

BAB II

TINJAUAN PUSTAKA

2.1 Perancangan

Perancangan adalah penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah berfungsi dalam bentuk system flowchart. (Nafisah, 2003).

2.2 Aplikasi

Menurut (Sitanggang, D., Sihombing, N., Silalahi, L., & Sirait, M. M. 2020) Aplikasi biasanya berupa perangkat lunak yang berbentuk *software* yang berisi kesatuan perintah atau program yang dibuat untuk melaksanakan sebuah pekerjaan yang diinginkan. Selain itu aplikasi juga mempunyai fungsi sebagai pelayan kebutuhan beberapa aktivitas yang dilakukan oleh manusia seperti sistem untuk *software* jual – beli, permainan atau *online game*, pelayanan masyarakat dan hampir semua proses yang dilakukan oleh manusia dapat dibantu dengan menggunakan suatu aplikasi.

2.3 Aplikasi Database

Aplikasi Database merupakan salah satu jenis aplikasi computer, yang banyak di gunakan dalam dunia bisnis, di sebut juga sebagai aplikasi bisnis setiap Lembaga atau organisasi, baik profi atau non profit, yang ingin administrasi nya rapi dan terkelola dengan baik, maka akan memerlukan aplikasi untuk mengelola data base nya.

Aplikasi database adalah aplikasi yang berurusan dengan bagaimana melakukan pengelolaan data dengan memanfaatkan system database, kemudian memprosesnya untuk menghasilkan informasi yang di perlukan oleh manajemen sebagai pengguna informasi, untuk membantu dalam proses pengambilan keputusan.

2.4 Divisi Intelkam

Intelijen Keamanan Polri yang selanjutnya disingkat Intelkam Polri adalah

Intelijen yang diimplementasikan dalam penyelenggaraan fungsi kepolisian sebagai salah satu fungsi pemerintahan negara, dalam rangka mewujudkan keamanan dalam negeri (LEMDIKLAT POLRI 2020)

2.5 SKCK

SKCK adalah surat keterangan resmi yang diberikan oleh Polri kepada seseorang warga masyarakat untuk memenuhi permohonan dari yang bersangkutan untuk suatu keperluan karena adanya ketentuan yang mempersyaratkan berdasarkan hasil penelitian biodata anteseden orang tersebut.

2.6 Android

Android merupakan sistem operasi berbasis linux yang bersifat terbuka (open source) dan dirancang untuk perangkat seluler layar sentuh seperti *smartphone* dan komputer tablet. Tampilan android didasarkan pada manipulasi langsung, menggunakan masukan sentuh yang serupa dengan tindakan di dunia nyata, seperti menggesek, mengetuk, mencubit dan membalikkan cubitan untuk memanipulasi obyek di layar. Sifat android yang terbuka telah membuat bermunculannya sejumlah besar komunitas 21 pengembang aplikasi untuk menggunakan android sebagai dasar proyek pembuatan aplikasi, dengan menambahkan fitur-fitur baru bagi android pada perangkat yang secara resmi dirilis dengan menggunakan sistem operasi lain. (Salbino, 2014).

Android adalah sebuah kumpulan perangkat lunak untuk perangkat mobile yang mencakup sistem operasi, *middleware* dan aplikasi utama mobile. Android memiliki empat karakteristik sebagai berikut:

Terbuka Android dibangun untuk benar-benar terbuka sehingga sebuah aplikasi dapat memanggil salah satu fungsi inti ponsel seperti membuat panggilan, mengirim pesan teks, menggunakan kamera, dan lain-lain. Android menggunakan sebuah mesin virtual yang dirancang khusus untuk mengoptimalkan sumber daya memori dan perangkat keras yang terdapat didalam perangkat. Android merupakan open source, dapat secara bebas diperluas untuk memasukkan teknologi baru yang lebih maju pada saat teknologi tersebut muncul. Platform ini akan terus berkembang untuk membangun aplikasi *mobile* yang inovatif.

Semua aplikasi dibuat sama Android tidak memberikan perbedaan terhadap aplikasi utama dari telepon dan aplikasi pihak ketiga (*third-party application*). Semua aplikasi dapat dibangun untuk memiliki akses yang sama terhadap kemampuan sebuah telepon dalam menyediakan layanan dan aplikasi yang luas terhadap para pengguna.

Memecahkan hambatan pada aplikasi Android memecah hambatan untuk membangun aplikasi yang baru dan inovatif, misalnya, pengembang dapat menggabungkan informasi yang diperoleh dari web dengan data pada ponsel seseorang seperti kontak pengguna, kalender, atau lokasi geografis.

Pengembangan aplikasi yang cepat dan mudah Android menyediakan akses yang sangat luas kepada pengguna untuk menggunakan library yang diperlukan dan tools yang dapat digunakan untuk membangun aplikasi yang semakin baik. Android memiliki sekumpulan tools yang dapat digunakan sehingga membantu para pengembang dalam meningkatkan produktivitas pada saat membangun aplikasi yang dibuat Google Inc. sepenuhnya membangun Android dan menjadikannya bersifat terbuka (*open source*) sehingga para pengembang dapat menggunakan Android tanpa mengeluarkan biaya untuk lisensi dari Google dan dapat membangun Android tanpa adanya batasan-batasan. *Android Software Development Kit* (SDK) menyediakan alat dan *Application Programming Interface* (API) yang diperlukan untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java.

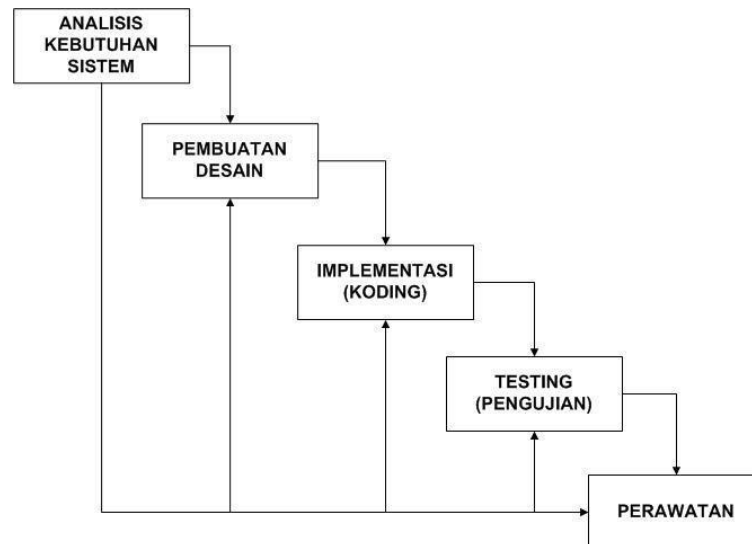
2.7 Android Studio

Android Studio adalah sebuah IDE untuk Android Development yang dikenalkan pihak google pada acara Google I/O di tahun 2013. Android Studio merupakan suatu pengembangan dari Eclipse IDE. (Smith, 2017).

2.8 Waterfall Programing

Waterfall Model Design, adalah model SDLC pertama yang banyak digunakan Software Engineering. Tujuannya, tidak lain ialah untuk memastikan keberhasilan proyek. Pendekatan ini, membagi keseluruhan proses menjadi beberapa fase. Yang mana, hasil (output) Dalam pendekatan “The Waterfall”,

seluruh proses pengembangan perangkat lunak dibagi menjadi beberapa fase terpisah. Dalam model Air Terjun ini, biasanya, hasil dari satu fase bertindak sebagai input untuk fase berikutnya secara berurutan.



Gambar 2.1 Diagram Waterfall

Kelebihan dan Kekurangan *Waterfall Model*

Kelebihan yang jelas terlihat dengan menggunakan *Waterfall Model* adalah dimungkinkannya departemensi yang memudahkan kontrol. *Deadline* dapat ditentukan untuk setiap fase dan produk dapat dikembangkan secara maksimal dengan melewati seluruh fase.

Beberapa keuntungan utama dari Model Air Terjun adalah sebagai berikut:

- Sederhana dan mudah dimengerti dan digunakan
- Mudah dikelola karena kekakuan model. Setiap fase memiliki hasil tertentu dan proses peninjauan.
- Fase diproses dan diselesaikan satu per satu.
- Bekerja dengan baik untuk proyek yang lebih kecil di mana persyaratan dipahami dengan sangat baik.
- Tahapan yang jelas.
- Mudah mengatur tugas.

- Proses dan hasilnya didokumentasikan dengan baik.

Kerugian dari pengembangan air terjun adalah bahwa ia tidak memungkinkan banyak refleksi atau revisi. Setelah aplikasi berada dalam tahap pengujian, sangat sulit untuk kembali dan mengubah sesuatu yang tidak terdokumentasi dengan baik atau dipikirkan dalam tahap konsep.

Kerugian utama dari Model Air Terjun adalah sebagai berikut :

- Tidak ada perangkat lunak yang berfungsi yang diproduksi hingga larut selama siklus hidup.
- Jumlah risiko dan ketidakpastian yang tinggi.
- Bukan model yang baik untuk proyek yang kompleks dan berorientasi objek.
- Model yang buruk untuk proyek yang panjang dan berkelanjutan.
- Tidak cocok untuk proyek-proyek di mana persyaratan berada pada risiko perubahan sedang hingga tinggi. Jadi, risiko dan ketidakpastian tinggi dengan model proses ini.
- Sulit untuk mengukur hal

2.9 Clustering

Menurut Eko Prasetyo (2012:6), pengelompokan data-data ke dalam sejumlah kelompok (cluster) berdasarkan kesamaan karakteristik masing-masing data pada kelompok-kelompok yang ada. Pengelompokan data dibedakan menurut fakultas dan jurusan. Menurut struktur, pengelompokan dibagi dua, yaitu hierarki dan partitioning. Dalam hierarki, satu data tunggal bisa dianggap sebuah kelompok, dua atau lebih. Pengelompokan partitioning membagi setiap data hanya menjadi anggota satu kelompok. Menurut keanggotaan data dalam kelompok, dibagi menjadi dua, yaitu eksklusif dan tumpang-tindih. Dalam kategori eksklusif, sebuah data bisa dipastikan hanya menjadi anggota satu kelompok dan tidak menjadi anggota kelompok yang lain. Sedangkan kategori tumpang-tindih adalah metode pengelompokan yang membolehkan sebuah data menjadi anggota lebih dari satu kelompok.











2.10 Use Case Diagram

Use Case adalah deskripsi fungsi sebuah system dari perspektif pengguna. Use Case bekerja dengan cara mendeskripsikan tipikal interaksi antara pengguna sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah system dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan system disebut sebagai Scenario. Setiap Scenario menggambarkan urutan kejadian. Setiap urutan di inisialisasi oleh orang, system yang lain, perangkat keras atau urutan waktu. Dengan demikian, secara singkat bias dikatakan Use Case adalah serangkaian Scenario yang digabungkan bersaa-sama oleh tujuan umum pengguna. Use Case dibuat berdasarkan kebutuhan Aktor. Use Case harus merupakan 'apa' yang dikerjakan software aplikasi, bukan 'bagaimana' software aplikasi mengerjakannya .

berikut ini adalah Simbol-simbol yang digunakan dalam *Use Case Diagram*

Simbol Use Case Diagram






Tabel 2.1 simbol Use Case Diagram

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|-----------------------|--|
| 1 |  | <i>Actor</i> | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> . |
| 2 |  | <i>Dependency</i> | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>). |
| 3 |  | <i>Generalization</i> | Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>). |
| 4 |  | <i>Include</i> | Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> . |
| 5 |  | <i>Extend</i> | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan. |
| 6 |  | <i>Association</i> | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 7 |  | <i>System</i> | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 8 |  | <i>Use Case</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor |
| 9 |  | <i>Collaboration</i> | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>). |
| 10 |  | <i>Note</i> | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi |

2.11 Activity Diagram

Activity Diagram menggambarkan rangkaian aliran dari aktifitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti use case atau interaksi. Activity Diagram berupa flow chart yang digunakan untuk memperlihatkan aliran kerja dari sistem. Notasi yang digunakan dalam activ.


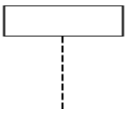

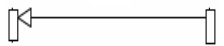
Tabel 2.2 Simbol Activity Diagram

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|----------------------------|---|
| 1 |  | <i>Actifity</i> | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain |
| 2 |  | <i>Action</i> | State dari sistem yang mencerminkan eksekusi dari suatu aksi |
| 3 |  | <i>Initial Node</i> | Bagaimana objek dibentuk atau diawali. |
| 4 |  | <i>Actifity Final Node</i> | Bagaimana objek dibentuk dan dihancurkan |
| 5 |  | <i>Fork Node</i> | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran |

2.12 Sequence Diagram

Sequence diagram merupakan UML yang menggambarkan interaksi antar objek di dalam dan disekitar sistem, termasuk pengguna, display, dan sebagainya berupa message yang digambarkan terhadap waktu, berikut adalah symbol-simbol Sequence diagram terdapat pada tabel

Tabel 2.3 Simbol-simbol Sequence Diagram

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|-----------------|--|
| 1 |  | <i>Actor</i> | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi. |
| 1 |  | <i>LifeLine</i> | Objek <i>entity</i> , antarmuka yang saling berinteraksi. |
| 2 |  | <i>Message</i> | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |
| 3 |  | <i>Message</i> | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |

2.13 Algoritma K-Medoids

Menurut Listiyanti, Youanda Alim Syahbana & Henim (2016) "Algoritma k-medoids adalah algoritma pengelompokan yang berkaitan dengan algoritma k-medoids dan algoritma medoidshift". Algoritma k-medoids dikembangkan oleh Leonard Kaufman dan Peter J. Rousseeuw pada tahun 1987. Algoritma k-medoids sering disebut juga algoritma Partitioning Around Medoids (PAM)[6]. Algoritma k-medoids memiliki kesamaan dengan algoritma k-medoids yaitu sama-sama termasuk algoritma partitioning. Algoritma partitioning merupakan algoritma pengelompokan data ke dalam sejumlah cluster tanpa adanya struktur hirarki antara satu dengan yang lainnya. Algoritma k-medoids memiliki keunggulan dibandingkan dengan algoritma k-medoids. "K-medoids memiliki kinerja yang lebih optimal jika jumlah data yang digunakan berjumlah sedikit. Algoritma ini menggunakan objek pada kumpulan objek untuk mewakili sebuah cluster, Objek yang terpilih untuk mewakili sebuah cluster disebut medoids". Beberapa langkah-langkah dalam perhitungan algoritma k-medoids adalah:

2.13.1 Preposing data

1. Cleaning data

Menghapus data-data yang tidak relevan atau tidak valid

2. Transformasi data

Dengan mengubah menjadi angka numerik

3. Normalisasi data

Rumus normalisasi data terdapat pada gambar 2.3 berikut

$$x^1 = \frac{x_i - \min(x)}{\max(x) - \min(x)}$$

Gambar 2.2 Rumus normalisasi data

2.13.2 Perhitungan secara manual

1. Tentukan k (jumlah cluster) yang di inginkan

2. Pilih secara acak medoid awal sebanyak K dari N data

3. Hitung jarak masing-masing objek ke medoid sementara

Menggunakan rumus equidience distance ang terdapat pda gambar 2.2

$$d = \sqrt{(x^1 + x^2)^2 + (y^1 + y^2)^2}$$

Gambar 2.3 Rumus Equidistance Distance

4. Tandai jarak terdekat objek ke medoid dan hitung totalnya
5. Penentuan anggota cluster terhadap medoids sementara
6. Lakukan literasi medoids
7. Hitung total simpangan dengan mencari selisih total medoid baru total medois lama.

1.14 Penelitian Terkait

Table 2.4 Tabel Penelitian Terkait

| No | Nama | Judul | Latar belakang | Uraian | hasil | kesimpulan |
|----|-------------|--|--|--|---|---|
| 1 | Nadia Firly | Crate your own android application (2018) | Pembelajaran cara membuat aplikasi android | Cara membuat aplikasi android | Memberikan pengajaran pembuatan aplikasi dan tips | Memberikan keuntungan dengan membuat aplikasi android |
| 2 | Betha Sidik | Mengembangkan framework aplikasi Database codeigniter 3 (2019) | Pengembangan aplikasi untuk memenuhi kebutuhan | Iplementasi database dengan framework igniter 3 pada data karyawan | Pengembangan database pada framework code igniter 3 | Sudah dapat membuat prototipe aplikasi |

Tabel 2.4 (Lanjutan)

| | | | | | | |
|---|---|--|---|--|---|--|
| 3 | Fintri Indriyani ¹ , Eni Irfiani ² | Clustering Data Penjualan pada Toko Perlengkapan Outdoor Menggunakan Metode K-Means (2019) | Kemampuan untuk memperkirakan volume penjualan secara akurat pada setiap item yang dijual di toko sangat penting guna kelangsungan hidup pada bisnis. | Analisis penjualan dengan clustering | Penerapan metode K-Means dalam pengelompokan data penjualan pada Toko Genta Corp dapat menghasilkan rekomendasi barang yang laris, Kurang laris dan cukup laris. | Untuk pengembangan lebih lanjut dan agar pihak toko mudah dalam penggunaan metode ini maka perlu dibuat sebuah aplikasi dengan acuan metode K-Means. |
| 4 | Astrika Praokta Fialine ¹ , Daffa Alya Alodia ^{2*} , Desy Endriani ³ , Edy Widodo ⁴ | Implementasi Metode K-Medoids Clustering untuk Pengelompokan Provinsi di Indonesia Berdasarkan Indikator Pendidikan (2021) | memastikan pendidikan yang inklusif dan berkualitas setara, serta mendukung kesempatan belajar seumur hidup bagi semua. | satu tujuan dalam Sustainable Development Goals (SDGs) | Dari hasil analisis, diperoleh tiga cluster. Cluster pertama terdapat 6 provinsi dengan indikator pendidikan tinggi, cluster kedua terdapat 13 provinsi dengan indikator pendidikan sedang, dan cluster ketiga terdapat 15 provinsi dengan indikator pendidikan rendah. | 15 provinsi dengan indikator pendidikan rendah. |

Tabel 2.4 (Lanjut)

| | | | | | | |
|---|---|--|---|--|--|--|
| 5 | Nisar, Wasilah Haris Kusumajaya | Pemanfaatan K Means Clustering dalam Pengelompokan Judul Skripsi (2022) | Peningkatan jumlah mahasiswa dan variasi judul skripsi menyebabkan banyaknya data skripsi yang harus diolah. Hal ini menyebabkan kesulitan dalam pengelompokan data skripsi | kesulitan dalam pengelompokan data skripsi | clustering K-Means melakukan pengelompokan skripsi dengan membagi data menjadi sejumlah k cluster yang ditentukan, dan menggunakan perhitungan jarak untuk mengukur kemiripan antar data | Data mining menggunakan k-mean mampu melakukan pengelompokan data dalam jumlah besar dengan sangat cepat sehingga dapat membantu mempercepat proses pengelom |
| 6 | Haryati Ningrum 1, Eka Irawan 2, Muhammad Ridwan Lubis3 | Implementasi Metode K-Medoids Clustering Dalam Pengelompokan Data Penyakit Alergi Pada Anak (2021) | untuk dapat memutuskan mata rantai penyakit alergi khususnya pada anak diperlukan imunisasi minimal 95%. Untuk bisa mewujudkan hal ini pemerintah | Data diperoleh dari BPS mengenai data persentasi penyakit alergi pada anak berdasarkan provinsi di Indonesia pada tahun 2011-2019. | dimana diperoleh 21 provinsi dengan cluster rendah yang menjadi pusat perhatian bagi pemerintah terutama Bagi Dinas Kesehatan | Penerapan data mining dengan metode k-medoids clustering dapat diterapkan. |

| | | | | | | |
|--|--|--|---|--|--|--|
| | | | memerlukan data yang akurat untuk melakukan pengelompokan data penderita penyakit alergi pada anak. | | | |
|--|--|--|---|--|--|--|