Software Development Professional Program

Software Development Professional Program

Comprehensive 3-Month Curriculum

Month 1: Fundamentals & Frontend Engineering

Goal: Master the core logic of programming and build responsive, interactive UI.

Week 1: Foundations of Software Engineering

- **Introduction to SDLC:** Phases of the Software Development Life Cycle (Requirements, Design, Implementation, Testing, Deployment).
- **Programming Logic:** Introduction to algorithms, pseudo-code, flowcharts, and basic data structures.
- **Environment Setup:** Configuring IDEs (VS Code), Command Line basics, and local runtime environments.
- **Version Control:** Fundamental Git commands (init, add, commit, push, pull) and collaborating via GitHub.

Week 2: Advanced Web Layouts (HTML5 & CSS3)

- **Semantic HTML:** Structure, accessibility, and SEO-friendly markup.
- **Modern CSS:** Flexbox and CSS Grid for complex layouts.
- **Responsive Design:** Media queries and mobile-first development strategies.
- **CSS Frameworks:** Introduction to Tailwind CSS or Bootstrap for rapid prototyping.

Week 3: JavaScript Programming Essentials

- **JS Fundamentals:** Variables, data types, operators, and control flow (if/else, loops).
- **Functions & Scope:** Declarative vs. arrow functions, closures, and lexical scope.
- **DOM Manipulation:** Selecting elements, event listeners, and dynamic UI updates.
- **ES6+ Features:** Destructuring, spread operators, template literals, and modules.

Week 4: Frontend Framework Mastery (React.js)

- **React Core:** Components, Props, and JSX syntax.
- **State Management:** Handling local state with `useState` and side effects with `useEffect`.
- **Routing:** Implementing Single Page Application (SPA) navigation with React Router.
- **API Integration:** Fetching data from external services using Axios or Fetch API.

Month 2: Backend Development & Databases

Goal: Build secure, scalable server-side logic and manage persistent data storage.

Week 5: Server-Side Programming (Node.js)

- **Node.js Runtime:** Event loop, non-blocking I/O, and the module system.
- **NPM Ecosystem:** Managing dependencies and understanding `package.json`.
- **Asynchronous JS:** Advanced Promises, Async/Await, and error handling.
- **CLI Tools:** Building custom command-line utilities for automation.

Week 6: Web APIs & Frameworks (Express.js)

- **Server Setup:** Creating an Express server and defining middleware.
- **RESTful Routing:** Designing standard API endpoints (GET, POST, PUT, DELETE).
- **Request/Response:** Handling headers, query parameters, and JSON bodies.
- **Validation:** Implementing data validation using libraries like Joi or Zod.

Week 7: Database Design & Management (SQL & NoSQL)

- **Relational Databases (PostgreSQL/MySQL):** Schema design, normalization, and complex SQL queries.
- **Non-Relational Databases (MongoDB):** Collections, documents, and flexible schemas.
- **ORM/ODM Integration:** Using Sequelize (SQL) or Mongoose (NoSQL) for database interaction.
- **Modeling:** Designing relationships (One-to-One, One-to-Many, Many-to-Many).

Week 8: Security, Authentication & Deployment

- **Authentication:** User registration and login using Password Hashing (Bcrypt).
- **Authorization:** Securing routes with JSON Web Tokens (JWT) and role-based access.
- **Application Security:** Protecting against XSS, CSRF, and SQL Injection.
- **Deployment Basics:** Hosting backend services on platforms like Render or AWS EC2.

Month 3: Advanced Topics & Portfolio Project

Goal: Integrate DevSecOps practices and complete a professional-grade Project.

Week 9: Testing & Quality Assurance

- **Unit Testing:** Writing test cases for isolated functions using Jest or Mocha.
- **Integration Testing:** Testing API endpoints and database interactions.
- **Debugging:** Advanced techniques using browser DevTools and IDE debuggers.
- **Test-Driven Development (TDD):** Implementing features by writing tests first.

Week 10: DevOps & CI/CD Pipelines

- **Containerization:** Introduction to Docker and creating containerized environments.
- **Automation:** Setting up CI/CD pipelines with GitHub Actions for automated testing and deployment.
- **Cloud Infrastructure:** Overview of AWS services (S3 for storage, RDS for databases).
- **Monitoring:** Basic logging and application performance monitoring.

Week 11: Advanced Architectural Patterns

- **Design Patterns:** Understanding MVC (Model-View-Controller) and Singleton patterns.
- **Microservices:** Introduction to distributed systems and inter-process communication.
- **Performance Optimization:** Caching strategies (Redis) and frontend performance tuning.
- **Documentation:** Generating API documentation using Swagger.

Week 12: Capstone Project & Career Readiness

- **Project Execution:** Building a full-stack, feature-rich application (e.g., E-commerce or Social Media Dashboard).
- **Code Review:** Participating in peer reviews and refactoring for production standards.
- **Portfolio Building:** Hosting the final project and optimizing the GitHub profile.
- **Interview Prep:** Technical interview simulation, resume workshop, and soft skills training.