

Node.js Lab3

You will build part of small Nursery, the system will have only one administrator with static username and password. Teachers should register, and administrator will add children to the system. **Back end will be created as RESTFUL API with json responses with correct status code.**

Requirements :

- 1- Create 3 middlewares on server
 - a- Middleware to write request url and method using **morgan** package (search for morgan on npm)
 - b- General middleware for **not Found** url pathes with 404 status code.
 - c- One Error handling middleware that will catch all system Errors with 500 status code.

Search: add another middleware called CORS, to allow front end in the future to communicate with your back end server.

Create your routes with server side validation for each end point using express-validator.

Now start create your routes

For lab 4

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

Child Data: _id(Number), fullName, age , level (one of PreKG,KG1,KG2), address (city,street and building)

Class Data: _id(Number), name, supervisor (teacher id number), children which is array of children ids

- 1- Create **teacherRoute** that contains
 - a- **/teachers** - get, to get all teachers.
 - b- **/teachers/:id** –get teacher by id
 - c- **/teachers** post , to add new teacher.
 - d- **/teachers** put, to update teacher user Data
 - e- **/teahers** delete, to delete specified teacher.
 - f- **/teachers/supervisors** get, to get all class supervisors.

check your router using postman

2-Create `childRouter` that contains

- a- `/child` - get, to get all children.
- b- `/child/:id` get to get child by Id.
- f-`/child/calss/:id` to get a child class info.
- c- `/child` post , to add new child.
- d-`/child` put, to update child user Data
- e- `/child` delete, to delete specified child.

check your router using postman

3- Create `classRouter` that contains

- a- `/class` - get, to get all classes.
- b-`/class/:id` – get to get class by Id.
- c- `/class` post , to add new class Data
- d-`/class` put, to update class user
- e- `/class` delete, to delete specified class.
- f- `/class/child /:id` get , to get class children info.
- g-`/class/teacher/:id` get, to get class supervisor info.

check your router using postman