ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Integrated Global Medical History System (IGMHS)

P. Rohini Devi [1], B. Arun Kumar [2], S. Vasantha [3], B. Vamsi Sai Ram [4]
Dr. Penumerthy Vishnu*

1,2, 3, 4 Student, Dept of CSE, NS Raju Institute of Technology Visakhapatnam, Andhra Pradesh, India
* Professor, Dept. of CSE, NS Raju Institute of Technology Visakhapatnam, Andhra Pradesh, India
rohini1651999@gmail.com, arunkumarballa503@gmail.com, vasanthvassu823@gmail.com, vamsibalivada98@gmail.com, vishnnu_mahesh@yahoo.com

Abstract

Our today's lifestyle puts health and healthcare at the highest priority in our journey of life. All of us are conscious on health, our social and economic life is globalized. Due to stressful and hectic activities, we need timely and continuous availability of medical and health services. This necessitates the availability of medical history and records of all of us as time demands. Traditionally, the medical records and data was owned or known to only local medical practitioners in local area. But, today due to globalization, we travel often; we do not know when the emergencies crop up. As technological developments are progressing at geometric progression, in specific Information and Digital Technologies, an **Integrated Global Medical History System (IGMHS)** has great utility. This, along with our knowledge acquired thru seminar motivated and prompted us to choose this project subject. Besides the **Health domain, the data science. Big data, Data integration** have become buzz words of today's IT industry.

IGMHS is intended to Integrate Global Medical History of all citizens and provide timely information to medical practioners. This necessitates us to Integrate Data from multiple locations, multiple source, do data messaging, data transformation, data compression, and prepare and maintain a Big Data Store. **IGMHS** core is individual or need to extract individual medical history and records of a specific citizen of a specific country, this leads us to extract small information from large volume of Data, in short **"SMALL DATA FROM BIG DATA. IGMHS** also demands usage of data Visualization techniques, methods and language translations. The goal and successes parameter of **IGMHS** is the availability of medical history of any human, at any time, at any location on the globe with few clicks on Monitor. Though the reality of **IGMHS** is dreamy, huge task, lots of barriers but still we are positive and think there must be a start for everything.

Keywards: IGMHS, Small Data from Big Data,

1. Introduction

Traditionally the medical data of patients was in paper files and usually kept or stored with patients. In normal times it is fine but in case of need or emergency, if, doctors need medical history or past diagnosis, the patients may not find it immediately or might have

Cite this article as: P.Rohini Devi, B.Arun Kumar, S.Vasantha, B.Vamsi Sai Ram and Dr.Penumerthy Vishnu "Integrated Global Medical History System (IGMHS)", International Journal & Magazine of Engineering, Technology, Management and Research (IJMETMR), ISSN 2348-4845, Volume 7 Issue 4, 2020, Page 7-16

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

even lost. Sometimes, keeping for years the papers may deplete and lost info. This is too difficult in case if, we need info relating to 10 or 15 years of pass time. This problem might be incremental because of frequent changing doctors, requirement of medical treatment at different locations at different times. The problem aggregates further due to illiteracy and lack of knowing the importance of medical data. The problem statement can be quoted as "Having the medical data of a patient in times of need, frequency, accuracy, beyond the borders with least expense".

Suppose if, we consider a person, who is travelling from India to US .he may fall ill at any time or anywhere. At that point of time, the people around him, or the doctors do not know about his health history. He cannot carry all his medical history all the time. Hence, there is a problem in getting previous medical data, whenever and wherever possible.

1.2 Proposed System

IGMHS is intended to Integrate Global Medical History of all citizens of all countries and provide timely information to medical practitioners. Our idea is to Distributed Server's store medical history of all citizens of all countries. A universal medical card ID is issued to the patient, through this either the doctor or patient should be able to access his past info. This system integrates data from multiple sources and locations prepares data for "**Big Data Store**" IGMHS core is individual, it needs is to extract individual medical history and records of a specific citizen of a specific country, i.e. extract small information from large volume of Data, in short "**Small Data from Big Data**".

IGMHS demands usage of data Visualization techniques, methods and language translations. Medical history is huge. IGMHS is a system in which all the medical information from birth to death is collected worldwide. The goal and success parameter of IGMHS is the availability of medical history of any human, at any time, at any location on the globe with few clicks. Though the reality of IGMHS is dreamy, huge task, lots of barriers but still we are positive and think there must be a start for everything.

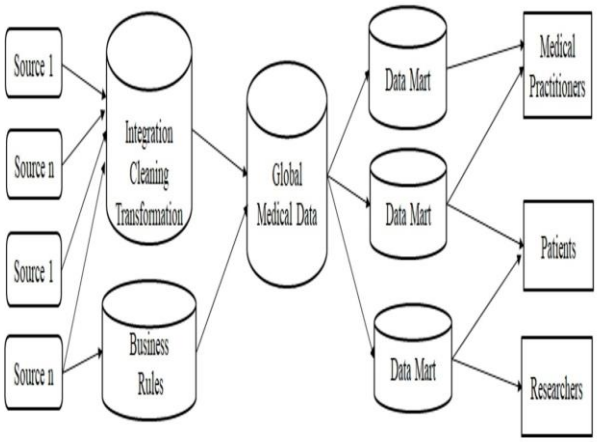
The scope of this project work is to develop a prototype of application with basic Modules. The main Functionality of Modules, data Integration, basic user queries is implemented. Though in reality we need live data, we tested the application with our own test data. Due to time and other resource constrains, this is only a proof of Concept.

2.1 System Architecture

A system architecture or systems architecture is the conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system.

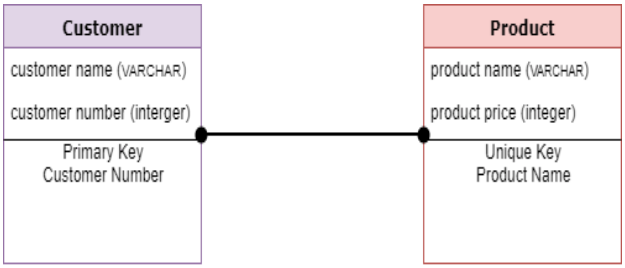
Following is the Architecture Diagram for IGMHS

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal



2.2 Physical Diagram
Physical Data Model

A Physical Data Model describes the database specific implementation of the data model. It offers an abstraction of the database and helps generate schema. This is because of the richness of meta-data offered by a Physical Data Model.

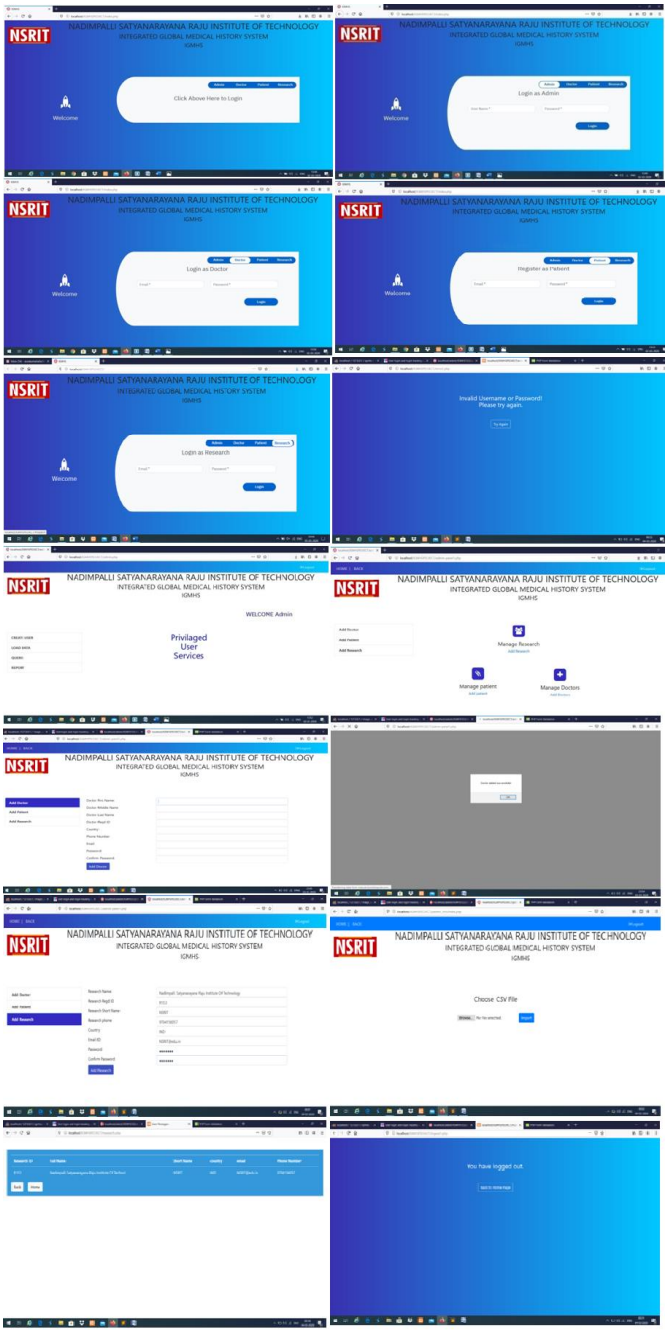


This type of Data model also helps to visualize database structure. It helps to model database columns keys, constraints, indexes, triggers, and other RDBMS features.

Characteristics of a physical data model:

- The physical data model describes data need for a single project or application though it maybe integrated with other physical data models based on project scope.
- Data Model contains relationships between tables that which addresses cardinality and nullability of the relationships.
- Developed for a specific version of a DBMS, location, data storage or technology to be used in the project.
- Columns should have exact data types, lengths assigned and default values.

3.1 IGMHS - Input Screen and Output Screen



ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Fig. S1: Main Screen login Screen, S2: Admin login screen, S3: Doctor login Screen , S4: Patient login Screen, S5: Research Login screen, S6:Given invalid input to the admin for the login screen, S7: Showing the output screen for admin while enter Invalid username and password, S8: Given valid input to the admin for the login screen, S9: output screen for admin login, S10: After click on the Create user we will get the below output Screen, S11: After click on the Add Doctor we will get the below output Screen, S12: Below screen is enter the data of the doctor detail, S13: output Screen for add Doctor detail, S14: After clicking ok on the above screen we get s10 then we click on the add Patient we get the below screen , S15: Below Screen is enter the data of the patient detail., S16: Output Screen for add patient detail, S16: After clicking ok on the above screen we get s10 then we click on the add Research we get the below screen, S17: Below Screen enter the data of the research detail, S18: Output Screen for add research detail, S19: After Clicking ok on the above screen. We get s10. Then we click on home button it got to the below screen. S20: When we click on the load data. We get the below page. S21: In the below screen we have Selected one for dropdown list. Then we click on submit button. S22: After click on the submit button. We get the below page. S23: click on the Browse and select the excel file with the extend .csv and click on import button S24: After click on the import button. It show the below output screen S25: After click on the home button it go to the below page, S26: After click on the queries button we get the below page and click on the doctor list. S27: Enter the Doctor unique ID and click on the search button, S28: output screen for search doctor unique id is below, S29: After click on the back button we will get the screen then we click on patient list, S30: Enter the International PatientID and click on the search button, S31: output screen for International Patientid is below, S32: enter Research ID, S33: output for research id, S34: In the below screen if you enter Doctor Id we get the list of Patient details who had went to that doctor. S35: Below is the logout screen for our Application

3.2 Execution Sample Code

Sample code Contain for the HTML, CSS, JS and PHP

Sample Code of PHP : fun3.php

```
<?php
session_start();
/*database connectivity*/
$con=mysqli_connect("localhost","root","
","igmhs");
/* admin Login data */
if(isset($_POST['adsub'])){
        $username=$_POST['username1'];
        $password=$_POST['password2'];
   $query="select * from users where
user_type_code='1' and
user_name='$username' and
password='$password';";
        $result=mysqli_query($con,$query);
        if(mysqli_num_rows($result)==1)
        {

        $_SESSION['username']=$username;
                header("Location:admin.php");
        }
        else
                header("Location:error2.php");
}
?>
```

4.1 System Testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based

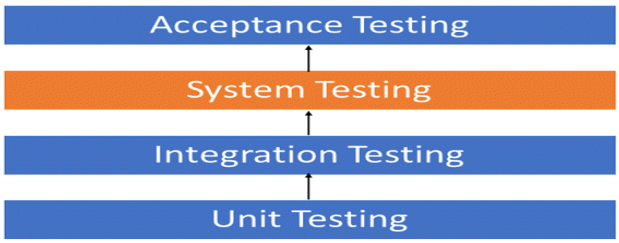
ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

system. Ultimately, the software is interfaced with other software/hardware systems. System Testing is actually a series of different tests whose sole purpose is to exercise the full computer-based system. There are different types of Tests performed on Software Code,

Some of the important types of testing's and their hierarchy are:

4.2 Types of Testing:

Software Testing Hierarchy:

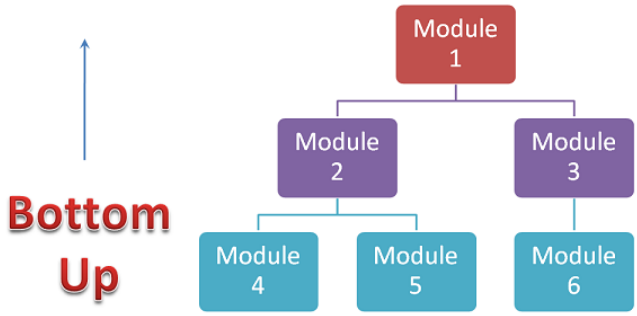


As with almost any software engineering process, software testing has a prescribed order in which things should be done. The following is a list of software testing categories arranged in chronological order. These are the steps taken to fully test new software in preparation for marketing it:

- Unit testing performed on each module or block of code during development. Unit Testing is normally done by the programmer who writes the code.
- Integration testing done before, during and after integration of a new module into the main software package. This involves testing of each individual code module. One piece of software can contain several modules which are often created by several different programmers. It is crucial to test each module's effect on the entire program model.
- System testing done by a professional testing agent on the completed software product before it is introduced to the market.
- Acceptance testing - beta testing of the product done by the actual end users.

4.2.1 Bottom – Up Integration

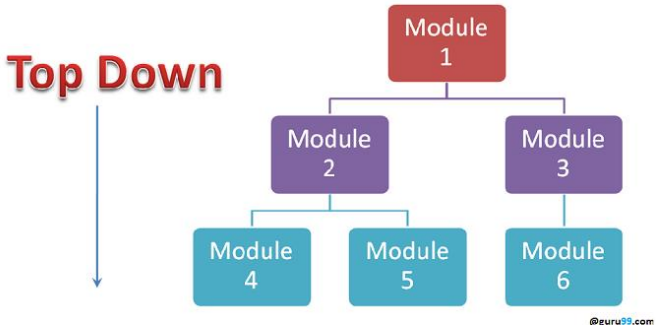
Diagrammatic Representation:



4.2.2 Top-down Integration:

In Top to down approach, testing takes place from top to down following the control flow of the software system. Takes help of stubs for testing.

Diagrammatic Representation:



The various **Different Types of System Testing include:**

1. Usability Testing- mainly focuses on the user's ease to use the application, flexibility in handling controls and

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

ability of the system to meet its objectives

2. Load Testing- is necessary to know that a software solution will perform under real-life loads.
3. Regression Testing- involves testing done to make sure none of the changes made over the course of the development process have caused new bugs. It also makes sure no old bugs appear from the addition of new software modules over time.
4. Recovery testing - is done to demonstrate a software solution is reliable, trustworthy and can successfully recoup from possible crashes.
5. Migration testing- is done to ensure that the software can be moved from older system infrastructures to current system infrastructures without any issues.
6. Functional Testing - Also known as functional completeness testing, Functional Testing involves trying to think of any possible missing functions. Testers might make a list of additional functionalities that a product could have to improve it during functional testing.
7. Hardware/Software Testing - IBM refers to Hardware/Software testing as "HW/SW Testing". This is when the tester focuses his/her attention on the interactions between the hardware and software during system testing.

4.3 Acceptance Testing:

Acceptance Testing is a method of software testing where a system is tested for acceptability. The major aim of this test is to evaluate the compliance of the system with the business requirements and assess whether it is acceptable for delivery or not.

Types of Acceptance Testing:

1. **User Acceptance Testing (UAT):** User acceptance testing is used to determine whether the product is working for the user correctly. Specific requirements which are quite often used by the customers are primarily picked for the testing purpose. This is also termed as *End-User* Testing.
2. **Business Acceptance Testing (BAT):** BAT is used to determine whether the product meets the business goals and purposes or not. BAT mainly focuses on business profits which are quite challenging due to the changing market conditions and new technologies so that the current implementation may have to being changed which result in extra budgets.
3. **Contract Acceptance Testing (CAT):** CAT is a contract which specifies that once the product goes live, within a predetermined period, the acceptance test must be performed and it should pass all the acceptance use cases. Here is a contract termed as Service Level Agreement (SLA), which includes the terms where the payment will be made only if the Product services are in-line with all the requirements, which means the contract is fulfilled.

   Sometimes, this contract happens before the product goes live. There should be a well defined contract in terms of the period of testing, areas of testing,

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

conditions on issues encountered at later stages, payments, etc.

4. **Operational Acceptance Testing (OAT):**
   OAT is used to determine the operational readiness of the product and is a non-functional testing. It mainly includes testing of recovery, compatibility, maintainability, reliability etc.
   OAT assures the stability of the product before it is released to the production.

4.4 Verification and Validation:

Verification and validation are independent procedures that are used together for checking that a product, service, or system meets requirements and specifications and that it fulfills its intended purpose. Verification is the process of checking that software achieves its goal without any bugs. It is the process to ensure whether the product that is developed is right or not. It verifies whether the developed product fulfils the requirements that we have.
Software Validation is a process of evaluating software product, so as to ensure that the software meets the pre-defined and specified business requirements as well as the end users/customers' demands and expectations. Both, the verification and validation is a software testing activity, and verification is followed by the validation. Validation is usually carried out at the end of the software development. It is basically, performed with the intent to check that whether the developed software is built as per pre-decided software requirement specifications (SRS) and if it caters to full fill the customers' actual needs in the real environment.

5. Implementation

Implementation simply means carrying out the activities described in our work plan. The implementation phase is where you and your project team actually do the project work to produce the deliverables. ... This phase is typically where approved changes are implemented. Most often, changes are identified by looking at performance and quality control data.

5.1 Objectives of the Implementation Phase:

The objectives of the implementation phase can be summarized as follow:

- Putting the action plan into operation.
- Achieving tangible change and improvements.
- Ensuring that new infrastructure, new institutions and new resources are sustainable in every aspect.
- Ensuring that any unforeseen conflicts that might arise during this stage are resolved.
- Ensuring transparency with regard to finances .
- Ensuring that potential benefits are not captured by elites at the expenses of poorer social groups.

5.2 Project Implementation Activities:

1) Project Implementation Starts With a Plan. ...
2) Involve all the Operations Team. ...
3) Prepare Training Materials. ...
4) Schedule Enough Time for Training.
5) Communicate, Communicate, Communicate.
6) Prepare Operational Manual ,User Manual and Technical Manual

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

5.3 Software Implementation Process – Issues and Challenges:

- Software Legacy
- Software Certifications
- IT Infrastructure and Integration
- Change Management Strategy and Software Implementation
- Changes in the Team Structure
- Software Portability and Backward Compatibility
- Data Migration
- Logistic Problems in Software Implementation phases

5.4 Smooth Software Implementation Process Flow:

1) Constant monitoring is everything. Audit each milestone in coordination with the software provider.
2) Change needs to be thoroughly planned using project management suites or change management software but complex subtask-lists can be managed with software like process.
3) Always keep track of tools, processes, and people. Without a proper change management plan the roll out process is often ineffective. As mentioned above, it's all about setting a goal, assigning proper metrics, and push for maximum software adoption to meet the ROI requirements. Also define specific milestones and criteria for success.
4) Create a solid task force and allow innovators and tech enthusiasts in different teams to support the implementation team and motivate the most risk-averse members of their teams by spreading the vision.
5) Involve the software vendor in the preparation phases. Appoint a change manager, somebody in charge of operations management who is responsible for the entire rollout. Make sure that all parties involved also carry out a proven risk management analysis and plan. The combination of risk management and change management will lead to a healthy balance of traditionalism and innovation.

5.5 Advantages:

1) Implementation gives the opportunity to see the plans become a reality
2) Execution of projects allows end-users to have access to better services and living environment
3) Success stories and experiences can be shared with specialists from other cities and towns, encouraging others to adopt similar approaches, which in turn may improve water resources management in the local area

5.6 The Diagrammatical or Pictorial Representation of Software Release Management:

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

5.7 Software Deployments – Steps:

1) Creating and maintaining up-to-date and ready-to-install software packages.
2) Configuring the target computers before the installation or un-installation of the package.
3) Installing or un-installing the software on the target computers.
4) Configuring the target computers post installation or un-installation.
5) Upgrading existing software.

5.8. IGMHS – Implementation:

As IGMHS is presently a Prototype, we implemented it on one of the Hardware that includes both server and client components.

6. Conclusion:

As a prototype project we tried, implementing all features required for a full fledged software project. The Initial Planning, Analysis and Design phases gave us good analytical skills. In problem identification phase we had good brainstorming capabilities and increased our thinking capabilities. In design phase, we learned the practical utility of software engineering concepts and their applications in live projects. The development stage was interesting as we have to learn new concepts like connectivity of client code with database server, and parameterize middleware with appropriate port numbers, host names, database name etc., The testing was good as we could see the output and in good number of case exceptions and fixing logic of the code. Though we did not have full experience of implementing live, but porting the application, setting the environment in new hardware was interesting. The implementation taught us the usefulness of scripting some additional code like table creation and windows batch files.

In totally, the involving and doing software project was good learning, interesting and inspiring. Though the reality of **IGMHS** is dreamy, huge task, lots of barriers but still we are positive and think there must be a start for everything.

7. References:

7.1 Text Books:

Software Engineering:

1. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
2. Software Engineering, a Precise Approach, Pankaj Jalote, Wiley India, 2010

DBMS & PL/SQL:

1. Introduction to Database Systems, CJ Date, Pearson
2. Database Systems - The Complete Book, H G Molina, J D Ullman, J Widom Pearson
3. Oracle: The Complete Reference by Oracle Press

Data Warehousing and Data Mining:

1. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson.
2. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier.

WEB technologies:

1. Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
2. Web Technologies, Uttam K Roy, Oxford

Software Project Management:

1. Bob Hughes & Mike Cotterell, TATA Mcgraw-Hill
2. Software Project Management, Walker Royce: Pearson Education, 2005.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Web Links:

- https://www.healthcareitnews.com/news/go-figure-key-big-data-actually-small-data
- https://www.macadamian.com/learn/big-data-vs-small-data