Steganografi dengan Teknik Indikasi Piksel

Gavrila Tiominar Sianturi–2013730025

1 Data Skripsi

Pembimbing utama/tunggal: **Mariskha Tri Adithia**
Pembimbing pendamping: -
Kode Topik : **MTA4304***
Topik ini sudah dikerjakan selama : **1** semester
Pengambilan pertama kali topik ini pada : Semester **43 - Ganjil 17/18**
Pengambilan pertama kali topik ini di kuliah : **Skripsi 1**
Tipe Laporan : **B -** Dokumen untuk reviewer pada presentasi dan **review Skripsi 1**

2 Detail Perkembangan Pengerjaan Skripsi

Detail bagian pekerjaan skripsi sesuai dengan rencan kerja/laporan perkembangan terkahir :

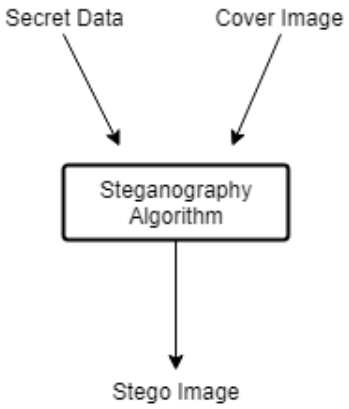
1. **Melakukan studi literatur mengenai dasar-dasar steganografi, metode steganografi dengan teknik indikasi piksel, dan metode steganografi dengan algoritma *Triple-A***
   Status : Diganti (metode *Triple-A* diganti menjadi metode *Least Significant Bit*).
   Hasil : Berikut merupakan hasil studi literatur mengenai dasar-dasar steganografi, steganografi dengan teknik *Least Significant Bit*, dan steganografi dengan teknik indikasi piksel.

   - **Steganografi**
     Steganografi merupakan seni dan ilmu untuk menyembunyikan pesan rahasia (*secret data*) di dalam suatu *cover media*. Kata steganografi berasal dari bahasa Yunani yang berarti tulisan yang tersembunyi. Tujuan dilakukannya steganografi adalah untuk mengirimkan pesan rahasia melalui suatu media, dimana media tersebut dapat dikirimkan melalui apapun dan diterima oleh siapa saja, tanpa menimbulkan kecurigaan bahwa ada pesan rahasia yang disembunyikan di dalamnya.

     Steganografi dapat dilakukan dengan menggunakan berbagai media. Media yang dapat digunakan antara lain adalah audio, video, gambar, teks, dan lain sebagainya. Media steganografi yang akan digunakan pada skripsi ini adalah gambar, dimana informasi rahasia akan disembunyikan ke dalam piksel-piksel pada gambar. Hasil dari implementasi steganografi akan berupa gambar yang di dalam piksel-pikselnya terdapat informasi rahasia. Gambar hasil implementasi steganografi disebut *stego image*. Gambar 1 memperlihatkan alur proses implementasi steganografi dengan media gambar. *Secret data* dan *cover media* (atau yang dapat juga disebut *cover image*) akan diproses oleh algoritma steganografi dan proses tersebut akan menghasilkan *stego image*.

     Steganografi dengan media gambar melibatkan beberapa aspek untuk dipertimbangkan. Berikut merupakan beberapa aspek diantaranya yang akan digunakan pada skripsi ini.

     – Kapasitas (*capacity*)
       Kapasitas merupakan jumlah data yang dapat disembunyikan pada gambar tanpa mengubah gambar tersebut secara signifikan.
     – Transparansi (*invisibility*)
       Transparansi berarti hasil dari proses penyembunyian tidak menimbulkan kecurigaan bagi pihak yang tidak berkepentingan.



Gambar 1: Steganografi dengan media gambar

– Keamanan (*security*)
  Keamanan berarti *secret data* tidak bisa didapatkan dengan mudah oleh pihak-pihak yang tidak berkepentingan.

Aspek-aspek tersebut digunakan sebagai ukuran pembanding berbagai metode steganografi. Aspek-aspek ini berpengaruh satu sama lain, seperti apabila kapasitas bertambah maka ada kemungkinan transparansi akan berkurang karena akan semakin banyak nilai piksel yang diganti.

Pada skripsi ini, *cover media* yang digunakan untuk menyembunyikan *secret data* berupa gambar berwarna atau gambar RGB. Setiap piksel pada gambar RGB terdiri dari tiga *channel* warna, yaitu merah, hijau, dan biru. Nilai setiap *channel* warna pada setiap piksel direpresentasikan dengan 8 bit angka biner, sehingga setiap piksel direpresentasikan dengan 24 bit angka biner.

- **Steganografi dengan Teknik *Least Significant Bit***
  Steganografi dengan metode *Least Significant Bit* (LSB) merupakan salah satu metode yang sudah banyak digunakan karena implementasinya yang tergolong sederhana. Penyembunyian *secret data* di dalam *cover media* dilakukan dengan cara menyisipkan *secret data*, yang sudah diubah ke dalam bentuk 8 bit angka biner ASCII, ke setiap satu bit terakhir setiap *channel* warna pada setiap piksel. Sebagai contoh, *secret data* yang ingin disembunyikan adalah A. Karakter A apabila diubah dalam bentuk biner ASCII adalah 01000001. Piksel *cover media* yang digunakan adalah:

  00100111 11101001 11001000
  00100111 11001000 11101001
  11001000 00100111 11101001

  Maka, hasil piksel setelah diimplementasikan steganografi dengan teknik LSB adalah:

  00100110 11101001 11001000
  00100110 11001000 11101000
  11001000 00100111 11101001

  Penyisipan *secret data* dilakukan pada bit terakhir setiap *channel* warna dengan tujuan supaya perubahan nilai warna pada piksel tidak besar sehingga tidak menunjukkan perbedaan yang signifikan antara *cover media* yang asli dengan *cover media* yang sudah disisipkan *secret data*.

- **Steganografi dengan Teknik Indikasi Piksel**
  Steganografi dengan teknik indikasi piksel merupakan metode pengembangan dari teknik LSB.

Sama seperti LSB, *secret data* diubah terlebih dahulu ke dalam bentuk biner ASCII. Bit-bit *secret data* juga tetap disembunyikan pada bit terakhir suatu *channel* warna, namun tidak semua *channel* warna pasti dipakai untuk menyembunyikan *secret data*. Salah satu *channel* warna akan digunakan sebagai indikator untuk menentukan penyisipan bit *secret data* pada *channel* warna lainnya.

Langkah pertama yang dilakukan dalam implementasi teknik indikasi piksel adalah menghitung panjang karakter *secret data*. Panjang karakter tersebut akan diubah dalam bentuk 8 bit angka biner dan disisipkan ke dalam bit terakhir pada 8 byte piksel pertama *cover media*. Penyisipan bit *secret data* akan dimulai pada baris piksel yang selanjutnya, sehingga akan tersisa satu *channel* warna pada piksel akhir penyisipan panjang *secret data* yang tidak disisipkan apapun.

Langkah selanjutnya dalam implementasi teknik indikasi piksel adalah menentukan *channel* warna yang menjadi indikator. *Channel* warna yang menjadi indikator dipilih berdasarkan panjang karakter *secret data*. Apabila panjang *secret data* merupakan bilangan genap, maka *channel* warna yang menjadi indikator adalah merah. Apabila panjang *secret data* merupakan bilangan prima, maka *channel* warna yang menjadi indikator adalah biru. Apabila panjang *secret data* bukan bilangan genap atau prima, maka *channel* warna yang akan menjadi indikator adalah hijau.

Panjang *secret data* juga akan digunakan untuk menentukan warna yang menjadi *channel* 1 dan *channel* 2 dalam penyisipan bit *secret data*. Panjang *secret data* akan diubah ke dalam bentuk biner untuk menentukan apakah panjang *secret data* tersebut termasuk *odd parity* atau *even parity*. Apabila jumlah bit 1 pada biner panjang *secret data* berjumlah ganjil, maka *channel* 1 dan 2 ditentukan berdasarkan kolom *odd parity*. Sebaliknya, apabila jumlah bit 1 pada biner panjang *secret data* berjumlah genap, maka *channel* 1 dan 2 ditentukan berdasarkan kolom *even parity*. Tabel 1 memperlihatkan ketentuan yang digunakan untuk menentukan *channel* indikator, *channel* 1, dan *channel* 2 yang digunakan untuk menyisipkan *secret data*.

Tabel 1: Tabel kriteria pemilihan indikator *channel* warna

Tipe panjang *secret data* (N)	Pemilihan *channel* indikator	Pemilihan *channel* 1 dan 2	
		Odd parity	Even Parity
Genap	R	GB	BG
Prima	B	RG	GR
Lainnya	G	RB	BR

Setelah ditetapkan *channel* warna yang menjadi indikator, *channel* 1, dan *channel* 2, langkah yang harus dilakukan adalah menyisipkan bit-bit *secret data* berdasarkan nilai *channel* warna indikator pada piksel tersebut. Nilai *channel* warna direpresentasikan dalam 8 bit angka biner. Dua bit terakhir nilai *channel* warna indikator digunakan untuk menentukan jumlah dan posisi penyisipan bit-bit *secret data* pada suatu piksel. Apabila dua bit terakhir pada nilai *channel* warna adalah 00, maka tidak ada bit *secret data* yang disembunyikan dalam *channel* 1 dan 2 pada piksel tersebut. Apabila dua bit terakhir pada nilai *channel* warna indikator adalah 01, maka tidak ada bit *secret data* yang disembunyikan dalam *channel* 1 piksel tersebut dan ada 2 bit data yang disembunyikan dalam *channel* 2 piksel tersebut. Apabila dua bit terakhir pada nilai *channel* warna indikator adalah 10, maka ada 2 bit data yang disembunyikan dalam *channel* 1 piksel tersebut dan tidak ada bit *secret data* yang disembunyikan dalam *channel* 2 piksel tersebut. Apabila dua bit terakhir pada nilai *channel* warna indikator adalah 11, maka ada 2 bit data yang disembunyikan dalam *channel* 1 piksel tersebut dan ada 2 bit data yang disembunyikan dalam

channel 2 piksel tersebut. Tabel 2 memperlihatkan bagaimana data akan disisipkan dalam satu piksel berdasarkan dua bit terakhir nilai *channel* warna yang menjadi indikator.

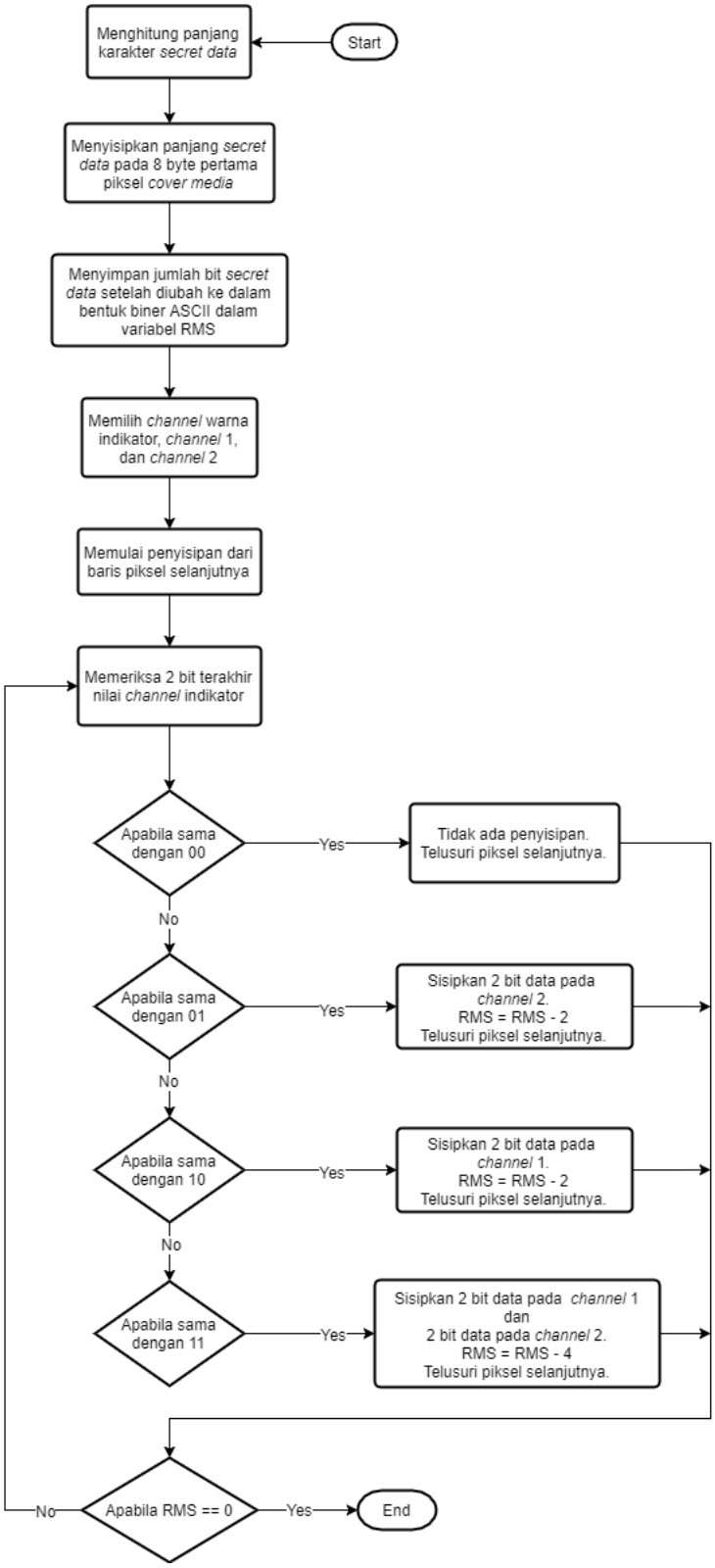
Tabel 2: Tabel kriteria pemilihan indikator *channel* warna

Channel indikator	*Channel* 1	*Channel* 2
00	Tidak ada data yang disisipkan	Tidak ada data yang disisipkan
01	Tidak ada data yang disisipkan	Disisipkan 2 bit data
10	Disisipkan 2 bit data	Tidak ada data yang disisipkan
11	Disisipkan 2 bit data	Disisipkan 2 bit data

Penyisipan bit-bit *secret data* dilakukan dengan menggunakan iterasi. Dalam satu piksel dapat disisipkan 0 sampai 4 bit *secret data* yang ditentukan berdasarkan nilai *channel* indikator. Oleh karena penyisipan bit-bit *secret data* dilakukan secara iterasi, maka dibutuhkan suatu variabel untuk menentukan kondisi berhenti iterasi tersebut. Variabel *Remaining Message Size* (RMS) digunakan untuk menentukan kondisi berhenti dengan memeriksa apakah seluruh bit *secret data* sudah disisipkan pada *cover media* atau belum. Variabel RMS pada awalnya diinisialisasi dengan jumlah bit *secret data* yang akan disisipkan. Setiap iterasi penyisipan bit-bit *secret data* pada piksel, variabel RMS akan dikurangi dengan jumlah bit *secret data* yang disisipkan pada piksel tersebut. Iterasi penyisipan akan berhenti ketika variabel RMS sudah bernilai 0, dimana seluruh bit *secret data* sudah disisipkan.

Gambar 2 memperlihatkan cara implementasi teknik indikasi piksel secara keseluruhan. Berikut merupakan penjelasan setiap langkah yang terdapat pada ilustrasi teknik indikasi piksel dalam Gambar 2.

- Menghitung panjang karakter *secret data*
  Panjang *secret data* dihitung per karakter.
- Menyisipkan panjang *secret data* pada 8 byte pertama piksel *cover media*
  Panjang *secret data* diubah ke dalam bentuk 8 bit angka biner kemudian disisipkan dalam bit terakhir 8 byte pertama piksel *cover media*.
- Menyimpan jumlah bit *secret data* setelah diubah ke dalam bentuk biner ASCII dalam variabel RMS
  Secret data diubah ke dalam bentuk biner ASCII. Masing-masing karakter direpresentasikan dalam 8 bit angka biner. Jumlah seluruh bit bentuk ASCII *secret data* dimasukkan ke dalam variabel RMS.
- Memilih *channel* warna indikator, *channel* 1, dan *channel* 2
  Memilih *channel* warna yang menjadi indikator, *channel* 1, dan *channel* 2 berdasarkan panjang *secret data*. *Channel* indikator, *channel* 1, dan *channel* 2 dipilih berdasarkan ketentuan pada Tabel 1.
- Memeriksa 2 bit terakhir nilai *channel* indikator
  Melakukan penyisipan bit *secret data* berdasarkan dua bit terakhir nilai *channel* indikator. Berikut merupakan proses yang dilakukan berdasarkan dua bit terakhir nilai *channel* indikator.
  - Apabila bernilai 00, tidak akan ada perubahan maka akan langsung dilakukan penelusuran piksel selanjutnya apabila variabel RMS belum bernilai 0. Apabila variabel RMS sudah bernilai 0, maka iterasi akan berhenti.
  - Apabila bernilai 01, dilakukan penyisipan 2 bit *secret data* pada 2 bit terakhir *channel* 2. Kemudian variabel RMS akan dikurangi dengan 2 karena pada piksel tersebut terdapat 2



Gambar 2: Algoritma implementasi teknik indikasi piksel

bit *secret data* yang disembunyikan. Setelah itu, dilakukan penelusuran piksel selanjutnya apabila variabel RMS belum bernilai 0. Apabila variabel RMS sudah bernilai 0, maka iterasi akan berhenti.

* Apabila bernilai 10, dilakukan penyisipan 2 bit *secret data* pada 2 bit terakhir *channel* 1. Kemudian variabel RMS akan dikurangi dengan 2 karena pada piksel tersebut terdapat 2 bit *secret data* yang disembunyikan. Setelah itu, dilakukan penelusuran piksel selanjutnya apabila variabel RMS belum bernilai 0. Apabila variabel RMS sudah bernilai 0, maka iterasi akan berhenti.
* Apabila bernilai 11, dilakukan penyisipan 2 bit *secret data* pada 2 bit terakhir *channel* 1 dan 2 bit *secret data* pada 2 bit terakhir *channel* 2. Kemudian variabel RMS akan dikurangi dengan 4 karena pada piksel tersebut terdapat 4 bit *secret data* yang disembunyikan. Setelah itu, dilakukan penelusuran piksel selanjutnya apabila variabel RMS belum bernilai 0. Apabila variabel RMS sudah bernilai 0, maka iterasi akan berhenti.

2. **Mengimplementasikan teknik indikasi piksel dan algoritma *Triple-A* secara manual**
   Status : Diganti (metode *Triple-A* diganti menjadi metode *Least Significant Bit*).
   Hasil : Berikut merupakan contoh implementasi manual steganografi dengan teknik LSB dan teknik indikasi piksel.

   Secret data : HELLO
   Panjang *secret data* (N) : 5
   Contoh nilai-nilai setiap *channel* warna pada piksel-piksel *cover media* diperlihatkan pada Tabel 3.

Tabel 3: Tabel nilai piksel pada *cover media*

R	G	B
00100000	01011101	10100011
00111111	01011101	10011011
00100001	01011110	10100001
00100010	01011101	10011010
00100101	01011111	10011011
00100111	01100001	10100011
00101000	01100001	10100110
00100101	01100000	10100011
00100111	01100011	10100010
00101001	01100110	10100111
00101101	01101000	10101011
00110011	01101110	10110010
00110001	01101011	10110001
00110001	01100010	10110011
00110110	01010010	10011011
00101000	01010110	10011001
00101010	01010110	10010100
00100010	01000010	10010001
00110101	01000110	10010111

Langkah pertama yang dilakukan untuk mengimplementasikan kedua metode steganografi ini adalah dengan mengubah *secret data* ke dalam bentuk biner ASCII. Setiap karakter diubah ke dalam bentuk 8 bit biner ASCII. Pada contoh ini, kata "HELLO" apabila diterjemahkan ke dalam bentuk biner ASCII menjadi 01001000 01000101 01001100 01001100 01001111. Setiap karakter pada *secret data* diubah

dalam bentuk biner ASCII supaya dapat disisipkan ke dalam nilai-nilai piksel yang juga berupa angka biner.

- **Implementasi dengan Teknik LSB**
  Penyisipan dengan metode LSB dilakukan dengan mengganti setiap satu bit terakhir pada setiap *channel* warna dalam setiap piksel dengan satu bit *secret data* secara berurutan. Tabel 4 menunjukkan perubahan nilai piksel hasil implementasi steganografi dengan teknik LSB.

Tabel 4: Tabel perbandingan nilai piksel awal dan nilai piksel setelah disisipkan bit *secret data* dengan metode LSB

R	G	B
00100000	01011101	10100011
00111111	01011101	10011011
00100001	01011110	10100001
00100010	01011101	10011010
00100101	01011111	10011011
00100111	01100001	10100011
00101000	01100001	10100110
00100101	01100000	10100011
00100111	01100011	10100010
00101001	01100110	10100111
00101101	01101000	10101011
00110011	01101110	10110010
00110001	01101011	10110001
00110001	01100010	10110011
00110110	01010010	10011011
00101000	01010110	10011001
00101010	01010110	10010100
00100010	01000010	10010001
00110101	01000110	10010111

R	G	B
00100000	01011101	10100010
00111110	01011101	10011010
00100000	01011110	10100000
00100011	01011100	10011010
00100100	01011111	10011010
00100111	01100000	10100011
00101000	01100000	10100111
00100101	01100000	10100010
00100110	01100011	10100010
00101000	01100111	10100111
00101100	01101000	10101010
00110011	01101110	10110010
00110001	01101011	10110001
00110001	01100010	10110011
00110110	01010010	10011011
00101000	01010110	10011001
00101010	01010110	10010100
00100010	01000010	10010001
00110101	01000110	10010111

- **Implementasi dengan Teknik Indikasi Piksel**
  Langkah pertama yang dilakukan pada implementasi teknik indikasi piksel adalah menyimpan panjang *secret data*. Panjang *secret data* akan disimpan dalam 8 bit angka biner karena panjang *secret data* akan disimpan pada setiap bit terakhir 8 byte piksel pertama pada *cover media*. Pada contoh ini, panjang *secret data* adalah 5 karakter, dalam bentuk binernya 00000101. Tabel 5 memperlihatkan hasil penyisipan panjang *secret data* pada 8 byte pertama piksel *cover media*.

Tabel 5: Tabel perbandingan nilai piksel awal dan nilai piksel setelah disisipkan panjang *secret data*

R	G	B
0010000**0**	0101110**0**	1010001**0**
0011111**0**	0101110**0**	1001101**1**
0010000**0**	0101111**1**	10100001

R	G	B
00100000	01011101	10100011
00111111	01011101	10011011
00100001	01011110	10100001

Langkah kedua dalam implementasi teknik indikasi piksel adalah menentukan *channel* warna yang menjadi indikator. Dua bit terakhir nilai *channel* warna indikator pada suatu piksel menjadi penentu letak penyisipan bit *secret data* pada piksel tersebut. *Channel* warna yang menjadi indikator dipilih berdasarkan panjang *secret data*. Ketentuan pemilihan *channel* warna yang menjadi indikator terdapat pada Tabel 1. Pada contoh ini, panjang *secret data* adalah 5. Oleh karena

5 adalah bilangan prima, maka *channel* warna yang menjadi indikator adalah biru.

Setelah ditentukan *channel* warna yang menjadi indikator, selanjutnya adalah menentukan *channel* warna mana yang menjadi *channel* 1 dan 2. Pada *channel* 1 dan 2 akan disisipkan bit *secret data* berdasarkan nilai pada *channel* warna indikator. Pemilihan *channel* 1 dan 2 ditentukan berdasarkan apakah panjang *secret data* (dalam bentuk biner) termasuk *odd* atau *even parity*. Panjang *secret data* termasuk *odd parity* apabila jumlah bit 1 pada binernya berjumlah ganjil. Sedangkan, panjang *secret data* termasuk *even parity* apabila jumlah bit 1 pada binernya berjumlah genap. Pada contoh ini, biner dari panjang *secret data* adalah 00000101. Jumlah bit 1 pada biner tersebut adalah genap. Berdasarkan Tabel 1, apabila indikator adalah biru dan panjang *secret data* termasuk *even parity*, maka *channel* 1 adalah hijau dan *channel* 2 adalah merah.

Langkah terakhir adalah penyisipan bit-bit *secret data* berdasarkan nilai *channel* warna indikator. Tabel 6 memperlihatkan perubahan nilai piksel hasil implementasi steganografi dengan indikator biru, *channel* 1 hijau, dan *channel* 2 merah. Tiga baris piksel pertama pada Tabel 6 sebelah kanan, yang dicetak tebal, digunakan untuk menyimpan panjang karakter *secret data* dalam bentuk biner. Kemudian penyisipan bit *secret data* dimulai pada baris piksel ke-4. Dua bit terakhir pada kolom B yang dicetak tebal merupakan nilai yang menjadi indikator letak penyisipan bit pada piksel tersebut berdasarkan ketentuan pada Tabel 2. Bit-bit pada kolom R dan G yang diberi garis bawah merupakan bit-bit *secret data* yang tersimpan pada nilai-nilai *channel* warna R ataupun G. Bit *secret data* disisipkan secara terurut berdasarkan urutan *channel* 1 dan *channel* 2.

Tabel 6: Tabel perbandingan nilai piksel awal dan nilai piksel setelah disisipkan bit *secret data* dengan teknik indikasi piksel

R	G	B
00100000	01011101	10100011
00111111	01011101	10011011
00100001	01011110	10100001
00100010	01011101	10011010
00100101	01011111	10011011
00100111	01100001	10100011
00101000	01100001	10100110
00100101	01100000	10100011
00100111	01100011	10100010
00101001	01100110	10100111
00101101	01101000	10101011
00110011	01101110	10110010
00110001	01101011	10110001
00110001	01100010	10110011
00110110	01010010	10011011
00101000	01010110	10011001
00101010	01010110	10010100
00100010	01000010	10010001
00110101	01000110	10010111

R	G	B
00100000	01011100	10100010
00111110	01011100	10011011
00100000	01011111	10100001
00100010	01011101	10011010
00100110	01011100	10011011
00100101	01100000	10100011
00101000	01100000	10100010
00100101	01100001	10100011
00100111	01100001	10100010
00101011	01100100	10100111
00101101	01101000	10101011
00110011	01101100	10110010
00110011	01101011	10110001
00110001	01100000	10110011
00110111	01010000	10011011
00101011	01010110	10011001
00101010	01010110	10010100
00100010	01000010	10010001
00110101	01000110	10010111

3. **Melakukan analisis kebutuhan**
   Status : Ada sejak rencana kerja skripsi.
   Hasil : Berikut merupakan hasil analisis terhadap metode LSB dan teknik indikasi piksel beserta analisis kebutuhan pada perangkat lunak.

Steganografi dengan Metode LSB

Implementasi steganografi dengan metode LSB tergolong sangat sederhana. Bit-bit *secret data* disisipkan pada bit-bit terakhir setiap nilai *channel* warna pada setiap piksel secara berurutan. Implementasinya yang terlalu sederhana membuat bit-bit *secret data* bisa didapatkan oleh pihak-pihak yang tidak berkepentingan dengan mudah. Oleh karena itu, steganografi dengan metode LSB kurang baik dari segi keamanan.

Berdasarkan contoh implementasi LSB pada Tabel 4, didapatkan rata-rata jumlah bit *secret data* yang dapat disembunyikan dalam setiap piksel adalah 2,86. Rata-rata jumlah bit diperoleh dengan menggunakan persamaan 1.

$$Rata - rata\ jumlah\ bit\ secret\ data\ per\ piksel = \frac{\sum bit\ secret\ data}{\sum piksel\ yang\ dipakai\ untuk\ penyisipan} \quad (1)$$

Steganografi dengan Teknik Indikasi Piksel

Steganografi dengan teknik indikasi piksel menggunakan prinsip yang sama dengan metode LSB dengan menyisipkan bit-bit *secret data* pada bit-bit terakhir *cover media*. Kelebihan dari steganografi dengan teknik indikasi piksel adalah pola penyembunyian tidak pasti, melainkan bergantung pada nilai *channel* warna yang menjadi indikator. Selain itu, *channel* warna yang menjadi indikator juga ditentukan berdasarkan panjang *secret data*, dimana akan berbeda-beda dalam setiap contoh kasusnya. Berdasarkan aspek keamanan, teknik ini tentu lebih baik daripada LSB karena polanya tidak dapat ditebak semudah menebak pola LSB.

Rata-rata jumlah bit *secret data* yang dapat disembunyikan pada setiap piksel berdasarkan Tabel 6 adalah 3,07. Jumlah ini pun lebih besar daripada kapasitas penyisipan bit *secret data* dengan metode LSB. Oleh karena itu, berdasarkan aspek-aspek yang dipertimbangkan pada skripsi ini, teknik indikasi piksel dapat dikatakan lebih baik daripada metode LSB.

Modifikasi Steganografi dengan Teknik Indikasi Piksel

Teknik indikasi piksel memang lebih baik daripada metode LSB. Namun, terdapat kelemahan pada teknik ini. Pemilihan *channel* warna yang menjadi indikator serta *channel* 1 dan 2 ditentukan berdasarkan panjang *secret data*. Sementara panjang *secret data* dapat dengan mudah diketahui karena disimpan pada 8 byte pertama pada piksel *cover media*. Oleh karena itu, dengan implementasi seperti ini bit-bit *secret data* masih dapat ditebak oleh pihak-pihak yang tidak berkepentingan.

Pada skripsi ini, akan dilakukan sedikit modifikasi terhadap implementasi steganografi dengan teknik indikasi piksel. Agar bit-bit *secret data* lebih sulit untuk didapatkan oleh pihak-pihak yang tidak berkepentingan, maka panjang *secret data* tidak akan disisipkan ke dalam nilai-nilai piksel. Selain itu, akan dibangkitkan angka acak untuk menentukan baris piksel awal yang digunakan untuk penyisipan bit-bit *secret data*. Angka acak ini tidak ikut disisipkan ke dalam *cover media*. Dengan modifikasi ini, bit-bit *secret data* akan lebih sulit ditebak oleh pihak-pihak yang tidak berkepentingan.

Analisis Kebutuhan Perangkat Lunak

Pada skripsi ini akan dibangun perangkat lunak yang dapat mengimplementasikan steganografi dengan

teknik LSB dan teknik indikasi piksel. Berdasarkan studi literatur dan implementasi manual, dapat diketahui cara penyisipan *secret data* dengan menggunakan teknik LSB dan teknik indikasi piksel. Untuk dapat mengimplementasikan kedua metode ini, diperlukan *secret data* dan *cover media*. Oleh karena itu, perangkat lunak membutuhkan masukan berupa teks sebagai *secret data* dan gambar RGB sebagai *cover media*. *Secret data* akan disisipkan pada *cover media* dalam bentuk bit-bit angka biner, sehingga perangkat lunak akan mengubah teks *secret data* ke dalam bentuk biner ASCII. Perangkat lunak juga akan melakukan ekstraksi nilai setiap *channel* warna pada setiap piksel. Nilai-nilai tersebut dibutuhkan untuk melakukan penyisipan bit-bit *secret data*.

Pengguna kemudian akan memilih metode steganografi yang ingin digunakan. Apabila pengguna memilih metode LSB, perangkat lunak akan melakukan penyisipan setiap bit *secret data* pada setiap bit terakhir nilai *channel* warna setiap piksel. Nilai-nilai piksel setelah dilakukan penyisipan bit-bit *secret data* akan disimpan. Perangkat lunak akan menghitung kapasitas berdasarkan persamaan 1 pada bagian sebelumnya. Kapasitas dihitung untuk menjadi ukuran pembanding teknik LSB dengan teknik indikasi piksel. Perangkat lunak kemudian akan menampilkan *stego object* berupa gambar dengan nilai-nilai piksel setelah dilakukan penyisipan beserta hasil perhitungan kapasitas.
Apabila pengguna memilih teknik indikasi piksel, maka akan diimplementasikan teknik indikasi piksel dengan modifikasi. Perangkat lunak akan menghitung panjang karakter *secret data* yang digunakan untuk menentukan *channel* warna yang menjadi indikator, *channel* 1, dan *channel* 2. Perangkat lunak kemudian akan mengidentifikasi apakah panjang karakter tersebut termasuk bilangan genap, prima, atau yang lainnya. Selain itu, perangkat lunak juga akan menentukan apakah panjang karakter tersebut termasuk *odd* atau *even parity*. Kemudian perangkat lunak akan menentukan *channel* warna yang menjadi indikator, *channel* 1, dan *channel* 2 berdasarkan Tabel 1. Pada penerapan teknik indikasi piksel yang belum dimodifikasi, panjang karakter *secret data* akan diubah ke dalam bentuk biner dan disisipkan ke dalam 8 byte piksel pertama. Namun, penyisipan panjang *secret data* di dalam *cover media* dapat memberikan informasi mengenai *channel* warna yang menjadi indikator, *channel* 1, dan *channel* 2, sehingga *secret data* dapat ditebak dengan mudah. Oleh karena itu, pada penerapan teknik indikasi piksel dengan modifikasi, penyisipan panjang karakter *secret data* tidak akan dilakukan. Selain itu, perangkat lunak akan diminta membangkitkan angka acak untuk menentukan baris piksel awal penyisipan bit *secret data*. Panjang *secret data* dan angka acak akan akan digunakan sebagai kunci rahasia. Kunci rahasia tersebut harus diketahui oleh pengguna karena akan digunakan untuk melakukan ekstraksi *secret data* dari *stego object*. Perangkat lunak selanjutnya akan melakukan penyisipan bit berdasarkan nilai *channel* warna indikator seperti pada ketentuan Tabel 2. Nilai-nilai *channel* warna yang baru akan didapatkan setelah dilakukan penyisipan seluruh bit *secret data*. Sama seperti implementasi teknik LSB, perangkat lunak akan melakukan perhitungan kapasitas *secret data* yang dapat disisipkan pada *cover media*. Gambar dengan nilai-nilai piksel baru dan hasil perhitungan kapasitas akan ditampilkan oleh perangkat lunak.

Dari implementasi kedua metode steganografi tersebut, didapatkan kapasitas sebagai ukuran pembanding kedua metode. Semakin besar kapasitas *secret data* yang dapat disisipkan ke dalam *cover media* tanpa menimbulkan perubahan yang signifikan, makan semakin baik metode tersebut. Gambar 3 menunjukkan aliran proses implementasi steganografi dengan teknik LSB dan teknik indikasi piksel pada perangkat lunak.

Berikut merupakan deskripsi setiap langkah proses implementasi steganografi yang terdapat pada Gambar 3.

- Memasukkan gambar sebagai *cover media* dari direktori

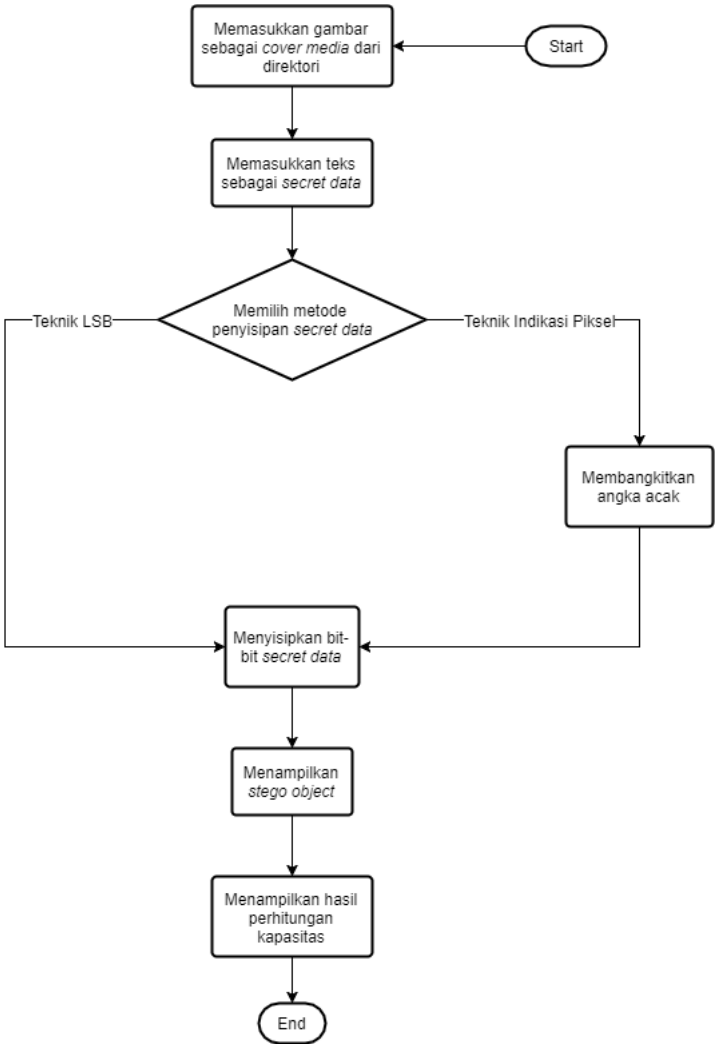
Pengguna memilih gambar sebagai input yang akan menjadi *cover media* dari direktori.

- Memasukkan teks sebagai *secret data*
  Pengguna memasukkan input berupa teks yang akan disembunyikan di dalam input gambar. Teks tersebut nantinya akan diubah oleh perangkat lunak ke dalam bentuk biner ASCII untuk disisipkan ke dalam *cover media*.
- Memilih metode penyisipan *secret data*
  Pengguna memilih apakah metode LSB atau teknik indikasi piksel yang akan digunakan untuk menyisipkan *secret data* di dalam *cover media*.
- Membangkitkan angka acak
  Apabila memilih metode teknik indikasi piksel, perangkat lunak akan membangkitkan angka acak, dimana angka acak tersebut akan digunakan untuk menentukan baris piksel awal penyisipan bit *secret data*. Pengguna harus mengetahui angka acak tersebut supaya dapat melakukukan ekstraksi.
- Menyisipkan bit *secret data*
  Perangkat lunak melakukan penyisipan bit *secret data* ke dalam *cover media* dengan metode yang sudah dipilih. Apabila pengguna memilih metode LSB, perangkat lunak langsung melakukan penyisipan bit berdasarkan metode LSB. Apabila pengguna memilih teknik indikasi piksel, perangkat lunak terlebih dahulu menentukan *channel* yang menjadi indikator, *channel* 1, dan *channel* 2. Kemudian perangkat lunak melakukan penyisipan bit berdasarkan teknik indikasi piksel dengan modifikasi. Perhitungan kapasitas kedua metode steganografi dilakukan bersamaan dengan penyisipan bit-bit *secret data*.
- Menampilkan *stego object*
  Perangkat lunak menampilkan hasil penyisipan bit-bit *secret data* ke dalam *cover media* dalam bentuk gambar.
- Menampilkan hasil perhitungan kapasitas *secret data* dalam *cover media* Perangkat lunak akan menampilkan hasil perhitungan kapasitas *secret data* yang akan menjadi ukuran pembanding kedua metode.

Ekstraksi *secret data* dapat dilakukan setelah mendapatkan *stego object* hasil implementasi steganografi dengan salah satu dari kedua metode. Perangkat lunak membutuhkan masukan berupa gambar yang ingin diekstraksi. Gambar tersebut harus merupakan hasil implementasi salah satu metode steganografi. Apabila gambar tersebut merupakan hasil implementasi LSB, maka ekstraksi harus dilakukan dengan metode LSB. Apabila gambar tersebut merupakan hasil implementasi teknik indikasi piksel, maka ekstraksi harus dilakukan dengan teknik indikasi piksel. Perangkat lunak terlebih dahulu harus mengekstraksi nilai-nilai *channel* warna pada setiap piksel gambar *stego object*. Nilai-nilai *channel* warna akan diubah ke dalam bentuk 8 bit angka biner.

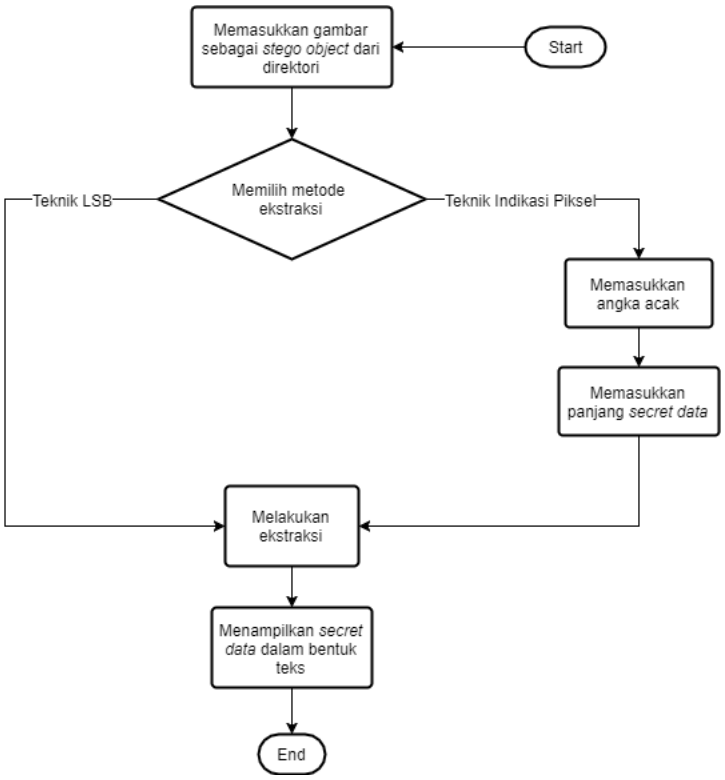
Apabila ekstraksi dilakukan dengan metode LSB, perangkat lunak akan menelusuri piksel-piksel pada *stego object* dan menyimpan bit-bit terakhir pada setiap nilai *channel* warna dalam bentuk biner. Bit-bit tersebut akan disimpan dan nantinya diubah ke dalam bentuk teks berdasarkan kode ASCII. Perangkat lunak akan menampilkan *secret data* hasil ekstraksi dalam bentuk teks.

Apabila ekstraksi dilakukan dengan teknik indikasi piksel dengan modifikasi, maka perangkat lunak akan meminta pengguna memasukkan angka yang menentukan baris piksel awal yang ingin ditelusuri. Selain itu, perangkat lunak juga akan meminta masukan berupa panjang karakter *secret data*. Perangkat lunak kemudian akan menentukan *channel* yang menjadi indikator, *channel* 1, dan *channel* 2.



Gambar 3: *Flowchart* proses implementasi steganografi pada perangkat lunak

Bit-bit *secret data* akan disimpan berdasarkan nilai *channel* yang menjadi indikator pada setiap baris piksel. Bit-bit *secret data* akan diubah ke dalam bentuk teks berdasarkan kode ASCII. Perangkat lunak akan menampilkan *secret data* hasil ekstraksi dalam bentuk teks. Gambar 4 menunjukkan aliran proses ekstraksi *secret data* dengan teknik LSB dan teknik indikasi piksel pada perangkat lunak.



Gambar 4: *Flowchart* proses ekstraksi pada perangkat lunak

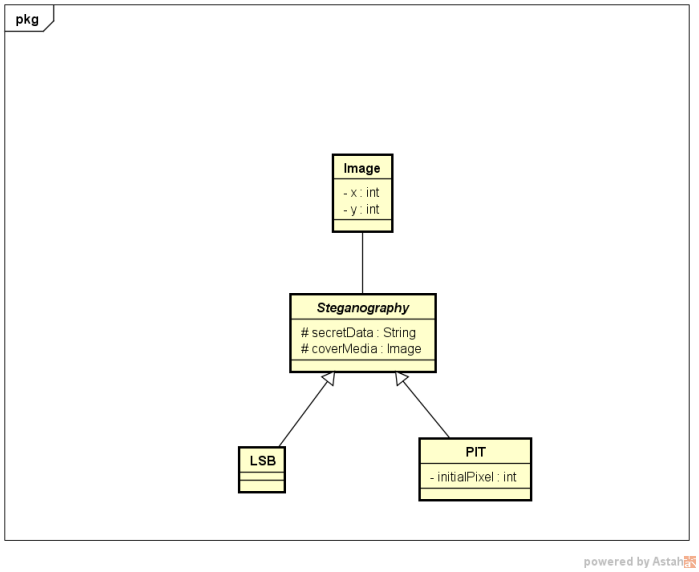
Berikut merupakan deskripsi setiap langkah proses ekstraksi *secret data* yang terdapat pada Gambar 4.

- Memilih gambar dari direktori
  Pengguna memilih gambar dari direktori untuk dilakukan ekstraksi.
- Memilih metode ekstraksi
  Pengguna memilih apakah metode LSB atau teknik indikasi piksel yang akan digunakan untuk melakukan ekstraksi.
- Memasukkan angka acak
  Apabila memilih metode teknik indikasi piksel, maka pengguna harus memasukkan angka yang menyatakan baris piksel awal dilakukan ekstraksi. Angka ini didapatkan dari hasil pembangkitan angka acak pada proses implementasi steganografi.
- Memasukkan panjang *secret data*
  Selain memasukkan angka acak, pengguna juga harus memasukkan panjang *secret data* yang merupakan salah satu kunci untuk melakukan ekstraksi. Melalui panjang *secret data*, perangkat lunak dapat menentukan *channel* warna yang menjadi indikator, *channel* 1, dan *channel* 2.
- Melakukan ekstraksi
  Perangkat lunak melakukan ekstraksi terhadap gambar untuk mendapatkan bit *secret data* dengan menggunakan metode yang sudah dipilih.

- Menampilkan *secret data* berupa teks
  Perangkat lunak menyimpan bit-bit *secret data* hasil ekstraksi gambar, kemudian menampilkannya setelah diubah dalam bentuk teks.

4. **Melakukan perancangan perangkat lunak**
   Status : Ada sejak rencana kerja skripsi.
   Hasil : Perancangan perangkat lunak yang diselesaikan pada skripsi 1 adalah diagram kelas awal. Berikut merupakan hasil perancangan perangkat lunak yang diselesaikan pada skripsi 1.

   Diagram Kelas Awal
   Diagram kelas awal merupakan gambaran rancangan awal hubungan antara kelas satu dan lainnya yang digunakan untuk membangun perangkat lunak. Gambar 5 menunjukkan bahwa kelas yang akan digunakan berjumlah 4 kelas. Berikut merupakan rincian kelas beserta atribut-atributnya.



Gambar 5: Diagram kelas awal

(a) Kelas Image
Kelas Image merepresentasikan *cover media* yang digunakan pada penyisipan bit *secret data*. Kelas ini membantu kelas Steganography untuk mengakses nilai-nilai piksel pada *cover media*. Kelas Image memiliki dua buah atribut, antara lain:

- Atribut x : Atribut x merepresentasikan nilai absis dalam koordinat letak piksel pada gambar.
- Atribut y : Atribut y merepresentasikan nilai ordinat dalam koordinat letak piksel pada gambar.

(b) Kelas Steganography
Kelas Steganography merupakan kelas utama yang mengimplementasikan metode penyisipan bit. Kelas ini bertipe abstrak dan memiliki dua *subclass*,yaitu kelas LSB dan kelas PIT. Kelas Steganography memiliki dua atribut, antara lain:

- Atribut secretData : Atribut secretData merepresentasikan *secret data* yang akan disembunyikan dalam tipe String.
- Atribut coverMedia : Atribut coverMedia merepresentasikan *cover media* yang akan digunakan untuk menyembunyikan *secret data* dalam tipe Image.

(c) Kelas LSB
Kelas LSB merupakan kelas turunan dari kelas Steganography yang akan mengimplementasikan penyisipan bit *secret data* pada *cover media* dengan metode LSB.

(d) Kelas PIT
Kelas PIT merupakan kelas turunan dari kelas Steganography yang akan mengimplementasikan penyisipan bit *secret data* pada *cover media* dengan teknik indikasi piksel.

5. **Mengimplementasikan teknik indikasi piksel dan algoritma *Triple-A* pada perangkat lunak**
Status : Diganti (metode *Triple-A* diganti menjadi metode *Least Significant Bit*).
Hasil :

6. **Melakukan pengujian teknik indikasi piksel dan algoritma *Triple-A***
Status : Diganti (metode *Triple-A* diganti menjadi metode *Least Significant Bit*).
Hasil :

7. **Melakukan analisis terhadap hasil pengujian**
Status : Ada sejak rencana kerja skripsi.
Hasil :

8. **Menulis dokumen skripsi**
Status : Ada sejak rencana kerja skripsi.
Hasil :

3 Pencapaian Rencana Kerja

Persentase penyelesaian skripsi sampai dengan dokumen ini dibuat dapat dilihat pada tabel berikut :

1*	2*(%)	3*(%)	4*(%)	5*	6*(%)
1	15	15			15
2	5	5			5
3	10	10			10
4	10	5	5	Diagram kelas awal dikerjakan pada skripsi 1.	5
5	25		25		
6	10		10		
7	5		5		
8	20	5	15	Pendahuluan, dasar teori, dan analisis dikerjakan pada skripsi 1.	5
Total	100	40	60		40

Keterangan (*)
1 : Bagian pengerjaan Skripsi (nomor disesuaikan dengan detail pengerjaan di bagian 5)
2 : Persentase total
3 : Persentase yang akan diselesaikan di Skripsi 1
4 : Persentase yang akan diselesaikan di Skripsi 2
5 : Penjelasan singkat apa yang dilakukan di S1 (Skripsi 1) atau S2 (skripsi 2)
6 : Persentase yang sidah diselesaikan sampai saat ini

Bandung, 28/11/2017

Gavrila Tiominar Sianturi

Menyetujui,

Nama: Mariskha Tri Adithia
Pembimbing Tunggal