



**UNIVERSIDADE COMUNITÁRIA DA REGIÃO DE CHAPECÓ –
UNOCHAPECÓ**

**CIÊNCIA DA COMPUTAÇÃO 2025.2
DESENVOLVIMENTO WEB**

**ISABELLA CHISTÉ CRAMER, LUCAS GUERRA RODRIGUES, NICOLE
SCHWARZ, RUDIVAN ZONTA DE OLIVEIRA**

TRABALHO FINAL – DESENVOLVIMENTO WEB

CHAPECÓ / SC

2025

1. FASE DE PLANEJAMENTO

Repositório Github: <https://github.com/Rudivan060/desenvolvimento-web>.

Empresa fictícia: clínica médica.

Objetivo: permitir que pacientes visualizem informações da instituição, acessem especialidades e médicos disponíveis para agendamento de consultas e conheçam mais sobre a empresa.

Páginas pretendidas: menu inicial, médicos, consultas, agenda, cadastro de profissionais (com CRUD), contato, sobre nós.

Estilos: modelo de plataforma semelhante aos de atendimento e agendamento de clínicas médicas reais.

2. DESENVOLVIMENTO DO FRONT-END

O Front-End do projeto foi desenvolvido utilizando .HTML, .CSS e JavaScript, garantindo interface intuitiva e interativa.

2.1. Páginas HTML

- **index.html:** página inicial da clínica, contendo apresentação geral;
- **medicos.html:** lista os médicos cadastrados e suas especialidades;
- **consultas.html:** interface para visualizar consultas e horários agendados;
- **agenda.html:** mostra o calendário e agenda de horários disponíveis por médicos;
- **cadastro-profissional.html:** formulário para cadastro de novos profissionais (CRUD);
- **contato.html:** página com formulários de contato e informações;
- **sobre-nos.html:** apresentação institucional da clínica e informações gerais.

2.2. CSS (style.css)

- Responsável pela estilização de todas as páginas;
- Define cores, fontes, layouts, grids de todos os elementos;

2.3. JavaScript (script.js)

- Interações com a interface, como manipulação do DOM, validação de formulários, envio de dados ao Back-End, renderização dinâmica das páginas de médicos e agenda.

3. DESENVOLVIMENTO DO BACK-END

3.1. Arquitetura da Aplicação

O Back-End foi desenvolvido seguindo a arquitetura em camadas SCR (Service, Controller, Repository), com o objetivo de garantir melhor organização, segurança e facilidade de manutenção.

3.2. Repository

Responsável pelo acesso e execução de consultas ao banco de dados, realizando apenas operações essenciais, como selecionar, inserir, atualizar e remover registros.

3.3. Service

Concentra toda a lógica e regras de negócio da aplicação, como validação dos dados recebidos, regras de integridade e processamento antes de enviar para o Repository. Ou seja, atua como intermediário entre o Controller e Repository.

3.4. Controller

Recebe e responde às requisições HTTP enviadas pelo cliente. Interpreta os dados recebidos, chama os métodos adequados do Service, retorna respostas no formato Json e controla códigos de status HTTP. Funciona como intermediário entre Service e Front-end.

3.5. Middlewares

Foram projetados de maneira simples, atuando como registro de histórico de requisições, funções auxiliares para padronização.

3.6. Reatividade

Capacidade de refletir mudanças no sistema sem a necessidade de recarregar a página inteira a partir do retorno do back-end. Assim o front-end consegue atualizar listas, exibir novos registros e modificar elementos da interface.

4. PRINCIPAIS DESAFIOS

Um dos principais desafios foi a organização e integração de todas as estruturas do Front-End com o Back-End, pois houveram diversas complicações ao longo do desenvolvimento do projeto, levando em consideração que deveria ser simples e compreensível para todos os integrantes do grupo.

Além disso, as interações do terminal com o repositório da ferramenta Github também proporcionou algumas dificuldades e aprendizados, solucionando problemas de commits, clonagem, recebimento das alterações de cada integrante, dentre outros.

5. DIVISÃO DE TAREFAS

INTEGRANTE	TAREFAS REALIZADAS
Isabella Chisté Cramer	Criação e edição do Front-End, ou seja: elaboração das páginas .html, definição de estilos e responsividade através do arquivo style.css, interações do arquivo script.js.
Lucas Guerra Rodrigues	Criação do Back-End, ou seja: definição de rotas da API, requisições, repositórios, interações através do arquivo services e aplicação do CRUD (criar, ler, editar e deletar).
Nicole Schwarz	Ajustes nas telas .html e estilos específicos através do arquivo style.css.
Rudivan Zonta de Oliveira	Ajustes no Front-End e Back-End, portanto houve colaboração em: melhoria de rotas da API, interações do arquivo script.js, melhoria de páginas .html.

6. REFLEXÕES POR INTEGRANTE

INTEGRANTE	VISÃO ACERCA DO TRABALHO
Isabella Chisté Cramer	<p>“Trabalhar em grupo trouxe alguns desafios, especialmente porque nem todos estavam disponíveis nos mesmos horários. Além disso, cada um tinha sua própria maneira de estruturar e programar, gerando diferenças nas soluções e nos caminhos escolhidos.</p> <p>Tivemos várias dificuldades e ajustes ao longo do trabalho, para organização de uma dinâmica mais acessível que todos pudessem compreender e participar.</p> <p>Por outro lado, essa diversidade trouxe pontos positivos. Esse processo ajudou a desenvolver mais paciência, comunicação e flexibilidade. No fim, atingimos os resultados de forma satisfatória, mesmo passando por diversos obstáculos.”</p>
Lucas Guerra Rodrigues	<p>“Trabalhar em grupo tem seus desafios, até pela questão de disponibilidade, experiência e desempenho de cada integrante.</p> <p>Tivemos vários ajustes extras conforme avançávamos no projeto, houveram problemas de upload de alterações através do Github (clonagem, commits, dentre outros), estruturação de arquivos e divisão de tarefas.</p> <p>Mesmo com dificuldades, atingimos os resultados desejados.”</p>
Nicole Schwarz	<p>“No decorrer do trabalho, percebi que muitas vezes é necessário mudar o que foi planejado, mas não perdendo a "essência", tivemos trocas de layout, modelo, ideias etc.</p> <p>Como nos outros momentos, algumas partes empacaram, sendo necessário solicitar auxílio da IA para melhor compreensão dos componentes.</p>

	<p>No geral, o trabalho ajudou a firmar o conteúdo aprendido em aula, desenvolver mais autonomia nas decisões, saber que é possível errar e refazer a parte em questão e compreender que um bom resultado depende tanto da técnica quanto da capacidade de adaptar o código quando necessário.”</p>
Rudivan Zonta de Oliveira	<p>“Durante esse trabalho, ficou evidente a diferença das maneiras de programar de cada um. Se um time não tiver coesão, a probabilidade de fazer atualizações no código que funcione apenas para você é muito alta.</p> <p>Tivemos vários problemas como esse ao decorrer do trabalho, isso nos gerou muita dificuldade e estresse até conseguirmos alinhar um modo de trabalho para que todos possam contribuir de maneira mais assertiva.</p> <p>A partir disso conseguimos chegar nos resultados que desejávamos, e que, no nosso ponto de vista atendeu as expectativas criadas.”</p>