

kompresi *lossless*. Ide dasar dari metode *RLE* adalah menyingkat n -*pixel* yang berdampingan menjadi pasangan nilai nd , dimana n menyatakan banyaknya perulangan *pixel* yang sama, dan d menyatakan nilai *pixel* tersebut.

Implementasi Algoritma *Run Length Encoding (RLE)*

Kita coba implementasikan dengan citra grayscale ukuran 6x6 bit (8 derajat keabuan) sebagai berikut:

```
1 1 1 7 1 3
4 4 6 1 2 2
7 7 7 5 5 5
6 4 4 2 2 2
5 5 2 2 2 1
2 3 3 3 0 0
```

Kode *RLE* merupakan pasangan intensitas warna dan banyaknya intensitas warna yang muncul berurutan. Sehingga di dapat kode *RLE* nya adalah sebagai berikut:

```
1 3 7 1 1 1 3 1 4 2 6 1 1 1 2 2 7 3 5 3 6 1 4 2 2 3
5 2 2 3 1 1 2 1 3 3 0 2
```

Cara membaca kode diatas adalah : intensitas warna 1 muncul 3 kali, intensitas warna 7 muncul 1 kali, intensitas warna 1 muncul 1 kali, intensitas warna 3 muncul 1 kali, dan seterusnya. Cocokkan dengan citra aslinya.

- a. *RLE* merupakan metode kompresi yang banyak didukung oleh format file gambar seperti: TIFF, BMP, PCX.
- b. *RLE* bekerja dengan mengurangi ukuran fisik dengan adanya pengulangan string dari deretan karakter / byte data.
- c. String perulangan ini dinamakan RUN, dan biasanya di kodekan dalam 2 byte. Byte pertama merupakan jumlah perulangan dan byte kedua adalah karakter yang diulang.

Cocok untuk pemampatan citra yang memiliki kelompok pixel berderajat keabuan yang sama.

Contoh citra 10x10 dengan 8 derajat

0	0	0	0	0	2	2	2	2	2
0	0	0	1	1	1	1	2	2	2
1	1	1	1	1	1	1	1	1	1
4	4	4	4	3	3	3	3	2	2
3	3	3	5	5	7	7	7	7	6
2	2	6	0	0	0	0	1	1	0
3	3	4	4	3	2	2	2	1	1
0	0	0	0	0	0	0	0	1	1
1	1	1	1	0	0	0	2	2	2
3	3	3	2	2	2	1	1	1	1

(0, 5), (2, 5)

(0, 3), (1, 4), (2, 3)

(1, 10)

(4, 4), (3, 4), (2, 2)

(3, 3), (5, 2), (7, 4), (6, 1)

(2, 2), (6, 1), (0, 4), (1, 2), (0, 1)

(3, 2), (4, 2), (3, 1), (2, 2), (1, 2)

(0, 8), (1, 2)

(1, 4), (0, 3), (2, 3)

(3, 3), (2, 3), (1, 4)

Pasangan derajat keabuan (p) dan jumlah pixel (q)

Ukuran citra sebelum dimampatkan (1 derajat keabuan = 3 bit) adalah

$100 \times 3 \text{ bit} = 300 \text{ bit}$

Ukuran citra setelah pemampatan (run length=4) adalah $(31 \times 3) + (31$

$\times 4) \text{ bit} = 217 \text{ bit}$

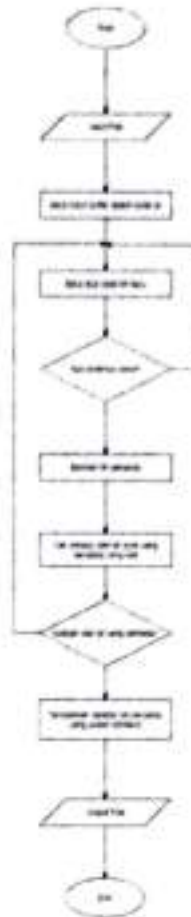
Nisbah pemampatan = $100\% - \frac{217}{300} \times 100\% = 27.67\%$, yang artinya

27.67% (Sumber : Ahmad.2012)

Flowchart Algoritma Run Length Encoding (RLE)

Pada gambar dibawah ini merupakan dari proses kompresi algoritma

Run Length Encoding (RLE)



Gambar 1.1 *Flowchart Algoritma Run Length Encoding*
(Sumber : Maria.2013)

Dari gambaran diatas diperoleh keterangan sebagai berikut :

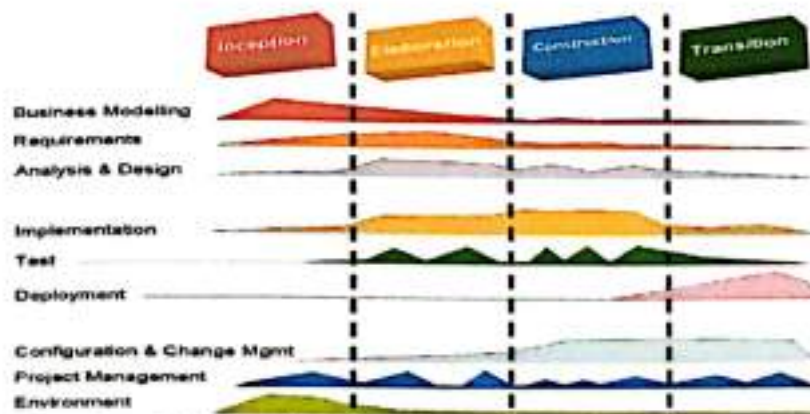
- Lihat apakah terdapat deretan karakter yang sama secara berurutan lebih dari tiga karakter, jika memenuhi lakukan pemampatan. Misal pada deretan karakter yang sama secara berurutan sebanyak 8 karakter, jadi lebih dari tiga dan dapat dilakukan pemampatan.
- Berikan bit benanda pada file pamampatan, bit penanda disini berupa 8 deretan bit yang boleh dipilih sembarang asalkan digunakan secara konsisten pada seluruh bit penanda pemampatan. Bit penanda ini berfungsi untuk menandai bahwa karakter selanjutnya adalah karakter pemampatan sehingga tidak membingungkan pada saat mengembalikan file yang sudah dimampatkan ke file aslinya. Misal bit penanda ini dipilih 11111110.

- c. Tambahkan deretan bit untuk menyatakan jumlah karakter yang sama berurutan, pada contoh diatas karakter yang sama berurutan sebanyak delapan kali, jadi diberikan deretan bit 00001000 (8 desimal).
- d. Tambahkan deretan bit yang menyatakan karakter yang berulang, misal pada contoh diatas karakter yang berulang adalah 01000001 atau karakter A pada karakter ASCII

1.8.3. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah dengan metode RUP. Menurut Rosa A.S (2013, Rekayasa Perangkat Lunak) Rational Unified Process (RUP) adalah pendekatan pengembangan perangkat lunak yang dilakukan berulang ulang (Iterative), fokus pada arsitektur, lebih diarahkan berdasarkan penggunaan kasus.

Metode Rational Unified Process (RUP) merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai best practices yang terdapat dalam industri pengembangan perangkat lunak. RUP menggunakan konsep object oriented, dengan aktifitas yang berfokus pada pengembangan model dengan menggunakan Unified Model Language (UML).



Gambar 1.2 Arsitektur Rational Unified Process. (Rossa, as. 2013)

Pada Gambar 1.1 menjelaskan bahwa RUP memiliki empat tahapan atau fase yang dapat dilakukan pula secara iteratif, yaitu tahap *Inception*, *Elaboration*, *Construction*, serta *Transition*.

Berikut 4 tahapan RUP dan penjelasannya :

1.7. Pertanyaan Penelitian

Adapun pertanyaan penelitian yang ditanyakan dalam penulisan skripsi ini adalah sebagai berikut :

1. Apakah aplikasi yang dibuat ini dapat digunakan Desa Cisantana untuk melayani masyarakat?
2. Apakah algoritma *Run Length Encoding (RLE)* dapat diimplementasikan pada Aplikasi Aduan dan Saran Masyarakat di Desa Cisantana?
3. Apakah algoritma *Run Length Encoding (RLE)* dapat melakukan kompresi pada citra image sehingga dapat menghemat penyimpanan pada server?

1.8. Metodologi Penelitian

Metodologi penelitian adalah proses atau cara ilmiah untuk mendapatkan data yang akan digunakan untuk keperluan penelitian. Adapun metodologi yang digunakan adalah sebagai berikut :

1.8.1. Metode Pengumpulan Data

1. Observasi

Metode pengumpulan data dengan melakukan pengamatan mengenai Desa Cisantana Kabupaten Kuningan guna mendapatkan informasi yang akurat.

2. Wawancara

Metode pengumpulan data dengan melakukan tanya jawab kepada bapak Ano Suratno, A.Md. selaku Kepala Desa Cisantana guna mendapatkan informasi seputar sistem aduan dan saran masyarakat yang masih memerlukan pengembangan.

3. Studi Pustaka

Metode pengumpulan data dengan mempelajari atau mencari informasi dari berbagai jurnal dan buku yang berhubungan dengan permasalahan yang ada. Sumber - sumber ini digunakan untuk melengkapi informasi yang diperlukan dalam mendukung pelaksanaan penelitian ini.

1.8.2. Metode Pemecahan Masalah

Algoritma *Run Length Encoding (RLE)*

Algoritma *Run Length Encoding (RLE)* ini merupakan salah satu

1. *Inception* (Permulaan)

Pada tahap ini lebih memodelkan proses bisnis yang di butuhkan (*business modelling*) dan mendefinisikan kebutuhan sistem yang akan dibuat (*requirements*). Dalam hal ini penulis melakukan pengumpulan data dengan cara melakukan observasi mengenai Desa Cisantana dengan melakukan wawancara terkait informasi mengenai bagaimana sistem Aduan dan Saran Masyarakat yang sedang berjalan tersebut. Data tersebut kemudian di analisis sebagai pemenuhan kebutuhan untuk melakukan perancangan aplikasi yang dibuat.

2. *Elaboration* (Perluasan atau Perencanaan)

Tahap untuk melakukan desain secara lengkap berdasarkan hasil analisis ditahap *inception*. Aktivitas yang dilakukan pada tahap ini antara lain mencakup pembuatan desain arsitektur subsistem (*architecture pattern*), desain komponen sistem, desain format data (protokol komunikasi), desain antarmuka/tampilan, desain peta aliran tampilan, penentuan *design pattern* yang digunakan, pemodelan diagram UML (*Unified Modelling Language*) yaitu *Use Case Diagram*, *Class Diagram*, *Activity Diagram* menggunakan aplikasi *Rational Rose*.

Pada tahap ini penulis melakukan perancangan *user interface*, *database*, dengan menggunakan sebuah dokumen yaitu spesifikasi kebutuhan perangkat lunak (SKPL).

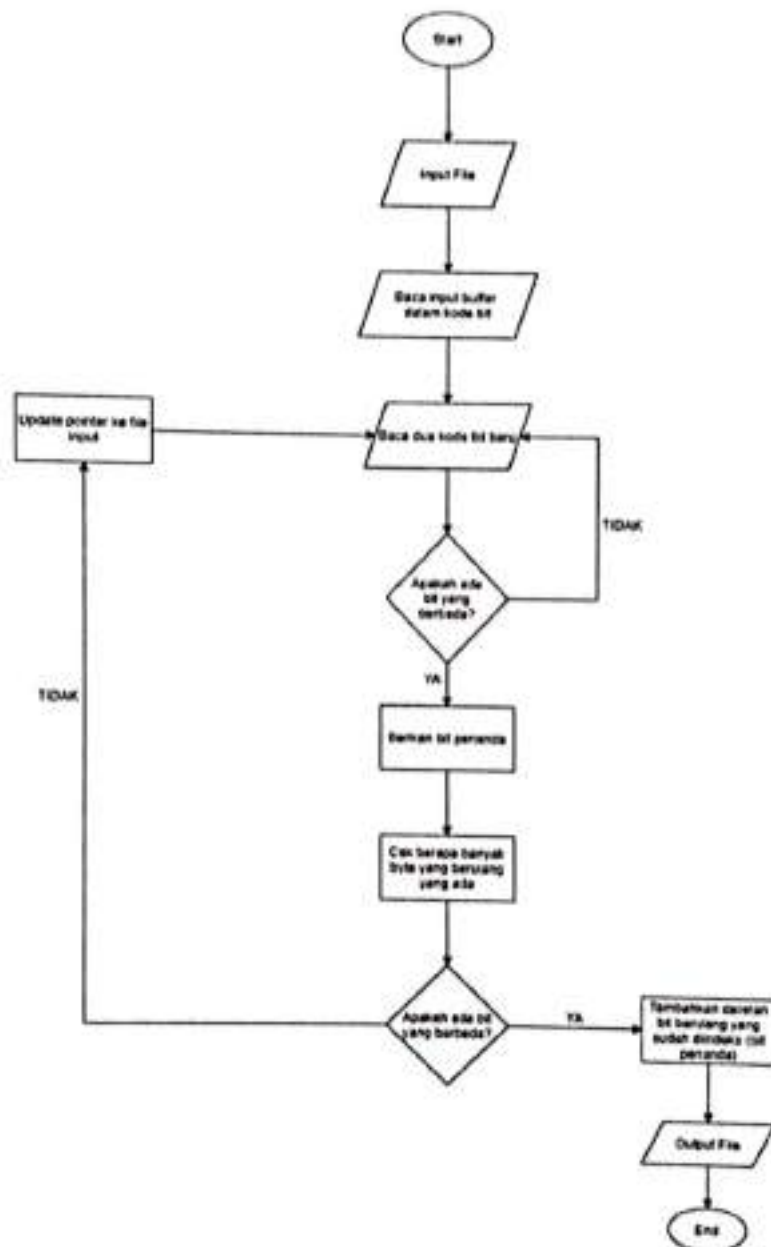
3. *Construction* (Konstruksi)

Tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem, implementasi kode program dan pengujian perangkat lunak. Pada tahap ini penulis mulai melakukan penulisan kode program (*coding*) dengan menggunakan bahasa pemrograman java untuk android (*client*), PHP untuk sisi web (*server*) dan MySQL untuk pembuatan *database*.

4. *Transition* (Transisi)

Pada tahap ini dilakukan pengujian, penyerahan dan pelatihan penggunaan aplikasi kepada *user* serta pemeliharaan penggunaannya. Pengujian pada aplikasi ini dilakukan dengan menggunakan *blackbox testing* dan *whitebox testing* untuk memastikan sistem berjalan dengan baik.

6. Dua nilai piksel selanjutnya akan dibaca dan dilakukan perulangan pada langkah 2 sampai dengan langkah 5, hingga semua piksel pada citra di periksa.
7. Piksel yang telah terbentuk akan menjadi citra hasil kompresi RLE.



Gambar 2.2 *Flowchart* Algoritma RLE
(Aditya Wijaya, 2012)

BAB III

ANALISIS DAN PERANCANGAN

3.1. Analisis Sistem

Konsep kegiatan analisis ini akan membahas mengenai algoritma yang akan digunakan pada aplikasi ini yaitu dengan menggunakan algoritma *Run Length Encoding* (RLE) untuk proses kompresi citra digital.

Analisis sistem bertujuan untuk melakukan identifikasi persoalan-persoalan yang muncul dalam pembuatan sistem. Hal ini dilakukan agar saat analisis dan perancangan sistem, mengimplementasi sistem yang sudah dirancang, menguji coba sistem yang telah dibuat. Evaluasi dan analisis hasil uji coba sistem proses perancangan aplikasi tidak terjadi kesalahan-kesalahan. Sehingga sistem dapat berjalan dengan baik dan selesai tepat pada waktu yang telah ditentukan.

3.1.1. Analisis Masalah

Sarana penyampaian aduan dan saran masyarakat Desa Cisantana pada sistem berbasis web yang sudah ada, terhadap ketidakpuasan dan permasalahan yang ada di lingkungan Desa Cisantana masih belum optimal, karena pada sistem yang sudah ada sekarang *server* sistem tersebut sering mengalami *lag* disebabkan dengan banyaknya *file* media pada sistem tersebut yang ukurannya terlalu besar sehingga dibutuhkannya kompresi untuk memperkecil ukuran *file* media tersebut. Bahkan, tidak sedikit masyarakat Desa Cisantana yang masih belum mengetahui adanya sarana penyampaian aduan dan saran masyarakat yang sudah dikembangkan melalui *website*, dan tidak sedikit pula masyarakat yang masih bingung harus ke mana untuk mengadukan keluhannya. Hal ini menyebabkan permasalahan yang ada hanya akan menjadi buah bibir di lingkungan Desa Cisantana dan tak kunjung diproses. Sebagai contoh, fasilitas disalah satu jalan rusak dan mengganggu pengguna jalan. Kebanyakan masyarakat, masih bingung ketika masyarakat sudah melaporkan permasalahan ini apakah desa sudah menganggapi laporan tersebut atau belum.

Smartphone adalah sebuah *mobile phone* yang dibangun dengan sistem operasi komputer, memiliki kemampuan kalkulasi canggih beserta konektivitas membuatnya lebih dari sekedar telepon. Sehingga terdapat banyak citra digital

yang terdapat dalam *smartphone*. Citra digital tersebut akan membutuhkan tempat penyimpanan yang sangat besar jika tidak dilakukan kompresi. Kompresi adalah proses mengubah data menjadi sekumpulan kode untuk menghemat tempat penyimpanan dengan satu atau tanpa mengurangi kualitas dari citra serta mempercepat waktu transmisi data.

Run Length Encoding (RLE) merupakan metode kompresi. Cara kerja kompresi dari metode RLE adalah dengan cara menjumlahkan pengulangan *byte*/karakter yang sama berturut-turut dan menampilkan hanya sebuah karakter yang mengalami pengulangan disertai dengan nilai jumlah pengulangan *byte*/karakter, sedangkan untuk *byte*/karakter yang tidak terjadi pengulangan *byte* maka karakter tersebut tidak akan dikompresi.

Permasalahan yang akan dibahas disini adalah bagaimana proses kompresi menggunakan metode RLE untuk meringankan kinerja server pada saat memproses file citra. Karena itulah penulis memakai media *smartphone android* untuk mengimplementasikan aplikasi kompresi ini menggunakan metode *Run Length Encoding* (RLE), karena user yang akan memakai aplikasi ini hampir semua memiliki *smartphone android*.

3.1.2. Analisis Sistem Yang Sedang Berjalan

Analisis aplikasi didefinisikan sebagai penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan dan hambatan yang terjadi pada kebutuhan yang mana diharapkan dapat diusulkan, Adapun sistem yang sedang berjalan sekarang untuk proses pelaporan aduan dan saran masyarakat kepada desa cisantana masih belum optimal :

3.1.5.2. Kebutuhan Perangkat Lunak

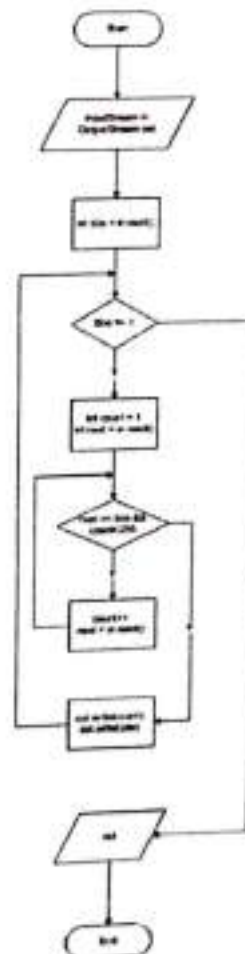
Analisis perangkat lunak bertujuan untuk mengetahui secara tepat perangkat lunak apa saja yang dibutuhkan untuk menjalankan suatu aplikasi. Adapun perangkat lunak yang dibutuhkan adalah :

Tabel 3.3 Perangkat lunak pembangun aplikasi adalah :

Sistem Operasi	Windows 10
Java Platform	Jdk 1.8
IDE	Android Studio

3.2. Analisis Algoritma *Run Length Encoding* (RLE)

Run Length Encoding (RLE) adalah bentuk paling mudah dari teknik kompresi data lossless dimana sederetan data dengan nilai yang sama secara berurutan akan disimpan menjadi sebuah nilai data dan jumlahnya. Metode ini digunakan untuk mengompresi citra yang memiliki kelompok-kelompok byte yang sama.



Gambar 3.3 Flowchart Kompresi Algoritma *Run Length Encoding*

Berikut adalah proses kompresi file citra digital dengan algoritma Run Length Encoding (RLE). Contoh, misalnya sebuah citra digital dengan ukuran 3.669.205 byte akan dilakukan kompresi dengan algoritma Run Length Encoding (RLE). Disini hanya akan dicontohkan kompresi sebanyak 100 byte pertama karena melakukan perhitungan manual untuk 3.669.205 byte tidak cukup di naskah.



Gambar 3.4 Citra Belum Terkompres

Tabel 3.4 Tabel *Byte* Citra Belum Terkompres Dalam *Hexadecimal*

7C	3F	00	00	80	00	00	00	00	03
FE	00	00	10	00	00	00	30	00	01
FF	FF	FD	F7	FF	FF	FF	FF	FF	FF
FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

Langkah 1 :

Cari byte-byte tetangga yang sama :

7C sebanyak 1 buah dalam hexadecimal 01
 3F sebanyak 1 buah dalam hexadecimal 01
 00 sebanyak 2 buah dalam hexadecimal 02
 80 sebanyak 1 buah dalam hexadecimal 01
 00 sebanyak 4 buah dalam hexadecimal 04
 03 sebanyak 1 buah dalam hexadecimal 01
 FE sebanyak 1 buah dalam hexadecimal 01
 00 sebanyak 2 buah dalam hexadecimal 02
 10 sebanyak 1 buah dalam hexadecimal 01
 00 sebanyak 3 buah dalam hexadecimal 03

30 sebanyak 1 buah dalam hexadecimal 01
 00 sebanyak 1 buah dalam hexadecimal 01
 01 sebanyak 1 buah dalam hexadecimal 01
 FF sebanyak 2 buah dalam hexadecimal 02
 FD sebanyak 1 buah dalam hexadecimal 01
 F7 sebanyak 1 buah dalam hexadecimal 01
 FF sebanyak 26 buah dalam hexadecimal 1A

Table 3.5 Citra Dengan *Byte* Yang Sama

7C	3F	00	00	80	00	00	00	00	03
FE	00	00	10	00	00	00	30	00	01
FF	FF	FD	F7	FF	FF	FF	FF	FF	FF
FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

Langkah 2 :

Lakukan kompresi byte-byte tetangga yang sama

Hasil Kompresi :

(7C,01) (3F,01) (00,02) (80,01) (00,04) (03,01) (FE,01) (00,02) (10,01)
 (00,03) (30,01) (00,01) (01,01) (FF,02) (FD,01) (F7,01) (FF,1A)

Setelah selesai melakukan kompresi menggunakan metode Run Length Encoding, Langkah selanjutnya melakukan pengkodean dari data tersebut.

Hasil Pengkodean :

7C 01 3F 01 00 02 80 01 00 04 03 01 FE 01 00 02 10 01 00 03 30 01 00 01
 01 01 FF 02 FD 01 F7 01 FF 1A

Semuanya = 34 byte

Berdasarkan hasil diatas maka diperoleh :

- Ukuran data sebelum dikompresi = 100 byte.
- Ukuran data setelah dikompresi = 34 byte.

Ukuran citra digital telah berkurang sebanyak 100 byte – 34 byte = 66 byte.

3.3. Perancangan Sistem

Analisis perancangan sistem menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem. Analisis yang dilakukan dimodelkan dengan menggunakan UML (*Unified Modeling Language*). Tahap-tahap pemodelan dalam analisis tersebut antara lain usecase diagram, activity diagram, class diagram, sequence diagram.

3.3.1. Use Case Diagram

Use case merupakan gambaran skenario dari interaksi antara user dengan sistem. Sebuah diagram use case menggambarkan hubungan antara actor dan kegiatan yang dapat dilakukannya terhadap aplikasi. Berikut adalah *use case* diagram dari sistem yang dibangun :



Gambar 3.5 Use Case Sistem

BAB III

ANALISIS DAN PERANCANGAN

3.1. Analisis Sistem

Konsep kegiatan analisis ini akan membahas mengenai algoritma yang akan digunakan pada aplikasi ini yaitu dengan menggunakan algoritma *Run Length Encoding* (RLE) untuk proses kompresi citra digital.

Analisis sistem bertujuan untuk melakukan identifikasi persoalan-persoalan yang muncul dalam pembuatan sistem. Hal ini dilakukan agar saat analisis dan perancangan sistem, mengimplementasi sistem yang sudah dirancang, menguji coba sistem yang telah dibuat. Evaluasi dan analisis hasil uji coba sistem proses perancangan aplikasi tidak terjadi kesalahan-kesalahan. Sehingga sistem dapat berjalan dengan baik dan selesai tepat pada waktu yang telah ditentukan.

3.1.1. Analisis Masalah

Sarana penyampaian aduan dan saran masyarakat Desa Cisantana pada sistem berbasis web yang sudah ada, terhadap ketidakpuasan dan permasalahan yang ada di lingkungan Desa Cisantana masih belum optimal, karena pada sistem yang sudah ada sekarang *server* sistem tersebut sering mengalami *lag* disebabkan dengan banyaknya *file* media pada sistem tersebut yang ukurannya terlalu besar sehingga dibutuhkannya kompresi untuk memperkecil ukuran *file* media tersebut. Bahkan, tidak sedikit masyarakat Desa Cisantana yang masih belum mengetahui adanya sarana penyampaian aduan dan saran masyarakat yang sudah dikembangkan melalui *website*, dan tidak sedikit pula masyarakat yang masih bingung harus ke mana untuk mengadukan keluhannya. Hal ini menyebabkan permasalahan yang ada hanya akan menjadi buah bibir di lingkungan Desa Cisantana dan tak kunjung diproses. Sebagai contoh, fasilitas disalah satu jalan rusak dan mengganggu pengguna jalan. Kebanyakan masyarakat, masih bingung ketika masyarakat sudah melaporkan permasalahan ini apakah desa sudah menganggapi laporan tersebut atau belum.

Smartphone adalah sebuah *mobile phone* yang dibangun dengan sistem operasi komputer, memiliki kemampuan kalkulasi canggih beserta konektivitas membuatnya lebih dari sekedar telepon. Sehingga terdapat banyak citra digital

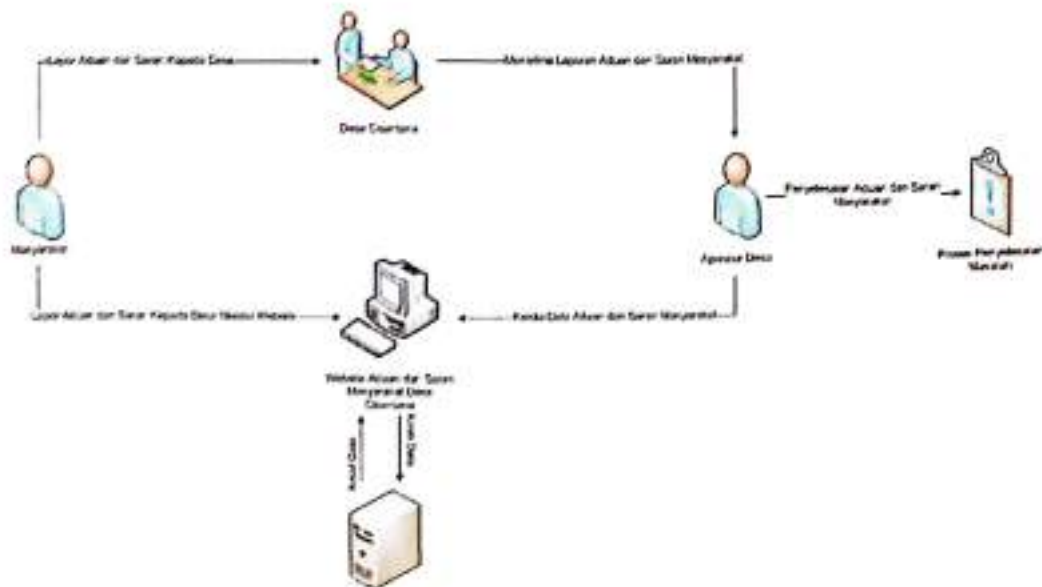
yang terdapat dalam *smartphone*. Citra digital tersebut akan membutuhkan tempat penyimpanan yang sangat besar jika tidak dilakukan kompresi. Kompresi adalah proses mengubah data menjadi sekumpulan kode untuk menghemat tempat penyimpanan dengan satu atau tanpa mengurangi kualitas dari citra serta mempercepat waktu transmisi data.

Run Length Encoding (RLE) merupakan metode kompresi. Cara kerja kompresi dari metode RLE adalah dengan cara menjumlahkan pengulangan *byte*/karakter yang sama berturut-turut dan menampilkan hanya sebuah karakter yang mengalami pengulangan disertai dengan nilai jumlah pengulangan *byte*/karakter, sedangkan untuk *byte*/karakter yang tidak terjadi pengulangan *byte* maka karakter tersebut tidak akan dikompresi.

Permasalahan yang akan dibahas disini adalah bagaimana proses kompresi menggunakan metode RLE untuk meringankan kinerja server pada saat memproses file citra. Karena itulah penulis memakai media *smartphone android* untuk mengimplementasikan aplikasi kompresi ini menggunakan metode *Run Length Encoding* (RLE), karena user yang akan memakai aplikasi ini hampir semua memiliki *smartphone android*.

3.1.2. Analisis Sistem Yang Sedang Berjalan

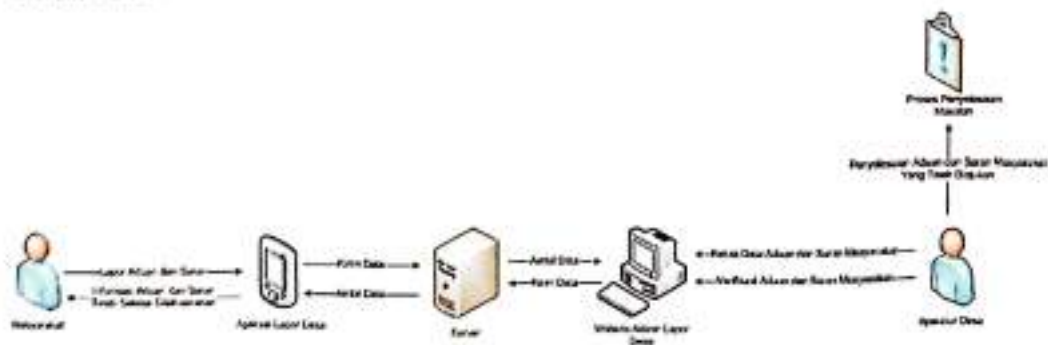
Analisis aplikasi didefinisikan sebagai penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan dan hambatan yang terjadi pada kebutuhan yang mana diharapkan dapat diusulkan. Adapun sistem yang sedang berjalan sekarang untuk proses pelaporan aduan dan saran masyarakat kepada desa cisantana masih belum optimal :



Gambar 3.1 Rich Picture Sistem Yang Sedang Berjalan

3.1.3. Analisis Sistem Yang Diusulkan

Sistem yang diusulkan akan mempermudah masyarakat dalam pelaporan aduan dan saran masyarakat kepada desa cisantana, selain itu prosesnya cepat dikarenakan pelaporan melalui aplikasi cukup cepat dan flexible. Dan bisa dilakukan dengan mudah menggunakan smartphone android, berikut sistem yang diusulkan dalam bentuk Rich Picture Diagram sistem yang akan diusulkan :



Gambar 3.2 Rich Picture Sistem Yang Diusulkan

3.1.4. Kebutuhan Fungsional

Kebutuhan fungsional berisi proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional juga berisi informasi-informasi apa saja yang harus mempunyai *functional requirements* sebagai berikut :

1. Sistem mampu melakukan kompresi terhadap citra digital yang dimasukan.

2. Sistem mampu melakukan dekompresi kembali citra digital yang telah dikompresi.
3. Sistem dapat membantu masyarakat dalam mengajukan aduan dan saran kepada desa.
4. Sistem dapat membantu desa dalam membuat keputusan dari hasil aduan dan saran yang diberikan masyarakat.
5. Terdapat petunjuk penggunaan aplikasi.

3.1.5. Kebutuhan Non Fungsional

Kebutuhan non fungsional menjabarkan apa saja yang harus dimiliki oleh sistem agar dapat berjalan. Tujuan dari fase ini adalah memahami dengan sebenar-benarnya kebutuhan dari sistem yang akan dibangun pada aplikasi kompresi citra. Kebutuhan non fungsional meliputi ketersediaan perangkat keras, perangkat lunak dan pengguna.

3.1.5.1. Kebutuhan Perangkat Keras

Perangkat keras ini adalah salah satu perangkat yang sangat penting, karena tanpa perangkat keras yang memenuhi syarat sebuah aplikasi yang dibangun tidak akan berjalan tanpa adanya perangkat keras. Perangkat keras yang digunakan dalam pembuatan aplikasi ini adalah dengan spesifikasi sebagai berikut :

Tabel 3.1 Spesifikasi untuk PC (Personal Computer) adalah :

Processor	Intel® Core™i3 2.0GHz
Memory	8 GB DDR4
Display	14"
Hardisk	1 TB HDD

Tabel 3.2 Spesifikasi untuk Smartphone adalah :

Sistem Operasi	Android 5.1 Lollipop
Display	4"
RAM	1 GB

3.1.5.2. Kebutuhan Perangkat Lunak

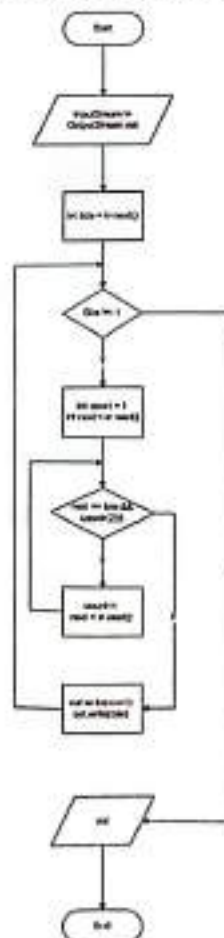
Analisis perangkat lunak bertujuan untuk mengetahui secara tepat perangkat lunak apa saja yang dibutuhkan untuk menjalankan suatu aplikasi. Adapun perangkat lunak yang dibutuhkan adalah :

Tabel 3.3 Perangkat lunak pembangun aplikasi adalah :

Sistem Operasi	Windows 10
Java Platform	Jdk 1.8
IDE	Android Studio

3.2. Analisis Algoritma *Run Length Encoding* (RLE)

Run Length Encoding (RLE) adalah bentuk paling mudah dari teknik kompresi data lossless dimana sederetan data dengan nilai yang sama secara berurutan akan disimpan menjadi sebuah nilai data dan jumlahnya. Metode ini digunakan untuk mengompresi citra yang memiliki kelompok-kelompok byte yang sama.

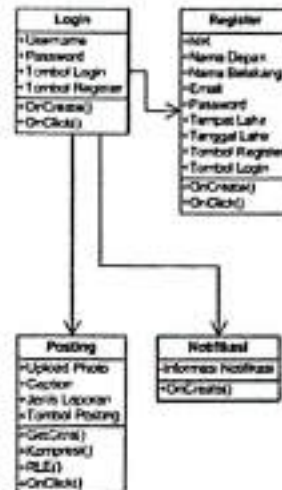


Gambar 3.3 Flowchart Kompresi Algoritma *Run Length Encoding*

3.3.4. Class Diagram

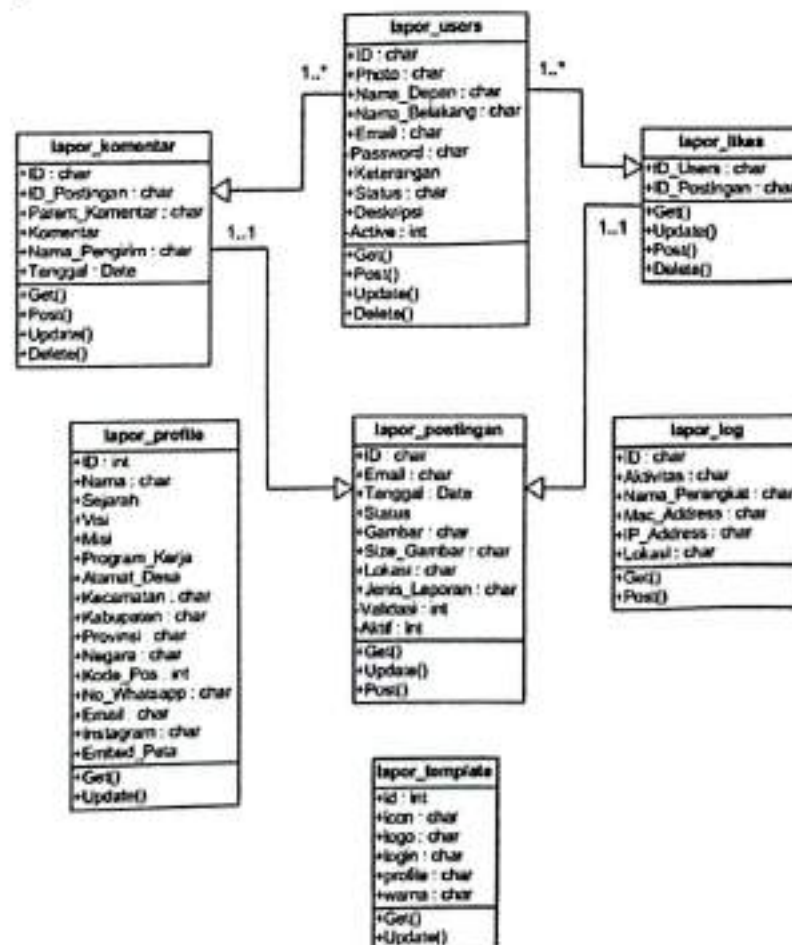
Class diagram menggambarkan keadaan atribut suatu sistem. Berikut adalah gambar class diagram untuk aplikasi aduan dan saran masyarakat (Lapor Desa) ini ditunjukkan pada gambar berikut :

1. Class Diagram User



Gambar 3.17 Class Diagram User

2. Class Diagram Admin



Gambar 3.18 Class Diagram Admin

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini akan dilakukan implementasi dan pengujian terhadap sistem. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman. Setelah diimplementasikan maka dilakukan pengujian terhadap sistem dan dilihat kekurangan-kekurangan pada aplikasi untuk pengembangan sistem selanjutnya.

4.1. Implementasi Sistem

Implementasi merupakan tahap sistem siap untuk dioperasikan. Implementasi disini penulis akan menjelaskansetiap form pada aplikasi aduan dan saran masyarakat (Lapor Desa) dengan menggunakan algoritma *Run Length Encoding* (RLE) berdasarkan desain dari aplikasi tersebut.

4.2. Perangkat Implementasi

4.2.1. Kebutuhan Perangkat Lunak

Dalam pembuatan aplikasi aduan dan saran masyarakat (Lapor Desa), penulis menggunakan beberapa perangkat lunak. Diantaranya :

Tabel 4.1 Kebutuhan perangkat lunak

Perangkat Lunak (Software)	
<i>Operating System</i> (OS)	Windows 10 64bit
Java Platform	Jdk 1.8
Text Editor	Android Studio
Perancangan Sistem	Microsoft Visio 2007

4.2.2. Kebutuhan Perangkat Keras

Dalam pembuatan aplikasi ini, penulis menggunakan spesifikasi perangkat keras sebagai berikut :

Tabel 4.2 Kebutuhan perangkat keras untuk Laptop

Perangkat Keras (Hardware)	
Processor	Intel ® Core™ i3
Memory	8 GB DDR4
Display	14"
Hardisk	1TB HDD



Gambar 4.16 Tampilan Tema



4.4. Pengujian Perangkat Lunak

4.4.1. Pengujian Kompresi

Hasil yang didapatkan melalui penelitian ini berupa hasil dari proses kompresi system. Pengujian dilakukan terhadap 5 buah citra yang berbeda. Adapun hasil uji coba, dapat dilihat dibawah ini :

- $Compression Ratio = \frac{ukuran\ data\ setelah\ dikompresi}{ukuran\ data\ sebelum\ dikompresi}$
- $Space Saving = \left(1 - \frac{ukuran\ data\ setelah\ dikompresi}{ukuran\ data\ sebelum\ dikompresi}\right) \times 100\%$

1. Gambar ke-1

Sebelum Kompresi	Setelah Kompresi
	
Ukuran 4,80 Megabyte	Ukuran 1,68 Megabyte

- $Compression Ratio = \frac{1,68}{4,80} \times 100\%$
 $= 0,35 \times 100\%$
 $= 35\%$

$$\begin{aligned}
 - \text{Space Saving} &= 1 - \frac{1.68}{4.80} \times 100\% \\
 &= (1 - 0,35) \times 100\% \\
 &= 0,65 \times 100\% \\
 &= 65\%
 \end{aligned}$$

2. Gambar ke-2



$$\begin{aligned}
 - \text{Compression Ratio} &= \frac{1.97}{4.81} \times 100\% \\
 &= 0,45 \times 100\% \\
 &= 45\%
 \end{aligned}$$

$$\begin{aligned}
 - \text{Space Saving} &= 1 - \frac{1.97}{4.81} \times 100\% \\
 &= (1 - 0,45) \times 100\% \\
 &= 0,55 \times 100\% \\
 &= 55\%
 \end{aligned}$$

4.4.2. Black Box Testing

Pengujian *black box* ini digunakan untuk menguji fungsi-fungsi khusus dari sebuah perangkat lunak yang dirancang, kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut.

Adapun hasil dari pengujian *black box* yang telah penulis lakukan adalah pada tabel 4.4 sebagai berikut :

Tabel 4.4 Pengujian *Black Box*

No	Fungsi yang diuji	Skenario	Hasil yang diharapkan	Hasil Uji
1	Sistem Operasi	Menjalankan Aplikasi Laporan Desa pada Sistem Operasi Android Versi 5 (Lollipop)	Aplikasi berjalan dengan baik.	Valid
2	Tampilan Splash Screen	Membuka aplikasi dan halaman splash screen	Muncul tampilan splash screen	Valid
3	Tampilan Login	Masuk ke halaman login	Menampilkan halaman login, jika session login masih ada maka akan dialihkan ke halaman home	Valid
4	Tampilan Register	Mengklik tombol register	Menampilkan halaman register yang dapat memilih photo profile	Valid

4.4.3. White Box Testing

Pengujian *white box* adalah sebuah pengujian yang dilakukan dengan melihat kedalam modul dan kode-kode yang ada di dalam aplikasi. Tujuannya adalah sebagai petunjuk untuk mendapatkan program yang benar secara menyeluruh sehingga aplikasi aduan dan saran masyarakat (Lapor Desa) dengan algoritma RLE yang dirancang mampu menghasilkan *interface* dan output yang sesuai dengan kebutuhan dan harapan. Secara sekilas dapat diambil kesimpulan pengujian *white box* merupakan petunjuk untuk mendapatkan program yang benar.

Proses kompresi dengan algoritma *RLE* akan ditunjukkan oleh tabel 4.6 berikut.

Tabel 4.5 Pengujian White Box RLE

Node	Kode Program
1	<code>public void compress(InputStream in, OutputStream out){</code>
2	<code>try{</code>
3	<code>int bite = in.read();</code>
4	<code>while(bite != 1){</code>
5	<code>int count = 1;</code> <code>int next = in.read();</code>
6	<code>while(next == bite && count < 255){</code>
7	<code>count++;</code> <code>next = in.read();</code>
8	<code>}</code>
9	<code>out.write(count);</code> <code>out.write(bite);</code> <code>bite = next;</code>
10	<code>}</code>
11	<code>} catch (IOException e) {</code>
12	<code>e.printStackTrace();</code>
13	<code>}</code>
14	<code>}</code>

Lapor Desa
ini sudah
cukup baik?

a. Analisis pertanyaan pertama

Dari tabel diatas dapat disimpulkan bahwa untuk pertanyaan No.1 berdasarkan hasil dari jawaban 15 responden mendapatkan hasil 45% yang berarti tampilan pada aplikasi Lapor Desa ini cukup menarik.

b. Analisis pertanyaan kedua

Dari tabel diatas dapat disimpulkan bahwa untuk pertanyaan No.2 berdasarkan hasil dari jawaban 15 responden mendapatkan hasil 89% yang berarti menu yang terdapat pada aplikasi Lapor Desa ini sangat mudah dipahami.

c. Analisis pertanyaan ketiga

Dari tabel diatas dapat disimpulkan bahwa untuk pertanyaan No.3 berdasarkan hasil dari jawaban 15 responden mendapatkan hasil 100% yang berarti dengan adanya aplikasi Lapor Desa ini sangat dapat membantu masyarakat untuk mengajukan aduan dan saran kepada desa.

d. Analisis pertanyaan keempat

Dari tabel diatas dapat disimpulkan bahwa untuk pertanyaan No.4 berdasarkan hasil dari jawaban 15 responden mendapatkan hasil 83% yang berarti aplikasi Lapor Desa ini sudah berjalan dengan baik.

e. Analisis pertanyaan kelima

Dari tabel diatas dapat disimpulkan bahwa untuk pertanyaan No.5 berdasarkan hasil dari jawaban 15 responden mendapatkan hasil 77% yang berarti aplikasi Lapor Desa sudah cukup baik.