Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра програмного забезпечення



**ЗВІТ**

**Про виконання лабораторної роботи № 2:**

**«Тестування веб-додатків з використанням фреймворку Selenium»**

**Лектор:**

Ваврук І. Є.

**Виконав:**

студент групи ПЗІП-11

Бойчук Т.В.

**Прийняла:**

Ваврук І. Є.

∑ = \_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_

Львів – 2019

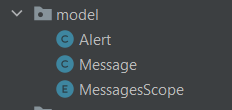
**Мета роботи:** Розробити java проект з використанням maven та log4j, створити автотести для тестування веб-сервісу gmail з використанням TestNg бібліотеки, Maven та фреймворку Selenium.

1. Створити Maven проект
2. Використовувати TestNG & Selenium для розроблення тестів
3. Одне завдання = один тест
4. Використовувати property file(s) для налаштувань проекту
5. Використовувати шаблон PageObject
6. Використовувати xpath згідно правил (не динамічно побудовані)
7. Використовувати DataProvider
8. Сформувати звіт

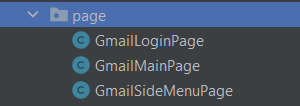
**ВИКОНАННЯ**

Відповідно до шаблону PageObject було створено:

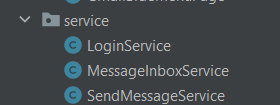
3 моделі



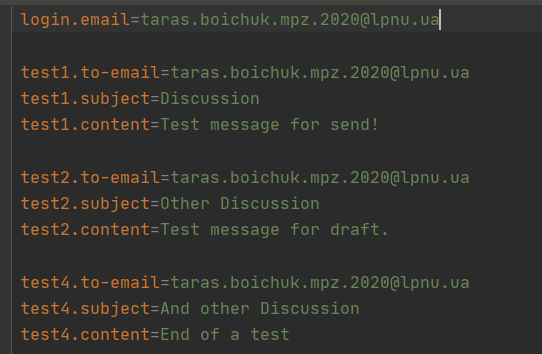
3 сторінки



3 сервіси



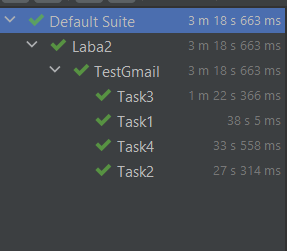
Було також створено клас PropertyManager, сутність якого надаватиме потрібні значення для наших тестів. PropertyManager використовуватиме значення які описані у файлі selenium-config.properties.



Код тестування:

**public class** TestGmail {  
  
 **private static** PropertyManager *propertyManager*;  
 **private static** ChromeDriverService *service*;  
 **private** WebDriver **webDriver**;  
 **private** LoginService **loginService**;  
 **private** MessageInboxService **messageInboxService**;  
 **private** SendMessageService **sendMessageService**;  
  
 @BeforeSuite  
 **public void** setup() {  
  
 *propertyManager* = PropertyManager.*getInstance*();  
 **loginService** = **new** LoginService();  
 **messageInboxService** = **new** MessageInboxService();  
 **sendMessageService** = **new** SendMessageService();  
 }  
  
 @BeforeTest  
 @SneakyThrows  
 **public void** doBeforeTest() {  
 *service* = **new** ChromeDriverService.Builder()  
 .usingDriverExecutable(**new** File(**"chromedriver.exe"**))  
 .usingAnyFreePort()  
 .build();  
  
 *service*.start();  
 }  
  
 @BeforeMethod  
 **public void** before(){  
 **webDriver** = **new** ChromeDriver();  
 **loginService**.setWebDriver(**webDriver**);  
 **messageInboxService**.setWebDriver(**webDriver**);  
 **sendMessageService**.setWebDriver(**webDriver**);  
 }  
  
  
 @Test(description = Constants.***TEST1\_DESC***)  
 **public void** Task1() {  
  
 **loginService**.loginToGoogleAccount(*propertyManager*.getEmail(), *propertyManager*.getPassword());  
 Assert.*assertEquals*(**webDriver**.getCurrentUrl(), **"https://mail.google.com/mail/u/0/#inbox"**);  
  
 **sendMessageService**.sendMessage(*propertyManager*.getTo1(), *propertyManager*.getSubject1(), *propertyManager*.getContent1());  
  
 **var** sentMessagesCountBeforeDelete = **messageInboxService**.getMessagesCount(MessagesScope.***SENT***);  
  
 System.***out***.println(sentMessagesCountBeforeDelete);  
 **messageInboxService**.removeMessage(1, MessagesScope.***SENT***);  
 **var** sentMessagesCountAfterDelete = **messageInboxService**.getMessagesCount(MessagesScope.***SENT***);  
  
 Assert.*assertEquals*(sentMessagesCountBeforeDelete - sentMessagesCountAfterDelete, sentMessagesCountBeforeDelete == 1 ? 0 : 1);  
 }  
  
 @Test(description = Constants.***TEST2\_DESC***)  
 **public void** Task2() {  
  
 **loginService**.loginToGoogleAccount(*propertyManager*.getEmail(), *propertyManager*.getPassword());  
 Assert.*assertEquals*(**webDriver**.getCurrentUrl(), **"https://mail.google.com/mail/u/0/#inbox"**);  
  
 **sendMessageService**.saveMessageAsDraft(*propertyManager*.getTo2(), *propertyManager*.getSubject2(), *propertyManager*.getContent2());  
  
 **var** messageInfo = **messageInboxService**.getDraftMessageInfo(1);  
  
 Assert.assertEquals(messageInfo.getTo(), *propertyManager*.getTo2());  
 Assert.assertEquals(messageInfo.getSubject(), *propertyManager*.getSubject2());  
 Assert.assertEquals(messageInfo.getContent(), *propertyManager*.getContent2());  
  
 **sendMessageService**.sendDraftedMessage(1);  
 }  
  
 @Test(description = Constants.***TEST3\_DESC***)  
 **public void** Task3() **throws** InterruptedException {  
  
 **loginService**.loginToGoogleAccount(*propertyManager*.getEmail(), *propertyManager*.getPassword());  
 Assert.*assertEquals*(**webDriver**.getCurrentUrl(), **"https://mail.google.com/mail/u/0/#inbox"**);  
  
 **var** messageInfoFromInbox1 = **messageInboxService**.getMessageInfo(1, MessagesScope.***INBOX***);  
 **var** messageInfoFromInbox2 = **messageInboxService**.getMessageInfo(2, MessagesScope.***INBOX***);  
 **var** messageInfoFromInbox3 = **messageInboxService**.getMessageInfo(3, MessagesScope.***INBOX***);  
  
 **messageInboxService**.markMessagesAsImportant(List.*of*(1, 2, 3), MessagesScope.***INBOX***);  
  
  
  
 **var** messageInfoFromImportant1 = **messageInboxService**.getMessageInfo(1, MessagesScope.***IMPORTANT***);  
 **var** messageInfoFromImportant2 = **messageInboxService**.getMessageInfo(2, MessagesScope.***IMPORTANT***);  
 **var** messageInfoFromImportant3 = **messageInboxService**.getMessageInfo(3, MessagesScope.***IMPORTANT***);  
  
 Assert.*assertEquals*(messageInfoFromInbox1, messageInfoFromImportant1);  
 Assert.*assertEquals*(messageInfoFromInbox2, messageInfoFromImportant2);  
 Assert.*assertEquals*(messageInfoFromInbox3, messageInfoFromImportant3);  
  
 **var** sentMessagesCountBeforeDelete = **messageInboxService**.getMessagesCount(MessagesScope.***IMPORTANT***);  
 **messageInboxService**.removeMessageList(List.*of*(1, 2, 3), MessagesScope.***IMPORTANT***);  
  
 Thread.*sleep*(1500);  
 System.***out***.println(**"ФУУх"**);  
 **var** sentMessagesCountAfterDelete = **messageInboxService**.getMessagesCount(MessagesScope.***IMPORTANT***);  
  
 Assert.*assertEquals*(sentMessagesCountBeforeDelete - sentMessagesCountAfterDelete, sentMessagesCountBeforeDelete == 3 ? 2 : 3);  
 }  
  
 @Test(description = Constants.***TEST4\_DESC***)  
 **public void** Task4() {  
  
 **loginService**.loginToGoogleAccount(*propertyManager*.getEmail(), *propertyManager*.getPassword());  
 Assert.*assertEquals*(**webDriver**.getCurrentUrl(), **"https://mail.google.com/mail/u/0/#inbox"**);  
  
 **var** alert = **sendMessageService**.sendMessageWithInvalidEmail(*propertyManager*.getSubject4(), *propertyManager*.getContent4());  
  
 Assert.assertEquals(alert.getContent(), ***ALERT\_CONTENT***);  
 Assert.assertEquals(alert.getTitle(), ***ALERT\_TITLE***);  
  
 **sendMessageService**.sendMessageAfterAlert(*propertyManager*.getTo4());  
  
 **var** messageInfo = **messageInboxService**.getMessageInfo(1, MessagesScope.***SENT***);  
  
 Assert.assertEquals(messageInfo.getTo(), *propertyManager*.getTo4());  
 Assert.assertEquals(messageInfo.getSubject(), *propertyManager*.getSubject4());  
 Assert.assertEquals(messageInfo.getContent(), *propertyManager*.getContent4());  
 }  
  
 @AfterMethod  
 **public void** doAfterTest() {  
  
 **webDriver**.close();  
 **webDriver**.quit();  
 }  
  
 @AfterSuite  
 **public void** doAfterSuite(){  
 **webDriver**.quit();  
 *service*.stop();  
 }  
}

Результати тестування:



**ВИСНОВОК**

В результаті виконання даної роботи було розроблено java проект з використанням maven та log4j, створино автотести для 4 завдань для тестування веб-сервісу gmail з використанням TestNg бібліотеки, Maven та фреймворку Selenium.