

# Web Application Development with Object Oriented Programming

Ashish Ojha, Ritesh Tiwari, Kunal Kadam, Prof. Khandu Khot

Information Technology, Mumbai University, Maharashtra, India

## ABSTRACT

Development of web application nowadays can hardly survive without object oriented approach except for the purpose of just information display. The complexity of application development and the need for content organization has raised the need for web application developers to embrace object oriented programming approach. This paper exposes the impact of object oriented programming on web application development. The exposition was done through a detailed study and analysis of information from secondary sources. The internet was usefully employed to access journal articles for both national and international sources. Our study enables web developers and designers to understand web application features, tools and methodologies for developing web application. It also keeps researchers and scholars abreast of the boost which OOP has brought into Web Applications development.

**Keywords :** OOP, DOM, XML, AJAX, PDA, GSM, HTML

## I. INTRODUCTION

According to, Web Applications are web sites which are dynamic in nature and uses server side programming to allow a good interaction between the user form at the front end, and the database at the back-end. Such applications include but are not limited to: Online Banking, Online Reservations, ecommerce or Shopping Applications, Online Training, Online Polls, Blogs, Online Forums, Social media sites, Location based services, and Content Management Systems. Web applications are mostly preferred to their off-line counterparts because of their ability to provide multiuser interface; centralized maintenance and updates; thin client integration; distributed processing; multilingual programming; and cross platform interoperability using either the Internet, Intranet or Extranet technologies. Web applications have become very popular in the software market because of this popularity; most application developers are converting their legacy applications to the Web platform. Hence, today, virtually all kinds of applications are supported on the Web. These applications range from small scale services to large scale enterprise software.

Early web implementations were only based on low level tools that do not support high level abstraction for sharing and reuse. This lack of appropriate abstractions made it difficult to create frameworks that supported the reuse of different existing components. However, with the evolution of web application in the late 90s, Web (software) engineers realized the need for modularization of concerns to accommodate big and complex applications. Hence, technologies like Document Object Model (DOM) and Extensible Markup Language (XML) which supports object oriented principles (such as modularization, encapsulation and abstraction) were developed. This bridged the gap between higher and lower level implementation. Today different frameworks and tools have been developed with maximum support of object oriented concept for web application development.

Our objective in this paper is to expose the impact of object oriented programming paradigm on web application development. Our study will also enable web developers and designers to understand web application features and popular languages, tools, and methodologies for developing web application.

The remaining sections of this paper are arranged as follows: In section 2 and 3, we reviewed computer classes and computer application; the features and popularity of web applications; web application development languages and tools, and methodologies for developing web application. However, our discussion in this section will be mainly focusing on the functionality and history/evolution of the tools. In section 4, we discussed the impacts of object oriented programming on web application. Finally, we concluded our paper with the contribution of object oriented programming on web application development.

## II. METHODS AND MATERIAL

### LITERATURE REVIEW

Many of the studies in the area of web application development have mainly focused on the evolution of web application and comparison of web application development languages. Jazayeri wrote on trends and status quo of web application, Ronacher presented security related issues in web application.

Vosro and Kourie wrote on concepts and web framework. Purer highlighted some differences, advantages and drawbacks of PHP, Python and Ruby. He compared the languages based on history, evolution, popularity, syntax, semantics, features, security and performance in web application environments. Cholakov analyzed PHP and summarized some drawbacks. Gellersen and Gaedke in their article, overviewed object oriented web applications and identified object-oriented model for web applications, they found that XML technology contributes in enabling high level abstractions for design level modeling in a markup language. Mattsson identified the strengths and weaknesses of object oriented frameworks. Finifter and Wagner explored the relation between web application development tools and security. Chatzigeorgiou et al, evaluated object oriented design with link analysis. Paikens and Arnicans explored the use of design patterns in PHP-based web application frameworks. French presents a new methodology for developing web applications and web development life cycle. Copeland et al, in their article titled "Which web development tool is right for you" discussed and compared various tools for web application development.

However, not too many studies have been conducted in the area of impact of object oriented programming on web application development. This research aims at discussing the impact of object oriented programming on web application.

### 2.1 Computers and Computer Applications

Although, early computer applications runs on mainframe computers, and application development were mainly on a low level languages accessed through a low technology terminal device. However, the popularity of computer applications started when the Personal Computers (PC) were invented. This technological advancement made computers more affordable and accessible. Also software developers started developing standalone applications that run locally on the PC. PC networking technology was subsequently discovered and client-server applications were created, which allowed PCs to remotely connect to servers and share information. Create a link between PC networking and the internet. With the constant decrease in prices of microcontroller; high speed of internet connectivity and increase user participation on Web platforms, the Web became a dominant platform for applications development.

### 2.2 Classes of Computer Applications

There are several classifications of computer applications: based on uses (Graphics applications, accounting applications, etc); based on the developer (proprietary or non-proprietary). In-lieu of our discussion, we shall classify computer applications based the platform they run. These include:- □

**Desktop Application:** These are applications which run on the desktop computers. They are developed to be installed and run on a particular system hardware and operating system. One major advantage of desktop applications is that they are standalone applications; hence they do not need network connection to operate.

➤ **Web Applications:** these are client-server software applications that run on Web browser. Unlike desktop applications that depend mainly on the operating system or hardware platform, a Web application depends on the web browser and server. Any computer device with web browser and internet connection can run web application

irrespective of the operating system and hardware platform .

- **Embedded Applications:** these are applications that are written inside Read-Only Memories (ROM) of computing controlled appliances. These types of applications differ significantly from desktop and web application mainly in environmental conditions such as real time, performance expectations, maintenance difficulty, changing platforms, just to mention .
- **Mobile Applications:** these are applications that run on mobile devices such as PDA, GSM and Android phones, etc. The early mobile phones were mainly capable of making and receiving calls, sending and receiving Short Message Service (SMS), without sophisticated applications. With Wireless Application Protocol (WAP) standard, new mobile applications that run on WAP browser started to emerge . Just like standalone desktop applications, mobile applications can be standalone and as well web based or even embedded.

### 2.3 Web Application, Their Features and Popularity

Though early stage of the Web evolution, Web 1.0, which existed between 1990 and 2000 [8] enjoyed some level of growth due to its multiuser interface; single point maintenance and updates; distributed and hyperlinked documents etc. The level of popularity and user activity was still low owing to the fact that most of the websites developed using web1.0 was static and operated in brochure architecture with only professional web designers producing the content for users to access.

The Web platform we enjoy today, Web 2.0 has grown tremendously with resurgence of popularity and interest from millions of companies and billions of users across the world. Web 2.0 operates in architecture of participation where companies only provide the platform and users generate the content. Most sites on the Internet today like wikis, blogs and social media sites all present user generated content bringing the shift from few powerful professionals (programmers) to many empowered users . In-lieu of this development, so many tools have been introduced to enable more user participation such as the Rich Internet

Applications (RIA) and AJAX (Asynchronous JavaScript and XML) technologies. These technologies are used to develop web applications, which look and behave like desktop applications.

At the root of this Web 2.0 evolution, which is triggered by technologies like AJAX, Document Object Model (DOM), RIA, frameworks etc, is enshrined the concept of Object-Oriented Programmed (OOP). Hence, the knowledge of these tools and OOP cannot be ignored.

### 2.4 Web Application Tools and Languages

The World Wide Web operates in client-server architecture with the user making request using his/her Web browser at the client-side and the Web server, on the server-side, responding to user request. The entire Web 2.0 technologies can simply be categorized into AJAX technologies, RIA technologies and Frameworks. Hence, we shall discuss Web application tools and languages under the following: Client-side technologies, Server-side technologies and Web application Frameworks

#### 2.4.1. Client-Side Technologies

These are Web technologies, tools and languages that run on the browser. Some of these tools include but are not limited to the following:

- **Web Browser:** The major responsibilities of web browsers are: (1) to generate and send request to web servers. These requests can be from hyperlinks, submitted forms, Uniform Resource Locator (URLs) and Hypertext Markup Language (HTML) pages. The request could also involve auxiliary resources like applets, and images, which accept and interpret responses from web servers to produce visual representation; (2) to render results in web browser window or tools from third party based on the type of content. Some of the most popular web browsers are Internet Explorer, Chrome, Firefox, Safari and Opera, etc.
- **RIAs client technologies:** Rich Internet Applications (RIA) are Web applications that behave and look like desktop applications. RIAs are the major attraction of the Web 2.0. RIAs client

technologies are RIA technologies that run on the browser. This include Microsoft Silverlight, Linux Moonlight, Adobe Flash, Adobe Flex, etc

- **AJAX technologies:** Asynchronous JavaScript and XML (AJAX): this is one of the key technologies of Web 2.0 and RIAs. AJAX allows different sections/regions of a Web page to be refreshed separately while the user is still

on the same page. In RIA, AJAX technologies provide the fast response of the application. Common AJAX technologies are XHTML (Extensible Hypertext Markup Language), CSS (Cascaded Style Sheet), JavaScript, DOM, XML (Extensible Markup Language) . AJAX implements the Object Oriented Model. For example using the DOM, each element of a Web page can be treated as an object which can be created, modified or removed independently.

#### 2.4.2. Server-Side Technologies

These are technologies, tools and languages that run on the Web server. Some of tools include:

- **Web Server:** these are software applications that store the hyperlinked documents and other Web resources for users to access. IIS (Microsoft Internet Information Services) and Apache (developed by Apache Software Foundation) are the most popular Web servers today.
- **Database Management Systems (DBMS):** most Websites of today's Internet use DBMS to store their data. These databases are managed with DBMSs which are installed separately or as a suite. Common DBMS on the Internet are Structure Query Language (SQL) based. These include: SQL Server, MySQL, and Java DB.
- **RIAs Server technologies:** these are RIA languages and frameworks that run on Web Server. They include Perl, PHP, Ruby, Ruby on Rails, JavaServer Faces, Active Server Pages (ASP & ASP.NET) and Web Services.

#### 2.4.3 Web Application Frameworks

Web application frameworks are sets of precompiled libraries, components and software tools arranged in such a way to separate content from presentation. This is done by dedicating the task of presentation formatting to web designers, while developers will be responsible for program logic and access the content [17]. This practice creates a consistent architecture among designers and developers for developing and accessing request framework elements that can be embedded within the presented page. Such requests including support for authentication, data access and transformation, session and state management. Frameworks provide the rich interfaces for RIAs giving them similar appearance as their desktop counterparts.

The table 1 shows some of the frameworks and tool support for web application development including IDEs.

**Table 1.** Web Application Development Tools, Frameworks and IDEs

Tools	Frameworks	IDE
PHP	Zend, CakePHP, Symfony Project, Seagull, WACT, Prado, Zoop, PHP on TRAX, eZ Components, Codelgniter etc.	Micromedia Dreamweaver, NetBeans, JetBrains, WebMatrix3 etc
JSP	Struct 2, JavaServer Faces (JSF), Spring MVC, Wicket, Stripes, Tapestry, RIFE, Seam, Google Web Toolkit (GWT), OpenXava etc.	Eclipse, JavaBean, JBuilder, Micromedia Dreamweaver, Sun Java Studio Creator, Borland JBuilder, Stylus Studio, Oracle JDeveloper etc.
ASP	ASP .NET MVC Framework, ASP.NET Dynamic Data, Mono Rail, OpenRasta, Component Based Scalable Logic Architecture(CBSLA), Base One Foundation Component Library(BOFCL) etc.	Visual Basic .NET, Cloudio, WebMatrix3, MonoDevelop, JetBrains, WeBuilder, NetBeans etc.

### 3. Web Application Development Methodologies

A web application is a client to server application that generally adopt web browser as its client, the browser sends requests to the server and the server generate responses and send to the browser . There are several approaches to web application development. The choice of development approach depends on several factors such as the nature of the information, scale of the application, frequency of content update, experience of the developer, and many more. In this paper, we briefly look at 3 methodologies.

1. **System Development Life Cycle (SDLC):** is the traditional process of developing software or web applications by including research to identify and define the application requirements, information analysis, architectural design and specifications blueprint, team involvement, programming, testing and bug fixing, system testing, implementation and maintenance .
2. **Agile Software Development:** is a set of methodologies (like scum) that uses few resources to build software faster. Agile methodologies are often used for systems whose user requirements are constantly changing.
3. **Object- Oriented Methodologies:** over the years a lot of software methodologies have been proposed by different authors using object-oriented concept to address contemporary issues on the Web. They include:
  - a. WebComposition: is an object-oriented methodology that decomposes Web applications into components. Each component (page, site etc) is defined in terms of its state and behavior. This approach maintains support for software reuse, high level abstraction, run-time management and maintenance throughout the development cycle of the web application .
  - b. W3Objects: this methodology encapsulates the web elements as objects with well defined interfaces for interaction with other objects. Each object can inherited to build new ones. Using W3objects, the problem of referential integrity and transparent object migration can be well handled .
  - c. Jessica: is a web engineering system that employs object-oriented abstraction model of HTML++ (a methodology similar to W3objects). Jessica system is made up of two major component: an Object-oriented language component of the web application and a compiler that maps the abstract system definition into highly dynamic Web services.

### III. RESULTS AND DISCUSSION

Object oriented programming is currently the flexible pro-gramming pattern that every software developer or programmer employs, this OOP concept applies to both web application development and standalone software development. Many programming languages and frameworks particularly for web applications are now

structured along-side OOP techniques. It is clear that all the recent frameworks for web application development maintain a solid structure of object oriented techniques and design patterns. It is therefore necessary to understand what impact object oriented approach has on web application development. We have considered this under positive impacts and challenges posed by the approach.

#### 4.1 Positive Impacts of OOP on Web Application

The popularity of web applications (web Apps) was the reason for the applications developers to evolve a more robust technique for the web App development. The adoption of OOP techniques in web App development has brought about the following:

1. Reduce the number of code a programmer needs to write: if the application a programmer is developing requires a similar functionality of already developed application, the programmer just need to edit little code to meet the programmers required function than writing the whole code from scratch.
2. Enable reuse of design and code function: in developing a web application a programmer do not need to write the whole code for the application, he/she can easily copy design or code function from other Apps and add to his/her project to suite his/her application needs.
3. Transfer design knowledge and experience to a developer: from the reuse of design developed by others, design knowledge and experience can easily be transferred.
4. Improved Maintainability of web application: since object oriented approach breaks complex programs into separate modules. Functions can easily be removed or updated on web page without affecting other functions.
5. Reduce the cost and time of developing a web application: With availability of function codes (for example one can get a JavaScript code that displays time in an application and integrate into a program) and rich library of object oriented programming, there is a reduced time and cost for web App development.
6. Improve application access restriction from unauthorized person: The concept of encapsulation

in object oriented approach enables web application to specify access right to a function or content on web application. In fact, this benefit to web application has actually contributed a lot to the acceptance of web applications in several sensitive areas like e-Banking and e-commerce.

7. Create opportunity for web application development framework: using the object oriented concept, reusable framework elements have been developed. This can be embedded within the pages of a website to provide authentication, data access and transformation, session and state management.
8. It creates opportunity for integration of external web application with other web application such as Intranet and Extranet, by allowing object/module to be accessible to other functions or programs.
9. Proper computation and organization of complex tasks: object oriented approach helps to specify responsibilities and the distribution of these responsibilities among the cooperating objects. This helps to solve complex tasks and create opportunity for multi-tasking.

#### 4.2 Challenges with OOP on Web Application development

- 10.
1. Though the OOP technique sounds wonderful for web App development, it does present some challenges to web App developers. The most serious among these challenges include:
2. Requires expert knowledge: developers or programmers who are just starting-out will find it difficult to integrate HTML and Scripting language as it requires a descent level of understanding of various Object Oriented Programming (OOP) techniques and principles
3. Difficult to find bugs in code especially when embedding scripting language with HTML code. Again, one can introduce unknown bug through code reuse from copying a function from other sources.
4. It creates opportunity for cyber attack through the support for integration of code from third party application. Object oriented pattern accepts integration of object from other source which can permit application attack.

#### IV. CONCLUSION

The popularity of the Web and its impacts as a client-server system has led to a vast collection of web application. That was not so until frameworks and other technologies that supported object-oriented concepts were introduced. Our study has shown the various classes of computer and its application development more especially in the area of web applications. The object-oriented paradigm has been shown to be the most secured and flexible pattern for developing web application by shifting task from functional modules to a more confined level. Despite the few challenges with the integration of object-oriented concept in Web development, our conclusion is that the impact of object oriented approach on web application development (especially in the areas of resource reuse, extensibility, maintenance, scalability and delegation) are implicit: This impact has been tremendously positive and has brought about a large acceptance of web applications. Hence web App development has become a boom due to OOP.

#### V. REFERENCES

- [1]. Athula Ginige, (2014). Web Engineering: Methodologies for Developing Large and Maintainable Web Based Information Systems.
- [2]. Barta Robbert A., & Schranz Markus W. (1998). "Jessica: An Object-Oriented Hypermedia Publishing Processor: Computer Networks and ISDN Systems. Proceedings of the 7th International WWW Conference. Vol. 30(1-7), pp.281-290.
- [3]. Borland JBuilder (2002) Web Application Developers Guide; Borland Software Corporation, 100 Enterprise Way, Scotts Valley, CA 95066-3249
- [4]. Chappell David (2011). What is an Application Platform?; Microsoft Corporation, Chappell & Associates.
- [5]. Chatzigeorgiou Alexander, Xanthos Spiros & Stephanides George (2004). Evaluating Object Oriented designs with link analysis; Proceeding of the 26th International Conference on Software Engineering, IEEE Computer Society
- [6]. Cholakov Nikolaj, (2008). On some drawbacks of the php platform, CompSysTech '08: Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing (New York, NY, USA), ACM, pp. II.7.
- [7]. Copeland Dennis R., Corbo Raymond C., Falkenthal Susan A., Fisher James L., & Sandler

- Mark N. (2000). Which Web Development Tool Is Right for You? IT Pro IEEE
- [8]. Deitel, P. J. & Deitel, H. M. (2007), Java How To Program, USA, Pearson Inc., 7th Ed., pp. 421 - 423.
- [9]. Ebert Christof & Jones Capers (2009): Embedded Software: Facts, Figures, and Future: IEEE Computer Society
- [10]. Finifter Mathew & Wagner David (2010). Exploring the Relationship between Web Application Development Tools and Security
- [11]. French M. Aaron (2011). Web Development Life Cycle: New Methodology for Developing Web Applications, Journal of Internet Banking and Commerce, vol.16, no.2
- [12]. Gellersen, H. & Gaedke, M. (1999). Object-Oriented Web Application Development. IEEE Internet Computing. Accessed from <http://computer.org/internet> on 3/8/15
- [13]. Gellerson, H. W.; Wicke, R.; and Gaedke, M. (1997). WebComposition: an object-oriented support system for the Web engineering lifecycle, Computer Networks and ISDN Systems, 29(8-13): 1429-38, April 1997.
- [14]. Ingham, D.; Little, M.; Caughey, S. and Shrivastava, S. (1995) W3Objects: Bringing Object-Oriented Technology to the Web , Fourth International World Wide Web Conference, Boston, Massachusetts, 1995, USA. URL: <http://www.w3.org/pub/Conferences/WWW4/Papers2/14.1/>
- [15]. James Governor, Dion Hinchcliffe & Duane Nickull (2009). Web 2.0 Architectures. 1st ED, O'REILLY Media Inc. ISBN: 978-0-596-51443-3
- [16]. Jazayeri Mehdi, (2007) Some trends in web application development, FOSE '07: 2007 Future of Software Engineering (Washington, DC, USA), IEEE Computer Society, pp. 199
- [17]. Kohan, B. (2015). Guide to Web Application Development: Guides, Resources, and Best Practices. Accessed from <http://www.comentum.com/guide-to-web-application-development.html> on 02/08/15
- [18]. Mattsson Michael, (1996). Object-Oriented Frameworks: A Survey of Methodological Issues. CODEN: LUTEDX(TECS-3066)/1-131
- [19]. Paikens Andris and Arnicans Guntis (2008). Use of Design Patterns in PHP-Based Web Application Frameworks, Latvijas Universitātes Raksti:Datorzinātne Un Informācijas Tehnoloģijas
- [20]. Purer Klaus (2009). PHP vs. Python vs. Ruby - The web scripting language shootout; Vienna University of Technology, Institute of Computer Languages, Compilers and Languages Group, 185.307 Seminar aus Programmiersprachen
- [21]. Ronacher Armin, (2006) Sicherheit in Webanwendungen, <http://dev.pocoo.org/blackbird/fachbereichsarbeit.pdf>
- [22]. Sareh Aghaei, Mohammad Ali Nematbakhsh & Hadi Khosravi Farsani (2012). Evolution of the World Wide Web: From Web 1.0 to Web 4.0, International Journal of Web & Semantic Technology (IJWeST), Vol.3(1)
- [23]. Shklar Leon and Rosen Richard (2003) Web Application Architecture: Principles, Protocols and Practices, John Wiley & Sons Ltd, ISBN 0-471-48656-6
- [24]. Tejas Vithani & Anand Kumar, (2014). Modeling the Mobile Application Development Lifecycle: IMECS, Hong Kong, ISBN: 978-988-19252-5-1, Vol.1
- [25]. Voslro Iwan and Kourie Derrick G., (2008). Server-centric web frameworks: An overview, ACM Comput. Surv. 40 (2).