

UDC 004.42

APPROACHES TO THE DEVELOPMENT OF THE FRONT-END PART OF A WEB APPLICATION
ON THE EXAMPLE OF "WEBSITE FOR THE SALE OF GAMES"

ULADZIMIR TRAPEZNIKAU, RYKHARD BOHUSH
Polotsk State University, Belarus

The article presents the analysis of the technologies used in the implementation of web application interfaces. The example of the implementation of the web interface of one of the pages of the web application is given.

Introduction. Various software products accessed through a web interface are called today web applications or sometimes also web-based systems. A web application is a client-server application in which the client acts as a browser, and the server is a web server. The logic of the web application is distributed between the server and the client, the data is stored mainly on the server, information is exchanged over the network. One of the advantages of this approach is the fact that clients do not depend on the specific operating system of the user, so web applications are cross-platform services.

A significant advantage of building web applications to support standard browser functions is that the functions must be executed independently of the client's operating system. Instead of writing different versions for Microsoft Windows, Mac OS X, GNU / Linux and other operating systems, the application is created once for an arbitrary chosen platform and is deployed on it.

The relevance of creating a web application is justified by the fact that with the development of portable computers and mobile devices, the client part of the web application can be used not only on desktop computers and laptops, but also on the mentioned mobile devices.

Means of solving the problem. Let's consider the main approaches and means for implementing the interface of the developed web application covering the amateur handball of the Republic of Belarus.

As an authentication system, the ASP.NET Identity system will be used, which allows to authorize through external services, manage roles to differentiate access to data, validate email and phone by SMS, and validate passwords [1].

To optimize the appearance of the site for various types of devices and to simplify the task of graphic design of the site, Twitter Bootstrap, jQuery, AJAX are used.

jQuery is a javascript library, the use of which makes development of javascript code much faster and easier. In the recent past, this library allowed to quickly develop scripts. The jQuery library helps easily access any DOM element, access and manipulate the attributes and contents of DOM elements. Also, the jQuery library provides a convenient API for working with AJAX [2].

AJAX, or, longer, Asynchronous Javascript And Xml is a technology for interacting with the server without reloading the pages. Due to this, the response time is reduced and the web application interactivity more resembles the desktop. For example, it is possible to leave comments on the page summarizing the results of the competition and immediately see the result of adding a comment without reloading the page. To exchange data with the server, a special XMLHttpRequest object is used, which can send a request and receive a response from the server. When updating data, the web page does not completely restart, and web applications become faster and more convenient [3].

Bootstrap is a framework that Twitter has developed. It is designed to facilitate the construction of a graphical interface. The library itself includes a huge number of elements such as buttons, web forms, navigation blocks and much more. Of course, if nothing is initially changed, the resulting form will be similar to that of many other developers using the same framework, but it should be borne in mind that in the early stages of development a fairly accurate interface can be obtained, and eventually it can still be customized. The source codes are also distributed by the MIT license, which allows free use and change of technology [4].

ASP.NET Core MVC is a cross-platform, open source development framework for building web applications and services using the Model View Controller (MVC) pattern. It is a successor to several legacy Microsoft web development frameworks and merges the functionality previously found in ASP.NET MVC, ASP.NET Web Pages, and ASP.NET Web API in a single modular framework. ASP.NET Core MVC offers orders of magnitude better performance than legacy ASP.NET and can be deployed almost anywhere including Windows Server, Mi-

ITC, Electronics, Programming

icrosoft Azure, Linux, and macOS. It also has built-in tooling that simplifies packaging ASP.NET Core MVC applications for use with container architectures such as Docker and Pivotal Cloud Foundry.

ASP.NET Core MVC shares many of the same programming constructs as ASP.NET MVC classic but has been rewritten from the ground up on top of Core CLR. It is more lightweight and significantly faster than ASP.NET MVC classic. Benchmarks conducted by the ASP.NET team have shown that ASP.NET Core can process more than 1.15 million requests per second with 12.6Gbps throughput. This is a 2,300 percent improvement over ASP.NET 4.6.

Knockout is a JavaScript library that helps to create rich, responsive display and editor user interfaces with a clean underlying data model. Any time you have sections of UI that update dynamically (e.g., changing depending on the user's actions or when an external data source changes), KO can help to implement it more simply and maintainably.

Headline features:

- Elegant dependency tracking - automatically updates the right parts of your UI whenever a data model changes.
- Declarative bindings are a simple and obvious way to connect parts of UI to the data model. It is possible to construct a complex dynamic UIs easily using arbitrarily nested binding contexts.
- Trivially extensible - implement custom behaviors as new declarative bindings for easy reuse in just a few lines of code.
- Additional benefits:
 - Pure JavaScript library - works with any server or client-side technology
 - Can be added on top of an existing web application without requiring major architectural changes
 - Compact - around 13kb after gzipping
 - Works on any mainstream browser (IE 6+, Firefox 2+, Chrome, Safari, Edge, others)
 - Comprehensive suite of specifications (developed BDD-style) means its correct functioning can easily be verified on new browsers and platforms [5].

Interface design. On the main page of the site one can see games, filtered by genres and supported OS. At the top of each page of the site, using the master page, one will see the navigation bar and registration, login, exit buttons from the personal cabinet, bucket (fig.).

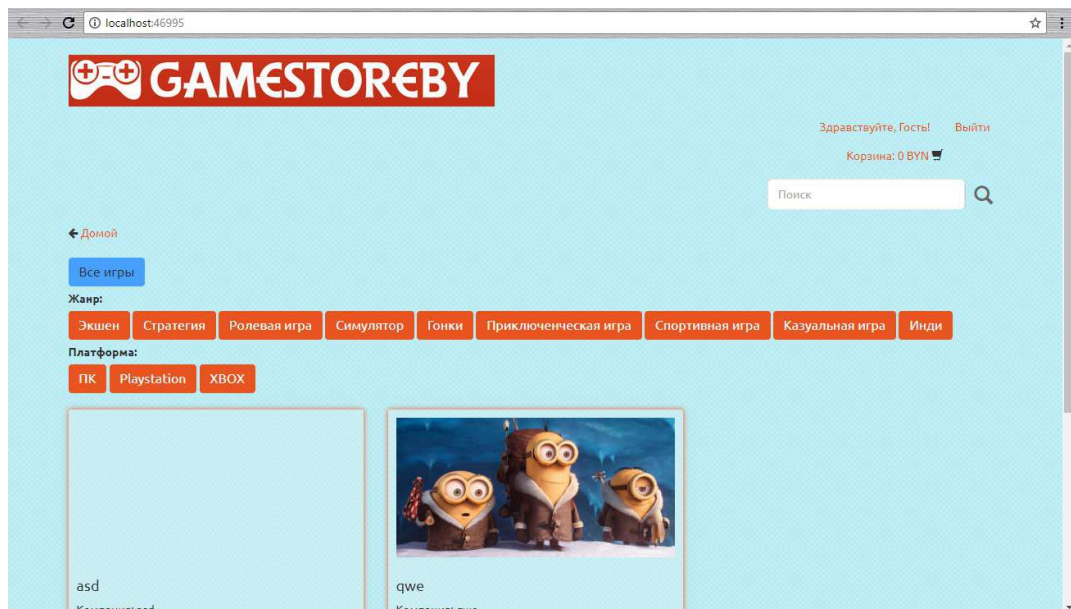


Figure. – Home page

When clicking on the "Registration" link, the user will be redirected to the registration page. On this page, you must enter a name, unique email, password and password confirmation. If the user fill out the forms correctly, they will be redirected to the login page.

Also, on the right side of the screen on the personal cabinet page, the user can change the avatar by downloading an image up to 4 MB in size.

Conclusion. In conclusion, we would like to say that when developing web interfaces it is necessary to take into account not only the beautiful external component, but also important factors such as "usability", ease of use, understandability for the end user, browser independence, cross-platform, adaptability for various sizes screens, dynamism. And all these requirements can be achieved using a small set of frameworks and libraries, as in this case it's Twitter Bootstrap, AJAX, jQuery, Knockout.

REFERENCES

1. Metanit - site about programming [Electronic resource]. – Access mode: <http://metanit.com>. – Date of access: 27.02.2019.
2. jQuery - Description of the library [Electronic resource]. – Access mode: <https://htmlweb.ru/java/jquery.php>. – Date of access: 27.02.2019.
3. Introduction to AJAX [Electronic resource]. – Access mode: <http://javascript.ru/ajax/intro>. – Date of access: 27.02.2019.
4. Bootstrap [Electronic resource]. – Access mode: <http://getbootstrap.com>. – Date of access: 27/02/2019.
5. Knockout [Electronic resource]. – Access mode: <https://knockoutjs.com/documentation/introduction.html>. – Date of access: 27.02.2019.
6. jQuery-plugin DataTables [Electronic resource]. – Access mode: <https://datatables.net>. – Date of access: 27.02.2019.