

Best Practices for Web Development using Grails and Django

Lizbet-Paola Mora-Murguía, Giner Alor-Hernández, Beatriz Alejandra Olivares-Zepahua,
Luis Ángel Reyes-Hernández, and Ana María Chávez-Trejo

Instituto Tecnológico de Orizaba, Division of Research and Postgraduate Studies,
Orizaba City, Veracruz, Mexico
pao2189@hotmail.com, galor@itorizaba.edu.mx, bolivares@ito-
depi.edu.mx, l.a.reyes.h@gmail.com, achavez@itorizaba.edu.mx

Abstract. A best practice is a technique or an important aspect that helps to develop Web applications more efficiently. Best practices on Web frameworks reduce the development time and effort, saving money, increasing the quality of code, enabling to create friendly and interactive applications. This paper is focused on identifying and providing according to the experiences and requirements of the users, the best practices for Web development by using Grails and Django Web frameworks. With these best practices, 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. As proof of concept we have developed a set of Web applications by using best practices as HTML5, Comet support, AJAX, ORM, extensibility, among others.

Keywords: Web frameworks, Grails, Django, Web 2.0.

1 Introduction

A framework is a high-level solution for the reuse of software pieces, a step forward in simple library-based reuse that allows the sharing of not only common functions but also the generic logic of a domain application. It also ensures a better level of quality for the final product, given the fact that an important fragment of the application is already found within the framework and it has already been tested [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, to mention but a few. Grails and Django have emerged as Web frameworks open source for their features as scalability, functionality, reduction of development time, increase the quality of code by using minimal resources. These Web frameworks are based on Groovy and Python respectively. Additionally, these Web frameworks allow developing applications in an easy and interactive way which are very important for software developers and Web engineering.

Best practices on Web development have been proposed and applied facilitating the development and improving the final product quality. Some examples are Testing a Website: Best Practices [2], Jakarta Struts Live and best practices [1], Struts Survival Guide, Basics to best practices [3], Applying best practices in website redesign: the Queens College Libraries experience [4], A Framework for Process Management in Service Oriented Virtual Organizations [5], Staying Afloat in an Expanding Sea of Choices: Emerging Best Practices for Eclipse Rich Client Platform Development [6].

This work is structured as follows. Section 2 discusses the state-of-the-art on the development of Web applications by using best practices. Section 3 provides a brief description and overview of the main Web frameworks for Web development. Section 4 addresses the description in detail of best practices on Web development. Section 5 presents a comparison and discussion among Ruby on Rails, JSF, Struts, CakePHP, Lift and Yii Web frameworks. Finally, section 6 remarks the conclusions and future work emphasizing our contribution.

2 Related Works

In recent years, several works have been proposed with the aim of obtaining the best practices of Web frameworks. Also, these works provide information related to some main features of Web frameworks. Jim et al. [7] determined the necessary knowledge for the internationalization of standards, maintenance, access and usability for people with disabilities as well as inspections and tests in the use of websites. The research introduced the IEEE Std 2001-2002 standard as a best practice to enhance websites design. Darlene F. et al. [8] discussed that most developers are focused on functionality specifically without taking into account the design and content bringing to implement reengineering. To solve this, a series of best practices is presented in order to create content and graphical interfaces of web applications in a clear way. Xufeng L. et al. [9] 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, image, video, documents, among others. Richard T. et al. [10] evaluated the experience of the developers, balance, agility, discipline, cost-benefit, maturity and competencies of organizations, for the determination of a set of best practice. J. Barrie et al. [11] emphasized the lack of evidence about the use of best practices used effectively. The research recommended disseminating information about them among evaluators and advisors. Alex N. et al. [12] analyzed that the mobile data traffic is most used on iOS and android platforms. The research proposed to use the best practices of desktop applications for fastest experiences in mobile applications. James M. et al. [13] examined the best practices in website re-design. The research established for its two interconnected parts, the web development process and web design. The authors demonstrated how best practices were applied to coordinate a library web site redesign project and to engineer the website for optimum usability, resulting in the creation of a new improved website. S. Rodrigo et al. [14] determined that

for most applications, there are multiple tasks that are repetitive, complex and prone to errors, which proposed a framework of multilayer object oriented automation of basic tasks for business applications. W. Harrison et al. [15] mentioned that the term "best practice" has often been misused in some publications, in this context they defined as experience the true meaning of a best practice contributing to a better performance.

As can be observed, the aforementioned works suffer from several drawbacks such as a) an analysis of different Web frameworks based on their best practices is missed; b) only the best practices of Java-based frameworks are well-known; c) there are not guides and reports for implementation of best practices with Django and Grails frameworks; This work tries to solve the aforementioned deficiencies by identifying and providing best practices as well as implementation examples.

3 Frameworks for Web Development

There are several types of Web frameworks: user interface-oriented such as Java Server Faces, document publishing-oriented as Cocoon, event control-oriented such as Struts, actions and controls-based directly managing incoming requests with structure in particular as Grails, and component-based which encapsulate the logic of reusable components such as Tapestry [16]. The main objectives of a Web framework are: 1) rapid application development, 2) reuse existing code. The most widely used Web frameworks implement the MVC pattern and they have features such as tags support, HTML integration, form validation, extensibility, using templates, Ajax support and internationalization. Struts, JSF, CakePHP, Rails, Yii, Grails and Django are compared through their best practices. Table 1 shows seven Web frameworks which were chosen because they are the most used in entertainment-focused companies like Netflix, Facebook[®], SKY[®], ESPN[®], Twitter[®] or YouTube[®].

Table 1. Types of Web frameworks

Frame-work	License	General Description
Struts	Open Source, (Apache)	It is a Java-based open source used to create Web applications based on servlets and Java Server Pages technology (JSP) [17]
JSF	Open Source, (CDDL+GPL)	It is a Java-based web application framework intended to simplify development integration of web-based user interfaces [18]
CakePHP	Open Source, (MIT/X11)	It is a PHP-based having a basic structure for Web applications; its main purpose is to allow work in a structured and quickly without loss of flexibility [19].
Ruby on Rails	Open Source, (MIT)	It uses the MVC pattern written in Ruby, a programming language for dynamic types like Python, Smalltalk, and Perl [20].
Yii	Open Source, (BSD)	This framework is ideal for high performance, written in PHP 5 oriented rapid construction of large-scale applications. Its main

		goals are ease, extensibility and efficiency by using the MVC pattern [21].
Grails	Open Source (Apache 2.0)	Java-based framework that uses a scripting language called Groovy, but it has no specific IDE plugins for each one of the existing Java-based IDEs.
Django	Open Source, (BSD)	This framework uses Python programming language, which promotes rapid development and clean design, practical, concise and powerful dynamic Web pages. This framework uses the pattern (Model-View- Template) [22].

Although these frameworks are open source, they use different programming languages and other general characteristics. However, best practices are common among them.

4 Best Practices for Web Development

Best practices are activities, technical or important issues identified by users, who have rendered excellent service and they are expected to achieve similar results on similar contexts. The use of best practices has benefits such as save time, save money, increase code quality and create easy and interactive applications.

Best practices follow standards, unit testing, code reuse, version control and use recognized design patterns. In this paper, the main purpose was to gather and present a list of the most relevant engineering practices prevalent in the development community today by analyzing best practices reported by others studies of Web frameworks such as books and Web development community; identifying those that enable three of the most important quality criteria for success of Web applications 1) Reliability, 2) Usability and 3) Security. In Table 2, best practices selected for Web development are described in order to generate code quality, standardize the development, ease of maintenance, providing security and scalability, facilitate reuse, maintenance and integration evenly and encourage rapid application development.

Table 2. Best Practices for Web Development

Best Practice	Description
Scaffolding	It is a technique supported by some model-view-controller frameworks, in which the programmer may write a specification that describes how the application database may be used.
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 correctly filled before it is processed. There are two methods for validation: client-side and server-side.
Using Templates	Templates allow modifying Web pages and reusing Web page compo-

	nents much simpler.
Customization for error messages	Custom error messages provide more informative, or friendly, feedback than standard error messages. It aims to the user to understand more clearly each mistake for a process that user performs.
Security	It permits prevention of common vulnerabilities such as SQL injection, loss of authentication and session management, access restriction fails to URL, among others.
HTML5	It establishes a series of new elements and attributes that reflect typical usage on modern Web sites.
AJAX Support	It is used for s asynchronous client-server interaction in order to develop interactive Web applications.
Comet Support	Comet is a technique that allows a Web application to push messages from server to client.
Extensibility	It adds functionality and customizes the behavior of the application with plugins or other options.
ORM	It is a programming technique for converting data between incompatible types systems in object-oriented programing languages.
Compliance to W3C Standards	They are a set of recommendations that serve as reference to build Web applications with quality, accessible, interoperable and efficient, where increasingly develop robust applications.

Some companies have adopted the use of best practices for the development of their own web sites. Some examples are official sites of Netflix[®], Sky TV[®] Guide, Ticketbis[®], Yellow Pages[®] and Vodafone Music[®].

There are other best practices for software development such as: pattern matching, SiteMap, cloud computing, REST (Representational State Transfer) support, wiring, Parallel rendering, which they are out of the aim and scope of this work.

5 Comparative Analysis of Best Practices among Web Frameworks

In this section a comparative analysis of best practices for Web development is presented. Essentially, the selection of the best Web development framework has become more complex, specifically due to three reasons: 1) a broader perspective, the Web technology landscape has undergone a radical change; 2) there is more choice of technology; and 3) requirements for modern Web applications are changing. After an extensive research [23], the best Java-based frameworks Web were obtained, these frameworks are JSF and Struts. Grails is currently one of the most popular open source web framework for the Java platform due to it takes advantage of Groovy's dynamic language features. CakePHP is one of the best Web frameworks in the PHP programming language according to the top 10 ranking¹. Finally Ruby on Rails is the most popular framework of the Ruby programming language².

¹ http://www.mustap.com/phpzone_post_73_top-10-php-mvc-frameworks.

This comparative analysis is supported by JSF, Ruby on Rails, Struts, CakePHP, Yii, Grails and Django Web frameworks as shown in Table 3.

Table 3. Best practices for Web development supported by Struts, JSF, CakePHP, Ruby on Rails, Yii, Grails and Django

Best Practices	Struts	JSF	Cake PHP	Ruby on Rails	Yii	Grails	Django
Scaffolding	No	Yes	Yes	Yes	Yes	Yes	Yes
Internationalization	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Forms validation	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Using Templates	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Customization for error messages	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Security	Yes	Yes	Yes	Yes	Yes	Yes	Yes
HTML5	No	No	No	No	Yes	Yes	Yes
AJAX Support	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Comet Support	No	No	No	No	Yes	Yes	Yes
Extensibility	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ORM	Yes	Yes	Yes	Yes	Yes	Yes	Yes

The support of best practices on Grails and Django-based Web applications are described below.

Scaffolding: Django has an interesting aspect by offering support to build friendly interfaces in order to dominate the CRUD (Create, Read, Update and Delete) operations by using SQLite3 database. Django has support for other databases as MySQL, Postgres and Oracle by configuring the settings.py file; to automatically generate the classes and methods in Python-based code to interact with the database.

Internationalization: Grails has support (i18n) by leveraging the MVC architectural pattern, customizing the text that appears in the views according to the user location. A Locale Object represents a specific geographical area, this will need to specify it by its language code in the Message Bundles folder in the properties files. In Fig. 1, a screenshot of a Grails-based Web application by using Internationalization switching to Spanish-México and English language is shown.

² http://en.wikipedia.org/wiki/Comparison_of_web_application_frameworks

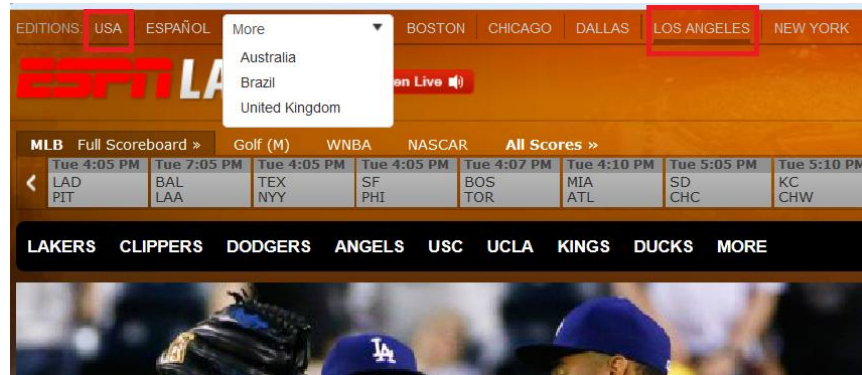


Fig. 1 Grails-based Web application using internationalization

Form Validation: Django provides some validation rules in the description of the fields within the file where the form methods are located but not in the view. By using the Python programming language, these fields are specified, for instance, `required = True` indicates that the field is required, throwing a warning message in the event when an entry is not valid.

Using Templates: Grails has a set of predefined templates which allows the rendering process of Web pages. The templates are located in a specific folder and their names always starts with low underlined. In Fig. 2, an example of the use of templates for a Grails-based Web application is shown. This Web application represents an online shopping web site displaying information about the author and other details in the footer.

Custom error messages: Warn user of possible problems and notify when the system state changes is simple with Django since into the design view such messages can be directly modified in the CSS files of the Web application.

Comet support: Grails provides Comet support through different plug-ins such as Atmosphere, Cometd, ICEPush, among others. CometD is a scalable HTTP-based event routing bus that uses an Ajax Push technology pattern known as Comet. This plugin allows to Grails-based application sending asynchronous notifications to HTTP clients by using CometD and the Bayeux protocol. The Atmosphere project is a portable Ajax-Push/Comet and WebSocket framework. ICEpush is purely web-based notification, allowing group members to receive a callback when a push is sent to a group. ICEpush allows adding Ajax Push collaboration features to any web application.



Fig. 2 Grails-based Web Application using templates.

In Fig. 3, an example of employing the Comet support by using a Web chat for the ESPN web site is shown. This example represents a Grails-based Web application which allows multiple conversations/interactions with students. The Atmosphere plugin is used and with this feature Web application is busy, dynamic and interactive.

Security: Django is designed to automatically protect Web applications in different ways. The most common is the user administration by generating a new project and a super user in the database in order to manage privileges to these super users or users normal. These files user management are implicit in the Django admin folder.

ORM: Grails not only leverages Hibernate under the covers as its persistence layer, but also implements its own Object Relational Mapping layer for Groovy, known as GORM. The Grails Object Relational Mapping (GORM) is a core component of the Grails web application framework and uses DSL- style techniques to apply persistence to regular Groovy classes. In Fig. 4 a Grails-based Web application with ORM support is shown. This Web application describes a pet hospital where the veterinarian adds information about the owner, which has one or many pets and each pet can have zero, one or many visits to the hospital. HypersonicSQL is used which is the default database in Grails.

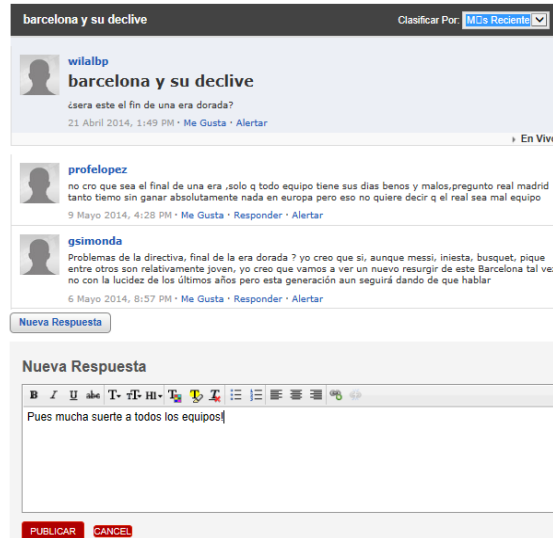


Fig. 3 Web Application Web using Comet with Grails



Fig. 4 Web Application using ORM with Grails

There are common problems with the development of Web-based projects such as: 1) developers frequently make the same mistakes in the development phase, which implies spend time and money, 2) spending huge amounts of time and money training new team members, 3) having difficulty with multi-person projects because each team member has his own way of doing things. In order to avoid these problems, this work intends to moti-

vate the use of Best Practices on Web development into the Software Engineering and Web engineering communities. As can be inferred, in this work best practices on Web development have been proposed and applied facilitating the development and improving the final product quality.

6 Conclusions and Future Work

We have presented a comparison of Web frameworks through of best practices in order to establish the best practices for Web development. The use of best practices allows developing better and more efficient Web applications. The results show that Django and Grails offer more features for developing Web applications than other frameworks. 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. Therefore best practices are very important in the software engineering community, due to the fact that such practices allow decreasing errors in the implementation phase.

As future directions, the authors are considering to obtain new best practices such as Caching, Testing, SiteMap and Wiring. Also, future research will include the comparison with other PHP-based Web frameworks such as Yii in order to provide a comprehensive analysis among them. This analysis will allow developers to choose the Web framework that more meet their requirements or the Web framework that has more advantages for a given project.

Acknowledgments

This work was supported by the General Council of Superior Technological Education of Mexico (DGEST). Additionally, this work was sponsored by the National Council of Science and Technology (CONACYT) and the Public Education Secretary (SEP) through PROMEP.

References

1. Ford, N.: Art of Java Web Development: Struts, Tapestry, Commons, Velocity, Junit, Axis, Cocoon, Internetbeans. Webwork. Manning Publications Co., Greenwich, CT, USA pp. 624, (2003)
2. Stout, G.: Testing a Website: Best practices. In: http://www.heromotocorp.com/sp-pe/uploads/Annual_Reports/pdf/20130315113443-pdf-68.pdf; Accessed May 2, 2014.
3. Hightower, R.; Jakarta Struts Live. Highlands Ranch, Colorado: SourceBeat, LLC, pp. 273, (2004)

4. Shenoy, S.; Mallya, N.: Struts Survival Guide: Basics to Best Practices. Austin TX: Objectsource LLC, pp. 221 (2004)
5. Danesh, M.H.; Raahemi, B.; Kamali, M.A.: A framework for process management in service oriented virtual organizations, Next Generation Web Services Practices (NWeSP), 2011. In Proceedings of the 7th International Conference on. DOI: 10.1109/NWeSP.2011.6088146, pp 12-17 (2011)
6. Kornstaedt, A.; Reisch, E.: Staying afloat in an expanding sea of choices: emerging best practices for eclipse rich client platform development. Software Engineering, 2010 ACM/IEEE 32nd International Conference on, Vol. 2. DOI: 10.1145/1810295.1810305, pp. 59-67, May (2010)
7. Jim, I.; Web Site Engineering Best Practice Standards (IEEE 2001). In Proceedings of the Fourth International Workshop on Web Site Evolution (WSE'02) (WSE '02). IEEE Computer Society, Washington, DC, USA, pp. 81, (2002)
8. Fisher, D.; Wisniewski, J.: Web Development Best Practices, Magazine Control-Shift, pp. 57-60. Feb (2012)
9. Xufeng, L.; Ginige, A.: Smart business objects for web applications: A new approach to model business objects. In First International Conference, ICSOFT 2006, Setúbal, Portugal, pp. 11-14. Sept (2006)
10. T. Richard.: Seven Pitfalls to Avoid in the Hunt for Best Practices. IEEE Software, Volume 20 Issue 1, pp. 67-69. Jan (2003)
11. Thompson, J.B.; Fox, A. J.: Best Practice: Is this the Cinderella Area of Software Engineering?. In Proceedings of the 18th Conference on Software Engineering Education & Training (CSEET'05), pp. 137-144. April (2005)
12. Nicolaou, A.: Best Practices on the Move: Building Web Apps for Mobile Devices, Magazine Communications of the ACM, Vol. 56, No.8, pp. 45-51. August (2013)
13. Mellone, J. T.; Williams, D. J.: Applying best practices in web site redesign: the Queens College Libraries experience, OCLC Systems & Services. Vol. 26 Issue: 3, pp.177-197, Dec (2010)
14. Soares, R.; Gomes, A.; Batista, L.; Atem, R.: A Multi-Layer Framework for Enterprise Application Development, Vol. 205, Research and Practical Issues of Enterprise Information Systems. pp. 285-296. April (2006)
15. Harrison, W.: Best Practices--Who Says?. IEEE Computer Society Press. Los Alamitos, CA, USA. Vol. 21 Issue 1. pp. 8-11. Jan (2004)
16. Gutiérrez, J.: ¿Qué es un framework Web?. Available in: http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf Accessed May 12, 2014
17. Shenoy, S.: Getting started. Struts Survival Guide, Ed. New York: ObjectSource LLC. pp. 25. (2005).
18. Horstmann, C.: JavaServerFaces 2.0, DZone Refcardz. PC Magazine, pp.1-6. (2009).
19. Curry, M.: Super awesome advanced CakePHP Tips, Ed. United States: creative commons. pp.10 (2009).
20. Dave, T.: Introduction to Ruby, Agile Web development with Rails. Pragmatic Bookshelf; Edition 1, pp. 247-248 (2005).
21. Ponce, T.; Villena, C.; Unoja, R.; Lopez, M.; Perez, Y.: Analysis of Web framework Yii, pp. 1-4, Sep (2012).
22. Kaplan, J.; Holovaty, A.: The Definitive Guide to Django: Web Development Done Right, Apress Berkely, CA, USA; 2 Edition, pp 144-196, (2009).
23. Wikimedia Foundation, Inc.: Java view technologies and frameworks. Available in: http://en.wikipedia.org/wiki/Java_view_technologies_and_frameworks Accessed Jul 20, 2014.

