A
Project Report On

"Authenticated Medical Documents Releasing with Privacy Protection and Release Control"

Submitted in Partial Fulfillment of the requirements for the award of the degree

DIPLOMA
In
COMPUTER SCIENCE ENGINEERING
(DCSE)
From
STATE BOARD OF TECHNICAL EDUCATION AND TRAINING
HYDERABAD
By

VAMSHI KRISHNA
22383-CS-096

Under The Guidance of

FATHIMA MADAM
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
MOTHER THERESSA COLLEGE OF ENGINEERING & TECHNOLOGY
(Approved by AICTE, New Delhi & Affiliated to SBTET, Hyderabad)
Peddapalli (Dt). Telangana. India -505 174

MOTHER THERESSA COLLEGE OF ENGINEERING & TECHNOLOGY

(Approved by AICTE, New Delhi & Affiliated to SBTET, Hyderabad)

Peddapalli (Dt). Telangana. India -505 174

CERTIFICATE

This is to certify that the Project entitled "Authenticated Medical Documents Releasing with Privacy Protection and Release Control" is a bonafied work done and submitted by

VAMSHI KRISHNA	22383-CS-096
PRAVEEN	22383-CS-097
ARUN	22383-CS-085
AKHILA	22383-CS-076
SAI KEERTHANA	22383-CS-055
G PRIYANKA	22383-CS-064

for the partial fulfillment of the requirement for the award of Degree of Diploma in Computer Science Engineering (DCSE) from SBTET – HYD.

FATHIMA MADAM
INTERNAL GUIDE

SHASHI
HEAD OF THE DEPT.

EXTERNAL EXAMINER

DR. T.SRINIVAS
PRINCIPAL

DECLARATION

I am hereby declare that the entire work embodied in this project entitled **"Authenticated Medical Documents Releasing with Privacy Protection and Release Control"** has been carried out by me. No part of it has been submitted for the award of any Degree or Diploma at any other University or Institution. I, further declare that this project dissertation is based on my work carried out at "**MOTHER THERESSA COLLEGE OF ENGINEERING & TECHNOLOGY"** in the final year Diploma course.

Date:

Signature

Place: Peddapalli

VAMSHI KRISHNA	22383-CS-096
PRAVEEN	22383-CS-097
ARUN	22383-CS-085
AKHILA	22383-CS-076
SAI KEERTHANA	22383-CS-055
G PRIYANKA	22383-CS-064

ACKNOWLEDGEMENT

I express my sincere gratitude and thankfulness towards **FATHIMA MADAM** Assistant Professor for her valuable time and guidance throughout this project work.

I feel privileged to thank **SHASHI** Head of the DCSE Department, for his encouragement during the progress of this project

I feel privileged to offer my sincere thanks and deep sense of gratitude to Principal **Dr. T. SRINIVAS***,* for expressing their confidence in me by letting me work on a project of this magnitude and using latest technologies and providing their support, help & encouragement in completing this project.

I also thankful for the constant encouragement, support and all sorts of help from both **Teaching** and **Non-Teaching staff** of **DCSE DEPARTMENT** to bring out this project work successfully.

I am thankful to **MOTHER THERESSA COLLEGE OF ENGINEERING & TECHNOLOGY** for providing required facilities during the project work.

VAMSHI KRISHNA	22383-CS-096
PRAVEEN	22383-CS-097
ARUN	22383-CS-085
AKHILA	22383-CS-076
SAI KEERTHANA	22383-CS-055
G PRIYANKA	22383-CS-064

CONTENTS

TITLES

- ABSTRACT
- INTRODUCTION
- LITERATURE SURVEY
- SYSTEM ANALYSIS
- SYSTEM STUDY
- SYSTEM DESIGN AND DEVELOPMENT
- IMPLEMENTATION
- SYSTEM TESTING
- OUTPUT SCREENS
- REFERENCES
- CONCLUSION

ABSTRACT

Medical document validation is a systematic process aimed at ensuring the accuracy, completeness, and compliance of healthcare documents, including patient records, treatment plans, and billing statements. This validation is crucial for maintaining the integrity of patient information, supporting clinical decision-making, and ensuring adherence to regulatory standards. By rigorously reviewing and verifying medical documents, healthcare organizations can minimize errors, enhance patient care quality, and mitigate legal and financial risks.
Additionally, effective validation contributes to reliable data for research and quality assurance initiatives. As the healthcare landscape increasingly shifts toward digital documentation, the role of medical document validation becomes paramount in safeguarding patient safety and optimizing healthcare delivery.

Electronic Health Records (EHRs) are digital systems that store comprehensive patient health information, replacing traditional paper-based records. EHRs enhance the accessibility and sharing of patient data among healthcare providers, facilitating coordinated and efficient care.
They include critical information such as medical history, diagnoses, medications, treatment plans, immunizations, and test results. The implementation of EHRs promotes improved patient safety through reduced errors, streamlined workflows, and enhanced communication. Additionally, EHRs support data analytics for population health management, quality improvement, and research initiatives. Despite challenges related to interoperability, cost, and user training, EHRs represent a pivotal advancement in modern healthcare, driving better outcomes and patient engagement in the care process.

INTRODUCTION

Medical document validation:

Medical document validation is a critical process in healthcare that ensures the accuracy, completeness, and compliance of medical records and documents. This validation is essential for maintaining the integrity of patient information and supporting clinical decision-making, billing, and regulatory compliance.

The process involves reviewing medical documents—such as patient charts, treatment plans, and billing statements—to confirm that they meet established standards and guidelines. This includes verifying that the information is consistent, correctly coded, and adheres to legal and ethical requirements.

Effective medical document validation helps minimize errors, enhances the quality of patient care, and protects healthcare organizations from legal and financial repercussions. It also plays a vital role in research and quality assurance by ensuring that data used for studies and audits is reliable and valid. As healthcare increasingly relies on electronic systems, the importance of robust validation processes continues to grow, making it a key element in delivering safe and effective patient care.

Electronic Health Records :

Electronic Health Records (EHRs) are digital versions of patients' paper charts and are designed to streamline the management of patient information in healthcare settings. They provide a comprehensive view of a patient's medical history, including diagnoses, medications, treatment plans, immunization dates, allergies, and test results.

EHRs enhance the ability of healthcare providers to make informed decisions and deliver coordinated care by ensuring that accurate and up-to-date information is easily accessible. They also facilitate improved communication among providers, reduce errors, and support better patient engagement through features like patient portals.

Additionally, EHRs often include tools for data analytics, allowing for population health management and research, ultimately contributing to improved healthcare outcomes and efficiency. Their implementation, however, can present challenges, including cost, training, and interoperability between different systems. Overall, EHRs are a crucial component of modern healthcare infrastructure.

LITERATURE SURVEY

1.Introduction

Medical document validation plays a critical role in ensuring the accuracy and reliability of electronic health records (EHRs). EHRs are digital versions of patients' paper charts and are essential for modern healthcare, allowing for the systematic collection of patient information. This literature survey explores the significance of medical document validation, methodologies employed, challenges faced, and the overall impact of EHRs on healthcare delivery.

2.Overview of Electronic Health Records (EHRs)

EHRs are comprehensive digital records that encompass a patient's medical history, medications, lab results, and treatment plans. They enhance the accessibility and sharing of patient data among healthcare providers, contributing to improved care coordination. The implementation of EHRs has been linked to numerous benefits, including increased efficiency in documentation, reduced errors, and better patient outcomes.

3.Medical Document Validation

Medical document validation refers to the processes used to ensure the accuracy, completeness, and reliability of medical documentation. Validating medical documents is essential for maintaining high standards of care and compliance with regulatory requirements.

3.1Types of Validation

- Manual Validation: Involves healthcare professionals reviewing documents to ensure accuracy and completeness. While thorough, manual validation can be time-consuming and subject to human error.

-Automated Validation: Employs technology, such as natural language processing (NLP) and machine learning, to analyze and validate documents. Automated systems can enhance efficiency and accuracy, although they require robust algorithms and training data.

3.2Quality Assurance Techniques

Common quality assurance techniques include:

- Audits and Peer Reviews: Regular audits help identify discrepancies and improve documentation practices.

- Standardized Checklists: Utilizing checklists can guide healthcare providers in maintaining consistent and complete documentation.

4.Challenges in Document Validation

Despite the advantages, several challenges hinder effective document validation:

4.1Human Factors

- Documentation Variability: Different healthcare providers may document information in varying formats and levels of detail, leading to inconsistencies.

- Resistance to Change: Healthcare professionals may resist adopting new technologies or validation practices, impacting the effectiveness of implementation.

4.2Technological Challenges

- Interoperability Issues: Disparate EHR systems may not communicate effectively, resulting in data silos and incomplete patient records.

- Data Integrity: Errors in data entry, such as typos or incorrect information, can compromise the validity of medical documents.

5.Impact of EHRs on Medical Document Validation

EHRs have a profound impact on the validation of medical documents, both positively and negatively.

5.1Positive Impacts

- Improved Accuracy: EHRs facilitate accurate and up-to-date documentation, reducing the likelihood of errors.

- Enhanced Clinical Decision Support: Validated documents contribute to better clinical decision- making, as healthcare providers can access accurate patient information quickly.

5.2Negative Impacts

- Data Overload: The abundance of information in EHRs can overwhelm healthcare providers, potentially leading to misinterpretation or oversight.

- Quality of Data: If EHRs are not properly validated, there may be risks of incomplete or erroneous data, adversely affecting patient care.

6.Case Studies and Examples

Numerous case studies illustrate the importance of effective document validation in EHRs. For example, a hospital that implemented a comprehensive validation protocol reported a significant reduction in documentation errors and improved patient outcomes. Such examples underscore the need for continuous improvement in validation practices.

7.Future Trends and Directions

The future of medical document validation in EHRs is likely to be influenced by several emerging trends:

- Advancements in Technology: Innovations such as artificial intelligence and blockchain may enhance the accuracy and security of medical documents.

- Regulatory Changes: Ongoing developments in healthcare regulations may shape documentation practices and validation requirements.

8.Conclusion

In conclusion, effective medical document validation is essential for maintaining the integrity of electronic health records. While there are significant challenges to overcome, the benefits of accurate documentation are clear. Ongoing research and technological advancements will play a crucial role in enhancing validation practices and ultimately improving patient care.

9.References

Include relevant academic journals, articles, and guidelines here.

SYSTEM ANALYSIS

EXISTING SYSTEM

- In the current healthcare landscape, many medical institutions still rely on fragmented and sometimes manual processes for managing patient records.

- Medical documents are stored in electronic health record (EHR) systems, but these systems often lack seamless integration across different providers, making it difficult to validate documents efficiently.

- The validation process in existing systems typically includes verifying entries manually, which increases the risk of human error, tampering, or fraudulent activity.

- Privacy and release control mechanisms are often inadequate in the existing systems.

- While basic authentication methods are in place, they are frequently insufficient to guarantee patient confidentiality.

- Data breaches remain a major concern, and the release of sensitive information is often poorly controlled, with limited or no ability for patients to regulate who can access their data.

ADVANTAGES AND DISADVANTAGES OF EXISTING SYSTEM

- **Widespread Adoption:**

- Many healthcare providers already use Electronic Health Record (EHR) systems, which offer a centralized location for storing medical documents. These systems are widely accepted in the healthcare industry, making them accessible and familiar.

➢**Basic Functionality:**

- Existing systems provide core functionalities such as storing patient data, accessing medical history, and sharing records within a network of healthcare providers. This allows for improved coordination of patient care compared to paper-based systems.

- **Manual Errors and Data Inconsistencies:**

- Many existing systems rely on manual data entry and validation processes, increasing the risk of human error, inaccurate information, and data inconsistency across different healthcare providers.

PROPOSED SYSTEM

- **Interoperability**: Implement standardized data formats and protocols to ensure seamless data exchange between different healthcare providers and systems.
- **Blockchain Technology**: Use blockchain to enhance data security, privacy, and integrity. Blockchain can also address issues like single points of failure and data breaches.
- **Advanced Encryption**: Employ advanced encryption methods, such as homomorphic encryption and zero-knowledge proof, to protect sensitive patient data.
- **User-Friendly Interface**: Design an intuitive and easy-to- use interface for both healthcare providers and patients, enabling efficient access and man agement of health records.
- **Real-Time Data Access**: Enable real- time access to patient records, allowing healthcare providers to make informed decisions quickly.
- **Predictive Analytics**: Integrate predictive analytics to help identify potential health issue s and provide proactive care

HARDWARE REQUIREMENT

➢**Server Hardware**

- **Purpose:** To host the application, database, and other services.
- **Specifications:**
- **Processor:** Multi-core processors (e.g., Intel Xeon or AMD EPYC) for high performance.
- **RAM:** Minimum 16-32 GB (or more depending on the scale) for efficient data processing.
- **Storage:** SSDs for faster data access (1 TB or more, scalable as needed).
- **Network Interface:** Gigabit Ethernet for high-speed connectivity.

Network Infrastructure

- **Purpose:** To facilitate secure communication between devices and the server. **Components:**
- **Routers:** For directing data traffic between the network and the internet.
- **Switches:** To connect multiple devices within the local network.
- **Firewall:** Hardware firewall for enhanced security against external threats.

➢**Workstations/Client Devices**

- **Purpose:** For healthcare professionals to access the system.
- **Specifications:**
- **Computers:** Desktops or laptops with at least 8 GB of RAM and a modern processor (e.g., Intel i5 or AMD Ryzen 5). • **Monitors:** Dual monitors recommended for improved productivity.

➢**Scanners**

- **Purpose:** To digitize paper-based medical documents for validation.
- **Specifications:**
- **Type:** High-speed document scanners with OCR capabilities.
- **Features:** Duplex scanning and high resolution (at least 300 DPI).

➢**Backup and Storage Solutions**

- **Purpose:** To securely store backups of medical documents and data.
- **Components:**
- **NAS (Network Attached Storage):** For centralized backup and data storage.
- **External Backup Drives:** For off-site backup options

Security Hardware

- **Purpose:** To enhance the security of the infrastructure.
- **Components:**
- **Surge Protectors:** To protect hardware from power surges.
- **Uninterruptible Power Supply (UPS):** To provide backup power during outages.

SOFTWARE REQUIREMENT

➢**Operating System**

- **Server OS:** Windows Server or Linux (e.g., Ubuntu Server, CentOS)
- **Client OS:** Windows, macOS, or Linux for user workstations

➢**Database Management System (DBMS)**

- **Options:**
- SQL Database: PostgreSQL or MySQL

- NoSQL Database: MongoDB (for unstructured data)

➢**Web Server**

- **Options:** Apache or Nginx for hosting the application

➢**Application Framework**

- **Frontend:** React, Angular, or Vue.js for user interface development
- **Backend:** Node.js, Python Flask, or Django for server-side logic

➢**Document Management System (DMS)**

- Software for organizing, storing, and retrieving medical documents
- Examples: M-Files, DocuWare (if integrating third-party solutions)

Scanning and OCR Software

- **Examples:** ABBYY FineReader or Tesseract for optical character recognition

2. SYSTEM STUDY

2.1FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

ECONOMICAL FEASIBILITY

TECHNICAL FEASIBILITY

SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

PRELIMINARY INVESTIGATION

The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages, there is a search engine ,address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, ie. preliminary investigation begins. The activity has three parts:

- **Request Clarification**
- **Feasibility Study**
- **Request Approval**
-

REQUEST CLARIFICATION

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely
what the system requires.

Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network(LAN). In today's busy schedule man need everything should be provided in a readymade manner. So taking into consideration of the vastly use of the net in day to day life, the corresponding development of the portal came into existence.

FEASIBILITY ANALYSIS

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

- **Operational Feasibility**
- **Economic Feasibility**
- **Technical Feasibility**

Operational Feasibility

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Economic Feasibility

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

Technical Feasibility

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and WebLogic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

4.3.3 REQUEST APPROVAL

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

SYSTEM DESIGN AND DEVELOPMENT

INPUT DESIGN

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

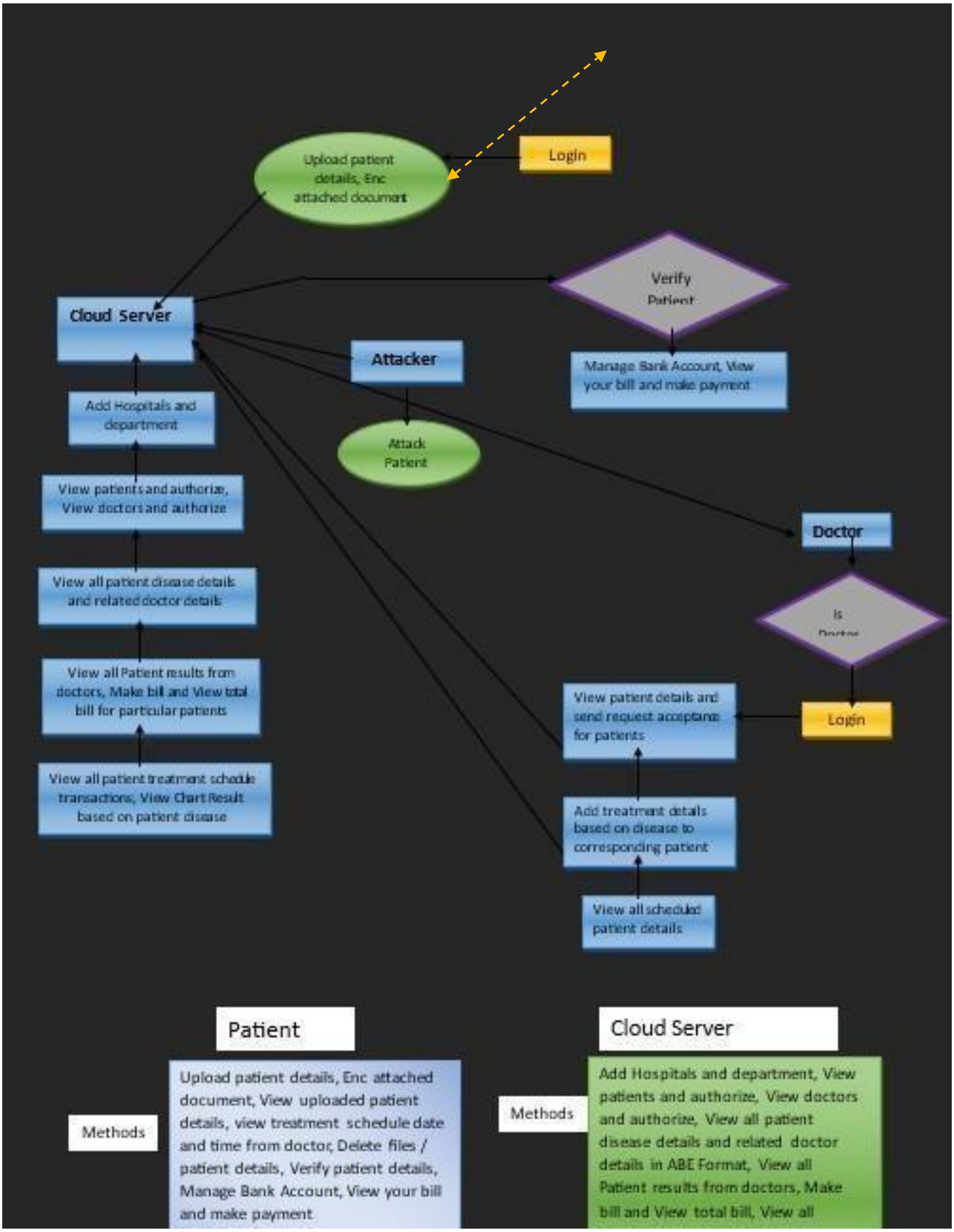
Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

OUTPUT DESIGN

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.



Upload patient details, Enc attached document
Login
Verify Patient
Cloud Server
Attacker
Manage Bank Account, View your bill and make payment
Add Hospitals and department
Attack Patient
View patients and authorize, View doctors and authorize
Doctor
View all patient disease details and related doctor details
Is Doctor
View all Patient results from doctors, Make bill and View total bill for particular patients
View patient details and send request acceptance for patients
Login
View all patient treatment schedule transactions, View Chart Result based on patient disease
Add treatment details based on disease to corresponding patient
View all scheduled patient details

Patient

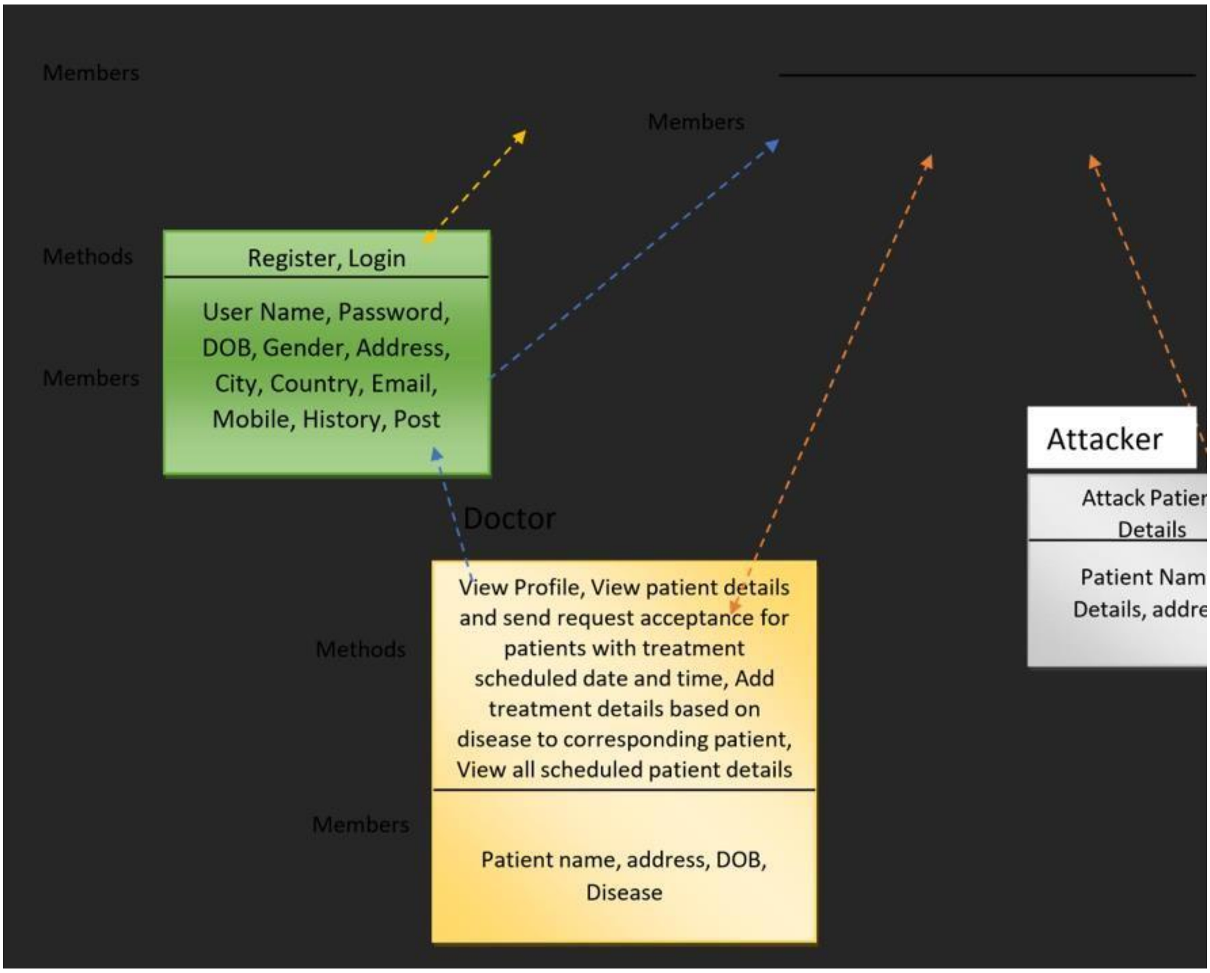
Methods

Upload patient details, Enc attached document, View uploaded patient details, view treatment schedule date and time from doctor, Delete files / patient details, Verify patient details, Manage Bank Account, View your bill and make payment

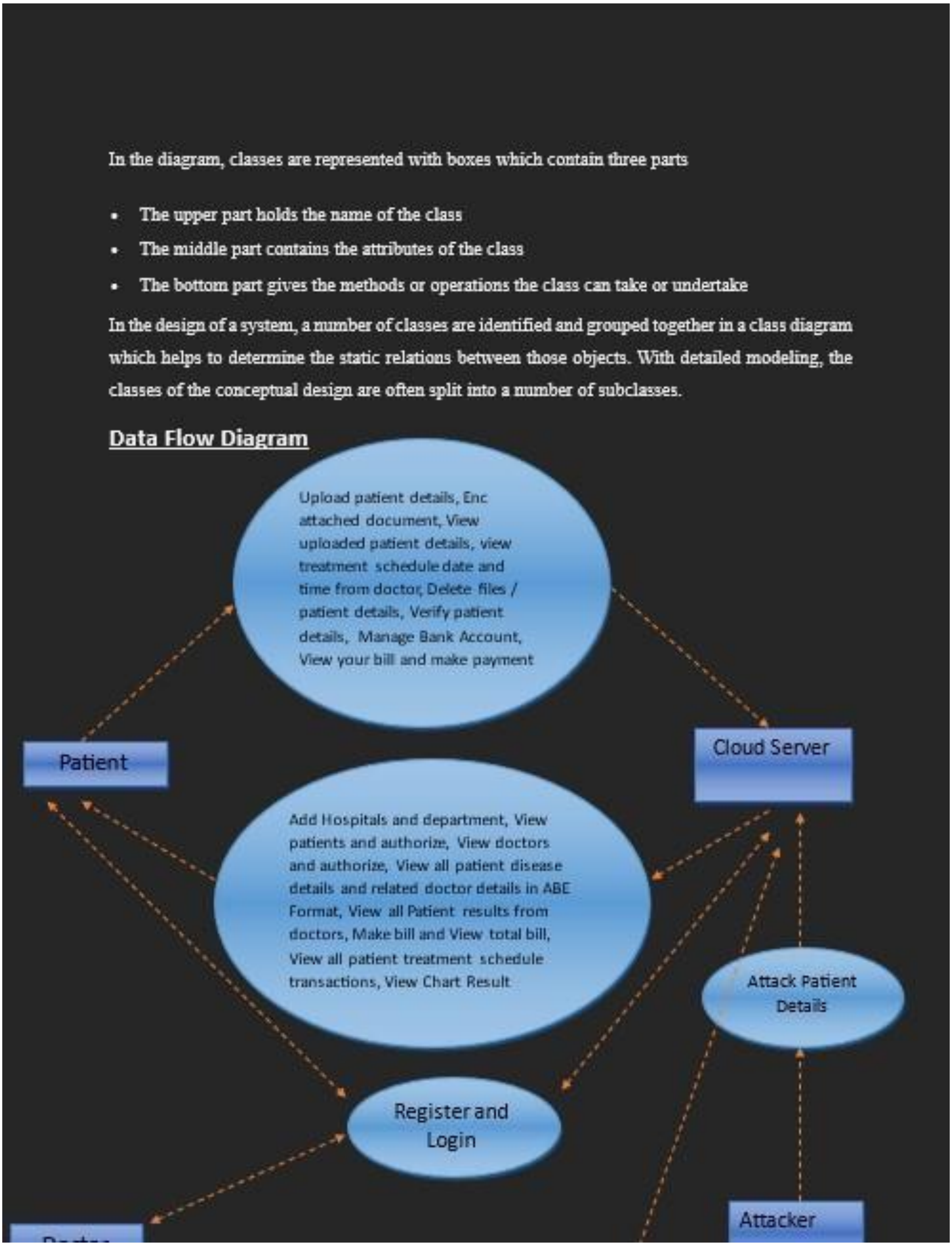
Cloud Server

Methods

Add Hospitals and department, View patients and authorize, View doctors and authorize, View all patient disease details and related doctor details in ABE Format, View all Patient results from doctors, Make bill and View total bill, View all


Members
Members
Methods
Register, Login
Members
User Name, Password, DOB, Gender, Address, City, Country, Email, Mobile, History, Post
Attacker
Attack Patier Details
Patient Nam Details, addre
Doctor
Methods
View Profile, View patient details and send request acceptance for patients with treatment scheduled date and time, Add treatment details based on disease to corresponding patient, View all scheduled patient details
Members
Patient name, address, DOB, Disease

The class diagram is the main building block of object orientedmodeling. It is used both for general conceptual modelingof the systematic of the application, and for detailed modeling translating the models intoprogrammingcode. Class diagrams can also be used for modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed.



In the diagram, classes are represented with boxes which contain three parts

- The upper part holds the name of the class
- The middle part contains the attributes of the class
- The bottom part gives the methods or operations the class can take or undertake

In the design of a system, a number of classes are identified and grouped together in a class diagram which helps to determine the static relations between those objects. With detailed modeling, the classes of the conceptual design are often split into a number of subclasses.

Data Flow Diagram

Upload patient details, Enc attached document, View uploaded patient details, view treatment schedule date and time from doctor, Delete files / patient details, Verify patient details, Manage Bank Account, View your bill and make payment
Patient
Cloud Server
Add Hospitals and department, View patients and authorize, View doctors and authorize, View all patient disease details and related doctor details in ABE Format, View all Patient results from doctors, Make bill and View total bill, View all patient treatment schedule transactions, View Chart Result
Attack Patient Details
Register and Login
Attacker
Doctor

IMPLEMENTATION

Patient:

A patient outsources her documents to the cloud server to provide convenient and reliable data access to the corresponding search doctors. To protect the data privacy, the patient encrypts the original documents under an access policy using attribute- based encryption. To improve the search efficiency, she also generates some keyword for each outsourced document. The corresponding index is then generated according to the keywords using the secret key of the secure kNN scheme. After that, the patient sends the encrypted documents, and the corresponding indexes to the cloud server, and submits the secret key to the search doctors.

Cloud server:

A cloud server is an intermediary entity which stores the encrypted documents and the corresponding indexes received from patients, and then provides data access and search services to authorized search doctors. When a search doctor sends a trapdoor to the cloud server, it would return a collection of matching documents based on certain operations.

Doctor:

An authorized doctor can obtain the secret key from the patient, where this key can be used to generate trapdoors. When she needs to search the outsourced documents stored in the cloud server, she will generate a search keyword set. Then according to the keyword set, the doctor uses the secret key to generate a trapdoor and sends it to the cloud server. Finally, she receives the matching document collection from the cloud server and decrypts them with the ABE key received from the trusted authority. After getting the health information of the patient, the doctor can also outsource medical report to the cloud server by the same way. For simplicity, we just consider one-way communication in our schemes.

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.
Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as

specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot
“see” into it. The test provides inputs and responds to outputs without considering how the software works.

6.1Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

6.2Integration_Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

6.3Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

SYSTEM TESTING TESTING METHODOLOGIES

The following are the Testing Methodologies:

- **Unit Testing.**
- **Integration Testing.**
- **User Acceptance Testing.**
- **Output Testing.**
- **Validation Testing.**

Unit Testing

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module's control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing.

During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

Integration Testing

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

The following are the types of Integration Testing:

1. Top Down Integration

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

2. Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- ✦ The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- ✦ A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- ✦ The cluster is tested.
- ✦ Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

7.1.3User Acceptance Testing

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

7.1.4Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways
– one is on screen and another in printed format.

7.1.5Validation Checking

Validation checks are performed on the following fields.

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables.
Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

Using Artificial Test Data:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package "Virtual Private Network" has satisfied all the requirements specified as per software requirement specification and was accepted.

7.2USER TRAINING

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

7.3MAINTAINENCE

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future.

The coding and designing is simple and easy to understand which will make maintenance easier.

TESTING STRATEGY :

A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

SYSTEM TESTING:

Software once validated must be combined with other system elements (e.g. Hardware, people, database). System testing verifies that all the elements are proper and that overall system function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

UNIT TESTING:

In unit testing different are modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. Using the detailed design description as a guide, important Conrail paths are tested to uncover errors within the boundary of the modules. This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

In Due Course, latest technology advancements will be taken into consideration. As part of technical build-up many components of the networking system will be generic in nature so that future projects can either use or interact with this. The future holds a lot to offer to the development and refinement of this project.

OTHER TESTING METHODOLOGIES

User Acceptance Testing

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system
users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

Validation Checking

Validation checks are performed on the following fields.

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables.
Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data:

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

Using Artificial Test Data:

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.
The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package "Virtual Private Network" has satisfied all the requirements specified as per software requirement specification and was accepted.

USER TRAINING

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

MAINTAINENCE

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which will make maintenance easier.

TESTING STRATEGY :

A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

SYSTEM TESTING:

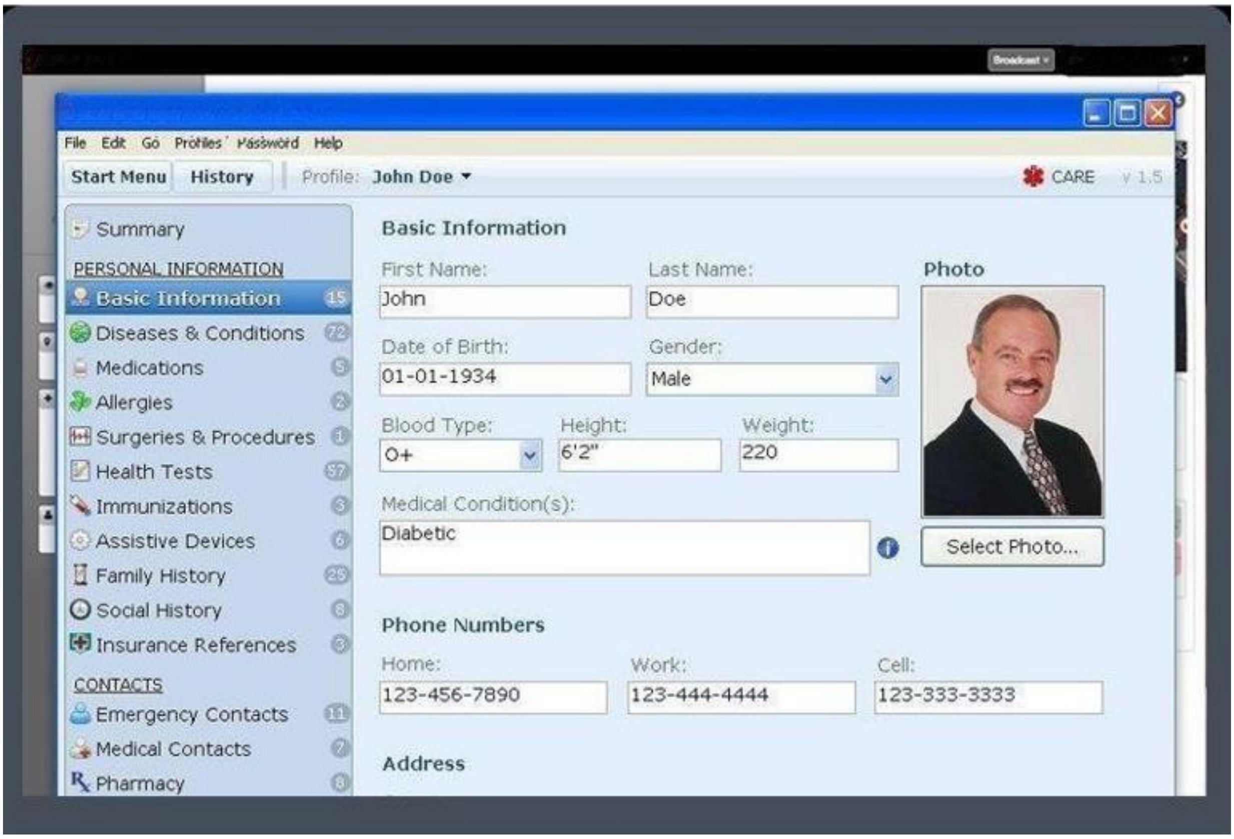
Software once validated must be combined with other system elements (e.g. Hardware, people, database). System testing verifies that all the elements are proper and that overall system function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

UNIT TESTING:

In unit testing different are modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. Using the detailed design description as a guide, important Conrail paths are tested to uncover errors within the boundary of the modules. This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

In Due Course, latest technology advancements will be taken into consideration. As part of technical build-up many components of the networking system will be generic in nature so that future projects can either use or interact with this. The future holds a lot to offer to the development and refinement of this project.

OUTPUT SCREENS



Basic Information

First Name: John

Last Name: Doe

Date of Birth: 01-01-1934

Gender: Male

Blood Type: O+

Height: 6'2"

Weight: 220

Medical Condition(s): Diabetic

Phone Numbers

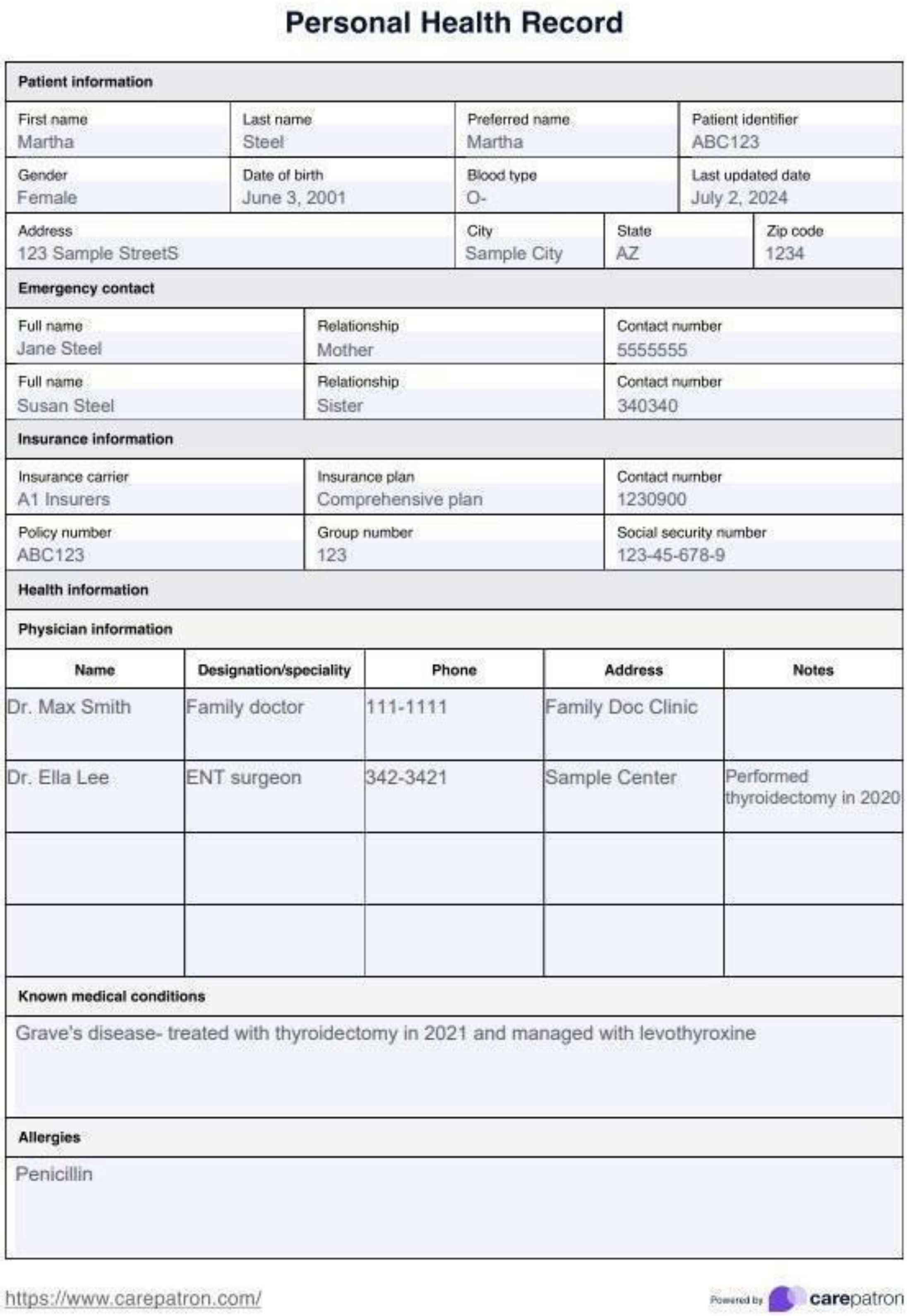
Home: 123-456-7890

Work: 123-444-4444

Cell: 123-333-3333

Address

EENS



Personal Health Record

Patient information

First name Martha	Last name Steel	Preferred name Martha	Patient identifier ABC123
Gender Female	Date of birth June 3, 2001	Blood type O-	Last updated date July 2, 2024
Address 123 Sample StreetS	City Sample City	State AZ	Zip code 1234

Emergency contact

Full name Jane Steel	Relationship Mother	Contact number 5555555
Full name Susan Steel	Relationship Sister	Contact number 340340

Insurance information

Insurance carrier A1 Insurers	Insurance plan Comprehensive plan	Contact number 1230900
Policy number ABC123	Group number 123	Social security number 123-45-678-9

Health information

Physician information

Name	Designation/speciality	Phone	Address	Notes
Dr. Max Smith	Family doctor	111-1111	Family Doc Clinic	
Dr. Ella Lee	ENT surgeon	342-3421	Sample Center	Performed thyroidectomy in 2020

Known medical conditions

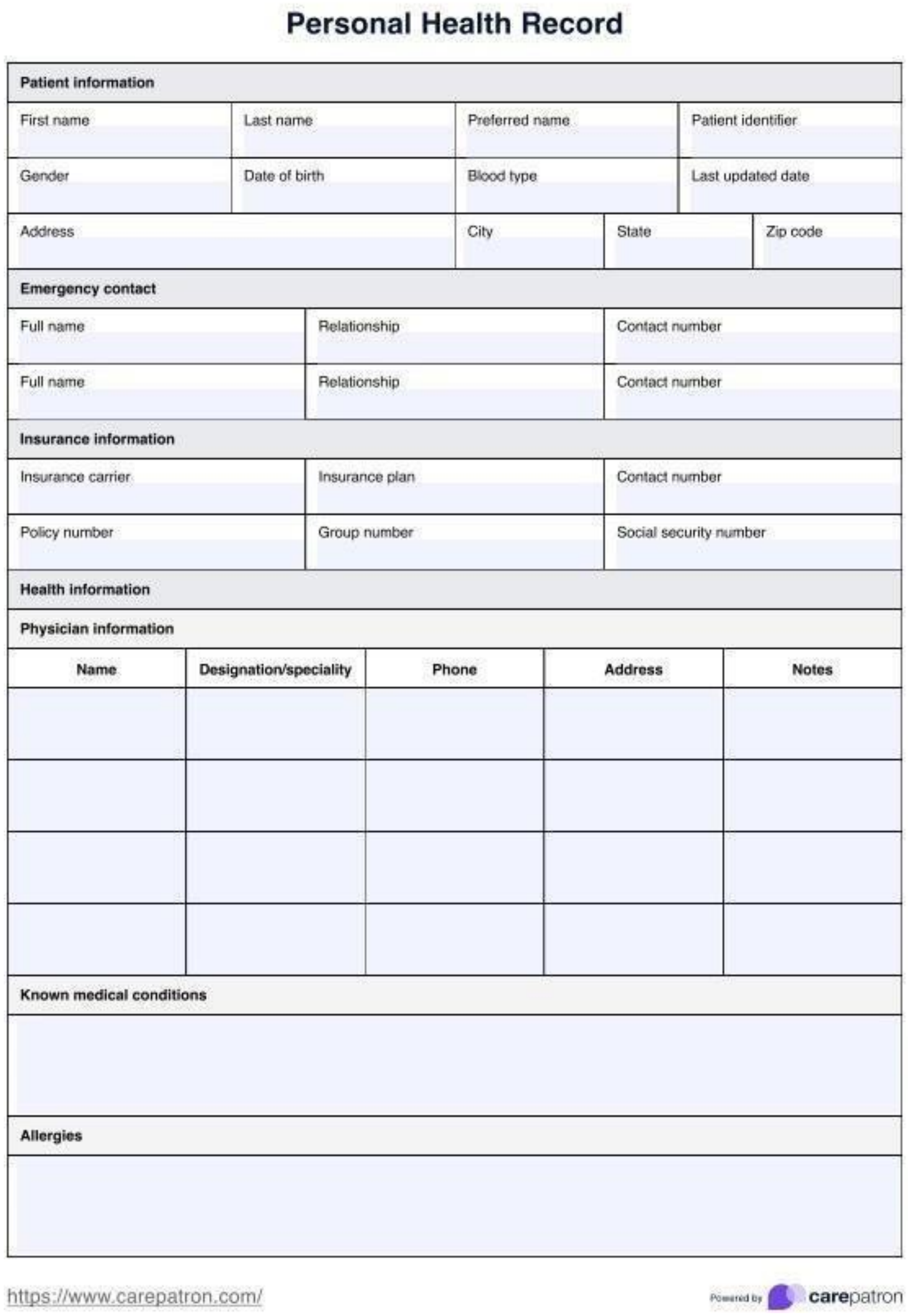
Grave's disease- treated with thyroidectomy in 2021 and managed with levothyroxine

Allergies

Penicillin

https://www.carepatron.com/

Powered by carepatron



Personal Health Record

Patient information

First name	Last name	Preferred name	Patient identifier
Gender	Date of birth	Blood type	Last updated date

Address	City	State	Zip code

Emergency contact

Full name	Relationship	Contact number
Full name	Relationship	Contact number

Insurance information

Insurance carrier	Insurance plan	Contact number
Policy number	Group number	Social security number

Health information

Physician information

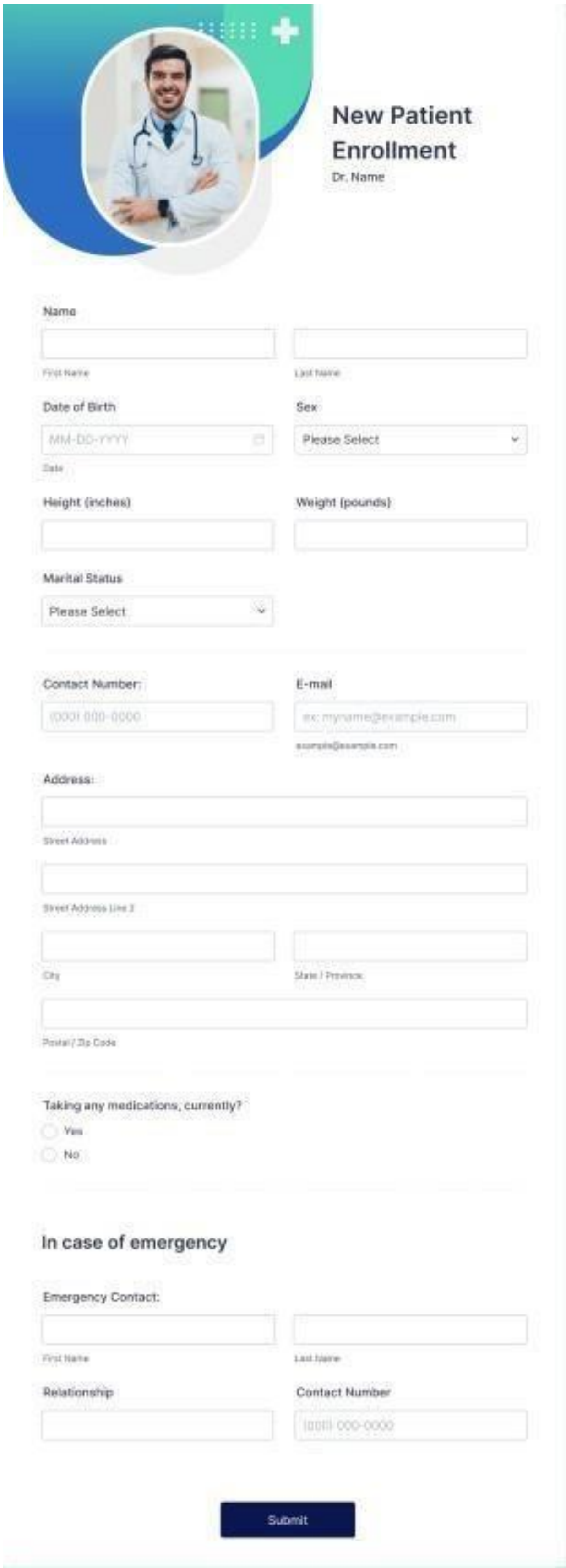
Name	Designation/speciality	Phone	Address	Notes

Known medical conditions

Allergies

https://www.carepatron.com/

Powered by carepatron



New Patient Enrollment

Dr. Name

Name

First Name

Last Name

Date of Birth

MM-DD-YYYY

Date

Sex

Please Select

Height (inches)

Weight (pounds)

Marital Status

Please Select

Contact Number:

(000) 000-0000

E-mail

ex: myname@example.com

example@example.com

Address:

Street Address

Street Address Line 2

City

State / Province

Postal / Zip Code

Taking any medications, currently?

○ Yes

○ No

In case of emergency

Emergency Contact:

First Name

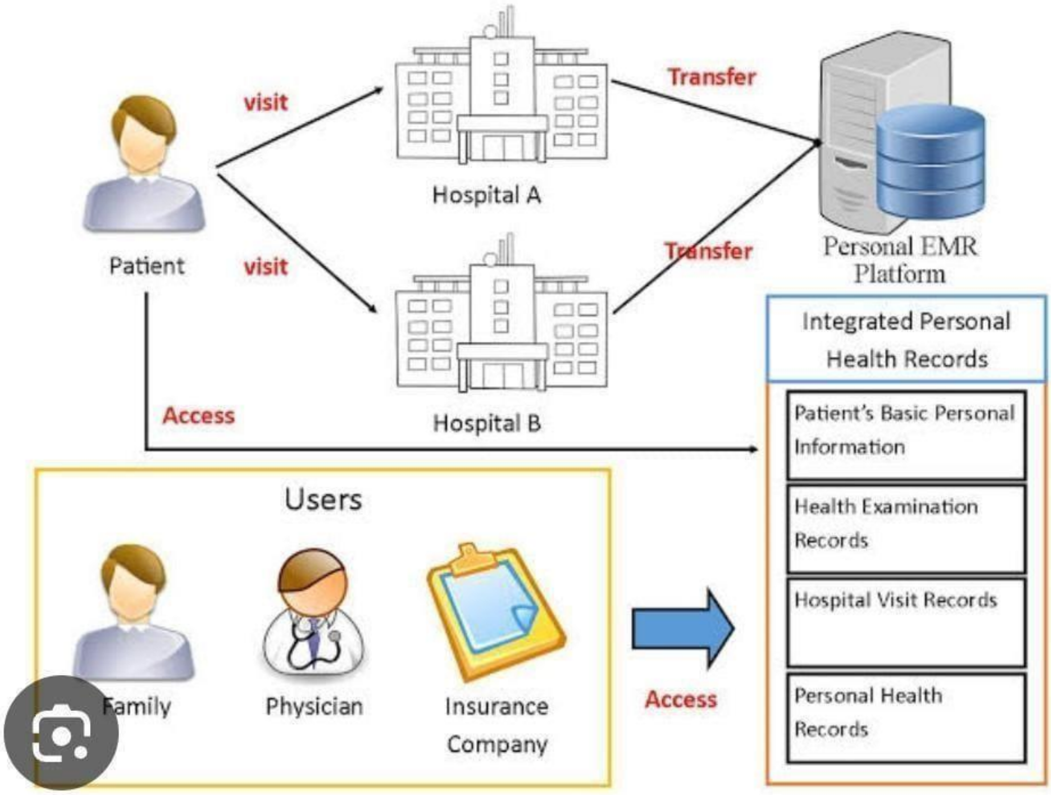
Last Name

Relationship

Contact Number

(000) 000-0000

Submit



ELECTRONIC HEALTH RECORDS (EHR)

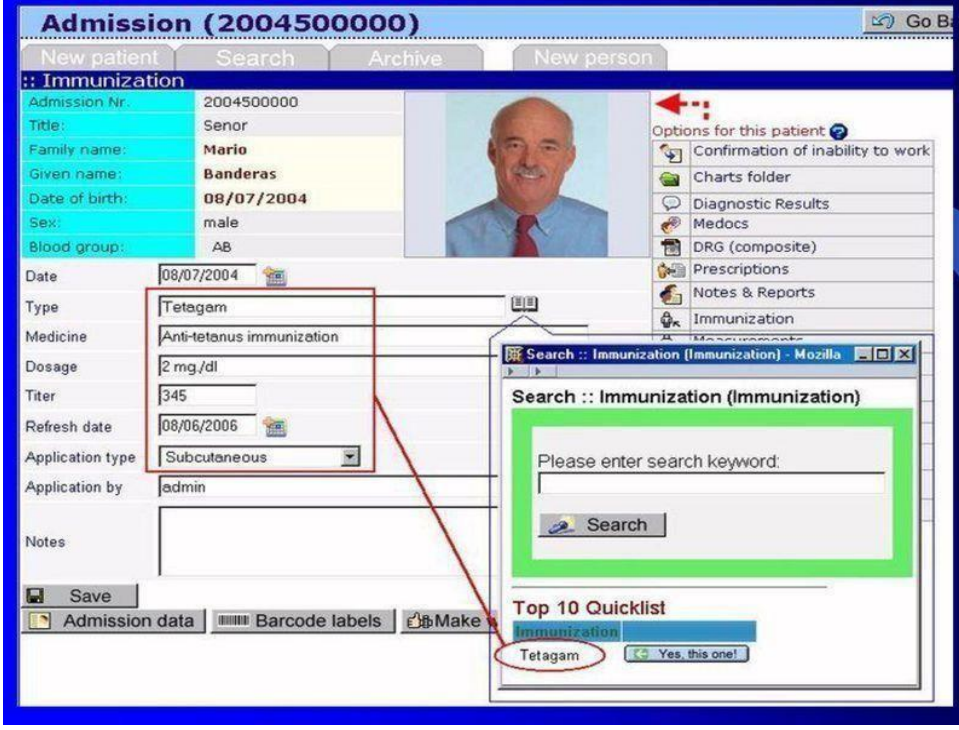
What is Electronic Health Record?

ELECTRONIC HEALTH RECORDS

ELECTRONIC HEALTH RECORDS (EHR)

An electronic health record is a collection of secure, real-time electronically-stored health information records for a patient that make information available instantly and securely to authorized users. It is a digital version of a patient's paper chart and provides a comprehensive digital view of a patient's health history.

3.3. EHR example



Admission (2004500000)

:: Immunization

Admission Nr.	2004500000
Title:	Senor
Family name:	Mario
Given name:	Banderas
Date of birth:	08/07/2004
Sex:	male
Blood group:	AB

Date	08/07/2004
Type	Tetagam
Medicine	Anti-tetanus immunization
Dosage	2 mg./dl
Titer	345
Refresh date	08/06/2006
Application type	Subcutaneous
Application by	admin
Notes	

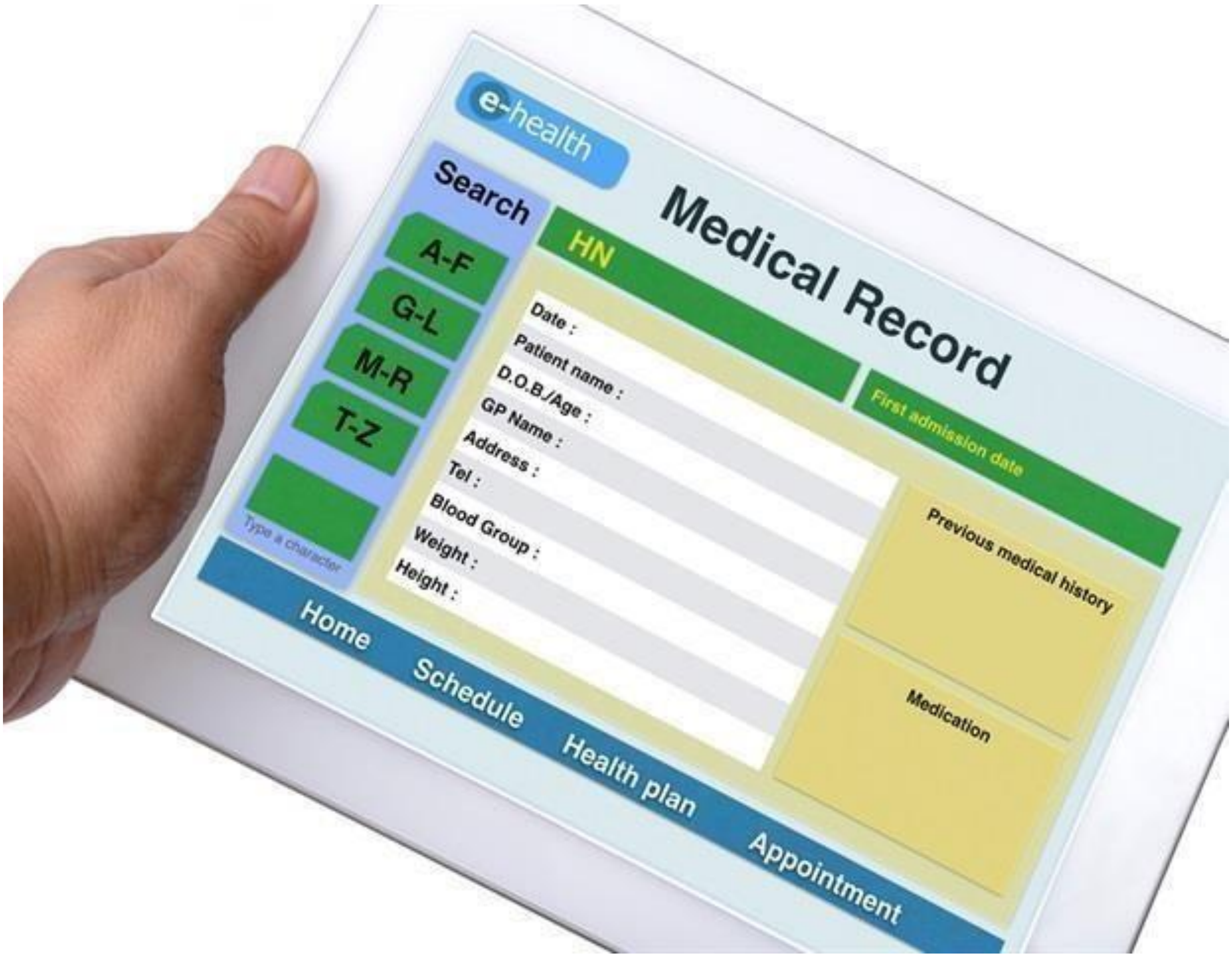
Search :: Immunization (Immunization)

Please enter search keyword:

Top 10 Quicklist

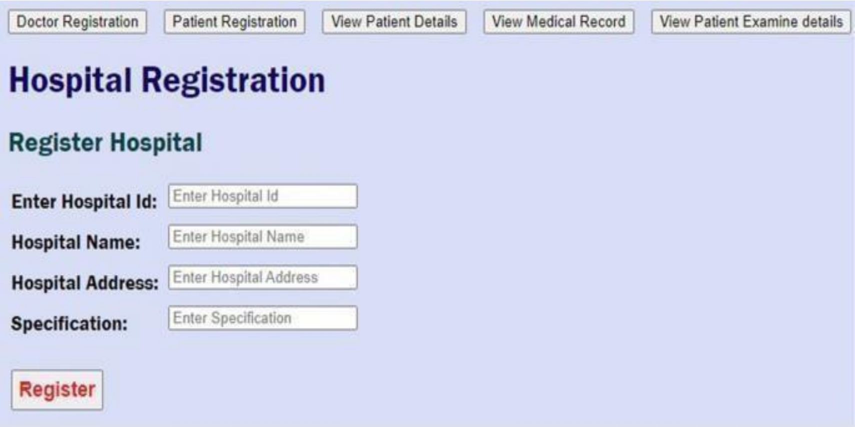
Immunization	
Tetagam	Yes, this one!

Electronic health record (EHR) with image and document links


e-health
Medical Record
Search
A-F
G-L
M-R
T-Z
Type a character
HN
First admission date
Date :
Patient name :
D.O.B./Age :
GP Name :
Address :
Tel :
Blood Group :
Weight :
Height :
Previous medical history
Medication
Home
Schedule
Health plan
Appointment

OLD

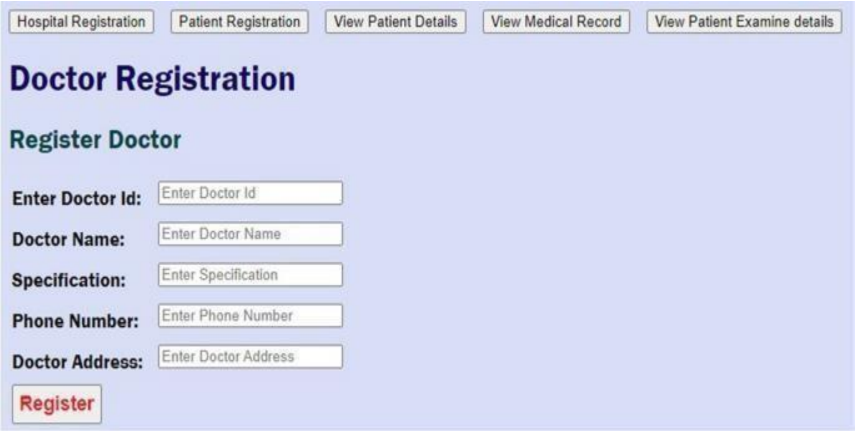
NEW



Hospital Registration

Register Hospital

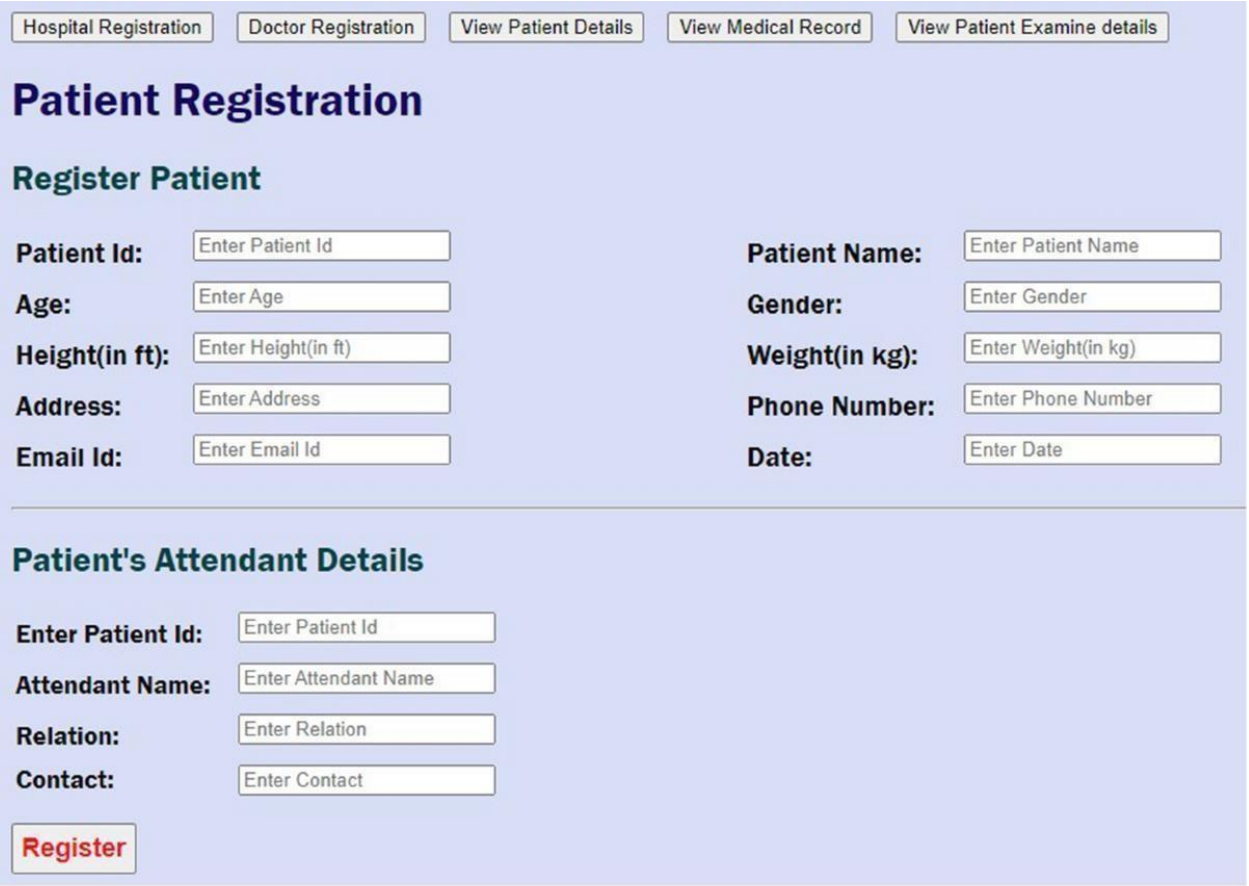
Enter Hospital Id:	Enter Hospital Id
Hospital Name:	Enter Hospital Name
Hospital Address:	Enter Hospital Address
Specification:	Enter Specification



Doctor Registration

Register Doctor

Enter Doctor Id:	Enter Doctor Id
Doctor Name:	Enter Doctor Name
Specification:	Enter Specification
Phone Number:	Enter Phone Number
Doctor Address:	Enter Doctor Address



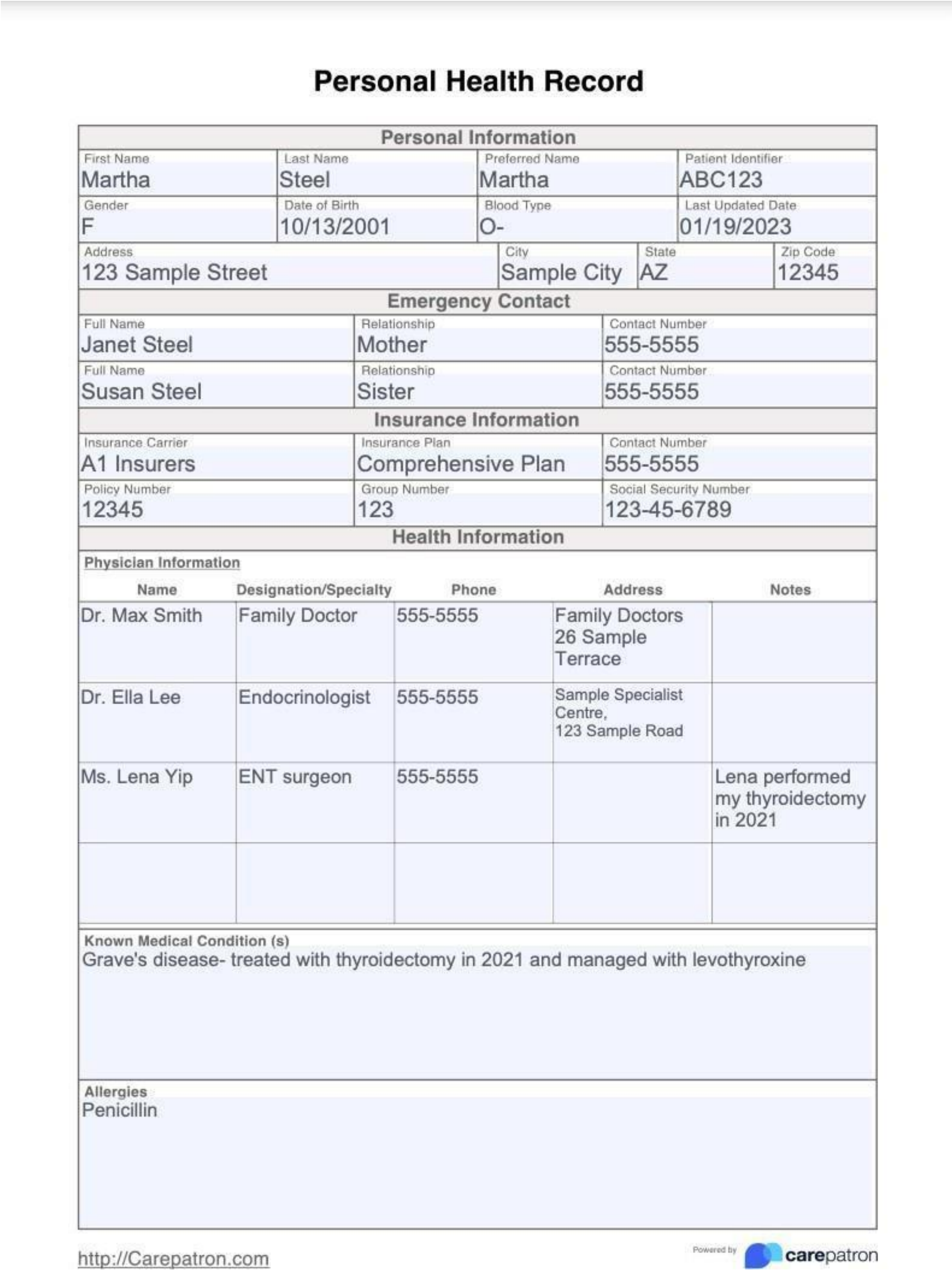
Patient Registration

Register Patient

Patient Id:	Enter Patient Id	Patient Name:	Enter Patient Name
Age:	Enter Age	Gender:	Enter Gender
Height(in ft):	Enter Height(in ft)	Weight(in kg):	Enter Weight(in kg)
Address:	Enter Address	Phone Number:	Enter Phone Number
Email Id:	Enter Email Id	Date:	Enter Date

Patient's Attendant Details

Enter Patient Id:	Enter Patient Id
Attendant Name:	Enter Attendant Name
Relation:	Enter Relation
Contact:	Enter Contact



Personal Health Record

Personal Information

First Name	Last Name	Preferred Name	Patient Identifier
Martha	Steel	Martha	ABC123
Gender	Date of Birth	Blood Type	Last Updated Date
F	10/13/2001	O-	01/19/2023

Address	City	State	Zip Code
123 Sample Street	Sample City	AZ	12345

Emergency Contact

Full Name	Relationship	Contact Number
Janet Steel	Mother	555-5555
Full Name	Relationship	Contact Number
Susan Steel	Sister	555-5555

Insurance Information

Insurance Carrier	Insurance Plan	Contact Number
A1 Insurers	Comprehensive Plan	555-5555
Policy Number	Group Number	Social Security Number
12345	123	123-45-6789

Health Information

Physician Information

Name	Designation/Specialty	Phone	Address	Notes
Dr. Max Smith	Family Doctor	555-5555	Family Doctors 26 Sample Terrace	
Dr. Ella Lee	Endocrinologist	555-5555	Sample Specialist Centre, 123 Sample Road	
Ms. Lena Yip	ENT surgeon	555-5555		Lena performed my thyroidectomy in 2021

Known Medical Condition (s)

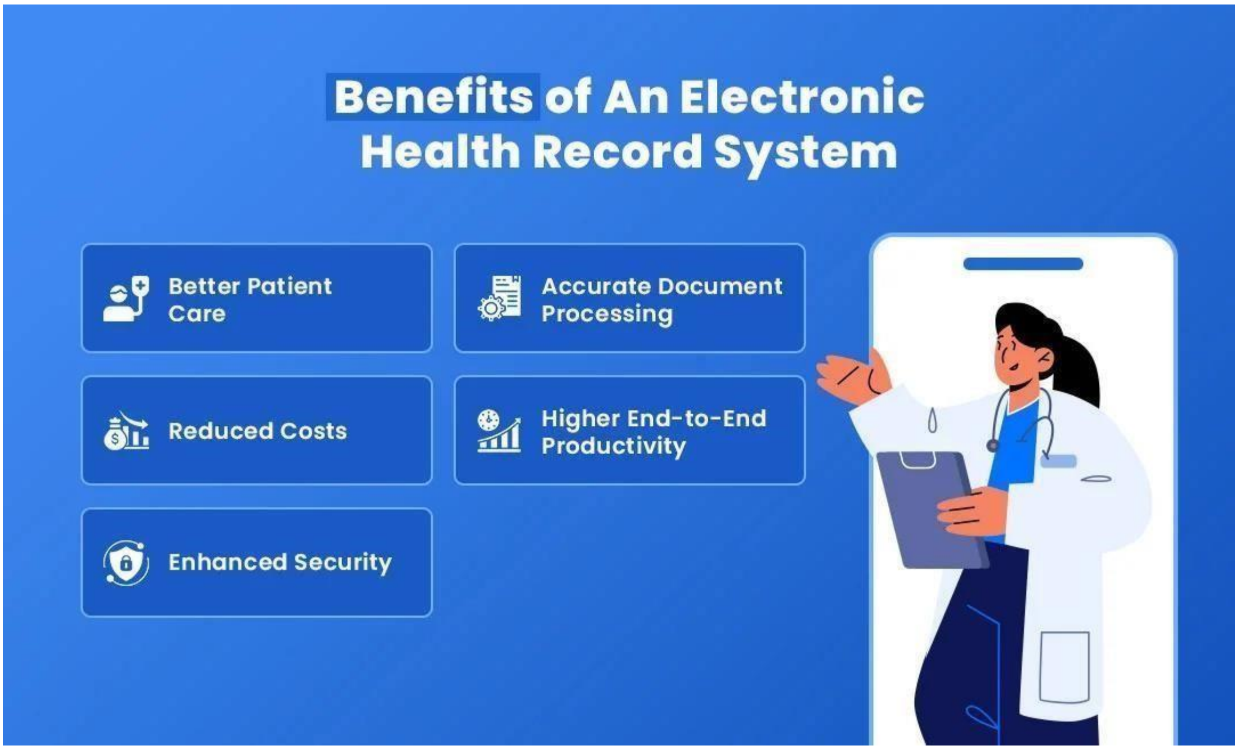
Grave's disease- treated with thyroidectomy in 2021 and managed with levothyroxine

Allergies

Penicillin

http://Carepatron.com

Powered by carepatron



Benefits of An Electronic Health Record System

- Better Patient Care
- Accurate Document Processing
- Reduced Costs
- Higher End-to-End Productivity
- Enhanced Security

REFERENCES

[1] X. Chen, J. Li, J. Weng, J. Ma, and W. Lou, “Verifiable computation over large database with incremental updates,” IEEE transactions on Computers, vol. 65, no. 10, pp. 3184–3195, 2016.

[2] X. Chen, J. Li, X. Huang, J. Ma, and W. Lou, “New publicly verifiable databases with efficient updates,” IEEE Transactions on Dependable and Secure Computing, vol. 12, no. 5, pp. 546–556, 2015.

[3] X. Chen, X. Huang, J. Li, J. Ma, W. Lou, and D. S. Wong, “New algorithms for secure outsourcing of large-scale systems of linear equations,” IEEE transactions on information forensics and security, vol. 10, no. 1, pp. 69–78, 2015.

[4] X. Chen, J. Li, J. Ma, Q. Tang, and W. Lou, “New algorithms for secure outsourcing of modular exponentiations,” IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 9, pp. 2386–2396, 2014.

[5] J. Wang, X. Chen, X. Huang, I. You, and Y. Xiang, “Verifiable auditing for outsourced database in cloud computing,” IEEE transactions on computers, no. 1, pp. 1–1, 2015.

[6] T. Jiang, X. Chen, and J. Ma, “Public integrity auditing for shared dynamic cloud data with group user revocation,” IEEE Transactions on Computers, vol. 65, no. 8, pp. 2363–2373, 2016.

[7] X. Zhang, T. Jiang, K.-C. Li, A. Castiglione, and X. Chen, “New publicly verifiable computation for batch matrix multiplication,” Information Sciences, 2017.

CONCLUSION

The electronic health system represents a transformative shift in healthcare delivery, facilitating improved access to patient information and enhancing the quality of care. The integration of electronic health records (EHRs) streamlines data management, enabling healthcare providers to make informed decisions based on comprehensive patient histories. However, as the reliance on digital systems grows, so does the need for robust security and privacy measures to protect sensitive medical information.Effective validation of medical documents within these systems is essential to maintain data integrity and uphold patient confidentiality. By addressing the challenges of interoperability, compliance with regulations, and user training, the electronic health system can realize its full potential. Future advancements, such as incorporating artificial intelligence and blockchain technology, may further enhance the reliability and security of EHRs.Ultimately, the success of electronic health systems hinges on a commitment to safeguarding patient privacy while fostering collaboration among stakeholders, paving the way for a more efficient and effective healthcare landscape.