**Designing with Figma & AI**

* Overview of Figma and its features
* Creating a Figma account and navigating the interface
* Understanding Figma's workspace (panels, toolbar, properties)
* Basic drawing tools and shapes
* Grouping, aligning, and arranging elements
* Elements of UI Design: Color, Typography, and Iconography
* Grids and Layout Systems
* Auto Layouts and its use
* Creating reusable components with variants
* Creating interactive prototypes, adding and managing interactions
* Using transitions and animations
* Using auto-layout for responsive design
* Introduction to AI in UI/UX design
* AI-powered design tools (Adobe Sensei, Uizard, Figma AI plugins)

**Introduction to HTML & Web Page Structure**

* What is HTML? Role in Web Development
* Structure of an HTML Document
* Basic HTML Elements (Headings, Paragraphs, Links, Lists)
* Images & Figure
* Hands-on: Creating a Simple Web Page

**AI-Assisted HTML Forms & Tables**

* Forms & Input Fields (Text, Email, Password, Buttons)
* Form Validation & Attributes
* Creating Tables in HTML
* Using AI for Form Validation & Autocomplete Suggestions
* Hands-on: Building a Registration Form

**Semantic HTML & Best Practices**

* Understanding Semantic Tags (article, section, nav, footer)
* SEO-Friendly HTML Markup
* Using AI for SEO Optimization & HTML Cleanup
* Hands-on: Building a Blog Layout with Semantic HTML

**Version Control with Git**

* Introduction to version control
* Installing and setting up Git
* Basic Git commands (init, clone, add, commit, push, pull)
* Understanding branches and merging
* Resolving merge conflicts
* Working with remote repositories (GitHub)

**Introduction to CSS & AI in Design**

* What is CSS?
* Types of CSS: Inline, Internal, External
* CSS Syntax & Selectors
* CSS Units (px, em, rem, %, vw, vh)
* AI in Web Design:
  + AI-powered CSS generators (e.g., CSS Scan, Pika)
  + AI-based color palette tools (e.g., Khroma, Colormind)

**Styling Text & AI-Driven Typography**

* Font Properties (font-size, font-family, font-weight, line-height)
* Text Decoration & Alignment
* Using Google Fonts & AI font pairing tools (e.g., FontJoy)
* Implementing AI-generated typography for better readability

**CSS Position Layout**

* Static, Relative, Absolute, Sticky & Fixed
* Z-index
* Overflow
* Implementing AI-generated for Position

**CSS Layouts & AI-Enhanced Design**

* Display properties (block, inline, flex, grid)
* Box Model: margin, padding, border, width, height
* Flexbox & AI layout recommendations
* CSS Grid & AI-assisted responsive design tools
* AI-powered automatic layout generation (e.g., Figma’s AI features)

**Responsive Design & AI Optimization**

* Media Queries & Viewport
* CSS Variables & Custom Properties
* AI-generated adaptive designs for different screen sizes
* AI-powered automated testing for responsive designs (e.g., Google Lighthouse, BrowserStack)
* Introduction to AI-driven CSS frameworks (e.g., Tailwind CSS with AI utilities)

**CSS Animations, Transitions & AI Motion Design**

* CSS Transitions & Transformations (rotate, scale, translate)
* Keyframe Animations (@keyframes)
* CSS Animation Libraries (e.g., Animate.css, GSAP)
* AI-generated animations for UI/UX enhancements
* AI-assisted motion design (e.g., Framer Motion AI)

**Introduction to SASS/SCSS & AI in Styling**

* What is SASS?
* Differences between SASS and SCSS
* Setting up a development environment
* Installing SASS via CLI & package managers
* AI in CSS Preprocessing:
  + AI-powered CSS generators
  + AI-based color scheme and design tools (Colormind, Khroma)
  + Variables, Nesting, & AI-Generated Styles
* Variables, Nesting
* Mixins, Functions & AI-Enhanced Code Efficiency
* Extending & Modularizing with AI Automation
* Advanced Features & AI-Powered Frameworks

**UI Frameworks**

* Bootstrap
  + Colors, Spacing, and Typography
  + Bootstrap Grid System
  + Bootstrap Components
  + Creating a custom Bootstrap theme
* Tailwind CSS
  + Colors, Spacing, and Typography
  + Backgrounds, Borders, and Shadows
  + Flexbox & Grid system
  + Responsive design principles
  + Transitions & Animations

**JavaScript ES6+ Features & DOM Manipulation**

* Variables (var, let, const)
* Data Types & Operators
* Loops, Functions, and Scope
* ES6 Features (Arrow Functions, Spread Operator, Destructuring)
* Hands-on: Building a Calculator with JavaScript

**DOM Manipulation & Events**

* Selecting Elements (querySelector, getElementById)
* Handling Events (click, input, keypress)
* Modifying HTML & CSS Dynamically
* Hands-on: Building a To-Do List App

**DOM Manipulation & Events**

* Selecting Elements (querySelector, getElementById)
* Handling Events (click, input, keypress)
* Modifying HTML & CSS Dynamically
* Hands-on: Building a To-Do List App

**JQuery**

* Overview of jQuery and its advantages
* Basic jQuery syntax and selectors
* jQuery Events and Effects (show, hide, toggle, fade, slide etc)
* Using jQuery plugins

**Asynchronous JavaScript (Fetch API, Promises, Async/Await)**

* Callbacks & Promises
* Fetch API for HTTP Requests
* Async/Await for API Calls
* Hands-on: Fetching Data from an API and Displaying it on UI

**Introduction to React**

* JSX and Components
  + Creating functional and class components.
  + Understanding JSX syntax.
* State and Props
  + Managing component state.
  + Passing and validating props.
* Lifecycle Methods
  + Overview of React lifecycle phases.

**React-Router-Dom**

* Advanced routing concepts (dynamic routes, nested routes).

**Form Handling**

* Using Formik or React Hook Form for form validation.

**Introduction to APIs**

* Differences between REST, GRPC, SOAP, and GraphQL.

**Introduction to Next.js**

* Introduction to Next.js and its advantages
* Setting up a Next.js project
* Understanding project structure and file conventions

**Styling in Next.js**

* CSS Modules in Next.js
* Using Tailwind CSS with Next.js
* Styled Components and Emotion
* Global Styles and Theme Customization

**Next.js Routing and Navigation**

* Pages and File-based Routing
* Dynamic and Nested Routes

**Fetching and Managing Data**

* Fetching Data with fetch() and axios
* Server-side Rendering (SSR) vs Static Site Generation (SSG)
* Incremental Static Regeneration (ISR)

**State Management in Next.js**

* Context API and useReducer Hook

**Performance Optimization**

* Image Optimization with Next.js Image Component
* Lazy Loading and Code Splitting
* Optimizing API Calls and Caching
* SEO Best Practices in Next.js

**AI Integration in React/Next.js**

* Adding AI-powered features like chatbots or recommendation systems

**Deployment and Hosting in React/Next.js**

* Deploying Next.js on Vercel

**Projects (As per the Institute Rules, all four of the following projects will be done by the Student themselves! On ReactJs / NextJs)**

* E-commerce Website
* Corporate Website
* News Portal Website
* Travel & Tours Website