PREDICTED PAPER 2025

Web Computing (Paper Code: 48892)

T.E. Computer Engineering & AI-DS, Semester V

Duration: 3 Hours | Total Marks: 80

By: Nitin Gupta

Instructions to Candidates:

1. Question No. 1 is compulsory
2. Attempt any three questions from remaining five questions
3. Assume suitable data if necessary and justify the assumptions
4. Figures to the right indicate full marks

Q1. Answer the following questions: [20 Marks]

A. [5 Marks]

What is DNS? Explain the working of DNS with its components. Describe the step-by-step process of DNS resolution from client to authoritative server.

B. [5 Marks]

Write a JavaScript function that validates a password field with the following requirements:

- Minimum 8 characters
- At least one uppercase letter
- At least one lowercase letter
- At least one digit
- At least one special character (@, #, $, %, &)

Display appropriate error messages for validation failures.

C. [5 Marks]

Explain the concepts of Arrow Functions in JavaScript ES6 with examples. How do arrow functions differ from regular functions in terms of 'this' binding?

D. [5 Marks]

What is a callback in Node.js? Explain with a suitable example. What is callback hell and how can it be avoided?

Q2. [20 Marks]

A. [10 Marks]

Explain the concept of React Hooks. What are the rules of using Hooks? Provide detailed examples of:

1. useState hook - Create a toggle button
2. useEffect hook - Fetch data from API on component mount
3. Custom hooks - Create a useLocalStorage hook

Explain when to use each hook and their lifecycle behavior.

B. [10 Marks]

What is Express.js? Explain the features and advantages of using Express.js. Write a complete Express application that:

1. Creates a server on port 3000
2. Implements middleware for request logging
3. Has routes: GET /, POST /api/users, GET /api/users/:id
4. Uses body-parser for JSON data
5. Implements error handling middleware
6. Returns appropriate status codes

Q3. [20 Marks]

A. [10 Marks]

Explain the Document Object Model (DOM) and its levels (Level 0, 1, 2, 3). Write a comprehensive JavaScript program that:

1. Creates a form dynamically with name, email, and phone fields
2. Validates the form inputs

3. Displays error messages next to invalid fields
4. On successful validation, displays the data in a table
5. Includes a button to reset the form

B. [10 Marks]

Explain the concept of state and props in React. How do they differ, and how are they used in components? Create a React application with:

1. Parent component "ProductList" that maintains array of products in state
2. Child component "Product" that receives product details via props
3. Functionality to add new products
4. Display product count
5. Demonstrate props drilling and state lifting

Q4. [20 Marks]

A. [10 Marks]

Compare and contrast MVC, FLUX, and Redux architectures. For each architecture, explain:

1. Core concepts and components
2. Data flow pattern (with diagram)
3. Advantages and disadvantages
4. Use cases in modern web applications
5. Code example showing state management

Which architecture would you choose for a large-scale e-commerce application and why?

B. [10 Marks]

Write a Node.js program using Express to create a RESTful API server that:

1. Handles GET and POST requests for managing student records
2. GET /students - Returns all students
3. GET /students/:id - Returns specific student
4. POST /students - Adds new student
5. PUT /students/:id - Updates student details
6. DELETE /students/:id - Deletes student
7. Implements proper routing structure
8. Includes input validation
9. Implements error handling with appropriate HTTP status codes

10. Uses in-memory array to store data

Provide complete code with explanations for each route.

Q5. [20 Marks]

A. [10 Marks]

Describe the architecture of Node.js with a neat diagram. Explain:

1. Event-driven programming model
2. Event loop and its phases in detail
3. Non-blocking I/O operations
4. Libuv library role
5. Thread pool
6. How Node.js handles 10,000 concurrent requests

Write an example demonstrating asynchronous file operations with proper error handling using:

- Callbacks
- Promises
- Async/Await

B. [10 Marks]

Explain how React Router can be used to create a single-page application (SPA). Write a complete React application with React Router that includes:

1. Navigation bar with links
2. Routes for: Home, About, Products, Product Detail, Contact, 404 Not Found
3. Nested routing for products
4. Route parameters to display individual product
5. Programmatic navigation
6. Protected routes (authentication check)
7. Route guards

Provide complete code with explanation of BrowserRouter, Route, Link, and useNavigate.

Q6. [20 Marks]

A. [10 Marks]

Differentiate between ES5 and ES6. Explain the following ES6 features with detailed examples and use cases:

1. Arrow Functions (including lexical 'this')
2. Classes and Inheritance
3. Template Literals
4. Destructuring (objects and arrays)
5. Spread and Rest operators
6. Promises
7. let and const
8. Default parameters
9. Modules (import/export)

How do these features improve JavaScript code quality and maintainability?

B. [10 Marks]

You are tasked to design a comprehensive form validation system in JavaScript for an online job application portal. Create a form with the following fields and validations:

Fields:

- Full Name (required, only alphabets and spaces, 3-50 characters)
- Username (required, alphanumeric, 5-15 characters, check availability)
- Email (required, valid format, domain must be from allowed list: gmail.com, yahoo.com, outlook.com)
- Phone (required, exactly 10 digits, must start with 6/7/8/9)
- Date of Birth (required, age must be between 21-35 years)
- Password (required, minimum 8 characters, must contain uppercase, lowercase, digit, special character)
- Confirm Password (must match password)
- Resume Upload (PDF/DOC only, max 2MB)
- Experience (required, select dropdown: 0-1, 1-3, 3-5, 5+ years)
- Skills (checkboxes, at least 2 must be selected)
- Terms and Conditions (checkbox, must be checked)

Requirements:

1. Real-time validation as user types
2. Display specific error messages for each field
3. Disable submit button until all validations pass

4. Show password strength meter
5. Implement proper error handling
6. Ensure data security (XSS prevention)
7. Display success message on valid submission
8. Implement form reset functionality

Provide complete HTML and JavaScript code with detailed comments explaining the validation logic.

END OF PAPER

Best of luck for your 2025 exams!

Notes for Students:

High Priority Topics for 2025:

1. React Hooks (useState, useEffect, custom hooks)
2. Express.js with REST API development
3. State Management (Redux, Context API)
4. Form Validation (Complex scenarios)
5. Node.js Event Loop and Asynchronous Programming
6. Modern ES6+ Features
7. React Router and SPA development
8. MVC vs FLUX vs Redux comparison

Exam Tips:

- Practice writing complete code, not just theory
- Draw diagrams for architecture questions
- Include error handling in all programming questions
- Understand the "why" behind each concept
- Prepare code examples for all major topics
- Focus on practical implementation

Compiled by: Nitin Gupta

**Good luck with your preparation! **