Proposal for DSpace Web MVC
ROBERT QIN ZHENGQUAN

Email: u0700323@nus.edu.sg

Short description: In my experiences of building enterprise applications (Tourist Portal and Video rental system) my JSP pages were often "peppered" with scriptlets containing certain business logic. I am interested in taking on this project because I have witnessed for myself how logic at the presentation layer could create tight coupling scenarios and obstruct the evolvement of the front layer coding, thus impairing the quality of my code.

Additional info: http://max.mailbigfile.com/7c426b328011d66b0630...

Email: u0700323@nus.edu.sg

Phone: +65 9297 1236

MSN: qin_zhengquan@hotmail.com

SKYPE: qin.zhengquan

Proposal for DSpace Web MVC

"DSpace Web MVC" By Robert Qin, 2011

Self description

I am a year 3 **Information Systems Student** specializing in **Service Science Systems** at the **National University of Singapore (NUS)**. I have previously been involved in two enterprise applications development projects.

1. **Online Tourist Portal**

2. **Online Video Rental Portal**

3. **Embedding of external SDK into iPhone/iPad apps**
Online Tourist Portal

In the online tourist portal project, my main scope of responsibilities involved the creation of a tourist map application that retrieves the various stakeholders' meta-data information and displays them via the use of the**Google Map API**. I was also involved in the creation of a tourist analytics platform that uses data-mining techniques to search through the various purchasing and customer demographics information and to identify various sales and customer behavioral patterns and present them to the end-user via the use of **JFree charts**.

Online Video Rental Portal

In the online video rental portal project, I took my enterprise application development education to the next level by embarking on the use of web services to encourage interoperability via the use of auto-generated WSDL and

SOAP XML contracts. My scope responsibility was to create a platform for users to upload videos, add inventory, performed sales analysis of the various inventory items and to allow customers to watch online videos.

Extracts of my source code are available at the following links:

Online Tourist Portal: http://max.mailbigfile.com/aeefd9149fd1a61fd248e59f1f01c2b6/listFiles.php

Online Video Rental Portal: http://max.mailbigfile.com/2d5a34530f1a7d5fa8aacc3fb6154219/listFiles.php

<u>Embedding of external SDK (Open source) into iPhone apps</u>

I had a one year internship stint with a software development company in Beijing, China. I was mainly involved in project management to coordinate the work responsibilities between the developers and designers in the deliveries of an iPhone application that is embedded with our company's SmarTots API.

Part of my responsibilities includes embedding the external API method calls (Local and Remote) into the developers' software code. I had to manually look through every developer's code to understand the specific portion to embed the required API calls.

The table below depicts my skills sets and competency level.

Technical Skill set	
Type	**Competency Level**
Java	**Advance**
HTML, CSS	**Beginner**
Javascript	**Intermediate**
JSP, Servlets	**Advance**
Google Maps API	**Beginner**
JFree charts	**Advance**
Enterprise Javabeans	**Advance**
Web services	**Intermediate**
XML	**Beginner**
C	**Beginner**
SQL	**Intermediate**
Development Environment Skill set	
Netbeans	**Advance**
Glassfish/Tomcat	**Advance**
Spring Web MVC Framework	**Beginner**
Freemarker	**None**
None Technical Skill set	
English (Written)	**Excellent**
English (Spoken)	**Excellent**
Chinese (Written)	**Average**
Chinese (Spoken)	**Average**
Presentation	**Excellent**

Reasons for joining this project

In my experiences of building enterprise applications (Tourist Portal and Video rental system) my JSP pages were often "peppered" with scriptlets containing certain business logic. For example, within the JSP page itself, my scriptlets would contain method calls, like (get methods) to ensure the required content is displayed on the page.

JSP pages should not contain too much logic as it creates tight coupling scenarios and can create obstruction to web design works. For example, the combination of javascript, scriptlet logic and html code could complicate web design development at the front end layer. Scriptlet logic should best be confined within the Servlet and JSP should mainly contain HTML code.

I am interested in taking on this project because I have witnessed for myself how logic at the presentation layer could create tight coupling scenarios and obstruct the evolvement of the front layer coding, thus impairing the quality of my code.

Above all, I wished to utilize the opportunity of the GSOC program to brush up on the quality of my software development skills by working with expert developers in the industry to develop real world applications. I strongly believe the experience gained from the GSOC program would put me in good standing with my future employers when I transit to the workforce upon graduation from NUS.

Project Proposal

Problem Statement: Separation of processing logic from presentation layer at the Web UI component in the DSpace application layer.

Problem Elaboration:
Although the DSpace three-tier architecture follows the model-view-controller (MVC) architecture, with most of the logic encapsulated in the logic layer, there is still a fair amount of processing logic involved in the Web UI component at the application layer. For example, in the "**full.jsp**" file under the browse folder of DSpace JSP-UI, there is a thick chunk of scriplet logic embedded from lines **30** to **164**.

Solution:

The current proposal is to adopt the Spring Web MVC framework as a means to enforce the MVC framework architecture and the use of the Freemarker template engine to restrict the amount of logic at the presentation layer.

To apply it to the DSpace context, for example, the amount of scriptlet logic in the "**full.jsp**" file could be drastically reduced if they were placed within the **Controller** component. For example, all the string variables could have been instantiated within Servlets and have their values defined there using the various**getMethods()** from the business logic layer.

Once the values have been defined, we can then call **request.setAttribute("value", value)** within the Servlet to have all the String values passed into an object. These objects can then be retrieved using the**request.getAttribute** method in the JSP page. The current approach seems to contain the **request.setAttribute** method call within the JSP pages as well.

The use of the Spring Web MVC, with its dependency injection could also create loose coupling amongst the logic layer and perhaps improve performance as a result.

Breakdown of goals within MVC framework

Views: To ensure that JavaServer Pages (JSP) are strictly involved only in presentation layer and does not contain processing logic.

Controller: To ensure that the HTTP request and response logic processing are handled by the specific Servlets.

Model: To ensure that complex logic algorithms computation is handled entirely at the business logic layer. For example, session beans, web services.

Proposal Timeline

Before April 20:

- Installation of DSpace with netbeans
- Initiate contact with the Mentor to confirm the specific milestones required in the project development
- Initiate contact with the Mentor on the specific skill sets required for this project
- To code a complete Spring Web MVC application with Netbeans
- To read up the DSpace API documentation

April 20 – May 23 (Before the official coding time):

- To remain in contact with the mentor via email and IRC and to clarify existing doubts in the process of learning and discovery about DSpace.
- Continue reading up on DSpace API documentation, especially the specific required components
- To re-model previous enterprise applications based on the Spring MVC framework
- Using of FreeMarker engine for presentation layer code generation

May 23 – June 18 (Official coding period starts):

- Understanding the project objectives of the first phase deliverables
- Coding on the first phase deliverables
- Recoding all logic from the JSP (Method call handlers) into relevant Servlets.
- System documentation
- Reporting of any system bugs and errors on https://jira.duraspace.org/browse/DWMVC
- Weekly scrum with mentor on progress statuses including clarification on any existing doubts

June 18 – July 5:

- Progressive unit testing on the previous milestone deliverable
- Post implementation review of the previous milestone deliverable
- Coding on the next mile stone deliverable
- System documentation
- Reporting of any system bugs and errors on https://jira.duraspace.org/browse/DWMVC
- Weekly scrum with mentor on progress statuses including clarification on any existing doubts

JULY 6th MID TERM EVALUATION

July 6 – July 15:

- To prepare a presentation report to update developers on the developed functionalities, exceptions and bugs encountered
- Make further changes in the code to improve the Functionality, Exception handling, Bug Removal.
- To maintain contact with the developers and to update them on the progress
- Consultation with the mentor on the next phase of milestones deliverables
- Coding on the second phase deliverable
- Phase 2 system documentation

July 15 – July 25:

- To be in constant touch with the developers and to let them know about our progress.
- Progressive unit testing on the previous milestone deliverable
- Post implementation review of the previous milestone deliverable
- Coding on the next mile stone deliverable
- Most of the time will be consumed for rigorous testing and bug fixes.
- Coding on phase three deliverables
- Phase 3 system documentation

July 25 – July 31:

- Post implementation review

A Buffer of two weeks has been kept for any unpredictable delay.

Email: u0700323@nus.edu.sg

Phone: +65 9297 1236

MSN: qin_zhengquan@hotmail.com
SKYPE: qin.zhengquan