Universitat Politècnica de Catalunya

Barcelona School of Informatics

Development of a web tool for editing heraldry shields

Memory

Bachelor's Degree in Informatics Engineering (Information Systems)

Author: Xavier Ferrer Costa

Director: Enric Mayol

Date: 24/11/2014
Resum

Aquest document és el Treball de Final de Grau fet per en Xavier Ferrer Costa i dirigit per l’Enric Mayol Sarroca.

El projecte consisteix en una eina web que permetrà als seus usuaris la creació d'escuts seguint la ciència de l'Heràldica. Aquesta és la ciència de la història que estudia específicament l'ús sistemàtic d'emblemes hereditaris plasmats sobre un escut d'armes, la seva composició i el seu significat. L'usuari a part de poder crear escuts podrà compartir experiències, coneixements i opinions amb altres usuaris, creant així una petita xarxa social.

El propòsit d'aquest projecte consisteix en dotar als amants de l'Heràldica de dos espais integrats que actualment no existeixen a la nostra Societat.

Summary

This document is the final Bachelor's degree project done by Xavier Ferrer Costa and directed by Enric Mayol Sarroca.

The project is a web tool that allows users to create their shields according to the science of Heraldry. This is the science of the history that studies specifically the systematic use of hereditary emblems reflected on a coat of arms, its composition and its meaning. The user can create shields and share experiences, knowledge and opinions with other users, creating a small network.

The purpose of this project is to provide to lovers of Heraldry two integrated areas that currently do not exist in our society.
Index

1 Introduction .................................................................................................................. 11
  1.1 Motivation .................................................................................................................. 11
  1.2 Formulation of the problem ...................................................................................... 12
  1.3 Goals .......................................................................................................................... 13
  1.4 Technical competences ............................................................................................ 13

2 Research ....................................................................................................................... 14
  2.1 Heraldry ........................................................................................................................ 14
    2.1.2 Composition ........................................................................................................... 14
    2.1.3 Parts of the shield ................................................................................................. 15
    2.1.4 Enamels ................................................................................................................ 15
    2.1.5 Divisions ............................................................................................................... 16
    2.1.6 Basic pieces .......................................................................................................... 17
    2.1.7 Figures .................................................................................................................. 18
    2.1.8 Crowns ................................................................................................................ 20
  2.2 Existing applications .................................................................................................. 21
    2.2.1 Examples .............................................................................................................. 21
    2.2.2 Existing applications conclusions ...................................................................... 22

3 Context ........................................................................................................................ 23
  3.1 Actors involved .......................................................................................................... 23
    3.1.1 Developer .............................................................................................................. 23
    3.1.2 Project director ................................................................................................... 23
    3.1.3 Final users ........................................................................................................... 23

4 Scope ............................................................................................................................. 24
  4.1 Overview .................................................................................................................... 24
  4.2 Goals ........................................................................................................................ 24
  4.3 Risks .......................................................................................................................... 24

5 Methodology ................................................................................................................ 27

6 Project plan ................................................................................................................... 29
  6.1 Project stages ............................................................................................................ 29
    6.1.1 Project planning ................................................................................................... 29
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1.2 Analysis and design of the project</td>
<td>29</td>
</tr>
<tr>
<td>6.1.3 Iterations project</td>
<td>30</td>
</tr>
<tr>
<td>6.1.4 Completion</td>
<td>31</td>
</tr>
<tr>
<td>7 Planification and budget</td>
<td>32</td>
</tr>
<tr>
<td>7.1 Estimated time</td>
<td>32</td>
</tr>
<tr>
<td>7.2 Budget</td>
<td>34</td>
</tr>
<tr>
<td>7.2.1 Human resources</td>
<td>34</td>
</tr>
<tr>
<td>7.2.2 Material resources</td>
<td>34</td>
</tr>
<tr>
<td>7.2.3 Budget summary</td>
<td>35</td>
</tr>
<tr>
<td>8 Sustainability i social engagement</td>
<td>36</td>
</tr>
<tr>
<td>9 Analysis and specification</td>
<td>37</td>
</tr>
<tr>
<td>9.1 System Architecture</td>
<td>37</td>
</tr>
<tr>
<td>9.2 Requirements analysis</td>
<td>39</td>
</tr>
<tr>
<td>9.2.1 Functional requirements</td>
<td>39</td>
</tr>
<tr>
<td>9.2.2 Non functionals requirements</td>
<td>40</td>
</tr>
<tr>
<td>9.3 Use cases</td>
<td>41</td>
</tr>
<tr>
<td>9.4 Database model</td>
<td>42</td>
</tr>
<tr>
<td>9.5 Storyboards</td>
<td>46</td>
</tr>
<tr>
<td>9.5.1 Web</td>
<td>46</td>
</tr>
<tr>
<td>9.5.2 Tool</td>
<td>52</td>
</tr>
<tr>
<td>10 Development System</td>
<td>54</td>
</tr>
<tr>
<td>10.1 Preparation of the environment</td>
<td>54</td>
</tr>
<tr>
<td>10.2 Joomla template configuration</td>
<td>55</td>
</tr>
<tr>
<td>10.2.1 Installation and configuration modules</td>
<td>57</td>
</tr>
<tr>
<td>10.2.2 Menus configuration</td>
<td>58</td>
</tr>
<tr>
<td>10.3 Creation of tables</td>
<td>61</td>
</tr>
<tr>
<td>10.4 Personal shields</td>
<td>63</td>
</tr>
<tr>
<td>10.5 Like and Comment</td>
<td>70</td>
</tr>
<tr>
<td>10.5.1 Like</td>
<td>70</td>
</tr>
<tr>
<td>10.5.2 Like consultation</td>
<td>73</td>
</tr>
<tr>
<td>10.5.3 Comment</td>
<td>77</td>
</tr>
<tr>
<td>10.6 Publics shields</td>
<td>85</td>
</tr>
<tr>
<td>10.7 Profile</td>
<td>88</td>
</tr>
<tr>
<td>10.8 Catalan Heraldry</td>
<td>91</td>
</tr>
<tr>
<td>Illustration</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>1</td>
<td>Parts of the shield</td>
</tr>
<tr>
<td>2</td>
<td>Blazon</td>
</tr>
<tr>
<td>3</td>
<td>Puncher</td>
</tr>
<tr>
<td>4</td>
<td>Inkwell Ideas</td>
</tr>
<tr>
<td>5</td>
<td>Wampserver logo</td>
</tr>
<tr>
<td>6</td>
<td>Joomla logo</td>
</tr>
<tr>
<td>7</td>
<td>Drupal logo</td>
</tr>
<tr>
<td>8</td>
<td>Wordpress logo</td>
</tr>
<tr>
<td>9</td>
<td>How CMS Works</td>
</tr>
<tr>
<td>10</td>
<td>Data diagram</td>
</tr>
<tr>
<td>11</td>
<td>Landing page Storyboard</td>
</tr>
<tr>
<td>12</td>
<td>Image in Storyboard</td>
</tr>
<tr>
<td>13</td>
<td>Search of words</td>
</tr>
<tr>
<td>14</td>
<td>Management module users</td>
</tr>
<tr>
<td>15</td>
<td>Post-login screen</td>
</tr>
<tr>
<td>16</td>
<td>Storyboard Personal Shields</td>
</tr>
<tr>
<td>17</td>
<td>Storyboard public shields</td>
</tr>
<tr>
<td>18</td>
<td>Storyboard Profile</td>
</tr>
<tr>
<td>19</td>
<td>Storyboard Catalan Heraldry</td>
</tr>
<tr>
<td>20</td>
<td>Storyboard News</td>
</tr>
<tr>
<td>21</td>
<td>Storyboard Contact</td>
</tr>
<tr>
<td>22</td>
<td>Storyboard Tool</td>
</tr>
<tr>
<td>23</td>
<td>pjo_astoria demo</td>
</tr>
<tr>
<td>24</td>
<td>HeraldiCat Logo</td>
</tr>
<tr>
<td>25</td>
<td>Template image</td>
</tr>
<tr>
<td>26</td>
<td>Template header</td>
</tr>
<tr>
<td>27</td>
<td>User management configure</td>
</tr>
<tr>
<td>28</td>
<td>Search configure</td>
</tr>
<tr>
<td>29</td>
<td>User online configuration module</td>
</tr>
<tr>
<td>30</td>
<td>Configured template</td>
</tr>
<tr>
<td>31</td>
<td>Manager menus</td>
</tr>
<tr>
<td>32</td>
<td>&quot;Menú Principal&quot;</td>
</tr>
<tr>
<td>33</td>
<td>&quot;Menú Principal&quot; Bar</td>
</tr>
<tr>
<td>34</td>
<td>&quot;Menú Usuari&quot;</td>
</tr>
<tr>
<td>35</td>
<td>&quot;Menú Usuari&quot; Bar</td>
</tr>
<tr>
<td>36</td>
<td>Exit redirect</td>
</tr>
<tr>
<td>37</td>
<td>Acces redirect</td>
</tr>
<tr>
<td>38</td>
<td>Personal Shields header</td>
</tr>
<tr>
<td>39</td>
<td>Shield with &quot;Like&quot;</td>
</tr>
<tr>
<td>40</td>
<td>Shiel without &quot;Like&quot;</td>
</tr>
<tr>
<td>41</td>
<td>Personal shields result</td>
</tr>
<tr>
<td>42</td>
<td>Like consult image</td>
</tr>
</tbody>
</table>
# Table of used tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Enamels I</td>
<td>15</td>
</tr>
<tr>
<td>Table 2</td>
<td>Furs</td>
<td>15</td>
</tr>
<tr>
<td>Table 3</td>
<td>Basic and distributions</td>
<td>16</td>
</tr>
<tr>
<td>Table 4</td>
<td>Irregular and unequal</td>
<td>16</td>
</tr>
<tr>
<td>Table 5</td>
<td>Other divisions I</td>
<td>16</td>
</tr>
<tr>
<td>Table 6</td>
<td>Basic pieces I</td>
<td>17</td>
</tr>
<tr>
<td>Table 7</td>
<td>Basic pieces II</td>
<td>17</td>
</tr>
<tr>
<td>Table 8</td>
<td>Basic pieces III</td>
<td>17</td>
</tr>
<tr>
<td>Table 9</td>
<td>Basic pieces IV</td>
<td>18</td>
</tr>
<tr>
<td>Table 10</td>
<td>Imaginary Figures</td>
<td>18</td>
</tr>
<tr>
<td>Table 11</td>
<td>Buildings Figures I</td>
<td>19</td>
</tr>
<tr>
<td>Table 12</td>
<td>Buildings Figures II</td>
<td>20</td>
</tr>
<tr>
<td>Table 13</td>
<td>Nobles Crowns I</td>
<td>20</td>
</tr>
<tr>
<td>Table 14</td>
<td>Nobles Crowns II</td>
<td>20</td>
</tr>
<tr>
<td>Table 15</td>
<td>Civil Positions Crowns</td>
<td>21</td>
</tr>
<tr>
<td>Table 16</td>
<td>Estimated time</td>
<td>32</td>
</tr>
<tr>
<td>Table 17</td>
<td>Human resources</td>
<td>34</td>
</tr>
<tr>
<td>Table 18</td>
<td>Software resources</td>
<td>35</td>
</tr>
<tr>
<td>Table 19</td>
<td>Hardware resources</td>
<td>35</td>
</tr>
<tr>
<td>Table 20</td>
<td>Budget summary</td>
<td>35</td>
</tr>
<tr>
<td>Table 21</td>
<td>Table BBDD Users</td>
<td>43</td>
</tr>
<tr>
<td>Table 22</td>
<td>Table BBDD Users_Avatar</td>
<td>43</td>
</tr>
<tr>
<td>Table 23</td>
<td>Table BBDD Obres</td>
<td>44</td>
</tr>
<tr>
<td>Table 24</td>
<td>Table BBDD Obres_Historic</td>
<td>44</td>
</tr>
<tr>
<td>Table 25</td>
<td>Table BBDD Obres_Publiques</td>
<td>45</td>
</tr>
<tr>
<td>Table 26</td>
<td>Table BBDD Obres_Likes</td>
<td>45</td>
</tr>
<tr>
<td>Table 27</td>
<td>Table BBDD Obres_Comment</td>
<td>45</td>
</tr>
<tr>
<td>Table 28</td>
<td>Menus differences</td>
<td>48</td>
</tr>
<tr>
<td>Table 29</td>
<td>SlideShow images</td>
<td>56</td>
</tr>
<tr>
<td>Table 30</td>
<td>Hexadecimal values enamels</td>
<td>103</td>
</tr>
<tr>
<td>Table 31</td>
<td>Division code</td>
<td>105</td>
</tr>
<tr>
<td>Table 32</td>
<td>Figures header</td>
<td>105</td>
</tr>
<tr>
<td>Table 33</td>
<td>Figures code</td>
<td>106</td>
</tr>
<tr>
<td>Table 34</td>
<td>Crowns code</td>
<td>107</td>
</tr>
<tr>
<td>Table 35</td>
<td>New line code</td>
<td>107</td>
</tr>
<tr>
<td>Table 36</td>
<td>Historic actions code</td>
<td>108</td>
</tr>
</tbody>
</table>
1 Introduction

In the following document you can find how I developed this final project. A web tool for editing heraldy shields.

Currently it is not known much about the reality of heraldry. Heraldry is known as the science and auxiliar art of the history that studies specifically the systematic use of hereditary emblems reflected on a coat of arms, its composition and its meaning.

The feature that distinguishes the heraldic symbols from other types can be summarized in three pillars: the assumption of a rigid set of rules for the composition of the emblem, the shape of the emblem in a coat of arms and the transmission of hereditary Armouries as a lineage identifier. [1]

There are few companies that decide to invest in software of this kind, therefore, the few applications that exist today are the result of a developers lovers of science but are far from quality software.

In the same way that there are few softwares, there are not many websites dedicated to this science areas where users can share opinions and experiences.

1.1 Motivation

Motivation is the main tool to be able to push any project forward, overcoming obstacles in the course of its development and be able to take the best decisions possible in every situation.

In my case, the main motivation was to be able to learn how to develop a web tool in order to achieve maximum knowledge about its technology, particularly those languages related to web, that we have present in our lives and I don't see in our Degree in Informatics Engineering.

From the beggining I knew that the system would be complex and long, but, for this reason I increased the level of effort to be able to develop the project with the maximum possible quality.
1.2 Formulation of the problem

Heraldry is a science that was born a long time ago. Since 1125 it has appeared as a sign system for the recognition of individual knights on the battlefield. [1] It has been an evolution that continues to these days. However, it is not this way how science has been able to pass through the digital world.

There are very few versions that have developed tools for coat of arms. Most with outdated technology, poor degree usability and a visual aspect that does not invite to be used. All this is because this world has not created a need for a large volume of business, and the companies of the software sector could not find an opportunity to profit from. The result of all this is the little quantity and variety products developed by people who are lovers of this science and were interested to share their applications with the community.

One very important is that these applications were developed in the past and are not updated to the new technologies offered. There are not applications that allow you to create and modify their shields from a web portal. Or these are softwares to download and install on the personal computer of the user or are softwares that only allow the downloading of the final result and you don’t have options for future modifications.

Regarding spaces to share opinions, knowledge and experiences we find two profiles. On one hand the typical web pages with informative content but little dynamics, where the user can not interact with other people in the community. On the other hand we find standard forums where users can share content but miss customizing the theme and those different contents from the own pages of users who make use.

Starting from these non existing requirements, starts the idea to develop this project that integrates a web tool of heraldic software where users can manage their shields with a space where users can share their opinions. All this, with the advantage that new technologies offer us, with a nice visual appearance and acceptable level of usability.
1.3 Goals

The main goals of the project are:

- Learn skills to develop the necessary technologies.
- The development of the project will require a basic knowledge of PHP, HTML, CSS, jQuery and JSON to the website, ActionScript3 and Flash for the heraldic software and MySQL to store the data.
- To be able to store and display the information that will synchronize properly with a server.
- Be able to manage all personal creations and to exchange information through a web portal.

1.4 Technical competences

Then, you can see the list of technical competences related to this project.

- **CSI2.4:** To demonstrate knowledge and capacity to apply systems based on Internet (e-commerce, e-learning, etc.). [A little bit]
- **CSI4.1:** Participate actively in the specification of information systems and communication. [Enough]
- **CSI3.3:** Evaluate technology offers for the development of information systems and management. [Enough]
- **CSI4.2:** Participate actively in the design, implementation and maintenance of information systems and communication. [In depth]
2 Research

2.1 Heraldry

Heraldry is the science that studies specifically the systematic use of emblems hereditary reflected on a coat of arms.

The feature that distinguishes the heraldic symbols of other types can be summarized in three pillars: the assumption of a rigid set of rules for the composition of the emblem, the shape of the emblem in a coat of arms and the transmission of hereditary Armouries as a lineage identifier. [1]

2.1.2 Composition

- Symmetry.

It seeks a balance and harmony in the composition and placement of the pieces on the shield.

- Fullness.

The set of figures must occupy the largest area in the shield, without touching its border.[2]
2.1.3 Parts of the shield

Illustration 1 - Parts of the shield

2.1.4 Enamels

The basic rule is that the enamel can not be juxtaposed metal with metal and color with color. There are colors that are considered neutral, we also have linings that can be combined together. [3]

<table>
<thead>
<tr>
<th>Main colors</th>
<th>Neutral Enamels</th>
<th>Main metals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azure</td>
<td>Gulas</td>
<td>Sinople</td>
</tr>
<tr>
<td>Purple</td>
<td>Sable</td>
<td>Gold</td>
</tr>
<tr>
<td>Sable</td>
<td></td>
<td>Silver</td>
</tr>
</tbody>
</table>

Table 1 - Enamels I

<table>
<thead>
<tr>
<th>Furs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ermine</td>
</tr>
<tr>
<td>Vair</td>
</tr>
</tbody>
</table>

Table 2 - Furs
2.1.5 Divisions

In a single shield the parts can represent the heraldry of several lineages, the lineages can refer to a joint family. Each division is studied separately. [2]

<table>
<thead>
<tr>
<th>Basic</th>
<th>Distributions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Party per fess</td>
<td>Party per pale</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Party per pale</td>
<td>Party per bend</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Party per bend</td>
<td>Party per bend sinister</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Party per bend sinister</td>
<td>Party per cross</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Party per cross</td>
<td>Party per saltire</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 3 – Basic and distributions

<table>
<thead>
<tr>
<th>Irregular and unequal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slanting in mast</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

Table 4 - Irregular and unequal

<table>
<thead>
<tr>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Party in arrow</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

Table 5 – Other divisions I
2.1.6 Basic pieces

The pieces of the shield are not the result of a division like the partitions, the pieces are applied above it. [4]

<table>
<thead>
<tr>
<th>Basic pieces</th>
<th>Chief</th>
<th>Pale</th>
<th>Fess</th>
<th>Cross</th>
<th>Bend</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
</tr>
</tbody>
</table>

Table 6 - Basic pieces I

<table>
<thead>
<tr>
<th>Basic pieces</th>
<th>Bend sinister</th>
<th>Saltire</th>
<th>Chevron</th>
<th>Pall</th>
<th>Foot</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
</tr>
</tbody>
</table>

Table 7 - Basic pieces II

<table>
<thead>
<tr>
<th>Basic pieces</th>
<th>Bordure</th>
<th>Orla</th>
<th>Canton</th>
<th>Francquarter</th>
<th>Gironyy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
<td><img src="image13.png" alt="Image" /></td>
<td><img src="image14.png" alt="Image" /></td>
<td><img src="image15.png" alt="Image" /></td>
</tr>
</tbody>
</table>

Table 8 - Basic pieces III
Table 9 - Basic pieces IV

2.1.7 Figures

The figures are a class of load heraldic that are represented on a coat of arms. We have natural, artificial, imaginary and buildings. [4]

Table 10 - Naturals Figures I

Table 11 - Naturals Figures II
### Naturals

<table>
<thead>
<tr>
<th>Sheep</th>
<th>Hand</th>
<th>Fish</th>
<th>Fish</th>
<th>Head of beef</th>
</tr>
</thead>
</table>

Table 12 - Naturals Figures III

### Artificials

<table>
<thead>
<tr>
<th>Wheel</th>
<th>Libra</th>
<th>Mace</th>
<th>Shovel</th>
</tr>
</thead>
</table>

Table 13 - Artificials Figures I

### Artificials

<table>
<thead>
<tr>
<th>Slabs</th>
<th>Tupin</th>
<th>Cauldron</th>
<th>Sword</th>
</tr>
</thead>
</table>

Table 14 - Artificials Figures II

### Imaginary

<table>
<thead>
<tr>
<th>Griffin</th>
<th>Dragon</th>
</tr>
</thead>
</table>

Table 15 - Imaginary Figures
2.1.8 Crowns

Crowns are ornaments that are situated on the top of the shield. [4]

---

Table 16 - Buildings Figures I

<table>
<thead>
<tr>
<th>Buildings</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Castler</td>
<td>Tower</td>
</tr>
</tbody>
</table>

Table 17 - Buildings Figures II

<table>
<thead>
<tr>
<th>Buildings</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Palace</td>
<td>Sanctuary</td>
</tr>
</tbody>
</table>

---

Table 18 - Nobles Crowns I

<table>
<thead>
<tr>
<th>Nobles Crowns</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>King</td>
<td>Prince</td>
</tr>
</tbody>
</table>

Table 19 - Nobles Crowns II

<table>
<thead>
<tr>
<th>Nobles Crowns</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>Viscount</td>
</tr>
</tbody>
</table>
2.2 Existing applications

2.2.1 Examples

There is a research on existing software for creating heraldic shields. We can see examples that are different from one another that show the current state of heraldic softwares.

- Blazon

Blazon is a downloadable software, it is ready to run for Windows 95, so we can get an idea of the antiquity of the software. It is an unfriendly and pixelated interface and for this reason it can not be used for a long time. Among the most positive things we can find the many features and strict monitoring of the rules of using enamels. Among other things, it allows you to save and print your creations. Learn more by visiting their website. [5]
• Puncher

Puncher is a website that lets you generate the shield thought the selected data. It is very fast to use, but does not follow any rules. The options are very basic, there is little variety. You can not download the creations. Learn more by visiting their website. [6]

• Inkwell Ideas

Inkwell Ideas is a website creation software among which we find this designer of coat of arms. Besides slowness, it is a little usable tool, little intuitive and withouth the option to undo the last action. Do not follow any rules of heraldry. Wide range of possibilities. Export and import option for future editions. Download image in PNG format. Learn more by visiting their website. [7]

2.2.2 Existing applications conclusions

After analyzing the software described, seek other options and see opinions from different user in communities related to the lovers of this science, we can conclude that today there is no software with a friendly interface and an acceptable level of usability. There is no space to create, save and edit entirely from the web site. Thus, you must design a new software from scratch. Can use any of the existing ideas but the level of similarity with these will be very low, the tool will be quite different. Neither will use the technology that they use, because is outdated and little usable, shows an unacceptable response time.
3 Context

3.1 Actors involved

Then, the actors that are involved in this project:

3.1.1 Developer

In this case will be the responsible person for do the project development system. Will be responsible for taking important decisions throughout the implementation and deployment of the system.

3.1.2 Project director

Their role will be to supervise and advise the developer to achieve the project success.

3.1.3 Final users

The final users are the lovers of heraldry that they will use both the web tool for creating heraldic shields as the small social network that allows the exchange of views and experiences with other users.
4 Scope

The following will describe the scope of the project.

4.1 Overview

This project is to provide the heraldic lovers two integrated areas that do not currently exist in our society. On one hand a space where the users can create heraldic shields. On the other hand give a small social network where the users can share information between other users.

4.2 Goals

The main goals of this projects are:

- An image editing software to that the user can create their own shields.
- Creation a small social network. A community where the users can share experiences and knowledge about Heraldry.

4.3 Risks

This project there are a number of obstacles and risks that must be borne in mind when making development tasks. For these risks we can minimize the impact defining a clears mitigation strategies and contingency plan.
• Loss of work.

Could be the case that for any reason the work is lost.

*Probability:* High.

*Impact:* High.

*Mitigation:* Make of backups.

*Contingency:* Make use of backups to recover the lost information.

• Lack of the technology.

Exists de possibility that the developer unknown the technology used to realize the project. In the case that the develop not choose a good option the project could have inconsistencies or configuration problems.

*Probability:* Very high.

*Impact:* High.

*Mitigation:* Comprehensive study of the different technologic possibilities, analyze the features of each one and realize an alternative list.

*Contingency:* Make use of the alternative list.

• Error in the planification.

Some stages of the project could have a low difficult planification that the real, so, do not give period enough of work to develop correctly.

*Probability:* Medium.

*Impact:* Medium.

*Mitigation:* Comprehensive study of each stage and an adequate estimation with safety margins

*Contingency:* Make a replanification of the development calendar.
• Lack of heraldry.

It could happen some mistake with the rules of the science.

*Probability*: High.

*Impact*: Low.

*Mitigation*: Comprehensive study of the Heraldry, gathering as much information as possible available to the developer.

*Contingency*: We do not have any plan in this case.

• Experiences in projects of this magnitude.

The experience of the author in projects of this magnitude is low and could be a risk to realize that.

*Probability*: Medium.

*Impact*: High.

*Mitigation*: Good planification, good methodology of work and supervision of the director.

*Contingency*: Don’t exist any plan in this case.
This Project follow the RUP methodology (Rational Unified Process). It is a process based on iterations, focused on the architecture and is based in use cases. [8]

The development structure is divided into four stages.

- Planification of the project (Inception).
- Analysis and design the project (Elaboration).
- Iterations project (Construction).
- End (Transition).

The iterations have been distributed in the development of this project such that we always have a functional version at the end of each of them. However, this method has not been applied to 100% from the beginning because the developer already knew all tasks to do.

The metodology of work will be to separate the different functions of the project in iterations in order to achieve goals in small periods of time. It marks a work schedule with a start dates and finish dates for the different stages.

For each iteration:

- Analysis.

Analyze everything to develop the tasks to do. The develop will study the requirements to accomplish, the diferents options to do that and a little guide to follow.

- Implementation.

Development of the code and interface to perform following the analysis guide.

- Test.

Personal test of the result. Search for errors or improvements and correction of these.
• Validation.

Once the author take good the iteration, it want validate with the experiences of other users. This users can be close to the author (family, friends...) depends the kind of function developed.

• Completion.

Once completed the stage, the develop will be a study of the global state of the project, it will be analyzed if are necessary changes in the scheduling of work.

ZOHO (www.zoho.com) will be the tool to monitor the project. Zoho is a free website with many applications where use Zoho Projects, an project manager that offers total control to manage project of any type, assigns tasks with deadline, take account of tasks already done, or that are needed to be done, monitor the ended tasks and, and other functions like upload files. [9]

In the realization of the memory of this project, the developer document everything together, without differentiating the iterations in order to detail with more precision the development.
6 Project plan

To planificate the project the tasks to develop are divided into four main stages. The developer has tried to divide the project in four parts that allow to be developed with the least effect of changes between them, that is, if the developer want to make changes in one part affect the minimum in the rest of the stages. These are the stages:

6.1 Project stages

For all stages of development the developer are saved a little space to make extensions in the case that sufficient time is available. There will also be extensive testing for each of them so that when finish the stages are evaluated with 100% finished.

6.1.1 Project planning

This stage are part of the GEP course. This course includes a series of sections. These sections are:

- Introduction
- Scope of the project.
- Schedule plan.
- Budget and sustainable.
- Research and references.

6.1.2 Analysis and design of the project

The main goal of this stage is make an accurate analysis of the project to develop the design.

Regarding the analysis, the developer will define the goals and needs along with the different use cases.
The design of the project consist in the creation of the program architecture, the data model, class diagram and storyboards.

### 6.1.3 Iterations project

The main part of the Project. This part is very broad and it has decided to divide in different iterations:

- **Preparation of the environment.**

  This phase is intended to prepare the work environment needed to develop the remaining phases.

- **Registration and login.**

  This phase consist in build the user registration area like any other website with registers. Once registered, the first thing that the user will do is access to the website (login), so the developer will design this function. Here is the first time when the developer start to work with database.

- **Editing shields.**

  The main part of the project, this function allows the user the edition of shields following the Heraldry. The good performance of the data and interface require a long time of test and validations. In this phase, the developer will try to take the expert opinion of members of the Societat Catalana de Genealogia, Heràldica, Sigil·lografia, Vexil·lologa i Nobilìari (SCGHSVN) to validate the software.

- **Personal area.**

  Design the personal area of the user. This is where the user will see a list with all shields designed.
• Social network.

Construction of all functionalities for sharing experiences and knowledge about the rules of Heraldry between users.

• Integration.

This phase consist in the integration of all functionalities developed in the previous phases. Once finished the phase, the project will work totally.

• Final corrections,

Once the project works, the developer will continue to improve in order to reach a level of excellence, this phase will be especially to improve the web design in order to make it as attractive as possible.

6.1.4 Completion

The completion stage of the project, where the developer will be finished and tested the tool. In this phase, the developer will complete de last details and write the memory of the project and make a final revision of the application and the content of the memory.
7 Planification and budget

7.1 Estimated time

Below, you can see a table with the estimate of time spent in each stage of the project plan. The 380 hours of the stage of iteration project (3) is the sum of different iterations of the project.

<table>
<thead>
<tr>
<th>Stage</th>
<th>Effort (hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Planning</td>
<td>75</td>
</tr>
<tr>
<td>Analysis and design of the project</td>
<td>100</td>
</tr>
<tr>
<td>Iteration project</td>
<td>380</td>
</tr>
<tr>
<td>Preparation of the environment</td>
<td>10</td>
</tr>
<tr>
<td>Register and login</td>
<td>30</td>
</tr>
<tr>
<td>Editing shields</td>
<td>150</td>
</tr>
<tr>
<td>Personal area</td>
<td>80</td>
</tr>
<tr>
<td>Social network</td>
<td>70</td>
</tr>
<tr>
<td>Integration</td>
<td>20</td>
</tr>
<tr>
<td>Final corrections</td>
<td>20</td>
</tr>
<tr>
<td>Completion</td>
<td>50</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>605</strong></td>
</tr>
</tbody>
</table>

Table 21 – Estimated time

For each stage of the project the developer has added a margin for preventive complications. So, this way, one mistake in the planification do not affect the rest of stages/iterations of the project.

On the next page we can see the planification in the Gantt diagram.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Project planning</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 Scope</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 Schedule plan</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 Budget</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 Presentation preview</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6 Contextualization and references</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7 Specialty</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8 Documentation and presentation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9 Analysis and design of the project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10 Goals and requirements</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 Data model</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12 Class diagram</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13 Sequence diagram</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14 Storyboards</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15 Iteration project</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16 Preparation of the environment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17 Register and login</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18 Editing shields</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19 Personal area</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20 Social network</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21 Integration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22 Final corrections</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23 Completion</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24 Documentation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25 Presentation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7.2 Budget

Below, you can see a list of the different costs that the developer have been identified in order to develop the project, there are humans and materials.

7.2.1 Human resources

For human resources, the quantity will be proportional to the time spent by the software developer. Taking this salaries:

- Project Manager: 40 € / hora
- Software developer: 20 € / hora
- Tester: 15 € / hora

<table>
<thead>
<tr>
<th>Stage</th>
<th>Hores spent</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Planning</td>
<td>75 0 0</td>
<td>3.000 €</td>
</tr>
<tr>
<td>Analysis and design of the project</td>
<td>100 0 0</td>
<td>4.000 €</td>
</tr>
<tr>
<td>Iteration project</td>
<td>63 269 48</td>
<td>8.620 €</td>
</tr>
<tr>
<td>- Preparation of the environment</td>
<td>2 8 0</td>
<td>240 €</td>
</tr>
<tr>
<td>- Register and login</td>
<td>5 25 0</td>
<td>700 €</td>
</tr>
<tr>
<td>- Editing shields</td>
<td>20 110 20</td>
<td>3.300 €</td>
</tr>
<tr>
<td>- Personal area</td>
<td>10 60 10</td>
<td>1.750 €</td>
</tr>
<tr>
<td>- Social network</td>
<td>10 50 10</td>
<td>1.550 €</td>
</tr>
<tr>
<td>- Integration</td>
<td>8 8 4</td>
<td>540 €</td>
</tr>
<tr>
<td>- Final corrections</td>
<td>8 8 4</td>
<td>540 €</td>
</tr>
<tr>
<td>Completion</td>
<td>25 294 48</td>
<td>1.500 €</td>
</tr>
<tr>
<td>Total</td>
<td>263 294 48</td>
<td>17.120 €</td>
</tr>
</tbody>
</table>

Table 22 – Human resources

7.2.2 Material resources

The next tables show the costs of each material, separating hardware and software.
We can see the cost of depreciation considering the duration of the project and the useful life of each item.
- Software

<table>
<thead>
<tr>
<th>Product</th>
<th>Units</th>
<th>Price</th>
<th>Useful life</th>
<th>Depreciated cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>DELL XPS L501X</td>
<td>1</td>
<td>980.95 €</td>
<td>5</td>
<td>73.57 €</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>980.95 €</strong></td>
<td><strong>5</strong></td>
<td><strong>73.57 €</strong></td>
</tr>
</tbody>
</table>

Table 23 – Software resources

- Hardware

<table>
<thead>
<tr>
<th>Product</th>
<th>Units</th>
<th>Price</th>
<th>Useful life</th>
<th>Depreciated cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows 7</td>
<td>1</td>
<td>88.10 €</td>
<td>4</td>
<td>8.26 €</td>
</tr>
<tr>
<td>Microsoft Office 2013</td>
<td>1</td>
<td>119.00 €</td>
<td>4</td>
<td>11.16 €</td>
</tr>
<tr>
<td>Enterprise Architect</td>
<td>1</td>
<td>253.95 €</td>
<td>4</td>
<td>23.81 €</td>
</tr>
<tr>
<td>Zoho Project</td>
<td>1</td>
<td>0.00 €</td>
<td>N/A</td>
<td>0.00 €</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td><strong>461.05 €</strong></td>
<td><strong>N/A</strong></td>
<td><strong>43.22 €</strong></td>
</tr>
</tbody>
</table>

Table 24 – Hardware resources

- Microsoft Windows 7 like OS (Operative System).
- Microsoft Office 2013 like text editor and presentation tool.
- Enterprise Architect for the design stage of the project. Allow create data models, class diagram, sequence diagrams, requirements...
- Zoho Project like project manager for take the control of the planification of the project.

7.2.3 Budget summary

<table>
<thead>
<tr>
<th>Concept</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human resources</td>
<td>17.120,00 €</td>
</tr>
<tr>
<td>Material resources</td>
<td>116,79 €</td>
</tr>
<tr>
<td>Hardware</td>
<td>73.57 €</td>
</tr>
<tr>
<td>Software</td>
<td>43,22 €</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>17.236,79 €</strong></td>
</tr>
</tbody>
</table>

Table 25 – Budget summary
8 Sustainability i social engagement

At around this project, there is no environmental impact, both negative and positive that can be taken because the development and the final result of this system are not relationated directly or indirectly to a particular environment aspect.

In the social aspect the project will have a strong impact on those lovers of Heraldry. Thanks the tool they will receive the following benefits:

- An area where the user can experiment with the rules of Heraldry

The lovers of this science have now an online space where they can experiment in a friendly interface and they can enjoy the full of possibilities of the software. Until now, not exists a software of this theme with a level of depth of this magnitude.

- Improve the relations and network of knowledge between the users.

Thanks the social network, the users can be share experiences, knowledge and opinions about Heraldry, this will create a small social network with a significant flow of data and knowledge.
9 Analysis and specification

In this section the author will take the analysis of requirements and specification of the system. The author describe in detail the full operation of the system in order to clearly specify the functional and not functional requirements.

9.1 System Architecture

The main idea is to store the shields of users on a server, in order to display the content or edit them. So first we need a system that allows the management of information in a database. The author has decided to MySQL as a system management database because he has experience with that relational database system and it is easy to use. In order to use this database the author has decided to use WAMP SERVER, this is an infrastructure that includes APACHE as web server, MySQL as database management and PHP (and others) as a programming language.

The system requires a user manager. The developer could develop this from the beginning or use one of the options that current technology offers. For this reason, after a study of the different options the developer has decided to use Joomla. Joomla is a content management system (CMS) and open source software to publish content on the World Wide Web at the same time is a framework of application development type Model-View-Controller (MVC). It is written in PHP and uses MySQL database system.

Joomla includes many modules, one of this is the user manager, very interesting for our system. The manager includes register, login, logout and other options as if the user forget the password or username. Other interesting modules are the search words or the information of the connected users in the system. Another important aspect is the opportunity that Joomla offer in order to upload...
templates or extensions that allow us to make backups and avoid the risk of lose the information.

The author had other similar options such as Drupal or Wordpress. One of the main problems for Drupal is that the experts are sure that the developer need a high learning curve. The low usage to dedicate at the CMS is not advisable to devote an extra time. The developer want to be dedicate more effort in other important areas in the our product.

Wordpress has difficulties with regard to resources consumption because it loads many scripts for each page. Other main issue that did the developer discard this option was the difficulty to modify the programming code.[11]

In the picture below we can see how CMS works.
9.2 Requirements analysis

Then, the analysis of system requirements.

9.2.1 Functional requirements

Functional requirements define the set of functions that the system must able to perform. The author has define depending the environment:

Web level:

- Shows correctly the information of the shields for each user. This includes:
  - Image with shield.
  - Name.
  - Creation date.
  - Modification date.
  - Social buttons “Like” and “Comment” in the case that the shield is public.
  - Information social buttons with the number of “Like” and “Comment” in the case that the shield is public.
  - Link to edit the shield.

- Shows correctly the information for public shields. This includes:
  - Image with shield.
  - Username of the creator the shield.
  - Name.
  - Creation date.
  - Modification date.
  - Social buttons “Like” and “Comment”.
  - Information social buttons with the number of “Like” and “Comment”.

- Shows correctly the information of the user’s profile. This includes:
  - Current Avatar.
  - Username.
  - Number of created shields.
  - Sum of “Like” in all their shields.
  - Button that allow the user change avatar image.

- Shows correctly the information about Catalan Heraldry guide.
• Shows correctly the information about news related to Heraldry.
• Shows correctly the information to allow the user to contact the administrator of the system.

Tool level:

• Must load correctly the information in the case that the shield is a modification and not a new creation.
• Must save the information when the user wants to save the shield.
• Must show correctly the shield in the case that the user presses a division.
• Must allow drawing new lines in order to create new not-existing divisions.
• Must show correctly the figure in order to manage in the case that the user presses on the figure.
• Must show correctly the figure in the case that the user drags to the shield.
• Must delete the figure in the case that the user sends the figure to the trash.
• Must show correctly the shield in the case that the user presses a crown.
• Must allow painting an area of the shield.
• Must allow making undo of the last action performed.
• Must show specific information of actions performed adding these in the historical.
• Must allow undo any action taken throughout history.

9.2.2 Non functionals requirements

The non-functional requirements define various aspects of the system developed but not about functions that should be able to do. These are:

• Fast answer from an interaction user.
• Operative.
• Maintainable.
• Scalability, with database and application function.
• Usable and friendly system interface.
• Functional for Chrome, Explorer and Firefox.
• Secure.
• Updated at the regulation of LOPD (Organic law of data protection).
9.3 Use cases

This section defines the different use cases to know the tasks that the user can perform in the system:

Web level:

- User management. This includes the user registration, login and logout.
- View the shields created.
- Create a new shield.
- Modificate a shield.
- Press click the button "Like" to public shield. Also the option to return the previous state.
- Comment a public shield. Also the option to delete a comment performed.
- Be able to modify the profile image (avatar).
- Be able to consult the Catalan Heraldry guide.
- Be able to consult the news relationated with the Heraldry.
- Be able to consult an area with information to contact with the system Administrator.

Tool level:

- Give a name to the shield.
- Be able to mark the shield as public in order to the rest of community users can see the shield.
- Download the shield in PNG format.
- Select a division.
- Draw new lines in free format in order to create new not exist divisions.
- Select figures and put on the shield.
- Send a figure to trash.
- Select a crown.
- Paint the shield.
- Undo any action taken with history. Also an "undo" button to undo the last action.
- Save the shield.
9.4 Database model

At the time of install Joomla, this already insert a number of tables in the database. The developer has tried don’t modify the tables, the reason is that probably in future upgrades the system need the tables. Another reason is that way the developer are sure that this modifications not affect the system.

Illustration 10 – Data diagram
We can analyze each table.

**Users:**

La Table Users contains the information for all users registered in the system.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>User ID. PK (Primary Key).</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the user.</td>
</tr>
<tr>
<td>Username</td>
<td>Personal name that gets the user to be identified in the system. UNIQUE (Can not exist two users with the same username).</td>
</tr>
<tr>
<td>Email</td>
<td>Email of the l'usuari. The function of this field is to send an email to user with information in order to activate the account or change the configuration.</td>
</tr>
<tr>
<td>Password</td>
<td>The password of the user to enter in the system.</td>
</tr>
<tr>
<td>Edicio</td>
<td>Boolean that identifies whether the user is making a new creation or editing an exist.</td>
</tr>
</tbody>
</table>

Table 26 - Table BBDD Users

**Users_Avatar**

The author wants to create a small social network, it wants that the user has an image (avatar) associated with it. The table Users_Avatar contains:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>User ID.</td>
</tr>
<tr>
<td>Imatge</td>
<td>BLOB type field that keeps the image.</td>
</tr>
<tr>
<td>Tipo</td>
<td>Field that contains the information of the image extension. GIF, JPEG, PNG or .PJPEG.</td>
</tr>
</tbody>
</table>

Table 27 - Table BBDD Users_Avatar

To provide greater security, reliability, and speed to the system the developer has decided that each user has the rest of the table in a custom way. For example, the user "xavier5" will have the following tables: Obres_xavier5, Obres_Historic_xavier5, Obres_Likes_xavier5 and Obres_Comment_xavier5.
La Obres table contains the information of the works created by the user. This information includes:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Work ID to associated with other tables.</td>
</tr>
<tr>
<td>Nom_obra</td>
<td>Name of work.</td>
</tr>
<tr>
<td>Public</td>
<td>Boolean that indicates if the work is public or not.</td>
</tr>
<tr>
<td>idHistoric</td>
<td>Contain the ID that it is associated with Obres_Historic table.</td>
</tr>
<tr>
<td>dataCrea</td>
<td>Creation date.</td>
</tr>
<tr>
<td>dataMod</td>
<td>Last modification date.</td>
</tr>
<tr>
<td>Imatge</td>
<td>BLOB with the image shield in PNG format.</td>
</tr>
</tbody>
</table>

Table 28 - Table BBDD Obres

Obres_Historic

This table contain the information that the user performed to create the shield. This table could be convert in a field of Obres table but the developed has think that in a future someone could develop a system function to recover old versions of the shield.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Work ID</td>
</tr>
<tr>
<td>His</td>
<td>String that includes all actions performed by the user in the creation of shield. They are a set of numerical values and symbols “.” and “+” that you will understand in other section.</td>
</tr>
</tbody>
</table>

Table 29 - Table BBDD Obres_Historic

Obres_Publiques

The table Obres_publiques is joint for the users, includes the basic information of the works that they have been configured as public. Some fields of this table are redundant, so the information already exists in other tables of databases but the developer decide this option to improve the speed and simplify the consults to database. In this table we find:
Field Description
IdIntern Internal ID.
ID ID that associate the work with ID of Obres table.
Username Username of the user that create the work.
Public Boolean that indicates if the work is public or not. This already exist in Obres table, the developer has added the field in this table because if one work passes a private state the developer do not want delete the line and just modify this field if the user want to make public again in the future.
Likes Numeric field that indicates the number of "Like" of the work.

Table 30 - Table BBDD Obres_Publiques

Obres_Likes

This table includes informations about the “Likes” in the shields.

Field Description
idObra ID associated with public work.
idUser ID of the user who made “Like”.

Table 31 - Table BBDD Obres_Likes

Obres_Comment

This table includes information about the “Comment” in the shields.

Field Description
idObra ID associated with public work.
idUser ID of the user who made the comment.
Comment Field with the content of the comment.
ID Internal ID and PK, one user can be more than one comment in the same work.

Table 32 - Table BBDD Obres_Comment
9.5 Storyboards

9.5.1 Web

A storyboard is a graphical representation before its realization. Then we can see the storyboards made before the development of the system. When the author made the storyboards he had already selected a Joomla template and the author knew the modules interesting to integrate in the system.

Now, we can see the landing page view.

![Landing page Storyboard](image)

We can fix that we are in the top two images that have the shape of a cross. On the left will place a logo ("brand") and on the right an image related to an issue of Heraldry. Just below an horizontal menu with buttons. In other section the author explain in detail, but it is important to see that this buttons are available for a person who has not registered or make login in the system.

On the left side there are two modules of Joomla. On the one hand we have the search words. On the other hand, just below, we have the connected users information.
On the right side we have the management module users. Where the user introduce their username and password to enter in the system. On below this module we see three links to functionalities of the management user.

At the center of the page will have a Slideshow, a number of images related with the heraldic world that they are changing while the user does not make any interaction.

The user will see a screen where the Administrator user will inform about the news of the system when the user make "login".

On the right side the user can see the user management module with a welcome message and a button to make "logout".
If compare the last image we can see some buttons that have been added in the horizontal menu. In the next table the author explain the differences between connected users and not connected users.

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Connected user</th>
<th>Not connected user</th>
<th>Differences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inici</td>
<td>Start page</td>
<td>X</td>
<td>X</td>
<td>One contain images, the other information of the system. Un conté imatges</td>
</tr>
<tr>
<td>Escuts Personals</td>
<td>Area with created shields</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Escuts Públics</td>
<td>Area with shields marked as public</td>
<td>X</td>
<td>X</td>
<td>The user can see who has made “like” or comment but can not interact with other users.</td>
</tr>
<tr>
<td>Perfil</td>
<td>Area with personal information</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Heràldica Catalana</td>
<td>Area with information about Catalan Heraldry</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Notícies</td>
<td>Area where the user can read news relationated with the heraldic world</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Contacte</td>
<td>Area to contact with the Administrator system</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

Table 33 - Menus differences

Thus, the author has tried that the not registered user can see the maximum possible functionalities in order to motivate them to enter in the system.
In the picture below you can see the personal shields section. Here the user can see all their creations ordered by create date. On the top side we can see the button to create a new shield and information with the total number of shields created. To the right of each shield the user have a link to edit it.

If the shield is public, social button “Like” and “Comment” will appear. On the right are two images, an image relationated with "Like" that in the moment to click the button will display all users who made the action. On the right of this image appear the number of users who made "Like". Further right we have exactly the same information but for "Comments".

Illustration 16 - Storyboard Personal Shields

In the public shields section the user see a list of all public shields. It appears exactly the same information for personal shield, just is added the username that created the shield. In case of the shield was made by the own user will appear a link to edit it.

Illustration 17 - Storyboard public shields
In the profile section will appear an image with the current Avatar and a button to upload a new image. Probably, in this area the author will added some information about the user interactions to make it more complete.

In the Catalan Heraldry section, the user can see information about the Catalan Heraldry rules. The user must press in every title to open that section and read that.
The administrator can add news about the Heraldry World like a blog. These news will sorted by most recent entries.

Illustration 20 - Storyboard News

In the Contact section the user will have a small form to contact the system administrator. The form will have a free field to write what the user want and a list-item with the reason to contact the administrator, so the administrator can filter.

Illustration 21 - Storyboard Contact
### 9.5.2 Tool

Then, the author has drawn the design that has been prepared for the screen where the user can edit the shield.

![Storyboard Tool](image)

**Illustration 22 - Storyboard Tool**

In the image you can identify a number of different areas:

- **Header:** Global Information of the shield. Here, the user will introduce the name and can press the checkbox to mark the shield and make it public for other users. On right side we can see the button "Download" to make a copy of the image of the shield and the "Save" button to finish the edition.

- **Menus:** Just below the header we see a horizontal bar with menus that are directly relationated with the different areas of Catalan Heraldry. When the user press a menu will appear images that serve as buttons, for each button that is pressed the tool is expected to react to user deman.
• Information: At the top left we have a small area where the user will receive simple instructions, three or four words, how to use the tool. It has also been thought that this area used to be that if you select a new figure is created a copy in this area and from there the user can make drag and drop the shield.

• Editing tools: At the bottom left you can see a set of buttons to editate the shield. These buttons are:
  • Undo: Button to undo the last action.
  • Normal cursor: Return to normal state.
  • Line: Where the user press this button the user can draw lines to create new non existing divisions.
  • Paint cursor: When the user select this option the user can paint the areas where press above with the color that is marked in the Colorpicker.
  • ColorPicker: Grill with the colors of the Catalan Heraldry.

• Trash: Located in the upper right this area serve the user to drag and drop the figures in the case that the user want to delete them.

• History: In the bottom right the user see a set with each action performed by the user. In the case that the user want to delete some action the user can do clicking on the line of history.

• Shield: In the middle of the view and the hearth of the tool. All interaction will modify on this area. Both the case of “Save” and “Download” where the user create or edit an image in PNG format.
10 Development System

The author has divided the development of the system in different parts. You can see how these parts have been developed for each of these.

10.1 Preparation of the environment

This phase is basically the installation of WAMPSERVER server as we explained in the planning project. In reference [12] you can see a link to installation guide.

The second step is the installation of Joomla. In reference [13] you can see a link to a installation using Joomla WAMPSERVER guide.

The third step is the general configuration of Joomla. The developer starts installing the template decided to use after a period of search the different options. Finally the developer decided to use the template "pjo_astoria". Then you can see a demo done by the developer of the template. [14]

Illustration 23 - pjo_astoria demo

This template was made more to be used for business but it is perfectly adaptable our needs. In reference [15] you can read how to install Joomla template.
10.2 Joomla template configuration

Once installed template the developer proceed to configure it. The first step is add a logo at the top left. For this reason the developer made an image that tries to play with "Heraldry" and "Catalan" words thus forming "HeraldiCat" word. In this brand added some lines with azure and gold colors. Two enamels used in the heraldry.

To the right of the logo the template offers the possibility to display an image. This image can not be inserted from the configurator Joomla but the developer can insert directly from the CSS code.

```css
.header
{
    background-color: transparent;
    min-height: 80px;
    background-image: url(../images/backgrounds/header_top.png);
    background-repeat: no-repeat;
    background-position: 101% 98%;
}
```

Therefore it has been decided to modify the route of the background-image so that it has the path to the image that has finally decided to accompany the logo on all pages except the editing tool. [16]
With just these two configurations we are sure that all pages have a Joomla template header as shown below:

![Template header](image)

Illustration 26 – Template header

The next step has been to insert slideshow images. Here are the images that appear when the user is on the landing page to the system. Five attractive images have been inserted for lovers of heraldry.

![Slideshow images](images)

Table 34 – SlideShow images

In this SlideShow has modified the settings in order to the navigation buttons disappear because they are unattractive. Another thing that has changed is the type of motion effect, the developer choosed fade effect, also has changed the times of the rotation, now is more slow.
To complete the configuration of template the developer has removed the buttons that they was linked with different social networks (Twitter, Google+, MySpace...) because is not currently contemplated in the system.

### 10.2.1 Installation and configuration modules

Once done the configuration of the global Joomla template, we can install different modules that we searched previously.

The first step is configure user management module, known as Joomla module "Access." For aesthetic reason in this module the developer will hide the "Show Title" and place the module in the position "right" of the template. The developer modified the settings in order to make visible the module on all pages where the system use of the template.

The next module to be installed is the "Search Words", in Joomla the module is called "Search". In this module, such as user module the developer has hidden the Title and has assigned the module to all pages. The position int this case has been in “left”. Another modification was the box width in order to adapt the module to the template. Finally the developer has decided that the text comes to default to guide the user to search words in that area is “Cerca...”.

Finally, the last module that the developer installed was the users connected, called in Joomla as “Who is online”. The developer did the same configuration like the Search module. Joomla itself place the two modules in the same position puts one under the one. The developer decided that the the system inform both registered users and visitors.
Once installed and configured all modules already have an image like this on every page. Please refer to the provided image for a visual representation.

**Illustration 30 – Configured template**

### 10.2.2 Menus configuration

In this section the developer configure the different modules that we have in our system. Here comes the first problem to us and we have two types of menus, depending on whether the user has make login or not. The problem is that Joomla support only a menu. In order to provide a solution to this problem the developer has been decided to double the templates and assign a menu to each template.

Thus, we have on one hand the "Menú Principal" for the users who has not login into the system and the "Menú usuari" for those users that did it.

**Illustration 31 – Manager menus**
For the “Menú principal” the developer has created those sections described:

- Inici.
- Escuts Públics.
- Heràldica Catalana.
- Noticies.
- Contacte.

The result of this configuration is a horizontal bar as we can see in the picture below:

![Illustration 32 – “Menú Principal”](image)

The developer has created different sections as Joomla articles, so in those articles the developer has the need to insert HTML, PHP, CSS and JavaScript code the developer will install the Sourcerer plug-in that allow insert such source for articles. [17]
The developer did the same for “Menú usuari”. We define the sections:

- Inici.
- Escuts Personals.
- Escuts Públics.
- Perfil
- Heràldica Catalana.
- Notícies.
- Contacte.

Illustration 34 – “Menú Usuari”

The result of this configuration is a horizontal bar as we can see in the picture below:

Illustration 35 – “Menú Usuari” Bar
The fact that the developer uses two templates, one for each menu, means that we should also change the workflow when the user makes "login" and "logout". So the developer has been entered in the user management module and modify "access redirection page" (login) and "exit redirect page" (logout), putting home screens for each menu. Thus it is totally controlled the changes templates for the own module.

![Illustration 37 - Access redirect](image1)
![Illustration 36 - Exit redirect](image2)

### 10.3 Creation of tables

In the section on analysis and specification the author explained how he had decided to implement the system database. For this reason it has been modified code PHP that came with the module to create the necessary tables.

The following code is used on all pages where exists connection to the database. The first three lines kept as variables different fields of User Table and then there is the connection to the server and database.

```php
<?php
    $usuario = JFactory::getUser()->get('name');
    $username = JFactory::getUser()->get('username');
    $userID = JFactory::getUser()->get('id');

    // Server Connect
    $conexion = mysql_connect("localhost", "root", 
""")
    or die ("Server connection Fail");

    // DDBB Name
    mysql_select_db("joomla")
    or die ("No connect DDBB");
```
To create each of the tables the developer has been added before the name a "token" that what was decided when he did the installation of Joomla.

- Creation of Obres table:

```php
//Concatenate with user to create the table
$nom_gen = "c5dl3_obres_";
$nom_part = $nom_gen . $username;

//Create if not exitst
$instruccion = "CREATE TABLE IF NOT EXISTS \$nom_part. (id INT, nom_obra VARCHAR(150), public int(1), idHistoric int(10), dataCrea VARCHAR(10), dataMod VARCHAR(10), imatge LONGBLOB)";

$crea = mysql_query ($instruccion, $conexion) or die ("Error en la creacio de la taula");
```

- Creation of Obres_Historic table:

```php
$nom_gen_his = "c5dl3_obres_historic_";
$nom_part_his = $nom_gen_his . $username;

//Create if not exitst
$instruccion = "CREATE TABLE IF NOT EXISTS \$nom_part_his. (id INT, his TEXT)";
$crea = mysql_query ($instruccion, $conexion) or die ("Error in the creation of nom_part_his table");
```

- Creation of Obres_Likes table:

```php
//Concatenate with user to create the table
$nom_gen = "c5dl3_obres_likes_";
$nom_like = $nom_gen . $username;

//Create if not exitst
$instruccion = "CREATE TABLE IF NOT EXISTS \$nom_like. (idObra INT(11), idUser INT(11))";
$crea = mysql_query ($instruccion, $conexion) or die ("Error in the creation of likes table");
```
**10.4 Personal shields**

This area explain how manage the personal shields section.

First we have an HTML with CSS classes which are associated with the “Activitat” title and a button to create a new shield.

```html
<h1 class="Titol">Activitat
<a href="novaObra.php" target="_self">
<input type="button" name="boton" value="Crear nou escut" class="boton" />
</a>
</h1>
```
The developer has created a “boton” class with the same properties as the button of management user module to maintain the uniformity in the system.

Then, we want to see a informative line with the number of shields created until now.

```css
//CSS
.numEscuts{
  color:#C10F05;
  font-size:14px;
}

//PHP
$nom_gen = "c5dl3_obres_
$nom_part = $nom_gen . $username;

//select
$instruccion=("SELECT id, nom_obra, dataCrea, dataMod, public FROM ",$nom_part." order by id DESC");
$consulta = mysql_query ($instruccion, $conexion)
or die ("Fail");

//Obres rows
$nfilas = mysql_num_rows ($consulta);
echo '<br><div class="numEscuts"> Fins ara has creat '.$nfilas.' escuts.</div><br>';
```

With all code we have the Personal Shields header like the picture below:

Illustration 38 – Personal Shields header

Then, we want to see a list of shields created, its information and a link to edit these shields.

```css
h3 {
  color:#000000;
  font: bold 20px/1.5 Helvetica, Verdana, sans-serif;
}
```
This part is fixed to all shield of wheter or not public. Here load:

- Image shield.
- Name of the work.
- A link to edit the shield.
- Created date.
- Modified date.

If the shield is marked as public or not the user can see more information.

In public case, we search idIntern field of Obres_publiques table and count how many "Like" and "Comment" has been made in the shield.

```php
//PHP
if ($resultat['public'] == 1){
    $instruccion=("SELECT idIntern FROM c5dl3_obres_publiques
                   WHERE id = ".$identificador." AND username = ".$username." ");
```
$consultaIDIntern = mysql_query ($instruccion, $conexion) or die ("Fail c5dl3_obres_publiques");
$resIDIntern = mysql_fetch_array ($consultaIDIntern);
$idIntern = $resIDIntern[0];

// Number of comment
$nom_comment = "c5dl3_obres_comment_" . $username;

$instruccion="SELECT idUser FROM ".nom_comment..TableName WHERE idObra = ".$resultat['id'].");
$consultaComment = mysql_query ($instruccion, $conexion) or die ("Fail likes");
$nfilasComment = mysql_num_rows ($consultaComment);

// Number of like
$nom_like = "c5dl3_obres_likes_" . $username;

$instruccion="SELECT idUser FROM ".nom_like.TableName WHERE idObra = ".$resultat['id'].");
$consultaLike = mysql_query ($instruccion, $conexion) or die ("Fail likes");
$nfilasLike = mysql_num_rows ($consultaLike);

We have more variables. An user can give “Like” himself and that will change how see the buttons. For this reason the code search if the user made “Like” himself.

$instruccion="SELECT idUser FROM ".nom_like.TableName WHERE idObra = ".$resultat['id'].
AND idUser = ".$userID.");
$consultaLikeUser = mysql_query ($instruccion, $conexion) or die ("Fail likes");
$nfilasLikeUser = mysql_num_rows ($consultaLikeUser);

If positive result:

//CSS
.botton-like{
    float:right;
    margin: 0;
    opacity: 0.5;
}
.botton-like:hover {
    opacity: 1;
}
.num-social{
    float:right;
}
.botton-social{
    color: #FFF;
    position:relative;
    text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
In this code we have:

- A button in "Like" status and "Ja no m'agrada" words for if the user wants to return to initial status.
- A button "Comment" that later we see what function it has.
- A button with PNG format relationated with the "Like".
- The number of "Like" has received the shield.
- A button with PNG format relationated with the "Comment".
- The number of "Comment" has received the shield.
Thus, in case of the shield is public and where the user has made "Like" himself, is as follows:

For the case where the user has not been done "Like" himself just have to change one line.

```css
.boton-social{
    color: #FFF;
    position:relative;
    text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
    background-color: #006DCC;
    border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.25);
    -webkit-border-radius: 4px;
    -moz-border-radius: 4px;
    border-radius: 4px;
    width: 110px;
}
```

```php
//PHP
else {
    echo "<div>
    <button id='boton-like'.$idIntern.' type='button' class='boton-social'
    onclick='Like('.$idIntern.').'M'agrada</button>
    </div>";
}
```

This way the "Like" button has the same class than "Comment".
In the case that the shield is not public, we make a line break to maintain the uniformity.

```php
else {
    echo '<br>';
}
```

Finally, to separate the different fields the developer has decided to show a line separator unless it is the last shield.

```css
.barra_separadora {
    border-bottom: 1px solid #000000;
    border-radius: 1px 1px 0 0;
    box-shadow: 0 13px 2px -10px rgba(0, 0, 0, 0.4);
    color: #FFFFFF;
    line-height: 20px;
    padding: 5px;
}
```

Therefore, considering the three cases mentioned, we have a final result as we can see below:

Illustration 41 – Personal shields result
10.5 Like and Comment

In the above codes we have seen a number of buttons for public shields. In this section we will see how the author has developed these buttons.

10.5.1 Like

We start with “Like” button, remember that we had inserted the code in PHP

```html
<button id='boton-like'.$idIntern.' type='button' class='boton-social'
onclick='Like($idIntern.)'&gt;M'agrada&lt;/button>
```

If you see this button uses the “onclick” function, this means that in the case of the user press the button the “Like” JavaScript code with the idIntern field of the shield. This is the code:

```javascript
//JAVASCRIPT
function Like(id) {
    jQuery(document).ready(function ($) {
        if ($("#boton-like"+id).hasClass('boton-social')) {
            $('#boton-like'+id).addClass('boton-no-magrada');
            $('#boton-like'+id).removeClass('boton-social');
            $('#boton-like'+id).html('Ja no m'agrada');
            var param = {'id':id};

            $.ajax({    //create an ajax request
                data: param,
                type: "POST",
                url: "/joomla2/like.php",
                dataType: "html", //expect html to be returned
                success: function(response){
                    $('#num-likes'+id).html(response);
                }
            });
        } else {
            NotLike(id);
        }
    });
}
```
This feature is what makes changes CSS class button and then Ajax call “like.php” with the ID. This returns a HTML that basically it will be the new value of “Like” in the shield. Is possible that in one moment the user press “Like” and without refresh the page will back the previous state the developer add the “else” that executes the function “NotLike”.

This is the Like.php code:

```php
//PHP
$id = $_POST['id'];

(instruccion="SELECT likes, username, id FROM c5dl3_obres_publiques WHERE idIntern = ".\$id. ");
$consulta = mysql_query (instruccion, $conexion)
or die ("Fail in c5dl3_obres_publiques");

$resultat = mysql_fetch_array ($consulta);
$resultat[0] = $resultat[0] + 1;

(instruccion = "UPDATE c5dl3_obres_publiques
SET likes = ".\$resultat[0]."
WHERE idIntern = "\$.id. ");
$update = mysql_query ($instruccion, $conexion)
or die ("Fail update to c5dl3_obres_publiques");

$usernameObra = $resultat[1];
$idObra = $resultat[2];

//Concatenate with the user
$nom_gen = "c5dl3_obres_likes_";
$nom_like = $nom_gen . $usernameObra;

(instruccion = "INSERT INTO ".$nom_like. VALUES("$.idObra."","$.username.");
$insert = mysql_query ($instruccio, $conexion)
or die ("Fail in the insert");

echo $resultat[0];
```

This code will update the table Obres_Publiques and will insert a line in Obres_Likes. Finally returns the value of the new number of "Like".

Otherwise follow exactly the same steps but change the instructions.

The call now is "notlike.php".

```javascript
//JAVASCRIPT
function NotLike(id){
  jQuery(document).ready(function ($) {
    if ($("#boton-like").hasClass("boton-no-magrada")) {
      $("#boton-like"+id).addClass("boton-social");
    }
  });
}
```
In this case subtract one unit to the number of "Like" and remove the line.

```php
$userID = JFactory::getUser()->get('id');

$instruccion = "SELECT likes, username, id FROM c5dl3_obres_publiques WHERE idIntern = ".$id." ";
$consulta = mysql_query ($instruccion, $conexion)
or die ("Fail in the select of likes in c5dl3_obres_publiques");

$resultat = mysql_fetch_array ($consulta);
$resultat[0] = $resultat[0] - 1;

$instruccion = "UPDATE c5dl3_obres_publiques
    SET likes = ".$resultat[0]."
    WHERE idIntern = ".$id." ";
$update = mysql_query ($instruccion, $conexion)
or die ("Fail in the update to c5dl3_obres_publiques");

$usernameObra = $resultat[1];
$idObra = $resultat[2];
//Concatenate with the user
$nom_gen = "c5dl3_obres_likes_";
$nom_like = $nom_gen . $usernameObra;

$instruccio = ("DELETE FROM ".$nom_like." WHERE idObra = ".$idObra." AND idUser = ".$userID." ");
$delete = mysql_query ($instruccio, $conexion)
or die ("Fail in the DELETE");

echo $resultat[0];
```
10.5.2 Like consultation

In this section, the developer explains how developed the functionality to query users who have made "Like" to shield public.

This process starts with an image that it makes the function of button, you can see below:

Illustration 42 – Like consult image

To be noted that the user can interact with this button the developer has modified the CSS class so when the user hover its mouse over the image becomes darker, as we compare the following images:

Illustration 43 – Comparation cursors button

This is the code of the button:

```html
<input type='image' class='boton-like' src='/joomla2/images/boto_like.png' onclick='buscaLike(".$idIntern.")' />
```

When the user clicks on the image run the following function "buscaLike".

```javascript
//JAVASCRIPT
function buscaLike(id){
    document.getElementById('fade').style.display='block';
    document.getElementById('light').style.display='block';

    jQuery(document).ready(function ($) {
        var param = {"id":id};

        $.ajax({
            //create an ajax request to load_page.php
            data: param,
            type: "POST",
            url: "/joomla2/display.php",
            dataType: "html", //expect html to be returned
            success: function(response){
                $('#msgid').html(response);
            }
        });
    });
}
```
This code activates a hidden field HTML with a CSS class associated that darkens the screen and open a small window where the user can see the users that have made "Like" with "msgid" field, this is where display.php return a HTML code.

To return to the general state the user can click the close button.

```css
.overlay{
    display: none;
    position: fixed;
    top: 0%;
    left: 0%;
    width: 100%;
    height: 100%;
    background-color: black;
    z-index:1001;
    -moz-opacity: 0.5;
    opacity:.50;
    filter: alpha(opacity=50);
}
.modal {
    display: none;
    position: fixed;
    top: 25%;
    left: 25%;
    width: 50%;
    height: 50%;
    padding: 16px;
    border-radius:10px;
    background-color: white;
    z-index:1002;
    overflow: auto;
}
.boto-tanca{
    float:right;
}
```

//HTML
```
<div id='fade' class='overlay'></div>
<div id='light' class='modal'>
    <a class="boto-tanca" href="javascript:void(0)" onclick = "tancaModal()">Tanca</a>
    <div id="msgid"></div>
</div>
```

Now, we can see what returns display.php:

```css
#container
{
    width: 350px;
}
```
```php
$idIntern = $_POST['id'];

$instruccin = "SELECT id, username FROM c5dl3_obres_publiques WHERE idIntern = ".$idIntern." ");
$consulta = mysql_query ($instruccin, $conexion)
or die ("Fail to Select c5dl3_obres_publiques");

$resultat = mysql_fetch_array ($consulta);

//Concatenate with the users
$nom_gen = "c5dl3_obres_likes_";
$nom_like = $nom_gen . $resultat[1];

$instruccin = "SELECT idUser FROM ".$nom_like." WHERE idObra = ".$resultat[0]." ");
$consulta = mysql_query ($instruccin, $conexion)
or die ("Fail in select likes");

$nfilas = mysql_num_rows ($consulta);

echo '<body>
    <div id="container">
        <div id="comments">

        for ($i=1; $i<=$nfilas; $i++){
```
$resultat = mysql_fetch_array ($consulta);

$instruccion="SELECT  username, id FROM c5dl3_users WHERE id = ".$resultat[0]." ");
$consultaUser = mysql_query ($instruccion, $conexion) or die ("Fail in select username");

$resultatUser = mysql_fetch_array ($consultaUser);

echo '<div class="comment">
    <div class="avatar">
        <img src="/joomla2/imatge.php?id='.$resultatUser[1].'" />
    </div>
    <div class="autoComment">
        <span>'.$resultatUser[0].' </span>
    </div>
</div>
';

};

For each user who made "Like" the shield that was being viewed the code go to Users Table searching for the username and ID through its load the image of the avatar using "imatge.php" code. All this information appears in containers made with CSS. As we can see in the image below as an example.
10.5.3 Comment

The “Comment” button is used to view the comments made. Once inside the new window the user can insert new comments or delete old comments. So these two buttons perform the same function.

If we see the codes we can check that in any press the function “buscaComment” will be executed.

```php
//PHP
//Blue button
<button type='button' class='boton-social'
onclick='buscaComment(".$idIntern.")'>Comentar</button>

//Image
<input type='image' class='boton-like' src='/joomla2/images/boto_comment.png'
onclick='buscaComment(".$idIntern.")' />
```

The code of these buttons is the same than “Like” with a little difference.

```javascript
//JAVASCRIPT
function buscaComment(id){
    document.getElementById('fade').style.display='block';
    document.getElementById('light-comment').style.display='block';

    jQuery(document).ready(function ($) {
        var param = {"id":id);

        $.ajax({  //create an ajax request
            data: param,
            type: "POST",
            url: "/joomla2/display-comment.php",
            dataType: "html", //expect html to be returned
            success: function(response){
                $('#msgid-comment').html(response);
                $('#auxhidden').text(''+id+'');
            }
        });
    });
}
```
This code searches in "display-comment.php" and returns the HTML content that is written in "msgid-comment" field.

This is the "display-comment.php" code.

```php
$idIntern = $_POST['id'];

$instruccion = "SELECT id, username FROM c5dl3_obres_publiques WHERE idIntern = ";
$consulta = mysql_query ($instruccion, $conexion)
or die ("Fail in select to c5dl3_obres_publiques");

$resultat = mysql_fetch_array ($consulta);

//Concatenate with user
$nom_gen = "c5dl3_obres_comment_";
$nom_comment = $nom_gen . $resultat[1];

$instruccion = "SELECT idUser, comment, id FROM ";
$consulta = mysql_query ($instruccion, $conexion)
or die ("Fail in select to comment");

$nfilas = mysql_num_rows ($consulta);

for ($i=1; $i<=$nfilas; $i++){
    $resultat = mysql_fetch_array ($consulta);
    $idLinia = $resultat[2];

    $instruccion = "SELECT username, id FROM c5dl3_users WHERE id = ";
    $consultaUser = mysql_query ($instruccion, $conexion)
or die ("Fail in select to user");
```
$resultatUser = mysql_fetch_array ($consultaUser);

echo '<div id="borrar'.$i.'" class="comment">';

if ( $resultatUser[0] == $usernameExe) {
    echo '<input type="image" class="boto-delete" src="/joomla2/images/boto_delete.png" onclick="Elimina('.$i.','.$idIntern.', '.$idLinia.')" />
';
}

echo '<div class="avatar">' .<img src="/joomla2/imatge.php?id='.$resultatUser[1].'" /> </div>' .<div id="acom'.$i.'" class="autoComment">
    <span id="sp'.$i.'">'.$resultatUser[0].'</span>
    <div id="comentari'.$i.'">'.$resultat[1].'</div>
</div>' .</div>' ;

}

Until now the code is identical than "Like" except that the developer has added an image to delete the comment in the case that the comment was made by itself. This button also darken in the case that the user hover the cursor over it. The other difference is that the developer has added a new line with the comment (<div id="comentari'.$i.'">'.$resultat[1].'</div> ).

Illustration 46 – Comments

In the case that the user clicks to delete a comment, run the “Elimina” function:

```php
function Elimina(id, idIntern, idLinia){
    jQuery(document).ready(function(){
        var param = {"idLinia":idLinia, "idIntern":idIntern};
        $.ajax{
            data: param,
            type: "POST",
            url: "/joomla2/delete-comment.php",
        }
    }
}
```
"Delete-comment.php" is a very simple code that removes the line of Obres_Comment table. The final function "remove" removes the container completely.

```php
$idLinia = $_POST['idLinia'];
$idIntern = $_POST['idIntern'];

$instruccio =("SELECT username FROM c5dl3_obres_puliques WHERE idIntern = ".$idIntern." ");
$consulta = mysql_query ($instruccio, $conexion)
or die ("Fain in select to c5dl3_obres_puliques");

$resultat = mysql_fetch_array ($consulta);

//Concatenate with the user
$nom_gen = "c5dl3_obres_comment_";
$nom_comment = $nom_gen . $resultat[0];

$instruccio =("DELETE FROM ",.$nom_comment." WHERE id = ".$idLinia." ");
$delete = mysql_query ($instruccio, $conexion)
or die ("Fain in delete comment");

echo "OK";
```

The main difference with the code of "Like" is that if the user has made "login" (remember that one who has not done can also see the comments) is added to a container at the end side in order to the user can insert a new comment.

```css
#commentBox
{
    background: #edeff4;
    height: 25px;
    overflow: hidden;
    display: block;
    padding: 20px;
}
#postComment input
{
    width: 100%;
}
```
This code shows a container like the image below.

Illustration 47 – New comment container

If the user interacts with this container runs the “Comment” function.

```php
function Comment(id, nfilas, event){
    if(event.keyCode === 13){
        var aux = $('#newComment').val();
        nfilas = nfilas + 1;

        jQuery(document).ready(function(){
            var param = {"idIntern":id, "newComment":aux };
            $.ajax({ //create an ajax request to load_page.php
                data: param,
                type: "POST",
                url: "/joomla2/comment.php",
                dataType: "json",
                success: function(data){
                    var comment = $('<div id=borrar'+nfilas+'></div>').addClass('comment');
                    var avatar = $('</div>').addClass('avatar');
                    var img = $('</img/>').attr({'src':''+data[2]+'});
                    var text = $("<div></div>").addClass('autoComment').html('<span>'+ data[0] + '</span>"+"<div>'+$('#newComment').val()+'</div>"';
                    var comment = $('</div>').addClass('autoComment').html('"<div>'+data[0]+'</div>"');
                }
            });
        });
    }
}
```
This code makes a call to "comment.php" where it will perform an insert of a new line, but the difference is that now the code returns a JSON document. JSON is a standard based on the design to sharing simple structures data and lists [18]. This way, the value that returns is used to insert a new container like those taken to put PHP code and see available a new container to add a new comment.

This the "comment.php" code.

```
<?php

$idIntern = $_POST['idIntern'];
$instruccion="SELECT  username, id FROM c5dl3_obres_publiques WHERE idIntern = "$idIntern." ";
$consulta = mysql_query ($instruccion, $conexion)
or die ("Fail in select likes to c5dl3_obres_publiques");
$resultat = mysql_fetch_array ($consulta);

$usernameObra = $resultat[0];
$nom_gen = "c5dl3_obres_comment_";
$nom_comment = $nom_gen . $usernameObra;

$instruccio = "SELECT MAX(id) FROM "$nom_comment." ";
$consulta = mysql_query ($instruccio, $conexion)
or die ("Fail in select to comment");
$resId = mysql_fetch_array ($consulta);
$id = $resId[0] + 1;

$instruccio = "INSERT INTO ".$nom_comment." VALUES('".$resultat[1]."','".$username."',".$comment. ",".$id."');
```
$insert = mysql_query ($instruccio, $conexion) or die ("Fail in insert");

$src = "/joomla2/imatge.php?id=";

$arr = array();
$arr[0] = $usernameExe;
$arr[1] = $id;
$arr[2] = $src;

echo json_encode($arr);

The first part of the code is responsible for inserting a new line to Obres_Comment table, the end part is responsible for return an array with the username, id of the public shield and the path to load the user avatar.

Illustration 48 – Consult comments and container

Finally, we have the last difference with the “M’agrada” code. When the user close the window we want to update the number of comments made to the shield. For this reason the developer has added this code that execute “tancaModal()”.

//HTML
<div id='light-comment' class='modal'>
    <a class="boto-tanca" href="javascript:void(0)" onclick = "tancaModal()">Tanca</a>
    <div id="msgid-comment" style="padding-bottom: 10px;"></div>
</div>
The first part hides the window of "Comment" or "Like" to return to the original state. "Num_comment.php" is a very simple code that returns the new value of comments received.

```php
$aux = $_POST['aux'];
$instruccio="SELECT id, username FROM c5dl3_obres_publiques WHERE idIntern = "+$aux." ";
$consulta = mysql_query ($instruccio, $conexion)
or die ("Fail in select to c5dl3_obres_publiques");

$resultat = mysql_fetch_array ($consulta);

//Concatenate with user
$nom_gen = "c5dl3_obres_comment_";
$nom_comment = $nom_gen . $resultat[1];

$instruccio="SELECT idUser, comment, id FROM ".$nom_comment." WHERE idObra = "+$resultat[0]." ";
$consulta = mysql_query ($instruccio, $conexion)
or die ("Fail in select to comment");

$nfilas = mysql_num_rows ($consulta);
echo $nfilas;
```
10.6 Publics shields

The operation of public shields is almost same as for the personal shields with some little modifications.

The first difference is the information that we get for each shield. In the personal shields we had a link to edit the shield that now not appears. Now appears new information with the username of the user who created the shield. This is the line <h6 class='Creador'>Creat per: ".$username." </h6>.

```php
//PHP
//Load obles_puliques
$instruccion="SELECT idIntern, id, username FROM c5dl3_obres_puliques WHERE public = 1 order by idIntern DESC";
$consulta = mysql_query ($instruccion, $conexion) or die ("Fail in select to c5dl3_obres_puliques");

$nfilas = mysql_num_rows ($consulta);
for ($i=1; $i<=$nfilas; $i++){
    $resultat = mysql_fetch_array ($consulta);
    $idIntern = $resultat[0];
    $identificador = $resultat[1];
    $username = $resultat[2];
    $nom_part = $nom_gen . $username;
    //select to get shields info
    $instruccion="SELECT id, nom_obra, dataCrea, dataMod FROM "$nom_part." WHERE id = "$identificador." ";
    $consultaUser = mysql_query ($instruccion, $conexion) or die ("Fail in select user");
```
$resultatUser = mysql_fetch_array ($consultaUser);

// likes table
$nom_like = $nom_gen_likes . $username;

$instruccio=("SELECT idUser FROM ". $nom_like." WHERE idObra = ". $resultatUser['id'] . ");
$consultaLike = mysql_query ($instruccio, $conexion) or die ("Fallo en la consulta de likes");
$nfilasLike = mysql_num_rows ($consultaLike);

// comment table
$nom_comment = $nom_gen_comment . $username;

$instruccio=("SELECT idUser FROM ". $nom_comment." WHERE idObra = ". $resultatUser['id'] . ");
$consultaComment = mysql_query ($instruccio, $conexion) or die ("Fallo en la consulta de likes");
$nfilasComment = mysql_numRows ($consultaComment);

// If it is not connected
if ($userID == NULL){
    echo"<li>
        <img src="imatge_obra.php?id_almost_equal_userid&user='%s'" height="120" width="120" />
        <h6 class='Creador'> Creat per: %s</h6>
        <h3 class='h3'> %s</h3>
        <p>Data creació: %s</p>
        <p>Data última modificació: %s</p>
        <div>
            <p class='num-social'>%d</p>
            <input type='image' class='boton-like' src='/joomla2/images/boto_like.png' onclick='buscaLike("%d")'/>
            <p class='num-social'>%d</p>
            <input type='image' class='boton-like' src='/joomla2/images/boto_like.png' onclick='buscaLike("%d")'/>
        </div>
    </li>
    <br>
    if ($i!=$nfilas){
        echo '<div class="barra_separadora">'</n
    }
} // If it is connected
else {
    ... //Same code from Personal shields
}
Another difference is when the user has made "login" or it is as a visitor. Remember that the visitors can see the public shields with the option of seeing the users who made "Like" and commented the shield. For this reason also the developer controls each case because if the person has made "login" is just the same that personal shields section while if a visitor does not shows.

Comparison of the results:

**Escuts Públics**

Illustration 50 - Connected user in public shields

Illustration 49 – Not connected user in public shields.

And when the user opens the comments:

Illustration 51 – Comments for connected users

Illustration 52 – Comments for not connected users
10.7 Profile

Finally, we have a first part with information as:

- Avatar.
- Username.
- Number of created shields.
- Total number of "Likes".

It is possible that the user has not change never his image and the author wants to have it, the system assigns a default image.

```php
// PHP
$nom_gen = "c5dl3_obres_";
$nom_part = $nom_gen . $username;
$instructuion=("SELECT id FROM ". $nom_part . "");
$consulta = mysql_query ($instructuion, $conexion) or die ("Fail in the select");
$nEscuts = mysql_num_rows ($consulta);

// Number of likes
$nom_gen_likes = "c5dl3_obres_likes_";
$nom_like = $nom_gen_likes . $username;
$instructuion=("SELECT idUser FROM ". $nom_like . "");
$consultaLike = mysql_query ($instructuion, $conexion) or die ("Fail in the select of likes");
$nLike = mysql_num_rows ($consultaLike);
```

```html
// HTML
<h1 class="Titol">Perfil</h1>
<br>
<div class="intro"> Avatar actual:</div>
<br>
```
This code shows information part like the image below.

### Perfil

**Avatar actual:**

Now, the developer inserts a vertical bar and a space to be able to change the avatar.

Before the modification, the code checks if the new image has gif, jpeg, png or pjpeg format and that the image not exceeding 20 KB.

```php
if(isset($_POST['boton'])){  
    // Condicionem la nova imatge a tipo gif, jpeg, png i pjpeg. Tampoc pot superar 20 KB
    if ((($_FILES['archivo']['type'] == "image/gif") ||
        ($_FILES['archivo']['type'] == "image/jpeg") ||
        ($_FILES['archivo']['type'] == "image/png") ||
        ($_FILES['archivo']['type'] == "image/pjpeg")) &&
        ($_FILES['archivo']['size'] < 20000)) {
        if ($_FILES['archivo']['error'] > 0) {
```
```php
$imagen_temporal = $_FILES['archivo']['tmp_name'];
//tipu arxiu (png, jpeg...)
$tipo = $_FILES['archivo']['type'];
$fp = fopen($imagen_temporal, 'r+b');
$data = fread($fp, filesize($imagen_temporal));
fclose($fp);

//jump characters
$data = mysql_escape_string($data);

//See if the user update
$instruccion = "UPDATE c5dl3_users_avatar SET avatar = '".$data."', tipo = '".$tipo."'
WHERE id = '".$userID."’;
$insert = mysql_query ($instruccion, $conexion)
or die ("Fail in update avatar");
```

The result:

**Perfil**

Avatar actual

[Avatar Image]

Nom d'usuari: joanlaporta

Nº Escuts creats: 3

Nº Total de m'agradè: 2

Puja una imatge:

Seleccionar archivo

Ningún archivo seleccionado

Puja
10.8 Catalan Heraldry

To create the Catalan Heraldry guide the developer has decided to insert text directly into HTML paragraphs and uploading images identified by a header. This paragraph will be visible depending on user clicks. So that it will manage a JavaScript function that shows and hides paragraphs per user click.

```html
//CSS (News)
.STitol{
  color:#C10F05;
  font-size: 12px;
}

//HTML
<h1 class="Titol">Heràldica Catalana</h1>

<h2 class="STitol" onClick="obre(1)">Origen de l'heràldica i primeres armories</h2>
<p>L'heràldica és ... //words</p>
<p>... //words</p>
<img src="/images/heraldica/fig_naturals.jpg" width="615" height="493" />
<h2 class="STitol" onClick="obre(2)" href="#2" name="2">L'heràldica cívica</h2>
<div id=s2 style="text-align:justify">
...
</div>
<h2 class="STitol" onClick="obre(3)">Classes d'esmalts</h2>
<div id=s3 style="text-align:justify">
...
</div>

//JAVASCRIPT
function obre(id){
  document.getElementById('s1').style.display='none';
  document.getElementById('s2').style.display='none';
  document.getElementById('s3').style.display='none';
  document.getElementById('s4').style.display='none';
  document.getElementById('s5').style.display='none';
  document.getElementById('s6').style.display='none';
  document.getElementById('s7').style.display='none';
  jQuery(document).ready(function ($) { 
    if ( $('#num').text() == id){
      document.getElementById('s'+id+').style.display='none';
      if ( $('#num').text() == $('#num2').text()){
        document.getElementById('s'+id+').style.display='block';
        $('#num').text('0');
      }
    }
  });
}
Below an example of opening Enamels classes section.

![Illustration 55 - Catalan heraldry guide](image)

### 10.9 Tool

For the development of heraldry editing tool the developer has made an FLASH application with ActionScript (AS3) code. Much of the previous work to the introduction the code has been the draw of the architecture of the screen, This includes drawing lines, squares, buttons, lists...
10.9.1 Start

Once the page loads the tool makes a number of actions.

Create vectors with:

- Historic: History of actions performed by the user.
- HistoricParticions: History with the divisions that the user has pressed.
- HistoricFiguresSprite: History figures are managed (created, modified, deleted) the user.
- HistoricTimbres: History with the crowns that the user has pressed.

Load and situate the buttons for each functionality that interact with the user by buttons, as in the example.

Illustration 56 – Buttons and situation

Load and situate two Bitmaps. The reason to create the Bitmaps is simple, one is for the part that includes the outline of the shield, while the other includes the outline and the part where the tool inserts the crown. The latter is the bitmap saved when the user finish the work.

Illustration 57 - Bitmap2

Illustration 58 - Bitmap1
Locate correctly the buttons related to editing tools. The buttons except the first and last have a particularity, depending the option selected by the user will appear more dark. This option is not available in Flash, so the developer has duplicated the buttons and now we have the same buttons in clear version and dark version. The developer manage the control of these buttons in order to show or hidden each buttons based on user interaction.

Illustration 59 - Editing tools

Then load the background of the shield. The load background function paint on Bitmap1 using the draw AS3 function the image marked as background.

Also initializes a series of global variables that are user to have the control of user interaction:

- idHistoric: Action ID history
- checkPublicStatus: Boolean that stores the information if the shield is public or not.
- idCanvas: Sprites ID of the figures.

Finally once already loaded the global configuration, the code calls a function that makes a query to a PHP code to consult if the user is editing the shield or new.

```javascript
// Function to connect DDBB
function cargaBBDD():void
{
    cargador = new URLLoader();
    cargador.addEventListener(Event.COMPLETE, dadesCargades);
    cargador.dataFormat = URLLoaderDataFormat.VARIABLES;
    cargador.load(new URLRequest("cargaBBDD.php"));
}
```
This code reads if the user is editing using the "edicio" field in Users table. In the case that the result is 0 this means that the user is creating a new work. Otherwise goes to Obra table to search the name and Obres_Historic table to load the string with the content of actions performed by the user. Later, in other section, the author explains how to manage the values of this string.

```php
// Search if the user is editing
$instruccion = "SELECT edicio FROM c5dl3_users WHERE username = '" . $username . "";";
$consulta = mysql_query ($instruccion, $conexion) or die ("Fail in the select to users");
$res = mysql_fetch_row($consulta);

$id_obra = $res[0];
if ($id_obra == 0) {
    $nom = 'Nova obra';
    $public = '';
    $historic = '';
} else {
    // Concatenate with the user
    $nom_gen = "c5dl3_obres_";
    $nom_part = $nom_gen . $username;
    // Load
    $instruccion = "SELECT nom_obra, public FROM "$nom_part." WHERE id = " . $res[0] . "";";
    $consulta = mysql_query ($instruccion, $conexion) or die ("Fail in select 1");
    $res = mysql_fetch_row($consulta);
    $nom = $res[0];
    $public = $res[1];

    // Concatenate with the user
    $nom_gen_his = "c5dl3_obres_historic_";
    $nom_part_his = $nom_gen_his . $username;
    // Load
    $instruccion = "SELECT his FROM "$nom_part_his." WHERE id = "$id_obra."";";
    $consulta = mysql_query ($instruccion, $conexion) or die ("Fail in select 2");
    $res = mysql_fetch_row($consulta);
    $historic = $res[0];
}
echo "nom=$nom&public=$public&historic=$historic";
```

Before the complete of this starting part the developer also has inserted the function that checks if the shield is marked as public or not.
If the user clicks on the button will appear a new window to decide where the user save the PNG image. This is the same image like in Save case.

10.9.2 Menu controls

We have three big Menus (Divisions, Figures and Crowns), these also have other submenus that finally show to user the buttons available for interact with the tool. The developer has inserted the code that manage the buttons to hide/show depending on user interaction.

Below, you can see the submenus of each menu

- Divisions -> Basic

Illustration 61 – Basic divisions

- Divisions -> Irregular

Illustration 62 – Irregular divisions

- Figures -> Naturals. The developer has divided the figures into two sections. The user can view the white button on the right and interact with it.

Illustration 63 - Naturals figures I
And returned to the state before with white button on the left.

Illustration 64 - Naturals figures II

- Figures -> Artificials

Illustration 65 - Artificials figures

- Figures -> Imaginaries

Illustration 66 - Imaginaries figures

- Figures -> Buildings

Illustration 67 - Buildings figures

- Crowns -> Nobles

Illustration 68 - Nobles crowns I
10.9.3 Divisions

This section explains how works the system that manage the divisions. Each division button has an ID that identify it. The code that manage is the same for all partitions, the only difference is the ID. In the case that the user clicks a division the system follow the next steps:

- The system add the action to the Historic vector (with the ID of the vector, number two to mark that is a division interaction and the ID of the division).
- The system add the action to the HisParticions vector (with the ID of the Historic vector and the ID of the division).
- The system add the action to the history (transmits 'Part' and the ID of the division).
- The system executes “carregaParticio” function. This function paint on the Bitmap1 an image that has stored in the library. We can see an example.
This way, the system creates the division on the bitmap. In order to not mix the division performed so far, the system executes a reset function, the system runs the historic data and only paints the division in the case that it has the same ID like the last position on hisParticions vector.

![Illustration 73 – Division result](image)

### 10.9.4 Figures

This section explains how the system manages the figures. Each of the figures has a button that identifies them. The code that manages the figures are the same for all figures, the only difference is the ID. If the user presses a figure follow these steps:

- The system creates and Sprite, an element in order to be treated in AS3, that is an Bitmap with the loaded image from the library.
- ID is assigned to Sprite, because the user can load the same figure more than once.
- The interaction is added to historic vector (with the ID of vector historic, a number 31 that is the mark of new figure, the ID of the figure, the ID of the Sprite and position (x,y) that will be the position of the Sprite in the window).

![Illustration 74 – Figure creation](image)
- The system adds the interaction to hisFiguresSprite vector (with the ID of the historic, the own sprite and the ID of the Sprite).
- The system add the action to the history (transmits 'NovaFigura' and the ID of the figure).

![Illustration 75 – New figure history](image)

- The system assigns the function to sprite in order to the user can drag the figure on the screen. The system controls the position (x,y) each time that the user drop the figure. In the case that the user drags and drop the figure above the trash the system follow the next steps:
  - Hide the sprite, this is not deleted because later it is difficult to recover later. This way, only put visible again.
  - The system adds to historic vector (with the ID of the historic vector, number 33 that is the mark of deleted figure, the ID of the figure and the ID of the Sprite).
  - The system add the action to the history (transmits 'EliminaFigura' and the ID of the figure).

![Illustration 76 – Delete figure history](image)

If the user drags and drop in other section the system runs:

- The system adds to historic vector (with the ID of the historic vector, number 32 that is the mark of move figure, the ID of the figure, the ID of the Sprite and the new position (x,y) that finish the figure in the window).
- The system add the action to the history (transmits 'MouFigura' and the ID of the figure).

![Illustration 77 – Move figure history](image)
10.9.5 Crowns

This section explains how the system manages the crowns. Each of the crowns has a button that identifies. The code that manages the crowns are the same for all crowns, the only difference is the ID. If the user presses a crown follow these steps:

- The system loads the image from the library, create a Sprite to be managed and it is placed on top of the outline of the shield.
- The system adds to historic vector (with the ID of the historic vector, number 4 that is the mark of crown and the ID of the crow).
- The system adds to hisTimbres vector (with ID history and own Sprite).
- The system adds the action to the history (transmits ‘Timbre’ and the ID of the crown).
- The tool only allows one crown in the shield. For this reason the system executes a function that places the Sprites of the hisTimbres vector as hidden except the last. This way, the system always shows the last crown selected.
10.9.6 Lines

This section explains how the system manages the lines to form a new division does not exist. In order to add new lines the user must press the button of the line in the editing tools area.

The user can draw lines on the shield now. An important thing of the system is that allow the user draw a new line outside the shield and this is not paint because the system repaints the outside of the shield and this part of line does not appear.

When the user draw the line:

- The system creates an Sprite in the shape of the line and is painted on Bitmap1.
- The system adds to historic vector (with the ID of the historic vector, number 5 that is the mark of new line, the origin position \((x,y)\) and finish position \((x,y)\)).
- The system adds the action to the history (transmits ‘novaLinea’).
- The system repaints the outside of the shield.

![Illustration 81 – Line](image)

### 10.9.7 Paint

For the paint part, the first action performed by the developer was the limitation of the enamels that will be selectable in the colorPicker. Only those colours that are part of the Catalan Heraldry.

So, the developer has searched their hexadecimal values:

<table>
<thead>
<tr>
<th>Enamels</th>
<th>Hexadecimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azure</td>
<td>0x003399</td>
</tr>
<tr>
<td>Gulas</td>
<td>0xFF0000</td>
</tr>
<tr>
<td>Sinople</td>
<td>0x00FF00</td>
</tr>
<tr>
<td>Porple</td>
<td>0x330099</td>
</tr>
<tr>
<td>Sable</td>
<td>0x000000</td>
</tr>
<tr>
<td>Gold</td>
<td>0xFFF000</td>
</tr>
<tr>
<td>Silver</td>
<td>0x999999</td>
</tr>
</tbody>
</table>

Table 35 – Hexadecimal values enamels

![Illustration 82 – ColorPicker](image)
The system creates a variable that is storing the current value of the ColorPicker.

In order to paint the first step that the user must make is press the paint button in the editing tools area.

Once positioned the system paints the color that has the colorPicker in the area that includes the exact point where the cursor was located when was press by the user. The system follow the next steps:

- Paint the area using “floodFill” AS3 function with the colour and the position (x,y) of the cursor.
- Add to historic vector (with the ID of the historic vector, number 6 that is the mark of new paint are, the position (x,y) and the code of the colour).
- The system add the action to the history (transmits ‘Pint’ and the colour code).
10.9.8 History

This section explains the configuration of the historic vector and history that appears at the bottom right of the software arms.

We start talking about history. So far we have seen that all features calls to this history through coding. We see how this encoding and what is the result.

- Divisions

<table>
<thead>
<tr>
<th>Code</th>
<th>ID</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part</td>
<td>1</td>
<td>Partició - Partit</td>
</tr>
<tr>
<td>Part</td>
<td>2</td>
<td>Partició - Truncat</td>
</tr>
<tr>
<td>Part</td>
<td>3</td>
<td>Partició - Trinxat</td>
</tr>
<tr>
<td>Part</td>
<td>4</td>
<td>Partició - Tallat</td>
</tr>
<tr>
<td>Part</td>
<td>5</td>
<td>Partició - Quarterat creu</td>
</tr>
<tr>
<td>Part</td>
<td>6</td>
<td>Partició - Quarterat sator</td>
</tr>
<tr>
<td>Part</td>
<td>7</td>
<td>Partició - Tercejat pal</td>
</tr>
<tr>
<td>Part</td>
<td>8</td>
<td>Partició - Tercejat faixa</td>
</tr>
<tr>
<td>Part</td>
<td>9</td>
<td>Partició - Tercejat banda</td>
</tr>
<tr>
<td>Part</td>
<td>10</td>
<td>Partició - Tercejat barra</td>
</tr>
<tr>
<td>Part</td>
<td>11</td>
<td>Partició - Dextrat</td>
</tr>
<tr>
<td>Part</td>
<td>12</td>
<td>Partició - Sinistrat</td>
</tr>
<tr>
<td>Part</td>
<td>13</td>
<td>Partició - Flanquejat</td>
</tr>
</tbody>
</table>

Table 36 – Division code

- Figures

For figures the developer uses a header after a final will be accompanied by the ID.

<table>
<thead>
<tr>
<th>Code</th>
<th>ID</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>NovaFigura</td>
<td>X</td>
<td>Imatge</td>
</tr>
<tr>
<td>MouFigura</td>
<td>X</td>
<td>Mogut</td>
</tr>
<tr>
<td>EliminaFigura</td>
<td>X</td>
<td>Eliminat</td>
</tr>
</tbody>
</table>

Table 37 – Figures header
<table>
<thead>
<tr>
<th>Code</th>
<th>ID</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>1</td>
<td>- Oreneta</td>
</tr>
<tr>
<td>X</td>
<td>2</td>
<td>- Xoriguer</td>
</tr>
<tr>
<td>X</td>
<td>3</td>
<td>- Lleó</td>
</tr>
<tr>
<td>X</td>
<td>4</td>
<td>- Cérvol</td>
</tr>
<tr>
<td>X</td>
<td>7</td>
<td>- Ulls</td>
</tr>
<tr>
<td>X</td>
<td>8</td>
<td>- Agnus Dei</td>
</tr>
<tr>
<td>X</td>
<td>9</td>
<td>- Coloma</td>
</tr>
<tr>
<td>X</td>
<td>10</td>
<td>- Llagosta</td>
</tr>
<tr>
<td>X</td>
<td>11</td>
<td>- Sol</td>
</tr>
<tr>
<td>X</td>
<td>12</td>
<td>- Mà</td>
</tr>
<tr>
<td>X</td>
<td>13</td>
<td>- Moll</td>
</tr>
<tr>
<td>X</td>
<td>14</td>
<td>- Peix</td>
</tr>
<tr>
<td>X</td>
<td>15</td>
<td>- Bou</td>
</tr>
<tr>
<td>X</td>
<td>16</td>
<td>- Maça</td>
</tr>
<tr>
<td>X</td>
<td>17</td>
<td>- Tupins</td>
</tr>
<tr>
<td>X</td>
<td>18</td>
<td>- Espasa</td>
</tr>
<tr>
<td>X</td>
<td>19</td>
<td>- Calders</td>
</tr>
<tr>
<td>X</td>
<td>20</td>
<td>- Roda</td>
</tr>
<tr>
<td>X</td>
<td>21</td>
<td>- Griu</td>
</tr>
<tr>
<td>X</td>
<td>22</td>
<td>- Drac</td>
</tr>
<tr>
<td>X</td>
<td>23</td>
<td>- Castell</td>
</tr>
<tr>
<td>X</td>
<td>24</td>
<td>- Torre</td>
</tr>
<tr>
<td>X</td>
<td>25</td>
<td>- Domus</td>
</tr>
<tr>
<td>X</td>
<td>26</td>
<td>- Església</td>
</tr>
<tr>
<td>X</td>
<td>27</td>
<td>- Palau</td>
</tr>
<tr>
<td>X</td>
<td>28</td>
<td>- Santuari</td>
</tr>
<tr>
<td>X</td>
<td>29</td>
<td>- Muralla</td>
</tr>
<tr>
<td>X</td>
<td>30</td>
<td>- Ermita</td>
</tr>
</tbody>
</table>

Table 38 – Figures code
In the history part we have the “Undo” button, this is the first button in the editing tools area.

When the user clicks this button the system search the last row of the “historic” vector and depends the code the system will make some actions.
This is the codification of this vector:

<table>
<thead>
<tr>
<th>Code</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Division</td>
</tr>
<tr>
<td>31</td>
<td>New figure</td>
</tr>
<tr>
<td>32</td>
<td>Move figure</td>
</tr>
<tr>
<td>33</td>
<td>Delete figure</td>
</tr>
<tr>
<td>4</td>
<td>Crowns</td>
</tr>
<tr>
<td>5</td>
<td>Lines</td>
</tr>
<tr>
<td>6</td>
<td>Paint</td>
</tr>
</tbody>
</table>

Table 42 – Historic actions code

In the case the user clicks “Undo”:

- Deletes the last line of the history.
- In case of the code is 2 the system runs pop function on hisParticions vector.
- In case of the code is 31 the system hide the figure with the ID code of the Sprite.
- In case of the code is 4 the system runs pop function on hisTimbres vector.
- Runs pop function on historic vector.
- Resets the process executing the historic actions because the result can be very different now.

The other option is that the user deletes any line of the history. In this case the system follows a similar process with some differences:

- Deletes the line selected in the history.
- In case of the code is 2 runs “splice” (delete a line of vector) of hisParticions vector with the same ID of historic vector.
- In case of the code is 31 hide the figure that has the same ID Sprite like the line to delete.
- In case of the code is 4 runs “splice” of hisTimbres vector with the same ID of historic vector.
- Runs splice of the historic vector.
- Resets the process executing the historic actions because the result can be very different now.
10.9.9 Save

If the user press the save button the system generates a number of actions in order to interact with the database. This process takes a little period of time but enough for the user thinks that the tool does not respond the request.

For this reason, the systems shows an image as the below:

![Illustration 85 – Loading image](image)

The next step consists in the build of the historic vector, this vector is saved in Obres_historic table. The structure must be a string. The solution proposed by the developer was keep the format of rows and columns. The process puts together the fields of rows with "-" as a separator and also puts together the columns with "+" as a separator. This structure has been implemented through the "join" AS3 function.

This is an example of the result:

```
1-2-2+2-6-439-443-443+573+3-6-170-206-65289+4-659-
232-15776969+5-6-110-81-13105+6-31-3+1-75-250+7-31-3-
1-478-256+8-31-22-2-1-75-250+9-32-22-2-334-295+10-31-21-
3-75-250+11-32-21-3-339+435+12-31-25-4+75-250+13+32-
21-3-4-468-449+14-4+10|
```

![Illustration 86 – Historic String Example](image)

Once the developer has already prepared the string, the system prepares the image to be stored in the database and displayed on personal shields and public shields. For this reason the first step is take those figures availables in hisFiguresSprite vector and paint them with the "draw" AS3 function in Bitmap1. Later, the system takes the Bitmap1 and paint also with the "draw" AS3 function in Bitmap2. Then, the system takes the last Sprite on hisTimbres (the current crown) and draw in Bitmap2. Finally, the system paint the areas outside the shield of white color with "floodFill" AS3 function in order to adapt the image to aesthetics of the website. To save in the database the developer use the PNGEncoder package with "encode" function to save the image as a string.
And finally the system takes the name of the shield and the variable that is stored the information whether the shield is public or not, and call the “bbddFlash.php” code.

The bbddFlash.php code receives the values and depending on whether it makes an update or a new insert in the tables affected.

In the case that the shield has not name, the system says the shield as “Sense nom”.

```php
//PHP
$nomEscut=$_POST['nomObraPHP'];
if ($nomEscut == '') {
    $nomEscut = 'Sense nom';
}
$public=$_POST['checkPublicPHP'];
$historic=$_POST['historicPHP'];
$image=$_POST['image'];
```

The system looks to the users Table to see the “edicio” field, this tells us if the user is editing or creating a new shield. The “date” function gives us the current data in “dd-MM-YYYY” format.

```php
//PHP
(instruccion = "SELECT edicio FROM c5dl3_users WHERE username = ".$username." ");
$consulta = mysql_query ($instruccion, $conexion)
or die ("Fail en la consulta");
$res=mysql_fetch_row($consulta);

//Concatenate with the user
$nom_gen = "c5dl3_obres_";
$nom_part = $nom_gen . $username;
$nom_gen_his = "c5dl3_obres_historic_";
$nom_part_his = $nom_gen_his . $username;
$data = date("d-m-Y");
```

In case that the shield is new the system inserts a new line in Obres_Historic and Obres, also new line in Obres_Publques in the case that the shield is public.

```php
//PHP
// New shield
if ($res[0] == 0){
    // Insert to historic
    $instruccion = "select id from ".$nom_part_his." order by id desc";
    $consulta = mysql_query ($instruccion, $conexion)
or die ("Fallo in select to historic");
```
$nfilas = mysql_num_rows($consulta);
$id=$nfilas+1; // id

$instruccio = "INSERT INTO ".$nom_part_his." VALUES(".$id.";'>".$historic."');
$insert = mysql_query($instruccio, $conexion)
or die ("Fail insert to historic");

$idHis = $id;

// Insert to obres
// Load Table to put id
$instruccio = "select id from ".$nom_part." order by id desc";
$consulta = mysql_query($instruccio, $conexion)
or die ("Fail in select");

// ID
$nfilas = mysql_num_rows($consulta);
$id=$nfilas+1; // id

// id, nom_obra, public, idHistoric, dataCrea, dataMod, imatge
$instruccio = "INSERT INTO ".$nom_part." VALUES(".$id.";'>".$nomEscut.";'>".$public.";'>".$idHis.";'>".$data.";'>".$data.";'>".$image.");
$insert = mysql_query($instruccio, $conexion)
or die ("Fail in insert");

if ($public == 1){
    $in restruccio="SELECT idIntern FROM c5dl3_obres_publiques");
    $consulta = mysql_query($in restruccio, $conexion)
or die ("Fail in select to obres_publiques");
    $idIntern = mysql_num_rows($consulta);
    $idIntern = $idIntern + 1;

    $instruccio = "INSERT INTO c5dl3_obres_publiques VALUES(".$idIntern.";'>".$id.";'>".$username.";'>".$public.";'>0");
    $insert = mysql_query($instruccio, $conexion)
or die ("Fail in insert to obres_publiques");
}

In the case that the systems updates the “his” field in the Obres_Historic table the code looks in Obres_Publiques if the shield exists as a public. If the result is positive, the system updates the line, otherwise and the condition to now make public the system inserts a new line.
else{
    $instruccion = "UPDATE\.$nom_part.\ SET nom_obra = '\$.nomEscut.'",
               public = '\$.public.'",
               dataMod = '\$.data.'",
               imatge = '\$.image.'",
               WHERE id = '\$.res[0].''";
    $update = mysql_query ($instruccion, $conexion)
               or die ("Fail in update");

    $instruccion = "UPDATE\.$nom_part_his.\ SET
               his = '\$.historic.'",
               WHERE id = '\$.res[0].''";
    $update = mysql_query ($instruccion, $conexion)
               or die ("Fail in update to historic");

    // Looking for if exists the line
    $instruccion = "SELECT COUNT(*) FROM c5dl3_obres_publiques
               WHERE id = '\$.res[0].''" AND username = '\$.username.'";
    $consulta = mysql_query ($instruccion, $conexion)
               or die ("Fail in select to obres_publiques");
    $numLin = mysql_fetch_row($consulta);

    if ($numLin[0] == 1){
        $instruccion = "UPDATE c5dl3_obres_publiques SET
               public = '\$.public.'",
               WHERE id = '\$.res[0].''" AND username = '\$.username.'";
        $update = mysql_query ($instruccion, $conexion)
               or die ("Fail in update to obres_publiques");
    }
    else {
        if ($public == 1){
            $instruccion = "SELECT idIntern FROM c5dl3_obres_publiques";
            $consulta = mysql_query ($instruccion, $conexion)
                         or die ("Fail in select to obres_publiques");
            $idIntern = mysql_num_rows ($consulta);
            $idIntern = $idIntern + 1;

            $instruccion = "INSERT INTO c5dl3_obres_publiques
            VALUES(''$idIntern.'', '$res[0].'', '$username.'', '$public.'', '0')";
            $insert = mysql_query ($instruccion, $conexion)
                         or die ("Fail in select to obres_publiques");
        }
    }
}
10.9.10 Load

At the starting point we had seen that one of the functions that run boot software is whether the user is editing a shield or not. If the result was positive, this loaded the following data:

- Name of the shield.
- Public checkbox.
- String with performed actions.

This string allows the system to recover the state of the shield when the user saves it.

The same way that when the user saves the shield the system uses the “-” and “+” symbols to separate rows and columns, now the system uses “split” AS3 function to recover the last status of historic vector. For each line the systems adds the information to history.

Also, the system controls:

- If the action is a division the system adds to hisParticiones vector.
- If the action is a figure the system adds to hisFiguresSprite vector.
- If the action is a crown the system adds to hisTimbres vector.

This is the codification:

- Divisions:

<table>
<thead>
<tr>
<th>IdHistoric</th>
<th>2</th>
<th>Division ID</th>
</tr>
</thead>
</table>

- Figures

<table>
<thead>
<tr>
<th>IdHistoric</th>
<th>31 – Create</th>
<th>32 – Move</th>
<th>33 - Delete</th>
<th>Figure ID</th>
<th>Sprite</th>
<th>X position</th>
<th>Y position</th>
</tr>
</thead>
</table>

- Crowns

<table>
<thead>
<tr>
<th>IdHistoric</th>
<th>4</th>
<th>Figure ID</th>
</tr>
</thead>
</table>
Thanks to this coding, the system knows what the user execution. To recover the status before the saved, the systems recover the historic vector, and divisions, figures and crowns auxiliars vectors. From this information the system executes the functions performed by the user and the result is the same that the current state of the shield.

If the load of database does not work successfully for any reason the system will inform the user with an information error. Similarly, if the result is successful, will appear a message informing the user that the data has been loaded correctly.
11 Final project status

11.1 Summary

Regarding the functionalities planned to start the project plan the author has completed most of the goals. This project was very ambitious in its scope, which made use of many different technologies and had high level of complexity.

The author has been tried to ensure the successful of the functionalities necessary for each area of the system, tool and social network. With a high degree of usability and maintain the attractive level. From there, the way forward was expanded features not essential but important to maintain the characteristics described previously.

11.1.2 Tests performed

One of the priorities of the developer of the project was the validation of each functionality. For this reason, the developer made tests in people close him following the process described in the methodology project section. The job of the developer in this case has been to observe and record the reactions of each tester after a request. The developer made the test thinking in the different situations where the user can be found in the system.

To make the validation of the most varied possible has been done three different profiles:

- Tester 1: Young people of 23 years accustomed to using new technologies.
- Tester 2: Man between 40-45 years with regular use over the computer.
- Tester 3: Woman of 60 years with a rare use of the computer.

The author has not asked to each tester all functionalities to avoid the fatigue.
• Register, login and logout:

The author requests the users to register, login and logout in the system.

**Developer test:** The developer has verified the correct functioning of each of the functions, especially the flow of menus and views. Also, checks the creation of tables for each user in the database.

**Tester 1:** The tester 1 has registered in the system without problems, he does not like to insert again the email. Once registered login is done quickly. He did logout directly closing the window.

**Result:** The validation of the functionality is ok.

• Create new shield.

The author requests the users to create a new shield.

**Developer test:** The creation button responds correctly. It may surprise the fact that disappears the Joomla template but for reasons of space is needed. The developer checks the “edicio” field status in the database for the different possibilities.

**Tester 1:** The tester 1 navigates easy to personal shields and press quickly the button to create a new shield. Once has appear the new window he waits some seconds probably for the amount of information received in a short time.

**Tester 2:** The tester 2 reviews the different section of the system before the creation of the shield. Finally, he goes to personal shields and press the button. Once inside quickly moved through the menus of the tool.

**Tester 3:** The tester 3 reacted in the same way that the tester 1. He entered without problems and has been stopped for a moment.

**Result:** The validation of the functionality is ok, but the author considers that maybe in a future there should be an informative message when the user opens the tool.
• Edit existing shield.

The author requests the users to edit an existing shield.

Developer test: The link to edit the shield answers correctly. The developer checks the pass of data from historic and the modification of "edicio" field of the Users table. If the volume of data is large perhaps the system takes a while to load.

Tester 1: The tester 1 enters into the system and navigates quickly to personal shields, he finds the link and starts editing without problems.

Tester 3: The tester 3 enters into the system and navigates to public shields before goes to personal shields. He changed in short time, we understand that the user has noticed quickly that he did a mistake. He finds the link and edit without problems.

Result: The validation of the functionality is ok, but the author considers that maybe in a future there should be a link to edit from public shields.

• Using the social network

The author requests the users to interact with other users.

Developer test: The buttons “Like” and "Comment" answer correctly. The developer looks that the values are correctly. The developer makes random test inserting and deleting comments to ensure the good operation of these functionalities. Ensures that changes are correctly in the database.

Tester 1: The tester 1 enters into the system and goes quickly to public shields section. There, he sees the first shield and press “Like”, later he sees that him username appears in the list of users that made “Like” on the shield. Then click the "Comment" button and write a short text. He says that this function "is quite good".

Tester 2: The tester 2 enters into the system and for some reason that does not comment he goes to personal shields. From there he press “Like” in one shield. Then he goes to public shields and press “Comment” on the first shield. He writes a couple of lines, he deletes one of them later.

Tester 3: The tester 3 enters into the system and goes quickly to public shields. From there takes a while to decide what it does. Press a couple of “Like” in a few shields and decides leave the window. The observer requests him if he can come back and write a message to
another user and the tester answer how can do that. The observer explains. The tester re-enters and writes a message to the first shield.

**Result:** The validation of the functionality is ok.

- **Change current avatar.**

The author requests the users to change the current avatar that appears when the user makes “Like” or “Comment”.

**Developer test:** The load of a new image and upgrade works correctly. Additional information is loaded correctly.

**Tester 1:** The user tries to upload the first image that he finds in his computer. This image does not pass the necessary conditions and appears error message. Then, he takes other image and now this is uploaded correctly.

**Tester 3:** The user first accesses to personal shields, he makes an scroll and he said that this functionality does not exist in this section. Then read the different sections of the horizontal bar and access in Profile. Take the first image in his computer and realizes the upgrade.

**Result:** The validation of the functionality is ok, but the author thinks that maybe in a future this functionality can be executed from other sections of the system.

- **Using the Catalan heraldry guide**

The author asks the users what enamels are allowed in the Catalan heraldry.

**Developer test:** The interaction with the Catalan heraldry guide is correct, the information appears correctly depending when the user opens or closes the different sections.

**Tester 2:** The user goes to “Search” Joomla module, write the words “esmalts” and select the first page in the search found. Then, open the enamels section.

**Tester 3:** The tester 3 search on the horizontal bar and press the “Heràldica Catalana” menu, later reads the different sections and click on enamels.

**Result:** The validation of the functionality is ok.
• Using the tool

The author requests the users to create a new shield.

Developer test: Throughout the development of the tool the developer has been checking every functionality and button operation. With special attention to the most delicate part of the software: the management of history and the data exchange with the server.

Tester 1: The user opens the tool. The first thing that makes is write the name of the shield and mark it as public. Then he navigates through the menus in order to see all options available. Click a few figures and drags to the shield. Click a division that does not like and click another. Draw a few lines and paint with different colours the areas of the shield. Before save, he click in the one crown.

Note observer: The feeling is that the tool is intuitive to be able to make use of easy and comfortable. The user has been satisfied with the creation.

Tester 2: The user opens the tool. He navigates between the divisions and clicks four divisions. He passes to figures and creates a lot, many of them are deposited in the “trash”. Finally we left the shield with three figures. He looks the crown section and selects one. He tries to drag it to the “trash”. The system does not answer. He tries to click other crown and thinks that now is good. Finally paints the different areas of the shield before save without name.

Note observer: The feeling is that the tool is easy to use, but some functionalities like the removing the crown can confuse the user because it is very similar to the figures. Maybe in the future should be studied in a way that this section will be clearer.

Tester 3: The user opens the tool. The user remains while watch the different areas of the screen and begins to draw a couple of lines to the shield. One of these is short. Select a color and paint one area. This area was not completely divided and the new image was not that the user expected. The user click the “undo” button to be completely clean the shield and repeats the process. Now, the result is how the user expected and he starts to see the menus. Selects a division and it appears to the shield. The user shows a little surprised that the lines are keeping in the shield and decides to return to “undo” all actions. Selects a division, a few of figures that locates in the shield. In the section of crowns he selects one of them. This crown does not like the user and he clicks “undo” and select another. Finally he paints the areas of the shield, puts a name and saves the work.
Note observer: This was the more important test. The functionalities worked correctly and the tool gives the feeling that is effective. The author concludes that in a future should try to help the users to understand the use of the tool in order to the user can understand and uses all functionalities.

**Result:** The validation of current functionalities of the tool is ok.

### 11.2 Discarded functionalities

Some functionalities planned at the beginning of the project have been discarded for reasons of time, complexity and priorities.

The first functionality was the development to add figures in the heraldic tool software. The reason was the complexity of mounting this function because the figures are attached to different divisions. The software does not identify the divisions, we only have a bitmap with black pixels that they separate the areas to be painted.

The second functionality was the contact section. The author had planned a section with a form to contact with the administrator of the system. This form will send and email. For priority reason, the developer was decided that this function would be developed in the case that exist more time.

The third functionality was the possibility of edit one shield from the public shields section. The developer gave low priority to this functionality because this function is already exist in the personal shields section.

### 11.3 Future expansions

We have a number of future expansions that could be implemented in a new version of the system more the three discarded functions.

**Tool level:**

- Be able to select other types of shields.
- Possibility for the user to introduce new figures and crowns.
- Possibility for the user to change the size of the figures.
- Modify the tool to follow the rules of enamels.
- Be able to download the image in other formats than PNG.
Web level:

- Be able to make friends with other users, it would see their public shields directly.
- Be able to send messages between users directly without having to do that in a shield.
- Possibility that the user has a header with the interactions received from other users.
- Be able to download the image directly without enter the tool.
- Be able to recover old versions of the shields.
Conclusions and final assessment

The main personal goal for the realization of this project was learn new technologies. New technologies that I do not see in the degree. The experience has been very beneficial because the project allowed me to know a lot of technologies some of which can be very profitable in the future.

In the current job market and in the near future an important role is the web developer, actually has increase the cloud tools where the business want a minimum of knowledge about this technology.

Flash is a technology little required currently in the workplace but even this will add a new point in my experience in the world of computing.

The interest of the author for the shields comes from I was a kid. With the family when we see a flag in some town commented what could be the meaning of each of the parts of the shield. Continue to evolve with this type of information of the Catalan Heraldry was also quite interesting.

It should be mentioned that the development of this system has not been easy, the fact that the developer started without knowledge about all technologies used has meant that the developer has need to learning process in a significant period of time. The author has taken decisions and prioritize the development of the most important functionalities for the system and discard others interessant functions. This experience will also be very useful for the author in future planning and implementation projects.

Personally I am satisfied with the results, I have been already fulfilled almost all the goals planned at the beginning, even ended doing some extra which was not foreseen at the specification but I think that would be interesting to implement as I was developing the system.

I think that the result is a interesting system for lovers of this science, with a many options to improvement but that the system has nothing to envy to other types of systems other topics.
12.1 Technical competences evaluated

Technical competences initially defined in the project:

- **CSI2.4:** To demonstrate knowledge and capacity to apply systems based on Internet (e-commerce, e-learning, etc.). [A little bit]
- **CSI4.1:** Participate actively in the specification of information systems and communication. [Enough]
- **CSI3.3:** Evaluate technology offers for the development of information systems and management. [Enough]
- **CSI4.2:** Participate actively in the design, implementation and maintenance of information systems and communication. [In depth]

It details what part of the project has been worked with each of them:

- **CSI2.4:** The author has installed and configured a CMS such as Joomla. This CMS is very broad and offers many possibilities, so once installed and configured the author has been reviewing to achieve improve in the system.
- **CSI4.1:** In the planned phase of the project the author had to think how deployed information systems. Both the web area and tool area, the synchronization between the two parties and the server.
- **CSI3.3:** In the planned and development phase the author has made a continuous task to investigate what were the best options to develop information systems and communication.
- **CSI4.2:** This has been the basis of project, after a good specification in the planned phase the author implemented information systems and communication. Thanks to the good development of this phase the author have been completed with an efficient system.
13 Bibliography

[1] Wikipedia, the Heraldry – The free encyclopedia
http://ca.wikipedia.org/wiki/Her%C3%A0ldica – March 3, 2014

[2] Heraldic portal web in Spain
http://www.heraldaria.com/dise%C3%B1oH.php#3 – March 3, 2014

http://ca.wikipedia.org/wiki/Esmalt_%28her%C3%A0ldica%29 – March 3, 2014


[5] Blazon, software to create coat of arms

[6] Puncher, software to create coat of arms

[7] Inkwell Ideas, website of software's creation


[9] ZOHO – Set of web applications
http://www.zoho.com – September 1, 2014

www.wampserver.com – September 1, 2014


[12] Installation Wampserver guide
http://es.slideshare.net/verodmar/wampserver-gua-de-instalacin – September 10, 2014

[13] Installation Joomla and Wampserver guide


[16] Heraldry Redraws - Heraldic images


[18] JSON, introduction to JSON
www.json.org – Setembre 10, 2014