

AR EN JAVASCRIPT : A-FRAME ET AR.JS

Le sujet abordé ici est l'AR (augmented reality) sans appli propriétaire, juste en web application.

Avantage : rien à télécharger ni installer .

1. Création d'un environnement VR : A-frame (framework 3D).....	2
1.1 Exemple commenté :.....	2
1.2 Scène VR.....	4
2. Création d'un environnement AR.....	4
2.1 Exemple de code commenté.....	4
2.2 Résultat : scène AR.....	6
3. Exemple Hello VR.....	6
4. Suivi des mains : hand tracking.....	7
5. Interface utilisateur : Responsive UI.....	7
5.1 Le code :.....	8
6. Visualiser image 360°.....	10
6.1 Le code.....	10
7. Lecteur video 360.....	11
7.1 Le code.....	11
8. Géolocalisation GPS + reconnaissance image.....	12
9. Géolocalisation + reconnaissance par marqueur.....	12
10. Animation.....	12
11. Réalisation interface utilisateur avec A-frame.....	13
12. Utilisation Mind-AR.....	14
13. Bibliographie.....	15

Remarque : une librairie Mind-AR est apparue depuis peu pour encore simplifier le développement (voir bibliographie), associée à A-frame on peut facilement créer une IHM VR.

1. Crédit d'un environnement VR : A-frame (framework 3D)

Source et apprentissage : A-frame school.

A-frame permet de facilement créer un environnement 3D décrit par des balises.

1.1 Exemple commenté :

```
1. <html>
2.   <head>
3.     <script src="https://aframe.io/releases/1.1.0/aframe.min.js"></script>
4.   <style>
5.     .buttons {
6.       position: absolute;
7.       bottom: 0;
8.       left: 0;
9.       width: 100%;
10.      height: 5em;
11.      display: flex;
12.      justify-content: center;
13.      align-items: center;
14.      z-index: 10;
15.    }
16.  </style>
17. </head>
18. <body>
19.   <div class="buttons">
20.     <button>Green</button>
21.   </div>
22. <a-scene>
23.   <a-assets>
24.     <a-asset-item id="plante01-obj" src="assets/3Dplante01/3Dplante01.obj"></a-asset-item>
25.     <a-asset-item id="plante01-mtl" src="assets/3Dplante01/3Dplante01.mtl"></a-asset-item>
26.   </a-assets>
27.   <a-entity position="-1 1 -5" scale="0.1 0.1 0.1" obj-model="obj: #plante01-obj; mtl: #plante01-mtl"></a-entity>
28.   <a-box position="-2 0 -6" rotation="20 45 0" color="#4CC3D9"></a-box>
29.   <a-box position="-1 2.5 -10" rotation="20 0 0" color="#0000FF"></a-box>
30.
31. </a-scene>
```

AR en javascript : A-Frame et AR.js -

```
32. </body>
33. </html>
```

ligne 3 : inclure le script aframe

ligne22 : inclure le tag <a-scene> : les éléments (entity) sont toutes inclus dedans

ligne31 : fin de la scène

une scène comprend des objets : cube, cercle, entité, caméra lumière...

Afin de faciliter le chargement de la page web on place les objets à télécharger dans une balise <a-assets>, on les identifie puis on les utilise.

Ligne 24-25 : préchargement de l'arbre au format (obj + mtl) (modèle récupéré sur <https://archive3d.net> gratuitement)

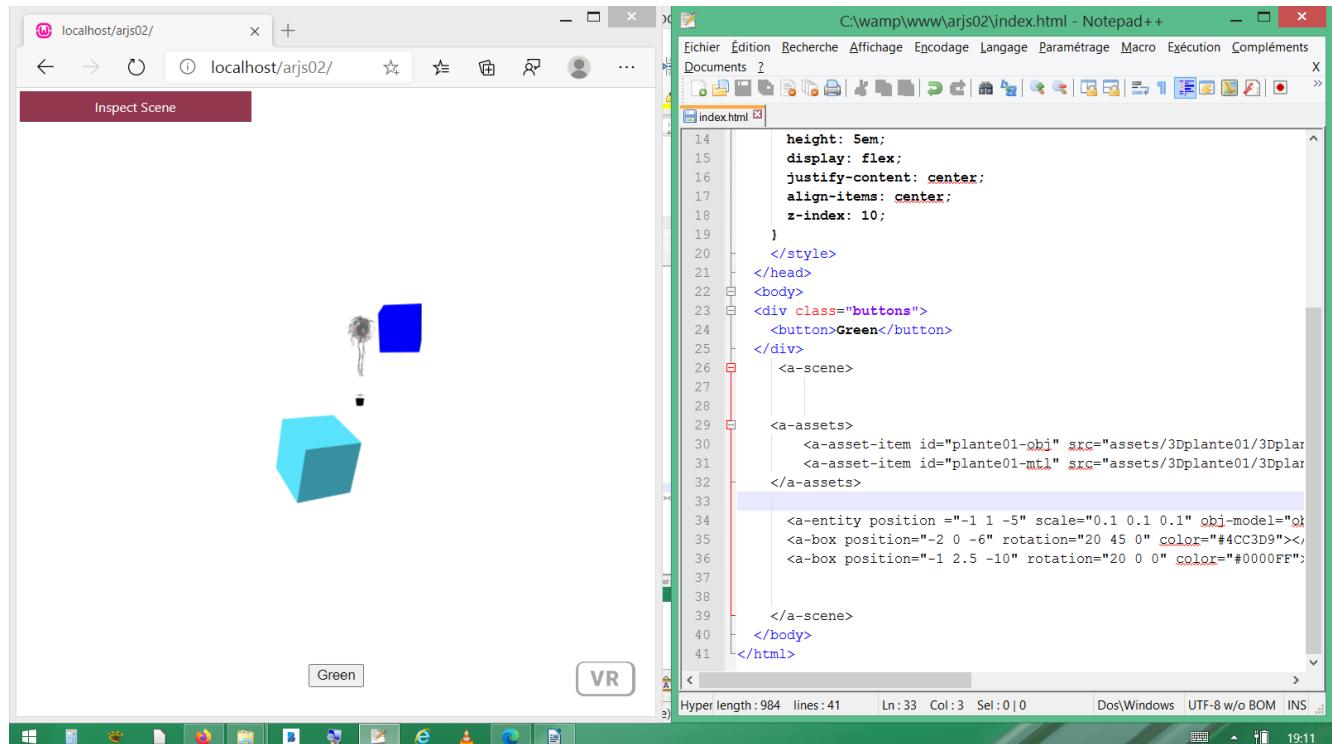
ligne 27 : utilisation de l'arbre pour qu'il s'affiche

ligne 28-29 : deux cubes (objets de bases dans a-frame

les propriétés utilisables sont détaillées sur :
<https://aframe.io/docs/1.1.0/introduction/>

1.2 Scène VR

Le résultat de la scène est :



Du HTML peut être inclus dans le fichier.(Le bouton dans l'exemple)

Un inspector est fourni avec A-frame, pour le lancer il suffit de faire Ctrl+Alt+i dans la page web.

2. Création d'un environnement AR

2.1 Exemple de code commenté

Dans le code précédent il suffit de rajouter :

le script vers AR.js : (ligne 4)

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

l'attribut arjs à la scène : (ligne 23)

```
<a-scene embedded arjs>
```

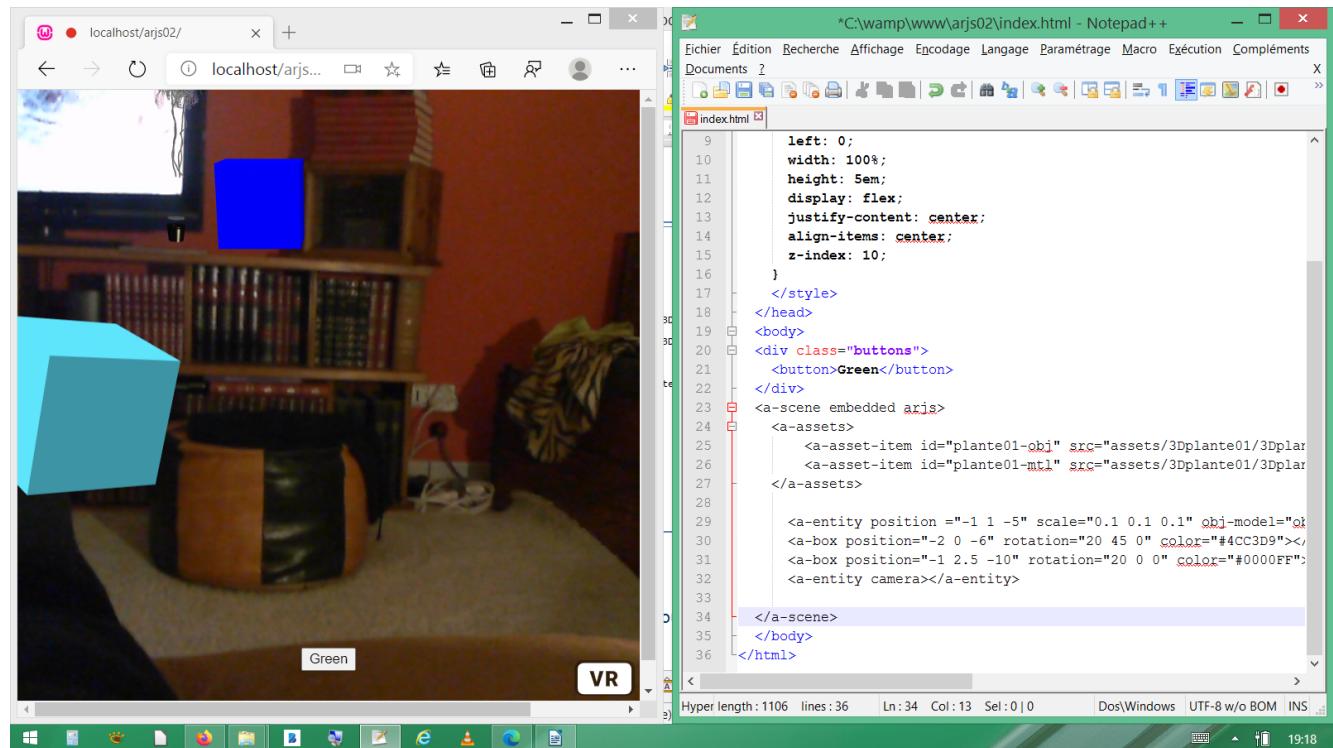
une caméra : (ligne 32)

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

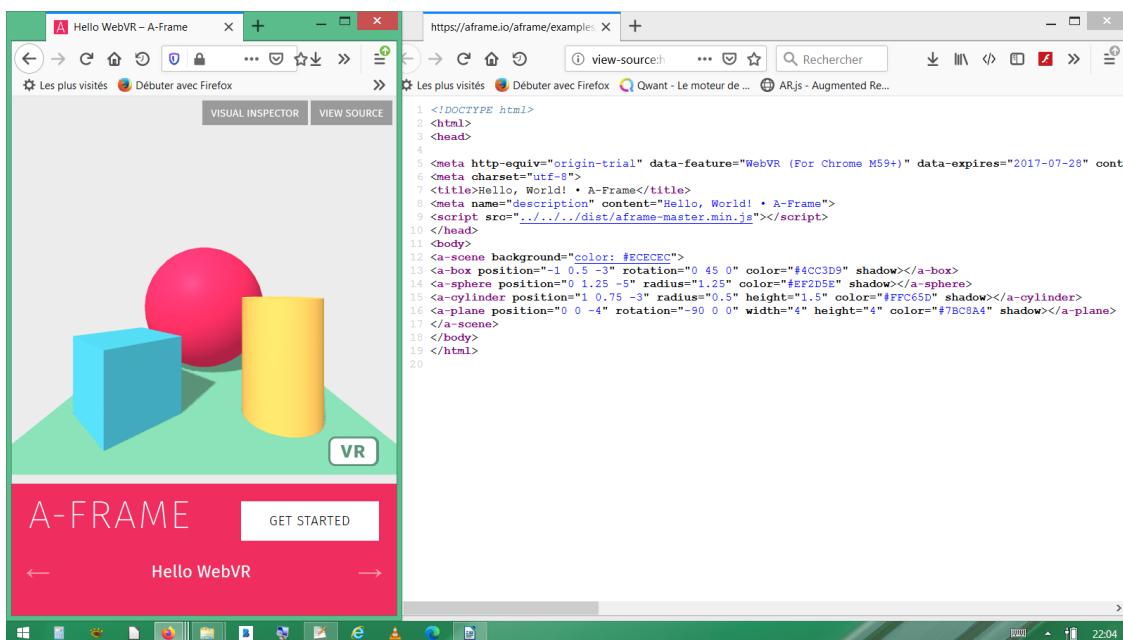
Le code complet est donc :

```
1. <html>
2.   <head>
3.     <script src="https://aframe.io/releases/1.1.0/aframe.min.js"></script>
4.     <script src="https://raw.githack.com/AR-js-org/AR.js/master/aframe/build/aframe-ar.js"></script>
5.   <style>
6.     .buttons {
7.       position: absolute;
8.       bottom: 0;
9.       left: 0;
10.      width: 100%;
11.      height: 5em;
12.      display: flex;
13.      justify-content: center;
14.      align-items: center;
15.      z-index: 10;
16.    }
17.   </style>
18. </head>
19. <body>
20. <div class="buttons">
21.   <button>Green</button>
22. </div>
23. <a-scene embedded arjs>
24.   <a-assets>
25.     <a-asset-item id="plante01-obj" src="assets/3Dplante01/3Dplante01.obj"></a-asset-item>
26.     <a-asset-item id="plante01-mtl" src="assets/3Dplante01/3Dplante01.mtl"></a-asset-item>
27.   </a-assets>
28.
29.   <a-entity position="-1 1 -5" scale="0.1 0.1 0.1" obj-model="obj: #plante01-obj; mtl: #plante01-mtl"></a-entity>
30.   <a-box position="-2 0 -6" rotation="20 45 0" color="#4CC3D9"></a-box>
31.   <a-box position="-1 2.5 -10" rotation="20 0 0" color="#0000FF"></a-box>
32.   <a-entity camera></a-entity>
33.
34. </a-scene>
35. </body>
36. </html>
```

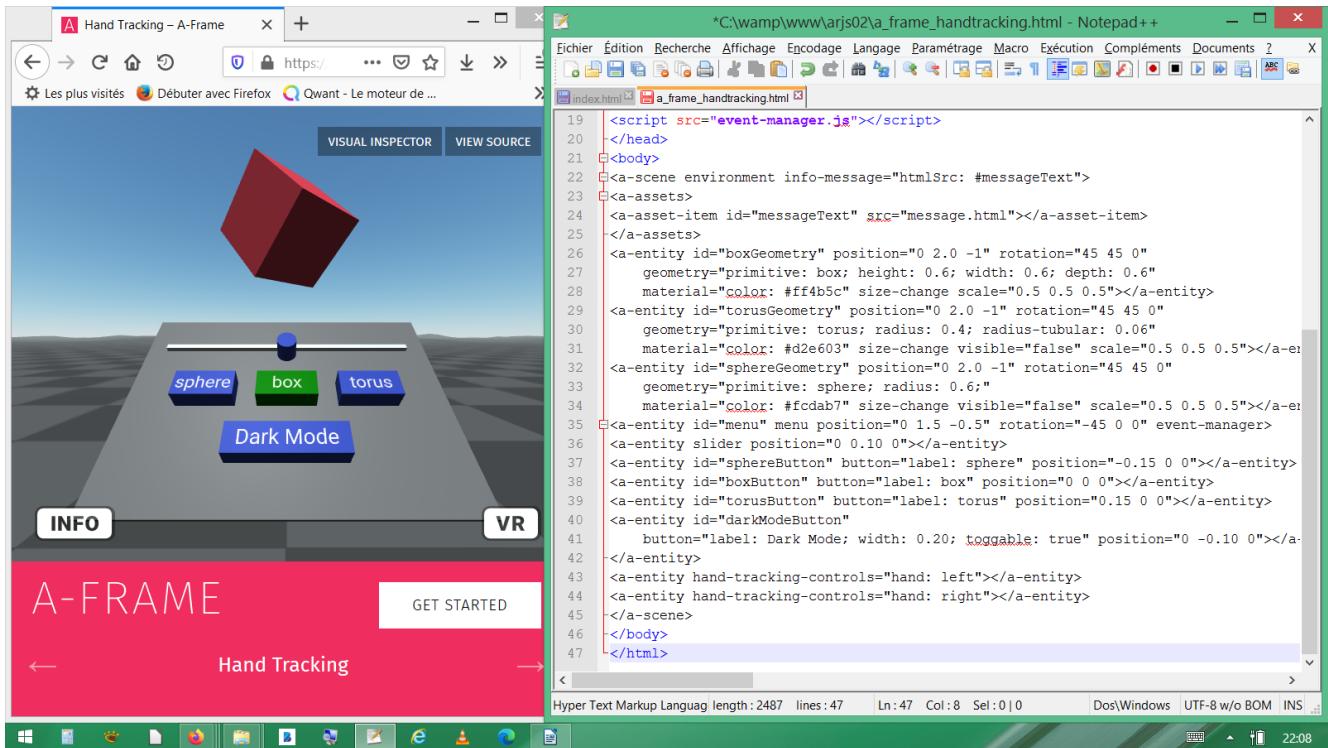
2.2 Résultat : scène AR



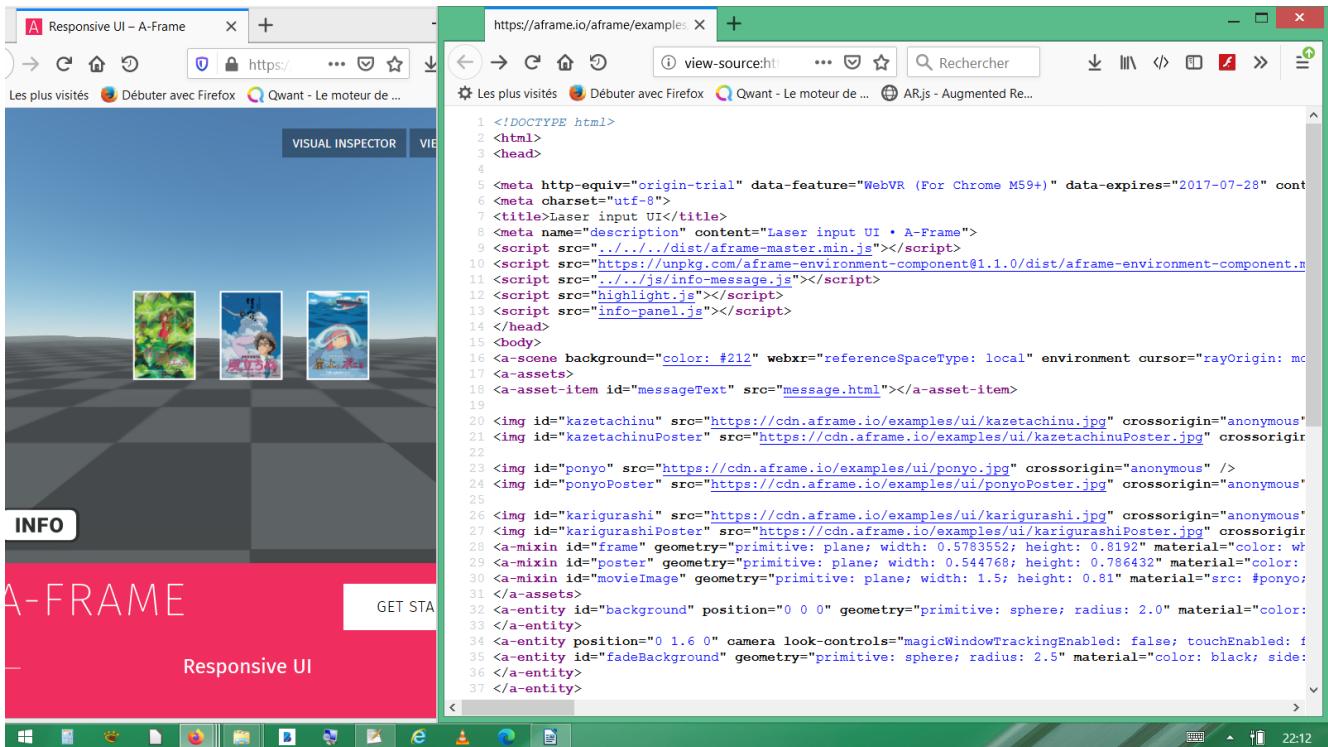
3. Exemple Hello VR



4. Suivi des mains : hand tracking



5. Interface utilisateur : Responsive UI



5.1 Le code :

```

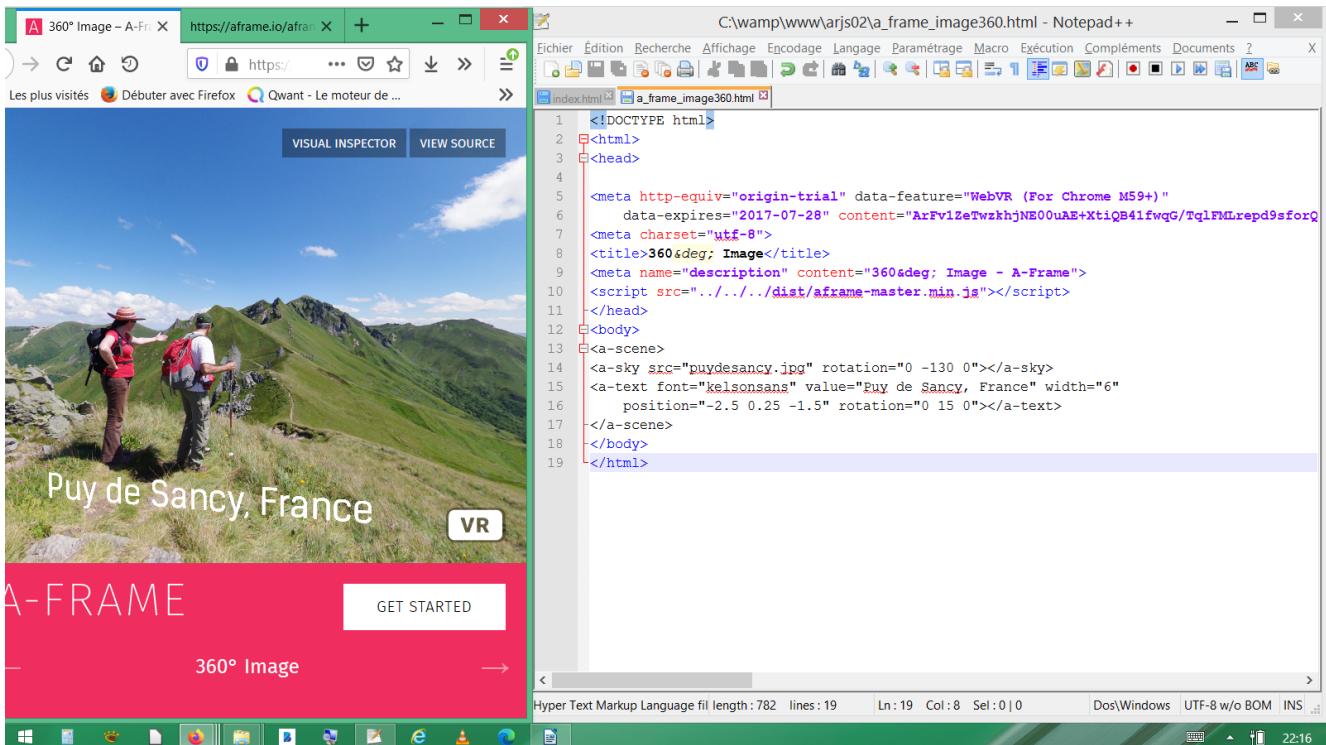
1. <!DOCTYPE html>
2. <html>
3. <head>
4.
5. <meta http-equiv="origin-trial" data-feature="WebVR (For Chrome M59+)" data-expires="2017-07-28"
content="ArFv1ZeTwzkhjNE00uAE+XtiQB41fwqG/TqlFMLrepd9sforQSVQE/tgfIbUMYNuNre4QR1k4/z8xp2mV3dbhwwAAABeeyJ
vcmlnaW4i0iJodHRwczovL2FmcmtZS5pbzo0NDMiLCJmZWFOdXJ1IjoiV2ViVlIxLjEiLCJleHBpcnkiOjE1MDEyMTcwMDIsImlzU3V
iZG9tYWluIjp0cnVlFQ==">
6. <meta charset="utf-8">
7. <title>Laser input UI</title>
8. <meta name="description" content="Laser input UI • A-Frame">
9. <script src="../../dist/aframe-master.min.js"></script>
10. <script src="https://unpkg.com/aframe-environment-component@1.1.0/dist/aframe-environment-component.min.js"></script>
11. <script src="../../js/info-message.js"></script>
12. <script src="highlight.js"></script>
13. <script src="info-panel.js"></script>
14. </head>
15. <body>
16. <a-scene background="color: #212" webxr="referenceSpaceType: local" environment cursor="rayOrigin: mouse; fuse: false" raycaster="objects: .raycastable" info-message="htmlSrc: #messageText">
17. <a-assets>
18. <a-asset-item id="messageText" src="message.html"></a-asset-item>
19.
20. 
21. 
22.
23. 
24. 
25.
26. 
27. 
28. <a-mixin id="frame" geometry="primitive: plane; width: 0.5783552; height: 0.8192" material="color: white; shader: flat" animation_scale="property: scale; to: 1.2 1.2 1.2; dur: 200; startEvents: mouseenter" animation_scale_reverse="property: scale; to: 1 1 1; dur: 200; startEvents: mouseleave"></a-mixin>
29. <a-mixin id="poster" geometry="primitive: plane; width: 0.544768; height: 0.786432" material="color: white; shader: flat" material="shader: flat" position="0 0 0.005"></a-mixin>
30. <a-mixin id="movieImage" geometry="primitive: plane; width: 1.5; height: 0.81" material="src: #ponyo; shader: flat; transparent: true" position="0 0.495 0.002"></a-mixin>
31. </a-assets>
32. <a-entity id="background" position="0 0 0" geometry="primitive: sphere; radius: 2.0" material="color: red; side: back; shader: flat" scale="0.001 0.001 0.001" visible="false" class="raycastable">
33. </a-entity>
34. <a-entity position="0 1.6 0" camera look-controls="magicWindowTrackingEnabled: false; touchEnabled: false; mouseEnabled: false">
35. <a-entity id="fadeBackground" geometry="primitive: sphere; radius: 2.5" material="color: black; side: back; shader: flat; transparent: true; opacity: 0.6" visible="false">
36. </a-entity>

```

AR en javascript : A-Frame et AR.js -

```
37. </a-entity>
38.
39. <a-entity id="leftHand" laser-controls="hand: left" raycaster="objects: .raycastable"></a-entity>
40. <a-entity id="rightHand" laser-controls="hand: right" raycaster="objects: .raycastable" line="color: #118A7E"></a-entity>
41. <a-entity id="ui" position="0 1.6 -2.5">
42.
43. <a-entity id="menu" highlight>
44. <a-entity id="karigurashiButton" position="-0.8 0 0" mixin="frame" class="raycastable menu-button">
45. <a-entity material="src: #karigurashiPoster;" mixin="poster"></a-entity>
46. </a-entity>
47. <a-entity id="kazetachinuButton" position="0 0 0" mixin="frame" class="raycastable menu-button">
48. <a-entity material="src: #kazetachinuPoster" mixin="poster"></a-entity>
49. </a-entity>
50. <a-entity id="ponyoButton" position="0.8 0 0" mixin="frame" class="raycastable menu-button">
51. <a-entity material="src: #ponyoPoster" mixin="poster"></a-entity>
52. </a-entity>
53. </a-entity>
54.
55. <a-entity id="infoPanel" position="0 0 0.5" info-panel visible="false" scale="0.001 0.001 0.001" geometry="primitive: plane; width: 1.5; height: 1.8" material="color: #333333; shader: flat; transparent: false" class="raycastable">
56. <a-entity id="ponyoMovieImage" mixin="movieImage" material="src: #ponyo" visible="false"></a-entity>
57. <a-entity id="kazetachinuMovieImage" mixin="movieImage" material="src: #kazetachinu" visible="false"></a-entity>
58. <a-entity id="karigurashiMovieImage" mixin="movieImage" material="src: #karigurashi" visible="false"></a-entity>
59. <a-entity id="movieTitle" position="-0.68 -0.1 0" text="shader: msdf; anchor: left; width: 1.5; font: https://cdn.aframe.io/examples/ui/Viga-Regular.json; color: white; value: Ponyo (2003)"></a-entity>
60. <a-entity id="movieDescription" position="-0.68 -0.2 0" text="baseline: top; shader: msdf; anchor: left; font: https://cdn.aframe.io/examples/ui/Viga-Regular.json; color: white; value: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."></a-entity>
61. </a-entity>
62. </a-entity>
63. </a-scene>
64. </body>
65. </html>
66.
```

6. Visualiser image 360°



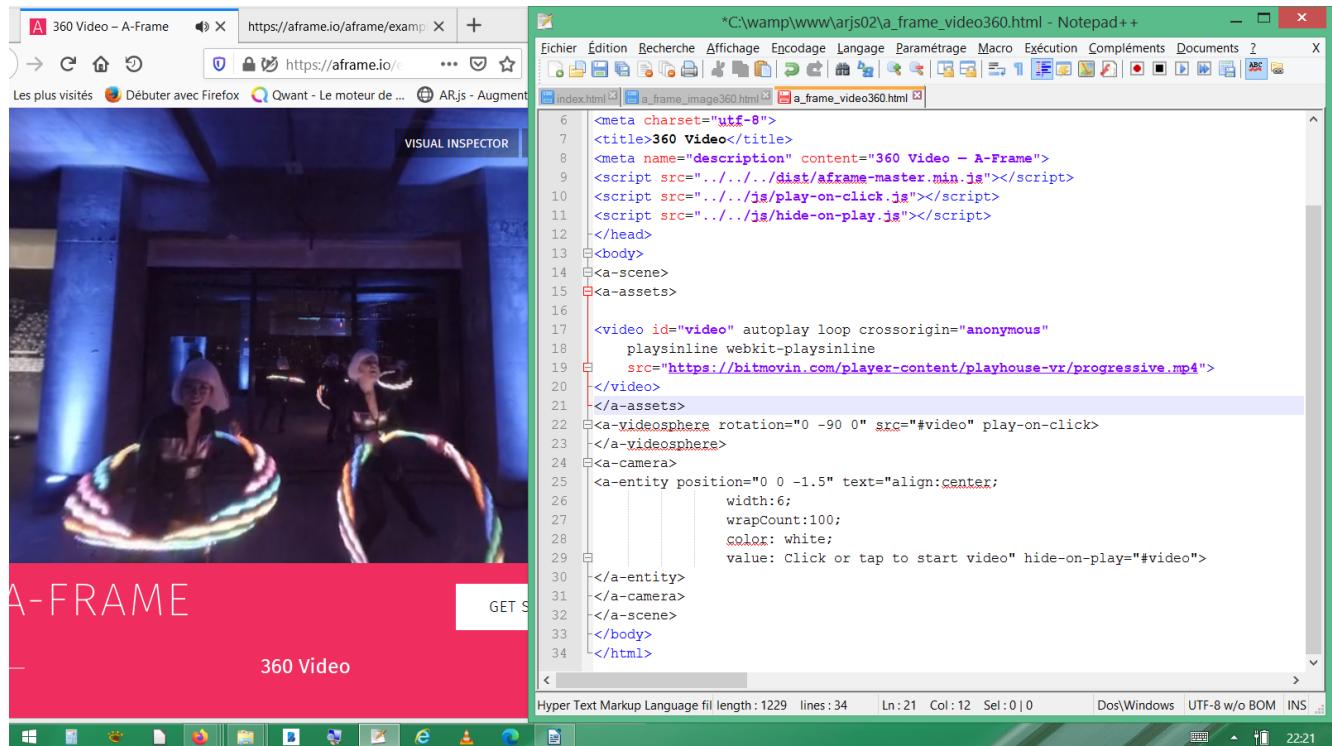
6.1 Le code

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4.
5. <meta http-equiv="origin-trial" data-feature="WebVR (For Chrome M59+)"
   data-expires="2017-07-28" content="ArFv1ZeTwzkhjNE00uAE+XtiQB41fwqG/TqlFMLrepd9sforQSvQE/tgfIbUMYNuNre4QR1k4/z8xp2mV3dbhwwAAABeeyJ
   vcmlnaW4i0iJodHRwczovL2FmcmtZS5pbzo0NDMiLCJmZWf0dXJ1IjoiV2ViVlIxLjEiLCJleHBpcnkiOjE1MDEyMTcwMDIsImlzU3V
   iZG9tYWluIjp0cnVlfQ==">
6. <meta charset="utf-8">
7. <title>360&deg; Image</title>
8. <meta name="description" content="360&deg; Image - A-Frame">
9. <script src="../../dist/aframe-master.min.js"></script>
10. </head>
11. <body>
12. <a-scene>
13.   <a-sky src="puydesancy.jpg" rotation="0 -130 0"></a-sky>
14.   <a-text font="kelsonsans" value="Puy de Sancy, France" width="6"
      position="-2.5 0.25 -1.5" rotation="0 15 0"></a-text>
15. </a-scene>
16. </body>
17. </html>

```

7. Lecteur vidéo 360



7.1 Le code

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4.
5. <meta http-equiv="origin-trial" data-feature="WebVR (For Chrome M59+)" data-expires="2017-07-28"
   content="ArFv1ZeTwzkhjNE00uAE+XtiQB41fwgQ/TqlFMLrepd9sforQSvQE/tgfibUMYNuNre4QR1k4/z8xp2mV3dbhwwAAABeeyJ
   vcmlnaW4i0iJodHRwczovL2FmcmtzS5pbzo0NDMiLCJmZWFOdXJ1IjoiV2ViVlIxLjEiLCJleHBpcnkiOjE1MDEyMTcwMDIsImlzU3V
   iZG9tYWluIjp0cnVlfQ==">
6. <meta charset="utf-8">
7. <title>360 Video</title>
8. <meta name="description" content="360 Video – A-Frame">
9. <script src="../../dist/aframe-master.min.js"></script>
10. <script src="../../js/play-on-click.js"></script>
11. <script src="../../js/hide-on-play.js"></script>
12. </head>
13. <body>
14. <a-scene>
15. <a-assets>
16.
17. <video id="video" autoplay loop crossorigin="anonymous"
18.       playsinline webkit-playsinline
19.       src="https://bitmovin.com/player-content/playhouse-vr/progressive.mp4">
20. </video>
21. </a-assets>
22. <a-videosphere rotation="0 -90 0" src="#video" play-on-click>
23. </a-videosphere>
24. <a-camera>
25.   <a-entity position="0 0 -1.5" text="align:center;
26.           width:6;
27.           wrapCount:100;
28.           color: white;
29.           value: Click or tap to start video" hide-on-play="#video">
30.   </a-entity>
31. </a-camera>
32. </a-scene>
33. </body>
34. </html>

```

AR en javascript : A-Frame et AR.js -

```
23. </a-videosphere>
24. <a-camera>
25. <a-entity position="0 0 -1.5" text="align:center;
26.           width:6;
27.           wrapCount:100;
28.           color: white;
29.           value: Click or tap to start video" hide-on-play="#video">
30. </a-entity>
31. </a-camera>
32. </a-scene>
33. </body>
34. </html>
```

8. Géolocalisation GPS + reconnaissance image

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

9. Géolocalisation + reconnaissance par marqueur

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

10. Animation

Rotation de l'objet

Les codes <animation> ne fonctionnent pas !

Le code suivant fonctionne bien (extrait de animation.html de github.com/stemkoski/AR.js-examples.)

Ajout du code dans la balise entity à faire tourner :

```
<a-entity
  id="carteCI"
  position ="0 0 -4"
  rotation="0 -90 -90"
```

AR en javascript : A-Frame et AR.js -

```
scale="1 1 1"
obj-model="obj: #CI_obj; mtl: #CI_mtl"
animation__rotate = "property: rotation; dur: 10000; easing: linear; dir: normal; from:0 0 0; to: 0
360 0; loop: true;">

</a-entity>
```

L'objet tourne correctement.

11. Réalisation interface utilisateur avec A-frame

Création d'une interface utilisateur : GUI : <https://rdub80.github.io/aframe-gui/>

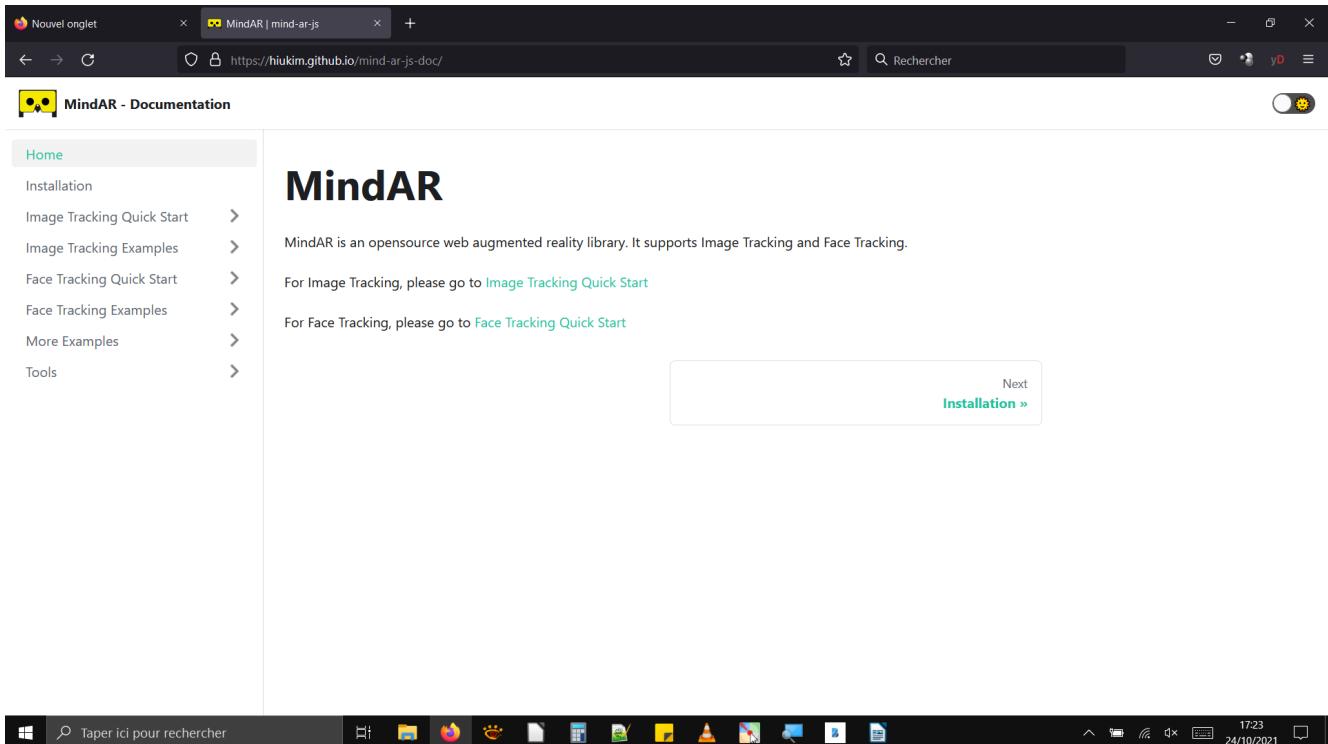


Les exemples sont ici : <https://rdub80.github.io/aframe-gui/examples/index.html>

12. Utilisation Mind-AR

L'utilisation de Mind-AR est simplifié par rapport à ARJS car nous n'avons pas besoin d'un serveur sécurisé. Votre wamp installé sur votre PC suffira à faire les tests.

Sur Qwant chercher : 'mind-ar' et aller vers le lien de la doc :



suivez les liens en bas à droite et expérimenter les exemples.

13. Bibliographie

A-frame : <https://aframe.io/> (pdf : <https://riptutorial.com/Download/aframe.pdf>)

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

AR-js : tutoriel en anglais YouTube :
<https://www.youtube.com/channel/UCK2U85UiKw45EBAUS1YU85Q>
(danielo Pasquariello)

évènement clic... : <https://github.com/amitwaghmare/aframe-event-set-component>

modèle 3d gratuit : <https://archive3d.net/>

Tutoriel en Français : <https://riptutorial.com/fr/aframe/example/31091/demarrer-pour-ar>

Exemples complets et opérationnels : github.com/stemkoski/AR.js-examples. : A FAIRE

Mind-AR : librairie permettant de rapidement créer un appli web AR (image tracking ou face tracking) : <https://hiukim.github.io/mind-ar-js-doc/>

Création d'une interface utilisateur : GUI : <https://rdub80.github.io/aframe-gui/>