

## **MERN Stack Training**

**Company:** Sensation Software Solutions

**Student Name:** Gurvinder Singh

**Training Duration:** 6 Months

**Days:** 87

---

### **Objective of the Day**

The objective of **Day 87** was to implement the **complete CRUD functionality** for tasks and configure secure **file upload handling** using Multer. The goal was to ensure that tasks can be created, updated, fetched, and deleted efficiently with optional image uploads.

---

### **Introduction to API Development**

The trainer started the day by explaining RESTful API principles and how frontend applications communicate with backend servers. Emphasis was placed on HTTP methods such as **GET, POST, PUT, and DELETE**.

---

### **Multer Middleware Configuration**

A custom Multer middleware was implemented with:

- File size limitation
- Allowed formats (JPEG, PNG)

- Rejection of unsupported files

This ensured secure and optimized file handling.

---

## Create Task API

The **POST /api/tasks** endpoint was implemented to:

- Parse multipart form data
- Upload image if provided
- Generate image URL
- Save task data to MongoDB

Special handling was done to allow task creation **with or without an image**.

---

## Other CRUD APIs Implementation

Additional APIs were implemented:

- **GET /api/tasks** – Fetch all tasks
- **GET /api/tasks/:id** – Fetch single task
- **PUT /api/tasks/:id** – Update task data
- **DELETE /api/tasks/:id** – Delete task

Filtering by status was added using query parameters.

---

## **API Testing Using Postman**

All APIs were thoroughly tested using Postman to verify:

- Data validation
  - Error handling
  - File upload functionality
  - Filtering logic
- 

## **Learning Outcome**

- Strong understanding of CRUD APIs
  - Hands-on experience with file uploads
  - Backend debugging and testing skills
- 

## **Conclusion**

Day 87 successfully completed the **backend development phase**, delivering a fully functional task management API system.

.