



TRABALHO
INTERDISCIPLINAR:
BACKEND

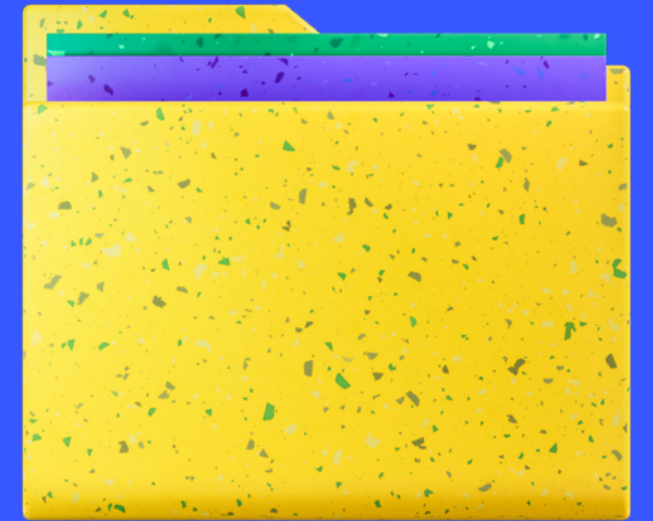
Code Wizard

Grupo: Guilherme Vieira, Henrique
Fialho, Henrique Moura, Larissa Cássia
de Miranda



Visão Geral

- Muitos estudantes enfrentam dificuldades em encontrar material didático adequado às suas necessidades específicas de aprendizado



OBJETIVOS

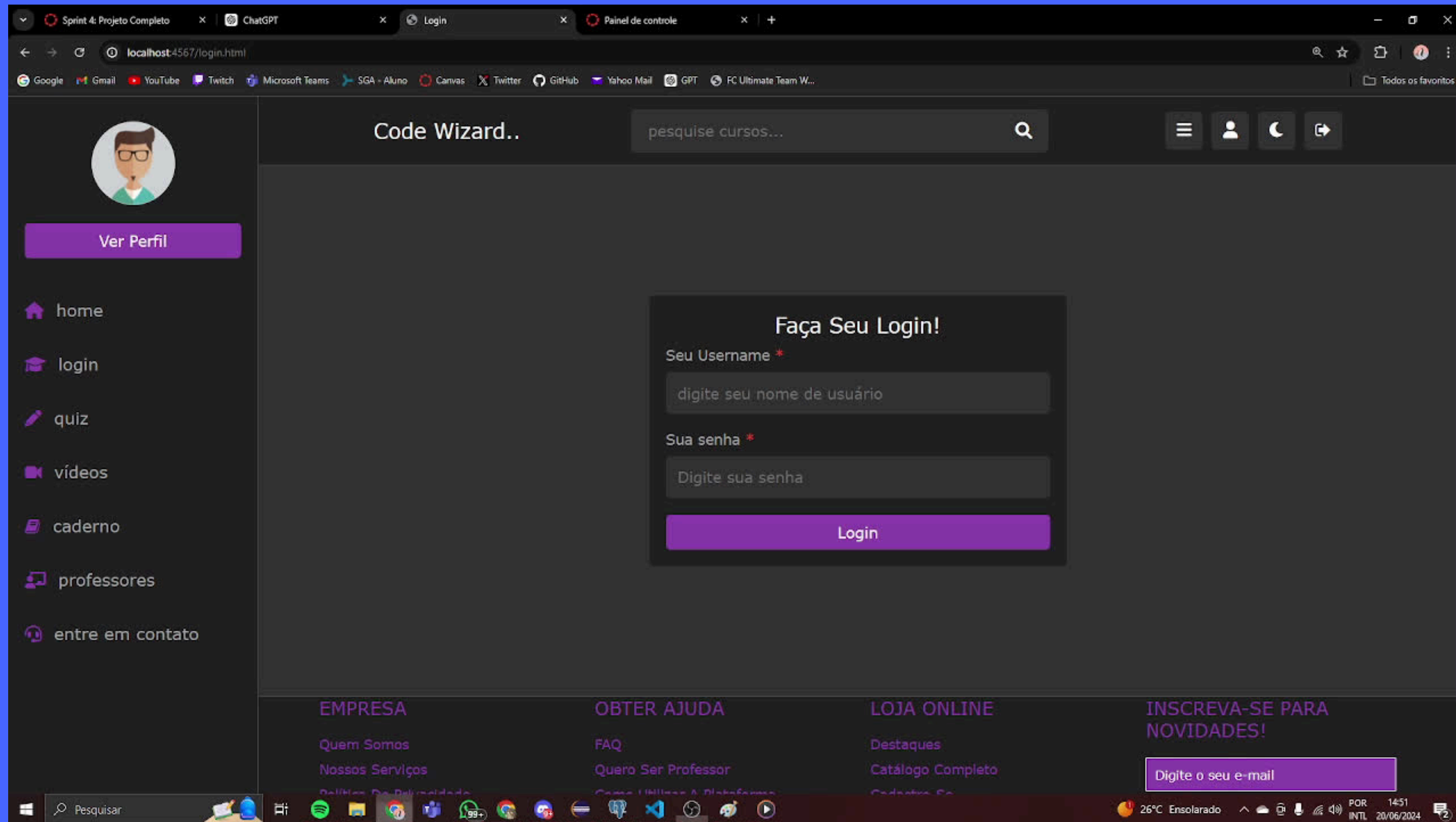
- O objetivo é fornecer uma plataforma interativa e adaptativa, que ofereça recursos educacionais de alta qualidade em programação

METAS

- Nessa Sprint a meta, que foi cumprida, era entregar o backend totalmente funcional, com o caminho de dados de acordo com o orientado pelos professores



A aplicação



Tecnologias utilizadas:



Maven



Criptografia

Ao criar o usuário criptografamos a senha por meio do MD5

```
public static String toMD5(String senha) throws Exception {  
    MessageDigest m=MessageDigest.getInstance("MD5");  
    m.update(senha.getBytes(),0, senha.length());  
    return new BigInteger(1,m.digest()).toString(16);  
}
```

O caminho dos dados

1 -Usuário clica em cadastrar pergunta

Cadastre Uma Nova Pergunta

Conteúdo da Pergunta *

ID do Vídeo Associado *

Nível da Pergunta *

Selecione um nível

Cadastrar Pergunta



2- O form chama o método post /submit-question

```
<section class="form-container">
  <!-- Formulário modificado para cadastro de perguntas -->
  <form action="/submit-question" method="post">
    <h3>Cadastre uma nova pergunta</h3>

    <p>Conteúdo da Pergunta <span>*</span></p>
    <input type="text" name="content" placeholder="Digite o conteúdo da pergunta" />

    <p>ID do Vídeo Associado <span>*</span></p>
    <input type="number" name="videoID" placeholder="Digite o ID do vídeo relacionado" />

    <p>Nível da Pergunta <span>*</span></p>
    <select name="level" required class="box">
      <option value="">Selecione um nível</option>
      <option value="easy">Fácil</option>
      <option value="medium">Médio</option>
      <option value="hard">Difícil</option>
    </select>

    <input type="submit" value="Cadastrar Pergunta" name="submit" />
  </form>
```



O caminho dos dados

3 - Na aplicação.java se encontra esse post, que chama uma função encontrada em questionService

```
public class app {  
  
    public static void main(String[] args) {  
        port(4567);  
        staticFiles.location("/public");  
  
        before((request, response) -> {  
            response.header("Access-Control-Allow-Origin", "*");  
            response.header("Access-Control-Allow-Methods", "GET, POST");  
            response.header("Access-Control-Allow-Headers", "Content-Type, Authorization");  
            // Rotas que não requerem autenticação  
            List<String> publicRoutes = Arrays.asList("/login", "/login.html", "/register-u  
  
            if (!publicRoutes.contains(request.pathInfo())) {  
                if (request.session().attribute("user") == null) {  
                    response.redirect("/login.html");  
                    halt();  
                }  
            }  
        });  
  
        questionService questionService = new questionService();  
  
        get("/questions", (req, res) -> questionService.getAllQuestions(req, res));  
        post("/submit-answers", (req, res) -> questionService.submitAnswers(req, res));  
        post("/submit-question", (req, res) -> questionService.createQuestion(req, res));  
        get("/get-questions", (req, res) -> questionService.getQuestions(req, res));
```

4- A função createQuestion chama o método insert da DAO

```
public String createQuestion(Request req, Response res) {  
    String content = req.queryParams("content");  
    int videoID = Integer.parseInt(req.queryParams("videoID"));  
    String level = req.queryParams("level");  
  
    QUESTION newQuestion = new QUESTION(-1, content, videoID, level);  
    questionDao.insert(newQuestion);  
  
    res.type("text/html");  
    return "<script>alert('Pergunta cadastrada com sucesso!'); window  
}
```



O caminho dos dados

5- Dao se conecta ao banco de dados, insere na tabela e retorna para createQuestion

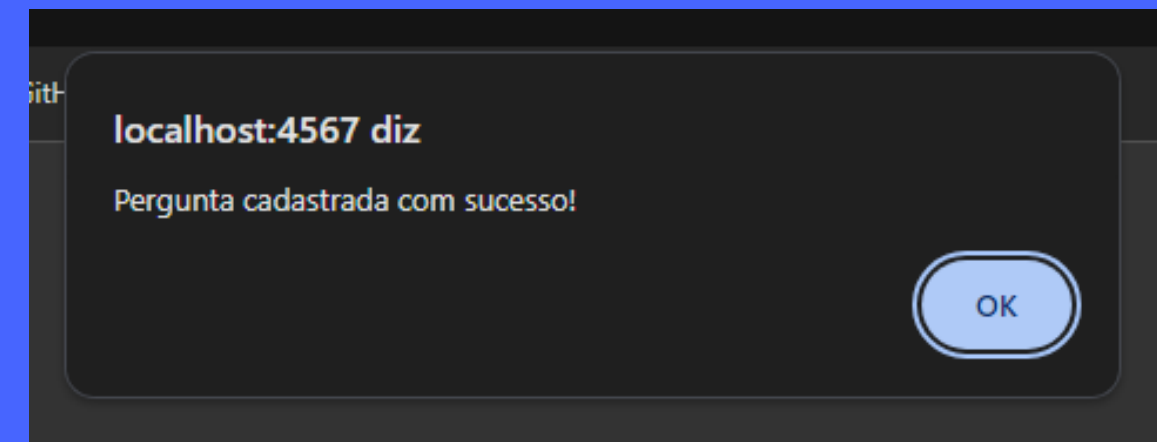
```
public boolean insert(QUESTION question) {  
    boolean status = false;  
    try {  
        PreparedStatement st = conexao.prepareStatement("insert into perguntas (content, video_id, level) values (?, ?, ?)");  
        st.setString(1, question.getContent());  
        st.setInt(2, question.getVideoID());  
        st.setString(3, question.getLevel());  
        st.executeUpdate();  
        st.close();  
        status = true;  
    } catch (SQLException u) {  
        throw new RuntimeException(u);  
    }  
    return status;  
}
```



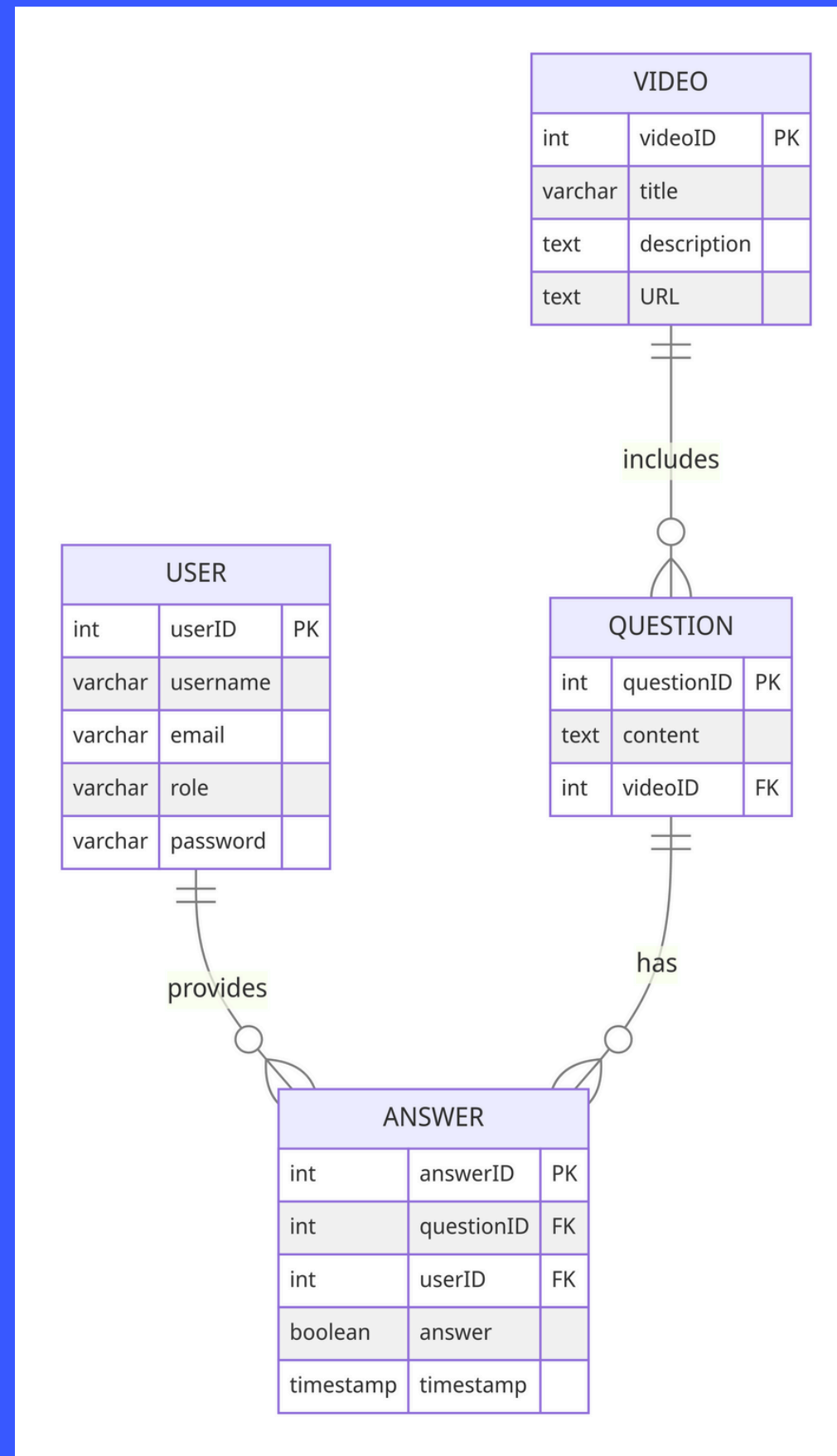
6- A função createQuestion retorna para aplicação.java que retorna ao front a mensagem de êxito

```
res.type("text/html");  
return "<script>alert('Pergunta cadastrada com sucesso!')";
```

7- Front-End exibe a mensagem, sinalizando que a troca entre front, back e BD foi bem sucedida



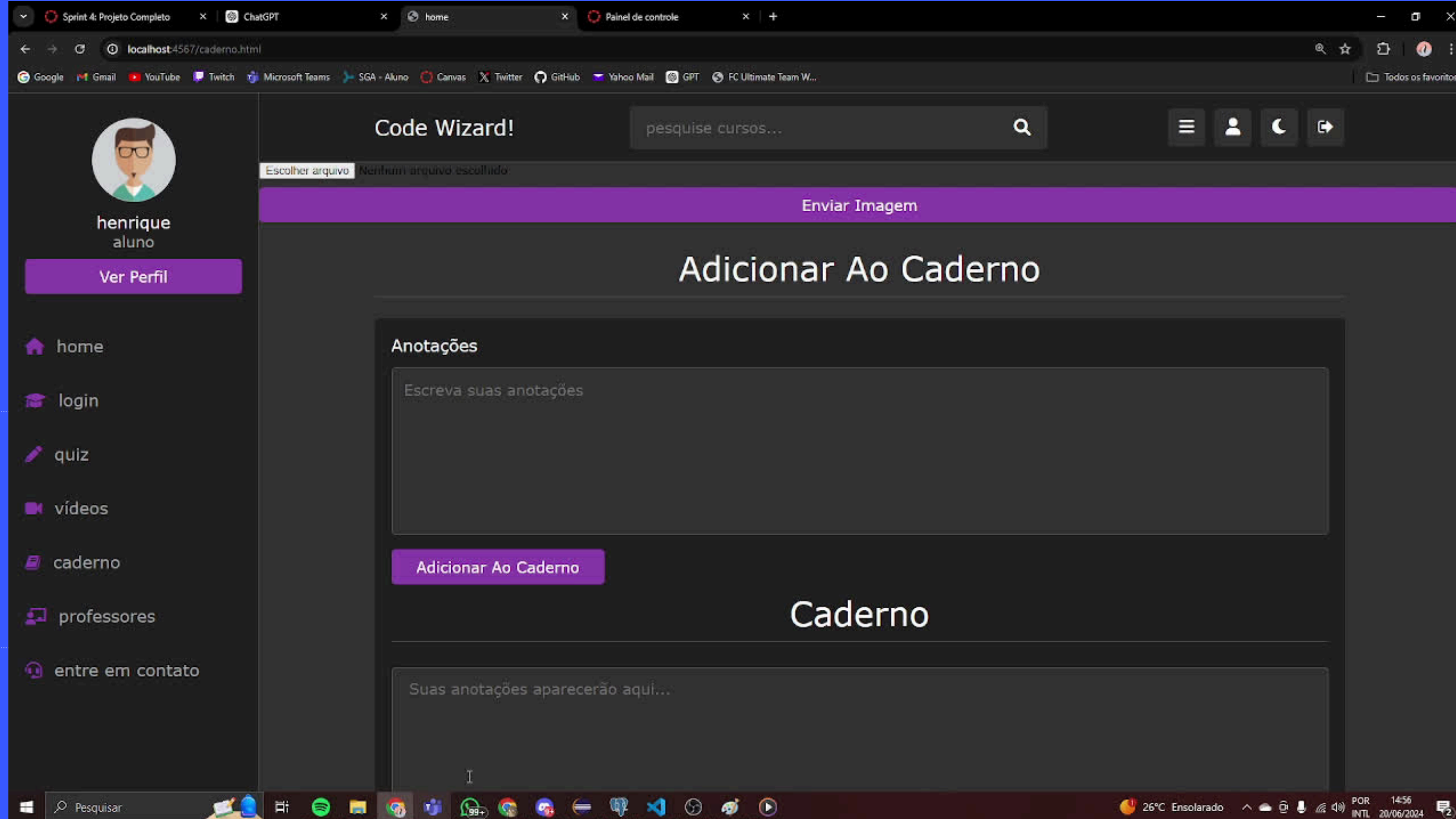
Banco de Dados



IS Canvas

Ferramental de IA <ul style="list-style-type: none">• Técnicas aplicadas: Reconhecimento Óptico de Caracteres (OCR).• Softwares e recursos: Azure Computer Vision para OCR; Java para <u>scripting</u>; SDK do Azure para integração.• Integração de sistemas: Integração com bancos de dados para armazenamento de texto.	Entradas <ul style="list-style-type: none">• Dados disponíveis: Imagens de páginas de caderno capturadas por câmera ou scanner.• Dados efetivamente utilizados: Imagens claras e legíveis dos textos escritos à mão no caderno.	Proposição de valor <p>Valor para o cliente: Facilitar o estudo e a organização para pessoas que aprendem melhor com escrita física, permitindo fácil revisão e acesso digital ao conteúdo escrito à mão. E permitindo um compartilhamento com os outros usuários da plataforma</p>	Equipe <p>Membros do grupo 01</p>	Clientes <p>Clientes internos e externos: Estudantes, profissionais que utilizam muita escrita manual, instituições de ensino.</p>
	Saídas <ul style="list-style-type: none">• Saídas esperadas: Texto digitalizado formatado a partir das imagens capturadas.• Métricas a serem otimizadas: Precisão do texto reconhecido, velocidade de processamento.		Stakeholders Chaves <p>Tomadores de decisão: Gestores de produto em empresas de tecnologia educacional, diretores de instituições de ensino que adotam ferramentas tecnológicas.</p>	
Custos <ul style="list-style-type: none">• Custos envolvidos: Licenciamento do Azure Computer Vision, desenvolvimento e manutenção do software, marketing.• Custos fixos e variáveis: Custos fixos incluem licenciamento e salários; custos variáveis podem incluir escalonamento de servidor conforme o número de usuários.			Receitas <ul style="list-style-type: none">• Novas receitas?: Assinaturas do serviço de caderno digital, vendas de aplicativos integrados.• Reduzir custos internos?: Redução de custos com material didático físico em instituições que adotam a tecnologia.	

Especificação dos Sistemas Inteligentes



Foi utilizado o azure computer vision, para fazer reconhecimento de texto

[Voltar ao índice](#)



Requisitos

<i>IDENTIFICADOR</i>	<i>REQUISITO</i>	<i>CLASSIFICAÇÃO</i>
RF01	Login: Usuário consegue se cadastrar como aluno e como professor	Essencial
RF02	Usabilidade Aluno: Aluno consegue fazer o quiz e assistir os vídeos	Essencial
RF03	Usabilidade Professor: Professor consegue fazer upload de vídeos e perguntas	Essencial
RF04	Comentários: Aluno consegue fazer comentários nos vídeos	Importante
RF05	Caderno: Sistema inteligente implementado para leitura de texto	Importante

Todos os requisitos funcionais foram atingidos!

Links Importantes



Repositório: <https://github.com/ICEI-PUC-Minas-CC-TI/plmg-cc-ti2-2024-1-g01-codewizard>



Vídeo Completo: <https://www.youtube.com/watch?v=5qeGQGdQJi8>
Vídeo do Slide 3: <https://www.youtube.com/watch?v=LGbpVEb9-n8>
Vídeo do Slide 10: <https://www.youtube.com/watch?v=60PHyS3ifok>

Obrigado!



[Voltar ao índice](#)

