

Trabalho 1: Desenvolvimento de Aplicações Web - Front-end

Este trabalho corresponde à primeira entrega do projeto da disciplina **Desenvolvimento de Aplicações Web**, que será desenvolvido de forma incremental ao longo do semestre. O objetivo principal é a construção do **front-end** da aplicação proposta por cada equipe.

Prazos

- Entrega (limite para atualização do git): 12:00 horas de 29/09/2025.
- Entrevistas Técnicas / Apresentação: Aula do dia 29/09/2025.

Entregáveis

Cada equipe (3 integrantes) deverá entregar **duas versões do front-end**:

1. Protótipo Inicial

- Implementado utilizando apenas **HTML, CSS, JavaScript e Bootstrap**.
- Deve apresentar as principais telas da aplicação, com foco em **estrutura, layout e usabilidade básica**.

2. Protótipo Avançado

- Evolução do protótipo inicial, implementada em **React**.
- Deve priorizar a **componentização da aplicação**, visando organização, reuso e manutenção.
- Deve incluir a versão estática do projeto para execução no lado-cliente

Instruções de Entrega

- A entrega será considerada com base no **repositório Git da equipe**.
- As duas versões devem estar claramente **identificadas e separadas** no repositório (ex.: branches, pastas ou tags).
- Cada equipe deverá enviar no **Moodle** apenas os **links correspondentes** a cada versão.
- Ambas as versões devem estar **documentadas adequadamente**, incluindo:

- Instruções de instalação e execução;
- Estrutura do projeto;
- Breve descrição das telas e funcionalidades implementadas.

Entrevista Técnica

Após a entrega, cada equipe participará de uma **entrevista técnica** com o professor.

- O objetivo é avaliar o **domínio do grupo sobre o projeto desenvolvido**, abordando decisões técnicas, organização do código, uso das tecnologias e justificativas de implementação.
- Todos os integrantes deverão participar.
- A **nota final do trabalho será ponderada pelo desempenho individual na entrevista**, de forma a garantir o domínio do conteúdo trabalhado.
- A ponderação será no formato Nota Rubrica * Nota Entrevista, onde Nota Entrevista será:
 - Domínio muito bom: 0,8 a 1,0
 - Bom domínio: 0,6 a 0,8
 - Domínio regular: 0,4 a 0,6
 - Domínio insatisfatório: 0,2 a 0,4
 - Ausência ou incapacidade de defesa: 0,0 a 0,2.

Critérios de Avaliação

A avaliação será feita de acordo com a seguinte rubrica:

Critério	Excelente (nota máxima)	Muito Bom	Bom	Regular	Insatisfatório
----------	-------------------------	-----------	-----	---------	----------------

Organização e Estrutura do Código (10%)	Estrutura de pastas e arquivos muito bem organizada; separação clara e intuitiva das versões; código padronizado e bem comentado.	Estrutura bem organizada, com separação clara das versões; poucos ajustes de padronização necessários.	Estrutura adequada, mas com inconsistências de organização ou padronização; separação das versões aceitável.	Estrutura pouco organizada; separação das versões confusa ou mal documentada.	Estrutura desorganizada; ausência de separação clara das versões; código ilegível.
Qualidade do Protótipo Inicial (30%)	Protótipo completo, funcional e visualmente consistente; ótima usabilidade; excelente uso de HTML, CSS, JS e Bootstrap.	Protótipo funcional e com bom layout; pequenas falhas de usabilidade ou consistência visual.	Protótipo razoavelmente funcional; algumas falhas evidentes de layout/usabilidade .	Protótipo incompleto ou pouco funcional; navegação limitada ou confusa.	Protótipo ausente ou não funcional.
Qualidade do Protótipo Avançado (50%)	Implementação em React completa, com componentização clara, bem estruturada e reutilizável; evolução significativa em relação ao protótipo inicial.	Implementação em React funcional, com boa componentização; evolução perceptível em relação ao protótipo inicial.	Implementação em React adequada, mas com componentização limitada; evolução parcial em relação ao protótipo inicial.	Implementação básica em React, com pouca ou nenhuma componentização; evolução mínima em relação ao protótipo inicial.	Implementação ausente ou insuficiente em React.
Documentação (10%)	README completo, claro e bem estruturado; inclui instruções detalhadas de instalação, execução e descrição das funcionalidades; ambas as versões bem documentadas.	README consistente, com todas as informações necessárias; pequenos ajustes de clareza poderiam melhorar.	README presente e funcional, mas com informações incompletas ou superficiais.	README muito básico, sem detalhes importantes para execução ou entendimento.	Ausência de documentação ou documentação insuficiente.