AUTOMATION SYSTEM INFORMATION MANAGEMENT AND DRUG HISTORY IN THE HEALTH SUBCENTRES SAN ANTONIO DE IBARRA

Developed by: Egdo. Jorge Aníbal Ipiales Ipiales
Telf: *062 933310;* **Cel:** *097292992*
Email: *jipiales2006@hotmail.com; j_ipiales2005@yahoo.com*
UTN-2011

SUMMARY

For the realization of this project was to model first the entity-relationship diagram, a free tool helped MysqlWorkBench, which facilitated this design, because it works visually, once this model we had to get tools programming, so it was decided to Netbeans (version 6.9.1) and symfony 1.4, a framework which helps greatly with PHP programming, helped with the same Doctrine, is now the default ORM for symfony from version 1.4, accelerate web design projects.

The knowledge of these tools, more than basic knowledge of JavaScript, CSS, ajax and html helped the completion of this project to facilitate the work of the Department of Statistics in the Health Sub-center San Antonio de Ibarra.

INTRODUCTION

As the automation requirements of information in public or private entities increase, programming technology should go hand in hand with these requirements, because at present web programming has been increasing along with the popularity of the Internet, so the need to increase the knowledge of system design tools for the web.

One of the drawbacks with the designers, refers to the acquisition tools to achieve this goal, so a great alternative you have is to use free tools, which we can get off from the web, as there is a variety of these, the which we can use it without any monetary payment and performance does not differ greatly with paid tools.

The creation of tools that is now quite widespread as are the frameworks has been to expedite the process of software development for the web, this is the case Symfony, as there is good documentation on the network and which discussions blogs in turn help to a good understanding of this tool.

In this paper we will indicate the processes necessary to develop web applications using netbeans, symfony and helped in this case to apply for project development for the web in the health sub-center "San Antonio" with a local web server WampServer 2i.

MATERIALS AND METHODS

In this paper, first we designed the entity relationship diagram helped MysqlWorkBench (version 1.5.18), the same which is free and has a visual environment.

Figure 1: Logo *MysqlWorkBench.*

After designing the optimal scheme of the database for this application, the entity relationship diagram was as follows:

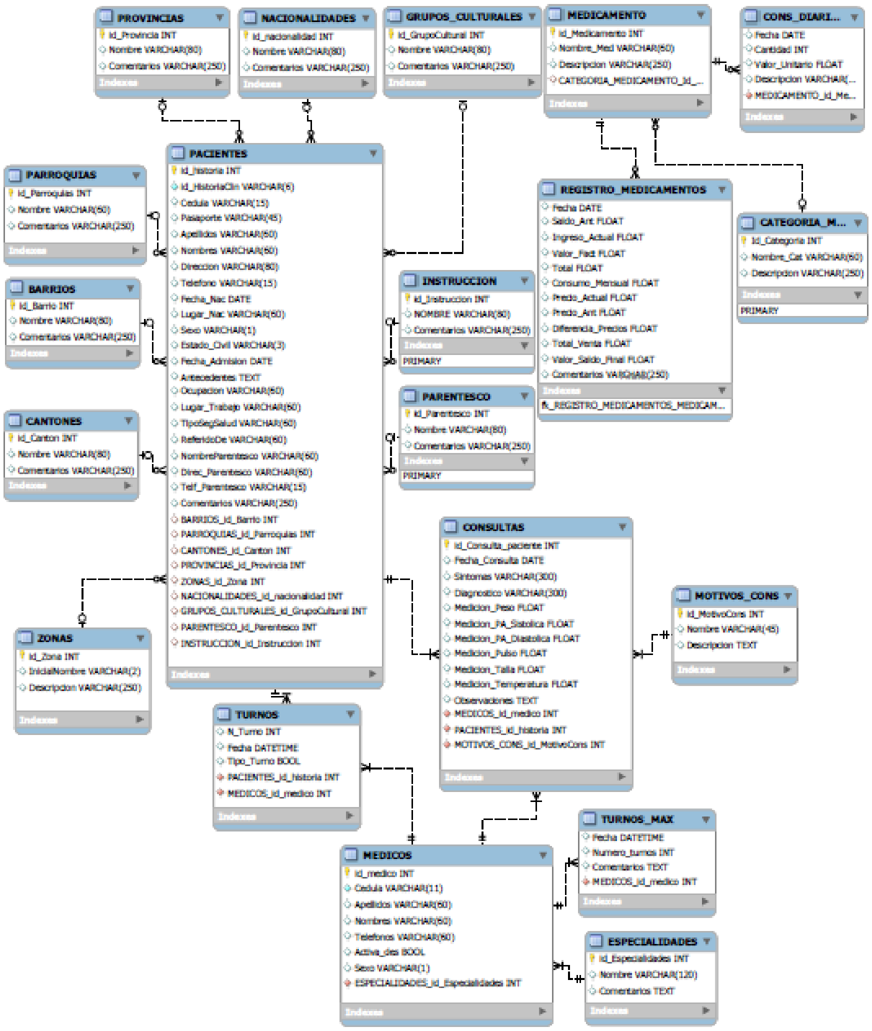


Figure 2: *Entity-Relationship Diagram of the database for the health subcenter system of San Antonio.*

The tables were designed for the system of health subcenter San Antonio that are shown in Figure 2, were designed in the tool you quoted above.

This diagram was the basis to bring this scheme to the physical database engine to Mysql database, it is installed together with the applicationWamp2i.

To run the application, we first run our web server, which in our case is the WampServer 2i,so we install it installed, otherwise it will not work, so to install it first must be downloaded from the Internet(http:///www.wampserver.com/en/), if you have installed should see an icon like this:

Figure 3: *WampServer icon representing 2i*

SYMFONY SETTINGS IN NETBEANS.

As mentioned in the introduction, our development tools will be Netbeans Symfony helped to facilitate programming with this framework and make it visually.

As mentioned in the introduction, our development tools will be Netbeans Symfony helped to facilitate programming with this framework and make it visually.

Symfony is based on MVC (Model View Controller), ie, the logical structure of the projects created are divided into:

- **Applications:** that is our project,
- **Classes:** Classes types of functionality that includes our application (user management, news management, management of comments, etc..) And
- **Modules:** which is which details how to make each feature of the class to which they are assigned (high, delete and edit users, etc..).

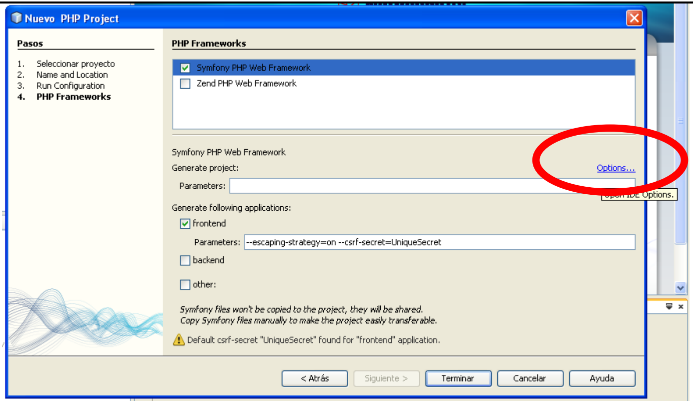
Now we turn to configure Netbeans to work with symfony, as follows:

First we have a "LAMP" (Linux-Apache-MySQL-PHP / Python / PERL) installed on your computer, in our case we use WAMP SERVER 2.0.

Then download the latest stable version of symfony from its website, for the start of our project was the latest stable Symfony 1.4.

After downloading the package usually comes packaged, unzip in the directory on the web, as we work locally, create a folder on drive C: /, as follows: "C: \ SubcentroSalud \ system \" , then the full address where the decompressed will be something like: C: \ SubcentroSalud \ system \ lib \ vendor \ symfony

In our next step we opened Netbeans, and then create a new PHP project, the wizard will guide us step by step, where we only give our project name and location of our project files, but in the step where we choose the framework (PHP frameworks) obviously choose Symphony. When we select various configuration options appear. We'll have to click on Options on the right of the window, as shown in **Figure 4**:



Nuevo PHP Project

Pasos

1. Seleccionar proyecto
2. Name and Location
3. Run Configuration
4. **PHP Frameworks**

PHP Frameworks

☑ Symfony PHP Web Framework
☐ Zend PHP Web Framework

Symfony PHP Web Framework

Generate project:

Parameters:

Generate following applications:

☑ frontend

Parameters: --escaping-strategy=on --csrf-secret=UniqueSecret

☐ backend

☐ other:

Symfony files won't be copied to the project, they will be shared.
Copy Symfony files manually to make the project easily transferable.

⚠ Default csrf-secret "UniqueSecret" found for "frontend" application.

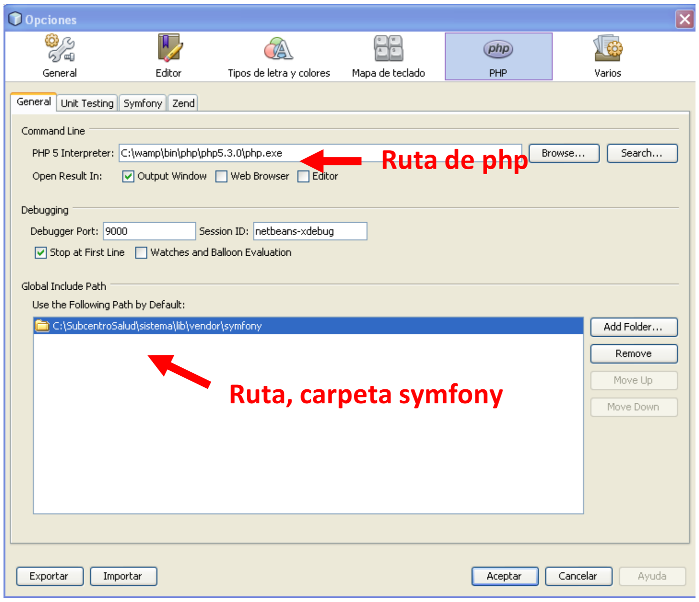
Figure 4. *Location of Options to configure symfony in Netbeans*

After clicking on Options ..., we will open the PHP configuration panel in this window click on the tab where PHP will indicate the path where the PHP executable we have installed LAMP, in our case in the following path:

C:\ wamp\bin\php\php5.3.0\php.exe

In addition, this window will indicate the Symfony installation path by clicking the Add Folder, which will have decompressed folder in the directory of our project

C:\SubcentroSalud\system\lib\vendor\symfony



Opciones

Command Line

PHP 5 Interpreter: C:\wamp\bin\php\php5.3.0\php.exe

Open Result In: ☑ Output Window ☐ Web Browser ☐ Editor

Debugging

Debugger Port: 9000 Session ID: netbeans-xdebug

☑ Stop at First Line ☐ Watches and Balloon Evaluation

Global Include Path

Use the Following Path by Default:

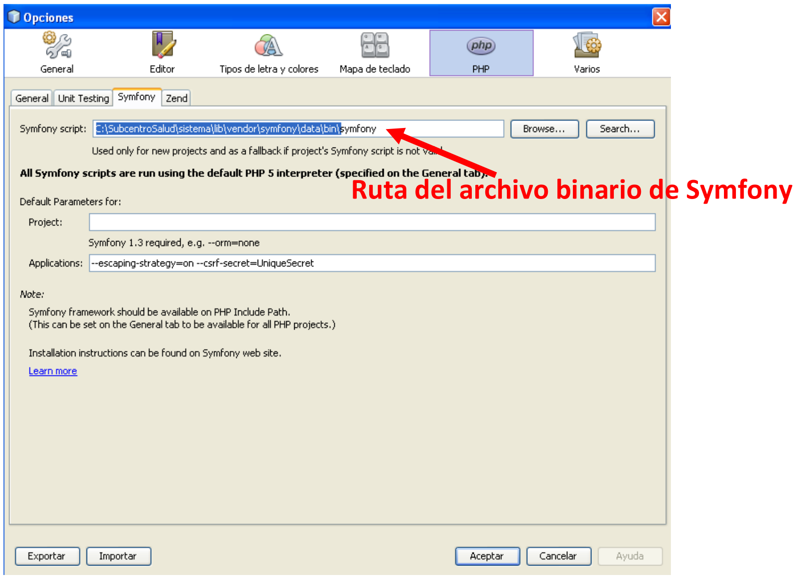
C:\SubcentroSalud\sistema\lib\vendor\symfony

Figure 5. *Indicate Php path and folder in the project Symfony*

Now, Symfony tab of this screen will indicate the path to the binary framework, which in our case is found at the following address:

C:\SubcentroSalud\system\lib\vendor\symfony\data\bin\symfony

This is indicated in the figure below:



Symfony script: C:\SubcentroSalud\sistema\lib\vendor\symfony\data\bin\symfony

Used only for new projects and as a fallback if project's Symfony script is not valid.

All Symfony scripts are run using the default PHP 5 interpreter (specified on the General tab).

Ruta del archivo binario de Symfony

Default Parameters for:

Project:

Symfony 1.3 required, e.g. --orm=none

Applications: --escaping-strategy=on --csrf-secret=UniqueSecret

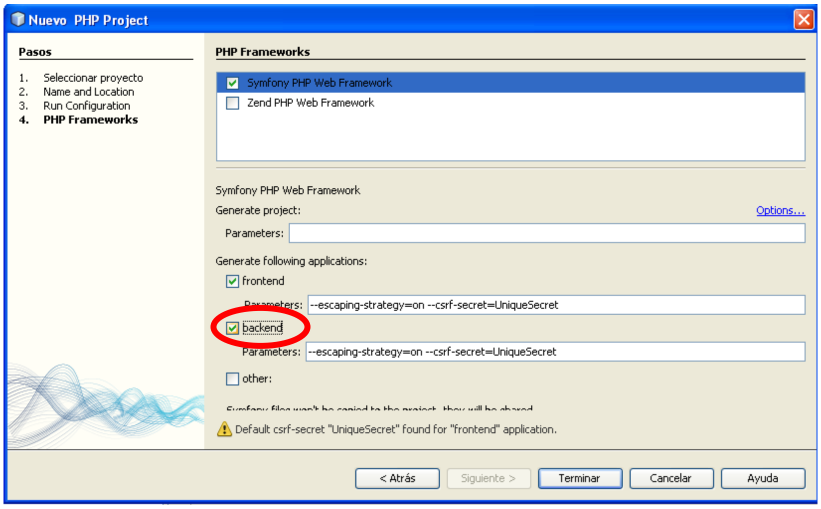
Note:

Symfony framework should be available on PHP Include Path.
(This can be set on the General tab to be available for all PHP projects.)

Installation instructions can be found on Symfony web site.

Figure 6. *Location Symfony binary file*

Finally, we click OK, but before finishing, select the checkbox Backend, where it should be a seen as we indicated in the graph below, this in order that our project manage the database after that if you click on Finish.



Pasos

1. Seleccionar proyecto
2. Name and Location
3. Run Configuration
4. **PHP Frameworks**

PHP Frameworks

☑ Symfony PHP Web Framework
☐ Zend PHP Web Framework

Symfony PHP Web Framework

Generate project:

Parameters:

Generate following applications:

☑ frontend

Parameters: --escaping-strategy=on --csrf-secret=UniqueSecret

☑ backend

Parameters: --escaping-strategy=on --csrf-secret=UniqueSecret

☐ other:

Symfony files won't be copied to the project, they will be shared

⚠ Default csrf-secret "UniqueSecret" found for "frontend" application.

Figure 7. Location backend to run the DB.

If there is any problem during all the steps that the wizard guided us, we will have configured and installed our symfony project in NetBeans.

CONFIGURING APACHE WEB SERVER

As we solve this, we must first set the host in this case our local host will be virtual, so that it runs under a different name in another host and port (in this case port 80).

On the Internet we found several examples of how to create a virtual host to run the project locally, here's an example:

```
NameVirtualHost *:80
<VirtualHost *:80>

ServerName miApp
DocumentRoot "C:\Users\Juanjo\Documents\NetBeansProjects\miApp\ web"
DirectoryIndex index.php
<Directory "C:\Users\Juanjo\Documents\NetBeansProjects\miApp\ web">
AllowOverride All
Allow from All
</Directory>

#solo para la barra de depuración:

Alias /sf "C:\symfony-1.4.11\data\web\sf"
<Directory "C:\symfony-1.4.11\data\web\sf">
AllowOverride All
Allow from All
</Directory>
</VirtualHost>
```

What we do is we take any of these codes and edit it to attach to your project, take into consideration that we will use Symfony as a framework for programming with PHP, which is why this configuration refers to this folder where we will symfony, so we would be as follows:

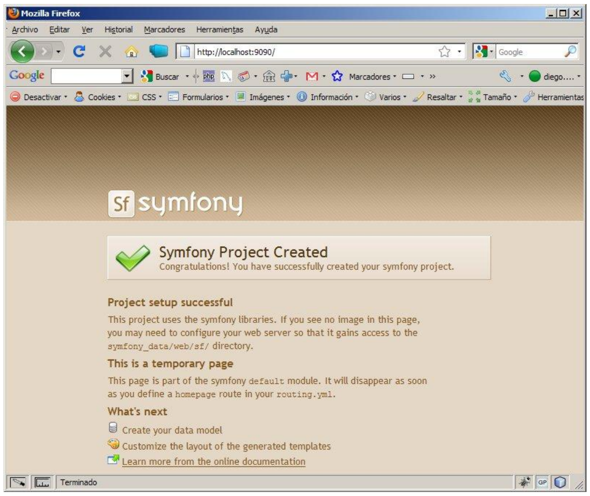
```
# Esta es la configuración de Sistema para Subcentro de Salud
#Listen 127.0.0.1:80

<VirtualHost 127.0.0.1:80>
  ServerName sistema
  DocumentRoot "C:\SubcentroSalud\sistema\web"
  DirectoryIndex index.php
  Alias /sf "C:\SubcentroSalud\sistema\lib\vendor\symfony\data\web\sf"
  <Directory "C:\SubcentroSalud\sistema\lib\vendor\symfony\data\web\sf">
    AllowOverride All
    Allow from All
  </Directory>
  <Directory "C:\SubcentroSalud\sistema\web">
    AllowOverride All
    Allow from All
  </Directory>
</VirtualHost>
```

As symfony already established routes in the project folder, we have to follow these steps:

1. Copy the contents (Ctrl + C) the code above (the VirtualHost)

2. Open the httpd.conf file located in the following path: C: \ wamp \ bin \ apache \ Apache2.2.11 \ conf,
3. Paste it (Ctrl + V), usually at the end of the contents of this file,
4. Save,
5. And finally restart Apache services.

If all went well, and if you have configured Apache correctly, open the browser and note the following address: http://localhost:80, but we can also do it from Netbeans, typing the F6 key to execute the project. And so we should get a screen similar to the following:


Mozilla Firefox
Archivo Editar Ver Historial Marcadores Herramientas Ayuda
http://localhost:9090/
Google
Google Buscar Marcadores diego....
Desactivar Cookies CSS Formularios Imágenes Información Varios Resaltar Tamaño Herramientas
Sf symfony
Terminado

Symfony Project Created
Congratulations! You have successfully created your symfony project.

Project setup successful
This project uses the symfony libraries. If you see no image in this page, you may need to configure your web server so that it gains access to the `symfony_data/web/sf/` directory.

This is a temporary page
This page is part of the symfony `default` module. It will disappear as soon as you define a `homepage` route in your `routing.yml`.

What's next
Create your data model
Customize the layout of the generated templates
Learn more from the online documentation

Figure 8. *Symfony screen at the end of basic configurations*

All this is to run the application as a virtual server, so you must have added the virtual domain hosts file of your operating system to go to this address C: \ Windows \ System32 \ drivers \ etc \ hosts and added:

```
127.0.0.1        localhost
127.0.0.1        sistema
```

So to enter our project http://sistema single digits in your browser (Mozilla Firefox 4 or higher preferred).

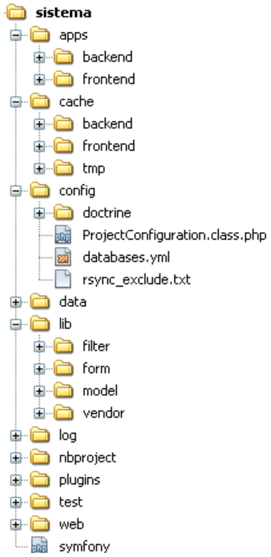
SYSTEM

After making the necessary settings for your application to work, we analyze the structure of our project, it was done with the help of Netbeans 6.9.1., The latest stable version to date of initiation of our project.

To begin to develop projects with symfony, you must have a knowledge of at least basic HTML, CSS, programming this way learn to be a little more orderly in relation to what you have to see the place where they will go code programming.

Well, sometimes you use only HTML, in other cases we should use PHP alone or a mixture of them also refer to files CSS for styling our pages.

It seems important to understand how the code structure Symfony. Any project created by Symfony is a series of folders and files containing the framework code base. The tree of folders in a project is:



```
sistema
  apps
    backend
    frontend
  cache
    backend
    frontend
    tmp
  config
    doctrine
    ProjectConfiguration.class.php
    databases.yml
    rsync_exclude.txt
  data
  lib
    filter
    form
    model
    vendor
  log
  nbproject
  plugins
  test
  web
  symfony
```

Figure 9. *File structure of symfony project*

In addition, a project created with symfony are divided into:

- **Applications:** that can have the same modules can usually be two backend and frontend applications.
- **Modules:** These in turn tend to coincide with the tables in the database created as the project purpose.

Then we have the database and we are clear as is the structure of a symfony project, we can insert our code where deemed necessary.

For this we need to create a folder where we will create all the folders and copy the folder Synfony downloaded from their official website, in our case we will help with a tool "Netbeans"

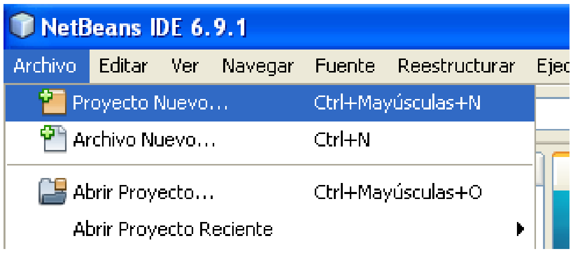


Figure 10. *Create a new project in symfony with Netbeans.*

This way of creating the project replaces the following code:

symfony generate: project system

Whichever way we create our project, we have created the entire tree of directories and files needed to launch our website with this framework.

Then we create our modules, but if we use Netbeans as it is in our case to create our project and also creates two default applications.

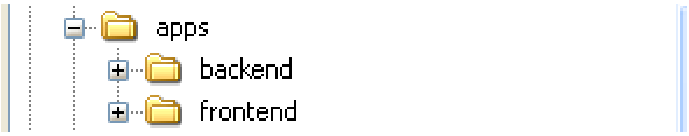


Figure 11. Folder structure of the application.

However if you do not use Netbeans, you must create them manually, using a code like the following:

- symfony generate: app - escaping-strategy = on - csrf-secret = UniqueSecret frontend
- symfony generate: app - escaping-strategy = on - csrf-secret = UniqueSecret backend

When calling the generate: app task, we have also been two security related options:

- escaping-strategy: Enables output escaping to prevent XSS attacks
- csrf-secret: Enables session tokens in forms to prevent CSRF attacks

The next step is to connect our application to the database, for this, Symfony has a databases.yml configuration file in which we will indicate where it is located, to which database and which user and password is to be made.

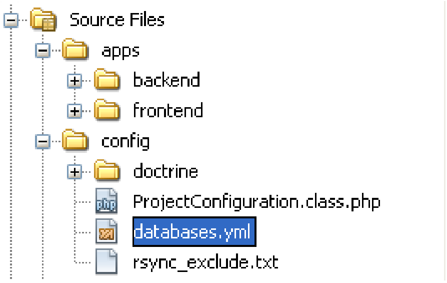


***Figure 12.** Location of the configuration file of the database for the project.*

This file usually contains the following code similar to the following:

```
all:
  doctrine:
    class: sfDoctrineDatabase
    param:
      dsn: 'mysql:host=localhost;dbname=bdblog'
      username: root
      password: null
```

With this, we have configured our project database. Now we tell Doctrine to generate schema.yml from the database:

symfony doctrine:build-schema

Then we create the model classes to be generated to manage the database through objects, as follows:

symfony doctrine:build-model

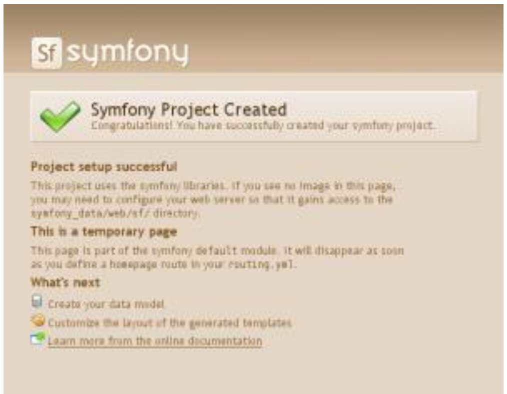
In a similar way to create doctrine, filters, forms, etc. To not perform all these actions separately, using a single command, as follows:

Symfony doctrine:build –all

Only with the above command must be careful if you already have the database created above, as this is a way to reset the database and leave it blank, or rebuilds.

As we noted above, we use Netbeans, so that these lines of code can be run visually.

If all went well, we can go with your browser to see the screen Symfony where we said that the project has been created for that visit http://sistema/, as follows:



Sf symfony

Symfony Project Created

Congratulations! You have successfully created your symfony project.

Project setup successful

This project uses the symfony libraries. If you see no image in this page, you may need to configure your web server so that it gains access to the symfony_data/web/sf/ directory.

This is a temporary page

This page is part of the symfony default module. It will disappear as soon as you define a homepage route in your routing.yml.

What's next

- Create your data model
- Customize the layout of the generated templates
- Learn more from the online documentation

Figure 13. *Initial screen of a symfony project created.*

From now on no more than create modules in our applications such as Backend and Frontend, in addition to insert code to manipulate and control our vision for the system.

NOTE: The way to run a symfony command is simple, just right click on the name of the application in which we are working and a window will appear as shown in **Figure 14**:

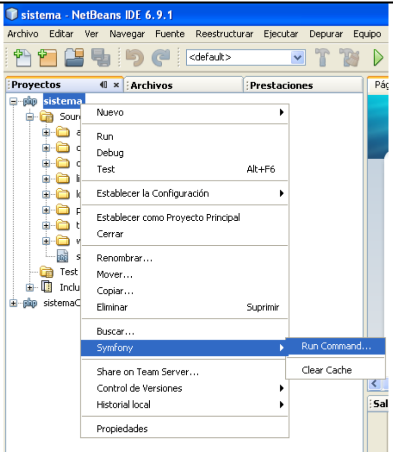


***Figure 14:** Execute commands in Netbeans Symfony.*

As we left click on Run Command ... option from the dropdown menu as shown in **Figure 14**. File structure of symfony project

The structure of our project so we can see graphically in **Figure 15**.

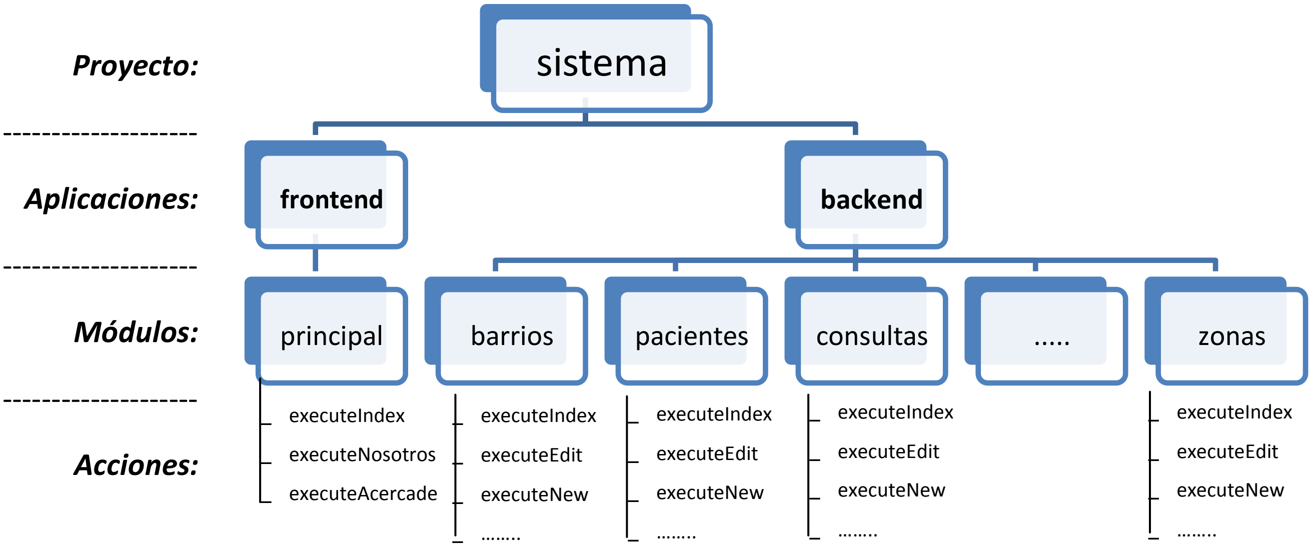


Figure 15: Structure of a symfony project.

RESULTS AND DISCUSSION

Both for the design of the database, web server and development tools free system was used in its entirety, so we did not have to pay any license, this is a significant advantage that greatly helped the development.

The system is aimed at nurses and doctors who provide professional services in the Health Sub-center San Antonio de Ibarra, they will have access to the database through an intranet in the building of this house of health.

Regarding the database, we record this information to:

- Patient Medical Records.
- Personal Data Doctors serving in the nursing home.
- Data needed drugs categorized and recording inputs and outputs of the same.
- Basic data shifts that occur daily to the patients.

All this, plus the ability to restore or create backup copies of data in a respective time.

WITH RESPECT TO THE TABLES OF THE DATABASE

The tables in the database that was designed for the system is provided solely for the sub-centers and urgent needs of the nursing home.

For this, define the data model, so you need an ORM (Doctrine or Propel) to interact with the database and create the first module of the application.

The advantage of using Symfony is that this tool is responsible for most of the work, making it easy to create a fully functional web module without writing much code PHP.

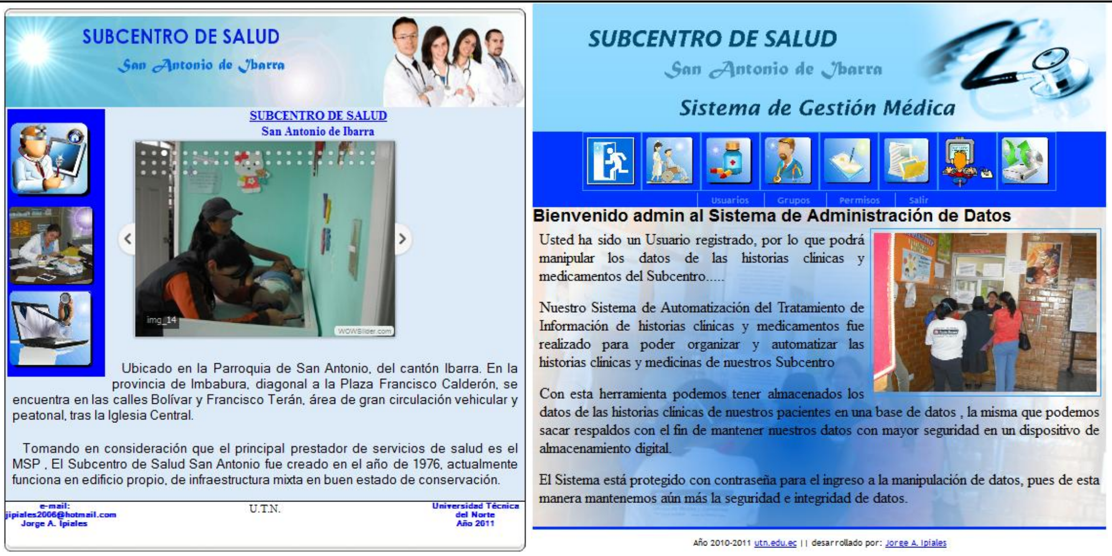
All data Medical Records, Doctors, Drugs, Shifts, Consultations, put it in a relational database. Now, as symfony is an object-oriented framework, this tool gives us the opportunity to work with objects. For example, we prefer to use objects to having to write SQL statements to obtain records from the database, because this way we will be more attached to the object-oriented programming.

As for working with objects in a relational database, you must perform a "mapping" or conversion between the information in the database and PHP objects to this we can perform them with some tools called ORM, Symfony as default includes two of the most used: Propel and Doctrine, for symfony version (Version 1.4) was used, it comes standard Doctrine.

The ORM proceeds from the description of each table and the relationships between tables to create the same PHP classes that are needed to work with objects.

PRESENTATION OF THE SYSTEM:

Gaps are some options that could be corrected, as regards the presentation of pages, which can be improved by increasing flash animations to make it look even more beautiful, but basic knowledge is used style sheets and javascript to achieve a more appropriate presentation to reach our goal, we have the result in **Figure 16**:



Ubicado en la Parroquia de San Antonio, del cantón Ibarra. En la provincia de Imbabura, diagonal a la Plaza Francisco Calderón, se encuentra en las calles Bolívar y Francisco Terán, área de gran circulación vehicular y peatonal, tras la Iglesia Central.

Tomando en consideración que el principal prestador de servicios de salud es el MSP , El Subcentro de Salud San Antonio fue creado en el año de 1976, actualmente funciona en edificio propio, de infraestructura mixta en buen estado de conservación.

Bienvenido admin al Sistema de Administración de Datos

Usted ha sido un Usuario registrado, por lo que podrá manipular los datos de las historias clínicas y medicamentos del Subcentro.....

Nuestro Sistema de Automatización del Tratamiento de Información de historias clínicas y medicamentos fue realizado para poder organizar y automatizar las historias clínicas y medicinas de nuestros Subcentro

Con esta herramienta podemos tener almacenados los datos de las historias clínicas de nuestros pacientes en una base de datos , la misma que podemos sacar respaldos con el fin de mantener nuestros datos con mayor seguridad en un dispositivo de almacenamiento digital.

El Sistema está protegido con contraseña para el ingreso a la manipulación de datos, pues de esta manera mantenemos aún más la seguridad e integridad de datos.

Figure 16: *Presentation Screens health sub-system of San Antonio.*

CONCLUSIONS

- The Health Sub-center San Antonio is the main health is the parish of San Antonio de Ibarra, is why the number of stories have grown, as people come here of their surrounding neighborhoods.
- The population growth also makes health services require urgent update in terms of tools for information processing to avoid creating crowds.
- Free tools exist today that offer great advantages for the designer of web sites such as Symfony.
- A framework automates common tasks so that application development is greatly simplified.
- A framework changes the program criteria as it provides a structure to the source code, thus creating code more readable and easier to maintain.

LITERATURE CITED

To find more information about the framework used, and specifically Doctrine admin generator, can be found in Chapter 12 of the book "Practical symfony" in Spanish (Day 12) in the web address http://www.symfony-project .org/jobeet/1_2/Doctrine/es /.

Other sources cited:

- 🕮 POTENCIER, Fabien & Zaninotto, François, Symfony 1.0 DEFINITIVE GUIDE, Creative Commons, Year 2008.
- 🕮 POTENCIER, Fabien & Zaninotto, François, symfony 1.2, The Definitive Guide, Creative Commons, Year 2008.
- 🕮 Perez, Javier Eguiluz, Introduction to AJAX, Creative Commons, Year 2008.