

Tugas 3 IF3260 Grafika Komputer

3D WebGL Articulated Model

Cakupan Materi

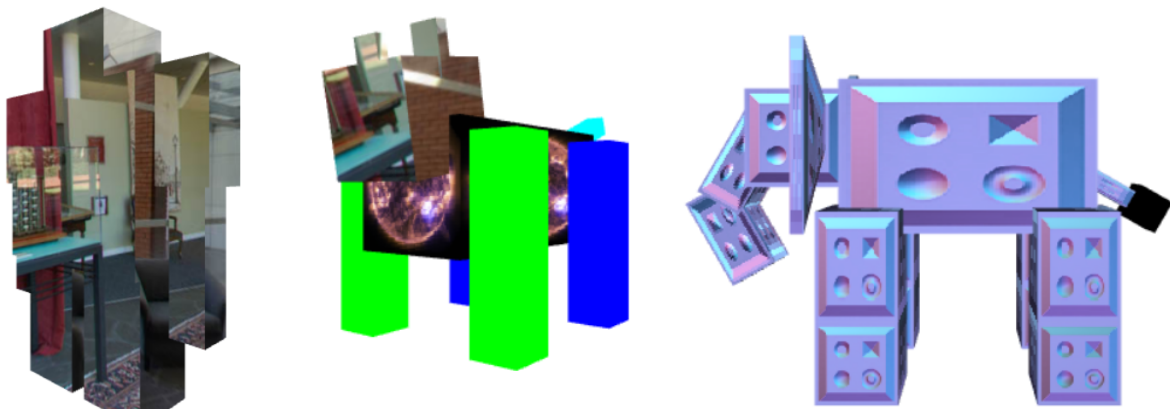
- WebGL 3D (dan materi tubes 2 lainnya, Polygon, Polyhedron, Geometry, Projection)
- Transformations (translation, rotation, scale)
- Input and Output
- Articulated Model Architecture dan Relationships
- Animasi 3D
- Tekstur Permukaan

Teknis Umum

- Satu kelompok beranggotakan 4-5 orang yang dibentuk asisten
- Menggunakan github repo yang akan dinilai kontribusi masing-masing individu lewat commit
- Deadline 20 April pukul 23.59, metode pengumpulan diinformasikan kemudian

Spesifikasi

1. **[General Implementation]** Menggunakan WebGL Murni, tanpa library/framework tambahan. Jika memerlukan fungsi-fungsi yang ada di library wrapper, dapat dibuat sendiri.
2. **[General Model Count]** Buat X buah model objek articulated dimana X = jumlah anggota kelompok. Setiap anggota wajib menyumbang satu objek articulated yang berbeda. Berikut adalah contoh objek articulated yang dimaksudkan:



3. **[Load and Save File]** 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. **[General View Controls]** Buat interaksi untuk view model keseluruhan sehingga bisa:
 - Mengubah jenis proyeksi untuk menampilkan articulated model (orthographic, oblique atau perspective)
 - Melakukan rotasi, translasi dan scaling 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.
 - Enable dan disable shading
 - Ubah tekstur permukaan model: bump, reflective, custom
5. **[Animation]** Buatlah 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. **[Component Controls]** Tambahkan 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. Contoh pada gambar di bawah.

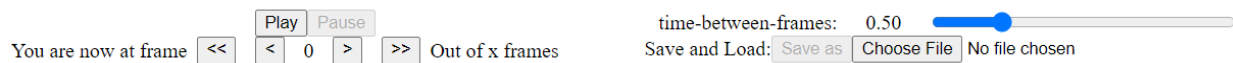
Body Rotation



Spesifikasi lanjutan punya 2 jalur yakni ke arah Component dan Animation. Jalur ini dibuat agar lebih terfokus pada fitur-fitur lanjutan yang sejalan saja. Pilih 1 jalur, lalu wajib implementasikan 2 nomor spesifikasi. Lebih dari 2 akan dihitung nilai bonus, hingga 4 nomor. Perhatikan bahwa peserta boleh mengerjakan spesifikasi lanjutan dari 2 jalur sekaligus, atau mengusulkan fitur ide orisinal yang menurut peserta dapat memberikan argumen agar asisten menganggap fitur tersebut sebagai lanjutan.

Spesifikasi Lanjut (Animation Heavy)

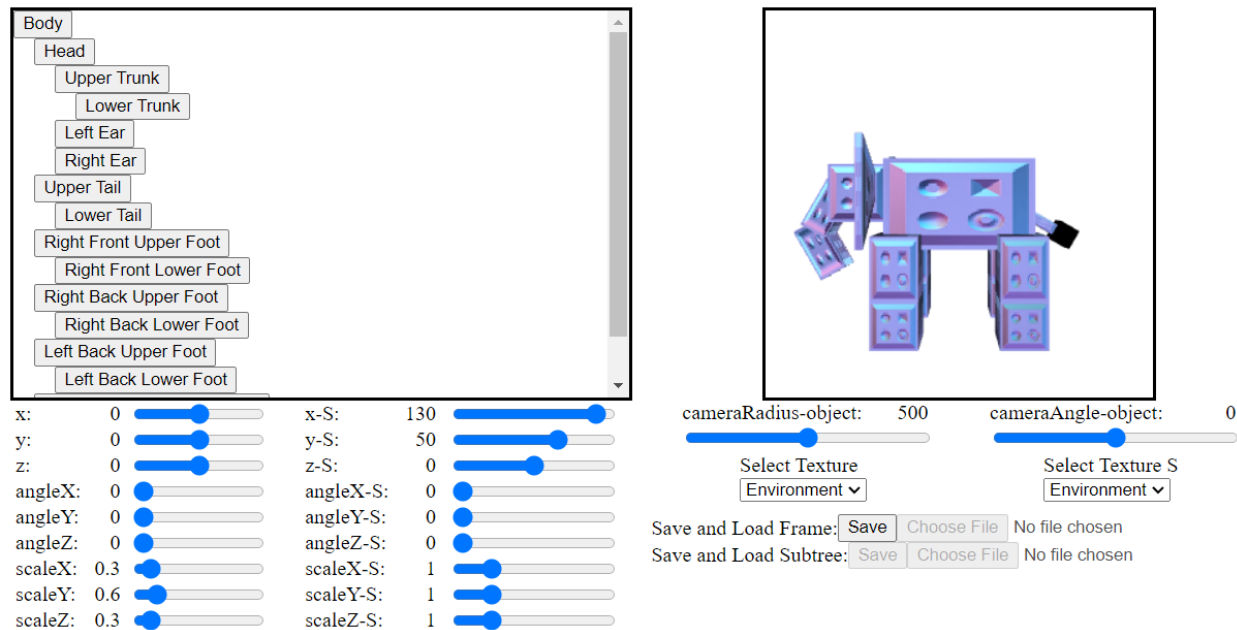
Animation Controller



7. **[Save dan Load (cont)]** Implementasi fitur ini mengharuskan format file model dan animasi terpisah. Animasi umum diimplementasikan sebagai rangkaian frame, sehingga penjelasan akan menggunakan asumsi tersebut.
Load di sini memiliki maksud memproses kumpulan frame yang terdefiniskan dalam file unggahan agar dapat diaplikasikan ke articulated model yang sesuai.
Save di sini memiliki maksud menyimpan kondisi articulated model saat ini (semua hubungan transformasinya) ke dalam sebuah frame yang dapat diunduh. Fitur ini dapat dimanfaatkan untuk peserta lebih mudah membuat tiap frame karena dibantu UI sendiri, ketimbang mengedit angka-angka dalam file yang tidak user friendly
8. **[Frame Controller]** Segala hubungan tentang manajemen frame, diantaranya:
 - Fitur penambahan frame baru dalam urutan tertentu
 - UI untuk mengurutkan frame, paling tidak swap 2 frame bersebelahan
 - UI untuk edit frame
9. **[Frame Smoothing]** Implementasi fitur penyisipan sejumlah frame diantara 2 frame yang awalnya bersebelahan. Kumpulan transformasi yang disisipkan bernilai diantara 2 frame dengan algoritma penentuan nilai dibebaskan (umumnya aritmatika saja cukup). Jika sukses akan memberikan ilusi animasi yang lebih halus berkat frame yang lebih banyak. Ini juga memudahkan pembuatan frame awal cukup keyframe-keyframe saja
10. **[Animation Controller]** Segala hubungan tentang manajemen animasi player lebih lanjut, diantaranya:
 - Play, pause, reverse, dan replay button.
 - Setup speed atau frame per sec
 - Setup custom loopable start dan end
11. **[Adaptive Frames]** Seharusnya karena struktur pohon artikulasi yang berbeda, besar kemungkinan sebuah animasi yang berupa kumpulan transformasi dari urutan frame, tidak dapat diaplikasikan ke model yang tidak sesuai. Oleh karena itu implementasikan error handling, atau setidaknya sebuah algoritma sehingga masih dapat mengaplikasikan animasi yang sebetulnya ditujukan pada model lain.

Spesifikasi Lanjut (Component Heavy)

Object Controller



12. **[Component Tree]** Agar lanjutan dari spesifikasi wajib **Component Controls** lebih efektif, tidak realistis jika membuat tampilan UI yang menampilkan kontrol setiap komponen sekaligus, karena bagaimana jika model memiliki banyak komponen. Oleh karena itu tampilkanlah struktur pohon dari komponen yang dapat diklik untuk menandakan sekarang user sedang mengontrol komponen tersebut. Hal ini juga menyebabkan jumlah slider dari component controllers tidak bertambah atau berkurang.
13. **[Component View Controls (Single)]** Tambahkan kontrol transformasi serta viewnya untuk komponen terpilih. Kontrol mencakupi sesuai fitur **General View Controls**:
- Mengubah jenis proyeksi untuk menampilkan articulated model (orthographic, oblique atau perspective)
 - Melakukan rotasi, translasi dan scaling 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.
 - Enable dan disable shading
 - Ubah tekstur permukaan model: bump, reflective, custom
14. **[Component View Controls (Subtree)]** Tambahkan kontrol transformasi serta viewnya untuk subtree komponen terpilih. Subtree merupakan semua komponen yang merupakan keturunan dari komponen terpilih. Kontrol mencakupi sesuai fitur **General View Controls**. Pengubahan dari transformasi akan memberikan efek pada seluruh komponen keturunannya. Kelompok yang mengimplementasikan fitur ini dapat menghapus tampilan **General View** awal, karena pada dasarnya dapat diakses bila memilih akar pohon untuk di-view dan dikontrol subtreenya.

15. **[Add Component]** Implementasikan fitur menambahkan sebuah articulated model sebagai anak dari existing component. Objek default berupa kubus dengan transformasi default. Jangan lupa memperbarui **Component Tree**.
16. **[Component Save and Load]** Implemenkan fitur save dan load komponen tunggal atau subtree komponen yang sedang terpilih.
Save memiliki maksud dapat menghasilkan file yang menyatakan komponen tunggal (lengan atas saja, kepala saja) dari articulated sesuai kondisi saat itu, atau menyatakan subtree komponen (keseluruhan tangan, keseluruhan belalai gajah).
Load menawarkan kemungkinan beragam untuk mengganti objek tunggal komponen terpilih (ganti lengan atas kanan sekarang menjadi kaki bawah kir), atau keseluruhan subtree komponen terpilih (misal mengganti keseluruhan kaki menjadi keseluruhan kaki gajah dari model sebelah).

Penilaian

- Source code akan dinilai dari kesesuaian dengan spesifikasi, serta kontribusi individu lewat commit history
- Melalui demo dengan asisten, teknis akan diumumkan kemudian, kemungkinan menggunakan virtual meeting (zoom/gmeet) atau rekam video dengan batasan waktu
- Laporan tugas berisi deskripsi, hasil, manual/contoh fungsionalitas program yang dibuat, **BUKAN** source code, serta keindahan website **TIDAK** dinilai

Deliverables

- Berdasarkan poin-poin di atas, buatlah repository di github dengan nama **IF3260_Tugas3_KXX_GYY** dengan XX adalah nomor kelas dan YY adalah nomor kelompok yang minimal berisi:
 - Folder src: html, css, dan js file maupun file tambahan yang diperlukan
 - Folder test: Contoh articulated model dan animation yang dapat di load
 - Folder doc: Berisi laporan dalam pdf
 - Readme: Deskripsi singkat cara menjalankan program
- Invite asisten sebagai collaborator di repository sesuai dengan yang telah dialokasikan di sheet pembagian kelompok.
- File dengan format .zip dari folder repository dengan nama file **IF3260_Tugas3_KXX_GYY** dengan XX adalah nomor kelas dan YY adalah nomor kelompok