Defects

Defect	01.01.01 Incorrect status code for attempted login with invalid credentials
Date	14/07/2024
Details	POST request to `/Auth/login` with invalid username or password returns response with status code 400, when 401 may be more suitable
Expected Result	Response with status code 401
Actual Result	Response with status code 400
Test Data	username: string password: global, username: "sparta" password "string"
Environment	Containerized API running on Docker desktop
Future Steps	Consider changing status code of unsuccessful login attempt from 400 to 401 for unauthorised

Defect	02.01 Get courses works without auth
Date	15/07/24
Details	Get courses request works without Authorization header and bearer token
Expected Result	Sending a GET request to `/api/Courses` endpoint without bearer token in `Authorization` header should receive response with 401 status code.
Actual Result	Response has status code 200 and body contains course information
Test Data	
Environment	Containerized API running on Docker desktop
Future Steps	Ask product owner whether this endpoint requires authorization or not

Defect	GET non-existent course incorrect status code
Date	12/07/2024
Details	GET request to endpoints `/api/Courses/{id}` when given a non-existing ID don't receive informative response.
Expected Result	Response with status code 404 and body containing descriptive message informing client that no course with that id exists.
Actual Result	Response with status code 204. No response body

Defect	GET non-existent course incorrect status code
Test Data	id = 0
Environment	Containerized API running on Docker desktop
Future Steps	Correct response status code and add response body similar to below

Response suggestion

{ "message":"No Course found with id 0"}

Defect	Get single course works without auth
Date	15/07/24
Details	Get single course request works without Authorization header and bearer token
Expected Result	Sending a GET request to `/api/Courses/{id}` endpoint without bearer token in `Authorization` header should receive response with 401 status code.
Actual Result	Response has status code 200 and body contains course information
Test Data	
Environment	Containerized API running on Docker desktop
Future Steps	Ask product owner whether this endpoint requires authorization or not

Defect	GET single Spartan incorrect status codes
Date	12/07/2024
Details	GET request to endpoints `/api/Spartans/{id}` when given a non-existing ID don't receive informative response.
Expected Result	Response with status code 404 and body containing descriptive message informing client that no Spartan with that id exists.
Actual Result	Response with status code 204. No response body
Test Data	id = 0
Environment	Containerized API running on Docker desktop
Future Steps	Correct response status code and add response body similar to below

Response suggestion

{ "message":"No Spartan found with id 0"}

Defect	DELETE Spartan response doesn't provide any information
Date	12/07/2024
Details	Successful delete Spartan operation response body and headers contain no useful information to inform client of successful deletion.
Expected Result	Response body containing message informing client of successful deletion
Actual Result	No response body or informative headers
Test Data	id = 1
Environment	Containerized API running on Docker desktop
Future Steps	Add response body with message informing client of successful deletion.

Defect	Incorrect status code for successful update spartan
Date	18/07/2024
Details	Successfully updating a Spartan receives a response with status code 204 and no descriptive message
Expected Result	Response with status code 200 and message confirming message confirming successful operation.
Actual Result	Response with status code 204 and no descriptive error message.
Test Data	See below
Environment	Containerized API running on Docker desktop
Future Steps	Change response code to 200 and consider adding descriptive message

Test data

```
{
  "id": 2,
  "firstName": "string",
  "lastName": "string",
  "university": "string",
  "degree": "string",
  "course": {
    "id": 0,
    "name": "string",
    "stream": {
      "id": 0,
      "name": "string",
      "courses": []
    },
    "streamId": 0,
    "trainer": "string",
```

```
    "startDate": "2024-07-18T09:11:25.329Z",
    "endDate": "2024-07-18T09:11:25.329Z",
    "spartans": []
  },
  "courseId": 0,
  "graduated": true
}
```

Defect	Update Spartan operation works when data is missing
Date	18/07/2024
Details	Updating a spartan is successful even when some data field(s) is missing from the request body.
Expected Result	Response with status code 400 and descriptive message saying some data is missing.
Actual Result	Response with status code 204 and database is updated with given data.
Test Data	See below
Environment	Containerized API running on Docker desktop
Future Steps	Contact product owner to see whether the update Spartan operation should fail if any data is missing or whether it is acceptable to only provide the data the user wants to update.

Test Data

```
{
  "id": 3,
  "firstName": "string",
  "lastName": "string",
  "university": "string",
  "degree": "string",
  "course": {
    "id": 0,
    "name": "string",
    "stream": {
      "id": 0,
      "name": "string",
      "courses": []
    },
    "streamId": 0,
    "trainer": "string",
    "startDate": "2024-07-18T09:11:25.329Z",
    "endDate": "2024-07-18T09:11:25.329Z",
    "spartans": []
  },
  "courseId": 0
}
```

Defect	Update non-existing Spartan returns incorrect response code
Date	18/07/2024
Details	Attempting to update a Spartan by giving an ID that doesn't belong to a Spartan returns the wrong error code and no descriptive error message
Expected Result	Response should have status code 404 and response body should include error message informing client that no Spartan with the given ID exists
Actual Result	Response with status code 400 and no descriptive error message
Test Data	See below
Environment	Containerized API running on Docker desktop
Future Steps	Change response status code and add descriptive error message.

Test Data

Path Params

id = 4

Request Body

```
{
  "id": 2,
  "firstName": "string",
  "lastName": "string",
  "university": "string",
  "degree": "string",
  "course": {
    "id": 0,
    "name": "string",
    "stream": {
      "id": 0,
      "name": "string",
      "courses": []
    },
    "streamId": 0,
    "trainer": "string",
    "startDate": "2024-07-18T09:11:25.329Z",
    "endDate": "2024-07-18T09:11:25.329Z",
    "spartans": []
  },
  "courseId": 0,
  "graduated": true
}
```