UDC 004.588

IMPORTANCE OF JSX AND CSS ON THE EXAMPLE OF THE DEVELOPED GAME WEB APPLICATION "LEARNING CSS FEATURES"

R. GUSEVSKY, I. BURACHONAK
Polotsk State University, Belarus

The rationale for the development of modern gaming applications is given. The technologies used to create web applications are analyzed. CSS technologies and JSX technology are considered in detail, with the aim of using them for demonstration in the form of an entertaining game "Learning CSS Features".

Most people spend their free time in games, especially children and teenagers. There is such an approach to learning as submitting material for study while playing that will be useful and interesting to people of any age. Under this topic, a cross-platform, gaming web application "Learning CSS Features" is developed.

At the initial stage of its development, the "web" consisted only of static HTML pages and when the address was opened, the server returned the page content. Page-making, styles, scripts – everything was in the mass. With the development of Internet technologies, there are more and more single-page web applications, where the main part of the logic is focused on the client side. On any Uniform Resource Locator (URL), the server gives the same static HTML, and the JavaScript code in the user's browser determines the current address and makes a request to the server for the necessary data. When you click on the link, the page does not reload entirely, but the URL is updated, and the JavaScript code makes a new request for the next page. The emergence of web applications has fundamentally changed the approach to development. There are new special frameworks designed for creating web applications with a large amount of code and complex user interface logic. Rendering (the process of getting an image from a model using a computer program) of data now takes place on the client, the page resources store static html templates that are filled with data received from the server. The most suitable way to organize such a mass of code is the component approach. Each more or less independent block is made out into a separate component, with its own styles, template and JavaScript (JS), and then the whole page is assembled from these blocks. This approach is implemented by most modern JS frameworks, but with one difference: Angular, Ember encourage the creation of a separate file with a template, and React offers to write HTML inside JS [1]. The technology that allows us to write HTML inside JS is JavaScript XML (JSX) technology.

JSX is a technology that was introduced by React. This technology extends The JavaScript syntax, which looks similar to XML. Many people think that using JSX is like a mixture of HTML and JavaScript, but this is not exactly true, we can also manipulate CSS. JSX is mainly used as building blocks in React. Essentially, JSX is nothing more than a component in which we can control page making, logic, and styles. Everything is in one place, which already gives a huge advantage in development.

CSS is considered as cascading style sheets, which are a consequence of the further development of HTML and give us the opportunity to move to the next level of presentation of information. Style sheets allow you to separate the semantic content of the page and its design. When learning CSS, you may encounter a number of problems: different styles, properties, and technologies for layout of blocks and tables. HTML is said to be a markup language that is a tool for creating hyperlinks, inserting images, tables, and much more [2].

In the modern world, web technologies and programming languages are developing rapidly, hence, the number of web developers, as well as users who want to learn or develop their programming skills, is increasing. However, when studying web technologies, many people face various problems, and spend a lot of time searching for solutions. In order to facilitate the study of CSS, it is proposed to develop a web application that focuses on teaching the complex CSS properties and its layout technologies: flexbox, grid layout, working with animation, positioning elements on a web page using the position property. Thanks to visualization and game presentation, the technologies under consideration will become more accessible for studying and memorizing material. You can hardly become a programmer just by playing. But you can get a positive charge at the start, study the basic algorithmic constructs and programming logic, acquire the skills of action planning in solving practical problems [3].

When implementing this web application, it is planned to use the following technologies: ReactJS as a framework on the client side, the Node.JS software platform and the Express framework as well as the Javascript programming language for developing the server side. ReactJS is an open source JavaScript library for developing single-page and mobile applications. Its goal is to provide high speed, simplicity and scalability.

As a library for developing user interfaces, React is often used with other libraries such as Redux [4]. Node.js is a software platform based on the V8 engine (translating JavaScript into machine code) that turns JavaScript from a highly specialized language into a general-purpose language [5]. Express is a minimalistic and flexible web framework for Node.js applications, providing an extensive set of functions for mobile and web applications [6]. JavaScript is a full-fledged dynamic programming language that is applied to an HTML document and can provide dynamic interactivity on websites [7].

The application will consist of two pages: the main page with the listed items of the studied technologies and various CSS properties, and a dynamically loaded page with the game itself, which characterizes the selected technology.

Next, we will describe the layout technologies and properties that are used in the application and which people have most problems with. In CSS, there are 4 most problematic technologies: flexbox, grid layout, working with animation, positioning elements on a web page using the position property.

Flexbox is a new way to place blocks on a page. This is a technology created specifically for the layout of elements, as opposed to floats. With Flexbox, you can easily align elements horizontally and vertically, change the direction and order of elements, stretch blocks to the full height of the parent, or nail them to the bottom edge [8]. But there is one aspect – this technology is not supported by all browsers:

1) IE: 11.0, 10.0 -ms-;
2) Firefox: 28.0, 18.0 -moz-;
3) Chrome: 29.0, 21.0 -webkit-;
4) Safari: 6.1 -webkit-;
5) Opera: 12.10 -webkit-;
6) iOS Safari: 6.1 -webkit;
7) Opera Mini: 8;
8) Android Browser: 4.4, 4.1 -webkit;
9) Chrome for Android: 44.

Problems with this technology occur when the support for older browsers plays a huge role, in this case, you need to abandon this technology. However, at the moment the development is moving towards using new technologies rather than supporting older browsers. It should be noted that many users do not know what properties exist and how to use them for the correct arrangement of elements. There is both the old syntax and the new one, and each of them has its own aspects and this also needs to be sorted out. Thus, all these aspects will be considered in the developed application.

The next technology to be considered is Grid layout. Grid layout-brings a two-dimensional layout tool to the web, with the ability to place elements in rows and columns. A CSS grid can be used to achieve many different layouts. It divides the page into large regions, or defines relationships in terms of size, position, and layer, between parts of the control built from HTML primitives. Like tables, Grid layout allows the author to position elements in columns and rows. However, unlike tables, Gridlayout does not have a content structure, so it is not possible to include a large number of layouts in tables [9].

CSS3-animation makes sites dynamic. It animates web pages, improving user interaction. Unlike CSS3 transitions, animation creation is based on keyframes, which allow you to automatically play and repeat effects for a given time, as well as stop the animation inside the loop.

CSS3 animation can be used for almost all html elements, as well as for pseudo-elements :before and :after. When creating an animation, you should not forget about possible performance problems, since changing some properties requires a lot of resources [10].

Next, consider the CSS position property, which sets the way the element is positioned relative to the browser window or other objects on the web page. This property takes 4 basic values:

1) Absolute – it indicates that the element is absolutely positioned, while other elements are displayed on the web page as if there was no absolutely positioned element. The position of the element is set by the left, top, right, and bottom properties, and the position property of the parent element also affects the position. So, if the parent value of position is set to static or there is no parent, then the coordinates are counted from the edge of the browser window. If the parent's position value is set to fixed, relative or absolute, then the coordinates are counted from the edge of the parent element;

2) Fixed – this property is close to absolute in its action, but unlike it, it is bound to the point specified by the left , top, right, and bottom properties on the screen and does not change its position when scrolling the web page.

3) Relative – the position of the element is set relative to its original location. Adding the left , top, right, and bottom properties changes the item's position and moves it one way or another from its original location;

4) Static elements are displayed as usual. Using the left, top, right, and bottom properties does not produce any results.

5) Inherit-inherits the value of the parent [11].

Thus, the above- described CSS technologies and properties will initially be included in the web application being developed in order to eliminate developers' weaknesses, as well as to introduce and provide the correct knowledge base for users to work with these technologies.

Conclusion. All the nuances described in the article will be taken into account in the developed web application. When developing software, most people are looking for certain technologies that will fit their criteria. If you consider JSX as a way to use it together with ReactJS, it has a number of advantages that can be supplemented with various libraries, which will help you quickly and efficiently develop software. In the future, it is planned to refine this application by adding new CSS technologies.

REFERENCES

1. Web2Studio.Yarkiye resheniya(Bright decisions) Available at: http://webstudio2u.net/ru/programming/127-programming.html (Accessed at 07.02.2020).
2. Habr. Interesnyye stat'i po programmirovaniyu (Interesting programming articles)Available at: https://habr.com/ru/post/311226 (Accessed at 01.02.2020).
3. Habr. Igry, kotoryye uchat programmirovaniyu (Games that teach programming) Available at: https://m.habrahabr.ru/post/273003 (Accessed at 07.02.2020).
4. ReactJS / Wikipedia– 2020. Available at: https://ru.wikipedia.org/wiki/React(Accessed at 08.02.2020).
5. NodeJS / Wikipedia. – 2020. Available at: https://ru.wikipedia.org/wiki/Node.js(Accessed at 08.02.2020).
6. Express. Freymvork veb-prilozheniy Node.js (Web application framework Node.js)– 2020. Available at: http://expressjs.com/ru. (Accessed at 08.02.2020).
7. Osnovy Javascript/MDN (Javascriptbasics/MDN) – 2020. Available at: https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/JavaScript_basics (Accessed at 08.02.2020).
8. CSS 3 flexbox. Opisaniye vsekh svoystv modulya, primery verstki(Description of all module properties, layout examples – 2020. Available at: https://html5book.ru/css3-flexbox (Accessed at 12.02.2020).
9. CSS Grid Layout/MDN– 2020. Available at: https://developer.mozilla.org/ru/docs/Web/CSS/CSS_Grid_Layout (Accessed at 13.02.2020).
10. CSS3-animatsiya(CSS3 animation)– 2020. Available at: https://html5book.ru/css3-animation. (Accessed at 14.02.2020).
11. Position / htmlbook.ru– 2020. Available at: http://htmlbook.ru/css/position. (Accessed at 14.02.2020).