**Technical Skills**

1. **JavaScript/ES6+**:
   * Proficiency in modern JavaScript, including ES6+ features like async/await, destructuring, and modules.
2. **Node.js**:
   * Building server-side logic.
   * Handling asynchronous operations.
   * Working with modules and npm packages.
3. **Express.js**:
   * Setting up a web server.
   * Creating RESTful APIs.
   * Middleware for handling requests and responses.
4. **MongoDB**:
   * Designing and managing a NoSQL database.
   * CRUD operations (Create, Read, Update, Delete).
   * Mongoose for schema definitions and validations.
5. **React.js**:
   * Building interactive UIs.
   * Component-based architecture.
   * State management with hooks (useState, useEffect).
   * React Router for navigation.
6. **HTML/CSS**:
   * Designing responsive layouts.
   * Styling components using CSS or CSS-in-JS libraries like styled-components.
7. **Authentication & Authorization**:
   * Implementing user authentication (JWT, OAuth).
   * Role-based access control.
8. **Deployment**:
   * Deploying applications on platforms like Heroku, AWS, or DigitalOcean.
   * CI/CD pipelines for continuous integration and deployment.

**Components of a Food Delivery Portal**

1. **User Authentication**:
   * Sign up, login, and logout functionality.
   * Password recovery.
   * OAuth integration (e.g., Google, Facebook login).
2. **Restaurant Management**:
   * Admin panel for restaurant owners to manage their menus, prices, and availability.
   * CRUD operations for restaurants and menu items.
3. **Menu and Ordering System**:
   * Displaying menus with categories and items.
   * Adding items to the cart.
   * Checkout process and order placement.
4. **User Profile and Order History**:
   * User profile management.
   * Viewing past orders and order statuses.
5. **Search and Filtering**:
   * Searching for restaurants by name, cuisine, or location.
   * Filtering menus by categories.
6. **Real-Time Order Tracking**:
   * Tracking order status from preparation to delivery.
   * Notifications for order updates.
7. **Payment Integration**:
   * Integrating payment gateways (Stripe, PayPal).
   * Secure payment processing.
8. **Ratings and Reviews**:
   * Allowing users to rate and review restaurants and dishes.
   * Displaying ratings and reviews.
9. **Admin Dashboard**:
   * Managing users, restaurants, orders, and payments.
   * Analytics and reports.

**Example Project Structure**

**Backend (Node.js, Express, MongoDB):**

1. **Server Setup**:
   * Initialize a Node.js project.
   * Install necessary packages: express, mongoose, jsonwebtoken, bcryptjs, etc.
2. **Database Configuration**:
   * Connect to MongoDB using Mongoose.
   * Define schemas for users, restaurants, menu items, and orders.
3. **API Routes**:
   * Define routes for user authentication, restaurant management, menu items, orders, etc.
   * Implement route handlers and middleware for validation and error handling.
4. **Authentication**:
   * Implement JWT-based authentication.
   * Protect routes using authentication middleware.

**Frontend (React.js):**

1. **Project Setup**:
   * Create a React project using Create React App.
   * Install necessary packages: react-router-dom, axios, redux (optional), etc.
2. **Components**:
   * Create reusable components for header, footer, forms, modals, etc.
   * Build pages for home, login, signup, restaurant listing, menu, cart, and order tracking.
3. **State Management**:
   * Use React hooks for state management.
   * Optionally, use Redux for more complex state management needs.
4. **API Integration**:
   * Use Axios to make API requests to the backend.
   * Handle loading states and error handling.
5. **Styling**:
   * Use CSS modules, styled-components, or another styling solution.
   * Ensure the application is responsive and mobile-friendly.

**Learning Resources**

1. **Official Documentation**:
   * Node.js
   * Express.js
   * [MongoDB](https://docs.mongodb.com/)
   * React.js
2. **Online Courses**:
   * Udemy, Coursera, and freeCodeCamp offer courses on MERN stack development.
3. **Community and Forums**:
   * Stack Overflow, Reddit, and GitHub for community support and code examples.

Building a food delivery portal is a comprehensive project that covers a wide range of skills and technologies, providing a great opportunity to learn and apply full-st