

BAB II

LANDASAN TEORI

2.1 Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya Open Handset Alliance, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel Android pertama mulai dijual pada bulan Oktober 2008.

Android adalah sistem operasi dengan sumber terbuka, dan Google merilis kodenya di bawah Lisensi Apache. Kode dengan sumber terbuka dan lisensi perizinan pada Android memungkinkan perangkat lunak untuk dimodifikasi secara bebas dan didistribusikan oleh para pembuat perangkat, operator nirkabel, dan pengembang aplikasi. Selain itu, Android memiliki sebuah komunitas yang besar dalam pengembang aplikasi (*apps*) yang memperluas fungsionalitas perangkat, umumnya ditulis dalam versi kustomisasi bahasa pemrograman Java. Pada tahun 2013 lebih dari satu juta aplikasi yang tersedia untuk android di playstore, dan hamper 500 miliar aplikasi telah di unduh dari took aplikasi utaman android dan google play. Sebuah survei pada bulan April-Mei 2013 menemukan bahwa Android adalah platform paling populer bagi para pengembang, digunakan oleh 71% pengembang aplikasi bergerak. Di Google I/O 2014, Google melaporkan terdapat lebih dari satu miliar pengguna aktif bulanan Android, meningkat dari 583 juta pada bulan Juni 2013.

Android adalah sebuah sistem linux yang mencakup sistem operasi, middleware, dan aplikasi. Awal mula android dikembangkan oleh android Inc, sebuah perusahaan baru yang dibeli oleh Android Inc. Untuk pengembangannya, dibentuklah Open Handset Alliance (OHA), konsorsium dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia (*Nazrudin Safaat H, 2011, p.1*).

2.1.1 Statistik Distribusi OS Android

Google baru-baru ini merilis angka statistik distribusi sistem operasi Android di berbagai perangkat mobile di seluruh dunia pada bulan April 2017. Dalam statistik tersebut terlihat Android 5.0 dan 5.1 Lollipop masih menjadi yang tertinggi dengan persentase total 32%. Di urutan kedua berhasil dikuasai oleh Android 6.0 Marshmallow yang memiliki persentase mencapai 31,2%. Kedua sistem operasi ini memang layak menduduki angka tertinggi karena saat ini smartphone Android memang masih banyak yang menggunakan Android Lollipop dan Marshmallow mengingat vendor-vendor di seluruh dunia masih mempercayakan kedua sistem operasi ini pada smartphone buatannya. Sementara pada urutan ketiga diduduki Android KitKat dengan persentase 20% yang disusul oleh Android Jelly Bean dengan persentase total 10,1%. kedua sistem operasi ini masih banyak digunakan pada smartphone entry-level di berbagai negara. Selanjutnya, Android versi Ice Cream Sandwich dan Gingerbread sudah mulai ditinggalkan pengguna Android karena hanya memiliki persentase 0,9%. Hal ini tentu wajar karena Google sendiri sudah menghentikan dukungan untuk sistem operasi Gingerbread ini pada Q1 2017. Sementara versi Ice Cream Sandwich yang meski terus melempem tetapi masih akan didukung dalam waktu lebih lama.

2.1.2 Peranti Pengembang Android

Situs Android Developers menyediakan *Android SDK (Software development kit)* yang memudahkan untuk membuat aplikasi android. Peranti yang diperlukan untuk pengembangan aplikasi android antara lain *Java Development Kit (JDK)*, Eclipse, Android SDK, dan *Android Development Tools (ADT)*. *Java Development Kit (JDK)* adalah perangkat pengembang aplikasi java yang sangat diperlukan untuk membuat aplikasi android. Hal tersebut dikarenakan android menggunakan bahasa pemrograman java (Kadir, 2013). Java adalah bahasa pemrograman yang multi platform dan multi device. Secara umum java merupakan bahasa pemrograman berorientasi objek murni yang dibuat berdasarkan kemampuan terbaik bahasa pemrograman objek sebelumnya. Java diciptakan oleh oleh James Gosling, seorang developer dari Sun Microsystems pada tahun 1991 dengan nama semula Oak. Pada Januari 1995, karena nama Oak kurang komersil dan sebuah perangkat lunak sudah terdaftar dengan merk dagang tersebut, sehingga di ambil nama penggantinya menjadi Java. Nama Java sendiri diambil dari kopi murni yang digiling langsung dari biji kopi (kopi tubruk) kesukaan Gosling.

Eclipse adalah perangkat pengembang aplikasi yang tergolong sebagai IDE (integrated development environment), karena menyediakan berbagai fasilitas untuk pembuatan aplikasi. Perangkat lunak ini digunakan sebagai peranti pengembang aplikasi yang menggunakan bahasa seperti java, C++, dan Python (Kadir, 2013).

Android SDK adalah kumpulan *software* yang berisi mengenai pustaka, *debugger* (alat pencari kesalahan program), emulator (peniru perangkat bergerak), dokumentasi, kode contoh, dan panduan (Kadir, 2013). Dalam pengembangan aplikasi android menyediakan Android SDK yang menyediakan tools dan API untuk para pengembang aplikasi dengan platform Android. Android menggunakan Java sebagai bahasa pemrogramannya.

2.1.3 Kelebihan atau Keunggulan Android

Kelebihan-kelebihan dari android adalah sebagai berikut :

1. User Friendly – Kalimat ini sangat melekat pada sistem operasi Windows miliknya microsoft, ibaratnya kita dengan sangat mudah mengoperasikan komputer hanya dengan belajar beberapa hari bahkan beberapa jam saja, dan ini juga melekat pada Android yang berjalan pada Smartphone.
2. Notifications – Anda dengan sangat mudah mendapatkan notifikasi dari smartphone android dengan mengatur beberapa akun Email, SMS , Voice Dial, Update dan lain sebagainya.
3. Tampilan – Dari segi tampilan, Android tidak kalah bagusnya dari iOS miliknya Apple, karena memang dari awal android hampir mengusung teknologi iOS, hanya saja ini versi murahnya.
4. Open Source – Operating system ini memang dibuat open source oleh penciptanya, karena memang berbasis kernel Linux. Jadi jangan kaget jika diluar sana ada banyak sekali Custom Rom untuk masing-masing perangkat android.
5. Aplikasi – Untuk Aplikasi anda akan disajikan jutaan pilihan aplikasi yang menarik dari yang gratis hingga berbayar, dan anda bisa mendownloadnya di Google Play, siapa lagi kalau bukan miliknya Mbah Google.
6. Dll – Dan masih banyak lagi kelebihan dan fitur yang dimiliki Android, dan anda bisa mengembangkannya sesuai dengan keinginan anda, karena Sistem operasi ini Open Source dan User Friendly.

2.1.4 Kekurangan atau Kelemahan Android

Kekurangan-kekurangan dari android adalah sebagai berikut :

1. Update System – Seperti atikel saya disini untuk melakukan update system tidaklah mudah, anda harus menunggu dari masing-masing Vendor untuk merilis Update Versi yang terbaru. Tentu ini sangat tidak menyenangkan.
2. Batterai Cepat Habis – Ini sering terjadi jika anda menyalakan paket data dan menggunakan widget serta aplikasi yang berjalan secara berlebihan, untuk mengatasinya anda harus mengurangi aktivitas aplikasi pada smartphone anda.
3. Lemot atau Lag – Sebenarnya kalau ini ada kaitannya dengan spesifikasi dari masing-masing perangkat, namun ada kalanya Android ini tidak bersahabat dengan beberapa aplikasi dikarenakan Ram ataupun Prosesornya yang kurang memadai, jadi sesuaikan aplikasi dengan perangkatnya.

2.1.5 Jenis-jenis Android

1. Android Jellybean

Versi 4.1 sampai 4.3 merupakan awal kesuksesan bagi Android hingga bisa dikenal banyak pengguna. Versi ini sendiri menawarkan beragam fitur optimalisasi kamera dengan layanan panorama dan sektor grafis lainnya.

2. Android Kitkat 4.4

Pada versi 4.4 ini Android mengeluarkan gebrakan baru berupa teknologi asisten digital yang disebut “OK Google”. Versi kitkat ini sendiri disebut-sebut sebagai versi Android terbaik dan menjadi yang paling favorit karena dapat mendukung banyak jenis perangkat smartphone. Kitkat rilis sejak Oktober 2013.

3. Android Lollipop 5.0 dan 5.1

Pada Juni 2014, Android versi 5.0 dirilis dengan mengeluarkan fitur untuk mendukung penggunaan RAM di atas 3 GB. Pada versi ini pula desain userinterface semakin baik. Namun versi ini dinilai tidak terlalu membawa perubahan yang signifikan dibanding pendahulunya.

4. Android Marshmallow 6.0

Android versi ini menawarkan gaya baru yang lebih dinamis dengan menambahkan sensor pemindai sidik jari sebagai fitur keamanan smartphone. Kemudian hadir pula fitur search bar dan mendukung fitur multi-window. Marshmallow dirilis pada Oktober 2015.

5. Android Nougat 7.0 dan 7.1

Android versi ini menghadirkan kualitas google assistant yang jauh lebih baik dibanding versi sebelumnya. Nougat sendiri mengeluarkan fitur split-screen yang berhasil menyedot banyak perhatian pengguna smartphone. Nougat 7.0 resmi dirilis sejak Maret 2016.

6. Android Oreo 8.0

Versi ini dirilis pada Agustus 2017 dengan mengeluarkan berbagai fitur multitasking dengan sistem UI yang rapih.

7. Android Pie 9.0

Android 9.0 resmi dirilis sejak Agustus 2018. Versi ini sendiri merupakan sistem operasi yang banyak digunakan oleh smartphone Android saat ini. Dirancang dengan fitur smart-reply dan adaptive battery. Serta membawa perubahan dalam hal navigasi tombol fisik pada home, back dan recent apps.

2.2 Pengetian Aplikasi

Aplikasi adalah suatu atribut yang terdiri dari beberapa form, report yang disusun sebaik mungkin agar dapat mengakses data. Aplikasi merupakan suatu program yang berisikan perintah-perintah untuk melakukan pengolahan data, secara umum adalah suatu proses dari cara manual yang ditransformasikan ke komputer dengan membuat sistem atau program agar data diolah lebih berdaya guna secara optimal (Rahman dan Santoso 2015:79).

2.3 Sistem Pakar

Sistem Pakar adalah suatu sistem yang digunakan untuk konsultasi yang berisikan informasi pengetahuan dari pakar. Pengetahuan dari pakar digunakan sebagai dasar oleh Sistem Pakar untuk menjawab pertanyaan (konsultasi).Kepakaran (*expertise*) adalah rangkaian pelatihan, membaca, dan pengalaman dengan pengetahuan yang ekstensif dan spesifik. Tujuan Sistem Pakar adalah untuk mentransfer kepakaran dari seorang pakar ke komputer, kemudian ke orang lain (yang bukan pakar). Sistem pakar merupakan salah satu bagian dari kecerdasan buatan. Sistem ini dirancang untuk menirukan keahlian seorang pakar dalam menjawab pertanyaan dan menyelesaikan suatu permasalahan baik di bidang kesehatan atau kedokteran, bisnis, ekonomi dan sebagainya.

Sistem pakar merupakan sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran untuk memecahkan masalah yang hanya dapat dipecahkan oleh seorang pakar (Jarti & Trisno,2017)

2.3.1 Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan merupakan representasi dari pengetahuan (*knowledge representation*) dari seorang pakar. Basis pengetahuan tersusun atas fakta yang berupa obyek dan aturan (*rule*). Daftar fakta-fakta (*facts list*) berisikan hasil observasi dan sesuatu kenyataan yang dibutuhkan selama pengolahan. Basis pengetahuan ini adalah inti dari pemrograman sistem pakar yang menyimpan informasi dan aturan-aturan bagi sistem pakar.

2.3.2 Mesin Inferensi (*Inference Engine*)

Mesin inferensi merupakan bagian sistem pakar yang mengandung mekanisme fungsi berpikir dan pola pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme ini digunakan untuk menguji aturan dalam urutan tertentu untuk mencari yang sesuai dengan kondisi awal atau yang sedang berjalan. Kemudian diarahkan untuk mengendalikan langkah-langkah yang diambil untuk menyelesaikan masalah.

2.4 Metode Naïve Bayes

Naive Bayes merupakan sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Algoritma naïve Bayes merupakan perhitungan yang mengasumsikan semua atribut independen atau tidak saling ketergantungan yang diberikan oleh nilai pada variabel kelas. Definisi lain mengatakan Naive Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya (**Patil and Sherekar 2013**).

Naive Bayes didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai output. Dengan kata lain, diberikan nilai output, probabilitas mengamati secara bersama adalah produk dari probabilitas individu. Keuntungan penggunaan Naive Bayes adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (Training Data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. Naive Bayes sering bekerja jauh lebih baik dalam kebanyakan situasi dunia nyata yang kompleks dari pada yang diharapkan.

Algoritma Naive Bayes berbasiskan perhitungan probabilistik dengan asumsi bahwa setiap fitur yang digunakan saling lepas. Naive Bayes merupakan metode klasifikasi teks yang paling populer digunakan. Beberapa kelebihan algoritma ini adalah memiliki kecepatan pembelajaran dan toleransi terhadap nilai yang hilang dari sistem atau fitur menangani data numerik, algoritma ini menggunakan probability density function, artinya data dianggap mengikuti distribusi normal untuk kemudian dihitung nilai rata-rata dan simpangan bakunya.

2.4.1 Algoritma *naïve bayes*

Algoritma Naïve Bayes adalah salah satu algoritma yang terdapat pada teknik klasifikasi. Naïve Bayes dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga disebut sebagai Teorema Bayes. Teorema Bayes dikombinasikan dengan Naïve dimana diasumsikan dengan kondisi antar atribut yang saling bebas. Klasifikasi Naïve Bayes diasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak berhubungan dengan ciri dari kelas lain. Persamaan dari teorema Bayes adalah sebagai berikut.

$$P(H|X) = \frac{P(X|H) \cdot P(H)}{P(X)}$$

Keterangan:

X : Data dengan kelas yang belum diketahui

H : Hipotesis data X adalah suatu kelas spesifik

$P(H|X)$: Probabilitas hipotesis data H berdasarkan kondisi X (posteriori probability)

$P(H)$: Probabilitas hipotesis H (prior probability)

$P(X|H)$: Probabilitas X berdasarkan kondisi pada hipotesis H

$P(X)$: Probabilitas X

Untuk menjelaskan teorema *naïve bayes*, perlu diketahui bahwa proses klasifikasi perlu sejumlah petunjuk untuk menentukan kelas yang cocok bagi sampel yang dianalisis. Maka dari itu, teorema *naïve bayes* di atas disesuaikan seperti berikut :

$$P(C|F_1 \dots F_n) = \frac{P(C)P(F_1 \dots F_n|C)}{P(F_1 \dots F_n)}$$

Diketahui variable C merepresentasikan kelas, sementara variabel $F_1 \dots F_n$ merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi. Maka dari itu, rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas C (Posterior) adalah peluang.

munculnya kelas C (sebelum masuknya sampel, biasanya disebut prior), dikali dengan peluang munculnya karakteristik sampel pada kelas C (biasanya disebut *likelihood*), dibagi dengan peluang munculnya karakteristik sampel secara global (biasanya disebut *evidence*). Maka rumus diatas bisa ditulis dengan sederhana yaitu sebagai berikut:

$$Posterior = \frac{prior \times likelihood}{Evidence}$$

Nilai *Evidence* akan tetap untuk setiap kelas pada satu sampel. Nilai dari posterior tersebut akan dibandingkan dengan nilai-nilai posterior kelas lainnya untuk dapat menentukan ke kelas suatu sampel yang akan diklasifikasikan. Penjelasan lebih lanjut dalam rumus *Bayes* akan dilakukan dengan menjabarkan $(C|F_1 \dots F_n)$ menggunakan aturan perkalian sebagai berikut:

$$\begin{aligned} (C|F_1 \dots F_n) &= (C)(F_1 \dots F_n|C) \\ &= (C)(F_1|C)P(C)P(F_2 \dots F_n|C, F_1) \\ &= P(C)P(F_1|C)P(C)P(F_2|C, F_1)P(F_3, \dots, F_n|C, F_1, F_2) \\ &= (C)(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1, F_2), P(F_4, \dots, F_n|C, F_1, F_2, F_3) \\ &= (C)(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1, F_2) \dots P(F_n|C, F_1, F_2, F_3, \dots, F_{n-1}) \end{aligned}$$

Setelah itu ada alur dari metode *Naïve Bayes* yaitu sebagai berikut:

1. Baca data training
2. Hitung jumlah dan probabilitas, jika data numeric maka harus dilakukan:
 - a. Cari nilai mean dan standar deviasi dari masing-masing parameter yang data numeric
 - b. Cari nilai probabilitas dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut
3. Menghasilkan nilai dalam tabel mean, standar deviasi, dan probabilitas
4. Solusi dapat dihasilkan

2.5 Bidik Misi

Bidikmisi merupakan bantuan pendidikan yang diberikan pemerintah bagi mereka yang memiliki potensi tinggi namun tidak memiliki finansial yang mencukupi untuk melanjutkan ke jenjang perguruan tinggi.

Berbagai jenis beasiswa dan atau bantuan biaya pendidikan baik oleh pemerintah pusat, pemerintah daerah maupun dari dunia usaha atau industri sebenarnya sudah banyak diluncurkan. Akan tetapi bantuan yang diberikan relatif belum dapat memenuhi kebutuhan studi, jumlah sasaran dan belum menjamin keberlangsungan studi mahasiswa hingga selesai.

Sehingga pemerintah melalui Direktorat Jenderal Pembelajaran dan Kemahasiswaan, Kementerian Riset Teknologi dan Pendidikan Tinggi mulai tahun 2010 meluncurkan Program Bantuan Biaya Pendidikan Bidikmisi yaitu bantuan biaya pendidikan bagi calon mahasiswa tidak mampu secara ekonomi dan memiliki potensi akademik baik untuk menempuh pendidikan di perguruan tinggi pada program studi unggulan sampai lulus tepat waktu.

Adapun persyaratan bagi para calon pendaftar beasiswa bidikmisi :

1. Siswa SMA/SMK/MA/MAK atau sederajat yang akan lulus pada tahun 2018.
2. Lulusan tahun 2018 yang bukan penerima Bidikmisi dan tidak bertentangan dengan ketentuan penerimaan mahasiswa baru di masing-masing perguruan tinggi.
3. Usia paling tinggi pada saat mendaftar adalah 21 tahun.
4. Tidak mampu secara ekonomi dengan kriteria:
 - a. Siswa penerima Beasiswa Siswa Miskin (BSM)
 - b. Pemegang Kartu Pengaman Sosial (KPS) atau sejenisnya.
 - c. Pendapatan kotor gabungan orangtua/wali (suami istri) sebesar-besarnya Rp3.000.000,00 per-bulan. Untuk pekerjaan nonformal/informal pendapatan yang dimaksud adalah rata-rata penghasilan per bulan dalam satu tahun terakhir.
 - d. Pendapatan kotor gabungan orangtua/wali dibagi jumlah anggota keluarga sebesar-besarnya Rp750.000,00 setiap bulannya.
5. Pendidikan orang tua/wali setinggi-tingginya S1 (Strata 1) atau Diploma 4.
6. Berpotensi akademik baik berdasarkan rekomendasi kepala sekolah.

2.5.1 Tujuan Bidik Misi

Tujuan diselenggarakannya Beasiswa Bidikmisi adalah untuk meningkatkan akses dan kesempatan belajar di perguruan tinggi bagi peserta didik yang tidak mampu secara ekonomi dan berpotensi akademik baik, memberi bantuan biaya pendidikan kepada calon/mahasiswa yang memenuhi kriteria untuk menempuh pendidikan program Diploma atau Sarjana sampai selesai

dan tepat waktu, meningkatkan prestasi mahasiswa. Beberapa tujuan beasiswa bidik misi adalah sebagai berikut :

1. Meningkatkan akses dan kesempatan belajar di perguruan tinggi bagi peserta didik yang tidak mampu secara ekonomi dan berpotensi akademik baik;
2. Memberi bantuan biaya pendidikan kepada calon/mahasiswa yang memenuhi kriteria untuk menempuh pendidikan program Diploma/Sarjana sampai selesai dan tepat waktu;
3. Meningkatkan prestasi mahasiswa, baik pada bidang kurikuler, ko-kurikuler maupun ekstra kurikuler;
4. Menimbulkan dampak iring bagi mahasiswa dan calon mahasiswa lain untuk selalu meningkatkan prestasi dan kompetif;
5. Melahirkan lulusan yang mandiri, produktif dan memiliki kepedulian sosial, sehingga mampu berperan dalam upaya pemutusan mata rantai kemiskinan dan pemberdayaan masyarakat.

2.5.2 Sasaran Bidik Misi

Sasaran program adalah lulusan satuan pendidikan SMA/SMK/MA atau yang sederajat yang akan lulus pada tahun tersebut atau sudah lulus satu tahun sebelumnya yang memiliki potensi akademik baik dan tidak mampu secara ekonomi.

2.6 Perangkat Lunak Pengembangan Sistem

Komputer membutuhkan software untuk beroperasi dan membutuhkan sistem operasi atau program-program untuk membuat komponen-komponen komputer bekerja secara baik. Software juga sering digunakan untuk menunjukkan semua program yang dapat dipakai dalam sistem komputer. Dalam pengertian yang sempit, istilah ini menunjuk pada sebuah program yang dapat mempermudah pemakai dari berbagai jenis komputer untuk mendayagunakan hardware dengan baik. Untuk merancang dan membangun aplikasi ini pembuat membutuh kan software penunjang untuk memaksimalkanya yakni sebagai berikut .

2.6.1 Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu - Integrated Development Environment (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA . Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas saat membuat aplikasi Android, misalnya:

1. Sistem versi berbasis Gradle yang fleksibel
2. Emulator yang cepat dan kaya fitur
3. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android
4. Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
5. Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh
6. Alat pengujian dan kerangka kerja yang ekstensif
7. Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain
8. Dukungan C++ dan NDK
9. Dukungan bawaan untuk Google Cloud Platform, mempermudah pengintegrasian Google Cloud Messaging dan App Engine.

2.6.2 Struktur Proyek

Setiap proyek di Android Studio berisi satu atau lebih modul dengan file kode sumber dan file sumber daya. Jenis modul meliputi:

1. Modul aplikasi Android
2. Modul perpustakaan
3. Modul Google App Engine Secara default,

Android Studio menampilkan file proyek dalam tampilan proyek Android. Tampilan ini disusun oleh modul untuk menyediakan akses cepat ke file sumber utama proyek. Semua file build terlihat di tingkat atas di bawah Gradle Scripts dan setiap modul aplikasi berisi folder berikut:

1. bermanifestasi: Berisi file AndroidManifest.xml.

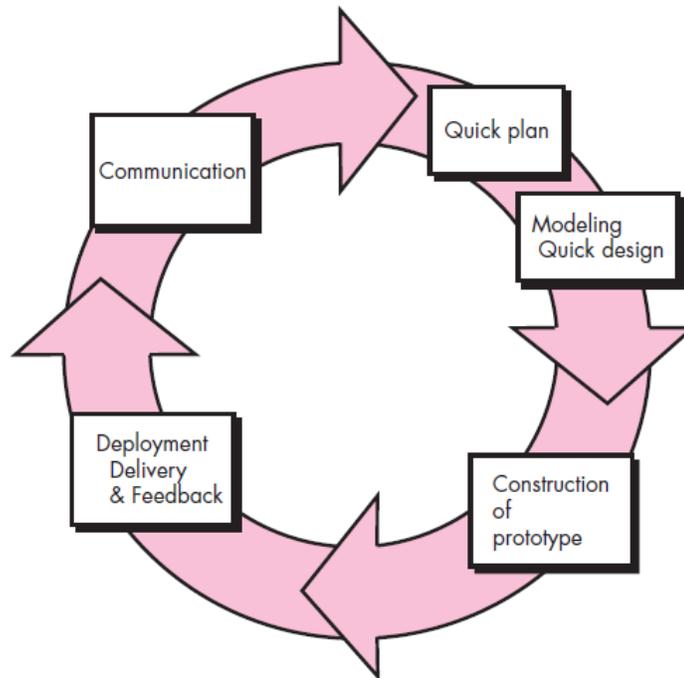
2. `java`: Berisi file kode sumber Java, termasuk kode uji JUnit.
3. `res`: Berisi semua sumber daya non-kode, seperti tata letak XML, string UI, dan gambar bitmap.

Struktur proyek Android pada disk berbeda dari representasi yang rata ini. Untuk melihat struktur file proyek yang sebenarnya, pilih Project from the Project dropdown.

Kemudian dapat menyesuaikan tampilan file proyek agar fokus pada aspek spesifik pengembangan aplikasi. Misalnya, memilih tampilan Masalah dari proyek yang menampilkan tautan ke file sumber yang berisi kesalahan pengkodean dan sintaks yang dikenali, seperti tag penutup elemen XML yang hilang dalam file tata letak.

2.7 Metode Pengembangan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan dalam sistem penunjang keputusan pemilihan ini dengan menggunakan model *prototype*. Menurut Pressman (2012) dalam melakukan perancangan sistem yang akan dikembangkan dapat menggunakan metode *prototype*. Metode ini cocok untuk mengembangkan sebuah perangkat yang akan dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna, kemudian membuat sebuah rancangan kilat yang selanjutnya akan dievaluasi kembali sebelum diproduksi secara benar. *Prototype* bukanlah merupakan sesuatu yang lengkap, tetapi sesuatu yang harus dievaluasi dan dimodifikasi kembali untuk lebih memahami kebutuhan pengguna. Model *prototype* dapat dilihat pada gambar 2.9 berikut. (Najih Munawar, 2012)



Gambar 2.1 Model Prototype (Sumber : Najih Munawar, 2012)

Penjelasan dari gambar diatas adalah sebagai berikut :

- a. Tahap pertama adalah *communication* dan pengumpulan data awal yaitu analisis terhadap kebutuhan pengguna. Tahap suatu perencanaan yang di lakukan, mulai dari menciptakan dan melaksanakan proses untuk memastikan bahwa perencanaan tersebut berkualitas tinggi, terpercaya, efisiensi biaya.
- b. Tahap kedua adalah *quick plan* yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali.
- c. Tahap ketiga adalah *modelling quick design* yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali.
- d. Tahap keempat adalah *construction of prototype* (pembentukan *prototype*) adalah pembuatan perangkat *prototype* termasuk pengujian dan penyempurnaan.
- e. Tahap keenam adalah *deployment, delivery, and feedback* serta produksi akhir adalah tahap penyerahan sistem ke pengguna dan umpan balik serta memproduksi perangkat secara benar sehingga dapat digunakan oleh pengguna.

2.8 Pengujian Black-Box Testing

Metode pengujian pada aplikasi ini menggunakan Black Box Testing yaitu yang berfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input

dan melakukan pengetesan pada spesifikasi fungsional program.

Black Box Testing bukanlah solusi alternatif dari White Box Testing tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh White Box Testing (Pressman 2012).

Black Box Testing cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (interface errors).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (performance errors).

2.9 XAMPP

Xampp merupakan salah satu *software web server* yang dapat digunakan untuk mengatasi masalah PHP dan MySQL jika dirasa sulit. PHP, MySQL, Apache terhubung dalam satu *software* yaitu XAMPP. Penginstalan XAMPP tidak perlu melakukan perubahan pada *register windows* dan tidak perlu mengedit *file* konfigurasinya. Menginstall PHP, MySQL, ataupun Apache. XAMPP merupakan kependekan dari X yaitu *multi platform*, bisa windows, linux, Mac OS, maupun Salaris, A yaitu Apache HTTP Server, M yaitu MySQL, *Database Server*, P yaitu PHP *Scripting language*, dan P selanjutnya yaitu *Perl Scripting Language*, dengan adanya XAMPP mempermudah *user* untuk membuat *server local* sendiri (XAMPP, 2008). (Marsindi, 2013)

2.10 Unified Modeling Language (UML)

Menurut (Rosa A.S dan Salahudin, 2015), *unified modeling language (UML)* adalah salah satu *standar* bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat aplikasi dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

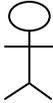
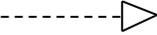
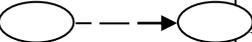
Dengan demikian, penulis dapat mengutarakan bahwa metode *UML (Unified Modeling Language)* merupakan sebuah metode atau sebuah bahasa yang digunakan dalam menterjemahkan, menjelaskan, memodelkan, mendefinisikan suatu sistem dengan bentuk simbol-simbol tertentu yang bertujuan untuk memberikan penjelasan-penjelasan detail dari sebuah sistem.

2.10.1 Use Case Diagram

Menurut (Rosa dan Salahudin,2015), menguraikan *use case* diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Use case digunakan untuk mengetahui fungsi yang ada di dalam sebuah sistem informasi dan aktor yang berhak menggunakan fungsi-fungsi itu.

Tabel 2.1 Simbol *Use Case Diagram*

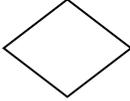
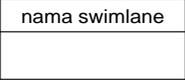
Simbol	Keterangan
<i>Use Case</i> 	Menggambarkan bagaimana seseorang akan menggunakan atau memanfaatkan sistem.
Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri.
Asosiasi 	Komunikasi antara <i>use case</i> dan aktor yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
Generalisasi 	Sebagai penghubung antara aktor- <i>use case</i> atau <i>use case-use case</i> .
<<Include>> 	<i>Include Relationship</i> (relasi cakupan) : Memungkinkan suatu <i>use case</i> untuk menggunakan fungsionalitas yang disediakan oleh
<<Extend>> 	<i>Extend Relationship</i> : Memungkinkan relasi <i>use case</i> memiliki kemungkinan untuk memperluas fungsionalitas

2.10.2 Activity Diagram

Activity Diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Tabel 2.2 Simbol *Activity Diagram*

Simbol	Keterangan
--------	------------

Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan 	Asosiasi percabangan dimana ada pilihan aktivitas lebih dari satu.
Penggabungan 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas.

2.10.3 Sequence Diagram

Sequence diagram menggambarkan kolaborasi dinamis untuk menunjukkan rangkaian pesan yang dikirim antar objek juga interaksi antar objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Dalam sequence diagram terdapat 2 simbol yaitu:

- a. Actor, untuk menggambarkan pengguna sistem.
- b. Lifeline, untuk menggambarkan kelas dan objek.

2.10.4 Class Diagram

Class diagram menggambarkan struktur data dan desripsi class, package, dan objek beserta hubungan satu sama lain. Class diagram berfungsi untuk menjelaskan tipe dari objek sistem dan hubungannya dengan objek yang lain. Class memiliki 3 area pokok yaitu nama, atribut dan metode.

