**Frontend**

Allgemein

Das Frontend wurde mit Hilfe von Bootstrap erstellt und zusätzlich an einigen Stellen mit weitere CSS-Styles ergänzt. Des Weiteren wurde jQuery implementiert. Für einige Frontend-Funktionen und Überprüfungen wurde JavaScript eingesetzt. Im Folgenden werden die Dateien mit ihren Funktionen bzw. ihren Eigenschaften, welche für das Frontend von Relevanz sind, einzeln genannt und beschrieben.

Startseite

Die Startseite enthält den Auftritt nach Außen und ermöglicht es dem Nutzer sich anzumelden bzw. sich zu registrieren. Des Weiteren enthält es das Impressum.

* index.php

Die Index.php Datei stellt die Startseite mit der Anmeldung bzw. Registrierung dar. Diese Funktionen werden durch das Laden einer Navbar je nach Anmeldestatus angezeigt, sodass bereits angemeldete Nutzer über ein in der Navbar integriertes Dropdown Menü durch die Seite Navigieren können. Diese Navbar ist auch auf allen anderen Seiten zur Darstellung des Frontends eingebunden.

* navbar.html

Diese Datei enthält die Navbar für die Nutzer, welche noch nicht angemeldet sind bzw. sich erst registrieren müssen oder ihr Passwort vergessen haben.

* navbar\_log.php

Diese Datei enthält die Navbar für angemeldete Nutzer, welche mit Hilfe dieser auf der Seite navigieren können (Profil/Freunde/Zur Startseite/Ausloggen).

* pw\_vergessen.php

Diese Datei enthält das Modal zum anfordern eines neuen Passwortes.

Diese Datei enthält das Modal zum registrieren eines neuen Nutzers.

* footer.php

Hier wird das Copyright und das Impressum gesetzt. Durch das Einbinden auf jeder Seite ist die Erscheinung immer dieselbe.

Dashboard

Das Dashboard enthält alle Dateien, welche direkt mit den Projekten zu tun haben.

* profil.php

In profil.php sind zusätzliche CSS-Styles eingebunden, um zum eine Modal-Box und zum anderen eine Timeline zu realisieren. Hier werden die Projekte des Nutzers angezeigt und er kann mit ihnen interagieren (Ansicht/Annehmen/Ablehnen). Des Weiteren enthält es eine Sidebar mit der der Nutzer im Umfeld seiner Projekte Navigieren kann.

* projekte.php

Hier erhält der Nutzer eine Übersicht über alle Projekte, welche er selbst erstellt oder zu denen er hinzugefügt wurde. Des Weiteren enthält es neben den anstehenden Projekten auch alte Projekte. Auch hier kann der Nutzer mit einer Sidebar im Umfeld der Projekte navigieren.

* neue\_projekte.php

In dieser Datei wird das Formular zur Erstellung eines neuen Projektes dargestellt. Auch hier kann der Nutzer mit einer Sidebar im Umfeld der Projekte navigieren.

* aendern.html

Diese Datei enthält das Modal zum ändern von Projekten.

* gast\_loeschen\_modal.php

Diese Datei enthält das Modal zum löschen von Gästen (Sicherheitsabfrage).

* projekt\_loeschen\_modal.php

Diese Datei enthält das Modal zum löschen von Projekten (Sicherheitsabfrage).

Profil

* profil\_bearbeiten.php

Hier kann der Nutzer sein Profil bearbeiten (Benutzername/Email/Passwort ändern) und löschen. Hier werden die nachfolgenden Dateien für das Modal eingebunden. Diese Datei enthält außerdem weitere CSS-Styles für ein Modal.

* account\_loeschen.html

Diese Datei enthält das Modal zum löschen des eigenen Accounts (Sicherheitsabfrage).

* email\_aendern.html

Diese Datei enthält das Modal zum ändern der eigenen Email.

* passwort\_aendern.html

Diese Datei enthält das Modal zum ändern des eigenen Passworts.

* username\_aendern.html

Diese Datei enthält das Modal zum ändern des eigenen Benutzernamens.

Freunde

* freunde.php

Hier werden die Freunde des Nutzers angezeigt und es können neue Freunde hinzugefügt werden. Diese Datei enthält außerdem weitere CSS-Styles für ein Modal.

* freunde\_hinzufuegen.html

Diese Datei enthält das Modal zum hinzufügen von Freunden.

Errors

* error.html

Diese Datei enthält eine Fehlermeldung, die beim entstehen von Fehlern angezeigt wird.

**Technische Dokumentation Back-End**

Unser Back-End basiert auf PHP und einer Verbindung zu einem MYSQL Server. Dort haben wir unsere Tabellen angelegt und Administrieren die Seite über phpMyAdmin. Für viele Verbindungen zwischen dem dynamischen Frond-End und dem Back-End nutzen wir JavaScript. Dadurch können wir die vielen Features von PHP nutzen und trotzdem eine dynamische Seite erstellen und die modern und schlank wirkt.

Den größten Teil vom Back-Eend nehmen bei weitem die Datenbankverbindungen ein. Von den möglichen APIs haben wir uns für PDO entschieden. Dieses hat uns die Möglichkeit gegeben unsere Datenbank Version später auszuwählen. Zusätzlich bringt es mehrere Features mit die MysSQLi nicht hat wie zum Beispiel das einfache Erstellen von Arrays und das automatische erkennen von Variablentypen .Die Verbindungen werden immer durch dieselbe Struktur erzeugt. Zuerst wird die Verbindung mit der Datenbank aufgebaut. Dazu werden die Anmeldedaten in variablen gespeichert:



Im nächsten Schritt wird in einem Try/Catch block die Verbindung mit der Datenbank aufgebaut:



Hierbei wird nicht nur die Verbindung aufgebaut, sondern auch die Fehlermeldungen eingestellt und im Falle eines Fehlers ausgegeben. Im Try-block folgt daraufhin die Ein- oder Ausgabe. Damit dies sicher abläuft und vor SQL-Injection geschützt ist werden hier Prepared Statement genutzt. Das bedeutet das die Datenbank informiert wird welche Daten abgefragt werden und danach nur noch die Parameter bekommt:



So können die Daten sicher gespeichert werden oder wie im Beispiel einfach verwendet werden. Die Inputs werden zusätzlich nur mit der htmlspecialchars() Methode gesichert. Passwörter werden noch zusätzlich durch die password\_hash() Methode gehasht. Beim lesen werden sie dann nur über die password\_verify() Methode überprüft. So werden die Passwörter nie als Klartext gespeichert und sind in der Datenbank sicher mit bcrypt verschlüsselt.

Eine weitere wichtige Funktion unseres Back-Ends bietet die config.php und footer.php. Diese Dateien werden auf jeder Seite oben und unten Included. Dadurch kann der Footer und die Eigenschaften der Seite leicht auf der gesamten Website verändert werden. Die Config.php sorgt auch für die Sicherheit der Seite. Neben grundlegende Funktion wie den Nutzerdaten-lesen leitet sie nicht eingeloggte User auf die Startseite weiter. Dafür nutzen wir das in PHP integrierte Session System. Angemeldete Nutzer haben ihren Namen in der Session hinterlegt und müssen sich so nicht jedes Mal neu Anmelden.

Im Folgenden eine kurze Erläuterung über die Dateien der Website:

**Registrierung.php**

Überprüft die eingegebenen Daten mit der Datenbank. Ist der Username und die Email noch nicht vorhanden wird eine Email mit einem Code gesendet und dieser in die Datenbank zusammen mit den Nutzerdaten gespeichert.

**Pwvergessen.php**

Ermöglicht es sein Passwort zu ändern, wenn es vergessen wurde. Dafür erzeugt es ein zufällig erstelltes Passwort und sendet es an den Nutzer. Gleichzeitig wird das Passwort gehasht in der Datenbank gespeichert. Der Nutzer kann daraufhin das Passwort aus der Mail lesen sich Einloggen und sein Passwort ändern.

**Logout.php**

Löscht alle Daten aus der Session und beendet sie. Der Nutzer wird danach auf die Startseite verwiesen.

**Login.php**

Die Eingabedaten werden mit den Daten aus der Datenbank überprüft. Sind die Daten Identisch wird für den User eine Session erstellt. Hat er den Code noch nicht eingegeben wird das auch in der Session gespeichert um ihn dann nach dem Code zu fragen.

**Freunde\_db.php**

Hier werden zuerst die Freunde des Aktuellen Nutzers gesucht. Diese werden dann in einer foreach -schleife in Tabellenform ausgegeben. Bei jedem Freund wird zusätzlich noch das nächste gemeinsame Event angezeigt. Die Daten werden dazu schon in der Datenbank logisch gesucht. Dadurch wird zwar die Datenbankanfrage komplexer, die Logik auf der Website allerdings simpler und performanter.

**Profil\_bearbeiten Ordner (account\_loeschen.php/email\_aendern\_cancel.php/email\_aendern\_confirm.php/email\_aendern.php/freunde\_hinzufuegen.php/passwort\_aendern.php/username\_aendern.php)**

Die PHP Dateien in diesem Ordner dienen alle dazu Nutzerdaten zu bearbeiten. Sie sind daher alle recht simpel. Entweder tragen sie nur etwas in die Datenbank ein oder sie überprüfen es davor noch.

**Eigene\_projekte\_db.php**

Sucht nach Projekten von bei denen man selber der Besitzer ist (Besitzer =1). Diese werden dann mit einer Schleife ausgegeben.

**projekt\_ansicht\_orga\_gast.php**

Überprüft ob der eingetragene Gast existiert. Ist dies der Fall wird er dem Projekt hinzugefügt, wenn er noch nicht Mitglied ist.

**Projekt\_ansicht\_orga\_getraenk.php**

Überprüft ob ein Getränk schon in der Datenbank ist. Wenn dies noch nicht der Fall ist wird es in die Datenbank eigetragen. Danach werden die Menge und das Produkt der Liste hinzugefügt. Die Summe wird mal 100 genommen um Kommazahlen ohne Probleme darzustellen. In der Ausgabe wird dann /100 gerechnet.

**Projekt\_erstellen.php**

Die Daten für ein neues Projekt werden in der Datenbank gespeichert.

**Projekt\_getraenke.php**

Hier werden die Getränke die einem Projekt zugewiesen sind herausgesucht. Danach wird eine Tabelle erstellt in der in jeder Zeile der Name des Produkts und die Summe der mengen herausgesucht wird. So kann sich der User übersichtlich anschauen welche Getränke vorhanden sind.

**Projekt\_zuabsagen.php**

Diese Datei erzeugt die Anzeige der Gäste für die Projektansicht. Es werden die Gäste rausgesucht. In einer Schleife werden dann wieder für jeden Gast der Status und die Getränke rausgesucht.

**Projekt\_db.php**

Erzeugt die Ansicht aller Projekte. Im ersten Teil werden die Projekte gesucht die in der Zukunft stattfinden. Im unteren Teil werden dann die Vergangenen Projekte angezeigt.

**Projekt\_get\_daten.php**

Sucht nach einem Projekt in der Zukunft.

**Projekte\_set\_daten.php**

Aktualisiert die Daten für ein Projekt.

**Footer.php**

Beinhaltet den Footer.

**Config.php**

Prüft ob die Session vorhanden ist und auf welcher Seite sich der Nutzer befindet. Je nachdem ob er Angemeldet ist und auf welcher Seite er ist wird er auf eine andere Seite weitergeleitet. Der Seitentitel kann hier auch verändert werden. Weiter unten hat er noch Funktionen um die Nutzerdaten und Freundesdaten aus der Datenbank zu bekommen. Diese können von seiten genutzt werden wo diese Datei included ist.