

HTML5, future to solve cross –platform issue in serious game development

Mazen Ismaeel Ghareb
Computer Science Department,
University of Human Development
Sulimanya, KRG, Iraq
mazen.ismaeel@uhd.edu.iq

Abstract— serious games as a learning medium have progressed in the previous couple of years. They have been connected to support learning in different fields, for example, security, medical services, and instruction. Serious games can scale from low spending recreations up to high spending diversions relying upon the recreations' destinations and elements. For example, military may use recreational software with 3D reenactment. This is the majority of scenarios in learning environments focusing on motivation to learn. Nevertheless, game development stays as a tedious, complex, and difficult process. With the use of HTML5, however, there is a growing shift towards web apps as opposed to native code. HTML5 technologies are enabling game applications to run in the browser with some native app functionality. HTML5 has received a great deal of attention since its release in 2009, there are numerous articles and discussions on the Internet. In this work, we present how we can use HTML5 canvas as game engine and use it as a cross platform engine for developing a serious game with implementation of some simple games. We mention the advantage and simplicity of canvas HTML5 for web game development. In addition, we review further cross-platform compilers that support HTML5. Finally, we make a comparison of 10 of cross platform apps that support HTML5 in game development in order to help specify which cross platform can be used in game development generally and serious game specifically.

Index Terms— HTML5, Serious game, E-learning, Web programming, Java script, Cross platform.

I. INTRODUCTION

In the past two decades, markup languages have evolved as a form of web communication and have attained recognition because of their ability to create, capture and share information over the World Wide Web (WWW). The commercial use of Markup Languages has led to exponential competitive benefits. In the current competitive business world, the web standard which has gained most attention among the web development community is HTML 5, the latest in the evolution of the web markup languages. HTML 5 is a development platform upon which very successful web based games

can be developed, especially when making use of what are called 'game engines'. Game engines are frameworks, which can be used to develop games. A game engine facilitates such development, so as to easily implement core functionalities such as video, animations, sound, physics, collision and many more [1]. Each game engine is unique and designed to achieve a specific set of functionalities within a game [2]. A major benefit of using HTML5 game engines is that they are very lightweight and modular and support JavaScript-friendly syntax [3]. Moreover, HTML 5 game engines can perform complex tasks which are required when building social games. Before the popularity of HTML 5, social games were developed with an emphasis on scripting and animation platforms. However, HTML 5 caught the attention of developers due to its rich features. More research is focused on games that can be utilized for something more than only fun, for something great and different. Evidently, games can make us smarter; more advantageous or even help us improve mind functions [4]. These diversions are alleged genuine amusements, an ironic expression by the name. Still, as characterized by Michael and Chen, serious games are recreations that don't have diversions, satisfaction or fun as their basic role[5]. In such definition, be that as it may, could likewise conceal a reason that the serious game configuration is confronting the difficulties of making a drawing in game play as it puts more focus on the most proficient method to devise interactions, learning, and so on than otherwise. Experience demonstrates that it is frequently a difficult expectation taking in a charming, energizing, and enhancing background [6]. Therefore, In this work, we present how the use of HTML5 canvas as game engine, and as a cross platform engine for developing a serious game with implementation of some simple games. It is interesting to mention the advantage and simplicity of canvas HTML5 for web game developments. Additionally, we review cross-platform compilers that support HTML5, and make a comparison of 10 of cross platform apps that support HTML5 in

game development. This is so that we can specify which cross platform app can be used in game development generally and serious game specifically.

II. BACKGROUND

Today's mobile market offers an unlimited opportunity, in terms of variety and platforms requiring these particular platform support for the product. Cross platform implies that the product ideas are implemented and worked on multiple platforms [7]. Such sort of programming might be isolated into two types:

- Requiring singular compiling and building for each supported device
- Straightforwardly running on any stage without extraordinary planning, e.g. a product written in a specific language or a pre-compiled portable byte code for which the interpreter functions at run-time or as a plugin -these are basic (or standard) segments of all platforms [7]. Utilizing distinctive languages requires different improvement environment. For instance iOS requires Objective-C1with Xcode2 and then again Android requires JAVA3 with upheld IDEs. The minimum environment can be recognized as: the web browser, a hybrid of Web and other components , and self independent environments[9]. Applications utilizing the web browser as a run-time environment are normally called mobile web applications. Exploiting base mobile browser support, there is typically a single version required for various sorts of devices. This results in two principal issues:

1. They can't be introduced on the telephone as a different application .
2. They don't have access to particular telephone functionalities and gadgets like notifications, use of inherent equipment or cross application correspondence. On the other hand Hybrid approaches rose up out of the absence of local functionalities [10]. They worked like the already said web applications utilizing the program run-time environment, however, likewise are wrapped in a local engine. This additional layer permits making Programming interface call particular for the gadget permitting exploiting hardware possibilities. Finally self independent environments are attempted starting with no outside help and not in view of any other beforehand existing engines. These are planned and created by the necessities of applications they will be utilized in. A case of such a domain is Adobe Incorporated Runtime4, created by Adobe Frameworks for building Rich Web applications (RIA) that can be kept running as desktop applications or on cell phones [11].

A. Types of Cross Platforms compilers

Cross platform compilers are the tools that focus on translating a single source code into local applications. These instruments connect the prerequisites of the local

gadget APIs with the picked programming languages of the single source code [12]. They permit wrapping up a prior created HTML5 application and output a package accessible for transfer to various stages. Figure 1 shows a different type of cross platforms.

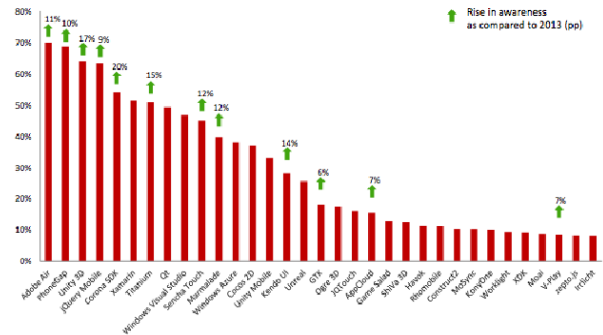


Figure 1: Cross-platform tools known by developers [13]

B. HTML5

HTML5(Hyper Text Markup Language) is increasingly used to get ready sites to be rendered by a web program. It depicts each component contained on a single page by set of given particular labels, permitting to control them and set up as sought by the client. After just about 10 years of HTML4, which was for the most part centered around static presentations, in 2008 W3C (World Wide Web Consortium) introduced a working draft of HTML5 (last discharge has been making arrangements for 2014). The fundamental contrast between those two variants is the emphasis on web applications. With the advancement of the Internet the pages have started to get more intuitive and focus is on the client created content [14]. Among this rich assortment of elements, one needs unique consideration: canvas. It is a component that takes into consideration dynamic shapes and bitmaps rendering while utilizing JavaScript. With its picture control functionalities it encourages the era of diagrams, presentations and amusements specifically in the program, not requiring a module, for example, Blaze. It is critical for the cross stage implies, that canvas component is upheld by practically every program on stationary machines and cell phones [14].

C. JavaScript

JavaScript is an interpreted programming language with Object Oriented capabilities. It was created at Netscape Communications in the mid 90's. In spite of some naming, syntax, and standard library likenesses, JavaScript and Java are generally random and have altogether different semantics. The punctuation of JavaScript is derived from C, while the semantics and configuration are not - Plan programming languages

Adobe (2007). It is normally utilized as a part of web program, and in that unique situation, permitting the scripts to connect with the client, control the web program and adjust the archive content that shows up in the web program window. It is regularly called "client side" JavaScript to accentuate that scripts are controlled by the customer PC as opposed to on the server. In the mid 2000's node.js was discharged, which presents JavaScript likewise as a server side language. The language developed rapidly and has been standardized by the European Computer Manufactures Association (ECMA). Official name of JavaScript, as per ECMA-262 standard is ECMAScript.

D. HTML 5 Canvases

The canvas component is a component defined in HTML code utilizing width and height attributes. The genuine force of the canvas component, be that as it may, is proficient by exploiting the HTML5 Canvas Programming interface. This Programming interface is utilized by composing JavaScript that can get to the canvas territory through a full arrangement of drawing capabilities, therefore taking into account progressively produced illustrations. The HTML5 Canvas' list of capabilities is a perfect possibility for delivering a wide range of 2D and 3D games. Every HTML5 canvas component must have a specific circumstance. The setting defines what HTML5 Canvas Programming interface, you'll be utilizing. The 2d setting is utilized for drawing 2D illustrations and controlling bitmap pictures. The 3d setting is utilized for 3D representation, creation and control [15].

E. What is a serious game

Serious games depend on associating a genuine reason to learn and advancements from the computer game industry [16]. A later definition is that Serious games are amusements that don't have diversion as their main role [17]. Serious games traverse over an extensive variety of fields, for example, instruction, treatment, promoting, protection, exploration, and so on and in light of the fact that there is no accord on the space limits for Serious games, a meaning of Serious games is still an open subject [18]. Some "particular" definitions mirror these distinctions and are utilized to compel a restricted perspective of the way of "Serious games" [19]. Still, two normal perspectives that were distinguished in an audit of new definitions which incorporate every single amusement classification and application fields are that 1) a Serious game has a reason beyond stimulation and that 2) the target of a Serious game is to utilize the alluring states of the diversion to fill a genuine need [20]. Along these lines we abandon this classification if the "other intention" is superimposed to stimulation or is fairly parallel to it. Besides, this definition empowers us to likewise incorporate diversions used to perform "reason moving",

that is recreations that are not planned with the main role of "seriousness", however their utilization case can be the same as with genuine amusements.

III. IMPLEMENTATION WITH HTML5 CANVAS

We had experience creating a serious game with C# called JOJO and we have faced many graphical plug-in issues [39]. Later, our two groups of student's final year project was to build a prototype of the game using HTML5 canvas. One prototype for creating a tutorial for learning Kurdish language by an idea of falling objects: a game for children aged 4-6 requiring them to recognize objects and how to spell them. They use Java Script with HTML5 as shown in figure2.



Figure 2: HTML5 Learning Kurdish Canvas game Ahmed , Shahad, Nadia

Another prototype was created as a web page using canvas and Java Script to solve puzzles, and learning writing (Kurdish Language) and give the user feedback of its scores. The web application uses HTML5 canvas CSS3 and Java script. As it is shown in figure3, and figure 4 respectively.



Figure 3: HTML5 learning Kurdish writing game application Hiwa , Karman , Karwx

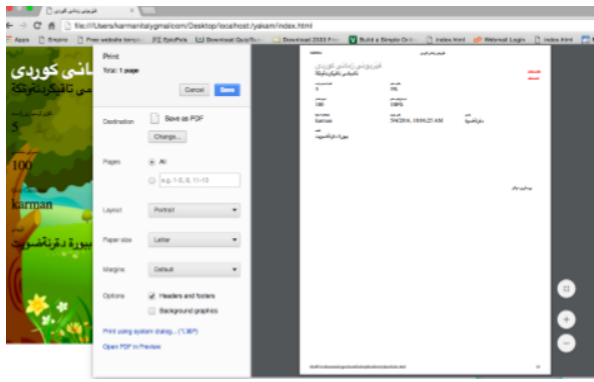


Figure 4: HTML5 learning Kurdish writing game application Hiwa , Karman , Karwx

In addition to these, there is a simple game of fruit collector created by canvas that counts the fruit you collect and if you collect a bomb the game will be over. As it is shown in figure 5.

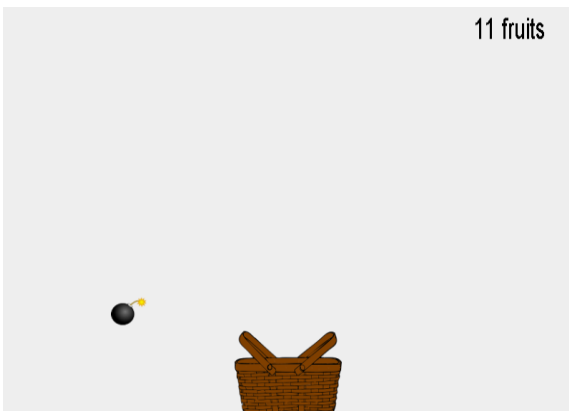


Figure 5: Fruits collector canvas game

So in previous examples we shows how we can use canvas to build different kinds of games and applications so the main advantages are Canvas as a feature of HTML5 is thoroughly free and open source! That is to say, that the designers don't require costly programming like 'Adobe Flash' to really code more modules! You don't have to assemble anything. Therefore this allows quick and ready implementation. Which leads us to another enormous favorable position due to the canvas component. It essentially needn't bother with any program modules either like Adobe Flash Player. Modern programs move realistic errands to a GPU for speedier handling rather than a CPU. In such a circumstance, it is the CPU that assumes control of different errands all the more effectively, so general execution is speed-ed up which was what had been experienced in the above projects. Since a prompt mode framework doesn't keep up its own model, this allows for a lot of adaptability. Since your code controls all parts of when and how something is attracted to the screen. Great for managing numerous components.

Contrasted with a held model framework where each and every expansion of your scene takes up additional memory, prompt mode frameworks don't have that issue. Despite the fact that HTML5 is an open standard that can be actualized in each program, its support and execution leaves a considerable measure to wish for. Old programs, as IE8 and more established, don't support the canvas component. These programs still have a considerable piece of the overall industry. The execution is likewise an issue, particularly with games that have a considerable measure of enhancements like partial frameworks; the execution fluctuates incredibly among various browsers. The absence of good development tools is additionally an issue, while this is something that will resolve itself after some time as new tools and updates are available; the present devices accessibility can't contrast themselves and an improvement domain like Visual Studio or Adobe Flash Professional. The 2D drawing API of canvas doesn't have the elements required to make complex enhancements for recreations, it's outfitted more towards empowering the designers to draw dynamic substance like diagrams than making games. Flot is a decent illustration If you need more propelled drawing highlights like shaders or even a basic multiplicative mix, you have to move to WebGL. Moving to WebGL gives you the favorable circumstances with more design elements and a higher execution, yet it has less support crosswise over browsers. It can ease back when attracting to huge ranges. How rapidly a redraw finish is dependent on to the quantity of pixels you are re-drawing, for example, if you are addressing a large area, things could get slower in the event that you are not watchful and don't upgrade appropriately. It is complex. Since you are taking care of a greater amount of what it takes to animate something to show on the screen, there are significantly more detail elements for you to keep track of. This is also true for serious games because they are more complex than traditional games so the canvas can not support possessing many objects at the same time. Also at the University of Human Development we have created a serious game called JOJO using C# with some graphical plug-ins but it was too complicated to build a serious game out of it [39].

IV. CROSS PLATFORM REQUIREMENTS COMPONENTS

According to the building and experience of developing some game with HTML5, we recommended these components each cross platform should contain this is also regarding building serious game:

- Multiple OS support and mobile support: The platform must support several OS and mobile platform such as android and iOS.
- Free of Charge: it will attract the developers if the cross platform will open source.

- User interface: native cross platform does not support rich User Interface such as 2D and 3D animation and multimedia should be supported.
- Security : most of cross platform not fully secured it need more research to support this issue.
- Support of plug-extension: it is required to install plug in on top of existing applications for more compatibilities with new requirements.
- Accessing build in devices and features: it is important that the tool able to access builds in features and devices such as camera, sensors and geo-locations.

V. CROSS PLATFORM COMPARISON

There are many of game engine written in JavaScript and different languages that can be used with HTML5 to build a platform in depended games. Some of these engines will be discussed here:

- I. Impact: It is a JavaScript system that encourages the engineers to build HTML 5 game for Smartphones and desktop programs [21]. The game engine gives distinctive layers to chip away at, and each can be stacked on top of another. Every layer can be upgraded with special conduct and examples.
- II. PlayCanvas: is an open source game engine having powerful instruments to create 3-D games. The game engine is a web facilitated and can be utilized on the cloud furthermore introduced on an individual workstation. As, it runs altogether on JavaScript and is 100 percent subject to HTML5, games created on PlayCanvas can be kept running on cross platforms. Designers can assemble 3D models and fare these to FBX record arrangement, and afterward this document can be utilized as a part of PlayCanvas with the assistance of simplified components [22].
- III. Turbulenz: is a game engine, extraordinarily intended to create social HTML 5 games. Turbulenz gives a Software Development toolbox (SDK) which incorporates nearby gaming server, documentation, resource handling apparatuses and tests [23]. It has a JavaScript engine, which contains immensely essential administrations, for example, illustrations, material science, sound, liveliness and systems administration. Turbulenz is additionally furnished with APIs to give functionalities, for example, a pioneer board, spare recreations, identifications, companion list, talks, installments and difficulties.
- IV. CoCos2d-JS-It is encourages the game developers to create games for HTML5 perfect programs and JavaScript Bindings (JSB). The APIs utilized as a part of Cocos2d-JS are gotten from Cocos2d-x. Cocos2d-JS comprises of Coco2d-html5 and Cocos2d-x JSB [24]. Cocos2d-x is an MIT authorized (Massachusetts Institution of Technology) and open source game engine. Engineers can create intelligent cross programs recreations, applications and other intuitive projects [24].
- V. Game Closure: is created totally with JavaScript. It is uncommonly intended to work with HTML5. Amusement in Diversion Conclusion can be produced in any content tool and can be tried on all web programs. This gives engineers the chance to utilize progressed investigating apparatuses exhibit in programs like Google Chrome and Mozilla. The engineers can compose a single code for iOS and android and can utilize a single command to make an establishment bundle for separate working frameworks. It gives the development kit known as DevKit, which have a web recreation for moment testing of the games for iOS or android environment [25].
- VI. Gamvas: It is an open source game engine authorized under MIT. It is a 2D game engine uniquely intended to work with HTML5. It is composed in JavaScript, and it is most appropriate for business utilize. It has Box2D material science reenactment, which permits the game characters and protests having qualities, for example, grinding, mass and power. Gamvas support asset administration that educates the designers concerning number of assets they have utilized. On the basis of this data, the engineers can plan a stacking screen in the game . This loading screen may demonstrate a "progress bar" to portray the ideal opportunity for loading the game[26].
- VII. Kiwi.js: It is an open source game engine uniquely intended to create HTML5 portable and desktop games , it supports WebGL for 2D and 3D rendering. It utilizes cocoonJS to make and distribute the HTML5 recreations [27].
- VIII. CraftyJS: is created in JavaScript and support HTML5. The engineers have the decision to pick "Canvas" or Document Object Model (DOM) which allows for developers to make the games with most appropriate advancements (canvas or DOM).The favorable position of building up the game in CraftyJS is that the record size of the source is sufficiently little that the development does not have to stress over dealing with the network bottleneck. It supports software architecture, product design (SOP). SOP is a usage of the piece based programming utilizing a database structure as opposed to implementing inheritance (Nystrom 2014).
- IX. LycheeJS: is likewise a JavaScript engine for creating 2D games. It supports canvas, simple

DirectionMedia Layer (SDL), OpenGL and WebGL for activity rendering. LycheeJS permits the improvement for all present day program and equipment stages (mobiles, gaming consoles and Desktops) [29].

- X. melonJS: is additionally an open-source JavaScript motor. It supports multi-channel HTML sound for installing the sounds. It is given a "state manager" effectively to manage game states, menu, stacking and state exchanging. It also has a component for move impacts that is critical in the game visuals and online. It is additionally given extremely advanced math Programming interface for vector and network components. In any case, the disadvantages of it is that it doesn't support WebGL, and there is no immediate support for 3D activity [30].

From the investigation of various game engines, a comparison matrix is drawn.

In the below table it compares 10 platforms, according to several games developed by final year projects. These were chosen because all supported or working as a plugin with HTML5 so it depended on the specific project which of these language to use but in general it was impact which was the factor for best choice.

A. Other game engines:

- Construct: is a game development instrument uniquely intended to take a shot at "Windows". It is an effective instrument for making 2D recreations. Designers won't bother with any specific experience, instruction or programming information. It is intended to be visual, natural and quickly understandable [32].
- CreateJS: is an open source suite, which is supported by Adobe, Microsoft, AOL, Mozilla and gskinner. It gives JavaScript devices and Libraries to making intelligent recreations that can be distributed on the web by means of HTML5. It has four noteworthy libraries which incorporate EASELJS, TWEENJS, SOUNDJS and PRELOADJS. [33]. These JavaScript libraries can be utilized to create enhanced and intuitive HTML5 recreations.
- Pixi.js : is acclaimed for its capacity to deliver rapid 2D rendering [34]. It is a rendering engine which bolsters cross-platform game advancement and WebGL (Pixi.js 2013d). Pixi.js additionally supported by multi-touch intelligence for the touch-screen smart devices. Pixi.js support distinctive visual impacts, for example, tinting and blend modes.
- GameMaker Studio: is most appropriate for amateur and section level diversion engineers. It is a quick to-create structure which gives cross-platform game creation. Before making the game , the designers can make the models rapidly and can draw an

essential thought regarding the yet-to-create a diversion [35]. GameMaker Studio gives send out element to convey recreations for consoles, mobiles and desktops as prepared to-run executable applications. It can be published online by trading the game in HTML5 . GameMaker is equipped with YoYo Compiler (YYC) which bridle CPU assets proficiently and in the long run builds the execution of the amusements. GameMaker is most appropriate to create business diversions, and it adapts the amusements.

- Phonegap: is free, open source and based on open standards. In 2014 Phonegap opened Phonegap Build6service as a team with Adobe. This administration offers one private application for nothing. From 9.99\$ it is conceivable to extend the measure of applications to 25. Else they are facilitated on an open GitHub archive. Adobe ID's are likewise used to Phonegap Build. It is supported by iOS, Android, Blackberry OS, Windows Phone, Ubuntu, Firefox OS.

B. HTML5 game Performance and Productivity:

All through the historical backdrop of the web, third party program , for example, "flash Player", "Skype Snap to Call" and numerous others have assumed a critical part in empowering a web-program to display the online multimedia to the clients. However, plug-ins have burdens as they keep running with the help program support and utilize additional framework assets. Additionally, they make the framework more open to outer security assaults. The other hindrance of modules is that they are not intended to bolster touch reaction that is extremely vital in current touch devices. Likewise, plugging are not open source, but rather restrictive advancements that make it troublesome for program merchants to make their programs to backings various types and sorts [37].

The answer for cutting edge touchscreen and program similarity issues was at first included the HTML5 particulars. This element in HTML5 detail is known as "HTML5 Storage", "Local Storage" or "Web Storage". In any case, because of inward choices inside W3C, "Web Storage" was then characterized in its autonomous detail . "HTML5 Storage" is a procedure in which site pages can locally store information key/esteem matches on the customer program. The key esteem sets continue even after the client moves to another tab/window or even subsequent to shutting the web program . HTML5 game engine outfit the "web Storage" property that HTML5 gives. "Web Storage" effectively nearby tenacious stockpiling and session stockpiling and it utilizes bigger customer side stockpiling when contrasted with cookies. "Web Storage" gives a limit of 5 Megabytes for every area for local storage. The upside of utilizing "Web Storage" is that it offers powerful apportioning of steady and session stockpiling and

permits the programs to store just those information things those are essential for local use. Data items that are transient and not pertinent after they get to be not put stored [38].

VI. CONCLUSION

This paper deals with the question of how we can use HTML5 canvas as game engine and use it as a cross platform engine for developing a serious game. Also mentioning the advantage and simplicity of canvas HTML5 for web game developments. However HTML5 canvas has many drawbacks regarding building more complex and attractive games with AI, UI, 2D and 3D as it is hard to be supported only by canvas. HTML5 is a markup language which will keep running on any device on the off chance that equipped with a modern browser. HTML5, CSS, JavaScript and the Bootstrap system (HTML5 based rendering structure) give components, for example, which support web SWL database, video and sound tags, support for movement and support for adaptable vector design etc. Newer HTML5 labels give support to responsive plan which helps a page to naturally alter its view as per a device's screen measure. Since the whole code base of HTML is kept on a remote server, HTML5 based web applications give top of the line memory administration capacity. HTML5 base applications can be created utilizing different programming models which can be executed crosswise over devices or working frameworks, for example, the web MVC. Moreover, it is not supported by different platforms such mobile devices. So we have conducted a cross platform comparison between 10 game development platforms that support HTML5. It was during this comparison it became clear that several of these platforms can be used in game programming especially serious games with all their components. For instance, ImpactJS have the preferred standpoint that it preloads all the game resources, for example, pictures, sounds, elements and their states before executing the game. ImpactJS have other propelled components, for example, CocoonJS, quickened HTML5 execution, testing and investigating and a decent documentation for the novice to create proficient games. Moreover, we show the components of designing a serious game and web game in general. The research made with the end goal of this paper may help future game engineers to focus on more proficiency driven choices amid the plan and improvement phase. In spite of the complexities of keeping various materials accessible for research, and subjects of inquiries, a top down approach is more useful. One of such cases might render utilizing distinctive JavaScript engines like V8 utilized by Chrome or Nitro as a part of Safari. Such tasks may require further study and comparisons of engines, and going deeper inside assemblage procedures of different programs. Finally we conclude that HTML5 canvas can not developed complex serious games with many 3D or

2D interactive unless we use some support platform work with HTML5 in order to complete all the components of the game. In future work we will use one of these platforms to develop education games for learning Kurdish Language.

VII. ACKNOWLEDGMENT

I would like to thank very much the staff of University of Human Development College of Science and Technology – the Department of Computer Science. Special thanks to final year students: Shahad, Ahmed, Nadia ,Karman ,Karwx and Hiwa for their final year projects as web game applications and I mention their work in this paper.

REFERENCES

- [2] E. Lengyel, *Game Engine Gems 2*. CRC Press, 2011.
- [3] A. Vahldick, A. J. Mendes, and M. J. Marcelino, "Towards a Constructionist Serious Game Engine," in *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016*, 2016, pp. 361–368.
- [4] D. Laurie, W. Wisden, and P. Seeburg, "The distribution of thirteen GABAA receptor subunit mRNAs in the rat brain. III. Embryonic and postnatal development," *The Journal of neuroscience*, vol. 12, no. 11, pp. 4151–4172, 1992.
- [5] O. Sourina, Q. Wang, and M. K. Nguyen, "EEG-based 'serious' games and monitoring tools for pain management," *Studies in health technology and informatics*, vol. 163, pp. 606–610, 2010.
- [6] T. Susi, M. Johannesson, and P. Backlund, "Serious games: An overview," 2007..
- [7] B. Huynh-Kim-Bang, J. Wisdom, and J.-M. Labat, "Design patterns in serious games: A blue print for combining fun and learning," *Project SE-SG*, available at <http://seriousgames.lip6.fr/DesignPatterns>, 2010.
- [8] R. Levitt and J. M. Piro, "Innovation in Education through Web-Based Instruction: Digital and Cross-Platform Storytelling," *Revolutionizing Education through Web-Based Instruction*, p. 131, 2016.
- [9] S. Nilsson, "Implementation of a Continuous Integration and Continuous Delivery System for Cross-Platform Mobile Application Development," 2016.
- [10] C. Rieger and T. A. Majchrzak, "Weighted Evaluation Framework for Cross-Platform App Development Approaches," in *EuroSymposium on Systems Analysis and Design*, 2016, pp. 18–39.
- [11] A. Monzavi and R. Darwish, "Comparison of HTML5 and Native Client Development on Mobile Platforms," 2015.
- [12] J. Ross, "System and method for integrated hardware platform for flash applications with distributed objects," 2013.
- [13] P. R. de Andrade, A. B. Albuquerque, O. F. Frota, R. V. Silveira, and F. A. da Silva, "Cross platform app: a comparative study," *arXiv preprint arXiv:1503.03511*, 2015.
- [14] G. Anthes, "HTML5 leads a web revolution," *Communications of the ACM*, vol. 55, no. 7, pp. 16–17, 2012.
- [15] M. Pilgrim, *HTML5: up and running*. "O'Reilly Media, Inc.," 2010.
- [16] S. Fulton and J. Fulton, *HTML5 canvas*. "O'Reilly Media, Inc.," 2013.
- [17] C. Girard, J. Ecalle, and A. Magnan, "Serious games as new educational tools: how effective are they? A meta-analysis of recent studies," *Journal of Computer Assisted Learning*, vol. 29, no. 3, pp. 207–219, 2013.

- [18] vZavcer, S. Mayr, and P. Petta, "Design pattern canvas: An introduction to unified serious game design patterns," *Interdisciplinary Description of Complex Systems*, vol. 12, no. 4, pp. 280–292, 2014.
- [19] F. Laamarti, M. Eid, and A. E. Saddik, "An overview of serious games," *International Journal of Computer Games Technology*, vol. 2014, p. 11, 2014.
- [20] S. Mader, S. Natkin, and G. Levieux, "How to analyse therapeutic games: the player/game/therapy model," in *International Conference on Entertainment Computing*, 2012, pp. 193–206.
- [21] S. McCallum and C. Boletsis, "A taxonomy of serious games for dementia," in *Games for Health*, Springer, 2013, pp. 219–232.
- [22] impactjs, "impactjs." [Online]. Available: <http://impactjs.com/>. [Accessed: 01–08-2016].
- [23]
- [24] playcanvas, "playcanvas." [Online]. Available: <https://playcanvas.com/>. [Accessed: 01–08-2016].
- [25] Turbulenz, "Turbulenz." [Online]. Available: <http://www.turbulenz.biz/>. [Accessed: 01–08-2016].
- [26] Turbulenz, "Turbulenz." [Online]. Available: <http://www.turbulenz.biz/>. [Accessed: 01–08-2016].
- [27] cocos2d-x, "cocos2d-x." [Online]. Available: <http://www.cocos2d-x.org/>. [Accessed: 01–08-2016].
- [28] gameclosure, "gameclosure." [Online]. Available: <http://www.gameclosure.com/>. [Accessed: 01–08-2016].
- [29] gamvas-wcb, "gamvas-wcb." [Online]. Available: <http://www.indicdb.com/engines/gamvas-wcb/>. [Accessed: 01–08-2016].
- [30] Lychcc, "Lychcc." [Online]. Available: <https://Lychcc.js.org/>. [Accessed: 01–08-2016].
- [31] melonjs, "melonjs." [Online]. Available: <http://melonjs.org/>. [Accessed: 01–01-2016].
- [32] construct2, "construct2." [Online]. Available: <https://www.scirra.com/construct2>. [Accessed: 01–08-2016].
- [33] Scirra, "Scirra." [Online]. Available: www.scirra.com.
- [34] Createjs, "createjs." [Online]. Available: <http://www.createjs.com/>. [Accessed: 01–08-2016].
- [35] Pixijs, "Pixijs." [Online]. Available: <http://www.pixijs.com/>. [Accessed: 01–08-2016].
- [36] gamemaker, "gamemaker." [Online]. Available: <http://www.yoyogames.com/gamemaker>. [Accessed: 01–08-2016].
- [37] phonecgap.com, "phonecgap." [Online]. Available: <http://phonecgap.com/>. [Accessed: 01–08-2016].
- [38] Microsoft, "Get ready for plug-in free browsing." [Online]. Available: <http://msdimicrosoft.com/emmau/library/ic/hh968248%28v;vs.85%29.aspx>. [Accessed: 01–08-2016].
- [39] A "What is HTML5 Storage?," [Online]. Available: <http://www.gwtproject.org/doc/latest/DcvGuidcHtm15Storage.html>. [Accessed: 01–08-2016].
- [40] A. M. Darwesh, M. I. Ghareb, and S. Karimi, "Towards a Serious Game for Kurdish Language Learning."