Assignment 3 - Jonathan Healy - IPFS (Interplanetary File System) Introduction

Introduction:

I have chosen to look at IPFS for my final project. According to ipfs.io, IPFS is, "A peer-to-peer hypermedia protocol to make the web faster, safer, and more open." Scrolling further down the page is a large sentence that says, "IPFS aims to replace HTTP and build a better web for all of us." Some of these statements have a 'too good to be true' type of ring to them but apparently IPFS is real tech.

I am going to copy/ paste the abstract to the IPFS white paper because it's interesting and I don't have the technical know-how yet to explain it all in my own words:

The InterPlanetary File System (IPFS) is a peer-to-peer distributed file system that seeks to connect all computing devices with the same system of files. In some ways, IPFS is similar to the Web, but IPFS could be seen as a single BitTorrent swarm, exchanging objects within one Git repository. In other words, IPFS provides a high through- put content-addressed block storage model, with content- addressed hyper links. This forms a generalized Merkle DAG, a data structure upon which one can build versioned file systems, blockchains, and even a Permanent Web. IPFS combines a distributed hashtable, an incentivized block exchange, and a self-certifying namespace. IPFS has no single point of failure, and nodes do not need to trust each other.

Key concepts that are coming up here: 'peer-to-peer distributed file system', 'BitTorrent', 'Git repository', 'Merkle DAG', 'DHT'. Because these are all things that I really want to learn more about, I am going to be taking a deep dive into IPFS. For my project I will be looking at some of these major concepts in more depth. I also earned my Ethereum Developer's Certificate from Consensys this past summer and IPFS potentially solves a huge problem associated with using the Ethereum blockchain.

Ethereum works great for smart contracts and keeping track of digital assets but the cost to store anything more than a few kilobytes in size is just way too expensive. If you want to use Ethereum to digitize and secure land purchase agreements in an untrusted environment, for example, you need something like IPFS to actually store the agreements themselves (there are places in the world where corrupt officials will steal land titles from people). You would store the actual transaction between two people on the blockchain along with the link to the hash on IPFS that points to where the agreement is located in the peer to peer file system.

The Code:

The purpose of this program is to: 1) Store a file directly on the ipfs network and to 2) Store the ipfs link/ hash that we get from the first part on the Ethereum blockchain to have a secure record of submitting the file. Part of the reason that someone would want to do this is that it is prohibitively expensive to store data on the Ethereum blockchain.

Testing: The only way to test this program is to run through all the steps below manually.

1. Upload your file through the app you and check to see your transaction on etherscan.org.
2. Install IPFS from the last step and retrieve your file using the IPFS Hash #

Note: The code for this assignment comes from this tutorial that is available online:

https://itnext.io/build-a-simple-ethereum-interplanetary-file-system-ipfs-react-js-dapp-23ff4914ce4e

Run metamask:

1. Install metamask in your browser: https://metamask.io/

2. Open metamask. Choose the Rinkeby Test Network option.

3. You can use the seed phrase from the file: rinkeby.txt in metamask to log in to the Rinkeby test network. (Don't steal all my test eth!)

Run React app:

These are the basic instructions from the tutorial on how to run the react app:

npm i create-react-app
npm install react-bootstrap
npm install fs-extra
npm install ipfs-api
npm install web3@^1.0.0-beta.26

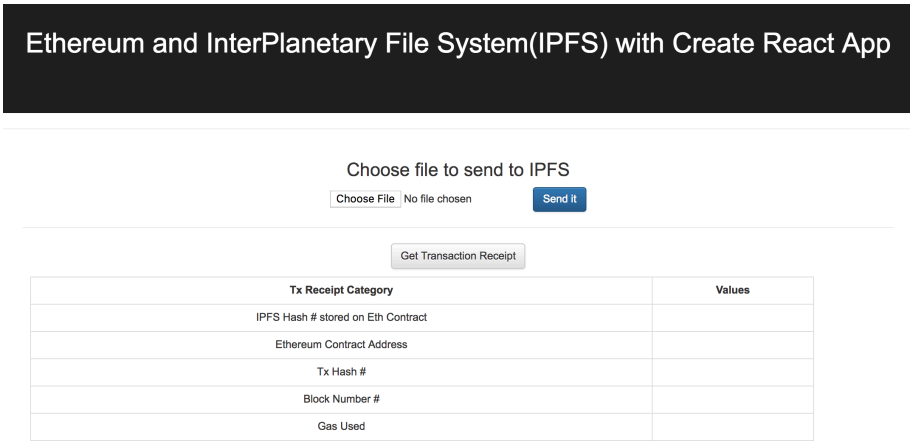
Note: if you have not used create-react-app until now, you may have to install it globally first

1. sudo npm install -g create-react-app
or npm install -g create-react-app

2. create-react-app eth-ipfs

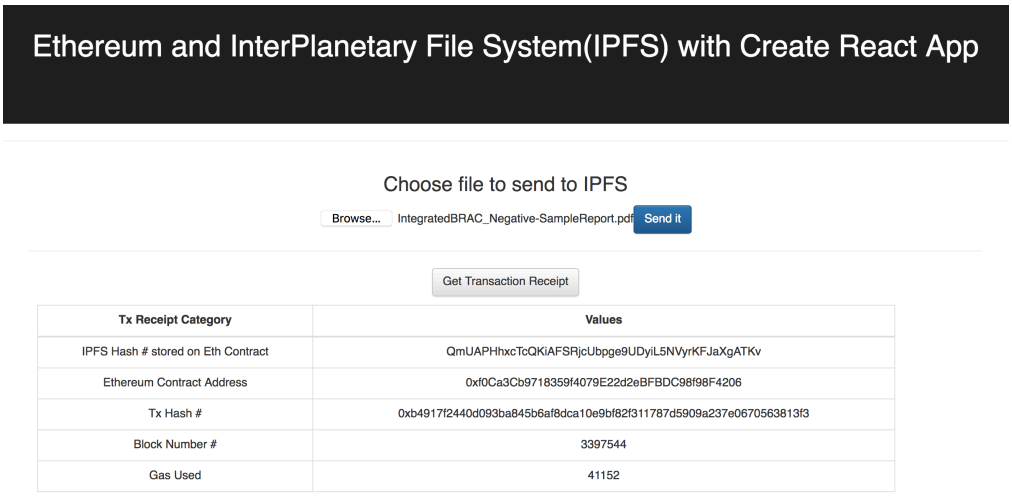
3. cd into eth-ipfs and "npm start"

This is what the app looks like when it first starts:



Upload a file to IPFS:

This part is self explanatory and should go smoothly as long as you are logged into the Rinkeby Ethereum test network in Metamask. You will have to approve a transaction in Metamask to move to the next part.



Tx Receipt Category	Values
IPFS Hash # stored on Eth Contract	QmUAPHhxcTcQKiAFSRjcUbpge9UDyiL5NVyrKFJaXgATKv
Ethereum Contract Address	0xf0Ca3Cb9718359f4079E22d2eBFBDC98f98F4206
Tx Hash #	0xb4917f2440d093ba845b6af8dca10e9bf82f311787d5909a237e0670563813f3
Block Number #	3397544
Gas Used	41152

Note the IPFS hash #. We can look this up later to retrieve our file. We can also search etherscan.org with the Ethereum Contract Address to get this hash # and view our contract.

Run IPFS to retrieve our file from the network:

The instructions for this part can be found at: https://docs.ipfs.io/introduction/usage/

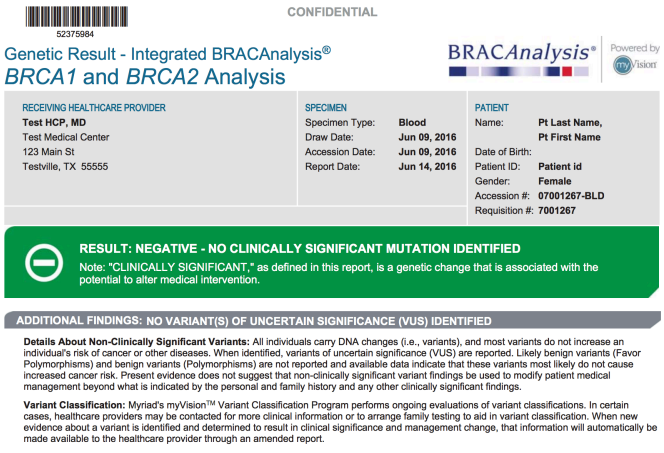
To install ipfs: https://docs.ipfs.io/introduction/install/

Start the daemon: ipfs daemon

To download our file off of ipfs:

ipfs cat QmUAPHhxcTcQKiAFSRjcUbpge9UDyiL5NVyrKFJaXgATKv >demo.pdf

Voila! (This is not someone's real medical file).



52375984

CONFIDENTIAL

Genetic Result - Integrated BRACAnalysis®
BRCA1 and *BRCA2* Analysis

BRACAnalysis® Powered by myVision

RECEIVING HEALTHCARE PROVIDER
Test HCP, MD
Test Medical Center
123 Main St
Testville, TX 55555

SPECIMEN
Specimen Type: **Blood**
Draw Date: **Jun 09, 2016**
Accession Date: **Jun 09, 2016**
Report Date: **Jun 14, 2016**

PATIENT
Name: **Pt Last Name, Pt First Name**
Date of Birth:
Patient ID: **Patient id**
Gender: **Female**
Accession #: **07001267-BLD**
Requisition #: **7001267**

RESULT: NEGATIVE - NO CLINICALLY SIGNIFICANT MUTATION IDENTIFIED
Note: "CLINICALLY SIGNIFICANT," as defined in this report, is a genetic change that is associated with the potential to alter medical intervention.

ADDITIONAL FINDINGS: NO VARIANT(S) OF UNCERTAIN SIGNIFICANCE (VUS) IDENTIFIED

Details About Non-Clinically Significant Variants: All individuals carry DNA changes (i.e., variants), and most variants do not increase an individual's risk of cancer or other diseases. When identified, variants of uncertain significance (VUS) are reported. Likely benign variants (Favor Polymorphisms) and benign variants (Polymorphisms) are not reported and available data indicate that these variants most likely do not cause increased cancer risk. Present evidence does not suggest that non-clinically significant variant findings be used to modify patient medical management beyond what is indicated by the personal and family history and any other clinically significant findings.

Variant Classification: Myriad's myVision™ Variant Classification Program performs ongoing evaluations of variant classifications. In certain cases, healthcare providers may be contacted for more clinical information or to arrange family testing to aid in variant classification. When new evidence about a variant is identified and determined to result in clinical significance and management change, that information will automatically be made available to the healthcare provider through an amended report.