

Quick Start - Sistema de Automações

Instalação Rápida

```
# 1. Execute o setup
chmod +x scripts/setup.sh
./scripts/setup.sh

# 2. Configure credenciais
nano .env

# 3. Execute o pipeline
tsx agents/monitor-agent.ts
```

Credenciais Necessárias

Edite o arquivo `.env` e configure:

```
# Manus
MANUS_API_KEY=your_manus_api_key_here
MANUS_USER_ID=your_manus_user_id_here

# Gemini
GEMINI_API_KEY=your_gemini_api_key_here

# Notion
NOTION_API_KEY=your_notion_api_key_here

# Supabase
SUPABASE_URL=your_supabase_url_here
SUPABASE_ANON_KEY=your_supabase_anon_key_here
```

Comandos Principais

Pipeline Completo (Recomendado)

```
# Executar uma vez
tsx agents/monitor-agent.ts

# Modo daemon (a cada 6 horas)
tsx agents/monitor-agent.ts --daemon
```

Automações Individuais

```
# Classificar mensagens
tsx automations/classifier.ts

# Sincronizar com Notion
tsx automations/notion-sync.ts SEU_DATABASE_ID

# Exportar para Obsidian
tsx automations/obsidian-export.ts
```

Agentes Individuais

```
# Classificador (com watch)
tsx agents/classifier-agent.ts --watch

# Extrator de conteúdo
tsx agents/extractor-agent.ts

# Roteador
tsx agents/router-agent.ts

# Analisador de sentimento
tsx agents/sentiment-agent.ts

# Ver mensagens urgentes
tsx agents/sentiment-agent.ts --high-priority
```



Verificar Status

```
# Estatísticas de classificação
tsx automations/classifier.ts

# Estatísticas de sentimento
tsx agents/sentiment-agent.ts

# Status geral
tsx agents/monitor-agent.ts
```



Troubleshooting

Erro: “Cannot find module”

```
# Reinstalar dependências
pnpm install
# ou
npm install
```

Erro: “GEMINI_API_KEY não configurado”

```
# Verificar .env
cat .env | grep GEMINI_API_KEY

# Se vazio, editar:
nano .env
```

Erro de conexão com Supabase

```
# Verificar URL e keys no .env
# Testar conexão:
curl https://ehonimzyiexnfjfbisys.supabase.co
```



Documentação Completa

- **AUTOMATIONS.md**: Documentação detalhada
- **AUTOMATION_TESTS.md**: Relatório de testes
- **.env.example**: Template de configuração



Exemplos de Uso

Cenário 1: Processar novas mensagens

```
# Executar pipeline completo
tsx agents/monitor-agent.ts
```

Cenário 2: Apenas classificar

```
# Classificar mensagens não processadas
tsx automations/classifier.ts
```

Cenário 3: Sincronizar com Notion

```
# Sincronizar prompts
tsx automations/notion-sync.ts SEU_DATABASE_ID
```

Cenário 4: Modo contínuo (produção)

```
# Iniciar daemon (a cada 6 horas)
nohup tsx agents/monitor-agent.ts --daemon > monitor.log 2>&1 &

# Ver logs
tail -f monitor.log

# Parar daemon
pkill -f "monitor-agent"
```

Checklist Pré-Execução

- [] Dependências instaladas (`pnpm install`)
- [] Arquivo `.env` configurado
- [] Credenciais Gemini, Supabase e Notion válidas
- [] Banco Supabase com tabela `messages`
- [] Database Notion criado e compartilhado
- [] Vault Obsidian configurado



Dicas

1. **Primeira execução:** Comece com `tsx agents/monitor-agent.ts` para testar tudo
2. **Desenvolvimento:** Use agentes individuais para testar componentes
3. **Produção:** Use `--daemon` para execução contínua
4. **Monitoramento:** Sempre verifique logs e estatísticas



Suporte

Problemas? Consulte:

1. AUTOMATIONS.md (seção Troubleshooting)
2. Logs de execução
3. Verificar credenciais no `.env`

Última atualização: 18 de Dezembro de 2024