

UNIVERSITY OF THESSALY
SCHOOL OF ENGINEERING



DEPARTMENT OF ELECTRICAL
& COMPUTER ENGINEERING

Project: Unit Tests

Μάθημα: Τεχνολογίες blockchain και αποκεντρωμένες εφαρμογές

Διδάσκων: ΦΩΤΗ ΜΑΓΔΑΛΗΝΗ

Ακ. έτος 2020-21, εαρινό εξάμηνο

Ονοματεπώνυμο: Καραμούστου Βασιλική

AEM: 02424

Contents

<i>Project: Unit Tests</i>	1
Unit Test 1	3
Unit Test 2	3
Unit Test 3	4
Unit Test 4	4
Αποτελέσματα	5

Unit Test 1

Ζητούμενο: Unit test όπου ο highest bidder επιχειρεί να κάνει withdraw. Το unit test πρέπει να αποτυγχάνει.

Ο κώδικας που υλοποιεί το unit test παρατίθεται παρακάτω.

```
const Auction = artifacts.require("Auction");
const truffleAssert = require('truffle-assertions');

contract("Auction unit test 1", async accounts => {
  it("withdraw() should fail if msg.sender is the highestBidder", async () => {
    const instance = await Auction.deployed();
    await instance.bid({value:50000, from:accounts[1]});
    await truffleAssert.reverts(
      instance.withdraw({from:accounts[1]}),
      "You can not be the highestBidder in order to proceed!"
    );
  });
});
```

Unit Test 2

Ζητούμενο: Unit test που επιβεβαιώνει ότι η μεταβλητή highestBid τίθεται σωστά μετά από ένα νέο bid.

Ο κώδικας που υλοποιεί το unit test παρατίθεται παρακάτω.

```
const Auction = artifacts.require("Auction");
const truffleAssert = require('truffle-assertions');

contract("Auction unit test 2", async accounts => {
  it("HighestBid should get updated after a new bid", async () => {
    const instance = await Auction.deployed();
    await instance.bid({value:60000, from:accounts[1]});
    highestBid = await instance.highestBid();
    assert.equal(highestBid, 60000);
  });
});
```

Unit Test 3

Ζητούμενο: Unit test που επιβεβαιώνει ότι το balance του contract αυξάνεται κατάλληλα μετά από κάθε bid.

Ο κώδικας που υλοποιεί το unit test παρατίθεται παρακάτω.

```
const Auction = artifacts.require("Auction");
const truffleAssert = require('truffle-assertions');

contract("Auction unit test 3", async accounts => {
  it("Contract's balance should get updated after a new bid", async () => {
    const instance = await Auction.deployed();
    const oldBalance = await web3.eth.getBalance(instance.address);
    assert.equal(oldBalance, 0);
    await instance.bid({value:100000, from:accounts[2]});
    const newBalance = await web3.eth.getBalance(instance.address);
    assert.equal(newBalance, 100000);
  });
});
```

Unit Test 4

Ζητούμενο: Unit test που τρέχει όλη τη διαδικασία των προηγούμενων εργαστηρίων (bid 5, bid 10, withdraw 5) και επιβεβαιώνει ότι η διαδικασία δεν αποτυγχάνει.

Ο κώδικας που υλοποιεί το unit test παρατίθεται παρακάτω.

```
const Auction = artifacts.require("Auction");
const truffleAssert = require('truffle-assertions');

contract("Auction unit test 4", async accounts => {
  it("withdraw() should not fail if msg.sender is not the highestBidder", async () => {
    const instance = await Auction.deployed();
    const Balance1 = await web3.eth.getBalance(instance.address);
    assert.equal(Balance1, 0);
    await instance.bid({value:5000000000000000000, from:accounts[3]});
    const Balance2 = await web3.eth.getBalance(instance.address);
    assert.equal(Balance2, 5000000000000000000);
    await instance.bid({value:10000000000000000000, from:accounts[4]});
    const Balance3 = await web3.eth.getBalance(instance.address);
    assert.equal(Balance3, 15000000000000000000);
    await instance.withdraw({from:accounts[3]});
    const Balance4 = await web3.eth.getBalance(instance.address);
    assert.equal(Balance4, 10000000000000000000);
  });
});
```

Αποτελέσματα

Τα αποτελέσματα που λαμβάνονται από τον έλεγχο των παραπάνω unit tests είναι:

```
vasia@vasia-HP-Pavilion-15-Notebook-PC:~/Auction/test$ truffle test
Using network 'development'.

Compiling your contracts...
=====
> Everything is up to date, there is nothing to compile.

Contract: Auction unit test 1
  ✓ withdraw() should fail if msg.sender is the highestBidder (729ms)

Contract: Auction unit test 2
  ✓ HighestBid should get updated after a new bid (71ms)

Contract: Auction unit test 3
  ✓ Contract's balance should get updated after a new bid (74ms)

Contract: Auction unit test 4
  ✓ withdraw() should not fail if msg.sender is not the highestBidder (157ms)

4 passing (1s)
```