

ETEC BARTOLOMEU BUENO DA SILVA – ANHANGUERA
CURSO TÉCNICO EM INFORMÁTICA PARA INTERNET INTEGRADO AO
ENSINO MÉDIO

MIKE WILL BENTO DO REGO

DOCUMENTAÇÃO DE TELAS E CÓDIGOS

Santana de Parnaíba
2024

LISTA DE ILUSTRAÇÕES

Figura 1 – Index	6
Figura 2 – Cadastrar Usuário	7
Figura 3 – Login Usuário	8
Figura 4 – Biblioteca de Jogos	9
Figura 5 – Pesquisa Biblioteca	10
Figura 6 – Cadastrar Jogo.....	11
Figura 7 – Biblioteca de Jogos Admin	12
Figura 8 – Rota Adicionar Usuário	13
Figura 8 – Rota Adicionar Usuário	15
Figura 9 – Rota Atualizar Jogo	16
Figura 10 – Parâmetros Session	17

LISTA DE ABREVIATURAS E SIGLAS

RPGs Role-playing game

MMORPGs Massive Multiplayer Online Role-Playing Game

Id. Identification

SUMÁRIO

1. SOBRE O SITE	5
2. DOCUMENTAÇÃO DAS TELAS DA APLICAÇÃO	5
2.1. EXPLICAÇÃO INDEX	5
2.2. EXPLICAÇÃO CADASTRO USUÁRIO	6
2.3. EXPLICAÇÃO LOGIN USUÁRIO	7
2.4. EXPLICAÇÃO BIBLIOTECA DE JOGOS	8
2.5. EXPLICAÇÃO PESQUISA NA BIBLIOTECA	9
2.6. EXPLICAÇÃO CADASTRO JOGO	10
2.7. EXPLICAÇÃO BIBLIOTECA DE JOGOS ADMIN	11
2.8. EXPLICAÇÃO ALTERAR JOGO	12
3. SELEÇÃO DE 3 TELAS DE CÓDIGO	13
3.1. EXPLICAÇÃO ROTA PARA ADICIONAR USUÁRIO	13
3.2. EXPLICAÇÃO ROTA PARA ATUALIZAR O JOGO	15
3.3. EXPLICAÇÃO PARÂMETROS SESSION.....	17

1. SOBRE O SITE

O site é uma plataforma desenvolvida para armazenar conteúdo sobre RPGs e MMORPGs. Nele, os usuários podem cadastrar seus jogos favoritos. A proposta é reunir em um único espaço as informações mais completas sobre os melhores títulos do gênero, facilitando a busca por eles.

2. DOCUMENTAÇÃO DAS TELAS DA APLICAÇÃO

Nesta seção, apresentaremos uma visão geral das principais páginas do site, destacando suas funcionalidades e o que o usuário encontrará em cada uma delas. Seja para explorar recursos, interagir com o conteúdo ou navegar entre as opções disponíveis.

2.1. EXPLICAÇÃO INDEX

Na barra de navegação do site irá conter:

- A logo do site que se clicada redirecionara o usuário para o index do site.
- Um menu para poder navegar para outras páginas do site que o conteúdo varia dependendo se estiver logado ou não, se não estiver o conteúdo será “Cadastrar Usuário” que redireciona para a página onde será feito o cadastro do usuário, “Login Usuário” página onde o usuário faz o login e “Biblioteca de Jogos” onde ele pode acessar os jogos cadastrados no site, caso estiver logado o conteúdo será “Cadastrar Jogo” que manda para pagina de cadastrar os jogos, “Biblioteca de Jogos” onde poderá ver os jogos cadastrados e “Sair da Conta” que encerra a sessão do usuário.

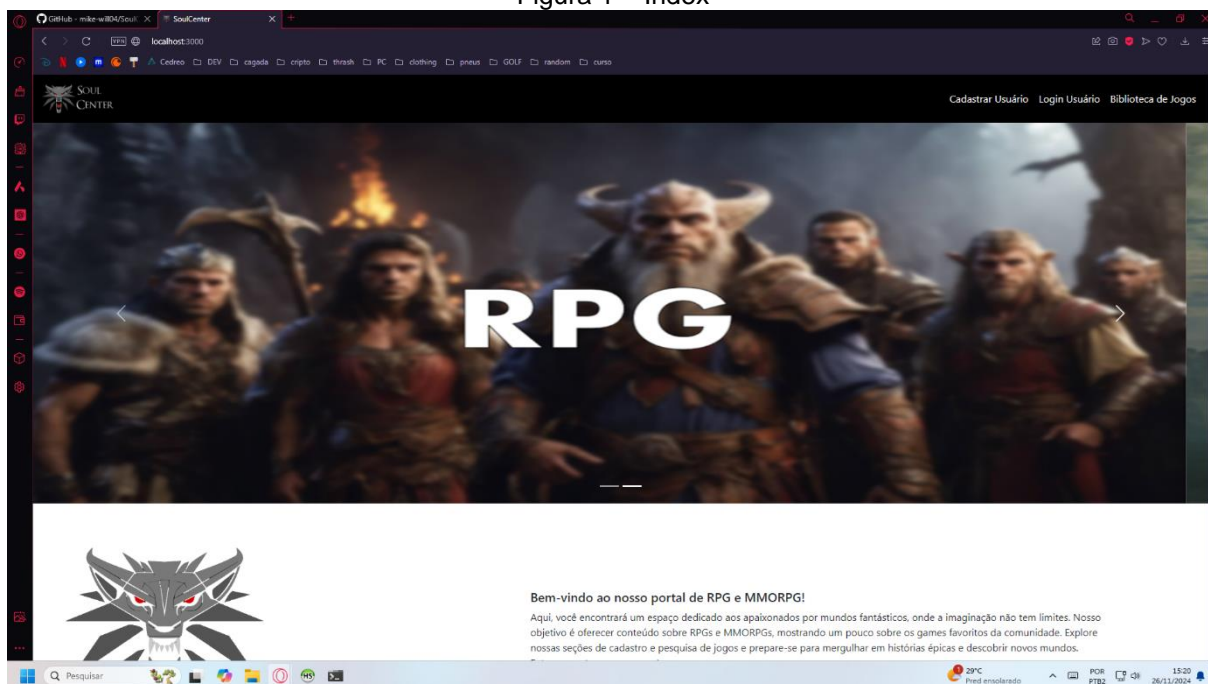
O carrossel irá transitar entre duas imagens uma para representar os jogos de RPGs e outra para os de MMORPGs. E nele contém botões para avançar ou voltar as imagens.

Logo abaixo do carrossel, há uma área com:

- Uma imagem decorativa da logo do site.
- Um texto de boas-vindas ao site.

E por fim o rodapé onde contem quem desenvolveu o site e os direitos.

Figura 1 – Index



Fonte: Próprio Autor

2.2. EXPLICAÇÃO CADASTRO USUÁRIO

No centro da página irá conter um formulário para que o usuário insira as informações para realizar o seu cadastro. E os campos presentes nesse formulário são:

- Usuário: Campo para inserir o nome ou um apelido.
- Email: Campo para o email.
- Senha: Campo para criar a senha.

E por fim um botão de cadastrar para poder mandar essas informações.

Figura 2 – Cadastrar Usuário

Cadastrar Usuário

Usuário:

Email:

Senha:

Cadastrar

Site desenvolvido por:
Arthur, Felipe, Gabriel e Mike Will
38
© 2024 SoulCenter - Todos direitos reservados.

Fonte: Próprio Autor

2.3. EXPLICAÇÃO LOGIN USUÁRIO

Nessa página contém um formulário onde o usuário irá inserir suas informações para realizar o login no site e os campos presentes são:

- Usuário ou email: Campo para o usuário digitar seu email ou usuário.
- Senha: Campo para digitar a senha.

Figura 3 – Login Usuário

Usuário ou Email:

Usuário ou Email

Senha:

Senha

Logar

Site desenvolvido por:
Arthur, Felipe, Gabriel e Mike Will
3B
© 2024 SoulCenter - Todos direitos reservados.

Fonte: Próprio Autor

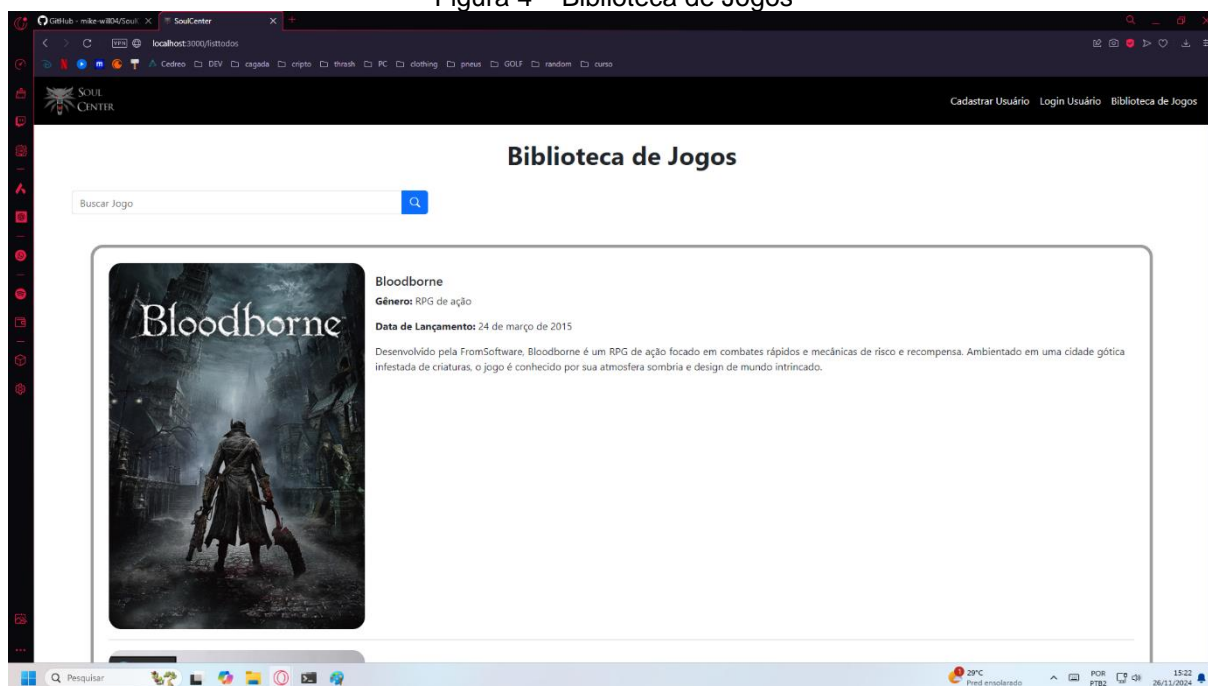
2.4. EXPLICAÇÃO BIBLIOTECA DE JOGOS

No topo da tela tem um formulário que permite o usuário pesquisar por um jogo em específico podendo ser pelo nome, gênero, data de lançamento ou um resumo.

Abaixo dele contem a lista de jogos cadastrados no site contendo certos detalhes como:

- Imagem do jogo.
- Nome do jogo.
- Gênero do jogo.
- Data de lançamento.
- Resumo.

Figura 4 – Biblioteca de Jogos



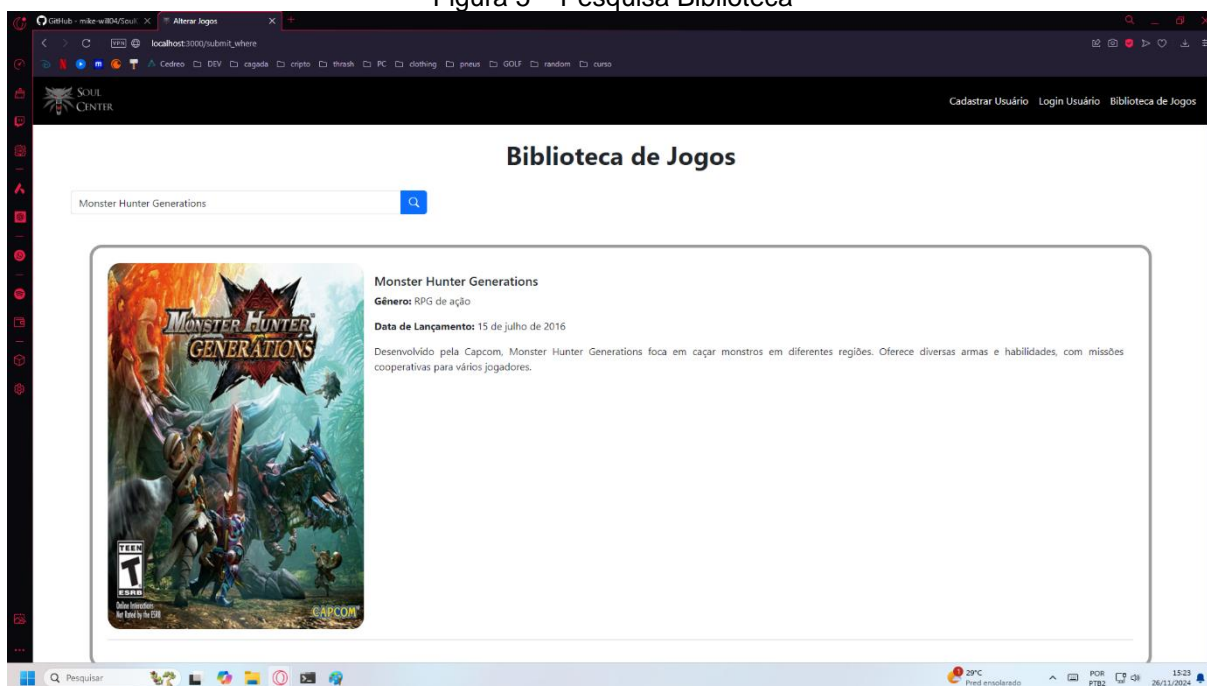
Fonte: Próprio Autor

2.5. EXPLICAÇÃO PESQUISA NA BIBLIOTECA

O processamento da busca ocorre quando o campo de pesquisa “buscar” é extraído pela requisição “req.body”, se o valor recebido do campo for “*”, a consulta vai retornar todos os jogos. Caso contrário a consulta vai ser filtrada por meio da ID, nome, gênero, data de lançamento ou o resumo.

A consulta do banco de dados é realizada de duas formas, para a busca geral é feito um “select * from jogo”, já para buscas específicas é adicionada a cláusula “where” junto com o valor do campo da pesquisa.

Figura 5 – Pesquisa Biblioteca



Fonte: Próprio Autor

2.6. EXPLICAÇÃO CADASTRO JOGO

A tela contém um formulário para realizar o cadastro de um novo jogo contendo os seguintes campos:

- Nome do jogo: Campo obrigatório para cadastrar o nome do jogo.
- Gênero: Campo obrigatório para cadastrar o gênero.
- Data de Lançamento: Campo obrigatório para o lançamento do jogo.
- Resumo: Campo obrigatório para uma curta descrição do jogo.
- Imagem: Campo obrigatório para carregar uma imagem que represente o jogo.

No fim do formulário contém um botão para realizar o cadastro. E o formulário utiliza o método "Post" para enviar as informações e também a funcionalidade "enctype='multipart/form-data'" para permitir o upload das imagens.

Figura 6 – Cadastrar Jogo

The screenshot shows a web browser window with the URL 'localhost:3000/cadastro'. The page title is 'Cadastrar Jogo'. The form contains the following fields:

- Nome:
- Gênero:
- Data de Lançamento:
- Resumo:
- Imagem: Escolher arquivo Nenhum arquivo escolhido

A blue button labeled 'Cadastrar' is located at the bottom right of the form. The footer of the page reads: 'Site desenvolvido por: Arthur, Felipe, Gabriel e Mike Will', '38', and '© 2024 SoulCenter - Todos direitos reservados.'

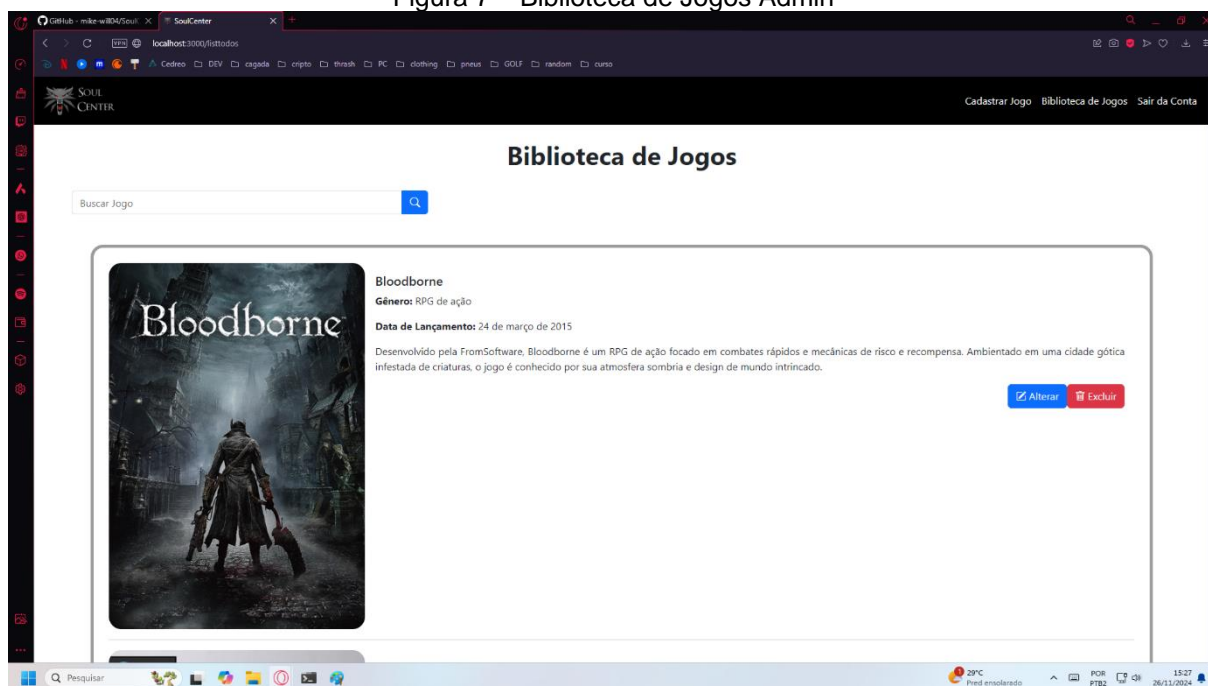
Fonte: Próprio Autor

2.7. EXPLICAÇÃO BIBLIOTECA DE JOGOS ADMIN

Diferentemente da página da biblioteca para os usuários, essa página é exclusiva para os administradores do site, que possuem algumas funções exclusivas:

- **Alterar:** Por meio do botão 'alterar' localizado abaixo do resumo de cada jogo, o administrador consegue alterar os jogos cadastrados, corrigindo possíveis erros de digitação presentes nos textos.
- **Excluir:** Esse botão permite que o administrador exclua qualquer jogo cadastrado no site.

Figura 7 – Biblioteca de Jogos Admin



Fonte: Próprio Autor

2.8. EXPLICAÇÃO ALTERAR JOGO

Essa pagina serve para o administrador conseguir alterar as informações cadastradas, para isso o formulário retorna as informações do jogo escolhido sendo necessário apenas alterar o conteúdo do campo e clicar no botão “Alterar” para atualizar as informações.

Figura 8 – Rota Adicionar Usuário

Alterar o Jogo

Nome:
The Legend of Zelda: Breath of the Wild

Gênero:
RPG de ação/Aventura

Data de Lançamento:
3 de março de 2017

Resumo:
Desenvolvido pela Nintendo, é um dos jogos mais revolucionários da série, oferecendo um vasto mu

Imagem:
Escolher arquivo Nenhum arquivo escolhido

Alterar

Site desenvolvido por:
Arthur, Felipe, Gabriel e Mike Will

Fonte: Próprio Autor

3. SELEÇÃO DE 3 TELAS DE CÓDIGO

Nesta seção, será explorado o funcionamento de partes específicas do projeto, com a apresentação de trechos de código e a explicação de suas funcionalidades. Cada exemplo será analisado para demonstrar como as partes do sistema se integram, contribuindo para uma experiência fluida e interativa para o usuário.

3.1. EXPLICAÇÃO ROTA PARA ADICIONAR USUÁRIO

Captura dos Dados do Formulário

- O código desestrutura os dados enviados na solicitação do corpo da requisição.

Verificação de Usuário e Email Existentes

- Consulta SQL: Antes de criar um novo usuário, a aplicação verifica se o nome de usuário ou o email já estão cadastrados no banco de dados.
- “checkSql”: A consulta busca no banco de dados um usuário ou email que seja igual ao que foi fornecido pelo formulário.
- Se algum usuário ou email já estiver registrado, o código retornará um alerta informando ao usuário que a informação já foi registrada, evitando duplicação.

Exibição de Erro se Já Existir

- Verificação de Resultados: Se a consulta retornar algum registro, isso significa que o usuário ou email já existe no banco de dados.
- Um alerta é exibido informando ao usuário que ele deve tentar novamente com dados diferentes.

Criptografando a Senha

- Criptografia com bcrypt: A senha fornecida pelo usuário é criptografada usando o pacote “bcrypt” com um salt de 10.
- O hash gerado é armazenado no banco de dados, garantindo que a senha original nunca seja salva em texto simples.

Inserção do Novo Usuário

- Inserção no Banco de Dados: Se o nome de usuário e o email não estiverem registrados, a aplicação procede com a inserção de um novo registro de usuário no banco de dados.
 - “usuário”: Nome de usuário.
 - “email”: Endereço de email.
 - “hashedPassword”: Senha criptografada.
 - '0': O campo “admin” é definido como '0', indicando que o usuário não é um administrador.

Exibição de Erro ou Sucesso

- Erro ao Inserir: Se ocorrer um erro durante a inserção do usuário, uma mensagem de erro será exibida.
- Sucesso: Se a inserção for bem-sucedida, um alerta informará ao usuário que o cadastro foi concluído com sucesso e, em seguida, o usuário será redirecionado para a página de login.

Tratamento de Erro Interno

- Erro Genérico: Se algum erro ocorrer fora da consulta, ele será capturado pelo bloco “catch”, e uma mensagem de erro genérica será enviada ao usuário.

Figura 8 – Rota Adicionar Usuário

```

139 app.post('/usuario_novo', async (req, res) => {
140   const { usuario, email, senha } = req.body;
141
142   // Consulta para verificar se já existe o usuário no email
143   const checkSql = "SELECT * FROM usuario WHERE usuario = ? OR email = ?";
144   const query = db.query(checkSql, [usuario, email]);
145   const [err, results] = await query.promise();
146   if (err) {
147     return res.send({
148       status: 500,
149       message: "Erro ao verificar usuário: " + err.message
150     });
151   }
152   // Se existir, retorna um alerta
153   if (results.length > 0) {
154     return res.send({
155       status: 400,
156       message: "Usuário ou email já cadastrado. Tente novamente."
157     });
158   }
159   // Criptografar a senha antes de salvar no banco
160   const hashedPassword = await bcrypt.hash(senha, 10); // Gere o hash da senha com 10 saltos
161
162   // Insere o novo usuário
163   const sql = "INSERT INTO usuario (usuario, email, senha, senha) VALUES (?, ?, ?, ?)";
164   const query = db.query(sql, [usuario, email, senha, hashedPassword]);
165   const [err, results] = await query.promise();
166   if (err) {
167     return res.send({
168       status: 500,
169       message: "Erro ao cadastrar usuário: " + err.message
170     });
171   }
172   // Redireciona para a página de login após o alerta
173   res.send({
174     status: 200,
175     message: "Cadastro realizado com sucesso!"
176   });
177   window.location.href = "/login"; // Redireciona para a página de login
178 }
179
180 // Erro ao processar a solicitação
181 catch (err) {
182   console.error("Erro ao processar a solicitação:", err);
183   res.send({
184     status: 500,
185     message: "Erro interno no servidor. Tente novamente mais tarde."
186   });
187 }
188 }
189
190 // Tratamento de erros
191 app.use((err, req, res, next) => {
192   console.error(err.stack);
193   res.status(500).send("Erro interno no servidor");
194 });

```

Fonte: Próprio Autor

3.2. EXPLICAÇÃO ROTA PARA ATUALIZAR O JOGO

Rota e Middleware de Upload

- Rota: A rota “/submit_update” é acionada quando um formulário de atualização é enviado.
- Middleware “upload.single('imagem')”:
 - Garante que o arquivo enviado seja tratado e salvo no servidor antes que a lógica principal da rota seja executada.
 - O arquivo fica acessível como “req.file” dentro da rota.

Desestruturação e Verificação de Dados

- “req.body”: Os campos enviados no formulário são desestruturados para facilitar o uso.
- “req.file”: Verifica se um arquivo foi enviado:
 - Se um arquivo foi enviado, seu nome é armazenado em “image”.
 - Se não, “image” será “null”.

Construção da Query SQL

- Query inicial: Atualiza os campos básicos do jogo (nome, genero, lancamento, resumo).
- Parâmetros: Os valores para substituir os “?” na query são armazenados no array “params”.

Verificação e Inclusão de Imagem

- Se uma nova imagem foi enviada, o campo “imagem” também é atualizado.
- A query e os parâmetros são ajustados para incluir a nova imagem.

Finalização da Query

- Especifica o jogo a ser atualizado com base no “id”.
- O “id” é adicionado à lista de parâmetros.

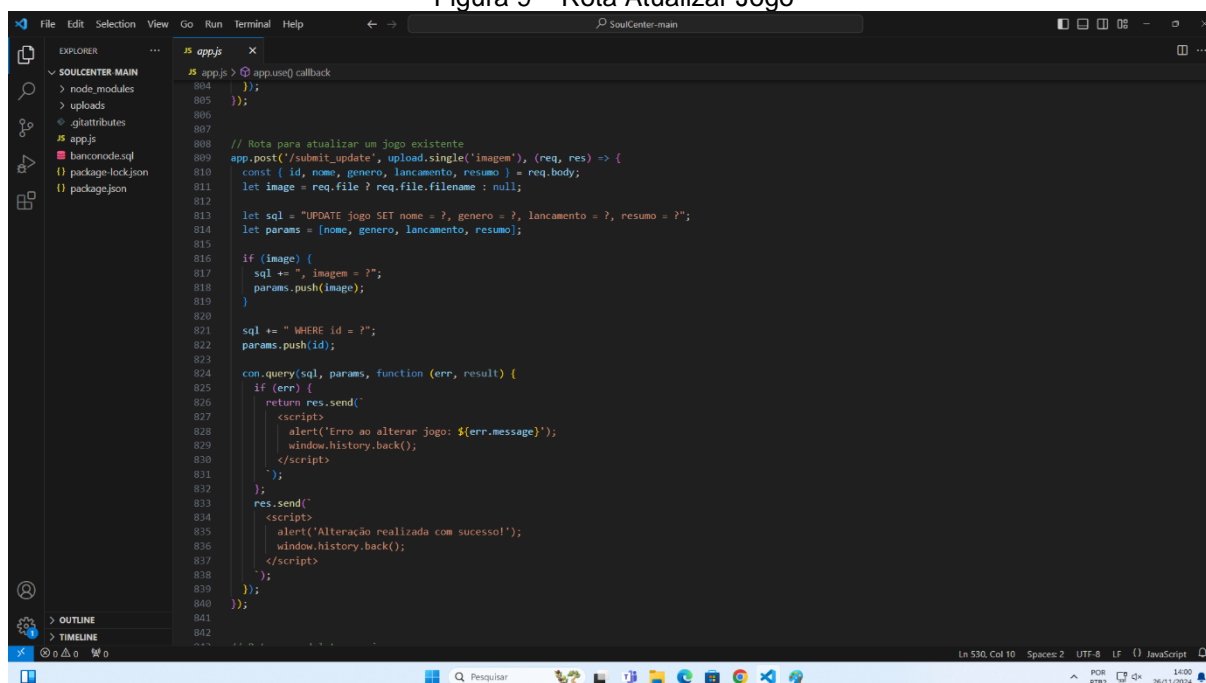
Execução da Query no Banco de Dados

- “con.query”: Executa a query com os valores definidos em “params”.

Feedback ao Usuário

- Após a atualização, o usuário recebe um alerta no navegador e o navegador volta à página anterior.

Figura 9 – Rota Atualizar Jogo



Fonte: Próprio Autor

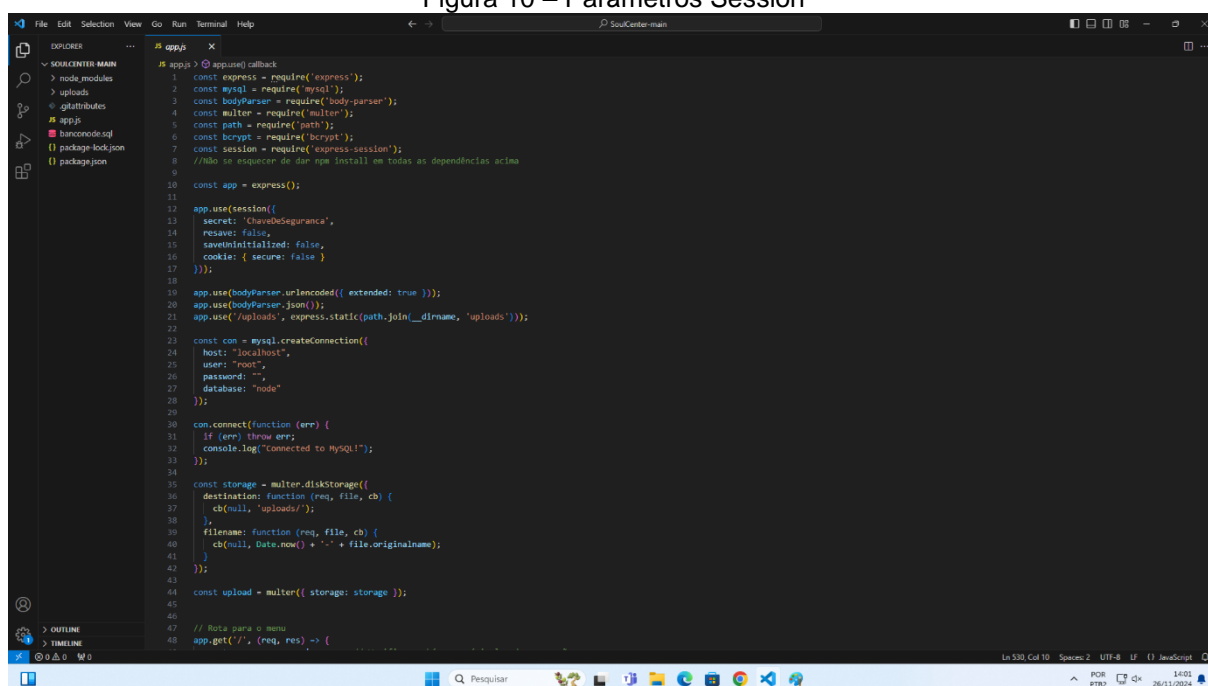
3.3. EXPLICAÇÃO PARÂMETROS SESSION

O “app.use” aplica o middleware de sessão ao aplicativo Express. Isso faz com que todas as requisições que passem pelo servidor tenham suporte a sessões.

Configurações da Sessão:

- secret: 'ChaveDeSeguranca'
 - Define uma chave secreta usada para assinar o ID da sessão armazenado no cookie do cliente.
- resave: false
 - Define se a sessão será salva novamente no armazenamento de sessão, mesmo que não tenha sido modificada durante a requisição.
- saveUninitialized: false
 - Determina se uma nova sessão deve ser criada para cada requisição, mesmo que ela não tenha sido modificada ou utilizada.
- cookie: { secure: false }
 - secure: false: Indica que o cookie da sessão não exige HTTPS. Isso é útil em desenvolvimento, mas em produção deve ser configurado como true para segurança adicional.

Figura 10 – Parâmetros Session



Fonte: Próprio Autor