**IPA-Dokumentation**

**Erweiterung des Competec Lernportals**

Ein Bild, das Text, Whiteboard, Handschrift, Im Haus enthält.

Automatisch generierte Beschreibung

Inhaltsverzeichnis

[Allgemeine Informationen 4](#_Toc160118568)

[Versionierung Dokumentation 4](#_Toc160118569)

[Versionierung Praktische Arbeit 4](#_Toc160118570)

[Zusatzinformationen 4](#_Toc160118571)

[Teil 1 – Umfeld und Ablauf 5](#_Toc160118572)

[Detaillierte Aufgabenstellung 5](#_Toc160118573)

[Titel der Arbeit 5](#_Toc160118574)

[Ausgangslage 5](#_Toc160118575)

[Teilauftrag 1: Login und Benutzerverwaltung 6](#_Toc160118576)

[Teilauftrag 2: Bilderverwaltung 7](#_Toc160118577)

[Teilauftrag 3: Fehlerbehandlung des Datenimports 7](#_Toc160118578)

[Mittel und Methoden 8](#_Toc160118579)

[Deklaration der Vorkenntnisse 9](#_Toc160118580)

[Deklaration der Vorarbeiten 9](#_Toc160118581)

[Neue Lerninhalte 9](#_Toc160118582)

[Arbeiten in den letzten 6 Monaten 10](#_Toc160118583)

[Deklaration der benutzten Firmenstandards 10](#_Toc160118584)

[Projektaufbauorganisation 11](#_Toc160118585)

[Projektmethode IPERKA 12](#_Toc160118586)

[Einfluss konkret auf meine IPA: 12](#_Toc160118587)

[Zeitplan 13](#_Toc160118588)

[Arbeitspakete 14](#_Toc160118589)

[Organisation der Arbeitsergebnisse 17](#_Toc160118590)

[Versionierung 17](#_Toc160118591)

[Backup 17](#_Toc160118592)

[Quellcode / Skripts 17](#_Toc160118593)

[Arbeitsplatz 17](#_Toc160118594)

[Arbeitsjournal 18](#_Toc160118595)

[Tag 1 – 29.02.2024 18](#_Toc160118596)

[Tag 2 – 04.03.2024 19](#_Toc160118597)

[Tag 3 – 05.03.2024 20](#_Toc160118598)

[Tag 4 – 06.03.2024 21](#_Toc160118599)

[Tag 5 – 07.03.2024 22](#_Toc160118600)

[Tag 6 – 11.03.2024 23](#_Toc160118601)

[Tag 7 – 12.03.2024 24](#_Toc160118602)

[Tag 8 – 13.03.2024 25](#_Toc160118603)

[Tag 9 – 14.03.2024 26](#_Toc160118604)

[Tag 10 – 18.03.2024 27](#_Toc160118605)

[Teil 2 – Projekt 28](#_Toc160118606)

[Kurzfassung 28](#_Toc160118607)

[Ausgangslage 28](#_Toc160118608)

[Umsetzung 28](#_Toc160118609)

[Ergebnis 28](#_Toc160118610)

[Projektbeschreibung 29](#_Toc160118611)

[Phase "Informieren" 30](#_Toc160118612)

# 

# Allgemeine Informationen

## Versionierung Dokumentation

|  |  |  |
| --- | --- | --- |
| Version | Datum | Branch |
| 1 | 29.02.2024 | https://github.com/2David4/IPA-Dokumentation/tree/ipa-day1 |
| 2 | 04.03.2024 |  |
| 3 | 05.03.2024 |  |
| 4 | 06.03.2024 |  |
| 5 | 07.03.2024 |  |
| 6 | 11.03.2024 |  |
| 7 | 12.03.2024 |  |
| 8 | 13.03.2024 |  |
| 9 | 14.02.2024 |  |
| 10 | 18.03.2024 |  |

## Versionierung Praktische Arbeit

|  |  |
| --- | --- |
| Projekt | Branch |
| Competec/ita-talent-api | https://github.com/Competec/ita-talent-api/tree/ipa-du |
| Competec/ita-talent-client | https://github.com/Competec/ita-talent-client/tree/ipa-du |

## Zusatzinformationen

|  |  |
| --- | --- |
| Dokumenttitel | Dokumentation |
| Autor | David Unterguggenberger |
| Dateiname |  |
| Ablageort | Lokaler Speicher, Github, Google Drive |

# Teil 1 – Umfeld und Ablauf

## Detaillierte Aufgabenstellung

### Titel der Arbeit

Administrative Verwaltung des CompAcademy Lernportals (Personalentwicklung)

### Ausgangslage

Die Competec-Gruppe besteht aus den Handelsunternehmen Alltron AG, Jamei AG, Medidor AG, Schoch Vögtli AG und BRACK.CH AG, dem Logistikdienstleister Competec Logistik AG und der Competec Service AG, die alle zentralen Dienste übernimmt. Die Firmengruppe mit Hauptsitz in Mägenwil AG beschäftigt rund 1400 Mitarbeitende und bildet 77 Lernende in diversen Berufsbildern aus. Die Grundausbildung der IT-Lernenden geschieht in der internen IT-Academy. Als Teil der Ausbildung werden verschiedenen Applikationen für unsere Firmengruppe entwickelt. Die Lernenden können dabei von der Entwicklung über die Projektleitung bis zur Inbetriebnahme verschiedene Rollen in der Softwareentwicklung einnehmen.

Die interne Personalentwicklung (im weiteren als "CompAcademy" bezeichnet) schreibt Kursangebote aus, auf welche sich die Mitarbeiter der Competec anmelden können. Die Verwaltung der Kurse geschieht auf der Talent Management Lösung von Umantis AG (im weiteren als "LMS" bezeichnet). Aufgrund der Benutzerunfreundlichkeit des LMS wurde die IT-Academy beauftragt eine visuell ansprechendere Lösung zu entwickeln. Daraus entstand die Plattform "CompAcademy Lernportal" (im weiteren als "Lernportal" bezeichnet), welche die Daten des LMS in regelmässigen Zeitabständen abfragt und benutzerfreundlich aufbereitet. Aufgrund von organisatorischen Schwierigkeiten wurde eine vereinfachte Variante der Anforderungen produktiv aufgeschaltet. Nun sollen die verbleibenden Features in einem Update umgesetzt werden.

Aktueller Stand

==============

Das Lernportal besteht aus einem Java Backend, welches mittels Spring Boot implementiert wurde. Dazu wurde das UI mit React implementiert, welches statisch bereitgestellt wird. Die Kommunikation zwischen den Systemen wird mit einer Websocket Verbindung sichergestellt.

Im Lernportal wurden bereits folgende Funktionalitäten implementiert.

Automatischer Datenimport

-------------------------------------

Im LMS wurde ein automatischer Export erstellt, welcher alle Kurse zur Verfügung stellt. Dieser funktioniert über einen REST-Endpunkt, welcher die Daten im XML-Format zurückgibt.

Diese Schnittstelle wird vom Lernportal im 10 Minuten Takt angefragt. Die Kursdaten werden anschliessend konvertiert, normalisiert und in einer PostgreSQL Datenbank abgespeichert. Gleichzeitig wird aus dem Titel und der Beschreibung der Kurse ein Suchindex erstellt.

Webseite / UI

------------------

Die Website besteht aus den folgenden Ansichten

• Suchresultat Seite

• Kurskatalog aufgeteilt auf Kurskategorien (Für alle Mitarbeiter, Für deine Fachrolle, Für alle Führungskräfte) mit Filtermöglichkeiten nach Typ, Organisation und Unterkategorien

• Detaillierte Kursansicht mit den Detailinformationen, Anmeldung und Teilveranstaltungen.

Mockup

-----------

Das Mockup der gesamten Webapplikation, einschliesslich der IPA-Erweiterungen, wurde bereits mit Figma erstellt. Dabei bildet es einen Richtwert und muss nicht pixelgenau übertragen werden.

Detaillierte Aufgabenstellung

Die Erweiterung des Lernportals ist in drei Teilaufträge gegliedert. Das Deployment ist nicht teil der IPA und muss nur lokal lauffähig sein.

### Teilauftrag 1: Login und Benutzerverwaltung

=======================================

Das Lernportal soll mit einem geschützten Administrationsbereich erweitert werden. Auf diesen kann nur mit einer gültigen Username- und Passwortkombination zugegriffen werden. Zusätzlich soll dafür eine Benutzerverwaltung erstellt werden.

Anforderungen

---------------------

• Ein Benutzer besteht mindestens aus den folgenden Datenpunkten

-- Benutzername

-- E-Mail

-- Passwort

-- Aktiv / inaktiv

• Der Benutzername muss eindeutig sein.

• E-Mails müssen validiert werden.

• Passwörter müssen mindestens 4 Zeichen lang sein.

• Deaktivierte Benutzer können sich nicht einloggen.

• Nach erfolgreichem Login soll der Benutzer 24 Stunden Zugriff auf die Applikation haben, bevor dieser sich erneut anmelden muss.

• Angemeldete Benutzer können sich ausloggen.

• Die administrativen Navigationselemente sind nur für eingeloggte Benutzer ersichtlich

• Die Benutzerverwaltung ermöglicht das Erstellen, Bearbeiten und Deaktivieren von Benutzer.

• Alle angemeldeten Benutzer haben die gleichen Berechtigungen

• Angemeldete Benutzer können alle Benutzer ohne Einschränkung bearbeiten (Es ist ein internes Tool ohne Hierarchie / Rollenkonzept ausser "Angemeldet oder nicht")

### Teilauftrag 2: Bilderverwaltung

===========================

Um die Kursangebote attraktiver zu gestalten, sollen die Kurse mit einem Bild versehen werden. Da im LMS keine Bilder erfasst werden können, muss die Verwaltung im Lernportal implementiert werden.

Anforderungen

---------------------

• Ein Bild besteht mindestens aus den folgenden Datenpunkten

-- Bild

-- Beschreibung

-- Kurs ID

• Die Bilderverwaltung ermöglicht das Erstellen, Bearbeiten und Löschen von Kursbildern.

• Wenn ein Kursbild vorhanden ist, wird dieses im Kurskatalog und der detaillierten Kursansicht dargestellt.

• Die Bilderverwaltung ist nur für eingeloggte Benutzer ersichtlich.

• Die Assoziation zwischen Bild und Kurs geschieht über die Kurs ID

• Bilder welche breiter als 1024px sind werden auf 1024px herunterskaliert.

• Folgende Bildformate werden unterstützt: PNG, JPG

• Bilder werden über eine REST-Schnittstelle angefragt, hochgeladen, angepasst und gelöscht.

• Die Bilder werden in der Datenbank abgespeichert.

• Die Bilddaten werden unabhängig von den Daten des Imports gespeichert.

• Wenn kein Kursbild vorhanden ist, wird nichts angezeigt.

### Teilauftrag 3: Fehlerbehandlung des Datenimports

=============================================

Beim Datenimport vom LMS in das Lernportal werden die Daten aufbereitet und normalisiert. Transformierungen werden aktuell nur in die Logdateien geschrieben. Damit diese Datenveränderungen nachvollzogen werden können, sollen diese den Administratoren sichtbar gemacht werden. Dadurch können Administratoren Erfassungsfehler im LMS-Quellsystem korrigieren.

Diese Transformierungen können diverse Ursachen haben. Dabei sind aktuell folgende bekannt:

• Fehlende Daten bei einem Pflichtfeld

-- Beispiel: Kurs hat keine Kategorie

-- Resultat: Standartwert wird angenommen

• Daten können nicht konvertiert werden

-- Beispiel: String anstatt Zahl

-- Resultat: Datenpunkt wird ignoriert

• Kurs referenziert einen nicht exportierten Kurs

-- Beispiel: Referenzierter Kurs wurde historisiert

-- Resultat: Kurs wird ignoriert

Zusätzlich soll bei jeden Datenimport der Zeitpunkt und Ausführungsstatus festgehalten werden.

Anforderungen

---------------------

• Transformierung des Datenimports werden in der Datenbank festgehalten.

• Eine Transformierung besteht mindestens aus den folgenden Datenpunkten

-- Kurs ID

-- Typ (Kritisch, Konvertierung, fehlende Daten)

-- Grund / Ursprung

-- Resultat

• Für jeden Datenimport werden die alten Transformationseinträge gelöscht

• Bei jeden Datenimport wird der Zeitpunkt und Status in der Datenbank festgehalten

• Ein Datenimport besteht mindestens aus den folgenden Datenpunkten

-- Zeitpunkt

-- Status (OK, Netzwerkfehler, Genereller Fehler)

• Die letzten 10 Datenimporteinträge sollen angezeigt werden.

### Mittel und Methoden

Versionsverwaltung

--------------------------

Der Quellcode soll auf dem Competec GitHub versioniert werden.

Dabei sind folgende Repositorien zu benutzen:

• ita\_talent\_api

• ita\_talent\_client

Code / Projektrichtlinien

--------------------------------

Der bestehende Projektaufbau soll übernommen und dem Auftrag entsprechend erweitert werden.

Style Guidelines

---------------------

Intellij Google Java Style Guide | https://github.com/google/styleguide/blob/gh-pages/intellij-java-google-style.xml

Teststrategie

-----------------

Um eine korrekte Ausführung zu gewährleisten, sollten alle Anforderungen manuell getestet werden. Dabei müssen alle Anforderungen validiert und abgedeckt werden. Dabei sind keine automatisierten Tests gefordert.

Umgebung

----------------

Firmen-Notebook, Windows 10, IntelliJ IDEA Ultimate, Figma, Git, PostgreSQL, Java 17, weitere Tools nach Bedarf

Technologien

------------------

Java 17, JavaScript, PostgreSQL, Spring Boot 3.1.x, React 18.2.x

Mockup

-----------

Figma Projekt im IT-Academy Team.

### Deklaration der Vorkenntnisse

In den vergangenen Monaten hat der Auszubildende folgende Vorkenntnisse erarbeitet.

• Backend

-- Java

-- Spring Boot

• Frontend

-- TypeScript

-- React

• Datenbank

-- PostgreSQL

• Protokoll / Format

-- Rest

-- XML

-- JSON

### Deklaration der Vorarbeiten

In einem Probeprojekt die Kenntnisse im Datenaustausch mit Websockets und programmatische Bildbearbeitung in Java vertiefen.

Das GitHub Projekt ist in einem sauberen Zustand und ein aktueller Branch für die IPA wurde erstellt.

### Neue Lerninhalte

Der Auszubildende wird keine neuen Lerninhalte erarbeiten.

### Arbeiten in den letzten 6 Monaten

IT-Academy Community

--------------------------------

Der Auszubildende war massgebend an dem Projekt "IT-Academy Community" beteiligt. Dabei handelt es sich um eine Webplattform bestehend aus einem React Frontend und Spring Boot Backend. In diesem Projekt arbeitete der Auszubildende an der Planung, Datenbankentwurf, Backend, Frontend und Deployment.

Tools/

-------

Das Projekt "Tools/" ist eine Webseite, welche alle aktiv intern genutzten Applikationen übersichtlich darstellt. Der Auszubildende übernahm in diesem Projekt den Lead und arbeitete an der Planung, Datenerhebung, Frontend und Deployment der Applikation.

Testprojekt

---------------

Zur Vorbereitung auf die IPA erstellte der Auszubildende ein Testprojekt. Dabei konnten alle fehlenden Kenntnisse erarbeitet werden. Um ein tiefes Verständnis zu erarbeiten, wurde die initiale Erstellung, Datenbankentwurf, Backend und Frontend von dem Auszubildenden umgesetzt.

## Deklaration der benutzten Firmenstandards

Im Rahmen meiner Individuellen Praktischen Arbeit (IPA) bei der Competec AG habe ich mich streng an die internen Firmenstandards gehalten, um sowohl die Konsistenz als auch die professionelle Qualität meiner Arbeit sicherzustellen.

Für die Erstellung der Dokumentation habe ich die unternehmenseigene Word-Vorlage genutzt. Diese Vorlage ist speziell darauf ausgelegt, die Lesbarkeit und Übersichtlichkeit von Berichten und Dokumenten zu optimieren und gleichzeitig das professionelle Erscheinungsbild von Competec zu wahren.

| Rolle | Name | Aufgabe und Verantwortung |
| --- | --- | --- |
| Kandidatin,  Kandidat | David Unterguggenberger | Die Anforderungen in der Aufgabenstellung  werden innerhalb der gegebenen Zeit vollständig  und korrekt erfüllt und umgesetzt. Die volle  Verantwortung wird übernommen |
| Berufsbildnerin,  Berufsbildner | Christian Walder | Kann verschiedene, unterstützende und administrative Aufgaben übernehmen, wertet jedoch die IPA-Arbeit nicht aus. |
| Verantwortliche Fachkraft | Flo Eberle | Entwickelt die Aufgabenstellung. Garantiert, dass die KAND während der IPA ungestört arbeiten können und beurteilt die erzielten Resultate der Arbeit. Ist verantwortlich für einen reibungslosen Ablauf der IPA |
| Hauptexpertin,  Hauptexperte | Jan Hohenheim | Besucht die Lehrfirma drei Mal und beurteilt die erbrachte Leistung und Qualität der Arbeit. Er ist für eine korrekte Umsetzung und Bewertung der IPA verantwortlich. Die Fachgespräche können auf Grund der Corona Pandemie auch vom Homeoffice aus durchgeführt werden. |
| Nebenexperte,  Nebenexpertin | Sara Zarubica | Ist für die korrekte Protokollierung der Präsentation, der Demo und des Fachgesprächs |

## Projektaufbauorganisation

## Projektmethode IPERKA

Die IPERKA-Methode ist ein strukturierter Ansatz für die Planung und Durchführung von Projekten. Die Abkürzung IPERKA steht für die sechs Phasen des Projektmanagements: Informieren, Planen, Entscheiden, Realisieren, Kontrollieren und Auswerten. Dieser methodische Rahmen bietet eine klare Richtlinie für die systematische Bearbeitung und Reflexion von Projektaufgaben.

### Einfluss konkret auf meine IPA:

|  |  |
| --- | --- |
| Phase | Aktivitäten |
| Informieren | Recherche über bestehende Lösungen für Login- und Benutzerverwaltungssysteme, Bilderverwaltung und Fehlerbehandlung bei Datenimporten. Analyse der Anforderungen und der Bewertungskriterien. |
| Planen | Entwicklung eines Konzepts für die Umsetzung der Aufgabenstellung, einschliesslich Datenbankmodell. Festlegung von Meilensteinen und Arbeitspakete. |
| Entscheiden | Festlegung der Architektur und Struktur der Datenbank, Auswahl spezifischer Methoden für die Fehlerbehandlung im Datenimport sowie Entscheidungen bezüglich der Implementierungsdetails für Login und Benutzerverwaltung, unter Berücksichtigung der Effizienz und Sicherheit. |
| Realisieren | Jetzt wird das geplante in die Realität umgesetzt! Implementierung der Benutzerverwaltung, einschliesslich Login-Funktionalität und Sicherheitsmassnahmen. Entwicklung der Bilderverwaltungsfunktionalität und Integration in das bestehende Lernportal. Umsetzung der Fehlerbehandlung für den Datenimport. |
| Kontrollieren | Durchführung von Tests zur Überprüfung der Funktionalitäten und Sicherheit der implementierten Features. Überwachung der Leistung und Anpassung basierend auf Testergebnissen. |
| Auswerten | Bewertung des Projektfortschritts und der erreichten Ergebnisse im Vergleich zu den ursprünglichen Zielen. Reflexion über Herausforderungen und Lernerfahrungen. |

## Zeitplan

## Arbeitspakete

| **ID** | **Inhalt** |
| --- | --- |
| 00 | Dokumentation der gesamten Projektarbeit, einschliesslich technischer Spezifikationen, Designentscheidungen und Implementierungsdetails. |
| 01 | Strukturierung und Planung der Dokumentationsinhalte. Hier wird festgelegt, welche Informationen wo dokumentiert werden, um eine kohärente und leicht navigierbare Dokumentation zu erstellen. Nach Wegleitung und Kriterienkatalog. |
| 02 | Überprüfung der Dokumentation auf sprachliche Korrektheit. Ziel ist es, eine professionelle und fehlerfreie Dokumentation zu gewährleisten. |
| 03 | Organisation und Durchführung von Treffen mit Expertenteam. |
| **Phase Informieren:** | |
| 04 | Analyse der Projektanforderungen und -ziele. Hier wird ein tiefes Verständnis der Aufgabenstellung entwickelt, um eine solide Basis für die Projektplanung zu schaffen. |
| 05 | Detaillierte Bewertung der Projektanforderungen anhand eines Kriterienkatalogs. Ziel ist es, Prioritäten zu setzen und sicherzustellen, dass alle Kriterien berücksichtigt werden. |
| 06 | Recherche und Sammlung von Informationen, die für die Umsetzung des Projekts relevant sind. Dies umfasst technische Daten, Branchentrends und Best Practices. |
| **Phase Planen:** | |
| 07 | Entwicklung eines detaillierten Projektzeitplans, der alle Phasen, Meilensteine und Fristen umfasst. Dieser Zeitplan dient als Wegweiser für die ganze IPA. | |
| 08 | Erstellung eines Konzepts, das den aktuellen Stand des Projekts und die Ausgangssituation beschreibt. Dies dient als Ausgangspunkt für die Entwicklung von Lösungsansätzen. | |
| 09 | Entwicklung eines umfassenden Konzepts für die Umsetzung der IPA aufgabenstellung. | |
| 10 | Entwurf eines Datenbankmodells, das die Speicherung und Organisation aller erforderlichen Daten unterstützt. Dieses Modell wird die Grundlage für die Entwicklung der neuen Tabellen sein. | |
| 11 | Ausarbeitung eines umfassenden Testplans, der sicherstellt, dass alle Aspekte der Aufgabenstellung gründlich getestet werden, um Fehler zu minimieren und die Qualität der Endprodukte zu gewährleisten. | |
| **Phase Entscheiden:** | | |
| 12 | Entscheidung für eine spezifische Lösungsvariante nach Bewertung aller möglichen Optionen. Diese Entscheidung basiert auf einer Analyse von Vor- und Nachteilen jeder Variante im Hinblick auf die Aufgabenstellung. | |
| **Phase Realisieren:** | |
| 13 | Implementierung der Datenbank gemäss dem zuvor entworfenen Modell. Dies umfasst die Einrichtung von Tabellen und deren Beziehungen. |
| 14 | Entwicklung von Mechanismen zur Erkennung und Handhabung von Fehlern während des Datenimportprozesses vom LMS entstehen. Ziel ist es, die Integrität der in die Datenbank eingespeisten Daten sicherzustellen. |
| 15 | Entwicklung von Backend-Funktionen zur Verwaltung von Bildern, einschliesslich Upload, skalierung, Speicherung, Bearbeitung und Löschung von Bildern gemäss der Aufgabenstellung. |
| 16 | Implementierung von Sicherheitsmechanismen für das Backend, die es Benutzern ermöglichen, sich sicher einzuloggen und auszuloggen, sowie die Verwaltung von Sessions. |
| 17 | Entwicklung von Backend-Funktionen zur Verwaltung von Benutzerkonten, einschliesslich Erstellung, Bearbeitung und Deaktivierung von Benutzern. |
| 18 | Zeitraum für letzte Anpassungen am Backend und Puffer für unvorhergesehene Aufgaben oder Verzögerungen. Ziel ist es, eine stabile und vollständig funktionierende Backend-Infrastruktur sicherzustellen. |
| 19 | Anpassung und Optimierung der Kommunikation zwischen Frontend und Backend, um eine reibungslose Datenübertragung und Interaktion zu gewährleisten. |
| 20 | Implementierung von Sicherheitsmechanismen im Frontend, um den Zugriff auf geschützte Bereiche der Anwendung zu kontrollieren, sowie die Einrichtung von Routing-Regeln für eine intuitive Navigation. |
| 21 | Entwicklung einer robusten Authentifizierungsinfrastruktur im Frontend, die es Benutzern ermöglicht, sich sicher anzumelden und Zugriff auf autorisierte Inhalte zu erhalten. |
| 22 | Erstellung von Frontend-Komponenten für Login- und Logout-Funktionalitäten, einschliesslich Benutzeroberflächen für die Eingabe von Anmeldedaten und die Verwaltung von Benutzersitzungen. |
| 23 | Implementierung einer Benutzeroberfläche und Logik für den Upload von Bildern durch den Benutzer, einschliesslich Feedbackmechanismen zur Anzeige des Upload-Status. |
| 24 | Entwicklung von Frontend-Komponenten zur Anzeige von Bildern zum jeweiligen Event. |
| 25 | Erstellung von Benutzeroberflächen und Logiken im Frontend zur Verwaltung von Bildern, einschliesslich Funktionen zum Erstellen, Bearbeiten und Löschen von Bildern. |
| 26 | Entwicklung einer Benutzeroberfläche zur Anzeige von Datenimport-Transformationen, die Administratoren Einblick in Probleme gibt, die während des Imports passiert sind. |
| 27 | Implementierung einer Komponente im Frontend zur Anzeige der letzten Datenimporte, einschliesslich Status und Zeitpunkt, um Administratoren einen Überblick über Importaktivitäten zu geben. |
| 28 | Entwicklung von Benutzeroberflächen und Funktionalitäten im Frontend zur Verwaltung von Benutzerkonten, einschliesslich der Erstellung, Bearbeitung und Deaktivierung von Benutzern. |
| 29 | Zeitraum für letzte Anpassungen im Frontend und Puffer für unvorhergesehene Aufgaben oder Probleme. Ziel ist es, ein benutzerfreundliches und fehlerfreies Frontend zu gewährleisten. |
| 30 | Das Verfahren umfasst die Erstellung eines Docker-Images der Anwendung, welches anschliessend auf einem internen Linux-Server bereitgestellt wird. Zusätzlich wird eine Aktualisierung der Datenbank auf dem Server durchgeführt. |
| **Phase Kontrollieren:** | |
| 31 | Entwicklung von Testfällen basierend auf den Projektanforderungen, um eine umfassende Überprüfung der Funktionalität und Leistung der Anwendung zu ermöglichen. |
| 32 | Durchführung der erstellten Testfälle, um die korrekte Funktionsweise der Anwendung sicherzustellen und potenzielle Fehler oder Probleme zu identifizieren. |
| **Phase Auswerten:** | |
| 33 | Bewertung der implementierten Benutzerverwaltungsfunktionen, um sicherzustellen, dass sie der Aufgabenstellung entsprechen und effektiv funktionieren. | |
| 34 | Überprüfung der Bilderverwaltungsfunktionen auf ihre Effektivität und Übereinstimmung mit der Aufgabenstellung, einschliesslich der Benutzerfreundlichkeit und der Qualität der Bildbearbeitung. | |
| 35 | Analyse der Fehlerbehandlungsmechanismen, um sicherzustellen, dass Fehler effektiv erkannt und verarbeitet werden, insbesondere im Kontext des Datenimports. | |
| 36 | Rückblick auf den gesamten IPA-verlauf, um Lernerfahrungen, Herausforderungen und Erfolge zu identifizieren. Dies dient der persönlichen Entwicklung und der Verbesserung zukünftiger Projekte. | |
| 37 | Zusammenfassung der Projektergebnisse, Erkenntnisse und des erreichten Fortschritts im Vergleich zu den ursprünglichen Zielen. Bietet einen abschliessenden Überblick über die IPA. | |

## Meilensteine

| **Meilenstein** | **Geplanter Zeitpunkt** | **Bemerkungen** |
| --- | --- | --- |
| 12 | Mo 04.03.2024  17 Uhr | Verweis Arbeitspaket 12 |
| 18 | Mi 06.03.2024  15 Uhr | Verweis Arbeitspaket 18 |
| 29 | Di 12.03.2024  15 Uhr | Verweis Arbeitspaket 29 |

## Organisation der Arbeitsergebnisse

Um die Organisation der Arbeitsergebnisse umfassend zu gewährleisten, setze ich auf eine diversifizierte Strategie, die lokale Speicherung, Cloud-Dienste Google Drive für Dokumentation, sowie mehrere GitHub-Repositories einschliesst. Speziell für die Dokumentation dient das Repository "2David4/IPA-Dokumentation" (Versionierung Dokumentation) als zentrale Sammelstelle, ergänzt durch zwei weitere Repositories für den Quellcode, um eine klare Trennung und effiziente Verwaltung zu sichern.

### Versionierung

Für die tägliche Arbeit und Versionskontrolle werden spezifische Branches erstellt, die den Fortschritt dokumentieren und eine klare Struktur bieten.

### Backup

Ein automatisiertes Backup-System sichert alle Änderungen bei jedem Push auf den Master-Branch des Dokumentations-Repositories auf einen dafür abgesehen Google Drive Ordner.

### Quellcode / Skripts

Für den Sourcecode der Projekte "Competec/ita-talent-api" und "Competec /ita-talent-client" werden dedizierte Branches (Versionierung Praktische Arbeit) für die IPA erstellt, um Entwicklungen klar zu trennen und zu versionieren.

### Arbeitsplatz

Der physische Arbeitsplatz ist systematisch organisiert, wobei relevante Kriterien und Zusammenhänge visuell auf einem Bulletin Board dargestellt werden. Diese Organisation fördert nicht nur die Effizienz, sondern auch die Einhaltung der Projektanforderungen.

## Arbeitsjournal

### Tag 1 – 29.02.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
| 01 | Dokumentationsstruktur und Teil 1 der Dokumentation erstellen | Erledigt |
| 04 & 05 | Kriterienkatalog und Aufgabenstellung analysieren | Erledigt |
| 07 | Zeitplan für die kompletten 10 Tage erstellen | Erledigt |
| 03 | Expertengespräch mit dem ganzen Expertenteam auf Zoom | Erledigt |
| Ausgeführte Arbeiten: | | |
| Erstellung des Aufbaus für die Dokumentation, inklusive des Titelblatts und der Gliederung des ersten Teils des Berichts. | | |
| Ausarbeitung eines detaillierten Zeitplans für die gesamte Dauer der IPA, unterteilt in 1, 2, 4-Stunden-Blöcke, entsprechend den Empfehlungen im Kriterienkatalog. | | |
| Herausforderungen: | | |
| Beim Erstellen des Zeitplans hatte ich anfangs Schwierigkeiten, einen gut überschaubaren Plan in Excel zu erstellen, da es echt eine Herausforderung ist so viele verschiedene wichtige Daten anschaulich darzustellen. | | |
| Beanspruchte Hilfestellungen: | | |
| Keine externen Hilfestellungen wurden in Anspruch genommen. | | |
| Vergleich mit Zeitplan: | | |
| Zeitplan wurde eingehalten. | | |
| Reflexion: | | |
| Der erste Tag startete mit einer gründlichen Einarbeitung in die Projektvorgaben, was mir einen klaren Fahrplan lieferte. Die Erstellung des Zeitplans in Excel erwies sich zunächst als Herausforderung, doch durch gezieltes Spacing und Farbgebung fand ich eine effektive Lösung. Diese Anpassung steigerte nicht nur die Übersichtlichkeit, sondern auch meine Zuversicht in die kommenden Aufgaben. | | |

### Tag 2 – 04.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
| 06 | Recherche und Sammlung von Informationen |  |
| 08, 09, 10 | Erstellung Konzepte der Ausgangslage, Datenbank, Realisierung und eines Testplans |  |
| 12 | Entscheidung für eine spezifische Lösungsvariante |  |
| 13 | Implementierung der Datenbank |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 3 – 05.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 4 – 06.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 5 – 07.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 6 – 11.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 7 – 12.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 8 – 13.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 9 – 14.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

### Tag 10 – 18.03.2024

| ID | Tagesziele | Status |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| Ausgeführte Arbeiten: | | |
|  | | |
|  | | |
| Herausforderungen: | | |
|  | | |
|  | | |
| Beanspruchte Hilfestellungen: | | |
|  | | |
| Vergleich mit Zeitplan: | | |
|  | | |
| Reflexion: | | |
|  | | |

Teil 2 – Projekt

## Kurzfassung

### Ausgangslage

…

### Umsetzung

…

### Ergebnis

## Phase "Informieren"

### Strategien zur Fehlerbehandlung

Bei der Weiterentwicklung des "CompAcademy Lernportals" habe ich mich intensiv mit dem Teilauftrag 3 auseinandergesetzt, der sich auf die Fehlerbehandlung, den Einsatz von XML-Adaptern und die Entwicklung von einer Fehlerbehandlung konzentriert. Während dieser Informationsphase habe ich zwei Hauptansätze zur Fehlerbehandlung evaluiert und zudem eine Methode zur Identifizierung von Problemen in generischen XML-Adaptern recherchiert.

Fehlerbehandlung: Zwei Hauptansätze

**1. Benutzerdefinierte Exceptions:** Diese Variante sieht vor, dass der XML-Adapter bei Auftreten eines Fehlers eine spezifisch definierte Exception wirft. Dieser Ansatz ermöglicht eine direkte und klare Kommunikation von Fehlern, stellt aber die Herausforderung dar, den Fehlerkontext präzise zu übermitteln.

**2. Transformations-DTO mit Fehlerdetails:** Im Gegensatz zum direkten Werfen von Exceptions sieht dieser Ansatz vor, dass der XML-Adapter ein Transformations-DTO zurückgibt, das während der Verarbeitung erstellt wird. Fehlerdetails werden in diesem DTO festgehalten, was eine umfassende Sicht auf den Fehlerkontext ermöglicht, ohne den Prozessfluss durch Exceptions zu unterbrechen.

**Dynamische Feldidentifikation**

Ein signifikantes Problem in beiden Ansätzen ist die Identifikation des spezifischen Feldes, das den Fehler verursacht, insbesondere weil die XML-Adapter generisch konzipiert sind und für mehrere Felder verwendet werden. In den XML-Adaptern selbst sind weder die ID noch der Feldname vorhanden, lediglich der Wert, was die Zuweisung und Rückverfolgung von Fehlern erschwert.

Um dieses Problem zu lösen, habe ich mich mit Techniken zur dynamischen Feldidentifikation in Java beschäftigt. Der Schlüssel liegt in der Verwendung der Java Reflection, speziell der Methode getDeclaredFields(), die es ermöglicht, alle Felder eines DTOs zur Laufzeit abzurufen. Diese Information kann dann genutzt werden, um in einer ausführlichen switch-Case-Logik spezifische Behandlungen oder Fehlermeldungen für jedes Feld zu definieren.

Diese Methode erlaubt es uns, flexible und wiederverwendbare XML-Adapter zu implementieren, die nicht nur auf die aktuelle Event-Datenstruktur beschränkt sind, sondern auch für zukünftige Anwendungen und Datenimporte, wie beispielsweise aus MyAbacus, adaptiert werden können.

**Quellen und Recherche**

Die Entwicklung dieser Ansätze basierte auf einer Kombination aus Fachliteratur, Online-Ressourcen und Diskussionen in Entwicklergemeinschaften. Insbesondere Artikel über Java Reflection und benutzerdefinierte Exceptions boten tiefgreifende Einblicke und praxisnahe Lösungsansätze.

Informationen zur Feldidentifikation: <https://www.baeldung.com/java-reflection-class-fields>

Informationen zum Marshalling Prozess der XML-Adapter: <https://docs.spring.io/spring-framework/reference/data-access/oxm.html>

Informationen zu Exception Bubbling und Best Practices: <https://coding-examples.com/java/exception-bubbling-in-java/>

**Zusammenfassung**

Die sorgfältige Erkundung und Bewertung der Fehlerbehandlungsoptionen sowie der dynamischen Feldidentifikation ohne den Einsatz von Spring bildeten eine solide Grundlage für die Implementierung effektiver Strategien im "CompAcademy Lernportal". Diese Erkenntnisse tragen wesentlich zur Verbesserung der Datenverarbeitungsrobustheit bei und ermöglichen eine flexible Erweiterung der Plattform für zukünftige Anforderungen.



### Informationsbeschaffung Bilderverwaltung

Diese Phase der Informationsbeschaffung vom Teilauftrag 2 konzentrierte sich auf die Integration von Spring Security, die Handhabung von Bildern mittels MultipartFile, Bildskalierung und -validierung. Ziel war es, eine sichere und benutzerfreundliche Lösung zu schaffen, die den visuellen Ansprüchen des Lernportals gerecht wird und dabei die technischen Anforderungen der Aufgabenstellung erfüllt.

**Sicherheitskonzepte mit Spring Security**

Die Einführung von Spring Security dient der Absicherung der Bildverwaltungsfunktionen, um sicherzustellen, dass nur autorisierte Benutzer Zugriff auf das Hochladen und Verwalten von Bildern haben. Dies erforderte eine tiefgehende Recherche von der Spring Security Dokumentation zu Authentifizierungs- und Autorisierungsmechanismen in Spring-basierten Anwendungen.

**Effiziente Bildverarbeitung**

Ein zentraler Aspekt der Bildverwaltung ist die effektive Handhabung von Bilduploads. Hierbei erwies sich MultipartFile als Schlüsselkomponente für das Empfangen und Verarbeiten von Bilddateien. Die Untersuchung der Spring File Upload Dokumentation galt der optimalen Anwendung von MultipartFile für den Bildupload-Prozess.

**Bildskalierung und Formatvalidierung**

Das Skalieren von Bildern, um eine maximale Breite von 1024px zu gewährleisten, und die Validierung der Bildformate (ausschliesslich PNG und JPG) waren weitere kritische Themen. Informationen zur Implementierung von Bildskalierungsmethoden wie Java Bildskalierung Dokumentation und Validierungslogiken waren von entscheidender Bedeutung, um die Konsistenz und Qualität der Bildinhalte zu sichern.

**Varianten der Endpunktgestaltung**

In der Informationsphase stiess ich auf zwei Varianten zur Gestaltung der Endpunkte für den Bildupload:

**Variante 1:** Nutzt @ModelAttribute zur automatischen Zuordnung der eingehenden Daten zu einem DTO.

**Variante 2:** Verwendet @RequestParam für eine explizite Definition und Verarbeitung der Upload-Parameter.

Beide Ansätze bieten spezifische Vorteile hinsichtlich Flexibilität und Strukturierung der Datenübermittlung, was eine fundierte Entscheidungsgrundlage für die Implementierung bildet.

**Quellen und Recherche**

Die Informationsphase stützte sich auf eine Vielzahl von Ressourcen, darunter:

Spring Security Dokumentation: <https://docs.spring.io/spring-security/reference/servlet/configuration/java.html>

Spring File Upload Dokumentation: <https://www.baeldung.com/spring-boot-thymeleaf-image-upload>

Java Bildskalierung Dokumentation: <https://www.baeldung.com/java-resize-image>

Diese Quellen boten wertvolle Einblicke in bewährte Methoden und technische Lösungen.

**Zusammenfassung**

Die gründliche Informationsbeschaffung zu den Kernaspekten der Bildverwaltung hat eine solide Basis für die Entwicklung der Bildverwaltungsfunktionen im "CompAcademy Lernportal" geschaffen. Durch die Bewertung von Sicherheitsaspekten, Bildverarbeitungstechniken und Endpunktgestaltungsvarianten ist es nun möglich, eine Lösung zu implementieren, die sowohl den technischen Anforderungen als auch den Nutzererwartungen entspricht. Die nächste Phase wird die praktische Umsetzung der gesammelten Erkenntnisse umfassen, mit dem Ziel, eine optimierte und sichere Bildverwaltung im Lernportal zu realisieren.



## Phase "Planen"

### Ist-Zustand des Systems

Im "Ist-Zustand des Systems" der IPA zur Erweiterung des "CompAcademy Lernportals" zeigt die initiale Skizze deutlich die bestehende Systemarchitektur und die Kommunikationspfade zwischen den Schlüsselkomponenten. Ein farbcodiertes Schema hilft dabei, die unterschiedlichen Kommunikationsmethoden im System zu veranschaulichen: Blau kennzeichnet die Verwendung von REST-APIs, während Grün für die bereits etablierte Websocket-Kommunikation steht.

Die Systemstruktur besteht aus mehreren Hauptkomponenten:

* Client (UI): Entwickelt mit React, stellt diese Komponente eine dynamische Schnittstelle für die Nutzer dar. Durch die Websocket-Kommunikation (grün dargestellt) wird eine direkte und reaktive Verbindung zum Backend hergestellt, was die Aktualität der Daten auf der Benutzeroberfläche gewährleistet.
* Backend (API): Das Backend, strukturiert um Websockets, nutzt Listener und Responder zusammen mit einem Exchange DTO, um eine effiziente und reaktive Datenverarbeitung sowie -übertragung zu ermöglichen. Diese Architektur unterstützt eine nahtlose Echtzeit-Kommunikation zwischen Client und Server.
* Datenbank (DB): In der PostgreSQL-Datenbank werden kritische Daten wie Kursinformationen gespeichert. Die enge Integration mit dem Backend ermöglicht eine effiziente Verwaltung und Abfrage dieser Daten.
* LMS: Das externe Learning Management System (LMS) von Umantis AG versorgt das System regelmässig mit den notwendigen Kursdaten. Über einen REST-Endpunkt (in Blau hervorgehoben) werden diese Informationen alle 10 Minuten abgerufen, um sie anschliessend im Lernportal zu konvertieren, zu normalisieren und in der Datenbank zu speichern.

Die Skizze des "Ist-Zustands des Systems" unterstreicht die zentrale Rolle der Websocket-basierten Backend-Architektur, die eine effektive Verbindung zwischen dem Client, der Datenbank und dem LMS herstellt. Die Verwendung von Farbcodes erleichtert das Verständnis der verschiedenen Kommunikationswege und liefert eine klare Basis für das Gesamtbild der Systemarchitektur.

### Soll-Zustand des Systems

Im "Soll-Zustand des Systems" für der IPA zur Erweiterung des "CompAcademy Lernportals" zeigt die aktualisierte Skizze die geplanten Neuerungen und Erweiterungen, die während dieser IPA implementiert werden sollen. Die Skizze nutzt ein farbcodiertes Schema, in dem die neu hinzukommenden Features in Rot hervorgehoben sind, um einen klaren Überblick über die geplanten Änderungen zu geben. Diese Neuerungen umfassen:

* Benutzerverwaltung (Person): Eine neue Komponente zur Verwaltung der Benutzerdaten, die es ermöglicht, Benutzerprofile effizient zu erstellen, zu bearbeiten und zu verwalten.
* Authentikation und Sessionhandling (Session): Diese Funktionalität soll ein sicheres Login-Verfahren und eine effiziente Verwaltung von Benutzersitzungen gewährleisten. Sie ist essenziell für die Sicherheit und Personalisierung des Nutzererlebnisses.
* Anzeige von Fehlern (Transformation): Eine wichtige Erweiterung zur Visualisierung von Fehlern, die während der Datenkonvertierung und -normalisierung auftreten können. Dies ermöglicht eine schnellere Identifizierung und Behebung von Problemen.
* Imports (Datum des Imports und Status): Die Implementierung einer Übersicht über die Importvorgänge, einschliesslich Zeitpunkt und Status, bietet Transparenz und Kontrolle über den Datenimport aus dem LMS.
* Bilderverwaltung (Image): Um die Kursangebote visuell ansprechender zu gestalten, wird eine vollständige Bilderverwaltung eingeführt. Diese Komponente wird über REST-Endpunkte angesprochen, um eine effiziente Verarbeitung und Darstellung von Bildern zu ermöglichen, getrennt von der Textlastigen Kommunikation via Websockets.

Die Entscheidung, für die Bilderverwaltung auf REST-Endpunkte zu setzen, gründet sich nicht nur auf die technische Notwendigkeit, die Ladezeiten für Bilder zu optimieren und eine unabhängige Handhabung von Bild- und Textinhalten zu ermöglichen, sondern folgt auch explizit den Vorgaben der Aufgabenstellung (Bilder werden über eine REST-Schnittstelle angefragt, hochgeladen, angepasst und gelöscht.). Dieser Ansatz erlaubt es, Bilder effizient und getrennt vom textbasierten Inhalt zu laden, was einen erheblichen Beitrag zur Performancesteigerung und Verbesserung der Benutzerfreundlichkeit des Lernportals leistet. Die klare Trennung von Bild- und Textdaten durch die Nutzung von REST-Endpunkten für die Bilderverwaltung entspricht somit nicht nur den technischen Bestrebungen nach einer optimierten Nutzererfahrung, sondern erfüllt auch die spezifischen Anforderungen, die in der Aufgabenstellung für dieses Projekt festgelegt wurden.

Durch diese Erweiterungen wird das Lernportal nicht nur um wichtige Funktionen reicher, sondern auch in Bezug auf Sicherheit, Benutzerfreundlichkeit und Effizienz deutlich verbessert. Die Integration neuer Features im "Soll-Zustand des Systems" markiert einen entscheidenden Schritt vorwärts in der Entwicklung des "CompAcademy Lernportals", um den Anforderungen und Erwartungen der Nutzer gerecht zu werden und die Verwaltung von Kursangeboten innerhalb der Competec-Gruppe zu optimieren.

### Datenbankmodell



|  |  |
| --- | --- |
|  | **Bestehende Tabellen (vor der IPA)** |
|  | **Neue Tabellen (während der IPA hinzugefügt)** |

Bei der Planung meiner IPA zur Erweiterung des "CompAcademy Lernportals" habe ich ein detailliertes Datenbankmodell entworfen, das die Kernstrukturen und die neu einzuführenden Elemente klar darstellt. Durch ein farbcodiertes Schema werden die bereits existierenden Bestandteile (in Blau) und die im Zuge der IPA hinzugefügten Erweiterungen (in Grün) unterschieden. Dieser Ansatz gewährleistet eine sofortige visuelle Unterscheidung und erleichtert das Verständnis der Datenbankarchitektur.

**Bestehende Strukturen und Beziehungen**

Das Modell zeigt die bestehenden Tabellen wie Event, Event\_Subcategory, Event\_Speaker, Event\_Event\_Speaker und Search, die die Basis des Lernportals bilden. Die Event-Tabelle, mit einer Selbstbeziehung über das Feld parent\_id, ermöglicht die Abbildung von Event-Hierarchien. Diese Struktur stellt sicher, dass Haupt- und Unter-Events effizient verwaltet werden können.

**Neue Erweiterungen in der Datenbank**

Während meiner IPA führe ich neue Tabellen ein:

* **Person:** Für die erweiterte Benutzerverwaltung, die die Grundlage für personalisierte Nutzererfahrungen im Lernportal schafft.
* **Person\_Session:** Unterstützt das neue Session-Handling, das für die Authentifizierung und Sessionverwaltung unerlässlich ist.
* **Image:** Ermöglicht das Hinzufügen von Bildern zu Kursen, was die Attraktivität der Kursangebote steigert.
* **Event\_Transformation und Event\_Import:** Diese Tabellen dienen der Überwachung und Verwaltung von Datenimport- und Transformationsprozessen, was eine detaillierte Fehleranalyse und -behandlung ermöglicht.

**Selbstbeziehung und Datenbeziehungen**

Ein besonderes Merkmal ist die Selbstbeziehung in der Event-Tabelle, die eine flexible Darstellung von Event-Strukturen erlaubt. Diese Konzeption spiegelt die Komplexität und Dynamik der Eventverwaltung wider und bietet eine solide Basis für die Darstellung von Event-Beziehungen im Lernportal.

**Zusammenfassung**

Das entworfene Datenbankmodell bildet eine wesentliche Grundlage für die erfolgreiche Umsetzung meiner IPA. Die klare Unterscheidung zwischen bestehenden und neuen Datenstrukturen durch Farbcodierung verbessert nicht nur die Übersichtlichkeit, sondern erleichtert auch die zukünftige Entwicklung und Erweiterung der Plattform. Die Planung dieses Datenbankmodells ist ein entscheidender Schritt, um die Funktionalität und Benutzerfreundlichkeit des "CompAcademy Lernportals" zu erweitern und zu optimieren. Im weiteren Verlauf meiner IPA wird dieses Modell als Richtlinie für die Entwicklung und Implementierung der Datenbankstruktur dienen.

### Tabellenstruktur

Im Rahmen meiner IPA zur Erweiterung des "CompAcademy Lernportals" ist die Entwicklung einer robusten Datenbankstruktur von zentraler Bedeutung. Diese Struktur soll nicht nur die spezifischen Anforderungen der drei Teilaufträge erfüllen, sondern auch die Grundlage für zukünftige Erweiterungen bieten. Jede Tabelle folgt einem einheitlichen Schema für Metadaten, das die Nachverfolgbarkeit und Verwaltung der Datensätze erleichtert:

**Standardmetadaten**

* date\_created: Zeitstempel der Erstellung des Datensatzes.
* user\_created: Benutzer, der den Datensatz erstellt hat.
* date\_modified: Zeitstempel der letzten Änderung des Datensatzes.
* user\_modified: Benutzer, der den Datensatz zuletzt geändert hat.
* is\_deleted: Markierung, ob der Datensatz als gelöscht betrachtet wird.

Nachfolgend sind die spezifischen Tabellen und deren Felder beschrieben, die zur Unterstützung der geforderten Funktionalitäten implementiert werden:

**Teilauftrag 1: Login und Benutzerverwaltung**

**Tabelle: person**

* id (UUID): Eindeutiger Identifikator für jede Person.
* name (VARCHAR): Benutzername, einzigartig.
* password (VARCHAR): Verschlüsseltes Passwort.
* email (VARCHAR): E-Mail-Adresse, einzigartig und für die Identifikation beim Login verwendet.
* is\_active (BOOLEAN): Zeigt an, ob das Benutzerkonto aktiv ist. Alternativ kann is\_deleted aus den Standardmetadaten verwendet werden. (siehe Gedankengänge)
* Standardmetadaten: Wie oben beschrieben.

**Tabelle: person\_session**

* id (UUID): Eindeutiger Identifikator für jede Session. (siehe Gedankengänge)
* person\_id (UUID): Fremdschlüssel, der auf die person Tabelle verweist und den Benutzer der Session identifiziert.
* token (VARCHAR): Einzigartiger Token für die Session, der für die Authentifizierung bei API-Anfragen verwendet wird.
* expires (TIMESTAMP): Das Ablaufdatum und -zeit der Session.
* Standardmetadaten: Wie oben beschrieben.

**Gedankengänge Person Tabelle**

**Variante mit is\_active:** Ermöglicht explizite Kontrolle über den Status eines Benutzers (aktiv/inaktiv), was besonders nützlich ist, um temporäre Deaktivierungen zu verwalten.

**Variante mit is\_deleted als Ersatz für is\_active:** Vereinfacht die Struktur, indem nur ein Feld für die Statusverwaltung (gelöscht/nicht gelöscht) genutzt wird. Könnte jedoch die Flexibilität in der Verwaltung von Benutzerkonten einschränken.

**Gedankengänge Person\_session Tabelle**

**Primärschlüssel:** Die Verwendung einer UUID für id ermöglicht eine eindeutige Identifizierung jeder Session.

**Token als Primärschlüssel?:** Eine Überlegung könnte sein, den Token selbst als Primärschlüssel zu verwenden, um die Eindeutigkeit zu garantieren. Allerdings könnten sich dadurch Einschränkungen in der Flexibilität ergeben.

**Teilauftrag 2: Bilderverwaltung**

**Tabelle: image**

* event\_id (INT) / id (UUID): Als Primärschlüssel genutzt, wobei event\_id die Einzigartigkeit pro Event sicherstellt. (siehe Gedankengänge)
* image (BYTEA): Das eigentliche Bild.
* description (VARCHAR): Bildbeschreibung.
* Standardmetadaten: Wie oben beschrieben.

**Gedankengänge zur Image Tabelle**

Die Gestaltung der Image-Tabelle bedarf einer sorgfältigen Überlegung, insbesondere im Hinblick auf die Verwaltung der Beziehung zwischen Bildern und Events. Hier sind zwei Varianten und die dazugehörigen Gedankengänge:

**Variante 1:** Verwendung von event\_id als Primärschlüssel (PK)

Ein Bild pro Event: Durch die Nutzung der event\_id als Primärschlüssel wird sichergestellt, dass jedes Event genau ein Bild zugeordnet haben kann. Dies vereinfacht die Zuordnung und vermeidet Redundanzen.

**Aktualisierung und Soft-Delete:** Wenn ein Bild zu einem Event aktualisiert oder gelöscht werden soll, würde der is\_deleted-Status des vorhandenen Datensatzes verwendet:

**Bildlöschung:** Setzt is\_deleted auf true, behält aber den Datensatz in der Datenbank. Dies ermöglicht eine einfache Wiederherstellung und beibehält die Historie.

Bildaktualisierung: Beim Hinzufügen eines neuen Bildes zu einem Event, das bereits ein Bild hat, müsste zunächst geprüft werden, ob bereits ein Datensatz mit der entsprechenden event\_id existiert und is\_deleted auf false gesetzt werden. Das neue Bild ersetzt das alte, und der Datensatz wird aktualisiert, anstatt einen neuen zu erstellen.

**Variante 2:** Verwendung von UUID als Primärschlüssel

Mehr Flexibilität: Die Verwendung einer UUID als Primärschlüssel erlaubt es, mehrere Bilder pro Event zu haben, wobei zu jedem Zeitpunkt nur eines als aktuell markiert ist (is\_deleted = false). Dies könnte die Verwaltung von Bildversionen oder temporären Bildaktualisierungen erleichtern.

Vereinfachte Logik für Hinzufügen und Löschen: Mit einer UUID als Primärschlüssel wäre die Logik für das Hinzufügen und Löschen von Bildern direkter. Ein neues Bild hinzuzufügen, würde einfach einen neuen Datensatz erstellen, ohne den Status eines vorhandenen Datensatzes überprüfen zu müssen. Ebenso würde das Löschen eines Bildes einfach den is\_deleted-Status des zugehörigen Datensatzes ändern, ohne sich um die event\_id-Uniqueness sorgen zu müssen.

**Teilauftrag 3: Fehlerbehandlung des Datenimports**

**Tabelle: event\_transformation**

* id (UUID): Eindeutiger Identifikator der Transformation.
* event\_id (INT): Verweist auf die betroffene Event-ID.
* title (VARCHAR): Titel der Transformation.
* type\_enum (VARCHAR): Typ der Transformation (z.B. Kritisch, Konvertierung).
* cause (VARCHAR): Ursache der Transformation.
* effect (VARCHAR): Effekt der Transformation auf die Daten.
* Standardmetadaten: Wie oben beschrieben.

**Event\_Import**

* id (UUID): Eindeutiger Identifikator des Importvorgangs.
* status\_enum (VARCHAR): Status des Importvorgangs (z.B. OK, Fehler).
* import\_time (TIMESTAMP): Optional, für den genauen Zeitpunkt des Importbeginns, falls eine Unterscheidung von date\_created notwendig ist. (siehe Gedankengänge)
* Standardmetadaten: Wie oben beschrieben.

**Gedankengänge zu Importzeitpunkten:**

**Verwendung von date\_created:** Dieser Ansatz würde den Zeitpunkt erfassen, zu dem der Importdatensatz in der Datenbank erstellt wurde. Obwohl dies eine einfache Lösung darstellt, spiegelt es möglicherweise nicht den tatsächlichen Beginn des Importprozesses wider, vor allem, wenn zwischen dem Start des Imports und der Datensatzerstellung in der Datenbank eine Verzögerung besteht.

**Einführung eines zusätzlichen Feldes import\_time:** Diese Variante erlaubt die explizite Speicherung des genauen Zeitpunkts, zu dem der Import gestartet wurde, unabhängig vom Zeitpunkt der Datensatzerstellung (date\_created). Dies bietet eine präzisere Nachverfolgung der Importvorgänge, erhöht jedoch die Komplexität des Datenmodells und des Importprozesses, da der Importzeitpunkt explizit im Code festgehalten und beim Erstellen des Datensatzes eingefügt werden muss.

### Testkonzept

## Phase "Entscheiden"