



پروژه ی نرم افزار دوره کارشناسی

گزارش شماره ۹ - پیاده سازی پروژه (بخش پنجم)

آریا رادمهر - ۹۷۴۶۳۱۲۵

دکتر سجاد حق زاد کلیدبری

March 2, 2022

بخش پنجم (کیف پول، کلید ها و تراکنش ها)

فایل index.test.js

در فایل index.test.js شروع به نوشتن تست هایی میکنیم که قرار است وضعیت یک کیف پول (wallet) را در هر سناریویی شرح دهند.

Describe -> Wallet

it -> has a balance

it -> has a public key

Describe -> signing data

it -> verifies a signature

it -> does not verify an invalid signature

Describe -> createTransaction ()

Describe -> and the amount is more than balance

it -> throws an error

Describe -> and the amount is valid

it -> creates an instance of `Transaction`

it -> matches the transaction input with the wallet

it -> outputs the amount the recipient

فایل index.js

سپس در فایل index.js شروع به نوشتن کد میکنیم به گونه ای که تست های نوشته شده در فایل index.test.js را با موفقیت پشت سر بگذارند.

Class Wallet {

constructor ()

```
    sing ( data )  
    createTransaction({ recipient, amount })  
}
```

transaction.test.js فایل

در فایل transaction.test.js نیز سناریو های یک تراکنش را مورد بررسی قرار میدهیم. این سناریو ها همچنین شامل چک کردن اعتبار تراکنش ها و همچنین بروز رسانی تراکنش ها بعد ثبت یک یا چند تراکنش میباشد.

Describe -> Transaction

it -> has an `id`

Describe -> outputMap

it -> has an `outputMap`

it -> outputs the amount to the recipient

it -> outputs the remaining balance for the `senderWallet`

Describe -> input

it -> has an `input`

it -> has a `timestamp` in the input

it -> sets the `amount` to the `senderWallet` balance

it -> sets the `address` to the `senderWallet` public key

it -> signs the input with the `senderWallet`

Describe -> validTransaction ()

Describe -> when transaction is valid

it -> returns true

Describe -> when transaction is not valid

Describe -> and a transaction outputMap value is not valid

it -> returns false and logs an error

Describe -> and a transaction input signature is not valid

it -> returns false and logs an error

Describe -> update ()

Describe -> and the amount is invalid

it -> throws an error

Describe -> and the amount is valid

it -> outputs the amount to the next recipient

it -> subtracts the amount from the original sender output amount

it -> maintains a total that matches the input amount

it -> re-signs the transaction

Describe -> and another update for the same recipient

it -> adds to the recipient amount

it -> should subtracts the amount from the original sender output amount

transaction.js فایل

در فایل transaction.js نیز یک متد سازنده به همراه ساختار اصلی تراکنش ها (ساخت خروجی، ورودی، بروز رسانی و چک کردن هر یک از تراکنش ها) میباشد.

Class Transaction {

constructor (senderWallet, recipient, amount)

createOutputMap ({ senderWallet, recipient, amount })

createInput ({ senderWallet, outputMap })

update ({ senderWallet, recipient, amount })

static validTransaction (transaction)

}