**LAPORAN PRAKTIKUM**

**PENGUJIAN PERANGKAT LUNAK**

**MODUL 6**

**Automation Unit Testing (Menguji Class dengan PHPUNIT)**

****

**DISUSUN OLEH :**

|  |  |
| --- | --- |
| **2250081053** | **MUKHAMAD DIVA M.A** |

**DSE B**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS SAINS DAN INFORMATIKA**

**UNIVERSITAS JENDERAL ACHMAD YANI**

**TAHUN AJARAN 2024/2025**

# DAFTAR ISI

[DAFTAR ISI i](#_Toc180361779)

[DAFTAR GAMBAR ii](#_Toc180361780)

[BAB I HASIL PRAKTIKUM 1](#_Toc180361781)

[1.1 Judul Pekerjaan Praktikum 1](#_Toc180361782)

[1.1.1 Judul Langkah Kerja/Nama Source Code \*(Pilih Salah Satu) 1](#_Toc180361783)

[1.1.2 Screenshoot 1](#_Toc180361784)

[1.1.3 Analisis 1](#_Toc180361785)

[BAB II TUGAS PRAKTIKUM 2](#_Toc180361786)

[2.1 Judul Tugas Praktikum 2](#_Toc180361787)

[2.1.1 Judul Langkah Kerja/Nama Source Code 2](#_Toc180361788)

[2.1.2 Screenshoot 2](#_Toc180361789)

[2.1.3 Analisis 2](#_Toc180361790)

[BAB III KESIMPULAN 3](#_Toc180361791)

# DAFTAR GAMBAR

# 

# HASIL PRAKTIKUM

## Latihan

### Stub

1. Buat File baru pada folder bin dengan nama MyClass.php

<?php

namespace PHPUnit\Framework;

use PHPUnit\Framework\TestCase;

class MyClass

{

public function MyMethod()

{

return "hello";

}

}

?>

1. Buat file baru pada folder bin dengan nama StubTest1.php

<?php

namespace PHPUnit\Framework;

use PHPUnit\Framework\TestCase;

include "MyClass.php";

class StubTest1 extends TestCase

{

public function testStub()

{

$stub = $this->createMock(MyClass::class);

$stub->method('MyMethod')->willReturn("foo");

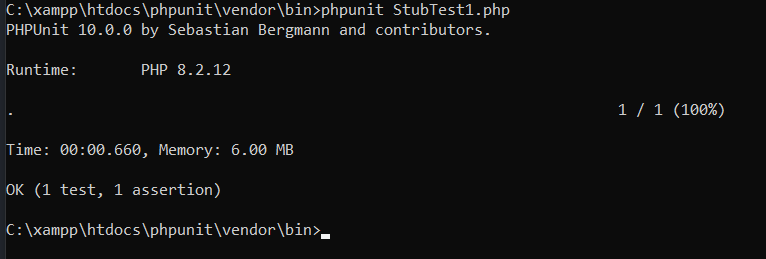
$this->assertSame("foo", $stub->MyMethod());

}

}

?>

1. Jalankan testcase dengan phpunit



Gambar 1 Test StubTest1.php

1. Ubah kode pada StubTest1.php

public function testStub()

{

$stub = $this->createMock(MyClass::class);

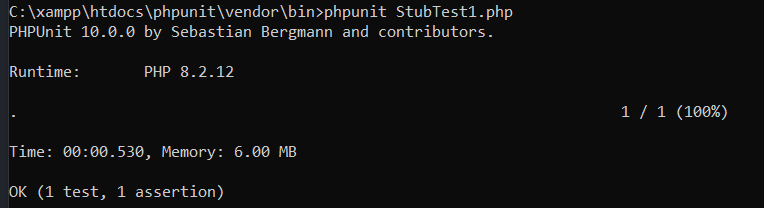
$stub->method('MyMethod')->will($this-

>returnArgument(0));

$this->assertSame('foo', $stub->MyMethod('foo'));

}

1. Jalankan Test



Gambar 2 Test ke 2 StubTest1.php

Analisis :

### Testable Class

1. Buat file baru di folder bin dengan nama Testable.php

<?php

namespace PHPUnit\Framework;

class Testable

{

public $trueProperty = true;

public $resetMe = true;

public $testArray = array(

'first key' => 1,

'second key' => 2

);private $testString = "I do love me some strings";

public function \_\_construct()

{

}

public function addValues($valueOne,$valueTwo) {

return $valueOne+$valueTwo;

}

public function getTestString()

{

return $this->testString;

}

}

?>

1. Buat file baru dengan nama TestableTest.php

<?php

namespace PHPUnit\Framework;

use PHPUnit\Framework\TestCase;

include "Testable.php";

class TestableTest extends TestCase

{

private $\_testable = null;

public function setUp()

{$this->\_testable = new Testable();

}

public function tearDown()

{

$this->\_testable = null;

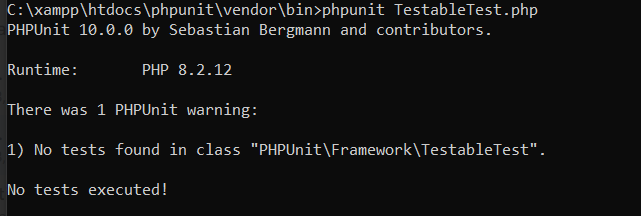
}

//Selanjutnya lanjutkan dari sini

}

?>

1. Jalankan testcase



Gambar 3 Test TestableTest.php

Analisis:

### True or false

1. Buat file baru di bin dengan nama TestableTest1.php

<?php

namespace PHPUnit\Framework;

use PHPUnit\Framework\TestCase;

include "Testable.php";

class TestableTest1 extends TestCase

{

private $\_testable = null;

public function setUp()

{

$this->\_testable = new Testable();

}

public function tearDown()

{

$this->\_testable = null;

}

//True or False

public function testTruePropertyIsTrue()

{

$this->assertTrue($this->\_testable->trueProperty,"trueProperty isn't true");

}

public function testTruePropertyIsFalse()

{

$this->assertFalse($this->\_testable->trueProperty,"trueProperty isn't false");

}

}

?>

1. Jalankan Testcase

# TUGAS PRAKTIKUM

## [Judul Pekerjaan Praktikum]

### [Judul Langkah Kerja/Nama Source Code \*(Pilih Salah Satu)]

### Screenshoot

### Analisis

# KESIMPULAN

[Gunakan style “Kesimpulan”] Tuliskan hasil pengamatan dari kegiatan praktikum mengenai modul yang dikerjakan yang juga relevan dengan tujuan dari pertemuannya.