



## Programação para a Web II - 2020/2 Trabalho Prático

### Introdução

O trabalho prático da disciplina de Programação Web II será a implementação de uma loja virtual para venda de produtos diversos. A aplicação deverá ser desenvolvida em Node.js, Express, Sequelize e React.

### Regras do Trabalho Prático

A implementação da aplicação deverá seguir TODAS as regras abaixo:

**Regra 01:** A aplicação deverá ser uma **Single Page Application (SPA)**, onde toda sua funcionalidade está concentrada em uma única página que é carregada no browser do usuário assim que ele acessa a aplicação na primeira vez. O backend deverá ser desenvolvido usando Express e Sequelize, e o frontend deverá ser desenvolvido em React.

**Regra 02:** A aplicação terá 2 tipos de usuários: cliente e colaborador. Para cadastrar um novo colaborador, é necessário que outro colaborador acesse a aplicação e preencha um formulário de cadastro de novo usuário. Em relação aos usuários clientes, esses poderão fazer seu próprio cadastro na aplicação através da rota **/signup**. Os campos do formulário de cadastro (tanto de cliente quanto de colaborador) são: nome completo (precisa ter entre 5 e 100 caracteres), endereço de e-mail (precisa ser um email válido), senha de acesso e confirmação de senha. A senha precisa ter 6 caracteres ou mais, e é importante verificar se a senha digitada pelo usuário é igual à senha digitada no campo de confirmação.

**Regra 03:** Os usuários colaboradores poderão adicionar novos produtos no banco através de um formulário da aplicação. Os campos desse formulário são: nome, descrição, preço e estoque. Os usuários clientes não terão acesso a esse formulário.

**Regra 04:** Os usuários clientes ou visitantes (que ainda não fizeram login) poderão acessar as páginas de cada produto, selecionar uma quantidade a ser comprada, e adicioná-lo em seu carrinho de compra. Quando o usuário decidir concluir a compra, ele deverá efetuar o login (caso ainda não o tenha feito) e então será direcionado para uma página onde poderá escolher o endereço de entrega dentre os endereços já cadastrados, ou então poderá cadastrar um novo endereço através de um formulário.

**Regra 05:** As senhas deverão ser armazenadas no banco de dados de forma criptografada, através do módulo **bcrypt** (vide slides da disciplina).

**Regra 06:** Quando o usuário não estiver logado na aplicação, o menu superior deverá conter apenas as opções: **Sobre** (rota /sobre), **Meu carrinho** (rota /cart), **Login** (rota /login) e **SignUp** (rota /signup). Ao acessar a tela de login (que deverá conter apenas os campos **email** e **senha**) e informar as credenciais corretamente, o menu superior deverá passar a conter as seguintes opções: **Sobre** (rota /sobre), **Meu carrinho** (rota /cart) e **Logout** (rota /logout).

**Regra 07:** O banco de dados deverá obedecer o esquema da Figura 1. Cada uma das tabelas deverá conter um Modelo e ao menos uma Migração. A tabela **TipoUsuario** deverá

ser alimentada através de um **seeder** do Sequelize (vide slides da disciplina).

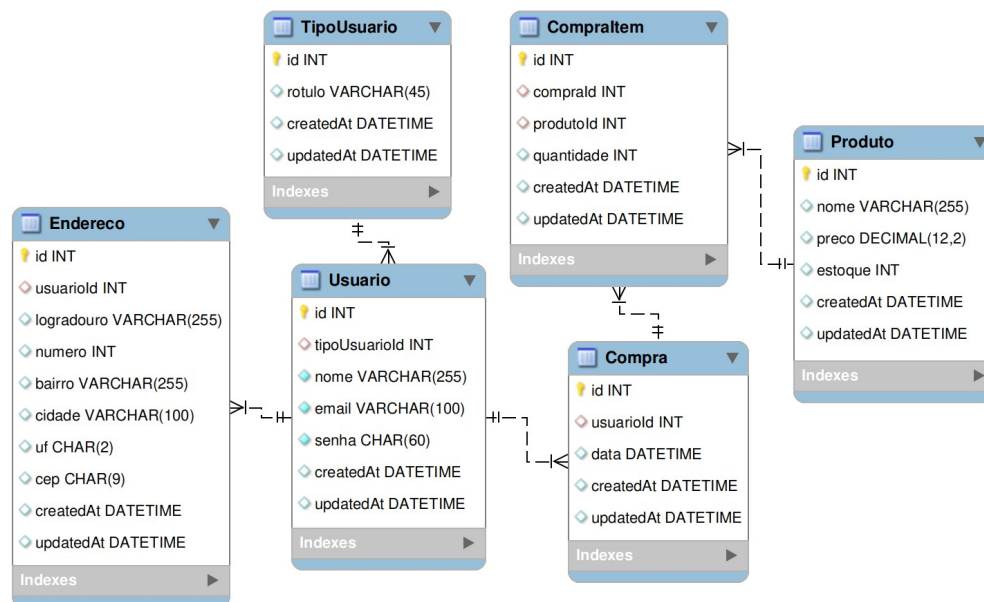


Figura 1: Esquema do banco de dados.

**Regra 08:** A página de produtos irá conter a listagem paginada de todos os produtos da loja virtual, onde cada página deverá mostrar um total de 10 produtos. Essa página deverá conter um mecanismo de busca, onde os usuários poderão digitar palavras chave para buscar pelos produtos desejados.

**Regra 09:** A aplicação deverá ter uma rota **/sobre**, que deverá conter informações sobre a loja virtual.

### Observações

O trabalho é individual, e não é permitido usar código desenvolvido por outros alunos da disciplina.

### Data e Forma de Entrega

A entrega do trabalho deverá ser feita até o dia **19 de novembro (sexta-feira)**. É importante que o repositório da disciplina contenha um diretório para o backend outro para o frontend.

O backend de sua aplicação deverá rodar na **porta 3020**, enquanto o frontend deverá rodar na **porta 3021**. Para instalar e corrigir sua aplicação, o professor irá seguir os seguintes passos no diretório backend da aplicação:

1. Executar o comando **docker-compose up**
2. Executar o comando **npm install**
3. Rodar as migrações e os seeders
4. Executar o comando **npm start**

Da mesma forma, o professor irá seguir os seguintes passos no diretório frontend da aplicação:

1. Executar o comando **npm install**
2. Executar o comando **npm start**

3. Abrir sua aplicação através do endereço **`http://localhost:3021`**