

# **API DOCUMENTATION**

# Currency

- localhost:8080/currency/info/{id} - [GET]

Obter as informações sobre uma determinada criptomoeda a partir do ID, como por exemplo, nome, symbol

- localhost:8080/currency - [GET]

Obter um pageable com todas as criptomoedas

- localhost:8080/currency/{id} - [GET]

Obter um pageable com todos os markets de uma determinada criptomoeda a partir do seu ID

- localhost:8080/currency/search - [GET]

Obter todas as moedas que contenham no nome um determinado texto, passado como RequestParam.

**Nota:** Endpoint utilizado na barra de pesquisa

- localhost:8080/currency - [POST]

Cria uma criptomoeda.

- localhost:8080/currency/del/{id} - [DELETE]

Remove uma criptomoeda com o respectivo id.

# Users

- localhost:8080/user - [GET]

Obter um pageable todos os utilizadores

- localhost:8080/user - [POST]

Cria um utilizador.

- localhost:8080/user/{id} - [DELETE]

Remove um utilizador com o respetivo id.

# Extensions

- localhost:8080/extension - [GET]

Obtém um pageable de todas as extensões disponíveis.

- localhost:8080/extensions - [GET]

Obtém uma lista de todas as extensões disponíveis. Difere do endpoint apresentado anteriormente porque precisamos de formas diferentes de apresentar a mesma informação.

- localhost:8080/extension/{id} - [GET]

Obtém um pageable de todas as extensões de um determinado user, a partir do seu *id*.

- localhost:8080/extension - [POST]

Criação de uma extensão, precisa do id do user, bem como do nome, descrição e *path* para o endpoint da extensão.

- localhost:8080/extension/{id} - [DELETE]

Remove uma extensão com o id apresentado.

## Portfolio

- localhost:8080/portfolio - [GET]

Obtém a lista de portfólios de um determinado utilizador. O id do respetivo utilizador é passado como RequestParam.

- localhost:8080/portfolio/{id} - [DELETE]

Remove um portfólio de um determinado utilizador. O id do portfólio é passado no url e o userID é recebido por RequestParam.

- localhost:8080/portfolio - [POST]

Criar um portfólio. O parâmetro necessário para criar um portfólio é apenas um nome, que é passado como RequestParam. Para além disto, associa este novo portfólio ao utilizador que o criou, portanto o userID também é passado como RequestParam.

- localhost:8080/portfolio/join - [POST]

Adiciona um utilizador a um portfólio a partir de uma public key. A public key e o userID são passados como RequestParam.

- localhost:8080/portfolio/{id} - [GET]

Obter um portfólio através do um id.

- localhost:8080/portfolio/{id}/details - [GET]

Obter detalhes de um portfólio através do um id.

- localhost:8080/portfolio/{id}/transactions - [GET]

Retorna um pageable de todas as transações de um determinado portfólio.

- localhost:8080/portfolio/users - [POST]

Retorna a lista de utilizadores pertencentes a um determinado portfólio, a partir da public key providenciada.

- localhost:8080/portfolio/extension/{id} - [GET]

Obter uma lista das extensões de um determinado portfólio através de um id.

- localhost:8080/portfolio/extension - [POST]

Associa uma extensão a um portfólio. O parâmetro id serve de identificação do portfólio e o parâmetro path identifica a extensão.

- localhost:8080/portfolio/extensions - [DELETE]

Remove uma extensão que esteja associada a um portfólio. O parâmetro id serve de identificação do portfólio e o parâmetro path identifica a extensão.

## Market

- localhost:8080/market - [POST]

Cria um mercado.

- localhost:8080/market - [GET]

Retorna um pageable de todos os markets disponíveis. Permitindo, também, o sorting de atributos de um determinado market (id, preço, moeda de origem...) de modo ascendente ou descendente.

- localhost:8080/market/{id} - [GET]

Identifica todos os tickers de um determinado mercado, identificado pelo seu *id*, e fornece os dados necessários para a criação do gráfico de um mercado.

- localhost:8080/market/{id}/orders/sell - [GET]

Retorna um pageable de todas as ordens de venda relativos a um determinado mercado, identificado pelo seu *id*.

- localhost:8080/market/{id}/orders/buy - [GET]

Retorna um pageable de todas as ordens de compra relativos a um determinado mercado, identificado pelo seu *id*.

## Order

- localhost:8080/order - [GET]

Retorna um pageable de todas as ordens, independentemente do seu tipo.

- localhost:8080/order - [POST]

Cria uma ordem a partir do id de um mercado, do id de um portfólio, da quantidade e do valor da ordem providenciados. Caso a quantidade seja igual a zero, a ordem é inválida. Por outro lado, se a quantidade oferecida for superior a zero, significa que o utilizador pretende fazer uma ordem de compra da moeda de

destino. Caso contrário, se a quantidade oferecida for inferior a zero, o utilizador pretende fazer uma ordem de venda da moeda de destino. O sistema verifica se o portfólio possui a moeda necessária para realizar a ordem e se as quantidades introduzidas não excedem aquelas que o portfólio possui. Caso isto se verifique, uma ordem de compra ou de venda é realizada.

- localhost:8080/order/{id} - [DELETE]

Remove uma ordem a partir do seu id.

## Transaction

- localhost:8080/transaction - [GET]

Retorna um pageable de todas as transações realizadas.

- localhost:8080/transaction - [POST]

Cria uma transação.

- localhost:8080/transaction/{id} - [DELETE]

Remove uma transação a partir do *id* providenciado.

## Ticker

- localhost:8080/ticker - [GET]

Retorna um pageable de todos os tickers.

- localhost:8080/ticker - [POST]

Cria um ticker.

- localhost:8080/ticker/{id} - [DELETE]

Remove um ticker a partir do *id* providenciado.

## Authentication

- localhost:8080/login - [GET]

Os parâmetros verificam se o utilizador existe e em caso afirmativo, autêntica-o

- localhost:8080/signup - [POST]

Aceita um User no parâmetro e caso o email não esteja em uso, adiciona-o.