

# Guia de Implementação e Implantação - IA Mestre em Compras

Este guia detalhado explica como implementar e implantar o sistema IA Mestre em Compras para seu grupo de peças e pneus.

## Índice

1. Visão Geral
2. Pré-requisitos
3. Configuração do Google Sheets
4. Configuração do Google Apps Script
5. Preparação do Servidor Web
6. Instalação e Configuração da Aplicação
7. Teste do Sistema
8. Manutenção e Atualização
9. Solução de Problemas

## Visão Geral

O sistema IA Mestre em Compras é composto por:

1. **Backend Google Sheets:** Planilhas contendo os dados de vendas, estoque e compras
2. **API Google Apps Script:** Scripts que processam os dados e implementam os algoritmos de IA
3. **Frontend Web:** Interface de usuário para visualização das análises e recomendações

## Pré-requisitos

- Conta Google com acesso ao Google Sheets e Google Apps Script
- Planilhas do Google organizadas conforme a estrutura especificada:
  - Planilha "Relatório de Vendas IA"
  - Planilha "Estoque por Estabelecimento IA"

- Planilha "NF Entrada\_Produtos IA"
- Servidor web para hospedagem da aplicação frontend (opcional, pode ser utilizado um serviço como GitHub Pages)
- Conhecimentos básicos de HTML, CSS e JavaScript

## Configuração do Google Sheets

### 1. Preparar as Planilhas

Certifique-se de que suas planilhas estão organizadas com as seguintes estruturas:

#### Relatório de Vendas IA:

- Colunas: 

Date Venda	Estabelecimento	Cód Item	Cód Fabricação	Item	Quantidade Total	Faturamento
------------	-----------------	----------	----------------	------	------------------	-------------

#### Estoque por Estabelecimento IA:

- Colunas: 

Linha Itens	Cód Item	Cód Fabricação	Item	AFPNEUS	BNH	FILIAL	06	FL03 CORON	FL04 GUAR.	FL05 ITAP	FL07	MATRIZ	Sum
-------------	----------	----------------	------	---------	-----	--------	----	------------	------------	-----------	------	--------	-----

#### NF Entrada\_Produtos IA:

- Colunas: 

dtalanco	Estabelecimento	N.F (Compras)	Fornecedor	Cód Item	Cód Fabricação	Marca	Item	Qtd Compra	Custo de compra
----------	-----------------	---------------	------------	----------	----------------	-------	------	------------	-----------------

### 2. Configurar Permissões de Acesso

1. Abra cada uma das planilhas no Google Sheets
2. Clique em "Compartilhar" no canto superior direito
3. Configure o acesso para "Qualquer pessoa com o link" com permissão "Leitor"
4. Anote os IDs das planilhas (a parte da URL entre `/d/` e `/edit`)

## Configuração do Google Apps Script

### 1. Criar um Novo Projeto

1. Acesse [script.google.com](https://script.google.com)

2. Clique em "Novo Projeto"
3. Renomeie o projeto para "IA Mestre em Compras - Backend"

## 2. Copiar o Código do Backend

1. Substitua o código gerado automaticamente pelo código do arquivo `google-sheets-integration.js`
2. Atualize as variáveis globais com os IDs das suas planilhas:

javascript

```
const VENDAS_SHEET_ID = 'ID_DA_SUA_PLANILHA_RELATORIO_VENDAS';  
const ESTOQUE_SHEET_ID = 'ID_DA_SUA_PLANILHA_ESTOQUE';  
const NF_ENTRADA_SHEET_ID = 'ID_DA_SUA_PLANILHA_NF_ENTRADA';
```

## 3. Implementar os Algoritmos de IA

1. Crie um novo arquivo no projeto clicando em "+" no menu de arquivos
2. Nomeie o arquivo como "IAAlgorithms.gs"
3. Cole o código do arquivo `ia-algoritmos.js` neste novo arquivo

## 4. Publicar como API Web

1. Clique em "Implantar" > "Nova implantação"
2. Selecione "Aplicativo da Web" como tipo
3. Configure as seguintes opções:
  - Descrição: "API IA Mestre em Compras"
  - Executar como: "Eu mesmo" (sua conta)
  - Quem tem acesso: "Qualquer pessoa"
4. Clique em "Implantar"
5. Anote a URL da API Web que será gerada

## 5. Configurar Gatilhos

1. Clique no ícone de relógio (Gatilhos) no menu lateral

2. Clique em "Adicionar Gatilho"

3. Configure:

- Escolha qual função executar: `gerarRecomendacoes`
- Evento de origem selecionado: "Tempo"
- Selecionar tipo de gatilho baseado no tempo: "Diário"
- Selecionar hora do dia: "Manhã (1-12)"

4. Clique em "Salvar"

## Preparação do Servidor Web

Você pode hospedar a aplicação de diferentes formas:

### Opção 1: GitHub Pages (Gratuito)

1. Crie um repositório no GitHub
2. Suba os arquivos do frontend para o repositório
3. Ative o GitHub Pages nas configurações do repositório

### Opção 2: Servidor Web Próprio

1. Configure um servidor web (Apache, Nginx, etc.)
2. Transfira os arquivos do frontend para o diretório público do servidor

### Opção 3: Serviços de Hospedagem

1. Contrate um serviço de hospedagem web (Netlify, Vercel, etc.)
2. Siga as instruções do provedor para fazer o deploy da aplicação

## Instalação e Configuração da Aplicação

### 1. Preparar os Arquivos do Frontend

1. Baixe os arquivos do frontend (HTML, CSS, JavaScript)
2. Abra o arquivo `google-sheets-connector.js`
3. Atualize a configuração com a URL da sua API:

javascript

```
config: {  
  apiUrl: 'https://script.google.com/macros/s/SEU_ID_DO_SCRIPT_PUBLICADO/exec  
  planilhas: {  
    vendas: 'ID_DA_SUA_PLANILHA_RELATORIO_VENDAS',  
    estoque: 'ID_DA_SUA_PLANILHA_ESTOQUE',  
    nfEntrada: 'ID_DA_SUA_PLANILHA_NF_ENTRADA'  
  }  
}
```

## 2. Personalizar a Interface

Você pode personalizar a interface de acordo com as necessidades específicas do seu negócio:

1. Edite o arquivo HTML principal para ajustar os elementos da interface
2. Modifique o CSS para adaptar cores, fontes e estilos à identidade visual da sua empresa
3. Ajuste os scripts JavaScript para incluir análises específicas do seu negócio

## 3. Implementar Integrações Adicionais (Opcional)

Caso necessite, você pode implementar integrações com outros sistemas:

- Sistema ERP ou PDV
- Plataformas de e-commerce
- Aplicativos móveis

## Teste do Sistema

### 1. Validação do Backend

1. Acesse o URL da API no navegador, adicionando `?action=getVendas` ao final
2. Verifique se os dados são retornados corretamente em formato JSON
3. Teste outras ações como `getEstoque` e `getNFEntrada`

## 2. Teste do Frontend

1. Acesse a aplicação pelo navegador
2. Verifique se os dados estão sendo carregados corretamente
3. Teste todas as funcionalidades:
  - Filtragem de dados
  - Visualização de recomendações
  - Geração de relatórios
  - Navegação entre telas

## 3. Teste de Carga (Opcional)

Se você tiver um grande volume de dados, é recomendável realizar testes de carga:

1. Prepare um conjunto de dados de teste com volume similar ao real
2. Execute operações simultâneas para verificar o desempenho
3. Ajuste os algoritmos ou a infraestrutura se necessário

## Manutenção e Atualização

### 1. Atualização Diária de Dados

Configure processos para garantir que os dados sejam atualizados diariamente:

1. Estabeleça um protocolo para atualização das planilhas
2. Verifique se o gatilho diário do Google Apps Script está funcionando
3. Monitore logs para identificar possíveis erros

### 2. Backup Regular

Implemente uma rotina de backup para proteção dos dados:

1. Utilize a função `realizarBackup()` semanalmente
2. Armazene cópias das planilhas em local seguro
3. Documente todas as configurações para recuperação em caso de falha

### 3. Evolução do Sistema

Planeje melhorias contínuas no sistema:

1. Colete feedback dos usuários
2. Analise o desempenho dos algoritmos de IA
3. Implemente novas funcionalidades conforme necessário

## Solução de Problemas

### Problemas Comuns e Soluções

#### 1. Erro de autorização na API

- Verifique as permissões de acesso das planilhas
- Certifique-se de que a API está publicada corretamente

#### 2. Dados não aparecem na interface

- Verifique se as planilhas estão estruturadas corretamente
- Confira os logs do console do navegador para identificar erros
- Verifique a conexão com a API

#### 3. Algoritmos de IA não estão produzindo resultados precisos

- Verifique se há dados suficientes para análise
- Ajuste os parâmetros de configuração da IA
- Revise a qualidade dos dados de entrada

#### 4. Lentidão no carregamento

- Otimize as consultas à API
- Implemente cache de dados no frontend
- Considere particionar grandes conjuntos de dados

## Suporte Técnico

Em caso de problemas mais complexos:

1. Consulte a documentação completa do sistema

2. Verifique os logs de erro no Google Apps Script e no console do navegador
  3. Entre em contato com o desenvolvedor para suporte especializado
- 

## Recursos Adicionais

- [Documentação do Google Sheets API](#)
  - [Documentação do Google Apps Script](#)
  - [Tutoriais sobre JavaScript](#)
  - [Otimização de Desempenho Web](#)
- 

*Desenvolvido por IA Mestre em Compras - 2025*