

Fach AP

Berufsschule für Fachinformatik München

Projektdokumentation

Gruppe H: Sebastian Roßberger, Maximilian Beutner

Node.js (Backend), Angular (Frontend), Bootstrap (HTML-Styling),

Thema: Forum (Reddit-Abklatsch)

Inhaltsverzeichnis

Projektbeschreibung	3
Thema.....	3
Application-Stack.....	3
Farbschema	3
SCRUM-Protokolle	4
Vorwort	4
SCRUM-Board.....	4
Datenbank	5
Besonderheiten	5
Schema	5
Fazit	6
Schwierigkeiten	6
Zufriedenstellung	6
GitHub-Repository.....	6

Projektbeschreibung

Thema

Unsere Projektarbeit handelt generell von einem multi-themigen Forum, ähnlich wie man es von Reddit kennt. Unsere Idee war ein Forum, welches Benutzererstellte Posts in von Nutzern erstellten Topics speichert und diese leicht navigierbar macht. Zusätzlich können Admins Labels erstellen, welche von Nutzern an Ihren Posts angehängt werden können. Die Nutzer sollen sich dort registrieren und anmelden können.

Application-Stack

Unsere Frameworks haben wir anhand von den Anforderungen und unseren Knowhows entschieden. Da wir eine Benutzeroberfläche benötigen, haben wir uns auf eine Web-Applikation in *TypeScript* *Angular.js* entschieden. Die Datenbank, welche die Daten speichert, ist eine *MariaDB*-Datenbank, da wir bei dieser ebenfalls reichlich Erfahrung mitbringen. Damit unsere Frontend Applikation mit der Datenbank kommunizieren kann wird eine *REST*-Schnittstelle benötigt. Diese haben wir in *Node.js* mit dem API-Server Package *ExpressJS* geschrieben, da diese eine Antwort in *JSON*-Format anbieten und in der Sprache JavaScript geschrieben wird.

Farbschema

Unsere Farbpalette besteht aus mehreren Tönen von Grasgrün mit sekundären Weiß- und Grautönen. Für Formular-Aktionen verwenden wir für das Anlegen und Bearbeiten von Daten blau und für das Löschen von Datensätzen rot.

SCRUM-Protokolle

Vorwort

Unsere SCRUM-Protokolle bestehen aus einem Auszug der Issues unseres GitHub-Boards. Man muss dazu sagen, dass die Karten bei Laufzeit erstellt wurden. Jedoch haben wir trotzdem (leider unprotokollierte) SCRUM-Meetings gehalten, wir haben lediglich keine Notizen von diesen gemacht. In manchen Fällen wurden Karten vergessen zu erstellen und in den meisten Fällen wurden diese im Nachhinein erstellt. Leider fehlen allerdings auch ein paar wenige Karten.

SCRUM-Board

Aufgabe	Arbeitszeit	Bearbeiter
GitHub Repository & Board einrichten	1 Stunde	Sebastian R., Maximilian B.
Entwicklungsumgebung aufsetzen	1 Stunde	Maximilian B.
UI Gestaltung planen & Homepage entwerfen	2 Stunden	Sebastian R.
Angular Basic Routing konfigurieren	1 Stunde	Maximilian B.
Datenbankschema planen & umsetzen	1,5 Stunden	Maximilian B.
Entwurf für Posts Seite erstellen	2,5 Stunden	Sebastian R.
Entwurf für User Seite erstellen	1 Stunde	Sebastian R.
Auf Projektnamen einigen	0,5 Stunden	Sebastian R., Maximilian B.
Code-Migration aus Projektentwurf	2 Stunden	Maximilian B.
Entwürfe in Angular-Projekt umziehen (nur Front-End)	2,5 Stunden	Sebastian R.
Plugin für Fontawesome-Icons integrieren	1,5 Stunden	Sebastian R.
API für User in Back-End coden	3 Stunden	Maximilian B.
Front-End für Login bauen	1,5 Stunden	Sebastian R.
Datenmodelle in Angular anlegen	1,25 Stunden	Maximilian B.
API für Posts & Kommentare in Back-End coden	2 Stunden	Maximilian B.
Front-End für Topic bauen	1 Stunde	Sebastian R.
API für Topics in Back-End coden	3 Stunden	Maximilian B.
Suchleiste aus Navigationsleiste entfernen	0,25 Stunden	Sebastian R.
Labels in Posts Front-End implementieren	2,5 Stunden	Sebastian R.
Front-End Login funktional integrieren	2,5 Stunden	Maximilian B.
Sitemap erstellen	0,5 Stunden	Sebastian R.
Topic Erstellung ins Front-End einbauen	1,25 Stunden	Sebastian R.
Kommentar Erstellung ins Front-End einbauen	0,75 Stunden	Sebastian R.
Label Verwaltung in Back- & Front-End coden	4,5 Stunden	Sebastian R.
Post Editor in Front- & Back-End integrieren	6,5 Stunden	Sebastian R.
User-, Post- und Topic API ins Front-End integrieren	ca. 13-15 Stunden	Maximilian B.
Bugfixing gesamt	7 Stunden	Sebastian R., Maximilian B.
Datenbank-Dump erstellen und Daten vorbereiten	1 Stunde	Sebastian R.

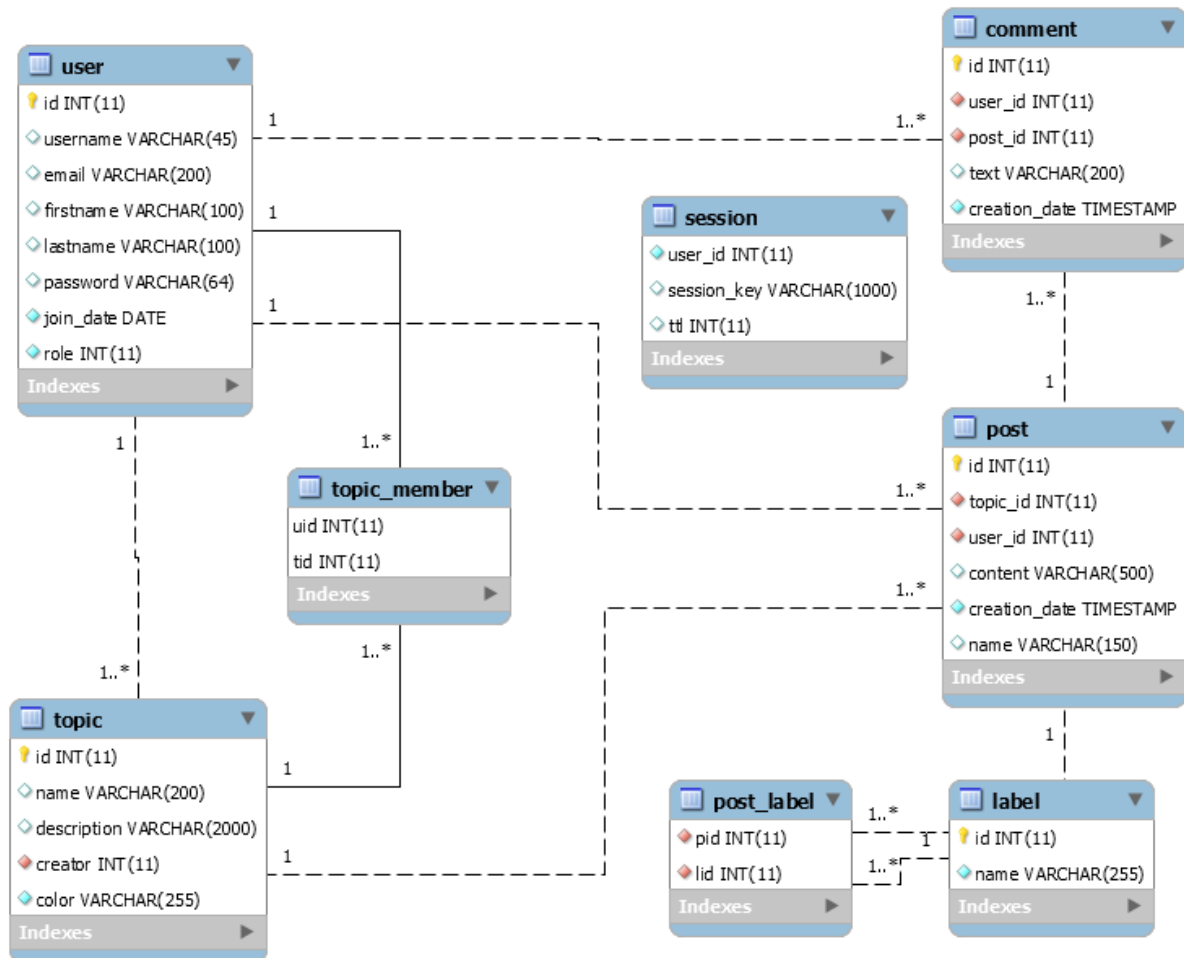
Summe der gesamten Arbeitszeit: 75,75 Stunden

Datenbank

Besonderheiten

Wir haben unsere User Sessions in einer Datenbank-Tabelle festgehalten, was nicht sehr gewöhnlich ist. Dies haben wir nur gemacht, da dies sehr schnell einzurichten ist und funktionell sich nicht unterscheidet. Eine öffentlich-zugängliche Anwendung sollte dies aber aus Sicherheitsbedenken nicht verwenden, für eine Anwendung, welche nur in einem privaten Kreis verwendet wird, ist dies aber irrelevant.

Schema



Fazit

Schwierigkeiten

Wir hatten natürlich auch Schwierigkeiten. Vor allem gegen Ende hatten wir sehr viel Stress und mussten teilweise bis in die Nacht daran arbeiten, um ein zufriedenstellendes Produkt liefern zu können. Dies liegt an unserer überambitionierten Planung. In der Theorie schien das Projekt nicht sehr schwer, weshalb wir bereits in der ersten Planung Quality-of-Life Features geplant haben. Erst als es bereits zu spät war um das Thema zu ändern oder zu minimieren bemerkten wir das Problem. Darunter litt auch unsere Handhabung des GitHub-Boards und somit die SCRUM-Protokollierung, was wiederum sich auf die Effizienz unserer Coding-Sessions negativ auswirkte.

Zufriedenstellung

Trotz allen Problemen und dem ausgearteten Umfang unserer Arbeit sind wir sehr zufrieden mit unserem Projekt und haben die meisten geplanten Features umsetzen können. Wir hoffen mit dem fertigen Projekt sie von unserer Zufriedenstellung überzeugen zu können.

GitHub-Repository

Anbei noch das GitHub Repository des Projekts:



https://github.com/Probastian/berufsschule_projekt