ISSN:0975-9646

Cross Tier Framework for Rapid Web Application Development

Avinash D'Silva[1], Majesly Correia[2], Pruthvish Vyas[3]

Department of Computer Engineering
Xavier Institute of Engineering, Mumbai, India

Abstract— **A web framework enables the developers to easily integrate complex components into their web application, which otherwise would have been a task of writing the code from scratch. X-Frame creates web applications on fly using a code generator that eliminates redundant client side scripts/code. The MVC structure used in our framework enables proper structuring of the developer's web application. This paper describes how our framework can be used to build a well structured web application using various in-built functionalities.**

Keywords— **MVC architecture, ORM, AJAX, Code Generator, Hot Code Push.**

INTRODUCTION

Web applications like online shopping websites, web portals, and chat websites form a significant part of modern day life. A developer has to write a lot of code for creating a modern web application this result in redundancy in the code base, but this redundancy cannot be eliminated because each function in the code base is slightly different from the other. Another issue that developers face is that they tend to write a lot of code from scratch. Security is one of the biggest concerns of any web applications.

X-Frame is a framework for rapid web application development which reduces the development time by using a code generator thereby eliminating the redundant client side scripts/code. It provides the user to create web applications using the inbuilt functions. The framework provides libraries for most common functionalities.

The framework is built on Model View Controller architecture which allows easier maintenance of the code base. The hot code push feature resolves the issue related to up gradation for live deployment which will guarantee zero downtime during the upgrade process. A lot of time is spent in writing SQL queries; ORM (Object Relational Mapping) feature simplifies the whole process, to make coding easier and faster.

I. METHODOLOGY

A. MVC Architecture

Model View Controller architecture separates the application object (model) from the way it is represented to the user (view) from the way in which the user controls it (controller).

The model object knows about all the data that needs to be displayed. It also knows about all the operations that can be applied to transform that object. However, it know nothing whatever about the GUI, the manner in which the data are to be displayed, nor the GUI actions that are used to manipulate the data.

The view object refers to the model. It uses the query methods of the model to obtain data from the model and then displays the information. A view renders the contents of a model. It is the view's responsibility to maintain consistency in its presentation when the model changes. X-Frame uses MVC2 architecture.

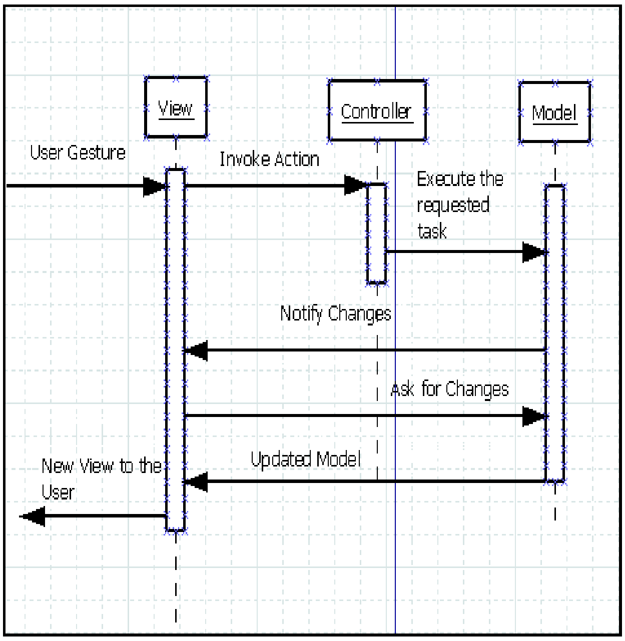

Fig.1 Sequence diagram for MVC architecture

B. ORM

In object-oriented programming, data management tasks act on object-oriented (OO) objects that are almost always non-scalar values. For example, consider phone number entry that represents a single person along with zero or more phone numbers. This could be modelled in an object-oriented implementation by a "Person object" with attributes/fields to hold each data item that the entry comprises: the person's name, a list of phone numbers, and a list of addresses.

The list of phone numbers would itself contain "PhoneNumber objects" and so on. The address book entry is treated as a single object by the programming language. Various methods can be associated with the object, such as a method to return the preferred phone number, the object, such as a method to return the preferred phone number.

However, many popular database products such as structured query language database management systems (SQL DBMS) can only store and manipulate scalar values such as integers and strings organized within tables. The programmer must both convert the object values into group of simpler values for storage in the database (and convert them back upon retrieval), or only use simple scalar values within the program. ORM (Object Relational Mapping) is used to implement the first approach.

ORM manages the translation of objects into relational databases, and vice-versa. It makes easier to adopt good database design and enables familiar object oriented concepts for manipulating data. In our framework, we have used RedBeanPHP to do the Object Relational Mapping.

C. Built-in AJAX

AJAX is the technology by which we can create web sites using dynamic features. AJAX stands for Asynchronous JavaScript and XML. X-Frame uses JSON (JavaScript Object Notation) instead of XML due to its light weight nature. The AJAX-style of development is used in websites like Google, Amazon, and Flicker. AJAX allows pages to request small bits of information from the server instead of entire pages. This incremental updating of pages eliminates the page refresh problem and slow response that have plagued Web applications since their inception.

Benefits of AJAX:

- User can use standard web browsers like I.E, Mozilla, Firefox, Netscape etc.
- User is not forced to wait till the whole page is loaded or refreshed.
- Even in case of an error in one segment other segments can stay usable.
- It normally avoids scrollbars and makes user experience flexible.

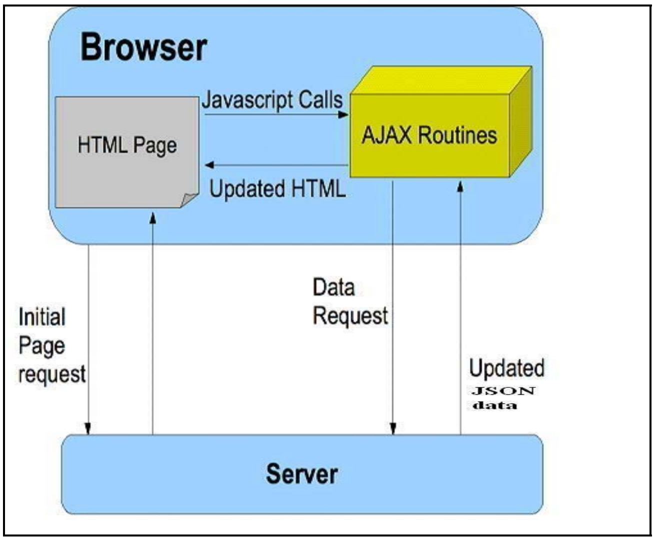

Fig.2 AJAX Mechanism

D. Security Module

Cross-site scripting (also known as XSS) occurs when a web application gathers malicious data from a user. Often attackers will inject JavaScript, VBScript, ActiveX, HTML, or Flash into a vulnerable application to fool other application users and gather data from them. For example, a poorly designed forum system may display user input in forum posts without any checking. An attacker can then inject a piece of malicious JavaScript code into a post so that when other users read this post, the JavaScript runs unexpectedly on their computers.

One of the most important measures to prevent XSS attacks is to check user input before displaying them. One can do HTML-encoding with the user input to achieve this goal. However, in some situations, HTML-encoding may not be preferable because it disables all HTML tags.

Another issue is the SQL injection; your application may be susceptible to SQL injection attacks when you incorporate invalidated user input into the database queries. Particularly susceptible is a code that constructs dynamic SQL statements with unfiltered user input.

Consider the following instance of code,

SqlDataAdapter myCommand= new SqlDataAdapter (“SELECT * FROM Users WHERE UserName=’ ” + txtuid.Text +” ‘ ”,conn);

Attackers can inject SQL by terminating the intended SQL statement with the single quote character followed by a semicolon character to begin a new command, and then executing the command of their choice.

Consider the following character string entered into the txtuid field

‘OR 1=1

This results in the following statement being submitted to the database for execution

SELECT * FROM Users WHERE UserName=’’ OR 1=1

Because 1=1 is always true, the attacker retrieves every row of data from Users’ table.

X-Frame provides in-built security mechanism for different type of hacking attempts.

Other features of the security module:

- ACL-support
- Prepared statements/parameterized queries
- CSRF protection
- XSS protection
- Database-based sessions
- Form validation/filtering
- input validation
- output validation
- context-aware output escaping

E. Code Generator

The code generator is the heart of the framework. In X-Frame, code generation is the process by which the framework's code generator converts some intermediate short representation of source code into a form that can be readily executed by an interpreter or a virtual machine.

X-Frame's compilers typically perform multiple passes over various intermediate forms. This multi-stage process will be used because many algorithms for code optimization are easier to apply one at a time, or because the input to one optimization relies on the completed processing performed by another optimization. This organization also facilitates the creation of a single compiler that can target multiple client systems, as only the last of the code generation stages (the front end java script) needs to change from target to target.

The input to the code generator typically consists of a raw source code file or an abstract syntax tree. The tree is converted into a linear sequence of instructions, usually in an intermediate language which is understandable by further stages. Further stages of compilation may or may not be referred to as "code generation", depending on whether they involve a significant change in the representation of the program.

Code generation will likely save a lot of developers time as it will not only do the compilers job of translating code but also generating the required code needed by the system to execute successfully.

F. *Intelligent Routing*

The framework provides automatic request to code routing mechanism which helps in reducing a lot of spaghetti code.

G. *Rendering Engine*

The framework provides a rendering engine which will make it easier to write templates for the web.

H. *In-built Libraries*

The framework provides libraries for most common functionalities.

Example:

client side:

```
function post() {
   var name = $('#name').value();
   var email = $('#email').value();

   $.post(url, {
      name: name,
      email: email
   }, function(data) {
      if (data == "success") {
         show_success();
      } else {
         show_failure();
      }
   })
}
```

server side:

```
function update() {

   $name = $_POST['name'];
   $email = $_POST['email'];

   $result = save($name, $email);
   return $result;

}
```

our code...

```
function update_click() {

   $name = sel('name').value();
   $email = sel('email').value();

   if (save($name, $email))
      show_success();
   else
      show_failure();

}
```

The above instance of code represents the traditional client and server side code required for developing the web application, whereas latter is the code written using our framework which comparatively has lesser lines of code thereby the developers can carry out their work more quickly using X-Frame.

II. Project Layout

In our framework, there are 2 components namely the application and system. The client code is written in the application component and the server code is written in the system component.

In application component we have model, view, controller and the event handler. For example, whenever user types http://localhost/xampp/t1/p1 as URL, the request is passed to the server through an entry point. If the required package is not present it throws an error else the request is passed to the system component which calls the controller.

The controller then starts loading the model and view, concurrently the event handler is loaded and it renders the data to the browser (control returned to the system).

If the client clicks on any event, control is again to the system via the entry point, it then loads the event handler which does the required code generation and the required output is displayed to the client.

III. Requirements

- HARDWARE REQUIREMENTS
  - ✓ 1GHz+ Processor
  - ✓ 128 MB RAM
  - ✓ 1 GB Disk Space

- SOFTWARE REQUIREMENTS
  - ✓ Any UNIX based OS
  - ✓ Web Server with PHP module
  - ✓ Any RDBMS

IV. conclusion

In this paper we discussed how the traditional method of web application development can lead to various drawbacks like code redundancy, security issues, development time, and improper structure. X-Frame is a more convenient way to build secure, user friendly web application due to the features like MVC, Code Generator, ORM, in-built AJAX and various libraries enabling the developers to create bug free web application.

V. Future Work

The framework can be easily extended to add more functionality. Moreover the Hot Code Push feature in our framework resolves the issue related to up gradation for live deployment which will guarantee zero downtime during the upgrssssade process.

Acknowledgment

We are highly indebted to our project guide, Ms. Sushama Khanvilkar, head of computer department, for her exemplary guidance and monitoring and for sharing her views about the project. Also, we like to thank all our faculties from computer engineering department at Xavier Institute of Engineering for their continued and relentless support.

We would like to express our thanks to the principal Dr. Y.D.Venkatesh for providing us with the required infrastructure for our research and project.

We would like to thank our parents and friends for their kind co-operation and encouragement.

Above all we would like to thank God for blessing us throughout the project work.

References

[1] "Develop'sbook.com,"[Online].Available:http://www.developersbook.com/

[2] Deven Shah. MARK STAMP'S INFORMATION SECURITY: PRINCIPLES AND PRACTICE. Wiley India Pvt.Ltd, 2009.

[3] "OWASP,"[Online].Available:https://www.owasp.org/index.php/Types_of_Cross-Site_Scripting

[4] "Wikipedia,"[Online].Available:https://en.www.wikipedia.org/wiki/Object-relational_mapping.

[5] Dr. Samrat Vivekanand Omprakash Khanna and Mr Mijal Mistry,"Impact of AJAX in web applicationd", International Journal of Advanced Engineering Technology[Volume 3, Issue 1, January-March, 2012]