WEB BASED IDE FOR INTERFACING VIEW-CONTROLLER

A Writing Project
Presented to
The Faculty of the Department of Computer Science
San Jose State University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science

Advisor
Dr. Chris Pollett

By
Tejasvi Palvai
May 2010

ABSTRACT

WEB BASED IDE FOR INTERFACING VIEW CONTROLLER

The main idea of this project is to develop a web-based IDE that enables users to create XHTML pages using the drag and drop mechanism in Struts. Struts extends Java Servlet API to encourage developers to adopt the MVC (Model-View-Controller) architecture. I will be developing a web application for interfacing the view and controller parts of the MVC architecture. This web-based IDE connects the View and Controller components when the user drags and drops elements in the UI of the IDE. With this project, users can create dynamic web pages without having any knowledge in XHTML or SQL queries.

Table of Contents:

Introduction:

An IDE (Integrated development environment) is a software application that provides comprehensive facilities like a source code editor, compiler, and a debugger for software development. There are many IDEs available for programmers, such as Eclipse, Net beans, and JDeveloper. However, these IDEs are desktop-based applications.

A Web based IDE is an online programming environment with a file manager and a console interface for compiling, executing and it also supports many languages.

The main advantages of Web based IDE when compared to the a desktop IDEs are:

i) Instant Development and Code Testing: web-based ide avoids installing or configuring any desktop IDE, thus users can begin to write any code without wasting time in environment setting or configurations.

ii) Access the Code from anywhere: Since web-based ide is an online service, it can be accessed from anywhere.

iii) Layout: Web applications tend to be sparser in information density, using white space, font changes, and other layout techniques to guide the user in certain focused directions.

iv) Increases visual sophistication: Web users tend to expect a higher degree of visual sophistication from their applications. Animation, transparency, even sound cues are fairly common.

An example of a web-based IDE is Bespin. Bespin is a web-based integrated development environment (IDE) that allows developers to collaborate on code-based projects. Bespin features a very fast code editor with syntax highlighting

and access to frequently used actions via the Command Bar. Work can be resumed from any browser at any time without the need to save.

The web- based IDE I am developing is similar to Bespin but has the drag and drop functionality. This project will be built using the Struts framework and JQuery JavaScript With this web-based IDE, users can create tables and elements using the drag and drop functionality. These elements can be made dynamic when connected to the controller components. When created, these elements get stored in the database and allow users to perform various SQL operations. Without any prior knowledge of XHTML and SQL/server side languages, one can create dynamic web pages.

Struts is an open source web application framework developed as an Apache Jakarta Framework that uses the MVC architecture. MVC is considered an architectural pattern used in software engineering. It separates the business logic and application data from the presentation data to the user. The Model is responsible for providing the data from the database and storing it into database. View represents the user view of the applications and is responsible for taking the input from the user; whereas, the Controller is the intermediary between Model and View. Controller is responsible for receiving the response from the client.

.

Deliverable 1) Demo Application Using Struts:

Aim: The main aim of this deliverable was to learn Struts and develop a demo application using the same. This is the basic demo project implemented using struts to understand the MVC architecture. This helps in understanding the functionality of the web-based IDE.

Implementation:

In Struts, first we have to write an ActionForm that maintains the session state web application. ActionForm in this project is LoginForm.java. This ActionForm extends the ActionForm class. LoginForm takes the input from the user and validates the form. The Validate () method is used to validate the inputs. If any of the fields is left empty or entered incorrectly, error messages are added to the ActionMapping object. The ActionForm object is automatically populated on the server side with data entered from a form on the client side. It must have setters and getters for all the form fields. The ActionForm class is pre-populated with form data when the form is submitted.

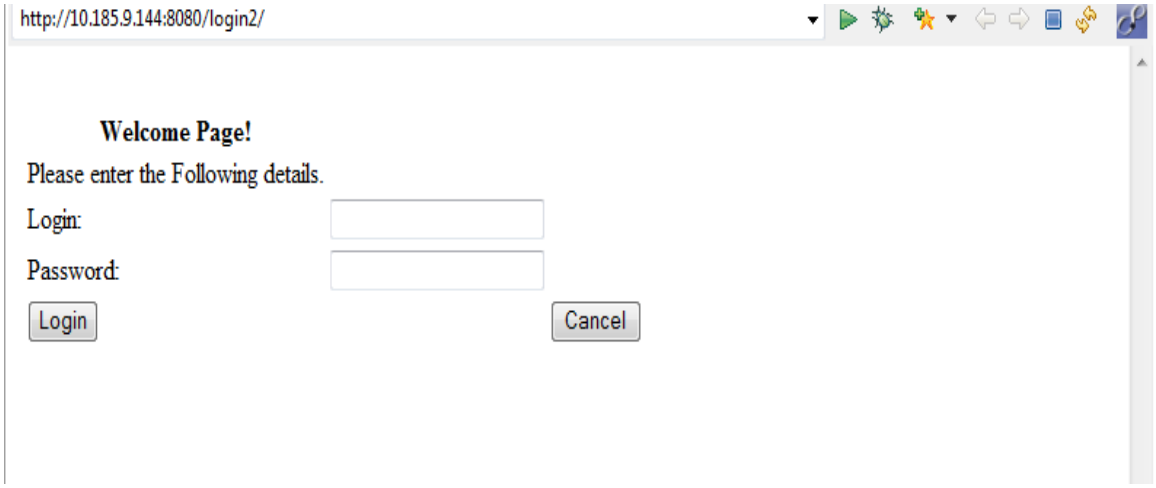
Login Action class (Login action. java) is the Controller part of the application. Action class extends the base class org.apache.struts.action.Action. It is the center of all the Action. It processes the specified HTTP request, creates the corresponding HTTP response, and returns an Action Forward instance describing where and how the control should be forwarded. It forwards the control to the view chosen to display the result of action i.e. to "success.jsp" here.

View Component:

The view component usually is made up of JSP (Java Server Pages). JSP can contain the static HTML. In this deliverable the JSP pages are login.jsp success.jsp, failure.jsp. Login.jsp is our form for entering the login details like username/login ID and password. User enter the values in the form and

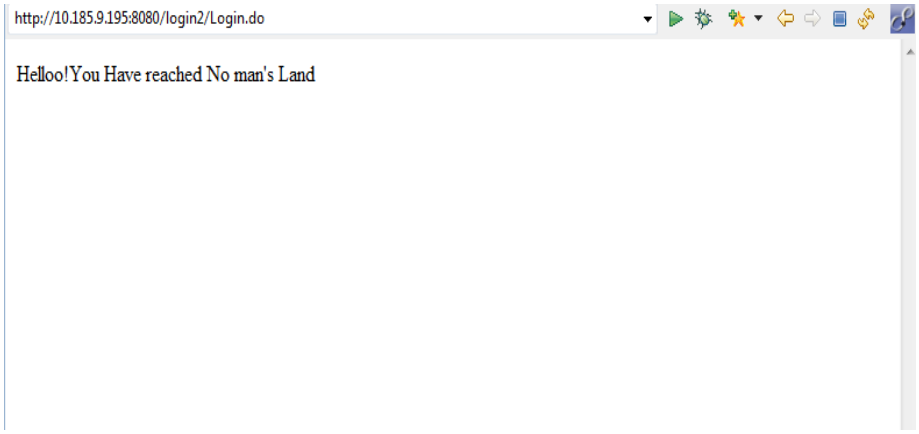
will click on the submit form. Form validation is done on the server side and error message will be displayed on the jsp page. To display the error messages on the jsp page **<html:errors/>** tag is used. The **<html:errors/>** tag displays all the errors in one go. The following figure is the Welcome page.

Figure i): Welcome page



If any of the columns are left empty or entered incorrectly, then the user will be redirected to failure.jsp, which prompts the user to enter the Login details again correctly. If all the details are entered correctly, it will be redirected to the success.jsp.The following figure shows the Success page.

Figure 2: Success page



Deliverable 2: Performance Testing of Java Script Frameworks:

Aim: The main aim of this deliverable was to test the performance of different JavaScript frameworks and select the best framework for this project.

The different JavaScripts available are:

i) JQuery
ii) YUI (Yahoo User Interface)
iii) Dojo
iv) GWT (Google Web toolkit)
v) Echo3
vi) MooTools
vi) Prototype.

Though there are many JavaScripts available, I chose JQuery, YUI and DOJO to do the performance Testing.

JQuery: JQuery is very easy to use and understand. The Core Library is only 15Kb in size. It is fast and has concise JavaScript library that simplifies HTML document traversing, event handling, animating, and Ajax interaction for rapid web development. It is designed to change the way you write JavaScript.

YUI: YUI is fully documented with a great API browser. It is usually very consistent and contains unit test performance but has a very few utility or helper functions/Methods. It is usually heavy and lacks the use of chaining methods together.

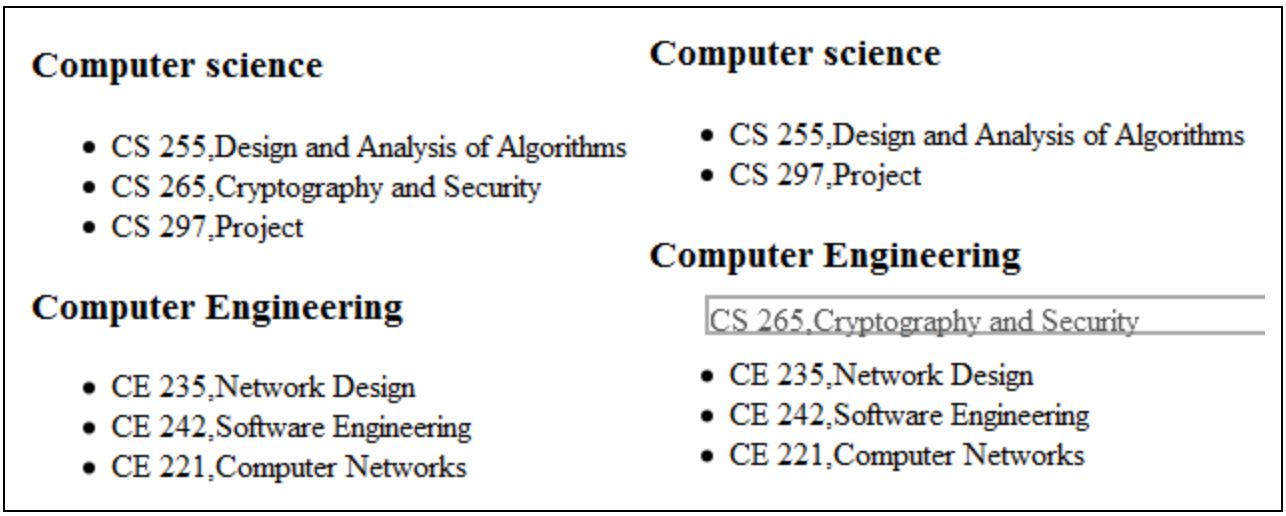
DOJO: Dojo is an open source modular JavaScript library designed to ease the rapid development of cross platform, JavaScript applications, and web sites. Dojo also provides many widgets to develop the UI for web applications. It is one of the robust ajax frameworks that can be used to develop enterprise grade application.

Implementation

The Project below shows two departments (Computer Science & Computer Engineering). One can drag and drop courses from one department to another. In the table, the CS 265 course from Computer Science department is being dragged and dropped into the Computer Engineering department. In the same way courses from CE can also be drag and dropped into the CS department.

The Following figure shows the drag and drop functionality. It is implemented using JQuery, YUI and Dojo

Figure: iii): Drag and Drop example using Java Scripts



Using Drag and drop Example, the three frameworks were tested and following are the results.

- Memory Testing: YUI uses more memory when compared to JQuery and DOJO .For the example shown above, YUI took around 78k and JQuery used 65 K of memory. (This was tested using task manager in windows).
- JQuery performs well in almost all the browsers and also has highly effective and short code. It was very fast in firefox where as YUI was slow compared to JQuery and DOJO

- Dojo is not compatible in Internet Explorer 8, Mozilla Firefox 3.1, Beta2 and Safari 3.2.1

Following are the test results Using Slick speed.

Figure iv) Test results using Slick speed.

selectors	MooTools 1.2	JQuery 1.2.6	Prototype 1.6.0.2	YUI 2.5.2 Selector *beta*	Dojo 1.1.1
body	1 ms | 1 found	1 ms | 1 found	12 ms | 1 found	1 ms | 1 found	2 ms | 1 found
div	2 ms | 51 found	1 ms | 51 found	9 ms | 51 found	1 ms | 51 found	2 ms | 51 found
body div	2 ms | 51 found	2 ms | 51 found	18 ms | 51 found	2 ms | 51 found	2 ms | 51 found
div p	4 ms | 140 found	5 ms | 140 found	13 ms | 140 found	11 ms | 140 found	2 ms | 140 found
div > p	3 ms | 134 found	5 ms | 134 found	11 ms | 134 found	4 ms | 134 found	1 ms | 134 found
div + p	2 ms | 22 found	3 ms | 22 found	12 ms | 22 found	8 ms | 22 found	2 ms | 22 found
div ~ p	10 ms | 183 found	11 ms | 183 found	43 ms | 183 found	8 ms | 183 found	3 ms | 183 found
div[class^=exa][class$=mple]	3 ms | 43 found	2 ms | 43 found	10 ms | 43 found	2 ms | 43 found	2 ms | 43 found
div p a	4 ms | 12 found	5 ms | 12 found	13 ms | 12 found	19 ms | 12 found	2 ms | 12 found
div, p, a	9 ms | 671 found	11 ms | 671 found	33 ms | 671 found	13 ms | 671 found	2 ms | 671 found
.note	16 ms | 14 found	15 ms | 14 found	8 ms | 14 found	36 ms | 14 found	1 ms | 14 found
div.example	2 ms | 43 found	1 ms | 43 found	10 ms | 43 found	2 ms | 43 found	2 ms | 43 found

In many browsers and for some of the selectors tests, Dojo Query and YUI did not work or not compatible, where as **JQuery passed them all.** Hence JQuery is considered as the best among the existing frameworks.

After considering the above test results, I decided to use JQuery JavaScript for my project.

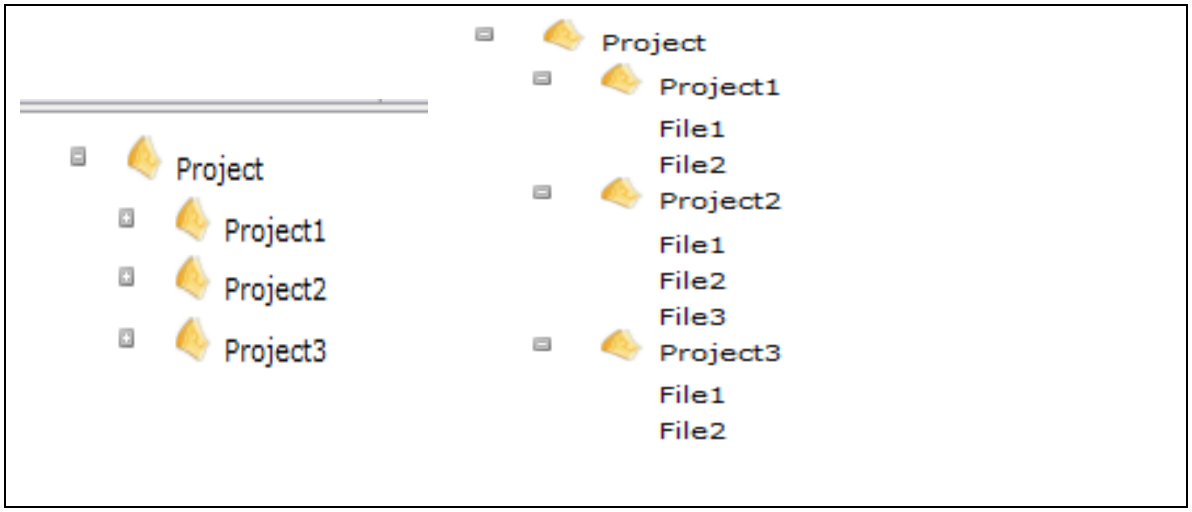
Deliverable 3: File system for IDE

Aim: The main aim of this deliverable was to develop the file system that is required for developing the web based IDE.

Implementation: In this deliverable, file system for the IDE was developed. The file system has many project folders; where in each folder has many files in it. Though dragging and dropping create these files, this particular deliverable is only concentrating on the layout of the file system.

When clicked on the plus symbol beside each folder, the files in each project will be displayed and the image beside the folder turns to minus symbol indicating that all the contents in a folder are displayed. After creating an element, it can be dragged into a project folder and file would be added into the folder.

Figure v) File System in IDE:



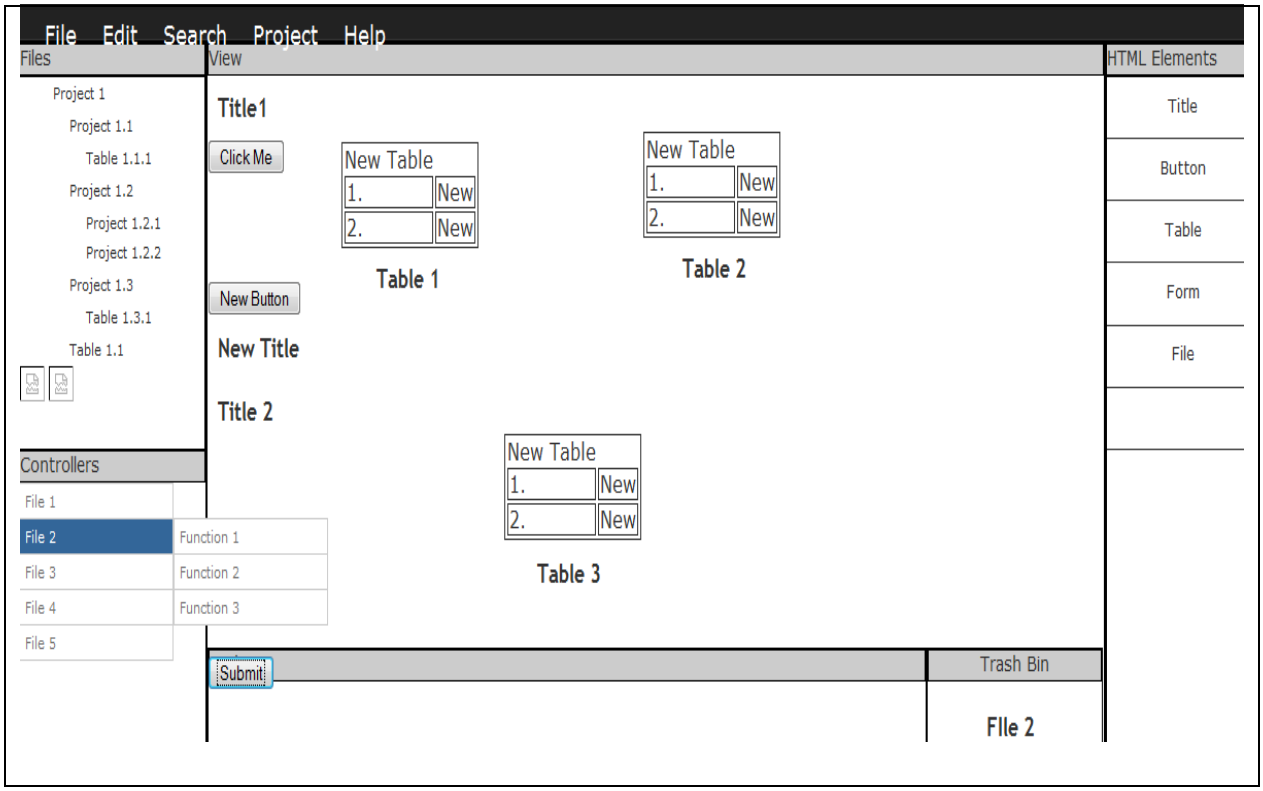
File	
Edit	New
Search	Open File
Project	Save
Help	Save as

This is implemented using JQuery JavaScript framework.

Deliverable 4 : Demo project involving drag and drop :

Aim: The main aim of this deliverable was to create a simple HTML page that demonstrates the drag-and-drop feature of JQuery framework. The view part of the final project looks more or less the similar way as shown in figure VI.

Figure vi: View part of the IDE



Implementation:

The Layout developed includes all the basic functionality of the drop downs and the drag and drops. The complete layout of the IDE is mainly categorized as the following:

a) Horizontal Navigation Bar

b) File system

   i) Project div.

   ii) Controller div

c) View

   i) Page view

   ii) Code view

d) HTML tool bar

The Horizontal navigation bar is the one, similar to that of any IDE. The bar includes file, edit, search, project, and help. Each, in turn, has its own drop down of options.

The left-most part of the IDE has files and controllers. Files consist of project folders and tables. When a new table is created by drag and drop, it will be added to a project folder. New files are also created in a similar way as tables.

Controllers have different files (File 1 and File 2.) and each file has different functions. These functions can be dragged and dropped on to the canvas, and when a new function is created; it gets added to that particular file. All the functions inside a file can be viewed by placing the mouse on the file. This can be achieved by using the method "onmousever ()".

The right-most part of the IDE has HTML elements. Users can create these Html elements without having any prior knowledge in XHTML. This IDE has various elements like title, button, table and file. When you create a new file or a table by dragging the element, it prompts the user to give it a new name. The element gets created after the user gives it a; even if the user does not gives a name; it gets created with the default name. If it is a function, it can be dragged and dropped into the controller file; if it is a file or table, it can be placed in the project folders

File	
Edit	New
Search	Open File
Project	Save
Help	Save as

The top part of the IDE has a navigation tool bar. It is a horizontal and standard menu bar. The top-level menu items are next to each other from left to right, and all other submenu items are stacked vertically. New files can be created and deleted. Users can paste contents from one file to another. This project will be used for the future work in implementing my final web-based IDE

Conclusion :

The four deliverables I worked in this semester for CS 297 helped me in understanding the concepts like Struts, MVC architecture, web-based IDEs in detail. The basic layout for the IDE has been developed and also the framework required for implementing this project was selected by conducting the performance tests on different JavaScripts.These deliverables laid the solid foundation, which will help me in implementing my final project in my next semester.

In CS 298, I will extend the present project and develop a complete web-based IDE in Fall 2010. I will be developing the drag and drop functionalities for IDE elements and also the functionality to interface the view and controller components of the MVC architecture.

References.

[1]Implementing MVC Architecture Using Struts http://www.oracle.com/technology/sample_code/tech/java/j2ee/jintdemo/tutorials/Struts.html

[2] Struts tutorial by Benmira free http://benmira.free.fr/en/j2ee/struts3.htm

[3] Bespin Web based IDE http://www.rotorcreative.com/interactive_bespin.php#

[4] ECCO – A Web Based IDE inside your Browser http://ecco.sourceforge.net/

[5]JavaScript Frameworks:http://blog.creonfx.com/javascript/mootools-vs-jquery-vs-prototype-vs-yui-vs-dojo-comparison-revised

[6] YUI drag and drop http://developer.yahoo.com/yui/dragdrop/.