Epidemic Analysis
(Web Application for Epidemic Analysis and Prediction)

[1]Ankit, [1]Aakash Pathak, [1]Anmol Khandelwal, [1]Ashish Sinha, [2]Sini Anna Alex
[1]Department of Computer Science, MS Ramaiah Institute of Technology, Bangalore
[2]Associate Professor, Department of Computer Science, MS Ramaiah Institute of Technology,
Bangalore reach.ankit@yahoo.com, aakash.msrit@gmail.com, anmolkhandelwal100@hotmail.com, ashishnature@gmail.com,
sinialex@msrit.edu

Abstract— There are times when people find themselves in a state of medical emergency and they have no idea which Is the idea hospital to get the best medical attention. Availability of an application which can process existing information and give them enough ground to choose from a set of options can have a significant impact on their lives. This will save them a lot of time and money. Also, it may even save someone's life.

In this paper, we describe our attempts to develop and deploy a relational database system on our server and then connect to this server through web browsers. Smartphones are reaching the rural areas quite rapidly but people don't have the knowledge and skills to use the phones to its full potential. An easy-to-use application just helps them in the task of finding the most suitable hospital available to them. A very important pre-requisite for such an informative server is to collect the data from different hospitals and clinics. This attempt is based on a dummy data set. Our focus is to make the application functional on the web and then make it work on a smartphone.

A lot of factors and questions are analyzed by the application like: the most serious epidemic in an area; number of people affected by a particular disease; hospitals which serve the needs of people suffering from a particular disease; which places need more specialized doctors and similar questions are meant to be answered. Chemists can use this information to keep themselves available with the drugs and medicines needed in case of a particular epidemic beforehand. The work presented in this paper is part of a planned long term initiative to develop a way to help people, and make significant benefits to the society.

Index Terms— **HANA:** in-memory computing database, **SAP:** A German product development company, **SAPUI5:** Online library available for implementing SAP's user interface, **XSJS**: JavaScript file used for server side script.

——————————— ◆ ———————————

1 INTRODUCTION

INTERNETIZATION is a word that was coined by Professor Constantine Passaris[1] of the University of New Brunswick (Canada). It describes the pervasive and far reaching influence of the internet and the World Wide Web on all aspects of human endeavour in our contemporary society. Internetization has had a significant influence on the scope and magnitude of the new global economy of the 21st Century. People use internet for almost everything these days. They need to find coffee shops, stationary shops, vegetable shops, gift shops and a lot of other things. Numerous applications are available for such kind of use. It has been a general trend where these web applications are first deployed on web servers and accessed through browsers and then the same servers are connected through various mobile operating systems, like Android, Blackberry, Windows, Apple etc.

All different kinds of applications are developed for human benefit and are meant to make their lives a lot easier and better. It reduces the time they need to perform a search and saves them from a lot of hard work too.

We plan to implement the application with data from a specific region. Our focus is to choose a few districts in Karnataka, and use the medical history of these places in our application servers. The initial implementation is just to understand how the application will function. Around fifty

areas with different pin codes will be chosen and information about the patients, diseases and doctors of specific hospitals will be used. Currently what we plan to use is a dummy data set which can be easily mapped to the real data after the application works successfully. Each area is expected to have about three or four hospitals with an average number of ten departments. Also, each department will have around four or five doctors, which makes it an average number of thirty five doctors in each hospital. The information about the hospitals, doctors, diseases and patients will be kept in tables used in relational database management systems. These tables are what we call master tables. These tables contain very basic information about the entities which will be explained later. The fine details like the name, place, type of the entity is contained in the master tables. Another table, transaction table, will contain the information about each case, explaining which patient approached which hospital and the illness he was suffering from. All these tables will be made accessible to the users of the application and they can perform a search on a particular hospital, or an epidemic. They can learn about the death rates of a particular area, the percentage of people suffering from an illness, best place for the cure of such a disease, etc.

For analysis, we use an online available library which will

be used to plot graphs and pie charts which can compare the impact of different diseases or the number of hospitals in a region. The graph gives a visual insight about the effects of different diseases. All this information will make it easy for a person to predict which epidemic can occur at a particular region. Though it cannot be ensured that the prediction will be correct at all times but it will be a very helpful parameter in epidemic study. This is not only a cost-effective method but also an efficient way to keep the data organized and available at all times to the user of the application.

2 COMPOSITION OF THE APPLICATION

Our plan is to use 3-tier architecture. The application server layer in the middle provides stability to the application. Our preference is to use MySQL[4] as the back end support since a lot of documentation is already present on it. JavaScript[5] will be used for the front-end designing. A user interface library called SAPUI5[6] is available online which will be used together with JavaScript. The details of each of these layers are pointed out in the next few sections.

3 BACK END - MYSQL

MySQL is the world's most used open source relational database management system as of now that runs as a server providing multi-user access to a number of databases. The SQL phrase stands for Structured Query Language.

MySQL is a proper choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack. Free software-open source projects that require a full-featured database management system often use MySQL.

Tables	Attributes				
Doctor	Unique ID	Name	Hospital ID	Specialty	
Hospital	Unique ID	Name	No. of doctors	Area	
Patient	Unique ID	Name	Disease ID	Area	
Disease	Unique ID	Name	Percentage of people affected	Death rate	
Transaction	Unique ID	Patient ID	Disease ID	Doctor ID	Hospital ID

Fig 1: The most important attributes of the master table and the transaction table.

As discussed in the 'Introduction' section, our plan is to make 4 master tables. The master database records all the system-level information for a SQL server system. This includes instance-wide metadata such as logon accounts, endpoints, linked servers and system configuration settings. The 4 master tables will have information about Doctor, Hospital, Patient and Disease. The attributes of the 4 tables can be easily understood by figure 1 above. Another table called the transaction

table will have the detail of every case, it will show which patient was suffering from which disease and is associated with which hospital and doctor. The details of this table can also be understood from the table shown above.

4 FRONT END - JAVASCRIPT

There are a lot of advantages of using JavaScript as the front end programming language in web applications. Our focus here is to use the MVC (Model-View-Controller) framework because it is easy to code and understand. Different views are associated with their own controller logic in the MVC architecture.

When we use JavaScript MVC framework, much of the back-end becomes a restful API which rarely changes. This leads us to have a good architecture with only one back-end API's. Front end can be any mobile device or web applications. User experience logic becomes concentrated in JavaScript which allows us to push new UI changes often without breaking anything. To perform several complicate functions, we have many libraries support which can be easily included in or app.

The library which will be used for the complicated functionalities is SAPUI5 library kit. It is developed by the developers of SAP Labs and is available online on the following link: https://openui5.hana.ondemand.com. It can be used by installing the software in any version of java eclipse. We plan to use the newest, i.e. Kepler version. The home view is planned to contain the different tables which can be navigated through the navigation panel. A snapshot is shown in figure 3 in the following page.

Each table will also have a search field associated with it which can be used to get to a particular entity directly. As seen in the snapshot, there is an additional tab for Analysis. This panel will contain the different graphs which will be used to compare any two components. This part is the most important one which is the crux of the application since it is used for the prediction and analysis of diseases.
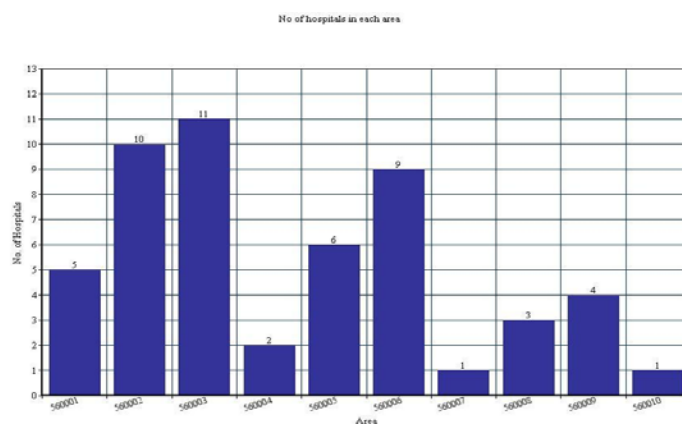
No of hospitals in each area

Fig 2: Comparison of different areas – Number of hospitals.

As an example, we have shown a bar chart above. The x-axis represents the pin codes or area in a particular region and the y-axis shows the number of hospitals in that area. This

gives an insight into the fact that the areas with pin code 560004, 560007 and 560010 lack proper medical support. Using charts instead of numbers gives a clear picture of the current scenario.

A number of other charts and graphs can be generated from the data which will be held in the tables. As another example we have shown a pie chart in the next figure. The chart shows the number of people affected by different epidemic diseases in a certain area. The area can be taken as an input. Also, the chart can show the value corresponding to a certain interval of time.

The different colors in the chart represent different epidemic diseases. The area which is cut out represents measles. The charts can be dynamically made according to the values input by the user. Fig 4 shows the other details.
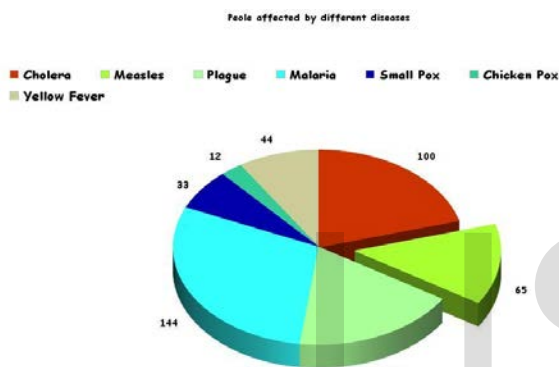

Fig 4: No. of people affected by a particular epidemic disease.

5 MIDDLE LAYER – APPLICATION SERVER

The middle layer consists of the application server. This is again a JavaScript file with some extra functionality. The file type is .xsjs and is generally used with SAP HANA in-memory database applications. It is also known as JavaScript Serer.

This file receives the commands from the front end JavaScript files and forwards the request to the database. It acts as one point of contact for all the JS files. All procedures and CRUD operations are defined in this file.

Two additional files are required for the application to work properly, these are .xsaccess and .xsapp. The exact purpose of these files is still not clear but they are necessary since we are using SAP UI5 Application Development environment. The same application can also be modified at the front end and used for mobile applications. The syntax used by XSJS files is similar to the JDBC/ODBC[6] environment. A code snapshot of the XSJS file is also shown below in figure 5 for understanding.

```javascript
function downloadExcel() {
    var body = '';

    try {
        var query = 'SELECT TOP 25000 \"PurchaseOr
                  + 'FROM \"sap.hana.democontent.epm
        $.trace.debug(query);
        var conn = $.db.getConnection();
        var pstmt = conn.prepareStatement(query);
        var rs = pstmt.executeQuery();
```

Fig 5: A code snapshot from the JavaScript Server file.

The file architecture is such that the view-controller JavaScript files are kept in a folder which manages only the client side and the server side XSJS files are kept separately in a folder responsible for server side application.

6 APPLICATIONS

6.1 Epidemic Prediction

The first and foremost application which is actually the motto behind this web tool is that it should help in predicting the dangerous epidemic diseases. The best way to predict the future is by having a thorough study of the past. We process the data into information in such a way that it gives a proper picture about the current scenario. The same information is

shown by the help of charts and graphs so that it can assist in predicting the future trends.

6.2 A Search Tool

If the tool is public and available to the users, they can use filters and find details about a particular hospital, doctor, disease or patient. The filters will help in reducing the huge ocean of data and the search can be based on a particular city or even an area code.

6.3 Chemists and Druggists

If the tool is public and available to the users, they can use filters and find details about a particular hospital, doctor, disease or patient. The filters will help in reducing the huge ocean of data and the search can be based on a particular city or even an area code.

6.4 Other Uses

Vivid details about a hospital, like which specialist doctor the hospital has and which one it lacks can be seen from the analysis tab of the tool. It can be checked which area particularly suffers from an epidemic and then preventive measures can be taken to stop the influence. Information about unheard diseases, its symptoms, its causes can be studied upon before it's too late.

7 CONCLUSION

The paper provides a way to process the data into some sort of meaningful information which could be used to help people who have medical grievances. The information can be used to predict and analyze what sorts of diseases are common in a particular area. This does not provide a 100% guarantee about the prediction but it does tell a lot about what is expected to happen in near future. The implementation of our application is based on only a particular area which can be extended to any scale whatsoever.

If implemented properly, this tool will be a great help for people from rural areas, where they lack the knowledge and skills since our application uses a very basic user interface.

REFERENCES

[1] Professor Constantine E. Passaris, University of New Brunswick (1975), Canada, worked for a lot of publications related to Finance and Economy.

[2] Carl Gould and Zhendong Su, "JDBC Checker: A Static Analysis Toll for SQL/JDBC Applications", International Conference on Software Engineering, 2004.

[3] Maria JoseSuarez-Cabal and Javier Tuya, University of Oviedo, Spain, "Using an SQL Coverage Measurement for Testing Database Applications", Foundations of Software Engineering, 2004.

[4] MySQL 5.7 Documentation: MySQL Reference Manual, http://dev.mysql.com/doc/

[5] JavaScript and HTML DOM Reference – W3Schools, http://www.w3schools.com/jsref/

[6] J. Smith, "Basic JDBC Connectivity for Sybase Databases", Unpublished, March 2012.

[7] Documentation on SAP HANA (High-Performance Analytic Appliance)

[8] Macario Polo, Mario Piattini and Francisco Ruiz, University of Castilla-La Mancha, Spain, "Reflective CRUD: Create, Read, Update, & Delete", Unpublished 2001.

[9] Simon Holm Jenser and Anders Moller, Aarhus University Denmark and Peter Thiemann, University of Freiburg, Germany, "Type Analysis for javaScript", June 2002.

[10] SAPUI5 Documentation, Reference and Demokit, https://openui5.hana.ondemand.com