300DevDays Curriculum Overview

Month 1: HTML Fundamentals
Week 1: Introduction to HTML
Week 2: HTML Structure and Elements
Week 3: HTML Forms and Input
Week 4: HTML Semantics and Best Practices

Month 2: CSS Styling
Week 1: Introduction to CSS
Week 2: CSS Selectors and Properties
Week 3: CSS Layout and Box Model
Week 4: CSS Flexbox and Grid

Month 3: JavaScript Basics
Week 1: Introduction to JavaScript
Week 2: Variables and Data Types
Week 3: Functions and Control Flow
Week 4: DOM Manipulation with JavaScript

Month 4: Advanced JavaScript
Week 1: Object-Oriented Programming in JavaScript
Week 2: Asynchronous JavaScript and Promises
Week 3: ES6+ Features and Modules
Week 4: JavaScript Best Practices

Month 5: Front-End Frameworks
Week 1: Introduction to Vue.js
Week 2: Vue Components and Routing
Week 3: Introduction to React
Week 4: React Components and State Management

Month 6: Back-End Development with Node.js
Week 1: Introduction to Node.js
Week 2: Building RESTful APIs with Node.js
Week 3: Express.js Framework
Week 4: Data Storage with MongoDB

Month 7: Advanced Back-End Development
Week 1: Authentication and Authorization
Week 2: SQL Databases and PostgreSQL
Week 3: API Security and Best Practices
Week 4: Deployment and Hosting
Month 8-10: Project-Based Learning

- During these three months, students will work on real-world projects using the technologies they've learned.
- Each project will focus on a specific aspect of full-stack engineering and will be mentored and coached by experienced professionals.
- Students will participate in interactive workshops and live classes for project guidance and knowledge reinforcement.
- Throughout the entire 10-month program, there will be ongoing mentorship and coaching, as well as self-paced learning opportunities.

- The goal is to provide a comprehensive, hands-on education in full-stack engineering, preparing students for real-world scenarios and industry demands.

Month 1 - HTML Fundamentals

Week 1: Introduction to HTML

- Class 1: Introduction to Web Development
  - Understanding the internet and how websites work
  - History and evolution of HTML
  - Setting up a development environment
- Class 2: Structure of an HTML Document
  - Basic HTML structure: `<!DOCTYPE>, <html>, <head>, and <body>`
  - HTML tags, elements, and attributes
  - Creating your first HTML page

Week 2: HTML Structure and Elements
-- Class 3: Text Formatting and Headings

- Using text formatting tags: `<p>, <h1> to <h6>, <strong>, <em>, etc.`
- Creating lists: `<ul>, <ol>, and <li>`
- Class 4: Links and Images
  - Creating hyperlinks with `<a>` tags
  - Embedding images with tags
  - Best practices for image optimization

Week 3: HTML Forms and Input

- Class 5: Building Forms
  - Creating forms with `<form>` elements
  - Adding form controls: text fields, checkboxes, radio buttons, and dropdowns
  - Form submission and the `<button>` element
- Class 6: HTML Semantic Elements
  - Understanding semantic HTML: `<header>, <nav>, <article>, <section>, etc.`
  - Enhancing accessibility and SEO with semantic elements

Week 4: HTML Semantics and Best Practices

- Class 7: Multimedia Elements
  - Embedding audio and video: `<audio>` and `<video>` elements
  - Using `<iframe>` for embedding external content
  - Web accessibility considerations for multimedia
- Class 8: HTML5 APIs and Best Practices
  - Introduction to HTML5 APIs: localStorage, sessionStorage, and more
  - Best practices for writing clean and semantically correct HTML
  - Review and recap of Month 1 content

Throughout the month, there will be practical exercises and assignments to reinforce the concepts covered in each class. Students will also have access to resources and support for self-paced learning.

At the end of Month 1, students should have a solid understanding of HTML fundamentals, enabling them to create well-structured web pages.

Month 2 - CSS Styling

Week 1: Introduction to CSS

- Class 1: Cascading Style Sheets (CSS) Introduction
  - What is CSS and its role in web development
  - Inline vs. internal vs. external CSS
  - Basic CSS syntax and selectors
- Class 2: CSS Selectors and Properties
  - Understanding CSS selectors: element, class, ID, and pseudo-classes
  - Applying styles with CSS properties: colour, font-size, margin, padding, etc.
  - Using the "box model" concept

Week 2: CSS Layout and Box Model

- Class 3: Box Model in Depth
  - Understanding the box model: content, padding, border, margin
  - Box sizing and how it affects layout
  - Creating responsive layouts with CSS
- Class 4: CSS Positioning and Display
  - CSS positioning: static, relative, absolute, fixed
  - Display property: block, inline, inline-block
  - Centering elements horizontally and vertically

Week 3: CSS Flexbox and Grid

- Class 5: Introduction to CSS Flexbox
  - Flexbox fundamentals: containers and items
  - Creating flexible and responsive layouts with Flexbox
  - Solving common layout challenges
- Class 6: CSS Grid Layout
  - Introduction to CSS Grid
  - Defining grid containers and grid items
  - Building complex grid-based layouts

Week 4: Responsive Web Design

- Class 7: Media Queries and Responsive Design
  - Media queries and breakpoints
  - Creating responsive designs for various devices
  - Mobile-first vs. desktop-first approaches
- Class 8: CSS Best Practices and Optimization
  - CSS performance optimization techniques
  - Organising CSS with methodologies like BEM or SMACSS
  - Review and recap of Month 2 content

Throughout the month, students will practice applying styles to HTML content, building progressively more complex layouts. They will also work on responsive design projects to ensure their websites adapt to different screen sizes.

Practical exercises and assignments will reinforce the concepts learned in each class, and students will have access to resources for self-paced learning. By the end of Month 2, students should have a strong foundation in CSS styling and layout techniques.

Month 3 JavaScript Basics

Week 1: Introduction to JavaScript

- Class 1: JavaScript Fundamentals
  - What is JavaScript and its role in web development
  - JavaScript's evolution and compatibility
  - Setting up a JavaScript development environment
- Class 2: Variables and Data Types
  - Declaring variables: var, let, const
  - Primitive data types: string, number, boolean, undefined, null
  - Type coercion and type conversion

Week 2: Functions and Control Flow

- Class 3: Functions in JavaScript
  - Defining functions: function declarations vs. expressions
  - Function parameters and return values
  - Scope and closures
- Class 4: Conditional Statements and Loops
  - If statements, switch statements
  - Iteration with for, while, and do-while loops
  - Control flow and decision-making in JavaScript

Week 3: DOM Manipulation with JavaScript

- Class 5: Introduction to the Document Object Model (DOM)
  - Understanding the DOM and its structure
  - Accessing and manipulating DOM elements
  - Event handling and listeners
- Class 6: DOM Manipulation and Interactive Web Pages
  - Modifying HTML content and attributes
  - Creating and removing DOM elements dynamically
  - Practical exercises in building interactive web pages

Week 4: Advanced JavaScript Concepts

- Class 7: Object-Oriented Programming in JavaScript
  - Objects and prototypes
  - Constructors and classes (ES6)
  - Object-oriented programming principles
- Class 8: Asynchronous JavaScript and Promises
  - Asynchronous programming with callbacks
  - Introduction to Promises for handling async operations
  - Error handling with try...catch
  - Review and recap of Month 3 content

Throughout Month 3, students will practice writing JavaScript code to manipulate the DOM, create interactive web pages, and gain a solid understanding of fundamental programming concepts. Practical exercises and assignments will reinforce their knowledge, and resources will be provided for self-paced learning.

By the end of Month 3, students should have a strong foundation in JavaScript basics, ready to build more complex web applications in the following months.

Month 4 - Advanced JavaScript

Week 1: Object-Oriented Programming in JavaScript

- Class 1: Objects and Prototypes
  - Understanding objects in JavaScript
  - Prototypes and inheritance
  - Object manipulation and extensions
- Class 2: Constructors and Classes (ES6)
  - Constructor functions and their usage
  - Introduction to ES6 classes
  - Class constructors, methods, and properties

Week 2: Asynchronous JavaScript and Promises

- Class 3: Asynchronous Programming with Callbacks
  - Understanding asynchronous JavaScript
  - Callback functions and their usage
  - Callback hell and its solutions
- Class 4: Introduction to Promises
  - Promise basics: creating, resolving, and rejecting promises
  - Chaining promises for sequential operations
  - Error handling with promises and the .catch() method

Week 3: ES6+ Features and Modules

- Class 5: ES6+ Features
  - Arrow functions and their benefits
  - Template literals and string interpolation
  - Destructuring assignments and the spread operator
- Class 6: ES6 Modules and Module Systems
  - Import and export statements in ES6
  - Organizing code into modules
  - Using third-party libraries and modules

Week 4: JavaScript Best Practices

- Class 7: Code Quality and Style
  - Writing clean and maintainable code
  - Code formatting and linting tools
  - Code commenting and documentation
- Class 8: JavaScript Debugging, Performance & NodeJS
  - Debugging techniques and tools
  - Profiling and optimizing JavaScript code
  - Browser developer tools and extensions
  - Review and recap of Month 4 content
  - Introduction to NodeJS

Throughout Month 4, students will deepen their understanding of JavaScript by exploring advanced topics such as object-oriented programming, asynchronous programming, and ES6+ features. Practical exercises, coding challenges, and debugging sessions will help solidify their knowledge. Additionally, best practices for writing efficient and maintainable code will be emphasised.

By the end of Month 4, students should be well-versed in advanced JavaScript concepts and ready to tackle more complex web development tasks.

Month 5 - Front-End Frameworks

Week 1: Introduction to Vue.js

- Class 1: Getting Started with Vue.js
  - What is Vue.js and its advantages
  - Setting up a Vue.js project
  - Vue.js templates, directives, and data binding
- Class 2: Vue Components and Routing
  - Creating Vue components
  - Understanding component communication
  - Introduction to Vue Router for navigation

Week 2: Advanced Vue.js Features

- Class 3: State Management with Pinia
  - Managing application state with Pinia
  - Understanding Pinia components & structure
  - Building a Pinia store
- Class 4: Advanced Vue Directives
  - Exploring custom directives
  - Event handling and modifiers
  - More Vue Topics
  - Practical exercises with Vue directives

Week 3: Introduction to React

- Class 5: React Fundamentals
  - Introduction to React and JSX
  - Components and props
  - Creating a React application
- Class 6: React Components and State Management
  - Managing component state
  - Component lifecycle methods
  - React Hooks for functional components

Week 4: React Components and State Management

- Class 7: React Router and Navigation
  - Routing in React with React Router
  - Nested routes and route parameters
  - Building navigation menus
- Class 8: Comparing Vue.js and React
  - A comparative analysis of Vue.js and React
  - When to use Vue.js vs. React
  - Review and recap of Month 5 content

Throughout Month 5, students will explore two popular front-end JavaScript frameworks, Vue.js and React. They will learn how to build interactive user interfaces, manage component state, and handle routing for single-page applications.

Practical projects and exercises will help students gain hands-on experience with these frameworks. By the end of Month 5, students should have a strong understanding of front-end development using Vue.js and React, setting the stage for more advanced projects in the following months.

Month 6 - Back-End Development with Node.js

Week 1: Introduction to Node.js

- Class 1: Node.js Fundamentals
  - Understanding Node.js and its role in back-end development
  - Setting up a Node.js project and environment
  - Node.js core modules and npm (Node Package Manager)
- Class 2: Building Your First Node.js Server
  - Creating a basic HTTP server with Node.js
  - Handling HTTP requests and responses
  - Routing and middleware in Node.js

Week 2: Building RESTful APIs with Node.js

- Class 3: Introduction to RESTful APIs
  - What are RESTful APIs and their principles
  - Designing RESTful routes and endpoints
  - Handling HTTP methods: GET, POST, PUT, DELETE
- Class 4: Express.js Framework
  - Introduction to Express.js
  - Setting up an Express application
  - Routing with Express and middleware usage

Week 3: Data Storage with MongoDB

- Class 5: Introduction to Databases
  - Understanding databases and their types
  - Introduction to NoSQL databases
  - Setting up and connecting to MongoDB
- Class 6: MongoDB and Mongoose
  - CRUD operations with MongoDB
  - Introduction to Mongoose ODM (Object Data Modeling)
  - Defining schemas and models in Mongoose

Week 4: Advanced Back-End Development

- Class 7: Authentication and Authorization
  - Implementing user authentication with Passport.js
  - Role-based access control (RBAC)
  - Securing RESTful APIs
- Class 8: SQL Databases and PostgreSQL
  - Introduction to SQL databases
  - Setting up a PostgreSQL database
  - SQL queries and data manipulation
  - Review and recap of Month 6 content

Throughout Month 6, students will focus on back-end development using Node.js and related technologies. They will learn how to create RESTful APIs, manage data with MongoDB, and secure their applications. Additionally, they will gain exposure to SQL databases and PostgreSQL, broadening their database knowledge. Practical exercises and projects will allow students to apply their skills in building robust server-side applications.

By the end of Month 6, students should be proficient in back-end development with Node.js, ready to tackle more complex projects in the subsequent months.

Month 7 - Advanced Back-End Development

Week 1: Authentication and Authorisation

- Class 1: User Authentication
  - Implementing user registration and login
  - Using JSON Web Tokens (JWT) for authentication
  - Securing routes and protecting sensitive data
- Class 2: Role-Based Access Control (RBAC)
  - Understanding RBAC principles
  - Assigning roles and permissions
  - Managing user roles and access levels

Week 2: SQL Databases and PostgreSQL

- Class 3: Introduction to SQL Databases
  - Exploring SQL databases and their advantages
  - SQL vs. NoSQL databases
  - Setting up a PostgreSQL database
- Class 4: SQL Queries and Data Manipulation
  - Writing SQL queries: SELECT, INSERT, UPDATE, DELETE
  - Joins and relationships between database tables
  - Handling transactions and database integrity

Week 3: API Security and Best Practices

- Class 5: API Security Best Practices
  - Securing RESTful APIs against common threats
  - Rate limiting and API abuse prevention
  - Handling CORS (Cross-Origin Resource Sharing)
- Class 6: Documentation and Testing
  - Generating API documentation with tools like Swagger
  - Writing unit tests and integration tests
  - Test-driven development (TDD) principles

Week 4: Deployment and Hosting

- Class 7: Deployment Strategies
  - Preparing applications for deployment
  - Deployment options: cloud platforms, VPS, and PaaS
  - Continuous integration and continuous deployment (CI/CD)
- Class 8: Performance Optimization
  - Performance profiling and monitoring
  - Caching strategies
  - Scaling strategies and load balancing
  - Review and recap of Month 7 content

Throughout Month 7, students will focus on advanced back-end development topics, including authentication, authorization, SQL databases, and API security. They will also learn best practices for documenting and testing their APIs. Additionally, students will explore deployment strategies and performance optimization techniques to ensure their applications are efficient and scalable. Practical projects and hands-on exercises will help students apply these concepts to real-world scenarios.

By the end of Month 7, students should have a strong grasp of advanced back-end development principles and be well-prepared for more complex challenges in the following months.

Month 8 - 10 - Project-Based Learning

Week 1: Project Planning and Ideation

- Class 1: Defining Project Scope and Objectives
  - Setting project goals and objectives
  - Identifying project stakeholders and their needs
  - Creating a project plan and timeline
- Class 2: Ideation and Brainstorming
  - Generating project ideas and concepts
  - Selecting a project idea and forming teams
  - Creating a project proposal and initial design

Week 2: Project Development and Prototyping

- Class 3: Building the MVP (Minimum Viable Product)
  - Defining project features and requirements
  - Prototyping
  - Selecting technologies and tools for the project
- Class 4: Development Sprints
  - Agile development methodologies
  - Sprint planning and execution
  - Version control and collaboration with Git

Week 3: Project Implementation and Testing

- Class 5: Implementing Project Features
  - Developing project components and functionality
  - Code reviews and quality assurance
  - Iterative development and continuous improvement
- Class 6: Testing and Quality Assurance
  - Types of testing: unit, integration, and end-to-end
  - Writing and executing test cases
  - Debugging and resolving issues

Week 4: Project Presentation and Deployment

- Class 7: Project Presentation
  - Preparing for project presentations
  - Demonstrating project features and functionality
  - Gathering feedback and making improvements
- Class 8: Deployment and Hosting
  - Deploying the project to a production environment
  - Configuring servers and databases
  - Ensuring security and performance in the live environment

Throughout Months 8 to 10, students will work on real-world projects, applying the knowledge and skills they've acquired in previous months. They will engage in the entire project lifecycle, from planning and ideation to implementation, testing, and deployment. Mentors and instructors will provide guidance and support, and students will participate in interactive workshops and live classes for project assistance and knowledge reinforcement.

By the end of this project-based learning period, students will have gained practical experience in full-stack development, working with industry professionals on meaningful projects. This experience will prepare them for real-world scenarios and

help them build a strong portfolio to showcase their skills to potential employers or clients.

Month 1 Workshops

Week 1: Introduction to HTML
Workshop Scenario 1: Building a Personal Website

- Scenario: Participants are tasked with creating a personal website using HTML. They can include an "About Me" section, contact information, and links to their social media profiles.
- Objectives: To apply the basics of HTML, including creating an HTML document, using headings, paragraphs, and links.
- Activities: Participants will start with a blank HTML file and gradually build their personal website, adding content and formatting.

Week 2: HTML Structure and Elements
Workshop Scenario 2: Creating a Recipe Page

- Scenario: Participants are asked to create a web page that displays a recipe. The page should include the list of ingredients, cooking instructions, and an image of the dish.
- Objectives: To practice structuring an HTML document, using lists, and embedding images.
- Activities: Participants will begin with an HTML template and fill in the content for their recipe page, applying appropriate HTML elements for organization.

Week 3: HTML Forms and Input
Workshop Scenario 3: Designing a Contact Form

- Scenario: Participants are challenged to design a contact form for a fictional website. The form should include fields for name, email, message, and a submit button.
- Objectives: To learn how to create HTML forms, input elements, and handle user input.
- Activities: Participants will create an HTML form, define form fields, and set up the form to submit data to a specified email address.

Week 4: HTML Semantics and Best Practices
Workshop Scenario 4: Building a Blog Post

- Scenario: Participants are provided with content for a blog post and are tasked with structuring it using semantic HTML elements. They must create a well-organized and semantically correct blog post.
- Objectives: To reinforce the importance of semantic HTML and best practices for content structure.
- Activities: Participants will review the provided blog content and restructure it using semantic HTML elements like
  ,
  ,
  , and
  .

These workshop scenarios are designed to be hands-on and practical, allowing participants to apply what they've learned during the weekly classes. They reinforce HTML fundamentals while engaging participants in creative and real-world scenarios.

Month 2 Workshops

Week 1: Introduction to CSS
Scenario 1: CSS Styling for a Personal Blog

In this workshop, students will create a simple personal blog webpage using HTML and CSS. They'll learn how to style headings, paragraphs, and links using CSS properties like font-size, color, and text-decoration. The objective is to create an aesthetically pleasing blog layout with a custom color scheme and typography.

Week 2: CSS Layout and Box Model
Scenario 2: Building a Responsive Portfolio Page

Students will build a portfolio webpage that showcases their skills and projects. They'll apply CSS layout techniques like flexbox and grid to create responsive sections for projects, contact information, and a bio. The workshop will focus on creating a visually appealing and responsive layout that adapts to various screen sizes.

Week 3: CSS Flexbox and Grid
Scenario 3: Designing a Product Showcase

In this workshop, students will design a product showcase page for an e-commerce website. They'll use CSS flexbox and grid to create a grid of product cards with images, titles, and prices. The workshop will emphasize the use of flexbox for aligning content within each card and grid for organizing the overall layout.

Week 4: Responsive Web Design
Scenario 4: Creating a Restaurant Menu

Students will create a responsive restaurant menu webpage using CSS media queries. The workshop will involve designing a menu layout that adjusts its appearance based on the device's screen size. Students will learn how to hide or show menu sections and items to provide an optimal user experience on both desktop and mobile devices.

Month 3 Workshops

Week 1: Introduction to JavaScript

Scenario 1: Interactive Quiz App

- Workshop Description: In this workshop, students will create a simple interactive quiz application using HTML and JavaScript. They will learn how to structure the quiz questions, validate user input, and provide feedback based on user responses. This scenario will reinforce the basics of JavaScript and DOM manipulation.

Scenario 2: To-Do List Application

- Workshop Description: Students will build a to-do list application from scratch. They will learn how to add and remove tasks dynamically, toggle task completion, and save tasks to local storage. This workshop will help students practice working with JavaScript functions, event listeners, and the DOM.

Week 2: Advanced JavaScript Concepts

Scenario 3: Image Slider

- Workshop Description: In this workshop, students will create an image slider using JavaScript and CSS. They will learn how to implement image transitions, navigation controls, and automatic sliding. This scenario will reinforce concepts like working with timers, event handling, and managing CSS styles dynamically.

Scenario 4: Interactive Form Validation

- Workshop Description: Students will build a form validation script that provides real-time feedback to users as they fill out a form. They will learn how to validate input fields for various criteria such as email addresses and passwords. This workshop will emphasize the use of regular expressions and advanced JavaScript techniques.

These workshops are designed to be hands-on and practical, allowing students to apply the JavaScript skills they've learned in a real-world context. Each scenario focuses on specific JavaScript concepts while encouraging problem-solving and creativity.

Month 4 Workshops

Week 1 Workshop: Object-Oriented JavaScript
Scenario: In this workshop, students will work on building a simple JavaScript application that models a virtual library. They will create JavaScript objects to represent books, authors, and libraries, exploring object-oriented programming concepts. Students will learn how to define constructors, create instances of objects, and use prototypes to add methods and properties to objects. They will practice organizing and managing data effectively using object-oriented principles.

Week 2 Workshop: Asynchronous JavaScript and Promises
Scenario: In this workshop, students will develop a weather application that fetches weather data from an external API asynchronously. They will work with asynchronous JavaScript, making API requests and handling responses using promises. Students will learn how to manage multiple asynchronous operations, chain promises for sequential tasks, and handle errors gracefully. This workshop will emphasize practical experience with asynchronous programming.

Week 3 Workshop: Exploring ES6+ Features
Scenario: In this workshop, students will work on a coding challenge that encourages them to use ES6+ features like arrow functions, template literals, and destructuring assignments. They will be tasked with refactoring an existing JavaScript codebase to make it more concise and readable using these modern JavaScript features. This workshop will highlight the benefits of ES6+ syntax and encourage students to adopt best practices in their coding.

Week 4 Workshop: Debugging and Performance Optimization
Scenario: In this workshop, students will analyze and optimize a web application's performance. They will identify bottlenecks and performance issues using browser developer tools and profiling techniques. Students will work on practical exercises to optimize code, implement caching strategies, and improve the application's load time. They will gain hands-on experience in debugging and optimizing JavaScript code for better performance.

These workshop scenarios align with the topics covered in Month 4 and provide students with valuable hands-on experience, reinforcing their understanding of advanced JavaScript concepts.

Month 5 Workshops

Certainly, here are four workshop scenarios, one for each week in Month 5, which focuses on Front-End Frameworks (Vue.js and React):

Week 1 Workshop: Vue.js Introduction

- Scenario: Build a To-Do List App with Vue.js
  - Description: In this workshop, students will create a simple To-Do List application using Vue.js. They will learn how to set up a Vue project, create Vue components, and use Vue directives for data binding. Students will build an interactive web app where they can add, update, and remove tasks.

Week 2 Workshop: Advanced Vue.js Features

- Scenario: Building a Blog with Vue Router and Pinia
  - Description: In this workshop, students will extend their Vue.js skills by building a blog application. They will explore advanced Vue Router concepts for navigation between blog posts and use Pinia for state management. Students will implement features like pagination and user comments.

Week 3 Workshop: Introduction to React

- Scenario: Creating a Recipe Finder App with React
  - Description: In this workshop, students will dive into React by building a Recipe Finder application. They will learn how to create React components, handle user input, and manage component state. Students will build a user-friendly app where they can search for recipes based on ingredients.

Week 4 Workshop: React Router and Comparing Vue.js vs. React

- Scenario: Building a Portfolio Website with React Router
  - Description: In this workshop, students will explore React Router by creating a personal portfolio website. They will set up different routes for portfolio sections like projects, blog, and contact information. Additionally, students will compare and contrast Vue.js and React by implementing similar functionality in both frameworks to understand their strengths and differences.

These workshops provide hands-on experience and practical projects to reinforce the concepts learned during the month. They allow students to apply their knowledge to real-world scenarios and build tangible web applications using Vue.js and React.

Month 6 Workshops

Certainly, here are four workshop scenarios, one for each week of Month 6, which focuses on Back-End Development with Node.js:

Week 1: Introduction to Node.js
Workshop Scenario: Building a Basic HTTP Server

In this workshop, students will apply what they've learned about Node.js fundamentals to build a basic HTTP server from scratch. They will create a server that listens on a specific port, handles incoming HTTP requests, and responds with appropriate HTTP status codes and messages. Students will gain hands-on experience setting up the server, defining routes, and handling different HTTP methods like GET and POST.

Week 2: Building RESTful APIs with Node.js
Workshop Scenario: Creating a RESTful API for a To-Do List

In this workshop, students will dive into RESTful API development using Node.js and Express.js. They will work on building a RESTful API for a simple To-Do List application. Students will define routes for creating, reading, updating, and deleting tasks, ensuring that the API adheres to RESTful principles. They will also implement basic validation and error handling for the API endpoints.

Week 3: Data Storage with MongoDB
Workshop Scenario: Building a CRUD Application with MongoDB and Mongoose

In this workshop, students will focus on data storage using MongoDB and Mongoose, an Object Data Modeling library for MongoDB. They will create a CRUD (Create, Read, Update, Delete) application that interacts with a MongoDB database. Students will define Mongoose schemas, models, and use them to perform database operations like adding, retrieving, updating, and deleting records.

Week 4: Advanced Back-End Development
Workshop Scenario: Implementing User Authentication with Passport.js

In this workshop, students will explore advanced back-end development by implementing user authentication and authorization using Passport.js. They will create a user authentication system, allowing users to register, log in, and access protected routes. Students will work on integrating Passport.js strategies for local authentication and learn how to manage user sessions and secure routes effectively.

Each of these workshops will provide students with practical experience in applying the concepts and tools covered in Month 6. They will work on real-world scenarios and gain confidence in their back-end development skills using Node.js and related technologies.

Month 7 Workshops

Week 1 Workshop: Authentication and Authorization

Scenario 1: Implementing OAuth 2.0 for Social Login

- Workshop Description: In this scenario, students will learn how to integrate OAuth 2.0 authentication for social login into a Node.js application. They will implement authentication with popular platforms like Google or Facebook and understand the OAuth flow.
- Workshop Objectives:
  - Set up OAuth 2.0 authentication.
  - Create routes and controllers for social login.
  - Handle user profile information.
  - Ensure security and authorization.

Scenario 2: Role-Based Access Control (RBAC)

- Workshop Description: This workshop focuses on RBAC implementation. Students will create a system with different user roles and permissions. They will learn to restrict access to specific routes based on user roles and validate user actions.
- Workshop Objectives:
  - Design RBAC roles and permissions.
  - Implement middleware for authorization checks.
  - Secure API endpoints based on user roles.
  - Test role-based access control.

Week 2 Workshop: SQL Databases and PostgreSQL

Scenario 3: Building a RESTful API with PostgreSQL

- Workshop Description: Students will build a RESTful API using Node.js and PostgreSQL as the database. They will create endpoints for CRUD operations and explore SQL queries for data manipulation.
- Workshop Objectives:
  - Set up a PostgreSQL database.
  - Create RESTful API routes.
  - Perform SQL queries for data retrieval and manipulation.
  - Test the API using Postman or a similar tool.

Scenario 4: Data Migration and Seeding with Knex.js

- Workshop Description: In this workshop, students will work with Knex.js, a query builder for Node.js. They will learn to perform database migrations, which are essential for versioning database schema changes, and database seeding to populate initial data.
- Workshop Objectives:
  - Install and configure Knex.js.
  - Create and run database migrations.
  - Seed the database with sample data.
  - Understand the importance of database versioning.

Week 3 Workshop: API Security and Best Practices

Scenario 5: Implementing API Rate Limiting

- Workshop Description: Students will explore API rate limiting as a security measure to prevent abuse and ensure fair usage of APIs. They will implement rate limiting middleware and test its effectiveness.
- Workshop Objectives:
  - Implement rate limiting middleware.
  - Define rate limits for different routes.
  - Test the API for rate limiting enforcement.
  - Handle exceeded rate limits gracefully.

Scenario 6: API Documentation with Swagger

- Workshop Description: In this workshop, students will create interactive API documentation using Swagger. They will learn how to define API endpoints, request and response schemas, and generate user-friendly documentation.
- Workshop Objectives:
  - Set up Swagger for API documentation.
  - Define API endpoints and models.
  - Generate interactive API documentation.
  - Showcase the importance of clear API documentation.

Week 4 Workshop: Deployment and Hosting

Scenario 7: Continuous Integration and Continuous Deployment (CI/CD) with GitHub Actions

- Workshop Description: Students will set up a CI/CD pipeline using GitHub Actions. They will automate the testing and deployment process for their Node.js application whenever changes are pushed to the repository.
- Workshop Objectives:
  - Configure GitHub Actions workflows.
  - Automate testing with each push.
  - Deploy the application to a hosting platform.
  - Monitor the CI/CD pipeline for errors.

Scenario 8: Performance Optimization and Monitoring

- Workshop Description: This workshop focuses on performance optimization and monitoring. Students will learn to use tools like New Relic or similar solutions to monitor their deployed applications and optimize performance.
- Workshop Objectives:
  - Set up performance monitoring tools.
  - Identify performance bottlenecks.
  - Optimize code and database queries.
  - Monitor application health and performance.

These scenarios will provide hands-on experience and practical skills to students during Month 7 of their advanced back-end development program. Each scenario addresses key topics in authentication, authorization, databases, API security, and deployment, ensuring that students are well-prepared for real-world development challenges.

Month 8 Workshops

Week 1: Project Planning and Ideation Workshop

- Scenario 1: "Group Product Presentation"
  - Students form small teams and play the roles of product owner. Each team must create a project proposal for a new web application, complete with a pitch presentation. This workshop simulates the initial project ideation and planning phase, where students learn to define objectives, project scope, and target audiences.

Week 2: Project Development and Prototyping Workshop

- Scenario 2: "Wireframing and Prototyping Challenge"
  - Students are given a real-world problem and are tasked with creating wireframes and prototypes for a web or mobile application. They will use wireframing and prototyping tools to visually design the user interface and demonstrate the project's key features. This workshop emphasizes the importance of planning and designing before coding.

Week 3: Project Implementation and Testing Workshop

- Scenario 3: "Code Review and Bug Bash"
  - Students conduct a code review of a fellow student's project work. They identify code quality issues, security vulnerabilities, or potential improvements. After the code review, a "bug bash" session is held, where students intentionally introduce bugs into a project, and their peers must find and fix them. This workshop enhances coding and debugging skills while fostering collaboration.

Week 4: Project Presentation and Deployment Workshop

- Scenario 4: "Demo Day and Deployment Simulation"
  - Students prepare for a simulated "Demo Day" where they present their completed projects to a panel of instructors or mentors. They showcase the project's features, functionality, and design choices. After the presentations, there is a deployment simulation where students deploy their projects to a staging environment, ensuring everything works smoothly before going live. This workshop hones presentation and deployment skills.

These scenarios provide hands-on experiences that align with the objectives of each week in Month 8, reinforcing project management, development, testing, and deployment skills while simulating real-world challenges and scenarios that students may encounter in their careers.