

DOCUMENTAÇÃO DO PROJETO DE AUTOMAÇÃO – N8N

Atualização de Cadastro de Clientes em Planilha via API

1. Visão Geral do Projeto

Este projeto tem como objetivo **automatizar a atualização das informações dos clientes** registrados em uma planilha Google Sheets, consumindo dados de uma API externa que fornece os dados atualizados de cada cliente.

A automação é executada no **n8n**, que lê cada linha da planilha, consulta a API e grava os dados atualizados de volta na mesma planilha.

2. Objetivos

- Automatizar o processo de atualização de clientes.
- Reduzir erros manuais no preenchimento da planilha.
- Garantir que os dados consultados via API estejam sempre atualizados.
- Otimizar tempo da operação.

3. Arquitetura da Automação

Entradas

- Planilha contendo os clientes (Macrorregião, Região Edafoclimática, UF, Município e CEP).
- Dados da API (resposta JSON).

Processo

1. n8n lê cada linha da planilha.
2. Filtra apenas as linhas que precisam ser atualizadas.
3. Faz uma requisição HTTP à API passando o identificador do cliente.
4. Recebe os dados atualizados da API.
5. Escreve o novo valor de volta na planilha.

Saídas

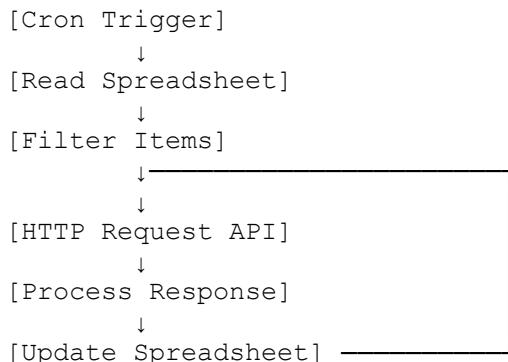
- Planilha atualizada com as novas informações.
- Registro de sucesso ou falha em log (opcional).

4. Fluxo Completo (n8n)

Nós utilizados

1. **Trigger**
 - Define a frequência de execução (ex: a cada 1h / diariamente).
2. **Google Sheets - Read**
 - Lê as informações da planilha.
3. **Node (Filtro)**
 - Identifica quais registros precisam ser atualizados.
 - Exemplo: campo "ano_ultima_atividade" está vazio.
4. **HTTP Request**
 - Faz a chamada para a API usando as variáveis do item:
 - URL: `https://viacep.com.br/ws/{{ $json['CEP'] }}`
5. **Function Node (Organizar Retorno)**
 - Trata o JSON retornado da API.
 - Extrai os campos de interesse.
6. **Google Sheets – Update**
 - Atualiza as colunas específicas da planilha.
7. **Node de Log**
 - Guarda erros ou execuções bem-sucedidas.

5. Estrutura do Fluxo (Pseudo-diagrama)



6. Campos Atualizados

Campo na Planilha	Campo no JSON da API	Descrição
CEP	CEP	CEP Preenchido

7. Requisitos Técnicos

Permissões

- Acesso à API (ViaCEP.).
- Acesso de escrita/leitura na planilha.

8. Exemplo de Configuração de HTTP Request

URL

`https://viacep.com.br/ws/{{ $json['CEP'] }}`

9. Tratamento de Erros

- Caso a API retorne erro, o fluxo grava o registro com *status = erro*.
 - O fluxo pode usar um **IF** para repetir tentativas.
 - Bad requests são enviados para um nó "Erro → Notificação".
-

10. Como Implantar em Produção

1. Testar em uma cópia da planilha.
 2. Usar **Executar Manualmente** no n8n para validar retornos.
 3. Ativar o trigger somente após todos os testes.
-

11. Manutenção do Projeto

- Atualizar o token da API quando expirar.
- Verificar campos novos adicionados pela API.
- Fazer backup semanal da planilha.
- Monitorar falhas no painel de execuções do n8n.