

LAPORAN TUGAS BESAR

IF3260 - GRAFIKA KOMPUTER

KELOMPOK

SAWY



DISUSUN OLEH

YANUAR SANO NUR RASYID 13521110

AHMAD GHULAM ILHAM 13521118

SADDAM ANNAIS SHAQUILLE 13521121

WILLIAM NIXON 13521123

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

TA 2023/2024

## **DAFTAR ISI**

<b>DAFTAR ISI</b>	<b>2</b>
<b>DESKRIPSI</b>	<b>3</b>
<b>HASIL</b>	<b>4</b>
<b>FUNGSIONALITAS</b>	<b>5</b>
1. Models	5
Save & Load	5
Arsip Model Articulated	6
Arsip Model Hollow	7
2. Kamera	10
3. Material & Shading	12
4. Animation	17
5. Component	20
6. Fitur Lanjutan	21
- Dual Canvas	21
- Tweening	21
- GPU Picking	23
- Grayscale Postprocessing	23

## **DESKRIPSI**

Pada tugas ini, WebGL murni digunakan untuk mengimplementasikan sebuah aplikasi web yang memungkinkan pengguna untuk memvisualisasikan model hollow dan articulated pada kanvas. Program dibangun dengan menggunakan React.js, dan tidak menggunakan library eksternal WebGL.

Program dapat memanipulasi beberapa model hollow dan articulated, dengan definisi sebagai berikut:

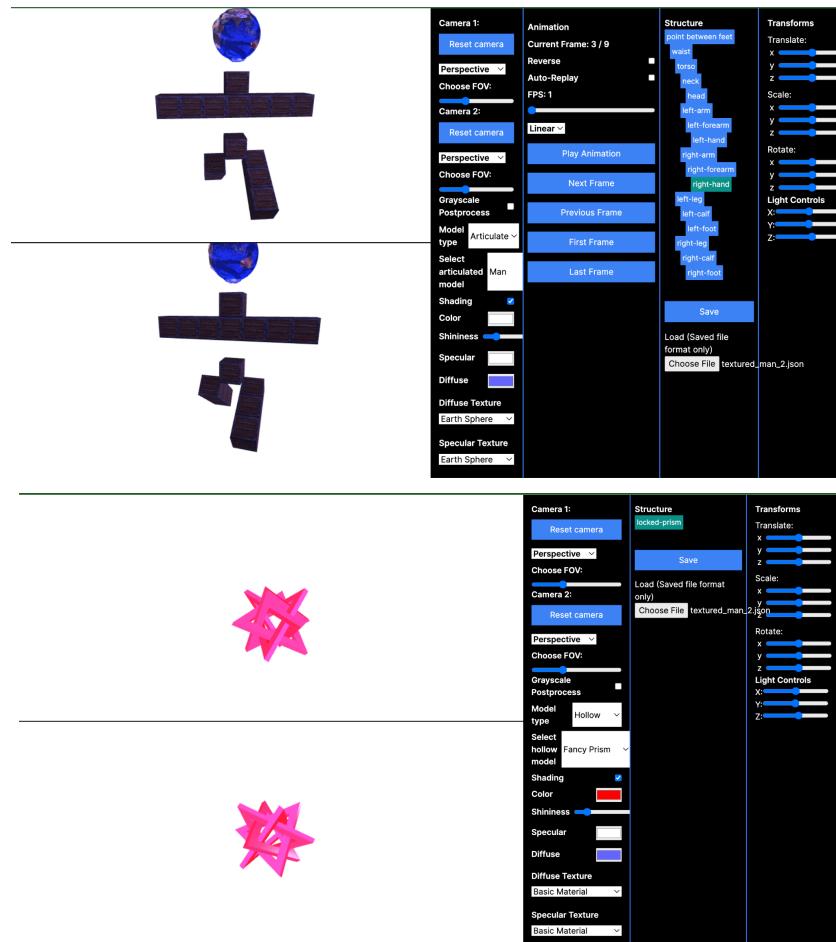
- Model Hollow: Objek ini terdiri dari satu mesh saja, sehingga representasi semua posisi titik ada dalam satu geometri.
- Model Articulated: Model ini terdiri dari beberapa mesh (lebih dari satu) dan memiliki sebuah rig untuk merepresentasikan setiap keterhubungan antar mesh.

Untuk setiap model, dapat dilakukan transformasi geometri seperti translasi, rotasi, dan dilatasi. Transformasi dan manipulasi geometri dapat dilakukan dengan men-select objek terlebih dahulu. Pengguna juga dapat mengubah warna dari model tersebut. Model yang telah digambar pada canvas dapat disimpan dan dimuat kembali dengan menggunakan file JSON. Aplikasi mendukung animasi untuk setiap model articulated yang telah dibuat. Selain itu, terdapat juga kamera dengan 3 jenis proyeksi yang dapat dilakukan *orbit-control*. Terakhir, objek di dalam aplikasi direpresentasikan dengan *scene graph*, sehingga membantu dalam visualisasi hirarki dan hubungan antar objek, serta memudahkan manipulasi dan pengaturan adegan secara keseluruhan

Beberapa fitur lanjutan yang diimplementasikan oleh kelompok kami adalah:

- Post Processing Grayscale: Menambahkan efek grayscale pada kanvas yang sudah di-render.
- Dukungan untuk 2 Canvas: Menampilkan dua kanvas yang bisa digunakan secara bersamaan untuk menggambar dan mengedit model.
- GPU Picker: Memungkinkan pemilihan objek pada kanvas menggunakan GPU
- Tweening: Menambahkan animasi halus (tweening) antar frame untuk transisi yang lebih smooth.

## HASIL



Berikut merupakan hasil cuplikan program kami. Pada gambar diatas, telah digambarkan 2 model yaitu model articulated berbentuk orang yang terdiri atas sphere dan cube yang telah diberi berbagai tekstur. Model ini sedang dianimasikan seolah-olah sedang berjalan. Selain itu, terdapat juga model hollow berwarna pink yang dapat dilihat pada gambar kedua.

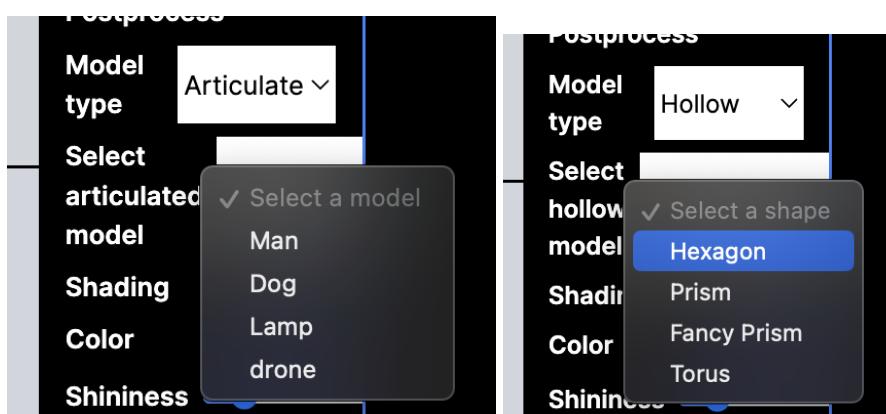
Aplikasi terdiri dari 2 kanvas, yang memiliki beberapa tab. Tab pertama untuk kamera, shading, dan juga pemilihan model. Tab kedua untuk animasi, yang hanya berlaku pada model articulated built-in (agar animasi dapat berjalan dengan lancar). Tab ketiga untuk menampilkan scene-graph dari model, dan juga opsi untuk melakukan save/load model, dan yang terakhir untuk melakukan operasi TRS pada model.

## FUNGSIONALITAS

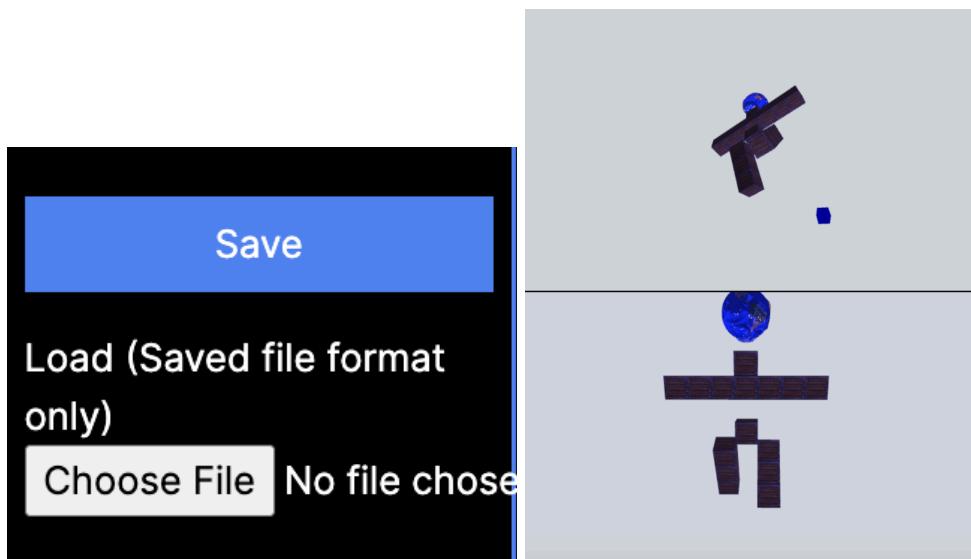
### 1. Models

#### Save & Load

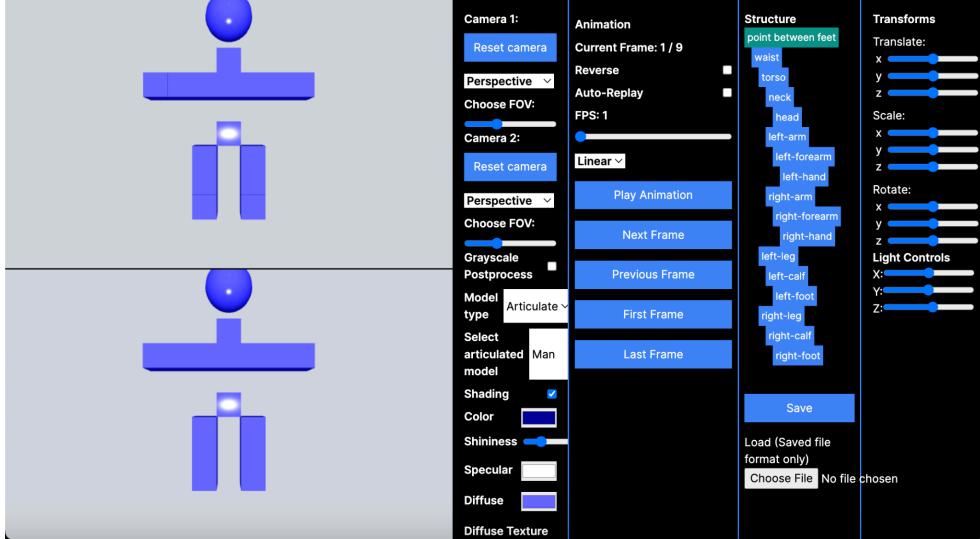
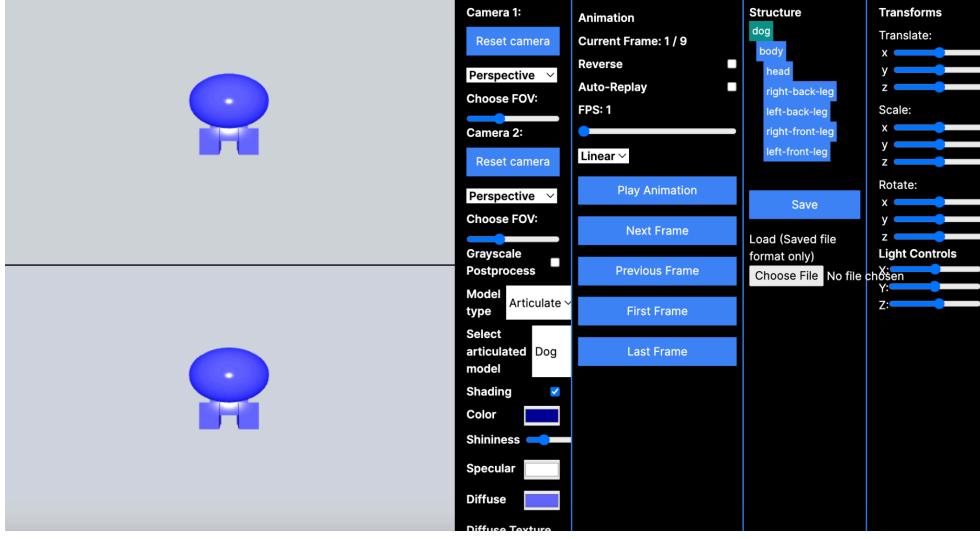
Base models dapat di load menggunakan GUI dari Tab 1 (paling kiri). Pada dropdown, akan tersedia pilihan berbagai template objek hollow dan articulated yang dapat dipilih. Base model ini kemudian akan di load pada canvas, dan siap dilakukan transformasi, ataupun penggantian material seperti pemberian warna, tekstur, dst.



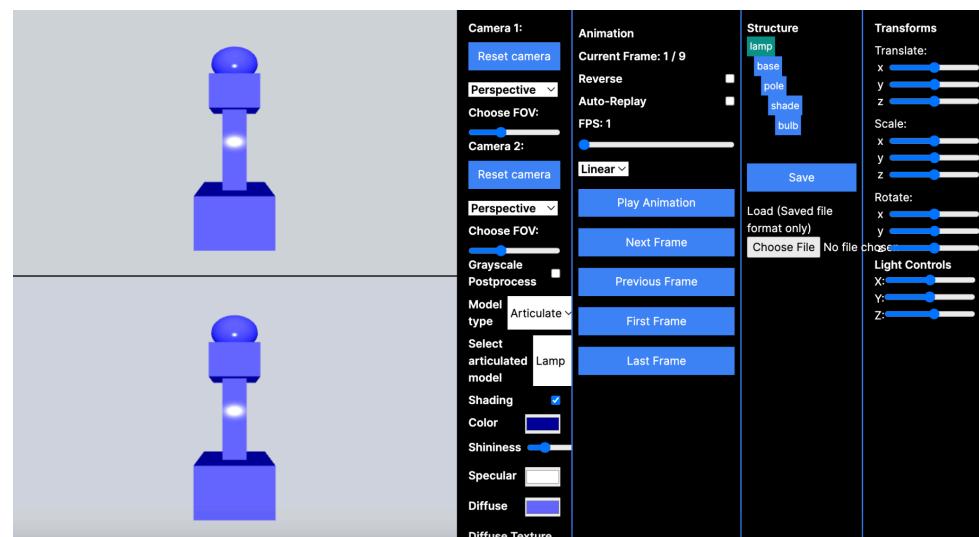
Model pada canvas yang telah diberikan berbagai transformasi dapat disimpan dengan menekan tombol save. Setelah itu, model yang telah di save dapat di load kembali dengan memilih file tersebut. Fitur save mampu menyimpan warna, tekstur, dan transformasi dari objek-objek di dalam canvas.



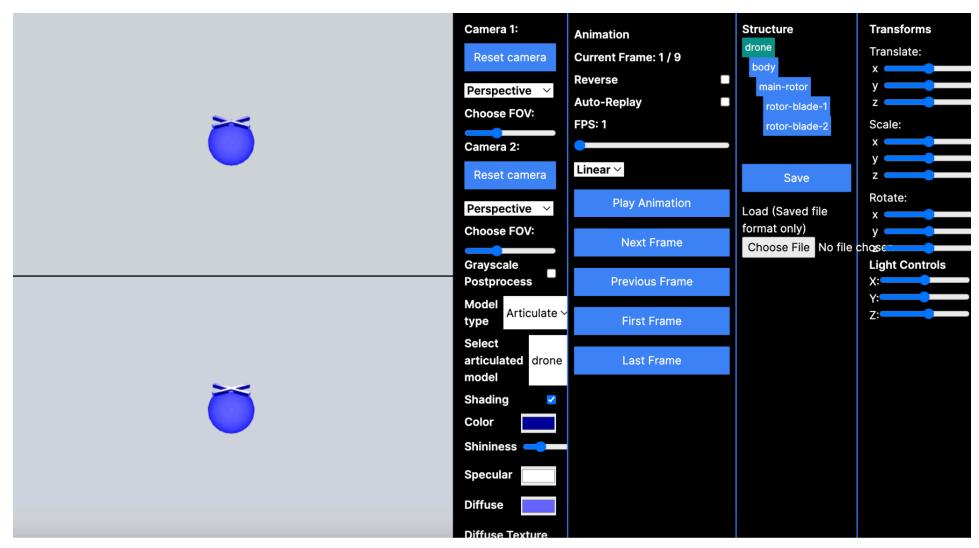
## Arsip Model Articulated

Model	Gambar
Man	
Dog	

Lamp

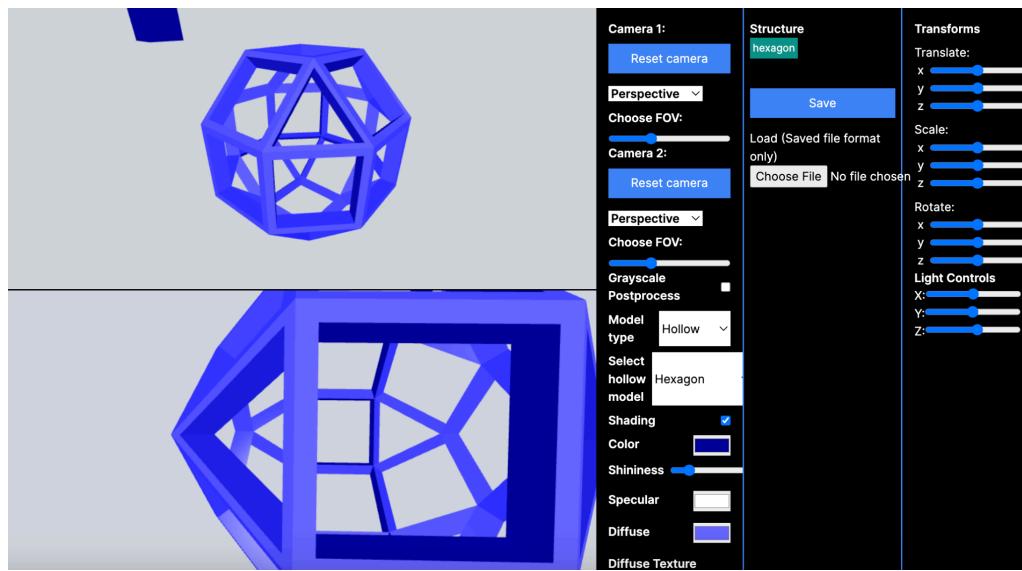


Drone

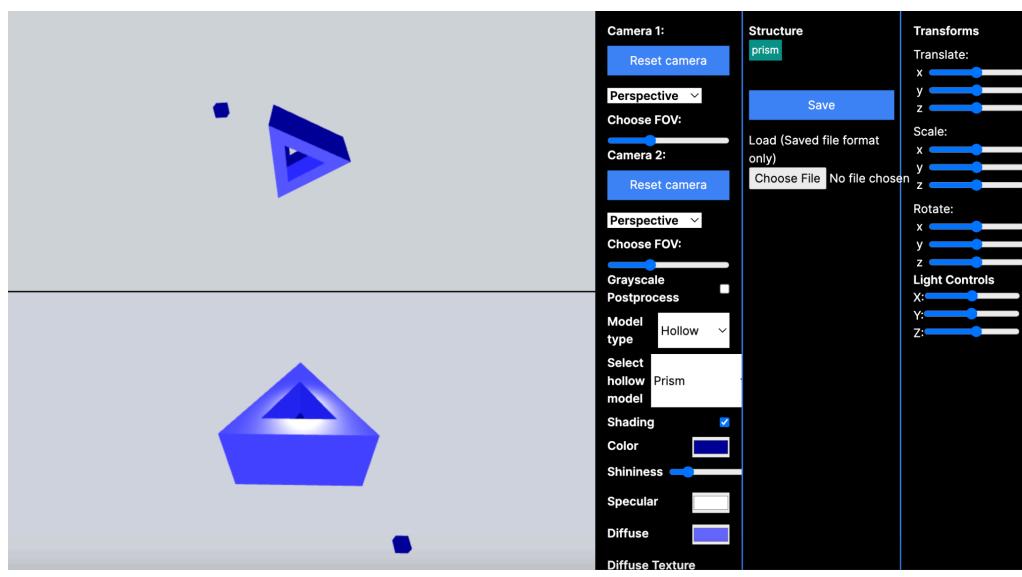


### Arsip Model Hollow

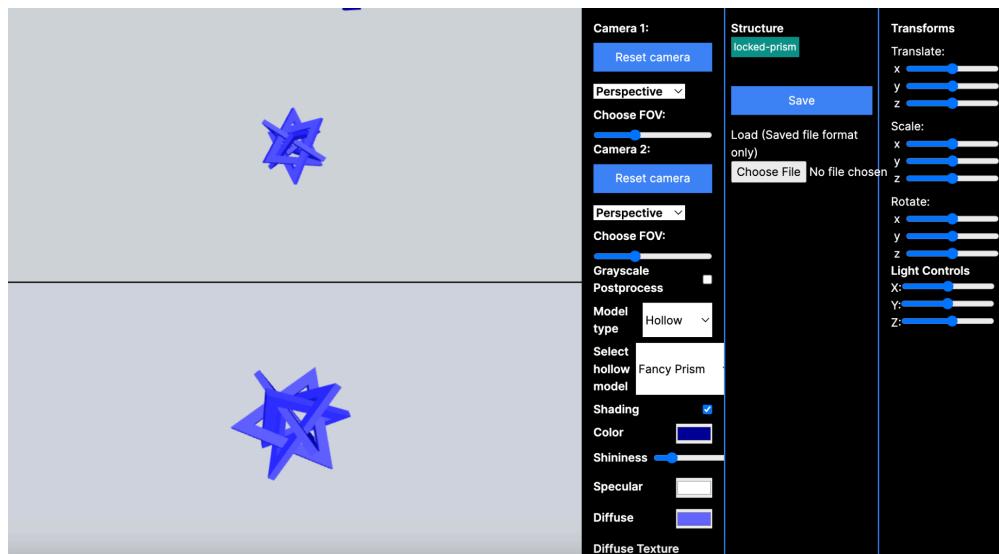
Hexagon



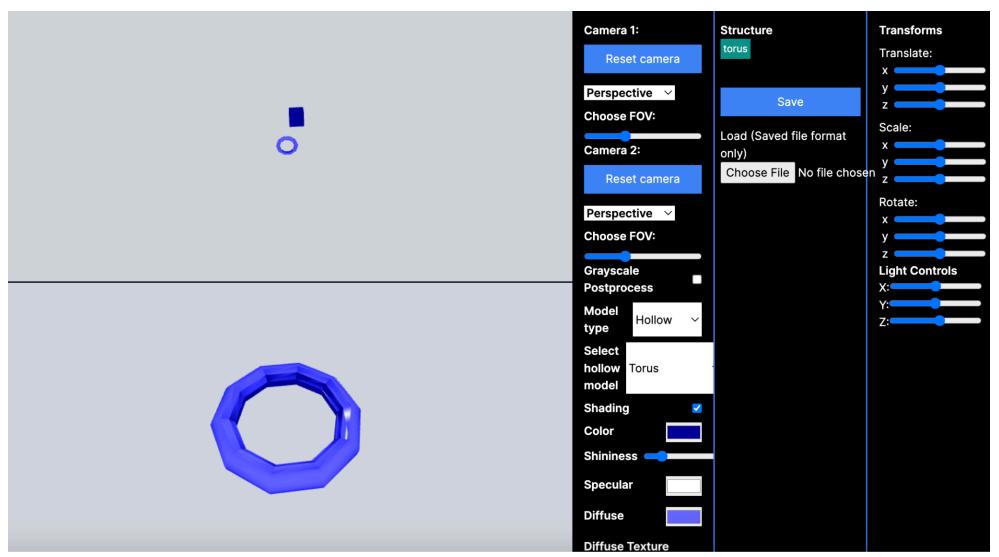
Prism



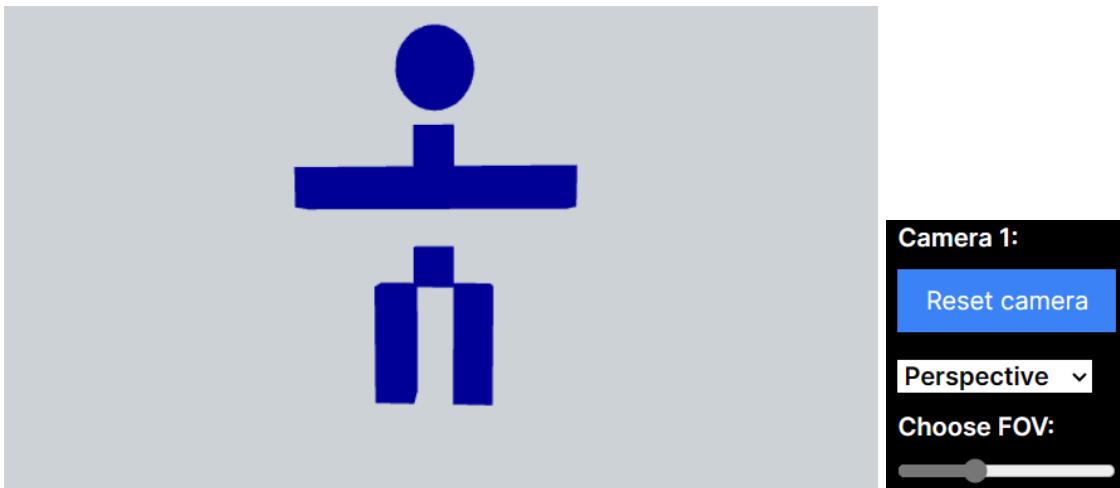
Fancy Prism



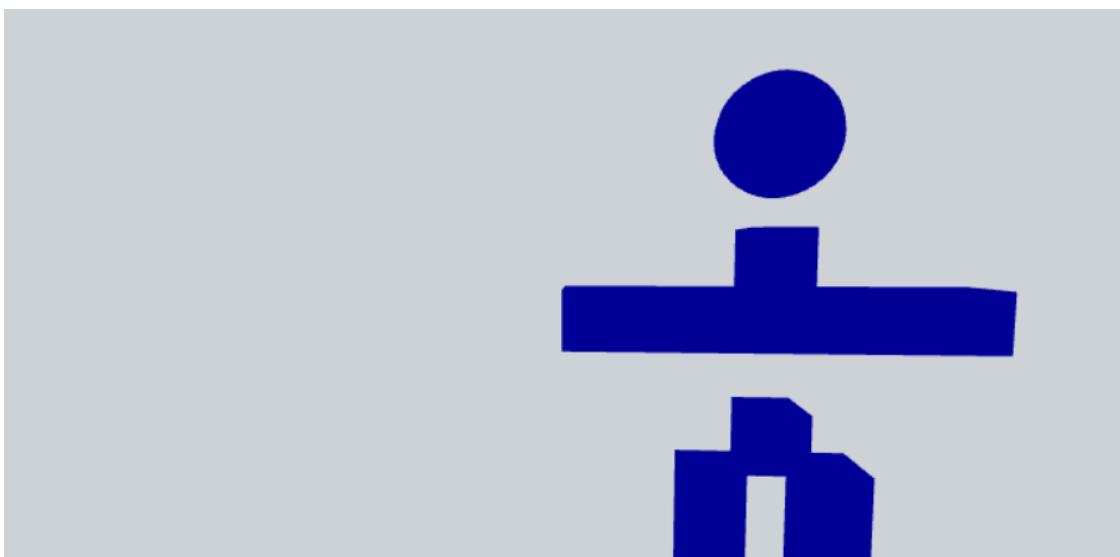
Torus



## 2. Kamera



Tiap kamera dapat memilih jenis projectionnya dan juga besarn FOV (hanya untuk perspective). Selain itu, juga dapat melakukan *reset camera*, menjadi seperti di awal.

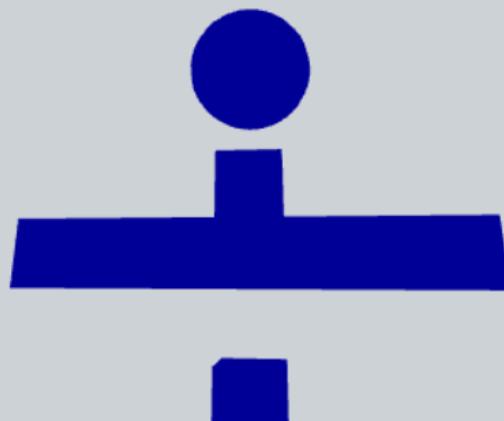


Gunakan WASD untuk menggerakan kamera (translation).

W untuk ke Y positif, S untuk ke Y negatif, A untuk X negatif, dan D untuk ke X positif



*Hold and drag* mouse untuk melakukan orbit control. Gunakan scroll pada mouse untuk merubah radius antara kamera dengan model yang sedang dilihat.



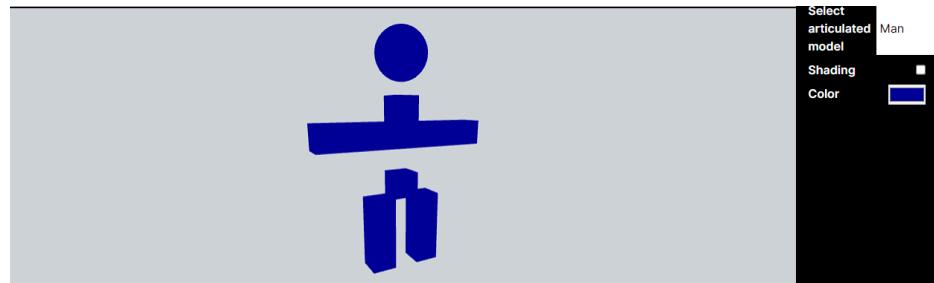
*shift + hold and drag* mouse untuk menggerakan sudut pandang kamera (rotation)

### 3. Material & Shading

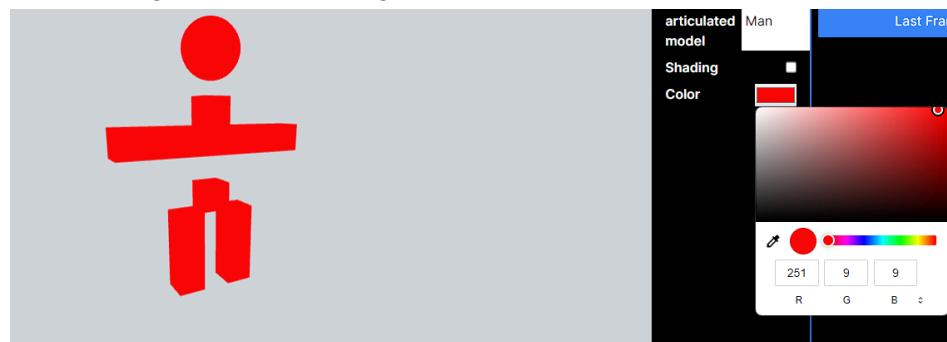
Terdapat dua buah macam material yaitu Basic dan Phong.

- **Basic**

Menampilkan shading berupa warna dasar (ambient light) model.

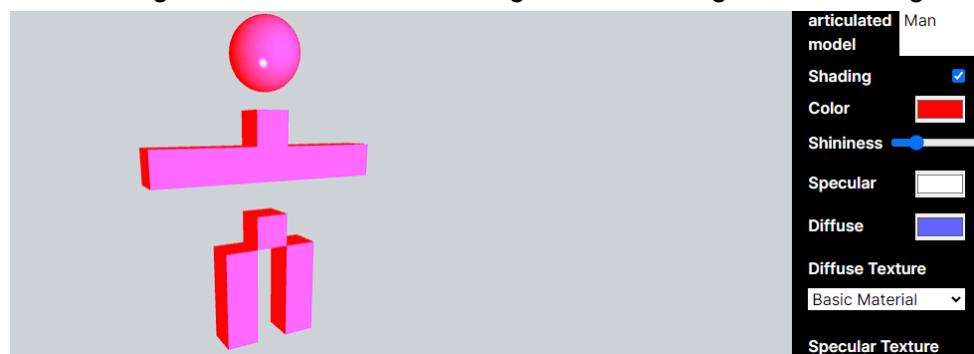


Dapat mengubah warna dengan memilih warna pada menu Color

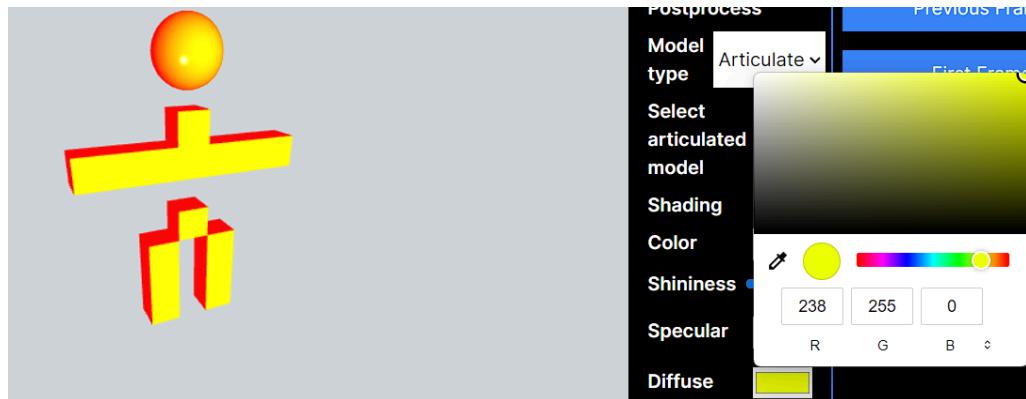


- **Phong**

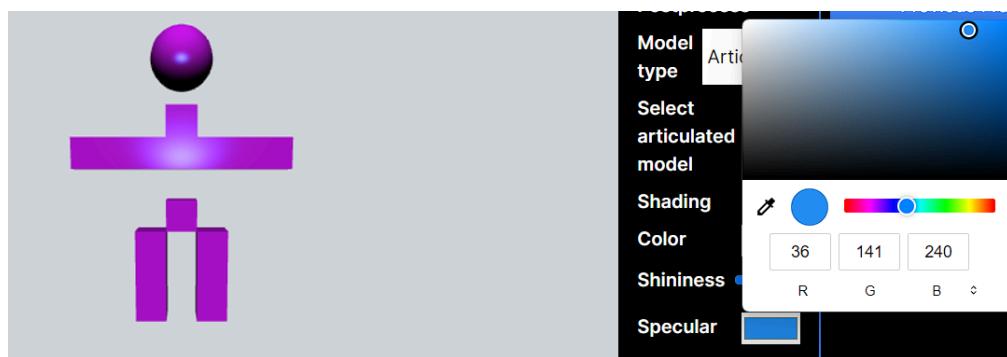
Menampilkan shading yang dipengaruhi oleh *directional light* dengan posisi awal searah dengan kamera. Diaktifkan dengan mencentang menu Shading.



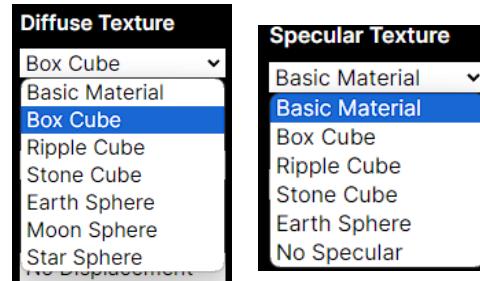
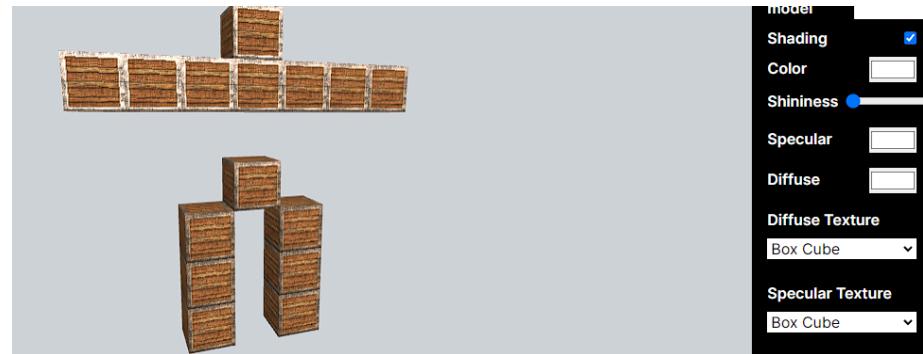
Selain dapat mengubah *ambient color* seperti pada mode Basic, pengguna dapat mengubah *diffuse color* pada menu Diffuse.



Pengguna dapat mengubah *specular color* dan *shininess* pada menu Specular dan Shininess.



Pengguna dapat memakai tekstur *diffuse* dan juga tekstur *specular* pada menu Diffuse Texture dan Specular Texture dengan memilih tekstur yang tersedia.

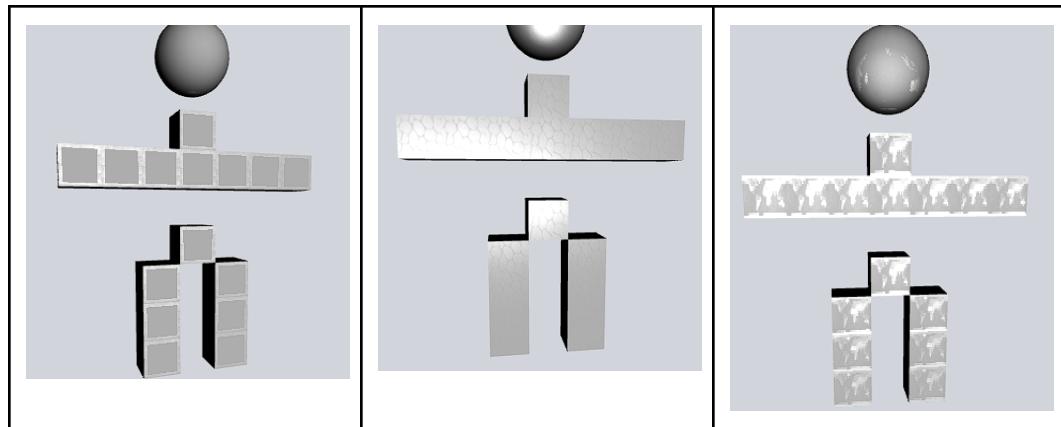


### Diffuse Texture

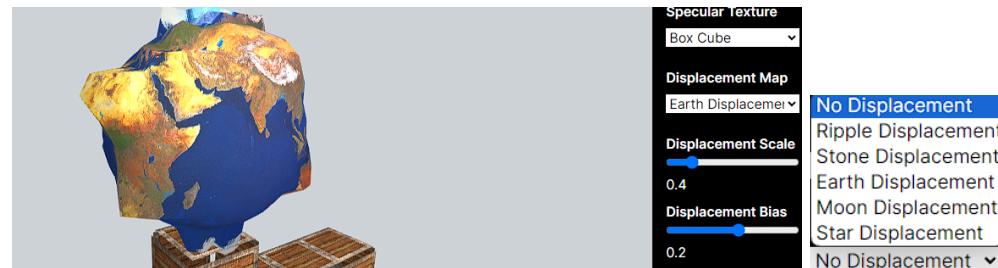
Box Cube	Stone Cube	Moon Sphere

### Specular Texture

Box Cube	Stone Cube	Earth Sphere
----------	------------	--------------



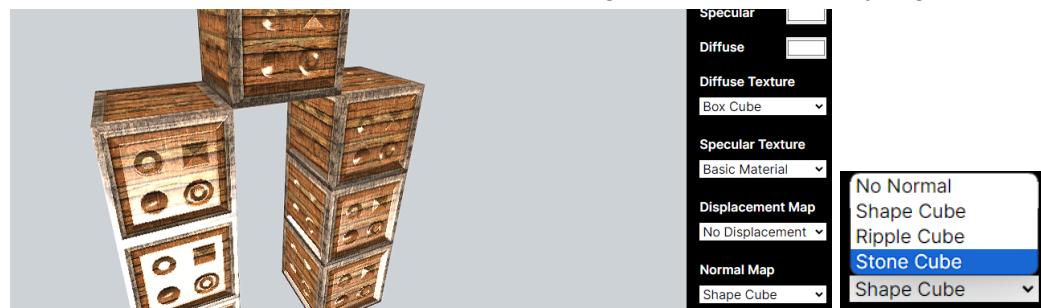
Pengguna dapat memilih tekstur *displacement map* pada menu Displacement Map serta dapat mengatur *scale* dan *bias* pada menu Displacement Scale dan Displacement Bias.



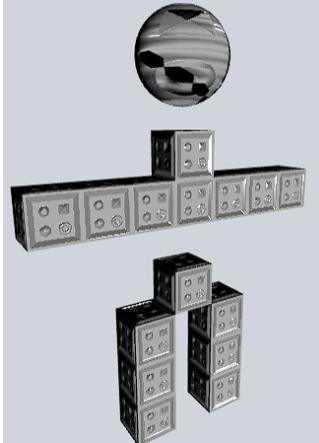
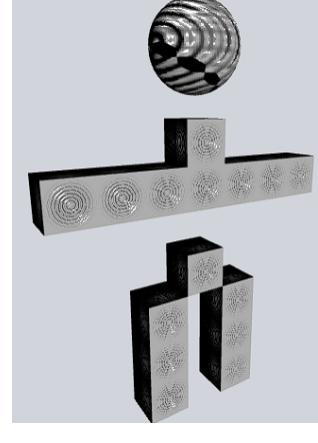
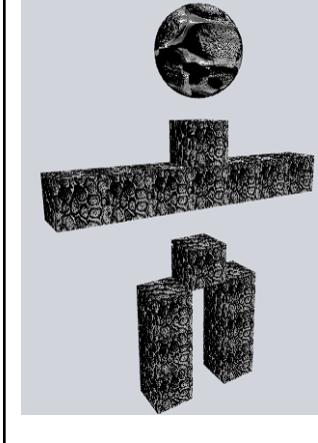
### Displacement Map

Ripple Displacement	Earth Displacement	Star Displacement

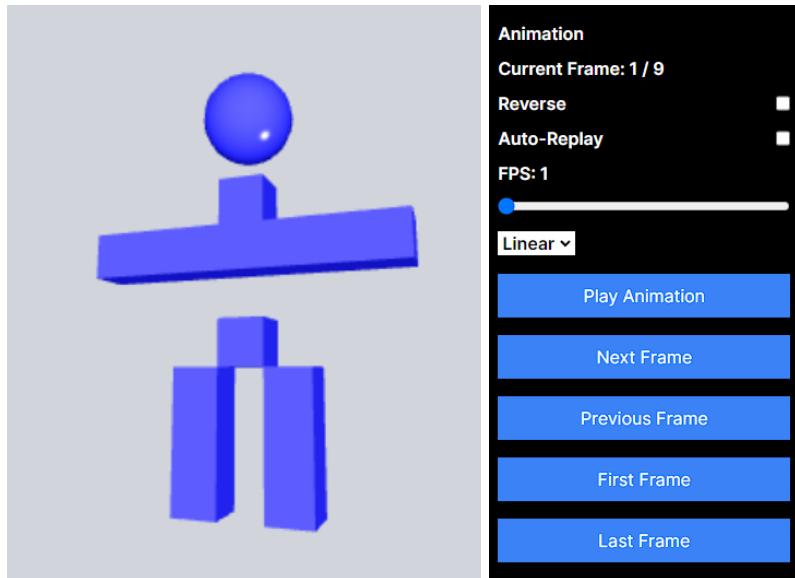
Pengguna dapat memilih tekstur normal map pada menu Normal Map untuk memakai tekstur normal map pada model dengan memilih tekstur yang tersedia.



### Normal Map

Shape Cube	Ripple Cube	Stone Cube
 A screenshot of a 3D scene showing a horizontal row of metallic cubes with circular patterns and a smaller stack of cubes below it. Above the row is a circular preview image showing a smooth, featureless surface.	 A screenshot of a 3D scene showing a horizontal row of metallic cubes with circular patterns and a smaller stack of cubes below it. Above the row is a circular preview image showing concentric circular ripples on a surface.	 A screenshot of a 3D scene showing a horizontal row of metallic cubes with circular patterns and a smaller stack of cubes below it. Above the row is a circular preview image showing a highly detailed, rough stone-like texture.

#### 4. Animation



Setiap model articulated disertai dengan logika animasi yang tersimpan dalam file berbeda antara file model dan file animasi. Setiap animasi terdiri atas 9 frame yang menyimpan informasi TRS (Translation, Rotation, Scale) dari node-node yang ada pada model.



Terdapat indikator frame yang dijalankan sekarang beserta total frame animasi pada bagian paling atas. Indikator ini juga dapat membantu pengguna mengetahui frame yang sedang di-render pada canvas saat ini ketika menggunakan tombol kontrol frame.

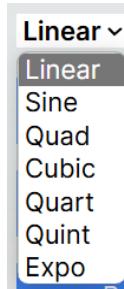


Terdapat dua toggle yang dapat dicentang oleh pengguna, yaitu:

- Reverse: Pengguna dapat mengubah arah atau urutan pemutaran frame animasi. Secara default animasi dijalankan menaik (dari frame pertama menuju frame terakhir), sedangkan ketika toggle Reverse dicentang animasi akan dijalankan menurun (dari frame terakhir menuju frame pertama).
- Auto-Replay: Pengguna dapat mengubah looping pemutaran animasi. Secara default animasi akan berhenti pada frame terakhir atau frame pertama (bergantung pada toggle Reverse), sedangkan ketika toggle Auto-Replay dicentang animasi akan dijalankan secara terus menerus (frame otomatis kembali ke frame pertama atau frame terakhir dan dijalankan kembali menuju frame pertama atau terakhir, bergantung pada toggle Reverse).



Terdapat slider FPS yang dapat digeser oleh pengguna untuk mengatur berapa banyak frame yang dapat ditampilkan dalam setiap detiknya. Secara default nilai FPS adalah 1 dan nilai maksimal dari slider untuk FPS adalah 30.



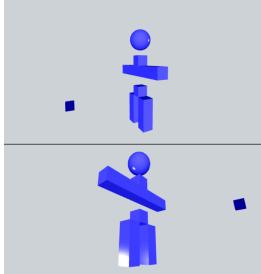
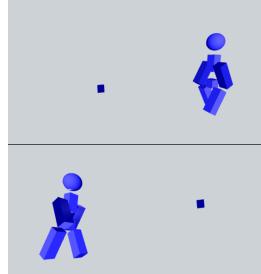
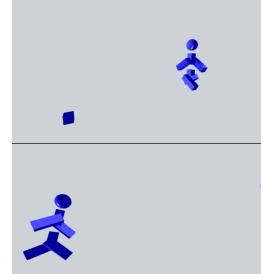
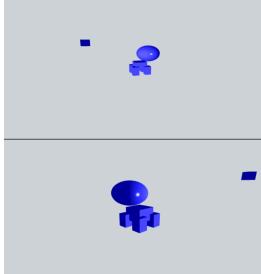
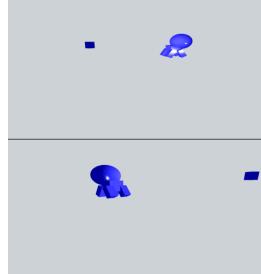
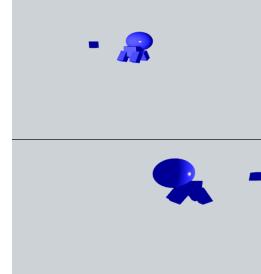
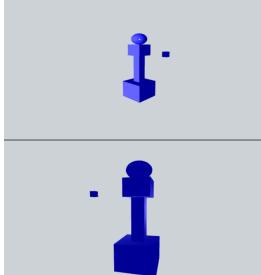
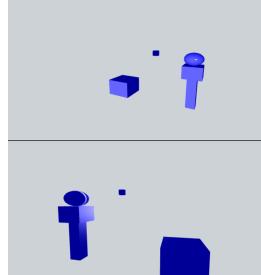
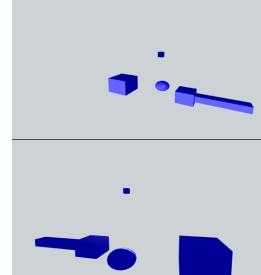
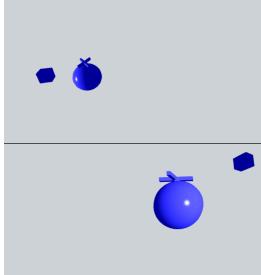
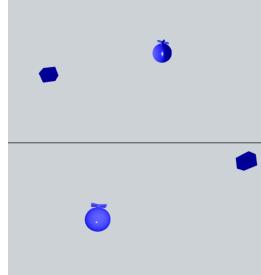
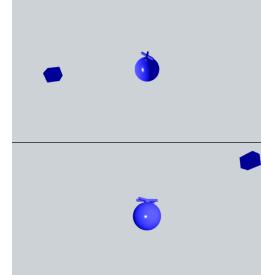
Terdapat dropdown easing function yang dapat digunakan oleh pengguna untuk memilih jenis fungsi easing function yang digunakan pada tweening antar frame animasi. Secara default digunakan fungsi Linear dan terdapat 6 fungsi lain yang dapat dipilih oleh pengguna, yaitu Sine, Quad, Cubic, Quart, Quint, dan Expo. Penjelasan lebih lanjut mengenai tweening dapat dilihat pada bagian fitur lanjutan.



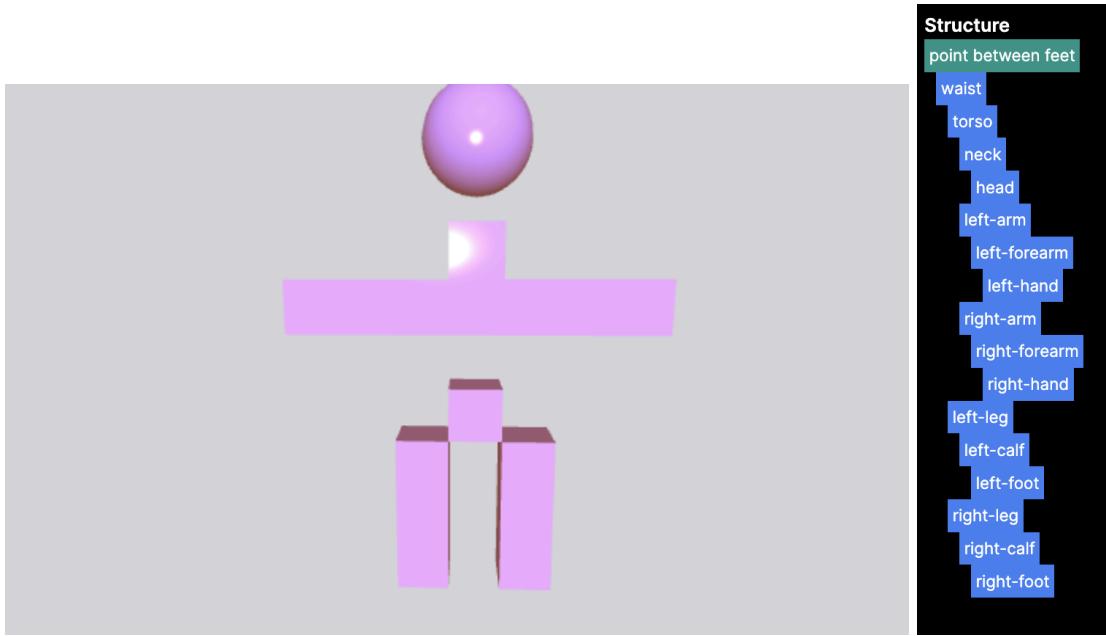
Terdapat lima button yang dapat digunakan oleh pengguna untuk mengontrol frame animasi, yaitu:

- Play Animation / Pause Animation: Pengguna dapat menjalankan atau menghentikan animasi pada model. Secara default, animasi tidak dijalankan sehingga pengguna harus menekan tombol Play Animation untuk menjalankan animasi. Untuk menghentikan animasi, pengguna dapat menekan tombol Pause Animation.
- Next Frame: Pengguna dapat memajukan frame animasi model yang di-render pada canvas saat ini sejauh satu frame tanpa dipengaruhi oleh toggle Reverse.
- Previous Frame: Pengguna dapat memundurkan frame animasi model yang di-render pada canvas saat ini sejauh satu frame tanpa dipengaruhi oleh toggle Reverse.
- First Frame: Pengguna dapat mengubah frame animasi model yang di-render pada canvas saat ini menjadi frame pertama animasi tanpa dipengaruhi toggle Reverse.

- Last Frame: Pengguna dapat mengubah frame animasi model yang di-render pada canvas saat ini menjadi frame terakhir animasi tanpa dipengaruhi toggle Reverse.

Man	Frame	1	4	7
	Canvas			
Dog	Frame	2	6	9
	Canvas			
Lamp	Frame	1	3	6
	Canvas			
Drone	Frame	4	7	1
	Canvas			

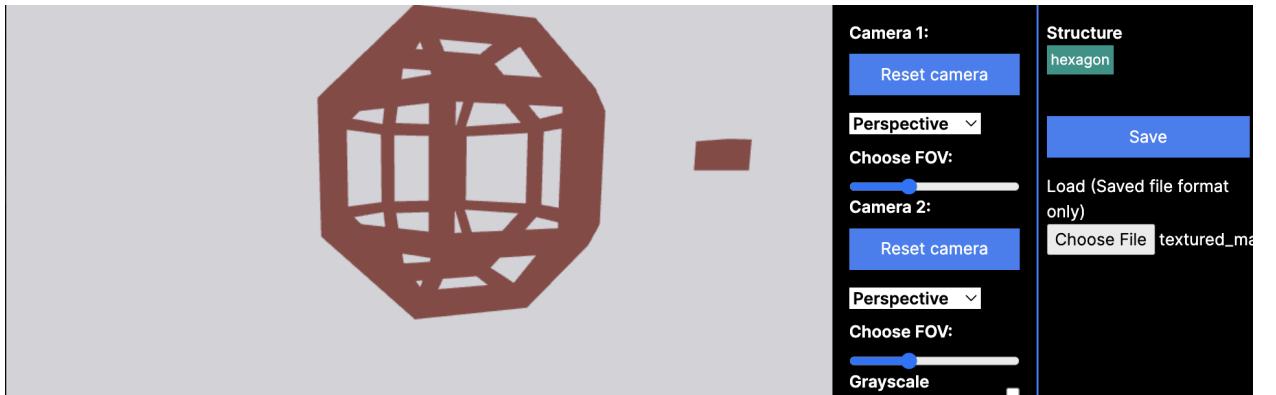
## 5. Component



Semua model articulated memiliki scene graph yang akan ditampilkan pada tab canvas. Komponen yang sedang terseleksi akan memiliki warna hijau pada scene graph tersebut. Scene graph menunjukkan keterhubungan parent-child dari setiap mesh. Operasi TRS dan berbagai setter lainnya yang dilakukan pada parent akan mempengaruhi komponen anak.



Pada contoh ini, dilakukan rotasi translasi dan scaling pada bagian leher dari model. Sebagai orang tua dari mesh head, maka komponen head juga akan terpengaruhi oleh perubahan TRS orangtuanya, yaitu neck.

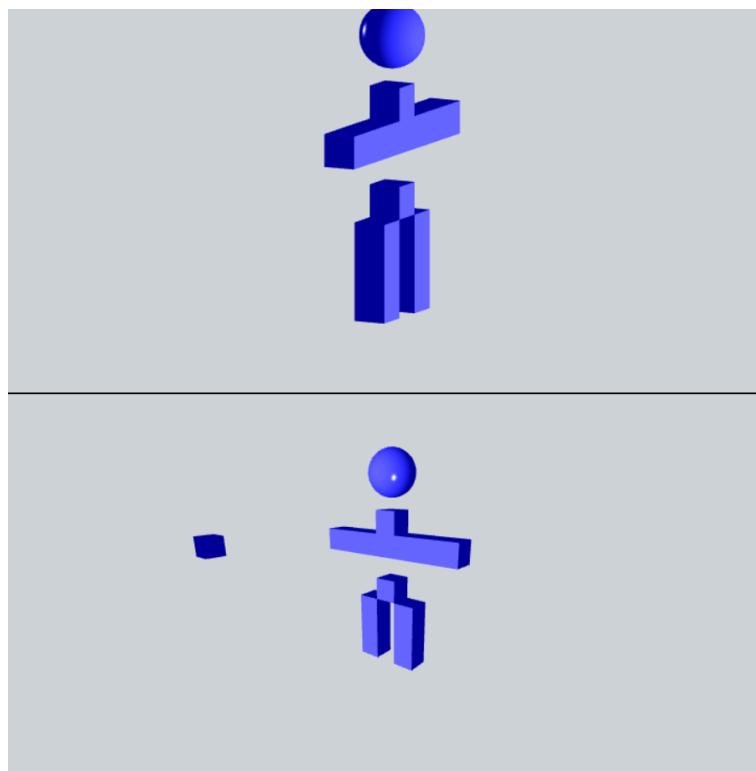


Objek hollow tidak memiliki anak pada bagian struktur komponennya. Operasi TRS pada objek-objek lain pada canvas dapat dilakukan dengan cara berikut:

- Cahaya, hanya tersedia 1 cahaya global yang dapat dimodifikasi lewat slider pada bagian kanan.
- Camera, terdapat 2 camera, 1 untuk setiap canvas. TRS pada kamera dapat dilakukan dengan orbit control dan WASD. Lihat bagian fungsionalitas **Kamera** untuk cara pengoperasian yang lebih detail.

## 6. Fitur Lanjutan

- Dual Canvas



Fitur dual canvas memungkinkan pengguna untuk melihat objek menggunakan 2 *point of view* kamera yang berbeda. Tiap kamera dapat diatur supaya memiliki letak, arah, dan tipe proyeksi yang berbeda-beda. Tiap objek pada kedua canvas merupakan objek yang sama sehingga objek di kedua canvas tersebut sinkron. Pada kamera yang pertama dapat juga letak kamera yang kedua.

Pada fitur lanjutan ini, objek scene yang dirender merupakan objek yang sama. Akan tetapi, terdapat perbedaan yaitu terdapat objek kamera 1 dan kamera 2 yang juga akan dirender bersama dengan objek utama. Setiap kali ada perubahan, letak objek kamera 1 dan 2 akan disimpan dan akan dirender pada canvas yang bersesuaian sehingga objek kamera 1 dapat terlihat oleh kamera 2 dan sebaliknya.

#### - Tweening

Fitur Tweening memungkinkan transisi antar frame pada animasi untuk berjalan dengan lebih halus dengan memanfaatkan informasi delta TRS (perubahan translasi, rotasi, dan scale antar frame), delta second (progress absolute dalam satuan detik, rentang 0 sampai dengan 1), dan easing function (jenis fungsi interpolasi yang digunakan). Perubahan implementasi dari fitur wajib Animation terdapat pada method update dari kelas Animator, yaitu jika sebelumnya update hanya dilakukan ketika delta second sudah melebihi 1 (dengan kata lain frame harus berpindah sejauh 1 ke depan atau belakang), pada fitur lanjutan tweening update tetap dilakukan meskipun delta second belum melebihi 1.

Untuk mencari seberapa jauh besaran TRS yang harus di-render pada canvas, terlebih dahulu dicari delta TRS antara kedua frame yang bersebelahan. Kemudian, delta second yang belum mencapai 1 menjadi faktor pengali dari delta TRS. Terakhir, fungsi interpolasi yang dipilih akan menentukan laju perubahan berdasarkan nilai delta second. Secara perlahan frame akan berpindah menuju frame selanjutnya atau sebelumnya dengan lebih halus. Dengan begitu, jumlah frame animasi tetap sama, tetapi animasi seolah-olah terkesan lebih halus akibat adanya sisipan “frame baru” meskipun belum waktunya berpindah frame (delta second sudah melebihi 1).

Pada implementasi fitur lanjutan tweening, digunakan varian akselerasi In dengan 6 jenis fungsi interpolasi, yaitu Sine, Quad, Cubic, Quart, Quint, dan Expo. Berikut merupakan contoh perhitungan dengan menggunakan nilai awal a, nilai akhir b, dan perubahan detik t.

Jenis	Ilustrasi	Contoh Perhitungan ( $a = 1$ , $b = 2$ , $t = 0.5$ )
Linear (default)	-	$a + (b - a) * t$

		= 1.5
easeInSine	 easeInSine	$a + (b - a) * \text{Math.sin}(t * \text{Math.PI} / 2)$ $= 1.7071$
easeInQuad	 easeInQuad	$a + (b - a) * \text{Math.pow}(t, 2)$ $= 1.25$
easeInCubic	 easeInCubic	$a + (b - a) * \text{Math.pow}(t, 3)$ $= 1.125$
easeInQuart	 easeInQuart	$a + (b - a) * \text{Math.pow}(t, 4)$ $= 1.0625$
easeInQuint	 easeInQuint	$a + (b - a) * \text{Math.pow}(t, 5)$ $= 1.03125$
easeInExpo	 easeInExpo	$a + (b - a) * \text{Math.pow}(2, 10 * (t - 1))$ $= 1.03125$

#### - GPU Picking

Fitur GPU picking memungkinkan pengguna untuk memilih komponen pada scene graph dengan mengklik komponen tersebut pada kanvas. Fitur ini memanfaatkan penyimpanan data pada GPU, di mana koordinat mouse akan dipetakan kepada sebuah objek lewat ID yang disimpan lewat warna pada GPU.

Pada fitur lanjutan ini, setiap objek di render dengan warna yang akan menjadi ID pengenal. Rendering ini akan dilakukan pada sebuah tekstur frame buffer tersendiri di luar layar, sehingga pengguna tidak akan melihat warna yang aneh. Ketika kanvas di klik, koordinat kanvas akan digunakan untuk melihat warna yang telah tersimpan di GPU pada koordinat tersebut, dan akan mengembalikan warna /ID objek yang terletak pada piksel tersebut.

- **Grayscale Postprocessing**

Pada fitur ini, diterapkan efek grayscale pada gambar yang telah dirender. Untuk melakukan hal ini, dilakukan rendering seluruh scene ke dalam framebuffer terlebih dahulu, kemudian memproses hasil render menggunakan shader tersendiri untuk menerapkan efek yang diinginkan, yaitu mengkonversinya ke grayscale.

Langkah-langkah Implementasi:

1. Inisialisasi Texture & Renderbuffer: Pertama-tama, sebuah texture dibuat untuk menyimpan hasil render. Tekstur ini akan digunakan sebagai input untuk proses selanjutnya. Renderbuffer digunakan untuk menyimpan informasi kedalaman (depth) dari hasil render. Digunakan untuk menjaga informasi tentang kedalaman objek saat scene dirender ke framebuffer.
2. Render ke Framebuffer: Framebuffer akan digunakan untuk mengarahkan rendering ke texture yang telah disiapkan. Semua operasi rendering berikutnya akan menulis ke texture target ini, bukan ke layar langsung. Scene kemudian dirender ke framebuffer. Menggambar scene menggunakan program shader utama, hasil render ini akan disimpan ke dalam framebuffer.
3. Mengaplikasikan Shader Postprocessing: Setelah scene dirender ke framebuffer, kita dapat menggunakan shader preprocessing pada gambar tersebut. Untuk menerapkan shader post processing pada seluruh gambar, sebuah quad layar penuh (persegi empat) digambar. Posisi dan ukuran quad ini diatur sedemikian rupa sehingga menutupi seluruh layar, memungkinkan shader untuk memproses setiap piksel dari texture yang dirender.
4. Grayscale: Shader fragment bertanggung jawab untuk memproses setiap piksel dari texture yang di render. Dalam kasus ini, shader menggunakan rumus khusus untuk mengubah warna piksel menjadi grayscale. Rumus tersebut menghitung nilai grayscale dengan menggabungkan komponen warna RGB menggunakan rumus:
  - Merah: 0.2126
  - Hijau: 0.7152

- Biru: 0.0722

Beberapa variabel yang dipakai shader yaitu:

- Uniforms: `u_texture` adalah uniform yang merujuk ke texture yang telah dirender ke framebuffer.
- Attributes: `a_position` digunakan untuk menentukan posisi dari quad layar penuh yang akan dirender.
- Textures: `targetTexture` adalah texture yang digunakan sebagai target rendering untuk menyimpan hasil awal sebelum diproses oleh shader postprocessing.