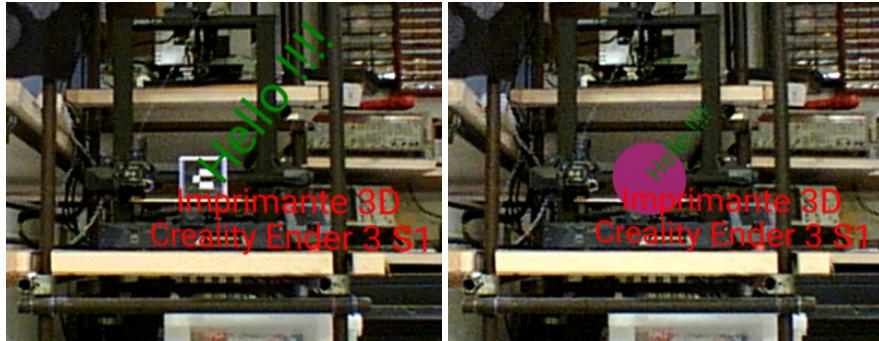


*Voici l'exemple suivant avec le marqueur visible et le marqueur caché par un cercle mauve.*



```
Individuel.html [changes/repo/repo/Bureau/Colin/website] - Notepad++  
File Edit Insert View Source Languages Setting Help [Envir. Env.]  
  
<html>  
  <head>  
    <title> Individuel </title>  
  </head>  
  <body>  
    <h1> Individuel </h1>  
  
    <div id="content">  
      <h2> Individuel </h2>  
      <p> Individuel </p>  
    </div>  
  
    <script>  
      console.log("app Martini");  
      const app = Martini()  
      app.get("/", function(req, res) {  
        res.send("Hello world")  
      })  
      app.listen(3000)  
    </script>  
  </body>  
</html>
```

**Ce code HTML** est une page web simple qui utilise A-Frame et AR.js pour créer une application de réalité augmentée (AR). Voici une explication détaillée de chaque partie du code :

## **<head>**

**<title>** : Définit le titre de la page web qui s'affiche dans l'onglet du navigateur. Ici, le titre est "Ma première app AR".

```
<script src="https://aframe.io/releases/1.7.0/aframe.min.js"></script>
```

Charge la bibliothèque A-Frame, un framework pour créer des expériences de réalité virtuelle (VR) et augmentée (AR) en utilisant des éléments HTML.

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

**Charge AR.js**, une extension d'A-Frame qui permet d'ajouter des fonctionnalités de réalité augmentée, comme la reconnaissance de marqueurs.

## <body>

### **< a-scene >:**

C'est l'élément **principal** d'A-Frame qui représente la scène 3D. Il contient divers attributs pour configurer la scène AR :

**embedded**: Indique que la scène est intégrée dans la page web plutôt que d'occuper tout l'écran.

**args:** Configure les paramètres spécifiques à AR.js :

1. sourceType: `webcam` : Utilise la webcam comme source vidéo.
  2. detectionMode: `mono\_and\_matrix` : Déetecte à la fois les marqueurs simples et les matrices de marqueurs.
  3. matrixCodeType: `3x3` : Spécifie le type de matrice de marqueurs (ici, une matrice  $3 \times 3$ ).
  4. trackingMethod: `best` : Utilise la meilleure méthode de suivi disponible.
  5. changeMatrixMode: `modelViewMatrix` : Change le mode de matrice pour le rendu.

**renderer:** Configure les paramètres du rendu :

1. sortObjects: true : Trie les objets pour un rendu correct.
  2. antialias: true : Active l'anti-aliasing pour des bords plus lisses.

3. colorManagement: true : Gère les couleurs pour un rendu plus précis.
4. logarithmicDepthBuffer: true : Utilise un tampon de profondeur logarithmique pour améliorer le rendu des objets proches et éloignés.

**vr-mode-ui="enabled: false"**: Désactive l'interface utilisateur du mode VR.

**smooth**: Lisse les mouvements de la caméra pour une expérience plus fluide

**smoothCount, smoothTolerance, smoothThreshold** : Paramètres pour contrôler le lissage.

**sourceWidth, sourceHeight, displayWidth, displayHeight**: Définit les dimensions de la source vidéo et de l'affichage.

**Ce code configure une scène AR de base** qui utilise la webcam pour détecter des marqueurs et afficher des objets 3D en réalité augmentée.

Pour une application complète, vous devrez ajouter des éléments 3D (comme `<a-box>`, `<a-sphere>`, etc.) à l'intérieur de la balise `<a-scene>`. Il est aussi possible d'ajouter du son, une vidéo...

[exemple video](#)

[exemple son](#)

brief presque tout est possible de faire mais parfois il faut penser à ajouter la bib correspondante dans le head du fichier HTML : Par exemple pour insérer un décor :

`unpkg.com/aframe-environment-component@1.5.x/dist/aframe-environment-component.min.js">`

**Se référencer à** <https://aframe.io/> [accès A-Frame]]

Article extrait de : <http://www.lesporteslogiques.net/wiki/> - **WIKI Les Portes Logiques**

Adresse : [http://www.lesporteslogiques.net/wiki/ressource/code/application\\_au\\_fablab](http://www.lesporteslogiques.net/wiki/ressource/code/application_au_fablab)

Article mis à jour: **2025/03/27 16:16**