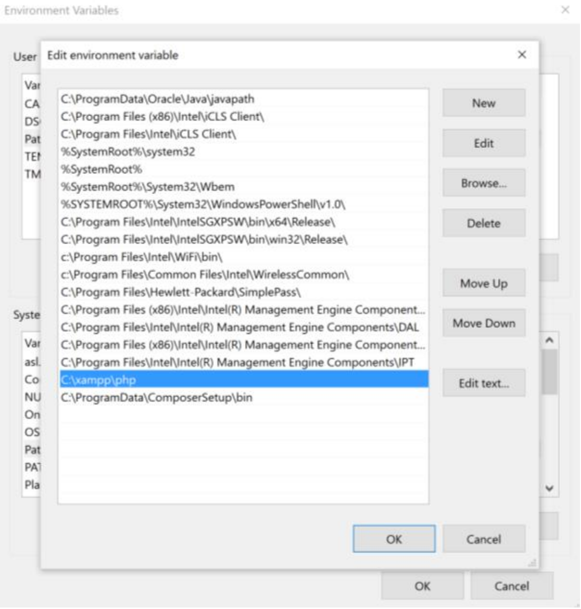
Lab: Simulated acceptance testing using Codeception

CPE 333 Software Engineering
Computer Engineering Department
King Mongkut's University of Technology
Due date: Wednesday, April 6th, 2016 at 11:59 PM.

Acceptance testing can be performed by a non-technical person. That person can be your tester, manager or even client. If you are developing a web-application (and probably you are) the tester needs nothing more than a web browser to check that your site works correctly - Codeception guide on acceptance testing http://codeception.com/docs/03-AcceptanceTests

Requirement: PHP5.3

- Install PHP5.3 or higher: If you have already installed it, skip this step.
- If you are using Windows and have already installed XAMPP, you do not need to install PHP.
    - http://windows.php.net/download/
- If you do not have PHP and would like to install it, the easiest way is to install XAMPP on Windows.
- Please check that PHP is known by your system. If not, check that you have added PHP path to your system environment.



```
C:\ProgramData\Oracle\Java\javapath
C:\Program Files (x86)\Intel\iCLS Client\
C:\Program Files\Intel\iCLS Client\
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
C:\Program Files\Intel\IntelSGXPSW\bin\x64\Release\
C:\Program Files\Intel\IntelSGXPSW\bin\win32\Release\
c:\Program Files\Intel\WiFi\bin\
c:\Program Files\Common Files\Intel\WirelessCommon\
C:\Program Files\Hewlett-Packard\SimplePass\
C:\Program Files (x86)\Intel\Intel(R) Management Engine Component...
C:\Program Files\Intel\Intel(R) Management Engine Components\DAL
C:\Program Files (x86)\Intel\Intel(R) Management Engine Component...
C:\Program Files\Intel\Intel(R) Management Engine Components\IPT
C:\xampp\php
C:\ProgramData\ComposerSetup\bin
```

- If you are using other OS, please follow the instructions at http://php.net/downloads.php

Instructions

1. If you are using Unix-like OS, to install Codeception, you can use the following commands.

Install PHAR globally

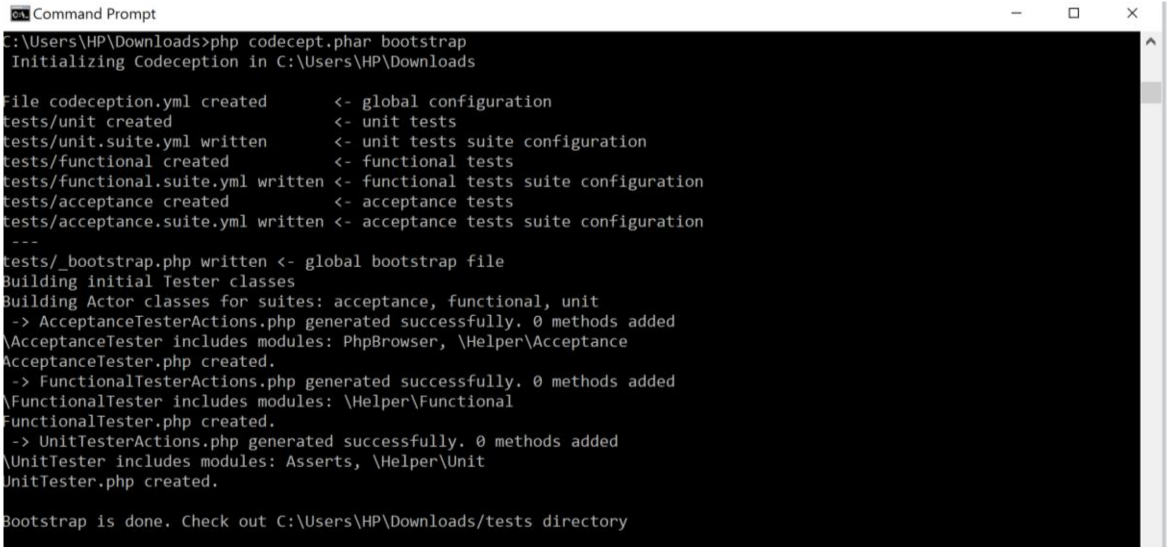
```
sudo curl -LsS http://codeception.com/codecept.phar -o /usr/local/bin/codecept
sudo chmod a+x /usr/local/bin/codecept
```

More details: http://codeception.com/quickstart#.VvoLQ_l942w

2. If you are using Windows, download codecept.phar at http://codeception.com/install#.VvoLtfl942y . Assume that the file has been downloaded and located in ..\Downloads.

3. To create project structure and config files for Codeception, in project root folder, type:

```
php codecept.phar bootstrap
```

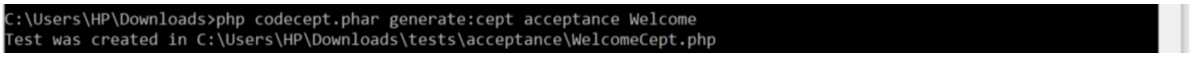


```
C:\Users\HP\Downloads>php codecept.phar bootstrap
 Initializing Codeception in C:\Users\HP\Downloads

File codeception.yml created       <- global configuration
tests/unit created                 <- unit tests
tests/unit.suite.yml written       <- unit tests suite configuration
tests/functional created           <- functional tests
tests/functional.suite.yml written <- functional tests suite configuration
tests/acceptance created           <- acceptance tests
tests/acceptance.suite.yml written <- acceptance tests suite configuration
 ---
tests/_bootstrap.php written <- global bootstrap file
Building initial Tester classes
Building Actor classes for suites: acceptance, functional, unit
 -> AcceptanceTesterActions.php generated successfully. 0 methods added
\AcceptanceTester includes modules: PhpBrowser, \Helper\Acceptance
AcceptanceTester.php created.
 -> FunctionalTesterActions.php generated successfully. 0 methods added
\FunctionalTester includes modules: \Helper\Functional
FunctionalTester.php created.
 -> UnitTesterActions.php generated successfully. 0 methods added
\UnitTester includes modules: Asserts, \Helper\Unit
UnitTester.php created.

Bootstrap is done. Check out C:\Users\HP\Downloads/tests directory
```

4. Create an acceptance test by using

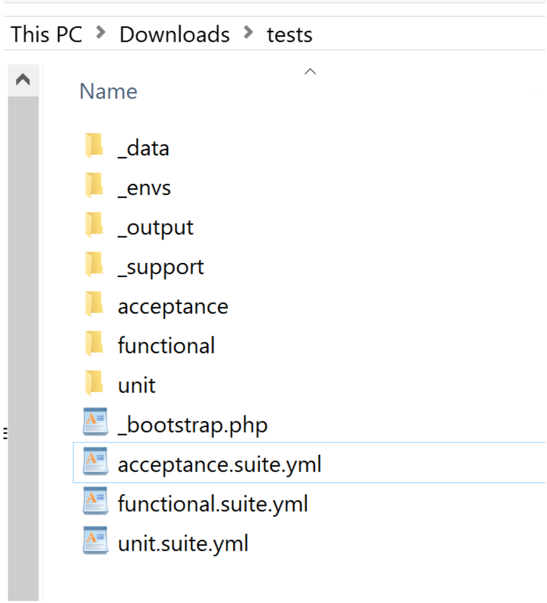
```
codecept generate:cept acceptance {name}
```

Note that {name} is your test name. In the following figure, the name of the test is Welcome.



```
C:\Users\HP\Downloads>php codecept.phar generate:cept acceptance Welcome
Test was created in C:\Users\HP\Downloads\tests\acceptance\WelcomeCept.php
```

5. Go to your root directory of the test. You will see the file named acceptance.suite.yml .



This PC › Downloads › tests

Name

_data
_envs
_output
_support
acceptance
functional
unit
_bootstrap.php
acceptance.suite.yml
functional.suite.yml
unit.suite.yml

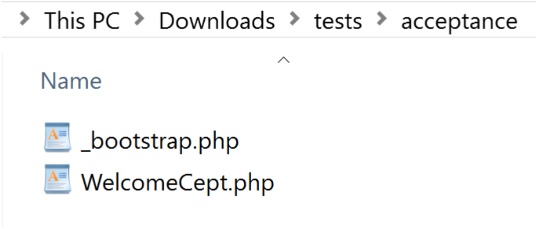
6. Open the file. You will see the following content. Change the url to [**http://modpd167.lib.kmutt.ac.th:51902/**](http://modpd167.lib.kmutt.ac.th:51902/) **which is the link to the new MyLE. Save and close the file.**

```
# Codeception Test Suite Configuration
#
# Suite for acceptance tests.
# Perform tests in browser using the WebDriver or PhpBrowser.
# If you need both WebDriver and PHPBrowser tests - create a separate suite.

class_name: AcceptanceTester
modules:
    enabled:
        - PhpBrowser:
            url: http://modpd167.lib.kmutt.ac.th:51902/
        - \Helper\Acceptance
```

Each time you change configuration in Codeception you should run the build command.

```
php codeception.phar build
```

7. Then, go to ..\tests\acceptance. You will see WelcomeCept.php. Open the file with a text editor you prefer. It's now time to write your first test.

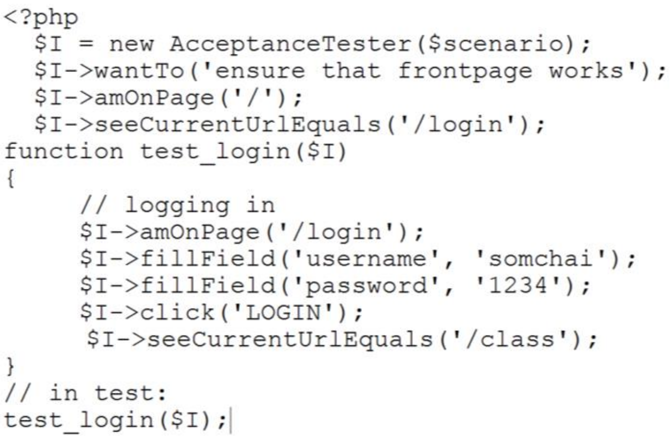


This PC > Downloads > tests > acceptance

Name

_bootstrap.php

WelcomeCept.php

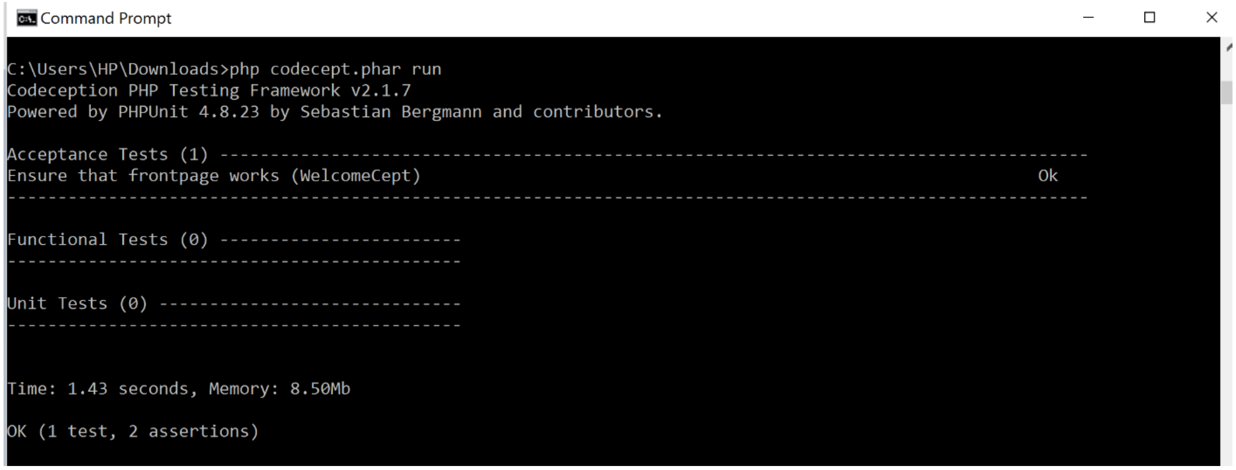


```
<?php
  $I = new AcceptanceTester($scenario);
  $I->wantTo('ensure that frontpage works');
  $I->amOnPage('/');
  $I->seeCurrentUrlEquals('/login');
function test_login($I)
{
      // logging in
      $I->amOnPage('/login');
      $I->fillField('username', 'somchai');
      $I->fillField('password', '1234');
      $I->click('LOGIN');
       $I->seeCurrentUrlEquals('/class');
}
// in test:
test_login($I);
```

Note that the username and password to log in to myLE as the student role are khanok.rat and 1234, respectively. The username and password to log in to myLE as the teacher role are somchai and 1234.

8. Then run the test. Type:

php codecept.phar run



```
C:\Users\HP\Downloads>php codecept.phar run
Codeception PHP Testing Framework v2.1.7
Powered by PHPUnit 4.8.23 by Sebastian Bergmann and contributors.

Acceptance Tests (1) --------------------------------------------------------------------------------------
Ensure that frontpage works (WelcomeCept)                                                           Ok
------------------------------------------------------------------------------------------------------------

Functional Tests (0) -----------------------
--------------------------------------------

Unit Tests (0) -----------------------------
--------------------------------------------


Time: 1.43 seconds, Memory: 8.50Mb

OK (1 test, 2 assertions)
```

Then observe the test result.

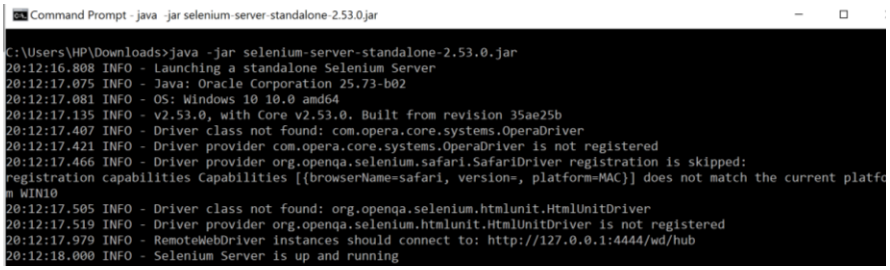
Your PhpBrowser tests we wrote previously can be executed inside a real browser with Selenium WebDriver.

Common PhpBrowser drawbacks:

- you can click only on links with valid urls or form submit buttons
- you cannot fill fields that are not inside a form
- you cannot work with JavaScript interactions: modal windows, datepickers, etc.

To use WebDriver, follow the below steps

1. Download Selenium Server
2. Launch the daemon: java -jar selenium-server-standalone-2.xx.xxx.jar



Command Prompt - java -jar selenium-server-standalone-2.53.0.jar

```
C:\Users\HP\Downloads>java -jar selenium-server-standalone-2.53.0.jar
20:12:16.808 INFO - Launching a standalone Selenium Server
20:12:17.075 INFO - Java: Oracle Corporation 25.73-b02
20:12:17.081 INFO - OS: Windows 10 10.0 amd64
20:12:17.135 INFO - v2.53.0, with Core v2.53.0. Built from revision 35ae25b
20:12:17.407 INFO - Driver class not found: com.opera.core.systems.OperaDriver
20:12:17.421 INFO - Driver provider com.opera.core.systems.OperaDriver is not registered
20:12:17.466 INFO - Driver provider org.openqa.selenium.safari.SafariDriver registration is skipped:
registration capabilities Capabilities [{browserName=safari, version=, platform=MAC}] does not match the current platfo
m WIN10
20:12:17.505 INFO - Driver class not found: org.openqa.selenium.htmlunit.HtmlUnitDriver
20:12:17.519 INFO - Driver provider org.openqa.selenium.htmlunit.HtmlUnitDriver is not registered
20:12:17.979 INFO - RemoteWebDriver instances should connect to: http://127.0.0.1:4444/wd/hub
20:12:18.000 INFO - Selenium Server is up and running
```

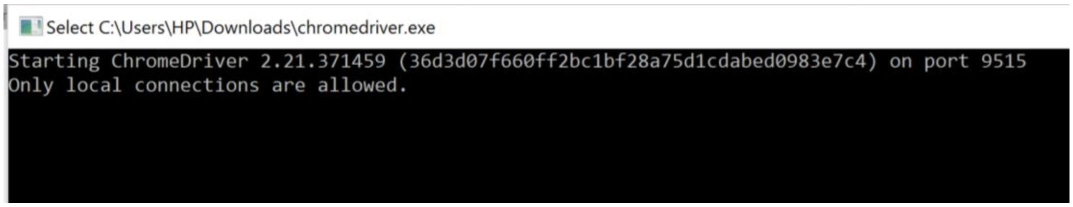
3. Configure this module (in acceptance.suite.yml) by setting url and browser:

   The only thing we need to change is to reconfigure and **rebuild** the AcceptanceTester class, to use WebDriver instead of PhpBrowser.

```
#modules:
#       enabled:
#           - WebDriver:
#               url: 'http://modpd167.lib.kmutt.ac.th:51902/'
#               browser: chrome
```

4. Launch ChromeDriver (WebDriver for Chrome) or others depended on your browser.

   https://sites.google.com/a/chromium.org/chromedriver/downloads



```
Starting ChromeDriver 2.21.371459 (36d3d07f660ff2bc1bf28a75d1cdabed0983e7c4) on port 9515
Only local connections are allowed.
```

5. Run your test again. Chrome will be automatically run and the test starts.

9. Select one test scenario. The below table is an example of each test and its test result. You have to show your own table in the report. You are NOT allowed to select the logging in scenarios as the example for your own work.

Test ID	Test Description	Test Data	Step	Pre-Condition	Post-Condition	Expected Result	Actual Result	Status
1	User can login if he enters the correct username and password.	username: somchai password: 1234	1. Open the homepage 2. Enter 'somchai' for username 3. Enter '1234' for password 4. Click 'LOGIN' button	User account must exist in the database	None	The webpage redirect to the class list page. The username is shown in the top right menu.	The webpage redirect to the class list page. The username is shown in the top right menu.	Pass
2	User cannot login if he doesn't enter username or password	username: somchai	1. Open the homepage 2. Enter 'somchai' for username 3. Click 'LOGIN' button	User account must exist in the database	None	The login form displays the error message and prevents user from logging in.	The webpage redirect to the class list page. The username is shown in the top right menu.	Fail

10. Since this work is to simulate the acceptance testing of MyLE, before write you own automated test scripts for testing the website, you should go the website **http://modpd167.lib.kmutt.ac.th:51902/.** Then, try to use the website and select one scenario (per group) that you think you are going to write your own tests.

Note that

- The username and password to log in to MyLE as the student role are "khanok.rat" and "1234", respectively. The username and password to log in to MyLE as the teacher role are "somchai" and "1234".
- Functions available for you to write your test are available at

http://codeception.com/docs/modules/PhpBrowser

http://codeception.com/docs/modules/WebDriver

You may need to know an attribute/text of a web element that you are going to refer to.

For example, “Click” function

click

Perform a click on a link or a button, given by a locator. If a fuzzy locator is given, the page will be searched for a button, link, or image matching the locator string. For buttons, the “value” attribute, “name” attribute, and inner text are searched. For links, the link text is searched. For images, the “alt” attribute and inner text of any parent links are searched.

The second parameter is a context (CSS or XPath locator) to narrow the search.

Note that if the locator matches a button of type submit, the form will be submitted.

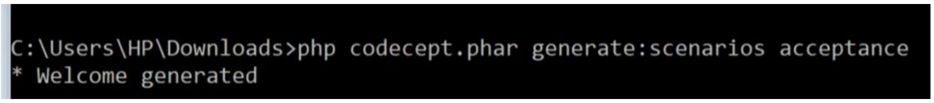
```
<?php
// simple link
$I->click('Logout');
// button of form
$I->click('Submit');
// CSS button
$I->click('#form input[type=submit]');
// XPath
$I->click('//form/*[ * `type=submit]');`
// link in context
$I->click('Logout', '#nav');
// using strict locator
$I->click(['link' => 'Login']);
?>
```

11. Submit your report and all the files in your test folder on MyLE.

    In your report, you should provide

    1. The table as shown in the step 9.

    2. Your test flow step-by-step by using the following command.



```
C:\Users\HP\Downloads>php codecept.phar generate:scenarios acceptance
* Welcome generated
```

The above command is used to generate the test flow of the Welcome test. The generated output file will be located at ..\tests_data\scenarios\.

The example of generated test flow of the Welcome test is as follows:

```
I WANT TO ENSURE THAT FRONTPAGE WORKS

I am on page '/'
I see current url equals '/logn'
I am on page '/login'
I fill field 'username'," 'somchai'
I fill field 'password'," '1234'
I click 'LOGIN'
I see current url equals '/class'
```

3. A capture of your test result.