

Certificadora de Competência Comum

Feito por Bruno, Gabriel, Murilo, Rafael

Projeto Front-End

Visão geral

Produzir materiais de aprendizagem digitais e gratuitos voltados à disciplina de Front-end, oferecendo uma experiência de estudo simples, atrativa e de fácil acesso.

Objetivos

1. **Desenvolvimento de aplicações Web:** tecnologias fundamentais e arquitetura clienteservidor;
2. **HTML e CSS:** Modelar estruturas de aplicações web com HTML e CSS, de forma autorregulada;
3. **Design Responsivo:** CSS 3.0, Media-querie: Projetar layout de aplicações web com Design Responsivo e autorregulação. Além disso, implementar aplicações web de acordo com padrões de codificação, regras de design responsivo, acessibilidade na web e otimização de motores de busca, para diferentes plataformas.
4. **Padronização, acessibilidade na web e Search-Engine-Optimization (SEO):** Implementar aplicações web de acordo com padrões de codificação, regras de design responsivo, acessibilidade na web e otimização de motores de busca, para diferentes plataformas;

5. **Linguagens de scripting:** Projetar aplicações web utilizando JavaScript, DOM e eventos com APIs de persistência de dados no navegador web e controle de eventos, com objetividade e clareza.
6. **Manipulação da página web e controle de eventos:** Projetar aplicações web utilizando JavaScript, DOM e eventos com APIs de persistência de dados no navegador web e controle de eventos, com objetividade e clareza.

Especificações

1. Criar vídeos educativos:

Fazer vídeos que ensinam os conceitos básicos de programação, com exemplos fáceis de entender.

2. Montar material interativo:

Organizar um conjunto de códigos que mostrem, na prática, o que foi explicado nos vídeos.

3. Ensinar de forma divertida:

Usar métodos mais leves e divertidos para ajudar as pessoas a aprender programação com mais facilidade.

4. Incentivar o trabalho em grupo:

Motivar as pessoas a trabalharem juntas, trocando ideias e aprendendo umas com as outras.

5. Compartilhar o que foi criado:

Divulgar vídeos e materiais para o público, ajudando mais pessoas a aprender e mostrando o trabalho da instituição.

Planejamento de horas

Atividade / Matéria	Horas Estimadas
Reuniões semanais	60h
Planejamento Trello	20h
Desenvolvimento de aplicações Web: fundamentos para iniciantes	60h
HTML e CSS (linguagem de marcação e estilização)	50h
Design Responsivo (CSS3 e Media Queries)	50h
Padronização, acessibilidade e SEO (noções práticas)	40h
JavaScript (linguagem de scripting - básico)	55h
JavaScript e DOM (manipulação e controle de eventos)	55h
Gravação das aulas	20h
Edição das aulas	20h
Estruturação do GitHub (repositório, workflow,pages)	15h
Criação do site	20h
Revisão do projeto e testes	15h
Total Geral	480h

Detalhamento: "Desenvolvimento de aplicações Web — fundamentos para iniciantes"

Este módulo foi reescrito de forma simples, para quem nunca programou. O objetivo é dar uma visão prática e sem jargões:

- **O que é um site e como ele funciona** — navegador (cliente) e servidor (quem entrega os dados).
- **Requisições e respostas** (explicado com analogia: pedir um cardápio num restaurante e receber o prato).
- **Quando e por que usar APIs** (pequenos serviços que retornam dados, por exemplo uma lista de exercícios).
- **Conceitos leves de segurança:** não salvar senhas em texto, atenção ao compartilhar dados pessoais (apenas noções).
- **Ferramentas práticas:** exemplo de uma página que faz um pedido simples (GET) e mostra resultados na tela usando fetch, sem complexidade de servidores reais — usaremos mock simples.

Atividades práticas previstas (foco em aprender fazendo):

- 1) **Criar uma página que busca uma lista de itens (ex.: tópicos de estudo)** e mostra em uma tabela;
- 2) **Fazer um formulário simples** que envia dados para um endpoint mock e mostra mensagem de confirmação;
- 3) **Tratar erros comuns** (sem conexão, resposta vazia) e exibir mensagens amigáveis;
- 4) **Publicar a parte visual no GitHub Pages** para os alunos acessarem (mostrar passo a passo).

Planejamento individual de horas (resumo por membro)

Bruno — Total: 120h

GitHub: workflows, README, CI/CD, Pages — 25h - HTML/CSS (templates, integração) — 20h - Documentação técnica e lab guides — 20h - Revisão de código e suporte a deploy — 20h - Planejamento, Trello, commits e atualização de backlog — 15h - Reuniões semanais (obrigatórias) — 12h - Revisões finais / correções — 2h

Gabriel — Total: 120h

Gravação de aulas (roteiro, gravação) — 35h - Conteúdo JS: gravação e exercícios — 30h - Testes e revisão do site — 20h - Produção de exercícios interativos — 18h - Revisões e apoio pedagógico — 5h - Reuniões semanais (obrigatórias) — 12h - Revisões finais / correções — 2h

Murilo — Total: 120h

Design responsivo e UI components — 35h - Acessibilidade e checklist WAI-ARIA — 30h - Criação de materiais e assets visuais — 25h - Testes de usabilidade e acessibilidade — 18h - Apoio Trello e documentação — 0h (incluído em outras tarefas) — 0h - Reuniões semanais (obrigatórias) — 12h - Revisões finais / correções — 2h

Rafael — Total: 120h

Edição de vídeo (produção e pós) — 45h - Suporte ao front-end e integração de mídia — 25h - Planejamento e logística de gravação — 15h - Organização de entregáveis no Drive e revisão final — 23h - Reuniões semanais (obrigatórias) — 12h - Revisões finais / correções — 2h

Os números acima são uma generalização do processo. Cada membro pode ajustar a alocação, desde que o total final por pessoa seja 120h. As atividades executadas por mais de um membro devem ter suas horas somadas para efeito de total do projeto (por exemplo, atividade de 4h por membro com 2 participantes gera 8h registradas no total do projeto).

Metodologia

O projeto seguirá metodologias ágeis, utilizando **Scrum e Kanban** para organização.

- **Kanban (Trello):** gerenciamento das tarefas.
- **Scrum:** reuniões semanais, retrospectivas e definição de metas por sprint.

Sprint 1 — Fundamentos e setup (semanas 1–2) — total do sprint: 120 h (30h por membro)

Bruno — 30 h

- GitHub (workflows, README, branches) — 8 h
- Planejamento / backlog no Trello — 5 h
- Templates HTML básicos (estrutura inicial) — 8 h
- Reuniões semanais (presença) — 4 h
- Revisão inicial / checklist de aceite dos templates — 5 h

Gabriel — 30 h

- Roteiro das primeiras aulas e setup de gravação — 10 h
- Apoio ao planejamento (Trello / backlog) — 5 h

7

- Conteúdo inicial HTML/CSS (páginas de exemplo) — 5 h
- Reuniões semanais (presença) — 3 h
- Estudo/roteiro introdutório de JavaScript (pré-gravação) — 7 h

Murilo — 30 h

- Design dos templates e grid responsivo básico — 10 h
- Checklist de acessibilidade (WAI-ARIA, semântica) — 6 h
- Criação de assets visuais e exemplos (icons, imagens) — 6 h
- Reuniões semanais (presença) — 4 h
- Preparar exercícios iniciais / material prático — 4 h

Rafael — 30 h

- Setup de edição e fluxo de pós-produção — 10 h
 - Logística de gravação (local, roteiro técnico) — 6 h
 - Integração inicial de mídia nas páginas (upload/test) — 5 h
 - Reuniões semanais (presença) — 4 h
 - Edição inicial das primeiras gravações — 5 h
-

Sprint 2 — HTML/CSS e responsivo (semanas 3–4) — total do sprint: 120 h

Bruno — 30 h

- Publicação / integração com GitHub Pages — 8 h
- Componentes HTML/CSS reutilizáveis (cards, header, footer) — 10 h
- Reuniões semanais (presença) — 3 h
- Padronização de classes / guidelines de estilo — 4 h
- Documentação técnica básica no repositório — 5 h

Gabriel — 30 h

- Gravação das aulas de HTML e CSS (vários vídeos curtos) — 12 h
- Preparar exercícios para os alunos (guias passo a passo) — 8 h
- Reuniões semanais (presença) — 3 h
- Revisão/ajuste de roteiros conforme feedback — 7 h

Murilo — 30 h

- Responsividade: media queries e testes em 3 tamanhos — 12 h
- Ajustes de acessibilidade visuais (contraste, foco) — 6 h
- Produção de assets e exemplos visuais — 6 h

9

- Reuniões semanais (presença) — 3 h
- **Apoio no refinamento dos templates — 3 h**

Rafael — 30 h

- Edição dos vídeos de HTML/CSS — 12 h
 - Integração de mídia nas páginas e testes de playback — 8 h
 - Reuniões semanais (presença) — 3 h
 - Suporte ao deploy / correções pós-deploy — 7 h
-

Sprint 3 — JavaScript básico e interação (semanas 5–6) — total do sprint: 120 h

Bruno — 30 h

- Criar exemplos simples de consumo de dados (fetch com mock) — 10 h
- Exemplos e mini-labs de JS (variáveis, funções) — 6 h
- Reuniões semanais (presença) — 3 h
- Revisão de PRs / documentação técnica dos exemplos — 6 h
- Suporte à integração (front/back mock) — 5 h

Gabriel — 30 h

- Gravação das aulas de JavaScript básico (vários vídeos) — 12 h
- Criar exercícios interativos (mostrar/esconder, formulários) — 10 h
- Reuniões semanais (presença) — 3 h
- Revisões de roteiro e acompanhamento pedagógico — 5 h

Murilo — 30 h

- Implementar exercícios interativos (HTML + JS simples) — 12 h
- Ajustes de UI/UX nas interações (feedback visual) — 8 h
- Reuniões semanais (presença) — 3 h
- Testes de acessibilidade nas interações — 7 h

Rafael — 30 h

- Edição dos vídeos de JS e tutoriais — 12 h
 - Produzir mini-tutoriais em vídeo (ex.: como usar o exercício) — 8 h
 - Reuniões semanais (presença) — 3 h
 - Compressão/otimização de mídia para web — 7 h
-

Sprint 4 — Integração, revisão e publicação (semanas 7–8) — total do sprint: 120 h

Bruno — 30 h

- Revisão final do repositório, CI/CD simples e documentação — 10 h
- Limpeza do repo / organização de pastas e tags/releases — 6 h
- Reuniões semanais (presença) — 4 h
- Preparar release e instruções de uso/instalação — 5 h
- Suporte ao deploy final / correções rápidas — 5 h

Gabriel — 30 h

- Finalizar gravações faltantes (caso necessário) — 8 h
- Revisão pedagógica dos materiais e checklist de aprendizagens — 10 h
- Reuniões semanais (presença) — 3 h
- Apoio nos testes finais com alunos beta / feedback — 9 h

Murilo — 30 h

- Auditoria final de acessibilidade e performance simples — 10 h
- Ajustes SEO básico (meta tags, título, sitemaps) — 8 h
- Criar material complementar (resumos, cheatsheets) — 7 h

12

- Reuniões semanais (presença) — 3 h
- Apoio nos testes finais — 2 h

Rafael — 30 h

- Edição final dos vídeos e exportação master — 12 h
- Organização dos entregáveis no Drive / indexação — 10 h
- Reuniões semanais (presença) — 4 h
- QA final do site (playback, responsividade) — 4 h

Estrutura do site

- **Seção Inicial:** trará uma breve descrição do curso, seus objetivos principais e uma imagem de destaque que represente o tema do projeto.
- **Área de Conteúdo:** reunirá todas as aulas em vídeo, materiais de estudo e exercícios práticos, organizados para facilitar a navegação do usuário.
- **Seção de Contato:** espaço com informações sobre os membros do grupo e links diretos para seus perfis profissionais (GitHub e LinkedIn).

Ferramentas de Apoio

- **GitHub Pages:** utilizado para hospedar e disponibilizar o site online.
- **Trello:** será a base para o controle das atividades e acompanhamento do progresso.

- **Google Drive:** servirá como repositório para arquivos, documentos e vídeos do projeto.
- **Visual Studio Code:** editor de código escolhido como ambiente principal de programação.

GitHub

- **As mensagens devem ser objetivas;**
- **Cada commit deve seguir o padrão: tipo da alteração** (feat, fix, docs, style, refactor, test) **+ descrição curta.**
- **Recomenda-se fazer commits frequentes e focados em pequenas partes do código, garantindo clareza no histórico de mudanças.**