eetac Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Content Management System oriented to sports organizations

Title: Content Management System oriented to sports organizations

Author: Juan Manuel Córdoba Hernández

Director: Roc Meseguer

Date: October 8th, 2014

Título: Sistema de Gestión de Contenidos orientado a entidades deportivas

Autor: Juan Manuel Córdoba Hernández

Director: Roc Meseguer

Fecha: 8 de Octubre de 2014

Resumen

Un Sistema de Gestión de Contenidos o por sus siglas en inglés CMS (Content Management System) es una aplicación web que permite que tanto sus administradores como usuarios se preocupen del contenido que publican prácticamente la totalidad del tiempo, despreocupándose del desarrollo y mantenimiento de la propia aplicación.

Una muestra de su popularidad es que en la actualidad el 37,2% de todas las páginas web publicadas en internet utilizan como base una aplicación de estas características.

Este documento analiza el mercado existente de CMS para valorar si las funcionalidades que aportan cumplen los requerimientos de entidades deportivas para gestionar y publicar la información que generan.

Posteriormente se describe el proceso de diseño e implementación del sistema de gestión de contenidos orientado a entidades deportivas adaptado a los requerimientos específicos recojidos de usuarios responsables de entidades reales.

En el documento también se evalúa el sistema de gestión de contenidos tomando como marco el estándar ISO/IEC 25000 ya que especifica de manera objetiva qué elementos de un producto de software pueden ser analizados, así como los métodos para realizar la evaluación y el análisis de los datos obtenidos

Además, se analizan las líneas futuras y se reflexiona acerca del ciclo de desarrollo del Sistema de Gestión de Contenidos orientado a entidades deportivas.

Title: Content Management System oriented to sports organizations

Author: Juan Manuel Córdoba Hernández

Director: Roc Meseguer

Date: October 8th, 2014

Overview

A Content Management System is a web application that allows its administrators and its users to care about the content they publish almost all the time, and not wasting time in the development and maintaining process.

A sign of its popularity today is that 36% of all Internet web pages published use an application of these characteristics as the basis.

This paper analyzes the existing CMS market to rate whether their functionalities fit sports organizations requeriments to manage and publish the information generated.

Then, the process of design and implementation of content management system oriented to sports organizations adapted to the specific requirements of real entities responsible users is described.

The content management system is also evaluated taking as the ISO/IEC 25000 standard as specified objectively what elements of a software product can be analyzed, as well as methods for evaluation and analysis data collected.

Finally, future lines are analyzed and a reflection on the development cycle of the Content Management System oriented to sporting entities is done.

INDEX

CHAPTER 1. INTRODUCTION

As time passes and technology progresses, computer desktop applications, smartphone applications or web applicattions, offer users many options to manage all types of information. But on many occasions and depending on the need of the user, can be halfway between applications that provide very general features and applications that provide very specific functionality.

Today there are three types of applications to transmit or disseminate information trough the internet, depending on the manner and purpose for which they were developed. First type is handmade web applications created by the person that also generates the content, requiring him to have technical knowledge to create and maintain both the application code and manage saved information. Second type is proprietary applications that store information on private company servers without the user worry rather than enter the content (Facebook, Twitter or LinkedIn). The third type of application is Content Management Systems, from now CMS, allowing users to administrate and manage their information in public or private servers, but without requiring too hight technical knowledge.

These two last types of applications are the most popular actually in the internet, having Facebook a number of 1280 million of unic users monthly, having Twitter a number of 255 million of unic users monthly or LInkedIn, that has a number of 300 million of unic users monthly (see [1]). By the other hand, the CMS are the responsibles of hosting 37,2% of all the webs published in the internet, according to w3techs.com (see [2]).

The disadvantage of the use of Facebook, Twitter or LinkedIn (representing proprietary applications) is mainly the use that these companies do with our information, and the number of features available are limited and not expandable by the user by an easy way.

The main advantage of CMS, like Wordpress, Joomla or Moodle, is that they allow administrator users to focus on the original content and not to develop de web application. This is because these applications are prepared to be installed and used. The disadvantages are that these CMS require a minimum technical aknowledgement by the user in order to be able to be installed succesfully in a server, and although they are modifiable and expandable, these applications have basic functionalities.

In the moment of choose between these types of applications, it is very important to think about the use that the hoster of the information is going to do, and the functionalities available.

The present document presents the positive and negative points about using an existing CMS to expand it according to certain requirements and developing a new Content Management System from zero.

Later, a description of the different steps done to develop a prototype of a Content Management System is done.

Finally, the Content Management System is evaluated taking into account the standard ISO/IEC 25000 as objectively describes which elements of a software product can be analyzed, and the methods for evaluation and analysis of the data obtained.

1.1. Motivation

As the person responsable of carrying out the task to manage the information of the "Club de Voleibol Prat de Llobregat", I studied the requirements of the entity to manage and get the most out of the information it had, and also the possibilities that the market offers in the world of the content management systems and propietary applications.

Propietary applications where quickly discarded because they did not allow implementing particular functionalities, like to develop a little application to calculate the classification of a list of teams in a particular league. On the other hand, content management systems partially cover some requirements of the sports entities but not all of them.

The main reason that the information managers of sports organizations in using these tools is that due to limited technical knowledge about configuration web servers, they are forced to use applications that do not offer the specific services they need, limiting itself to write articles about their teams, results, standings and players.

This fact created the dubt about to spend time studing one particular and free to use content management system to develop necessary modules, or to spend time developing a totally new application that feets all the needs of the organization.

That is why it is thought necessary to develop and deliver a product adjusted to the needs of these sports organizations so that, naturally, to manage the information that is easy for them to get, and offer easy and attractive for both the players themselves the club and other visitors who are interested in the club either have an affinity for the sport or be linked to the population in which they are located.

1.2. Objectives

The main objetive of this Project is to develop a Content Management System oriented to Sport organizations as a web application. This application should be installable in any web server that supports PHP language and has a MySQL database engine.

The content management system is the product of the desires of many people in the world of sports entities that do not find out an application that fits their particular needs. The web application itselfs is a collection of different objectives in the form of needs or functionalities.

The ultimate goal of the project is to evaluate the application so that the conclussions serve as a tool to improve the content management system. This goal is achieved through a number of methods listed in ISO/IEC 25000 standard (see [8]).

1.3. Project Organization

With the aim to make easier the reading of this document, it has been divided into five chapters that progressively introduce the subject that is dedicated for. The structure of this document reflects the order that has been taken to design, implement and evaluate the project. The document is structured as follows:

Chapter 2 shows the selection of functionalities desired by the sports entities, the functionalities incorporated inside the most popular Content Managenet Systems in the market, and also a discussion about the chance of using one of the existing CMS or to create a new one.

Chapter 3 makes a brief introduction about the necessary technologies to design and develop the new web application. It is also described the benefits of having chosen the architecture of the application, and finally a little graphical demonstration in screenshots is done.

Chapter 4 presents the external quality and quality in use evaluation of the developed application. This evaluation is carried out taking as a reference the ISO/IEC 25000 standard because it allows stablishing an objective method to evaluate the external quality and quality in use of any application.

Finally, Chapter 5 analyzes the achieved objetives from the development of the project and presents the future enhancements that could be applied.

1.4. Environmental impact

This Project has been developed in the world of internet technologies, so that its direct impact over the environment is minimmum. In order to measure the direct generated environmental impact of this project, the power consumption of the servers that hosts the application should be quantificated. Also the power consumption of the machines that built the devices that host the application should be measured, but this is not the scope of the project.

CHAPTER 2. FUNCTIONALITIES STUDY

This chapter reports the functionalities included in popular applications of content management system, and they are compared with the requirements collected from viewed users, with the objective of assess whether these popular content management systems are valid to satisfy the user needs.

At the end of the chapter brief conclusions to justify the developing of a new prototype of web application to manage contents are exposed, and a collection of implemented modules and functionalities is listed.

2.1. Analysis of popular web applications functionalities

Actually there are many Content Management Systems in the market that have in coomon put in the hands of the web masters tools to facilitate management of their contents. Most of these CMS have in common the management of the content introduced such text, images, videos, registered user information, configuration of the visual aspect through templates or modules to provide extra functionalities.

According to a study published by the consulting pingdom.com (see [3]) at the end of 2013, 60 of the existing 634 million websites in the world were managed with Wordpress CMS, showing the relevance of content managers to administrate websites.

The table 2.1 (see [4]) shows a resume of the percentage of Content Management Systems used by the first 100 blogs in number of visits in 2013 worldwide.

Table 2.1. Market share of the first 100 popular CMS in terms of visits

CMS	CMS Use (%)
Wordpress	52
Custom	12
Drupal	7
N/a	6
Gawker	5
BlogSmith	4
Movable	4
TypePad	4
Blogger	3
Ceros	1
Joomla	1
Tumblr	1

While it is true that the appearance of these web applications served to support all those who wished to publish and manage a journalistic information, it can be seen two massive trends using these applications to support the sector of photography and on the other hand the e-commerce sector.

In the process of searching for a tool that could help a sports club to enter and manage information generated daily, were installed on a server the Content Management Systems: Wordpress, Joomla and Drupal, presuminging that they could be the most would meet the needs of the club.

Below is shown the Table 2.2 that compares the most relevant modules that have been found and its related functionalities in the installed Content Management Systems

Table 2.2. Important features in the three most popular CMS

Module	Functionality	Wordpres	Drupal	Joomla
Installation	Installation of the system	x	x	x
Articles	Create new	x	x	x
	Show list	x	x	x
	Edit article	x	x	x
	Tags	x		
	Category	x		x
	Statistics		x	x
	Search articles		x	x
Comments	Show list	x	x	
	Edit comment	x	x	
Appearance	Internal themes	x	x	x
	External themes	x	x	x
	Internal widgets	x	x	x
	External widgets	x	x	x
	Menu	x	x	x
	Background	x	x	x
	Header	x	x	x
	Footer	x	x	x
Users	Show list	x	x	x
	Create new	x	x	x
	Edit user	x	x	x
	Groups			
	User messages			
Content	Articles	x	x	x
	Pages	x	x	
	Links	x	x	
	Images	x	x	x
	Videos	x	x	x
Privacy	Search engines	x		
Actualizations			x	x
Web editor		x		
Settings		x	x	x
Statistics	Sessions		x	
	Users		x	
	Articles		x	

It is not the objective of the development of the web application prototype to copy exactly all the modules and relate functionalities about the most popular CMS, but it is obvious that an application of this nature should contain strategic areas in order to create articles, manage users of the application, and manage the comments that are submitted by the visitors.

The functionalities that will be developed in the prototype of Content Management System and, therefore they will be similar to the functionalities analyzed in the popular CMS, are those related to the installation, management of the articles, comments, users and content, as seen in table 2.3.

Table 2.3. Features implemented in Wordpress, Drupal and Joomla to be reimplemented

Module	Functionality	Wordpress	Drupal	Joomla
Instalation	Installation of the system	x	x	x
Articles	Create new	x	x	x
	Show list	x	x	x
	Edit article	x	x	x
Comments	Show list	x	x	
	Edit comment	x	x	
Users	Show list	x	x	x
	Create new	x	x	x
	Edit user	x	x	x
Content	Create new	x	x	x
	Images	x	x	x

2.2. Required functionalities

In order to decide about using an existing Content Management System or to develop a new one, a collection of required functionalities was personally obtained doing interviews with each web administrator of sports entities known.

In order of importance a list of modules was done, which in turn grouped functionalities, in order to document them, oppose them with popular CMS and assess whether the presence or absence of these fit the needs of users was made.

Season module: it is used to specify when a season starts and when a season finish, it gives a name to the season and describes the sport federation that regulates the competition. This information is essential because all the information generated by the teams will be related to the season. This module also administrates the categories and the information of all the teams involved into the entity.

Calendar module: it is used to introduce the information relative to the matches such date, hour, localization, teams playing the match and results. This module provides information before the match is played and afterwards it display the classification of the teams involved in the competition.

Chronicles module: it is similar to articles module but focused on the finished matches. It is used to introduce and display information related to finished mathes, being accessible from ech team section or from echa team calendar.

Sports person module: it is used to introduce and display information related to the persons that join the teams of the club such players, coaches or technicians. This information is personal data, postion in the team or track record.

Tournament module: it is used to store and display information related to tournaments organized by the club, being important to make easy to the visitors to sign up their teams in a competition.

Entitie module: it is used to store and display the information that identifies the club and provide contact information like physical address, e-mail address or telephone, in order to present it in an attractive way and easy to understand.

Photo gallery module: it is used to make easy the action of upload photos and create albums by the administrators and also to provide the visualization of these photografies by the users.

Infraestructures module: it is used to introduce and display information relative to the places where the activities of the club are done. These places can be satiums or pavilions, and it can be introduced the locations of the club or the others. The information of this module can be linked with the information of a match inside the calendar to make easy to find the location of any place.

Sponsors module: it is used to register and display the information about all those organizations collaborating in any way with the club, to offer them certain visibility in some places of the Content Management System.

Social networks module: it is used to introduce and display the social networks used by the club. This links can be deletedor added allowing the users to easy access to specific information stored in these social netowrks.

2.3. Comparison between requirements and functionalities

Following table 2.4 shows all the modules that should be implemented in the desired Content Management System and those compared with the modules of the previously analyzed CMS.

Table 2.4. Comparison of Popular CMS modules and modules to develop

Module	Wordpress	Drupal	Joomla	Prototipo
Installation	x	x	x	x
Users	x	x	x	x
Articles	x	x	x	x
Comments	x	x		x
Images	x	x	x	x
Social network links				x
Seasons				x
Calendar				x
Persons				x
Tournaments				x
Club information				x
Photo gallery				x
Chronicles				x
Infraestructure				x
Sponsors				x

As it can be seen in table 2.4, the modules demanded by those responsible for managing information of sports entities are broader than those that provide the analyzed Content Management Systems.

The main reason why this occurs is that made Content Management Systems are oriented to the transmission of general contents and any person wishing to transmit from a newspaper article, a recipe or simply a holiday photographs.

The list of modules defendants justify the creation of a prototype of Content Management System that allows managing data as text and images to create articles, store numerical data to perform operations and automatically calculate classifications, storing or handling photographs itself server.

2.4. Implemented modules and funcionalities

After the analysis of features of the exiting Content Management Systems, collecting functionalities required by users and the comparison between them is possible to show in Table 2.5 the list of features that have been developed and these grouped into modules.

Table 2.5. List of modules and functionalities implemented in the application

Modules	Functionalities
Installation	Install the system
Users	Show, Create, Edit, Delete, List, Validate, Authenticate
Articles	Show, Create, Edit, Delete, List
Comments	Show, Create, Edit, Delete, List
Images	Show album, Show photografies, Create album, Edit album, Add photografies, Delete photografies, List album, List photografies.
Social network links	Show, Create, Edit, Delete, List
Seasons	Show, Create, Edit, Delete, List
Categories	Show, Create, Edit, Delete, List
Teams	Show, Create, Edit, Delete, List
Persons	Show, Create, Edit, Delete, List
Calendar	Show calendar, Show classification, Create, Edit, Delete, List, Calculate classification
Infraestructures	Show, Create, Edit, Delete, List
Tournaments	Show, Create, Edit, Delete, List
Sponsors	Show, Create, Edit, Delete, List
Club information	Show, Create, Edit, Delete, List
Chronicles	Show, Create, Edit, Delete, List

CHAPTER 3. DESIGN AND IMPLEMENTATION

This is the most technical chapter of the Project as a reference to the architecture of the web application and the tools used to develop it.

The first section presents the context in which it has been developed the web application and the resources used to design and implement the project are justified. The second section specifies where the application is available for download and the software architecture that has been used to develop the content management system. Finally the three possible roles of the application are described and screenshots of the content management system as a graphic demonstration of the application are shown.

3.1. Context description

As noted in the introduction to this paper, the main objective of the project is to develop a web application and content management system, so that the context in which it takes place is to web applications hosted on servers accessible by a group of Internet users through.

Programming languages used in this application have been different because each has a specific function. For the visual part of the application has used HTML and CSS, to improve the user interface on the side of the web browser used JavaScript, and managing pages and interaction between the web server and the database has been used PHP.

The database is indispensable for the web application because its aim is to persists in time the data introduced by users. So that, it has been essential the use of a tool like that being accessible from the web application and hosted in the same server in order to store and access information. It is important to know what kind of actions will be done frequently to the database, because it can be a reason to decide the type of database engine. It is also important to know or to predict the number of conections that can be established at the same time in order to design an scalable structure application. The database used in the Content Management System has been MySQL because it has free license to open source projects, it is a relational database that allows making searches linking various tables and it is multiuser, which is a necessary characteristic in order to support as many simultaneous users you might have the application.

One feature that aims to provide the project is that it can be installed on any hardware that supports any operating system that integrates a web server with PHP engine and MySQL databases available, which are widespread on today's technologies.

As evidence of this, the web application has also been tested in Windows environments, demonstrating that the application dependencies are limited to the web server and database.

3.2. Application architecture

The CMS prototype has been built with the Model-View-Controller MVC pattern now on, because it allows a fast and modular development of different functionalities to be coded in the web application.

The MVC architectural pattern was first described in 1979 (see [5]) to Smalltalk and defines the application code must be separated into three independent layers:

- Model: contains the business logic and, in this case, the core of the functionality,
- View: contain the user interfaces or templates,
- Controller: reacts with user requests and executes the appropriate action to create the models and display interfaces.

At the time of developing the functionalities demanded by users has been very useful to follow this pattern because thanks to the Controller it is clear the difference between the Views and the logical part (Model) of the application, allowing more agile changes in any of these layers without affecting the development of the other.

In the example of Figure 3.1 it can be seen the basic example of the MVC pattern, which from the first page (View) a user performs an action to view a second page. The request to see the second page is maintained by the controller, which manages what Model is responsible for providing the following view. Finally, the Model manager displays the following page (View) to the user who requested it.
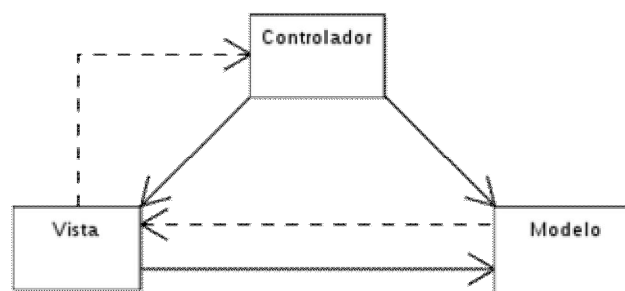
Figure 3.1. Diagram of the Model View Controller pattern

In the prototype of Content Management System developed there are three directories, where can be found the practice representation of this theoretical model.

In the main directory can be found the Controllers, in the object directory can be found the Models and in the views directory can be found the interfaces or Views of all those functionalities developed in the prototype.

It is true that there are not only these directories in the application, but they are the core. It can be found also the javascript directory called "scripts", configuration files or general function files used commonly in Controllers or Models.

An example of a theoretical functionality converted in real code would be the installation functionality of the application (see Figure 3.2):
-    the Controller is represented by the file "install.php",
-    the Model is represented by the file "objetos/class.install.php",
-    the Views are represented by the files inside the directory "vistas/install/".
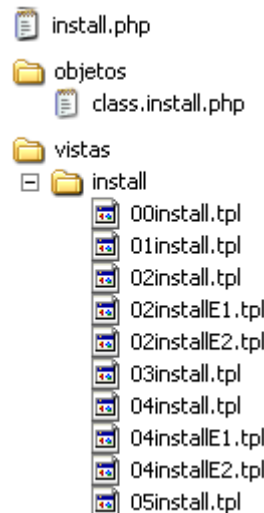
Figure 3.2. Example of installation functionality

Interfaces or views are places where users can make use of the functionality implemented in the web application. As a simple example can be seen in Figure 3.3, where the view from the top bar of the app changes depending on whether the user is accessing visitor, registered or administrator. This is discussed more specifically in section 3.3.
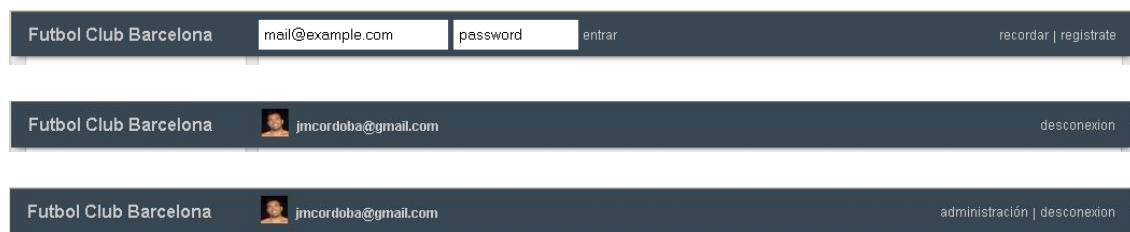
Figure 3.3. Header of the application depending on the user role: visitor, member, manager

Before starting to develop the code of the Models it was taking into account the compiled list of modules in order to initially define the list of functionalities depending on the access permission of the user. Using Model-View-Controller pattern allows the developer to create separate methods as functionalities and encapsulate them into de Modules.

Table 3.1 shows the completive list of functionalities depending on the authorization of the user.

Table 3.1. Functionalities depending on where they are used

	Front view	**Administration menu view**
Users	Create, Validate, Show, Authenticate	Create, Edit, Delete, List
Articles	Show	Create, Edit, Delete, List
Comments	Create, Show	Create, Edit, Delete, List
Images	Show album, Show photografies	Create album, Edit album, Delete album, List album, Create photografies, Edit photografies, Delete photografies, List photografies
Social network links	Show	Create, Edit, Delete, List
Seasons	Show	Create, Edit, Delete, List
Categories	Show	Create, Edit, Delete, List
Teams	Show	Create, Edit, Delete, List
Persons	Show	Create, Edit, Delete, List
Calendar	Show calendar, Show classification	Create, Edit, Delete, List, Calculate classification
Infraestructure	Show	Create, Edit, Delete, List
Tournaments	Show	Create, Edit, Delete, List
Sponsor	Show	Create, Edit, Delete, List
Club information	Show	Create, Edit, Delete, List
Chronicles	Show	Create, Edit, Delete, List

This table is not dedicated just to define the functionalities, but also to define the interfaces of the application that make use of the functionalities developed in the Models. For example, the function devoted to obtain certain items can be used in two different interfaces like Show and List, through the Controller.

3.2.1. Frontal view

Generally from the main interface of the application can only be done readings information stored in the database, except for the comments that can be introduced at the very sight of the items or chronicles if the user is registered, being the only exception in which a non-administrator has the ability to enter information into the database of the application.

As it can be seen in the Figure 3.5, in the list of articles interface does not exist the functionality to introduce a new or to modify one of them, it can only be done a reading.

Figure 3.5. List of articles in the content management system

3.2.2. Administration view

After validating the administrator user it is possible to view items in the administration menu, which are presented in a list of items that have already been written on the application.

Figure 3.6 shows the graphical representation of the developed functionalities for the administration interface in the Articles module. The following functionalities are implemented:

- list of articles,
- create a new articles,
- edit an existing article,
- delete an existing article.
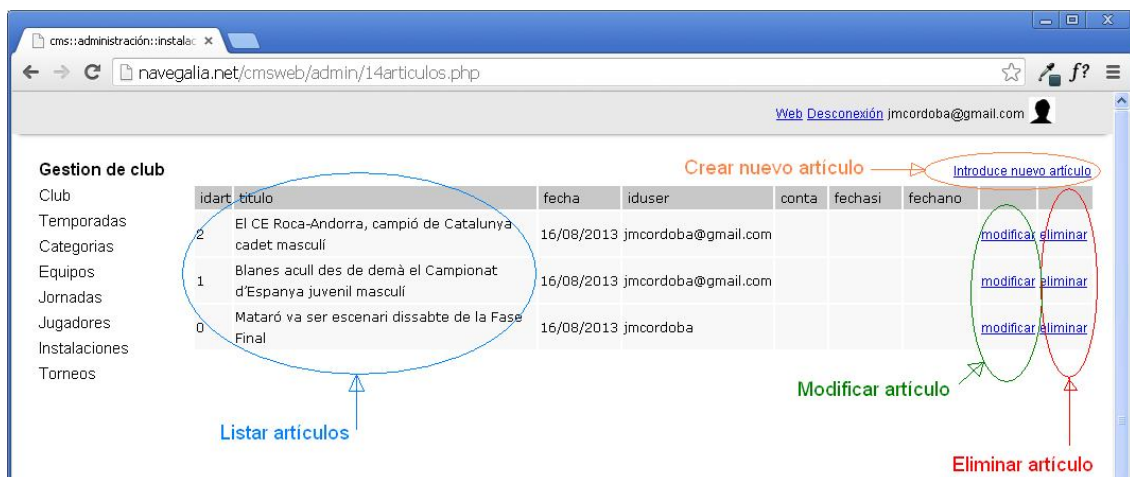
Figure 3.6. Administration menu: list of articles

When the administrator accesses to the creation of a new article it is displayed the interface shown in Figure 3.7, where the information relative to a new article is able to be filled. If a new article is introduced, the reading and writing operation to the database is done through the Model.
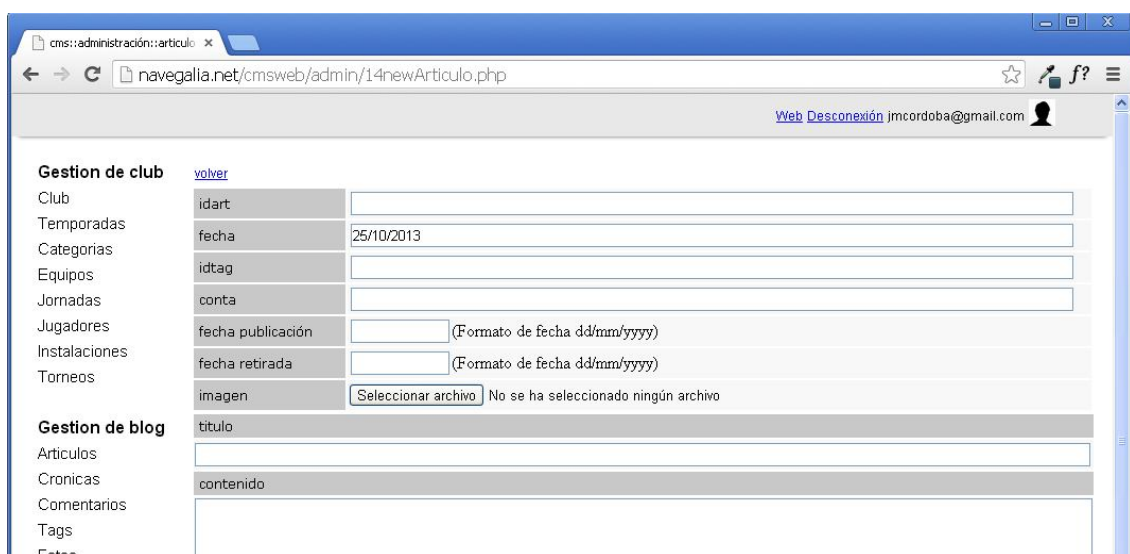
Figure 3.7. Administration menu: new article

3.2.3. Interaction between user and application example

To illustrate the communication between a user and the application, an example is shown as the installation of the application on the web server, so that it is possible to appreciate the succession of different views or through the controller interfaces. In this example it can be seen the separation of files and what relationship exists between them.

The Content Management System web application consists of a set of files compressed in a ZIP file and can be downloaded from the URL http://www.navegalia.net/cmsweb.php.

The minimum requirements for successful installation are:

- Web server capable to precess PHP code,
- MySQL database engine,
- Minimum hard drive space of 10MBytes.

When the application is just downloaded is not functional because it does not have the database conection parameters neither the necessary tables to store the information, so that it is necessary the user to acced from the navigator to the URL http://tu-url.com/install.php in order to initiate the installation, which consists of two basic steps.
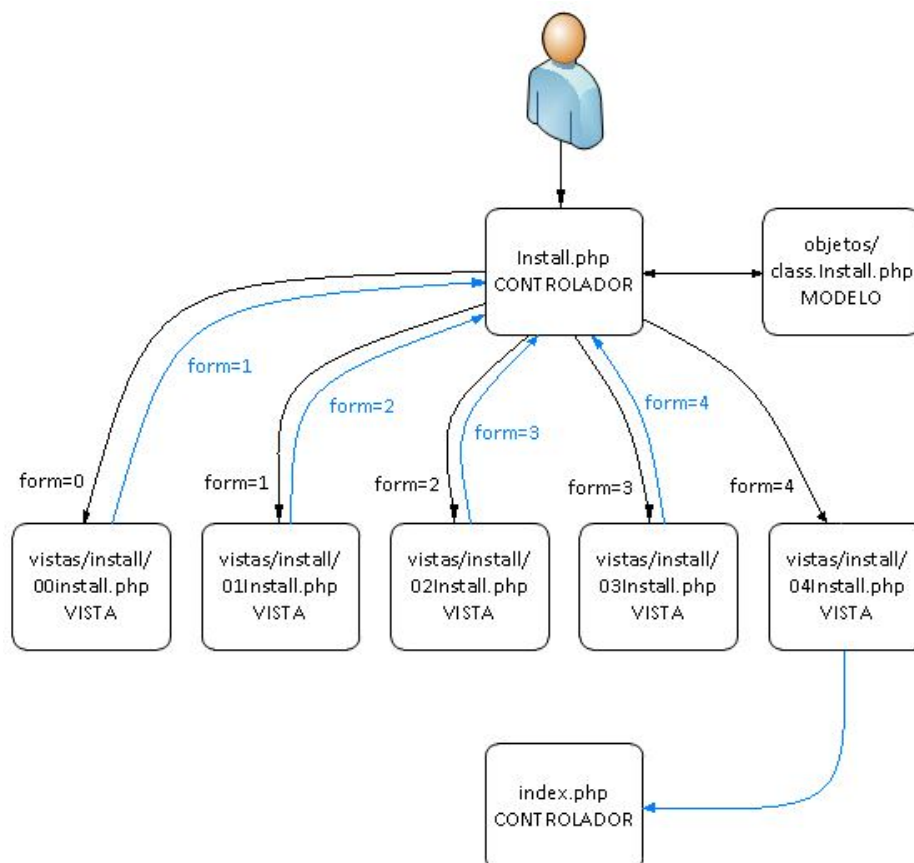
Figure 3.8. File connections in the installation of the web application

In the diagram in Figure 3.8 it can be seen how the user makes use of the file at all times "install.php" which acts as a Controller, which in turn also makes use only of the file "objects / class.install.php" serving as a Model, but interacts with the different views hosted in the "views / install /" directory.

In the first interface "00install.php" a welcome message is displayed and prompts the user to press a button to move to the next view.

In the second step "01install.php" MySQL database conection data is required to the administrator user as it can be seen in Figure 3.9.

Figure 3.9. Configuration dialog of the database connection

In the third step two actions are performed using the Model:
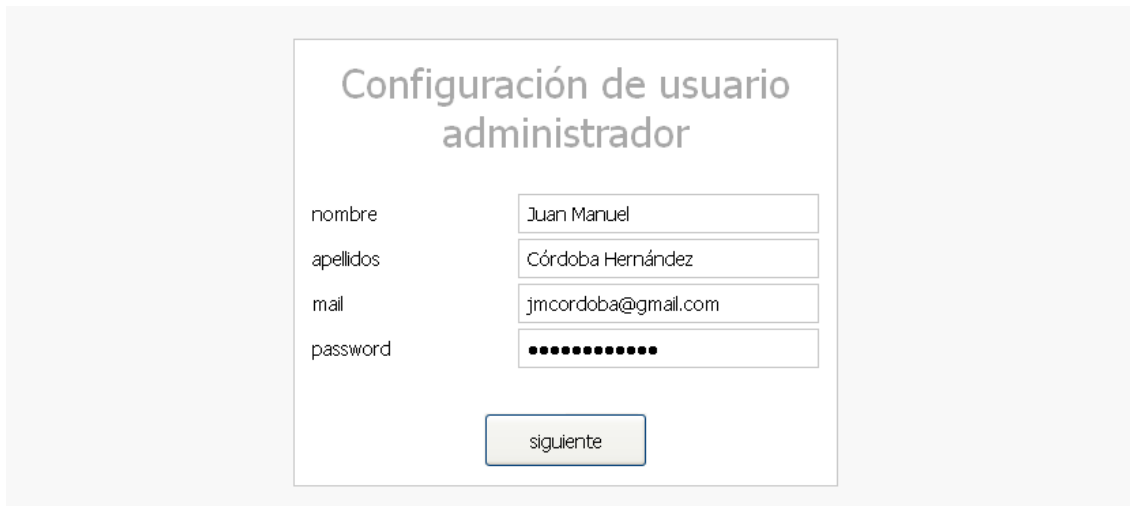
- Necessary tables are created in order to store the information of the Content Management System.
- The file "conf.php" is created ni order to save the data related to the conexion with the database as a source to the application each time this information is required.

If the connection with the database works correctly, "02install.php" interface displays a message confirming that tasks have been performed and invite the user administrator to continue.

The fourth step "03install.php" configures the user administrator (Figure 3.10) and this data is stored in the table intended for it again using the Model.

If everything works correctly you can appreciate the "04install.php" interface which confirms the correct installation of the application and invites the administrator to start using it.

Figure 3.10. Configuration dialog of the administrator user

So far the only person able to manage all the information of the web application is the user introduced in the installation interface, and therefore the unic user with Administrator role.

3.3.   Application roles

In order to control access to the functions of Content Management System is necessary to determine the type of users that will use the application and the range of actions that have enabled.

In the prototype of CMS have been designed three types of users (see Figure 3.11):

- Visitor users with the only possibility of view but not to introduce content,
- Registered users with the additional possibility of introducing comments into the articles and chronicles of the application,
- Administrator users that can do anything possible in the application like managing all the content.
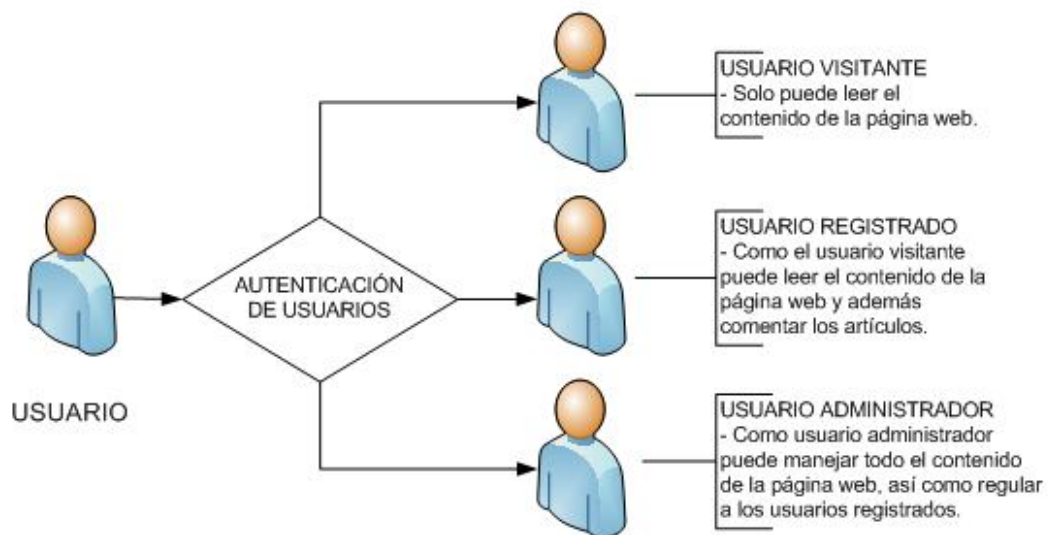
Figure 3.11. Role definition of the content management system

Users have the opportunity to register and authenticate from the header of any of the pages of the CMS, as can be seen in Figure 3.12.

Figure 3.12. Header menú for register or login

3.4. Demo

This section provides a graphic demonstration of what the web application can offer. In Figure 3.13 can be seen the main screen of the application in the case that information has already been introduced to the club.

At the top of the page can bee seen the area of user authentication where entering email and password it is possible to identify the user and expand the type of possible actions within the application.

On the other hand, it is also possible to remember the password or sign up if the user was not yet. For the user to authenticate, could be observed in the upper part of the application image in Figure 3.14, in which the user's photograph, email, link to access the administration menu if user is administrator and the ability to disconnect from the application appears.

Figure 3.13. Main application scren shot

CV Prat   jmcordoba@gmail.com   administración | desconexion

Figure 3.14. Header of the application for administrator user

The rest of the application screen is divided into the left side, the right side and the bottom. The left side serves as a menu of options to redirect all sections of the application as it offers club information, contact area, photo gallery, tournament information and details of the club teams. The right side is dedicated to articles written by the club to communicate with their users. The lower zone is dedicated to show the sponsors that work with the club and specify what the conditions of use of the application as well as a legal notice are.

RED1

PUMA

Condiciones de uso

Aviso legal
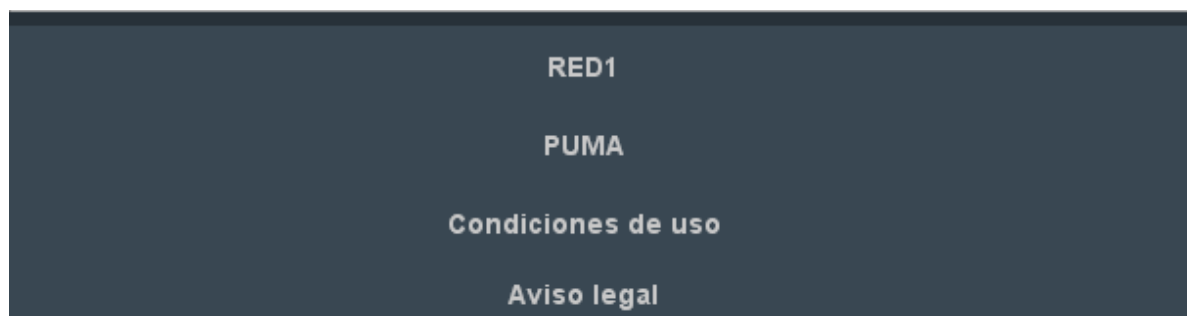
Figure 3.15. Sponsors, general conditions and legal advice

If a user wants to read an entire article of the application, it is possible to do so by clicking the headline of the article and the interface devoted to the entire article would be shown like Figure 3.16.

Figure 3.16. View of an article

In Figure 3.16 it can also be seen as part of the application is virtually the same as it retains header user identification at the top, the menu on the left to access any part of the application and the bottom zone in which it is possible to connect with social networking club or sponsors.

The difference between the last interface and the actual is that meanwhile before there was a list of articles, in the article interface it can be seen the attached image with larger size, the content of the article, the possibility to add a comment if the user is identified and the commentarie related to the article.

In the case that there is a properly configured computer, it is possible to click on the name of the team in left menu application to display the options they have incorporated. As shown in Figure 3.17 these options are:

- Team information,
- Match calendar,
- Match chronicles,
- General classification up to now.

Figure 3.17. Team options view

At this point it is possible to access these four types of specific information for each of the teams that have configured. In figure 3.18 can be seen the interface of a team classification.

Figure 3.18. Team classification view

Just as in the full display of an article, the area that shows the different information needed by the user is the one that is to the right of the menu selection.

All sections of the front of the application follow this design pattern. With this idea it is intended to focus on the content required to be shown, fitting it into a pre-builded frame.

It is possible to view a selection of screenshots from all sections of the front of the application in Annex 1.

On the other hand, an administrator user can access the administration menu by clicking on the link at the top of the application. The screen will appear to be the one shown in Figure 3.19.

Figure 3.19. Administration menu views

All menu screens administration are structured in the same way, at the top there is a small horizontal area to see the picture identifying the user administrator, email the admin user, the option to return to the web and disconnect as administrator option and return to the front of the application.

On the left there is a menu for direct access to manage any application module on the right side and the information that can be created, modified or deleted regarding each module is.

The visible section when entering to the administration menu is the information that brings the club the following attributes: name, president, street, zip code, telephone, fax, email, photography Coat, palmares and history. At the time of saving the information is already possible to access the main view of the application and see the available information to all visitors of the application.

It is possible to view a selection of screenshots from all sections of the administration menu of the application in Annex 2.

CHAPTER 4. EVALUATION

This chapter discusses about the process that has been followed to evaluate the developed application.

First, some methods that allow carrying out the evaluation with standardized parameters to establish objective assessments have been searched. These methods have been found in the standards ISO/IEC 9126, ISO/IEC 14598 and ISO/IEC 25000, being the last one of them the chosen standard. Once justified the decision of the assessment method that has been adopted, it has been designed taking into account the areas of the application to evaluate and the evaluation itself was run. Finally, the results of the evaluation are exposed and commented.

4.1. Evaluation of standards

To establish a mechanism for evaluating the software developed has made a study of various existing standards to date. **To put this section** are listed below the most relevant documents used to support the correct understanding.

- ISO/IEC 9126 entitled "Quality Characteristics and guidelines for Their Use" (see [6]), published in 1991, it is the standard that best defined the characteristics of software products, as well as methodologies for measuring the quality and use of applications.
- ISO/IEC 14598 entitled "Software product evaluation" (see [7]), published in 1999, it is the standard that specifically defines the characteristics and methods of evaluation of a software product.
- ISO/IEC 25000 titulado "Software product Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE" (see [8]), published in 2005.

Currently, the ISO/IEC 9126 and ISO/IEC 14598 standards are still used to be very robust in terms of the concepts in their reviews but are being replaced by the ISO/IEC 25000 standard, since the latter embraces the concepts of the first two standards. The ISO/IEC 25000 standard changes, adds or deletes some concepts or nomenclature to make them more understandable in order to focus more on the product.

For evaluating the prototype web application developed, it has taken as reference the ISO/IEC 25000 standard as it is currently enjoying the support of the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC), and standard that can bring value to the project.

However, in Annexes 3 and 4 summarize the concepts provided by the ISO/IEC 9126 and ISO/IEC 14598 standards because they have been of great help to understand the basis of the evaluation of the application, but slightly beyond the scope of the project have been partially replaced.

4.2. ISO/IEC 25000

The overall goal of creating a catalog of ISO/IEC 25000 standard (see [9]) Software product Quality Requirements and Evaluation (SQuaRE) is moving towards a set of logically organized documents, rich, unified and covers two fundamental aspects: the specification requirements and evaluation of the quality of software, carried by a method of quantifying the quality of software.

The benefits of adopting this standard on the predecessors are:

- Guide to the specification of software quality requirements.
- Coordination between the processes of measurement and evaluation of software quality.

The major differences with the standard predecessors are:

- Introduction of a new model of general reference.
- Introduction of quality measure elements within the measurement section quality.
- Introduction to section quality requirements.
- Introduction and review of assessment processes.

To evaluate a software product using this standard as a tool necessary to make a brief introduction to the model of internal and external quality, the quality model used and the way that the product is evaluated.

Both the model of internal, external quality or use, clearly define aspects of the product in each case can be evaluated. These models also define a typical question and the manner in which the response must be obtained.

4.3. Internal and external quality

Internal quality measures provide the users with the ability to measure the quality of the intermediate deliverables and thereby predict the quality of the final product. The external quality measure is used to measure the behaviour of the system in the environment in which it is intended to operate.

The standard defines the pattern of internal and external quality of the product of eight features, some of which are divided into subcaractics, which refer to static properties of software and dynamic properties of hardware.

The features and subfeatures define consistent terminology to specify, measure and evaluate systems or software products. They also provide a list of quality characteristics that can be fully compared by quality requirements.

Table 4.1. Characteristics of Internal and external quality model

Functional suitability	Performance efficiency	Compatibility	Usability
Functional completeness	Time behavior	Co-existence	Appropriateness recognisability
Functinal correctness	Resource utilisation	Interoperability	Learnability
Functional appropriateness	Capacity		Operability
			User error protection
			User interface aesthetics
			Accessibility

Reliability	Security	Maintainability	Portability
Maturity	Confidentiality	Modularity	Adaptability
Availability	Integrity	Reusability	Installability
Fault tolerance	Non-repudiation	Analysability	Replaceability
Recoverability	Accountability	Modifiability	
	Authenticity	Testability	

In the Annex 5 is possible to extend the information on the characteristics of internal and external quality.

4.4. Quality in use

According to the standard, the definition of quality in use is the ability of a software product to provide specific users to achieve specific goals with effectiveness, productivity, safety and satisfaction in a specific context. The quality model used is composed of a number of characteristics, sub-characteristics and metrics to help understand the concept of quality in use.

Table 4.2. Characteristics of quality in use model

Effectiveness	Efficiency	Security	Satisfaction
Task completion	Time efficiency	Risk mitigation	Satisfaction scale
Task effectiveness	Relative task time	Return of investment (ROI)	Satisfaction questionnaire
Error frequency	Task efficiency	Time to achieve a return of investment ROI	Discretional usage
	Relative task efficiency	Relative business performance	Discretional utilization of functions
	Economic productivity	Balanced score cards	Customer complaints
	Productive proportion	Delivery time	Trust scale
	Relative number of user actions	Missing items	Pleasure scale
		Revenue for each customer	Comfort scale
		Errors with economic consequences	
		Software corruption	
		User health and safety frequency	
		User health and safety impact	
		Safety of people affected by use of the system	
		Environmental impact	

In the Annex 6 is possible to extend the information on the characteristics of quality in use.

4.5. Evaluation process

The aim of the evaluation process is:

- list the parts of the product and the features that will be evaluated,
- list the features that will be measures and its metrics,
- design the method to be used and the score to be awarded to quantify the quality of these features
- specify the range of scores under which it is acceptable or not each of the characteristics,
- and make conclusions about the results of the evaluated product.

Table 4.3. Evaluation process activities

Evaluation process
Establish the requirements of the evaluation
Specify the evaluation
Design the evaluation
Execute the evaluation
Conclude the evaluation

In the Annex 7 there is more information related to the evaluation process of an application.

In an effort to give credence to the results of the evaluation is necessary that the following are met:

- Repeatability: the repeated evaluation of the same component for the same specification delivers results understood as identical even if by different people.
- Reproducibility: the evaluation of a component can be played with the same specifications and provide results almost identical understood even if by different people.
- Fairness: that the evaluation result can not be biased towards a particular outcome.
- Objectivity: the results provided by the evaluation are specific and not the result of an open or subjective reasoning.

4.6.   Quality software evaluation

Up to this point have been defined three types of possible qualities to be evaluated in a software product: internal quality, external quality and quality in use.

As seen in the table 4.4, the internal quality of the software is related to the architecture of the project, source code and design, while the external quality and quality in use is related to the final product, that is, with the product that has been requested by users.

Table 4.4. Type of qualities as requirements

Requirements	Type of qualities to measure
Project	
Source code	Internal Quality
Design	
Product	Quality in use and external quality

Since the initial objective of the project is to meet the requirements of a group of users, software evaluation will focus on the external quality and quality in use.

Specification of the evaluation:

-   The objective of the evaluation is to determine the valuation of the project for its potential users.
-   The profile of users that will be used to interact with the software developed and will be observed to obtain the results, is a user that manages computers routinely for tasks of drafting documents or web browsing, with some knowledge of web servers.
-   The application can be successful if the total score is below 60%, will be satisfactory if it is between 60% and 90% and will be excellen-te if it exceeds 90%.

The product being evaluated is a web application that must be installed on a web server and it is usable using a web browser.

It has created a virtual machine with the following features for each user to have the ideal environment: Operating System Ubuntu 12.04 LTS 32bit desktop, a core CPU, 20 GB HDD, 1GB RAM, Apache 2.2.24 web server, database data server Mysql 5.4.17, PHP 5.4 and Mozilla Firefox 26.0 web browser.

4.6.1. Modules and metrics of external quality evaluation

In Table 4.5 the features and subfeatures external quality to be evaluated, and the maximum score are described, in order to have all combined to form a unic score that subsequently be weighted with quality in use.

The data related to subfeatures efficiency in performance are obtained using the JMeter program, while data regarding to the other features are obtained by observing users evaluating the application.

Table 4.5. External quality characteristics and subcharacteristics

Features	Sub-feature	Points
Functional suitability	Functional completeness	10%
Performance efficiency	Response time	10%
Performance efficiency	CPU utilization	10%
Performance efficiency	Memory utilization	10%
Performance efficiency	Max number of online requests	10%
Performance efficiency	Max number of simultaneous access	10%
Performance efficiency	Max Bandwidth of transmission system	10%
Security	Access controllability	10%
Security	Authentication methods	10%
Realiability	Script and SQL injection	10%

In Table 4.6 is specified how the final score for each sub-feature are obtained, and it will be show if each feature is unsatisfaction, satisfaction or excellence.

Table 4.6. Valuation method of external quality

Features	Sub-feature	Method
Functional suitability	Functional completeness	x functionalities developed successfully / 16 required functionalities = score
Performance efficiency	Response time	(1/x response time in seconds) = score Less than a second is maximum score
Performance efficiency	CPU utilization	1-% CPU utilization = score
Performance efficiency	Memory utilization	1-% Memory utilization = score
Performance efficiency	Max number of online requests	(x request / 900)*100 = score Maximum score is 900 request per second or more.
Performance efficiency	Max number of simultaneous access	(x access / 15)*100 = score Maximum score is 15 simultaneous access or more.
Performance efficiency	Max Bandwidth of transmission system	0 kbps = Max score 100 Mbps = Min score 1-((BW(kbps) *100)/100Mbps) = score
Security	Access controllability	1 - (not allowed access pages / allowed access) = score
Security	Authentication methods	1 - (non authenticated access / authenticated access) = score
Realiability	Script and SQL injection	1 - (successfully attacked pages / attacked pages) = score

4.6.1.1. External quality evaluation results

After doing the test of external quality assessment of the application the data compiled in Table 4.7 are collected and have a total weighted assessment of 71.7% of the points.

In the Annex 8 several graphs of all the metrics corresponding to the measures in Table 4.7 are exposed and obtained with different applications.

Table 4.7. External quality evaluation results

Features	Sub-features	Result	Score
Functional suitability	Functional completeness	100%	10%
Performance efficiency	Response time	1 page (147kB) in 147 ms	10%
Performance efficiency	CPU utilization	98,7%	1%
Performance efficiency	Memory utilization	360 MB de 1536 MB	8%
Performance efficiency	Max number of online requests	529 pet /60 sec	5,9%
Performance efficiency	Max number of simultaneous access	8'8 req/seg	5,9%
Performance efficiency	Max Bandwidth of transmission system	212 kbit/s	9,9%
Security	Access controllability	100%	10%
Security	Authentication methods	100%	10%
Realiability	Script and SQL injection	100%	10%
TOTAL			**71,7%**

In the first instance this assessment serves to focus on those aspects which have obtained a lower score and thus return to rethink those technical aspects that have helped to develop the functionalities. But it also lets you put the application in the proper place depending on the environment in which it is established, as there are metrics that depend not only on the application itself, but the hardware and operating system that makes it work.

Finally, conducting an external quality assessment of an application, it is possible to obtain objective arguments to explain the overall application behavior along with other essential elements for its operation is obtained.

4.6.2. Modules and metrics of quality in use evaluation

In Table 4.8 the tasks to be performed are described in terms of the user profile. Some tasks for assessing the quality in use can be performed as a guest user application but most of them must be run as administrator user.

Table 4.8. Tasks to evaluate quality in use

Guest user	Administrator
Registration	Installation of the application
Validation	Create a new article including an image
Introduction of a comment in an article	Introduction of a new social network link
	Introduction of a new sponsor
	Introduction of a new tournament
	Creation of a new image gallery
	Introduction of a new image in the new created gallery
	Introduction of a new season
	Introduction of a new category
	Introduction of a new team with an image
	Introduction of a new player with an image
	Introduction of a new installation

To define the evaluation of quality in use is necessary to indicate the features and subfeatures to be evaluated in each task and what weight will be given. This information is summarized briefly in Table 4.9.

Table 4.9. Quality in use characteristics and subcharacteristics

Feature	Sub-feature	Score
Effectiveness	Task completion	25%
Effectiveness	Task effectiveness	25%
Efficiency	Task efficiency	25%
Satisfaction	Satisfaction scale	25%
TOTAL		100%

In the Table 4.10 may be found the method that has been used to assess sub-features in each of the tasks performed in Table 4.9. All the scores of each user will be essential to obtain a single global score of the application developed.

Table 4.10. Quality in use evaluation method

Característica	Subcaracterística	Valoración
Effectiveness	Task completion	X=A/B; where A is the number of completed tasks and B the number of started tasks.
Effectiveness	Task effectiveness	X=1-sum(Ai); where Ai are the errors produced in the subtasks of a task.
Productivity	Task efficiency	X= Task efficiency/Time completion of a task.
Satisfaction	Satisfaction scale	X=A/B; where A is the result of the questionnaire and B the average of the other questionnaire results.

To get the result of the evaluation of the first five points, each user was watched while they performed the assigned tasks and finally they were contrasted directly asking if they were sure of completing the task.

To get the result of the satisfaction scale was used a type of questionnaire that can be found in Annex 9. The questionnaire is called SUS for its acronym in English System Usability Scale (see [10]) and was originally created by John Brooke in 1986, allowing evaluating a set of products and services including hardware, software, websites and applications. This questionnaire attempts to measure the feelings that the user perceives by using the application and that, being within subjectivity is relevant at the time to be asking the same questions the evaluators users.

4.6.2.1. Quality in use evaluation results

After doing the evaluation test about quality in use of the application the scores are compiled in Table 4.11 and have a total weighted value of 83% of the points.

Table 4.11. Quality in use evaluation results

Evaluator	Task completion	Task effectiveness	Task efficiency	Satisfaction scale	Total
Nº 1	100%	100%	70,60%	54%	81,15%
Nº 2	100%	100%	82,66%	50%	83,17%
Nº 3	100%	100%	84,66%	70%	88,67%
Nº 4	100%	100%	72,40%	46%	79,60%
Nº 5	100%	100%	74%	52%	81,50%
Nº 6	100%	100%	79%	66%	86,15%
TOTAL	**100%**	**100%**	**77%**	**56%**	**83%**

The first thing that can be notice of the assessment is that the first two subfeatures evaluated have a 100% rating as they are strictly objective tasks that relate to compliance of each task, while the last two sub-features evaluated are kind of subjectivity, within parameters, and therefore have different ratings depending on the user performing the test.

It is very interesting to evaluate quality in use of the application to see that each user uses the application differently and consequently have different perceptions of the quality of the application.

While the questions in this assessment are made to obtain valuations of certain tasks and are grouped according to the type of quality to be evaluated, it should be noted that some of these valuations are at a given moment, and in these cases would interesting to conduct several assessments over a considerable period of time to observe the evolution of the scores.

It is certainly a valuable tool used to modify the application to visual and functional level in order to increase the quality in use.

4.7. Quality evaluation global results

Analyzing the scores as a result of external quality evaluation and a quality in use evaluation, it has been obtained the scores of 71.7% and 83% respectively.

At this point it must remember the range of results that had been proposed before running the test designed for the application, which was unsatisfactory to lower scores of 60%, satisfactory for scores between 60% and 90%, and excellent for ratings above 90%.

Thus the result of both external quality evaluation and the quality in use evaluation is satisfactory, because they are between 60% and 90%.

It could be valuable the option to make an average between the results of these qualities evaluated, but no sense getting an average score of 60% (satisfaction mark) as a result of the average of 40% and 80%. This could be unacceptable if the application does not work optimally, for example because it consumes a lot of resources of the server where you are staying, even if the perception of the user was excellent.

In any case, the ultimate goal of such assessments is not only to obtain a valuation but use it to analyze the areas evaluated continuously in order to improve them. These improvements can be both at the technical and aesthetic level performance, but more interesting is that the overall quality of an application is a whole made up of different pieces that thanks to the standard ISO/IEC 25000 can be evaluated objectively.

CHAPTER 5. CONCLUSIONS

This final chapter presents the results that have been achieved during the development of the Content Management System. Furthermore, a list of future engancements is shown, identifying the work that remains to do. Finally, some personal conclussions are exposed as a result of the feelings developing the application and writing this document.

5.1. Objectives achieved

This document has presented a prototype of Content Management System oriented to Sports Organizations, as a result of developing a web application according to certain requirements. These requirements, listed in chapter 2, were the starting point of the project and where the basis to develop an application installable and usable by any user who has the suitable desktop environment, also listed in chapter 3.

Another objective, although it is intengible, is to satisfy in terms of gratifying, a group of persons that did not locate a web application that feet their meets in terms of functionalities.

Finally, a technicall method to evaluate a web application has been designed and applied taking into account the subjective methods that makes the application to offer different feelings depending on the persons that use it. The objective has been achieved by studying the international standard ISO/IEC 25000 that provides robustness to the results and gives us value to improve and evolving the application itself.

5.2. Future enhancements

The prototype of content management system can be improved from several areas:

- Software architecture: Once the web application has been developed by writing the totality of the code, it would be natural to integrate an existing Framework, like Zend (see [11]) or Symfony (see [12]), in order to let them to administrate the repetitive tasks and give the software developer to focus on create new functionalities faster.

- Search Engine Optimization (SEO): is the way of making the content more visible to the search engines like Google, Yahoo or Bing. It is possible by coding the content of the application using a collection of standardized tags (see [13]).

- Hardware optimization: it is highly recommended to optimize the code of the application to proffit fully the hardware resources and increase the system performance. The code can be optimized reducing it to the minimum necessary and using the functions given by PHP that

consumes fewer resources. It is possible to increase the system performance writing javascript code that uses the client navigator to show content. It is also possible to increase the system performance using a cache system to process the requests that are equal, and instead of process all of them, this system process the first one and respond to the others with a copy of the first response.

- Functionalities: it is posible to develop more usefull functionalities in the content management system, like a newsletter module to advice about the next event, tournament o photografy álbum to the registered users. It is also possible to develop an online shop to sell merchandising or tickets.

- Documentation: it would be interesting to create a handbook or guide for the users that download the web application to help them to resolve the typical issues that could appear.

- Evaluation: enlarge the ítems to be evaluated envolves to invest more time doing tests and analyzing their results, but they improve the product, so that the evaluators point out the flaws and virtues of the application.

5.3. Personal conclusions

The realization of this project has allowed to me to explore all the areas of the development cycle of a web application:

- The emergence of a need.
- Requirements gathering.
- Design of functionalities.
- Develop an application including these functionalities.
- Develop an application in order to be installable and easy using.
- Evaluate the developed application with the same users that suggested the requirements.

Each one of the steps of this project has required for me to use different capabilities such listen to the persons/users when they explained to me ther needs, but also it was important for me to make the right answer in order to make sure to develop the accurate functionality to solve their issues.

It has been necessary to plan and design the structure of the web application in order to develop it in an efficient way and in the estimated time.

Finally it has been necessary to use an international standard to be able to design an evaluation method and then run it to obtain objective technical results and also subjective results like the perception of the quality in use, which was unknown for me before starting with this project.

The overall rating of the Project is very positive as they have met all the initial objectives.

ANNEX 1. FRONTAL VIEW SCREENSHOTS

1.1. WELCOME SCREEN APPLICATION

Figure A1.1. Welcome screen application

1.2. CLUB'S INFORMATION SCREEN

Figure A1.2. Club's information screen

1.3. INFRAESTRUCTURE LOCATION SCREEN

Figure A1.3. Infraestructure location screen

1.4. CONTACT SCREEN

Figure A1.4. Contact screen

1.5. PHOTO GALLERY SCREEN
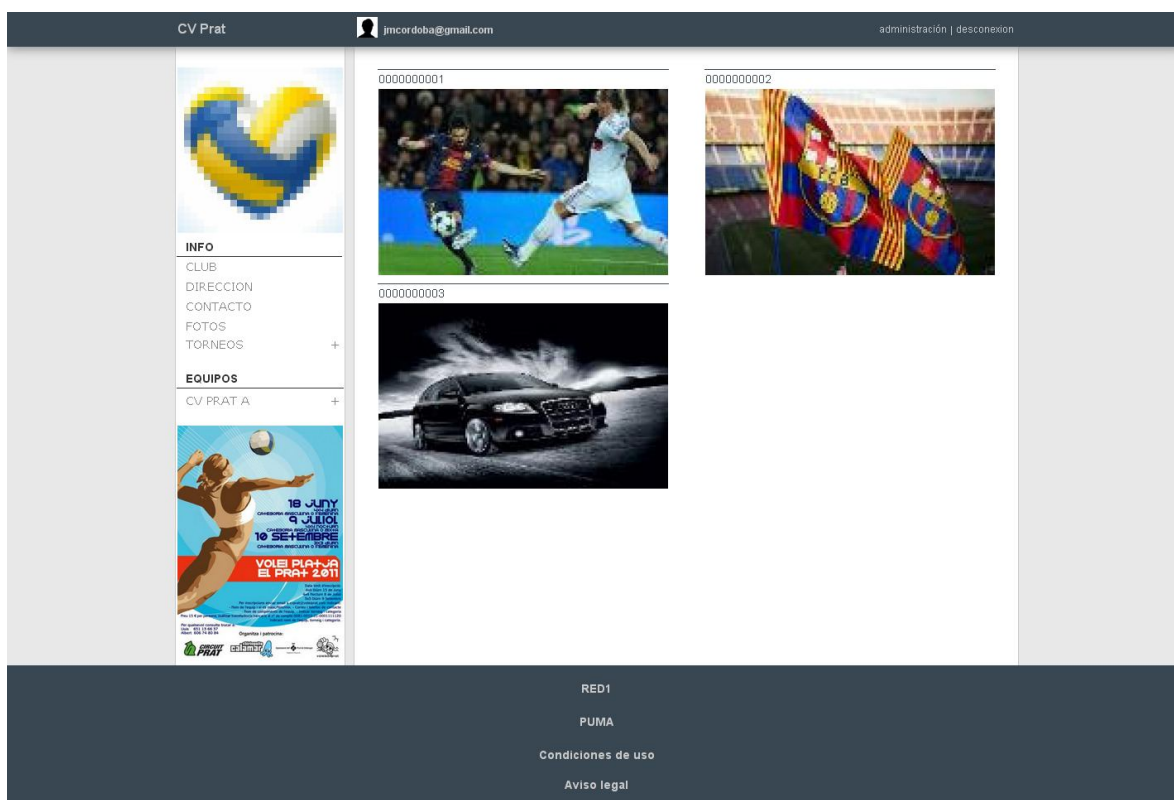
Figure A1.5. Photo gallery screen

1.6. TOURNAMENTS INFORMATION SCREEN
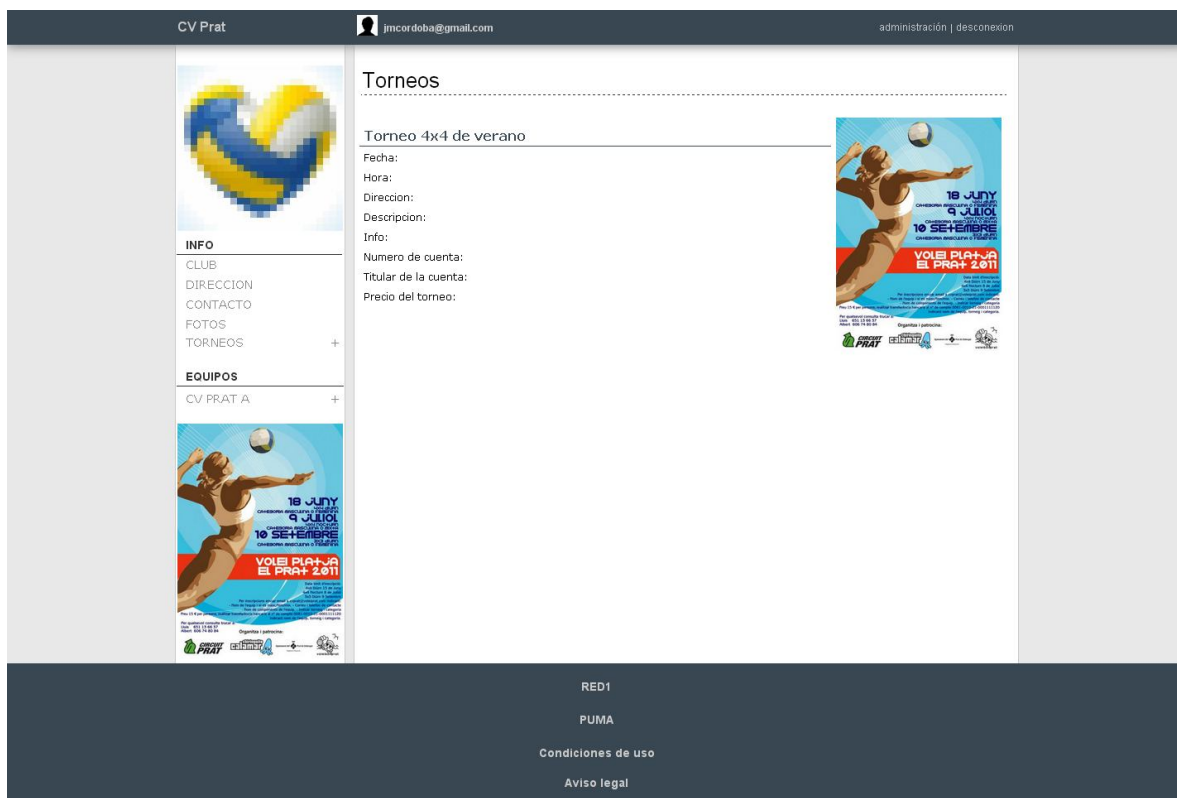
Figure A1.6. Tournaments information screen

1.7. TEAM INFORMATION SCREEN
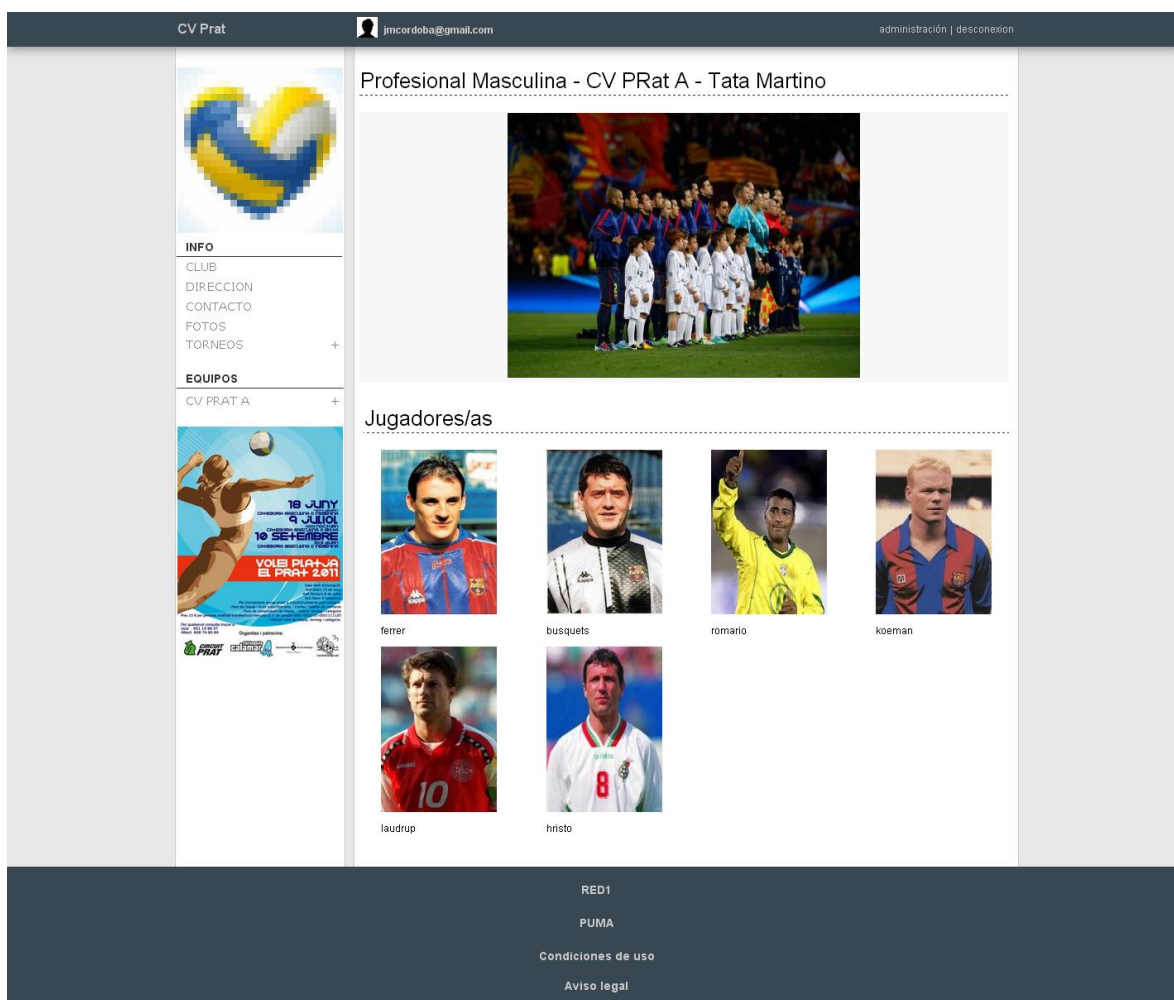
Figure A1.7. Team information screen

1.8. TEAM CALENADR SCREEN

Figure A1.8. Team calendar screen

1.9. TEAM CHRONICLES SCREEN
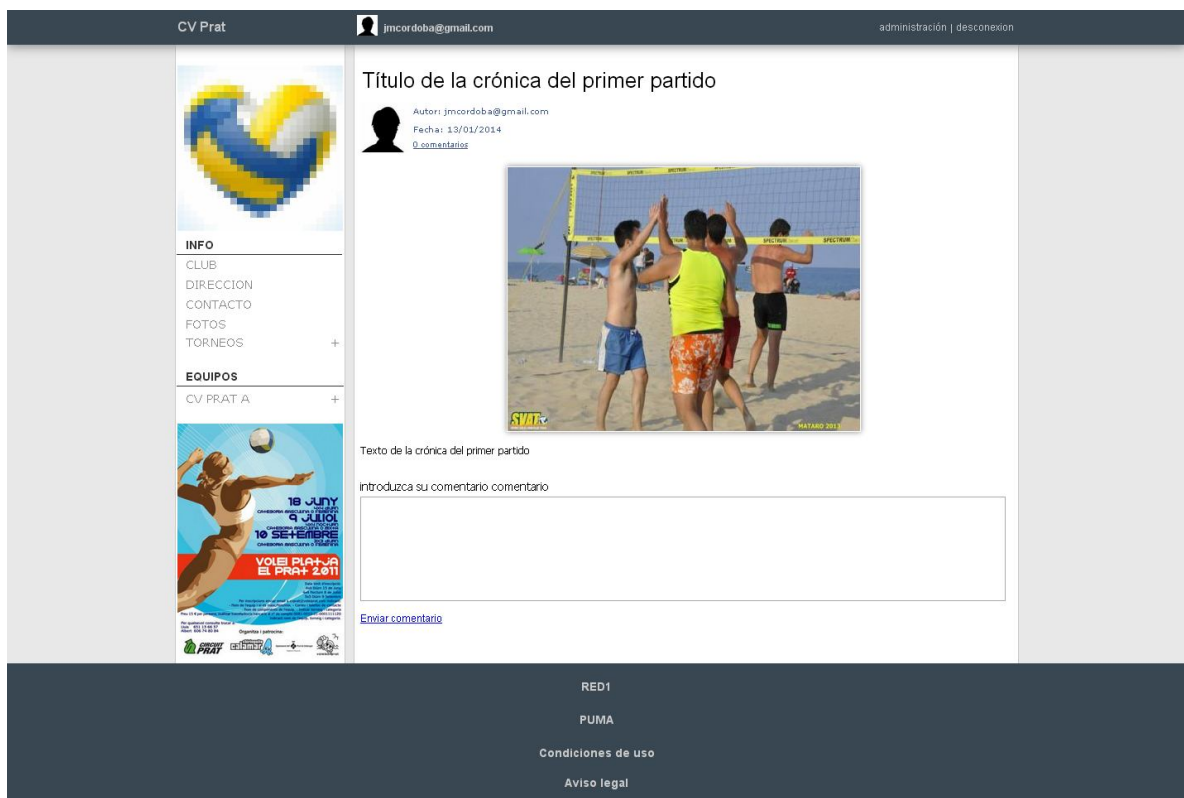
Figure A1.9. Team chronicles screen

1.10. TEAM CLASSIFICATION SCREEN

Figure A1.10. Team classification screen

ANNEX 2. ADMINISTRATION VIEW SCREENSHOTS

2.1. TEAM INFORMATION EDITION SCREEN

Figure A2.1. Team information edition screen

2.2. WELCOME APLICATION EDITION SCREEN

Figure A2.2. Welcome application edition screen

2.3. SEASON CREATION/EDITION SCREEN

Figure A2.3. Season creation/edition screen

2.4. CATEGORIES WELCOME SCREEN

Figure A2.4. Categories welcome screen

2.5. CATEGORIES CREATION/EDITION SCREEN

Figure A2.5. Categories creation/edition screen

2.6. TEAM ADMINISTRATION WELCOME SCREEN

Gestion de club   Introduce nuevo equipo
Club
Temporadas
Categorias
Equipos
Jornadas
Jugadores
Instalaciones
Torneos

idteam	idcat	nombre	mister	foto	conta		
1	1	CV PRat A	Tata Martino	admin/images/equipo/team1.gif		modificar	eliminar

Gestion de blog
Articulos
Cronicas
Comentarios
Tags
Fotos

Gestion de web
Formularios
Redes sociales
Usuarios
Patrocinadores
Volver a la web

Figure A2.6. Team administration welcome screen

2.7. TEAM CREATION/EDITION SCREEN

Figure A2.7. Team creation/edition screen

2.8. TEAM CALENDAR WELCOME SCREEN

			Web Desconexión jmcordoba@gmail.com		
Gestion de club	categoria	equipo	j. totales	j. disputadas	editar jornadas
Club	Profesional Masculina	CV PRat A	6	2 de 6	editar
Temporadas					
Categorias					
Equipos					
Jornadas					
Jugadores					
Instalaciones					
Torneos					
Gestion de blog					
Articulos					
Cronicas					
Comentarios					
Tags					
Fotos					
Gestion de web					
Formularios					
Redes sociales					
Usuarios					
Patrocinadores					
Volver a la web					

Figure A2.8. Team calendar welcome screen

2.9. CALENDAR CREATION/EDITION SCREEN

Figure A2.9. Calendar creation/edition screen

2.10. ADMINISTRATION OF PLAYERS WELCOME SCREEN

Figure A2.10. Administration of players welcome screen

2.11. PLAYERS CREATION/EDITION SCREEN

Figure A2.11. Players creation/edition screen

2.12. INFRASTRUCTURE ADMINISTRATION WELCOME SCREEN

Web Desconexión jmcordoba@gmail.com

Gestion de club

Introduce nueva instalación

Club

idinst	name	direccion	telefono	mail	persona	foto		
1	Pabellón Júlio Méndez	Carrer del Riu Guadalquivir, 1-13 08820 Prat de Llobregat	934 78 76 60	info@cvprat.com	Júlio Méndez	admin/images /instalaciones /insta1.gif	modificar	eliminar

Temporadas
Categorias
Equipos
Jornadas
Jugadores
Instalaciones
Torneos

Gestion de blog
Articulos
Cronicas
Comentarios
Tags
Fotos

Gestion de web
Formularios
Redes sociales
Usuarios
Patrocinadores
Volver a la web

Figure A2.12. Infraestructures administration welcome screen

2.13. INFRASTRUCTURE CREATION/EDITION SCREEN

Figure A2.13. Infraestructure creation/edition screen

2.14. TOURNAMENT ADMINISTRATION WELCOME SCREEN

Figure A2.14. Tournament administration welcome screen

2.15. TOURNAMENT CREATION/EDITION SCREEN

Figure A2.15. Tournament creation/edition screen

2.16. ARTICLES ADMINISTRATION WELCOME SCREEN

Figure A2.16. Articles administration welcome screen

2.17. ARTICLES CREATION/EDITION SCREEN

Figure A2.17. Articles creation/edition screen

2.18. CHRONICLES ADMINISTRATION WELCOME SCREEN

Figure A2.18. Chronicles administration welcome screen

2.19. CHRONICLES CREATION/EDITION SCREEN

Figure A2.19. Chronicles creation/edition screen

2.20. COMMENTS ADMINISTRATION WELCOME SCREEN

Figure A2.20. Comments administration welcome screen

2.21. COMMENTS CREATION/EDITION SCREEN

Figure A2.21. Comments creation/edition screen

2.22. PHOTO GALLERY ADMINISTRATION WELCOME SCREEN

Figure A2.22. Photo gallery administration welcome screen

2.23. PHOTO GALLERY CREATION/EDITION SCREEN
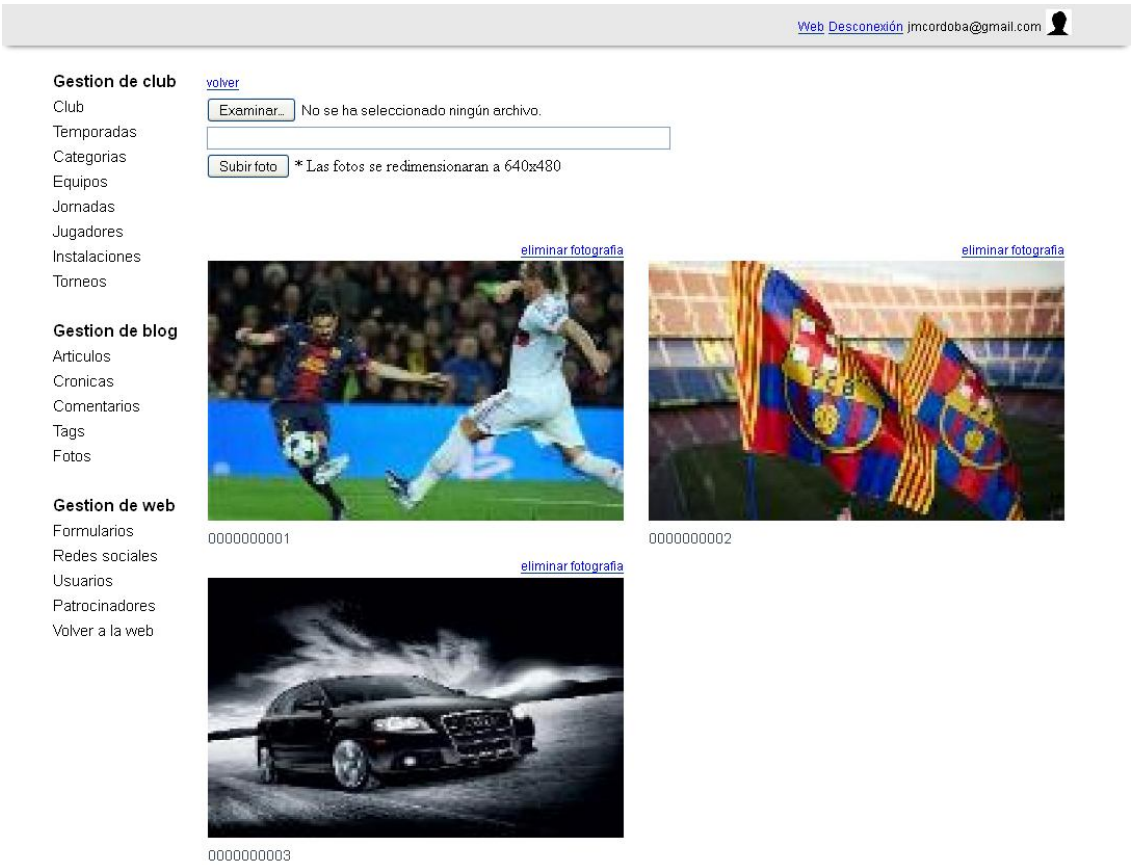
Figure A2.23. Photo gallery creation/edition screen

2.24. SOCIAL NETWORK LINKS ADMINISTRATION WELCOME SCREEN

Figure A2.24. Social network links administration welcome screen

2.25. SOCIAL NETWORK LINKS CREATION/EDITION SCREEN

Figure A2.25. Social network links creation/edition screen

2.26. USER ADMINISTRATION WELCOME SCREEN

Figure A2.26. User administration welcome screen

2.27. USER CREATION/EDITION SCREEN

Figure A2.27. User creation/edition screen

2.28. SPONSOR ADMINISTRATION WELCOME SCREEN

Figure A2.28. Sponsor administration welcome screen

2.29. SPONSOR CREATION/EDITION SCREEN

Figure A2.29. Sponsor creation/edition screen

ANNEX 3. ISO/IEC 9126

3.1. Introduction

As software applications have grown in number, so has the importance of the quality of the softare, technological specifications and the assessment of both the quality of software as a product and process quality objectively and qualitative development.

However, a framework is needed to provide a set of tools to assess the quality of software. This international standard was produced as one of the documents which form the framework.

3.2. Objective

This international standard defines six characteristics that describe software quality. These features provide a base to provide greater understanding and description of the software quality. These guidelines describe the use of quality characteristics for evaluating software quality.

This standard does not provide sub-features, metrics and methods to measure, compare or evaluate. This standard is adhered to the definition of quality in the standard ISO 8402. In the Annex A are defininen a list of sub-features.

The definition of the characteristics and process models associated quality assessment in this International Standard are applicable when the requirements for the assessment of software quality are specified during their life cycle.

Its features should be applied to all kinds of software, including computer programs and data stored in firmware. This standard is oriented to data acquisition, development, support, maintenance and auditing software.

3.3. Quality software characteristics

The quality of software can be evaluated under the following characteristics:

- Functionality: A set of attributes that bear on the existence of a set of functions and their specified properties. The functions are those that satisfy stated or implied needs.
  - Suitability.
  - Accuracy.
  - Interoperability.
  - Security.
  - Functional compliance.

- Reliability: A set of attributes that bear on the capability of software to maintain its level of performance under stated conditions for a stated period of time.
  - Maturity.
  - Fault tolerance.
  - Recoverability.
  - Reliability compliance.

- Usability: A set of attributes that bear on the effort needed for use, and on the individual assessment of such use, by a stated or implied set of users.
  - Understandability.
  - Learnability.
  - Operability.
  - Attractiveness.
  - Usability compliance.

- Efficiency: A set of attributes that bear on the relationship between the level of performance of the software and the amount of resources used, under stated conditions.
  - Time behaviour.
  - Resource utilization.
  - Efficiency compliance.

- Maintainability: A set of attributes that bear on the effort needed to make specified modifications.
  - Analyzability.
  - Changeability.
  - Stability.
  - Testability.
  - Maintainability compliance.

- Portability: A set of attributes that bear on the ability of software to be transferred from one environment to another.
  - Adaptability.
  - Installability.
  - Co-Existence.
  - Replaceability.
  - Portability compliance.

3.4. Guidelines for the use of quality characteristics

In order to evaluate all the characteristics of quality of a software product, it shold be followed the next guidelines:

- Define the requirements of quality of a product.
- Evaluate the software specifications for compliance with quality requirements during the development process.
- Describe the characteristics and attributes of the software.
- Evaluate the software developed after delivery.
- Evaluate the software developed after acceptance.

3.5. Types of evaluators of software quality

There are three different actors or points of view of the developed software:

- The user point of view:
    o The standard ISO/IEC 8402 defines the product quality from the point of view of the user.
    o Users are interested in the use of the software, its behavior and the effects of using the software.
    o Users evaluate the software without knowing the internal aspects of it and how it has developed.
    o Some questions to make to the users could be:
        ▪ Are this functions available in the software?
        ▪ How reliable is this software?
        ▪ How efficient is this software?
        ▪ Is this software easy to use?
        ▪ How easy is to move this software to another environment?

- The developer point of view:
    o The development process requires user and developer agree on the features to meet the software because the user is defining the requirements and finally acceptance.
    o Because the developer is responsible for the final quality of the product, must maintain the quality of the internal aspects of the software not relevant for the user without compromising the quality of the features in which a user is fixed.

- The manager point of view:
    o There should be more interested in the overall quality of software in a particular characteristic, defining priorities according to quality criteria and financial.
    o Must balance with software quality with management approaches such as time spent, cost of specific functionalities, in order to optimize quality at a given cost, human resources and delivery times.

3.6. Internal and external software quality

The definition of internal quality given in the ISO/IEC 9126-1 standard is given by "all attributes of a product determines its ability to satisfy stated and implied needs when used under specified conditions".

Internal quality is specified by a quality model (similar to the model 9126), and can be measured and evaluated using static attributes of documents such as:
- Specification of requirements.
- Architecture or design.
- Source code pieces.

In early stages of the software lifecycle is possible to measure, evaluate and control the internal quality of these products, but internal quality assurance is not generally sufficient to ensure external quality.

The definition of external quality is "the degree to which a product meets explicit and implied needs when used under specified conditions".

The external quality is also specified by a quality model (similar to the model 9126), and can be measured and evaluated by dynamic properties of the executable code in a computer system, that is, when a module or the entire application is executed on a computer or network by simulating as closely as possible a real environment.

In later phases of the software life cycle (mainly in various stages of testing or operational state and a software product or Web application), it is possible to measure, evaluate and control the quality of these external executables products.

3.7. Quality in use

Quality in use as a model to model different and complementary quality software product quality is specified.

Table 4.2. Quality in use

Quality in use
Effectiveness
Productivity
Security
Satisfaction

Quality in use is the combined effect of the six characteristics that define the internal and external quality.

3.8. Reviews

The document does not specify any punctuation for defined features and sub-features, but suggests that the external evaluator of the software give a numerical scale of values on each characteristic evaluated score. This way, you can translate subjective assessments of a particular group of users to a score that will eventually be the valuation of each feature of the software product.

It also defines the possible external evaluators of a software product could be: proprietary software, an auditor, developer, maintainer, supplier, user or quality manager.

The standard indicates generally how to measure quality in use of a software product. Literally defines that quality in use metrics measure the degree to which a product meets the needs of a specific user, to achieve specific goals with effectiveness, productivity, safety and satisfaction and quality needs of users can be specified as quality requirements through quality in use metrics, external metrics, and sometimes by internal metrics. These requirements are specified as metrics should be used as evaluation criteria when the product is evaluated.

Considered for evaluation three groups or categories of evaluation methods: Testing, Inspection and Consulting, each of which includes the possibility of using a set of specific methods or techniques, for example:

-   Thinking Aloud Protocol,
-   Heuristic Evaluation and Registration of Real Use (Logging Actual Use)
-   Surveys,
-   Inspection of Physical Characteristics.

ANNEX 4. ISO/IEC 14598

The ISO/IEC 14598 standard deals with the evaluation of software quality and no longer considers the dissociated quality of user needs. This quality in use can be evaluated according to effectiveness, productivity and satisfaction with which a specific kind of user can achieve specific goals in a particular work environment.

In the context of a process of quality assurance of a software product quality attributes (internal and external) should be the means, and quality in use to achieve the goal.

The standard ISO/IEC 14598 also defines the user requirements can be expressed as a set of desirable for Measurement and Quality Assessment in Web Applications Using requirements. Among the models evaluation process is important to refer to ISO/IEC 14598-5, which was originally included in ISO/IEC 9126.

This standard is oriented to developers, possible buyers of the software and independent evaluators. The results of the assessments can be used by managers, developers or maintainers in order to compare them with the initial requirements and implement changes or updates to the product. They can also be used by analysts to establish relations between measures of internal and external quality.

Following is a brief description of the six documents of this standard:

- ISO/IEC 14598-1 General view: provides a summary of the other five stages, explains the relationship between product evaluation and software quality model. Activities: (Establish evaluation requirements, specify the evaluation, the evaluation Plan, Execute evaluation).

- ISO/IEC 14598-2 Planning and managing: contains requirements and guidelines for support functions such as planning and evaluation management software product. Activities: (Preparation of policies, definition of objectives, identification technology, allocation of responsibilities, evaluation of software developed and purchased).

- ISO/IEC 14598-3 Process of developers: it is used by organizations planning to develop a product or improve an existing one, performs product evaluations using indicators that can predict the quality of the final products. Activities: (Organizing, Planning, Specification, Design, Installation).

- ISO/IEC 14598-4 Process of comparators: it is used by organizations seeking to compare or refuse a product of existing software, is applied for the purpose of acceptance of a product. Activities: (Requirements, Specification evaluation, design evaluation, performance evaluations).

- ISO/IEC 14598-5 Process of evaluators: this process is used by organizations responsible for assessing, provides requirements and guidelines for evaluating the software product. Promotes the following process (repeatable, Reproducible, Impartial, Objective) Activities: (Traceability, Results, Problems, Improvements, Conclusions).

- ISO/IEC 14598-6 Evaluation module: specifies the measurements will be taken on the quality attributes that were defined in the previous step, provides guidelines for the assessment documentation. Activities: (Introduction, Scope, Tickets, Results).

The standard ISO/IEC 14598 provides a working frame to evaluate the quality of all type of software, stating the requirements to be measured and analyzed in this preccess. The standard also provides tools to ensure proper assessment and mitigating software errors that can happen during the execution.

ANNEX 5. ISO/IEC 25000 – INTERNAL AND EXTERNAL QUALITY

5.1. Functional adequacy

This feature represents the ability of the software product to provide functions which meet stated and implied needs when the product is used under specified conditions. This feature is further subdivided into the following sub-features:

- Functional completeness.
- Functional correctness.
- Computacional accuracy.
- Functional appropriateness.

5.2. Performance efficiency

This feature represents the performance relative to the amount of resources used under certain conditions. This feature is further subdivided into the following sub-features:

- Time behavior measures:

    - Response time.
    - Turnaround time.
    - Throughput.

- Resource utilization:

    - CPU utilization.
    - Memory utilization.
    - I/O devices utilization.

- Capacity measures.

    - Max. number of online requests can be processed per unit of time.
    - Max. number of simultaneous accesses at a certain time.
    - Max. Bandwidth that the system can transmit.

5.3. Compatibility

This feature represents the capability of two or more systems or components to exchange information and/or perform their required functions when they share the same hardware or software environment. This feature is further subdivided into the following sub-features:

- Co-existence.
- Interoperability:

- Connectivity with external systems.
- Data exchangeability.

5.4. Usability

This feature is the capacity of the software product to be understood, learned, used and be attractive to the user, when used under certain conditions. This feature is further subdivided into the following sub-features:

- Appropriateness recognisability:

  - Description completeness.
  - Demonstration capability.

- Learnability.
- Operability:

  - Operational consistency.
  - Message clarity.
  - Customizing possibility.

- User error protection:

  - Input validity checking.
  - Avoidance of incorrect operation.

- User interface aesthetics.
- Accessibility.

5.5. Realiability

This feature is the ability of a system or component to perform the specified functions, when used under conditions and for certain time period. This feature is further subdivided into the following sub-features:

- Maturity:

  - Fault removal.
  - Test coverage.
  - Mean time between failures.

- Availability:

  - Service time ratio.
  - Mean down time.

- Fault tolerance:

  - Failure avoidance.
  - Redundancy.

- Recoverability.

5.6. Security

This feature is the ability to protect information and data so that unauthorized persons or they can not read or modify systems. This feature is further subdivided into the following sub-features:

- Confidentiality:

    - Access controllability.
    - Data encryption.

- Integrity.
- Non-repudiation.
- Access auditability.
- Authenticity

5.7. Maintainability

This feature represents the ability of the software product to be modified effectively and efficiently due to developmental, corrective or perfective needs. This feature is further subdivided into the following sub-features:

- Modularity.

- Reusability.

- Analysability:

    - Audit trail capability.

    - Diagnosis function sufficiency.

- Modifiability:

    - Localization degree of correction impact.

    - Modification complexity.

- Testability:

    - Functional completeness of embedded test functions.

    - Autonomous testability.

    - Test restartability.

5.8. Portability

This feature represents the ability of the product or component being transferred effectively and efficiently as a hardware environment, software, operational use or another. This feature is further subdivided into the following subfeatures:

- Adaptability:

  - Hardware environmental adaptability.

  - System software environmental adaptability.

  - Organisational environmental adaptability.

- Installability:

  - Installation time efficiency.

  - Ease of installation.

- Replaceability:

  - User support function consistency.

  - Functional inclusiveness.

  - Continous usage of data.

ANNEX 6. ISO/IEC 25000 – QUALITY IN USE

6.1. Effectiveness

Software's capability to allow users to obtain specific goals specified precisely and completely in the context of software.

- Task completion.
- Task effectiveness.
- Error frequency.

6.2. Productivity

Feature of the software to be used at full capacity, in relation to the effectiveness achieved in a specified context of use.

- Time efficiency.
- Relative task time.
- Task efficiency.
- Relative task efficiency.
- Economic productivity.
- Productive proportion.
- Relative number of user actions.

6.3. Security

Capability of the software product to achieve acceptable levels of risk in harmming persons, the business or the software.

- Risk mitigation.
- Return Of Investment (ROI).
- Relative business performance.
- Balanced Score Card.
- Delivery time.
- Missing items.
- Errors with economic consequences.
- Software corruption.
- User health and safety frequency.
- User health and safety impact.
- Safety of people affected by use of the system.
- Environmental Impact.

6.4.   Satisfaction

Software's capability to satisfy users in a context of use.

- Satisfaction scale.
- Satisfaction questionaire.
- Discretional usage.
- Discretional utilization of functions.
- Proportion of customer complaints.
- Trust scale.
- Comfort scale.

ANNEX 7. EVALUATION PROCESS

7.1. First step

The first step of the evaluation process is to establish the requirements of the assessment:

- Establish the objective of the assessment.
- Obtain the quality requirements of the product.
- Identify the parts of the product to be evaluated.
- Define the rigor of the evaluation.

7.2. Second step

In the second step of the evaluation process evaluation modules (composed of metrics, measurement techniques and tools) and decision criteria to be applied in the evaluation are specified.

- Select assessment modules.
- Define the decision criteria for metrics
- Define the decision criteria of evaluation.

7.3. Third step

In the third step of the evaluation process the assessment is designed: in this activity plan assessment activities to be performed is defined.

- Plan assessment activities.
- Assessment activities taking into account the availability of resources should be planned.
- The evaluation plan should be reviewed and updated.

7.4. Fourth step

In the fourth step the evaluation is executed: in this step assessment activities are executed obtaining quality metrics and applying the evaluation criteria.

- Perform meassurements.
- Apply decision criteria for metrics.
- In the latter task should apply the decision criteria in terms of quality characteristics and sub-characteristics.

7.5. Fifth step

The last step is to complete the evaluation: in this activity evaluation of software product quality is concluded, making the score report to be delivered to the client and reviewing the results obtained with it.

- Review the results of the evaluation.
- Create the evaluation report.
- Review the quality of the evaluation and obtain feedback.
- Manage the evaluation data.
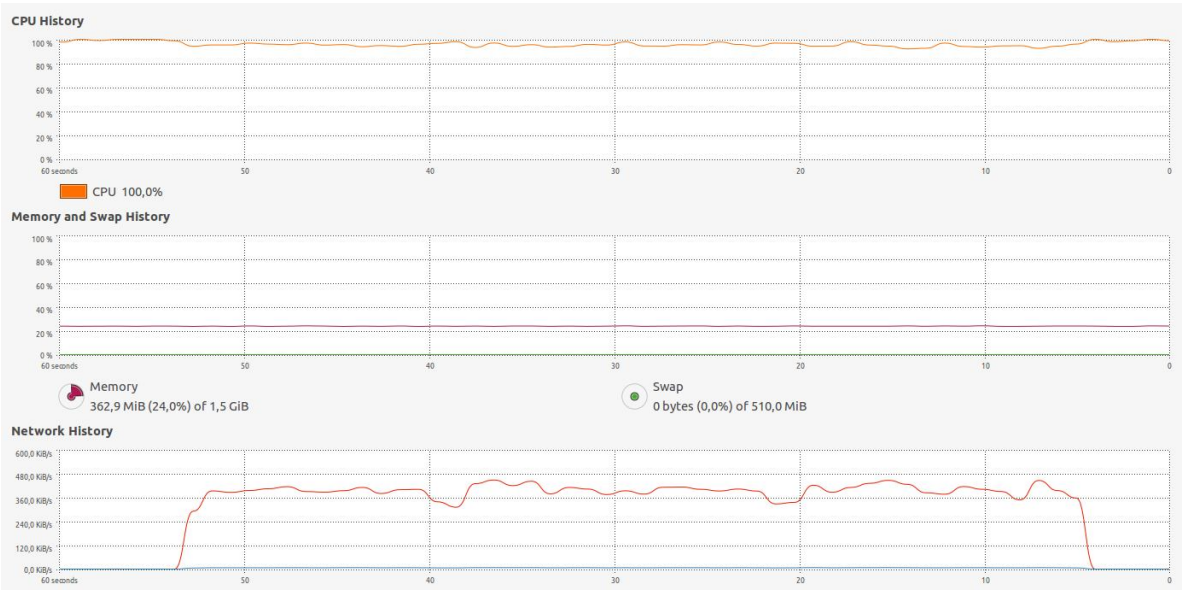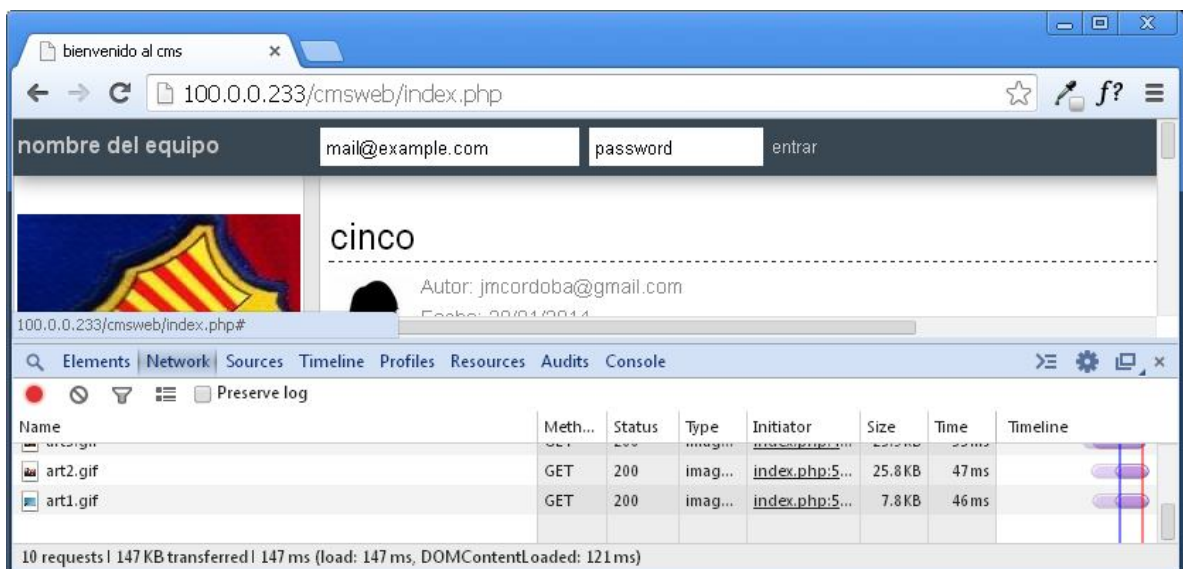
ANNEX 8. EXTERNAL QUALITY MEASURES

Figure A8.1. Graph server statistics during a stress test with Ubuntu System Monitor

Figure A8.2. Graph server statistics during a stress test with JMeter

Figure A8.3. Time to load the main page of the application

ANNEX 9. SYSTEM USABILITY SCALE SURVEY

Strongly disagree   Strongly agree

1. I think that I would like to use this system frequently

1   2   3   4   5

2. I found the system unnecessarily complex

1   2   3   4   5

3. I thought the system was easy to use

1   2   3   4   5

4. I think that I would need the support of a technical person to be able to use this system

1   2   3   4   5

5. I found the various functions in this system were well integrated

1   2   3   4   5

6. I thought there was too much inconsistency in this system

1   2   3   4   5

7. I would imagine that most people would learn to use this system very quickly

1   2   3   4   5

8. I found the system very cumbersome to use

1   2   3   4   5

9. I felt very confident using the system

1   2   3   4   5

10. I needed to learn a lot of things before I could get going with this system

1   2   3   4   5

Figure A9.1. SUS Survey

BIBLIOGRAPHY

[1] Social network statistics:
Facebook statistics: http://newsroom.fb.com/company-info/
Twitter statistics: https://about.twitter.com/company
Linkedin statistics: http://press.linkedin.com/about

[2] w3techs.com:
http://w3techs.com/technologies/overview/content_management/all

[3] pingdom.com:
http://royal.pingdom.com/2013/01/16/internet-2012-in-numbers/

[4] pingdom.com:
http://royal.pingdom.com/2013/05/07/wordpress-top-100-blogs/

[5] MVC
http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html
http://www.amazon.es/Pro-PHP-MVC-Professional-Apress/dp/1430241640

[6] ISO/IEC 9216:
http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=22749

[7] ISO/IEC 14598:
http://www.iso.org/iso/iso_catalogue/catalogue_ics/catalogue_detail_ics.htm?csnumber=24902

[8] ISO/IEC 2500:
http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=35733

[9] ISO/IEC 2500:
http://iso25000.com/

[10] System Usability Scale:
http://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

[11] Zend Framework:
http://www.zend.com/

[12] Symfony Framework:
http://symfony.com/

[13] Google SEO guide:
http://static.googleusercontent.com/media/www.google.es/es/es/webmasters/docs/guia_optimizacion_motores_busqueda.pdf