User Stories

You are working under an e-commerce company and you are going to extend the platform with functionalities that allow sellers to perform several tasks. Design a simple http server with endpoints that serve the following functions.

Task Requirements:
1. Develop the following endpoints using Node.js
2. Use:
    a. express.js for simple server
    b. MySQL as the database
3. Use appropriate error messages for all endpoints. Example:

```
{
  "message": "Unable to list products for home audio."
}
```

4. Host the code on Github **private repository,** with README.md that documented the API endpoints and instruction to run the source code
5. Upon completion of the tasks, add **ng-plentisoft** as a collaborator for the Github repository.

Assessment Criteria
1. Code readability and structure
2. Coding conventions
3. Security practices
4. Normalized database design

Note: You can use Postman for API testing.

1. Register product to a specified category

The seller would like to register a product under specified category. A product can be assigned under multiple categories as well. This API should create new products and categories if either of them does not exist in the system.

Endpoint: POST /api/product/add
Headers: Content-Type: application/json
Sample request:

```
{
  "product": "Samsung HW-R650 Soundbar",
  "category": [
    "home entertainment",
    "home audio"
  ]
}
```

Response:
- 200 (success)
- 409 (failed)

2. List products under specified categories
The seller would like to retrieve a list of products under a given list of categories.

Endpoint: GET /api/product/list
Sample request: /api/product/list?category=home%20audio
Response:
- 200 (success)

```json
{
  "products": [
    "Samsung HW-R650 Soundbar",
    "LG 65UJ652T UHD 4K TV",
    "SONY PlayStation 4 Mega Pack 2"
  ]
}
```

- 409 (failed)

Endpoint: GET /api/product/list
Sample request:
/api/product/list?category=home%20entertainment&category=gaming%20console
Response:
- 200 (success)

```json
{
  "products": [
    "SONY PlayStation 4 Mega Pack 2",
    "Nintendo Switch",
    "Xbox One X"
  ]
}
```

- 409 (failed)

3. Delete a product
The seller is able to delete a product that is no longer supplied.

Endpoint: POST /api/product/delete
Header: Content-Type: application/json

Request sample:

```json
{
  "product": "Xbox One X"
}
```

Response:
- 204 (success)
- 409 (failed)