Below is a comprehensive checklist of possible API endpoints for the University Course Management System assignment. Each endpoint lists the HTTP method, URL pattern, path parameters, query parameters (if any), and the expected request body.

**Admin Endpoints**

**Student Management**

* **GET** /api/admin/students
  + **Parameters:** None
  + **Request Body:** None
  + **Description:** Retrieve a list of all students (optional paging/query parameters may be supported).
* **GET** /api/admin/students/{studentId}
  + **Path Parameter:** studentId (Long)
  + **Request Body:** None
  + **Description:** Retrieve details for a specific student.
* **POST** /api/admin/students
  + **Request Body:** JSON object containing student details (e.g., name, email, city, salary, etc.).
  + **Description:** Create a new student record.
* **PUT** /api/admin/students/{studentId}
  + **Path Parameter:** studentId
  + **Request Body:** JSON object with updated student details.
  + **Description:** Update an existing student’s details.
* **DELETE** /api/admin/students/{studentId}
  + **Path Parameter:** studentId
  + **Request Body:** None
  + **Description:** Delete a student record.

**Instructor Management**

* **GET** /api/admin/instructors
  + **Parameters:** None
  + **Request Body:** None
  + **Description:** Retrieve a list of all instructors.
* **GET** /api/admin/instructors/{instructorId}
  + **Path Parameter:** instructorId
  + **Request Body:** None
  + **Description:** Retrieve details for a specific instructor.
* **POST** /api/admin/instructors
  + **Request Body:** JSON object with instructor details (e.g., name, email, department).
  + **Description:** Create a new instructor.
* **PUT** /api/admin/instructors/{instructorId}
  + **Path Parameter:** instructorId
  + **Request Body:** JSON object with updated instructor details.
  + **Description:** Update an instructor’s details.
* **DELETE** /api/admin/instructors/{instructorId}
  + **Path Parameter:** instructorId
  + **Request Body:** None
  + **Description:** Delete an instructor.

**Course Management**

* **GET** /api/admin/courses
  + **Parameters:** None
  + **Request Body:** None
  + **Description:** Retrieve a list of all courses.
* **GET** /api/admin/courses/{courseId}
  + **Path Parameter:** courseId
  + **Request Body:** None
  + **Description:** Retrieve details for a specific course.
* **POST** /api/admin/courses
  + **Request Body:** JSON object with course details (e.g., courseCode, courseName, description, creditHours).
  + **Description:** Create a new course.
* **PUT** /api/admin/courses/{courseId}
  + **Path Parameter:** courseId
  + **Request Body:** JSON object with updated course details.
  + **Description:** Update an existing course.
* **DELETE** /api/admin/courses/{courseId}
  + **Path Parameter:** courseId
  + **Request Body:** None
  + **Description:** Delete a course.
* **POST** /api/admin/courses/{courseId}/assign-instructor
  + **Path Parameter:** courseId
  + **Request Body:** JSON object containing instructor information (typically an instructorId).
  + **Description:** Assign an instructor to a course.
* **GET** /api/admin/courses/{courseId}/students
  + **Path Parameter:** courseId
  + **Request Body:** None
  + **Description:** Retrieve the list of students enrolled in a specific course.
* **GET** /api/admin/courses/by-instructor
  + **Query Parameter:** email (Instructor's email)
  + **Request Body:** None
  + **Description:** Retrieve courses taught by a specific instructor.
* **GET** /api/admin/courses/{courseId}/enrollment-count
  + **Path Parameter:** courseId
  + **Request Body:** None
  + **Description:** Get the count of students enrolled in a course.

**Assignment Management**

* **GET** /api/admin/assignments
  + **Parameters:** None
  + **Request Body:** None
  + **Description:** Retrieve a list of all assignments.
* **GET** /api/admin/assignments/{assignmentId}
  + **Path Parameter:** assignmentId
  + **Request Body:** None
  + **Description:** Retrieve details for a specific assignment.
* **POST** /api/admin/assignments
  + **Request Body:** JSON object with assignment details (e.g., title, dueDate, maxScore, and possibly courseId to associate it with a course).
  + **Description:** Create a new assignment.
* **PUT** /api/admin/assignments/{assignmentId}
  + **Path Parameter:** assignmentId
  + **Request Body:** JSON object with updated assignment details.
  + **Description:** Update an assignment.
* **DELETE** /api/admin/assignments/{assignmentId}
  + **Path Parameter:** assignmentId
  + **Request Body:** None
  + **Description:** Delete an assignment.
* **GET** /api/admin/assignments/due-before
  + **Query Parameter:** date (formatted as YYYY-MM-DD)
  + **Request Body:** None
  + **Description:** Retrieve assignments due before a specified date.

**Student Endpoints**

**Student Profile and Enrollment**

* **GET** /api/students/{studentId}
  + **Path Parameter:** studentId
  + **Request Body:** None
  + **Description:** Retrieve the student’s profile and detailed information.
* **PUT** /api/students/{studentId}
  + **Path Parameter:** studentId
  + **Request Body:** JSON object with updated student profile details.
  + **Description:** Update the student's profile.
* **GET** /api/students/{studentId}/courses
  + **Path Parameter:** studentId
  + **Request Body:** None
  + **Description:** Retrieve the list of courses the student is enrolled in.
* **POST** /api/students/{studentId}/courses/{courseId}
  + **Path Parameters:** studentId, courseId
  + **Request Body:** (Optional) May be empty; if needed, can include additional enrollment info.
  + **Description:** Enroll the student in a course.
* **DELETE** /api/students/{studentId}/courses/{courseId}
  + **Path Parameters:** studentId, courseId
  + **Request Body:** None
  + **Description:** Withdraw the student from a course.
* **GET** /api/students/{studentId}/assignments
  + **Path Parameter:** studentId
  + **Request Body:** None
  + **Description:** Retrieve assignments for the courses the student is enrolled in.

**Additional Endpoints (for authentication)**

* **POST** /api/auth/register
  + **Request Body:** JSON object with user registration details (e.g., username, password, email).
  + **Description:** Register a new user.
* **POST** /api/auth/login
  + **Request Body:** JSON object with login credentials (e.g., username and password).
  + **Description:** Authenticate a user and obtain a token or session information.