



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Android

Android merupakan sistem operasi untuk perangkat *mobile* berbasis Linux. Pada dasarnya, sistem pada android berisikan tumpukan dari beberapa lapisan agar perangkat keras dapat berinteraksi dengan *user* (Fransiscus, 2014).

2.1.1 Fitur-fitur Android

Konfigurasi pada perangkat android dapat berbeda antara satu perangkat dengan perangkat lain karena android terbuka bagi manufaktur untuk memodifikasi sesuai kebutuhan mereka. Namun, android sendiri mendukung fitur-fitur tertentu (Ragil, 2013) sebagai berikut:

1. Penyimpanan menggunakan SQLite. SQLite adalah *database relational* yang ringan untuk menyimpan data.
2. Koneksi mendukung GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth (termasuk A2DP dan AVRVP), WiFi, LTE dan WiMAX.
3. Pesan mendukung SMS dan MMS.
4. *Web browser* menggunakan *open source* WebKit termasuk *engine* Chrome v8 Javascript di dalamnya.
5. Media yang didukung antara lain: H.263, H.264 (3GP atau MP4), MPEG-4 SP, AMR, AMR-WB (3GP container), ACC, HE-ACC (MP4 atau 3GP), MP3, MIDI, Ogg Vorbis, WAV. JPEG, PNG, GIF dan BMP.

6. *Framework* aplikasi yang mendukung penggantian komponen dan *reusable*.
7. *Hardware* terdapat kamera, kompas digital, *accelerometer sensor*, *proximity sensor* dan GPS.
8. *Multi-touch*.
9. *Multi-tasking*.
10. Dukungan *flash*.

2.2 IDE Eclipse

Eclipse adalah sebuah *Integrated Development Environment (IDE)* untuk pengembangan perangkat lunak dan dapat dijalankan pada semua *platform*. Sifat-sifat Eclipse yaitu *multi-platform*, *multi-language*, dan *multi-role*. Eclipse bersifat *open-source* dan dapat diunduh secara gratis. (Lindung, 2012).

2.3 Android SDK

Android SDK merupakan *tools Application Programming Interface (API)* yang dibutuhkan dalam pengembangan aplikasi pada *platform* android menggunakan bahasa pemrograman Java. (Siti dkk., 2014).

2.4 Google Maps Android API

Website resmi developer android menjelaskan bahwa Google Maps Android API adalah peta yang memungkinkan pengguna untuk menjelajahi dunia. Beberapa fitur dari API ini yaitu mengidentifikasi lokasi dengan *custom marker*, menambah data peta dengan *image overlay*, menanamkan satu atau lebih peta

sebagai *fragmen*, dan banyak lagi. Versi dari Google Maps Android API yang terakhir yaitu versi 2 atau Google Maps Android API v2.

2.4.1 Google Maps Directions API

Website resmi *developer* Google menjelaskan bahwa Google Directions API adalah layanan yang dapat menghitung arah antar lokasi menggunakan sebuah HTTP *request*. Beberapa model yang dapat dicari menggunakan API ini yaitu transportasi, termasuk *transit*, *driving*, *walking* atau *cycling*. *Directions* yang dapat diterima memungkinkan untuk informasi yang lebih spesifik seperti asal, tujuan dan *waypoints* baik sebagai *string* teks atau koordinat lintang dan bujur.

2.5 Kota Tangerang

Website resmi Kota Tangerang menjelaskan bahwa Kota Tangerang merupakan sebuah kota yang terletak di Provinsi Banten. Sebelah utara berbatasan dengan Kecamatan Teluk Naga dan Kecamatan Sepatan Kabupaten Tangerang, sebelah selatan berbatasan dengan Kecamatan Curug, Kecamatan Serpong dengan DKI Jakarta, sedangkan sebelah Barat berbatasan dengan Kecamatan Cikupa Kabupaten Tangerang.

Letak Kota Tangerang tersebut sangat strategis karena berada di antara Ibukota Negara DKI Jakarta dan Kabupaten Tangerang. Sesuai dengan Instruksi Presiden Nomor 13 Tahun 1976 tentang Pengembangan Jabotabek (Jakarta, Bogor, Tangerang, Bekasi), Kota Tangerang merupakan salah satu daerah penyangga Ibukota Negara DKI Jakarta.

Posisi Kota Tangerang tersebut menjadikan pertumbuhannya pesat. Pada satu sisi wilayah Kota Tangerang menjadi daerah limpahan berbagai kegiatan di Ibukota Negara DKI Jakarta. Di sisi lain Kota Tangerang dapat menjadi daerah kolektor pengembangan wilayah Kabupaten Tangerang sebagai daerah dengan sumber daya alam yang produktif.

2.5.1 Pejabat Pengelola Informasi dan Dokumentasi

Informasi merupakan kebutuhan utama setiap orang. Hak memperoleh informasi adalah salah satu dari hak asasi manusia. Hal ini tercantum dalam Undang-Undang Dasar Negara Republik Indonesia Tahun 1945 Pasal 28 F. Dalam pasal tersebut disebutkan bahwa setiap orang berhak untuk berkomunikasi dan memperoleh informasi untuk mengembangkan pribadi dan lingkungan sosialnya, serta berhak untuk mencari, memperoleh, memiliki, dan menyimpan informasi dengan menggunakan segala jenis saluran yang tersedia.

Website resmi Kota Tangerang menjelaskan bahwa Indonesia menerapkan dan menjunjung tinggi prinsip keterbukaan, terutama dalam penyelenggaraan negara. Salah satu elemen penting dalam mewujudkan penyelenggaraan negara yang terbuka adalah hak publik untuk memperoleh informasi. Hak atas informasi menjadi sangat penting karena makin terbuka penyelenggaraan negara untuk diawasi publik, penyelenggaraan negara tersebut makin dapat dipertanggungjawabkan. Hak setiap orang untuk memperoleh Informasi juga untuk meningkatkan kualitas pelibatan masyarakat dalam proses pengambilan keputusan publik. Partisipasi masyarakat tidak banyak berarti tanpa jaminan keterbukaan informasi publik.

Setiap badan publik mempunyai kewajiban untuk membuka akses atas informasi publik untuk masyarakat luas. Lebih detilnya, menurut laman *web* Kota Tangerang menjelaskan bahwa informasi publik adalah informasi yang dihasilkan, disimpan, dikelola, dikirim, dan/atau diterima oleh suatu badan publik yang berkaitan dengan penyelenggara dan penyelenggaraan negara dan/atau penyelenggara dan penyelenggaraan badan publik serta informasi lain yang berkaitan dengan kepentingan publik.

Sesuai dengan amanat pasal 13 UU No.14 Tahun 2008, Pemerintah Kota Tangerang sebagai salah satu badan publik telah membentuk Pejabat Pengelola Informasi dan Dokumentasi (PPID) melalui Keputusan Walikota Nomor 800/Kep.284-Infokom/2011. Dengan terbentuknya PPID pemohon informasi sesuai dengan haknya dapat memperoleh informasi publik yang dihasilkan oleh Dinas Informasi dan Komunikasi Pemerintah Kota Tangerang dengan ketentuan dalam UU No. 14 Tahun 2008.

Berdasarkan Keputusan Walikota Nomor 800/Kep.284-Infokom/2011 tentang Penunjukan PPID di Lingkungan Pemerintah Kota Tangerang, berikut adalah Pejabat PPID di tiap SKPD/BUMD di Lingkungan Pemerintah Kota Tangerang:

1. Sekretariat Daerah
2. Sekretariat DPRD
3. Inspektorat
4. Dinas Pemadam Kebakaran
5. Dinas Pendidikan
6. Dinas Pekerjaan Umum

7. Dinas Tata Kota
8. Dinas Kesehatan
9. Dinas Sosial
10. Dinas Kependudukan dan Pencatatan Sipil
11. Dinas Perhubungan
12. Dinas Ketenagakerjaan
13. Dinas Perindustrian Perdagangan dan Koperasi
14. Dinas Pemuda, Olahraga, Budaya dan Pariwisata
15. Dinas Pertanian
16. Dinas Kebersihan dan Pertamanan
17. Dinas Informasi dan Komunikasi
18. Dinas Pengelola Keuangan dan Aset Daerah
19. Badan Perencanaan Daerah
20. Badan Kepegawaian Pendidikan dan Pelatihan
21. Badan Pengelola Lingkungan Hidup
22. Badan Pemberdayaan Masyarakat dan Keluarga Berencana
23. Badan Pelayanan dan Perijinan Terpadu
24. Satuan Polisi Pamong Praja
25. Kantor Arsip
26. Kantor Penelitian Pengembangan dan Statistik
27. Kantor Perpustakaan
28. Kantor Kesatuan Bangsa dan Perlindungan Masyarakat
29. Perusahaan Daerah Air Minum Tirta Benteng
30. Perusahaan Daerah PASAR

31. Kecamatan Tangerang
32. Kecamatan Karawaci
33. Kecamatan Jatiuwung
34. Kecamatan Periuk
35. Kecamatan Neglasari
36. Kecamatan Batuceper
37. Kecamatan Benda
38. Kecamatan Ciledug
39. Kecamatan Larangan
40. Kecamatan Karang Tengah
41. Kecamatan Pinang
42. Kecamatan Cibodas
43. Kecamatan Cipondoh

2.6 Angkutan

Berdasarkan Peraturan Pemerintah No. 41 tahun 1993 tentang Angkutan Jalan, angkutan adalah pemindahan orang dan atau barang dari satu tempat ke tempat lain dengan menggunakan kendaraan. Sedangkan kendaraan umum adalah setiap kendaraan bermotor yang disediakan untuk dipergunakan oleh umum dengan dipungut bayaran. Kendaraan umum untuk penumpang dapat menggunakan mobil penumpang yang dilengkapi sebanyak-banyaknya 8 (delapan) tempat duduk tidak termasuk tempat duduk pengemudi, baik dengan maupun tanpa perlengkapan pengangkutan bagasi, dan mobil bus yang dilengkapi lebih dari 8 (delapan) tempat duduk tidak termasuk tempat duduk pengemudi,

baik dengan maupun tanpa perlengkapan pengangkutan bagasi. Pelayanan angkutan diatur dalam trayek tetap dan teratur yang dilakukan dalam jaringan trayek dengan jadwal tetap atau tidak berjadwal.

Berdasarkan Peraturan Walikota Nomor 55 tahun 2014 tentang Tarif Angkutan Penumpang Umum Dalam Wilayah Kota Tangerang terdapat 21 kode trayek angkutan penumpang umum pada wilayah Kota Tangerang. Kode-kode trayek tersebut dapat dilihat pada Gambar 2.1. berikut:

NO	KODE TRAYEK	LINTASAN TRAYEK
1.	T. 01	TERMINAL PORIS PLAWAD - TERMINAL CIMONE - JATAKE
2.	T. 02	TERMINAL PORIS PLAWAD - TERMINAL CIMONE - PERUM I
3.	T. 03	TERMINAL PORIS PLAWAD - PASAR BARU - CADAS
4.	T. 04	TERMINAL PORIS PLAWAD - SEWAN - SELAPAJANG PP
5.	T. 05	TERMINAL PORIS PLAWAD - KEBON BESAR - DUTAGARDEN
6.	T. 06	TERMINAL PORIS PLAWAD - KEBON BESAR - PAJANG
7.	T. 07	TERMINAL PORIS PLAWAD - CIKOKOL - PASAR ANYAR
8.	T. 08	GEMBOR - CIKOKOL - TERMINAL PORIS PLAWAD
9.	T. 09	BATU CEPER - TERMINAL PORIS PLAWAD - STADION BENTENG
10.	T. 10	TERMINAL PORIS PLAWAD - AMPERA - CIPONDOH - JL. MARGA
11.	T. 11	TERMINAL PORIS PLAWAD - AMPERA - PONDOK BAHAR
12.	T. 12	CILEDUG - CIKOKOL - TERMINAL PORIS PLAWAD
13.	T. 14	CILEDUG - CIPETE - CIKOKOL - TERMINAL PORIS PLAWAD
14.	T. 15	CILEDUG - PASAR BENGKOK - KUNCIRAN
15.	T. 16	CILEDUG - PUSDIKLANTAS - CIKOKOL - TERMINAL PORIS PLAWAD
16.	AK.02	PINTU M1 BANDARA - TERMINAL 2 BANDARA - RAWA BOKOR - TERMINAL 1 BANDARA - PINTU M1 BANDARA
17.	A. 03A	CIMONE - HARAPAN KITA - CURUG
18.	R. 06B	PERUM KELAPA DUA - TERMINAL PORIS PLAWAD
19.	R. 11/11A	PERUMNAS III,II,I - CIKOKOL - TERMINAL PORIS PLAWAD
20.	R. 14	PERUMNAS II - CIMONE - CIKOKOL - TERMINAL PORIS PLAWAD
21.	R. 15A	CIMONE - REGENCY - KOTABUMI

Gambar 2.1. Trayek Angkutan Penumpang Umum di Wilayah Kota Tangerang (Sumber: Peraturan Walikota Nomor 55 Tahun 2014)

2.7 Algoritma Boyer-Moore

Algoritma Boyer-Moore merupakan salah satu algoritma pencarian *string* yang paling efisien dalam aplikasi sehari-hari, dikembangkan oleh Robert S.

Boyer dan J. Strother Moore pada tahun 1977 (Rama, 2008). Ide utama dari algoritma ini yaitu dengan melakukan pencocokkan karakter mulai dari kanan, dan bukan dari kiri, sehingga akan lebih banyak informasi yang didapat (Vina, 2013).

Algoritma Boyer-Moore memiliki kelebihan dibanding algoritma pencarian *string* lainnya seperti *Brute Force*, *Kunth-Morris-Pratt* yang membandingkan satu-persatu karakter dari kiri ke kanan, yaitu dengan memeriksa sedikit karakter kemudian diketahui bahwa *string* yang dicari tidak ditemukan dan dapat digeser ke posisi berikutnya sehingga memiliki loncatan karakter yang besar yang mempercepat pencarian *string*. Namun, ketika semua karakter memiliki kesamaan tetapi karakter paling kiri berbeda, pencarian ini memerlukan waktu sedikit lebih lama (Mego, 2015).

2.7.1 Prinsip Dasar

Algoritma Boyer-Moore mempunyai empat konsep dasar dalam proses pencarian, yaitu *Preprocessing*, *Right-to-left-scan*, *Bad-character-rule*, dan *Good-suffix-rule*. *Precomputation* dari Boyer-Moore terdiri dari *bad character preprocessing* dan *good-suffix preprocessing*. Prinsip utama dari algoritma Boyer-Moore adalah melakukan perbandingan antara *pattern* dengan teks. Perbandingan dimulai dengan membandingkan antara karakter paling kanan dari *pattern* dengan teks. Ketika terjadi kecocokkan, perbandingan dilanjutkan dengan karakter disebelah kiri dari yang dibandingkan sampai ke karakter pertama dari *pattern*. Ketika terjadi ketidakcocokkan, dilakukan pergeseran yang ditentukan oleh 2 fungsi pergeseran, yaitu *bad character shift* dan *good suffix shift*. Aturan dari *bad*

character shift digunakan untuk menghindari pengulangan perbandingan yang gagal dari suatu karakter teks dengan *pattern*. Sedangkan aturan dari *good suffix shift* digunakan untuk menangani kasus yang di dalamnya terdapat pengulangan karakter pada *pattern* (Rama, 2008).

2.7.2 Bad Character Shift

Jika karakter teks yang sedang dibandingkan tidak cocok muncul pada posisi manapun di *pattern*, maka *pattern* digeser sehingga sejajar dengan simbol teks. Pada Tabel 2.1. berikut merupakan contoh yang menggambarkan situasi ini.

Tabel 2.1. Kasus ketidakcocokkan perbandingan karakter
(Sumber: <http://www.inf.fh-flensburg.de/lang/algorithmen/pattern/bmen.htm>)

indeks	0	1	2	3	4	5	6	7	8	9	...
teks	a	b	b	a	b	a	b	a	c	b	a
pattern	b	a	b	a	c						
			b	a	b	a	c				

Pada Tabel 2.1. perbandingan antara karakter “c” dengan karakter “b” menghasilkan ketidakcocokkan. Karakter “b” muncul pada *pattern* di posisi 0 dan 2. Selanjutnya *pattern* dapat digeser sehingga karakter “b” paling kanan sejajar dengan karakter “b” pada teks.

Dalam menghitung tabel *Bad Character Shift* (Kencana, 2014) digunakan langkah-langkah sebagai berikut:

Contoh *string*: manaman

Panjang: 7 karakter

1. Lakukan pencacahan dari posisi paling kanan string ke posisi awal dimulai dengan nilai 0 karena karena terletak pada posisi terakhir (karakter “n”).
2. Mundur ke posisi selanjutnya, nilai pencacah ditambah 1, jika karakter di posisi ini belum pernah ditemukan, maka nilai pergeserannya adalah sama dengan nilai pencacah (karakter “a”).
3. Mundur ke posisi selanjutnya. Karakter “m” nilai pergeserannya 2.
4. Mundur ke posisi selanjutnya. Karakter “a” sudah pernah ditemukan sehingga nilai pergeserannya sama dengan nilai pergeseran karakter “a” yang sudah ditemukan yaitu 1.
5. Begitu seterusnya sampai posisi awal dari *string*.

Tabel 2.2. Tabel Bad Character Shift
(Sumber: Kencana, 2014)

Posisi	1	2	3	4	5	6	7
String	M	A	N	A	M	A	N
Pergeseran	2	1	0	1	2	1	0

Untuk karakter selain “m, a, n”, nilai pergeseran sebesar panjang string, yaitu tujuh karakter.

2.7.3 Good Suffix Shift

Aturan *bad character shift* dapat mengalami kesalahan ketika terjadi situasi ketidakcocokkan, kemudian karakter teks yang tidak cocok yang muncul pada *pattern* menghasilkan nilai pergeseran negatif. Pada tabel 2.3. suffix “ab” telah cocok. Kemudian *pattern* dapat digeser sampai kemunculan selanjutnya dari “ab” sejajar dengan teks, dalam kasus ini pada posisi 2.

Tabel 2.3. Kasus pergeseran sesuai kemunculan *good suffix*
 (Sumber: <http://www.inf.fh-flensburg.de/lang/algorithmen/pattern/bmen.htm>)

indeks	0	1	2	3	4	5	6	7	8	9	...
teks	a	b	a	a	b	a	b	a	c	b	a
pattern	c	a	b	a	b						
			c	a	b	a	b				

Pada kasus selanjutnya seperti pada Tabel 2.4. *suffix* “ab” telah cocok, tetapi tidak ada kemunculan “ab” dalam *pattern*. Oleh karena itu, *pattern* dapat digeser setelah “ab” pada teks.

Tabel 2.4. Kasus pergeseran tidak ada kemunculan *suffix*
 (Sumber: <http://www.inf.fh-flensburg.de/lang/algorithmen/pattern/bmen.htm>)

indeks	0	1	2	3	4	5	6	7	8	9	...
teks	a	b	a	a	b	a	b	a	c	b	a
pattern	c	b	a	a	b						
						c	b	a	a	b	

Pada kasus selanjutnya seperti pada Tabel 2.5. diketahui *suffix* “bab” telah cocok. Kemudian tidak ditemukan “bab” lain dalam *pattern*. Tetapi dalam kasus ini, *pattern* tidak dapat digeser ke posisi 5 seperti sebelumnya, tetapi hanya ke posisi 3 karena sebuah *prefix* dari *pattern* yaitu “ab” cocok dengan akhir dari “bab”.

Tabel 2.5. Kasus pergeseran sesuai *prefix* dalam *suffix*
 (Sumber: <http://www.inf.fh-flensburg.de/lang/algorithmen/pattern/bmen.htm>)

indeks	0	1	2	3	4	5	6	7	8	9	...
teks	a	a	b	a	b	a	b	a	c	b	a
pattern	a	b	b	a	b						
				a	b	b	a	b			

Dalam menghitung tabel *Good Suffix Shift* (Kencana, 2014) digunakan langkah-langkah sebagai berikut (Contoh *string*: manaman):

1. Jika karakter pada posisi ke 7 bukan karakter “n”, maka pindah 1 posisi, berlaku untuk semua *pattern* yang dicari.
2. Jika karakter “n” sudah cocok tetapi karakter sebelum “n” bukan “a”, maka pindah sejumlah 7 posisi sehingga *pattern* melewati teks karena sudah pasti “manambn” bukan “manaman”.
3. Jika karakter “an” sudah cocok tetapi karakter sebelum “an” bukan “m”, maka pindah sejumlah 7 posisi sehingga *pattern* melewati teks karena sudah pasti “manaban” bukan “manaman”.
4. Jika karakter “man” sudah cocok tetapi karakter sebelum “man” bukan “a”, maka pindah sejumlah 4 posisi sehingga posisi *pattern* berada atau bersesuaian dengan akhiran “man” yang sudah ditemukan sebelumnya, karena bisa saja akhiran “man” merupakan awalan dari *pattern* “manaman” yang berikutnya.
5. Jika karakter “aman” sudah cocok tetapi karakter sebelum “aman” bukan “n”, maka pindah sejumlah 4 posisi sehingga posisi *pattern* berada/bersesuaian dengan akhiran “man” yang sudah ditemukan sebelumnya karena bisa saja akhiran “man” merupakan awalan dari *pattern* “manaman” yang berikutnya.

Selanjutnya sama, pergeseran paling memungkinkan dalam tabel adalah sejumlah 4 posisi.

Tabel 2.6. Tabel Good Suffix Shift
(Sumber: Kencana, 2014)

Posisi	1	2	3	4	5	6	7
String	M	A	N	A	M	A	N
Pergeseran	4	4	4	4	7	7	1

2.8 Skala Likert

Pada tahun 1932 Rensis Likert dalam laporannya menjelaskan penggunaan skala yang pada akhirnya diberi nama sesuai namanya. Ramadhan (2011) mengemukakan bahwa skala Likert adalah suatu skala psikometrik yang umum digunakan pada kuesioner maupun riset berupa survei dengan menggunakan beberapa pilihan skala, biasanya lima pilihan skala: Sangat Setuju, Setuju, Netral, Tidak Setuju, dan Sangat Tidak Setuju, dengan masing-masing bobot untuk setiap tingkatnya. Setiap bobot digunakan untuk proses analisa dan penarikan kesimpulan. Rumus dari skala likert adalah sebagai berikut:

$$Total\ Skor = \frac{Total\ Responden}{Bobot\ Skala} \quad \text{rumus (1)}$$

$$Interval = \frac{100}{Jumlah\ Tingkatan\ Skala} \quad \text{rumus (2)}$$

$$Indeks\ Skor = \frac{Total\ Skor}{Skor\ Bobot\ Tertinggi \times Total\ Responden} \times 100 \quad \text{rumus (3)}$$