SYSTEM DESIGN OF CERTIFIED DIGITAL DOCUMENT SHARING PLATFORM USING BLOCKCHAIN

Priscilla Jofani Oetomo

STMIK LIKMI, Bandung, West Java, Indonesia

jo.pris@yahoo.com

ABSTRACT

This paper aims to build the design of a secure and trustworthy platform of digital document sharing system, where students may share their officially obtained educational documents, e.g. Certificates or Academic Transcripts, with potential employers. The system was designed on top of the Ethereum Blockchain, coupled with IPFS as the document storage system. The design has been arranged in such a way as to work as a nationwide solution.

Keywords: Blockchain, Certified Digital Document, Ethereum, Trust

1. INTRODUCTION

This paper will describe an early system design of a sharing platform system, where students may share their officially obtained certified digital educational documents with potential employers. Taking security in mind as the main priority, the system is designed on top of the very safe blockchain technology, while also making use of the content-addressed storage IPFS as the document storage system. This system uses Ethereum blockchain, which is a programmable blockchain technology, for its foundations.

The following items will be included in this paper:

a. Analysis of existing document sharing platform systems.
b. Design of an improved document sharing platform system.

And the following will be excluded from this paper:

a. The prototype of the document sharing platform system.
b. The end-product of the document sharing platform system.

In order to design a working system, an extensive amount of research needs to be carried out beforehand. This paper uses materials from other research papers, as well as observation of existing systems.

2. RELATED WORKS

As written in [1], some educational document sharing platforms have already been previously developed. Blockcert [2] is an open-source system developed as an incubation project by MIT Media Lab Learning Initiative [3] and Learning Machine [4], where people may create, share, and verify certified educational documents. TrueRec [5], a system developed by SAP is also a document sharing system, that only stores the hash of the certificate, and not the certificate itself. It is also an open source system, running on Ethereum. Gradubique [6], designed by Thinh Nguyen, is yet another example, and with its paper being published online, it is an even clearer picture of the design of a document sharing platform system using blockchain.

Other than the aforementioned systems, another system that the author is most familiar with, is called My eQuals [7], which the author has had an opportunity to experience first-

hand, as a student. It is a system based for Australian and New Zealand universities, and is possibly the closest to providing a nation-wide system. Unfortunately, there are still a lot of educational providers in Australia and New Zealand who have not been registered with My eQuals yet.

3. METHODOLOGY

3.1. CERTIFIED DIGITAL DOCUMENT

A digital document is an electronic copy of an academic transcript, degree certificate, or other kinds of documents. In its most basic form, it is built up out of the same components as what is usually written on a paper-based document. According to [8], its components include the certified user/entity's name, the user/entity's public key, the certification authority's name, and a digital signature of the document. Stored inside a secure system, a digital document might even be more trustworthy than a paper-based one. It is not easy to replicate a digital document using the signature of the issuing organisation or institute unless they could somehow hack into the system or log in using the organisation's account. Even if someone were to use a fake digital document, the system provides a simple verification feature to check whether the ID exists and is valid.

According to [9], there are several limitations to both paper-based and digital certificates, which are summarized in Table 1.

Table 1
Comparison between paper-based, non-blockchain and blockchain-based certificates

Category	Paper-Based Certificate	Non-Blockchain Digital Certificate	Blockchain-Based Digital Certificate
Forgery	Security measures make it difficult to forge.	Easy to forge without digital signature.	Almost impossible.
Storing Method and Protection	Relatively easy to store and protect.	More effort in storing. Depending on the security measure and backup systems, might also be easier to steal or destroy.	Almost impossible to steal because blockchain is very secure. Almost impossible to destroy because the only way to completely destroy a record is to destroy every copy of it in each and every one of the computers connected to the system.
Control	Awardee has full control.	Only issuer can modify or revoke (depending on the system design). Awardee has full control for sharing.	Only issuer can modify or revoke (depending on the system design). Awardee has full control for sharing.
Verification Process	Verification carried out by the issuer with the help of registry (single point of failure). Would still exist and be valid	Verification carried out by the issuer with the help of registry (single point of failure). Would not be valid without	Open to public verification method, independent from organisation. Would still be valid even when

	even without registry.	registry.	the issuer no longer exists.

Figure 1 displays the steps involved in the generation process of a secure digital certificate, by the user "Alice". Firstly, the document to be signed will go through a hashing algorithm in order to generate its unique digital fingerprint. It is important for the system to use a strong hashing algorithm, like SHA-256 to make sure that it is collision-free and secure. Next, the resulting hash value will then be signed using the private key of the signer, in this case, "Alice". In the example, the hash value is signed using an RSA algorithm, and it is by far the most common algorithm used to sign digital documents [10].

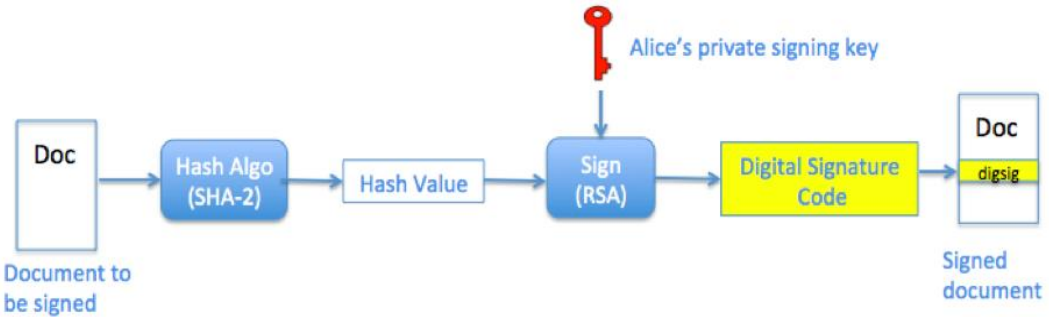

Figure 1
Generating a Certified Digital Document [10]

Without being verified, a digital certificate loses its value. Figure 2 below illustrates the flow of the verification process of a digital document. The first step is to run the original document through a hashing algorithm to get its hash value. Then, the verifier will verify the signed document using the public key of the original signer, in this case, is the user "Alice". The purpose of this verification process is to extract the signature from the signed document, which involves reversing the signing process and, in turn, will produce the document's hash value. The last step involves matching the hash value from the first and second steps. The document is considered to be verified if both hash values match.

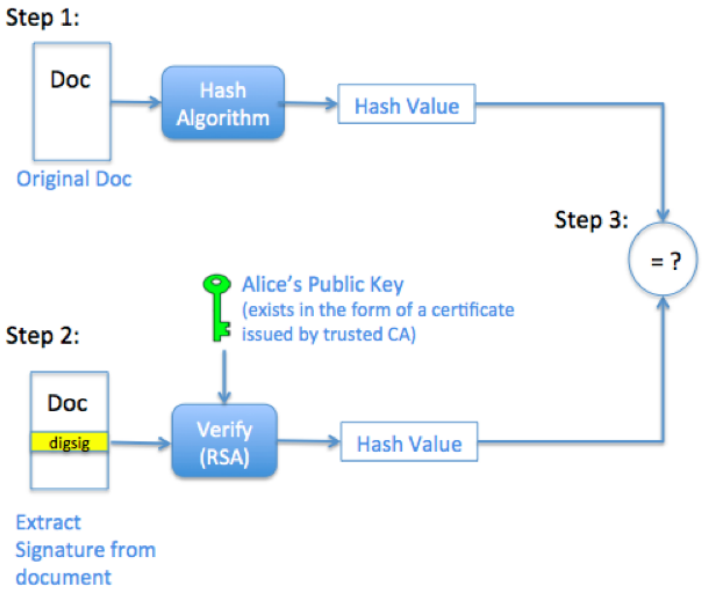

Figure 2
Verifying a Certified Digital Document [10]

This works, because the verification process uses the original signer's public key, which is a match of the private key used to sign the document in the first place. When the hash

values produced a match, it means that the signed document must have been signed by a matching private key of the public key used in the verification process [10].

3.2. BLOCKCHAIN SHARING SYSTEM

To ensure the integrity, consistency, security and trustworthiness of the sharing system, blockchain was chosen as the foundation. For this type of system, many, if not all, preceding systems have used consortium access type, in which there exist several roles with different accesses. Typically, the roles in those systems are:

a. Highest Authority: those who can register the issuing organisation. Usually the government.
b. Issuing organisation: those who may issue, update and revoke the documents.
c. Students: the owners/awardee of said documents. May share the documents, but may not issue, update or revoke.

The roles listed above were summarised from several sharing systems, however it is merely a guideline, and may be modified as needed to suit the design of the system. Aside from that, there also exist a general pattern of steps from the start of the issuing process, until the end of the sharing process carried out by the students, which are listed below [9].

a. Issuing:
   Carried out by the issuing organisation, in this case, the university. It is where the issuing organisation uploads a student's document into the system.
b. Verification:
   The verification of the document, i.e. whether the organisation really did issue the document. This process is carried out by the system, but will directly involve the organisation. One of the techniques commonly used is to use a security feature embedded into the document itself, for example by verifying the digital signature of the issuing organisation.
c. Sharing:
   The sharing of the document by the student. After the verification process, the student will be given access to the issued document. The students will be able to share the documents to third parties and they may easily backtrace and verify the document on the trusted sharing system.

Moreover, the author of [9] mentioned that there are several ways to create a trustworthy sharing system:

a. Method for Identity-Verification:

There needs to be a way to verify the identity of the people/organisations involved, as this system requires communications between multiple parties (issuer, certificate holder, a third-party whose access was gained through the link shared by the certificate holder) According to [9], identity-verifications are usually done either by having another certificate that can prove the validity of the holder's identity or by involving a third-party to process the verification.

b. Standardised Processes for Issue & Certification:

To create a more trustworthy system, there needs to be a standard when uploading a document. The most common and simplest method of standardisation would be a series of criteria, where the documents may only be issued after the certificate holder has met said criteria. A high and strict standard must be employed as there is not a single organisation in charge of the system, it is used and maintained together by all of its users.

c. Mechanisms for Regulation and Assurance:

No matter how secure a system may be, there will always be some parts of the system the users can control, and that makes it less secure. Often times, we may find some undisciplined users that would intentionally or unintentionally disturb the stability of the system. When this happens, there needs to be a way to report and remove such user, to maintain the level of trustworthy.

d. Security Features

The level of trustworthy must also be maintained during document upload processes. There must be a way to identify the validity of the document, including but not limited to, whether the document was valid and not forged, and whether the issuing organisation really did issue the document. The first one may be solved by having a kind of signature or ID that can only be produced by the issuer. The second one may be solved by having a database maintained by the issuer or an open centralised database, called a registry, where the public may be able to check whether the document really was issued by the claimed issuer.

e. Accessibility

Lastly, the documents and the verification process must be easily accessible to those who need them. These involve the certificate holder and the third-party whom the holder share a document with. The documents themselves must also contain easy-to-follow instruction on to how to verify them. They must also be highly legible and easy to use.

4. RESULTS AND DISCUSSIONS

The result of this project is a system design of the certified digital document sharing platform. The designs that are included in this paper will be in the form of Use Case Diagram, Architecture Diagram, and Wireframes. Firstly, we will look at the Use Case Diagram, that serves as the earliest design. Secondly, the Architecture Diagram will further describe the general system as well as how each feature works. Lastly, the Wireframes will give the general idea of the user interface of the system. This paper will only include the most important features, focusing only on the steps needed to issue, verify and share the document.

4.1. USE CASE DIAGRAM

The Use Case Diagram in Figure 3 lists out the features and the actors of the system being designed in this paper, which will be explained in detail down below.

a. Login

The login feature is accessible by all roles, except for the Public role. This is essentially a normal login system using username-password combination, the only difference is that the system does not have an easy to access register feature, meaning that registrations can only be done through invitations as it is not freely and publicly available to everyone.

b. Register University

This is a feature accessible to the Government user, which will be held by Indonesia's official government agents. This is where the Government can register the Universities into the system. The request to join the system must be sent through external means (e.g. through a formal email) from the University to the Government. After the registration has been requested by the Government, the University will receive an email containing a one-time-link, which will take them to the login page, where they may set up a password for future logins. The link will expire in 24 hours.

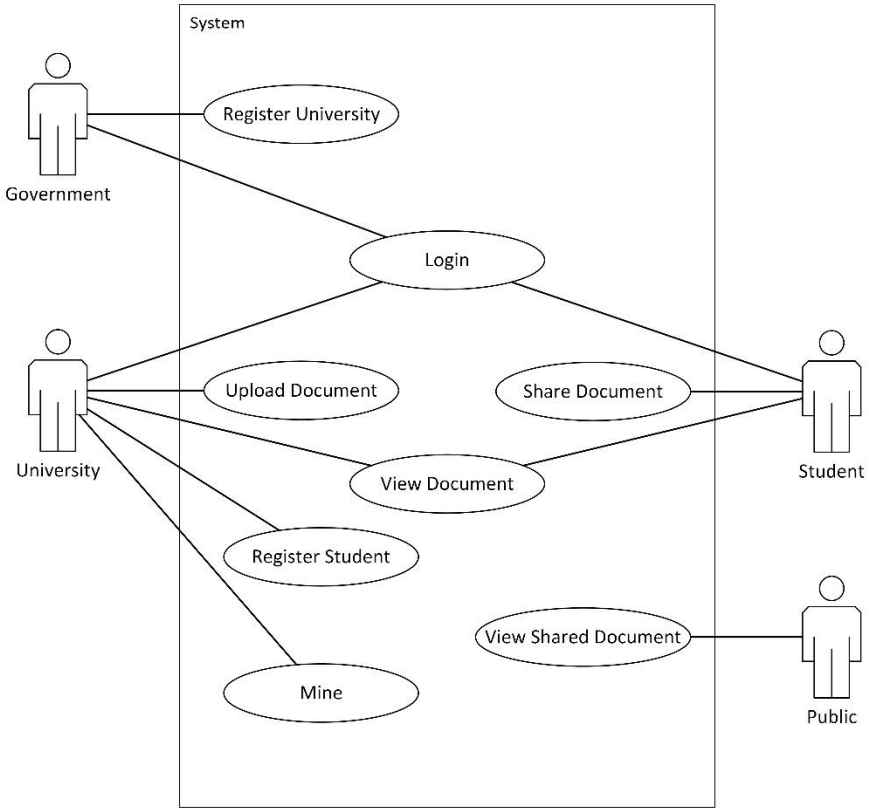

Figure 3
Use Case Diagram

c. Register Student

This is a feature accessible to the University role. The university is responsible in registering all of their enrolled students into the system so that they may obtain a login access. Just like the register university feature, after the registration has been requested by the University, the Student will receive an email containing a one-time-link, which will take them to the login page, where they may set up a password for future logins. The link will expire in 24 hours.

d. Upload Document

This is a feature accessible to the University role. Here is where the university may upload the certified documents of their students and assign it to the corresponding student. The uploaded document will be stored in IPFS. This feature will incur a fee in ETH, Ethereum native currency.

e. View Document

This feature is accessible to the University and Student roles. In order to view a document, the user must be involved in the document itself, that is, they may either be the issuing university, or the document owner.

f. Share Document

This feature is accessible to the Student role. In order to share a document, the user must be the document owner. The sharing feature will generate a link that will take anyone with the link, straight to the view shared document feature. The document owner may remove the link when it is no longer needed.

g. Link Ethereum Wallet

This feature is accessible to the University role. As explained above, uploading a document requires a fee to be paid using ETH. Thus, a feature to link an Ethereum Wallet exists. This system does not come with an Ethereum mining feature.

h. View Shared Document

This feature is not limited to any role. Anyone with the link to a document may be able to view the corresponding document, whether they are logged in or unauthorised. The link may be removed at any time by the document owner.

4.2. ARCHITECTURE DIAGRAM

After determining the features to include in the system, we may then proceed to the design process. The system itself will make use of some already existing open-source applications, rather than creating them all anew. The reason for this being, that the existing applications have already been maximised in terms of performance and security. For the sake of satisfying the security requirements that this system must possess, it is better to use those tested open-source applications. Figure 4 below describes generally how each component in the system interacts. A more detailed explanation below.

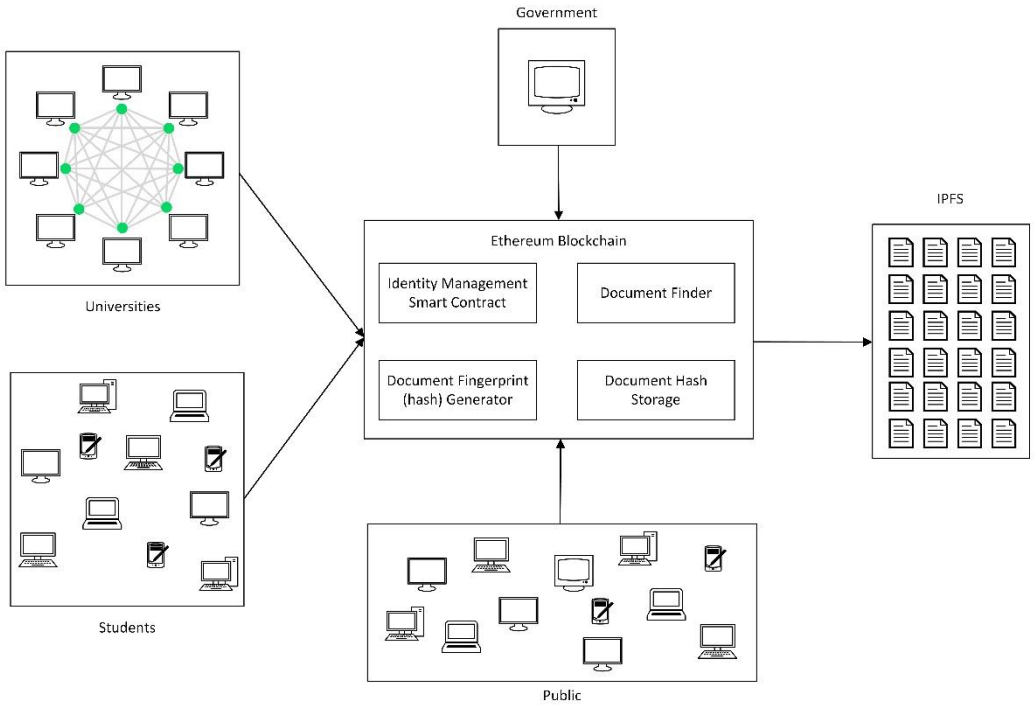

Figure 4
General Architecture Design of the System

a. Government

In this system, the Government role is held by a single entity and their task is solely to register the Universities who wanted to join the system. Figure 5 illustrates the University Registration process. Firstly, the University will request an account registration through means other than this system (e.g. through a formal email) to the Government. Secondly, The Government will then respond by registering the University into the system, by inputting their data into the system. After submitting the form, the system will send an email to the University with a link to complete the registration process, which will expire in 24 hours. The University will then have to open the link and create a password for their account.

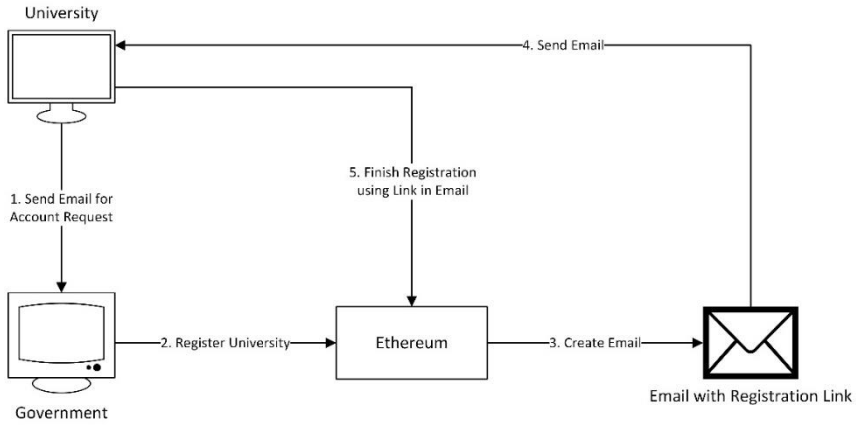

Figure 5
Register University Illustration

b. Universities: Register Student Feature

Figure 6 illustrates the Student Registration feature. Firstly, the University will register their enrolled Students, using the data they have on them. The Student Registration feature needs far less data compared to the University Registration, due to there being a need for the University to hold their end of responsibility as the issuer of documents, while the Students are only at the receiving ends. After the registration process, the system will generate an email with a registration link attached, to be sent to the Student, which will expire in 24 hours. Next, the Student will have to open the link and create a password for their account, and the account creation is complete.

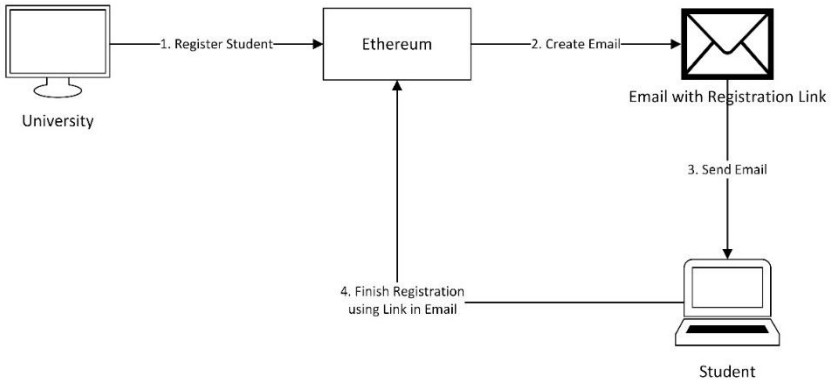

Figure 6
Register Student Illustration

c. Universities: Upload Document Feature

Figure 7 illustrates the Upload Document feature. Firstly, the issuing University will upload the document into IPFS through the system. In order to upload a document, the University must pay a certain fee. Next, IPFS will return the document's hash value, after it has been successfully uploaded. The two following processes are essentially a smart contract process. The document's hash value returned by IPFS will then be signed with the issuing University's Private Key. The signed document will then go through a Hash Function to obtain its hash value. Lastly, the Signed Document's hash value will be stored in Ethereum for tracking and lookup purposes.

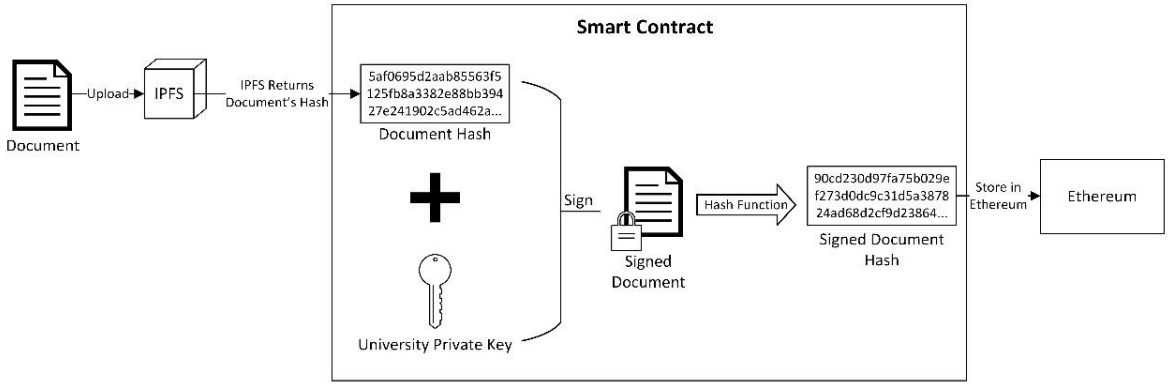

Figure 7
Upload Document Illustration

d. View Document Feature by University Role

Firstly, the University will make a request to view the documents. The system will look through for any documents that have been issued by the University, and for every document reference it finds, a request to IPFS for all said documents' hashes will be made. The IPFS will return all the documents, which will then be displayed to the user.

e. Link Ethereum Wallet Feature

Firstly, the University will make a request to link their Ethereum Wallet into the system. The system will then make a request to an installed Wallet Manager Browser Plugin (e.g. MetaMask). Had the request succeeded, the link will get created and now the University will be able to pay using their connected Ethereum Wallet.

f. View Document Feature by Student Role

Firstly, the Student will make a request to view the documents that have been issued to them. The system will look for said documents. For every document it finds, a request to IPFS for all said documents' hashes will be made. The IPFS will return said documents, which will then be displayed to the user.

g. Share Document Feature

Firstly, the Student will make a request to share the document that have been issued to them. The system will then generate a link for anyone to access the file, store it in Ethereum, and return it to the Student.

h. Public

The Public is a role for everyone, whether they are logged in or not. The Public only has access to one feature, which is to view the previously shared document. Figure 8 illustrates the View Shared Document. Firstly, the Public user will make a request to view a document. The system will then search for the document in question, and then make a get request call to IPFS to retrieve the document. IPFS will return the document, which will then be displayed, along with its general information, to the requesting user. The general information will include the issuing university, student name, and document type.

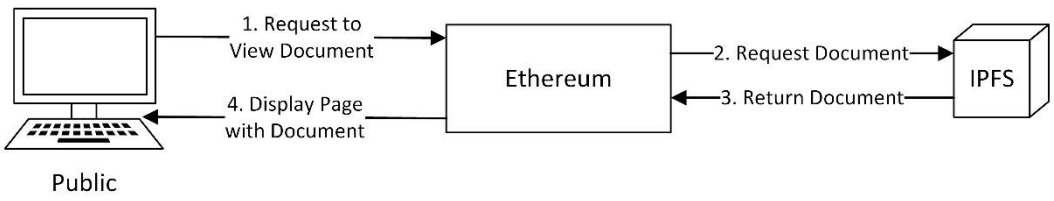

Figure 8
View Shared Document Illustration

i. Ethereum Blockchain: Identity Management Smart Contract

This feature will use an open-source application. The reason being: there have been many secure and tested open-source Identity Management tools readily available. Building a new application will not guarantee its safety, while it is one of the most important features the system must employ. The application that will be used have not been determined yet, but some applications that may be used include uPort [11] and Sovrin [12].

j. Ethereum Blockchain: Document Fingerprint (hash) Generator

A document fingerprint generator is used to store the information of an uploaded document. The document must be securely signed and for this, a Smart Contract will be used.

k. Ethereum Blockchain: Document Finder

A document finder is used when users make requests to view some documents. The document finder will look through the system to find the stored document information, grab the hashed code without the signature, and make a get request to IPFS to retrieve it.

l. Ethereum Blockchain: Document Hash Storage

A document hash storage is essentially a feature to store the information of uploaded documents. The uploaded documents will be signed in the system and information regarding its issuing university, issued student, document type, document name and issued date, as well as the signed document's hash itself will be stored. The storage of such information, which is illustrated in Figure 9, will be done as follows. Firstly, a string of text with semi-colon separated (CSV) format of the information will be created. Next, the text will be converted into hex, as blockchain will only store hex values. Lastly, it will be inserted into Ethereum.

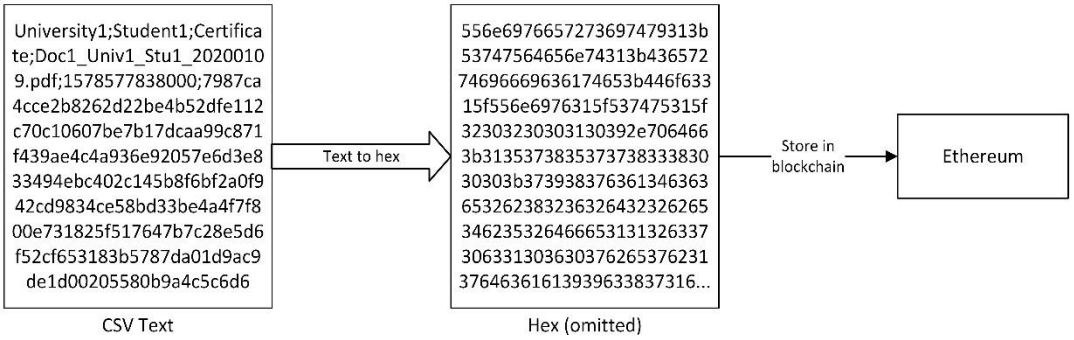

Figure 9
Document Hash Storage Illustration

m. IPFS

IPFS is the storage that this system will use to store all the documents. Being a content-addressed storage, IPFS is perfect for blockchain integration. It will be one of the most-used application in this system, as the system deals with document related features. A document upload to IPFS will always return its hash value, so that it will be able to be inserted and tracked inside the blockchain the system is built upon.

4.3. WIREFRAMES

A prototype of the system is outside of the scope of this paper. However, the wireframe is part of the system design, so that the readers may easily picture what the finished product may look like. The wireframes in this paper uses many placeholder texts, the most often encountered out of all, would be the system's name and logo.

As a disclaimer, the author states that all placeholder is completely made up as a simulation to what it would look like if the system will ever be developed. Should there be any copyrighted material found after this paper is published, the author was not aware of it at the time of writing.

a. Login Page

This is the first page anyone can access before logging in. All page requests made by an unauthorised user (except for pages that are opened to public and does not need to be authorised) will be redirected to the login page. Figure 10 shows that the login page consists of only the elements that are commonly found on other websites.

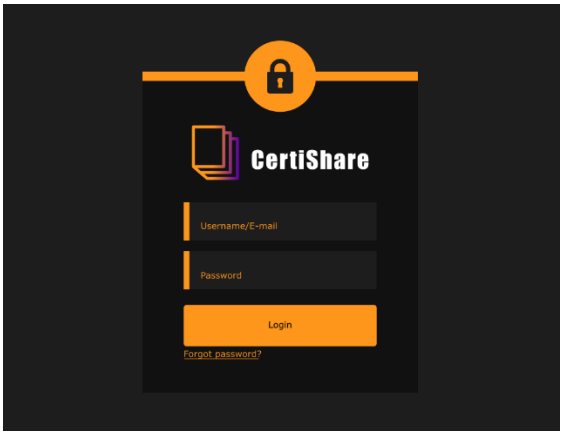

Figure 10
Login Page

b. First Registration Page

Figure 11 shows the First Registration Page which is a page that is only accessible through an invitation link. The purpose of this page is to act as a page to finish the registration of an account. When a user clicks on a link that has been generated by the system after a registration form has been filled by the responsible parties, it will prefill the username field, which will then be disabled and cannot be edited.

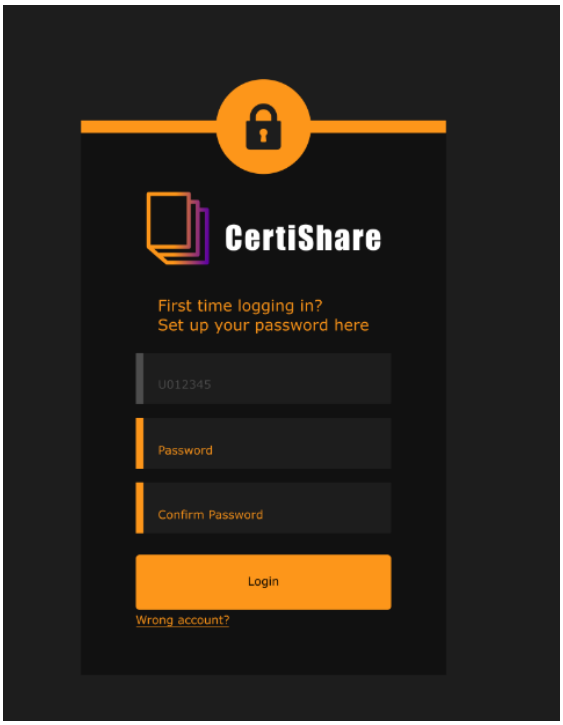

Figure 11
First Registration Page

c. Register University

Figure 13 represents the web-form to register a university. This page is accessible to the Government role. The most important information to be stored is the University ID, which for this system is determined to be SKPT, with reference from Dikti's official website [13].

d. Register Students

Figure 12 represents the web-form to register a student. This page is accessible to the University role. The Register Student feature consists of extremely minimal data. The reason for a minimisation of data, in summary, is because inserting a large chunk of data to the blockchain network requires a considerably big amount of effort and resources. To exercise integrity, consistency and security, the system still needs a considerable amount of data on each university, but a student's data need not be as extensive.

e. University Account

Figure 14 illustrates the account page of the University user, where they also have a feature to link their Ethereum Wallet. The figure displays an already linked Ethereum Wallet page. When unlinked, the right side of the page will only display a button to link the wallet, instead of a full information of the linked wallet as in the figure.

f. Upload Documents

Figure 15 displays the page to upload documents. This page is accessible to the University role. A fee will be incurred in ETH currency for every document upload.

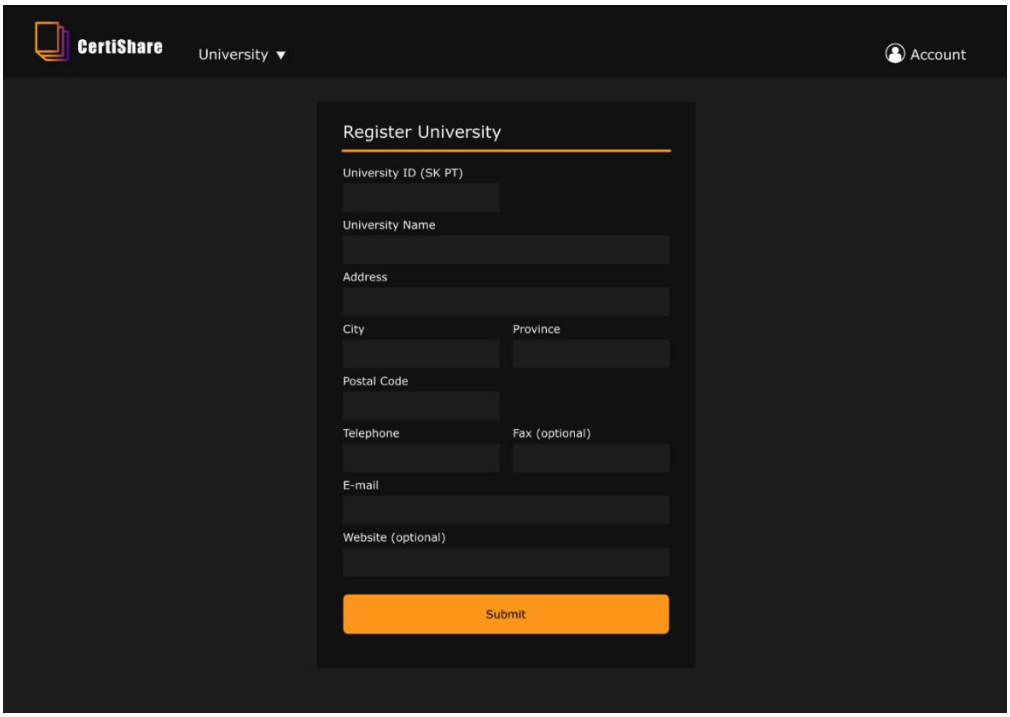

Figure 13
Register University Page

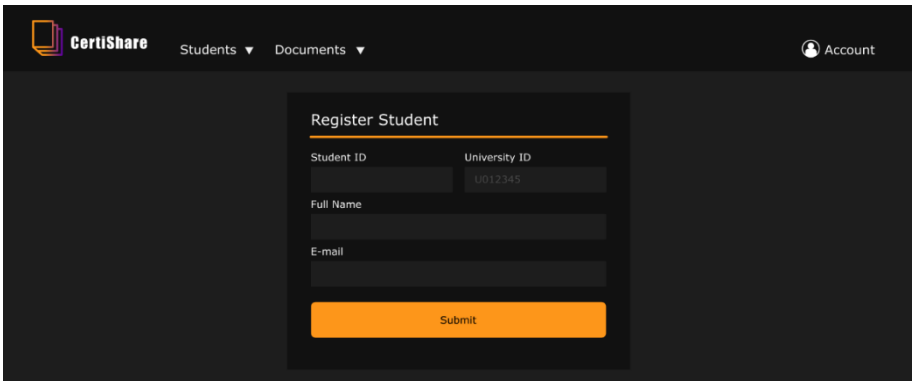

Figure 12
Register Student Page

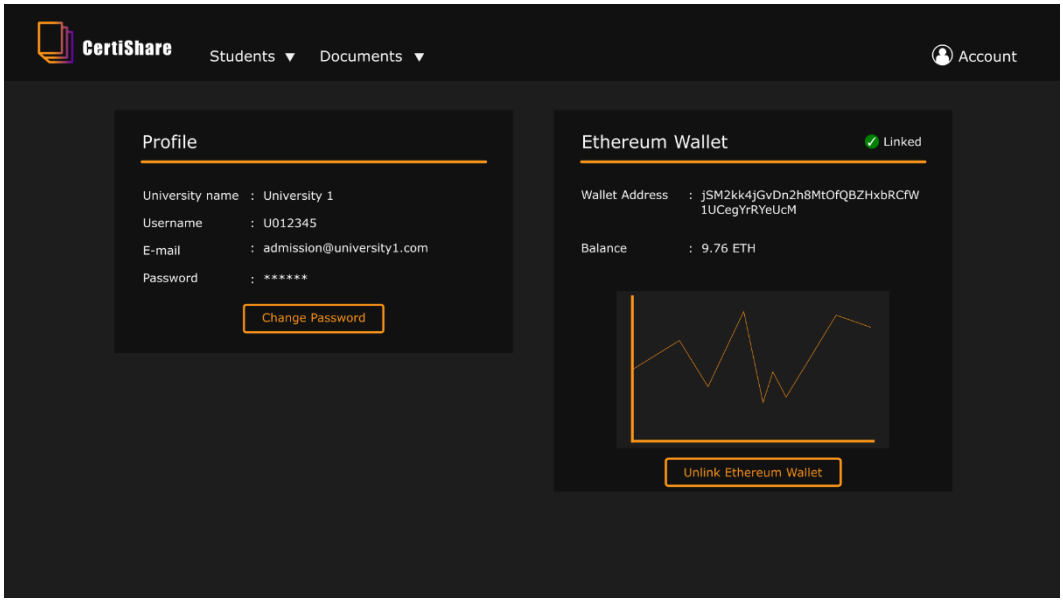

Figure 14
University Account Page, with a Linked Ethereum Wallet

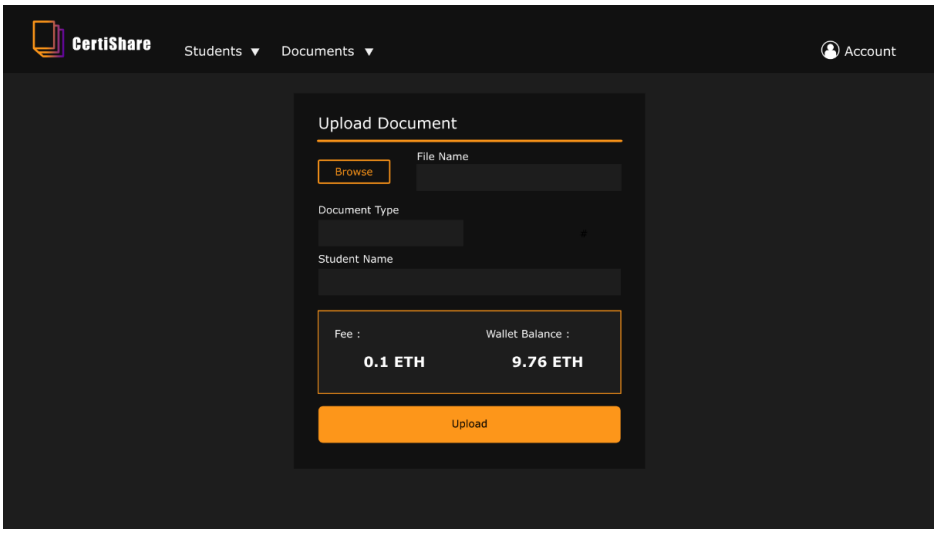

Figure 15
Upload Document Page

g. View Documents

Figure 16 represents the page to view the documents from the University role, while Figure 17 represents the page from the Student role. The student may share their document with third parties.

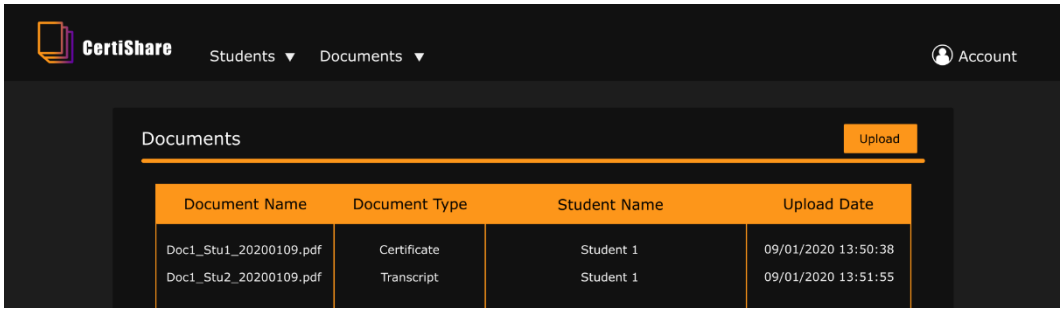

Figure 16
View Documents Page by University Role

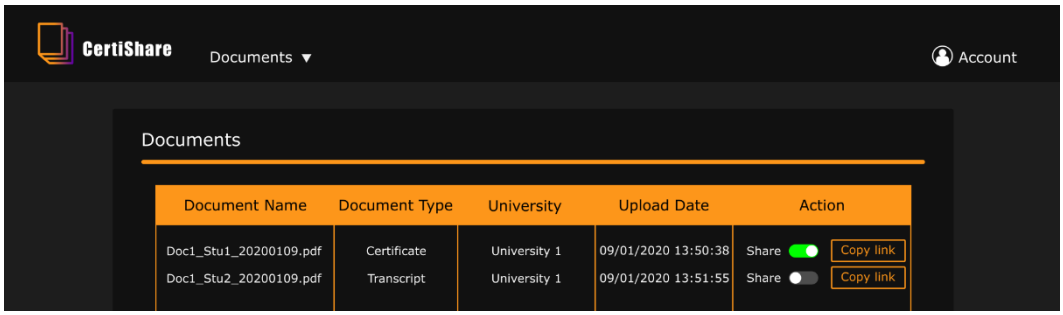

Figure 17
View Documents Page by Student Role

h. View Shared Document

Figure 18 represents the page where people with links may view the shared document. This page is accessible by the Public role, i.e. anyone.

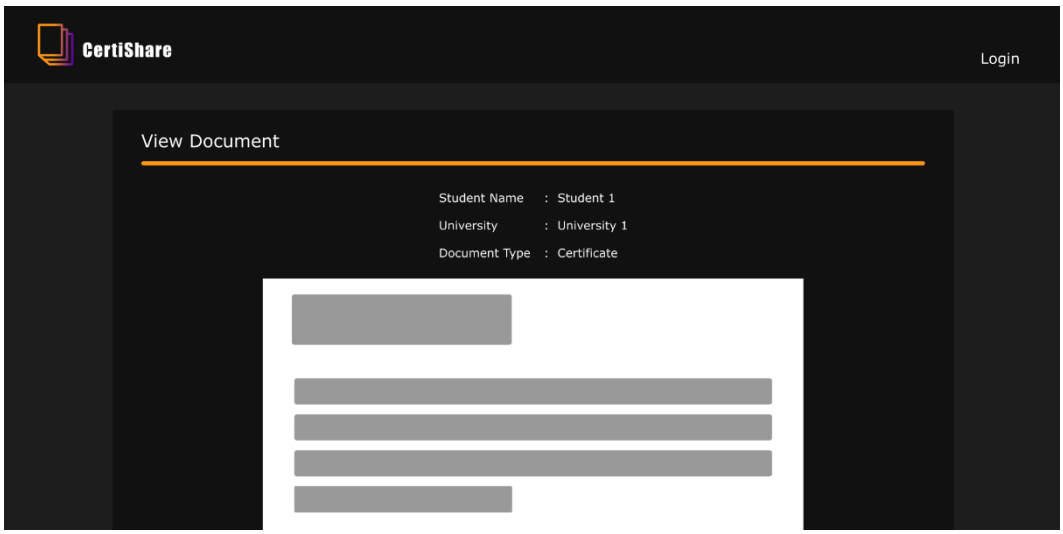

Figure 18
View Shared Document Page

5. CONCLUSIONS AND FUTURE WORKS

After careful analysis and design of the system, the author has drawn several conclusions. A secure platform to share official documents, specifically in the education department is necessary. Blockchain provides the solution, as it is a very secure and effective technology if implemented correctly. Building a system also requires incorporating many other existing applications to produce a more effective and secure end-product. In addition to that, a careful and thorough analysis must also be carried out to design a product correctly and effectively.

However, the system design presented in this paper is merely an early design and is thus, far from perfect. There are yet many feature improvements to be made in the future. One such improvement could be made regarding the document change or revoke request. The current design only allows for the upload of a new document, but there may be times when a document was incorrectly uploaded and may need to be changed or revoked. Another improvement to be made could be in the form of limiting the document sharing feature. The current system allows for document sharing by using a link that may be enabled and disabled by the corresponding user. A limitation on the document sharing feature would present the students with more options to choose whom to share it with and for how long the link may be available.

REFERENCES

[1] S. Kolvenbach, R. Ruland, W. Gräther and W. Prinz, “Blockchain 4 Education,” European Society for Socially Embedded Technologies (EUSSET), Nancy, France, 2018.

[2] Media Lab Learning Initiative and Learning Machine, “Digital Certificates Project,” Media Lab Learning Initiative and Learning Machine, [Online]. Available: http://certificates.media.mit.edu/. [Accessed 18 March 2020].

[3] “ML Learning Initiative,” MIT Media Lab Learning Initiative, [Online]. Available: https://learn.media.mit.edu/. [Accessed 18 March 2020].

[4] “Hyland,” Leaning Machine, [Online]. Available: https://www.learningmachine.com/. [Accessed 18 March 2020].

[5] B. Boeser, “Meet TrueRec by SAP: Trusted Digital Credentials Powered by Blockchain,” Systems, Applications, and Products in Data Processing (SAP), 24 July

2017. [Online]. Available: https://news.sap.com/2017/07/meet-truerec-by-sap-trusted-digital-credentials-powered-by-blockchain/. [Accessed 18 March 2020].

[6] T. Nguyen, Gradubique: An Academic Transcript Database Using Blockchain Architecture, San José: San José State University, 2018.

[7] "My eQuals," My eQuals, [Online]. Available: https://www.myequals.edu.au/. [Accessed 18 March 2020].

[8] M. O'Brien and G. R. S. Weir, "Understanding digital certificates," University of Strathclyde, Glasgow, United Kingdom, 2008.

[9] A. Grech and A. F. Camilleri, "Blockchain in Education," Publications Office of the European Union, Luxembourg, 2017.

[10] Ascertia Limited, "Basics of Digital Signatures & PKI," Ascertia Limited, 2017.

[11] "uPort," uPort, [Online]. Available: https://www.uport.me/. [Accessed 15 April 2020].

[12] "Sovrin," Sovrin Foundation, [Online]. Available: https://sovrin.org/. [Accessed 15 April 2020].

[13] "PDDIKTI," PDDIKTI, [Online]. Available: https://forlap.ristekdikti.go.id/. [Accessed 16 April 2020].