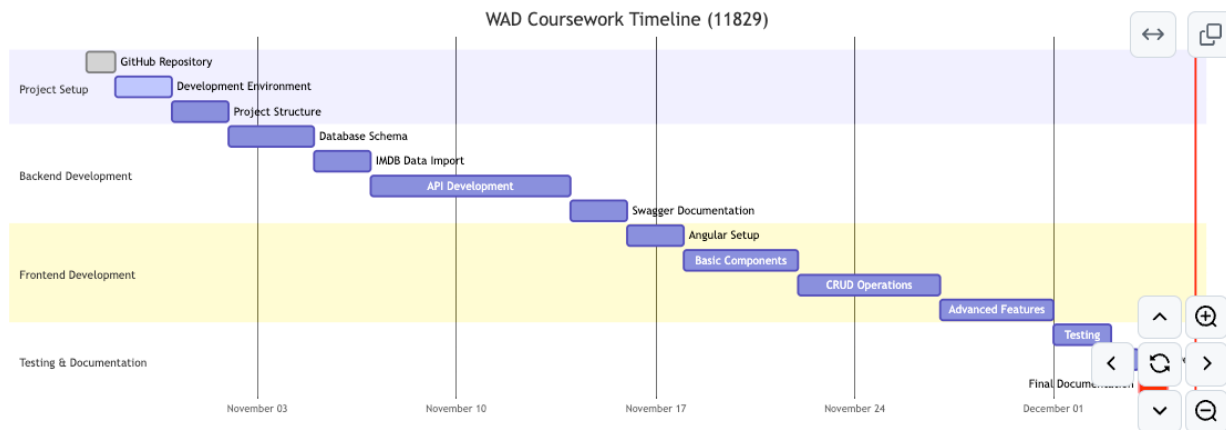


This document is a coursework report for the Web Applications Development (WAD) module by the student with ID 00011829 for Semester 1 at Westminster International University in Tashkent (WIUT).

GitHub repository link: 00011829

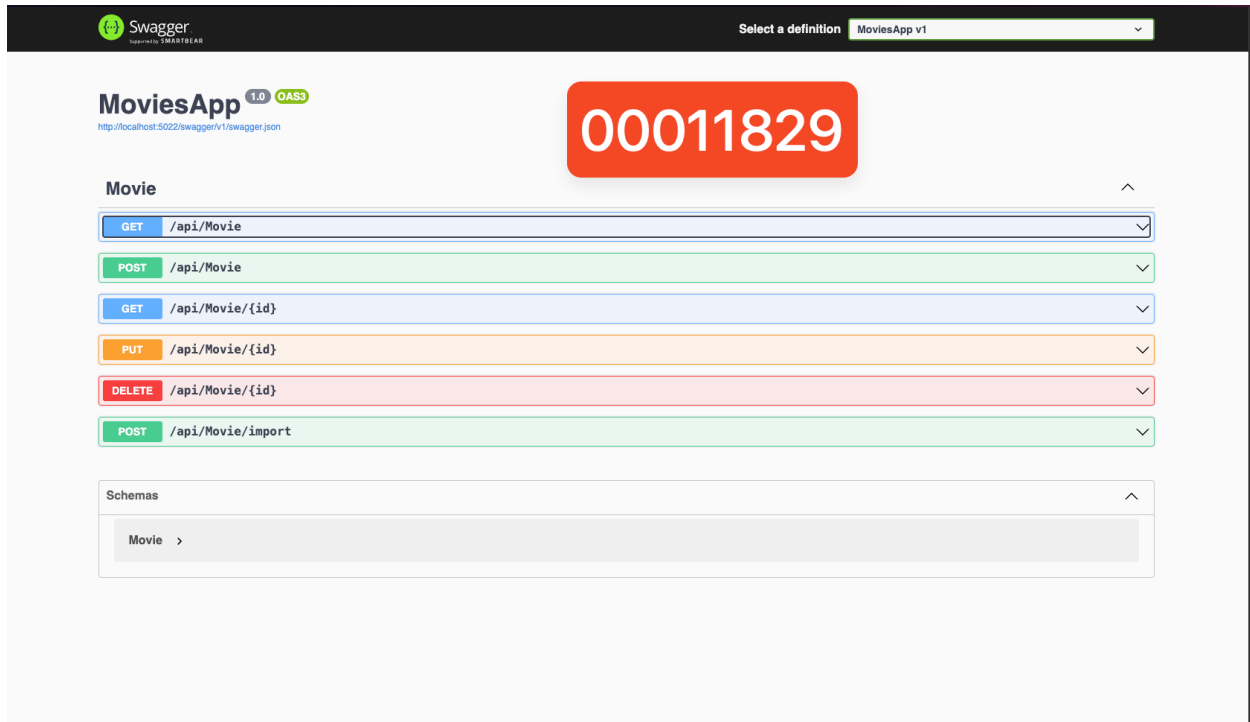
Development plan:



(Of course, I couldn't follow the timeline, but I managed to create this chart at the start of November, and can be seen in the coursework's GitHub repository).

## API (Backend) screenshots

Here is the screenshot of my Swagger UI after everything was set up successfully and no endpoints were run yet:



Since my app is a movies app, I decided to use [the IMDB top 1000 movies dataset from Kaggle](#). It was in a CSV format, so I wrote a quick python script to convert it into a desirable JSON format (suitable for my application, see scripts/csv\_to\_json.py file). Then, I created the necessary components to enable the “Import” endpoint in the API. The below screenshot shows the swagger Ui state after I click on “Try it out” and then “Execute”:

POST

/api/Movie/import

11829

Cancel

Parameters

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'POST' \
  'http://localhost:5022/api/Movie/import' \
  -H 'accept: */*' \
  -d ''
```

Request URL

```
http://localhost:5022/api/Movie/import
```

Server response

Code	Details
200	<div>Response body</div> <div>Movies imported successfully</div> <div>Download</div> <div>Response headers</div> <div><pre>content-type: text/plain; charset=utf-8 date: Thu, 05 Dec 2024 18:09:35 GMT server: Kestrel transfer-encoding: chunked</pre></div>

Responses

Code	Description	Links
200	Success	No links

As you can see, I got a success message telling me the movies were imported. So, let's check out one of the movies (this is the GET endpoint before running it):

MoviesApp 1.0 OAS3

http://localhost:5022/swagger/v1/swagger.json

11829

Movie

GET

/api/Movie

Try it out

Parameters

No parameters

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

Example Value

Schema

```
[
  {
    "id": 0,
    "title": "string",
    "year": 0,
    "genre": "string",
    "rating": 10,
    "description": "string"
  }
]
```

Here is the same endpoint after running it:

The screenshot shows a REST client interface with the following sections:

- Movie** (Title bar)
- GET /api/Movie** (Method and URL bar)
- Parameters** (Tab): No parameters. A red box with the number **11829** is overlaid on this section.
- Execute** (Button)
- Clear** (Button)
- Responses** (Section header)
- Curl** (Text area):

```
curl -X 'GET' \  
  'http://localhost:5022/api/Movie' \  
  -H 'accept: text/plain'
```
- Request URL** (Text area): `http://localhost:5022/api/Movie`
- Server response** (Section header)
- Code** (Tab):
  - 200** (Status)
  - Response body** (Text area):

```
{  
  "id": 1,  
  "title": "The Shawshank Redemption",  
  "year": 1994,  
  "genre": "Drama",  
  "rating": 9.3,  
  "description": "Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency."  
},  
{  
  "id": 2,  
  "title": "The Godfather",  
  "year": 1972,  
  "genre": "Crime, Drama",  
  "rating": 9.2,  
  "description": "An organized crime dynasty's aging patriarch transfers control of his clandestine empire to his reluctant son."  
},  
{  
  "id": 3,  
  "title": "The Dark Knight",  
  "year": 2008,  
  "genre": "Action, Crime, Drama"
```

As you can see, the imported movies are present. Now, I will try to add a new movie with a POST request:

MoviesApp1.0OAS3

http://localhost:5022/swagger/v1/swagger.json

11829

Movie

GET/api/Movie

POST/api/Movie

Parameters

No parameters

Request body

application/json

Example Value | Schema

```
{
  "id": 0,
  "title": "string",
  "year": 0,
  "genre": "string",
  "rating": 10,
  "description": "string"
}
```

Responses

Code	Description	Links
200	Success	No links

After:

**POST** /api/Movie

Parameters Cancel Reset

No parameters

Request body application/json

```
{
  "id": 1001,
  "title": "Extraction 2",
  "year": 2023,
  "genre": "Action",
  "rating": 7,
  "description": "Back from the brink of death, commando Tyler Rake embarks on a dangerous mission to save a ruthless gangster's imprisoned family."
}
```

**11829**

Execute Clear

Responses

Curl

```
curl -X 'POST' \
  'http://localhost:5022/api/Movie' \
  -H 'accept: text/plain' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 1001,
    "title": "Extraction 2",
    "year": 2023,
    "genre": "Action",
    "rating": 7,
    "description": "Back from the brink of death, commando Tyler Rake embarks on a dangerous mission to save a ruthless gangster's imprisoned family."
  }'
```

Request URL

<http://localhost:5022/api/Movie>

Server response

Code Details

201 Undocumented

Response body

```
{
  "id": 1001,
  "title": "Extraction 2",
  "year": 2023,
  "genre": "Action",
  "rating": 7,
  "description": "Back from the brink of death, commando Tyler Rake embarks on a dangerous mission to save a ruthless gangster's imprisoned family."
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Thu, 05 Dec 2024 18:28:21 GMT
location: http://localhost:5022/api/Movie/1001
server: Kestrel
transfer-encoding: chunked
```

Responses

Code	Description	Links
------	-------------	-------

Now, I will try to get a movie using its ID. Let me try the one I just added with id 1001:

Before:

GET

/api/Movie/{id}

^

Parameters

Cancel

Name	Description
id <small>required</small>	
integer(\$int32)	
(path)	

1001

11829

Execute

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header

Example Value | Schema

```
{
  "id": 0,
  "title": "string",
  "year": 0,
  "genre": "string",
  "rating": 10,
  "description": "string"
}
```

After:

GET

/api/Movie/{id}

Parameters

Cancel

Name	Description
id <span>required</span>	
integer(int32)	1001
(path)	

Execute

Clear

Responses

Curl

curl -X 'GET' \  
'http://localhost:5022/api/Movie/1001' \  
-H 'accept: text/plain'

Request URL

http://localhost:5022/api/Movie/1001

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{   "id": 1001,   "title": "Extraction 2",   "year": 2023,   "genre": "Action",   "rating": 7,   "description": "Back from the brink of death, commando Tyler Rake embarks on a dangerous mission to save a ruthless gangster's imprisoned family." }</pre></div><div><div>Download</div></div></div> <div><div>Response headers</div><div><pre>content-type: application/json; charset=utf-8 date: Thu, 05 Dec 2024 18:39:34 GMT server: Kestrel transfer-encoding: chunked</pre></div></div>

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

Now, I will update the details of one of the films using a PUT request:

Before:



PUT

/api/Movie/{id}

Cancel

Reset

Parameters

Name	Description
<b>id</b> <small>required</small>	
<small>integer(int32)</small>	123
<small>(path)</small>	

11829

Request body

application/json

```
{
  "id": 123,
  "title": "This is a new title",
  "year": 10000,
  "genre": "No genre",
  "rating": 10,
  "description": "string"
}
```

Execute

Responses

Code	Description	Links
200	Success	No links

As you can see, I am updating a movie with 123 ID with some gibberish content. Here is what the endpoint looks like after execution:

```
{
  "id": 123,
  "title": "This is a new title",
  "year": 10000,
  "genre": "No genre",
  "rating": 10,
  "description": "string"
}
```

11829

ExecuteClear

Responses

Curl

```
curl -X 'PUT' \
  'http://localhost:5022/api/Movie/123' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "id": 123,
    "title": "This is a new title",
    "year": 10000,
    "genre": "No genre",
    "rating": 10,
    "description": "string"
  }'
```

Request URL

```
http://localhost:5022/api/Movie/123
```

Server response

Code	Details
204	<div>Response headers<pre>date: Thu, 05 Dec 2024 18:33:18 GMT server: Kestrel</pre></div>

Responses

Code	Description	Links
200	Success	No links

Now, I will delete this movie since it makes no sense to keep it in my database:

Before:

DELETE /api/Movie/{id}

Parameters

Name	Description
id * required	
integer(int32)	
(path)	

123

11829

Execute

Responses

Code	Description	Links
200	Success	No links

After:

DELETE

/api/Movie/{id}

Parameters

Cancel

Name	Description
id <small>required</small>	
integer(\$int32)	123
(path)	

11829

ExecuteClear

Responses

Curl

curl -X 'DELETE' \  
'http://localhost:5022/api/Movie/123' \  
-H 'accept: \*/\*'

Request URL

http://localhost:5022/api/Movie/123

Server response

Code	Details
204	

Undocumented

Response headers

date: Thu, 05 Dec 2024 18:34:39 GMT  
server: Kestrel

Responses

Code	Description	Links
200	Success	No links

To double-check, let's do a GET request for that deleted movie:

GET

/api/Movie/{id}

⌵

Parameters

Cancel

Name	Description
id <span>required</span>	
integer(int32)	123
(path)	

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
'http://localhost:5022/api/Movie/123' \
-H 'accept: text/plain'
```

Request URL

```
http://localhost:5022/api/Movie/123
```

Server response

Code	Details
404	Undocumented Error: Not Found

Response body

```
{
  "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5",
  "title": "Not Found",
  "status": 404,
  "traceId": "00-55a0b79f8037932094d3adeb698a0fd8-62aa49a937854cfd-00"
}
```

Download

Response headers

```
content-type: application/problem+json; charset=utf-8
date: Thu, 05 Dec 2024 18:35:59 GMT
server: Kestrel
transfer-encoding: chunked
```

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

As you can see, I got a not found error.