

Criação dos Modelos para o MongoDB e criação de endpoints



DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

AGENDA

- Finalização da configuração Flask com PyMongo
- Criação de Schemas de Validação e criação de Collection dinâmica
- Criação de endpoints com conexão ao MongoDB

Finalização da configuração Flask com PyMongo

Criar a pasta `utils` e dentro dela, criar o arquivo `__init__.py`, com o código abaixo:

```
from flask_pymongo import PyMongo
```

```
mongo = PyMongo()
```

Finalização da configuração Flask com PyMongo

Atualizar o arquivo `src/app/__init__.py`, com o código abaixo:

```
import os

from flask import Flask

from src.app.config import app_config

from src.app.swagger import create_swagger

from flask_cors import CORS

from src.app.utils import mongo

app = Flask(__name__)

app.config.from_object(app_config[os.getenv("FLASK_ENV")])

create_swagger(app)

mongo.init_app(app)

mongo_client = mongo.db

CORS(app)
```

Finalização da configuração Flask com PyMongo

Atualizar o arquivo `app.py`, com o código abaixo:

```
from src.app import app
```

```
from src.app.routes import routes
```

```
routes(app)
```

```
if __name__ == "__main__":
```

```
    app.run()
```

Criação de Schemas de Validação e criação de Collection dinâmica

Criar dentro da pasta models, o arquivo movies.py, com o código abaixo:

```
def create_collection_movies(mongo_client):  
    movies_validator = {  
        "$jsonSchema": {  
            "bsonType": "object",  
            "properties": {  
                "type": {  
                    "bsonType": "string",  
                    "description": "Se é um filme ou uma série",  
                    "enum": ["Movie", "TV Show"]  
                },  
            },  
        },  
    }
```

Criação de Schemas de Validação e criação de Collection dinâmica

```
"title": {  
  "bsonType": "string",  
  "description": "O título",  
},  
"release_year": {  
  "bsonType": "int",  
  "description": "Ano de lançamento do filme"  
},  
"duration": {  
  "bsonType": "string",  
  "description": "Duração",  
},
```

Criação de Schemas de Validação e criação de Collection dinâmica

```
"description": {  
  "bsonType": "string",  
  "description": "Descrição sobre o filme",  
},  
"cast": {  
  "bsonType": "array",  
  "description": "Elenco do filme",  
  "items": {  
    "bsonType": "string",  
    "description": "Pessoa do filme",  
  }  
}  
},  
}}
```


Criação de Schemas de Validação e criação de Collection dinâmica

```
try:  
    mongo_client.create_collection("movies")  
except Exception as e:  
    print(e)  
  
mongo_client.command("collMod", "movies", validator=movies_validator)
```

Criação de Schemas de Validação e criação de Collection dinâmica

Criar dentro da pasta models, o arquivo cast.py, com o código abaixo:

```
def create_collection_cast(mongo_client):  
    cast_validator = {  
        "$jsonSchema": {  
            "bsonType": "object",  
            "required": ["full_name"],  
            "properties": {  
                "full_name": {  
                    "bsonType": "string",  
                    "description": "O nome da pessoa do elenco"  
                },  
            },  
        },  
    }  
}
```

Criação de Schemas de Validação e criação de Collection dinâmica

```
try:
```

```
    mongo_client.create_collection("cast")
```

```
except Exception as e:
```

```
    print(e)
```

```
mongo_client.command("collMod", "cast", validator=cast_validator)
```