An MVC Approach to Design Content Management System

Kibreab Gizatu
Clinton Health Access Initiative, Addis Ababa, Ethiopia
kibreabg@gmail.com

Nassir Dino
HiLCoE School of Computer Science and Technology, Addis Ababa, Ethiopia
nassir@hilcoe.com.et

Abstract

In the process of designing and implementing web applications, there exist many challenges. This work aims to identify those challenges and propose solution. Hence, a number of anticipated challenges that could be encountered on the process of developing web applications have been considered to illustrate the problem area of this paper.

The web application chosen for the case study is a content management system that will help us to realize the anticipated privileges of using MVC architecture style over candidate architecture styles like SOA.

The paper concentrates mainly on the candidate architectural aspects of web applications. Architecture Centric Design Methodology (ACDM) is followed to show the architectural aspects of web application. The web framework applied in this case study is Struts 2 and the design pattern is Model View Controller.

Keywords: Web Applications; Software Architecture; Design Pattern; Content Management System; Model View Controller; Web Framework

1. Introduction

The demand of software in the form of web applications is increasingly growing. The demand of business and technology services using web applications like online booking an airline ticket, hotel reservation, becoming part of social networking site, and so on are escalating. This increasing demand on using web applications brings about a strong focus on their efficient development. Therefore software practitioners should strive to produce high quality and standardized web applications that give much value to users.

Consequently, the main purpose of this paper is identifying the difficulties and challenges in designing web applications and proposing a solution for them. One of them is the proper choice of a software architecture style that best fits the web application. The architectural design decisions that we employ to develop our web applications need to be carefully examined to produce the most competent and efficient applications. These decisions directly affect the characteristics of the resulting system like maintainability, flexibility, reuse, agility, interoperability, and scalability. This results with alternative architectures that incorporate loose coupling of the system's modules, maintainability, flexibility, usability, security, and the like.

The proposal of this paper is that the above mentioned and other more challenges can be best taken care of and solved by using the Model View Controller (MVC) architecture. This architectural pattern is best suited for the design of web applications and is implementable by the various programming languages such as ASP.NET, PHP, Java and some more. MVC has now grown into a stage where it became a framework rather than just a design pattern. In this paper the intent is to show how a well-structured web application can be developed using Java's Struts web MVC framework. This will be demonstrated by using the approach of a case study and development of a working prototype of a Content Management System (CMS) web application.

The methodology used in order to verify critical problems and to validate the solution were interviews, literature review, case study, and prototyping. In the development process, everything that the architectural framework has benefited is recorded. The remaining part of the paper is organized as follows. Section 2 states the background focusing on the motivation. Section 3 is about the

solution. Section 4 is on Case study. Section 5 is on related work and Section 6 concludes the paper.

2. Background

The importance and necessity of web applications is well recognized by the country wide software industry practitioners, researchers, the academia and by concerned government agencies. This enlightens us to allot a considerable effort to better the development procedures of web applications. There are many challenges that developers face while in the process of developing web applications. As software developers, we have been experiencing many of these challenges. The following list shows the summary of the problems to be tackled and reasons for the initiation of this research:

- Various challenges of web application development emanate from the fact that the web continuously evolves due to the need for legacy systems integration, traffic handling issues, and existence of infrastructure variation over the World Wide Web and the globe.
- Developers face difficulties trying to convert the sophisticatedly designed architecture documents by the system architects.
- There are struggles on how to simplify various challenges in the development of web applications such as validation, error-handling, decoupling the system's modules, attractive user interfaces, etc. Because of these struggles, a need to automate these processes arises.
- A huge difficulty arises when trying to make web applications compatible to a number of technologies like browsers and the flexibility to implement user interfaces in any technology assumed best fit (like ASP.NET, JSP, Velocity, XLST).
- Simplifying the integration of language packs to web applications.
- The number of difficulties faced trying to test web applications.
- Usually too much effort is spent to promote code reuse.
- The need to automate procedures like database access and session management; because of their nature to be repetitively executed.
- Incorporating a well-designed security module and dealing with bandwidth usage issues for areas with slow Internet connection (which is usually tackled with caching).
- The need to facilitate team projects that would constitute modules which are easily broken down to member developers.

The ultimate aim of this paper is showing a good mechanism to tackle the above mentioned problems that are encountered during web application development efforts.

3. The Proposed Solution

The design of web applications is by far different from designing websites or other form-based applications. A website is a set of related web pages containing content (media), including text, video, music, audio, images, etc. [1]. Form-based applications, on the other hand, are smart clients that are graphically rich, easy to deploy and update, can work when they are connected to or disconnected from the Internet, and can access resources on the local computer in a secure manner [2]. Designing web applications is different from the above mentioned application types because the web brings about so many additional features that make development very difficult. It adds a lot more variables to consider in the process of application development.

A web application is actually a web site where user input, including navigation through the site, and data entry affects the state of the business. In this kind of application, developers should deal with issues like integrating the newly developed system with legacy system databases, handling the constantly evolving nature of the web, and handling the traffic caused by a large number of users. There are, of course, other essential points to consider such as the application's security and aesthetics. In this kind of environment, fast development and quick response time for user interaction are also necessary. The architecture of the whole system also depends upon the variety of hardware devices that are involved such as web servers, client computers, browsers, application servers, the file system, and the database servers. This forces the developer to think

architecture wise and come up with a structure that results in an efficient end product.

The general objective of this paper is suggesting an architectural solution that solves the problems mentioned and illustrating this by implementing a case study web application.

4. Experimentation/Prototype

The implementation of the CMS prototype for in this paper is taken care of using the following technologies.

- Eclipse Java EE IDE for Web Developers,
- Struts 2 MVC web framework,
- Hibernate Object Relational Mapping (ORM) framework, and
- MySQL relational database.

The project structure of the CMS prototype application is composed of several folders to represent the various components of an MVC application. Figure s shows how the project is structured inside Eclipse IDE. The project is entitled CMSAdmin. The main folders inside the project are located inside the Java Resources: src folder and inside WebContent folder. The package folders inside the Java Resources: src folder are named com.cmsAdmin.dao, com.cmsAdmin.domain, and com.cmsAdmin.web. The DAO folder contains data access objects that contain code that queries and writes to the database. The Domain folder contains the entity classes that represent the database tables of the project. The Web folder contains the action classes that serve as the Model in the MVC triad. The rest of the files in the source folder are xml files that serve for the various configurations of the Struts 2 application implementation.

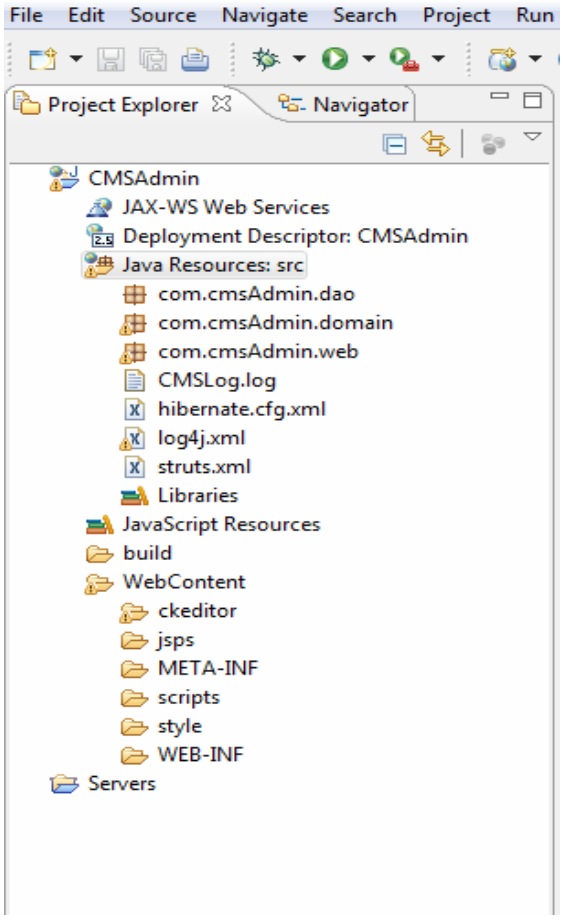

Figure 1: Project Structure inside Eclipse IDE

The other important folder is the WebContent folder. This folder contains most of the resources that relate to the View section of the MVC triads. It contains the JSP pages in the jsps folder, various scripting resources in the scripts folder, cascading style sheets (CSS) in the style folder, and web application configuration xml (web.xml) in the WEB-INF folder.

The very reason why the project is structured the way it is and the fact that the technologies mentioned are utilized is to facilitate the provision of solution to the problem statement.

5. Related Work

Over the course of research for this paper, it has been noticed that not much resource is available that concentrates on the architectural aspects of web applications. It was very difficult to find literature that emphasized upon web frameworks, web application architectures, and current application design best practices. The main sources of literature were books, articles on the Internet, blogs, and

forums. But this doesn't mean that there was absolutely nothing that was available and relating to the soul focus of this project.

There have been some works done to show the importance of careful architecting upon web applications to produce quality products as in [3, [4]. These works clearly show the current trend of web architectures and try to propose architectural solutions that best fit the demands. They show that web applications are more accessible and have lower maintenance and deployment costs than traditional systems. They classify web applications as brochure, service oriented, data intensive, and information systems web applications. These works also point out that web applications are under constant evolution because of the demands of the web and, therefore, require an architecture that best cops with these requirements.

Also Ziemer [5] in an essay for architecture of web applications pointed out that elaborating a sound architecture for a software system is highly crucial. It was also pointed out that the architecture chosen for designing a certain web application should take into consideration the fact that development of web applications is highly evolutional and ever changing.

As explained in [6], the need to incorporate architectural frameworks in the development of applications is becoming mandatory. The need is even doubled when it comes to web applications. The architectural frameworks available today solve much of the overhead involved through the development of web applications. They now come up with their own libraries for data access, session management, templating and promote code reuse. The article also points out that most application frameworks found these days are based on the Model-View-Controller (MVC) architectural pattern. The frameworks use this pattern for the main purpose of decoupling data model with business rules from the user interface. It continues to explain the use of the frameworks by pointing out that they take care of much overhead related to database access mapping, URL mapping, and even in some the provision of tools to create and provide web services.

According to Hennebrueder [7], a complete list of available frameworks found these days would exceed over 200. Over the top of the list are Apache Shale, JavaServer Faces, JBoss Seam, Stripes, Struts 2, Spring, Hibernate, Ajax libraries, JQuery, etc. This clearly shows there is a desperate need to standardize the development of web applications and change our development culture. We have to be able to choose the best architectural framework out there that fits into the kind of web application that we are developing and apply it.

As pointed out in [8], the web application platform has become very powerful and allows for developing all classes of applications that run over the web. Current web applications demand the functional feature set and look and feel of traditional desktop applications. The writer continues to suggest that the core workflow of these applications remained the same. Also a web framework helps developers to be relieved from the mundane concerns of the domain by the provision of reusable architectural solution to the core web application workflows.

6. Conclusion

The specific objectives of this paper were to:

- clearly show the advantages of integrating architectural frameworks in the development of web applications,
- point out the extra benefits of using the MVC framework to design the kind of web applications demanded these days by comparing it with other alternative architectures like Service Oriented-Architecture,
- investigate the requirements of the CMS web application that is going to be used for the case study, and
- test and evaluate the developed system to show that the whole procedure of the development has in fact proved to be a solution to the problems discussed earlier.

This paper attempted to show what challenges exist in the effort of developing web applications. The main purpose of the paper was to uncover what the challenges were and providing a sound method or an approach that solves them.

After the prototype application has been implemented, the findings suggest that through the process of developing the case study application (CMS), most of the challenges identified have been successfully solved. That means a sound approach to how those challenges can be eased up and tackled is identified by applying the web framework Struts 2 which implements the MVC architectural pattern. The rest of the challenges on the other hand could not be treated through the making of the application using Struts 2 framework and MVC architectural pattern. This was because the solution for these problems required a more sophisticated series of procedures that can't be visible enough just by implementing a single case web application. They required a little more study coverage of other areas such as the web's nature, browsers, security issues and the like. These remaining challenges are:

- Legacy system integration
- Handling infrastructure variation over the web
- Traffic handling over the web
- Browser compatibilities
- Security module integration
- Handling low bandwidth issues (caching)

It is the assumption and best hope of this paper that these problems be given some attention and get researched upon in future works. The fact of the matter is these challenges relating to web application design and implementation still exist.

References

[1] Wikipedia, the free encyclopedia, "Website", en.wikipedia.org/wiki/Website, Last-accessed-Mar 12, 2012.

[2] MSDN Library, "Windows Forms and Smart Client Applications", http://msdn.microsoft.com/en-us/library/8bxxy49h.aspx, Last-accessed: Mar 12, 2012.

[3] Roy T. F. and Richard N. T., "Principled Design of the Modern Web Architecture", ICSE 2000 Limerick, Ireland.

[4] Guntram G. and Martin G., "An Evolution-Oriented Architecture for Web Applications", In Second Nordic Workshop on Software Architecture, NOSA, 1999.

[5] Sven Ziemer, "An Architecture for Web Applications Essay in DIF 8914 Distributed Information Systems", November 28, 2002.

[6] Multiple (wiki), "Web application framework", Docforge, Retrieved 2010-01-19, Accessed on October 27, 2011.

[7] Sebastian Hennebrueder, "Choosing Web-Frameworks", http://www.laliluna.de/articles/posts/the-web-framework-evaluation.html, Accessed on October 27, 2011.

[8] Donald B. & Chad M.D., "Struts 2 In Action", 2007.