

Aluno: João Luiz de Oliveira Todescato

1 Nossa trabalho foi de criar uma biblioteca fictícia onde o admin cadastra e gere os livros e autores, enquanto o usuário comum apenas visualiza

2 Eu fiquei com a parte do front-end pelo Bruno já ter mais experiência no back, criando assim uma facilidade na hora de criar o código. Eu usei javascript e html, junto com um pouco de css, por ser aquilo que eu mais tinha segurança, pois foi ensinado na aula.

Eu usei de bases de códigos antigos para começar com a página de login onde eu usei bootstrap para agilizar o processo de criação para não ficar muito tempo embelezando o site o mesmo foi repetido para o início e inicioadm.

Depois foi criado o Javascript do site onde tudo interativo acontece nele, ele detém (o verificador de papel do usuário, guarda esse papel, o login e o register de usuários, a apresentação dos livros onde cada livro adicionado é posto na tela, e o crud de usuários e livros)

3 essa foi a parte do Bruno

4

Login e Register (handleLogin, handleRegister)

A parte mais difícil é onde o código interage com o backend para autenticação pois as funções handleLogin, handleRegister e handleLoginWithCredentials são bem complexas de se implementarem sem erro, tendo a capacidade de estragar bastante o código, mas deu tudo certo pois no HandleLogin gerenciando o envio do formulário de login (o mesmo de HandleRegister), que são enviados por uma requisição POST para a API. Se for bem-sucedido, armazena o usuário via setStoredUser() e **redireciona** para Inicio.html (USER) ou InicioAdm.html (ADMIN). Inclui tratamento de erros e manipulação de *spinner/botão* durante o carregamento. Também com o handleLoginWithCredentials(username, password) função auxiliadora que ajuda depois da realização do registro.

Funções CRUD de Administração

As funções que gerenciam Livros (saveBook, confirmDelete) e Usuários (saveUser, confirmDeleteUser) foram as mais complexas nessa parte por envolverem permissões e múltiplas rotas, o que me causou bastante confusão no começo.

Onde a função saveBook que gerencia o envio do formulário de livro. Verifica se há um bookID no campo oculto: se sim, faz um PUT, se não, faz um POST e Executa a validação de campos.

E também a função confirmDeleteUser que faz o envio da requisição DELETE para remover um usuário, que por algum motivo eu não estava conseguindo fazer funcionar por um bom tempo

5 O Bruno Cardoso ficou com a parte do BackEnd e eu fiquei com o FrontEnd