J. Health Inform. 2011 *Outubro-Dezembro; 3(4):137-45* **Original Article**

JOURNAL OF HEALTH INFORMATICS

SBIM: An Internet Manager for Sharing Biological and Medical Images

SBIM: Um Gerenciador na Internet para Compartilhamento de Imagens Médicas e Biológicas

SBIM: Un Administrador en Internet para Compartirlas Imágenes Médica y Biológica

Luiz Antônio Pereira Neves[1], Gustavo Molitor Porcides[2], Gilson Antonio Giraldi[3]

ABSTRACT

Keywords: Database Management Systems; Bioinformatics; Online systems

Objective: The available image managers basically include facilities to store, discuss and share detailed images from all over the web. In fact, a literature analysis shows that the available systems are limited for image analysis. So, the main goal of this work is to present an image manager for medical and biological databases that allows researchers to share and manipulate images in an environment that integrates data and analysis facilities. **Methods:** In order to achieve the expected goal we firstly perform a literature review to define basic services, the data modeling of the system and select the development platform: the programming languages PHP and Javascript, as well as the Database Management System PostgreSQL 8.4.7 The SBIM has been tested based on the methodologies developed specifically for web applications validation. It was assembled a group of 40 individuals to evaluate items like performance, usability and graphical interfaces. **Results:** The SBIM system provides efficient tools to share and analyze images stored in its databases. **Conclusion:** SBIM is an efficient image manager system to store, discuss and analyze medical and biological images over the web.

RESUMO

Descritores: Sistemas de Gerenciamento de Base de Dados; Bioinformática; Sistemas em linha

Objetivo: Os sistemas para gerenciamento e compartilhamento de imagens médicas disponíveis oferecem basicamente uma plataforma para armazenamento e troca de dados e informações via Web. Uma análise da literatura mostra que os recursos para processamento e análise de imagens são muito limitados. Assim, com o objetivo de oferecer uma alternativa aos sistemas disponíveis na Web que integre compartilhamento e manipulação de imagens médicas, está sendo desenvolvido o Shared Biological Image Manager (SBIM). **Métodos**: Foi feita uma análise da literatura para identificação dos serviços que o sistema deve oferecer, bem como a escolha de uma plataforma que facilitasse o desenvolvimento: linguagens de programação PHP e Javascript com o sistema gerenciador de bancos de dados PostgreSQL. O sistema foi testado usando metodologias específicas para avaliação de Aplicações Web. Foi escolhido um grupo de 40 voluntários para testar itens como interface gráfica, desempenho e usabilidade. **Resultados**: O sistema fornece ferramentas eficientes para gerenciamento e compartilhamento de imagens, troca de informações entre os usuários, bem como uma plataforma para análise das imagens. **Conclusão:** Foram obtidos resultados satisfatórios que são indicadores da qualidade do sistema SBIM para armazenamento e manipulação de imagens.

RESUMEN

Descriptores: Sistemas de Administración de Bases de Datos; Bioinformática; Sistemas en línea

El sistema de administración y compartimento de Imágenes médicas disponibles ofrecen básicamente una plataforma para almacenamiento y cambia de datos y informaciones vía web. Un análisis de literatura muestra que el recursos para el tratamiento y el análisis de imágenes son muchos limitada. Así con el objetivo de ofrecer una alternativa a los sistemas disponibles en la Web que se integra compartir y manipulación de imágenes medico, se está desarrollando la Shared Biological Image Manager (SBIM). **Métodos:** Se hizo un análisis de literatura para la identificación de servicios que el sistema debe proporcionar, así como la elección de una plataforma que facilita el desarrollo: lenguaje de programación PHP y Javascript con el bases de datos de PostgreSQL. El sistema fue probado con metodologias específicas para aplicaciones Web. Fue elegido un grupo de 40 voluntarios para poner a prueba elementos como la interfaz gráfica de usuario, rendimiento y facilidad de uso. **Resultados:** El sistema proporciona herramientas eficaces para gestionar y compartir imágenes, intercambio de información entre los usuarios, así como una plataforma para el análisis de imágenes. Conclusión: Se obtuvieron resultados satisfactorios que son indicativos de la SBIM sistema de calidad para almacenar y manipular imágenes.

[1] *Professor Titular da Universidade Federal do Paraná - UFPR, Curitiba (PR), Brasil.*
[2] *Curso Superior em Tecnologia em Análise e Desenvolvimento de Sistemas 2009-2011. Bolsista de PIBIC/UFPR, Universidade Federal do Paraná - UFPR, Curitiba (PR), Brasil.*
[3] *Laboratório Nacional da Computação Científica Pesquisador - LNCC. Coordenador do Programa de Pós-Graduação de Mestrado e Doutorado em Modelagem Computacional do LNCC, Petrópolis (RJ), Brasil.*

Autor Correspondente: **Luiz Antônio Pereira Neves**
e-mail: **lapneves@gmail.com**

Artigo recebido: **26/06/2011**
Aprovado: **13/10/2011**

INTRODUCTION

One of the main contributions of the World Wide Web to society is the possibility to share text documents, images and videos, making them accessible from anywhere, as long as there is an Internet connection. The Internet offers a worldwide network infrastructure that connects nearly two billion people and transports roughly 10 billion gigabytes of data a month[1].

In the last decades, the spread of web applications has improved the Internet flexibility in accessing computing resources such as processing and storage capacity. These applications are usually formed by three tiers. For Bouchiha et al[2], the first tier is the Web browser, used to visualize Web pages. In the middle tier are programming languages used to generate dynamic content, like PHP (Hyperterxt Preprocessor) and ASP (Active Server Pages). The last tier is the database that will store useful information used by the Website. The Web browser, in the client side, sends requests to the second tier that then will retrieve and insert data into the database.

Image Managers, like Morphbank[3], Biology Image Library[4] and JCB Data Viewer[5] to name a few, compose a class of Web applications. Basically, these computer systems include facilities to store, discuss and share detailed images from all over the world which improves scientist collaboration and education activities. In this work, we focus on image managers for medical and biological image databases because there is a lack of Web applications in these areas that allow researchers to share and manipulate images in an environment that integrates data and analysis facilities.

The field of medical imaging analysis had a fast development in the past decades. It is an important tool for diagnosis and planning procedures in almost all areas in the medicine. Besides, and more important for this work, application of computational methods to study complex physiological systems need a cooperative work between research groups which may be geographically and institutionally dispersed. In Brazil, such need motivates the development of the INCT–MACC (National Institute of Science and Technology in Scientific Computing Aided Medicine)[6], a virtual organization working in the thematic of medicine assisted by scientific computing. In the context of the INCT-MACC, the medical imaging analysis has specific requirements that involve algorithms for segmentation, feature extraction, geometric modeling, tracking, registration, classification and visualization. Bioinformatic images, like those from the Brazilian Genoma Project, may also be stored and manipulated using the proposed system. Therefore, there are plenty of projects that may use this system.

Hence, a fundamental point which emerges from the necessities of these projects is that we should consider image managers from an integrated global view of data and processing programs, putting together tools for image storage and manipulation, as well as offering functional ways to share information. In order to fulfill this requirement, the Shared Biological Image Manager (SBIM) has been developed. The system allows users to have great control over the content, giving them the possibility to group images by category, search images by size, format, compression and color channel. It also has a message manager, to allow the communication among researchers.

Therefore, the main relevance of this project is to create an integrated Web environment where researchers can keep and manipulate images using tools and programs that will improve the image analysis in a fast and reliable Web application.

RELATED WORKS

In this section we survey image manager systems trying to identify basic requirements and challenges related to user interface, search engines, such as Content-Based Image Retrieval (CBIR) algorithms and processing resources.

In a wide sense, image managers are part of the kernel of any image-based information system. Therefore, Web-based systems for image collections and visualization, Web applications for training and education, CBIR and hospital information systems (HIS) incorporate some kind of image manager in their engines.

For instance, the Biology Image Library[4] is basically an online collection of images, movies, illustrations and animations from biology and biomedicine areas including Histology, Pathology, Microbiology, Molecular and Cellular Biology. The library aims to allow users to look for visually striking and scientifically reliable material without tools for image analysis and processing. The JCB Data Viewer[5] is a web-based tool for visualization of multi-dimensional image data files. The system has simple tools for image analysis (zoon, pan, others), but its main goal is to handle supplement material associated with manuscripts submitted to the Journal of Cell Biology (JCB). In the same class of systems, we find the Morphbank[3] which is a continuously growing image database focused on increasing knowledge of scientists and students about biodiversity. The images deposited in the Morphbank repositories include data from comparative anatomy, morphological phylogenetics, taxonomy among others.

Nowadays, the traditional medical teaching and training has been complemented with web based education systems. In Guliato et al[7] it is described a Web teaching system for training the interpretation of mammograms. In this system, users can obtain texts about specific subjects, access bibliographic references and retrieve cases from a database. Also, this system has image processing tools for drawing markings on the image to identify the contour of the breast or the lession. Afterwards, the contours are submitted to the system and evaluated. Wangenheim et al[8] proposes a methodology to interconnect DICOM (DIGITAL IMAGING AND COMMUNICATIONS IN MEDICINE) workstations, allowing real-time meetings over the Internet for discussion about the images. The main goal of that work is to present a collaborative system that enables recording, sharing, and reenacting teleradiological collaborative discussions as well as diagnosis sessions for documentation and teaching purposes. The whole process to create a collaborative section involves its initialization, data exchange and closing procedures. All these steps are implemented through the CycAppDCM[8], an Application Layer Protocol that is layered right on top of the

Transmission Control Protocol/Internet Protocol (TCP/IP). On the other paper, Pires et al.[9] present a Mammographic image manager that is very useful for image diagnostic training by allowing simulated appraisals.

CBIR is a class of techniques which uses visual contents (features) to search images from a database following a query image given by the users. Therefore, when designing a CBIR system we must be careful about interfaces to driven the query image definition, utilization of feedback, feature space specification, feature matching and output presentation. Therefore, the application of CBIR for computer-aided diagnosis involves aspects from user interfaces, system design and image processing techniques. Such complexity is in the heart of the problem described in Welter et al[10]. This work pointed out that despite of their potential, CBIR techniques have not yet been established in clinical practice due to the lack of an unified data concept integrating CBIR with picture archiving and communication systems (PACS). Therefore, authors suggest that CBIR systems applied to computer-aided diagnosis should integrate their results in a DICOM reporting document. On the other hand, there are many image processing issues to improve CBIR efficiency. For instance[11], presents a CBIR system with focus on classification of breast tissue density. Texture features are defined using singular value decomposition (SVD) and a support vector machine (SVM) is applied for image retrieval.

The hospital is itself a system in which human beings and machines carry out specific actions following established rules to achieve an integrated care. Therefore, HIS must provide communication and processing information to allow the suitable degree interoperability. In this context, medical image storage, processing and mining become central issues not only due to the huge data volume to be considered but also because image analysis is fundamental for diagnosis and treatment procedures. For instance, Azevedo-Marques et al.[12-13] describe a system that integrates the radiology information system (RIS) and the PACS of the Radiodiagnosis Service of the *Hospital das Clínicas de Ribeirão Preto (Brazil)*. The proposed solution has a fast distribution system and friendly interface because it uses Internet technologies, such as HTML (Hypertext Markup Language) and ASP.

Basically, the mentioned systems include resources to store, discuss and share detailed images from all over the web, training and education as well as search engines in an image database. We observe a lack of Web applications in medical and biological image that allow researchers to share and manipulate images in an environment that integrates data and analysis facilities. This is the main motivation for our work.

However, before presenting our system, we must identify from the above survey common requirements for image managers applications. We observe that, in general, these applications systems needs to provide the following services:

- Image storage: Provide a standarized way to store images, allowing easy access to them.

- Image sharing: Allow researchers to share images in the Internet using a Web Browser.

- Algorithms for image analysis: Implement tools to improve the image analysis, like zoom and marking tools.

- Algorithms for image processing: Implement algorithms for digital image processing, aiming to integrate several tools in a single system.

- Communication: Provide ways for researchers to communicate without the need to use tools from outside the proposed system.

These are basic services that should be provided by SBIM system. The next sections show the way SBIM addresses the corresponding requirements.

METHODOLOGY

In order to achieve the expected results a four step methodology is being used, as shown in Figure 1.

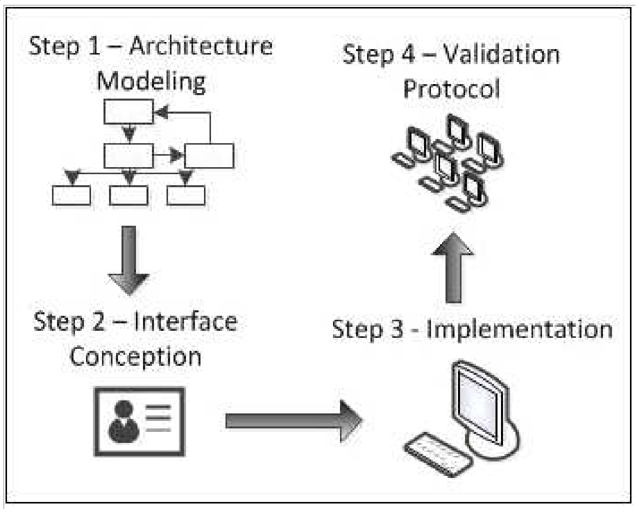

Figure 1 – Used Methodology

The first step consists of analyzing several papers in order to define and implement the data modeling of the system and select the best tools to develop the project. Step 2 defines how the system will handle the images and designs the user interface. In the third step, it is being implemented on a Web environment. During the last stage, it is being tested based on the techniques proposed by Pressman[14], developed specifically for web applications.

Step 1 - Architecture Modeling

The database model consists of 15 tables, based on Porcides et al[15]. The database's ERD (Entity-Relationship Diagram), developed with DBDesigner, is shown in Figure 2.

The main table is called image. In this table all the data about the image, like compression, channel, format, size, width, height, resolution and keywords are stored. Most of the other tables are connected to it

The tables that have information related to the image are used to index these images and, afterwards, users can use this information to retrieve images more selectively.

There is the table called history. It is used to keep track of everything that the users have done. It will store what an user did (upload, delete or modify an image), when he did it, which image he manipulated and how long it took. Also, the system can keep track of when users login or logout. In this way, problems can be solved faster by

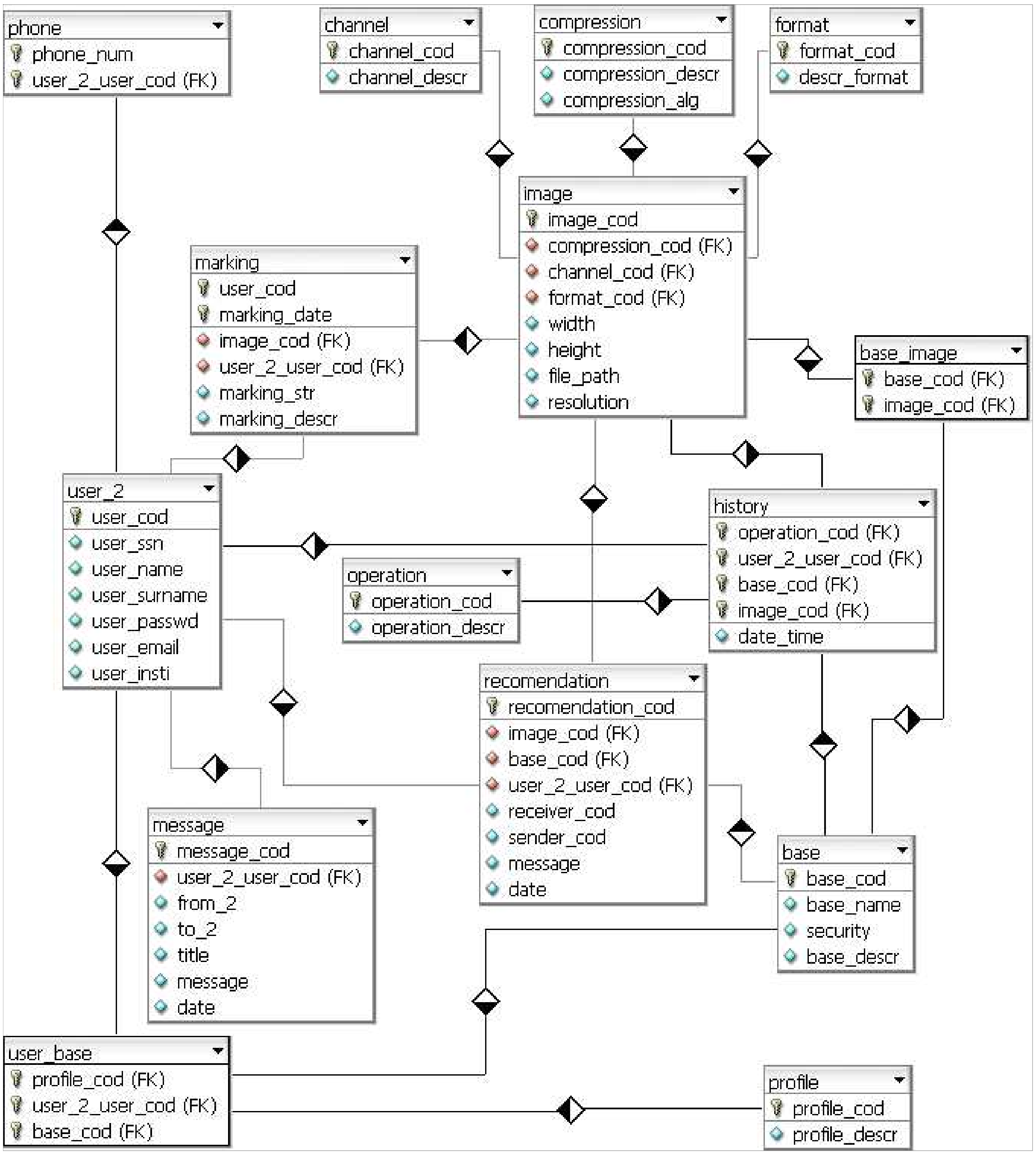

Figure 2 – Entity-Relationship Diagram of SBIM

tracking its possible causes.

Also, this database has tables to store image formats, compression types and channels, as well as information about the users, messages and image markings.

The user's information is stored in user table. This information will be used by the search engine to find them using name, surname, e-mail address or affiliation.

The profile table is used to store the access levels. These access levels are used to determine which image databases an user can access and what he can do on it.

Message and Recommendation tables are used to store the messages sent by the users to others using the SBIM message system.

Therefore, this data model allows the development of a system with good performance to retrieve data from the database and an easy and effective way to create a permission system, with each user having different access levels to several bases.

Step 2 - Interface Development

The main concern is to develop an intuitive and easy to use interface that won't require a long learning time.

The interface is being developed using only two colors. As proposed by Oppermann[16], the first one is white, a light and neutral color that will draw attention to the main content of the website because it contrasts with blue, the other color that composes the interface. This color is applied to buttons, menus and decoration using several hues of blue. So the website doesn't become monotonous and boring.

As shown in Figure 3, the website's interface is divided in four main areas. The database list on the left side displays the name of all the databases registered in the website (Figure 3-1). The area on the right side will display, if the user isn't logged in, the form to login and a button to create a new account. Otherwise it will show the control panel, where the user can access his messages and change his account information (Figure 3-2).

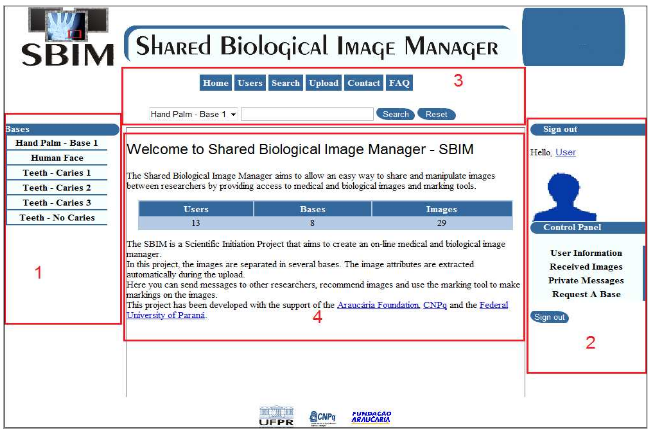

Figure 3 – Main Screen of SBIM

In the upper area it is the advanced search, database list, user list, contact form and the main menu, where users can find the buttons to upload images (Figure 3-3). In the center of the page we can find the content, which may vary from page to page (Figure 3-4). Figure 4 shows the image list of a database.

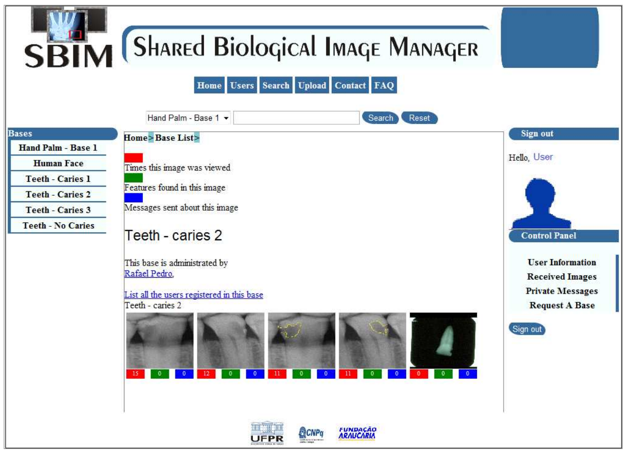

Figure 4 – Image List

Step 3 - Implementation

The system is being implemented using the programming languages PHP and Javascript. The DBMS (Database Management System) PostgreSQL 8.4.7 was chosen because our previous research showed that PostgreSQL has a greater support to the functionalities that we need and it is faster than the others DBMSs in most cases[15]. The system is free and has a good user support via Internet.

The main features of the website are the zoom tool and marking tool. Besides, CBIR functionalities will be implemented soon which will compose another important feature of the system.

The data marking tool, developed with Javascript and HTML5 programming languages, allows users to make markings on images and assign text to them. One interesting feature about this tool is that it doesn't actually change the image at all. All the marking data like text, color and coordinates are stored in the database, separately from the image file. Therefore, the system doesn't need to make a copy of the image each time an user makes a new marking on it, since they don't affect the image. Users can see a marking by just clicking on the corresponding button. With this tool users can save information like regions of interest and a text with their descriptions, features related to a structure in the image, and so on. These data can be shared with other experts using the system. The marking tool is shown in Figure 5.

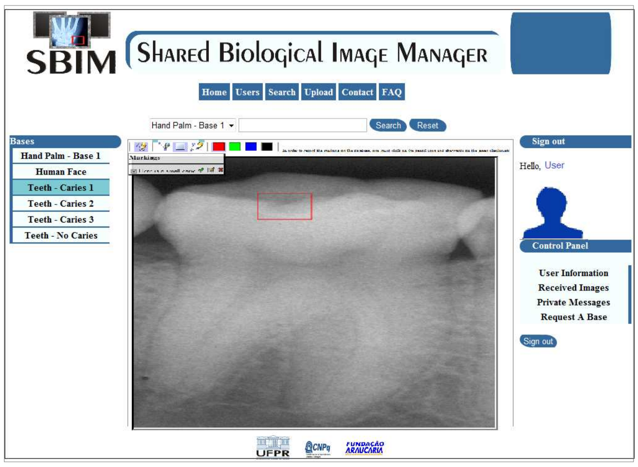

Figure 5 – Marking Tool of SBIM

Another interesting aspect of the web application is the zoom tool. It allows users to zoom in and out images, controlling the size of the area to be zoomed and the zoom rate, just by using the mouse. It has a very high zoom rate, of about 1000%.

This tool simplifies the visualization of small details that, otherwise, could be missed. The zoom tool is shown in Figure 6.

Figure 6 – Zoom Tool of SBIM

Another feature of the SBIM project is the automatic feature extractor. This tool extracts image information automatically, during the upload. The extracted items are width, height, size, compression, format, channel and resolution. This is a real time process that starts as soon as the page to upload the image is loaded. This data is used to retrieve images using the Advanced Search Form, where the user can specify several options. In this search mode the user can specify search criteria, like maximum and minimum width, height and resolution, a specified format or compression.

Also, there is the message system, which works similarly to an e-mail service, but inside the website. These messages may be composed by a text with attached images that are hosted in the SBIM website. These messages will be stored until they are deleted by its owner.

For the security of the website, several access levels were implemented to grant a greater control over the website utilization. There are three access levels with different profiles. The first one is Researcher, which is assigned to users by default. A Researcher can access bases, view images, send messages and upload images. The second level is Base Administrator. A Base Administrator can own a database, having total control over it. He can delete images, invite users to analyze the database, as well as remove them from the database. Also, he can allow users to download images or block the download. To create a database, an user must send a message to one of the website administrators asking for the database creation. If the request is approved, a database will be created with the user who requested it as the administrator. The last level is System Administrator, who has total control over the whole website. He can make all the things that other users do plus create databases and remove users.

These functionalities are being implemented to provide the user with a simple and efficient way to share and manipulate images in the Internet.

Step 4 – Validation Protocol

The validation protocol consists of using several test techniques in order to analyze several aspects of the website, such as performance, interface, usability and compability.

For the content test and usability test, we assembled a group of 40 individuals. These users, researchers in the areas of informatics and medicine, as well as regular Internet users, aided us in these tests by answering a checklist, shown in Figure 7, about the system interface.

The performance test also requires the help from users, since it needs a high amount of data flux. For this test, two groups of five people, all from our research group,aided us by providing images and uploading them to the system.

SBIM - Shared Biological Image Manager
Usability Checklist

This checklist aims to obtain the opinion of the user about the SBIM system. In this checklist, the functionalities, privisibility, ease of use and understanding of the system are analysed.

1 - Ease of use
( )Easy to learn
( )Simple
( )Hard to learn
( )Confusing

2 - Ease of understanding
( )Clear
( )Ambiguous
( )Confusing

3 - Predictability
( )Predictable
( )Usually predictable
( )Usually unpredictable
( )Unpredictable

4 - Functionalities
( )All resources are working
( )Some resources aren't working
Which? ______________
( )All resources aren't working

5 - Comments and suggestions
____
____
____

Figure 7 – Usability Checklist.

The test techniques used were proposed by Pressman[14] and are described below:

- Content Test: The objective of this test is to verify if the static content of the website is concise, precise and conclusive.

- Database Test: This test aims to detect flaws in the way the dynamically generated content is shown to the users.

- Interface Test: In this test the interface and its functionalities are tested to verify how they behave in several Web browsers.

- Usability Test: The objective of this test is to analyze the user perception about the system's features, like interface and functionalities.

- Compatibility Test: This test aims to find out problems on the website when it is used with different combinations of browsers, screen resolutions, operating systems and network resources.

- Component Test: This test is applied to forms and error messages. Its main objective is to verify if the forms are sending the data correctly and the error messages are conclusive.

- Navigation Test: The objective of this test was to confirm the results obtained in the Interface Test and, moreover, validate the search engines.

- Configuration Test: This test was made in the server side and aimed to analyze the server hardware, security and capability.

- Performance Test: This test also focuses on the server. The objective is to test the database management system, storage and network limitations of the server. Also it is used to find out the maximum number of users the server can handle.

RESULTS, TEST STEPS AND EVALUATION

Based on the described tests, we obtained the following results:

Content Test: The groups of users that participate in the Usability Test analyzed the static content and defined it as conclusive and easy to understand. Also, to make it even easier to use we developed a FAQ (Frequently Asked Questions), with answers to the more common questions about the system.

Database Test: During this stage we tested the database using several SQL (Structured Query Language) queries to insert, update and delete data from the database. All the flaws that appeared in the system were fixed. Most of these flaws were related to the search engines, which have the most complex SQL queries.

Interface Test: In this stage, by using the W3C validator[17] and manual tests, all the links and redirect, as well as forms and pop-up windows were validated. This test detected that the zoom tool doesn't work on Opera web browser and has not yet been repaired.

Usability Test: The method used for this test was to use a checklist where a group of users validated features like ease of using and understanding and other functionalities. At the end of the checklist the user should make a suggestion to improve the system.

Figure 8-1 shows that 30% of them considered easy to learn the system functionalities and 55% considered it simple. Only 15% of the users considered it hard to use and nobody considered it confusing. This test aimed to gather information about how user-friendly the interface is and if the users feel comfortable using the website.

In figure 8-2 most users, 62.5%, think the system has a clean interface. On the other hand, 35% think the system is ambiguous and 2.5% considered the system confusing. Using the data from this test, the website was modified to improve the messages that it displays, making it more understandable and clear.

Figure 9-1 illustrates the test about the predictability of the system. This test analyzes if the users think it is hard to know what a functionality, such as a button or link will do. In this test 40% of the users considered the system predictable and 50% think it is usually predictable. Only 7.5% considered it usually unpredictable and 2.5% think it is unpredictable.

Figure 9-2 shows the functionality test, which aim to discover flaws in the main features of the Web application. This test confirmed that the Zoom Tool doesn't work on Opera web browser and, therefore, it is being fixed. In fact, 77.5% of the testers used all the tools without any problem and 22.5% reported that the Zoom Tool is not working inside Opera.

Compatibility Test: The system was tested with Windows XP 32-bits, Windows 7 64-bits and Debian x64 with the browsers Opera, Epiphany, Mozilla Firefox, Ice Weasel, Internet Explorer and Google Chrome. This test verified that the system behaves as expected in all the operating systems and, in most cases, in all the web browsers except for the zoom tool that doesn't works on Opera.

Component Test: During this stage the main objective was to detect errors in the forms. The forms were tested using many combinations of form fields, leaving one or more fields with static values and changing the others. Also, error messages were tested and changed to display detailed and conclusive information about the errors. By analyzing the results of this test we found some inconsistencies that were fixed, mostly related to search engines not returning the expected results.

Navigation Test: This stage used the same method of the Component Test to validate the advanced search engine. It needed a very detailed analysis due to the fact that it contains many search options. The advanced search engine presented some problems due to the various search options. These problems were solved by developing improved SQL queries.

Configuration Test: In this test we found out that the server where the system is hosted is well suited to handle many images and users at the same time without presenting problems.

Performance Test: For the Performance test the website was loaded with 80 images each one with a file size of 2.5MB approximately. Then, groups of users were assembled to validate the interface with the checklist for the Usability Test.

Each group was formed by five users. Each member of the group uploaded the image from different locations, to verify if the distance from the server would affect the performance of the website in the client side. The uploading process took about 45 seconds when five concurrent users send one image of about 2.6MB each one. Also, we used the Apache Benchmark[18] to test the performance.

The results are shown in Table 1. The first column shows the number of concurrent users and the number of simultaneous requests.

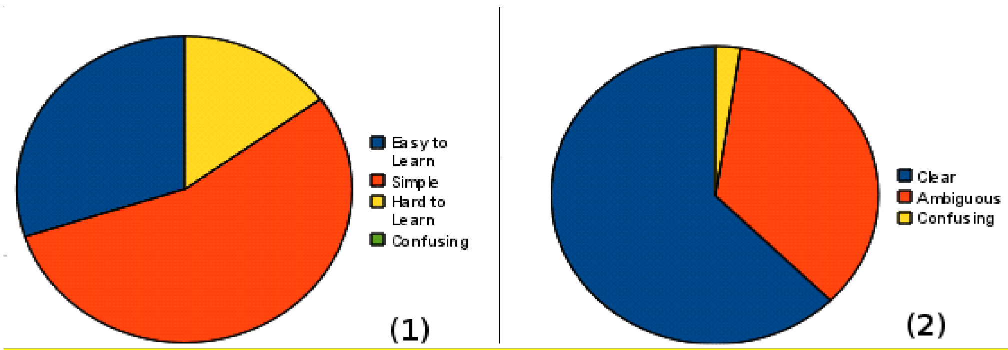

Figure 8 – Ease of Use and Understanding

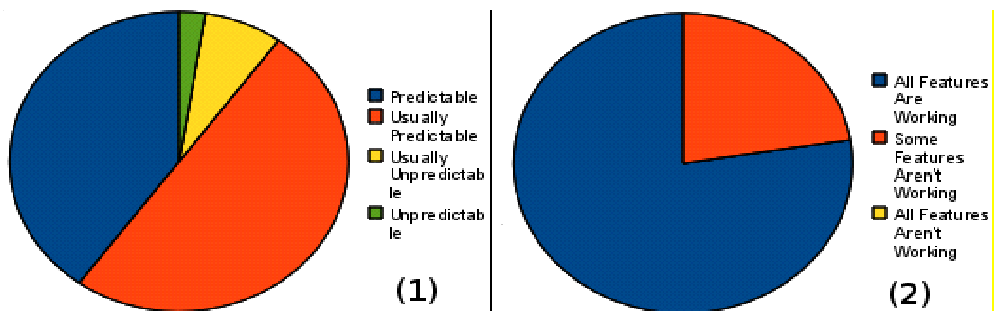

Figure 9 – Functionalities and predictability

Table 1 – Performance Test (Curitiba, 2011)

Concurrent Users/ Requests	Time taken for test	Total transfered	Complete requests	Failed requests
32/100	0.626 seconds	58800 bytes	100	0
64/500	10.163 seconds	294000 bytes	500	0
128/640	8.979 seconds	376320 bytes	640	0
320/1000	2.347 seconds	589764 bytes	1000	0
320/10000	7.806 seconds	5888232 bytes	10000	0
320/100000	25.517 seconds	588845864 bytes	100000	0
640/100000	48.952 seconds	588000000 bytes	100000	0

Table 1 shows that the server has a low latency even with a great number of requests from many concurrent users. Also, the table shows that there are no considerable performance issues and losses of requests. Though the number of requests and users has been greatly increased, the performance of the server didn't deteriorate to a point where requests started to fail or take a long time to be completed.

ANALYSIS AND DISCUSSION

We shall observe that SBIM is developed using portable softwares (PHP, Javascript and PostgreSQL). That is way the compatibility test verifies that the system properly behaves in a variety of operating systems and browsers.

SBIM is clearly organized into packages and its classes relying mostly on composition to build new classes which simplifies its maintenance and extensibility. Besides, these are low cost tasks due to the fact that the development platform is free.

The SBIM system is already accessible and functional. It has 29 images and 8 bases. There are 13 registered users, who are uploading new images, making suggestions to improve the system and detecting bugs, which could not be detected during the validation protocol. The proposed SBIM can be seen at http://200.236.3.11/~gustavo/sbim/.

Figure 10 shows the current state of SBIM. Figure 10-1 displays the form to request the creation of an image database. Figure 10-2 represents the FAQ, where the most common questions are answered. Figure 10-3 shows the advanced search form. The small colored squares on Figure 10-4 displays how many times an image has been viewed, how many messages were sent about it and how many features were detected by the yet to be implemented CBIR algorithms.

The usability test aimed to gather information about how user-friendly the interface is and if the users feel comfortable using the website. This is a fundamental point because the end users of SBIM system will be involved in

Figure 10 – SBIM Overview

multidisciplinary tasks from biology, image-guided diagnosis and computer science. For instance, in the medical image analysis, the doctors observe a x-ray and elaborate a diagnosis. Moreover, computational scientists develop image processing methods to translate the doctor´s knowledge into a feature vector. In this process, the doctors and computer scientists must discuss and share viewpoints and experiences. Meanwhile, users might write markings on the image and assign comments to it in order to keep the main points of a discussion.

The marking tools and e-mail service facilities can fulfill the communication issues but we need enough performance even considering a standard internet connection. The results presented for performance tests show that all these requirements are addressed by SBIM.

CONCLUSION AND FUTURE WORKS

The SBIM system provides interactive tools to share and analyze the images stored in its databases. This is implemented through algorithms for zooming and marking, a message system and automatic feature extraction resources. Therefore, it is achieving it's objective, that is to offer an efficient way to share images integrated with tools to analyze them.

The results from the interface tests pointed out modifications to improve the messages that it displays in order to make it more understandable. We are implementing such modifications for the next release of the SBIM system. Besides, in further works, a CBIR functionality will be added in the system. Also, other image processing techniques will be implemented to improve the facilities for image analysis.

REFERENCES

1. Gammon K. Nature.com. Networking: four ways to reinvent the Internet [Online]. 2010 Feb [cited 2011 feb 10]. Available from: URL: http://www.nature.com/news///2010/100203/full/463602a.html.
2. Bouchiha B, Malki M, Mostefai A, El K. A framework for reegineering Web applications to Web services. INFOCOMP. Journal of Computer Science. 2008; 7(1):19-26.
3. Morphbank [Online]. [cited 2011 feb 7]. Available from: http://www.morphbank.net/.
4. Biology Image Library [Online]. [cited 2011 feb 8]. Available from: http://www.biologyimagelibrary.com/.
5. JCB Data Viewer [Online]. [cited 2011 feb 7]. Available from: http://jcb-dataviewer.rupress.org.
6. MACC [Online]. [cited 2011 mar 19]. Available from: http://macc.lncc.br/.
7. Guliato D, Bôaventura RS, Maia MA, Rangayyan RM, Simedo MS, Macedo TAA. INDIAM – An e-learning system for the interpretation of mammograms. J Digit Imaging. 2009; 22(4):405-20.
8. Wangenheim AV, Prüsse M, Maia RS, Abdala DD, Regert AG, Nobre LFS, Comunello E. Recording and reenactment of collaborative diagnosis sessions using DICOM. J Digit Imaging. 2009; 22(6):605-19.
9. Pires SR, Medeiros RB, Schiabel H. Banco de imagens mamográficas para treinamento na interpretação de imagens. Radiol Bras. 2004; 239-44.
10. Welter P, Riesmeier J, Fischer B, Grouls C, Kuhl C, Deserno T. Bridging the integration gap between imaging and information systems: a uniform data concept for content-based image retrieval in computer aided-diagnosis. JAMIA. 2011; 18(4):506-10.
11. Oliveira JEE, Araújo AA, Deserno TM. Content-based image retrieval applied to BI-RADS tissue classification in screening mammography. World J Radiol. 2011 Jan; 3(1):24-31.
12. Azevedo-Marques PM, Caritá EC, Benedicto AA, Sanches PR. Implantação de um RIS/PACS no Hospital das Clínicas de Ribeirão Preto: uma solução baseada em Web. Radio Bras. 2005; 37-43.
13. Azevedo-Marques PM, Trad CS, Júnior JE, Santos AC. Implantação de um Mini-PACS (Sistema de Arquivamento e Distribuição de Imagens) em Hospital Universitário. Radiol Bras. 2001; 221-4.
14. Pressman RS. Engenharia de software. 6th ed. São Paulo: Pearson Makron Books; 1995.
15. Porcides GM, Stein LH, Kamada T, Neves LAP, Giraldi GA. An on-line medical imaging management for shared research in the Web using pattern features. In: Anais do VI Workshop da Visão Computacional; 4-7 julho 2010; Presidente Prudente, São Paulo:UNESP; 2010.
16. Oppermann R. User-interface design. Institute for Applied Information Technology, Germany: 2010.
17. World Wide Web Consortium [Online]. [cited 2011 feb 3]. Available from: URL: http://www.w3.org/QA/Tools/.
18. The Apache Software Foundation [Online]. [cited 2011 feb 10]. Available from: URL: http://httpd.apache.org/docs/2.0/programs/ab.html.