

Article

# BS-GeoEduNet 1.0: Blockchain-Assisted Serverless Framework for Geospatial Educational Information Networks

Meenakshi Kandpal <sup>1</sup>, Veena Goswami <sup>2</sup> , Yash Pritwani <sup>3</sup> , Rabindra K. Barik <sup>2</sup> and Manob Jyoti Saikia <sup>4,\*</sup> 

<sup>1</sup> School of Computer Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar 751024, India

<sup>2</sup> School of Computer Applications, Kalinga Institute of Industrial Technology, Bhubaneswar 751024, India

<sup>3</sup> Department of Computer Science and Engineering, Odisha University of Technology and Research, Bhubaneswar 751029, India

<sup>4</sup> Department of Electrical Engineering, University of North Florida, Jacksonville, FL 32224, USA

\* Correspondence: manob.saikia@unf.edu

**Abstract:** The integration of a blockchain-supported serverless computing framework enhances the performance of computational and analytical operations and the provision of services within internet-based data centers, rather than depending on independent desktop computers. Therefore, in the present research paper, a blockchain-assisted serverless framework for geospatial data visualizations is implemented. The proposed *BS-GeoEduNet 1.0* framework leverages the capabilities of AWS Lambda for serverless computing, providing a reliable and efficient solution for data storage, analysis, and distribution. The proposed framework incorporates AES encryption, decryption layers, and queue implementation to achieve a scalable approach for handling larger files. It implements a queueing mechanism during the heavier input/output processes of file processing by using Apache KAFKA, enabling the system to handle large volumes of data efficiently. It concludes with the visualization of all geospatial-enabled NIT/IIT details on the proposed framework, which utilizes the data fetched from MongoDB. The experimental findings validate the reliability and efficiency of the proposed system, demonstrating its efficacy in geospatial data storage and processing.

**Keywords:** blockchain; serverless computing; AWS Lambda; AES encryption; decryption; data visualization; queueing model; MongoDB



**Citation:** Kandpal, M.; Goswami, V.; Pritwani, Y.; Barik, R.K.; Saikia, M.J. *BS-GeoEduNet 1.0: Blockchain-Assisted Serverless Framework for Geospatial Educational Information Networks*. *ISPRS Int. J. Geo-Inf.* **2024**, *13*, 274. <https://doi.org/10.3390/ijgi13080274>

Academic Editors: Wolfgang Kainz and Eliseo Clementini

Received: 20 May 2024

Revised: 22 July 2024

Accepted: 30 July 2024

Published: 1 August 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

As we move into the age of digitization, a reliable method for managing and processing vast quantities of data is becoming increasingly important in a variety of fields, including healthcare, banking, transportation, the environment, and education. In particular, in education, it is necessary to develop a secure and effective system that promotes educational infrastructure across the web. The integration of advanced technologies like blockchain and serverless computing into educational data management is key to overcoming complex challenges. Additionally, authorities and governments have encouraged academics and product developers to produce societal, environmental, and economic solutions through the application of cutting-edge technologies, such as serverless computing [1–3] and blockchain technology [4]. This is in response to the fact that these technologies are expected to play an increasingly important role in the future. In the past, educational information systems were developed with centralized trustworthy processes serving as the intermediary between the consumer and the cloud provider [5]. These systems' enormous amounts of data were sent to centralized cloud infrastructures so that the intelligence they contained could be processed.

Blockchain technology improves data security, transparency, and collaboration significantly, which makes it an appealing tool for a geographic educational information network. Geospatial data, including maps, location-based educational materials, and research findings, can have their authenticity and integrity guaranteed by the network through the

utilization of blockchain technology's unchangeable ledger. This security feature reduces the possibility of illegal data manipulation or tampering, which is essential for preserving the dependability and accuracy of educational materials. In addition, the decentralized nature of blockchain reduces dependency on centralized servers and improves availability and fault tolerance by decentralizing the network infrastructure. Additionally, by enabling the creation of decentralized learning environments and applications and supporting the safe and transparent sharing of educational resources, the integration of blockchain technology fosters innovation in geospatial education. All things considered, blockchain technology greatly improves the effectiveness, security, and collaboration potential of geospatial educational networks, opening the door for more inclusive and successful learning environments around the world.

This research work investigates the potential applications of serverless computing and blockchain technology in educational information systems [6]. An innovative system for storing and visualizing data based on serverless computing and blockchain technology is presented here. Blockchain technology is a decentralized distributed ledger that can be used to track blocks or validate transactions without the involvement of any trusted parties. The technology has spread over three different domains, namely transactions, assets, and smart contracts. This research paper presents a detailed exploration of system development, implementation, and evaluation. This system effectively integrates serverless computing and blockchain technology, secure encryption methods, and efficient data processing mechanisms.

The primary focus of this paper revolves around creating a data storage system that is both scalable and robust. Utilizing the power of Amazon Web Services (AWS) Lambda, a serverless computing platform by AWS, the system introduces a paradigm shift in data storage architecture. Traditional data storage and processing methods often encounter limitations related to scalability, cost-effectiveness, and responsiveness. In contrast, the use of serverless computing, particularly AWS Lambda, eliminates the need for manual infrastructure management and dynamically allocates resources as needed. This approach offers a solution that scales effortlessly with growing data volumes and user demands. At the core of this architecture is the integration of blockchain technology, renowned for its decentralized and immutable nature. By adopting a serverless blockchain platform, the system ensures data integrity, transparency, and traceability. Each piece of information, such as the geospatial coordinates of educational institutes, is encrypted and securely stored on the blockchain, generating a unique key. This not only ensures data security during transmission and storage but also simplifies auditing processes, a crucial aspect of educational data management.

Furthermore, the system employs advanced encryption standard (AES) encryption and decryption layers to safeguard data confidentiality [7]. This ensures that sensitive information remains private and secure from unauthorized access. When combined with blockchain technology, this multi-layered security approach guarantees end-to-end data protection, a critical consideration when handling educational data containing sensitive location-based information. The intricacies of the system are exemplified through the development of a web application designed to visualize the geospatial coordinates of educational institutes. This application employs a user-centric workflow, allowing users to upload CSV files containing coordinate data. Subsequently, the data undergo encryption and are uploaded onto the blockchain, generating a corresponding key for secure storage. The decryption process, initiated using the key, retrieves the files from the blockchain and decrypts them, preparing the data for normalization and storage in MongoDB.

One of the notable achievements of the proposed system is its emphasis on scalability. To ensure efficient handling of data processing fetched from the blockchain, a queueing model utilizing Apache Kafka is integrated. This model efficiently manages data processing during heavier input/output (IO) processes, contributing to the system's ability to handle larger datasets with optimal performance. The efficacy of the system is demonstrated through an experimental analysis that validates its reliability and efficiency. The success-

ful implementation showcases its potential to revolutionize geospatial data storage and processing within the education sector. This paper concludes with the visualization of geospatial coordinates on Google Maps, underscoring the practical implications and value of the proposed system.

### 1.1. Motivations

The main motivation of the research is to integrate modern technologies to offer a comprehensive solution for educational data management, addressing critical challenges and empowering users. In today's digital era, efficient management of educational data is imperative. This research stems from the need to harness advanced technologies like blockchain and serverless computing for secure, scalable, and user-friendly educational data analysis and management with the following benefits.

- **Scalability and robustness:** Traditional data storage methods often struggle with scalability and cost-efficiency. Leveraging AWS Lambda for serverless computing promises dynamic resource allocation, ensuring effortless scalability as data volumes grow.
- **Data security and integrity:** Data breaches are a constant concern. Blockchain technology guarantees data integrity and transparency, while encryption layers protect sensitive information during transmission and storage, addressing data security comprehensively.
- **Confidentiality:** Protecting sensitive educational data is paramount. Advanced encryption safeguards confidentiality, especially for location-based information, complemented by blockchain's end-to-end data protection.
- **User-centric visualization:** Effective data visualization is essential. The web application developed in this research allows users to intuitively upload and visualize geospatial data, enhancing practicality and usability.
- **Efficiency in data processing:** Efficient data processing, especially with blockchain, is achieved through an Apache Kafka-based queueing model, enhancing system performance during heavy IO processes.
- **Real-world application:** The research's experimental analysis validates the system's reliability and efficiency, showcasing its potential to transform geospatial data management in education.

### 1.2. Contributions

This research paper makes the following contributions:

- It proposes *BS-GeoEduNet 1.0* i.e., a blockchain-assisted serverless framework for geospatial data visualization.
- It demonstrates geospatial educational institutes' location visualization through the proposed framework, showcasing the exact location of educational institutes.
- It presents a mathematical and analytical queueing model that underlies the proposed framework for better scalability solutions.
- It discusses the performance analysis of the proposed model.
- It performs geospatial educational network data visualizations as part of a case study of all NITs/IITs in India.
- Finally, experimental findings and performance evaluations of the proposed architecture are presented along with graphs.

The *BS-GeoEduNet 1.0* framework offers several unique contributions to the field of geospatial educational networks. Firstly, it integrates blockchain technology with serverless computing, leveraging AWS Lambda for dynamic resource allocation, which ensures robust data processing capabilities without the need for manual infrastructure management. Secondly, the use of AES encryption combined with Apache Kafka for secure and efficient data handling represents a significant advancement in data security and scalability. These innovations address specific gaps and limitations in existing solutions, particularly in the secure, scalable, and efficient management of geospatial educational data.

### 1.3. Organizations

The remainder of this paper is organized in the following fashion. The related studies, with an overview of the blockchain-assisted serverless framework, its applications in various fields, and geospatial data visualizations, are presented in Section 2. Section 3 illustrates the proposed framework descriptions. In Section 4, an evaluation of the analytical queueing model is presented. Section 5 discusses the case study of a blockchain-assisted serverless framework for a geospatial education information network. Section 6 illustrates the results and presents a discussion of the present study. Finally, Section 7 presents the concluding remarks and potential future directions of the present research.

## 2. Related Work

This section presents a comprehensive review of blockchain-based serverless frameworks and their applications in various scenarios.

Numerous methods for using blockchain with geographical data have been investigated in the literature in recent years. However, there has not been much research on serverless frameworks aided by blockchain. This section provides a general overview of how serverless frameworks and blockchain technologies are used for geospatial data visualization. Blockchain-assisted serverless framework integration improves security, scalability, and cost-effectiveness while facilitating real-time insights, transparency, and regulatory compliance in geospatial data visualization. Together, these advantages give businesses the ability to innovate in geospatial data applications, optimize processes, and make well-informed decisions.

Several works on blockchain technology show that it provides notably improved results in IOT, healthcare, agriculture, supply chain management, digital transformation, data governance, etc. [4,8–13]. However, various approaches have been explored in the literature for the threats that persist in geospatial data. Integrating blockchain with geospatial data visualization where accurate and immutable location data are critical ensures the authenticity and integrity of location-based data. Blockchain technology has gained significant attention for its potential to revolutionize data management and security. By employing a decentralized and immutable ledger, blockchain offers enhanced transparency and trust, which ensures data integrity and eliminates the need for intermediaries, in geospatial data visualization applications.

Farnaghi and Mansourian [14] proposed the development of decentralized applications (DApps) based on blockchain technology for public participatory GIS (PPGIS) to create an open, transparent, and accountable environment. They highlighted blockchain-based PPGIS safe data storage, unalterable data, and open accessibility for institutions and citizens. They developed a prototype PPGIS DApp for participatory site selection that compares and orders factors using analytical hierarchy. The recommended PPGIS application efficiently meets urban decision-making transparency standards. Also, Ozcelik [15] discussed the application of blockchain technology in land administration, specifically in land registration. They stressed the importance of current land data and the dangers of unauthorized record changes. Blockchain may record land transactions and rights, decentralized and tamper-proof, according to the author. They developed a blockchain-based geospatial architectural paradigm for real-time land register systems and emphasized the significance of integrating spatial and temporal information [16]. Integrating blockchain with geospatial data visualization ensures the authenticity and integrity of location-based data.

Serverless computing is widely regarded as the subsequent stage in the development of the computer industry [17]. In this paradigm, the user is not responsible for the administration of any servers. It is another name for the software architecture known as ‘function-as-a-service’, in which an application is made up of a collection of one-time functions that are activated in response to events or other triggers [18]. When a function is triggered, a container that contains the runtime environment, such as node.js, is provisioned, and the function is then injected into the container. It is possible to recycle the container and use it for another purpose during the subsequent invocation, so its removal

is delayed. Computing without servers has several advantages, including auto-scaling and fault tolerance [19,20]. Serverless cloud computing has gained attention for its ability to distribute resources and services, allowing data to be cached on end-user devices and stored/accessed on centralized servers [21,22]. Serverless computing complements geospatial data by providing a scalable and cost-effective framework for processing and visualizing large datasets.

The authors of [23,24] proposed a system design for geospatial data processing using serverless technologies and open standards. It aims to minimize human intervention and resource consumption while promoting interoperability. Implementing a public cloud provider demonstrates feasibility and compatibility with best practices. Furthermore, Mete and Yomralioglu [16] described the development of a serverless cloud GIS framework for a platform for land value. They stored, analyzed, and shared geographic data using the Aurora Serverless PostgreSQL database on Amazon Web Services (AWS). Compared to an Amazon Relational Database Service (RDS) instance, the serverless database performs better, with faster query response times. The study stored and distributed raster-based land value map tiles on S3 and AWS Lambda. Online mapping framework Mapbox GL JS visualizes shared data and maps. They underlined cloud computing's scalability, cost-effectiveness, and infrastructure-free features. Serverless computing makes blockchain applications more scalable, cost-effective, manageable, and secure, making it a good alternative for developers building and deploying decentralized solutions.

Moreover, Huang et al. [25] noted that HCloud is a trusted Joint Cloud platform for IoT systems, utilizing serverless computing and blockchain technology. It enabled the deployment of several serverless functions on various clouds by a scheduling policy provided by the customer. For scheduling, HCloud provides uniform interfaces, transformation methods, and a single cloud broker. Golec et al. [26] introduced AIBLOCK, a blockchain-based framework integrated with Google Cloud Platform (GCP)–Cloud Functions for ensuring data integrity in AI-based COVID-19 diagnostic studies. The framework integrates serverless computing and blockchain technology to analyze health data obtained from IoT devices, identify instances of COVID-19 through machine learning models, and mitigate unauthorized interventions. The study assesses various machine learning models and showcases the supremacy of serverless computing. The performance results demonstrate the efficacy of AIBLOCK in IoT applications.

Yussupov et al. [27] introduced serverless architectures that highlighted the advantages of using provider-managed components like function-as-a-service (FaaS) and database-as-a-service (DBaaS) to lessen maintenance requirements for developers. The study considered the decentralized, immutable, and accountable interactions of smart contracts and blockchain technology to examine how they may be incorporated into serverless architectures. Benedict [28] proposed a serverless blockchain-enabled IIoT architecture for societal applications, specifically focusing on air quality monitoring for smart cities. It draws attention to the energy inefficiency and resource underutilization problems in current IIoT systems and emphasizes the benefits of leveraging blockchain and serverless computing. The secure transfer of air quality sensor data through edge, fog, and cloud computing layers is covered in the article [29]. Additionally, it offers a list of effective serverless operations for IIoT applications and predicts that this architecture will stimulate further advancements and study in the area.

The authors of [30–32] highlighted the underutilization of personal computers worldwide, with a typical CPU utilization of only 24.5% and an average usage of 4.5 h daily. The authors suggested ChainFaaS, an open blockchain-based serverless platform to utilize this unused processing capacity. ChainFaaS aims to enhance the developer experience while lowering expenses, boosting transparency, and ensuring dependability. The prototype, which uses Hyperledger Fabric for decentralized management and shows the platform's viability with acceptable performance, can potentially lessen the need for data centers. On GitHub, the source code and documentation are accessible.

Overall, a blockchain-assisted serverless framework for geospatial data offers significant advantages by combining the decentralized security and transparency of blockchain with the scalability and cost-effectiveness of serverless computing. This framework ensures immutable audit trails of geospatial data transactions, enhancing data integrity and trustworthiness. Serverless computing enables dynamic scaling of computational resources based on demand, optimizing performance for geospatial applications that require varying levels of processing power. Decentralized authentication and authorization mechanisms provided by blockchain enhance security, allowing for secure sharing and access control of sensitive geospatial data across multiple stakeholders. Additionally, the event-driven nature of serverless architectures aligns well with geospatial data workflows, enabling real-time processing and analysis of location-based information.

### 3. Description of Proposed Framework

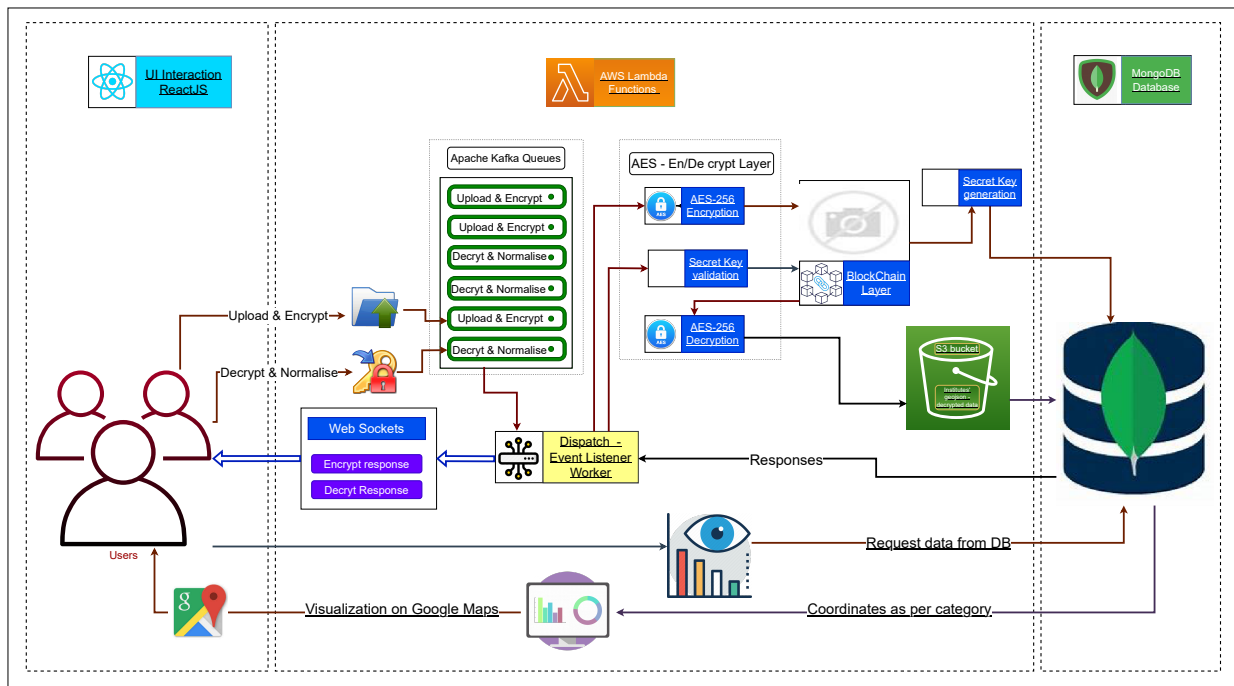
This research paper focuses on implementing data storage in the blockchain using the AWS Lambda serverless blockchain platform with AES encryption, decryption layer, and queue implementation. The objective is to develop a secure and scalable solution for storing and retrieving data in a blockchain network.

To achieve this, a web application was developed as an example to showcase the functionality. The application allows users to upload a CSV file containing geospatial coordinate data of institutes. The data are then encrypted and uploaded to the blockchain, and a corresponding encryption key is obtained. In the next step, the key is provided to extract the files from the blockchain, decrypt them, normalize the data, and store it in MongoDB, a NoSQL database.

The web application includes a visualization tab displaying the coordinates on a Google Map, fetched from the MongoDB database. Queueing methods are employed while fetching data from the blockchain to ensure scalability for larger files. The proposed model leverages the AWS Lambda serverless blockchain platform for its implementation. This platform provides a scalable and managed infrastructure for deploying blockchain applications. The AES encryption and decryption layer ensures the security and confidentiality of the data stored in the blockchain. The queue implementation facilitates efficient data retrieval from the blockchain, allowing for seamless scalability. This section of the research paper will present a detailed description of the proposed model, outlining the architectural components, workflow, and mathematical models associated with each system layer. The goal is to provide a comprehensive understanding of the proposed solution and its underlying mechanisms, enabling readers to replicate and further explore the implementation for data storage in the blockchain using the AWS Lambda serverless blockchain platform with AES encryption, decryption layer, and queue implementation. The proposed framework is presented in Figure 1.

#### 3.1. Client Layer

The client layer is the primary interface that enables users to interact with the system. Users upload CSV files through the front-end interface, initiating a sequence of events that involve data processing, encryption, storage, and visualization. It provides a user-friendly interface for exploring and interacting with the processed data. Users can select specific categories to retrieve relevant data from the MongoDB database. The retrieved data, containing coordinates, are then visualized on a Google Maps screen. This layer bridges the gap between raw data and meaningful insights, enhancing user experience and facilitating data-driven decisions.



**Figure 1.** Proposed framework: BS-GeoEduNet 1.0.

### 3.2. Serverless Computing Layer

The serverless computing layer, discussed in the introduction, is a crucial component of the proposed system. It consists of multiple sublayers that contribute to the overall functionality of the architecture. The serverless layer constitutes the heart of the system's functionality. It comprises a network of AWS Lambda functions orchestrated to perform specific tasks seamlessly. These functions are designed to handle various aspects of the data processing pipeline. The AES encryption and decryption functions ensure data security. At the same time, the Apache Kafka queuing mechanism manages the incoming requests, ensuring orderly processing and the IPFS blockchain layer is responsible for storing files. A cloud computing execution model called serverless computing, or function-as-a-service (FaaS), allows the cloud provider to dynamically handle the allocation and provisioning of servers. Under this architecture, capacity planning, server scalability, and maintenance are all automatically handled by the cloud provider. The phrase "serverless" refers to the fact that the developer or organization does not expressly need to handle the physical or virtual servers; it does not imply that there are no servers involved. In summary, serverless computing represents a paradigm shift in cloud computing by offering modern application development and deployment more agility, cost-effectiveness, scalability, and reduced operational complexity. Applying serverless computing to blockchain technology in geospatial learning networks has several benefits. Because of its scalability, which adjusts resources automatically based on workload, blockchain applications can effectively handle a wide range of transaction volumes and computing needs. This elasticity prevents over-provisioning during times of low activity, which reduces expenses while also optimizing performance. Serverless platforms allow developers and administrators to concentrate on creating and implementing blockchain applications instead of handling servers by abstracting away the complexity of infrastructure management. Serverless architectures are event-driven, which fits very nicely with blockchain's need for real-time updates and interactions. This allows for the easy integration of instructional content and geospatial data amongst decentralized nodes. Additionally, serverless computing fosters interoperability with external systems, enables high availability and fault tolerance, and improves development agility, all of which are essential for sustaining dependable and effective blockchain-powered geospatial educational networks.

### 3.2.1. Apache Kafka Queueing

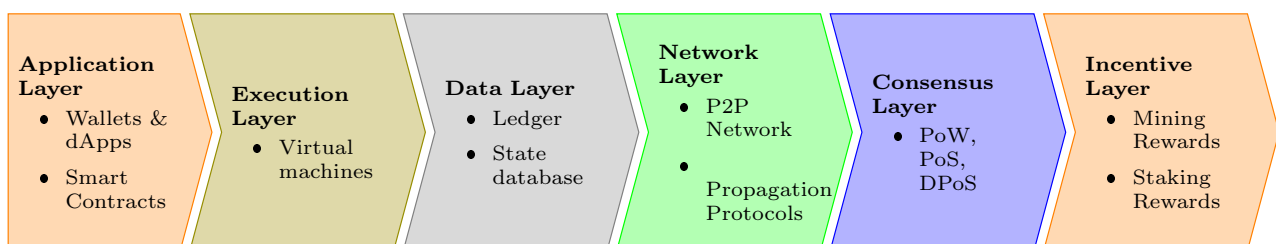
This function manages incoming requests, ensuring they are queued for orderly processing. It prevents overwhelming the system and helps manage resource allocation effectively. It helps to handle large requests effectively.

### 3.2.2. IPFS Blockchain Layer

The IPFS layer represents the integration of a blockchain platform into the proposed system. In this case, the IPFS system is run on an off-chain network, which is used for file storage. This layer ensures secure and decentralized data storage, offering immutability and data integrity. Data are distributed and decentralized when kept in a blockchain. A closer look at the storing mechanism is provided here:

- **Blocks:** Details are arranged in blocks. A list of transactions, a timestamp, a nonce, and the hash of the previous block are all included in each block.
- **Chain of blocks:** The blocks are sequentially connected. A chain is created when every new block includes the hash of the one before it.
- **Distributed ledger:** A network of nodes maintains the blockchain ledger. A copy of the complete blockchain is stored on each node.
- **Merkle trees:** Transactions are arranged in a Merkle tree structure within each block. This makes it possible to verify the block's transactions quickly and securely.

Verification on a blockchain ensures the trustworthiness and consistency of data. A blockchain system's architecture is made up of multiple layers and parts. It has an application layer, execution layer, data layer, network layer, consensus layer, and incentive layer. The application layer communicates with the final user. It comprises programs like smart contracts, decentralized applications, and wallets. The execution layer is responsible for smart contracts and transactions. Virtual machines execute smart contract codes. The data layer keeps the blockchain's information. This comprises the state database, which keeps track of the blockchain's current status, and the ledger, which houses all transactions and blocks. The network layer oversees inter-node communication. Nodes propagate blocks and transactions while upholding consensus. The consensus layer ensures that distributed nodes agree on the state of the blockchain. Different consensus techniques, including PoW, PoS, or DPoS, are used by various blockchains. The incentive layer offers financial rewards to nodes for truthful network participation. Miners who solve puzzles in PoW are rewarded with cryptocurrency. According to the coins they have staked, validators in PoS receive rewards. A condensed representation of the blockchain architecture is shown in Figure 2.



**Figure 2.** Layered structure of the blockchain architecture.

### 3.2.3. AES Encryption and Decryption Layer

When it comes to avoiding unauthorized access to sensitive information, data encryption is absolutely necessary. An encryption and decryption layer is incorporated to enhance the security and confidentiality of the data stored in the blockchain. When it comes to developing a decentralized and unchangeable coordinate system, the encryption and decryption algorithms of the advanced encryption standard (AES) offer the most effective option. AES algorithms encrypt the data before uploading the data to the blockchain. Similarly, decryption algorithms retrieve and decrypt the data when necessary.



### 3.2.4. S3-Storage Service Interaction

This function handles file uploads and downloads to and from Amazon S3. It ensures reliable and scalable storage for encrypted files.

### 3.2.5. Dispatch Worker

At the heart of the processing pipeline, this function coordinates the journey of user data. It manages the entire sequence, orchestrating S3 uploads, MongoDB updates, IPFS interactions, and AES encryption/decryption.

### 3.3. Database Layer

The MongoDB database layer forms the foundation for data storage and retrieval. It consists of collections tailored to different data stages. The first collection stores details about processed files, including metadata and IPFS hashes. The second collection holds the normalized JSON structures of decrypted data. These collections enable efficient data management, retrieval, and analysis.

## 4. Queueing Based Analytical Approach

Estimating relationships between parameters that describe the task inputs, processing, and system architecture is necessary. The design parameters are the queue length, server speed, mean task completion time, and number of processing elements. These, in succession, are based on entry arrival rates, the rate at which requests in the process of completion result in new fees, and the penalty related to the loss of requests due to queue overruns. For more details, one may refer to the work by Mott and Arabadjis [33]. Figure 3 describes the queueing analytical approach for the proposed framework.

The model comprises  $r$  parallel processors connected by a usual queue that can hold a maximum of  $D$  requests ( $D$  is the queue buffer size). The completion of a request is achieved by assigning it to a processor. It is assumed that the service time is distributed exponentially with a mean rate of  $1/\mu_f$ . After the task is released upon completion of service, the processor processes the next job in the queue. The processor becomes idle if the queue is empty. If a job is completed, the system may exit or generate another job sent to the queue's finish line with the probability of  $\varphi$ . A task will be lost if it encounters a full queue. The system assumes that external arrivals are Poisson-distributed and have a mean input rate  $\lambda_f$ . The processors process a mean of  $\Lambda$  (throughput), and  $\Lambda\varphi$  is the feedback arrival rate. The total request rate of the system is determined by the addition of external and feedback requests,  $\lambda_f + \Lambda\varphi$ . As  $\Lambda$  requests enter the queue, requests are lost at a rate of  $\lambda_f + \Lambda\varphi - \Lambda$  requests. There is an assumption that the quantities  $\lambda_f$ ,  $\mu_f$ , and  $\varphi$  are already known. The queue must be empty if there are idle processors, as per the mechanism mentioned earlier. In contrast, if the queue is not vacant, all servers remain busy. The system buffer size is  $K = D + r$  because a complete set-up includes  $D$  in the queue and  $r$  in the processing (which means all servers are used).

Let  $N_s(m)$  represent the number of requests in the blockchain system at time  $m$ . In the same manner, let  $N_q(m)$  and  $N_b(m)$  represent the number of requests in the queue and the number of requests in the processing at time  $m$ . Then we have the following:  $N_s(m) = N_q(m) + N_b(m)$

$$N_q(m) = \begin{cases} 0, & \text{for } 0 \leq N_s(m) \leq r \\ N_s(m) - r, & \text{for } r \leq N_s(m) \leq K \end{cases}$$

and

$$N_b(m) = \begin{cases} N_s(m), & \text{for } 0 \leq N_s(m) \leq r \\ r, & \text{for } r \leq N_s(m) \leq K \end{cases}$$

The probability that there are requests for  $\ell$  in the system at time  $m$  is defined as  $P_\ell(m) = P[N_s(m) = \ell]$ . At a steady state,  $P_\ell = \lim_{m \rightarrow \infty} P_\ell(m)$ . Applying the probabilistic approach, we obtain the following:

$$\begin{aligned} \lambda_f P_0 &= \mu_f(1 - \varphi)P_1, \quad n = 0, \\ (\lambda_f + \ell\mu_f(1 - \varphi))P_\ell &= \lambda_f P_{\ell-1} + (\ell + 1)\mu_f(1 - \varphi)P_{\ell+1}, \\ & \quad 1 \leq \ell \leq r - 1, \\ (\lambda_f + r\mu_f(1 - \varphi))P_\ell &= \lambda_f P_{\ell-1} + r\mu_f(1 - \varphi)P_{\ell+1}, \\ & \quad r \leq \ell \leq K - 1, \\ r\mu_f(1 - \varphi)P_K &= \lambda_f P_{K-1} \end{aligned}$$

Solving the above equations for  $P_\ell$ , we obtain the following:

$$P_\ell = \begin{cases} \frac{\xi^\ell}{\ell!} P_0, & 1 \leq \ell \leq r - 1 \\ \frac{\xi^\ell}{r^{\ell-r} r!} P_0, & r \leq \ell \leq K \end{cases}$$

where  $\xi = \frac{\lambda_f}{\mu_f(1-\varphi)}$  is the system traffic intensity. Finally, using the normalization condition  $\sum_{\ell=0}^K P_\ell = 1$ , we obtain the following:

$$P_0 = \left[ \sum_{\ell=0}^r \frac{\xi^\ell}{\ell!} + \frac{\xi^{r+1}}{r!(r-\xi)} \left\{ 1 - \left( \frac{\xi}{r} \right)^D \right\} \right]^{-1}. \tag{1}$$

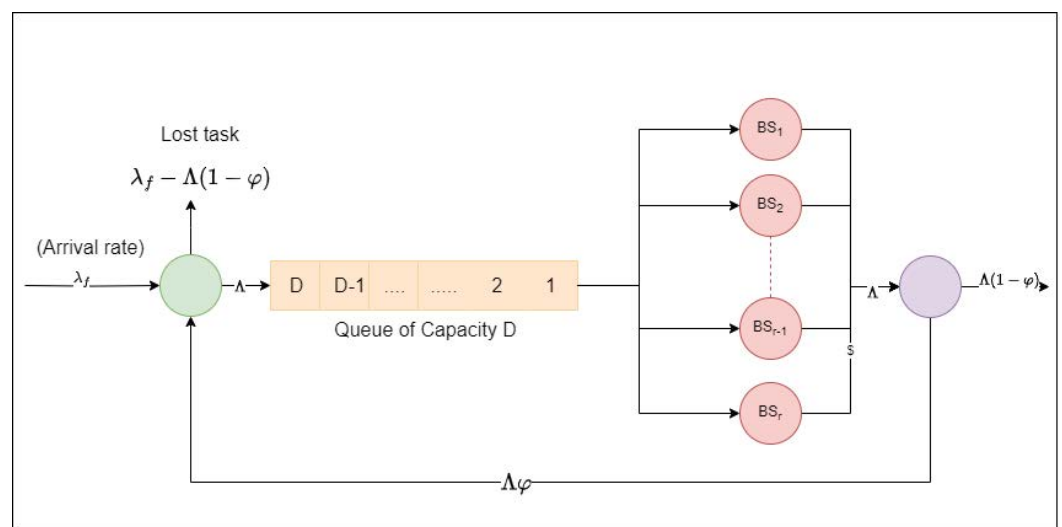


Figure 3. Queueing analytical approach for the proposed framework.

#### 4.1. Performance Analysis

The system model’s performance measures can be described as follows:

- The average number of requests in the queue and the average number in processing should be referred to as  $L_q$  and  $E(B)$ , respectively. These are provided by the following:

$$\begin{aligned} L_q &= \sum_{\ell=r+1}^K (\ell - r)P_\ell \\ E(B) &= \sum_{\ell=0}^r \ell P_\ell + r \sum_{\ell=r+1}^K P_\ell \end{aligned}$$

- The utilization factor of each server is  $\rho = \frac{\lambda_f}{\mu_f r (1-\varphi)}$ .
- The mean number of tasks in the blockchain system is

$$L_s = \sum_{\ell=0}^K \ell P_\ell = L_q + E(B)$$

- The throughput, called  $\Lambda$ , represents the average number of requests served per second, which is as follows:

$$\Lambda = \mu_f E(B)$$

- The mean time a request takes to go through the blockchain system is known as the average sojourn time ( $W_s$ ). The wait time in queue,  $W_q$ , and the processing time ( $1/\mu_f$ ) are the components of  $W_s$ ;

$$W_s = W_q + \frac{1}{\mu_f}$$

- We can find  $W_s$  using Little's theorem. The system should have an average of the same number of requests ( $L_s$ ) for an arriving request as it leaves upon departing. The throughput of  $\Lambda$  times  $W_s$  is the only thing that matters, resulting in the following:

$$W_s = \frac{L_s}{\Lambda} = \frac{1}{\mu_f} + \frac{L_q}{\mu_f}$$

- The total arrival input rate and the lost task rate are  $\lambda_f + \Lambda\varphi$  and  $\lambda_f + \Lambda\varphi - \Lambda$ , respectively.
- The ratio between the lost request rate and total input arrival rate is known as  $P_L$ , which is the fraction of lost requests. That is,

$$P_L = \frac{\lambda_f + \Lambda\varphi - \Lambda}{\lambda_f + \Lambda\varphi}$$

#### 4.2. Numerical and Simulation Examples

To illustrate the results found by the presented queueing model, we use MAPLE 2022 software. Figure 4 depicts an effect of traffic intensity per processor on the normalized waiting time  $\mu_f W_s$  (that is, the waiting time measured in terms of the mean service time). The queue capacity and number of processors are shown by the illustration on each curve. The 16/8 curve is intended for an eight-processor system that is fed by a capacity queue of 16. Each curve is close to an asymptotic upper bound for large  $\rho$ . It should be noted that enhancing  $D$  and  $r$  leads to a reduction in the waiting time for minuscule  $\rho$  and an increase in the waiting time for reasonably larger  $\rho$ . The reason for this increase is that there are fewer job losses as the queue buffer size increases proportionately.

Figure 5 illustrates the impact of traffic intensity per processor on the fraction of processor busy time,  $E(B)/r$ , for various values of  $D/r$ . The arcs are largely a function of the ratio, but they also have distinct functions of  $D$  and  $r$ . The curves show that the percentage of busy servers increases as the number of servers increases, as expected. The longer queue stores requests that would be lost with a shorter queue as a buffer.

Figure 6 presents the utilization factor per server versus the fraction of requests lost. Note that the fraction of requests lost is susceptible to the queue buffer size. Increasing the queue buffer size reduces  $P_L$  for small values of  $\rho$ . For higher values of  $\rho$ , the value of  $P_L$  is almost static.

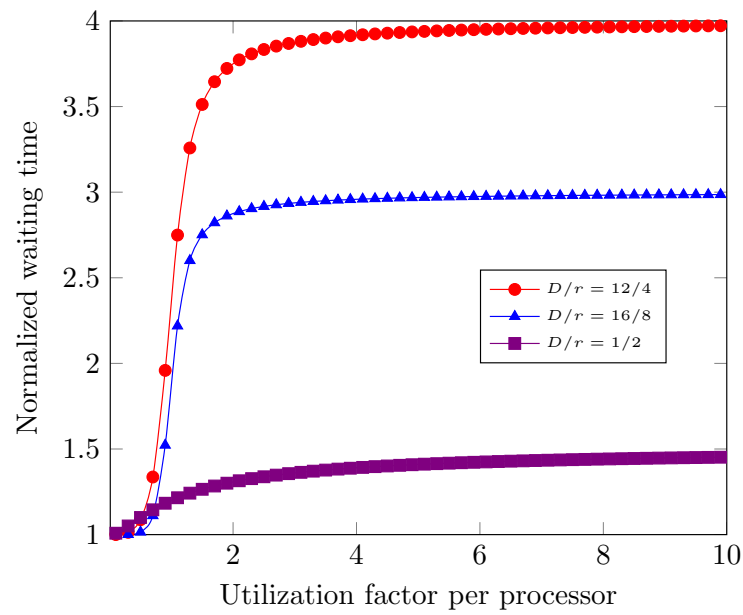


Figure 4. Utilization factor per processor vs. normalized waiting time.

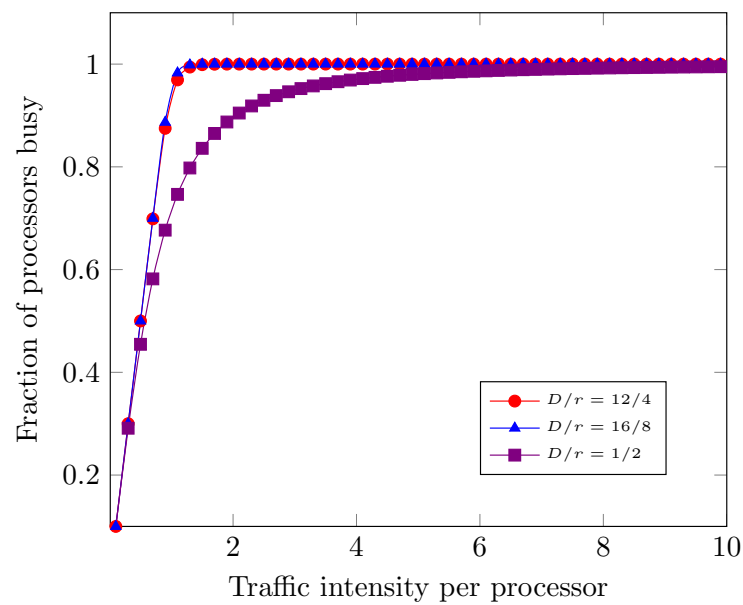


Figure 5. Utilization factor per processor vs. fraction of processor busy time.

Figures 7 and 8 illustrate the impact of  $\varphi$  and  $r$  on  $W_s$  and  $E(B)$ , respectively. From Figure 7, we note that the waiting time for fixed  $\varphi$  decreases as the number of processors  $r$  increases. Further, with fixed  $r$ , the waiting time increases when the  $\varphi$  increases. From Figure 8, we observe that for fixed  $\varphi$ , the  $E(B)$  is almost static when the number of processors  $r$  increases. Further, with fixed  $r$ ,  $E(B)$  increases when  $\varphi$  increases. Thus, we can set up a permissible number of processor  $r$  and the sufficient value of  $\varphi$  in the system to lower  $W_s$  and to keep  $E(B)$  balance.

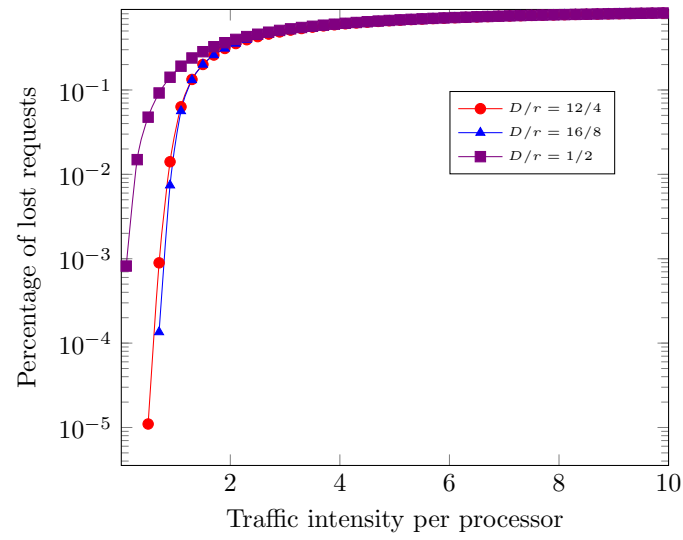


Figure 6. Utilization factor per processor vs. fraction of requests lost.

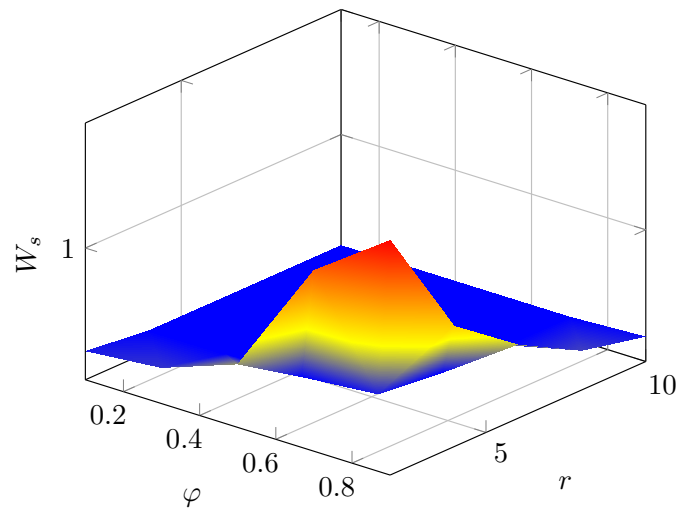


Figure 7.  $\varphi$  vs. processor vs.  $W_s$ .

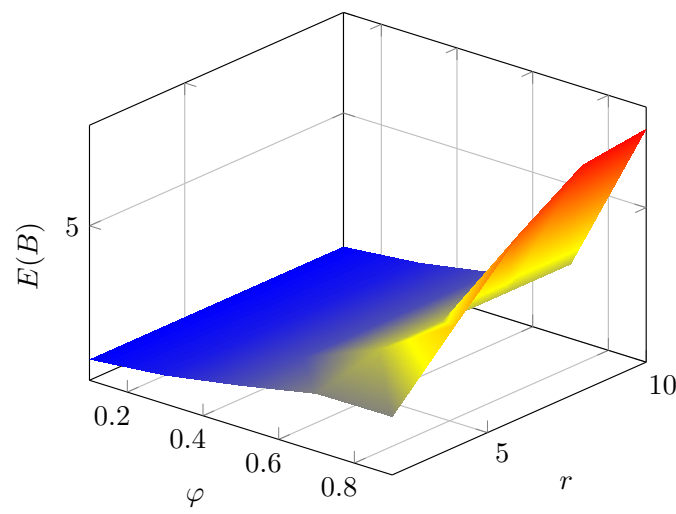


Figure 8.  $\varphi$  vs. processor vs.  $E(B)$ .

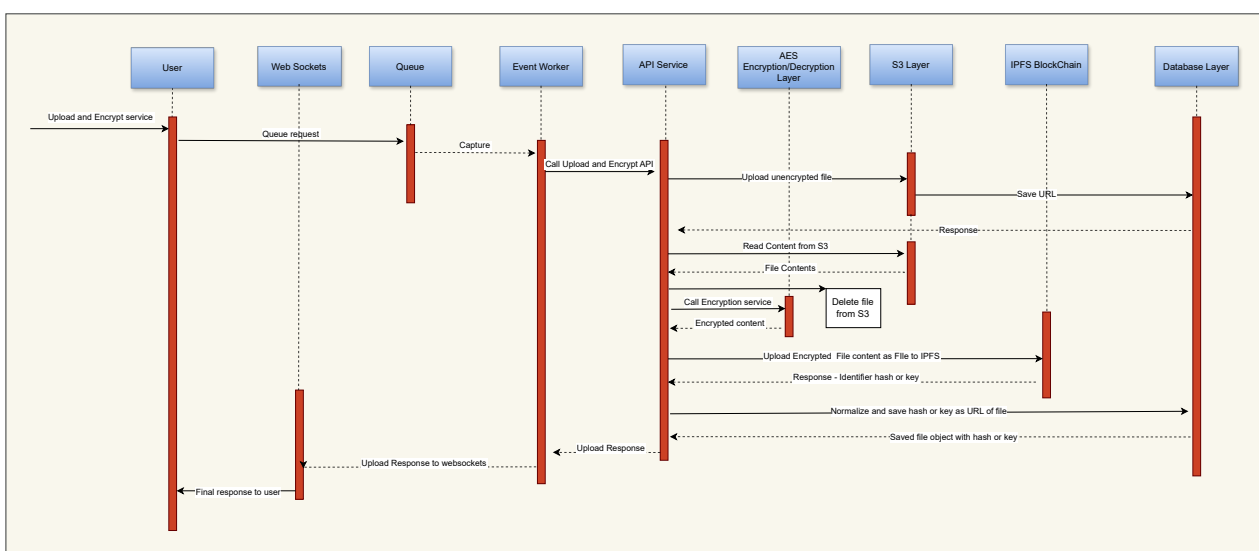
## 5. Case Study: Geospatial Educational Information Network

Institutions need efficient tools to manage, analyze, and visualize location-based information in an increasingly data-driven educational landscape. To address this need, our team embarked on a project to develop a comprehensive system that allows users to upload CSV files containing institute coordinates and relevant information, such as founding dates and categories like IITs, NITs, IIITs, or other educational institutes. This system allows users to visualize the exact locations of institutes on a map based on selected categories, facilitating data-driven decision-making and enhancing geographical insights.

The Geospatial Educational Information Network case study demonstrates the necessity of blockchain technology to ensure data integrity, security, and transparency. By leveraging blockchain, we create an immutable record of geospatial data, crucial for maintaining the accuracy and trustworthiness of educational information. Traditional methods lack these robust security features, making blockchain an essential component of our framework. This robust security ensures that the data remain tamper-proof and verifiable, which is vital for educational institutions. The challenge was to design and implement a seamless data processing pipeline that starts with user-generated CSV files, progresses through data encryption, storage, and normalization, and ends with interactive visualization. The system must ensure data security, efficient processing, and an intuitive user experience. Moreover, it needed to allow users to filter data based on categories and retrieve location-based insights.

The developed system consists of the client layer, serverless layer, and database layer. Each layer plays a crucial role in the end-to-end process, and there is the proper mechanism for the entire process having a set of data in each step of the flow, which can be broken down primarily into three functionalities and are as follows:

A user initiates the process in this flow by uploading a CSV file through the frontend interface. The uploaded file undergoes a series of orchestrated actions in the serverless layer. The file is queued using Apache Kafka for organized processing. A dispatch worker manages the queued requests, sequentially executing various tasks. These tasks include uploading the file to Amazon S3 for secure storage, saving essential details in MongoDB, encrypting the file's contents using AES encryption, and finally, uploading the encrypted file to the IPFS off-chain system. An IPFS hash or key is received and stored in the MongoDB database, marking the completion of this phase. Figure 9 illustrates the sequence diagram for functionality 1 where the user uploads CSV and data processing.



**Figure 9.** Sequence diagram for functionality 1: User uploads CSV and data processing. This diagram illustrates the steps involved in the user uploading a CSV file and the subsequent data processing steps within the system.

Users view the list of encrypted files available to them through the front end. Users have the option to select specific files for decryption, normalization, and subsequent storage in the database. Upon user selection, Lambda functions are triggered in the serverless layer. These functions follow a similar pattern as before, utilizing Apache Kafka for queue management and a dispatch worker for processing. The dispatch worker retrieves the IPFS hash from MongoDB, uses it to fetch the encrypted file from IPFS, and then decrypts the file’s contents using AES decryption. The decrypted data are normalized into a JSON structure, and the normalized data are saved in a separate MongoDB collection. Figure 10 describes the sequence diagram for functionality 2 where user decryption and associate normalization process has performed.

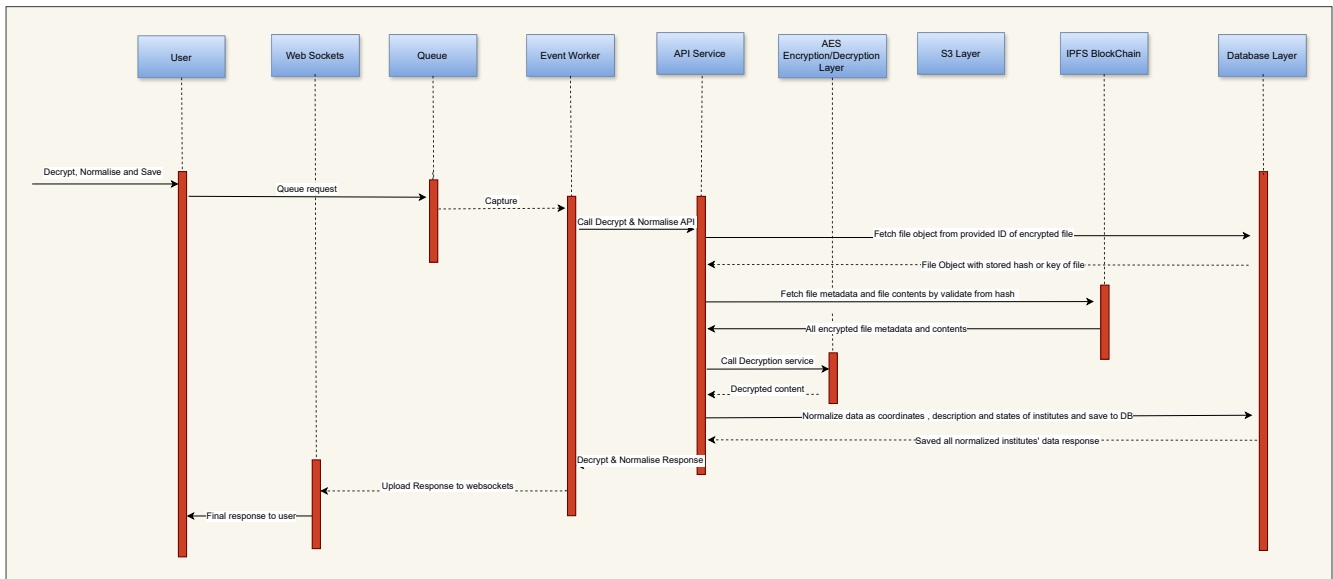


Figure 10. Sequence diagram for functionality 2: user decryption and normalization.

This flow focuses on providing users with a visual representation of the data. Users interact with the front end to select a category of institutes they are interested in. The front end communicates with the database layer to retrieve relevant data based on the chosen category. The retrieved data typically include institute coordinates and associated information. This dataset is then passed to the Google Maps API, which processes the coordinates and displays them on a map interface. Users can now visualize the locations of institutes based on their chosen category, enabling them to gain geographical insights and make informed decisions. Figure 11 demonstrates the sequence diagram for functionality 3 with data visualization in the proposed framework. Figure 12 represents the overall sequence diagram for the overall functionality of the proposed BS-GeoEduNet 1.0 framework.

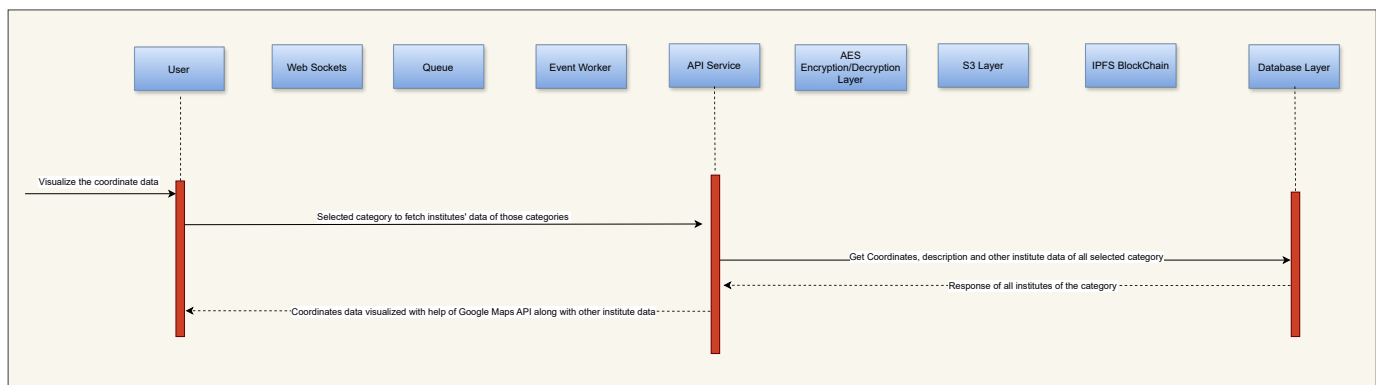


Figure 11. Sequence diagram for functionality 3: data visualization.

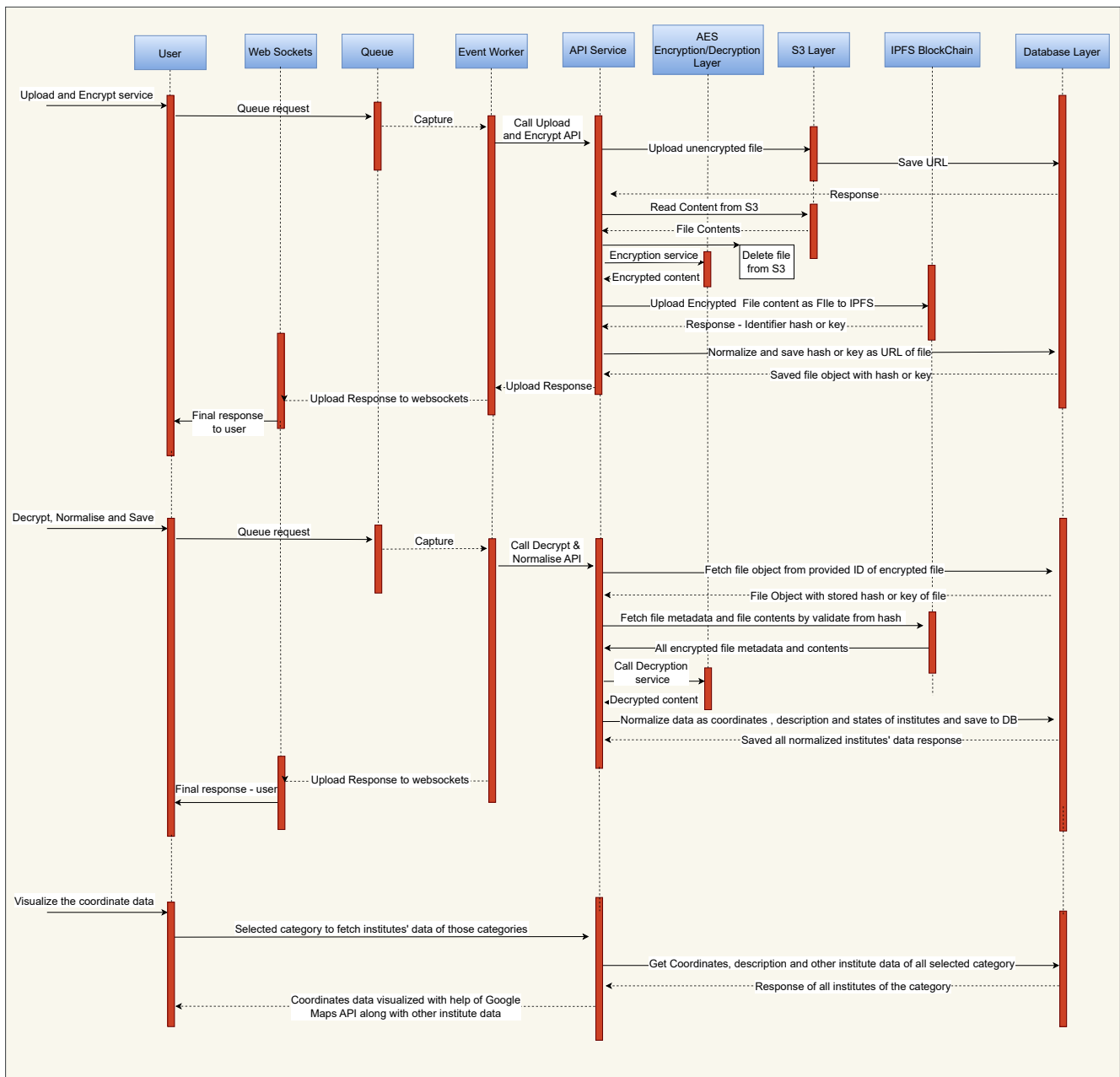


Figure 12. Sequence diagram for functionality of BS-GeoEduNet 1.0.

### 6. Result and Discussions

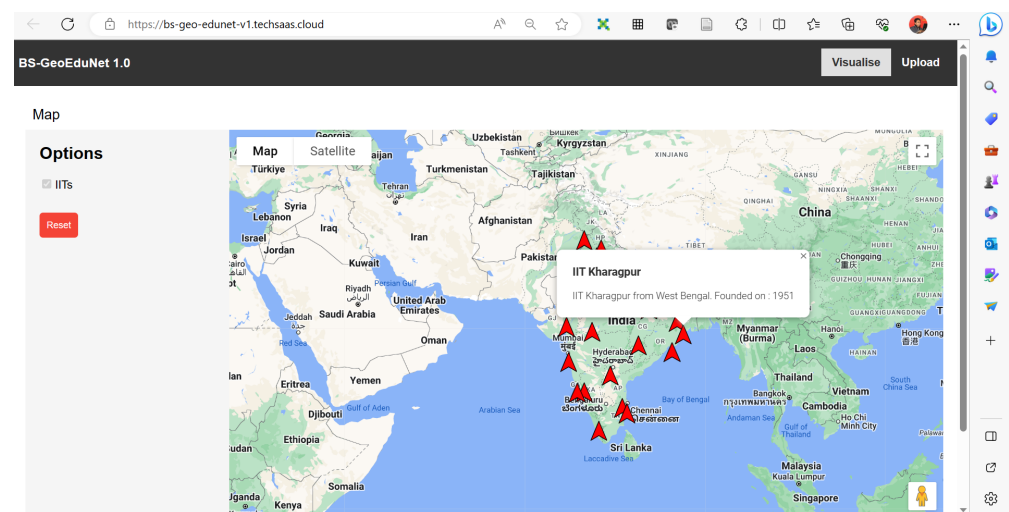
The successful implementation of the visualization system has yielded highly favorable results across various components of the proposed model. Through rigorous testing and evaluation, the system has demonstrated its ability to securely process, store, and visualize location-based data from user-uploaded CSV files. The proposed framework can be accessed at <https://bs-geo-edunet-v1.techsaas.cloud/>, last accessed on 12 May 2024.

The benefits of the proposed framework are as follows:

1. **Efficient data processing:** The utilization of AWS Lambda functions in a serverless architecture has proven highly efficient in processing incoming requests. The incorporation of the Apache Kafka queueing mechanism ensures streamlined request management and effective resource allocation, leading to optimal performance. Performance tests indicate that the system maintains low latency and high throughput even during peak usage periods.



2. **Robust encryption and decryption:** The AES encryption and decryption functions have successfully bolstered data security throughout the processing pipeline. Uploaded files are encrypted before storage on IPFS, ensuring confidentiality. Rigorous testing has validated the accuracy of decryption and normalization processes during user-selected file retrieval, affirming the system's reliability.
3. **Effective data storage and retrieval:** The MongoDB database layer has exhibited consistent and reliable data storage and retrieval capabilities. The system's design, featuring distinct collections for encrypted file details and normalized data structures, facilitates efficient data organization and retrieval. Performance benchmarks indicate swift response times, showcasing the database layer's efficiency even with substantial datasets.
4. **Intuitive data visualization:** The integration of the data visualization layer with the Google Maps API has created a seamless and intuitive user experience. Users can effortlessly interact with the interface to visualize institute locations based on selected categories. This accurate representation of geographical data on the map interface improves user comprehension and decision-making as shown in Figure 13.



**Figure 13.** Geospatial data visualizations show the total number of IITs in India with the help of *BS-GeoEduNet 1.0*.

As the educational sector becomes more dependent on digital infrastructure, the integration of blockchain technology and serverless computing for geographic data visualization is becoming more and more significant. This innovative combination provides a scalable, affordable, and safe way to manage and display geographic data in educational environments. When combined, these technologies provide accurate geospatial information in real-time and improve learning through dynamic, interactive representations that may be used in a wide range of fields, from urban planning to environmental research. Leveraging blockchain and serverless computing for geospatial data visualization not only modernizes the educational process but also equips students for a technologically advanced world, as digital infrastructure becomes increasingly essential to educational institutions. The successful development and implementation of *BS-GeoEduNet 1.0* highlights several noteworthy achievements and areas of consideration.

1. **Scalability and performance potential:** The chosen serverless architecture, coupled with the Apache Kafka queueing mechanism, establishes a robust foundation for system scalability. The architecture's ability to handle increased workloads efficiently positions the system for future growth. This scalability, combined with consistently low latency, guarantees a satisfying user experience regardless of user volume.
2. **Data security and user confidence:** The integration of AES encryption and decryption mechanisms contributes significantly to data security along with storage in the off-

chain blockchain network of IPFS. By adopting a stringent approach to encryption, the system safeguards sensitive institute information from unauthorized access. This security provision cultivates user trust and confidence in the system's ability to handle and protect their data.

3. **User-centric design:** The frontend interface's user-centric design ensures ease of use and accessibility. Users can effortlessly upload CSV files, select categories, and interact with the map interface. This user-friendly approach facilitates seamless interaction, enabling users to derive valuable insights from the visualized data.
4. **Potential for future advancements:** The success of the system sets the stage for potential future enhancements. These could include integration with advanced machine learning algorithms for predictive analysis or incorporating real-time data updates to provide users with the most current information. Exploring alternative decentralized storage solutions beyond IPFS could further enhance data availability and resilience.

However, exact similar integrations have not existed until now, and the contributions differ from previous research. Table 1 compares the existing models with our proposed model. The authors of [11,34] stated that IoT devices have limited resources, making them incapable of executing large computational operations. To address this issue, the serverless paradigm, with its benefits such as dynamic scalability and infrastructure management, can be utilized to meet the requirements of IoT-based applications. However, due to the heterogeneous structure of IoT, user trust must also be considered when implementing this integration. This problem can be overcome by using a blockchain that guarantees data immutability and ensures that any data generated by the IoT device is not modified. The authors propose a BlockFaas framework that supports dynamic scalability and guarantees security and privacy by integrating a serverless platform and blockchain architecture into latency-sensitive artificial intelligence (AI)-based healthcare applications. Also, Singh [24] presents a new approach to managing real-time electricity data in smart grids. The author introduced a federated learning model that ensures data privacy by using serverless cloud computing and local blockchain-enabled servers in home area networks. The model reduces costs, lowers the risk of cyberattacks, and improves accuracy in energy requirement predictions for smart grids. Benedict [28] introduced a novel IoT architecture that combines serverless computing and blockchain technology to enhance efficiency, security, and decentralization in societal applications. The author emphasized the importance of serverless computing and blockchain in IoT, with a focus on applications like air quality monitoring in smart cities. The paper showcased how this approach optimizes data transactions and outlines powerful serverless functions for IoT developers, offering innovative solutions for societal benefit. Furthermore, blockchain technology is significant and novel for healthcare platforms, as Mallick et al. [35] showed, as it allows for data privacy, anonymity, and validity through the consensus process. The author suggested BC-Geo, a cutting-edge decentralized geospatial service architecture powered by blockchain technology for intelligent healthcare systems. To address the technical issues in the majority of the present healthcare system, the suggested framework, in contrast to earlier frameworks, provides decentralized privacy protection regulations, immutability, scalability, geospatial mapping, and patient prioritizing. Our proposed framework, *BS-GeoEduNet 1.0*, focuses on creating a scalable and robust data storage system using Amazon Web Services (AWS) Lambda, a serverless computing platform. Traditional data storage methods face challenges in scalability, cost, and responsiveness, whereas AWS Lambda dynamically allocates resources, eliminating the need for manual infrastructure management. This approach ensures effortless scalability with increasing data and user demands. By integrating blockchain technology, which is known for its decentralized and immutable nature, the system secures data integrity, transparency, and traceability. Geospatial coordinates of educational institutes are encrypted and stored on the blockchain, ensuring data security and simplifying auditing processes.

**Table 1.** Comparison of existing models with our proposed framework.

Model Used	Blockchain with Serverless Integration	Data Storage	Queuing Approach	Geospatial Domain
AQMS [28]	Yes	No	No	No
BFMLP [24]	Yes	No	No	No
BloMT [11]	Yes	Yes	No	No
BlockFaas [34]	Yes	Yes	No	No
BC-Geo [35]	No	No	Yes	Yes
<i>BS-GeoEduNet 1.0</i>	Yes	Yes	Yes	Yes

## 7. Conclusions and Future Work

In conclusion, the proposed *BS-GeoEduNet 1.0* framework effectively integrates modern technology and innovative design to address educational data management challenges. This research details a comprehensive architecture that empowers educational institutions to efficiently process, secure, store, and visualize location-based data. The system's multi-layered approach, including client, serverless, database, and data visualization layers, ensures intuitive and secure user interaction with geographical data. The framework has shown exceptional performance in rigorous testing, providing timely and accurate data processing under varying workloads. Robust encryption and secure storage mechanisms guarantee data security and user privacy while integrating with Google Maps API enhances data visualization. Future research could further extend the system's capabilities:

- Predictive analysis: Integrate machine learning for predictive insights.
- Enhanced visualizations: Develop advanced visualization techniques like heat maps and clustering.
- Extended data sources: Enrich data by integrating external APIs.
- User collaboration: Enable features for user annotation, commenting, and collaboration.
- Accessibility and usability: Improve the front-end interface for better user experience and inclusivity.

The successful implementation of this system demonstrates the potential of cutting-edge technologies in educational data management, offering a foundation for future research and innovation.

**Author Contributions:** Conceptualization, Meenakshi Kandpal, Veena Goswami, Yash Pritwani, Rabindra K. Barik and Manob Jyoti Saikia; methodology, Meenakshi Kandpal, Veena Goswami, Yash Pritwani and Rabindra K. Barik; software, Meenakshi Kandpal, Veena Goswami and Yash Pritwani; validation, Veena Goswami and Manob Jyoti Saikia; formal analysis, Meenakshi Kandpal and Yash Pritwani; investigation, Veena Goswami and Rabindra K. Barik; resources, Manob Jyoti Saikia; data curation, Meenakshi Kandpal and Yash Pritwani; writing—original draft preparation, Meenakshi Kandpal, Veena Goswami, Yash Pritwani, Rabindra K. Barik and Manob Jyoti Saikia; writing—review and editing, Manob Jyoti Saikia; visualization, Meenakshi Kandpal and Yash Pritwani; supervision, Veena Goswami and Manob Jyoti Saikia; project administration, Manob Jyoti Saikia. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author.

**Acknowledgments:** The authors thank the Biomedical Sensors & Systems Lab, University of North Florida, Jacksonville, FL 32224, USA for supporting this research and the article processing charges.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Arutyunyan, M.; Berestovskyy, A.; Bratschi-Kaye, A.; Degenbaev, U.; Drijvers, M.; El-Ashi, I.; Kaestle, S.; Kashitsyn, R.; Kot, M.; Pignolet, Y.A.; et al. Decentralized and Stateful Serverless Computing on the Internet Computer Blockchain. In Proceedings of the 2023 USENIX Annual Technical Conference (USENIX ATC 23), Boston, MA, USA, 10–12 July 2023; pp. 329–343.
2. Ghorbian, M.; Ghobaei-Arani, M. A Blockchain-Enabled Serverless Security Mechanism for IoT-Based Drones. In *Building Cybersecurity Applications with Blockchain and Smart Contracts*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 55–82.
3. Chellasamy, N.; Akram, F.; Rajesh, N.; Umamaheswari, R. Application of Serverless Computing in Blockchain Distributed Technology. In Proceedings of the 2023 International Conference on Disruptive Technologies (ICDT), Greater Noida, India, 11–12 May 2023; pp. 115–118.
4. Gurzhii, A.; Islam, A.K.M.N.; Haque, A.K.M.B.; Marella, V. Blockchain Enabled Digital Transformation: A Systematic Literature Review. *IEEE Access* **2022**, *10*, 79584–79605. [[CrossRef](#)]
5. Almaiah, M.A.; Al-Khasawneh, A. Investigating the main determinants of mobile cloud computing adoption in university campus. *Educ. Inf. Technol.* **2020**, *25*, 3087–3107. [[CrossRef](#)]
6. Qasem, Y.A.M.; Abdullah, R.; Jusoh, Y.Y.; Atan, R.; Asadi, S. Cloud Computing Adoption in Higher Education Institutions: A Systematic Review. *IEEE Access* **2019**, *7*, 63722–63744. [[CrossRef](#)]
7. Wright, M.A. The evolution of the Advanced Encryption Standard. *Netw. Secur.* **1999**, *1999*, 11–14. [[CrossRef](#)]
8. Liu, X.; Sun, S.X.; Huang, G. Decentralized services computing paradigm for blockchain-based data governance: Programmability, interoperability, and intelligence. *IEEE Trans. Serv. Comput.* **2019**, *13*, 343–355.
9. Tomar, A.; Gupta, N.; Rani, D.; Tripathi, S. Blockchain-assisted authenticated key agreement scheme for IoT-based healthcare system. *Internet Things* **2023**, *23*, 100849. [[CrossRef](#)]
10. Malik, H.; Anees, T.; Faheem, M.; Chaudhry, M.U.; Ali, A.; Asghar, M.N. Blockchain and Internet of Things in Smart Cities and Drug Supply Management: Open Issues, Opportunities, and Future Directions. *Internet Things* **2023**, *23*, 100860. [[CrossRef](#)]
11. Khan, A.A.; Wagan, A.A.; Laghari, A.A.; Gilal, A.R.; Aziz, I.A.; Talpur, B.A. BioMT: A state-of-the-art consortium serverless network architecture for healthcare system using blockchain smart contracts. *IEEE Access* **2022**, *10*, 78887–78898. [[CrossRef](#)]
12. Ramzan, S.; Aqdu, A.; Ravi, V.; Koundal, D.; Amin, R.; Al Ghamdi, M.A. Healthcare applications using blockchain technology: Motivations and challenges. *IEEE Trans. Eng. Manag.* **2022**, *70*, 2874–2890. [[CrossRef](#)]
13. Chatterjee, K.; Singh, A.; Neha. A blockchain-enabled security framework for smart agriculture. *Comput. Electr. Eng.* **2023**, *106*, 108594. [[CrossRef](#)]
14. Farnaghi, M.; Mansourian, A. Blockchain, an enabling technology for transparent and accountable decentralized public participatory GIS. *Cities* **2020**, *105*, 102850. [[CrossRef](#)]
15. Ozelik, A.E. Blockchain-oriented geospatial architecture model for real-time land registration. *Surv. Rev.* **2022**, *56*, 1–17. [[CrossRef](#)]
16. Mete, M.O.; Yomralioglu, T. Implementation of serverless cloud GIS platform for land valuation. *Int. J. Digit. Earth* **2021**, *14*, 836–850. [[CrossRef](#)]
17. Li, Y.; Lin, Y.; Wang, Y.; Ye, K.; Xu, C. Serverless Computing: State-of-the-Art, Challenges and Opportunities. *IEEE Trans. Serv. Comput.* **2023**, *16*, 1522–1539. [[CrossRef](#)]
18. Mampage, A.; Karunasekera, S.; Buyya, R. A holistic view on resource management in serverless computing environments: Taxonomy and future directions. *ACM Comput. Surv. (CSUR)* **2022**, *54*, 1–36. [[CrossRef](#)]
19. Shafiei, H.; Khonsari, A.; Mousavi, P. Serverless computing: A survey of opportunities, challenges, and applications. *ACM Comput. Surv.* **2022**, *54*, 1–32. [[CrossRef](#)]
20. Cassel, G.A.S.; Rodrigues, V.F.; da Rosa Righi, R.; Bez, M.R.; Nepomuceno, A.C.; da Costa, C.A. Serverless computing for Internet of Things: A systematic literature review. *Future Gener. Comput. Syst.* **2022**, *128*, 299–316. [[CrossRef](#)]
21. Golec, M.; Ozturac, R.; Pooranian, Z.; Gill, S.S.; Buyya, R. IFaaSBus: A security-and privacy-based lightweight framework for serverless computing using IoT and machine learning. *IEEE Trans. Ind. Informatics* **2021**, *18*, 3522–3529. [[CrossRef](#)]
22. Xie, R.; Tang, Q.; Qiao, S.; Zhu, H.; Yu, F.R.; Huang, T. When serverless computing meets edge computing: Architecture, challenges, and open issues. *IEEE Wirel. Commun.* **2021**, *28*, 126–133. [[CrossRef](#)]
23. Pakdil, M.E.; Çelik, R.N. Serverless geospatial data processing workflow system design. *ISPRS Int. J. Geo-Inf.* **2021**, *11*, 20. [[CrossRef](#)]
24. Singh, P.; Masud, M.; Hossain, M.S.; Kaur, A.; Muhammad, G.; Ghoneim, A. Privacy-preserving serverless computing using federated learning for smart grids. *IEEE Trans. Ind. Informatics* **2021**, *18*, 7843–7852. [[CrossRef](#)]
25. Huang, Z.; Mi, Z.; Hua, Z. HCloud: A trusted JointCloud serverless platform for IoT systems with blockchain. *China Commun.* **2020**, *17*, 1–10. [[CrossRef](#)]
26. Golec, M.; Gill, S.S.; Parlikad, A.K.; Uhlig, S. HealthFaaS: AI based Smart Healthcare System for Heart Patients using Serverless Computing. *IEEE Internet Things J.* **2023**, *10*, 18469–18476. [[CrossRef](#)]
27. Yussupov, V.; Falazi, G.; Breitenbücher, U.; Leymann, F. On the serverless nature of blockchains and smart contracts. *arXiv* **2020**, arXiv:2011.12729.
28. Benedict, S. Serverless blockchain-enabled architecture for iot societal applications. *IEEE Trans. Comput. Soc. Syst.* **2020**, *7*, 1146–1158. [[CrossRef](#)]

29. Gill, S.S. Quantum and blockchain based Serverless edge computing: A vision, model, new trends and future directions. *Internet Technol. Lett.* **2021**, *7*, e275. [[CrossRef](#)]
30. Ghaemi, S.; Khazaei, H.; Musilek, P. Chainfaas: An open blockchain-based serverless platform. *IEEE Access* **2020**, *8*, 131760–131778. [[CrossRef](#)]
31. Song, J.; Zhang, P.; Alkubati, M.; Bao, Y.; Yu, G. Research advances on blockchain-as-a-service: Architectures, applications and challenges. *Digit. Commun. Netw.* **2022**, *8*, 466–475. [[CrossRef](#)]
32. Chen, H.; Zhang, L.J. Fbaas: Functional blockchain as a service. In *Blockchain–ICBC 2018, Proceedings of the First International Conference, Held as Part of the Services Conference Federation, SCF 2018, Seattle, WA, USA, 25–30 June 2018*; Proceedings 1; Springer: Berlin/Heidelberg, Germany, 2018; pp. 243–250.
33. Mott, D.; Arabadjis, G. Multimicroprocessor With Queue Memories. In *Proceedings of the COMPCON'77, Washington, DC, USA, 6–9 September 1977*; pp. 418–419.
34. Golec, M.; Gill, S.S.; Golec, M.; Xu, M.; Ghosh, S.K.; Kanhere, S.S.; Rana, O.; Uhlig, S. BlockFaaS: Blockchain-enabled serverless computing framework for AI-driven IoT healthcare applications. *J. Grid Comput.* **2023**, *21*, 63. [[CrossRef](#)]
35. Mallick, S.R.; Lenka, R.K.; Goswami, V.; Sharma, S.; Dalai, A.K.; Das, H.; Barik, R.K. Bcgeo: Blockchain-assisted geospatial web service for smart healthcare system. *IEEE Access* **2023**, *11*, 58610–58623. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.