**Складові системи запуску тестів:**

1. Сервер безперервної інтеграції – **Jenkins.** (Для цього необхідний ще репозиторій **git**)
2. Складальник – **Maven.**
3. Фрейворк запуска тестів – **TestNG, jUnit…**
4. **Тести** (ЯП: java).
5. Драйвер системи, що тестується – **WebDriver.**
6. Браузер – **Firefox.**
7. Система, що тестується.

**Налаштування середовища java на локальному комп’ютері.**

Завантажити дистрибутив **JDK** та інсталювати на локальній машині. Бажано інсталювати в корінь диску С. Назва папки «java» без пробілів(для змінного середовища). Змінне середовище налаштовується для того, щоб була можливість працювати з додатком безпосередньо з консолі не вказуючи повний шлях до виконуваного файлу додатка.

Налаштування змінного середовища КП для java:

1. Перейти в *Система\Додаткові параметри системи\Змінне середовище\Системні змінні.*
2. Створити: «Ім’я змінної»=«JAVA\_HOME»; «Значення змінної»= «**C:\Java\jdk1.8.0\_92**»(шлях до встановленої java).
3. Редагувати системну змінну “Path”. В поле «Значення змінної» через «;» без пробіла додати запис «**%JAVA\_HOME%\bin**»

Перевірити інсталювання та налаштування **java**,а також визначення версії: в консоль ввести команду «**java -version»**.

**Налаштування Maven.**

Завантажити дистрибутив **apache-maven** та інсталювати на локальній машині. Бажано інсталювати в С:\java. Назва папки «java» без пробілів(для змінного середовища).

Налаштування змінного середовища КП для Maven:

1. Перейти в *Система\Додаткові параметри системи\Змінне середовище\Системні змінні.*
2. Створити: «Ім’я змінної»=«M3\_HOME»; «Значення змінної»= **«C:\Java\apache-maven-3.3.9»(**шлях до встановленого Maven).
3. Редагувати системну змінну “Path”. В поле «Значення змінної» через «;» без пробіла додати запис «**%M3\_HOME%\bin**».

Перевірити інсталювання та налаштування **maven**: в консоль ввести команду «**mvn -version»**.

**Створення проекту із архітипу.**

1. Необхідно перейти на git та клонувати з репозиторія Баранцева архітип:

git clone git://github.com/barancev/Selenium2-Java-QuickStart-Archetype.git

cd Selenium2-Java-QuickStart-Archetype (перейти в діректорю)

mvn install (виконати команду – інсталювати архітип)

1. Генеруємо проект на основі архітипу:

mvn archetype:generate -DarchetypeGroupId=com.sebarmeli -DarchetypeArtifactId=selenium2-java-quickstart-archetype -DarchetypeVersion=0.3 -DgroupId=<mygroupId> -DartifactId=<myartifactId>,

де ми вказуємо DgroupId=ua.bms.test(назва групи розробки, унікальний ідентифікатор групи проектів в рамках компанії) та DartifactId=selenium\_java\_training(назва проекту, тобто унікальний ідентифікатор проекту в рамка групах проектів).

1. В результаті створюється структура підкаталогів.

1-рівень: папка «selenium\_java\_training»

2-рівень: папка “src” та файл “pom.xml”(опис проекту, де описані різні залежності від фреймворка(TestNG, jUnit), від браузера, селеніума…)

3-рівень папки src: папки “test”(будуть знаходитися самі тести, плагіни, які будуть запускати ці тести) та “main”(розміщують допоміжний код, також тут знаходиться фабрика ВебДрайвера)

4-рівень папки «test»: папки “resources” та ”java”

5-рівень папки «resources»: файл testng.xml – конфігураційний файл, який перелічує тести, що будуть запускатися

5-рівень папка “java”: ua/bms/test/pages/ - знаходяться самі тести з базовим тестом для всіх класів TestBase.java.

Після створення проекту, необхідно створити для нього репозиторій на git.

**Налаштування IDE.**

1. Необхідно інсталювати Eclipse.
2. За допомогою Maven створюємо проект для Eclipse IDE:
   1. В консолі переходимо в папку з проектом: selenium\_java\_training;
   2. Вводимо команду: mvn eclipse:eclipse;
   3. Таким чином в директорії проекту створюються додаткові файли для імпорту проекту в Eclipse.
3. Перейти в Eclipse ->File->Import->General->Existing project into Workspace. Обираємо папку selenium\_java\_training та додаємо проект в Eclipse.
4. Налаштовуємо всі залежності бібліотек в Eclipse(якщо наявна червона позначка на директорії проекту в Eclipse):
   1. Window->Preferences->Java->Build Path->Classpath Variables. Повинна бути змінна M2\_REPO. Якщо її нема, то необхідно додати. New -> Name=M2\_REPO, Path= «users\Vova\.m2\repository»(Шлях до робочої директорії Мавен .m2, куди Мавен складає свої тимчасові файли – локальний репозиторій. Зазвичай знаходиться в users\Vova\.m2\repository).

**Створення заготовки для тесту з допомогою рекордера.**

1. Використовується Mozilla Firefox.
2. Завантажуємо плагін Selenium IDE для Mozilla з <http://www.seleniumhq.org/> або з <https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/> .
3. Переходимо на сайт, який ми будемо тестувати.
4. Запускаємо Selenium IDE -> Option->Option->General “Enable experimental features”=TRUE.
5. Для того, щоб експортований файл (запис рекордера) включити в наш тест(в наш проект, що був згенерований з архітипу), потрібно змінити налаштування форматування:
   1. Optional переходимо на закладку Formats копіюємо налаштування, які вже є(обираємо наприклад Java/Junit 4/WebDriver -> Source -> копіюємо все);
   2. Add -> вставляємо раніше скопійоване, називаємо, наприклад, «My Format» -> Save -> OK

Або правимо існуючий (змінюємо Header, Footer, package)

* 1. Заходимо знову в Option->Option-> Formats та обираємо наш створений формат.

1. Копіюємо текст тесту та вставляємо в пакет, що розташований в директоріях src/test/java/назва пакету.
2. В тексті тесту змінюємо назву пакета та назву класу.
3. Необхідно також завантажити та встановити додатковий плагін для запуску TestNG Test: Help -> “Install New Software” Work with=” http://beust.com/eclipse ”, “Contact all update sites…”=FALSE.
4. Зайти на сторінку сайту, який буде тестуватися, з формою авторизації, виконати авторизацію із записуванням з допомогою рекордера;
5. Скопіювати код з закладки «Source»;
6. Створити новий клас в src/test/java з назвою LoginTest. В цьому класі виконати правку назви пакета та назви класу(клас повинен наслідувати клас TestBase)
7. Необхідно налаштувати src/main/resources/application.properties:

site.url=http://10.100.7.71:57775/osop/LoginForm.csp

browser.name=firefox

1. Вказуэмо клас(тест) який потрібно буде запускати: src/test/resources/testng.xml -> class name=….
2. Запускаємо тести. Run As – TestNG Test класу LoginTest.java АБО в консолі в папці проекту – mvn test.

Перевірка результатів тестування зберігається html файлі в target/surefire-reports/index.html