

THE ADVANTAGE AND DISADVANTAGES OF USING PHP, JAVASCRIPT TECHNOLOGY IN CREATING THE SERVER PART OF THE PLATFORM

Bilolov Inomjon Uktamovich

The Docent of the Fergana Branch of Tashkent University of Information

Technologies named after Muhammad al-Khwarizmi

bilolov1959@mail.ru

Khomidov Rakhmatullo Mirzarahmon ugli

The Master of the Second Course of Fergana Branch of Tashkent

University of Information Technologies named after Muhammad al-Khwarizmi

rahmatulloxomidov@gamil.com

ABSTRACT

Creating networks of “smart things” found in the physical world (e.g., with RFID, wireless sensor and actuator networks, embedded devices) on a large scale has become the goal of a variety of recent research activities. Rather than exposing real-world data and functionality through vertical system designs, we propose to make them an integral part of the Web. As a result, smart things become easier to build upon. In such an architecture, popular Web technologies (e.g., HTML, JavaScript, Ajax, PHP, Ruby) can be used to build applications involving smart things, and users can leverage well-known Web mechanisms (e.g., browsing, searching, bookmarking, caching, linking) to interact with and share these devices.

Keywords: PHP, server, Javascript, technology, platform.

INTRODUCTION

Modern technologies for creating and maintaining Web sites are focused on platforms that allow effective content management and data coming from visitors to the site. As a rule, such solutions are based on server technologies such as ASP, ASP.NET, JSP, PHP or use powerful ready-made tools for creating corporate websites, focused on the implementation of these technologies. Let's take a closer look at popular information technologies. Creating web pages from fragments of server code is a technology ASP, ASP.NET (Active Server Pages). This is a commercially available technology developed by Microsoft with which the webmaster can create dynamically updating web pages on his own. A characteristic feature of this technology is the ability to separate the functional part of development from the design creation.

MATERIALS AND METHODS

JSP technology (Java Server Pages) is a technology for creating Java server pages. The JSP specification is an extension of the Java Servlet API for creating dynamic web pages on a web server. This cross-platform is an alternative to Microsoft's ASP technology. Sun specification called JSF (Java Server Faces) implements JSP technology, which describes the rules of creating web applications with user-friendly interface and focuses on the development of server-side components of the interface. One of the first technologies to create server-executed web applications was Common Gateway Interface (CGI) technology. It allowed for the development

and execution of server applications accessed by the name (and parameters) specified in the URL. Depending on the chosen protocol, incoming information of such web applications is considered to be directly an HTTP header code or a search engine request. CGI applications are console applications that generate HTML code sent to the browser.

RESULTS AND DISCUSSION

As more and more devices are getting connected to the Internet, the next logical step is to use the World Wide Web and its associated technologies as a platform for smart things (i.e., sensor and actuator networks, embedded devices, electronic appliances and digitally enhanced everyday objects). Several years ago, in the Cool Town project, Kindberg et al. (Kindberg et al. 2002) proposed to link physical objects with Web pages containing information and associated services. Using infrared interfaces or bar codes on objects, users could retrieve the URI of the associated page simply by interacting with the object. Another way to use the Web for real-world objects is to incorporate smart things into a standardised Web service architecture (using standards, such as SOAP, WSDL, UDDI) (Guinard et al. 2010d). In practice, this would often be too heavy and complex for simple objects.

Instead of these heavyweight Web services (SOAP/WSDL, etc.), often referred to as WS-* technologies, recent “Web of Things” projects (Wilde 2007; Guinard et al. 2010c; Luckenbach et al. 2005; Stirbu 2008) have explored simple embedded Hypertext Transfer Protocol (HTTP) servers and Web 2.0 technology. In fact, recent embedded Web servers with advanced features (such as concurrent connections or server push for event notifications), can be implemented with only 8 KB of memory and no operating system support, thanks to efficient cross-layer TCP/HTTP optimisations, and can therefore run on tiny embedded systems, such as smart cards (Duquennoy et al. 2009). Since embedded Web servers in an Internet of Things generally have fewer resources than Web clients, such as browsers or mobile phones, Asynchronous JavaScript and XML (Ajax) has proven to be a good way of transferring some of the server workload to the client.

So far, projects and initiatives, subsumed here under the umbrella term “Internet of Things”, have focused mainly on establishing connectivity in a variety of challenging and constrained networking environments. A promising next step is to build scalable interaction models on top of this basic network connectivity and thus focus on the application layer. In the Web of Things concept, smart things and their services are fully integrated in the Web by reusing and adapting technologies and patterns commonly used for traditional Web content. More precisely, tiny Web servers are embedded into smart things and the REST architectural style (Richardson and Ruby 2007; Fielding 2000) is applied to resources in the physical world (Guinard et al. 2010c; Luckenbach et al. 2005; Duquennoy et al. 2009; Hui and Culler 2008). The essence of REST is to focus on creating loosely coupled services on the Web, so that they can be easily reused. REST is the architectural style of the Web (implemented by URIs, HTTP, and standardised media types, such as HTML and Extensible Markup Language (XML) and uses URIs for identifying resources on the Web. It abstracts services in a uniform interface (HTTP’s methods) from their application-specific semantics and provides mechanisms for clients to select the best possible representations for interactions.

Among other popular technologies implementing the creation of web pages with fragments of code executed on the server, we should highlight the non-commercial freeware technology PHP (Personal Home Pages). This technology is based on the use of CGI-applications that interprets embedded in HTML-page code in scripting language. The main feature of PHP language is its practicality. NRC provides the programmer a tool for fast and efficient tasks solution. It is extremely flexible to the needs of the developer. Although RNR is traditionally recommended for use in conjunction with HTML-code, but RNR can just as well be integrated into JavaScript, WML, XML and other Internet programming languages. The considered technologies provide modern functionality, effective processes support of sites creation and filling of information resources.

The results of the analysis allow us to conclude that there are advantages of performance characteristics of PHP-technology. The main advantages of PHP, as we see, are practicality, efficiency, performance and flexibility. PHP frameworks have recently gained popularity and became the basic platform for the web applications development. Using these systems can save a lot of time, reduce the burden on the development process by eliminating the problem of repetitive code, and quickly create quality applications. Meanwhile, the use of PHP frameworks makes the process of creating a program much easier and more functional.

Based on the above analysis, the Yii2 framework is better suited for solving the tasks of the mentioned type. This system is perfectly tuned for the projects of such complexity; it does not require additional configuration, has built-in support of multilingualism, is user-friendly, and provides a high level of security for the products developed with it. Also, for the system development you need to use markup language HTML5 and cascading style sheet CSS3. This is the best and most convenient tools with a similar purpose. They allow you to build Web-pages with a variety of sizes and shapes. The programming language java script uses for interactive and active user interaction with the web application.

CONCLUSION

Java script is a dynamic, object-oriented programming language. It is an implementation of ECMAScript most often used as part of the browser that allows the code to run on the client side, thus removing the burden on the main server, which runs the basic functionality of the system. This language can also be used for server-side programming, game development, fixed and mobile applications, scripting in applications, inside PDF documents, etc. JavaScript is classified as a prototype (a subset of object-oriented), scripting programming language with dynamic typing. Besides prototypical JavaScript also partially supports other programming paradigms (imperative and partially functional) and some related architectural features, in particular: dynamic and weak typing, automatic memory management, prototype mimicking, functions as first-class objects. Given the identified advantages and disadvantages of the analyzed tools for creating web applications designed to keep statistics, the following technologies are best suited: Yii2, Java Script, HTML5 and CSS3.

REFERENCES

1. Dr. Afif J. Almgawish, Dr. Alexandre F. Ossyka, Dr Issa S. Software Support For Programming Language Tutorials. / A. J. Afif, O.F. Alexandre, S. Issa // World of Computer Science and Information Technology Journal (WCSIT). –2013. –№9. –p.144-149.
2. Zakas N.C. Professional JavaScript for Web Developers, 3rd Edition / N.C. Zakas // Wrox, – 2012. –960 p.
3. www.java.com