2.8.2 Jenis-Jenis Android

Berikut adalah Jenis-jenis Android disertai dengan fitur yang dimiliki [15] :

1. **Android 1.0 Apple Pie**

   Android ini dirilis pada tanggal 23 September 2008,

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. Download dan updates via Android Market
   b. Web Browser
   c. Support Camera
   d. Sinkronisasi antara Gmail, Contacts dan Google Agenda
   e. Google Maps
   f. Aplikasi YouTube

2. **Android 1.1 Banana Bread**

   Android ini dirilis pada tanggal 9 Februari 2009

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. “Show” & “Hide” numeric keyboard, pada aplikasi telepon
   b. Kemampuan untuk menyimpan MMS attachments.

3. **Android 1.5 Cupcake**

   Android ini dirilis pada tanggal 30 April 2009

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. Support Bluetooth A2DP, AVRCP
   b. Soft-keyboard dengan prediksi text
   c. Record/watch videos

4. **Android 1.6 Donut**

   Android ini dirilis pada tanggal 15 September 2009

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. Gesture framework
   b. Turn-by-turn navigation

5. **Android 2.0 Eclair**

   Android ini dirilis pada tanggal 26 Oktober 2009

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. HTML
   b. Digital zoom
   c. Support Microsoft Exchange
   d. Bluetooth 2.1
   e. Live Wallpapers
   f. Updated UI

6. **Android 2.2 Froyo (Frozen Yogurt)**

   Android ini dirilis pada tanggal 20 Mei 2010

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. Peningkatan Speed
   b. Implementasi JIT
   c. USB Tethering
   d. Aplikasi instalasi untuk perluasan memori
   e. Support file upload pada the browser
   f. Animated GIFs

7. **Android 2.3 Gingerbread**

   Android ini dirilis pada tanggal 6 Desember 2010

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. Updated UI
   b. Peningkatan keyboard ease of use
   c. Peningkatan copy/paste
   d. peningkatan power management
   e. Fitur Social networking
   f. Support NFC (Near Field Communication)
   g. Support Native VoIP/SIP
   h. Support Video call

8. **Android 3.0 Honeycomb**

   Android ini dirilis pada tanggal 22 Februari 2011

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. Support Multi core
   b. Support Tablet lebih baik
   c. Updated 3D UI
   d. Layar Utama (homescreens) yang bisa diatur
   e. Melihat aplikasi yang barusan dibuka
   f. Menyempurnakan layout keyboard
   g. Transport protocol untuk Media/Picture
   h. video chat Google Talk
   i. Google eBooks
   j. "Private browsing"
   k. System-wide Clipboard
   l. HTTP Live streaming

9. **Android 4.0 Icecream Sandwich**

   Android ini dirilis pada tanggal 18 Oktober 2011

   Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

   a. Peningkatan text input dan koreksi ejaan
   b. Control over network data
   c. Support aplikasi Email EAS v14
   d. Fitur WI-FI direct
   e. Support BlueTooth Health Device Profile

10. **Android 4.1 Jelly Bean**

    Android ini dirilis pada tanggal 9 Juli 2012

    Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

    a. Support Google Now
    b. Support Voice Search (pencarian dengan perintah suara)
    c. Peningkatan kecepatan
    d. Peningkatan aplikasi Camera

11. **Android 4.4 KitKat**

    Android ini dirilis pada tanggal 31 Oktober 2013

    Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

    a. Terdapat fitur Screen recording, untuk merekam kegiatan yang terjadi pada layar smartphone kita.
    b. New Translucent system UI
    c. Peningkatan akses notifikasi
    d. System-wide settings untuk closed captioning
    e. Peningkatan kinerja

12. **Android 5.0 Lollipop**

    Android ini dirilis pada tanggal 17 Oktober 2014

    Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

    a. Desain baru (Material)
    b. Peningkatan kecepatan
    c. Peningkatan daya tahan battery

13. **Android 6.0 Marshmallow**

    Android ini dirilis pada tanggal 5 Oktober 2015

    Fitur dan perbaikan yang terdapat pada jenis android ini meliputi :

    a. Support USB Type-C (lihat type dan jenis USB)
    b. Support autentikasi sidik jari *(Fingerprint Authentication)*
    c. Daya tahan battery lebih meningkat dengan manajemen konsumsi battery oleh Doze
    d. Permissions dashboard
    e. Akses System UI Tuner
    f. Support sistem pembayaran dengan Android Pay yang berkolaborasi dengan Fingerprint Authentication sehingga terjamin keamanannya.
    g. Tambahan fungsi Google Now yang tidak sekedar melayani perintah suara

Gambar 2.3 Jenis-Jenis Android

2.9 AndroidSoftware Development Kit (Android SDK)

Android Software Development Kit (Android SDK)adalah sebuah alat yang diperuntukan bagi para programmer yang akan mengembangkan suatu sistem aplikasi android. Android SDK terdiri atas debugger, libraries, handset emulator, dokumentasi, contoh kode dan tutorial. Android SDK juga mencangkup beberapa alat pengembangan yang komprehensif [16].

2.10 Dalvik Debug Monitor Server (DDMS)

Dalvik Debug Monitor Server (DDMS),merupakan tools debugging yang ada pada sistem operasi android yang menyedikan proses pengambilan gambar layar pada perangkat mobile, informasi thread, dan heap pada perangkat *mobile*. *Dalvik Debug Monitor Serve* diintegrasikan ke dalam Eclipse dan juga terdapat di dalam direktori pada Android SDK.

2.11 Android Development Tools (ADT)

*Android Development Tools*merupakan plugin untuk Eclipse yang dirancang untuk perancangan sistem aplikasi android. *Android Development Tools*memungkinkan Eclipse dipakai dalam merancang dan membuat aplikasi Android baru,membuat tampilan *User Interface*, menambah komponen berdasar *framework* API Android, debug aplikasi serta pemaketan aplikasi android.

2.12 Android Studio

Android Studio [17], merupakan Lingkungan Pengembangan Terpadu – *Integrated Development Environment* (IDE) dalam mengembangkan aplikasi Android yang berdasarkan IntelliJ IDEA. Android Studio juga menawarkan berbagai fitur yang dapat meningkatkan produktivitas pada saat pembuatan aplikasi Android, misalnya : Sistem versi berbasis Gradle yang sangat fleksibel, Emulator yang cepat dan dilengkapi dengan berbagai macam fitur, Instant Run yang berfungsi mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru, Template kode dengan integrasi GitHub yang memiliki tujuan membuat fitur aplikasi yang sama dan mengimpor kode contoh, selain itu juga dilengkapi dukungan bawaan untuk *Google Cloud Platform*, mempermudah integrasi *Google Cloud Messaging* dan *App Engine.*

Setiap proyek pada Android Studio memiliki satu ata lebih modul yang dilengkapi file kode sumber dan file sumber daya. Jenis-jenis modul mencangkup, Modul Apliaksi Android, Modul Pustaka dan Modul *Google App Engine* .

Android Studio mendukung berbagai veri sistem kontrol termasuk Git, GitHub, CVS, Mercurial, Subversion, serta Penyimpanan *Google Cloud Source*.

2.13 Alasan menggunakan Android Application Development

Apabila bicara mengenai berbagai macam fitur yang disediakan oleh sistem operasi android yang mampu membantu kita dalam membuat keputusan alasan mengapa kita harus mengembangkan aplikasi android :

1. Proses pengembangan aplikasi android dirasa dapat mempermudah dalam mengidentifikasi dan memperoleh keuntungan dari SDK Android dalam proses pengembangan aplikasi yang kreatif dan inovatif dari sistem operasi android.
2. Dalam proses pengembangan aplikasi android, bahasa pemograman yang digunakan adalah bahasa C/C++ dan bahasa pemograman Java. Sehingga pengembang sistem dapat memahami dengan mudah aplikasi baru yang akan dikembangkan.

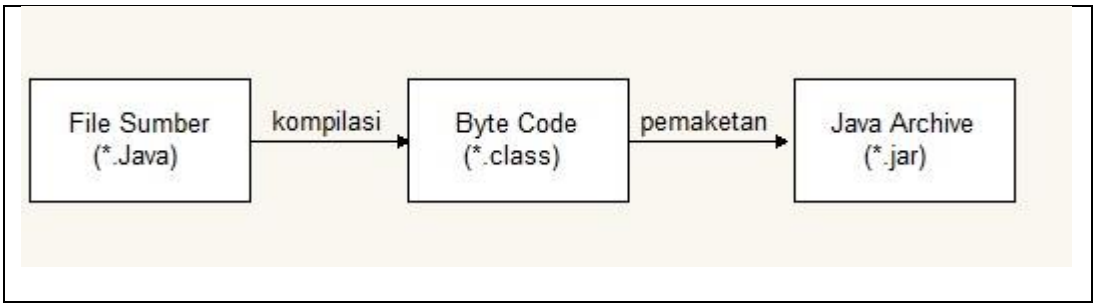
2.14 Java sebagai bahasa pemograman

Java dikembangkan oleh *Sun Microsystems* yang diterbitkan pada tahun 1995. Java adalah bahasa pemograman yang bisa digunakan di berbagai macam *platform*. Inilah yang menjadi dasar kesuksesan bagi Java , bahwa aplikasi Java dapat dijalankan di berbagai sistem operasi seperti, Mosaic, Internet Explorer ataupun Netscape Navigator. Didalam proses pembuatan aplikasi Java langkah pertama kali yang dilakukan adalah dengan melakukan pengetikan kode sumber pada sebuah file yang berakhiran '.java".

Tetapi saat ini telah tersedia Java IDE *(Integrated Development Environment)*yang dapat memudahkan pembuatan aplikasi Java seperti Sun NetBean, Sun Forte dan Eclipse. Harus diperhatikan dalam penamaan file harus sama dengan nama class yang sudah ditulis, dikarenakan penamaan file ini bersifat sensitif-case.

Apabila pembuatan kode sumber sudah selesai dilakukan maka pada tahapan selanjutnya adalah tahap kompilasi. Tahapan ini sumber akan diterjemahkan dan di artikan oleh kompiler Java. Ketika terdapat kesalahan pada file sumber maka dibuatlah sebuah file objek yang berisi kodefikasi dari file sumber.

File objek ini memiliki akhiran '.class' yang disebut *byte-code*. File ini adalah file aplikasi Java yang bisa diproses atau dieksekusi. Tetapi apabila aplikasi semakin kompleks maka akan memiliki banyak sekali file *byte-code* yang dihasilkan. File tersebut dapat dikelompokan menjadi satu file tunggal, pengelompokan file inilah yang disebut 'Java Archive' dimana file tersebut berakhiran 'jar'.



Gambar 2.4 Tahapan kompilasi Java

Setelah dikelompokan, *byte-code*tetap bisa dioperasikan tanpa harus perlu mengekstraknya terlebih dahulu untuk menggunakannya. *Byte-code*Java dimuat oleh '*class loader'* dari disk lokal, *disk network* serta internet ke dalam memori. Apabila *byte-code*sudah tersimpan di memori maka semua kode yang ada didalam memori diperiksa guna menanggulangi kode yang melanggar keamanan sesuai dengan standar yang sudah ditetapkan.

Adanya keberadaan internet, Java telah menyiapkannya dari segi keamanan. Aplikasi Java yang berasal dari sumber yang tidak dapat dipercaya seperti internet, maka Java akan memberi batasan untuk mengakses sistem sehingga sistem tidak dapat dirusak oleh aplikasi. Eksekusi yang memberikan batasan biasanya disebut dengan istilah *'sand box'*. Apabila kode sudah selesai dilakukan pengecekan maka masing-masing kode yang ada di *byte-code*akan diperiksa hal ini untuk menghindari potensi gangguan dan perusakan kestabilan sistem.

2.15 Java Development Kit (JDK)

*Java Development Kit (JDK)*merupakan *software* yang dipakai guna melakukan proses kompilasi aplikasi java ke *bytecode*. JDK berguna bagi programmer untuk membenahi bug, mengcompile, serta menjalankan tambahan program inti yang memakai bahasa pemograman Java.

2.16 SQLite

SQLite adalah suatu proyek bersifat public domain dimana proyek ini merupakan proyek yang dikerjakan oleh D.Richard Hipp. Bermula dari suatu sistem manajemen relasi database yang bersifat *ACID-complaint* dan mempunyai pustaka kecil yang ditulis menggunakan bahasa pemograman C. SQLite juga menjadi salah satu embbeded database dapat diandalkan untuk dipergunakan pada aplikasi enterprise. Berikut adalah kekurangan dan kelebihan dari SQLite.

Tabel 2.2 Kekurangan dan Kelebihan SQLite

Kelebihan SQLite	Kekurangan SQLite
1. Kinerjanya SQLite lebih cepat apabila disandingkan dengan database yang lain. 2. Memakan penggunaan memori yang relatif sedikit. 3. Hanya membutuhkan 1 *library* dalam mengkoneksikan *database*. 4. Dapat diakses di berbagai macam *platform* serta flexibel apabila data ingin dipindahkan karena tidak perlu melakukan setting administrasi. 5. Sudah memenuhi standart *Atomicity, Consistensy, Isolation*, serta *Durability* (ACID). 6. Berlisensi *Public* domain sehingga bebas untuk di distribusikan kembali. 7. Sudahmens*support*ANSI 92 SQL *Standart.*	1. Memiliki *Check Constraint* dimana biasanya ini digunakan untuk proses pemeriksaan, tetapi sudah tidak dipakai lagi karena sudah digantikan oleh NOT NULL dan UNIQUE. 2. SQLite sengaja tidak menggunakan kembali fungsi *Foreign Key Constraint.* 3. Harus membuat ulang apabila kita ingin merubah struktur suatu tabel karena SQLite tidak dilengkapi dengan *Alter Table*. 4. Tidak didukung dengan penggunaan perintah *query* atau *sub query* didalam *query.* 5. Hak izin akses bergantung pada penggunaan sistem operasi karena SQLite melakukan proses pembacaan dan penulisan pada file *disk.*

2.17 Unified Modeling Language (UML)

Unified Modeling Language (UML) [18], merupakan bahasa visual yang berfungsi untuk menjelaskan, merancang, mendokumentasikan aspek-aspek yang ada pada suatu sistem. Berikut adalah fungi dari penggunaan UML :

1. Sebagai *blueprint* karena dengan menggunakan UML akan menjadi sangat detail dan lengkap dalam melakukan proses perancangan suatu sistem.
2. Mampu memodelkan sistem yang memiliki konsep berorientasi pada objek
3. Memberika model atau bentuk yang siap untuk digunakan
4. Memberikan bahasa visual kepada user dari berbagai macam proses rekayasa perangkat maupun berbagai macam pemograman.

Pada tahun 1900an UML ditemukan oleh Grady Booch, James Rumbaugh, serta Ivan Jacobson, kemudian dikembangan kembali oleh konsortium UML hingga menjadi versi 1.0 di tahun 1997.

2.17.1 Class Diagram

Class diagram berfungsi untuk mendeskripsikan pola dari sistem, cara kerja dari class diagram adalah dengan mendefinisikan kelas yang ada untuk kemudian dibangun sebuah perangkat lunak.

Class pada struktur sistem memiliki fungsi masing-masing yang sesuai dengan kebutuhan sistem. Untuk membentuk susunan class perlu mempunyai jenis-jenis class berikut :

1. Class Utama

   Class ini mempunyai fungsi awal yang diproses ketika sistem sedang berjalan.
2. Class yang mengatur tampilan sistem

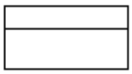
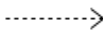
   Class ini akan mengatur dan mendefinisikan tampilan sistem ke user.
3. Class yang berasal dari hasil pendefinisian *usecase*

   Class ini akan mengatur fungsi mana yang harus dijalankan terlebih dahulu dari proses pendefinisian *usecase.*

4. Class hasil dari pendefinisian data

   Class yang dapat digabungkan antara class satu dengan yang lainnya karena class ini bersifat tidak mengikat.

Tabel 2.3 Simbol Class Diagram

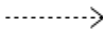
NO	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan antara objek anak atau *descenden,*berbagi perilaku dan struktur data dari objek yang terdapattepat diatasobjek induk (*ancestor*).
2		*Nary Association*	Langkah untuk menghindari asosiasi dengan memiliki lebih dari 2 objek.
3		*Class*	Himpunan dari beberapaobjek yang berbagi atribut dan operasi yang sama.
4		*Collaboration*	Gambaran dari rangkaian aksi yang dimunculkan sistem dan menghasilkan hasil yang terukur bagi aktor.
5		*Realization*	Operasi yang dilakukan oleh sebuah objek.
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri*(independent)*dapatberpengaruh pada elemen yang bergantung pada elemen yang tidak mandiri
7		*Association*	Menghubungkan antara objek satu dengan objek yang lainnya

2.17.2 Usecase

Usecase menggambarkan interaksi aktor terhadap sistem informasi yang akan dibuat. Biasanya didalam satu sistem informasi memiliki satu atau lebih aktor. Terdapat dua hal penting yang ada pada usecase yakni pendefinisian aktor serta usecase.

1. Aktor merupakan orang yang terlibat langsung atau berinteraksi dengan sistem yang akan dibuat.
2. Usecase merupakan diagram yang memiliki sifat statis, dimana diagram ini akan memodelkan dan mengorganisasikan perilaku aktor dari sistem yang akan dibuat.

Tabel 2.4 Simbol Use case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menjelaskan secara detail himpunan peran yang dimainkan oleh penggunapada saat berinteraksi dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada sebuah elemen mandiri*(independent),*berpengaruh pada elemen yang bergantung pada elemen yang tidak mandiri
3		*Generalization*	Hubungan objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang terdapattepat diatasobjek induk (*ancestor*).
4		*Include*	Menspesifikasi bahwa *usecase*adalah sumber secara *eksplisit*.
5		*Extend*	Menspesifikasikan bahwa *usecase* memperluas perilaku dari *use case* sumber pada satu titik yang diberikan.