

Assessment

Module 8: JavaScript Essentials and Advanced

Case Overview

Scenario:

You are an **intermediate JavaScript developer** at **Swan Solution Pvt. Ltd.** Your team has been assigned create a **modular, smart, and interactive Hospital Appointment Web App** for a Hospital Management. The form should be responsive, perform real-time validation, and follow clean coding practices.

Core functionality	
1. Appointment Form	<p>Create a form with the following fields:</p> <ul style="list-style-type: none">• Applicant Name (Text)• Email Address (Email)• Phone Number (Number)• Date of Appointment (Date)• Time of Appointment (Time)• Medical disease Comment(Textarea) <ul style="list-style-type: none">• Use Flexbox/Grid layout with media queries for responsiveness on mobile and tablets.

2. Form Handling with Advanced JavaScript	<p>Class: AppointmentHandler</p> <table border="1"> <thead> <tr> <th>Method</th><th>Functionality</th></tr> </thead> <tbody> <tr> <td>validateForm()</td><td> Validate fields using regex and rules: <ul style="list-style-type: none"> • Name ≥ 3 characters • Valid Email • Phone = 10 digits • Valid future Date & Time • Cover Letter ≥ 50 characters </td></tr> <tr> <td>saveToLocalStorage()</td><td>Store validated applications as an array in localStorage</td></tr> <tr> <td>clearForm()</td><td>Reset all form fields after successful submission</td></tr> <tr> <td>showMessage(type, message)</td><td>Display styled success/error alerts</td></tr> <tr> <td>addEventListeners()</td><td>Use event delegation and attach real-time validation inputs (oninput or onblur)</td></tr> </tbody> </table>	Method	Functionality	validateForm()	Validate fields using regex and rules: <ul style="list-style-type: none"> • Name ≥ 3 characters • Valid Email • Phone = 10 digits • Valid future Date & Time • Cover Letter ≥ 50 characters 	saveToLocalStorage()	Store validated applications as an array in localStorage	clearForm()	Reset all form fields after successful submission	showMessage(type, message)	Display styled success/error alerts	addEventListeners()	Use event delegation and attach real-time validation inputs (oninput or onblur)
Method	Functionality												
validateForm()	Validate fields using regex and rules: <ul style="list-style-type: none"> • Name ≥ 3 characters • Valid Email • Phone = 10 digits • Valid future Date & Time • Cover Letter ≥ 50 characters 												
saveToLocalStorage()	Store validated applications as an array in localStorage												
clearForm()	Reset all form fields after successful submission												
showMessage(type, message)	Display styled success/error alerts												
addEventListeners()	Use event delegation and attach real-time validation inputs (oninput or onblur)												
3. View Appointment Page (view-appointment.html)	<p>Class: Appointment Viewer</p> <ul style="list-style-type: none"> • Fetch and display data from localStorage in a table. • Add a search input to filter appointment by Name or Appointment Date. • Show “No applications found” message when data is empty or filtered results don’t match. 												

Optional Bonus Tasks

- Add Delete button beside each application row to remove it from localStorage.
- Modularize code using a separate utility module (storageUtils.js) for getItem() / setItem().
- Structure JavaScript using ES Modules and optionally use a bundler like Vite/Webpack.

Validation Rules

Field	Rule
Applicant Name	≥ 3 characters
Email Address	Valid email format
Phone Number	Exactly 10 digits
Appointment Date	Valid future Date & Time
Cover Letter	Minimum 50 characters
All Fields	Required (no empty submission allowed)

Key Competencies Tested

- ES6 Classes and Modular Code
- DOM Manipulation and Clean UI
- Event Delegation and Real-Time Input Validation
- localStorage CRUD operations
- Responsive Design with CSS Grid/Flexbox
- Search and Filter Functionality
- Reusable Component and Utility Design

Reflective Thinking Questions

- How would this app differ if data were stored on a backend (e.g., Firebase, Node API)?
- What improvements can be made to handle validation errors more accessibly (e.g., ARIA)?
- How secure is storing job applications in localStorage? What are safer alternatives?
- How can this form be scaled into a Progressive Web App (PWA) for offline job fairs?