

Realidade Aumentada Mobile

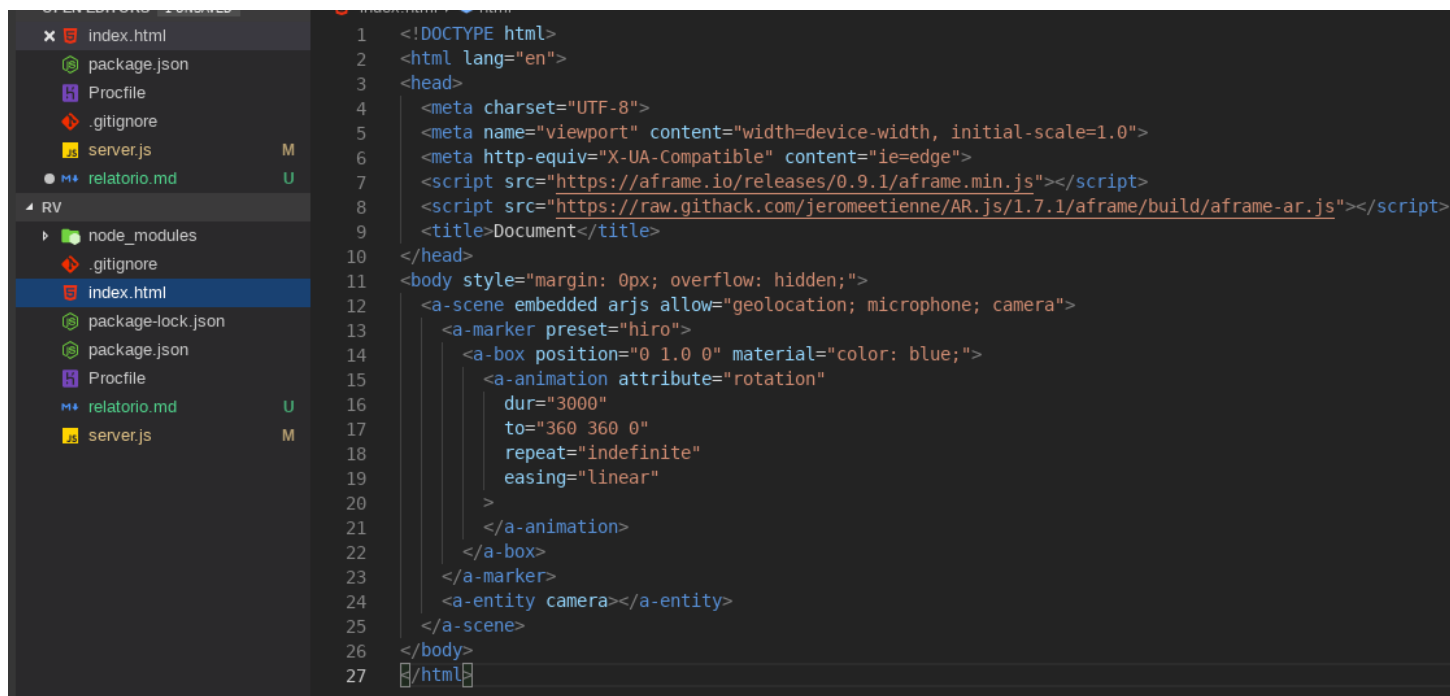
Essa aplicação de realidade aumentada foi feita em javascript na plataforma node.

A aplicação executa no navegador de qualquer celular com camera frontal. Para fazer tal aplicação foi utilizado uma biblioteca em javascript chamada de AR.js.

A aplicação é servida através do servidor node.js e foi implantada no host da heroku.

Código da view da aplicação

É nessa parte do código que está configurado o uso da biblioteca AR.js, e também estão configurados o o marcador, o uso da camera para detectar o marcador e o objeto virtual gerado.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <meta http-equiv="X-UA-Compatible" content="ie=edge">
7   <script src="https://aframe.io/releases/0.9.1/aframe.min.js"></script>
8   <script src="https://raw.githubusercontent.com/jeromeetienne/AR.js/1.7.1/aframe/build/aframe-ar.js"></script>
9 </head>
10 <body style="margin: 0px; overflow: hidden;">
11   <a-scene embedded arjs allow="geolocation; microphone; camera">
12     <a-marker preset="hiro">
13       <a-box position="0 1.0 0" material="color: blue;">
14         <a-animation attribute="rotation"
15           dur="3000"
16           to="360 360 0"
17           repeat="indefinite"
18           easing="linear"
19         >
20       </a-animation>
21     </a-box>
22   </a-marker>
23   <a-entity camera></a-entity>
24 </a-scene>
25 </body>
26 </html>
```

Código do servidor

Esse é código onde está configurado o servidor node, por meio do qual a aplicação será disponibilizada em uma determinada porta.



```
1 const express = require('express')
2 const app = express()
3 const path = require('path')
4 const cors = require('cors')
5 app.set("/", "html")
6 app.use(cors())
7 app.use(express.static(path.join(__dirname, "/")))
8 app.use(express.json())
9 app.use(express.urlencoded({ extended: true }))
10 app.use(function (req, res, next) {
11   res.setHeader('Access-Control-Allow-Origin', `http://localhost:${process.env.PORT}`);
12   res.setHeader('Access-Control-Allow-Methods', 'GET, POST, OPTIONS, PUT, PATCH, DELETE');
13   res.setHeader('Access-Control-Allow-Headers', 'X-Requested-With,content-type');
14   res.setHeader('Access-Control-Allow-Credentials', true);
15   next();
16 });
17
18 app.get('/', (req, res) => {
19   res.render('index')
20 })
21
22 app.listen(process.env.PORT, () => {
23   console.log('Listening on port')
24 })
```

Link de acesso à aplicação

link para a [RV](#) disponibilizada no heroku