frontend | Unterstützt das System mi den Programmiersprachen und den Web Interface für die Logik der LMS. |
| Backend | Sorgt für die serverseitige Service und Verbindungsaufbau zwischen System und Server |
| VM | Verwaltung und Bereitstellung der Materialien für das LMS über SSH erreichbar |
| DBMS | Zur Unterstützung und Verwaltung der relationalen Datenbank wird das Datenbank Management System verwendet. |
| Operating System | Verwaltet die Dateistruktur und speichert die Ressourcen für das System, wie z.B. Materialien, Daten |
| File Management | Ordnerstruktur für die Organisation des Projektes und enthält relevante Dateien |

Tabelle 6: Ebene 1 der Bausteinsicht

## Ebene 2

|  |  |
| --- | --- |
| Schnittstellen/Pakete | Verantwortung |
| Web Browser | Darstellung der LMS über eine webbasierte Benutzeroberfläche |
| PDF-Generator | Für die Erzeugung und der Konvertierung der Bilder zu einem PDF-Format |
| MySQL | Bearbeitung und Speicherung der relationalen Datenbank |
| Files | Alle relevanten Dateien werden in Das Betriebssystem abgespeichert. |

Tabelle 7: Ebene 2 der Bausteinsicht

## Systemkomponente

### Web Interface

Für die Nutzung des LMS muss der Benutzer sich über einen Webbrowser mit seinen Zugangsdaten anmelden. Danach ist für ihn die Funktionalitäten und das LMS sichtbar.

### Frontend

Das System ruft über den Server die jeweiligen Klassen und Schnittstellen auf und stellt diese für den Nutzer dar. Mithilfe von CSS, Bootstrap und den EJS wird die Datebank und auf der Webseite sichtbar gemacht. Der Datenspeicher leitet die jeweiligen Informationen an das System weiter und diese werden auf der Webseite dargestellt.

Bootstrap ist weit verbreitet und bietet viele Design-Elemente, die einfach angepasst werden können. Außerdem bietet Bootstrap eine gute Unterstützung für mobile Geräte. Konkret wurde hier die Version 5.2.2 verwendet, welche zum zeit der Entwicklung aktuell war.

### Backend

Für das Backend wird JavaScript-Laufzeitumgebung Node.js gewählt. Das Seiten-Routing wird ExpressJS verwendet und für die Weiterleitungen und HTTP-Abfragen in Node.js. Mit dem Paket „express-session“ ist es möglich einem Nutzer eine Session im Browser zu geben, sodass Anmeldungen und Registrierungen durchgeführt werden können. Um Formulare vernünftig auslesen und weiterverarbeiten zu können werden die Pakete „formidable“ und „body-parser“ angewendet. Für das Seiten-Routing wird ExpressJS verwendet. Es vereinfacht Weiterleitungen und HTTP-Abfragen in Node.js.

### Server

Als Server wird eine virtuelle Maschine (VM) der Hochschule RheinMain (HSRM) verwendet. Auf dieser VM läuft das Betriebssystem Debian 11. Die Hardware wird von der HSRM zur Verfügung gestellt und von ihnen verwaltet. Auf dem Server werden Materialien für das LMS gespeichert und es läuft darauf die Datenbank. Der Server ist aufgrund von Sicherheitsvorkehrungen von der Hochschule nur über SSH erreichbar. Deswegen wurde für jeden Entwickler ein SSH-Keypair erstellt, welches auf dem Server und auf dem Rechner des Entwicklers hinterlegt ist.

### Datenbank

Damit auf die Datenbank und die Materialien zugegriffen werden kann, muss auf den Server über SSH zugegriffen werden. Dieser Zugriff geschieht mit den Paketen „ssh2“ für die SSH-Verbindung und „mysql2“ für den Zugriff auf die Daten aus der Datenbank.

Im Überordner „material“ werden die Dateien je nach Datentyp im Unterordner gespeichert (z.B. PDFs im Ordner „pdfs“, PNGs im Ordner „imgs“, usw.). Mittels Shell-Skript wird jede Nacht eine Sicherung der Datenbank angelegt. Dabei werden alle Tabellen und Daten in einer SQL-Datei abgespeichert, sodass im Krisenfall die neueste SQL-Datei ausgeführt werden kann, wodurch der DB-Stand des Vortages wiederhergestellt werden kann. Diese Dateien bleiben 30 Tage vorhanden, sodass eine Backup-Datei nach 30 Tagen automatisch gelöscht wird.

### PDF-Generator

Für eines der Kernthemen, der PDF-Erzeugung, wird das Paket „pdf-lib“ eingesetzt. Damit können ein oder mehrere Bilder in eine PDF konvertiert werden. Zusätzlich gibt es Einstellungen für das Layout der zu erzeugenden PDF.

### Datei-Management

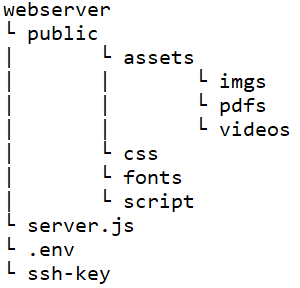


Abbildung 15: Datei-Management

Um eine vernünftige Organisation des Projektes gewährleisten zu können, wurde eine vorab festgelegte Ordnung der Datei-Struktur eingehalten. In der Abbildung 15 wurden die einzelnen Dateien mit Ausnahme von relevanten Dateien, die im Folgenden beschrieben werden, weggelassen. Dafür gibt es auf oberster Ebene die Dateien „package-lock.json“ und „package.json“, sowie den Ordner „node\_modules“, mit denen die von npm installierten Pakete verwaltet und dokumentiert werden. Alle neu erstellten Dateien und Programme befinden sich im Ordner „webserver“. Darin befindet sich der Ordner „public“, welcher alle Nicht-HTML-Dateien enthält, die für den Nutzer sichtbar sind. Im Unterordner „assets“ von „public“ befinden sich weitere Unterordner für die verschieden verwendeten Medien „imgs“, „pdfs“ und „videos“. In den Unterordnern „css“, „fonts“ und „script“ von „public“ liegen des Stylings, Schriftarten und clientseitigen Skripte der Webanwendung. Neben dem Ordner „public“ befinden sich im Ordner „webserver“ die serverseitigen Skripte, beginnend mit der Datei „server.js“, welche zum Start des Servers ausgeführt werden muss. In der Datei „.env“ befinden sich die Umgebungsvariablen (Environment Variables), welche für den Zugang zum Datei- und Datenbankserver benötigt werden. Auch wird der im Ordner „ssh-key“ gespeicherte Private Key des Nutzers für den Zugang zum Server verwendet. Im Ordner „views“ liegen die EJS-Seiten, welche als Template-Sprache zusätzlich zu HTML verwendet wird. Darin gibt es noch einen Unterordner „partials“ für EJS-Dateien, die keine eigenständige Seite darstellen, sondern nur von anderen EJS-Dateien eingebunden werden.

Neben dieser Ordnerstruktur wurde darauf geachtet, dass keine Monolithen entstehen, also der komplette Code innerhalb einer Datei entwickelt wird. Damit soll eine einfache Versionierung ermöglicht werden, sodass beispielsweise das Mergen mittels Git einfacher durchgeführt werden kann. Dementsprechend gibt es beim clientseitigen Styling über CSS und Skripting über JavaScript eine eigene Datei für jede EJS-Seite, sofern es notwendig ist. Trotzdem gibt es eine globale Skripting und Styling-Datei für Styles und Skripte, die auf mehreren oder allen Seiten angewendet wird, wie beispielsweise im Header bzw. der Suchleiste. Des Weiteren wurde der serverseitige Code auf mehrere Dateien ausgelagert, sodass in der Startdatei „server.js“ nur das Routing, die Nutzer-Session und die Datenbank-Befehle stattfinden. Zusätzliche Logik, wie die Datenbankanfragen oder die PDF-Generierung werden in separaten Dateien durchgeführt. Auch wurden redundante Elemente wie der Header der Webseite als „partial“ einmal entwickelt und anschließend von allen Dateien eingebunden.

# Laufzeitsicht

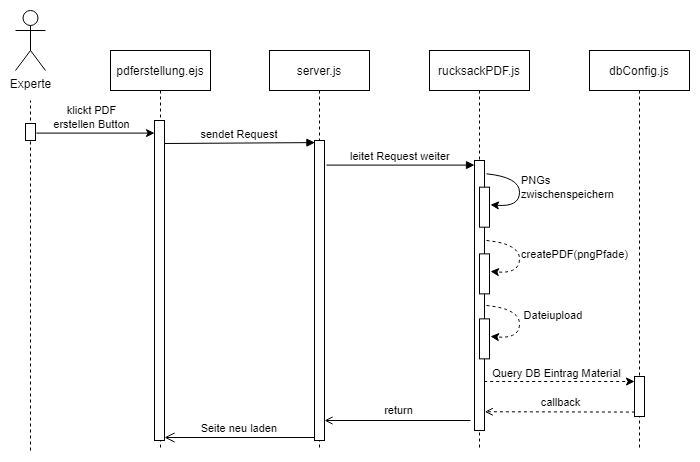


Abbildung 16: Sequenzdiagramm für die PDF-Erstellung über Rucksäcke

In Abbildung 16 wird die PDF-Erstellung über die Rucksack-Seite dargestellt. Dabei wird durch die Bestätigung des Nutzers ein PDF mit allen PNGs des gewählten Rucksacks erstellt. Diese erstellt ein Material in Form eines PDFs und einen dazugehörigen Eintrag in der Datenbank.

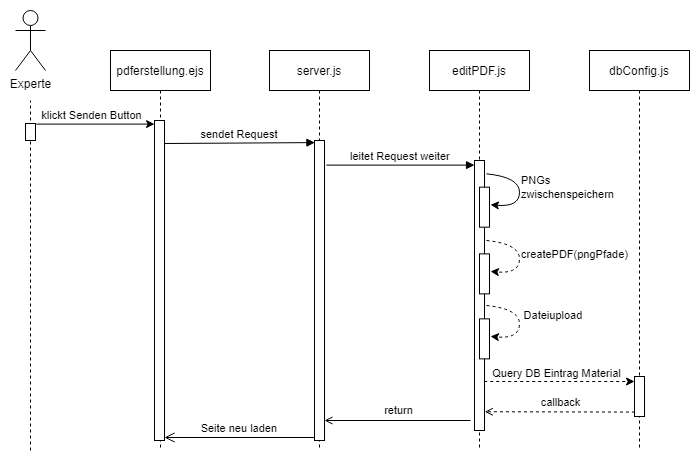


Abbildung 17: Sequenzdiagramm für die Erstellung einer PDF über die PDF-Gen-Seite

In Abbildung 17 wird durch die Bestätigung eine PDF aus selbstgewählten Materialien oder neuen, lokalen Dateien generiert. Diese PDF wird als Material abgespeichert und ein DB-Eintrag wird hinterlegt.

## Dateihandling

### Dateidownload

Mithilfe des Dateidownloads werden gespeicherte Dateien vom Server für die Webseite bereitgestellt. Beim Funktionsaufruf werden der Dateityp und der Dateiname der erwarteten Datei übergeben. Weil alle Dateien nach dem gleichen Schema gespeichert wurden (./material/[Dateityp]/[Dateiname]), reichen diese Informationen aus, um die Datei eindeutig zu bestimmen. Die Datei wird im lokalen Pfad „./public/assets/[Dateityp]/[Dateiname]“ gespeichert. Für den Datenverkehr wird mithilfe des Pakets „ssh2-sftp-client“ heruntergeladen.

### Dateiupload

Der Dateiupload wird u.a. für das Hochladen von neuen Materialien verwendet. Dabei wird durch das Absenden des Formulars bei der Materialerstellung die Funktion aufgerufen. Übergeben werden die Daten aus dem gesendeten Formular. Sie enthalten die zu speichernden Dateien mit ihren Pfaden, Dateitypen, Namen, usw. Außerdem wird der neu gewählte Name des Materials, sowie der Materialtyp mit übergeben. Zuerst muss geprüft werden, ob der gewählte Materialtyp eine Datei enthält. Materialen vom Typ Link, Buch oder YouTube haben keine Datei und können direkt mittels INSERT-Befehl in der DB gespeichert werden. Handelt es sich aber um eine PDF, ein PNG oder ein Video, dann muss die mitgesendete Datei auf dem Server gespeichert werden, sofern sie vorhanden ist. Dafür muss geprüft werden, ob der gewählte Materialtyp mit dem Dateityp übereinstimmt. Zusätzlich wird für den Namen geprüft, ob die Dateiendung mit angegeben wurde, oder ob dieser Dateiname bereits auf dem Server vorhanden ist. Je nachdem muss der Dateiname angepasst werden, um Duplikate zu vermeiden. Sobald der Dateiname angepasst wurde, wird der INSERT-Befehl ausgeführt und die Datei mittels PUT auf dem Server in passenden Pfaden gespeichert.

## Datenbankanbindung

Damit ein SQL-Befehl von Node.js aus auf der MySQL-DB ausgeführt werden kann, muss zuerst per SSH mit dem Hochschulserver verbunden werden, auf dem die MySQL-DB angelegt wurde. Dafür werden die hinterlegten Umgebungsvariablen und der SSH-Key verwendet. Sobald eine Verbindung mit dem Server vorhanden ist, wird mit weiteren Umgebungsvariablen eine Verbindung mit der DB auf dem Server durchgeführt. Nach Abschluss dieses Verbindungsaufbaus kann ein SQL-Befehl ausgeführt bzw. übergeben werden. Dieser Aufruf liefert nach seiner Ausführung ein Result zurück welches als Callback zurückgegeben werden kann.

## Suche

Sobald Schlagwörter in die Suchleiste eingegeben und die Suche über die ENTER-Taste oder die Such-Lupe angestartet wurde, werden auf der Serverseite die Suchergebnisse selektiert. Vorab wird in die Tabelle „NutzerSuchen“ ein Eintrag mit der Suchanfrage hinterlegt, sodass diese auf der Startseite als Verlauf angezeigt werden können.

Danach werden einmal die Themen mit den Schlagwörtern durchsucht. Sofern ein Hashtag vor einem Schlagwort geschrieben wurde, wird dieses Wort nur in der Tabelle Themen gesucht. Dabei ist darauf zu achten, dass es keine redundanten Ergebnisse gibt. Wenn beispielsweise die Suche „C C++“ gesucht wird, dann besteht die Möglichkeit, dass der Themenbereich „Programmierung“ zweimal selektiert wird, jeweils einmal pro Schlagwort. Zusätzlich werden die Inhalte nach den Schlagworten durchsucht. Hier muss ebenfalls auf redundante Ergebnisse geachtet werden. Wenn ein Inhalt mit mehreren Themen markiert, wurde die in den Schlagwörtern vorkommen, soll es trotzdem nur einmal als Ergebnis angezeigt werden.

Je nachdem, welchen Materialtypen ein Inhalt hat, wird es verschieden präsentiert. Videos und YouTube-Links haben einen kleinen Videoplayer, welcher über den Link oder den Namen auf den ursprünglichen Pfad des Inhalts weiterleiten. PNGs und PDFs werden im Kleinformat angezeigt und können ebenfalls über den Namen oder den Link vergrößert dargestellt werden. Bei Büchern wird die ISBN, das Kapitel und die Seite mitangezeigt, sofern sie hinterlegt sind und bei Links ist nur der Link hinterlegt. Die Themen werden in ihrem eigenen Reiter angezeigt. Die Inhalte werden aufgeteilt in Übungen und Lernstoffen. Bei allen drei Reitern wird dem Nutzer die Anzahl der gefundenen Themen und Inhalte ausgegeben.

Bei den Themen werden immer nur die Themenbereiche angezeigt und deren Unterthemen als klickbare Themen darunter. Bei den Inhalten wird zu jedem gefundenen Inhalt dessen verknüpfte Themen darunter angezeigt. Auch diese Themen sind klickbar. Wird eines der Themen angeklickt, dann startet man damit automatisch eine neue Suche mit dem angeklickten Thema. Außerdem gibt bei allen Inhalten für Experten die Möglichkeit diesen Inhalt direkt einen Rucksack anzufügen, in dem der Inhalt noch nicht ist.

## PDF-Generierung

Sobald die PDF-Generierung auf der dazugehörigen EJS-Seite vom Experten angestartet wird, bekommt die Datei „server.js“ den Request, welcher an die Datei „editPDF.js“ weitergeleitet wird. Dort werden die gewählten PNGs entweder vom Server geholt oder vom lokalen Nutzer im Webserver zwischengespeichert. Mit den zwischengespeicherten PNGs kann die PDF generiert werden. Dafür wird das Paket „pdf-lib“ verwendet. Die PNGs werden zusammen mit den passenden Parametern übergeben und als Rückgabe gibt es die generierte PDF. Anschließend wird diese PDF auf den Webserver in den passenden Material-Ordner geladen und ein Eintrag in der Datenbank geschrieben. Abschließend bekommt der Server die Info, die Webseite neu zuladen.

## Rucksack-Filter

Bei den Rucksäcken kann jeweils gefiltert werden. Einmal besteht die Möglichkeit, nur Lernstoffe oder nur Übungen anzuzeigen. Und zweiteres können die Materialtypen gefiltert werden.

## Startseite

Auf der Startseite werden zwei Tabellen und ein Infotext angezeigt. Eine Tabelle zeigt die letzten Suchen an, die der Nutzer durchgeführt hat und die andere Tabelle zeigt alle zugewiesenen bzw. erstellten Rucksäcke an.

## Authentifizierung

### Session

Für die Session wird das Paket „express-session“ verwendet. Sobald sich ein Nutzer einloggt, wird ihm eine Session in Form eines Cookies zugewiesen. Auf Serverseite werden globale Variablen mit den Nutzerdaten gefüllt, sodass der Nutzer auch nach Weiterleitungen auf andere Unterseiten der Anwendung als angemeldeter Nutzer erkannt wird. Ist ein Nutzer nicht angemeldet und versucht auf eine Seite zuzugreifen, dann wird er automatisch auf die Login-Seite umgeleitet. Nach erfolgreicher Anmeldung wird der Nutzer auf die Startseite weitergeleitet. Jede Session hat eine Ablaufzeit von zwei Stunden. Nach Ablauf der Zeit wir der Nutzer automatisch ausgeloggt, sodass er sich neu einloggen muss.

### Login / Registrierung

Ein Nutzer kann sich über das Registrierungsformular registrieren und über das Login-Formular einloggen. Auf diesen beiden Seiten werden die Sidebar und der Header mit Ausnahme des Logos ausgeblendet, sodass nur eingeloggte Nutzer diese Funktionen nutzen können. Bei der Registrierung werden Vor- und Nachname, sowie E-Mail und Passwort abgefragt. Außerdem muss angegeben werden, ob eine Person ein Experte ist oder nicht. Je nachdem hat der Nutzer verschiedene Funktionalitäten. Für die Registrierung wird ein Eintrag per INSERT-Befehl in der Datenbank angelegt, welcher im Login-Formular zur Authentifizierung verwendet wird. Nach erfolgreichem Login hat der Nutzer je nach Berechtigungsstufe (Azubi oder Experte) alle Funktionalitäten zur Verfügung.

# Verteilungssicht

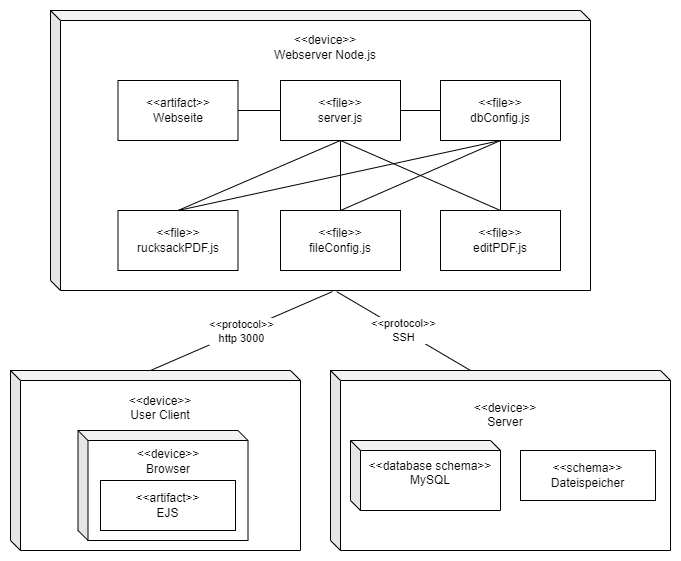


Abbildung 18: Verteilungssicht

|  |  |
| --- | --- |
| **Node / artifact** | **Beschreibung** |
| Webserver Node.js | Dient zur Schnittstellen-Implementierung. |
| User Client | Verantwortlich für die Anfragen und Daten zwischen dem Endgerät und dem Server, und dient als Darstellung auf den Endgeräten. |
| Server | Die vom Client gesendeten Anfragen werden vom Server empfangen und verarbeitet. |
| EJS | Einbettung von JavaScript-Elementen in HTML-Seiten. |
| Webseite | Wie Inhalte auf Geräten und Netzwerken verteilt werden. |
| Browser | Ein aktueller Browser zum Zugreifen auf einzelne EJS-Seiten. Alle herkömmlichen Browser (wie Chrome, Firefox, Safari etc…) sind dazu im Stande. |

Tabelle 8: Beschreibung der Verteilungssicht

# Konzepte

## Konzeption der Webseite mit Balsamiq Wireframes

Bereits in der Konzeptionsphase des Projekts wurde mit dem Projektbetreuer darüber gesprochen, welche Tools sich anbieten um einen Design-Prototyp zu entwerfen und diesen dann auch bei neuen Ideen auszubauen bzw. zu verändern. Es wurde sich schließlich für das Tool „Balsamiq“ entschieden (<https://balsamiq.com/>), da es den Bedürfnissen in den Punkten Einfachheit und Schnelligkeit entspricht.

Über einige Wochen wurden mit Balsamiq Ideen zunächst als grobe Entwürfe erstellt, diese dann weiterhin diskutiert und schließlich in echten Code umgesetzt. Oft wurden so einzelne Unterseiten erweitert oder abgeändert.

Das Tool hilft sehr beim schnellen Besprechen einer Design-Idee und reicht vollkommen aus als optische Vorlage für die Entwicklung der HTML/CSS Seite.

## Profilseite (Frontend) + PW ändern Feature

Für die sozialen Aspekte, für die administrative Verwaltung und für einige Kernfunktionen der Lernplattform werden User-Accounts gebraucht. Es wurde sich für ein User-System entschieden, welches zwischen Experten (Ausbildern) und Azubis unterscheidet.

Um das eigene Profil (Kontaktdaten) und um die eigene Zuweisung zu Themenbereichen zu bearbeiten, wurde eine Profilseite erstellt. Über die Zuweisung von Themen („Meine Themen“) teilt der Experte/Ausbilder dem System mit, in welchen Themengebieten er sich besonders gut auskennt und kann somit möglicherweise als Ansprechpartner von Azubis gefunden und kontaktiert werden. Des Weiteren wurde hier auch eine Passwort-ändern-Funktion eingebaut.

Auf der technischen Ebene werden hier die Inhalte über ein MySQL SELECT (Tabelle „Nutzer) Befehl ermittelt und die Bearbeitung der eigenen Themen erfolgt über INSERT INTO (bei hinzufügen neuer Themen) und über DELETE (beim Entfernen von Themen) bei der Tabelle „NutzerThema“.

## Dashboard / Fortschritts-Seite

Das Dashboard wurde im Laufe des Projekts mehrfach neu konzipiert, da es zwischenzeitlich Überlappungen mit anderen Features der Platform gab.

Die Vorgabe für das Festhalten eines Lernfortschrittes stand beim Dashboard immer im Vordergrund, jedoch hat sich die Idee des Fortschrittes mit der Einführung des Rucksacks geändert.

Am Ende wurde sich für ein Themen und Rucksack-basiertes Einschätzungs-System entschlossen:

* + - * Azubis können sich pro Rucksack und pro Thema selbst einschätzen mit einer Prozent-Angabe zwischen 0 und 100.
      * Experten sehen im Dashboard für jeden Azubi die jeweilige Selbsteinschätzung und können dazu noch eine Expertenseitige Schätzung (auch 0-100%) abgeben.

Auf der technischen Ebene passieren hier hauptsächlich MySQL SELECT und UPDATE TABLE - Befehle an der Datenbank. Es wird hier in die Tabelle „NutzerThema“ jeweils eine Azubi- und eine Experten- Bewertung gespeichert. Die Namen der Themen werden aus der Tabelle „Thema“ genommen.

## Fehlermeldungen/Statusmeldungen

Für eine gute User-Experience ist Benutzer-Feedback des Systems wichtig. Es wurden also für einige Features, bzw. Vorgänge des Systems Statusmeldungen in Form von roten (Fehler) oder grünen (Erfolg) Banner eingebaut.

## Datenbank

Die Datenbank wurde in der dritten Normalform erstellt. Das bedeutet, dass kein Nichtschlüsselattribut transitiv von einem Kandidatenschlüssel abhängig ist und die zweite und erste Normalform erfüllt ist (Nichtschlüsselattribute sind abhängig von jedem Schlüsselkandidaten und Wertebereiche der Attribute des Relationstypen sind atomar. Auf diese Weise hat die Datenbank eine gute Balance aus Redundanz, Performance und Flexibilität. (Kleinere und nicht beeinträchtigende Fehler haben sich vereinzelnd in die Datenbank eingeschlichen. Diese haben jedoch keinen Effekt auf das Projekt.)  
Die Umsetzung der Regeln in der Datenbank sind in folgenden ER-Diagramm ersichtlich.

Ein Bild, das Text, Schild, Screenshot, mehrere enthält.

Automatisch generierte Beschreibung

Abbildung 19: Datenbankdiagramm

Alle Relevanten Daten werden in der entsprechenden Tabelle möglichst vollständig eingetragen. Zwischen diesen Tabellen sind Verbindungstabellen (meist benannt nach den zu verbindenden Tabellen) um „n zu m“ Beziehungen aufzulösen. Fünf Tabellen bilden dabei eine Ausnahme. „Typ“ ist eine erweiternde Tabelle zu Material, in der die möglichen unterschiedlichen Typen von Materialien vordefiniert werden können. Dies wurde in dieser Form erstellt, damit ein einfacher Überblick über alle vorhandenen Typen möglich ist und die Bearbeitung erleichtert wird.  
In der Tabelle „Farbe“ werden noch freie Farbcodes für Überthemen gespeichert (genaueres dazu in der Beschreibung der Themen Seite). Auf diese Weise wird garantiert, dass Überthemen und deren Unterthemen Farblich differenziert werden können.

„InhaltVideo“, „InhaltPDF“ und „InhaltBuch“ sind ähnlich wie „Typ“ erweiternde Tabellen. Informationen, welche nur für spezifische Inhalte gespeichert werden sollen, können in diesen Tabellen eingetragen werden.

## Programmierung

### Thema

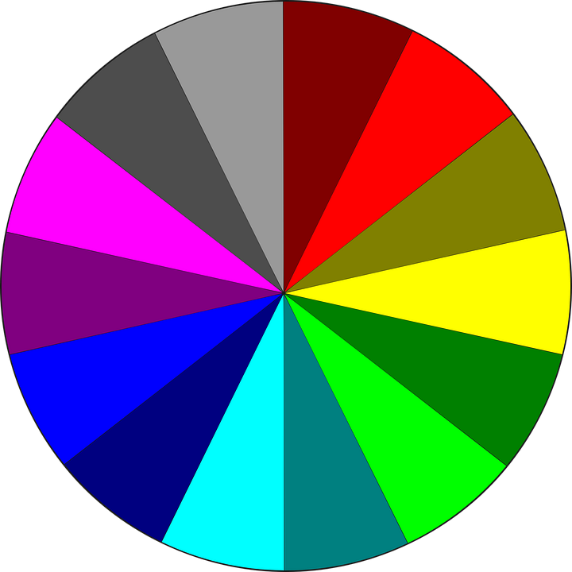


Abbildung 20: Skizze des Farbdiagramms

Themen dienen dazu Inhalte zu kategorisieren. Auf diese Weise ist es möglich Experten die Möglichkeit zu geben sich selbst Themenbereiche zuzuordnen. Sollten Azubis eine Frage zu einem Thema haben, dann können sie damit herausfinden, welcher Experte mit hoher Sicherheit behilflich sein kann.

Zusätzlich können Inhalte nicht nur durch ihren eigenen Namen, sondern auch mithilfe der zugewiesenen Themen gesucht werden. Um die Übersicht zu erleichtern, wurde sich dazu entschieden Überthemen und Unterthemen zu erstellen. Ein Überthema beschreibt in diesem Fall den Bereich, in dem sich dieses befindet (wie zum Beispiel „Programmierung“). Unterthemen hingegen gehen spezifischer auf den Inhalt ein (wie zum Beispiel „Java, PHP oder C++“). Um dieses System zu erleichtern und für die Nutzer leicht erkenntlich darzustellen, erhalten Überthemen einen eigenen Farbbereich, in dem sich das Überthema und alle dessen Unterthemen angezeigt werden. Um die Integrität des Farbsystems garantieren zu können wird eine Farbe so lange blockiert, bis ein Überthema zusammen mit allen dessen Unterthemen gelöscht wird.

Dies limitiert die Anzahl an Überthemen, die gleichzeitig auf die Anzahl der Farbcodes erstellt werden können. Dies kann jedoch durch das Hinzufügen neuer Farbcodes für Überthemen in der Datenbank „Farbe“ und kleine Abänderungen des Farb-Skripts erweitert werden.

Die Übersicht der Themen wird in einem Kuchendiagramm mithilfe der Vorlage „Zoomable Sunburst“ aus der D3 JavaScript-Bibliothek dargestellt. Damit werden die Themen grafisch dargestellt. Dabei wird für jedes Thema ein eigener Slice erstellt, der mit der jeweiligen Farbe aus der Datenbank gefüllt wird und bei einem Klick die Suche mit dem jeweiligen Thema startet. Ein weiterer Vorteil war, dass man durch D3 die Erstellung des Diagramms dynamisch erfolgen lassen kann, da eine JSON dafür verwendet wird, welche vorher durch die Datenbankeinträge erzeugt wird. Das heißt, es würde zu keinen Komplikationen führen, wenn noch Themen eingefügt oder gelöscht werden, weil die Erzeugung bei jedem Aufruf stattfindet.

### Rucksack

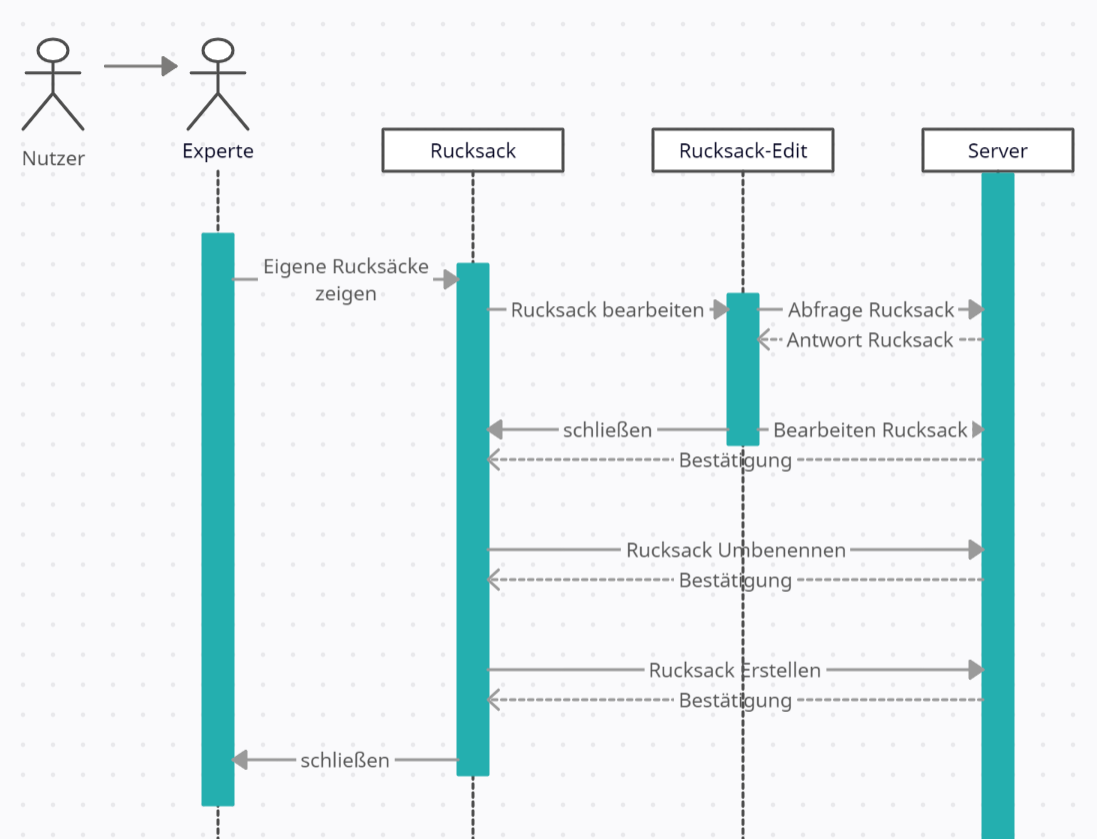


Abbildung 21: Sequenzdiagramm für die Verwaltung des Rucksacks

Rucksäcke dienen dazu eine Sammlung von ausgewählten Lerninhalten speziell für einzelne oder eine Gruppe von Azubis zuschneiden zu können.

Wie schon in SZ1 beschrieben, können Experten (AK1) zusätzlich zu der PDF-Generierung Lerninhalte zusammenstellen. Rucksäcke haben den Vorteil modular aufgebaut zu sein, wodurch das hinzufügen oder entfernen von Inhalten und Nutzern, die diesen Rucksack nutzen können, leicht gestalten zu können.

Die Experten-Oberfläche gestattet aus diesem Grund zusätzlich zum Erstellen, Löschen und Umbenennen von Rucksäcken, auch für selbst erstellte Rucksäcke eine weitere Bearbeitungsoberfläche. Dort finden Experten auch Informationen über den selbst eingeschätzten Fortschritt der Azubis zu diesem Rucksack.

Die Funktionen Löschen und Erstellen funktionieren ähnlich wie auf anderen Seiten wie Thema, Inhalt und Material.

Die Erweiterte Rucksack Oberfläche ermöglicht es dem Experten durch Auswahl eines Elements, dessen ID an den Server zu senden, damit diese in der Datenbank als Verbindung zwischen Rucksack und deren Inhalte und Nutzer verknüpft werden.

### Inhalte erstellen

Auf der Seite Inhalte können neue Dateien hochgeladen werden. Dafür wird eine Form verwendet, welches die Daten für einen neuen Inhalt übergibt. Für das Aussehen wird hier das Form-Style von Bootstrap verwendet. Außerdem sind der Material- und Themen-Picker ebenfalls eine Funktion aus Bootstrap und wurden verwendet, um zum einen die Möglichkeit haben nach einem Thema oder Material aus der Datenbank mittels einer live-search zu suchen. Zum anderen, um einen Multiselektor zu haben, der es möglich macht einem Thema mehrere Themen zuzuteilen, falls das nötig ist. Die Eingabefelder für die Inhaltserstellung werden je nach ausgewähltem Material angepasst, da beispielsweise eine PDF andere Infos als ein Video benötigt. Inhalte werden verwendet, sodass man ein Material in verschiedene Bereiche unterteilen kann, weil längere Videos durchaus mehrere Themen zu verschiedenen Zeitpunkten anschneiden könnten, und macht es so einfacher bestimmte Information zu finden.

## Webanwendung starten

Um die Webanwendung zu starten, müssen folgende Schritte durchgeführt werden:

1. SSH-Keypair erstellen
   1. Public-Key auf dem Server in der Datei „./root/.ssh/authorized\_keys“ anfügen
   2. Private Key im Ordner „webserver/ssh-key/“ mit dem Namen „id\_rsa“ ablegen
2. Die Datei „.env“ mit den Umgebungsvariablen im Ordner „webserver/“ ablegen
3. Über VPN mit dem Hochschulnetz verbinden oder im Hochschulnetz sein
4. Node.js installieren
5. Den Befehl „npm install“ in der Konsole ausführen

Diese Schritte müssen einmalig zur Einrichtung durchgeführt werden. Anschließend kann der Server jedes Mal mit Schritt 6 gestartet werden.

1. Den Befehl „cd webserver; node server.js“ ausführen

# Architekturentscheidungen

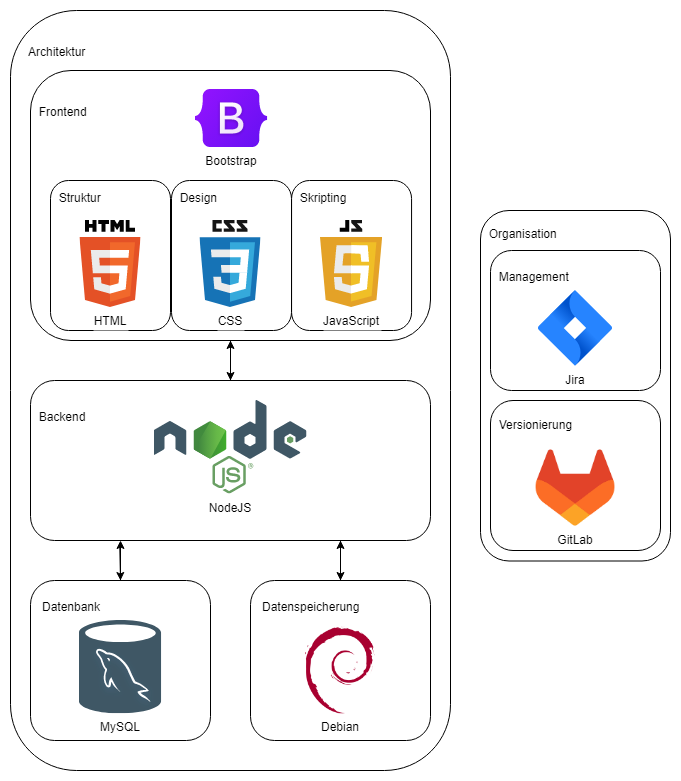


Abbildung 22: Darstellung der Architektur

|  |  |
| --- | --- |
| Technologie | Funktionalität |
| Bootstrap | Framework für Benutzeroberfläche |
| HTML | Strucktur Benutzeroberfläche |
| CSS | Gestaltung Benutzeroberfläche |
| JavaScript | Skripting und dynamisches Laden der Benutzeroberfläche |
| NodeJS | Webserver für Backend, Schnittstelle zu Datenbank und Dateispeicher |
| MySQL | Realtionales DBMS |
| Debian | Dateispeicher und Datenbank-Host |

Tabelle 9: Beschreibung der Architektur

Aufgrund der Vorkenntnisse der Entwickler, wurde für das Backend die JavaScript-Laufzeitumgebung Node.js gewählt. Für Alternativen, wie bspw. Django in Python oder Spring Boot in Java hätte sich vorher eingelesen werden müssen, was zusätzliche Zeit in Anspruch genommen hätte. Neben diesem Hauptgrund bietet Node.js viele vorentwickelte Pakete an, welche dank des Paketmanagers npm einfach installiert und genutzt werden können.

Als Datenbank wurde sich für relationale Datenbank mit MySQL als DB-Verwaltungssystem (DBMS) entschieden. Für die Ausmaße dieses Projektes machen die Unterschiede der DBMS keinen Unterschied, weswegen das populärste Produkt gewählt wurde. Es wurde eine relationale DB und keine dokumentorientierte DB genommen, weil relationale DBs weniger komplex sind und eine höhere Konsistenz gegenüber dokumentorientieren DBs vorweisen.

Mithilfe der Template-Sprache Embedded JavaScript (EJS) ist es möglich Daten aus der Datenbank serverseitig auf der Webseite anzuzeigen. Die Daten werden von Node.js an EJS übergeben und vor dem Laden der Webseite an die passenden Stellen des HTML gesetzt. Dementsprechend ist es möglich mithilfe von EJS Bedingungen oder Schleifen in den HTML-Code einzubauen.

Alle Pakete wurden aufgrund ihrer Popularität und Einfachheit in ihrer Anwendung gewählt. Dadurch konnten sich die Entwickler vergleichsweise schnell in die Techniken einlesen und hatten eine große Online-Community für Unterstützung und Nachfragen bei Fehlern oder Problemen.

# Manuelles Testing

Legende:

P = Positiv, N = Negativ, E = Ergebnis, S = Seite

T = Themen, R = Rucksack, I = Inhalt, M = Material, PG = PDF-Generierung, F = Fortschritt

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Fall** | **S** | **Anwendung** | **Testbeschreibung** | **Testziel** | **E** |
| 1. | T | Kuchendiagramm | Prüfung, ob eine Themenauswahl möglich ist. | Tester kann auf die T zugreifen und auswählen. | P |
| 2. | T | Themenbereich erstellen | Prüfung, ob ein Themenbereich erstellt werden kann. | Tester kann ein Themenbereich erstellen. | P |
| 3. | T | Themenbereich erstellen | Prüfung, ob der Name bei einer Themenbereich Erstellung leer sein. | Tester kann Themenbereich erstellen, wenn der Name nicht ausgefüllt ist. | N |
| 4. | T | Themenbereich löschen | Prüfung, ob ein Themenbereich gelöscht werden kann. | Tester kann vorhandenen Themenbereich löschen:  Hinweis vor Löschvorgang: Warnung! Folgende Inhalte werden entfernt:   * Sämtliche Unterthemen. | P |
| 5. | T | T hinzufügen | Prüfung, ob ein T in einen Themenbereich hinzugefügt werden kann. | Tester kann ein T einem Themenbereich hinzufügen (BSP: Themenbereich: Softwaretechnik wird mit einem Unterthema: UML-Diagramm erweitert).  Hinweis nach Hinzufügen:   * Neues T hinzugefügt. | P |
| 6. | T | T löschen | Prüfung, ob T in Themenbereichen gelöscht werden kann. | Tester kann das erstelle T in einem Themenbereich löschen.  Hinweis nach Löschen:   * T entfernt. | P |
| 7. | R | R erstellen. | Prüfung, ob R ohne Angabe eines Namens erstellt werden kann. | Tester kann keinen R erstellen, wenn er keinen Namen angibt. Fehlermeldung:   * Rucksackname darf nicht leer sein! | N |
| 8. | R | R erstellen. | Prüfung, ob R mit einem Namen erstellt werden kann. | Tester wählt einen Namen für den zu erstellenden R aus und erstellt diesen.  Hinweis nach Erstellung:   * Neuer R angelegt. | P |
| 9. | R | R erstellen. | Prüfung, ob R gelöscht werden kann. | Tester kann den erstellten R löschen.  Hinweis vor Löschvorgang:  Warnung! Folgende Inhalte werden entfernt:   * Sämtliche Inhalte dieses R. * Der Fortschritt dieses R. * Sämtliche User-Verknüpfungen dieses R   Zwei Button erscheinen:   1. Abbrechen (Grau) 2. Ja, ich bin mir sicher (Rot)   Hinweis nach Löschvorgang:   * R gelöscht. | P |
| **Fall** | **S** | **Anwendung** | **Testbeschreibung** | **Testziel** | **E** |
| 10. | R | R umbenennen. | Prüfung, ob R umbenannt werden kann. | Tester kann R umbenennen.  Hinweis nach Umbenennung:   * R wurde umbenannt. | P |
| 11. | R | R bearbeiten. | Prüfung, ob R bearbeitet werden kann. | Tester kann den R bearbeiten. | P |
| 12. | R | R mit I befüllen. | Prüfung, ob der R mit Inhalten befüllt werden kann. | Tester kann den R mit Inhalten erweitern  Hinweis nach Hinzufügen von I:   * Inhalt zum R hinzugefügt. | P |
| 13. | R | I vom R löschen. | Prüfung, ob I einzeln oder ob der gesamte Inhalt gelöscht werden kann. | Tester kann einzelne I löschen.  Hinweis nach Löschvorgang einzelner I:   * I aus R entfernt.   Hinweis nach Löschvorgang einzelner I:   * Alle I des R entfernt. | P |
| 14. | R | R mit einem Nutzer teilen. | Prüfung, ob ein R einem Nutzer zugeteilt werden kann. | Tester kann den R einem Nutzer zuteilen.  Hinweis nach Zuteilung des R:   * Nutzer-Zugriff auf Rucksack hinzugefügt. | P |
| 15. | R | Nutzer können von R gelöscht werden. | Prüfung, ob Nutzer zu einem zugeteilten R entfernt werden kann. | Tester kann einzelne Nutzer löschen.  Hinweis nach Löschvorgang des Nutzers:   * Nutzer-Zugriff auf R entfernt   Tester kann alle zugeteilten Nutzer löschen.  Hinweis nach Löschvorgang mehrere Nutzer:   * Sämtliche Nutzer-Zugriffe entfernt. | P |
| 16. | R | Azubi kann R sehen. | Prüfung, ob der Azubi seinen zugeteilten R sehen kann. | Tester kann den zugeteilten R unter Rucksackseite einsehen und der R ist auf der Home-Seite ebenfalls aufgelistet, wenn der R dem Nutzer zugeteilt wurde. | P |
| 17. | R | Azubi kann R öffnen und bearbeiten. | Prüfung, ob der Azubi seinen zugeteilten R aufklappen und den I sehen kann, welches in 2 Kategorien aufgeteilt sind (Lernstoffe, Übung) | Tester kann dem zugewiesenen R aufklappen und den befüllten R bearbeiten und auch nach Stoffarten filtern. | P |
| 18. | R | Azubi sieht seinen Lernfortschritt | Prüfung, ob der Azubi seinen Lernfortschritt zu dem zugeteilten R sehen kann. | Tester kann Lernfortschritt einsehen. | P |
| 19. | PG | PDF erstellen. | Prüfung, ob ein unbenanntes PDF erstellt werden kann. | Tester kann keine PDF erstellen, ohne vorher einen Namen definiert zu haben.  Hinweis bei Erstellung einer PDF ohne Namen:   * Verweis auf das Feld mit Dateiname -> Füllen Sie das Feld hier aus. | N |
| 20. | PG | PDF vom eigenen Rechner aus erstellen. | Prüfung, ob eine PDF aus den PNGs von dem eigenen Rechner erstellt werden kann. | Tester kann von seinem Rechner aus mehrere PNGs auswählen und diese dann als PDF generieren lassen. Die generierte PDF wird in M gespeichert und kann von dort aus aufgerufen werden. | P |
| 21. | PG | PDF von PNGs aus den M erstellen. | Prüfung, ob eine PDF aus mehreren PNGs aus dem Material-Ordner erstellt werden. Mehrauswahl wird in einem Dropdown-Menü festgehalten. | Tester kann eine PDF generieren lassen, die aus mehreren PNGs, die auf der Seite verfügbar sind, erstellen. Diese werden in M angezeigt | P |
| **Fall** | **S** | **Anwendung** | **Testbeschreibung** | **Testziel** | **E** |
| 22 | M | Neue M können hinzugefügt werden. | Prüfung, ob eine Hinzufügung von M, abhängig ihres Formates, erstellt werden kann.  Materialtypen:   * PDF (.pdf) * Img (.png) * Video (mp3.) * Buch(https://example.com) * Website(<https://example.com>) * YouTube (https://www.youtube.com/watch?v=) | Tester kann das ausgewählte Format mit der passenden Datei hochladen. Die hochgeladene Datei wird anschließen bei M gespeichert. | P |
| 23. | M | Vorhandene M umbenennen. | Prüfung, ob die Umbenennung von bestehenden M möglich ist. | Tester kann bestehende M umbenennen. | P |
| 24. | M | Vorhandene M löschen. | Prüfung, ob das Löschen der vorhandenen M möglich ist. | Tester kann bestehende M löschen. | P |
| 25. | I | I ohne Angabe erstellen. | Prüfung, ob I ohne eine Eingabe des Dateinamens erstellt werden kann. | Tester kann keine I ohne Benennung erstellen.  Hinweis vor Erstellung eines unbenannten I:   * Verweis auf das Feld mit Dateiname -> Füllen Sie das Feld hier aus. | N |
| 26. | I | I erstellen. | Prüfung, ob I erstellt werden können. | Tester kann I erstellen und diese den einzelnen T gebieten zuteilen. | P |
| 27. | I | I umbenennen/löschen | Prüfung, ob vorhandene I umbenannt oder gelöscht werden kann. | Tester kann vorhandene Dateien umbenennen und diese auch löschen. | P |
| **Fall** | **S** | **Anwendung** | **Testbeschreibung** | **Testziel** | **E** |
| 28. | F | Azubi auswählen. | Prüfung, ob ein Azubi zur Wissens-Einschätzung ausgewählt werden kann. | Tester kann einen Azubi zur Wissens-Einschätzung auswählen.  Hinweis zum Auswählen eines Azubis:   * Verweis auf: Bitte Azubi links auswählen. | P |
| 29. | F | Azubi einschätzen. | Prüfung, ob ein Azubi mit seinen ausgewählten T eingeschätzt werden kann. | Tester kann den ausgewählten Azubi anhand einer Leiste (zwischen 0-100) einschätzen.  Hinweis nach Einschätzen des Azubis:   * Erfolgreich gespeichert. | P |
| 30. | F | Selbsteinschätzung vom Azubi | Prüfung, ob der Azubi eine Selbsteinschätzung zu den vorhandenen T tätigen kann. | Tester kann die Selbsteinschätzung vornehmen und jedem T durch eine Leiste (0-100%) einen Wert zuteilen.  Hinweis nach Selbsteinschätzen des Azubis:   * Erfolgreich gespeichert. | P |

Tabelle 10: Auflistung der Testfälle

# Fazit und Ausblick

Nachdem die Grundlagen mittels Hochschulserver und MySQL-DB vorhanden waren, konnten die vom Stakeholder vorgegebenen Hauptziele angegangen werden. Dabei wurde mit Node.js als Basis ein Webserver aufgesetzt, welcher sowohl als Schnittstelle zur Datenbank, als auch für das Backend der Webanwendung verwendet wurde. Dank bereits vorentwickelter Pakete für Node.js, wie beispielsweise Express oder EJS wurde die Entwicklung für das Entwickler-Team erleichtert. Nachdem im Frontend für Bootstrap als Design-Vorlage gewählt wurde, konnten des Projekts angegangen werden (siehe Kapitel 1.3). Im erstellten Prototypen der Webanwendung wurden die drei Hauptanforderungen umgesetzt.

Darüber hinaus gab es noch einige Ideen und Ziele beim Abschluss des Projekts, welche z.T. aus zeitlichen Gründen nicht umgesetzt werden konnten:

|  |  |
| --- | --- |
| Priorität | Beschreibung |
| 1 |  |

* Möglichkeit für eine Mehrfachauswahl beim Hinzufügen von Inhalten und Nutzer
* Nachträgliche Anpassung von den Themen der Inhalte
* Überthemen umbennenbar machen
* Themen zu Lehrjahren unterordnen (zu mehreren wenn nötig)
* Rucksack-Icon für jeden Rucksack im Rucksack-Tab
* Optisch: Highlighting des zuletzt geklickten Tabs in der Sidebar
* Statusmeldungen (Erfolge und Fehler) plattformweit einbauen (fehlt bei ein paar Seiten)
* Generell mehr und ausführlichere Kommentare im Code
* HTTP-Error Seiten(404, 501, ...)
* Webseite über öffentliche Domain verfügbar machen
* Inhalte von der Inhaltsseite in Rucksack anfügen
* Dropdown unter Materialien auf Materialseite mit dazugehörigen Inhalten
* Materialname neben Inahlt auf Inhaltsseite
* Fehler-/ Erfolgsmeldungen für alle Seiten
* mehr Testen
* zusätzlicher Einsatz von SCSS/SASS bzw. JS-Framework (React, Vue, ...)
* PDF-Generierung aus verschiedenen Dateitypen, also nicht nur PNG, evtl. auch JPG und PNG gleichzeitig, oder Textdateien und Bilder, ...
* Bilder aus PDFs erzeugen
* Dropdown bei Eingabe in Suchfeld
* Suche erweitern, sodass nicht nur der Inhaltsname gesucht wird, sondern evtl. auch die Beschreibung oder der Typ (PDF, IMG, ...)
* Vorschau der Inhalte auf Inhaltsseite (ähnlich Suche, Rucksack)
* DB-Tabellen anpassen, sodass ISBN nicht bei jedem Inhalt eines Buches eingegeben wird, sondern einmal bei Materialien
* "Tutorial" für Erst-Nutzer/neu registrierte Nutzer
* Option, bei der Materialerstellung automatisch Inhalte zu erstellen (z.B. bei PDF direkt Seiten als Inhalt festlegen)
* Bildliche Roadmap für Azubis, als Erweiterung zu den Rucksäcken, so eine Art Rucksacksammlung
* Belohnungssystem für erfolgreich absolvierte Rucksäcke
* als ganz wilde Ideen auf KI-Basis:
* einfügen eines Chat-Bots, der einfache Fragen beantworten kann
* Text-Erkennung, die Text aus PDFs/Prüfungen erkennt und als Fragen in DB schreibt
* ISBN-Nummer in material speichern, statt bei Inhalt
* Beim Hochladen von Materialienkeine Sonderzeichen in Namen erlauben  
  (um Fehler in der Datenablage zu vermeiden)
* evtl. Ablage in Form von IDs statt Name
* Fehler beim PDF-Gen: Materialien Auswahl macht den Bildschirm immer breiter je mehr Materialien ausgewählt sind.
* Beim Rucksack-Edit: Rucksack Überschrift als Name statt ID
* Effizienz: Bei manchen Features werden immer alle Daten aus der DB geladen anstatt nur das nötige nach einem Mausklick
* Optisch: Bei Suchergebnis-Seite: Manchmal längeres Laden, das Skript der Schriftfarbe wird erst am Ende des Ladevorgangs umgesetzt

# Anhang

## Glossar

|  |  |
| --- | --- |
| **Begriff** | **Definition** |
| NodeJS | Die plattformunabhängige Laufzeitzeitumgebung basiert auf die Skriptsprache JavaScript und unterstützt Serverseitige Anwendungen für die Aufbau mit Datenbanken und Webserver. |
| EJS | EJS steht für Embedded JavaScript System, um dynamische Webseiten zu erstellen. Es verknüpft HTML und JavaScript miteinander und in Verbindung zu NodeJS verwendet werden. |
| JIRA | JIRA ist ein Projektmanagement, welche auf einer webbasierte Benutzeroberfläche beruht. Es dient zur Planung, Erstellung, Bearbeitung sowie die Organisation von Projekten bereit. |
| DBeaver | Das Tool ist ein Datenbankmanager zum Erstellen von Datenbankmodellen. Mithilfe davon können verschiedene Abfragen erstellt und verwaltet werden. |
| Balsamiq | Mithilfe der Software kann ein webbasierte Mockup-Tool in Form von Wireframes erstellt werden. |
| CRUD | CRUD beschreibt die Manipulationsmöglichkeiten CREATE, READ, UPDATE und DELETE von Daten |

## Abbildungsliste

[Abbildung 1: Stakeholder 4](#_Toc126327408)

[Abbildung 2: Use-Case Diagramm für die Organisation von Informationen und Prüfungen 6](#_Toc126327409)

[Abbildung 3: Use-Case Diagramm für die Zusammenstellung von Übungseinheiten 6](#_Toc126327410)

[Abbildung 4: Use-Case Diagramm für die Dokumentation des Lernfortschritts 7](#_Toc126327411)

[Abbildung 5: Persona Experte 7](#_Toc126327412)

[Abbildung 6: Persona Azubi 8](#_Toc126327413)

[Abbildung 7: Aktivitätsdiagramm für das Erstellen eines Rucksack 9](#_Toc126327414)

[Abbildung 8: Aktivitätsdiagramm für das Erstellen von Themen 10](#_Toc126327415)

[Abbildung 9: Aktivitätsdiagramm für die Erstellung von Inhalten 11](#_Toc126327416)

[Abbildung 10: Aktivitätsdiagramm für das Einsehen und Bearbeiten eines Rucksacks 12](#_Toc126327417)

[Abbildung 11: Aktivitätsdiagramm für die Selbsteinschätzung 13](#_Toc126327418)

[Abbildung 12: Meilensteine / Roadmap 17](#_Toc126327419)

[Abbildung 13: Kontextdiagramm 19](#_Toc126327420)

[Abbildung 14: Bausteinsicht 22](#_Toc126327421)

[Abbildung 15: Datei-Management 25](#_Toc126327422)

[Abbildung 16: Sequenzdiagramm für die PDF-Erstellung über Rucksäcke 26](#_Toc126327423)

[Abbildung 17: Sequenzdiagramm für die Erstellung einer PDF über die PDF-Gen-Seite 27](#_Toc126327424)

[Abbildung 18: Verteilungssicht 31](#_Toc126327425)

[Abbildung 19: Datenbankdiagramm 34](#_Toc126327426)

[Abbildung 20: Skizze des Farbdiagramms 36](#_Toc126327427)

[Abbildung 21: Sequenzdiagramm für die Verwaltung des Rucksacks 37](#_Toc126327428)

[Abbildung 22: Darstellung der Architektur 39](#_Toc126327429)

## Tabellenliste

[Tabelle 1: Stakeholder 5](#_Toc126327330)

[Tabelle 2: Nicht-Funktionale Anforderungen 15](#_Toc126327331)

[Tabelle 3: Teamplanung 17](#_Toc126327332)

[Tabelle 4: Beschreibung der Meilensteine 18](#_Toc126327333)

[Tabelle 5: Beschreibung des Kontextdiagramms 20](#_Toc126327334)

[Tabelle 6: Ebene 1 der Bausteinsicht 22](#_Toc126327335)

[Tabelle 7: Ebene 2 der Bausteinsicht 23](#_Toc126327336)

[Tabelle 8: Beschreibung der Verteilungssicht 32](#_Toc126327337)

[Tabelle 9: Beschreibung der Architektur 40](#_Toc126327338)

[Tabelle 10: Auflistung der Testfälle 47](#_Toc126327339)