CS 493

Final Project Proposal

5/26/2023

Members:
Tanya Bihari
Hao Truong
Adilbek Bazarkulov
Trevyn Exley

Outline for our database:

- **USER: This entity represents the users of the Tarpaulin application. Each user can have one of three roles: admin, instructor, and student. These roles determine the permissions to perform certain API actions.**

  id (Primary Key)
  name (string)
  email (string: email)
  password (string)
  role (string: admin, student, or instructor)

- **COURSE: This entity represents the courses being managed in Tarpaulin. Each course has basic information and associated data of other entity types, including a list of enrolled students and a set of assignments.**

  id (Primary Key)
  subject_code (string)
  number (integer >= 0)

title (string)
term (string)
instructor (Foreign Key referencing USER.id)
students (list Foreign Key referencing USER.id)
assignments (list Foreign Key referencing ASSIGNMENT.id)

- **ASSIGNMENT: This entity represents a single assignment for a Tarpaulin course. Each assignment belongs to a specific course and has basic information such as a title, due date, etc. It also has a list of individual student submissions.**

  id (Primary Key)
  title (string)
  points (integer >= 0)
  due_date (string: date)
  course_id (Foreign Key referencing COURSE.id)
  submissions(list Foreign Key referencing SUBMISSION.id)

- **SUBMISSION: This entity represents a single student submission for an assignment in Tarpaulin. Each submission belongs both to its assignment and to the student who submitted it, and it is marked with a submission timestamp. Each submission is also associated with a specific file, which will be uploaded to the Tarpaulin API and stored, so it can be downloaded later. Finally, each submission may be assigned a grade.**

  id (Primary Key)
  submission_timestamp
  grade
  assignment_id (Foreign Key referencing ASSIGNMENT.id)
  student_id (Foreign Key referencing USER.id)

- **Relations:**

    - A USER can have multiple COURSEs they are teaching (for instructors) or enrolled in (for students).
    - A COURSE can have multiple ASSIGNMENTs.
    - An ASSIGNMENT can receive multiple SUBMISSIONs.
    - A USER (student) can submit multiple SUBMISSIONs.

- A list of the different services you plan to use to power your API (e.g. MySQL, MongoDB, RabbitMQ, Redis, etc.).
  MongoDB, Postman

- A description of the division of labor for the project between the members of your team (see below for more info about this).

Tanya: Create assignment endpoints, authorization

Hao: Create course endpoints, validation for courses

Adilbek: Create models for the data entities using mongoose. Create User endpoints. User validation (role permissions). Pagination for the entities.

Trevyn: File submission (file should be stored in the database and available for download), Rate limiting, Initial server setup