

3D WEBGL ARTICULATED MODEL

LAPORAN TUGAS

diajukan untuk memenuhi persyaratan
IF3260 Grafika Komputer



Oleh
Kelompok 8

Fransiskus Davin Anwari	13520025
Timothy Stanley Setiawan	13520028
Jeremy S.O.N. Simbolon	13520042
Christopher Jeffrey	13520055

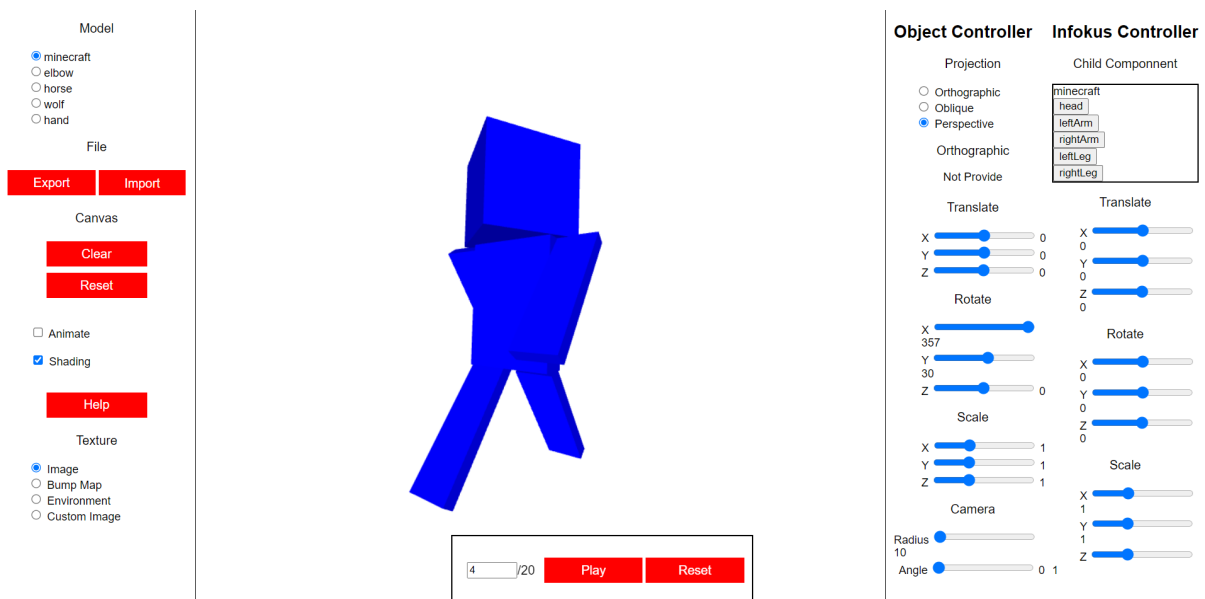
TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
BANDUNG
2023

A. Deskripsi

Tugas ini mewajibkan kami untuk membuat program 3D WebGL *Articulated Model* dengan ketentuan sebagai berikut,

1. Implementasi umum program dilakukan menggunakan WebGL murni tanpa library/framework tambahan. Jika memerlukan fungsi-fungsi yang ada di library wrapper, dapat dibuat sendiri.
2. Membuat X buah model objek articulated dimana X = jumlah anggota kelompok. Setiap anggota wajib menyumbang satu *articulated model* yang berbeda.
3. Definisi semua model disimpan dalam satu file yang mudah dibuka dan diedit (berisi daftar koordinat dan warna setiap komponen dari *articulated model*, serta hubungan transformasi *parent-child* masing-masing komponen). Struktur isi file bisa juga tidak mengikuti hint di atas, asalkan dapat dijelaskan dan diimplementasikan fitur-fitur spesifikasi wajib lainnya.
4. Membuat kontrol *view model* umum sehingga bisa
 - mengubah jenis proyeksi untuk menampilkan *articulated model* (*orthographic*, *oblique*, atau *perspective*);
 - melakukan rotasi, translasi dan penskalaan dari *articulated model*;
 - mengubah jarak (radius) kamera *view* untuk mendekat atau menjauh dari model serta menggerakkan kamera untuk mengitari model-model;
 - me-reset ke *default view controls*;
 - me-enable dan *disable shading*; dan
 - mengubah tekstur permukaan model antara *bump*, *refelctive*, dan *custom*.
5. Membuat kumpulan logika animasi yang dapat diaplikasikan pada *articulated model*. Info animasi ini dapat disimpan pada file yang sama dengan *articulated model*, atau dalam *file* terpisah. Animasi ini dapat kemudian dimainkan. Setiap anggota kelompok bertanggung jawab membuat minimal 1 animasi yang menemani model buatannya, minimal berisi 9 *frame*.
6. Menambahkan kontrol untuk semua komponen yang ada dalam *articulated model*. Kontrol ini berupa transformasi yang dinyatakan dalam hubungan *parent-child*. Setidaknya dua kontroler dari transformasi (*scale*, *rotate*, *translate*) harus tersedia untuk setiap komponen model. Ini juga berarti jumlah *slider* berubah ketika model yang di-load berubah, berbanding lurus pada jumlah komponen dalam model.
7. [BONUS] Menampilkan struktur pohon dari komponen yang dapat diklik untuk menandakan sekarang *user* sedang mengontrol komponen tersebut. Hal ini juga menyebabkan jumlah slider dari kontroler komponen tidak bertambah atau berkurang.
8. [BONUS] Menambahkan kontrol transformasi serta *view*-nya untuk komponen terpilih. Kontrol mencakupi fitur yang tertera pada kontrol *view model* umum.
9. [BONUS] Menambahkan kontrol transformasi serta *view*-nya untuk *subtree* komponen terpilih. *Subtree* merupakan semua komponen yang merupakan keturunan dari komponen terpilih. Kontrol mencakupi fitur yang tertera pada kontrol *view model* umum. Pengubahan dari transformasi akan memberikan efek pada seluruh komponen keturunannya.

B. Hasil



C. Manual

Program terdiri dari tiga bagian utama, yakni bagian *sidebar* kiri, bagian kanvas tengah, dan bagian *sidebar* kanan. Bagian *sidebar* kiri merupakan kontroler kanvas yang dapat digunakan untuk melakukan perubahan pada kanvas. Bagian kanvas tengah merupakan tempat model digambarkan ke layar yang juga mencakup kontroler animasi model. Bagian *sidebar* kanan merupakan kontroler model yang dapat digunakan untuk melakukan translasi, rotasi, dan penskalaan model terhadap konfigurasi kamera menggunakan mode proyeksi yang diinginkan.

Pada *sidebar* kiri, terdapat lima pilihan model yang termasuk dalam program. Pengguna dapat memilih salah satu dari kelima model tersebut dengan mengeklik model yang diinginkan. Selain itu, terdapat tombol *export* dan *import* yang dapat digunakan untuk mengeksport model yang telah ditransformasi serta mengimpor model yang diinginkan pengguna ke kanvas. Selanjutnya, terdapat kontroler kanvas yang dapat digunakan untuk menghapus model yang telah digambarkan dari layar serta mengembalikan posisi model ke posisi semula. Lalu, terdapat pula opsi animasi yang dapat dipilih pengguna untuk memerintahkan kamera untuk mengitari model secara otomatis. Terdapat pula opsi *shading* yang dapat dipilih pengguna untuk mengaktifkan atau menonaktifkan *shading* pada model yang digambarkan pada kanvas. Kemudian, terdapat tombol *help* yang dapat ditekan pengguna untuk mengakses panduan singkat mengenai penggunaan program. Terakhir, terdapat opsi tekstur yang dapat dipilih pengguna untuk memilih mode pemetaan tekstur kepada model.

Pada bagian tengah, terdapat kanvas yang digunakan untuk menggambar model yang dipilih oleh pengguna. Selain itu, terdapat kontroler animasi di bagian bawah kanvas yang dapat digunakan untuk memutar animasi yang telah dibuat bersamaan dengan model yang ada. Untuk mengembalikan model ke posisi semula sebelum dijalankan animasi, pengguna dapat menekan tombol *reset* yang ada.

Pada *sidebar* kanan, terdapat opsi mode proyeksi yang dapat dipilih pengguna untuk dikenakan kepada model dengan mengeklik mode yang diinginkan. Selain itu, terdapat pohon komponen sebagai representasi relasi antarkomponen pada model yang digambarkan ke layar. Pengguna dapat memilih komponen mana yang hendak dijadikan fokus kontrol dengan mengeklik tombol yang ada pada pohon. Kemudian, terdapat dua pasang kelompok kontroler dengan peran berbeda. Kelompok kontroler kiri bertanggung jawab atas transformasi global model, sedangkan kelompok kontroler kanan bertanggung jawab atas transformasi komponen model yang sebelumnya telah dipilih pada pohon komponen. Untuk melakukan transformasi, pengguna dapat menggeser *slider* yang tersedia. Terakhir, terdapat kontroler kamera yang dapat digunakan untuk mengatur radius kamera serta sudut orientasi kamera terhadap model. Untuk mengubah konfigurasi kamera, pengguna dapat menggeser *slider* yang bersesuaian.