EduBlock

HSGamer

9/20/2022

Table of contents

Chapter 1

Preface

This is a book containing reports of EduBlock

Chapter 2

Plan (VI)

2.1 Phase 1 - Khởi động

Thời gian: 01/08/2022 - 04/09/2022

Mô tả: Đây là giai đoạn lên ý tưởng và thực hiện prototype đầu tiên cho dự án Học bạ điện tử, với phạm vi của giai đoạn là thực hiện hệ thống trên 1 trường, và thử nghiệm trên nền tảng public blockchain (Dfinity).

Sản phẩm: - Tài liệu yêu cầu - Tài liệu giới thiệu dự án - Tài liệu cấu trúc hệ thống Phase 1 - Hi-Fi Prototype Phase 1

2.2 Phase 2 - Vượt chướng ngại vật

Thời gian: 05/09/2022 - 30/09/2022

Mô tả: Đây là giai đoạn quan trọng để mở rộng phạm vi của dự án lên quy mô toàn quốc, bao gồm nghiên cứu các công nghệ cần cho việc thiết kế hệ thống phù hợp với quy trình xét học bạ hiện tại.

2.2.1 Task 1 - Mở rộng phạm vi

Mô tả: Mở rộng phạm vi dự án và xây dựng cái tài liệu mới dựa trên các tài liệu từ Phase 1. Chủ yếu là thiết kế mô hình hệ thống cho phạm vi được mở rộng này.

Lưu ý: Phạm vi lần này là mở rộng quy mô lên các Sở Giáo Dục của toàn quốc, và phạm vi học bạ bao gồm các điểm số nhỏ (Kiểm tra miệng, Kiểm tra 15 phút,…).

Sản phẩm: - Tài liệu yêu cầu - Tài liệu giới thiệu dự án theo phạm vi mới - Tài liệu cấu trúc hệ thống Phase 2

2.2.2 Task 2 - Thiết kế Backend

Mô tả: Thiết kế hệ thống backend theo phạm vi mới. Tìm hiểu công nghệ Hyperledger Fabric để xây dựng mạng lưới Private Blockchain phù hợp với yêu cầu.

Lưu ý: Nếu không kịp tìm hiểu Hyperledger Fabric hoặc nó không phù hợp thì vẫn có thể thiết kế hệ thống backend dạng MVC với phạm vi của từng Sở Giáo Dục và tìm cách liên kết các Sở Giáo Dục lại thành mạng lưới truyền thông tin.

Sản phẩm - Tài liệu thiết kế Backend - Tài liệu Endpoint của Web API - Hi-Fi Prototype Phase 2 cho Backend

2.2.3 Task 3 - Thiết kế Frontend

Mô tả: Thiết kế Backend sử dụng Framework phù hợp cho người lập trình. Mục tiêu là sử dụng Framework để vừa làm nhanh, phù hợp với người lập trình web thuần HTML, CSS và JS mà không có các logic phức tạp và lỗi giao diện.

Lưu ý: Có thể tự thiết kế Fake Web API để thử nghiệm việc kết nối đến Backend.

Ghi chú: Svelte?

Sản phẩm - Tài liệu thiết kế Frontend - Hi-Fi Prototype Phase 2 cho Frontend

2.2.4 Task 4 - Mở rộng tính năng nâng cao

Mô tả: Bao gồm sử dụng công nghệ nhận diện để nạp học bạ giấy lên hệ thống.

Lưu ý: Ở Phase này là nghiên cứu tìm cách thêm tính năng, những tính năng trong Task này nếu xét là không kịp thì có thể không làm.

Sản phẩm - Tài liệu và Prototype liên quan

2.3 Phase 3 - Tăng tốc

Thời gian: 01/10/2022 - 31/10/2022

Mô tả: Đây là giai đoạn thực hiện sản phẩm cuối cùng. Bao gồm hoàn thiện Frontend, Backend và các tính năng liên quan; Kết nối Frontend và Backend; Bổ sung tài liệu cài đặt hệ thống.

Sản phẩm - Hệ thống cuối, chưa qua Test - Tài liệu hệ thống - Hướng dẫn cài đặt hệ thống

2.4 Phase 4 - Về đích

Thời gian: 01/11/2022 - 30/11/2022

Mô tả: Đây là giai đoạn test hệ thống và viết báo cáo

Sản phẩm - Hệ thống cuối, đã Test - Tài liệu Testing - Tài liệu hệ thống cuối - Báo cáo dự án

Chapter 3

Requirement (Hackathon)

3.1 Introduction

This is the specification for a student record system that store the student records on the BlockChain network. The system is designed to provide a user-friendly interface for teachers to update their students' academic records and a full replacement of traditional academic record papers.

3.2 About Existing System

Many high schools in Viet Nam use one of existing online systems to store students' academic records and notify their parents about recent records, but they still use traditional record papers for post-graduation & university / college enrollment. Therefore, The teachers find it difficult to update their students' records because they have to update the records on both the online system & the record paper. Moreover, the online system is a centralized system that only the admins and the teachers can see and interact, so there is little transparency for students who want to see their records at any time.

3.3 User Requirement

3.3.1 Main Function

1. Normal User
    - Request Registration
        - The user must fill a form of personal information
        - The user must choose a role to request: Student or Teacher
        - The user can send the request form and wait for an admin to review
    - Login
        - The user must have an account on the system to login
2. Student
    - See Student Records

    - The user must login to use this function
    - The user can only see his academic records
3. Teacher
    - View Class
        - The user must login to use this function
        - The user can only see his own classes
        - The user can filter his classes by name or year
    - View Student
        - The user must login to use this function
        - The user can only see his own classes
        - The user can choose a class and see its student
        - The user can filter his students by name
    - Update Student Records
        - The user must login to use this function
        - The user can only update students of his classes
        - The user can choose a student and see his academic records
        - The user can update the student's records
4. Admin
    - View Class
        - The user must login to use this function
        - The user can see created classes
        - The user can filter the classes by name or year
    - Create Class
        - The user must login to use this function
        - The user must assign a teacher to the class as a home teacher
        - The user must assign students to the class
    - View Student
        - The user must login to use this function
        - The user can see registered students
        - The user can filter the students by name
    - View Teacher
        - The user must login to use this function
        - The user can see registered teachers
        - The user can filter the teachers by name
    - Review Registration
        - The user must login to use this function
        - The user can see the waiting requests
        - The user can accept or deny a request
        - The user can filter the requests by username or fullname
        - For teacher requests, the user can export a list to give to the On-Chain Admin
5. On-Chain Admin
    - Add/Remove Teachers
        - The user must interact with the BlockChain (On-Chain) service to use this function
        - The user can add or remove a list of teachers received from the Admin

3.3.2 Non-Function

- The system easy to maintain & upgrade
- The user interface is clear, idiot-proof, easy to use & friendly
- The system is available on 24/7

3.4 System Requirement

3.4.1 System Component

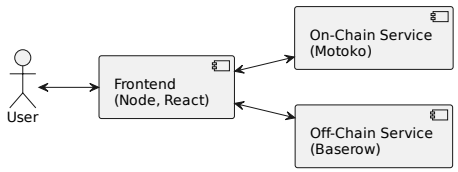


Figure 3.1: Overview of System component

Component	Description
Frontend	The interface of the system, responsible for UI/UX
Off-Chain Service	Store the personal information of the user, the registration requests and the details of classes & students
On-Chain Service	Store the student's academic records by grade & Allow teachers to update their students' records

3.4.2 Actor Description

Actor	Description
Student	The smallest actor of the system who can only see his academic records
Teacher	A person who can update the student's academic records
Admin	A manager of the system who manage teachers, classes & students
On-Chain Admin	A subset of Admin who interacts with the On-Chain service

3.4.3 Use case diagram

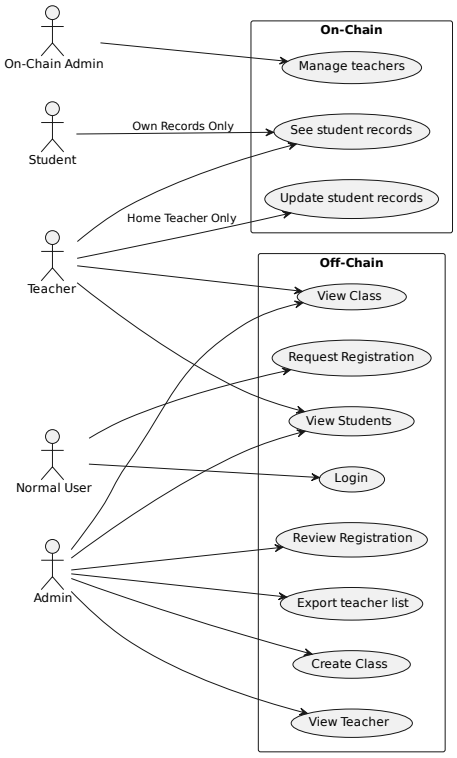


Figure 3.2: Use case diagram

Chapter 4

Requirement (CP)

4.1 System Architecture

- Without BlockChain

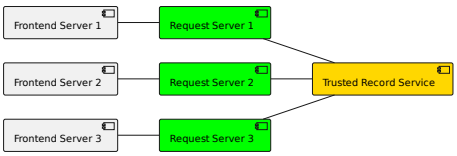


Figure 4.1: System architecture without BlockChain

- With BlockChain

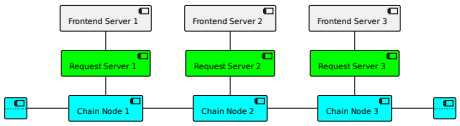


Figure 4.2: System architecture with BlockChain

Component	Description
Chain Node (CN)	A node of the blockchain. This stores the records and handles the history and transaction requests from the Request Server (Change/View the score, information, etc.)
Trusted Record Service (TRS)	Similar to Chain Node, but this is a centralized & trusted (by all nodes) service that stores the records and handles requests from the Request Server

Component	Description
Request Server	The off-chain backend of a CN / TRS. This stores the pending requests from the user and is the only way to call a request to the CN / TRS. Each Request Server may have a different way to handle user requests (Voting, Direct Request, etc.)
Frontend Server	Provide the UX/UI for interacting with the Request Server

4.2 Database

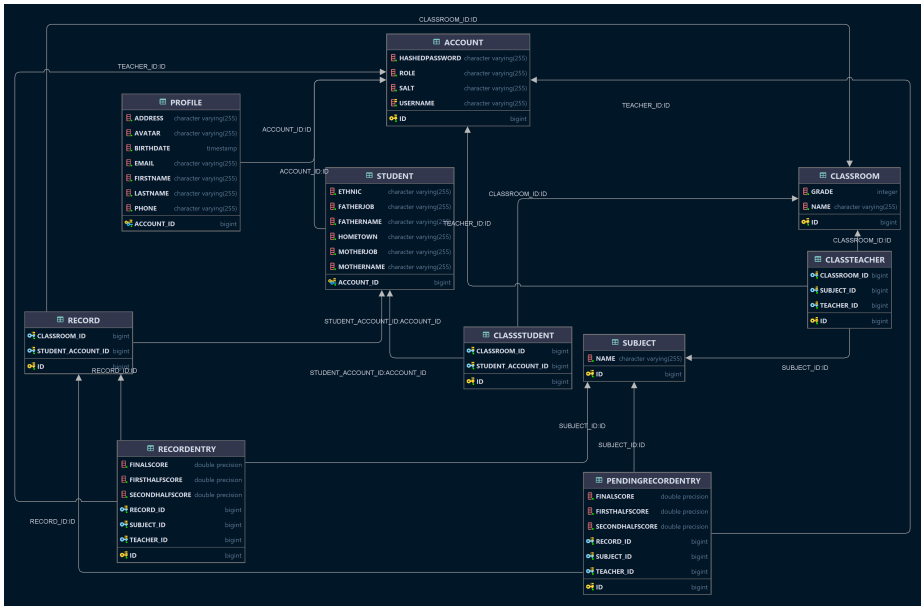


Figure 4.3: Database diagram

Chapter 5

Design Article (VI)

Bước đầu thực hiện số hóa hoàn toàn học bạ ở Việt Nam

Schools and departments are using modern technologies for storing and managing academic records digitally. However, the implementation is not consistent as many places still rely on traditional ways to proceed transcripts, and it has major problems like being centralized, third-party required, untraceable, untrustworthy, etc. This paper attempts to present an approach to the problem by listing the major characteristics of a reliable system; explain how and which Blockchain technologies, especially Hyperledger Fabric, can be applied to the situation; and provide three concepts: a centralized system, a multi-school network using Blockchain, and a multiple school-department one that expands the multi-school one and gives the storage and management to departments of education. This paper will be used as a foundation for similar topics and products to implement the concepts.

The number of academic institutions still use manual processes to store and transfer academic records like transcripts and certifications between institutions and to potential organizations, despite the fact that many large institutions are now adopting the modern practice of maintaining electronic academic records.The process can take up to several days, just for students who want to review their own transcripts, so a common transfer for a student can take anywhere from a few weeks to a month. Due to the time required to process and submit appeal requests using the widely used paper method, more serious errors could happen and the process could take several months. In addition to the significant wait time and the possibility of physical damage or loss of records during storage and transportation, there is also the risk of credential tampering by fraudulent parties. The cost of processing time, manual work effort, postage, and transit fees, as well as the storage and shipping of physical records, are also very expensive.

The emerging solutions are primarily based on email-based solutions or the transfer of PDF files while remaining limited by nationality, privacy and secu-

rity barriers. Although the popularity of cryptocurrencies and NFTs has led to the implementation of Blockchain as a host of applications in the financial sector, the field is more diverse in both technical and application areas [1]. As distributed applications are increasingly applied in various fields such as data storage (including handling medical records and healthcare [2,5]), Cloud and Grid Computing [3], e-vote [4] , Service for IoT [6], Banking system[7] and foremost is the field of Education. Academic institutions can benefit from blockchain technology to provide a decentralized and immutable ledger to confirm the integrity of academic records [10]. Then, solutions for storing and anti-fraud of online electronic degrees have also been conceived to bring the initial benefits of applying Blockchain technology [8][9]. While these solutions provide a more modern approach to the storage and the transfer of academic records, there are still limitations in terms of widespread adoption, auditability, and scalability. A successful solution for storing and exchanging electronic school records will include Security and Privacy, Scalability and at the same time benefit from the advantages of blockchain technology as Distributed, Transparency further described in Section II.

The goal of our proposed system is to address the limitations of existing solutions by utilizing Blockchain technology to provide a secure, verifiable, and tamper-proof method of storing, accessing, managing, and exchanging electronic school records between institutions. Part II will go over the technologies involved. Part III presents our proposed solution and design, and Section IV provides concluding remarks.

5.1 State of the art

In terms of digital solutions, in recent years, there have been technologies to manage and store academic records digitally. The biggest disadvantage of the above solutions is the centralization of data, so it will have a big impact if the system crashes, leading to data loss, in addition to the reliability of the data when it is stored by a third party.

One of the achievements of the online storage industry is the invention of Blockchain. This is a form of distributed database in which data is stored in a block and connects the blocks in a chain by hashing mechanism [11]. If we consider the solution of digitizing academic records alone, Blockchain has the following advantages [12]:

- Decentralized: Data is preserved in many member machines participating in the network. When data is lost, it can be restored from member computers.
- Transparency: The hashing mechanism makes the data in one block verified by other blocks, making the data immutable. Therefore, the data on the Blockchain network is verified, indestructible, and cannot be faked, so it has clarity, transparency, and can be trusted.
- No third party interference: Data is verified and stored by network members, different from traditional solutions when having to depend on a trusted third party as a storage place.

For transactions to take place on the Blockchain network, we have a Smart Contract mechanism. This is a pre-programmed set of rules to regulate the conditions and order of execution of a transaction on the Blockchain network, signed by the parties involved, verified by the members of the network, and without interference of third parties. It is used to process data on the network, and also set conditions that limit the execution of operations on data on the network. It is a self-run program on the Blockchain network so it is transparent [13].

Another important feature of a record management system is the privacy of students' personal information, which forces systems to have an authorization feature so that only authorized persons can view student information on transcripts. With Blockchain, there are a number of open source projects to implement the above properties. One of them is Hyperledger Fabric. This is an open-source project from the Hyperledger series about private and permissioned Blockchain. [14]. Unlike public Blockchains like Bitcoin or Etherium, where anyone can participate and make transactions, Fabric network members are registered for attention from a Member Service Provider. The Smart Contract mechanism is extensible to limit the access and data processing rights of members registered in the network. An example of applying Hyperledger Fabric in education storage can be said to be Sony Global Education when they choose to implement a cetification archiving system, which can limit data access for stakeholders [15]. For a solution to digitalize records, we can post a student's personal information and transcripts online and limit access to and edit records to only the school where the student resides. We can also create a feature that only exports grades from records stored on the network for statistics and analysis.

5.2 Approach & Concept

5.2.1 Centralized

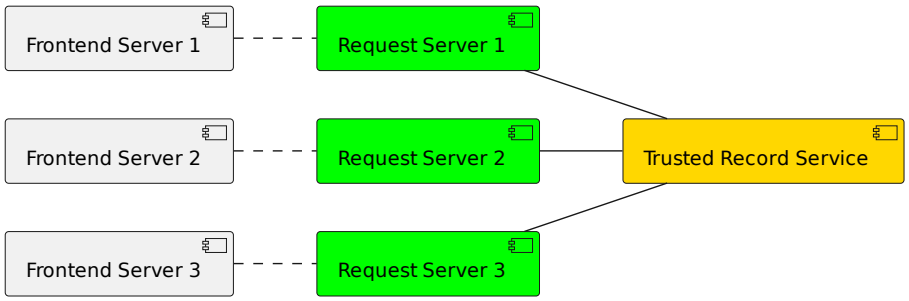


Figure 5.1: A centralized system

Figure 5.1 is an overview of our solution, represented as a centralized system, including:

- Trusted Record Service (TRS)

  This is the center of the system, which will store the records that have been verified by the schools, and the revision history of each record. This

center must be trusted by members of the system.

Only schools participating in the system can access the center to get records, and only schools associated with the record are authorized to get background information and submit a request to update and correct them.

- Request Server (RS)

  Each school in the system will be assigned a server to operate on the TRS, which contains a license indicating the school's identity and the permissions in the TRS.

  This is also the place to save the transcripts being edited and handle correction requests from users (teachers, students, etc.), including verifying the validity and accuracy of such correction requests, generating a complete transcript from valid requests to submit to the TRS with the license.

  Schools may have different ways of verifying the request for editing records, depending on the specificity of each one. Therefore, the Request Server of a school may be different, but the operation on TRS is the same because the verified transcript must be sent with a license from that school.

- Frontend Server (FS)

  This is the interactive server for the RS to retrieve, display the school records and send a request to edit the records on an easy-to-see, easy-to-interact interface. It can be part of the management page of the school.

  Since the RS of each school may be different, the interface of the FS may also be different to adapt to the specifics of that school.

The workflow of the system is shown in Figure 5.2

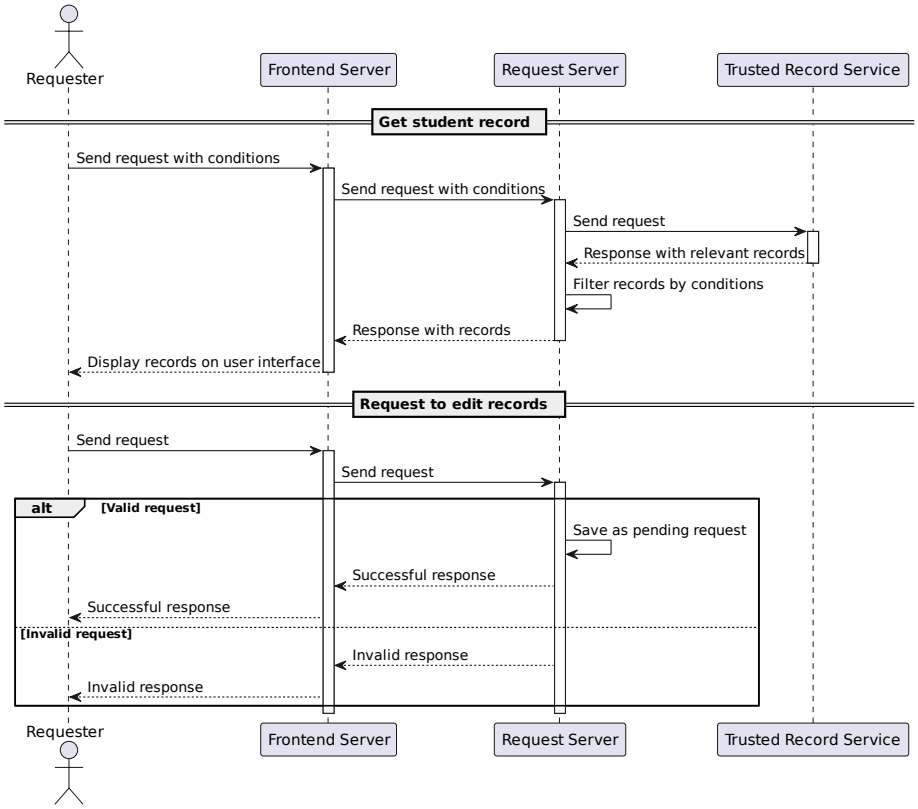
"Requester" is the student or teacher, and "Validator" is the homeroom teacher representing the school associated with the record requested to be edited in the Request Server.

5.2.2 Decentralized & Multiple schools

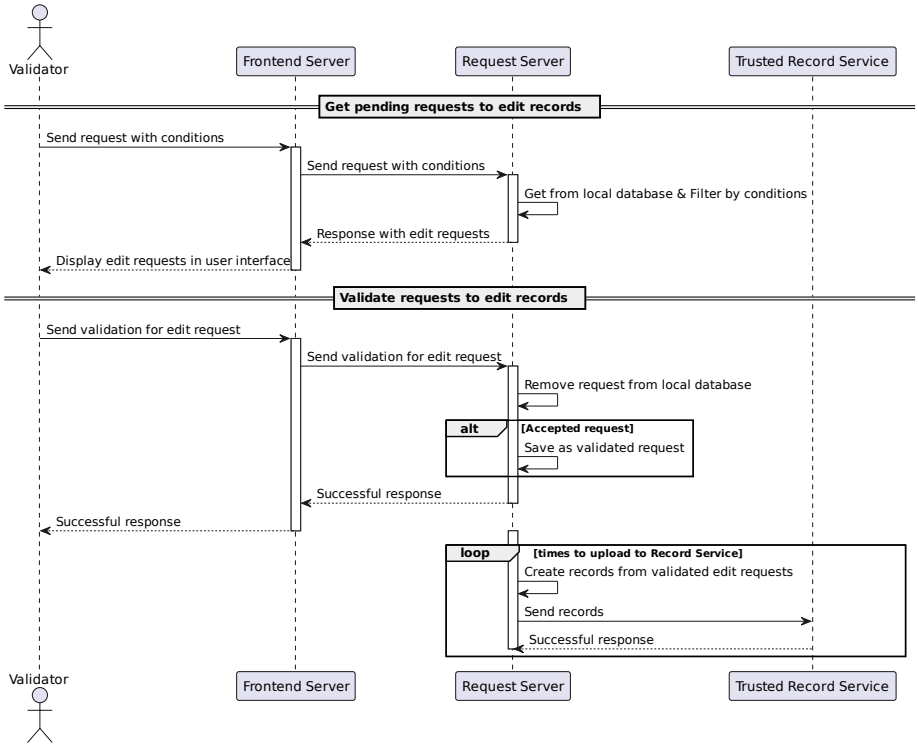
We can extend the system by applying decentralized data technology. Specifically, we can split Trusted Record Service into Chain Nodes and leave Chain Nodes for Schools like Figure 5.3. Now a Chain Node that receives a request to edit records from the Request Server will not only store it on that Chain Node, but will also send that request to neighboring Chain Nodes for storage.

This level is often used in private school systems that do not follow the formal school procedures according to the Department of Education, so the authority of the schools is equivalent.

A system at this level will ensure the security & persistency of academic record data in case a school's Chain Node goes down. If then, we can use the license from the Request Server to the neighboring Chain Nodes to retrieve the data to restore the collapsed Chain Node. However, the disadvantage is that each Chain Node will have to store a large amount of data from not only the school holding that Chain Node but also the data of neighboring Chain Nodes.



(a) From requester-view



(b) From validator-view

Figure 5.2: The sequence diagram of the system

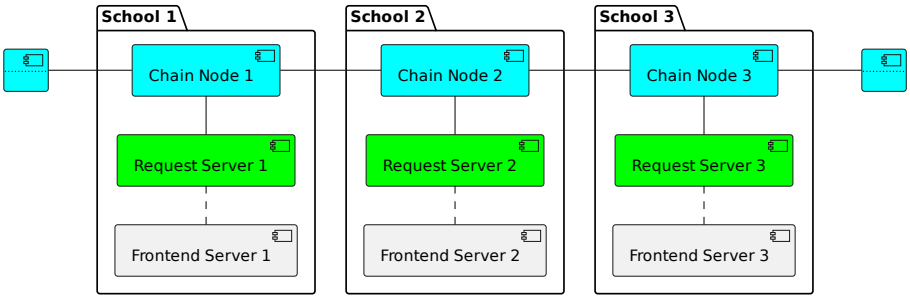


Figure 5.3: A decentralized system with multiple schools

5.2.3 Mutiple departments

We can level up the system to an inter-departmental scale so that each Chain Node will be maintained by each educational department like Figure 5.4.

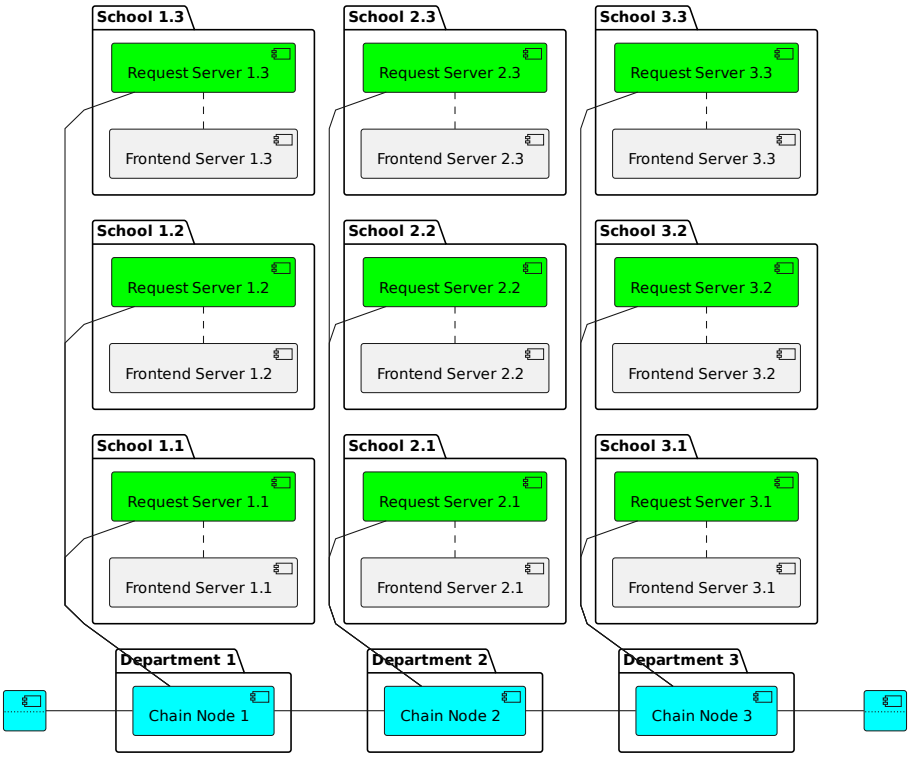


Figure 5.4: A inter-departmental decentralized system

In this system, the Chain Node will be initilized by a party trusted by the departments (usually the Ministry of Education), and those departments will issue licenses to operate on the Chain Node to the Request Server of schools belonging to that department.

The system at this level can be applied in regular schools, where the update of

records follows a strict process from departments to schools. This is also the most trusted system that can be used for online admission to the above schools because records at this level are guaranteed by both the school and the relevant educational department.

5.3 Conclusion

In this study, we presented a solution for managing and storing electronic academic records as a replacement for the traditional academic record based on distributed storage technology used by Blockchain, where the data is stored in a block and the blocks are connected on a chain by hashing.Our network enables us to manage data in the network using transactions via smart contracts. From there next, we show how to set up a multi-tier network and processes. Our network enables us to decentralize organizations and system users through arranging chain nodes, verifying transactions with smart contracts, archiving modification history and restoring data of a node using data from other nodes.

From this design, it can be concluded and proposed to organize a Permissioned Blockchain network with a multi-tier design. The main advantage of applying Permissioned Blockchain technology is its resistance to many threats and cyber attacks, rely on the hashing mechanism and the nodes on the Blockchain can prevent data breaches. And moreover, it offers a host of unique features such as improved reliability, better fault tolerance, faster and more efficient operation, and scalability.

And thus, the management of documents for the field of education has the potential to be significantly impacted by the integration of Blockchain, the hyperledger framework, and smart contract technologies across academic records.

5.4 Nguồn tham khảo

[1] M. Walport, Distributed ledger technology: Beyond Blockchain, UK Government Office for Science (2016).

[2] Duong-Trung, N., Son, H. X., Le, H. T., & Phan, T. T. (2020). Smart Care. Proceedings of the 2020 4th International Conference on Cryptography, Security and Privacy. https://doi.org/10.1145/3377644.3377667

[3] Liang, Xueping & Shetty, Sachin & Tosh, Deepak & Kamhoua, Charles & Kwiat, Kevin & Njilla, Laurent. (2017). ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability. 10.1109/CCGRID.2017.8.

[4] Vanita Jain, Akanshu Raj & Abhishek Tanwar & Mridul Khurana. "eVote – A Decentralized Voting Platform". Digital Twin Technology, CRC Press, 2021.

[5] Mettler, M. (2016). Blockchain technology in healthcare: The revolution starts here. 2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom). doi:10.1109/healthcom.2016.7749510

[6] Samaniego, M., & Deters, R. (2016). Blockchain as a Service for IoT. 2016 IEEE International Conference on Internet of Things (iThings) and IEEE

Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). doi:10.1109/ithings-greencom-cpscom-smartdata.2016.102

[7] Chowdhury, M. , Suchana, K. , Alam, S. and Khan, M. (2021) Blockchain Application in Banking System. Journal of Software Engineering and Applications, 14, 298-311. doi: 10.4236/jsea.2021.147018.

[8] Ghaffar, A., & Hussain, M. (2019). BCEAP - A Blockchain Embedded Academic Paradigm to Augment Legacy Education through Application. Proceedings of the 3rd International Conference on Future Networks and Distributed Systems - ICFNDS '19. doi:10.1145/3341325.3342036

[9] Vidal, F. R., Gouveia, F., & Soares, C. (2020). Revocation Mechanisms for Academic Certificates Stored on a Blockchain. 2020 15th Iberian Conference on Information Systems and Technologies (CISTI). doi:10.23919/cisti49556.2020.9141088

[10] M. Sharples and J. Domingue, "The Blockchain and Kudos: A Distributed System for Educational Record, Reputation and Reward," in 11th European Conference on Technology Enhanced Learning, Lyon, France, 2016.

[11] Nofer, M., Gomber, P., Hinz, O., & Schiereck, D. (2017). Blockchain. Business & Information Systems Engineering, 59(3), 183–187. doi:10.1007/s12599-017-0467-3

[12] Lu, Y. (2019). The Blockchain: State-of-the-Art and Research Challenges. Journal of Industrial Information Integration. doi:10.1016/j.jii.2019.04.002

[13] Singh, A., Parizi, R. M., Zhang, Q., Choo, K.-K. R., & Dehghantanha, A. (2019). Blockchain Smart Contracts Formalization: Approaches and Challenges to Address Vulnerabilities. Computers & Security, 101654. doi:10.1016/j.cose.2019.101654

[14] Androulaki, E., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., … Laventman, G. (2018). Hyperledger fabric. Proceedings of the Thirteenth EuroSys Conference on - EuroSys '18. doi:10.1145/3190508.3190538

[15] "Sony Global Education Develops Technology Using Blockchain for Open Sharing of Academic Proficiency and Progress Records". Sony Group Portal - Sony Global Headquarters, http://www.sony.com/en/SonyInfo/News/Press/201602/16-0222E/index.html.

Chapter 6

Hyperledger Fabric - Road Map

6.1 Key Concepts

- Wiki
- Short Version
- Simplilearn
- Demystifying Hyperledger Fabric
  - Part 1
  - Part 2
  - Part 3

6.2 Develop Chaincode (Smart Contract) /w Java

- Writing Your First Chaincode
- Running a Fabric Application
  - Source code (Asset Transfer)
- Fabcar (Chaincode)
- Secured asset transfer
  - Source code (Go)
- Using Private Data
  - Source code
- ABStore

6.3 Setup Network /w Minifabric

- Playlist
- Docs
- Deploying a production network
  - Only for reference

6.4 Develop Client /w Java

- Fabcar (Client)
- Fabric Gateway SDK
  - Javadocs
  - Reference
- Asset Transfer

References