Labman APIs(Kaini, black-done)

Green mean Don finished

**API: /equipment/search**

* Method: GET
* Description: Search for equipments by type name.
* Parameters:
  + keyword (type\_name, required): The search keyword.
* Response:
  + equipments (array): An array of equipment objects that match the search query.
  + equipment object:
    - type\_id (integer): The unique identifier of the equipment.
    - type\_name (string): The name of the equipment.
    - total\_amount (integer): The total number of number of the equipment.
    - available\_amount (integer): The number of available number of the equipment.

**API: /equipment**

* Method: GET
* Description: Get list of all equipment.
* Response:
  + equipment object:
    - type\_id (integer): The unique identifier of the equipment.
    - type\_name (string): The name of the equipment.
    - total\_amount (integer): The total number of the equipment.
    - available\_amount (integer): The number of available number of the equipment.

**API: /equipment**

* Method: POST
* Description: Add a new equipment type to the system.
* Parameters:
  + type\_name (string, required): The name of the new equipment.
  + total\_amount (integer, required): The total number of the new equipment.
  + available\_amount (integer, required): The number of available equipment.
* Inner function: set ‘removable’ to be 1, insert into table.
* Response:
  + message (string): A message indicating whether the addition was successful or not.

**API: /equipment/edit**

* Method: PUT
* Description: Edit an equipment.
* Parameters:
  + type\_id (integer, required): The unique identifier of the equipment to be edited.
  + type\_name (string, optional): The updated name of the equipment.
  + total\_amount (integer, optional): The updated total number of the equipment.
  + available\_amount (integer, optional): The updated number of available equipment.
* Response:
  + message (string): A message indicating whether the editing was successful or not.

**API: /equipments/delete**

* Method: DELETE
* Description: Delete an equipment.
* Parameters:
  + type\_id (integer, required): The unique identifier of the equipment to be deleted.
* Response:
  + message (string): A message indicating whether the delete was successful or not.

**API: /users**

* Method: GET
* Description: Get list of all users.
* Response:
  + user object:
  + user\_id (integer): The unique identifier of the user.
  + user\_name (string): The name of the user.
  + email (string): The email of the user.

**API: /users**

* Method: POST
* Description: Create a new user.
* Parameters:
  + user\_name (string, required): The name of the user.
  + email (string, required): The email of the user.
  + password(string): set default password from lab manager
* Response:
  + user object:
  + user\_id (integer): The unique identifier of the created user.
  + user\_name (string): The student id.
  + email (string): The email of the user.

**API: /users/edit**

* Method: PUT
* Description: edit a user.
* Parameters:
  + user\_id (integer): The unique identifier of the created user.
  + user\_name (string, required): The name of the user.
  + email (string, required): The email of the user.
  + password(string): set default password from lab manager
* Response:
  + user object:
  + user\_name (string): The student id.
  + email (string): The email of the user.
  + password(string): set default password from lab manager

**API: /users/{user\_name}**

* Method: GET
* Description: Get a user by user\_name.
* Parameters:
  + user\_name (string, required): The name of the user to get.
* Response:
  + user object:
    - user\_id (integer): The unique identifier of the user.
    - user\_name (string): The name of the user.
    - email (string): The email of the user.

**API: /users/delete**

* Method: DELETE
* Description: delete a user.
* Parameters:
  + user\_id (integer): The unique identifier of the created user.
* Response:
  + message (string): A message indicating whether the user was deleted successfully or not.