

**SKRIPSI**

**VISUALISASI KURIKULUM 2018 DENGAN VIS.JS DAN  
ELECTRON**



**Joshua Delavo Setiaddi**

**NPM: 2017730028**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN  
2021**



**UNDERGRADUATE THESIS**

**«JUDUL BAHASA INGGRIS»**



**Joshua Delavo Setiaddi**

**NPM: 2017730028**

**DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES  
PARAHYANGAN CATHOLIC UNIVERSITY  
2021**



# LEMBAR PENGESAHAN

## VISUALISASI KURIKULUM 2018 DENGAN VIS.JS DAN ELECTRON

Joshua Delavo Setiaddi

NPM: 2017730028

Bandung, 09 April 2021

Menyetujui,

Pembimbing Utama

Pembimbing Pendamping

Pascal Alfadian, Nugroho, M.Comp.

«pembimbing pendamping/2»

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:

### **VISUALISASI KURIKULUM 2018 DENGAN VIS.JS DAN ELECTRON**

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 09 April 2021



Joshua Delavo Setiaddi  
NPM: 2017730028





## **ABSTRAK**

«Tuliskan abstrak anda di sini, dalam bahasa Indonesia»

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



## **ABSTRACT**

«Tuliskan abstrak anda di sini, dalam bahasa Inggris»

**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 ...»

Bandung, April 2021

Penulis





# DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
1 PENDAHULUAN	1
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	2
1.3 Tujuan . . . . .	2
1.4 Batasan Masalah . . . . .	2
1.5 Metodologi . . . . .	2
1.6 Sistematika Pembahasan . . . . .	3
1.7 Skripsi . . . . .	4
1.8 L <sup>A</sup> T <sub>E</sub> X . . . . .	4
1.9 Template Skripsi FTIS UNPAR . . . . .	4
1.9.1 Tabel . . . . .	5
1.9.2 Kutipan . . . . .	5
1.9.3 Gambar . . . . .	5
1.9.4 Kode Program . . . . .	7
1.9.5 Notasi . . . . .	8
DAFTAR REFERENSI	9
A KODE PROGRAM	11
B HASIL EKSPERIMEN	13



## DAFTAR GAMBAR

1.1	Logo Electron . . . . .	1
1.2	Logo VisJs . . . . .	2
1.3	Gambar <i>Serpentes</i> dalam format png . . . . .	6
1.4	Ular kecil . . . . .	7
1.5	<i>Serpentes</i> betina . . . . .	8
B.1	Hasil 1 . . . . .	13
B.2	Hasil 2 . . . . .	13
B.3	Hasil 3 . . . . .	13
B.4	Hasil 4 . . . . .	13



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Dibutuhkan waktu yang lama bagi para pengembang web yang ingin membuat aplikasi *desktop*, dimana mereka perlu mempelajari bahasa baru bersamaan dengan kerangka kerjanya. Hal ini tentunya akan menjadi penghalang bagi mereka yang mau mengembangkan aplikasi *desktop*. Maka dari itu, saat ini terdapat sebuah kerangka kerja yang tersedia bagi pengembang web untuk membuat aplikasi *desktop* lintas *platform*. *Electron* adalah salah satu kerangka kerja terbaru yang memungkinkan pengembang membuat aplikasi *desktop* asli dengan teknologi web populer : JavaScript, HTML5, dan CSS. Dengan *Electron*, pengembang web dapat menggunakan keterampilan yang mereka miliki untuk membangun aplikasi yang memiliki banyak kemampuan seperti aplikasi *desktop* asli. *Elektron* telah menjadi sangat populer sejak dirilis dan digunakan oleh perusahaan seperti *Microsoft*, *Facebook*, *Slack*, dan *Docker*. Aplikasi ini dapat dikemas untuk dapat berjalan langsung di *macOS*, *Windows*, dan *Linux*. Bisa juga didistribusikan melalui *Mac App Store* atau *Microsoft Store*. Logo *Electron* dapat dilihat di Gambar 1.1.



Gambar 1.1: Logo Electron

<sup>1</sup>

Dalam pembuatan sebuah aplikasi *desktop*, tentunya kita memerlukan *library* untuk membantu kita dalam membuat aplikasi tersebut. Terdapat banyak *library* visualisasi yang tersedia, contohnya : *Vega*, *D3*, dan *Vis.js*. Namun untuk *library* visualisasi *Vega* dan *D3* terlalu *powerful* untuk menangani proyek visualisasi kurikulum dalam bentuk *tree* dan *timeline*. *Library* visualisasi *Vis.js* sudah sangat memadai untuk menangani proyek tersebut. *Vis.js* adalah sebuah *library* visualisasi berbasis *browser* yang bersifat dinamis. *Library* ini dirancang agar mudah digunakan untuk menangani data dinamis dalam jumlah yang besar dan memungkinkan untuk memanipulasi serta berinteraksi dengan data tersebut. *Library* ini terdiri dari komponen - komponen, seperti *DataSet*, *Timeline*, *Network*, *Graph2d*, dan *Graph3d*. Logo *Vis.js* dapat dilihat pada gambar 1.2.

---

<sup>1</sup><https://cdn.hipwallpaper.com/i/62/53/yK5zr0.jpg>



Gambar 1.2: Logo VisJs

2

Saat mahasiswa akan melakukan *FRS*, seringkali mereka kesulitan untuk melihat kurikulum tahun ajaran yang berlaku. Karena setiap kurikulum memiliki aturan yang berbeda dalam pengambilan matakuliah ataupun matakuliah yang disediakan, maka mahasiswa kadang bingung untuk memilih matakuliah apa yang akan diambil di semester berikutnya.

Maka dari itu, pada skripsi ini akan dibuat sebuah aplikasi visualisasi kurikulum 2018 berbasis *Electron* dengan menggunakan *library Vis.js*. Dengan aplikasi ini diharapkan mahasiswa dapat lebih mudah untuk melihat kurikulum yang ada sehingga mempermudah mereka untuk memilih matakuliah apa yang akan diambil di semester berikutnya, kemudian dengan penggunaan *framework cross platform Electron* ini diharapkan semua mahasiswa pengguna *macOs*, *Windows*, dan *Linux* dapat mengaksesnya dengan mudah.

## 1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas pada skripsi ini adalah :

1. Bagaimana cara memvisualisasikan kurikulum 2018 dalam bentuk *tree*?
2. Bagaimana cara memvisualisasikan kurikulum 2018 dalam bentuk *timeline*?

## 1.3 Tujuan

Tujuan yang akan dicapai dari penulisan skripsi ini adalah :

1. Memahami cara memvisualisasikan kurikulum 2018 dalam bentuk *tree*.
2. Memahami cara memvisualisasikan kurikulum 2018 dalam bentuk *timeline*.

## 1.4 Batasan Masalah

Dalam skripsi ini dibuat batasan-batasan masalah dalam pembuatan perangkat lunak visualisasi kurikulum 2018. Batasan-batasan masalah yang ditetapkan adalah sebagai berikut :

1. Perangkat lunak ini hanya memvisualisasikan kurikulum milik Universitas Katolik Parahyangan tahun 2018.
2. Perangkat lunak ini hanya memvisualisasikan dalam bentuk *tree* dan *timeline*.
3. Perangkat lunak ini hanya akan menggunakan bahasa *HTML*, *PHP* dan *JavaScript*

## 1.5 Metodologi

Metodologi penelitian berisi langkah-langkah dengan urutan yang tepat agar skripsi ini dapat berjalan sistematis. Bagian - bagian pengerjaan skripsi ini adalah :

---

<sup>2</sup><https://i.pining.com/originals/e1/df/59/e1df59c6d0ca7aaf3746c2238e075c48.png>

1. Melakukan studi tentang *framework Electron* dan *Library Vis js*.
2. Mempelajari cara membuat aplikasi berbasis *Electron*.
3. Mempelajari cara memvisualisasikan data dalam bentuk *tree* dan *timeline* dengan *Vis js*.
4. Mempelajari data kurikulum 2018 di github beserta cara pengambilan datanya.
5. Merancang aplikasi berbasis *Electron*.
6. Merancang visualisasi kurikulum 2018 dalam bentuk *tree*.
7. Merancang visualisasi kurikulum 2018 dalam bentuk *timeline*.
8. Mendesain antarmuka aplikasi.
9. Melakukan pengujian dan eksperimen.
10. Membuat dokumen skripsi.

## 1.6 Sistematika Pembahasan

Rencananya skripsi ini terdiri dari empat bab, yaitu pendahuluan, landasan teori, implementasi dan analisis, dan kesimpulan dan saran. Pada Bab I membahas latar belakang dibuatnya skripsi, rumusan masalah yang terdapat pada skripsi, tujuan skripsi ini dibuat, batasan masalah agar skripsi yang dibuat tidak terlalu luas, dan metodologi yang berisi langkah - langkah pengerjaan skripsi agar berjalan sistematis.

Bab II berisi teori - teori yang berfungsi sebagai referensi dalam pembuatan skripsi dan membantu dalam menyelesaikan masalah pada skripsi. Bab III berisi implementasi perangkat lunak yang berlandaskan teori - teori yang telah dipelajari serta analisis terhadap perangkat lunak yang telah dibuat. Bab IV berisi kesimpulan skripsi yang telah dibuat dan juga saran yang ditujukan untuk skripsi berikutnya.

## 1.7 Skripsi

Rencananya akan diisi dengan penjelasan umum mengenai buku skripsi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetur eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetur tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

## 1.8 L<sup>A</sup>T<sub>E</sub>X

Mengapa menggunakan L<sup>A</sup>T<sub>E</sub>X untuk buku skripsi dan apa keunggulan/kerugiannya bagi mahasiswa dan pembuat template.

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

## 1.9 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.

### 1.9.1 Tabel

Berikut adalah contoh pembuatan tabel. Penempatan tabel dan gambar secara umum diatur secara otomatis oleh  $\text{\LaTeX}$ , 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 1.1 berikut ini:

Tabel 1.1: 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 1.2 dan Tabel 1.3 berikut ini adalah tabel dengan sel yang berwarna dan ada dua tabel yang bersebelahan.

Tabel 1.2: 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 1.3: 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

### 1.9.2 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: [1]
- Bab dalam buku: [2]
- Artikel dari Jurnal: [3]
- Artikel dari prosiding seminar/konferensi: [4]
- Skripsi/Thesis/Disertasi: [5] [6] [7]
- Technical/Scientific Report: [8]
- RFC (Request For Comments): [9]
- Technical Documentation/Technical Manual: [10] [11] [12]
- Paten: [13]
- Tidak dipublikasikan: [14] [15]
- Laman web: [16]
- Lain-lain: [17]

### 1.9.3 Gambar

Pada hampir semua editor, penempatan gambar di dalam dokumen  $\text{\LaTeX}$  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:



Gambar 1.3: Gambar *Serpentes* dalam format png

- 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 1.3), tipe JPG (Gambar 1.4) dan tipe PDF (Gambar 1.5)
- 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 1.3 ditempatkan di bagian atas halaman, walaupun penempatannya dilakukan setelah penulisan 3 paragraf setelah penjelasan ini.
- Gambar 1.4 dengan skala 0.5 ditempatkan di antara dua buah paragraf. Perhatikan penulisannya di dalam file bab2.tex!
- Gambar 1.5 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. 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 1.4: 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.

### 1.9.4 Kode Program

Kode program dalam bahasa tertentu seringkali harus ditulis di dalam bab, bukan hanya dilampirkan di bagian Lampiran. Kode 1.1 menampilkan penggunaan karakter-karakter yang umum digunakan dalam sebuah program yang ditulis dengan bahasa C.

Kode 1.1: Kode untuk menampilkan karakter-karakter aneh

```
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;
```



Gambar 1.5: *Serpentes* jantan

```

17| }
18|
19| // Fonts for Displaying Program Code in LATEX
20| // Adrian P. Robson, nepsweb.co.uk
21| // 8 October 2012
22| // http://nepsweb.co.uk/docs/progfonts.pdf

```

### 1.9.5 Notasi

Simbol-simbol (matematika) yang sering digunakan sepanjang penulisan skripsi, dapat dimasukkan ke dalam “Daftar Notasi”. Daftar ini ada di halaman depan sebelum Bab 1. Cara memasukkan sebuah simbol ke dalam Daftar Notasi adalah menggunakan perintah `\nomenclature`. Contoh:

```
\nomenclature[]{$A$}{luas kandang ular}
```

Argumen opsional digunakan untuk mengurutkan notasi. Silahkan lihat sendiri dokumentasi package `nomenc1`

## DAFTAR REFERENSI

- [1] de Berg, M., Cheong, O., van Kreveld, M. J., dan Overmars, M. (2008) *Computational Geometry: Algorithms and Applications*, 3rd edition. Springer-Verlag, Berlin.
- [2] van Kreveld, M. J. (2004) Geographic information systems. Bagian dari Goodman, J. E. dan O'Rourke, J. (ed.), *Handbook of Discrete and Computational Geometry*. Chapman & Hall/CRC, Boca Raton.
- [3] Buchin, K., Buchin, M., van Kreveld, M. J., Löffler, M., Silveira, R. I., Wenk, C., dan Wiratma, L. (2013) Median trajectories. *Algorithmica*, **66**, 595–614.
- [4] van Kreveld, M. J. dan Wiratma, L. (2011) Median trajectories using well-visited regions and shortest paths. *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, Chicago, USA, 1-4 November, pp. 241–250. ACM, New York.
- [5] Lionov (2002) Animasi algoritma sweepline untuk membangun diagram voronoi. Skripsi. Universitas Katolik Parahyangan, Indonesia.
- [6] Wiratma, L. (2010) Following the majority: a new algorithm for computing a median trajectory. Thesis. Utrecht University, The Netherlands.
- [7] Wiratma, L. (2022) Coming Not Too Soon, Later, Delay, Someday, Hopefully. Disertasi. Utrecht University, The Netherlands.
- [8] van kreveld, M., van Lankveld, T., dan Veltkamp, R. (2013) Watertight scenes from urban lidar and planar surfaces. Technical Report UU-CS-2013-007. Utrecht University, The Netherlands.
- [9] Rekhter, Y. dan Li, T. (1994) A border gateway protocol 4 (bgp-4). RFC 1654. RFC Editor, <http://www.rfc-editor.org>.
- [10] ITU-T Z.500 (1997) *Framework on formal methods in conformance testing*. International Telecommunications Union. Geneva, Switzerland.
- [11] Version 9.0.0 (2016) *The Unicode Standard*. The Unicode Consortium. Mountain View, USA.
- [12] Version 7.0 Nougat (2016) *Android API Reference Manual*. Google dan Open Handset Alliance. Mountain View, USA.
- [13] Webb, R., Daruca, O., dan Alfadian, P. (2012) *Method of optimizing a text message communication between a server and a secure element*. Paten no. EP2479956 (A1). European Patent Organisation. Munich, Germany.
- [14] Wiratma, L. (2009) Median trajectory. Report for GMT Experimentation Project at Utrecht University.
- [15] Lionov (2011) Polymorphism pada C++. Catatan kuliah AKS341 Pemrograman Sistem di Universitas Katolik Parahyangan, Bandung. <http://tinyurl.com/lionov>. 30 September 2016.

- [16] Erickson, J. (2003) CG models of computation? <http://www.computational-geometry.org/mailling-lists/compgeom-announce/2003-December/000852.html>. 30 September 2016.
- [17] AGUNG (2012) Menjajal tango 12. Majalah HAI no 02, Januari 2012.

# LAMPIRAN A

## KODE PROGRAM

Kode 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
```

Kode 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