



## پروژه ی نرم افزار دوره کارشناسی

گزارش شماره ۹ - پیاده سازی پروژه ( بخش اول )

آریا رادمهر - ۹۷۴۶۳۱۲۵

دکتر سجاد حق زاد کلیدبری

February 11, 2022

## تکنولوژی های بکار رفته:

### ۱- جاوا اسکریپت ( JavaScript )

جاوا اسکریپت نوعی زبان برنامه نویسی سطح بالا است که دارای ویژگی هایی از جمله کامپایل درجا، چند الگویی، چند پارادایمی و سرعت بالا است. این زبان در کنار HTML و CSS، یکی از فن آوری های هسته ای دنیای وب است. بدلیل استفاده کتابخانه های معتبر این زبان در دیگر بخش های پروژه از جمله قسمت های رابط کاربری ( User Interface ) از این زبان به عنوان زبان اصلی پروژه استفاده شده است.

### ۲- کتابخانه ری اکت ( React )

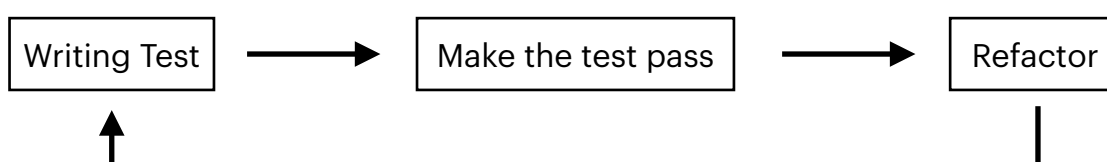
React یک کتابخانه front-end برای زبان JavaScript میباشد. این کتابخانه ی رایگان و متن باز برای ساخت رابط های کاربری بر اساس اجزای UI است. ری اکت هم اکنون توسط شرکت متا ( Meta ) و جامعه ای از توسعه دهندگان و شرکت ها نگهداری می شود. از این کتابخانه برای نمایش خروجی ها و رابط های کاربری پروژه استفاده شده است.

### ۳- چارچوب جست ( Jest )

Jest یک چارچوب ( Framework ) آزمایشی جاوا اسکریپت است که توسط شرکت متا ( Meta ) نگهداری می شود و توسط سازندگان با تمرکز بر سادگی و پشتیبانی از برنامه های کاربردی دنیای وب طراحی و ساخته شده است. از این چارچوب برای تست سلامت کد ها و پیشبرد صحیح هر بخش پروژه استفاده شده است.

## فرآیند توسعه ( Software Development )

فرآیند توسعه این پروژه با رویکرد Test Driven Development یا به اختصار TDD است. در این نوع فرآیند توسعه نرم افزار، ابتدا تست هایی برای کد های اصلی برنامه ( که در ابتدا بدون هیچ محتوایی هستند ) نوشته میشوند. سپس به تدریج سعی میشود تا کد ها بصورتی تکمیل شوند که در انتها تمامی تست ها را با موفقیت به پایان برسانند.



از ویژگی های مثبت این نوع رویکرد توسعه میتوان به موارد زیر اشاره کرد.

- ۱- طراحی بهتر برنامه و کیفیت بالاتر کد ها.
- ۲- ساخت مستندات پروژه به صورت دقیق تر.
- ۳- کاهش زمان مورد نیاز برای توسعه پروژه.
- ۴- انعطاف پذیری کد و نگهداری آسان تر.
- ۵- قابلیت اعتماد و اطمینان بالاتر به کد و نرم افزار نهایی.
- ۶- صرفه جویی در هزینه های پروژه در طولانی مدت.

## پارادایم برنامه نویسی

پارادایم برنامه نویسی در این پروژه ، شی گرای ( Object Oriented Programming ) یا به اختصار OOP میباشد. در این پارادایم با تمرکز به افزار کد های با ویژگی مشترک به شی ها سعی بر توسعه برنامه و ساخت پروژه میشود.

## ساخت بخش اول ( بلاک ها )

### فایل Block.test.js

در بخش اول بعد از نصب تکنولوژی های لازم، شروع به ساخت فایل Block.test.js میکنیم. این فایل در اصل یک فایل تست بر محوریت چارچوب Jest برای فایل Block.js میباشد. در Block.test.js ویژگی های یک بلاک را توصیف کرده و آنرا پیاده سازی میکنیم.

#### **Describe -> Block**

it -> has a timestamp, lastHash, hash, and data property

#### **Describe -> Genesis**

it -> returns a Block instance

it -> returns the genesis data of genesis block

#### **Describe -> mineBlock**

it -> returns a Block instance

it -> sets the `lastHash` to be the `hash` of the last block

it -> sets the `data`

it -> sets a `timestamp`

it -> creates a SHA-256 `hash` based on the proper inputs

it -> sets a `hash` that matches the difficulty criteria

### فایل Block.js

فایل Block.js برای ساخت یک بلاک میباشد. رویکرد این بلاک همانطور که در قبل نیز گفته شد، ساخت اسکلت کلی بلاک ها در بلاکچین است، به طوری که دارای موارد زیر باشد. در فایل Block.js یک کلاس با نام Block ایجاد میکنیم و متد constructor آنرا که مقادیر Elements در غالب یک شی ( object ) میگیرد، پیاده سازی میکنیم.

Class Block ( )

constructor ( { } )

static genesis ( )

static mineBlock ( )

### config.js فایل

همانطور که قبلا گفته شد، در ابتدا برای شروع کار با یک بلاک، نیاز به بلاک اولیه ( genesis ) داریم. در فایل config.js شروع به پیاده سازی یک بلاک اولیه میکنیم. از این بلاک در آینده برای ساخت اولین بلاک از بلاکچین استفاده میشود.

```
Const GENESIS_DATA = { Timestamp , lastHash , data , hash }
```