

Tugas 1
2D Web Based CAD (Computer-Aided-Design)
IF3260 Grafika Komputer



Dibuat oleh Kelompok G06 K03:

Tony Eko Yuwono	13518030
Daffa Pratama Putra	13518033
Hengky Surya Angkasa	13518048

Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2021

I. Deskripsi Tugas

Pada tugas 1 ini, kami diminta untuk membuat model poligon untuk kebutuhan arsitektur menggunakan kakas WebGL. WebGL adalah sebuah teknologi web, yang menggunakan Javascript API, yang dapat membuat grafika 3D ke dalam web tanpa memerlukan *plug-in* tambahan. *Website* yang dibuat dapat menampilkan gambar model berupa garis, persegi, serta poligon. Pengguna dapat menyimpan model yang sudah dihasilkan dalam sebuah *file* untuk nantinya dapat dibuka kembali. *File* tersebut berisikan keterangan model, berupa daftar koordinat dan warna untuk setiap poligon.

Fitur dasar yang dapat dilakukan pada aplikasi yang dibuat adalah menggambar garis, persegi, dan poligon sederhana. Terdapat juga beberapa fitur interaksi yang disediakan oleh *website* ini, yakni pengguna dapat menggeser titik kontrol dengan *mouse*, mengubah panjang garis, mengubah ukuran sisi persegi, mengubah warna poligon dengan input bebas, serta menu *help* yang memudahkan pengguna baru untuk memahami penggunaan dari *website* tersebut. Aplikasi juga dapat menyimpan daftar koordinat dan warna dari poligon yang dibuat dalam bentuk *text file* dan dapat membaca kembali *text file* tersebut untuk mendapatkan atribut pembentuk poligon.

II. Fungsionalitas Program

Fungsionalitas program pada aplikasi yang dibuat dibagi ke dalam beberapa bagian berdasarkan tugas dan kegunaannya. Fungsionalitas tersebut adalah sebagai berikut:

1. GLObjects

Pada aplikasi ini, kami membentuk kelas *glObjects* untuk menampung objek-objek yang digambar pada canvas. Secara umum, fungsi dari kelas *glObjects* ini dapat menambahkan objek yang ingin digambar ke canvas dan mengupdate posisi titik kontrol sesuai *cursor* user. Fungsionalitas kelas *glObjects* dapat dilihat pada file “*index.js*”. Berikut adalah method yang tersedia pada kelas *glObjects*.

Method / Property	Deskripsi
constructor	Konstruktor pembentuk <i>glObject</i> menginisiasi array penampungan objek-objek yang akan digambar
push	Menambahkan objek yang digambar ke dalam array penampungan

updateSelectedObject	Memperbarui objek yang dipilih pada canvas melalui mouse click
updateSelectedObject Color	Memperbarui warna dari objek yang dipilih
resizeSelectedObject	Mengubah ukuran dari objek persegi/poligon yang dipilih
clearObjects	Menghapus semua objek yang telah disimpan
update	Memperbarui objek-objek yang digambar pada canvas dengan melakukan render terhadap semua objek
save	Menyimpan semua objek pada glObject ke dalam sebuah JSON
renderAll	Membersihkan canvas dan melakukan render terhadap semua objek yang akan digambar

2. Line

Objek garis direpresentasikan dalam bentuk kelas Line. Kelas ini dapat menerima input dari pengguna dan menggambarannya ke canvas. Fungsionalitas kelas Line dapat dilihat pada file “Line.js”. Berikut adalah method yang terdapat pada kelas Line.

Method / Property	Deskripsi
constructor	Konstruktor pembentuk Line dengan menerima dua titik masukan pembentuk garis dan warna garis
isCoordinateInside	Mengecek apakah koordinat yang diklik pada canvas merupakan titik yang terletak di dekat simpul-simpul pada garis terkait (menggunakan <i>margin of error</i> 0.05)
getClosestPoint	Mendapatkan titik simpul dari garis terkait yang paling dekat dengan <i>control point</i> user
toJson	Mengembalikan atribut pembentuk garis ke dalam format JSON
render	Melakukan rendering atau menggambar garis ke canvas

3. Point

Objek titik direpresentasikan dalam bentuk kelas Point. Kelas ini menerima koordinat titik (x,y) dan menggambarannya ke canvas. Fungsionalitasnya dapat dilihat pada file “Point.js”. Berikut adalah method yang terdapat pada kelas Point.

Method / Property	Deskripsi
constructor	Konstruktor pembentuk Point dengan menerima masukan titik (x,y) dan warna
move	Menggeser koordinat titik (x,y) sehingga didapat koordinat titik yang baru
render	Melakukan rendering atau menggambar titik ke canvas

4. Square

Objek persegi direpresentasikan dalam bentuk kelas Square. Kelas dapat menggambar persegi dengan ukuran default-nya adalah 1 dan dengan warna yang dapat dipilih dengan color picker. Fungsionalitasnya dapat dilihat pada file “Square.js”. Berikut adalah method yang terdapat pada kelas Square.

Method / Property	Deskripsi
constructor	Konstruktor pembentuk persegi dengan menerima masukan berupa titik tengah, ukuran, dan warna persegi
calculateEdgesPosition	Menghitung posisi setiap sisi (kiri, kanan, atas, bawah) dari Square sesuai dengan center point dan ukuran persegi
isCoordinateInside	Mengecek apakah koordinat canvas yang diklik merupakan titik di dalam persegi
resize	Mengubah ukuran persegi berdasarkan masukan dari pengguna
move	Memindahkan persegi ke koordinat lain
toJson	Mengembalikan atribut constructor dari objek dalam format JSON
render	Melakukan rendering atau menggambar persegi ke canvas

5. Hexagon

Objek segi 6 atau hexagon direpresentasikan ke dalam bentuk kelas Hexagon. Kelas ini dapat menggambar objek hexagon ke atas canvas dengan warna yang dapat dikustomisasi. Fungsionalitas dari kelas hexagon dapat dilihat pada file “Polygon.js”. Berikut adalah method yang terdapat pada kelas Hexagon.

Method / Property	Deskripsi
constructor	Konstruktor pembentuk hexagon dengan menerima masukan titik tengah hexagon beserta warnanya
calculateVertices	Memproses nilai tiap vertex dari hexagon berdasarkan centerPoint
isCoordinateInside	Mengecek apakah titik yang diklik pada canvas terletak di dalam hexagon
move	Menggeser objek hexagon ke koordinat lain
toJson	Mengembalikan atribut constructor dari objek dalam format JSON
render	Melakukan rendering atau menggambar hexagon ke canvas

6. Octagon

Objek segi 8 atau octagon direpresentasikan ke dalam bentuk kelas Octagon. Kelas ini dapat menggambar objek octagon ke atas canvas dengan warna yang dapat dikustomisasi. Secara umum, kelas ini sama seperti kelas Hexagon, namun terdapat 8 titik pembentuknya objeknya. Fungsionalitas dari kelas octagon dapat dilihat pada file “Polygon.js”. Berikut adalah method yang terdapat pada kelas Octagon.

Method / Property	Deskripsi
constructor	Konstruktor pembentuk octagon dengan menerima masukan titik tengah hexagon beserta warnanya
calculateVertices	Memproses nilai tiap vertex dari octagon berdasarkan centerPoint
isCoordinateInside	Mengecek apakah titik yang diklik pada canvas terletak di dalam octagon
move	Menggeser objek octagon ke koordinat lain
toJson	Mengembalikan atribut constructor dari objek dalam format JSON
render	Melakukan rendering atau menggambar octagon ke canvas

7. Polygon

Objek segi banyak atau polygon dalam aplikasi ini direpresentasikan menggunakan kelas Polygon. Kelas ini dapat menggambar segi banyak menggunakan interaksi mouse click pada

canvas. Fungsionalitas dari kelas Polygon dapat dilihat pada file “Polygon.js”. Berikut adalah method yang terdapat pada kelas Polygon

Method / Property	Deskripsi
constructor	Konstruktor pembentuk polygon menerima masukan berupa titik-titik pembentuk polygon beserta warnanya
getCenterPoint	Mendapatkan titik tengah polygon
calculateVertices	Memproses nilai tiap vertex dari polygon berdasarkan centerPoint
convertInputToVertices	Mengubah input klik vertex2 menjadi sesuai dengan format vertex3 webgl pada saat create polygon
isCoordinateInside	Mengecek apakah titik yang diklik pada canvas terletak di dalam polygon
isInsideTriangle	Mengecek sebuah titik berada didalam polygon dengan 3 titik / segitiga
area	Menghitung luas dari polygon
move	Menggeser objek polygon ke koordinat lain
toJson	Mengembalikan atribut constructor dari objek dalam format JSON
render	Melakukan rendering atau menggambar polygon ke canvas

8. Utility

Utility merupakan fungsi tambahan yang digunakan dalam menggambar objek ke dalam canvas. Fungsionalitas utility dapat dilihat pada file “utils.js”. Berikut adalah method yang terdapat pada utility.

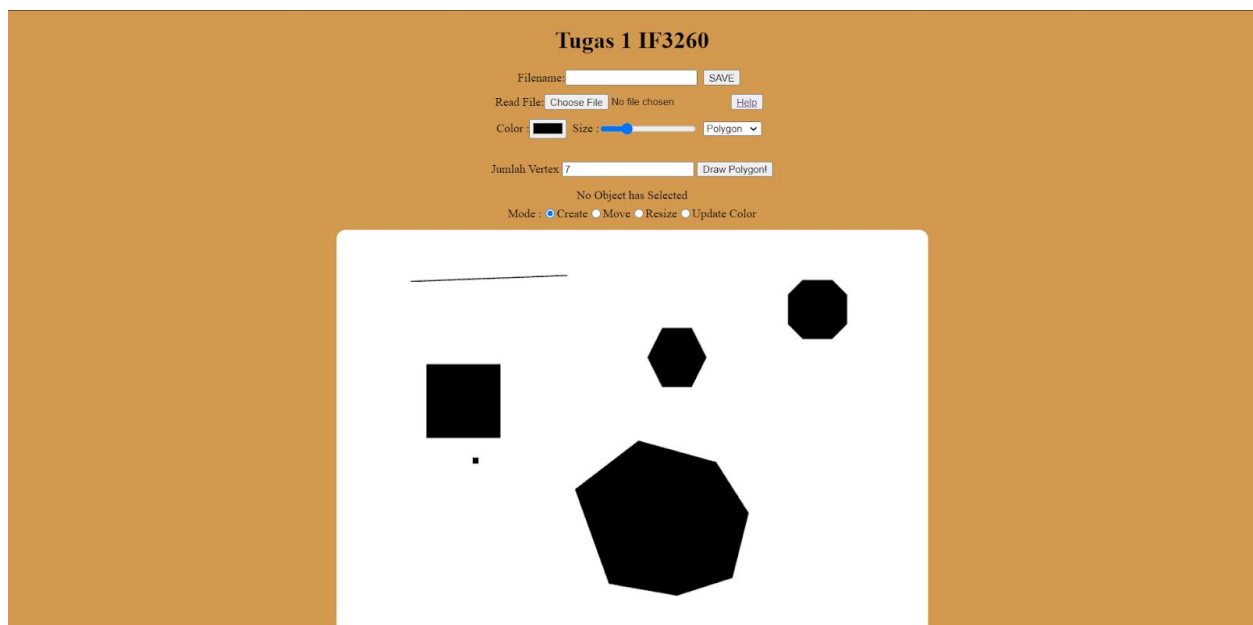
Method / Property	Deskripsi
createVertexBuffer	Membuat vertex buffer dari vertices pembentuk objek
createIndexBuffer	Membuat index buffer
createVertexShader	Membuat vertex shader
createFragmentShader	Membuat fragment shader dari warna objek
createShaderProgram	Membuat shader program dari vertex shader dan fragment shader

associateShaderToObjBuffer	Load vertex shader (koordinatnya) ke vertex buffer
clearCanvas	Membersihkan canvas dengan menghapus semua objek yang telah digambar
hexToRgb	Mengkonversi nilai hexadecimal menjadi nilai RGB
euclideanDistance	Mencari jarak antara dua titik menggunakan perhitungan euclidean distance

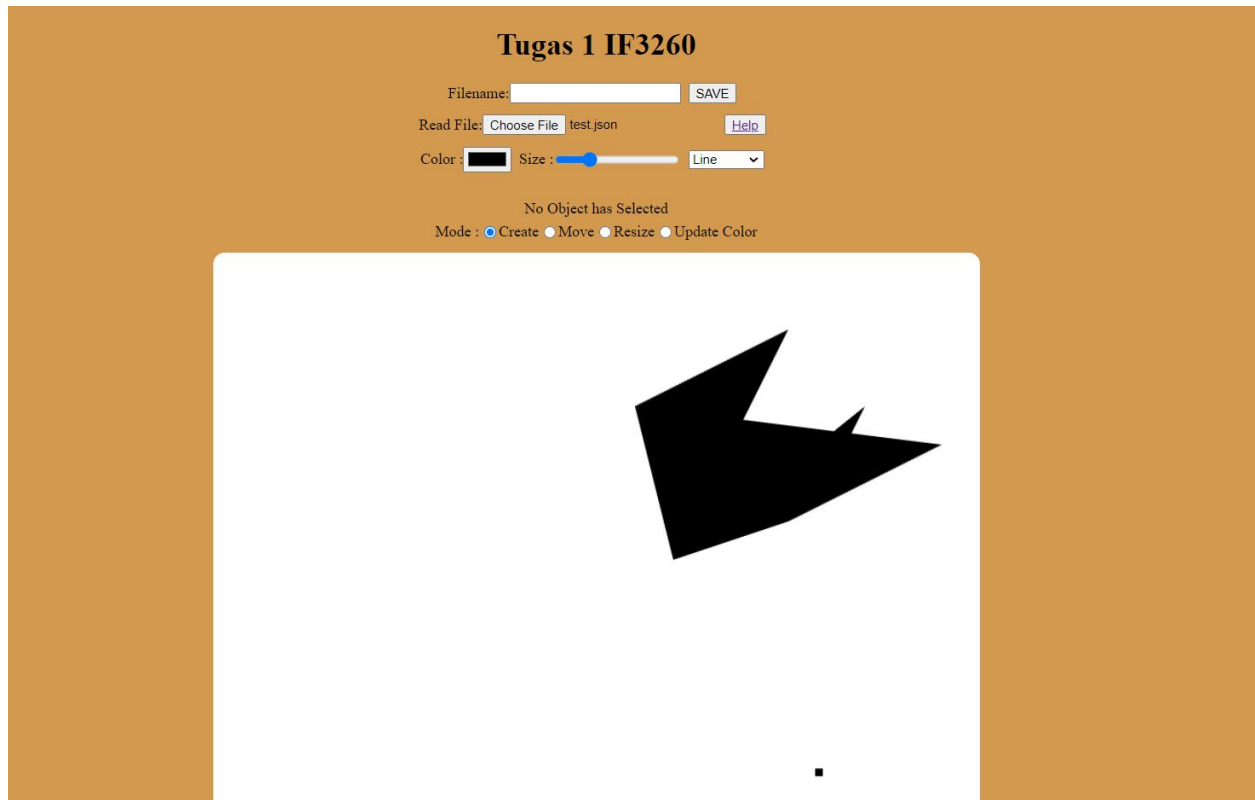
III. Hasil

Berikut adalah screenshot dari aplikasi yang kami buat.

1. Menggambar objek

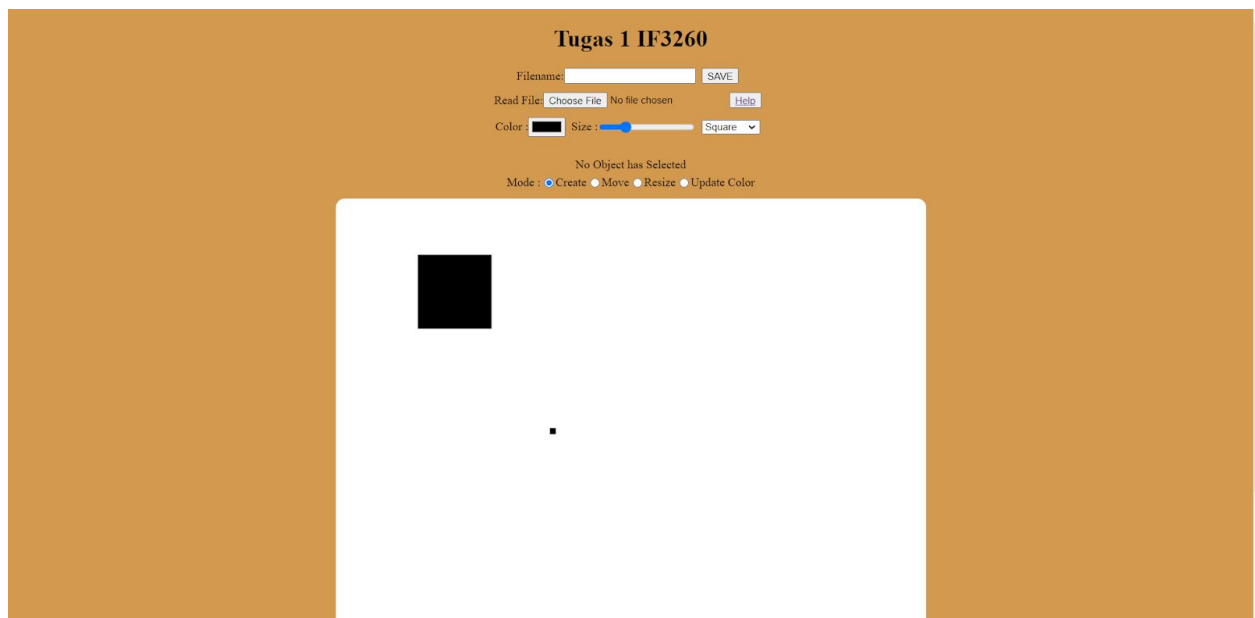


2. Menggambar objek menggunakan file eksternal

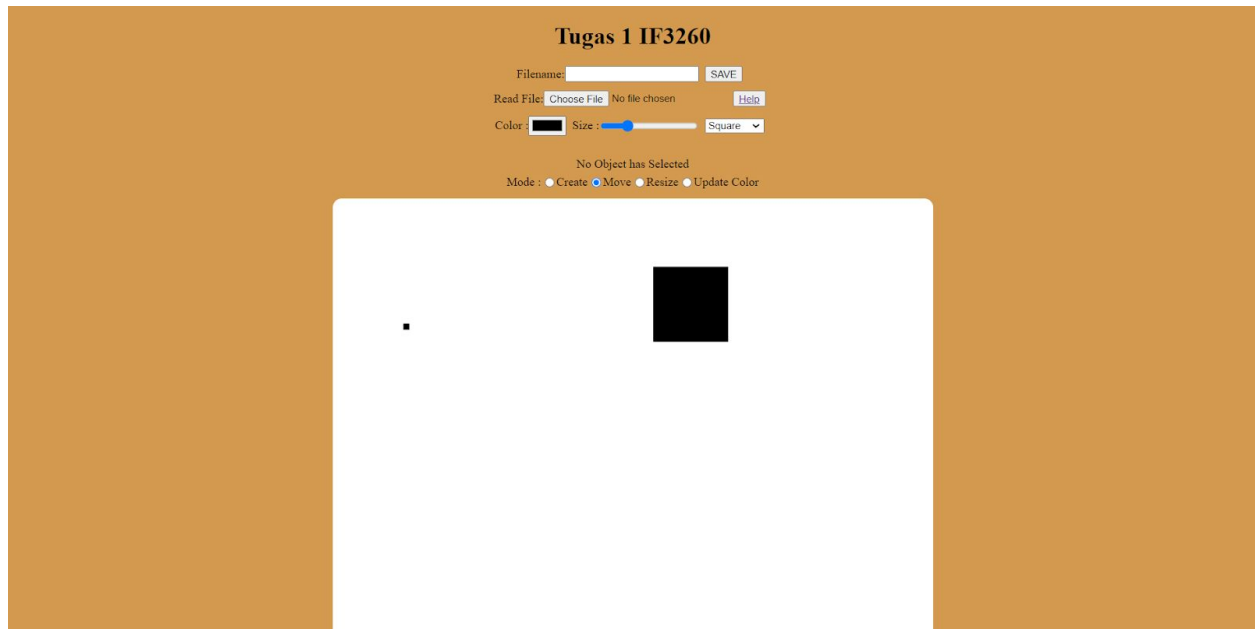


3. Memindahkan objek

- **Sebelum dipindah**

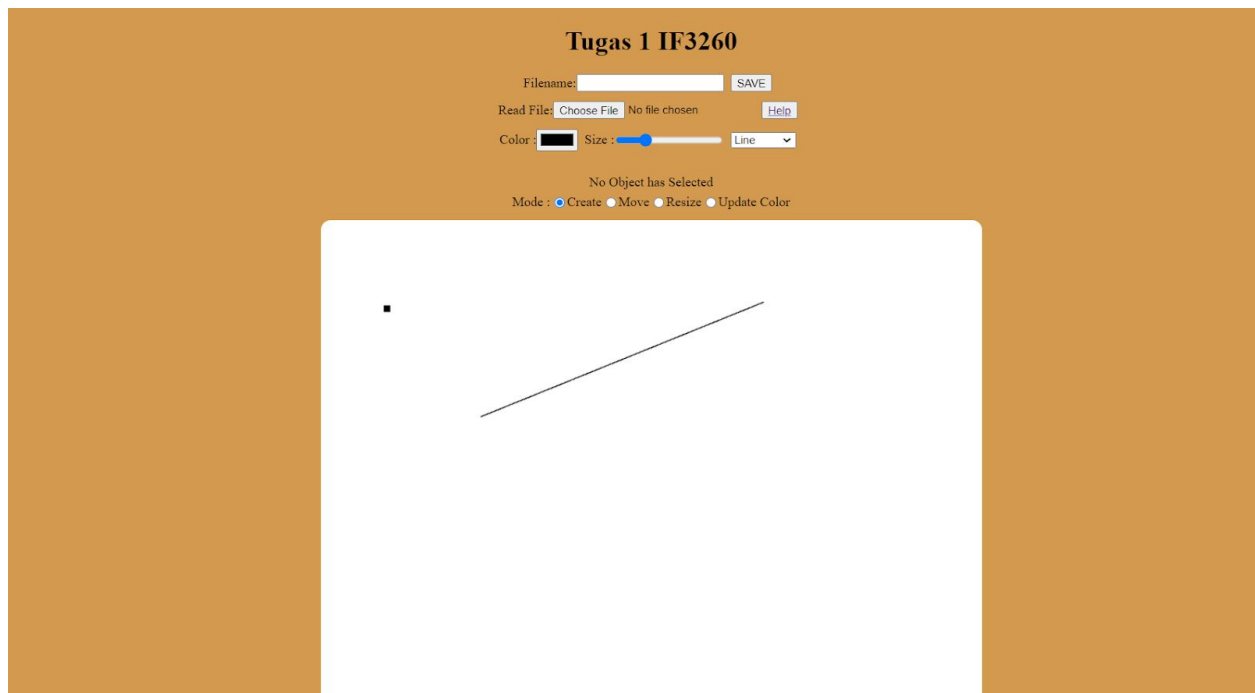


- **Sesudah dipindah**

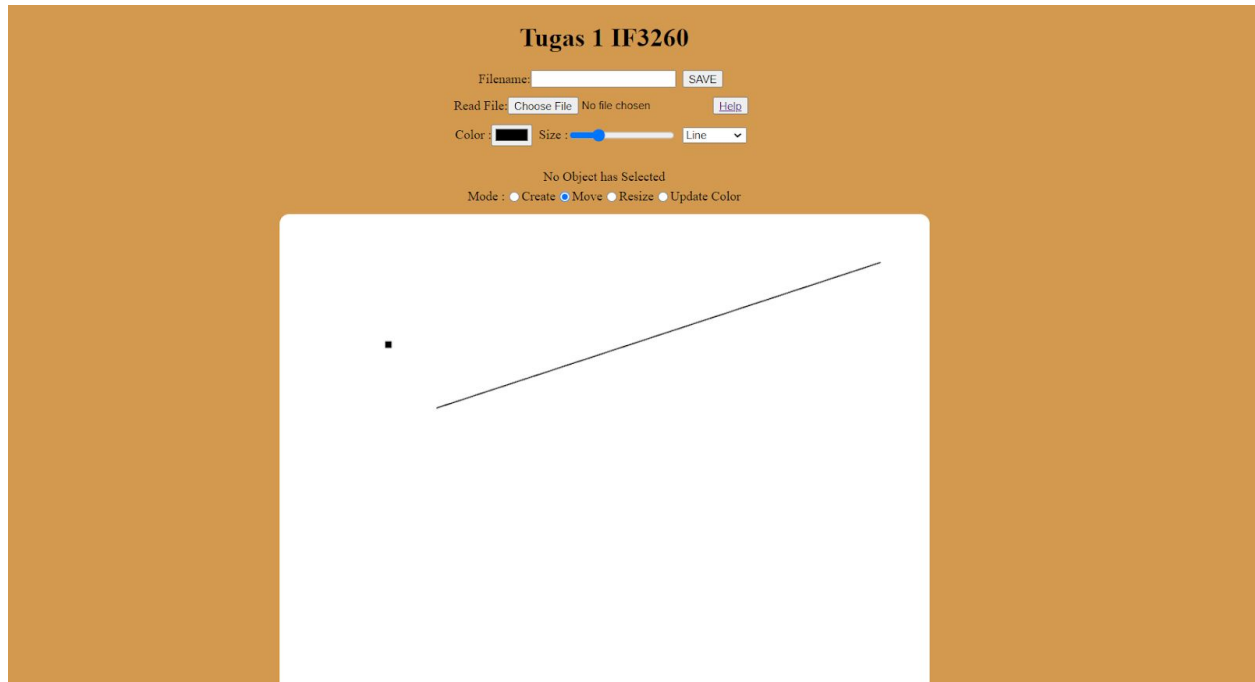


4. Mengubah panjang garis

- **Sebelum diubah**

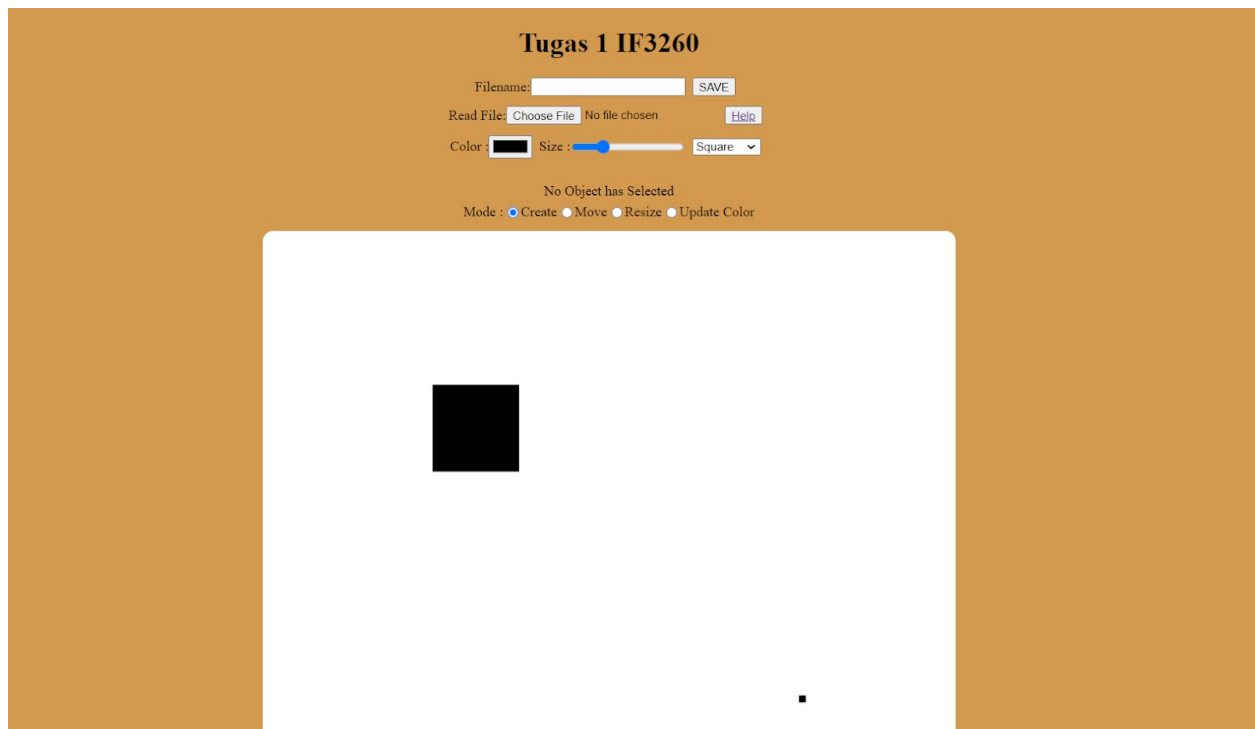


- **Setelah diubah**

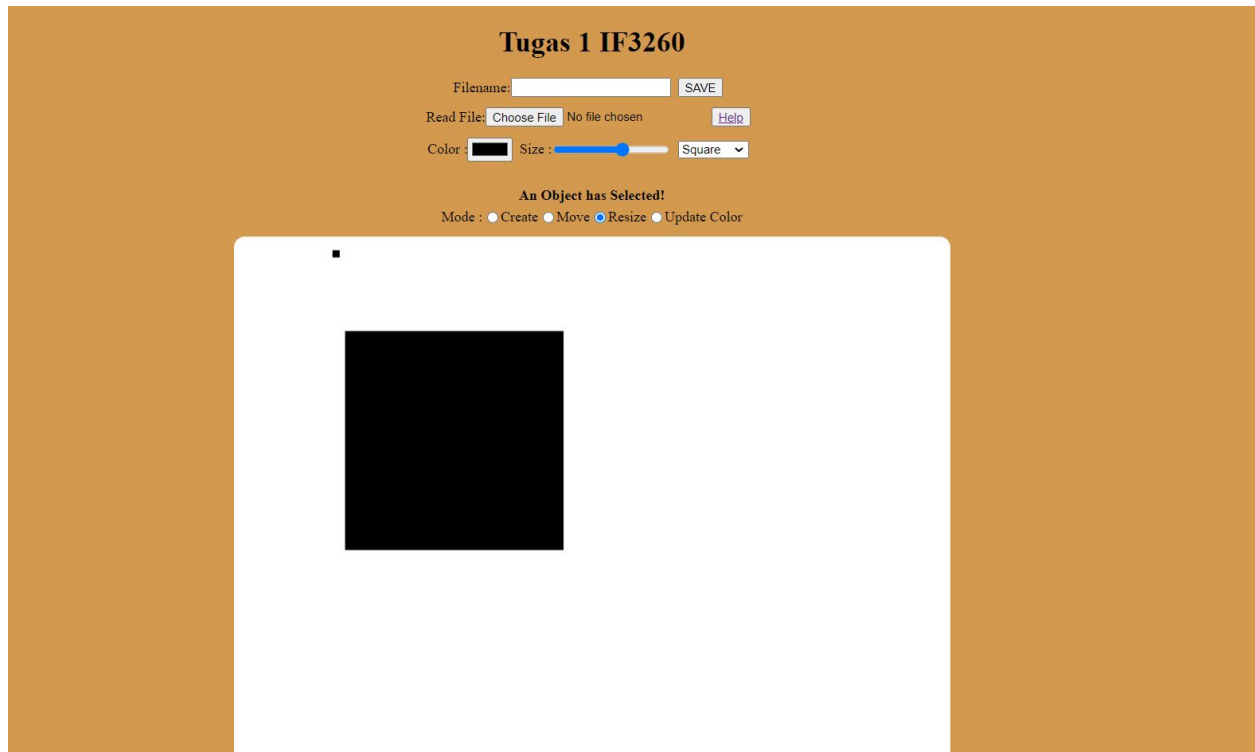


5. Mengubah ukuran persegi

- **Sebelum diubah**

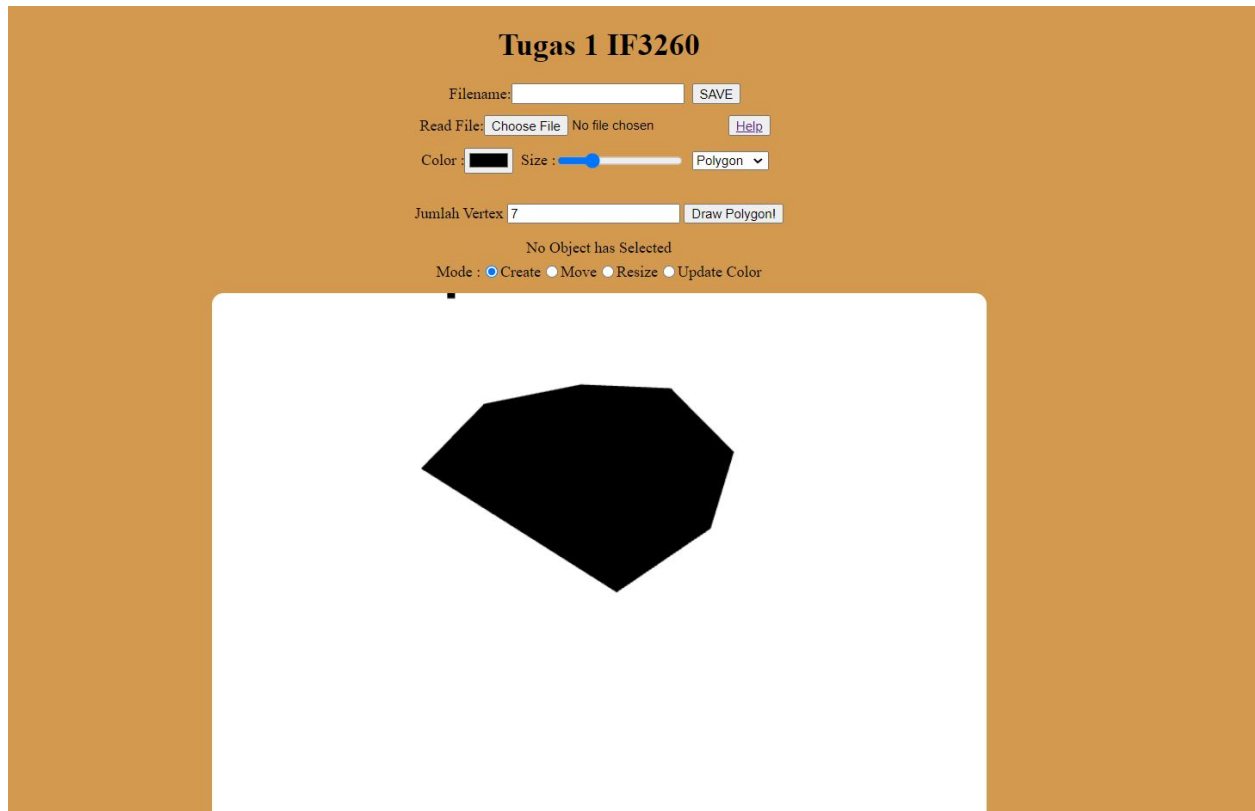


- **Setelah diubah**

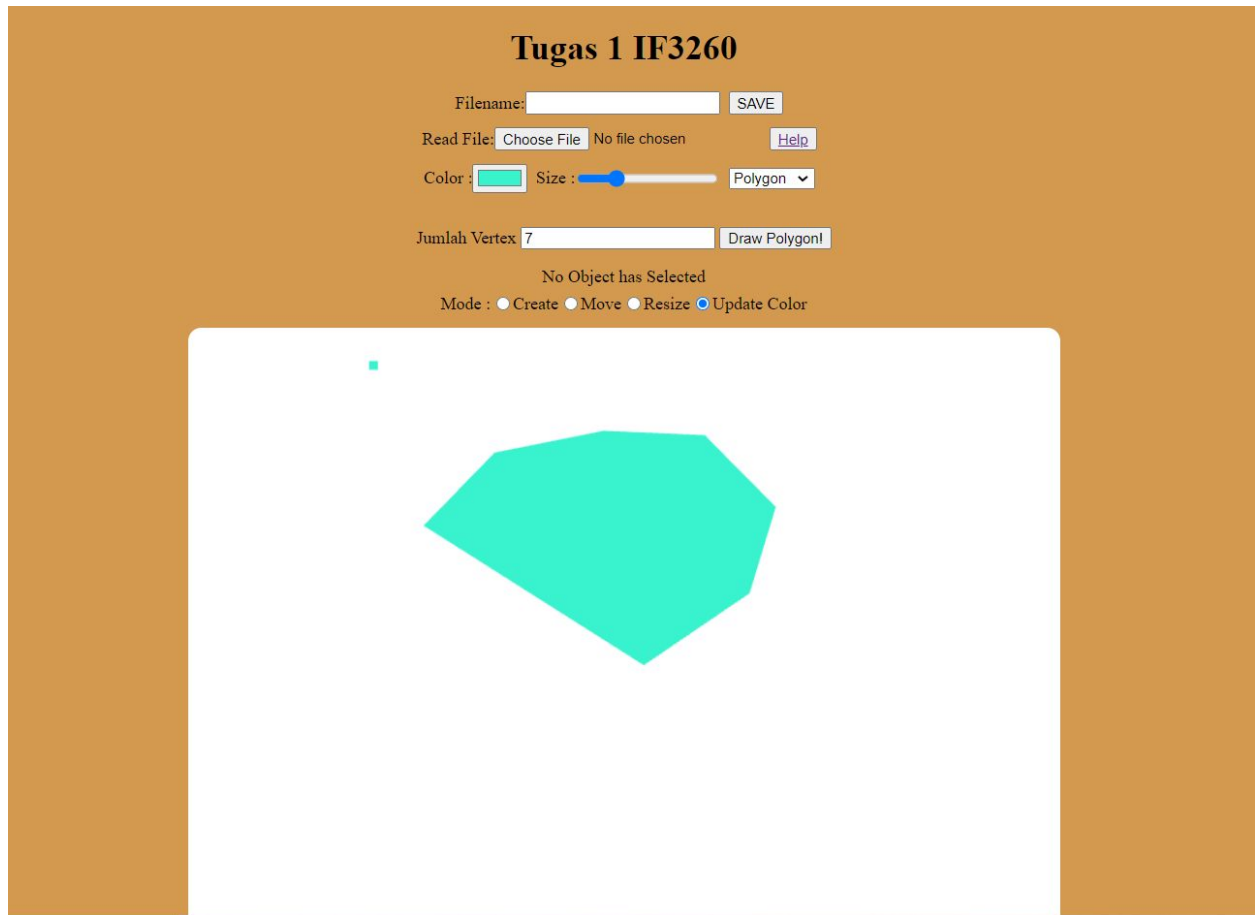


6. Mengubah warna objek

- **Sebelum diubah**

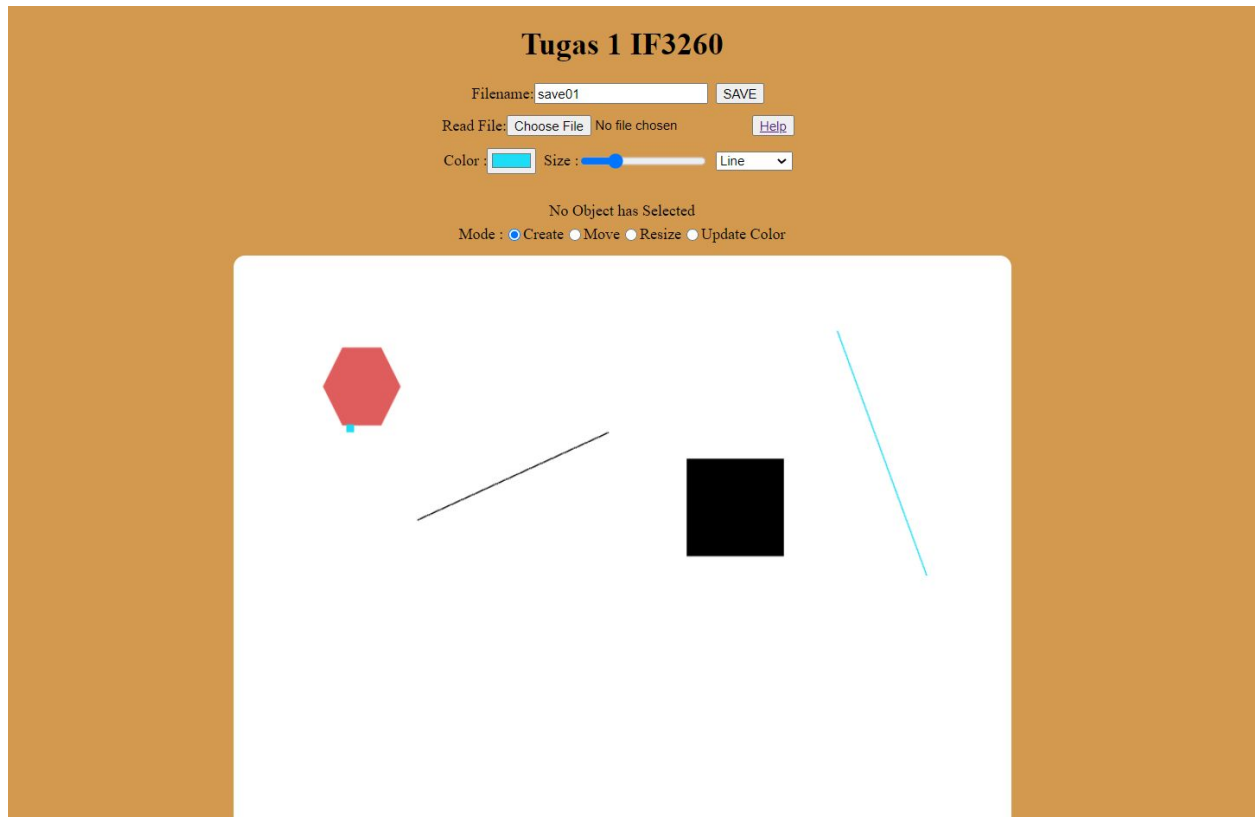


- Setelah diubah



7. Menyimpan file model

- **Objek yang disimpan**



- Hasil penyimpanan



8. Menu help

Menu Help

[Back To Home](#)

How to Draw

1. Pilih mode **Create**
2. Pilih model objek
3. Pilih warna dari objek
4. Gambar!
 - **Line** : klik 2 (dua) titik pada canvas
 - **Square, Hexagon, Octagon** : pilih **ukuran** objek dan klik pada canvas
 - **Polygon** : pilih banyaknya sisi polygon dan klik titik pembentuk polygon pada canvas

How to Move Square, Octagon, Hexagon, Polygon

1. Pilih mode **Move**
2. Klik pada objek yang akan dipindah
3. Jika muncul tulisan *An object has selected*, berarti objek telah dipilih
4. Klik pada titik lain pada canvas untuk memindahkan

How to Resize/Move Line

1. Pilih mode **Move**
2. Klik pada salah satu titik pada garis
3. Klik pada titik lain untuk mengubah panjang garis

How to Resize Square

1. Pilih mode **Resize**
2. Klik pada objek yang akan diubah ukurannya
3. Jika muncul tulisan *An object has selected*, berarti objek telah dipilih
4. Ganti ukuran objek menggunakan slider Size
5. Klik canvas untuk melakukan *unselect* objek

How to Update Color

1. Pilih mode **Update Color**
2. Pilih warna menggunakan color picker
3. Klik pada objek yang akan diubah warnanya

IV. Kontribusi Individual

NIM	Nama	Kontribusi
13518030	Tony Eko Yuwono	<ul style="list-style-type: none">• Membuat kelas <code>glObjects</code>, <code>Point</code>, <code>Square</code>• Membuat fitur <i>create</i>, <i>move</i>, <i>resize</i>, dan <i>change color</i> pada <code>Square</code>• Menyempurnakan fitur <i>create</i> dan <i>move</i> <code>Line</code> menggunakan <code>mouse onclick</code>• Membuat fitur <code>save</code> ke file eksternal• Membantu laporan
13518033	Daffa Pratama Putra	<ul style="list-style-type: none">• Membuat kelas <code>Line</code>• Membuat fitur menggambar garis• Membuat fitur menerima input dari file eksternal (<code>json</code>)• Membuat menu <code>help</code>• Membuat laporan
13518048	Hengky Surya Angkasa	<ul style="list-style-type: none">• Membuat kelas <code>Polygon</code>, <code>Hexagon</code>, dan <code>Octagon</code>• Membuat fitur menggambar (<i>create</i>) dan <i>move</i> <code>Polygon</code>, <code>Hexagon</code>, <code>Polygon</code> pada <code>canvas</code>• Membuat fitur <i>dropdown list</i> untuk memilih objek yang ingin digambar• Membantu laporan