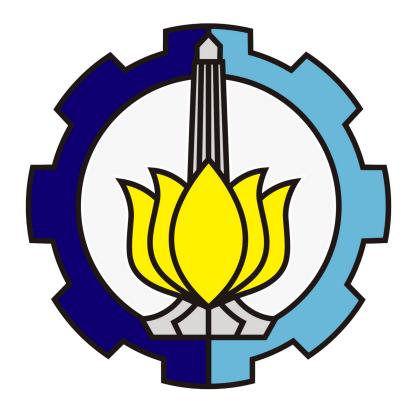
# Laporan ETS WebGL



Disusun Oleh : Aufa Nabil Amiri - 0721 17 4000 0029

Teknik Komputer Fakultas Teknologi Elektro dan Informatika Cerdas Institut Teknologi Sepuluh Nopember

### 1 File HTML

### Setup Canvas

Listing 1: file index.html

```
1 <html lang="en">
   <head>
     <title>ETS</title>
     <meta charset="utf-8" />
   </head>
   <body onload="startup();">
     <canvas id="canvas" width="500" height="500"></canvas>
     <div style="position: absolute; top: 550px; color: black←</pre>
        ; z-index: 10">
       Keyboard:
10
       ul>
11
         W untuk maju
12
         S untuk mundul
         D untuk rotate ke kanan
         A untuk rotate ke kiri
         panah atas untuk ke atas (max. 20)
         <li>panah bawah untuk ke bawah (min.0)</li>
       </div>
```

Melakukan inisiasi *canvas* pada html dengan cara melakukan <canvas ⇔ id="canvas"width="500"height="500"></canvas>. Selain itu, juga dilakukan pemanggilan fungsi startup() pada saat file html sudah terload sepenuhnya.

### Setup Vertex Shader

Listing 2: file vertexShader

```
13 }
14 </script>
```

Nantinya, setiap vertex yang dikirimkan ke vertexShader akan dikalikan dengan uProjectionMatrix yang digunakan untuk memposisikan menentukan bagaimana behaviour "kamera" dalam scene. Selanjutnya, akan mengalami proses perkalian dengan uModelViewMatrix yang berfungsi untuk menentukan lokasi model dalam koordinat global.

### Setup Fragment Shader

Listing 3: file fragmentShader

FragmentShader akan digunakan untuk menentukan warna model yang akan ditampilkan di layar.

### Load Script yang Dibutuhkan

Listing 4: file Load Script

Disini dilakukan beberapa load script - script yang akan digunakan dalam program nantinya. gl-matrix-min.js digunakan untuk mempermudah penghitungan matrix orde 3 dan 4 yang akan digunakan saat melakukan animasi. utils.js berisi beberapa fungsi penting seperti loadWebglContext yang digunakan untuk mendapatkan context webGL sebelum kita dapat menampilkan objek apapun. initShader.js digunakan untuk melakukan compile terhadap vertexShader dan fragmentShader. Dan index.js merupak-

an file utama yang paling penting karena berisi merupakan tempat fungsi startup berada

### 2 File Utils.js

Terdapat beberapa fungsi penting di file utils. js ini,

#### • createGLContext

Berfungsi untuk mendapatkan webGL context yang akan dipakai di seluruh bagian index.js nantinya.

#### • getShaderfromDOM

Mendapatkan ShaderSource baik itu adalah vertexShader maupun fragmentShader dari file index.html yang sudah dibuat sebelumnya.

#### • createSphere

Menghitung vertex yang akan digunakan untuk membentuk suatu model sphere.

Listing 5: fungsi createSphere

```
1 function createSphere(div, color) {
    var positions = [];
    for (var i = 0; i <= div; ++i) {</pre>
      var ai = (i * Math.PI) / div;
      var si = Math.sin(ai);
      var ci = Math.cos(ai);
      for (var j = 0; j <= div; ++j) {</pre>
         var aj = (j * 2 * Math.PI) / div;
        var sj = Math.sin(aj);
9
        var cj = Math.cos(aj);
10
        positions = positions.concat([si * sj, ci, si * \leftarrow
            cj]);
      }
12
    }
13
14
    var indices = [];
15
    for (var i = 0; i < div; ++i) {</pre>
16
      for (var j = 0; j < div; ++j) {
17
         var p1 = i * (div + 1) + j;
18
         var p2 = p1 + (div + 1);
19
         indices = indices.concat([p1, p2, p1 + 1, p1 + 1,\leftarrow
20
             p2, p2 + 1]);
21
    }
22
```

```
23
    var colors = [];
^{24}
    for (var i = 0; i != indices.length; i++) {
      colors = colors.concat(color);
26
27
28
    return {
      vertexData: positions,
30
      indices: indices,
31
      colors: colors,
32
    };
33
34 }
```

## 3 File initShader.js