



Milestone: 5 Submission

User Story :- Secure student interface

API being tested : <http://127.0.0.1:5000/api/user>

Inputs :

Request Mode :- **POST**

Json Body as

```
{
  "email": "testuser@example.com",
  "username": "testuser",
  "password1": "testpassword",
}
```

Expected output:

HTTP Status Code =200

Return Json Result as

```
{
  "username": "testuser",
  "email": "testuser@example.com"
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as

```
{
  "username": "testuser",
  "email": "testuser@example.com"
}
```

Result- Success



Milestone: 5 Submission

```
def test_user_signup():  
    api_url = 'http://127.0.0.1:5000/api/user'  
    data = {  
        "email": "testuser@example.com",  
        "username": "testuser",  
        "password1": "testpassword",  
    }  
    # Make the POST request  
    response = requests.post(api_url, json=data)  
    # Assert the status code is 200 (Success)  
    assert response.status_code == 200
```

User Story :- Providing multiple learning paths

API being tested : http://127.0.0.1:5000/api/learning_path/1

Inputs :

Request Mode :- **POST**

Json Body as

```
{  
    "term_1": ["BDM", "MAD1", "DBMS", "MAD1 Project"],  
    "term_2": ["PDSA", "MLF", "MAD2", "MAD2 Project"],  
    "term_3": ["BA", "MLT", "Java", "TDS"],  
    "term_4": ["MLP", "SC", "MLT Project", "MLP Project"]  
}
```

Expected output:

HTTP Status Code =200

Return Json Result as

```
{  
    "image_path": "learning_path_20231206113952.png"  
}
```



Milestone: 5 Submission

Actual Output :

HTTP Status Code =200

Return Json Result as

```
{
  "image_path": "learning_path_20231206113952.png"
}
```

Result- Success

```
def test_generate_learning_path():
    api_url = 'http://127.0.0.1:5000/api/learning_path/1'
    course_data = {
        "term_1": ["BDM", "MAD1", "DBMS", "MAD1 Project"],
        "term_2": ["PDSA", "MLF", "MAD2", "MAD2 Project"],
        "term_3": ["BA", "MLT", "Java", "TDS"],
        "term_4": ["MLP", "SC", "MLT Project", "MLP Project"]
    }
    data = {
        "course_data": course_data
    }
    response = requests.post(api_url, json=data)
    assert response.status_code == 201
```

User Story :- Providing multiple learning paths

API being tested : http://127.0.0.1:5000/api/learning_path/1

Inputs :

Request Mode :- **POST**

Json Body as

```
{}
```

Expected output:

HTTP Status Code =400

```
{
  "message": "Request must contain a valid JSON payload"
}
```

Actual Output :

HTTP Status Code =400

```
{
  "message": "Request must contain a valid JSON payload"
}
```



Milestone: 5 Submission

Result- Success

```
def test_generate_learning_path_errors():  
    api_url = 'http://127.0.0.1:5000/api/learning_path/1'  
    # Intentionally omitting the course_data to trigger an error  
    data = {}  
    # Make the POST request.  
    response = requests.post(api_url, json=data)  
    # Assert the status code is 400  
    assert response.status_code == 400
```

User Story :- Providing multiple learning paths

API being tested : http://127.0.0.1:5000/api/learning_path/1

Inputs :

Request Mode :- GET

User ID :- 1

Expected output:

HTTP Status Code =200

Return Json Result as

```
{  
    "image_path": "learning_path_20231206113952.png"  
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as

```
{  
    "image_path": "learning_path_20231206113952.png"  
}
```

Result- Success

```
def test_current_learning_path():  
    api_url = 'http://127.0.0.1:5000/api/learning_path/1'  
    # Make the GET request  
    response = requests.get(api_url)  
    # Assert the status code is 200  
    assert response.status_code == 200
```



Milestone: 5 Submission

User Story :- Providing multiple learning paths

API being tested : http://127.0.0.1:5000/api/learning_path/x

Inputs :

Request Mode :- **GET**

User ID :- 1

Expected output:

HTTP Status Code =404

```
{  
  "message": "No learning path found for the user"  
}
```

Actual Output :

HTTP Status Code =404

```
{  
  "message": "No learning path found for the user"  
}
```

Result- Success

```
def test_current_learning_path_error():  
    api_url = 'http://127.0.0.1:5000/api/learning_path/x'  
    # Make the GET request  
    response = requests.get(api_url)  
    # Assert the status code is 404  
    assert response.status_code == 404
```

User Story :- Exploring popular learning paths

API being tested : http://127.0.0.1:5000/api/topRated_learning_paths

Inputs :

Request Mode :- **GET**

Expected output:

HTTP Status Code =200

Return Json Result as

```
[  
  {"image_path": "learning_path_20231206113952.png"},  
  ...  
]
```



Milestone: 5 Submission

```
{
  "image_path": "learning_path_20231206113953.png",
  "image_path": "learning_path_20231206113954.png",
  "image_path": "learning_path_20231206113955.png",
  "image_path": "learning_path_20231206113956.png",
  "image_path": "learning_path_20231206113957.png"
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as

```
[
  {
    "image_path": "learning_path_20231206113952.png",
    "image_path": "learning_path_20231206113953.png",
    "image_path": "learning_path_20231206113954.png",
    "image_path": "learning_path_20231206113955.png",
    "image_path": "learning_path_20231206113956.png",
    "image_path": "learning_path_20231206113957.png"
  }
]
```

Result- Success

```
def test_top_rated_learning_paths():
    api_url = 'http://127.0.0.1:5000/api/top_rated_learning_paths'
    # Make the GET request
    response = requests.get(api_url)
    # Assert the status code is 200 (Success)
    assert response.status_code == 200
```

User Story :- Exploring popular learning paths

API being tested : http://127.0.0.1:5000/api/learning_path/1

Inputs :

Request Mode :- **GET**

Expected output:

HTTP Status Code =404

Return Json Result as

```
{
  "message": "No learning paths found"
}
```



Milestone: 5 Submission

Actual Output :

```
HTTP Status Code =404
Return Json Result as
{
  "message": "No learning paths found"
}
```

Result- Success

```
def test_top_rated_learning_paths_error():
    api_url = 'http://127.0.0.1:5000/api/top_rated_learning_paths'
    # Make the GET request
    response = requests.get(api_url)
    # Assert the status code is 404 (Not Found)
    assert response.status_code == 404
```

User Story :- Feedback on the learning paths

API being tested : http://127.0.0.1:5000/api/all_learning_paths

Inputs :

Request Mode :- GET

Expected output:

```
HTTP Status Code =200
Return Json Result as
{
  [
    {"image_path": "learning_path_20231206113952.png"},
    {"image_path": "learning_path_20231206113953.png"},
    {"image_path": "learning_path_20231206113954.png"},
    {"image_path": "learning_path_20231206113955.png"},
    {"image_path": "learning_path_20231206113956.png"},
    {"image_path": "learning_path_20231206113957.png"}
  ]
}
```

Actual Output :

```
HTTP Status Code =200
Return Json Result as
{
  [
```



Milestone: 5 Submission

```
{
  "image_path": "learning_path_20231206113952.png",
  "image_path": "learning_path_20231206113953.png",
  "image_path": "learning_path_20231206113954.png",
  "image_path": "learning_path_20231206113955.png",
  "image_path": "learning_path_20231206113956.png",
  "image_path": "learning_path_20231206113957.png"
}
```

Result- Success

```
def test_all_learning_paths():
    api_url = 'http://127.0.0.1:5000/api/all_learning_paths'
    # Make the GET request
    response = requests.get(api_url)
    # Assert the status code is 200 (Success)
    assert response.status_code == 200
```

User Story :- Feedback on the learning paths

API being tested : <http://127.0.0.1:5000/api/feedbacks/1>

Inputs :

Request Mode :- **GET**

User ID :- 1

Expected output:

HTTP Status Code =200

Return Json Result as

```
{
  "id": 1,
  "learning_path_id": 101,
  "rating": 5,
  "remarks": "Excellent content!",
  "created": "2023-12-06T11:39:52Z"
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as

```
{
  "id": 1,
```




Milestone: 5 Submission

```
"learning_path_id": 101,  
"rating": 5,  
"remarks": "Excellent content!",  
"created": "2023-12-06T11:39:52Z"  
}
```

Result- Success

```
def test_learning_path_feedbacks(learning_path_id):  
    api_url = f'http://127.0.0.1:5000/feedbacks/1'  
    # Make the GET request  
    response = requests.get(api_url)  
    # Assert the status code is 200 (Success)  
    assert response.status_code == 200
```

User Story :- Feedback on the learning paths

API being tested : <http://127.0.0.1:5000/api/feedbacks/1>

Inputs :

Request Mode :- **POST**

User ID :- 1

Jason data as

```
{  
    'rating': 5,  
    'remarks': 'Great learning experience!',  
}
```

Expected output:

HTTP Status Code =201

Return Json Result as

```
{  
    "learning_path_id": 101,  
    "rating": 5,  
    "remarks": "Excellent content!",  
    "created": "2023-12-06T11:39:52Z"  
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as



Milestone: 5 Submission

```
{  
  "learning_path_id": 101,  
  "rating": 5,  
  "remarks": "Excellent content!",  
  "created": "2023-12-06T11:39:52Z"  
}
```

Result- Success

```
def test_post_learning_path_feedbacks(learning_path_id):  
    api_url = 'http://127.0.0.1:5000/feedbacks/1'  
    # Define the data you want to send in the POST request  
    data = {  
        'rating': 5,  
        'remarks': 'Great learning experience!',  
    }  
    # Make the POST request  
    response = requests.post(api_url, json=data)  
    # Assert the status code is 201 (Created)  
    assert response.status_code == 201
```

User Story :- Feedback on the learning paths

API being tested : <http://127.0.0.1:5000/api/feedbacks/1>

Inputs :

Request Mode :- **POST**

User ID :- 1

Jason data as

```
{  
    'rating': -1,  
    'remarks': 'Great learning experience!',  
}
```

Expected output:

HTTP Status Code =201

Return Json Result as

```
{  
  "learning_path_id": 101,  
  "rating": 5,  
  "remarks": "Excellent content!",  
}
```



Milestone: 5 Submission

```
"created": "2023-12-06T11:39:52Z"
}
```

Actual Output :

Error throws in pytest

Result- Fail

```
def test_post_learning_path_feedbacks_error(learning_path_id):
    api_url = 'http://127.0.0.1:5000/feedbacks/1'
    # Define the data with missing required fields for error case
    data = {
        'remarks': 'Great learning experience!',
    }
    # Make the POST request
    response = requests.post(api_url, json=data)
    # Assert the status code is 400 (Bad Request)
    assert response.status_code == 400
```

User Story :- Considering course prerequisites and corequisites

API being tested : <http://127.0.0.1:5000/api/courses>

Inputs :

Request Mode :- **GET**

Expected output:

HTTP Status Code =200

Return Json Result as

```
{
  [
    {
      "id": 1,
      "name": "BDM",
      "prerequisite": "",
      "corequisite": "",
      "type_": "Data Science",
      "level": "2",
      "project": "BDM Project"
    },
    {
      "id": 2,
      "name": "MAD1",
```



Milestone: 5 Submission

```
"prerequisite": "",
"corequisite": "",
"type_": "Programming",
"level": "2",
"project": "MAD1 Project"
},
{
  "id": 3,
  "name": "MAD2",
  "prerequisite": "MAD1",
  "corequisite": "",
  "type_": "Programming",
  "level": "2",
  "project": "MAD2 Project"
},
]
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as

```
{
  [
    {
      "id": 1,
      "name": "BDM",
      "prerequisite": "",
      "corequisite": "",
      "type_": "Data Science",
      "level": "2",
      "project": "BDM Project"
    },
    {
      "id": 2,
      "name": "MAD1",
      "prerequisite": "",
      "corequisite": "",
      "type_": "Programming",
      "level": "2",
      "project": "MAD1 Project"
    }
  ]
}
```



Milestone: 5 Submission

```
},  
{  
  "id": 3,  
  "name": "MAD2",  
  "prerequisite": "MAD1",  
  "corequisite": "",  
  "type_": "Programming",  
  "level": "2",  
  "project": "MAD2 Project"  
},  
]  
}
```

Result- Success

```
def test_get_courses():  
    # Make the GET request  
    api_url = 'http://127.0.0.1:5000/api/courses'  
    response = requests.get(api_url)  
    # Assert the status code is 200 (Success)  
    assert response.status_code == 200
```

User Story :- Considering course prerequisites and corequisites

API being tested : <http://127.0.0.1:5000/api/courses>

Inputs :

Request Mode :- **POST**

Json as

```
{  
  'course_id': 4,  
  'course_name': 'New Course',  
  'pre_req': 'Prerequisite',  
  'co_req': 'Corequisite',  
  'type_': 'Programming',  
  'level': '4',  
  'project': 'Project',  
}
```

Expected output:

HTTP Status Code =201

Return Json Result as



Milestone: 5 Submission

```
{  
  'course_id': 4,  
  'course_name': 'New Course',  
  'pre_req': 'Prerequisite',  
  'co_req': 'Corequisite',  
  'type_': 'Programming',  
  'level': '4',  
  'project': 'Project',  
}
```

Actual Output :

HTTP Status Code =201

Return Json Result as

```
{  
  'course_id': 4,  
  'course_name': 'New Course',  
  'pre_req': 'Prerequisite',  
  'co_req': 'Corequisite',  
  'type_': 'Programming',  
  'level': '4',  
  'project': 'Project',  
}
```



Milestone: 5 Submission

Result- Success

```
def test_post_course():  
    api_url = 'http://127.0.0.1:5000/api/course'  
    data = {  
        'course_id': 'CS004',  
        'course_name': 'New Course',  
        'pre_req': 'Prerequisite',  
        'co_req': 'Corequisite',  
        'type': 'Programming',  
        'level': '4',  
        'project': 'Project',  
    }  
    # Make the POST request  
    response = requests.post(api_url, json=data)  
    # Assert the status code is 201 (Success)  
    assert response.status_code == 201
```

User Story :- Considering course prerequisites and corequisites

API being tested : <http://127.0.0.1:5000/api/courses/CS004>

Inputs :

Request Mode :- **PUT**

Course id:- CS004

Json as

```
{  
    'course_name': 'Course',  
    'pre_req': 'Updated Prerequisite',  
    'co_req': 'Updated Corequisite',  
    'type_': 'Data Science',  
    'level': '2',  
    'project': 'Updated Project',
```



Milestone: 5 Submission

```
}
```

Expected output:

HTTP Status Code =200

Return Json Result as

```
{
  'course_name': 'Course',
  'pre_req': 'Updated Prerequisite',
  'co_req': 'Updated Corequisite',
  'type_': 'Data Science',
  'level': '2',
  'project': 'Updated Project',
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as

```
{
  'course_name': 'Course',
  'pre_req': 'Updated Prerequisite',
  'co_req': 'Updated Corequisite',
  'type_': 'Data Science',
  'level': '2',
  'project': 'Updated Project',
}
```

Result- Success



Milestone: 5 Submission

```
def test_put_course():  
    api_url = 'http://127.0.0.1:5000/api/course/CS004'  
    # Replace {id} with the actual course_id  
    data = {  
        'course_name': 'Course',  
        'pre_req': 'Updated Prerequisite',  
        'co_req': 'Updated Corequisite',  
        'type': 'Data Science',  
        'level': '2',  
        'project': 'Updated Project',  
    }  
    response = requests.put(api_url, json=data)  
    # Assert the status code is 200 (Success)  
    assert response.status_code == 200
```

User Story :- Considering course prerequisites and corequisites

API being tested : <http://127.0.0.1:5000/api/course/CS004>

Inputs :

Request Mode :- **PUT**

Course id:- CSS04

Expected output:

HTTP Status Code =400

Actual Output :

HTTP Status Code =400

Result- Success

```
def test_put_course_error():  
    api_url = 'http://127.0.0.1:5000/api/course/CSS04'  
    # Intentionally giving wrong course id  
    data = {}  
    response = requests.put(api_url, json=data)  
    # Assert the status code is 404 (Not Found)  
    assert response.status_code == 404
```



Milestone: 5 Submission

User Story :- Considering course prerequisites and corequisites

API being tested : <http://127.0.0.1:5000/api/course/CS004>

Inputs :

Request Mode :- **DELETE**

Course id:- CS004

Expected output:

HTTP Status Code =200

Actual Output :

HTTP Status Code =200

Result- Success

```
def test_delete_course():
    api_url = 'http://127.0.0.1:5000/api/courses'
    course_id = 4
    # Make the DELETE request
    response = requests.delete(f'{api_url}/{course_id}')
    # Assert the status code is 200 (Success)
    assert response.status_code == 200
```

User Story :- Considering course prerequisites and corequisites

API being tested : <http://127.0.0.1:5000/api/course/CS004>

Inputs :

Request Mode :- **DELETE**

Course id:- 4

Expected output:

HTTP Status Code =500

Actual Output :

HTTP Status Code =500

Result- Success

```
def test_delete_course_error():
    api_url = 'http://127.0.0.1:5000/api/course/CSS004'
    # Make the DELETE request
    response = requests.delete(api_url)
    # Assert the status code is 500 (Success)
    assert response.status_code == 500
```



Milestone: 5 Submission

User Story :- Enhancing recommendations through feedback

API being tested : <http://127.0.0.1:5000/api/CourseFeedback/1>

Inputs :

Request Mode :- **GET**

User Id =1

Expected output:

HTTP Status Code =200

Return Json Result as

```
{
  "id": 1,
  "user_id": 1,
  "subject_id": "2",
  "rating": 4,
  "remarks": "Very good update"
}
```

Actual Output :

HTTP Status Code =200

Return Json Result as

```
{
  "id": 1,
  "user_id": 1,
  "subject_id": "2",
  "rating": 4,
  "remarks": "Very good update"
}
```

Result- Success

```
def test_api_call_success():
    api_url = 'http://127.0.0.1:5000/api/CourseFeedback/1'

    # Make the API call
    response = requests.get(api_url)

    # Assert the status code is 200
    assert response.status_code == 200
```



Milestone: 5 Submission

User Story :- Enhancing recommendations through feedback

API being tested : <http://127.0.0.1:5000/api/CourseFeedback/10>

Inputs :

Request Mode :- **GET**

User Id =10

Expected output:

HTTP Status Code = 404 NOT FOUND

Return Json Result as

```
{  
  "message": "No feedback for given by user"  
}
```

Actual Output :

HTTP Status Code = 404 NOT FOUND

Return Json Result as

```
{  
  "message": "No feedback for given by user"  
}
```

Result- Success

```
def test_api_call_fail():  
    api_url = 'http://127.0.0.1:5000/api/CourseFeedback/10'  
  
    # Make the API call  
    response = requests.get(api_url)  
  
    # Assert the status code is 403  
    assert response.status_code == 404
```

User Story :- Enhancing recommendations through feedback

API being tested : <http://127.0.0.1:5000/api/CourseFeedback/10000000000000000000>

Inputs :

Request Mode :- **GET**

User Id =10000000000000000000

Expected output:

HTTP Status Code = 403 FORBIDDEN

Return Json Result as

```
{  
  "message": "Internal Server Error"  
}
```

Actual Output :



Milestone: 5 Submission

HTTP Status Code = 403 FORBIDDEN

Return Json Result as

```
{  
  "message": "Internal Server Error"  
}
```

Result- Success

```
def test_api_call_fail_1():  
    api_url = 'http://127.0.0.1:5000/api/CourseFeedback/1000000000000000000'  
  
    # Make the API call  
    response = requests.get(api_url)  
  
    # Assert the status code is 500  
    assert response.status_code == 500
```

User Story :- Enhancing recommendations through feedback

API being tested : <http://127.0.0.1:5000/api/CourseFeedback/>

Inputs :

Request Mode :- **POST**

User Id =1

Json Body

```
{  
  "subject_id": 4,  
  "rating": 5,  
  "remarks": "Very good"  
}
```

Expected output:

HTTP Status Code = 201 CREATED

```
{  
  "id": 15,  
  "user_id": 2,  
  "subject_id": "4",  
  "rating": 5,  
  "remarks": "Very good"  
}
```



Milestone: 5 Submission

Actual Output :

HTTP Status Code = 201 CREATED

```
{  
  "id": 15,  
  "user_id": 2,  
  "subject_id": "4",  
  "rating": 5,  
  "remarks": "Very good"  
}
```

Result- Success

```
def test_post_request():  
  
    api_url = 'http://127.0.0.1:5000/api/CourseFeedback/1'  
  
    # Define the data you want to send in the POST request  
    data = {  
        "subject_id": '4',  
        "rating": '5',  
        "remarks": "Very good"  
    }  
  
    # Make the POST request. PLEASE use JSON so that data will pass as Json  
    response = requests.post(api_url, json=data)  
  
    # Assert the status code is 201 or any other expected status code  
    assert response.status_code == 201
```

User Story :- Enhancing recommendations through feedback

API being tested : <http://127.0.0.1:5000/api/CourseFeedback/1>

Inputs :

Request Mode :- **POST**

User Id =1

Json Body

```
{  
  "subject_id": ,  
  "rating": 5,  
  "remarks": "Very good"  
}
```



Milestone: 5 Submission

Expected output:

HTTP Status Code = 403 FORBIDDEN
Return Json Result as
{
 "message": "Internal Server Error"
}

Actual Output :

Error throws in pytest

Result- Fail

```
def test_post_request_fail():  
  
    api_url = 'http://127.0.0.1:5000/api/CourseFeedback/1'  
  
    # Define the data you want to send in the POST request  
    data = {  
        "subject_id": 1,  
        "rating": 4,  
        "remarks": "Very good"  
    }  
  
    # Make the POST request. PLEASE use JSON so that data will pass as Json  
    response = requests.post(api_url, json=data)  
  
    # Assert the status code is 201 or any other expected status code  
    assert response.status_code == 201
```

User Story :- Enhancing recommendations through feedback

API being tested : <http://127.0.0.1:5000/api/CourseFeedback/1/1>

Inputs :

Request Mode :- **PUT**

User Id =1

Subject ID =1

Json Body

```
{  
  "rating": 4,  
  "remarks": "Very good update"  
}
```



Milestone: 5 Submission

Expected output:

```
HTTP Status Code = 201 CREATED
{
  "id": 1,
  "user_id": 1,
  "subject_id": "2",
  "rating": 4,
  "remarks": "Very good update"
}
```

Actual Output :

```
HTTP Status Code = 201 CREATED
{
  "id": 1,
  "user_id": 1,
  "subject_id": "2",
  "rating": 4,
  "remarks": "Very good update"
}
```

Result- Success

```
def test_put_request():

    api_url = 'http://127.0.0.1:5000/api/CourseFeedback/1/2'

    # Define the data you want to send in the POST request
    data = {
        "rating": 5,
        "remarks": "Very good"
    }

    # Make the POST request. PLEASE use JSON so that data will pass as Json
    response = requests.put(api_url, json=data)

    # Assert the status code is 201 or any other expected status code
    assert response.status_code == 201
```




Milestone: 5 Submission

User Story :- Feedback on the learning paths

API being tested : <http://127.0.0.1:5000/api/StudentAcademicInfo/1/F3-2024>

Inputs :

Request Mode :- **GET**

User Id =1

Subject ID =1

Json Body

```
{  
  "rating": 4,  
  "remarks": "Very good update"  
}
```

Expected output:

HTTP Status Code = 200 OK

```
{  
  "roll_number": "1",  
  "term_id": "F3-2024",  
  "course_id": "100",  
  "avg_ga": 90.0,  
  "quiz1": 10.0,  
  "quiz2": 10.0,  
  "end_term": 10.0,  
  "oppe1": 10.0,  
  "oppe2": 10.0,  
  "project": 10.0,  
  "final_score": 10.0  
}
```

Actual Output :

HTTP Status Code = 200 OK

```
{  
  "roll_number": "1",  
  "term_id": "F3-2024",  
  "course_id": "100",  
  "avg_ga": 90.0,  
  "quiz1": 10.0,  
  "quiz2": 10.0,  
  "end_term": 10.0,  
  "oppe1": 10.0,  
  "oppe2": 10.0,  
  "project": 10.0,  
  "final_score": 10.0  
}
```



Milestone: 5 Submission

}

Result- Success

```
def test_StudentAcademic_api_call_success():
    api_url = 'http://127.0.0.1:5000/api/StudentAcademicInfo/1/F3-2024'

    # Make the API call
    response = requests.get(api_url)

    # Assert the status code is 200
    assert response.status_code == 200
```

User Story :- Feedback on the learning paths

API being tested : http://127.0.0.1:5000/api/StudentProfile

Inputs :

Request Mode :- **POST**

```
{
  "roll_number":2,
  "academic_interests": "Higher study",
  "learning_goals" : "DP",
  "schedules" : "3 month",
  "commitments" : "Higher study+1"
}
```

Expected output:

```
HTTP Status Code = 201 CREATED
{
  "roll_number": "2",
  "academic_interests": "Higher study",
  "learning_goals": "DP",
  "schedules": "3 month",
  "commitments": "Higher study+1"
}
```

Actual Output :

```
HTTP Status Code = 201 CREATED
{
  "roll_number": "2",
  "academic_interests": "Higher study",
  "learning_goals": "DP",
  "schedules": "3 month",
```



Milestone: 5 Submission

```
"commitments": "Higher study+1"
}
```

Result- Success

```
def test_Studentprofile_post_call_success():
    api_url = 'http://127.0.0.1:5000/api/StudentProfile'

    # Define the data you want to send in the POST request
    data = {
        "roll_number":4,
        "academic_interests": "Higher study",
        "learning_goals" : "DP",
        "schedules" : "3 month",
        "commitments" : "Higher study+1"
    }

    # Make the POST request. PLEASE use JSON so that data will pass as Json
    response = requests.post(api_url, json=data)

    # Assert the status code is 201 or any other expected status code
    assert response.status_code == 201
```

User Story :- Feedback on the learning paths

API being tested : http://127.0.0.1:5000/api/StudentProfile/2

Inputs :

Request Mode :- **GET**

RollNumberParam =2

Expected output:

HTTP Status Code = 200 OK

```
{
  "roll_number": "2",
  "academic_interests": "Higher study",
  "learning_goals": "DP",
  "schedules": "3 month",
  "commitments": "Higher study+1"
}
```



Milestone: 5 Submission

Actual Output :

```
HTTP Status Code = 200 OK
{
  "roll_number": "2",
  "academic_interests": "Higher study",
  "learning_goals": "DP",
  "schedules": "3 month",
  "commitments": "Higher study+1"
}
```

Result- Success

```
def test_studentprofile_get_call_success():
    api_url = 'http://127.0.0.1:5000/api/StudentProfile/2'

    # Make the API call
    response = requests.get(api_url)

    # Assert the status code is 200
    assert response.status_code == 200
```

User Story :- Feedback on the learning paths

API being tested : <http://127.0.0.1:5000/StudentProfile/2999999999999999>

Inputs :

Request Mode :- **GET**
RollNumberParam =2999999999999999

Expected output:

```
HTTP Status Code = 404 NOT FOUND
Return Json Result as
{
  "message": "No student profile found"
}
```

Actual Output :

```
HTTP Status Code = 404 NOT FOUND
Return Json Result as
{
  "message": "No student profile found"
}
```

Result- Success



Milestone: 5 Submission

```
def test_Studentprofile_get_call_fail():  
    api_url = 'http://127.0.0.1:5000/api/StudentProfile/2999999999999'  
  
    # Make the API call  
    response = requests.get(api_url)  
  
    # Assert the status code is 200  
    assert response.status_code == 404
```

User Story :- Upload Student Academic Info

API Used: http://127.0.0.1:5000/api/add_student_academic_info

Inputs :

Request Mode :- **POST**

```
{  
  "roll_number": "S013",  
  "term_id": 1,  
  "course_id": "CS101",  
  "avg_ga": 85.5,  
  "quiz1": 75.0,  
  "quiz2": 88.5,  
  "end_term": 92.0,  
  "oppe1": 90.0,  
  "oppe2": 87.5,  
  "project": 95.0,  
  "final_score": 89.7  
}
```

Expected output:

HTTP Status Code = 201 CREATED

```
{  
  "roll_number": "S013",  
  "term_id": 1,  
  "course_id": "CS101",  
  "avg_ga": 85.5,  
  "quiz1": 75.0,  
  "quiz2": 88.5,  
  "end_term": 92.0,  
  "oppe1": 90.0,  
  "oppe2": 87.5,  
  "project": 95.0,  
  "final_score": 89.7  
}
```



Milestone: 5 Submission

```
}
```

Actual Output :

HTTP Status Code = 201 CREATED

```
{  
  "roll_number": "S013",  
  "term_id": 1,  
  "course_id": "CS101",  
  "avg_ga": 85.5,  
  "quiz1": 75.0,  
  "quiz2": 88.5,  
  "end_term": 92.0,  
  "oppe1": 90.0,  
  "oppe2": 87.5,  
  "project": 95.0,  
  "final_score": 89.7  
}
```

Result- Success

```
def test_add_student_academic_info_api_success():  
    api_url = 'http://127.0.0.1:5000/api/add_student_academic_info'  
  
    data = {  
        "roll_number": "S040",  
        "term_id": 1,  
        "course_id": "CS101",  
        "avg_ga": 85.5,  
        "quiz1": 75.0,  
        "quiz2": 88.5,  
        "end_term": 92.0,  
        "oppe1": 90.0,  
        "oppe2": 87.5,  
        "project": 95.0,  
        "final_score": 89.7  
    }  
  
    response = requests.post(api_url, json=data)  
    assert response.status_code == 201
```



Milestone: 5 Submission

User Story :- Upload Student GPA Details

API Used: http://127.0.0.1:5000/api/add_gpa

Inputs :

Request Mode :- **POST**

```
{
  "roll_number": "S037",
  "term_id": 3,
  "sgpa": 7.8,
  "cgpa": 7.6
}
```

Expected output:

HTTP Status Code = 201 CREATED

```
{
  "roll_number": "S037",
  "term_id": 3,
  "sgpa": 7.8,
  "cgpa": 7.6
}
```

Actual Output :

HTTP Status Code = 201 CREATED

```
{
  "roll_number": "S037",
  "term_id": 3,
  "sgpa": 7.8,
  "cgpa": 7.6
}
```

Result- Success



Milestone: 5 Submission

```
def test_add_gpa_api_success():  
    api_url = 'http://127.0.0.1:5000/api/add_gpa'  
  
    data = {  
        "roll_number": "S040",  
        "term_id": 3,  
        "sgpa": 7.8,  
        "cgpa": 7.6  
    }  
  
    response = requests.post(api_url, json=data)  
    assert response.status_code == 201
```

User Story :- Upload Student Offline Education Details

API Used: http://127.0.0.1:5000/api/add_offline_education

Inputs :

Request Mode :- **POST**

```
{  
    "degree": "Bachelor of Science",  
    "major": "Computer Science",  
    "end_year": "2022"  
}
```

Expected output:

HTTP Status Code = 201 CREATED

```
{  
    "degree": "Bachelor of Science",  
    "major": "Computer Science",  
    "end_year": "2022"  
}
```

Actual Output :

HTTP Status Code = 201 CREATED

```
{  
    "degree": "Bachelor of Science",  
    "major": "Computer Science",  
    "end_year": "2022"  
}
```

Result- Success



Milestone: 5 Submission

```
def test_add_offline_education_api_success():
    api_url = 'http://127.0.0.1:5000/api/add_offline_education'

    data = {
        "roll_number": "S040",
        "degree": "Bachelor of Science",
        "major": "Computer Science",
        "end_year": "2022"
    }

    response = requests.post(api_url, json=data)
    assert response.status_code == 201
```

User Story :- Analytics

API Used: <http://127.0.0.1:5000/api/admin>

Inputs :

Request Mode :- **GET**

Email:- admin@admin.com

Expected output:

HTTP Status Code = 200

```
{
  "tstudents": [
    {"id": 1, "name": "Student1", "roll_number": "21f1000123", "email": "student1@example.com"},
    {"id": 2, "name": "Student2", "roll_number": "21f1000124", "email": "student2@example.com"},
    {"id": 3, "name": "Student3", "roll_number": "21f1000125", "email": "student3@example.com"}
    // ... (data for all students)
  ],
  "tcourses": 16, // Total number of courses
  "counts": {
    "1": 10, // Number of learning paths with a rating of 1
    "2": 15, // Number of learning paths with a rating of 2
    "3": 20, // Number of learning paths with a rating of 3
    "4": 25, // Number of learning paths with a rating of 4
    "5": 30 // Number of learning paths with a rating of 5
  },
  "StudentsDataSummaryReport": {},
  "StudentsDataSummaryReportImage": "static/IMG/StudentSatisfactionMetric.png"
}
```



Milestone: 5 Submission

Actual Output :

HTTP Status Code = 200

HTTP Status Code = 200

```
{
  "students": [
    {"id": 1, "name": "Student1", "roll_number": "21f1000123", "email": "student1@example.com"},
    {"id": 2, "name": "Student2", "roll_number": "21f1000124", "email": "student2@example.com"},
    {"id": 3, "name": "Student3", "roll_number": "21f1000125", "email": "student3@example.com"}
    // ... (data for all students)
  ],
  "courses": 16, // Total number of courses
  "counts": {
    "1": 10, // Number of learning paths with a rating of 1
    "2": 15, // Number of learning paths with a rating of 2
    "3": 20, // Number of learning paths with a rating of 3
    "4": 25, // Number of learning paths with a rating of 4
    "5": 30 // Number of learning paths with a rating of 5
  },
  "StudentsDataSummaryReport": {},
  "StudentsDataSummaryReportImage": "static/IMG/StudentSatisfactionMetric.png"
}
```

Result- Success

```
def test_admin_data():
    # Replace {email} with the actual admin email
    api_url = 'http://127.0.0.1:5000/api/admin'
    email = 'admin@example.com'
    # Make the GET request
    response = requests.get(f'{api_url}/{email}')
    # Assert the status code is 200 (Success)
    assert response.status_code == 200
```