

Documentação do Sistema Capitalize System

Sistema Avançado de Backtesting e Validação de Estratégias para o Mercado Financeiro Brasileiro

- Versão:** 2.0
- Data:** 22/05/2025
- Autor:** Equipe de Desenvolvimento Capitalize System

Índice

- [1. Visão Geral do Projeto](#)
- [2. Requisitos do Sistema](#)
- [3. Arquitetura do Sistema](#)
- [4. Estrutura de Diretórios](#)
- [5. Roadmap de Desenvolvimento](#)
- [6. MVPs e Métricas de Sucesso](#)
- [7. Fluxos de Trabalho](#)
- [8. Gestão de Dados](#)
- [9. Riscos e Estratégias de Mitigação](#)
- [10. Integração com MetaTrader](#)
- [11. Apêndices](#)

1. Visão Geral do Projeto

O **Capitalize System** é um sistema abrangente de backtesting para estratégias de trading no mercado financeiro brasileiro, abrangendo tanto o mercado à vista quanto o mercado futuro. O sistema é projetado para permitir a avaliação rigorosa de estratégias de trading através de diversos testes de robustez, visualização dinâmica dos resultados, e capacidade de executar backtests em múltiplos ativos e estratégias simultaneamente.

1.1 Objetivo Principal

Desenvolver uma plataforma capaz de testar, validar e selecionar estratégias de trading robustas para o mercado financeiro brasileiro, com capacidade de escalar para milhares de estratégias e facilitar sua eventual execução via MetaTrader 5.

1.2 Principais Funcionalidades

- Backtesting de estratégias** com visualização em tempo real
- Execução simultânea** de testes em múltiplos ativos e estratégias

- **Testes de robustez** integrados e automatizados
- **Data mining** para descoberta de novas estratégias
- **Gestão de portfólios** de estratégias
- **Interface gráfica** intuitiva para operação do sistema
- **Integração completa** com MetaTrader 5

1.3 Mercado Alvo e Especificações

1.3.1 Ativos Operados

O sistema é projetado especificamente para o mercado financeiro brasileiro:

Ações:

- Ações componentes do índice Ibovespa (IBOV)
- Referência: B3 - Composição da Carteira do Ibovespa

Mercado Futuro:

- Contratos futuros listados na B3
- Futuros de índices, juros, moedas, commodities e criptoativos

1.3.2 Taxas e Comissões

Modelo realista de custos operacionais do mercado brasileiro:

- Taxas de corretagem (configurável por corretora)
- Emolumentos da B3
- Taxas de registro e liquidação
- Imposto sobre operações financeiras (IOF)
- Slippage estimado por tipo de ativo e liquidez

1.3.3 Horários de Negociação

Sistema respeita os horários oficiais da B3 para cada classe de ativo, com verificações automáticas para garantir operações apenas nos períodos apropriados.

2. Requisitos do Sistema

2.1 Requisitos Funcionais

2.1.1 Motor de Backtesting

- Processamento de dados OHLC em barras de 1 minuto

- Compatibilidade com estratégias MetaTrader
- Suporte para múltiplos ativos simultâneos
- Gerenciamento eficiente de memória
- Cálculos precisos de métricas de desempenho

2.1.2 Visualização e Interface

- Visualização dinâmica de execução de ordens
- Dashboard interativo com Dash/Plotly
- Visualização comparativa entre estratégias
- Exportação de relatórios (PDF, Excel, imagens)

2.1.3 Testes de Robustez

- Teste cego (out-of-sample)
- Dropout aleatório de sinais
- Teste em múltiplos timeframes
- Simulação de Monte Carlo
- Walk-Forward Analysis (WFA)
- Walk-Forward Matrix

2.1.4 Data Mining

- Descoberta automatizada de estratégias
- Validação cruzada robusta
- Prevenção de overfitting
- Categorização automática

2.1.5 Gestão de Portfólio

- Análise de correlação
- Otimização de alocação
- Simulação de portfólios
- Métricas combinadas

2.2 Requisitos Não-Funcionais

2.2.1 Desempenho

- Processar 1 milhão de barras em < 10 segundos
- Backtest de 100 estratégias simultâneas

- Uso eficiente de CPU/memória

2.2.2 Escalabilidade

- Suporte para 1000+ estratégias
- Arquitetura modular extensível
- APIs bem documentadas

2.2.3 Segurança

- Proteção de propriedade intelectual
- Backup automático
- Controle de acesso

2.3 Requisitos Técnicos

Hardware

Mínimo:

- Intel Core i5 ou AMD Ryzen 5
- 16GB RAM
- 256GB SSD

Recomendado:

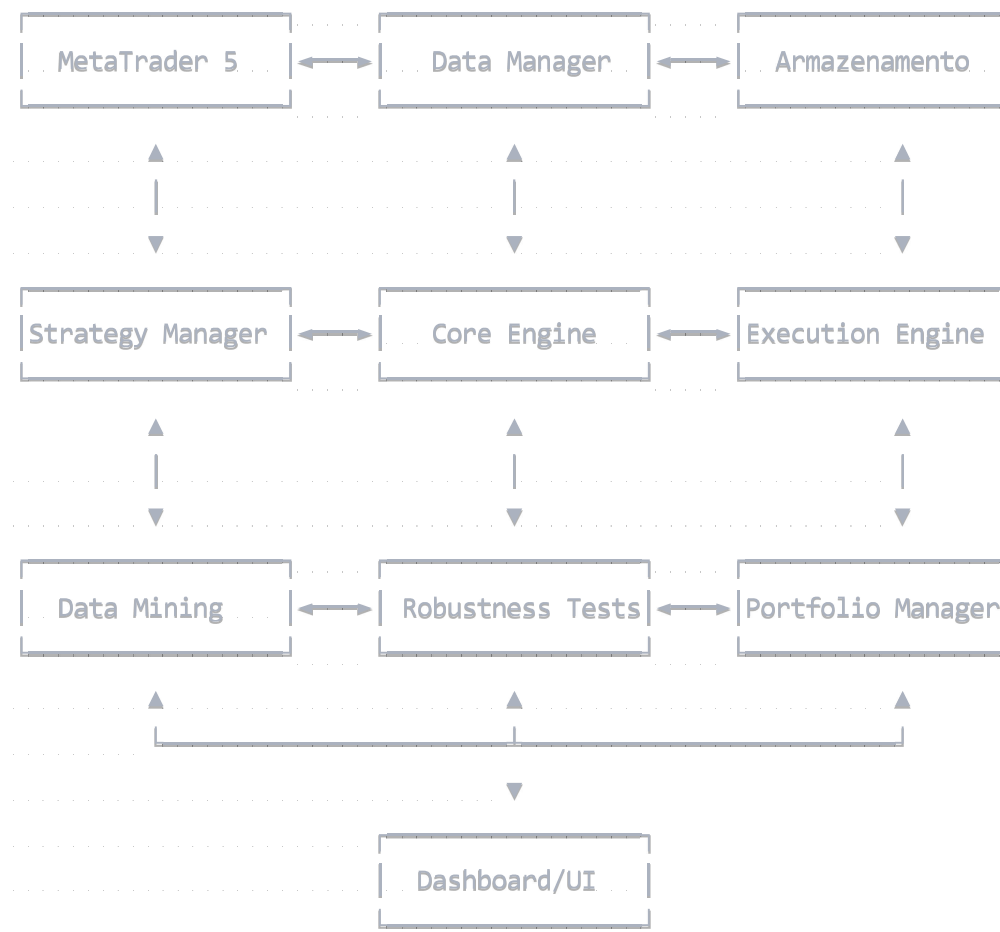
- Intel Core i7/i9 ou AMD Ryzen 7/9
- 32GB+ RAM
- 1TB+ SSD NVMe
- GPU (opcional para aceleração)

Software

- Windows 10/11 (64-bit)
- Python 3.9+
- MetaTrader 5
- VectorBT e dependências

3. Arquitetura do Sistema

3.1 Componentes Principais



3.2 Descrição dos Componentes

1. **Core Engine:** Motor central baseado em VectorBT
2. **Data Manager:** Importação e gestão de dados do MT5
3. **Strategy Manager:** Gestão e versionamento de estratégias
4. **Execution Engine:** Motor de execução de backtests
5. **Robustness Test Suite:** Suite de testes de robustez
6. **Data Mining Engine:** Descoberta de padrões
7. **Portfolio Manager:** Gestão de portfólios
8. **Dashboard:** Interface Dash/Plotly

4. Estrutura de Diretórios

capitalize_system/

```
|
├── core/..... # Núcleo do sistema
│   ├── __init__.py
│   ├── engine.py..... # Motor VectorBT
│   ├── metrics.py..... # Métricas de desempenho
│   ├── events.py..... # Sistema de eventos
│   └── exceptions.py..... # Exceções customizadas
|
├── data/..... # Gerenciamento de dados
│   ├── importers/..... # Importadores MT5, CSV
│   ├── processors/..... # Limpeza e transformação
│   ├── storage/..... # Armazenamento
│   └── providers/..... # Provedores de dados
|
├── strategies/..... # Estratégias de trading
│   ├── base/..... # Classes base
│   ├── trend/..... # Estratégias de tendência
│   ├── mean_reversion/..... # Reversão à média
│   ├── volatility/..... # Baseadas em volatilidade
│   └── custom/..... # Personalizadas
|
├── robustness/..... # Testes de robustez
│   ├── blind_test.py..... # Teste cego
│   ├── dropout.py..... # Dropout de sinais
│   ├── monte_carlo.py..... # Simulação Monte Carlo
│   └── walk_forward.py..... # Walk-Forward Analysis
|
├── visualization/..... # Visualização
│   ├── charts/..... # Componentes de gráficos
│   ├── dashboard/..... # Componentes dashboard
│   └── exporters/..... # Exportação PDF/Excel
|
├── ui/..... # Interface de usuário
│   ├── app.py..... # Aplicação Dash principal
│   ├── layouts/..... # Layouts de páginas
│   └── callbacks/..... # Callbacks Dash
|
├── utils/..... # Utilitários
│   ├── config.py..... # Configuração
│   ├── logging.py..... # Sistema de logging
│   └── validation.py..... # Validação
|
├── tests/..... # Testes unitários
├── docs/..... # Documentação
└── config/..... # Arquivos de configuração
```

5. Roadmap de Desenvolvimento

Fase 1: Fundação (Meses 1-2)

- ☒ 1.1 Configuração do ambiente
- ☒ 1.2 Sistema de logging
- ☒ 1.3 Data Manager básico
- ☒ 1.4 Core Engine básico
- ☒ 1.5 Classes base para estratégias
- ☐ 1.6 Visualização básica
- ☐ 1.7 Integração e testes
- ☐ 1.8 Validação e documentação

Fase 2: Backtesting Completo (Meses 3-4)

- Sistema completo de backtesting
- Visualização dinâmica avançada
- Dashboard com Dash
- Sistema de relatórios

Fase 3: Robustez e Múltiplas Estratégias (Meses 5-7)

- Framework para múltiplas estratégias
- Implementação de todos os testes de robustez
- Otimização de desempenho
- Sistema avançado de relatórios

Fase 4: Portfolio Management (Meses 8-10)

- Gestão completa de portfólios
- Análise de correlação
- Otimização de alocação
- Simulações avançadas

Fase 5: Data Mining (Meses 11-13)

- Motor de data mining
- Validação cruzada robusta
- Prevenção de overfitting

- Interface de configuração

Fase 6: Integração MT5 e Execução (Meses 14-15)

- Integração completa com MT5
- Sistema de execução de ordens
- Monitoramento em tempo real
- Sincronização de dados

Fase 7: Finalização e Deploy (Mês 16)

- Otimização final
 - Documentação completa
 - Testes de aceitação
 - Preparação para produção
-

6. MVPs e Métricas de Sucesso

MVP 1 - Básico (Fases 1-2)

Funcionalidades:

- Backtest simples com uma estratégia
- Visualização básica
- Métricas essenciais

Métricas de Sucesso:

- Processar 100k barras em < 5 segundos
- Gerar gráficos em < 2 segundos
- Zero erros críticos

MVP 2 - Intermediário (Fases 3-4)

Funcionalidades:

- Múltiplas estratégias simultâneas
- Testes de robustez completos
- Gestão básica de portfólio

Métricas de Sucesso:

- 10 estratégias simultâneas
- 5 testes de robustez em < 30 segundos

- Dashboard responsivo

MVP 3 - Completo (Fases 5-7)

Funcionalidades:

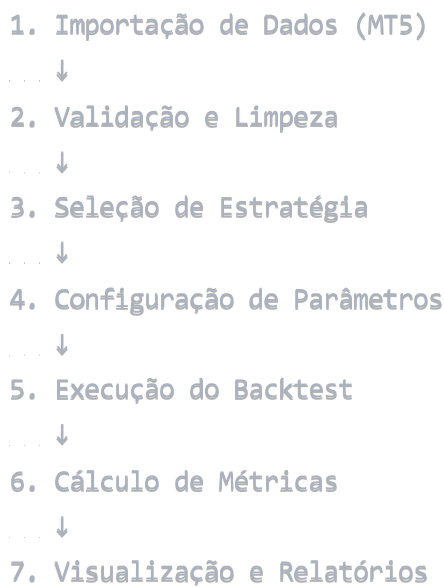
- Data mining funcional
- Execução via MT5
- Sistema completo integrado

Métricas de Sucesso:

- 1000+ estratégias gerenciadas
 - Descoberta de 10+ estratégias/dia
 - Latência execução < 100ms
-

7. Fluxos de Trabalho

7.1 Fluxo de Backtesting



7.2 Fluxo de Robustez

1. Estratégia Validada

... ↓

2. Seleção de Testes

... ↓

3. Execução Paralela

... ↓

4. Consolidação de Resultados

... ↓

5. Score de Robustez

... ↓

6. Decisão (Aprovar/Rejeitar)

8. Gestão de Dados

8.1 Estratégia de Armazenamento

- **Dados de mercado:** Últimos 5 anos em resolução de 1 minuto
- **Resultados:** Todos os backtests por 1 ano
- **Estratégias:** Versionamento completo

8.2 Política de Atualização

- **Dados intraday:** Atualização a cada 5 minutos
- **Dados EOD:** Atualização diária às 19h
- **Limpeza:** Arquivamento mensal de dados antigos

8.3 Backup e Recuperação

- **Backup incremental:** Diário
 - **Backup completo:** Semanal
 - **Teste de recuperação:** Mensal
 - **Retenção:** 3 meses
-

9. Riscos e Estratégias de Mitigação

9.1 Riscos Técnicos

Risco	Impacto	Probabilidade	Mitigação
Limitações do VectorBT	Alto	Média	Desenvolvimento de extensões customizadas
Mudanças na API do MT5	Alto	Baixa	Camada de abstração, testes regulares
Performance inadequada	Médio	Média	Otimização contínua, cache inteligente
Overfitting no data mining	Alto	Alta	Validação cruzada rigorosa

9.2 Riscos de Mercado

Risco	Impacto	Probabilidade	Mitigação
Mudanças regulatórias B3	Alto	Baixa	Arquitetura flexível
Alteração de taxas	Médio	Alta	Sistema configurável
Novos instrumentos	Baixo	Média	Design extensível

10. Integração com MetaTrader

10.1 Importação de Dados

- Conexão via API Python oficial
- Suporte a múltiplos timeframes
- Sincronização automática
- Validação de integridade

10.2 Execução de Ordens

- Framework de envio de sinais
- Confirmação em tempo real
- Log detalhado de execuções
- Gestão de erros robusta

10.3 Monitoramento

- Dashboard de posições abertas
- Alertas configuráveis
- Relatórios de performance
- Sincronização de estado

11. Apêndices

Apêndice A: Glossário

- **Backtest:** Simulação com dados históricos

- **Drawdown:** Queda do pico ao vale
- **Sharpe Ratio:** Retorno ajustado ao risco
- **Slippage:** Diferença entre preço esperado e executado
- **Walk-Forward:** Otimização em janelas deslizantes

Apêndice B: Referências

1. Documentação VectorBT: <https://vectorbt.dev/>
2. API Python MT5: https://www.mql5.com/en/docs/python_metatrader5
3. Dash Documentation: <https://dash.plotly.com/>
4. B3 - Bolsa do Brasil: <https://www.b3.com.br/>

Apêndice C: Contatos

- **Equipe de Desenvolvimento:** dev@capitalizesystem.com
- **Suporte Técnico:** support@capitalizesystem.com
- **Documentação Online:** docs.capitalizesystem.com