Software Requirement Specification Document For Digital Certificates Using Blockchain

Alley El-Dorry, Sherif Abd El Khalek, Mohamed Reda, Shehab El-Din Mohamed

March 19, 2020

1 Introduction

1.1 Purpose of this document

This Software Requirement Specification Document was created with the purpose of stating our system's requirements. This system's purpose is the prevention of the forgery of the academic certificates and making it easier for students to issue their certificates than the traditional way, this is done by the help of the blockchain technology using Linux Foundation's Hyperledeger Fabric. This document will define the purpose and the software and hardware specification of the system and also providing an overview of the system's goals and achievements.

1.2 Scope of this document

The Digital Certificates Using Blockchain system is designed to target students, universities, Ministry of Higher Education and companies. The main scope of this system is to take advantage of the main characteristics of blockchain which make it secure. Students will be able to issue their certificates. Companies will be able to verify the certificates of their employees. Private universities will take approval of the Ministry of Education before issuing a certificate for the student. This process my be expanded in the future by adding more universities and companies to the system.

1.3 Overview

As seen in Figure 1 the network consists of 4 types of physical organizations: public universities, private universities, companies and the Ministry of Higher Education and Scientific Research. There are also graduates who are considered as actors with a specific role within their university. Each participant or actor within the network has their specific permissions. Graduates can request their certificate from their university, companies can verify the validity of any

certificate. Misr International University (MIU), which is a private university, has the permission to register its graduates and also can query its graduates certificates but it can't issue certificates since it's a private university. However, Ain Shams University (ASU), which is a public university, has the permission to register its student and issue certificates to them and also can query its graduates certificates. Moreover, the Ministry Of Higher Education can query the whole ledger and can issue certificates to private universities' graduates. The companies will be considered as actors that have the permission of validating a certificate so they can access the validation portal and validate certificates. The transaction data will be the graduates certificates snapshot.
The process start with a student who logs in and requests a certificate and pays a certain fee to complete his request from his university , in case if it it is a public university, the university can directly issue the student's certificate to him, but if it is a private university it will need to request approval from the Ministry Of Higher Education to issue the certificate once it's approved the certificate is issued and once a certificate is issued it will store it and its validation key on the ledger, once the fresh graduate applies for a job in a company , the company will use his validation key to validate his certificate and check for its existance on the ledger.

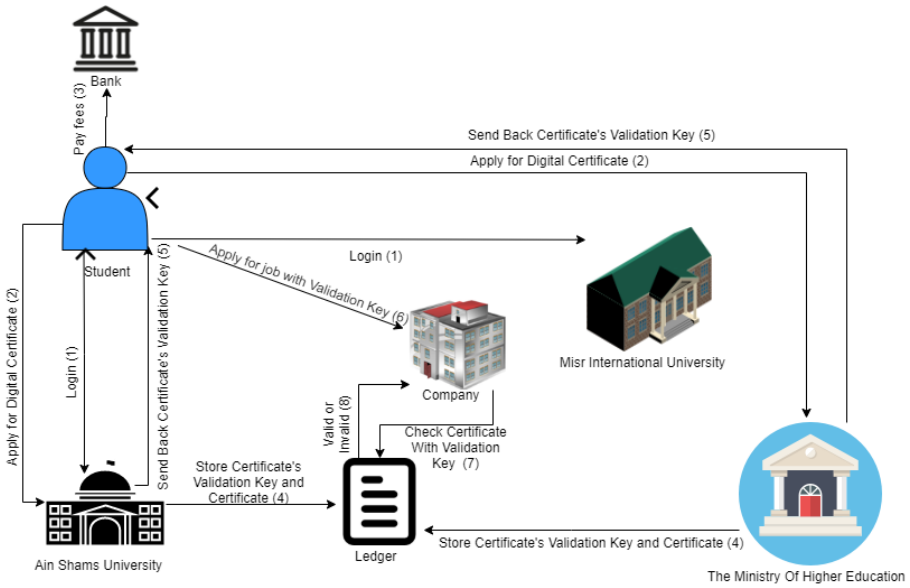


Figure 1: System Overview

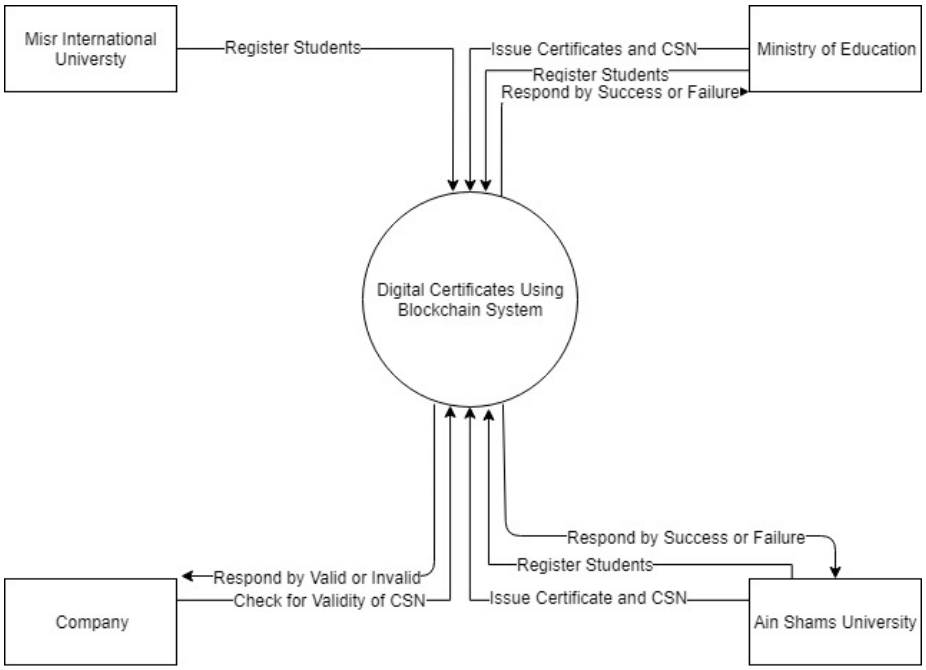


Figure 2: Context Diagram

1.4 Business Context

The forgery of certificates is a huge problem in our society and that's why were motivated to do this system, also we were motivated by other universities previous work. 94% of the forged certificates were Egyptian and that 47 cases of fraud in scientific certificate were seized during 2018, it was pointed out that 44 of them were Egyptian universities [1]. So our system's main objective is to offer an easy way to secure the certificates from being forged and also easier retrieval and verification of it.

Figure 3: Business Model

2 General Description

2.1 Product Functions

The system's main functionalities are:

1. The student will be able to review the digital certificates on his account through the system.
2. Universities will issue certificates for its own students easily using the system as well as each university will be able to review its own students' certificates.
3. The Ministry of Higher education can use the system to review any data from any university.
4. Recruiting companies can verify the applicant's certificates on the system.
5. The system will be composed of a network, its participants are: The Ministry of Higher Education , public universities and private universities.

2.2 Similar System Information

2.2.1 Taiwan University

The University of Taiwan has developed a blockchain application for the aim of verifying digital certificates. The application was implemented on the Ethereum platform due to its security and its rule of controlling data was managed by entities which are chosen by the Administrator not random individuals. Their system was run by the EVM (Ethereum Virtual Machine), which is blockchain app that allows developers to execut commands dynamically and not in a fixed manner, it executes using Solidity programming Language.
The system's process starts with a newly graduated student who applies for a digital certificate, his school simply reviews his request, once it is found to be valid the system records a new serial number for his/her certificate in the blockchain and sends back to the student a digital certificate which has a QR code. By the time a student applies for a job he/she sends the digital certificate with the QR code to the company applied, which is later reviewed by the company to check its validity. [2]

Figure 4: A Validated Certificate [2]

2.2.2 University of Jordan

Tarek Kanan , Ahamd Turki Obaidat , Majduleen Al-Lahham of the University of Jordan are the developers of Smartcerts who were inspired by Blockcerts ,which is the first digital certificate application. They used Ethereum Blockchain to take advantage of its Smart Contract; which enables the system to follow cer-

tain business logic to run the process of the certificate verification. In addition, they used Solidity and its IDE remix as a compiler in case of errors. The system consists of two types of users, a User and Admin. The user can view his/her certificates with his National ID. While the Admin acts as the root of the system. An admin can add new administrators in the network, create new certificates, and issue them to a user.[3]

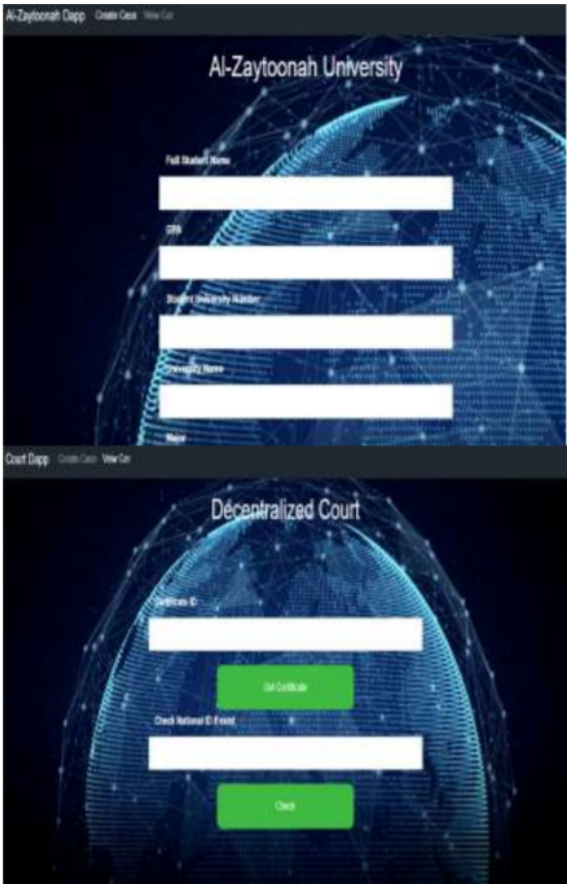


Figure 5: Part Of The System's UI
[3]

2.3 User Characteristics

The system has five main users for the flow of the system's cycle and all require basic knowledge of interacting with a web application interface.

1. Student: Can use the system to register himself in his enrolled college, request his/her certificates and review them.

2. Public Universities: Can issue certificates for the registered student and can review certificates for its own students.

3. Private Universities : Can review certificates for its own student.

4. Ministry Of Higher Education: Can issue certificates for students in private universities and can review all certificates across all universities.

5. Company's recruitment staff: Takes the applicant's serial number to validate his digital certificates.

2.4 User Problem Statement

The problem here is the wide forgery of School, Bachelor and Masters Degrees in Egypt which may have disastrous impact on the society and the academic community. Secondly, the inconvenience of the process of requesting and issuing a certificate and the huge waste of time it costs , That is what the system aims at solving.

2.5 User Objectives

The student wants to receive a digital certificate from his university using the system and review them , he wants to apply for a job in a company and the recruiting company needs to verify his applied certificates using our system.

2.6 General Constraints

Since the system is going to be a web application it's going to need a device with an internet browser.

3 Functional Requirements

3.1 Certificate Exists

Function Title	Certificate Exists
ID	FR1
Description	For Checking if a certificate exists in the world state.
Action	looks for a certificate with specific key in the world state.
Input	transaction context, Certificate validation key
Output	True or False
Pre-condition	Logged in as public/private University Admin or Mohesr Admin
Post-condition	N/A

3.2 Issue Certificate

Function Title	Issue Certificate
ID	FR2
Description	For issuing a certificate for student.
Action	Issuer creates a cerifcate and adds it to the blockchain then sends to the student a validation key and a QR code
Input	transaction context, Certificate Data, Student national ID
Output	Validation key and QR code
Pre-condition	Logged in as a public university or MOHESR Admin
Post-condition	a new transaction is added to the blockchain and the world state.

3.3 Read Certificate

Function Title	Read Certificate
ID	FR3
Description	For querying a transaction data (certificate) from the world state.
Action	Reads the value of a specific key.
Input	transaction context, Certificate validation key
Output	Certificate Data
Pre-condition	Logged in as public/private University Admin or Mohesr Admin
Post-condition	N/A

3.4 Update Certificate

Function Title	Update Certificate
ID	FR4
Description	For Updating a certificate.
Action	Updates a certificate of a specific key.
Input	transaction context, Certificate validation key,certificate data
Output	Success or Failure
Pre-condition	Logged in as public/private University Admin or Mohesr Admin
Post-condition	a new transaction made to the ledger and world state is changed

3.5 Delete Certificate

Function Title	Delete Certificate
ID	FR5
Description	For deleting certificate from world state.
Action	Makes the certificate unavailable.
Input	transaction context, Certificate validation key
Output	Success or Failure message
Pre-condition	Logged in as public University Admin or Mohesr Admin
Post-condition	Certificate removed from world state

3.6 Enroll admin

Function Title	Enroll admin
ID	FR6
Description	For enrolling an admin to the CA server.
Action	Adds admin identity to the CA server.
Input	Admin username, Admin Password, path to store x.509 certificate (PKI certificates used by hyperledger fabric)
Output	Success or Failure
Pre-condition	N/A
Post-condition	Admin identity added to the CA server

3.7 Register User In Ca

Function Title	Register User In CA
ID	FR7
Description	For registering a User to the CA server.
Action	Adds user identity to the CA server.
Input	User's username, User's Password, path to store x.509 certificate (PKI certificates used by hyperledger fabric)
Output	Success or Failure
Pre-condition	N/A
Post-condition	User identity added to the CA server

3.8 Validate Certificate

Function Title	Validate Certificate
ID	FR8
Description	For the recruiting organization to check the validity of the applicant's graduation certificate.
Action	Check the blockchain for a non expired certificate with a specific validation key
Input	Certificate validation key
Output	The certificate of the applicant in case if there is a valid one , if not it will show a message informing that
Pre-condition	Logged in as a recruiting organization
Post-condition	certificate validated

3.9 Request Certificate

Function Title	Request Certificate
ID	FR9
Description	The student requests issuance of his certificate from his university
Action	A request is sent to the student's university
Input	Student's national ID
Output	Success or Failure Message
Pre-condition	Logged in as a student in his university's portal
Post-condition	A request is sent to the University

3.10 Respond to Request

Function Title	Respond to Request
ID	FR10
Description	The university responds to the student's request of issuance.
Action	The request is either approved or declined
Input	Student Email
Output	Success or Failure
Pre-condition	Logged in as a university Admin
Post-condition	Request accepted or declined message sent to the student

3.11 Generate QR code

Function Title	Generate QR code
ID	FR11
Description	For generating a QR Code.
Action	Generates a QR code from a string.
Input	String
Output	QR Code
Pre-condition	N/A
Post-condition	N/A

3.12 Read QR code

Function Title	Read QR code
ID	FR12
Description	For Reading a QR Code.
Action	Generates a String from a QR Code.
Input	Uint8ClampedArray, width, height
Output	String
Pre-condition	N/A
Post-condition	N/A

3.13 Generate CSR

Function Title	GenerateCSR
ID	FR14
Description	Creates the certificate signing request which is responsible for the identity of the identity of each user who signs in.
Action	Generates the certificate signing request for the user.
Input	N/A
Output	String
Pre-condition	The user must sign in
Post-condition	must use the signing request for the certificate authority to validate identity.

3.14 Validate User

Function Title	ValidateUser
ID	FR15
Description	The Certificate authority uses the signing request to validate the user.
Action	Validates the user.
Input	User's sign request
Output	Validation Message
Pre-condition	The user must sign in
Post-condition	N/A

3.15 Readx509Path

Function Title	Readx509Path
ID	FR16
Description	Uses the user's member path to read his x509 certificate.
Action	reads his x509 certificate.
Input	user's member path
Output	Validation Message
Pre-condition	N/A
Post-condition	N/A

3.16 Pay

Function Title	Pay
ID	FR17
Description	The student upon requesting for his academic certificate he needs to pay a certain fee.
Action	checks the user's bank balance and consumes the required fee to request an academic certificate.
Input	CreditCard Number , CreditCard Password
Output	Confirmation Message
Pre-condition	The credit card's current balance must be bigger than or equal to the fee of requesting a certificate (30 EGP)
Post-condition	N/A

3.17 Add Company

Function Title	Add Company
ID	FR18
Description	The regulator Adds a company to the network
Action	A company will be added to the network.
Input	CompanyName
Output	Confirmation Message
Pre-condition	Valid Data Entry
Post-condition	N/A

3.18 Add University

Function Title	Add University
ID	FR19
Description	The regulator Adds a university to the network
Action	A university will be added to the network.
Input	UniversityName
Output	Confirmation Message
Pre-condition	Valid Data Entry
Post-condition	N/A

4 Interface Requirements

Here is how the system interacts with the users to through the web application.

4.1 User Interfaces

4.1.1 GUI

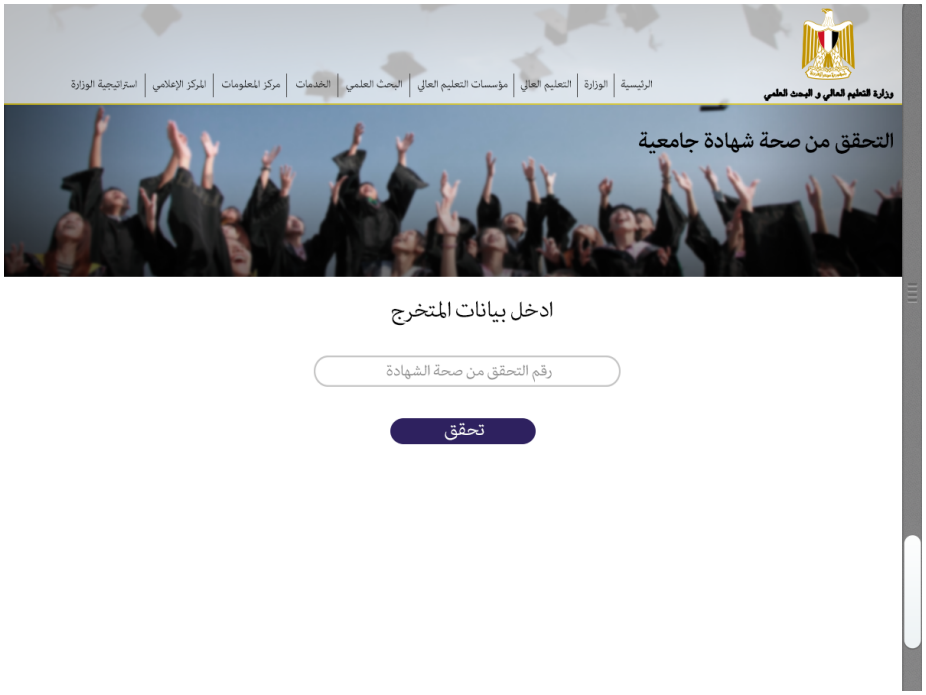
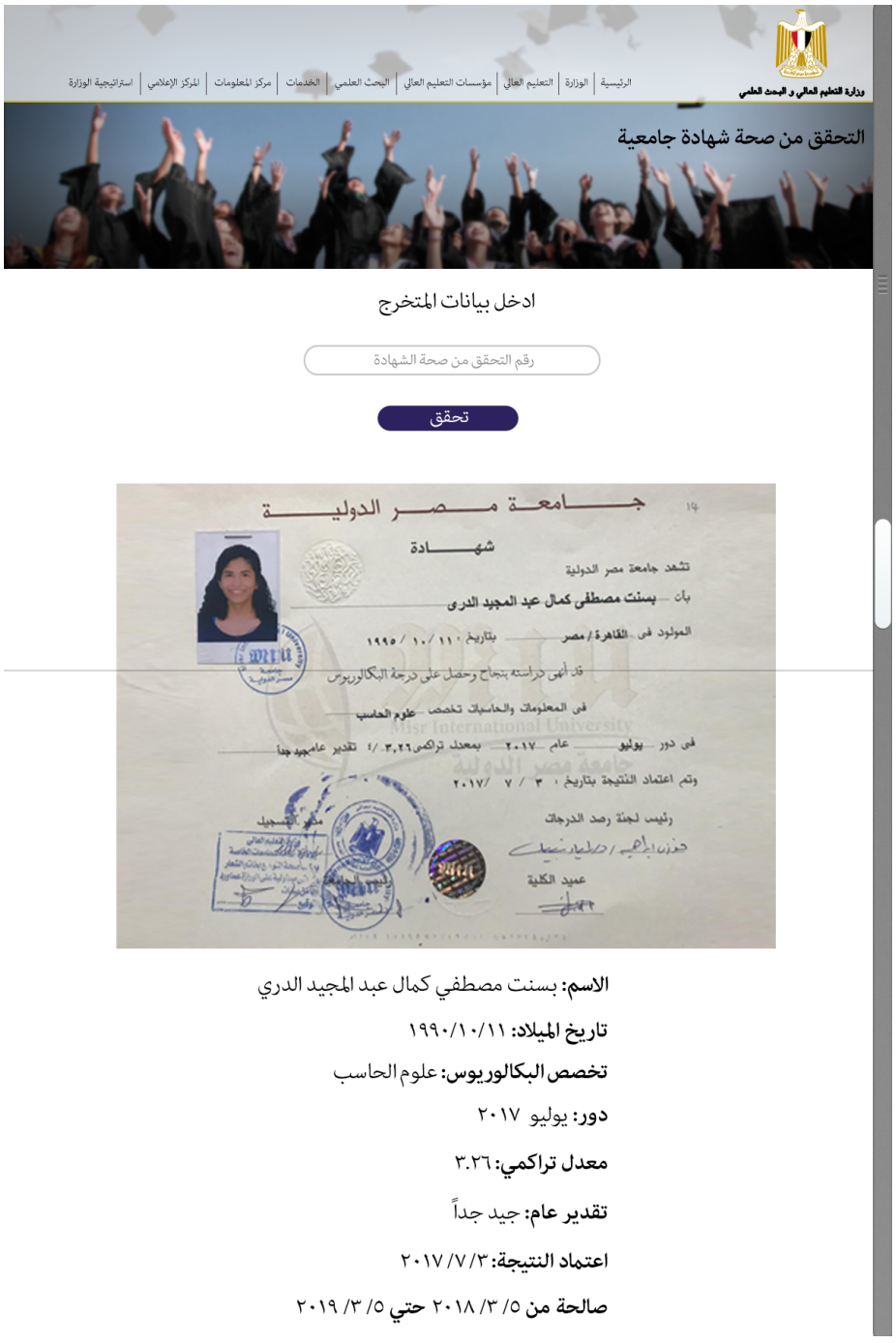


Figure 6: MOHESR webapp



الاسم: بسنت مصطفي كمال عبد المجيد الدري

تاريخ الميلاد: ١٩٩٠/١٠/١١

تخصص البكالوريوس: علوم الحاسب

دور: يوليو ٢٠١٧

معدل تراكمي: ٣.٢٦

تقدير عام: جيد جداً

اعتماد النتيجة: ٢٠١٧/٧/٣

صالحة من ٢٠١٨/٣/٥ **حتي** ٢٠١٩/٣/٥

Figure 7: Valid certificate

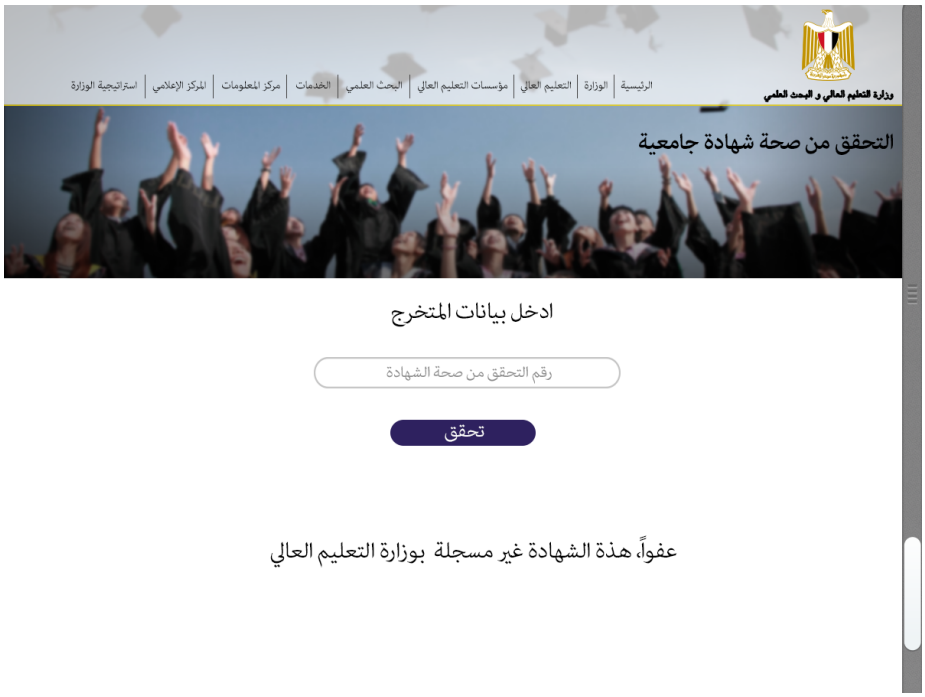


Figure 8: Invalid certificate

4.1.2 CLI

1) Hyperledger Fabric Commands:

- It's a tool to manage the blockchain netowrk with simple commands.

2) Docker Compose:

- Compose is a tool for defining and running multi-container Docker applications. With Compose, you use a YAML file to configure your application's services. Then, with a single command, you create and start all the services from your configuration.

2) Node.js:

- is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

4) Git:

- Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

4.1.3 API

- Fabric Contract API
- Fabric Network API
- Fabric CA Client API
- Fabric Client API
- QRCode Generator API

4.2 Software Interfaces

Our Blockchain solution will be developed by using Hyperledger Fabric and for the chaincode will be using JavaScript. The web application will be developed using Node.js.

5 Performance Requirements

The blockchain will be stored and executed on a cloud server so as the ledger and the database so there will be no intensive tasks on the client hardware.

6 Design Constraints

6.1 Standards Compliance

The software developers should develop the blockchain with the help of the Hyperledger Fabric documentation for proper executing of the blockchain.

6.2 Software Limitations

1. The admin of the network must be on Linux to start, stop or manage the network.
2. Any user on the network must have a internet connection.

6.3 Hardware Limitations

- Any laptop or mobile device that can load web pages for end users
- Cloud Specifications:
  - A minimum of 32 core Intel.
  - A minimum of 5 Giga-bit network performance.
  - A minimum of 128 Gigabyte of memory.
  - Unlimited storage.

7 Other non-functional attributes

7.1 Security

There is no central authority so, No one can change any data on the network for their personal benefits, this is achieved through blockchain and transport layer security and add another layer of security for the system by using encryption.

7.2 Reliability

The system will be dealing with a lot of data so it is important that the transactions are done without errors or delays , the system's use of blockchain's structure and benefits makes the system incredibly trustworthy and reliable for use.

7.3 Portability

Being portable will make it available for use to all sorts of devices and platforms so it will be handy to use mostly anywhere when you need it which will be better compared to the traditional methods of issuing a certificate.

7.4 Usability

The system's user interface shouldn't be complicated as the number of tasks done is few so it will be easy to memorize how to use it , making it simple for anyone to use and being able to get used to using it.

7.5 Scalability

The system is made ready for future changes, being scalable makes it dynamic and easy to alter and change the code in case of any change of requirements , domain changes or future updates.

8 Preliminary Object-Oriented Domain Analysis

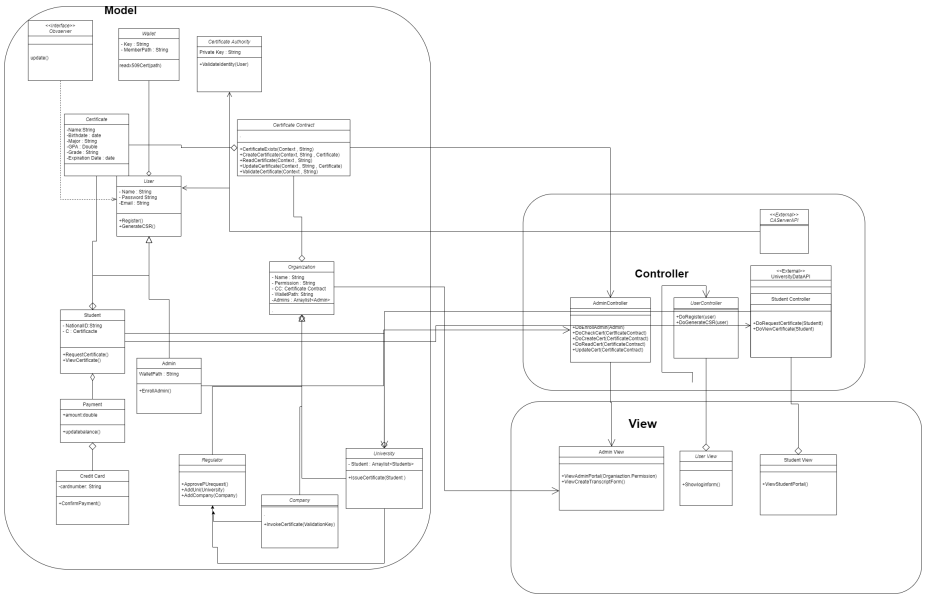


Figure 9: Class Diagram

8.1 Class Descriptions

8.1.1

Class Name	User
Super Classes	N/A
Sub Classes	Student , Admin .
Purpose	Encapsulate the basic attributes of all users in the system.
Collaboration	Aggregates Wallet , implements observer interface.
Functions	Register , GenerateCSR

8.1.2

Class Name	Student
Super Classes	User
Sub Classes	N/A .
Purpose	Presents the student.
Collaboration	Student aggregates Certificate and Payment , extends user class and associates class StudentController.
Functions	RequestCertificate , ViewCertificate.

8.1.3

Class Name	Admin
Super Classes	Student
Sub Classes	N/A .
Purpose	Represents the admin.
Collaboration	extends user class , aggregated Organization.
Functions	CertificateExists , CreateCertificate , ReadCertificate, Upate-Certificate, ValidateCertificate.

8.1.4

Class Name	Organization
Super Classes	N/A
Sub Classes	Regulator , Company , University .
Purpose	Encapsulates the basic information of the the organizations in the network.
Collaboration	Aggregates Admin and Certificate Contract.
Functions	N/A.

8.1.5

Class Name	Regulator
Super Classes	Organization
Sub Classes	N/A.
Purpose	Represents the regulator in the network.
Collaboration	Associates Company and University , extends University.
Functions	ApprovePURequet , AddUni , AddCompany

8.1.6

Class Name	Company
Super Classes	Organization
Sub Classes	N/A .
Purpose	Represents the company in the network.
Collaboration	extends Organization.
Functions	InvokeCertificate.

8.1.7

Class Name	University
Super Classes	Organization
Sub Classes	N/A .
Purpose	Represents the university in the network .
Collaboration	Extends Organization , aggregates Student
Functions	IssueCertificate.

9 Network Topology

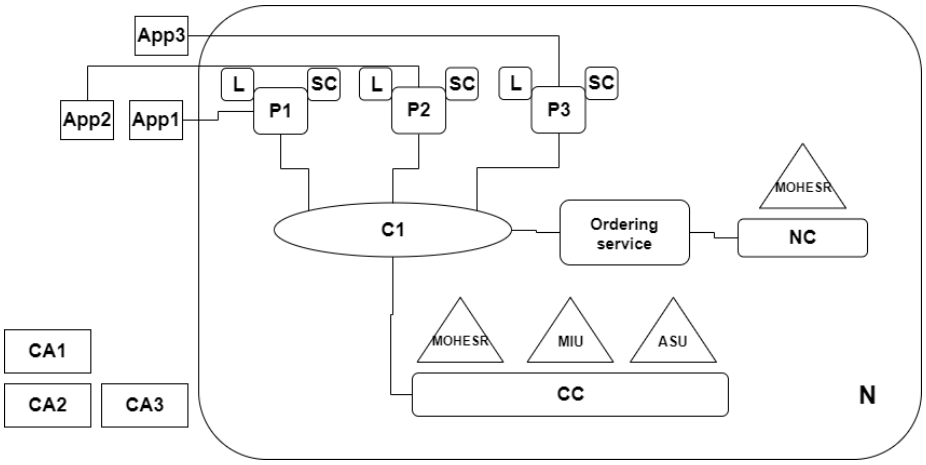


Figure 10: The Network Topology

As can be seen in figure 10 the network consists of three participants MOHESR, MIU, and ASU communicating over one channel [10] C1 with channel configuration CC . Although any university's ledger transactions (certificates) should only be private to it and to the ministry of higher education, while the other universities ledger shouldn't have those transactions available publicly on their ledger. However, we have one channel for all participants. This is because we are using Private Data Collection (PDC)[8] that was introduced in Fabric v1.2. It allows a defined subset of organizations on a the same channel the ability to endorse, commit, or query their private data without having to create a separate channel. However, the transaction data will be hashed for other organizations on the same channel. PDC is used since if we depend on using separate channels for privacy, each university in the network will need its own channel with the ministry. As a result it will cause a huge overhead with the network scaling. There are three peer [**?**] clusters P1, P2 and P3, each P is a cluster of peers owned by a specific university or by the ministry. Each one holding the chaincode[7] SC [7] and the ledger L.There are three certificate authorities CA1, CA2 and CA3; one for each participating organization for registering its participants identities and generating Enrollment Certificates (ECerts) . Initially the network configuration NC is configured to accept the Ministry Of Higher Education and Scientific Research as the network administrator. There is the ordering service [6] which consists of one SOLO ordering node for ordering transactions which is just used for development. Finally, there are 3 web applications App1, App2, and App3. Each App is connected to a participating organization peers in order for the actors of these organizations to be able to interact with the blockchain in requesting, issuance or verification of certificates, according to their permissions.

Anyone or organization that needs to verify certificates should be considered as a MOHESR actor in order to be able to use App3 to verify certificates.

10 Operational Scenarios

10.1 Private university certificate issuance

A Student in Misr International University (MIU) wants to receive his certificate, MIU is a private university so his certificate request must be made to The Ministry Of Higher Education (MOHSER) , and so when the request is sent and MOHSER confirms it , the student's certificate gets issued to him.

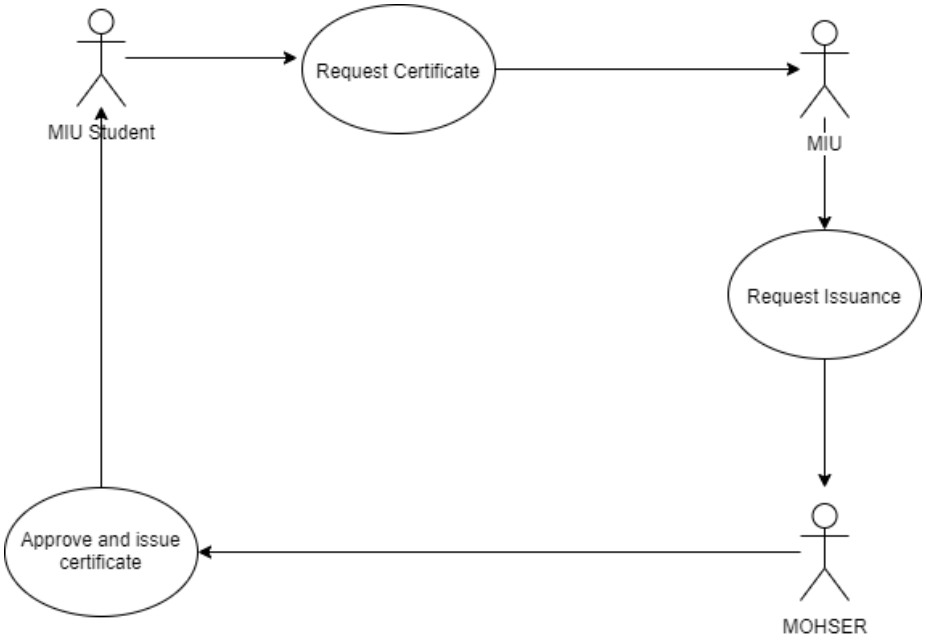


Figure 11: Private University Certificate Issuance Use Case

10.2 Public university certificate issuance

A Student in Ain Shams University (ASU) wants to receive his certificate, ASU is a public university so his/her certificate is made directly to the university , once the request is sucessfully validated and is granted it sucessfully issues the certificate to the student.

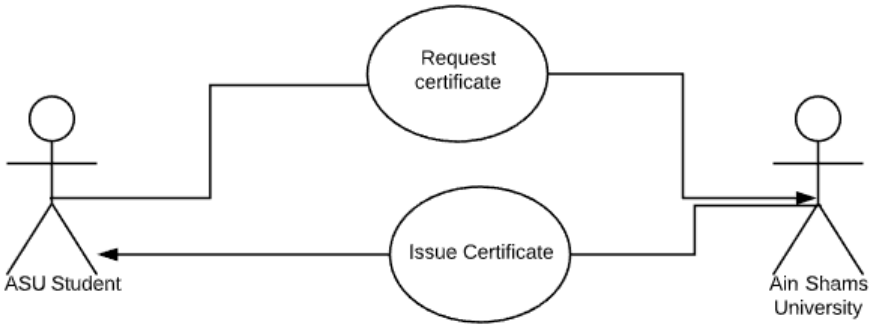


Figure 12: Public University Certificate Issuance Use Case

10.3 Student applies for a job in a company

Once someone applies for a job in a company that supports the system , he will give them his identification which the recruitment staff will take and will type it into the system to validate the applicant's certificate , the system will go to through to the records in MOHSER and if it exists it will send back the verification message.

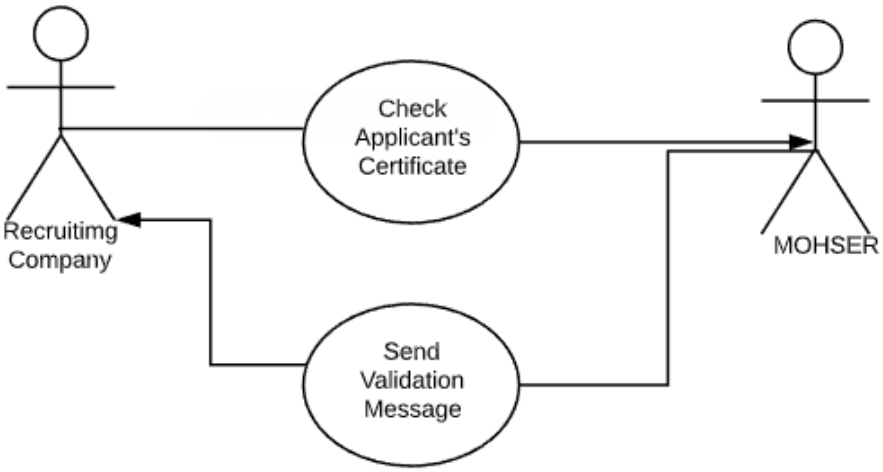


Figure 13: Public University Certificate Issuance Use Case

11 Preliminary Schedule Adjusted

Task	Start Date	End Date
Idea Discussion	18-7-2019	21-7-2019
Idea Research	21-7-2019	1-9-2019
Survey and Proposal	12-9-2019	8-9-2019
Implementing Prototype	4-10-2019	8-10-2019
Proposal Presentation	8-10-2019	8-10-2019
Designing Application	8-10-2019	30-10-2019
Implementing GUI Design	30-10-2019	12-11-2019
Designing Database	12-11-2019	20-11-2019
Designing Class Diagram	20-11-2019	27-11-2019
SRS Writing	27-11-2019	8-12-2019
SRS Presentation	8-12-2019	14-12-2019
Implementing Application	14-12-2019	5-2-2020
SDD Writing	5-2-2020	14-2-2020
SDD Presentation	14-2-2020	21-2-2020
Validation and Testing	21-2-2020	10-3-2020
Writing Paper	10-3-2020	25-3-2020
Delivering Papers	25-3-2020	1-4-2020
Writing Thesis	1-4-2020	20-5-2020
Delivering Thesis	20-5-2020	30-5-2020
Final Presentation	24-6-2020	24-6-2020

12 Preliminary Budget Adjusted

- Cloud Service (AWS, IBM Cloud).
- Kubernetes cluster (K8S) (containing Hyperledger Fabric).
- Domain name.

13 Appendices

13.1 Definitions, Acronyms, Abbreviations

(CSN) Certificate Serial Number.
(MIU) Misr International University.
(MOE) Ministery of Education.
(ASU) Ain Shams University.
(CA) Certificate Authority.
(PKI) Public Key Infrastructure.

13.2 Collected material

14 References

References

[1] "Kuwait - Around 47 cases of forged science degrees discovered so far." MENAFN, Arab Times, 9,December,2018, https://www.menafn.com/1097808046/Kuwait—Around-47-cases-of-forged-science-degrees-discovered-so-far

[2] J. Cheng, N. Lee, C. Chi and Y. Chen, "Blockchain and smart contract for digital certificate," 2018 IEEE International Conference on Applied System Invention (ICASI), Chiba, 2018, pp. 1046-1051. doi: 10.1109/ICASI.2018.8394455

[3] Kanan, T., Obaidat, A. T., & Al-Lahham, M. (2019). SmartCert BlockChain Imperative for Educational Certificates. 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT). doi:10.1109/jeeit.2019.8717505

[4] "Introduction" Hyperledger (2019) https://hyperledger-fabric.readthedocs.io/en/release-1.4/whatis.html

[5] "Peers" Hyperledger (2019) https://hyperledger-fabric.readthedocs.io/en/release-1.4/peers/peers.html

[6] "The ordering service" Hyperledger (2019) https://hyperledger-fabric.readthedocs.io/en/release-1.4/orderer/orderingservice.html

[7] "Smart Contracts and Chaincodes" Hyperledger (2019) https://hyperledger-fabric.readthedocs.io/en/release-1.4/smartcontract/smartcontract.html

[8] "Private data" Hyperledger (2019) https://hyperledger-fabric.readthedocs.io/en/release-1.4/private-data/private-data.html

[9] "Certificate Authority" Hyperledger" (2019) https://hyperledger-fabric-ca.readthedocs.io/en/release-1.4/

[10] "Channels" Hyperledger (2019) https://hyperledger-fabric.readthedocs.io/en/release-1.4/channels.html