

Investment

Uma API para controle de investimento, utilizando Docker, Docker compose, node, mysql e express

--- Início dos comandos ---

Comando para clonar o repositório:

```
git clone https://github.com/ViniciusCaique/docker-express-api.git
```

Comando para instalar as dependencias do projeto:

```
npm install
```

Comando para iniciar o docker compose junto com a imagem:

```
docker-compose up -d --build
```

-h: localhost, -P: porta, --protocol=tcp: é opcional e habilita aplicações trocarem mensagem com a network, -u: usuario, -p: passwordd, não é bom informar na linha de código.

```
mysql -h localhost -P 3306 --protocol=tcp -u {username} -p
```

Entrar como root

```
create user 'gojo'@'localhost' IDENTIFIED BY 'adm123';
```

Dar privilégio para o usuario

```
grant all privileges on . to 'gojo'@'localhost' with grant option;
```

Exit

Entrar na conta do usuário

```
mysql -u gojo -p
```

Criar a tabela de usuario

```
CREATE TABLE usuarios (  
    id_user INT NOT NULL AUTO_INCREMENT,  
    name VARCHAR(255) NOT NULL,  
    email VARCHAR(255) NOT NULL,  
    password VARCHAR(255) NOT NULL,
```

```
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
PRIMARY KEY (id_user),  
CONSTRAINT UQ_Usuarios_Email UNIQUE (email)  
);
```

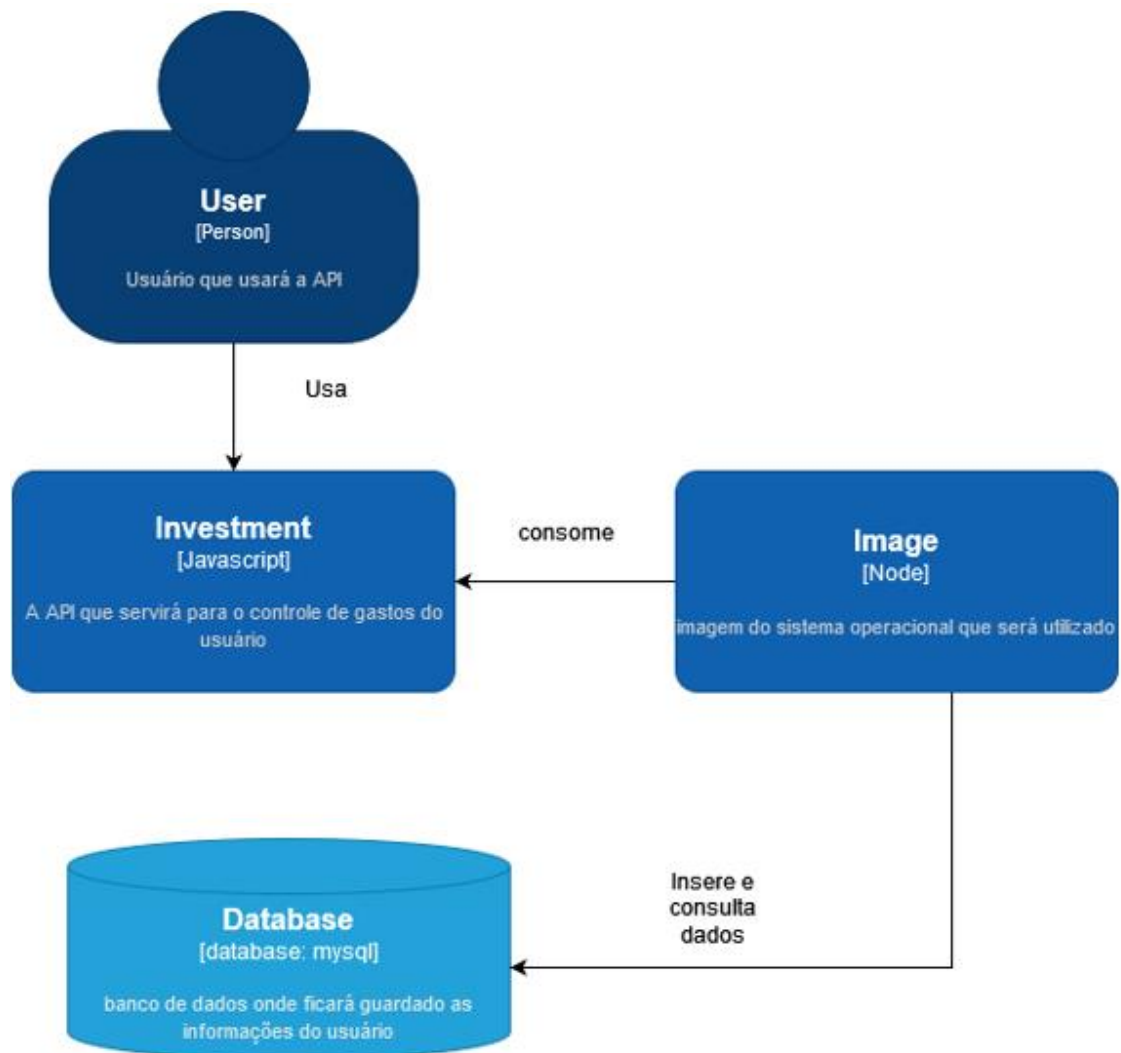
Exit

Encerra o docker compose:

docker-compose down

--- Fim dos comandos ---

Arquitetura do projeto:



Documentação da API

Endpoints

- Usuario
 - Cadastrar
 - Atualizar
 - Listar todos
 - Mostrar detalhes
 - Apagar

Cadastrar usuario

`POST` /api/usuario

campo	tipo	Obrigatório	Descrição
-----	-----	:-----:	-----
nome	text	sim	Campo para o usuario informa seu nome
email	text	sim	Campo onde o usuario informa o email
password	text	sim	Campo reservado para a senha do usuario

Exemplo de corpo de requisição

```
```js
{
 "nome": "Fernando",
 "email": "Fernando@gmail.com",
 "password": "jP5@o37ZKxd3"
}
```

### Exemplo de corpo da resposta

### Códigos de resposta

código	descrição
-   -	
201	dados cadastrado com sucesso
400	os campos enviados são inválidos

Atualizar usuario

``PUT`/api/usuario/{id}`

Exemplo de corpo de requisição

```
``js
{
 "nome": "Cesar",
 "email": "Cesar@gmail.com",
 "password": "jP5@o37ZKxd3"
}
...

```

Códigos de resposta

| código | descrição

| - | -

| 201 | dados cadastrado com sucesso

| 400 | os campos enviados são inválidos

Listar todos

``GET` /api/usuario`

```
``js
{
 "nome": "Fernando",
 "email": "Fernando@gmail.com",
 "password": "jP5@o37ZKxd3"
}
{
 "nome": "Cesar",
 "email": "Cesar@gmail.com",

```

```
"password": "X1z12Z@Bprd3"
}
...
```

#### Códigos de resposta

| código | descrição

| - | -

| 200 | dados retornados com sucesso

| 402 | não existe usuário cadastrado

#### Detalhar usuário

```
`GET` /api/usuario/{id}
```

```
``js
```

```
{
 "id": 1,
 "nome": "Fernando",
 "email": "Fernando@gmail.com",
 "password": "jP5@o37ZKxd3"} ``
```

#### Códigos de resposta

| código | descrição

| - | -

| 200 | dados retornados com sucesso

| 402 | não existe usuário cadastrada

#### Apagar usuário

```
`DELETE` /api/usuario/{id}
```

| código | descrição

| - | -

| 204 | dado apagado com sucesso

| 401 | não existe dado com o id informado