

Γραφικά Υπολογιστών

Ιόνιο Πανεπιστήμιο
Τμήμα Πληροφορικής

Στέργιος Παλαμάς, Επίκουρος Καθηγητής

Μάθημα 4:

Web AR, Web VR

Η επαυξημένη πραγματικότητα (Augmented Reality – AR), γνωρίζει σήμερα άνθηση χάρη στην ευρεία εφαρμογή της μέσω φορητών συσκευών όπως τα Smartphones και τα Tablets με χρήση της ενσωματωμένης τους κάμερας.

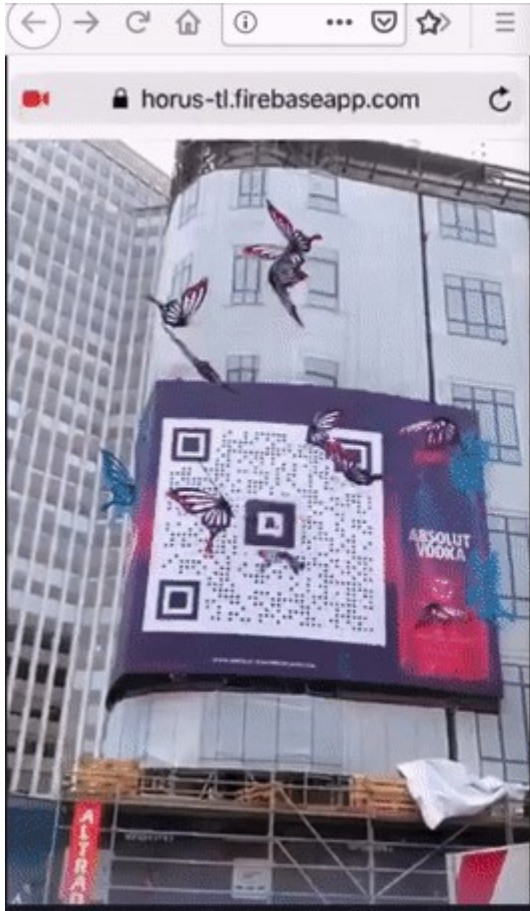
Υπάρχουν διάφοροι τρόποι για την ανάπτυξη εφαρμογών AR, όπως:

- SDKs για ανάπτυξη σε εργαλεία που παράγουν native εφαρμογές
- SDKs για μηχανές γραφικών (πχ Unity)
- Εργαλεία «υψηλού επιπέδου» ([πχ indoAR](#))

Μια πρόσφατη λύση ανάπτυξης εφαρμογών AR βασίζεται σε τεχνολογίες Web, και συγκεκριμένα στη γλώσσα Javascript και τη βιβλιοθήκη AR.js που επιτρέπει την υλοποίηση απλών εφαρμογών AR που εκτελούνται εντός του browser της συσκευής και χρησιμοποιούν την κάμερά της και τους αισθητήρες (πχ γυροσκόπιο).

<https://ar-js-org.github.io/AR.js-Docs/>





AR types

AR.js features the following types of Augmented Reality, on the Web:

Image Tracking, when a 2D images is found by the camera, it's possible to show some kind of content on top of it, or near it. The content can be a 2D image, a GIF, a 3D model (also animated) and a 2D video too. Cases of use: Augmented Art, learning (Augmented books), Augmented flyers, advertising, etc.

Location Based AR, this kind of AR uses real-world places in order to show Augmented Reality content, on the user device. The experiences that can be built with this library are those that use a user's position in the real world. The user can move (ideally outdoor) and through their smartphones they can see AR content where places are in the real world. Moving around and rotating the phone will make the AR content change according to users position and rotation (so places are 'anchored' in their real position, and appear bigger/smaller according to their distance from the user). With this solution it's possible to build experiences like interactive support for tourist guides, assistance when exploring a new city, find places of interest like buildings, museums, restaurants, hotels and so on. It's also possible to build learning experiences like treasure hunts, and biology or history learning games, or use this technology for situated art (visual art experiences bound to specific real world coordinates).

Marker Tracking, When a marker is found by the camera, it's possible to show some content (same as Image Tracking). Markers are very stable but limited in shape, color and size. It is suggested for those experiences where are required a lot of different markers with different content. Examples of use: (Augmented books), Augmented flyers, advertising.

Η Βιβλιοθήκη AR.js βασίζεται σε δύο άλλες βιβλιοθήκες για την απεικόνιση 3D γραφικών σε Browsers: την threejs και την A-Frame



<https://threejs.org/>

<https://aframe.io/>

AR.js with Image Tracking + Location Based AR

Import AFRAME version:

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

Import three.js version:

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

AR.js with Marker Tracking + Location Based AR:

Import AFRAME version:

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

Import three.js version:

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

Σε μια κενή HTML σελίδα εισάγουμε τις απαραίτητες βιβλιοθήκες Javascript

```
<html>
<head>
<script src="https://aframe.io/releases/0.9.2/aframe.min.js"></script>
<script src="https://cdn.rawgit.com/jeromeetienne/AR.js/1.7.7/aframe/build/aframe-ar.js"></script>
<script src="https://raw.githack.com/donmccurdy/aframe-extras/master/dist/aframe-extras.loaders.min.js"></script>
</head>

<body>

</body>
</html>
```


Στο <body> της HTML σελίδας εισάγουμε μια σκηνή με το tag <a-scene> η οποία είναι απαραίτητη για να τοποθετήσουμε σε αυτή το περιεχόμενο που θέλουμε να δείξουμε

```
<a-scene embedded arjs='sourceType: webcam; debugUIEnabled: false;  
detectionMode: mono_and_matrix; matrixCodeType: 3x3;'>
```

```
</a-scene>
```

Στο <a-scene> της HTML σελίδας εισάγουμε ένα video προκειμένου να το χρησιμοποιήσουμε αργότερα για να το δείξουμε. Υλικό που είναι διαθέσιμο για τη σκηνή το ονομάζουμε Asset.

```
<a-assets> <video crossorigin="anonymous" id="videoID" autoplay  
loop="true" type="video/mp4" preload="auto" src="url to the  
video"> </a-assets>
```

Αντικαθιστούμε με τη διεύθυνση του αρχείου video

Προσθέτουμε στη συνέχεια (μέσα στο <a-scene>) Markers – εικονίδια τα οποία όταν ανιχνευτούν από την κάμερα θα ενεργοποιούν την εμφάνιση περιεχομένου σε AR

```
<a-marker type="pattern" preset="custom" url="url to the marker"> </a-marker>
```



Διεύθυνση του αρχείου που περιέχει το Market

[Εργαλείο δημιουργίας Markers με custom pattern](#)



Μπορούμε να ορίσουμε 3D μοντέλα – βασικά σχήματα τα οποία θα εμφανίζουμε στην οθόνη. Ο παρακάτω κώδικας ορίζει έναν κύβο και μέσα στον ημιδιαφανή κύβο έναν «κόμπο/περιέλιξη» (torus-knot).

Βάζουμε τα παρακάτω ΜΕΣΑ στο <a-marker> προκειμένου να εμφανίζονται με την ανίχνευση του συγκεκριμένου marker

Κύβος ημιδιάφανος

```
<a-marker type="pattern" preset="custom" url=  
"https://raw.githubusercontent.com/arjanvanbekkum/arjanvanbekkum.github.io/master/augmentedreality/epicshit.patt" >  
  <a-box position="0 0.5 0" material="opacity: 0.5; side: double" geometry="">  
    <a-torus-knot radius="0.26" radius-tubular="0.05" animation="property: rotation; to: 0 360 0; loop: true; dur: 5000" ></a-torus-knot>  
  </a-box>  
</a-marker>
```

Περίπλεξη (torus)

Μπορούμε να ορίσουμε marker που να ενεργοποιεί την εμφάνιση video (αυτού που είχαμε ορίσει με το <asset> προηγουμένως.

```
<a-marker vidhandler preset="custom" type="pattern" url="url to the pattern">  
  <a-video src="#videoID" width="1.4375" height="1" rotation="270 0 0">  
  </a-video>  
</a-marker>
```



Το ID που είχαμε δώσει στο Video όταν το ορίσαμε στο <asset>

Βάζοντας τον παρακάτω κώδικα στο <HEAD> της HTML σελίδας, μέσα σε ένα <SCRIPT> TAG , ορίζουμε έναν handler για την αρχικοποίηση του video , την έναρξη αναπαραγωγής και την παύση όταν ανιχνεύεται ή όχι το marker.

```
AFRAME.registerComponent('vidhandler', {
  // ...
  init: function () {
    this.toggle = false;
    this.vid = document.querySelector("#videoID")
    this.vid.pause();
  },
  tick: function () {
    if (this.el.object3D.visible == true) {
      if (!this.toggle) {
        this.toggle = true;
        this.vid.play();
      }
    } else {
      this.toggle = false;
      this.vid.pause();
    }
  }
});
```

Μπορούμε να ορίσουμε marker που να ενεργοποιεί ένα 3d μοντέλο με κίνηση.

Η διεύθυνση από όπου θα ανακτηθεί το μοντέλο

```
<a-marker id="memarker" preset="hiro">  
  <a-gltf-model src="url to the gltf model" position="0 0 0" rotation="0 0 0"  
  scale="0.03 0.03 0.03" animation-mixer="clip: Take 001; loop:repeat">  
</a-gltf-model> </a-marker>
```

... ή ένα στατικό μοντέλο

Custom marker που βρίσκεται στο ίδιο folder με την html σελίδα

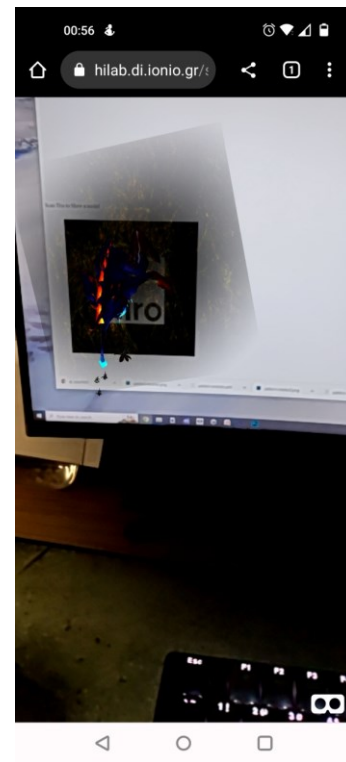
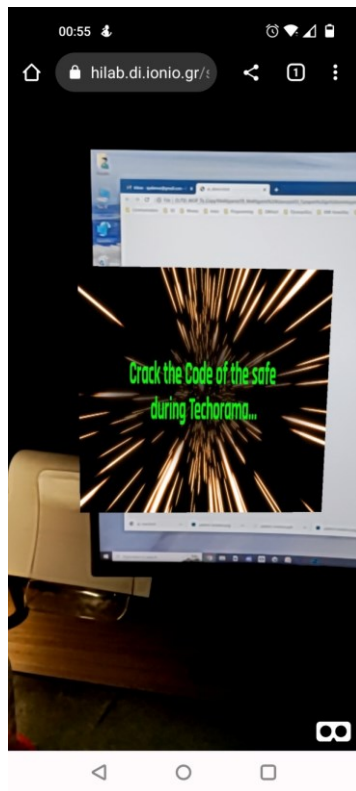
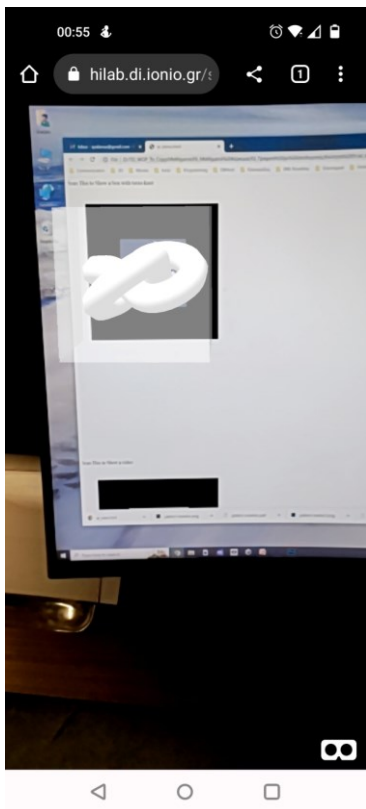
```
<a-marker type="pattern" preset="custom" url="test_marker.patt" >  
  <a-gltf-model src="statue.glb" position="0 0 0" rotation="0 -140 40" scale="1.5 1.5 1.5"> </a-gltf-model>  
</a-marker>
```

Μπορούμε να εμφανίζουμε το μοντέλο με κάποια περιστροφή και κλίμακα

3d μοντέλο (GLB format) που βρίσκεται στο ίδιο folder με την html σελίδα

Τέλος πρέπει να βάλουμε και μια camera μέσα στο <a-scene>

```
<a-entity camera=""></a-entity>
```

Ανοίξτε σε υπολογιστή τη Σελίδα με τα Markers: <http://spalamas.eu/WebAR/markers.html>

Ανοίξτε στο κινητό τη σελίδα με τη WebAR εφαρμογή
(προσοχή χρειάζεται https για να επιτρέψει ο browser να ανοίξει η κάμερα): <https://spalamas.eu/WebAR>

A-FRAME

Blog

A-Frame 1.1.0 - AR, Quest 2 ...

Examples

Hello WebVR

Model Viewer

Hand Tracking

Responsive UI

360° Image

360 Video

Anime UI

BeatSaver Viewer

Moon Rider

Gunters of OASIS 🐱

Supercraft 🐱

Super Says 🐱

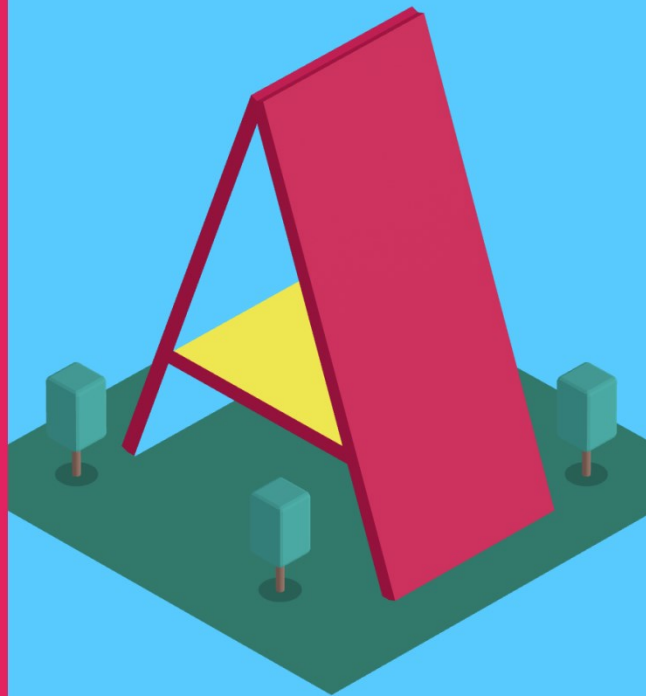
Towermax Fitness 🐱

A-Blast 🐱

A-Painter 🐱

A Saturday Night 🐱

Subscribe to get updates on A-Frame and our VR projects!



[DOCS](#) [FAQ](#) [BLOG](#) [COMMUNITY](#) [SHOWCASE](#)

A web framework for building 3D/AR/VR experiences

Make 3D worlds with HTML and Entity-Component
For Quest, Rift, WMR, SteamVR, mobile, desktop

GET STARTED

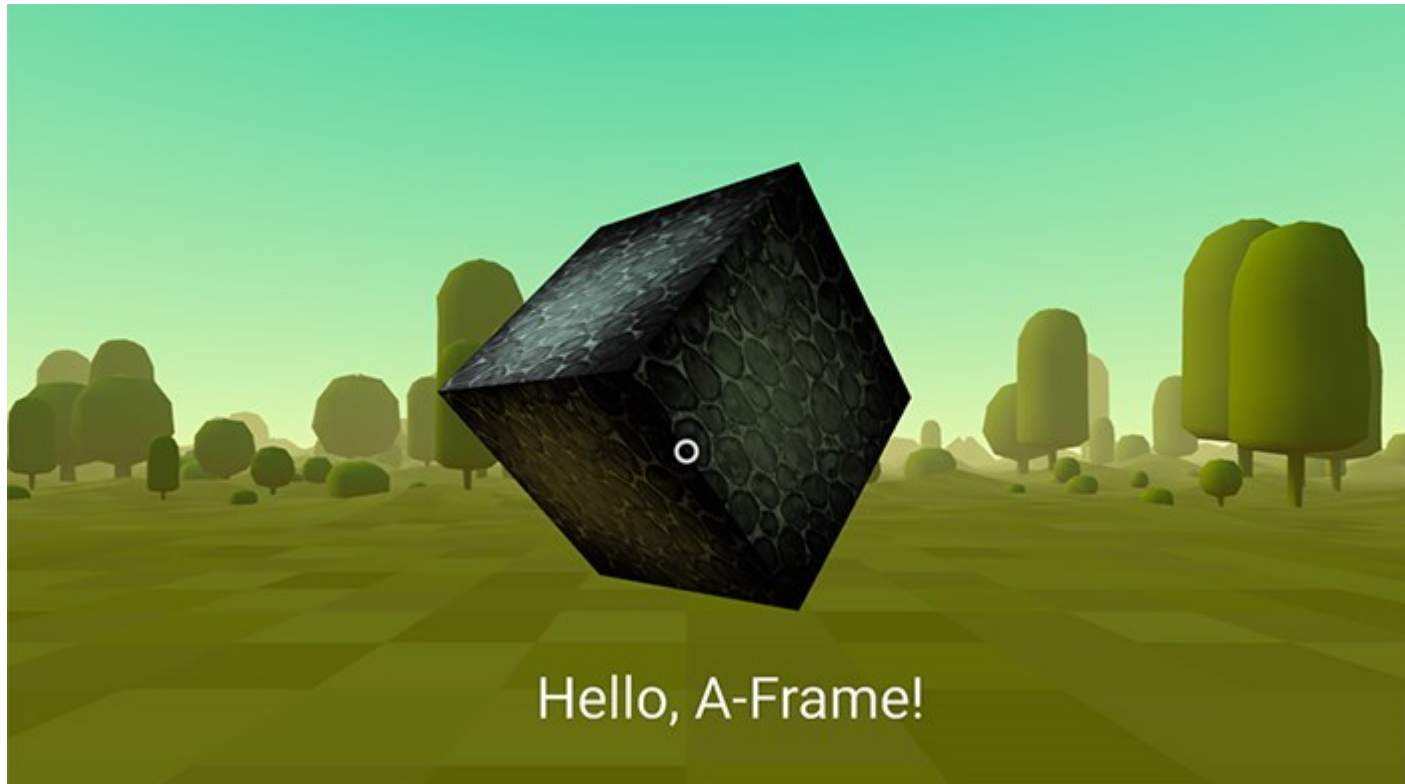
Special Sponsors



SUPERMEDIUM

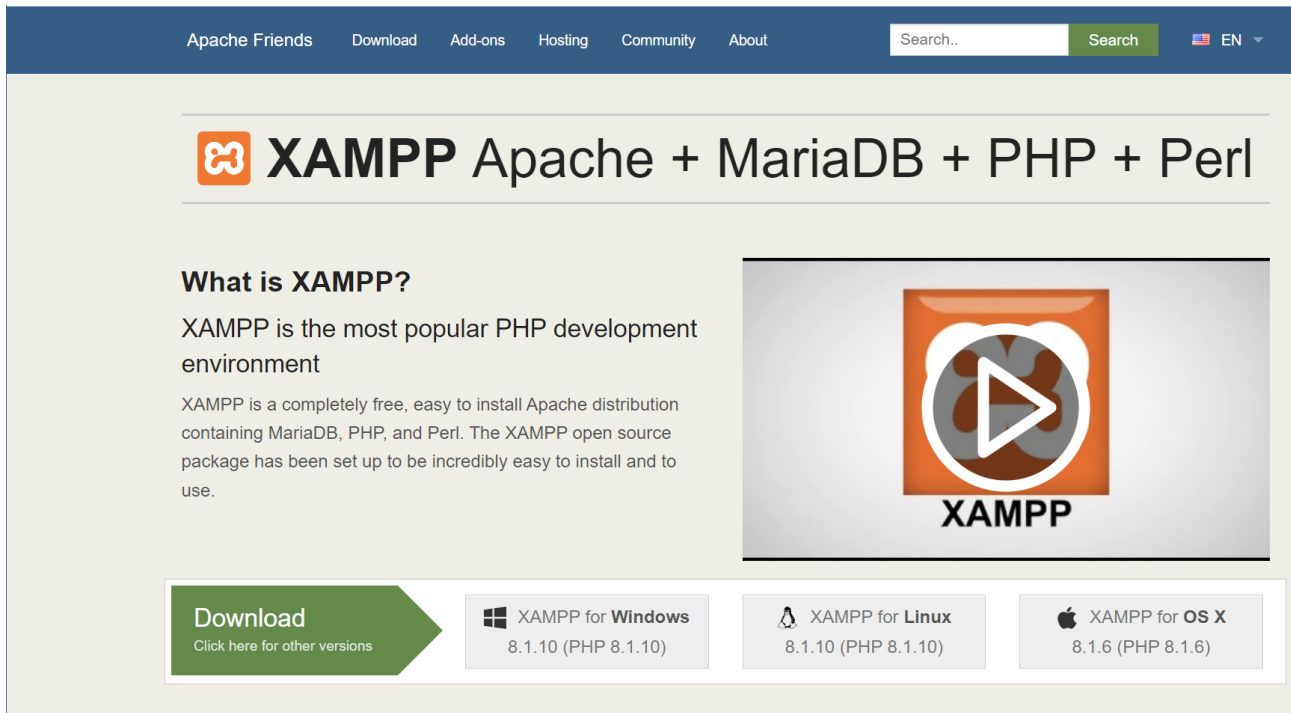
<https://aframe.io/>

Building a Basic Scene



Όταν δουλεύουμε με το A-Frame τα αρχεία πρέπει να τα φιλοξενούμε σε Web Server, ο οποίος μπορεί να τρέχει και στο δικό μας υπολογιστή.

Ένας απλός τρόπος για να εγκαταστήσουμε τον Apache Web Server στο PC μας είναι εγκαθιστώντας το XAMPP που επιπλέον εγκαθιστά και τη Βάση Δεδομένων MariaDB (MySQL) και τη γλώσσα προγραμματισμού PHP.



The screenshot shows the XAMPP website homepage. At the top, there is a navigation bar with links for Apache Friends, Download, Add-ons, Hosting, Community, and About. A search bar and a language selector (EN) are also present. The main heading is "XAMPP Apache + MariaDB + PHP + Perl". Below this, there is a section titled "What is XAMPP?" which explains that XAMPP is a popular PHP development environment, free and easy to install, containing MariaDB, PHP, and Perl. To the right of this text is a video player showing the XAMPP logo. At the bottom, there are three download buttons: "Download" (with a link to other versions), "XAMPP for Windows 8.1.10 (PHP 8.1.10)", "XAMPP for Linux 8.1.10 (PHP 8.1.10)", and "XAMPP for OS X 8.1.6 (PHP 8.1.6)".

<https://www.apachefriends.org/>

Starting with HTML

```
<html>
  <head>
    <script src="https://aframe.io/releases/1.3.0/aframe.min.js"></script>
  </head>

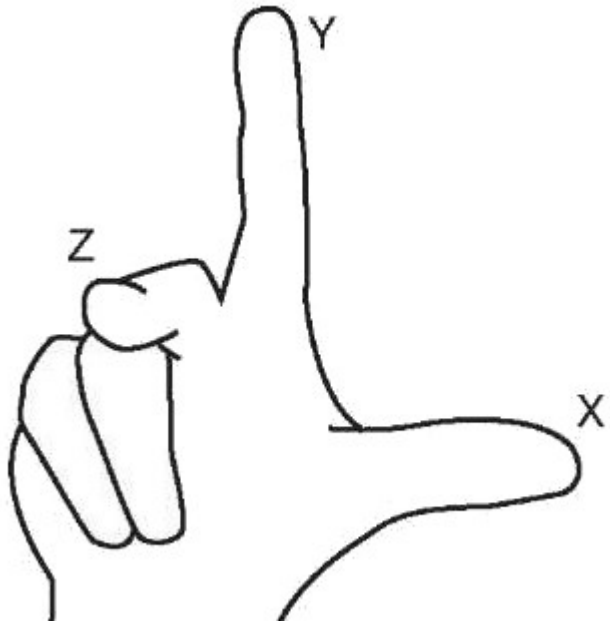
  <body>
    <a-scene>
    </a-scene>
  </body>
</html>
```

Ξεκινάμε με μια άδεια σκηνή γράφοντας τον παραπάνω κώδικα σε ένα HTML αρχείο

Προσθέτουμε μια οντότητα (entity) στη σκηνή

```
<a-scene>  
  <a-box color="red"></a-box>  
</a-scene>
```

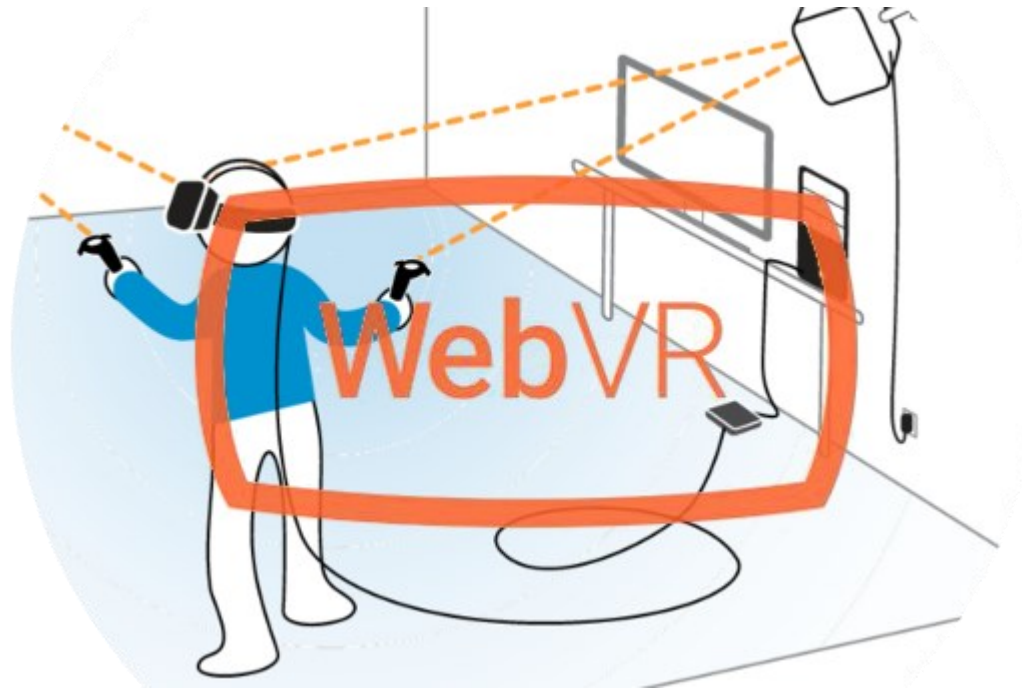
Προσθέτουμε στη σκηνή έναν κόκκινο κύβο



Σύστημα συντεταγμένων

```
<a-scene>  
  <a-box color="red" position="0 2 -5" rotation="0 45 45" scale="2 2 2"></a-box>  
</a-scene>
```

Μετακινούμε τον κύβο στη θέση 0.2.-5, τον περιστρέφουμε κατά 45 μοίρες στους άξονες Y και Z και τον μεγεθύνουμε κατά 2 σε όλους τους άξονες



Στον υπολογιστή μπορούμε να κινούμαστε με τα πλήκτρα WSAD και να στρέφουμε το βλήμα με click και drag του Mouse.

Στο κινητό μπορούμε να περιστρέφουμε το κινητό

Προσθέτοντας ένα περιβάλλον

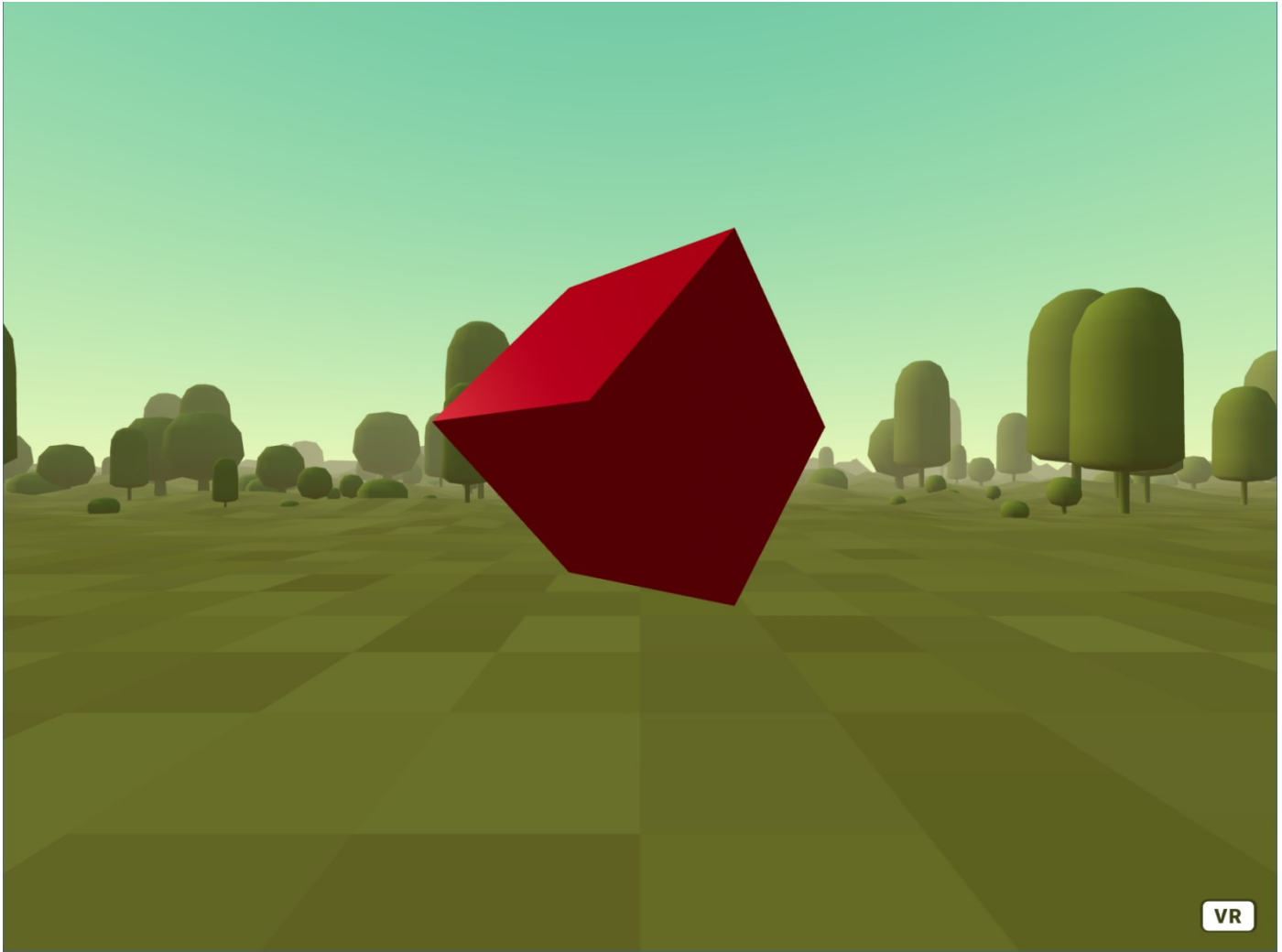
Πρώτα προσθέτουμε τη βιβλιοθήκη για τα περιβάλλοντα

```
<head>
  <script src="https://aframe.io/releases/1.3.0/aframe.min.js"></script>
  <script src="https://unpkg.com/aframe-environment-component/dist/aframe-environment-component.min.js"></script>
</head>
```

Μπορούμε μετά να προσθέσουμε ένα από τα έτοιμα περιβάλλοντα

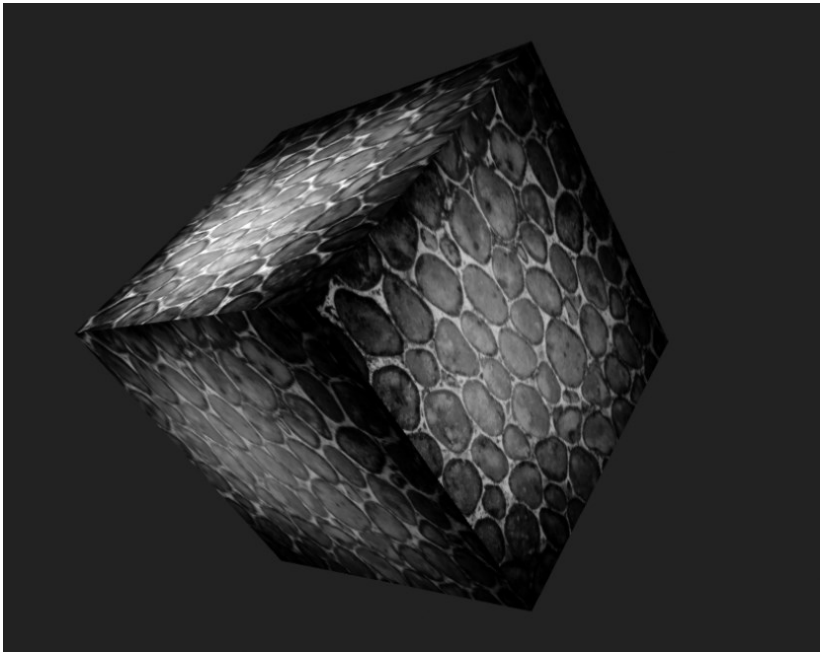
```
<a-scene>
  <a-box color="red" position="0 2 -5" rotation="0 45 45" scale="2 2 2">
  </a-box>

  <!-- Out of the box environment! -->
  <a-entity environment="preset: forest; dressingAmount: 500"></a-entity>
</a-scene>
```



Προσθέτοντας μια υφή (texture)

```
<a-scene>  
  <a-box src="https://i.imgur.com/mYmbrp.jpg" position="0 2 -5" rotation="0 45 45"  
  scale="2 2 2">  
  </a-box>  
  
  <a-sky color="#222"></a-sky>  
</a-scene>
```



Αρχείο εικόνας με την υφή (texture)

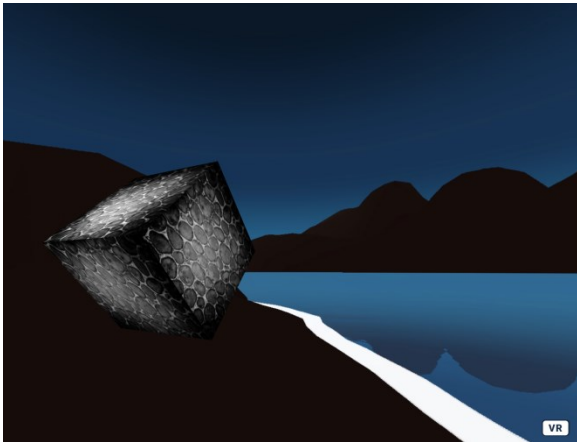
Εδώ χρησιμοποιούμε μια εικόνα που φιλοξενείται σε έναν web server στο Internet. Αν η εικόνα βρίσκεται στον δικό μας server προσαρμόζουμε τη διεύθυνση κατάλληλα.

Ορίζοντας μια εικόνα σαν φόντο

```
<a-scene>
  <a-assets>
    
    
  </a-assets>

  <a-box src="#boxTexture" position="0 2 -5" rotation="0 45 45" scale="2 2 2">
  </a-box>

  <a-sky src="#skyTexture">
  </a-sky>
</a-scene>
```



Εδώ χρησιμοποιούμε τα <a-assets> για να ορίσουμε αρχεία που μετά μπορούμε να τα χρησιμοποιούμε μέσα στη σκηνή πχ σαν textures ή background

Προσθέτοντας ένα επίπεδο σαν έδαφος

Προσθέτουμε ένα texture για το έδαφος

```
<a-assets>  
<!-- ... -->  
  
<!-- ... -->  
</a-assets>
```

```
<a-plane src="#groundTexture" rotation="-90 0 0" width="30" height="30" repeat="10 10">  
</a-plane>
```

Περιστροφή για να έχει σωστό προσανατολισμό

Διαστάσεις

Επαναλήψεις του texture
στις δύο διαστάσεις

Προσθέτοντας φωτισμό

Περιβάλλον φωτισμός

```
<a-light type="ambient" color="#445451"></a-light>  
<a-light type="point" intensity="2" position="2 4 4"></a-light>
```

Σημειακός φωτισμός

Προσθέτοντας κίνηση στον κύβο

```
<a-box src="#boxTexture" position="0 2 -5" rotation="0 45 45" scale="2 2 2"  
animation="property: object3D.position.y; to: 2.2; dir: alternate; dur: 2000; loop:  
true">  
</a-box>
```

Animate the entity's object3D's position's Y axis.

Animate to **2.2** which is 20 centimeters higher.

Alternate the *dir* (direction) of the animation on each repeated cycle of the animation so it goes back and forth.

Last for 2000 millisecond *dur* (duration) on each cycle.


Loop or repeat the animation forever.

Προσθέτοντας αλληλεπιδράσεις

Προσθέτουμε στη σκηνή μια κάμερα με έναν cursor

```
<a-camera>  
  <a-cursor></a-cursor>  
</a-camera>
```

```
<script>  
  var boxEl = document.querySelector('a-box');  
  
  boxEl.addEventListener('mouseenter', function () {  
    boxEl.setAttribute('scale', {x: 3, y: 3, z: 3});  
  });  
  
  boxEl.addEventListener('mouseleave', function () {  
    boxEl.setAttribute('scale', {x: 1, y: 1, z: 1});  
  });  
</script>
```



Προσθέτουμε στον κύβο έναν EventListener που ελέγχει πότε ο cursor μπαίνει στο εσωτερικό του κύβου και όταν αυτό συμβεί τριπλασιάζει το μέγεθός του . Επίσης προσθέτουμε EventListener που όταν ο cursor βγει από τον κύβο τον επαναφέρει στο αρχικό του μέγεθος

Το παραπάνω <script> πρέπει να το βάλουμε ανάμεσα στο </a-scene> και το </body>

Προσθέτοντας Ήχο

Μπορούμε να προσθέσουμε ήχο βάζοντας το αντίστοιχο TAG <audio>

```
<audio src="https://cdn.aframe.io/basic-guide/audio/backgroundnoise.wav"  
autoplay preload>  
</audio>
```

Αυτόματη αναπαραγωγή ήχου

Διεύθυνση αρχείου ήχου

Πρέπει να δηλώσουμε στο Chrome ότι επιτρέπουμε την αναπαραγωγή ήχου από τη σελίδα!!!!

Προσθέτοντας σημειακό (spatial) Ήχο

Μπορούμε να προσθέσουμε ήχο σε συγκεκριμένη θέση βάζοντας το αντίστοιχο TAG `<a-sound>` μέσα στη σκηνή. Η ένταση θα αυξάνεται όταν πλησιάζουμε και θα μειώνεται όταν απομακρυνόμαστε από την ηχητική πηγή.

```
<a-scene>  
<!-- ... -->  
<a-sound src="https://cdn.aframe.io/basic-guide/audio/backgroundnoise.wav"  
autoplay="true" position="-3 1 -4">  
</a-sound>  
<!-- ... -->  
</a-scene>
```

Διεύθυνση αρχείου ήχου

Θέση του ήχου

Πρέπει να δηλώσουμε στο Chrome ότι επιτρέπουμε την αναπαραγωγή ήχου από τη σελίδα!!!!

Προσθέτοντας Κείμενο

```
<a-entity  
text="value: Hello, A-Frame!; color: #BBB"  
position="-0.9 0.2 -3"  
scale="1.5 1.5 1.5">  
</a-entity>
```

Τον κώδικα και οδηγίες εκτέλεσης για τα δύο παραδείγματα (Web VR και Web AR) θα βρείτε στα συνοδευτικά αρχεία με το υλικό