

A Configurable Web System for Generating E-Commerce Web Sites

Carlos R. Jaimez-González, René Ortiz-González

Departamento de Tecnologías de la Información,
Universidad Autónoma Metropolitana, Unidad Cuajimalpa,
Av. Vasco de Quiroga 4871, Col. Santa Fe Cuajimalpa, C.P. 05300, México, D.F.
{cjaimez,209363198}@correo.cua.uam.mx

Abstract. In order to expand their opportunities of selling products and services, conventional stores have been creating their own e-commerce web sites, where they can carry out electronic transactions from any place at any time. Store owners and their clients benefit from e-commerce web sites, because they are permanently online and provide catalogues of products and services with images, descriptions, specifications, prices, comments from clients, recommendations, among other features; so clients have enough information to make a decision on the products and services they will buy. There are some tools available on the web that generate this type of web sites, but they do not provide the functionality of displaying the web site on mobile devices automatically. This paper presents a configurable web system for generating and administering e-commerce web sites, which are automatically displayed on mobile devices, using the responsive web design pattern.

Keywords. Configurable Web System; E-Commerce Web Site; Responsive Web Design; Web-Based Application.

1 Introduction

In order to expand their opportunities of selling products and services, conventional stores have been creating their own e-commerce web sites, where they can carry out electronic transactions from any place at any time. Store owners and their clients benefit from e-commerce web sites, because they are permanently online and provide catalogues of products and services with images, descriptions, specifications, prices, comments from clients, recommendations, among other features; so clients have enough information to make a decision on the products and services they will buy. A fundamental feature of an e-commerce web site is the shopping cart, in which a client puts the products wanted, so that when the selection is finished, clients proceed to checkout, where they pay for the products through credit card, debit card, paypal, or other mechanism. Once the products have been paid for, in most of the cases the seller sends them to the address provided by clients.

Some of the main features and components of e-commerce web sites are the following: user registration for clients; visualization of catalogues of products and ser-

vices with several levels of categories; visualization of information of each product or service, such as its name, description or specification, price, and others; product finder; recommendation of products; shopping cart for storing products; selection of payment and online payment; selection of delivery for products; visualization of purchasing history; modules for administration of catalogues, such as categories, products, payments, deliveries, sales history, sales statistics, among others.

The use of devices such as smartphones and tablets has increased the development of mobile applications, which allow accessing different web sites. The visualization of a web site in a mobile device is different to the one in a laptop or desktop computer; this is mainly due to the difference in the resolution of mobile devices. In order to have an effective and correct visualization on any device or computer, it is necessary in most of the cases to modify the web site, so that it can be displayed in a satisfactory way according to the device. Although there are some tools that generate e-commerce web sites, such as *osCommerce* [1], *Magento* [2], *Prestashop* [3], *Opencart* [4], none of them provide the functionality of having the resulting web site visualized automatically on mobile devices. This paper presents a configurable web system for generating and administering e-commerce web sites, through the selection of some configuration options. Additionally, the e-commerce web sites created will be displayed automatically on mobile devices, without requiring the user to modify the web site.

The rest of the paper is organized as follows. Section 2 presents the main components of an e-commerce web site and some related work. The architecture of the web system is introduced in section 3. Section 4 describes the implementation of the web system, including a site map, some web user interfaces, and the technologies used to implement it. Finally, conclusions and future work are provided in section 5.

2 Related Work

This section provides a description of the main components of an e-commerce web site, and reviews some existing tools that generate e-commerce web sites.

2.1 E-Commerce Web Site Components

There are three components of an e-commerce web site: online store or front-end; administration panel or back-end; and the data persistence component. The main features of each component are provided in the following paragraphs.

Online store. The client interacts with this component, in which it is possible to visualize the offered products, carry out transactions, and there is communication with the store. The main features and modules that compose it are the following: products catalogue, product view, categories of products, promotions, recommendations, product search, user registration, shopping cart, payments, deliveries, among others.

Administration panel. This component is used by the system administrator to create the online store; it has options to add, update and delete catalogues in the system, such as products, categories, clients, payments, deliveries, promotions, etc. It can also search historical data, as well as statistics of the store, and managing the entire

system. The main features and modules that compose it are the following: management of products, categories, payments, deliveries, statistics, historical data, invoices, promotions, users, communication between users, among others.

Data persistence. This component is in charge of storing the information of the system; it guarantees its integrity and persistence. Most of the e-commerce systems examined are based on relational models of persistence, among which are the following database management systems: MySQL, PostgreSQL, Oracle, SQL Server.

2.2 Existing Tools

osCommerce [1]. According to its documentation, it is an online program for e-commerce and administration, which was developed in PHP by Harold Ponce de Leon. In order to create an online store with *osCommerce*, it is required a MySQL database, and an Apache web server. *osCommerce* is composed of two parts: the products catalogue; and the administration module to manage the online store; it also allows updating products, categories, languages, new offers, currency types, orders, clients, payment and delivery types, etc. An online store created with *osCommerce* is divided in three panels: the left panel shows categories of products, selection of manufacturers of products, product finder with advanced search, new products, information for shipping, private notice, conditions of use, and a contact form; the central panel shows the catalogue of products with images, descriptions, and prices; and the right panel shows the shopping cart, the bestsellers, special products, product reviews, and conversion of currencies. At the top of the web site there are three buttons: one for accessing the shopping cart content, one for going to checkout, and one for the user account.

Magento [2]. It is a system for creating online stores, where its source code is available and free. It is possible to upgrade to commercial versions, and increase the functionality of the online store. The main features of *Magento* are the following: it can create traditional online stores with physical products; but it also allows creating online stores with downloadable material, such as music or ebooks. It allows multi-stores; it has a search system; it allows complete customization of the web site through an administration panel; it has a module for orders management; etc. An online store generated with *Magento* is divided in three panels: the left panel shows two random offers of products, and popular tags for searching products; the central panel shows some other offers of products with discounts, and some of the best seller products with their images, descriptions, and prices; and the right panel shows a module for comparing products, the shopping cart, an advertisement, and a survey. At the top of the web page, there are the following links for the user of the online store: my account, my wish list, my shopping cart, checkout, log in, and product finder.

Prestashop [3]. It is an open source solution that combines the PHP language, an Apache web server, and a MySQL database. It allows store owners to maintain their online stores without any license costs. *Prestashop* is flexible in terms of design and distribution of the web page. It uses a hook mechanism to put any module in any position on a web page, such as the shopping cart, currencies, product search, etc. The hooks are calls to components, which are positioned using JavaScript code. Among its main features are the following: multi-language support; it is possible to create cus-

tomized web pages, including images and other multimedia components; several payment modules; integrated clients module; definition of tangible and virtual (digital) products; use of bar codes, etc. An online store generated with *Prestashop* has a three-column layout, where the central column has the catalogue of the store, and a list of outstanding products. The left column contains a section with tags of products, categories, manufacturers, and general information, such as deliveries, legal notice, conditions of use and contact. The right column has the shopping cart at the top, and a section for new products. The user menu is displayed at the top of the site, with links for contact, site map, favorites, currency converter, user account, and product search.

Opencart [4]. It is an open source e-commerce system, which is based on the PHP programming language. *OpenCart* was created with the following features: it is a light version derived from *osCommerce*; it has good documentation for developers; interchangeable template system with easy development; it uses the Model-View-Controller (MVC) design pattern; it is modular; multiple languages; several ways of payments; comments on products; different ways of delivery; evaluation of products with star ratings; etc. An online store generated with *OpenCart* is simple and orientated to the Web 2.0. Its interface preserves the layout of three columns: on the left column the categories of products, and other information, such as about us, privacy policy, terms and conditions, contact us, and site map; the catalogue of products is located on the center, with images, descriptions, and prices for each product; the right column contains the shopping cart and the best sellers. The user menu in this system is in the form of tabs, and it has the main page, log in, account, shopping cart, checkout, product search by product, by category, and advanced search.

Other systems. There are several systems for creating e-commerce web sites. The most used systems were described and analyzed in the previous sections, but some other systems can be found in the comparative table of shopping cart software shown in [5]. However, there are many of them unavailable or with obsolete versions. Some of the systems still in use are the following: Zen-Cart [6], Go-eCommerce [7], Shoppizer [8], Konakart [9], and JadaSite [10]. The scripts described in this section have been created using Java Server Pages (JSP) mainly. These systems have not been used as much as the other systems analyzed, because they are complex Java applications; they are difficult of customization; they do not have a version for displaying the resulting e-commerce web sites on mobile devices; and given its architecture and implementation, it is difficult to incorporate such functionality.

3 Web System Architecture

The web system presented in this paper is modular, client-server, with a template system for its appearance; and it is based on the JSP technology, and uses a MySQL database. JSP and MySQL were chosen mainly because they are robust, portable, and widely used in the market. The system architecture is supported on the Model-View-Controller (MVC) design pattern, and the Responsive Web Design (RWD) pattern for adjusting its web interfaces to different devices. The system architecture is composed of five components: the core, the Java beans, the model, the actions and the views.

3.1 Components of the Web System

The Model Component is in charge of the data persistence; it maintains connections to the database; it carries out transactions; and executes all the necessary methods for maintaining the data in the database. This component is divided in three subcomponents: a model class with all the methods for accessing data; a web model class, which is in charge of obtaining data from the application context; and the init model class, which initializes all the necessary information before establishing connections.

The Action Components are in charge of the communication with the View Components. They are implemented as classes with some attributes that obtain requests from clients, pass requests to the controller, obtain context from the application, and provide the model of the application. Following the MVC design pattern, the Action Components work directly with the model and return a response to the controller.

The Bean Components allow creating specific objects for manipulating data from the database. There are bean components for users, products, categories, and for any other entity in the system; which have *set* and *get* methods for reading and setting their attributes; they also have a method to retrieve their *string* representation. The bean components obtain data as follows: views request user data to the controller; the controller searches and executes the corresponding action; the action returns a result set object; a method in the bean component converts the result set into the type of the bean, and returns it to the view; the view uses this object as any other bean. It should be noticed that the data is encapsulated in the bean component.

The View Components are the interface to the user of the system. These components are implemented with XHTML and JSP. The responses from the controller are actually calls to JSP; they obtain parameters that allow them displaying information. An special function of the system presented in this paper, is the ability of obtaining information from the model without the need of an action bean.

It should be noticed that the Action and Bean Components were incorporated because they allow delegating intermediary actions between controller and model.

3.2 Model-View-Controller (MVC)

The Model-View-Controller (MVC) design pattern is an ordered set of subsystems, which are composed of other systems that are below them, and provide the base of the implementation for those systems that are above. The objects in each layer are normally independent, and there is a client-server relationship among the bottom and the top layers. The system presented in this paper is based on the MVC design pattern, which is divided in three layers: the view, the controller and the model. The *View* is the presentation layer, which provides the graphic interface and it is in charge of the presentation of the data to the user, as well as to collect information from the user. The *Controller* is the business layer, which provides the functionality of the application; it is the layer where user and system interchange messages; it is also in charge of managing the requests from the user, and executing the necessary operations to solve such requests. The *Model* is the persistent layer; it stores all the information of the system; it also allows access to the information in a secure and controlled way; this

level is composed of the database and a class called model, which contains all the methods that provide access to the data. The MVC design pattern was chosen because it allows modifying the layers independently, without affecting the rest of the system.

3.3 Responsive Web Design (RWD)

The Responsive Web Design (RWD) is a recently created pattern; it has been adopted by web developers, because it tries to solve the problems that exist when creating web pages for different devices. Figure 1 shows a web page in different resolutions, where the elements of the page are distributed according to the resolution of the device where the web page is being displayed.

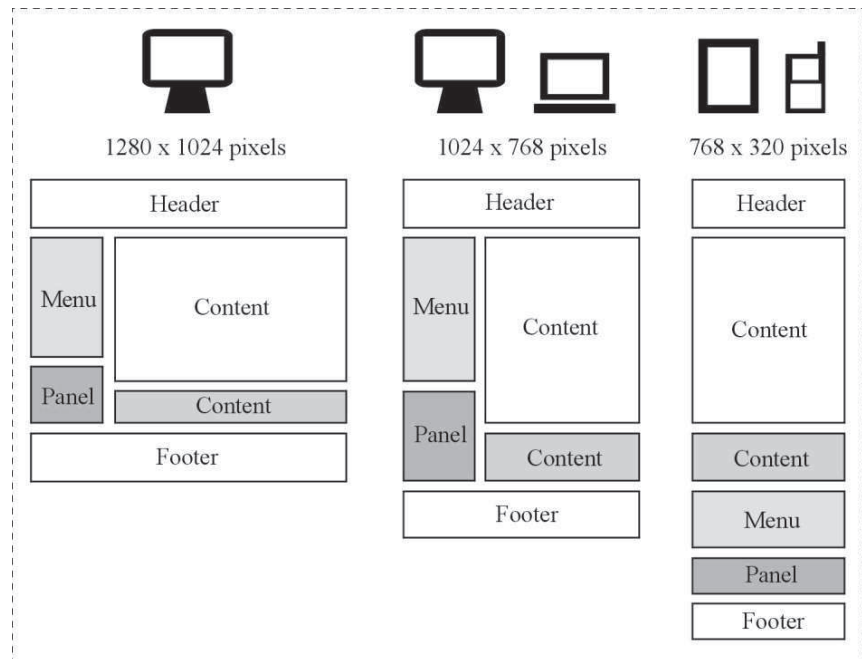


Fig. 1. Distribution of elements on the web page according to the resolution.

This RWD pattern is centered in a unique design that is adaptable to different devices and their resolutions, without developing specific implementations; it is mainly based on Media Queries (MQ), which are part of the last CSS 3 specification, and allow changing styles according to the resolution of the device where the web page is visualized. For example, if a web page is designed for a resolution of 1024 pixels, the font size would be 12 pixels; however, when changing the resolution, the font size would be the same size, making it difficult to read the web page. MQ can determine parts of the CSS code to appear, or change attributes of classes depending on the resolution. This way, there can be sections of CSS code for different resolutions.

4 Web System Implementation

This section provides details about the implementation of the system. It first provides a site map with all the web pages that compose the system; it shows the interface of some parts of the system; it also describes some tests on different devices; and finally it describes the technologies and tools used for developing the system.

4.1 Site Map

Figure 2 shows the site map of the web system, which is considered to be simple and organized, compared to those of the products analyzed. It considers three different users: administrators, registered and non-registered users of the e-commerce web site.

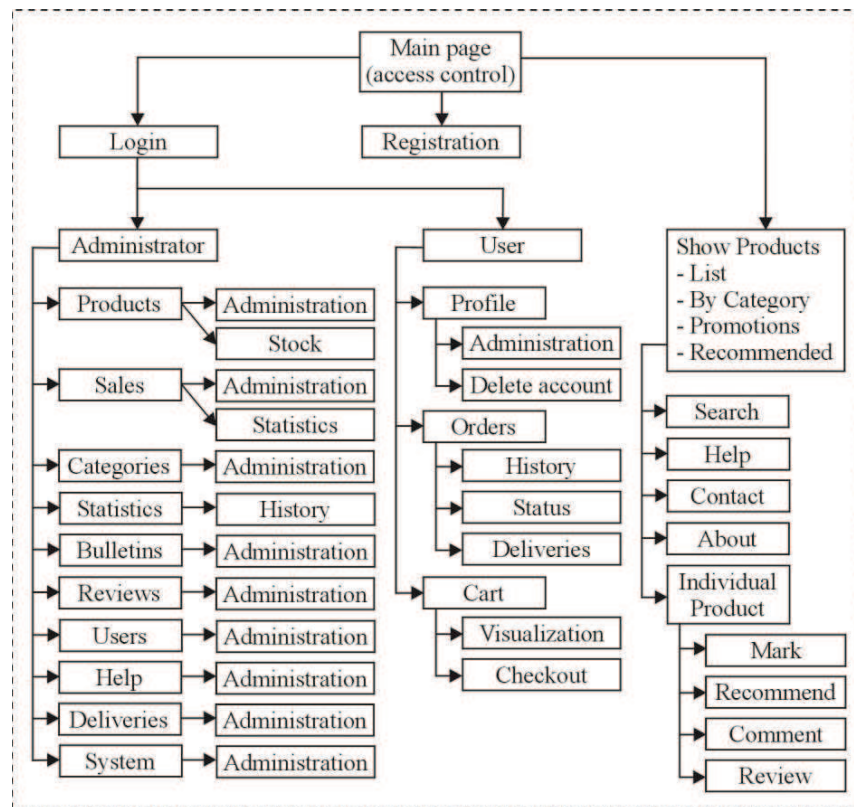


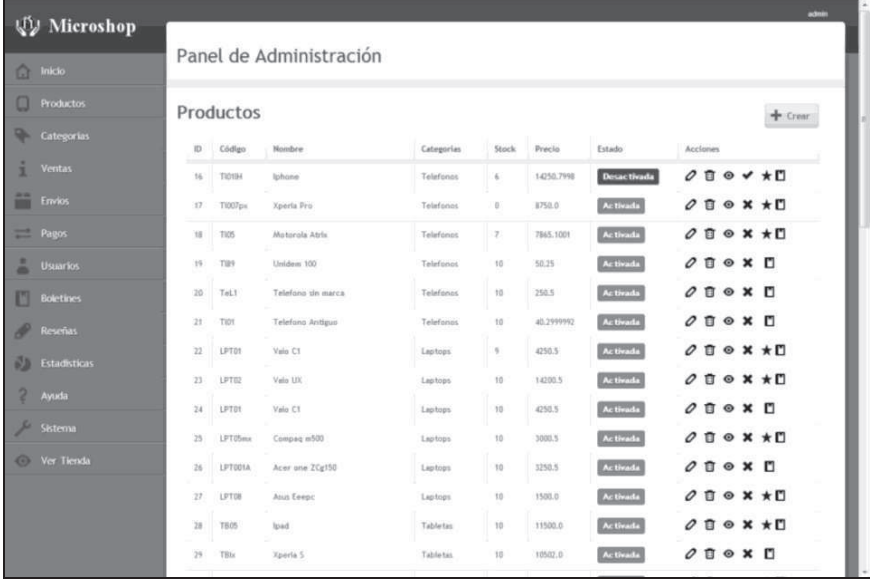
Fig. 2. Map of the web site for administrators, registered and non-registered users.

The left side of the site map illustrates the actions that an administrator can carry out on the system, such as administering products, sales, categories, statistics, bulletins, reviews, users, and deliveries; the center of the figure illustrates the actions that a

registered user can carry out on the system, such as editing its profile, modifying its orders, and its shopping cart; and the right side of the figure illustrates the actions that non-registered users, administrators, and registered users can carry out on the system, such as visualizing products, searching products, viewing promotions and recommendations, marking products, recommending products, commenting on products, reviewing products, looking for help, and contacting the web site administrator.

4.2 Web Interface

This subsection provides some screenshots of an e-commerce web site generated with the web system presented in this paper. The screenshots show a web page with a list of products, in three different resolutions, according to the device where the web site is being displayed. Figure 3 shows the web interface displayed on a desktop computer with a resolution of 1240 x 1024 pixels, which is the maximum; and Figure 4 shows two web interfaces, the first one displayed on a tablet with a resolution of 960 x 720 pixels, and the second one displayed on a smartphone with a resolution of 320 x 480 pixels. The web page is divided in two panels: the left panel shows the menus for all the categories of products, news, reviews, bulletins, among others; and the right panel visualizes the list of products, with their ids, names, category, stock, price, state, and some actions. The screenshots in Figure 4 show the same web page, but with reduced information and simplified menus, due to the resolution of the devices used. For example, in the web page shown on the smartphone, the left panel is hidden, but can be displayed through a button on the top of the page.



ID	Código	Nombre	Categorías	Stock	Precio	Estado	Acciones
16	TID184	Iphone	Telefonos	4	14250.7998	Desactivado	[Edit] [Delete] [Add] [Star]
17	TID07pe	Xperia Pro	Telefonos	0	8750.0	Act. Usado	[Edit] [Delete] [Add] [Star]
18	TID5	Motorola Atrix	Telefonos	7	7845.1001	Act. Usado	[Edit] [Delete] [Add] [Star]
19	TID1	Unides 100	Telefonos	10	50.25	Act. Usado	[Edit] [Delete] [Add] [Star]
20	Tel1	Telefono de marca	Telefonos	10	250.5	Act. Usado	[Edit] [Delete] [Add] [Star]
21	TID1	Telefono Antiguos	Telefonos	10	40.2999992	Act. Usado	[Edit] [Delete] [Add] [Star]
22	LPT01	Vale C1	Laptops	9	4250.5	Act. Usado	[Edit] [Delete] [Add] [Star]
23	LPT02	Vale UX	Laptops	10	14000.5	Act. Usado	[Edit] [Delete] [Add] [Star]
24	LPT01	Vale C1	Laptops	10	4250.5	Act. Usado	[Edit] [Delete] [Add] [Star]
25	LPT05mx	Compaq n500	Laptops	10	3000.5	Act. Usado	[Edit] [Delete] [Add] [Star]
26	LPT001A	Acer asie 20g150	Laptops	10	3250.5	Act. Usado	[Edit] [Delete] [Add] [Star]
27	LPT08	Asus EeePC	Laptops	10	1500.0	Act. Usado	[Edit] [Delete] [Add] [Star]
28	TID5	ipad	Tabletas	10	11500.0	Act. Usado	[Edit] [Delete] [Add] [Star]
29	TIDx	Xperia S	Tabletas	10	10500.0	Act. Usado	[Edit] [Delete] [Add] [Star]

Fig. 3. E-commerce web site displayed on a desktop computer (1240x1024).

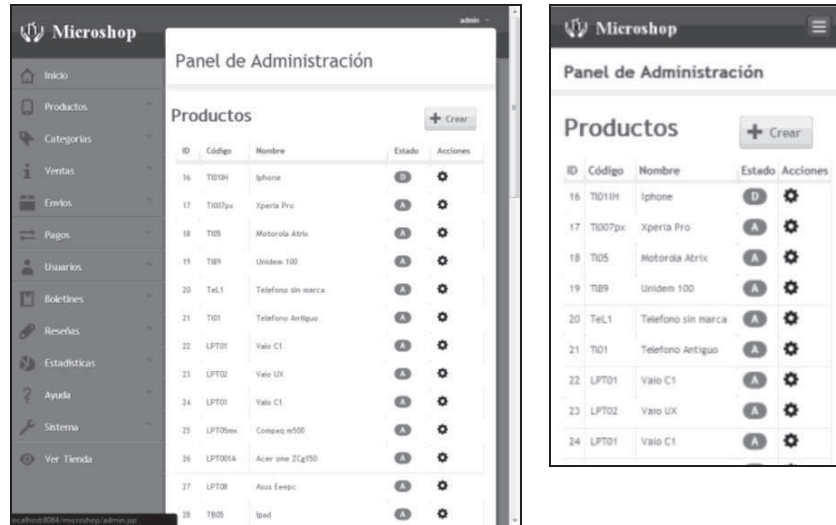


Fig. 3. E-commerce web site displayed on a tablet (960x720), and on a smartphone (320x480).

4.3 Tests on Different Devices

The functionality of the web system was tested by generating an e-commerce web site, and testing every functionality on three different devices: a desktop Intel i7, 8GB RAM, Windows 7 Operating System, with a monitor with resolution of 1024 x 1240 pixels; a tablet HP Touch Pad SnapDragon 1.2GHz, 1GB RAM, Android Ice Cream Sandwich Operating System, with a resolution of 960 x 720 pixels; and a Smartphone Sony Xperia mini pro SnapDragon, 1.6Ghz, 1GB RAM, Android Ice Cream Sandwich Operating System, with a resolution of 320 x 480 pixels. The web browsers used were the following: Firefox V.17, Google Chrome V.25, and Internet Explorer 8, for the desktop computer; and Firefox Mobile, Google Chrome Mobile, and Android web browser, for the tablet and the smartphone.

4.4 Technologies and Tools Used

The main technologies and languages used for the implementation of the web system were the following: Apache Tomcat web server, JSP, servlets and Java beans; MySQL database management system; HTML and XHTML for the web page content; CSS for the presentation and appearance of the web pages; JavaScript for validations and dynamic generation of content for the web pages; JQuery framework for some of the functions that facilitate some tasks; Twitter Bootstrap framework for the creation of complex structures, such as predefined columns and basic components; Java beans as reusable software components; servlets for the business logic of the system, and the implementation of the controller; and JSP for processing information and generating dynamically the web interface of the system.

The tools used for the implementation of the system are free and open source: Star UML was used for creating UML diagrams for the analysis and design stages; NetBeans as an IDE for creating and editing Java classes, JSP, CSS and JavaScript files; Apache Tomcat web server for deploying the web application; DBDesigner as the IDE for modeling the database, creating an entity-relation model, and a relational model; Web Developer Toolbar is a plug-in for different web browsers, which has debugging tools, and other web page design facilities, such as the function to resize the web pages to test different resolutions without closing the web browser; ResizemyBrowser is a web page that resizes the web browser to a specific resolution, which allows testing for mobile devices.

5 Conclusions and Future Work

This paper presented a configurable web system for generating e-commerce web sites, which can be displayed on mobile devices adjusting the resolution automatically. We have successfully tested the functionality of this prototype on three different devices: a desktop computer with the resolution of 1240x1024 pixels, a tablet with a resolution of 960x720 pixels, and a smartphone with a resolution of 320x480 pixels.

The architecture of the system is client-server, supported on the Model View Controller design pattern for the internal functionality, using a controller, a model, and several actions; and the Responsive Web Design pattern for the web interface. They have proved to be design patterns that adapt to these types of applications. The architecture of the system is composed of five components: the core, the Java beans, the model, the actions and the views. A site map of the web system was also presented with all the functionality for administrators, registered and non-registered users.

It is needed to carry out more functionality and usability tests with final users, and administrators of e-commerce web sites, in order to evaluate the system and adjust it accordingly. As part of the future work, it is also planned to incorporate a recommender system for products, based on user preferences and profiles.

References

1. OsCommerce. Available at: <http://www.oscommerce.com/>. Last accessed in June 2014.
2. Magento. Available at: <http://www.magento.com/>. Last accessed in June 2014.
3. Prestashop. Available at: <http://www.prestashop.com/>. Last accessed in June 2014.
4. Opencart. Available at: <http://www.opencart.org/>. Last accessed in June 2014.
5. Comparative table of shopping cart software. Available at: http://en.wikipedia.org/wiki/Comparison_of_shopping_cart_software. Last accessed in June 2014.
6. Zen-Cart. Available at: <http://www.zen-cart.com/>. Last accessed in June 2014.
7. Go eCommerce. Available at: <http://www.go-ecommerce.com/>. Last accessed in June 2014.
8. Shoppizer. Available at: <http://www.shoppizer.com/>. Last accessed in June 2014.
9. Konakart. Available at: <http://www.konakart.com/>. Last accessed in June 2014.
10. Jada-Site. Available at: <http://www.jadasite.com>. Last accessed in June 2014.