Anish Malladi
Kevin Yao
Ph360 Final Paper
Prof. Anita Raja

The IPO Blockchain

Introduction

Companies finance themselves in different ways. While start-ups tend to gravitate towards angel investors and venture capitalists, more mature companies have historically tended to choose to exchange equity for financial capital. Raising money through equity sale has traditionally been done for a very long time. The original system of auctioning off equity was used extensively by the Dutch. This system is more commonly known today as an IPO, or an Initial Public Offering. Another colloquial terms for an IPO is "going public" because an IPO allows company stock to be traded publicly on the open market. Unfortunately, over time, the process has become more expensive, opaquer, and less accessible. These factors indicate that an IPO is an ideal case for implementing a blockchain based method, and this study provides a proof-of-concept application that implements the fundamental components of the application.

Use Case

Background

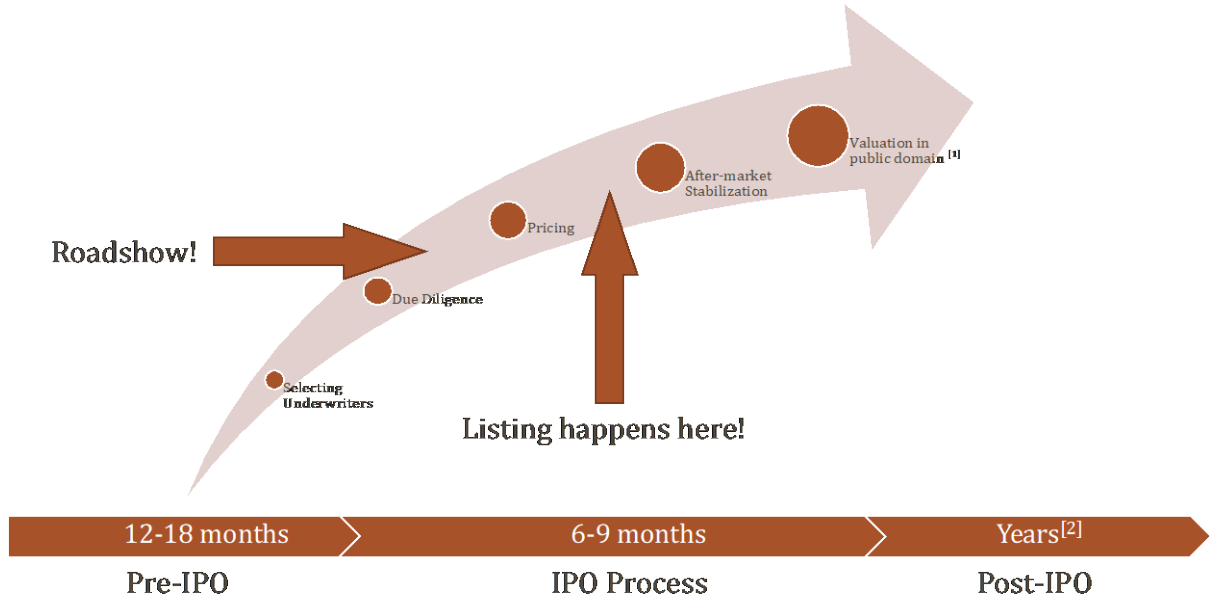
Traditional methods of financing are IPOs, loans from banks, or through equity sale to private firms, known as private equity. Of these methods, IPOs are the only method that allow for the owners of the company to retain some amount of control over their own firm. Loans from banks invariably require using a portion of the company as collateral. Private equity firms, on the other hand, prefer to buy out large amounts of the company, preferring to hold majority stake so as to implement better management methods. As such, raising capital comes at the cost of being cut out of the business, which is not ideal for many business owners.

The recent, disruptive methods in the market were expected to be wildly successful. Specifically, the methods are crowdfunding and ICOs or Initial Coin Offerings. Crowdfunding platforms like Kickstarter were intended to be sources of finance in exchange for a future commodity, however the platforms involved placing orders before manufacturing was committed to. As a result, delays and sometimes cancellation of orders led to a general sense of disappointment in the system, which further resulted in a lack of trust in crowdfunding as a viable platform. ICOs, on the other hand, are recent, only one or two years old. The system involves raising money in exchange for cryptocurrency "coins" and "tokens". While this may seem progressive and incredibly useful, the fundamental problem with such a system is the lack of a backing

for the token. The token is a simple representational form of currency, and there is no true commodity exchange. Several scams have already been conducted in this fashion, and the lack of trust between investor and company is detrimental towards the company's future and prospects in the market.

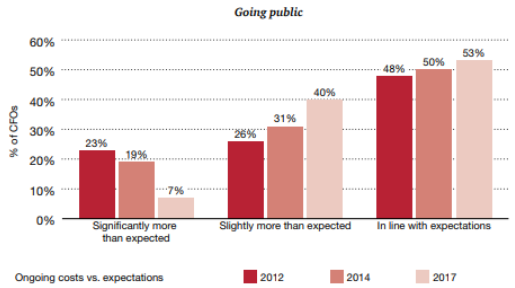
Problem Statement

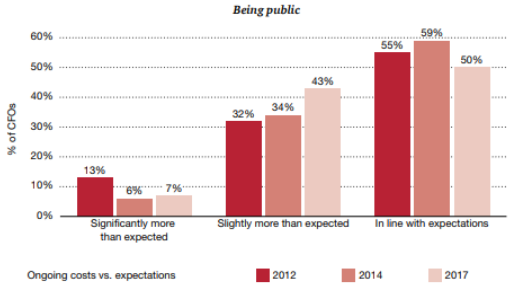
Clearly, a new disruptive, and yet trust-based method is required for allowing companies to raise capital. An IPO is a prime use-case for implementing a blockchain based solution. In order to examine where the greatest amount of impact can be made by a blockchain, it was necessary to examine the IPO process as a whole. The flowchart below illustrates the IPO process as a whole:



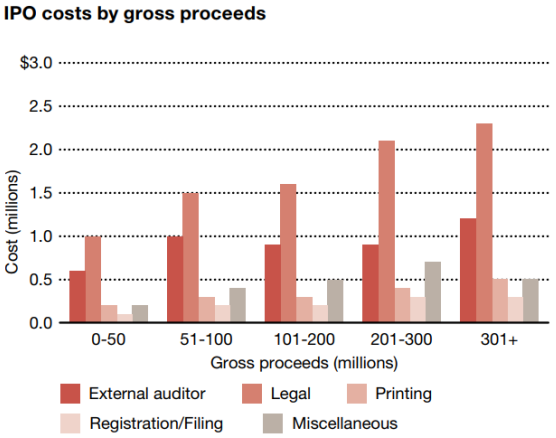
The first three stages, selecting underwriters, due diligence, pricing, and the roadshow and initial sale events, are parts of the traditional IPO process. It is important to consider the stabilisation of the system and the public valuation of the company as parts of the process, even if they fall within the phase of a company's lifecycle commonly referred to as "being public".

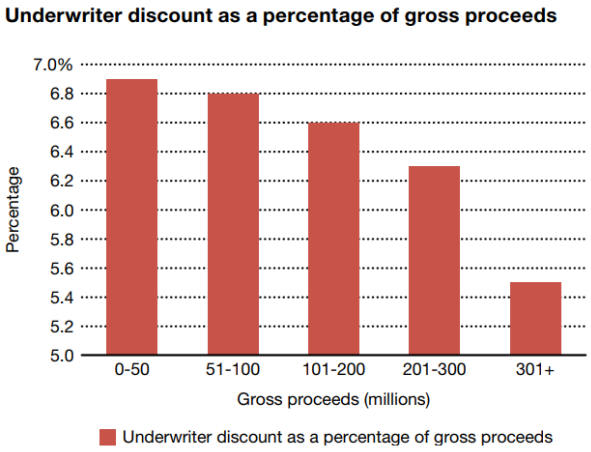
The primary objective of a blockchain solution would be to reduce the costs associated with this process. An auditing firm, pricewaterhousecooper, conducted a study on the costs of going and being public, with an extensive analysis as to which factors were major cost drivers. The study found that almost 50% of CFOs (Chief Financial Officers) anticipated costs that were significantly, or somewhat less that the actual burden of the IPO. These graphs are seen below.





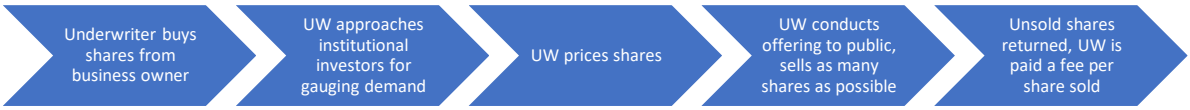
The cost breakdown of the IPO can also be assessed as a percentage of the proceeds of the IPO. The reason that costs are expressed this way is primarily attributed to the fact that fees paid to the underwriter are expressed as a percentage of the total sale. The charts below indicate these costs in detail.





On average, underwriter costs account for around 7% of the value of the IPO. An underwriter is a specific set of employees of an investment bank who are well versed with IPOs. The underwriter as an entity takes on the risk of the IPO - signing an engagement letter that specifies a fee for buying the shares of the company at the beginning of the process, conducting the IPO, and then returning any unsold shares to the company. The underwriter charges a fee regardless of the outcome of the IPO to cover their costs, usually with a hefty margin attached. In the event that the IPO fails, the shares are either returned to the company, or the underwriter becomes an investor who presents the company with a certain amount of capital. Essentially, by taking on the risk of the process, the underwriters can charge high fees. The other fees, namely the legal, printing and accounting fees are indirectly related to the investment bank serving as underwriter. The bank handles the legal paperwork, given that they have a predefined network with the governmental organisations involved as well as the relevant auditors.

The primary reason that this transferral of risk is accepted is because of the vast network of investors that the banks have built over time. The banks need not approach any investor “cold”, since they have predefined relationships which they tap in order to sell the shares and profit. Since private institutional investors form the vast majority of the stakeholders in the system, this system yields benefits, both in terms of conducting a successful IPO, and allowing the banks to make a profit in the process.



Reducing the amount of risk burdened by the underwriter is a possible way of reducing the costs associated with the system. Further, reducing the reliance on the underwriter (as seen in the chart above) to implement legal framework allows the companies to reduce paid fees. By conducting the sale via auction on a blockchain and enforcing the legal documents and such as immutable transactions on a blockchain, the integrity of the process is conserved, costs are reduced, and as an added advantage, transparency is more easily managed.

Framework and Methods

General framework details

Hyperledger is an open source collaboration effort to implement blockchain frameworks for researchers and entrepreneurs to use. The blockchain solution for the IPO process covered in this project was implemented using the Hyperledger frameworks - specifically Hyperledger Fabric and Hyperledger Composer. Hyperledger Fabric is a lower level blockchain framework application whose specific implementation allows users to easily utilize components such as consensus, membership services, and smart contracts. The smart contracts are implemented using "chaincode", which can be understood as the underlying logic of the system. Hyperledger Composer is a higher-level framework which uses Hyperledger Fabric. It is a set of tools for building smart contracts and applications, and also features the ability to create reusable blockchain models, generate REST APIs based on the defined blockchain model, and test in web-based, interactive IDE with deployment capabilities.

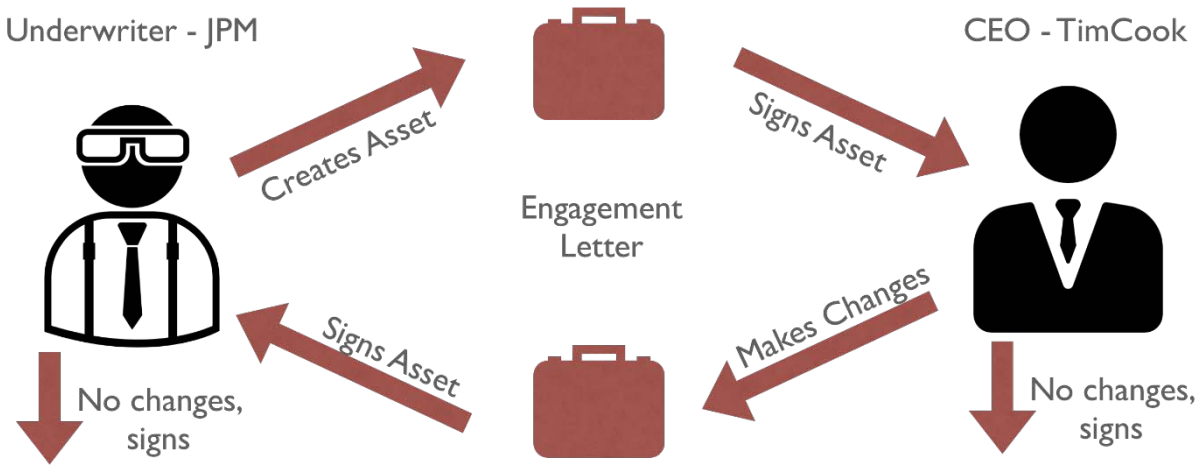
This project uses Hyperledger Composer because of several reasons. Its ability to quickly develop blockchain applications without requiring low level implementation of smart contracts, etc, is useful since the project is simply a prototype. Many ideas were implemented in the beginning, only for most of them to fail. In order to quickly implement solutions and test these solutions, it was important to have a well-established framework which had stable foundations and useful features. Composer is also known for its robustness and heavy documentation, which comes very useful when implementing and testing various aspects of the blockchain logic. In order to adequately develop reasonable solutions, knowledge of the underlying framework needed to be understood. This is accomplished easily through the Composer documentation. Furthermore, it aids in development of Blockchain concepts, which is useful for users who are relatively new to Blockchain technology.

Implementation

Using the Hyperledger frameworks, key aspects of the IPO process were modelled as various assets and transactions objects. Primary entities involved in the IPO process were modelled as participant objects. This solution implementation can be broken into two specific parts: the "lifecycle" phase and the auction phase.

Lifecycle Phase

The lifecycle phase involves the transactions that govern the transfer of legal documentation among entities. To illustrate this, two sample documents were implemented. These were the engagement letter and the registration certificate. The engagement letter is a document signifying the engagement of the bank as the company's underwriter, with an agreement to pay the underwriter a certain percentage of returns. A workflow for the transaction can be seen below:



The engagementLetter is an asset, that must be created prior to invoking the transaction call for the ELetterTx transaction. A sample call is shown below:

```
{
  "$class": "org.ipoblockchain.Transactions.ELetterTx",
  "el": "resource:org.ipoblockchain.Assets.engagementLetter#EL-1",
  "underwriter": "resource:org.ipoblockchain.Contractor.Underwriter#JPM",
  "ceo": "resource:org.ipoblockchain.Business.CEO#TimCook",
  "transactionID": "",
  "participantSig": "",
  "newHash": ""
}
```

The asset, when created includes a letterHash attribute. This attribute is crosschecked with the newHash transaction attribute. If the two match, then the system assigns the participantSig to the relevant attribute in the asset. If the two hashes do not match, then the system erases any signatures on the asset, updates the letterHash

attribute to the newHash, and then assigns the participantSig to the relevant attribute in the asset, depending on the current participant who is submitting the transaction.

This system was extended to three users in the registration certificate. The registration certificate is a document filed by the company with the Securities and Exchange Commission (SEC) which details the stock value, includes a complete filing of risk and so on. This document needs to be approved by the SEC before the company is allowed to continue in the IPO process. The transaction call is shown below.

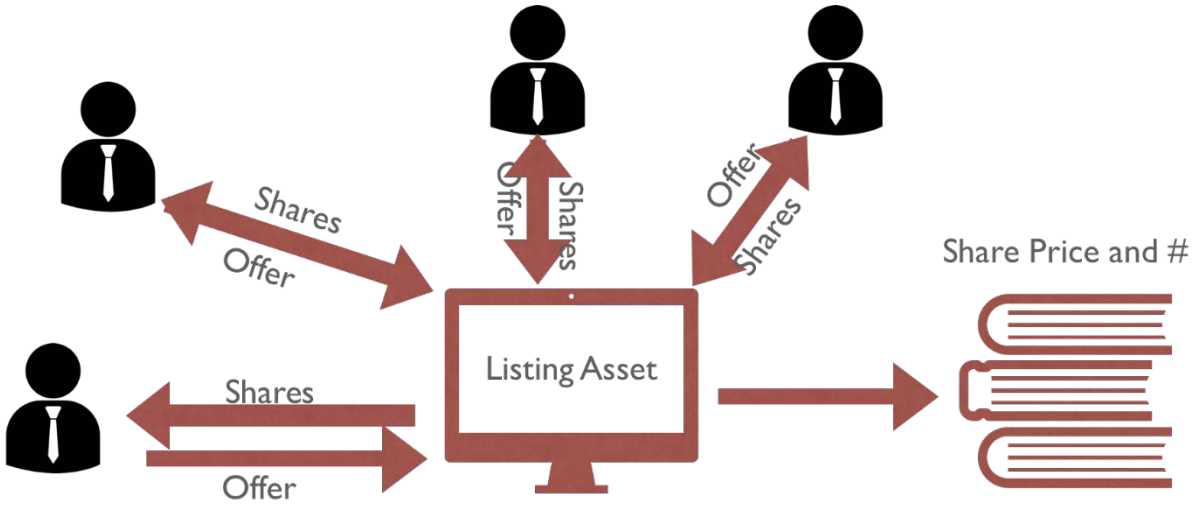
```
{
  "$class": "org.ipoblockchain.Transactions.RCertificateTx",
  "rc": "resource:org.ipoblockchain.Assets.registrationCertificate#4758",
  "underwriter": "resource:org.ipoblockchain.Contractor.Underwriter#8141",
  "ceo": "resource:org.ipoblockchain.Business.CEO#2561",
  "sec": "resource:org.ipoblockchain.Gov.SEC#7452",
  "transactionID": "",
  "participantSig": "",
  "newHash": ""
}
```

This transaction is incredibly similar to the ELetterTx transaction, except that the function is wrapped in another conditional to include the SEC in the system. Other than that, it follows the same logic as ELetterTx, erasing the signatures in case of conflict in certificateHash and transferring the owner over at each stage.

Auction Phase

In order to further reduce company costs, the blockchain application can determine its final share price by using a Dutch auction listing. The idea is to automate processes which are costly, and the reason for this is twofold. In typical IPO processes, the underwriter is compensated enormously primarily due to the amount of risk it holds. The underwriter must first accumulate all the shares which the company is planning to sell, and then offer allocations of the shares to various investors. This solution renders this responsibility to be no longer necessary. Instead of requiring the underwriter to hold the total shares being sold, the shares being sold are stored in the blockchain.

Additionally, during this phase, the blockchain application automatically determines the final, optimal share price of the company's shares. In essence, an evaluation of the company is determined algorithmically. This is no longer an issue for underwriters -- thus, the high compensation for underwriters is slowly being chipped away. A representation of the way the auction works is seen below.



The auction phase is enacted when a company creates an auction listing asset object on the blockchain. This is generally through the CEO of the company, so the ability to perform this action is specific to CEO participants. A sample listing asset which contains data fields is shown below:

```
{
  "$class": "org.ipoblockchain.Auction.Listing",
  "listingID": "LIST1",
  "totalShares": 10000,
  "remainingShares": 10000,
  "capitalRaised": 0,
  "company": "resource:org.ipoblockchain.Business.Company#AAPL"
}
```

Here, the important attributes are the totalShares, remainingShares, and capitalRaised. totalShares holds a constant value based on the CEO's input to the listing asset creation and determines the number of shares that are planned to be sold to investors. These shares are, in effect, stored on the blockchain when the listing asset is created. Currently, the remainingShares is equal to the totalShares because none of the shares have been sold to investors. However, this is expected to change after investors have made offers, and the CEO has decided to allocate shares and determine his/her final share price. capitalRaised is an attribute which will hold the amount of capital raised from the shares sold to investors. This is essentially $(totalShares - remainingShares) * sharePrice$, where sharePrice is an attribute which holds the final share price after the auction.

As an investor participating in this network, listings are important. Investor participants in the network can see listing assets in the blockchain after they are created. Due to the decentralized nature of the blockchain network, these participants are automatically granted access to the contents of the listing, which can easily hold company facts, statistics, and other information relevant to investors.

At this point, there are several caveats to the blockchain application that should be mentioned. First, this auction can only work if an adequate network of investors is created. Currently, underwriters have held the responsibility of maintaining a network of relationships to investors (e.g. BlackRock). This responsibility is to be replaced by the blockchain network, which should comprise of many investors, as well as company CEOs, underwriters, lawyers, etc. Therefore, for this blockchain application to be utilized in production, a network of investors should first be obtained. If this network of investors is collected properly, the responsibility of the underwriter as the primary hub to the investor network is no longer required. The role, and thus compensation, of the underwriter is thereby reduced.

Given an adequate investor network, listings should be met with an influx of offers, or bids, for company shares. Once an investor has decided on what it values a company and the amount at which it is planning to invest, it can make an offer. For this particular implementation, the offers are formatted loosely according to a traditional Dutch auction.

An investor will make an offer which states the total amount of capital it is willing to invest, and the minimum number of shares at which it agrees to receive in return for its offer. Here, an investor is essentially creating demand for the company's shares and an evaluation of the company's worth -- this is information which will be used later to calculate the final share price based on the total demand and each evaluation. A sample offer transaction for 2,000 shares at $20/share is shown below:

```
{
  "$class": "org.ipoblockchain.Auction.Offer",
  "bid": {
    "$class": "org.ipoblockchain.Auction.Bid",
    "price": 20,
    "shares": 2000
  },
  "listing": "resource:org.ipoblockchain.Auction.Listing#LIST1",
  "investor": "resource:org.ipoblockchain.Investor.privInvestor#4887"
}
```

Here, it is important to see the benefits of this solution in this aspect. Investors here are limited to neither private nor public investors -- rather, both are participants in the network with almost exact same rights. Instead of limiting initial offerings to large institutional investors, this blockchain application opens initial offerings to the public as well. This allows the company to obtain a larger understanding of the overall market demand for its company's shares, and therefore it can (automatically) price its shares at an efficient share price.

Once many offers have been made for the company's listing, the company can decide to close the auction by creating a closeBidding transaction. This transaction, based on the Dutch auction algorithm, calculates the final share price and allocates shares appropriately to the investors.

The Dutch auction algorithm utilizes the elasticity model of supply and demand. Given a pool of offers, the algorithm should calculate the optimal share price, which is the lowest price at which all the shares will be sold. Therefore, it is not guaranteed (in fact, very unlikely) that all the investors will receive an allocation of shares. The investors who will receive an allocation of shares will receive and adjusted number of shares based on the number of shares which it designated in the offer. The adjusted number of shares is based on the following formula:

$$elasticity = \frac{percentage\ change\ in\ quantity\ demanded}{percentage\ change\ in\ price}$$

$$elasticity = \frac{\frac{\Delta Q}{Q}}{\frac{\Delta P}{P}} = \frac{P\Delta Q}{Q\Delta P}$$

Where P is the share price, Q is the quantity of shares demanded at price P, and Δ represents a change in the relevant quantity.

An assumption made here was that the elasticity was always 1, no matter the share price. This is not always true, however, adjusting the elasticity would require another data table which severely overcomplicates the algorithm. Given that the development was targeted specifically at implementing the system with a high-level understanding, an elasticity was assumed, which further allowed the following relationship:

$$percent\ change\ in\ Q\ =\ percentage\ change\ in\ P$$

$$Q'\ =\ Q\ +\Delta Q$$

$$Q' = Q\left(1 + \frac{\Delta P}{P}\right)$$

Thus, given a single bid, it was possible to ascertain the number of shares the person would demand at any given price. Using the elastically model, adjusted shares are allocated to investors with the following algorithm:

Algorithm 1 Dutch Auction Optimal Price Algorithm

Require: $list_{offers}, listing$

procedure GETOPTIMALPRICE($list_{offers}$, $Listing$)

$price_{old}, shares_{old} \leftarrow null, null$

for $offer_i$ in SORTED($list_{offers}$) **do**

$(price, shares, investor) \leftarrow offer_i$

$adjustedShares \leftarrow$ GETADJUSTEDSHARES($price_{old}, price, shares_{cum}$)

$shares_{cum} \leftarrow shares_{cum} + adjustedShares$

$list_{investors}$.APPEND($investor$)

if $shares_{cum} > listing$.GET($totalShares$) **then**

break

$price_{old} \leftarrow price$

$price_{optimal} \leftarrow price$

return $list_{investors}, price_{optimal}$

The closeBidding transaction can only be created by the CEO of the company associated with the listing. The following is a simple transaction call:

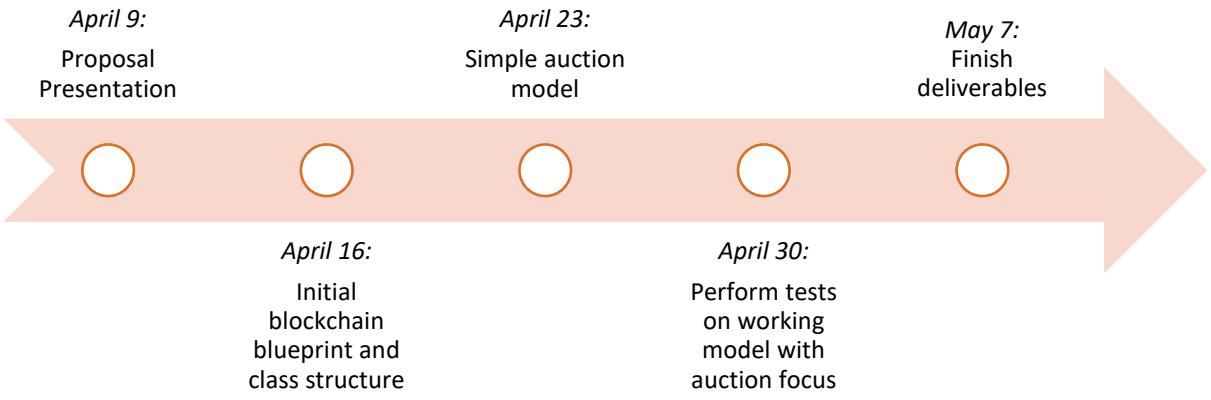
```
{
  "$class": "org.ipoblockchain.Auction.CloseBidding",
  "listing": "resource:org.ipoblockchain.Auction.Listing#LIST1"
}
```

The investors will receive their allocated, and adjusted, number of shares after the closeBidding transaction. Additionally, a final share price will appear as an attribute in the listing asset, and the capitalRaised attribute should no longer be 0.

Project roles

Both Kevin and Anish worked on model class definitions (e.g. .cto files) and setting up the underlying blueprint for the application. The lifecycle portion of the application was developed by both Kevin and Anish. From thereon, Anish set up the underlying Hyperledger Framework infrastructure for deployment, testing, and debugging, as well as permission declarations for the various participants in the network. Kevin worked specifically on the auction phase, which mainly consisted on writing a relevant and useful Dutch auction algorithm implementation.

Timeline



Successes and Challenges

At the outset of the project, the biggest expected challenge was working with a new framework with which we had no familiarity. Hyperledger's extensive documentation was quite useful in diagnosing and rectifying the errors in the framework itself. The actual development was done in Composer Playground - an IDE that allowed for quick development and testing.

The biggest challenge we faced in the process was the creation of the front-end UI. Hyperledger Composer automatically generates a REST server, and the associated API to connect to it. The tutorials indicate using a Yeomen generator to generate an Angular Application to use as a front end. However, we over defined our asset classes in order to allow for future functionality. Specifically, the engagementLetter asset contained a Boolean attribute syndicate, which was to be used to indicate the presence of multiple underwriters (known as a 'syndicate of underwriters') in the IPO. The Angular interface defaulted the Boolean attribute to null instead of false, which meant that the system was trying to read null into an initialised variable that could not be null. This caused the server to crash and the angular app to not work. It took a long time to debug this small error, and the fix we implemented was simply to remove the attribute in order to ensure that the rest of the front end was functional. As of the writing of this report, the assets can all be created and edited from the front end. While the app defines forms for submitting transactions, the actual UI interface for initialising the form and submitting it does not exist and needs to be implemented. We also recommend a workaround for Boolean variables which default to false, perhaps by using an enum{} definition, which uses radio buttons on the interface instead of a checkbox.

However, even with these challenges, the logic was fairly simple to implement in terms of understanding what needed to be done. We were able to design a fairly simple workflow, and that helped in establishing extremely clear underlying logic that did not require much editing to establish functionality.

Evaluation

This project's success is determined primarily by the functionality of the blockchain application. The two phases of the blockchain application should be implemented efficiently and correctly; this is evaluated using unit tests and trial demos to other users. Furthermore, the project is on the GitHub repository https://www.github.com/keyao21/IPOBlockchain, which is available to public and open to evaluation from other users.

Furthermore, the blockchain application required a multiagent aspect -- this is accomplished by learning and utilizing the Hyperledger Composer framework properly for distributed systems. Nodes can be initialized with various identities connected to participants in the network and can interact with the other nodes in the network based on their designated functionalities (i.e. investors create offer transactions that affect the listing asset in the Blockchain, and other participants in the network are able to see the offers). Additionally, an auction network was created with this multiagent aspect in mind.

Though a front end is still undergoing tests, the underlying angular skeleton is working. Further usage documentation will be written, but most of the usability is dependent on the user understanding the Hyperledger Composer framework through tutorials online. For simplicity, the project has several utility scripts to automatically set up parts of the application.

Though small aspects can still be cleaned up, the main objectives set from the start have been accomplished in this application. Through development of the application, possibilities for additional features were kept in mind so that contributions or future developments can be relatively easily made.

References

1. https://corporatefinanceinstitute.com/resources/knowledge/finance/ipo-process/
2. https://www.pwc.com/us/en/services/deals/library/cost-of-an-ipo.html
3. https://www.theipojournal.com/2016/03/2016-ipo-study-ipo-fees-and-expenses/
4. https://www.pwc.com/us/en/deals/publications/assets/cost-of-an-ipo.pdf
5. https://chainium.io/uploads/files/Chainium-Business-Whitepaper-v3.pdf
6. http://www.the-blockchain.com/2016/03/17/overstock-com-announces-historic-blockchain-public-offering/
7. Consensys, "Designing a good Blockchain Use Case". Moodle, Accessed 09.04.18
8. Hyperledger Composer - https://hyperledger.github.io/composer/latest/
9. Demos, Telis. (21 June 2012) What Is a Dutch Auction? – Deal Journal – WSJ. Blogs.wsj.com. Retrieved on 2012-10-16.
10. Hyperledger Fabric - https://hyperledger-fabric.readthedocs.io/en/release-1.1/