

# Hochschule Bremerhaven

**Fachbereich II**  
**Management und Informationssysteme**  
**Wirtschaftsinformatik B.Sc.**

## **Abschlussbericht für das Projekt ”Fitnessraumvermietungssystem”**

vorgelegt im Rahmen des Moduls  
**Technik für Wirtschaftsinformatik**

**Autor:in und MatNr.:**

Hossein Akbari MatNr. 39940  
Junior Lesage Ekane Njoh MatNr. 40128  
Steve Aguiwo II MatNr. 40088  
Refat Haj Issa MatNr. 38556

Bremerhaven, den 8. Oktober 2023

---

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung und Motivation</b>	<b>5</b>
1.1	Ideenfindung für das Projekt . . . . .	5
1.2	Spezifikation . . . . .	5
<b>2</b>	<b>Konzeptionsbeschreibung</b>	<b>6</b>
2.1	Zusammenarbeit und Projektplanung . . . . .	6
2.2	Anwendungsfalldiagramm . . . . .	7
2.3	Aktivitätsdiagramm . . . . .	8
2.4	Klassendiagramm . . . . .	10
2.5	Realisierungsbeschreibung . . . . .	11
2.6	Verwendete Techniken . . . . .	11
2.6.1	HTML und CSS - Hossein . . . . .	11
2.6.2	HTML und CSS: Statische und dynamische Webseiten - Hossein . . . . .	12
2.6.3	JavaScript - Hossein . . . . .	13
2.6.4	JavaScript: Entstehung - Hossein . . . . .	13
2.6.5	JavaScript: Serverseitige und clientseitige Anwendun- gen - Hossein . . . . .	14
2.6.6	PHP - Hossein . . . . .	15
2.6.7	SQL-Datenbanken - Hossein . . . . .	16
2.6.8	MySQL als wichtigstes und beliebtestes Datenbanksys- tem - Hossein . . . . .	16
2.6.9	SQL als Datenbanksprache - Hossein . . . . .	17
2.6.10	Redis - Hossein . . . . .	17
2.6.11	Git: Nutzung von Versionsverwaltungssystem - Hossein	17
2.6.12	Git: Aufgaben der Versionsverwaltung - Hossein . . . .	18
2.6.13	Git: Entstehungsgeschichte - Hossein . . . . .	19
2.7	Implementieren der Techniken zu einem Softwaresystem . . . .	20
2.7.1	HTML in Anwendung bei unserem Projekt - Junior . .	20
2.7.2	CSS in Anwendung bei unserem Projekt - Junior . . .	24
2.7.3	JavaScript in Anwendung bei unserem Projekt - Junior	32
2.7.4	Single-Page-Application - Junior . . . . .	32
2.7.5	JavaScript-Funktionen - Junior . . . . .	33
2.7.6	Funktionen mit Übergabewerten - Junior . . . . .	34
2.7.7	Zugriff auf die Inhalte des DOM-Baums - Junior . . . .	35
2.7.8	Events in JavaScript - Junior . . . . .	35
2.7.9	AJAX-Aufruf - Junior . . . . .	36
2.7.10	Die if-Abfrage und Callback - Junior . . . . .	39
2.7.11	Die if-Abfrage - Junior . . . . .	39

---

2.7.12	Callback - Junior . . . . .	40
2.7.13	PHP mit CSS, HTML und JavaScript - Refat . . . . .	40
2.7.14	HTML-Code mit PHP ausgeben - Refat . . . . .	41
2.7.15	Init-Fehlermeldung - Refat . . . . .	42
2.7.16	PHP: Assoziative Arrays - Refat . . . . .	43
2.7.17	PHP: Wozu werden assoziative Arrays verwendet - Refat	43
2.7.18	PHP: Sessions und ihre Anwendung - Refat . . . . .	44
2.7.19	Was sind Schleifen überhaupt? - Refat . . . . .	45
2.7.20	PHP: Funktionen - Refat . . . . .	46
2.7.21	PHP: eine Funktion in das PHP-Programm einbinden - Refat . . . . .	47
2.7.22	Zugriff auf MySQL-Datenbank durch PHP - Hossein .	48
2.7.23	Redis: Definition - Refat . . . . .	51
2.7.24	Redis: Key-Value-Store und Message-Broker - Refat . .	52
2.7.25	Redis: Wann sollte eine Key-value-Datenbank verwen- det werden? - Refat . . . . .	53
2.7.26	Redis: Anwendungsfälle für Schlüssel-Wert-Datenbanken - Refat . . . . .	54
2.7.27	Redis: Redis mit PHP verbinden (new, pconnect, auth, set, publish) . Refat . . . . .	55
2.7.28	Git - Steve . . . . .	55
2.8	Externe Prozesse (Junior) . . . . .	63
2.8.1	Cronjob . . . . .	64
2.8.2	Cronjob in Anwendung bei unserem Projekt . . . . .	64
2.8.3	Noch weitere Skripte . . . . .	65
2.9	Lasttest (Junior) . . . . .	66
<b>3</b>	<b>Reflexion und Fazit</b>	<b>69</b>
	<b>Literatur</b>	<b>69</b>

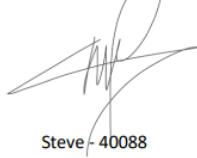
#### Eigenständigkeitserklärung

Hiermit bestätigen wir, dass wir die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Hilfsmittel benutzt haben. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken (dazu zählen auch Internetquellen) entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht.

Hossein - 39940

Hossein

Junior - 40128



Steve - 40088



Refat - 38556



---

# 1 Einleitung und Motivation

Im Rahmen des Moduls "Technik für Wirtschaftsinformatik" haben wir als Team f die Aufgabenstellung bekommen, mit den erworbenen Techniken aus der Veranstaltung, ein Softwaresystem zu programmieren, indem die Techniken in einem sinnvollen Zusammenhang miteinander implementiert werden sollen.

## 1.1 Ideenfindung für das Projekt

Am Anfang hatte jeder aus dem Team die Aufgabenstellung, sich selbstständig zu überlegen, welche Art von Softwaresystem, seiner Vorstellung nach in Frage kommen würde. Nach Dieser, wurden bei einem Meeting, die zu erbringenden Ideen gegeneinander abgewogen und ausdiskutiert, bis zu dem Zeitpunkt, dass das Team sich auf eine herausgebrachte Idee einigt. Hierfür wurden die vorgestellten Ideen nach Bestimmten Kriterien wie Umsetzung, Komplexität, Struktur und etc. beurteilt.

Letztendlich ist das Ergebnis der Diskussion ein Softwaresystem gewesen, mit der man an Fitnessinteressenten, die als Benutzer des Softwaresystems gelten, Fitnessräume mit verschiedenen Fitnesszielen vermieten kann, mit der Bezeichnung als "Fitnessraumvermietungssystem".

## 1.2 Spezifikation

Basierend auf dieser Grundlage haben wir für die Softwarespezifikation uns mit den Details der Software beschäftigt. Hierfür haben wir durch Brainstorming und das Erstellen von einem Mindmap, über die Funktionalitäten, Abläufe, Schnittstellen, etc. gemeinsam assoziiert, uns die hervorgebrachten Ideen notiert und somit einen grob eingeschätzten Leitfaden für das weitere Vorgehen des zu Entwickelnden und zu Implementierenden Softwaresystems erstellt.

Durch das Softwaresystem sollen Benutzer, von den Fitnessräumen und die damit einhergehenden Fitnesskursen, die vom Trainer bzw. Admin angeboten werden, sich beliebig für eine anmelden, abmelden, alle zu verfügbaren Fitnesskurse ansehen, davon wiederum die angemeldeten Kurse ansehen und etc.

Genauere Informationen gibt es im Nachhinein.

---

## 2 Konzeptionsbeschreibung

### 2.1 Zusammenarbeit und Projektplanung

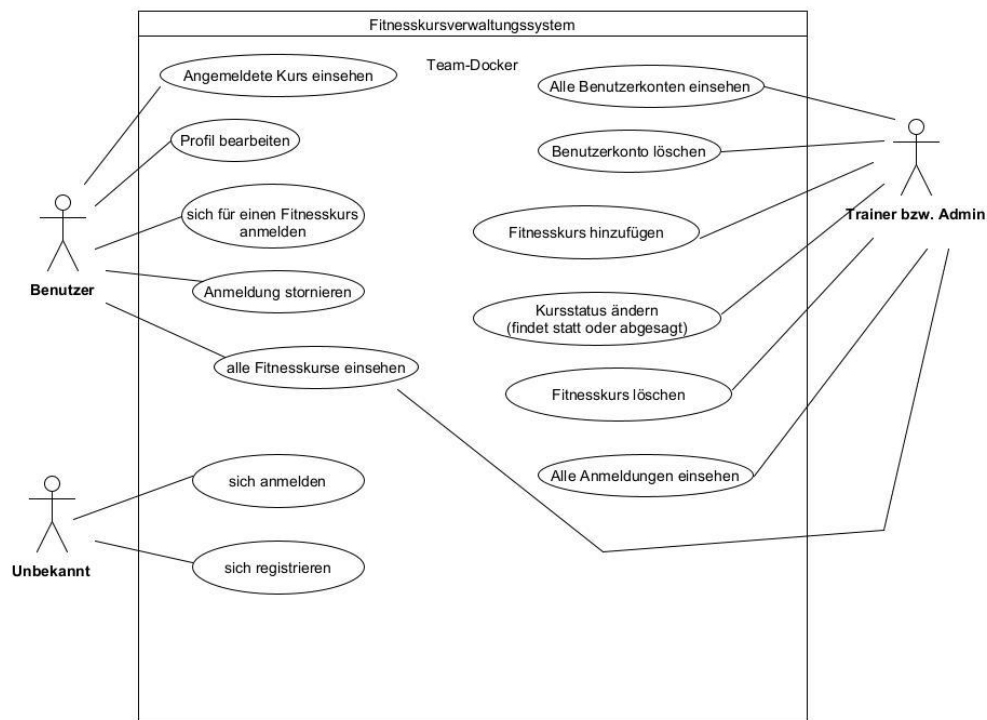
Als wir angefangen haben, an das Projekt zu arbeiten, haben wir uns zunächst um die zu erreichenden Anforderungen gekümmert und uns die Frage gestellt, in welche Richtung das Softwaresystem gehen sollte. Dieser Ansatz bildete die Grundlage für die weitere Vorgehensweise unseres Projektes.

Was unsere interne Zusammenarbeit zwischen uns als mehrere Entwickler angeht, so haben wir durch regelmäßige Online-Meetings, oder Präsenstreffen an der Hochschule dafür gesorgt, um den bisher aktuell erbrachten Leistungen zu kontrollieren, sowie den Informationsaustausch, sowie Kommunikation im Projekt flüssig zu halten, und gleichzeitig darauf zu achten, dass jeder Einzelne, sowie das gesamte Team im Allgemeinen von dem eigentlichen Leitfaden, in welche Richtung das Projekt läuft, nicht abweicht. Wir haben jeden Einzelnen im Team durch planungsvolle Aufgabenverteilung, die zu ertragende Verantwortung mit in die Hand gegeben, und diese regelmäßig in Sprints organisiert, wodurch es uns ermöglicht wurde, die bisher erbrachten Fortschritte zu verfolgen, die möglichen in Zukunft auftauchenden Problemen im Voraus zu identifizieren, und an diese im Voraus schon Anpassungen vorzunehmen.

Was die technisch zu umzusetzende Programmierung angeht, so haben wir hierfür das Versionsverwaltungssystem Git verwendet, mit der zwischen uns als mehrere Entwickler, ebenfalls eine effiziente Softwareentwicklung und Zusammenarbeit ermöglicht wurde. Hierdurch hatte jeder Entwickler die Möglichkeit, parallel an verschiedenen Teilen des Codes, bei einer Datei zu arbeiten, und diese am Ende zusammenzuführen. Ferner wurden zwischendurch ebenfalls regelmäßig Code-Reviews und Tests durchgeführt, um hierdurch die sorgfältige Softwareentwicklung während einer längeren Zeit, bezogen auf die Qualität der Entwicklung, zu gewährleisten.

## 2.2 Anwendungsfalldiagramm

Für das gemeinsame Programmieren und Implementieren des Softwareprojektes haben wir in erster Stelle mithilfe eines Anwendungsfalldiagrammes, uns um die funktionalen Anforderungen des Softwaresystems gekümmert.



Mithilfe des Anwendungsfalldiagrammes wurden die funktionalen Anforderungen, die im Softwaresystem implementiert worden sind, graphisch dargestellt. Hierdurch ist zu erkennen, dass das Softwaresystem drei Akteure beinhaltet (Unbekannt, Benutzer, sowie Trainer bzw. Admin). Geht man als ein unbekannter Akteur auf das Softwaresystem drauf, so muss man sich zuerst entweder als Benutzer, oder als Trainer bzw. Admin registrieren. Danach hat man die Möglichkeit, sich in das System anzumelden.

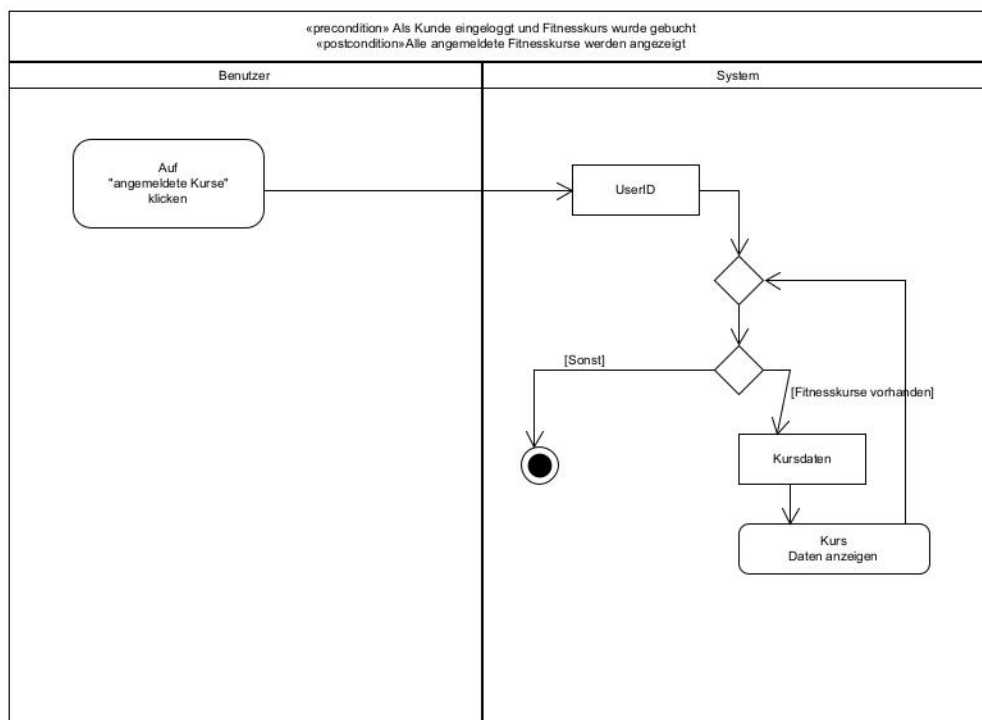
Ist man bei der Anmeldung ein Benutzer des Systems, also sozusagen ein Kunde, der Fitnesskurse und die damit verbundenen Fitnessräume buchen möchte, so hat man, was die funktionalen Optionen angeht, die Möglichkeit, neben Profil bearbeiten, sich alle zur Verfügung gestellten Kurse anzusehen. Darüber hinaus kann man sich für einen Kurs anmelden, sowie sich alle angemeldeten Kurse in Form einer Liste anzuschauen. Im Bereich „Angemeldete Kurse einsehen“ gibt es wiederum die Option, seine Anmeldung für einen angemeldeten Kurse wieder zu stornieren.

Ist man bei der Anmeldung hingegen ein Trainer bzw. Admin des Systems, so hat man, was die funktionalen Optionen angeht, die Möglichkeit, alle Benutzerkonten einzusehen, sowie diese wieder löschen. Außerdem hat man die Möglichkeit Fitnesskurse hinzuzufügen, diese wieder löschen, sowie den Kursstatus als "findet statt" oder "abgesagt", zu ändern. Darüber hinaus hat man neben der Option, "Alle Anmeldungen einsehen", die Möglichkeit, wie der Benutzer, auf der anderen Seiten, sich ebenfalls alle Fitnesskurse anzusehen.

## 2.3 Aktivitätsdiagramm

Um zu manchen Funktionalitäten des Softwaresystems die internen Abläufe und Interaktionen zu modellieren, haben wir zu zwei Funktionen ebenfalls jeweils ein Aktivitätsdiagramm erstellt.

Angemeldete Kurse einsehen:

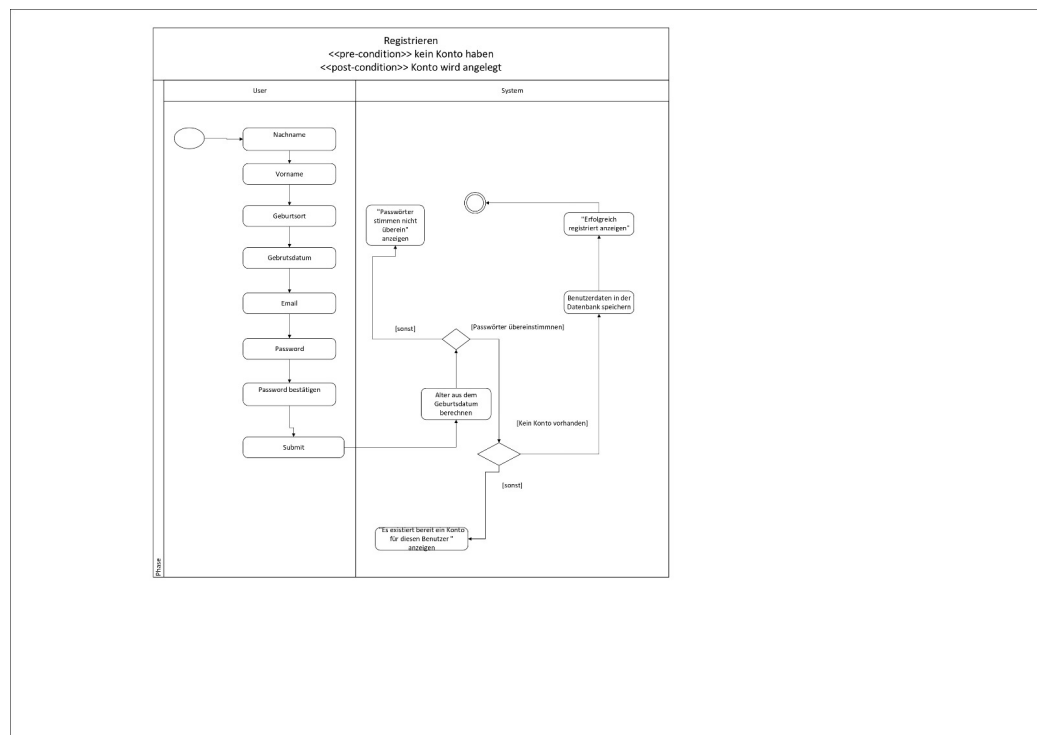


Das folgende Aktivitätsdiagramm modelliert den internen Ablauf dafür, wie nach dem Klicken auf **Angemeldete Kurse einsehen**, die angemeldeten Kurse für ein Benutzer in Form einer Liste gezeigt werden.



Die Aktivität fängt erst an, wenn der Benutzer auf die Schaltfläche **Angemeldete Kurse einsehen** klickt. Nach dem der Benutzer auf **Angemeldete Kurse einsehen** geklickt hat, wird der UserID des Benutzers aus der Sitzung an den entsprechenden PHP-Skript übergeben. Anschließend wird geprüft ob der Kunde ein Fitnesskurs gebucht hat. Und solange gebuchte Fitnesskurse für den übergebenen UserID gefunden werden, werden die auch dem Benutzer in Form einer Tabelle gezeigt. Sonst endet das Aktivitätsdiagramm.

Registrierung:

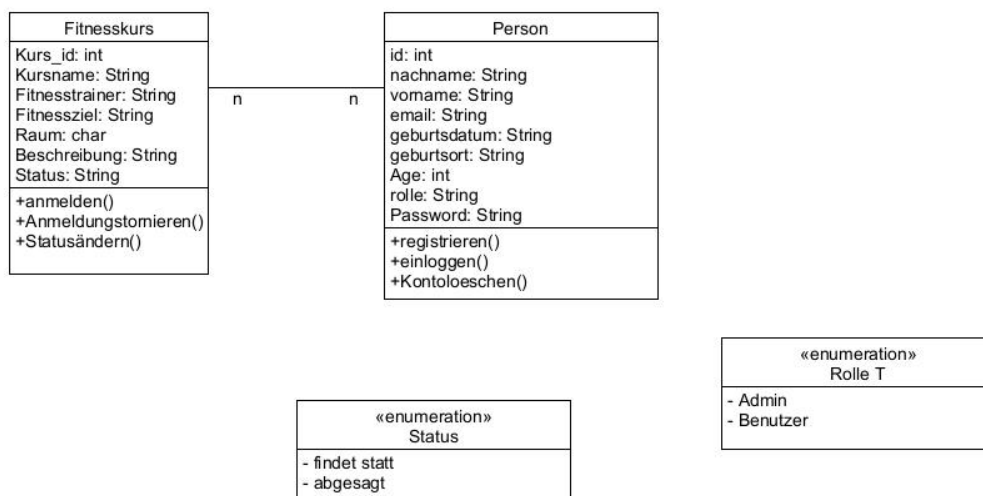


Das folgende Aktivitätsdiagramm modelliert den internen Ablauf dafür, wie nach dem Klicken auf **Registrierung**, die Seite wo der Benutzer sich zum ersten Mal registriert, auftaucht. Die Aktivität fängt erst an, wenn der Benutzer auf die Schaltfläche **Registrierung** klickt. Nach dem der Benutzer auf **Registrierung** geklickt hat, wird der Benutzer nach der Eingabe seinem Daten, wie Vorname, Nachname, Geburtsort, Geburtstag, sowie ein Password zwei mal eingeben. Sobald die beide Passwörter übereinstimmen und de Konto in Datenbank nicht existiert, wird die Registrierung erfolgreich passieren.

---

## 2.4 Klassendiagramm

Als Team haben wir es als wichtig empfunden, zu unsere zwei Klassen im Softwaresystem ebenfalls ein Klassendiagramm erstellen, mit der modelliert wird, in welchen Beziehungen und in welcher Struktur die verschiedenen Klassen und Objekte im Softwaresystem stehen.



In unserer Anwendung gibt es nur zwei Klassen, und zwar Person und Fitnesskurs. Beide Klassen stehen in engem Zusammenhang miteinander. Eine Person kann sich für mehrere Fitnesskurse anmelden und mehrere Personen können am selben Fitnesskurs teilnehmen.

Zu den Spezifikationen der beiden Klassen:

Die Klasse Fitnesskurs besteht aus einem KursID als Integer und einem Kursname. Jedem Fitnesskurs ist ein Fitnesstrainer, ein Fitnessziel, ein Raum, eine kurze Beschreibung und ein Status zugewiesen. Da kann man sich anmelden, die Anmeldung für einen Kurs stornieren und den Status vom Kurs ändern. Die Klasse Person macht ein ID, ein Nachname, ein Vorname, eine E-Mail, ein Geburtsdatum, ein Geburtsort, ein Alter, eine Rolle und ein Password aus. Eine Person kann registrieren, sich einloggen und es können auch Personen bzw. Konten gelöscht werden (Admin).

---

## 2.5 Realisierungsbeschreibung

## 2.6 Verwendete Techniken

Die benutzten Technologien für das Projekt sind HTML, CSS, JavaScript, PHP, SQL, Redis und Git. Im Folgenden werden die verwendeten Technologien kurz aufgelistet und erklärt. Im Nachhinein wird darüber diskutiert, wie die Technologien bei der Verwendung in Zusammenhang stehen.

### 2.6.1 HTML und CSS - Hossein

Geht man an den Ursprüngen des Internets heran, so findet man heraus, dass ca. in den achtziger Jahren, damit man in der Lage ist Datenübertragung zu betreiben, man dafür mehrere Techniken zur Hand haben musste. Hierfür wurde 1. HTTP (Hyper Text Transfer Protocol) entwickelt, der den Zweck hatte Daten zu übermitteln. Hinzufügend dazu kam ein Server-Software, mit dem Zweck die HTTP-Anfragen der Anwender zu beantworten. Bei der Frage, wie man die Informationen darstellen möchte, kamen die Entwickler auf die Idee, eine Auszeichnungssprache zu entwickeln, mit der man einzelne Textelemente (z.B. Überschriften oder Absätze) deutlich machen, sowie übersichtlicher darstellen kann. Somit kam dann die sogenannte Auszeichnungssprache als HTML (Hyper Text Markup Language) (vgl. Fuchs 2019a: 15-16).

HTML ist jedoch keine Programmiersprache, sondern eine Auszeichnungssprache. Eine Programmiersprache hingegen gestaltet Computerprogramme, wofür sie über Ein- und Ausgabebefehle verfügt. Hierfür kann der Anwender Eingaben machen, oder das Programm kann Daten aus anderen Quellen einlesen. Die Ergebnisse werden entweder über Bild ausgegeben, oder in eine Datenbank gespeichert. Noch hinzufügend ist eine Programmiersprache in der Lage dazu, Daten in Variablen festzuhalten, mit der Rechenoperationen durchgeführt werden können. Zudem lässt sich sagen, dass hierdurch Texte in Zeichenkettenform verarbeitet werden. Zum Schluss beinhaltet eine Programmiersprache Steuerungsanweisungen, bei den Aktionen an Bedingungen gebunden werden können und wiederholt werden können.

HTML hingegen hat diese Funktionen nicht. HTML kann nur bestimmte Textstellen auf dem Browser, als Überschrift oder als Zeile einer Liste anzeigen, oder besser gesagt auszeichnen. HTML sorgt quasi nur dafür, einem Text-Dokument eine Struktur zu geben, sie kann jedoch keinen Ablauf oder Berechnung durchführen (vgl. Fuchs 2019a: 17-18).

---

Zu Anfangszeiten des Internets, als die Nachfrage danach groß war, die Gestaltungsmöglichkeiten der einfachen Webentwicklung, also HTML zu erweitern, wurde neben HTML parallel eine weitere Sprache, von Berners-Lee, mit dem Namen CSS (Cascading Style Sheets), für das Layout der Seite entwickelt (vgl. Fuchs 2019a: 16-17).

### **2.6.2 HTML und CSS: Statische und dynamische Webseiten - Hossein**

Jedes Mal, wenn eine Internetseite aufgerufen wird, dadurch, dass ein Benutzer auf eine Internetseite raufgeht, wird durch den Browser, durch eine Anfrage, vom Server, eine Datei angefordert. Die Internetseite stellt daraufhin die Inhalte der Datei auf dem Browser dar. Hierzu unterscheidet man jedoch zwischen statischen und dynamischen Webseiten:

Unter einer statischen Webseite versteht man, wenn die Datei, die aufgerufen wird, immer die gleichen Inhalte liefert. Hierbei wird der Inhalt niemals verändert, egal von wem oder zu welchem Zeitpunkt die Datei aufgerufen wird. Möchte man den Inhalt der Datei verändern, so muss man in die Datei manuell eingehen und durch einen manuellen Eingriff den Inhalt verändern, diese neue Änderung speichern, die Datei aktualisieren und diese dann letztendlich wieder hochladen.

Bei dynamischen Webseiten ist die Datei, die aufgerufen wird, zwar immer die gleiche Datei, die Inhalte dessen hingegen sind veränderbar. Die Inhalte der Datei, sowie das Layout werden durch ein Computerprogramm verändert. Die programmierten Computerprogramme rufen Daten aus der Datenbank auf, die dann wiederum in die Seite eingefügt werden (vgl. Fuchs 2019a: 19).

Möchte man eine dynamische Webseite programmieren, so sind hierfür als erstes die Programmiersprachen PHP und JavaScript nötig. Für die genauere Gestaltung einer dynamischen Webseite, muss man ein Programm schreiben, der aus irgendeiner Datenbank, Dateien oder anderen Internetseiten, Informationen aufruft und diese dann in Form einer neuen Seite, hochlädt. Möchte man dies tun, so muss das Programm eine HTML-Seite erzeugen, die dann an den Browser geleitet wird und dort angezeigt wird. Bei der Verwendung von PHP müssen die Ausgabebefehle, HTML-Code erzeugen (vgl. Fuchs 2019a: 20-21). Somit lässt sich sagen, dass HTML für die Erstellung von dynamischen Webseiten eine wichtige Bedeutung hat und somit als Grundlage für die Programmierung von PHP, JavaScript und anderen Sprachen gilt.

---

Da wir bei unseren Projekt eine dynamische Webanwendung bauen wollen, ist dies auch der Grund für die Verwendung von HTML und CSS als Grundlage für die Verwendung von PHP und JavaScript.

### **2.6.3 JavaScript - Hossein**

Die Programmiersprache JavaScript wird offiziell als General Purpose Language (GPL) eingeordnet, die meistens mit dem Zweck verwendet wird, dynamische Webseiten zu entwickeln, mit der man bei der Webseite den Aufbau, das Layout und den Inhalt der Webseite verändern kann. Durch diese große Spezialisierung, mit der sich die Sprache groß durchgesetzt hat, gilt JavaScript, neben HTML und CSS, als eine wichtige Web-Technologie (vgl. Fuchs 2019b: 14-15).

Noch wichtig zu wissen ist, dass JavaScript niemals als Ersetzung von HTML betrachtet werden sollte, sondern eher als Ergänzung dazu. Hierbei versteht man, dass JavaScript Elemente in den HTML-Code reinimplementiert werden, wodurch dann der Aufbau, das Layout und der Inhalt der Webseite verändert werden (vgl. Fuchs 2019b: 20).

### **2.6.4 JavaScript: Entstehung - Hossein**

Zu den Anfangszeiten des Internets, als die Webentwicklung sich entwickelte, kam man auf die Idee, dass für die weitere Entwicklung, um dies zu beschleunigen, Programmierer zu mehr fähig sein müssten, als nur statische Webseiten zu bauen. Sie müssten mehr die Möglichkeit haben, dynamische Webseiten zu programmieren, für die man die Inhalte von Internetseiten, dynamisch zu verändern. Daher kam die Idee auf, eine weitere Sprache als Ergänzung zu HTML zu programmieren, mit der man das machen könnte.

Die neue Sprache sollte in den HTML-Code rein implementierbar sein, um damit den HTML-Inhalt dynamisch ändern zu können. Außerdem sollte die Sprache, für Webdesignern, die nicht viel Ahnung von Programmieren hatten, leicht erlernbar und benutzbar sein (vgl. Fuchs 2019b: 16-17).

---

### 2.6.5 JavaScript: Serverseitige und clientseitige Anwendungen - Hossein

Spricht man davon, dass man dynamische Internetseiten erstellen möchte, so muss hierfür ein Computerprogramm erstellt werden, dessen Inhalte an den individuellen Anforderungen angepasst werden müssen. Dieser Prozess geschieht an zwei Orten:

Der eine Ort hierfür ist der Server des Anwenders. In diesem Gebiet spricht man von der sogenannten serverseitigen Programmierung, wobei bei jedem Aufruf eine neue Seite erzeugt wird, wofür wiederum immer ein neuer HTML-Code generiert wird. Hierdurch können die Inhalte dynamisch angepasst werden. Sobald die Seite jedoch geladen wurde, wird die Seite statisch und weitere Änderungen sind nicht mehr möglich. Die Dynamik wird dadurch gewährleistet, dass die Inhalte bei der Erstellung je nach Bedarf zusammengestellt werden, wofür Datenbankwerte verwertet und bei jedem Aufruf eine individuelle Seite erstellt wird.

Der andere Ort hingegen ist der Rechner des Anwenders. In diesem Gebiet spricht man von der sogenannten clientseitigen Programmierung. Spricht man hingegen von clientseitigen Anwendungen, so ist gemeint, dass dem Besucher, vom Server ein Programm übergeben wird, die von dem Besucher dann auf dem eigenen Computer ausgeführt wird. Hierdurch wird die Erstellung von dynamischen Webseiten erlaubt, die ebenfalls für den Endbenutzer dynamisch ist.

Entscheidet man sich für die Verwendung serverseitig gestalteter Webseiten, so genießt man den Vorteil, dass man hierdurch Zugriff auf Datenbanken und Daten hat, die auf dem Server gespeichert sind. Zusätzlich gibt es den Vorteil, dass bei der Verwendung dieser, die Sicherheit hoch ist, da der Benutzer das Programm auf dem eigenen Server ausführt.

Entscheidet man sich hingegen für clientseitige Programmierung, genießt man den Vorteil, dass hierdurch die Einstellungen des Benutzers, die individuell sind, unter Berücksichtigung genommen wird. Als Beispiel hierzu zeigt ein clientseitiges Programm immer die aktuelle Uhrzeit in der Zeitzone, in der sich der Benutzer ebenfalls befindet, wobei dies bei der serverseitigen Programmierung nicht der Fall ist. Als Zusatz kommt hinzu, dass man die Werte aktualisieren kann, ohne dass man die Seite neu laden muss. Dadurch, dass es zwischen Server und Client weniger Kommunikation erforderlich ist, kommt es bei der clientseitigen Programmierung auch zu weniger Datenverkehr.

---

Es lässt sich sagen, dass JavaScript hauptsächlich für die clientseitige Programmierung zum Einsatz kommt. Möchte man mit der Verwendung irgendwelche Werte dauerhaft speichern, so ist JavaScript dafür eher weniger geeignet. Stattdessen wird JavaScript für gestalterische Elemente und integrierte Funktionen in Webseiten verwendet, Beispielsweise Währungsrechner. Unter der Bedingung, dass diese nicht sicherheitskritisch sind, kommt die Verwendung von JavaScript auch oft für die Validierung von Formulardaten zum Einsatz.

Möchte man in der Praxis programmieren, so ist die Verwendung serverseitiger Programmierung, als auch die Verwendung clientseitiger Programmierung üblich. Möchte man von der serverseitigen Programmierung profitieren, so kommt hierzu meist die Programmiersprache PHP zugute, wodurch HTML-Dokumente erstellt werden.

Bei der Verwendung von JavaScript ist die Implementierung von AJAX (Asynchronous JavaScript and XML) viel beliebt, da es hierdurch möglich ist, zwischen Server und Client Daten auszutauschen, ohne dafür die Seite komplett neuzuladen. Hierdurch versteht man erneut eine Verbindung zwischen serverseitiger und clientseitiger Programmierung, wofür JavaScript eine essentielle Grundlage darstellt (vgl. Fuchs 2019b: 20-23).

### **2.6.6 PHP - Hossein**

Damals im Jahre 1995 hatte der Programmierer Rasmus Lerdorf, damit er die Zugriffe auf seinen Online-Lebenslauf aufzeichnen konnte, in der Programmiersprache Perl, ein Skript geschrieben. Als Resultat davon entstand hierdurch mit der Zeit eine Ansammlung von Skripten, mit dem Zweck, dass er seine eigene Homepage programmiert. Hierdurch entstand der Name PHP (Personal Home Page Tools). Durch das neue Entwicklerteam im Nachhinein hatte die Abkürzung eine neue Bedeutung, nämlich Hypertext Preprocessor.

Zu dem Zeitpunkt, als das World Wide Web sich immer weiterentwickelte, brauchte man für die Gestaltung dynamische Webseiten, eine gut geeignete Programmiersprache. Aufgrund ihrer Spezialisierung setzte sich PHP mit seiner Version 4 als Standardsprache für dynamische Webanwendungen gut durch, und ist heute deshalb die Grundlage dafür. Mit der Entwicklung ihrer nächsten Version, machten die Entwickler PHP dann nun zu einer objektorientierten Programmiersprache.

---

Da PHP auf fast allen Webservern im Voraus schon installiert ist, eignet sich die Sprache gut für serverseitig programmierte dynamische Webseiten. Durch das Ergebnis einer Studie, die aufzeigte, dass im Jahre 2017 ca. 83 Prozent aller Internetauftritte mit PHP programmiert worden sind, wird schnell klar, wie wichtig PHP für die Entwicklung des Internets ist (vgl. Bonacina 2019: 12-13).

### **2.6.7 SQL-Datenbanken - Hossein**

Neben PHP spielen für die Entwicklung dynamischer Webanwendungen, bzw. für die Speicherung von Daten, auch SQL-Datenbanken eine wichtige Rolle, weshalb es wichtig ist zu verstehen, was man unter einer Datenbank versteht, welche Möglichkeiten der Anwendung dieser bietet, und in welchem Zusammenhang die Datenbanksprache SQL damit steht.

Spricht man von einer Datenbank, so spricht man immer von zwei Teilen: Einerseits geht es dabei um die Daten, die gespeichert werden. Dies kann der Anwender der Datenbank abrufen, verändern, sowie löschen. Andererseits spricht man von den sogenannten Datenbankmanagementsystem (DBSM), in der die eigentlichen Verwaltungsarbeiten der Daten erledigt werden (vgl. Bonacina 2019: 15-17).

### **2.6.8 MySQL als wichtigstes und beliebtestes Datenbanksystem - Hossein**

Ein wichtiges Datenbankmanagementsystem ist hierbei die sogenannte MySQL. MySQL wird aufgrund ihrer Beliebtheit, sowie seiner großen Bedeutung, schon von großen IT-Unternehmen wie Facebook, YouTube, oder Twitter genutzt. Kommt es zur Benutzung von MySQL, so genießt man gleichermaßen den Vorteil, dass MySQL gut mit der Programmiersprache PHP arbeitet (vgl. Bonacina 2019: 134).

Der Ursprung für die Gründe, weshalb MySQL als Datenbanksystem so wichtig und beliebt ist, in der aktuellen Zeit, hat etwas damit zu tun, wie in den späten 90er Jahren das Internet sich ausgebreitet hat. Ab Ende der 90er Jahre nahm die Anzahl der dynamisch entwickelten Webanwendungen im Internet groß zu, wofür gleichzeitig dann auch hierfür ein Datenbanksoftware nötig war. Aufgrund dessen, weil durch eine Open-Source-Software die Kosten für das Betreiben einer Internetseite zu sinken möglich war, kam der Begriff LAMP in seiner zentralen Rolle.



---

Die Abkürzung LAMP steht für das Betriebssystem Linux, den Webserver Apache, das Datenbanksystem MySQL und die Programmiersprache PHP. Hierbei handelt es sich um vier lizenzfreie Techniken, mit denen sich auf eine simple und gute Weise, dynamische Webseiten entwickeln lässt. Aufgrund dieser Chronologie wiederum entwickelte sich MySQL zu dem am häufigsten verwendeten Datenbanksystem der Welt (vgl. Fuchs 2020: 28-29).

### **2.6.9 SQL als Datenbanksprache - Hossein**

Spricht man außerdem über die Strukturierung der Daten, bei SQL-Datenbanken, so spricht man von sogenannten relationalen Datenbanken, wobei die Daten in Tabellen dargestellt werden.

Möchte man beim DBMS auf die Daten, die dort gespeichert sind zugreifen, so kommt die Datenbanksprache SQL (Structured Query Language) zum Einsatz, mit der Hauptaufgabe, die Daten, die in DBMS gespeichert sind, durch die vorhandenen SQL-Befehle, mit denen man die Daten anfordern, einfügen, löschen und ändern kann, abzufragen (vgl. Bonacina 2019: 15-17).

### **2.6.10 Redis - Hossein**

Spricht man von Redis, so versteht man darunter eine leistungsstarke in-memory Remote-Datenbank, der dem Anwender viele umfangreiche Funktionen bietet. Des Weiteren unterstützt werden dabei fünf verschiedene Arten von Datenstrukturen, mit der man eine breite Menge an Problemen effizient lösen kann. Hinzufügend lässt sich sagen, dass Redis hohe Leistung bietet, sowie auch Replikation und Skalierbarkeit, angenommen von der Prototypenherstellung, bis hin zur Verarbeitung von großen Datenmengen, sowie hoher Anfragequote. Hierdurch wird eine einfache Lösung von großer Herausforderung ermöglicht, ohne der komplexen Abstraktionen von anderen Datenbanksystemen (vgl. Carlson 2013: 3-4).

### **2.6.11 Git: Nutzung von Versionsverwaltungssystem - Hossein**

Schreibt man als Softwareentwickler einfache Programme, so ist es hierbei leicht, dass man den Überblick über die bereits geschriebenen Codezeilen nicht verliert, weshalb die Verwaltung der bereits geschriebenen Programme nicht schwierig ist. Hierbei handelt es sich meistens um eine einzige Datei, in der weniger als hundert Zeilen Code geschrieben worden ist. Möchte man bei diesem wenigen Codeumfang, den Code verwalten, dabei nach Fehlern zu suchen, oder etwas dazu zu ergänzen, so lässt sich das innerhalb weniger Minuten tun.

---

Möchte man auf einer professionellen Ebene hingegen programmieren, so merkt man schnell, dass das geschriebene Programm nach kurzer Zeit viel umfangreicher wird. Nimmt man dabei Rücksicht darauf, dass die geschriebenen Programme 1. deutlich mehr als hundert Zeilen Code beinhalten, sowie 2. Mehrere Dateien beinhalten, so ist das Einhalten des Überblicks in der Programmierung viel schwieriger. Kommt es vor, dass man bei einigen Fehlern oder Überarbeitungen nicht mehr weiterkommt und diese ebenfalls nicht lösen kann, so hat man durch ein Versionsverwaltungssystem die Möglichkeit, den geschriebenen Code viel leichter zu verwalten, dabei den Code neu erstellen, die alte Version des Codes wiederherzustellen, die letzten kleinen Arbeitsschritten rückgängig zu machen, etc., um damit Nebenwirkungen, die in der Wirtschaft mit massiven Schadenskosten verbunden sind, ebenfalls zu vermeiden.

Mit einer Versionsverwaltungssystem hat man, grob gesagt, in der Softwareentwicklung, die Möglichkeit, die einzelnen Versionen des Codes automatisch zu speichern, mit der Bereitstellung vieler andere Funktionen, wie zum Beispiel das Erstellen von verschiedenen Entwicklungszweigen, die Zusammenführung dieser im Nachhinein, und so weiter... Diese Möglichkeit erleichtert die Softwareentwicklung auf einer professionellen Ebene enorm.

Einer dieser Verwaltungssysteme, die in der Softwareentwicklung meist zum Einsatz kommt, ist Git (vgl. Fuchs 2021: 7-9).

#### **2.6.12 Git: Aufgaben der Versionsverwaltung - Hossein**

Möchte man sich genauer die Aufgaben der Versionsverwaltung anschauen, so lässt sich sagen, dass ein Versionsverwaltungssystem in der Lage sein muss, die einzelnen Dateien, die während der Softwareentwicklung entstehen, zu archivieren. Und dies gilt für alle Versionen, die im Verlaufe der Entwicklung eines Softwaresystems, die im Laufe der Zeit entstehen.

Darüber hinaus muss das Softwaresystem in der Lage sein, die verschiedenen Änderungen eines Softwaresystems zu protokollieren. Hierbei müssten bei jeder Änderung gekennzeichnet werden, welcher Softwareentwickler zu welchem Zeitpunkt, eine Änderung vorgenommen hat. Dadurch soll zum Beispiel der Beteiligungsumfang eines Softwareentwicklers für ein Softwaresystem zu erkennen sein.

---

Des Weiteren hat die Versionsverwaltung die Aufgabe, immer die aktuelle Version, also das Dokument, dass die neueste Entwicklung des Dokumentes zeigt, als Grundlage für die Weitere Entwicklung vorzulegen. Kommt es mal vor, dass die aktuelle Version nicht mehr brauchbar ist, muss die Versionsverwaltung die Möglichkeit geben, eine frühere Version bereitzustellen, um Diese als die aktuelle Version zu markieren.

Schließlich hat die Versionsverwaltung die Aufgabe, die Koordination der Arbeit zwischen mehreren Entwicklern zu ermöglichen, und hierzu die wichtigen Elemente wie Protokollierung, Rechteverwaltung, sowie klare Regelungen für das Lösen von Konflikten zwischen Entwicklern, sowie die Bestimmung der Hauptversion, zu beinhalten (vgl. Fuchs 2021: 9-10).

### **2.6.13 Git: Entstehungsgeschichte - Hossein**

Ursprünglich betrachtet entstand Git aus dem Grund, da während der Entwicklung von Linux, was als ein sehr umfangreiches Projekt angesehen wurde, dadurch die Notwendigkeit eines Versionsverwaltungssystems hervorgerufen worden ist.

In den Anfangszeiten verwendete das Linux-Team das System BitKeeper. Da diese mit der Zeit jedoch nicht mehr Open-Source gewesen ist, wohingegen die Entwickler jedoch auf ein Open-Source-Software, für die sich nichts bezahlen mussten, angewiesen waren, entwickelte Linus Torvalds ein eigenes Versionsverwaltungssystem, das als ein verteiltes System so konzipiert worden ist, dass es ähnliche Arbeitsabläufe wie BitKeeper, sowie hohe Sicherheit gegen Verfälschungen zu bieten hatte.

Die Entwicklung von Git dauerte nur wenige Tage und wurde schnell zu einem bedeutenden Bestandteil der Open-Source-Entwicklung (vgl. Fuchs 2021: 14-16).

---

## 2.7 Implementieren der Techniken zu einem Softwaresystem

Ab diesem Abschnitt geht es darum, über den in den Vorherigen Abschnitt erwähnten Techniken, die zu einem Softwaresystem zusammen implementiert sind, zu erzählen.

### 2.7.1 HTML in Anwendung bei unserem Projekt - Junior



```
1 <!DOCTYPE html>
2 <html lang="de">
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1 maximum-scale=1.0, user-scalable=no">
6     <title id="title">Home</title>
7     <script type="text/javascript" src="script.js"></script>
8     <link rel="stylesheet" type="text/css" href="styles.css">
9   </head>
10  <body>
11    <header></header>
12    <nav align="center" id="navbar"></nav>
13    <br><br>
14    <section>
15      <div id="main" align="center" ></div>
16    </section>
17    <br><br><br>
18    <footer>
19      <p class="eingabe" style="font-size:20px">2023 &copy; <i>TFW-2023-F</i></p>
20    </footer>
21  </body>
22 </html>
```

Da es sich bei unserem Projekt um eine webbasierte Anwendung handelt, war es für uns notwendig, HTML zu verwenden, damit unsere Anwendung von einem Webbrowser gelesen und interpretiert werden kann. Dennoch haben wir uns für eine Art von Webanwendung, die mit dem Nutzer interagiert und nicht nur statisch bleibt, daher sieht die Grundstruktur unserer Index-Seite sehr einfach aus und es ist auch tatsächlich so, dass wir die am einfachsten gestaltet haben, weil wir die Möglichkeit haben wollten das Verhalten unserer Seite je nach Nutzerinteraktion zu ändern bzw. anzupassen.

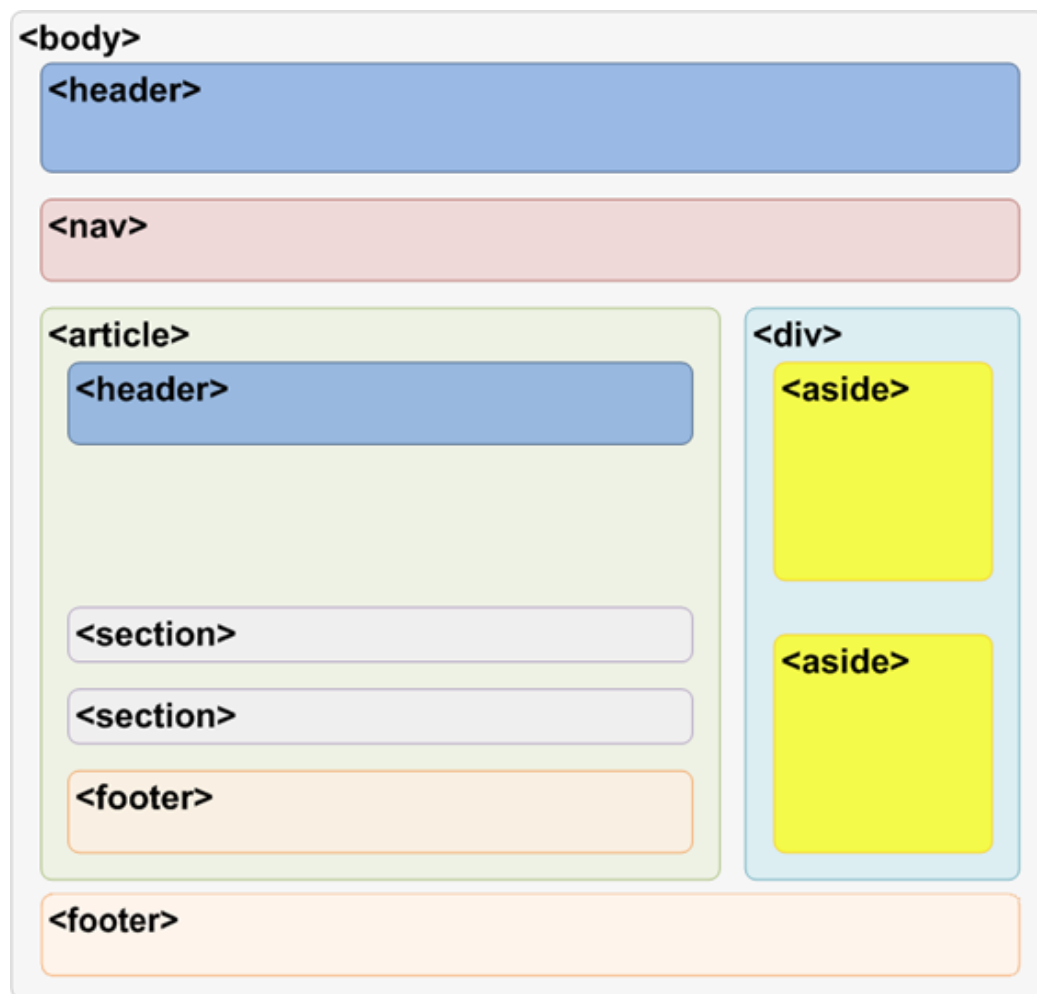
Definiert sind hier nur einiges wie etwa der Titel der Seite, die Sprache der Seite, zwei Verlinkungen, die jeweils zu einer JavaScript-Datei namens `script.js` und zu einer CSS-Datei namens `styles.css`.

Den Körper der Seite (Body) wurde in drei Teilen zerlegt und zwar noch ein Kopf, ein Teil für die Navigationsleiste, ein Abschnitt (auf Englisch: Section entspricht wortwörtlich dem Körper vom "Body") und gut zuletzt haben wir auch einen Fuß (auf Englisch: Footer) für den Körper definiert.

---

Bei allen Seiten ist eine richtige Codierung wichtig, daher steht im Bild oben diesen folgenden Tag: `<meta charset='UTF-8'>`. Dadurch wird sichergestellt, dass die in UTF-8 (Unicode Transformation Format 8-bit) korrekt dargestellt werden.

Unsere Seite haben wir so gestaltet nicht nur, weil es zu unseren Wünschen passen, sondern auch weil es die Basis von HTML5 bildet.



(*BlogNT : le Blog des Nouvelles Technologies* 2011)

Mit unseren vordefiniert und bereits zugewiesenen IDs und Klassen stellen wir sicher, dass wir im nächsten Schritt mit JavaScript unsere Seite, so beeinflussen können wie wir es wünschen.

---

Anwendung von CSS - Es gibt drei Arten von CSS Anwendungen:

Ein gutes Beispiel hierfür wäre Folgendes: `<h1 style="color: red"> Eine Überschrift mit einem Layout im HTML-Tag</h1>` (vgl. Fuchs 2019a: 181).

Diese Methode ist unkompliziert, birgt aber die Gefahr, dass man die gleichen Eigenschaften, die man bereits nur einem Tag vergeben hat, noch einmal braucht.

Durch das Vornehmen der CSS-Angaben im Kopfbereich:

```
-----
1.<style>
2.  Selektor 1, Selektor 2, ... Selektor n {
3.  Eigenschaft 1: Wert 1;
4.  Eigenschaft 2: Wert 2;
5.  .
6.  .
7.  Eigenschaft m Wert m;
8.  }
9.</style>
-----
```

(vgl. Fuchs 2019a: 186).

Eine andere Möglichkeit besteht darin, die CSS-Vorgaben im head-Bereich der HTML-Datei anzubringen. Hier kann man das Layout für jedes einzelne HTML-Tag, das im body-Bereich steht, vorgeben. Das sorgt bereits dafür, dass das Layout deutlich besser von den Inhalten getrennt ist. Daher ist diese Vorgehensweise wesentlich besser, als die Vorgaben in den einzelnen Tags anzubringen. Darüber hinaus kann man sich auf diese Weise viel Arbeit sparen, da man die Tags zu Gruppen zusammenfassen kann. Das heißt, dass eine CSS-Angabe ausreicht, um viele verschiedene HTML-Tags zu beeinflussen. Um die Layoutvorgaben innerhalb des Kopfbereichs zu machen, muss man hier das style-Tag einfügen. Dieses umschließt alle CSS-Vorgaben, die wir machen wollen. Die Reihenfolge innerhalb des head-Bereichs ist dabei nicht von Bedeutung. Man kann das style-Tag also entweder vor den Meta-Tags und dem Titel anbringen oder danach.

---

Durch die Erzeugung eines separaten CSS-Dokuments:

Darüber hinaus gibt es eine dritte Möglichkeit, um CSS-Vorgaben für ein Dokument zu machen. Man kann diese auch in eine separate Datei schreiben. Anschließend muss man diese lediglich in der HTML-Datei verlinken. Das führt dazu, dass der Browser alle Vorgaben, die wir in der externen Datei gemacht haben, bei der Darstellung der Seite berücksichtigt. Wenn man sich für diese Alternative entscheidet, sind die Inhalte vollständig von den Layout-Vorgaben getrennt. Das sorgt für viele Vorteile - insbesondere bei der Gestaltung größerer Websites. Wenn man beispielsweise das Design einer Website mit vielen Hundert Unterseiten überarbeiten will, wäre es mit einem riesigen Aufwand verbunden, diese Änderungen in jedem einzelnen Dokument vorzunehmen. Wenn man hingegen eine externe Datei verwendet, kann jede einzelne Unterseite auf diese zugreifen. Sollte man nun eine Änderung an ihr vornehmen, wirkt sich diese auf alle Seiten aus, die eingebunden wurden. Das stellt eine riesige Arbeitersparnis dar. Aufgrund dieser Vorteile ist es sinnvoll, immer externe CSS-Dateien zu verwenden.

Um eine CSS-Datei zu erstellen, muss man im Texteditor ein leeres Dokument öffnen. Hier werden nun genau die gleichen Vorgaben eingefügt, die im vorherigen Beispiel zwischen den style-Tags standen. Die Tags selbst sind dabei jedoch nicht notwendig. Der Inhalt der CSS-Datei sieht demnach so aus:

```
-----  
1.h1, p {  
2.  color: red;  
3.  font-size: 12px;  
4.}  
5.h2, p {  
6.  background-color: turquoise;  
7.}  
-----
```

Jetzt müssen diese Spezifikationen als CSS gespeichert werden. Dazu kommt die Endung .css zum Einsatz. Es ist üblich, die Layoutvorgaben in einer Datei mit dem Namen style.css abzuspeichern. Wenn diese Aufgabe erledigt wurde, ist die externe CSS-Datei bereits fertiggestellt. Nun muss man diese Datei noch in die HTML-Seite einbinden. Hierfür sind im Gegensatz zum vorherigen Beispiel keine style-Tags notwendig. Anstatt dessen kommt ein link-Tag zum Einsatz. Dieses enthält zum einen das Attribut. rel='stylesheet'.

---

Dieses zeigt an, dass es sich hierbei um ein Stylesheet für die Layout-Vorgaben handelt. Zum anderen ist es wie bei Links auf andere Seiten notwendig, das href-Attribut einzufügen und hier die entsprechende Datei anzugeben. Wenn man sie in einem anderen Ordner gespeichert hat, ist zusätzlich der Pfad notwendig. Das ist in diesem Beispiel jedoch nicht der Fall. Daher sieht das Link-Tag so aus:

```
<link rel='stylesheet' href='styles.css'> (vgl. Fuchs 2019a: 189).
```

### 2.7.2 CSS in Anwendung bei unserem Projekt - Junior

Bei unserem Projekt haben wir uns dafür entschieden unseren CSS-Code von unserem HTML-Code zu trennen, damit effizienter gearbeitet werden kann. Beim Codieren haben wir uns erstmal mit dem Aussehen der gesamten Seite auseinandergesetzt. Hierfür haben wir Folgendes benutzt:

```
1  *{
2      padding: 0;
3      margin: 0;
4  }
5  .offline.inverted {
6      -webkit-filter: invert(100%);
7  }
8  html{
9      height: 100%;
10 }
11 body{
12     height: 100%;
13     background: radial-gradient(#33383f, rgb(42, 47,53));
14     background-repeat: no-repeat;
15     background-attachment: fixed;
16     background-size: cover;
17 }
```

Von der Zeile 1 bis 4: Auffällig ist erstmal da die Anwesenheit eines Sternchens (\*). Es ist ein sogenannter Universal-Selektor, der auf alle HTML-Elemente in einer Webseite abzielt. Die darin enthaltenen Eigenschaften padding und margin werden auf null gesetzt. Dieser CSS-Code blockiert die standardmäßigen Abstände für alle HTML-Elemente auf null. Das bedeutet, dass sowohl der Innenabstand als auch der Außenabstand auf null gesetzt werden. Dies hilft, konsistente und vorhersehbare Layouts zu erstellen, indem es sicherstellt, dass keine unerwünschten Abstände zwischen den HTML-Elementen auftreten.

Von der Zeile 5 bis 7: Dieser CSS-Code definiert eine Regel für HTML-Elemente, die sowohl die Klassen offline als auch 'inverted' haben. Er verwendet das '-webkit-filter'-Attribut, um einen visuellen Effekt zu erzeugen.



---

In diesem Fall wird die Farbinvertierung mit `'invert'` angewendet, was dazu führt, dass die Farben des betreffenden Elements vollständig invertiert werden. Dies kann nützlich sein, um visuelle Effekte auf bestimmte Elemente anzuwenden, beispielsweise um den Bildschirminhalt umzukehren. Das haben wir eingefügt, weil der Entwickler-Tool vom Browser uns das vorgeschlagen hat.

Von der Zeile 8 bis 10: Dieser CSS-Code setzt die Höhe des HTML-Elements auf 100%. Das HTML-Element ist das Äußerste Element unserer Webseite. Das Setzen der Höhe auf 100% stellt sicher, dass es immer die volle Höhe des sichtbaren Bereichs des Browserfensters einnimmt.

Von der Zeile 11 bis 17: Hier wird die Gestaltung des `'body'`-Elements festgelegt. Die Höhe wird ebenfalls auf 100% gesetzt, um sicherzustellen, dass der gesamte sichtbare Bildschirmbereich abgedeckt wird. Der Hintergrund des `'body'` wird mit einer radialen Verlaufsfüllung erstellt, die von einer dunklen Farbe zu einer etwas helleren Farbe `rgb` verläuft. Die `'background-repeat'`-, `'background-attachment'`- und `'background-size'`-Eigenschaften werden verwendet, um sicherzustellen, dass der Hintergrund keine Wiederholung zeigt, an das Fenster angeheftet ist und den gesamten verfügbaren Raum abdeckt.

Nachdem wir diesen Aspekt für die Seite berücksichtigt und erledigt hatten, haben wir den Fokus auf die Navigationsleiste gesetzt, denn eine schöne Navigationsleiste ist aus Nutzersicht immer angenehmer. Hierfür haben wir uns für folgenden Design entschlossen:



---

Da unser Design relativ einfach ist, war auch die Implementierung einfach und schnell:

```
46 ▼ ul{
47     list-style-type: none;
48     margin: 0;
49     padding: 0;
50     overflow: hidden;
51     background-color: skyblue;
52     border: 3px solid ;
53     border-radius: 30px;
54 }
55 ▼ li{
56     float: left;
57     align-content: center;
58     padding: 40px;
59     font-size: 30px;
60 }
61 ▼ li:hover{
62     background-color: blue;
63     border-radius: 10;
64 }
65 ▼ .active {
66     background-color: mediumblue;
67 }
68 ▼ .kalago-div{
69     padding: 0;
70     margin: 0;
71     border: 3px solid indianred;
72     border-radius: 5vh;
73     align-items: left;
74     text-align: left;
75 }
```

Von der Zeile 46 bis 54: Unsere Navigationsleiste haben wir als Liste von Elementen beschrieben. Dies wird durch „ul“ (auf Englisch Unordered List) im Code werden sehr viele Eigenschaften beschrieben.

„list-style-type: none;“: Dadurch werden die standardmäßigen Aufzählungszeichen oder Bindestriche aus Listen entfernt , was besonders nützlich ist, wenn Sie eine benutzerdefinierte Listenansicht erstellen möchten.

„margin: 0; und padding: 0;“: Durch das Zurücksetzen von Rand und Abstand auf Null wird der Standardabstand entfernt, der normalerweise auf Listen angewendet wird, und ermöglicht die vollständige Kontrolle über den Abstand von Listen.

overflow: hidden;“: Diese Regel verhindert, dass der Inhalt der Liste über den Rahmen hinausragt. „background-color: skyblue;“: Legt die Hintergrundfarbe der Liste auf Himmelblau fest.

---

”border: 3px solid; und Randradius: 30px;”: fügt der Liste einen festen, pixel-dicken Rand hinzu und rundet sie ab.

Das haben wir eingesetzt, denn es erlaubt uns eine Gestaltungsfreiheit für Listen, einschließlich Anpassung von Aufzählungszeichen und Abständen. Außerdem verbessert es die visuelle Konsistenz der Anwendung durch das zurücksetzen von Rand und Füllung (margin and padding). Dazu erlaubt es auch die Gestaltung einer ansprechenden benutzerdefinierten Ansicht der Liste.

Von der Zeile 55 bis 60: Da wird das Aussehen der Elemente der Liste beschrieben.

- ”float: left;”: Da werden alle Elemente der Liste nebeneinander in einer horizontalen Anordnung platziert.
- ”align-content: center;”: Hier wird der Inhalt der Listenelemente in Bezug auf ihre vertikale Ausrichtung zentriert.
- ”padding: 40px;”: Dadurch wird jedem Element der Liste ein Innenabstand von 40px hinzugefügt.
- ”font-size: 30px;”: Dieses Style legt die Schriftgröße der Listenelemente auf 30px fest.

Mit diesen Eigenschaften konnte eine horizontale Menüleiste oder Navigationsleiste erstellt werden, was die Lesbarkeit und die Benutzerfreundlichkeit durch eine größere Schriftgröße und passende Abstände verbessert.

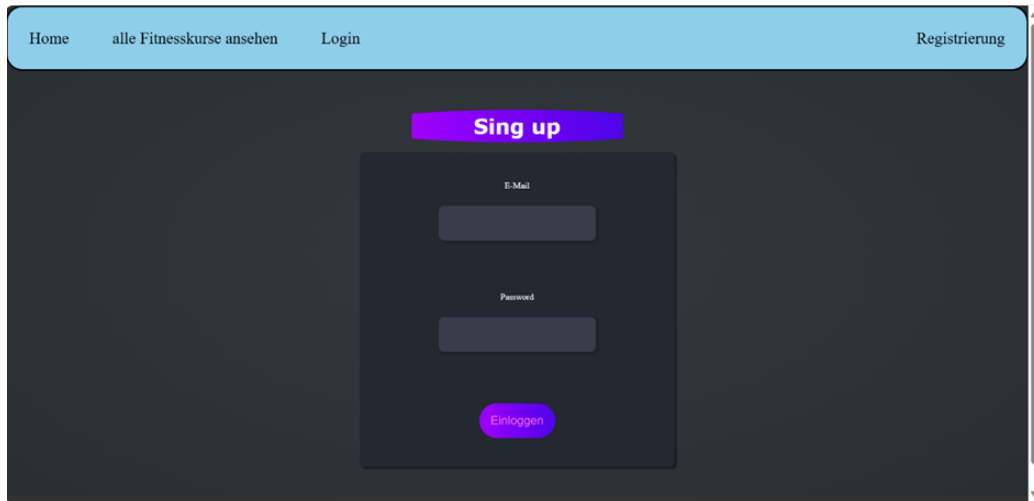
Von der Zeile 61 bis 64 :

- ”background-color: blue; und ”border-radius: 10;”: Dies ändert die Hintergrundfarbe beim Überfahren mit der Maus.

Dadurch wird dem Benutzer klargemacht, auf welchem Element der Leiste er gerade mit der Maus ist. Dies sorgt dann für eine gute Nutzerinteraktion, indem es visuelles Feedback für interaktive Elemente bietet.

---

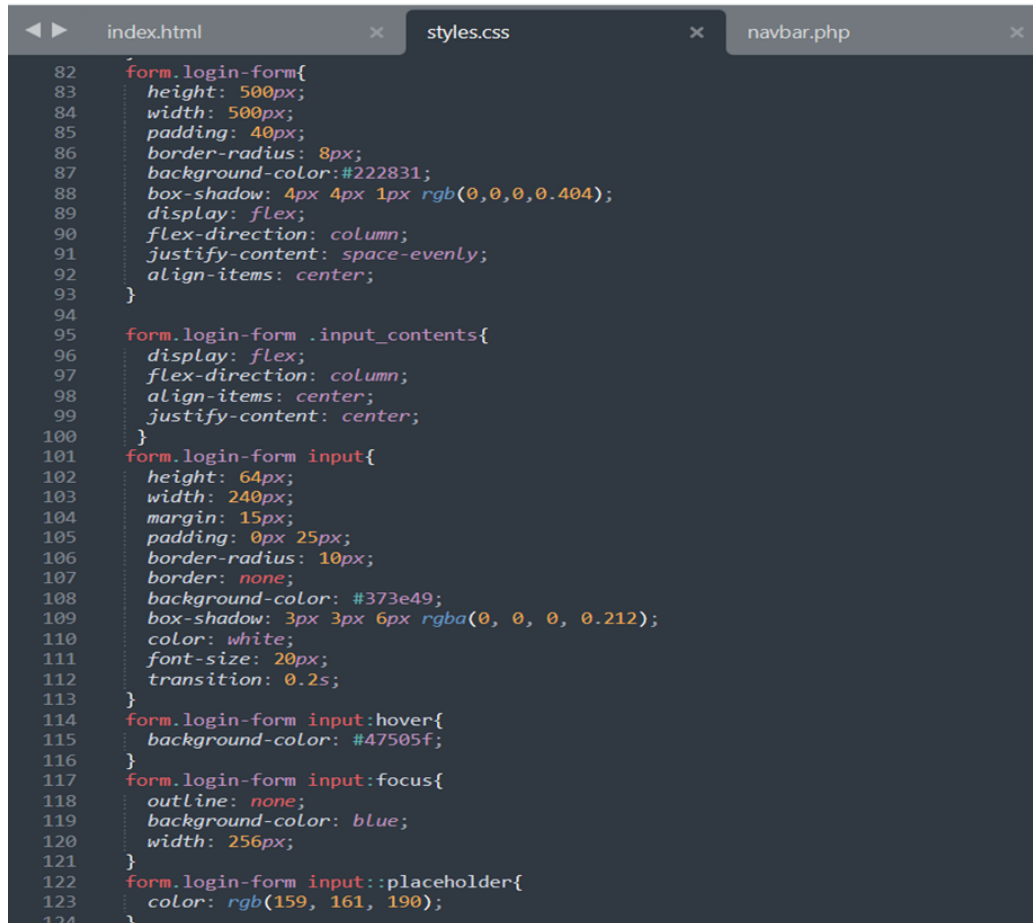
Interessant an unserem Code ist auch das Layout unseres Loginformulars, das mit JavaScript generiert wird:



Beim ersten Blick verliebt man sich sofort darin, was für einen Nutzer nur attraktiv sein kann. Der Code hierfür ist lang und war nicht leicht zu implementieren, weil einige von uns vorher definierten CSS-Eigenschaften andere gestört oder nicht anwendbar gemacht hat. Aber die konnten wir auch später nach sorgfältiger Prüfung vom Code fixen.

Da die Implementierung vom Code sehr zeitaufwendig war, ist der Code entsprechend auch lang. Daher wird im Folgenden nur ein Teil davon abgebildet und auch erläutert. Dabei ist es erstmal wichtig zu betonen, dass wir den Code so geschrieben haben, dass er sich jeder Bildschirmgröße anpassen sollte. Damit die Anwendung sowohl mit einem Computer bzw. Laptop als auch mit einem Tablet-PC oder mit einem Smartphone benutzbar ist.

Hier der Code und seine Erläuterung:

A screenshot of a code editor with three tabs: 'index.html', 'styles.css', and 'navbar.php'. The 'styles.css' tab is active, showing CSS code for a login form. The code is as follows:

```
82 form.login-form{
83   height: 500px;
84   width: 500px;
85   padding: 40px;
86   border-radius: 8px;
87   background-color: #222831;
88   box-shadow: 4px 4px 1px rgb(0,0,0,0.404);
89   display: flex;
90   flex-direction: column;
91   justify-content: space-evenly;
92   align-items: center;
93 }
94
95 form.login-form .input_contents{
96   display: flex;
97   flex-direction: column;
98   align-items: center;
99   justify-content: center;
100 }
101 form.login-form input{
102   height: 64px;
103   width: 240px;
104   margin: 15px;
105   padding: 0px 25px;
106   border-radius: 10px;
107   border: none;
108   background-color: #373e49;
109   box-shadow: 3px 3px 6px rgba(0, 0, 0, 0.212);
110   color: white;
111   font-size: 20px;
112   transition: 0.2s;
113 }
114 form.login-form input:hover{
115   background-color: #47505f;
116 }
117 form.login-form input:focus{
118   outline: none;
119   background-color: blue;
120   width: 256px;
121 }
122 form.login-form input::placeholder{
123   color: rgb(159, 161, 190);
124 }
```

Also ab der Zeile 83 bis 93 wird darauf abgezielt, das Erscheinungsbild vom Login-Formular, das per JavaScript mit einer Zugehörigkeit zur Klasse „login-form“ zu definieren und zu gestalten. Die Höhe und die Breite (auf Englisch: height und width) wird durch auf 500 Pixel gesetzt. Danach wird einen Innenabstand (auf Englisch: padding) von 40 Pixeln rund um den Inhalt vom Formular hinzugefügt. Das Formular wollten wir nicht eckig haben, deswegen wurden die Ecken vom Formular mit einem Radius von 8 Pixeln gerundet. Für das Formular haben wir eine Hintergrundfarbe (auf Englisch: background-color). Wir haben eine dunkle Farbe ausgewählt. Dem Formular haben wir auch Schatten „(box-shadow: 4px 4px 1px rgb(0,0,0,0.404))“ hinzugefügt.

---

Dieser Schatten hat eine leichte Verschiebung von 4 Pixeln nach rechts und 4 Pixeln nach unten sowie eine Weichheit von 1 Pixel. Die Farbe des Schattens ist ein leicht transparentes Grau. Mit dem CSS-Attribut `display: flex` wurde der Display-Stil auf `flex` festgelegt, was bedeutet, dass das Element, auf das dieses Attribut angewendet wird, zu einem sogenannten „flexiblen Container“ wird, der die Anordnung seiner inneren Elemente steuert. In einem flexiblen Container kann man die Ausrichtung, Reihenfolge und Verteilung der inneren Elemente flexibel und responsiv steuern. Daher haben wir das ausgewählt.

Das Setzen auf `flex` war erforderlich, damit wir die Ausrichtung mit `flex-direction: column;` festlegen können, sodass die inneren Elemente unseres Formulars vertikal von oben nach unten angeordnete werden. Und dann mit `justify-content: space-evenly` konnte den verfügbaren Platz gleichmäßig zwischen den inneren Elementen vertikal verteilt werden, wodurch der Inhalt vom Formular in der Mitte ausgerichtet wird. Schließlich wurden alle inneren Elemente horizontal in der Mitte des Formulars zentriert.

Nachdem das grobe Aussehen fertig bearbeitet wurde, mussten wir noch das Layout für die einzelnen Elemente festlegen. Das geschieht von der Zeile 101 bis 123. Im ersten CSS-Stück geht es um das Formular-Container-Styling. Es definiert das Styling für den Container des Login-Formulars. Die Eingabefelder für Benutzername und Passwort werden dabei platziert. Das Styling benutzt die Flexbox-Eigenschaften, um die Eingabefelder nicht nur horizontal, sondern auch vertikal anzuordnen. Es sorgt dadurch für eine ansprechende Ausrichtung der Elemente, die sich innerhalb des Formulars befinden.

Dann folgt das Layout von Eingabefeldern. Dieser Abschnitt beschreibt das Styling der Eingabefelder innerhalb des Login-Formulars. Die Eingabefelder verfügen sowohl über eine festgelegte Höhe und Breite, Ecken, die abgerundet wurden, eine Hintergrundfarbe und einen dezenten Schatten für die visuelle Tiefe. Zudem wird die Schriftfarbe auf weiß gesetzt und ein Übergangseffekt von 0,0 Sekunden wird definiert, damit die Benutzerinteraktion attraktiv gestaltet wird.

---

Wichtig war auch ein Hover-Effekt für die Eingabefelder. Also, wenn der Mauszeiger über ein Eingabefeld schwebt, ändert sich die Hintergrundfarbe leicht. Es wird hellgrau und nicht mehr grau. Dies verleiht dem Formular eine interaktive Benutzeroberfläche, indem wieder ein visuelles Feedback gegeben wird. Und sollte eins der Eingabefelder den Fokus erhalten, zum Beispiel durch das Klicken oder Navigieren mit der Tab-Taste, so wird die Hintergrundfarbe blau und die Breite vom betroffenen Eingabefeld gleichzeitig geringfügig erhöht, um den Fokus visuell zu betonen.

Hier auch mussten wir uns mit dem Platzhalter beschäftigen. Hierbei haben wir uns überlegt, dass wir den Platzhaltertext in einer bestimmten Farbe anzeigen lassen, damit er vom eigentlichen Benutzereingabetext zu unterscheiden. Für die Farbe haben wir einen bläulichen Grauton gewählt. Er wird durch das RGB-Farbsystem definiert, wobei Rot, Grün und Blau angegeben sind. 159 steht da für den Rotanteil, der relativ niedrig ist. 161 steht hierbei für den Grünanteil, der ebenfalls relativ niedrig ist. 190 entspricht hier dem Blauanteil, was auf einen Blaustich hinweist.

Insgesamt haben wir unser Login-Formular so gestaltet, dass es für ein ansprechendes und benutzerfreundliches Erscheinungsbild mit visuellen Effekten, die die Benutzerinteraktion verbessern. Das Formular ist auch leicht verständlich mit klaren Hinweisen, sodass der Benutzer auf Anhieb wissen kann, was zu machen ist.

---

### 2.7.3 JavaScript in Anwendung bei unserem Projekt - Junior

Bei der Implementierung unserer Anwendung hatten wir die Anforderung einer Single-Page-Appkliaktion (auf Englisch: Single-Page-Application abgekürzt: SPA) zu bauen. Aber was ist eigentlich eine SPA?

### 2.7.4 Single-Page-Application - Junior

Programmiermethoden für Webanwendungen und asynchrones JavaScript und XML wurden entwickelt, um die SSR-Probleme zu lösen, was schließlich zur Single Page Application führte. SPAs sind Technologie, sie sind in der Browser-Version verfügbar. Wenn die Anwendungen dynamische Daten benötigen, senden sie Anfragen an die Web-API. Die Web-API ruft die Daten aus der Datenbank ab und sendet sie im JSON-Format zurück. Die Website wird auf der Seite im Browser gerendert. Es gibt kein Neuladen der gesamten Seite mehr. Die folgenden Seiten werden derzeit aktualisiert, sind jedoch nicht gültig, was eine reibungslose Benutzererfahrung bietet, die eher der einer auf einem Computer installierten Desktop-Anwendung entspricht. (vgl. Sun 2019: 1).

Der Aufbau einer SPA basiert in Frontend basiert nur auf die Verwendung von HTML in Kombination mit JavaScript und in Backend auf die Benutzung von anderen Programmiersprachen wie PHP, Java mit Tomcat, Bash usw.

Wie wurde unsere SPA entworfen?

Für den Entwurf einer SPA benötigt man die oben genannten Elemente. Gerade verfügen wir schon über eine leere HTML-Seite, die die Basis unserer Anwendung darstellt. Doch ist die Seite wie gesagt noch leer. Daher müssen Elemente hinzugefügt werden. Das erste Element, das hierbei hinzugefügt wurde ist die Navigationsleiste. Die wurde mit JavaScript hinzugefügt ohne die Seite noch einmal zu laden, dafür wurde eine JavaScript-Funktion verwendet.



---

### 2.7.5 JavaScript-Funktionen - Junior

Eine Funktion in JavaScript dient dazu, eine vordefinierte Abfolge von Befehlen außerhalb vom Hauptprogramm abzuspeichern. Damit wird möglich, sie durch die Nennung des Funktionsnamens an einer anderen Stelle aufzurufen. Deswegen ist es erstmal erforderlich der Funktion einen Namen zuzuweisen. Die Auswahl ist dabei frei. Es müssen allerdings die gleichen regeln wie bei den Variablennamen zu beachten. Vor dem Namen der Funktion muss das Schlüsselwort "function" geschrieben werden. Dieser weist deutlich darauf hin, dass es hierbei um eine Funktion geht. (vgl. Fuchs 2019a: 134)

Nach dem Funktionsnamen folgt eine runde Klammer, in der auch ein Parameter übergeben werden kann. Bei uns im Projekt fanden wir interessant die Funktion `init` vorzustellen, weil die den Aufruf unserer Hauptfunktion `navbar` möglich macht:

```
356
357  function init(event){
358      :      navbar(event)
359  }
360  window.onload = init;
```

Die Funktion `init` wird nur benutzt um die Hauptfunktion `navbar` aufzurufen. Die Funktion wird selber erst aufgerufen, wenn die Seite komplett geladen wurde und alle Elemente aus dem DOM vorhanden sind.

DOM steht für Document Object Model und ist eine Programmierschnittstelle (API) für HTML- und XML-Dokumente. Das DOM repräsentiert die gesamte Struktur eines Dokuments wie beispielsweise eine Webseite in einem baumartigen Objektmodell, das in JavaScript oder anderen Programmiersprachen manipuliert werden. Er stellt die hierarchische Struktur eines Dokuments dar. Jedes Element auf einer Webseite, wie Überschriften, Absätzen, Bilder, Links, `div`-Tags usw. wird im DOM als Objekt dargestellt.

Eine Funktion zu schreiben ist eine relativ einfache Sache je nach dem was abgezielt wird. Doch muss man schon wissen, wo die Funktion geschrieben werden kann. Denn wenn die nicht an die richtige Stelle geschrieben wird, kann die auch nicht funktionieren.

---

Eine Funktion kann man entweder in dem script-Tag (<script></script>) direkt im HTML-Code schreiben oder in einer separaten Datei mit der Endung .js beispielsweise script.js wie bei uns. Da muss man aber auf eine korrekte Einbindung der Datei achten. Dafür kann man eine Quelle (src) im script-Tag eingeben:

```
<script type="text/javascript" src="script.js"></script>
```

### 2.7.6 Funktionen mit Übergabewerten - Junior

Manchmal ist bei der Benutzung von Funktionen, dass diese einige Werte aus dem Hauptprogramm für ihre Berechnungen verwendet. Das war bei uns im Projekt der Fall, weil wir die Daten von mehreren Formularen auswerten und die auch korrekt übergeben wollten. Bei den ersten Versuchen ist es uns aufgefallen, dass die Formulare nicht eindeutig vom Programm unterschieden werden. Und wenn wir eine Aktion auf ein Formular geführt haben, hat sich dies immer nur die erste Zeile vom Formular beeinflusst, egal wo geklickt wurde. Daher haben wir uns dann entschieden folgende Technik zu benutzen. Also werden die Formulare per PHP generiert werden, wird auch eine eindeutige ID geschrieben, die bei der Funktionsaufruf übergeben wird, sodass jedes Formular vom Programm klar zu unterschieden ist.

```
85 //Konten loeschen
86 function loeschen(userID){
87     let formID = "theform" + userID
88     const form = document.getElementById(formID)
89     let formData = new FormData(form)
90     form.onsubmit = (event)=>{
91         event.preventDefault();
92     }
93     let req13 = new XMLHttpRequest
94     req13.open('POST', 'konto_loeschen.php', true);
95     req13.onload = () =>{
96         if(req13.readyState == 4 && req13.status == 200){
97             document.getElementById('outpout').innerHTML = req13.responseText
98             setTimeout(kontensehen,5000)
99         }
100     }
101     req13.send(formData)
102 }
103
```

Anmerkung: Und auch wenn die Funktion keinen Parameter übergeben bekommt, kriegt die trotzdem einen unsichtbaren Parameter, obwohl sie gar keinen erwartet. Dieser Parameter heißt "this". Dieses this-Schlüsselwort bezieht sich normalerweise auf das Objekt, auf dem die Funktion aufgerufen wird.

---

### 2.7.7 Zugriff auf die Inhalte des DOM-Baums - Junior

Bei der Benutzung von JavaScript kommt es stets dazu, dass man auf bestimmte Elemente zugreifen will, aber bevor man es tun, ist es wichtig deren Struktur zu verstehen. Diese Struktur wird wie bereits oben erwähnt als DOM-Baum bezeichnet. Die lässt sich halt dadurch erklären, dass die HTML-Seite eine baumartige Struktur mit zahlreichen Knotenpunkten aufweist, die einem Stammbaum ähnlichsieht.

Das Ursprungselement ist dabei das document-Element. Dieses beinhaltet alle weiteren Bestandteile der Seite. Und es ist davon das Objekt `document.Element` abgeleitet. In diesem Objekt werden alle innerhalb des `<html>`-Tags stehende Bereiche enthalten. In diesen befinden sich beispielsweise das `body`- und das `head`-Element. Der Zugriff auf diese Elemente geschieht über die Befehle `document.body` und `document.head`. (vgl. Fuchs 2019a: 242 ff.)

Es kann allerdings auch gezielt auf bestimmte Inhalte zugegriffen werden. Bei uns im Projekt geschieht der Zugriff auf die Inhalte vom DOM-Baum hauptsächlich mit `id` über den Befehl `"document.getElementById"`. Dies benutzen wir für viele Dinge und vor allem für die Auswertung von Formularen und das Abschicken von Daten. Unsere Seitenstruktur war aus unserer Sicht sehr umfangreich und hatte auch mehreren Code-Zeilen, daher haben wir uns dafür entschieden. Im oben abgebildeten Code verwenden wir noch dazu ein `".innerHTML"`, damit der Inhalt vom Element auf das zugegriffen wird auch entsprechend geändert wird.

### 2.7.8 Events in JavaScript - Junior

Wenn man die oben beigefügte JavaScript-Funktion betrachtet, fällt sofort auf, dass dort ein Parameter übergeben wird. Dieser lautet `event` und bezieht sich tatsächlich auf ein Event. JavaScript ist eigentlich eine eventbasierte Programmiersprache.

Events sind in JavaScript eines der wichtigsten Hilfsmittel für die Interaktion zwischen dem Webbrowser und dem JavaScript-Programm. Jedes Mal, wenn der Benutzer eine bestimmte Aktion durchführt, löst der Browser ein entsprechendes Event aus. Das kommt beispielsweise, wenn der Besucher einer Webseite ein Element anklickt, wenn er mit der Maus darüberfährt, wenn er ein Fenster schließt oder wenn er eine Eingabe mit der Tastatur macht.

---

Beim Aufruf einer Webseite durch den Benutzer, werden wie bereits gesagt verschiedene Events aufgelöst. Diese Ereignisse sind zum Beispiel: click, contextmenu, mouseover, mouseout, musedown, mousedown, mouseup, mousemove, dbclick, submit, focus, keydown, keyup, DOMContentLoaded, transitioned (vgl. Fuchs 2019a: 202).

Aber bei uns im Projekt haben wir uns nur auf bestimmte Ereignisse fokussiert wie das Klicken und Abschicken (auf Englisch: click, submit).

### **2.7.9 AJAX-Aufruf - Junior**

Für die Erstellung unserer Anwendung wollten wir eine Seite bauen, die den Server nicht viel beansprucht, die auch für den Benutzer leicht verwenden wird und zusätzlich noch längere Wartezeiten für das Neuladen der Seite vermeiden. Dies ist noch ein Grund, warum wir uns für eine SPA entschieden haben.

Für eine SPA ist die Benutzung von JavaScript erforderlich. Und dabei wird die Hauptseite (index.html) verwendet um das Ergebnis der verschiedenen Nutzerinteraktion und Abfragen anzuzeigen, doch zu sagen, dass es JavaScript macht reicht nicht aus. Jeder stellt sich dann die Frage wie das geschieht? Wie kommuniziert JavaScript mit anderen PHP-Skripten?

Die Antwort ist mit AJAX-Aufrufen. AJAX sorgt für eine erhebliche Erleichterung. Diese Technik macht es möglich, eine präzise Anfrage an den Server zu senden, um genau die benötigten Inhalte anzufordern. Diese kann man daraufhin in die bestehende Seite einfügen. Bei diesem Vorgang werden jedoch nur die Bausteine ausgetauscht. Der Rest der Seite bleibt erhalten, was die Ladezeiten erheblich reduziert.

Eine praktische Verwendung von AJAX finden wir in unserer Navigationsleiste, deren Inhalt je nach Status und Rolle der Sitzung variiert.

```

222 //Verwaltung der Navbar und main Inhalt. Callback von anderen Funktionen
223 function navbar(event){
224     let req1 = new XMLHttpRequest()
225     req1.onreadystatechange = function xhr(){
226         if(req1.readyState == 4 && req1.status == 200){
227
228             document.getElementById('navbar').innerHTML = req1.responseText
229
230             if(document.getElementById('home') != null){
231                 document.getElementById('home').addEventListener('click', home)
232             }
233             if(document.getElementById('youshallnotpass') != null){
234                 document.getElementById('youshallnotpass').addEventListener('click', youshallnotpass)
235             }
236             if(document.getElementById('kurse') != null){
237                 document.getElementById('kurse').addEventListener('click', allfitnesskurse)
238             }
239             if(document.getElementById('kurse2') != null){
240                 document.getElementById('kurse2').addEventListener('click', kurse2)
241             }
242             if(document.getElementById('login') != null){
243                 document.getElementById('login').addEventListener('click', loginformular)
244             }
245             if(document.getElementById('registrieren') != null){
246                 document.getElementById('registrieren').addEventListener('click', registrierungsformular)
247             }
248             if(document.getElementById('logout') != null){
249                 document.getElementById('logout').addEventListener('click', logout)
250             }
251             if(document.getElementById('benutzerkonten') != null){
252                 document.getElementById('benutzerkonten').addEventListener('click', kontensehen)
253             }
254             if(document.getElementById('katalog') != null){
255                 document.getElementById('katalog').addEventListener('click', katalog)
256             }
257             if(document.getElementById('registered') != null){
258                 document.getElementById('registered').addEventListener('click', anmeldungen)
259             }
260             if(document.getElementById('aendern') != null){
261                 document.getElementById('aendern').addEventListener('click', status)
262             }
263         }
264     }
265     req1.open('GET', 'navbar.php?t='+Date.now(), true)
266     req1.send()
267 }

```

Die Funktion „navbar“ wird von der Funktion initäufgerufen, die selbst von dem "window.onloadEreignis.

Für den AJAX-Aufruf war es erstmal notwendig ein XMLHttpRequest-Objekt zu erzeugen. Dabei handelt es sich um ein vordefiniertes Objekt, das in allen modernen Browser vorhanden ist:

```
let req1 = new XMLHttpRequest()
```

Danach ist folgende Codezeile notwendig:

```
req1.onreadystatechange = function xhr(){..}
```

Anmerkung: Wir hätten hier auch eine Variante verwenden können. Also eine sogenannte anonymous-Funktion benutzen können. Darüber wird nachher berichtet.

---

Das Attribut `onreadystatechange` des `XMLHttpRequest`-Objekts enthält eine Funktion, die dann ausgeführt wird, wenn sich der Status der aktuellen Anfrage ändert. Das ist unter anderem der Fall, wenn das Programm die angeforderten Inhalte vom Server übermittelt bekommt. Doch auch wenn man die Inhalte erneut anfordert, ändert sich der entsprechende Status, sodass die Funktion aufgerufen wird. Mit dem obigen Befehl geben wir an, dass bei jeder Statusänderung die Funktion `xhr()` aufgerufen werden soll.

Gibt es auch einige Statusänderungen, auf die wir gar nicht reagieren wollen. Eine Aktion ist in unserem Fall nur dann notwendig, wenn die Daten korrekt übermittelt wurden. Im genannten Fall hat dann das `readyState`-Attribut den Wert 4. Aus diesem Grund haben wir den kompletten Inhalt der Funktion innerhalb einer `if`-Abfrage geschrieben, die überprüft, ob das `readyState`-Attribut diesen Wert aufweist.

Zudem verfügt das `XMLHttpRequest`-Objekt über ein Attribut mit der Bezeichnung `status`. Dieses hilft uns dabei zu wissen, ob die Übermittlung erfolgreich war. Dafür steht dann der Wert 200. Weil es nur bei korrekter Übermittlung sinnvoll ist, die neuen Inhalte anzuzeigen, soll diese Bedingung ebenfalls in die `if`-Abfrage aufgenommen werden. Es ergibt sich dann Folgendes:

```
if (req1.readyState == 4 && req1.status == 200)
```

Im nächsten Schritt ist es notwendig, die `open()`-Methode auf das `XMLHttpRequest`-Objekt anzuwenden:

```
req1.open('GET', 'navbar.php?t='+Date.now(), true)
```

Genau sowie beim Übermitteln von Formularen muss man auch bei der Verwendung der `open()`-Methode mit den Begriffen `GET` und `POST` die Art der Übermittlung vorgeben. Das spielt aber nur dann eine Rolle, wenn man mit `AJAX` auch Daten an den Server übermitteln will. Danach steht die Datei bzw. die Adresse der Datei, die erreicht werden soll und zum Schluss steht der Begriff `true`. Ein riesiger Vorteil von `AJAX` besteht darin, dass diese Technik asynchron arbeitet. Allerdings kann man auch für einen synchronen Ablauf sorgen. Dies ist machbar, wenn man hier den Wert `false` eingibt. Das Programm wartet dann bis die Daten geladen sind und ist daher während dieser Zeit nicht für weitere Eingaben verfügbar (vgl. Fuchs 2019a: 363).

---

Am Ende muss noch die Methode `send()` aufgerufen werden. Mit dieser Methode wird dafür gesorgt, dass die Seite die entsprechende Anfrage an den Server absendet und auf diese Weise die Daten anfordert.

Nachdem die Anfrage gesendet wurde und vom entsprechenden PHP-Skript bearbeitet wurde, ist es auch möglich, die Antwort vom Server zu benutzen und als Text anzuzeigen. Dafür wird das `responseText`-Attribut verwendet. Es wird auf das Element im DOM mit der id „navbar“ zugegriffen und die Antwort wird angehängt:

```
Document.getElementById('navbar').innerHTML = req1.responseText
```

### 2.7.10 Die if-Abfrage und Callback - Junior

Dass die Antwort vorhanden und für den Benutzer sichtbar ist macht nicht alles. Es garantiert noch nicht die Ausführung anderer Funktionen und auch zum richtigen Zeitpunkt nach dem richtigen Ereignis. Um dies zu beheben, haben wir viele unterschiedliche if-Abfragen verwendet. Es wird erstmal geprüft ob, das richtige Element im DOM vorhanden ist, damit die richtige Funktion nach dem damit verbundenen Ereignis durch einen Callback aufgerufen wird.

### 2.7.11 Die if-Abfrage - Junior

Eine if-Abfrage wird generell verwendet, um eine Bedingung aufzustellen, sodass die korrespondierenden Befehle nur dann ausgeführt werden, wenn die Bedingung erfüllt ist. Die Syntax einer if-Abfrage ist sehr einfach und funktioniert ein bisschen wie im alltäglichen Leben. Also braucht man eine Bedingung und eine Aktion, die danach ausgeführt wird. Es kann auch eine andere Aktion im else-Teil hinzugefügt werden, damit trotzdem etwas auch passiert, wenn die Bedingung nicht erfüllt ist.

```
312     if (document.getElementById('theform8') !== null) {
313         document.getElementById('theform8').addEventListener('submit', function (event) {
314             event.preventDefault();
315             aendern()
316         });
317     } else {
318         console.log('nicht gefunden')
319     }
```

---

### 2.7.12 Callback - Junior

Beim Programmieren mit JavaScript kommt es sehr häufig vor, dass man eine Funktion mithilfe einer anderen Funktion aufrufen will. Und die eine Funktion dann als Argument für die zweite Funktion gibt und die noch innerhalb derselben Funktion aufruft. Dies wird als Callback bezeichnet. Ein gutes Beispiel bei uns für die Verwendung von Callbacks lässt sich in der Funktion `navbar()` ablesen, da wo wir der Funktion `addEventListener()` Funktionen als Argumente übergeben. Ein anderes Beispiel wäre die Funktion `loginformular()` :

```
105 // Aufruf vom Loginformular, das vor dem Submit-Event überprüft wird
106 function loginformular(){
107     document.getElementById('main').innerHTML = "<div id='outpout'></div><br><br><div id='main'><h2
108     document.getElementById('myform').addEventListener('submit', function(event){
109         event.preventDefault()
110         anmelden()
111     });
112 }
113
```

### 2.7.13 PHP mit CSS, HTML und JavaScript - Refat

In der Vergangenheit wurde bereits darüber geschrieben, was PHP ist und in welchen Bereichen PHP zum Einsatz kommt. Im Folgenden wird der genauere aus technischer Sicht betrachtete Einsatz von PHP behandelt.

Um PHP, HTML und CSS in einer Webanwendung zu verbinden, muss man zuerst verstehen, wie diese Technologien zusammenarbeiten. Hierbei wird eine HTML-Seite als Grundstruktur für die Webanwendung verwendet. Hinzufügend erstellt man eine CSS-Datei, um das Layout und das Erscheinungsbild der Webseiten zu gestalten. Und dann verknüpft man diese CSS-Datei im `<head>`-Bereich der HTML-Seite. Man kann PHP-Code direkt in HTML einbetten, indem man PHP-Tags verwendet. Der PHP-Code wird auf dem Server ausgeführt und kann dynamische Inhalte generieren, die in die HTML-Ausgabe eingefügt werden.

Wir wissen bereits, dass dynamische Webseiten mithilfe von PHP entwickelt werden können. Obwohl dies auf den ersten Blick widersprüchlich erscheinen mag, ist es wichtig zu beachten, dass praktisch alle mit PHP erstellten Webseiten auch HTML-Code verwenden.



---

Die Verbindung besteht darin, dass PHP-Code oft zur Generierung von HTML führt. Wenn das Programm ausgeführt wird und der Server über die erforderlichen Informationen verfügt, wird eine standardmäßige HTML-Seite generiert. Diese wird vom Webbrowser des Benutzers empfangen und interpretiert, als handle es sich um eine einzelne statische Seite. Der dynamische Teil erfolgt während der Ausführung auf dem Server, wenn einzigartige Seiten erstellt werden.

Der folgende Schritt zeigt, wie wichtig es für PHP-Programmierer ist, viel über HTML zu wissen. Es versteht sich von selbst, dass Sie, wenn Sie Software erstellen möchten, deren Ergebnis HTML ist, wissen müssen, wie es aufgebaut ist und wie es funktioniert. Außerdem ist es üblich, nur die PHP-Programmiersprache zu verwenden, um bestimmte Teile einer Website zu erstellen. Dies bedeutet, dass zuerst eine normale HTML-Seite erstellt wird. PHP-Codes werden nur hinzugefügt, um die erforderlichen Daten einzugeben, wenn eine Änderung des Inhalts erforderlich ist.

#### **2.7.14 HTML-Code mit PHP ausgeben - Refat**

Zunächst einmal, wird gezeigt, wie das erreicht werden kann. PHP kann verwendet werden, um HTML zu erstellen. Theoretisch kann diese Arbeit leicht erledigt werden. Hierfür wird die StandardDruckfunktion verwendet. Es wird jedoch kein einfacher Text verwendet. Tags müssen mit HTML angegeben werden. Zum Beispiel ist es ausreichend, die entsprechenden Überschriften zu markieren. `<h1>`-Tags sollten in der Druckfunktion verwendet werden.

Es versteht sich von selbst, dass Anführungszeichen um eine Zeichenkette herum verwendet werden sollten. Dies ist entscheidend, da PHP spezialisierte Funktionen aufgrund des vorhandenseins von spitzen Klammern bietet. Wenn sie nicht ordnungsgemäß als Zeichenketten erkannt werden, kann dies zu Syntaxproblemen führen.

---

### 2.7.15 Init-Fehlermeldung - Refat

Eine **Init-Fehlermeldung** in PHP bezieht sich auf Probleme, die beim ersten Laden der Anwendung auftreten. Diese Signale sind normalerweise ungenau und können auf verschiedene Ursachen zurückzuführen sein.

Was die Aktivierung der Fehlerprotokollierung angeht, so muss sichergestellt werden, dass die Fehlerprotokollierung in der PHP-Anwendung aktiviert ist. Dies kann man in der PHP-Konfigurationsdatei `php.ini` oder direkt im PHP-Code mit `error_reporting(E_ALL)` festlegen. Dadurch werden detailliertere Fehlermeldungen angezeigt, die bei der Fehlerdiagnose helfen können.

Was die Identifizierung der genauen Fehlermeldung angeht, so wird hierfür erfordert, dass man das Fehlerprotokoll des Browsers oder das Systemprotokoll auf präzisere Fehlermeldungen überprüft. Die Datei und die Zeilennummer des Fehlers sind in diesen Meldungen normalerweise zu finden.

Für den Verifizierungscode sollte man, wenn in PHP eine **Init-Fehlermeldung** angezeigt wird, einen Blick auf den Code werfen. Hierfür muss nach Syntaxfehlern, fehlenden Dateien und einer falschen Konfiguration gesucht werden. Die Protokolle des Web-Servers (Wie Apache- oder Nginx-Protokolle) können ebenfalls nützliche Informationen über das Problem enthalten. Hierfür muss man daher sicherstellen, dass man diese überprüft.

Zusätzlich muss man sicherstellen, dass die Anwendung reibungslos auf der von sich verwendeten PHP-Version läuft. Inkompatibilitäten zwischen älteren Programmen und neueren PHP-Versionen können zu Init-Problemen führen. Ebenso sollte man prüfen und sicherstellen, ob ausreichend Speicherplatz auf dem Server vorhanden ist. Man muss sicherstellen, dass der Server die Systemanforderungen (CPU, RAM) für die reibungslose Ausführung ihrer PHP-Anwendung erfüllt. Man muss sicherstellen, dass die Konfigurationseinstellungen der Anwendung und alle ihre Abhängigkeiten korrekt sind.

---

### 2.7.16 PHP: Assoziative Arrays - Refat

Zum Speichern und Abrufen von Informationen mit benannten Schlüsseln sind assoziative Arrays in PHP eine wichtige Datenstruktur. Im Gegensatz zu numerischen Arrays, die auf aufeinanderfolgende Zahlen angewiesen sind, können assoziative Arrays jeden Wert als Schlüssel verwenden. Werte aus Online-Formularen werden ebenfalls über assoziative Arrays übertragen. GET-Argumente sind ein Schlüssel-Wert-Paar, das am Ende der URL nach einem Fragezeichen angehängt wird und über die Variable `$_GET` abgerufen werden kann.

### 2.7.17 PHP: Wozu werden assoziative Arrays verwendet - Refat

Es gibt viele Anwendungen für assoziative Arrays in der Programmierung, aber ihre Hauptverwendung besteht darin, große Datenmengen zu organisieren und zugänglich zu machen. Hier sind einige typische Anwendungen von assoziativen Arrays:

Zu aller Erst werden sie verwendet, um strukturierte Informationen wie Kontaktdaten, Konfigurationseinstellungen und mehr zu speichern und abzurufen. Zweitens werden sie in Webanwendungen oft verwendet, um die Daten, die aus HTML-Formularen gesendet werden, zu speichern und zu verarbeiten. Drittens kommen sie bei der Verarbeitung von Abfrageergebnissen aus Datenbanken zum Einsatz, um Datenzeilen mit Spaltennamen als Schlüssel zu speichern. Viertens werden sie häufig in Datenübertragungsformaten wie JSON verwendet, um strukturierte Daten zwischen verschiedenen Systemen auszutauschen. Fünftens werden sie in Webanwendungen verwendet, um Sitzungsinformationen für Benutzer zu speichern, einschließlich Authentifizierungsinformationen, Benutzereinstellungen und temporäre Daten. Sechstens können sie verwendet werden, um Benutzereinstellungen und Profile in Anwendungen zu verwalten. Siebtens eignen sie sich gut, um Daten in Kategorien oder Gruppen zu organisieren, wobei die Schlüssel die Kategorien repräsentieren und die Werte die zugehörigen Daten enthalten. Achters bieten sie eine übersichtliche Möglichkeit, Daten zu organisieren, indem sie klare Zuordnung zwischen Schlüsseln und Werte ermöglichen.

Assoziative Arrays sind also eine flexible und vielseitige Datenstruktur, die in vielen Programmiersprachen verwendet wird, um strukturierte Daten zu verwalten und auf sie zuzugreifen. Sie erleichtern die Organisation und Verarbeitung von Informationen in einer klaren und intuitiven Form.

---

### 2.7.18 PHP: Sessions und ihre Anwendung - Refat

Sessions sind eine Funktion von PHP, die es Benutzern ermöglicht, Daten über mehrere Seitenaufrufe hinweg auf dem Server zu speichern und abzurufen. Sessions ermöglichen es, Benutzerdaten sicher über verschiedene Seiten einer Website zu übertragen, ohne sie in URLs oder Formularen offenzulegen. Sessions werden aktiviert, wenn sich ein Benutzer auf einer Website anmeldet.

Die Verwendung von Sessions in PHP bietet eine Reihe von Vorteilen, die die Entwicklung und Nutzung von Webanwendungen erleichtern. Hier sind einige der besten Aspekte von PHP-Sessions:

Zu aller Erst bleiben Daten zwischen Seitenansichten gleich:

Sessions ermöglichen es dem Benutzer also, Daten zu speichern, die zwischen Seitenansichten gleich bleiben. Dies ermöglicht es auch, die Sitzung und Informationen eines Benutzers zu verfolgen, während er oder sie sich auf seiner/ihrer Website bewegt.

Des Weiteren ermöglichen es Sessions, Daten sicher auf dem Server zu speichern, anstatt sie in URLs oder Formularen zu übertragen. Dies verringert die Wahrscheinlichkeit, dass private Informationen nach Außen gelangen oder Sicherheitsprobleme verursachen, da sie nicht in den Browserprotokollen oder URL-Parametern sichtbar sind.

Noch sind Sessions ein Schlüsselkonzept für die Benutzeridentifikation. Sie ermöglichen es, Benutzer nach dem Einloggen zu identifizieren und ihren Zustand über verschiedene Seitenansichten hinweg zu verfolgen, ohne den Benutzer erneut anmelden zu lassen.

Außerdem helfen Sessiions E-Commerce-Anwendungen, den Warenkorb eines Benutzers über mehrere Bildschirme hinweg zu verfolgen. Dies ermöglicht es den Menschen, auf einer Website einzukaufen, Artikel hinzuzufügen, zu entfernen und zu überprüfen.

Ferner ermöglichen Sessions die Anpassung von Inhalten und die Art und Weise, wie Benutzern damit interagieren. Sie können Einstellungen oder Auswahlmöglichkeiten speichern, die speziell für jeden Benutzer sind, und die Benutzeroberfläche basierend auf den Sitzungsdaten ändern.

---

Noch zusätzlich ermöglichen es Sessions, das Verhalten von Benutzern zu beobachten und Sicherheitskontrollen einzurichten. Man kann beispielsweise verfolgen, wie lange eine Person nichts getan hat, und sie automatisch ausloggen, wenn die Zeit abgelaufen ist.

Zusätzlich sind Sessions hilfreich bei der Entwicklung von Webanwendungen, da sie den Entwicklungsprozess erleichtern. Sitzungsvariablen können verwendet werden, um Daten und Problemprotokolle zu speichern, die dann zur Analyse verwendet werden können.

Außerdem können Sessions auf mehrere Computer dupliziert oder an andere Speicheroptionen wie Datenbanken oder Redis gesendet werden, um große Apps schneller und zuverlässiger auszuführen.

Zuletzt ermöglichen Sessions es, zu verfolgen, was Benutzer tun, und können für die Analyse verwendet werden. Sie helfen dabei herauszufinden, wie Menschen ihre Webseite nutzen.

Insgesamt sind Sessions ein leistungsstarkes Werkzeug in der Webentwicklung, das es ermöglicht, Benutzerdaten auf sichere und effiziente Weise über verschiedene Seitenansichten hinweg zu verwalten und zu nutzen. Die sind eines der wichtigsten Konzepte für die Umsetzung von Benutzerinteraktionen, Identifikation und Personalisierung in Webanwendungen.

#### **2.7.19 Was sind Schleifen überhaupt? - Refat**

Um die Funktion einer Schleife zu erklären, soll ein ganz einfaches – wenn auch nicht unbedingt sinnvolles – Beispiel dienen. Hierbei soll das Programm lediglich zehn Mal hintereinander das Wort Hallo auf dem Bildschirm ausgeben. Eine derartige Aufgabe kommt bei der praktischen Arbeit sicherlich nur sehr selten vor, sie verdeutlicht jedoch anschaulich, wie eine Schleife funktioniert (vgl. Bonacina 2019: 75).

In PHP gibt es zwei Hauptarten von Schleifen:

1-For-Schleife: Eine For-Schleife wird verwendet, wenn man es genau wissen will, wie oft Sie einen Codeblock ausführen möchte.

2-While-Schleife: Eine While-Schleife wird verwendet, wenn man nicht im Voraus weiß, wie oft der Codeblock ausgeführt werden soll.

---

Schleifen sind hilfreich, um Aufgaben zu wiederholen, die häufig durchgeführt werden müssen. Sie können auch verwendet werden, um Daten zu verarbeiten oder Berechnungen durchzuführen. Jetzt nennen wir Beispiele für die Verwendung von Schleifen in unserem Projekt:

```
48
49 // Annahme: $query enthält das Ergebnis der Datenbankabfrage
50 while ($row = mysqli_fetch_assoc($query)) {
51     $trainer = $row['Fitnesstrainer'];
52 }
```

Die While-Schleife wird verwendet, um die Daten aus der Datenbank abzurufen. Hier wird die Schleife dafür verwendet, um Daten aus der Variable \$query zu lesen und in dem Array \$trainerCourses zu speichern.

```
69 foreach ($courses as $course) {
70     echo "<td>";
71     echo "Trainer: " . $course['Fitnesstrainer'] . "<br>";
72     echo "Kurs: " . $course['kursname'] . "<br> Status: " . $course['Status'] . "<br>";
73     echo "<form id='myform3'>";
74     echo "<input type='hidden' name=kurs_id value=" . $course['kurs_ID'] . ">";
75     echo "<input type='hidden' name=kunden_id value=" . $_SESSION['id'] . ">";
76     echo "<button id='teilnehmen' type='submit'>jetzt teilnehmen</button>";
77     echo "</form>";
78     echo "</td>";
79 }
```

Die zweite Foreach-Schleife wird verwendet, um die Kurse zu jedem Trainer anzuzeigen. Die Schleife beginnt mit dem Array \$courses, das die Kurse für den aktuellen Trainer enthält. Die Schleife wird dann solange ausgeführt, bis das Array leer ist.

### 2.7.20 PHP: Funktionen - Refat

Rückgabewerte der Funktionen: Für den Rückgabewert können unterschiedliche Variablentypen zum Einsatz kommen. Es ist auf diese Weise daher nicht nur möglich, Zahlen an das Hauptprogramm zu übermitteln, sondern auch Zeichenketten und Wahrheitswerte. Um den Rückgabewert im Hauptprogramm aufzunehmen, ist es möglich, einer Variablen den Wert der entsprechenden Funktion zuzuweisen. Das geschieht genau wie bei der Zuweisung gewöhnlicher Werte durch ein Gleichheitszeichen (vgl. Bonacina 2019: 96).

---

```
<?php
function verdopplung ($wert)
{
    return ($wert*2);
}
$ergebnis = verdopplung(5);
print $ergebnis;
?>
```

### 2.7.21 PHP: eine Funktion in das PHP-Programm einbinden - Refat

Um eine Funktion in einem PHP-Programm zu verwenden, ist es erforderlich, sie in das Hauptprogramm einzubinden. Hierfür gibt es verschiedene Möglichkeiten. Hierbei wurde die Funktion dem eigentlichen Programm vorangestellt. Der Abruf erfolgte ganz einfach über den Funktionsnamen. Diese Vorgehensweise ist zwar sehr einfach, sie führt jedoch dazu, dass einer der wesentlichen Vorteile der Verwendung von Funktionen verloren geht: die übersichtliche Gestaltung des Programms. Wenn bei einer umfangreichen Software mehr als hundert Funktionen zum Einsatz kommen, führt das zu einem sehr langen Programmcode, bei dem es schwierig ist, die einzelnen Elemente zu erkennen.

Aus diesem Grund kommt diese Methode nur bei sehr einfachen Programmen zum Einsatz. Bei umfangreicheren Anwendungen ist es üblich, die Funktionen in eigene Dateien auszulagern. Das bedeutet, dass hierfür eine neue Datei erstellt werden muss. Um darauf zuzugreifen, ist es notwendig, die entsprechenden Dateien vor dem Aufruf der Funktion in das Hauptprogramm einzubinden. Wie das funktioniert, soll an der Funktion `function verdopplung()` beschrieben werden, die im vorherigen Abschnitt erstellt wurde. Um diese auszulagern, ist es erforderlich, zunächst eine neue Datei zu erstellen. Um die Zuordnung zu erleichtern, sollte der Dateiname dabei dem Funktionsnamen entsprechen. Auch hierbei kommt die Endung `.php` zum Einsatz. In dieser Datei steht lediglich der Bereich, in dem die Funktion definiert wurde – wie immer in den entsprechenden PHP-Tags ( vgl. Bonacina 2019: 98):

---

```
<?php
function verdopplung ($wert)
{
    return ($wert*2);
}
?>
```

### 2.7.22 Zugriff auf MySQL-Datenbank durch PHP - Hossein

Wie in der Vergangenheit bereits schon erwähnt, dient PHP am besten dazu, um im Backend-Bereich eingesetzt zu werden, um damit zum Beispiel von einer Datenbank, Daten hervorzurufen, und mit diesen Daten im Nachhinein zu arbeiten. Möchte man dies mit MySQL Datenbank machen, so muss der Entwickler zuerst mit den grundlegenden SQL-Befehlen vertraut sein. Hierzu ist es wichtig zu wissen, wie man mit SQL, Datenbanken erstellt, Tabellen anlegt und die darin enthaltenen Daten manipuliert.

Im Nachhinein geht es darum, wie man durch PHP und den SQL-Befehlen, eine Verbindung zur SQL-Datenbank aufbaut.

Möchte man sich genauer gesagt mit den SQL-Befehlen beschäftigen, so ergeben sich aus den SQL-Befehlen, drei Teilbereiche, die sich alle in ihrer Funktionalität unterscheiden. Einerseits spricht man von der sogenannten "Data Definition Language (DDL)", andererseits von der sogenannten "Data Control Language (DCL)", und Ferner von der sogenannten "Date Manipulation Language (DML)".

Aus dem Bereich von DDL, der für die Erstellung der Strukturen einer Datenbank gedacht ist, gibt es den wichtigsten Befehl als "CREATE", der als aller erstes zum Einsatz kommt und zugleich für die Erstellung einer Datenbank gedacht ist. Andererseits lassen sich damit Tabellen mit individuell gestalteten Strukturen aufbauen. Als Gegenzug zu diesem Befehl, gibt es den Befehl "DROP", dessen Zweck die Löschung von Datenbanken, Tabellen und weiteren Dingen ist. Für das Ändern der Strukturen einer Datenbank gibt es den Befehl "ALTER".

Aus dem Bereich von DCL, der für das Ausgeben und Ziehen von Zugriffsrechten für Daten da ist, gibt es den Befehl "GRANT" mit dem man einen Nutzer Zugriffsrechte erteilen kann. Im Gegenzug dessen gibt es den Befehl "ERVOKE", mit dem man einem Nutzer die Zugriffsrechte entziehen kann.



---

Aus dem Bereich von DML, der für die Manipulation der Daten wie das Eingeben, Löschen, sowie Ändern der Daten zuständig ist, gibt es den Befehl “SELECT“, der den Zweck erfüllt, Daten aus einer Datenbank abzurufen. Ferner gibt es die Befehle “INSERT“, “UPDATE“, sowie “DELETE“, mit dem man Daten in einer Datenbank Einfügen, Verändern, sowie Löschen kann (vgl. Bonacina 2019: 137-138).

Kommt es mal in der Praxis dazu, eine Verbindung mit der Datenbank aufzubauen, so tut man dies in den meisten Fällen mit einem PHP-Programm, mit dessen Hilfe die Automatisierung der Verbindung zur Datenbank und der Manipulation, das Abrufen, und etc. der Daten, gewährleistet wird.

Möchte man über ein PHP-Programm eine Verbindung zu einer Datenbank aufbauen, so muss sich der Benutzer als Erstes authentifizieren. Dies tut er durch einen Benutzernamen und einem Passwort. Für die eigentliche Erstellung der Verbindung gibt es beispielweise die Methode mit den mysqli-Funktionen, mit denen man neben der Verbindung zur Datenbank, verschiedene Befehle eingeben kann.

Eine Verbindung zur Datenbank wird mit dem Befehl `mysqli_connect()` erstellt, der im Voraus jedoch in eine Variable initialisiert werden muss. In den Klammern dessen schreibt man den Host, den Benutzernamen, das Passwort, sowie den Namen der Datenbank. Nach der Erstellung der Verbindung prüft man mit einer if-else-Verbindung, ob die Verbindung im Falle False aufgebaut worden ist, oder im Falle True nicht aufgebaut worden ist. Tritt der Fall False auf, so ist es sinnvoll eine Beispielsausgabe wie “Die Verbindung wurde aufgebaut” zu erzeugen. Andernfalls, also im Falle True, als Beispiel die Ausgabe “Keine Verbindung zur Datenbank möglich”, wobei zugleich eine Fehlermeldung erzeugt wird.

Nach dem Beenden des Programms ist es wichtig, die Verbindung zur Datenbank ebenfalls zu schließen. Dies tut man wiederum mit dem Befehl “`mysqli_close()`“.

---

Setzt man die erwähnte Anleitung durch, so ergibt sich dadurch das folgende Programm:

```
-----
1. <?php
2. $dbh = mysqli_connect("localhost",
3.                       "user1",
4.                       "abc",
5.                       "onlineshopDB");
6.
7. if (mysqli_connect_error()) {
8.     print "Keine_Verbindung_zur_Datenbank_möglich:_" .
9.     mysqli_connect_error();
10. }
11. else {
12.     print "Verbindung_erfolgreich_hergestellt.";
13. }
14. mysqli_close($dbh);
15. ?>
-----
```

(vgl. Bonacina 2019: 161-162).

Möchte man im Falle true, bei dem die Verbindung zur Datenbank erstellt wird, die Datenbank füllen, so muss hierfür zuerst eine Tabelle erzeugt werden. Mit den weiteren SQL-Befehlen lassen sich dementsprechend auch die Spalten und die individuell ausgewählten Datentypen pro Spalte, wie folgt erzeugen:

```
-----
1. CREATE TABLE os_bestellungen
2. (b_id integer unsigned not null auto_increment primary key,
3.  b_artikeldnummer integer unsigned not null,
4.  b_kundendnummer integer unsigned not null,
5.  b_preis decimal(7,2) unsigned not null,
6.  b_anzahl integer unsigned);
-----
```

---

(vgl. Bonacina 2019: 164).

Möchte man die Daten einer Tabelle, von einer Datenbank löschen, so tut man dies mit dem folgenden Befehl:

```
-----
1. $sql = DELETE FROM os_bestellungen
2.         WHERE p_id= 1 ;
-----
```

Möchte man hingegen die Daten einer Tabelle, von einer Datenbank ändern, so tut man dies mit dem folgenden Befehl:

```
-----
1. $sql = UPDATE os_bestellungen
2.         SET b_anzahl = b_anzahl + 1
3.         WHERE b_id = 1; ;
-----
```

(vgl. Bonacina 2019: 168).

Möchte man die Daten einer Tabelle, von einer Datenbank abrufen, so tut man dies mit dem folgenden Befehl:

```
-----
1. $sql = SELECT b_preis
2.         FROM os_bestellungen; ;
-----
```

(vgl. Bonacina 2019: 169).

Diese Befehle erstellen die Grundlagen, mit denen man eine Verbindung zur Datenbank aufbauen, mit den Daten umgehen, und diese mithilfe von PHP und weiteren Techniken weiterbearbeiten kann, je nachdem welches Ziel der Entwickler hat.

### 2.7.23 Redis: Definition - Refat

In einem vorherigen Kapitel wurde kurz angesprochen, was man unter Redis versteht. Hierbei sollen nun etwas genauer über den Sinn und Zweck, sowie dessen Funktionalitäten eingegangen werden. Ferner wird die Redis-Technologie als Solches in ihre fundamentalen Teile zergliedert, sowie dabei auf seine Anwendung reduziert.

Redis ist eine entfernte Speicherdatenbank mit hoher Geschwindigkeit, Replikation und einem neuartigen Datenformat, das verwendet werden kann,

---

um eine Plattform zur Lösung komplexer Probleme zu entwickeln. Redis' Fähigkeit, mit einer Vielzahl von Problemen umzugehen, resultiert direkt aus seiner Unterstützung für fünf verschiedene Datenstrukturen. Dadurch können Probleme ohne den begrifflichen Aufwand, den andere Datenbanken erfordern, gelöst werden.

Redis verfügt über zusätzliche Funktionen wie Replikation, Persistenz und clientseitiges Sharding, die es skalierbar machen, von einem einfachen Prototyping-Tool bis hin zu einer produktionsbereiten Datenbank, die Hunderte von Gigabyte an Daten und Millionen von Abfragen pro Sekunde verarbeiten kann.

Redis ist eine schnelle, nicht-relationale Datenbank, die fünf verschiedene Arten von Wertzuordnungen für einen bestimmten Schlüssel speichern kann. Redis ermöglicht das clientseitige Sharding, um die Schreibgeschwindigkeit zu erhöhen, und Wiederholung, um die Leseleistung zu steigern.

#### **2.7.24 Redis: Key-Value-Store und Message-Broker - Refat**

Im Bereich der Datenbankstrukturen handelt es sich bei einer Key-Value-Datenbank (auch als "Key-Value-Store" bezeichnet) um eine der einfachsten Datenbankarten. In diesen Datenbanken sind die Schlüssel immer Zeichenfolgen, während die Werte jeden Datentyp und jede Größe haben können. Die Einrichtung und Nutzung einer solchen Datenbank erfordert nur wenig Aufwand. Redis unterstützt sowohl die Speicherung von Key-Value-Paaren als auch die Nachrichtenvermittlung. Redis ermöglicht die Durchführung atomarer, verteilter Operationen. Dadurch können komplexe Aufgaben zuverlässig und sicher ausgeführt werden. Hier sind einige Beispiele:

Zuerst erfüllt Redis die Aufgabe eines atomaren Zählers, bei dem Inkremente und Dekremente gleichzeitig stattfinden. Der Zählerwert wird durch gleichzeitige Aktivitäten nicht willkürlich geändert. Zweitens unterstützt Redis die atomare Verwaltung von Listen, Mengen und Hashes, sodass Listenelemente, Set-Elemente und Hash-Einträge sicher hinzugefügt, entfernt und aktualisiert werden können. Dies gewährleistet die Genauigkeit auch dann, wenn mehrere Benutzer gleichzeitig auf die Daten zugreifen. Drittens kann Redis verwendet werden, um entweder einen Stack (nach dem LIFO-Prinzip; Last In, First Out) oder eine Warteschlange (nach dem FIFO-Prinzip; First In, First Out) zu erstellen. Redis garantiert die Atomarität seiner Operationen, was in Situationen wichtig ist, in denen Aufgaben oder Nachrichten in einer bestimmten Reihenfolge verarbeitet werden müssen.

---

Diese spezielle NoSQL-Datenbank speichert eindeutige Schlüssel und Verweise auf ihre zugehörigen Datenelemente in einer Hashtabelle, wie der Name schon sagt. Die Werte können einfache Skalare wie Ganzzahlen oder komplexere Formen wie JSON, Listen, BLOBs usw. sein. Integer, Zeichenfolgen, JSON-Dokumente und Arrays sind alle gültige Datentypen zur Speicherung und Abfrage von Informationen mithilfe eines Schlüssels. Typischerweise bietet sie eine ausgezeichnete Leistung und kann an die Bedürfnisse einer Organisation angepasst werden. Schlüssel-Wert-Speicher verfügen über keine Abfragesprache, ermöglichen jedoch das Hinzufügen und Entfernen von Schlüssel-Wert-Paaren, wobei einige Anbieter recht ausgeklügelte Funktionen bieten. Werte können nicht abgefragt oder durchsucht werden, nur der Schlüssel kann abgefragt werden

#### **2.7.25 Redis: Wann sollte eine Key-value-Datenbank verwendet werden? - Refat**

Key-Value-Datenbanken sind besonders geeignet, wenn eine Anwendung eine große Menge kleiner und kontinuierlicher Lese- und Schreibzugriffe verarbeiten muss. Diese Datenbanken sind für einen effizienten und schnellen Zugriff auf Daten konzipiert, die als Schlüssel-Wert-Paare gespeichert sind. Ob die Daten volatil oder häufig wechselnd sind, können Key-Value-Datenbanken eine leistungsstarke In-Memory-Zugriff bieten, was sie ideal für Anwendungsfälle macht, die eine schnelle Abfrage und Änderung von Daten erfordern.

Zudem sind Key-Value-Datenbanken gut geeignet, zur Speicherung grundlegender Informationen wie Kundenangaben, Benutzerprofile oder einfache Konfigurationen. In diesen Szenarien kann jede Information mit einem eindeutigen Schlüssel verknüpft werden, was eine einfache Abfrage und Aktualisierung basierend auf dem Schlüsselwert ermöglicht. Zum Beispiel kann eine Key-Value-Datenbank verwendet werden, um Webseiten mit der URL als Schlüssel und dem Webseiteninhalt als Wert zu speichern. Ebenso können Einkaufswageninhalte, Produktkategorien oder E-Commerce-Produktdetails effizient mit Key-Value-Datenbanken verwaltet werden.

Ebenso sind Schlüssel-Wert-Datenbanken eine gute Wahl für Anwendungen, die seltene Aktualisierungen oder einfache Abfragen erfordern. Wenn eine Anwendung hauptsächlich auf Datenabruf und grundlegenden CRUD-Vorgängen (Erstellen, Lesen, Aktualisieren, Löschen) basiert, bieten Schlüssel-Wert-Datenbanken eine effiziente und unkomplizierte Lösung.

---

Diese Datenbanken legen Wert auf Einfachheit und schnellen Datenzugriff und sind daher für Anwendungsfälle geeignet, die schnelle Suchvorgänge und Änderungen erfordern, ohne komplexe Abfragefunktionen oder umfangreiche Datenmanipulationen durchzuführen.

Darüber hinaus ermöglichen Key-Value-Datenbanken schnelle In-Memory-Zugriffe und sind ideal für Anwendungen, die eine hohe Anzahl kurzer, häufiger Lese- und Aktualisierungsvorgänge von Daten verarbeiten müssen, die sich ändern können.

#### **2.7.26 Redis: Anwendungsfälle für Schlüssel-Wert-Datenbanken - Refat**

Erstens sind Schlüssel-Wert-Datenbanken hervorragend für die Verwaltung von Sitzungsdaten in Anwendungen mit einer großen Anzahl von Benutzern geeignet. Sitzungen können Authentifizierungstoken, Benutzereinstellungen oder temporäre Daten enthalten, die alle effizient in diesen Datenbanken gespeichert und später abgefragt werden können. Aufgrund ihres schnellen InMemory-Zugriffs und ihrer Fähigkeit, große Mengen kleiner Lese- und Aktualisierungsvorgänge zu bewältigen, sind Schlüssel-Wert-Datenbanken die beste Option für das Sitzungsmanagement in Anwendungen mit einer großen Benutzerbasis.

Hinzufügend werden Schlüssel-Wert-Datenbanken häufig als Cache-Layer verwendet, um Anwendungsreaktionen zu beschleunigen. Durch das Zwischenspeichern von Daten, auf die häufig zugegriffen wird, in einem Schlüssel-Wert-Speicher können Anwendungen den Bedarf an teuren und zeitaufwändigen Vorgängen reduzieren, z. B. Datenbankabfragen oder komplexe Berechnungen. Diese CachingStrategie ermöglicht einen schnelleren Datenabruf, was zu einer verbesserten Anwendungsleistung und Reaktionsfähigkeit führt.

Zudem kommen Schlüssel-Wert-Datenbanken für die Aufbewahrung personenbezogener Daten bestimmter Nutzer in Verwendung. Die Aufbewahrung personenbezogener Daten bestimmter Nutzer ist ein Prozess, der sorgfältig und gemäß den geltenden Datenschutzbestimmungen und rechtlichen Anforderungen durchgeführt werden muss. Dabei muss man darauf achten, dass die Speicherung personenbezogener Daten bestimmter Nutzer eine rechtliche und ethische Verantwortung darstellt.

Schlüssel-Wert-Datenbanken sind aufgrund ihrer einzigartigen Funktionen und Fähigkeiten für bestimmte Anwendungsfälle geeignet. Um ein besseres

---

Verständnis für sie zu entwickeln, ist es hilfreich, sie mit anderen Arten von Datenbanken wie relationalen Datenbanken, Graphdatenbanken und Dokumentdatenbanken zu vergleichen. Dieser Vergleich ermöglicht es, die jeweiligen Stärken und Schwächen dieser Datenbanktypen herauszustellen.

#### **2.7.27 Redis: Redis mit PHP verbinden (new, pconnect, auth, set, publish) . Refat**

In PHP steht das aktivierte Redis-Modul zur Verfügung. Sie kann sich mit Redis verbinden, indem sie neues Redis-Objekt erstellt und dann über Methoden wie `pconnect`, `auth`, `set`, `publish` und andere auf den Redis-Server zugreifen. Die in Redis verwendeten Methoden `pconnect`, `auth`, `set` und `publish` sind Teil der PHP-Redis-Erweiterung und werden verwendet, um verschiedene Operationen mit Redis durchzuführen:

`pconnect`: Diese Methode wird verwendet, um eine persistente Verbindung zu einem Redis-Server herzustellen. Eine persistente Verbindung ist eine langanhaltende Verbindung, die wiederverwendet wird, um die Overhead-Kosten der Verbindungsherstellung zu reduzieren.

`auth`: Falls das Redis-Server eine Authentifizierung erfordert, wird diese Methode verwendet, um das Passwort anzugeben und sich am Redis-Server anzumelden.

`set`: Diese Methode wird verwendet, um einen Wert unter einem bestimmten Schlüssel in Redis zu speichern.

`publish`: Mit dieser Methode kann man eine Nachricht in einen Redis-Kanal (Channel) veröffentlichen. Dies ist nützlich, um das Publish-Subscribe-Muster zu implementieren, bei dem Nachrichten an verschiedene Teile einer Anwendung gesendet werden können.

#### **2.7.28 Git - Steve**

Entwickler stießen auf Probleme wie Zusammenführungskonflikte, Codeverlust, ineffiziente Koordination und Zeitverschwendung, wenn sie an Entwicklungsprojekten mit anderen arbeiten. Sie mussten sich oft auf manuelle Methoden verlassen, um ihren Code zu teilen und zu sichern, was zu Fehlern, Zeitverschwendung und Ressourcenverschwendung führte. Um dieses Problem zu lösen, wurde Git entwickelt.

---

In den letzten Jahren hat die Verwendung von Git plötzlich zugenommen. Große Open-Source-Projekte wie Linux, Entwicklerteams, Solo-Entwickler und sogar Studenten nutzen das beliebte Versionskontrollsystem

Git ist ein verteiltes Versionskontrollsystem, das entwickelt wurde, um die Verwaltung von Quellcode zu erleichtern. Es ermöglicht Entwicklern, Änderungen am Code nachzuverfolgen, verschiedene Versionen zu verwalten und effektiv zusammenzuarbeiten. Git bietet auch Funktionen wie Branching und Merging, um Konflikte zu vermeiden und den Entwicklungsprozess zu optimieren.

Durch die Verwendung von Git können Entwickler transparent arbeiten, Änderungen verfolgen und effizienter zusammenarbeiten, was zu einer verbesserten Koordination und Zeitersparnis führt.

Git ist dezentralisiert, das bedeutet, dass es keinen zentralen Server braucht, um ältere Dateien zu speichern. Das wird auch als Repository bezeichnet.

Außerdem ist Git auch eine Sammlung von Dienstprogrammen in der Kommandozeile, die Änderungen in Dateien verfolgen und aufzeichnen. Man kann auch alle möglichen Dateien wie Textdateien und sogar Bild-Dateien "tracken".

Durch diese Funktionalität kann man alte Versionen seines Projekts wiederherstellen, miteinander vergleichen, analysieren, Änderungen zusammenführen (mergen) und vieles mehr.

Für unsere Projektarbeit haben wir Git benutzt um effektiv an dem Projekt zu arbeiten. Damit kann dann jeder Mitarbeiter eine eigene Kopie des Projekts besitzen.

Um mit Git arbeiten zu können, haben wir erstmal auf unser hopper konfiguriert.

Wir haben dann ein bares Repository erstellt.

```
tfw-2023-f@hopper:~/steve$ mkdir bares
tfw-2023-f@hopper:~/steve$ chmod o-rwx bares
tfw-2023-f@hopper:~/steve$ ls -ld bares
drwxr-x--- 2 tfw-2023-f tfw-2023-f 4096 17. Sep 15:10 bares
```



---

Wir haben hier ein Repository `bares` erstellt und direkt mit dem Befehl `"chmod o-rwx"bares` ausgeführt. Mit `chmod o-rwx` ändern wir dann die Berechtigungen für unser Verzeichnis `"bares"` (mit `o-rwx` "remove read write and execute for others "). Das bedeutet, dass nur Benutzer mit den Rechten des Verzeichnisses (in unserem Fall `tfw-2023-f`) die Berechtigung zum Lesen, Schreiben oder Ausführen des Verzeichnisses haben.

Danach muss dann die Username und die E-Mail-adresse konfiguriert werden. Es wird dann also folgende Befehle ausgeführt:

```
-git config --global user.name "name"
```

Diese Konfiguration legt den Git-Benutzernamen fest. Wenn wir Änderungen vornehmen, wird die Name mit diesen Commits verknüpft.

```
- git config --global user.email name@gmail.com
```

Diese Einstellung legt deine E-Mail-Adresse für Git fest. Es wird verwendet, um den Account zu identifizieren, wenn wir mit Git arbeiten.

Mit dem Befehle: `"git config user.name"` können wir unsere username einsehen. Bei unserem Projekte ist unsere username `"tfw-2023-f"`.



```
tfw-2023-f@hopper:~$ git config user.name
tfw-2023-f
tfw-2023-f@hopper:~$
```

---

- `git init`:

Der Befehl "git init" wird verwendet, um ein neues Git-Repository in einem Verzeichnis zu initialisieren. Dies erzeugt ein neues leeres Repository, indem du die Änderungen in deinem Projekt verfolgen kannst. Dies ist der erste Schritt, um Git in einem Projekt zu verwenden.

Der Befehl "git init" sollte ungefähr folgendes Output liefern:

```
tfw-2023-f@hopper:~/steve/datei$ git init
Leeres Git-Repository in /home/tfw-2023-f/steve/datei/.git/ initialisiert
tfw-2023-f@hopper:~/steve/datei$
```

- `git status`:

Der Befehl "git status" zeigt uns die Dateien, die seit dem letzten Commit geändert wurden, die Dateien, die zum Commit bereit sind und die Dateien, die noch nicht von Git verfolgt wurden. Dies gibt uns einen Überblick über den aktuellen Status unseres Projekts und hilft uns zu verstehen, welche Änderungen gespeichert werden können.

Die Ausgabe von dem Befehl `git status` ist:

```
tfw-2023-f@hopper:~/junior$ git status
Auf Branch main
Ihr Branch ist auf demselben Stand wie 'origin/main'.
nichts zu committen, Arbeitsverzeichnis unverändert
```

Das hier bedeutet, dass das Repository aktuell leer ist "nicht zu committen", zudem sagt uns, dass keine Dateien vorhanden sind, die sie in das Repository einfügen können.

Diese Ausgabe wird angegeben wenn das Repository nur initialisiert wurde. Um den aktuellen Status von unserem Repository zu sehen müssen wir immer den Befehl `git status` geben (vgl. Fuchs 2021: 41-42)

- `git add` (Dateien zum Repository hinzufügen):

Der Befehl "git add" ist generell für "Änderung hinzufügen" und wird benötigt, wenn die Dateien bearbeitet haben nicht nur beim ersten hinzufügen. Die fügt dann eine Änderung aus dem Arbeitsverzeichnis zur Staging-Umgebung.

---

Dies ermöglicht Git, einzelne Änderungen zu überwachen, die wir an diesen Dateien vorgenommen haben. Indem wir die Dateien zum Index hinzufügen, können wir dann Git darauf vorbereiten, diese Änderungen auf organisierte und transparente Weise zu speichern. Dies ist ein effektiver Weg, um die Arbeit zu koordinieren und die verschiedenen Versionen unseres Projekts zu verwalten.

```
tfw-2023-f@hopper:~/refat$ git status
Auf Branch main
Ihr Branch ist auf demselben Stand wie 'origin/main'.

Unversionierte Dateien:
  (benutzen Sie "git add <Datei>...", um die Änderungen zum Commit vorzumerken)
       katalog1.php

nichts zum Commit vorgemerkt, aber es gibt unversionierte Dateien
(benutzen Sie "git add" zum Versionieren)
```

(vgl. Fuchs 2021: 42).

Die Datei muss dann jetzt auf das Repository hinzugefügt werden. Daher benutzen wir entweder den Befehl `git add` gefolgt von name der datei, die wir hinzufügen wollen. Oder wir können einfach so alle Dateien auf einmal hinzufügen mit `git add -A`.

`git add -A`

In diesen Fall werden alle Dateien in dem Verzeichnis hinzugefügt

Um zu überprüfen dass die Dateien erfolgreich hinzugefügt wurde, müssen wir noch den Befehl `git status` eingeben und wir werden dann diese Output am Rechner bekommen.

```
tfw-2023-f@hopper:~/refat$ git add -A
tfw-2023-f@hopper:~/refat$ git status
Auf Branch main
Ihr Branch ist auf demselben Stand wie 'origin/main'.

Zum Commit vorgemerkte Änderungen:
  (benutzen Sie "git restore --staged <Datei>..." zum Entfernen aus der Staging-Area)
       neue Datei:      katalog1.php
```

Das hier zeigt, dass die Datei zum Repository erfolgreich hinzugefügt wurde.

---

-git commit:

Ein Commit repräsentiert den Zustand unseres Repositorys zu einem bestimmten Zeitpunkt. Um die Änderungen zu validieren, die wir an unserem Projekt vorgenommen haben, verwenden wir "git commit". Dies erstellt eine neue Version mit einer Meldung (Nachricht), die die Änderungen erklärt. Es ist sehr wichtig, um unseren Fortschritt zu verfolgen, mit anderen Entwicklern zusammenzuarbeiten und bei Bedarf zurück zu gehen. Um "git commit" zu verwenden, müssen wir zuerst den Befehl "git add" verwenden, um die geänderten Dateien zum Index hinzuzufügen, und dann den Befehl "git commit" mit einer beschreibenden Meldung ausführen. Dies ist ein wesentlicher Bestandteil des Versionsverwaltungsprozesses Ihres Projekts.

Um einen neuen Commit zu erstellen, müssen wir mindestens eine Änderung im Staging-Bereich hinzugefügt haben (git add).

```
tfw-2023-f@hopper:~/refat$ git commit -m "steve hat die neue dateie hinzugefügt."  
[main 2b2cca0] steve hat die neue dateie hinzugefügt.  
1 file changed, 1 insertion(+)  
create mode 100644 katalog1.php
```

Git bestätigt, dass der Vorgang erfolgreich abgeschlossen wurde und die Datei von nun an verwaltet wird.

-git show

Der Befehl "git show" ist ein Git-Befehl, der uns hilft, die Details eines bestimmten commits zu sehen. Wenn wir "git show" ausführen, können wir die vorgenommenen Änderungen, die Commit-Nachricht, den Autor, das Datum und andere relevante Informationen sehen. Dies ist nützlich, um die in der Vergangenheit vorgenommenen Änderungen zu überprüfen und die Projekthistorie zu verstehen. Die Output könnte dann so aussehen:

```
tfw-2023-f@hopper:~/steve/toto$ git show
commit d88f1451b718871be54c9cec5af38ea404c6af75 (HEAD -> main)
Author: tfw-2023-f <tfw-2023-f@hs-bremerhaven.de>
Date:   Wed Sep 6 08:16:25 2023 +0200

    steaguiwoii hat das gemacht

diff --git a/datei.txt b/datei.txt
new file mode 100644
index 0000000..bf67b6d
--- /dev/null
+++ b/datei.txt
@@ -0,0 +1 @@
+ogjifdgfj
```

-git pull

Der git-pull-Befehl wird verwendet, um Inhalte aus einem entfernten Repository zu holen und herunterzuladen und das lokale Repository sofort zu aktualisieren. Das Zusammenführen von Remote-Upstream-Änderungen in Ihr lokales Repository ist in Git-basierten Collaboration-Workflows eine Der git pull Befehl ist eigentlich eine Kombination von zwei anderen Kommandos, git fetch gefolgt von git merge. Im ersten Schritt führt git pull einen git fetch auf den lokalen Zweig aus, auf den HEAD zeigt. Sobald der Inhalt heruntergeladen ist, tritt git pull in einen Merge-Workflow ein. Ein neuer Merge-Commit wird erstellt und HEAD aktualisiert, um auf den neuen Commit zu verweisen.

Die Ausführung von git pull ergibt folgende Ausgabe:

```
fw-2023-f@hopper:~$ git pull steve
remote: Objekte aufzählen: 42, fertig.
remote: Zähle Objekte: 100% (42/42), fertig.
remote: Komprimiere Objekte: 100% (38/38), fertig.
remote: Gesamt 42 (Delta 14), Wiederverwendet 0 (Delta 0), Pack wiederverwendet 0
ntpacke Objekte: 100% (42/42), 16.44 KiB | 765.00 KiB/s, fertig.
on steve
* branch          HEAD      -> FETCH_HEAD
fw-2023-f@hopper:~$
```

-git push

Git push ist ein Befehl, mit dem du Code-Änderungen an ein externes Repository wie GitHub senden kannst. Auf diese Weise können wir unsere Arbeit mit anderen Entwicklern teilen und Codeversionen zwischen Computer und dem entfernten Projektarchiv synchronisieren. Es ist eine praktische Möglichkeit, zusammenzuarbeiten und Ihren Code auf dem neuesten Stand zu halten.

---

Um den Befehl `git push` zu verwenden, müssen wir zuerst sicherstellen, dass wir ein Git-Repository in unserem Projekt initialisiert haben. Dann können wir unsere Änderungen mit `git add` und `git commit` hinzufügen und bestätigen. Wenn wir bereit sind, unsere Änderungen an das entfernte Projektarchiv zu senden, können wir den Befehl `git push` verwenden, gefolgt vom Namen des entfernten Projektarchivs und dem Zweig, in den du deine Änderungen verschieben möchtest.

-`git branch`

Ein branch in Git ist eine parallele Version deines Quellcodes. Es ermöglicht uns, an Funktionen oder Fehlerbehebungen zu arbeiten, ohne den Hauptzweig zu beeinflussen. Branch ermöglicht es uns, sicher zu experimentieren und zu entwickeln, da es vom Rest des Codes isoliert ist. Wenn wir mit der Arbeit an einer branch fertig sind, können Sie ihn mit dem Hauptzweig zusammenführen, um Ihre Änderungen einzubinden. Mit dem Befehl `git branch` können wir Branches erstellen, abrufen, löschen.

Der Standard-Branch jedes Repositorys heißt `master`. Um zusätzliche Branches zu erstellen, benutze den Befehl `git branch <name>`. Und mit dem Befehl `git checkout <name>` können wir in Git zwischen verschiedenen Branches wechseln

```
tfw-2023-f@hopper:~/steve$ git checkout aguiwo
Zu Branch 'aguiwo' gewechselt
tfw-2023-f@hopper:~/steve$
```

-`git log`

`Git log` wird benutzt um Liste von commits und deren IDs zu sehen. Jeder Commit verfügt über eine eindeutige ID. Diese ID wird in Form einer Reihe von Zahlen und Symbolen dargestellt.

---

```
commit 7fbc463689ecaf63d1f576d5a6ac3c0a4f19b53 (HEAD -> main)
Author: tfw-2023-f <tfw-2023-f@hs-bremerhaven.de>
Date: Mon Sep 4 21:45:46 2023 +0200

    steaguiwoii hat änderung hier gemacht

commit 39105cc73400e7d3910b89fe337d4a56bee14978
Author: tfw-2023-f <tfw-2023-f@hs-bremerhaven.de>
Date: Sun Sep 3 20:09:28 2023 +0200

    Hossein: Kommentar in Navbar hinzugefügt

commit 7346c3adea59ff81e5af1b07f4d5ee98904ef4cc
Author: tfw-2023-f <tfw-2023-f@hs-bremerhaven.de>
Date: Thu Aug 31 09:36:05 2023 +0200

    Weiterlung vom Regsitrieren zum Login mit js von Junior

commit 66dc2adc5809c9d8436404cb02c85efcd339d45c
Author: tfw-2023-f <tfw-2023-f@hs-bremerhaven.de>
Date: Mon Aug 28 12:04:38 2023 +0200

    Ende von Testen
```

(vgl. Arek 2022; vgl. Radfelder 2023; vgl. Haenel und Plenz 2016).

## 2.8 Externe Prozesse (Junior)

Im Rahmen der Implementierung unserer Projektidee haben wir uns vorgestellt, dass wir die Teilnehmeranzahl an Fitnesskursen auf 15 Personen pro Kurs und pro Tag begrenzen können, damit auch Redis bei der Umsetzung auftauchen kann. Das haben wir umgesetzt und dabei hatten wir uns auch gedacht, dass es besser wäre, wenn die Tabellen in der Datenbank tagtäglich aktualisiert werden könnten. Dies war notwendig, damit sich jeden Tag neue Nutzer für einen Kurs anmelden können und auch damit der Admin nicht jeden am Tag danach die Anmeldung vom Tag vorher immer sieht und er auch weiß, wer am heutigen Fitnesskurs teilnimmt. Daher brauchten wir einen Prozess, der die Tabelle jeden Tag leert, sodass nur die Anmeldungen für den laufenden Tag in die Tabelle gespeichert werden und entsprechend angezeigt werden. Daher haben wir uns entschieden zum Cronjob zu greifen.

---

### 2.8.1 Cronjob

Cron steht für "command run on notice", also sinngemäß, ein Befehl, der unter bestimmten Bedingungen ausgeführt wird. Cron erinnert natürlich auch an die griechische Mythengestalt Chronos, den Herrscher über die Zeit. Der CronJob ist also ein Befehl, der zu einer bestimmten Zeit ausgeführt wird und am Computer eingerichtet wird. Uaf der Kommandozeile auch Crontab genannt und dient dazu, Skripten wiederholt aufrufen zu können. Dabei haben wir eine weitreichende Kontrolle, zu welchen Zeitpunkten ein Skript aufgerufen werden soll. Die Datei, in der Sie angeben, welche Skripten wann laufen, wird irritierenderweise auch crontab genannt. (vergl. Rathmann und Wieskotten 2004 : 282)

### 2.8.2 Cronjob in Anwendung bei unserem Projekt

Bei unserem Projekt hatten wir uns zwei kleine bash-Skripte aufgebaut. Das erste "redis-projekt-tfw.sh" ist dafür zuständig die Zähler (auf Englisch: Counter) in Redis, die bei jeder Kursanmeldung oder -abmeldung inkrementiert bzw. dekrementiert werden, auf Null zu setzen und beim erfolgreich abgeschlossenen Vorgang wird der Satz: **Es hat geklappt** von Redis mit Zeitstempel in die Datei `crontab.log` gespeichert.

Das zweite Skript "reset-table.sh" stellt sicher, dass die eine Zeile in der Datei `init.sql` "DELETE FROM kurs\_angemeldet" ausgeführt wird und bei erfolgreich abgeschlossenen Vorgang ebenfalls eine positive Antwort mit Zeitstempel in `crontab.log` geschrieben wird.

Nun zum Cronjob: Wir haben folgende Zeilen zu unserem Cronjob hinzugefügt mit dem Befehl "`crontab -e`":

```
-----
1. 50 03 * * * /var/www/html/docker-tfw-2023-f/Projekt-tfw/redis-projekt-tfw.sh
2. 50 03 * * * /var/www/html/docker-tfw-2023-f/Projekt-tfw/reset_table.sh
-----
```

Hier stellen die zwei ersten Parameter jeweils die Minuten und die Uhrzeit der Ausführung dar. Die drei letzten Sterne stehen für Datum, Monat und Tag einer Woche. Eine explizite Erklärung liefert uns den Befehl "`crontab -e`" als Kommentare bei der ersten Ausführung zur Initialisierung vom Conjob. Hier wird dann sichergestellt, dass die zwei oben erwähnten Skripte jeden Tag um 03 Uhr 05 ausgeführt werden.



---

Anmerkung: Wir hätten das auch so machen können, dass wir in die Anmeldungen in der Datenbank mit Zeitstempel speichern und jedes Mal nur die aktuellsten gespeicherten Daten anzeigen.

### 2.8.3 Noch weitere Skripte

Bei so einem Projekt, wo viele Leute beim Codieren beteiligt sind, war es notwendig Git zur Versionsverwaltung zu nutzen und wenn wir uns alle einig waren, dass es in Ordnung war haben wir zum Testen die bereits gefertigten Teile mit einem deploy-Skript zum richtigen Verzeichnis verschoben. Unser Skript heißt "deploy.sh" und sieht so aus:

```
-----  
1. #!/bin/bash  
2. scp -r $1/* mydocker:/var/www/html/docker-$USER-web/Projekt-tfw/  
-----
```

Es besteht einfach aus einem Shebang und eine Zeile Code. Das Skript erwartet einen Argument \$1, das irgendwelches unserer Arbeitsverzeichnisse, wo gearbeitet, gepullt und gepusht wurde, ist.

Am Anfang hat jeder versucht aus dem eigenen Computer zu arbeiten, war es auch notwendig die angefertigten Dateien auf Hopper hochzuladen, bevor die Arbeitsumgebung mit git eingerichtet wurde. Dafür hatte sich jeder von uns, der einmal aus dem eigenen Computer gearbeitet hat so ein ähnliches deploy-Skript zur Verfügung:

```
#!/bin/bash  
scp -r Projekt-tfw hopper:  
ssh -t hopper sudo -iu tfw-2023-f scp -r /home/junekanenjoh/Projekt-tfw .  
ssh -t hopper sudo -iu tfw-2023-f ssh mydocker|  
~
```

---

## 2.9 Lasttest (Junior)

Üblicherweise gehört zu der Implementierung einer Anwendung auch eine Testphase dazu. Bei uns hatten wir die Anwendung bis hin zur Überprüfung aller Funktionen, die wir codiert haben. Allerdings wollten wir noch darüber hinaus gehen und prüfen, ob unsere Anwendung noch funktionsfähig wäre, wenn die von eine Fülle von Anfragen belastet wird. Dann entschieden wir uns einen Lasttest durchzuführen.

Dafür mussten wir uns vorab für ein geeignetes Werkzeug entscheiden. Zur Verfügung stand uns auf der Kommandozeile „ab“, „wrk“ und „curl“. Dann hatten wir auch die Möglichkeit ein moderneres Werkzeug zu benutzen und zwar K6, das mit JavaScript läuft. Nach Überlegung haben wir uns für „curl“ entschieden. Auch wenn wir damit die Grenzen nicht wirklich ausreizen konnten.

Wir haben uns ein kleines Bash-Skript aufgebaut und damit unser PHP-Skript zur Registrierung zehn tausend Anfragen gesendet. Beobachtet haben wir das Verhalten vom Server erstmal über vier Stunden. Dann hatten wir das Problem, dass die Daten, die wir aus „hbv\_dockerstatswatch“ entnommen hatten, nach zu viel Zeit nicht mehr richtig gespeichert werden. haben wir die Testzeit reduziert, bis wir auf eine Stunde gekommen sind. Zur Datenspeicherung und Beobachtung haben wir folgende Skripte verwendet:

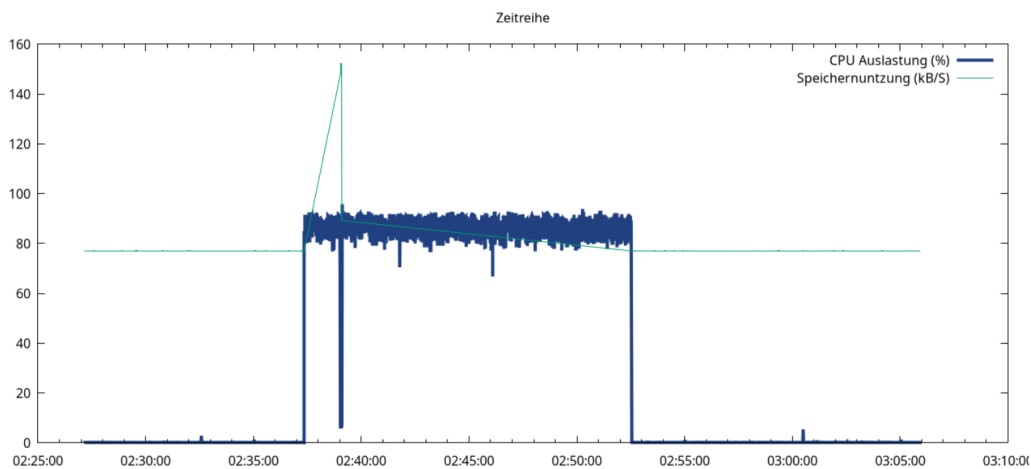
watch.sh

```
-----
1. #!/bin/bash
2. while hbv_dockerstatswatch | cut -f 1 >> hbv.log; do echo "ok";done
-----
```

Dieses Skript war dafür zuständig die Statistiken aus dem „hbv\_dockerstatswatch“ in einer Datei umzuleiten. Die Daten waren aber noch nicht einsatzbereit für gnuplot. Daher mussten wir die nochmal kurz bearbeiten und dafür das Skript „hbv2.sh“ geschrieben: hbv2.sh

```
-----
1. #!/bin/bash
2. # Extrahiere CPU-Auslastung und entferne '%' Zeichen
cut -d " " -f 2,9,14 hbv.log | sed 's/%//g' | sed 's/MiB//g' | \
sed 's/kB//g' | sed 's/\\//g' | sed 's/GiB//g' > cpu_und_speicher.dat
-----
```

Da haben wir die Daten in einer Form gebracht, die wir dann mit gnuplot als Grafik darstellen können. Dann ließen wir unser gnuplot-Skript „try.gp“ laufen.



---

```
1.set terminal pngcairo size 1280,600 font 'sans,12'
2.set out '/var/www/html/tfw-2023-f/lasttest.png'
3.set title "Zeitreihe"
4.set xdata time
5.set timefmt '%H:%M:%S'
6.set format x '%H:%M:%S'
7.plot 'cpu_und_speicher.dat' using 1:2 title 'CPU Auslastung (%)' \
with lines \
    linewidth 4 linecolor rgb '#204080', \
    'cpu_und_speicher.dat' using 1:3 \
with lines \
    title 'Speichernutzung (kB/S)'
```

---

Am Ende bekamen wir folgende Grafik ausgegeben:

wie auf der Grafik zu sehen ist, lief alles ganz normal bevor wir anfangen, zahlreiche Anfragen zu senden. Dann stieg die CPU-Auslastung hoch und zugleich auch die Speichernutzung. Beide sind über die gesamte Zeit, an der wir die Seite belastet haben fast stabil geblieben und fingen dann auch einiger Zeit danach an, zu sinken, weil alle Anfragen schon empfangen und bearbeitet wurden. Eine Bemerkung ist allerdings, dass wir beobachtet haben, dass sich die Seite so verhält, als würde uns „hopper“ zusätzlichen CPU-Speicher und Arbeitsspeicher zur Verfügung stellen, wenn benötigt wird. Zudem haben wir

uns nicht nur auf die Beobachtung unseres Docker-Containers begrenzt, wir wollten wissen, was auf der Seite passiert und was auch auf „hopper“ passiert.

Bei unserer Seite haben wir bemerkt, dass es mehr Zeit nahm, sich einzuloggen. Dies war ein Zeichen dafür, dass alles auf Atomizität bei der Datenbank läuft. Alles muss erst eine Anfrage bearbeitet werden, bevor eine andere angenommen und bearbeitet wird, was diese kleine Wartezeit verursacht hat. Auf „hopper“ haben wir mit dem Befehl „top“ das Verhalten vom Server beobachtet. Dabei haben wir festgestellt, dass wir im Laufe der Testphase die gleiche CPU-Auslastung (1,6) hatten und auch Speichernutzung (0,0). Schlussfolgern konnten wir nur, dass hopper sehr viele Ressourcen hat, sodass die Ausführung eines einzigen Programm keinen großen Einfluss hat.

Anmerkung: Ob die Operationen atomär laufen haben wir nicht nur auf die Seite geprüft. Wir ließen noch zwei mal „hbv2.sh“ auf separaten Fenstern mit „tmux“ laufen. Dabei haben wir festgestellt, dass es hier auch eine kurze Wartezeit gab, weil die Datenbank die Operationen nacheinander bearbeitet.

---

### 3 Reflexion und Fazit

Reflektieren wir als Team auf das bereits entwickelte Softwareprojekt zurück, für welche wir während des ganzen Semesters uns vorbereitet haben, so haben wir hierdurch nicht nur technische Fähigkeiten erworben, sondern zudem auch wertvolle Einblicke aus Entwicklerperspektive in die Welt der Softwareentwicklung, bzw. Webentwicklung gesammelt, mit der die Bedeutung effektiver Zusammenarbeit uns nähergebracht wurde. Die zu erlernenden Technologien aus dem Bereich der Webentwicklung, einschließlich HTML, CSS, JavaScript, PHP, MySQL sowie Git, die die Technik für Wirtschaftsinformatik auszeichnen, wurden uns durch dieses Projekt als zukünftige Softwareentwickler, für unsere weitere zukünftige Zusammenarbeit mit in die Hand gegeben.

Blickt man intern in den Verlauf unserer Softwareentwicklung hinein, so stellt man fest, dass für die Umsetzung unseres Ziels für das Softwaresystem, die Verwendung der Versionsverwaltung Git, durch die Möglichkeit für die Arbeit an parallel ablaufenden Codes und die damit verbundenen Änderungen, sich als effizient ausgezeichnet hat. Hinzukommend half uns Git dafür regelmäßig auch den Entwicklungsfortschritt beizubehalten.

Des Weiteren wurden uns durch die weiteren Technologien aus dem Bereich der Webentwicklung wie HTML, CSS, JavaScript, PHP und MySQL, eine gute Grundlage für die Arbeit für den Bereich der Webentwicklung verschafft. Einschließlich hierfür wurden uns beigebracht, benutzerfreundliche Oberflächen zu gestalten, Datenbanken zu verwalten, inhaltlich komplizierte Funktionen in unserem Code zu integrieren, sowie auch während des Entwickelns, für die Gewährleistung der Qualität des Codes, regelmäßige Code-Reviews, sowie Tests durchzuführen.

Neben der technischen Erkenntnisgewinnung zogen wir durch das Projekt ebenso die Lehre, zudem auch soziale Lernerfahrungen zu sammeln. Durch die regelmäßigen Online-Meetings sowie die damit parallel stattfindenden Präsenstreffen wurden unsere Sinne für die Koordination der Projektentwicklung, verschärft. Dies förderte nicht nur die individuelle Entwicklung jedes Einzelnen, sondern stärkte auch das Teamgefühl mit. Gemeinsam als Team müssten wir lernen, Schritt für Schritt Kompromisse einzugehen, Entscheidungen zum Wohle der Allgemeinheit zu treffen, sowie zugleich sicherstellen, dass wir vom Entwicklungspfad nicht abweichen.

Im Allgemeinen war das Projekt also eine lehrreiche & lohnende Erfahrung.

---

## Literatur

- Arek (2022). *Git Lernen in 30 Minuten – Anfänger Tutorial*. 07.10.2023. URL: <https://lerneprogrammieren.de/git/>.
- BlogNT : le Blog des Nouvelles Technologies (2011). 11.09.2023. URL: <https://www.blog-nouvelles-technologies.fr/wp-content/uploads/2011/08/la-structure-dune-page-html5-2.png>.
- Bonacina, Michael (2019). *PHP und MySQL für Einsteiger* -. BMU Media Verlag. ISBN: 978-3-966-45008-9.
- Carlson, Josiah L (2013). *Redis in Action* -. Birmingham: Manning. ISBN: 978-1-617-29085-5.
- Fuchs, Paul (2019a). *HTML5 und CSS3 für Einsteiger* -. BMU Media Verlag. ISBN: 978-3-966-45980-8.
- (2019b). *JavaScript Programmieren für Einsteiger* -. Landshut: BMU Media GmbH. ISBN: 978-3-966-45019-5.
- (2020). *SQL Handbuch für Einsteiger - der leichte Weg zum SQL-Experten*. Landshut: BMU Media GmbH. ISBN: 978-3-966-45071-3.
- (2021). *Git Handbuch für Einsteiger - der leichte Weg zum Git-Experten*. Landshut: BMU Media GmbH. ISBN: 978-3-966-45118-5.
- Haenel, Valentin und Julius Plenz (2016). *Git - verteilte Versionsverwaltung für Code und Dokumente*. München: Open Source Press. ISBN: 978-3-955-39119-5.
- Radfelder, Oliver (2023). *git kurz und knapp*. 07.10.2023. URL: <https://informatik.hs-bremerhaven.de/oradfelder/git.html>.
- Rathmann, Bettina und Christa Wieskotten (2004). *Jetzt lerne ich Shell-Programmierung - Effektiv mit der Linux-/Unix-Kommandozeile arbeiten*. München: Pearson Deutschland GmbH. ISBN: 978-3-827-26754-2.
- Sun, Yiyi (2019). *Practical Application Development with AppRun - Building Reliable, High-Performance Web Apps Using Elm-Inspired Architecture, Event Pub-Sub, and Components*. New York: Apress. ISBN: 978-1-484-24069-4.