

LAPORAN TUGAS BESAR I
IF3260 GRAFIKA KOMPUTER



DISUSUN OLEH
GEOMETRYBOIS

MUHAMMAD FARREL D. R	13521048
MUHAMMAD FADHIL AMRI	13521066
AKMAL MAHARDIKA N. P	13521070

PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG

2024

DAFTAR ISI

DAFTAR ISI	2
BAB I	3
DESKRIPSI	3
1.1 Deskripsi Persoalan	3
1.2 Deskripsi Program	3
BAB II	6
HASIL	6
2.1 Model Garis	6
2.2 Model Persegi	6
2.3 Model Persegi Panjang	7
2.4 Model Polygon	7
2.5 Model Gabungan	8
BAB III	9
MANUAL	9
3.1 Model Garis	9
3.2 Model Persegi	10
3.3 Model Persegi Panjang	11
3.4 Model Polygon	13
3.5 Save & Load	16

BAB I

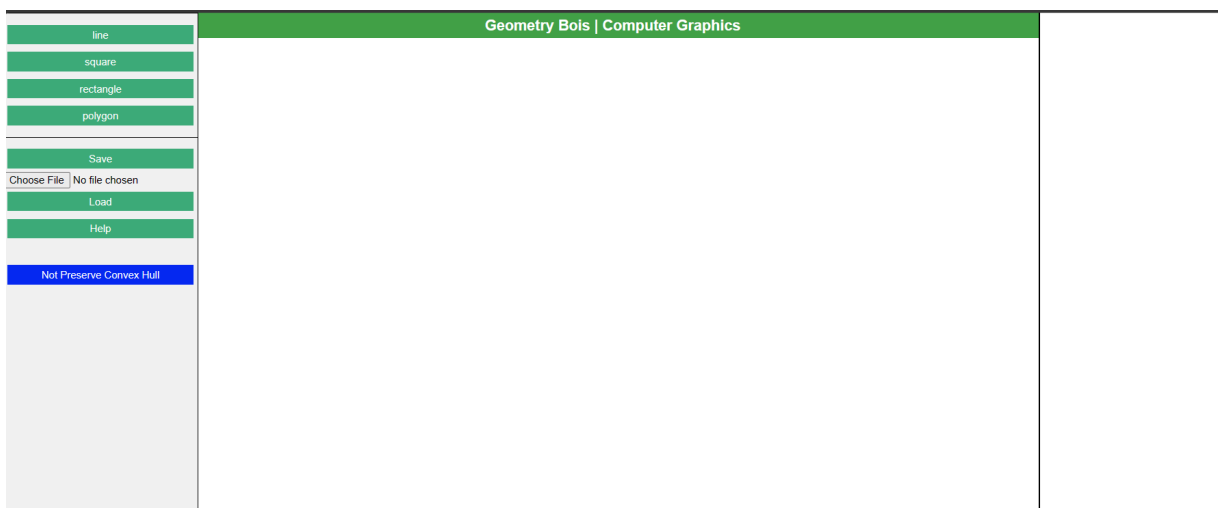
DESKRIPSI

1.1 Deskripsi Persoalan

Tugas besar ini akan menjadi pengenalan terhadap indahnya dunia grafika komputer dan melihatkan secara konkrit bagaimana permainan video tiga dimensi yang sering kamu mainkan itu di-render dalam layar monitor dua dimensi. Target platform yang akan digunakan adalah WebGL. Berikut adalah cakupan materi yang akan digunakan pada tugas besar ini:

- WebGL 2D
- Line, Polygon
- Transformations (translation, rotation, dilation)
- Geometry
- Input and Output

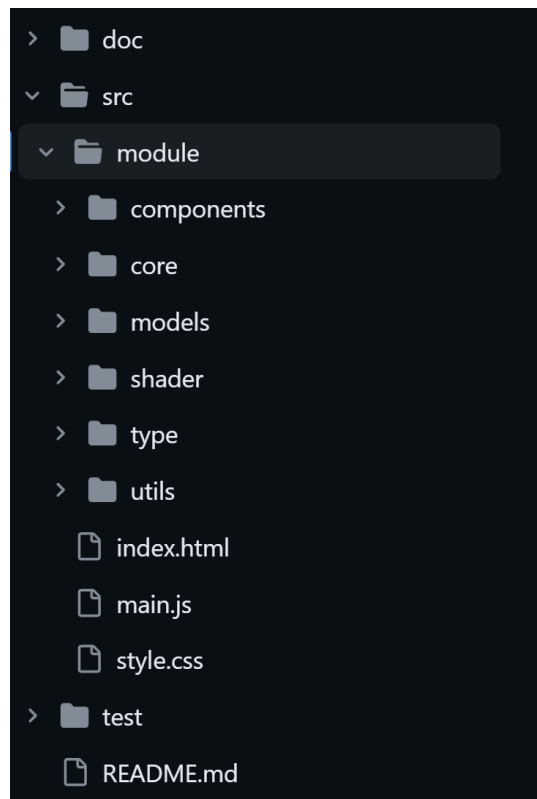
1.2 Deskripsi Program



Gambar 1.2.1. Tampilan Platform WebGL

Berdasarkan tampilan platform WebGL di atas, pengguna mampu menggambar model bentuk yang diinginkan di kanvas yang terletak di tengah *website*. Di panel kiri, terdapat berbagai tombol yang dapat ditekan untuk menggambar model berdasarkan bentuknya, baik itu garis, persegi, persegi panjang, maupun poligon. Pengguna mampu menyimpan model yang telah dibentuk dalam file .json dengan menekan tombol save. Pengguna juga dapat me-load file .json berisi data model dengan menekan tombol load. Pengguna dapat memilih dalam mampu mempertahankan *convex hull* atau tidak dengan menekan toggle “Preserve Convex Hull”. Di panel sebelah kanan, terdapat beberapa input tersembunyi yang muncul saat suatu bentuk diklik. Bentuk dapat ditranslasi, dirotasi, dan didilatasi dengan input-input ini. Penjelasan selengkapnya di bab III.

Berikut adalah direktori program kami (<https://github.com/GAIB20/tugas-besar-grafkom-1-geometrybois/tree/main/src/module>):



Gambar 1.2.2. Direktori Program WebGL

Berikut adalah penjelasan file-file yang berada dalam setiap folder dalam “module” beserta fungsinya.

Tabel 1.2.1 Penjelasan Program WebGL

Folder	Nama File	Fungsi
components	clickedShapeInfo.js	Menyimpan informasi mengenai bentuk yang diklik pengguna di kanvas. Atribut kelas ini menyimpan informasi drawer, bentuk yang diedit, indeks verteks yang diklik, warna, jumlah maksimum verteks, dan banyak verteks yang digambar. Kelas ini mengendalikan aksi input di panel kanan yang berkaitan dengan bentuk yang diklik.
	drawingInfo.js	Menyimpan informasi bentuk yang digambar. Mirip clickedShapeInfo, namun hanya sebatas menampilkan informasi bentuk tersebut agar dapat ditampilkan di panel kanan.
core	GLDrawing.js	Mengendalikan penggambaran bentuk secara dasar. File ini memanggil metode useProgram dan melakukan setupBuffer dalam segi posisi, warna, dan matrix uniform.
	drawer.js	Mengendalikan berbagai kasus penggambaran bentuk, baik itu menggambar semua bentuk, titik-titik sudut, maupun menentukan bentuk apa yang akan digambar.
models	node2.js	Berperan sebagai identifikasi dasar model berupa id dan nama.

	shape.js	Kelas bentuk paling dasar yang meng-extend Node2. Mengandung berbagai informasi seperti posisi, skala, rotasi, shear, matriks transformasi, sudut, dan titik sudut.
	garis.js	Meng-extend Shape, untuk membentuk garis
	persegi.js	Meng-extend Shape, untuk membentuk persegi
	persegiPanjang.js	Meng-extend Shape, untuk membentuk persegi panjang
	polygon.js	Meng-extend Shape, untuk membentuk poligon. Kelas ini memiliki fungsi khusus Convex Hull
shader	fragmentShader.js	Membuat fragment shader untuk memberikan warna
	vertexShader.js	Membuat vertex shader untuk memberikan koordinat clip space
type	bufferType.js	Menyimpan tipe buffer (posisi, warna) dalam bentuk enumerasi
	shapeType.js	Menyimpan tipe bentuk dalam bentuk <i>dictionary</i>
utils	CanvasResize.js	Menyesuaikan ukuran kanvas berdasarkan isi box-nya.
	color.js	Mengatur konversi warna dalam bentuk Hex dan RGB.
	convexhull.js	Mengatur convex hull dari penggambaran poligon
	coordinate2D.js	Menyimpan informasi absis dan ordinat
	GLSLProgram.js	Merepresentasikan program GLSL.
	point.js	Meng-extend kelas Coordinate2D, namun juga menyimpan informasi warna.
	transform2.js	Menyimpan segala hal yang berkaitan dengan matriks transformasi
	vector2.js	Merepresentasikan vektor.

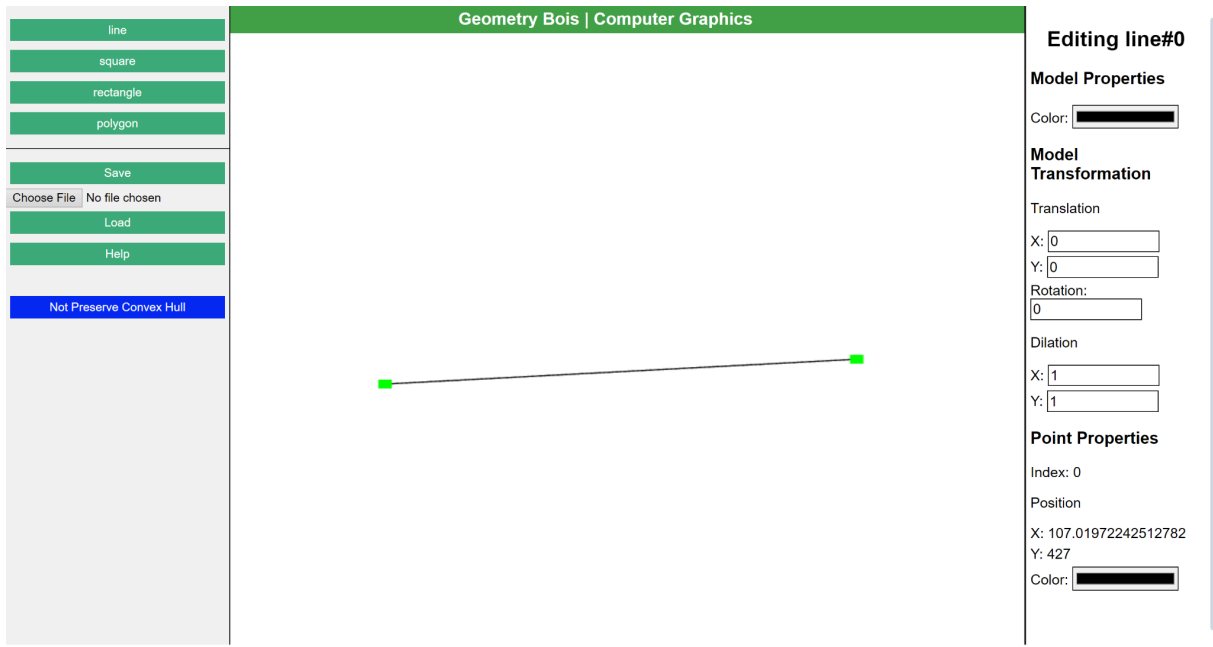
Di luar itu, terdapat file main.js yang mengendalikan interaksi dari pengguna secara langsung dari layar (dengan event listener dan aksi mouse). Kami menggunakan *state-based action*, di mana penggambaran dikategorikan ke dalam lima state:

- IDLE: Pengguna tidak mengklik kanvas, atau pengguna mengklik bagian kanvas yang kosong.
- EDIT: Pengguna mengklik salah satu bentuk di kanvas
- DRAW: Pengguna mengklik tombol opsi Shape di panel kiri
- ONDRAWING: Pengguna menggambar Shape baru
- ONEDITING: Pengguna mengklik verteks dan memodifikasi verteks di salah satu bentuk di kanvas.

BAB II

HASIL

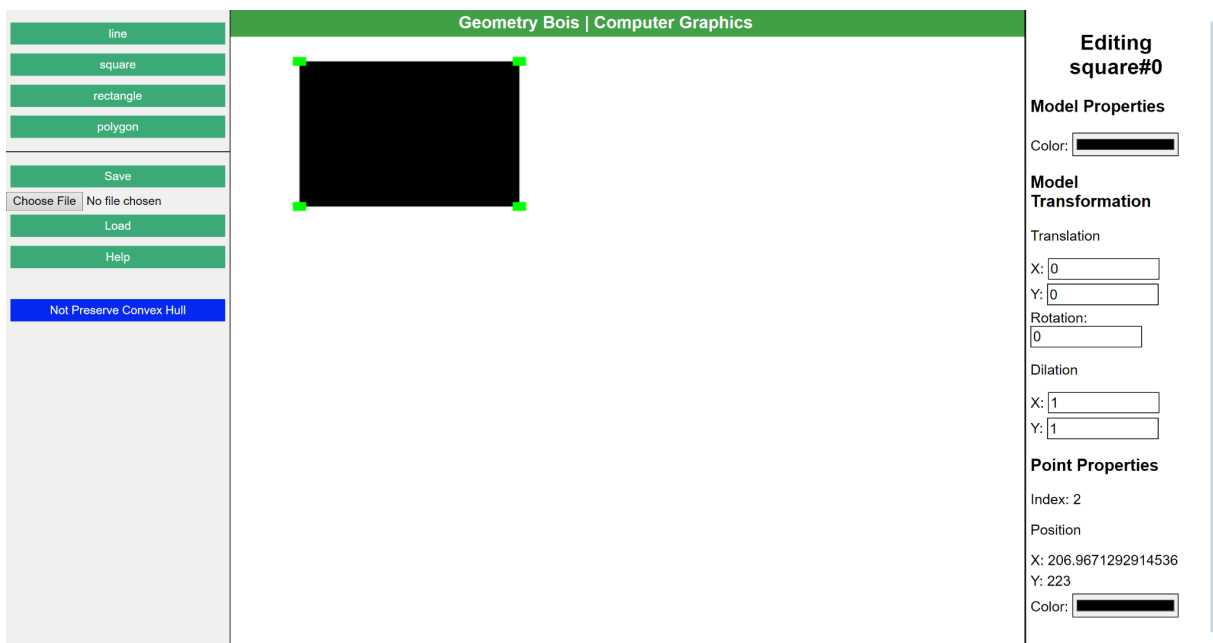
2.1 Model Garis



Gambar 2.1.1. Menggambar garis

Gambar tersebut dibentuk dari satu garis dengan warna dasar hitam.

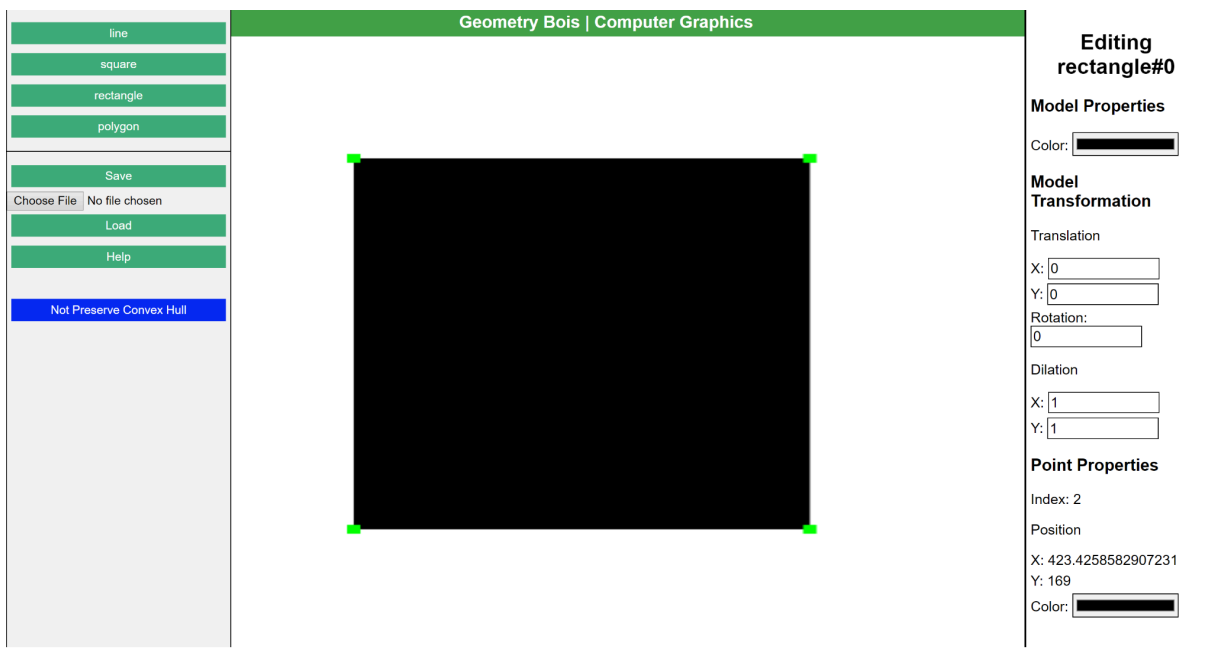
2.2 Model Persegi



Gambar 2.2.1. Menggambar persegi

Gambar tersebut dibentuk dari persegi garis dengan warna dasar hitam.

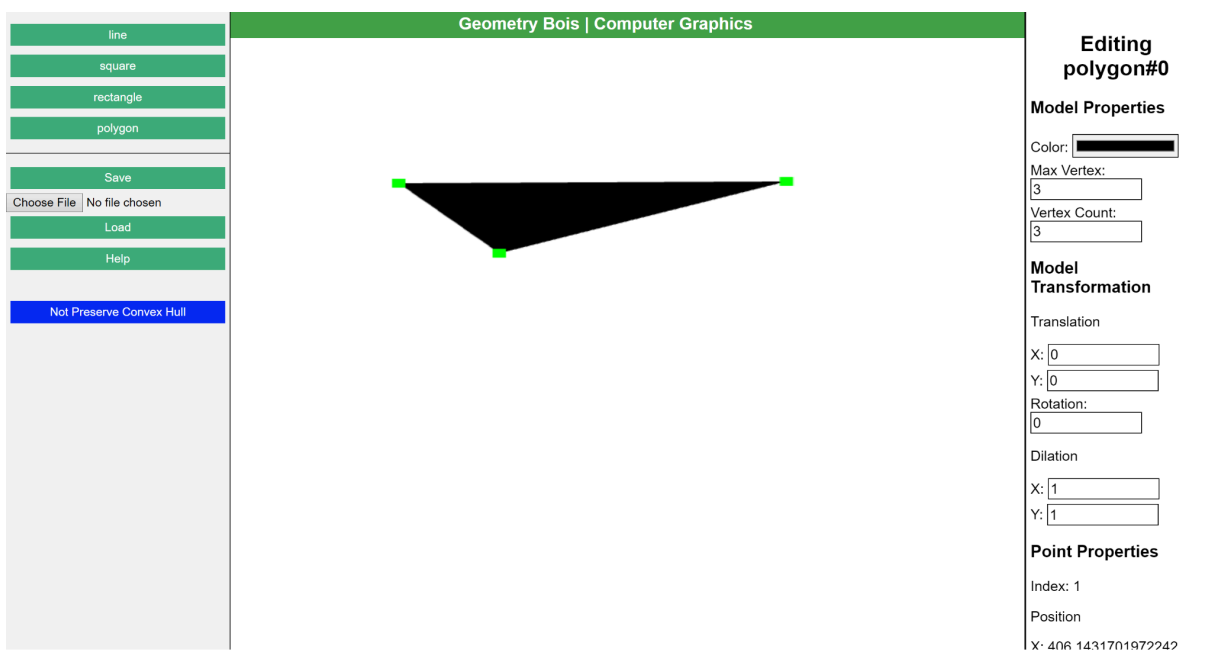
2.3 Model Persegi Panjang



Gambar 2.3.1. Menggambar persegi panjang

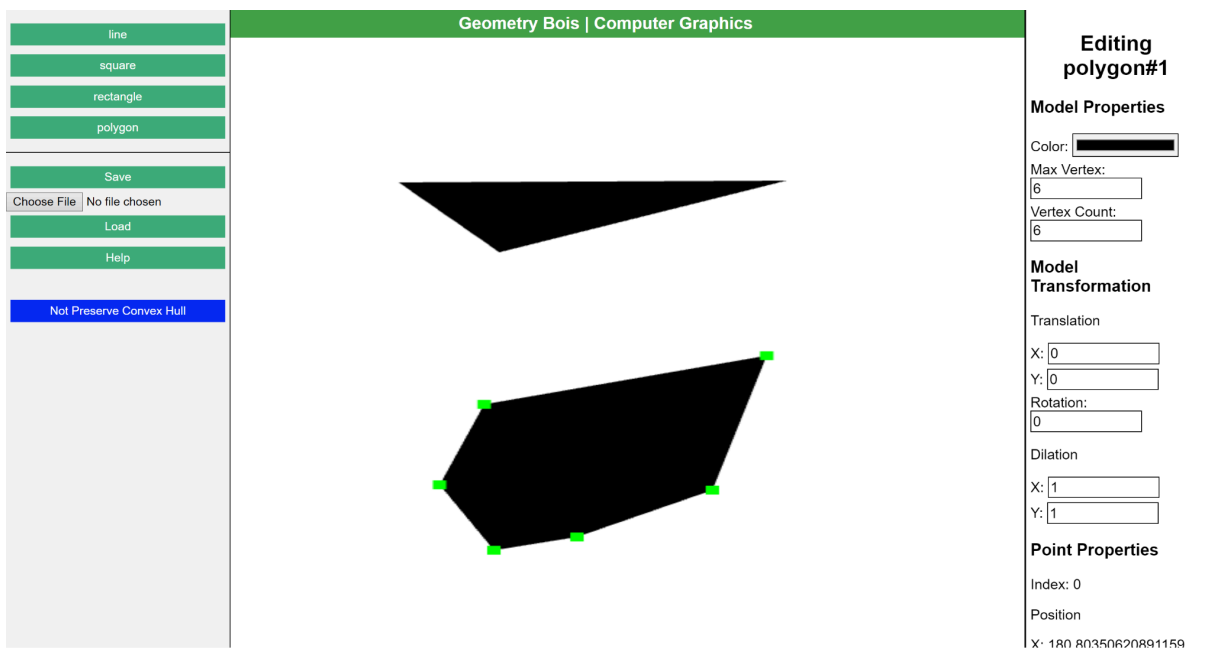
Gambar tersebut dibentuk dari satu persegi panjang dengan warna dasar hitam.

2.4 Model Polygon



Gambar 2.4.1. Menggambar poligon segitiga

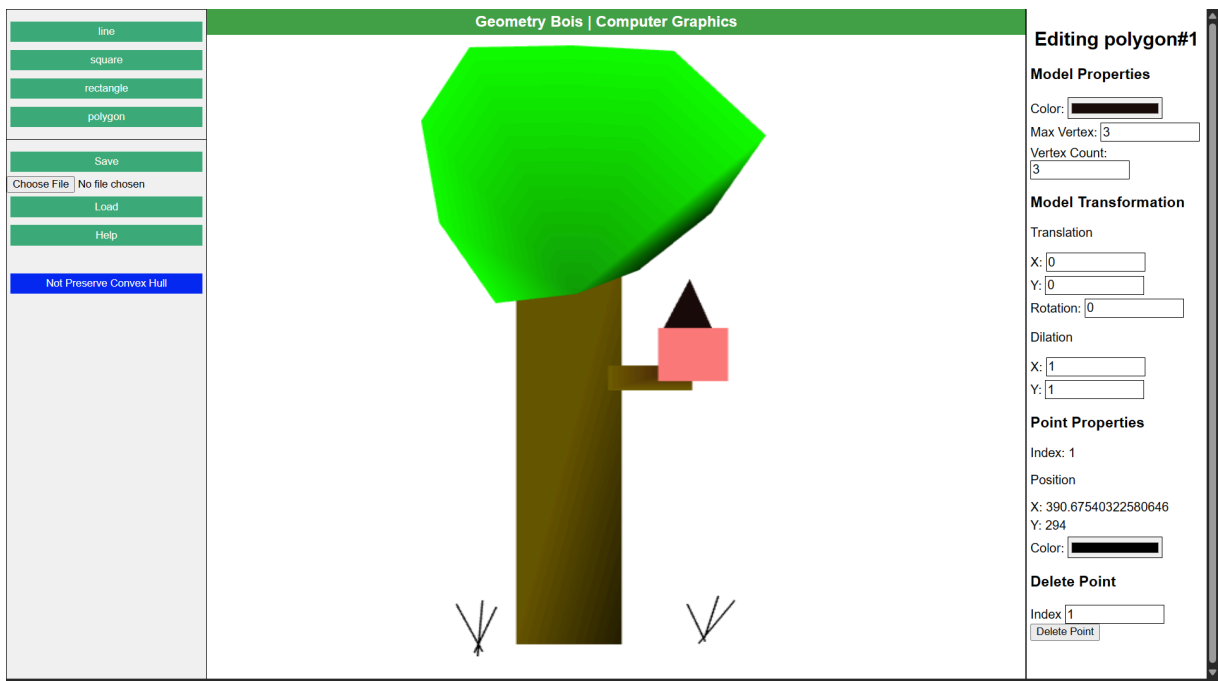
Gambar tersebut dibentuk dari satu polygon dengan warna dasar hitam dan tiga buah titik.



Gambar 2.4.2. Menggambar poligon segienam

Gambar tersebut dibentuk dari dua polygon dengan warna dasar hitam. Satu polygon memiliki tiga buah titik. Polygon lain memiliki 6 buah titik.

2.5 Model Gabungan



Gambar 2.5.1. Menggambar dari gabungan

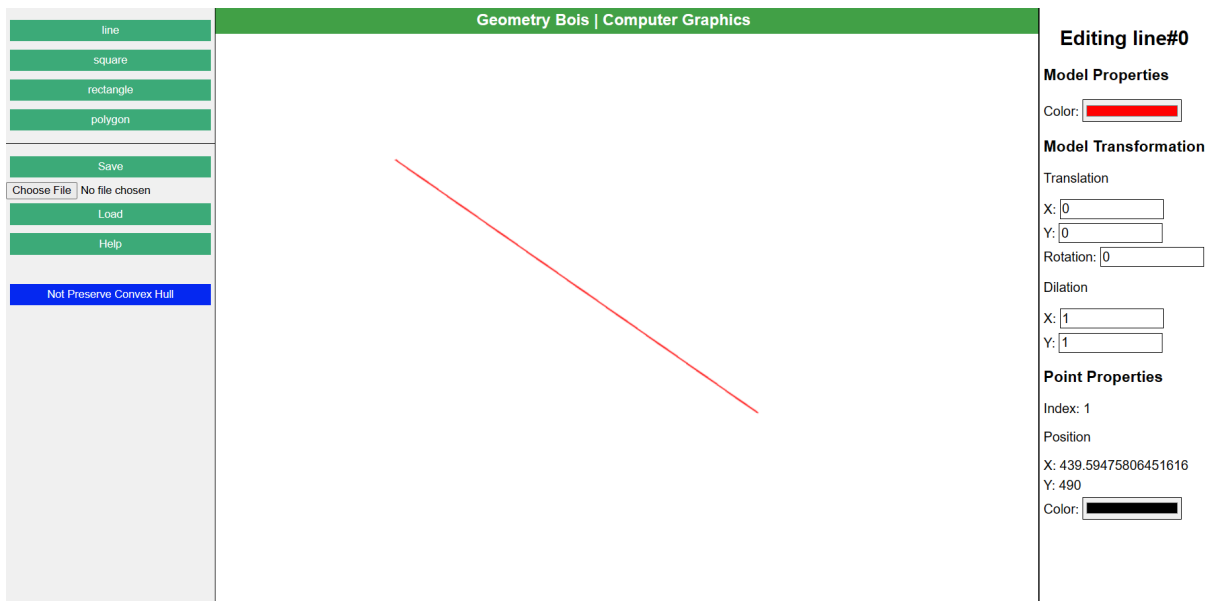
Gambar pohon dengan rumah burung tersebut dibentuk dari sebelas buah model. Sebuah poligon dengan 10 titik sebagai dedaunan. Dua buah persegi panjang sebagai batang dan dahan pohon. Enam buah garis sebagai rumput. Sebuah persegi sebagai rumah. Terakhir, sebuah poligon dengan tiga titik sebagai atap.

BAB III

MANUAL

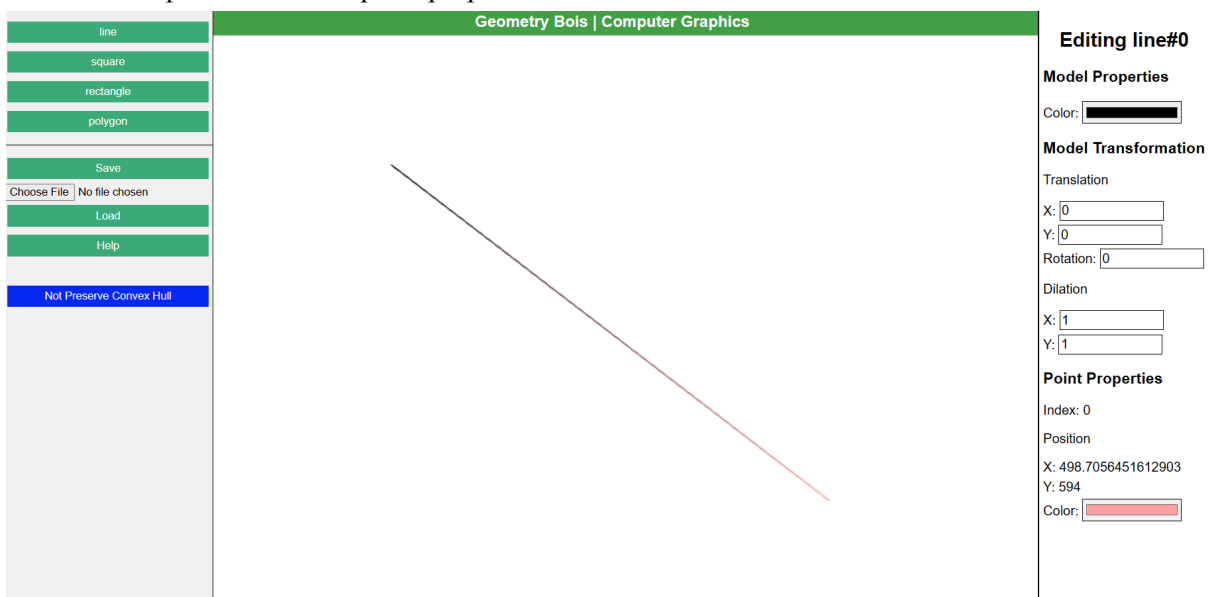
3.1 Model Garis

- Klik kanvas, kemudian lakukan *klik* untuk menentukan titik pertama garis yang diinginkan.
- Gerakkan tetikus untuk mengatur ukuran, setelah itu klik kembali untuk membentuk titik kedua garis, garis selesai dibentuk.
- Setelah garis dibentuk, klik salah satu titik sudut untuk melakukan modifikasi bentuk (panel kanan akan muncul).
- Pengguna bisa mengubah warna bentuk dengan mengklik color picker di panel kanan di bawah model properties.



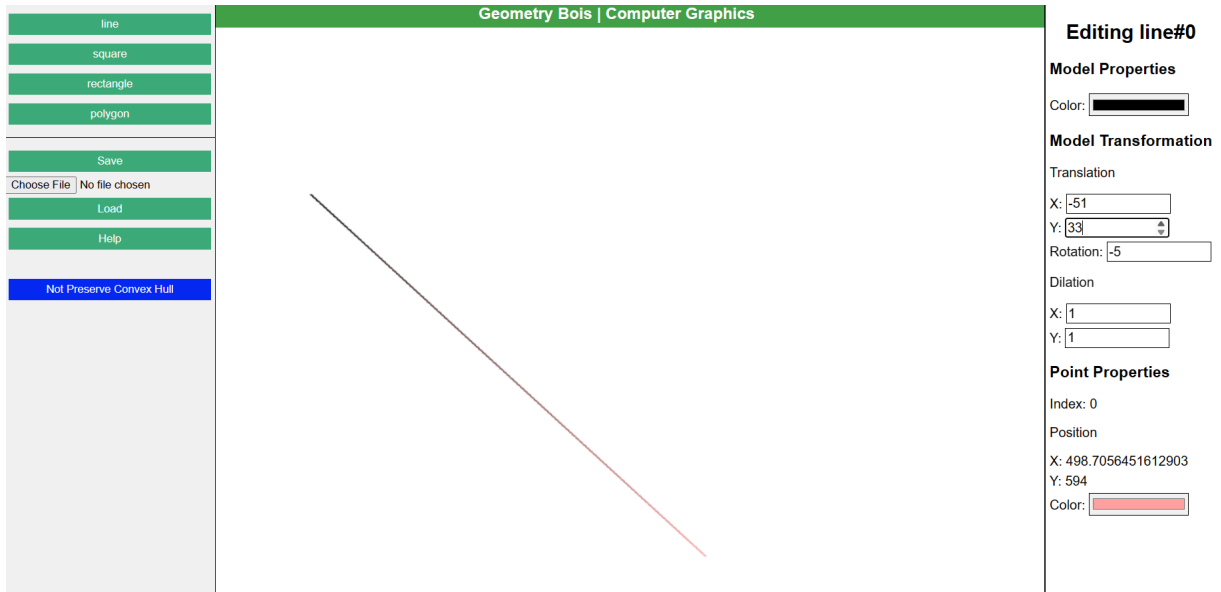
Gambar 3.2.1. Mengubah warna bentuk

- Pengguna bisa mengklik salah satu titik sudut dan mengubah warna sudut tersebut melalui color picker di bawah point properties.



Gambar 3.2.2. Mengubah warna sudut

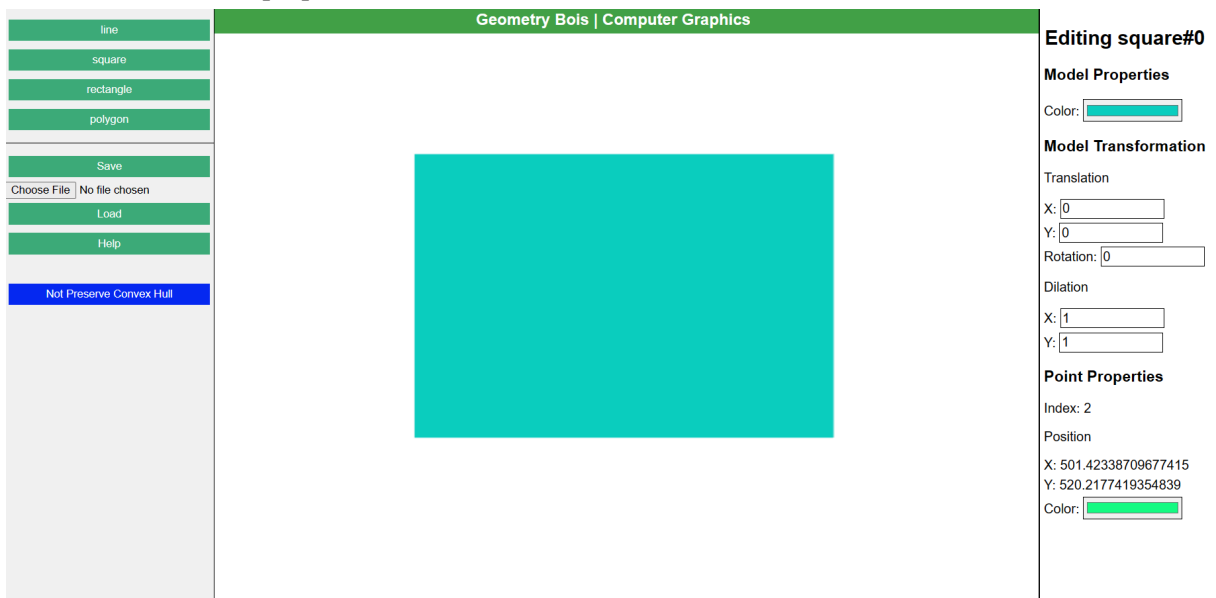
- Pengguna bisa menyesuaikan lokasi titik sudut dengan menggunakan *drag and drop* atau input translasi, rotasi, atau dilatasi.



Gambar 3.2.3. Melakukan transformasi pada garis dengan input

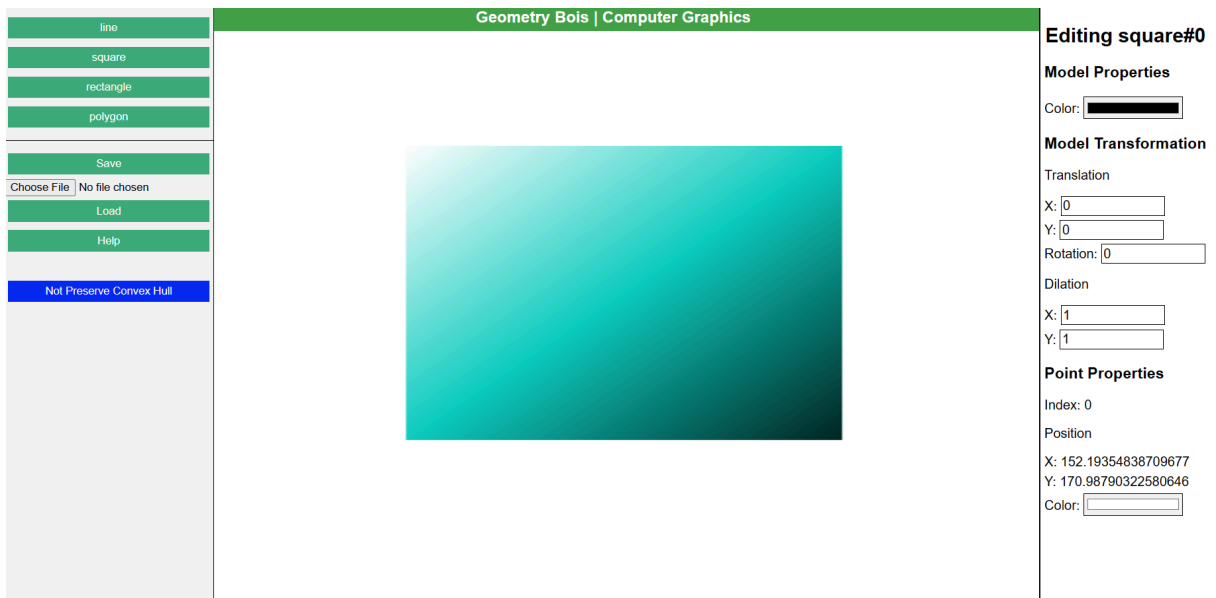
3.2 Model Persegi

- Klik kanvas, kemudian lakukan *klik* untuk menentukan titik tengah persegi yang diinginkan.
- Gerakkan tetikus untuk mengatur ukuran persegi, setelah itu klik kembali untuk membentuk persegi, persegi selesai dibentuk.
- Setelah persegi dibentuk, klik salah satu titik sudut untuk melakukan modifikasi bentuk (panel kanan akan muncul).
- Pengguna bisa mengubah warna bentuk dengan mengklik color picker di panel kanan di bawah model properties.



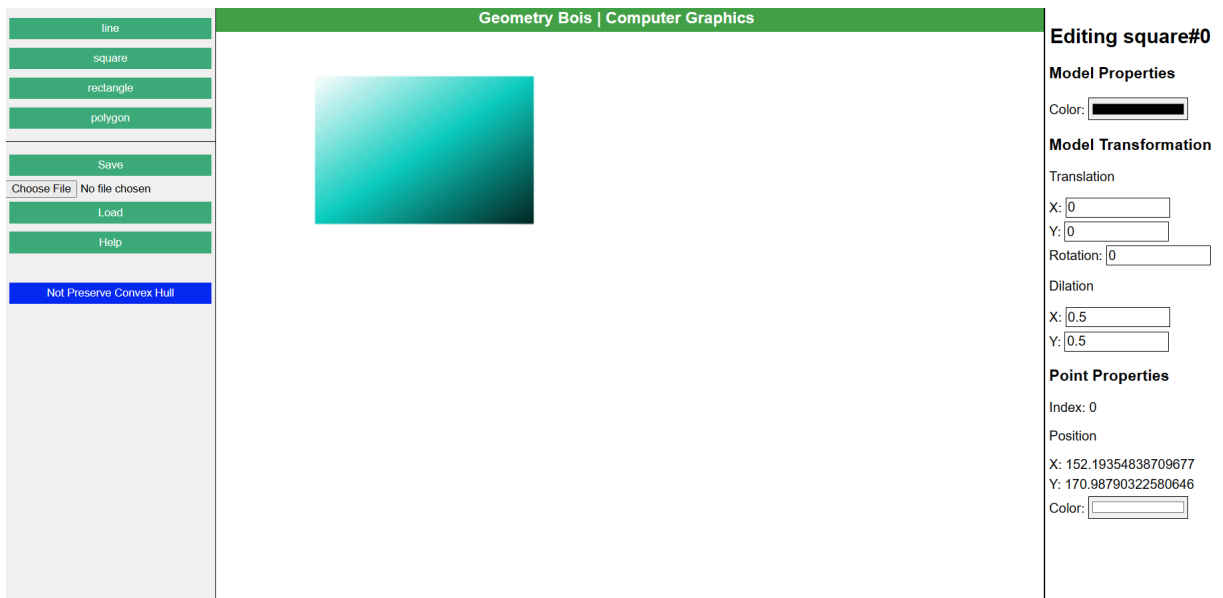
Gambar 3.2.1. Mengubah warna bentuk

- Pengguna bisa mengklik salah satu titik sudut dan mengubah warna sudut tersebut melalui color picker di bawah point properties.



Gambar 3.2.2. Mengubah warna sudut

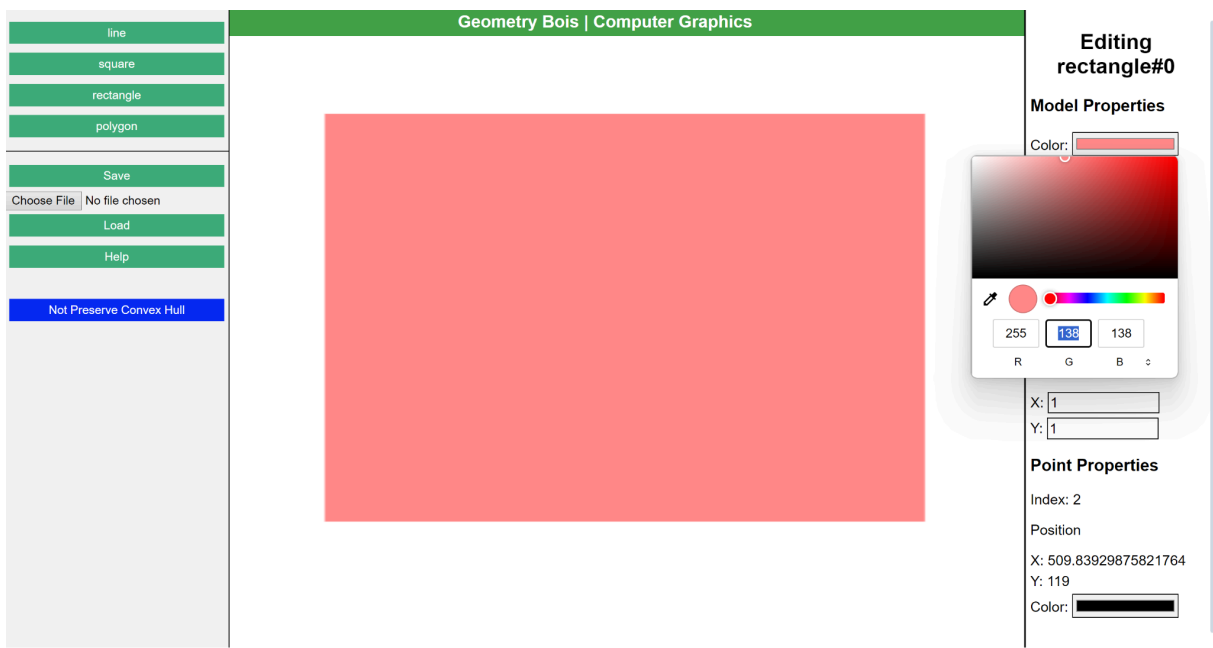
- Pengguna bisa menyesuaikan lokasi titik sudut dengan menggunakan *drag and drop* atau input translasi, rotasi, atau dilatasi.



Gambar 3.2.3. Melakukan transformasi pada persegi dengan input

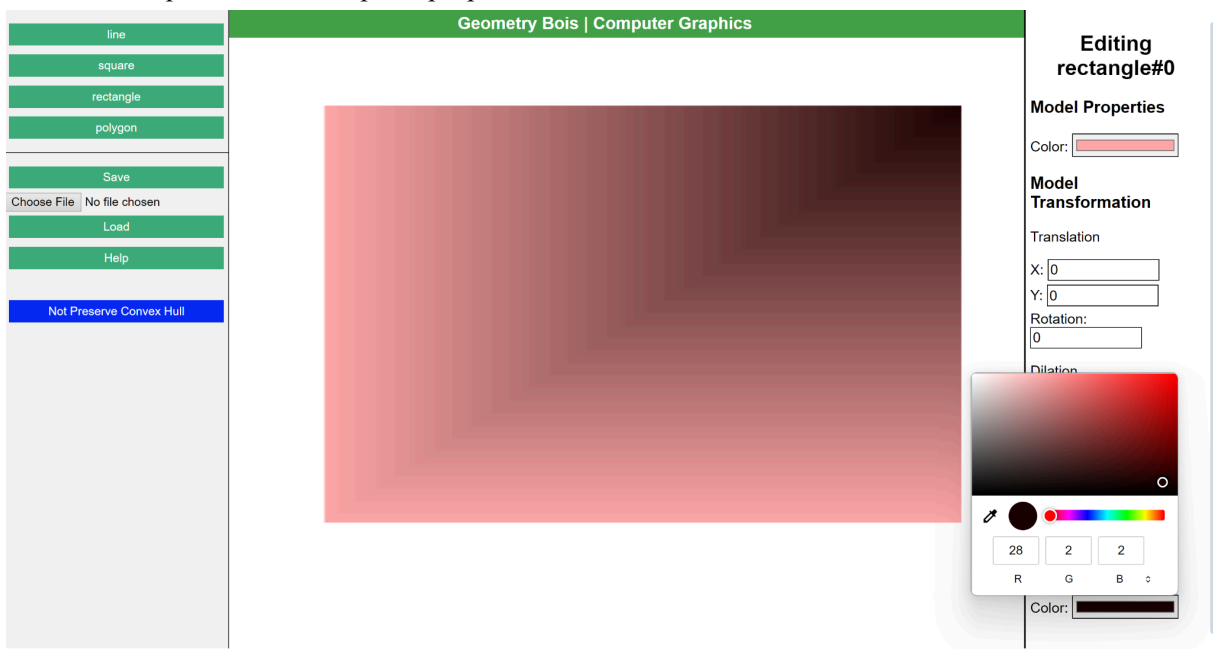
3.3 Model Persegi Panjang

- Klik kanvas, kemudian lakukan *drag and drop* untuk menentukan ukuran persegi panjang yang diinginkan.
- Setelah persegi panjang dibentuk, klik salah satu titik sudut untuk melakukan modifikasi bentuk (panel kanan akan muncul).
- Pengguna bisa mengubah warna bentuk dengan mengklik color picker di panel kanan di bawah model properties.



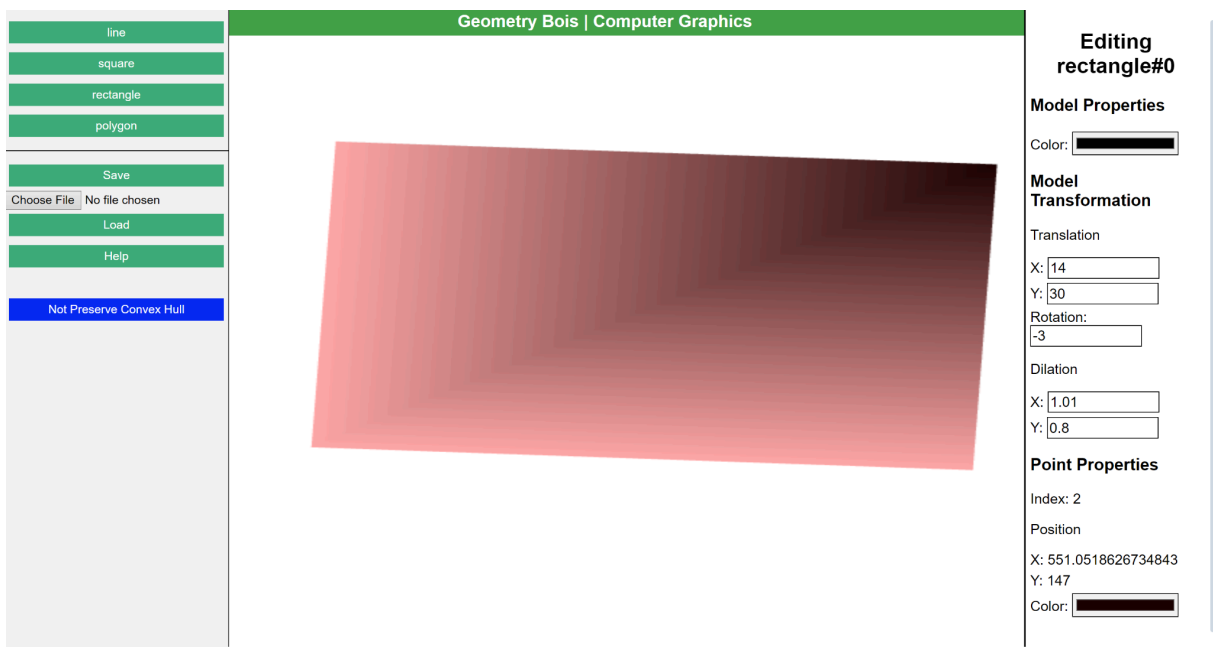
Gambar 3.3.1. Mengubah warna bentuk

- Pengguna bisa mengklik salah satu titik sudut dan mengubah warna sudut tersebut melalui color picker di bawah point properties.



Gambar 3.3.2. Mengubah warna sudut

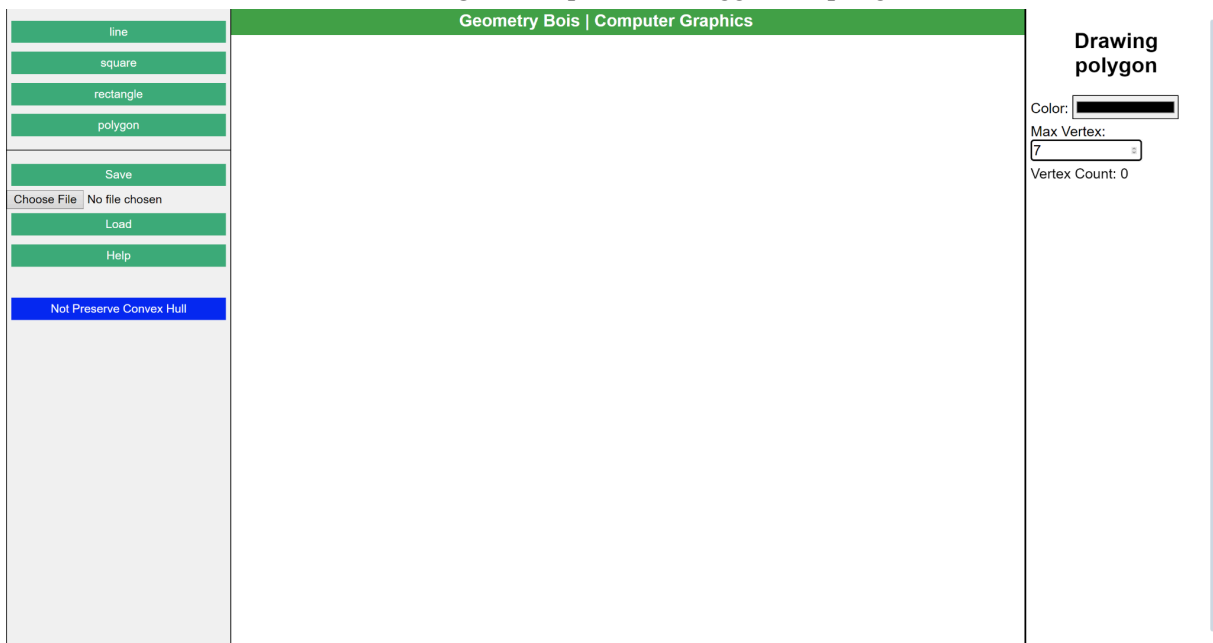
- Pengguna bisa menyesuaikan lokasi titik sudut dengan menggunakan *drag and drop* atau input translasi, rotasi, atau dilatasi.



.Gambar 3.3.3. Melakukan transformasi Rectangle dengan input

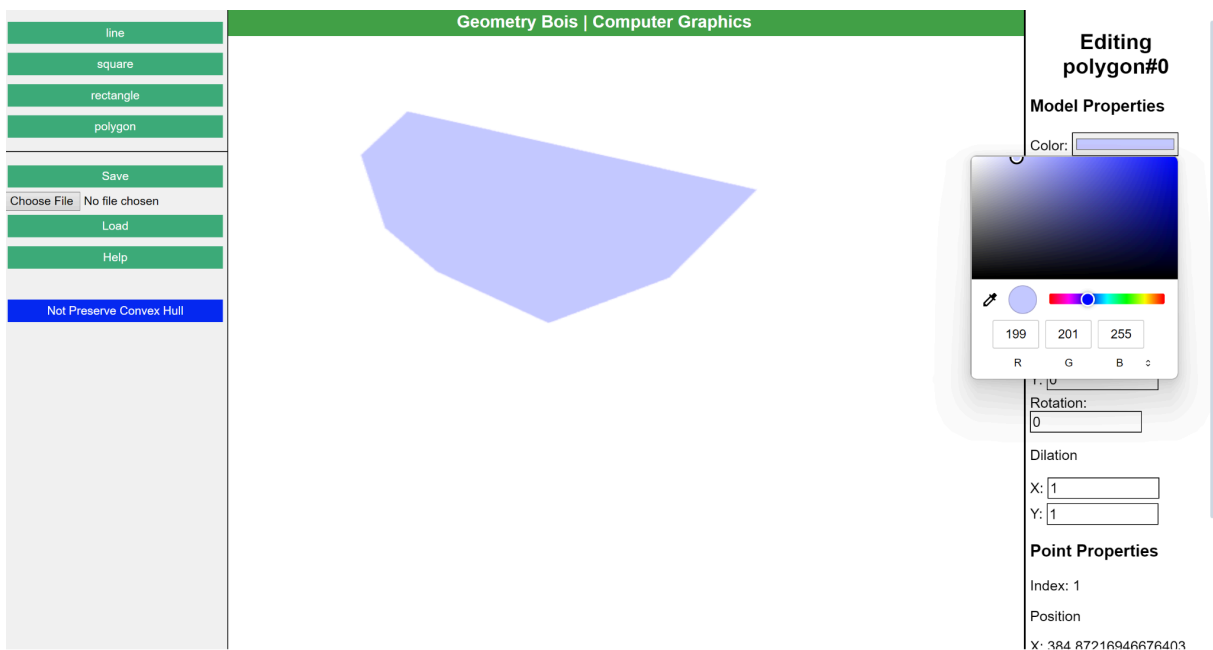
3.4 Model Polygon

- Klik tombol polygon di panel kiri, lalu tentukan maksimum verteks yang ingin dibentuk. Klik kanvas, kemudian lakukan *drag and drop* untuk menggambar poligon.



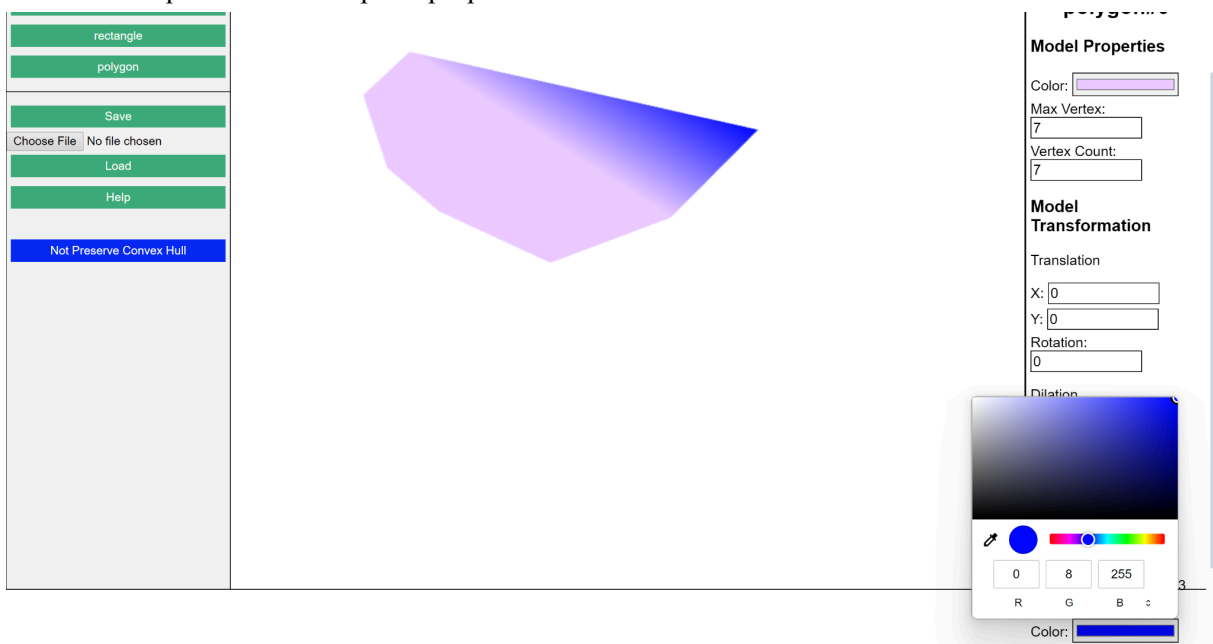
Gambar 3.4.1. Menentukan verteks poligon

- Setelah poligon dibentuk, klik salah satu titik sudut untuk melakukan modifikasi bentuk (panel kanan akan muncul).
- Pengguna bisa mengubah warna bentuk dengan mengklik color picker di panel kanan di bawah model properties.



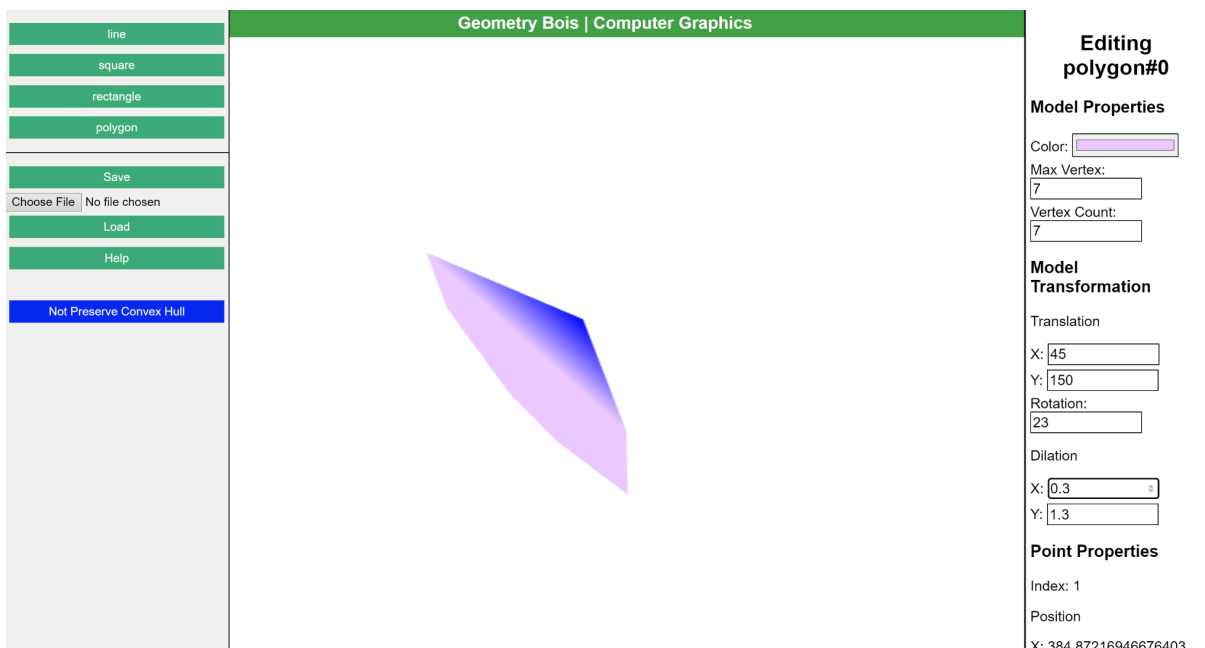
Gambar 3.4.2. Mengubah warna bentuk poligon

- Pengguna bisa mengklik salah satu titik sudut dan mengubah warna sudut tersebut melalui color picker di bawah point properties.



Gambar 3.4.3. Mengubah warna sudut poligon

- Pengguna bisa menyesuaikan lokasi titik sudut dengan menggunakan *drag and drop* atau input translasi, rotasi, atau dilatasi.



.Gambar 3.4.4. Melakukan transformasi Polygon dengan input

- Pengguna bisa menghapus titik sudut dengan mengklik titik sudut yang ingin dihapus, atau mengisi input “delete point” berdasarkan indeks point.

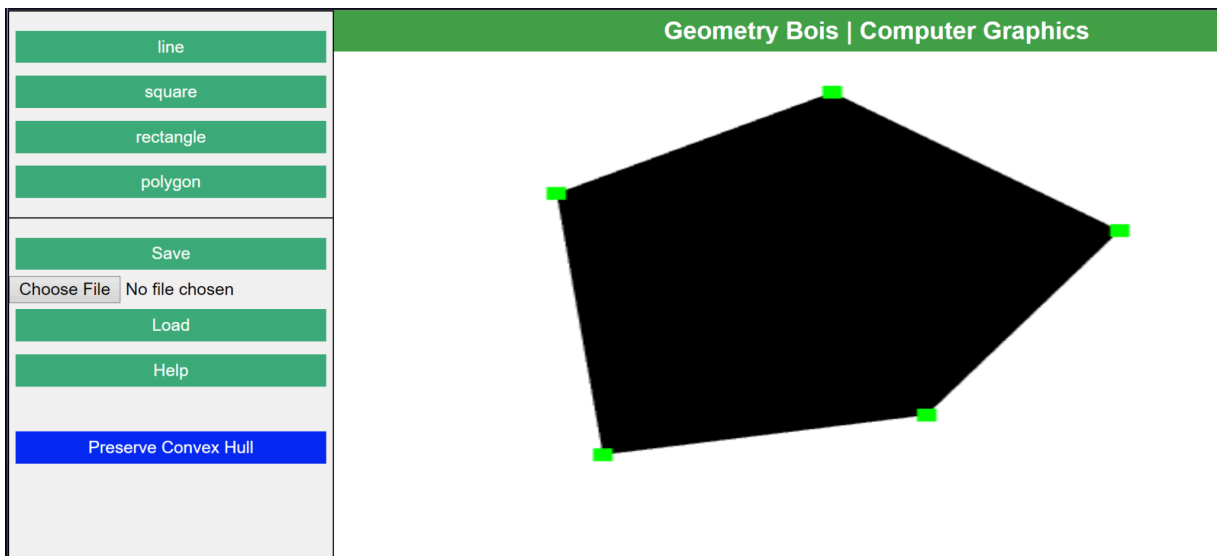
Delete Point

Index

Delete Point

.Gambar 3.4.5. Menghapus titik sudut poligon

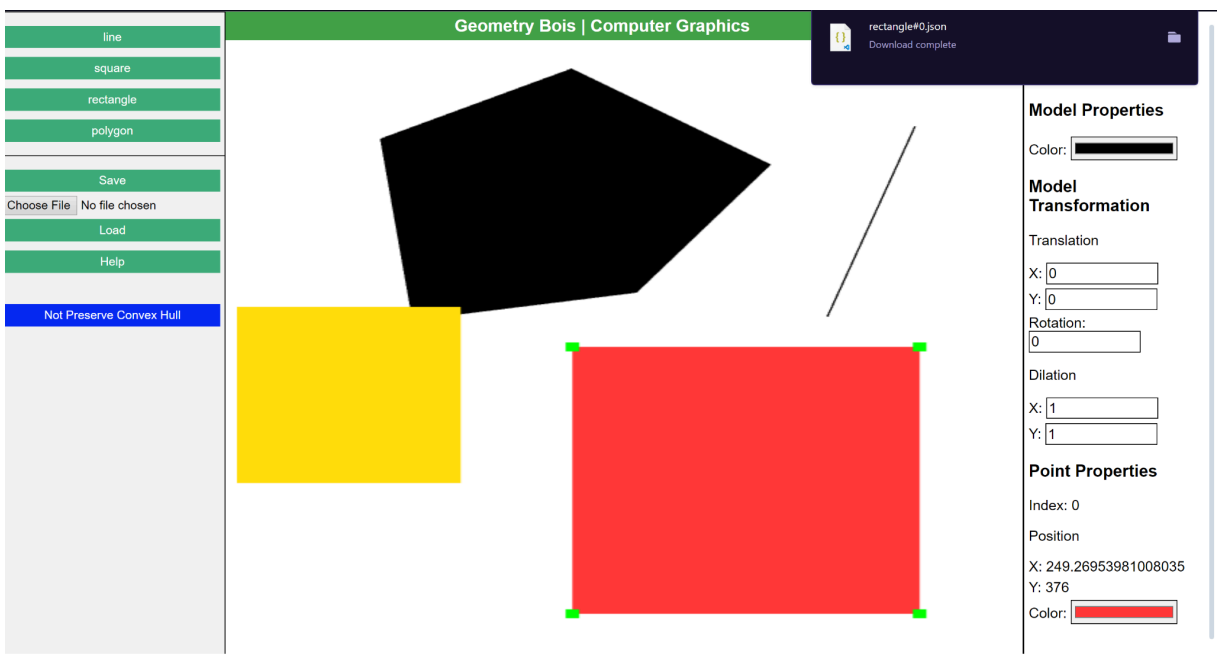
- Pengguna mampu mempertahankan bentuk poligon berdasarkan convex hull dengan mengklik tombol toggle “Not Preserve Convex Hull”.



.Gambar 3.4.6. Convex hull poligon

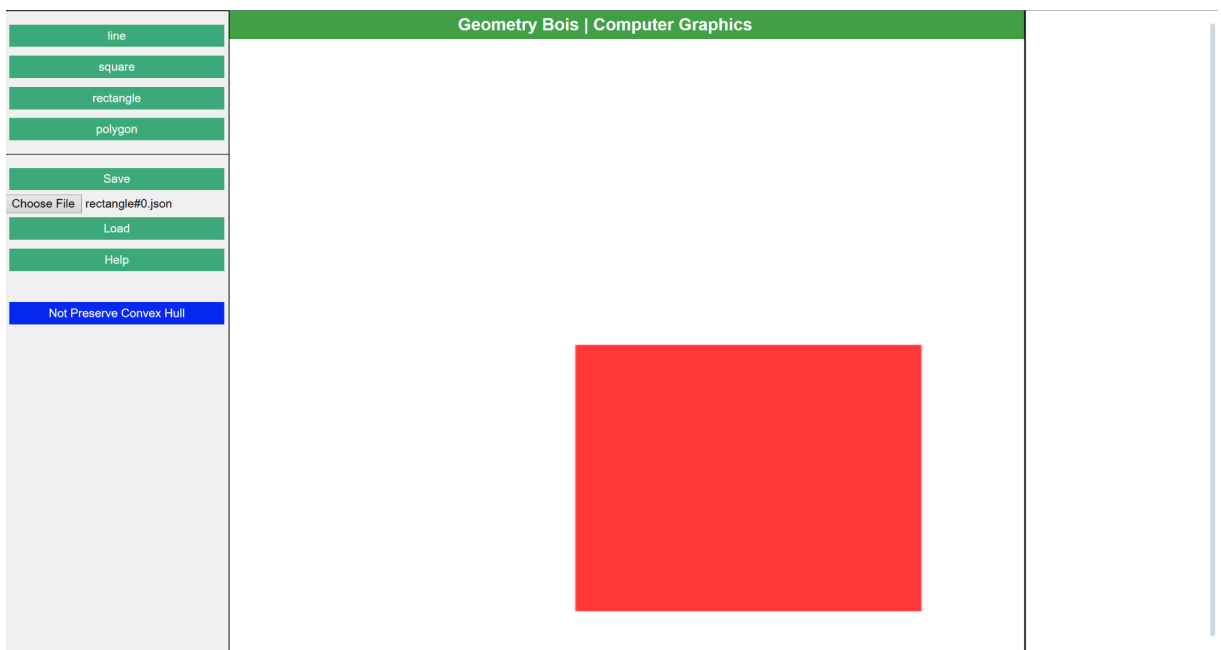
3.5 Save & Load

- Pilih salah satu model yang ingin di-save, misalnya Rectangle. Lalu tekan tombol save, dan file .json yang berisi informasi Rectangle tersebut akan di-download secara otomatis.



.Gambar 3.5.1. Save model Rectangle

- Di kanvas baru, klik tombol “Choose File”, pilih file .json Rectangle, lalu tekan tombol Load. Model Rectangle akan dimuat seperti sebelumnya.



.Gambar 3.5.2. Load model Rectangle