**COURSE NAME Full Stack Development ( Lab)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Code** | **Course Name** | **Teaching Scheme**  **(Teaching Hours)** | | | **Credits Assigned** | | | |
| **Theory** | **Practical** | **Tutorial** | **Theory** | **Practical** | **Tut** | **Total** |
| NITPL52 | Full Stack Development |  | 2 |  |  | 1 |  | 1 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Examination Scheme** | | | | | | | |
| **Course**  **Code** | **Course Name** | **Theory** | | | **Term Work** | **Practical**  **&**  **Oral** | **Total** |
| **Internal Assessment** | | **End Sem**  **Exam** |
| **Mid-Term    Test** | **Continuous Assessment** |
| NITPL52 | Full Stack Development |  |  |  | 50 | 25 | 75 |

|  |  |
| --- | --- |
| **Course Prerequisite:** Web Programming with Basics of HTML, CSS, JavaScript, React fundamentals, Node.js, Express, and MongoDB basics, Git/GitHub familiarity | |
| **Course Objectives:** | |
| 1 | Build responsive full-stack apps using the MERN stack |
| 2 | Use Tailwind CSS to create modern, scalable UIs |
| 3 | Manage state and data flow efficiently with Redux or Context API |
| 4 | Secure full-stack apps with authentication and authorization |
| 5 | Enable real-time features using WebSockets |
| 6 | Test and deploy applications with DevOps best practices |
| **Course Outcomes:**  After successful completion of the course students will be able to: | |
| 1 | **Explain the architecture and workflow of full stack web development** using the MERN stack and modern development tools. |
| 2 | **Design and develop responsive user interfaces** using React and Tailwind CSS, applying utility-first CSS principles and interactive UI components. |
| 3 | **Implement state management and asynchronous data handling** using React Hooks, Redux or Context API, and integrate third-party REST/GraphQL APIs. |
| 4 | **Build and secure backend applications** using Node.js, Express, and MongoDB, incorporating authentication, authorization, and middleware patterns. |
| 5 | **Integrate front-end and back-end systems** to develop dynamic, real-time full-stack applications using REST APIs and WebSockets. |
| 6 | **Test, deploy, and monitor full-stack applications** using CI/CD pipelines, Docker, GitHub Actions, and deployment platforms such as Vercel and Render. |

|  |  |
| --- | --- |
| **Suggested Experiments:** Students are required to complete all experiments. | |
| Star (\*) marked experiments are compulsory. | |
| **Sr. No.** | **Name of the Experiment** |
| 1\* | Build responsive and interactive UIs using **Tailwind CSS** |
| 2\* | Experiment based on React Hooks (useEffect, useContext, custom hooks) |
| 3\* | Manage complex state with **Redux or Context API** |
| 4\* | REST API Design with MongoDB + Mongoose Integration, |
| 5\* | Create secure, production-ready **RESTful APIs** |
| 6\* | Implement **authentication and user roles** with JWT |
| 7\* | Deploy full-stack apps using **DevOps tools and Docker** |
| 8\* | Enable **real-time communication** via WebSockets |
| 9\* | Containerizing App with Docker |
| 10\* | CI/CD Deployment with GitHub Actions + Render/Vercel |

### Mini Project Suggestion:

### Build and deploy a full-stack collaborative task manager or chat app incorporating: Responsive UI with Tailwind CSS, React with Hooks and Context/Redux, RESTful API with authentication, Real-time updates via WebSockets, Containerization and CI/CD pipeline

### Note: Suggested List of Experiments is indicative. However, flexibility lie with individual course instructor to design and introduce new, innovative and challenging experiments, (limited to maximum 30% variation to the suggested list) from within the curriculum, so that the fundamentals and applications can be explored to give greater clarity to the students and they can be motivated to think differently.

|  |  |
| --- | --- |
| **Term Work:** | |
| 1 | Term work should consist of 10 experiments. |
| 2 | The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work. |
| 3 | Total 50 Marks  (Experiments: 30-Marks, Mini Project: 15 Marks, Attendance : 5-Marks) |

Practical/Oral Exam: 25 Marks Practical/Oral Exam will be held based on the above syllabus.