

Internship Task: Hotel Booking Form

Task Name:

Hotel Booking Form

Duration:

10 Days

Task Description:

The task is to design and develop a hotel booking form that includes fields for check-in/check-out dates, room type, guest count, and special requests. The form should include proper validation for date selection and ensure that the check-out date is later than the check-in date. This task will help students practice form validation, responsive design, and dynamic form behavior.

Objectives:

- Create a functional hotel booking form with multiple fields.
 - Implement date validation to ensure the check-out date is after the check-in date.
 - Use proper form validation for required fields.
 - Ensure the form is responsive and works across various screen sizes.
 - Use GitHub for version control to track progress.
-

Features to Implement:

1. Form Fields:

- **Check-in Date:** A date input field to select the check-in date.
- **Check-out Date:** A date input field for selecting the check-out date. Ensure the check-out date cannot be before the check-in date.
- **Room Type:** Dropdown or select field with options such as Single, Double, Suite, etc.
- **Guest Count:** Number input for the number of guests (adults and children).
- **Special Requests:** A textarea for any special requests or notes (optional).
- **Submit Button:** A button to submit the form after validation.

2. Date Validation:

- Validate that the check-out date is after the check-in date.
- Optionally, restrict dates (e.g., no bookings for past dates, certain date ranges, etc.).

3. Responsive Design:

- Ensure the form is responsive and works well on mobile, tablet, and desktop devices.
- Use Flexbox or CSS Grid to layout the form fields in an accessible and attractive manner.

4. Optional Enhancements:

- Display dynamic messages or error alerts if validation fails.
- Add tooltips or information icons to guide users on how to fill out certain fields (e.g., room types or special requests).
- Use JavaScript to dynamically update available dates or options (e.g., limiting check-out dates based on the check-in date).

Technology Stack:

- HTML5
- CSS3 (using frameworks like Bootstrap or Tailwind CSS for responsiveness)
- JavaScript (for form validation and interactivity)
- Optional Enhancements:
 - Use libraries like jQuery UI for date pickers or custom date validation logic.

Interactive Components:

- Date pickers for check-in/check-out dates.
- Form validation with error handling and dynamic feedback messages.
- Conditional field behavior, e.g., adjusting available options based on previous selections.

Version Control:

- Students must create a repository on GitHub.
- Commit changes regularly (with meaningful commit messages).
- Push the final project code to the GitHub repository.

Presentation:

- Deploy the project using platforms like GitHub Pages, Netlify, or Vercel.
- Prepare a brief presentation/demo (5-10 minutes) explaining:
 - Features of the hotel booking form.
 - How validation works and how the form interacts with the user.
 - Technologies used and challenges faced during development.

Deliverables:

1. A fully functional hotel booking form with all required fields and validations.
2. Source code pushed to the GitHub repository.
3. Link to the deployed webpage.
4. Documentation (README file) containing:
 - Project Overview.
 - Instructions to run the project.
 - Technologies used.

Task Evaluation Criteria:

1. Design Quality: UI/UX, responsiveness, and overall appearance. (25%)
2. Functionality: Date validation, form submission, and interactivity. (25%)
3. Code Quality: Clean, readable, and well-structured code. (20%)
4. GitHub Usage: Proper commits and repository structure. (15%)
5. Presentation/Demo: Clear explanation of work and project deployment. (15%)

Tools and Resources:

- Code Editors: VS Code, Sublime Text, etc.
- Version Control: Git and GitHub
- CSS Frameworks: Bootstrap or Tailwind CSS
- JavaScript Libraries (Optional): jQuery UI for date pickers, etc.

- **Deployment Platforms: GitHub Pages, Netlify, Vercel**
-

Deadline:

The task must be submitted within 10 days from the date of assignment.