TITLE OF THE MINI PROJECT - II

A MINI PROJECT-II REPORT

Submitted by

SRIDHAR M	**(1801227)**
VIGNESHWARAN R	**(1801250)**
VISNUPIRIYAN K	**(1801258)**
VENGATESH R	**(1801267)**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

SRI RAMAKRISHNA ENGINEERING COLLEGE

[Educational Service: SNR Sons Charitable Trust]
[Autonomous Institution, Accredited by NAAC with ‘A’ Grade]
[Approved by AICTE and Permanently Affiliated to Anna University, Chennai]
[ISO 9001:2008 Certified and All Eligible Programmes Accredited by NBA]
Vattamalaipalayam, N.G.G.O. Colony Post,

COIMBATORE – 641 022

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2021

BONAFIDE CERTIFICATE

16CS266 – MINI PROJECT II

Certified that this Mini Project - II Report **“HEALTH CARE MANAGEMENT USING BLOCKCHAIN”** is the bonafide work of **“Sridhar M, Vigneshwarar R, Visnupiriyan K, Vengetash R”** who carried out the project under my supervision.

SIGNATURE	**SIGNATURE**
Dr.Grace Selvarani	Mr.S.Hariharagopalan
HEAD OF THE DEPARTMENT	**SUPERVISOR**
Professor,	Associate Professor
Computer Science and Engineering,	Computer Science and Engineering,
Sri Ramakrishna Engineering College,	Sri Ramakrishna Engineering College,
Coimbatore-641022.	Coimbatore-641022.

Submitted for the Mini Project Viva-Voce Presentation held on ____________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We affirm that the project work titled “**HEALTH CARE MANAGEMENT USING BLOCKCHAIN**” being submitted in partial fulfilment for the award of Bachelor of Engineering is the original work carried out by us. It was not formed the part of any other project work submitted for award of any degree or diploma, either in this or any other university.

(Signature of the Candidates)

SRIDHAR M	(1801227)
VIGNESHWARAN R	(1801250)
VISNUPIRIYAN K	(1801258)
VENGATESH R	(1801267)

I certify that the declaration made above by the candidate is true.

(Signature of the Guide)

Mr.S.Hariharagopalan(MCA,Ph.D).

Designation,

Department of Computer Science and Engineering

ACKNOWLEDGEMENT

We express our gratitude to **Sri. D. LAKSHMINARAYANASWAMY,** Managing Trustee, **Sri. R. SUNDAR,** Joint Managing Trustee, SNR Sons Charitable Trust, Coimbatore for providing excellent facilities to carry out our project.

We express our deepest gratitude to our Principal, **Dr. N. R. ALAMELU, Ph.D.,** for her valuable guidance and blessings.

We are indebted to our Head of the Department, **Dr. A. Grace Selvarani, Ph.D.,** Department of Computer Science and Engineering who modelled us both technically and morally for achieving great success in life.

We express our thanks to our Project Coordinator, **Mrs.R.Madhumathi.,** Assistant Professor (Sl. Grade) Department of Computer Science and Engineering for his great inspiration.

Words are inadequate to offer thanks to our respected guide. We wish to express our sincere thanks to **Mr.S.Hariharagopalan,Ph.D ,** Assistant Professor /CSE**,** Department of Computer Science and Engineering, who gives constant encouragement and support throughout this project work and who makes this project a successful one.

We also thank all the staff members and technicians of our Department for their help in making this project a successful one.

TABLE OF CONTENTS

ABSTRACT

Blockchain is emerging as a promising candidate for the uberization of Internet services. It is a decentralized, secure, and auditable solution for exchanging, and authenticating information via transactions, without the need of a trusted third party. Therefore, blockchain technology has recently been integrated with health care domain. The decentralized database in BT emphasizes data security and privacy. Also, the consensus mechanism in it makes sure that data is secured and legitimate. Still, it raises new security issues such as majority attack and double-spending. To handle the aforementioned issues, data are encrypted using Advanced Encryption Standards (AES) algorithm. A blockchain powered health information exchange could unlock the true value of interoperability. Blockchain-based systems have the potential to reduce or eliminate the friction and costs of current intermediaries. The promise of blockchain has widespread implications for stakeholders in the health care ecosystem. Capitalizing on this technology has the potential to connect fragmented systems to generate insights and to better assess the value of care. In the long term, a nationwide blockchain network for electronic medical records may improve efficiencies and support better health outcomes for patients. At its core, blockchain is a distributed system recording and storing transaction records. More specifically, blockchain is a shared, immutable record of peer-to-peer transactions built from linked transaction blocks and stored in a digital ledger. Blockchain relies on established cryptographic techniques to allow each participant in a network to interact (e.g., store, exchange, and view information), without pre-existing trust between the parties.

Objectives

The technology without central authority of management is called Blockchain. In other words, there is no intermediate party is required for transferring and storing the assets from one to other using Peer-to-Peer communication. Blockchain solutions may prove to be an exception in that some applications can improve security and optimize the ability to exchange and track value. This pandemic has not only overwhelmed the healthcare sector but has also highlighted the need for information technology-based solutions for effective and trustworthy management of such pandemics. Availability of data, easy accessibility to data and efficient data sharing among all the stakeholders while preserving data privacy and security are vital aspects for detection, tracking, diagnosis and regular interventions of infectious diseases. There is a growing consensus among researchers that blockchain has an ability to overcome these issues and has the potential to make a revolution in the healthcare sector with its unique qualities of decentralization, anonymity, distributed ledger, robustness, and cryptography

LIST OF FIGURES

LIST OF TABLES AND ABBREVIATIONS

A

Advanced Encryption Standards (AES)

D

Data Low Diagram (DFD)

E

Entity Relationship Diagram (ER diagram)
Electronic Health Record (EHR)

L

Long Short-term Memory (LSTM)

INTRODUCTION

A blockchain is a distributed software network that functions both as a digital ledger and a mechanism enabling the secure transfer of assets without an intermediary. Just as the internet is a technology that facilitates the digital flow of information, blockchain is a technology that facilitates the digital exchange of units of value. Anything from currencies to land titles to votes can be tokenized, stored, and exchanged on a blockchain network

Applications in healthcare Blockchains in healthcare can be envisaged in five primary areas:

Managing electronic medical record (EMR) data Protection of healthcare data Personal health record data management Point-of-care genomics management Electronics health records data management.

Blockchain technology has the potential to transform health care, placing the patient at the center of the health care ecosystem and increasing the security, privacy, and interoperability of health data. This technology could provide a new model for health information exchanges (HIE) by making electronic medical records more efficient, disintermediated, and secure. While it is not a panacea, this new, rapidly evolving field provides fertile ground for experimentation, investment, and proof-of-concept testing

LITERATURE REVIEW

Title: AI Enabled Blockchain Smart Contracts: Cyber Resilient Energy Infrastructure and IoT

Author: Michael Mylrea

The commoditization of trust has been the topic of science fiction, futuristic novels and theoretical study for the last century. Advances in blockchain and artificial intelligence technology continue to make science-fiction a reality, automating and replacing the need for 3rd party intermediaries and other trust mechanisms, potentially disrupting many critical industries. Blockchain enabled smart contracts show potential to exchange value without third party trust mechanisms. The combination of artificial intelligence, cryptography, distributed trust algorithms or smart contracts have paved the way to a more efficient and secure way to exchange value, goods and services. This paper explores how blockchain technology could potentiality automate and modernize energy and the internet of things to help evolve energy infrastructure to an increasingly automated, distributed, clean and resilient system. This is timely as the U.S. power grid and the array of things that it connects to is a complex system of systems in which the nation's economy, national security and livelihood depends on.

Title: Dynamic Access Control Policy based on Blockchain and Machine Learning for the Internet of Things

Author: Aissam OUTCHAKOUCHT, Hamza ES-SAMAALI, Jean Philippe LEROY

The Internet of Things (IoT) is now destroying the barriers between the real and digital worlds. However, one of the huge problems that can slow down the development of this global wave, or even stop it, concerns security and privacy requirements. The criticality of these latter comes especially from the fact that the smart objects may contain very intimate information or even may be responsible for protecting people's lives. In this paper, the focus is on access control in the IoT context by proposing a dynamic and fully distributed security policy. Our proposal will be based, on one hand, on the concept of the blockchain to ensure the distributed aspect strongly recommended in the IoT; and on the other hand on machine learning algorithms, particularly on reinforcement learning category, in order to provide a dynamic, optimized and self-adjusted security policy.

Title: A Survey on Blockchain Based Smart Applications

Author: Saranya, Mythili

Most of commercial and industry applications are centralized and time consuming and also need third party to audit the information. Need of middle man in every application are prone to error sometimes. The Blockchain technology merely conquers the issues of centralized –middle party communication. The decentralized computation and information sharing platform that enables multiple authoritative domains, to cooperate, coordinate and collaborate each other with rational decision-making process. A Blockchain is an open distributed ledger, which can record transactions between two parties efficiently and in a verifiable and permanent way. A decentralized database with strong consistency support provides to update the local copy of the global information. Every transaction ensures the integrity at the last minute of the Blockchain to make the entire chain to be tamper proof. Here our study gives the overview of blockchain terminologies, consensus algorithms and various fields of applications of Blockchain

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The need of secure big data storage service is more desirable than ever to date. The basic requirement of the service is to guarantee the confidentiality of the data. However, the anonymity of the service clients, one of the most essential aspects of privacy, should be considered simultaneously. Moreover, the service also should provide practical and fine-grained encrypted data sharing such that a data owner is allowed to share a cipher text of data among others under some specified conditions.

3.2 PROPOSED SYSTEM

Proposed system focuses on the problem of secure health records on encrypted data, another type of similarity searches important for multi-criteria decision making. Proposed system outsources its electronic health records to the server and the data is encrypted to ensure data confidentiality. Data owners may share their outsourced data with a number of users, who might want to only retrieve the data files they are interested in. One of the most popular ways to do so is through keyword-based retrieval. Keyword- based retrieval is a typical data service and widely applied in plain text scenarios, in which users retrieve relevant files in a file set based on keywords. However, it turns out to be a difficult task in cipher text scenario due to limited operations on encrypted data. We are using navie bayes algorithms for classifying the health records if it needs to be highly secure, medium secure, low secure content.

SYSTEM SPECIFICATIONS

4.1 HARDWARE SPECIFICATIONS

PROCESSOR	:	Intel i3 Processor 2.5GHZ
HARD DISK CAPACITY	:	400 GB
MONITOR	:	15 “SAMTRON MONITOR
INTERNAL MEMORY CA	:	4 GB
KEYBOARD	:	LOGITECH OF 104 KEYS
CPU CLOCK	:	1.08 GHz
MOUSE	:	LOGITECH MOUSE

4.2 SOFTWARE SPECIFICATIONS

OPERATING SYSTEM	:	WINDOWS 10
FRONT END	:	PHP
BACK END	:	MYSQL

METHODOLOGY

5.1 MODULES

Data owner Module

Data owner register:

Data owner register their details like name, email, phone, username and password.

Data owner Login:

Data owner logs into the system with registered username and password.

Data owner medical report history:

Data owner enter their health history attribute details like age, gender, whether they are overweight, whether they are mentally stressed, , whether they have pin in chest, blood pressure detail, blood sugar details. All the data owner files gets encrypted using AES algorithm, files will be classified using navie bayes classification algorithm and gets stored in server. We are using navie bayes algorithms for classifying the health records if it needs to be highly secure, medium secure, low secure content. The data owner files have attributes like report name, report keyword etc.

View Data User Request:

In this module Data owner can view the file access request send by the data user. File name, report name will be displayed to Data owner. Data owner can grant the access to Data user and access key will notified to the Data user.

Data user Module

Data user register:

Data user registers their details like name, email, phone, username and password.

Data user Login:

Data user logs into the system with username and password.

Data User Send Request:

In this module Data user can search the report file by specifying report keyword. The file is shown based on report keyword given by Data owner. Data user can send the file access request to Data owner. After Data owner approval Data user can view the file.

View file

In this module Data user can view report file by specifying the access key given by the data owner. If correct access key is given encrypted file will be decrypted and shown to user. If wrong access key is given, error message is thrown to user.

Software Description

PHP

PHP stands for Hypertext Preprocessor. PHP scripts run inside Apache server or Microsoft IIS. PHP and Apache server are free. PHP code is very easy. PHP is the most used server-side scripting language. PHP files contain PHP scripts and HTML. PHP files have the extension “php”, “php3”, “php4”, or “html”.

Using PHP

- Generate dynamic web pages. PHP can display different content to different user or display different content at different times of the day.
- Process the contents of HTML forms. We can use a PHP to retrieve and respond to the data entered into an HTML form.
- Can create database-driven web pages. A PHP can insert new data or retrieve existing data from a database such a MySQL.

Working of PHP

PHP is a standard HTML file that is extended with additional features. Like a standard HTML file, PHP contains HTML tag that can be interpreted and displayed by a web browser. Anything we could normally place in an HTML file Java applets, Blinking text, server side scripts. we can place in PHP. However, PHP has three important features that make it unique.

- PHP contains server-side scripts.
- PHP provides several built-in objects.

HTML

HTML is an application of the Standard Generalized Markup Language (SGML), which was approved as an international standard in the year 1986. SGML provides a way to encode hyper documents so they can be interchanged.

SGML is also a Meta language for formally describing document markup system. Infact HTML uses SGML to define a language that describes a WWW hyper document's structure and inter connectivity.

Following the rigors of SGML, TBL bore HTML to the world in 1990. Since then, many of us have it to be easy to use but sometimes quite limiting. These limiting factors are being addressed but the World Wide Web Consortium (aka W3c) at MIT. But HTML had to start somewhere, and its success argues that it didn't start out too badly.

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. HTML is a computer language devised to allow website creation. These websites can then be viewed by anyone else connected to the Internet. It is relatively easy to learn, with the basics being accessible to most people in one sitting; and quite powerful in what it allows you to create. It is constantly undergoing revision and evolution to meet the demands and requirements of the growing Internet audience under the direction of the W3C, the organization charged with designing and maintaining the language.

CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device. The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

JAVASCRIPT:

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities. JavaScript was first known as Live Script, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name Live Script. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers

JavaScript is a case-sensitive language. This means that the language keywords, variables, function names, and any other identifiers must always be typed with a consistent capitalization of letters. So the identifiers Time and TIME will convey different meanings in JavaScript.

MYSQL:

MySQL Server is a powerful database management system and the user can create application that requires little or no programming. It supports GUI features and an entire programming language, PhpMyAdmin which can be used to develop richer and more developed application. There are quite a few reasons, the first being that MySQL is a feature rich program that can handle any database related task you have. You can create places to store your data build tools that make it easy to read and modify your database contents, and ask questions of your data. MySQL is a relational database, a database that stores information about related objects. In MySQL that database means a collection of tables that hold data. It collectively stores all the other related objects such as queries, forms and reports that are used to implement function effectively.

The MySQL database can act as a back-end database for PHP as a front end, MySQL supports the user with its powerful database management functions. A beginner can create his/her own database very simply by some mouse clicks. Another good reason to use MySQL as back-end tool is that it is a component of the overwhelmingly popular Open-source software

Database:

A database is simply a collection of used data just like phone book. MySQL database include such objects as tables, queries, forms, and more.

Tables:

In MySQL tables are collection of similar data. With all tables can be organized differently, and contain mostly different information- but they should all be in the same database file. For instance, we may have a database file called video store. Containing tables named members, tapes, reservations and so on. These tables are stored in the same database file because they are often used together to create reports to help to fill out on screen forms.

Relational database:

MySQL is a relational database. Relational databases tools like access can help us manage information in three important ways.

- Reduce redundancy
- Facilitate the sharing of information
- Keep data accurate.

Fields

Fields are places in a table where we store individual chunks of information.

Primary key and other indexed fields:

MySQL use key fields and indexing to help speed many database operations. We can tell MySQL, which should be key fields, or MySQL can assign them automatically.

Controls and objects:

Queries are access objects us display, print and use our data. They can be things like field labels that we drag around when designing reports. Or they can be pictures, or titles for reports, or boxes containing the results of calculations.

Queries and dataset:

Queries are request to information. When access responds with its list of data, that response constitutes a dataset. A dynamic set of data meeting our query criteria. Because of the way access is designed, dataset are updated even after we have made our query.

Forms:

Forms are on screen arrangement that make it easy to enter and read data. we can also print the forms if we want to. We can design form our self, or let the access auto form feature.

Reports:

Reports are paper copies of dataset. We can also print reports to disk, if we like. Access helps us to create the reports. There are even wizards for complex printouts.

Properties:

Properties are the specification we assigned to parts of our database design. We can define properties for fields, forms, controls and most other access objects.

Feasibility Study

Feasibility is a practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software or not.

Such information as resource availability, cost estimate for software development, benefits of the software to organization, and cost to be incurred on its maintenance are considered. The objective of the feasibility study is to establish the reasons for developing a software that is acceptable to users, adaptable to change, and comfortable to standards.

Types of Feasibility

1. Technical feasibility
2. Operational feasibility
3. Economic feasibility.

Technical Feasibility

Technical Feasibility assesses the current resources and technology, which are required to accomplish user requirement in the software within the allocated time and For this, the software development team ascertains whether the current resources and technology can be upgraded or added in the software to accomplish specified user requirements.

Technical feasibility performs the following tasks:-

1. It analyses the technical and capabilities of the software development team members.
2. It determines whether the relevant technology is stable and established.
3. It ascertains that the technology chosen for software development has large number of user so that they can be consulted when problems arise, or when improvements are required.

Operational Feasibility

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. This feasibility is dependent on human resource and involves visualizing whether or not the software will operate after it is developed, and be operated once it is installed.

It also performs the following tasks: -

1. It determines whether or on Economic Feasibility

Economic Feasibility

Economic feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on. For this, it is essential to consider expenses made on purchases and activities required to carry out software development. In addition it is necessary to consider the benefits that can be achieved by developing the software.

1) Cost incurred on software development to produce long-term gains for an organization.
2) Cost required to conduct full software investigation.
3) Cost of hardware, software, development team and training.

Feasibility Study Process :-

Feasibility study comprises the following steps: -

1. Information assessment: -Identifies information about whether the system helps in achieving the objectives of the organization. It also verifies that the system can be implemented using new technology and within the budget, and whether the system can be integrated with the existing system.

2. Information collection: - Specifies the sources from where information about software can be obtained. Generally, these sources include users, and the software development team.

3. Report writing: - Uses a feasibility report, which is the conclusion of the feasibility by the software development team. In includes the recommendation whether the software development should continue or not.

BLOCKCHAIN:

A blockchain is essentially a digital ledger of transactions that is duplicated and distributed across the entire network of computer systems on the blockchain. Each block in the chain contains a number of transactions, and every time a new transaction occurs on the blockchain, a record of that transaction is added to every participant's ledger. The decentralized database managed by multiple participants is known as Distributed Ledger Technology (DLT).

1.Immutability:

There are some exciting blockchain features but among them "Immutability" is undoubtedly one of the key features of blockchain technology. But why is this technology uncorrupted? Let's start with a connecting blockchain with immutability.

2. Decentralized

The network is decentralized meaning it doesn't have any governing authority or a single person looking after the framework. Rather a group of nodes maintains the network making it decentralized. This is one of the key features of blockchain technology that works perfectly. Let me make it simpler. Blockchain puts us users in a straightforward position. As the system doesn't require any governing authority, we can directly access it from the web and store our assets there.

3. Enhanced Security

As it gets rid of the need for a central authority, no one can just simply change any characteristics of the network for their benefit. Using encryption ensures another layer of security for the system. It’s extremely secure because it offers a special disguise – Cryptography.

4. Distributed Ledgers

Usually, a public ledger will provide every information about a transaction and the participant. It’s all out in the open, nowhere to hide. Although the case for private or federated blockchain is a bit different. But still, in those cases, many people can see what really goes on in the ledger. That’s because the ledger on the network is maintained by all other users on the system. This distributed computational power across the computers to ensure a better outcome

5. Consensus

Every blockchain thrives because of the consensus algorithms. The architecture is cleverly designed, and consensus algorithms are at the core of this architecture. Every blockchain has a consensus to help the network make decisions. In simple terms, the consensus is a decision-making process for the group of nodes active on the network. Here, the nodes can come to an agreement quickly and relatively faster. When millions of nodes are validating a transaction, a consensus is absolutely necessary for a system to run smoothly. You could think of it as kind of a voting system, where the majority wins, and the minority has to support it

6. Faster Settlement

Traditional banking systems are quite slow. Sometimes it can take days to process a transaction after finalizing all settlements. It also can be corrupted quite easily. Blockchain offers a faster settlement compared to traditional banking systems. This way a user can transfer money relatively faster, which saves a lot of time in the long run.These blockchain features make life easier for foreign workers and help to understand Why Blockchain is Important. Many people travel to another country in search of a better life and job and leave families behind. However, sending money to their families overseas takes a lot of time and could become fatal in times of need.

5.2 DESIGN / BLOCK DIAGRAM

FUNDAMENTAL DESIGN CONCEPTS

System design is a “how to” approach to creation of a new system. System design goes through 2 phases. They are

- Logical design

- Physical design

Logical design reviews the present physical system, prepares input and output specifications, makes edit security and control specifications

Physical design maps out the details of the physical system, plans, system implementation, device a test and implementation plan.

DESIGN PROCESS

INPUT DESIGN

Input design is the process of converting the user-oriented. Input to a computer based format. The goal of the input design is to make the data entry easier , logical and free error. Errors in the input data are controlled by the input design. The quality of the input determines the quality of the system output.

All the data entry screen are interactive in nature, so that the user can directly enter into data according to the prompted messages. The user are also can directly enter into data according to the prompted messages. The users are also provided with option of selecting an appropriate input from a list of values. This will reduce the number of error, which are otherwise likely to arise if they were to be entered by the user itself.

Input design is one of the most important phase of the system design. Input design is the process where the input received in the system are planned and designed, so as to get necessary information from the user, eliminating the information that is not required. The aim of the input design is to ensure the maximum possible levels of accuracy and also ensures that the input is accessible that understood by the user. The input design is the part of overall system design, which requires very careful attention. If the data going into the system is incorrect then the processing and output will magnify the errors.

The objectives considered during input design are:

- Nature of input processing.
- Flexibility and thoroughness of validation rules.
- Handling of properties within the input documents.
- Screen design to ensure accuracy and efficiency of the input relationship with files.
- Careful design of the input also involves attention to error handling, controls, batching and validation procedures.

Input design features can ensure the reliability of the system and produce result from accurate data or they can result in the production of erroneous information.

Data Flow Diagram (DFD)

The first step is to draw a data flow diagram (DFD). The DFD was first developed by Larry Constantine as a way of expressing system requirements in graphical form.

A DFD also known as a “bubble chart” has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So, it is the starting point of the design phase that functionally decomposes the requirements specifications down to the lowest level of detail. A DFD consists of series of bubbles join by the data flows in the system.

The purpose of data flow diagrams is to provide a semantic bridge between users and systems developers. The diagrams are:

- Graphical, eliminating thousands of words;
- Logical representations, modeling WHAT a system does, rather than physical models showing HOW it does it;
- Hierarchical, showing systems at any level of detail; and
- jargon less, allowing user understanding and reviewing.

The goal of data flow diagramming is to have a commonly understood model of a system. The diagrams are the basis of structured systems analysis. Data flow diagrams are supported by other techniques of structured systems analysis such as data structure diagrams, data dictionaries, and procedure-representing techniques such as decision tables, decision trees, and structured English.

External Entity

An external entity is a source or destination of a data flow, which is outside the area of study. Only those entities, which originate or receive data, are represented on a business process diagram. The symbol used is an oval containing a meaningful and unique identifier.

Process

A process shows a transformation or manipulation of data flows within the system. The symbol used is a rectangular box, which contains 3 descriptive elements: Firstly an identification number appears in the upper left hand corner. This is allocated arbitrarily at the top level and serves as a unique reference. Secondly, a location appears to the right of the identifier and describes where in the system the process takes place.

Data Flow

A data flow shows the flow of information from its source to its destination. A data flow is represented by a line, with arrowheads showing the direction of flow. Information always flows to or from a process and may be written, verbal or electronic. Each data flow may be referenced by the processes or data stores at its head and tail, or by a description of its contents.

Data Store

A data store is a holding place for information within the system: It is represented by an open ended narrow rectangle. Data stores may be long-term files such as sales ledgers, or may be short-term accumulations: for example batches of documents that are waiting to be processed. Each data store should be given a reference followed by an arbitrary number.

Resource Flow

A resource flow shows the flow of any physical material from its source to its destination. For this reason they are sometimes referred to as physical flows. The physical material in question should be given a meaningful name. Resource flows are usually restricted to early, high-level diagrams and are used when a description of the physical flow of materials is considered to be important to help the analysis.

OUTPUT DESIGN

The output form of the system is either by screen or by hard copies. Output design aims at communicating the results of the processing of the users. The reports are generated to suit the needs of the users .The reports have to be generated with appropriate levels. In our project outputs are generated by asp as html pages. As its web application output is designed in a very user-friendly this will be through screen most of the time.

CODE DESIGN

The main purpose of code design is to simplify the coding and to achieve better performance and quality with free of errors. The coding is prepared in such a way that the internal procedures are more meaningful validation manager is displayed for each column. The coding of the variables is done in such a way that one other than person who developed the packages can understand its purpose.

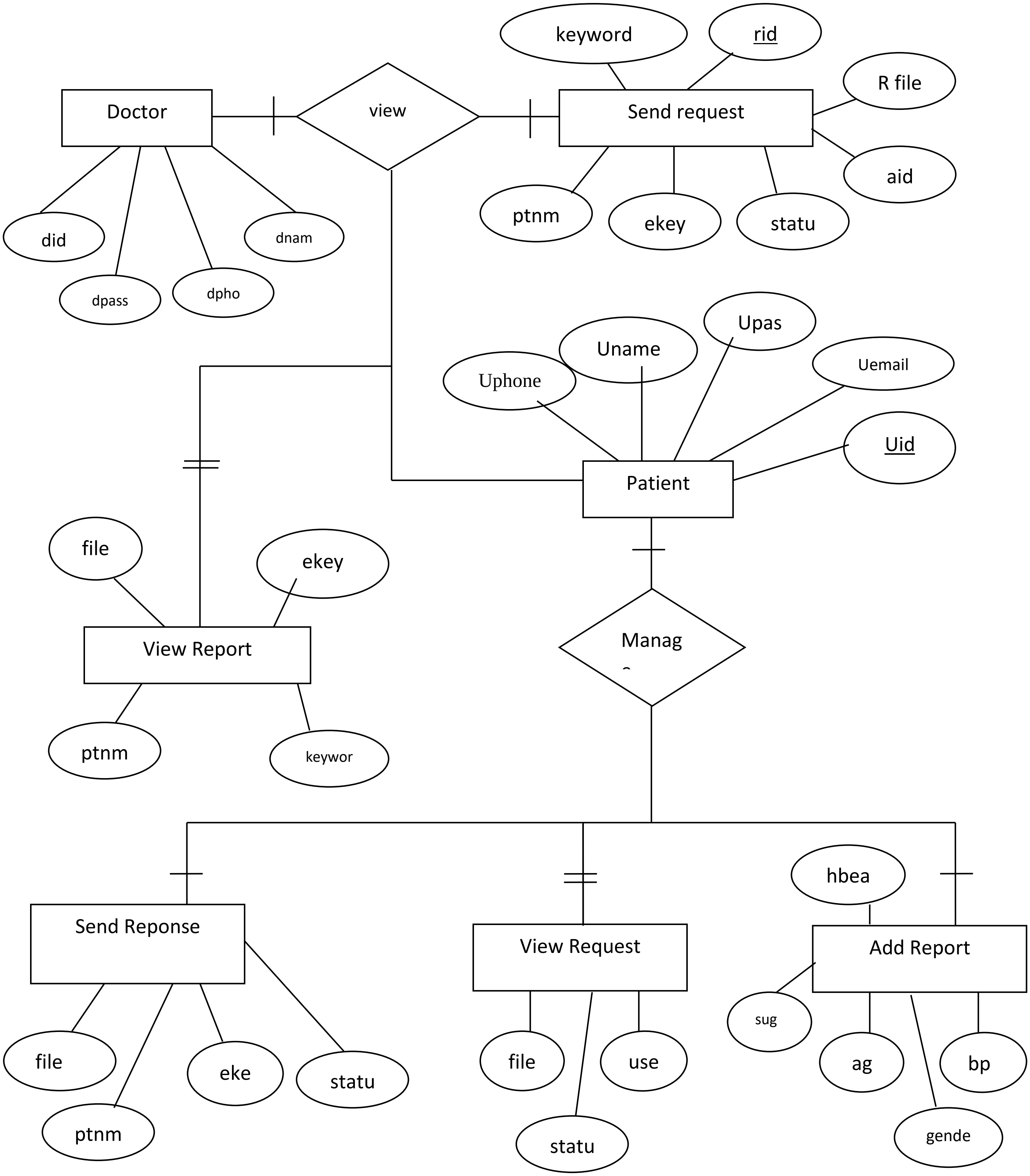
To reduce the server load, the project is designed in a way that most of the Validation of fields is done as client side validation, which will be more effective.

DATABASE DESIGN:

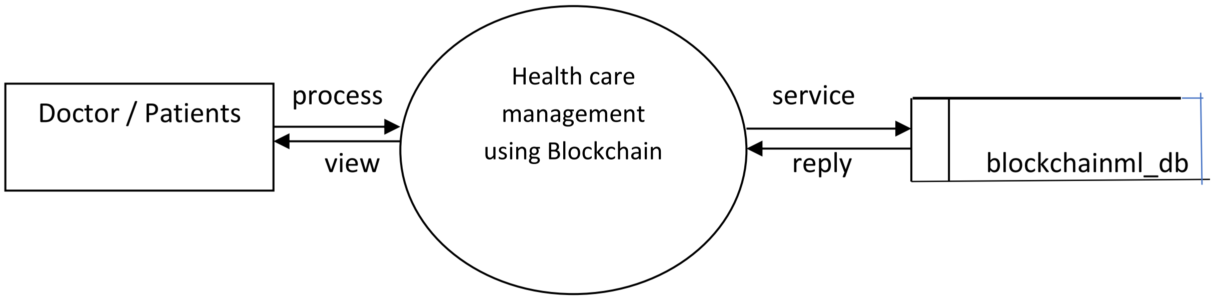
The database design involves creation of tables that are represented in physical database as stored files. They have their own existence. Each table constitute of rows and columns where each row can be viewed as record that consists of related information and column can be viewed as field of data of same type. The table is also designed with some position can have a null value.

The database design of project is designed in such a way values are kept without redundancy and with normalized format.

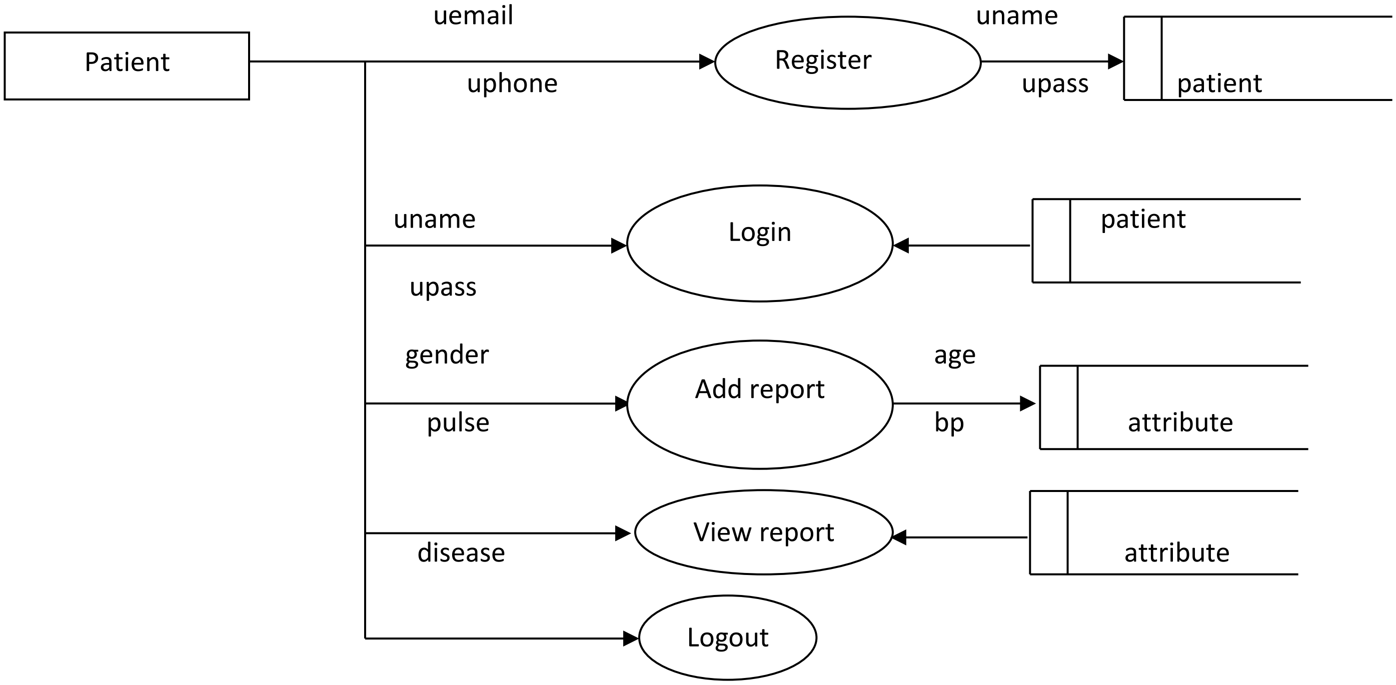
BLOCK DIAGRAM:



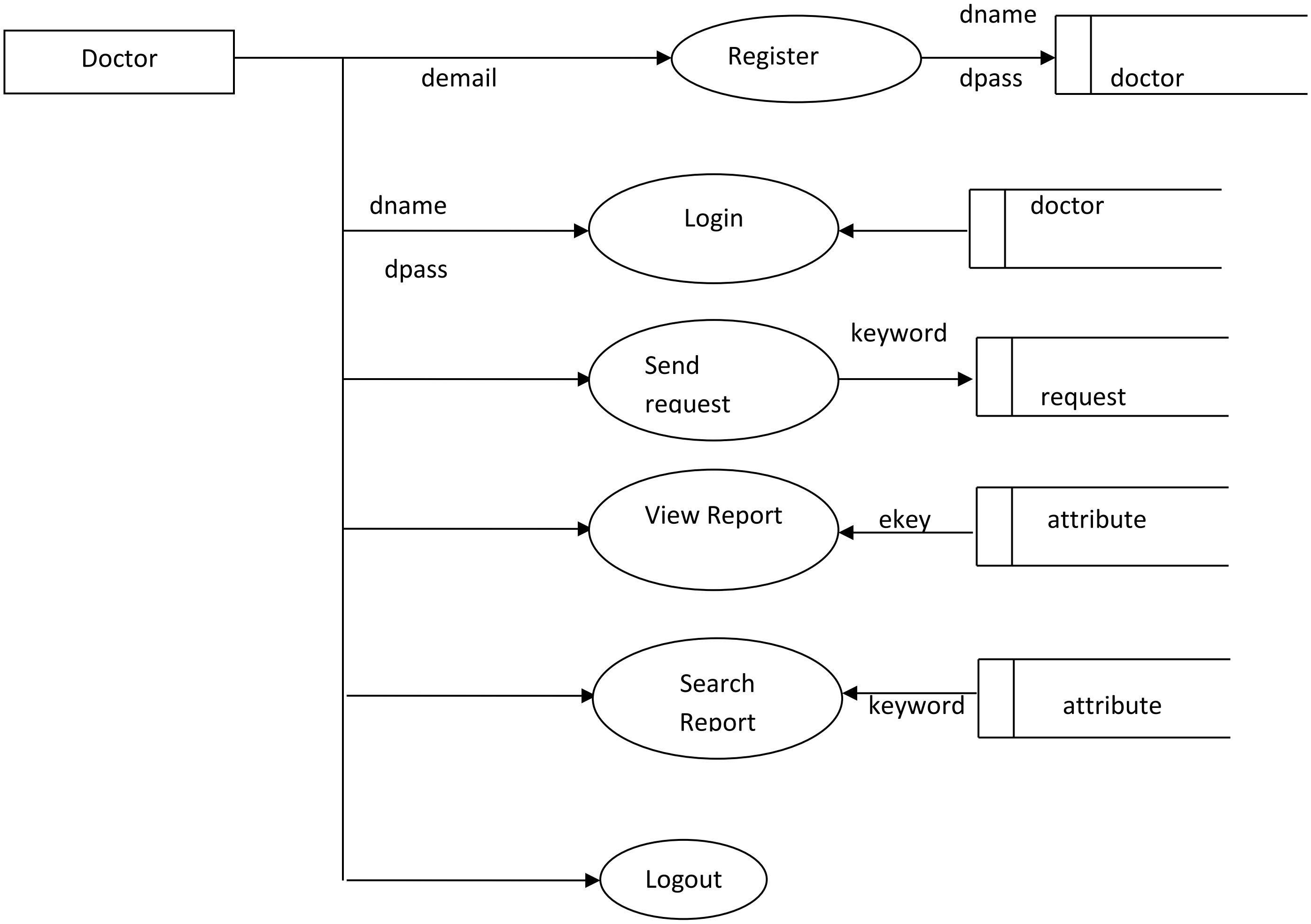
Level 0


Doctor / Patients
process
view
Health care management using Blockchain
service
reply
blockchainml_db

Patient:


Patient
uemail
uphone
Register
uname
upass
patient
uname
upass
Login
patient
gender
pulse
Add report
age
bp
attribute
disease
View report
attribute
Logout

Doctor:


Doctor
demail
Register
dname
dpass
doctor
dname
dpass
Login
doctor
Send request
keyword
request
View Report
ekey
attribute
Search Report
keyword
attribute
Logout

5.3 IMPLEMENTATION

DEVELOPMENT APPROACH

TOP-DOWN APPROACH

The importance of new system is that it is user friendly and a better interface with user's working on it. It can overcome the problems of manual system and the security problem.

Top-down approach of software development is the incremental approach to the construction of program structure. Modules are integrated by moving through the control hierarchy, beginning with the main control module. Module subordinate to the main control modules is incorporate into the structure in either a depth first or breadth first manner.

The top down approach is performed in a serious of five steps

1. The main module that is overall software is divided into five modules that are under the control of the main control module.
2. Depending on the top down approach selected subordinate stubs is replaced one at a time with actual components.
3. Tests are conducted as each component is integrated
4. On completion of each test another stub is replaced with real time component.
5. Regression testing may be conducted to ensure the new errors have not been introduced.

TESTING AND IMPLEMENTATION

SYSTEM TESTING

It is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because web-based system and applications reside on network and inter-operate with many different operating systems, browsers, hardware platforms and communication protocols. Thus, searching for errors is significant challenge for web applications.

Testing issues:

1. Client GUI should be considered.
2. Target environment and platform considerations
3. Distributed database considerations
4. Distributed processing consideration

TESTING AND METHODLOGIES:

System testing is the state of implementation, which is aimed at ensuring that the system works accurately and efficiently as expect before live operation, commences. It certifies that the whole set of programs hang together System testing requires a test plan that consists of several key activities and steps for run program, string, system and user acceptance testing. The implementation of newly design package is important in adopting a successful new system.

Testing is important stage in software development. System test is implementation should be a confirmation that all is correct and an opportunity to show the users that the system works as they expected It accounts the largest percentage of technical effort in software development process.

Testing phase is the development phase that validates the code against the functional specifications. Testing is a vital to the achievement of the system goals. The objective of testing is to discover errors. To fulfill this objective a series of test step such as the unit test, integration test, validation and system test where planned and executed.

Unit testing

Here each program is tested individually so any error apply unit is debugged. The sample data are given for the unit testing. The unit test results are recorded for further references. During unit testing the functions of the program unit validation and the limitations are tested.

Unit testing is testing changes made in a existing or new program this test is carried out during the programming and each module is found to be working satisfactorily. For example in the registration form after entering all the fields we click the submit button. When submit button is clicked, all the data in form are validated. Only after validation entries will be added to the database.

Unit testing comprises the set of tests performed by an individual prior to integration of the unit into large system. The situation is illustrated in as follows

Coding-> Debugging ->Unit testing -> Integration testing

The four categories of test that a programmer will typically perform on a program unit

1.Functional test
2.Performance test
3.Stress Test
4.Structure test

Functional test involve exercising the code with nominal input values for which the expected results are known as well as boundary values and special values.

Performance testing determines the amount of execution time spent in various parts of unit program through put and response time and device utilization by the program.

A variation of stress testing called sensitivity testing in same situations a very small range of data contained in a bound of valid data may cause extreme and even erroneous processing or profound performance degradation.

Structured testing is concerned with a exercising the internal logic of a program and traversing paths. Functional testing, stress testing performance testing are referred as “black box” testing and structure testing is referred as “white box” testing

VALIDATION TESTING

Software validation is achieved through a serious of testes that demonstrate conformity with requirements. Thus, the proposed system under consideration has been tested by validation & found to be working satisfactory.

OUTPUT TESTING

Asking the user about the format required by them tests the output generated by the system under consideration. It can be done in two ways, One on screen and other on printer format. The output format on the screen is found to be correct as the format designed n system test.

SYSTEM TESTING

In the system testing the whole system is tested for interface between each module and program units are tested and recorded. This testing is done with sample data. The securities, communication between interfaces is tested

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system although each test has a different purpose all work to verify that all system elements properly integrated and perform allocate function.

It involves two kinds of activities namely

1. Integrated testing
2. Acceptance testing

Integrated testing

Integrated testing is a systematic technique for constructing tests to uncover errors associated with interface.

Objective is to take unit tested modules and build a program structure that has been dictated by design

Acceptance testing

Acceptance testing involves planning an execution of a functional test, performance test and stress test to verify that the implemented system satisfies the requirement.

The acceptance testing is the final stage of the user the various possibilities of the data are entered and the results are tested.

Validation testing

Software validation is achieved through a series of test that demonstrates the conformity and requirements. Thus the proposed system under consideration has to be tested by validation and found to be working satisfactorily. For example in customer enters phone number field should contain number otherwise it produces an error message similarly in all the forms the fields are validated

Testing results

All the tests should be traceable to customer requirements the focus of testing will shift progressively from programs Exhaustive testing is not possible To be more effective testing should be which has probability of finding errors

The following are the attributes of good test

1. A good test has a probability of finding a errors

2. A good test should be “best of breeds”

3. A good test to neither simple nor too complex

QUALITY ASSURANCE

Quality assurance consists of the auditing and reporting functions of management. The goal of quality assurance is to provide management with the data entries necessary to be informed about the product quality thereby gaining the goal of insight and confidence that the product quality is meeting

Greater emphasis on quality in organization requires quality assurance. To be an integral part of the information system development .The development process must include checks throughout the process to ensure that the final product meets the original user requirements.

Quality assurance thus becomes an important component of the development process It’s included in the industry standard (IEEE 1993) On the development process quality assurance process is integrated into a linear development cycle through validation and verification performed at crucial system development steps .The goals of the management is to institute and monitor a quality assurance program with in the development process

Quality assurance induces

1. Validation of the system against requirements

2. Checks for errors in design documents and in the system itself

3. Quality assurance for usability

Quality assurance Goals:

Correctness: The extent to which the program meets the system specifications and user objectives

Reliability: The degree to which the system performs its intended functions overtime

Efficiency: The amount of computer resources required by a program to perform a function

Usability: The effort required learning and operating a system

Maintainability: To use with which program errors are located and corrected

Testability*:* The effort required a testing a program to ensure its correct performance

Portability: To ease of transporting a program from hardware configuration to another

Accuracy: The required position in input editing computation and output

GENERIC RISKS:

Risk identification is the systematic attempt to specify threats to the project plan (estimates the schedule resource overloading etc.). By identifying know and predictable risk the first step is to avoiding them. When possible and controlling them when necessary, there are two types of risk.

1. Generic Risk
2. Product specific risk

Generic risks are potential threats to every software project. Only those with a clear understanding of technology can identify product specific risk The people and the environment that is specific to the project at a hand and to identify the product specific risk and the project the plan and the software statement of scope are examined and answer to the following question is developed.

What special characteristics of this product may threaten the project plan.

One method for identifying risk is to create a risk item and checklists. The checklist can be used for risk identification and focus on some subset to know and predictable risk in the following sub categories.

1. Product risk
2. Risk associated with overall size of software to built or modified
3. Business imparts
4. Risk associated with constraints imposed with management
5. Customer characteristics

Risk associated with sophisticated of the customer and developers ability to communicate with the customer in a timely manner.

Different categories of risks are considered

Project Risks

It identify a potential budgetary, schedule, personnel like staffing, organizing, resource, customer requirement, problems and their impact on a software project

Technical risks

Technical risks identify potential design implementation interface, verification, and maintenance problems.

SECURITY TECHNOLOGIES AND POLICIES:

Any system developed should be secured & protected against possible hazards. Security measures are provided to prevent unauthorized access to database at various levels. Password protection & simple procedures to change the unauthorized access are provided to the users.

The user will have to enter the user name and password and if it is validated he can participate in auction. Otherwise if he/she is a new user he should get registered and then he can place an order

When he/she registered they should provide authentication through jpg files (like ration card Xerox, voter identity card Xerox). A multi layer security architecture comprising firewalls filtering routers encryption & digital certification must be assured in this project in real time that order details are protected from unauthorized access.

IMPLEMENTATION:

Implementation is the stage in the project where the theoretical design is turned into a working system. The most crucial stage is achieving a successful new system and giving a user confidence in that the new system will work efficiently and effectively in the implementation stage. The stage consists of

- Testing a developed program with sample data
- Detection and correction of error
- Creating whether the system meets a user requirement.
- Making necessary changes as desired by users.
- Training user personal.

IMPLEMENTATION PROCEDURES

The implementation phase is less creative than system design. A system design may be dropped at any time prior to implementation, although it becomes more difficult when it goes to the design phase. The final report of the implementation phase includes procedural flowcharts, record layouts, and a workable plan for implementing the candidate system design into a operational design.

USER TRAINING

It is designed to prepare the users for testing & converting the system. There is several ways to trail the users they are:

1) User manual

2) Help screens

3) Training demonstrations.

1) User manual:

The summary of important functions about the system & software can be provided as a document to the user. User training is designed to prepare the user for testing and convening a system

The summary of important functions about the system and the software can be provided as a document to the user

1. Open http page
2. Type the file name with URL index .php in the address bar
3. Index. php is opened existing user the type the username and password
4. Click the submit button

2) Help screens:

This features now available in every software package, especially when it is used with a menu. The user selects the “Help” option from the menu. The System success the necessary description or information for user reference.

3) Training demonstration:

Another user training element is a training demonstration. Live demonstration with personal contact is extremely effective for training users.

OPERATIONAL DOCUMENTATION:

Documentation means of communication; it establishes the design and performance criteria of the project. Documentation is descriptive information that portrays the use and /or operation of the system. The user will have to enter the user name and password if it is valid he participate in auction. Otherwise if it is new user he needs to register

Documentation means of communication; it establishes design & performance criteria for phases of the project. Documentation is descriptive information that portrays he use &/or operation of the system.

1) Documentation tools:

Document production & desktop publishing tool support nearly ever aspect of software developers. Most software development organizations spend a substantial amount of time developing documents, and in many cases the documentation process itself is quite inefficient. It is not use unusual for a software development effort on documentation. For this reason, Documentation tools provide an important opportunity to improve productivity.

2) Document restructuring:

Creating document is far too timed consuming. If the system work's, we'll live with what we have. In some cases, this is the correct approach. It is not possible to recreate document for hundreds of computer programs.

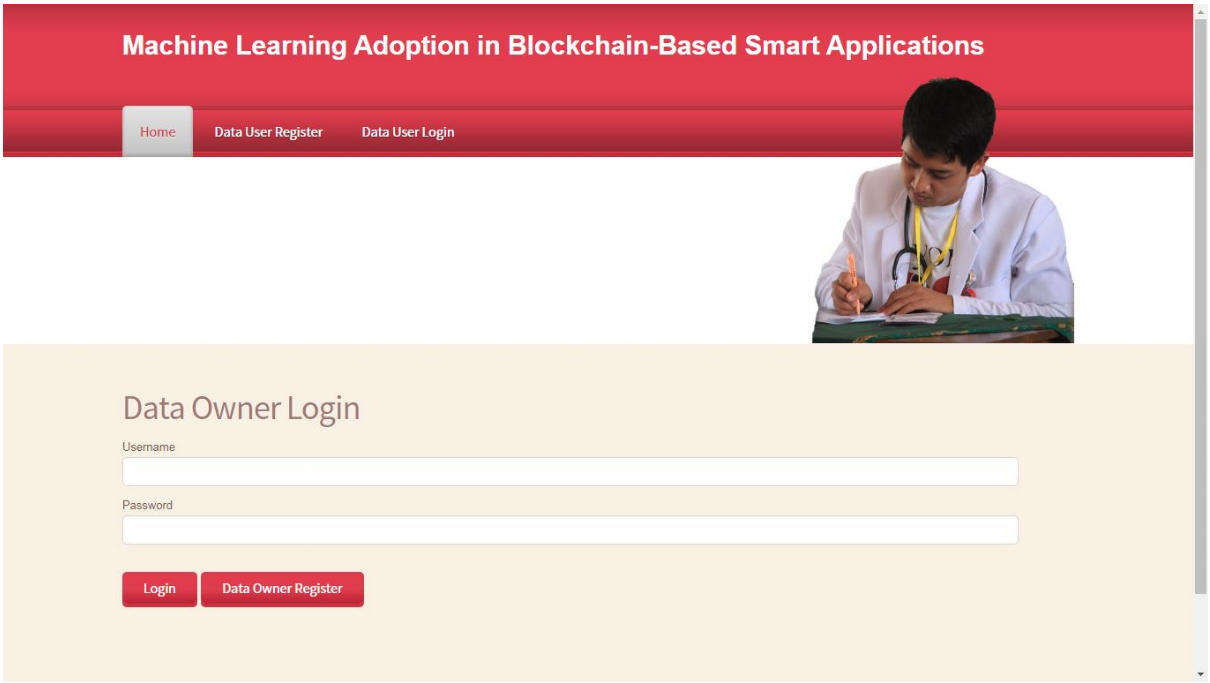
Documentation must be updated, but we have limited resources. It may not be necessary to fully redocument an application. Rather, those portions of the system that are currently undergoing change are fully documented.

The system is business critical and must be fully redocumented. Even in this case, an intelligent approach is to pare documentation to an essential minimum.

RESULTS:

User module:

Login:



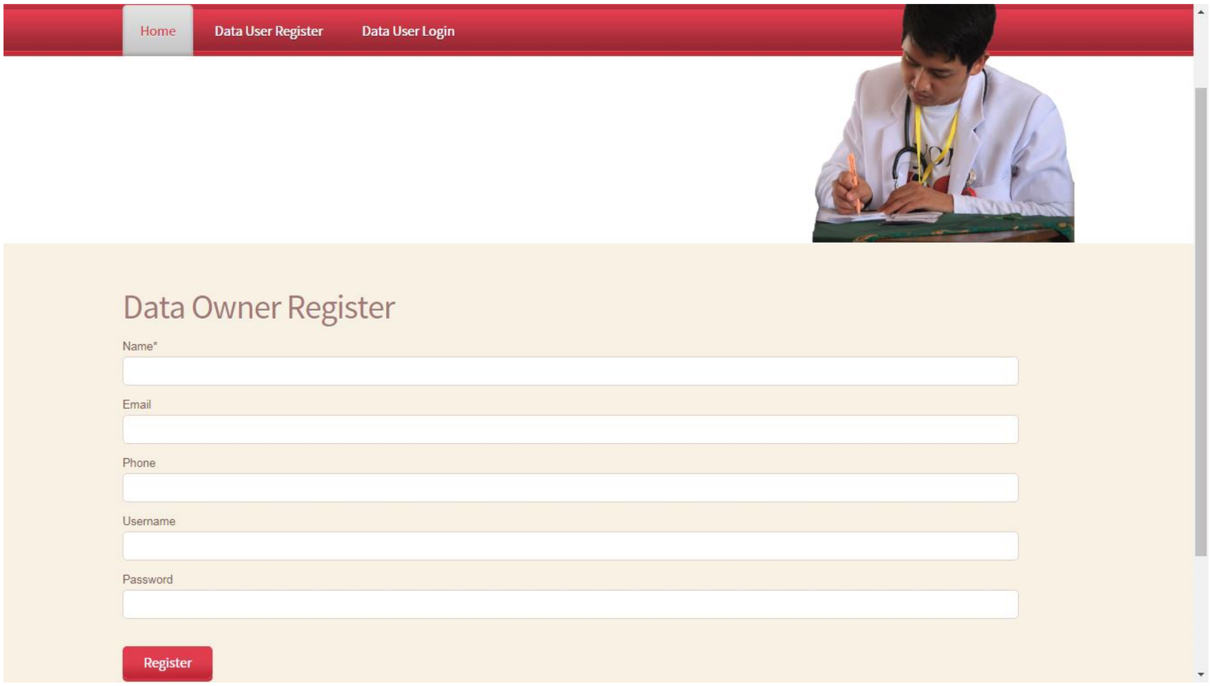
Machine Learning Adoption in Blockchain-Based Smart Applications

Data Owner Login

Username

Password

Register:



Data Owner Register

Name*

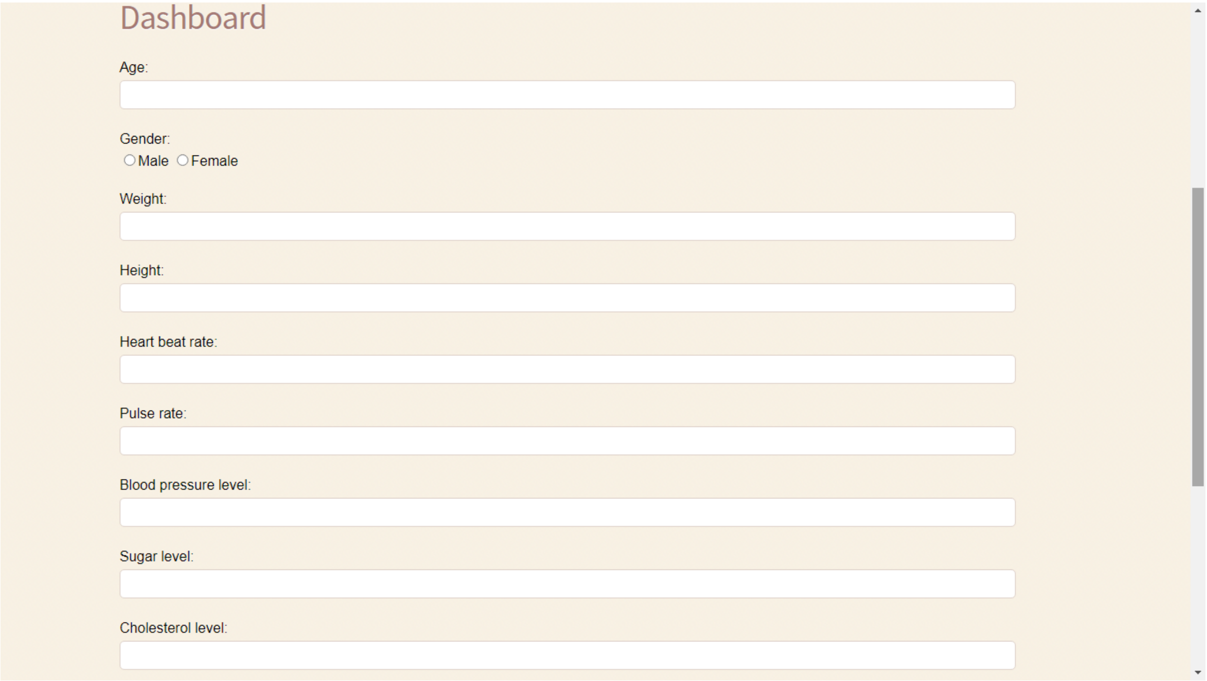
Email

Phone

Username

Password

Add health details:



Dashboard

Age:

Gender:
○Male ○Female

Weight:

Height:

Heart beat rate:

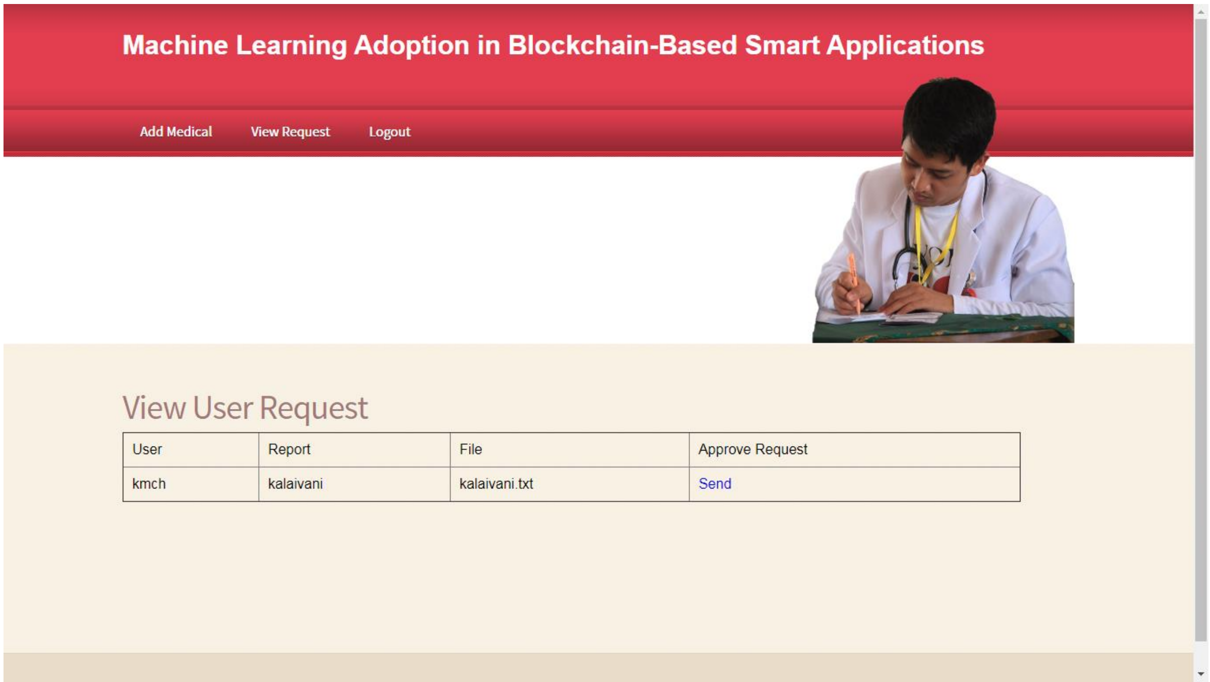
Pulse rate:

Blood pressure level:

Sugar level:

Cholesterol level:

View request send by doctor:

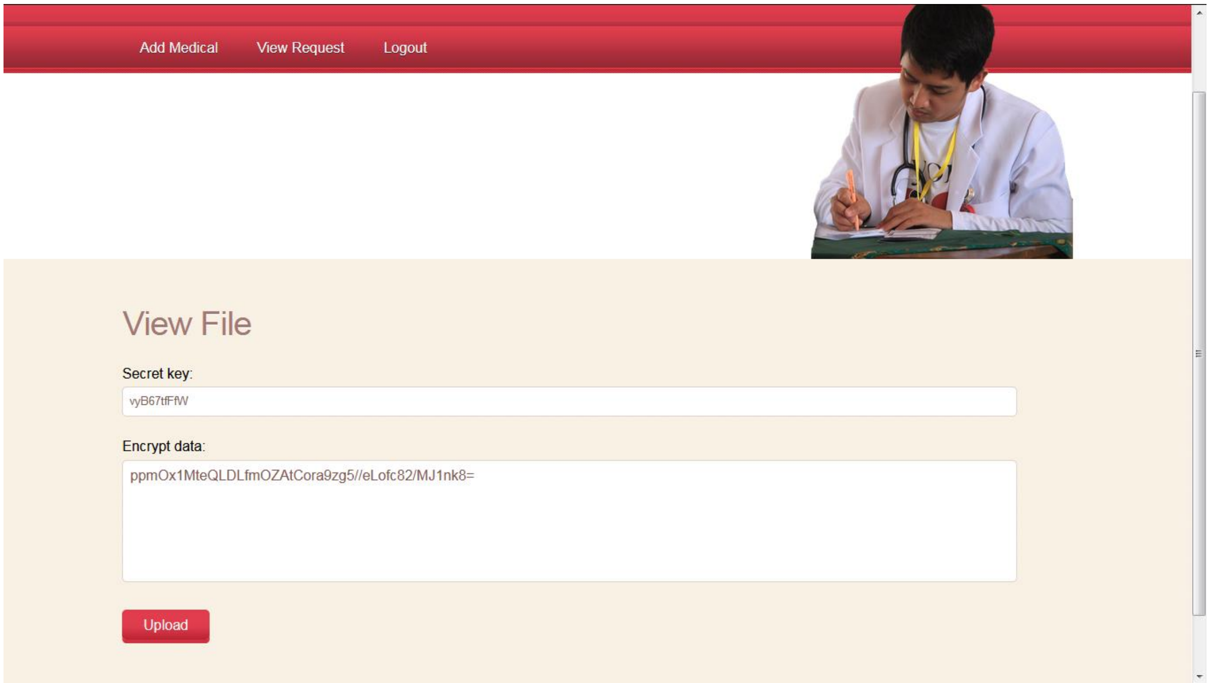


Machine Learning Adoption in Blockchain-Based Smart Applications

View User Request

User	Report	File	Approve Request
kmch	kalaivani	kalaivani.txt	Send

File encryption:



View File

Secret key:

vyB67tfFfW

Encrypt data:

ppmOx1MteQLDLfmOZAtCora9zg5//eLofc82/MJ1nk8=

CONCLUSION

Blockchain Technology gives great number of opportunities if utilized properly and is seen beyond bitcoin. With blockchain, the dominance of central authority could be eliminated and so the commission. Machine Learning models can directly feed with data (however the rights will be managed by central authority). This will increase the accuracy and efficiency of machine learning models and so their usability. Healthcare industry directly correlates with the life of a person. This could help patients as well as doctors. we proposed a fully secure protocol on encrypted data. It ensures semantic security in that the data servers knows nothing about the data including indirect data patterns, query, as well as the query result. In addition, the client and data owner do not need to participate in the computation. We also presented a secure dominance protocol which can be used by encrypted queries as well as other queries. Furthermore, we demonstrated, data encryption to further reduce the computation load. Finally, we presented our implementation of the protocol and demonstrated the feasibility and efficiency of the solution.

FUTURE SCOPE

As for future work, we plan to optimize the communication time complexity to further improve the performance of the protocol. In terms of future scope the practical implementation of this model will be there. This model can be further extended for Inventory prevent fraud. The recent advancements in Blockchain and ML have made them path-breaking technologies. The distributed ledger has the possibility to work as the backbone of various smart applications such as smart cities, UAV, SG, data trading.

APPENDIX

User Module:

Login

```
<?php include "header.php"; ?>

<div class="s_bg">

<div class="wrap">

<!---728x90--->

<div class="cont_main">

<div class="contact">

<div class="section group">

<div class="col span_2_of_3">

 <div class="contact-form">

<h3>Data Owner Login</h3>

<form method="post" action="">

<div>

<span><label>Username</label></span>

<span><input name="uname" type="text" class="textbox"></span>

 </div>


          <!---728x90--->

 <div>

<span><label>Password</label></span>
```

```
<span><input name="upass" type="password" class="textbox"></span>
  </div>


   <div>
<span><input type="submit" value="Login" name="submit"></span>
 <a href="register.php"><span><input type="button" value="Data Owner
Register" name=""></span></a>
  </div>
    </form>
   </div>
</div>
  <div class="clear"></div>
 </div>
  <!---728x90--->
</div>
</div>
</div>
</div>
<?php
if(isset($_POST['submit']))
{
$uname=$_POST['uname'];
$upass=$_POST['upass'];
$con=mysql_query("select * from patient where uname='$uname' and
upass='$upass'")or die(mysq_error());
$n=mysql_num_rows($con);
if($n>0)
{
$_SESSION['uname']=$uname;
echo '<script type="text/javascript">alert("Data Owner logged in
successfully")</script>';
echo '<meta http-equiv="refresh" content="0;url=dashboard.php">';
}
else
{
echo '<script type="text/javascript">alert("Data Owner login details
mismatch")</script>';
}
}
?>
<?php include "footer.php"; ?>
```

Register:

```
<?php include "header.php"; ?>
<div class="s_bg">
<div class="wrap">
<!---728x90--->
<div class="cont_main">
<div class="contact">
<div class="section group">
<div class="col span_2_of_3">
 <div class="contact-form">
<h3>Data Owner Register</h3>
 <form method="post" action="">
                         <div>
<span><label>Name*</label></span>
<span><input name="pname" type="text" class="textbox" required></span>
</div>
  <div>
<span><label>Email</label></span>
<span><input name="uemail" type="email" class="textbox"></span>
 </div>
<!---728x90--->
<div>
<span><label>Phone</label></span>
```

```
<span><input name="uphone" type="text" class="textbox" required></span>
</div>



<div>
<span><label>Username</label></span>
<span><input name="uname" type="text" class="textbox" required></span>
 </div>
<!---728x90--->
<div>
<span><label>Password</label></span>
<span><input name="upass" type="password" class="textbox" required></span>
  </div>

  <div>
<span><input type="submit" value="Register" name="submit"></span>

 </div>
  </form>
 </div>
  </div>

<div class="clear"></div>
```

```
 </div>
  <!---728x90--->
</div>
</div>
</div>
</div>
<?php
if(isset($_POST['submit']))
{
$pname=$_POST['pname'];
$uemail=$_POST['uemail'];
$uphone=$_POST['uphone'];
$uname=$_POST['uname'];
$upass=$_POST['upass'];
$con=mysql_query("select * from patient where uname='$uname' and
upass='$upass'")or die(mysq_error());
$n=mysql_num_rows($con);
if($n>0)
{
echo '<script type="text/javascript">alert("Data Owner account already
registered")</script>';
}
else
{
mysql_query("insert into patient(pname,uemail,uphone,uname,upass)values('$pname','$uemail','$uphone','$uname','$upass')")or die(mysq_error());
echo '<script type="text/javascript">alert("Data Owner account registered successfully")</script>';
echo '<meta http-equiv="refresh" content="0;url=index.php">';
}
}
?>
<?php include "footer.php"; ?>
```

Add health details

```
<?php include "header.php"; ?>
<div class="s_bg">
<div class="wrap">
<!---728x90--->
<div class="cont_main">
<div class="contact">
<div class="section group">
<div class="col span_2_of_3">
<div class="contact-form pdb">
<h3>Patient dashboard</h3>
<form method="post" action="">
```

```
 <div>
<span>Your age</span><br/>
   <span><input name="age" type="radio" class="textbox" value="30"> <30
 <input name="age" type="radio" class="textbox" value="31"> >30
   </span><br/><br/>
</div>

<div>
<span>Gender</span><br/>
<span><input name="gender" type="radio" class="textbox" value="male"> Male
<input name="gender" type="radio" class="textbox" value="female"> Female
 </span><br/><br/>
    </div>

<div>
You have smoking habit<br/>
<span><input name="smoking" type="radio" class="textbox" value="1"> Yes
 <input name="smoking" type="radio" class="textbox" value="0"> No
 </span><br/><br/>
</div>
 <div>
You   have Alcohol Intake
<br/>
```

```
<span><input name="alcohol" type="radio" class="textbox" value="1"> Yes
 <input name="alcohol" type="radio" class="textbox" value="0"> No
</span><br/><br/>
</div>
<div>
Do you Exercise Regularly<br/>
<span><input name="exercise" type="radio" class="textbox" value="0"> Yes
 <input name="exercise" type="radio" class="textbox" value="1"> No
 </span><br/><br/>
</div>
<div>
Do you intake fast food    <br/>
<span><input name="fast_food" type="radio" class="textbox" value="1"> Yes
 <input name="fast_food" type="radio" class="textbox" value="0"> No
  </span><br/><br/>
</div>
<div>
Do you have history of Hereditary heart disease <br/>
<span><input name="hereditary" type="radio" class="textbox" value="1"> Yes
 <input name="hereditary" type="radio" class="textbox" value="0"> No
</span><br/><br/>
</div>
```

```
<div>

Are you overweight

<br/>

<span><input name="overweight" type="radio" class="textbox" value="1"> Yes

 <input name="overweight" type="radio" class="textbox" value="0"> No

</span><br/><br/>

</div>

<div>

Are you mentally stressed

<br/>

<span><input name="stress" type="radio" class="textbox" value="1"> Yes

 <input name="stress" type="radio" class="textbox" value="0"> No

 </span><br/><br/>

</div>

<div>

Your heart beat rate

<br/>

<span><input name="heart_beat" type="radio" class="textbox" value="0"> <60bpm

<input name="heart_beat" type="radio" class="textbox" value=""> 60-100bpm<input name="heart_beat" type="radio" class="textbox" value="1"> >100bpm

</span><br/><br/>
```

```
</div>
<div>
You have chest pain
<br/>
<span><input name="chest_pain" type="radio" class="textbox" value="1"> Yes
                         <input name="chest_pain" type="radio" class="textbox"
value="0"> No
 </span><br/><br/>
</div>
<div>
You have Fasting Blood Sugar
<br/>
<span><input name="blood_sugar" type="radio" class="textbox" value="1">
>120 mg/dl
<input name="blood_sugar" type="radio" class="textbox" value="0"> <120 mg/dl
 </span><br/><br/>
</div>
<div>
You have Blood Pressure<br/>
<span><input name="blood_pressure" type="radio" class="textbox" value="1">
>130
 <input name="blood_pressure" type="radio" class="textbox" value="0"> <130
 </span>
```

```
</div>
<div>
<span><input type="submit" value="Submit" name="submit"></span>
 </div>
  </form>
  </div>
  </div>
<div class="clear"></div>
 </div>
 <!---728x90--->
</div>
</div>
</div>
</div>
<?php
if(isset($_POST['submit']))
{
$pname=$_SESSION['uname'];
$age=$_POST['age'];
$gender=$_POST['gender'];
$smoking=$_POST['smoking'];
$alcohol=$_POST['alcohol'];
$exercise=$_POST['exercise'];
```

```
$fast_food=$_POST['fast_food'];

$hereditary=$_POST['hereditary'];

$overweight=$_POST['overweight'];

$stress=$_POST['stress'];

$heart_beat=$_POST['heart_beat'];

$chest_pain=$_POST['chest_pain'];

$blood_sugar=$_POST['blood_sugar'];

$blood_pressure=$_POST['blood_pressure'];

$con=mysql_query("select * from history where pname='$pname'")or
die(mysq_error());

$n=mysql_num_rows($con);

if($n>0)

{

echo '<script type="text/javascript">alert("Patient history already
added")</script>';

}

else

{

mysql_query("insert into
history(pname,age,gender,smoking,alcohol,exercise,fast_food,hereditary,overweig
ht,stress,heart_beat,chest_pain,blood_sugar,blood_pressure)values('$pname','$age','
$gender','$smoking','$alcohol','$exercise','$fast_food','$hereditary','$overweight','$s
tress','$heart_beat','$chest_pain','$blood_sugar','$blood_pressure')")or
die(mysq_error());
echo '<script type="text/javascript">alert("Patient history added successfully")</script>';

echo '<meta http-equiv="refresh" content="0;url=view-prediction.php">';

}

}

?>

<?php include "footer.php"; ?>
```

View Request sent by doctor

```
<?php include "header.php"; ?>
<div class="s_bg">
<div class="wrap">
<!---728x90--->
<div class="cont_main">
<div class="contact">
<div class="section group">
<div class="col span_2_of_3">
 <div class="contact-form pdb">
<h3>View User Request</h3>
<table border="1" cellpadding="10" cellspacing="0" width="100%">
<thead><tr><td>User</td><td>Report</td><td>File</td><td>Approve Request</td></tr></thead>
 <?php
 $ptnm=$_SESSION['uname'];
$hq=mysql_query("select * from request where ptnm='$ptnm'");

        while($hr=mysql_fetch_array($hq))
                        {
                        $dname=$hr['dname'];
                        $aid=$hr['aid'];
                        $rfile=$hr['rfile'];
```

```
                              $rname=$hr['rname'];
                              echo
"<tr><td>$dname</td><td>$rname</td><td>$rfile</td><td><a
href='view_request.php?action=approve&aid=$aid'>Send</a></td></tr>";
                               }


?>
</table>
 <br><br><br>
  </div>
  </div>
<div class="clear"></div>
</div>
 <!---728x90--->
</div>
</div>
</div>
</div>
<?php
if($_GET['action']=='approve')
{
$aid=$_GET['aid'];
mysql_query("update request set status='1' where aid='$aid'");
```

```
echo "<script type='text/javascript'>alert('File access request approved');</script>";

}

?>

<?php include "footer.php"; ?>
```

File Encryption

```
<?php include "header.php"; ?>

<div class="s_bg">

<div class="wrap">

<!---728x90--->

<div class="cont_main">

<div class="contact">

<div class="section group">

<div class="col span_2_of_3">

 <div class="contact-form pdb">

<h3>Dashboard</h3>

<form id="ContactForm" action="" method="post" enctype="multipart/form-data">

<?php

function generateRandomString($length = 10) {

    $characters = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';

    $charactersLength = strlen($characters);

    $randomString = '';
    for ($i = 0; $i < $length; $i++) {
$randomString .= $characters[rand(0, $charactersLength - 1)];
    }
    return $randomString;
}
?>
 <div>
 <div class="wrapper"> <span>Age:</span>
<input type="text" class="input" name='age'>
 </div>
<div class="wrapper"> <span>Gender:</span>
<input type="radio" name='gender' value="male">Male
<input type="radio" name='gender' value="female">Female
 </div>
  <div class="wrapper"> <span>Weight:</span>
<input type="text" class="input" name='weight'>
 </div>
 <div class="wrapper"> <span>Height:</span>
  <input type="text" class="input" name='height'>
</div>
  <div class="wrapper"> <span>Heart beat rate:</span>
<input type="text" class="input" name='hbeat'>
 </div>
 <div class="wrapper"> <span>Pulse rate:</span>
  <input type="text" class="input" name='prate'>
  </div>
 <div class="wrapper"> <span>Blood pressure level:</span>
<input type="text" class="input" name='bp'>
 </div>
 <div class="wrapper"> <span>Sugar level:</span>
 <input type="text" class="input" name='sugar'>
  </div>
 <div class="wrapper"> <span>Cholesterol level:</span>
 <input type="text" class="input" name='cholesterol'>
</div>
  <div class="wrapper"> <span>Report name:</span>
 <input type="text" class="input" name='rname'>
</div>
 <div class="wrapper"> <span>Report keyword:</span>
 <input type="text" class="input" name='rkeyword'>
 </div>
<label>Upload report:</label><br><br>
<input type="file" value="" name="rfile" required><br>
```

```
<input type="hidden" name="ekey" value="<?php echo generateRandomString(); ?>" />

 <input type="submit" name="submit" value=" Add "  class="button"></div>

</form>

 </div>

</div>

<div class="clear"></div>

 </div>

<!---728x90--->

</div>

</div>

</div>

</div>

<?php

if(isset($_POST['submit']))

{

$age=$_POST['age'];

$gender=$_POST['gender'];

$weight=$_POST['weight'];

$height=$_POST['height'];

$hbeat=$_POST['hbeat'];

$prate=$_POST['prate'];

$bp=$_POST['bp'];
```

```
$sugar=$_POST['sugar'];
$cholesterol=$_POST['cholesterol'];
$rname=$_POST['rname'];
$ekey=$_POST['ekey'];
$rkeyword=$_POST['rkeyword'];
$rfile=$_FILES['rfile']['name'];
move_uploaded_file($_FILES['rfile']['tmp_name'],"upload/$rfile");
class AES
{
     const M_CBC = 'cbc';
    const M_CFB = 'cfb';
    const M_ECB = 'ecb';
    const M_NOFB = 'nofb';
    const M_OFB = 'ofb';
    const M_STREAM = 'stream';

    protected $key;
    protected $cipher;
    protected $data;
    protected $mode;
    protected $IV;
  /**
  *
```

```
    * @param type $data

    * @param type $key

    * @param type $blockSize

    * @param type $mode

    */

      function __construct($data = null, $key = null, $blockSize = null, $mode =
null) {

        $this->setData($data);

        $this->setKey($key);

        $this->setBlockSize($blockSize);

        $this->setMode($mode);

        $this->setIV("");

      }


    /**

    *

    * @param type $data

    */

      public function setData($data) {

        $this->data = $data;

      }


    /**
```

```
*
* @param type $key
*/
  public function setKey($key) {
    $this->key = $key;
  }
 /**
   *
    * @param type $blockSize
    */
     public function setBlockSize($blockSize) {
     switch ($blockSize) {
      case 128:
      $this->cipher = MCRYPT_RIJNDAEL_128;
      break;

      case 192:
      $this->cipher = MCRYPT_RIJNDAEL_192;
      break;

      case 256:
      $this->cipher = MCRYPT_RIJNDAEL_256;
      break;
    }
  }

/**
*
* @param type $mode
*/
  public function setMode($mode) {
    switch ($mode) {
      case AES::M_CBC:
      $this->mode = MCRYPT_MODE_CBC;
      break;
      case AES::M_CFB:
      $this->mode = MCRYPT_MODE_CFB;
      break;
      case AES::M_ECB:
      $this->mode = MCRYPT_MODE_ECB;
      break;
      case AES::M_NOFB:
      $this->mode = MCRYPT_MODE_NOFB;
      break;
      case AES::M_OFB:
      $this->mode = MCRYPT_MODE_OFB;
      break;
      case AES::M_STREAM:
      $this->mode = MCRYPT_MODE_STREAM;
      break;
      default:
      $this->mode = MCRYPT_MODE_ECB;
      break;
    }
  }

/**
*
* @return boolean
*/
  public function validateParams() {
    if ($this->data != null &&
        $this->key != null &&
        $this->cipher != null) {
      return true;
    } else {
      return FALSE;
    }
  }
```

```
      public function setIV($IV) {
            $this->IV = $IV;
        }


      protected function getIV() {
          if ($this->IV == "") {
            $this->IV = mcrypt_create_iv(mcrypt_get_iv_size($this->cipher, $this-
>mode), MCRYPT_RAND);
          }
          return $this->IV;
      }


    /**
    * @return type
    * @throws Exception
    */
      public function encrypt() {


        if ($this->validateParams()) {
          return trim(base64_encode(
            mcrypt_encrypt(
              $this->cipher, $this->key, $this->data, $this->mode, $this->getIV())));
    } else {
     throw new Exception('Invlid params!');
    }
   }
  /**
  *
  * @return type
  * @throws Exception
  */
   public function decrypt() {
    if ($this->validateParams()) {
     return trim(mcrypt_decrypt(
      $this->cipher, $this->key, base64_decode($this->data), $this->mode, $this->getIV()));
    } else {
     throw new Exception('Invlid params!');
    }
   }


  }


$myfile = fopen("upload/$rfile", "r") or die("Unable to open file!");
$inputText = fread($myfile,filesize("upload/$rfile"));
```

```
$inputKey = "$ekey";
$blockSize = 256;
$aes = new AES($inputText, $inputKey, $blockSize);
$enc = $aes->encrypt();
$aes->setData($enc);
$encmyfile = fopen("upload/$rfile", "w") or die("Unable to open file!");
$txt = "$enc";
fwrite($encmyfile, $txt);
echo "<a href='upload/$rfile' target='_blank'><input type='button' value='Click here to view Encrypted file' /></a>";
$ptnm=$_SESSION['uname'];
$_SESSION['rfile']=$rfile;
$q=mysql_query("select * from  attribute where ptnm='$ptnm'")or die(mysql_error());
$n=mysql_num_rows($q);
if($n>0)
{
/*echo "<script type='text/javascript'>alert('Patient health details already added');</script>";*/
mysql_query("update  attribute set age='$age',gender='$gender',weight='$weight',height='$height',hbeat='$hbeat',prate='$prate',bp='$bp',sugar='$sugar',cholesterol='$cholesterol',rname='$rname',rkeyword='$rkeyword',rfile='$rfile',ekey='$ekey' where ptnm='$ptnm'")or die(mysql_error());
echo "<script type='text/javascript'>alert('Health details updated successfully');</script>";
echo '<meta http-equiv="refresh" content="0;url=view_file.php">';
}
else
{
mysql_query("insert into attribute(age,gender,weight,height,hbeat,prate,bp,sugar,cholesterol,ptnm,rname,rkeyword,rfile,ekey)values('$age','$gender','$weight','$height','$hbeat','$prate','$bp','$sugar','$cholesterol','$ptnm','$rname','$rkeyword','$rfile','$ekey')")or die(mysql_error());
echo "<script type='text/javascript'>alert('Health details added successfully');</script>";
echo '<meta http-equiv="refresh" content="0;url=view_file.php">';
}
}
?>
<?php include "footer.php"; ?>
```

REFERENCES:

BOOK REFERENCES:

1. Pinyaphat Tasatanattakool and Chian Techapanupreeda, “Blockchain: Challenges and Applications”
2. G. Prisco (2016, April), The Blockchain for Healthcare: Gem Launches Gem Health Network With Philips Blockchain Lab, [Online]. Available: https://bitcoinmagazine.com/articles/theblockchain-for-heathcare-gemlaunches-gem-health-networkwith-philips-blockchain-lab-1461674938
3. Matthias Mettler, “Blockchain Technology in Healthcare”, 2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)
4. Beginning PHP and MySQL: From Novice to Professional, Third Edition
5. Practical Web 2.0 Applications with PHP

WEBSITES:

1. http://php.net/manual/en/tutorial.php
2. https://www.guru99.com/php-tutorials.html
3. https://www.w3schools.com/php/default.asp
4. https://www.homeandlearn.co.uk/php/php.html
5. https://teamtreehouse.com/tracks/beginning-php