Batch: B2 Roll No.: 16010121110

Experiment No. 12

TITLE: Introduction to AR.js

AIM:
Explore the AR.JS for Web AR
Design the object using any designing tool like blender
Use AR.js to augment it in real world

Expected OUTCOME of Experiment:

Get experience of working with Augmented and Virtual Reality

Books/ Journals/ Websites referred:

https://www.youtube.com/watch?v=2ypJ9CFOK5U&list=PLTgRMOcmRb3Nx2LF5EHU4MtmpAQBafVgE&index=1&ab_channel=Packt

---

Steps to perform:

1) **Import CDN a-frame**
2) **Make a simple app using mobile camera and AR**
3) **Use a-frame interactions**
4) **Host staticly**

Drive or GitHub link:

https://github.com/Aatmaj-Zephyr/Website-Playground/blob/698929b07a6546dcbe2cd83339125f096f6e80d6/go.html

```
<!DOCTYPE html>
<html>
<head>

<meta http-equiv="origin-trial" data-feature="WebVR (For Chrome M59+)" >
<meta http-equiv="origin-trial" data-feature=" WebXR Device API (For Chrome M69+)" >
<meta charset="utf-8">
<title>Video</title>
<meta name="description" content="Video - A-Frame">
<script
src="https://cdnjs.cloudflare.com/ajax/libs/aframe/1.4.2/aframe.js"></script>

</head>
<script>

var score=0;
window.onload = function (){
   console.log("loading website")
   var clickSound = document.getElementById("clickSound");
   var victorySound = document.getElementById("victorySound");


};

document.addEventListener('mousedown', function() {
   clickSound.pause();//if previous sound is still playing
   // Play the audio
   clickSound.currentTime = 0; // Rewind the audio to the beginning
   clickSound.play();

 });

function updateScore(){ //update the score in the scoreboard window
   document.getElementById("scoreboard").setAttribute("value", score);
}
```

```
function addObj() {
  const scene = document.getElementById('scene');
  const object = document.createElement('a-sphere');

  object.setAttribute('radius', '0.5');
  object.setAttribute('color', '#743b8f');

  object.addEventListener('mousedown', function() {

    victorySound.pause();//if previous sound is still playing
    // Play the audio
    victorySound.currentTime = 0; // Rewind the audio to the beginning
    victorySound.play();

    scene.removeChild(object);
    console.log("clicked...")
    score+=1;
    updateScore()
    console.log("score="+score)

  });
  // id of the object to repeat, appear and disappear randomly

    randomPos=""+((0.5-Math.random())*20)+" "+((0.5-Math.random())*20)+"
"+((0.5-Math.random())*20)+""
    object.setAttribute('position',randomPos)
    console.log(randomPos)

  scene.appendChild(object);
}




setInterval(addObj, 3000);
 function clicked(){
    console.log("Clicked..")

 }
    </script>

<body>



    <audio id="clickSound" style="display: none;">
        <source src="bullet1.mp3" type="audio/mp3">
```

```
    </audio>
    <audio id="victorySound" style="display: none;">
      <source src="victory.mp3" type="audio/mp3">

    </audio>
<a-scene id="scene">
    <!-- Create a plane -->
    <a-plane color="#b4ff85" width="4" height="2" position="3 5
-4.01"></a-plane>

    <!-- Add text on the plane -->
    <a-text id="scoreboard" value="Welcome!" color="black" position="3 5 -4"
align="center"></a-text>

<a-assets>


</a-assets>

</a-entity>
<a-entity id="rig" position="25 10 0">
  <a-entity id="camera" camera look-controls></a-entity>
</a-entity>



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

</body>
</html>
```

Output(s) (Screen Shots):

Conclusion and discussion:

Thus we have understood how a-frame Ar works.
We have implemented a simple game in AR. The balloons shall pop up and we have to hit them in AR. We understood how to develop Ar in the web. Ar in the web is easy to develop and deploy. However the problem is that it is not very good in the terms of performance.

Date: 14 oct 2023

Signature of faculty in-charge