

Programação Web I

UNIDADE CURRICULAR

Ficha 08 – Preparação para Exame

FICHA

Abra o VSC e crie um projeto Vue com o nome **cocktail-explorer**Lembre-se de selecionar Vue Router e Pinia durante o setup.Utilize exclusivamente Options API.

API – Informação Essencial

Base URL: <https://www.thecocktailedb.com/api/json/v1/1/>

Endpoint	Descrição
<code>random.php</code>	Retorna um cocktail aleatório
<code>lookup.php?i={id}</code>	Retorna detalhes de um cocktail por ID
<code>search.php?s={nome}</code>	Pesquisa cocktails por nome
<code>filter.php?c={categoria}</code>	Lista cocktails por categoria
<code>list.php?c=list</code>	Lista todas as categorias disponíveis

O endpoint `filter.php` retorna dados simplificados (apenas `idDrink`, `strDrink`, `strDrinkThumb`). Para obter detalhes completos, é necessário utilizar o endpoint `lookup.php`.

- 1) Configure o projeto Vue com a seguinte estrutura de rotas
 - `HomeView` - página principal com listagem de cocktails
 - `CocktailDetailView` - página de detalhes de um cocktail
 - `RandomCocktailView` - página para descobrir cocktails aleatórios
- 2) Crie um ficheiro com funções utilitárias para comunicar com a API
 - Implemente as seguintes funções assíncronas:
 - `getRandomCocktail()` - obtém um cocktail aleatório
 - `getCocktailById(id)` - obtém detalhes de um cocktail específico
 - `searchCocktails(query)` - pesquisa cocktails por nome
 - `getCategories()` - obtém lista de categorias disponíveis

- o `get CocktailsByCategory(category)` - lista cocktails de uma categoria
- 3) Crie uma store `useCocktailsStore`
- A store deve gerir o seguinte estado:
 - o Lista de cocktails para a homepage
 - o Cocktail atualmente selecionado (para página de detalhes)
 - o Lista de categorias disponíveis
 - o Estado de carregamento (loading)
 - o Mensagem de erro (caso ocorra) - Implemente actions para:
 - o Carregar cocktails por categoria
 - o Carregar detalhes de um cocktail por ID
 - o Carregar um cocktail aleatório
 - o Carregar lista de categorias
 - o Pesquisar cocktails por nome - Getters:
 - o Implemente um getter que processe o cocktail atual e retorne um array limpo de ingredientes, combinando cada ingrediente com a sua medida correspondente

A API retorna os ingredientes em campos separados (`strIngredient1, strIngredient2, ...` até `strIngredient15`), muitos dos quais podem ser null. As medidas seguem o mesmo padrão (`strMeasure1, strMeasure2`, etc.).

Exemplo de output esperado:

```
[  
  { ingredient: "Tequila", measure: "1 1/2 oz" },  
  { ingredient: "Triple sec", measure: "1/2 oz" },  
  { ingredient: "Lime juice", measure: "1 oz" }  
]
```

 - o Implemente getters auxiliares para verificar se há cocktails carregados e se ocorreu algum erro.

4) Implemente a página principal da aplicação

 - Filtro por categoria:
 - o Dropdown/select populado com as categorias da API
 - o Ao selecionar uma categoria, a lista de cocktails deve atualizar
 - o Deve mostrar uma categoria por defeito
 - o Mostrar indicador do modo atual com contagem, por exemplo: “Categoria: Cocktail (25 cocktails)” - Pesquisa por Nome:
 - o Campo de texto para pesquisar cocktails
 - o A pesquisa deve ser acionada por um botão ou ao pressionar Enter
 - o Durante uma pesquisa ativa:

1. O dropdown de categorias deve ficar desativado. E, perto do mesmo, apresentar algo como: "(pesquisa ativa)"
 2. Modificar o indicador para mostrar algo como: "Resultados para 'marg' (6 encontrados)"
 3. Incluir botão para limpar a pesquisa e voltar ao modo de categoria
- Listagem de Cocktails:
 - Apresentar cada cocktail num card (deve ser reutilizável) e em formato grid
 - Cada cocktail deve mostrar: imagem, nome
 - Ao clicar num cocktail, navegar para a página de detalhes
 - Mostrar indicador de loading enquanto os dados carregam
 - Mostrar mensagem apropriada se não houver resultados ou em caso de erro
- 5) Implemente a página para descobrir cocktails aleatórios
- Apresentar o cocktail num card com imagem e nome
 - Botão “Descobrir Outro” que carrega um novo cocktail aleatório
 - Ao clicar no card, navegar para a página de detalhes
- 6) Implemente a página de detalhes de um cocktail
- Carregamento de Dados:
 - Obter o ID do cocktail a partir dos parâmetros da rota
 - Carregar os detalhes completos do cocktail ao entrar na página
 - Apresentar as seguintes informações do cocktail:
 - Imagem (tamanho adequado)
 - Nome
 - Categoria (strCategory)
 - Tipo de copo (strGlass)
 - Se é alcoólico ou não (strAlcoholic)
 - Instruções de preparação (strInstructions)
 - Lista de ingredientes com quantidades (utilizar o getter da store)
 - Botão para voltar à página anterior ou à homepage
- 7) Funcionalidade Extra: Favoritos
- Implemente um sistema de favoritos que permita ao utilizador marcar cocktails:
 - O utilizador pode marcar/desmarcar cocktails como favoritos
 - Os favoritos devem persistir entre sessões (sobreviver a refresh da página)
 - Mostrar visualmente quais cocktails estão marcados como favoritos