Progressive Web Application automation framework handling

1. Verify PWA Installation

PWAs can be installed as standalone apps. You can test the installation process by simulating user interactions.

2. Testing Offline Capabilities

PWAs should work offline when cached correctly. You can test this by disabling the network in your automation framework.

3. Validate Service Worker Registration

Service workers are the backbone of a PWA. Ensure they are registered and functioning as expected.

4. Verify Web App Manifest

Ensure the web app manifest file is loaded and has the correct properties (e.g., icons, start URL, display mode).

5. Test Push Notifications

For testing push notifications, you'll need to:

1. Subscribe the browser to push notifications.
2. Simulate a push event and verify the notification.

6. Test Add to Homescreen (A2HS)

Simulate and validate the "Add to Homescreen" prompt.

7. Testing Responsiveness

PWAs are expected to be responsive. Use viewport emulation to test various screen sizes.

8. CI/CD Integration

Use headless mode for running PWA tests in CI environments.
Integrate with tools like GitHub Actions, Jenkins, or CircleCI.

9. Reporting and Metrics

Capture important metrics, such as:
- Time to first byte (TTFB)
- Service worker activation time
- Offline response time

Use tools like Lighthouse for detailed audits.

10. Handle Multi-Browser Testing

Ensure the PWA works consistently across browsers:

- Use Playwright for cross-browser testing.
- Test in Chrome, Edge, and Firefox for optimal coverage.