**Original Research Article**

**A differential privacy-preserving framework for secure cloud-based medical information systems**

**ABSTRACT**

Ensuring patient privacy within healthcare systems is paramount, necessitating stringent measures to safeguard personal health information (PHI) from unauthorized access. This study is targeted at presents a robust model designed to facilitate secured access to patients' health data stored in a database, while preserving anonymity. Access control techniques are employed to tailor access privileges based on the role of users, including doctors, nurses, and administrative staff. This study employs the Laplace Mechanism as a strategic approach for the implementation of Differential Privacy when applied to a specific function denoted as 𝑓, which is intended to be executed on a designated database. The effective execution of the Laplace Mechanism is achieved by systematically introducing a calculated amount of noise into the output generated by the function 𝑓, wherein the magnitude of this noise is determined by a specific parameter ε, as will be delineated in the subsequent sections of this discussion.The research entails the development and deployment of a web-based application utilizing contemporary web technologies, with data storage implemented on a cloud-based database hosting service. The front-end of the application is hosted on Netlify, while the backend is deployed on Heroku. To bolster security, patient and staff data are encrypted using the "simple-encryptor" JavaScript library at the front-end, mitigating the risk of unauthorized data access. This underscores its reliability in ensuring both data security and privacy within healthcare settings.

***Keywords****: Access Control, Security models, cloud computing, medical information system, patients’ data, data security, healthcare system*

**1 INTRODUCTION**

Patients’ information privacy in healthcare is crucial for building trust and ensuring quality care. Hence, healthcare professionals are saddle with the responsibility of safeguarding the privacy of patient health information, which includes medical vital records and diagnoses. This privacy will help to foster a trusting environment where patients feel comfortable sharing their health concerns and reporting symptoms honestly, leading to better diagnoses and treatment plans and this will also help to preserve the integrity of the healthcare professionals. However, failure to maintain privacy of patient’s health records can cause the patients personal problems such as social isolation, psychological problems, shame, discrimination, embarrassment and discouragement and fear of seeking medical attention in the future.

Access control is a technique that enables us to emphasize a selected restriction on access to data/privileges of authorized users. Therefore, identification, authentication and authorization are the three major activities that make up the required model to access a secured system. The mechanism of access control allows subject (users) to use their credential to identify themselves as legitimate users and help gain access to resources (Nancy, 2015). There are only two main types of access control: physical and logical.

According to a survey conducted during this research, it was discovered that patients do not feel secure to come and meet medical professionals in the hospital despite the need for them to meet the medical professionals in situations that warrants medical attention, because of stigmatization, embarrassment and discrimination experienced because of the health practitioners not protecting their privacy. Hence, this information security model was developed. In the implementation of this information security model, client server authentication, data masking and permission control techniques were considered.

Client server authentication is an information security technology that authenticates a particular client to certain information in such a way that the identity of the client is provided to the server using either a username or password. This technology can be used to prevent unauthorized access to information/data, as most organizations will adopt it for meeting privacy compliance. The target of privacy in a security system is not just to protect the contents of the message but to protect the identities of communication parties (Liaoliang 2018). In this medical information security system, the medical professionals must not know a patient’s identity at any time. However, the medical professional could be granted access to other health data of the patient.

This paper is centered on the development of a security model designed to preserve the privacy of patients in a cloud based medical information system by granting only authorized persons access to patients’ health data without knowledge of the patient’s identity. This study majorly focused on the health care practitioners on the quest to ensure privacy in patients’ data.

**1.1 LITERATURE REVIEW**

It has been difficult to ignore the fact that security and privacy of patients’ health data is an important aspect of the research, and several computing techniques have been proposed for preserving the privacy of patients. However, so many literatures have been reviewed but a few related ones shall be discussed in this section below.

In the realm of academic research, Insaf (2019) conducted an extensive investigation focused on the intricacies of privacy-preserving access control mechanisms specifically designed for the sharing of health data within cloud environments. The researchers undertook a comprehensive survey of significant contemporary studies aimed at addressing the pressing challenges associated with access control and privacy in cloud-based healthcare systems. Following this thorough examination, they proposed a hybrid solution for access control that effectively combines elements of Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) models. This innovative approach has been shown to facilitate secure, flexible, and adaptable access to sensitive data, thereby fostering a more harmonious relationship between patients and healthcare organizations, albeit with some inherent limitations.

(Misra *et al*., 2019) did a critical study with other researchers on cloud-based health care management with a target on identifying privacy issues and analyzing their effect on cloud-based health care services. Three models were proposed in this work for identifying both the direct and moderating effect of these privacy issues over critical success factors for successfully adopting a cloud-based medical care service.

Azeez et al. (2019) engaged in a meticulous analysis of the privacy and security challenges prevalent in cloud-based electronic health systems, with a specific focus on reviewing existing mechanisms employed to address these critical issues. Their research methodology was designed to respond to a series of pertinent research questions, such as the identification of security and privacy challenges within electronic health systems. After systematically identifying these challenges, the researchers progressed to explore potential solutions aimed at mitigating the identified issues to ensure the security of electronic health data. Furthermore, they provided insightful recommendations regarding future directions that could be pursued to enhance the privacy and security of electronic health systems. However, it is important to note that this research was somewhat constrained by the absence of a developed model specifically tailored to address the security issues associated with electronic health; rather, it primarily focused on providing guidance on methods for maintaining data privacy and security.

(Alanezi, 2019) Conducted a comprehensive analysis that cuts across existing methods and models which have been proposed for solving security and privacy issues of electronic health care systems. They proposed a new intelligent based security and privacy model which maintains and supports the security and privacy of electronic health system after completing the analysis on the existing model. Their model was targeted at several security and privacy issues that affect the electronic health environment. They designed their model in such a way that it accepts requests via a user interface agent, this user interface agent connects the users to health records. It authorizes users who provide correct username and password with necessary protocols being defined by the user interface agent. Furthermore, the database was divided into three regions such as environment, patient region, and current medical information region. The environmental region is made up of the location and time at which the data was obtained, while the patient region is made up of patient’s personal information such as age, status and finally the medical information region is made up of patient’s data such as heart rate, operation history and so on.

Anil (2012) meticulously conceptualized and executed an innovative cloud-based model specifically tailored for a rural healthcare information system, with the overarching objective of significantly enhancing the quality of patient care in such a manner that various disparate systems are meticulously synchronized to promote seamless communication with external systems. This synchronization is vital to ensure the continuity and maintenance of healthcare services, thereby addressing the unique challenges often faced in rural settings where access to comprehensive healthcare information is critical for effective patient management and treatment.

In a collaborative effort, Adebayo et al. (2014) partnered with a consortium of researchers to successfully deploy a sophisticated enterprise cloud-based electronic health record system within a healthcare institution. They systematically designed, implemented, and rigorously tested their work in a manner that facilitates the recording, retrieval, archiving, and updating of patients' medical records, thereby streamlining the management of healthcare data. Within this innovative framework, the cloud served as a centralized database, functioning as a singular data repository for all participating hospitals, while the middleware established a uniform platform for the electronic health record systems utilized across different healthcare facilities. To ensure robust security, they employed authentication server techniques, which granted access solely to authorized users possessing the correct passwords, while unequivocally denying any unauthorized individuals access to sensitive records and crucial resources within the system. Furthermore, in the course of this comprehensive study, they developed an electronic web portal that acted as a vital linkage between the application and the cloud infrastructure. The tools leveraged for the development of this sophisticated system included the Java Development Kit, MySQL, NetBeans IDE, WAMP server, web browsers, among others. In the concluding stages, prior to testing the developed system, they meticulously deployed the system across twenty distinct personal computers in conjunction with one central server, facilitating simulations under varied attack scenarios to evaluate the system’s resilience and effectiveness. Nevertheless, a notable limitation of this research was identified in the realm of privacy, as the medical information stored within the cloud database inadvertently revealed patients' names for identification purposes, raising significant concerns regarding confidentiality.

Sultana et al. (2014) undertook an extensive investigation into the development of cloud-based smart and interconnected data systems specifically designed for health care applications. Their research was methodically structured to enable seamless data integration from a multitude of disparate sources, facilitate cloud-based data access, conduct comprehensive data analysis, and ensure efficient data storage. In the execution of their development, they utilized the Aneka cloud platform, and the effectiveness of their meticulously crafted framework was demonstrated through applications involving patients’ health records and health mapping tools, thereby underscoring the potential of cloud technology in transforming health care data management.

Maulik (2015) conceptualized a robust framework rooted in cloud computing technology that is specifically tailored for healthcare systems, while also incorporating advanced clustering techniques to facilitate region-wise diagnosis. Their model was implemented through a web analyzer interface that effectively displays a variety of results derived from various data mining techniques, illustrating how these techniques can be seamlessly integrated into a cloud-based healthcare application. The development process predominantly utilized the JAVA programming language, while the Weka API was instrumental in integrating data mining techniques into the Eclipse Integrated Development Environment (IDE). The architecture of their model was ingeniously designed such that upon the selection of a specific scenario, data from the corresponding database is efficiently retrieved and subsequently converted into a CSV format. This functionality, along with the CSV data file in conjunction with the Weka API, allows for the presentation of results, contingent upon the application of appropriate data mining techniques, particularly in the context of clustering, thereby showcasing the model's potential efficacy in enhancing healthcare analytics.

**2. MATERIAL AND METHODS**

In this research, a sophisticated cloud computing network that employs the principles of a symmetric key algorithm for the purpose of securing data through encryption has been meticulously considered. The specific symmetric key algorithm that has been adopted for this paper is known as the Data Encryption Standard, commonly referred to as DES, which was selected primarily due to its known vulnerabilities to brute-force attacks, a consequence of its relatively short encryption key length, which can be exploited by attackers. This encryption algorithm boasts a commendable reputation for its robust internal structure and effective design methodologies that reinforce its security capabilities. In order to enhance the overall security framework against brute-force attacks, an innovative prevention mechanism will be introduced, which will incorporate the use of a One-time password system designed to provide users with secure access, in conjunction with a Hash function that will be employed to generate the DES Key by salting the user’s password, thereby adding an additional layer of complexity to the encryption process. The initial phase of the development process is focused on conducting thorough preliminary research aimed at accurately identifying the fundamental requirements, which will subsequently be implemented and rigorously tested to ensure efficacy and reliability. To successfully accomplish the outlined objectives of this project, various technologies such as HTML (Hyper Text Mark-up Language), CSS (Cascading Style Sheet), and JavaScript—serving both front-end and back-end functionalities—were utilized to design the user interface and facilitate interaction between the user and the server. Furthermore, a WAMP (Windows Apache MySQL PHP) stack was employed in the generation of the one-time password and the DES key, as well as in enforcing comprehensive password policies, securely storing user credentials, and conducting robust authentication processes. The overarching research design encapsulates the methodology of structuring the system under examination, meticulously adhering to the specifications of processing requirements that have been established. The primary objectives of this design project are fundamentally aimed at ensuring the adequate security of user information that is stored within the cloud computing environment, which relies on the Data Encryption Standard as its core encryption algorithm.

**VITAL SIGNS COLLECTION**

**DATA MASKING**

**ACCESS CONTROL**

**USER AUTHENTICATION**

**CLOUD BASED DATA STORAGE**

Figure 1: Block diagram of the developed system

**2.1 Password Policy**

Moreover, to fortify the security measures in place, stringent password policies were systematically enforced for every user interacting with the system; these policies are critical as they provide an additional level of security to the Brute Force Prevention System that has been implemented. It is imperative that these established policies be adhered to meticulously for users to gain access to the system; failure to comply with these regulations may lead to situations where access to the system is denied to the user entirely. The stipulations outlined within these policies include the requirement that every password must contain at least one lowercase letter, one uppercase letter, one digit, and one special symbol from the designated set of characters (\\_$%^&+=§!.); additionally, there is a minimum length requirement of eight characters that must be met. Furthermore, if a user attempts to log in more than five times without successfully authenticating, this will result in a denial of access to the account, thereby enhancing security against unauthorized attempts. In addition, any attempt to change a password without accurately inputting the previous password will also lead to access denial, underscoring the importance of safeguarding user accounts. Moreover, users are required to change their passwords periodically; failure to do so may result in the account being blocked, thus ensuring ongoing security measures are upheld.

**2.2 Data Collection**

Data acquisition is the first step in the development of a security model in a cloud based medical information system. The data acquired is the medical information of patients. The data collected for the development of the security model for safeguarding the privacy of patients in a cloud based medical information system was in two formats. The information of medical doctors, Nurses and Administrative staff of the selected hospital and the medical information with vital signs of patients. In this work, the data used was acquired from Delta state Hospital, Ogidigben, Delta State and input into the system for storage in the database.

**2.3 Data Masking**

Data masking represents a crucial process that serves the significant purpose of safeguarding sensitive information within the confines of a database. This sophisticated technique is employed primarily to maintain the confidentiality and privacy of patients' personal data, which is of paramount importance in the healthcare sector. The act of data masking is specifically designed to obscure the identity of patients, thereby preventing users from gaining access to identifiable information concerning individuals stored within the cloud database. For the specific objectives of this project, the dataset utilized was subjected to various data masking techniques before it was securely stored in the cloud environment, ensuring that the sensitive information contained within the database is thoroughly protected and well secured against unauthorized access.

Graphical user interface

Description automatically generated with low confidence

Figure 2: Masked information of patients

**2.4 Access Control**

The implementation of access control measures was meticulously executed to facilitate appropriate access to data residing in the cloud, employing the correct access control methodologies tailored to the specific needs of the system. This comprehensive access control process was also extended to the underlying model, which was designed to limit access to the cloud database exclusively to authorized users, utilizing specific credentials such as user identification numbers and passwords to effectively mitigate the risk of any potential breaches in data privacy. Consequently, this system of access control empowers administrators to effectively regulate and monitor the way users engage with the data, ensuring that only those with legitimate authorization can interact with sensitive information. The importance of access control cannot be overstated, as the establishment of a robust security model inherently necessitates the integration of an access control mechanism to prevent unauthorized users from gaining access to critical data resources.

**2.5 User Authentication**

At the pivotal stage of the login process, it is imperative that the identities of users were thoroughly verified prior to granting them access to the data stored within the system, with the verification process being dependent on the specific roles assigned to each user. This advanced technology ensures that users are securely identified using their email addresses and passwords, thereby granting them the necessary access to medical information regarding patients that reside within the cloud infrastructure. Upon successful authentication of the user's identity, the system immediately proceeds to authorize access to the available data, thereby facilitating a seamless experience for legitimate users. Conversely, should a user fail to meet the required authorization criteria, they are unequivocally denied access to the sensitive data, thereby upholding the integrity and security of the information system.

**2.6 Cloud Based Data Storage**

In relation to the storage solutions for this project, it is noteworthy that various medical information pertaining to patients was strategically stored within the cloud to enhance security measures. The underlying mechanism of the cloud functions as a network of interconnected computers, which collectively represent the expansive capabilities of the internet referred to colloquially as "the cloud." To bolster the information technology infrastructure of medical organizations, the adoption of cloud technology emerges as a critical component in providing a secure and efficient system for managing sensitive data. During the execution phase of the cloud database implementation, a range of cloud services, including prominent platforms such as Azure, Heroku, and AWS, were meticulously evaluated for their respective capabilities. A comprehensive evaluation will be conducted on the performance metrics of the various cloud services utilized for data storage, with the objective of identifying the service that offers the highest level of accuracy and reliability.

**2.7 Definition of ε- differential privacy**

Let us denote ε as a positive real number, which symbolizes a quantity that is greater than zero and useful in the context of mathematical formulations, and allow A to represent a sophisticated randomized algorithm that meticulously processes a given dataset as input, which is intrinsically linked to the actions and behaviors of the presumed entity or individual that possesses the data in question. Furthermore, we shall define Im(A) as the image resulting from the application of A. The algorithm A is rigorously characterized as achieving 𝜖-differential privacy if, for every conceivable pair of datasets denoted as 𝐷1 and 𝐷2, which differ by precisely a single element—essentially reflecting the information pertaining to one individual individual within the dataset—and for all possible subsets 𝑆 derived from Im(A): Differential privacy proffers robust and substantial assurances that significantly enhance the empirical design and analytical scrutiny of differentially private mechanisms, attributing to its remarkable composability, its effectiveness in post-processing stages, and its formidable resilience against potential degradation that may arise in scenarios involving correlated data.

**2.7.1 Differential Privacy preservation model**

In the context of this research project, the Laplace Mechanism is employed as a strategic approach for the implementation of Differential Privacy when applied to a specific function denoted as 𝑓, which is intended to be executed on a designated database. The effective execution of the Laplace Mechanism is achieved by systematically introducing a calculated amount of noise into the output generated by the function 𝑓, wherein the magnitude of this noise is determined by a specific parameter ε, as will be delineated in the subsequent sections of this discussion.

Let us consider the function represented as 𝑓(𝑥1, 𝑥2, 𝑥3,…,𝑥𝑛), which signifies a function that operates on a dataset encapsulated within a database structured as 𝑥 = (𝑥1, 𝑥2, 𝑥3,…,𝑥𝑛); for the purposes of illustration, we can posit that 𝑓 may represent a function that calculates either the average value or the standard deviation of a collection of numerical values. We will also define Δ𝑓 as the maximum difference, formally expressed as Max𝑥𝑥′|𝑓(𝑥) − 𝑓(𝑥′)|, evaluated across all neighboring databases x and x'. As a result, Δ𝑓 is identified as the “sensitivity” of the function 𝑓, which serves to represent the maximal disparity in values that the function 𝑓 may assume when executed on neighboring databases x and x', which differ by precisely one data element, thereby highlighting the potential impact of individual data alterations on the output. For example, if the function 𝑓 is tasked with computing the average of a set of values, then it follows that Δ𝑓 would equal 1/𝑛, and conversely, if the function 𝑓 is engaged in the computation of the standard deviation of a set of values, then Δ𝑓 would be represented as 1/√𝑛. Ultimately, we shall denote 𝑣 as the noise that is meticulously incorporated into the output generated by the function 𝑓, where this noise 𝑣 is sampled in a stochastic manner from a specific probability distribution that is symmetrically centered around zero.

Pr[𝑣] = (1/2𝑏)e^(-|𝑣|/𝑏), where it is essential to note that 𝑏 is defined as Δ𝑓/𝜖.

Consequently, the result obtained from executing the function 𝑓 on a particular database x is succinctly

**2.8 System Algorithm and Data Flow**

The developed security model based in cloud-based medical information system works according to the process flow below in Figure 3

Insert Authorized Password

Password Authentication

Is Password in the database?

NO

Deny Access

YES

Grant Access to Patient health data in the cloud

#### Figure 3: Process flow of the developed model

**2.8.2 Description of the system algorithm**

The entire functionalities of the model can be broken down as follows:

1. There are four categories of users on the system: System Administrator, Admin, Doctors, and Nurses. These are the primary users with distinct responsibilities.
2. The responsibilities of each user are as follows:
   1. System Administrator: A system admin can view the dashboard, view all participants with their credentials, edit user details, delete users, and deactivate and activate users.
   2. Administrators: Administrators can view the dashboard, create patients, and view their details, edit patients, and delete patients from the model. These actions are called CRUD (Create, Read, Update, Delete) operations.
   3. Doctors: Doctors can do everything an admin can do. In addition, doctors can perform CRUD operations on patients’ health records, treatments, and payments.
   4. Nurses: Nurses can do everything doctors can do.
   5. All users can view and edit their profiles, view the dashboard, login into the system, change their passwords and reset their passwords.

Note that users can only view the records of patients they created.

1. A system administrator is initially created on the system as the only default user**.**
2. The system administrator can then create other participants utilizing their staff IDs, emails, and other necessary information. (Users’ emails are unique on the system, meaning there can’t be two users with the same email). The system administrator also assigns passwords to the users at this point.
3. The created users can then perform their respective actions on the model as described in ii. Above.

**3 RESULTS AND DISCUSSION**

The various segments comprising the frontend, backend, and database components of this sophisticated system are designed to function autonomously, thereby ensuring that each part operates independently and does not rely on the others for its functionality. This carefully considered architectural approach was deliberately adopted to mitigate the inherent complexities that often accompany the management of a singular, cohesive project, while simultaneously facilitating a more streamlined process for identifying and rectifying errors, as each section can be addressed in isolation without the need for cross-dependencies. The intricate flow of data within this system is characterized by a seamless communication channel where the frontend interacts with the backend, and in turn, the backend maintains a dialogue with both the frontend and the database. To elucidate this process further, it is important to note that whenever the frontend transmits user data to the backend, the backend dutifully receives this information, processes it accordingly, and subsequently stores it within its own confines. Conversely, when the frontend initiates a request for data from the backend, the backend meticulously processes this request, reaches out to the database in pursuit of the accurate information, and returns this data back to the frontend for user accessibility.

The crucial connection that exists between the frontend and backend components is facilitated by a communication protocol known as “HTTP,” which stands for Hypertext Transfer Protocol, and this protocol serves as the foundation for data interchange between the two. In practical terms, the frontend, which in this context may refer to a web browser or similar interface, sends out an HTTP request directed towards the backend, which in response, generates an HTTP response that is formulated in a manner that is comprehensible to the frontend system. Within the scope of this project, the frontend is equipped to send and receive HTTP requests to and from the backend by utilizing two highly regarded and efficient JavaScript libraries, namely “Axios” and “React Query,” both of which significantly enhance the process. To further optimize the user experience, the data that is returned by the backend is strategically cached within the user’s browser, thereby allowing for quicker access to frequently requested information. The connection that exists between the backend and the database in this research paper is established through the application of a JavaScript Object-Relational Mapping (ORM) library known as Objection.js, which serves to facilitate the interaction between the backend code and the underlying database infrastructure. Objection.js is specifically designed for the Node.js environment and strives to maintain a minimalistic approach, ensuring that it does not obstruct the developer's workflow while simultaneously empowering them to harness the full capabilities of SQL and the underlying database engine, all while ensuring that routine operations remain both straightforward and enjoyable. Built upon the robust SQL query builder called Knex, Objection.js provides a powerful toolset for managing database interactions. Furthermore, Figures 6, 7, and 8 present illustrative evidence showcasing the results obtained during the rigorous testing phase of the developed security model, which was specifically designed to safeguard the privacy of patients' sensitive data within a cloud-based medical information system.

**3.1 Description of System functionalities and Results**

* **User Login View** 
  + 1. Visit the system website by entering the URL [https://health-project-app.netlify.app](https://health-project-app.netlify.app/) in your browser address tab.
    2. Input an email address that is valid and password (Password must be alphanumeric and must contain numbers and special characters). Click proceeds to continue login process.
    3. The backend receives the request and checks the database to see if the credential provided is valid.
    4. If the credential is valid, the backend returns a prompt that contains success response code and response body to the frontend. If there is no user with the credentials, it returns an object that contains failure response code and response body.
    5. The front end gets the signal from the backend and interprets it by reading both the response code and the body of the response. For valid users that have not been deactivated, the frontend logs them in. If not valid or is deactivated, the front end declines their log in.

A login screen with white text and blue and white squares

Description automatically generated

Figure 4: Image showing login-in view of the system

A login screen with white text and pink and blue boxes

Description automatically generated

Figure 5: Image showing log-in view of user with incorrect credentials on the system

* **System Dashboard View** 
  + 1. Utilizers that are successfully logged in are immediately shown on the dashboard.
    2. The dashboard gives an overview of the entire system to the utilizers. The dashboard contains information like the total number of staff (both active and inactive) on the platform, the total number of patients on the system, etc. (as seen in the figure below).

A screenshot of a computer

Description automatically generated

Figure 6: Image showing the dashboard view of a logged-in system administrator on the system

* **User Profile and Change Password View** 
  + 1. All logged-in users on the model have access to their own profile information. They not only have access to the information, but they also can modify some of their information (see image below)
    2. All logged-in users on the system also have access to change their passwords.

A screenshot of a computer

Description automatically generated

Figure 7: Image showing the view used by users to change their passwords.

* **Users Creation and Viewing All Users** 
  + 1. A model administrator has the right and privilege to include a new user on the platform and to see all created users on the platform.
    2. He also can delete, deactivate, or reactivate a user on the platform.

A screenshot of a computer

Description automatically generated

Figure 8: Image showing the view of all users on the platform

* **Patients Creation and Viewing All Patients** 
  1. Admins, Doctors, and Nurses on the model can create patients and can also see a list of all the patients on the model.
  2. The unique ids generated by the model for the patients when they are created are hidden from all users except the user that created the specific patient.
  3. Patients have their own unique ids which they can utilize to get treatment when they visit the hospital facility.

A screenshot of a computer

Description automatically generated

Figure 9: Image showing the view to creating a new patient

* **Patients Health Record Creation and Viewing All Patients’ Health Records** 
  + 1. Doctors and Nurses on the model can create health records for patients and can also see a list of all the patients’ health records on the system.
    2. A doctor or nurse can only see the health records created by him/her alone. For doctors and nurses to see the records not created by them, they need to get the unique id of the patient whose health record they want to view.

A screenshot of a computer

Description automatically generated

Figure 10: Image showing the view to creating a new health record for a patient

* **Patients Treatment Record Creation and Viewing All Patients’ Treatments** 
  + 1. Doctors and Nurses on the model can also create treatment records for patients after each patient has been treated. They can also see a list of all the patients’ treatment records on the model.
    2. A doctor or nurse can only see the treatment records created by him/her alone. For doctors and nurses to see the records not created by them, it is required to have the patient unique id whose treatment record they want to view.

A screenshot of a computer

Description automatically generated

Figure 11: Image showing the view to creating a new treatment record for patients

* **Patients Payment Record Creation and Viewing All Patients’ Payments** 
  1. Admins, Doctors, and Nurses on the model can create payment records for patients after each patient has been treated. They can also see a list of all the patients’ payment records on the model.
  2. An admin, doctor, or nurse can only see the payment records created by him/her alone. For admins, doctors, and nurses to see the records not created by them, they need to get the unique id of the patient whose treatment record they want to view.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedFigure 12: Image showing the view to creating a new payment record for a patient.

Figure 13: Image showing the view that contains the payment records of all patients.

**3.2 Performance Evaluation**

The system was evaluated by different users (using both authorized users and unauthorized users), based on its response to grant access and deny access to user. Regardless of it saves information after reading the password. Regardless of whether it grants access to authorized users, or whether it denies unauthorized user access to data. Regardless, it was able to hide the identity of patients which the database belongs to, and if it was able to perform data management, etc.

Table 1: Performance evaluation of access control.

|  |  |  |
| --- | --- | --- |
| Functional Requirements | YES | NO |
| Read Authorized user. | YES |  |
| Read Unauthorized user | YES |  |
| Grant access to authorized user | YES |  |
| Grant access to unauthorized user |  | NO |
| Read more than one user at a time |  | NO |
| Save user information |  | NO |

**4. CONCLUSION**

This paper has presented a system to safeguard the privacy of patients. With a functional prototype web-based application, users, patients records and treatments were created. The specific web frameworks used for the development are HTML, CSS, SCSS, Bootstrap, React.js, Redux, Typescript, JavaScript, Node.js, and PostgreSQL. The deployment of the front-end section of this project was done on Netlify while the deployment of the backend section of this project was done on Heroku. The database used for this model is PostgreSQL and it’s deployed on Heroku. The front-end section of this project was built majorly with React.js. React is a declarative, efficient, and flexible JavaScript library, for building user interfaces, while the backend section of this project was built using Node.js. The front end and the backend java script have been used to design the interface which interacts with the utilizer and the server while the windows Apache MySQL, PHP were used to generate a one-time password system which authorized users are granted permission to the cloud database. While this system was being developed, various data masking techniques were studied, various client-server authentication techniques and permission control techniques, and the local differential privacy technique was used to safeguard patients’ identity in the cloud-based medical system.

Disclaimer (Artificial intelligence)

Option 1:

I hereby declare that this research work was done by me and NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc) and text to image generators have been used during the writing or editing of the manuscript.

Author(s) hereby declare that NO generative AI technologies such as Large Language Models (ChatGPT, COPILOT, etc.) and text-to-image generators have been used during the writing or editing of this manuscript.

Option 2:

Author(s) hereby declare that generative AI technologies such as Large Language Models, etc. have been used during the writing or editing of manuscripts. This explanation will include the name, version, model, and source of the generative AI technology and as well as all input prompts provided to the generative AI technology

Details of the AI usage are given below:

1.

2.

3.

REFERENCES

[1] M. A Alanezi, Z. F. (2019). Intelligent-based E-healthcare Systems: Towards Security and Privacy. International Journal of Computer Science and Network Security, Vol 19, Issue No 3, PP 16-23.

[2] Anil PD, R. M. (2012). Privacy Preservation Measure using t-closeness with combined l-diversity and anonymity. International Journal of Advanced Research in Computer Science and Electronics Engineering, Vol 1, issue 8, PP 28-33.

[3] Bai Qing-hai, Z. Y. (2011). Study on the Access Control Model in Information Security.

[4] G. Chauhan. (2013). A Review of Privacy Preservation Techniques. International Journal of Engineering Research & Technology, Vol 2, Issue 12, PP 495.

[5] Adebayo A. Abayomi-Alli, A. J.-A. (2014). An Enterprise Cloud-Based Electronic Health Records System. Journal of Computer Science and Information Technology, Vol. 2, No. 2, pp. 21-36.

[6] Adeyanju I. A, O. E. (2015). Performance evaluation of Different Support Vector Machine Kernels for Face Emotion Recognition. SAI Intelligent Systems Conference (pp. PP 804-806). London: SAI Intelligent Systems.

[7] Cross Strait Quad-Regional Radio Science and Wireless Technology Conference, IEEE, PP 830-834.

[8] C. A Natividad Peña, A. E. (2019). Security model to protect patient data in mHealth systems through a Blockchain network . 17th LACCEI International Multi-Conference for Engineering, Education, and Technology: Industry, Innovation, And Infrastructure for Sustainable Cities and Communities, (pp. PP 1-6). Jamaica.

[9] Chen K, L. L. (2011). Geometric data perturbation for privacy preserving outsourced data mining. ACM Journal of Knowledge and Information Systems, volume 29, issue no 3, pp 657-695.

[10] F. Harmon, R. A. (1989). Reading between the Lines. Peterborough, New Hampshire, USA: Helmers Publishing, Inc,.

[11] F.A Onik, S. S.-A.-M. (2012). A Secured Cloud based Health Care Data Management System. International Journal of Computer Applications, Volume 49, Issue No.12, PP 24-30.

[12] Fang Liu, T. L. (2018). A Clustering -Anonymity Privacy-Preserving Method for Wearable IoT Devices. Hindawi Security and Communication Networks, PP 1-8.

[13] Gajanayake R, I. R. (2012). Privacy Oriented Access Control for Electronic health record.

[14] James, W. G. (1993). On introducing noise into the bus-contention channel. Proceedings of the IEEE Symposium on Security and Privacy, pp 90–98.

[15] Insaf Boumezbeur, K. Z. (2019). Privacy-preserving access control for sharing health data in cloud environment. The 8th International Seminary on Computer Science Research at Feminine, (pp. PP 25-31).

[16] Jain A. K, R. A. (2004). An Introduction to Biometric Recognition. IEEE Transactions on Circuits and Systems for Video Technologies, Vol 14, Issue no 1, PP 1-66.

[17] Jemal Hanen, Z. K. (2016). An enhanced healthcare system in mobile cloud computing environment. Vietnam Journal of Computer Science, PP 267–277.

[18] Juels A, R. R. (2003). The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. CCS'03. Washington, DC, USA: 9.

[19] Abouelmehdi, A. B.-H. (2018). Big healthcare data: preserving security and privacy. Journal of big data, PP 1-18.

[20] Kanchana, R. V. (2015). Security Model for Healthcare Application In Cloud Computing. International Journal of Computer Science and Engineering Communications, Vol.3, Issue 2, PP 627-635.

[21] Kato Mivule. (2012). Utilizing Noise Addition for Data Privacy an overview. Computer science department, Bowen University.

[22] Liaoliang Jiang, T. L. (2018). Anonymous Communication via Anonymous Identity-Based Encryption and Its Application in IoT. Hindawi Wireless Communications and Mobile Computing, PP 1-8.

[23] Maulik Parekh, S. B. (2015). Designing a Cloud based Framework for HealthCare System and applying Clustering techniques for Region Wise Diagnosis. 2nd International Symposium on Big Data and Cloud Computing: Procedia Computer Science (pp. PP 537 – 542). Elsevier B.V.

[24] Mijanur Rahman, S. N. (2016). Biometric student registration and verification system.1-3.

[25] Mityaev, S. B. (2013). Image binarization for end-to-end text understanding in natural images.

[26] Mu-Hsing KUO, A. K. (2013). IT Intelligence Cloud Computing for Health Information Management. Health management.org, Volume 13, Issue 2, PP 1.

[27] N.A Azeez, C. V. (2019). Security and privacy issues in e-health cloud-based system: A comprehensive content analysis. Egyptian Informatics Journal, PP 97-108.

[28] Nguyen. (2019, June 30). Understanding differential privacy. Retrieved from towrads data science:

[29] O.P Chaurasia. (2012). An Approach to Fingerprint Image Preprocessing. International Journal of image Graphics and Signal Processing, Vol 6, issue no 5, PP 29-35.

[30] R.P Padhy, M. R. (2012). Design and Implementation of a Cloud based Rural Healthcare Information System Model. Universal journal of applied computer science and technology, Vol 2, issue 1, PP 149-157.

[31] Ravikumar G K, B. J. (2011). Design of Data Masking Architecture and Analysis of Data Masking Techniques for Testing. International Journal of Engineering Science and Technology, Vol. 3 No. 6, PP 5153.

[32] Chenthara, K. A. (2019). Security and Privacy-Preserving Challenges of e-Health Solutions in Cloud Computing. IEEE Access, Vol 7, PP 74361-74382.

[33] Naisha Sultana, G. R. (2014). CLOUD-BASED DEVELOPMENT OF SMART AND CONNECTED DATA IN HEALTHCARE APPLICATION. International Journal of Distributed and Parallel Systems, Vol 5, issue no 6, PP 1- 11.

[34] Shoewu, O. I. (2012). Development of attendance management system using biometrics. The pacific journal of science and technology, Vol 13, issue 1, PP 300-307.

[35] Simson L. G, J. A. (2005). RFID privacy: An overview of problems and proposed solutions. IEEE Security and Privacy, Vol 3, issue 3, PP 34–43.

[36] Stallings W, B. L. (2008). Computer Security. Principles and Practice.

[37] ubhas C. Misra, A. K. (2019). Cloud-based healthcare management: Identifying the privacy concerns and their moderating effect on the cloud-based health-care services. Indian Institute of Technology (pp. PP 1-18). Kanpur India: John Wiley & Sons, Ltd.

[38] Vinoth Kumar, S. (2016). A Brief Survey on Privacy Preserving Techniques in Data Mining. IOSR Journal of Computer Engineering, Volume 18, Issue 4, PP 47-51.

[39] Wenrong Z, Y. Y. (2014). Content-Based Access Control: Use Data Content to Assist Access Control for Large-Scale Content-Centric Databases. IEEE International Conference on Big Data, PP 701-710