​**ARTIFICIAL INTELLIGENCE-BASED BUSINESS STRATEGY FOR OPTIMIZED ADVERTISING**

**2021-152**

A picture containing text, clipart

Description automatically generated

**Individual Final Project Thesis**

**Thathsarani W C**

**IT18158564**

**B.Sc. (Hons) Degree in Information Technology**

**Specialization: Information Technology**

**Department of Information Technology**

**Sri Lanka Institute of Information Technology**

**Submitted on the 13th of October 2021**

October 2021

# Declaration of Originality

I declare that this is my own work, and that this Research thesis does not contain any material previously submitted for a degree or diploma at any other university or institute of higher learning without acknowledgment, and that it does not contain any material previously published or written by another person, except where acknowledgement is made in the text.

|  |  |  |
| --- | --- | --- |
| Name | IT Number | Signature |
| Thathsarani W C | IT1815854 | A picture containing text  Description automatically generated |

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

…………………..............      …………………..............

Signature of the supervisor          Research Project Coordinator, SLIIT

(Ms. Dinuka Wijendra) (Ms. Jenny Kishara)

Date: 10/13/2021

# ABSTRACT

Following the introduction of Bitcoin, a slew of computer science and electronic currency projects based on blockchain technology arose. Since the blockchain authorize contributors of a distributed system to concur dependably of a shared image of the system and monitor system commute, the promise of decentralized systems has piqued attention. To ensure that the truthful parties receive fair compensation from the decentralized blockchain. Blockchain has already been linked to digital currencies, most notably Bitcoin. Blockchain applications are now being investigated in a variety of sectors as a safe and cost-effective means to construct and operate a distributed database and preserve records for digital transactions of all types. However, there are certain unanswered questions: "How will users' and advertising firms' privacy be protected?" Because the blockchain works as a peer-to-peer (P2P) network and all data is updated in real-time, it allows users and advertising firms to securely update and verify data. As a result, the anticipated system will be cryptographically secure. "How can we safeguard the system?" Etc. This study attempts to solve the challenges of safeguarding marketers' and users' privacy.

# ACKNOWLEDGEMENT

I'd want to offer my heartfelt appreciation to Ms Dinuka Wijendra, my supervisor, for her honesty and encouragement, which I will never forget. Ms. Dinuka has been an inspiration to me as I've struggled through this study. Our course coordinator, Mr. Jagath Wickramarathne, deserves special recognition for his invaluable assistance in delivering course materials and sharing his vast knowledge of the module. This thesis would not have been feasible without the help of the team members, who guided me through the research process from the beginning, allowing me to get a better grasp of the subject. I am grateful for the wonderful chances they provided for me to progress professionally and for the extraordinary experiences they planned for me. It is a privilege to study with them.

I am thankful for my family' constant love and support, which keeps me focused and self-assured. Because they believed in me, they assisted me in achieving my objectives and achievements. My siblings deserve my appreciation because they keep me focused, reminding me of what matters in life, and are always encouraging of my aspirations.

# DEDICATION

I'd want to dedicate my study to my university, parents, and friends, all of whom have been a constant source of inspiration for me. They have instilled in us bravery, enthusiasm, and a sense of purpose.

My sincere appreciation also goes out to everyone who helped in any way; this project would not have been possible without their support.

Contents

[Declaration of Originality 2](#_Toc85063194)

[ABSTRACT 3](#_Toc85063195)

[ACKNOWLEDGEMENT 4](#_Toc85063196)

[DEDICATION 5](#_Toc85063197)

[LIST OF ABBREVIATIONS 8](#_Toc85063198)

[1. INTRODUCTION 9](#_Toc85063199)

[1.1 Background and Literature Survey 11](#_Toc85063200)

[1.2 Research Gap & Research Problem 16](#_Toc85063201)

[1.2.1 Research Gap 16](#_Toc85063202)

[1.2.2 Research Problem 17](#_Toc85063203)

[1.3 Research Objectives 19](#_Toc85063204)

[1.3.1 Main objective 19](#_Toc85063205)

[1.3.2 Specific objectives 19](#_Toc85063206)

[2. METHODOLOGY 20](#_Toc85063207)

[2.1 METHODOLOGY 20](#_Toc85063208)

[2.1.1 Requirement analyzing phase 20](#_Toc85063209)

[2.1.2 System Overview Diagram 21](#_Toc85063210)

[2.1.3 The procedure of the implementation 22](#_Toc85063211)

[2.3 Testing and Implementation 32](#_Toc85063212)

[2.3.1 Implementation Process 32](#_Toc85063213)

[2.3.2 Testing Process 43](#_Toc85063214)

[3. Results and Discussions 47](#_Toc85063215)

[3.1 Results 47](#_Toc85063216)

[3.1.1 Research Findings 47](#_Toc85063217)

[3.2 Discussions 48](#_Toc85063218)

[4. CONCLUSION 49](#_Toc85063219)

[5. APPENDIX 50](#_Toc85063220)

[6. REFERENCES 50](#_Toc85063221)

[Fig . 1: Guy Zyskind Architecture Diagram 11](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623339)

[Fig . 2 Sagar Shah Architecture Diagram 12](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623340)

[Fig . 3: System Overview Diagram 19](file:///C:\Users\ASUS\Desktop\thesis\IT18158564_Individual_Report.docx#_Toc84623341)

[Fig . 4 Implementation Process 20](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623342)

[Fig . 5: Key Feature of Mongodb 22](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623343)

[Fig . 6 Centralized vs Decentralized 23](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623344)

[Fig . 7 Ethereum Process 25](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623345)

[Fig . 8: Express.js initializing 28](file:///C:\Users\ASUS\Desktop\thesis\IT18158564_Individual_Report.docx#_Toc84623346)

[Fig . 9: parsing request 29](file:///C:\Users\ASUS\Desktop\thesis\IT18158564_Individual_Report.docx#_Toc84623347)

[Fig . 10: Routs and error handling 30](file:///C:\Users\ASUS\Desktop\thesis\IT18158564_Individual_Report.docx#_Toc84623348)

[Fig . 11: Demography Model 31](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623349)

[Fig . 12: User model 32](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623350)

[Fig . 13: Define models 33](https://mysliit-my.sharepoint.com/personal/it18158564_my_sliit_lk/Documents/IT18158564%20(1)%20(1).docx#_Toc84623351)

[Fig . 14: Get user profile 34](file:///C:\Users\ASUS\Desktop\thesis\IT18158564_Individual_Report.docx#_Toc84623352)

[Fig . 15: Logout Session 35](file:///C:\Users\ASUS\Desktop\thesis\IT18158564_Individual_Report.docx#_Toc84623353)

|  |  |
| --- | --- |
| Abbreviations | Description |
| API | Application Programming Interface |
| DLT | Distributed Ledger Technology |
| IoT | Internet of things |
| MVP | Minimum Viable Product |
| TV | Television |
| AI | Artificial Intelligence |
| DB | Data Base |
| Dapps | Decentralized Application |
| EVM | Ethereum Virtual Machine |
| SSPL | The Server Side Public License |
| URL | Uniform Resource Locator |

# LIST OF ABBREVIATIONS

# 1. INTRODUCTION

Advertising is a form of commercial communication in which a product, service, or concept is promoted or sold via the use of an explicitly sponsored, non-personal message. In 1941 after the release of the first paid advertisement in the United States, then in the 20th-century advertising grew rapidly in direct mail, television, radio, the mobile devices, and the internet with new technologies. The industry which has been slowly evolving is now expanding on a large scale. In this research, we must be very close to the advertisements and their viewers. We had to gather data from the user and the advertisement companies. Therefore, user privacy is a very important point to address according to this situation and especially in this research project. It is gathering more sensitive data to track emotions. In that case, we have to constantly track the user’s facial images. In any case, if this sensitive data falls into the hands of someone else it will be creating a dangerous situation. Neither I nor anyone else can say that this database or a hosted platform is hacking proof because it can be retrieved by a cyber-attack or by hacking. In any company or organization, there will be security flows. In that case, anyone cannot be certified as a 100% hacking-proof database or a hosted platform. In this sense data is exposed in any way, all sensitive information will be shared among others. When the focus is on the same user privacy is a must.

Blockchain technology is widely acknowledged as a foundation for enabling transaction verification [1], may be used to create a transparent and decentralized advertisement distribution system. The blockchain is a decentralized digital database of transactions arranged into blocks and cryptographically signed. Each block is cryptographically connected to the one before it after it has been validated and a consensus decision has been reached. As new blocks are placed, older ones become increasingly difficult to alter (i.e., creating resistance against tampering). This mechanism has become an origin of fresh optimism in recent years because to its vast range of applications, including banking, supply chain management, medical care, and intrusion detection, to mention a few. The mechanism imprint has recently been observed in intellectual property or copyright protection applications. Openness, decentralization, secure databases, mutual administration, trackability, protection and reputation, the digital coin, and programmable contracts are all essential aspects of blockchain technology for safeguarding digital intellectual property and assuring traceability. In recent years, there has been a rapid growth of decentralized apps based on blockchain technology, but academics have given little attention to the fusion of copyright protection and blockchain technologies. Aside from a few commercially accessible blockchain-based copyright protection systems, there are only a few blockchain-based copyright protection strategies in the literature. [2] [3] The researchers recently looked into the use of blockchain technology in a variety of online multimedia applications, including advertising and, Medicare, social networks, and content distribution networks. We hope to learn more about how blockchain technology may be used to tackle challenges connected to copyright infringement in this project. Another consideration in an AI-based advertising system is data security. As a result, we needed to devise a solid method for ensuring the security of real-time updated data. Our team picked a blockchain that could securely store our data once we got the notion of decentralizing the information.

## 1.1 Background and Literature Survey

In the current world, several research articles and various implementations have been done to discover and analyze customer input. The bulk of the studies used sentiment analysis and techniques to classify texts. By reading some similar research papers, we may learn about studies that have been done in a similar study field. I've attached a couple of research articles below that are similar to the developed component. I've listed some research that is comparable to the created component below. Decentralizing Privacy: Protecting Personal Data with Blockchain is a research paper written by "Guy Ziskind," "Oz Nathan," and "Alex Sandy Pentland.” The protection of user data is the major emphasis of this study. They planned to use the blockchain to decentralize all user information. They plan to make it a third-party component that can be utilized by any other application to keep user data safe in their system. [4]

Diagram

Description automatically generated

Fig . 1: Guy Zyskind Architecture Diagram

The rapid rise in documented cases of spying and security breaches that jeopardize users' privacy calls into question the present method, in which third parties gather and manage enormous quantities of personal data. Bit-coin has proven in the financial arena that utilizing a decentralized network of peers and a public ledger, trustworthy, auditable computing is achievable. They propose a decentralized personal data management system in this article that assures people own and control their data, and they plan to develop a protocol that transforms a blockchain into an automated access-control manager that does not rely on third-party trust. Unlike Bitcoin, our system's transactions aren't purely financial; instead, they're used to convey commands like saving, searching, and sharing data. Finally, they addressed potential blockchain developments that might combine them into a well-rounded answer for society's make trust on computing challenges.

A study paper "Blockchain Voting System" was done at Northeastern University by "Sagar Shah," "Qaish Kanchwala," and "Huaiqian Mi." A client-server architecture is combined with a blockchain technology in the suggested solution. A smartphone or PC with a decent internet connection and camera is the bare minimum for a voter. If these conditions are not satisfied, disadvantaged voters must have access to alternative arrangements such as pop-up internet cafés and PCs. [5]

Figure 2 depicts how the user interacts with the various components of the system. The system is separated into two portions, each of which explains the functionality of each component as well as the system's operations.

Diagram

Description automatically generated

Fig . 2 Sagar Shah Architecture Diagram

“M. Chanson”, “A. Bogner”, “F. Wortmann”, and “E. Fleisch” have been conducted in a research paper as “Blockchain as a privacy enabler, an odometer fraud prevention system”. The researchers have designed a prototype of a system that contain of a dongle, a laptop running a Node.js application, a public Ethereum blockchain, and an Amazon NoSQL database. The dongle communicates with the car’s onboard diagnostics II (OBD-II) and retrieves odometer and GPS values. Two types of values are used in tandem to re-check the distance traveled and odometer value. When the odometer value is decreased, it can be detected via GPS data. This also means the values are gathered often sufficient to map the GPS values to the odometer values. Data is sent to the onboard laptop which hashes the data, adds a timestamp and a nonce, and then writes it to a public Ethereum blockchain. Locally, the data is encrypted using AES before being uploaded to Amazon DynamoDB. When a user wants to check the odometer values, they request a certificate from their mobile application, which interacts with the laptop through REST API. The data is fetched from the Amazon server, decrypted locally on the laptop, verified against the Ethereum blockchain and then a certificate is generated, which can be shared with a third party. The proposed solution faces quite many challenges in the time ahead. The data is vulnerable when sending from the OBD-II to the application. The application requires a discrete laptop. The authors point out their doubts about the Ethereum scalability and whether it persists more reliably than a government-controlled database. The authors plan to move forward with the title, with a focal point on end-user and interaction with the suggested system[6].

"Anders Martoja" published a paper titled "Blockchain in Distributed Systems as a Privacy Enabler," with the goal of preventing audit data. The technology he utilized to create this system was blockchain. The simplest word for the blockchain, according to the author, is a log of records that expands as new details are added. Latest data is entered in the form of records or blocks. A pointer to the preceding block is included in each block. A record also includes a date for when the action block was inserted, as well as the actual data that should be retained in the block. To keep the blocks safe, they are encoded. In a blockchain, changing a record necessitates changing every block that follows it. A great part of the systems having the same blockchain must be modified with for the network to accept the changed blockchain. As a result, blockchain technology is impervious to change[7].

“Zibin Zheng and Shaoan Xie” have been conducted a research paper and they have mentioned decentralization, persistence, anonymity is just a number of the advantages of blockchain. This applications include a wide range of titles, including cryptocurrencies, financial services, risk management, the internet of things (IoT), and public and social services. Although numerous researchers have focused on various applications of blockchain technology, there is yet to be a thorough assessment of the technology from both a technological and an application standpoint. To close this break, we conducted a comprehensive blockchain technology study. This article, in particular, explains the blockchain taxonomy, examines common blockchain consensus algorithms, evaluates blockchain applications, and discusses technical problems as well as recent developments in addressing the challenges. Furthermore, this article discusses the future paths of blockchain technology [8].

“RUI ZHANG” and “RUI XUE” published a research paper and there they have mentioned that In an open environment, blockchain provides a novel method to storing information, conducting transactions, completing activities, and building trust. Many people regard blockchain as a cryptography and cybersecurity technological breakthrough, with applications spanning from widely deployed cryptocurrency systems like Bitcoin to smart contracts, smart grids, and the Internet of Things, among others. Despite the fact that blockchain has sparked increased interest in academics and business in recent years, the security and privacy of blockchains remain a hot topic when it comes to using blockchain in various applications. The fundamental security features that are supported as necessary criteria and building blocks for Bitcoin-like cryptocurrency systems are then described, followed by the extra security and privacy properties that are desirable in many blockchain applications. Finally, we look at the security and privacy approaches used in blockchain-based systems to achieve these security features, such as representative consensus algorithms, hash chained storage, mixing protocols, anonymous signatures, non-interactive zero-knowledge proof, and so on. We believe that this survey will aid readers in gaining a thorough knowledge of blockchain security and privacy in terms of ideas, characteristics, methodologies, and systems [8].

## 1.2 Research Gap & Research Problem

### 1.2.1 Research Gap

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Functions​** | **Research 1 [1]​** | **Research 2 [2][3]​** | **Research 3[4] ​** | **My component​** |
| Using Blockchain​ | No​ | Yes​ | Yes​ | Yes​ |
| Using Smart Contracts​ | No​ | No​ | No​ | Yes​ |
| Advertising Secure​ | Yes​ | No​ | No​ | Yes​ |
| Using IPFS​ | No​ | No​ | No​ | No​ |
| Comodo security​ | Yes​ | No​ | No​ | No​ |

Table 1: Research Gap

As described above, we found that there are similar systems that have already been developed using the principle of blockchain during the literature review, but in those systems, there are many disadvantages.

In order to perform any action, each blockchain-based application needs to go through the public distributed ledger. Systems with an immense client base need to refer to an incredibly broad ledger in order to conduct any blockchain action. It will create a major problem with results. We plan to use smart contracts within the blockchain to prevent this problem. So that device can, with high protection, process data in a very efficient manner.

According to our study, we can find several safety concerns in advertising systems based on AI. Most of the advertising companies Structures distribute ads between TV channels by a physical exchange. Such strategies would create a major security challenge. Instead of this kind of mechanism, the normal way for this phase will be applied through this study. We can store information in a safe way and distribute it when appropriate. Similarly, it is possible to store the information we collect from our viewers in such a way that they do not have to fear anything. In addition to that, IPFS also provides a stable link between network nodes. These mechanisms will ensure user privacy and data security as well.

None of the AI-based advertisement systems currently has proper procedures securely for the store TV viewers' information. Blockchain will give the private data back power. Refer to the existing systems based on the blockchain. In order to drive the answer to the front end, such systems take considerable time. Smart contracts help to receive a very successful response from the blockchain.

### 1.2.2 Research Problem

Today AI-based advertising has been introduced to many countries in the world. Instead of playing advertisements unnecessarily with no choice, In most cases, TV viewers skip advertisements or switch channels, or engage in other activities. Time and money are wasted in ways we never thought possible. In many cases, playing advertisements in vain is detrimental to many parties. AI-based advertising system resolves time-consuming issues.

In addition, with the advancement of technology today, it is imperative to secure separate data. It is a must strictly protect the advertisements provided by the advertising companies. So, it must be concerned about the trust placed in us by those companies. And also need to securely store the data that we collect from TV viewers' TV cameras. when there is any vulnerability and If data is leaked on any issue or in any way it can be the reason for the many violations. In order to avoid mistakes and shortcomings of the existing advertising methods introduced AI-based advertising system was introduced. This system also has some positive and negative effects on advertising. Blockchain technology resolves most of the drawbacks of the AI-based advertising system. Blockchain technology uses Miners, who can give their computational power to validate each and every transaction of the system. the most important thing in this technology is, each and every advertisement maintains a public distributed ledger that retain track of all the transactions. Because of these facts, no one can be able to attack or change the data in the system.

## 1.3 Research Objectives

### 1.3.1 Main objective

The major aim of this research is to increase user preferences by optimizing user-preferred and user-centric advertisements. This means the audience is capable to view advertisements related to their interests. Here factors like age, gender, and peer groups are taken into consideration when determining the most suitable advertisement on the TV during a commercial break. This project will be beneficial for both the audience and the advertisement firm by letting the audience make an interest in interested goods which will be advertised and the ability to advertising firms to create advertisement content based on age, gender, and peer group which will directly affect the users in purchasing goods and services more effectively and efficiently.

### 1.3.2 Specific objectives

* Securing both initially captured data and advertisements to ensuring privacy issues
* Design and implement an admin panel to upload video advertisements and tags through an API.
* Handling the blockchain process using an Ethereum based blockchain.

# 2. METHODOLOGY

## 2.1 METHODOLOGY

### 2.1.1 Requirement analyzing phase

Initially when research on this study area got to know that nobody wants to have their data stored by someone else. Therefore, we have to go on a better solution. Then went through to know and find the best way to secure the data. Here using Ethereum for store the Images, that means user’s most sensitive data. There is a unique, canonical machine in the Ethereum world (known as the Ethereum Virtual Machine, or EVM) which state that everybody on the Ethereum platform accepts on. Every Ethereum node (participant in the Ethereum network) retains a copy of this computer's current state. Whatever participant can also broadcast a request for this machine to execute any computation they choose. When a request like this is broadcast, other network members confirm, verify, and conduct out (or "execute") the calculation. The EVM's state changes as a result of this operation, which is published and transmitted all through the system.

### 2.1.2 System Overview Diagram

**Diagram

Description automatically generated**

Fig . 3: System Overview Diagram

### 2.1.3 The procedure of the implementation

Diagram

Description automatically generated

Fig . 4 Implementation Process

In this part, there indicate whole backend, all the components of the application are connect through the backend. In this case, this component is consists of data gathering, storing, and create endpoints to the data gathering part. Before saving the data into the collection it is a must to add blockchain encryption,

Blockchain is a technique of storing data in such a way that altering, hacking, or defrauding it is difficult or impossible. A blockchain is a digital ledger of transactions that is replicated and disseminated throughout the network of computer systems that make up the blockchain. Each block on the chain contains a number of transactions, and whenever a new transaction occurs on the blockchain, a record of it is added to each participant's ledger. Distributed Ledger Technology (DLT) is a decentralized database that is managed by a large number of people (DLT).

Send data to MongoDB after applying blockchain encryption. MongoDB Inc. created it, and it's released under the Server Side Public License (SSPL). It employs JSON-like documents with optional schemas and is classified as a NoSQL database software. It's a cross-platform document-oriented database application that's open source. MongoDB has a number of distinguishing characteristics. We chose MongoDB for the solution because of those essential features.

Chart, diagram, bubble chart

Description automatically generated

Fig . 5: Key Feature of Mongodb

In today's society, our physical and digital identities collide in a million different ways. It's difficult to share and acquire data with partners and third-party providers to expedite company operations while keeping the data safe and authentic. Last year was a watershed moment for blockchain technology. Across its numerous applications, blockchain enables the publication of identification information among multiple entities while keeping a transparent cryptographic record, thanks to its trust, distribution, and immutability. Sensitive data is subjected to a stringent clearance procedure and is encrypted so that only authorized individuals may access it.

The created system is a decentralized system it implies that this system is control and managing by several entities, not just one entity. The Application Programming Interface (API) build with node.js .It is a centralized system. The whole backend part works from one node.js server but blockchain needs to be integrated into security and the existing centralized system needs to be decentralized to increase security. The image below will help you get a better idea.

Diagram

Description automatically generated

Fig . 6 Centralized vs Decentralized

That is how blockchain is used to decentralized. For that process use an already existing decentralized blockchain which is Ethereum. Our server is deployed on an Ethereum network. Ethereum network is a decentralized system, Ethereum will stick the platform to developers to build their applications on top of their platform, This platform is fully decentralized, It Implies that there are many clusters of machines. When the application is deployed, it is executed in the same way as the function or data is deployed. As an example when you first create an object, you first deploy a class. All you have to do is call it New Class. Then it is saved in any location of the decentralized system. Then all we need to do is call the ID when retrieving the data. Afterward, the system handles all those requests. By building the application on Ethereum, the system will be automatically decentralized and secure. Ethereum is a public, open-source, blockchain-based distributed ledger that supports smart contracts (scripting). It allows developers to create blockchain apps with business logic that run in a faithless environment while using the Ethereum network's high availability.it opens the path to a global financial system in which all you need is an Internet connection to access trustless apps, goods, and services. Without the need for third parties or the possibility of censorship, anybody may connect with the Ethereum network and participate in this digital economy. While the term "contract" conjures up images of formal agreements, "smart contracts" in Ethereum are just bits of code that run on the blockchain and are guaranteed to generate the same outcome for everyone who uses them. These may be used to build a variety of Decentralized Applications (Dapps), such as games, digital collections, online voting systems, financial goods, and more.

Solidity programming language is the language used in the Ethereum Virtual Machine(EVM). Solidity is object-oriented, high-level language deploy to develop smart contracts. It is formed by JavaScript, C++ and Python and solidity is a curly bracket language. The image below will help you get a better idea of Ethereum.

Graphical user interface, text

Description automatically generated with medium confidence

Fig . 7 Ethereum Process

According to my research area mainly there are three models in API, that are

1. Demography model
2. Emotion and facial features model
3. User model

Properties are UserId and images are in an array as a string and timestamp are used as the property of the emotions. In the user model, there is user authentication and create users.There are many endpoints to this component it will be described beyond here.

Briefly about emotions and facial features, One or more movements or postures of the muscles beneath the skin of the face constitute a facial expression. These motions, according to one set of disputed hypotheses, communicate an individual's emotional state to onlookers.  In this case, the user’s emotions take center stage because it is very helpful for identity which advertisements are they like, Here we mainly concern about 7 facial features that are sadness, surprise, happiness, contempt, fear, anger, and disgust.

In the emotions and facial features endpoint there we store emotional and facial features that come to the data endpoint as an array this is a post endpoint and data reload as a post request. That post request embedded the emotions and facial features that we gathered by the UserId. That means the user’s facial features collected from the beginning to the end will be sent to this endpoint along with the time stamp. And then save it in the database.

The next endpoint saves the demographic data in the database. In this case, it sends a post request and takes images from the front-end, and converts it as a based 64 encoded string. Then send it as a post request to the API. Finally, save the request as a based 64 encoded request in the database. Then the demography\_userid endpoint sends a get type request. This is designed to get demographic data from the particular user. To collect user’s emotions and facial features designed another endpoint as Emotions\_userid,

Recommended\_Ads endpoint is a post request type endpoint. In this endpoint. All data demographic data, emotional data, and facial features send to this endpoint then send all data into the recommended model that was created and trained. According to the gathered data, the recommended model will filter an advertisement that matches the user's age, gender, emotions, and facial features. After that in this endpoint, the recommended advertisement number send into the frontend. In this part send only the advertisement number into the frontend. because particular advertisement assigned with an AdvertisementId.In the last endpoint, it retrieves the advertisement Uniform Resource Locator (URL) and sends it to the front-end. And in the advertisement endpoint, it sends a post request and sends an advertisement URL as the body parameter,

**2.2 Commercialization aspects of the product**

Traditional media is still used by the majority of Sri Lankan media users. Traditional media advertising are mostly targeted based on demographics and time. Because the advertising technique is obsolete, a significant amount of money is being squandered. This loss impacts not just the marketer, but also the content consumer. The consumer is forced to view what he or she does not want to watch. The goal of advertising is to enlighten the audience about the product and the firm, as well as persuade them that the marketers' items are the best on the market. The number of impressions, click true rate, and direct purchases are all used to determine the success rate of digital ads. However, there is no robust, quick, or accurate technique for quantifying them in traditional advertising. We recommend incorporating digital ad tactics into traditional ad campaigns. Because the majority of Sri Lankans still rely on conventional advertising, there is a big market for our goods. In addition, we demonstrated in this study that our prototype is capable of proposing user-favored advertising. Apart from these benefits, with the aid of the gadget we are putting into the television, we can determine the number of impressions and if the audience enjoyed the ad presented to them in real-time.

Graphical user interface, text, application, chat or text message

Description automatically generated

Fig . 8 : Facebook Page

Graphical user interface, text, application, chat or text message

Description automatically generated

Fig . 9 :Comment Section

## 2.3 Testing and Implementation

### 2.3.1 Implementation Process

**T**his research area contains the whole backend and the encryption part there for this is deployed on the Ethereum platform. Initially created the system as a Minimum Variable product(MVP). Minimum Variable product contains that it is a version of a product that has only enough functionality for early consumers to use and offer input for future product development. Developers might save long and (ultimately) needless labor if they focus on providing an MVP. Instead, they iterate on functioning versions and respond to input, questioning and verifying assumptions about the requirements of a product. It might also entail conducting market research beforehand. because of all build a system backend on the stack of Nodejs, Expressjs, and MongoDB. Express.js, or simply Express, is a back-end web application framework for Node.js that was published under the MIT License as free and open-source software. It is intended for the development of web applications and APIs. It's been dubbed Node.js' de facto standard server framework and Node.js is a server-side programming language that's mostly used for non-blocking, event-driven servers like standard websites and back-end API services, although it was built with real-time, push-based architectures in mind. Every browser has its JS engine, and node.js is one of them. Then the MongoDB is developed by MongoDB Inc. and licensed under the Server Side Public License (SSPL). Classified as a NoSQL database program, It uses JSON-like documents with optional schemas. It is a source-available cross-platform document-oriented database program. By using this stack build a Minimum Variable product(MVP) for the backend to save, store and retrieve the data.

Then the whole backend should build on the Ethereum platform at the same time the system becomes auto decentralized and security increases as required. Because security is the major part of this research. This task involves dealing with highly sensitive data. If the data leaks in any way, It will create a lot of dangerous situations. Therefore, the responsibility to protect sensitive data should not be compromised and Ethereum helps to accomplish that.

Text

Description automatically generated

Fig . 10: Express.js initializing

As a dependency of Node.js here used express.js. It is used for designing and building web applications quickly and easily. Then used cors for import cross regent, when briefly describe the cors, Cross-Origin Resource Sharing (CORS) is an acronym for cross-origin resource sharing. It is a technique that allows or restricts access to requested resources on a web server based on the origin of the HTTP request. This policy is intended to prevent other websites or domains from accessing a certain web server. Then initiate MongoDB inside the mongo store and at the end all datastore on the MongoDB cloud. It is the Quickest and Easiest Way to Deploy, Operate, and Scale MongoDB on the Cloud. For that used a free tier cluster and initialize the connection. Then register an event to lock the console when there is an error occurs. That is what is called DB on error in the code. The console locks when trigger an event. After opening the connection once, the open function used to be a one-time trigger, there open the connection and sending a lock message.

In the beginning import the cross regent, now the library should enable the cross regent by the line app.use (cors ()). Then the session is encrypted. After the encryption part, the secret word is used as the work hard then it is able to use a random string in this point. then encode the session as above and instance of the MongoDB save in the variable that is used as a store.

Text

Description automatically generated

Fig . 11: parsing request

The Node.js body parsing middleware is called body-parser. It's in charge of processing incoming request bodies in a middleware before you deal with them. In this case, body-parser initialize how much data can be passed through a Get or Post request. then send a body parameter in JSON format. It is initialized as JSON. Then URL encoded implies that URL is sent into query parameters after encoded, query parameters have initialized. App. use body-parser URL encoded is the standard way to use enable URL encoded query parameters, In next line extended: False; refers to the data sent through the query parameters only for standard length, security protocols and guidelines suggest that a server's maxQueryStrings limit should be set to 1024 characters. While the whole length of the URL, including the query string, should not exceed 2048 characters. because of that, this line sends the limitations for the URL. Then there is app,use(express,static(\_dirname +)’/views’)); it is a static view to register the users.

Text

Description automatically generated

Fig . 12: Routs and error handling

Routs are decided to when initialized data is reached to any router and send it into the roots then redirect the data into the routes file. After this session server initializing and listen to data on the 3210 port.

According to this component, there are three models in API, that are Demography model, Emotion and the facial features model User model Properties are UserId and images are in an array as a string and timestamp are used as the property of the emotions. In the user model, there is user authentication and create users.

Text

Description automatically generated

Fig . 13: Demography Model

In the demography model used user id and the images, there is a userId for the particular image that assigned, According to that there is an image array and it sends base 64 encoded string. Base 64 encoded string used when it comes to encoding binary data that has to be saved and transferred through the medium that is designed to handle textual data, Base64 encoding techniques are widely employed. This is done to guarantee that the data is not tampered with during transmission. Save the images that come through the base64 encoded string as an array.it is the procedure of the above image.

In the emotions model, there are all parameters that need to collect from the user the parameters are timestamp, userid, anger, attention, brow furrow, brow raise, cheek raise, chin raise, contempt, dimpler, disgust, engagement, eye closure, eye widen, fear, Inner brow raise, jaw-drop, lip frighten, joy, etc, then the data is saved in the mongoose model as the collection of emotions.

Text

Description automatically generated

Fig . 14: User model

In the user model, it registers a user by the parameters according to the model and saves the data, In this model, there is both user registration and user authentication, Parameters of this model are usernames, email, password, and password confirmation.

Text

Description automatically generated

Fig . 15: Define models

Then in the routes.js file initialized and define the models that are emotions, demography data, and users, It has models loaded in it. there are emotions. In the postman collection, there is body parameter as an array assing as face params. Then alternate the particular object and set the userid in the objectives. Then create the await emotions in the time stamp and the userid. At that point create an emotions model and save the data and, send a response whether it is successful or unsuccessful.

In the emotions and facial features endpoint there we store emotional and facial features that come to the data endpoint as an array this is a post endpoint and data reload as a post request. That post request embedded the emotions and facial features that we gathered by the UserId. That means the user’s facial features collected from the beginning to the end will be sent to this endpoint along with the time stamp. And then save it in the database.

The demography part creates an object with the use of userId and images parameters. Images are sent as an array then save the object in the demography collection and send a response whether successful or unsuccessful. The next endpoint saves the demographic data in the database. In this case, it sends a post request and takes images from the front-end, and converts it as a based 64 encoded string. Then send it as a post request to the API. Finally, save the request as a based 64 encoded request in the database. Then the demography\_userid endpoint sends a get type request. This is designed to get demographic data from the particular user. To collect users’ emotions and facial features designed another endpoint as Emotions\_userid. Then in the next part, there is to create a user. In the frontend there create a user and check the userId password and password confirmation. when the details are confirmed start to create a user. After that check the email, At this point, there is the same route to both authenticate and lock the data. Here it is a must to check the body parameter whether there indicates accounts or emails and passwords, if it is emails and passwords the system authenticates the user if not creates a user.

Text

Description automatically generated

Fig . 16: Get user profile

In this code, the segment Get profile refers to get a registered user’s profile. When getting a user profile there automatically creates a user session when logging into the profile. There is userId and it is embedded. after that send data into the front end by using the userId.

Text

Description automatically generated

Fig . 17: Logout Session

In the above session, there are router logout functions that line refers to the session log out and clear the session. After clear the session and logout, destroy the requested session otherwise it is returned and redirect.

In the below segment [Fig: 16] first line import csv file and call the method then run the function with the parameters request, response, next and count contains array count of the csv. It calls console log //console.log(‘came here’) then start the Javascript file.Then there is declare the variable py. The word "spawn" refers to a parent process creating a child process. Of addition, the parent process can continue asynchronously or wait till the child process completes its execution. Python's parallel computing module aims to start a process by following these steps: Create the object-processing procedure. Coun the Data in the Classify.py send the data through the array.Then read the array on the function and map the data.After end the function. Then if function is success send a true message to the frontend and write the string value.

Text

Description automatically generated

Fig . 18: read and write classify data

To create a router it use the gollowing code segment. It send a post type request and call the path classify.As the parameters it gives three inputs request, response and next. There is a condition and there is a parameter in body as request and inside it there is emotion.Then write the data through the string array and convert the data into csv and return the amotions and length.If success send the datalength to the front end and if it is unsuccesful send a message as “Please try again”.

Text

Description automatically generated

Fig . 19 : create a router

In the below image it contains the whole dashboard there is

* Compute units
* Median Response
* Success
* Rate Limited
* Success of Last 24hrs
* Total Request of Last 24hrs
* Invalid Requests of Last 24hrs

Graphical user interface, treemap chart

Description automatically generated

Fig . 20: Testing Dashboard

Fig . 21: Testing Dashboard

### 2.3.2 Testing Process

Postman is just an API framework that allows you to build and use APIs. Postman streamlines cooperation and reduces each phase of the API cycle, allowing you to develop better APIs quicker. Following snap shots are showing that the testing process using the Postman platform. By using notifications and privacy warnings, search, analytics, and more, the Postman platform provides enhanced analytics and insights into all of your API activities. In this snapshot it contains the timestamp cording to the given username. It shows the successful results. As it we can check and test each component.

Graphical user interface, text, application, email

Description automatically generated

Fig . 22: postman testing

In the below image there is Ethereum console and there is each and every testing part. There is recent requests, Recent Invalid Requests and Recent Rate Limited Request.

Anyone can take a better idea by went through the following image.

Graphical user interface, application

Description automatically generated

Fig . 23 : Testing Ethereum Console

In this image there is the compilation process of the Node Package Execute (npx) and the compilation has finished successfully. After that there is npx deployment.

Text

Description automatically generated

Fig . 24: npx hardhat compile

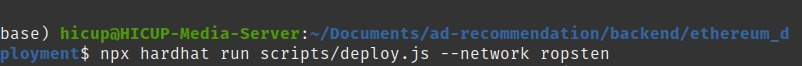


Fig . 25: npx deploy

# Results and Discussions

## Results

## Research Findings

In the research finding it using Ethereum. Ethereum recommend for the use of transaction and store the transactions. Our server is deployed on an Ethereum network. Ethereum network is a decentralized system, Ethereum will stick the platform to developers to build their applications on top of their platform, This platform is fully decentralized, It Implies that there are many clusters of machines. When the application is deployed, it is executed in the same way as the function or data is deployed. As an example when you first create an object, you first deploy a class.

## Discussions

Secure user data is a major concern according to this research. In the case of recommending advertisements through the users’ demographic data and emotions, it is collected the sensitive data from each user. The data's integrity and security are consequently important, as any loss or corruption may result in user privacy. Smart contracts in a private Ethereum blockchain were suggested in 2019. Such systems would safeguard data against loss in the event of a failover, as well as accidental or purposeful corruption. Also has larger ramifications, demonstrating the possibility for blockchain technology to be used to tackle real-world difficulties in addition to cryptocurrencies.

In this research study, there is a proof-of-concept solution to store user data using the blockchain with the help of the Ethereum platform. The solutions are going through by addressing the need for data security and accessibility in sharing these sensitive data and used for practical needs in the real world for memory efficiency and for time. The study consists of that blockchain technology offers not only immutability and the security of the user data, but also blockchain offers practicality and efficiency too. Ethereum Smart contracts build in Ethereum platform It was able to develop efficient solutions according to this study. Building up a separate blockchain on Ethereum necessitates extensive platform expertise, and implementing the contract is a time-consuming procedure. This procedure may be reduced into an executable program, reducing the requirement for platform expertise. Nevertheless, there will still be flaws in Ethereum software, including such web3.js, the Ethereum JavaScript API. We were able to work around these software difficulties, but we recognize that this platform must be more stable before academics start utilizing it as a centralized database.

# CONCLUSION

Securing user data is a huge problem in future because now a days there is many parties are used to do corruptions or hacking. Today, Bitcoin and Ethereum are the most well-known and valued cryptocurrencies. They are built on blockchain technology, which is designed to create a peer-to-peer network security framework based on the agreement of the majority of nodes. There has been a significant increase in the number of hashing algorithms, cryptocurrencies, and consensus agreements in the networks during the last several years. In conclusion, we provided a challenging approach for storing and accessing user data in a smart contract on the Ethereum blockchain, as well as an additional rapid Data approach that greatly improved examining these issues and durability. It is rapidly spreading and becoming popular technique, blockchain and Ethereum. This component has to be built around Solidity's limitations, such as the maximum number of local variables allowed in a function, the 'gas' restriction, and other Ethereum-specific features. Our solutions show how blockchain technology may be used in much research, but they might also be used to solve a number of other securities, storage and other challenges.

# APPENDIX

A picture containing graphical user interface

Description automatically generated

Fig . 26 : Similarity Index

# REFERENCES

[1] Drew Springall, Travis Finkenauer, Zakir Durumeric, Jason Kitcat, Harri Hursti, Margaret  MacAlpine and J. Alex Halderman, ” Estonian I-Voting System”. Available:

<https://dl.acm.org/citation.cfm?id=2660315>

[2] Anil K. Jain, Lin Hong, Sharath Pankanti and Ruud Bolle, “An Identity-Authentication  System Using Fingerprints ” Available:

http://biometrics.cse.msu.edu/Publications/Fingerprint/JainEtAlIdentityAuthUsingFp\_ProcI EEE97.pdf

[3] Shiri Azenkot, Kyle Rector, Richard E. Ladner, and Jacob O. Wobbrock, “PassChords:  Secure Multi-Touch Authentication for Blind People” Computer Science & Engineering,  The Information School DUB Group University of Washington. Available:

<http://faculty.washington.edu/wobbrock/pubs/assets-12.01.pdf>

[4] Guy Zyskind, Oz Nathan, Alex ’Sandy’ Pentland “Decentralizing Privacy: Using

Blockchain to Protect Personal Data”, Available:

<http://homepage.divms.uiowa.edu/~ghosh/blockchain.pdf>

[5] Sagar Shah, Qaish Kanchwala, Huaiqian Mi, “Blockchain Voting System” Northeastern  University.Available: <http://www.economist.com/sites/default/files/northeastern.pdf>

[6] M. Chanson, A. Bogner, F. Wortmann, and E. Fleisch, “Blockchain as a privacy enabler, An odometer fraud prevention system” New York, NY, USA, pp. 13–16, 2017. Available: <https://dl.acm.org/citation.cfm?doid=3123024.3123078>

[7]Zheng, Zibin, et al. “Blockchain Challenges, and Opportunities: A Survey.” International Journal of Web and Grid Services, Oct. 2018. world, [www.inderscienceonline.com](http://www.inderscienceonline.com), <https://www.inderscienceonline.com/doi/abs/10.1504/IJWGS.2018.095647>.

[8]Zhang, Rui, et al. “Security and Privacy on Blockchain.” ACM Computing Surveys, vol. 52, no. 3, July 2019, p. 51:1-51:34. July 2019, https://doi.org/10.1145/3316481. [9] Neil McClure and Kermit Lohry, “Electronic voting system”. Available:

<https://patents.google.com/patent/US6250548B1/en>

[10] Andrew Barnes, Christopher Brake, and Thomas Perry, “Digital Voting with the use of  Blockchain Technology” Available:

<https://crypto.stackexchange.com/questions/3474/approach-towards-anonymous-e-voting>

[11] Chris Karlof, Naveen Sastry and David Wagner, “Cryptographic Voting Protocols: A  Systems Perspective” Available:

<https://www.usenix.org/event/sec05/tech/full_papers/karlof/karlof.pdf>

[12] Anders Martoja, “Blockchain in distributed systems as a privacy enabler”. Available:  https://pdfs.semanticscholar.org/6cfa/09c94cc6a914d0a32e9204cec8fa343fdb23.pdf

[13] “Postman API Platform.” Postman, <https://www.postman.com/product/what-is-postman>

[14] “Home.” Ethereum.Org, <https://ethereum.org>

[15] Blockchain Explained: What Is Blockchain? | Euromoney Learning.//www.euromoney.com/learning/blockchain-explained/what-is-blockchain

[16] MongoDB. Webinar: MongoDB 2.6 New Security Features. <https://www.slideshare.net/mongodb/11-14-26-securityfeatures>.

[17] Chen, Ting, et al. “Understanding Ethereum via Graph Analysis.” ACM Transactions on Internet Technology, vol. 20, no. 2, Apr. 2020, p. 18:1-18:32. May 2020, <https://doi.org/10.1145/3381036>.