

Experimental Information Management System with Creation of Questionnaire Using NoSQL Databases

Blanca Alicia Rico Jiménez¹, Jose Daniel Perez Ramirez²,
Fernando Hernandez Molina², Laura Ivonne Garay Jiménez³

¹ Instituto Politécnico Nacional, UPIITA, Computer science, CDMX,
Mexico

² Instituto Politécnico Nacional, UPIITA. Telematics, CDMX,
Mexico

³ Instituto Politécnico Nacional, UPIITA, SEPI, CDMX,
Mexico

bricoj@ipn.mx, danprjs@gmail.com, fdohm.1994@gmail.com, lgaray@ipn.mx

Abstract. This article presents the development of a computer system that through a Web application allows to manage information generated in a laboratory of signal acquisition and processing, where research studies were carried out monitoring electrical measurements. The on line management of the information allows users to share and access from different places. One of the most critical points in the management of this information is that the experimental protocols must be stored in a single way, thus a relational data base was proposed. On the other hand, the generation of different questionnaires associated to a protocol implies that database structure should consider different form and sizes so the use of a NoSQL database is proposed thus ensuring the system to upgrade and be adaptable to several kind of studies.

Keywords: NoSQL data base, structured data bases, clinical studies, dynamic questionnaires.

1 Introduction

The increase of information available from clinical studies, biomedical systems and evaluations, offers the opportunity to the clinical area personnel, to the investigators and to the students to carry out integral and complex studies in search of information that allows to prevent, treat and monitor diseases in a more efficient way. The obtained data from experimental protocols, where the data in question is generated by different people in different places, like hospitals, experimental laboratories, etcetera, requires of tools that can help in the study and processing of biological signals in an ubiquitous way.

This project is intended to provide an accessible solution through a system that manage information such as, the conditions before and during registry as well as

signal conditioning characteristics that determine the interpretation of those signals. This system is a tool that gives freedom to the specialist to create questionnaires for his/her specific area of research, creating his/her own questions and storing the answers associated to the patient for future analysis, once or several times.

2 Background

The great volumes of *Big Data* represent a challenge and an opportunity for the innovators and all those that worry about the information management. In the area of the health, the adequate management of *Big Data*, substantially increases the possibility of obtaining more effective knowledge on the sufferings and related symptoms on a great scale. Modelling and inferring on them are conditioned by the integrity and completeness of the information [2]. As an answer to the need of managing huge volumes of information appears the NoSQL Database System, term coined at the end of the 90s and which includes all the structured storage technologies which do not fulfil relational schema [4]. The term NoSQL has evolved not to stop using the SQL language, but to refer to systems which are no traditional SGBD (Data Bases Management Systems), and those which do not impose a data structure in the form of tables and the relationship among them, if not to provide a much more flexible scheme [6].

3 Description of the System

The present article shows the design and implementation of a biological signal storage system and the dynamic creation of questionnaires, the Protocol Management System (SAP) allows the management of clinical studies through a Web application. The solution was to integrate different technologies that allowed to obtain a system for storing the clinical studies and their protocols. The first part contemplates the gathering of the required general information from the person in charge of the study and the information about signal conditioning for its interpretation such as sampling frequency, type, number and duration of the records of the study, and solve it with a relational database. The second part, where the subject or patient will be interrogated one or several times about parameters related to his/her health condition. All of them have to be specific to the study. For example: duration of fasting, if he is taking drugs that could affect the signal response, etc.

Each question is defined by the author of the study and the software let him to choose the type of answer. This entails a problem in the design with relational databases. Reason why it was proposed to generate dynamically questionnaires which are store in a NoSQL Database. The person in charge, “responsible”, of the study will be able to generate and to select questions according to his/her knowledge and/or to the needs of the new study and then he creates the questionnaire. It was considered that the information should be analyzed by the experts in the area of study so the system considers the role named “physician”, who can consult and download the information of the studies through remote access and in order to safeguard the integrity of the system and to give maintenance to the same, the “administrator”

figure was created. The general structure appears in Figure 1 and in the following sections their elements will be discussed.

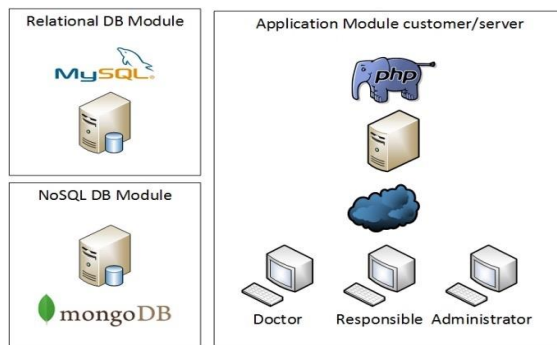


Fig. 1. Architecture of the SAP system.

3.1 Module 1: Relational Database

This model must guarantee that the data are represented in a unique and structured way, conformed by rows and columns [9], fulfilling the characteristics of atomicity and integrity of the data of the clinical protocols and of measurements, information of the patient, the responsible of the study and of the archives containing the signal records, because the information contained in the system must have a defined structure for example: archives txt and data files, that contain the relevant clinical information to the study, technical conditions of registration, etc., the administration was proposed through an URL storage and it was also implemented in the SGBD MySQL. The relational diagram of the database show in Figure. 2.

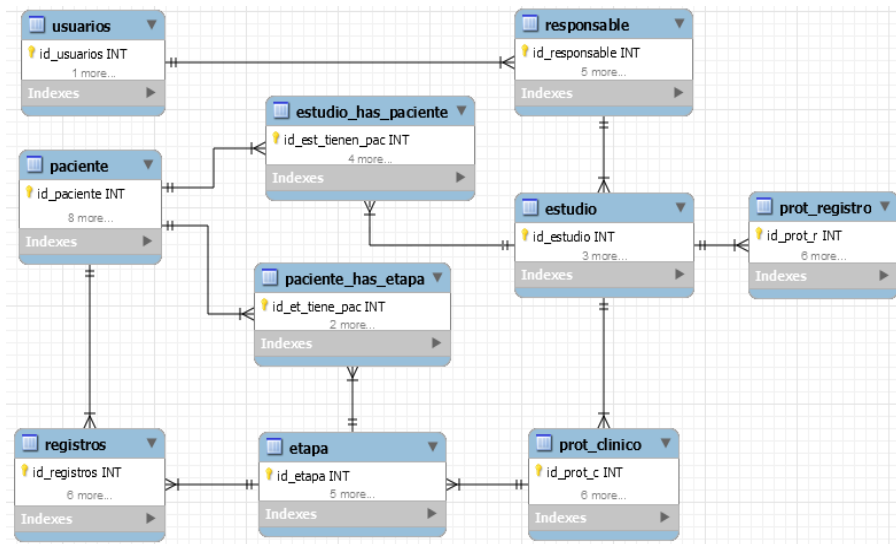


Fig. 2. Relational model database (diagram in Spanish).

3.2 Module 2: NoSQL Database (Not only SQL)

The application of the NoSQL Databases has experienced an important growth lately. The great flexibility which they offer and the possibilities of the optimisation in their designs according to the problem to be solve turn them into an attractive variant to consider to the management of information application developers at large scale [4]. For this system it was necessary to count on a tool that allows generating questionnaires for patients according to the clinical study and the specific protocol that is being carried out on him/her. This technical requirement conditioned the design and development of the system, so it required the creation of questionnaires with dynamic storage.

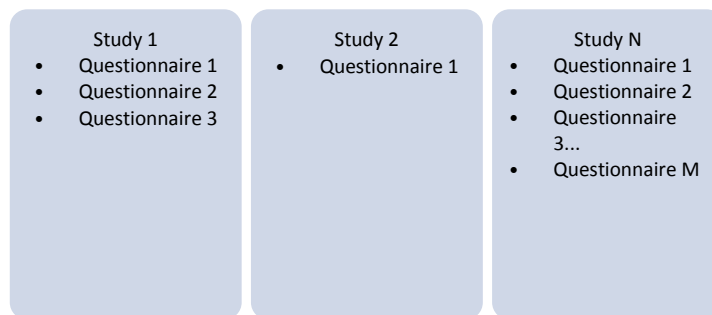


Fig. 3. Design of dynamic questionnaires for M patient in N where each questionnaire can have R questions.

As it is observed in Figure. 3, the questionnaires have different contents since each study can have N questionnaires, each questionnaire R questions that can be answered with an open answer as well as selected into multiple option, and these as well can be answered by M participants in the study. This technical requirement conditioned the design and development of the system, so it required the creation of questionnaires with dynamic storage. It was decided to use the NoSQL Databases oriented to documents, specifically MongoDB, using the category of key – value storage, in them a unique key is assigned (primary key) to a value that is typically an arbitrary chain. The operation of finding the associated value to a key is called “lookup” (indexing), and the relationship between a key and its value is called correlation (binding) [3].

3.3 Module 3: Web Interface

For this last module, the “backend” of the system was implemented in language PHP, whereas for “frontend” the HTML5, CSS3, Javascript and the jQuery bookstore languages were used. It was implemented in a HTTP web server of Apache open code installed on the operating system Ubuntu 14.04. Integration in this way allows to have a Web surrounding with the purpose of facilitating access to the information from the user’s place just with an Internet line and a navigating Web.

As the system will be used in a ubiquitous way by users with different types of needs, three types of roles were generated: administrator, responsible and physician.

In Table 1, the permissions for each one are described according to their needs. An example of the Web application interfaces is presented in Figure. 4. In this view, the user can edit the already registered studies, such as the stages, patient information and then he/she can add the records associated to that patient. All this information is stored in the SQL database.

Table 1. Users and permissions in the system.

Permission	Administrator	Responsible	Physician
To register new users	X		
To create studies		X	
To eliminate studies		X	
To see the existing studies		X	X
To unload archives of the study		X	X
To add registry archives		X	
To unload registry archives		X	X
To create questionnaire		X	
To respond questionnaire		X	

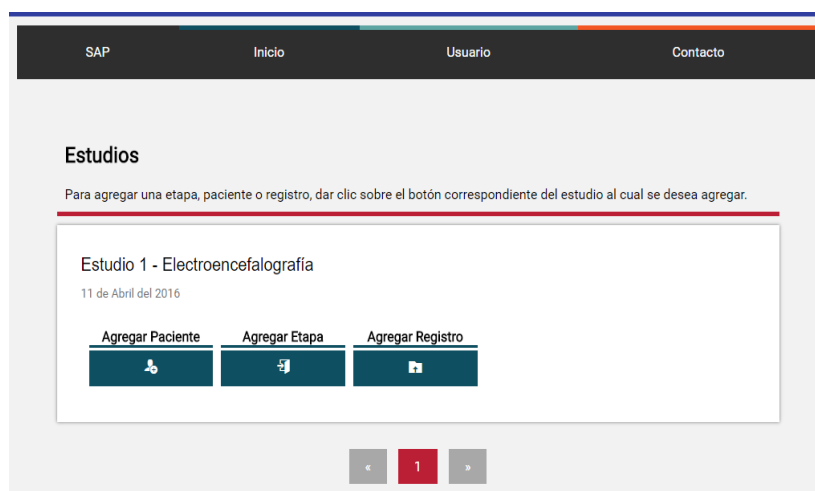


Fig. 4. Study editing interface (screenshot in Spanish).

In Figure. 5, it is shown the interface that displays all relevant information previously included as well as the dynamic questionnaires management module, which performs the administration of them consulting the NoSQL database.

4 Dynamic Questionnaires

Once the person in charge defines the characteristics of the study, he or she generates the recording procedure named recording protocol, the clinical protocol and the questionnaire(s) to be applied. This questionnaire can be answered whenever it is required meanwhile a recording session exists, so the design of the database must consider the answers associated to multiple applications of the questionnaires according to the number of proposed records.

The dynamism in the definition of the structure for each responsible user is obtained because this NoSQL Database is oriented to documents that are located with a particular key/value, each value can be a whole number, a chain of characters, an adjustment of values or even another document, these characteristics allowed to obtain the required dynamics in questionnaires in a simple and agile way [4].

Fig. 5. Study information view (screenshot in Spanish).

For the implementation MongoDB was used, [8] its name comes from “*humongous*”, gigantic and is a data base of open code based on JSON documents [5]. These documents have dynamic schemes, instead of the static tables of the SQL language, it has a format for data interchange similar to XML but with a simpler structure [1]. The document is the unit of storage of a JSON object.

In MongoDB, documents JSON are grouped in a collection. A collection is the equivalent to a DBMS relation. The JSON documents from same collection not necessarily become attached to the same scheme or configuration [1].

```

Name: Questionnaire 1
Study: 1
Question 1: What is your occupation?
Answer: _____
Question 2: How many hours does you use the computer continuously?
a) Less than a b) One c) More than two d) Three or more
    
```

Fig. 6. Example of a questionnaire contained in the document type JSON.

The advantage obtained by choosing a NoSQL Database oriented to documents like MongoDB is that each questionnaire can be developed freely, without depending on the format of questionnaires that already exist within the collection of the database. In Figure. 6, a proposed questionnaire is exemplified and in Figure. 7, the modified structure for the JSON files that conform the data in MongoDB is presented.

Cuestionario 1	Cuestionario 2
<pre> "id_cuestionario" : 1, "pregunta1" : { "oracion" : ¿...?", "tipo" : "abierta" }, "pregunta2" : { "oracion" : ¿...?", "tipo" : "opcion_multiple" } </pre>	<pre> "id_cuestionario" : 2, "pregunta1" : { "oracion" : ¿...?", "tipo" : "abierta" }, "pregunta2" : { "oracion" : ¿...?", "tipo" : "opcion_multiple" } </pre>

Fig. 7. Json files belonging to a collection in the used NoSQL database (questionnaire in Spanish).

The number and type of questions could be different for each study but all of them will be grouped in a collection of files easily accessed.

5 Combination of Relational Database and NoSQL

The general architecture of the system is hybrid because it shows two databases with different structures, the relational and the oriented to NoSQL documents. When managing these two types of data bases in the same system, it was necessary to make a link between the identifiers of each study in the relational database with the respective questionnaires in the NoSQL Database. This was achieved using specific PHP sentences in the “*backend*” of the system.

The methodology used to link the identifiers between the two data bases is summarized in the following steps:

- 1) **Add a study**, implies to add all the information necessary to be kept in the relational database, thus obtaining this way a sole identifier (ID) or primary key for each stored study.

2) **Create a questionnaire**, through the Web application, the person in charge can generate questionnaires with the required characteristics, then they are stored in the NoSQL Database.

3) **Request of the identifier of the study**, subsequent to the creation of the questionnaire, it is required to assign it to a pre-existing clinical study in the system. For that an identifier of the said study is requested through consulting the SQL relational data system.

4) **Carry out a connection between the identifiers**, the system sends the identifier of the clinical study along with the rest of the other necessary data for the creation of a questionnaire to the NoSQL Database, then storing the identifier in a field of the questionnaire.

With the previous methodology it is achieved that the system is coherent and maintains the consistencies that the system demands between the clinical studies and their questionnaires.

4 Evaluation and Results

The application was lodged in the Web server and it can be consulted using any Web Navigator in the www.lpssepi.upiita.ipn.mx page which it is administrated by the laboratory of instrumentation and processing of signal (LISP). The evaluations were made with a sample of: 1 user as administrator, 5 users as responsible role and 5 users as the physician role. All of them used SAP indistinctly in or outside the Institute during a week for generating and manage their studies. The participants recorded physiological signals logs and associated them to their clinical studies for further analysis. They used to generate non structured databases with general programming software such as Matlab®, Labview®, Excel® and they themselves are not Telematics professionals or similar, within an age range between 20 to 46, being the average age 39.

The procedure for the evaluation of the software entailed 1) A talk explaining the purpose and operation of the software, 2) Delivery of the User's Manual, 3) Delivery of instructions and questionnaire evaluation form, 4) Allocation of tasks according to Table 2.

Table 2. Activities made for the evaluation of software.

Administrator	<ol style="list-style-type: none"> 1. You will have to register to 5 users in the system. 2. To consult the information of one of them. 3. It eliminates a user.
Person in charge	<ol style="list-style-type: none"> 1. It generates a study 2. To create questionnaire 3. To add registry archives
Physician	<ol style="list-style-type: none"> 1. To consult a clinical study with its registries

The criteria considered for the evaluation of software were: functionality and easy to use, technical support, performance, safety, integrity, availability, all of which were evaluated in a rank from 1 to 4, where the headings valued less than 3 should be

improved and all the headings with a value greater than 3, the user considered them as acceptable. 6 out of 10 users evaluated the performance of the program above 3. The results by characteristics are presented in Table 3.

Table 3. Results of the feedback of the evaluated characteristics of SAP.

Characteristics	To improve [Evaluation < 3]	Good performance [Evaluation >= 3]
Functionality and ease of use	2.66	
Support	2.35	
Performance		3.42
Safety		3.25
Availability		3.1

7 Discussion

Although specialized programs in generating software for an application of recording and generation of signals data bases already exist, like LABVIEW® or MATLAB® and it is a common practice that each person in charge of a study of signals, makes his/her own system of registry and saves it with the computing programs available to them and they do it using their own personal computers, it is unlikely that the information can be shared because they do not follow a saving standard. On the other hand, centers such as Physionet that manage a large number of recordings have public web services in order to share this information but it is necessary to cover certain specific requirements.

The proposed system used the two types of data bases in a transparent way to the end user through a Web application and in 100% of the cases where it was used, it achieved to generate the storage of clinical studies according to the set out requirements, the management of the data bases and it maintained its integrity. On the other hand in the results of the evaluation of the SAP system web application, it was observed that it fulfilled the expectations of 66% of the users, and further improvements were proposed. It is necessary that: 1) the administrator have access to all the information of the stored studies, 2) to extend the permissions in the information for the physicians, 3) due to the diversity in the user profiles, the documentary technical support of the system should be improved and reorganized, as well as to provide user's quick guide according to the role.

8 Conclusions

This proposal of application allowed to unify and standardize the type of relevant information in a follow up of the information management in an experimental laboratory where it is known that the size of the database will increase with time. The results demonstrated that the group, which evaluated the system, considered that SAP fulfils the expectations of managing the information on line and by different type of

users, not having to worry about the size and content of the archive generated by the study and about some particular characteristics such as the generation and management of the questions required by each one. In the technological development, it can be said that the use of NoSQL Databases to create health management systems is a good alternative, if the need is to cover the dynamic growth of the information without a defined structure of the data. Further work in web functionality and support is required but this type of hybrid systems has allowed to solve some of the dilemmas in managing databases in Big Data Systems.

Acknowledgments. This work was supported by Instituto Politécnico Nacional (IPN, México) with grant SIP20161730. The authors wish to thank IPN for the BEIFI scholarship granted to the student F. Hernández.

References

1. Arboleda, F. J., Rendón, J. E., Vásquez, R.: Una comparación de rendimiento entre Oracle y MongoDB (Comparing efficiency between Oracle and MongoDB). *Ciencia e Ingeniería Neogranadina (Neogranadine Science and Engineering)*, 26(1), pp. 109–129 (2016)
2. Grover, P., Johari, R.: Review of Big Data Tools for HealthCare System. In: 6th International Conference - Cloud System and Big Data Engineering (Confluence International Conference (2016)
3. Han, J., Haihong, E., Le, G., Du, J.: Survey on NoSQL database. In: *Pervasive Computing and Applications (ICPCA)*, 2011 6th International Conference on, Port Elizabeth, pp. 363–366 (2011)
4. Ayaida, M., Beaujet, B., Valentin, F., Bradmetz, F., Picco, G., Rojbi, B.: Sensor vehicular platform for road maintenance. In: *International Wireless Communications and Mobile Computing Conference (IWCMC)*, Paphos, Cyprus, pp. 535–540 (2016)
5. Rocha, V.: Sharing Product Information for Supporting Collaborative Product Development. Dept. Production and Systems, School of Engineering, University of Minho, Braga, Portugal, October (2016)
6. Martín, A., Chávez, S., Rodríguez, N., Valenzuela, A., Murazzo, M.: Bases de Datos NoSql en Cloud Computing (NoSQL Databases in Cloud Computing). En: *VX Workshop de Investigadores en Ciencias de la Computación, (Computing Science Researcher's Workshop)*, Paranda–Entre Rios (2013)
7. Martínez, A.: Herramienta de Evaluación Automática mediante Bases de Datos NoSQL aplicando Métrica v3 (Automatic evaluation using NoSQL Databases applying v3 metrics). Tesis, Departamento de Ingeniería Telemática Escuela Técnica Superior de Ingeniería Universidad de Sevilla, Sevilla, Department of Telematics Engineering, Technical School, University of Sevilla (2014)
8. MongoDB Inc.: Introduction to MongoDB. <http://docs.mongodb.org/manual/core/introduction>
9. Reinoso, E., Maldonado, C., Muñoz, R., Damiano, L., Abrutsky, M.: Bases de Datos (Databases). Alfaomega, 1ª ed, Buenos Aires, Argentina, pp. 27 (2012)