

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Aplikasi

Menurut Hasugian (2014) aplikasi berasal dari kata *application* yaitu bentuk benda dari kata kerja *to apply* yang dalam bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu subkelas perangkat lunak komputer yang menggunakan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pemakai. Aplikasi menurut Sanjaya (2015) adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu. Contoh utama perangkat lunak aplikasi adalah program pengolah kata, lembar kerja, dan pemutar media. Kumpulan aplikasi komputer yang digabung menjadi suatu paket biasanya disebut paket atau suite aplikasi (*application suite*). Contohnya adalah Microsoft Office dan OpenOffice.org, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Umumnya aplikasi-aplikasi tersebut memiliki kemampuan untuk saling berinteraksi sehingga menguntungkan pemakai. Contohnya, suatu lembar kerja dapat dimasukkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah.

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan.

2.2 Algoritma Brute Force

Brute force adalah sebuah pendekatan yang langsung (straightforward) untuk memecahkan suatu masalah, biasanya didasarkan pada pernyataan masalah

(problem statement) dan definisi konsep yang dilibatkan. Algoritma brute force memecahkan masalah dengan sangat sederhana, langsung dan dengan cara yang jelas (obvious way).

a. Kelebihan Algoritma Brute Force

Berikut ini beberapa kelebihan yang dimiliki oleh brute force, yaitu:

1. Algoritma brute force dapat digunakan untuk memecahkan hampir sebagian besar masalah.
2. Sederhana dan mudah dimengerti.
3. Menghasilkan algoritma yang layak untuk beberapa masalah penting seperti pencarian, pengurutan, pencocokan string, perkalian matriks.
4. Menghasilkan algoritma baku (standar) untuk tugas-tugas komputasi seperti penjumlahan/perkalian N buah bilangan, menentukan elemen minimum atau maksimum ditabel.

b. Cara Kerja Algoritma Brute Force Berikut ini beberapa langkah cara kerja yang dimiliki oleh brute force, yaitu:

1. Mula-mula string dicocokkan pada awal teks.
2. Dengan bergerak dari kiri kekanan, dibandingkan setiap karakter di dalam string dengan karakter yang bersesuaian di dalam teks, jika sesuai dibandingkan tersebut mengeluarkan hasil.
3. Jika string belum ditemukan kecocokan dari teks belum habis, maka geser string satu karakter ke kanan dan berulang langkah ke 2.

2.3 Android

Android menurut (Safaat, 2012) merupakan sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Android umum digunakan di smartphone dan juga tablet PC. Fungsinya sama seperti sistem operasi Symbian di Nokia, iOS di Apple dan BlackBerry OS.

Menurut Dedekind (2009) Android adalah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, middlewaredan aplikasi. Android menyediakan platform yang terbuka bagi para pengembang untuk

menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel/smartphone. Kemudian untuk mengembangkan Android dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia.

Android tidak terikat ke satu merek *Handphone* saja, beberapa vendor terkenal yang sudah memakai Android antara lain Samsung, Sony Ericsson, HTC, Nexus, Motorola, dan lain-lain. Pada Juli 2000, Google bekerjasama dengan Android Inc., perusahaan yang berada di Palo Alto, California Amerika Serikat.

Para pendiri Android Inc. bekerja pada Google, diantaranya Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Saat itu banyak yang menganggap fungsi Android Inc. hanyalah sebagai perangkat lunak pada telepon seluler. Sejak saat itu muncul rumor bahwa Google hendak memasuki pasar telepon seluler. Di perusahaan Google, tim yang dipimpin Rubin bertugas mengembangkan program perangkat seluler yang didukung oleh kernel Linux. Hal ini menunjukkan indikasi bahwa Google sedang bersiap menghadapi persaingan dalam pasar telepon seluler. Versi android terbaru yaitu versi 4.0. (*Ice Cream Sandwich*).

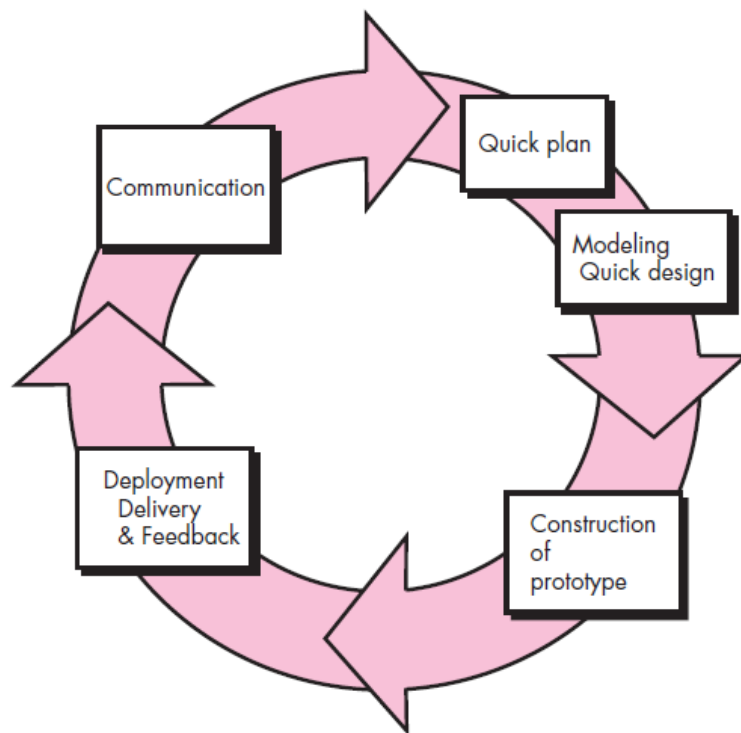
Android juga sudah bergabung dengan beberapa smart mobile seperti LG, Samsung, Sony Ericsson, dan lainnya. Sekitar September 2007 sebuah studi melaporkan bahwa Google mengajukan hak paten aplikasi telepon seluler (akhirnya Google mengenalkan Nexus One, salah satu jenis telepon pintar GSM yang menggunakan Android pada sistem operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran pada 5 Januari 2010). Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan *Open Handset Alliance*, OHA mengumumkan produk perdana mereka, Android, perangkat bergerak (*Mobile*) yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru.

Banyak smartphone dan PC Tablet menggunakan sistem operasi dengan versi yang berbeda. Semakin tinggi versi, fiturnya semakin canggih dan banyak. Telepon pertama yang memakai sistem operasi Android adalah HTC Dream yang dirilis pada tanggal 22 oktober 2008.

2.4 Metode Pengembangan Sistem

Metodelogi yang digunakan untuk membangun sistem ini adalah Model Prototype. Model ini merupakan sebuah pendekatan terhadap pengembangan perangkat lunak yang sistematis, dengan beberapa tahapan, yaitu: System Engineering, Analysis, Design, Coding, Testing dan Maintenance.

Pressman (2010) mengutarakan bahwa prototype adalah pengembangan yang cepat dan pengujian terhadap model kerja (prototipe) dari aplikasi baru melalui proses interaksi dan berulang-ulang yang biasa digunakan ahli sistem informasi dan ahli bisnis. Prototype disebut juga desain aplikasi cepat (rapid application design/RAD) karena menyederhanakan dan mempercepat desain sistem. Bagian user kesulitan mengungkapkan keinginannya untuk mendapatkan aplikasi yang sesuai dengan kebutuhannya. Kesulitan ini yang perlu diselesaikan oleh analis dengan memahami kebutuhan user dan menerjemahkannya ke dalam bentuk model (prototipe). Model ini selanjutnya diperbaiki secara terus menerus sampai sesuai dengan kebutuhan user. Model prototype dapat dilihat pada gambar 2.1 berikut.



Gambar 2.1 Model Prototype (Pressman, 2015)

Penjelasan dari gambar 2.1 diatas adalah sebagai berikut :

- a. Communication dan pengumpulan data awal yaitu tahap suatu perencanaan yang di lakukan, mulai dari menciptakan dan melaksanakan proses untuk memastikan bahwa perencanaan tersebut berkualitas tinggi, terpercaya, efisiensi biaya.
- b. Quick plan yaitu analisis terhadap kebutuhan pengguna.
- c. Modelling quick design yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali.
- d. Construction of prototype adalah pembuatan perangkat prototype termasuk pengujian dan penyempurnaan.
- e. Deployment, delivery, and feedback adalah tahap penyerahan sistem ke pengguna dan umpan balik.

2.5 Unified Modeling Language (UML)

Menurut (Mulyani, 2016) *Unified Modeling Language (UML)* adalah sebuah Teknik pengembangan sistem yang menggunakan Bahasa garfish sebagai alat untuk pendokumentasian dan melakukan spesifikasi pada sistem. Tujuan Penggunaan UML yaitu untuk memodelkan suatu sistem yang menggunakan konsep berorientasi objek dan menciptakan bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin. Menurut Mulyani (2016) tipe-tipe diagram UML adalah sebagai berikut :

2.5.1 Use Case Diagram

Use case diagram adalah gambar dari beberapa atau seluruh aktor dan *use case* dengan tujuan yang mengenali interaksi mereka dalam suatu sistem. *Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* mepresentasikan sebuah interaksi antara actor dan sistem.

Dalam *use case diagram* terdapat istilah seperti aktor, *use case* dan *case relationship*

Tabel 2.1 Komponen *Use Case Diagram*

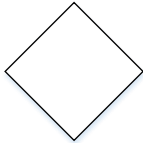
Simbol	Nama elemen	Keterangan
	Aktor	Merupakan kesatuan eksternal yang berinteraksi dengan sistem
	Use Case	Rangkaian / uraian sekelompok yang saling terkait dan membentuk sistem
	Asosiasi	Penghubung antar elemen (aktor/use case) didalam sistem.
	Generalisasi	Sebuah elemen yang dihasilkan dari pewarisan elemen lain.

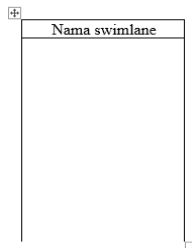
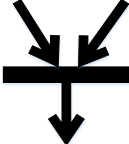
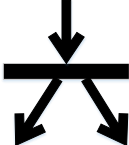
<<extend>>	Extend	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
<<include>>	Include	<i>Use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan.

2.5.2 Activity Diagram

Activity Diagram yang disediakan oleh UML melengkapi *use case* yang telah dibuat sebelumnya memberikan representasi grafis dari aliran–aliran interaksi di dalam suatu skenario yang sifatnya spesifik. Mirip dengan diagram alir, suatu diagram aktifitas menggunakan sebuah kotak yang berisi lengkung untuk menggambarkan fungsi tertentu yang ada dalam suatu sistem yang akan dikembangkan, sementara itu tanda panah menggambarkan aliran didalam sistem dan seterusnya.

Tabel 2.2 Komponen *Activity Diagram*

Simbol	Keterangan
	<i>Activity</i> : Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Initial Node</i> : Bagaimana objek dibentuk atau diawali
	<i>Activity Final Node</i> : Bagaimana objek dibentuk dan diakhiri.
	<i>Decision</i> : Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu.

	<i>Swimlane</i> : Memisalkan organisasi bisnis yang bertanggung jawab terhadap aktifitas yang terjadi.
	<i>Join</i> : Digunakan untuk menunjukkan kegiatan yang digabungkan.
	<i>Fork</i> : Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel

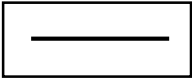
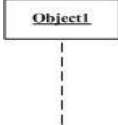
2.5.3 Sequence Diagram

Sequence diagram di UML terutama digunakan untuk memodelkan interaksi antara aktor dan objek dalam sistem dan interaksi antara obyek itu sendiri. UML memiliki sintaks yang kaya untuk sequence diagram, yang memungkinkan berbagai jenis interaksi yang dimodelkan. Sesuai namanya, sequence diagram menunjukkan urutan interaksi yang terjadi antara use case. Sequence diagram memiliki dua buah karakteristik yaitu :

1. Setiap objek memiliki *lifeline* yang digambarkan dengan garis putus-putus vertikal dan garis ini menunjukkan daur hidup dari sebuah objek.
2. Terdapat fokus kontrol yang digambarkan dengan sebuah persegi panjang yang tipis dan tinggi. Fokus kontrol ini menunjukkan periode waktu selama sebuah objek melakukan sebuah event.

Tabel 2.3 Komponen *Sequence Diagram*

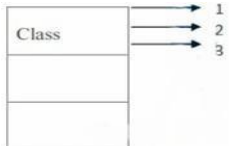
Simbol	Nama Elemen	Keteranga
	<i>Life Line</i>	Objek entitas, antar muka yang saling berinteraksu

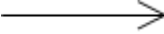
	<i>Activation</i>	Menggambarkan hubungan antar objek dengan <i>message</i>
	<i>Message (call)</i>	Menggambarkan alur <i>message</i> yang merupakan kejadian objek pengirim <i>life line</i> ke objek penerima <i>life line</i>
	<i>Message (return)</i>	Menggambarkan alur pengambilan <i>message</i> ke objek pemanggil dan tanda bahwa objek penerima telah menyelesaikan
	<i>Object</i>	Object adalah instance dari sebuah class yang dituliskan tersusun secara horizontal diikuti lifeline

2.5.4 Class Diagram

Class Diagram adalah diagram yang menunjukkan class-class yang ada dari sebuah sistem dan saling berhubungan secara diagram ini menggambarkan alur struktur statis dari sebuah sistem. Karena itu *Class Diagram* merupakan tulang punggung atau kekuatan dasar dari hampir setiap metode berorientasi objek termasuk UML.

Tabel 2.4 Komponen *Class Diagram*

Simbol	Nama Elemen	Keterangan
	<i>Class</i>	Simbol untuk membangun sebuah pemrograman dengan objek. Terdiri 3 bagian, bagian atas adalah nama kelas, bagian tengah adalah atribut dan bagian bawah adalah metode dari kelas.
	Generalisasi	Simbol yang menandakan adanya generalisasi dari kelas input untuk menghasilkan data yang

	Asosiasi berarah / directed association	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai
	Kebergantungan /dependency	Relasi antar kelas dengan makna ketergantungan antar kelas
	Agresi /aggregation	Relasi antar kelas dengan makna semua – bagian (<i>whole-part</i>)

2.6 Database

Beberapa pengertian *database* menurut para ahli:

1. Menurut Kustiyaningsih (2011), “*Database* adalah Struktur penyimpanan data. Untuk menambah, mengakses dan memperoses data yang disimpan dalam sebuah *database* komputer, diperlukan *sistem* manajemen *database* seperti MySQL Server”.
2. Menurut Anhar (2010), “*Database* adalah sekumpulan tabel-tabel yang berisi data dan merupakan kumpulan dari *field* atau kolom. Struktur file yang menyusun sebuah *database* adalah *Data Record* dan *Field*”.

Berdasarkan beberapa pendapat para ahli yang dikemukakan di atas dapat ditarik kesimpulan bahwa *database* adalah sekelompok data yang mempunyai ciri-ciri khusus dan dapat dikelola sedemikian rupa sehingga bisa menghasilkan sebuah format data yang baru.

2.7 MySQL

Beberapa pengertian MySQL menurut para ahli:

1. Menurut Raharjo (2011:21), “MySQL merupakan RDBMS (atau *server database*) yang mengelola *database* dengan cepat menampung dalam jumlah sangat besar dan dapat di akses oleh banyak *user*”.
2. Menurut Kadir (2008:2), “MySQL adalah sebuah software *open*

source yang digunakan untuk membuat sebuah *database*.”

Berdasarkan pendapat yang dikemukakan di atas dapat ditarik kesimpulan bahwa *MySQL* adalah suatu software atau program yang digunakan untuk membuat sebuah *database* yang bersifat *open source*.

2.8 Penelitian Terdahulu

Penelitian terdahulu dalam pada penelitian ini penulis mengambil 5 penelitian yaitu:

Tabel 2.5 Penelitian Terkait

No	Peneliti	Judul	Masalah
1	Bayu Widia Santoso, Firdiansyah Sundawa, dan Muhammad Azhari (2016)	Implementasi Algoritma Brute Force Sebagai Mesin Pencari (Search Engine) Berbasis Web Pada Database	Keterbatasan kemampuan sumber daya manusia dalam hal ini dokter atau perawat dalam mencari informasi yang diinginkan dalam waktu yang lebih singkat. Keterbatasan inilah yang sering kali menyebabkan tertundanya tugas-tugas. Untuk itu, dibuatlah search engine untuk mempercepat dan memudahkan dalam pencarian informasi yang dibutuhkan.
2	Amin Siddiq Sumi, Purnawansyah, dan Lukman Syafie (2018)	Analisa Penerapan Algoritma Brute Force Dalam Pencocokan String	Pencarian string juga diharapkan untuk dapat menuntaskan pekerjaannya dengan cepat. Sehingga hasil dari pencarian itu, dapat digunakan tepat, cepat, dan efisien. Contoh dari penggunaan pencarian string, antara lain adalah search engine untuk website, pencarian kata di suatu dokumen, pembuatan program kamus, dan data mining. Untuk memecahkan masalah ini, dibuatlah suatu algoritma yang memanfaatkan kemampuan computer untuk menjalankannya. Algoritma ini melakukan pencarian dengan

			suatu jalan tertentu yang terurut, tidak ambigu, dan dapat dijalankan dengan baik oleh komputer.
3	Nur Afif (2018)	Implementasi Algoritma Brute Force Dalam Perancangan Aplikasi Penelusuran Skripsi	Proses menemukan ide skripsi sampai persetujuan judulnya oleh dosen, membutuhkan waktu yang cukup lama. Banyak faktor yang menyebabkan hal ini terjadi. Salah satunya adalah seringnya terjadi kesamaan ide/topik skripsi yang diajukan oleh mahasiswa dengan skripsi yang sudah ada sebelumnya. Hal ini disebabkan karena kurangnya informasi mahasiswa mengenai topik-topik skripsi yang sudah pernah diangkat. Akses koleksi skripsi di perpustakaan Fakultas Sains dan Teknologi maupun di perpustakaan pusat UIN Alauddin Makassar masih terbatas
4	Heny Pratiwi, Ita Arfyanti dan Dendy Kurniawan(2016)	Implementasi Algoritma Brute Force Dalam Aplikasi Kamus Istilah Kesehatan	Untuk mencari kosa kata yang diinginkan masih terdapat kesulitan dalam proses pencariannya karena kamus-kamus tersebut berukuran besar dan tebal sehingga waktu yang diperlukan dalam proses pencarian menjadi lama dan tidak efektif. Oleh sebab itu, dibuat suatu aplikasi kamus berbasis desktop yang bisa digunakan dengan secara lebih efisien dalam mencari istilah kata pada kamus yang sebenarnya. Pada penelitian ini dirancang dengan menggunakan algoritma string matching.
5	Anis Mirza (2017)	Pencarian Data Tiker Maintanance	Proses pencarian data tiket maintencance dilakukan dengan

		Menggunakan Metode Brute Force	mengandalkan data-data softcopy yang dikirim melalui email oleh karyawan dan hardcopy. Dengan digunakan sistem tersebut menyulitkan admin untuk memperoleh informasi laporan maintenance yang diperlukan, karena harus mencari data laporan karyawan di storage computer.
6	Yuni Puspita Sari, Isnandar Agus, Endang Puspita Sari (2021)	Smart Trolley Pada Fitrinopane Swalayan Dengan Menerapkan Metode Brute Force Berbasis Mobile	Dengan jumlah kasir yang terbatas, tentu saja menimbulkan permasalahan seperti penumpukan antrean dikasir. Pada sisi konsumen, seringkali konsumen tidak memperhatikan total harga dari belanjaan yang telah diambil. Sehingga, menimbulkan permasalahan ketidaksesuaian budget dengan total akumulasi dari barang yang diambil.
7	Iskandar Saputro Bakri, Hariyanto Wibowo (2019)	Rancang Bangun Aplikasi Kamus Semboyan PT. Kereta Api Indonesia (Persero) Berbasis Android	PT. Kereta Api Indonesia (Persero) memiliki 98 pengertian. Jadi sangatlah sulit jika kita ingin mencari informasi mengenai suatu pengertian semboyan kereta api yang diinginkan. Dan akan cukup lama menemukan pengertian semboyan yang dicari secara manual. Pencarian pada nama semboyan kereta api sangat perlu mengingat banyaknya nama semboyan. Sehingga, dengan menggunakan satu kata kunci dari nama semboyan, dapat dengan cepat mendapatkan nama semboyan yang dicari.