Campus Assessment -Shipsy

1. Schema Design

Object Chosen: Task Manager
Reason: Simple yet demonstrates text, enum, and boolean fields.

Field Name	Type	Description
id	UUID/String	Unique task identifier
title	String	Short title for the task
description	Text	Detailed explanation of the task
priority	Enum	One of [Low, Medium, High]
isCompleted	Boolean	True if task is done
createdAt	DateTime	Automatically set on creation

2. Pseudocode for Module & Class Structure

```
Application
├── AuthController
│      ├── login(username, password)
│      └── validateUser()
│
├── TaskController
│      ├── createTask(title, description, priority, isCompleted)
│      ├── getTasks(filters, page)
│      ├── updateTask(id, data)
│      └── deleteTask(id)
│
├── Routes
│      ├── POST /api/login        → AuthController.login
│      ├── POST /api/tasks        → TaskController.createTask
│      ├── GET /api/tasks         → TaskController.getTasks
│      ├── PUT /api/tasks/:id      → TaskController.updateTask
│      └── DELETE /api/tasks/:id → TaskController.deleteTask
│
├── Views
│      ├── loginPage.html
│      ├── taskForm.html
│      └── taskList.html
│
└── server.js → Initializes application & binds routes
```

3. Test Plan

Test Case	Input	Expected Output
Positive: Create Task	{title: 'Buy milk', priority: 'High'}	201 Created ☐ Task object with ID
Positive: Fetch Tasks	GET /api/tasks	List of tasks with pagination info
Negative: Missing Title	{priority: 'Low'}	400 Bad Request ☐ Error: 'Title is required
Negative: Invalid Priority	{title: 'Test', priority: 'Urgent'}	400 Bad Request ☐ Error: 'Invalid priority
Edge: Title too long	{title: <256 chars>, priority: 'Low'}	400 Bad Request ☐ Error: 'Title too long
Edge: Empty Pagination	GET /api/tasks?page=99	Empty array with total=0

4. Reflection

If given more time,

 I would:

- Integrate JWT-based authentication for security.

- Replace in-memory storage with PostgreSQL or MongoDB for persistence.

 - Add search functionality and advanced filtering.

- Write unit and integration tests using Jest.

- Implement CI/CD pipeline for automated deployment and testing.

5.AI Prompt History with Reasoning

1.　　　"Generate Node.js Express routes for CRUD operations on a task object" ☐ To speed up development and ensure proper REST structure.

2.　　　"Suggest a schema for a task management system with priority and completion status" ☐ To validate database design.

3.　　　"List edge cases for creating a task in a task manager app" ☐ To improve coverage in the test plan.

4.　　　"Create basic HTML & CSS for a simple login page" ☐ To ensure a clean and responsive design quick