

## Course API

### 1. Get Course Count

- **URL:** /courses/count
- **Method:** GET
- **Description:** Get the total number of courses.
- **Response:**
  - **200 OK:** { "count": <number> }
  - **500 Internal Server Error:** { "message": "Internal server error" }

### 1. Get All Courses

- **URL:** /courses
- **Method:** GET
- **Description:** Get a list of all courses.
- **Response:**
  - **200 OK:** [ { "course\_id": <string>, "course\_name": <string>, "department": <string>, "fee": <number> }, ... ]
  - **500 Internal Server Error:** { "message": "Internal server error" }

### 2. Get Course by ID

- **URL:** /courses/:id
- **Method:** GET
- **Description:** Get a course by its ID.
- **Parameters:** id (Course ID)
- **Response:**
  - **200 OK:** { "course\_id": <string>, "course\_name": <string>, "department": <string>, "fee": <number> }
  - **404 Not Found:** { "message": "Course not found" }
  - **500 Internal Server Error:** { "message": "Internal server error" }

### 3. Get Courses by Department

- **URL:** /courses/department/:name
- **Method:** GET
- **Description:** Get courses by department name.

- **Parameters:** name (Department Name)
- **Response:**
  - **200 OK:** [ { "course\_id": <string>, "course\_name": <string>, "department": <string>, "fee": <number> }, ... ]
  - **500 Internal Server Error:** { "message": "Internal server error" }

#### 4. Add New Course

- **URL:** /courses
- **Method:** POST
- **Description:** Add a new course.
- **Request Body:**

json

Copy code

```
{
  "course_id": <string>,
  "course_name": <string>,
  "department": <string>,
  "fee": <number>
}
```

- **Response:**
  - **201 Created:** { "message": "Course added successfully", "result": { ... } }
  - **500 Internal Server Error:** { "message": "Internal server error" }

#### 5. Update Course

- **URL:** /courses/:id
- **Method:** PUT
- **Description:** Update a course by its ID.
- **Parameters:** id (Course ID)
- **Request Body:**

json

Copy code

```
{  
  "course_name": <string>,  
  "department": <string>,  
  "fee": <number>  
}
```

- **Response:**

- **200 OK:** { "message": "Course updated successfully" }
- **404 Not Found:** { "message": "Course not found" }
- **500 Internal Server Error:** { "message": "Internal server error" }

## 6. Delete Course

- **URL:** /courses/:id
- **Method:** DELETE
- **Description:** Delete a course by its ID.
- **Parameters:** id (Course ID)
- **Response:**
  - **200 OK:** { "message": "Course deleted successfully" }
  - **404 Not Found:** { "message": "Course not found" }
  - **500 Internal Server Error:** { "message": "Internal server error" }

---

## Student API

### 1. Get Student Count

- **URL:** /students/count
- **Method:** GET
- **Description:** Get the total number of students.
- **Response:**
  - **200 OK:** { "count": <number> }
  - **500 Internal Server Error:** { "message": "Internal server error" }

### 2. Get All Students

- **URL:** /students

- **Method:** GET
- **Description:** Get a list of all students.
- **Response:**
  - **200 OK:** [ { "student\_id": <string>, "first\_name": <string>, "last\_name": <string>, "birthday": <string>, "address": <string>, "contact\_number": <string>, "course\_id": <string>, "course\_name": <string> }, ... ]
  - **500 Internal Server Error:** { "message": "Internal server error" }

### 3. Get Student by ID

- **URL:** /students/:id
- **Method:** GET
- **Description:** Get a student by their ID.
- **Parameters:** id (Student ID)
- **Response:**
  - **200 OK:** { "student\_id": <string>, "first\_name": <string>, "last\_name": <string>, "birthday": <string>, "address": <string>, "contact\_number": <string>, "course\_id": <string>, "course\_name": <string> }
  - **404 Not Found:** { "message": "Student not found" }
  - **500 Internal Server Error:** { "message": "Internal server error" }

### 4. Get Students by Course

- **URL:** /students/course/:name
- **Method:** GET
- **Description:** Get students by course name.
- **Parameters:** name (Course Name)
- **Response:**
  - **200 OK:** [ { "student\_id": <string>, "first\_name": <string>, "last\_name": <string>, "birthday": <string>, "address": <string>, "contact\_number": <string>, "course\_id": <string>, "course\_name": <string> }, ... ]
  - **500 Internal Server Error:** { "message": "Internal server error" }

### 5. Add New Student

- **URL:** /students
- **Method:** POST

- **Description:** Add a new student.
- **Request Body:**

json

Copy code

```
{
  "student_id": <string>,
  "first_name": <string>,
  "last_name": <string>,
  "birthday": <string>,
  "address": <string>,
  "contact_number": <string>,
  "course_id": <string>,
  "course_name": <string>
}
```

- **Response:**
  - **200 OK:** { "message": "Student added successfully" }
  - **500 Internal Server Error:** { "message": "Internal server error" }

## 6. Update Student

- **URL:** /students/:id
- **Method:** PUT
- **Description:** Update a student by their ID.
- **Parameters:** id (Student ID)
- **Request Body:**

json

Copy code

```
{
  "first_name": <string>,
  "last_name": <string>,
  "birthday": <string>
}
```

```
"address": <string>,  
"contact_number": <string>,  
"course_id": <string>,  
"course_name": <string>  
}
```

- **Response:**

- **200 OK:** { "message": "Student updated successfully" }
- **404 Not Found:** { "message": "Student not found" }
- **500 Internal Server Error:** { "message": "Internal server error" }

## 7. Delete Student

- **URL:** /students/:id
- **Method:** DELETE
- **Description:** Delete a student by their ID.
- **Parameters:** id (Student ID)
- **Response:**
  - **200 OK:** { "message": "Student deleted successfully" }
  - **404 Not Found:** { "message": "Student not found" }
  - **500 Internal Server Error:** { "message": "Internal server error" }

---

## User Authentication API

### 1. Validate User Credentials

- **URL:** /users/validate
- **Method:** POST
- **Description:** Validate user credentials.
- **Request Body:**

json

Copy code

```
{  
  "username": <string>,  
  "password": <string>
```

```
"password": <string>  
}
```

- **Response:**

- **200 OK:** { "message": "Authentication successful" }
- **400 Bad Request:** { "message": "Username and password are required" }
- **401 Unauthorized:** { "message": "Invalid username or password" }
- **500 Internal Server Error:** { "message": "Internal server error" }