Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais (IFSULDEMINAS)
Engenharia de Computação
Projeto Prático – Programação Web II
Prof. Dr. Ricardo Ramos de Oliveira
Data de entrega: 28/10/2019

INSTITUTO FEDERAL
Sul de Minas Gerais
Campus
Poços de Caldas

Blog Pessoal

Observação: Antes de ler a descrição do trabalho é importante conhecer a sigla **CRUD** que significa ***Create***, ***Read***, ***Update*** e ***Delete***, ou seja, representa as 4 operações básicas do blog. Desenvolva o seu Blog pessoal com as tecnologias do **Angular Material** no *front-end*, **NodeJS** no *back-end e o framework* **expressJS** *para trabalhar com os web services.* Siga as especificações a seguir:

1 – Crie um banco de dados com o nome **blog** utilizando o banco de dados MySQL e o Workbench mostrados em sala de aula;

2 – Faça um CRUD de usuários para a tabela:

```
CREATE TABLE `user` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`name` varchar(100) NOT NULL,
`username` varchar(45) NOT NULL,
`email` varchar(50) DEFAULT NULL,
PRIMARY KEY (`id`)
);
```

3 - Faça um CRUD de postagens para a tabela:

```
CREATE TABLE `post` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`title` varchar(50) NOT NULL,
`body` text NOT NULL,
`user_id` int(11) NOT NULL,
PRIMARY KEY (`id`),

CONSTRAINT post_fk_user FOREIGN KEY (user_id) references user (id)
);
```

4 – Faça um CRUD de comentários para a tabela:

```
CREATE TABLE `comment` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`title` varchar(50) NOT NULL,
`body` text NOT NULL,
`email` varchar(50) NOT NULL,
`post_id` int(11) NOT NULL,
PRIMARY KEY (`id`),
```

```
CONSTRAINT comment_fk_post FOREIGN KEY (post_id) references post
(id)
);
```

5 – O blog deve conter o seguinte menu com os seguintes links:

5.1 – Principal – Página com todos os *posts* de todos os usuários ordenados por data e hora considerando do mais recente para o *post* mais antigo (Inserir os campos data e hora na tabela de posts do blog – utilize o comando *alter table* para isso);
5.2 – Os posts devem estar no formato de linha do tempo na página principal. Somente o usuário principal ou administrador consegue criar os posts e os demais usuários que tiverem acesso ao link do blog conseguem comentar os posts do usuário principal;
5.3 – Deve-se listar na página principal apenas os 10 primeiros posts de pelo menos 3 usuários diferentes;
5.4 – Insira um ou mais comentários em 2 posts para testar o blog;
5.5 – Cadastro – Essa página é responsável por cadastrar os usuários do sistema;
5.6 – Campo de busca – O campo de busca deverá aparecer no menu e poderá ser utilizado em qualquer página, a ideia é realizar uma busca nos posts e comentários dos usuários (A busca deve ser por substring independente de acentos, maiúsculas e minúsculas e não apenas buscas exatas.);
5.7 – Página sobre com as seguintes informações da dupla: Foto, nome completo, e-mail institucional, telefone, nome completo da instituição onde estuda e o nome do curso;
5.8 – Um post pode conter vários comentários de diferentes usuários;
5.9 – Página de Contato com os campos: Nome, E-mail e Mensagem com os botões enviar e limpar. Essa página deve conter também os ícones de contato para as redes sociais da dupla, conforme o exemplo a seguir:

Sugestão utilize a plataforma: www.addthis.com

5.10 – Construa um *layout* adequado como, por exemplo, o blog https://plasticneko.github.io/bulma-blog-simple/, lembre-se de estruturar o seu *layout* utilizando o conceito de *cards* para inserir diferentes tipos de conteúdo dentro do seu *layout*.

6 - Cada publicação de um post deverá conter uma imagem associada - Inserir Imagem no Blog. A imagem deve obrigatoriamente consumir a API de imagens: https://picsum.photos/

Observação:
Deve se construir duas aplicações, uma aplicação para o *front-end* e outra aplicação para o *back-end* e ambas devem-se comunicar por meio do *framework* expressjs. Implemente as operações de CRUD com os métodos HTTP GET, POST, PUT, PATCH e DELETE tanto no lado do servidor para a resposta fornecendo os dados do banco de dados através do web service REST quando do lado do cliente para realizar a requisição de forma adequada.

Material de Apoio – Importante

Parte 1 de 3
https://medium.com/@andrewchanm/criando-um-app-angular-7-e-consumindo-uma-api-rest-1-de-3-7169d90ed8c1
Parte 2 de 3
https://medium.com/@andrewchanm/criando-um-app-angular-7-e-consumindo-uma-api-rest-2-de-3-5747972ef56e

Parte 3 de 3
https://medium.com/@andrewchanm/criando-um-app-angular-7-e-consumindo-uma-api-rest-3-de-3-7d3b22aa09a6

Boas Práticas para o desenvolvimento de uma API REST

http://www.matera.com/blog/post/boas-praticas-para-desenvolvimento-de-apis-rest

Desafio – 0,5 Ponto Extra

1 - Tela de Login para os usuários
A tela de login deverá ter duas opções de acesso ao sistema

1ª Opção – Acesso por meio do cadastro do usuário no próprio blog. Caso o usuário ainda não esteja cadastrado ele conseguirá realizar o cadastro na própria tela de login por meio de um link para a página criada no item 5.2.
2ª Opção – O acesso acontecerá por meio de um Token de autenticação com o Facebook e com o Google

Os comentários de um post só poderão ser feito por usuários que estão logados no sistema, caso o usuário não esteja logado, deve-se exibir uma mensagem que o usuário não está logado e fornecer a opção do usuário se cadastrar por meio de um link que o direcionará para a página de cadastro de usuário

2 - Editor de texto avançado

O editor de texto mais avançado deverá conter as opções de negrito, itálico, inserir imagem e etc… (Sugestão, você poderá utilizar algum *plugin* para isso). O objetivo desse editor é facilitar a criação de posts no blog por usuários leigos em informática.