

**LAPORAN PROJEK UAS VR- AR KELAS (E1)**  
**SISTEM KOMPUTER GENAP**  
**TAHUN 2024**

**Nama Ketua : Patricia Michelle Octaviane Y**  
**NIM : 22202182**  
**Nama Anggota : Muhammad Oktavio K**  
**NIM : 22202084**  
**Project UAS VR-AR : Car Drift in Road**

**INSTITUT TEKNOLOGI DAN BISNIS ASIA MALANG**  
**FAKULTAS DESAIN DAN TEKNOLOGI**  
**JURUSAN SISTEM KOMPUTER**  
**2024**

## **SUB BAB LAPORAN PROJECT UAS**

- 1. Pendahuluan**
- 2. Project / platform yang di bahas pada VR \_AR**
- 3. Proses dan tahapan Membangun Object dengan platform UNITY atau Web.arjs**
  - a) Flowchat sistem**
  - b) Object detection X,Y,Z (Marker dll)**
  - c) scrip programing sederhana yang digunakan**
- 4. Hasil project uas dari platform UNITY atau Web.arjs**
- 5. Screanshot demo / link vidio demo**
- 6. Kesimpulan**
- 7. Refrensi**

## 1. Pendahuluan

Dalam beberapa tahun terakhir, teknologi Virtual Reality (VR) dan Augmented Reality (AR) telah mengalami perkembangan pesat, menawarkan pengalaman baru yang imersif dan interaktif. Salah satu aplikasi menarik dari AR adalah penggunaan teknologi marker-based AR, yang memungkinkan perangkat untuk mendeteksi dan melacak penanda khusus untuk menampilkan konten digital di dunia nyata. Salah satu penanda yang populer digunakan adalah marker Hiro.

**Marker-based AR** menggunakan penanda visual atau kode yang dapat dibaca oleh perangkat AR, seperti smartphone atau kacamata AR, untuk menampilkan objek digital, animasi, dan suara di lingkungan fisik pengguna. Marker Hiro adalah salah satu jenis

is penanda yang sering digunakan dalam aplikasi AR karena keandalannya dalam pelacakan dan kemampuannya untuk mengenali pola dengan akurasi tinggi.

Ketika sebuah perangkat AR mendeteksi marker Hiro, perangkat tersebut dapat memproses informasi dari penanda tersebut dan menampilkan konten digital yang telah diprogram, seperti animasi 3D atau efek suara.

Penggunaan teknologi Augmented Reality (AR) dengan marker Hiro bertujuan untuk meningkatkan interaksi dan pengalaman pengguna di berbagai bidang melalui visualisasi digital yang interaktif dan imersif. Teknologi ini memanfaatkan marker Hiro untuk memproyeksikan animasi dan suara di dunia nyata, memberikan informasi tambahan yang kaya dan kontekstual. Secara khusus, tujuan utama dari penerapan teknologi ini meliputi: Meningkatkan Pembelajaran dan Pendidikan, Memfasilitasi Pelatihan dan Simulasi, Meningkatkan Pengalaman Pemasaran dan Periklanan, Menyediakan Hiburan yang Lebih Interaktif, Meningkatkan Akses Informasi di Tempat.

Dengan memanfaatkan teknologi marker-based AR dan marker Hiro, kita dapat menciptakan solusi inovatif yang membawa manfaat signifikan dalam berbagai aspek kehidupan, mulai dari pendidikan dan pelatihan hingga pemasaran, hiburan, dan industri. Teknologi ini memungkinkan integrasi dunia digital dan fisik dengan cara yang lebih kaya, interaktif, dan efektif.

## 2. Project

Proyek yang menggabungkan Augmented Reality (AR), objek mobil yang ngedrift di atas jalan raya dan bersuara, penggunaan Visual Studio Code (VSCode) untuk mengoding, dan pengambilan objek dari Sketchfab, kita dapat melihat sebuah aplikasi atau pengalaman AR yang menarik.

Konsep Proyek Augmented Reality dengan Web AR.js :

Implementasi dan Hasil

Dengan menggunakan teknologi ini, Augmented Reality (AR) dengan menggunakan marker Hiro telah membawa dampak positif yang signifikan di berbagai bidang. Dalam pendidikan dan pelatihan, penggunaan marker Hiro memungkinkan pengalaman pembelajaran yang lebih mendalam dan interaktif.

Proses pengembangan AR berbasis web menggunakan ar.js melibatkan beberapa tahapan, antara lain:

#### 1. Pengumpulan dan Persiapan Data

Dalam tahap ini, kita mengumpulkan model 3D dan aset visual lainnya yang akan digunakan dalam aplikasi AR. Data ini bisa berupa model GLB atau format lain yang didukung oleh ar.js.

#### 2. Pemrosesan Data

Data yang telah dikumpulkan harus diproses dan disiapkan untuk digunakan dalam aplikasi AR. Ini melibatkan langkah-langkah seperti konversi format, optimisasi model 3D, dan penyesuaian ukuran file agar dapat dimuat dengan cepat di web.

#### 3. Pengembangan Aplikasi

Pada tahap ini, kita mulai mengembangkan aplikasi web menggunakan ar.js. Kita membuat HTML dan JavaScript untuk mengatur kamera, mendeteksi marker AR, dan menampilkan model 3D di atas marker tersebut. Ar.js menyediakan berbagai fitur yang memudahkan proses ini, seperti deteksi marker berbasis gambar dan lokasi.

#### 4. Pengujian dan Validasi

Setelah aplikasi dikembangkan, kita melakukan pengujian untuk memastikan bahwa aplikasi bekerja dengan baik di berbagai perangkat dan kondisi. Pengujian dilakukan untuk memastikan model 3D muncul dengan benar di atas marker dan interaksi berjalan lancar.

#### 5. Implementasi dan Distribusi

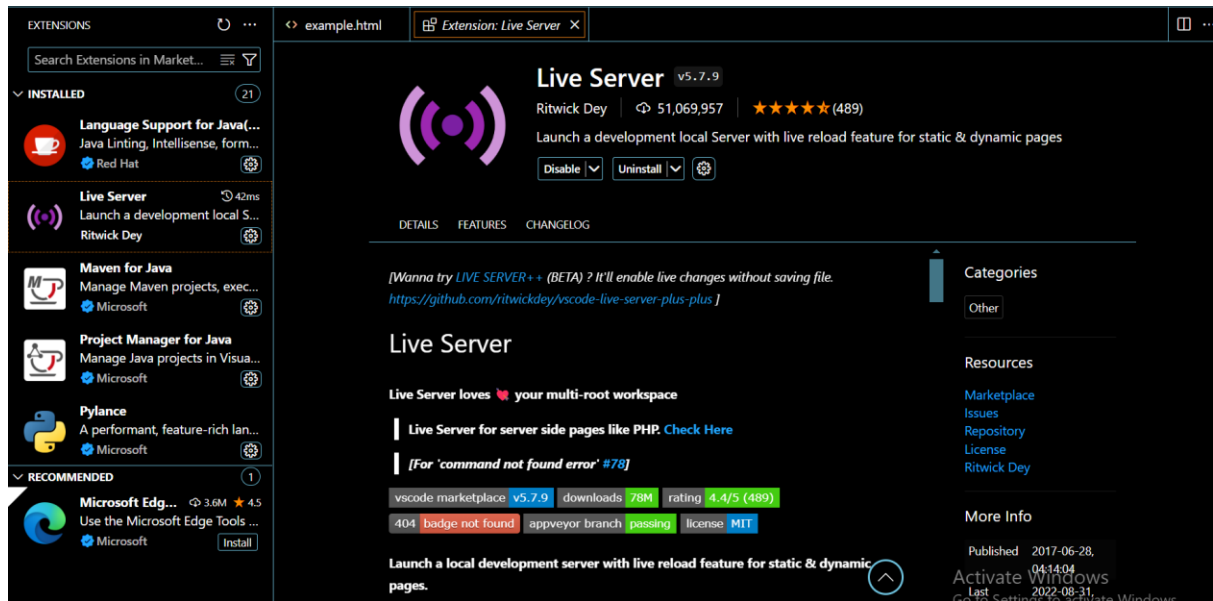
Setelah aplikasi selesai dan diuji, kita bisa mengimplementasikannya dalam berbagai konteks aplikasi web. Implementasi ini bisa mencakup integrasi dengan situs web yang sudah ada, pembuatan halaman web khusus untuk pengalaman AR, atau pengembangan aplikasi web progresif (PWA) untuk pengalaman AR yang lebih mendalam.

Proyek ini menggabungkan berbagai teknologi modern untuk menciptakan pengalaman yang menghibur dan edukatif. Pengembangan aplikasi ini memanfaatkan kekuatan VSCode sebagai editor kode yang fleksibel dan kuat, sambil mengintegrasikan objek 3D yang dinamis dari Sketchfab untuk memberikan visualisasi yang memukau dalam konteks AR.

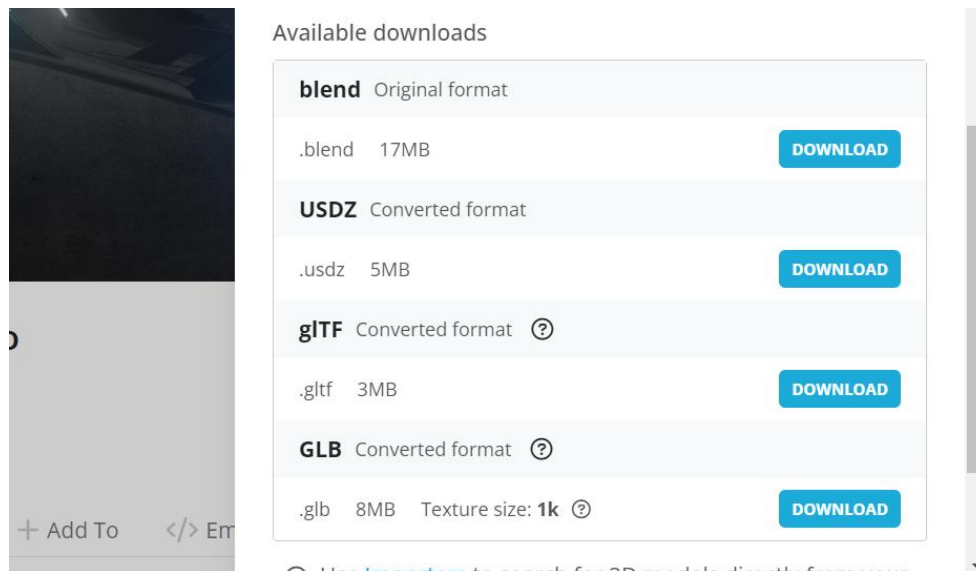
Dengan demikian, proyek ini tidak hanya mengeksplorasi batas-batas teknologi seperti AR dan pengembangan aplikasi, tetapi juga menghadirkan cara baru bagi pengguna untuk berinteraksi dengan konten digital dan dunia nyata secara simultan.

### 3. Proses dan Tahapan

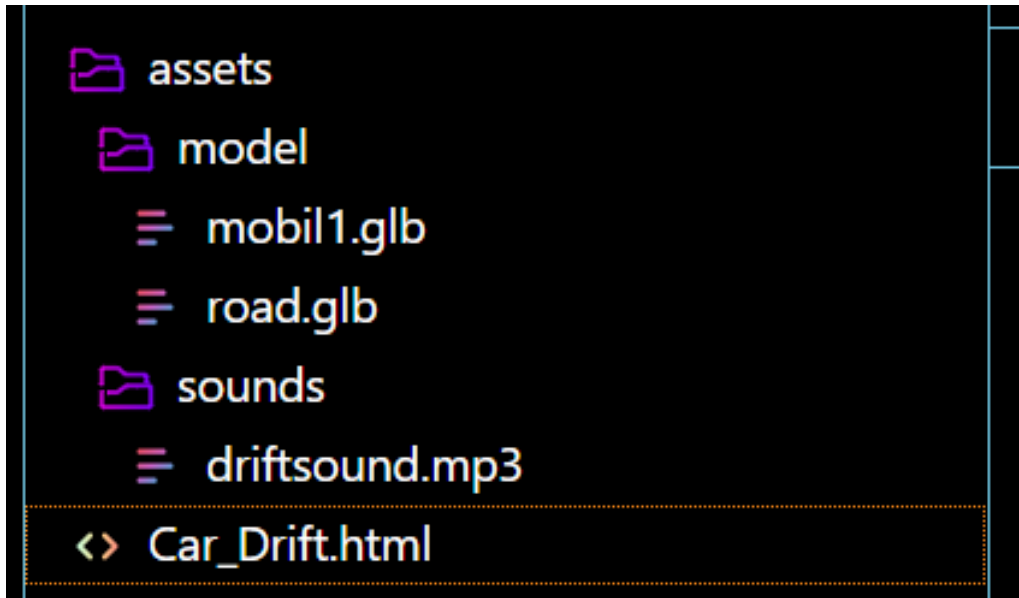
#### A. Proses :



Install extension live server pada VSCode



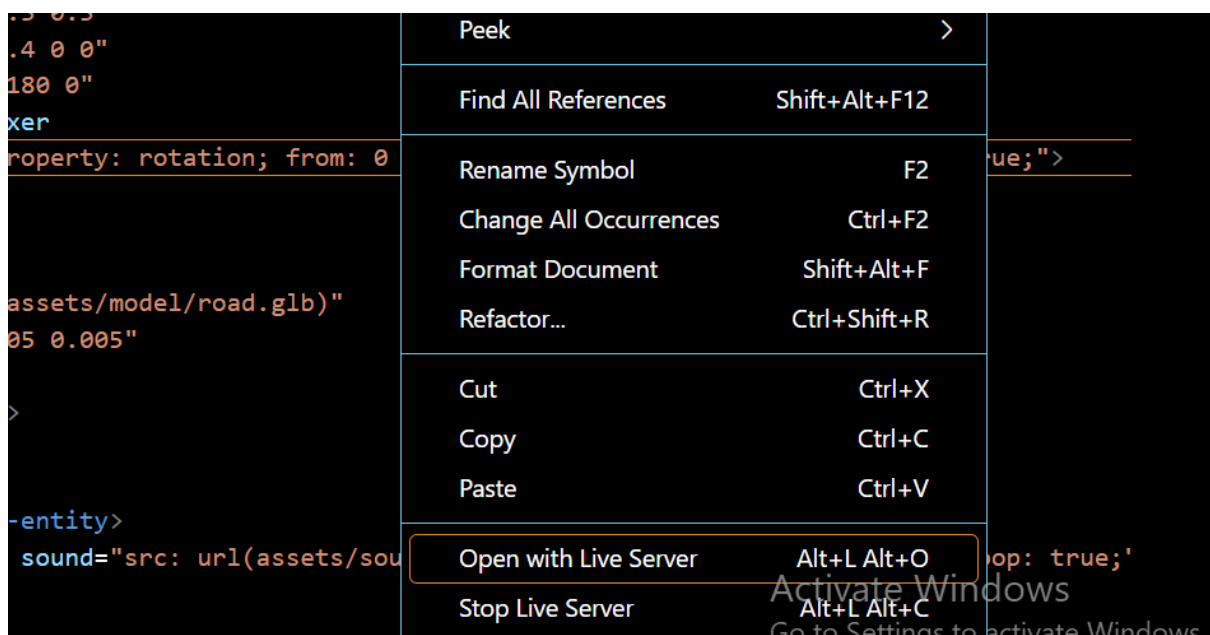
Download 3D Modelnya dengan format GLB



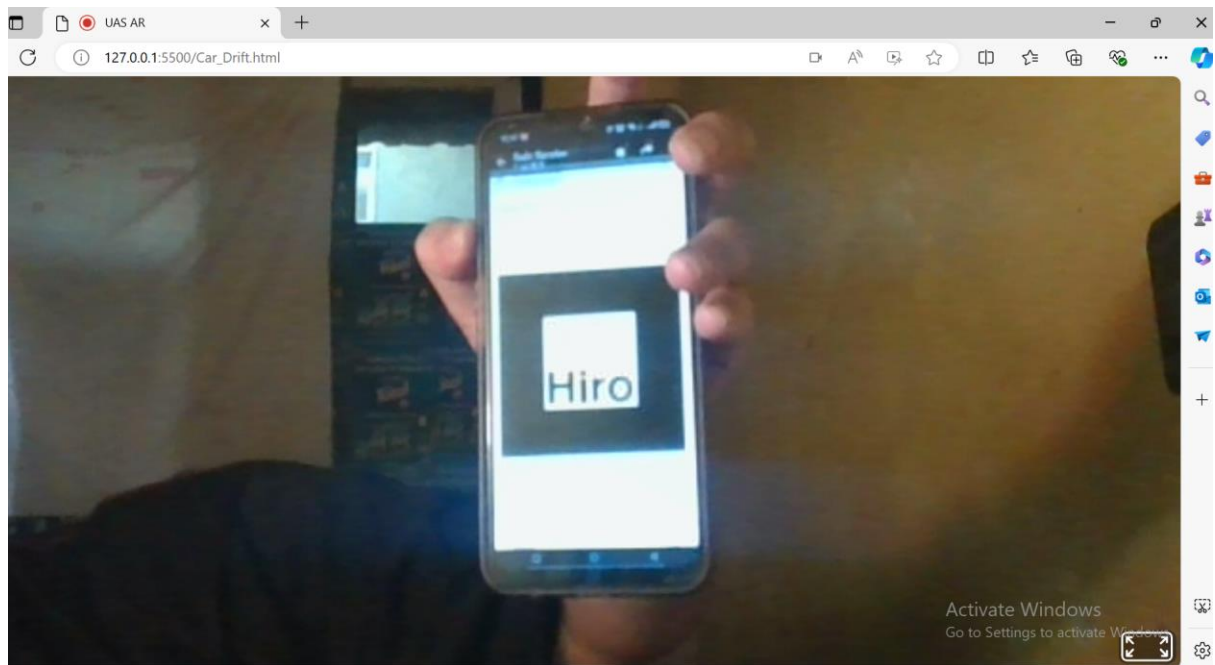
Masukan file 3Dnya ke dalam folder assets-model

```
<a-e
  id="mobil1"
  gltf-model="url(assets/model/mobil1.glb)"
  scale="0.3 0.3 0.3"
  position="-0.4 0 0"
  rotation="0 180 0"
  animation-mixer
  animation="property: rotation; from: 0 180 0; to: 0 540 0"
</a-entity>
```

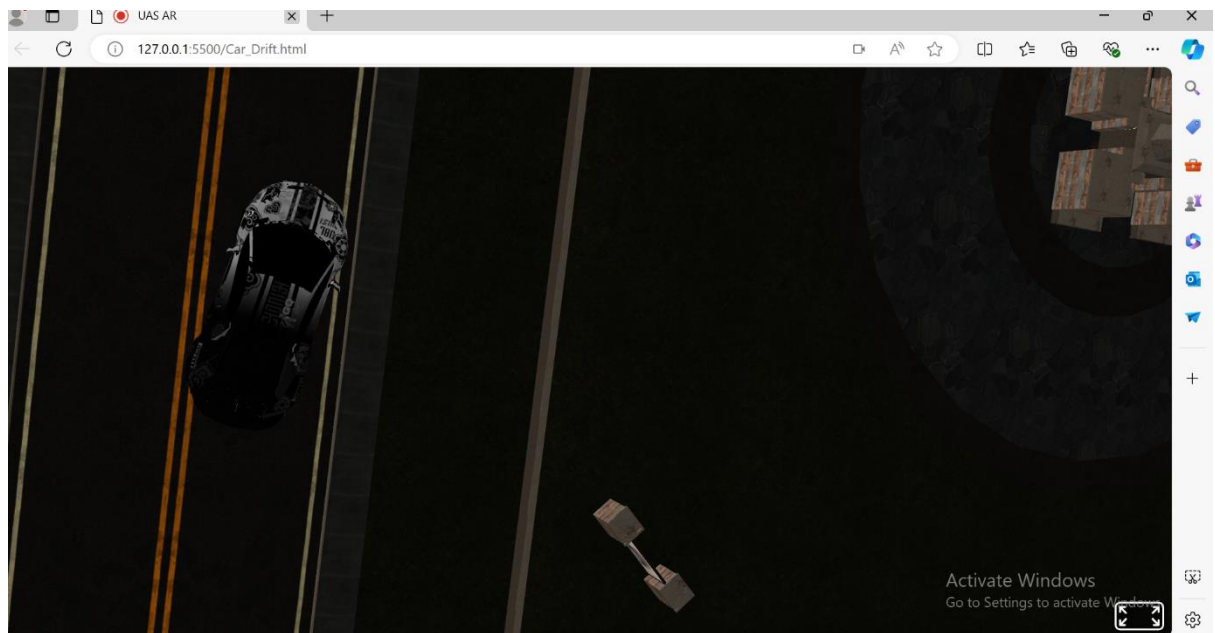
Deklarasikan file 3Dnya di dalam gltf-model



Untuk menjalankannya, klik kanan terus open with Live Server

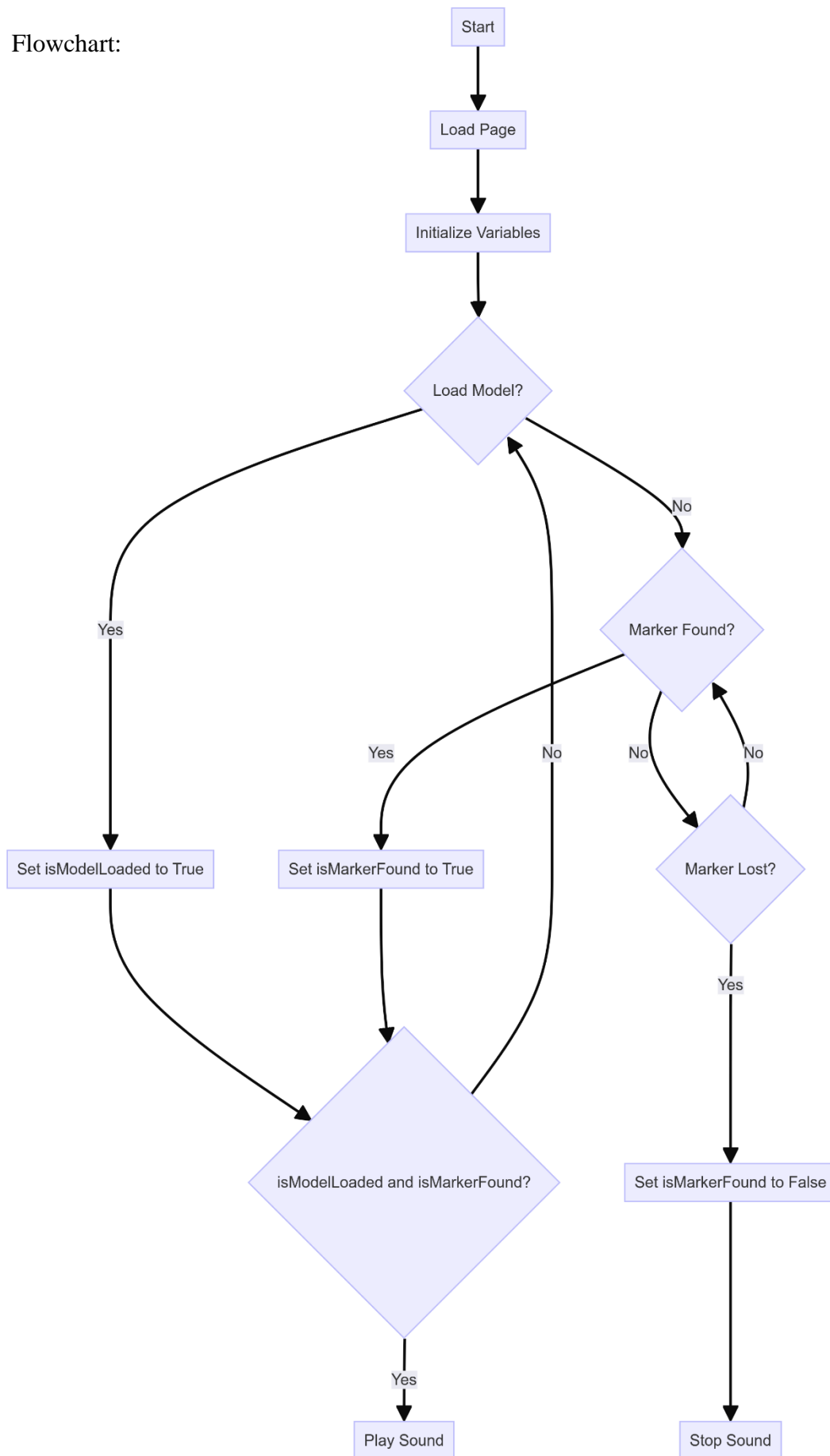


Setelah itu akan otomatis ke dalam servernya, dan di scan marker Hironya



Ini untuk hasil scannya

Flowchart:





Codingan:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>UAS AR </title>
  <script src="https://aframe.io/releases/1.4.0/aframe.min.js"></script>
  <script src="https://raw.githack.com/AR-js-
org/AR.js/master/aframe/build/aframe-ar.js"></script>
  <script src="https://cdn.jsdelivr.net/gh/donmccurdy/aframe-
extras@v6.1.0/dist/aframe-extras.min.js"></script>
</head>
<body style="margin: 0; overflow: hidden;">
  <a-scene embedded arjs>
    <a-marker preset="hiro" id="hiroMarker">
      <a-entity
        id="mobil1"
        gltf-model="url(assets/model/mobil1.glb)"
        scale="0.3 0.3 0.3"
        position="-0.4 0 0"
        rotation="0 180 0"
        animation-mixer
        animation="property: rotation; from: 0 180 0; to: 0 540 0;
dur: 10000; loop: true;">
      </a-entity>
      <a-entity
        id=""
        gltf-model="url(assets/model/road.glb)"
        scale="0.005 0.005 0.005"
        position="0 0 5"
        rotation="0 0 0">
      </a-entity>
    </a-marker>
    <a-entity camera></a-entity>
    <a-entity id="sound" sound="src: url(assets/sounds/driftsound.mp3);
autoplay: false; loop: true;"></a-entity>
  </a-scene>

  <script>
    var marker = document.querySelector('#hiroMarker');
    var model = document.querySelector('#mobil1');
    var sound = document.querySelector('#sound');

    var isModelLoaded = false;
    var isMarkerFound = false;

    model.addEventListener('model-loaded', function () {
```

```

        isModelLoaded = true;
        if (isModelLoaded && isMarkerFound) {
            sound.components.sound.playSound();
        }
    });

    marker.addEventListener('markerFound', function () {
        isMarkerFound = true;
        if (isModelLoaded && isMarkerFound) {
            sound.components.sound.playSound();
        }
    });

    marker.addEventListener('markerLost', function () {
        isMarkerFound = false;
        sound.components.sound.stopSound();
    });
</script>
</body>
</html>

```

#### 4. Hasil Project Web AR.js Car Drift in Road

- Link Download Project
- a. Google Drive : [https://drive.google.com/drive/folders/1-SULb7lqNG-VYgcBHqhii\\_kjxIK\\_83N7?usp=drive\\_link](https://drive.google.com/drive/folders/1-SULb7lqNG-VYgcBHqhii_kjxIK_83N7?usp=drive_link)
- b. Github : <https://github.com/PatriciaMOY/UasAR>
- URL github :  
Jalankan program di link ini :  
[https://github.com/PatriciaMOY/UasAR/blob/main/Car\\_Drift.html](https://github.com/PatriciaMOY/UasAR/blob/main/Car_Drift.html)
- Marker :



Proyek ini adalah contoh aplikasi Augmented Reality (AR) yang dikembangkan menggunakan Visual Studio Code (VSCode) sebagai alat pengembangan utama. Proyek ini menggunakan teknologi Web AR dengan framework seperti A-Frame dan AR.js, serta objek 3D yang diimpor dari Sketchfab. File `example.html` yang diunggah adalah sebuah proyek AR sederhana yang memanfaatkan marker untuk menampilkan model 3D Mobil yang ngedrift di atas jalan raya yang bersuara.

- Platform Yang Digunakan:
  1. Platform Pengembangan: Visual Studio Code (VSCode)
  2. Teknologi: A-Frame, AR.js
- Fitur Utama:
  1. Menampilkan model 3D mobil dan jalanan ketika marker terdeteksi.
  2. Memainkan suara yang berulang saat mobil muncul.
- Struktur Kode:
  1. HTML Struktur:
    - Header: Mengimpor library A-Frame, AR.js, dan A-Frame Extras untuk mendukung fitur AR dan animasi
    - Body: Menyusun scene A-Frame yang berisi marker AR dan model 3D dinosaurus.
  2. A-Frame Scene:
    - a-marker: Menggunakan preset "hiro" untuk mendeteksi marker AR.
    - a-entity: Menampilkan model 3D yang diimpor dari file GLB dengan skala dan posisi tertentu, serta menambahkan animasi.
    - a-sound: Memainkan suara yang bersumber dari file MP3 secara otomatis dan berulang.
- Penjelasan Setiap Bagian:
  1. Header:
    - Mengimpor A-Frame: Framework untuk membuat konten 3D dan AR berbasis web.
    - Mengimpor AR.js: Library untuk menambahkan fitur AR ke A-Frame.
    - Mengimpor A-Frame Extras: Plugin tambahan untuk mendukung animasi dan fitur lainnya.
  2. Body:
    - a-scene: Elemen utama A-Frame yang mengandung seluruh scene AR.
    - a-marker: Elemen yang mendeteksi marker AR dan menampilkan objek 3D saat marker terdeteksi.
    - a-entity: Elemen untuk menampilkan model 3D dinosaurus dengan skala dan posisi yang ditentukan. Juga menambahkan animasi ke model.
    - a-sound: Elemen untuk memainkan suara yang berulang saat objek muncul.
    -

## 5. Screenshot demo / link vidio demo

Link googledrive :

<https://drive.google.com/file/d/1PdgdG2xw55oLg7XFtUCQV2dEtW47P7bN/view?usp=sharing>

## 6. Kesimpulan

Proyek AR ini berhasil menunjukkan bagaimana teknologi AR berbasis web dapat dikembangkan menggunakan VSCode. Dengan menggabungkan A-Frame dan AR.js, proyek ini memungkinkan pembuatan aplikasi AR yang interaktif dan mudah diakses. Selain memberikan pengalaman yang menarik bagi pengguna, proyek ini juga mengungkap potensi besar teknologi AR dalam berbagai aplikasi, termasuk pendidikan dan hiburan. Dengan alat dan teknologi yang tepat, pengembangan aplikasi AR menjadi lebih mudah dan dapat diakses oleh berbagai kalangan pengembang.

Proyek ini menunjukkan penggunaan VSCode untuk mengembangkan aplikasi AR berbasis web. Dengan memanfaatkan A-Frame dan AR.js, proyek ini memungkinkan pengguna untuk melihat dan mendengar dinosaurus 3D ketika marker AR terdeteksi. Penggunaan VSCode dalam menulis dan mengelola kode memberikan lingkungan pengembangan yang efisien, dilengkapi dengan dukungan untuk berbagai ekstensi yang mempermudah proses pengembangan.

## 7. Referensi

- A-Frame Documentation: Dokumentasi resmi A-Frame yang menjelaskan cara membangun konten 3D dan AR menggunakan framework ini.  
[A-Frame Documentation](<https://aframe.io/docs/>)
- AR.js Documentation: Dokumentasi resmi AR.js yang memberikan panduan tentang bagaimana menambahkan fungsionalitas AR ke dalam proyek web.  
AR.js Documentation](<https://ar-js-org.github.io/AR.js-Docs/>)
- Visual Studio Code Documentation: Dokumentasi VSCode yang memberikan panduan tentang fitur-fitur, ekstensi, dan cara penggunaan editor ini untuk pengembangan web.  
[Visual Studio Code Documentation](<https://code.visualstudio.com/docs>)
- Sketchfab: Platform yang menyediakan model 3D, termasuk model dinosaurus yang digunakan dalam proyek.  
[Sketchfab](<https://sketchfab.com>)
- A-Frame Extras: Ekstensi tambahan untuk A-Frame yang mendukung berbagai fitur, termasuk animasi.  
[A-Frame Extras](<https://github.com/donmccurdy/aframe-extras>)
- Uas HTML File: File contoh `example.html` yang diunggah, menunjukkan implementasi proyek AR menggunakan A-Frame dan AR.js.  
[Uas HTML File](67)