INSTITUTE	**FACULTY OF TECHNOLOGY**
PROGRAM	**BACHELOR OF TECHNOLOGY (COMPUTER ENGINEERING)**
SEMESTER	**4**
COURSE TITLE	**ADVANCED WEB TECHNOLOGY**
COURSE CODE	**01CE1412**
COURSE CREDITS	**4**

Objective:

1 The objective of this syllabus is to equip students with advanced, industry-aligned skills in full-stack web development using modern technologies such as TypeScript, React, Node.js, Express, Next.js, and Socket.io, with a strong focus on scalability, security, and employability.

Course Outcomes: After completion of this course, student will be able to:

1 Apply JavaScript and TypeScript features to develop efficient and maintainable code.

2 Build interactive, scalable user interfaces using React with industry-relevant patterns

3 Analyze backend architectures to design secure and optimized services using Node.js, Express, and API integrations

4 Evaluate and implement full-stack architectures using TypeScript for scalability, maintainability, and performance optimization.

5 Apply modern frameworks and tools (Next.js, Socket.io) to enhance performance and interactivity in full-stack applications

Pre-requisite of course: Basic HTML, CSS, JavaScript, and prior knowledge of MongoDB

Teaching and Examination Scheme

Theory Hours	Tutorial Hours	Practical Hours	ESE	IA	CSE	Viva	Term Work
2	0	4	50	30	20	25	25

Contents : Unit	Topics	Contact Hours
1	**JavaScript & TypeScript Essentials** JavaScript refresher, variables, functions, arrays, ES6+ features (arrow functions, destructuring, template literals, modules, promises, async/await);, TypeScript fundamentals: static typing, interfaces,, classes, generics; Setting up a TypeScript project for frontend & backend.	5

DR. KRUNAL NARUMAL VAGHELA
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Contents : Unit	Topics	Contact Hours
2	**React.js Frontend Development** React fundamentals, components,, props, state, Hooks (useState, useEffect, useContext);, Forms & validation, Routing with React Router, API integration (Fetch, Axios), State management basics with Context API	10
3	**Backend Development with Node.js & Express:** Node.js setup & NPM, Express.js basics, routing, middleware, request/response cycle , REST API principles & creation, Request validation & error handling , Database connectivity (MongoDB/MySQL refresher); API testing with Postman.	6
4	**Web Application Security & Authentication:** Authentication basics: session vs token-based; JWT authentication flow, Role-based access control; Password hashing (bcrypt); API security: CORS,, Rate limiting, input sanitization; Environment variables & configuration security.	3
5	**Modern Web Practices:** Next.js basics: file-based routing, SSR, SSG, API routes;, Introduction to GraphQL (queries & mutations), Real-time communication with Socket.io, Performance considerations;, Best practices for structuring full-stack applications	4
Total Hours		**28**

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
1	**Practical 1** Develop a modular utility application (e.g., task calculator or data formatter) applying ES6+ syntax and TypeScript typing for maintainability and error prevention.	4
2	**Practical 2** Design a data-fetching module (e.g., weather or crypto dashboard) utilizing async/await and Promises for responsive API-driven functionality.	4
3	**Practical 3** Build a multi-component React interface (e.g., to-do manager or notes app) employing props and state for modular UI interaction.	4
4	**Practical 4** Create a React-based feedback form capturing and validating user data with controlled components and error feedback.	4
5	**Practical 5** Develop a multi-page product catalog application implementing routing and Context API-based state management for cart functionality.	4

DR. KRUNAL NARUMAL VAGHELA
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
6	**Practical 6** Create a backend service for student or product management performing CRUD operations and handling client requests via RESTful APIs.	4
7	**Practical 7** Implement secure login and authorization in the backend using JWT tokens and middleware validation for protected resources.	4
8	**Practical 8** Build a live chat or notification service enabling real-time event communication between multiple clients and server.	4
9	**Practical 9** Construct a React module (e.g., user profile or dashboard) using TypeScript interfaces and generics to ensure reliable and type-safe component behavior.	4
10	**Practical 10** Develop a Node.js + Express API using TypeScript that provides type- safe endpoints for a given dataset (e.g., employees or courses).	4
11	**Practical 11** Design a server-rendered blog or portfolio application using Next.js for SEO optimization and faster page rendering.	4
12	**Practical 12** Integrate a Next.js frontend with a Node.js backend API to perform CRUD operations and visualize real-time updates.	4
13	**Practical 13** Optimize bundle size, implement environment management, and deploy the full-stack app on a live platform (Vercel/Render).	4
14	**Practical 14** Conceive, design, and implement a mini full-stack application integrating React, Node.js, TypeScript, and Next.js to solve a real-world problem scenario.	4
Total Hours		**56**

Textbook :

1 The Definitive Guide, Flanagan, D. – JavaScript, O'Reilly Media, 2020

References:

1 Pro TypeScript: Application-Scale JavaScript Development, Pro TypeScript: Application-Scale JavaScript Development, Freeman, A., & Auer, S, Apress, 2019

2 Learning React: Modern Patterns for Developing React Apps, Learning React: Modern Patterns for Developing React Apps, Banks, A., & Porcello, E., O'Reilly Media ,3rd Edition, 2020

3 Node.js Web Development, Node.js Web Development, Herrington, J., Packt Publishing, 2022

References:

4 Building Server-side Rendering React Applications with Next.js,, Building Server-side Rendering React Applications with Next.js,, Chinnathambi, M. – Learning Next.js, Packt Publishing, 2022

Suggested Theory Distribution:

The suggested theory distribution as per Bloom's taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery					
Remember / Knowledge	**Understand**	**Apply**	**Analyze**	**Evaluate**	**Higher order Thinking / Creative**
0.00	0.00	60.00	20.00	20.00	0.00

Instructional Method:

1 The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use multimedia presentations, animations, and real-world examples.

2 The internal evaluation will be done on the basis of continuous evaluation of students in the class-room.

3 Students will use supplementary resources such as Coursera, NPTEL videos, e-courses, etc..

Supplementary Resources:

1 https://developer.mozilla.org/en-US/docs/Web/JavaScript (MDN Web Docs – JavaScript & TypeScript)

2 https://react.dev/ (Official React Documentation)

3 https://nodejs.org/en/docs (Official Node.js Documentation)

4 https://nextjs.org/learn (Official Next.js Learning Resources)

5 https://socket.io/docs/v4 (Socket.io Documentation for Real-time Communication)

6 https://graphql.org/learn/ (GraphQL Official Learning Guide)

DR. KRUNAL NARUMAL VAGHELA
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)