Relatório da Calculadora LISP

Universidade Salvador (UNIFACS)

Salvador, BA

2025

Integrantes:

Orientador	Alunos	Contribuição
Wellington Silveira Lacerda	Victor Telles Chaves	Lógica e Algoritmos
	Alice Martins Bahiense Bezerra Bauler	Designer de Interface
	Pedro Henrique de Oliveira Carvalho	Desenvolvedor Frontend

Salvador, BA

2025

Calculadora Científica React/Electron

1. Introdução

Este relatório documenta o desenvolvimento da Calculadora Científica, um aplicativo desktop multiplataforma construído com React, Electron e TailwindCSS. O projeto visa oferecer uma experiência moderna, intuitiva e robusta para cálculos matemáticos avançados, incluindo suporte a variáveis, histórico, números complexos e interface responsiva.

2. Objetivos

- Desenvolver uma calculadora científica com interface gráfica moderna.
- Permitir cálculos com números reais e complexos.
- Suportar variáveis definidas pelo usuário.
- Oferecer histórico de operações e ajuda integrada.
- Garantir usabilidade, acessibilidade e compatibilidade cross-platform.
- Integrar uma IA para servir de assistente nos cálculos.

3. Arquitetura e Tecnologias

Frontend: React 19.x (componentização, hooks, JSX)

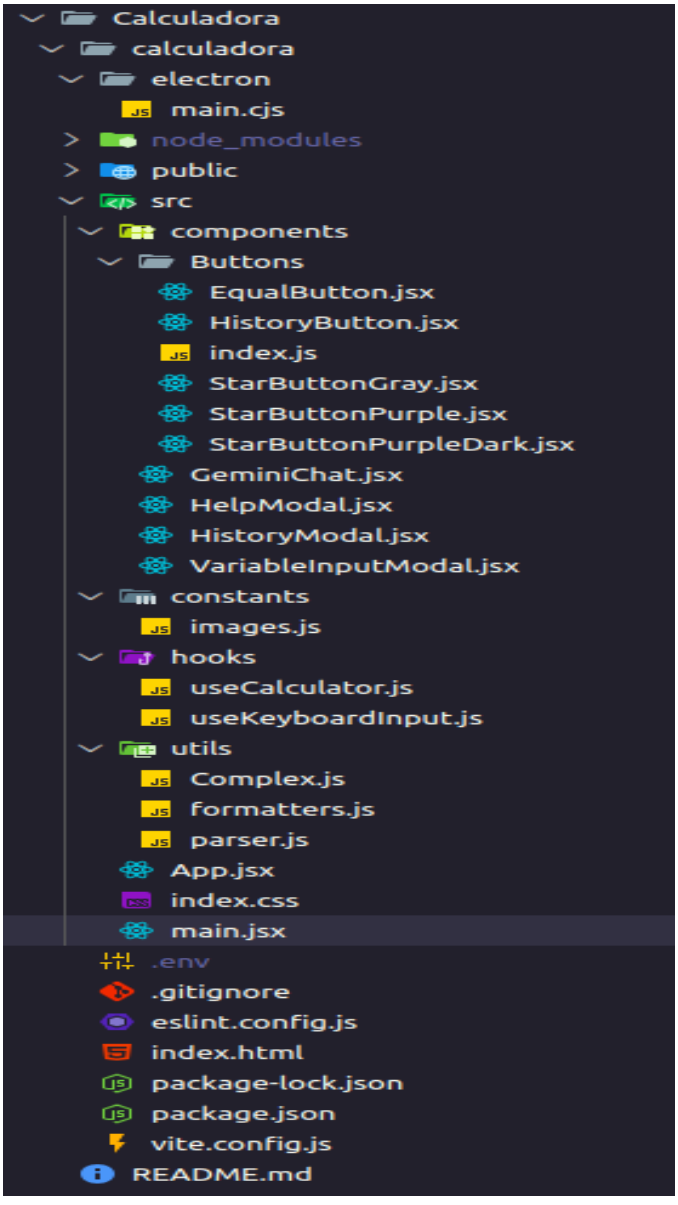
Estilização: TailwindCSS 4.x (utilitários, responsividade, scrollbar customizada)

Desktop: Electron (empacotamento e integração nativa)

Gerenciamento de Estado: Hooks customizados (**useCalculator**, **useKeyboardInput**)

Estrutura Modular: Separação clara entre componentes, hooks, utilitários e constantes

4. Estrutura de Pastas



5. Componentes Principais

5.1 Modais

- HistoryModal.jsx: Exibe o histórico das últimas 10 operações. Fecha com ESC ou botão X. Scrollbar customizada.
- HelpModal.jsx: Ajuda/documentação do sistema. Fecha com ESC ou botão X. Modal grande e responsivo.
- VariableInputModal.jsx: Permite entrada/edição de variáveis, inclusive números complexos. Fecha com ESC ou Cancelar.
- GeminiChat.jsx: contém toda a lógica da integração com a IA na calculadora.

5.2 Botões

- EqualButton, HistoryButton, StarButtonGray/Purple: Botões customizados, com feedback visual (isActive) e integração ao teclado.
- Barrel Export: components/Buttons/index.js centraliza exportação dos botões.

5.3 Hooks

- useCalculator.js: Gerencia estado global (histórico, variáveis, modais, cálculo).
- useKeyboardInput.js: Captura eventos de teclado, ativa botões, desativa input com modal aberto.

5.4 Utilitários

- Complex.js: Operações com números complexos.
- formatters.js: Formatação de resultados.
- parser.js: Análise de expressões matemáticas.

5.5 Constantes

- images.js: Centraliza URLs de imagens/ícones para uso consistente.

6. Funcionalidades

- Cálculos científicos: Suporte a operações avançadas, parênteses, funções matemáticas.
- Números complexos: Entrada e manipulação nativa.
- Variáveis: Definição, edição e uso em expressões.

- Assistente de IA: ajuda o usuário a entender como funciona o último cálculo feito.
- Histórico: Visualização e reutilização de operações anteriores.
- Ajuda integrada: Modal com instruções e exemplos.
- Acessibilidade: Fechamento de modais via ESC, feedback visual nos botões.
- Scrollbars customizadas: Aparência consistente em Chrome e Firefox.
- Responsividade: Layout adaptável a diferentes tamanhos de tela.

7. Lições Aprendidas

- Importância da modularização: Separação clara de responsabilidades facilita manutenção e testes.
- Atenção à acessibilidade: Pequenos detalhes (ESC para fechar, feedback visual) melhoram muito a experiência.
- Cross-browser: Customização de scrollbar exige atenção a diferenças entre navegadores.
- Integração com IA não é tão difícil, mas requer muito cuidado (principalmente com a chave de API).
- Documentação contínua: Manter README, guias e relatório atualizados agiliza entregas e apresentações.

8. Conclusão

O projeto atinge todos os objetivos propostos, entregando uma calculadora científica robusta, moderna, fácil de usar e ainda com uma IA integrada para auxiliar o usuário. A arquitetura modular, o uso de hooks customizados e a atenção à experiência do usuário garantem um produto de alta qualidade, pronto para uso acadêmico, profissional ou pessoal.

9. Detalhes

Temos deploy: https://calculadora-eight-lake.vercel.app/

Porém o assistente de IA é bloqueado no deploy por questões de segurança (por conta da chave de API, que não é uma boa ideia ser pública), então o assistente de IA só é acessado localmente.

Para mais informações sobre como usar a aplicação e desbloquear o assistente de IA, acesse o link do nosso repositório:https://github.com/VIctor-teles-Dev/Calculadora

Leia o README do projeto!