Bilgi Yönetimi Dergisi

Cilt: 5 Sayı: 1 Yıl: 2022

https://dergipark.org.tr/tr/pub/by

Peer-Reviewed Articles
Research Article

Article Info

Date submitted:	04.07.2021
Date accepted:	10.11.2021
Date early view:	25.04.2022
Date published:	30.06.2022

Makale Bilgisi

Gönderildiği tarih:	04.07.2021
Kabul tarihi:	10.11.2021
Erken görünüm:	25.04.2022
Yayınlanma tarihi:	30.06.2022

Keywords

Ethereum, Databases, Solidity Smart Contracts

Anahtar sözcükler

Ethereum, Veri Tabanları, Solidity Akıllı Sözleşmeler

DOI numarası

10.33721/by.961080

ORCID

0000-0001-8289-4690 (1)
0000-0002-0642-7908 (2)
0000-0002-2964-3320 (3)
0000-0002-3408-0083 (4)

A Software for Ethereum Databases and a Survey for Recommended Applications with ESRI Digital Maps Programming Software*

Ethereum Veri Tabanları İçin Bir Yazılım ve ESRI Dijital Haritalar Programlama Yazılımı ile Önerilen Uygulamalar İçin Bir Araştırma

Mounzer SAIJARE
Master Student, Ankara Yıldırım Beyazıt University, Department of Management Information Systems, saijare@yahoo.com

İhsan Tolga MEDENİ
Associate Prof. Dr., Ankara Yıldırım Beyazıt University, Department of Management Information Systems, tolgamedeni@gmail.com

Tunç Durmuş MEDENİ
Associate Prof. Dr., Ankara Yıldırım Beyazıt University, Department of Management Information Systems, tuncmedeni@gmail.com

Mehmet Serdar GÜZEL
Associate Prof. Dr., Ankara University, Department of Computer Engineering, mehmetserdarguzel@gmail.com

Abstract

Ethereum databases with their huge security and integrity can provide perfect mechanism for storing and sharing the data of the highly important projects. Qualitative and quantitative research methods were used in this paper to design software for Ethereum databases. Also, these methods were used to examine how and how much features can be added to the software. Designing the software considered the barriers and the limitations of Solidity smart contracts programming language and Ethereum blockchain platform. The achieved software can make any Ethereum database with parent-child relationships in between the tables. More professional features will be added in future research as soon as possible. The study used survey method to discuss recommended applications for the software integration with another software for ESRI digital maps programming. The recommended applications are the real estate business full automation and the transparent environmental management to protect the Mediterranean Sea from the pollution. However, most parts related to Ethereum databases software are covered. But to keep this paper in reasonable size, only the survey is included for the parts related to the software recommended applications with the digital maps software.

Öz

Yüksek güvenliğe sahip ve bütüncül yapısıyla Ethereum veri tabanları, yüksek derecede öneme sahip projelerin veri depolama ve veri paylaşımları için mükemmel bir mekanizma sağlayabilmektedir. Bu makalede, Ethereum veri tabanlarında kullanılmak üzere bir yazılım tasarlamak için nitel ve nicel metotlar kullanılmıştır. Ayrıca, bu metotlar, yazılıma nasıl ve ne kadar özellik eklenebileceğini incelemek için de kullanılmıştır. Yazılım tasarlanırken Ethereum blokzincir platformu ve bir akıllı sözleşme programlama dili olan Solidity'nin sınırlılıkları ve engelleri göz önünde bulundurulmuştur. Oluşturulan yazılım, Ethereum veri tabanında aralarında ebeveyn-çocuk ilişkileri olan herhangi bir tabloyu oluşturabilir. Gelecek çalışmada mümkün olduğu kadar daha fazla özellik eklenecektir.

**The research and publication process of this article was carried out in accordance with "Research and Publication Ethics". This paper is part of Mounzer Saijare's master thesis.*

Bu çalışma, ESRI dijital harita programlamada bir yazılımın diğer bir yazılımla entegre edilmesi için önerilen uygulamaları tartışmak için anket metodunu kullanmıştır. Önerilen uygulamalar; gayrimenkullerin tam otomasyonu ve Akdeniz'i çevre kirliliğinden korumak için şeffaf çevre yönetimidir. Ancak, Ethereum veri tabanlarına bağlı olan birçok kısım da kapsanmaktadır. Fakat, dijital harita yazılımı için önerilen uygulama yazılımlarıyla alakalı kısımlar için makaleyi makul bir boyutta tutmak adına sadece ankete yer verilmiştir.

1. Introduction

1.1. Introduction to Ethereum Blockchain

The exchange of information has been very important in human civilizations throughout history. From the starting point, the main target has been how to secure information sharing and how to make it decentralized (Fitzpatrick and McKeon, 2020). For ages, humans have used stamps and signatures to manage the trust and security of the information sharing.

Universities all over the world use paper certificates with stamps and signatures until this date. Also signed and stamped papers are used as contracts by companies, organizations and governments. The money also is special paper printed in specific secured methods. The secret writing (cryptography) has been used since many years ago with specific methods to write messages and to prevent access to these messages. Cryptography is came from Greek language, and it means the secret writing (Wikipedia contributors, 2019). MD5, SHA-1, SHA-2, SHA3 (Wikipedia Contributors, 2019) hash functions became the alternatives to old cryptography methods via the advancement of the computer era. For the encryption international standard there is now the Advanced Encryption Standard (AES) which replaced the Data Encryption Standard (DES) (Wikipedia Contributors, 2019).

David Chaum, the scientist in cryptography, made the first idea about making chain of data blocks which are cryptographically secured in 1982. Later in 1991 and 1992 Stuart Haber, W. Scott Stornetta and Dave Bayer made more important improvements to this idea (Sherman et al., 2019).

Anonymous identity programmer or group of programmers with nickname Satoshi Nakamoto in 2008 designed the first distributed program architecture that gives each node in the system the role of validating new data to be added to the cryptographically secured data block chain. It is impossible to edit any stored data as each block is connected to the block before, and modifying any block of data needs to modify all the blocks before, which is impossible according to the system mechanism. In 2009, Satoshi implemented his own Bitcoin distributed structure software and since then the Bitcoin transactions ledger database has been growing. This Bitcoin database is now over 200 GB in size. Satoshi Nakamoto called the Bitcoin network system block chain. Later, the concept became known as the blockchain (Narayanan et al., 2016).

19-year-old Vitalik Buterin emphasized that blockchain should be implemented for purposes other than payments and money transfers in 2013. He designed the Ethereum blockchain network in 2014 and Ethereum went online in 2015. Ethereum was the first blockchain platform to add the Solidity programming language for smart contract programming. Via Solidity, Ethereum smart contracts have brought unlimited applications to the blockchain space such as document verification, electronic voting, transparent management, transparent environmental management of natural resources and many other applications (Tapscott and Tapscott, 2018).

In this paper, the focus will be on the developed software to design any database on the Ethereum blockchain network with parent-child relationship between the tables. Subsequently, the paper will present a survey for future applications of the integration of this software with the ESRI (Esri: GIS Mapping Software, Spatial Data Analytics & Location Platform, 2019) digital map programming software.

The purpose of this paper is to show that a complex software can be achieved to design any complex database using special methods to deal with the barriers of the new Solidity programming language and to deal with the limitations of the new Ethereum virtual machine echo system. Also, apart from these barriers and limitations, the purpose of this paper is to show survey about how to integrate Ethereum database software with digital map programming software in applications that need a large amount of storage while storage is very expensive on the Ethereum blockchain network.

1.2. Problem Statement

With operating systems like Windows, Android, iOS, macOS, and programming language like C#, Java, Swift, and Python there will be unlimited possibilities for building any software. But with the very new virtual machine of the Ethereum blockchain network platform, and with Solidity, the very new Ethereum smart contract programming language, there will be limited possibilities for developing large, complex software.

Since the goal of this paper is to produce a program to design any database on the Ethereum platform with parent-child relationships in between the tables, this goal will face the hurdles and limitations of the Ethereum virtual machine and its Solidity programming language.

New mechanisms to produce this complex software must be developed in the Solidity programming language, which is the concept of this paper. On the other hand, we know that databases are the main tools for building any software or application, so the Ethereum database will be the main tool for building smart contracts, decentralized applications, or DApps. Since digital maps are a very important tool for information visualization, management and sharing, the focus of this paper will be on survey for future integration applications of Ethereum databases with ESRI digital maps programming software. But digital map applications need huge storage space to store digital map files, and storage prices on Ethereum are very expensive, so we need new mechanisms to solve this problem.

1.3. Statement of Purpose

The purpose of this paper is to develop a software to design any complex database on the Ethereum blockchain platform with parent-child relationships in between the tables. The paper will also present a survey for future applications of this software with another software for ESRI digital map programming.

1.4. Research Questions

The research study will answer the following questions (Cresswell, 2012):

1. With the limitations of Solidity language and time limit for the research, how can we create a software for Ethereum databases and how many features can be added to the software?
2. How and to what degree can surveying evaluate the future integration of the Ethereum databases software with another software for ESRI digital map programming in real estate business and in environmental management projects?

1.5. Null Hypotheses

The research contains two null hypotheses:

1. An editable database can be created on the Ethereum network.
2. The survey will support the idea that Ethereum databases software can be integrated with ESRI digital mapping programming software in real estate business and in environmental management projects.

1.6. Independent and Dependent Variables

For the first stage of software development, the independent variables are the sequence of variables, model structures, iterations, and programming time for each feature. Dependent variables are the features will be added to the software and the complexity of the objects transferred to Solidity.

For the second phase of surveying about the future applications for integration with ESRI digital map programming, the independent variables are transaction costs when using MD5 hash code, project types, and availability of digital map programming training. The dependent variables are the economic efficiency, economic value added, transparent management and the strategies that will be achieved.

2. Literature Review

In the literature review of this paper, the mechanism, limitations and barriers of the Ethereum blockchain platform and Solidity smart contract programming language will be discussed. Also, international case studies on Ethereum databases, blockchain databases, and decentralized databases in general will be discussed. Finally, ESRI digital map programming will be discussed.

2.1. Mechanisms of Ethereum 1.0 and Ethereum 2.0

In 2013, Vitalik Buterin was the first to talk about the idea of smart contracts. The idea came to Vitalik after he heard from his father, a professor of mathematics in Canada, about the Bitcoin blockchain platform, which started as the first crypto-secured blockchain in 2009. Vitalik saw on Bitcoin a great opportunity that comes from the nature of the blockchain, that to edit one data block it needs to edit all chain blocks before it, because all bocks are connected by cipher and timestamp hashes. This type of connection between the blocks makes it impossible to edit any data in any block in the Bitcoin blockchain (Narayanan et al., 2016; Tapscott and Tapscott, 2018).

In 2013, Vitalik said that the blockchain can be used in many fields other than payment. He introduced the idea of adding a programming language to the blockchain to produce bytecode that would be stored on the network. This bytecode will have the same properties as the data on the blockchain that cannot be modified, and this gives the token the trusted power to control very important actions just like sealed and signed paper contracts. Therefore, the name smart contracts came as the name of binary tokens on the Ethereum blockchain platform as the first blockchain platform with smart contracts. Smart contract applications are called decentralized applications or DApps (Wood, 2014; Tapscott and Tapscott, 2018).

2015 was the launch of the Ethereum blockchain 1.0 platform with its cryptocurrency Ethereum. This first phase passed 10 updates before the release of the first version of Ethereum 2.0 on December 1, 2020. Here's a quick look at the 12 platform updates (Ethereum Foundation, 2015; Ethereum Foundation, 2015; Ethereum Foundation, 2015; Ethereum Foundation, 2016; Ethereum Foundation, 2016; Ethereum Foundation, 2016; Ethereum Foundation, 2016; Ethereum Foundation, 2017; Ethereum Foundation, 2019; Ethereum Foundation, 2019; Ethereum Foundation, 2019; Hollander, 2020):

Table 1

The Structure of the Ethereum Databases in the Program

Update Name	Date	Block
Olympic	9.5.2015	This was just for testing purpose without blocks were mined
Frontier	30.7.2015	At block zero
Ice Age	8.9.2015	At block 200,000
Homestead	15.3.2016	At block 1,150,000
DAO Fork	20.7.2016	At block 1,920,000
Tangerine Whistle	18.10.2016	At block 2,463,000
Spurious Dragon	23.11.2016	At block 2,675,000
Byzantium	16.10.2017	At block 4,370,000
Constantinople (Petersburg)	28.2.2019	At block 7,280,000
Istanbul	8.12.2019	At block 9,069,000
Muir Glacier	1.1.2020	At block 9,200,000
Berlin	8.12.2019	At block 9,069,000

All of the above updates were based on Proof of Work (PoW) technology which needs quite a bit of computing with powerful computers and as a result PoW consumes a huge amount of electricity (ethereum.org, 2021).

On December 1, 2020, the first phase of Ethereum 2.0 began. This stage is considered to be stage 0 of Ethereum 2.0 and is called the Beacon Chain. The Beacon Series will work with Proof of Stake PoS

technology. The Beacon now operates separately from the Ethereum 1 main net in PoW. POS relies on user quotas to validate each transaction and mine a new block. PoS does not require powerful computers and it is environmentally friendly by consuming a small amount of electricity. PoS will lead to more network nodes participants which means more security and more decentralization. To participate in Beacon as a validator, the user must submit 32 Ether to the deposit contract address on the Ethereum network. From 04.11.2020 to 05.23.2021, this deposit contract had a balance of around 4,808,098 Ethers, but the Beacon network started on 12/1/2020 with more than 524,288 Ethers required to kick off (The Beacon Chain…, 2020).

The second step in Ethereum 2 will be the first phase which is two stages chains. The first version is expected in 2021, which will add 64 shards to the Beacon network, increasing transactions per second to 100,000. But in this version of shards chain smart contacts will not be implemented. The nodes will be responsible for selecting validators for each block in a dynamic random system. This system will make the possibility of the attack to be one in a billion, because the attackers must be at least 1/3 of all validators in Ethereum 2 and because the validators for each part are chosen dynamically and randomly by the Beacon chain (Ethereum 2.0 (Eth2), 2021).

The second version of Shard Chains, expected in 2021 or 2022, will have 3 possibilities about the implementation of the smart contracts:

- ✓ Assigning the implementation of smart contacts to Ethereum 1.0 network which will be transferred to PoS and will be added to the shards network. This option considers keeping transactions per second as much as 100,000.
- ✓ Add smart contracts executing for some of the 64 shards.
- ✓ More studies will be done on the 64 shards and the implementation of smart contracts.

Finally, the final step of Ethereum 2.0, expected in 2021 or 2022, will be the docking phase 2. In this final phase, Ethereum 1.0 will be moved from PoW to PoS and will join the shards blockchain. Also, smart contracts implementation will be added to Ethereum 2.0 (Ethereum.Org, 2021).

As Ethereum transactions per second will be raised from 15 in Ethereum 1.0 PoW to 100,000 in Ethereum 2.0 PoS, this will lead to new generations of DApps.

2.2. Solidity and Ethereum Smart Contract Programming Language and its Limitations and Barriers

Solidity, the smart contract development language was introduced by Gavin Wood in 2014. The Ethereum project team then started developing Solidity with team manager Christian Reitwiessner. Solidity is working on the Ethereum Virtual Machine EVM. Solidity operations that will change the state of the Ethereum database will require payment. The payment unit for the EVM is the gas unit. The user can set the unit price of gas in Ether for each transaction. The price of gas will affect the time and possibilities of transactions that miners will accept. Also, the user can set the gas limit for each transaction to the EVM. Gas limit entry is maximum amount of gas units that the user allows to be spent for specific transaction.

Solidity has three main types of functions (Solidity 0.8.6 documentation, 2021):

- ✓ Functions that change the state of the Ethereum database. These functions will consume gas units and have gas price and gas limit specifications.
- ✓ Functions that do not change the state of the Ethereum database but read information from that database.
- ✓ The functions that only performs computing operations on the EVM without changing the state or reading from the Ethereum database.

Since it is a very new language and is based on EVM on the blockchain, Solidity has these drawbacks and limitations (Solidity 0.8.6 documentation, 2021):

- ✓ Not all variant types are supported as normal programming language like C Sharp and Java. For example, these types are not supported: double and date.
- ✓ Structure variables cannot be sent to and cannot be received from public and external functions.
- ✓ Conversions between variable types in most cases are not possible.

- ✓ Functions have maximum lengths that cannot be exceeded. The maximum length of a function depends on the number of variables and the length of the code for that function.
- ✓ Smart contracts have a maximum size of 24KB per smart contact. This maximum size depends on the number of all variables and functions within the smart contract as well as the total length of the code.

2.3. CryptoKitties, the Ethereum DApp Game, and Ethereum Owning Mechanism

CryptoKitties is game, developed on 2017 by Dapper Labs, which depends on Ethereum blockchain network. This game mechanism illustrate how much Ethereum is secure via securing the ownership and breading rights of virtual cats to specific Ethereum account address. The virtual cats in the game have inherited attributes mimicking the DNA in the species. The inherited attributes are stored in Ethereum smart contacts where only the contact owner can sell, buy and bread his own virtual cats. This mechanism of owning and trading the virtual cats is the general mechanism on Ethereum to own smart contract. Owning smart contact on Ethereum can mean that you can own car or house if the smart contract represents car or house owning. Also, smart contract owning can mean that you own a version of software or a version of database on Ethereum network (Serada et al., 2020).

Smart contract owning mechanism is the key idea of this study to implement the software of Ethereum databases (which is developed within this study) in future integration applications with ESRI digital maps programming software.

On December 2017, the game increasing popularity showed that 15 transactions per second the capacity of Ethereum 1.0 needs to be highly increased. But the scalability of Ethereum was a main target of Ethereum milestones since it started on 2015. And the scalability is the main target of Ethereum 2.0 to raise the transactions per second to 100,000.

2.4. Ethereum Database Study of Yale University in USA, June 2020[1]

Yale University in USA chose Ethereum blockchain to make study about database for data related to pharmacogenomics. Pharmacogenomics is the science which study the relationship in between genes and medicines. Pharmacogenomics became so important to take the decisions about the medical treatments and especially in the age of Corona virus. Because of this importance, new mechanism should be found for storing and sharing Pharmacogenomics data. The reason Yale university´s study used Ethereum smart contract to store Pharmacogenomics data is the security and high integrity that Ethereum provides for data storing and sharing. For the same reason this thesis research used Ethereum, but with another important implementation which is the applications of ESRI digital maps programming.

The study used mapping tool of Solidity Ethereum smart contracts language. Mapping in Solidity is to make relation between variables, then you can retrieve the information in the style of look up for value by key. Mapping is like dictionary of key-value in the programming languages such as C Sharp and Java.

The target of the study was to test the storage and the query performance to Ethereum database which was designed by mapping tool in Solidity. Taking in considerations the barriers and the limitations of Ethereum and its smart contracts programming language Solidity (Gürsoy et al., 2020).

2.5. European Union SUNFISH Project 2017

This study is trying to explore the possibilities of applying blockchain database at the European Union level. This paper uses mixed research method in trying to determine these possibilities. For the qualitative part of this study, the main issue was a proposal to database structure to solve the problem of how they will achieve 3 main important factors of success. The 3 factors are the security, performance and the economic efficiency of the suggested European blockchain database.

Also, this study tries to find methods to determine the most important independent variables which can be controlled to achieve the above goals (Gaetani et al., 2017).

[1] *Ms. Gamze Gürsoy from Turkey is one of the 3 researchers who made the study*

2.6. Bluzelle the Decentralized Database outside Ethereum

Until now, there is no clear API to use the Bluzelle databse with Ethereum Solidity smart contracts. As in Bluzelle whitepaper, Bluzelle is decentralized database outside Ethereum blockchain and it is cheap decentralized database service comparing with storage on Ethereum blockchain. This database is made to provide cheap storage service for Ethereum applications developers. But to connect the Ethereum Solidity smart contracts to Bluzelle database there are expensive costs for the transactions, which make Bluzelle database not economically efficient. Also, the storage in this database is just in form of simple mapping in between keys and values.

The principles of Bluzelle database are the similar to Proof of Stake PoS mechanism in the blockchain. The users of this database are the developers of Ethereum decentralized applications who will pay in Ether crypto currency for the storage service. The users also can have Ethers if the service provider was not able to provide the required service quality. In the other side, the service providers should stake Ethers before starting to provide the database service. The provider can have more Ethers if he provides higher service quality, but he should stake higher Ethers and he will lose Ethers if he was not able to provide the required service quality (Caldarelli and Ellul, 2021).

2.7. The Blockchain of Marmara University in Turkey

The post-dated cheque is important financial tool in Turkey, but it has the non-redemption problem. So, the university of Marmara developed blockchain system to mimic the peer-to-peer credits similar to post-dated cheque. This blockchain system depends on Komodo blockchain platform solutions in USA. The Marmara university blockchain solution is called Marmara Credit Loops MCL. The MCL depends on blockchain smart contracts to create a type of insurance to avoid the nonpayment of the credit's loans.

The MCL is decentralized finance DeFi system targeted the real economy, and it is started on 2018. The Komodo blockchain which is the infrastructure of MCL is form of smart contract with relation to the Bitcoin blockchain network. MCL is designed not only for post-dated cheque, but also for other solutions such as bills and policies (Çetiner and Lee, 2019).

3. Method

As this research consists of two phases:

- ✓ The first stage is designing the Ethereum databases software.
- ✓ The second stage is proposing applications for the software integration with another software for full programming of ESRI digital maps.

Therefore, the research methods used here in these two stages will differ.

3.1. First Stage: Programming

The aim of the first stage of this paper is to design software that in its turn allows the user to design any database on the Ethereum blockchain network with parent-child relationships. The design of the software will take into account the barriers and limitations of Solidity and Ethereum mentioned in the literature review. Therefore, mixed research methods (Cresswell, 2012) are used in this paper to examine how and how much features can be added to the target software considering the time required to program each feature.

Methods to deal with each barrier and with each limitation of Solidity and Ethereum platform will be discussed (Solidity 0.8.6 documentation, 2021). But before explaining these methods, we will explain the methodology that was applied to get the methods.

3.1.1. Methodology

The Case-Based Reasoning (CBR) model uses inference to solve a problem through experience gained from dealing with a similar problem. CBR generates continuous learning because when a problem is solved, new knowledge will help in solving a similar problem in the future. CBR is an important tool for shaping knowledge management systems. CBR relies on the dynamic relationships between data,

information, and knowledge to help learning new knowledge through the management and reuse of accumulated prior knowledge (Aamodt and Plaza, 1994; Althoff and Weber, 2005; Watson, 2001).

The Learning-by-Doing Principle is based on practical experience to solve the problem. This model links theories and their practical applications. Also, the model is used in the engineering and scientific fields for active learning. The Learning-by-Doing Principle will generate the life cycle between observations, concepts, testing effects and experience. Knowledge systems are the basis for managing complex scientific fields and these systems require sustainability in learning from experience (Reese, 2011; Bot, Gossiaux, Rauch and Tabiou, 2005; Ord, 2012; Aamodt, 1995).

3.1.2. Applying the Methodology

By applying the methodology, five software have been created since February 2019 as steps to create the final Ethereum Databases software. The goal of these five software was to find methods to deal with the limitations and barriers of Solidity and Ethereum.

- ✓ The first software is about document verification on Ethereum. This software was made by using Solidity and JavaScript.
- ✓ The second software is the same as the first software, but it was made using Solidity and Java.
- ✓ The third software is a sample of Ethereum database. It was made by Solidity and Java.
- ✓ The fourth software is an advanced Ethereum database sample using Solidity and C Sharp.
- ✓ The fifth software is about advanced deep searching in databases using Java and SQL.

Figure 1

Documents Verifications on Ethereum by Using Solidity and Javascript

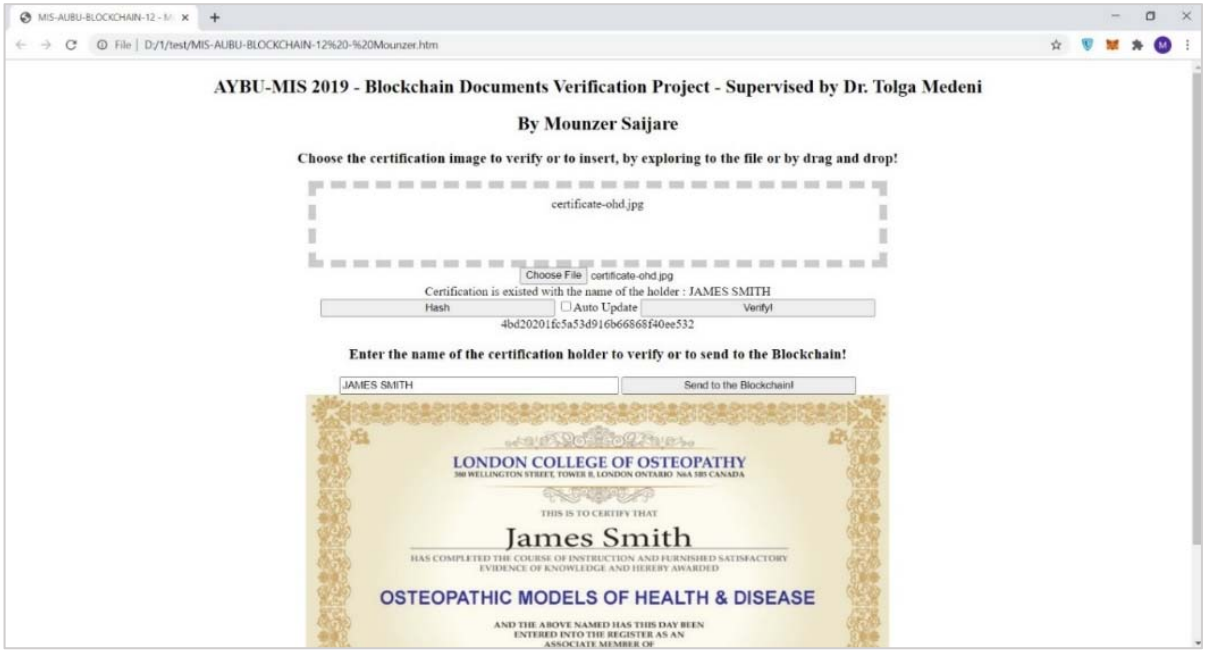


Figure 2

Documents Verifications on Ethereum by Using Solidity and Java.

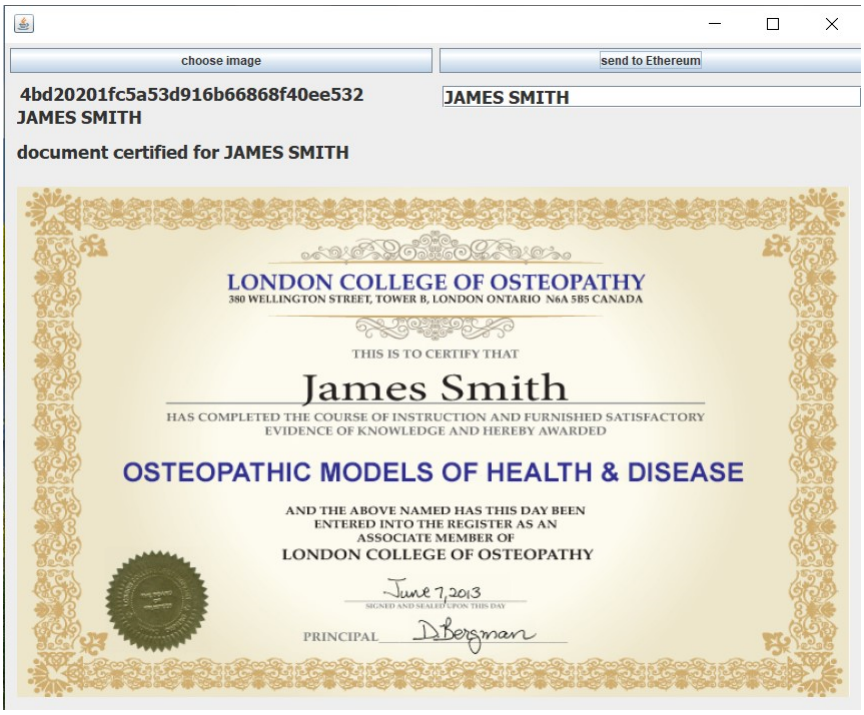


Figure 3

Ethereum Database Sample by Using Solidity and Java.

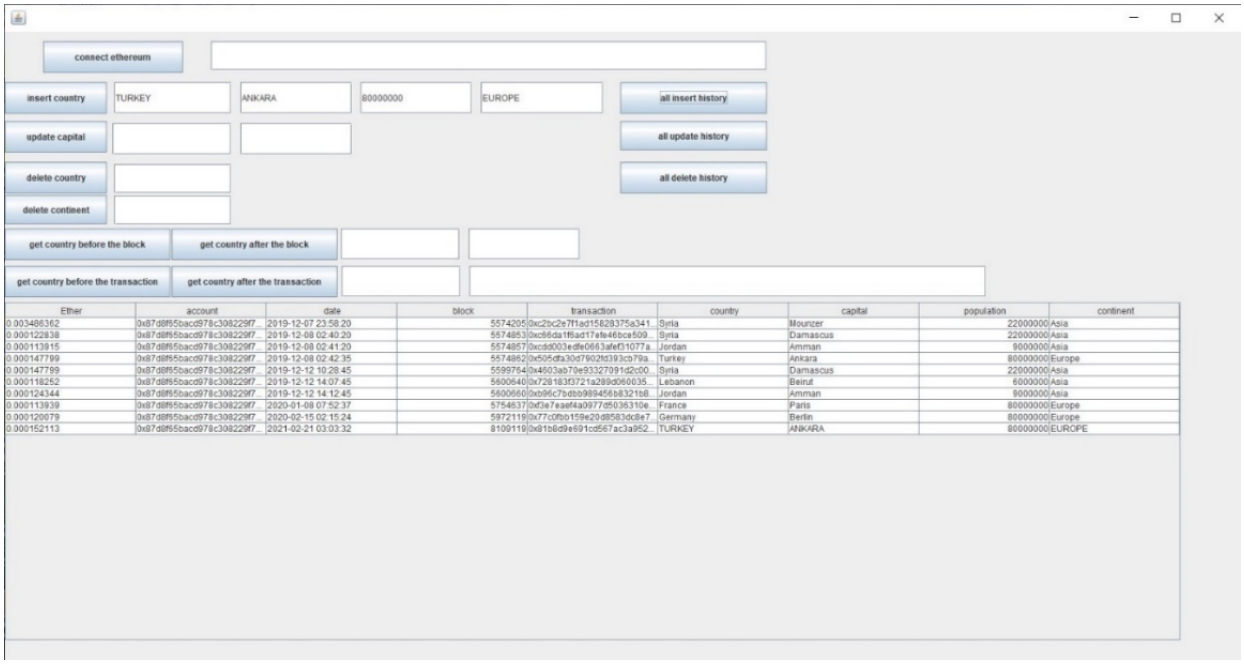


Figure 4

Ethereum Advance Database Sample by Using Solidity and C#

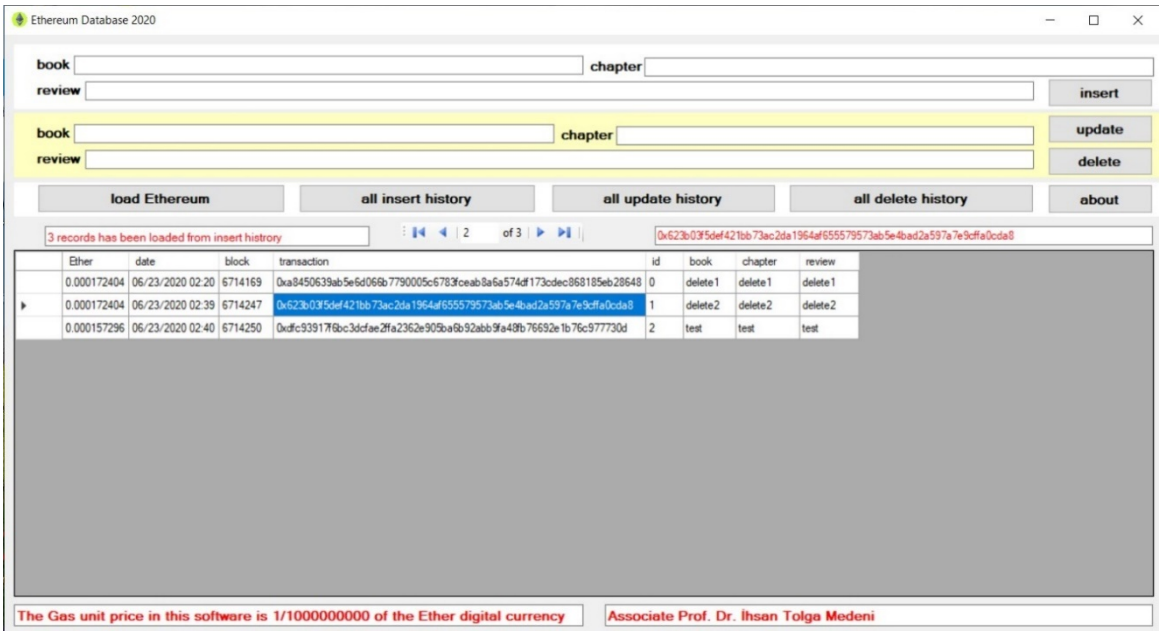
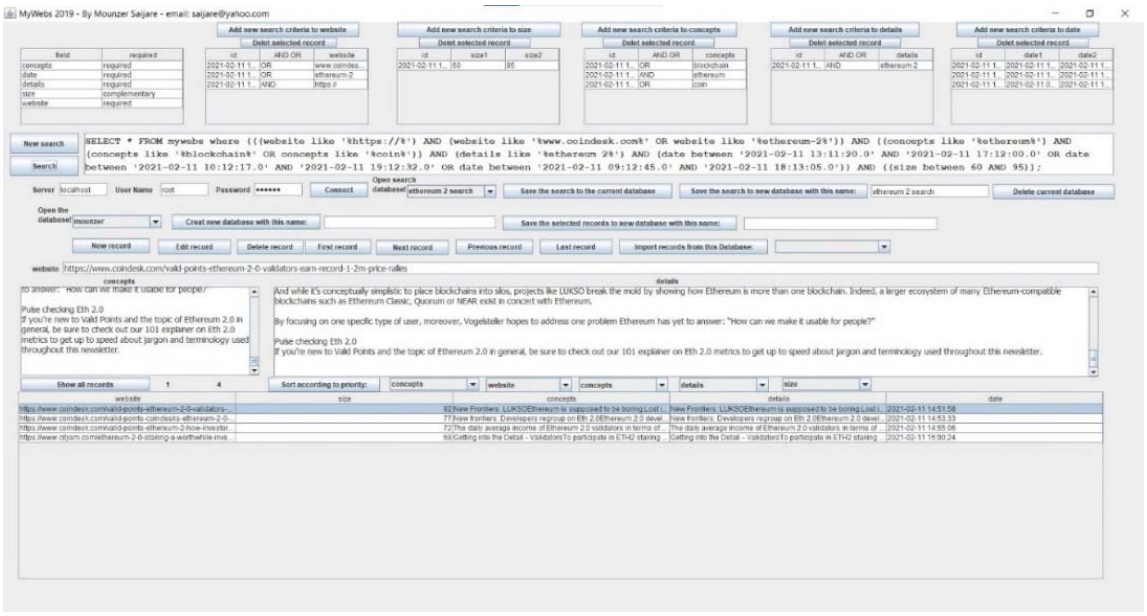


Figure 5

Software about Deep Advanced Search in Databases Using Java and SQL



3.1.3. The Methods Achieved After Applying the Methodology

With the using of the methodology explained above, many methods were concluded from the experience of developing much software (depending on Ethereum smart contracts programming). In the following these methods will be explained.

3.1.3.1. Method for Problem that not all Types of Variables are Supported like the Normal Programming Languages Such as C Sharp and Java

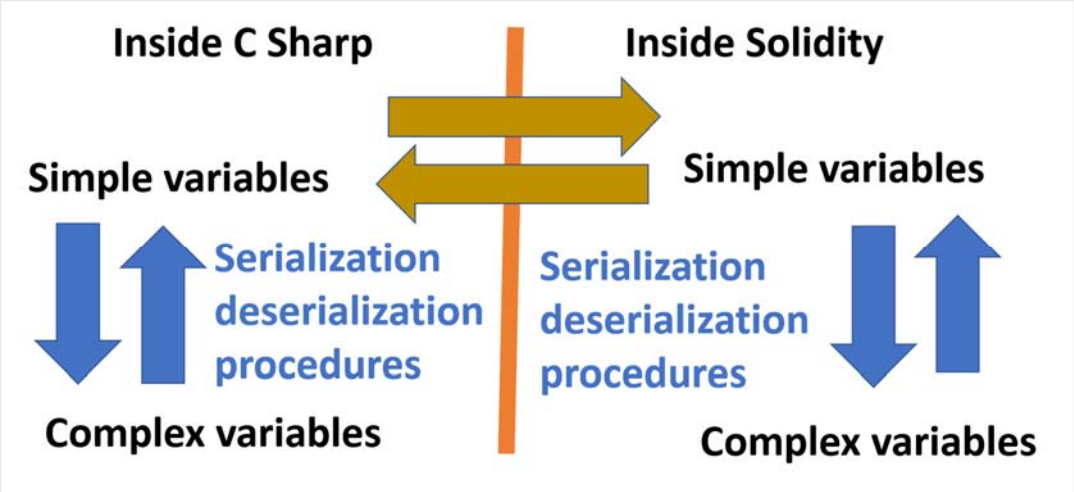
For example, these types are not supported: numeric double and date. This means that not all variables' types of traditional languages (like C Sharp and Java) are available on Solidity and Ethereum virtual machine EVM. Outside Solidity and Ethereum and for primitive variables, there will be no problems as for example there are many tools in C Sharp language to change variables types. But when the variables are being sent to Solidity, they should be sent using suitable types of available Solidity types to make the search and filtering processes available inside Solidity.

3.1.3.2. Method for Problem That Struct Variables cannot be sent to and cannot be received from the Public and the External Functions

The most important thing that struct, class and object variables types are not available for the arguments of the public functions in Solidity. This makes huge problem to transfer complex variables such as collections, arrays, structs and objects. To solve this problem, multi-level and very complex serialization and deserializations were made to transfer the complex data to and from Solidity. Serialization and deserializations were made outside and inside Solidity.

Figure 6

Serialization and Deserialization between C Sharp and Solidity in the Software



3.1.3.3. Method for Problem That Conversions in Between Variables Types in the Most Cases are not possible

In addition to that all types of variables are not available in Solidity, also the conversions between available types are not possible in the most cases. So as mentioned before, suitable types should be chosen outside Solidity to represent the messing types inside Solidity. This will make the search and filtering procedures available inside Solidity.

For example, to send date and double data types from C Sharp to Solidity, we can convert date and double data types to integer as integer data type is available in Solidity. So, we can later search, filter and compare these data types inside Solidity.

3.1.3.4. Method for Problem That Functions Have Maximum Lengths Which Cannot be exceeded. The Maximum Length of Function Depends on the Number of Variables and on the Code Length of This Function

This means that the variables number and the code in one function have specific limits. To solve this problem, the variables declarations inside the function should be reduced as much as possible although this may make the code longer and difficult to read. Also, to reduce the code length in a function, part of the code could be transferred to another functions. In addition to, the good construction of the structs variables can reduce the code in the functions to a very good degree.

3.1.3.5. Method for Problem That the Smart Contracts Have Maximum Size of 24 KB of Each Smart Contract. This Maximum Size Depends on the Number of All Variables and Functions inside the Smart Contract in Addition to the Total Code Length

For specific project with Solidity, maybe 24 KB will be enough for the smart contract. But when more features are needed in the project, this means more and more growing number of functions and code length which one day will exceed the 24 KB.

Inside the same contract, many procedures can be done to reduce the code size. The most important thing is to reduce the number of functions needed by studying the best structure for the structs and for the functions. Also, the relations in between the functions should be studied to reduce the number of functions.

When the contract starts to exceed the maximum size of 24 KB regardless of all Solidity good programming practices, then the library smart contracts will be the solutions.

Library contract is deployed contract on Ethereum network whose functions can be called from another deployed contracts on Ethereum network. Library contracts are executed within the context of the contract that call them. With library smart contracts, the Solidity smart contract can have access to codes more than 24 KB in size.

3.1.3.6. Method for Using Recursion

Recursion on function programming is to use the function inside itself. This will cause endless execution for the function, so a condition is set before reusing the function inside itself.

Recursion on structs and models' structures is to use the struct or the model inside itself. This can lead to a tree of parent-child relationships in between models, structs and object. Recursion is used in the software here to great the parent-child relationships in between database tables.

3.2 The Second Stage: A Survey for Recommendations

The objective of the second stage is surveying recommendations for integration applications of Ethereum database software with ESRI digital map programming software in real estate business and in the transparent environmental management projects (with a specific example on Mediterranean pollution monitoring).

We have seen the methods of the first stage and for the second stage the methods will be survey. For the survey method, the research survey (questionnaire) will be applied to specific people with specific backgrounds. The surveying will be applied after the following steps:

- ✓ Design and explanation of the Ethereum databases software.
- ✓ Explanation of the ESRI digital map programming software.

The aim of the survey will be to get indications of how and to what degree we can apply the integration of the two software in two types of projects:

- ✓ Real estate business automation projects.

- ✓ Transparent environmental management with a specific example on pollution control of the Mediterranean Sea.

In the appendix is the AYBU university Ethic Committee's decision number 16.04.2021-81 about the acceptance of the survey.

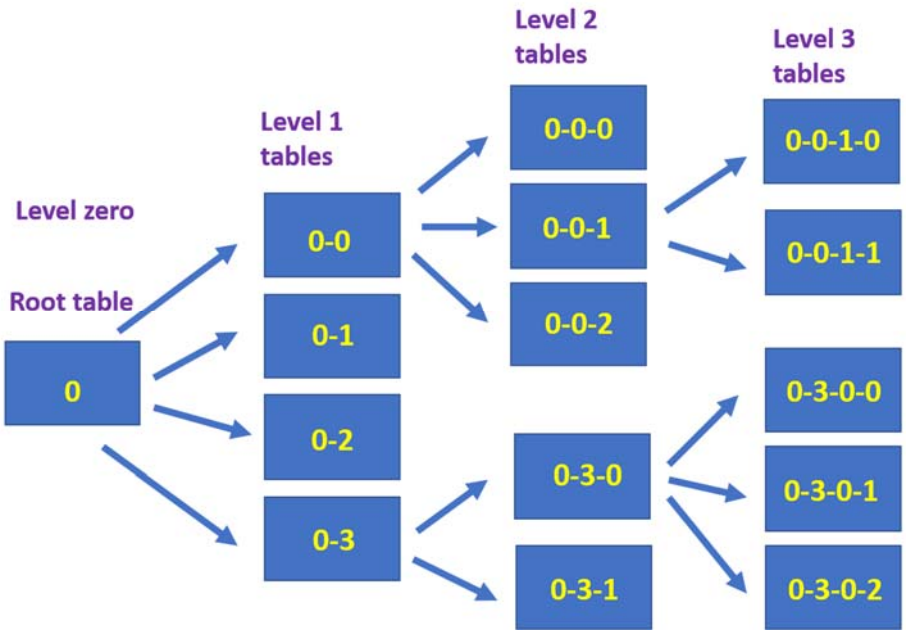
4. Results

4.1. Ethereum Database Software (Research Question One)

Using recursion in functions and in struct variables, the following structure was achieved for the Ethereum databases that are designed by the software:

Figure 7

The Structure of the Ethereum Databases in the Program



As shown in Figure 7, the tree represents the parent-child relationships in between the database tables. In any database designed by the program, two tables at the same level in the tree with the same name cannot have children with the same names. Therefore, a single table in the database is identified by its name, its parent table, and its level.

As mentioned earlier, every storing or updating of data on the Ethereum network needs gas units to be spent from the sender's Ethereum account. These actions are called transactions. Before submitting the transaction, the gas unit price for each job must be determined. The price of a unit of gas can range between 1 and 22 one of billion Ether (the currency of Ethereum). Depending on the gas price, the possibility and speed of the transaction will be determined. In the software, the price of gas is defined as one billionth of ether for one gas unit.

Three versions of the software have been achieved:

- ✓ Version for Android with MS Xamarin and C Sharp. This version is also ready for iPhone, because the same MS Xamarin project for this software can be deployed on Android and iPhone operating systems at the same time.
- ✓ Version for Windows using MS Visual Studio 2019 and C Sharp. MS Xamarin and C sharp can be used later for a future Macintosh OS release.

The Android and the Windows versions for Ethereum Database Part 1 in addition to full explanations about databases management by using the software on Android and on Windows operating systems can be found here in this Google Drive link:

https://drive.google.com/drive/folders/12D6-QeJhB0NttagASG3RRK6Z50oC5QRL?usp=sharing

The transactions can be followed here:

https://rinkeby.etherscan.io/address/0x0d277d12e8be26b136cc4795f92dda7fc93aa62c

The features that have been added to the program within this research are:

- ✓ Structure of the parent-child relationships for tables in a database.
- ✓ Design fields for a table with string data type only.
- ✓ Create, read, update and delete CRUD actions for any table and for any record.
- ✓ For table update procedure, only table name update is allowed to protect data integrity.
- ✓ The table is defined in the parent-child relationship tree in the database by its name, parent table, and level. Two tables in the same database cannot have the same three attributes.
- ✓ Errors related to user inputs are managed within solidity. This makes the Ethereum database does not accept any transaction until the input data is valid from the user. Also, this will enable the Ethereum databases to send errors messages to the users, which will make the interaction with the databases the same as with C Sharp, Java or with other languages. The developer who interacts with the Ethereum databases can see first if the data is valid or not before sending the transaction.
- ✓ Record events for all database activities at the database level and at the table level.

Table 2

The Benchmark of the Thesis Software with the Others Decentralized Databases

Database	Ethereum Databases software	Bluzelle databases	CryptoKitties, DApp game	USA Yale university database	EU SUNFISH project	Marmara university Turkey
Blockchain	Ethereum	Ethereum and another decentralized network	Ethereum	Ethereum	Special blockchain network	Komodo Platform USA
Storage mechanism	Tables with mother children relationship	Simple mapping	Smart contract normal storage	Simple mapping	Not specified	Special smart contracts
Storage cost	Ethereum prices	Ethereum prices and other costs	Ethereum prices	Ethereum prices	Not specified	Not specified
Table CRUD	Yes	No	No	No	Not specified	No
Record CRUD	Yes	No	No	No	Not specified	No

4.2. Case Study (Research Question One)

The case study will show the relationship between the research software and real application. Therefore, this case study is conducted to show the great importance of our software.

Documents verification in all areas will be one of the important applications of the software. For example, universities can use the software to publish their students' certifications on the Ethereum blockchain. But since storage on the Ethereum blockchain would be very expensive, instead of storing certificates files on the Ethereum network, MD5 hash codes for the files will be stored. And then any change in the files will change the MD5 hash codes drastically.

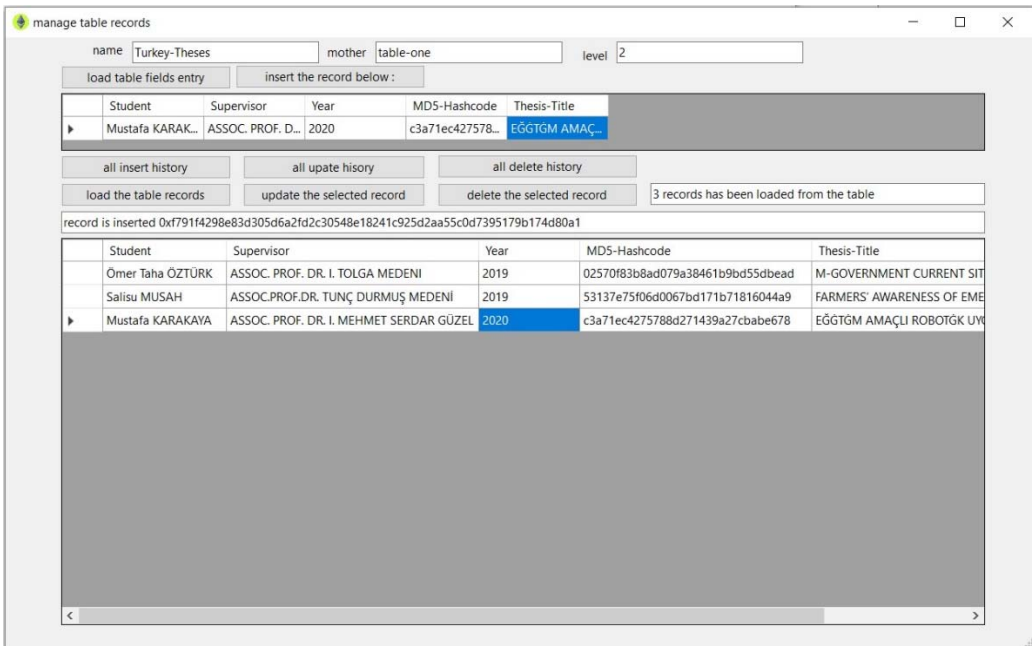
The case study is about YÖK TEZ MERKEZİ (The Databases of the Turkish National Thesis Center of the Council of Higher Education). A table named TURKEY-THESES has been created on the Ethereum database. 3 records have been added to this table containing information about 3 theses. The information includes MD5 hash codes for the PDF theses files. The Ethereum blockchain will now ensure that the three theses PDF files are verified by storing their MD5 hash codes. MD5 hash codes for a PDF file can be calculated online using one of these two websites: http://onlinemd5.com/ https://emn178.github.io/online-tools/md5_checksum.html

Hence these MD5 hash tokens can be compared to the MD5 hash tokens stored in the Ethereum database. The Ethereum database can be managed by the software of Ethereum databases.

The following Figure 8 illustrate this case study by a screenshot from the software of Ethereum databases:

Figure 8

Case Study for the Ethereum Databases Software



4.3. The Survey for the Software Recommendations for Sustainable Development with Software for ESRI Digital Map Programming (Research Question Two)

As the survey contains the software applications and another software for ESRI digital maps programming, so we should explain these issues before presenting the survey at the end of this results parts as following.

4.3.1. The Software Possible Applications for Sustainable Development

The program will design databases that will store data on the Ethereum blockchain network. Storing data on Ethereum is very expensive, so Ethereum databases are not alternatives to the enterprise databases that run on centralized servers and clusters like Oracle, MySQL, and SQL Server. But Ethereum databases will be high-level solutions for critical projects that need a high level of security, integrity and transparency for data storage and sharing. There are plenty of types for critical projects that need the security, integrity, and transparency of Ethereum databases to store and share data. The program can be used for:

- ✓ One of the most important applications of the Ethereum database software will be the business of the Ethereum NFT. With the upcoming Ethereum 2 network expected at the beginning of 2022, the

program could move the NFT business to be just like e-commerce today especially after completing Part 2 of the software in a future PhD study soon.

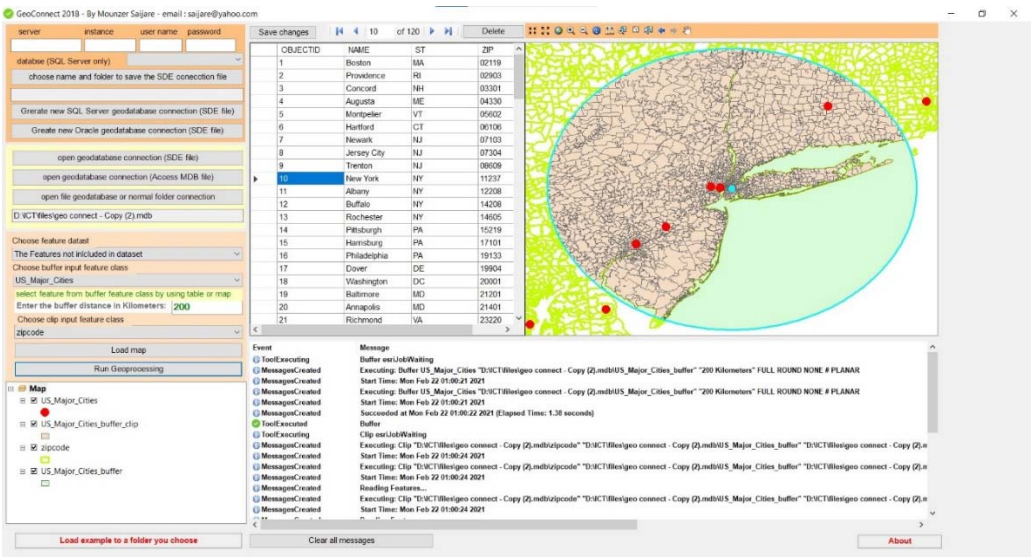
- ✓ Political applications of the transparent management, strategies and electronic voting that will lead to democracy.
- ✓ Commercial applications for secure and transparent management, in addition to the fact that smart contacts will replace paper contracts and eliminate the need for a third party for direct contracting between two parties.
- ✓ Verification of documents in all areas such as academic certificates, real estate documents, and digital map documents.
- ✓ Safe and transparent management of business, such as real estate business.
- ✓ Transparent management and environmental protection of the natural resources.

4.3.2. Software for ESRI Digital Maps Programming

We have developed software that shows how the full programming of ESRI digital maps can be achieved by programming of any sequence from the ESRI 200 geoprocessing tools (Saijare et al., 2020). Geoprocessing tool is a function that has inputs and outputs as digital map layers and can fulfill a specific task of modifying the input digital map layers. This digital map programming software starts from making the connection to the geodatabase. A geodatabase is the same as a regular database but contains spatial reference data as well as other data for describing shapes in digital map layers. Also, the procedure for connecting to the geodatabase in the software uses the programming of a specific geoprocessing tool whose output is as a file describing the connection. This is required for Oracle and SQL Server enterprise databases. Other MS Access geodatabases and folders geodatabases containing XML files can be opened directly. The second step in this software is to open the geodatabase, load the input digital map layers and give any additional input information for the desired sequence of geoprocessing tools. At this stage, the program performs a very important process, which is to extract the data of the digital map layers and put it in a database format. The great importance of data extraction that this makes any full automation of digital maps project to be possible. Automation can be calculations to certify digital map information to specific national or international standards. Also, extracting digital map information in database format will make it available for any search and filtering processes. The final step in this digital map software is to calculate the sequences required for geoprocessing tools and load the digital map layers for input and output. A full description of this ESRI digital maps programming software can be found in our paper "Digital Maps and Blockchain, Simplification of Information Sharing", see the references.

Figure 9

ESRI Digital Map Programming Software

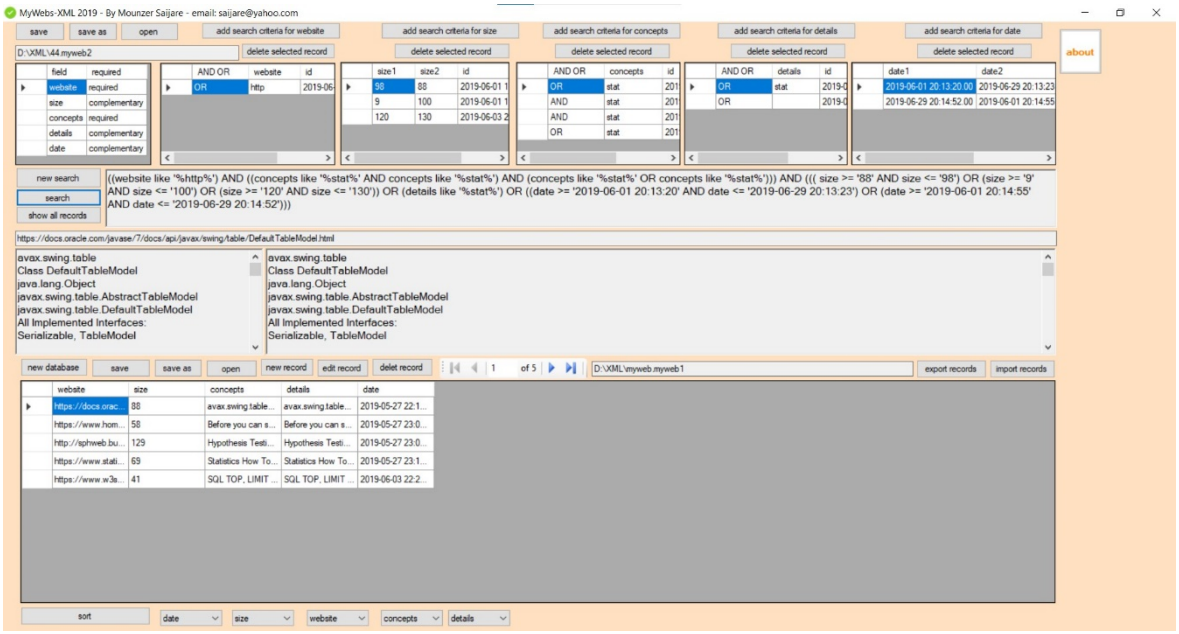


As in Figure 9, using the same methodology as the digital map programming software, any digital maps project can be automated. This methodology can be taught at the university, in the public sector, in the private sector, in the organization and in UN projects. National or international standards can be easily integrated with this software. Therefore, an integrated system can be created from data collection tools to data processing and programming according to national or international standards. And finally, visualization can be done on digital maps with other programming on digital map layers with any choice of consequences of the 200 ESRI geoprocessing tools. With this integrated system, decision makers can have powerful tools that support quick and correct actions and decisions.

As in Figure 10, there is another software developed within the master about deep databases searching which produces a very complex SQL statement. Deep search software can be combined with digital map programming software to search deep in geographical databases. As we can see below the user interface of this deep search software:

Figure 10

Software for Deep Search in Databases



In the following pages we will see survey about the integration of the software for ESRI digital maps programming with the software for Ethereum databases in two areas:

- ✓ Real estate business
- ✓ Transparent environmental management with a special case to protect the Mediterranean from pollution

4.3.3. The Survey for Sustainable Development Recommendations for Ethereum Databases Software with the Software for ESRI Digital Maps Programming

Now after the explanation of the two software we can make survey about the two suggested applications:

- ✓ Real estate business.
- ✓ Environmental management projects, with the case of Mediterranean Sea protection projects related to United Nations

We designed the survey with these four questions with four options answers for each one (EU Science Hub, 2021; Hjort et al., 2019):

Table 3

The Survey of the Two Software

Question one: The software for Ethereum databases can make any database on Ethereum network which will be stored on about 10000 server computers of Ethereum network around the world. So, these databases will have huge security and huge transparency. Files on computers have hash codes exactly like the human fingerprints and storing these files hash codes on Ethereum databases can replace the needs for the paper based important documents (such like paper business contracts, paper universities certifications, paper real estate documents). Also, storing files hash codes on Ethereum databases can prevent any changes on the original very important files (for example: certifications files, contracts files, the digital maps files of the real estate projects, the digital maps files of Mediterranean Sea pollution monitoring projects).

With the same methodology of the software for Ethereum databases and its security and transparency, do you think that we can apply the above-mentioned solutions?

Write here the number of your chosen answer from below: [ ]

1- It is very good idea because it will reduce the needs for paper certifications.
2- We should make pilot projects first, because Ethereum smart contracts is new technology.
3- It is early to apply Ethereum smart contacts technology.
4- Other. Please write here:

Question two: We have another software for ESRI digital maps programming and with the same methodology of this software we can make full automation for any digital maps project.

With the same methodology of this software, do you think that we can apply full automation for digital maps projects in the universities or in the public sectors or in the private sectors?

Write here the number of your chosen answer from below: [ ]

1- It is very good idea because it will make the projects to be significantly automated.
2- Programing of ESRI digital maps is difficult to be learned.
3- We should make pilot projects first to see the difficulties in teaching the programming of ESRI digital maps.
4- Other. Please write here:

Question three: Do you think that we can use the same methodology of the integration of the software for Ethereum databases and the software for ESRI digital maps programming in projects for the automation of the real estate business?

Write here the number of your chosen answer from below: [ ]

1- It is very good idea because this will join the governmental real state documents and the paying methods in the same software.
2- We should make pilot projects first, because Ethereum smart contracts is new technology.
3- It is early to apply Ethereum smart contacts technology.
4- Other. Please write here:

Question four: Do you think that we can use the same methodology of the integration of the software for Ethereum databases and the software for ESRI digital maps programming in United Nations projects for the protection of the Mediterranean Sea from pollution?

Write here the number of your chosen answer from below: [ ]

1- It is very good idea because this will show the digital maps of pollution in a very transparent method for all the Mediterranean countries.
2- We should make pilot projects first, because Ethereum smart contracts is new technology.
3- It is early to apply Ethereum smart contacts technology.
4- Other. Please write here:

These four questions were for the survey and the following are the results of the survey.

4.3.4. The Results of the Survey

Twenty-two female and male participants responded to the research survey. Participants' backgrounds include academic experts, Ethereum blockchain companies, sustainable development experts, lawyers, Ethereum investors, and university students associated with the Ethereum blockchain.

Regarding the first question, 36.4% (8/22) of the respondents answered that it is very good to implement the software for Ethereum databases as it will reduce the need for paper certificates. This percentage of participants is good and represents the growing awareness of the growing importance of Ethereum databases.

The largest percentage is 54.5% (12/22) for the first question option 2, about creating demo projects to explore the possibilities of Ethereum database applications. The reason is that the Ethereum blockchain is a new technology and needs a lot of research and studies on its implementation areas.

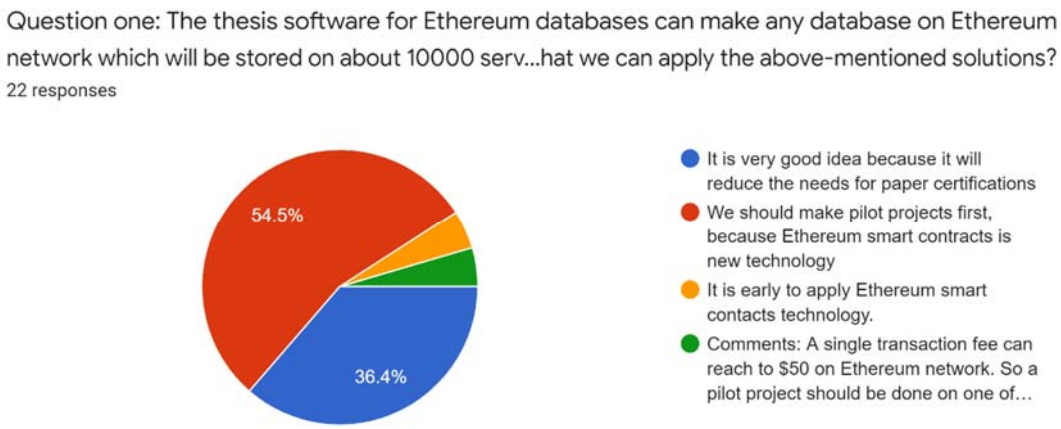
4.5% (1/22) of respondents said that we cannot implement Ethereum databases now because it is still a very new technology.

Finally on the first question, 4.5% (1/22) of respondents from Blok-Z Company are concerned about the cost of storage on the Ethereum Blockchain. Blok-Z is a company located in Berlin, Germany and Ankara Turkey working to promote the trading of clean and eco-friendly green energy using the Ethereum blockchain technology. Here is the exact answer of the participant from Blok-Z: "A single transaction fee can reach to $50 on Ethereum network. So, a pilot project should be done on one of the official test nets (like Robsten and Rinkeby). Classical server solutions can be considered to store the data to be verified on blockchain instead of storing on blockchain". The answers to Blok-Z concerns are discussed in the thesis report to keep this paper in reasonable size.

The following chart shows the results of the first question:

Figure 11

The Result of the First Question of the Survey



For the second question, we can see that 36.4% (8/22) of the respondents answered that programming of ESRI digital maps will automate digital map projects.

While 18.2% (4/22) of respondents said that learning to program ESRI digital maps is very difficult.

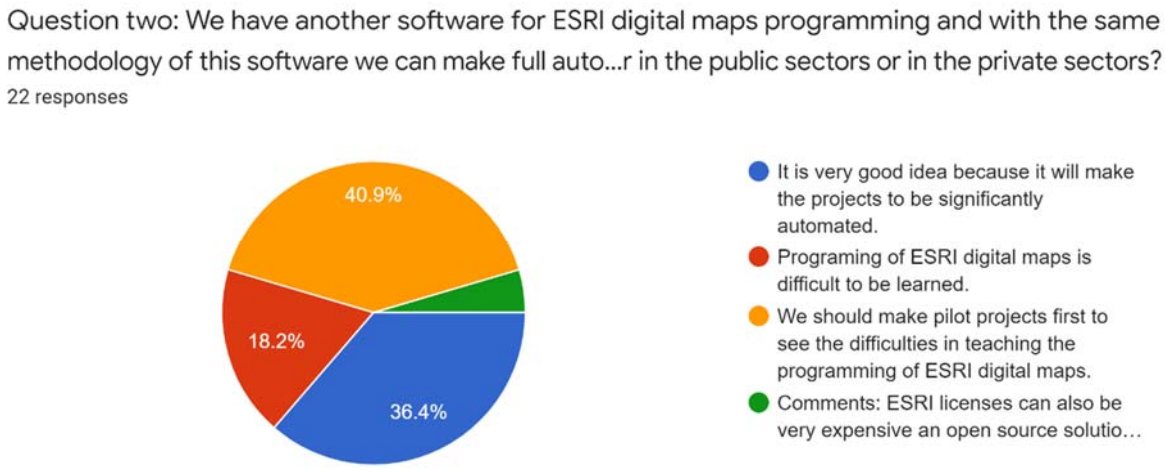
The largest proportion of respondents 40.9% (9/22) said that a pilot project should be done to find out the difficulties that will arise while teaching ESRI digital map programming in universities, in the public sector or in the private sector.

Finally, 4.5% (1/22) of the participants from Blok-Z are concerned about the high cost of ESRI software and exactly said: "ESRI licenses can also be very expensive an open-source solution can be considered like QGIS. Otherwise, the business model would be highly dependent on ESRI". These concerns are discussed in the thesis report to keep this paper in reasonable size.

The following chart shows the results of the second question:

Figure 12

The Result of the Second Question of the Survey



As for the third question, 40.9% (9/22) of respondents said it would be a good idea to use the integration of the two software to automate real estate business because it would join government documents of the real estate and the methods for payment in the same software on the Ethereum network.

The largest percentage of respondents 45.5% (10/22) said that we should do a pilot project and evaluate it, because Ethereum smart contracts are a very new technology.

4.5% (1/22) of respondents answered that it is too early to implement Ethereum smart contract technology.

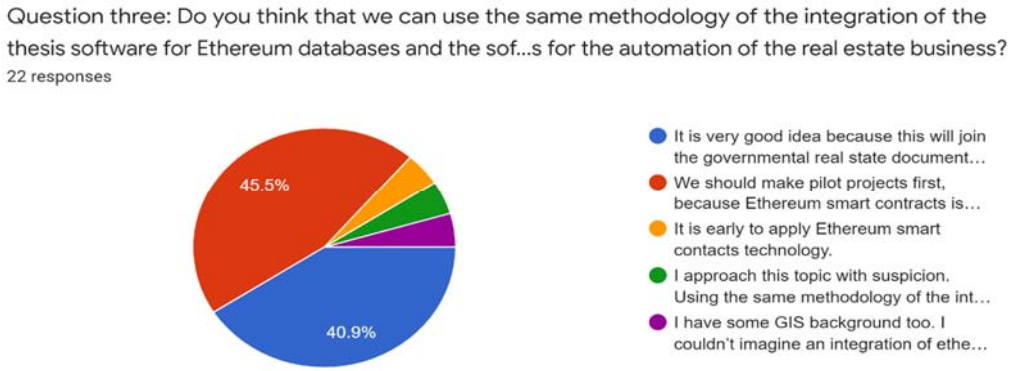
Participants from Blok-Z, which has a percentage of 4.5% (1/22), expressed concern about the integration between Ethereum databases software and ESRI digital mapping programming software: "I have some GIS background too. I couldn't imagine an integration of Ethereum and ESRI. Some more details should be given. I would say it definitely needs some proof of concepts". These concerns are discussed in detail on the thesis report to keep this paper in reasonable size.

Finally, 4.5% (1/22) of respondents, a lawyer who works on digital money laws, said that she has concerns about customer rights when we apply the integration of the two software: "I approach this topic with suspicion. Using the same methodology of the integration of the software for Ethereum databases and the software for ESRI digital maps programming in projects for the automation of the real estate business can lead to loss of rights of customers". These concerns regarding customer security and rights are explained in detail in the thesis report to keep this paper in reasonable size.

The following chart shows the results of the third question:

Figure 13

The Result of the Third Question of the Survey



For the fourth question, the largest proportion in all of this survey was 54.5% (12/22) of the respondents who said that the integration of the two software would be very beneficial for transparent environmental management of the Mediterranean Sea. Therefore, digital maps of pollution can be presented in transparent ways to all Mediterranean countries. We can point out from this high percentage that Ethereum databases and ESRI digital map programming will be very important for international and national environmental management projects.

31.8% (7/22) of respondents said that we should build a pilot project because Ethereum smart contracts are a very new technology.

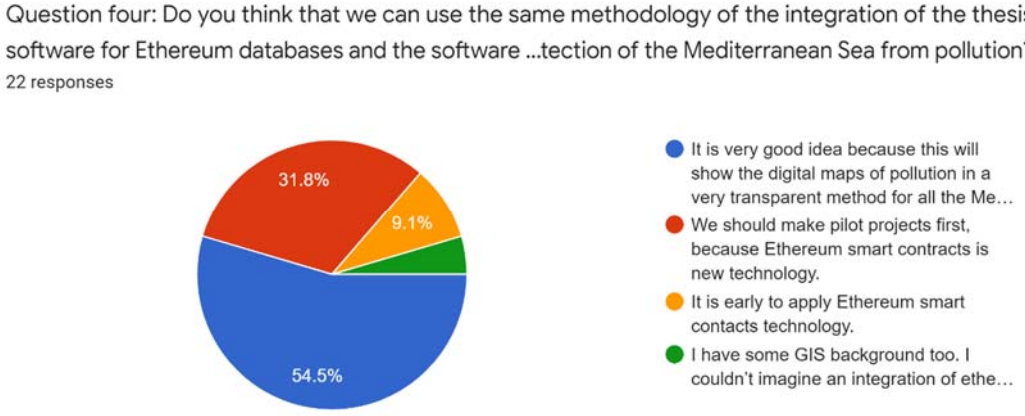
9.1% (2/22) of respondents answered that it is now too early to implement Ethereum smart contract technology.

Finally, for the fourth question, 4.5% (1/22) of respondents from Blok-Z Company gave the same concerns in the third question about the integration capabilities of Ethereum databases software and ESRI digital map programming software. These concerns are answered in the thesis report to keep this paper in reasonable size.

The following chart shows the results of the fourth question:

Figure 14

The Result of the Fourth Question of the Survey



4.3.5. The Conclusion of the Survey

The general results of the survey were positive about the integration of the 2 software to achieve transparent management. However, the participant saw that this transparent management will be more necessary in the global environmental projects to protect our planet from the pollution. The highly importance of the recommended applications made the participant to see there are no problems about the high costs of these 2 new technologies (Ethereum smart contact and ESRI digital maps programming). The survey gave good indicators about the future of these recommended applications. To keep this paper in reasonable size only the survey is included, but in the thesis report the two recommended applications are explained.

5. Conclusion

Ethereum Blockchain is new powerful technology. And this study concentrated on important infrastructure issue for any IT project, which is the databases. So, the concept of the study was to develop software, with Solidity smart contracts, C Sharp, Java and JavaScript, to design any Ethereum database with parent-child relationships in between the tables. After designing the first step of the software, a survey for recommendations for sustainable development applications with the second software of ESRI digital map programming were explained.

The second step of the software of Ethereum databases will be continued as soon as possible to add more professional features exactly like the features in the enterprise databases.

Compliance with Ethical Standards

Conflict of Interest: The authors declare that there is no conflict of interest.

Ethics Committee Permission: The ethical suitability of this research is approved by Ankara Yıldırım Beyazıt University Ethics Committee (31.08.2021 -2021-229).

Authors Contribution Rate Statement: The authors declare that they have contributed equally to the article.

Financial Support: No

References

Aamodt, A. (1995). Knowledge acquisition and learning by experience–the role of case-specific knowledge. *Machine learning and knowledge acquisition*, 8, 197-245.

Aamodt, A., & Plaza, E. (1994). *Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI communications*, 7(1), 39-59.

Althoff, K. D., & Weber, R. O. (2005). Knowledge management in case-based reasoning. *The Knowledge Engineering Review*, 20(3), 305-310.

Bot, L., Gossiaux, P. B., Rauch, C. P., & Tabiou, S. (2005). 'Learning by doing': a teaching method for active learning in scientific graduate education. *European journal of engineering education*, 30(1), 105-119.

Caldarelli, G., & Ellul, J. (2021). The Blockchain Oracle Problem in Decentralized Finance-A Multivocal Approach.

Çetiner, B. G., & Lee, J. Marmara Kredi Döngüleri: Vadeli Çeklerdeki Karşılıksızlık Sorununa Blokzinciri Çözümü.

Cresswell, J. W. (2012). Planning, conducting, and evaluating quantitative and qualitative research. *Educational Research*. https://doi. org/10.4135/9781483349435

Esri: GIS Mapping Software, Spatial Data Analytics & Location Platform. (2019). Esri.com. https://www.esri.com/en-us/home

Ethereum 2.0 (Eth2). (2021). Ethereum.org. https://ethereum.org/en/eth2/

Ethereum Foundation. (2015). Olympic: Frontier Pre-Release. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2015/05/09/olympic-frontier-pre-release/

Ethereum Foundation. (2015). The Ethereum Launch Process. Blog.ethereum.org. https://blog.ethereum.org/2015/03/03/ethereum-launch-process/

Ethereum Foundation. (2015.). Ethereum Protocol Update 1. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2015/08/04/ethereum-protocol-update-1/

Ethereum Foundation. (2016). CRITICAL UPDATE Re: DAO Vulnerability. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2016/06/17/critical-update-re-dao-vulnerability

Ethereum Foundation. (2016). FAQ: Upcoming Ethereum Hard Fork. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2016/10/18/faq-upcoming-ethereum-hard-fork/

Ethereum Foundation. (2016). Hard Fork No. 4: Spurious Dragon. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2016/11/18/hard-fork-no-4-spurious-dragon

Ethereum Foundation. (2016). Homestead Release. Ethereum.org. https://blog.ethereum.org/2016/02/29/homestead-release/

Ethereum Foundation. (2017). Byzantium HF Announcement. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2017/10/12/byzantium-hf-announcement/

Ethereum Foundation. (2019). Ethereum Constantinople/St. Petersburg Upgrade Announcement. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2019/02/22/ethereum-constantinople-st-petersburg-upgrade-announcement/.

Ethereum Foundation. (2019). Ethereum Istanbul Upgrade Announcement. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2019/11/20/ethereum-istanbul-upgrade-announcement/.

Ethereum Foundation. (2019). Security Alert: Ethereum Constantinople Postponement. Blog.ethereum.org. Retrieved November 21, 2021, from https://blog.ethereum.org/2019/01/15/security-alert-ethereum-constantinople-postponement/.

EU Science Hub - European Commission. (2021, June 23). *Sustainable production: Best available techniques*. https://ec.europa.eu/jrc/en/research-topic/sustainable-production-best-available-techniques.

Fitzpatrick, S. M., & McKeon, S. (2020). Banking on stone money: ancient antecedents to bitcoin. *Economic Anthropology*, 7(1), 7-21.

Gaetani, E., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A., & Sassone, V. (2017). Blockchain-based database to ensure data integrity in cloud computing environments.

Gürsoy, G., Brannon, C. M., & Gerstein, M. (2020). Using Ethereum blockchain to store and query pharmacogenomics data via smart contracts. *BMC Medical Genomics*, 13, 1-11.

Hjort, M., Skobelev, D., Almgren, R., Guseva, T., & Koh, T. (2019). Best Available Techniques and Sustainable Development Goals. *International Multidisciplinary Scientific GeoConference: SGEM*, 19(4.2), 185-192.

Hollander, L. (2020, May 4). The History of Ethereum Hard Forks. MyCrypto. https://medium.com/mycrypto/the-history-of-ethereum-hard-forks-6a6dae76d56f.

Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. (2016). Bitcoin and cryptocurrency technologies: a comprehensive introduction. *Princeton University Press*.

Ord, J. (2012). John Dewey and Experiential Learning: Developing the theory of youth work. *Youth & Policy*, 108(1), 55-72.

Reese, H. W. (2011). The learning-by-doing principle. *Behavioral development bulletin*, 17(1), 1.

Saijare, M., Medeni, T.D., Medeni, I.T., & Soylu, D. (2020). Digital Maps and Blockchain, Simplification of Information Sharing. IVUS.

Serada, A., Sihvonen, T., & Harviainen, J. T. (2020). CryptoKitties and the new ludic economy: how blockchain introduces value, ownership, and scarcity in digital gaming. *Games and Culture*, 1555412019898305.

Sherman, A., Javani, F., Zhang, H., & Golaszewski, E. (2019). On the Origins and Variations of Blockchain Technologies. IEEE Security & Privacy, 17, 72-77.

Solidity — Solidity 0.8.6 documentation. (2021). Docs.soliditylang.org. https://docs.soliditylang.org/en/v0.8.6/.

Tapscott, D., & Tapscott, A. (2018). *Blockchain revolution*. Portfolio: Reprint edition.

The Beacon Chain Ethereum 2.0 explainer you need to read first. (2020, May 23). Ethos.dev. https://ethos.dev/beacon-chain/.

Watson, I. (2001, May). Knowledge Management and Case-Based Reasoning: A Perfect Match?. *In FLAIRS Conference* (Vol. 2001, pp. 118-122).

Wikipedia Contributors. (2019, May 7). Advanced Encryption Standard. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Wikipedia Contributors. (2019, October 14). Cryptographic hash function. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Cryptographic_hash_function.

Wikipedia Contributors. (2019, October 23). Cryptography. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Cryptography.

Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151(2014), 1-32.