

BDD(Behave)

19 August 2023 08:55

What is BDD?

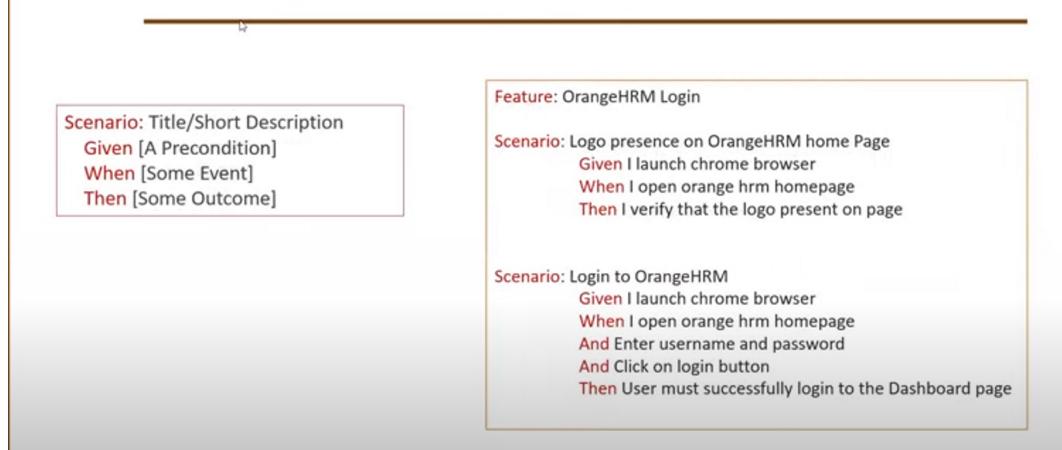
Agile software dev techniq. Developers(technical) and business partners(non-technincal).

Behave operates on directory conatains:

Feature files contains scenario written by business analyst/sponser

Step dir for python step implementation

BDD Scenarios and Feature Files In Gherkin



Components to understand BDD

Feature files---> .feature

|

Scenarios

|

Steps

Using Gherkin we can write the feature file

Scenario: Title/Short Description

```
Given [A Precondition]
When [Some Event]
Then [Some Outcome]
```

***And keyword to add 2 or more event or action in feature

[behave - PyPI](#)

Environment setup

1. Install python
2. Pycharm IDE
3. Selenium library
4. Behave pip install behave or pip install -u behave

First selenium Test

1. Create project---New project---><give name>---> craete project
2. Install selenium & behave in interpreter settings.
3. Selenium - Drivers
4. Folder structure
 - a. Feature- for feature files
 - b. Inside feature folder create Steps folder- for step defintion
 - c. Add feature file in feature folder <nopcommerce.feature>





- d. 1 | ✓
- e. Click and install the plugin
- f. To run feature file
 - i. Open terminal
 - ii. Give Behave <feature file path>
- iii. Run the file
- iv. Which will give you sample step definition
- v. copy all the step definition and create .py file in step folder and paste the copied step definition
- vi. Give "import behave import *" will import all gherkin keyworded
- vii. Also import selenium webdriver
- viii. Modify and write scenarios for each step

```

nopcommerce.feature x nop_login.py x
1  from behave import *
2  from selenium import webdriver
3  from selenium.webdriver.common.by import By
4
5
6  @given(u': Launch browser')
7  def launch_browser(context):
8      context.driver=webdriver.Chrome()
9
10
11  @when(u': Open nopcommerce page')
12  def open_app(context):
13      context.driver.get("https://admin-demo.nopcommerce.com/login?ReturnUrl=%2Fadmin%2E")
14
15
16  @then(u': Verify the presence of logo')
17  def verify_logo(context):
18      status = context.driver.find_element(By.XPATH, "/html/body/div[6]/div/div/div/div[1]/h1").is_displayed()
19      assert status==True
20
21
22  @then(u': Close browser')
23  def close_browser(context):
24      context.driver.quit()
25

```

```
(venv) C:\Users\user\PycharmProjects\shirlyApp\behave_framework\features>behave nopcommerce.feature
Feature: NopCommerce Login # nopcommerce.feature:1

  Scenario: Logo presence on NopCommerce login # nopcommerce.feature:2
    Given : Launch browser # steps/nop_login.py:6

  DevTools listening on ws://127.0.0.1:53902/devtools/browser/1ed4a48b-7949-4fc2-912b-c6ef1dde0cc9
    When : Open nopcommerce page # steps/nop_login.py:11
    Then : Verify the presence of logo # steps/nop_login.py:16
    And : Close browser # steps/nop_login.py:22

1 feature passed, 0 failed, 0 skipped
1 scenario passed, 0 failed, 0 skipped
4 steps passed, 0 failed, 0 skipped, 0 undefined
Took 0m9.515s
```

```
(venv) C:\Users\user\PycharmProjects\shirlyApp\behave_framework\features>
```

Pass step parameters

```
1 Feature: NopCommerce Login
2 Scenario: Login to NopCommerce app with valid parameter
3   Given : Launch browser
4   When : Open nopcommerce page
5   And : Enter Username "admin@yourstore.com" and Password "admin"
6   And : Click on Submit Button
7   Then : User must successfully enter into the dashboard page
```

Create the step file and pass parameters in {var}

```
from behave import *
from selenium import webdriver
from selenium.webdriver.common.by import By

@when(': Enter Username "{user}" and Password "{password}"')
def step_impl(context, user, password):
    context.driver.find_element(By.ID, 'Email').clear()
    context.driver.find_element(By.ID, 'Email').send_keys(user)
    context.driver.find_element(By.ID, 'Password').clear()
    context.driver.find_element(By.ID, 'Password').send_keys(password)

@when(': Click on Submit Button')
def step_impl(context):
    context.driver.find_element(By.XPATH, '//button[@type="submit"]').click()

@then(': User must successfully enter into the dashboard page')
def step_impl(context):
    status = context.driver.find_element(By.XPATH, '/html/body/div[3]/div[1]/div[1]/h1').text
    assert status=="Dashboard"
```

Scenario outline

Pass multiple parameter- use <> to define multiple parameter and list those parameters under example section

```

    Scenario Outline: Login to NopCommerce app with multiple parameter
      Given : Launch browser
      When : Open nopcommerce page
      And : Enter Username "<username>" and Password "<password>"
      And : Click on Submit Button
      Then : verify User login status "<status>"

    Examples:
    | username          | password | status |
    | admin@yourstore.com | admin   | True  |
    | admin@yourstore.com | admin123 | False |
    | abc@yourstore.com  | abc123  | False |
    | xyz@mystore.com   | mystore | False |

```

In step file create new step definition and handle the negative test cases using try...except block

```

@then(': verify User login status "{status}"')
def step_impl(context, status):
    try:
        dashboard_txt = context.driver.find_element(By.XPATH, '/html/body/div[3]/div[1]/div[1]/h1').text
    except:
        assert status=="False"
    else:
        if status=="True":
            assert dashboard_txt=="Dashboard"

```

Background

```

Feature: NopCommerce Login
  Scenario: login to nopcommerce website
    Given Launch browser
    When Open nopcommerce page
    And Enter Username "admin@yourstore.com" and Password "admin"
    And Click on Submit Button
    Then User must successfully enter into the dashboard page

  Scenario: customers
    Given Launch browser
    When Open nopcommerce page
    And Enter Username "admin@yourstore.com" and Password "admin"
    And Click on Submit Button
    Then User must successfully enter into the dashboard page
    When click on customers in dashborad
    Then customers page should display

  Scenario: online customers
    Given Launch browser
    When Open nopcommerce page
    And Enter Username "admin@yourstore.com" and Password "admin"
    And Click on Submit Button
    Then User must successfully enter into the dashboard page
    When click on online customers in dashborad
    Then online |customers page should display

```

Feature: NopCommerce Login

Scenario: login to nopcommerce website

Given Launch browser

When Open nopcommerce page

And Enter Username "admin@yourstore.com" and Password "admin"

And Click on Submit Button

Then User must successfully enter into the dashboard page

Scenario: customers

Given Launch browser

When Open nopcommerce page

And Enter Username "admin@yourstore.com" and Password "admin"

And Click on Submit Button

Then User must successfully enter into the dashboard page

When click on customers in dashborad

Then customers page should display

Scenario: online customers

Given Launch browser

When Open nopcommerce page

And Enter Username "admin@yourstore.com" and Password "admin"

And Click on Submit Button

Then User must successfully enter into the dashboard page

When click on online customers in dashborad

Then online customers page should display

Feature: NopCommerce Login

Background: Steps to get executed before each scenarios

Given Launch browser

When Open nopcommerce page

And Enter Username "admin@yourstore.com" and Password "admin"

And Click on Submit Button

Scenario: login to nopcommerce website

Then User must successfully enter into the dashboard page

Scenario: customers

When click on customers in dashborad

Then customers page should display

Scenario: online customers

When click on online customers in dashborad

Then online customers page should display

Step implementation for background

```
import time
```

```
from behave import *
```

```
from selenium import webdriver
```

```
from selenium.webdriver.common.by import By
```

```
@when('click on customers in dashborad')
```

```
def step_impl(context):
```

```
    context.driver.find_element(By.XPATH, '//ul[@role="menu"]/li[4]/a/p').click()
```

```
    context.driver.find_element(By.XPATH, '//ul[@role="menu"]/li[4]/ul/li[1]/a/p').click()
```

```
@then('customers page should display')
```

```
def step_impl(context):
```

```
    s_text = context.driver.find_element(By.XPATH, "//h1[contains(text(),'Customers')]").text
```

```
    assert s_text=="Customers"
```

```
@when('click on online customers in dashborad')
```

```
def step_impl(context):
```

```
    context.driver.find_element(By.XPATH, '//ul[@role="menu"]/li[4]/a/p').click()
```

```
    time.sleep(5)
```

```
context.driver.find_element(By.XPATH, '//ul[@role="menu"]/li[4]/ul/li[3]/a/p').click()

@then('online customers page should display')
def step_impl(context):
    s_text = context.driver.find_element(By.XPATH, '//h1[contains(text(),"Online customers")]').text
    assert s_text == "Online customers"
```

ROBOTFRAMEWORK

<https://robotframework.org/>

Pip install

- Robotframework
- Robotframework-seleniumlibrary

Robot selenium keyword

<https://robotframework.org/SeleniumLibrary/SeleniumLibrary.html>