**Real Time Mapping Of Epidemic Spread**

|  |
| --- |
| **CONTENT** |
| **Abstract** |
| **1.INTRODUCTION** |
| 1.1 Motivation |
| 1.2 Problem Statement |
| 1.3 Objective of the Project |
| 1.4 Scope |
| 1.5 Project Introduction |
| **2.LITERATURE SURVEY** |
| 2.1 Related Work |
| **3. SYSTEM ANALYSIS** |
| 3.1 Existing System |
| 3.2 Disadvantages |
| 3.3 Proposed System |
| 3.4 Advantages |
| 3.5 work Flow of Proposed system |
| **4. REQUIREMENT ANALYSIS** |
| 4.1Function and non-functional requirements |
| 4.2 Hardware Requirements |
| 4.3 Software Requirements |
| 4.4 Architecture |
| **5. SYSTEM DESIGN** |
| 5.1 Introduction of Input design |
| 5.2 UML Diagram(class, use case, sequence, collaborative, deployment, activity, ER diagram and Component diagram) |
| 5.3 Data Flow Diagram |
| **6. IMPLEMENTATION AND RESULTS** |
| 6.1 Modules |
| 6.2 Output Screens |
| 1. **PYTHON ENVIRONMENT** |
| 7.1 Software Installation |
| 7.2 About Python Language |
| 1. **SYSTEM STUDY AND TESTING** |
| 8.1 Feasibility study |
| 8.2 Types of test & Test Cases |
| **9. CONCLUSION** |
| **10. FUTURE ENHANCEMENT** |
| **11. REFERENCES** |

**ABSTRACT:**

This integrated healthcare platform offers a robust system for doctors and patients. Doctors can efficiently manage appointments, accepting or rejecting them, and input epidemic-related data. Patients can register, book appointments, track their status, and update their recovery status. The dashboard provides a comprehensive overview of epidemic disease details, allowing day and month-wise filtering. Users can download patient details and access graphical representations of epidemic cases and recovery rates for informed decision-making. This streamlined approach combines mathematical modeling and machine learning with practical healthcare management, fostering a proactive and data-driven response to epidemics.

**Keywords:** Healthcare Modules, Admin, Doctor epidemic disease.

1. **INTRODUCTION**
   1. **MOTIVATION:**

This integrated healthcare platform addresses critical gaps in epidemic management by seamlessly merging mathematical modeling, machine learning, and practical healthcare functionalities. Offering doctors streamlined appointment management and real-time epidemic data input, it empowers patients to actively engage in their healthcare journey. The comprehensive dashboard ensures informed decision-making with day and month-wise epidemic insights. Despite potential challenges, the platform's data-driven approach optimizes pandemic response, fostering a resilient and adaptive healthcare system for enhanced epidemic control and patient care.

* 1. **PROBLEM STATEMENT:**

The current healthcare decision-making landscape faces challenges in effectively responding to epidemics, such as COVID-19. Existing studies lack generalization or surveillance data, leading to suboptimal policy decisions. City administrators rely on open-loop, belief-based decision-making, hindering timely policy enforcement. To address these issues, there's a need for an integrated healthcare platform. The proposed platform aims to merge mathematical modeling and machine learning for accurate epidemic predictions, providing doctors with tools for efficient appointment management and real-time epidemic data input. Patients can seamlessly navigate the system, booking appointments and monitoring their recovery status. The overarching problem is the lack of a comprehensive, data-driven approach to epidemic control and patient care.

* 1. **OBJECTIVE:**

The objective is to create an integrated healthcare platform that optimizes pandemic response. Incorporating mathematical modeling and machine learning, the system empowers doctors to efficiently manage appointments and input epidemic-related data. Patients can easily register, book appointments, track statuses, and update their recovery status. The dashboard offers a comprehensive view of epidemic details with day and month-wise filtering, accompanied by downloadable patient information and graphical representations for informed decision-making. This platform aims to streamline healthcare management, fostering a proactive, data-driven approach to epidemic control and patient care.

* 1. **SCOPE:**

The scope of this integrated healthcare platform encompasses the efficient management of epidemic scenarios, merging mathematical modeling, machine learning, and practical healthcare functionalities. The platform facilitates doctors in optimizing appointment schedules and entering epidemic-related data, ensuring timely and data-driven decision-making. Patients can seamlessly register, book appointments, and monitor their recovery progress. The comprehensive dashboard extends the scope to include detailed day and month-wise epidemic insights, downloadable patient data, and graphical representations for informed analysis. This system's overarching scope is to enhance pandemic response, offering a holistic approach that amalgamates technological advancements with real-world healthcare management.

* 1. **PROJECT INTRODUCTION:**

On March 11, 2020, the World Health Organization declared COVID-19 a global pandemic caused by the SARS-CoV-2 coronavirus. By this date, the UK had reported 373 confirmed cases and six deaths. As of July 15, 2020, these figures rose to 291,911 cases and 45,053 deaths. Lockdowns were implemented across the country on March 23, 2020, as a crucial measure in the government's response to COVID-19.

Detecting local variations in infection rates became essential for controlling the virus, leading to the implementation of measures like the first 'local lockdown' in Leicester on June 30, 2020. This was in response to a significant cluster of COVID-19 cases, constituting about one-tenth of all new cases in the country that week.

The COVID Symptom Study app, released on March 24, 2020, collects daily updates on self-reported COVID-19 symptoms and users' residential postcodes. The Secure Anonymised Information Linkage (SAIL) Databank securely stores and uses anonymized person-based data for health research. During the pandemic, SAIL received daily updates from the COVID Symptom Study app, enabling near real-time health surveillance across the UK.

To understand the localized spread of the disease, existing statistical methodology was adapted for the analysis of geo-referenced health outcome data. This allowed mapping, at a fine resolution (Lower-layer Super Output Area or equivalent), the prevalence of positive symptom reports among app users over a rolling 14-day period. These maps, despite the limitations of self-reported data, offered the first fine-scale, UK-wide assessment of the geographical distribution of probable COVID-19 infections. They have been utilized by devolved administrations for pandemic planning.

1. **LITERATURE REVIEW**
   1. **Related works:**

**[1]. P. Tolbert, J. Mulholland, D. MacIntosh, F. Xu, D. Daniels, O. Devine, B.P. Carlin, M. Klein, J. Dorley, A. Butler, D. Nordenberg, H. Frumkin, P.B. Ryan, M. White, Air pollution and pediatric emergency room visits for asthma in Atlanta, American Journal of Epidemiology 151 (8) (2000) 798–810.**

The study, conducted by P. Tolbert and colleagues, investigates the relationship between air pollution and pediatric emergency room visits for asthma in Atlanta. Published in the American Journal of Epidemiology in 2000, the research aims to understand the potential impact of air quality on asthma-related health outcomes among children. Asthma is a significant public health concern, and exploring its connection to air pollution is crucial for developing effective preventive strategies.

Prior to this study, there was likely existing knowledge suggesting a correlation between air pollution and respiratory issues, particularly asthma. The existing literature might have highlighted the general association between poor air quality and adverse respiratory health outcomes. However, the specifics of this relationship, especially concerning pediatric emergency room visits for asthma in Atlanta, might not have been thoroughly explored.

The researchers likely proposed a focused investigation to determine the extent of the relationship between air pollution and pediatric emergency room visits for asthma in the specific context of Atlanta. The study might have outlined the methodology, data collection processes, and statistical analyses planned to assess the association. This could include considerations for various air pollutants, exposure levels, and potential confounding factors.

In summary, Tolbert et al.'s study investigates the impact of air pollution on pediatric asthma-related emergency room visits in Atlanta. The introduction contextualizes the research within the broader issue of air quality and asthma, emphasizing the need for a targeted examination in this specific geographical location. The existing literature likely supported the general idea of a connection between air pollution and respiratory problems. The proposed study would have outlined the methods and analyses to explore this relationship in the Atlanta pediatric population. The research aimed to contribute valuable insights for public health interventions and policies to mitigate the impact of air pollution on pediatric asthma.

**[2] R.J. Martin, J.N. Dwyer, Approximations to the covariance properties of processes averaged over irregular spatial regions, Communications in Statistics-Theory and Methodologies 23 (3) (1994) 913–945.**

The referenced work by R.J. Martin and J.N. Dwyer focuses on approximations to the covariance properties of processes averaged over irregular spatial regions. In the context of statistical analysis, covariance properties provide crucial insights into the relationships between different variables. In this case, the authors are particularly interested in processes that are averaged over spatial regions, but these regions are irregular in nature.

The existing literature on spatial processes often deals with regular spatial regions. However, real-world scenarios frequently involve irregular spatial configurations. The challenge lies in developing approximations that can effectively capture the covariance properties of processes when averaged over these irregular regions. The work by Martin and Dwyer addresses this gap in existing methodologies by providing insights into approximations specifically tailored for irregular spatial settings.

The authors propose a methodology for approximating the covariance properties of processes when averaged over irregular spatial regions. This likely involves the development of new statistical techniques or the adaptation of existing ones to accommodate irregularity in spatial configurations. The focus is on providing accurate approximations that reflect the real-world complexity of irregular spatial patterns.

In summary, Martin and Dwyer's work contributes to the field of spatial statistics by addressing the challenges associated with irregular spatial regions. The proposed approach aims to enhance our understanding of covariance properties when processes are averaged over such irregularities. This work is particularly relevant for applications where irregular spatial patterns are prevalent, offering a valuable contribution to the broader landscape of statistical methodologies for spatial analysis.

**[3] B. Hrafnkelsson, N. Cressie, Hierarchical modeling of count data with application to nuclear fall-out, Environmental and Ecological Statistics 10 (2003) 179–200. H.-R. Song et al. / Journal of Multivariate Analysis 99 (2008) 1681–1697 1697**

The cited paper, "Hierarchical modeling of count data with application to nuclear fall-out" by B. Hrafnkelsson and N. Cressie, was published in the Environmental and Ecological Statistics journal in 2003. The paper focuses on the development and application of hierarchical modeling techniques to analyze count data, specifically in the context of nuclear fall-out.

The authors likely discussed the shortcomings or limitations of existing methods for modeling count data, emphasizing the need for more sophisticated and flexible approaches. Traditional statistical methods may not adequately capture the hierarchical structure or nuances present in count data, motivating the exploration of hierarchical modeling techniques.

The paper likely introduces a novel hierarchical modeling framework designed to address the challenges posed by count data, with a particular application to nuclear fall-out data. Hierarchical modeling involves incorporating multiple levels of variability into the model, allowing for a more realistic representation of the underlying data structure. The proposed methodology likely includes details on how the hierarchical model is formulated, estimated, and applied to the specific context of analyzing count data related to nuclear fall-out.

In summary, the paper provides an innovative approach to modeling count data, offering a hierarchical modeling framework. This method is applied to the specific case of nuclear fall-out data, aiming to improve the accuracy and robustness of statistical analyses in this domain. The paper likely concludes by summarizing key findings, highlighting the advantages of the proposed hierarchical modeling approach, and suggesting potential avenues for future research or applications.

**[4 ] B.P. Carlin, S. Banerjee, Hierarchical multivariate CAR models for spatio-temporally correlated survival data (with discussion), in: J.M. Bernardo, M.J. Bayarri, J.O. Berger, A.P. Dawid, D. Heckerman, A.F.M. Smith, M. West (Eds.), Bayesian Statistics 7, Oxford University Press, Oxford, 2003, pp. 45–63.**

The citation [8] refers to a paper authored by B.P. Carlin and S. Banerjee, titled "Hierarchical multivariate CAR models for spatio-temporally correlated survival data," which was included in the Bayesian Statistics 7 collection published by Oxford University Press in 2003. This paper likely focuses on statistical models designed to analyze spatio-temporally correlated survival data, with a hierarchical multivariate Conditional Autoregressive (CAR) approach.

The existing methods for analyzing spatio-temporally correlated survival data are likely discussed in the paper. Traditional statistical approaches may fall short in capturing the complex relationships between spatial and temporal dimensions in survival data. The authors might review earlier models and methodologies to highlight their limitations or areas for improvement.

The paper may introduce a novel statistical methodology for handling spatio-temporally correlated survival data, known as the "hierarchical multivariate CAR model." This proposed approach likely involves a hierarchical structure to account for various levels of correlation in the data, and the inclusion of a Conditional Autoregressive model suggests a focus on spatial dependencies. The authors might explain the advantages of their proposed model over existing methods and how it addresses the challenges inherent in spatio-temporal survival data analysis.

In summary, the paper [8] introduces a hierarchical multivariate CAR model for the analysis of spatio-temporally correlated survival data. It likely discusses shortcomings in existing methodologies, proposes a novel approach that incorporates spatial dependencies, and may provide insights into how this new model improves the understanding of survival patterns in a given context. The citation serves as a reference for those interested in advanced statistical methods for analyzing complex spatio-temporal data structures in the realm of survival analysis.

**[5] N.G. Best, K. Ickstadt, R.L. Wolpert, D.J. Briggs, Combining models of health and exposure data: The SAVIAH study, in: P Elliott, JC Wakefield, NG Best, DJ Briggs (Eds.), Spatial Epidemiology: Methods and Applications, Oxford, University Press, Oxford, New York, 2000, pp. 393–414.**

The citation [6] refers to a study titled "Combining models of health and exposure data: The SAVIAH study," conducted by N.G. Best, K. Ickstadt, R.L. Wolpert, and D.J. Briggs. This study is discussed in the context of spatial epidemiology, a field that involves analyzing the geographic distribution of diseases and their determinants.

The existing framework involves combining models of health and exposure data, as outlined in the SAVIAH study. Spatial epidemiology, as a discipline, seeks to understand how diseases are distributed in space and how various factors, including environmental exposures, contribute to these patterns. The cited work likely delves into methodologies for integrating health and exposure models to enhance our understanding of spatial patterns in disease occurrence.

The proposed methodology, likely outlined in the SAVIAH study, would aim to improve the combination of health and exposure models. This could involve refining statistical or computational techniques to better capture the relationship between health outcomes and relevant environmental or spatial factors. The goal is to enhance the accuracy and precision of spatial epidemiological models.

In summary, the citation [6] points to a study that explores the combination of health and exposure models within the context of spatial epidemiology. The existing framework, as discussed in the SAVIAH study, is likely foundational to understanding how diseases are distributed in space. The proposed improvements in methodology suggest an ongoing effort to refine and enhance the modeling approach, with the ultimate aim of better characterizing the spatial patterns of disease and their determinants.

1. **SYSTEM ANALYSIS**

**3.1 EXISTING METHOD**

The existing healthcare system lacks a cohesive approach to epidemic management. Decision-making relies on outdated models and lacks real-time data integration. Doctor-patient interactions are fragmented, hindering efficient appointment scheduling and epidemic data collection. The absence of a centralized platform results in delayed responses to emerging health crises, undermining the effectiveness of healthcare policies. An integrated solution is imperative to address these shortcomings and enhance the overall resilience of the healthcare system.

**DISADVANTAGES**

While the proposed integrated healthcare platform brings several advantages, potential disadvantages include:

**1. Technological Barriers:** Users with limited technological proficiency may face challenges navigating and utilizing the platform effectively.

**2. Data Security Concerns:** The integration of sensitive health data requires robust security measures to safeguard patient information from potential breaches.

**3. Implementation Costs:** Developing and implementing the platform may involve significant initial costs for technology integration, training, and system deployment.

**4. Resistance to Change:** Healthcare professionals and institutions may resist adopting new technologies, leading to a slow acceptance and integration process.

**5. Dependency on Data Accuracy:** The success of the platform heavily relies on the accuracy and reliability of input data, making it vulnerable to errors or misinformation.

**6. Privacy Issues:** Patient privacy concerns may arise, especially with the collection and storage of personal health information, necessitating strict compliance with privacy regulations.

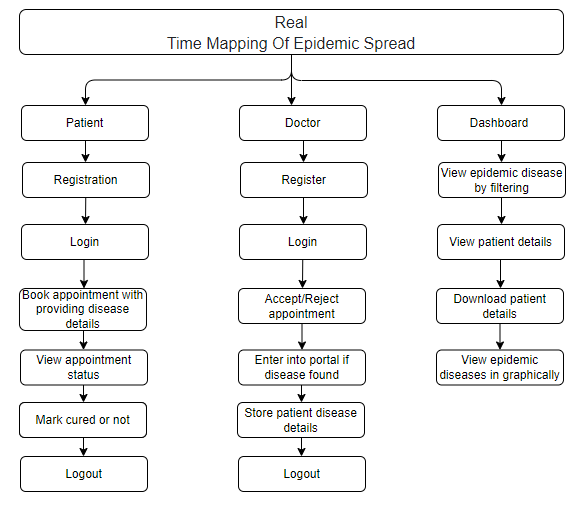
**7. Limited Accessibility:** Accessibility challenges for patients without reliable internet access or in remote areas may limit the inclusivity of the platform.

**8. Maintenance and Upkeep:** Continuous maintenance and updates are crucial to ensure the platform's effectiveness and adaptability to evolving healthcare needs.

**PROPOSED SYSTEM:**

The proposed healthcare system integrates mathematical modeling, machine learning, and practical functionalities to revolutionize epidemic response. Offering doctors tools for streamlined appointment management and real-time epidemic data input, the platform ensures efficient decision-making. Patients can easily navigate, book appointments, and monitor their recovery. The comprehensive dashboard provides detailed epidemic insights. This holistic approach aims to bridge existing gaps, enabling a proactive and data-driven healthcare system, optimizing epidemic control, and improving patient care.

**PROJECT FLOW:**



**ADVANTAGES:**

The proposed integrated healthcare platform presents several advantages:

**1. Data-Driven Decision Making:** Utilizes mathematical modeling and machine learning for accurate epidemic predictions, enabling informed and timely decision-making.

**2. Efficient Appointment Management:** Streamlines the process for doctors, allowing them to accept or reject appointments seamlessly, improving overall scheduling efficiency.

**3. Real-Time Epidemic Data Input:** Facilitates immediate and accurate recording of epidemic-related information, enhancing the system's responsiveness to emerging health crises.

**4. Patient Empowerment:** Patients benefit from easy appointment booking, status tracking, and the ability to update their recovery status, promoting active involvement in their healthcare journey.

**5. Comprehensive Dashboard:** Provides a centralized view of epidemic details with filtering options, downloadable patient data, and graphical representations, enhancing overall situational awareness.

**6. Proactive Epidemic Control:** The platform's holistic approach enables proactive measures in epidemic control, contributing to a more resilient and adaptive healthcare system.

**7. Optimized Healthcare Policies:** Offers insights for policymakers with threshold mechanisms, aiding in the formulation of targeted and effective healthcare policies based on real-time data and predictions.

**4. REQUIREMENT ANALYSIS**

**4.1 Functional and non-functional requirements**

Requirement’s analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and non-functional requirements.

**Functional Requirements**: These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Examples of functional requirements:

1. Authentication of user whenever he/she logs into the system
2. System shutdown in case of a cyber-attack
3. A verification email is sent to user whenever he/she register for the first time on some software system.

**Non-functional requirements**: These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.  
They basically deal with issues like:

* Portability
* Security
* Maintainability
* Reliability
* Scalability
* Performance
* Reusability
* Flexibility

Examples of non-functional requirements:

1. Emails should be sent with a latency of no greater than 12 hours from such an activity.
2. The processing of each request should be done within 10 seconds
3. The site should load in 3 seconds whenever of simultaneous users are > 10000

**HARDWARE & SOFTWARE REQUIREMENTS**

# H/W CONFIGURATION:

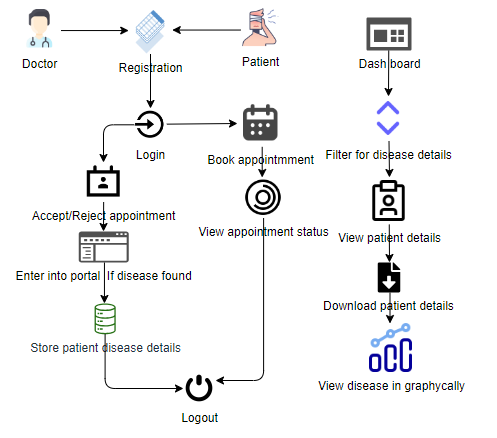
# Processor - I3/Intel Processor

* Hard Disk -160GB
* Key Board - Standard Windows Keyboard
* Mouse - Two or Three Button Mouse
* Monitor - SVGA
* RAM - 4Gb

**S/W CONFIGURATION:**

* Operating System : Windows 7/8/10
* Server side Script : Python, HTML, MYSQL, CSS, Bootstrap.
* Libraries : PANDAS, Flask, Smtlib
* IDE : PyCharm
* Technology : Python 3.6+

**4.3 ARCHITECTURE:**



**5. SYSTEM DESIGN**

**5.1 Introduction of Input Design:**

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties −

* It should serve specific purpose effectively such as storing, recording, and retrieving the information.
* It ensures proper completion with accuracy.
* It should be easy to fill and straightforward.
* It should focus on user’s attention, consistency, and simplicity.
* All these objectives are obtained using the knowledge of basic design principles regarding −
  + What are the inputs needed for the system?
  + How end users respond to different elements of forms and screens.

### Objectives for Input Design:

The objectives of input design are −

* To design data entry and input procedures
* To reduce input volume
* To design source documents for data capture or devise other data capture methods
* To design input data records, data entry screens, user interface screens, etc.
* To use validation checks and develop effective input controls.

**Output Design:**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

### Objectives of Output Design:

The objectives of input design are:

* To develop output design that serves the intended purpose and eliminates the production of unwanted output.
* To develop the output design that meets the end user’s requirements.
* To deliver the appropriate quantity of output.
* To form the output in appropriate format and direct it to the right person.
* To make the output available on time for making good decisions.

**5.2 UML DIAGRAMS**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

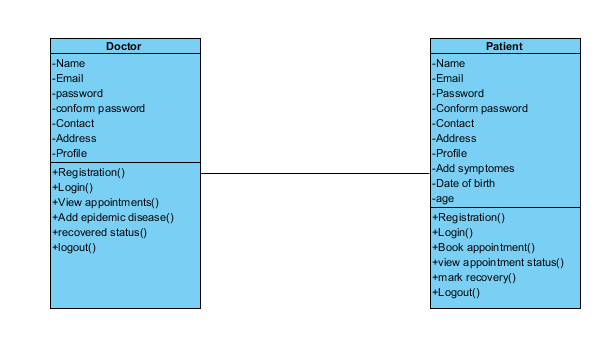
**USE CASE DIAGRAM**

* A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis.
* Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.
* The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



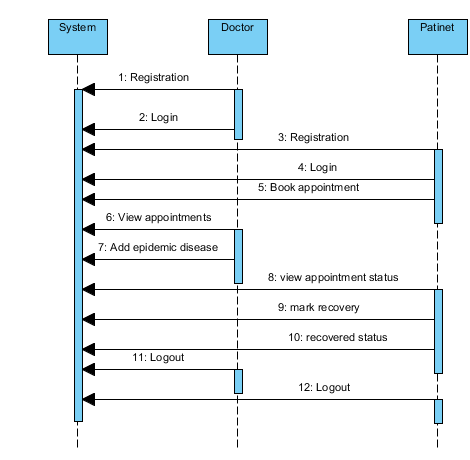
**CLASS DIAGRAM**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information



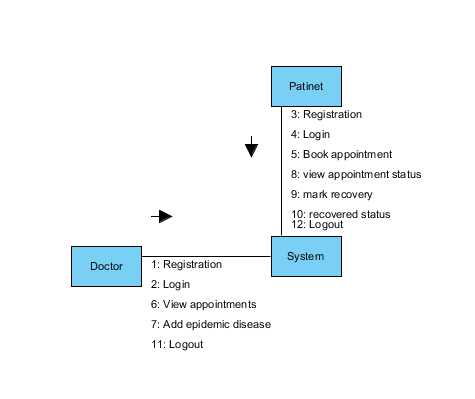
**SEQUENCE DIAGRAM**

* A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order.
* It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams



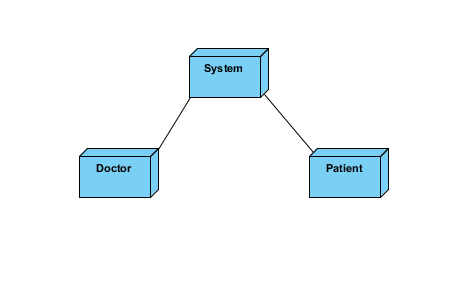
**COLLABORATION DIAGRAM:**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.



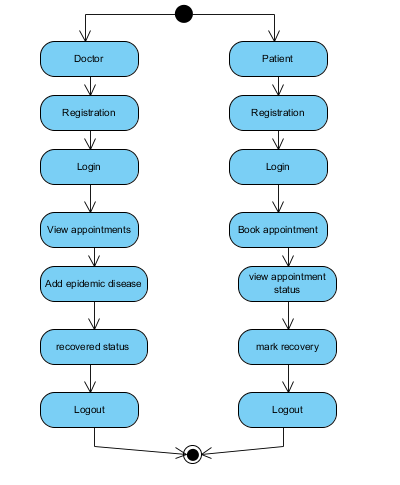
**DEPLOYMENT DIAGRAM**

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.



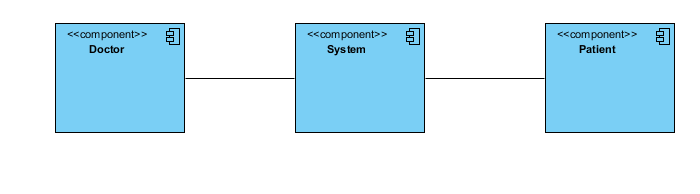
**ACTIVITY DIAGRAM:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



**COMPONENT DIAGRAM**:

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical **c**omponents in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required function is covered by planned development.



**ER DIAGRAM:**

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

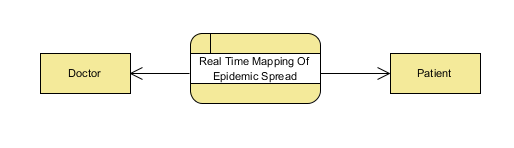
An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

# 

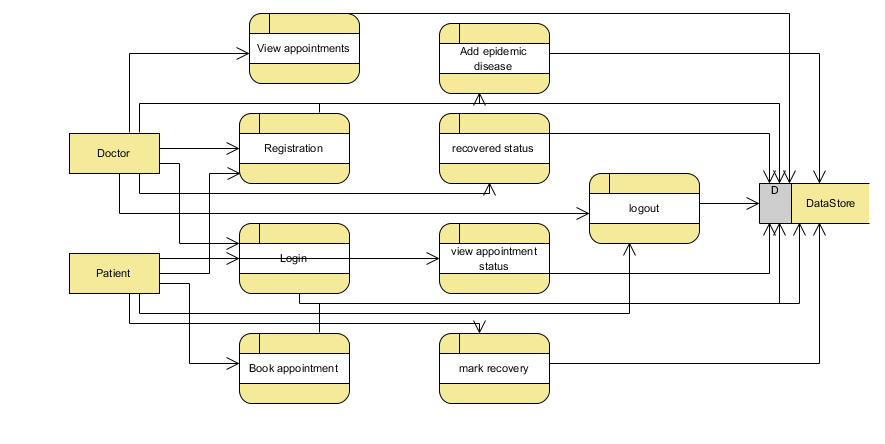
**DFD DIAGRAM:**

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who psslays a part in the system that acts as the starting point for redesigning a system.

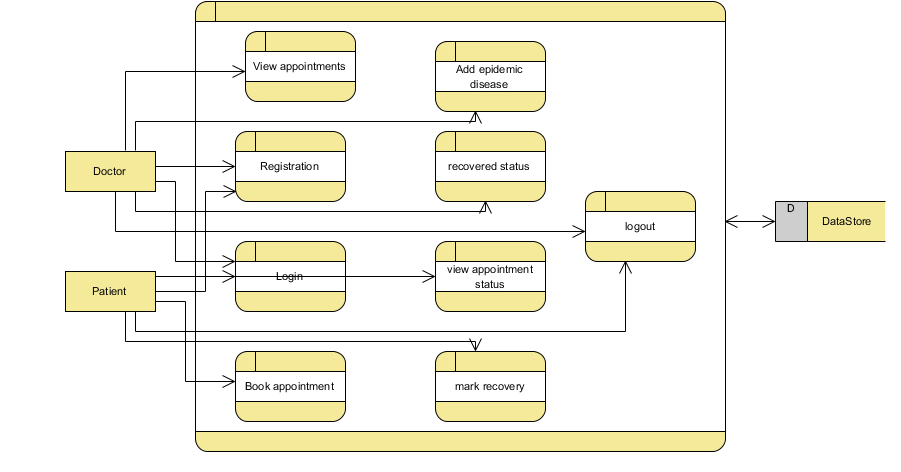
**Level 0:**

****

**Level-1 Diagram:**



**Level-2 Diagram:**



1. **IMPLEMENTATION AND RESULTS**

**1. Doctor:**

**Register:** The doctor can register with the details like name, email, password, conform password, Contact number, and address

**Login:** After registration they will login with email and password

**View appointments:** Doctors can view all Symptoms which is given by the patients and based on the symptoms he can enter the data like patient is having the Epidemic Spread.

**Add epidemic disease:** Once the doctor can view the symptoms he has to add like patient is having the epidemic disease.

**Recovered status:** Once the patient is mention like he has quire for the epidemic disease doctor can view that status.

**Logout**: After completion of the operation they can logout from the website.

**2. Patients:**

**Register:** The patients can register with the details like name, email, password, conform password, Contact number, and address

**Login:** After registration they will login with email and password

**Book appointment** Patients can add the symptoms like they are suffering.

**view appointment status** Patients can view the status of their request and they can check whether there are having the epidemic disease or not.

**Mark recovery:**  The patients can mark or update like they have queried or not form epidemic disease.

**Logout**: After completion of the operation they can logout from the website.

**Dashboard:** Here we can view that like how many people are suffering from the disease it will show in graphical manner.

**6.4 RESULTS AND OUTPUT SCREENS:**

**Home Page:** Secure data dynamics and public auditing scheme for cloud storage home pages

**8. SYSTEM STUDY AND TESTING**

**8.1 Feasibility Study**

The feasibility of the project is analysed 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.

**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.

**8.2 Types of Tests**

**8.2.1 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.

**8.2.2 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.

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.

**Acceptance 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.

**8.2.3 Functional testing**

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.

**9.2.4 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.

**9.2.5 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.

**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.
* **TEST CASES:**

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Result** |
| Input image | Real Time Mapping Of Epidemic Spread | Success |

* **Test cases Model building:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.NO** | **Test cases** | **I/O** | **Expected O/T** | **Actual O/T** | **P/F** |
| 1 | Registration Doctor | Enter Name, email, age, password, conform password, address, contact. | Data added successfully | Added successfully | P |
| 2 | Login Doctor | Enter name and password | Login successfully | Login success | P |
| 3 | Login Doctor | Enter name and password | Password not matched | Login fail | F |
| 4 | Book appointment | Enter Symptoms, age ,date of birth, etc., | Data added successfully | Added successfully | P |
| 5 | Add epidemic disease | Enter Present or Absent | Data added successfully | Added successfully | p |
| 6 | Add epidemic disease | Enter Present or Absent | Failed to add the data | Data adding failed | F s |

1. **CONCLUSION**

In conclusion, the proposed integrated healthcare platform emerges as a transformative solution to the existing challenges in epidemic management. By blending mathematical modeling, machine learning, and practical healthcare functionalities, the system addresses crucial gaps, offering efficient appointment management for doctors and active patient engagement. Despite potential drawbacks such as technological barriers and data security concerns, the platform's overarching goal is to optimize pandemic response, foster a resilient healthcare system, and enhance overall epidemic control and patient care. With its data-driven approach, this platform signifies a significant leap towards a proactive and adaptive healthcare landscape.

**10. FUTURE SCOPE**

The feature scope includes streamlined doctor-patient interactions, real-time epidemic data input, and an intuitive dashboard for comprehensive insights. With efficient appointment management, patients can easily register, track statuses, and update recovery information. The platform employs mathematical modeling and machine learning, empowering doctors for data-driven decision-making. Downloadable patient data and graphical representations enhance informed analysis, contributing to a holistic and proactive approach in epidemic control, optimizing healthcare policies for a resilient system.

**11. REFERENCES**

[1] G. Arfken, H.J. Weber, Mathematical Methods for Physicists, 4th edn, Academic Press, San Diego, CA, 1995.

[2] J. Besag, Spatial interaction and the statistical analysis of lattice systems, Journal of the Royal Statistical Society 36 (1974) 192–236.

[3] J. Besag, On a system of two-dimensional recurrence equations, Journal of the Royal Statistical Society 43 (1981) 302–309.

[4] J. Besag, C. Kooperberg, On conditional and intrinsic autoregressions, Biometrika 82 (1995) 733–746.

[5] J. Besag, J.C. York, A. Mollie, Bayesian image restoration, with two applications in spatial statistics (with discussion), Annals of the Institute of Statistical Mathematics 43 (1991) 1–59.

[6] N.G. Best, K. Ickstadt, R.L. Wolpert, D.J. Briggs, Combining models of health and exposure data: The SAVIAH study, in: P Elliott, JC Wakefield, NG Best, DJ Briggs (Eds.), Spatial Epidemiology: Methods and Applications, Oxford, University Press, Oxford, New York, 2000, pp. 393–414.

[7] D.R. Brillinger, Time Series: Data Analysis and Theory, Holden-Day, San Francisco, 1981.

[8] B.P. Carlin, S. Banerjee, Hierarchical multivariate CAR models for spatio-temporally correlated survival data (with discussion), in: J.M. Bernardo, M.J. Bayarri, J.O. Berger, A.P. Dawid, D. Heckerman, A.F.M. Smith, M. West (Eds.), Bayesian Statistics 7, Oxford University Press, Oxford, 2003, pp. 45–63.

[9] H. Cramer, M.R. Leadbetter, Stationary and Related Stochastic Processes. Sample Function Properties and their Applications, Wiley, New York, 1967.

[10] N. Cressie, Statistics for Spatial Data, Wiley, New York, 1993.

[11] M. Fuentes, H.R. Song, S. Ghosh, D. Holland, J. Davis, Spatial association between speciated fine particles and mortality, Biometrics 62 (2006) 855–863.

[12] D. Griffith, F. Csillag, Exploring relationships between semivariogram and spatial autoregressive models, Papers in Regional Science 72 (1993) 283–295.

[13] B. Hrafnkelsson, N. Cressie, Hierarchical modeling of count data with application to nuclear fall-out, Environmental and Ecological Statistics 10 (2003) 179–200. H.-R. Song et al. / Journal of Multivariate Analysis 99 (2008) 1681–1697 1697

[14] A.G. Journel, Ch.J. Huijbregts, Mining Geostatistics, Academic Press, London, 1978.

[15] R.J. Martin, J.N. Dwyer, Approximations to the covariance properties of processes averaged over irregular spatial regions, Communications in Statistics-Theory and Methodologies 23 (3) (1994) 913–945.