

Dokumentation Mitgliederverwaltung

SA

Kantonsschule Frauenfeld

Autor: Christian Köhler

Datum: 24.02.2022

Version: 1.0



Inhaltsverzeichnis

1	Allgemein	4
1.1	Aufgabenstellung der Schule	4
1.2	Aufgabenstellung des Auftraggebers	4
1.3	Deklaration der Vorkenntnisse	5
2	Projekt	5
2.1	Informieren	5
2.1.1	Analyse der Aufgabenstellungen	5
2.1.2	Benutzeranalyse	5
2.1.3	Ressourcen zur Informationsbeschaffung	5
2.2	Planung	6
2.2.1	Must-Haves	6
2.2.2	Nice-to-Haves	6
2.2.3	Use-Cases	7
2.2.4	Authorization Code Flow	7
2.2.5	Mockup	8
2.2.6	Zeitplan	9
2.2.7	Testfälle	10
2.2.8	Entwicklungsumgebungen	11
2.2.9	Datenbankschema	11
2.3	Entscheiden	11
2.3.1	Frontend	12
2.3.2	Backend	12
2.3.3	Lernressourcen	12
2.4	Realisation	12
2.4.1	Lerndokumentation	12
2.4.2	Zeitknappheit Abänderung des Fokus	14
2.4.3	Mitgliederverwaltung Datenbank	14
2.4.4	Frontend	14
2.4.5	Installationsanleitung	14
2.5	Kontrolle	14
2.5.1	Soll-Ist-Vergleich	15

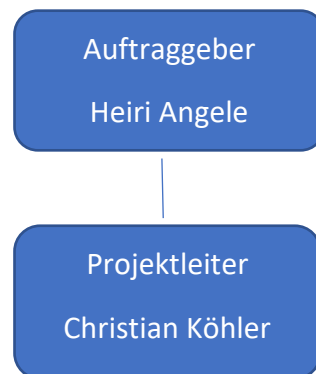
2.5.2	Testfälle ausgefüllt	16
2.6	Auswertung.....	17
2.6.1	Reflexion.....	17

1 Allgemein

1.1 Aufgabenstellung der Schule

Für diese Arbeit haben wir ein Semester Zeit. Dafür bekommen wir zwei Lektionen pro Woche zur Verfügung gestellt. Die Semesterarbeit ist eine Einzelarbeit, daher wurde die Dokumentation und der Code von mir geschrieben. Die Abgabe ist am 25. Februar 2022.

1.2 Aufgabenstellung des Auftraggebers



Heiri Angele ist ein Unternehmer. Wir haben uns an einem Anlass der Verbindung kennengelernt. In einer E-Mail beschrieb er das Projekt so: «Eine über OAuth 2.0 Autorisation gesicherte Verwaltung von Mitgliederdaten eines Vereines. Es soll ein Endpunkt in einem Webserver erstellt werden für Authentifizierung/Autorisierung der Clients und der Benutzer des Clients. Die gespeicherten Mitgliederdaten des angemeldeten Users sollen über einen weiteren Endpunkt ausgelesen und angezeigt werden können.»

Daraufhin trafen wir uns und er erklärte mir seine Vorstellung genauer. Sein Endziel ist es einen Service anzubieten, welcher allen Vereinen die Möglichkeit gibt, ihre Daten über Mitglieder zu verwalten und anschliessend diese Daten über eine API in ihre eigene Homepage einbinden zu können.

Da es nicht realistisch ist all das für die Semesterarbeit zu machen, hat er mir keinen genauen Auftrag gegeben, sondern einfach gesagt ich soll so weit wie möglich kommen. Wir haben uns darauf geeinigt, dass das dann ungefähr bis zur Mitgliederverwaltung mit OAuth sein wird, aber ohne das Einbinden in die Homepage. Darum wird in dieser Dokumentation nur die Mitgliederverwaltung berücksichtigt.

Weil eben die Daten später noch von einer API in andere Webseiten eingebettet und eventuell auch bearbeitet werden können, braucht man die Open Authorization. Ansonsten würde ein normales Login-System ausreichen.

Genauere Angaben, über wie ich das realisieren genau soll, hat er mir nicht gegeben. Das heisst ich habe eine freie Auswahl an Programmiersprachen Datenbanken und restlichem.

1.3 Deklaration der Vorkenntnisse

Vorkenntnisse:

- HTML, PHP, SQL, CSS: Schulkenntnisse 2.5 Jahre

Neues

- Open Authorization

2 Projekt

Die verwendete Projektmanagementmethode für dieses Projekt ist „IPERKA“, diese Methode eignet sich gut für individuelle Arbeiten dieser Grösse. Da mir vom Au

2.1 Informieren

Beim ersten Schritt geht es darum, den Auftrag zu verstehen und sich ein Bild des angestrebten Ziels zu machen. Da ich wenig zwingende Vorgaben bekommen habe, bin ich relativ frei in meiner Gestaltung habe mir selbst noch ein paar Ziele gesetzt, welche ich zur Liste hinzugefügt habe.

2.1.1 Analyse der Aufgabenstellungen

- Benutzerfreundliche Webseite
- Login durch OAuth geschützt
- Soll in allen gängigen Browsern funktionieren (Firefox, Edge, Chrome)

2.1.2 Benutzeranalyse

Um die Benutzeroberfläche zu gestalten, muss eine Benutzeranalyse durchgeführt werden. Aus zeitlichen Gründen wurde keine Umfassende Benutzeranalyse durchgeführt. Die unten aufgelisteten Punkte wurden von mir selbst geschrieben.

- Alter: 18-99
- Tätigkeit: Verwalter eines Vereins
- Computerkenntnisse: minimal

2.1.3 Ressourcen zur Informationsbeschaffung

Ein grosser Teil der Arbeit wird die Beschaffung und das Erlernen von neuen Informationen. Darum ist es wichtig diese gezielt zu wählen. Da OAuth für mich komplett neu ist, muss ich dort speziell den Fokus drauf richten. Es gibt viele verschiedene Anbieter, welche Kurse Artikel oder sonstiges zu diesem Thema anbieten.

2.1.3.1 Offizielle OAuth Dokumentation

Das ganze hat natürlich auch eine Dokumentation, welche auch sehr ausführlich ist. Diese findet man unter folgendem Link:

Pro: vollständig, ausführlich

Con: trocken, nicht Beginner freundlich

<https://oauth.net/specs/>

2.1.3.2 Tutorials im Internet

Es gibt eine Vielzahl an Tutorial im Internet. Eine einfache Google-Suche nach dem Begriff «Oauth» gibt zahlreiche Ergebnisse aus.

Pro: kostenlos, grosse Auswahl

Con: nicht immer vertrauenswürdig,

2.1.3.3 Udemy

Udemy ist eine US-amerikanische Online-Lernplattform. Es gibt viele Kurse zu verschiedenen Themen. Es gibt auch einige zum Thema Oauth, wie man unter folgendem Link sehen kann:

<https://www.udemy.com/topic/oauth/>

Pro: in Videoform

Cons: kosten

2.2 Planung

Eine genaue Planung ist wichtig um das Projekt strukturiert und rechtzeitig fertigzustellen. Nun werden mögliche Lösungsvarianten und das weitere Vorgehen ausgearbeitet. Die einzelnen Arbeitsschritte werden definiert, in eine Abfolge gebracht und der jeweilige Zeitbedarf geschätzt. Zur Erreichung der Terminvorgaben oder eigenen Ziele werden Meilensteine definiert. Ebenfalls werden notwendige Hilfs- und Arbeitsmittel zusammengestellt. Werden Arbeiten im Team erledigt, so sind die einzelnen Arbeitsschritte sinnvoll zu verteilen. Zudem werden Prüf- und Bewertungskriterien für die Kontrolle der Ausführung festgelegt.

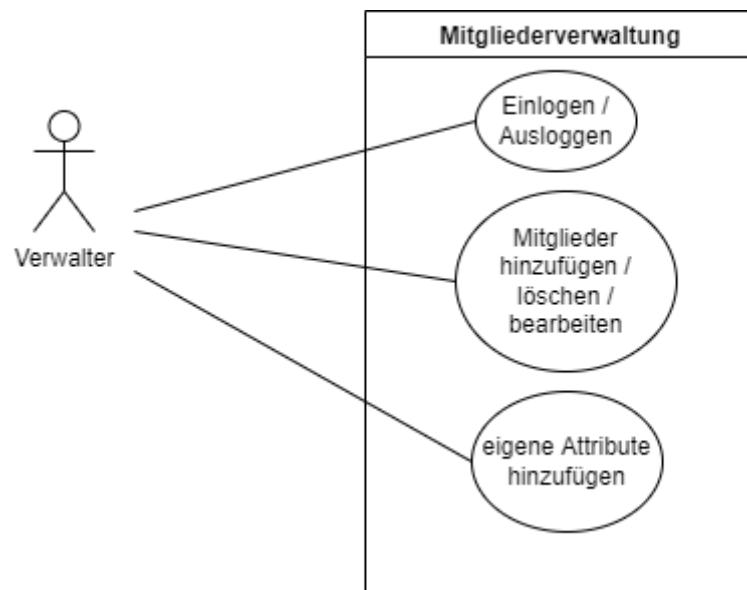
2.2.1 Must-Haves

Funktion	Beschreibung
Mitglieder hinzufügen / entfernen	Man soll mit wenigen Mausklicks Mitglieder hinzufügen oder entfernen.
Attribute bearbeiten	Man soll die Attribute jedes Mitgliedes leicht bearbeiten können
Durch den OAuth-Server einloggen.	Man soll auf den OAuth-Server umgeleitet werden, um sich dort einzuloggen, um anschliessend auf die Mitgliederverwaltung weitergeleitet zu werden.

2.2.2 Nice-to-Haves

Funktion	Beschreibung
Spezielle Attribute	Man soll für jedes Mitglied und oder Verein spezielle Attribute einstellen können.
API	(sehr nice-to-have) Eine API, welche es ermöglicht die Daten der Mitglieder auf die eigene Homepage darzustellen.

2.2.3 Use-Cases



2.2.4 Authorization Code Flow

Die Use-Cases für den Autorisatin habe ich nicht als einzelne Punkte im Use-Case-Diagramm dargestellt, sondern in Folgender Abbildung.

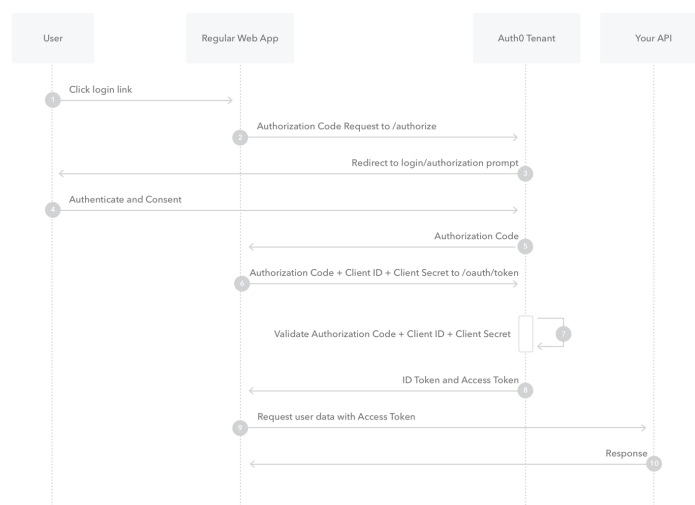
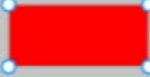







Bild 1 <https://auth0.com/docs/get-started/authentication-and-authorization-flow/authorization-code-flow>

2.2.5 Mockup

Mitgliederverwaltung		Login	
Vorname	Nachname	Löschen	Bearbeiten
Christian	Köhler		
Timon	Tommy		
Kai	Fuchs		

2.2.6 Zeitplan

1 Aufgabe definieren	September, Oktober, 30.09.2021	November,	Dezember,	Januar,	Februar, 25.02.2022
1.1 Ideen Sammeln					
1.2 Mit Auftraggeber absprechen					
1.3 Aufgabe definieren					
1.4 Mockup erstellen					
1.5 Meldeblatt abgeben					
2 Informieren					
2.1 Überblick verschaffen					
2.2 Aufgabe Analysieren					
2.3 Mockup / Use-Case					
2.4 ERM / DBS					
2.5 Testfälle erstellen					
2.6 Restliches Planen					
3 Entscheiden					
4 Realisation					
4.1 OAuth lernen					
4.2 Programmieren					
4.3 Datenbank erstellen					
4.4 Frontend programmieren					
5.3 Backend programmieren					
5.4 Dokumentation vollenden					
6 Alles überprüfen					
6.1 Abgabe					

2.2.7 Testfälle

Test	Ergebnis	Bestanden
Login		
klicken Sie auf den Login-Button	Das Login-Fenster erscheint	
Geben Sie für die E-Mail: „christian.k.1006@gmail.com“ ein und als Passwort: „123“	Man wird angemeldet und in index.php zurückgeleitet	
Geben Sie die richtige vorherige E-Mail, aber als	Fehlermeldung: Login-Daten stimmen nicht	
klicken Sie auf den Logout-Button	Man wird abgemeldet	
Klicken Sie in der Navbar wieder auf Mitgliederverwaltung	Man wird auf index.php umgeleitet	
Hinzufügen von Mitgliedern		
Klicken Sie auf hinzufügen	Es öffnet sich ein Eingabefenster	
Geben Sie bei Vornamen „Christian“ ein und lassen Sie den Nachnamen leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	
Geben Sie bei Nachnamen „Köhler“ ein und lassen Sie der Vorname leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	
Geben Sie bei Vornamen „Christian“ ein und bei Nachnamen „Köhler“ und geben sie als Geburtstag den 10.06.2003	Es wird ein neues Mitglied in der Tabelle erfasst mit diesen Daten	
Klicke auf das x oben	Das Bild wird wieder klein.	
Löschen von Mitglieder		
Klicken Sie auf dem Löschen bei dem vorher erstelltem Mitglied Christian Köhler	Das Mitglied verschwindet aus der Tabelle.	
Editieren		
Klicken Sie auf Editieren	Es öffnet sich ein Eingabefenster	
Geben Sie bei Vornamen „Christian“ ein und lassen Sie den Nachnamen leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	
Geben Sie bei Nachnamen „Köhler“ ein und lassen Sie der Vorname leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	
Geben Sie bei Vornamen „Christian“ ein und bei Nachnamen „Köhler“ und geben sie als Geburtstag den 10.06.2003	Es wird ein neues Mitglied in der Tabelle erfasst mit diesen Daten	
Klicke auf das x oben	Das Bild wird wieder klein.	

2.2.8 Entwicklungsumgebungen

2.2.8.1 Netbeans

Netbeans ist die Entwicklungsumgebung, die ich bis jetzt am meisten benutzt habe. Sie ist mir vertraut und ist zuverlässig.

2.2.8.2 Visual Studio Code

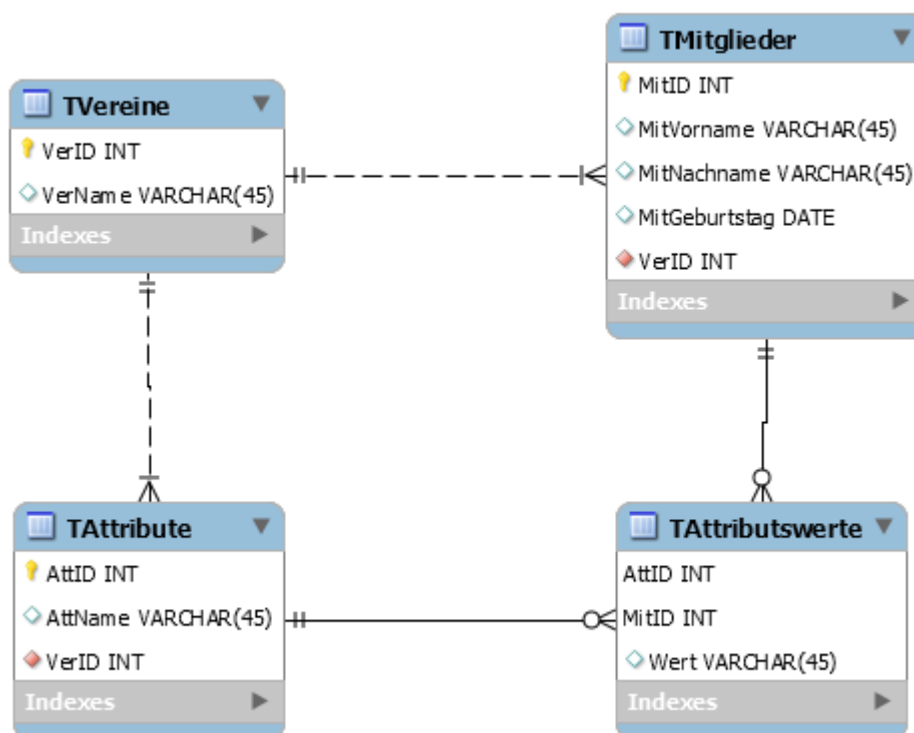
In letzter Zeit habe ich aber immer mehr zu VS Code gegriffen, da es auch viele nützliche Funktionen hat, welche Netbeans nicht hat. Darum werde ich wahrscheinlich hauptsächlich hiermit arbeiten und nur wenn ich spezielle Funktionen von Netbeans haben möchte, damit arbeiten.

2.2.8.3 MySQL Workbench

Dies ist mein bisher einzig benutztes Tool für SQL und Datenbanken, daher werde ich sehr wahrscheinlich auch für dieses Projekt diese Entwicklungsumgebung benutzen.

2.2.9 Datenbankschema

Obwohl das Hinzufügen von eigenen Attributen bei den Nice-To-Haves war, habe ich diese schon in der Datenbank eingebaut, weil es schwer ist diese später zu erweitern und es kein grosser Mehraufwand war.



2.3 Entscheiden

Da vieles für mich neu ist, kann ich nicht alles vorausplanen und werde manche Dinge im Laufe des Prozesses entscheiden müssen.

Nach der Planung muss eine Lösungsvariante ausgewählt werden. Meine Entscheidungen habe ich mit meinem Verstand und aus Intuition getroffen. Auf spezielle Methoden wie:

Evaluation, Nutzwertanalyse, Präferenzmatrix oder ähnlichem habe ich verzichtet. Entscheide sollten nicht hinausgezögert werden, da dies viel Zeit kosten kann. Oder es gibt nichts mehr zu entscheiden, weil plötzlich Sachzwänge vorliegen.

2.3.1 Frontend

Das Aussehen der Webseite kann unabhängig von OAuth gestaltet werden. Daher konnte ich zwischen vielen verschiedenen Optionen wählen. Da ich für dieses Projekt aber schon genug Neues dazulernen werden muss, werde ich in diesem Bereich kein Risiko eingehen und auf altbewährtes **Bootstrap** setzen.

2.3.2 Backend

Für das Backend wird natürlich OAuth benutzt. Da das aber nur der Authentifikation und Autorisierung ist, brauchen wir auch noch einen Ressource-Server, welcher **MySQL** benutzen wird. Für diese Datenbank habe ich mich entschieden, da ich mich da am besten auskenne. Ausserdem hat eine relationale Datenbank viele Vorteile.

2.3.3 Lernressourcen

Als meine primäre Ressource für das Erlernen von OAuth habe ich mich für folgenden Udemy Kurs entschieden:

<https://www.udemy.com/course/oauth-2-simplified/>

Dafür habe ich mich entschieden, weil ich lieber den Preis von 15 Fr. zahle und dann einen guten Kurs habe, anstatt mir irgendwelche mittelmässige YouTube Videos oder Artikel zu dem Thema zusammenkratzen zu müssen. Ausserdem gibt es bei diesem Kurs Übungen, welche mich hoffentlich gut darauf vorbereiten mein Projekt zu realisieren.

2.4 Realisation

Das Realisieren oder Ausführen tut bei Entwicklungsaufgaben einen zeitlichen Hauptteil einer Aufgabe umfassen. Der erstellte Arbeitsplan sollte nicht ohne zwingende Gründe geändert werden.

2.4.1 Lerndokumentation

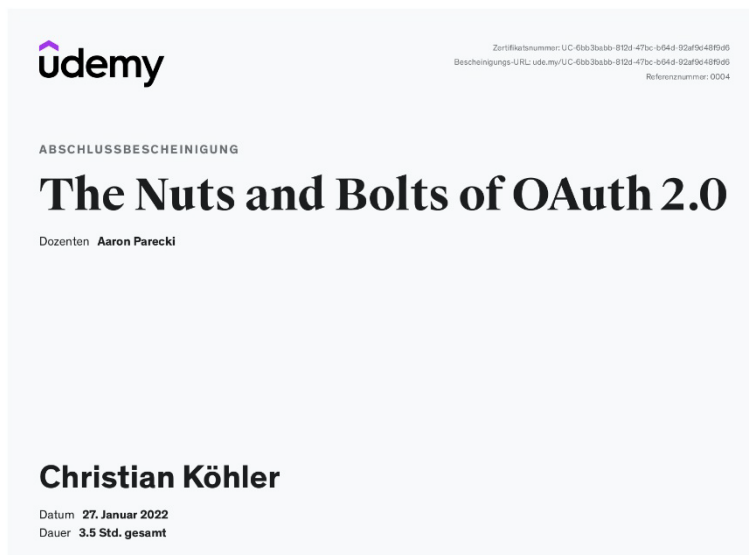
Da wie schon oft bereits erwähnt vieles für mich neu ist wird ein grosser Teil der Realisation in das Lernen und Verstehen für OAuth aufgewendet.

2.4.1.1 Udemy

#Der Kurs ist sehr umfangreich mit 3.5h Video und dazu noch Übungen. Ich habe zwar nie die Zeit gestoppt, aber wenn ich schätzen müsst, hätte ich drei bis vier Mal so viel Zeit verwendet wie nur den Videoinhalt. Das liegt daran, dass, obwohl in der Beschreibung stand, dass man keine besonderen Vorkenntnisse für den Kurs braucht, doch noch einige Begriffe benutzt wurden, von denen ich noch nichts gehört habe. Noch dazu kamen die Übungen, bei denen ich manchmal steckengeblieben bin. Das habe ich aber miteinberechnet.

2.4.1.1.1 Abschlussbescheinigung

Auch wenn diese Abschlussbescheinigung eigentlich keinen wirklichen Wert hat, habe ich sie vollständigkeitshalber doch noch hinzugefügt.



Abschnitt 1: Welcome	5 / 5 16 Min.
Abschnitt 2: API Security Concepts	6 / 6 28 Min.
Abschnitt 3: OAuth Clients	1 / 2 1 Min.
Abschnitt 4: OAuth for Server-Side Applications	2 / 3 15 Min.
Abschnitt 5: OAuth for Native Applications	6 / 6 22 Min.
Abschnitt 6: OAuth for Single-Page Applications	4 / 5 21 Min.
Abschnitt 7: OAuth for the Internet of Things	2 / 2 9 Min.
Abschnitt 8: Client Credentials Flow	2 / 3 5 Min.
Abschnitt 9: Introduction to OpenID Connect	5 / 6 22 Min.
Abschnitt 10: Protecting an API with OAuth	1 / 1 1 Min.
Abschnitt 11: Access Token Types and their Tradeoffs	3 / 4 12 Min.
Abschnitt 12: JWT Access Tokens	5 / 5 20 Min.
Abschnitt 13: Choosing Token Lifetimes	4 / 4 11 Min.
Abschnitt 14: Handling Revoked or Invalidated Access Tokens	5 / 5 12 Min.
Abschnitt 15: OAuth Scopes	4 / 4 11 Min.
Abschnitt 16: Conclusion	3 / 3 6 Min.

2.4.1.2 WSL

Während dem Lösen der Aufgaben, wurde immer öfters mit Linux oder Mac bespiele gemacht. Irgendwann war es mir zu blöd Windows alternativen zu benutzen und habe das Windows Subsystem for Linux heruntergeladen und eingerichtet. Vor allem für manuelle Curl-Commands und Git hat es mir geholfen.

2.4.1.3 OAuth Libraries

Nachdem ich die Theorie grösstenteils verstanden habe, kommt wieder eine kleine Planungs- und Entscheidungsphase.

Man könnte theoretisch alles selbst programmieren, aber das ist unrealistisch und unnötig. Dafür gibt es Libraries, welche auf folgender Seite aufgelistet sind:

<https://oauth.net/code/>

Okta war mir schon bekannt, denn ich habe alle Übungen des Udemy-Kurses damit absolviert, weil der Lehrer auch das benutzt hat. Trotzdem wollte ich noch alternativen anschauen, falls es doch besser ginge.

Schlussendlich bin ich aber bei Okta geblieben, denn für meine Ziele ist es ausreichend und ich kenne mich ja schon ein wenig damit aus.

2.4.2 Zeitknappheit Abänderung des Fokus

Da vor allem das Erlernen etwas länger ging als gedacht und ich bei der Einrichtung von OAuth immer wieder in Probleme stosse ist die Zeit knapp geworden.

Ich versuche zwar genau der Dokumentation von Okta selbst zu folgen, welche unter folgendem Link zu finden ist:

<https://developer.okta.com/>

Doch ich habe die ganze Sache unterschätzt und die Beispiele laufen bei mir nicht einmal korrekt. Anfangs war [Composer](#) ein Dependency Manager für PHP wahrscheinlich das Problem, aber danach wusste ich nicht mehr weiter, weil es auch sonst nicht funktioniert hat.

Obwohl das primäre Ziel der Arbeit eigentlich die OAuth-Sache war, setzte ich den Fokus auf die Mitgliederverwaltung selbst um, weil ich weiss, dass ich in diesem Bereich sicher etwas hinbekommen werde. Bei OAuth bin ich mir da nicht mehr so sicher.

2.4.3 Mitgliederverwaltung Datenbank

Die Datenbank zu erstellen war dafür nicht so schwer. Damit hatte ich schon Erfahrung. Das XAMPP-Tool habe ich zwar noch nie für MySQL-DB gebraucht, aber das ganze zum Laufen zu bekommen war schnell gemacht.

2.4.4 Frontend

Das Frontend habe ich aus einer Mischung zwischen Bootstrap JQuery Ajax gemacht. Von den letzten beiden habe ich nur etwas Erfahrung, aber es hat ausgereicht um diesen Teil ohne grosse weitere Komplikationen bewältigen können. Auf eine Erklärung des Codes wird verzichtet, da das nicht nötig ist.

2.4.5 Installationsanleitung

- Laden sie die Dateien von folgendem Githublink herunter:
<https://github.com/Christian257/SA>
- Bei XAMPP starten Sie PHP und MySQL
- Importieren Sie die Datenbank mit dem MV_Inserts.sql Script.
- Starten Sie index.php
- Fertig

2.5 Kontrolle

In der Kontrolle wird über den ganzen Ablauf nochmals ein Auge geworfen und geschaut, ob alles so funktioniert wie es geplant war. Das heisst z.B. nochmals durchlesen, nachrechnen, mit Vorgaben oder Planung vergleichen, testen, anwenden usw.

2.5.1 Soll-Ist-Vergleich

Vieles was in der Originalen Planung war, wurde nicht in die Tat umgesetzt. Sogar der eigentliche Fokus der Arbeit, welcher die OAuth Autorisation und Authentifikation war wurde nicht erreicht. Was erreicht wurde ist die Mitgliederverwaltung an sich. Trotzdem gibt es nur einen User und überhaupt kein Login.

2.5.2 Testfälle ausgefüllt

Getestet von: Christian Köhler

Getestet Version des Projekts:1.0.

Maschine + Betriebssystem: Eigener Laptop HP Envy x360 Windows 11

Test-Datum:24.02.22

Test	Ergebnis	Bestanden
Login		
klicken Sie auf den Login-Button	Das Login-Fenster erscheint	×
Geben Sie für die E-Mail: „christian.k.1006@gmail.com“ ein und als Passwort: „123“	Man wird angemeldet und in index.php zurückgeleitet	×
Geben Sie die richtige vorherige E-Mail, aber als	Fehlermeldung:Login-Daten stimmen nicht	×
klicken Sie auf den Logout-Button	Man wird abgemeldet	×
Klicken Sie in der Navbar wieder auf Mitgliederverwaltung	Man wird auf index.php umgeleitet	×
Hinzufügen von Mitgliedern		
Klicken Sie auf hinzufügen	Es öffnet sich ein Eingabefenster	✓
Geben Sie bei Vornamen „Christian“ ein und lassen Sie den Nachnamen leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	✓
Geben Sie bei Nachnamen „Köhler“ ein und lassen Sie der Vorname leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	✓
Geben Sie bei Vornamen „Christian“ ein und bei Nachnamen „Köhler“ und geben sie als Geburtstag den 10.06.2003	Es wird ein neues Mitglied in der Tabelle erfasst mit diesen Daten	✓
Klicke auf das x oben	Das Bild wird wieder klein.	✓
Löschen von Mitglieder		
Klicken Sie auf dem Löschen bei dem vorher erstelltem Mitglied Christian Köhler	Das Mitglied verschwindet aus der Tabelle.	✓
Editieren		
Klicken Sie auf editieren	Es öffnet sich ein Eingabefenster	✓
Geben Sie bei Vornamen „Christian“ ein und lassen Sie den Nachnamen leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	×
Geben Sie bei Nachnamen „Köhler“ ein und lassen Sie der Vorname leer	Fehlermeldung: Vor- und Nachname müssen ausgefüllt sein	×
Geben Sie bei Vornamen „Christian“ ein und bei Nachnamen „Köhler“ und geben sie als Geburtstag den 10.06.2003	Es wird ein neues Mitglied in der Tabelle erfasst mit diesen Daten	✓
Klicke auf das x oben	Das Bild wird wieder klein.	✓

2.6 Auswertung

Beim Auswerten geht man die Bearbeitung des Auftrages vom Informieren bis zum Kontrollieren noch einmal in Gedanken durch und beurteilt jeden einzelnen Schritt. Es geht nicht darum, sich eine Note zu geben, sondern sich selbst einzuschätzen und sich bewusst zu werden, was verbesserungswürdig ist, beibehalten werden sollte oder was man bei diesem Auftrag gelernt hat. Bei der Reflexion wird das Resultat, der Prozess, die Ressourcen betrachtet. Es werden Erkenntnisse formuliert und daraus Optimierungen abgeleitet.

2.6.1 Reflexion

Obwohl das Projekt nicht so lief wie geplant und es nicht wirklich fertig wurde, habe ich trotzdem einiges gelernt. Ich verstehe nun wie OAuth und weitere Sicherheitskonzepte funktionieren, auch wenn ich dies leider nicht in die Tat umsetzen konnte.

Das hat mehrere Gründe. Ich habe unterschätzt, wie lange ich brauchen werde, um OAuth zu verstehen. Der Udemy-Kurs allein hat nämlich nicht gereicht. Dass ich immer wieder Sachen nachschlagen musste, welche er als Grundwissen angesehen hat, hat viel Zeit beansprucht. Das lag aber nicht am Lehrer. Man kann diese Sache nicht an sich alleine lernen, es gab viele Dinge die auf einer Grundlage von etwas aufgebaut waren, welche ich noch nicht verstand oder beherrschte.

Obwohl ich frühzeitig angefangen habe mit dem Projekt ist die Zeit knapp geworden. Wenn ich noch etwas intensiver an dem Projekt dran gewesen wäre, hätte ich es vielleicht noch hinbekommen.

Was ich auch hätte tun können ist den Fokus früher umzusetzen. Die OAuth-Sache weglassen und die Mitgliederverwaltung an sich noch besser machen.

Eine Schwierigkeit war auch, dass es ein Thema ist, welches ich nicht kannte und somit viel lernen musste. Das war nicht besonders motivierend oder spannend. Im Nachhinein hätte ich lieber ein Projekt wählen sollen, wo ich mehr oder weniger direkt mit dem Programmieren anfangen hätte können. Das Programmieren hat mir nämlich am meisten Spass gemacht.

Schlussendlich bin ich zwar etwas unzufrieden, aber an Arbeit hat es nicht unbedingt gefehlt.