Creating a WCAG (Web Content Accessibility Guidelines) compliance test plan ensures that your website or application is accessible to users with disabilities. Using a tool like **BrowserStack** makes it easier to test accessibility across multiple devices and browsers. Below is a **detailed WCAG compliance test plan**, including a sample format, guidelines, and how to use BrowserStack effectively.

### **WCAG Compliance Test Plan**

#### **1. Objective**

To ensure that the website or application adheres to WCAG 2.1 AA standards, providing accessible content to all users, including those with disabilities.

#### **2. Scope**

* Target Levels: WCAG 2.1 (Levels A and AA)
* Platforms: Desktop and Mobile
* Browsers: Chrome, Firefox, Edge, Safari
* Tools: BrowserStack, screen readers, and browser extensions
* Test Pages: Home page, Login page, Dashboard, and any other critical user workflows.

### **3. Test Methodology**

#### **Manual Testing**

1. Verify content against WCAG principles:
   * Perceivable
   * Operable
   * Understandable
   * Robust
2. Use tools like:
   * BrowserStack's Accessibility testing feature.
   * Browser developer tools (e.g., Lighthouse in Chrome).
   * Screen readers (e.g., NVDA, VoiceOver).

#### **Automated Testing**

1. Use BrowserStack’s automated tools for WCAG checks.
2. Integrate with tools like **axe-core**, **Lighthouse**, or BrowserStack's Accessibility features.

#### **BrowserStack-Specific Setup**

* **Live Testing:** For manual accessibility tests across devices.
* **Automate Testing:** To verify WCAG compliance using automated scripts.
* **Integrations:** Use BrowserStack’s integration with Axe for automated accessibility scans.

### **4. Roles and Responsibilities**

| **Role** | **Responsibility** |
| --- | --- |
| Accessibility Lead | Define accessibility goals and test scope. |
| QA Tester | Perform manual and automated WCAG tests. |
| Developer | Fix WCAG violations found during testing. |
| Product Owner | Ensure accessibility is part of release goals. |

### **5. Test Cases**

#### **Sample Test Case Format**

| **Test ID** | **Requirement** | **Test Steps** | **Expected Result** | **Pass/Fail** | **Remarks** |
| --- | --- | --- | --- | --- | --- |
| WCAG-1.1.1 | Provide text alternatives for images (WCAG 1.1.1) | 1. Open the web page on BrowserStack.  2. Inspect all <img> tags.  3. Check if each image has an appropriate alt attribute. | All images should have descriptive alt text. | Pass |  |
| WCAG-2.4.3 | Provide focus order (WCAG 2.4.3) | 1. Open the page.  2. Navigate through elements using the Tab key.  3. Ensure the focus order is logical and intuitive. | Focus should follow a logical order. | Fail | Focus skips a critical button. |
| WCAG-1.4.3 | Ensure color contrast (WCAG 1.4.3) | 1. Use BrowserStack.  2. Use the contrast checker in BrowserStack's Accessibility tool.  3. Verify the contrast ratio is >= 4.5:1 for text. | Text contrast ratio meets WCAG AA standards. | Pass |  |
| WCAG-2.1.1 | Keyboard navigation (WCAG 2.1.1) | 1. Disconnect the mouse.  2. Navigate through the site using only the keyboard.  3. Test all interactive elements (buttons, forms, menus). | All interactive elements should be accessible via keyboard. | Pass |  |
| WCAG-4.1.2 | Screen reader compatibility (WCAG 4.1.2) | 1. Use NVDA or VoiceOver.  2. Navigate the page.  3. Check if screen reader announces labels and roles for all elements. | Screen reader should announce appropriate roles and labels. | Fail | Missing ARIA labels on form elements. |

### **6. Tools**

#### **BrowserStack for Accessibility Testing**

* **Live Testing:**Test manually on real devices for color contrast, focus order, and keyboard navigation.
* **Automate Testing:**Integrate tools like Axe or Lighthouse with BrowserStack for automated WCAG compliance scans.
* **Screen Readers:**Use BrowserStack's real devices to test with NVDA, JAWS, or VoiceOver.

#### **Other Accessibility Tools**

* **axe DevTools:** For automated WCAG checks.
* **Color Contrast Analyzer:** For testing color contrast.
* **Lighthouse:** Built into Chrome DevTools for auditing accessibility.
* **Screen Readers:** NVDA, VoiceOver, or JAWS.

### **7. Reporting**

#### **Report Format**

| **Page/Component** | **WCAG Guideline** | **Severity** | **Issue Description** | **Screenshot** | **Recommended Fix** |
| --- | --- | --- | --- | --- | --- |
| Login Page | WCAG 1.4.3 | High | Text contrast ratio is below 4.5:1 | <attached screenshot> | Use darker font color for better contrast. |
| Navigation Menu | WCAG 2.4.3 | Medium | Focus order is not intuitive | <attached screenshot> | Adjust tabindex to create logical flow. |
| Registration Form | WCAG 4.1.2 | High | Missing ARIA labels on input fields | <attached screenshot> | Add ARIA labels to all input fields. |

#### **Severity Levels**

* **High**: Blocks users with disabilities from completing essential actions.
* **Medium**: Causes inconvenience but does not block essential functionality.
* **Low**: Minor visual or operational issue.

### **8. Test Execution Process**

1. **Environment Setup:**
   * Use BrowserStack to test on a matrix of browsers and devices.
   * Verify configurations like screen reader availability and color contrast tools.
2. **Perform Tests:**
   * Conduct manual tests for keyboard navigation, focus order, and interactive elements.
   * Use BrowserStack's automated tools or integrations to identify WCAG violations.
3. **Log Issues:**
   * Document each issue with screenshots and recommendations.
   * Categorize issues by severity and guideline reference.
4. **Retest After Fixes:**
   * Use BrowserStack’s Live Testing to validate fixes.

### **9. Summary and Deliverables**

At the end of testing:

* **Accessibility Test Report:** Summarizing issues, fixes, and compliance status.
* **WCAG Compliance Metrics:** Percentage of compliance achieved.
* **Screenshots and Evidence:** To support all findings.

### **Sample Deliverable**

* **All issues logged in a test management tool (e.g., Jira).**
* **WCAG Compliance Report:**
  + Total pages tested: 5
  + Total issues: 12 (5 High, 4 Medium, 3 Low)
  + Compliance Level Achieved: 85%

### **Using BrowserStack for WCAG Testing**

BrowserStack provides a reliable platform to test on real devices and browsers. Here's how to leverage it:

1. **Live Testing:**
   * Navigate to BrowserStack Live.
   * Select a device and browser.
   * Use browser extensions like axe DevTools or Lighthouse for accessibility checks.
2. **Automated Testing:**
   * Set up BrowserStack Automate to run scripts using accessibility testing tools like axe-core.
   * Integrate these tools in your CI/CD pipeline for continuous accessibility checks.
3. **Device-Specific Testing:**
   * Use real iOS and Android devices on BrowserStack to test screen reader compatibility (VoiceOver, TalkBack).

This plan ensures a structured, effective WCAG compliance process with actionable deliverables and comprehensive testing across devices using BrowserStack.