Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 3 (2012) 214 – 223

Procedia Technology

The 2012 Iberoamerican Conference on Electronics Engineering and Computer Science

Developing Lift-based Web Applications Using Best Practices

María del Pilar Salas-Zárate[a]*, Giner Alor-Hernández[a], Alejandro Rodríguez-González[b]

[a]*Division of Research and Postgraduate Studies Instituto Tecnológico de Orizaba,Oriente 9 No. 852, Emiliano Zapata, 94320 Orizaba, México*

[b]*Computer Science Department, Universidad Carlos III de Madrid,Avenida Universidad No. 30, 28911, Madrid, Spain*

Abstract

Nowadays, there are a large number of frameworks for developing Web applications such as Struts, JSF, Ruby on Rails and CakePHP. Each framework has its own best practices facilitating the development of web applications, and giving benefits such as saving money, improving the development effort, and reducing the development time. Lift emerges as a new Web framework which has it is own features and learns from the mistakes of other Web frameworks. Some companies such as Siemens and IBM are developing theirs applications in Scala and Lift, as well as social networks like Twitter and Foursquare. However Lift is a new Web framework and for hence it has a lack of documentation for developing Web applications through of best practices. The contribution of this paper is to obtain and discusses the best practices in order to develop Web applications with Lift avoiding common mistakes of design and implementation that other Web frameworks have. With these best practices, the developers can develop more interactive and efficient Web applications integrating features of Web 2.0 technologies with less effort as well as exploiting the framework benefits. Authors consider this work provides a guide for the Web applications development, which impacts in the software engineering community decreasing the number of errors in the implementation phase. Finally, as proof of concept we developed a set of Lift-based Web applications taking into account some best practices such as actors, Comet support, sitemap, wiring and HTML5 support.

Keywords: Best practices; Lift; Scala

* Corresponding author. *E-mail address:* pilarsalas@acm.org.

doi:10.1016/j.protcy.2012.03.023

1. Introduction

A framework is a set of related classes and other supporting elements that make application development easier by supplying prebuilt parts [1]. There are many Java-based Web frameworks such as JSF and Struts, Ruby-based Web frameworks such as Ruby on Rails and PHP-based Web frameworks such as CakePHP, among others. Recently, Lift has emerged as a new Web framework founded as an open source project. It is not part of Scala itself but is built on top of Scala's functional features. Lift's design and implementation extensively apply the advantages of immutable data structures, higher-order functions, abstract data types, and pattern matching [2]. Lift has a main advantage the use of the Scala programming language [3]. Scala is also a hybrid functional and Object-oriented (OO) language, which means getting the power of the higher-level functional languages (such as Haskell, Scheme, to mention a few) while retaining the modularity and reusability of OO components. In particular, the FP (Functional Programming) concept of immutability is well represented in Scala, and is one of the simplest means to provide high scalability. Scala allows doing more in Lift with less lines of code [3]. Nowadays, there is few information about Lift, among this information there are only 3 books in the literature: 1) Simply Lift [4], 2) Exploring Lift [5] and 3) Lift in action [6]; but they do not consider the best practices for developing Lift-based Web applications.

The Lift best practices is necessary, because they help to develop Web applications more efficiently. As well as improving the development effort, reducing the development time, saving money, increasing of the quality of code, the creation of friendly and interactive applications. It is very important for the software engineering and Web engineering communities.

In this paper, the best practices for developing Lift-based Web applications are presented. These best practices were obtained based on the best practices of other frameworks such as Struts, JSF, Ruby on Rails and CakePHP.

This paper is structured as follows. In Section 2, related works are discussed, Section 3 provides a briefly description of 4 Web frameworks for Web application development. Section 4 presents the best practices in Web development. Section 5 presents the best practices supported for development of Lift-based Web applications. In section 6, a case study is presented, and finally, we present our conclusions and emphasize our contribution.

2. Related Works

D. Ghosh et al. [2] introduces the concept Lift; it is designed to make powerful techniques easily accessible, while keeping the overall framework simple and flexible. Lift makes it fun to develop because it lets focus on the interesting parts of coding. Lift has cherry-picked the best ideas from a number of other frameworks, while creating some novel ideas of its own. It is combination of solid foundation and new techniques that makes Lift so powerful. V. Hazrati [7] presents an example for single request processing. The lift code, running inside Tomcat, ran 4 times faster than the Rails code running inside Mongrel. However, the CPU utilization was less than 5% in the lift version, where it was 100% of 1 CPU (on a dual core machine) for the Rails version. P. David et al. [8] developed a multiuser, real-time chat application in Lift. Also, Scala's language features that make Lift possible are discussed. The application provides a single chat server that takes chat messages and redistributes the messages out to all listeners. C. Derek et al. [3] presents the applicability of the programming language Scala together with Lift as a development framework for the Web. Scala offers strong functional capabilities, and Lift exploits Scala's benefits. G. Michael [9] developed a real-time Web auction using advanced Ajax libraries, such as jQuery, make it easy to write Comet applications on the client side, but getting them to scale on the server is still a challenge. That is where the Scala programming language and the Lift Web application framework can step in and deliver a scalable back end for the Comet application. S. Balaraman [10] introduced the concepts GWT and Spring MVC. GWT is amazing for rich applications, there's a lot of Web programming that doesn't require its full power. Spring MVC is a Web framework, open and configurable system that doesn't impose too many restrictions and is easy to attach to other technologies. M. Scares et al. [11], proposed an object-oriented multi-layer framework. This framework allows automating basic structural tasks for enterprise applications, freeing developers from time-consuming, low aggregated value tasks and allowing them to concentrate on the actual goal of software development: the implementation of user requirements. The framework is constructed over three conceptual layers - presentation layer, business layer and service layer - each one playing a well-defined role in the architecture. W. Jason Gilmore [12] introduced the concept Zend Framework. In this article the purpose is threefold. First, it makes the case for why should seek to embrace one of the most crucial of these best practices, known as the Model-View-Controller (MVC) design architecture. Second, it introduces several of the most popular PHP-driven frameworks, each of which allows taking advantage of MVC, in addition to a variety of other time-saving features such as Ajax integration. Finally, it devotes additional time to the Zend Framework, which, although the newest of the bunch, is rapidly becoming the most popular of these framework solutions. D. Ridjanovic [13] presented a brief overview of the Web framework named Domain Model RAD. This framework is used for developing dynamic Web applications with a minimum amount of programming. Also, Domain Model RAD uses Domain Model Lite to represent a domain model of a Web application. L. Xufeng, et al. [14] proposed a novel concept called Smart Business Object (SBO). In essence, SBOs are Web-ready business objects. SBOs have high-level, Web-oriented attributes such as email, URL, video, image, document, among others.

3. Frameworks for Web Application Development

There are two kinds of Web frameworks: request-based and component-based frameworks. Request-based frameworks use controllers and actions that directly handle incoming requests. These frameworks have differences by the way in how data are structured and provided to the developer. Examples of these kinds of Web frameworks are Struts, Grails and Ruby on Rails.

Component-based frameworks abstract the developer away from the internals of the request handling and encapsulate the logic into reusable components, often independent from the Web medium. The different frameworks basically differentiate themselves by the provided component API and how components are combined together [1]. Examples of these kinds of Web frameworks are JSF(Java Server Faces), Tapestry or wicket.

The main goals of a Web framework are: 1) to accelerate the process development and; 2) to reuse existing code. The most used Web frameworks implement the MVC pattern and have features such as dispatching, navigation, tag support, conversion, use of HTML designer, validation, view technologies, extensibility, template, AJAX support and internationalization.

Web frameworks such as Struts, JSF, CakePHP and Rails are compared trough their best practices. These four Web frameworks were selected because they have great maturity and great use, which make them one of the most powerful Web frameworks. JSF is now the preeminent server-side Java Web framework. A strong point of JSF is its extensible component model, and a large number of third-party components have become available [15]. Apache Struts is an open source Java framework used for building Web applications based on the servlet and Java Server Pages (JSP) technologies [16]. Ruby on Rails is a Web application framework written in Ruby, a dynamically typed programming language similar to Python, Smalltalk, and Perl. Rails is an MVC (model-view-controller) framework [17]. CakePHP is a free, open-source, rapid development framework for PHP. It's a foundational structure for programmers to create Web applications. It is a primary goal is to enable to work in a structured and rapid manner–without loss of flexibility [18].

In the next section, the best practices in JSF, Ruby on Rails, Struts, CakePHP and Lift for Web development are presented.

4. Best Practices in Web Development

The best practices are techniques or important aspects identified by users, which helps to develop Web applications more efficiently. The use of best practices has benefits such as save time and save money, the increase of the quality of code and the creation of friendly and interactive applications. Some examples of companies that use best practices are Microsoft, Facebook, IBM, Yahoo and Google.

In this paper the best practices considered were selected based on the most popular best practices compared among different frameworks such as JSF, Struts, CakePHP, Ruby on Rails, which have already been studied and reported in the literature.

In Table 1, the common best practices in frameworks for Web application development described in section 2 are presented.

Table 1. Best practices to Web development in JSF, Ruby on Rails, Struts, CakePHP and Lift

Best practice	Description
Internationalization	It is the process of designing an application in order to be adapted to various languages and regions.
Forms validation	It is the process of checking that a form has been filled in correctly before it is processed. There are 2 methods for validating: server side and client side.
AJAX support	It is used for development interactivity applications.
ORM	It is a programming technique for converting data between incompatible types systems in object-oriented programing languages
Template framework	Template systems that support
REST Support	It allows the development of REST-based Web services
Customization and Extensibility	It adds functionality and customizes the behavior of the application with plugins or other options.
Testing	It provides stakeholders with information about the quality of the product or service under test

A summary of the best practices supported for JSF, Ruby on Rails, Struts, CakePHP and Lift is presented in Table 2.

Table 2. Summary of the best practices supported for Web development in JSF, Ruby on Rails, Struts, CakePHP and Lift

Best practice	**JSF**	**Ruby on Rails**	**Struts**	**CakePHP**	**Lift**
Internationalization	Yes	Yes	Yes	Yes	Yes
Forms validation	Yes	Yes	Yes	Yes	Yes
Ajax support	Yes	Yes	Yes	Yes	Yes
ORM(Object Relational Mapping)	Yes	Yes	Yes	Yes	Yes
Template framework	Yes	Yes	Yes	Yes	Yes
REST Support	Yes	Yes	Yes	Yes	Yes
Customization and Extensibility	Yes	Yes	Yes	Yes	Yes
Use actors	No	No	No	No	Yes
Comet support	No	No	No	No	Yes
Use pattern matching	No	Yes	No	No	Yes
HTML5 support	No	No	No	No	Yes

Best practice	JSF	Ruby on Rails	Struts	CakePHP	Lift
SiteMap or Automatically create menus	No	No	No	No	Yes
Use wiring	No	No	No	No	Yes
parallel rendering	No	No	No	No	Yes

In the next section, the best practices for Lift-based Web application development are presented.

5. Best Practices Supported for Development of Lift-based Web Applications

The best practices of Lift-based Web applications are described below.

- **Internationalization**: One of the best things about Lift is its flexible template and resource localization system. Lift supports a new type of resource bundle for I18N, covered in Localization. The new resource bundles can be global and per-page.
- **Forms validation**: Much of the Web is creating input forms for users to submit, validating those input forms and if the forms pass validation, an action is performed. If the forms do not pass validation, the user received a message with fields which caused the validation problems and the system gives an opportunity to fix them . Lift provides a single-screen input/validation mechanism called LiftScreen and a multi-page input/validation mechanism (with stateful next/previous buttons) called Wizard.
- **Custom Error Messages**: Feedback to the user is important. The application must be able to notify the user of errors, warn the user of potential problems, and notify the user when system status changes. Lift provides a unified model for such messages that can be used for static pages as well as for AJAX and Comet calls.
- **Ajax support**: It offers support for asynchronous client-server interaction. It is the process of exchanging data with a server, and update parts of a Web page without reloading the whole page.
- **ORM**: The ORMs included in Lift are Mapper and record, Mapper is an ORM system for relational databases that lets query database and represents data in Scala objects. Record is a thin layer over a persistence mechanism for persisting objects.
- **Template framework**: Lift uses a powerful template system based on the view-first approach, which allows modifying pages and reusing page components much simpler.
- **REST support**: REST is a simple architectural style which uses plain XML or JSON as a communication medium. Lift providing REST-based Web services.
- **Customization and Extensibility**: Lift offers more customization on this snippet than just emitting some XHTML. By specifying some prefixed attributes on the tag itself, attributes can be added to menu elements. One common customization of widget would be to override the CSS used. To do this, it provides its own style.css file.
- **Use Lift actors**: Lift uses actors and immutability as part of its implementation framework and encourages Web-application development based on events and messages. One of the main areas in Lift that uses actors as a basic architectural unit is its Comet support. Actors represent a model of computation based on asynchronous message-passing concurrency that does not restrict the message-arrival ordering.
- **Comet support**: Lift has a powerful support for Comet, it allows using Web 2.0 features with little effort. Comet is a technique that allows a web application to push messages from server to client.
- **Use Scala pattern matching**: Lift makes use of Scala's pattern matching to allow match incoming HTTP requests, extract values as part of the pattern matching process and return the results. Scala's pattern matching is powerful.

- **HTML5 support**: Lift supports parsing HTML5 input files and rendering HTML5 to the browser in addition to Lift's XHTML support. It offers features such as HTML 2D (canvas and SVG), HTML 3D (WebGL), geolocation, audio, video, among others.
- **SiteMap**: It defines navigation and allows creating hierarchical menus.
- **Wiring**: Lift's Wiring allows declaring relationships among the various elements on a page and when any of the precedent elements change, the dependent items are redisplayed on the next HTTP response [19].
- **Parallel rendering**: It allows access multiple external resources in parallel. Lift allows executing multiple snippets in parallel during the rendering of a Web page.

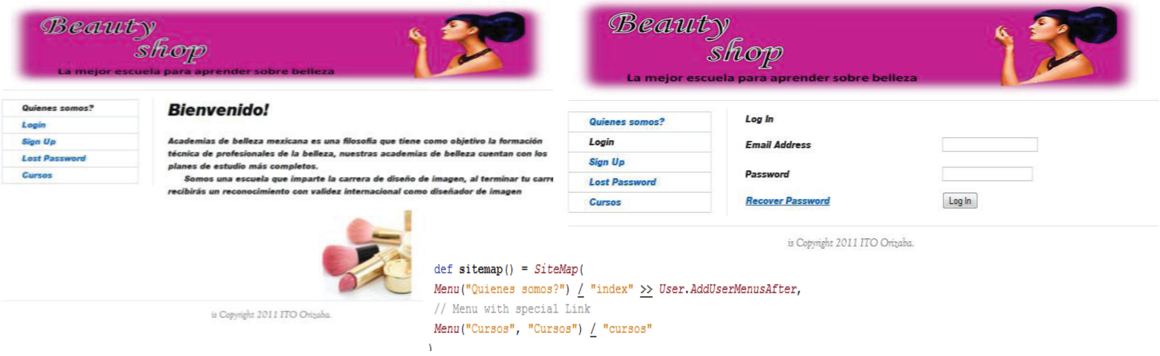
6. Case Study: Best Practices in Lift-based Web Applications

In this section, we discuss why lift is very powerful and interesting framework, and why the best practices mentioned in the above section and their usage are important on Web development.

Internalization in Lift allows Web application multi-language development; this is of great use on websites such as news. Other feature is the forms validation; it allows validating input forms before to be processing it. If the forms do not pass validation, the user received a message with fields which caused the validation problems and system gives an opportunity to fix the problems through of use the custom error messages which allow showing to the user messages more clear about of his errors, this is of great use on websites such as register in a store shopping online. Lift provides REST-based Web services, a simple architectural style which allows using API libraries such as Twitter®, Flickr®, Facebook® and others. It makes websites easier to use and more interesting to explore, an example is the website Foursquare® incorporating Twitter®.

Comet and actors are very interesting in Lift. These best practices allow an open communication among the server-side, client-side and asynchronous message-passing. It offers benefits such as a better real-time interaction, usability and eliminating the limitations of the page-by-page Web model. This is of great use in chat applications, or websites where the users can post comments, for example series or movies applications. HTML5 is very interesting too because it allows an easy development of interactive Web applications adding features that offers some traditional RIA frameworks, such as HTML 2D, HTML 3D, geo-location, drag and drop, audio and video among others. An advantage in terms of security in Lift is SiteMap. SiteMap provides a unified access control, site navigation, and menu system on Lift-based Web applications development.

Next, to show the best practices that other Web frameworks do not have, a set of Web applications are presented. In Fig. 1, a Web application with a menu automatically created and easily customized is presented. This screenshot shows the feature SiteMap in the development of a Lift-based Web application.



Academias de belleza mexicana es una filosofía que tiene como objetivo la formación técnica de profesionales de la belleza, nuestras academias de belleza cuentan con los planes de estudio más completos.

Somos una escuela que imparte la carrera de diseño de imagen, al terminar tu carre recibirás un reconocimiento con validez internacional como diseñador de imagen

is Copyright 2011 ITO Orizaba.

Email Address

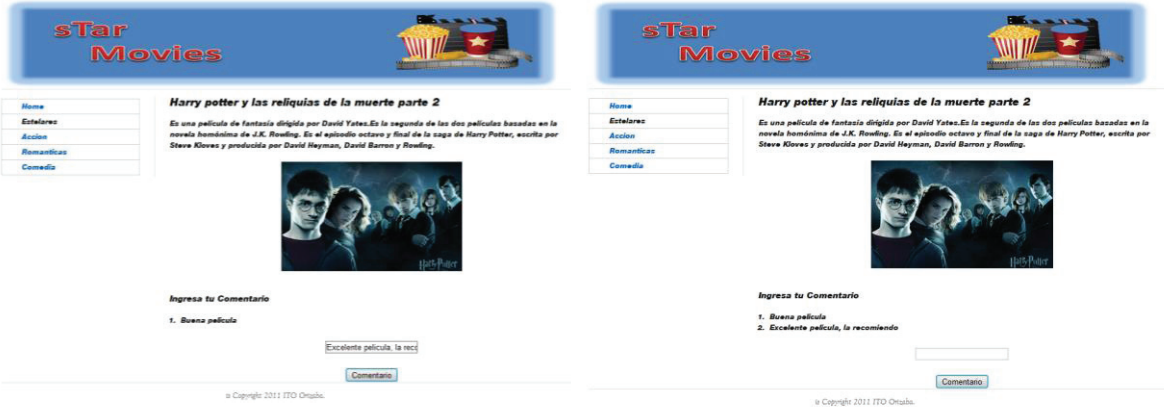
Password

is Copyright 2011 ITO Orizaba.

```
def sitemap() = SiteMap(
Menu("Quienes somos?") / "index" >> User.AddUserMenusAfter,
// Menu with special Link
Menu("Cursos", "Cursos") / "cursos"
)
```

Fig. 1. Lift-based Web application using customized menus.

In Fig. 2, a Web application with the feature of using actors in the development of Lift-based Web application is presented. This is a movies application, in which the users can type comments about any movie, and these comments are automatically displayed in the browser.



Harry potter y las reliquias de la muerte parte 2

Es una pelicula de fantasia dirigida por David Yates.Es la segunda de las dos peliculas basadas en la novela homónima de J.K. Rowling. Es el episodio octavo y final de la saga de Harry Potter, escrita por Steve Kloves y producida por David Heyman, David Barron y Rowling.

Ingresa tu Comentario

1. Buena pelicula

Excelente pelicula, la reco

Copyrigt 2011 ITO Orizaba.

Harry potter y las reliquias de la muerte parte 2

Es una pelicula de fantasia dirigida por David Yates.Es la segunda de las dos peliculas basadas en la novela homónima de J.K. Rowling. Es el episodio octavo y final de la saga de Harry Potter, escrita por Steve Kloves y producida por David Heyman, David Barron y Rowling.

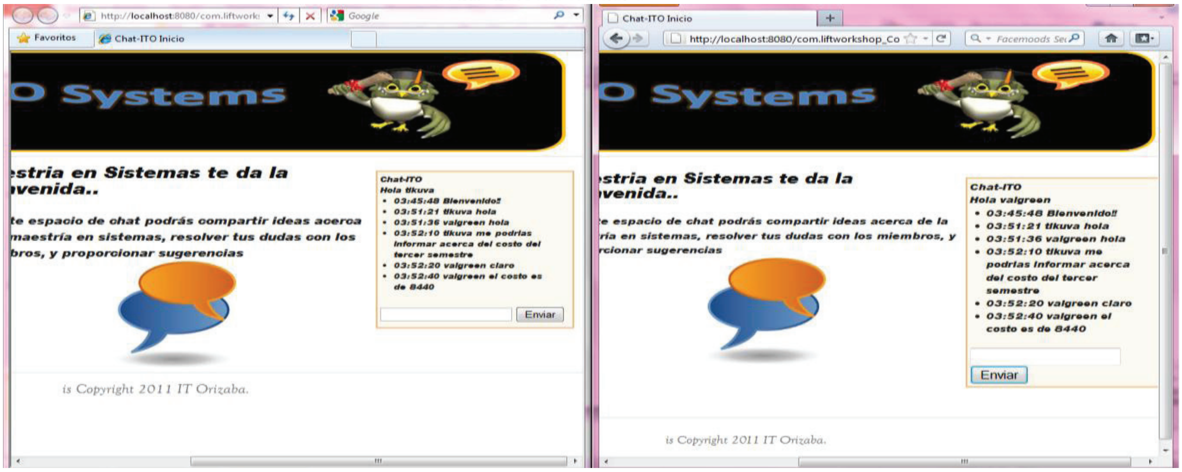
Ingresa tu Comentario

1. Buena pelicula
2. Excelente pelicula, la recomiendo

Copyright 2011 ITO Orizaba.

Fig. 2. Lift-based Web application using actors

In Fig. 3, a Web application with Comet support is presented. This is a Web chat application, in which students can share ideas or ask questions about of a professor of the Instituto Tecnologico de Orizaba.



http://localhost:8080/com.liftwork

stria en Sistemas te da la venida..

te espacio de chat podrás compartir ideas acerca maestría en sistemas, resolver tus dudas con los bros, y proporcionar sugerencias

Chat-ITO
Hola tlkuva
- 03:45:48 Bienvenido!!
- 03:51:21 tlkuva hola
- 03:51:36 valgreen hola
- 03:52:10 tlkuva me podrias informar acerca del costo del tercer semestre
- 03:52:20 valgreen claro
- 03:52:40 valgreen el costo es de 8440

is Copyright 2011 IT Orizaba.

http://localhost:8080/com.liftworkshop_Co

stria en Sistemas te da la venida..

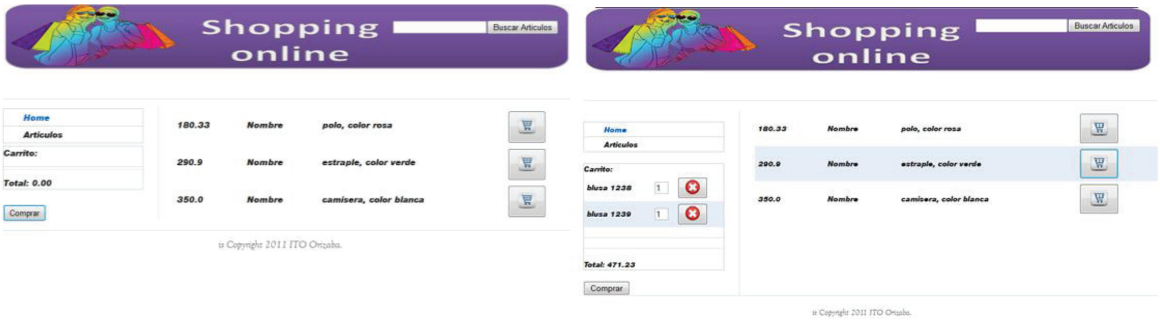
e espacio de chat podrás compartir ideas acerca de la ría en sistemas, resolver tus dudas con los miembros, y rcionar sugerencias

Chat-ITO
Hola valgreen
- 03:45:48 Bienvenido!!
- 03:51:21 tlkuva hola
- 03:51:36 valgreen hola
- 03:52:10 tlkuva me podrias informar acerca del costo del tercer semestre
- 03:52:20 valgreen claro
- 03:52:40 valgreen el costo es de 8440

is Copyright 2011 IT Orizaba.

Fig. 3. Lift-based Web application with Comet support

In Fig. 4, a Web application with the feature of using wiring is presented. This Web application is an online clothing store, in which the users can add items to the shopping cart. When the items are added or removed, the shopping cart is updated. Using this feature provides a solution for the management of complex dependencies in a single Web page, as in this example.



Carrito:

Total: 0.00

180.33	Nombre	polo, color rosa
290.9	Nombre	estraple, color verde
350.0	Nombre	camisera, color blanca

© Copyright 2011 ITO Orizaba.

Carrito:

blusa 1238	1
blusa 1239	1

Total: 471.23

180.33	Nombre	polo, color rosa
290.9	Nombre	estraple, color verde
350.0	Nombre	camisera, color blanca

© Copyright 2011 ITO Orizaba.

Fig. 4. Lift-based Web application using wiring

In Fig 5, a web application with HTML5 support is presented. This is a news Web application in different languages, which shows videos of different format. It shows the functionality of the tag <video> of HTML5, which is used to display a video in a HTML document. The screenshot shows other feature of HTML5, the geolocation API that allows to users can enjoy the benefits of various location-aware services.

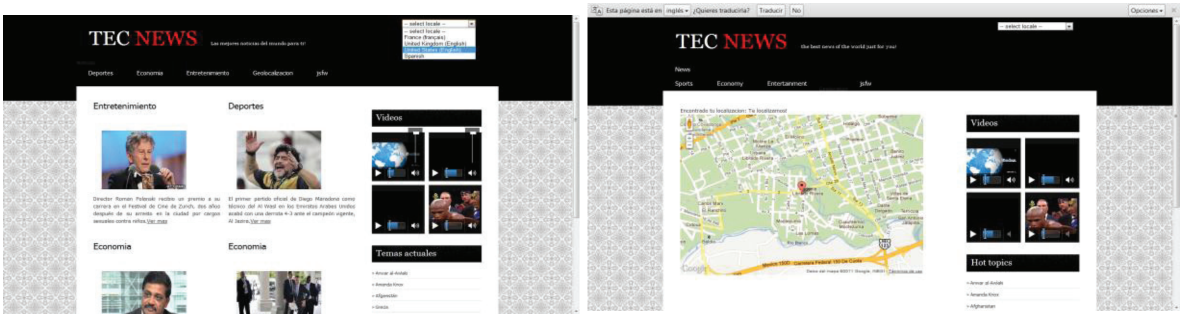


Fig. 5. Lift-base Web application with HTML5 support

Lift is a Web framework with important features: 1) Web applications are fast to build, easy to maintain and concise and offers advantages of security and developer productivity; 2) Lift offers a separation of presentation content and logic, Lift has a templating mechanism that does not allow any logic code in the view. The advantages of developing these Web applications through of best practices are faster development, decrease of code, save time, a tremendous advantage in terms of security and build dynamic applications.

Next, the advantages of using the best practices mentioned above (SiteMap, use actors, Comet support, Wiring and HTML5 support) are discussed. The use of SiteMap allows developers to define the access control rules for each Web page, avoiding the process of updating the XML-based security rules when the URLs change or the methods that calculate the access control change. It allows developing more secure Web applications. The use of actors on Web applications offers advantages such as unify threads and events (being efficient and scalable), automatically mapped to multiple JVM threads with the purpose of leverage multi-core processors.

Lift provides support functions and classes to simplify the development of applications that utilize techniques such as comet, in which the server sends data to the browser only when is needed, thus decreasing the latency and got an asynchronous communication, it allows developing more intuitive Web applications.

The use of Wiring in Lift has advantages due it can create very complex inter-relationships with the elements on the screen, when one is updated all the dependent elements are automatically updated. This is a tremendous save time and allows ease of maintenance of the Web site because the administrator does not have to know all the dependencies [4].
Finally, HTML5 support offers advantages allowing easily the development of Web applications with Web 2.0 features including audio, video, drag and drop , HTML 2D and HTML 3D among others.

7. Conclusions and Future Work

The use of best practices allows developing better Web applications and provides efficiency. Lift is a new framework with few information about itself, the analysis presented in this work shows that Lift offers more features to developing Web applications than others frameworks, such as comet support that allows a Web server to push data to a browser, without the browser explicitly requesting it, offering a better real-time interaction, other feature is SiteMap which is an advantage in terms of security. Lift has HTML5 support, it is an important feature which allows to easily integrate features of modern Web applications as too. Lift exploits language Scala benefits such as use pattern matching which provides a powerful tool for declaring business logic in a concise and maintainable way and use actors, they represent a model based on asynchronous message-passing concurrency. These features make Lift a powerful framework. With the use of these best practices, Web applications were developed in an interactive, intuitive and secure way, improving the development effort and reducing the development time.

As future work, we are considering to obtain new best practices such as security, use CSS selector and cloud computing support.

Acknowledgements

This work is supported by the General Council of Superior Technological Education of Mexico (DGEST). Additionally, this work is sponsored by the National Council of Science and Technology (CONACYT) and the Public Education Secretary (SEP) through PROMEP. Also, this work is supported by the Spanish Ministry of Industry, Tourism, and Commerce under the project TRAZAMED (IPT-090000-2010-007) and the Spanish Ministry of Science and Innovation under the project FLORA (TIN2011-27405).

References

[1] F. Neal. 2003. Art of Java Web Development: Struts, Tapestry, Commons, Velocity, Junit, Axis, Cocoon, Internetbeans, Webwork. Manning Publications Co., Greenwich, CT, USA.

[2] D. Ghosh and V. Steve. 2009. Scala and Lift ;Functional Recipes for the Web. IEEE Internet Computing 13, 3 (May 2009), 88-92. DOI=10.1109/MIC.2009.68 http://dx.doi.org/10.1109/MIC.2009.68

[3] C. Derek, D. Marius, P. David, W. Tyler "Basic Lift". In starting with Lift. 2008, pp. 9-23

[4] P. David.: " in Simply in Lift", 2010.

[5] C. Derek, D. Marius, W. Tyler. "in Exploring Lift", 2011.

[6] P. Timothy, "in Lift in action", Ed. Bath, England: Manning Early, 2011.

[7] V. Hazrati, "Overview of Lift web framework", Conference on Java technology., Pune India, 2009.

[8] P. David and V. Steve. 2010. A Chat Application in Lift. IEEE Internet Computing 14, 3 (May 2010), 88-91. DOI=10.1109/MIC.2010.61 http://dx.doi.org/10.1109/MIC.2010.61

[9] G. Michael , Build comet applications using Scala, Lift, and jQuery. Software architect eBay, 2009, pp. 1-21

[10] S. Balaraman. 2009. "integrating GWT with a powerful web framework", Review of "Pro Web 2.0 Application Development with GWT by S. Balaraman," Jeff Dwyer, Apress, 2008, ISBN: 9781590599853. Queue 7, 1 (January 2009), 40-40. DOI=10.1145/1508211.1508224 http://doi.acm.org/10.1145/1508211.1508224

[11] M. Scares , et al., 2006, in International federation for information Processing, Volume 205, Research and practical issues of enterprise information systems, eds. Tjoa, A.M., Xu, L., Chaudhry, S., (Boston:Springer), pp.285-296.

[12] W. Jason Gilmore. 2010. “MVC and the Zend framework”, Beginning PHP and MySQL: From Novice to Professional, Fourth Edition (4th ed.). Apress, Berkely, CA, USA. , pp.603-620.

[13] D. Ridjanovic, “Rapid development of web applications with web components”, Québec, Canada, 2007, pp. 99-103

[14] L. Xufeng , A. Ginige, “Smart business objects for web applications: A new approach to model business objects”, Sydney, Australia. 2006, pp.1-7

[15] H. Cay, JavaServerFaces 2.0 , DZone Refcardz .PC Magazine, 2009, pp.1-6

[16] S. Shenoy, "Getting started", Struts Survival Guide, Ed.New York: ObjectSource LLC. 2005, pp. 25

[17] D. Thomas et al.,"Introduction to Ruby",Agile Web development with Rails, Ed.Raleigh, North Carolina. 2008, pp. 647-648

[18] M. Curry, "what is CakePHP", Super awesome advanced CakePHP tips,Ed. United States: creative commons .2009, pp.10

[19] C.Derek , T. Weir, and Marius Danciu. 2009. The Definitive Guide to Lift: A Scala-Based Web Framework. Apress, Berkely, CA, USA.