

Projeto do Curso: Simulador de Mercado de Ações em Tempo Real

Descrição do Projeto

O projeto consiste no desenvolvimento de um simulador de mercado de ações que fornece dados em tempo real sobre o comportamento de ações fictícias, incluindo gráficos interativos, análise de desempenho histórico e cálculos de indicadores financeiros comuns (ex.: média móvel, volatilidade). O projeto utiliza Rust para a lógica de negócios e cálculos financeiros, e WebAssembly para garantir alta performance no navegador ao processar grandes volumes de dados.

Funcionalidades do Simulador

1. Simulação de Dados em Tempo Real:

- Gerador de dados aleatórios que simula o comportamento de várias ações no mercado.
- Rust gera dados de preços e volumes em intervalos regulares (ex.: a cada segundo), criando cenários de alta e baixa.

2. Análise de Indicadores Financeiros:

- Cálculo de indicadores como:
- Média móvel (SMA e EMA)
- Desvio padrão (para medir a volatilidade)
- Índice de Força Relativa (RSI)
- Rust processa os cálculos em tempo real; WebAssembly renderiza gráficos com alta performance.

3. Gráficos Interativos:

- Exibição em tempo real no navegador com JavaScript e WebAssembly.
- Gráficos exibem:
- Preço das ações ao longo do tempo.
- Volume de transações.
- Linhas de tendência (ex.: médias móveis).
- WebAssembly garante performance e atualização contínua.

4. Integração com APIs Reais (Opcional):

- Pode consumir dados reais de APIs como Yahoo Finance ou Alpha Vantage.
- Mesmo com dados reais, o processamento é local com Rust + WebAssembly.

5. Previsão e Cenários de Mercado:

- Comportamento simulado permite prever cenários futuros (ex.: alta ou queda).
- Rust executa modelos como extrapolação linear ou regressão.

6. Interatividade com o Usuário:

- Usuário pode selecionar ações, períodos e ajustar frequência de atualização.
- Simula compra e venda de ações fictícias com feedback de desempenho.

7. Otimização de Performance:

- Rust lida com grandes volumes de dados históricos sem prejudicar fluidez.

Tecnologias Utilizadas

- Rust: lógica de negócios, cálculos e simulação.
- WebAssembly: cálculos pesados no navegador com performance nativa.
- JavaScript: integração DOM e gráficos.
- APIs externas (opcional): dados financeiros reais.
- Ferramentas: wasm-pack, wasm-bindgen, web-sys.

Estrutura do Projeto

- Gerador de Dados: módulo Rust para geração periódica de preço, volume e timestamp.
- Indicadores Financeiros: módulo Rust com algoritmos de média móvel, volatilidade e RSI.
- Interface Web: comunicação Rust ↔ JavaScript com gráficos interativos via wasm-bindgen.
- Interação com Usuário: web UI com seleção de ações, períodos e simulação de investimentos.
- APIs (Opcional): módulo para substituição de dados simulados por reais.

Desafios Técnicos e Pilares de Aprendizado

- Gerenciamento de Memória e Performance com Rust.
- Integração eficiente entre Rust e JavaScript com WebAssembly.
- Escalabilidade da aplicação com múltiplas ações e grandes volumes de dados.

Entregáveis

- Simulador funcionando com lógica em Rust e interface WebAssembly.
- Gráficos interativos com indicadores visuais.
- Relatório comparativo de performance: Rust vs JavaScript.
- Documentação completa com instruções e arquitetura.
- Apresentação final com demonstração e análise técnica.