**Sugestões de Funcionalidades Interessantes para Adicionar:**

1. **Aporte Inicial:** Permitir que o usuário informe um valor inicial de patrimônio já investido, além dos aportes mensais. Isso é muito comum.
2. **Correção pela Inflação:** Adicionar um campo para o usuário inserir uma taxa de inflação anual estimada. O simulador poderia corrigir os aportes, o custo de vida desejado e, opcionalmente, os dividendos/preços ao longo do tempo para dar uma visão mais realista do poder de compra futuro.
3. **Valorização/Desvalorização do Ativo:** Permitir adicionar uma taxa de crescimento (ou decrescimento) anual estimada para o *preço* de cada ativo, além do rendimento via dividendos. Isso adicionaria outra dimensão de crescimento (ou risco) ao patrimônio.
4. **Taxas e Impostos:** Incluir opções para simular o impacto de taxas (corretagem, custódia) e impostos (como Imposto de Renda sobre dividendos ou ganho de capital, se aplicável no futuro ou para certos ativos).
5. **Rebalanceamento de Carteira:** Implementar uma opção para simular o rebalanceamento periódico (ex: anual ou semestral), onde o simulador venderia parte dos ativos que cresceram acima da alocação e compraria dos que ficaram abaixo, para manter os percentuais definidos.
6. **Aportes Variáveis:** Permitir que o usuário configure um aumento percentual anual nos aportes (ex: acompanhar um aumento salarial) ou até mesmo definir aportes diferentes para períodos específicos.
7. **Comparação de Cenários:** Possibilitar que o usuário salve uma simulação e depois altere alguns parâmetros (ex: valor do aporte, carteira de ativos) para rodar um novo cenário e comparar os resultados lado a lado.
8. **Exportar Resultados:** Adicionar botões para exportar a tabela detalhada para CSV/Excel e talvez salvar os gráficos como imagem (PNG).
9. **Salvar/Carregar Simulações:** Ir além do localStorage para ativos, permitindo salvar toda a configuração da simulação (aportes, período/meta, custo de vida, carteira) com um nome, e carregá-la posteriormente. Isso poderia usar localStorage para múltiplos "slots" ou exigir um backend simples.
10. **Busca de Ativos (Avançado):** Integrar com alguma API financeira (ex: Alpha Vantage, IEX Cloud, ou APIs brasileiras se disponíveis) para buscar informações de preço e dividendos recentes de ativos reais, facilitando o preenchimento para o usuário (isso adicionaria complexidade e dependência externa).
11. **Considerar a Inflação:**
    * **O quê:** Adicionar um campo para o usuário inserir uma taxa de inflação anual estimada (ex: 5%).
    * **Por quê:** A inflação corrói o poder de compra ao longo do tempo. Isso permitiria:
      + Corrigir a "Meta de Patrimônio" e o "Custo de Vida Mensal Desejado" pela inflação, mostrando o valor real necessário no futuro.
      + Apresentar os resultados finais (patrimônio, dividendos) também em valores "reais" (descontando a inflação), dando uma visão mais realista do poder de compra futuro.
    * **Impacto:** Aumenta significativamente o realismo da simulação de longo prazo.
12. **Estimativa de Valorização dos Ativos (Preço):**
    * **O quê:** Permitir que o usuário adicione uma taxa de valorização anual estimada *além* dos dividendos, seja para a carteira como um todo ou, de forma mais avançada, por ativo.
    * **Por quê:** Atualmente, o crescimento do patrimônio vem apenas dos aportes e do reinvestimento dos dividendos informados (o preço do ativo é fixo na simulação). A valorização (ou desvalorização) do preço dos ativos é um componente crucial do retorno total do investimento.
    * **Impacto:** Torna a simulação mais dinâmica e completa, refletindo melhor o comportamento esperado de ativos de renda variável. No entanto, adiciona mais uma variável especulativa.
13. **Aportes Mensais Variáveis/Crescentes:**
    * **O quê:** Adicionar uma opção para que o aporte mensal cresça anualmente a uma taxa percentual definida pelo usuário (ex: aumentar o aporte em 5% ao ano).
    * **Por quê:** É comum que a capacidade de aporte das pessoas aumente ao longo do tempo (promoções, etc.). Simular isso torna o planejamento mais alinhado à realidade de muitos investidores.
    * **Impacto:** Melhora o realismo da projeção de longo prazo.

**Novas Funcionalidades e Melhorias de UX:**

1. **Salvar e Carregar Cenários:**
   * **O quê:** Permitir que o usuário salve a configuração atual da simulação (aportes, período/meta, carteira de ativos, inflação, etc.) com um nome, usando o localStorage. Adicionar uma opção para carregar cenários salvos.
   * **Por quê:** Facilita testar diferentes estratégias ou revisitar simulações anteriores sem ter que inserir todos os dados novamente.
   * **Impacto:** Grande melhoria na usabilidade para usuários que desejam explorar múltiplas hipóteses.
2. **Comparação de Cenários (Avançado):**
   * **O quê:** Após implementar o "Salvar/Carregar", permitir que o usuário carregue dois cenários e veja os resultados (resumo e talvez gráficos) lado a lado.
   * **Por quê:** Ajuda o usuário a visualizar diretamente o impacto de diferentes variáveis (ex: aportar mais vs. menos, carteira A vs. B).
   * **Impacto:** Ferramenta poderosa para tomada de decisão, mas exige mudanças significativas na UI para exibir comparações.
3. **Importar/Exportar Carteira de Ativos:**
   * **O quê:** Adicionar botões para exportar a lista de ativos atual para um arquivo simples (CSV ou JSON) e para importar ativos de um arquivo nesse formato.
   * **Por quê:** Facilita o gerenciamento para usuários com muitos ativos ou que usam outras ferramentas/planilhas.
   * **Impacto:** Conveniência para usuários com carteiras mais complexas.
4. **Melhorias nos Gráficos:**
   * **O quê:** Adicionar mais interatividade, como:
     + Tooltips mais informativos (ex: mostrar o valor exato do patrimônio E dos aportes no mesmo ponto do gráfico de evolução).
     + Opção de habilitar/desabilitar datasets (ex: esconder a linha de "Total Aportado").
     + Talvez botões para exportar os gráficos como imagem (PNG/JPG).
   * **Por quê:** Aumenta a utilidade e a clareza das visualizações.
   * **Impacto:** Melhoria na análise dos dados visuais.
5. **Informações Adicionais nos Ativos:**
   * **O quê:** No modal de adicionar/editar ativo, talvez adicionar um campo opcional para "Link de Referência" (ex: StatusInvest, FundsExplorer) ou exibir o Yield Anual calculado diretamente no modal.
   * **Por quê:** Oferece contexto rápido ou acesso fácil a mais informações sobre o ativo.
   * **Impacto:** Pequena melhoria de conveniência.

**1. Feedback Durante o Cálculo (Loading State):**

* **Problema:** Simulações longas podem fazer a interface parecer "travada" enquanto o JavaScript calcula.
* **Sugestão:** Ao clicar no botão "Simular Resultados", desabilitar o botão e mudar seu texto/ícone para indicar que o cálculo está em andamento (ex: "Simulando..." com um ícone de spinner). Ao final do cálculo (dentro do finally da calcularSimulacaoPrincipal), reabilitar o botão e restaurar o texto/ícone original.
* **Benefício:** Dá um feedback visual imediato ao usuário de que a ação foi recebida e está sendo processada.
* **Implementação:** Requer pequenas alterações no CSS (para o estilo do botão desabilitado/loading) e no JS (adicionar/remover classe/atributo disabled e mudar innerHTML do botão no início e fim de calcularSimulacaoPrincipal).

**2. Clareza nos Resultados (Nominal vs. Real):**

* **Problema:** Exibir valores Nominais e Reais juntos no resumo pode ser confuso se não for bem claro.
* **Sugestão:** No resumo (#resultado-simulacao), em vez de apenas colocar o valor real entre parênteses, usar uma formatação mais explícita ou adicionar um title explicativo.
  + Exemplo 1 (Texto): Patrimônio Final: R$ 1.000 (Nominal) / R$ 800 (Real Hoje)
  + Exemplo 2 (Com Tooltip): Patrimônio Final: R$ 1.000 <small>(Real: R$ 800)</small> e adicionar um title="Valor nominal futuro / Valor real em poder de compra de hoje" no <span> do label. A versão atual já usa small e title, podemos apenas garantir que o texto do title seja claro.
* **Benefício:** Reduz a ambiguidade sobre o significado dos valores apresentados, especialmente com a inflação.
* **Implementação:** Ajustes na função generateSummaryHtml do JS.

**3. Organização da Seção de Ativos:**

* **Problema:** Ter a lista principal de ativos e a lista de prioridade arrastável visíveis ao mesmo tempo pode ocupar muito espaço ou parecer um pouco repetitivo.
* **Sugestão:** Manter a estrutura atual (com as duas listas), mas garantir que a seção de reordenar prioridade (#priority-reorder-section) esteja claramente posicionada **abaixo** da lista principal (#ativos-list) e do botão "Adicionar/Modificar". Isso cria um fluxo mais lógico: veja seus ativos, adicione/edite, *depois* ordene a prioridade deles. O CSS que forneci já deve ajudar na separação visual com bordas e espaçamento.
* **Benefício:** Melhora o fluxo visual e a organização da informação nessa seção.
* **Implementação:** Principalmente garantir a ordem correta dos divs no HTML (como feito na última versão HTML que te passei).

**4. Tabela Detalhada Responsiva:**

* **Problema:** Com a adição da coluna "Patrimônio Total (Real)", a tabela ficou ainda mais larga.
* **Sugestão:** Manter a solução atual do container com rolagem horizontal (.table-container com overflow-x: auto) como principal forma de lidar com a largura em telas menores. Podemos revisar o padding das células (th, td) no CSS para garantir que não seja excessivo, otimizando o espaço horizontal. Esconder colunas em telas menores é possível, mas complica bastante a implementação e pode ocultar dados úteis.
* **Benefício:** Garante que todos os dados estejam acessíveis em qualquer tela, mesmo que exija rolagem.
* **Implementação:** Ajustes finos no padding dentro das regras .data-table th, .data-table td no CSS, se necessário.

**5. Tooltips e Acessibilidade:**

* **Sugestão:** Adicionar atributos title em mais locais pode ser útil, por exemplo, nos ícones dos input-groups para explicar rapidamente o campo, ou em cabeçalhos de colunas da tabela. Usar aria-labels em botões puramente icônicos (se houver) pode melhorar a acessibilidade para leitores de tela.
* **Benefício:** Melhora a usabilidade e acessibilidade geral.
* **Implementação:** Adicionar atributos title e aria-label no HTML e no JS onde elementos são gerados dinamicamente.