**Preparação para o sitema de Login**

**login.ejs**

<%- include ('partials/navbar.ejs') %>

<%- include ('partials/header.ejs') %>

<div *class*="container col-4">

    <div *class*="card mt-4">

        <div *class*="card-header">

            <h2>Login de administrador</h2>

        </div>

        <div *class*="card-body m-auto">

            <p *style*="color:red; font-size: 0.7em;">Você precisa estar logado para acessar os recursos do painel.</p>

            <form *action*="authenticate" *method*="post">

                <label *for*="email">E-mail:</label>

                <input *class*="form-control" *name*="email" *type*="email" *placeholder*="email@exemplo.com" *required*><br>

                <label *for*="password">Senha:</label>

                <input *class*="form-control" *name*="password" *type*="password" *placeholder*="Insira sua senha" *required*><br>

                <button *type*="submit" *class*="btn btn-success btn-block">Entrar</button>

            </form>

            <hr>

            <h6>Não possui conta? <a *href*="/cadastro">Cadastre-se</a></h6>

        </div>

    </div>

</div>

<%- include ('partials/footer.ejs') %>

Texto

Descrição gerada automaticamente com confiança média

**UserController.js**

*import* express *from* "express"

constrouter= *express.Router*()

// *ROTA DE LOGIN*

router*.get*("/login", (req, res) => {

    res*.render*("login")

  })

*export* *default* router

**index.js:**

*import* UsersController *from* "./controllers/UsersController.js"

app*.use*("/", ClientesController)

app*.use*("/", PedidosController)

app*.use*("/", ProdutosController)

app*.use*("/", UsersController)

**localhost:8080/login:**

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**cadastro.ejs:**

<%- include ('partials/navbar.ejs') %>

<%- include ('partials/header.ejs') %>

<div *class*="container">

    <div *class*="card mt-4">

        <div *class*="card-header">

            <h2>Cadastro de usuário</h2>

        </div>

        <div *class*="card-body">

            <form *action*="/createUser" *method*="post">

                <label *for*="email">E-mail:</label>

                <input *class*="form-control" *name*="email" *type*="email" *placeholder*="email@exemplo.com" *required*><br>

                <label *for*="password">Senha:</label>

                <input *class*="form-control" *name*="password" *type*="password" *placeholder*="Insira uma senha de no mínimo 8 caracteres" *required* ><br>

                <button *type*="submit" *class*="btn btn-success btn-block">Cadastrar</button>

            </form>

        </div>

    </div>

</div>

<%- include ('partials/footer.ejs') %>

**UserController.js**

// *ROTA DE CADASTRO DE USUÁRIO*

router*.get*("/cadastro", (req, res) => {

  res*.render*("cadastro")

})

**localhost:8080/cadastro:**

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

**User.js**

*import* Sequelize *from* "sequelize"

*import* connection *from* "../config/sequelize-config.js"

constUser=connection*.define*('users',{

*email*:{

*type*:Sequelize*.STRING*,

*allowNull*: *false*

},

*password*:{

*type*:Sequelize*.STRING*,

*allowNull*: *false*

}

})

User*.sync*({force: false})

*export* *default* User

**UserController.js**

*import* express *from* "express"

*import* User *from* "../models/User.js"

constrouter= *express.Router*()

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

**UserController.js**

// *ROTA DE CRIAÇÃO DE USUÁRIO NO BANCO*

router*.post*("/createUser", (req, res) => {

  // *COLETANDO INFORMAÇÕES DO CORPO DA REQUISIÇÃO*

  constemail=req*.body.email*

  constpassword=req*.body.password*

  User*.create*({

    email : email,

    password : password,

  })*.then*(() => {

    res*.redirect*("/login")

  })

})

Interface gráfica do usuário, Texto, Aplicativo, Teams

Descrição gerada automaticamente

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média