

«SKRIPSI/TUGAS AKHIR»

«JUDUL BAHASA INDONESIA»



«Nama Lengkap»

NPM: «10 digit NPM UNPAR»

PROGRAM STUDI «MATEMATIKA/FISIKA/TEKNIK INFORMATIKA»
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
«tahun»

«FINAL PROJECT/UNDERGRADUATE THESIS»

«JUDUL BAHASA INGGRIS»



«Nama Lengkap»

NPM: «10 digit NPM UNPAR»

DEPARTMENT OF «MATHEMATICS/PHYSICS/INFORMATICS»
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
«tahun»

LEMBAR PENGESAHAN

«JUDUL BAHASA INDONESIA»

«Nama Lengkap»

NPM: «10 digit NPM UNPAR»

Bandung, «tanggal» «bulan» «tahun»

Menyetujui,

Pembimbing Utama

Pembimbing Pendamping

«pembimbing utama/1»

«pembimbing pendamping/2»

Ketua Tim Penguji

Anggota Tim Penguji

«penguji 1»

«penguji 2»

Mengetahui,

Ketua Program Studi

«Ketua Program Studi»

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa «skripsi/tugas akhir» dengan judul:

«JUDUL BAHASA INDONESIA»

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

«Nama Lengkap»
NPM: «10 digit NPM UNPAR»

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

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
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	2
2 LANDASAN TEORI	3
2.1 Skripsi	3
2.1.1 Pengertian Permainan Snake	3
2.1.2 HTML5 Canvas	4
2.1.3 Javascript dan jQuery	5
2.1.4 Github	8
2.2 L ^A T _E X	8
2.3 Template Skripsi FTIS UNPAR	9
2.3.1 Tabel	9
2.3.2 Kutipan	9
2.3.3 Gambar	10
A KODE PROGRAM	13
B HASIL EKSPERIMEN	15

DAFTAR GAMBAR

2.1	Pergerakan ular ke segala arah	3
2.2	Permainan Snake pada telepon genggam <i>Nokia</i>	4
2.3	Permainan <i>Slither.io</i> pada <i>Android</i>	4
2.4	Membuat canvas	4
2.5	<i>Document ready event</i>	5
2.6	Inisialisasi menggunakan <i>Javascript</i>	5
2.7	Inisialisasi menggunakan <i>jQuery</i>	6
2.8	Membuat objek pada <i>Javascript</i>	6
2.9	<i>Event keydown</i> pada tombol atas, bawah, kiri, dan kanan	7
2.10	Membuat kotak berpindah posisi	8
2.11	Gambar <i>Serpentes</i> dalam format png	11
2.12	Ular kecil	11
2.13	<i>Serpentes</i> betina	12
B.1	Hasil 1	15
B.2	Hasil 2	15
B.3	Hasil 3	15
B.4	Hasil 4	15

DAFTAR TABEL

2.1	Tabel contoh	9
2.2	Tabel bewarna(1)	9
2.3	Tabel bewarna(2)	9

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Snake merupakan sebuah permainan yang pertama kali dibuat oleh Peter Trefonas pada tahun 1978. Konsep *Snake* berasal dari permainan arkade yaitu *Blockade*. Awalnya *Snake* hanya dapat dimainkan pada komputer pribadi. Namun pada tahun 1997, *Snake* dapat dimainkan pada telepon genggam *Nokia* dan *Nokia* yang mempopulerkan *Snake*. Cara bermain *Snake* adalah pemain menggerakkan ular pada sebuah labirin. Ular tersebut harus mendapatkan makanan sebanyak-banyaknya tanpa menabrak dinding atau ular itu sendiri. Setiap memakan makanan, tubuh ular akan memanjang dan pemain akan semakin sulit untuk menggerakkan ular tersebut dengan bebas karena tubuh ular semakin lama akan menutupi labirin tersebut.

HTML(*Hyper Text Markup Language*) adalah sebuah bahasa markah yang digunakan untuk membuat halaman web. HTML5 merupakan HTML versi 5 yang terbaru dan penerus dari HTML4, XHTML1, dan DOM level 2 HTML. HTML5 memiliki beberapa elemen baru, salah satunya adalah HTML5 Canvas. HTML5 Canvas adalah tempat untuk menggambar *pixel-pixel* yang dapat ditulis menggunakan bahasa pemrograman *JavaScript*. *Javascript* adalah bahasa pemrograman tingkat tinggi yang digunakan untuk membuat halaman web menjadi lebih interaktif. *jQuery* adalah *library* dari *Javascript* untuk menyederhanakan *script* pada HTML. *Github* adalah layanan *web hosting* bersama untuk proyek pengembangan perangkat lunak yang menggunakan sistem *version control* yaitu *Git*. Dengan adanya *Github*, *programmer* dapat mengetahui perubahan yang pada *repository* tersebut.

Sekarang sudah banyak sekali *Snake* yang dapat dimainkan di *smartphone* dan web. Pergerakan ular tidak hanya atas, bawah, kiri, dan kanan saja, tetapi sudah dapat bergerak ke segala arah. Pada skripsi ini, penulis akan membuat *Snake* yang berbeda dari yang sudah ada sebelumnya. Ular pada *Snake* dapat bergerak ke segala arah dan orang lain dapat menambahkan labirin menggunakan mekanisme *pull request Github*. Dengan begitu, orang lain dapat menambahkan labirin sesuai dengan keinginannya dan pemain tidak akan cepat bosan karena labirin yang disediakan cukup banyak dan variatif.

1.2 Rumusan Masalah

Rumusan dari masalah yang akan dibahas pada skripsi ini adalah sebagai berikut:

- Bagaimana membangun permainan *Snake* menggunakan HTML5?
- Bagaimana cara menyimpan labirin pada file eksternal?
- Bagaimana cara menggunakan *pull request* pada *Github* agar orang lain dapat menambahkan labirin?

1.3 Tujuan

Tujuan-tujuan yang hendak dicapai melalui penulisan skripsi ini adalah sebagai berikut:

- Dapat membangun permainan *Snake* menggunakan HTML5.
- Dapat menyimpan labirin pada file eksternal.
- Dapat menggunakan *pull request* pada *Github* agar orang lain dapat menambahkan labirin.

1.4 Batasan Masalah

Beberapa batasan yang dibuat terkait dengan pengerjaan skripsi ini adalah sebagai berikut:

- Permainan ini hanya dapat dimainkan menggunakan *web browser* pada komputer.
- *Web browser* yang digunakan sudah mendukung HTML5 Canvas.

1.5 Metodologi

Metodologi pada penelitian ini adalah sebagai berikut:

1. Melakukan studi literatur tentang HTML5, *JavaScript*, *jQuery*, dan *pull request* pada *Github*.
2. Melakukan analisis dan menentukan objek-objek pada *Snake*.
3. Merancang algoritma untuk menggambar tubuh ular, pergerakan ular dan membuat labirin.
4. Mengimplementasikan keseluruhan algoritma.
5. Menambahkan labirin menggunakan *pull request* pada *Github*.
6. Melakukan pengujian.
7. Melakukan penarikan kesimpulan.

1.6 Sistematika Pembahasan

Sistematikan penulisan setiap bab pada penelitian ini adalah sebagai berikut:

1. Bab 1 berisikan latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan dari penelitian yang dilakukan.
2. Bab 2 berisikan dasar-dasar teori yang menunjang penelitian ini. Teori yang digunakan adalah: pengertian *Snake*, HTML5 Canvas, *Javascript*, *jQuery* dan *Github*.

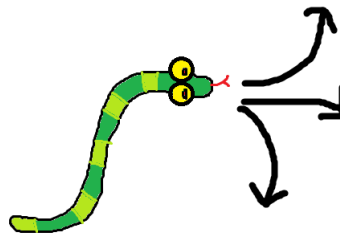
BAB 2

LANDASAN TEORI

2.1 Skripsi

2.1.1 Pengertian Permainan Snake

Snake merupakan permainan mengendalikan ular untuk mendapatkan makanan yang terdapat pada labirin. Dalam permainan ini, pemain mengendalikan ular untuk mendapatkan makanan sebanyak-banyaknya. Setiap ular memakan makanan, maka skor akan bertambah 1 poin dan tubuh ular akan bertambah panjang. Biasanya makanan hanya ada 1 saja pada sebuah labirin. Ketika makanan itu sudah termakan oleh ular, makanan tersebut akan ditempatkan secara acak. Ular dapat bergerak ke atas, bawah, kiri, dan kanan. Namun pada permainan *Snake* sekarang, ular sudah dapat bergerak ke segala arah seperti ilustrasi pada Gambar 2.1. Permainan akan berakhir jika ular menabrak dinding yang terdapat pada labirin atau ular tersebut menabrak tubuhnya sendiri.

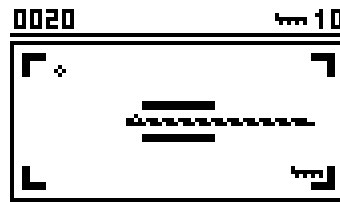
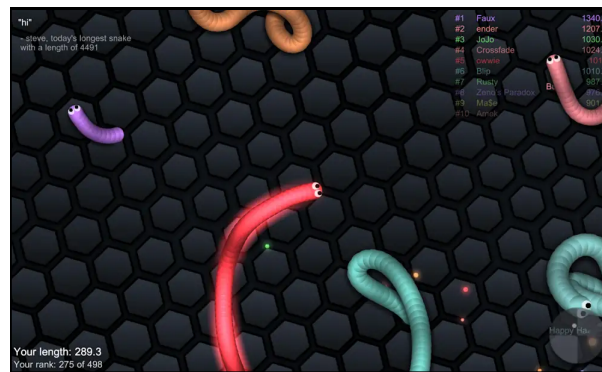


Gambar 2.1: Pergerakan ular ke segala arah

Permainan *Snake* ini dapat dimainkan secara *singleplayer* atau *multiplayer*. *Singleplayer game* adalah permainan yang dapat dimainkan oleh 1 pemain. *Multiplayer game* adalah permainan yang dapat dimainkan oleh beberapa pemain. Permainan *Snake* sekarang dimainkan secara *singleplayer*. Contoh *singleplayer game Snake* adalah *Snake* pada telepon genggam *Nokia* yang dapat dilihat pada Gambar 2.2¹ dan contoh *multiplayer game Snake* adalah *Slither.io* yang dapat dilihat Gambar 2.3². *Snake* dapat dimainkan menggunakan *smartphone* dan *web browser*.

¹[https://en.wikipedia.org/wiki/Snake_\(video_game_genre\)](https://en.wikipedia.org/wiki/Snake_(video_game_genre))

²<https://play.google.com/store/apps/details?id=air.com.hypah.io.slither>

Gambar 2.2: Permainan Snake pada telepon genggam *Nokia*Gambar 2.3: Permainan *Slither.io* pada *Android*

2.1.2 HTML5 Canvas

HTML(*Hyper Text Markup Language*) adalah bahasa markah yang digunakan untuk membuat halaman web. Untuk mengisi konten/isi dari halaman web, digunakan *tag*. HTML5(*Hyper Text Markup Language 5*) adalah versi HTML terbaru yang merupakan penerus dari HTML4, XHTML1, dan DOM *level 2* HTML. HTML5 memiliki beberapa elemen baru, salah satunya yaitu HTML5 Canvas³. HTML5 Canvas adalah sebuah tempat untuk menggambar *pixel-pixel* yang ditulis menggunakan *Javascript*. HTML5 Canvas API(*Application Programming Interface*) yang paling umum digunakan adalah *2D Context*. *2D Context* ini digunakan untuk membuat bentuk 2 dimensi, menampilkan gambar, membuat tulisan, menambahkan warna, dan membuat garis dan kurva. HTML5 Canvas tidak hanya sekedar untuk membuat bentuk dan menampilkan gambar saja, HTML5 Canvas dapat digunakan untuk membuat animasi, aplikasi web dan *games*. Untuk menambah canvas pada halaman HTML, tambahkan *tag* `<canvas>` pada *body* HTML seperti pada Gambar 2.4.

```
<body>
  <canvas id="snake" width="1340" height="550">
    <!-- Insert fallback content here -->
  </canvas>
</body>
```

Gambar 2.4: Membuat canvas

Pada Gambar 2.4, terdapat 3 atribut pada canvas. Berikut adalah penjelasan dari setiap atribut:

- *id*: *id* digunakan sebagai penanda yang akan digunakan oleh *Javascript*. Id dari canvas adalah *snake*.
- *Width*: *Width* adalah lebar canvas tersebut. Lebar dari canvas adalah 1340 *pixel*.
- *Height*: *Height* adalah tinggi dari canvas tersebut. Tinggi dari canvas adalah 550 *pixel*.

³<https://en.wikipedia.org/wiki/HTML5>

2.1.3 Javascript dan jQuery

Javascript adalah bahasa pemrograman tingkat tinggi dan merupakan bagian utama dalam pembuatan halaman web selain HTML(*Hyper Text Markup Language*) dan CSS(*Cascading Style Sheet*). *Javascript* membuat halaman web menjadi lebih interaktif. Karena itu, dalam pembuatan halaman web, *Javascript* selalu digunakan. Misalnya ketika pengunjung *website* menekan tombol, maka akan muncul gambar atau tulisan. Kode *Javascript* dituliskan di dalam tag `<script>` pada HTML. Sintaks *Javascript* hampir mirip dengan bahasa pemrograman Java. Perbedaan yang cukup mencolok adalah *Javascript* dapat menampung tipe data apa saja tanpa mendefinisikan tipe data tersebut pada variabel. Gunakan `'var'` untuk mendeklarasi variabel pada *Javascript*.

jQuery adalah *library* dari *Javascript* untuk menyederhanakan *script* pada HTML⁴. Penulisan menggunakan *jQuery* lebih pendek dibandingkan *Javascript*. *jQuery* membutuhkan sumber *library*. *Library* tersebut dapat disimpan secara internal(di file komputer) atau menggunakan CDN(*Content Delivery Network*). Sintaks pada *jQuery* adalah sebagai berikut:

$$$(selector).action()^{5};$$

Penjelasan dari sintaks tersebut adalah sebagai berikut:

- Simbol `$` : untuk mengakses *jQuery*.
- *Selector* : untuk menemukan elemen HTML. Selector yang biasa digunakan adalah *selector id*(`#`) dan *selector class*(`.`).
- *Action* : hal yang akan dilakukan oleh *selector*.

Semua *method* pada *jQuery* dituliskan di dalam *document ready event*. *Document ready event* mencegah kode *jQuery* tersebut dieksekusi terlebih dahulu sebelum dokumen sudah siap⁶. Cara penulisannya terdapat pada Gambar 2.5.

```
$(document).ready(function() {  
    // tuliskan jQuery method di sini  
});
```

Gambar 2.5: *Document ready event*

Inisialisasi dan Menggambar pada Canvas

Sesudah menuliskan tag `<canvas>` pada HTML, canvas tidak bisa langsung digambar. Sebelum menggambar pada canvas, perlu dilakukan inisialisasi menggunakan *Javascript*. Berikut adalah cara menginisialisasi canvas menggunakan *Javascript* pada Gambar 2.6 dan *jQuery* pada Gambar 2.7.

Gambar 2.6: Inisialisasi menggunakan *Javascript*

```
var myCanvas = document.getElementById("canvas");  
var context = canvas.getContext("2d");
```

⁴<https://id.wikipedia.org/wiki/JQuery>

⁵https://www.w3schools.com/jquery/jquery_syntax.asp

⁶https://www.w3schools.com/jquery/jquery_syntax.asp

Gambar 2.7: Inisialisasi menggunakan *jQuery*

```
var myCanvas = $('#canvas');
var context = canvas.get(0).getContext("2d");
```

Sesudah inisialisasi canvas, canvas tersebut dapat digambar dengan bentuk 2D, garis, kurva dan membuat tulisan. Selain untuk menggambar, bentuk-bentuk tersebut dapat diberi warna sesuai dengan keinginan. Berikut adalah perintah untuk membuat bentuk-bentuk 2D:

- *fillRect()* : menggambar sekaligus mewarnai persegi.
- *strokeRect()* : menggambar sebuah persegi.
- *arc()* : menggambar lingkaran.

Membuat Objek beserta Atribut dan Methodnya

Javascript mendukung OOP (*Object Oriented Programming*) sehingga *programmer* lebih mudah mengerti cara kerja program tersebut. *Object Oriented Programming* merupakan paradigma pemrograman berdasarkan konsep objek⁷. Objek tersebut dapat berupa benda-benda yang ada pada kehidupan sehari-hari. Setiap objek memiliki atribut dan *method*. Atribut adalah data dan karakteristik yang terdapat pada objek. *Method* adalah hal yang dapat dilakukan oleh objek tersebut. Contohnya, objek mobil memiliki atribut warna mobil, merk mobil, dan berat mobil. *Method* objek mobil adalah dapat berbelok, dapat berhenti dan dapat maju.

Pada *Javascript* untuk membuat objek, digunakan perintah *function*. Pada Gambar 2.8 adalah potongan kode untuk membuat objek pada *Javascript* dengan atribut dan *method*.

```
function Mobil() {
    this.merk = 'Toyota';
    this.warna = 'Hitam';
    this.berat = 1000;

    this.belokKanan = function() {
        //mobil berbelok ke kanan
    }

    this.belokKiri = function() {
        //mobil berbelok ke kiri
    }
}
```

Gambar 2.8: Membuat objek pada *Javascript*

Berdasarkan potongan kode pada Gambar 2.8, objek mobil tersebut memiliki atribut yaitu merk mobil, warna mobil dan berat mobil. *Method* dari objek mobil adalah mobil dapat berbelok ke kanan dan ke kiri.

Events

Event adalah sebuah cara untuk membuat halaman web menjadi lebih interaktif. *Javascript* akan menjalankan suatu hal apabila *event* tersebut terdeteksi⁸. Misalnya apabila sebuah tombol ditekan

⁷https://id.wikipedia.org/wiki/Pemrograman_berorientasi_objek

⁸https://www.w3schools.com/js/js_events.asp

pada *web browser*, maka akan muncul tanggal hari ini. Pada skripsi ini, event yang digunakan adalah event milik *jQuery* dan *event keyboard* akan digunakan. Setiap tombol pada *keyboard* memiliki kode unik/*key code* tersendiri. *Event-event* yang digunakan adalah sebagai berikut:

- *keydown* : fungsi akan dieksekusi apabila pengguna menekan sebuah tombol.
- *keyup* : fungsi akan dieksekusi apabila pengguna sedang tidak menekan sebuah tombol.

Kode-kode pada *keyboard* yang digunakan adalah tombol atas, bawah, kanan dan kiri. Pada Gambar 2.9 adalah potongan kode untuk membuat *event keydown*.

```
document.addEventListener('keydown', function(e) {  
    if (e.keyCode == 37) {  
        //bergerak ke kiri  
    }  
    else if (e.keyCode == 38) {  
        //bergerak ke kanan  
    }  
    else if (e.keyCode == 39) {  
        //bergerak ke atas  
    }  
    else if (e.keyCode == 40) {  
        //bergerak ke bawah  
    }  
});
```

Gambar 2.9: *Event keydown* pada tombol atas, bawah, kiri, dan kanan

Membuat Animasi

Javascript dan *jQuery* dapat digunakan untuk membuat animasi. Animasi adalah proses pergantian gambar yang hampir berbeda satu dengan yang lainnya secara cepat sehingga gambar tersebut seolah-olah bergerak⁹. Langkah-langkah untuk membuat animasi menggunakan *canvas* adalah :

- *Draw* : gambar objek tersebut
- *Update* : *update* posisi dari objek tersebut.
- *Clear* : hapus semua objek yang ada pada *canvas*.
- *Repeat* : lakukan 3 hal diatas secara berulang.

Untuk membuat animasi, membutuhkan perintah *setTimeout*. *setTimeout* dapat membatasi *framerate* sehingga animasi tidak bergerak terlalu cepat. Sintaks untuk *setTimeout* adalah *setTimeout(x,y)*, di mana *x* adalah sebuah fungsi yang ingin dibuat animasi dan *y* adalah *framerate* dari animasi tersebut. Gambar 2.10 menunjukkan cara untuk membuat animasi. Fungsi *animate()* di baris paling akhir harus dipanggil lagi supaya animasi dapat berjalan terus-menerus.

⁹<https://en.wikipedia.org/wiki/Animation>

```

var x = 0;
function animate() {
    x++;
    context.clearRect(0, 0, canvasWidth, canvasHeight);
    context.fillRect(x, 250, 10, 10);

    setTimeout(animate, 33);
};
animate();

```

Gambar 2.10: Membuat kotak berpindah posisi

2.1.4 Github

Github adalah layanan *web hosting* bersama untuk proyek pengembangan perangkat lunak yang menggunakan sistem *version control* yaitu *Git*¹⁰. *Git* dapat mengetahui perubahan pada file di komputer. *Github* biasanya digunakan oleh para *programmer* untuk menambahkan/merevisi *source code* dan sebagai media untuk berkolaborasi dalam proyek pembangunan perangkat lunak. *Source code* tersebut disimpan dikelompokkan pada file dan file tersebut disimpan di *repository*. Dalam 1 *repository*, *owner* (pengguna yang membuat *repository*) dapat mengundang pengguna lain untuk berkolaborasi. Ketika ada perubahan pada *repository*, maka *collaborator* (pengguna yang diberi hak untuk mengubah *repository*) dapat mengetahui *source code* atau file yang direvisi oleh *collaborator* lain.

Github memiliki banyak fitur untuk memudahkan kerja *programmer*. Fitur *Github* yang mendukung penelitian ini adalah *pull request*. *Pull request* memberitahu *collaborator* lain tentang perubahan pada *repository*. Dengan adanya *pull request*, pengguna dan *collaborator* lain dapat berdiskusi tentang perubahan tersebut. Perubahan tersebut harus di *merge* oleh *collaborator* dari *repository*. Apabila sudah di *merge* oleh *collaborator*, perubahan tersebut akan disimpan di *repository*¹¹.

2.2 L^AT_EX

Mengapa menggunakan L^AT_EX 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.

¹⁰<https://en.wikipedia.org/wiki/GitHub>

¹¹<https://help.github.com/articles/about-pull-requests>

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 \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 2.1 berikut ini:

Tabel 2.1: Tabel contoh

	v_{start}	\mathcal{S}_1	v_{end}
τ_1	1	12	20
τ_2	1		20
τ_3	1	9	20
τ_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}
τ_1	1	5	12	20
τ_2	1	8		20
τ_3	1	2/8/17	9	20
τ_4	1			20

Tabel 2.3: Tabel bewarna(2)

	v_{start}	\mathcal{S}_1	\mathcal{S}_2	v_{end}
τ_1	1	12	5	20
τ_2	1		8	20
τ_3	1	9	2/8/17	20
τ_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 \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:

- 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 2.11), tipe JPG (Gambar 2.12) dan tipe PDF (Gambar 2.13)
- 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*, \LaTeX 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 2.11 ditempatkan di bagian atas halaman, walaupun penempatannya dilakukan setelah penulisan 3 paragraf setelah penjelasan ini.
 - Gambar 2.12 dengan skala 0.5 ditempatkan di antara dua buah paragraf. Perhatikan penulisannya di dalam file `bab2.tex`!
 - Gambar 2.13 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.



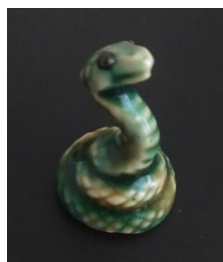
Gambar 2.11: Gambar *Serpentes* dalam format png

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 2.12: 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 2.13: *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
5 #include<stdio.h>
6
7 void myFunction( int input, float* output ) {
8     switch ( array[i] ) {
9         case 1: // This is silly code
10             if ( a >= 0 || b <= 3 && c != x )
11                 *output += 0.005 + 20050;
12             char = 'g';
13             b = 2^n + ~right_size - leftSize * MAX_SIZE;
14             c = (--aaa + &daa) / (bbb++ - ccc % 2 );
15             strcpy(a,"hello_$@?");
16         }
17         count = ~mask | 0x00FF00AA;
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