

Pruebas unitarias Agregar medios de pago

A diferencia del *sprint* anterior, en esta ocasión los medios de pagos están divididos en dos tipos: tarjetas de débito y de crédito.

Para realizar las pruebas unitarias se crean los datos de prueba, estos están separados en dos archivos: ***DebitcardFixture.php*** y ***CreditcardFixture.php***.

Los siguientes son imágenes que muestran dichos datos:

```
public $records = array(
    array(
        'id' => 1,
        'card_number' => '1185417256930701',
        'nip' => '1122',
        'csc' => '2211',
        'expiration_date' => '2018-12-10',
        'brand' => 'Visa',
        'balance' => 1000,
        'check_id' => 0
    ),
    array(
        'id' => 2,
        'card_number' => '2185417256930702',
        'nip' => '3344',
        'csc' => '4433',
        'expiration_date' => '2018-12-10',
        'brand' => 'Master Card',
        'balance' => 2000,
        'check_id' => 0
    ),
    array(
        'id' => 3,
        'card_number' => '3185417256930703',
        'nip' => '5566',
        'csc' => '6655',
        'expiration_date' => '2018-12-10',
        'brand' => 'American Express',
        'balance' => 3000,
        'check_id' => 0
    )
);
```

1. *Fixture* de tarjetas de débito

```

public $records = array(
    array(
        'id' => 100,
        'card_number' => '4185417256930711',
        'nip' => '1234',
        'csc' => '4321',
        'expiration_date' => '2018-12-10',
        'brand' => 'Visa',
        'card_limit' => 1000,
        'check_id' => 0
    ),
    array(
        'id' => 101,
        'card_number' => '5185417256930712',
        'nip' => '5678',
        'csc' => '8765',
        'expiration_date' => '2018-12-10',
        'brand' => 'Master Card',
        'card_limit' => 2000,
        'check_id' => 0
    ),
    array(
        'id' => 102,
        'card_number' => '6185417256930713',
        'nip' => '9012',
        'csc' => '2109',
        'expiration_date' => '2018-12-10',
        'brand' => 'American Express',
        'card_limit' => 3000,
        'check_id' => 0
    )
);

```

2. Fixture de tarjetas de crédito

El siguiente paso es crear las funciones que realizan las pruebas para los controladores de ambos tipos de tarjetas, en los archivos ***DebitcardControllerTest.php*** y ***CreditcardControllerTest.php***.

```

public function testAdd()
{
    $result = $this->testAction('/debitcard/add');
    debug($result);
}

```

3. Función de prueba para las tarjetas de débito

```
public function testAdd()  
{  
    $result = $this->testAction('/creditcard/add');  
    debug($result);  
}
```

4. Función de prueba para tarjetas de crédito

Como paso final se ejecutan las pruebas desde la aplicación. Las siguientes capturas de pantalla muestran que ambas pruebas han sido aprobadas.

Running DebitcardControllerTest

✓ PASSED

testAdd (0.10700607299805 seconds)

1/1 test methods complete: 1 passes, 0 fails, 0 assertions and 0 exceptions.

Time: 0.10700607299805 seconds

Peak memory: 6,307,080 bytes

[Run more tests](#) | [Show Passes](#) | [Enable Debug Output](#) | [Analyze Code Coverage](#)

5. Aprobación de prueba en tarjetas de débito

Running CreditcardControllerTest

✓ PASSED

testAdd (0.092005014419556 seconds)

1/1 test methods complete: 1 passes, 0 fails, 0 assertions and 0 exceptions.

Time: 0.092005014419556 seconds

Peak memory: 6,308,088 bytes

[Run more tests](#) | [Show Passes](#) | [Enable Debug Output](#) | [Analyze Code Coverage](#)

6. Aprobación de prueba en tarjetas de crédito