Open Freelance hub

CA326 Functional Specification

Dean Lynch & David Weir

Table of Contents

1. Introduction

1.1 Overview

This is an application to replace the middle-man from online freelance/contract work services by moving transactions onto the blockchain, providing a public, immutable ledger of past transactions to show a complete and unconcealable history of each party involved. It also gives small businesses and startups an alternative to hiring freelance workers, with an option to trade work with other small businesses and startups.

At the moment, the freelance and contractor workspace online is primarily dominated by private companies who host job postings and more importantly, they control the reviews of the users of their site. How do we know that the reviews shown by these companies are unaltered and that they are not cherry-picking the reviews they choose to show for certain users? For example, an employer threatens to move to an alternate service if a negative review isn’t removed, in an attempt to cover up their wrongdoing. Placing these transactions on a blockchain will provide a more open and honest marketplace for freelance job openings.

A unique aspect of this project would be the possibility for two parties to trade work/services with each other, which could benefit small businesses who are struggling with funding the services that they outsource. For example two small business owners, one specialising in Accounting and the other in building business plans, could swap book balancing in exchange for an analysis of the others business plan. In this case similarly, the blockchain would act as a public record of past exchanges, with reviews of the quality of work etc…

1.2 Business Context

The Open Freelance Hub could be deployed in competition alongside some of the private online freelance organisations that are in use in the real world today. This system would appeal to users of these sites, both employers and employees, as it would provide them with a more open and transparent marketplace to work in.

With the storage of transaction information and reviews in a public ledger, more power would be provided to the users of the site as they do not have to worry about the behind-the-scenes actions of a trusted third party. Small businesses and startups would also be attracted to this marketplace as it would allow them to trade work and services as opposed to paying for work entirely using the Ether cryptocurrency.

1.3 Scope

Our goal is to produce the minimum viable product (MVP) as soon as possible. The MVP in this case is a site that will allow two users to trade Ether with each others Ethereum accounts for some amount of work completed. Users should be able to browse job postings and apply to relevant jobs. These users should also be able to review each other: the most basic form of this will be a score out of five. As part of the MVP the information associated with each transaction will be stored on the Ethereum blockchain.

Once the MVP is created, we aim to develop the function of the site which will allow users to trade work with each other rather than with Ether alone. This feature may be out of scope as it may interfere with the development of basic site functionality.

The following are most likely out of scope, but would be good additions to this project. We hope to expand the review function by allowing users to provide a more detailed review of the other party. This is not part of the the MVP as it creates a difficult problem due to some users using profane language and the immutable nature of the blockchain. If all previous items are completed we will then aim to add some form of communication between users using the Ethereum whisper protocol. This is not part of the scope as the same functionality can be completed using existing Đapps.

1.4 Glossary

- **Blockchain**: A distributed digital public ledger that records transactions across multiple computers to prevent transaction history from being altered.
- **Smart Contracts**: Self-executing programmable contracts that allow for the transfer of currencies on the blockchain, and the updating of the public ledger.
- **Ethereum**: An open platform that allows for the development of decentralised applications.
- **Ether**: The common cryptocurrency used in the Ethereum environment.
- **Đapp**: An app consisting of a user interface and a decentralised backend built on the the Ethereum blockchain.
- **Gas Price**: The price paid to process a contract or transaction onto the Ethereum blockchain.
- **Solidity**: The language used to program smart contracts for the Ethereum blockchain.
- **web3.js**: A JavaScript API compatible with the Ethereum blockchain.
- **MetaMask**: A browser plugin for Chrome and Firefox which allows you to browse Đapps without running a full Ethereum Node.

2. General Description

2.1 Product/System Functions

The main focus of this project is to provide an open and completely honest platform for both employers and employees to buy and trade work contracts between one another. Employers will create job listings which workers can then apply to. Employees can build their skill profile, which employers can use to decide if the worker is a good fit for the position they are looking to fill. On top of this, both employer and employee will have a complete record of all previously fulfilled contracts and transactions, as well as a reviews of all the work provided by the opposite parties that the user has worked with.

Every transaction, contract and review created/completed will be directly associated with each account holder and saved onto the Ethereum blockchain. This will give the user peace of mind while reviewing potential employers/employees that every detail provided is genuine. What's unique about our service is that we plan to allow businesses to trade contracts with each other directly, only requiring each party to pay the maintenance fees. This feature is particularly aimed at small businesses who require urgent work, but may not have the necessary funds at the time.

2.2 User Characteristics

The prospective users for this site include:

- Any current users of similar sites who either wish to move to a more open marketplace or would like to start earning Ether for their work.
- Contractors who work primarily offline that are looking to modernise by moving their business online.
- Owners of small businesses or startups who outsource some of their work, these users in particular could benefit from the trading work service we hope to provide.

In order for users to be able to use this site correctly some basic knowledge of blockchain and Ethereum in particular will be needed, such as how Ether is transferred, the cost of gas/Ether, etc… Although the knowledge needed to perform transactions in Ether is minimised by the use of easy-to-use programs such as MetaMask for Chrome and others, which make it easier to transfer Ether between accounts.

2.3 Operational Scenarios

Scenario 1 - New User:

1. User enters site with connected Ethereum account.
2. User selects register account.
3. User chooses to create either an employer or freelancer account.
4. A freelance user will be asked to fill out a biography for their account, with experience, skills etc.. Employers will fill out their own employer information.

Scenario 2 - Employer Creates a Job:

1. User enters site with employer user account.
2. Employer selects create job.
3. Employer fills in description of job and tags it with appropriate skills.
4. The employer pays the necessary gas price and maintenance fee for posting
5. The job is live.

Scenario 3 - Employer Browsing Candidates:

1. Employer views their job listing.
2. Employer can search through all active candidates who are looking for employment based on the skills listed on their profile.
3. Employer can contact a freelancer to ask them to apply.
4. The employer will be provided with a list of all freelancers who have applied.

Scenario 4 - Freelancer Browsing Available Jobs:

1. User enters site with freelancer user account.
2. The landing page is displayed with all open jobs.
3. User can filter available jobs by the skills that the jobs are tagged with.
4. When the user finds a job they wish to apply for, they enter their price per hour and an estimate of time.
5. The freelancer's account is added to a list of candidates for the employer to view.

Scenario 5: Completing a Job:

1. A freelancer is chosen for a job
2. Freelancer completes the work.
3. Invoice for the hours worked is created by the employer.
4. The employer pays the freelancer, as well as the gas prices and maintenance fees.
5. Review of each user is completed by the other user.
6. The review is stored on the blockchain.

Scenario 6: Two Users Trading Work:

1. Employer posts a job, and indicates that they are available for trading work.
2. This employer contacts another employer who is available for trading or is contacted by another trade employer looking to trade work.
3. The two employers come to an agreement of how much work each should be completed by each party.
4. Employers agree that work is completed to a satisfactory amount.
5. The contract is completed.

Scenario 7: Review Containing Bad Language (If full reviews are implemented):

1. A job is completed and the payment is processed.
2. Both parties involved review each other, one user uses profanities in their review and the review is placed on the blockchain.
3. Another user views the employer from the previous transaction and views the reviews for this employer.
4. The review containing the profanities is censored.

2.4 Constraints

2.4.1 Learning:

We will need to spend some amount of time getting to grips development on the Ethereum platform (Programming smart contracts with Solidity and interaction between the blockchain and web3.js), as this is an unfamiliar style of programming to us.

2.4.2 Ethereum:

This project requires that the Ethereum environment does not become less stable than it has in the past months. As Ethereum is relatively new it is possible that we may be constrained by a feature that has not been added yet / is too buggy for general use.

2.4.3 Security:

A great deal of care must be taken when dealing with users currency, so we must ensure that our site functions reliably and efficiently. Less efficient code costs real world money when using Ethereum as it impacts the cost to add it to the blockchain.

2.4.4 Time:

The project must be completed by Friday 9th March 2018, which provides roughly three months from time of writing to complete the project (A relatively tight timeline when other college work is included).

2.4.5 Slander:

It was brought up during our proposal presentation that we will need some way to handle profanities and slander in the reviews. This will be a difficult problem due to the immutable nature of the blockchain.

3. Functional Requirements

3.1 New Users

Description:

- User enters the site through Ethereum connected browser. User registers their account as a freelancer or employer.

Criticality:

- The site cannot function without the users.

Technical Issues:

- Linking Ethereum accounts to the site and storing the associated information will be difficult due to the anonymity associated with Ethereum accounts.

Dependencies:

- N/A

3.2 Posting a Job:

Description:

- Employer accounts must be the only account that is allowed to post jobs to the site.

Criticality:

- This is very important to the use of the system, as employers will create jobs that the site is designed to cater for.

Technical Issues:

- Limiting posting features to certain Ethereum accounts could raise issues.

Dependencies:

- N/A

3.3 Applying to a Job:

Description:

- Freelance accounts can apply to open jobs with a price and estimate. A list of candidates that have applied is made accessible to the employer who posted the job.

Criticality:

- This is an integral part of the system as this is how workers for jobs will be selected.

Technical Issues:

- Possible issues with limiting functions to certain Ethereum accounts.

Dependencies:

- Relies on job browsing as freelancers must be able to find jobs. (Requirement 3.4)

3.4 Browsing All Open Jobs/Available Candidates:

Description:

- All users of the site, regardless of whether they are an employer or a freelance user, will be able to browse all jobs and candidates. Employers can contact candidates to apply to job.

Criticality:

- Very important piece of the system as this is how the two people needed for a transaction will come in contact.

Technical Issues:

- Pulling all available jobs from the blockchain in order to display them may cause issues.

Dependencies:

- Relies on jobs being posted. (Requirement 3.2)

3.5 Employer Accepts Candidate:

Description:

- Employer chooses a candidate from the list of applications. Candidate is notified.

Criticality:

- Another integral part of transactions on the site.

Technical Issues:

- Notifying the accepted candidate could cause issues, unsure if this can be accomplished through Ethereum.

Dependencies:

- Relies on the apply and job posting functionalities. (Requirements 3.3 & 3.4)

3.6 Trading Work Instead of Currency:

Description:

- Employers who are open to trading work will be able to browse other employers open to trading work in a similar way that they browse freelance users, contact each other in the same way, and complete a trade contract instead of a normal transaction contract.

Criticality:

- Not part of the basic functionality of the service but will be a nice addition on top of the basic functionality.

Technical Issues:

- Development of an Ethereum smart contract could be difficult when not primarily trading in Ether.

Dependencies:

- Will rely on the job browsing and job posting functionalities. (Requirements 3.4 & 3.2)

3.7 Contact Between Users:

Description:

- Users can contact each other to arrange the terms of the contracts.

Criticality:

- Not as critical as previous requirements as users could simply have their chosen emails associated with their accounts instead.

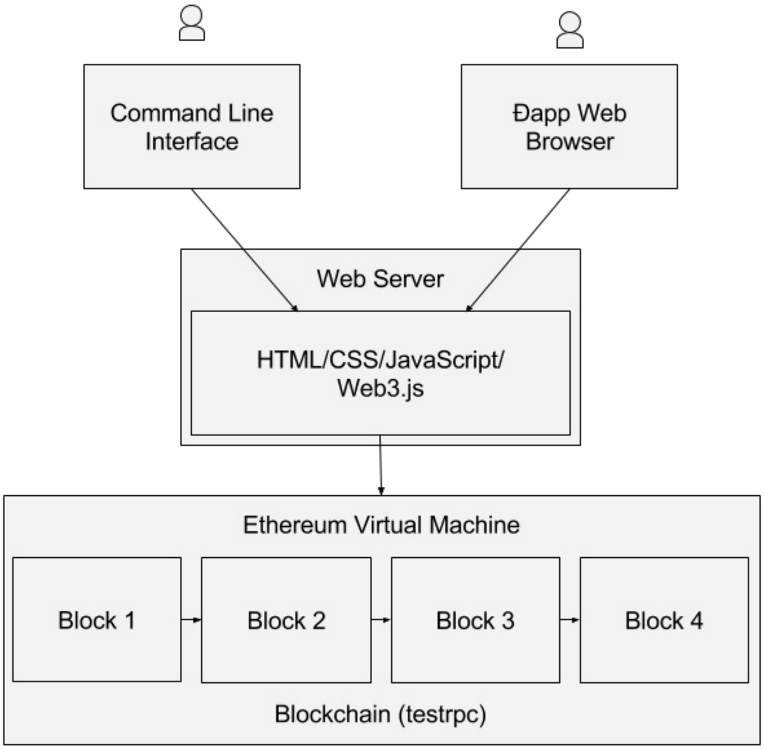
Technical Issues:

- A built-in version could be developed using the Ethereum whisper protocol, although the same could be achieved through email.

Dependencies:

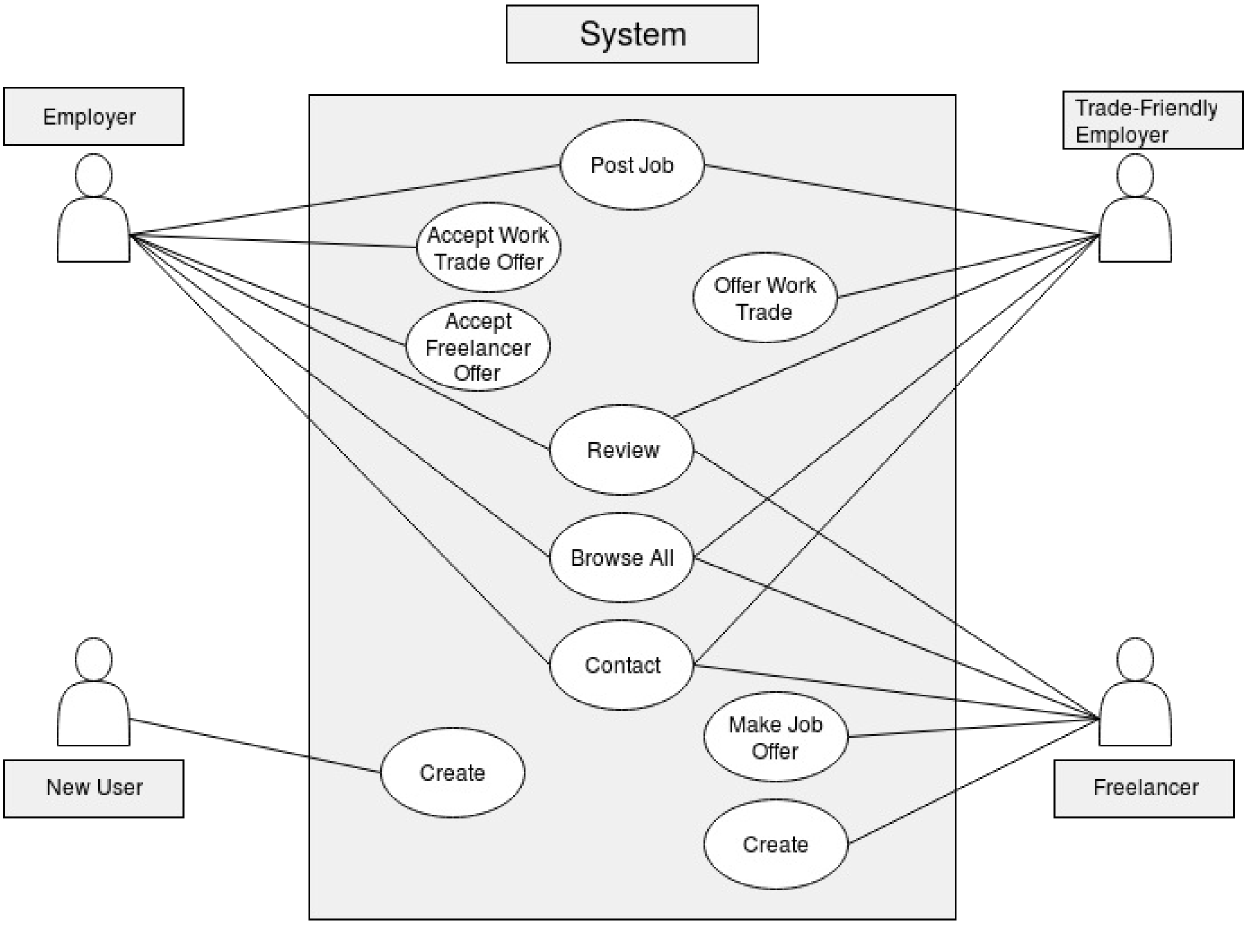
- Will rely on the job browsing and job posting functionalities. (Requirements 3.4 & 3.2)

4. System Architecture

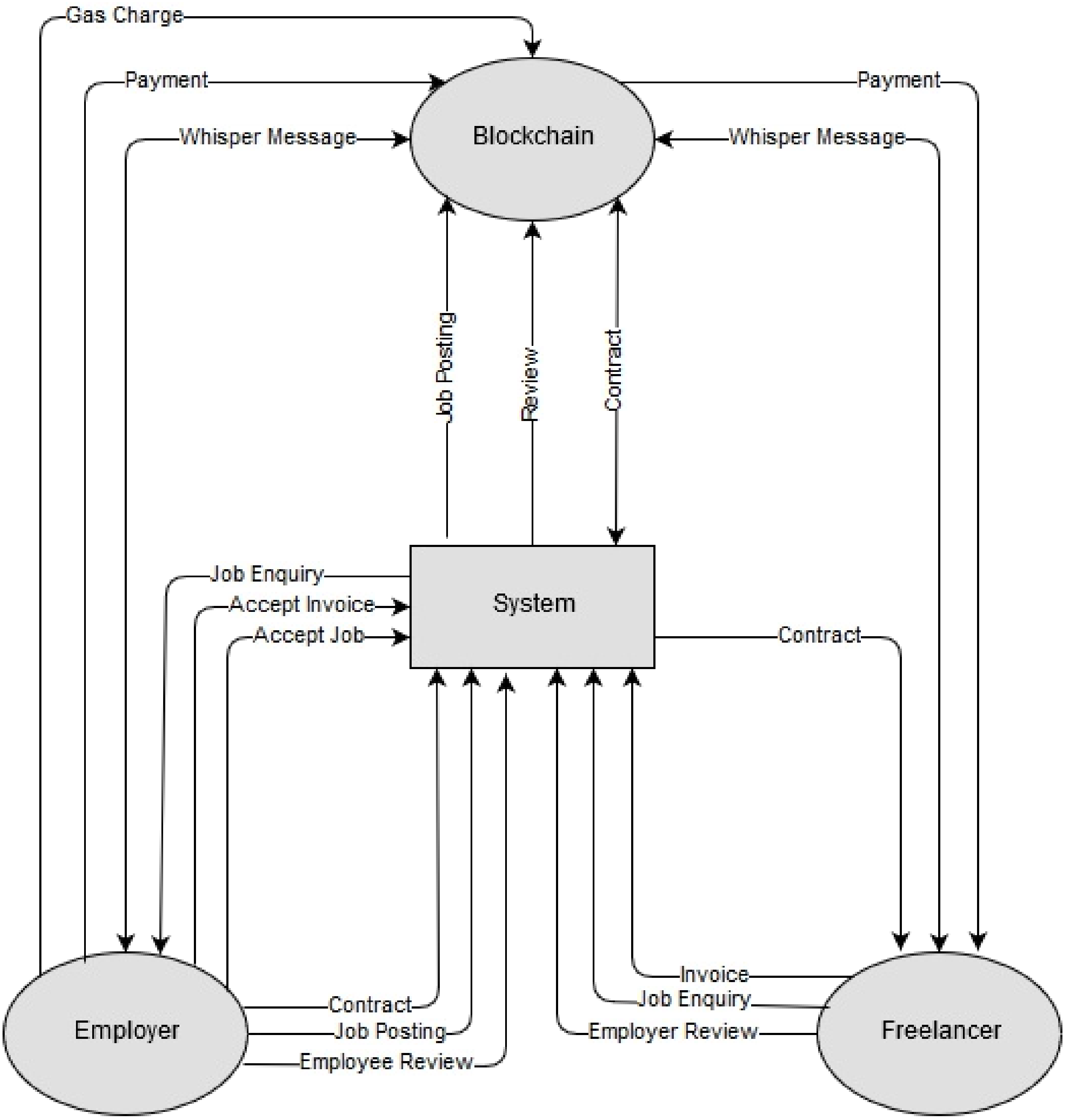


5. High-Level Design

5.1 User Diagram



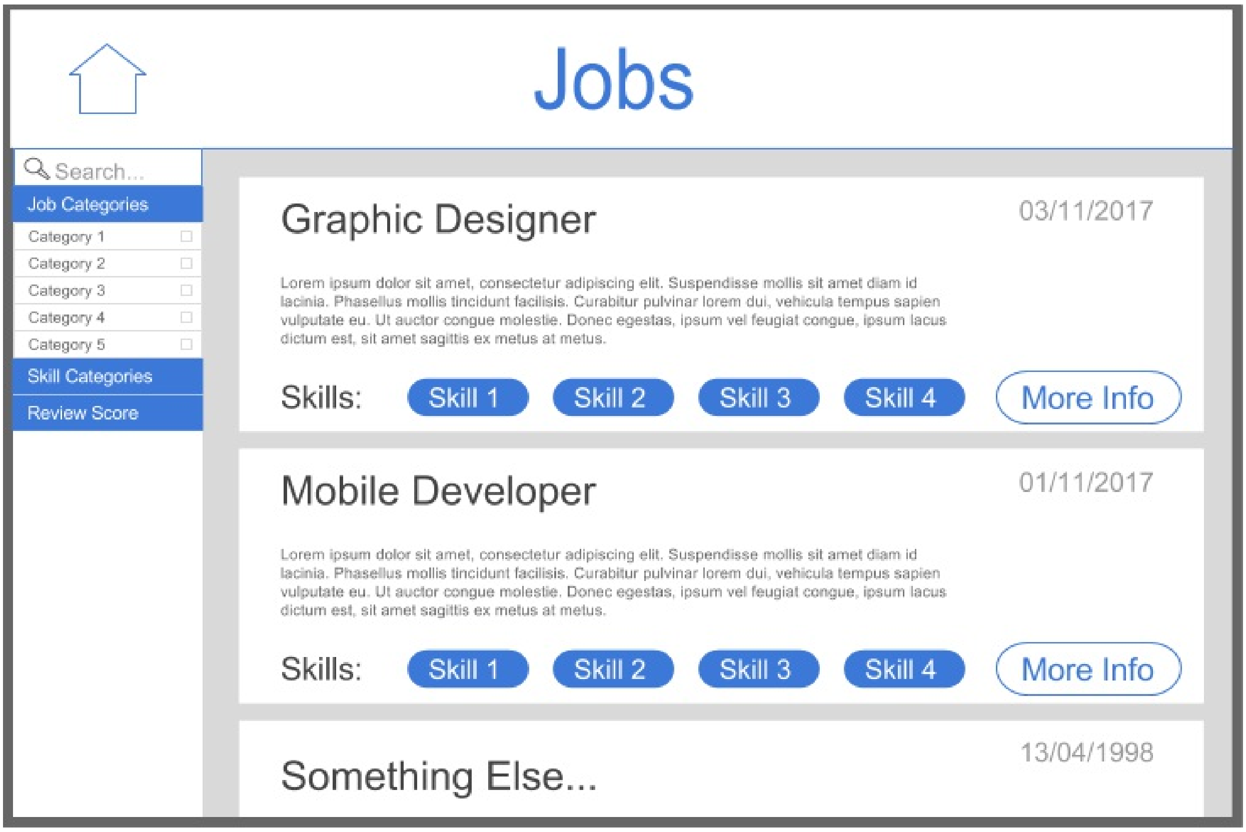
5.2 Context Diagram



6. User Interface

Below are mock ups of possible user interfaces that could be used on the site. The images showcase some of the information that will need to be displayed to users such as the skills required for a job/the skills a candidate has, and some of the past reviews of transactions associated with the user. The development of mockups forced a more in-depth analysis of the different features of the site.

6.1 Browse Jobs Mockup



Graphic Designer

03/11/2017

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse mollis sit amet diam id lacinia. Phasellus mollis tincidunt facilisis. Curabitur pulvinar lorem dui, vehicula tempus sapien vulputate eu. Ut auctor congue molestie. Donec egestas, ipsum vel feugiat congue, ipsum lacus dictum est, sit amet sagittis ex metus at metus.

Mobile Developer

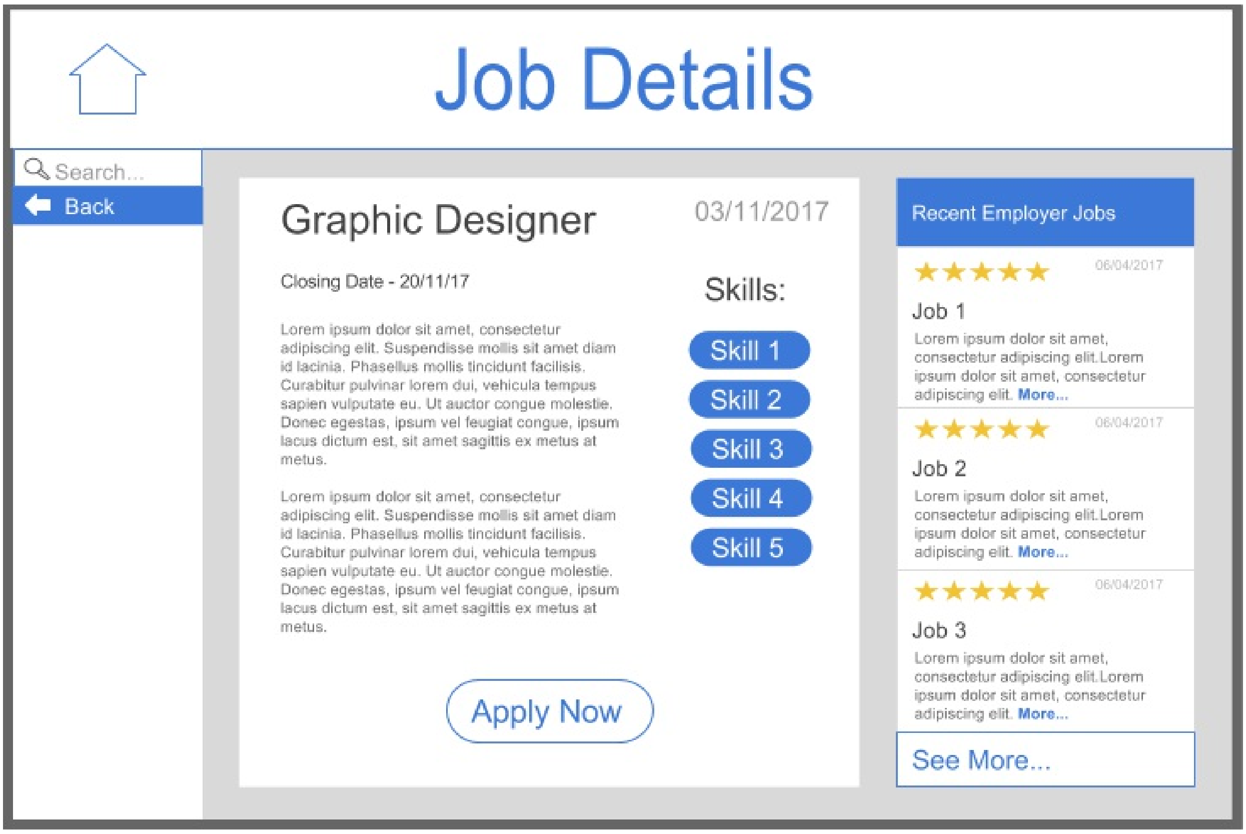
01/11/2017

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse mollis sit amet diam id lacinia. Phasellus mollis tincidunt facilisis. Curabitur pulvinar lorem dui, vehicula tempus sapien vulputate eu. Ut auctor congue molestie. Donec egestas, ipsum vel feugiat congue, ipsum lacus dictum est, sit amet sagittis ex metus at metus.

Something Else...

13/04/1998

6.2 Job Details Mockup



Graphic Designer

03/11/2017

Closing Date - 20/11/17

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse mollis sit amet diam id lacinia. Phasellus mollis tincidunt facilisis. Curabitur pulvinar lorem dui, vehicula tempus sapien vulputate eu. Ut auctor congue molestie. Donec egestas, ipsum vel feugiat congue, ipsum lacus dictum est, sit amet sagittis ex metus at metus.

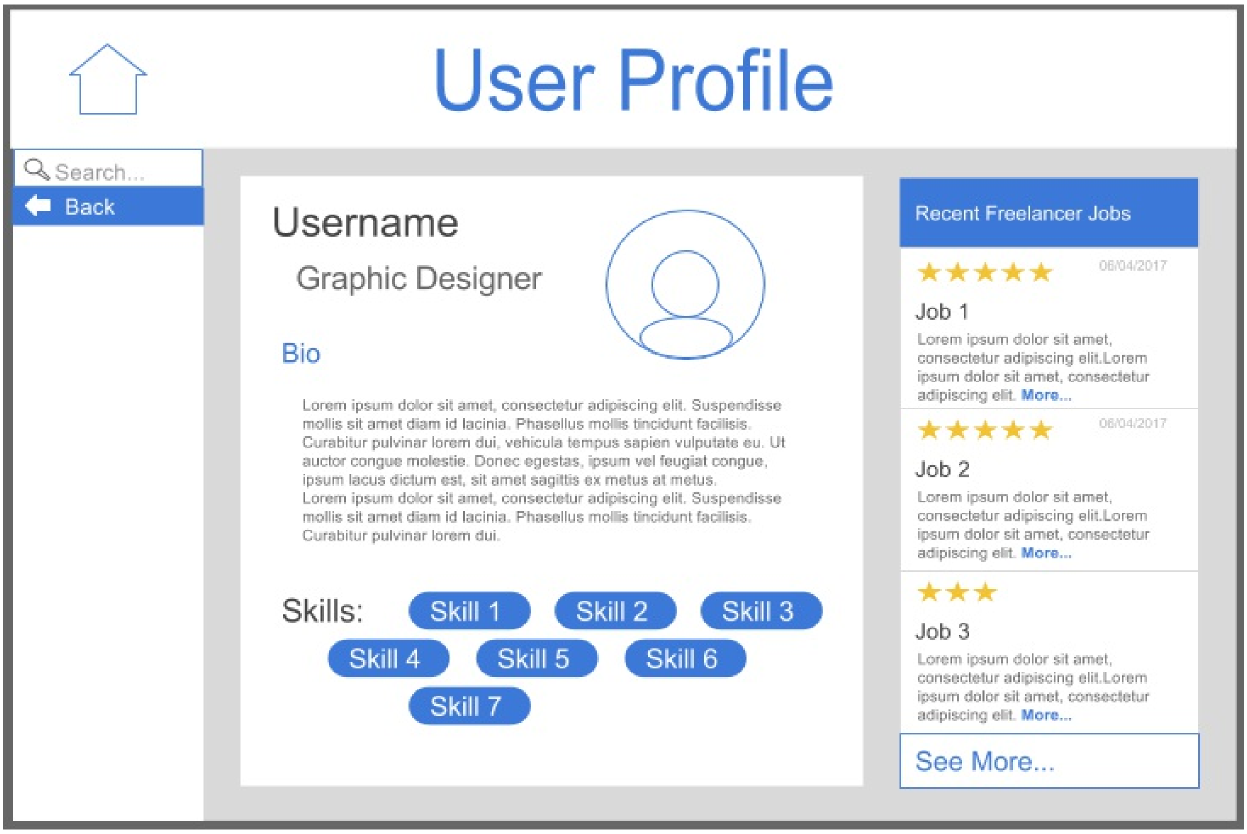
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse mollis sit amet diam id lacinia. Phasellus mollis tincidunt facilisis. Curabitur pulvinar lorem dui, vehicula tempus sapien vulputate eu. Ut auctor congue molestie. Donec egestas, ipsum vel feugiat congue, ipsum lacus dictum est, sit amet sagittis ex metus at metus.

★★★★★ 06/04/2017
Job 1
Lorem ipsum dolor sit amet, consectetur adipiscing elit.Lorem ipsum dolor sit amet, consectetur adipiscing elit. More...

★★★★★ 06/04/2017
Job 2
Lorem ipsum dolor sit amet, consectetur adipiscing elit.Lorem ipsum dolor sit amet, consectetur adipiscing elit. More...

★★★★★ 06/04/2017
Job 3
Lorem ipsum dolor sit amet, consectetur adipiscing elit.Lorem ipsum dolor sit amet, consectetur adipiscing elit. More...

6.3 User Details Mockup



Username

Graphic Designer

Bio

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse mollis sit amet diam id lacinia. Phasellus mollis tincidunt facilisis. Curabitur pulvinar lorem dui, vehicula tempus sapien vulputate eu. Ut auctor congue molestie. Donec egestas, ipsum vel feugiat congue, ipsum lacus dictum est, sit amet sagittis ex metus at metus. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse mollis sit amet diam id lacinia. Phasellus mollis tincidunt facilisis. Curabitur pulvinar lorem dui.

★★★★★ 06/04/2017
Job 1
Lorem ipsum dolor sit amet, consectetur adipiscing elit.Lorem ipsum dolor sit amet, consectetur adipiscing elit. More...

★★★★★ 06/04/2017
Job 2
Lorem ipsum dolor sit amet, consectetur adipiscing elit.Lorem ipsum dolor sit amet, consectetur adipiscing elit. More...

★★★
Job 3
Lorem ipsum dolor sit amet, consectetur adipiscing elit.Lorem ipsum dolor sit amet, consectetur adipiscing elit. More...

7. Test Plan

Testing will be an important part of this project. We will need to be certain of how our code works as cryptocurrency prices can be very volatile and as a result of this, a small error in our codes price or gas calculations could result in a substantial loss of money for some of our users.

We plan for this project to be continuously tested. The Truffle development framework we plan to use for smart contract development allows for the creation of unit test cases that can be run in groups, individually or can be programmed to run automatically whenever a build is compiled.

Once the basic site is complete we will create a series of simple test cases to act as our sanity test for each new build we deploy. In addition to this we will have test cases based around each new feature we implement after the minimum viable products development.

8. Preliminary Schedule

8.1 Overview of Schedule

We attend a meeting with our project supervisor every Wednesday at 10:00am. This is an open meeting with three other project groups also in attendance. This allows us to hear exactly how the other groups are progressing and the ideas that they've come up with, some of which could be helpful in our own project. This open meeting style also allows us to give our input on the ideas of the other groups, and vice versa. Each week we take turns to give a description of the project, what we accomplished in the past week, as well as what we aim to accomplish in the upcoming week. This helps us to stay on target and keep a steady progress.

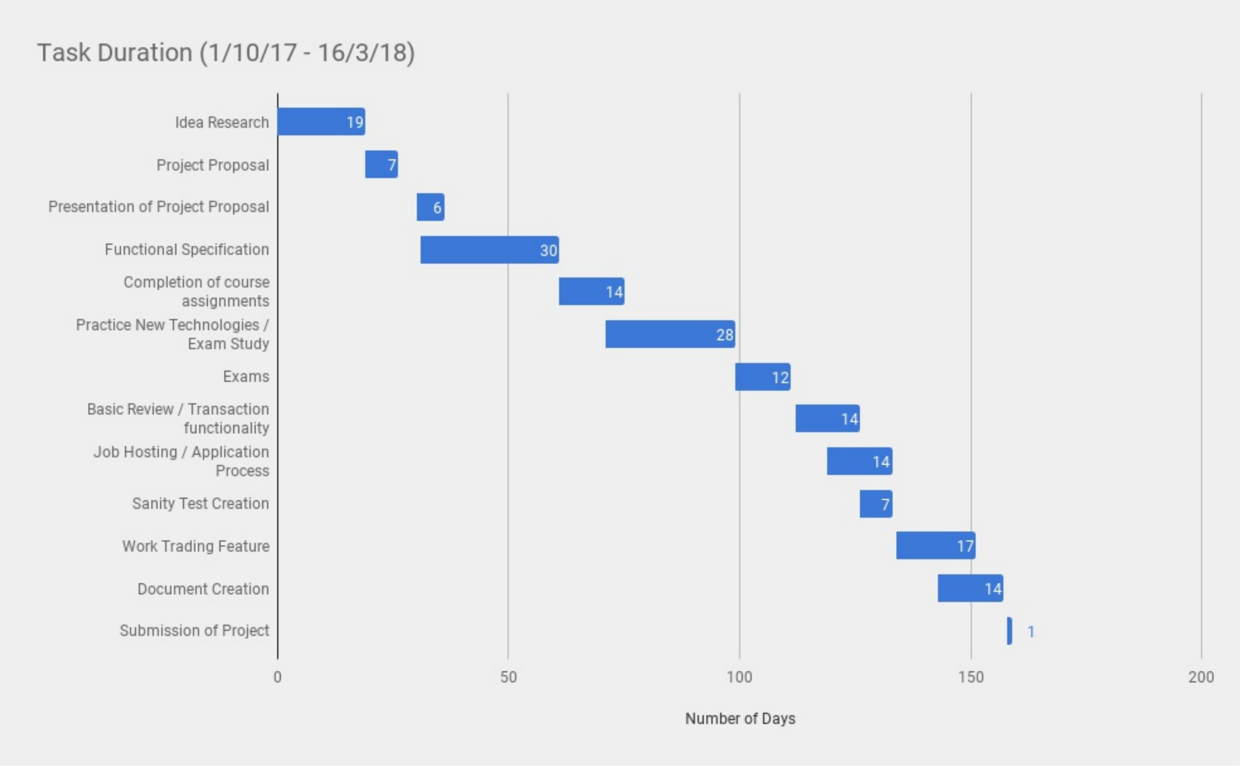
Below we have a provisional Task View and GANTT chart which we plan to update throughout the course of our project.

8.2 Task View

Task Name	Start	End	Duration	Status
Idea Research	**1/10/17**	**11:59 20/10/17**	**19d**	**Completed**
Idea Brainstorming	1/10/17	11:59 10/10/17	10d	Completed
Idea Feasibility Research	10/10/17	11:59 20/10/17	9d	Completed
Project Proposal	**20/10/17**	**17:00 27/10/17**	**7d**	**Completed**
Write Proposal	20/10/17	11:59 23/10/17	3d	Completed
Finalise Proposal	20/10/17	11:59 24/10/17	1d	Completed
Approved by Supervisor	24/10/17	11:00 25/10/17	1d	Completed
Submit Proposal	26/10/17	17:00 27/10/17	1d	Completed
Presentation of Project Proposal	**31/10/17**	**12:00 6/11/17**	**7d**	**Completed**
Prepare Slideshow	31/10/17	11:59 4/11/17	4d	Completed
Practice Presentation	5/11/17	12:00 6/11/17	3d	Completed
Submission of Functional Specification	**1/11/17**	**17:00 1/12/17**	**30d**	**Completed**
Project Development	**6/12/17**	**17:00 8/3/18**	**92d**	**In Progress**
Create GitLab Project Page	27/10/17	18:00 28/10/17	1d	Completed

Practice Solidity & web3.js	11/12/17	11:59 8/1/18	32d	In Progress
Basic Review / Transaction functionality	21/1/18	11:59 4/2/18	14d	-
Job Hosting / Application Process	28/1/18	11:59 11/2/18	14d	-
Sanity Test Creation	4/2/18	11:59 11/2/18	7d	-
Work Trading Feature	12/2/18	11:59 1/3/18	17d	-
Technical Manual / User Manual Creation	21/2/18	11:59 7/3/18	14d	-
Submission of Project	**8/3/18**	**17:00 9/3/18**	**1d**	**-**
Project Demonstrations	**1/3/18**	**17:00 16/3/18**	**16d**	**-**
Prepare Presentation	1/3/18	11:59 6/3/18	6d	-
Finalise Presentation	6/3/18	11.59 7/3/18	1d	-
Practice Presentation	5/3/18	11:59 11/3/18	4d	-
Present Project	12/3/18	17:00 16/1/18	4d	-

8.3 GANTT Chart



9. Appendices

Ethereum Development:

- https://ethereum.org/
- http://truffleframework.com/
- Various tutorials from https://medium.com/
- https://dappsforbeginners.wordpress.com/tutorials/introduction-to-development-on-ethereum/
- 'Building Blockchain Projects' by Narayan Prusty, Packt publishing
- 'Introducing Ethereum and Solidity' by Chris Dannen, Apress publishing

Amazon Web Services:

- https://aws.amazon.com/

Forums:

- https://ethereum.stackexchange.com/
- https://forum.ethereum.org/