

Tugas Besar

IF3260 Grafika Komputer

2D Primitive Elements



Disusun oleh:

13521060 Fatih Nararya R.I.

13521089 Kenneth Ezekiel Suprantonni

13521101 Arsa Izdiyar Islam

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

2023

DAFTAR ISI

DAFTAR ISI	2
BAB I	
DESKRIPSI HASIL	3
1. Deskripsi Umum	3
2. Struktur Aplikasi	3
3. Komponen	4
BAB II	
FUNGSIONALITAS	5
1. Fungsionalitas Utama	5
2. Fungsionalitas Tambahan	11
BAB III	
LAMPIRAN	13

BAB I

DESKRIPSI HASIL

1. Deskripsi Umum

Program ini dirancang untuk memberikan pengenalan yang mendalam tentang dunia grafika komputer, khususnya bagaimana menggambar geometri-geometri dasar pada layar monitor. Fokus dari tugas ini adalah pemahaman dan implementasi dasar-dasar WebGL, termasuk penggunaan vertex dan fragment shaders, manipulasi geometri, serta transformasi objek.

Program ini memungkinkan pengguna untuk membuat, mengedit, dan memvisualisasikan berbagai model 2D secara interaktif pada kanvas. Pengguna dapat menggambar garis, persegi, persegi panjang, dan poligon, serta menerapkan transformasi geometri seperti translasi, rotasi, dan dilatasi pada objek-objek tersebut.

Terdapat juga beberapa fitur untuk setiap model seperti mengubah warna titik sudut, menggerakkan titik sudut menggunakan drag-and-drop, dan menyimpan serta memuat kembali model. Selain itu, beberapa fitur lanjutan yang diimplementasikan dalam program adalah Convex Hull untuk poligon, Animasi, dan Penguncian titik sudut secara implisit.

Untuk mengembangkan program ini, digunakan WebGL murni tanpa library eksternal untuk WebGL, tetapi diperbolehkan menggunakan library lain untuk membangun UI atau aspek visual lainnya.

2. Struktur Aplikasi

Aplikasi dibuat menggunakan WebGL di dalam vanilla javascript (ditambah typescript) dengan bundler vite. Vanilla javascript dipilih karena sebagian besar dari *logic* dan mekanisme kerja aplikasi terdapat pada canvas WebGL, bukan DOM. Typescript dipilih untuk mempermudah typing dan memastikan program secure dari type bug.

Life Cycle dari aplikasi dimulai dari kelas application yang akan manage state dan data dari aplikasi. Pada struktur folder, terdapat folder shaders yang berisi file .glsl yaitu untuk menyimpan script vertex shader dan fragment shader. File .glsl ini dapat langsung diimport secara plaintext dibantu oleh plugin plaintext dari vite. Folder app-states berisi state-state dari aplikasi, mulai dari idle, drawing, dan select-shape.

State ini yang akan mempunyai handler event tersendiri dan memungkinkan untuk perubahan antar state. Folder lib berisi utility aplikasi seperti class program untuk type safely dan secara modular manage WebGL program, shader, dan variabel di dalamnya, komponen UI, dan juga model-model yang ada pada program.

3. Komponen

- Main
Main adalah program utama yang dipanggil, di dalamnya hanya menginisiasi canvas HTML dan memanggil Application Class dari aplikasi
- Application
Application Class adalah kelas utama yang mengatur keberjalanan aplikasi, menginisiasi program WebGL, Front-end component logic, logic aplikasi, perubahan state, dan juga menggambar seluruh bentuk-bentuk dengan program. Application class ini akan memiliki sebuah app state yang merupakan objek dari kelas-kelas pada folder app-states. Masing-masing state akan terhubung satu sama lain melalui logic event pada canvas dan juga DOM. Aplikasi class ini juga akan melakukan passing event kepada state tersebut.
- Program
Sebuah wrapper class untuk program WebGL yang menginisiasi program dengan vertex shader dan fragment shader serta membantu untuk melakukan setup uniform dan atribut.
- App-states
App-states adalah kumpulan kelas yang diturunkan dari BaseAppState Class, yang adalah kelas abstrak dari application state. State yang terdapat dalam aplikasi adalah idle, drawing, dan select-shape. State ini akan mengatur behavior aplikasi dalam event-event yang ada.
- Drawables
Drawables adalah kumpulan kelas yang diturunkan dari Drawable Class, yang adalah kelas abstrak dari bentuk geometri atau model. Bentuk geometri yang bisa digambar dalam aplikasi adalah line, rectangle, square, dan polygon. Masing-masing kelas akan mengimplementasikan sesuai dengan life cycle yang diperlukan aplikasi dan juga state yang ada sehingga dapat dilakukan polymorphism dengan baik.
- Komponen pembantu
Terdapat juga beberapa komponen pembantu yang melengkapi aplikasi
 - i. Primitives

Berisi tipe dan fungsi primitif yang berkaitan dengan vektor, point, dan transformasi.

ii. Toolbar

Berisi utilitas untuk mengatur toolbar pada aplikasi yang terdapat pada bagian kiri layar.

iii. Slider

Berisi utilitas untuk mengatur slider, yang akan digunakan pada rotasi.

iv. ColorPicker

Berisi utilitas untuk mengatur color picker.

v. FileInput

Berisi utilitas untuk melakukan import file untuk objek-objek yang ada pada aplikasi.

vi. Loader

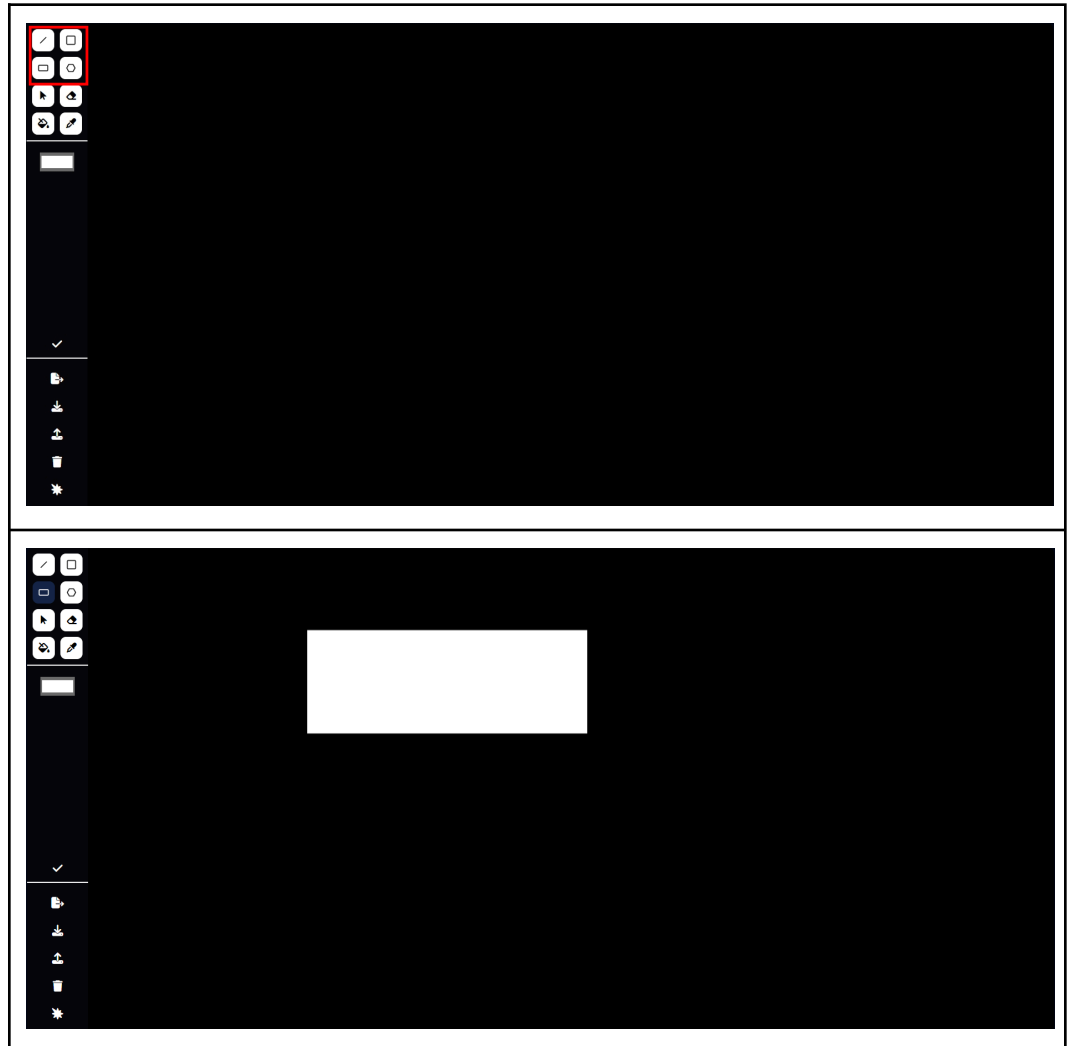
Berisi utilitas untuk melakukan parsing ataupun stringify untuk objek-objek yang ada pada aplikasi dengan bentuk data berupa JSON.

BAB II

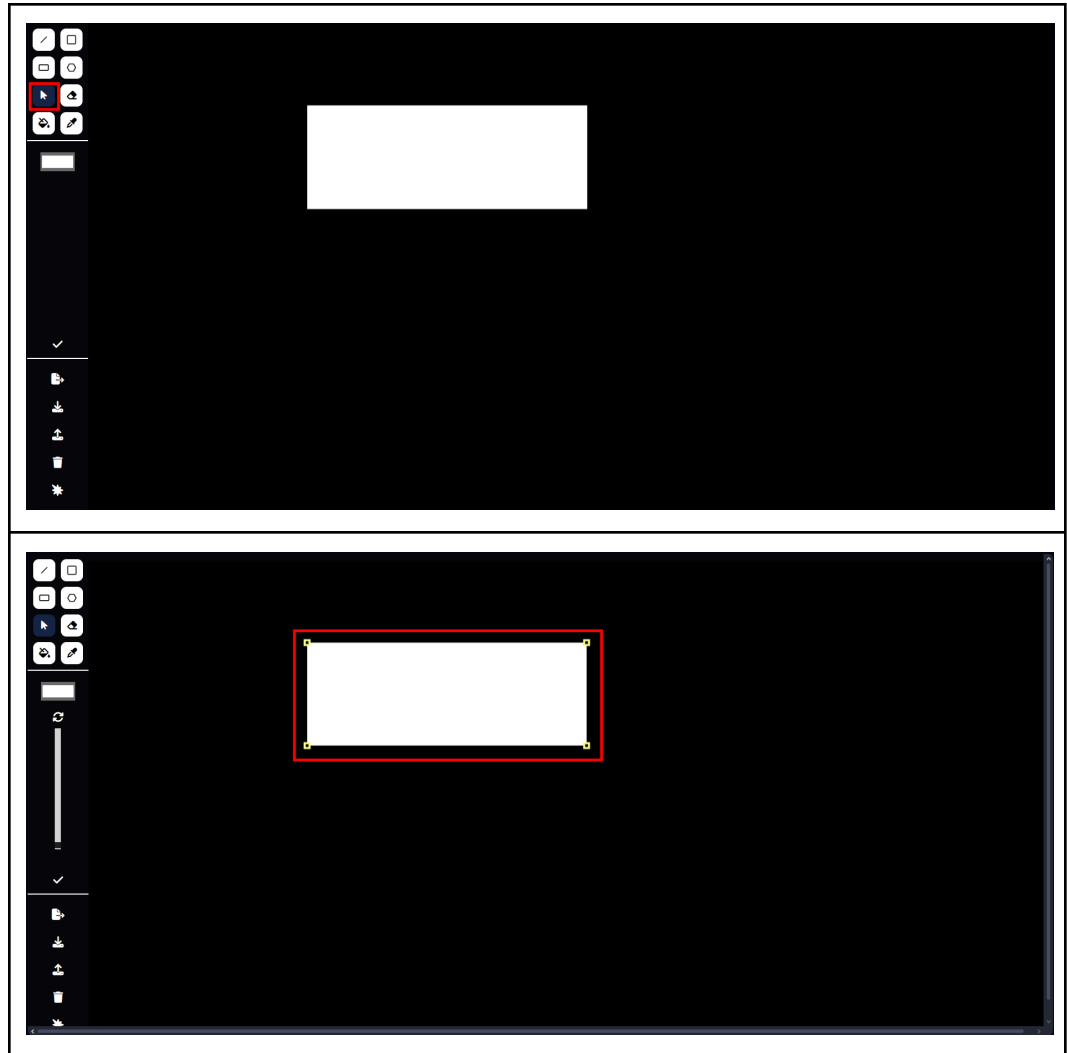
FUNGSIONALITAS

1. Fungsionalitas Utama

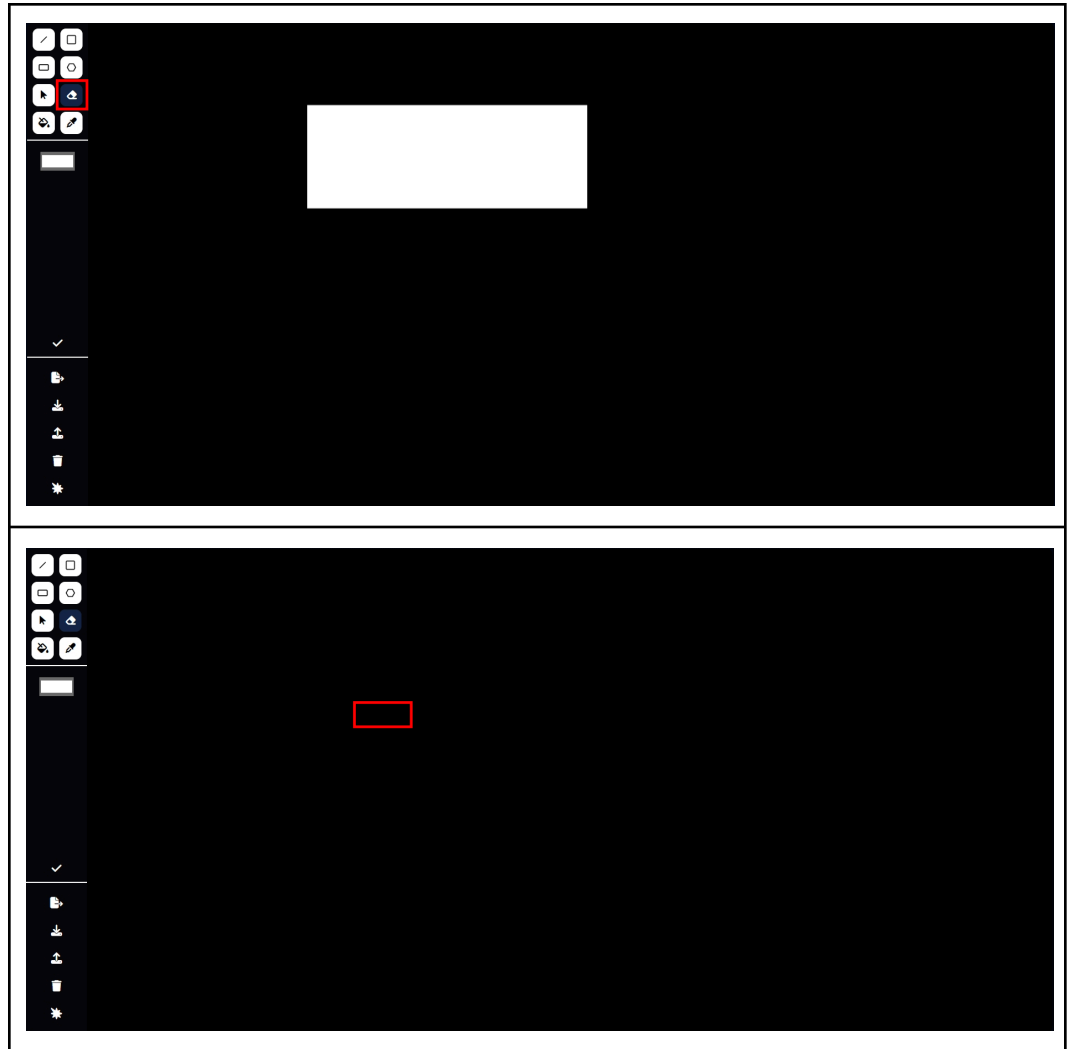
- Menggambar bentuk geometri dasar
Pilih salah satu bentuk yang terdapat pada *toolbar* lalu klik pada kanvas



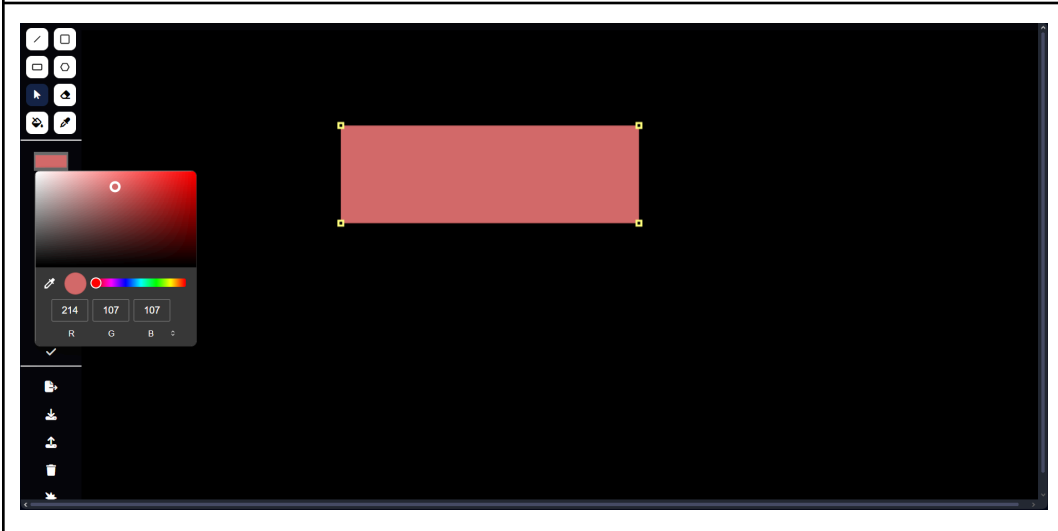
- Memilih gambar bentuk yang berada di kanvas
Pilih opsi select atau Shift + V lalu klik pada salah satu gambar



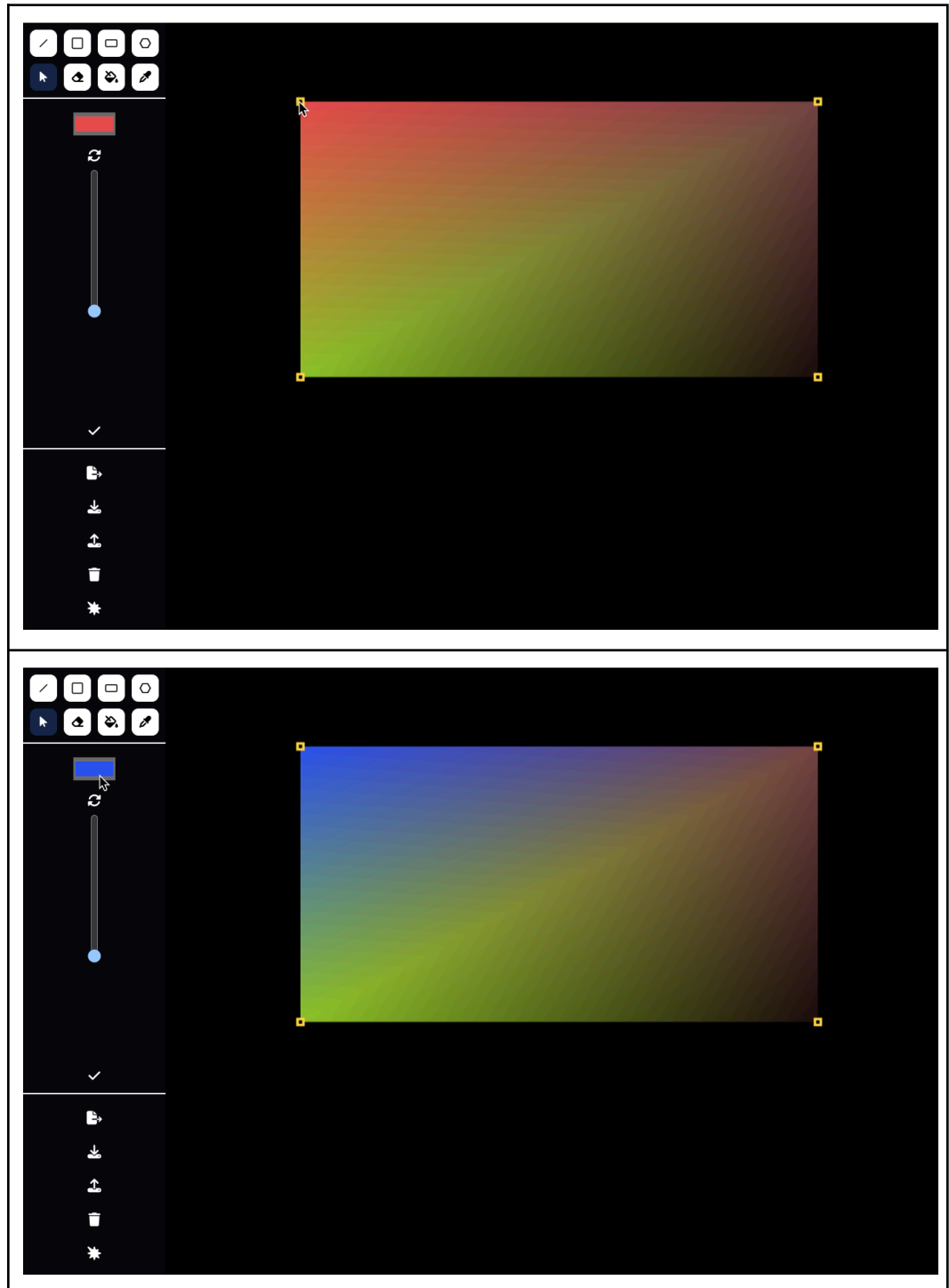
- Menghapus gambar bentuk yang berada di kanvas
Pilih opsi hapus atau Shift + D lalu klik pada salah satu gambar



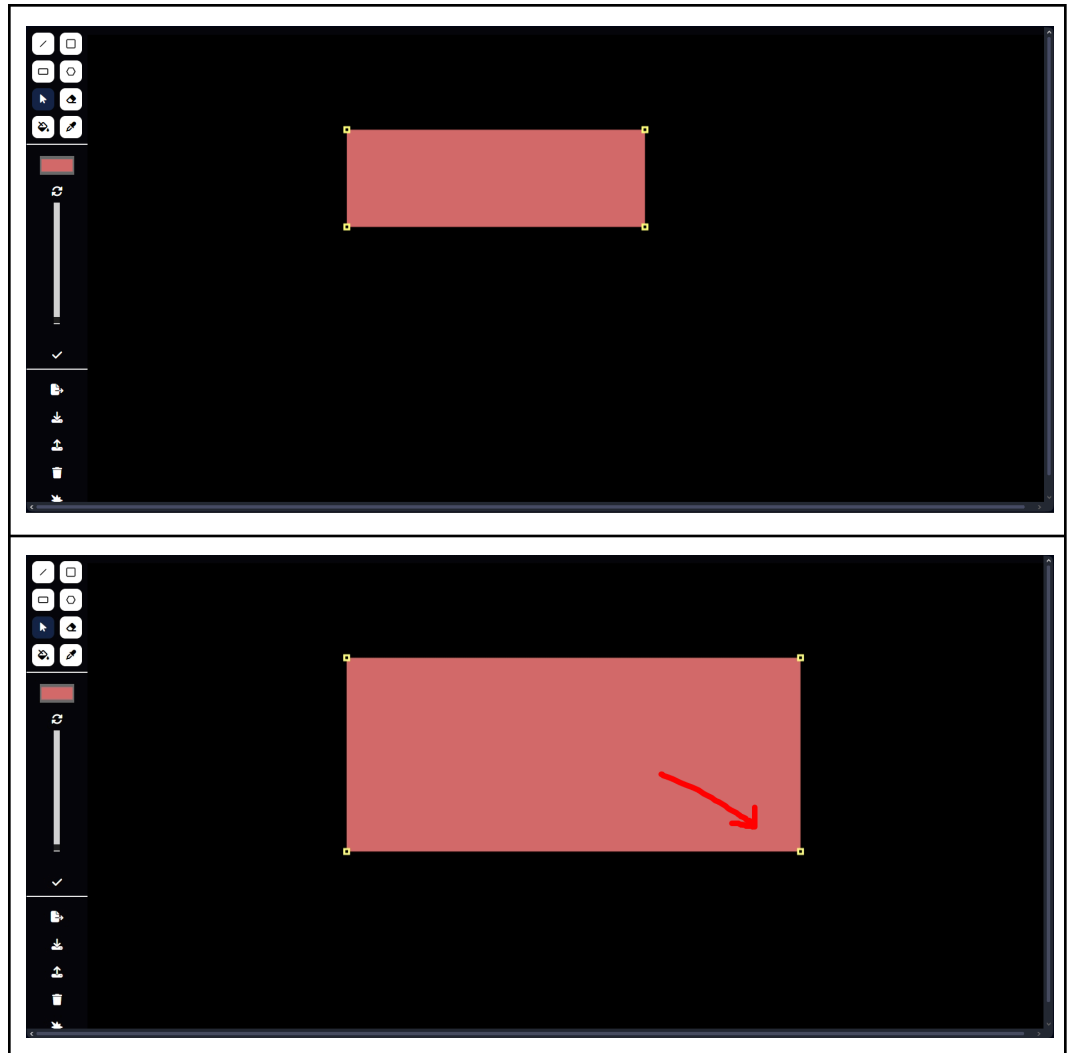
- Mengganti warna gambar bentuk
Select salah satu gambar lalu ganti warna dengan *color picker*



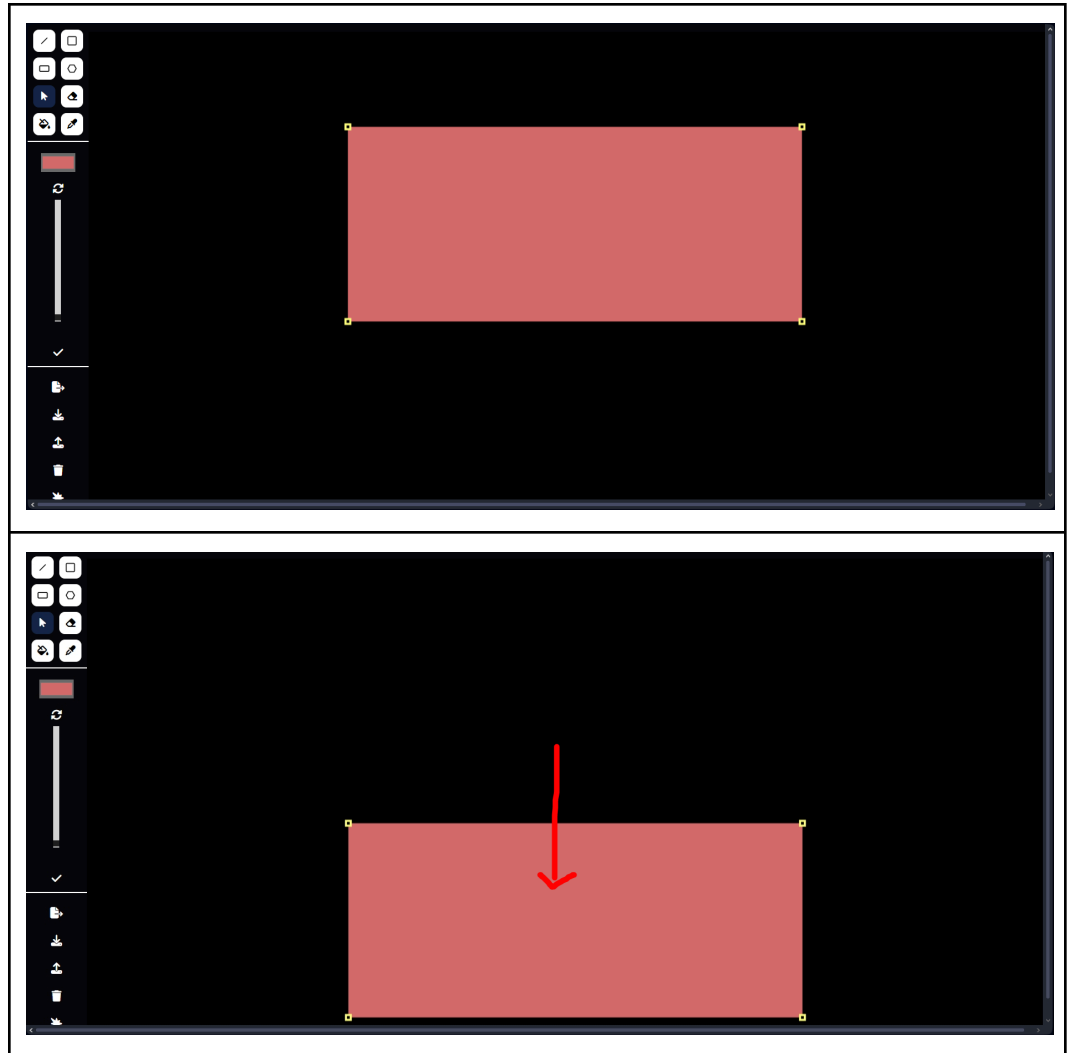
- Mengganti warna dari titik sudut
Select salah satu gambar, klik salah satu titik sudut, lalu ganti warna menggunakan *color picker*



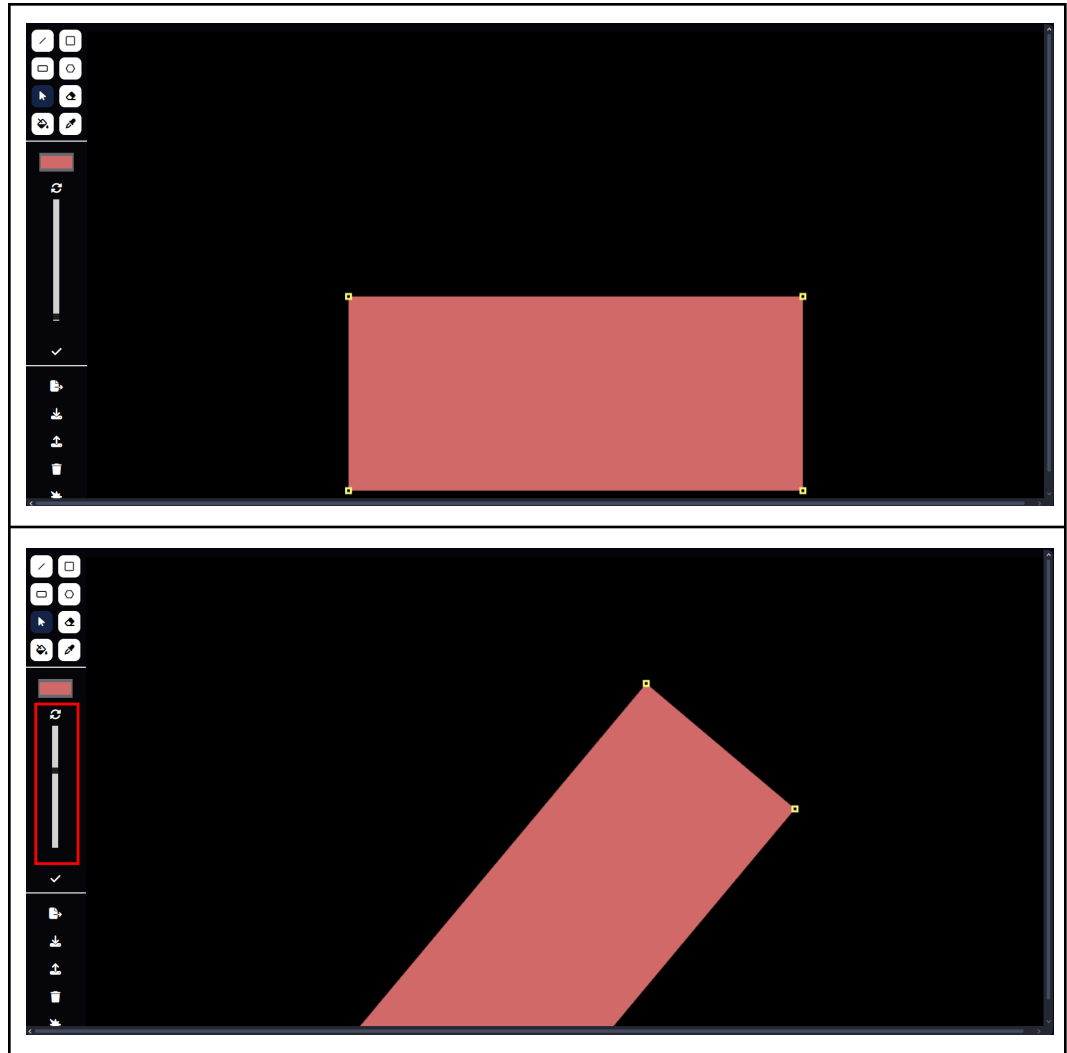
- Mengubah ukuran gambar bentuk yang berada di kanvas
Select salah satu gambar lalu lakukan drag pada salah satu titik sudut



- Menggeser gambar bentuk yang berada di kanvas
Select salah satu gambar lalu lakukan drag pada body gambar

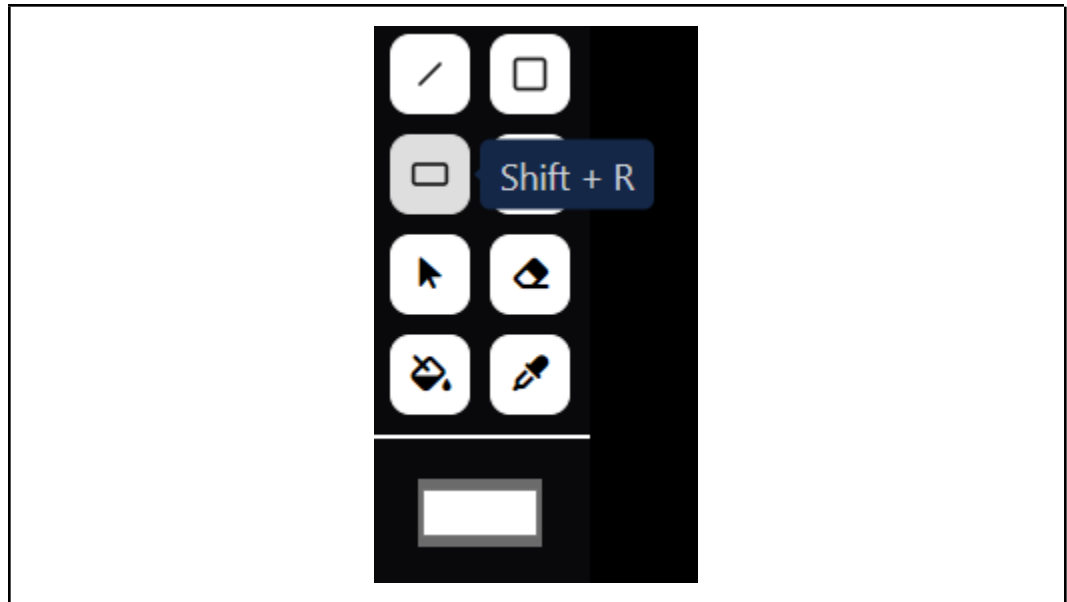


- Merotasi gambar bentuk yang berada di kanvas
Select salah satu gambar lalu geser slider yang ada pada toolbar

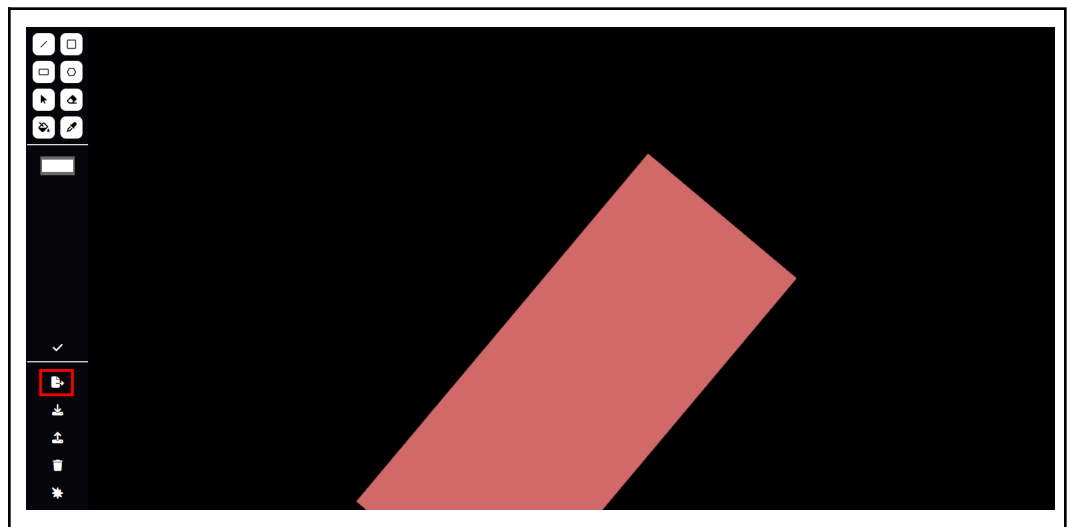


2. Fungsionalitas Tambahan

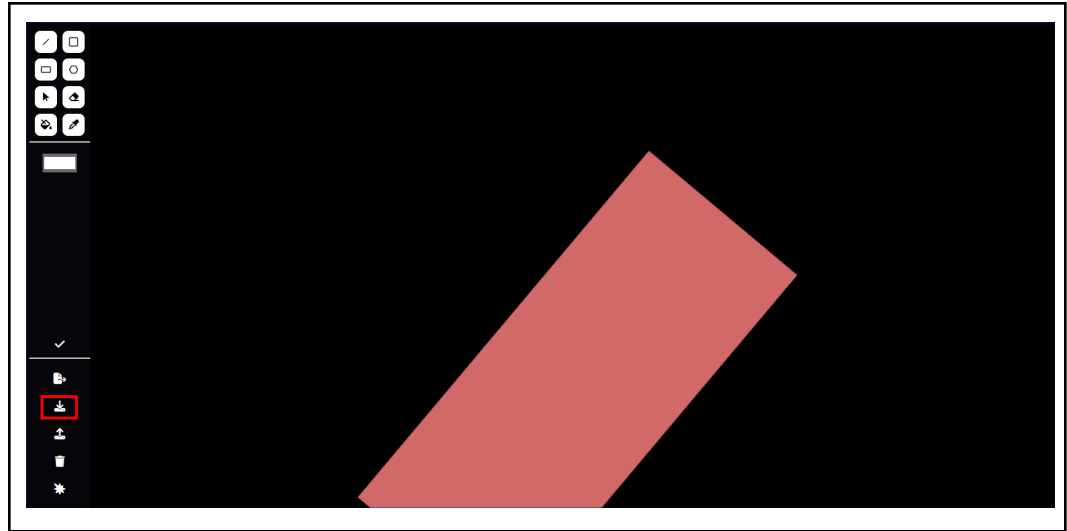
- Memilih toolbar menggunakan shortcut



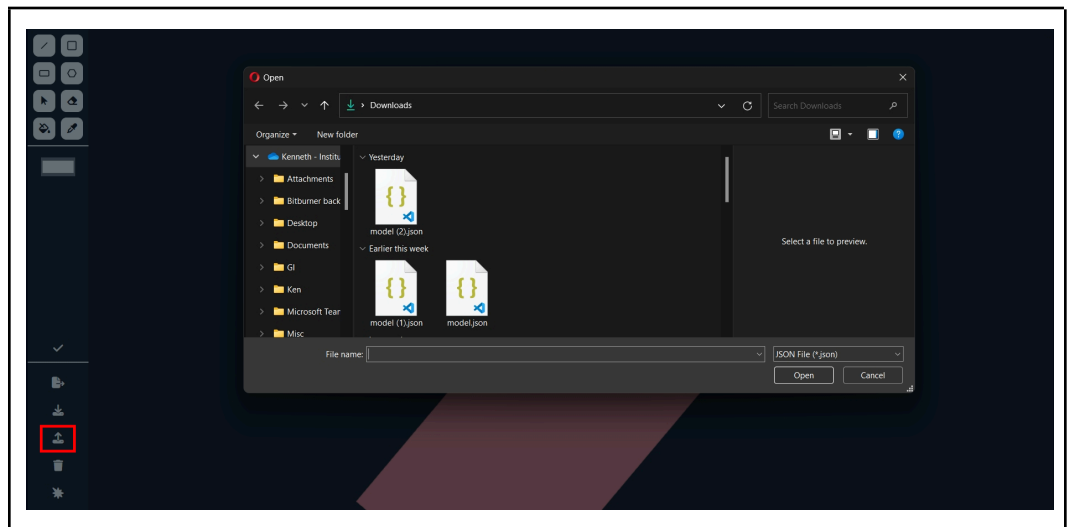
- Mengekspor gambar



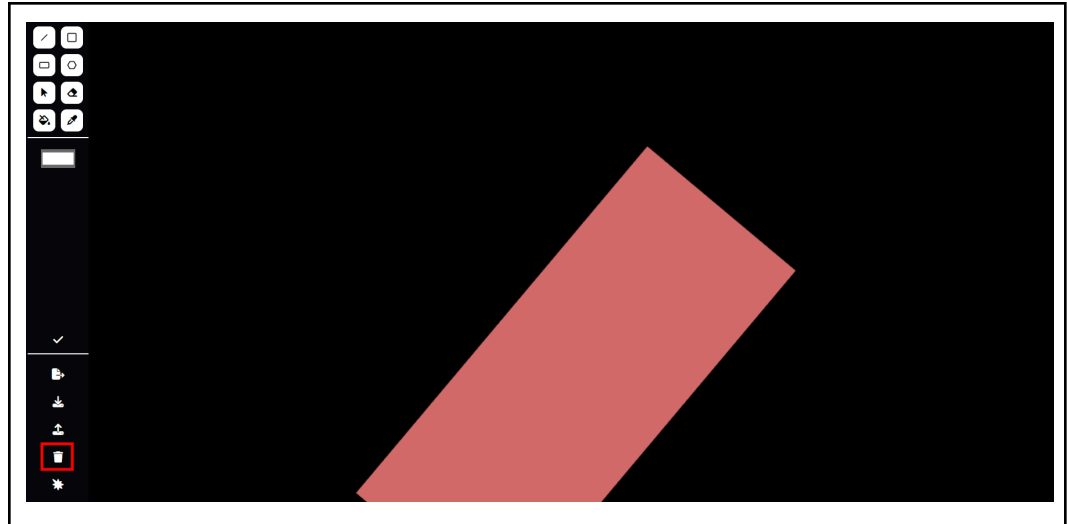
- Mengekspor model yang sudah digambar



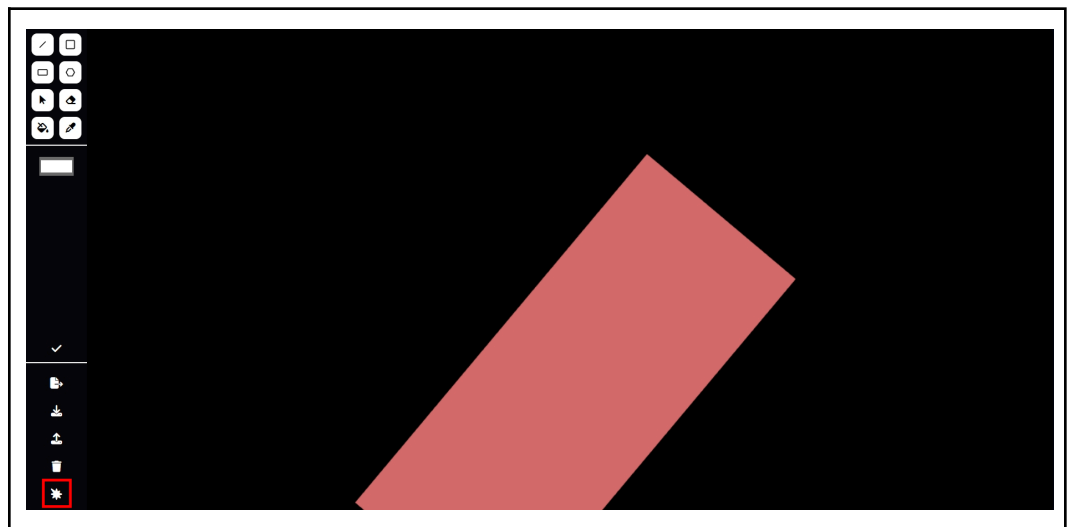
- Mengimport model yang sudah diekspor



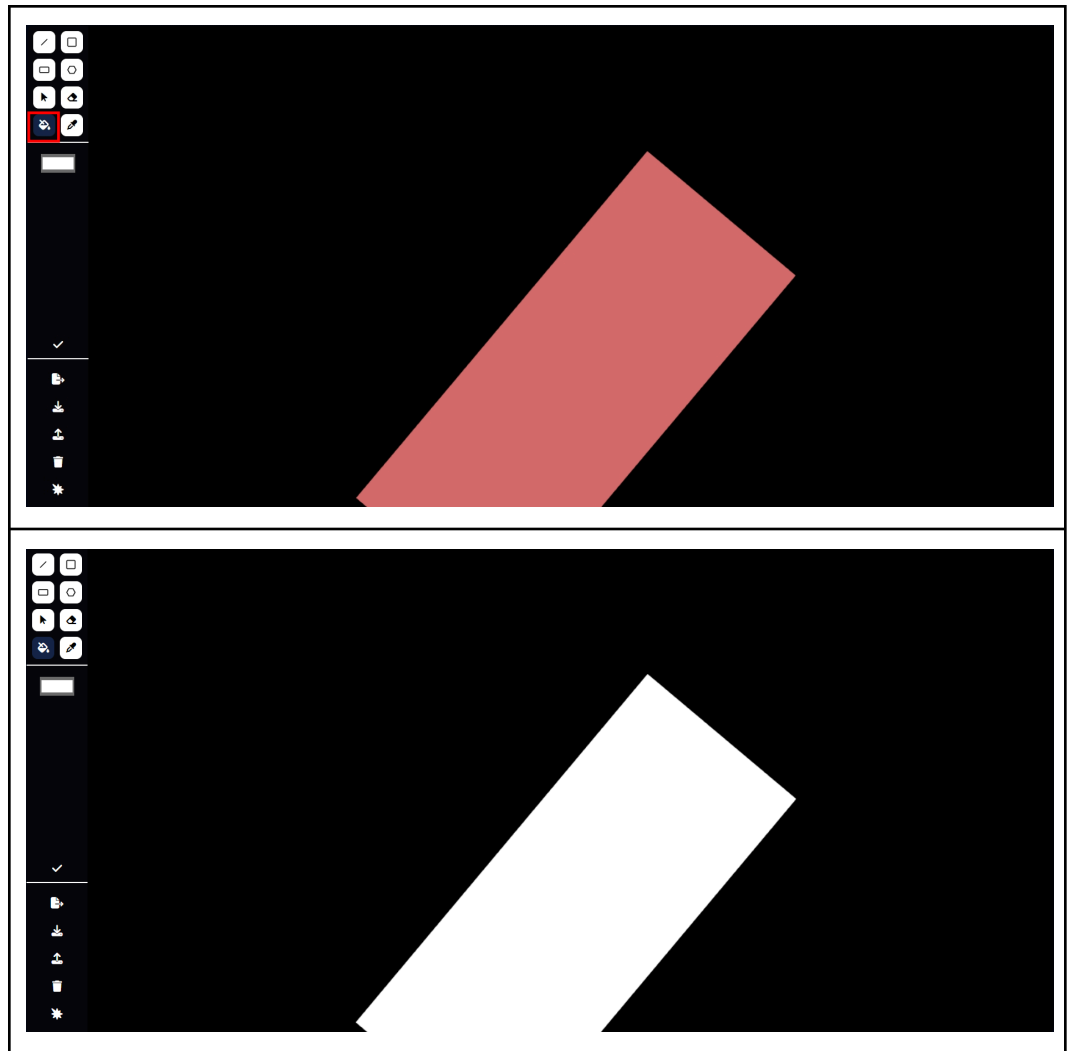
- Menghapus seluruh gambar pada kanvas



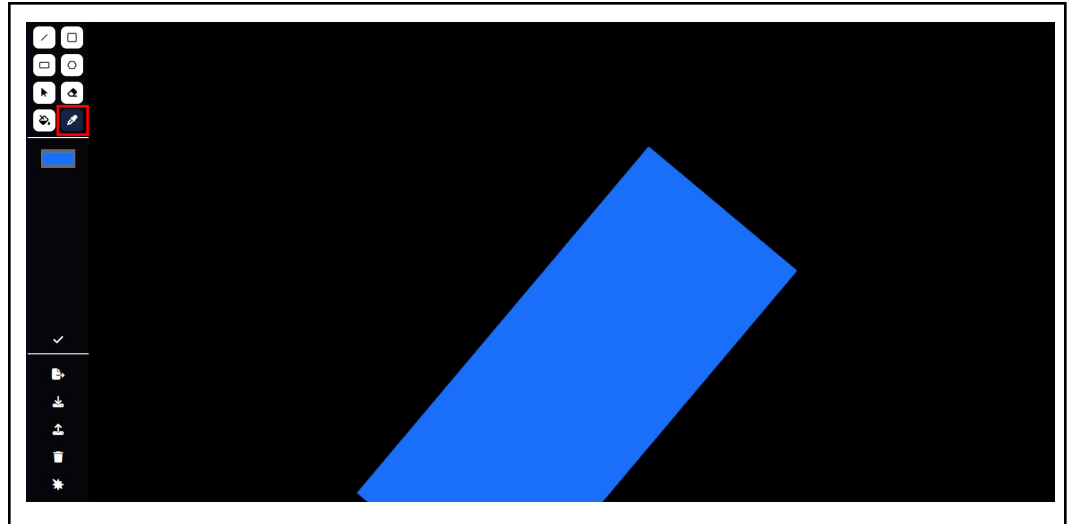
- Menghapus seluruh gambar dengan animasi



- Mengisi warna dengan *fill color*



- Mengambil warna dengan *color picker*. Jika warna sudut dari objek berbeda-beda maka akan diambil rata-ratanya.



- Menyimpan *state* terakhir aplikasi ke dalam *local storage*
Dilakukan secara otomatis.

BAB III

LAMPIRAN