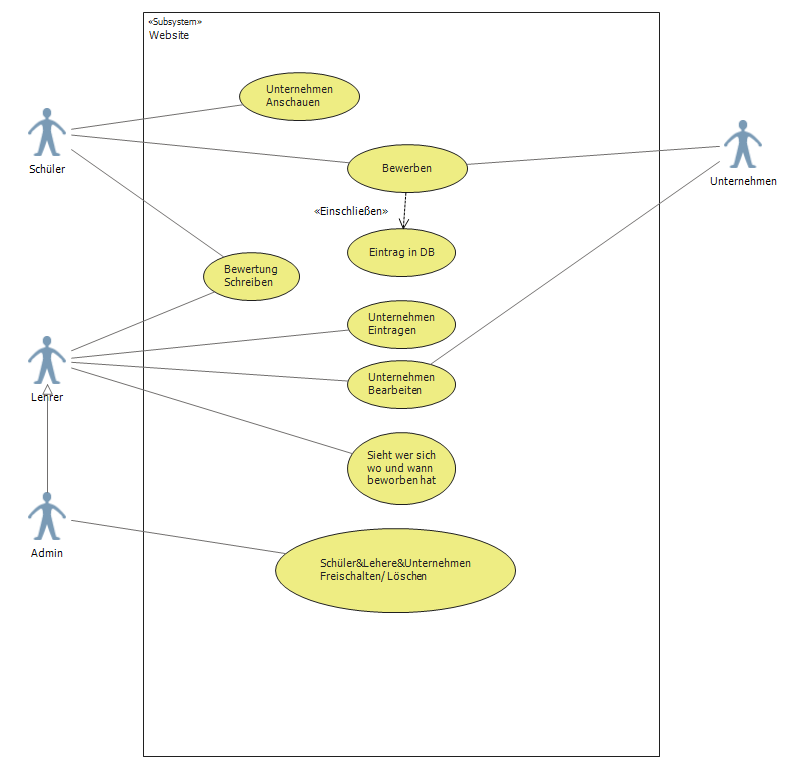
Planung

Zuerst haben wir uns überlegt was auf der Website möglich sein sollte. Nach kurzer Diskussion hatten wir uns auf Folgendes geeinigt:

* User können nach Unternehmen suchen die ein Praktikum anbieten
* Schüler können sich über die Website direkt bei den Unternehmen bewerben
* Unternehmen können angeben ob Sie immer Praktikanten annehmen oder nur für einen bestimmten Zeitraum
* Nur Admins und Lehrer können Unternehmen hinzufügen
* Lehrer und Admins können sehen wo sich Schüler beworben haben

Zur Veranschaulichung wurde auch ein Use-Case Diagramm erstellt:



Es wurde auch diskutiert ob Unternehmen Accounts bekommen sollten, womit Sie dann selbst ihr Unternehmen bearbeitet und Angebote erstellen könnten. Aus Zeitgründen wurde sich aber dagegen entschieden.

Es wurde auch überlegt, dass Unternehmen auch Ausbildungsstellen angeben können, so dass die Website von Schüler für mehre Sachen genutzt werden kann. Aus Zeitgründen wurde sich hier aber dagegen entschieden.

Außerdem sollte es auch für Unternehmen möglich sein sein Ausbildungsangebote in die Website rein zustellen. Aus Zeitgründen wurde sich hier auch dagegen entschieden.

Arbeitsaufteilung

Nachdem sich geeinigt wurde was die Website werden sollte, musste noch geklärt werden wer welche Arbeit übernimmt.

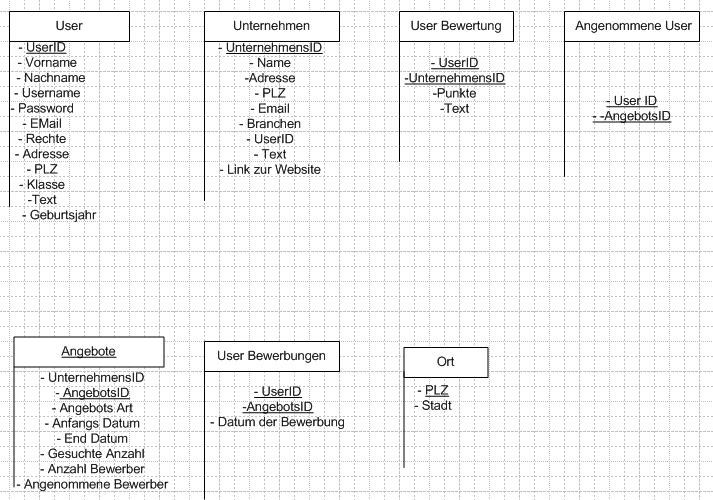
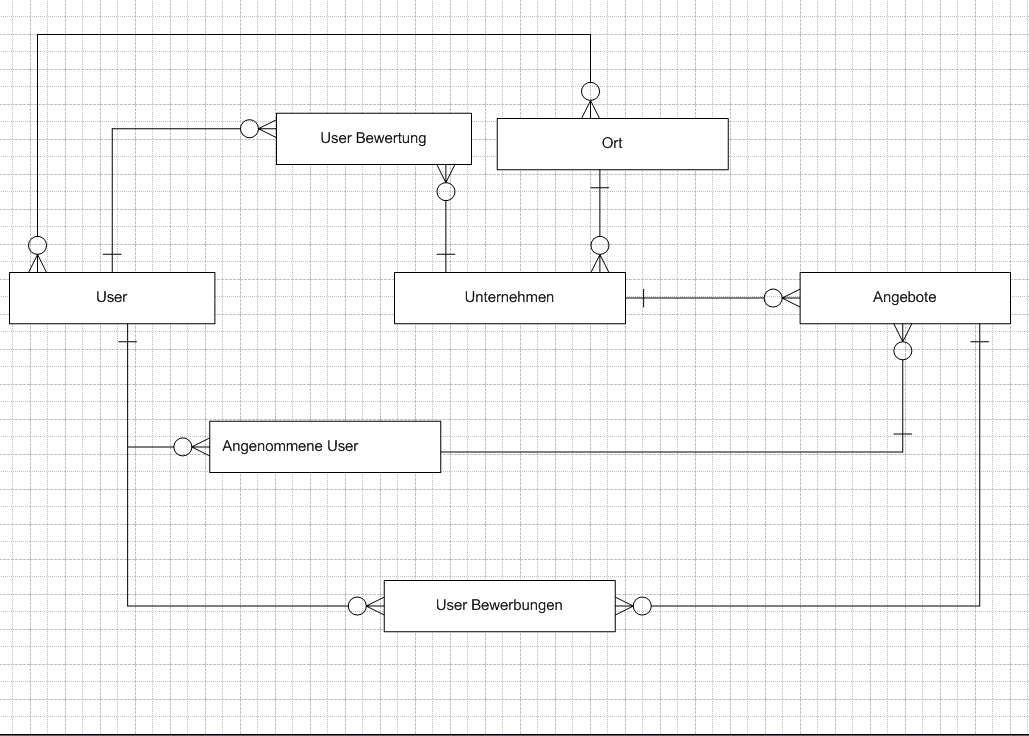
Lars: Lars hat die Aufgabe bekommen sich um den Gestalterischen Teil der Website zu kümmern.

Niklas: Niklas hat die Aufgabe bekommen sich um den Technischen Teil der Website zu kümmern.

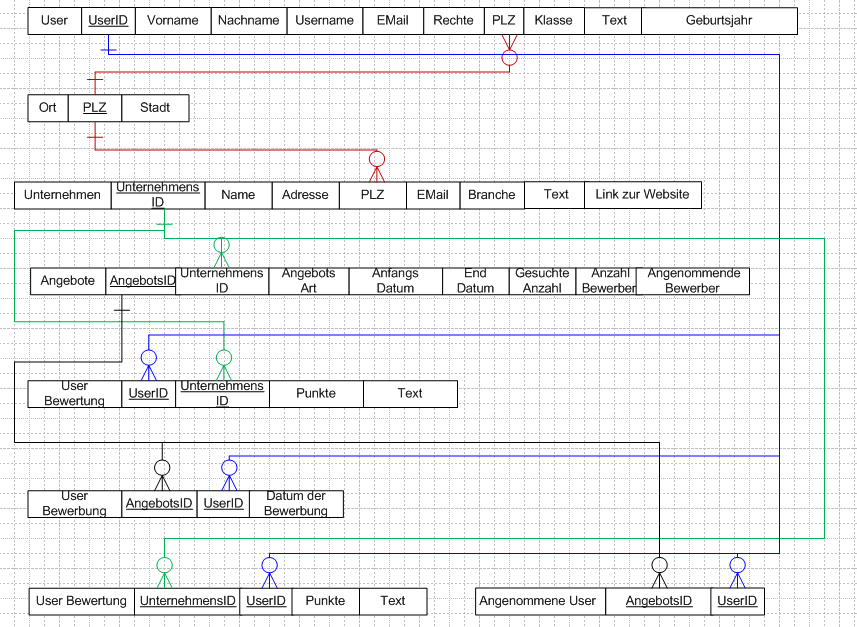
Sebastian: Sebastian hat die Aufgabe bekommen sich um die Datenbank zu kümmern und bei der Technischen Seite der Website zu helfen.

Aufbau der Datenbank

Entity Relationship Model



Datenstrukturdiagramm



In diesem Projekt haben wir uns entschieden gespeicherte Prozeduren zu verwenden. Eine Gespeicherte Prozeduren ist eine Art Funktion in der SQL Befehle gespeichert werden können, die beim Aufrufen der Prozedur ausgeführt werden. Gespeichert werden diese Prozeduren in der Datenbank.

*Anmerkung: In MySQL sind gespeicherte Prozeduren und Funktionen nicht dasselbe. Eine Prozedur führt nur die SQL Befehle aus die in ihr stehen, während eine Funktion noch zusätzlich immer eine return Variable hat die immer einen Wert zurück gibt.*

Der Grund warum wir uns für gespeicherte Prozeduren entschieden haben ist der Vorteil der Leistung, da der Ausführungsplan der für einen SQL Befehl erstellt werden muss gespeichert wird, also nicht bei jedem neuen Aufruf erstellt werden muss.

Ein weiter Praktischer Vorteil ist, dass gespeicherte Prozeduren gegen SQL-Injections schützen und dadurch die Arbeit besser Aufgeteilt werden kann, weil sich jeweils einer um den PHP und der andere um den SQL teil kümmern kann ohne dem anderen in dem Code gucken zu müssen.

Sicherheit

Neben gespeicherte Prozeduren werden auch noch Prepared Statements verwendet um nochmal sicher zu gehen das der User nicht in der Lage ist eine SQL-Injection durchzuführen. Die Prepared Statements werden aber nur verwendet wenn es sich um Felder handelt auf die jeder Normale User zugreifen kann. Bei Admins haben wir uns dagegen entschieden, weil wenn ein Admin wirklich gehackt wird es unserer Meinung nach in diesem Projekt keine Rolle mehr Spielt ob SQL-Injections noch möglich sind oder nicht.

Für den Fall das die Datenbank irgendwie in Falsche Hände geraten sollte wurden alle Passwörter noch gehasht bevor sie in der Datenbank Gespeichert wurden, damit nicht einfach alle Passwörter in Klartext sichtbar sind.

Probleme

Es ist auf Grund von geblockten Ports oder durch die Verwendung eines Proxys nicht möglich direkt auf die Google API von einem der Schul-PCs zuzugreifen. Darum hat Lars die Seite auf einen Webserver geladen.

Bei den Filtern standen wir vor dem Problem, dass es für jede Möglichkeit Theoretisch eine Gespeicherte Prozedur geben muss. In diesem Fall haben wir uns dann dafür entschieden auf Gespeicherte Prozeduren zu verzichten.

Es gab immer wieder Probleme mit dem Import der Datenbank. Auch auf dem Webserver. Das lag unter anderem an zu langsamen Schul-PC oder USB-Sticks oder auch an dem von Sebastian ausgewählten Datenbank Format „utf8\_croatian\_ci“. Dieses musste für den Webserver manuell geändert werden.

Benutzerhandbuch

Login

Für den Login in die PraktiDB Webseite wird der Benutzername oder die Email-Adresse und das Passwort des PraktiDB Kontos benötigt. Die Login Seite dient als Startseite für alle nicht angemeldeten Benutzer. Der Link unter dem Login Bereich bringt Sie zur Registrierung. Wenn sie sich erfolgreich eingeloggt haben werden sie zur Übersicht weitergeleitet. Wichtig: Ein Unternehmen kann sich nicht anmelden, da sie von den Lehrern und Admins verwaltet wird.

Registrieren

Es gibt drei Möglichkeiten sich für die PraktiDB zu registrieren: Als Schüler/in , als Firma und als Lehrer/in. Der Schüler/die Schülerin muss einfach nur das entsprechende Formular ausfüllen und kann sich dann einloggen. Die Firma muss erst vom Lehrer oder Admin freigeschaltet werden (mehr dazu unter Freischalten). Der Lehrer/die Lehrerin muss von einem Admin freigeschaltet werden. Aus Sicherheitsgründen ist nicht möglich sich als Admin zu registrieren. Um Admin zu werden sollten Sie sich zuerst als Schüler/in registrieren und dann manuell in der Datenbank die Spalte „vaUserRole“ ihres Kontos von „student“ zu „admin“ ändern. Wichtig: Ein Unternehmen kann sich nicht anmelden, da sie von den Lehrern und Admins verwaltet wird.

Navigationsleiste

Die Navigationsleiste enthält folgende Eintrage:

* Startseite bzw. Übersicht
* Profil: das Profil des angemeldeten Benutzers
* Kontrollzentrum (nur für Lehrer und Admins)
* Logout: meldet den Benutzer ab und sendet ihn zur Login Seite.

Übersicht (Overview)

Die Übersicht listet alle aktuellen Angebote. Diese Seite dient als Startseite aller angemeldeten Benutzer. Pro Angebot wird jeweils aufgelistet:

* das Unternehmen, das das Angebot anbietet,
* die Branche, in der das Angebot stattfindet,
* ein Zeitraum, in der das Angebot gültig ist,
* die Anzahl aller noch freien Plätze,
* die Entfernung, vom Benutzer zum Unternehmen (berechnet durch die Google API „DistanceMatrix“),
* die Durchschnittlich Bewertung des Unternehmens und
* ein Link über den sich der Benutzer für das Angebot bewerben kann.
* Das Unternehmen ist direkt zum dem Profil des Unternehmens verlinkt.

Bewerben

Bei dem Klick auf den Bewerben-Link in der Übersicht wird Nutzer zu einem „mailto“-Link weitergeleitet. Er hat nun selber die Entscheidung was er schreiben und anhängen möchte und ob er abschicken möchte. Der Nutzer wird schon als Bewerber markiert wenn er nur den Link zum Bewerben anklickt, da es leider nicht möglich ist, genauer zu überprüfen ob er sich wirklich beworben hat.

Profile

Benutzer

Das Benutzer Profil gibt einige Information zu dem Benutzer Preis. Dafür wird über die GET Methode die Benutzer-ID übergeben. Ist das Profil, das Profil des gerade angemeldeten Benutzers, dann hat dieser die Möglichkeit einen Text über sich und ein Profilbild hinzuzufügen im dem unten auf Ändern klickt. Ist der grade angemeldete Benutzer ein Lehrer oder Admin und das offene Profil, das eines Schülers, so wird auch ausgegeben wann der Schüler sich das letzte mal irgendwo beworben hat.

Unternehmen

Das Unternehmen Profil gibt einige Information zum Unternehmen Preis. Außerdem wird ein Route vom aktuell angemeldeten Benutzer zum Unternehmenssitz per Google Maps API angezeigt. Ist ein Lehrer oder Admin angemeldet hat er auch die Möglichkeit das zu bewerten. Die Bewertungen sind von jedem einsehbar über einen Link auf dieser Seite.

Profilliste

Es gibt keine öffentlich Profilliste. Die einzige Möglichkeit einige Profile aufzulisten ist die Klassenlist in dem Admin bzw. Lehrer Kontrollzentrum.

Kontrollzentrum

Auf das Kontrollzentrum haben nur Admins und Lehrer Zugriff wobei der Admin hier mehr Optionen hat. Der Lehrer darf:

* sich eine Klassenlist ausgeben lassen,
* Praktikumsangebote verwalten und erstellen,
* Unternehmen verwalten und erstellen,
* deaktivierte Unternehmen freischalten,
* sich eine Liste mit allen angenommen bzw. nicht angenommen Schülern ausgeben lassen,
* sich eine Liste mit allen besuchten bzw. nicht besuchten Praktikumsstellen ausgeben lassen,
* einen Schüler als angenommen markieren und
* eine Praktikumsstelle als besucht markieren.

Der Admin darf:

* alles was der Lehrer darf,
* deaktivierte Benutzer freischalten,
* Benutzer verwalten und erstellen und
* Orte verwalten und erstellen.

Deaktivierte Benutzer/Unternehmen freischalten

Benutzer und Unternehmen die erst noch freigeschaltet werden müssen werden in der Datenbank als deaktiviert markiert. Diese können von dem Kontrollzentrum aus unter „Deaktiverte Benutzer verwalten“ freigeschaltet oder gelöscht werden.

Google API

Es werden drei Google Maps APIs verwendet: „Google Maps Distance Matrix API“, „Google Maps Directions API“ und „Google Maps JavaScript API“. Um diese zu verwenden wird ein API Schlüssel benötigt. In der Datei „APIkey.txt“ ist so ein Schlüssel beigelegt, sollte dieser jedoch nicht mehr funktionieren so müssen sie einen neuen generieren. Gehen Sie dazu auf <https://console.developers.google.com/apis> und erstellen sie ein neues Projekt. Fügen Sie dann unter dem Dashboard die drei oben genannten APIs hinzu. Erstellen sie dann unter Zugangsdaten einen API-Schlüssel und schränken Sie ihn auf die drei APIs ein. Ersetzen Sie dann den Schlüssel in der „APIkey.txt“ Datei mit dem eben generierten Schlüssel.

Quellen

<http://php.net/>

<https://www.w3schools.com/>

<https://bit.ly/2JFUK0A>