electronics

MDPI

Article

Toward Building Smart Contract-Based Higher Education Systems Using Zero-Knowledge Ethereum Virtual Machine

Dénes László Fekete [1] and Attila Kiss [1,2,*]

[1] Department of Information Systems, ELTE Eötvös Loránd University, 1117 Budapest, Hungary
[2] Department of Informatics, János Selye University, 945 01 Komárno, Slovakia
* Correspondence: kiss@inf.elte.hu

Abstract: The issuing and verification of higher education certificates, including all higher education documents, still functions in a costly and inappropriately bureaucratic manner. Blockchain technology provides a more secure and consistent way to revolutionize the widely used generalized mechanisms and system concepts. In this paper, the most necessary requirements are examined regarding a blockchain-based higher education system, based on the most well-known research papers. Moreover, the opportunities of working on an education system by maintaining a decentralized structure organization are recommended as well. This paper recommends the most suitable blockchain scaling solution for the architecture of an education system which uses the most state-of-the-art EVM (Ethereum virtual machine) compatible approach to implement the higher education system with all the predefined requirements. It is proven that the explained smart contract-based higher education system, which uses zkEVM (zero-knowledge Ethereum virtual machine), consists of all necessary functionalities and satisfies all predefined requirements. In fact, the recommended system, by using a modular blockchain structure, implements all the functionality and capability of the examined related works in one system, namely GDPR (General Data Protection Regulation), which is compatible and more secure.

Keywords: higher education system; verification; smart contract; security; decentralization; blockchain; zero-knowledge proof; Ethereum virtual machine

Citation: Fekete, D.L.; Kiss, A. Toward Building Smart Contract-Based Higher Education Systems Using Zero-Knowledge Ethereum Virtual Machine. *Electronics* **2023**, *12*, 664. https://doi.org/10.3390/electronics12030664

Academic Editor: Xiangjie Kong, Priyadarsi Nanda and Ana Rosa Cavalli

Received: 9 January 2023
Revised: 24 January 2023
Accepted: 26 January 2023
Published: 28 January 2023

1. Introduction

Academic documents, degrees, and certifications provide undoubtable proof of a student's competence and accomplishments. The long-term tamper-proof ownership and validation of these documents play a vital role in a student's academic and professional career. The responsibility of issuing, storing, and archiving these documents falls in some manner onto HEIs (higher education institutions), which also usually determine, define, and modify the used framework and formats. The issuing and verification of these documents are highly dependent on the given HEIs' financial and geopolitical state [1]. The increasingly digitized world, having come about as a result of the COVID-19 pandemic, requires total digital access to all documents and services accessible previously. However, digitization does not by default imply stronger and more secure technological systems [2,3]. Overcoming these issues necessitates the use of the most accurate state-of-the-art technology in a global system to achieve the expected requirements before and after graduation.

The goal of this research paper is to use a modular blockchain stack and zkEVM (zero-knowledge Ethereum virtual machine) to create a permissionless verifiable education system in which the trust assumption can be lower between the entities and stakeholders than in other education systems known so far. Furthermore, the hybrid approach can reduce the attack vector and minimizes the possibility of malicious behavior.

From the point of view of verification, it is important to store as much data as possible about students on-chain [4–6]. Thus, the history of the academic interactions of students

can be traced back and validated. The verifiability of the related previous events also contributes to the verifiability of the current event. Therefore, it is essential to define the possible interactions (functionalities). Moreover, due to the immutable characteristic of the blockchain, there is a greater possibility to understand and support students' interactions [7–9] before, during, and after their learning process [10]. In addition, the proposed system provides users control of their data, and as such they can self-manage their digital data in a self-sovereign manner.

The structure of this paper is shown in Figure 1. Firstly, the current state of certification, of verification, and issuing is presented. In the next section, the current higher education research papers related to blockchain technology are examined. During the examination, the necessary requirements and essential functionalities for the recommended system are determined based on the surveys. After that, the concepts related to blockchain technology and the corresponding scaling solutions are briefly introduced, which are essential for understanding the later discussed design concept. After these, the components of the modular structure-based higher education system are introduced. A PoC (Proof of Concept) application is introduced, which is based on the recommended system. Then, the details of the recommended system are analyzed and discussed. Then, in the end, the future work and the corresponding possibilities are explained.

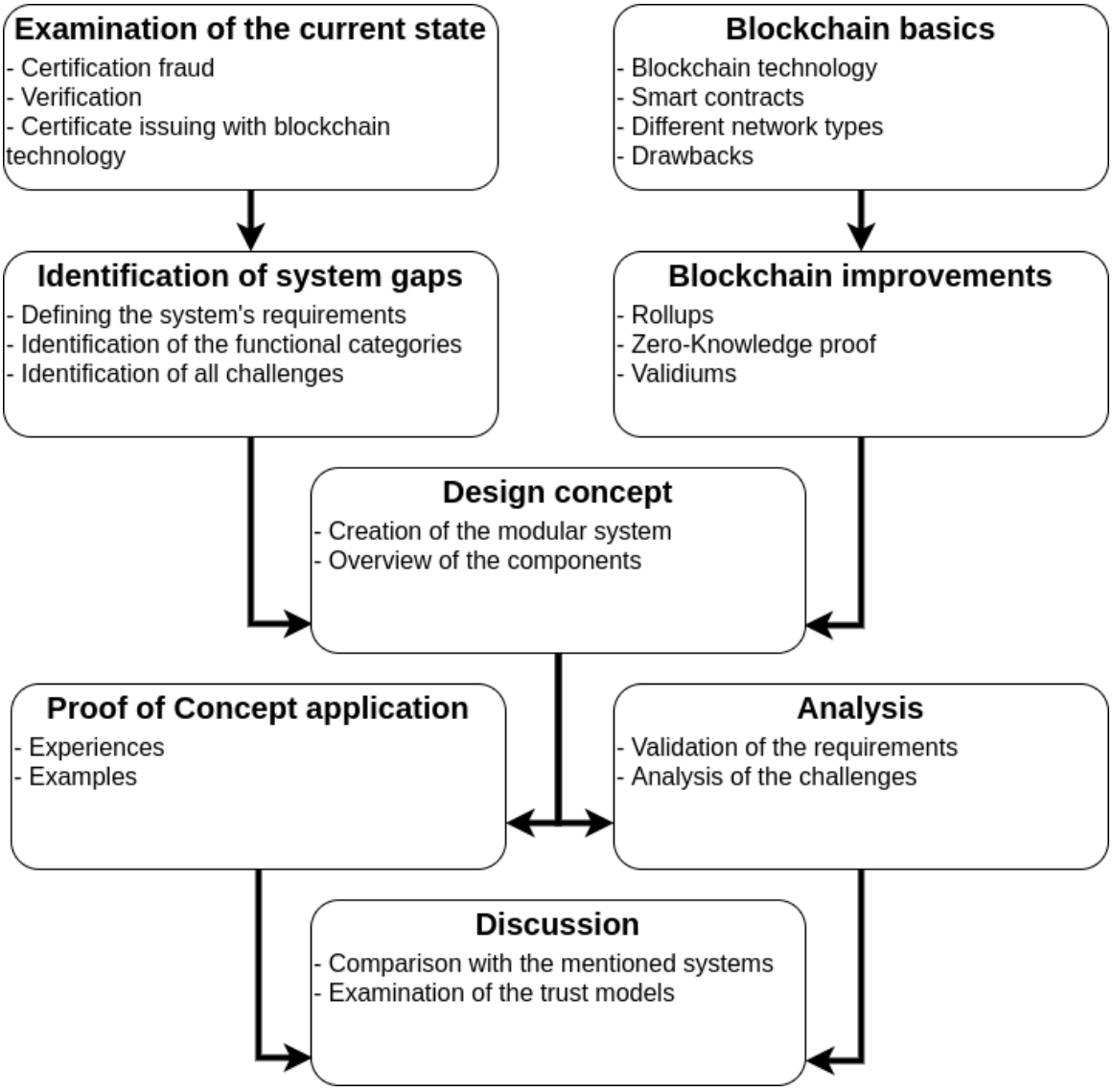

Figure 1. The structure of the paper.

2. The Current State of Certification Issuing

Examining European trends based on the information available on Eurostat, on average 4.5 million people receive a higher-education diploma every year in Europe. Between 2002 and 2016, the number of people who obtained a degree aged between 30 and 34 years old consistently increased. At the end of the examined period, this number had increased by 16% [11]. The labor market clearly demands and highly values [12] job-seekers who have one or more degrees. It is a necessary requirement for applicants who wish to work in their desired field. The processing time of these submitted applications is significantly high, due to the time-consuming method of authentication and the communication between assumed trusted third parties. In fact, this kind of process usually debits former students with financial obligations. Despite today's 'instant world', this process requires an inordinate amount of time and trust in untrusted third parties.

The official certification and proof of professional qualifications immediately provide better opportunities and further possibilities in the labour market; there is a proven parallel and correlation between the two [12,13]. Migration statistics reveal that close to 30% of first- and second-generation migrants possess a higher education degree. This trend implies the necessity of interoperability, which is transferable over borders and between HEIs. This necessity is highly important for them in the short (between HEIs) and long (certification validation in other countries) term. The success of interoperability does not depend on only one HEI, as the connection network is more complex and the number of corresponding entities is enormous. Low-income countries have a giant drawback from a financial point of view in maintaining and managing an up-to-date educational information system, as a result of their unstable political regimes, for example.

The necessity for effortless interoperability between HEIs in different countries can also be observed in Erasmus mobility in Europe as well. During this program, HEIs send their own students or receive students from foreign HEIs for HEI part-time education, at most one year. A mandatory requirement of this program is the accreditation of the received credits from the foreign university at the university of origin [14]. Without any global verification system, this process requires significant avoidable administration. However, there is already a defined unified credit system and format [15] between European HEIs. The used framework for supporting this process of accreditation is not widespread and it is also not fully automated. There is no connection between this framework and the HEIs' information system.

2.1. Verifications

The main limitation of the current certification verifications is the time-consuming, high-cost, and inefficient process regarding problematic ownership guarantees. A lack of transparency and availability, as well as the dependency on a trusted third party during the verification process, are common [16].

The adoption of digitalization started a decade ago. Storage and verification happen simultaneously in the online space, but trust in this kind of system is still low and ambiguous as a result of system errors, crashes, and hacking [17]. In general, an HEI uses different types of customer relationship management software and stores important information in more than one database. This type of traditional approach results in a fully centralized system where only the system maintainer controls the whole system [18]. A more distributed approach is indispensable for strong consistency in student information, smooth communication between entities, and simple certification verification.

In this corresponding paper [19], the researchers examined the verification habits through well-defined corresponding interview questions. The results show that the usual time period for verification is 12 days. Not only is the current system cost-inefficient, but it proves to be time-consuming also, as a result of the long verification chain. Another interesting result from this paper is that in the current digital world people still prefer paper-based certifications and attempt to validate the certainty of this form. Only in one of four cases arose the need for any further validation from the issuing HEI.

Both HEIs and students require the ability to prove that the legally authorized entity is in fact the issuer of a given certification [20]. Without it, verification cannot be considered complete.

An incomplete approach to educational systems is one in which the system does not include all available information about all students. The tracking of learning and actions of each student is necessary, as this is evidence of their performance for the student, HIS, and an external entity [21].

2.2. Certification Fraud

The fabrication of official certificates, most often a degree, happens in the labour market during a job application. CV fraud has negative consequences for all participating entities. It has negative impacts on the successful or the unsuccessful enrollment [22]. The previously mentioned statistic [19] points to how accessible and easy it is to commit this type of crime. Due to the previously mentioned statistics, since certifications embody values, applicants either lie about their skills and certifications or seek the black market. In this way, the black market generates continuously increasing profits year to year [23].

The United States leads globally in the number of illegitimate institutions with more than 300 currently in operation and also has more than 2 million fake degree certificates in circulation according to Grolleau [24]. It is followed by the United Kingdom, in which about 270 fake institutions operate [25]. These cases are not standalone; in Australia, up to 35% of candidates submitted falsified academic credentials in job applications [26]. Globally, the ACFEs (Association of Certified Fraud Examiners) found that 41% of job applicants submit fraudulent certificates annually [16]. A similar survey found that misleading or fictitious educational credentials and experience were presented by most candidates [16].

The fight against and the preparation for certification fraud is a necessary, although significant (GBP 40,000 a year) expense for larger companies, as it can address the inequality between the competing candidates and the dishonest manner of the selection process [19,27]. Globally, the cost to organizations of certificate fraud can reach around USD 600 billion every year [26].

It follows from the above findings that the problem of certification fraud has reached serious and alarming proportions and needs to be addressed urgently. An enormous number of research papers and statistics imply that the success of a reliable verification system is unavoidable for certifications due to industrial pressure.

2.3. Certificate Issuing with Blockchain Technology

Considering the current situation, HEIs are using traditional centralized applications and databases. Thus, the entities in the system raise concerns regarding the real level of privacy and note the absence of control over their interactions and data, and as a result, suffer the disadvantage of the centralized structure [28].

Achieving security, privacy, trust, and equality in the present century can be achieved with blockchain technology. Higher education is an area where the application of blockchain can provide users with a wide range of benefits, helping to promote mobility and avoiding hierarchic bureaucracy, anxieties, and hardships [29].

Personal data are not under the control of any third-party organization. From a privacy perspective, certificates are much more susceptible to data leaks in terms of personal information [30]. The core requirements [30] for an education system, which can be fulfilled with the use of blockchain technology, are strong links between entities, secure sharing, improved visibility, and certification of authenticity. However, in addition to these, we must pay attention to the resolution of the five limitations [16] mentioned by the authors that are present in traditional verification systems. This is all to eliminate the dependence on third parties, in order to have the verification process be more cost-effective and time-efficient and to develop a system in which ownership means an actual owner, not a dependence on the issuer.

3. Related Works

3.1. Functional Categorization

The application of blockchain technology in higher education is a rather actively researched area. Several surveys and reviews have been written in this field, but only some of them have the exact categorization of research papers. These collected surveys [31–35] categorize research papers on the higher education field into different categories according to their functionalities.

It is necessary to examine these categories because some categories do not belong to the traditional HEI system or there is an overlap between several categories. Specific functionalities such as providing feedback services, job opportunities, cooperative learning (new network of cooperation between students and professors in [35]), and secure content library described in [31–33] are not further explained or considered, as a smart contract-based system implies the implementation realities of each functionality. Thus, these functionalities do not have fundamental requirements against the design of the system architecture. Based on a well-designed open higher education system, the mentioned functions can be implemented as external services, as they are not embedded in the higher education system nor part of the basic processes of the higher education system. The accreditation of educational institutions [35] are not corresponding to the HEI system because this is on a higher level of abstraction.

The protection of intellectual property (copyrights) mentioned in [31,35] is not further explained, since this functionality is not the task of a generalized education system which is used by an HEI to fulfil and solve this problem. Global, worldwide systems are used for solving this problem which is independent of the higher education system [36,37].

The remaining categories make up the functional requirements of an education system that can verify all related information from the first interaction of a student to the last interaction. These functionalities help to determine which interactions are required on-chain to create a more trusted system. The categories from the mentioned surveys are presented in Table 1. These categories can be transformed into five new functionalities:

- Certificates issuing and verification;
- Student assessments and exams;
- Data management;
- Credit transfer/interoperability;
- Admissions;
- Payments.

Table 1. The different categorizations from surveys.

Author	Title	Categorizations
Alammary (2019)	Blockchain-based applications in education: A systematic review [31]	• Certificates management • Competencies and learning outcomes management • Evaluating students' professional ability • Securing collaborative learning environment • Protecting learning objects • Fees and credits transfer • Obtaining digital guardianship consent • Competitions management • Copyrights management • Enhancing students' interactions in e-learning • Supporting lifelong learning
Loukil (2021)	Blockchain adoption in education: A systematic literature review [32]	• Certificate/degree verification and revocation • User-centric educational record management • Students' professional ability evaluation • Blockchain-based educational institute systems • Online learning environment

Table 1. *Cont.*

Author	Title	Categorizations
Hameed (2019)	A review of blockchain-based educational projects [33]	• Content library • Storage of personal data • E-certificate • Scoring system • B2B approach • Token system • Cooperative learning • Job opportunities • Providing feedback services
Awaji (2020)	Blockchain-based applications in higher education: A systematic mapping study [34]	• Certificate/degree verification • Student assessments and exams • Credit transfer • Data management • Admissions • Review papers
Fedorova (2020)	Application of blockchain technology in higher education [35]	• Issue and storage of certificates and diplomas • Identification solutions • Protection of intellectual property • New network of cooperation between students and their professors • Formation of an academic passport (portfolio) • Payment for studies with a cryptocurrency • Accreditation of educational institutions • Administration of the educational process

3.1.1. Certificates Issuing and Verification

The most discussed field in the application of blockchain technology in higher education [38,39] is the issuing and verification of certificates. During the verification process of a certification, a third party, one which is not the owner and not the issuer, wants to validate the authenticity of the certificate whilst avoiding any privacy concerns the owner may have, and any trust assumption regarding the issuer. In addition to the degree, all the issuing and verification of official documents which correspond to a student are included as a requirement. In general, the most common approach is to separate the verification and all other functionalities into different systems. Thus, a hash of certification appears in the decentralized network with an identifiable and unverifiable history by an external event. In the better cases, they usually try to compress the hashes through different compression mechanisms to approach a more cost-efficient solution but to also create a strong trust assumption with a third party during a business-to-business process. An overview of the examined papers is presented in Table 2.

Currently, the dissemination of non-standard standards continues, foregoing the use of blockchain technology, for example, BADGR and Mozilla Open Badge within the EU (European Union) [40]. The issues of ownership, verification, and interoperability that arise with this approach could be solved with the use of a general protocol and cryptographic signatures. At some universities, the issuing of blockchain-based digital certificates is currently being tested in parallel with the traditional system, such as at the Malta College of Arts, Science & Technology, where the Blockcerts-based [41] digital certificates system is used [42], or The University of Nicosia and the Massachusetts Institute of Technology, where the mentioned standard was first implemented and used [42,43].

Independently of HEIs, various research groups are also actively addressing this topic in order to achieve a stronger and more secure decentralized system with a higher transaction rate [44]. In this paper, the verification and sharing of issued certifications are conducted using smart contracts in a blockchain network. Important personal data are still stored in a centralized database, but verification can be achieved in a peer-to-peer

network. Furthermore, the cost of the transactions is high due to the necessary constant communication with the smart contracts, as the Ethereum ecosystem is neither optimized nor capable of these types of operations and interactions.

During verification, independent user validation is typically missing and only the shared paper includes personal data. Privacy concerns are rather common [45,46] due to the public characteristics of the blockchain, the usage of which, by default, cannot be GDPR (General Data Protection Regulation) compatible. In the first place, few document management systems deal exclusively with GDPR compliance. Due to the lack of personal data and user validation, it is difficult to establish a clear correspondence between the certificate and its owner.

The issuance of certification can be due to an event, the logic of which is coded in a smart contract. This event can be initiated by an HEI or a student [47,48]. In this way, a more automatic and transparent issuing system can be implemented, where a student is responsible for their own certification process and state in a self-sovereign manner. This process can be further improved from a security perspective by using only on-chain data, since there is no invalid or untrusted data which is coming from outside the system [49]. Another approach is possible by using multi-signature contracts [4], in which an event is approved by entities with the appropriate rights, thereby reducing fraud from a single point of failure.

Table 2. Certificates issuing and verification summary table.

Author	Title	Findings
Rahardja (2021)	Immutable Ubiquitous Digital Certificate Authentication Using Blockchain Protocol [46]	Encryption by public key, permissioned network between HEIs
Cheng (2018)	Blockchain and smart contract for digital certificate [47]	Randomly generated unique id, the Ethereum Mainnet, files in a centralized database
Srivastava (2018)	A Distributed Credit Transfer Educational Framework based on blockchain [4]	Centralized key distribution, permission network, the longest chain problem, no smart contract capability, no encryption
Han (2018)	A novel blockchain-based education record-verification solution [44]	Files in a centralized database, permissioned network, smart contract-based policies
Kistaubayev (2022)	Ethereum-Based information System for Digital Higher Education Registry and Verification of Student Achievement Documents [6]	Files in a centralized database, smart contract based on the Ethereum Mainnet
Ayub Khan (2021)	Educational Blockchain: A Secure Degree Attestation and Verification Traceability Architecture for Higher Education Commission [39]	Permissioned Hyperledger Fabric network between HEIs, encryption for public storing
Vidal (2020)	Blockchain application in higher education diploma management and results analysis [38]	Hashes in a decentralized network, reliable timestamps for revocation

3.1.2. Student Assessments and Exams

Well-designed student assessment and exam functionalities in the system provide a secure communication platform for students and teachers. They facilitate the establishment of a more transparent mode of examination [5,50,51], but reduce the possibilities regarding forms of exams, as only questions with multiple-choice can be automated, as the correction of an essay form of the exam requires the interaction of an external entity with the given smart contract. This breaks the intention of automatization. In traditional HEI structures, there is little chance of this form of examination being fully adapted [52], but the usage of smart contracts to increase the performance of administrative tasks related to courses is fully suitable. It can help student admissions to given courses and help teachers with the administrative work related to courses [5,15,53,54]. In addition, a student can easily submit

a verifiable certification related to their courses with a chosen entity. A summary of the examined papers is presented in Table 3.

Table 3. Student assessments and exams summary table.

Author	Title	Findings
Morisio (2018)	Blockchain6based storage of students career [54]	Permissioned Ethereum network, centralized key distribution, smart contract-based policies
Wu (2021)	The application framework of blockchain technology in higher education based on the smart contract [53]	Permissioned network, a new contract for each new task or exam
Turkanović (2018)	EduCTX: A blockchain-based higher education credit platform [15]	Permissioned network between HEIs, centralized key distribution, centralized third-party service, tokenization
Lizcano (2020)	Blockchain-based approach to create a model of trust in open and ubiquitous higher education [52]	Permissionless network, not a traditional approach, a new structure for HEIs
Shen (2018)	Research on online quiz scheme based on double-layer consortium blockchain [51]	Permissioned network, double-layer consortium blockchain, single quiz
Ramos-Sosa (2020)	Blockchain and smart contracts for education [50]	Permissioned network, generated correction key
Bhosale (2021)	Revolutionizing Verification and Management of Educational Certificates with Self-Sovereign Student Identities using Blockchain [55]	Permissioned network, dependency from the government
Haïdar AM (2021)	The future of university education: Examination, transcript, and certificate system using blockchain [5]	Permissioned network, almost all functional requirements

3.1.3. Data Management

Many different types of data management techniques appear in higher education systems, as shown in Table 4. The architecture of data management can be in a centralized or decentralized manner. From the point of view of the main blockchain, the data are stored on-chain or off-chain. From a privacy point of view, they are either fully public or fully private, or the given storage format of the data is GDPR-compatible. The most common problem is the public on-chain storage of all data, which, in addition to having many privacy concerns, is also an extremely expensive approach [56]. In a permissioned manner, a centralized distributed system is implemented with blockchain characteristics [57,58]. The solutions mentioned in the papers [38,59,60] are less secure compared to a decentralized approach due to this centralized characteristic. Forged certificates are easier due to a too high trust assumption in centralized backend databases, which causes single-point-of-failure concerns as well. Moreover, revocation and verification are also impossible if malicious behavior is present in the system.

3.1.4. Credit Transfer/Interoperability

The need for simple interoperability between HEIs and the sharing of student data, such as transcripts of records with credits, has been present for decades. The need is constantly increasing due to the growing popularity and emergence of new exchange programs.

Table 4. Data management summary table.

Author	Title	Findings
Santos (2019)	A Decentralized Approach to Blockcerts Credential Revocation [59]	Based on Blockcert, revocation capability, on the Ethereum Mainnet
Ataşen (2020)	Blockchain-Based Digital Certification Platform: CertiDApp [56]	Smart contracts on the Ethereum Mainnet, files in a centralized database
Zhai (2022)	TVS: a trusted verification scheme for office documents based on blockchain [57]	Storing and verification on a permissioned HyperLedger Fabric network
Das (2022)	A blockchain-based integrated document management framework for construction applications [58]	Version-controlling system on a permissioned HyperLedger Fabric network, files in a centralized database

Data management mechanisms and the possibilities of effortless interoperability are closely related, since well-designed data management carries within itself simple widespread adaptation and interoperability as well. The adoption of completely unnecessary blockchain functionalities and protocols can be found among currently available research papers. An HEI credit in itself has an exact value by definition which is not transferable and non-fungible. It symbolizes a student's completion of a certain course. The work [4] tries to bridge the problem of interoperability using a tokenized implementation of credit transfers. In another research paper [15], a similarly tokenized approach was used for HEI credits, but following the European Credit Transfer and Accumulation System. Both of the mentioned approaches are based on storing transcripts of records. Thus, there is the opportunity to transfer credits, the corresponding information of courses, and all data about a given course, between HEIs with verifiable proof and strong transparency.

3.1.5. Admissions

The application process for an HEI can often be problematic and opaque. Applicants often do not know what requirements they must meet and what official documents they must submit to become successful applicants. This task is often the duty of the HEI's information system if there is no centralized national system in use. With the use of blockchain technology in this field, there is the opportunity to monitor and follow the real-time application status and have greater visibility over the application process [61]. In the research paper [62], instead of the Indian scholarship system, a smart contract-based system is proposed, which eliminates the inconsistencies of the current system and provides the possibility to follow and monitor processes. In addition to all this, a well-defined HEI admission system can simply facilitate the registration of new students [63]. This usually involves a student's public key registration supervised by an authority or a group of authorities. During both mentioned processes, it is important that the data management privacy aspect satisfies the given requirements. An overview of the examined papers is presented in Table 5.

Table 5. Admissions and payments summary table.

Author	Title	Findings
Mori (2019)	Digital university admission application system with study documents using smart contracts on blockchain [63]	Permissioned Ethereum network, tokenization, on-chain storing
Bedi (2020)	Smart contract-based central sector scheme of scholarship for college and university students [62]	Permissioned Ethereum network, supporting payments
Curmi (2018)	Blockchain-based certificate verification platform [61]	Smart contracts on the Ethereum Mainnet, centralized service
Rooksby (2017)	Trustless education? A blockchain system for university grades [64]	Smart contracts on the Ethereum Mainnet, supporting payments, tokenization
Rashid (2019)	TEduChain: A platform for crowdsourcing tertiary education fund using blockchain technology [65]	Permissioned network, supporting payments

3.1.6. Payments

Despite the fact that the use of blockchain technology is closely related to the fields of accounting and finance [21,27,66], it is rarely present in the applications of the examined research papers that focus on one problem. Payment transaction support is not the usual functionality of a blockchain-based educational system. Despite this, there are continuous and regular cash flows within the HEIs and with entities outside the HEIs. Currently, transactions within the HEI are carried out with the help of a trusted third party and accounting approval. This time-consuming process results in great dependence, inconsistency? and impenetrability, as mentioned in the paper [62]. The most common approach regarding this category is very futuristic, as self-issuing cryptocurrency [67] referred to as the "learn to earn" mechanism is usually used and implemented in blockchain-based education systems. With the help and motivation of tokens as money, the students are encouraged to achieve better results, to work in groups, or to do more social work [64]. This inspiration capital can come from external organizations [65,68], outside of the HEI. Even now, there are various scholarship possibilities. There is no need to create a new scheme or mechanism. It is more appropriate to concentrate on other aspects of this field, as a given HEI can simply process payment transactions within the institution itself without an untrusted third party. The payment support results in a more secure system with less trust assumption and less human administrative work.

3.2. Challenges

The characteristics of Blockchain technology are also its main challenges, which must be overcome in some way or another as they are fundamentally not suitable for application in a higher education system. The mentioned surveys overview the main challenges of blockchain technology adoption in an HEI.

3.2.1. Privacy

Due to their architectural approaches, education systems based on blockchain technology do not have complete anonymity. Ledger history, all transaction data, can be traced back by everybody, as everything is public and there is no privacy protection implemented. Moreover, the use of any kind of privacy protection always has corresponding cost overheads. Some aspects, for example private smart contracts, have no straightforward implementation, due to their very complex and complicated requirements. From a higher education system's point of view, it is imperative that users should be capable of maintaining their privacy regarding their transactions and their data [44].

3.2.2. Immutability

Due to the characteristics of blockchain technology, we cannot withdraw data from the database, since the mutually dependent block structure immutably [38] does not make this possible. For this reason, data protection in an on-chain manner is impossible due to the limitations of the user's operations and the lack of privacy protection.

3.2.3. Blockchain Usability

Regarding user adaptation, it is an inescapable challenge to make new technology easy to use. The user must fulfill many complicated and complex prerequisites in order to be able to interact with the blockchain-based network. Challenges that arise in blockchain usability depend on the user, and since there is a lack of necessary individual competence, adaptation cannot be achieved. Based on the interview questions in research papers [19,35], half of the respondents are unfamiliar with blockchain technology. Moreover, in the research paper [19], they specifically asked about the knowledge of the wallet which is a mandatory attribute for the usage of a blockchain network. A total of 30% of the respondents can apply and use a wallet. It is difficult to solve the problem of a lack of necessary knowledge from the point of view of technology. The only way is to have a greater focus on application interface design, to assist and provide a smoother user experience, and the possibility of easier adoption [69]. The support and organizational enthusiasm of the HEI's management significantly help blockchain adaptation [70]. Since there is no new functionality, only the extension of the existing ones, in this case, the adaption of the proposed system is more simplified [71]. The success of the adaption depends on the users' motivations and attitudes toward the use of the new system and operations, but the majority of the academic community is unaware of this technology [62].

3.2.4. Cost

Every transaction in a decentralized network has a cost, which is not present in a traditional education system. Furthermore, the architecture is more complex and it has higher hardware requirements than a traditional approach. Therefore, well-designed and optimized data management is essential for a cost-efficient system when considering resource costs and transaction costs [72].

3.2.5. Scalability

In a decentralized network, transaction congestion is a usual event as a result of its permissionless manner, as has been previously mentioned. Therefore, the time of processing a transaction and transaction latency increase. Transaction congestion cannot be solved by a single entity, because the given network is not controlled by a single entity. Only higher transaction cost can influence the result of transaction processing time. Regarding the blockchain trilemma, scalability tends to decrease due to the exclusive support of decentralization and security aspects. As a result, network application fields are reduced due to scalability limitations and inadequate performance.

The mentioned research papers in the different categories are worrisome in many places, such as security, scalability, and decentralization. It is difficult to adapt privacy-sensitive data into a blockchain system so that the system can be well interoperable between several entities, as well as having an appropriate transaction rate for the application. However, with a suitable modular architecture and the determination and variety of the stakeholders and entities, the system can include the mentioned functionalities and features. Moreover, it can also fulfill the mentioned requirements and solve the mentioned challenges.

4. Blockchain Basics

Blockchain technology has an enormous number of definitions which spread across the world wide web, as everybody attempts to take advantage of the current trends and have it be applicable for their needs. The most accurate definition is that of distributed ledger technology, which is replicated and shared among the members of a network. The unit of

the distributed system is the immutable block. Decentralization, as a mandatory attribute, is ambiguous as a result of the various enterprise adaptations and interpretations, as so-called blockchain-based systems often feature centralized distributed systems. Decentralization is the basic pillar of technology in the classical sense, and even in later constructed systems by organizations, decentralization is present at some level. Decentralization is the creation of a permissionless, open, and trustless network where there is no central authority and decision-making is community-based. The entities which take part in the maintenance of the system resolve issues in a democratic way.

The first appearance of blockchain technology was in the Bitcoin White Paper published under the pseudonym Satoshi Nakamoto in 2008 [73], in which he described how cryptography and a distributed ledger technology can be combined into an open digital currency application. It solved double-spending [74] and permissionless distributed system security and access problems [75], in which a set of non-trusting writers share a database with no trusted middleman [76]. The use of a generalized protocol is essential for general cooperation so that the participants in the system reach a consensus and have a unified global view of the world state.

Interactions happen through the use of asymmetric cryptography, private and public keys [77]. The owner of the transaction proves themself by signing the message of their private key. Thus, the network can validate the truth of the transaction with the assistance of the provided public key in the transaction. The usage of asymmetric key pairs allows the user to ensure authentication, integrity, and non-repudiation into the network.

Nodes which have the transaction history of the entire system, thus containing all the blocks of the given network, are referred to as full nodes of the given network. They broadcast all the received messages through the network to avoid network state inconsistencies. During block creation, the given collected and validated transactions are included in the current block with the current timestamp, but it cannot be considered final until the majority of the network has accepted it and until consensus is reached. This is a process called mining or validation. The miner or validator node broadcasts the last created block back to the network. The naming of the process and nodes depends on the consensus mechanism type. Validation-type consensus mechanisms generally speed up the finalization of transactions, as conflicting chains can be penalized.

4.1. Smart Contracts

Initially, decentralized systems which were implemented were able to transfer the assets of the given system between users, which provided developers with a minimal opportunity for higher overheads and usage. Smart contracts allow the development and implementation of formal applications based on a decentralized system for more real-world problems and sectors as well, giving developers and users a completely free hand.

The basic concept can be found in Nick Szabo's article [78], in which he combines computer protocols with user interfaces to execute the terms of a contract. A smart contract is a computer program which has self-verifying, self-executing, tamper-resistant properties. It allows bytes code to be executed without any third parties on the blockchain [79]. Automatically executing smart contracts provide cost-effective, transparent, and secure interactions [80].

Every smart contract has an actual valid state [81] which is approved by the nodes of the network. The states of the smart contracts create the world state of the blockchain-based network. The input of a smart contract is a transaction, the effect of which causes the execution of the implemented logic in the contract to change the smart contract's state. A given smart contract's execution can trigger different types of state changes and new triggers in other contracts in the network.

The best-known smart contract-supported blockchain ecosystem is Ethereum [82], based on the EVM (Ethereum virtual machine). With this virtual machine, users can define smart contracts which are written in Turing-complete language and can run them in a decentralized network. Solidity is the most common high-level programming language

which is translated into a stack-based bytecode language so that the EVM can interpret it. Compiled bytecodes are stored on the blockchain, but various workflows are available for auditing and trust [83], providing assurance of the implemented logic in the given contract. Considering the structure of the Ethereum layers, smart contracts are located under the application layer, in the contract layer, providing and ensuring decentralized application functionality.

Regarding Ethereum and smart contracts, it is important to mention that a different accounting method is used to keep track of user states, compared to Bitcoin. The main difference is how the state of the system is recorded in the chain. In the case of Bitcoin, the UTXO (unspent transaction output) accounting method is used. There are no accounts or wallets. In essence, the transactions between addresses are stored in a directed acyclic graph.

In the case of Ethereum, the account-based model is used as an accounting method [82]. This implementation choice made smart contract support more effusive since smart contract support is much more difficult with a UTXO accounting method. An account-based model maintains a database of network states. They are referred to as accounts states. A user's private key or the corresponding smart contract can control the corresponding account. However, the most important part is transaction initiation, as the origin of the whole interaction process is only a user since a smart contract cannot initiate a process by itself.

The mentioned double-spending problem is solved by the account-based model since individual coins cannot be tracked. A replay attack occurs when the receiver sends the same valid transaction to the Ethereum network repeatedly, ensuring the possibility of effectively draining the origin sender's account.

The Nonce account field traces the number of transactions sent from the given account and prevents the possibility of such fraudulent transactions, thus preventing replay attacks. Parallelization is more difficult to implement with an account-based account method, but there are various modern solutions for this drawback [84].

4.2. Trilemma

Due to the decentralized characteristic of blockchain, the CAP (Consistency, Availability, Partition) theorem [85] used in the world of traditional distributed systems is not suitable for the examination and categorization of blockchain systems. On the other hand, in blockchain systems, The scalability trilemma (blockchain trilemma) proposed by Vitalik Buterin (founder of Ethereum) can be used for similarities, dividends, and characteristic measurement. Trilemma points out that three important properties of a blockchain system, involving decentralization, security, and scalability, cannot perfectly co-exist, as shown in Figure 2. Every project has to deal with this trade-off in order to bring the most favorable architectural decisions to reach the most optimal solutions [86]. As presented in the paper [87], there is no appropriate consensus that can achieve all three simultaneously. In relation to the consensus mechanism, these corner points can be more practically referred to as fault tolerance, resource efficiency, and full transferability. It is important that these corner points are not decisive, binary questions. Different trade-offs can and should be applied in consideration of the given problem [88].

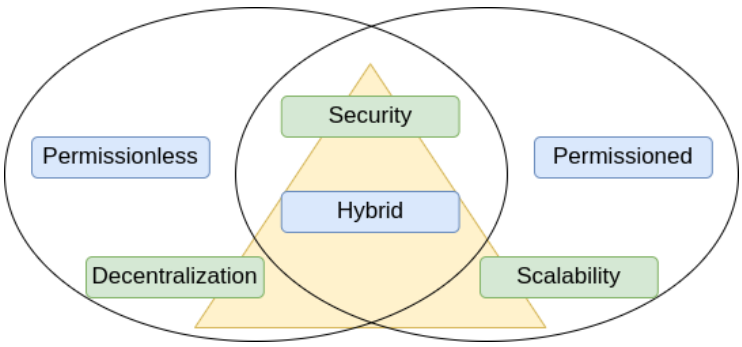

Figure 2. Blockchain trilemma with the permissioned and permissionless categorizations.

4.2.1. Decentralization

In a more decentralized system, there is no centralized entity that has absolute power over the system and would independently make important decisions. A decentralized network is controlled, governed, and directed by the community, the users. Decisions are voted upon by users of the given network in a community-based manner.

Low trust and low cost can make a network more decentralized. Low trust involves entities in the network being able to verify all rules separately, as well as independently computing the latest state. Low cost means that the cost of joining the network and the threshold criteria of the network are minimal. These attributes define the degree of decentralization.

4.2.2. Security

Security examination is possible from two points of view, from that of internal and of external attacks. During an internal attack, the attack is aimed at the consistency of a network and the strength of its immutability. Typical threats against a blockchain network are, for example, distributed denial of service attacks, Sybil attacks, collusion attacks, and penny spend attacks.

4.2.3. Scalability

Scalability defines the capacity and performance of a given network. This indicates how many users the given network can serve and how wide it can grow. The two most talked-about scalability metrics of blockchain scalability are transaction throughput and transaction confirmation latency. According to the current networks, the appropriate number of transactions per second for real worldwide adoption has not been reached [89]. However, in the following sections, scaling approaches will be discussed, the use of which can provide a completely acceptable and appropriate user experience. In addition, the blockchain trilemma is approached in the most appropriate way. This results in decentralized scalability achieving higher performance without noticeably increasing the network's trust assumptions.

4.3. Permissioned and Permissionless

In relation to the blockchain trilemma, it is important to mention the permissioned and permissionless categorizations, which differ from one another from the point of view of decentralization, and thus also conceptually and in adaptation [90]. An entity can freely join the permissionless network and has full rights, and can take part in network decision-making without permission. In the case of permissioned networks, the system is operated by centralized authorities, usually corporations, who maintain the whole network and determine the decisions about the network. Therefore, all nodes have some role and permission within the network, which allows them to take part in the network. A permissioned approach is an increasingly widespread technology in the industry, where privacy, as well as security and proprietary, are important, [91] due to the higher level of centralization. From another point of view, a permissioned network is complemented by an additional security layer, which is responsible for access control and determines which entity has the right to execute a given operation. This adjunct layer is managed by a centralized authority [92].

By reducing decentralization and requiring permission from the entities to become involved in the system, the network immediately achieves security improvements [93,94]. In addition to improving security, better results can be achieved in terms of scalability and performance [95,96].

It should be noted that the two approaches have different limitations and goals, along with the fact that they can be used to solve different problems [97]. It is important to mention the hybrid variant [98], which is a relatively new form. It combines the characteristics of permissioned and permissionless forms in one network, using them for different

functionalities. In general, the centralized permissioned network uses the decentralized permissioned network for communication or verification in the hybrid approach [99,100].

4.4. Monolithic and Modular

The most difficult component to satisfy of the trilemma is scalability. In order to understand the scaling solutions discussed later and the structure of the proposed network architecture, it is necessary to understand the difference between monolithic and modular architectures. The monolithic blockchain architecture approach can be divided into three components:

- Data availability (and consensus);
- Execution;
- Settlement.

In the monolithic approach, all three components are handled and integrated into only one network. Transaction sharing, ordering, and execution are operated in one network. This implies strong limitations and makes only horizontal scaling possible [101], which changes the functionality and protocol of the original network.

On the other hand, the approach of modular architecture makes it possible to create a stack structure with several layers. Moreover, it can provide the monolithic structure's degree of security and guarantee decentralization. By definition in the modular approach, at least one mentioned component is outsourced into an independent network. This kind of architectural approach implies great flexibility in terms of components. A modular blockchain is a network that takes on the task of only one given component. The division of the components into different independent layers allows for a more optimal solution from the point of view of security and scalability. Similar to the monolithic approach, the network managing the given component is independent and sovereign. In addition, in the network, which implements the functionality of a given component from the stack, due to its modular characteristics, its functionalities are independently implemented and the critical decision-making does not depend on underlying and overlying components.

The modular approach which divides the network into several components provides solutions for monolithic constraints and improves the monolithic approaches without drawbacks. Common constraints during the monolithic approach are:

- Inefficient transaction verification: all nodes which take part in the consensus mechanism must re-execute every transaction to verify the validation of the transactions, independently from each other;
- Enormous resource requirements: in a given monolithic blockchain network, the threshold criteria for joining the network is too high and too expensive, from the point of resources. It affects the degree of decentralization as has been mentioned earlier;
- Scalability: the completion of all tasks of the components in a given blockchain network limits transaction throughput and transaction confirmation latency. It can only be improved to the detriment of security or decentralization.

The publication of ordered transactions after consensus is the task and responsibility of data availability. This set of transactions defines the current status of the given chain. Without data availability, it is not possible to define the chain state. The responsibility of the data availability component is to ensure the validity of the data in a tamper-proof manner. One such solution to the data availability problem is an implementation of proof of availability [102], which is an approach of modular architecture. The blocks are compressed into significantly smaller proofs. This technique reduces network communication costs and data storage overhead.

Ordered transactions by consensus, which are accessible with the data availability component, are executed in the execute component. The result of ordered transaction processing is a new state. The new state is created from the last state by the state transition. The new state will be used by the settlement component.

The settlement component supports the preservation of the new state from the state transition after execution and helps ensure its validity. It ensures interoperability between states. In terms of security, it is very sensitive, which is why a strongly decentralized and secure monolithic chain is used for the implementation of this component.

The use of a monolithic blockchain is not optimal for real-world problems, in either a permissionless or a permissioned manner [103]. It is not possible to achieve the appropriate transaction throughput and transaction confirmation latency in a monolithic blockchain. It is impossible to maintain a network that is responsible for the tasks of every component, and also satisfies the three main points of the blockchain trilemma, which meet the privacy requirements [104].

4.5. GDPR

GDPR [105] was enforced in May 2018 in all EU countries regarding how commercial and public organizations process personal data. It was a major upgrade of data privacy policies in the EU, as the last corresponding data privacy regulations were published in 1995. In the time that has passed since, many new technologies and platforms have been developed in the world [106]. It was impossible to delay a major data privacy update any longer. The reforms that GDPR enacted were concerned more with the use of technologies, rather than the technologies themselves. Thus, in the case of blockchain, GDPR-compliant blockchain technology is a misnomer, what exists is GDPR-compliant applications and cases of given blockchain technologies [107].

GDPR defines six major data processing principles and three different roles. The EU Blockchain Observatory and Forum's report [108] sums up the connection between blockchain and GDPR. In short, two principles emerge which are important from this paper's point of view, regarding the use of blockchain technology. The first requirement is the identifiable controller which stores and controls the data. This role is enforced by GDPR and it is impossible to define in a decentralized system. The second requirement is the user's right to modify or erase its data. As has been mentioned earlier, blockchain stores the data in an immutable manner. It is also impossible to erase or modify on-chain data. On the other hand, the report mentions solutions for solving on-chain data storage. Personal data which have to complete these requirements can be stored off-chain and the corresponding hash is stored on-chain. It is still unclear whether the hash is personal data or not. However, as the requirements of GDPR are highly abstract, there is the possibility for organizations to interpret them on their own. As the EU report [108] and some published research papers [109,110] do not consider the hash as personal data, neither does this paper. Besides these, the public characteristic of blockchain technology is also problematic, as anyone can have limitless access to any on-chain personal data without any information regarding the access.

5. Improvements

The real-world application of blockchain technology with privacy requirements does not work without any scaling improvements. Blockchains in the traditional sense, such as Bitcoin and Ethereum [111], which are based on a monolithic architecture, have various scalability issues as a result of this. Fortunately, various scaling approaches are now being researched and tested, which can increase the transaction speed and transaction throughput of the network without sacrificing decentralization or security. Thus, with the help of the increased network capacity, there is the possibility to serve the huge number of users in a cost-efficient way through applications in the blockchain network [112,113]. There are many different possibilities and solutions to scaling. The main focus in this section will be on scaling methods related to the Ethereum blockchain ecosystem which will be presented. There are significant trade-offs between security, decentralization, and scalability [114–116] as has been mentioned in the blockchain trilemma problem. In addition, it is important to mention that scalability is a problem not only in public permissionless blockchain technology but also in the permissioned type as well, another heavily researched area [117,118].

There is no universally accepted scaling solution for all problems. The characteristics of the given problem determine the most suitable scaling solution by the optimal fulfillment of the requirements. Scaling solutions can be classified into two different categories:

- On-chain;
- Off-chain.

On-chain scaling solutions change the existing Ethereum protocol, basically implying hard forks. On the other hand, off-chain solutions are implemented separately from the Ethereum Mainnet. They do not require any changes in the existing Ethereum's protocol. Under Layer-1, the Ethereum Mainnet is to be understood. Additionally, Layer-2 refers to the networks that rely on the Layer-1 network for security and consensus.

The most well-known on-chain solution is sharding, which horizontally divides the database into smaller parts, thus ensuring greater transaction throughput [119]. The main point of the sharding solution is the creation and maintenance of new chains separate from one another. Sharding shares the network load between these new chains and it is not necessary to process every transaction for every node [120]. On-chain solutions are not discussed in detail in this paper, since the goal of the blockchain-based higher education system is not the creation of a new Layer-1 blockchain network. Preferably, this will be achieved using the functionalities of an existing network with an optimal scaling solution which will exploit the decentralization and security of the existing Layer-1 blockchain network by building a modular blockchain stack for higher education systems.

As has been previously mentioned, off-chain scaling solutions do not require modifications in the existing Ethereum Mainnet. They are implemented independently of Layer-1. Off-chain scaling solutions simply communicate with Layer-1. The communication and interoperability with the Ethereum Mainnet usually happen through different bridge approaches. The main goal of these solutions is to somehow outsource the processing and executing of transactions outside of the main chain, thereby reducing transaction costs and improving transaction throughput. The later used scaling solution is based on the official documentation of the Ethereum blockchain and the following research papers: [121–123].

5.1. Rollups

The goal of scaling solutions and thus rollups is to achieve better performance in a cost-efficient manner, without compromising on security or decentralization. In short, the goal of the rollups is to provide the same security and decentralization guarantees as the Ethereum Mainnet. Due to the traditional monolithic structure used, the incorrect use of the terms Layer-1 and Layer-2 is common in scaling. The most important condition of the Layer-2 solution is that its security is derived from the Ethereum Mainnet. In a modular context, these solutions include those which execute off-chain, but the settlement and data availability are based on the Ethereum Mainnet. The main driving force is the collection of several transactions into a batch, which will be shared in a single transaction on Layer-1, reducing gas fees. Thus, the cost of a shared single transaction on Layer-1 is divided among the transactions in the batch.

In addition, the states are stored on-chain, tracking state transitions from executed and shared off-chain transactions. There are two different proof mechanisms for the tracking and verifying of state transitions: fraud proof and validity proof. Fraud proof optimistically assumes that the shared transactions and states are valid and only performs an on-chain check if it is challenged. As long as the validity proof is generated off-chain by using a significant amount of resources, the final proof of the transactions and state transitions are tamper-proof.

These solutions mostly consist of an operator, validator, sequencer, and block procedure roles. Different terms are used in given specific solutions as the functionalities of roles are more specific. The requirements of these node roles are specialized, such as high computation capacity. For this reason, the structure of network maintenance is often centralized, due to the nodes' high need, but they are increasingly striving for decentralization [124]. In the case of sidechains, plasma chains, state channels, and validiums, assets are transferred

across a bridge to be used on another chain. Therefore, these chains run in parallel with the Ethereum Mainnet and occasionally interact with it. Consequently, these solutions do not derive their security or data availability from the Ethereum Mainnet.

Rollups concentrate on the execution component in an off-chain manner, while the Ethereum Mainnet is responsible for decentralization, security, and data availability. This improves scalability since rollups completely eliminate on-chain with off-chain transaction processing, which results in the Ethereum Mainnet having to process fewer transactions, thus being less congested. In addition to the fact that rollup users have the possibility of cheaper transactions, they benefit from and receive the same security and decentralization guarantees as in the Ethereum Mainnet.

5.2. ZK Proof

Before presenting the two remaining scalability solutions, it is important to briefly introduce a cryptographic technique which is called zero-knowledge proof, or in short ZK proof. The proof mechanism of both of these is based on this technique. The main point of the ZK proof method is that one entity, called a prover, can prove its statement is true to another entity (called a verifier) without revealing any further information aside from the shared true statement. The main challenge in the ZK proof is proving the knowledge of certain information without revealing any extra information. This makes it possible to prove secret information, without actually revealing the given secret information. The mentioned two main roles during the method are prover and verifier, whose names are used later for roles.

ZK proof has many different types of algorithms, for example, zk-SNARKs (zero-knowledge succinct non-interactive arguments of knowledge) [125] and zk-STARKs (zero-knowledge scalable transparent argument of knowledge) [126]. They can be categorized based on whether there is a challenge–response interaction or not. Therefore, the two types are interactive and non-interactive [127].

During zk-SNARKs, CRS (Common Reference String) provides public parameters which can be used for proving and verifying. Therefore, the given system security is based on the CRS setup, as the corresponding information regarding the creation of public parameters can be used for the generation of invalid validity proofs which are correct. Some techniques heavily attempt to solve this problem by using a multi-party computation ceremony. If there is an honest party in the ceremony, invalid validity proof generation is not possible. The necessity of the trusted setup is a trade-off, with respect to the generated proofs being verifiable rapidly and cheaply.

zk-STARKs are more transparent and scalable than zk-SNARKs. They can work without a trusted setup as they use publicly verifiable randomness to set up parameters for generating and verifying proofs. Moreover, the computation complexity is almost linearly related to proving and verifying. Therefore, zk-STARKs are more optimal for large datasets. On the other hand, the verification and the storage of the proof are costly operations in an on-chain manner.

Outsourced verifiable computation as off-chain computation can be verifiable through the use of ZK proof. With valid outputs, any third parties are capable of proving that they executed their work correctly. ZK proof in connection with blockchain technology is commonly used as a Layer-2 scaling solution, for private Layer-1 support, decentralized storage, or blockchain compression [128,129].

For proving a computation, the given code has to be compiled to a ZK-friendly format. However, some operations are not ZK-friendly (SHA and Keccak, due to the bitwise operator). The proof generation time is connected with the number of expensive and complex operations used. Moreover, it is important to mention that the proof generation requires a huge amount of resources. This kind of hardware is usually specialized and optimized for the generation of ZK proof.

5.3. zkRollups

zkRollups is very similar to optimistic rollups. Both of them move computation and state storage off-chain. Thus, the number of transactions on Layer-1 is drastically reduced. The main difference is the proof mechanism which is used by zkRollups. Instead of fraud proof, validity proof is produced for proving the correctness of off-chain executions of transactions and state transitions using ZK proof. zkRollups architecture is divided into two components:

- On-chain contracts: one of them includes the blocks, deposited funds and states. The other deals with the correction of ZK proof. It is the verifier in the sense of ZK proof naming;
- Off-chain virtual machine: off-chain transaction execution and state storage are complemented by this component.

The power of zkRollups is based on data availability, transaction finality, and censorship resistance. All off-chain executed transaction data are shared on the Ethereum Mainnet in the calldata; thus, any users can recreate the current state to prove the necessary proof to withdraw their deposited funds. No supernode with malicious behavior can steal the users' funds. Using validity proof, there is no necessity for a dispute period. The shared new state, which is finalized on the smart contract, is immediately valid without any concerns.

In zkRollups, as in ZK proof, there are provers and verifiers. A prover is usually called a sequencer as it collects and executes the transactions off-chain. A prover is usually a centralized entity as proof generation has enormous resource requirements and using validity proof it is not possible to take advantage of the centralized structure.

Validity proof is submitted by the prover, which creates the new state root which already includes the state changes by the current transactions, the current block. The prover must share the batch root for shared transactions and the new state root as well. If these submitted data are correct, the verifier smart contract accepts the new state commitment. Before the proof generation, the prover has to verify the same things as during Ethereum Consensus, for example, whether a transaction has the correct sign or whether the corresponding nonce is correct. After a given number of transactions, the transactions are aggregated into a batch which is compiled for the proving circuit to compile into a succinct ZK proof. The proving circuit computes the validity proof for each transaction one after the other. After the last computing, the last state root, corresponding to the last validity proof, is the new state root which is shared with the verifier smart contract. After the proof generation, the verifier smart contract can decide whether the state's transition from the pre-state root to the post-state root is valid by using the validity proof and batch root.

Users can enter the given zkRollup by depositing funds into the smart contract. These deposit interactions are queued and are waiting for the sequencer, as a sequencer can submit the deposit interaction in the corresponding zkRollup. After that, the user can use its fund in zkRollup. During a withdrawal request, the user sends its assets to a specific address. This implies the exit intention of the user. If the operator includes this transaction in a batch, the user can submit the withdrawal request in the on-chain smart contract. The smart contract verifies that the burning transaction has happened and executes the asset transfer to the specific requested address.

Despite the fact that EVM compatibility in circuits is more difficult and resource-intensive than the usual simple token transfer circuits, in the past two years great progress has been achieved in the ability to implement EVM compatibility in zkRollups. The development of this opportunity is still in the early phase and many competitors are attempting to make the best EVM-compatible zkRollup. Therefore, there are two different types of zkEVM approaches:

- Building ZK circuits for native EVM opcodes;
- Creating new languages for ZK proof computation.

The first one requires much time-intensive work to implement all EVM instruction sets in an arithmetic circuit. However, this implies full compatibility and support for existing

projects and tools from the Ethereum ecosystem. The second one takes advantage of the creation of a new programming language, which is more compatible with validity proofs. The main drawback is breaking the compatibility with the Ethereum infrastructure and resources. zkEVM will be discussed later in the architecture section.

zkRollups can guarantee immediate withdrawal from Layer-2 without any delays. Thus, users have better liquidity opportunities. The usage of validity proofs creates the opportunity for a secure system without using any incentivized models for security; thus, liveness assumption is not a problem in this kind of system. Moreover, one trusted party for tracking transactions to protect funds is not mandatory. On-chain transaction data storing solves the data availability problem. Trustless cryptographic mechanisms ensure the correctness of off-chain transactions. Besides these positives, the enormous resource requirements reduce the degree of decentralization and ensure the possibility of centralized operators influencing the system.

5.4. Validiums

As it has been mentioned earlier, this scaling solution is also based on a validity proof mechanism as zkRollups, using ZK proof. The main difference is the storing of the transaction data. Validiums do not store transaction data on the Ethereum Mainnet. Thus, it implies the data availability problem, because the transaction data are stored independently from the Ethereum Mainnet. Therefor,e users' funds can be frozen in the smart contract as they cannot recreate the Merkle proofs for withdrawing their funds. On the other hand, this implies faster withdrawals for users as they can withdraw their funds by only using the correct corresponding Merkle proofs.

Validiums are based on off-chain transaction execution by using ZK proof. On the Ethereum Mainnet, the verifier contract verifies the given validity proof as during zkRollups, but in the validiums case, there is another proof for data availability for verifying the existence of off-chain transaction data. There is another contract, called the main contract, which deals with the funds and state commitments.

Without using the Ethereum Mainnet for data availability, validiums still benefit the Ethereum Mainnet from settlement and security points of view. Settlement guarantees for validium users mean the accepted states cannot be reverted or edited if they are submitted once on-chain. Besides that, the verification of validity proofs is implemented in the immutable on-chain smart contract, which can reject the state transition if it is invalid.

Validiums work very similarly to zkRollups. They collect transactions into a batch which will be sent to a proving circuit. Its output is the validity proof which ensures that all the performed operations which were included in the batch are correct and valid. Besides the validity proof, there is the state root which consists of the accounts' state in a Merkle Tree. With the calculated new state root and the corresponding validity proof, a state update can be made if the verifier smart contract verifies the submitted data. The main contract updates the current state root to the submitted one.

During a user deposit, the operator has to include the corresponding transaction into a batch. Without this event, the user cannot transfer their funds in the given validium. User exit is similar to that in zkRollups. The user has to submit their intention and the operator will include this transaction into a batch. Then, the user can withdraw their fund from the on-chain smart contract. There is another option for leaving the given validium. It is an emergency exit against a centralized censorship situation. Like in other rollups, by providing Merkle proof the user can withdraw its fund independently from the operators.

The most cumbersome drawback of the validiums is data availability, as transaction data are not published on the Ethereum Mainnet, on account of validiums storing all transaction data off-chain. Without transaction data, the current state is not recreatable for any users. Thus, users are not capable of withdrawing their funds on their own. There are different solutions and approaches for the data availability problem. Most of them try to complete the system with centralized entities that take responsibility for data availability.

This results in a trusted third party, referred to as a data availability manager, having to share all transaction data by operators. Operators have to communicate with that party because this party's key is mandatory for interactions with validium's smart contracts.

The EVM compatibility of validiums is the same as in zkRollups, but in this case, there are no memory storage limitations. Validiums are not limited by Ethereum's data processing capacity, as validiums store little data on-chain. This results in validium being a purely off-chain scaling solution. With the use of validity proofs, validiums can provide higher security guarantees than other purely off-chain scaling solutions, such as plasma chains or sidechains. Almost the same positives and negatives as in zkRollups are repeated in the case of validiums. However, validiums can provide close to the highest throughput by expanding the Ethereum ecosystem by moving the execution and storage of the transactions to off-chain.

6. Design Concept

During the design stage, it was important that the requirements defined earlier were fulfilled by the proposed higher education system. These requirements include system-level ones, such as higher security, decentralization, scalability, and the blockchain trilemma, as well as functional ones, which were discussed in the related works section.

The base of the system design is a single organization's, HEI's, own superintendent and connections with other HEIs. Since an HEI is generally responsible for its own system and the data and operations included in it [130], it would be problematic to build a system based on the usual public characteristics of blockchain technology. This results in a significant difference to the current systems, and the given HEI would be at a disadvantage against other HEIs if all its secret internal data would be publicly accessible for anyone. Furthermore, the completely public string of all data is not GDPR-compatible. Due to the avoidance of publicity, the encrypted storage of data between several HEIs is problematic from several points of view. Privacy always has an additional cost, which can possibly affect scalability.

Moreover, the maintenance of a network from the HEIs' point of view in which they are forced to run and validate the operations of other HEIs and store their data implies an enormous additional cost. As with the joining of each new HEI, the expenses of a given HEI would double compared to the original painting cost. Considering a constantly growing and expanding network, this would result in runaway and unaffordable costs. Furthermore, the secrecy of smart contracts, which are used to create easier and simpler interoperability referred to as private smart contracts, in a permissionless manner are not evident. It is almost technically impossible to manage and verify the private smart contracts' states with fully private program codes, inputs, outputs, and computations as well. There is an entire research field dedicated to solving this problem using homomorphic encryption [131,132].

The use of private smart contracts in a permissioned manner is simpler, and more solutions are possible and available [91,133], but they involve additional operations and communications. In addition, these approaches are designed and optimized for organizations and do not approach the problem from the perspective of the users.

From the mentioned problems, it follows that each HEI must maintain its own education system independently of others, as usually happens now. With this, private data usage can be implemented with the appropriate scalability, in addition to the fact that the system is GDPR-compatible. However, the mentioned decentralized scalability is missing, since one entity controls the entire system, if the mentioned monolithic architecture approach is considered. Using the modular architecture approach, the education system maintaining a given HEI will be integrated into this stack, thereby guaranteeing a greater degree of decentralization of the system.

The given scaling solution plays an important role in ensuring that the data are not publicly stored and that the maintenance of the system and the execution of transactions are the tasks of only one single entity. This means that all data are stored off-chain. Moreover, in order to guarantee the system's security and correct operation with these requirements,

it is not possible to build the security on an incentive model. The proof mechanism of the selected scaling solution cannot have the trust assumption of the existence of an honest party, as this is too much to expect from simple users who do not want to take part in the system maintenance. This is not a good direction, to make users responsible for the working of a valid system. A cryptographic protocol is necessary for securing the network's validity. Therefore, the valid proof mechanism using ZK proof is the most optimal for fulfilling the requirements. Furthermore, it solves the liveness assumption problem. With the current state of research in zkEVM, there are many possibilities for using smart contracts with validity proof.

On the other hand, one of the main points of using a scaling solution is to make an education system which is capable of transferring money without any trusted third party. For this reason, it is necessary to handle the funds in the most secure way during their entire life cycle, and both during entry and exit. Furthermore, this functionality ensures that users can take part in the costs of the system by paying their transaction fees or by paying their service fees.

The selected scaling solution that fulfills the mentioned requirements is categorized as validiums. In terms of the modular structure, in this design, the settlement layer is a decentralized permissionless blockchain network, the Ethereum Mainnet, while the execution and data availability components are implemented in a permissioned network maintained by the HEI. Further on in this chapter, the individual components of the modular stack will be explained in detail, as they together form a decentralized verifiable higher education system.

6.1. Settlement Component

Among the three components, this is the most important from a security point of view, since the finalization of the blocks takes place in this component. From the current possibilities, from the point of view of security and decentralization, the most suitable and appropriate alternative is Bitcoin or Ethereum. Since the other possible solutions are optimized for scalability, security or decentralization are compromised. It is an unacceptable trade-off from a settlement component point of view. Among other things, the stability of decentralization [134,135] and smart contract support are important and significant during the choosing of the underlying network. Therefore, it is for this reason that Ethereum is the most suitable as the settlement component in this system. Furthermore, the wide token distribution and developer community continue to improve the benefits of Ethereum in the long term. In addition, it is important that the Solidity programming language for smart contracts be the same as the programming language used in the execution component. Only the settlement functionality of the Ethereum Mainnet is used in this architecture.

The settlement component can finalize the current block for the execution component. With the use of validity proof, the blocks that are finalized cannot be reverted due to the immutable manner of the blockchain and proof mechanism. The shared block can be considered as final, ensuring that the given state transition cannot be rolled back. This is the final settlement. The settlement component is incorruptible, always available, and resilient.

The settlement component provides periodical communication with the costly Ethereum Mainnet, thereby reducing transaction costs. During the production of a block, corresponding data which represent the current block effects are stored. In this case, it is the current state root which includes all corresponding information of the users and smart contracts at a given moment. It is submitted by the operator, who generates the corresponding validity proof. Its further task is to validate that the required number of signatures are included in the message and verify the correctness of validity proof. All in all, during an interaction, the submitted data are the signatures of the committee members, the current state root, and the corresponding validity proof. In the settlement component, the Ethereum Mainnet, this particular transaction, which is a given interaction for producing a block, will be executed and included in a block. This finalizes the current producing block and makes it easily verifiable.

The necessary bridge and other smart contracts serving the higher education system's correctness are deployed on the Ethereum Mainnet. In the case of the designed system, the smart contracts that are deployed on the Ethereum Mainnet can be divided into two categories based on their functionalities. One category is application smart contracts, which include the verifier and state contracts, and are used by the execution component. Another category is the committee smart contracts, which are supported by the data availability component. It consists of only one smart contract. The state contract implements the bridge functionality, which ensures interoperability between the two networks. The state contract stores the current state root, which provides the possibility of the verification of certifications or other defined features. The verification of the validity proofs provides the correctness of each submitted state transition by the verifier smart contract. This is achieved by using zk-SNARK circuits in this architecture. The verifier smart contract is created from the corresponding circuits to verify the submitted validity proofs. It is possible to use precompiled contracts to verify the validity proof which allows for the creation of the necessary algebraic circuits.

6.2. Execution Component

The heart of the designed system is the execution component as it is the most characterful and influential part of the stack. It usually includes some kind of consensus mechanism, but in this case, it is not necessary for off-chain transaction ordering and execution as only one organization is responsible for these tasks. This is possible due to the organization's interest being the correctness of the system, and in fact, malicious behaviors are prevented with the usage of validity proof mechanism. In this case, the execution component utilizes the mentioned settlement layer to produce new blocks. It does this by submitting the information and validity proof belonging to the current block with the mentioned state contract.

This type of higher education system approach can only be achieved thanks to the tremendous developments in this area in recent years by an enormous number of researchers. It enables zkEVM circuit implementation and is used with blockchain technology. This is possible due to several reasons. Customized optimization and a more flexible backend can be achieved by using polynomial commitment. Hardware acceleration makes proofing more efficient. Another important factor is the use of lookup tables and customized gadgets, which significantly reduce the overhead of the EVM circuit. A succinct proof is therefore possible to prove that the states are updated correctly after applying the transactions.

In order to understand the working of zkEVM, it is necessary to have a brief understanding of EVM. The EVM is a state machine that performs state transitions based on the power of some input, from the previous state to the new, current one. All smart contracts are stored in bytecode on the blockchain. These are compiled from their source codes. During a smart contract interaction, these bytecodes are loaded first. EVM opcodes, which are contained in bytecode, ensure the possibility of the interaction, and the execution of the operation implemented in the given smart contract. In summary, reading the bytecodes from the EVM's storage, executing the given computation, and then writing the result back to the EVM's storage are the requirements for zkEVM. It follows from this that the proof of zkEVM must prove that the bytecodes were correctly loaded, executed in the correct order, and finally written out well.

As already mentioned, there are two approaches to implementing a zkEVM. In the proposed system, a hybrid solution between the two is used, since there are more differences from the point of view of data structures compared to in the Ethereum Mainnet. These differences are in the block structure and state tree. These changes ensure the possibility of easier and faster proof generation, in addition to the fact that the control of minor modifications is also almost fully compatible with existing applications and tools. The replacement of the Keccak hash function required for the use of Merkle proofs with a different hash function is necessary because Keccak is not ZK-compatible. However, on the other hand,

given the current trends, the development of Ethereum is heading in the direction to replace the use of Merkle trees). With these changes, a faster prover time can be achieved than with the fully compatible approach without minor modifications, but it still takes a long time to generate proofs.

The Polygon zkEVM is used by the Polygon team in the designed system, implemented opcodes for the ZK circuits, with which the mentioned full Ethereum compatibility is achieved in almost all existing smart contracts, developer tools, and wallets. At the time of writing, Polygon zkEVM is being open-sourced. Therefore, it will be able to be used freely in this system. In addition, the use of recursive STARKs ensures higher performance in proof generation compared to other solutions with a similar approach. Compared to other projects, better results can be achieved in terms of proof generation time and scalability.

By default, zkEVM has many major components, but in this architecture, only the necessary components are used for implementing the execution component of a higher education system. zkNode and zkProver components are used beside a very similar bridge implementation. The operator has to address all corresponding roles to the execution component. Therefore, it is functional as a sequencer, synchronizer, and prover. It has to order the transaction in batches for a later proof generation. Furthermore, it has to synchronize the state between the deployed bridge on the Ethereum Mainnet and validium's state. One of the most important functionalities of the designed protocol is the possibility to request the corresponding proof to the state root of the action or document.

All corresponding rules of the implementation of EVM are implemented in the zkProver component. This component generates the validity proofs for the batches of transactions. It is the prover role of the operator. zkProver has quite a complex architecture with two main state machines. In a nutshell, the zkEVM expresses state changes in a polynomial form. Thus, all the valid transactions must satisfy the defined polynomial constraints. The proof generation consists of two parts, which are STARK and SNARK proof builders. Without the trusted setup requirement of the STARK proof builder it is easier to prove the satisfaction of all the polynomial constraints. Later, the correctness of recursive proofs generated by STARK is proved by using the SNARK proof builder. Thus, on-chain cost is lesser when using SNARK proofs.

The deployed bridge in the settlement component during an interaction obtains the current Merkle state root which includes all users' balances and states of smart contracts. It provides possibilities of emergency exit by submitting the corresponding Merkle proof. The submitted Merkle root represents the state of the zkEVM in the chain and thus the verification of a given information or event is possible on the settlement component.

6.3. Data Availability Component

Data availability is an important part of scaling solutions to guarantee security for users. In some cases, such as rollups, this makes it possible to use the same security guarantees as the Ethereum Mainnet. In the modular stack superstructure of the validium scaling solution, data availability is implemented independently of the Ethereum Mainnet, in an off-chain manner. Storing transaction data separately from the main chain ensures higher transaction throughput and makes possibilities for higher gas consumption with complex smart contracts. However, there is no data availability guarantee. Thus, the security is reduced because the data provider can misbehave or go offline.

It is important to distinguish between data availability and data retrievability, as these are completely different concepts. In the case of data retrievability, we want to obtain older information about the blockchain's history. These data consist of older blocks and receipts related to the older events of the given blockchain, as long as the data availability is relevant information related to the current shared block in the consensus mechanism. Therefore, transaction data are used for recreating the actual state of a given blockchain.

As already mentioned in the settlement layer, a committee contract on the Ethereum Mainnet addresses data availability. This contract is implemented by the DAC (Data Availability Committee) [136] functionality, instead of trusting in only one single data

provider, but it is possible that this component is provided by only one entity. Thus, the DAC has only one member.

DAC members are nodes which have unique addresses on the network. It is important that more organizations take part in the committee, ensuring greater distribution, and thus less trust assumption. It is important to minimize trust in only one organization. By default, without DAC, there is the possibility of one point of failure because the transaction data are handled by a single operator. However, with the help of more assumed trusted organizations, the security of data availability is improved in an emergency situation. In addition to all this, with the involvement of trusted organizations, users' privacy can be maintained and data can be managed in a private manner. It is possible to democratically remove a member from the committee contract. The success of the voting depends on the defined security level of the network. This is usually conducted over a longer period of time because during this time the soundness of the system can be damaged.

The DAC members receive the message containing the distributed current state root from the operator, which will be submitted on the Ethereum Mainnet. As well as this, they receive the current transactions, so that they can validate whether the current state root is correct. If the new state root is correct, the message is signed by their public keys. During the operations, DAC members update their current states and store the relevant data. The committee contract receives this message, which includes the actual state root, validity proof, and signatures. The valid proof verification and state transition only take place if the appropriate number of DAC members have signed the message. During an emergency state, the state contract does not accept new state transitions, it is not possible to execute new transactions. Only users can withdraw their funds. For this, an honest DAC member is required as a minimum for providing Merkle proof for the last valid state.

The HEI can create a profitable opportunity for the involved trusted parties by using an incentive model. The HEI creates its own DAC. DAC members must stake a predetermined amount of assets that guarantee their honest behavior, which can be flushed due to the given DAC member's malicious behavior. It is important that these members can be organizations that operate in DAC under legal contracts. Thus, there will be legal consequences for their malicious behavior (for example, a data breach), besides their financial losses. In addition, the DAC members have to receive a part of the transaction fees during the block creation. Thus, a more secure process between users and DAC members is guaranteed.

7. Proof of Concept Application

The purpose of the PoC application is to prove the functionality of the defined concept in reality. Smart contracts are written in Solidity programming language. The corresponding unit tests are in python and run by the Brownie testing framework. During the functional testing, Remix IDE and MetaMask were used. The PoC project is available in this repository.

Since the requirement of the network system is extremely high (zkProver: 1TB RAM with 128-core CPU), the PoC was deployed on the "Polygon zkEVM Testnet", not locally. In this regard, it is important to note that the PoC does not include the data availability component in the aforementioned form. The polygon zkEVM node was used for connection to the network. The deployed smart contracts are available on the permissionless test network at the following addresses (viewer):

- Teacher: 0x173D71646e388774960cA33b94106a00b81760Ea;
- Student: 0x33cCeb2279767D585EA2Bb0C1B2b875bDCEFf9e2;
- Degree: 0x1A54F40445ee8B1F11cAf0D46C4EE5fC2c63FB2a;
- University: 0x3dC33b50574deedCBedb6D1654E657A8BCD2edCf;
- CourseCatalog: 0xAd1b862ad9C0F53023fe2590e74Ea5053EF27Dc3.

Furthermore, the addresses of the wallets used during interaction on the permissionless testnet during the test session:

- University wallet: 0x42174c41FAed24ff21b6b704b208919a2844D10F;
- Teacher wallet: 0xbE26D3F09339656dEbEC30F85774258Ed401b63a;

- Student wallet: 0x553C8E64992c5726657E4D0143D8Fb1b8f179da6.

The centre of the application is the university smart contract, as shown in Figure 3. The university smart contract is the owner of the other contracts. Thus, the system operates on the basis of predetermined and publicly available policies, in which there can be no invalid state transition. The implementation of some smart contracts is based on tokenization (EIP-721), because it makes the testing easier to use and already existing tools can be used, for example, the MetaMask wallet. During the public testing, a wallet with a student role was able to issue a degree to itself, after fulfilling the specified requirements. This ownership can be verifiable from the current state root on the permissionless network, without any third party. Furthermore, the smart contract on the permissionless network only accepts valid state transitions between state roots.

The main experience is that it was found that by using the sequencer and the prover between the entity and the permissionless network, a hierarchical trustful chain is created on the permissionless network. Therefore, during the verification of a degree, it is sure that the relevant events preceding the issuance of the degree (such as the admission of a student or a teacher's course details) are also correct and valid. A state machine is created on the permission network which manages the HEI's cycle. Its end state is the creation of a degree, as shown in Figure 4.

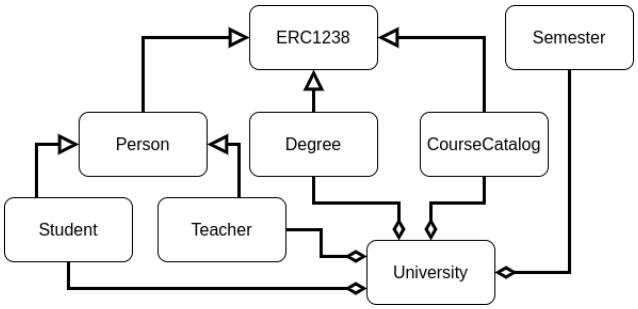

Figure 3. The class diagram of the PoC application.

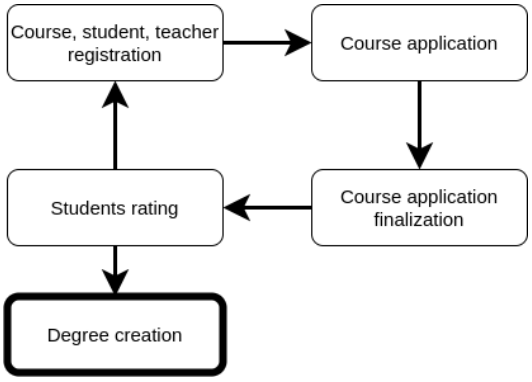

Figure 4. The states of the PoC application.

8. Analysis

8.1. Data Management

The most important structural change to the other blockchain-based higher education system is that the operator in the middle of the system maintains a single HEI, as Figure 5 shows. With the help of the validity proof mechanism, the system can still work in secure mode, despite the fact that it is operated by only one HEI. The permissionless distribution of data raises privacy concerns, and managing it with encryption involves enormous overhead in terms of computation and cost. In the permissioned manner, this approach cannot guarantee tamper-proof operation, since this type of structure is optimized for more organizations. In this case, the implementation of an HEI system in such an environment does not fit. Furthermore, the GDPR requires the determination of the involved parties

and the traction of personal data. Therefore, by exploiting smart contract compatibility, the implementation of a private IPFS with defined trusted parties in a permissioned manner is the most secure, as in DAC. This approach is similar to the one in the paper [137], with some changes that users have the opportunity to erase or modify their off-chain stored data from the database. Therefore, the execution component is supplemented with a data management component to implement the necessary functionalities.

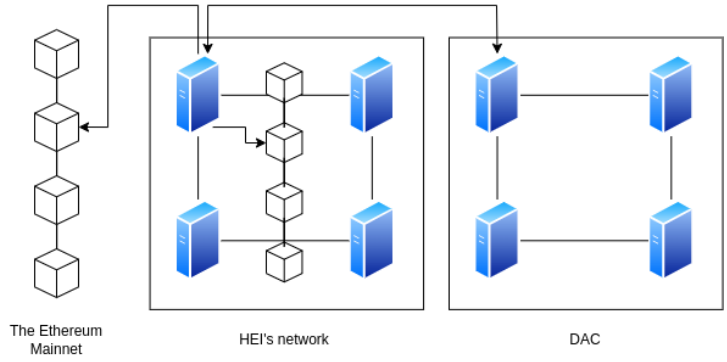

Figure 5. The connections in the system.

8.2. Credit Transfer/Interoperability

With the usage of smart contract compatibility, the two smart contract-based higher education systems using zkEVM have the opportunity for credit transfer via a predefined protocol. This can be achieved by using bridges. One HEI is part of the other HEI's system. Thus, they can supervise the credit transfers from their systems to the other ones. For this, it is necessary that the given HEI uses this recommended higher education system. The two HEIs can retroactively verify the events belonging to the given student, and if they have the correct access then the personal data can be verified as well. This results in faster and easier interoperability. Some parts of the current manual mechanism can be somewhat automated with predefined conditions. Without using the same system, the verification process is less time and work-consuming with the usage of the recommended system verification capabilities. Due to the higher interoperability, it is necessary to use some standard format for the documents (W3C-VC, W3C-VC-EDU). This helps to expand the technical dimension [138].

8.3. Admissions

In the application process, the applicant student will sign the application message with their public key which indicates their desire to take part in the application. With this, the application procedure starts. Then, the necessary personal data will be uploaded via the well-defined user interface to communicate with the operator. No privacy concerns arise with this mentioned data management method. The applicant can already verify the status of the application at this stage, because the interaction of the responsible person is also displayed on the applicant student's interface. The public key represents the given user in the system. Upon successful application, the given public key will receive further access to the system. Therefore, the given later student can be followed for the entire duration of their study, from their application to their graduation.

8.4. Student Assessments and Exams

Due to smart contract compatibility, it is possible to completely automate the entire process of the courses' lifecycle, from course registration to the corresponding final exam. The process of course registration can be implemented into smart contracts. Predetermined requirements, such as necessary prerequisite courses or a minimum grade average, can automatically decide who satisfies the requirements to register for a given course. Thus, the allocation of course places is transparent and ensures equal opportunities for all ap-

plicants. In the case of this proposed system, a given transaction, for example, a course registration, has minimal costs. In addition to the automated exam, it is also possible to submit the exam papers in a private and verifiable manner. This is important so that the implementation of the overture mechanism can be easily achieved within the system.

8.5. Certificates Issuing and Verification

The on-chain and off-chain tracking of the duration of a course makes it possible to obtain fully automatic certificate issuing. The students can obtain their degree if they have passed the necessary requirements. In this way, the type of misuse attacks can be minimized. The degree obtained by the student is a token in the system, for which the HEI creates a visual representation. The verification of given information, tokens, and visual certifications is always available through the settlement component by using the requested proof for the state root.

It is important to mention that this provides data retrieval and verification, but due to the centralized manner of data storing, there is a possibility that the retrieval function will be frozen due to malicious behaviour. Therefore, it is advisable to implement functionality for users to store the most important certifications and related information with the corresponding proof in decentralized storage. Thus, the given certification and information can always be verified in any circumstances.

8.6. Payments

Using a fully functional scaling solution such as the Ethereum Mainnet, it is possible to adapt any asset without any modifications. Thus, any agreement included in a smart contract can also use payment functionality. Therefore, it is possible that the system can deal with tuition fees, dividend fees, scholarships, or any other types of payments without any untrusted third parties. It is possible to adapt any token to the system, including stablecoins. In this way, the application of volatile cryptocurrencies to paid users can be avoided. With this, the higher education system has less trust assumption and less human administrative work in relation to this payment. It is important to note that when examining the current international regulatory trends [139], there is no single state in the developed countries that have recognized it as a legal tender, only a few as taxable assets. Furthermore, due to the novelty of the technology, there is a lot of uncertainty in taxation-related matters across countries [140].

8.7. Analysis of the Challenges

With the permissioned collection of transactions and data, no personal data are made public on the main chain. With this approach, the users have complete control over under what circumstances and how they want to store their data while maintaining the possibility of verification. The provision of completely decentralized data, on the other hand, lacks in this approach. However, it is important to note that with the current regulations, this would not be a suitable way to store personal data by an organization.

With outsourced data management and the off-chain approach, only the necessary data have immutability characteristics. The current state root and validity proof are the only part of the system which is fully immutability independent from the HEI. Additional data can be erased or modified from the system. With this approach, the system meets the requirements of the GDPR.

In smart contract-based business process execution, the verification of smart contracts is indispensable and mandatory for achieving trust and security. Program-level and contract-level verifications can bring confidence and reliability to the design and testing of the system [141].

There are lots of different types of smart contract vulnerabilities which can be detected by using tools, but these tools are not perfect. They are still under development [142]. They can help to minimize the risk of faults and bugs. The development and testing of these new contracts are in the very early stage, but a similar approach of the underlying system

as in the case of Arbitrium minimizes false assumptions. In the case of zkEVM, there is a new attack vector, because the validation of circuit correctness is pretty difficult. The entire system's correctness is based on the assumption of circuit correctness.

With this approach, the measure of blockchain usability is not reducing or increasing. It offers similar features as the Ethereum Mainnet. It is necessary to use a specific wallet, which differs from the implementation of higher education systems up to now. However, most of the user interface can be implemented similarly to a traditional higher education system user interface. Therefore, users do not need to be experts in blockchain technology to be able to use the system. A well-structured and implemented user interface can include all mentioned requirements and functionalities.

As it has been mentioned before, a higher education system which is built on a decentralized structure always has a cost overhead that is unlike the traditional one. In this designed system, there is an additional cost in terms of hardware resources and user transactions, as each interaction of the operator with the settlement component involves transaction fees.

In terms of hardware resources, the huge amount of hardware resources required for the generation of the validity proof is expensive and outstanding compared to traditional systems. The performance of a large number of expensive math operations for generating ZK proof can be improved by using specialized hardware such as FPGAs (Field Programmable Gate Arrays) and ASICs (Application Specific Integrated Circuits). However, many HEIs cannot afford such a high level of investment. However, distributed hardware resources between several HEIs can solve this problem. Compared to a traditional system, there is almost the same usage of storage capacity, if the operator does not deploy its own Ethereum full node for communicating with the settlement layer.

In terms of transaction cost, compared to the traditional system, this is a new concept within the higher education system. It depends on how much the given HEI wants to spend on the system's costs. The bridge deployed in the settlement component has different on-chain computation costs. Gas refers to the fee required to successfully execute a transaction or a contract on the Ethereum Mainnet.

Due to the use of DAC, the validation of signatures is necessary to ensure data availability, as without the appropriate number of signatures, the state transition in the state contract is not allowed. The verification of one signature costs 3000 gas. A DAC committee is usually made up of 7-10 members. Since there is a fixed number of members in the DAC of the designed system, the cost can be calculated using the predefined number of members for successful attestation. During the calculation the committee is made up of seven members, so the cost of verifying a DA attestation is roughly 21,000 gas. However, with the use of DAC, no transaction data are posted on-chain which provides cheaper cost and better privacy.

Another source of transaction cost is the state transition which is tracked by the state root and verified by the validity proof. Using zk-STARK proofs, the cost of the verification of the validity proof is around 350,000 gas in the Ethereum Mainnet. Moreover, the cost of updating the last state root with the new one is 20,000 gas, because the Merkle root is stored as a 32-byte word.

In summary, one interaction costs around 400,000 gas for the operator, the maintainer HEI. Calculating the average gas price, at the time of writing, the cost is around USD 25.00. An HEI must pay this amount of money for interacting with the settlement component, which makes the designed system more decentralized and verifiable. However, it is possible for the given HEI to force the users to pay their transaction fees, thus reducing its system's maintenance costs. In this case, the arising transaction fees require completely new interaction methods in the operation of the HEI, which cannot be easily achieved. Furthermore, users must have at least minimal knowledge of the blockchain technology domain in order to take part in the system. However, with the use of blockchain technology, social innovations can be achieved that create new social relations, involving new ways of

communicating. Adaptation is more important than automation or accessibility because the system will be in the trust ecosystem [143].

Using a validium type scaling solution, the transaction throughput of the system does not depend on the underlying decentralized settlement component. The used protocol has no internal limitations in terms of off-chain transaction execution. The performance of the system depends on the HEI hardware resources. zkEVM can handle up to 2000 transactions per second according to the statement of the Polygon team. However, in terms of the proposed system, even more can be achieved by off-chain execution of transactions and off-chain storage of transaction data. In this case, only hardware constraints limit the scalability of the system.

As already mentioned, time-consuming proof generation reduces transaction throughput and increases transaction latency. Considering the current situation, the Polygon zkEVM prover is capable of validating 500K gas units on a single CPU server (64 cores) in about 5 min, which can be significantly reduced further with the use of special hardware resources. However, immediate request–response interactions cannot be possible due to the long time period of proof generation.

Last but not least, it is important to mention SSI (self-sovereign identity) in connection with the proposed education system. It is a set of technologies which give back control to individuals over their digital identity. One of the main building blocks is the verifiable credential. In Section 2.1 of this book [144], the conditions of the verifiable credential are examined in relation to SSI. In brief, a credential must be verifiable in some way. The verifier must be able to determine who issued the credential and that the credential is not tampered, expired, or revoked. The proposed system fulfills these requirements. The entities are able to complete the verification process without third-party involvement. For interaction without the involvement of a third party, users need to store their own data in some form. Secure data storage (in Section 9.8.5 of this book [144]) is the most suitable for this. However, due to the early adaptation of blockchain technology and the support of the users, these documents are parallel stored in a centralized manner with the use of the DAC. This system property is configurable and can be left out later.

Different cultural differences must be taken into account when SSI technology is mentioned. In Eastern culture, there is a stronger emphasis on communal governance than on individual control (14.1 section in this book [144]). Furthermore, clear governance laws and rules are necessary for the blockchain-based educational system, so that the verification of the information is acceptable, for example, who can be the issuer, and who can host nodes [138]. However, the current state is characterized by a lack of government support and industry standards, and high regulatory uncertainty [145]. Examining the current trends, it is rare that the credentials stored only on the blockchain are legally recognized [139].

9. Discussion

All in all, after a detailed discussion of all functionalities and challenges corresponding to an HEI's system it is following that the recommended system includes all defined functionalities and satisfies all mentioned challenges. Compared with other systems which are defined in related works, it implies that the recommended system is the only system with these capabilities. This is based on a different architectural structure which implies a different trust model compared to others. There is no other system from the mentioned ones with a similar trust model or architecture.

All mentioned related works are compared with the recommended system in Tables 6 and 7. The comparison implies that there are few complex systems which implement almost all functionalities. Essentially, the systems concentrate on the realization of a few functionalities. The systems which are defined in the [4,5] papers have similar functional properties as the proposed system. However, it is important to note that these systems operate in a permissioned network. This does not allow for verification of anything in a trustlessness manner.

Table 6. Comparison with the mentioned systems.

	[46]	[47]	[4]	[44]	[39]	[38]	[54]	[55]	**This Paper**
Certificates issuing and verification	✓	✓	✓	✓	✓	✓		✓	✓
Data management	ON	OF	ON	OF	OF	OF	ON	ON	OF
Credit transfer			✓				✓		✓
Admissions			✓				✓		✓
Student assessments and exams			✓				✓	✓	✓
Payments									✓
Privacy	EN	CS	PN	CS	EN	CS	PN	PN	PN
Settlement layer	PN	PL	PN	PN	PN	PL	PN	PN	PL

PN: Permissioned Network, **PL**: Permissionless Network, **EN**: Encrypted, **CS**: Centralized Storage, **OC**: On-chain, **OF**: Off-chain.

Table 7. Comparison with the mentioned systems.

	[15]	[52]	[50]	[5]	[56]	[63]	[62]	[6]	**This Paper**
Certificates issuing and verification				✓	✓			✓	✓
Data management	ON	ON	ON	OF	OF	ON	ON	OF	OF
Credit transfer	✓			✓				✓	✓
Admissions	✓	✓	✓	✓		✓	✓	✓	✓
Student assessments and exams	✓	✓	✓	✓				✓	✓
Payments							✓		✓
Privacy	PN	PL	PN	PN	CS	PN	PN	PL	PN
Settlement layer	PN	PL	PN	PN	PL	PN	PN	PL	PL

PN: Permissioned Network, **PL**: Permissionless Network, **EN**: Encrypted, **CS**: Centralized Storage, **OC**: On-chain, **OF**: Off-chain.

Vitalik Buterin defined different trust models in this article corresponding to the blockchain networks. Based on the article, the mentioned systems can be divided into different categories, as Figure 6 shows. Since the mentioned systems are operated by a centralized entity, these systems belong to the centralized category, 1 of 1. In this category, the behavior of a certain actor determines the correct behavior of the system. The proposed system also falls into this category, but with the usage of zkEVM, the possibility of invalid state transitions, and transactions, is reduced to the minimum. The centralized entity, the operator, can only censor transactions. Furthermore, due to the mentioned modular architecture, data retrieval is still verifiable if there is only one honest entity in the system, 1 of N (small N). This is the case when DAC can be more powerful than a single provider. Furthermore, the verification is independent of the HEI system, because it is based on the permissionless network, while in the other mentioned systems with the same trust assumption, it is not possible to decide whether the result of a request is correct. Thus, the trust model of the proposed system in the worst case is the same as the mentioned systems.

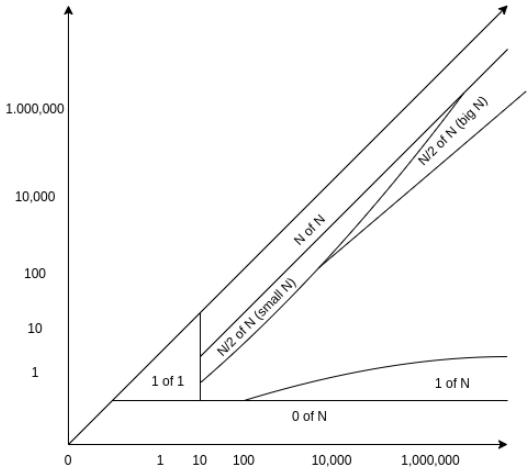

Figure 6. The different trust models.

The modular architecture ensures that the client's dependence on the centralized entity is minimized. The client functions are divided into three categories: verification (system or data verification), request (state transition request), and data retrieval (file or information retrieval). The architectural structure of the mentioned systems is categorized in a permissioned or permissionless category. In the case of permissioned category, the client is fully exposed to the permissioned network, a central entity, as Figure 7 shows. The client has no information about whether the system is working properly or whether the returned data is correct. The client can only interact with the system with the help of a third party. In the case of a permissionless manner, a permissionless distributed ledger is used as the system, as Figure 7 shows. With this type of system, privacy concerns arise, as it is completely public and immutable. Furthermore, the cost of the system increases, since the given system is integrated into a decentralized system that is also used by other entities. In the case of the proposed system, the client can verify the returned results by relying on a permissionless network, as shown in Figure 8. Furthermore, during data retrieval, the client can prove data authenticity without the centralized entity. The assumption remains at the same level as in the other categories.

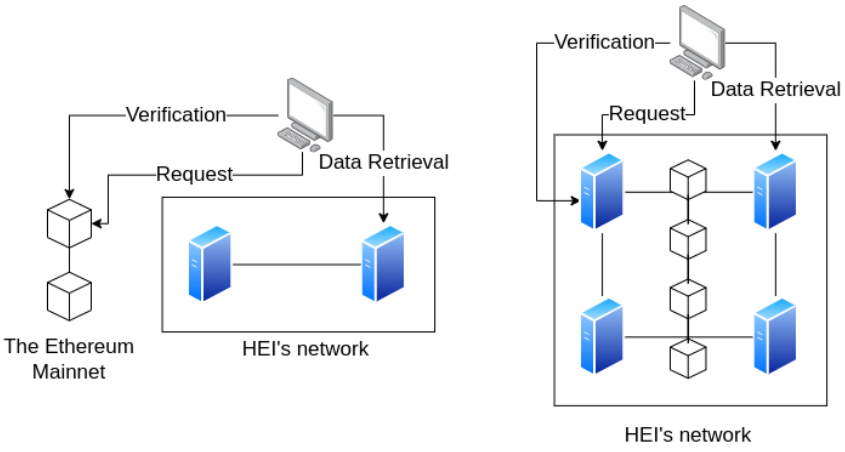

Figure 7. The connections in permissioned and permissionless systems.

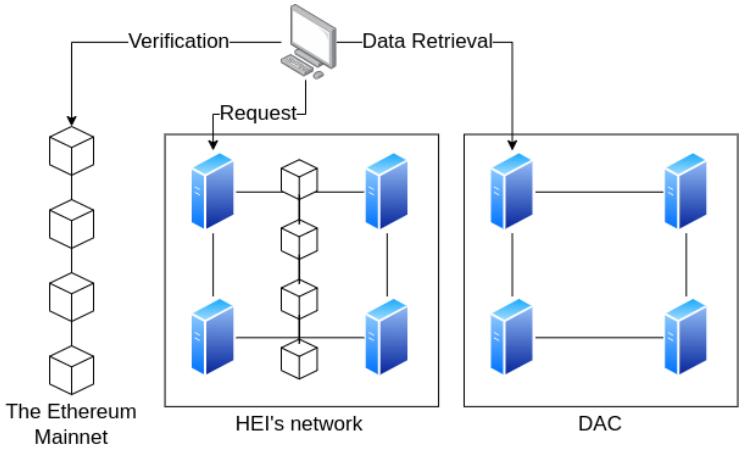

Figure 8. The connections in the recommended system.

Due to the cost of the settlement component transactions, it is advisable to set a predetermined transaction number and fee. However, considering this cost, the designed system always offers a cheaper alternative than the ones presented in the related works section. It is a subservient idea to divide the given events into well-defined time periods in order to reduce the number of necessary interactions with the settlement component. In addition, due to the long time of proof generation, it is necessary to supplement the system with a permissioned storage component, in which the users can receive and retrieve their requested Merkle proof for the corresponding state root, since it is not possible to determine the proof when the interaction happens. It is necessary to wait until the transactions are ordered and sorted into a batch by the operator. The usage of multi-signature interactions and all data that can be stored off-chain in the private IPFS can help to reduce the mentioned long time of proof generation, as the more complex operations imply a longer time in the proof generation.

10. Conclusions

The designed system satisfies the general higher education requirements and solves the mentioned challenges. The hybrid approach provides a possibility to improve the decentralization and security measures of a higher education system which by default is strongly dependent on the centralized structure. All events and certifications can be verified through the decentralized settlement component in a cost-effective and less bureaucratic and time-consuming way than in centralized traditional systems.

However, it is important to examine the designed system from the point of view of the current acceptance of this technology as a trustful verification method. Blockchain technology is still in its early stages and its adaptation is not widespread. Furthermore, it is necessary to validate the GDPR compatibility of the designed system in order to make sure the system complies with all regulations.

The used DAC is based on the assumption that there is a trusted organization in the committee. In this case, when a given HEI maintains its own education system, it is better to make the system wider with more involved organizations. This reduces the possibility of attacks and setbacks. Organizations participating in the committee can be held responsible for abuses committed by them. On the other hand, the most serious consequence of abuse is the freezing of the deposited funds of users in the bridge. The freezing is a temporary problematic period rather than the final state of the system. The system can be restored after the emergency state and the users can withdraw their funds.

Using Polygon zkEVM, one of the most up-to-date and fastest open-source projects, it is possible to implement a fully verifiable higher education system in a decentralized manner. By using zk-STARK proofs of the system, the on-chain transaction cost is reduced to a minimum. However, it is important to note that all zkEVM developments are currently

"works in progress" and the release of fully functional systems is planned for the second half of this year; however, the basic functionalities are already available.

After the zkEVM fully EVM-compatible release, a deeper examination and evaluation of gas consumption is necessary for the future. It is indispensable to have a wide-range examination of the cost consequences of decentralization. In addition, the examination of hardware consumption is also important, so that it will be possible to define the entry threshold to maintain the system and the cost of a semester more easily.

The recommended system building in a modular stack uses validity proof for providing the consistent working of a tamper-proof centralized system. It provides possibilities for the HEI to maintain the system in a more secure and decentralized manner on its own.

Author Contributions: Conceptualization, D.L.F. and A.K.; methodology, D.L.F. and A.K.; software, D.L.F.; validation, D.L.F. and A.K.; investigation, D.L.F. and A.K.; writing—original draft preparation, D.L.F. and A.K.; writing—review and editing, D.L.F. and A.K.; supervision, A.K.; project administration, A.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by grants of "Application Domain Specific Highly Reliable IT Solutions" project that has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the Thematic Excellence Programme TKP2020-NKA-06 (National Challenges Subprogramme) funding scheme.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACFE	Association of Certified Fraud Examiners
ASIC	Application-specific integrated circuit
CAP	Consistency, Availability, Partition
CRS	Common Reference String
DAC	Data Availability Committee
EU	European Union
EVM	Ethereum virtual machine
FPGA	Field programmable gate array
GDPR	General Data Protection Regulation
HEI	Higher education institution
PoC	Proof of concept
SSI	Self-sovereign identity
UTXO	Unspent transaction output
zkEVM	Zero-knowledge Ethereum virtual machine
zk-SNARK	Zero-knowledge succinct non-interactive arguments of knowledge
zk-STARK	Zero-knowledge scalable transparent argument of knowledge

References

1. Alnafrah, I.; Mouselli, S. Revitalizing blockchain technology potentials for smooth academic records management and verification in low-income countries. *Int. J. Educ. Dev.* **2021**, *85*, 102460.
2. Vidal, F.R.; Gouveia, F.; Soares, C. Revocation mechanisms for academic certificates stored on a blockchain. In Proceedings of the 2020 15th Iberian Conference on Information Systems and Technologies (CISTI), Seville, Spain, 24–27 June 2020.
3. Ezeudu, F.O.; Eya, N.M.; Nworgi, H.I. Application of Blockchain-based Technology in Chemistry Education Students. Data Management. *Int. J. Database Theory Appl.* **2018**, *11*, 11–22.
4. Srivastava, A.; Bhattacharya, P.; Singh, A.; Mathur, A.; Prakash, O.; Pradhan, R. A distributed credit transfer educational framework based on blockchain. In Proceedings of the 2018 Second International Conference on Advances in Computing, Control and Communication Technology (IAC3T), Allahabad, India, 21–23 September 2018.

5. Deenmahomed, Haïdar AM, Micheal M. Didier, and Roopesh K. Sungkur. The future of university education: Examination, transcript, and certificate system using blockchain. *Comput. Appl. Eng. Educ.* **2021**, *29*, 1234–1256.
6. Kistaubayev, Y.; Mutanov, G.; Mansurova, M.; Saxenbayeva, Z.; Shakan, Y. Ethereum-Based Information System for Digital Higher Education Registry and Verification of Student Achievement Documents. *Future Internet* **2022**, *15*, 3.
7. Fenichel, M.; Schweingruber, H.A. *Surrounded by Science: Learning Science in Informal Environments*; National Academies Press: Washington, DC, USA, 2010.
8. Androutsos, A.; Brinia, V. Developing and piloting a pedagogy for teaching innovation, collaboration, and co-creation in secondary education based on design thinking, digital transformation, and entrepreneurship. *Educ. Sci.* **2019**, *9*, 113.
9. Antonaci, A.; Klemke, R.; Lataster, J.; Kreijns, K.; Specht, M. Gamification of MOOCs adopting social presence and sense of community to increase user's engagement: An experimental study. In Transforming Learning with Meaningful Technologies, Proceedings of the European Conference on Technology Enhanced Learning, Delft, The Netherlands, 16–19 September 2019; Springer: Cham, Switzerland, 2019.
10. Gopane, T.J. Blockchain Technology and Smart Universities. In Proceedings of the 4th International Conference on the Internet, Cyber Security and Information Systems, Johannesburg, South Africa, 31 October–1 November 2019.
11. Oliver, M.; Moreno, J.; Prieto, G.; Benitez, D. Using blockchain as a tool for tracking and verification of official degrees: Business model. In Proceedings of the 29th European Regional Conference of the International Telecommunications Society (ITS): "Towards a Digital Future: Turning Technology into Markets?", Trento, Italy, 1–4 August 2018.
12. Healy, T.; Cote, S. *The Well-Being of Nations: The Role of Human and Social Capital. Education and Skills*; Organisation for Economic Cooperation and Development: Paris, France, 2001.
13. Baum, S. *Higher Education Earnings Premium: Value, Variation, and Trends*; Urban Institute: Washington, DC, USA, 2014.
14. Pelaitis, D.; Spathoulas, G. Developing a universal, decentralized and immutable Erasmus credit transfer system on blockchain. In Proceedings of the 2018 Innovations in Intelligent Systems and Applications (INISTA), Thessaloniki, Greece, 3–5 July 2018.
15. Turkanović, M.; Hölbl, M.; Košič, K.; Heričko, M.; Kamišalić, A. EduCTX: A blockchain-based higher education credit platform. *IEEE Access* **2018**, *6*, 5112–5127.
16. Ghazali, O.; Saleh, O.S. A graduation certificate verification model via utilization of the blockchain technology. *J. Telecommun. Electron. Comput. Eng.* **2018**, *10*, 29–34.
17. Liu, Z. *Paper to Digital: Documents in the Information Age*; ABC-CLIO: Santa Barbara, CA, USA, 2008.
18. Abougalala, R.A.; Amasha, A.; Areed, M.F.; Alkhalaf, S.; Khairy, D. Blockchain-enabled smart university: A framework. *J. Theor. Appl. Inf. Technol.* **2020**, *98*, 3531–3543.
19. Awaji, B.; Solaiman, E.; Marshall, L. Investigating the requirements for building a blockchain-based achievement record system. In Proceedings of the 5th International Conference on Information and Education Innovations, London, UK, 26–28 July 2020.
20. Moore, M.G. A sad reminder that diploma mills are still with us. *Am. J. Distance Educ.* **2009**, *23*, 175–178.
21. Yumna, H.; Khan, M.M.; Ikram, M.; Ilyas, S. Use of blockchain in education: A systematic literature review. In Proceedings of the 11th Asian Conference on Intelligent Information and Database Systems, Yogyakarta, Indonesia, 8–11 April 2019; Springer: Cham, Switzerland, 2019.
22. Ocheja, P.; Flanagan, B.; Ueda, H.; Ogata, H. Managing lifelong learning records through blockchain. *Res. Pract. Technol. Enhanc. Learn.* **2019**, *14*, 4.
23. Ezell, A.; Bear, J. *Degree Mills: The Billion-Dollar Industry That Has Sold over a Million Fake Diplomas*; Pyr Books: Amherst, NY, USA, 2005.
24. Grolleau, G.; Lakhal, T.; Mzoughi, N. An introduction to the Economics of Fake Degrees. *J. Econ. Issues* **2008**, *42*, 673–693.
25. Cohen, E.B.; Winch, R. *Diploma and Accreditation Mills: New Trends in Credential Abuse*; Verifile Accredibase: Bedford, UK, 2011.
26. Garwe, E.C. Qualification, award and recognition fraud in higher education in Zimbabwe. *J. Stud. Educ.* **2015**, *5*, 119–135.
27. Chen, G.; Xu, B.; Lu, M.; Chen, N.S. Exploring blockchain technology and its potential applications for education. *Smart Learn. Environ.* **2018**, *5*, 1.
28. Šipek, M.; Žagar, M.; Mihaljević, B.; Drašković, N. Application of Blockchain Technology for Educational Platform. In *Human Interaction, Emerging Technologies and Future Systems V, Proceedings of the 5th International Virtual Conference on Human Interaction and Emerging Technologies, IHIET 2021, 27–29 August 2021 and the 6th IHIET: Future Systems (IHIET-FS 2021), 28–30 October 2021, France*; Springer: Cham, Switzerland, 2021.
29. Arndt, T. Empowering University Students with Blockchain-Based Transcripts. In Proceedings of the 15th International Association for Development of the Information Society, Budapest, Hungary, 21–23 October 2018.
30. Bore, N.; Karumba, S.; Mutahi, J.; Darnell, S.S.; Wayua, C.; Weldemariam, K. Towards blockchain-enabled school information hub. In Proceedings of the Ninth International Conference on Information and Communication Technologies and Development, Lahore, Pakistan, 16–19 November 2017.
31. Alammary, A.; Alhazmi, S.; Almasri, M.; Gillani, S. Blockchain-based applications in education: A systematic review. *Appl. Sci.* **2019**, *9*, 2400.
32. Loukil, F.; Abed, M.; Boukadi, K. Blockchain adoption in education: A systematic literature review. *Educ. Inf. Technol.* **2021**, *26*, 5779–5797.
33. Hameed, B.; Khan, M.M.; Noman, A.; Ahmad, M.J.; Talib, M.R.; Ashfaq, F.; Usman, H.; Yousaf, M. A review of Blockchain based educational projects. *Int. J. Adv. Comput. Sci. Appl.* **2019**, *10* , 491–499.

34. Awaji, B.; Solaiman, E.; Albshri, A. Blockchain-based applications in higher education: A systematic mapping study. In Proceedings of the 5th International Conference on Information and Education Innovations, London, UK, 26–28 July 2020.
35. Fedorova, E.P.; Skobleva, E.I. Application of blockchain technology in higher education. *Eur. J. Contemp. Educ.* **2020**, *9*, 552–571.
36. Jain, S.C. Problems in international protection of intellectual property rights. *J. Int. Mark.* **1996**, *4*, 9–32.
37. Grossman, G.M.; Lai, E.L.C. International protection of intellectual property. *Am. Econ. Rev.* **2004**, *94*, 1635–1653.
38. Vidal, F.R.; Gouveia, F.; Soares, C. Blockchain application in higher education diploma management and results analysis. *Adv. Sci. Technol. Eng. Syst.* **2020**, *5*, 871–882.
39. Ayub, Khan, A.; Laghari, A.A.; Shaikh, A.A.; Bourouis, S.; Mamlouk, A.M.; Alshazly, H. Educational Blockchain: A Secure Degree Attestation and Verification Traceability Architecture for Higher Education Commission. *Appl. Sci.* **2021**, *11*, 10917.
40. Grech, A. and Camilleri, A. F. **(2017)** Blockchain in Education. Inamorato dos Santos, A. (ed.) EUR 28778 EN; http://doi.org/10.2760/60649
41. Schmidt, P. *Blockcerts—An Open Infrastructure for Academic Credentials on the Blockchain*; MLLearning: Gilgit, Pakistan, 2016.
42. Loewen, J.; Suhonen, J. I-DIGEST framework: Towards authentic learning for indigenous learners. *Smart Learn. Environ.* **2018**, *5*, 4.
43. Sharples, M.; Domingue, J. The blockchain and kudos: A distributed system for educational record, reputation and reward. In Proceedings of the 11th European Conference on Technology Enhanced Learning, Lyon, France, 13–16 September 2016; Springer: Cham, Switzerland, 2016.
44. Han, M.; Li, Z.; He, J.; Wu, D.; Xie, Y.; Baba, A. A novel blockchain-based education records verification solution. In Proceedings of the 19th Annual SIG Conference on Information Technology Education, Fort Lauderdale, FL, USA, 3–6 October 2018.
45. Yankov, B. Storing and Retrieval of Structured Data on blockchain with BlockChi and Ethereum. *Vanguard Sci. Intstruments Manag.* **2018**, *14* , 1–10.
46. Rahardja, U.; Hidayanto, A.N.; Putra, P.O.H.; Hardini, M. Immutable Ubiquitous Digital Certificate Authentication Using Blockchain Protocol. *J. Appl. Res. Technol.* **2021**, *19*, 308–321.
47. Cheng, J.C.; Lee, N.Y.; Chi, C.; Chen, Y.H. Blockchain and smart contract for digital certificate. In Proceedings of the 2018 IEEE International Conference on Applied System Invention (ICASI), Chiba, Japan, 13–17 April 2018.
48. Nizamuddin, N.; Salah, K.; Azad, M.A.; Arshad, J.; Rehman, M.H. Decentralized document version control using ethereum blockchain and IPFS. *Comput. Electr. Eng.* **2019**, *76*, 183–197.
49. Yakubov, A.; Shbair, W.; Wallbom, A.; Sanda, D. A blockchain-based PKI management framework. In Proceedings of the First IEEE/IFIP International Workshop on Managing and Managed by Blockchain (Man2Block) Colocated with IEEE/IFIP NOMS 2018, Tapei, Tawain, 23–27 April 2018.
50. Sosa, R.; del Pino, M.; Cabrera, D.; Moreno, B. *Blockchain and Smart Contracts for Education*; Munich Personal RePEc Archive: Munich, Germany, 2020.
51. Shen, H.; Xiao, Y. Research on online quiz scheme based on double-layer consortium blockchain. In Proceedings of the 2018 9th International Conference on Information Technology in Medicine and Education (ITME), Hangzhou, China, 19–21 October 2018.
52. Lizcano, D.; Lara, J.A.; White, B.; Aljawarneh, S. Blockchain-based approach to create a model of trust in open and ubiquitous higher education. *J. Comput. High. Educ.* **2020**, *32*, 109–134.
53. Wu, T.; Chang, M. The application framework of blockchain technology in higher education based on the smart contract. In Proceedings of the 2021 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), Macau, China, 5–7 December 2021.
54. Morisio, M.; Ardito, L.; Yokubov, B. Blockchain Based Storage of Students Career. Ph.D. Thesis, Politecnico di Torino, Turin, Italy, 2018.
55. Bhosale, H.; Kanki, R.; Jaiswal, G. Revolutionizing Verification and Management of Educational Certificates with Self-Sovereign Student Identities using Blockchain. *Int. Res. J. Eng. Technol.* **2021**, *8*, 4189–4194.
56. Ataşen, K.; Aslan, B.A. Blockchain Based Digital Certification Platform: CertiDApp. *J. Multidiscip. Eng. Sci. Technol.* **2020**, *7*, 12252–12255.
57. Zhai, X.; Pang, S.; Wang, M.; Qiao, S.; Lv, Z. TVS: A trusted verification scheme for office documents based on blockchain. *Complex Intell. Syst.* **2022** , Open Access, https://doi.org/10.1007/s40747-021-00617-1. 1–13.
58. Das, M.; Tao, X.; Liu, Y.; Cheng, J.C. A blockchain-based integrated document management framework for construction applications. *Autom. Constr.* **2022**, *133*, 104001.
59. Santos, J.; Duffy, K.H. A Decentralized Approach to Blockcerts Credential Revocation. 2018. Available online: https://github.com/WebOfTrustInfo/rwot5-boston/blob/master/final-documents/blockcerts-revocation.md (accessed on 25 September 2022).
60. Martiri, E.; Muca, G.; Xhina, E.; Hoxha, K. DMS-XT: A Blockchain-based Document Management System for Secure and Intelligent Archival. In *RTA-CSIT* ; CEUR-WS, Albania; 70–74. 2018.
61. Curmi, A.; Inguanez, F. Blockchain based certificate verification platform. In Proceedings of the BIS 2018 International Conference on Business Information Systems, Berlin, Germany, 18–20 july 2018; Springer: Cham, Switzerland, 2018.
62. Bedi, P.; Gole, P.; Dhiman, S.; Gupta, N. Smart contract based central sector scheme of scholarship for college and university students. *Procedia Comput. Sci.* **2020**, *171*, 790–799.

63. Mori, K.; Miwa, H. Digital university admission application system with study documents using smart contracts on blockchain. In Proceedings of the 11th International Conference on Intelligent Networking and Collaborative Systems, Oita, Japan, 5–7 September 2019; Springer: Cham, Switzerland, 2019.
64. Rooksby, J.; Dimitrov, K. Trustless education? A blockchain system for university grades. In *New Value Transactions: Understanding and Designing for Distributed Autonomous Organisations, Proceedings of the DIS'17: Designing Interactive Systems Conference, Edinburgh, UK, 10–14 June* 2017.
65. Rashid, M.A.; Deo, K.; Prasad, D.; Singh, K.; Ch, ; S.; Assaf, M. TEduChain: A platform for crowdsourcing tertiary education fund using blockchain technology. *arXiv* **2019**, arXiv:1901.06327.
66. Nofer, M.; Gomber, P.; Hinz, O.; Schiereck, D. Blockchain. *Bus. Inf. Syst. Eng.* **2017**, *59*, 183–187. https://doi.org/10.1007/s12599-017-0467-3.
67. Zhou, L.; Zhang, L.; Zhao, Y.; Zheng, R.; Song, K. A scientometric review of blockchain research. *Inf. Syst. e-Bus. Manag.* **2021**, *19*, 757–787.
68. Capetillo, A.; Camacho, D.; Alanis, M. Blockchained education: Challenging the long-standing model of academic institutions. *Int. J. Interact. Des. Manuf. (IJIDeM)* **2022**, *16*, 791–802.
69. Rahardja, U.; Hidayanto, A.N.; Hariguna, T.; Aini, Q. Design framework on tertiary education system in Indonesia using blockchain technology. In Proceedings of the 2019 7th International Conference on Cyber and IT Service Management (CITSM), Jakarta, Indonesia, 6–8 November 2019; Volume 7.
70. Clohessy, T.; Acton, T. Investigating the influence of organizational factors on blockchain adoption: An innovation theory perspective. *Ind. Manag. Data Syst.* **2019**, *119*, 1457–1491.
71. Taherdoost, H. A Critical Review of Blockchain Acceptance Models—Blockchain Technology Adoption Frameworks and Applications. *Computers* **2022**, *11*, 24.
72. Liu, L.; Han, M.; Zhou, Y.; Parizi, R. Eˆ2 C-Chain: A two-stage incentive education employment and skill certification blockchain. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019.
73. Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 22 September 2022).
74. Chohan, U.W. The double spending problem and cryptocurrencies. *SSRN Electron. J.* **2021** , 3090174. http://dx.doi.org/10.2139/ssrn.3090174.
75. Warfield, A.; Coady, Y.; Hutchinson, N. Identifying open problems in distributed systems. In Proceedings of the European Research Seminar on Advances in Distributed Systems (ERSADS), Bertinoro, Italy, 14–18 May 2001.
76. Greenspan, G. Avoiding the Pointless Blockchain Project. 2015. Available online: https://www.multichain.com/blog/2015/11/avoiding-pointless-blockchain-project (accessed on 22 September 2022).
77. Bergamo, P.; D'Arco, P.; De Santis, A.; Kocarev, L. Security of public-key cryptosystems based on Chebyshev polynomials. *IEEE Trans. Circuits Syst. Regul. Pap.* **2005**, *52*, 1382–1393.
78. Szabo, N. Formalizing and securing relationships on public networks. *First Monday* **1997**, *2* , 1–21.
79. Mohanta, B.K.; Panda, S.S.; Jena, D. An overview of smart contract and use cases in blockchain technology. In Proceedings of the 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bengaluru, India, 10–12 July 2018.
80. Fairfield, J.A. Smart contracts, Bitcoin bots, and consumer protection. *Wash. Lee L. Rev. Online* **2014**, *71*, 35.
81. Bahga, A.; Madisetti, V.K. Blockchain platform for industrial internet of things. *J. Softw. Eng. Appl.* **2016**, *9*, 533–546.
82. Buterin, V. A next-generation smart contract and decentralized application platform. *White Pap.* **2014**, *3*, 1–36.
83. Liu, H.; Liu, C.; Zhao, W.; Jiang, Y.; Sun, J. S-gram: Towards semantic-aware security auditing for ethereum smart contracts. In Proceedings of the 2018 33rd IEEE/ACM International Conference on Automated Software Engineering (ASE), Montpellier, France, 3–7 September 2018.
84. Kim, S.; Song, J.; Woo, S.; Kim, Y.; Park, S. Gas consumption-aware dynamic load balancing in ethereum sharding environments. In Proceedings of the 2019 IEEE 4th International Workshops on Foundations and Applications of Self* Systems (FAS*W), Umea, Sweden, 16–20 June 2019.
85. Gilbert, S.; Lynch, N. Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. *ACM SIGACT News* **2002**, *33*, 51–59.
86. Zhou, Q.; Huang, H.; Zheng, Z.; Bian, J. Solutions to scalability of blockchain: A survey. *IEEE Access* **2020**, *8*, 16440–16455.
87. Brunnermeier, M.; Abadi, J. *Blockchain Economics. No. w25407*; National Bureau of Economic Research: Cambridge, MA, USA, 2018.
88. Monte, G.D.; Pennino, D.; Pizzonia, M. Scaling blockchains without giving up decentralization and security: A solution to the blockchain scalability trilemma. In Proceedings of the 3rd Workshop on Cryptocurrencies and Blockchains for Distributed Systems, London, UK, 25 September 2020.
89. Zheng, P.; Zheng, Z.; Luo, X.; Chen, X.; Liu, X. A detailed and real-time performance monitoring framework for blockchain systems. In Proceedings of the 2018 IEEE/ACM 40th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP), Gothenburg, Sweden, 30 May–1 June 2018.
90. Behnke, K.; Janssen, M.F.W.H.A. Boundary conditions for traceability in food supply chains using blockchain technology. *Int. J. Inf. Manag.* **2020**, *52*, 101969.

91. Polge, J.; Robert, J.; Le Traon, Y. Permissioned blockchain frameworks in the industry: A comparison. *ICT Express* **2021**, *7*, 229–233.
92. Su, Q.; Zhang, R.; Xue, R.; Li, P. Revocable attribute-based signature for blockchain-based healthcare system. *IEEE Access* **2020**, *8*, 127884–127896.
93. Jesse, Y.-H.; Ko, D.; Choi, S.; Park, S.; Smolander, K. Where is current research on blockchain technology?—A systematic review. *PLoS ONE* **2016**, *11*, e0163477.
94. Swan, M. *Blockchain: Blueprint for a New Economy*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2015.
95. Dabbagh, M.; Choo, K.K.R.; Beheshti, A.; Tahir, M.; Safa, N.S. A survey of empirical performance evaluation of permissioned blockchain platforms: Challenges and opportunities. *Comput. Secur.* **2021**, *100*, 102078.
96. Fekete, D.L. Kiss, A. A Survey of Ledger Technology-Based Databases. *Future Internet* **2021**, *13*, 197.
97. Helliar, C.V.; Crawford, L.; Rocca, L.; Teodori, C.; Veneziani, M. Permissionless and permissioned blockchain diffusion. *Int. J. Inf. Manag.* **2020**, *54*, 102136.
98. Marar, H.W.; Marar, R.W. HYBRID BLOCKCHAIN. *Jordanian J. Comput. Inf. Technol. (JJCIT)* **2020**, *6* , 317–325.
99. Cui, Z.; Fei, X.; Zhang, S.; Cai, X.; Cao, Y.; Zhang, W.; Chen, J. A hybrid blockchain-based identity authentication scheme for multi-WSN. *IEEE Trans. Serv. Comput.* **2020**, *13*, 241–251.
100. Zhu, S.; Cai, Z.; Hu, H.; Li, Y.; Li, W. zkCrowd: A hybrid blockchain-based crowdsourcing platform. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4196–4205.
101. Kantesariya, S.; Goswami, D. Determining optimal shard size in a hierarchical blockchain architecture. In Proceedings of the 2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Toronto, ON, Canada, 2–6 May 2020.
102. Cohen, S.; Goren, G.; Kokoris-Kogias, L.; Sonnino, A.; Spiegelman, A. Proof of Availability & Retrieval in a Modular Blockchain Architecture. *Cryptol. ePrint Arch.* **2022** , 1–20.
103. Ali, M.S.; Vecchio, M.; Antonelli, F. Enabling a blockchain-based IoT edge. *IEEE Internet Things Mag.* **2018**, *1*, 24–29.
104. Reynolds, P.; Irwin, A.S. Tracking digital footprints: Anonymity within the bitcoin system. *J. Money Laund. Control.* **2017** , Vol. 20 No. 2, pp. 172–189.
105. Voigt, P.; Von dem, Bussche, A. The eu general data protection regulation (gdpr). In *A Practical Guide*, 1st ed.; Springer International Publishing: Cham, Switzerland, 2017; Volume 10, 10-5555.
106. Tankard, C. What the GDPR means for businesses. *Netw. Secur.* **2016**, *2016*, 5–8.
107. IT Governance Privacy Team. *EU general data protection regulation (gdpr)—An implementation and compliance guide*; IT Governance Ltd.: Ely, UK, 2020.
108. Lyons, T. *EU Blockchain Observatory and Forum*; Workshop Report; Government Services and Digital Identity: Brussels, Belgium, 2018.
109. Truong, N.B.; Sun, K.; Lee, G.M.; Guo, Y. Gdpr-compliant personal data management: A blockchain-based solution. *IEEE Trans. Inf. Forensics Secur.* **2019**, *15*, 1746–1761.
110. Zemler, F.; Westner, M. Blockchain and GDPR: Application scenarios and compliance requirements. In Proceedings of the 2019 Portland International Conference on Management of Engineering and Technology (PICMET), Portland, OR, USA, 25–29 August 2019.
111. Vujičić, D.; Jagodić, D.; Ranđić, S. Blockchain technology, bitcoin, and Ethereum: A brief overview. In Proceedings of the 2018 17th International Symposium Infoteh-Jahorina (infoteh), East Sarajevo, Bosnia and Herzegovina, 21–23 March 2018.
112. Wang, D.; Zhou, J.; Wang, A.; Finestone, M. Loopring: A Decentralized Token Exchange Protocol. 2018. Available online: https://loopring.org/resources/en_whitepaper.pdf (accessed on 20 September 2022).
113. Rondelet, A. Zecale: Reconciling privacy and scalability on ethereum. *arXiv* **2020**, arXiv:2008.05958.
114. Danezis, G.; Meiklejohn, S. Centrally banked cryptocurrencies. *arXiv* **2015**, arXiv:1505.06895.
115. Chauhan, A.; Malviya, O.P.; Verma, M.; Mor, T.S. Blockchain and scalability. In Proceedings of the 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C), Lisbon, Portugal, 16–20 July 2018.
116. Hafid, A.; Hafid, A.S.; Samih, M. Scaling blockchains: A comprehensive survey. *IEEE Access* **2020**, *8*, 125244–125262.
117. Feng, L.; Zhang, H.; Chen, Y.; Lou, L. Scalable dynamic multi-agent practical byzantine fault-tolerant consensus in permissioned blockchain. *Appl. Sci.* **2018**, *8*, 1919.
118. Di Ciccio, C.; Gabryelczyk, R.; García-Bañuelos, L.; Hernaus, T.; Hull, R.; Indihar, Štemberger, M.; Staples, M. Business Process Management: Blockchain and Central and Eastern Europe Forum. In *Lecture Notes in Business Information Processing* ;Springer; Switzerland, 2019.
119. Bez, M.; Fornari, G.; Vardanega, T. The scalability challenge of ethereum: An initial quantitative analysis. In Proceedings of the 2019 IEEE International Conference on Service-Oriented System Engineering (SOSE), San Francisco, CA, USA, 4–9 April 2019.
120. Cortes-Goicoechea, M.; Franceschini, L.; Bautista-Gomez, L. Resource analysis of Ethereum 2.0 clients. In Proceedings of the 2021 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), Paris, France, 27–30 September 2021.
121. Wang, K. Research and Insights. 2022. Available online: https://content-hub-static.crypto.com/wp-content/uploads/2022/02/Forensics_of_Attacks_and_Exploits_in_DeFi.pdf (accessed on 20 September 2022).
122. Gangwal, A.; Gangavalli, H.R.; Thirupathi, A. A Survey of Layer-Two Blockchain Protocols. *arXiv* **2022**, arXiv:2204.08032.
123. Schaffner, T.; Schaer, F. Scaling Public Blockchains—A Comprehensive Analysis of Optimistic and Zero-Knowledge Rollups. Master's Thesis, Center for Innovative Finance, University of Basel, Basel, Switzerland, 2021.

124. Thibault, L.T.; Sarry, T.; Hafid, A.S. Blockchain Scaling using Rollups: A Comprehensive Survey. *IEEE Access* **2022**, *10*, 93039–93054.
125. Ben-Sasson, E.; Chiesa, A.; Tromer, E.; Virza, M. Succinct Non-Interactive zero knowledge for a von neumann architecture. In Proceedings of the 23rd USENIX Security Symposium (USENIX Security 14), San Diego, CA, USA, 20–22 August 2014.
126. Ben-Sasson, E.; Bentov, I.; Horesh, Y.; Riabzev, M. Scalable, transparent, and post-quantum secure computational integrity. *Cryptol. ePrint Arch.* **2018** , Volume 2018, 1–83.
127. Lesavre, L.; Varin, P.; Mell, P.; Davidson, M.; Shook, J. A taxonomic approach to understanding emerging blockchain identity management systems. *arXiv* **2019**, arXiv:1908.00929.
128. Tomaz, A.E.B.; Do Nascimento, J.C.; Hafid, A.S.; De Souza, J.N. Preserving privacy in mobile health systems using non-interactive zero-knowledge proof and blockchain. *IEEE Access* **2020**, *8*, 204441–204458.
129. Yang, X.; Li, W. A zero-knowledge-proof-based digital identity management scheme in blockchain. *Comput. Secur.* **2020**, *99*, 102050.
130. Teichler, U. *Changing Patterns of the Higher Education System. The Experience of Three Decades. Higher Education Policy Series, 5*; Taylor and Francis Group: Bristol, UK, 1988.
131. Hellwig, D.P.; Huchzermeier, A. Distributed ledger technology and fully homomorphic encryption: Next-generation information-sharing for supply chain efficiency. In *Innovative Technology at the Interface of Finance and Operations*; Springer: Cham, Switzerland, 2022; pp. 1–49.
132. Sun, X.; Yu, F.R.; Zhang, P.; Sun, Z.; Xie, W.; Peng, X. A survey on zero-knowledge proof in blockchain. *IEEE Netw.* **2021**, *35*, 198–205.
133. Sompolinsky, Y.; Lewenberg, Y.; Zohar, A. Spectre: A fast and scalable cryptocurrency protocol. *Cryptol. ePrint Arch.* **2016** , Volume 2016, 1–65.
134. Lin, Q.; Li, C.; Zhao, X.; Chen, X. Measuring decentralization in bitcoin and ethereum using multiple metrics and granularities. In Proceedings of the 2021 IEEE 37th International Conference on Data Engineering Workshops (ICDEW), Chania, Greece, 19–22 April 2021.
135. Wu, K.; Peng, B.; Xie, H.; Zhan, S. A Coefficient of Variation Method to Measure the Extents of Decentralization for Bitcoin and Ethereum Networks. *Int. J. Netw. Secur.* **2020**, *22*, 191–200.
136. Tas, E.N.; Boneh, D. Cryptoeconomic Security for Data Availability Committees. *arXiv* **2022**, arXiv:2208.02999.
137. Kang, P.; Yang, W.; Zheng, J. Blockchain Private File Storage-Sharing Method Based on IPFS. *Sensors* **2022**, *22*, 5100.
138. Grech, A.; Sood, I.; Ariño, L. Blockchain, self-sovereign identity and digital credentials: Promise versus praxis in education. *Front. Blockchain* **2021**, *4*, 616779.
139. Cumming, D.J.; Johan, S.; Pant, A. Regulation of the crypto-economy: Managing risks, challenges, and regulatory uncertainty. *J. Risk Financ. Manag.* **2019**, *12*, 126.
140. Fulmer, N. Exploring the legal issues of blockchain applications. *Akron Law Rev.* **2019**, *52*, 5.
141. Krichen, M.; Lahami, M.; Al–Haija, Q.A. Formal Methods for the Verification of Smart Contracts: A Review. In Proceedings of the 2022 15th International Conference on Security of Information and Networks (SIN), Sousse, Tunisia, 11–13 November 2022.
142. Almakhour, M.; Sliman, L.; Samhat, A.E.; Mellouk, A. Verification of smart contracts: A survey. *Pervasive Mob. Comput.* **2020**, *67*, 101227.
143. Shin, D.; Ibahrine, M. The socio-technical assemblages of blockchain system: How blockchains are framed and how the framing reflects societal contexts. *Digit. Policy, Regul. Gov.* **2020**, *22*, 245–263.
144. Preukschat, A.; Reed, D. *Self-Sovereign Identity*; Manning Publications: Shelter Island, NY, USA, 2021.
145. Rejeb, A.; Keogh, J.G.; Zailani, S.; Treiblmaier, H.; Rejeb, K. Blockchain technology in the food industry: A review of potentials, challenges and future research directions. *Logistics* **2020**, *4*, 27.