BAB 2
TINJAUAN PUSTAKA

2.1 Profil Madrasah Diniyah Takmiliyah Al-Barokah

Madrasah Diniyah Takmiliyah(MDT) Al-Barokah merupakan lembaga pendidikan nonformal tingkat awal. struktur kurikulum madrasah ini dikelola dan dikembangkan dengan nuansa berpusat pada potensi dan kebutuhan santri serta menyesuaikan dengan kebutuhan masyarakat . Kompetensi bagi para santri masih bisa dikembangkan dan disesuaikan oleh pihak penyelenggaraan pendidikan.

2.1.1 Lokasi Madrasah Diniyah Takmiliyah Al-Barokah

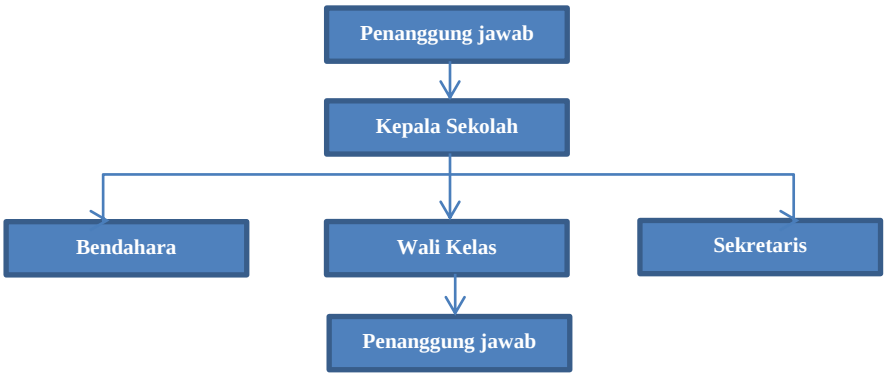
Madrasah Diniyah Takmiliya(MDT)Al-Barokah berlokasi di Jl.Baladewa Gg 2 RW 09 RT 04 Kelurahan Pajajaran Kecamatan Cicendo, Bandung Jawa barat 40173.

2.1.2 Sejarah Madrasah Diniyah Takmiliyah Al-Barokah

Sejarah pendirian Madrasah Diniyah Takmiliyah(MDT) Al-Barokah dimulai pada tahun 1995 setelah dilakukan pembangunan mesjid Jami' Al-Firdaus oleh para tokoh setempat dan seiring perjalannya waktu dan begitu cepatnya perkembangan budaya dan lingkungan semakin merusak generasi anak-anak, disisi lain lingkungan mesjid yang dipadati oleh para penduduk yang secara langsung mempunyai anak-anak baik yang mampu ataupun yang tidak mampu. Maka dengan dorongan agama dan sosial maka pada tahun 2002 didirikan MDT Al-Barokah diperuntukan bagi remaja dari mulai SD-SMP . Pendirian MDT Al-Barokah diinisiasi oleh para tokoh masyarakat yang peduli terhadap perkembangan pendidikan agama anak-anak.

2.1.3 Struktur Organisasi Madrasah Diniyah Al – Barokah

Struktur organisasi merupakan susuanan yang terdiri dari fungsi-fungsidan hubungan- hubungan yang menyatakan seluruh kegiatan untuk mencapai suatu sasaran. Struktru organisasi MDT Al-Barokah bisa dilihat sebagaai berikut :



Gambar 2.1 Struktur Organisasi

2.1.4 Deskripsi Kerja Lembaga

Penangung Jawab : Penangung Jawab sebagai pembina yang membawahi lembaga madrasah sekaligus sebagai penanggung jawab penyelenggaraan pendidikan MDT Al-Barokah.

Kepala Madrasah : Kepala Madrasah sebagai bagian yang mendapatkan tugas untuk mengelola lembaga madrasah dari mulai pengaturan oprasional,keuangan,kurikulum dan perangkat pendidikan dalam proses belajar mengajar sekaligus bertanggung jawab dalam pelaporan terhadap penanggung jawab.

Bendahara : Bendahara sebagai bagian dari pengelolaan masuk dan keluarnya keuangan baik dari eksternal lembaga (berupa infak dsb) maupun dari internal lembaga(iuran dan pembayaran santri).

Sekretaris : Sekretaris sebagai bagian yang bertanggung jawab dalam pencatatan maupun pembuatan administratif lembaga baik bagi para santri ataupun informasi di luar lembaga.

Wali Kelas : Wali Kelas yang di pegang langsung oleh kepala sekolah bertanggung jawab juga berjalannya pembelaajaran di kelas serta pertanggung jawaban kepada pihak wali santri.

Guru : Guru merupakan pengajar di kelas yang bertanggung jawab dalam pembelajaran di kelas.

2.1.5 Visi dan Misi

Visi dan Misi MDT Al-Barokah adalah sebagai berikut :

2.1.5.1 Visi

Visi dari MDT Al-Barokah yaitu membangun santri generasi Qurani yang Cerdas, Berprestasi dan Berakhlakul Karimah.

2.1.5.2 Misi

Sedangkan misi dari Madrasah Diniyah Takmiliyah Al-Barokah adalah sebagai berikut :

1. Melaksanakan pembinaan secara efektif, sehingga setiap santri dapat berkembang secara optimal dalam memahami dasar-dasar ajaran Islam.
2. Meningkatkan disiplin santri dalam amal ibadah kepada Allah swt.
3. Membimbing dan membina santri agar memiliki sifat-sifat kepribadian yang luhur dan berakhlaqul karimah.
4. Memberikan contoh positif terhadap masyarakat dan remaja di lingkungan pendidikan demi terciptanya masyarakat yang Islami

2.1.6 Logo MDT Al – Barokah

Berikut ini merupakan gambar logo MDT Al-Barokah



Gambar 2.2 Logo Organisasi

2.2 Tahfidzul Qur’an

Tahfidz Al-Quran terdiri dari dua suku kata, yaitu Tahfidz dan Al-Quran, yang mana keduanya mempunyai arti yang berbeda. Tahfidz yang berarti menghafal, menghafal dari kata dasar hafal yang dari bahasa arab hafidza - yahfadzu - hifdzan, yaitu lawan dari lupa, yaitu selalu ingat dan sedikit lupa. Menurut Abdul Aziz Abdul Ra'uf definisi menghafal adalah “proses mengulang sesuatu, baik dengan membaca atau mendengar”. Pekerjaan apapun jika sering diulang, pasti menjadi hafal”. Kemudian pengertian Al-Quran menurut istilah adalah kitab yang diturunkan kepada Rasulullah saw, ditulis dalam mushaf, dan diriwayatkan secara mutawatir tanpa keraguan. Setelah melihat definisi menghafal dan Al-Quran di atas dapat disimpulkan bahwa Tahfidz Al-Quran adalah proses untuk memelihara, menjaga dan melestarikan kemurnian Al-Quran yang diturunkan kepada Rasulullah saw di luar kepala agar tidak terjadi perubahan dan pemalsuan serta dapat menjaga dari kelupaan baik secara keseluruhan maupun sebagiannya [1].

2.3 *Game*

Game atau biasa disebut permainan adalah sebuah sistem membuat pemain terlibat dalam konflik buatan, disini pemain dapat berinteraksi dengan sistem dan konflik pada sebuah permainan adalah rekayasa, dalam permainan ada beberapa peraturan yang dapat membatasi perilaku pemain dan menentukan permainan. Permainan adalah sebuah kegiatan dengan tujuan bersenang-senang, mengisi waktu luang, atau berolahraga santai[7].

2.3.1 Game Edukasi

Game Edukasi adalah game yang dibuat untuk membantu pengguna dalam mempelajari sesuatu, baik tentang konsep, pemahaman ataupun latihan [8].

Menurut Hurd dan Jenuings, perancangan *Education game* yang baik haruslah memenuhi beberapa kriteria berikut ini: [8]

1. Nilai Keseluruhan (*Overall Value*)

   Nilai keseluruhan dari suatu game terpusat pada desain dan panjang durasi game.

2. Dapat Digunakan(*Usability*)

   Mudah digunakan adalah poin penting bagi pembuat game.

3. Keakuratan (*Accuracy*)

   Keakuratan diartikan sebagai bagaimana keberhasialn model sebuah game dapat dituangkan ke dalam percobaan atau perancangannya.

4. Kesesuaian (*Appropriateness*)

   Kesesuaian dapat diartikan bagaimana isi dan desain game dapat diadaptasikan terhadap keperluan user dengan baik.

5. Relevan (*Relevance*)

   Relevan artinya dapat menyampaikan isi game ke target user. Sistem harus dapat menuntun user untuk mencapai tujuan pembelajaran.

6. Objektifitas (*Objectives*)

   Objektifitas dalam menentukan tujuan user dan karakteristik dari keberhasilan atau kegagalan.

7. Umpan Balik (*Feedback*)

   Untuk membantu pemahaman user bahwa permainan (*performance*) mereka sesuai dengan objek game atau tidak, *feedback* harus disediakan.

2.4 *Speech Recognition*

Pengenalan ucapan atau pengenalan suara dalam istilah bahasa Inggris, *Automatic Speech recognition* (ASR) adalah suatu teknik dan sistem yang memungkinkan komputer dalam menerima sebuah masukan berupa kata yang diucapkan. Hasilnya dapat ditampilkan dalam bentuk tulisan yag dapat dibaca oleh perangkat teknologi sebagai sebuah perintah untuk melakukan suatu pekerjaan [3].

Alat pengenal ucapan, yang sering disebut dengan *Speech recognizer*, memerlukan contoh kata sebenarnya yang diucapkan pengguna. Contoh kata akan diubah mnejadi bentuk digital dan disimpan dalam komputer, kemudian digunakan sebagai basis data dalam mencocokkan kata yang diucapkan selanjutnya.

2.5 Android

Linux adalah basis dari sistem operasi android. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri dan digunakan oleh berbagai macam perangkat mobile. Pada saat perilisan perdana, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode – kode Android di bawah lisensi Apache. Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau Google Mail Services (GMS) dan kedua adalah yang benar–benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai Open Handset Distribution (OHD) [9]



Gambar 2.3 Logo Android [9]

2.5.1 Sejarah Android

Pada Juli 2000, Google bekerjasama dengan Android Inc., perusahaan yang terdapat di Palo Alto, California Amerika Serikat. Saat itu banyak yang menganggap fungsi Android Inc. hanyalah sebagai perangkat lunak pada telepon seluler. Sejak saat itu muncul rumor bahwa Google hendak memasuki pasar telepon seluler. Di perusahaan Google, Robin menjadi pemimpin dalam tim yang bertugas mengembangkan program perangkat seluler yang didukung oleh kernel Linux. Hal ini menunjukkan indikasi bahwa Google sedang bersiap menghadapi persaingan dalam pasar telepon seluler.[9]

2.5.2 Arsitektur Android

Secara garis besar Arsitektur Android adalah sebagai berikut :

2.5.2.1 *Applications dan Widgets*

Puncak dari diagram arsitektur Android adalah lapisan aplikasi dan widget. Lapisan aplikasi merupakan lapisan yang paling terlihat oleh pengguna saat program dijalankan. Pengguna hanya bakal menyaksikan program kala digunakan tanpa tahu sistem yang berjalan dibalik lapisan aplikasi[9].

2.5.2.2 Aplication Framworks

Kerangka aplikasi menyediakan kelas-kelas yang bisa digunakan untuk mengembangkan aplikasi Android. Bagian terpenting didalam kerangka aplikasi Android adalah sebagai tersebut [9]:

1. Activity Manager, berfungsi untuk mengontrol siklus hidup aplikasi dan menjaga keadaan ”Backstack“ untuk navigasi penggunaan.
2. Content Providers, berfungsi untuk merangkum data yang memungkinkan digunakan oleh aplikasi lainnya, seperti daftar nama.
3. Resource Manager, untuk mengatur sumber daya yang ada dalam program. Serta menyediakan akses sumber daya diluar kode program, seperti karakter, grafik, dan file layout.
4. Location Manager, berfungsi untuk memberikan informasi detail mengenai lokasi perangkat Android berada.
5. Notification Manager, mencakup berbagai macam peringatan seperti, pesan masuk, janji, dan lain sebagainya yang akan ditampilkan pada status bar.

2.5.2.3 Libraries

Android menggunakan beberapa paket pustaka yang terdapat pada C/C++ dengan standar Berkeley Software Distribution (BSD) hanya setengah dari yang aslinya untuk tertanam pada kernel Linux. Beberapa pustaka diantaranya[9]:

1. Media Library untuk memutar dan merekam berbagai macam format audio dan video.

2. Surface Manager untuk mengatur hak akses layer dari berbagai aplikasi.
3. Graphic Library termasuk didalamnya SGL dan OpenGL, untuk tampilan 2D dan 3D.
4. SQLite untuk mengatur relasi database yang digunakan pada aplikasi.
5. SSl dan WebKit untuk browser dan keamanan internet.

Pustaka-pustaka tersebut bukanlah aplikasi yang berjalan sendiri, namun hanya dapat digunakan oleh program yang berada di level atasnya. Sejak versi Android 1.5, pengembang dapat membuat dan menggunakan pustaka sendiri menggunakan Native Development Toolkit (NDK).

2.5.2.4 Android Runtime

Android Runtime merupakan mesin virtual yang membuat aplikasi Android menjadi lebih baik dengan paket pustaka yang telah ada. Dalam Android Runtime terdapat 2 bagian utama, yaitu [9]:

1. Pustaka Inti, Android dikembangkan melalui bahasa pemrograman Java, tapi Android Runtime bukanlah mesin virtual Java. Pustaka inti Android menyediakan hampir semua fungsi yang terdapat pada pustaka Java serta beberapa pustaka khusus Android.
2. Mesin Virtual Dalvik, Dalvik merupakan sebuah mesin virtual yang dikembangkan oleh Dan Bornstein yang terinspirasi dari nama sebuah perkampungan yang berada di Iceland. Dalvik hanyalah interpreter mesin virtual yang mengeksekusi file dalam format Dalvik Executable (*.dex).

2.5.2.5 Linux Kernel

Android dibangun di atas kernel Linux 2.6. Linux merupakan sistem operasi terbuka yang handal dalam manajemen memori dan proses. Oleh karenanya pada Android hanya terdapat beberapa pelayanan yang diperlukan seperti keamanan, manajemen memori, manajemen proses, jaringan dan driver. Kernel linux menyediakan driver layar, kamera, keypad, WiFi, Flash Memory,

audio, dan IPC (Interprocess Communication) untuk mengatur aplikasi dan keamanan.[9]

2.5.3 Komponen Aplikasi Android

Fitur penting Android adalah bahwa satu aplikasi dapat menggunakan elemen dari aplikasi lain (untuk aplikasi yang memungkinkan). Sebagai contoh, sebuah aplikasi memerlukan fitur scroller dan aplikasi lain telah mengembangkan fitur scroller yang baik dan memungkinkan aplikasi lain menggunakannya.

Agar fitur tersebut dapat bekerja, sistem harus dapat menjalankan aplikasi ketika setiap bagian aplikasi itu dibutuhkan, dan pemanggilan objek java untuk bagian itu. Oleh karenanya Android berbeda dari sistem-sistem lain, Android tidak memiliki satu tampilan utama program seperti fungsi main() pada aplikasi lain. Sebaliknya, aplikasi memiliki komponen penting yang memungkinkan sistem untuk memanggil dan menjalankan ketika dibutuhkan.

2.5.3.1 Activities

Activity merupakan bagian yang paling penting dalam sebuah aplikasi, karena Activity menyajikan tampilan visual program yang sedang digunakan oleh pengguna. Setiap Activity dideklarasikan dalam sebuah kelas yang bertugas untuk menampilkan antarmuka pengguna yang terdiri dari Views dan respon terhadap Event.

2.5.3.2 Services

Suatu service tidak memiliki tampilan antarmuka, melainkan berjalan di background untuk waktu yang tidak terbatas. Komponen service diproses tidak terlihat, memperbarui sumber data dan menampilkan notifikasi. Service digunakan untuk melakukan pengolahan data yang perlu terus diproses, bahkan ketika Activity tidak aktif atau tidak tampak.

2.5.3.3 Intents

Intens merupakan sebuah mekanisme untuk menggambarkan tindakan tertentu, seperti memilih foto, menampilkan halaman web, dan lain sebagainya. Intents tidak selalu dimulai dengan menjalankan aplikasi, namun juga digunakan

oleh sistem untuk memberitahukan ke aplikasi bila terjadi suatu hal, misal pesan masuk.

2.5.3.4 Broadcast Receiver

Broadcast Receivers merupakan komponen yang sebenarnya tidak melakukan apa-apa kecuali menerima dan bereaksi menyampaikan pemberitahuan. Sama halnya dengan service, Broadcast Receivers tidak menampilkan antarmuka pengguna. Namun, Broadcast Receivers dapat menggunakan Notification Manager untuk memberitahukan sesuatu kepada pengguna.

2.5.3.5 Content Providers

Content Providers digunakan untuk mengelola dan berbagi database. Data dapat disimpan dalam file sistem, dalam database SQLite, atau dengan cara lain yang pada prinsipnya sama. Dengan adanya Content Provider memungkinkan antar aplikasi untuk saling berbagi data. Komponen ini sangat berguna ketika sebuah aplikasi membutuhkan data dari aplikasi lain, sehingga mudah dalam penerapannya.

2.5.4 Tipe Aplikasi Android

Terdapat 3 kategori aplikasi android :

1. Foreground Activity

   Aplikasi yang hanya dapat dijalankan jika tampil pada layar dan tetap efektif walaupun tidak terlihat.
2. Background Service

   Aplikasi yang memiliki interaksi terbatas dengan user, selain dari pengaturan konfigurasi, semua dari prosesnya tidak tidak tampak pada layar.
3. Intermittent Activity

   Aplikasi yang masih membutuhkan beberapa masukkan dari pengguna, namun sebagian sangat efektif jika dijalankan di background dan jika diperlukan akan memberi tahu pengguna tentang kondisi tertentu.

Untuk aplikasi yang kompleks akan sulit untuk menentukan kategori aplikasi tersebut apalagi aplikasi memiliki ciri-ciri dari semua kategori. Oleh karenanya perlu pertimbangan bagaimana aplikasi tersebut digunakan dan menentukan kategori aplikasi yang sesuai.

2.6 Unified Modeling Language (UML)

Unified Modelling Language (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual (Braun, et. al. 2001). UML merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek (Whitten, et. al. 2004).[12]

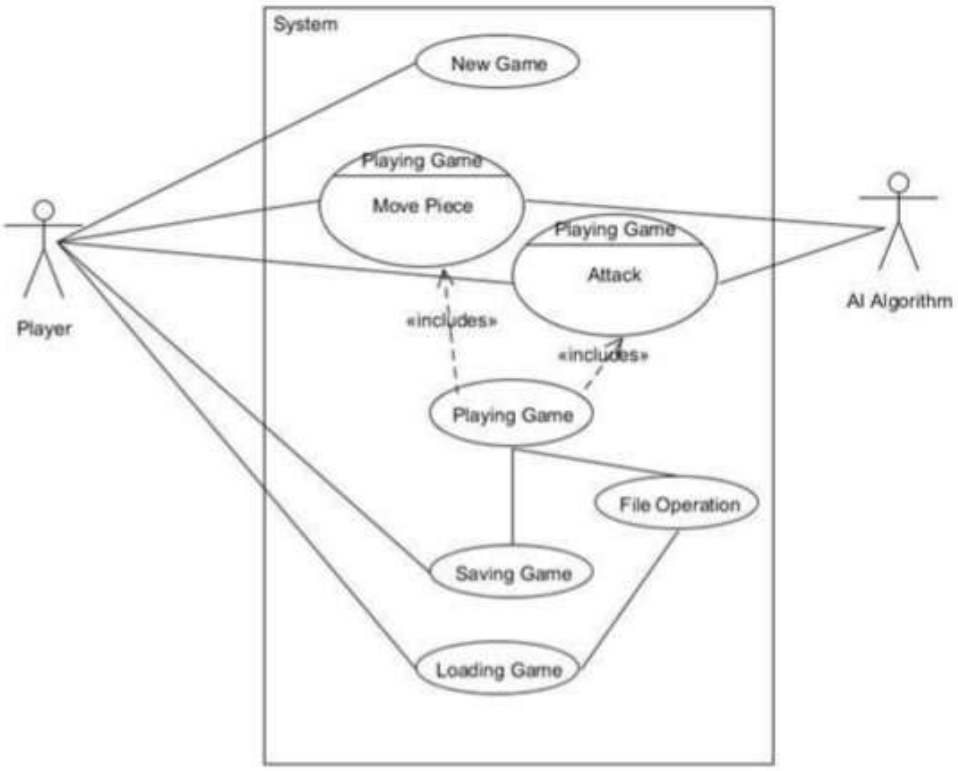
2.6.1 Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah use case merepresentasikan sebuah interaksi antara aktor dengan sistem. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. [12]

Use case diagram akan sangat menolong saat kita sedang menyusun requirement sebuah sistem, mengkomunikasikan rencana bersama dengan klien, dan merancang test case untuk seluruh feature yang tersedia pada sistem.

Sebuah use case juga mampu meng-extend use case lain bersama dengan behaviour-nya sendiri. Sementara hubungan generalisasi antar use case

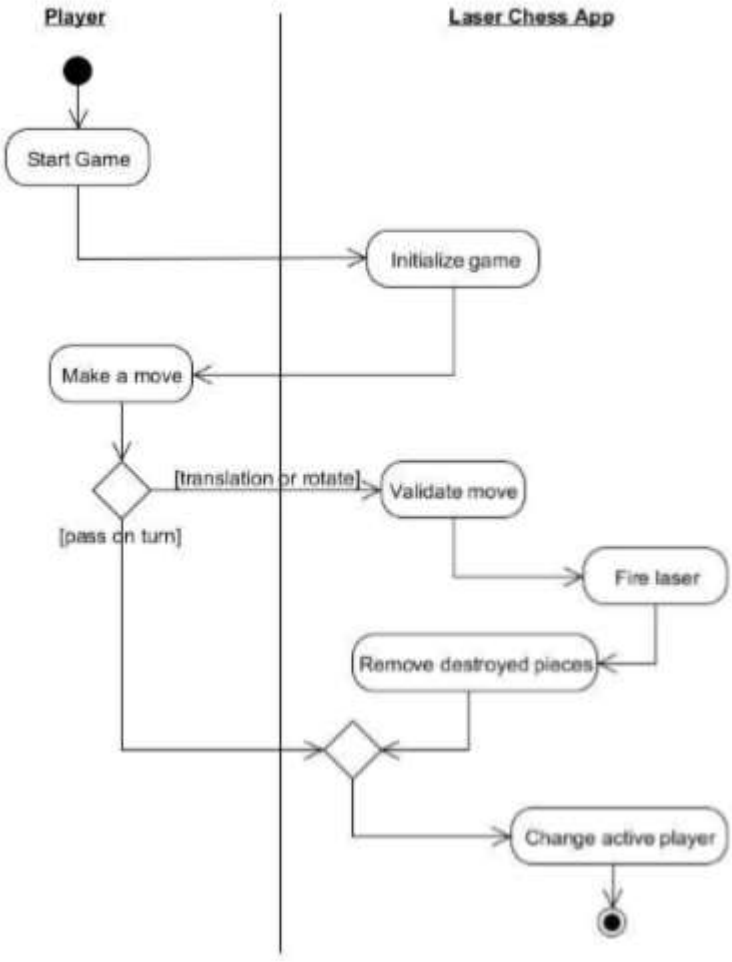
menunjukkan bahwa use case yang satu merupakan spesialisasi dari yang lain.



Gambar 2.4 Contoh Use Case[12]

2.6.2 Activity Diagram

Activity diagram merupakan state diagram khusus, di mana beberapa besar state adalah action dan beberapa besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh dikarenakan itu activity diagram tidak menjelaskan behaviour internal sebuah sistem (dan pertalian antar subsistem) secara eksak, namun lebih menjelaskan pada proses-proses dan jalur-jalur aktifitas dari level atas secara umum.[12]

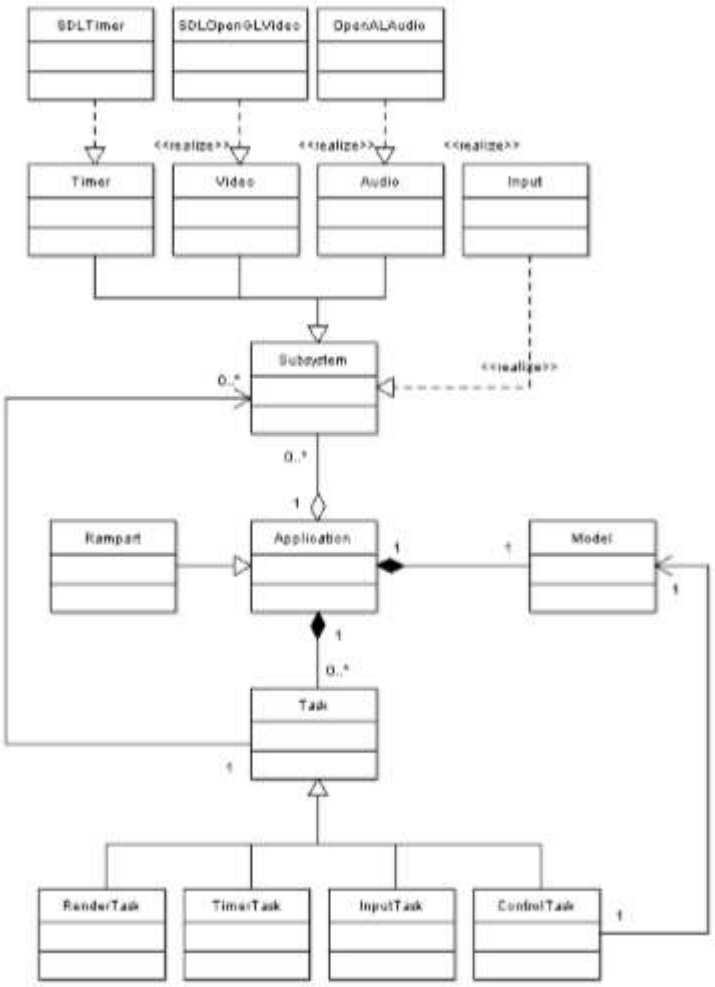


Gambar 2.5 Contoh Activity Diagram[12]

2.6.3 Class Diagram

Class diagram menunjang didalam memvisualisasikan susunan kelas-kelas dari suatu proses dan merupakan type diagram yang paling banyak dipakai. Selama langkah desain, class diagram berperan didalam menangkap susunan dari seluruh kelas yang membentuk arsitektur proses yang dibuat. [12]

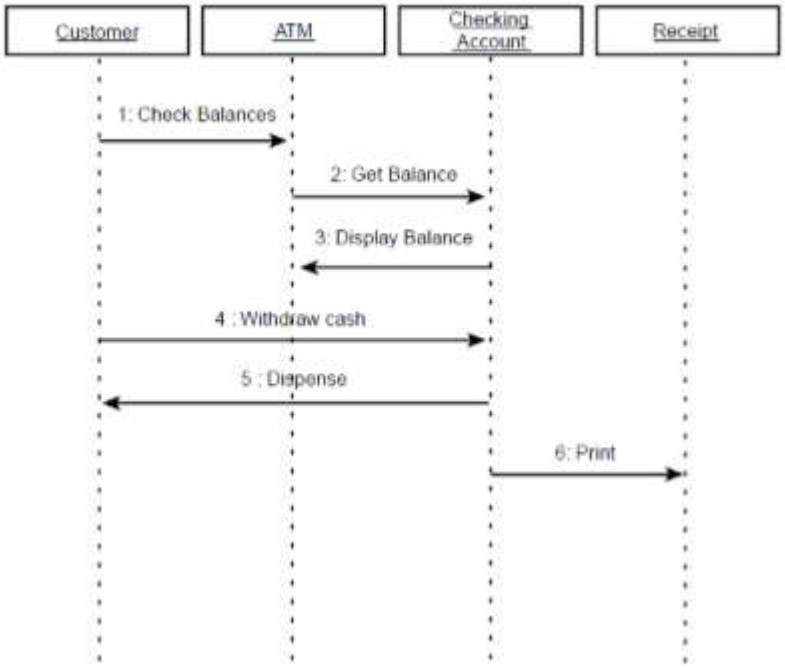
Class adalah sebuah spesifikasi yang jikalau diinstansiasi akan membuahkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class diagram menjelaskan struktur dan uraian class, package dan objek beserta interaksi satu mirip lain seperti containment, pewarisan, asosiasi, dan lain-lain seperti pada gambar berikut.



Gambar 2.6 Contoh Class Diagram[12]

2.6.4 Sequence Diagram

Diagram sequence menampilkan interaksi antar objek dalam dua dimensi. Dimensi vertikal adalah waktu, dimana waktu terjadi ke arah bawah. Sedangkan dimensi horizontal merepresentasikan objek-objek individual. Tiap objek (termasuk actor) tersebut mempunyai waktu aktif yang direpresentasikan dengan kolom vertikal yang disebut dengan lifeline. Pesan (message) direpresentasikan sebagai panah berasal dari satu lifeline ke lifeline yang lain. Secara mudahnya sequence diagram adalah deskripsi tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dijalankan untuk membuat sesuatu sesuai dengan use case diagram.[12]



Gambar 2.7 Contoh Sequence Diagram[12]

2.7 Tools yang Digunakan

Tools yang digunakan untuk membangun aplikasi yaitu Android Studio, Photoshop CS6, Corel Draw.

2.7.1 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, adalah sebagai berikut :[11]

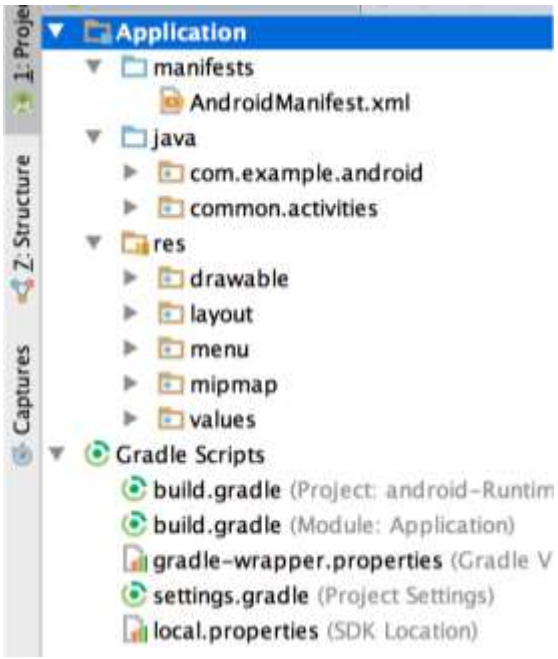
1. Sistem versi berbasis Gradle yang fleksibel
2. Emulator yang cepat dan memiliki banyak fitur
3. Lingkungan memudahkan untuk pengembangan bagi semua perangkat Android

4. Instant Run untuk melakukan 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 uji dan kerangka kerja yang fleksibel
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.

2.7.1.1 Struktur Proyek

Setiap proyek di Android Studio memiliki satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis-jenis modul mencakup[11]:

1. Modul aplikasi Android
2. Modul Pustaka
3. Modul Google App Engine



Gambar 2.8 Arsitektur Android Studio [11]

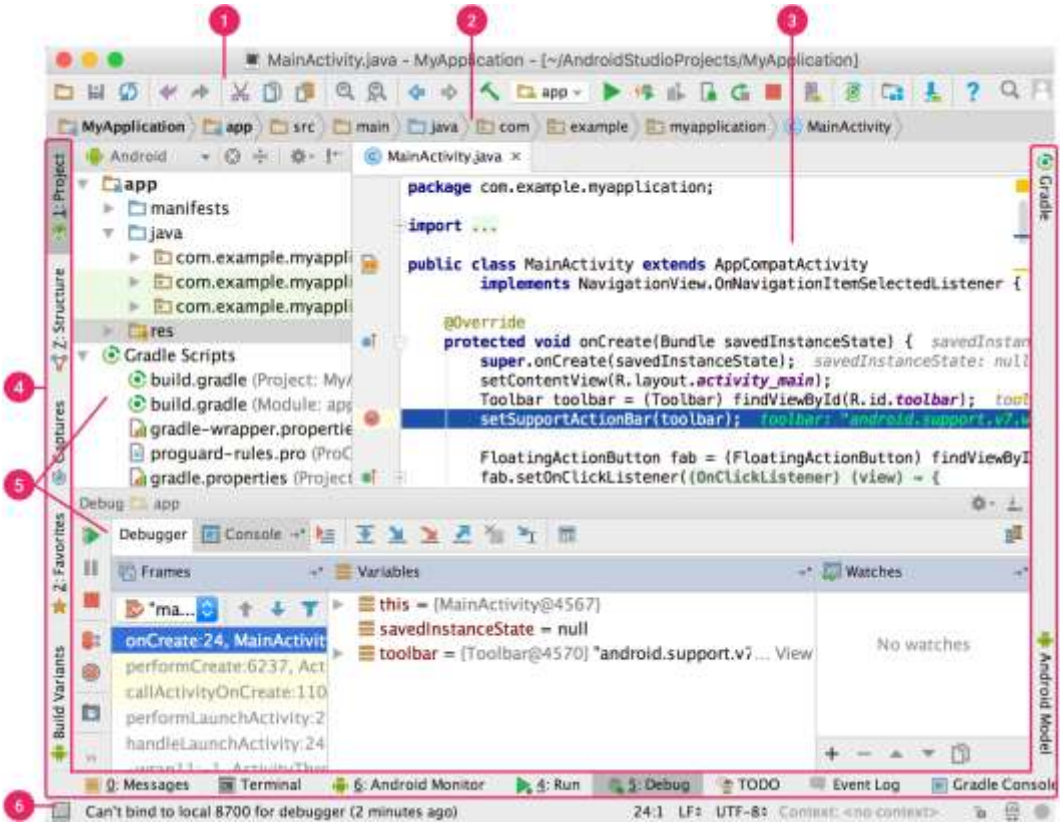
Normalnya Android Studio akan menampilkan file proyek Anda dalam tampilan proyek Android, seperti dalam gambar. Tampilan disusun berdasarkan modul untuk memberikan akses cepat ke file sumber utama proyek.

Semua file versi terlihat di bagian atas di bawah Gradle Scripts dan masing-masing modul aplikasi berisi folder berikut:

1. *manifests*: Berisi file AndroidManifest.xml.
2. java: Berisi file kode sumber Java, termasuk kode pengujian JUnit.
3. res: Berisi semua sumber daya bukan kode, seperti tata letak XML, *string* UI, dan gambar bitmap.

2.7.1.2 Antarmuka Pengguna

Jendela utama Android Studio terdiri dari beberapa bidang logika yang diidentifikasi dalam gambar berikut[11]:



Gambar 2.9 Antarmuka Android Studio[11]

1. Bagian alat : digunakan untuk melakukan berbagai jenis tindakan, termasuk menjalankan aplikasi dan meluncurkan berbagai macam alat Android.
2. Bagian navigasi : digunakan untuk bernavigasi di antara proyek dan membuka file untuk diubah. Bagian ini memberikan tampilan struktur yang terlihat lebih ringkas dalam jendela proyek.
3. Bagian text editor : tempat Anda membuat dan memodifikasi kode. Editor ini dapat berubah sesuai dengan file yang sedang dibuka.
4. Bagian *windows tool* muncul di luar jendela IDE dan berisi tombol yang memungkinkan memperbesar atau mengecilkan tampilan IDE pada layar.
5. Bagian untuk mengakses ke tugas tertentu seperti pengelolaan proyek, penelusuran, kontrol versi, dan banyak lagi.
6. Bagian status menampilkan status proyek dan IDE itu sendiri, serta setiap peringatan atau pesan.

2.7.2 Adobe Photosohop

Adobe Photoshop merupakan suatu aplikasi yang paling banak digunakan dikalangan desainer grafis dan pengolahan citra. Sejak pertama diperkenalkan oleh Adobe corporation pada dekade 90-an, Photoshop langsung mendapatkan tempat dikalangan profesional dan praktisi imege-editing dengan segala kecanggihan fitur dan kemampuan yang maksimal serta kemudahan dalam penggunaannya.

Adobe Photoshop yang dirilis pada tahun 2012 yang lalu oleh Adobe corp., adalah Adobe Photoshop CS dengan slogan yang cukup menjanjikan *The Professional Standard in Desktop Digital Imaging* merupakan pengembangan dari Adobe Photoshop generasi sebelumnya, dan pada edisi terakhir ini sudah dilengkapi dengan fasilitas Image Ready CS. Kelengkapan ini dimaksudkan untuk memberikan fleksibelitas, efisiensi dan kemudahan dalam penggunaan, sehingga dapat dengan mudah membuat dan menyunting image dengan kualitas tinggi yang

siap untuk dicetak, publisher, ataupun untuk ditempatkan di web, yang selama ini merupakan kendala untuk menempatkan image berkualitas tinggi pada web, karena berhubungan erat dengan beban loading data di sisi client pengguna.

2.7.2.1 Area Kerja Phototshop

Beberapa Area kerja Photoshop, diantaranya :

1. Menu Bar Berisi perintah utama untuk membuka file, save, mengubah ukuran gambar, filter dan lain-lain.
2. Option Berisi pilihan dari tool yang dipilih. Misalnya dipilih kuas/brush, maka ukuran/diameter brush ada di sini.
3. Gambar Menampilkan gambar yang sedang dibuat atau diperbaiki.
4. Pallete Well Cara cepat untuk mengakses palet brushes, tool resets dan Layer Comps. Juga dapat digunakan untuk meletakkan palet yang sering digunakan.
5. Toolbox Berisi tool untuk menyeleksi dan memodifikasi gambar.
6. Palette Berisi jendela-jendela kecil yang di dalamnya terdapat perintah dan pilihan untuk dokumen/gambar yang sedang dikerjakan.

2.7.3 Corel Draw

CorelDRAW adalah aplikasi untuk mengolah grafik bertipe vector yang dikembangkan dan dipasarkan oleh Corel Corporation di Ottawa, Kanada. CorelDRAW pertama kali dibuat pada tahun 1987 oleh programmer bernama Michael Bouillon dan Pat Beirne. Awalnya program CorelDRAW dirilis pada tahun 1989, CorelDRAW versi 1.x dan 2.x yang berjalan pada platform Windows.[10]

CorelDRAW sering menjadi pilihan utama bagi para seniman digital karena kemudahan yang ditawarkan. Beberapa tools yang ditawarkan oleh corel draw yang berbasis vector adalah : Note-edit tool,stroke before fill,mesh fill dan sebagainya. Corel draw juga memiliki perbedaan mencolok dibandingkan dengan saingannya, yang pertama adalah bahwa corel draw adalah satu paket software grafis, bukan hanya sebuah editor gambar berbasis vector. Peralatan – peralatan yang ada memungkinkan penggunanya untuk mengatur kontras, keseimbangan

warna bahkan mengubah dari mode RGB(RED, GREEN ,BLUE) menjadi CMYK (CYAN ,MAGENTA,YELLOW).