Software Reusability and Web Application Development

Naveen Kumar
Research Scholar, Department of Computer Science & Applications, Chaudhary Devi Lal University, Sirsa-125055
Email: nkjaglan@gmail.com
Girdhar Gopal
Research Scholar, Department of Computer Science & Applications, Kurukshetra University, Kurukshetra-136119
Email: girdhar.gopal@gmail.com
Dilbag Singh
Associate Professor, Department of Computer Science & Applications, Chaudhary Devi Lal University, Sirsa- 125055

----------ABSTRACT----------

The Web engineering is a discipline to develop web based applications. Web applications comprise a very broad area of applications. Many problems occur in development and management of web applications. Some of these problems are discussed in this paper like information storage and code repetition. The solutions to these problems like routing, user controls etc. are also discussed. The solutions described in this paper provide a managed way to tackle the web application contents and make easier updating process for applications.

Keywords: **Master Page, Module, User Control, Web Engineering.**

1. INTRODUCTION

Web engineering is a approach to development of web pages. Scripting languages and client-side technologies are used by web developer to provide efficient experiences to the users. Due to very short time- to-market and fast pace of technology development, reuse is often not one of the primary concerns. Developers encounter problems that have already been solved, when developers designing new applications and web pages. Rather than re-inventing the wheel, or spending time componentizing the already available solution, they resort to pragmatic reuse.

The web development includes web engineering and user interface design. Web engineering is defined as, *"Web Engineering is the application of systematic and quantifiable approaches (concepts, methods, techniques, tools) to cost-effective requirements analysis, design, implementation, testing, operation, and maintenance of high-quality Web applications. Web Engineering is also the scientific discipline concerned with the study of these approaches." [9]*

This paper is organized in following sections. Section II includes problem formulation related to web engineering and re-usability. In section III, related work about various approaches and tools regarding re-usability of codes in web development is provided. Some solutions for re-usability in web development like Master Pages, User Controls are discussed in Section IV. The conclusion is discussed in section V.

2. PROBLEM FORMULATION

While developing the web pages all information stored in a single component or in other words all information related to website is stored at the same folder location. This can cause a lot of confusion and errors while updating the components of web pages.

These types of errors can be minimized using modules.

But use of modules can again cause a problem that URL shows the path where the components of website stored on the web server.

Another problem during designing of web-site is the repetition of code, which consumes a lot of additional time and space unnecessarily.

3. RELATED WORK

Reusability of code is supported by a lot of approaches and designing in web engineering. These include Hunter-gatherer [5], Internet Scrap-book [6], HTMLviewPad [7], and ReWeb [8]; while in the more general domain of reusing Java code there is G & P (Gilligan and Procrustes). [1] Hunter-gatherer and Internet Scrapbook allow users to collect components from within Web pages, and to collect components from different Web pages into a newly created page. But since these approaches were developed in 1990's and early 2000, with the term "component" they refer to information components – most usually text paragraphs. These approaches are mostly used to create scrapbooks of data gathered from different web pages, and not to reuse web page controls [2].

This work is also related to program slicing, where by starting from a subset of a program's behavior, the program is reduced to a minimal form which still produces that behavior. This approach can be viewed as web application slicing with the goal of reducing the whole application (along with its code and resources) to a form in which only the visuals, the behavior, and the necessary resources of the selected control are maintained. In the web engineering domain Tonella and Ricca [8] define web application slicing as a process which results in a portion of the web application which exhibits the same behavior as the initial web application in terms of information of interest displayed to the user. In the same work they present a technique for web application slicing in the presence of dynamic code generation where they show how to build a system dependency graph for web applications. This work is dealing mostly with slicing PHP web applications, and was done when web applications were a lot less dynamic on the client side, and when OO design and AJAX applications were not yet widely spread. While still being relevant, a lot has changed in the web development practices and this algorithm needs improvements.

4. PROPOSED WORK

4.1 Modules

Set of standard independent instructions can be used to create a complex structure, such as an item of building or furniture. It allows the user for dynamic modification environment shown in Figure 1. While developing a new application, putting all information in a single component is not a good practice. This can cause an error while updating the components of web pages. These types of errors can be minimized with the help of modules.

But use of modules can again cause a problem that URL shows the path where the components of website stored on the web server. For solving this problem routing is used. Routing can be used as below:

```
<aside style="clear: both ;">
 <h4> About DCSA </h4>
  <ul>
```

  About the Department
  Staff Council
  Department Research Committee
  Map & Locations

</aside>

The above routing must be defined in the global file (global.axax) as below

```
<script runat="server">
   void Application_Start(object sender, EventArgs e)
{
      RegisterRoutes(RouteTable.Routes);
   }
   static void RegisterRoutes(RouteCollection routes)
{
      routes.MapPageRoute("Home", "Home",
"~/Default.aspx");
      routes.MapPageRoute("AboutDCSA",
"AboutDCSA",
"~/Default_Pages/AboutTheDepartment.aspx");
routes.MapPageRoute("DRC", "DRC",
"~/Default_Pages/DepartmentResearchCommittee.a
spx");
      routes.MapPageRoute("MAP", "MAP",
"~/Default_Pages/MapAndLocations.aspx");
      routes.MapPageRoute("StaffCouncil",
"StaffCouncil",
"~/Default_Pages/StaffCouncil.aspx");
   }
```

Figure 1: Diving an application in modules [4]

4.2 Master Pages

To create a consistent behavior and look for all pages or group of pages in a web application, Master Pages are used. It provides a template for other pages, with shared layout and functionality. It defines placeholders for the content, which can be overridden by child pages. The output result is a combination of the master page and the content page (child page). The content pages contain the content you want to display. The layout of the master page and the content of the child page are combined to produce the output when user request for content page.

Run time behavior of Master Page

I. The URL entered by the user for requesting of the page.
II. @Page instruction is read. If the instruction references a master page, the master page is read as well. If the pages have been requested first time, both pages are compiled.
III. Updated content of Master page are merged into content of child Page.
IV. The content of individual content controls is merged into the corresponding Content Place Holder control in the master page.
V. The resulting merged page is rendered to the browser.

From the user point of view, the combination or merged master and content pages are single page.

There will not any URL of master page so the URL of the page is that of the content page.

4.3 User Controls

Another solution to avoid repetition of code is User Controls, Identify the repeated code and create the User Control. It is used as a module. The architecture of User Control can be summarized in the following Figure 2.

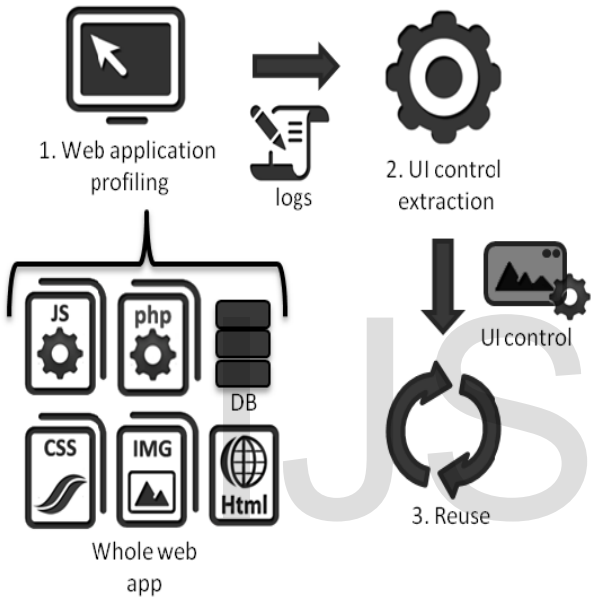

Figure 2: Reusing UI controls in Web applications [3]

Three steps are involved in User Control Development and use. Firstly the Web application profiling is developed then UI control will extract the required resources and code which will be reused in the third step. Web application profiling is to identify the set of repeated code. There is not a specific test to identify the repetition of code. The best way is that developer should record the repetition of code at the time of development. It will help the developer to find the eligible code for the User Controls.

Once all the User Controls have been developed, they can be reused at any place in the web application as a standalone module. Whenever the browser encounter the User Control it will fetch the corresponding compiled code from the server and paste it at the client side code. This code compilation and execution happens as the non transparent mode from the user. So the user will not be able to see any of these activities and hence is not able to differentiate between User Control code and other code.

User Controls can be developed as below:

<%@ Control Language="C#"
AutoEventWireup="true"
CodeFile="LeftResearch.ascx.cs"
Inherits="UserControls_LeftResearch" %>

The above User Control must be defined in the configuration file (Web.config) as below:

<add tagPrefix="ucLeftResearch"
tagName="LeftAsideResearch"
src="~/UserControls/LeftResearch.ascx"/>

This is followed by the definition of the User Control which consists of the code statements. In normal pages the User Control can be called as below:

<ucLeftResearch:LeftAsideResearch
ID="ULeftResearch" runat="server" />

5. Conclusion

Software reuse plays an important role in web development. In software development code is reused most of the times instead of development from scratch. Some time it makes the code difficult as small change in code needs lot of attention due to pasting at various places. In web development re-usability concerns are more important in comparison to stand alone application development as most the time. The user interface is almost similar for the

many web applications. There are various solutions available in literature to handle this problem. Some of them are discussed in paper like Master Pages, User Controls. Master page is used to create a consistent behavior and look for all pages or group of pages in a web application. User Controls are used for modules development, they are created and configured once for the application and then be reused any number of times in the application. These solutions are available in different technologies in various scenarios to be used in web development.

REFERENCES

[1] Holmes R., and Walker R.J., "Semi-Automating Pragmatic Reuse Tasks." In ASE '08: Proceedings of the 2008, 23rd IEEE/ACM, International Conference on Automated Software Engineering, pages 481–482. IEEE Computer Society, 2008.

[2] Krueger C. W., "Software reuse:, ACM Computer Surveys", 24(2):131–183, 1992.

[3] Maras Josip, "Automating Reuse In Web Application Development" ISBN 978-91-7485-140-3, ISSN 1651-4238, 7 April 2014.

[4] Modules, 12-02-2016, (https://www.google.co.in/?gws_rd=ssl#q=modules)

[5] Schraefel M., Zhu Y., Modjeska D., Wigdor D., and Zhao S., "Hunter Gatherer: Interaction Support for the Creation and Management of Within-Web-Page Collections" In 11th international conference on World Wide Web, pages 172–181, 2002.

[6] Sugiura A. and Koseki Y., "Internet scrapbook: creating personalized world wide web pages", In CHI '97: Extended abstracts on Human factors in computing systems, pages 343–344. ACM, 1997.

[7] Tanaka Y., Ito K., and Fujima J., "Meme Media for Clipping and Combining Web Resources", World Wide Web, 9:117–142, 2006.

[8] Tonella P. and Ricca F., "Web Application Slicing in Presence of Dynamic Code Generation. Automated Software Engg.", Vol-12(2), pp 259–288, 2005.

[9] Web Engineering, 02-02-2016, (http://webengg.blogspot.in/2010/05/definition-web-engineering.html)