Trabalho Backend

Node.js - REST

Universidade Federal do Ceará

Projeto de Interfaces Web

Prof. Victor Farias

**Data de entrega**: 20/10/20

Entrega zipado via Moodle

**Atributos das entidades**

**Usuário** (id, nome, email, senha)

**Post** (id, texto, likes, uid)

**Esquema de endpoints:**

* **Usuários** 
  + **POST** /api/usuarios
    - Recebe usuário e armazena
    - Retorna usuário com id
  + **GET** /api/usuarios
    - Retorna todos usuários
  + **GET** /api/usuarios/:id
    - Retorna usuário com id :id
  + **PUT** /api/usuarios/
    - Recebe usuário emodifica usuário logado
  + **DELETE**  /api/usuarios/
    - Remove usuário logado
  + **GET** /api/usuarios/:id/posts
    - Retorna todos posts de um usuário
* **Posts**
  + **GET**  /api/posts
    - Retorna todo os posts
  + **GET** /api/posts/:id
    - Retorna post com id :id
  + **POST** /api/posts
    - Recebe novo post e o armazena com uid do usuário logado
  + **PUT** /api/posts/:id
    - Recebe post com id :id e omodifica
    - Só pode modificar se for do usuário logado
  + **DELETE**  /api/posts/:id
    - Delete por com id :id
    - Só pode deletar se pertencer ao usuário logado
  + **GET** /api/posts/:id/usuario
    - Retorna usuário de um post

**Parte 1 - Estrutura**

1. Construa uma API Rest usando a estrutura de diretórios mostrada em sala segundo o esquema de endpoints
2. Nos endpoints GET, retorne uma resposta estática.
   1. Ex: endpoint GET /api/alunos/ retorna uma lista estática de alunos
3. Nos endpoints do tipo POST, PUT e GET, retorna somente uma mensagem informando que a operação foi executada com sucesso

**Parte 2 – Conexão banco**

1. Armazene todos os dados no MongoDB via Mongoose
2. Os endpoints GET /api/usuarios recebem parâmetros de consulta **nome** para pesquisar pelo nome (substring) e **email** para pesquisar pelo e-mail (exato)

**Parte 3 – Autenticação e segurança**

1. Adicionar o seguinte endpoint
   1. **POST** /api/usuarios/singin
      1. Recebe email e senha
      2. Retorna o token jwt
2. Ao armazenar usuário em GET /api/usuários, o banco deve armazenar apenas o hash das senhas
   1. A assinatura hash da senha nunca deve ser retornada com os usuários
   2. Todo os endpoints devem receber o token e validá-lo, salvo o POST /api/usuarios e o POST /api/usuarios/signin
3. Proibir que um usuário possa modificar (PUT e POST) ou remover recursos de outro usuário, mas que tenha o direito de acessá-los (GET).