**Task Description:**

Suppose we're building a school management system and need your help in designing the backend API for managing teachers, classes, courses, students, and their marks.

**Entities:**

* Teacher
* Class
* Course
* Student

**Relationships:**

* One student can have only one class at a time
* One student can have multiple courses **within a class**
* One teacher can teach multiple courses in multiple classes
* One student can have unique marks in a specific class for a specific course taught by a specific teacher

**Task Requirements:**

1. **Entity Relationship Diagram (ERD):** Create an Entity Relationship Diagram (ERD) depicting the entities and their relationships with cardinalities (e.g one-to-one, one-to-many, many-to-many).
2. **API Design:** Design a set of RESTful APIs for managing the data. The API should include functionalities for:
   * Creating, reading, updating, and deleting (CRUD) teachers, classes, courses, and students.
   * Retrieving a student's enrolled courses for a specific class.
   * Adding and retrieving marks for a student in a specific class for a specific course.
   * Implement searching and pagination features for retrieving lists of teachers, classes, courses, or students.
3. **Code Structure:** Provide a brief explanation of your preferred code structure for organizing the backend application using either Node.js (Express) or Django framework (your choice).

**Deliverables:**

Please submit your solutions (ERD & API design document) in a single document (e.g., PDF, Word Doc) & code using GitHub by 2nd July to [hr@softcodix.com](mailto:hr@softcodix.com)

**Evaluation Criteria:**

We will evaluate your submission based on:

* Accuracy and completeness of the ERD.
* Design and clarity of the API endpoints.
* Efficiency considerations for searching and pagination.
* Organization and modularity of the code structure (if provided)