

Event System

You will build small event System that manages events on ITI with speakers and students, the system will have administrator, Speakers (added by admin) and Student should register first on the system.

Users Roles: administrator , Speaker and Student

Authentication Router

a-/login → Login route for all Users

b-/registerStudent → Student register

Speaker Router

a- /speakers → to get all speakers. (All users)

b- /speakers/id → to get speaker by Id (All users)

c- /speakers → to add new Speaker. (Only Admins)

d-/speakers → to update speaker user Data (Only Admins or speaker itself)

e- /speakers → to delete specified speaker (Only Admins)

Student Router

a- /students → to get all students. (All users)

b- /students/id → to get student by Id (All users)

c- /students → to add new student. (any one through registration)

d-/students → to update student user Data (Admins or user itself)

e- /students → to delete specified student. (Only Admins)

Event Router

a- /events → to get all events. (All Users can Access)

b-/event/id → to get event by Id. (All Users can Access)

c- /events → to add new event Data (Only Admins)

d-/events → to update event user (Only Admins)

e- /events → to delete specified event. (Only Admins)

Models

Student Data: _id(number), fullName,password , and email

Speaker Data: _id(objectID), fullname,password, email , image (which is string)

Event Data: title, eventDate, main speaker, speakers which is array of speakers ids, students which is array of students Ids

Notes

- 1- You should use Express Framework
- 2- Create 4 middleware's on server.
 - a) middleware to write request URL and method using **morgan** package.
 - b) Middle ware for CORS.
 - c) general middleware for **not Found** URL paths.
 - d) One Error handling middleware that will catch all system errors.
- 3- You should Specify the role of user when login (/login) if is admin or speaker or student.
- 4- Use **express-validator** package to validate data is correct.
- 5- Use **JWT** to generate tokens.
- 6- use **dotenv** package to create env file that contains (dbUrl, secret keys, port number, etc.) Then use theses variables in your project.
- 7- Encrypt password using **bcrypt** package, that means usr passwords in DB will be encrypted.
- 8- use auto increment packages with **_ids** filed (client shouldn't send ids as requests data in add routes)
- 9- upload images using **multer** package and save the image on the server images folder, and image name string as image property for speaker and student objects in this task you should use HTML form with enctype attribute.