### **Documentação do Projeto: Monitoramento de Sites e Performance**

#### **Objetivo**

Monitorar sites principais da empresa, verificando status online/offline e medindo o tempo de resposta, classificando a latência em "Baixo", "Médio" ou "Alto". A ferramenta também gera logs diários detalhados.

#### **1. Estrutura do Projeto**

* **Linguagem Utilizada:** Python
* **Bibliotecas:**
  + gspread para interação com Google Sheets.
  + requests para monitoramento de sites.
  + datetime e time para registro de tempos.
* **Arquivos Criados:**
  + Planilha no Google Sheets para armazenar registros.
  + Logs diários em uma pasta compartilhada.

#### **2. Funcionalidades**

1. **Monitoramento de Sites:**
   * Verifica se os sites estão online ou offline.
   * Mede o tempo de resposta de cada site.
   * Classifica a latência em três categorias:
     + **Baixo:** ≤ 1 segundo.
     + **Médio:** > 1 e ≤ 3 segundos.
     + **Alto:** > 3 segundos.
2. **Registro de Logs:**
   * Criados arquivos diários com registros detalhados em uma pasta de logs.
3. **Atualização Dinâmica:**
   * Atualiza os status diretamente na planilha do Google Sheets.
   * Registra a "Marca" com base em palavras-chave como "TOTVS" e "FLUIG".
4. **Coluna de Check-in:**
   * Identifica o último registro de cada site com TRUE.
   * Marca registros anteriores como FALSE.

#### **3. Configurações Necessárias**

* **Credenciais JSON:**
  + Arquivo de credenciais do Google API (credenciais\_monitoramento.json) necessário para autenticação.
* **Planilha do Google Sheets:**
  + Planilha configurada com cabeçalhos:
    - Timestamp, Site, Status, Tempo de Resposta, Classificação, Check-in, Marca.
* **Configuração de Logs:**

Defina a pasta de destino para os arquivos de log diários, por exemplo:  
 G:\Drives compartilhados\TI-Integra\Instabilidade Ferramentas da empresa\log

#### **4. Estrutura do Código**

1. **Monitoramento:**
   * Mede o tempo de resposta dos sites usando requests.get.
   * Classifica e registra os resultados na planilha.
2. **Logs:**

Registra cada evento em arquivos diários no formato:  
 monitoramento\_log\_YYYY-MM-DD.txt

1. **Planilha:**
   * Usa gspread para ler e atualizar a planilha.
   * Atualiza dinamicamente as colunas Check-in e Marca.

#### **5. Como Usar**

1. Coloque o arquivo de credenciais JSON na pasta configurada.
2. Insira o URL da planilha do Google Sheets no código.
3. Liste os sites a serem monitorados no array sites.
4. Execute o código Python para iniciar o monitoramento.

#### **6. Observações**

* O código já possui delay integrado para evitar erros de quota no Google Sheets.
* Logs são gerados automaticamente por dia, organizando os resultados em arquivos separados.

Se precisar de mais algum ajuste ou detalhe, é só me avisar!