UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

ARTICLE ENGLISH

TITLE

"Modulo de Contabilidad para la Cooperativa de Ahorro y Crédito Unión Cochapamba"

AUTHOR: Edgar Daniel Picuasi Duque

DIRECTOR: DRA. CPA. MARÍA DE LA PORTILLA VERA   MBA

Ibarra – Ecuador

OVERVIEW

In the development of the Integrated System of Financial Information and in particular the implementation of the Countable Module for the saving cooperative and credit Union Cochapamba", several stages and processes have been continued, the same ones that it is described document presently.

The Integrated System of Financial Information consists of seven modules: Module Administration and Security, Module of Savings, Module Cashier, Module Accounting, Module Wallet, Module Clients, Module Audit.

The aplicativo was developed used the documentation methodology RUP, which you/they have been continued had allowed to lift the processes and details that bear the creation of the module, for the implementation of the same one open standards as it is the case of MVC that allows to maintain a separate application and distributed logically.

CHAPTER I

1. INTRODUCTION

The Cooperative of Saving and Credit Union Cochapamba from now on CUC with home in the parish of Ambuquí, Community of Chaupi Guaranguí, Canton Ibarra of the County of Imbabura, is constituted like a cooperative without ends of lucre and it will be governed by the law of Cooperative, its General Regulation, for other laws that applicable fueren and for an unique statute approved by the pertinent directive.

The responsibility of the Cooperative before third, is limited to its social capital and the one

capital that personal subscribed hubieren in the entity.

The Cooperative will have indefinite duration but however he/she will be able to be dissolved and to be liquidated by the causal ones and in form settled down in the law of Cooperative, its General Regulation and the norms pointed out in the statute.

1.1 Objectives of the Cooperative.

The objectives of the Cooperative are the following ones:

1. To promote the economic cooperation among their partners for whose execution will receive the savings, contribution certificates and deposits that these they carry out, it will make collections and payments as well as all those necessary operations for the invigoration of the credit cooperative inside the legal mark allowed by the cooperatives.

2. To grant loans to their members of conformity to the regulation that settles down for the effect.

3. To provide to their associates bigger capacity in the economic and social, by means of an appropriate education cooperativista.

4. To establish nexuses inside of and outside of the country with similar entities in benefit of the cooperative.

5. To obtain sources of internal and external financing for the development of the institution.

6. Other services and other activities that are framed in the law and Regulation of Cooperative to settle down and other laws that are applicable that bear to the social and economic improvement of their members.

7. To make inherent activities to look for financing with the purpose of to capitalize and to grant more benefits to the associates.

1.2 Current situation of the Cooperative.

The current situation of the cooperative you can be reflected through the articles that it is presented next.

Mission

"To be a highly efficient rural Cooperative and of wide covering, with a permanent innovation of products and saving services and credit with a culture of quality in the whole human resource."

Vision

"The Cooperative of Saving and Credit Union Cochapamba until the year 2015, it will develop and it will implement administrative and financial strategies that allow us to have big and competitive branches that generate microempresas without neglecting the environment, in such a way that their partners interactúen like a family in harmony with God and the nature."

CHAPTER II

2. Theoretical Framework

In this chapter the characteristics, operation and definitions are described of each one of the tools used for the development and implementation of the present project.

2.1 Legal norms

The legal or juridical norms are rules or classifications dictated by some competent authority according to an approach of value and whose nonfulfillment brings a sanction.

2.2 Tools and free standards

2.2.1 XML

It is a generalized language of marks (Expandable Markup Language), it is a language used to structure information in a document or in general in any file that contains text, as for example files of configuration of a program or a chart of data.
It has won a lot of popularity in the last years due to being an open and free standard,

created by the Consortium World Wide Web, W3C (the creators of the www), in collaboration with a panel that includes representatives of the main companies software producers.
The XML was proposed in 1996, and the first specification appeared in 1998. From then on their use has had a quick growth that one expects him to continue during next years.

2.2.2 HTML

It is a simple language that is good us for the creation of pages web or to write hipertexto, that is to say presented text in a structured way and easy to understand.

2.2.3 CSS

It is a language of leaves of styles created to control the presentation of the defined documents previously as HTML or XHTML, and it is presented as the best form of separating the contents and their presentation and it is indispensable for the creation of pages complex web.

2.2.4 JAVASCRIPT

JavaScript is a programming language that is used mainly to create pages dynamic web.
A page dynamic web is that that incorporates effects like text that he/she appears and it disappears, animations, actions that are activated when pressing bellboys and windows with warning messages to the user, technically, JavaScript is an interpreted programming language, for what is not necessary to compile the programs to execute them. In other words, the programs written with JavaScript can be proven directly in any navigator without necessity of intermediate processes.

2.3 Security in Systems Web

A system web makes use of the protocol HTTP for interactuar with the users or other

systems, for it the client uses a navigator, the problems of security can come from programs web, although in her biggest part they are consequence of shortcomings in the logic and the design of the own application.

The necessity to guarantee the security in the data that flow in the use of a system web makes that day by day they take measures to maintain safe the data.

So that a system web can be defined as insurance he/she should have in consideration these four characteristics:

1. Integrity: The information can only be modified by who is authorized and in a controlled way.

2. Confidentiality: The information should only be readable for those authorized.

3. Readiness: It should be available when one needs.

4. I don't reject: The use y/o modification of the information on the part of an user should be irrefutable, that is to say that the user cannot deny this action.

The following aspects show possible vulnerable points in applications web:

2.3.1 Security in the Client

A client is the final user of the system web and you/he/she is necessary that in him they take cautions to avoid the vulnerability against attacks toward the system, the actions carried out in the team should be controlled somehow, for it you can opt to carry out constant verifications of the software that settles, the use of some modernized antivirus and other programs or protection that avoid attacks.

2.3.2 Security in the Server

The development and setting in production of an application web require of several tools, the same ones that should complete an approach of security and to have the respective configurations that avoid attacks against the application.

1. the servant web. - it is considered the public face of the organization and it is without a doubt a target of attacks toward the information, therefore it is necessary to have caution with the services and permits that he/she lends the servant.

2. servant of Applications. - it Generally negotiates most of the functions of business logic and of access to the data of the application. It is very convenient to revise the log files periodically (access_log and error_log in the Apache servant of applications) to detect possible attacks to the servant of applications.

3. servant of databases. - The organizations usually make several errors that return vulnerable to the databases, as leaving the databases of the tests in the production servants, or to connect sensitive data to applications that give expensive to the web and they are of easy hackeo, one of the main attacks is the injection to SQL [structured consultation language], because the attackers introduce code ribbons SQL to the weak fields of the applications web.

2.3.3 Security on the Aplication

For the implementation of the security in the applications web should be considered the following ones alternative:

1) Access control.- A very important aspect of an application web belongs the access control from the users to restricted areas of the application here two concepts intervene:

1.1. Autentificación.- it is the process of determining if an user is who says to be been he can make in several ways. Some of them are:

- Autentificación basic HTTP.
- Autentificación based on the application

1.2. Authorization.- it is the act of checking if an user has the appropriate permission to consent to a certain file or to

carry out a certain action, once it has been authenticated.

To design the mechanism of access control demands:

- To determine the information that will be accessible for each user.

- To determine the access level from each user to the information.
- To specify a mechanism to grant and to revoke permits to the users.
- To provide functions to the authorized users: identification, desconexión, petition of help, consults and modification of personal information, password change, etc.
- To adjust the access levels to the information to the politics of security of the organization.

2) validation of entrance data.- The most frequent problem that you/they present the applications web is not to validate the entrance data correctly, this gives place to some of the most important vulnerabilities in the applications, as the injection SQL, the Cross-Site Scripting and the Buffer Overflow. Some of the aspects to take into account are:
1. entrance sources.
2. injection.
3. protection strategies.
4. specific vulnerabilities.

3) Sure programming.- to avoid or at least to diminish the vulnerabilities of an application web is very important to follow some correct programming practices. some of the most important are:
1. initialization of variables.
2. administration of errors.
3. protection of information.

2.3.4 Security in the Communication
SSL (Secure Socket Layer) it is a protocol to assure the transport of data between the client and the servant web, we can recognize a connection HTTP SSL it has more than enough because the prefix appears ' https' instead of ' http' in the URL.
The way in which the SSL works is the following one:
When the client requests a page SSL, the servant sends a certificate that is obtained of an authority reliable certificadora. The certificate contains the public key of the servant. After to make sure that the certificate is correct and that the servant is genuine, the client generates an aleatory number, the session key. The key is encriptada with the public key of the servant and correspondent. The servant desencripta the message with their private key. Now both sides only have a key of well-known session for them two. All communication from and toward them it is encriptada and desencriptada with the session key.
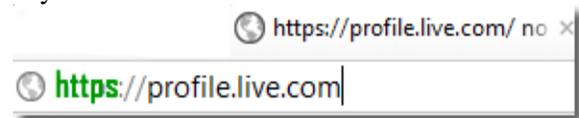
SSL provides a sure communication between client and servant allowing the mutual autentificación, the use of digital signatures and guaranteeing the privacy by means of encriptación

2.4 Data Base

2.4.1 PostgreSQL

The servant of databases Postgres offers a power additional notable when incorporating some concepts with those that the users will be able to extend the system easily. The concepts added Postgres are:

1. classes

2. inheritance

3. types

4. you work

2.5 Server Aplications Apache Tomcat
2.5.1 Introduction a Apache Tomcat

Apache Tomcat (also called Jakarta Tomcat or simply Tomcat) it works as a servlets container developed under the project Jakarta in the Apache Software Foundation. Tomcat implements the specifications of the servlets and of Java Server Pages (JSP).

Tomcat is maintained and developed by members of the Apache Software Foundation and independent volunteers, the users have free access to its code source and its binary form in the terms settled down in the Apache Software Licence.

The first distributions of Tomcat were the versions 3.0.x, the most recent versions are the 7.x that implement the specifications of Servlet 3.0 and of JSP 2.2.

Starting from the version 4.0, Tomcat uses the servlets container Catalina.

2.5.2 Structure of the Apache servant Tomcat

The binary distributions of Tomcat maintain the structure of directory, keeping in mind that the hierarchy of directory of installation of Tomcat includes the following thing:

1. bin. - This directory contains the beginning scripts, to stop, to restart and other scripts and executable for the operation of the servant.

2. common. - This directory contains a sinnúmero of common classes that can use the Container of ServletsCatalina and the applications web that are housed in the servant web.

3. conf. - This directory contains some configuration files server.xml that is the main file of configuration of Tomcat and the file web.xml which configures the values for defect for the different applications deployed in Tomcat including like when it is used you Hibernate yourself, RichFaces, etc. Also in this directory they are those

4.- Corresponding DTD for the configuration of Tomcat. The basic function of the DTD is the description of the format of data.

5.- Logs. - Here it is where Tomcat houses the messages when it is executed, he/she closes or errors exist when executing the applications.

6. server. - In this directory classes that are only used by the container of Servlets Catalina exist.

7. shared. - Location where they are the classes shared by all the applications web housed in the servant of applications Tomcat.

8. webapps. - This directory is where they should be copied or where all the files of an application web should be.

10. work. - the files are located that are believed that is to say during the execution of the servant the temporary storage of files and directory.

2.6 Platforms and frameworks

For the development of the aplicativo it was opted by the use of platforms and frameworks that are coupled to the necessities required for of users of the cooperative, it was also considered the existent documentation and the use easiness.

This platforms and frameworks are described next:

2.6.1 Java

So that a computer can understand the code lines that we write through a development environment (IDE), it is needed of a special program that he/she can interpret and to transform this code to a language of low level or language machine, for it the encardo of carrying out this function is the development platform in this case Java.

2.6.2 hibernate you

The connection and interaction of a system or application with a database are crucial since all the information should be supported, for it the readiness should be high, Hibernate you it allows to use the same programming knowledge guided to objects for the development of consultations and transactions in general.

Hibernate you it is a tool of Mapeo object-relacional (ORM) for the platform Java, facilitates the mapeo of attributes between a database traditional relacional and the pattern of objects of an application, by means of declarative files (XML) or annotations in the beans of the entities that allow to establish these relationships.

Capa de Persistencia

2.6.2 JSF (Java Server Faces)

The development of applications or systems web without the use of a methodology or a design pattern that it allows to maintain an order and a separation of the code and other elements make that in many occasions he/she mixes in oneself I file JSP user's interfaz, the validation rules, the access to the database, etc.,

2.6.3 RichFaces with framework RIA.

RichFaces is a framework of open code that he/she adds the ajax capacity in applications JSF without appealing JavaScript.

RichFaces takes advantage of to the maximum the cycle of life of JavaServer Faces, that is to say the validation, the conversion facilities and administration of the static and dynamic resources. The components of RichFaces come clever for their use out-of-the-box, for what the developers can save time immediately to take advantage of the characteristics of the components to create applications Web that provide improvements in the user's experience. RichFaces also includes a strong support for the skinnability of applications JSF.

2.6.4 Tool of Reports

2.6.4.1 JasperReports

1. It is a gratuitous tool and open source that it is composed of a group of bookstores Java to facilitate the generation of reports in our applications so much web as of desk. The reports are defined in a file XML which will be compiled by the bookstores jasperreport and they will generate a file .jasper that we will use to stuff and to show it in the final reports
2. JasperReports is totally written in Java and you can use in a great variety of applications of Java, including JEE or applications web, to generate dynamic content. The operating system one for JasperReports is not since a problem it works perfectly so much in platforms of Microsoft Windows like in GNU Linux and in both platforms he/she has been proven their good operation.
3. For the operation of JasperReports it is needed 2 main things basically the first one it is that is to say JDK 1.4 or a superior version the virtual machine of Java which makes possible the operation of any application Java and like it had been explained previously that JasperReports is totally written then in Java it is main requirement to have installed the JDK. And it is also needed of the following bookstores to more than those of JasperReports to begin to generate reports.
   1. jasperreports-javaflow-4.0.2.jar
   2. jasperreports-4.0.2.jar
   3. jasperreports-applet-4.0.2.jar
   4. jasperreports-fonts-4.0.2.jar

For the generation of Reports through you Hibernate yourself it is necessary also the bookstore that hsqldb.jar

Operation of JasperReports

The operation of JasperReports is similar to a compiler and an interpreter, that is to say that he/she translates each instruction or sentence

of the file XML to a language machine and immediately it is executed and the file characteristic of JasperReports is believed (.jasper).

The user takes charge of designing the report coding it in XML with the labels and attributes defined in a called file jasperreports.dtd which is part of JasperReports.

In the file XML the user defines all the parameters that were used in the report, basically describing this way where to place text, images, lines, details of charts, rectangles, how to acquire the data, as carrying out certain calculations to show calculated fields, etc.
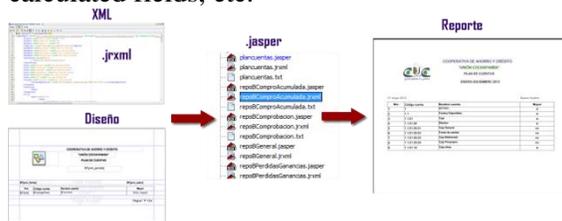
3. Operation of the System

Presently I surrender the Functional Models they are detailed elaborated as base for the computer development of the Integrated System of Financial Information. In this document the Financial Administration is defined as a system that integrates the subsystems of Administration and Security, Savings, Wallet, Clients, Boxes, Accounting, and Audit which are interrelated as for its normatividad, operability and information that generate, maintaining this way the characteristics and necessities characteristic of each one of these subsystems. Starting from the description of the functions characteristic of the sectors interveners in the processes of each one of the existent departments in the Cooperatives and of the information that you/they generate, we have designed to the Integrated System of Financial Information having as main modules the following ones:
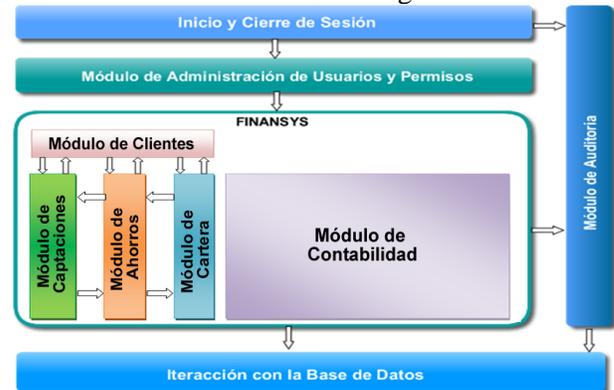
3.1 Description of the modules

Next it is indicated the description and operation of the module of Accounting.

CHAPTER VI

4. DESIGN AND DEVELOPMENT OF THE APLICATIVO

I. Incepción. - During the initial phase the central idea of the product is conceived, the document of vision arms, they are revised and it confirms our understanding on the central objectives of the business.
We should understand the arguments in reason favor the project it should be attempted. The incepción phase establishes the viability of the product and it defines the reach of the project.
II.   Elaboration. - During the elaboration phase most of the Cases of Use are specified in detail and the architecture of the system is designed.
 The significant risks are identified and they get ready the calendar, the work team and the cost of the project.

III.   Construction. - to Develop the product and to evolve the vision, the architecture and the plans until the product in a first version is clever to be a correspondent to the community of users it is that the focus of the product moves from the base architecture to a system the sufficiently complete thing as to take it to the user. He/she is carried out the design to take it to code source.

IV.   Transition. - In the transition phase the objective is to guarantee that the requirements have been completed, with the satisfaction of the interested parts. This phase often begins with a version beta of the application. Other activities include the preparation of the atmosphere, they are completed, they are identified and they correct defects. This phase finishes with the training or sociabilización of the product.

CHAPTER V

5. CONCLUSIONS   AND RECOMMENDATIONS

1.1 Conclusions

1.The inquiry and summary of functional requirements of the system is momentous since this allowed the obtaining of an effective countable module and with it a financial system of quality.
2. to have a methodology for the development of the software, it is constituted as indispensable since it clarifies the perspective of that wanted by the user.
3. the use of a language like java allow to maintain a complete control of the development of the application.
4. using free tools is beneficent for the institution, since it doesn't finance licenses and it can use the software without any restriction.
5. in a financial entity as in this case it is convenient and of great benefit to use an application client-servant, since all the operations stay centralized and all consent to oneself resource.
6. fusing different standards opened up in the development of the application allows to create friendlier applications with the final user.

1.2 RECOMMENDATIONS

1.   It is advisable to use a methodology in this case RUP in the summary of the information of the diverse processes and the analysis of requirements since they allow a good and clear development of the software.
2. for the development of an integrated system that it consists of several modules, it is necessary to use a versionamiento tool since it facilitates the centralized development.
3. it is necessary to take very in bill the dispositions requested by the final and other users involved in the use of the system, since

they have the experience in the handling of the information and other processes.

4. to use good programming practices allows to build applications of easy maintenance and mainly scalable.

5. to qualify the personnel with the developed system is very important since this way the final user familiarizes with the functionality of the aplicativo and it clears any doubt.

6. to maintain the clear information at code level and of database it is necessary, since in the development of the integrated system, the other modules can verify that it contains each field or method.

7. to document the errors obtained in the code process is useful, since in later cases one can make use of the previous suggestions.

8. it is necessary to treat topics of software costs, since in the labor environment questions arise on the part of interested people and it is not easy to present a real price.

6. BIBLIOGRAPHY AND REFERENCES

Books:

1) Birnam Stewart. (2011). *Java Distribuido*. E.E.U.U: McGrawHill.

2) Francisco Rueda Profesor, U. d. (2010). *Sistemas Operativos*. México: McGraw Hill.

3) Ing. Maribel Sabana Mendoza. (2010). *Libro de postgreSQL*. Quito: Grupo Editorial Megabyte.

4) Coop Unión Cochapamba. (2011). *Estatuto de la Cooperativa "Unión Cochapamba*. Ibarra: N/A.

5) Coop. Unión Cochapamba. (2011). *Plan Operativo Anual*. Ibarra: N/A.

Web:

1) Centos. (01 de 05 de 2011). *Centos*. Recuperado el 02 de 01 de 2012, de http://www.centos.org/

2) Corporation Creative Commons. (31 de 08 de 2009). *Libros Web*. Recuperado el 2 de 03 de 2012, de http://www.librosweb.es/

3) Fedora Project. (2 de 09 de 2008). *Security Guide*. Recuperado el 28 de 05 de 2012, de http://docs.fedoraproject.org/esES/Fedora /16/html/Security_Guide/index.html

4) Jboss org. (07 de 05 de 2011). *Richfaces*. Recuperado el 10 de 07 de 2012, de http://docs.jboss.org/richfaces/latest_3_3 _X/en/devguide/html/Introduction.html

5) Jsanroman. (05 de 05 de 2012). *Jasperreport*. Recuperado el 21 de 07 de 2012, de http://jsanroman.net/2007/11/C2BFque- es-jasper-reports-2/

6) Libros web. (01 de 01 de 2012). *librosweb.es*. Recuperado el 01 de 08 de 2012, de http://www.librosweb.es/

7) Masadelante. (02 de 05 de 2011). *Sistemas Operativos*. Recuperado el 03 de 04 de 2012, de http://www.masadelante.com/faqs/sistem a-operativo

8) Ndeveloper. (01 de 04 de 2011). *Ndeveloper*. Recuperado el 03 de 06 de 2012, de http://www.ndeveloper.com/ndeveloperD ocuments/documents/nDeveloper_JavaSe rverFaces.pdf

9) Normativa De Basilea. (2012). *Reformas Introducidas en el Sistema Financiero Ecuatoriano*. Quito: Norma.

10) OAS. (12 de 06 de 2011). *Ley General Institucion Financiera*. Recuperado el 14 de 03 de 2012, de

http://www.oas.org/juridico/mla/sp/ecu/sp_ecu-mla-law-finance.html

11) Owasp org. (15 de 01 de 2006). *Owasp.org*. Recuperado el 02 de 04 de 2012, de www.owasp.org/documentation/topten

12) Postgresql. (12 de 04 de 2010). *postgresql.org*. Recuperado el 07 de 05 de 2012, de http://www.postgresql.org.pe/articles/introduccion_a_postgresql.pdf

13) Richard Monson-Haefel. (2011). *Enterprise JavaBeans 2da Edición.* E.E.U.U: O' REILLY.

14) Rincon. (01 de 03 de 2011). *Concepto de Base de Datos*. Recuperado el 06 de 05 de 2012, de http://html.rincondelvago.com/concepto-de-base-de-datos.html

15) sbs.gob.ec. (12 de 29 de 2009). *Decreto Cooperativas*. Recuperado el 01 de 02 de 2012, de http://www.sbs.gob.ec/medios/PORTAL DOCS/downloads/normativa/decreto_194 _cooperativas_29_dic_09.pdf

16) The World Wide Web Consortium, W. (23 de Abril de 2003). *The World Wide Web Consortium (W3C)*. Recuperado el 03 de 02 de 2012, de http://www.w3.org/

17) Tom Negrino, D. S. (2011). *Javascript.* Estados Unidos : Tontitown.

18) Universidad de Sevilla. (12 de 04 de 2011). *LSI*. Recuperado el 02 de 03 de 2012, de http://www.lsi.us.es/~quivir

19) Wikipedia. (17 de 12 de 1996). *es.wikipedia.org*. Recuperado el 03 de 03 de 2012, de http://es.wikipedia.org/wiki/Hojas_de_estilo_en_cascada

20) Wikipedia. (08 de 04 de 2007). *es.wikipedia.org*. Recuperado el 01 de 03 de 2012, de http://es.wikipedia.org/wiki/HTML

21) Wikipedia. (20 de 04 de 2011). *es.wikipedia.org*. Recuperado el 07 de 04 de 2012, de http://es.wikipedia.org/wiki/Fedora_(distribución_Linux)

22) wikipedia Java. (03 de 07 de 2010). *Java*. Recuperado el 01 de 05 de 2012, de http://es.wikipedia.org/wiki/Plataforma_Java

23) wikipedia NetBeans. (01 de 04 de 2010). *NetBeans*. Recuperado el 29 de 05 de 2012, de http://es.wikipedia.org/wiki/NetBeans

24) Wikipedia, T. (03 de 05 de 2011). Recuperado el 23 de 05 de 2012, de http://es.wikipedia.org/wiki/Tomcat