

PAPER • OPEN ACCESS

## Development of career guidance quests using WebAR

To cite this article: D S Shepiliev *et al* 2021 *J. Phys.: Conf. Ser.* **1840** 012028

View the [article online](#) for updates and enhancements.



**240th ECS Meeting** ORLANDO, FL

Orange County Convention Center **Oct 10-14, 2021**

Abstract submission deadline extended: April 23rd

**SUBMIT NOW**

## Development of career guidance quests using WebAR

**D S Shepiliev<sup>1</sup>, S O Semerikov<sup>1,2,3</sup>, Yu V Yechkalo<sup>2</sup>, V V Tkachuk<sup>2</sup>,  
O M Markova<sup>2</sup>, Ye O Modlo<sup>4</sup>, I S. Mintii<sup>1</sup>, M M Mintii<sup>1</sup>, T V. Selivanova<sup>1</sup>,  
N K Maksyshko<sup>5</sup>, T A Vakaliuk<sup>6</sup>, V V Osadchyi<sup>7</sup>, R O Tarasenko<sup>8</sup>, S M Amelina<sup>8</sup>  
and A E Kiv<sup>9</sup>**

<sup>1</sup> Kryvyi Rih State Pedagogical University, 54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine

<sup>2</sup> Kryvyi Rih National University, 11 Vitalii Matusevych St., Kryvyi Rih, 50027, Ukraine

<sup>3</sup> Institute of Information Technologies and Learning Tools of NAES of Ukraine, 9 M. Berlynskoho Str., Kyiv, 04060, Ukraine

<sup>4</sup> State University of Economics and Technology, 5 Stephana Tilhy Str., Kryvyi Rih, 50006, Ukraine

<sup>5</sup> Zaporizhzhia National University, 33A Dniprovska Str., Zaporizhzhia, 69061, Ukraine

<sup>6</sup> Zhytomyr Polytechnic State University, 103 Chudnivsyka Str., Zhytomyr, 10005, Ukraine

<sup>7</sup> Bogdan Khmelnytsky Melitopol State Pedagogical University, 20 Hetmanska Str., Melitopol, 72300, Ukraine

<sup>8</sup> National University of Life and Environmental Sciences of Ukraine, 15 Heroiv Oborony Str., Kyiv, 03041, Ukraine

<sup>9</sup> Ben-Gurion University of the Negev, P.O.B. 653, Beer Sheva, 8410501, Israel

E-mail: semerikov@gmail.com

**Abstract.** Career guidance quests that involve solving intellectual and search problems in a game are associated with one's future career, being one of modern means of recruiting school leavers to vocational and higher educational institutions. Such specific computer tools of virtual and augmented reality as pupils' mobile Internet gadgets can enhance visual character and interactivity of quest problems. The article deals with applying augmented reality in the web environment to solving these tasks. The authors propose a prototype of a career guidance quest system using WebAR technology.

### 1. Introduction

Nowadays, augmented, or mixed, reality is a popular technology which is gaining in popularity among educators as well. The basic application of augmented reality (AR) to education is provision of visualization by visualizing computer models of systems and objects using general (a computer equipped with an integrated or external web-camera) and specific (AR goggles, VR helmets) tools, and adapted devices (mobile phones).

The latter are the most widespread devices among participants of the academic process. As of May 2020, there are 8.3 mln users of mobile games, 26.2% of which are 18-24-year olds [27]. According to 2019 data, Ukrainian mobile users at the age of 16-25 are the most frequent users of social networks



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/3.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

(92%), message exchange tools (66%), video applications (64%) and mobile games (50%) [11]. Comparison of the two latter sources enables drawing a conclusion that there are from 16 to 16.6 mobile users in Ukraine who are senior school pupils, students of all educational levels or young people. According to the Government Statistics of Ukraine, at the beginning of the 2019-2020 academic year, there are 1.440 mln students of colleges, secondary technical schools, universities, academies and institutes [9], 4.163 mln of which are 16-25-year-olds ([9], p. 26).

Comparison of the given statistics indicates that each student has two mobile numbers on average attached to one of the popular stores of mobile applications. It can be either a single mobile phone with two SIM-cards or two mobile phones. As of December 2020, 79.8% of mobile devices in Ukraine had the operating system Android installed [26], while the leading web browser was Chrome (67.8%) [24]. Considering that the third most popular browser, Opera, is built on the Chrome core [21], Chrome accounts for 80%.

In spite of some specific clients for social networking, message exchange, video watching and mobile games, a browser is a universal tool for performing relevant tasks and developing new software in the web-environment quickly. The main advantage of web-clients is their universality as there are Chrome versions for all mobile platforms. Low operating speed of programmes has been the basic disadvantage until recently. Appearance of WebAssembly in March 2017 enabled acceleration of web-programmes and transition of software designed for other platforms into the web-environment.

Access to high-quality libraries of computer vision, image recognition and monitoring (ARToolKit and its web-version JSARToolKit5 [2]) is the key to designing WebAR software. At the same time, joint application of JSARToolKit5 and WebGL reveals that:

- markers applied by JSARToolKit5 are a two-dimensional type of a barcode of low visualization;
- visualization of objects by means of WebGL requires in-depth understanding of principles of 3D graphics and greater amounts of a software code.

Thus, the mentioned WebAR development tools are aimed at enhancing visualization of the teaching process, yet they lack visualization and accessibility themselves, this fact indicating the research problem of selecting and testing software tools of WebAR development designed for beginners mastering the basics of web-design – students of lyceums and junior Informatics students.

The following article sections are aimed at solving the following tasks:

1. Performing a comparative analysis of WebAR designing tools in order to select those suitable for beginners.
2. Designing software prototypes with marker and markerless WebAR.

## 2. Review of WebAR designing tools

### 2.1. *Augmented reality: its notion and hardware for Web*

For the recent 60 years, it has been innovative (trendy, interesting, useful, etc.) to program AR, while its application to the web-environment is conditioned by current development of technology. To start this process, one needs an AR-compatible browser like Firefox or Chrome and a text editor (like mcedit or Sublime). Beginners need basic knowledge of web-design (HTML, CSS and JavaScript), while experience of working with web API and GitHub is also applicable.

Hereafter, by AR, we mean an ability of a device (a mobile device or a web browser) to monitor an image or represent a 3D object over the image. The main aim of AR is to show a computer model online in the real-mode space to establish interaction of a user in the real-mode space and a 3D model in the virtual one.

AR can be both marker and markerless. In case of the marker AR, a device tracks a 2D marker: when it is found, a 3D object is visualized. In the markerless variant, a device searches for a flat surface (a table, a floor, etc.) and place a 3D object there.

Using a camera, AR enables visualization of computer-generated objects in games, marketing and other software as in case of arranging furniture in a living room or trying on new clothes before buying. It is a great opportunity for business as it demonstrates a product before a customer really buys it [16].

There are specific devices designed for AR like helmets and headsets enabling a user to plunge into a modelled environment.

AR supplies a real-life world with 3D models controlled by mobile devices anyplace. Virtual reality (VR) makes a user get involved into a modelled world by means of head mounted devices (HMD).

Interactivity in AR and VR software is provided in a similar way. For example, VR actually uses controllers and occasionally hand-tracking enabling a user to interact with 3D objects inside a scene they are situated in.

Major hazards of using HMD in VR include:

- eye strain;
- dizziness and headaches after using HMD.

Unlike VR, AR does not possess any of these hazards for health. Yet, the ability of users to remain concentrated on what they are doing is disturbing, in particular, while using AR, for safety reasons.

Smartphones with iOS (11 and higher under iPhone and iPad control) and Android (7.0 and higher) are the most widespread type of devices for AR.

For web browsers, like Google Chrome and Mozilla Firefox, AR is applicable if they support WebRTC [18] and WebGL [17].

Microsoft HoloLens is an HMD-like AR headset which is designed currently [15]. Like Google Glass [14], it is designated for corporate use, yet, unlike Glass, it is compatible with Windows 10, not with Android.

## 2.2. Tools of visualizing computer models in the Web

WebGL (OpenGL ES for the Web) is API for 3D graphics in the web browser designed by the Khronos Group Inc [31]. WebGL uses a shader programming language GLSL (OpenGL Shader Language) and is part of an document object model (DOM API) of the browser. All the leading browser designers – Apple (Safari), Google (Chrome), Microsoft (Edge), and Mozilla (Firefox) are members of WebGL Working Group. The current version WebGL 2.0 corresponds to the standard OpenGL ES 3.0 API.

The programme described by WebGL contains both a JavaScript code and a C-like code GLSL. The example of simple software for building a triangle (figure 1) from [32] contains 105 rows of the code comprising five blocks:

1. Preparing a canvas (HTML object canvas) and obtaining a rendering context of WebGL.
2. Defining geometry attributes such as vertexes, indices, etc., and storing them in buffer objects.
3. Designing and compiling software for vertex and fragment shaders.
4. Connecting shader software and buffer objects.
5. Rendering a required image.

Application of WebGL for visualizing computer models in the Web is not the easiest way to do it: the code will account for thousands of rows and require high-level skills in designing 3D graphics by means of OpenGL. Considering the fact that the first research task involves selection of tools for beginners, it is reasonable to use libraries that make application of WebGL simpler.

Table 1 contains results of assessing efficiency of using JavaScript libraries for WebGL objects.

Considering the fact that the conducted analysis is designated for teaching beginners, there are the best-rated non-commercial libraries determined for working with WebGL – Babylon.js, Three.js and A-Frame.

**Babylon.js** [5] enables designing complex 2D-objects by using an essentially smaller code size than that for WebGL: the code size shown in figure 2 is 5-fold smaller than that of the WebGL-code for a simpler scene in figure 1. The designed scene is controlled by simple controlling elements. Unlike other rating participants, Babylon.js is a “physical” mover widely applied to designing computer games for the Web.

**Three.js** is a free 3D graphics library of general application written in JavaScript. On <https://threejs.org/>, one can find many high-quality demonstrations designed by using it [6]. Its author, Ricardo Miguel Cabello, known also as Mr.doob, is one of the WebGL pioneers. For this reason, this library is often used for designing other libraries.



Figure 1. The source code and the result of the simple WebGL programme running.

Table 1. Assessment of efficiency of using libraries for WebGL.

Title	Animation	Native audio	Access to the	Considering laws of physics	WebGL version	WebVR	Import	Export	License	General
A-Frame	+	+	-	-	1.0	+	Many formats (3)	HTML, three.js (2)	Free	10
CopperLicht	+	+	-	+	1.0	-	-	-	Free	5
OSG.JS	+	+	-	-	1.0	+	-	-	Free	5
Three.js	+	+	-	-	2.0	+	Many formats (9)	Many formats (4)	Free	19
Verge3D	+	+	-	+	1.0	+	Many formats (3)	glTF (1)	Commercial	9
Clara.io	+	-	-	+	1.0	+	Many formats (5)	Many formats (6)	Commercial	15
Babylon.js	+	+	-	+	2.0	+	Many formats (5)	Many formats (9)	Free	21

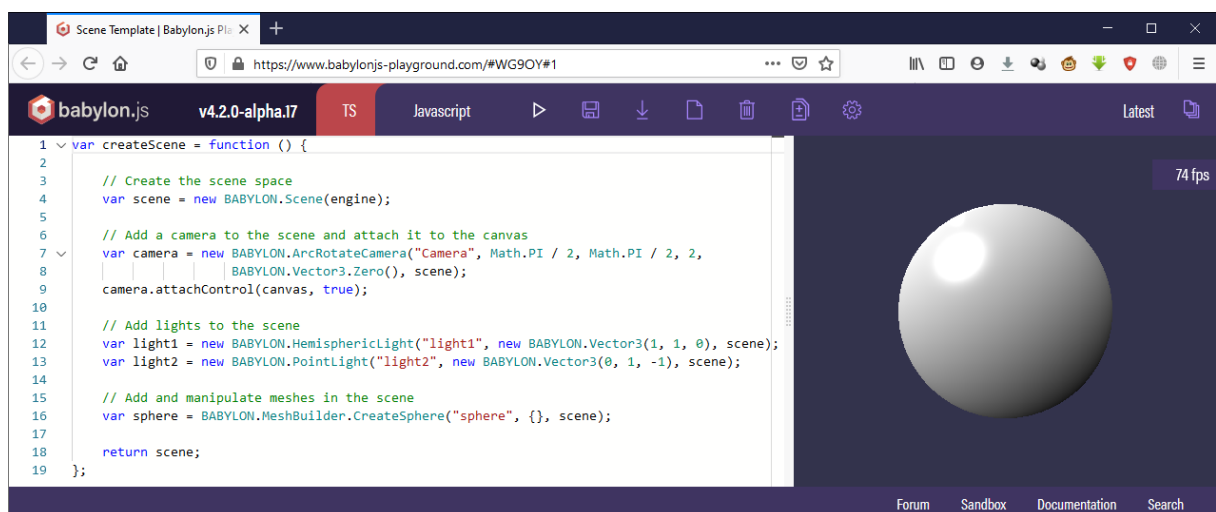
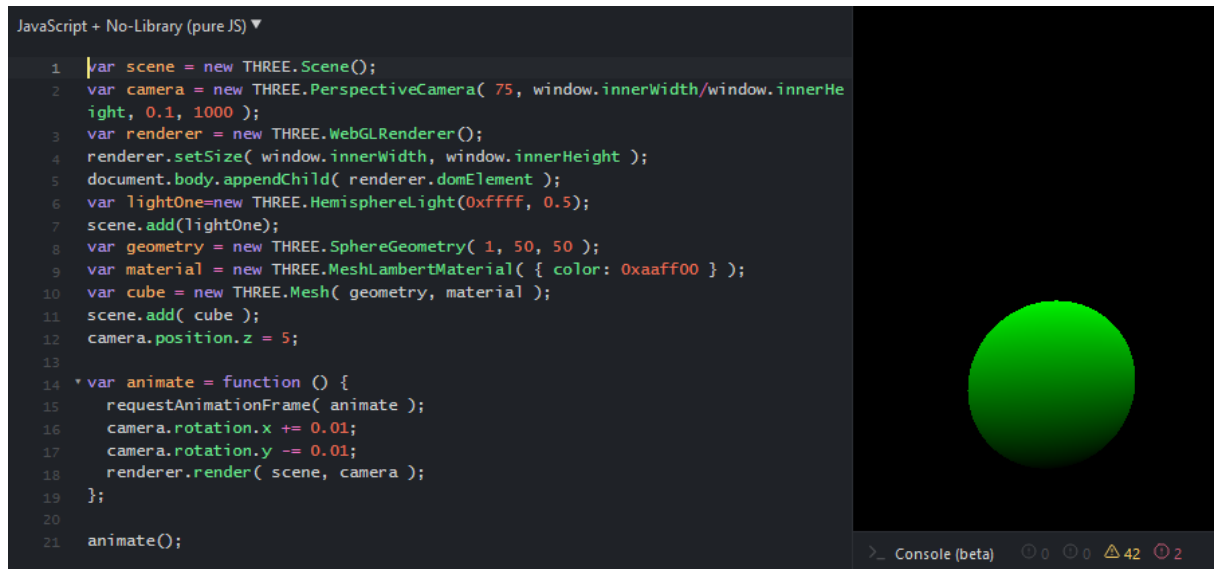


Figure 2. The source code and the result of the simple programme running in Babylon.js. Object designing in Three.js involves three steps:

- 1) determining object geometry – vectors, colours, etc.
- 2) determining materials – a rendering method for an object;
- 3) composing geometry and materials.

The code size required for designing the 3D scene by Three.js (figure 3) is a bit larger than that of Babylon.js. It is explained by combining in Babylon.js the operations of designing an object and those of adding it to the scene into a single call of a constructor of a relevant class.



**Figure 3.** The source code and the result of a simple programme running in Three.js.

**A-Frame** is actually a tool of rapid prototyping: a greater part of its application is a HTML-like code. A-Frame commands are described by tags similar to HTML ones, yet, unlike the latter, are interpreted not in a client-side browser, but are a means of accessing JavaScript on the server side [29].

Figure 4 is significant as the A-Frame code size required for designing the same sphere is a three-fold less than that of Babylon.js/Three.js:



**Figure 4.** The output source code and the result of the simple programme running in A-Frame.

Thus, in spite of the third place in the rating of functional opportunities, A-Frame is the leader in terms of visualization and accessibility among the considered libraries for 3D graphics on the Web. Considering the fact that A-Frame is a superstructure over Three.js, it is reasonable to master them in a loop mode starting from A-Frame.

### 2.3. Tools for tracking real-life objects

As [30] contains comparative analysis of the most widespread AR SDK, let us focus on those applicable to designing on the Web – WebAR SDK. Unfortunately, as of January 2021, there are only three free tools, each of which is associated with considered tools of building computer models.

In August 2019, Babylon.js announced **Babylon AR** [20], the project integrating Babylon.js and the computer vision library OpenCV. At present, there is implemented tracking of standard markers similar to QR-codes. In spite of the fact that designers of Babylon.js consider Babylon AR project being on the early stage, web-programmes designed are efficient only on WebXR-supporting devices [19]. This narrows the scope of applying Babylon AR to 8.1 and higher Android devices. In January 2021, there were 74% of such devices among Android ones in Ukraine [25]. Considering the trend of increasing this share, Babylon AR can be considered a promising project of WebAR software development starting from 2021-2022 [25].

ARToolKit, which is 20 years older than Babylon AR, is one of the most applicable AR libraries. **JSARToolKit** (ARToolKit.js) is currently supporting three types of square markers (with arbitrary drawing, a 2D code and markers) and NFT-markers (natural feature tracking) [2]. JSARToolKit is not specific for a WebGL library, yet, it is most frequently used together with Three.js: the example [28] of their joint work developed by Lee Stemkoski is given in figure 5.



**Figure 5.** The cube on the Hiro marker (joint application of Three.js and JSARToolKit).

The code size required for implementing functionality of the example makes about 100 rows and is much smaller than in case of JSARToolKit. This is due to applying THREE.js library – extension of Three.js for designing computer games, this fact making its functionality similar to Babylon.js [13].

Its author, Jerome Etienne [12], is the main designer of the first and the second version of AR.js, which, as of January 2021, has versions adapted to both A-Frame and Three.js. In the same way as A-Frame is a superstructure over Three.js, AR is based on JSARToolKit. Thus, this library supports all types of square (marker tracking) and NFT markers (image tracking) as in JSARToolKit, and also enables locating computer models according to their geospatial coordinates (location-based AR) [1].

When applied with A-Frame, the code required for the example similar to that in figure 5 takes several rows:

```

<a-scene embedded arjs>
  <a-marker preset="hiro">
    <a-box></a-box>
  </a-marker>
  <a-entity camera></a-entity>
</a-scene>

```

### 3. Designing a software prototype for career guidance AR-quests

#### 3.1. The concept of a career guidance quest

A quest is a journey aimed at performing a certain mission or achieving a goal. In literature, a quest is most often used to indicate a character's growth – how they change themselves and change others. Propp mentions that such fairy tales (tales of quests) have a general structure composed of about 150 steps ([23], p. 134)].

The Quest Federation of Ukraine defines a quest as an amateur sport and intellectual competition based on consecutive accomplishment of prepared tasks by teams or individuals [3].

During the game, teams solve logical tasks, perform a search in the field, design optimal routes, search for original solutions and prompts. After fulfilling a task, teams proceed to the next one. The team that accomplishes all the tasks fastest wins.

During career guidance events at Kryvyi Rih State Pedagogical University, there are arranged walking quests both in- and out-of-doors. Quest organizers should control time, task performance, check a team's presence at relevant points and task confirmation codes. Quest participants are to solve intellectual and search tasks in the form of a game associated with their future career, this quest type being called career guidance.

#### 3.2. WebAR tools for a career guidance quest

A career-guidance quest is to cover a set route with fixed stations passed by teams. Activated GPS-navigators of quest participants' mobile phones will monitor their passage and help them gather in a certain point of the route marked by an AR object with relevant geospatial coordinates.

The code prototype performing these actions is presented in [1]:

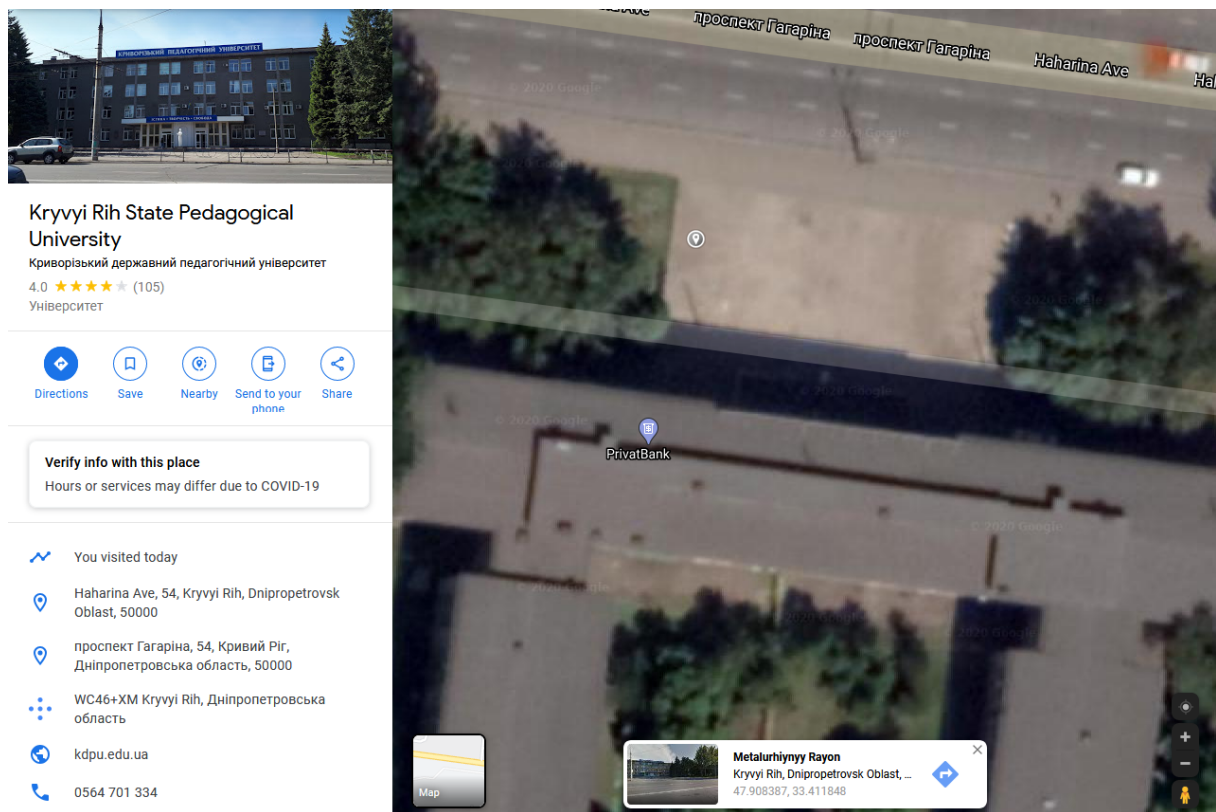
```

<script src="https://aframe.io/releases/1.0.4/aframe.min.js"></script>
<script src="https://unpkg.com/aframe-look-at-component@0.8.0/dist/aframe-look-at-component.min.js"></script>
<script src="https://raw.githubusercontent.com/AR-js-org/AR.js/master/aframe/build/aframe-ar-nft.js"></script>
<a-scene vr-mode-ui="enabled: false" embedded
  arjs="sourceType: webcam; debugUIEnabled: false;">
  <a-text
    value="TEXT WITH MARKERS OF STATION AND NEXT TASK"
    look-at="[gps-camera]"
    scale="120 120 120"
    gps-entity-place="latitude: 47.908387; longitude: 33.411848;">
</a-text>
  <a-camera gps-camera rotation-reader> </a-camera>
</a-scene>

```

Figure 6 contains a place on the square in front of the University main building that corresponds to the coordinates given in the code. Unfortunately, application of GPS location only to reach a station (an intermediate stage of the quest for which points are scored) is not always possible as in multistoried buildings of the University, which make the majority, one and the same coordinate corresponds to stations on different storeys.





**Figure 6.** Finding coordinates of the quest-station by using Google Maps.

To determine the station, it is expedient to combine several types of AR – either the location and the square marker one (aframe-ar.js is the version of AR.js 3 for this combination) or the location and the NFT marker one (aframe-ar-nft.js is the version of AR.js 3 for this combination).

The traditional square marker can be easily determined, but its presence indicates a task for quest participants. To avoid falsification of the results by transferring a marker to another place, auxiliary determination of coordinates is used.

The option of Ar.js 3 to track natural images that can be part of the interior makes quests more complicated, yet, more interesting. To use these images as NFT-markers, one should apply a generator from <https://carnaux.github.io/NFT-Marker-Creator/> and study recommendations on designing good markers [8].

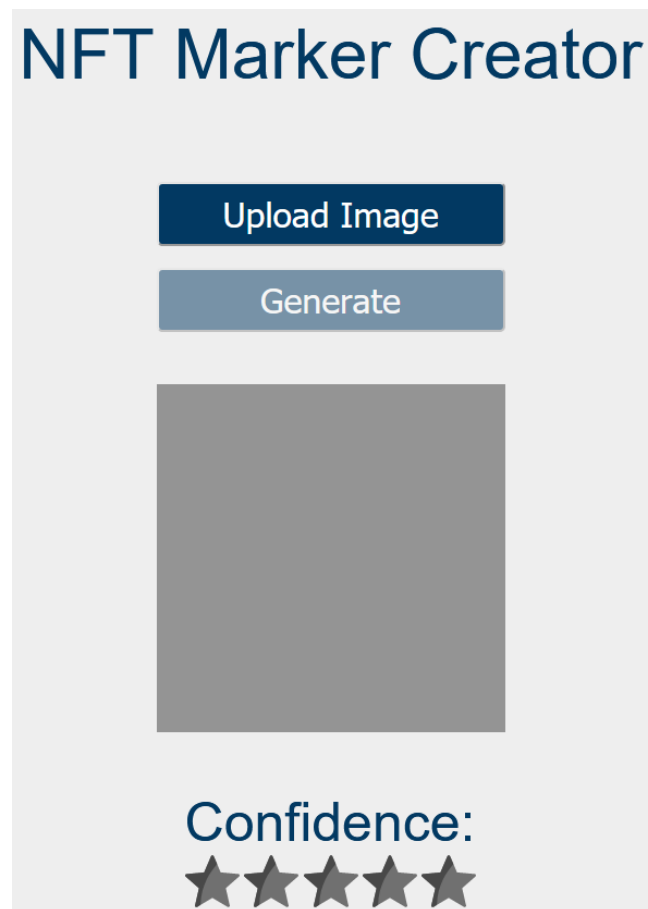
NFT Marker Creator (figure 7) has a confidence meter that indicates whether a marker will be good: the more stars it shows, the better the marker will be recognized.

In case of good lighting and a camera, images with a single star are also recognized well, yet, it is better to select images with four or five stars of confidence. The image should be visually complicated and have a high isolation capability, as, unlike square markers, details are essential here. Visually complicated images provide software with a great number of opportunities to track unique elements of the image that are easily recognized.

The physical size of a marker also affects its recognition. A device should be held closer to smaller images, while it is vice versa with larger ones.

Recognition quality also depends on screen illumination. Besides, cameras with low isolation capability usually function better when they are close to markers.

Printed markers can be located either on nontransparent paper or on a stand. Stands are a natural source of markers as all the University faculties have stands.



**Figure 7.** NFT Marker Creator.

### *3.3. Software development and testing*

Let us consider development a software prototype of a quest element associated with the stand of the Faculty of Physics and Mathematics of Kryvyi Rih State Pedagogical University (figure 8).



**Figure 8.** Department of Computer Science and Applied Mathematics in front of the stand of the Faculty of Physics and Mathematics.

Figure 8 has the highest confidence factor – 5 stars. After printing on nontransparent paper, colour chord of the image changes causing deterioration of the factor to 2 stars (figure 9).



**Figure 9.** Generation of a marker for the original photographic image (a) and the one scanned after printing (b).

Comparison of images in figures 9a and 9b provides an additional conclusion on importance of external lighting to correct balance of the white recognition and the high-quality one of NFT-markers.

After processing the image, the generator creates three files of the marker (in our case - kafera.fset, kafera.fset3 and kafera.iset) that are required for recognition. The last file contains the original image transformed with grey shades (figure 10b).



**Figure 10.** The high-quality image (a) and the internal representation of the image (b).

In the following code fragment, after recognizing the photographic NFT-marker, there appear data on people in the photo (figure 11) as an AR object (the plane with the texture) activated by pressing – there are web links to the Department of Computer Science and Applied Mathematics in a new browser window (figures 12, 13).

Стоять (зліва направо):

1. Мерзлікін Олександр Володимирович - к.пед.н., асистент ;
2. Моїсеєнко Наталя Володимирівна - к. ф. -м.н., доцент;
3. Моїсеєнко Михайло Вікторович - старший викладач ;
4. Шокалюк Світлана Вікторівна- к.пед. н., доцент;
5. Медведєв Дмитро Геннадійович;
6. Степанюк Олександр Миколайович - асистент;
7. Семеріков Сергій Олексійович - д.пед.н., професор;
8. Юрко Олександр Володимирович - старший викладач;
9. Міненко Павло Олександрович - д.ф. -м.н., професор;
10. Мерзлік Павло Володимирович - к.ф.-м.н., доцент ;
11. Соловйов Володимир Миколайович -д.ф.-м.н., професор,завідувач кафедри

Сидять(зліва направо):

12. Мінтій Ірина Сергіївна - к.пед. н.,доцент;
13. Шасмутдінова Наталя Олександрівна - старший лаборант;
14. Закарлюка Ірина Станіславівна - старший викладач;
15. Хараджян Наталя Анатоліївна - к.пед.н.,доцент.

Figure 11. The AR object for the NFT-marker.

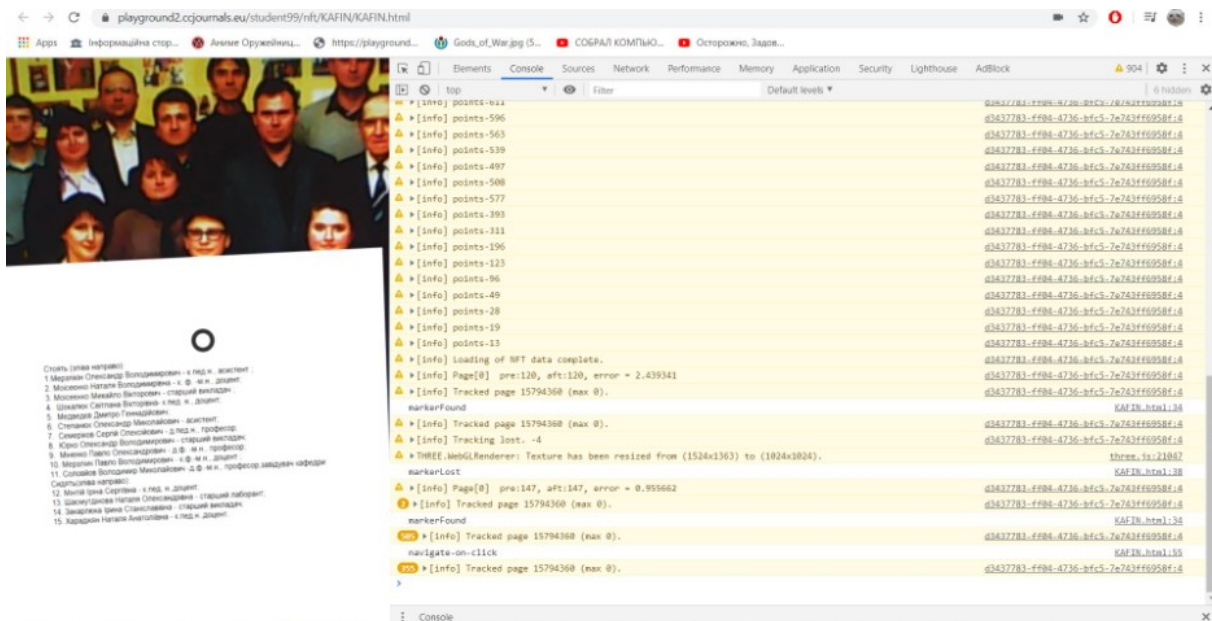


Figure 12. Software testing in the development mode.

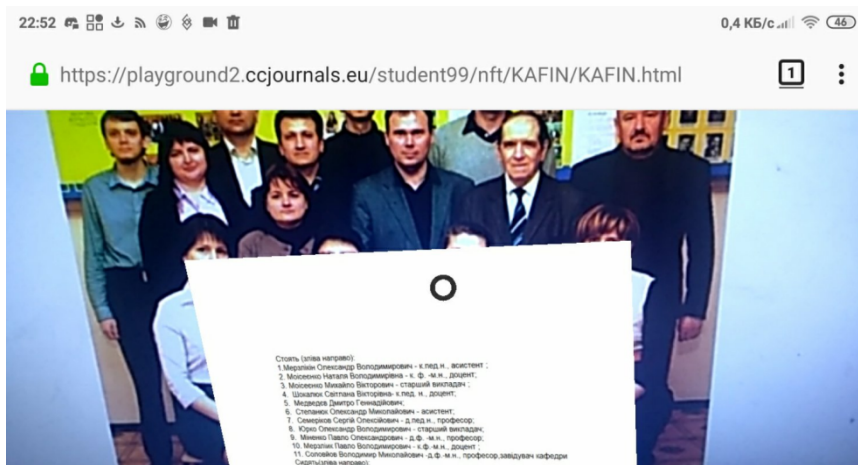


Figure 13. Software testing on a mobile Internet device.

The code snippet of the career guidance quest:

```
<script
src="https://cdn.jsdelivr.net/gh/aframevr/aframe@1c2407b26c61958baa93967b54
12487cd94b290b/dist/aframe-master.min.js"></script>

<script src="https://raw.githubusercontent.com/AR-js-
org/AR.js/master/aframe/build/aframe-ar-nft.js"></script>

<style>
  .arjs-loader {
    height: 100%;
    width: 100%;
    position: absolute;
    top: 0;
    left: 0;
    background-color: rgba(0, 0, 0, 0.8);
    z-index: 9999;
    display: flex;
    justify-content: center;
    align-items: center;
  }

  .arjs-loader div {
    text-align: center;
    font-size: 1.25em;
    color: white;
  }
</style>

<body style="margin : 0px; overflow: hidden;">

<script>
  AFRAME.registerComponent('registerevents', {
    init: function () {
      var marker = this.el;

      marker.addEventListener('markerFound', function() {
        console.log('markerFound', marker.id);
      });

      marker.addEventListener('markerLost', function() {
        console.log('markerLost', marker.id);
      });
    }
  });

  AFRAME.registerComponent('navigate-on-click', {
    schema: {
      url: {default: ''}
    },
    init: function () {
      var data = this.data;
      var el = this.el;
```

```

        el.addEventListener('click', function () {
            console.log('navigate-on-click');
            window.open(data.url, '_blank' );
        });
    }
});

AFRAME.registerComponent('jump', {
  init: function () {
    var obj = this.el;
    obj.addEventListener('click', function () {
      this.setAttribute('position', 150+100*(Math.random()-0.5)+ " 150 0");
      console.log('jump');
    });
  }
});
</script>

<!-- minimal loader shown until image descriptors are loaded -->
<div class="arjs-loader">
  <div>Loading, please wait...</div>
</div>
<a-scene vr-mode-ui="enabled: false;"
  renderer="logarithmicDepthBuffer: true;" embedded
  arjs="trackingMethod: best; sourceType: webcam; debugUIEnabled: false;">
  <!-- url should contain 3 files: .fset, .fset3 and .iset -->

  <a-nft registerevents
    type="nft"
    url="https://playground2.ccjournals.eu/student99/nft/KAFIN/kafera"
    smooth="true"
    smoothCount="10"
    smoothTolerance=".01"
    smoothThreshold="5">

  <a-entity geometry="primitive: box; width: 150; height: 150; depth: 1"
    material="src: url(kaferain.png)" position= "150 -100 0"
    scale = "4 4 2" rotation="-90 0 0"
    navigate-on-click="url:https://kdpu.edu.ua/informatyky-ta-prykladnoi-
matematyky/zahalna-informatsiia/vykladachi-kafedry"></a-entity>

</a-nft>

  <a-entity camera><a-cursor></a-cursor></a-entity>
</a-scene>
</body>

```

Figure 14 reveals examples of joint application of different marker types: there are matrix markers (QR-codes) on the left to access corresponding WebAR programmes and NFT-markers on the right to process corresponding WebAR programmes.



Figure 14. Joint application of different markers in WebAR.

#### 4. Conclusions

1. The review of WebAR software design enables recommendations for beginners to master the following combinations of visualization tools of computer models and tools for tracking real-life objects:

– A-Frame and AR.js is API for fast prototyping most software of which is a HTML-like code. A-Frame is applied to designing scenes, objects, animation and other 3D elements in the web browser. AR.js is aimed at tracking markers and enables a scene designed by A-Frame to be rendered directly on the marker;

– Three.js and ARToolKit.js are used for a deeper level to design software using JavaScript. Three.js uses WebGL to create high-quality 3D scenes in the web browser. JSARToolKit is used for marker tracking and low-level access to data from a camera of a device.

2. Technological conditions of teaching software development with WebAR involve an arbitrary operating system, Internet access, using of cloud storages to deposit materials and availability of at least two classes of devices to test programmes. To master the basics of WebAR software development, it is reasonable to combine A-Frame and AR.js, the scene testing in VR proceeding its AR design. With the scene complexity increasing, JavaScript application to designing A-Frame objects and processing events is accentuated. Designing multi-marker scenes requires different types of matrix markers provided by JSARToolKit, while joint functioning of models attached to different markers requires application of Three.js library. This reveals a dialectic character of teaching WebAR software development: new methods are introduced to overcome problems unsolved by available ones.

3. A quest is an amateur sport-intellectual competition based on accomplishing prepared tasks by teams or individual players. There are some walking quests, both in- and out-of-doors, organized by Kryvyi Rih State Pedagogical University as part of career guidance activity. Quest organizers should control time, task performance, check a team's presence at relevant points and task confirmation codes. Quest participants are to solve intellectual and search tasks in the form of a game associated with their future career, this quest type being called career guidance.

4. A career guidance quest should follow a set route with required fixation of stations passed by teams. Activated GPS-navigators in quest participants' mobile devices help their monitoring and make them gather in a certain route point marked by an AR object attached to certain geographic coordinates. To locate a station, it is expedient to combine several types of AR – the location and square-marker or the location and NFT-marker one. To complicate and facilitate quests is possible by using the option of Ar.js 3 to track natural images as parts of the interior by means of NFT-markers. The images should be visually complex and possess high isolation capacity. Recognition quality of NFT-markers also depends on their physical size, screen illumination, distance between a device and a marker, isolation capacity of a camera, external lighting, screen brightness and paper quality for a printed marker.

5. Designed software prototypes of the career guidance quest enable determining a participant's quest coordinates as AR objects attached as well as any other images of photographic quality in the career guidance quest space.

## References

- [1] AR.js organization 2020 AR-js-org/AR.js: Image tracking, Location Based AR, Marker tracking. All on the Web *GitHub* URL <https://github.com/AR-js-org/AR.js>
- [2] artoolkitX 2020 artoolkitx/jsartoolkit5: Javascript ARToolKit v5.x *GitHub* URL <https://github.com/artoolkitx/jsartoolkit5>
- [3] AutoQuest Group 2017 *Chto takoe KVEST? (What is a QUEST?)* URL [https://web.archive.org/web/20081013022413/http://www.kbect.net.ua/gamma\\_igr/chto\\_tak\\_oe\\_kvest](https://web.archive.org/web/20081013022413/http://www.kbect.net.ua/gamma_igr/chto_tak_oe_kvest)
- [4] Babylon.js 2019 *Babylon AR Preview CDN* URL <https://ar.babylonjs.com/>
- [5] Babylon.js 2021 *Babylon.js: Powerful, Beautiful, Simple, Open - Web-Based 3D At Its Best* URL <https://www.babylonjs.com/>
- [6] Cabello R 2021 *three.js – JavaScript 3D library* URL <https://threejs.org/>
- [7] Crockford D 2011 Crockford on JavaScript *YouTube* URL <https://youtu.be/playlist?list=PLEzQf147-uEpvTa1bHDNlxUL2kIHUMHJu>
- [8] de Oliveira D F G 2020 Creating good markers *Carnaux/NFT-Marker-Creator Wiki* URL <https://github.com/Carnaux/NFT-Marker-Creator/wiki/Creating-good-markers>



- [9] Derzhavna sluzhba statystyky Ukrainy 2019 *Rozpodil postiinoho naseleattia Ukrainy za stattiu ta vikom na 1 sichnia 2019 roku: statystychnyi zbirnyk (Distribution of the permanent population of Ukraine by sex and age as of January 1, 2019: statistical collection)* URL [http://database.ukrcensus.gov.ua/PXWEB2007/ukr/publ\\_new1/2020/zb\\_chuselnist%202019.pdf](http://database.ukrcensus.gov.ua/PXWEB2007/ukr/publ_new1/2020/zb_chuselnist%202019.pdf)
- [10] Derzhstat Ukrainy 2020 *Zaklady vyshchoi osvity (Institutions of higher education)* URL [http://www.ukrstat.gov.ua/operativ/operativ2005/osv\\_rik/osv\\_u/vuz\\_u.html](http://www.ukrstat.gov.ua/operativ/operativ2005/osv_rik/osv_u/vuz_u.html)
- [11] Elagina D 2020 Ukraine: popularity of app categories by age 2018 *Statista* URL <https://www.statista.com/statistics/1023304/ukraine-popularity-app-categories/>
- [12] Etienne J 2017 jeromeetienne/threex: Game Extensions for three.js *GitHub* <https://github.com/jeromeetienne/threex>
- [13] Etienne J 2017 *THREEx Game Extensions for Three.js* URL <https://www.threejsgames.com/extensions/>
- [14] Google 2020 *Glass* URL <https://www.google.com/glass/start/>
- [15] Microsoft 2021 Mixed Reality Technology for Business *Microsoft HoloLens* URL <https://www.microsoft.com/en-us/hololens>
- [16] Mintii I S and Soloviev V N 2018 Augmented Reality: Ukrainian Present Business and Future Education *CEUR Workshop Proceedings* **2257** 227–31
- [17] Mozilla and individual contributors 2021 WebGL: 2D and 3D graphics for the web – Web APIs *MDN* URL [https://developer.mozilla.org/en-US/docs/Web/API/WebGL\\_API](https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API)
- [18] Mozilla and individual contributors 2021 WebRTC API – Web APIs *MDN* URL [https://developer.mozilla.org/en-US/docs/Web/API/WebRTC\\_API](https://developer.mozilla.org/en-US/docs/Web/API/WebRTC_API)
- [19] Mozilla and individual contributors 2021 WebXR Device API – Web APIs *MDN* URL [https://developer.mozilla.org/en-US/docs/Web/API/WebXR\\_Device\\_API](https://developer.mozilla.org/en-US/docs/Web/API/WebXR_Device_API)
- [20] Murray J and Babylon.js 2019 Babylon AR *Medium* URL <https://medium.com/@babylonjs/babylon-ar-7823ab4a80c1>
- [21] Opera Software AS 2019 webkit *Dev.Opera* URL <https://dev.opera.com/tags/webkit/>
- [22] OSF HealthCare 2019 *Applications | Jump Simulation* URL [http://web.archive.org/web/20191130085337mp\\_/https://jumpsimulation.org/education/applications](http://web.archive.org/web/20191130085337mp_/https://jumpsimulation.org/education/applications)
- [23] Propp V 1968 *Morphology of the Folktale* (Austin: University of Texas Press) URL [https://monoskop.org/images/f/f3/Propp\\_Vladimir\\_Morphology\\_of\\_the\\_Folktale\\_2nd\\_ed.pdf](https://monoskop.org/images/f/f3/Propp_Vladimir_Morphology_of_the_Folktale_2nd_ed.pdf)
- [24] StatCounter 2021 Mobile & Tablet Browser Market Share Ukraine *StatCounter Global Stats* URL <https://gs.statcounter.com/browser-market-share/mobile-tablet/ukraine/>
- [25] StatCounter 2021 Mobile Android Version Market Share Ukraine *StatCounter Global Stats* URL <https://gs.statcounter.com/android-version-market-share/mobile/ukraine>
- [26] StatCounter 2021 Mobile Operating System Market Share Ukraine *StatCounter Global Stats* URL <https://gs.statcounter.com/os-market-share/mobile/ukraine>
- [27] Statista 2021 Mobile Games - Ukraine *Statista Market Forecast* URL <https://www.statista.com/outlook/211/338/mobile-games/ukraine>
- [28] Stemkoski L 2018 *Basic Cube* URL <https://stemkoski.github.io/AR-Examples/hello-cube.html>
- [29] Supermedium 2020 *A-Frame – Make WebVR* URL <https://aframe.io/>
- [30] Syrovatskyi O V, Semerikov S O, Modlo Ye O, Yechkalo Yu V and Zelinska S O 2018 Augmented reality software design for educational purposes *CEUR Workshop Proceedings* **2292** 193–225
- [31] The Khronos Group Inc. 2021 *WebGL Overview* URL <https://www.khronos.org/webgl/>
- [32] Tutorialspoint 2020 WebGL - Sample Application *Tutorialspoint* URL [https://www.tutorialspoint.com/webgl/webgl\\_sample\\_application.htm](https://www.tutorialspoint.com/webgl/webgl_sample_application.htm)
- [33] WebAssembly 2020 *WebAssembly* URL <https://webassembly.org/>