

FSD Laboratory 05

Aim:

Develop a responsive web design using Express Framework to perform CRUD operations and deploy with Node JS. use MongoDB.

Objectives:

- Develop a Full-Stack Web Application
- Demonstrate Backend Development and Deployment Proficiency

Theory: (Write down answers on pages)

1. What is the role of Express.js as a web framework for Node.js?
2. Explain the concept of CRUD operations in the context of a web application.
3. Why is MongoDB a suitable choice for this project?
4. What steps are involved in deploying a Node.js and Express application?

Problem Statement: Design webpage according to your roll number don't share the data with other (Upload data on your git account)

- 1. Project: A Blog Post Management System** Develop a responsive web application where users can **create**, **view**, **edit**, and **delete** blog posts. The front end will display a list of all posts and have forms for creating and editing. The back end, powered by Express, will expose a RESTful API with endpoints for POST (create a new post), GET (retrieve all posts and a single post by ID), PUT (update a post), and DELETE (remove a post). All post data must be stored persistently in a MongoDB database. **(Roll Number 1 to 20)**
- 2. Project: A Simple E-commerce Product Catalog** Design and build a responsive web page that acts as a product catalog. The Express back end will handle all product data. The front end will display products in a grid. An admin panel (or separate web page) will allow an administrator to **add** new products, **view** a list of all products, **modify** a product's details (name, price, image), and **remove** products. The product information will be stored in a MongoDB collection. **(Roll Number 21 to 40)**
- 3. Project: A User Profile Dashboard** Create a responsive web application for managing user profiles. The home page will show a list of user accounts. The Express back end will provide an API to **create** a new user account, **retrieve** a single user's profile details, **update** their information (like bio or contact details), and **delete** their profile. All user data, including a unique ID and a password, will be stored securely in a MongoDB database. **(Roll Number 41 to 60)**
- 4. Project: A Recipe Sharing Platform** Develop a responsive web application where users can share recipes. The front end will have a submission form for new recipes and a page to browse existing ones. The Express backend will implement CRUD functionality: POST /recipes to submit a new recipe, GET /recipes to retrieve a list of all recipes, GET /recipes/:id to view a specific recipe, PUT /recipes/:id to edit a recipe, and DELETE /recipes/:id to delete a recipe. All recipe data (ingredients, instructions, title) will be stored in MongoDB. **(Roll Number 61 Onwards)**