TECHNICAL UNIVERSITY OF NORTH

FACULTY OF ENGINEERING IN APPLIED SCIENCES

CAREER OF ENGINEERING IN COMPUTER SYSTEMS

EXECUTIVE SUMMARY

TITLE:

"CONTROL SYSTEM FOR INSPECTION PROCESS MANAGEMENT PLANNING GAD-I"

Author: Edison Raúl Rivadeneira Terán

Director: Ing. Irving Reascos

Ibarra – Ecuador

2014

GENERAL SUMMARY

This work is based on developing a system for process control in the Planning of GAD-I, specifically for the registration and control of inspections carried out to the properties for the subsequent adoption of a process, this with order to improve and expedite the process of getting the citizens, and help in decision-making at the Architects and officials within the department.

For its development, worked with the existing methodology in the Directorate of TIC which it is based on agile development methodologies like XP and SCRUM where its main advantage is the direct and ongoing work with users facilitating requirements elicitation and creation system.

The software tools used for the development of this project are based on the MVC architecture (Model View Controller) which is working with the Symfony framework on the server side and ExtJS4 for customer views. Besides the handler system database was used PostgreSQL.

The general operation of the system is to sort properties according to their location on the GIS map and generate a schedule of inspections which is allocated according to the current schedule of the date of inspection visit to each of the properties in which mainly recorded data and pictures obtained record who serve as support for future reference and reporting related to the registered inspection.

In conclusion it is determined that the development and use of the system generates better performance in the work of the Department as to the realization of the process because there is a monitoring and recording of inspections.

INTRODUCTION

Antecedents

One of the processes that take place within the Planning Department are inspections of construction taking place in the territory comprised within the limits of Canton Ibarra, that it is run by the inspectors which are responsible for verifying that the development of a work or construction is taking place in accordance with the plans and specifications approved parameters according to rules set out in Ordinances of Architecture and Town Planning Code and other relevant laws.

Mission GAD-I

The Autonomous Decentralized Government of San Miguel de Ibarra plans, regulates, implements and promotes sustainable integral development of Canton, through efficient and transparent service quality with the active participation of socially responsible citizenship in order to achieve the good life.

Vision GAD-I

Being a leading municipality in socially responsible management, to ensure fairness, honesty, hard work and efficiency why Ibarra will constitute a prosperous, attractive and inclusive canton, capital services and knowledge concerning the good life in the northern region of Ecuador.

Present status of the problem

Information on construction, inspection reports and other documents relating to this process are taken mostly manual and printed within folders so making it difficult to keep track and right, fast and effective management by inspectors to verify and construction works being carried out within the City of Ibarra thus results in the illegal construction of buildings that have not complied with the rules and requirements to carry them out.

Prospective problem

In the event that this project is not developed would remain irregular and uncoordinated inspections, so there will be many illegal constructions overlooked and results in the insecurity in the population to not build buildings that meet standards to improve and ensure quality of life.

Besides the delay in the process is presented in the Planning Department where the Inspectorate within some processes taking place within the same is required.

General Objective

Develop and implement a Process Control System Inspections for the Planning of the Autonomous Decentralized Government Ibarra GAD-I and thus facilitate monitoring, information management and inspection results within the processes running in this direction.

Specific Objectives

- Study the processes that are carried out to execute inspections and control constructs.
- Analyze the types of inspection running.
- Contribute to the analysis and design of processes.

- Making a business process reengineering to automate, organize and reduce the execution time.
- Study and apply a methodology for rapid development of system development.
- Training, learning and using the Symfony framework as a tool for system development.
- Train personnel involved in the inspections process.
- Inform the citizen of the date of property inspection.

Justification

The inspection process has as main beneficiaries citizens because it helps control the buildings meet safety standards to ensure the quality of life of people; addition to verifying that there are no illegal constructions which do not meet the existing requirements in the Planning Department.

Architects will also benefit by taking control and managing inspections to help coordinate more effectively the processes of Planning Department to be able to take statistics and make quick, agile and timely decisions.

Scope

Based on the information that must be stored, controlled and managed by the system creating these modules is determined:

- Schedule of Inspections
- Inspection Sheets
- Agenda for Inspections
- Registration Inspections (PC / Mobile)
- Reports

In addition the system has permissions for two types of users; an administrator role can create schedules and records of inspection and the inspector or architect role is responsible for creating inspections, registering and displaying statistical reports.

CHAPTER I

1. THEORITICAL FRAMEWORK

The tools and methodology used to develop the concepts of making known database languages, frameworks, reports and system development methodology described in this chapter.

1.1. Development Tools

1.1.1. Application Servers

❖ HTTP Apache

It is an open source HTTP web server for Unix platforms (BSD, GNU / Linux, etc..), Microsoft Windows, Macintosh and other, which implements the HTTP/1.1 protocol and the notion of virtual site. [1]

❖ Tomcat

Tomcat works as a servlet container developed under the Jakarta project at the Apache Software Foundation and implements specifications for servlets and JavaServer Pages (JSP) specifications from Sun Microsystems and can be used on any operating system that has the Java virtual machine. [2]

1.1.2. Databases

❖ PostgreSQL

PostgreSQL is a management system for object-relational databases, distributed under BSD license and its source code freely available. It is the management system open source BDD stronger market. [3]

❖ PostGIS

It is an extension to the system database PostgreSQL object-relational and allows the use of GIS objects. With PostGIS we can use all the objects in the OpenGIS specification as points, lines, polygons, multi-line, multi-point, geometric and mapping for collections. [4]

1.1.3. Programming Languages

❖ PHP

PHP is one of the most widespread languages in the Web server side. Allows embedding code fragments within the HTML page and perform certain actions in an easy and efficient way without having to generate programs in a different language. Provides various functions for operating databases in a simple way. [5]

❖ JAVASCRIPT

The JavaScript programming language is used in Web pages to fill gaps left by the plain HTML. Incorporating this language to our pages we increase the feature's functionality and user interaction, which becomes more dynamic pages. [6]

1.1.4. Frameworks

❖ Symfony

It is a PHP framework that facilitates and streamlines the development of web applications. Responsible for all common and boring aspects of web applications, allowing the programmer is dedicated to provide value by developing the unique characteristics of each project. Also separates the business logic, the logic of the presentation server and the web application. [7]

❖ Ext JS 4.0.7

It is a JavaScript library for developing interactive web applications using technologies such as AJAX, DHTML and DOM. It was developed by Sencha. [8]

❖ Sencha Touch 2

It is a JavaScript MVC framework that uses web standards HTML5 and CSS3 to create web applications for mobile devices with a touch screen with the appearance of being Native iOS, Android and Blackberry systems. Sencha Touch also allows developers to create applications for mobile platforms with browsers that implement the latest standards such as the WebKit browser engine. [9]

1.1.5. Maps

❖ MapServer

It is a rendering engine Open Source geographic data written in C. Beyond browsing GIS data, MapServer allows you create "geographic image maps", that is, maps contents for users. [10]

❖ MapFile

The main MapServer configuration file is a text file with extension. "Map", in which a number of parameters that define the available layers is included in the service, the style with which they represent, and symbolism format generate the image reference system, etc. [11]

❖ WMS (Web Map Services)

El servicio Web Map Services interactúa con los clientes a través de peticiones HTTP, produce espacialmente referenciada dinámicamente los mapas de información geográfica. Estos mapas se envían a los clientes en forma de imagen digital. [12]

❖ OpenLayers

It is a javascript library for free use to access, manipulate and display maps in web pages. It provides an API that allows the creation of web clients to access and manipulate geographic information from a variety of sources, and maps that can incorporate various dynamically added controls like zoom, measure distances and many other tools. [13]

❖ GeoExt2

It is open source and allows the creation of GIS desktop applications, such as through the web. This is a JavaScript framework combining GIS OpenLayers functionality user interface provided by the library ExtJS sencha. [14]

1.1.6. Reports

❖ JasperReports

JasperReports is a powerful open source reporting tool that has the ability to present reports on the screen, printer or into PDF, HTML, XLS, CSV and XML format. It is entirely written in Java and can be used in a multitude of Java enabled applications generating dynamic content. [15]

❖ iReport

It is a visual designer for JasperReports open source written in Java. It is a program that helps users and developers using the JasperReports library to visually design reports. Through a rich and easy to use interface, iReport provides the most important features for creating reports quickly. [16]

❖ PHP-JRU

PHP-JRU (PHP Utils Jasper Report) is a library designed to generate reports on the iReport tool designed from a simple application written in PHP. [17]

❖ PHP/Java Bridge

It is a bridge XML-based network protocol, that can be used to connect a native script engine PHP with Java virtual machine. Also allows developers to access Java classes in PHP. It also allows access to PHP scripts into Java classes. Because of this interoperability, developers can build interactive web pages using PHP, but delegate tasks to Java classes. [18]

1.2. Development Methodology

1.2.1. Definition

The GAD-I Methodology rapid development is used and is constituted by Scrum XP and methodologies, which has taken the most efficient of each process, in order to obtain more efficient projects and in short time.

The documents that are part of the software development methodology are:

❖ Technology Project

Document where describe the set of activities to achieve a specific objective of research, development and technology innovation in a range defined time and cost, in addition to specifying the scope, justification, beneficiaries among other aspects of the system to be developed.

❖ Minutes of Meeting

Document containing the topics discussed in a workshop with people who may be in the area of systems or with users get objective requirements for the development of the system and then specify the acquired commitments that resulted from the meeting.

❖ User Stories

Document containing the end user requests on the computer system to develop.

❖ Product Backlog

Document that lists all user stories determining the priority of each.

❖ Stack Iteration

Document containing the tasks to be performed based on the user stories on a timeline detailing the specific times for their realization.

❖ User Manual

Document focused to the end user which details the functionality of the system in each of its modules.

❖ Technical Manual

Document focused for specialists in the area of systems in which describes the system architecture, installation, configuration, system requirements including technical data..

❖ Act of Term Projects

Document which details which has been terminated the activities of the development process and the application is approved by the people who are given the project.

❖ Characterization of Applications

Document where specified the type of application development has some of its detailed characteristics in summary .

CHAPTER II

2. INSPECTION PROCESS

This chapter identifies and describes the process executed within the Planning

Department to carry out inspection visits to the properties.

2.1. Objective

Control that the infrastructure works are performed according to the requirements and technical specifications provided by the Planning Department and their respective regulations.

2.2. Scope

The procedure applies to inspections that are performed within the canton Ibarra, implemented by the Department of Planning and Development. Starts with the internal or external inspection requirement to a property to prepare the technical report.

2.3. Roles and Responsabilities

- Planning Director
- Technical OO.PP
- Arcchitect/a Planning
- Secretary OO.PP/ Planning

2.4. Inspection Process Overview

Is described are the steps that are performed in the inspection process:

- Receiving documents and verify the inspection requirement.
- Search on the map the location of the properties to be inspected.
- Develop schedule and route inspections by areas or convenience.
- Request the topographic team to comply with the inspections.
- Check the inspections list to perform and prepare equipment and materials necessary to perform the inspection.
- Create transport request for mobilization of personnel.
- If the inspection is not for a topographic lifting or planimeter perform visual inspection of the property and review of technical parameters according to ordinances and regulations..
- If the inspection is for a topographic lifting or planimeter see the procedures that must be performed for execution.
- If the inspection day is not over, move to the next property on schedule to perform the requested inspection.
- If the day of inspections completed, process data and prepare inspection report.

CHAPTER III

3. DESIGN AND DEVELOPMENT OF INSPECTION

In this chapter, the software is developed following the process established by the development methodology GAD-I Directorate of ICT that is based on both agile development methodologies best known SCRUM and XP.

3.1. Roles

The development of a computer system is a task that is composed of several elements as processes, methodologies, development technologies and each has its significance and purpose that are carried out with the participation of a group of people with different roles and responsibilities.

Below is listed the roles that have been part of the development of the system:

- User
- TIC Director
- Responsible for Software
- Computer Systems Analyst
- Programmer

3.2. Getting System Requirements

To obtain the system requirements had to hold meetings with direct users to execute the inspection process; which in this case are the Architects working in the Planning of GAD-I.

These meetings helped the process that is carried out to execute an inspection and what the requirements requested by the user for the system architect will support the monitoring and recording of inspections are; information and topics are documented in the minutes of meetings and requirements obtained from these are reflected in the user stories.

In addition, a survey was conducted to the Architects for getting information on the chips where recorded data of inspection.

3.3. Development of User Stories

All user stories created during the development of the project which includes the information requirements set by the user of the system and its functionality as demonstrated by the acceptance tests are described in this section.

3.4. System Design

After obtained and described the system requirements in the user stories defined which are modules and system user types ; well then and Core Architecture for GAD View Integrated System-I and System Functional Inspections architecture is described.

3.4.1. Integrated System Architecture

The architecture of the GAD-I Embedded System, is mainly based on the MVC design pattern, where each of these components was used according to the needs and characteristics of the framework used to develop the system; besides having the ability to add more layers to better structure the code and improve scalability.

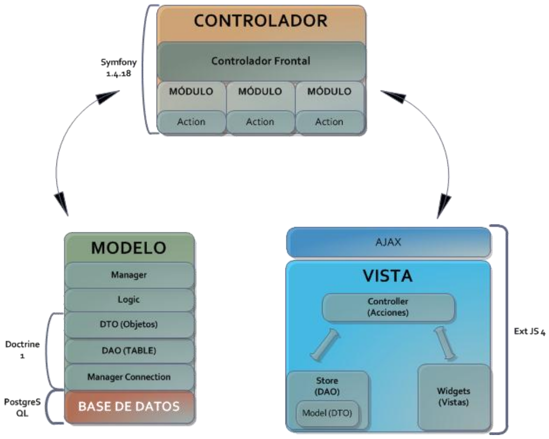

Figure 1: System architecture GAD-I

Model: This layer through using the Doctrine ORM allows communication and data collection through the DTO classes created from tables in the database and use the DAO to execute SQL and DQL queries.

The manager acts as an intermediary by which requests are received request from the controller who calls the queries SQL or DQL for obtaining data via persistent objects.

Controller: This layer is responsible for consultation and interact in the model data to be sent and presented in the view.

View: The view layer is created using the ExtJS framework which provides us with rich and dynamic components for better management and presentation of information to the client.

This framework uses as its main means of an Ajax request data connector, allowing improving the outcome of the response and access to it.

3.4.2. Core of view Integrated System

The GAD-I Core Embedded System related view is founded by a core architecture, so that this is more manageable and allows us to obtain a unique presentation and access to the system, as this has several applications have different functions depending on the permissions of the logged in user.

The Core of the view consists of the following layers:

Imi Ajax Connect: Allows communication between client view and the logic on the server (Front Controller).

Api Application: It is the primary instance of the Integrated System-level view that allows the management of controllers and views created in the execution of the application, that is responsible for eliminating instances and listeners assets to close an application.

Desktop: It is the main view of the Integrated System, is responsible for the load management applications within your environment.

Application Manager: This component managed the loading of the main view and an application assigning allowable options within the same depending on the user entering the Integral System.

Application: This component represents an application on the server level, in which each has layers: Windows Manager and Module.

3.4.3. Modules of Inspections System

According to the needs and requirements obtained is determined 2 types of users or roles for System Inspections which are:

User Manager: It is responsible for creating the Inspection Schedule and Inspection Sheets by process.

User Architect/Inspector: It is responsible for generating, recording, and query reports of inspections.

Then the developed inspection system modules are described.

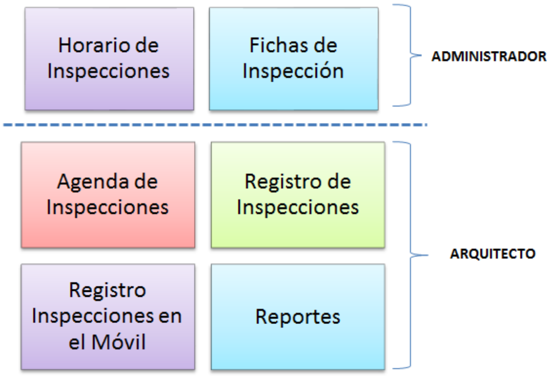

Figure 2: Modules System Inspections

Module Schedule Inspections: Module where are allocated parishes days and architects which information is collected to generate the schedule inspections then used to assign a date to visit the properties.

Listing Inspection Module: This module inspection records are created by process or type of process which contains categories and data that can be simple, single and multiple selection. This tab is used to record the data obtained at the time of inspection..

Agenda Inspection Module: This module are displayed procedures selected to perform inspection; further use of maps showing the location of the properties and displayed a created route that provides support at the time of inspection.

Module Registration Inspections: This module with web environment, inspections data such as are recorded: Photos, record, comments and people responsible for inspection. Besides that the inspections can then search through some filters like: dates, schedule, number of pending and approved inspections.

Registration Module at Mobile Inspections: This module is similar to the previous one except that its objective is to make a quick registration of the data obtained from the inspection; this is possible since it was created to run on mobile devices and use it in the exact place of the visit to each of the properties.

Reports Module: This module is used to generate and print graphs and statistical reports based on data inspections and can be used by architects for support in meeting existing indicators.

3.4.4. Functional Description System Inspections

The operation generally Inspections System is described in function of the 2 types of user:

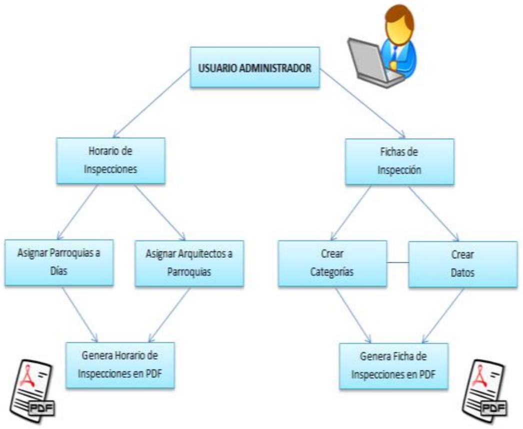
USUARIO ADMINISTRADOR
Horario de Inspecciones
Fichas de Inspección
Asignar Parroquias a Días
Asignar Arquitectos a Parroquias
Crear Categorías
Crear Datos
Genera Horario de Inspecciones en PDF
Genera Ficha de Inspecciones en PDF
PDF
PDF

Figure 3: System functionality based on User Administrator

Administrator: The administrator user has access to Schedule Inspection Module where by assigning parishes to days and architects can create and print the current Schedule of inspections which works within the Department of Planning.

This user also has access to module Sheets Inspections where you can view and create new records by type of procedure in which you can create or add categories and in turn these create the data that will contain the information that is used for recording inspection. This record can be viewed and displayed in PDF format.

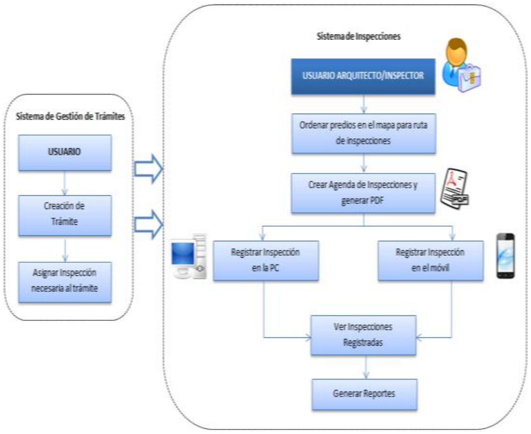
Sistema de Gestión de Trámites
USUARIO
Creación de Trámite
Asignar Inspección necesaria al trámite
Sistema de Inspecciones
USUARIO ARQUITECTO/INSPECTOR
Ordenar predios en el mapa para ruta de inspecciones
Crear Agenda de Inspecciones y generar PDF
PDF
Registrar Inspección en la PC
Registrar Inspección en el móvil
Ver Inspecciones Registradas
Generar Reportes

Figure 4: System functionality based on User Inspector

Inspector: This user has access to Agenda Inspections module that have a direct relationship with the Paperwork Management System as this is where the formalities necessary information are created.

After creating procedures with inspection are received and displayed in the Agenda Module Inspections and with the help of a map of the properties location, the user creates the route and agenda that results in generating a PDF file with the Basic information on the formalities and send an e-mail informing citizens the assigned date which will be visiting the property.

After having generated inspections, is performed registration that can be through a web and mobile environment ; these necessary data are recorded in the inspection and finally the user can check inspections and generate reports with statistical data inspections.

3.4.5. Product Backlog

The Product Backlog is the device where is listed all user stories are created from obtaining requirements , which are prioritized according to the tasks that must be performed. By this stack we can see a summary of the features and functionality of the system and then classify them as modules to develop.

Then is presented the stack of product obtained to develop the Process Control System Inspection:

PRODUCT BACKLOG		
ID	**Name User History**	**Priority**
1	Create Inspection Schedules	HIGH
2	Assigning architects parishes	HIGH
3	Create Listing Inspection	HIGH
4	Create categories and types of variables in the list of inspection	HIGH
5	Sort procedures to generate inspections	HIGH
6	View details of the procedure	HIGH
7	Send e-mail informing citizens the day of inspection	HIGH
8	Registration Inspections	HIGH
9	Crop Photos inspection	LOW
10	Consult the inspections performed	HIGH
11	Register the personnel responsible for inspection	MEDIUM

12	View Inspection Report done	HIGH
13	Registration Inspections Mobile Device	MEDIUM
14	Reports with Statistical Graphics	MEDIUM

3.4.6. Development of User Stories

After the creation of all user stories, are created iteration stacks where tasks are described needed to comply with the acceptance tests detailed in the stories; these tasks are organized and displayed as chronology. Finally is done capture screens created in the application that supports the tasks assigned to each user story.

CHAPTER IV

4. CONCLUSIONS AND RECOMMENDATIONS

The conclusions and recommendations obtained in the development of the "Process Control System for Inspections" are described in this final chapter.

4.1. CONCLUSIONS

The following conclusions are described based on the objectives established at the beginning of the project:

- By using the system can achieve a more efficient and organized management inspections including its recorded data..

- It was possible to study some of the processes and standards related to the inspections among which is the Plan Approval and Report of Urban Regulation.

- In the Planning Department the type of inspection more performed associated with the Approval of Architectural Plans in which a visit to the property is performed to check whether it meets the requirements and documents submitted to that procedure.

- Through the support of staff "process area" was performed an uprising and organizing of all processes running on the Planning Department where was included inspections and formalities related to it .

- The methodology used for the development of the inspection system is implemented in the "Systems Management GAD-I" which was created by the union and review of 2 agile development methodologies such as: XP and SCRUM.

- For system development was made study and training MVC framework Symfony 1. 4 but for the part of the views are equally studied and used the framework for building RIA applications ExtJS 4.0.

- At the end the of system development was made a presentation in the systems area and planning which stated in detail the operation and use of it; addition to delivering technical and user manuals that support for staff is involved.

- One of the objectives they wanted to accomplish is to inform the public that date will be visited his property for inspection; this could be primarily met through the creation of Agenda Inspections module which identifies and generate inspections creating a route based on the location of the properties on the map and sending emails to citizens at the date assigned to his parish in Inspection Schedules module.

- It was determined that for almost all procedures within the "Planning Department" the inspections are similar so 3 different modules for Approval of Plans, and Miscellaneous Jobs IRU were not created as initially determined.

- Also at submitted the disadvantage of not having clear a document inspection data register was decided to create the sheets Inspections module in which an administrator user can structure a record by process or type of process with the data needed to record.

- Another module that was determined to develop was the User Administration but due to the existence of a scheme in the database GAD-I which records everything related to users, permissions, menus, and other data and well the creation of the Integrated System GAD-I where in you have access to one or more applications based on permissions given to a user, was determined not creating this module.

- After learning more deeply about the inspection process through the lifting process that was undertaken and the implementation of a questionnaire to a part of the architects of Planning Direction are determined the need of create the modules for the Agenda and Registration Inspection.

- An additional module or application is created Registry Inspections from a mobile device which helps reduce paper usage and the time for the citizen to continue or complete the necessary formalities, this is because you have the option of registering data in the same moment to make an inspection to the properties.

- The inspection system has a direct relationship with the Paperwork Management System because it is in which you create a procedure and then the architects select that need inspection that then displayed in the Agenda Inspections module .

4.2. RECOMMENDATIONS

- You could create another module of the system which are created procedures directly with inspection for the use of it in other departments of GAD-I.

- To reduce development time in ExtJS is recommended using an IDE in which to quickly create user interfaces and so the programmer can mostly devoted to the business logic.

- To use the system and ExtJS 4 applications it is recommended to use latest browsers versions for normal execution.

- Because the mobile application is not a native application and runs in a web browser directly depending on the use of internet is recommended to use device data packet to access the same.

- Another recommendation on the mobile application is more research on using PhoneGap whose main characteristic is convert a non-native mobile application as it does Sencha Touch to a native application to install and run on devices with Android or IPhone without relying necessarily the internet.

- Socialize citizens about the new system and sending emails announcing the date of inspection to the property to prevent the accumulation of people in the Planning Department consulting on such information.

- Before starting a software project should be consulted and viewed directly with the people responsible for carrying out the process involved to be clear about what you want to automate and thereby avoid many future changes at the time of development.

REFERENCES

[1] Wikispaces, «Servidor HTTP Apache,» 20 Noviembre 2013. [En línea]. Available: http://conceptos-redes.wikispaces.com/Servidor+Web. [Último acceso: 20 Noviembre 2013].

[2] A. Wikispaces, «Características de Tomcat,» 21 Noviembre 2013. [En línea]. Available: http://apachefoundation.wikispaces.com/Apache+Tomcat. [Último acceso: 21 Noviembre 2013].

[3] R. Martínez, «Sobre PostgreSQL,» 02 Octubre 2010. [En línea]. Available: http://www.postgresql.org.es/sobre_postgresql. [Último acceso: 08 Febrero 2013].

[4] P. Ramsey y M. Martín, «Manual de PostGIS de Paul Ramsey traducido al Español,» 04 Abril 2012. [En línea]. Available: http://postgis.refractions.net/documentation/postgis-spanish.pdf. [Último acceso: 08 Febrero 2013].

[5] M. Sabana Mendoza, PHP 5 con PostgreSQL, Lima: Grupo editorial Megabyte, 2010.

[6] A. Parra, Guía Práctica de JavaScript, Madrid: Anaya Multimedia, 2012.

[7] S. E. «¿Qué es Symfony?,» 18 Octubre 2013. [En línea]. Available: http://www.symfony.es/que-es-symfony. [Último acceso: 18 Octubre 2013].

[8] C. Villa, «Fundamentos de ExtJS,» 15 Noviembre 2013. [En línea]. Available: http://www.quizzpot.com/courses/fundamentos-de-ext-4. [Último acceso: 15 Novimebre 2013].

[9] A. Kosmaczewski, Sencha Touch 2 Up and Running, Sebastopol (California): O'Reilly Media, 2013.

[10] MapServer, «MapServer,» 12 Noviembre 2013. [En línea]. Available: http://live.osgeo.org/es/overview/mapserver_overview.html. [Último acceso: 12 Noviembre 2013].

[11] M. Manso y D. Ballari, «El Archivo Map,» 15 Octubre 2013. [En línea]. Available: http://www.idee.es/resources/presentaciones/JIDEE07/ARTICULOS_JIDEE2007/Articulo1-Anexo-archivo-map.pdf.

[Último acceso: 15 Octubre 2013].

[12] D. Moncayo, «Web Map Services,» 15 Octubre 2013. [En línea]. Available: http://www.dspace.ups.edu.ec/bitstream/123456789/56/11/Capitulo5.pdf. [Último acceso: 15 Octubre 2013].

[13] S. Higuera, «OpenLayers,» 15 Noviembre 2013. [En línea]. Available: http://openlayers.bicimap.es/manualOpenLayers.html. [Último acceso: 15 Noviembre 2013].

[14] GeoExt, «¿Qué es GeoExt2?,» 15 Noviembre 2013. [En línea]. Available: http://geoext.github.io/geoext2. [Último acceso: 15 Noviembre 2013].

[15] C. M. Torres Arcos, Tesis Generación Dinámica de Reportes basado en Ireport y JasperReport, Quito: Escuela Politécnica Nacional, 2010.

[16] MyGnet, «Funcionamiento de JasperReport e iReport,» 17 Octubre 2013. [En línea]. Available: http://mygnet.net/articulos/java/301. [Último acceso: 17 Octubre 2013].

[17] R. Bruno, «Manual de PHP-JRU,» 12 Octubre 2013. [En línea]. Available: http://php-jru.sourceforge.net/docs/manual-PHP-JRU.pdf. [Último acceso: 12 Octubre 2013].

[18] PHP-JRU, «¿Que es PHP/Java Bridge?,» 15 Octubre 2013. [En línea]. Available: ftp://ftp.heanet.ie/disk1/sourceforge/p/ph/php-jru/legacy/manual_de_uso_PHP-JRU.pdf. [Último acceso: 15 Octubre 2013].