

Project Application

Project of application which is responsible for handling, filtering and caching data from API. Project is dockerized with Docker-Compose file configuration to run and manage both application and MongoDB container. Application also provides code testing.

Project Stack

- **API – REST**
- **Database - MongoDB**
- **Typescript**
- **NestJS**
- **Jest**
- **Docker**
- **Docker Compose**

Run project with docker

To build and run project with Docker run command: 'docker-compose up --build'

To run project: 'docker-compose up -d'

To check container logs: 'docker-compose logs -f'

To stop containers: 'docker-compose down'

Run Tests

To run tests use command: 'npm run test'. Bear in mind that project should be already working on docker containers before running a command.

Code guide

Node version 20.

Build project with Docker

.env file is required and should be placed into root folder.

.env example file

In order to start a project, you must place environment variables into project root folder

```
DATABASE_PORT=27017:27017
LOCAL_DB_PORT=27017
LOCAL_APP_PORT=3000
APP_PORT=3000:3000
DATABASE_NAME=mongo
URL=https://swapi.dev/api
```

Requests

URL	Description	Response status
GET /films	Returns all films from films category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /films/:id	Returns single film from films category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent.	200
GET /filmsByParams	Return array of word counts from film descriptions	200
GET /mostCommonPerson	Return a name of a character from the API that appears the most often across all of the openings of the film	200
GET /films/?page=:number	Returns the number of films according to pagination. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /people	Returns all people from people category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and	200

	request to another API is not being sent	
GET /person/:id	Returns single person from people category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent.	200
GET /people/?page=:number	Returns the number of people according to pagination. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /planets	Returns all planets from planets category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /planets/:id	Returns single planet from planets category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent.	200
GET /planets/?page=:number	Returns the number of planets according to pagination. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200

GET /species	Returns all species from species category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /species/:id	Returns single species from species category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent.	200
GET /species/?page=:number	Returns the number of species according to pagination. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /starships	Returns all starships from starships category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /starships/:id	Returns single starship from starships category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent.	200
GET /starship/?page=:number	Returns the number of starships according to pagination. Sends request to other API and cache data into db for	200

	24h. If data is already stored in db, data is taken from db and request to another API is not being sent	
GET /vehicles	Returns all vehicles from vehicles category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200
GET /vehicles/:id	Returns single vehicle from vehicles category. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent.	200
GET /vehicles/?page=:number	Returns the number of vehicles according to pagination. Sends request to other API and cache data into db for 24h. If data is already stored in db, data is taken from db and request to another API is not being sent	200