

BAB 2

TINJAUAN PUSTAKA

2.1 Landasan Teori

Landasan teori adalah penjelasan berbagai konsep dasar dan teori-teori yang berkaitan dengan pembangunan aplikasi mobile android untuk pengukuran bobot badan hewan ternak. Landasan teori ini akan menjadi dasar yang kuat dalam penelitian yang akan dilakukan, terkait teori-teori dalam mendukung penelitian ini akan dibahas pada bab ini.

2.2 Aplikasi Mobile

Aplikasi Mobile adalah sebuah aplikasi yang memungkinkan Anda melakukan mobilitas dengan menggunakan perlengkapan seperti PDA, telepon seluler atau Handphone. Dengan menggunakan aplikasi mobile, Anda dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, browsing dan lain sebagainya. Pemanfaatan aplikasi mobile untuk hiburan paling banyak digemari oleh hampir 70% pengguna telepon seluler, karena dengan memanfaatkan adanya fitur game, music player, sampai video player membuat kita menjadi semakin mudah menikmati hiburan kapan saja dan dimanapun.

Dalam pembuatan aplikasi, penulis akan membuat aplikasi dalam perangkat mobile berbasis android, karena lebih optimal yang dapat diatasi dengan kemampuan perangkat mobile dan dapat dijalankan dimana saja, serta memiliki ukuran yang dapat digenggam.

2.3 Hewan Ternak

Hewan Ternak adalah hewan yang dengan sengaja dipelihara sebagai sumber pangan, sumber bahan baku industri, atau sebagai pembantu pekerjaan manusia. Usaha pemeliharaan ternak disebut sebagai peternakan (atau perikanan, untuk kelompok hewan tertentu) dan merupakan bagian dari kegiatan pertanian secara umum. Hewan ternak merupakan hewan yang sengaja dikembangbiakkan untuk kebutuhan konsumsi maupun industri.

Ternak dapat berupa binatang apa pun (termasuk serangga dan vertebrata tingkat rendah seperti ikan dan katak). Namun, dalam percakapan sehari-hari orang biasanya merujuk kepada unggas dan mamalia domestik, seperti ayam, angsa, kalkun, atau itik untuk unggas, serta babi, sapi, kambing, domba, kuda, atau keledai untuk mamalia. Sebagai tambahan, di beberapa daerah di dunia juga dikenal hewan ternak yang khas seperti unta, llama, bison, burung unta, dan tikus belanda mungkin sengaja dipelihara sebagai ternak. Jenis ternak bervariasi di seluruh dunia dan tergantung pada sejumlah faktor seperti iklim, permintaan konsumen, daerah asal, budaya lokal, dan topografi.

Adapun ciri-ciri hewan ternak yaitu :

1. Bisa menghasilkan bahan yang dibutuhkan manusia entah itu untuk keperluan industri atau pangan misalnya kane susu, daging, telur, kulit, dan lainnya.
2. Hewan ternak bukan hewan liar yang hidup di alam liar meskipun ada yang menernakkan hewan liar.
3. Mudah dipelihara dan dikembangbiakkan
4. Memiliki nilai ekonomis

Contoh hewan yang biasa dternak yaitu :

1. Ayam
2. Angsa
3. Kalkun
4. Itik
5. babi
6. Sapi
7. Kambing
8. Domba
9. Kuda
10. Keledai
11. Kerbau

Bobot badan ternak berhubungan dengan pertumbuhan dan karkas yang dihasilkan, sedangkan bobot badan itu sendiri dipengaruhi sifat perdagingan, karkas dan gemuknya hewan, isi perut serta besarnya pertulangan kepala, kaki dan kulit. Umur dan jenis kelamin turut mempengaruhi bobot badan dan ukuran ternak. Bobot badan pada umumnya mempunyai hubungan positif dengan semua ukuran linier tubuh (Kidwell dan Mc Cormick, 1956).

Rumus Pendugaan Bobot Badan Ternak

Menurut Gafar (2007), rumus-rumus yang dapat digunakan untuk menduga bobot badan adalah:

$$\text{Rumus Schoorl (lbs)} = \frac{(\text{LD}_{(\text{cm})} + 22)^2}{100}$$

$$\text{Rumus Winter} = \frac{(\text{LD})_{(\text{inchi})}^2 \times \text{PB}_{(\text{inchi})}}{300} \quad (\text{dalam satuan pound})$$

$$\text{Rumus Smith} = \frac{(\text{LD}_{(\text{cm})} + 18)^2}{100}$$

Keterangan: LD = Lingkar Dada

PB = Panjang Badan

Dalam penelitian ini, penulis akan membatasi pengukuran hewan ternak yaitu pada hewan ruminansia besar (sapi dan kerbau) dan ruminansia kecil (kambing, domba) dan pendugaan bobot badan ternak dengan menggunakan Rumus Winter.

2.4 Sistem

Sistem adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan. Istilah ini sering dipergunakan untuk menggambarkan suatu set entitas yang berinteraksi, di mana suatu model matematika seringkali bisa dibuat.

Menurut sistem adalah suatu kumpulan elemen yang di dalamnya terdapat prosedur yang digunakan dalam rangka mencari suatu tujuan bersama melalui cara pengoperasian barang dan data pada saat tertentu. Cara ini ditujukan untuk mendapatkan data atau informasi yang diinginkan. Dibawah ini adalah karakteristik sistem dan klasifikasi sistem , dalam suatu proses dasar penelitian :

2.4.1 Karakteristik Sistem

Menurut Kusri dan Koniyo (2007:6) sistem mempunyai karakteristik atau sifat-sifat tertentu, diantaranya:

1. Komponen Sistem (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang saling bekerja sama membentuk suatu komponen sistem atau bagian-bagian dari sistem.

2. Batas Sistem (*Boundary*)

Merupakan daerah yang membatasi suatu sistem dengan sistem yang lain atau dengan lingkungan kerjanya.

3. Subsistem

Bagian-bagian dari sistem yang beraktivitas dan berinteraksi satu sama lain untuk mencapai tujuan dengan sasarannya masing-masing.

4. Lingkungan Luar Sistem (*Environment*)

Suatu sistem yang ada diluar dari atas sistem yang dipengaruhi oleh operasi sistem.

5. Penghubung Sistem (*Interface*)

Media penghubung antara suatu subsistem dengan subsistem lain. Adanya penghubung ini memungkinkan berbagai sumber daya mengalir dari suatu subsistem ke subsistem lainnya.

6. Masukan Sistem (*Input*)

Energi yang masuk ke dalam sistem, berupa perawatan dan sinyal. Masukkan perawatan adalah energi yang dimasukkan supaya sistem tersebut dapat berinteraksi.

7. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

8. Pengolahan Sistem (*Proces*)

Suatu sistem yang mempunyai suatu bagian pengolah yang akan mengubah masukan menjadi keluaran.

9. Sasaran Sistem (*Objective*)

Tujuan yang ingin dicapai oleh sistem, akan dikatakan berhasil apabila mengenai sasaran atau tujuan.

2.4.2 Klasifikasi Sistem

Menurut Kusri dan Koniyo (2007:7) mengatakan bahwa suatu sistem dapat diklasifikasikan sistem sebagai berikut [2]:

1. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau gagasan yang tidak tampak secara fisik. Sedangkan sistem fisik adalah sistem yang ada secara fisik.

2. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi karena proses alam. Sedangkan sistem buatan manusia adalah sistem yang dirancangan oleh manusia.

3. Sistem Tertentu dan Sistem Tak Tentu

Sistem tertentu adalah suatu sistem yang operasinya dapat diprediksi secara tepat. Sedangkan sistem tak tertentu adalah sistem dengan perilaku kedepan yang tidak dapat diprediksi.

1. Sistem Tertutup dan Sistem Terbuka

Sistem Tertutup adalah sistem yang tidak berhubungan dengan lingkungan luar atau otomatis. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dapat terpengaruh dengan keadaan lingkungan luarnya.

2.5 Android

Android merupakan sistem operasi yang memang khusus dirancang untuk smartphome dan tablet. Sistem Android ini memiliki basis Linux yang mana dijadikan sebagai pondasi dasar dari sistem operasi Android. Linux sendiri merupakan sistem operasi yang memang khusus dirancang untuk computer [3].

2.5.1 Kelebihan Android diantaranya yaitu :

1. User Friendly, yang dimaksudkan disini adalah sistem Android sangat mudah untuk dijalankan. Sama halnya pada sistem operasi Windows yang ada pada komputer. Bagi yang belum terlalu terbiasa dengan smartphome, kita hanya membutuhkan waktu sebentar saja untuk bisa mempelajari sistem Android [4].
2. Sangat mudah mendapat beragam notifikasi dari smartphome. Untuk mendapatkannya, kita bisa mengatur beberapa akun yang kita miliki seperti SMS, Email, Voice Dial, dan lainnya.
3. Segi tampilan sistem Android yang menarik dan tidak kalah baiknya dengan iOS (Apple). Hal ini dikarenakan dari awal, Android memang mengusung konsep dan teknologi iOS hanya saja Android merupakan versi murah dari iOS.
4. Sistem operasi ini memang memiliki konsep open source yang mana pengguna dapat bebas mengembangkan sistem android versi miliknya sendiri. Sehingga akan banyak sekali Custom ROM yang bisa digunakan.
5. Tersedia beragam pilihan aplikasi menarik, bahkan hingga jutaan aplikasi. Dari mulai aplikasi gratis hingga aplikasi berbayar, dan dapat di download langsung di Google Playstore yang tersedia pada smartphome.

2.5.2 Kekurangan Android diantaranya yaitu:

1. Update system yang kurang efektif. Sistem Android memang seringkali mengalami peningkatan versi yang ditawarkan kepada penggunanya. Namun untuk mengupdate system Android ini bukanlah hal yang mudah. Kita diharuskan untuk menunggu masing-masing vendor merilis resmi

update terbaru dari sistem android tersebut. Dan bagi beberapa pengguna, hal ini kurang begitu menyenangkan [5].

2. Baterai yang cepat habis, mungkin hal ini sering dialami oleh pengguna sistem Android saat ini. Apalagi jika sering menyalakan paket data serta menggunakan widget dan aplikasi yang berjalan terlalu berlebihan sehingga menyebabkan daya baterai berkurang dengan cepat. Untuk mengatasi hal tersebut, kita bisa mengurangi aktivitas yang ada pada smartphone.
3. Sering mengalami lemot atau lag, hal ini biasanya berkaitan dengan spesifikasi dari masing-masing perangkat seluler. Namun meskipun begitu, jika sistem android memang tidak bersahabat dengan aplikasi-aplikasi yang kita miliki tentu saja akan berdampak pada leletnya penggunaan smartphone. Hal ini dikaitkan dengan RAM atau prosesor yang memang kurang memadai. Sehingga akan lebih baik jika menyimpan aplikasi yang memang sesuai dengan smartphone yang dimiliki.

2.5.3 Jenis-jenis versi android :

1. Apple Pie / Alpha v1.0

Untuk OS pertama yang dimiliki oleh Android adalah Apple Pie, OS ini dirilis pada tanggal 23 September 2008.

2. Banana Bread v1.1

Untuk OS Android yang kedua adalah Banana Bread yang dirilis pada bulan Februari 2009 yang digunakan oleh Smartphone pengadopsi Android, HTC dengan merk Dream.

3. CupCake v1.5

Untuk OS yang selanjutnya adalah Cupcake, versi Android CupCake mulai menunjukkan kecanggihannya sebagai Sistem Operasi Android, dan ini kali pertama Google menjadikan nama makanan.

4. Donut v1.6

Versi Donut ini dirilis pada bulan September 2009, di versi ini Google mulai memperbaiki masalah/bug yang membuat Android ini lebih sempurna ditambah lagi dengan fitur navigasi turn-by-turn

5. Eclair v2.0-2.1

Dirilis pada Oktober 2009 di versi 2.0 dengan nama Eclair yang menjadi awal diaplikasikan Bluetooth 2.1 di Android. Setelah dirilis versi 2.0 diupdate lagi ke versi 2.1 yang mendapat fitur tambahan seperti multi-touch, Live Wallpaper dan juga Blitz kamera.

6. Froyo/Forzen Yogurt v2.2

Pada bulan Mei 2010 Paduan antara Google Chrome dengan Android mulai nampak dengan ditambahkan script Chrome dengan upaya untuk meningkatkan kecepatan kinerja.

Forzen Yogurt sendiri yakni memiliki makna Yogurt yang dibekukan, jadi seperti Ice Cream.

7. Gingerbread v2.3

Bulan Desember 2010 seiring dirilisnya versi Gingerbread, Android kedatangan salah satu merk Platform raksasa Smartphone yakni Samsung yang meluncurkan Samsung Nexus.

Di android versi 2.3 ini Google menambahkan fitur baru yaitu NFC, internet Calling, Download Manager

8. Honeycomb v3.0-3.2

Lanjutan dari Huruf berabjad H adalah Honeycomb yang dirilis pada Februari 2014 , untuk android versi ini diperuntukkan dan spesialis Tablet dimana era Tablet sedang menjadi-jadi di tahun ini, Update tak hanya berhenti di versi 3.0 saja namun juga dikembangkan ke versi 3.1 dan 3.2.

9. Ice Cream Sandwich v4.0

Revormasi Android mulai terlihat dengan adanya Ice Cream Sandwich dimana OS versi ini mulai bekerja disemua jenis platform dan smartphone seperti tablet , fiturnya juga menjadi bertambah diantaranya adalah ekstra multitasking dan notifikasi yang lebih banyak.

10. Jelly bean v4.1.2

Android versi Jelly Bean ini dirilis pada 27 Juni 2014 lewat konferensi I/O Google. Jelly Bean menjadi versi Android yang mendapatkan banyak update, tercatat 2 kali sudah mendapatkan update di Jelly Bean yakni versi 4.1.2 dimana berbeda dibandingkan versi sebelumnya adalah segi User Interface yang lebih elegan seta penambahan fitur Google Search.

11. KitKat v4.4.2

Android Versi 4.42 atau bernama KitKat ini paling banyak dipakai pada smartphone masa kini.

12. Lolipop v5.0

Android Lolipop adalah Android versi terbaru yang diluncurkan Google pada tahun 2014. Versi Lollipop pertama kali ditanamkan di Smartphone Google Nexus 6

13. Marsmelow v6.0

Google secara resmi mengeluarkan Android versi 6.0 yang diberi nama yaitu Marshmallow. Selain itu Android Marshmallow juga mendapatkan fitur fitur terbarunya.

14. Nougat v7.0

Nougat adalah versi Android termutakhir yang baru diperkenalkan pada ajang kumpul developer Google I/O, pertengahan 2016 ini. Setelah beberapa lamanya, Google menghadirkan OS Nougat secara resmi untuk publik. Pembaruan yang paling mendasar pada versi Nougat adalah kehadiran Google Assistant yang menggantikan Google Now. Asisten tersebut lebih bisa diandalkan untuk menjalankan berbagai fungsi.

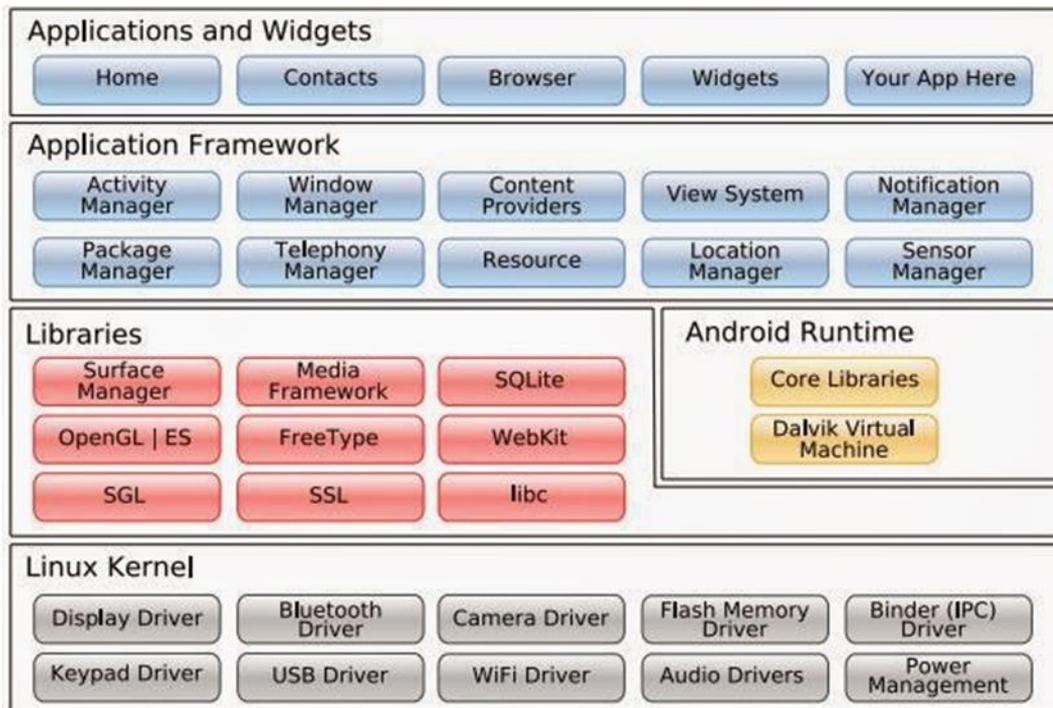
Fitur-fitur baru lainnya mencakup layar split-screen saat dipakai multitasking, serta fitur Doze yang telah dikenalkan sebelum di versi Android Marshmallow namun telah ditingkatkan. Android versi 7.0 atau Nougat juga memiliki dukungan terhadap platform virtual reality terbaru Google.

15. Oreo v8.0

Versi android ini resmi diperkenalkan oleh Google pada tanggal 22 Agustus 2017 yang lalu dan juga sudah secara resmi bisa diluncurkan langsung ke lapangan, tapi sebelum diresmikan versi android ini oleh google, nama "Oreo" tersebut sudah terendus sejak Android O pertama kali diperkenalkan di ajang Google I/O 2017 pada Mei 2017 lalu.

2.6 Arsitektur Android

Android dibangun menggunakan object oriented, dimana elemen-elemen penyusun sistem operasi berupa objek yang dapat digunakan kembali. Arsitektur android terdiri dari beberapa lapisan perangkat lunak atau software, yaitu layer applications and widget, application framework, libraries, android RunTime, dan Linux Kernel. Untuk penjelasan dari setiap lapisan, akan di dijelaskan dan digambarkan sebagai berikut :



Gambar 2.1 Arsitektur Android

1. Applications dan Widgets

Applications dan Widgets ini adalah layer di mana kita berhubungan dengan aplikasi saja, di mana biasanya kita download aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Di Layer terdapat aplikasi inti termasuk klien email, program SMS, kalender, peta, browser, kontak, dan lain-lain. Semua aplikasi ditulis menggunakan bahasa pemrograman Java.

2. Applications Frameworks

Android adalah "Open Development Platform" yaitu Android menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi resources, menjalankan service background, mengatur alarm, dan menambahkan status notifications, dan sebagainya. Pengembang memiliki akses penuh menuju API framework seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (reuse).

Sehingga bisa kita simpulkan Applications Framework ini adalah layer di mana para pembuat aplikasi melakukan pengembangan/pembuatan aplikasi yang akan dijalankan di sistem operasi android, karena pada layer inilah aplikasi dapat dirancang dan dibuat, seperti content-providers yang berupa sms dan panggilan telepon.

Komponen-komponen yang termasuk di dalam Applications Frameworks adalah sebagai berikut :

- a) Views
- b) Content Providers
- c) Resource Manager
- d) Notification Manager
- e) Activity Manager

3. Libraries

Libraries ini adalah layer di mana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses libraries untuk menjalankan aplikasi. Berjalan di atas kernel, Layer ini meliputi berbagai library C/C++ inti seperti Libc dan SSL, serta :

- a) Libraries media untuk pemutaran media audio dan video
- b) Libraries untuk manajemen tampilan
- c) Libraries Graphics mencakup SGL dan OpenGL untuk grafis 2D dan 3D
- d) Libraries SWLite untuk dukungan database

- e) Libraries SSL dan WebKit terintegrasi dengan web browser dan security
- f) Libraries liveWebcore mencakup modern web browser dengan engine embeded web view
- g) Libraires 3D yang mencakup implementasi OpenGL S 1.0 API's

4. Android Run Time

Layer yang membuat aplikasi Android dapat dijalankan di mana dalam prosesnya menggunakan implementasi Linux. Dalvik Virtual Machine (DVM) merupakan mesin yang membentuk dasar kerangka aplikasi Android. Di dalam Android Run Time Time dibagi menjadi dua bagian yaitu :

- a) Core libraries : Aplikasi Android dibangun dalam bahasa java, sementara Dalvik sebagai virtual mesinnya bukan Virtual Machine Java, sehingga diperlukan sebuah libraries yang berfungsi untuk menterjemahkan bahasa java/C yang ditangani oleh Core Libraries
- b) Dalvik Virtual Machine : Virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, di mana merupakan pengembangan yang mampu membuat linux kernel untuk melakukan threading dan manajemen tingkat rendah.

5. Linux Kernel

Linux Kernel adalah layer di mana inti dari operating sistem dari Android itu berada. Berisi file-file system yang mengatur sistem processing, memory, resource, driver, dan sistem-sistem operasi android lainnya. Linux kernel yang digunakan android adalah linux kernel relase 2.6. jika ingin mempelajari linux kernel ini, dapat mengunduh linux kernel secara gratis di situs www.kernel.org.

Alasan menggunakan Arsitektur Android pada penelitian ini karena android adalah sebuah sistem operasi yang dapat dijalankan di perangkat bergerak (mobile device) dan paling banyak digunakan dan android sangat support untuk fitur-fitur yang akan disediakan nantinya.

2.7 Android SDK

Android-SDK merupakan tools bagi para programmer yang ingin mengembangkan aplikasi berbasis google android. Android SDK mencakup

seperangkat alat pengembangan yang komprehensif. Android SDK terdiri dari debugger, libraries, handset emulator, dokumentasi, contoh kode, dan tutorial. Saat ini Android sudah mendukung arsitektur x86 pada Linux (distribusi Linux apapun untuk desktop modern), Mac OS X 10.4.8 atau lebih, Windows XP atau Vista. Persyaratan mencakup JDK, Apache Ant dan Python 2.2 atau yang lebih baru. IDE yang didukung secara resmi adalah Eclipse 3.2 atau lebih dengan menggunakan plugin Android Development Tools (ADT), dengan ini pengembang dapat menggunakan teks editor untuk mengedit file Java dan XML serta menggunakan peralatan command line untuk menciptakan, membangun, melakukan debug aplikasi Android dan pengendalian perangkat Android (misalnya, reboot, menginstal paket perangkat lunak dengan jarak jauh).

Alasan menggunakan Android SDK pada penelitian ini karena terdapat menjalankan program menjadi lebih cepat yang disebut *instant run*, mampu membuat aplikasi untuk semua perangkat android memiliki fitur SDK yang lengkap sehingga waktu pembuatan program menjadi lebih produktif.

2.8 Android Studio

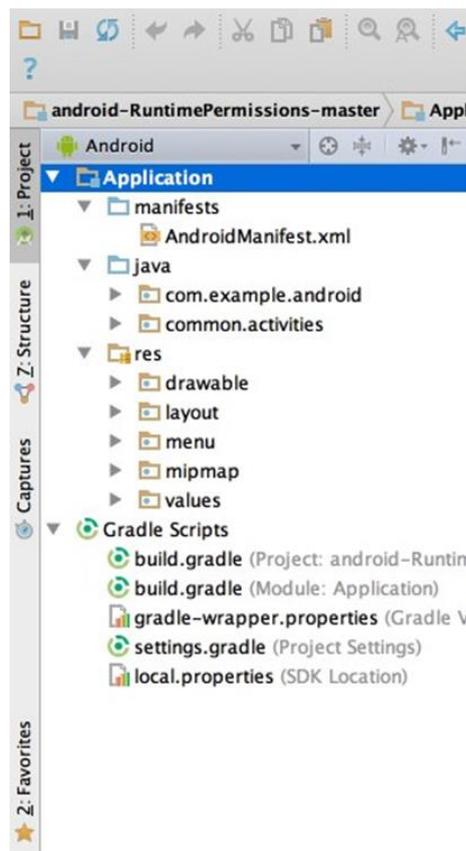
Pada awalnya mulanya, Android Inc merupakan sebuah perusahaan software kecil yang didirikan pada bulan Oktober 2003 di Palo Alto, California, USA. Didirikan oleh beberapa senior di beberapa perusahaan yang berbasis IT & Communication, Andy Rubin, Rich Miner, Nick Sears dan Chris White. Menurut Rubin, Android Inc Didirikan untuk mewujudkan mobile device yang lebih peka terhadap lokasi dan preferensi pemilik. Dengan kata lain, Android Inc ingin mewujudkan mobile device yang lebih mengerti pemiliknya.

Konsep yang dimiliki Android Inc ternyata menggugah minat Google untuk memilikinya. Pada bulan Agustus 2005, Akhirnya Android Inc diakuisisi oleh Google Inc. seluruh sahamnya dibeli oleh Google. Banyak yang memperkirakan nilai pembelian Android Inc Oleh Google adalah sebesar USD 50 juta. saat itu banyak yang berspekulasi bahwa akuisisi ini adalah langkah awal yang dilakukan Google untuk masuk ke pasar mobile phone.

Andy Rubin, Rich Miner, Nick Sears dan Chris White tetap di Android Inc yang dibeli Google, sehingga akhirnya mereka semua menjadi bagian dari raksasa Google dan sejarah Android. Saat itulah mereka mulai menggunakan platform Linux untuk membuat sistem operasi bagi mobile phone.

Maka dari situlah banyak pengembang sistem maupun software berlomba-lomba untuk membuat atau merancang sistem Android menggunakan software – software yang support dengan Android, dan sebagai contoh disini kita pengenalan Android Studio.

Laman ini berisi pengantar dasar fitur-fitur Android Studio [6]. Untuk memperoleh rangkuman perubahan terbaru, lihat catatan rilis Android Studio



Gambar 2.2 Android Studio

2.8 API

API adalah singkatan dari Application Programming Interface, dan memungkinkan developer untuk mengintegrasikan dua bagian dari aplikasi atau dengan aplikasi yang berbeda secara bersamaan. API terdiri dari berbagai elemen seperti function, protocols, dan tools lainnya yang memungkinkan developers untuk membuat aplikasi. Tujuan penggunaan API adalah untuk mempercepat proses development dengan menyediakan function secara terpisah sehingga developer tidak perlu membuat fitur yang serupa. Penerapan API akan sangat terasa jika fitur yang diinginkan sudah sangat kompleks, tentu membutuhkan waktu untuk membuat yang serupa dengannya. Misalnya: integrasi dengan payment gateway. Terdapat berbagai jenis sistem API yang dapat digunakan, termasuk sistem operasi, library, dan web.

API yang bekerja pada tingkat sistem operasi membantu aplikasi berkomunikasi dengan layer dasar dan satu sama lain mengikuti serangkaian protokol dan spesifikasi. Contoh yang dapat menggambarkan spesifikasi tersebut adalah POSIX (Portable Operating System Interface). Dengan menggunakan standar POSIX, aplikasi yang di-compile untuk bekerja pada sistem operasi tertentu juga dapat bekerja pada sistem lain yang memiliki kriteria yang sama. Software library juga memiliki peran penting dalam menciptakan compatibility antar sistem yang berbeda.

Aplikasi yang berinteraksi dengan library harus mengikuti serangkaian aturan yang ditentukan oleh API. Pendekatan ini memudahkan software developer untuk membuat aplikasi yang berkomunikasi dengan berbagai library tanpa harus memikirkan kembali strategi yang digunakan selama semua library mengikut API yang sama. Kelebihan lain dari metode ini menunjukkan betapa mudahnya menggunakan library yang sama dengan bahasa pemrograman yang berbeda.

2.9 ARCORE

ARCore merupakan pengembangan dari Project Tango, sebagaimana Google ingin agar pengalaman Augmented Reality ini bisa dirasakan oleh banyak orang para pengguna ponsel Android maka diciptakanlah ARCore ini oleh Google. Keduanya memang memberikan pengalaman dalam hal AR, namun pada Project Tango diperlukan spesifikasi hardware yang lebih, dibutuhkan dua kamera untuk menjalankan aplikasi AR seperti yang ada pada ponsel Asus ZenFone AR [7].

Di Project Tango, pengguna perlu berinvestasi membeli ponsel dengan spesifikasi semacam Asus dan dari manufaktur pembuat ponsel Android pun perlu membuat ponsel berspesifikasi serupa. Berbeda dengan ARCore, siapapun pengguna Android bisa menikmati pengalaman AR dengan semua ponsel asal sudah memiliki versi Android Nougat atau di atasnya. Saat ini Tango memberikan pengalaman AR yang lebih real karena dukungan hardware dibandingkan dengan ARCore. Walau begitu, seiring berjalannya waktu, Google akan meminta dan menekan para pembuat ponsel untuk meningkatkan kemampuan kamera maupun spesifikasi lainnya untuk mendukung perkembangan ARCore.

ARCore menggunakan tiga kemampuan dalam mengintegrasikan konten virtual ke dalam dunia nyata seperti yang nantinya akan terlihat dalam kamera ponsel. Tiga kemampuan tersebut terdiri dari Motion Tracking (memungkinkan ponsel paham dan melacak posisi relatifnya terhadap dunia nyata), Environmental understanding (memungkinkan ponsel untuk mendeteksi ukuran dan lokasi semua tipe permukaan, horisontal, vertikal dan sudut), dan Light Estimation (memungkinkan ponsel mengestimasi kondisi pencahayaan ruangan).

Seperti dijelaskan pada halaman developer Google, secara fundamental ARCore melakukan dua hal yaitu melacak posisi ponsel saat bergerak dan membangun pemahaman sendiri terhadap kondisi real atau dunia nyata. Teknologi pelacakan gerak ini menggunakan kamera ponsel untuk mengidentifikasi titik-titik yang menarik, fitur-fitur yang disediakan, dan melacak pergerakan titik-titik tersebut setiap waktu. Kombinasi antara pergerakan titik-titik dan pembacaan

melalui sensor inersia ponsel, ARCore mampu menentukan baik posisi maupun orientasi ponsel atas pergerakannya terhadap ruang.

ARCore mampu mendeteksi permukaan datar seperti meja maupun lantai, mampu pula mengestimasi pencahayaan ruang. Kapabilitas ini bergabung untuk memampukan ARCore membangun pemahaman tersendiri tentang kondisi real di sekitarnya.

Dengan kemampuannya, pengguna bisa meletakkan obyek apapun di dalamnya, anotasi atau catatan maupun informasi lain yang terintegrasi secara sempurna dengan dunia nyata. Google memberikan contoh, dengan meletakkan anak kucing di bagian pojok meja kopi, atau memberikan anotasi pada sebuah lukisan mengenai informasi biografi artis yang ada dalam lukisan tersebut. Motion tracking atau pelacakan gerak mengartikan bahwa kita dapat bergerak dan melihat objek dari sudut tertentu, dan bahkan apabila kita berputar balik dan meninggalkan ruangan, ketika kita kembali, maka anak kucing ataupun anotasi akan tetap berada pada posisi seperti sesaat kita meninggalkannya .

Adapun hubungan ARCORE pada penelitian ini karena aplikasi yang dibuat sangat membutuhkan fitur ini agar pengguna bisa mengukur panjang dan lingkar dada hewan ternak dengan memanfaatkan augmented reality yang di sediakan oleh ARCORE.

2.10 Java

Java merupakan bahasa pemrograman berorientasi objek yang dikembangkan oleh Sun Microsystems, suatu perusahaan yang terkenal dengan Workstation UNIX high-end.

Sejak dirilis pada tahun 1995, bahasa pemrograman Java dengan cepat memperoleh popularitas di kalangan para pemrogram. Keberhasilan ini disebabkan teknologi baru yang diperkenalkan Sun Microsystems yaitu Java Virtual Machine (JVM), yang memungkinkan sebuah aplikasi dijalankan di atas platform apa saja sepanjang pada mesin tersebut dipasang JVM [11].

Program yang dihasilkan dengan bahasa Java dapat berupa applet (aplikasi kecil yang berjalan di atas web browser) maupun berupa aplikasi mandiri yang

dijalankan dengan program Java Interpreter. Contoh program yang ditulis dengan bahasa Java adalah HotJava yang berupa sebuah web browser.

2.10.1 Karakteristik Java

Java adalah sebuah bahasa pemrograman berorientasi obyek murni. Jadi program – program Java berada dalam sebuah struktur kelas – kelas dan obyek – obyek. Pada dasarnya sintaks pada bahasa Java mirip dengan sintaks pada bahasa C atau C++. Java bertipe kuat (*strongly-typed*). Ini berarti semua tipe data terikat secara statis atau dengan kata lain setiap nama variabel diasosiasikan dengan sebuah tipe data tunggal yang dikenali pada saat kompilasi.

2.10.2 Kelebihan Java

1. Multiplatform

Kelebihan utama dari Java ialah dapat dijalankan di beberapa platform / sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. Dengan kelebihan ini pemrogram cukup menulis sebuah program Java dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin /bytecode) sekali lalu hasilnya dapat dijalankan di atas beberapa platform tanpa perubahan. Kelebihan ini memungkinkan sebuah program berbasis java dikerjakan diatas operating system Linux tetapi dijalankan dengan baik di atas Microsoft Windows. Platform yang didukung sampai saat ini adalah Microsoft Windows, Linux, Mac OS dan Sun Solaris. Penyebabnya adalah setiap sistem operasi menggunakan programnya sendiri-sendiri (yang dapat diunduh dari situs Java) untuk meninterpretasikan bytecode tersebut.

2. OOP (Object Oriented Programming - Pemrogram Berorientasi Objek)

yang artinya semua aspek yang terdapat di Java adalah Objek. Java merupakan salah satu bahasa pemrograman berbasis oebjek secara murni. Semua tipe data diturunkan dari kelas dasar yang disebut Object. Hal ini sangat memudahkan pemrogram untuk mendesain, membuat, mengembangkan dan mengalokasi kesalahan sebuah program dengan basis Java secara cepat, tepat, mudah dan terorganisir. Kelebihan ini menjadikan

Java sebagai salah satu bahasa pemrograman termudah, bahkan untuk fungsi fungsi yang advance seperti komunikasi antara komputer sekalipun.

3. Perpustakaan Kelas Yang Lengkap, Java terkenal dengan kelengkapan library/perpustakaan(kumpulan program program yang disertakan dalam pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.
4. Bergaya C++, memiliki sintaks seperti bahasa pemrograman [C++] sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java. Universitas-universitas di Amerika juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.
5. Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung (seperti halnya dalam bahasa C++ yang dipakai secara luas).

2.10.3 Kekurangan Dari Java

1. Tulis sekali, perbaiki di mana saja - Masih ada beberapa hal yang tidak kompatibel antara platform satu dengan platform lain. Untuk J2SE, misalnya SWT-AWT bridge yang sampai sekarang tidak berfungsi pada Mac OS X.
2. Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan bytecode yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga

terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/direverse-engineer.

3. Penggunaan memori yang banyak. Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ dan Pascal (lebih spesifik lagi, Delphi dan Object Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berkutat dengan mesin komputer berumur lebih dari 4 tahun.

Hubungan Java pada penelitian ini, karena aplikasi yang dibuat menggunakan perangkat lunak Android Studio dengan bahasa pemrograman yang digunakan adalah Java. Dengan menggunakan Java pada android studio karena terdapat fitur yang dapat ditemukan pada saat pembangunan aplikasi android.

2.11 OOP

OOP (Object Oriented Programming) adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Jadi setiap bagian dari suatu permasalahan adalah objek, nah objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saya ambil contoh Pesawat, Pesawat adalah sebuah objek. Pesawat itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, baling-baling, kursi, dll. Pesawat sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi, berkomunikasi dan saling mengirim pesan kepada objek-objek yang lainnya. Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling berkiriman pesan kepada objek yang lain [12].

2.11.1 Konsep OOP

1. Abstrak Class

- a) Kelas merupakan deskripsi abstrak informasi dan tingkah laku dari sekumpulan data.
- b) Kelas dapat diilustrasikan sebagai suatu cetak biru(blueprint) atau prototipe yang digunakan untuk menciptakan objek.
- c) Kelas merupakan tipe data bagi objek yang mengenkapsulasi data dan operasi pada data dalam suatu unit tunggal.
- d) Kelas mendefinisikan suatu struktur yang terdiri atas data kelas (data field), prosedur atau fungsi (method), dan sifat kelas (property).

2. Encapsulation

- a) Istilah enkapsulasi sebenarnya adalah kombinasi data dan fungsionalitas dalam sebuah unit tunggal sebagai bentuk untuk menyembunyikan detail informasi.
- b) Proses enkapsulasi memudahkan kita untuk menggunakan sebuah objek dari suatu kelas karena kita tidak perlu mengetahui segala hal secara rinci.
- c) Enkapsulasi menekankan pada antarmuka suatu kelas, atau dengan kata lain bagaimana menggunakan objek kelas tertentu.
Contoh: kelas mobil menyediakan antarmuka fungsi untuk menjalankan mobil tersebut, tanpa kita perlu tahu komposisi bahan bakar, udara dan kalor yang diperlukan untuk proses tersebut.

3. Inheritance

- a) Kita dapat mendefinisikan suatu kelas baru dengan mewarisi sifat dari kelas lain yang sudah ada.
- b) Penurunan sifat ini bisa dilakukan secara bertingkattingkat, sehingga semakin ke bawah kelas tersebut menjadi semakin spesifik.
- c) Sub kelas memungkinkan kita untuk melakukan spesifikasi detail dan perilaku khusus dari kelas supernya.

- d) Dengan konsep pewarisan, seorang programmer dapat menggunakan kode yang telah ditulisnya pada kelas super berulang kali pada kelas-kelas turunannya tanpa harus menulis ulang semua kodekode itu.

4. Polymorphism

- a) Polimorfisme merupakan kemampuan objekobjek yang berbeda kelas namun terkait dalam pewarisan untuk merespon secara berbeda terhadap suatu pesan yang sama.
- b) Polimorfisme juga dapat dikatakan kemampuan sebuah objek untuk memutuskan method mana yang akan diterapkan padanya, tergantung letak objek tersebut pada jenjang pewarisan.
- c) Method overriding.
- d) Method name overloading.

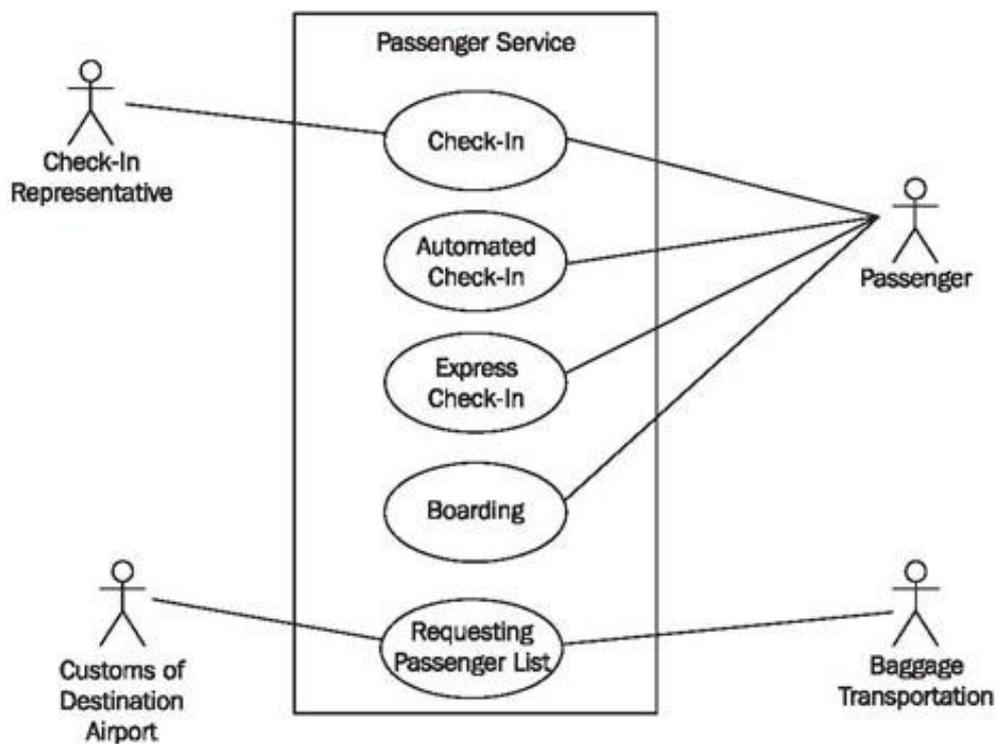
2.12 UML (Unified Modelling Language)

UML adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML merupakan singkatan dari Unified Modeling Language. UML juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena developer harus melakukan penelusuran dan mempelajari kode program. UML juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu developer ke developer lainnya. Tidak hanya antar developer terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya UML [13].

Terdapat beberapa diagram-diagram UML yang sering digunakan dalam pemodelan sebuah sistem berorientasi objek diantaranya:

2.12.1 Use Case Diagram

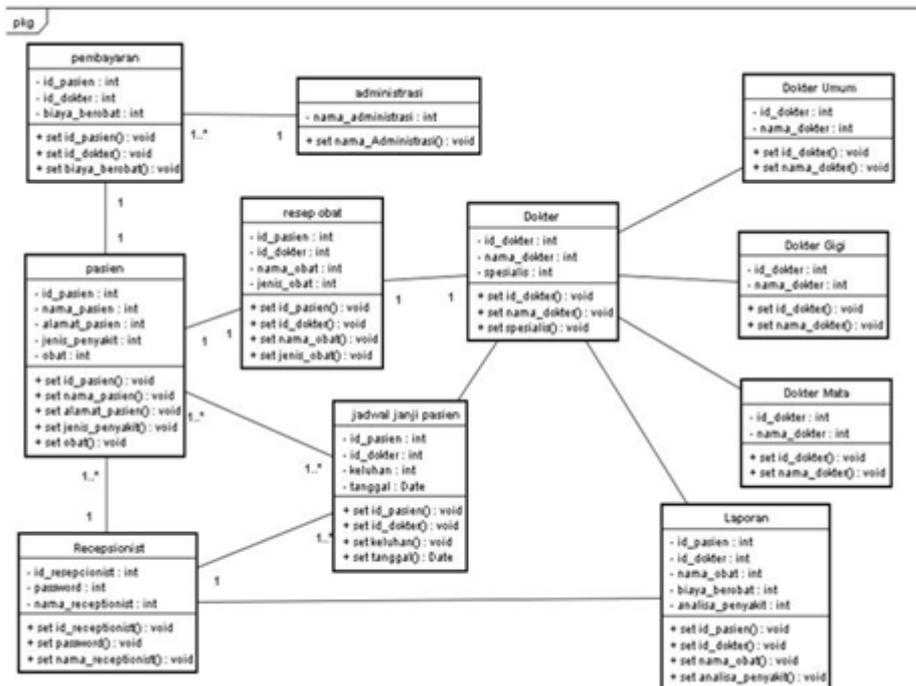
Use case diagram yaitu salah satu jenis diagram pada UML yang menggambarkan interaksi antara sistem dan aktor, use case diagram juga dapat men-deskripsikan tipe interaksi antara si pemakai sistem dengan sistemnya.



Gambar 2.3 Contoh Use Case Diagram

2.12.2 Class Diagram

Class diagram yaitu salah satu jenis diagram pada UML yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan. Jadi diagram ini dapat memberikan sebuah gambaran mengenai sistem maupun relasi-relasi yang terdapat pada sistem tersebut. Berikut contoh Class Diagram :



Gambar 2.4 Contoh Class Diagram

2.12.3 Activity Diagram

Diagram yang menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Berikut contoh Activity Diagram :

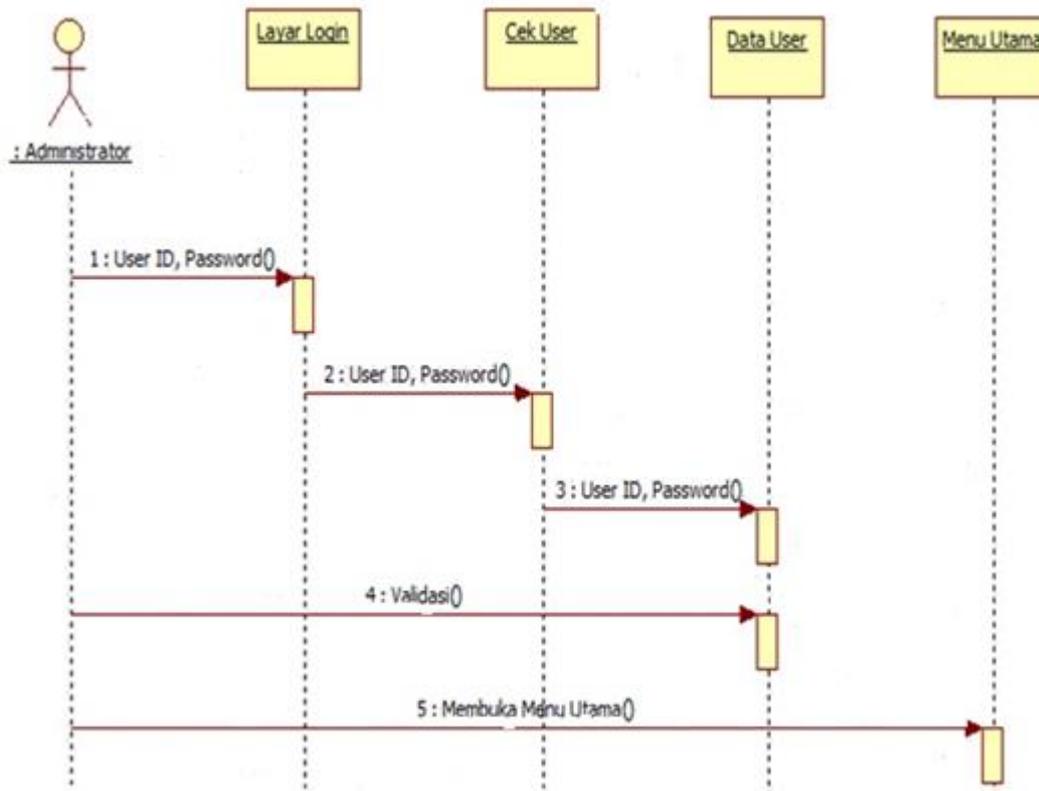


Gambar 2.5 Contoh Activity Diagram

2.12.4 Sequence Diagram

Sequence diagram yaitu salah satu jenis diagram pada UML yang menjelaskan interaksi objek yang berdasarkan urutan waktu, sequence diagram juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada use case diagram.

Berikut Contoh Sequence Diagram :



Gbr 8. Contoh Diagram Sequence

Gambar 2.6 Contoh Sequence Diagram

2.13 Wawancara

Wawancara adalah suatu percakapan antara dua atau lebih orang yang dilakukan oleh pewawancara dan narasumber [14].

Ada juga yang mengatakan bahwa definisi wawancara adalah suatu bentuk komunikasi lisan yang dilakukan secara terstruktur oleh dua orang atau lebih, baik secara langsung maupun jarak jauh, untuk membahas dan menggali informasi tertentu guna mencapai tujuan tertentu pula.

Wawancara (interview) memiliki tujuan yang jelas dan memiliki makna yang melebihi maksud dari percakapan biasa. Proses wawancara ini terjadi dengan adanya komunikasi bolak-balik antara pewawancara dengan orang yang diwawancarai, untuk menggali topik tertentu yang dibahas.

Menurut Sugiyono pengertian wawancara adalah suatu teknik pengumpulan data yang dilakukan secara terstruktur maupun tidak terstruktur dan dapat dilakukan melalui tatap muka maupun dengan menggunakan jaringan telepon.

Seperti yang disebutkan pada definisi wawancara di atas, fungsi wawancara secara umum adalah untuk menggali informasi dari narasumber. Selengkapnya, berikut ini adalah beberapa fungsi wawancara tersebut:

1. Menghindari kesalahan informasi/ data yang simpang siur.
2. Informasi/ data dari hasil wawancara merupakan pelengkap informasi awal.
3. Memperoleh informasi secara komprehensif, akurat, jujur, dan mendalam.
4. Mendapatkan informasi dan data yang objektif dan berimbang.
5. Menggali kemungkinan adanya perspektif baru atas suatu masalah.

Tujuan wawancara secara umum adalah untuk mendapatkan informasi yang akurat dari narasumber dengan menyampaikan beberapa pertanyaan tertentu kepada narasumber.

Secara khusus, berikut ini adalah beberapa tujuan wawancara:

1. Untuk menggali dan mendapatkan informasi atau data dari orang pertama (*primer*).
2. Untuk melengkapi informasi/ data yang dikumpulkan dari teknik pengumpulan data lainnya.
3. Untuk mendapatkan konfirmasi dengan menguji hasil pengumpulan data lainnya.

Dilihat dari sisi pelaksanaannya, wawancara dapat dikelompokkan menjadi tiga jenis. Berikut ini adalah jenis-jenis wawancara:

1. Wawancara Terpimpin

Ini adalah jenis wawancara dimana pewawancara sudah memiliki daftar pertanyaan yang lengkap dan terinci untuk diajukan kepada narasumber.

2. Wawancara Bebas Terpimpin

Ini adalah jenis wawancara dimana pewawancara melakukan kombinasi antara wawancara terpimpin dengan wawancara bebas, dimana pelaksanaannya sesuai dengan pedoman mengenai topik yang dibahas.

3. Wawancara Bebas

Wawancara bebas adalah jenis wawancara dimana pewawancara bebas memberikan pertanyaan kepada responden, namun harus tetap memperhatikan kaitan antara pertanyaan dengan data yang diperlukan. Pada wawancara bebasa terkadang pertanyaan menjadi tak terkendali jika tidak berhati-hati.

Pada penelitian ini penulis melakukan wawancara dengan responden yang bernama ibu Mpon salah satu peternak yang ada di kota bandung , dengan tujuan untuk mendapatkan informasi seputar hewan ternak.

2.14 Pengujian Alpha

Pengujian alpha adalah salah satu strategi pengujian perangkat lunak yang paling umum digunakan dalam pengembangan perangkat lunak, hal ini khusus digunakan oleh organisasi pengembangan produk dengan tujuan agar system yang dikembangkan terhindar dari cacat atau kegagalan penggunaan.

Pengujian ini menggunakan metode Black box yang berfokus pada persyaratan fungsional pada perangkat lunak.

Pengujian ini mencari dan menemukan kesalahan antara lain:

1. Fungsi yang tidak berjalan
2. Kesalahan dalam pengkodean
3. Kesalahan dari suatu proses kinerja
4. Kesalahan interface

Alasan menggunakan pengujian metode Black Box pada penelitian ini, karena berfokus pada spesifikasi fungsional dari perangkat lunak untuk memastikan setiap bagian sudah sesuai dengan alur proses yang ditetapkan dan memastikan semua kesalahan masukan yang dilakukan oleh pengguna dapat ditangani oleh system.