

Assessment

Module 8: JavaScript Essentials and Advanced

Case Overview

Scenario:

You are an intermediate JavaScript developer at **Swan Solution Pvt. Ltd.** The company wants a **Hotel Management System** to eliminate paper-based guest registration processes Web App. The app must be responsive, interactive, and follow good programming practices.

Core Functionality	
1. Responsive Form Creation	<p>Create a form with the following fields:</p> <p>Full Name (Text) Phone Number (Number) Email Address (Email) Address (Textarea) Aadhar Card Number (Number) Check in date & Check Out date (Date) No of Adult Purpose of visit (Textarea)</p> <p>Mobile-friendly layout using Bootstrap Grid</p>
2. Validation Rules with Advanced JavaScript	<p>Class: CustomerFormHandler validateForm() – Validate:</p> <p>Name ≥ 3 characters</p> <p>Phone exactly 10 digits Valid email format (Regex) Address details not empty Aadhar Card Number : Exactly 12 digits Check in date & Check Out date : Future date validation No of Adult : Number Validation Purpose of visit : Required Field</p> <p>saveToLocalStorage() – Save data as array in localStorage</p> <p>clearForm() – Reset form after submission</p>

	<p>showMessage() – Reusable UI message box for success/error</p> <p>Additional Requirements: Use event delegation for handling form events</p> <p>Implement real-time validation (on input or blur)</p>
3. View Submissions Page (view.html)	<p>Class: SubmissionViewer Read from localStorage</p> <p>Dynamically generate table of submissions</p> <p>Search bar to filter by Name or Date</p> <p>Show “No data found” message if results empty</p>
<p>Optional Bonus Tasks</p> <p>Add Delete button to remove individual records</p> <p>Use utility module for setItem() and getItem() (modularized code)</p> <p>Implement import/export (<script type="module"> or bundler like Vite/Webpack)</p>	

Validation Rules

- All fields required
- Name ≥ 3 characters
- Phone exactly 10 digits
- Valid email format
- Complaint ≥ 10 characters

Key Competencies Tested

- ES6 Classes & Modules
 - DOM Manipulation & Event Handling
 - Real-Time & Manual Form Validation
- LocalStorage CRUD Operations
- Dynamic HTML Rendering & Filtering

- Clean, Modular, and Reusable Code

Reflective Thinking Questions

- How could this app be enhanced by integrating with an **API** (e.g., saving data to a backend)?
- How can error handling be improved for a better **user experience**?
- What security concerns arise when storing customer data in `localStorage`?
- Could the app be scaled into a **PWA** (Progressive Web App) for offline use?