MAKERERE UNIVERSITY

COLLEGE OF COMPUTING & INFORMATION SCIENCES

STORAGE OF ACADEMIC RECORDS USING BLOCKCHAIN TECHNOLOGY
By
CSC 19-03

DEPARTMENT OF COMPUTER SCIENCE
SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY

A Project report Submitted to the School of Computing and Informatics Technology For the Study Leading to a Project Report in Partial Fulfillment of the Requirements for the Award of the Degree of Bachelor of Science in Computer Science Of Makerere University

Supervisor: Prof. Engineer Bainomugisha

Department of Computer Science
School of Computing and Informatics Technology, Makerere University
baino@cis.mak.ac.ug

April, 2019

Declaration

We Group CSC 19-3 do hereby declare that this Project Report is original and has not been published and/or submitted for any other degree award to any other University before.

GROUP MEMBERSHIP			
No.	Name	Registration Number	Signature
1	Araka Stephen Gift Mukoya	16/K/2148/EVE	
2	Agwa Daniel	16/U/2872/EVE	
3	Omoding John	16/U/11013/EVE	
4	Wafula Derrick	16/U/20275/EVE	

Approval

This Project Report has been submitted for examination with the approval of the following supervisor.

Professor Engineer Bainomugisha
BSc. CS, MSc. CS,
School of Computing & IT
College of Computing & IT
Makerere University
baino@cis.mak.ac.ug

Signed:
_____

Date:
_____

Dedication

We dedicate this report to the Almighty God without whom we can do nothing. We further dedicate it to our parents and guardians for their unceasing and selfless support throughout our stay in this university.

Acknowledgement

We are deeply indebted to our project supervisor Professor Engineer Baino-mugisha whose unlimited steadfast support, expertise and inspirations has made this project a great success. In a very special way, we thank him for every support he has rendered unto us to see that we succeed in this challenging study.

Special thanks go to our friends and families who have contained the hectic moments and stress we have been through during the course of the research project.

We thank the school for giving us the grand opportunity to work as a team which has indeed promoted our team work spirit and communication skills. We also thank individual group members for the good team spirit and solidarity.

Abstract

Students' final academic records (transcripts) in various institutions of learning are issued on paper. This creates room for some dishonest people to create counterfeit transcripts showing results they did not attain. Institutions of course have labels like stamps, seals among others to show that these documents are authentic. However, con-men have mastered the art of replicating these labels to look a lot like the original ones. In the end, many incapable people flood the employment industry with counterfeit academic records. Besides the problem of un-authentic results, there is lots of bureaucracy involved in obtaining the original copy of the transcript.

Our research findings indicate that sharing and storage of academic records on a distributed network and sharing them remotely would close the gap that allows development of counterfeit documents. This gives an alternative to having a hard copy transcript that is attained faster. It also limits the process of having to verify academic records repeatedly. Furthermore, it becomes easier for employers to verify a potential employee's records.

Storage of Academic Documents using Blockchain arose after the realization of the bureaucracy and loopholes described above. The system contains a web application that manages students' progressive academic records in a centralized manner and when the transcript is verified by the institution's authority, it is stored on a decentralized network using a protocol known as IPFS (Inter Planetary File Systems). On this network, each academic record has a unique identifier known as a hash value. This hash value is then stored on a blockchain network which itself is decentralized.

Transcripts are very important in the job market in the world today and having many counterfeits flooding the industry can be fatal to the economy. This research highlights the problems involved in the issuance of these transcript with Makerere University as the institution of focus. Further covered are the solutions to the challenges and how they were developed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

There have been efforts to streamline delivery of authentic information about students who attend higher institutions of learning.

In a bid to achieve this, Makerere University, like any other institutions, uses a conventional online system to manage students' results. Upon admission to the university, an account is opened for each student through which they can can view their progressive academic records which are uploaded by a centralized administration that comprises of the school registrars. For security purposes, users are required to enter credentials i.e Student Number / Registration Number and Password. Through this, only authentic users can access the system. The use of this online system has eased students' access to their records as opposed to the formally crowded notice boards where results were pinned. For purposes of security, the system runs on the university's intranet network.

Upon completion of their study at the University, students' are awarded with an inventory of the courses taken and grades earned throughout their course of study. This comes in the form of an academic transcript. However, since the issued transcript is in hard copy, this comes with a number of complications. For example security risks, duplication and forgery, ease of access and distribution.

Recently one of the breakthroughs in technology has been the management of records belonging to a group of people using a decentralized system. One may wonder how this is possible. Well, this is done using a technology known as blockchain.

A blockchain is a growing list of records which are linked using cryptography[1]. Blockchain can not only be used to store records, but also be used for their secure manipulation. By design, a blockchain is resistant to modification of the

data stored on it. It is "an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way".

In the context of Makerere University for example, blockchain technology would be a good alternative to storing syudents' transcripts. In addition to issuing transcripts in hard copy form, using blockchain guarantees one some extra features that would better the management and storage of students' records. For example [2] :

Transparency Both parties who are interested in viewing academic credentials can see them on the blockchain. This ensures that only people with ownership rights can make decisions about who has access to particular information.

Immutability Blockchains are the most secure source for storing the information right now. They rely on the integrity of the network to ensure the authenticity of the stored information. Thus, academic certificates stored on the blockchain are immutable.

Disintermediation - Using the blockchain to store and share academic credentials helps us bypass the need for a central controlling authority that manages and keeps records. This makes the overall process of storing credentials more trustworthy as there are no middlemen involved.

Collaboration Once the information becomes available on the blockchain, it is much easier to ascribe ownership, and therefore safer to share the information without the fear of this information being compromised.

In this project, we intend to build a blockchain application that will aid the management of students' records at Makerere University.

1.2 Problem Statement

In an article published by the Daily Monitor in March 2017, 87 per cent of graduates cannot find jobs. "According to National Planning Authority (NPA) statistics released from [6 12, March 2017], 700,000 people join the job market every year regardless of qualification but only 90,000 get something to do."[3] One of the major causes of this is the problem of fake academic papers. It is a well-known secret that while a considerable number of Ugandans tender in doctored documents to get employment, a huge number of people with genuine papers remain unemployed.

Currently, it is pretty easy to obtain a forged transcript. For as little as $100, one can acquire a fake transcript that required another person four years of study. Unfortunately for employers, it is not so obvious to tell the difference between authentic and counterfeit.

This project aims at tackling the above-mentioned shortcoming with a focus on the results testimonial issued to a student of the university. Enforcing means of verification of academic documents is our proposed approach to reducing these rates of unemployment, especially for university graduates.

1.3 Main Objective

The overall aim of this project is to improve authenticity of documents by implementing block chain technology.

1.3.1 Specific Objectives

The specific objectives of the study were:

- To ensure safety of students' records. Unlike paper-based records that can easily get lost, once put in block chain, the students' grades will be safe.

- To create an open, distributed catalog that makes it easy for students to share their transcripts with whom they please.

- To provide a medium of verification that can be used by employers in telling a true transcript from a counterfeit.

1.3.2 Scope

The challenge mentioned in the problem statement above is broad and cuts across a number of institutions of higher learning in the nation and even beyond borders. For this project however, the focus will be on Makerere University Main Campus located in Kampala, Uganda. Furthermore, the project will focus on issuance of the academic transcript to a student of the university among other documents.

1.3.3 Significance

In the year 2010, the African Development Banks (AfDBs) Partnership Forum had put national unemployment figures at 83 per cent. The World Bank had at the same time put youth unemployment in Kampala alone at 32.2 per cent and unemployment among university graduates in Kampala at 36 per cent.[4]
To many university students, this is a problem that may cause a lot of stress and worry among the majority. However, every problem has in it the seeds of its own solution.

No one would like to eat half baked bread leave alone the raw dough. The

employment industry is flooded with so many employees that have barely attained the knowledge to perform the tasks they are given simply because they were able to forge documents and convince their employer that theyre capable.

This project is aimed at closing such loop holes that allow people to get away with forged academic documents.

For employers, block chain technology will avoid them having to spend valuable time checking candidates' educational credentials by having to call universities or to pay a third party to do the job.

A country's economy becomes more productive as the proportion of educated workers increases since educated workers can more efficiently carry out tasks that require literacy and critical thinking. Uganda has an adult literacy rate of about 75%[5]. By providing them with the deserved access to employment and to participate in building the society, the economy will be much more productive and the country's GDP will be boosted.

Chapter 2

Literature Review

2.1   Background

Storing and verifying students' academic transcripts can be costly and time-consuming for academia, students and businesses alike. Through this project, we look to turn to blockchain technology for a solution. But first of all, let us start with a review of the blockchain technology and its current applications.

Bitcoin which is a peer to peer electronic cash system was revealed to the world in 2008 by Satoshi Nakamoto whose identity is still unknown and was offered to the open source community in 2009. The decentralised nature of the technology used by bitcoin came to be known as blockchain.
 Bitcoin uses cryptographic proof instead of the trust-in-the-third-party mechanism for two willing parties to execute an online transaction over the Internet. Each transaction is protected through a digital signature, is sent to the public key of the receiver, and is digitally signed using the private key of the sender. In order to spend money, the owner of the cryptocurrency needs to prove his ownership of the private key.
The entity receiving the digital currency then verifies the digital signature, which implies ownership of the corresponding private key, by using the public key of the sender on the respective transaction.
Each transaction is broadcasted to every node in the Bitcoin network and is then recorded in a public ledger after verification. Every single transaction needs to be verified for validity before it is recorded in the public ledger.

Storing documents on the blockchain

Up until 2014, blockchain did not seem to have much potential independent of bitcoin. In 2014, some people realised other potential uses of blockchain in disciplines like supply chain, healthcare, insurance, education among others.

Figure 2.1: How blockchain works

Over the past several years, there has been a keen interest in how we can use blockchains for storing documents.

There are two main ways to store a document on the blockchain. One option is to store the entire document itself on-chain. Alternatively, one can store a hash of it on the blockchain.

- **Storing Entire Document** - Storing a whole document on-chain is possible with certain blockchains.
  However, we found out that it is rarely a good idea because of something called access latency. Fully decentralized public blockchains have thousands of nodes and this means it takes long for network users to upload and download files, such as documents.

- **Storing a Hash** - This method involves storing a documents hash on-chain while keeping the whole document elsewhere. The document could be stored in a centralized database or on a distributed file storage system. The document can be put through a secure hash algorithm like SHA-256 and then the hash is stored in a block. We find this to be the most efficient method as it saves a huge amount of space and cost. Additionally, using the hash, one is able to tell if someone tampers with the original document.

There are few projects that focus on documents alone right now. Most are built around decentralized file storage, which includes documents.

One project that is focused specifically on documents, particularly signed doc-

uments, is Blocksign[6]. This uses the hash method. A user will sign the document and send it to Blocksign, where it is then hashed, and the hash is stored on the Bitcoin blockchain.

Other cryptocurrency projects designed for decentralized storage more generally include Siacoin, Storj and Cryptyk.

Siacoin[7] - uses their distributed network to store an encrypted version of one's document. The Siacoin network is comprised of hosts who provide storage and clients who desire storage. Clients and hosts agree upon contracts detailing the commitments made by the storage providers. Sia's own proof of work blockchain stores these contracts.

Storj[8] - runs atop the Ethereum[9] blockchain. A hash of the document is stored within a hash table on-chain. Additionally, its distributed network also stores your document.

Cryptyk[10] - an enterprise-focused platform to store documents, uses a blockchain more distantly than all of the above . You do not store any documents or hashes on-chain. Instead, a distributed cloud system stores the documents. The platform only uses a blockchain to manage and referee document access and sharing.

Described below are not only some of the contributions, but also weaknesses and gaps that are associated with this technology.

Contributions

Because decentralized applications run on the block chain, they benefit from all of its properties, which include:-

- **Immutability** A third party cannot make any changes to data.

- **Corruption & tamper proof** Apps are based on a network formed around the principle of consensus, making censorship impossible.

- **Secure** With no central point of failure and secured using cryptography, applications are well protected against hacking attacks and fraudulent activities.

- **Zero downtime** Apps never go down and can never be switched off.

For an example of the contribution of this technology, we look at the University of Nicosia in Cyprus, which is using the technology to record students' achievements. According to George Papageorgiou, a digital currency lecturer at the university, the technology is proving popular. He had this to say to CNBC News:
"We've only encountered enthusiasm in the practical uses so far and students

are glad to be able to verify, with their new knowledge and the blockchain, that their digital certificate is genuine and that it cannot be recreated.
We believe this instills confidence in both students and potential employers that (they) can check on their own, whether a presented certificate is real or not".[11]

This is proof that the implementation is already reaping fruits in some institutions around the world.

Weaknesses and gaps

However, despite all the possibilities offered by blockchain, there have been various challenges associated with it.

We observe challenges in both the perspective of the end-user, and we, the researchers. From the users view, according to Donald Clark [12], an EdTech entrepreneur and advisor of EdTech companies, some public sector organizations just dont like the innovation and stick to their institutional silos. This is basically because the technology has not been around for a long time which makes many potential users have doubts about its possibilities. To overcome this, we intend to train the parties in these institutions on how to use this technology and also show them the advantages.

From our research perspective, the major challenge is that the subject of study is of a relatively early stage. Blockchain technologies are under active development globally, and there may be recent advances that impact our findings.

In conclusion, it is important to note that blockchain is a technology that clearly has applications in the world of learning at individual, institutional and international levels. It is relevant in all sorts of contexts: schools, colleges, universities among others.

One thing we know for sure is that students have their eyes open and are looking for alternatives. Perhaps, like Bitcoin, the blockchain revolution will ultimately come from left of field.

Chapter 3

Methodology

3.1 Introduction

As stated earlier in the document, Unauthentic academic documents cause a great deal of trouble in the employment industry and a country's economy. The need for a more secure way of dispatching academic results is the major drive to the operation of this project with a focus on Makerere University.

This chapter contains a description of the techniques, methods and tools used during the research process. Also contained in this chapter, is a description of the implementation, testing and validation of the system.

3.2 Data Collection

Data collection for this project was approached using both qualitative and quantitative methods. The quantitative methods helped in evaluating the impact of the current results storage methods on the students, employees and other stakeholders involved. On the qualitative end of the card, we were able to obtain opinions on the present day methods of storing students' results. More importantly, learning stakeholders' opinions on a possibility of storing results in a decentralised manner was a major drive to use qualitative methods. The data collection process in this project was done through interviews with the various stakeholders of the system for example the students and administrative assistants at different colleges of the university. Document reviews were also done in the process of data collection.

From the vast number of stakeholders, sampling was done using the systematic sampling technique. This is a type of probability sampling where every element of the sample space has an equal chance of being picked.

3.2.1 Interviews

Interviews were carried out with administrative assistants at various colleges of the university. These are the university employees in charge of feeding students results into a system where they're managed. This made them a key source of information. Information sought included;

- Strengths of the current results storage system.

- Weaknesses of the current results storage system.

- Opinion on a possibility of a decentralised storage system for the students records.

The conversations held during the interviews were in person in the various offices of the different administrative assistants. The conversations with each respondent took approximately thirty minutes excluding the amount of time spent trying to make appointments.

Relevance of Interviews

- Feedback was quickly obtained from the respondents.

- There was room to get more information than what was intended in the interview script for example, we learnt of a results management and storage system that was used at the College of Computing and Information Sciences for about four years but was never adopted by the university.

- Since the conversations happened in person, there was room to read the non verbal cues from the respondents. This creates room to assess whatever the respondent is saying from their body language which would rather not be possible with other methods [13].

3.2.2 System Review

As students of the university, we had access to the current results storage system with privileges granted to students. We carried out a review of the system accessed through the url ar.mak.ac.ug. This method was a source of qualitative information as it's major purpose was to learn the strengths and weaknesses of the system. some of the attributes looked out for were security, availability among others.

Relevance of a System Review

- We were able to obtain first hand information from the system itself.

- Access to the source of information (The students' portal) is much easier as it is available for us to access.

3.2.3   Document Review

Through literature review, we collected information from already existing related research. This research was done on already existing block chain applications similar to the one this project is focused on for example Blocksign, Siacoin, Storj and Cryptyk. This information was gathered from online sources like the internet, journals and other relevant materials on the problem domain from libraries around the university.

3.3   Data Analysis

At this stage of the project, a vast amount of data had been collected and the team was then tasked to filter out the relevant data. A great deal of the data collected was qualitative. This therefore called for qualitative data analysis techniques. Our findings were examined against a predefined framework based on the objectives of the project. The major method used is the Inferential data analysis method. This approach studies the relationship between various variables for example the relationship between a student and an employer.

3.4   System Design and Analysis

In order to achieve efficient project management, we adapted the agile software development methodology where the requirements and solutions evolve together throughout the development process. This approach significantly contributed to the success of the project because we were able to anticipate the need for flexibility in time.

3.4.1   System Design

The system was designed using use case and class diagrams. The generated use case diagram visually express how the different users like students interact with the storage system.

In the system design process, we used the object-oriented analysis development method. The systems requirements were determined at this phase. The various classes and relationships among them were also identified at this stage. There are three major analysis techniques used together during the object oriented analysis namely;

- **Object modeling:** This involves developing the static structure of the software system in terms of objects. Here we identified the various classes like students, administrative assistants among others. Identified at this stage also include attributes of the various classes, associations and operations performed on them.

- **Dynamic modeling:** The major activity at this stage involved identifying relationships between the various classes. Some classes extended others that are abstract.

- **Functional modeling:** Here, we identified how the data within an object changes as the processes performed within it are executed. Identified also were the changes made on the data as it moves from one stage to another.

3.4.2   System Implementation

The implementation of the results storage system was done using the agile method of development. This involved continuous iteration of development i.e. different elements of the system like the file storage system were tested repeatedly through out the lifecycle of the project.

For the purposes of managing tasks, we used a method known as scrum which is a category under the agile method of development. Activities in this method included setting up sprints. These sprints covered a time period of one week, each with a set of tasks split among team members to be completed during the sprint. Each team member at the end of the day had to give a report on the activities asigned to him. This we did with daily stand up meetings held using google sheets[14].

For the split tasks, we used github for collaboration among team members. The detailed description of how github was used is further explained in section 6.2.1.

3.5   Testing and Validation

At this stage of the project, we assessed the system to ensure that it can operate in different environments on multiple platforms and also confirm that it satisfies the specified requirements set in the earlier stages of the project. The results storage system has been availed to potential users so the team can get feedback from them. The process here involved deploying a sample transcript on a decentralised network from one node and being able to access it from another node.

The testing was done in two major parts namely functional testing and non-functional testing. These are futher explained below;

Functional Testing

In this phase of testing, the system is tested against the requirements set at the beginning of the project. There are four methods under functional testing that are explained below in their order of execution.

1. **Unit Testing:** Here, we carried out tests on the individual modules of the system which included the web application that handles the results

and the IPFS file handler which stores the transcript on a decentralised network. Tested at this stage is also the smart contract that handles the storage of the transcripts' hash on a blockchain.

2. **Integration Testing:** The different individual modules that were tested in the above stage were tested when integrated together to ensure that once results are verified, the transcript is stored on a decentralised network and its hash value stored on a blockchain network. A hash value is the return value of a hash function after mapping data of an arbitrary size onto data of a fixed size[15].

3. **System Testing:** At this stage, we tested the entire system with all the modules integrated together for bugs. The goal here was to test for user-expected conditions.

4. **Acceptance testing:** Being the final stage, the system was availed to potential users to make sure it works as expected. The major drive here was to ensure that all the project goals and requirements had been met.

Non-Functional Testing

The major drive for this phase of the testing process was to ensure the non-functional requirements were met. This was achieved using four major methods explained below in their order of operation.

1. **Performance Testing:** The system was tested for different behavior under various conditions. An endurance test was carried out to monitor how the storage system would behave under sustained use.

2. **Security Testing:** A security test was carried out to ensure the data and information stored in this system is safe from unauthorized access.

3. **Usability Testing:** Here, the system was tested for usability from a users' perspective. Some of the tested aspects include the Graphical User Interface (GUI), the work flow of the system among other things.

4. **Compatibility Testing:** Finally, the system was tested in different environments. Different browsers like Google chrome and Mozilla Firefox were used to try to access the block chain network.

3.6 Conclusion

To realize all this, we had to put into play our project management skills and this called for the need to track our progress using daily Stand Up meetings. This is recorded and shared on Google sheets. An illustration of this is shown in Appendix A - figure 7.4.

Chapter 4

System Analysis and Design

This chapter describes the study, analysis and design of the system. It highlights the identified requirements and architectural design of the system.

4.1 Overview of the system

The system designed in this project is aimed at having students transcripts stored and shared on a blockchain network. The storage of transcripts on a blockchain network ensures more security in addition to their storage as hard copy documents. To emphasize security, the transcripts are first hashed into data of fixed size and then stored on a blockchain. The system uses IPFS (InterPlanetary File System)[16] to produce a hash code for a transcript. The resultant hash code is then stored on the Ethereum blockchain.

In order to simulate the blockchain locally, this system uses Ganache[**?**] which runs a local instance of the Ethereum blockchain on a user computer. To gain access to the network, the user needs a browser extension known as MetaMask. MetaMask is a bridge that allows users to visit the distributed web through their browsers. It allows one to run Ethereum dApps[18] right in one's browser. Addition of each transcript to the blockchain costs a certain amount of ether[19], the currency for the Ethereum blockchain.
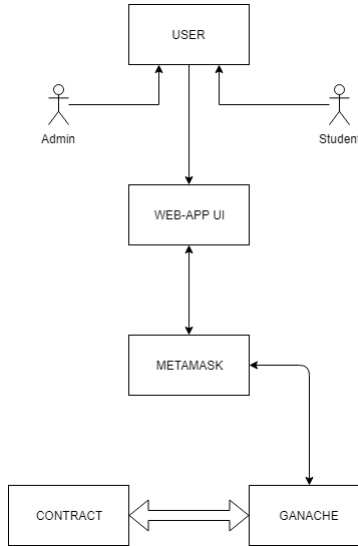
Figure 4.1: Simple system Architecture

4.2 System Analysis

4.2.1 Data Analysis

From the data retrieved during the research process, we found that the present system handling students records is prone to a number of errors and weaknesses.

The current system handling students results at Makerere University does not have a limit to the number of times a student can do a course as opposed to the actual policy by the University where a student is meant to sit for exams in a course not more than three times.

Another discovery from the data received was that the present-day system handling students results at Makerere University can accept an input of a percentage figure greater than 100%.

Another important note is that currently, students' transcripts exist only in hard copy and can only be picked and verified at the University.

4.2.2 System Requirements

The system uses IPFS and the Ethereum blockchain as its main foundation. MetaMask is a browser extension required for user access to the Ethereum blockchain via a browser e.g. Google Chrome, Mozilla Firefox among others. Below are the user requirements, functional and non-functional requirements.

User Requirements

U.1 Allow an administrative assistant enter students results.

U.2 Allow students to access their results without being able to change them.

U.3 Provide for students to easily share their results information with employers.

U.4 Allow employers to easily verify students' results.

Functional Requirements

F.1 Ensure user authentication through login.

F.2 Reject addition of inaccurate results.

F.3 Ensure that the students results are not altered.

F.4 Ensure students' transcript can be saved locally.

F.5 Uploading transcript to IPFS.

F.6 Saving IPFS hash to the blockchain.

Non Functional Requirements

N.1 The system must verify any addition to the blockchain.

N.2 The system must notify the system administrator incase of any unauthorized / filed transactions.

4.3 System Design

The system is dependant on two technologies i.e IPFS and Ethereum. This section includes an overview of how each of these technologies work.

- **IPFS**
  IPFS is an open-source, peer-to-peer distributed hypermedia protocol that aims to function as a ubiquitous file system for all computing devices. It is a complex and highly ambitious project with some serious and profound implications on the future development and structure of the Internet as we know it.

Figure 4.2: IPFS decentralized architecture.
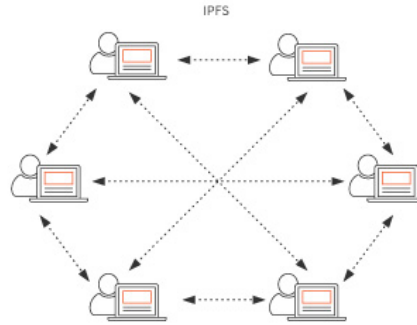
The links between the nodes in IPFS take the form of cryptographic hashes, and this is possible because of its Merkle DAG[20] (Directed Acyclic Graphs) data architecture. Each file and all blocks within it are given a unique identifier, which is a cryptographic hash. Duplicates are removed across the network and version history is tracked for every file.

- **Ethereum blockchain**
  Ethereum's architecture allows for an application to sign transactions offline and relay them to a public node. In this project, since we are running a local instance of the blockchain using Ganache, we have set up a web3 provider engine in our configuraion to transparently sign transactions offline, using Metamask, and send them to an IPFS node.

Figure 4.3: Ethereum blockchain architecture with offline Signing and public nodes.
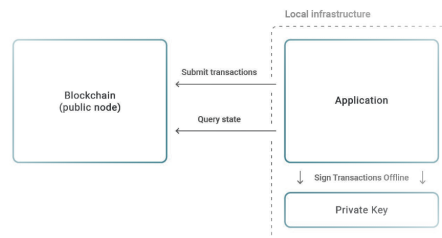
4.3.1 System Architecture

The structure of the system is composed of the following major components:

- A blockchain network built on the Ethereum platform and a local implementation of it running in Ganache.

- The IPFS distributed network for storage and peer-to-peer sharing of records.

- A students' results system that combines:

    1. Creation and Manipulation of results locally
    2. Storage, Access and distribution of results remotely

- A storage server for student information and transcripts

Below is a class diagram describing the structure of the system and a use case diagram depicting the interactions among the system's elements.

Class Diagram
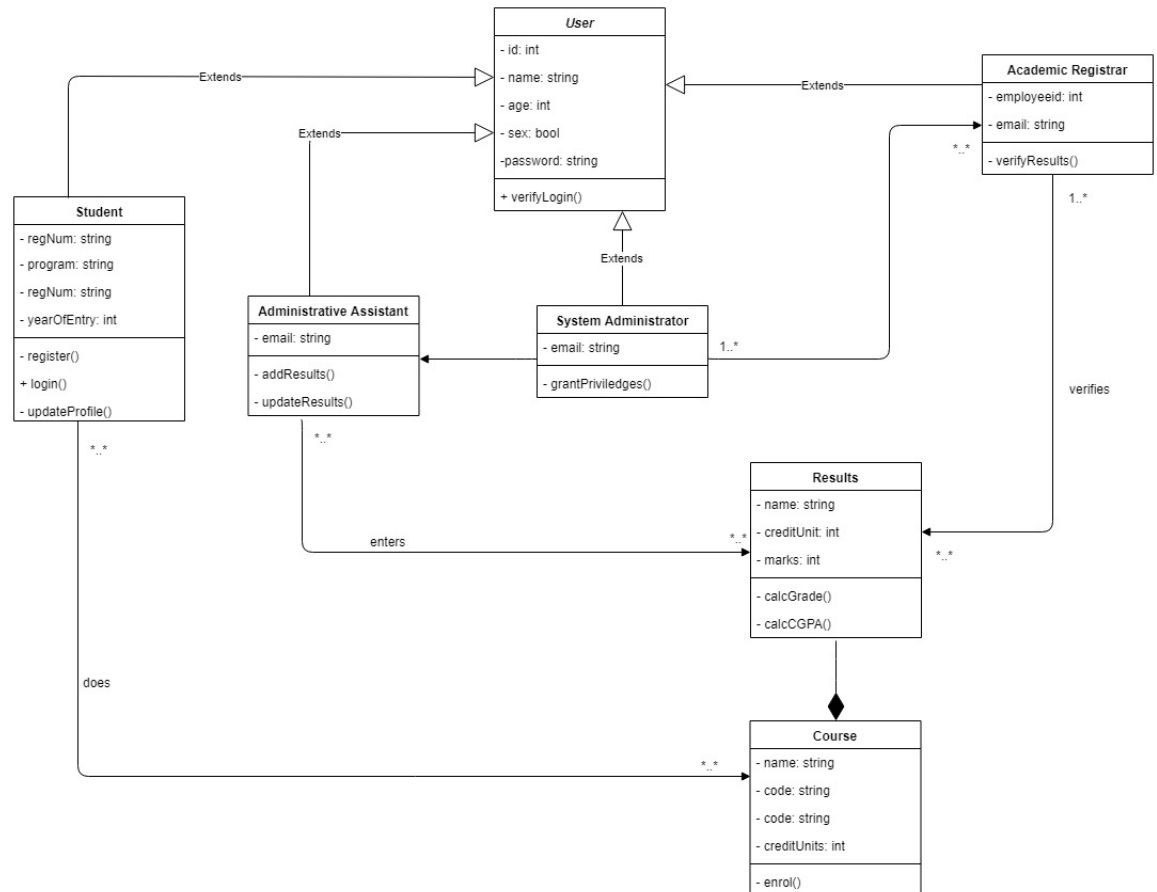
Figure 4.4: Class diagram

Use Case Diagram

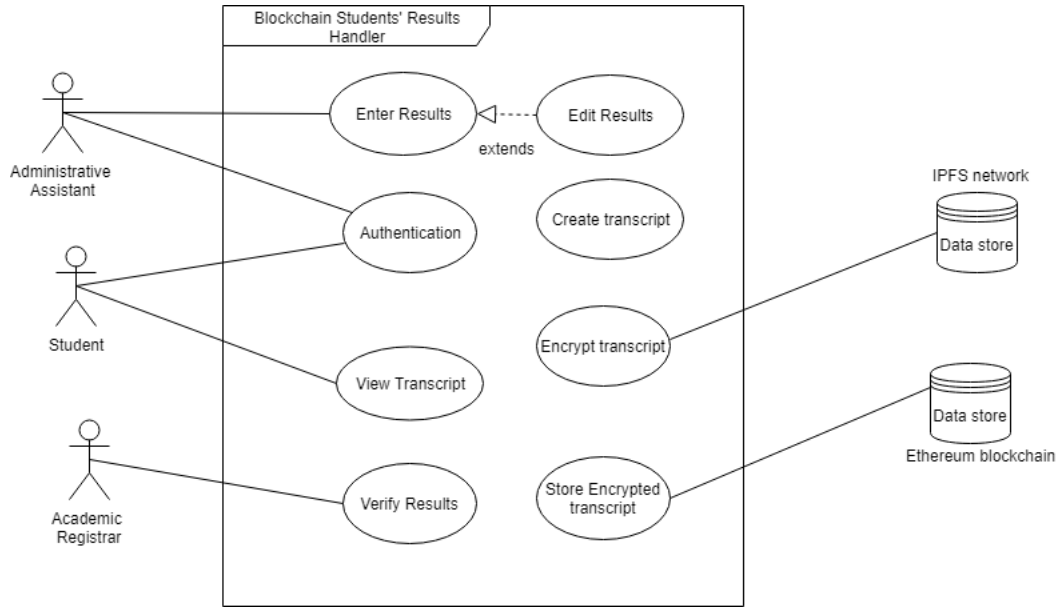

Figure 4.5: Use Case diagram

4.3.2 Design Constraints

The system for storing students' results on the blockchain is dependent on a number of factors, and these constrain its design in one way or another. Here below are some example constraints:

- **Internet Connectivity**
  Since the system runs online, this thereby implies the need for users to have access to the internet.

- **Browser Support**
  Any browser can be used to access the internet. However, accessing the Ethereum blockchain to use a dApp via a browser needs that particular browser to support the MetaMask extension. Currently, Metamask extensions are only available for Chrome, Firefox, Opera and Brave browsers. This constrains the system to these particular browsers, leaving out users that may not have the mentioned browsers.

- **Payment for the blockchain (in Cryptocurrency)**
  Writing to the blockchain costs gas. Gas in the context of Ethereum is a unit and a measurement for the computing power that is needed to execute certain operations in the Ethereum Virtual Machine (EVM). Use of the system would require an Ethereum wallet[21].

- **The Ganache limitation**
  During simulation of the blockchain network locally and implementation of the system using Ganache, the developers were limited to only ten virtual accounts to simulate the entire network of actual accounts.

Assumptions
Some assumptions were made during the design and development of the system. They include the following:

- The end-user is willing to learn and adopt to the technology that may be new to him/her.

- Users have a reliable connection to the internet.

- Users are familiar with common internet browsers and file extensions for these browsers.

- The system uses QR codes for purposes of verification. Therefore, it is assumed that users are familiar with QR codes and have access to a QR code scanner.

- Scalability of the system will not negatively affect the applications speed and reliability.

4.3.3 Design Methodology

We used the Agile software development (ASD) methodology. This methodology involves adaptive planning, evolutionary development, early delivery, and continual improvement. It also encourages rapid and flexible response to change. We focused on keeping code simple and testing often. This helped us to minimize risks such as bugs, cost overruns and the changing requirements.

4.3.4 Graphical User Interface (GUI) Design

This section provides the detailed design of the system and subsystem inputs and outputs relative to the user.

Inputs

- *User Credentials for Authentication:* - Access to the Resuslts system requires User Authentication depending on whether a user is an Administrator or Student.

- *MetaMask Password:* - In order for users to access the blockchain through their browser and via the Metamask, the extension requires a user password. Upon account creation on MetaMask, users are given a secret seed phrase. This seed phrase can be used for password recovery and for first time usage on a different device.

- **Results:** - When an administrative assistant logs into the system, he/she can enter the students results.

Outputs

- **Testimonial:** - This is progressively updated and available on the Results system. Before the results are forwarded to the blockchain, a student can access the system and view his/her progressive results.

- **Transcript:** - Upon completion of study, a students can view their transcripts which can be downloaded as a pdf document for purposes of printing. This pdf is then digitized and encrypted before it is shared and stored on a global network. Students have access to their transcripts from anywhere and the will to share it with employers.

4.3.5   External Interfaces

Hardware Interfaces

The Results System needs access to a local server for storage of testimonials and transcripts locally. These could also be backed up on a cloud server, depending on the institution's particular system.

To access the functionality of this system, a user is required to have a computer with internet connection and a browser that supports the MetaMask extension. These browsers include Google chrome, Mozilla Firefox, among others) with the MetaMask browser extension installed.

Software Interfaces

The functionalities of the external interfaces were developed using web-based scripting langauages like PHP, JavaScript and the smart contract for the Ethereum blockchain written in solidity.

Chapter 5

Presentation of Results

5.1 Introduction

The sections and subsections in this chapter describe the implementation, testing and results of the project. It starts by discussing how the web application to handle students' results was built. It then describes how The decentralized file storage element of the system was put up and how all these elements were integrated together.

5.2 System Implementation

This section and the subsections beneath it describe the Implementation and results of the project. Different elements of the system were built using different tools. The subsections below further describe how each element was built to get the complete system running from the web application that handles the students' results to the decentralized storage of transcripts and block chain store for the hash values.

5.2.1 Results System

The results system is an element of the system where results are initially entered and processed before they are finally stored on a decentralized network. This system is built to be accessed by students, administrative assistants, the academic registrar and the system administrator. All these have different levels of privileges when accessing the system for example, a student is only able to view his results and not edit them.

The first attempt to build this system was done entirely on a block chain network using smart contracts. A smart contract in simple terms can be defined as computer code running on top of a block chain that controls transfer of digital currencies or assets between parties under certain conditions[**?**]. This approach

was not feasible and economically sound because a lot of writes would have to be done to the block chain and this costs ether. Ether is fuel used on the Ethereum block chain network. The latter attempt involved building a web based centralized application that handles the results which are later stored in a decentralized manner. Building this application was broken down into the following parts.

Front End

The front end of the application runs on a web browser like Google Chrome or Mozilla Firefox. These web browsers have a uniform support for various programming languages that are used together to build the web pages we know today. Described below are the programming languages used to build the front end of the results system.

HTML(Hyper Text Mark-up Language): HTML is the standard markup language for creating web pages applications. Using HTML elements as the building blocks these pages, we were able to insert images, interactive forms among other items on the HTML pages.

```html
<form class="form-horizontal" method="post">
    <div class="form-group">
        <label for="inputEmail3" class="col-sm-2 control-label">Email</label>
        <div class="col-sm-10">
            <input type="text" name="username" class="form-control" id="inputEmail3" placeholder="Email">
        </div>
    </div>
    <div class="form-group">
        <label for="inputPassword3" class="col-sm-2 control-label">Password</label>
        <div class="col-sm-10">
            <input type="password" name="password" class="form-control" id="inputPassword3" placeholder="Password">
        </div>
    </div>

    <div class="form-group mt-20">
        <div class="col-sm-offset-2 col-sm-10">

            <button type="submit" name="login" class="btn btn-success btn-labeled pull-right">Sign in
                <span class="btn-label btn-label-right">
                <i class="fa fa-check"></i>
            </span>
        </button>
        </div>
    </div>
</form>
```
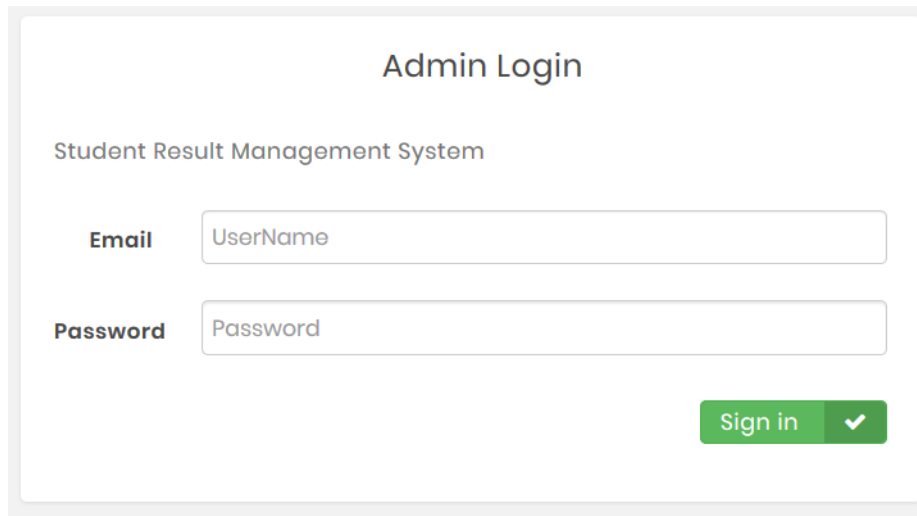
Figure 5.1: Form element used in HTML for an administrative assistant's login.

Figure 5.2: An output for the form in image 5.1 above.

During the development process, the web browser received the the HTML documents from local storage.

CSS(Cascading Styles Sheet): This is a style sheet language used to describe the presentation of a document written in a markup language like HTML in the case of this project. Some of the styling used on the web pages were fetched from CSS files in the bootstrap packages.

CSS was used to add animations to the web pages, style and position different elements in the HTML pages. External CSS was used in the development of the results system. External CSS as opposed to internal CSS, has the styling of the web page done in another file and that file just refereed to by the HTML web pages. This availed the same styling to be applied to multiple HTML files.

```
<link rel="stylesheet" href="css/bootstrap.min.css" media="screen" >
<link rel="stylesheet" href="css/font-awesome.min.css" media="screen" >
<link rel="stylesheet" href="css/animate-css/animate.min.css" media="screen" >
<link rel="stylesheet" href="css/main.css" media="screen" >
```

Figure 5.3: References in the header of an HTML document to style sheets in another location.

JavaScript: JavaScript is a light weight interpreted programming language. For the domain of the results system, JavaScript was majorly used to make the web pages interactive and dynamic. JavaScript used in the construction of the

25

results system was used to give alerts when a user logs in.

There are already existing open sourced web development tools that support the development of web pages. These tools were chosen majorly because they support the programming languages described earlier. Explained below is how these tools were used to achieve the front end of the results system.

Visual Studio Code: This is a light weight text editor that runs remotely on a desktop. VS code used to write code for most parts of the implementation. VS code was installed on local computers of various team members and each configured with a github account to ease the collaboration process. The configuration was done on a command line interface known as the terminal. All the changes pushed to github were done using commands and so were the changes pulled from github.

Twitter bootstrap: This is an open-source HTML, CSS and JavaScript framework that is used in front-end web development. We used this to enhance the appearance of the various pages of the application and also make them responsive.

Github: Github is a web based hosting service for version control using git, mostly used for computer code. We used this to co ordinate work among the various team members. Whenever a team member made changes, they were pushed to github and the other team members would pull the changes on their remote computers.

Browser Developer tools: This is an inbuilt functionality in most web browsers that can be used to inspect the currently-loaded web page. It shows HTML elements, style of the web page and JavaScript file written for the page.

5.2.2   Back end

Up to this part of the project, the students results are still stored in a centralized manner on a server. During the development process, the server was run locally on a Desktop Computer. There are a number of open source local development environments that can be used to develop the back end of a web application for example WampServer, Laragon, XAMPP, among others. XAMPP was used for the case of this project.

XAMPP

XAMPP which stands for Cross-platform, Apache MariaDB, PHP and Perl is a lightweight application that enables developers create a local web server for testing purposes. XAMPP contains key components to set up a web server i.e. Apache server, MariaDB for the database and an interpreter for the scripting language PHP. Explained below in detail are the components of XAMPP and

how they were used to achieve the target of the project.

PHP
PHP is a general purpose scripting language popularly used for web development with over 244 million sites that were using it by the year 2013. For the results system, PHP was majorly used to manipulate database values.

PHP has a number of libraries that can be called to perform specific functions. One of the aims of this project is to ensure safe storage of students' transcripts. These transcripts are generated from the results system as a PDF which is automatically stored on a decentralized network as we'll later see in section 5.2.3. PHP offers a number of libraries to generate PDFs for example FPDF, mPDF, DOMPDF among others. DOMPDF is the library chosen for this project.

MySQL
SQL stands for Structured Queried Language. It is a database language used for database creation, deletion, fetching rows and modifying them. It is one of the components provided in the XAMPP package. MySQL also consists of functions that are available for use when writing PHP scripts. In this project, these functions were majorly used to manipulate values in the database.

Apache Server
Apache is an open-source free web server software powering around 46% of web servers world wide. It is one of the components availed in the XAMPP package able to run locally. To access the web application locally, Apache is ran and the following url, http://127.0.0.1/ is used to access the server. Through out the development of the results system, Apache was run offline.

5.2.3   IPFS integration

IPFS is a protocol that establishes the peer-to-peer network with resources addressed based on their content instead of the physical location. Our system uses this protocol with aim of guaranteeing users a decentralized and unalterable storage at a fraction of the price they would have to give in transaction fees.
The PDF file generated by the Results system implementation above is the input of this next phase of implementation.
We implement this protocol in such a way that the network is not going to store files once you add them. Adding files to IPFS does not upload them anywhere and only means that you add them to the local repository you host on your node. Therefore, what our system does is we create a hashed representation of the file and share this as a reference to the file. An illustration of this can be seen in Appendix B - figures 7.6 and 7.7.

However, during the implementation of this project, we realized that connect-

ing to the Ethereum block chain is messy and complicated. We faced challenges which include:

1. The expense of storing data on the Ethereum block chain

2. Complexity in configuring the Ethereum geth client and

3. Scalability of infrastructure would be tough

Infura

In attempt to overcome the challenges listed above, we used Infura. Infura is a set of tools used to create an application that connects to the Ethereum block chain. It interacts with the Ethereum block chain and runs nodes on behalf of our system. The connection to Infura is illustrated in Appendix B - figure 7.8.

Using Infura's API made our system fast, scalable and provides extra data storage. However, despite the fact that Infura offers to do that work for us, it brings with it the cost of increased centralization. The output of this subsystem is a hashed URL which is a reference to our PDF file.

5.2.4 Ethereum block chain

As mentioned in the above section, using Infura and IPFS provides extra data storage for us. This is done is such a way that instead of storing the entire file on-chain, data can be stored separately i.e locally on a server from the results system in section 5.2.1, with just a hash stored on the block chain.
By storing the hash on the Ethereum block chain,the system takes advantage of the block chain principles of Immutability, Disinter mediation and Transparency.

Results

Viewing Results

This page can be accessed by a logged in student or an administrative assistant. The student can only view his/her results whereas the administrative assistant can view the results of various students. **Manipulating results**

In addtion to viewing results of various students, the administrative assistant can also add/edit students results.

5.3 Tests

5.3.1 Introduction

The purpose of this section is to present the test cases used to test and validate the functionality of the Students' Results system's features.

5.3.2 Scope

To validate all the known functions of the Students' Results handler.

5.3.3 Requirements

The requirements for the tests are listed below, as they were derived and given in Chapter 4 of this report.

User Requirements

U.1 Allow an administrative assistant enter students results.

U.2 Allow students to access their results without being able to change them.

U.3 Provide for students to easily share their results information with employers.

U.4 Allow employers to easily verify students' results.

Functional Requirements

F.1 Ensure user authentication through log in.

F.2 Reject addition of inaccurate results.

F.3 Ensure that the students results are not altered.

F.4 Ensure students' transcript can be saved locally.

F.5 Uploading transcript to IPFS.

F.6 Saving IPFS hash to the block chain.

Non-Functional Requirements

N.1 The system must verify any addition to the block chain.

N.2 The system must notify the system administrator in case of any unauthorized / failed transactions.

5.3.4 Test Cases

Test	Test Case	Pre-condition	Steps	Expected Output	Resu
USER REQUIREMENTS					
U.1	Allow Administrator assistant to enter student's results	Administrator must be logged in, Student Must be registered in system	Select Course, Select Year, Enter Student Name, Enter results	Results added to the database Notification for successfully entering results	Pass
U.2	Allow students to access their results without being able to change them.	Student must be registered. Student must be logged in	Enters Student ID, View Results	Results displayed if student ID entered is valid, Display an alert if Student ID is invalid	Pass
U.3	Provide for students to easily share their transcripts with employers.	Student transcript is uploaded to IPFS	Create a URL for the transcript	Access the Transcript via Internet	Pass
FUNCTIONAL REQUIREMENTS					
F.1	Ensure user authentication through log in, Case 1 - correct name and password, Case 2 - correct name and incorrect password, Case 3 - incorrect name and correct password, Case 4 - both name and password incorrect	User must be registered	Launch the app, Enter name, Enter password, Click Submit	Case 1 - User proceeds to dashboard, Cases 2, 3, 4 - Alert user and prevent log in	Pass
F.2	Reject addition of inaccurate results, Case - Add mark >100%	User must be administrator	Log in, Select Student, Select Add Result	Alert user of incorrect mark entered	Pass
F.3	Ensure that the students results are not altered, Case - Attempt to alter results	User must be Student	Log in, Try accessing database	Database should not be accessible to student	Pass
F.4	Ensure students' transcript can be saved locally, Case - Downloading PDF format of transcript	User must be logged in	Log In. Click Download button	Should download a PDF document	Pass

Table 5.1 continued from previous page

Test	Test Case Description	Pre-condition	Test Procedure	Expected Output	Result
F.5	Uploading transcript to IPFS.	User must be administrator, Transcript must be in PDF format	Save result as PDF, Ensure internet connection, Click Upload to IPFS	Create IPFS hash for the PDF and display transcript's URL	Pass
F.6	Saving IPFS hash to the block chain.	User must be administrator, User must be having an Ethereum account and etherem wallet, Browser must be having MetaMask	Launch app, Accept MetaMask connection request, Click to Save hash to block chain, Accept MetaMask transaction request	Hash upload to Ethereum, Ethereum gas deduction - Wallet updated	Fail
NON FUNCTIONAL REQUIREMENTS					
N.1	The system must verify any addition to the block chain.	User clicks upload to block chain		Notify user using metamask transaction notification	Pass
N.2	The system must notify the system administrator in case of any unauthorized / failed transactions.	Low Account Balance of Ethereum wallet	User clicks upload to block chain	notify user through MetaMask notification	Pass

Table 5.1: Table showing test cases for the system

5.3.5 Test Resuslts

The table below summarizes the results of the tests performed on the system to check its functionality.

System Name:	Students' Results Handler with Storage on block chain		
Test Results:			
Pass:	13	Pending:	0
Fail:	1	Number of Test Cases:	14

Table 5.2: Summary of Test Results

Chapter 6

Recommendations, Conclusion and Future Works

6.0.1 Conclusions

In conclusion, the major tasks set at the beginning of the project have been completed. All the elements of the students results' storage system have been built and put together to achieve the desired goal. Below, we summarize the progress of the project with respect to the specific objectives.

The first specific objective is to ensure safety of students' records. The results storage system stores the students transcript on decentralized network and can only be accessed by those who have the hash value for that transcript. Furthermore, the hash value for the transcript is stored on a blockchain network which has proven immutability characteristics.

The other objective is to create an open distributed catalog that makes it easy for students to share their transcripts. This system stores the transcript on a distributed network using a protocol known as IPFS. With this, the student can still share his/her transcript without worry of insecurity.

The third objective is to provide a medium for verification that can be used by employers in telling true transcripts from counterfeits. With a QR code appearing on the transcript, an employer is able to scan the document and verify its authenticity.

Considering a few days between the deadline of this report and the one of the presentation, certain aspects will be improved upon. These include increased level of detail and well designed poster for the presentation.

Limitations

During the research and lifetime of the project, a number of limitations were encountered. These were not just on the project but a possibility of the success of a decentralized method of results storage. These limitations are further explained below.

Limited Accounts: During the developments process, we only had 10 blockchain accounts to use while running tests. This greatly limited us from testing the ultimate potential of a distributed network.

Complexity: Blockchain is a new technology with a lot of new vocabulary. Although it has given room for cryptography to go mainstream, it is no place for beginners.

Since this results storage system is based on blockchain, a limitation to this is that many students at Makerere University are not well versed with the present day system that stores their results. Presenting blockchain to the university community could pose a greater challenge. Thankfully, there are a number of courses on blockchain out there today.

Transaction Costs: In the early stages of a decentralized network, transactions are usually faster due to the small number of nodes. As the number of nodes grow, the network grows along. Using the networks becomes more costly in terns of resources.

With time, a distributed results storage system like this one would grow. More students and employees would join the network over time. However, a major characteristic of blockchain is its scalability. This would be a good remedy for such a challenge.

6.0.2   Recommendations

With the what we have learned during the research process, we believe certain changes would be helpful for the success of such future projects.

- We recommend the university to create post project mentor-ship program. This would encourage students to focus on their enhancing their skills past their time of study.

- We recommend the College of Computing and Information Sciences to prepare students from the time they take on the course for their final year project. We believe this would reduce the pressures of doing the project alongside other course units in two semester.

6.0.3   Future Works

Many tests and experiments for this project have been left out due to limited availability of time and resources. Some of these include running tests on a real blockchain network and not just a local environment.

The scope of this project has been on building a results storage system with focus on one institution (Makerere University). Future works on this project will aim at expanding the storage network to more than one tertiary institution. Furthermore, the idea is to expand storage of students academic record right from the time they begin school to the time of completion.

Chapter 7

Appendices

7.1   Appendix A: Tools Used
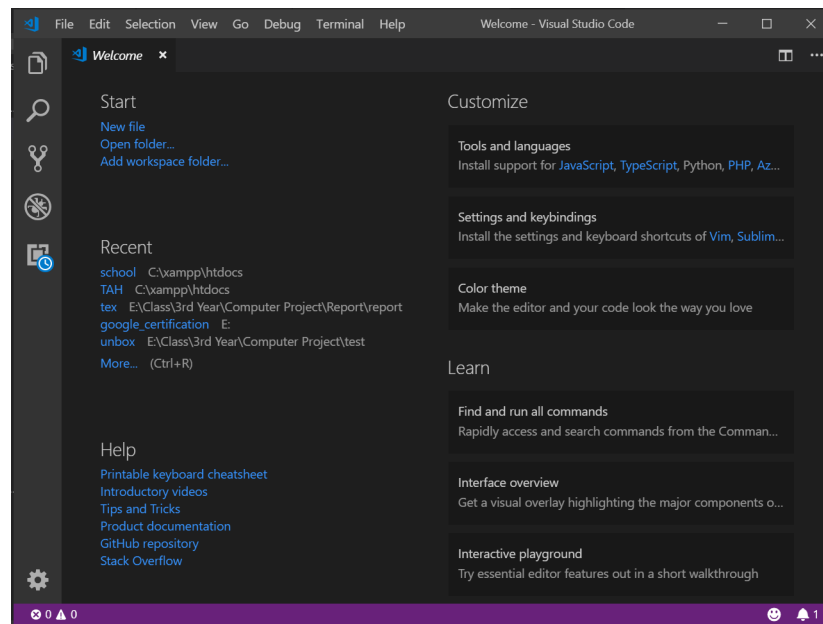
Figure 7.1: Visual Studio Code, the text editor used to write code for the system.
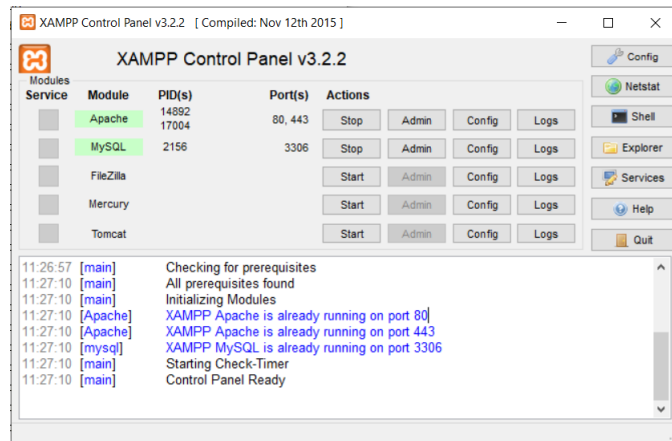
Figure 7.2: The control panel for the XAMPP local development environment.
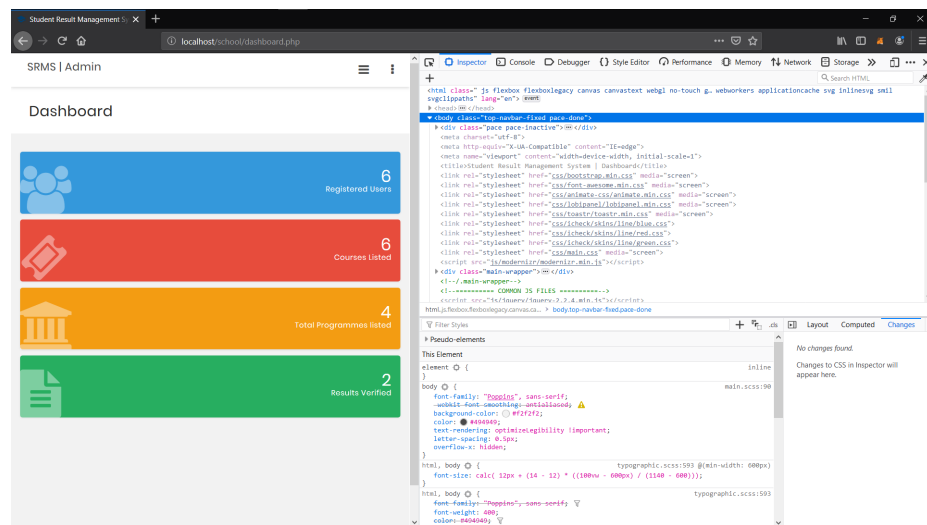
Figure 7.3: Developer tools opened in chrome to debug the Dashboard.

Figure 7.4: Google document for Stand Up meetings and follow up

Figure 7.5: Accounts in Ganache: A local instance of the Ethereum blockchain

7.2 Appendix B: Samples from the Implementation Process

```
57
58    // we get the file and convert it to format that IPFS can understand
59    captureFile = (event) => {
60      event.preventDefault()
61      const file = event.target.files[0];
62      const reader = new window.FileReader();
63      reader.readAsArrayBuffer(file);
64      reader.onloadend = () => {
65        // get array of data that we can pass to the buffer
66        // this gives us a format that we can post to IPFS
67        this.setState({ buffer: Buffer(reader.result) });
68        console.log("buffer", this.state.buffer);
69      }
```

Figure 7.6: Conversion of file to an IPFS-valid format

```
72
73    // here we want to add the buffer file ti IPFS
74    onSubmit = (event) => {
75      event.preventDefault();
76      ipfs.files.add(this.state.buffer, (error, result) => {
77        if (error) {
78          console.log(error);
79          return
80        }
81        this.setState({ ipfsHash: result[0].hash });
82        this.setState({ hashedPDF: `https://ipfs.io/ipfs/${this.state.ipfsHash}` });
83        console.log("ipfsHash", this.state.ipfsHash);
84      });
85    }
```

Figure 7.7: Conversion of file to an IPFS-valid format

```
1  /* to connect to ipfs, we will create an instance
2     of the api and connect to a remote node (via infura.io)
3     instead of creating our own local node
4  */
5
6  const IPFS = require('ipfs-api');
7  const ipfs = new IPFS({ host: 'ipfs.infura.io', port: 5001, protocol: 'https'});
8
9  export default ipfs;
```

Figure 7.8: Accessing IPFs public node using the Infura API
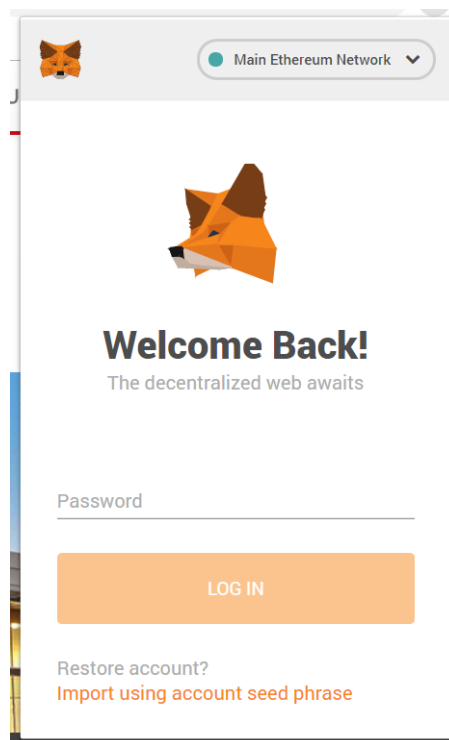
7.3   Appendix C: Images of the Application
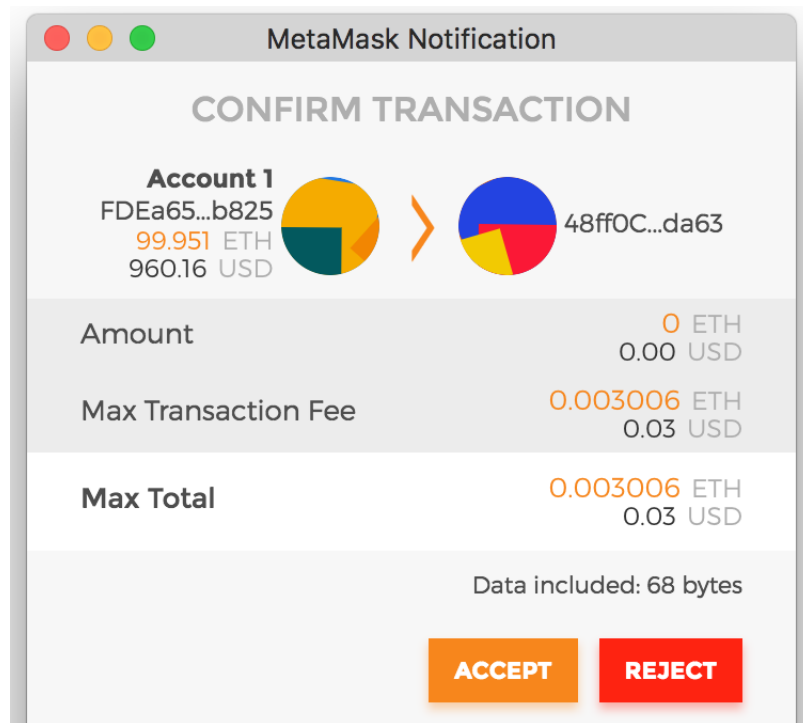

Figure 7.9: Logging in to the ethereum network

39

Figure 7.10: A transaction awaiting confirmation from the user.
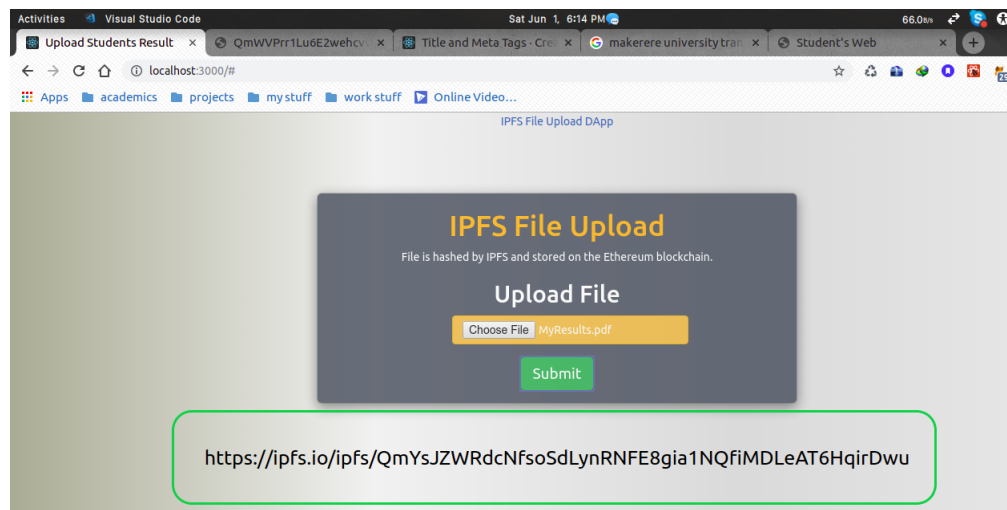
Figure 7.11: URL is generated using a hash of the Transcript and it is uploaded to the public node

Figure 7.12: Using the generated URL, the transcript can be accessed from the internet

Bibliography

[1] Margaret Rouse, TechTarget(2018, September). *Cryptography*[article],
    Available: https://searchsecurity.techtarget.com/definition/cryptography

[2] Records Keeper(2018). *Verify Academic Certifications*[article],
    Available: https://www.recordskeeper.co/verify-academic-certifications/

[3] *The scourge of fake academic papers*[article],
    Available: https://www.monitor.co.ug/News/Education/jobs-dark-world-
    fake-academic-documents/688336-4784694-bvjkc7z/index.html

[4] *High youth unemployment persists despite entrepreneurship train-
    ing*[article],
    Available: https://www.monitor.co.ug/SpecialReports/High-youth-
    unemployment-persists-entrepreneurship-training/688342-4644926-
    563r8fz/index.html

[5] *Uganda Literacy Rate*[article],
    Available: https://knoema.com/atlas/Uganda/topics/Education/Literacy/Adult-
    literacy-rate

[6] *Blocksign*
    Available: https://blocksign.com/

[7] *Siacoin*
    Available: https://sia.tech/

[8] *What is Storj? - Beginners Guide*[article],
    Available: https://coincentral.com/storj-beginners-guide/

[9] *Ethereum Project Blockchain platform to run smart contracts as they are
    coded to be*[article],
    Available: https://www.ethereum.org

[10] *Cryptyk*
    Available: https://www.cryptyk.io/

[11] Luke Graham, CNBC News (2016, May). *Schools are using bitcoin
    technology to track students:*[article],

Available: https://www.cnbc.com/2016/05/09/schools-are-recording-students-results-on-the-blockchain.html

[12] Donald Clark, OEB Global (2016, September. 12). *10 ways Blockchain could be used in education*[article],
Available: https://oeb.global/oeb-insights/10-ways-blockchain-could-be-used-in-education

[13] *Nonverbal Communication*[article],
Available: https://www.helpguide.org/articles/relationships-communication/nonverbal-communication.htm/

[14] *Google Sheets Daily Stand-Up Meeting Notes*[article],
Available: https://docs.google.com/document/d/1eJhi8pN2b1Nc29sU-jxtM-PKZ6UtxqFijfrP1fr14F0/edit?usp=sharing

[15] *Hash function*[article],
Available: https://en.wikipedia.org/wiki/Hash_function/

[16] *InterPlanetary File System*,
Available: https://ipfs.io

[17] *Ganache A one click blockchain*[article],
Available: https://truffleframework.com/ganache

[18] *Decentralized application*[article],
Available: https://en.wikipedia.org/wiki/Decentralized_application

[19] *What is ether?*[article],
Available: https://bitcoinmagazine.com/guides/what-ether/

[20] *Directed acyclic graph*[article],
Available: https://en.wikipedia.org/wiki/Directed_acyclic_graph

[21] *Ethereum Wallet - What is a wallet, and which one should I use?*[article],
Available: https://www.ethereum.org/use/