

## Documentação de atividade: Construindo uma Página Interativa com Framework/Biblioteca.

Aluno: Ícaro Gabriel Alves dos Reis.

Turma: FACET-SNP-307 – Desenvolvimento Web – T01 (2025.1 – 6N123)

### Justificativa da escolha da biblioteca/framework

Após pesquisas, optei pelo Vue 3 devido à sua abordagem intuitiva baseada em componentes, além da ampla disponibilidade de tutoriais e documentação, o que facilitou a implementação do projeto.

### Descrição das funcionalidades implementadas

**Página 1 Champions:** Exibe os últimos campeões do CBLOL com animações de scroll e hover nas imagens.

**Página 2 Teams:** Mostra todos os times campeões, permitindo busca e ordenação por data ou número de títulos cards expansíveis.

**Página 3 Players:** Lista os 10 jogadores com mais títulos, com filtros (ordenação por nome, títulos ou mais recente) e cards expansíveis.

Como componentes reutilizáveis foram desenvolvidos o Header com navegação responsiva e alternador de tema, Footer com link para o GitHub e ThemeToggle para o botão interativo com objetivo de alternar os temas com animação de flip.

Os recursos interativos foram o tema claro ou escuro que pode ser alternado, transições entre rotas (efeito de fade durante navegação entre as páginas), animações usando biblioteca AOS(scroll), transições Vue (hover e expansão de cards) e responsividade.

### Explicação de como a biblioteca/framework foi utilizada em cada parte do projeto

O Vue.js foi utilizado em todas as camadas do projeto, aproveitando o fato de ser componentizado.

Composition API foi utilizado no App.vue para gerenciamento de tema, no Champions.vue para animação de hover e no Players.vue/Teams.vue para filtros e ordenação.

Vue Router utilizado no router/index.js para configuração de rotas, Header.vue para os links de navegação e App.vue para transições entre rotas.

Computed Properties utilizado no Players.vue para filtragem e ordenação de jogadores e no Teams.vue para processamento de dados de times.

Além disso, a biblioteca AOS foi utilizada na implementação da animação de scroll da Champions.vue.

Instruções para executar e testar a página:

O projeto será disponibilizado em uma plataforma gratuita para maior simplicidade.

Desafios encontrados e soluções adotadas:

#### **Desafio 1: Persistência do Tema**

Problema: O tema não era mantido ao recarregar a página.

Solução: Armazenar preferência no localStorage e verificar preferência do sistema.

#### **Desafio 2: Layout Responsivo dos Cards**

Problema: Quebra de layout em telas pequenas.

Solução: Uso de CSS Grid com media queries e flex-wrap.

#### **Desafio 3: Animações com Dados Dinâmicos**

Problema: Efeitos de transição não funcionavam com filtros.

Solução: Uso de <transition-group> com flip-list.

#### **Desafio 4: Configuração de Deploy com Vue CLI**

Problema: Documentação desatualizada e tutoriais focados em Vite causavam erros no deploy.

Solução: Configuração personalizada no vue.config.js com publicPath dinâmico e uso do pacote gh-pages com script dedicado.

#### **Desafio 5: Redirecionamento Inicial no GitHub Pages**

Problema: O site não iniciava automaticamente na rota /champions como planejado, mesmo com o redirecionamento configurado no Vue Router. Ao acessar o link, a página carregava em branco ou exibia erros de rota.

Solução: Configuração do Vue Router para usar Hash Mode, redirecionamento explícito na rota raiz e script adicional no index.html para garantir o carregamento inicial.