

# SKRIPSI

## APLIKASI PRATINJAU 3 DIMENSI BERBASIS WEB



Nancy Valentina

NPM: 2014730049

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN

«tahun»



# UNDERGRADUATE THESIS

«JUDUL BAHASA INGGRIS»



Nancy Valentina

NPM: 2014730049

DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES  
PARAHYANGAN CATHOLIC UNIVERSITY

«tahun»



# LEMBAR PENGESAHAN

**APLIKASI PRATINJAU 3 DIMENSI BERBASIS WEB**

**Nancy Valentina**

**NPM: 2014730049**

**Bandung, «tanggal» «bulan» «tahun»**

**Menyetujui,**

**Pembimbing Utama**

**Pembimbing Pendamping**

**Pascal Alfadian, M.Comp.**

**Ketua Tim Penguji**

**Anggota Tim Penguji**

**«penguji 1»**

**«penguji 2»**

**Mengetahui,**

**Ketua Program Studi**

**Mariskha Tri Adithia, P.D.Eng**



## PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

### **APLIKASI PRATINJAU 3 DIMENSI BERBASIS WEB**

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,  
Tanggal «tanggal» «bulan» «tahun»

Meterai Rp. 6000
---------------------

Nancy Valentina  
NPM: 2014730049





## ABSTRAK

«Tuliskan abstrak anda di sini, dalam bahasa Indonesia»

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

**Kata-kata kunci:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Indonesia»



## ABSTRACT

«Tuliskan abstrak anda di sini, dalam bahasa Inggris»

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

**Keywords:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Inggris»



*«kepada siapa anda mempersembahkan skripsi ini...?»*



## KATA PENGANTAR

«Tuliskan kata pengantar dari anda di sini ...»

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Bandung, «bulan» «tahun»

Penulis





## DAFTAR ISI

<b>KATA PENGANTAR</b>	<b>xv</b>
<b>DAFTAR ISI</b>	<b>xvii</b>
<b>DAFTAR GAMBAR</b>	<b>xix</b>
<b>DAFTAR TABEL</b>	<b>xxi</b>
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	1
1.3 Tujuan . . . . .	2
1.4 Batasan Masalah . . . . .	2
1.5 Metodologi . . . . .	2
1.6 Sistematika Pembahasan . . . . .	3
<b>2 LANDASAN TEORI</b>	<b>5</b>
2.1 WebGL . . . . .	5
2.2 Pustaka Three.js . . . . .	9
2.3 Template Skripsi FTIS UNPAR . . . . .	10
2.3.1 Tabel . . . . .	11
2.3.2 Kutipan . . . . .	11
2.3.3 Gambar . . . . .	12
<b>A KODE PROGRAM</b>	<b>15</b>
<b>B HASIL EKSPERIMEN</b>	<b>17</b>



## DAFTAR GAMBAR

1.1	ruangan perkuliahan di Fakultas Teknologi Informasi dan Sains . . . . .	2
2.1	Gambar <i>Serpentes</i> dalam format png . . . . .	13
2.2	Ular kecil . . . . .	13
2.3	<i>Serpentes</i> betina . . . . .	14
B.1	Hasil 1 . . . . .	17
B.2	Hasil 2 . . . . .	17
B.3	Hasil 3 . . . . .	17
B.4	Hasil 4 . . . . .	17



## DAFTAR TABEL

2.1	Tabel contoh . . . . .	11
2.2	Tabel bewarna(1) . . . . .	11
2.3	Tabel bewarna(2) . . . . .	11



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Aplikasi pratinjau 3 dimensi merupakan sebuah perangkat lunak yang membantu penggunanya untuk meninjau kembali desain dari produk yang ingin dihasilkan secara 3 dimensi, sebelum pengguna tersebut melakukan implementasi pembuatan produk. Kelebihan dari aplikasi ini adalah pengguna dapat melakukan peninjauan dari berbagai sudut pandang untuk memaksimalkan hasil dari implementasi pembuatan produk. Aplikasi pratinjau tiga dimensi juga memungkinkan pengguna untuk merubah desain dari produk, hal ini bertujuan agar dapat membantu pengguna memutuskan desain produk yang paling sesuai. Pada dasarnya aplikasi pratinjau tiga dimensi bertujuan untuk membantu pengguna agar terhindar dari hasil pembuatan produk yang tidak sesuai dengan ekspektasi pengguna.

Penggunaan teknologi *web* pada aplikasi 3 dimensi dapat memudahkan pengguna untuk melakukan akses aplikasi tanpa harus melakukan instalasi aplikasi namun hanya menggunakan *browser*. Kemudian aplikasi berbasis web juga ramah untuk berbagai lingkungan sistem operasi seperti Windows, Linux, dan Mac OS sehingga tidak membatasi cakupan penggunaannya.

Pada skripsi ini, akan dibuat aplikasi pratinjau 3 dimensi berbasis web yang dapat memungkinkan pengguna untuk melakukan kustomisasi ruang belajar mengajar pada lingkungan perkuliahan. Melalui perangkat lunak ini, pengguna diharapkan dapat memiliki gambaran 3 dimensi mengenai ruangan belajar mengajar dengan komposisi warna dinding dan tekstur lantai yang tepat. Perangkat lunak akan dibuat dengan memanfaatkan WebGL dan pustaka Three.js. WebGL merupakan sebuah lintas platform, standar web bebas royalti untuk *Application Programming Interface* (API) grafis 3 dimensi level rendah yang berdasar dari OpenGL ES, terbuka untuk ECMAScript melalui elemen *canvas* HTML5. Sementara itu pustaka Three.js bertujuan membuat pustaka 3 dimensi yang mudah dan ringan untuk digunakan. Kemudian sebagai studi kasus, ruangan belajar mengajar yang akan digunakan untuk melakukan simulasi aplikasi pratinjau tiga dimensi berbasis *web* adalah salah satu ruangan perkuliahan di Fakultas Teknologi Informasi dan Sains. Ruangan tersebut dilengkapi dengan peralatan multimedia yang dapat menunjang pengajaran berbasis Teknologi Informasi seperti komputer, proyektor, serta koneksi internet yang dapat menunjang perkuliahan berbasis E-learning. Selain itu untuk menjamin kenyamanan selama perkuliahan, semua ruang kuliah dilengkapi pendingin udara.

### 1.2 Rumusan Masalah

Berikut ini masalah-masalah yang dibahas dalam skripsi ini:

- Bagaimana ruangan kelas dan objek pendukung lainnya dapat direpresentasikan dalam WebGL?
- Bagaimana membuat tampilan responsif pada aplikasi agar terlihat bagus saat dicetak?



Gambar 1.1: ruangan perkuliahan di Fakultas Teknologi Informasi dan Sains

### 1.3 Tujuan

Berikut ini tujuan-tujuan yang ingin dicapai dalam penelitian ini:

- Membangun aplikasi yang dapat merepresentasikan ruangan dalam WebGL.
- Membangun tampilan aplikasi yang responsif sehingga terlihat bagus saat dicetak.

### 1.4 Batasan Masalah

Terdapat beberapa batasan masalah dalam penelitian ini, yaitu:

1. Pengguna hanya dapat melakukan kustomisasi pada tekstur lantai, warna cat dinding bagian atas, dan warna cat dinding bagian bawah dari ruangan kelas.
2. Pengguna hanya dapat mengganti tekstur lantai, warna cat dinding bagian atas, dan warna cat dinding bagian bawah dengan 8 variasi.

### 1.5 Metodologi

Metodologi yang digunakan untuk menyusun penelitian ini adalah sebagai berikut:

1. Mempelajari standar WebGL sebagai *Application Programming Interface* untuk menampilkan grafis 3 dimensi pada *web browser*.
2. Mempelajari penggunaan Three.js sebagai *library* dari WebGL.
3. Memodelkan ruangan belajar mengajar secara 3 dimensi.



4. Melakukan analisis terhadap situs web yang akan dibangun.
5. Merancang tampilan situs web yang akan dibangun.
6. Mengimplementasikan situs web.
7. Melakukan pengujian terhadap situs web yang telah dibangun.
8. Menulis dokumen skripsi.

## 1.6 Sistematika Pembahasan

Pembahasan dalam buku skripsi ini dilakukan secara sistematis sebagai berikut:

- Bab 1 Pendahuluan Berisi latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan.
- Bab 2 Dasar Teori Berisi teori-teori dasar mengenai WebGL dan Three.js *library*.
- Bab 3 Analisis Berisi analisis masalah dan solusi, studi kasus, perancangan perangkat lunak, diagram aktivitas, *use case* diagram, dan diagram paket.
- Bab 4 Perancangan Berisi perancangan antarmuka dan diagram kelas.
- Bab 5 Implementasi Berisi implementasi antarmuka perangkat lunak, implementasi menggunakan WebGL dan *library* Three.js, pengujian perangkat lunak yang telah dibangun, dan kesimpulan berdasarkan pengujian.
- Bab 6 Kesimpulan dan Saran Berisi kesimpulan berdasarkan pengujian yang telah dilakukan dan saran untuk penelitian berikutnya.



## BAB 2

### LANDASAN TEORI

Bab ini berisi penjelasan mengenai teori-teori yang menjadi dasar penelitian ini, seperti WebGL dan Three.js *library*.

#### 2.1 WebGL

WebGL adalah sebuah Application Programming Interface (API) yang membangun objek 3 dimensi dengan mode langsung yang dirancang untuk *web*. WebGL diturunkan dari OpenGL ES 2.0, menyediakan fungsi pembangunan sejenis tetapi di dalam konteks HTML. WebGL dirancang sebagai konteks pembangunan objek pada elemen *canvas* HTML. *Canvas* pada HTML menyediakan suatu destinasi untuk pembangunan objek secara programatik pada halaman *web* dan memungkinkan menampilkan objek yang sedang dibangun menggunakan API pembangun objek yang berbeda. Berikut ini merupakan *interfaces* dan fungsionalitas yang ada pada WebGL:

##### 1. *Types*

Berikut ini merupakan tipe-tipe yang digunakan pada semua *interface* di bagian penjelasan selanjutnya.

```
typedef unsigned long   GLenum;
typedef boolean         GLboolean;
typedef unsigned long   GLbitfield;
typedef byte            GLbyte;
typedef short           GLshort;
typedef long             GLint;
typedef long             GLsizei;
typedef long long        GLintptr;
typedef long long        GLsizeiptr;
typedef octet           GLubyte;
typedef unsigned short   GLushort;
typedef unsigned long    GLuint;
typedef unrestricted float GLfloat;
typedef unrestricted float GLclampf;
```

##### 2. *WebGLContextAttributes*

*WebGLContextAttributes* merupakan kamus yang berisi atribut-atribut latar untuk menggambar yang diberikan melalui parameter kedua pada *getContext*. Berikut ini merupakan daftar nilai awal dari atribut pada *WebGLContextAttributes*, nilai awal ini akan digunakan apabila tidak ada parameter kedua yang diberikan kepada *getContext* atau jika objek pengguna yang tidak memiliki atribut pada namanya diberikan kepada *getContext*.

```
dictionary WebGLContextAttributes {
  GLboolean alpha = true;
```

```

GLboolean depth = true;
GLboolean stencil = false;
GLboolean antialias = true;
GLboolean premultipliedAlpha = true;
GLboolean preserveDrawingBuffer = false;
WebGLPowerPreference powerPreference = "default";
GLboolean failIfMajorPerformanceCaveat = false;
};

```

Berikut ini merupakan penjelasan setiap atribut pada *WebGLContextAttributes*

- *alpha*

Jika nilainya *true*, penyangga gambar telah memiliki *alpha channel* yang bertujuan untuk menampilkan operasi *alpha* destinasi OpenGL . Jika nilainya *false*, tidak ada penyangga *alpha* yang tersedia.

- *depth*

Jika nilainya *true*, penyangga gambar memiliki sebuah penyangga kedalaman yang setidaknya berisi 16 *bits*. Jika nilainya *false*, tidak ada penyangga kedalaman yang tersedia.

- *stencil*

Jika nilainya *true*, penyangga gambar memiliki penyangga stensil yang setidaknya berisi 8 *bits*. Jika nilainya *false*, tidak ada penyangga stensil yang tersedia.

- *antialias*

Jika nilainya *true* dan implementasinya mendukung *antialias* maka penyangga gambar akan menampilkan *antialias* menggunakan teknik yang dipilih dan kualitas. Jika nilainya *false* atau implementasi tidak mendukung *antialias* maka tidak ada *antialias* yang ditampilkan.

- *premultipliedAlpha*

Jika nilainya *true*, penyusun halaman akan mengasumsikan penyangga gambar memiliki warna dengan *premultiplied alpha*. Jika nilainya *false*, penyusun halaman akan mengasumsikan bahwa warna pada penyangga gambar bukan *premultiplied*.

- *preserveDrawingBuffer*

Jika nilainya *false* saat penyangga gambar mempresentasikan bagian dari penyangga gambar yang terdeskripsikan, konten-konten pada penyangga gambar akan dihapus ke nilai awalnya. Begitupun jug adengan elemen dari penyangga gambar seperti warna, kedalaman, dan stensil yang juga akan dihapus. Jika nilainya *true*, penyangga tidak akan dihapus dan akan mempresentasikan nilainya sampai nantinya dihapus atau ditulis kembali oleh penulisnya.

- *powerPreference*

Menyediakan petunjuk untuk agen pengguna yang mengindikasikan konfigurasi GPU yang cocok untuk konteks WebGL tersebut.

- *failIfMajorPerformanceCaveat*

Jika nilainya *true*, pembuatan konteks akan gagal jika implementasi menentukan bahwa performansi pada konteks WebGL yang dibuat akan sangat rendah pada aplikasi yang membuat persamaan pemanggilan OpenGL.

### 3. *WebGLObject*

*Interface WebGLObject* merupakan *interface* awal untuk diturunkan kepada semua objek GL.

```
interface WebGLObject {  
};
```

#### 4. *WebGLBuffer*

*Interface WebGLBuffer* merepresentasikan sebuah OpenGL *Buffer Object*.

```
interface WebGLBuffer : WebGLObject {  
};
```

#### 5. *WebGLFramebuffer*

*Interface WebGLFramebuffer* merepresentasikan sebuah OpenGL *Frame Buffer Object*.

```
interface WebGLFramebuffer : WebGLObject {  
};
```

#### 6. *WebGLProgram*

*Interface WebGLProgram* merepresentasikan sebuah OpenGL *Program Object*.

```
interface WebGLProgram : WebGLObject {  
};
```

#### 7. *WebGLRenderbuffer*

*Interface WebGLRenderbuffer* merepresentasikan sebuah OpenGL *Renderbuffer Object*.

```
interface WebGLRenderbuffer : WebGLObject {  
};
```

#### 8. *WebGLShader*

*Interface WebGLShader* merepresentasikan sebuah OpenGL *Shader Object*.

```
interface WebGLShader : WebGLObject {  
};
```

#### 9. *WebGLTexture*

*Interface WebGLTexture* merepresentasikan sebuah OpenGL *Texture Object*.

```
interface WebGLTexture : WebGLObject {  
};
```

#### 10. *WebGLUniformLocation*

*Interface WebGLUniformLocation* merepresentasikan lokasi dari variabel *uniform* pada program *shader*.

```
interface WebGLUniformLocation {  
};
```

#### 11. *WebGLActiveInfo*

*Interface WebGLActiveInfo* merepresentasikan informasi yang dikembalikan dari pemanggilan *getActiveAttrib* dan *getActiveUniform*.

```
interface WebGLActiveInfo {  
    readonly attribute GLint size;  
    readonly attribute GLenum type;  
    readonly attribute DOMString name;  
};
```

## 12. *WebGLShaderPrecisionFormat*

*Interface WebGLShaderPrecisionFormat* merepresentasikan informasi yang dikembalikan dari pemanggilan *getShaderPrecisionFormat*.

```
interface WebGLShaderPrecisionFormat {
    readonly attribute GLint rangeMin;
    readonly attribute GLint rangeMax;
    readonly attribute GLint precision;
};
```

## 13. *ArrayBuffer* dan *Typed Arrays*

*Vertex*, *index*, *texture*, dan data lainnya ditransfer ke implementasi WebGL menggunakan *ArrayBuffer*, *Typed Arrays*, dan *Data Views* seperti yang telah didefinisikan pada spesifikasi ECMAScript.

```
var numVertices = 100; // for example

// Hitung ukuran buffer yang dibutuhkan dalam bytes dan floats
var vertexSize = 3 * Float32Array.BYTES_PER_ELEMENT +
4 * Uint8Array.BYTES_PER_ELEMENT;
var vertexSizeInFloats = vertexSize / Float32Array.BYTES_PER_ELEMENT;

// Alokasikan buffer
var buf = new ArrayBuffer(numVertices * vertexSize);

// Map buffer ke Float32Array untuk mengakses posisi
var positionArray = new Float32Array(buf);

// Map buffer yang sama ke Uint8Array untuk mengakses warna
var colorArray = new Uint8Array(buf);

// Inisialisasi offset dari vertices dan warna pada buffer
var positionIdx = 0;
var colorIdx = 3 * Float32Array.BYTES_PER_ELEMENT;

// Inisialisasi buffer
for (var i = 0; i < numVertices; i++) {
    positionArray[positionIdx] = ...;
    positionArray[positionIdx + 1] = ...;
    positionArray[positionIdx + 2] = ...;
    colorArray[colorIdx] = ...;
    colorArray[colorIdx + 1] = ...;
    colorArray[colorIdx + 2] = ...;
    colorArray[colorIdx + 3] = ...;
    positionIdx += vertexSizeInFloats;
    colorIdx += vertexSize;
}
```

14. *WebGL Context WebGLRenderingContext* merepresentasikan API yang memungkinkan gaya pembangunan OpenGL ES 2.0 ke elemen *canvas*.

15. *WebGLContextEvent* WebGL menghasilkan sebuah *WebGLContextEvent* sebagai respon dari perubahan penting pada status konteks pembangunan WebGL. *Event* tersebut dikirim melalui

*DOM Event System* dan dilanjutkan ke *HTMLCanvasEvent* yang diasosiasikan dengan konteks pembangunan WebGL.

## 2.2 Pustaka Three.js

Pustaka Three.js ini bertujuan untuk membuat pustaka 3 dimensi yang mudah dan ringan untuk digunakan. Pustaka ini menyediakan `<canvas>`, `<svg>`, dan `CSS3D`, dan pembangun WebGL. Terdapat beberapa fungsi penting yang disediakan oleh pustaka Three.js dalam pembuatan grafis 3 dimensi, di antaranya adalah:

- *Animation*
  - *AnimationAction*, menjadwalkan penampilan dari animasi yang disimpan pada *AnimationClips*.
  - *AnimationClip*, seperangkat jalur *keyframe* yang dapat digunakan untuk merepresentasikan animasi.
  - *AnimationMixer*, alat untuk memainkan animasi pada beberapa objek di layar.
  - *AnimationObjectGroup*, sebuah grup objek yang menerima keadaan animasi yang dibagikan.
  - *AnimationUtils*, sebuah objek dengan berbagai fungsi untuk membantu animasi yang digunakan secara internal.
  - *KeyframeTrack*, sebuah urutan waktu dari *keyframe* yang berisi daftar waktu dan nilai terkait lainnya. Biasa digunakan untuk menganimasikan properti spesifik dari sebuah objek.
  - *PropertyBinding*, menyimpan referensi kepada properti asli pada graf layar yang digunakan secara internal.
  - *PropertyMixer*, penyangga properti graf layar yang memungkinkan akumulasi berat yang digunakan secara internal.
- *Cameras*
  - *Camera*, kelas abstrak untuk *cameras*. Kelas ini harus selalu diwarisi saat membangun suatu kamera.
  - *CubeCamera*, membuat 6 kamera yang dibangun pada *WebGLRenderTargetCube*.
  - *OrthographicCamera*, kamera yang menggunakan proyeksi ortografik.
  - *PerspectiveCamera*, kamera yang menggunakan proyeksi perspektif.
  - *StereoCamera*, dua buah *PerspektifCamera* yang digunakan untuk efek seperti *3D Anaglyph* dan *Parallax Barrier*.
- *Lights*
  - *AmbientLight*, sebuah cahaya yang menyinari objek secara global dan merata.
  - *DirectionalLight*, sebuah pancaran sinar dari arah yang spesifik.
  - *HemisphereLight*, sebuah cahaya yang penyinaran dilakukan tepat di atas layar dengan peleburan warna langit ke warna lantai.
  - *Light*, kelas abstrak untuk *Lights*.
  - *PointLight*, sebuah pancaran dari satu titik pada setiap arah.

- *RectAreaLight*, sebuah pancaran sinar seragam melewati permukaan bidang persegi panjang.
- *SpotLight*, sebuah pancaran dari satu titik pada setiap arah sepanjang bidang yang ukurannya dapat bertambah lebih jauh.
- *Objects*
  - *Bone*, sebuah tulang yang merupakan bagian dari kerangka.
  - *Group*, hampir sama dengan suatu *Object3D*.
  - *LensFlare*, membuat lensa suar tiruan yang mengikuti cahaya.
  - *Line*, sebuah garis yang kontinu.
  - *LineLoop*, sebuah line kontinu yang kembali ke awal.
  - *LineSegments*, beberapa garis yang ditarik antara beberapa pasang *vertex*.
  - *Mesh*, sebuah kelas yang merepresentasikan object dengan dasar segitiga.
  - *Points*, sebuah kelas yang merepresentasikan titik.
  - *Skeleton*, sebuah *array* dari tulang untuk membuat kerangka yang bisa digunakan pada *SkinnedMesh*.
  - *SkinnedMesh*, sebuah *mesh* yang mempunyai kerangka yang terdiri dari tulang dan digunakan untuk menganimasikan kumpulan *vertex* pada geometri.
- *Scenes*
  - *Fog*, kelas yang berisi parameter untuk mendefinisikan kabut.
  - *Scene*, sebuah layar yang memungkinkan untuk membuat dan menempatkan sesuatu pada pustaka Three.js.
- *Texture*
  - *Texture*, membuat tekstur untuk mengaplikasikan permukaan atau sebagai refleksi.
  - *CanvasTexture*, membuat tekstur dari suatu elemen *canvas*.

Selain itu Three.js *library* juga telah menyediakan *editor* dengan basis Graphical User Interface yang memungkinkan pengembang untuk mencoba terlebih dahulu sebelum mulai menulis kode program. *Editor* ini juga memungkinkan pengembang untuk melakukan ekspor objek 3 dimensi dalam format JSON.

## 2.3 Template Skripsi FTIS UNPAR

Akan dipaparkan bagaimana menggunakan template ini, termasuk petunjuk singkat membuat referensi, gambar dan tabel. Juga hal-hal lain yang belum terpikir sampai saat ini.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi



molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

### 2.3.1 Tabel

Berikut adalah contoh pembuatan tabel. Penempatan tabel dan gambar secara umum diatur secara otomatis oleh L<sup>A</sup>T<sub>E</sub>X, perhatikan contoh di file bab2.tex untuk melihat bagaimana cara memaksa tabel ditempatkan sesuai keinginan kita.

Perhatikan bawa berbeda dengan penempatan judul gambar gambar, keterangan tabel harus diletakkan di atas tabel!! Lihat Tabel 2.1 berikut ini:

Tabel 2.1: Tabel contoh

	$v_{start}$	$\mathcal{S}_1$	$v_{end}$
$\tau_1$	1	12	20
$\tau_2$	1		20
$\tau_3$	1	9	20
$\tau_4$	1		20

Tabel 2.2 dan Tabel 2.3 berikut ini adalah tabel dengan sel yang berwarna dan ada dua tabel yang bersebelahan.

Tabel 2.2: Tabel bewarna(1)

	$v_{start}$	$\mathcal{S}_2$	$\mathcal{S}_1$	$v_{end}$
$\tau_1$	1	5	12	20
$\tau_2$	1	8		20
$\tau_3$	1	2/8/17	9	20
$\tau_4$	1			20

Tabel 2.3: Tabel bewarna(2)

	$v_{start}$	$\mathcal{S}_1$	$\mathcal{S}_2$	$v_{end}$
$\tau_1$	1	12	5	20
$\tau_2$	1		8	20
$\tau_3$	1	9	2/8/17	20
$\tau_4$	1			20

### 2.3.2 Kutipan

Berikut contoh kutipan dari berbagai sumber, untuk keterangan lebih lengkap, silahkan membaca file referensi.bib yang disediakan juga di template ini. Contoh kutipan:

- Buku: [?]
- Bab dalam buku: [?]
- Artikel dari Jurnal: [?]
- Artikel dari prosiding seminar/konferensi: [?]
- Skripsi/Thesis/Disertasi: [?] [?] [?]
- Technical/Scientific Report: [?]
- RFC (Request For Comments): [?]
- Technical Documentation/Technical Manual: [?] [?] [?]
- Paten: [?]

- Tidak dipublikasikan: [?] [?]
- Laman web: [?]
- Lain-lain: [?]

### 2.3.3 Gambar

Pada hampir semua editor, penempatan gambar di dalam dokumen L<sup>A</sup>T<sub>E</sub>X tidak dapat dilakukan melalui proses *drag and drop*. Perhatikan contoh pada file bab2.tex untuk melihat bagaimana cara menempatkan gambar. Beberapa hal yang harus diperhatikan pada saat menempatkan gambar:

- Setiap gambar **harus** diacu di dalam teks (gunakan *field* LABEL)
- *Field* CAPTION digunakan untuk teks pengantar pada gambar. Terdapat dua bagian yaitu yang ada di antara tanda [ dan ] dan yang ada di antara tanda { dan }. Yang pertama akan muncul di Daftar Gambar, sedangkan yang kedua akan muncul di teks pengantar gambar. Untuk skripsi ini, samakan isi keduanya.
- Jenis file yang dapat digunakan sebagai gambar cukup banyak, tetapi yang paling populer adalah tipe PNG (lihat Gambar 2.1), tipe JPG (Gambar 2.2) dan tipe PDF (Gambar 2.3)
- Besarnya gambar dapat diatur dengan *field* SCALE.
- Penempatan gambar diatur menggunakan *placement specifier* (di antara tanda [ dan ] setelah deklarasi gambar. Yang umum digunakan adalah **H** untuk menempatkan gambar **sesuai** penempatannya di file .tex atau **h** yang berarti "kira-kira" di sini. Jika tidak menggunakan *placement specifier*, L<sup>A</sup>T<sub>E</sub>X akan menempatkan gambar secara otomatis untuk menghindari bagian kosong pada dokumen anda. Walaupun cara ini sangat mudah, hindarkan terjadinya penempatan dua gambar secara berurutan.
  - Gambar 2.1 ditempatkan di bagian atas halaman, walaupun penempatannya dilakukan setelah penulisan 3 paragraf setelah penjelasan ini.
  - Gambar 2.2 dengan skala 0.5 ditempatkan di antara dua buah paragraf. Perhatikan penulisannya di dalam file bab2.tex!
  - Gambar 2.3 ditempatkan menggunakan *specifier* **h**.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo.



Gambar 2.1: Gambar *Serpentes* dalam format png

Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

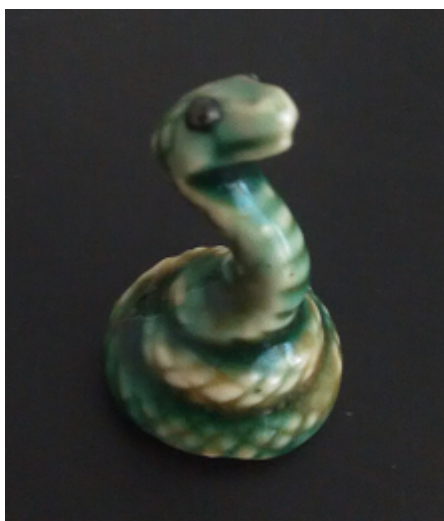
Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.



Gambar 2.2: Ular kecil

Etiam pede massa, dapibus vitae, rhoncus in, placerat posuere, odio. Vestibulum luctus commodo lacus. Morbi lacus dui, tempor sed, euismod eget, condimentum at, tortor. Phasellus aliquet odio ac lacus tempor faucibus. Praesent sed sem. Praesent iaculis. Cras rhoncus tellus sed justo ullamcorper sagittis. Donec quis orci. Sed ut tortor quis tellus euismod tincidunt. Suspendisse congue nisl eu elit. Aliquam tortor diam, tempus id, tristique eget, sodales vel, nulla. Praesent tellus mi, condimentum sed, viverra at, consectetur quis, lectus. In auctor vehicula orci. Sed pede sapien, euismod in, suscipit in, pharetra placerat, metus. Vivamus commodo dui non odio. Donec et felis.

Etiam suscipit aliquam arcu. Aliquam sit amet est ac purus bibendum congue. Sed in eros. Morbi non orci. Pellentesque mattis lacinia elit. Fusce molestie velit in ligula. Nullam et orci vitae nibh vulputate auctor. Aliquam eget purus. Nulla auctor wisi sed ipsum. Morbi porttitor tellus ac enim. Fusce ornare. Proin ipsum enim, tincidunt in, ornare venenatis, molestie a, augue. Donec vel pede in lacus sagittis porta. Sed hendrerit ipsum quis nisl. Suspendisse quis massa ac nibh pretium cursus. Sed sodales. Nam eu neque quis pede dignissim ornare. Maecenas eu purus ac urna tincidunt congue.



Gambar 2.3: *Serpentes jantan*

# LAMPIRAN A

## KODE PROGRAM

Listing A.1: MyCode.c

```
1 // This does not make algorithmic sense,
2 // but it shows off significant programming characters.
3
4 #include<stdio.h>
5
6 void myFunction( int input, float* output ) {
7     switch ( array[i] ) {
8         case 1: // This is silly code
9             if ( a >= 0 || b <= 3 && c != x )
10                 *output += 0.005 + 20050;
11             char = 'g';
12             b = 2^n + ~right_size - leftSize * MAX_SIZE;
13             c = (--aaa + &daa) / (bbb++ - ccc % 2 );
14             strcpy(a,"hello_$@?");
15         }
16         count = ~mask | 0x00FF00AA;
17     }
18 }
19
20 // Fonts for Displaying Program Code in LATEX
21 // Adrian P. Robson, nepsweb.co.uk
22 // 8 October 2012
23 // http://nepsweb.co.uk/docs/progfonts.pdf
```

Listing A.2: MyCode.java

```
1 import java.util.ArrayList;
2 import java.util.Collections;
3 import java.util.HashSet;
4
5 //class for set of vertices close to furthest edge
6 public class MyFurSet {
7     protected int id; //id of the set
8     protected MyEdge FurthestEdge; //the furthest edge
9     protected HashSet<MyVertex> set; //set of vertices close to furthest edge
10    protected ArrayList<ArrayList<Integer>> ordered; //list of all vertices in the set for each trajectory
11    protected ArrayList<Integer> closeID; //store the ID of all vertices
12    protected ArrayList<Double> closeDist; //store the distance of all vertices
13    protected int totaltrj; //total trajectories in the set
14
15    /*
16     * Constructor
17     * @param id : id of the set
18     * @param totaltrj : total number of trajectories in the set
19     * @param FurthestEdge : the furthest edge
20     */
21    public MyFurSet(int id,int totaltrj,MyEdge FurthestEdge) {
22        this.id = id;
23        this.totaltrj = totaltrj;
24        this.FurthestEdge = FurthestEdge;
25        set = new HashSet<MyVertex>();
26        ordered = new ArrayList<ArrayList<Integer>>();
27        for (int i=0;i<totaltrj;i++) ordered.add(new ArrayList<Integer>());
28        closeID = new ArrayList<Integer>(totaltrj);
29        closeDist = new ArrayList<Double>(totaltrj);
30        for (int i = 0;i <totaltrj;i++) {
31            closeID.add(-1);
32            closeDist.add(Double.MAX_VALUE);
33        }
34    }
35
36 }
```



## LAMPIRAN B

### HASIL EKSPERIMEN

Hasil eksperimen berikut dibuat dengan menggunakan TIKZPICTURE (bukan hasil excel yg diubah ke file bitmap). Sangat berguna jika ingin menampilkan tabel (yang kuantitasnya sangat banyak) yang datanya dihasilkan dari program komputer.



Gambar B.1: Hasil 1



Gambar B.2: Hasil 2



Gambar B.3: Hasil 3



Gambar B.4: Hasil 4