1. Due to usage of page object model, we cannot reuse the webelement. For each and every page, we have to define the webelement. E.g the header element on two pages.
2. Script performance issue: Wrong coding logic. E.g verifying of web element’s first col element. Here, team was fetching each and every row in loop, so it’s iterating each and every row. E.g if there are 500 rows.

Solution: Used descendent xpath to just fetch the col values rather navigating and looping through all rows.

1. Wrong wait techniques:
2. Multi browser support: Because of xath not working in IE.
3. Due to testing of remote database.
4. Faced lots of issue while setting up the maven build tool to execute test ng xml.
5. Due to frequent build due to agile methodology.
6. Frequent requirement change due to agile.
7. Setting up maven.
8. Due to remote database.
9. It is a good practice to have only independent test scripts when working on test automation.

# One should have the ability to ATOMIC TESTS- Avoid testing too many things in a single script.

* + Each test should test a single tiny bit of functionality.
  + Each test should have up to 20 steps.
  + Each test should run in under 10 seconds.

1. USE ONLY EXPLICIT WAITS-

* One of the best ways of making a script faster is by using only explicit waits.
* The script performance is better with explicit waits as HTML elements are accessed as soon as they become available.

1. Due to current implementation of Grid throwing lots of challenges as earlier framework was meant to execute tests in sequence. Now due to parallel execution, following challenges are being faced to upgrade the framework.

* Parallel updating or writing of test results.
* Taking failure screen shots

1. PRE-POPULATE COOKIES: To enhance the performance:

This way, page 2 will load directly with 25 results and in french.

Page 3 will load directly in chinese with 3 results.

Code Sample:

public void addCookie(String name, String value) {

Cookie pageSize = new Cookie.Builder(name, value)

.domain(“vpl.bibliocommons.com”)

.expiresOn(new Date(2016, 12, 31))

.isSecure(true)

.path(“/”)

.build();

driver.manage().addCookie(pageSize);

}

@Test

public void testResult() throws InterruptedException {

driver.get(“https://vpl.bibliocommons.com/search?q=java&t=keyword&#8221;);

Thread.sleep(5000);

addCookie(“page\_size”, “25”);

addCookie(“language”, “fr-CA”);

driver.get(“https://vpl.bibliocommons.com/search?page=1&q=java&t=keyword&#8221;);

Thread.sleep(5000);

addCookie(“page\_size”, “3”);

addCookie(“language”, “zh-CN”);

driver.get(“https://vpl.bibliocommons.com/search?page=2&q=java&t=keyword&#8221;);

Thread.sleep(5000);

}

1. Non availability of dedicated test environment at the start of automation project: E.g. OMAC application.
2. Sometimes, our application did not have good elements id’s or attributes.

Whatever code was working last night/or week would fail on the following day due to change or missing of id’s and attributes.



