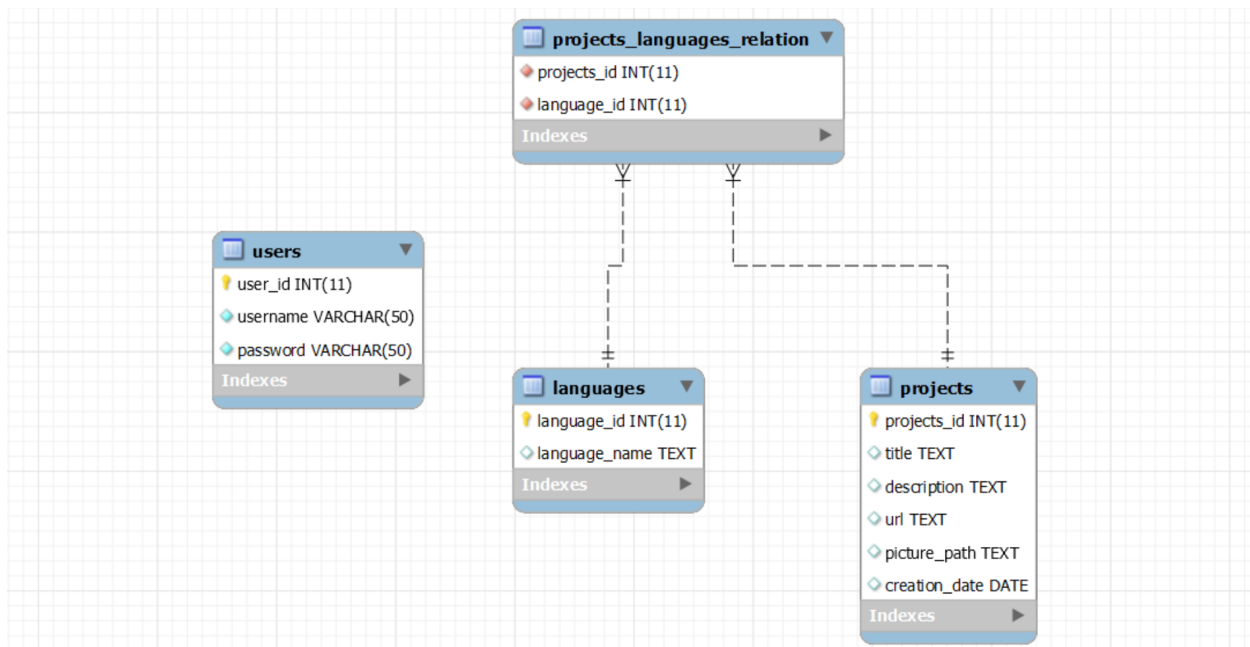


ICT Semesterprojekt Dokumentation auf Notion

Auswahl Projekt Thema

In letzter Zeit nutzen wir durch die Aufstockung von Mitarbeitern im Geschäft und durch Projekte in der Freizeit immer mehr GitHub. Eine Pinnwand, die unsere GitHub-Projekte übersichtlich darstellen, war für uns daher naheliegend.

ER-Model



SQL

Das SQL Script erstellt vier Tabellen, welche folgend kurz beschrieben werden.

users

separate Tabelle für User-Daten (id, username, password)

projects

Haupttabelle für Projekte und deren Daten (z. B. title, description, url etc.)

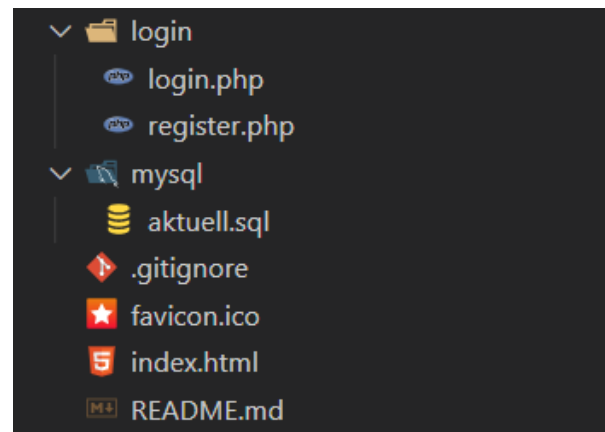
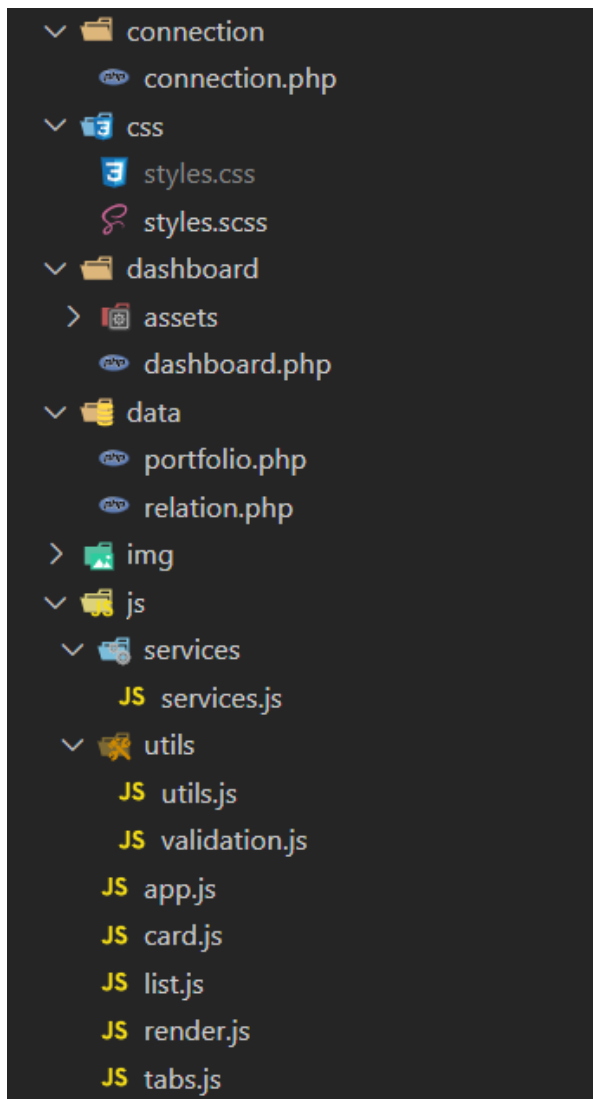
languages

Tabelle für Programmiersprachen, die einem Projekt zugewiesen werden können

projects_languages_relation

Zwischentabelle, die eine Relation zwischen projects und languages darstellt (Many-to-Many Relation), beinhaltet jeweils eine projects_id und eine language_id

Ordner und File Struktur



PHP

Der PHP Code ist in die vier Ordner connection, dashboard, data und login aufgeteilt.

Connection

Im Connection-Ordner ist die Verbindung zur Datenbank in einer Variable gespeichert. So kann dieses File dann jeweils bequem eingebunden werden, sollten wir eine Verbindung zur Datenbank brauchen.

Dashboard

Im Dashboard-Ordner ist das PHP-File für die Pinnwand enthalten, welches Session-Handling und ein Formular für das erstellen neuer Cards (Projekte) beinhaltet. Ausserdem sind dort auch

die HTML-Wrapper für die Cards und das Modal zu finden.

Login

Um die Bonuspunkte abzuräumen beinhaltet unser Projekt auch ein Login.

Das Login beinhaltet eine PHP-Funktion namens SESSION, welche einen sicheren Hash generiert, mithilfe dessen man den Client identifizieren kann. Im Dashboard wird man entsprechend zum Login weitergeleitet, wenn keine gültige Session-ID gefunden wird. So können sich nur User einloggen, die sich vorerst erfolgreich authentifizieren.

```
// bei Übereinstimmung (username und password richtig)
if ($count === 1) {
    // starten einer php session, speichern des usernamen aus sessionvariable
    session_start();
    $_SESSION['userid'] = $username;
    // weiterleitung zum Dashboard
    header("Location: /login/dashboard.php?username=" . $username);
} else {
    // fehlermeldung da kein User mit entsprechender username/passwort kombination gefunden wurde
    echo "
        <h1> Falsche Logindaten, bitte versuche es erneut oder registriere dich. </h1>
        <button onclick='history.back()' class='button'>Zurück</button>
    ";
}
```

JavaScript

services/services.js

In diesem File sind alle API-Services, also die Aufrufe an unsere REST-Endpoints. Bei Fehlern zeigen wir hier ausserdem eine Snackbar an.

utils/utils.js

Funktion für die Filterung der Sprachen und dem Ein- resp. Ausblenden des Modals.

utils/validation.js

Inputvalidation (onclick/onblur) und einblenden der Snackbar mit dynamischer Error-Message.

app.js

Alle JS-Files werden hier importiert und einige Event Listener werden definiert.

card.js

Logik für das Rendering der Cards und Funktionen für die Bearbeitung der Cards.

list.js

Logik für die Language Liste, mit welcher man die Programmiersprachen auswählen kann.

render.js

Im Render File ist wenig Logik es sind vor allem Ausgaben von HTML-Elementen vorhanden.

tabs.js

Eine Funktion für den Wechsel zwischen Registrierung und Login.

jsdocs

Wir haben uns während dieser Arbeit auch mit jsdocs vertraut gemacht. Unsere Kommentare haben wir daher nach diesem Standard dokumentiert, dafür gibt es nämlich auch schon praktische Kürzel in VS-Code. Das coole daran ist, dass man dann ein HTML-File mit allen Funktionen automatisch generieren kann. Um das HTML-File zu betrachten, muss der URL lediglich /docs angehängt werden.

bspw: <localhost/php-project/docs/>

CSS

Wir haben SCSS als Präprozessor für unser Styling benutzt. Wenn man SCSS kompiliert, wird daraus normales CSS generiert, welches wir dann in unser Projekt einbinden. Im Grunde ist SCSS nichts anderes wie CSS; es macht aber den Entwicklungsprozess etwas angenehmer. Wir haben ausserdem kein CSS-Framework eingebunden (bootstrap o. ä.), das ganze Projekt wurde mit purem CSS umgesetzt.

API-Services

Um die CRUD Funktionen in PHP zu erstellen haben wir eine Switch-Funktion mit je einem Case für eine Methode.

Weil die Daten mithilfe von JS aus den Input-Feldern zu PHP geschickt werden, müssen die Dateien welche sich im JSON Formate befinden, mit **json_decode** für PHP lesbar gemacht werden. Diese umkodierung wird für **POST** und **PUT** verwendet.

Für **GET** benötigt man **json_encode**.

Aufgerufen wird die API mit JavaScript.

Asynchrone Funktionen

Anstelle von **then/catch** haben wir für unsere asynchrone Funktionen **async/await** benutzt. Diese Funktion ist seit der Einführung von ES6 verfügbar, oft kann es unseren Code leserlicher machen, es erledigt aber die gleiche Aufgabe.

Hier ein Beispiel für den Aufruf unseres **GET** Endpoints:

```
// Projekte (Karten) laden und response als JSON zurückgeben
async function ladeProject() {
  const response = await fetch("../data/portfolio.php");

  if (response.ok) {
    return await response.json();
  }
  // Snackbar zeigen bei Fehler
  else {
    showSnackbar("Fehler beim Abrufen der Daten");
  }
}
```

Der Unterschied zwischen `async` und `then` liegt darin, dass `async` die Ausführung der Funktion anhält, bis das Promise erfüllt ist. Mit `then` wird die Funktion weiter ausgeführt aber den `.then`-Callback erst, wenn das Promise erfüllt ist.

weiterführende Informationen:

<https://www.smashingmagazine.com/2020/11/comparison-async-await-versus-then-catch/>

Cards

Beim Klicken auf eine Card öffnet sich ein Pop-up Window. Der Mülleimer ist für Löschen und der Stift für Bearbeitung. In der Übersichtsansicht sind der Titel, Beschreibung, Programmiersprachen und ein Bild zu sehen. In der Detailansicht ist zusätzlich der Link zum Projekt plus das Erstellungsdatum zu sehen.

Links

 [ICT Semesterprojekt Dokumentation auf Notion](#)




GitHub - 0xStoff/php-project

You can't perform that action at this time.

You signed in with another tab or window.

You signed out in another tab or window.

 <https://github.com/0xStoff/php-project>

0xStoff/**php-**
project



2 0 1 0
Contributors Issues Star Forks