

# Portfolio Manager API

O sistema descrito a seguir foi desenvolvido como parte do teste de processo seletivo para a vaga de Engenheiro de Software para a empresa XP Investimentos.

## Como Rodar o Projeto

### Pré-Requisitos

Os seguintes softwares devem ser instalados em sua máquina:

- [.NET SDK 8](#)
- [Docker](#)
- [Git](#)
- [Postman](#)

Além disso, certifique-se de que o docker-compose esteja instalado na máquina (normalmente vem junto da instalação do docker):

```
> docker-compose -v
```

```
C:\Users\LUCAS>docker-compose -v  
Docker Compose version v2.27.0-desktop.2
```

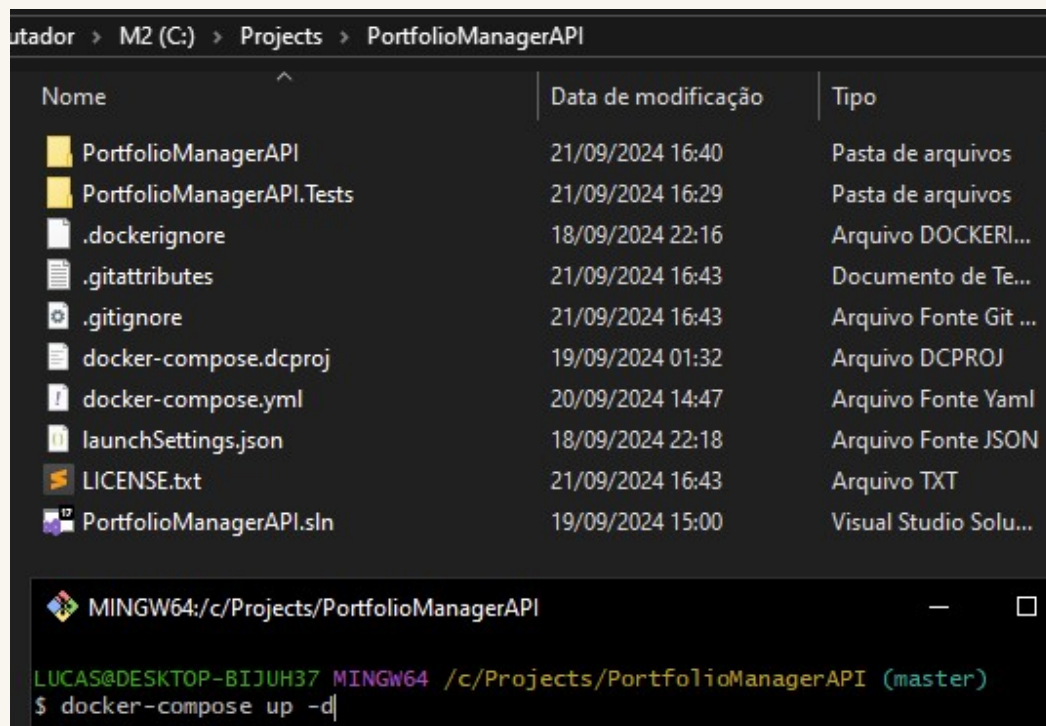
### 1. Clonar o Repositório

```
> git clone https://github.com/L968/PortfolioManagerAPI.git
```

### 2. Subir containers no docker

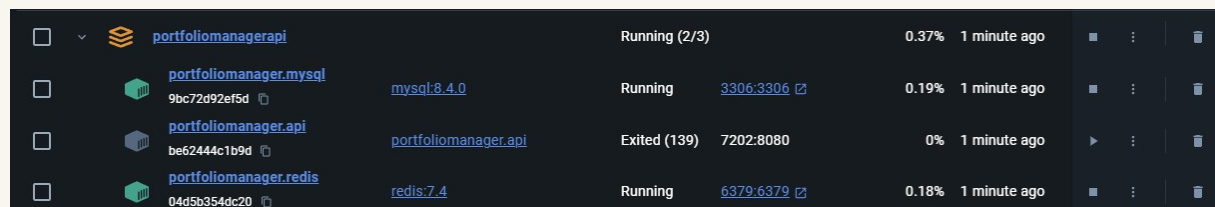
De dentro da pasta raiz da solution, execute o seguinte comando no console:

```
> docker-compose up -d
```



Além do comando subir os containers do MySQL e Redis, também irá construir a imagem da API, porém esta irá inicializar desligada (para não atrapalhar testes no Visual Studio).

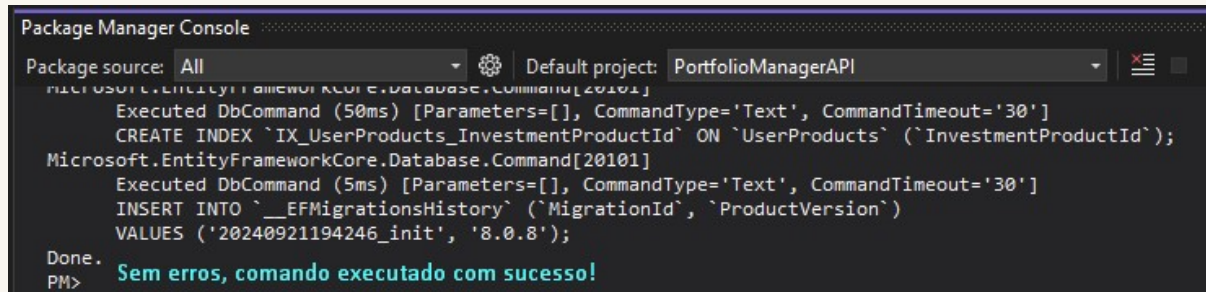
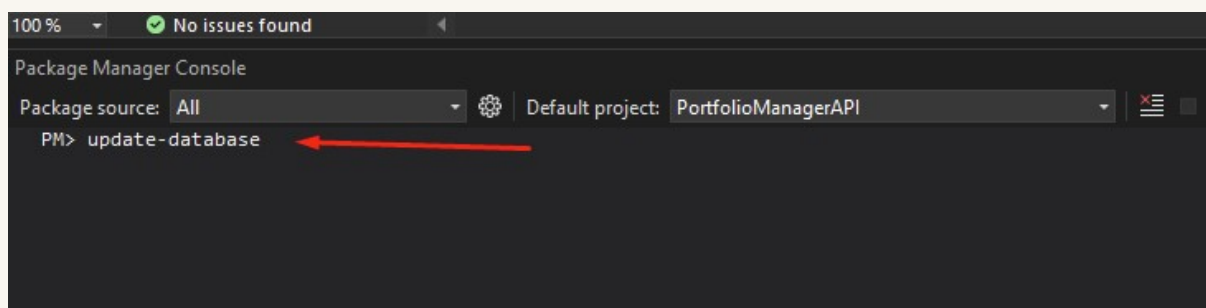
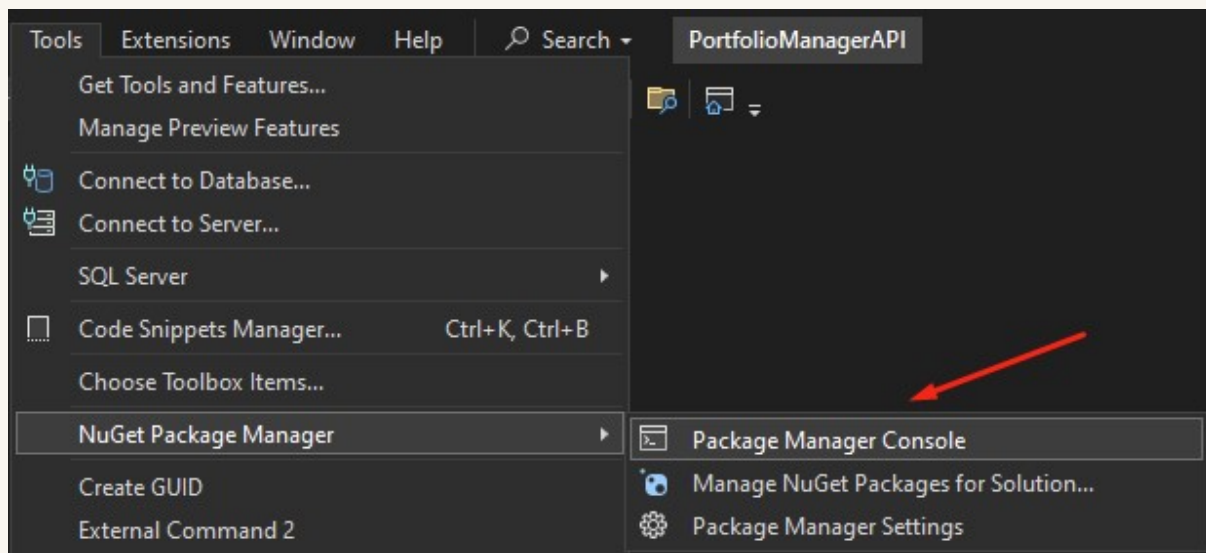
Se tudo ocorrer de acordo, o docker desktop irá se parecer com isso:



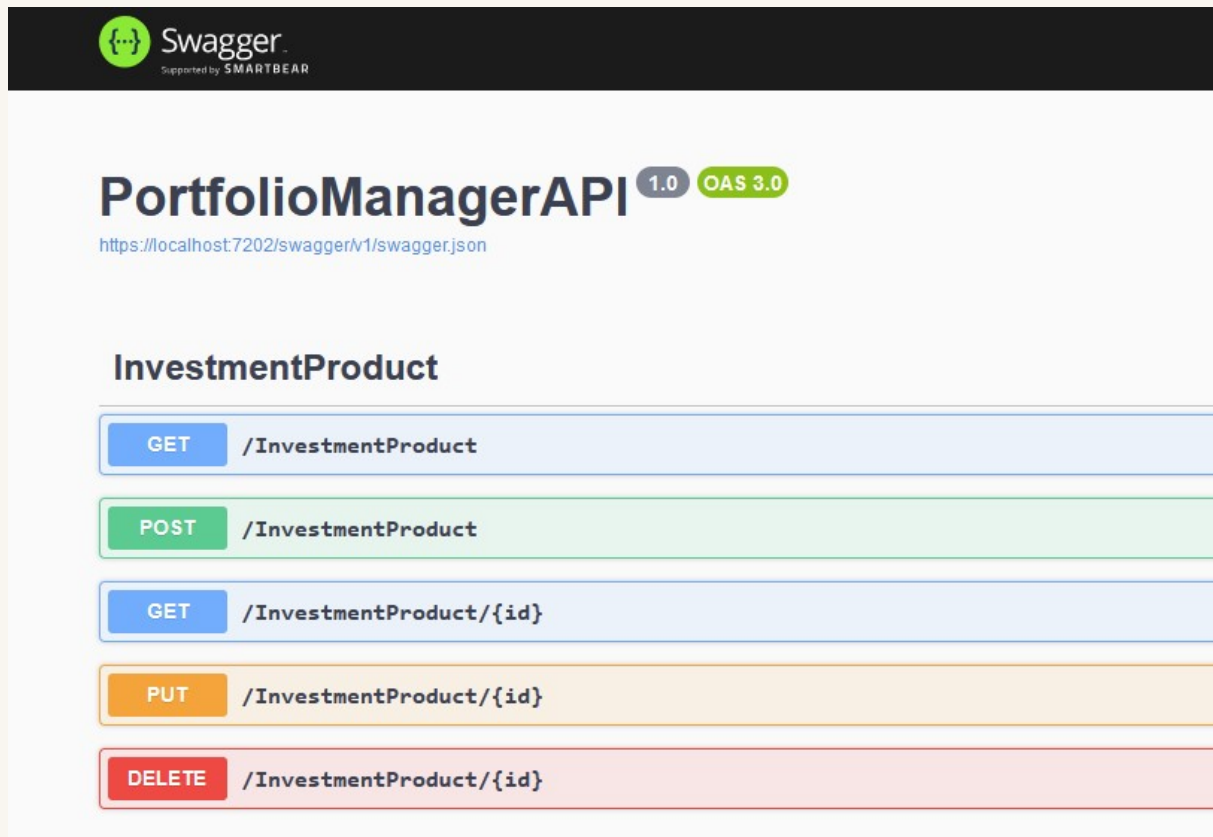
### 3. Executar as migrations

Abra o Visual Studio, e execute as migrations com o seguinte comando:

```
> update-database
```



## 4. Execute o projeto



## Como utilizar o projeto

É recomendável utilizar o postman para executar as requisições (As seguintes explicações serão feitas com o uso dele). Um arquivo de collection do postman foi enviado em anexo junto à este documento.

## Autenticação e Autorização

Todas as rotas necessitam de um token JWT para serem executadas, garantindo a autenticação, e autorização com base na role (perfil). Neste sistema, foram criadas 2 roles:

1. Admin
2. Regular

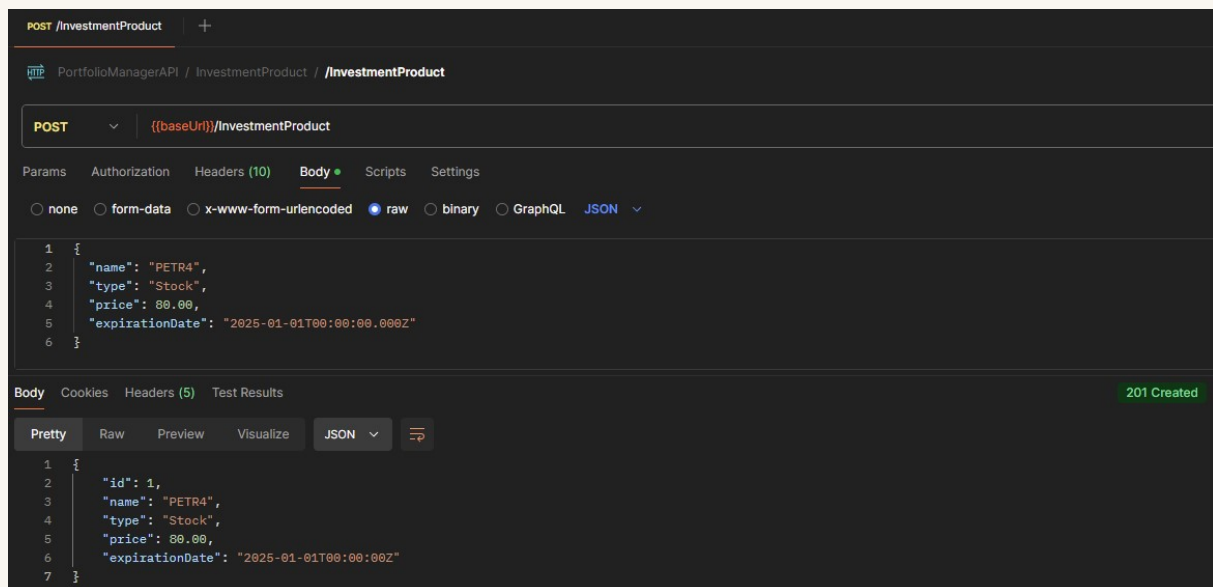


Todas as rotas do controller **User** utilizam o perfil “**regular**”, e todas as rotas de “**InvestmentProduct**” utilizam o token “**admin**”, **com exceção da listagem de todos os produtos em “GET /InvestmentProduct”**

Ao executar a requisição do token, o mesmo será salvo automaticamente pelo postman para ser usado nas outras requisições.

## Criando um produto de investimento

Dentro do controller “InvestmentProduct”, e com o token de admin gerado, execute a seguinte requisição. Dentro desse controller, temos todas as operações básicas de um CRUD.



Existem 2 Types:

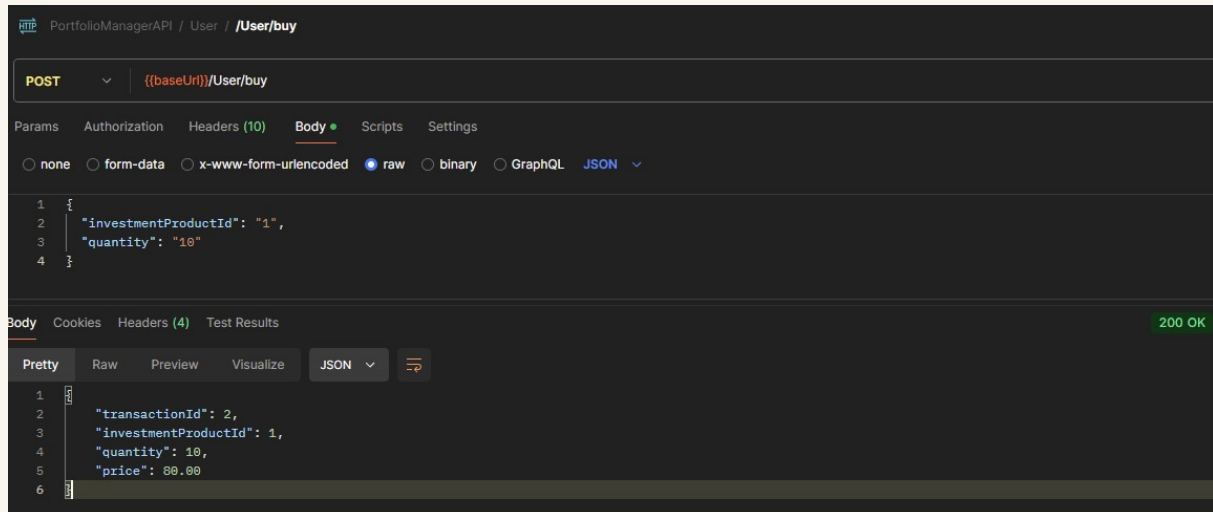
1. Stock
2. Fund

Porém eles não têm valor lógico no código, e são apenas para questões de apresentação.

# Comprando/Vendendo um produto de investimento

Depois de criar o produto, gere o token com a role “regular”, e execute a seguinte requisição para comprar um produto de investimento. Lembre-se de informar o “InvestmentProductId” que foi criado anteriormente.

A mesma lógica se aplica à venda.



# Job de envio de email

Foi utilizada a biblioteca **Hangfire** para este desenvolvimento. Para monitorar a job, execute a aplicação e acesse a seguinte página:

<https://localhost:7202/hangfire>