



Relatório de desenvolvimento de aplicativo de registro de ativos

Luís Felipe Bertuol

1 de setembro de 2023

1 Introdução:

Este relatório descreve o processo de desenvolvimento de um sistema de auxílio para investidores, que tem como objetivo monitorar e alertar sobre oportunidades de compra e venda de ativos na Bolsa de Valores. O projeto foi proposto como parte de um processo seletivo e foi desenvolvido utilizando a linguagem Python com o framework Django, com a adição de funcionalidades extras visando aprimorar a experiência do usuário.

O desenvolvimento do sistema foi dividido em etapas, que incluíram desde a obtenção de dados de cotações até a criação de uma interface web amigável para configuração e acompanhamento dos ativos monitorados.

2 Desenvolvimento:

2.1 Obtenção de Dados de Cotações:

Inicialmente, foi realizada uma pesquisa para encontrar uma API de bolsa de valores adequada. A escolha recaiu sobre a *Alpha Vantage*, que fornece dados financeiros em tempo real e gratuitamente. Um aplicativo simples em Python foi desenvolvido para recuperar os valores de ações de empresas a partir desta API.

2.2 Desenvolvimento da Interface Web:

O aplicativo de auxílio para investidores foi projetado com duas telas principais: a página inicial, "Home" e a página de gerenciamento de ativos, "Meus Ativos". Foi adotado um layout simples e responsivo, utilizando *Bootstrap* para garantir um *layout* um pouco mais agradável, apesar de esse não ser o foco do aplicativo.

2.3 Gerenciamento de Ativos:

Na página "Meus Ativos", os usuários têm a capacidade de adicionar ativos por meio de um campo de *input* que aceita a sigla do ativo (por exemplo, "MSFT" para Microsoft). Os dados dos ativos são armazenados em um banco de dados Django para posterior consulta e rastreamento.

2.4 Personalização de Parâmetros:

Para permitir que os investidores personalizem sua estratégia de investimento, foi incluída a funcionalidade de editar os parâmetros dos ativos. Os parâmetros incluem o limite superior e inferior do túnel de preço, bem como a periodicidade da checagem das cotações.

2.5 Exclusão de Ativos:

Foi implementada uma opção para remover ativos da lista de monitoramento, proporcionando flexibilidade ao usuário para ajustar sua carteira de investimentos conforme necessário.

2.6 Configuração de Email:

Um campo de entrada de email foi incorporado ao sistema, permitindo que os investidores especifiquem o endereço de email onde desejam receber notificações de compra e venda de ativos.

2.7 Ativação/Desativação de Rastreamento:

Para controlar o monitoramento dos ativos, botões de ligar e desligar o rastreamento foram adicionados, possibilitando que os investidores escolham quando desejam receber alertas.

2.8 Finalização da Página Inicial:

Na página "Home", um campo foi disponibilizado para inserir o ativo que o usuário deseja visualizar os dados. Além disso, um gráfico gerado com a biblioteca *Matplotlib* foi incorporado para exibir informações de preço ao longo do tempo de uma maneira mais interessante.

2.9 Integração de Email:

Foi criada uma conta de email dedicada para enviar notificações de compra e venda de ativos e essa funcionalidade foi integrada ao aplicativo para que os alertas sejam enviados de forma automática.

3 Tutorial de Inicialização:

3.1 Download do Projeto:

1. Acesse o repositório do projeto no GitHub através do link fornecido.
2. Clique no botão "Code" e escolha a opção "Download ZIP" para baixar os arquivos do projeto em seu computador.
3. Extraia o arquivo ZIP baixado para obter a pasta com os arquivos do projeto.

3.2 Instalação das Bibliotecas Python:

Para executar o sistema corretamente, é necessário instalar as bibliotecas Python listadas abaixo. Você pode instalá-las utilizando o pip, o gerenciador de pacotes do Python. Abra um terminal ou prompt de comando e execute os seguintes comandos:

- pip install django
- pip install APScheduler
- pip install requests
- pip install matplotlib

3.3 Execução do Sistema:

1. Navegue até a pasta "Aplicativo de Stock INOA".
2. No terminal execute: `python manage.py runserver`.
3. Vá para essa url no navegador: `http://127.0.0.1:8000/`

4 Tutorial de Utilização:

Agora que o sistema está em execução no navegador, você terá acesso às funcionalidades de configuração, rastreamento e acompanhamento de ativos. Siga as etapas a seguir:

4.1 Página Inicial (Home):

Na página inicial, você pode inserir a sigla de um ativo no campo disponível e clicar no botão para visualizar os dados desse ativo em um gráfico como mostra a Figura 1. O sistema apresenta um gráfico de preços que exibe informações sobre o histórico de preços de um ativo ao longo do tempo.

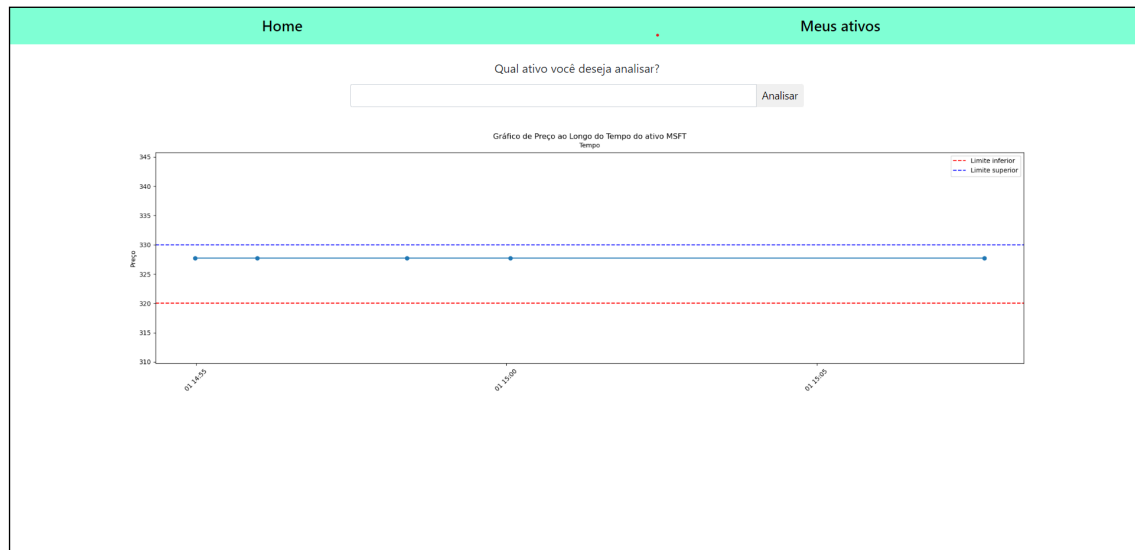


Figura 1: Gráfico mostrando que o valor do ativo "MSFT" não variou no tempo medido

4.2 Página "Meus Ativos":

Na página "Meus Ativos", você pode adicionar ativos da B3, personalizar os parâmetros de rastreamento, configurar o email para notificações e ativar/desativar o rastreamento de ativos como mostra a Figura 2.

A imagem mostra a interface da página "Meus Ativos" de um sistema web. No topo, há uma barra de navegação com "Home" e "Meus ativos". Abaixo, há dois botões: "Ativar rastreador" (verde) e "Desativar rastreador" (vermelho). Segue um campo de texto para "Email de avisos" com o valor "eu@teste.com" e um botão "Salvar". Abaixo, há um campo "Adicionar Ativo" com o exemplo "ex: MSFT" e um botão "Adicionar". A seção principal, intitulada "Meus Ativos", contém duas tabelas de configuração. A primeira tabela é para o ativo "MSFT", com campos para "Tempo de checagem em minutos" (valor 1), "Limite inferior" (320.0), "Limite superior" (330.0), e botões "Deletar Ativo" e "Salvar Ativo". A segunda tabela é para o ativo "IBM", com campos para "Tempo de checagem em minutos" (valor 1), "Limite inferior" (146.0), "Limite superior" (147.0), e botões "Deletar Ativo" e "Salvar Ativo".

Figura 2: Página "Meus Ativos"

4.2.1 Configurações do ativo:

Para as alterações serem registradas, os números devem ser válidos, e caso deseje um valor decimal use o ponto e não a vírgula. Além disso, a barra de minutos aceita apenas números inteiros. Para validar uma alteração, todos os espaços do mesmo ativo devem estar preenchidos.

4.2.2 Configurações de Email:

Certifique-se de configurar um endereço de email válido para receber notificações de compra e venda de ativos.

4.2.3 Ativação/Desativação de Rastreamento:

Use os botões de ligar e desligar o rastreamento para controlar quando deseja receber alertas sobre os ativos monitorados.

5 Conclusão:

Esse desafio foi muito interessante para mim, pois pude aprender um pouco mais como é um projeto com aplicabilidade real. Tive que me dedicar bastante para aprender a lidar com as tecnologias pedidas. Além disso, aprender mais sobre o *framework Django* e outras ferramentas, como envio de email automático e rastreamento do valor de mercado de um ativo foi muito satisfatório. Espero que tenha ficado claro o funcionamento do aplicativo, qualquer dúvida ou *bugs*, por favor me contatar, afinal o aplicativo pode não funcionar corretamente em uma máquina diferente.