Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Computer Science

Diploma Thesis
Visualizer of development data in ICT

Marc Nicolàs i Aragó

Supervisor: Ing. Martin Molhanec CSc.

Engineering Senior Computer Science
January 2008
Acknowledgements

First, I would like to thank my parents for supporting and have fought for all my life and my giving me the opportunity to receive training and studies that I wanted.
Secondly, I would like to thank especially the support he has given me the director of the project, Martin Molhanec, and also my professor from Barcelona, Leandro Navarro.
I would also like to thank Aida, for his unconditional support even in the worst moments. And also to thank Vicent, my roommate in Prague, who supported me every day.
Finally, I would like to thank the friends who have accompanied me throughout the degree.
Declaration:

I declare that I have made this diploma thesis only myself, and I have used sources which are on the reference list only.
I do not have any strong reason for usage of this work in accordance 60 of law num. 121/2000 Sb.

Prague, 18.1.2008 ...........................................
Abstract

The work represents the implementation of a system capable of displaying data development of the information and communication technologies. The main goal of the work is to develop a software from pieces of free software that allows to gather and to represent manifold indicators of development of the information’s society to be able to visualize it in a cute charts and graphs.

The work demonstrate an easy way to create charts and graphs from user’s data and use to embed it in own web sites.
Index

1. Introduction .......................................................................................... 13
2. Methodology ......................................................................................... 15
3. Initial studies ........................................................................................ 17
   Requirements ....................................................................................... 17
   Gantt diagram ..................................................................................... 18
   Risk analysis ....................................................................................... 18
   Technologies ....................................................................................... 19
   Apache ................................................................................................. 19
   PHP ........................................................................................................ 20
   MySQL ................................................................................................. 20
4. Libraries ............................................................................................... 20
   Graph tools .......................................................................................... 20
   Flash-based solutions .......................................................................... 21
   PHP/ASP Libraries ............................................................................... 25
   JavaScript-based solutions ................................................................... 26
   CSS-based solutions ............................................................................ 28
   Best solutions ...................................................................................... 28
   Script.aculo.us .................................................................................... 31
   Validation library ................................................................................ 31
5. Use Case Diagram ................................................................................ 32
   Actors ................................................................................................... 32
   Uses Cases .......................................................................................... 32
   Description of architecture .................................................................. 35
   Conceptual database model ................................................................. 35
4. Analysis & Design ................................................................................ 37
6. Use Case Specification .......................................................................... 37
   Register ............................................................................................... 37
   Consult development charts by index .................................................. 38
   Consult development charts by country .............................................. 39
   Consult development world graph ....................................................... 40
   Login .................................................................................................... 41
   Logout ................................................................................................... 42
   Change profile ..................................................................................... 43
   Change password ................................................................................ 43
   Delete user’s account .......................................................................... 44
   Upload/import data ............................................................................... 45
   List user’s data .................................................................................... 46
   Show user’s data ................................................................................ 47
   Delete user’s data ............................................................................... 48
   Create chart ......................................................................................... 48
   List user’s chart .................................................................................. 49
   Show user’s chart ............................................................................... 49
   Delete user’s chart ............................................................................. 50
   List users ............................................................................................. 51
   List admins .......................................................................................... 52
   Change user’s role ............................................................................... 52
   Delete user ........................................................................................... 53
List all user's data .......................................................... 53
List all user's charts ....................................................... 54
Physical database .......................................................... 55
users ........................................................................... 56
users_data ..................................................................... 56
users_charts .................................................................. 57
countries_world_map ...................................................... 57
GISW_countries ........................................................... 57
GISW_DOI ..................................................................... 58
GISW_KEI ..................................................................... 58
GISW_NRI ..................................................................... 59
GISW_ICTDI ................................................................... 59
GISW_EGRI ................................................................... 60
GISW_ERI ..................................................................... 60
5. Implementation ......................................................... 63
Structure of application .................................................. 63
   Admin folder ............................................................... 63
   GISW folder ................................................................ 64
   Includes folder ............................................................ 64
   Lib folder .................................................................... 65
   User folder ................................................................... 65
Importing data from GISW ............................................. 66
Charts and world map graphs ....................................... 68
   Charts ....................................................................... 68
   World map graphs ...................................................... 70
Testing .......................................................................... 72
6. Results & Conclusions ................................................ 73
Goals ............................................................................ 73
Future expansions ......................................................... 73
   More customization ..................................................... 73
   More importing options .............................................. 73
   New charts ................................................................... 73
Personal valuation ......................................................... 74
7. References .................................................................. 75
Appendix I: Installation guide ....................................... 77
   Install Apache ............................................................. 78
   Install MySQL ............................................................ 83
   Install PHP ................................................................. 89
   Install Zend Optimizer ................................................. 94
   Install phpMyAdmin ..................................................... 99
Setup application .......................................................... 100
   Copy web application files .......................................... 100
   Setting database ........................................................ 100
   Change db.php password ............................................ 101
   Setup internal file ....................................................... 101
Appendix II: Operation manual ...................................... 103
   User guide ................................................................. 103
   Consult development data in ICT ............................... 103
   Login and register ..................................................... 105
   Initial page and profile ................................................. 106
Upload data and create charts ................................................................. 109
Admin guide ......................................................................................... 113
Admin users ......................................................................................... 114
Admin data .......................................................................................... 114
Admin charts ....................................................................................... 115
1. Introduction

The aim of the project is to develop a software from pieces of free software that allows to gather and to represent manifold indicators of development of the information's society to be able to visualize it in a cute charts and graphs.

From the study on the development of the information and communication technologies (ICT) in the world, Global Information Society Watch (GISW)[1], in which the Universitat Politècnica de Catalunya (UPC) and Pangea.org have participated, this project wants to offer web application to show data in graphical form in a comprehensible way for a public constituted by scholars, adult citizens or researchers on indicator of development of the Information Society in the world.

Furthermore, with this visual tool I want to offer the possibility to create your own graphics importing from your own data by different sources.
2. Methodology

For development of this project I have chosen a Unified Software Development Process, also know as Unified Process, is a popular iterative and incremental software development process.

In concrete I have chosen the Rational Unified Process (RUP), because is the best-known and extensively documented refinement of the Unified Process.

RUP was created by the Rational Software Corporation, a division of IBM since 2002. It's based on a set of six key principles for business-driven development, that they are: Adapt the process, Balance stakeholder priorities, Collaborate across teams, Demonstrate value iteratively, Elevate level of abstraction, Focus continuously on quality, known as ABCDEF.

The RUP lifecycle has four phases:
- Inception phase
- Elaboration phase
- Construction phase
- Transition phase

Depending on the project we can do several iterations on each phase. In each iteration there are different tasks categorized into disciplines:
- Business modeling discipline
- Requirements discipline
- Analysis and design discipline
- Implementation discipline
- Test discipline
- Deployment discipline
- Configuration and change management discipline
- Project management discipline
- Environment discipline.
3. Initial studies

Requirements

The principal requirement is that the graphical tool has to be the most free software possible and anyone could use it freely. Will have to analyzes different graphical tools and choose for the most interesting and useful and then create one graphic tool like a mashup, from combination of more than one.

The tool has to be able to consult graphically development data from Global Information Society Watch 2007 report (GISW), this report is the first in a series of annual reports, looks at state of the field of information and communication technology (ICT) policy at local and global levels and particularly how policy impacts on the lives of people living in developing countries. Studies of the report have been for twenty-two countries from four regions: Africa (Democratic Republic of Congo, Egypt, Ethiopia, Kenya, Nigeria, South Africa and Uganda); Asia (Bangladesh, India, Pakistan and the Philippines); Latin America (Argentina, Brazil, Colombia, Ecuador, Mexico and Peru); and Eastern Europe (Bosnia and Herzegovina, Bulgaria, Croatia and Romania), with one country from a Western Europe (Spain). The ICT indices available online and that we will can visualize graphically are: Digital Opportunity Index, Knowledge Economy Index, Networked Readiness Index, Index of ICT Diffusion, E-Government Readiness Index and E-Readiness Index.

Any user has to be capable to create their charts and world’s graphs from their data using the tool. It means that one user must be able to import their own data to the web application in a easy way, and later will can create statistics charts and world’s graphs from the data.

As any other application, has to have an easy and interactive interface for all users to register, to login, to show/edit profile, to delete account and to admin their data and charts. Also has to have an admin interface to administrate the users, data and charts of all the system.
Gantt diagram
Here we have the Gantt diagram that shows the time of dedication planned for the different tasks or activities over total time duration of the project.

Risk analysis
In this subchapter, are going to analyze the risks of the project, with the possibility that it’s happens and the importance, 1 for the most important.

<table>
<thead>
<tr>
<th>Risk</th>
<th>Possibility</th>
<th>Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph tools not sufficiently powerful</td>
<td>25%</td>
<td>1</td>
</tr>
<tr>
<td>Difficulty import data development GISW</td>
<td>30%</td>
<td>2</td>
</tr>
<tr>
<td>Lost data of the database</td>
<td>5%</td>
<td>1</td>
</tr>
<tr>
<td>Insufficient time to implement everything</td>
<td>5%</td>
<td>2</td>
</tr>
</tbody>
</table>

Risk: Graph tools not sufficiently powerful
How to monitor it? Deeply analyzing the requirements with the tool.
How to mitigate it? Choose the tool that is used by many people with a good documentation, with a often updates.

Risk: Difficulty import data development GISW
How to monitor it? Analyzing different ways to import it.
How to mitigate it? Choose the easy and feasible way to import the data.

Risk: Lost data of the database
How to monitor it? Check if no data lost.
How to mitigate it? Make backups frequently.

Risk: Insufficient time to implement everything
How to monitor it? Check the planning if we are on time
How to mitigate it? If there isn’t enough time to do everything redo planning and prioritizing the most important.
Technologies

Apache

Apache\(^{[1]}\) is one of the greatest triumphs of free software, maintained for years as the king of web servers. With a huge team of volunteers throughout the entire network, it has managed to beat commercial companies of the stature of Microsoft and Netscape. Not only that, it has succeeded in attracting to the almighty IBM is supporting Apache both in the area of support and development.

Apache is a flexible web server, fast and efficient, continuously updated and adapted to the new protocols. Among its features we can highlight:

- **Multiplatform**: Apache may endure clearly and efficiently a wide variety of operating systems. And while Unix/Linux is the platform excellence of this product, in each new version provides a more efficient implementation on Windows.

- **Modular**: It can be adapted to different environments and needs, with different modules that provide support, and API programming modules, for the development of specific modules.

- Based on **threads** in the version 2.0, which improves performance with respect to their ancestors.

- It encourages **feedback** from users, getting new ideas, bug reports and patches for the solution of the same, which is a free support.

- In addition, there are specialized companies that provide **professional support**, at an affordable price, for those who require more sophisticated settings.

- It is developed in an **open** way, and it is completely free.

- **Extensible**: thanks to be modular various extensions have been developed among which PHP, a programming language server side that facilitates the integration of the web with the world of databases.

- **Secure**: Apache-SSL is the Apache server but with new security features such as encryption and authentication.

Despite all its strengths, the main disadvantage of this server is that his administration is done through scripting tools through the command line. In other words, unlike popular commercial servers, such as Microsoft-IIS, has no GUI management tool that allows work more comfortable.

Still, the rugged design and extensibility provided by the Apache, coupled with its position of freeware and the availability of its source code to the public,
Apache makes a good choice for companies that want to have a presence on the Internet, as well as to any individual.

Version used: Apache 2 Server

**PHP**

PHP\(^{[12]}\) is a computer programming language originally designed for producing dynamic web pages. The name PHP is a recursive initialism for PHP: Hypertext Preprocessor.

PHP is used mainly in server-side scripting, but can be used from a command line interface or in standalone graphical applications.

The main implementation is produced by The PHP Group and released under the PHP License. This implementation serves to define a standard for PHP, as there is no formal specification. It is considered to be free software by the Free Software Foundation.

Version used: 5.2.3

**MySQL**

MySQL\(^{[13]}\) is a SQL (Standard Query Language) database engine that is well-suited for most web sites, developed and distributed by MySQL AB. His indisputable performance, along with a great ease when it comes to integrating with scripting languages for the Web, such as Perl and PHP, place it as a great favorite among Webmasters.

MySQL is distributed under the GNU General Public License (GPL), probably the Open Source license better known. The GPL is contagious in the sense that when a program is released with another GPL program, resulting product should be distributed under the GPL.

Version used: MySQL 5.0.41

**Libraries**

**Graph tools**

Charts are supposed to visualize data in order to give a more profound understanding of the nature of a given problem or recent developments. Whatever type of data presentation we prefer (pie charts, bubble charts, bar graphs, network diagrams etc.), we can create charts in graphic editors manually or use special desktop-software instead. In both cases you have a major problem: once you'd like to update an old chart, or create a new one, you have to run the application and create new images over and over again. That's not flexible. That's also not usable if you'd like to update your chart live.
Server-based solutions, implemented with Flash, JavaScript or pure CSS, offer a more flexible alternative. In fact, since Flash offers significant advantages over static data presentation with CSS and JavaScript, most solutions use it for dynamic data visualization. The data itself is often stored in XML-files, which are loaded and updated via PHP or ASP. The price range varies enormously depending on the flexibility and level of customization we’d like to have. However, if we don’t want to pay, we don’t have to, there are powerful free solutions as well.

**Flash-based solutions**

*Open Flash Chart*

Open Flash Chart (OFC)\(^{[14]}\) is an open-source project. The Flash chart offers 35 chart variations; among them a number of bar charts, pie charts and line charts. Provided tutorials explain how the script can be extended with further functionality such as mouse-over effects and how the database can be queried for some values and the results then displayed in a graph. OFC uses Flash and text file, or a .php, Perl, Python, Java or another dynamic page; data can be stored in plain text (.txt). The package, available via Sourceforge, includes a number of examples and samples.

*FusionCharts Free*

FusionCharts Free\(^{[15]}\) is a Flash charting component that can be used to render data-driven & animated charts for web applications and presentations. It is a cross-browser and cross-platform solution that can be used with ASP.NET, ASP, PHP, JSP, ColdFusion, Ruby on Rails, simple HTML pages or PowerPoint Presentations to deliver interactive and powerful flash charts.
You can generate 2D/3D Column Charts, Line Charts, 2D/3D Pie & Doughnut Charts, Area Charts, Stacked Charts, Combination Charts and Candlestick Charts. The data is stored in XML. The full version of the application has more animated charts, visual effects (shadow, glow etc.).

**XML/SWF Charts**

XML/SWF Charts\(^{16}\) offers a simple and dynamic solution for interactive Flash-based charts. Columns, circles, bars and pies can be displayed in 3D. You create an XML source to describe a chart, then pass it to this tool's flash file to generate the chart. The XML source can be prepared manually, or generated dynamically using any scripting language (PHP, ASP, CFML, Perl, etc.).

It's free, however all charts are by default linked to the website source, unless you've purchased a license.

**PHP/SWF Charts**

PHP/SWF Charts\(^{17}\) is a simple, yet powerful PHP tool to create attractive web charts and graphs from dynamic data. This is the version for PHP of the same tool as XML/SWF Charts.

Use PHP scripts to generate or gather the data from databases, then pass it to this tool to generate Flash (swf) charts and graphs.

It's free, however all charts are by default linked to the website source, unless you've purchased a license.
amCharts

amCharts\(^{[18]}\) offers a flexible and easily customizable solution for generating charts "on the fly". There are 4 sets with predefined Pie & Donut, Line & Area, Column & Bar and Scatter & Bubble. Generated Flash-files are dynamic and can be presented in 2D or 3D. The loader can load data from XML or CSV, so you can easily export data from Excel, dynamically generate data file with PHP, ASP, .NET or other programming language. Some flash-charts also have animation effects.

amCharts is linkware which means that you have to place a link to the source if you'd like to use it for free. You can also purchase a single site license for 85 Euros.

amMap

amMap\(^{[19]}\) is another flash application from the same developer as amCharts, Antanas Marcelionis, also offers a customizable flash-based solution for interactive maps, amMap: same conditions, same level of flexibility.

Rich Chart Server for .NET

Rich Chart Server\(^{[20]}\) generates captivating interactive Flash charts from ASP.NET and Windows Forms applications. Rich Chart Server uses flexible templates to integrate dynamic content from a variety of data sources. Price $295.
**Flash Charts Pro**

Flash Charts Pro[^21], ASP Chart and PHP Flash Charting tool, comes with 9 main categories and more than 30 chart types. No installation is required. Upload the Flash file and generate a list of ingoing values - or use a script. Customization is possible via the XML file itself. The tool offers multi-lingual (UTF-8) characters support, customizable animation, grid component and dynamically resizable charts. The price range starts at $59. There is no trial-version.

![Pie Chart](image)

**Swift Chart Generator**

Swift Chart Generator[^22] is the server-side solution to dynamically serve eye-catching animated charts from Active Server Pages (ASP), PHP scripts or JSP scripts. Query data from a database, apply your own style template created with the Swift Chart authoring tool and Swift Chart Generator instantly generates high-impact charts in Flash format, PNG/JPG format, SVG format and PDF format. Pricing starts at $149.

![Bar Chart](image)

**AnyChart Flash**

AnyChart[^23] is a more powerful alternative to Open Flash Chart. It is fully cross-browser and cross-platform and can generate Pie, Bar, Line, Candlestick, Area, Column, Bubble, Spline, Dot/Marker, Scatter, Candlestick and Doughnut charts. It can be used with virtually all scripting languages: ASP.NET, ASP, PHP, JSP, ColdFusion, Ruby on Rails, simple HTML pages and also PowerPoint presentations.

![Area Chart](image)

AnyCharts is driven by XML as universal data interface, so you can visualize data from any data format: from files in text, Excel, CSV formats to MySQL, MS SQL, and Oracle database servers. Users can save the generated charts as an image for email sending or chart export. AnyChart is huge, and a developer licence starts at $649.00.
PHP/ASP Libraries

**Libchart**

Libchart[^24] is a simple PHP-library. You can create Line, Bar (horizontal and vertical) and Pie charts. The tool requires PHP compiled with GD Graphics Library and FreeType.

Libchart is a free software distributed under the terms of the GNU General Public License (GPL).

**Image_Graph**

[^25]: Image_Graph[^25], known as GraPHPite, utilizes PHP's object oriented model to enable users/developers to create highly versatile graphs in a simple way. There are 14 highly customizable chart types including line, bar, area, pie, impulse, dot/scatter, step, candlestick, box & whisker, radar, etc.

The free script requires PHP4 or PHP5 support with GD 2 support for optimal performance and PEAR package, including Image_Color and Image_Canvas package. The site offers over 80 samples of how the tool can be used.

**JpGraph**

[^26]: JpGraph[^26] is a Object-Oriented Graph creating library for PHP. The library is completely written in PHP and can be used in any PHP scripts. The library can be used to create numerous types of graphs either on-line or written to a file. Line plots, filled line plots, step line plots, bar and pie charts, ring and scatter plots, polar and canvas plots as well as Gantt Charts.

[^27]: .netCharting[^27] utilizes the .NET framework and GDI+ providing a managed charting solution for C# and VB.NET developers working with ASP.NET or WinForms. The tool offers a huge choice of available charts. In most cases 2D and 3D-views are available. Price range starts at $395 for a single site license.
JavaScript-based solutions

Emprise Javascript Charts

Emprise JavaScript Charts[28] is done in JavaScript, no Flash support is required. The script includes features such as hints, mouse tracking, mouse events, key tracking and events, zooming, scrolling, and crosshairs which might raise interactivity and user experience. EJSChart supports XML-formatted data.

Every aspect of the charting display can be configured and customized through well-documented properties and methods. An interesting solution, offering a number of designs and behaviours. Pricing starts at $100. There is a watermarked private version available for free.

Plotkit

PlotKit[28] is an experimental JavaScript-based chart and graph plotting library. It has support for HTML Canvas and SVG via Adobe SVG Viewer as well as native browser support. The script requires MochiKit JavaScript library, version 1.3 or higher. Released under the BSD License.

Plotr

Plotr[30] is a lightweight charting framework based on PlotKit draws simple charts. It’s free, and there is no use of Flash. There are also no interactive elements. Released under the BSD License.

jQuery Chart Plugin

jQuery Chart Plugin[31] is a extension for jQuery platform. You can create Line, Bar and Area charts.
Dojo Charting Engine

Dojo Charting Engine[32] provides many common chart types—including line, bar, area and pie charts—and gives you the flexibility to mix many types of charts for unprecedented visualization.

Timeplot

Timeplot[33] is a DHTML-based AJAX widget for plotting time series and overlay time-based events over.

Just like Google Maps, Timeplot can be used with zero software installation, server-side or client-side. You can populate Timeplot with data by pointing it to a space or comma separated file. Timeplot also supports Timeline's XML format, meaning that you can reuse the same data file of your Timeline and overlay them over a time series plot.
CSS-based solutions

These solutions provide easy methods to generate static charts and style them; however, they can't compete with the interactivity and the nice aspect of Flash, PHP libraries or JavaScript offers. These solutions are with very limited charts, only bars in the majority of the cases.

Ajax MGraph: http://dragan.yourtree.org/code/ajax-mgraph/
CSS Vertical bar Graphs:
http://meyerweb.com/eric/css/edge/bargraph/demo.html/
CSS Diagrams: http://www.surfare.net/~toolman/temp/diagram.html
CSS for Bar Graphs: http://applestooranges.com/blog/post/css-for-bar-graphs/?id=55
Pie Chart with CSS: http://petewilliamsagency.com/css/examples/pie/
Simple Accessible Charts: http://www.grassegger.at/xperiment/sac-simple-accessible-charts-css/
CSS Line Charts: http://www.khanate.co.uk/graph/
Vertical bar graphs with CSS and PHP:
http://www.terril.ca/design/vertical_bar_graphs/
An accessible bar chart: http://www.standards-schmandards.com/exhibits/barchart/
Stu Nicholls list bar chart: http://www.cssplay.co.uk/menu/barchart.html
Chart Demo (WebFX) (uses javascript):
http://webfx.eae.net/dhtml/chart/demo.html

Best solutions

Now, I will analyze more deeply the characteristics of the most interesting graph tools and then I will decide which one I will use.

In this project I'm going to use these charts types: column, line, area, world chart map and would be interesting to use also time lineal charts.

First, I discard all CSS-based solutions because they are very limited, without a nice aspect and the majority only do column charts.

Second, I also discard Rich Chart Server for .NET, .netCharting and Swiff Chart Generator because these solutions are based on ASP or ASP.NET, aren't free languages, or are based on Microsoft Windows platforms.

Third, I discard XML/SWF Charts, PHP/SWF Charts, Flash Charts Pro and AnyChart Flash because these solutions are necessary to pay for use it and they don't have a free version or the free version possible has a very big link to the website.

So, we have the options on the next table:
<table>
<thead>
<tr>
<th>Based on</th>
<th>Tool</th>
<th>Language</th>
<th>Type graphs</th>
<th>Animation</th>
<th>Interactive</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flash</td>
<td>Open Flash Chart</td>
<td>PHP</td>
<td>Column, 3D Bar, Area, Lines, Pie.</td>
<td>Yes</td>
<td>Yes</td>
<td>GNU General Public License (GPL)</td>
</tr>
<tr>
<td></td>
<td>FusionCharts Free</td>
<td>XML</td>
<td>2D/3D Column, Line, 2D/3D Pie &amp; Doughnut, Area, Stacked, Combination, Candlestick.</td>
<td>Yes</td>
<td>Yes</td>
<td>Terms of use: <a href="http://www.fusioncharts.com/free/TermsOfUse.asp?gMenuItemId=9">http://www.fusioncharts.com/free/TermsOfUse.asp?gMenuItemId=9</a></td>
</tr>
<tr>
<td></td>
<td>amCharts</td>
<td>XML</td>
<td>Pie &amp; Donut, Line &amp; Area, Column &amp; Bar and Scatter &amp; Bubble</td>
<td>Yes</td>
<td>Yes</td>
<td>Free version will display a small link to the amCharts/amMap website on your charts. Single site license: 85 €. Enterprise license: 275 €.</td>
</tr>
<tr>
<td></td>
<td>amMap</td>
<td>XML</td>
<td>World Map Chart: customize colors, size, add lines and text</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>PHP/ASP Libraries</td>
<td>Libchart</td>
<td>PHP</td>
<td>Horizontal &amp; vertical bar, line and pie charts.</td>
<td>No</td>
<td>No</td>
<td>GNU General Public License (GPL)</td>
</tr>
<tr>
<td></td>
<td>Image_Graph</td>
<td>PHP</td>
<td>Line, bar, area, pie, impulse, dot/scatter, step, candlestick, box &amp; whisker, radar and stacked charts.</td>
<td>No</td>
<td>No</td>
<td>Open Source</td>
</tr>
<tr>
<td></td>
<td>JpGraph</td>
<td>PHP</td>
<td>Line, Bar, 2D/3D Pie, Ring, Field, Splines, Geo maps, Stock, Polar, Radar, Gantt, Impuls, Error, Scatter, Ballon and Canvas.</td>
<td>No</td>
<td>No</td>
<td>For non-commercial, open-source or educational use: QPL 1.0. For commercial use: single license 85 €</td>
</tr>
<tr>
<td>JavaScript</td>
<td>Plotkit</td>
<td>Javascript</td>
<td>Bar, Line, Area and Pie.</td>
<td>No</td>
<td>No</td>
<td>BSD License</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Plotr</td>
<td>Javascript</td>
<td>Bar, Line, Area and Pie. With legend</td>
<td>No</td>
<td>No</td>
<td>BSD License</td>
<td></td>
</tr>
<tr>
<td>jQuery Chart Plugin</td>
<td>Javascript</td>
<td>Line, Bar and Area charts.</td>
<td>No</td>
<td>No</td>
<td>MIT and GPL licenses</td>
<td></td>
</tr>
<tr>
<td>Dojo Charting Engine</td>
<td>Javascript</td>
<td>Line, CurvedLine, DataBar, Area, CurvedArea, Pie, Scatter, Bar, StackedArea, StackedCurvedArea, HighLow, HighLowClose, HighLowOpenClose and Bubble.</td>
<td>No</td>
<td>No</td>
<td>Open Source: <a href="http://dojotoolkit.org/about/license">http://dojotoolkit.org/about/license</a></td>
<td></td>
</tr>
<tr>
<td>Timeplot</td>
<td>Javascript</td>
<td>Linial plot with events</td>
<td>No</td>
<td>Yes</td>
<td>BSD License</td>
<td></td>
</tr>
</tbody>
</table>
I’ve decided to use amMap for do the world map charts, I choose this solution although it’s not free and the free version have a little link to the website of the creator is the unique good solution I’ve found for this kind of map charts.

For typical charts I will use Open Flash Chart, because this solution is with flash, have animation and is interactive, and it’s open source under GPL License. I don’t discard to use Dojo Chart Engine too.

Finally, I thing that would be interesting to use Timeplot for do the time lineal charts with overlay events, it’s a nice solution with only javascript and it’s BSD License.

Script.aculo.us
Script.aculo.us[34] provides with easy-to-use, cross-browser user interface JavaScript libraries, that inside are animation framework, drag and drop, Ajax controls DOM utilities, and unit testing. It’s an add-on to the Prototype framework[35] and is released under the MIT license. Version: Script.aculo.us 1.8.0 with Prototype 1.6.0

Validation library
Validation library[36] is javascript library that helps to do easy validation field on forms. Requires Prototype and script.aculo.us. Version: 1.5.4.1
Use Case Diagram

Actors

Anonymous: is somebody who can access to the web application, this role only can consult development data from GISW and register to be a registered user.

Registered User: is not anonymous user because is registered, this role inherits from Anonymous and can login/logout on the system, upload/import data and creates charts.

Admin: is the administrator of the system, inherit from Registered User and can administrate the users, the user's data and the user's charts.

Uses Cases

Register: an anonymous user can register to the system, then can login like registered user and use services for registered users.

Consult development charts by index: shows on a visual chart the development data from GISW by the parameters of an index.

Consult development charts by country: show on visuals charts the development data from GISW by country.

Consult development world graph: show development data from GISW on a global view in a world map graph by an index and a parameter selected.
Login: validates a registered user or admin into the web application and show his personal start page.

Logout: user exits the web application and is converted like anonymous user.

Show profile: show all the profile of the user: name, surname, mail, member since and last login.

Change profile: modifies the user's profile: name, surname and mail.

Change password: modifies the user's password.

Delete user's account: delete the own user account and all user's data and charts.

Upload/import data: registered users and admin users can upload/import their data and use it later for generates their graphs.

List user's data: list all the data that belongs to the user.

Show user's data: show a user's data: name, description and data.

Delete user's data: delete a user's data and all user's charts that belongs to the data.
Create chart: registered users and admin users can create their charts from their own data.

List user's charts: list all the charts that belong to the user.

Show user's chart: show a user's chart: title, description, data's name and the chart.

Delete user's chart: delete a user's chart.

List users: list all users that exist on the system.

List admins: list all admins that exist on the system.

Change user's role: modifies the role of a user, convert a registered user to an admin, and vice versa.

Delete user: delete a user and all user's data and charts.

List all user's data: list all user's data that exist on the system.

List all user's charts: list all user's charts that exist on the system.
Description of architecture

Conceptual database model

<table>
<thead>
<tr>
<th>users</th>
<th>users_data</th>
<th>users_charts</th>
<th>countries_world_map</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0..n</td>
<td>0..n</td>
<td></td>
</tr>
</tbody>
</table>

**users**: contains the information about all users on the system.

**users_data**: contains all the data imported by the users.

**user_charts**: contains all the charts created by the users using their own data.

**countries_world_map**: contains all the countries that can be used in the flash world map.
**GISW_countries**: contains all the countries that the system has data about the indices analyzed in the report of GISW.

**GISW_DOI**: contains data from the countries by the Digital Opportunity Index.

**GISW KEI**: contains data from the countries by the Knowledge Economy Index.

**GISW_NRI**: contains data from the countries by the Networked Readiness Index.

**GISW_IICTD**: contains data from the countries by the Index of ICT Diffusion.

**GISW_EGRI**: contains data from the countries by the E-Government Readiness Index.

**GISW_ERI**: contains data from the countries by the E-Readiness Index.
4. Analysis & Design

Use Case Specification
For each use case of the diagrams described in the previous chapter we will make a specification in which we will take a description of the actions and requirements of the use case.

Register
Actors: Anonymous user.
Summary: register a new user to the system.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The anonymous user began this use case when entering on the register page of the web application and fill out the form with his name, surname, username, password, confirmation password and mail in the appropriate place.</td>
<td>2. Validate the form and shows that the new user has been successfully registered, with a link to login page.</td>
</tr>
</tbody>
</table>

Alternative:
- **Error 2**: Username is already used. Show error.
- **Error 2**: Username, password, confirmation password or mail fields are empty. Show error.
- **Error 2**: Password is less than 6 characters or is the same as username. Show error.
- **Error 2**: Password and confirmation password doesn’t match. Show error.
- **Error 2**: Mail field has not a valid email address. Show error.

Graphic interface:

![Registration form]

*Please fill out the form.*
- **Name:**
- **Surname:**
- **Username:**
- **Password:**
- **Confirm password:**
- **Mail:**

*Register: Reset*

![Registration form]

*Please fill out the form.*
- **Name:**
- **Surname:**
- **Username:**
- **Password:**
- **Confirm password:**
- **Mail:**

*Register: Reset*
Consult development charts by index

Actors: Anonymous user.
Summary: shows on a visual chart the development data from GISW by the attributes of an index.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The anonymous user began this use case when selecting the option to show development data for a concrete index and attribute of it.</td>
<td>2. Returns all information about the index selected and data chart for the selected attribute of the index.</td>
</tr>
</tbody>
</table>

Alternative:
- Error 2: Not selected any attribute of the index. Show for a default attribute of the index.

Graphic interface:

[Image of digital opportunity index (doI)]

Consult development charts by country

**Actors:** Anonymous user.

**Summary:** show on visuals charts the development data from GISW by a country.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The anonymous user began this use case when selecting the option to show development data for a concrete country.</td>
<td>2. Returns all data of the indices about the selected country.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 2:** Not exists indice's data for that country. Inform that.
- **Error 2:** Not exists that country. Show error.

**Graphic interface:**

![Graph showing digital opportunity index (DOI) for Uganda in 2005 and 2006](image)


<table>
<thead>
<tr>
<th>Year</th>
<th>Opportunity</th>
<th>Infrastructure</th>
<th>Utilisation</th>
<th>DOI</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>0.45</td>
<td>0.01</td>
<td>0</td>
<td>0.15</td>
<td>152</td>
</tr>
<tr>
<td>2006</td>
<td>0.46</td>
<td>0.02</td>
<td>0.01</td>
<td>0.16</td>
<td>158</td>
</tr>
</tbody>
</table>

- **knowledge economy index (kei)**
- **networked readiness index (nri)**
- **index of ict diffusion**
Consult development world graph

**Actors:** Anonymous user.

**Summary:** show development data from GISW on a global view in a world map graph by an index and attribute selected.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The anonymous user began this use case when selecting the option world map in the principal menu or in the report of an index.</td>
<td>2. Show the world map graph with development data for a concrete index, attribute and year selected.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 2:** Not selected any index. Show world map graph for a default index, attribute and year.
- **Error 2:** Not selected any attribute or year of the index. Show world map graph for a default attribute or year of the index.

**Graphic interface:**

![Digital Opportunity Index (DOI) map](image-url)
Login

Actors: Registered user.

Summary: validates a registered user or admin into the web application and show his personal start page.

Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when entering on the login page of</td>
<td>2. Validates the user and redirect to</td>
</tr>
<tr>
<td>the web application and enters user name and password in the appropriate place.</td>
<td>principal start page.</td>
</tr>
</tbody>
</table>

Alternative:
- Error 2: Username or password fields are empty. Show error.
- Error 2: Username or password fields are incorrect. Show error.

Graphic interface:
Logout
Actors: Registered user.
Summary: user exits the web application and is converted like anonymous user.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option logout.</td>
<td>2. Delete session information about registered user and redirects to home page.</td>
</tr>
</tbody>
</table>

Use case: Show profile
Actors: Registered user.
Summary: show the profile of the user: name, surname, mail, member since and last login.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when clicks over his username in the user menu.</td>
<td>2. Return the user's profile.</td>
</tr>
</tbody>
</table>

Graphic interface:

![marniko's profile]

Name: marc
Surname: nicolas
Mail: marc.mko@gmail.com

Edit profile
Change password
Member since: 2007-11-30 15:50:22
Last login: 2007-12-18 15:54:57
Delete your account
Change profile

Actors: Registered user.
Summary: modifies the user’s profile: name, surname and mail.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor’s actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option edit profile in the profile page and modifies the fields name, surname and mail in the appropriate place.</td>
<td>2. Validates the form and redirect to profile page.</td>
</tr>
</tbody>
</table>

Alternative:
- Error 2: Mail field is empty or isn’t a valid email address. Show error.

Graphic interface:

![Create profile interface](image)

Change password

Actors: Registered user.
Summary: modifies the user’s password.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor’s actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option change password in the profile page and enters the field password and new password two times in the appropriate place.</td>
<td>2. Validates the user’s password and new password. Modifies the user’s password and redirect to profile page.</td>
</tr>
</tbody>
</table>
Alternative:
- **Error 2:** Fields password, new password or confirmation of new password are empty. Show error.
- **Error 2:** New password and confirmation of new password doesn’t match. Show error.
- **Error 2:** Password is incorrect. Show error.

Graphic interface:

```
<table>
<thead>
<tr>
<th>Password:</th>
<th>Password:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The password is not correct!</td>
</tr>
<tr>
<td>New password:</td>
<td>New password:</td>
</tr>
<tr>
<td></td>
<td>Your confirmation password does not match your first password, please try again.</td>
</tr>
<tr>
<td>Retype new password:</td>
<td>Retype new password:</td>
</tr>
<tr>
<td></td>
<td>Change</td>
</tr>
</tbody>
</table>
```

Delete user’s account

**Actors:** Registered user.

**Summary:** delete the own user account and all user’s data and charts.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor’s actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option delete account in the profile page.</td>
<td>2. Ask to the user if are sure to delete the account.</td>
</tr>
<tr>
<td>3. User selects yes or no.</td>
<td>4. If the user selects yes delete all user information, data and charts belongs to the user and shows that the new user has been successfully deleted, with a link to home page. Else, user selects no, redirects to profile page.</td>
</tr>
</tbody>
</table>

Alternative:
- **Error 4:** User doesn’t exist. Show error.
Graphic interface:

```
delete account
All your data and charts will be deleted.
Are you sure?
Yes | No
```

```
user joel deleted correctly!
```

Upload/import data

**Actors:** Registered user.

**Summary:** registered users and admin users can upload/import their data and use later for generates their graphs.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option upload data in the user menu and selects a .csv file to upload.</td>
<td>2. Reads the csv file and keep the info of the file into a XML, show the data uploaded.</td>
</tr>
<tr>
<td>3. Enters a name and description for the data in the appropriate place.</td>
<td>4. Validate the form and save the new data, show that the new data has been successfully saved.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 2:** File field empty. Show error.
- **Error 2:** File is not a csv file. Show error.
- **Error 4:** Name field is empty. Show error.

Graphic interface:

```
upload your data
from a .csv file
Select file: [Browse] (File size must be less than 2 MB)
Upload
```
List user's data

Actors: Registered user.

Summary: list all the data that belongs to the user.

Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option my data.</td>
<td>2. Return a list of all data of the user.</td>
</tr>
</tbody>
</table>

Alternative:

- **Error 2**: Doesn't exist any data of the user. Inform that.
Show user’s data

**Actors:** Registered user.

**Summary:** show a user’s data: name, description and data.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor’s actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects to show a data with id.</td>
<td>2. Return the information about the data: name, description and data.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 2:** Doesn’t exist any data with this id. Show error.
- **Error 2:** The user is not an admin user and the data doesn’t belong to the user. Show error.

Graphic interface:

```
my data

<table>
<thead>
<tr>
<th>id</th>
<th>Name</th>
<th>Description</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>DOI</td>
<td>Digital Opportunity index</td>
<td>Show delete</td>
</tr>
<tr>
<td>11</td>
<td>New data</td>
<td>This data contains…</td>
<td>Show delete</td>
</tr>
<tr>
<td>12</td>
<td>New data</td>
<td>This data contains…</td>
<td>Show delete</td>
</tr>
</tbody>
</table>

No data uploaded

Upload new data
```

See data as XML

Back to my data

```
my data

You don't have access to see this data, don't belongs to you.

Back to my data
```

```
my data

Doesn't exists any data with id=2.

Back to my data
```

```
Delete user's data

**Actors:** Registered user.

**Summary:** delete a user's data and all user's charts that belongs to the data.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option delete data in the list data page.</td>
<td>2. Ask to the user if are sure to delete the data.</td>
</tr>
<tr>
<td>3. User selects yes or no.</td>
<td>4. If the user selects yes delete data, and all the charts that belongs to this data. Else, user selects no, don't do anything.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 4:** Data with the id doesn't exist. Show error.

Create chart

**Actors:** Registered user.

**Summary:** create a chart from the own data.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects create a chart in the user menu and enters a title and description, selects the data and chart type in appropriate place.</td>
<td>2. Validate the form and save the new chart, show that the new chart has been successfully saved.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 2:** Title field is empty. Show error.
List user’s chart

Actors: Registered user.

Summary: List all the charts that belong to the user.

Sequence of events:

<table>
<thead>
<tr>
<th>Actor’s actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option my charts.</td>
<td>2. Return a list of all charts of the user.</td>
</tr>
</tbody>
</table>

Alternative:
- Error 2: Doesn’t exist any chart of the user. Inform that.

Graphic interface:
Show user's chart

Actors: Registered user.
Summary: show a user's chart: title, description, data's name and the chart.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects to show a chart with id.</td>
<td>2. Return the information about the chart: title, description, data’s name and the chart.</td>
</tr>
</tbody>
</table>

Alternative:
- **Error 2**: Doesn't exist any chart with this id. Show error.
- **Error 2**: The user is not an admin user and the chart doesn't belong to the user. Show error.

Graphic interface:

![Graph Image](image-url)

my charts

doi area

Digital Opportunity Index
DOI 2005
Type chart Area


DOI area - Opportunity 2005

Back to my charts
Delete user's chart

**Actors:** Registered user.

**Summary:** delete a user's chart.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The registered user began this use case when selects the option delete chart in the list charts page.</td>
<td>2. Ask to the user if are sure to delete the chart.</td>
</tr>
<tr>
<td>3. User selects yes or no.</td>
<td>4. If the user selects yes delete chart. Else, user selects no, don't do anything.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 4:** Chart with this id doesn't exist. Show error.

List users

**Actors:** Admin user.

**Summary:** list all users that exist on the system.

**Sequence of events:**

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The admin user began this use case when selects the option admin users to see all the registered users.</td>
<td>2. Return a list of all users on the system.</td>
</tr>
</tbody>
</table>

**Alternative:**
- **Error 2:** Doesn't exist any user. Inform that.

Graphic interface:

```
admin all users
users [o] refresh
Id  Username    Name   Surname  Email          Created       Last login    Last modif.
9  marcokiko    marc    nicolas  marc@kiko@gmail.com  2007-11-30 15:30:22 2008-01-01 19:30:07  2008-01-01 19:30:07 convert to admin delete
7  marin        Martin   Mihanec  marin@outlook.com    2007-11-19 12:16:14 2007-12-30 23:03:26  2007-12-30 23:03:26 convert to admin delete
2  niko        niko     chan    niko@niko.co       2007-11-09 13:32:07  00:00:00 00:00:00  00:00:00 2007-12-28 16:28:47 convert to admin delete

admin all users
users [o] refresh
No users registered
```
List admins

Actors: Admin user.
Summary: List all admins that exist on the system.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The admin user began this use case when selects the option admin users to see all the admin users.</td>
<td>2. Return a list of all admin users on the system.</td>
</tr>
</tbody>
</table>

Alternative:
- Error 2: Doesn’t exist any admin. Inform that.

Graphic interface:

admin all users

<table>
<thead>
<tr>
<th>id</th>
<th>username</th>
<th>Name</th>
<th>Surname</th>
<th>mail</th>
<th>Created</th>
<th>Last login</th>
<th>Last modif.</th>
<th>actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>niko</td>
<td>nika</td>
<td>chan</td>
<td><a href="mailto:niko@nike.cz">niko@nike.cz</a></td>
<td>2007-11-08 13:52:01</td>
<td>2000-00-00 00:00:00</td>
<td>2008-01-01 21:11:16</td>
<td>delete</td>
</tr>
</tbody>
</table>

admin all users

<table>
<thead>
<tr>
<th>id</th>
<th>username</th>
<th>Name</th>
<th>Surname</th>
<th>mail</th>
<th>Created</th>
<th>Last login</th>
<th>Last modif.</th>
<th>actions</th>
</tr>
</thead>
</table>

Change user's role

Actors: Admin user.
Summary: Modifies the role of a user, convert a registered user to an admin, and vice versa.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The admin user began this use case when selects the option convert to admin or convert to user in the users or admin list page.</td>
<td>2. Update the role of the user or admin selected, if is admin it’s converted to a registered user, and vice versa. Refresh the lists of users and admins.</td>
</tr>
</tbody>
</table>
Alternative:
- **Error 2**: User or admin doesn't exists. Show error.

**Delete user**

**Actors**: Admin user.

**Summary**: delete a user and all user's data and charts.

**Sequence of events**:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The admin user began this use case when selects the option delete user in the list of users and admins.</td>
<td>2. Ask to the admin if are sure to delete the user or admin.</td>
</tr>
<tr>
<td>3. User selects yes or no.</td>
<td>4. If the user selects yes delete all user information, data and charts belongs to the user and refresh the lists of users or admins. Else, user selects no, redirects to profile page.</td>
</tr>
</tbody>
</table>

Alternative:
- **Error 4**: User doesn't exist. Show error.

**List all user's data**

**Actors**: Admin user.

**Summary**: list all user's data that exist on the system.

**Sequence of events**:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The admin user began this use case when selects the option admin user's data.</td>
<td>2. Return a list of all user's data on the system.</td>
</tr>
</tbody>
</table>

Alternative:
- **Error 2**: Doesn't exist any user's data. Inform that.
Graphic interface:

List all user's charts

Actors: Admin user.
Summary: list all user's charts that exist on the system.
Sequence of events:

<table>
<thead>
<tr>
<th>Actor's actions</th>
<th>System response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The admin user began this use case when selects the option admin user's charts.</td>
<td>2. Return a list of all user's charts on the system.</td>
</tr>
</tbody>
</table>

Alternative:
- Error 2: Doesn't exist any user's charts. Inform that.

Graphic interface:

admin all user's data

<table>
<thead>
<tr>
<th>id</th>
<th>Username</th>
<th>Name</th>
<th>Description</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>marc</td>
<td>Data Example</td>
<td>This data contains an example...</td>
<td>Show, delete</td>
</tr>
<tr>
<td>15</td>
<td>marc</td>
<td>Data Example</td>
<td>This data contains an example...</td>
<td>Show, delete</td>
</tr>
<tr>
<td>10</td>
<td>marc</td>
<td>DCI</td>
<td>Digital Opportunity Index</td>
<td>Show, delete</td>
</tr>
<tr>
<td>11</td>
<td>marc</td>
<td>New data</td>
<td>This data contains...</td>
<td>Show, delete</td>
</tr>
<tr>
<td>12</td>
<td>marc</td>
<td>New data</td>
<td>This data contains...</td>
<td>Show, delete</td>
</tr>
<tr>
<td>6</td>
<td>martin</td>
<td>nou</td>
<td>noves daudes</td>
<td>Show, delete</td>
</tr>
<tr>
<td>5</td>
<td>martin</td>
<td>prova XML</td>
<td>mira q gual!</td>
<td>Show, delete</td>
</tr>
</tbody>
</table>
Physical database
users

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>NOT NULL, AUTOINCREMENT</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>username</td>
<td>varchar(60)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>password</td>
<td>varchar(60)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(60)</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>surname</td>
<td>varchar(60)</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mail</td>
<td>varchar(64)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role</td>
<td>tinyint</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>created</td>
<td>timestamp</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>last_login</td>
<td>timestamp</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>last_modif</td>
<td>timestamp</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table description:
This table contains the information about all users on the system.

Descriptions of the fields:
- **id**: code that identifies a user.
- **username**: name that the user uses to enter to the system.
- **password**: encrypted text that the user needs to type to enter to enter to the system, encrypted with SHA1 algorithm.
- **name**: name of the user.
- **surname**: surname of the user.
- **mail**: email address of the user.
- **role**: means the role of the user, 1 for registered normal users and 2 for admin users.
- **created**: date of the creation of the user.
- **last_login**: date of the last login of the user.
- **last_modif**: date of the last modification of the user profile.

users_data

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>NOT NULL, AUTOINCREMENT</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>user_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td></td>
<td>users (id)</td>
</tr>
<tr>
<td>name</td>
<td>varchar(30)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>varchar(60)</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>data</td>
<td>integer</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table description:
This table contains all the data imported by the users.

Descriptions of the fields:
- **id**: code that identifies a user’s data.
- **user_id**: id that identifies the user owner of the data.
- **name**: name for the data.
- **description**: text that describes the content of the data.
- **data**: the data imported by the user, kept in XML format.
users_charts

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>NOT NULL, AUTOINCREMENT</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>data_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td></td>
<td>users_data (id)</td>
</tr>
<tr>
<td>title</td>
<td>varchar(30)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>varchar(60)</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>varchar(10)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table description:
This table contains all the charts created by the users using their own data.

Descriptions of the fields:
- **id**: code that identifies a user's chart.
- **data_id**: id that identifies the data that the chart uses.
- **title**: title for the chart.
- **description**: text that describes the chart.
- **type**: type of chart: "bar", "pie", "area" or "world".

countries_world_map

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>NOT NULL, AUTOINCREMENT</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(30)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mc_name</td>
<td>varchar(2)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table description:
This table contains all the countries that can be used in the flash world map.

Descriptions of the fields:
- **id**: code that identifies a country of the world.
- **name**: name of the country.
- **mc_name**: 2 letters that identifies the country on the flash world graph.

GISW_countries

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>NOT NULL, AUTOINCREMENT</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>varchar(30)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mc_name</td>
<td>varchar(2)</td>
<td>NOT NULL</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table description:
This table contains all the countries that the system has data about the indices analyzed in the report of GISW.

Descriptions of the fields:
- **id**: code that identifies the country.
- **name**: name of the country.
- **mc_name**: 2 letters that identifies the country on the flash world graph.
GISW.DOI

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>country_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>year</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>opportunity</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>infrastructure</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>utilisation</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DOI</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rank</td>
<td>integer</td>
<td>-</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table description:**
This table contains data from the countries by the Digital Opportunity Index (DOI).

**Descriptions of the fields:**
- **country_id**: id that identifies the country.
- **year**: year of the data for the country and the index.
- **opportunity**: value of the cluster Opportunity of the DOI, ranges between 0 and 1.
- **infrastructure**: value of the cluster Infrastructure of the DOI, ranges between 0 and 1.
- **utilisation**: value of the cluster Utilisation of the DOI, ranges between 0 and 1.
- **DOI**: value of the index for a country and year, ranges between 0 and 1.
- **rank**: position in the ranking of all countries analyzed in the DOI.

GISW.KEI

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>country_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>year</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>economic_inc_reg</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>innovation</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>education</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICT</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEI</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rank</td>
<td>integer</td>
<td>-</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table description:**
This table contains data from the countries by the Knowledge Economy Index (KEI).

**Descriptions of the fields:**
- **country_id**: id that identifies the country.
- **year**: year of the data for the country and the index.
- **economic_inc_reg**: value of the “Economic incentive and institutional regime” of the KEI, on a scale of 0 to 10.
- **innovation**: value of the "The innovation system" of the KEI, on a scale of 0 to 10.
- **education**: value of the "Education and human resources" of the KEI, on a scale of 0 to 10. Information and communication technology
- **ICT**: value of the "Information and communication technology" of the KEI, on a scale of 0 to 10.
- **KEI**: value of the index for a country and year, on a scale of 0 to 10.
- **rank**: position in the ranking of all countries analyzed in the KEI.

### GISW_NRI

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>country_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>year</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>NRI</td>
<td>float</td>
<td></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>rank</td>
<td>integer</td>
<td></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>rank2</td>
<td>integer</td>
<td></td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

**Table description:**
This table contains data from the countries by the Networked Readiness Index (NRI).

**Descriptions of the fields:**
- **country_id**: id that identifies the country.
- **year**: year of the data for the country and the index.
- **NRI**: value of the index for a country and year, on a scale of 1 to 7.
- **rank**: position in the ranking of all countries analyzed in the NRI on the year.
- **rank2**: position in the ranking of all countries analyzed in the NRI on the next year.

### GISW_IICTD

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>country_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>year</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>acces</td>
<td>float</td>
<td></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>connectivity</td>
<td>float</td>
<td></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>ICT_Diffusion</td>
<td>float</td>
<td></td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>rank</td>
<td>integer</td>
<td></td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

**Table description:**
This table contains data from the countries by the Index of ICT Diffusion (IICTD).

**Descriptions of the fields:**
- **country_id**: id that identifies the country.
- **year**: year of the data for the country and the index.
- **acces**: value of the block "Acces" of the IICTD.
- **connectivity**: value of the block "Connectivity" of the IICTD.
- **ICT_Diffusion**: value of the index for a country and year.
- **rank**: position in the ranking of all countries analyzed in the IICTD.

### GISW_EGRI

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>country_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>year</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>web_measure</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>human_capital</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>infrastructure</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E_Readiness</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rank</td>
<td>integer</td>
<td>-</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table description:**
This table contains data from the countries by the E-Government Readiness Index (EGRI).

**Descriptions of the fields:**
- **country_id**: id that identifies the country.
- **year**: year of the data for the country and the index.
- **web_measure**: value of the “Web Measure Index”.
- **human_capital**: value of the “Human Capital Index”.
- **infrastructure**: value of the “Telecommunication Infrastructure Index”.
- **E_Readiness**: value of the index for a country and year.
- **rank**: position in the ranking of all countries analyzed in the EGRI.

### GISW_ERI

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Restrictions</th>
<th>PK</th>
<th>Foreign Key</th>
</tr>
</thead>
<tbody>
<tr>
<td>country_id</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>year</td>
<td>integer</td>
<td>NOT NULL</td>
<td>*</td>
<td>GISW_countries (id)</td>
</tr>
<tr>
<td>connectivity</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>business_env</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>consumer_bus_ad</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>option</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>legal_policy_env</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>social_cultural_env</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e_services</td>
<td>float</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>score</td>
<td>integer</td>
<td>-</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table description:**
This table contains data from the countries by the E-Readiness Index (ERI).

**Descriptions of the fields:**
- **country_id**: id that identifies the country.
- **year**: year of the data for the country and the index.
- **connectivity**: value of the "Connectivity and technology infrastructure" of the ERI.
- **business_env**: value of the "Business environment" of the ERI.
- **consumer_bus_adoption**: value of the "Consumer and business adoption" of the ERI.
- **legal_policy_env**: value of the "Legal and policy environment" of the ERI.
- **social_cultural_env**: value of the "Social and cultural environment" of the ERI.
- **e_services**: value of the "Supporting e-services" of the ERI.
- **score**: value of the index for a country and year.
5. Implementation

As has been said in the initial studies I have installed a web server Apache 2, with PHP 5.2.3 and database MySQL 5.0.41.

Structure of application
The structure of the directories of the web application is the next:

Below is a description of each of these folders and files:

- **admin**: contains the pages that can only see the admin user.
- **gisw**: contains all the pages to see development data from GISW.
- **images**: contains the set of images that are used in the design of the application.
- **includes**: contains all the pages that are used by other pages or requested by ajax.
- **index.php**: is the home page.
- **lib**: contains all the libraries used on the application
- **login.php**: is the login page.
- **notAcces.php**: is the page showed when one user tries to enter to a page that don't have acces.
- **register.php**: is the page to register as a new user.
- **uploads**: is the folder to keep temporally the data files that users import to the application.
- **user**: contains the pages that can only see a registered user.

Admin folder

- **index.php**: is the initial page when admin is logged.
- **listCharts.php**: this page lists all the user’s charts of the application.
- **listData.php**: this page lists all the user’s data of the application.
- **listUsers.php**: this page lists all the user’s and admin users of the application.
GISW folder

- **country.php**: this page shows the charts and data of all indices available for a country.
- **index.php**: this is the index page of development data.
- **indices**: this folder contains all the pages for each index.
  - **doi.php**: this page shows the charts and data for Digital Opportunity Index.
  - **egri.php**: this page shows the charts and data for E-Government Readiness Index.
  - **eri.php**: this page shows the charts and data for E-Readiness Index.
  - **iictd.php**: this page shows the charts and data for Index of ICT Diffusion.
  - **kei.php**: this page shows the charts and data for Knowledge Economy Index.
  - **nri.php**: this page shows the charts and data for Networked Readiness Index.
- **map.php**: this page shows the world map graph for the data of available years, indices and parameters of the index.

Includes folder

All the pages on this folder are included on other pages or called via ajax.

- **admin**: contains pages included in admin pages or admin pages that calls it via ajax.
- **config.php**: this page contains variables for configuration of the application.
- **control.php**: this page is used to control the access on the pages.
- **css**: contains the Cascading Style Sheets files of the application.
- **db.php**: contains a php class with function to manipulate the database.
- **footers.php**: this is the footer page of the application.
- **gisw**: contains the pages that creates charts and graphs included in the gisw pages.
- **headers.php**: this is the header page of the application, contains the meta tags, includes to javascript files and css files.
- **import**: contains the files that reads the import data and converts it, and some file examples of Comma-Separated Values to import.
- **js**: contains the javascript files used in the application.
- **menu.php**: this is the left menu of any page of the application, it also can include the menus of user and admin.
- **up.php**: this is the up part of any page of the application.
- **user**: contains pages included in user pages or user pages that calls it via ajax.

**Lib folder**

- **ammap**: contains all the necessary files for the world map flash of amMap library.
- **php-ofc-library**: contains all the necessary files for the Open Flash Chart library.
- **prototype**: contains the javascript library of Prototype 1.6.0.
- **scriptaculous**: contains the javascript files of the script.aculo.us 1.8.0 library.
- **Validation**: contains the javascript library of Validation 1.5.4.1.

**User folder**

- **changePassword.php**: in this page the user can change the password.
- **chart**: contains the pages for the charts of the user.
  - **create.php**: this page is for create a new chart.
  - **list.php**: this page lists all the charts of the user.
  - **show.php**: this page shows a chart from the user.
- **data**: contains the pages relatedion with the data of the user.
  - **list.php**: this page lists all the data of the user.
  - **show.php**: this page shows a data from the user.
  - **upload.php**: this page is the first step for import a new data, upload a file.
  - **upload_done.php**: this page is the second step for import a new data, save the uploaded data.
- **index.php**: is the initial page when user is logged.
- **profile.php**: this page is for the profile of the user.
- **profile_show.php**: this page contains the profile of the user.
- **profile_edit.php**: this page is for edit the profile of the user.
Importing data from GISW

The part of importing the development data from the GISW report has been not a difficult task but yes a hard task. I was in contact to one of the GISW report for obtain a way to facilitate the importation of the data, but finally the only solution I had was copy directly from the website www.globalishwatch.org as I will explain.

1. Select and copy the data directly from the tables of the website.

2. Paste data on an editor.
3. Edit: using replaces spaces/tabs to convert it to useful array.

```php
$s001_index = array('Argentina', 0.96, 0.97, 0.39, 0.36, 0.15, 0.23, 0.47, 0.51, 51, 54),
array('Brazil'), 0.87, 0.52, 0.24, 0.27, 0.16, 0.24, 0.42, 0.48, 71, 65),
array('Colombia'), 0.88, 0.89, 0.19, 0.25, 0.08, 0.19, 0.38, 0.45, 88, 89),
array('Congo'), 0.89, 0.89, 0.16, 0.21, 0.02, 0.08, 0.36, 0.49, 100, 97),
array('Mexico'), 0.93, 0.04, 0.22, 0.39, 0.13, 0.25, 0.43, 0.47, 66, 69),
array('Peru'), 0.86, 0.82, 0.19, 0.27, 0.39, 0.48, 85, 96),
array('Bosnia and Herzegovina'), 0.93, 0.95, 0.27, 0.36, 0.85, 0.14, 0.42, 0.48, 75, 64),
array('Bulgaria'), 0.96, 0.97, 0.34, 0.49, 0.22, 0.51, 0.54, 46, 47),
array('Croatia'), 0.97, 0.08, 0.44, 0.47, 0.18, 0.14, 0.53, 45, 43),
array('Hungary'), 0.93, 0.96, 0.26, 0.31, 0.20, 0.30, 0.46, 0.52, 53, 58),
array('India'), 0.99, 0.99, 0.94, 0.04, 0.05, 0.29, 0.31, 119, 124),
array('Pakistan'), 0.73, 0.76, 0.05, 0.07, 0.08, 0.03, 0.26, 0.29, 127, 127),
array('Philippines'), 0.93, 0.93, 0.13, 0.15, 0.03, 0.04, 0.36, 0.35, 94, 100),
array('Dem. Rep. of the Congo (DR)'), 0.46, 0.22, 0.05, 0.02, 0.00, 0.16, 0.08, 150, 176),
array('Estonia'), 0.94, 0.96, 0.17, 0.22, 0.02, 0.04, 0.38, 0.41, 90, 91),
```

4. Create a php script that uses this array to keep this data to the database.

```php
foreach ($s001_index as $element)
{
    $country = $element[0];
    $sapp = $element[1];
    $inf = $element[2];
    $util = $element[3];
    $d01 = $element[4];
    $rank = $element[5];
    $sql = $db->getSQLCountry($country);

    $insert = 'INSERT INTO GISW_D01 (country_id, year, opportunity, infrastructure, utilisation, D01, rank) VALUES ('.
        $sql, 2005, $sapp, $inf, $util, $d01, $rank);'

    $db->consult($insert);
    if ($id->idConsult)
    {
        echo $db->error("error");
    }
}
```

With this hard task I was importing all the data of the indices of GISW report to keep it to the MySQL Database.
Charts and world map graphs

For the creation of the charts and world map graphs I have used 2 different libraries analyzed at the initial studies. For type of the charts bar, pie and area I have used Open Flash Chart, and for the world map graphs I have used the amMap. I'll explain how I have used these libraries.

Charts

Open Flash Chart is the library that I use to generate this charts, the library is located on the folder lib/php-ofc-library/, inside the web folder.

The library contains 3 basic files:
- open-flash-chart.swf: vector graphic file.
- open_flash_chart_object.php: contains the functions that creates the HTML code of the flash chart.
- open-flash-chart.php: contains all the functions to create the charts, add data and titles, and customize the size and many things.

I will show an simple example using this library:

1. Creates a page where we include the open_flash_chart_object.php and calls to the function open_flash_chart_object().

```php
<html>
<head></head>
<body>
<?php
include_once 'includes/config.php';
require_once 'lib/php-ofc-library/open_flash_chart_object.php';
?>
<body>
</html>
```

The function open_flash_chart_object() will create the HTML code for the flash object. For the data and construction of the chart object we have create a new php file, param_3d_bar.php on this example.

68
2. Creates the php file that builds the chart object and put the data.

```php
<?php
require_once( "/../db.php" );
require_once( "/../lib/php-efc-library/open-flash-chart.php" );
$db = new SQLite();
$db->connect();
$indice = $_GET["indice"]; $param = $_GET["param"];
$title = "$param - $indice";
$sg = new gphChart();
$sg->setTitle($title, 
/font-size:20px; color: #000; margin: 5px; padding:5px; padding-left: 20px; padding-right: 20px;'/);
$colors = array( "#633F45", "#33344D", "#e4b544", "#33344D" );
$sg->getBarForIndex($indice);
$i = 0;
while ($row = mysql_fetch_array($db->CONSULT)) {
    $year = $row["year"]; $value = $row["value"]; 
    $sg->addSeries($year, $value);
    $i++;
}
$sg->draw();
```

The flash object request this page with the 2 GET parameters, indice and param. At first, we include the file of the library open-flash-chart.php, receives the two parameters, creates a new graph object, reads the data from the database, customize some features of the chart and render all, the output of this page will go to the flash.
World map graphs

amMap is the library I use to create the flash world map graphs, the library is located on the folder *lib/ammap/*, inside the web folder.

The important files that the library contains basically are:
- **ammap.swf**: principal vector graphic file.
- **export.php**: used to export the flash to image.
- **maps**: folder that contains all the flash maps available.

I will show a simple example using it:

1. Creates a page where we include the swfobject ammap.swf.

```html
<html>
<head></head>
<?php
include_once 'includes/config.php';
include_once 'lib/php-ofc-library/open_flash_chart_object.php';

<body>
 <div id="flashcontent">
   <strong>You need to upgrade your Flash Player</strong>
 </div>
 <script type="text/javascript">
   // <![CDATA[
   var so = new SWFObject("<?php ??>/lib/ammap/ammap.swf", "ammap", "850", "500", "8", "#FFFFFF");
   so.addVariable("path", '<?php ??>/lib/ammap/');
   so.addVariable("map_id", "ammap");
   escape("<?php ??>/includes/gisw/map_data.php?indice=001&param=rank&year=2006");
   so.addVariable("settings_file");
   so.addVariable("loading_data", "Loading development data");
   so.addVariable("loading_settings", "Loading settings");
   so.write("flashcontent");
   // ]]>
 </script>
</body>
</html>
```

The flash needs two files: one with the data and other with the settings of the graph, the two files are XML.

Settings file contains characteristics like width, height, font, text size, colors, background, enable zoom...

On the next step we are going to see an example of data file.
2. Create the data file XML that contains which map file is used, areas with title, mc_name and value, and some labels.

```xml
<map map_file="maps/world.swf" url="#movie1" zoom="100%" zoom_x="6.58" zoom_y="96%">
  <area title="Argentina" mc_name="AR" value="0.96"></area>
  <area title="Brazil" mc_name="BR" value="0.87"></area>
  <area title="Colombia" mc_name="CO" value="0.88"></area>
  <area title="Ecuador" mc_name="EC" value="0.89"></area>
  <area title="Mexico" mc_name="MX" value="0.93"></area>
  <area title="Peru" mc_name="PE" value="0.86"></area>
  <area title="Bosnia and Herzegovina" mc_name="BA" value="0.93"></area>
  <area title="Bulgaria" mc_name="BG" value="0.96"></area>
  <area title="Croatia" mc_name="HR" value="0.97"></area>
  <area title="Romania" mc_name="RO" value="0.91"></area>
  <area title="Spain" mc_name="ES" value="0.99"></area>
  <area title="Bangladesh" mc_name="BD" value="0.60"></area>
  <area title="India" mc_name="IN" value="0.88"></area>
  <area title="Pakistan" mc_name="PK" value="0.73"></area>
  <area title="Philippines" mc_name="PH" value="0.93"></area>
  <area title="Asia" mc_name="" value="0.81"></area>
  <area title="Egypt" mc_name="EG" value="0.94"></area>
  <area title="Ethiopia" mc_name="ET" value="0.26"></area>
  <area title="Kenya" mc_name="KE" value="0.34"></area>
  <area title="Nigeria" mc_name="NG" value="0.41"></area>
  <area title="South Africa" mc_name="ZA" value="0.98"></area>
  <area title="Uganda" mc_name="UG" value="0.45"></area>
</area>
<movies>
  <movie old="movie1" text_box_x="38" text_box_y="366" text_box_width="190" text_box_height="120"
    title="Map chart example">
    <description></description>
  </movie>
</movies>
<labels>
  <label x="0" y="50" width="100%" align="center" text_size="16" color="#000000">
    <text><![CDATA[<b>2005]]></text>
    <description></description>
  </label>
</labels>
</map>
```
Testing

The testing phase is very important in order to obtain a properly functioning system software. Through such tests must be conducted to determine whether the implementation of the system satisfies the functionality, the restrictions and requirements prescribed by the phases of specification and analysis requirements. This work is not easy because we must have a good knowledge of the system requirements developed.

The tests can not be limited to the last stage of the development of the system after the implementation as the sole detection of faults at this point could seriously compromise the project's success.

For this reason, have been conducting tests as they are implemented, trying to see functioning as anticipated. If the result was not as expected, has been consulted specification and system design, implementation has been modified by correcting the error and has been re-launch testing. Thus, the task of testing the system has been spreading during the implementation stage.

As for the topics covered in the tests, have attempted to identify missing or incorrect functions, errors in the interfaces, errors in accessing data structures, performance errors and errors in the initialization and termination of execution. In addition, it has continued the execution of all independent courses of loops and conditional branches using all structures. It has tested both himself from the application developer whose expertise in the same, would have found errors sophisticated as well as outsiders. Thanks to the latter has been improved usability.

Nevertheless, it should be pointed out that the tests could not assure the total absence of errors, can only detect the presence. A test is nothing more than the execution of the system with the intention of discovering an error.
6. Results & Conclusions

Goals

The principal goal of this project has been achieved, this new software has been build from pieces of free software, like a mashup that combines different tools into one, and is able to represent manifold indicators of development of the information’s society in a cute charts and graphs.

The visualization of the different indicators of development data in ICT can be very helpful for understanding the GISW reports.

In addition, this web application offers the possibility to create your own charts and graphs importing your own data, and easily embed this charts and graphs to other sites.

Future expansions

There are many expansions that can be done for this application, some of them are to improve the features that right now the application have and some other to add new functionalities to the application.

More customization

Could be very great to have the possibility to customize the charts and graphs that one user creates.
For the charts bar, pie and area would be a good option to have the opportunity to choose the colors and size of the chart, show/hide some labels and parts of the chart.
And for the world map graphs would be a very good option to have the opportunity to choose the colors and size of the graph, and also the possibility to add links on click over the countries.

More importing options

The user can import right now from CSV files, would be interesting to add new options for upload data. Some possibilities can be with HTML table from a file or URL, Excel format files, or from other existing MySQL database or other type of database from other server.

New charts

Add new kind of charts, like Timeplot or other types.
Timeplot, analyzed in the initial studies, is a lineal chart with events very interesting but finally for lack of time could not be implemented.
Personal valuation

The realization of this project has brought me much knowledge because I haven’t only touched an item of information technology but several. With this project I have learned to search, to analyze, to choose; in concrete to search graphic tools, to analyze them, to choose the best solution that fits on the project. It also has helped me gain more experience in terms of programming in PHP to carry out the implementation of the project.

I am very proud of the result of the project. I think it can be very useful for GISW and for more other people that will can use it. It have served me to stay a little more closer to see the importance of ICT development in the countries and the importance of the graphical visualization of the data for take conclusions about development data.

This project has motivated me a lot that will be possibly used in the near future. I still have more motivation because the project is related to the development of ICT, in addition, I have always wanted to do a cooperative project and now with the final project of my degree is an approximation. It have also served me for be more interested by the cooperation world.
7. References

   URL: <http://www.globaliswatch.org/>

   URL: <http://www.apc.org>

[3] ITeM, Institut del Tercer Món (The Third World Institute). (online)
   URL: <http://www.item.org.uy>

[4] Cooperative Association for Internet Data Analysis (CAIDA). (online)
   URL: <http://www.caida.org>

   URL: <http://www.guiadelmundo.org.uy>

   URL: <http://www.gapminder.org>

[7] Maplecroft. (online)
   URL: <www.maplecroft.net>

[8] Global Education Project. (online)
   URL: <http://www.theglobaleducationproject.org/earth/index.php>

   URL: <http://www.mdgmonitor.org>

[10] Gantt Project. (online)
    URL: <http://ganttproject.biz/>

    URL: <http://www.apache.org/>

[12] PHP. (online)
    URL: <http://www.php.net/>

    URL: <http://www.mysql.com/>

[14] Open Flash Chart. (online)
    URL: <http://teethgrinder.co.uk/open-flash-chart/>

[15] Fusion Charts Free. (online)
    URL: <http://www.fusioncharts.com/free/>

[16] XML/SWF Charts. (online)
    URL: <http://www.maani.us/xml_charts/>

[17] PHP/SWF Charts. (online)
    URL: <http://www.maani.us/charts/index.php>

[18] amCharts. (online)
    URL: <http://www.amcharts.com/>

[19] amMap. (online)
    URL: <http://www.ammap.com/>


[21] Flash Charts Pro. (online)
    URL: <http://www.web-site-scripts.com/flash-charts/>

[22] Swiff Chart Generator. (online)
    URL: <http://www.globlfx.com/products/swfchart/>

[23] AnyChart Flash. (online)
    URL: <http://www.anychart.com/products/anychart/overview/>
[24] Libchart. (online)
URL: <http://naku.dohcrew.com/libchart/pages/introduction/>
[25] Image_Graph. (online)
URL: <http://pear.veggerby.dk/>
[26] JpGraph. (online)
URL <http://www.aditus.nu/jpgraph/>
[27] .netCharting. (online)
URL: <http://www.dotnetcharting.com/>
[28] Emprise Javascript Charts. (online)
URL: <http://www.ejschart.com/>
[29] Plotkit. (online)
URL: <http://www.liquidx.net/plotkit/>
[30] Plotr. (online)
URL: <http://solutoire.com/plotr/>
[31] jQuery Chart Plugin. (online)
URL: <http://www.reach1to1.com/sandbox/jquery/jqchart/>
[32] Dojo Charting Engine. (online)
[33] Timeplot. (online)
URL: <http://simile.mit.edu/timeplot/>
[34] Script.aculo.us. (online)
URL: <http://script.aculo.us/>
[35] Prototype JavaScript Framework. (online)
URL: <http://www.prototypejs.org/>
[36] Validation javascript library. (online)
URL: <http://www.tetlaw.id.au/view/javascript/really-easy-field-validation>