

SRM Madurai College for Engineering and Technology

Department: Information Technology
College Code: 9111
Student NM-ID: EEC121412BECD8580134A2ADB6F76377
Roll No: 23IT18
Date: 15.09.25
Phase: Phase 1
Technology Project Name: IBM FE To-Do List Application
Submitted by: Karishma.R
Mobile: 7904410709

Introduction

A To-Do List application assists users in managing and tracking daily tasks efficiently. This project aims to develop a simple, responsive web application integrated with a Node.js REST API. Phase 1 covers the problem statement, user identification, MVP (Minimum Viable Product) scope, and acceptance criteria.

Problem Statement

Students and professionals often struggle to organize their daily work. Existing solutions may be overly complex or lack clear API integration examples. There is a need for a lightweight, beginner-friendly To-Do List application that demonstrates REST API usage.

Objectives

- Build an easy-to-use web application.
- Demonstrate integration between frontend (React) and backend (Node.js REST API).
- Provide CRUD (Create, Read, Update, Delete) operations for tasks.
- Ensure the application is responsive, secure, and fast for task management.

Users & Stakeholders

Stakeholder	Role
End Users	Students / Working Professionals
Developers	Team implementing frontend & REST API
Faculty / Mentors	Guide & Evaluate
IBM / College	Platform & Curriculum Support

User Stories

- As a user, tasks can be added with a title and description.
 - Tasks can be marked as complete or incomplete.
 - Tasks can be edited or deleted.
 - As an admin/developer, API responses and logs can be monitored.
-

MVP Features

- Add, view, edit, and delete tasks.
 - Mark tasks as done or pending.
 - Optional simple login for authentication.
 - Data persistence using local storage or a database.
 - Responsive design for both desktop and mobile devices.
-

Wireframes (Sample)

- **Home Screen:** Header, task list, and “Add Task” button.
- **Task Form:** Fields for title, description, and save/cancel options.
- **List Controls:** Options to mark as done, edit, or delete tasks.

(Insert hand-drawn wireframes or Figma screenshots in the PPT as required.)

API Endpoint List & Acceptance Criteria

Method	Endpoint	Description
GET	/tasks	Fetch all tasks
POST	/tasks	Add a new task
PUT	/tasks/:id	Update a task
DELETE	/tasks/:id	Delete a task

Acceptance Criteria

- Users can create, update, delete, and mark tasks.
 - Data persists (local state or database).
 - API functions correctly in Postman.
 - Interface is clean and mobile-friendly.
 - Code is versioned on GitHub.
-

