BAB 2
TINJAUAN PUSTAKA

2.1 Tempat Penelitian

2.1.1 Profil Tempat Penelitian

SMP Negeri 48 Bandung terletak di Jalan Baturaden VIII Nomor 19 Kelurahan Mekarjaya Kecamatan Rancasari Kota Bandung. Berdiri sejak tahun 1982 dengan menempati tanah seluas 11.907 meter persegi. Masyarakat yang mendaftarkan diri di SMP Negeri 48 Bandung begitu besar selain itu jumlah Sekolah Dasar (SD) di kecamatan Rancasari berjumlah 45 SD sehingga peserta didik yang masuk ke SMP Negeri 48 dari tahun ke tahun meningkat terus. Dengan adanya kepercayaan masyarakat untuk menyekolahkan putera-puterinya di SMP Negeri 48 menjadi suatu motivasi dan tantangan bagi sekolah untuk dapat menyelenggarakan proses pembelajaran yang optimal sesuai dengan Standar Pelayanan Minimal Sekolah Menengah Pertama yang mengharapkan peserta didik SMP memiliki akhlak dan budi pekerti yang luhur, pengetahuan dan keterampilan dasar sesuai dengan kurikulum yang berlaku, kesehatan dan kebugaran, apresiasi seni dan dasar-dasar olah raga sesuai bakat dan minat, kemampuan untuk melanjutkan pendidikan ke jenjang yang lebih tinggi.

2.1.2 Visi dan Misi

Visi :

Unggul dalam prestasi yang memberikan karakter berbasis lingkungan di landasi iman dan taqwa.

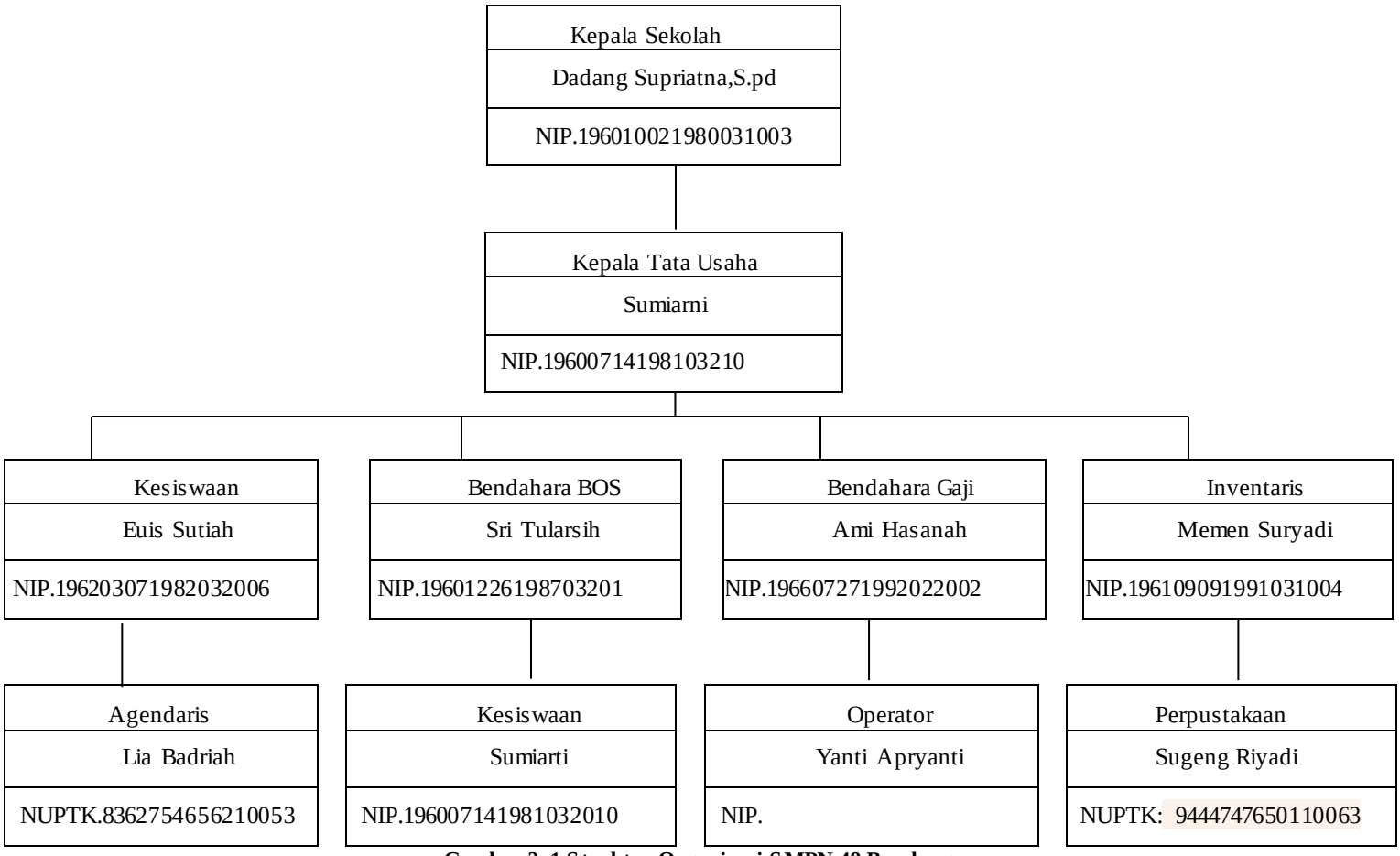
Misi :

1. Mengembangkan potensi di bidang akademis dan non akademis bagi seluruh warga sekolah.
2. Mewujudkan lingkungan belajar yang kondusif dengan layanan prima.
3. Menerapkan manajemen partisipatif dengan warga sekolah.

4. Meningkatkan harmonisasi antar warga sekolah dan lingkungan sekitarnya.
5. Meningkatkan lingkungan yang bersih , indah, sehat dan nyaman.
6. Mengamalkan karakter religus, disiplin, jujur dan tanggung jawab.
7. Menghasilkan lulusan yang berkualitas dan berdaya saing tinggi.

2.1.3 Struktur Organisasi

Struktur organisasi SMP Negeri 48 Bandung tahun ajaran 2014/2015



Gambar 2. 1 Struktur Organisasi SMPN 48 Bandung

2.1.4 Daftar Uraian Tugas Pekerjaan Pegawai

1. Kepala Tata Usaha
   a) Penyusunan Program Tata Usaha
   b) Pengurusan Administrasi
   c) Pembinaan Dan Pengembangan Karier Pegawai Tata Usaha
   d) Penyusunan Administrasi Perlengkapan Sekolah

e) Penyusunan Dan Penyajian Data Statistic Sekolah
f) Penyusunan Laporan Pelaksanaan Kegiatan Pengurusan Ketata Usahaan Secara Berkala
g) Membuat Laporan SPJ Provinsi

2. Bendahara Bos Pusat
   a) Membuat Laporan SPJ Bos Pusat
   b) Membayar Gaji Guru Dan Karyawan Honor
3. Kesiswaan
   a) Mengiis Buku Induk Kelas IX
   b) Mengerjakan Kredit Point Guru
   c) Mengisi Buku Klaper Dan Buku Mutasi
   d) Laporan Keadaan Siswa Persemester
4. Inventaris
   a) Mengerjakan Buku Induk Barang
   b) Mengerjakan Buku Golongan Barang
   c) Mengerjakan Buku Daftar Barang
   d) Mengerjakan Buku Pembelian Barang
   e) Mengerjakan Laporan Penghapusan Pemeliharaan Barang
   f) Membuat Laporan Bulanan
5. Bendahara Bos Kota
   a) Membuat Laporan Spj Bos Kota
   b) Membayar Gaji Guru Dan Karyawan Pns

2.2 Landasan Teori

2.2.1 Aplikasi

Aplikasi merupakan penerapan, menyimpan sesuatu hal data, permasalahan, pekerjaan ke dalam suatu sarana atau media yang dapat digunakan untuk menerapkan atau mengimplementasikan hal atau permasalahan yang ada sehingga berubah menjadi suatu bentuk yang baru tanpa menghilangkan nilai-nilai dasar dari hal data, permasalahan, pekerjaan itu sendiri [8].

Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Aplikasi merupakan rangkaian kegiatan atau perintah untuk dieksekusi oleh komputer. Program merupakan kumpulan *instructionset* yang akan dijalankan oleh pemroses, yaitu berupa *software*. Bagaimana sebuah sistem komputer berpikir diatur oleh program ini. Program inilah yang mengendalikan semua aktifitas yang ada pada pemroses. Program berisi konstruksi logika yang dibuat oleh manusia, dan sudah diterjemahkan ke dalam bahasa mesin sesuai dengan format yang ada pada *instructionset*.

2.2.2 Pembelajaran

Pembelajaran adalah suatu kegiatan yang dilakukan secara sadar dan sengaja. Tujuan pembelajaran adalah membantu siswa agar memperoleh berbagai pengalaman dan dengan pengalaman itu tingkah laku yang dimaksud meliputi pengetahuan, ketrampilan, dan nilai atau norma yang berfungsi sebagai pengendali sikap dan prilaku siswa. Tujuan pembelajaran menggambarkan kemampuan atau tingkat penguasaan yang diharapkan dicapai oleh siswa setelah mereka mengikuti suatu proses pembelajaran[9].

Tujuan pembelajaran adalah perubahan prilaku dan tingkah laku yang positif dari peserta didik setelah mengikuti kegiatan belajar mengajar, seperti perubahan yang secara psikologis akan tampil dalam tingkah laku (over behaviour) yang dapat diamati melalui alat indera oleh orang lain baik tutur katanya, motorik dan gaya hidupnya.

Hal ini dapat disimpulkan bahwa proses pembelajaran merupakan proses melibatkan guru dengan semua komponen tujuan, bahan, metode dan alat serta penilaian. Jadi proses pembelajaran merupakan suatu sistem yang saling terkait antar komponennya di dalam mencapai suatu tujuan yang telah ditetapkan.

2.2.3 Tahsin Qur'an

Tajwid menurut bahasa artinya tahsin yang berarti memperbaiki, menghias, mempercantik, membuat lebih baik (dari semula)(Kamus Al

Munawwir : 285), sedangkan tajwid menurut istilah berarti mengeluarkan setiap huruf dari makhrajnya dengan memberikan hal dan mustahaknya. Hak berkaitan dengan sifat – sifat urut hijaiyah sedangkan mustahaq berkaitan dengan hukum-hukum tajwid. Dari definisi diatas dapat disimpulakn bahwa ruang lingkup pembahasan ilmu tajwid mencakup 3 hal, yaitu makharijul huruf, hak huruf dan mustahak huruf. Makhraj berarti tempat keluarnya suara hurud seperti ba keluar dari dua bibir, kha keluar dari tenggorokan bagian atas. Hak huruf yaitu sifat yang senantiasa melekat pada setiap huruf, seperti hams (mengalirkan nafas) ketika membaca huruf ta atau jahr (tertahannya nafas) ketika membaca huruf ja, sedangkan mustahak huruf yaitu sifat yang muncul sewaktu-waktu karena hubungan huruf, seperti izhar (membaca dengan jelas) pada nun sakinak (mati) ketika bertemu huruf ‘ain atau idgham (melebur ke huruf berikutnya) pada nun sakinah ketika bertemu huruf lam[10].

Hukum mempelajari ilmu tajwid para ulama tajwid membagi tajwid menjadi dua bagian. Pertama tajwid teori yang disebut tajwid ‘ilmi, hukum mempelajarinya yaitu fardhu kifayah. Kedua, tajwid praktek yang disebut dengan tajwid ‘amali, hukum mempelajarinya fardhu ‘ain, berdasarkab firman Allah SWT.

وَرَتِّلِ ٱلْقُرْءَانَ تَرْتِيلًا

“ Dan bacalah Al Qur’an dengan tartil (perlahan-lahan)” (QS. Al Muzzammil : 4)

Penyusun ilmu tajwid dari segi praktek adalah Rasulullah SAW yang bacaannya beliau terima dari Jibril yang diterima dari Allah SWT, sedangkan dari segi teori orang pertama yang menyusun ilmu tajwid terdapat perbedaan pendapat, ada yang menyebut :

1. Al Khalil bin Ahmad Al Farahidi
2. Abul Aswad Ad Du’ali
3. Hafs bin Umar Ad Duri
4. Para Imam Qiraah

Ilmu tajwid termasuk ilmu yang sangat utama karena keterkaitannya secara langsung dengan kitab yang sangat mulai yaitu Al Qur'an. Tujuan dari ilmu tajwid adalah untuk menjaga lidah dari kesalahan membaca Al Qur'an yang meliputi Al Lahnul Jaliy dan Lahnul Khafiy. Kesalahan dalam membaca Al Qur'an (Al Lahn ulama tajwid menyebutnya) secara umum terbagi kepada dua bagian, yaitu Al Lahnul Jaliy dan Al Lahnul Khafiy. Al Lahnul Jaliy berarti kesalahan yang jelas, maksudnya kesalahan-keasalahan dalam membaca Al Qur'an yang dapat diketahui oleh ahli tajwid juga yang lainnya atau dalam keterangan yang lain disebutkan bahwa Al Lahnul Jaliy adalah kesalahan dalam membaca Al Qur'an yang menyalahi kaidah bahasa arab, seperti tertukar huruf, tertukar harakat, huruf pendek dibaca panjang atau sebaliknya. Sedangkan Al Lahnul Khafiy yaitu, kesalahan yang tersembunyi, maksudnya yaitu kesalahan-kesalahan dalam membaca Al Qur'an yang hanya diketahui oleh ahli tajwid tidak yang lainnya, atau dalam keterangan yang lain disebutkan bahwa Al Lahnul Khafiy adalah kesalahan yang lian disebutkan bahwa Al Lahnul Khafiy adalah kesalahan dalam membaca Al Qur'an yang menyalahi kaidah tajwid, seperti bacaan panjang 2 harakat dibaca 4 harakat, bacaan panjang 4 harakat dibaca 2 harakat, tergesa-gesa dalam gunnah.

2.2.4 Android

2.2.4.1 Sejarah Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, *Google Inc*. membeli Android *Inc*. yang merupakan pendatang baru yang membuat piranti lunak untuk ponsel atau *smartphone*. Kemudian untuk mengembangkan Android. Dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.ndroid adalah sebuah sistem operasi untuk perangkat mobile [2].

Pada saat perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan *open source* pada perangkat *mobile*. Di lain pihak, Google merilis kode-kode Android di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan *open platform* perangkat seluler.

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).

Sekitar September 2007 *Google* mengenalkan *Nexus One*, salah satu jenis smartphone yang menggunakan android sebagai sistem operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran pada 5 Januari 2010. Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros Communication, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan Open Handset Alliance, OHA mengumumkan produk perdana mereka, Android, perangkat mobile yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru.

Pada masa saat ini sebagian besar vendor-vendor smartphone sudah memproduksi smartphone berbasis Android, vendor-vendor itu antara lain HTC, Motorola, Samsung, LG, HKC, Huawei, Archos, Webstation Camangi, Dell, Nexus, SciPhone, WayteQ, Sony Ericsson, Acer, Philips, T-Mobile, Nexian, IMO, Asus dan masih banyak lagi vendor smartphone di dunia yang memproduksi android. Hal ini karena android itu adalah sistem operasi yang open source sehingga bebas didistribusikan dan dipakai oleh vendor manapun.

2.2.4.2 The Dalvik Virtual Machine (DVM)

Salah satu elemen kunci dari Android adalah *Dalvik Virtual Machine* (DVM) bukan di *Java Virtual Machine* (JVM), sebenarnya banyak persamaannya

dengan *Java Virtual Machine* (JVM) seperti Java ME (*Java Mobile Edition*), tetapi Android menggunakan *Virtual Machine* sendiri yang dikustomisasi dan dirancang untuk memastikan bahwa beberapa fitur- fitur berjalan lebih efisien pada perangkat *mobile* [2].

Semua *hardware* yang berbasis android dijalankan dengan menggunakan *Virtual Machine* untuk eksekusi aplikasi, pengembang tidak perlu khawatir tentang implementasi perangkat keras tertentu. *Dalvik Virtual Machine* mengeksekusi *executable file,* sebuah format yang dioptimalkan untuk memastikan memori yang digunakan sangat kecil. *The executable file* diciptakan dengan mengubah kelas bahasa *Java* dan dikompilasi menggunakan *tools* yang disediakan dalam SDK Android.

2.2.4.3 Android SDK (Software Development Kit)

Android SDK adalah *tools* API (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* android menggunakan bahasa pemrograman Java. Android merupakan *subset* perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di-*release* oleh Google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API unutk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Sebagai *platform* aplikasi-netral, Android memberi kesempatan untuk membuat aplikasi yang dibutuhkan yang bukan merupakan aplikasi bawaan *Handphone atau Smartphone*. Beberapa fitur-fitur Android yang paling penting adalah [2]:

- *Framework* Aplikasi yang mendukung penggantian komponen dan *reusable.*
- Mesin *Virtual Dalvik* dioptimalkan untuk perangkat *mobile*.
- *Integrated browser* berdasarkan *engine open source WebKit*.
- Grafis yang dioptimalkan dan didukung oleh *libraries* grafis 2D, grafis 3D berdasarkan spesifikasi opengl ES 1,0 (Opsional akselerasi *hardware*)
- SQLite untuk penyimpanan data.
- *Media Support* yang mendukung *audio, video*, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF), GSM *Telephony* (tergantung *hardware*)

- Bluetooth, EDGE, 3G dan WiFi (tergantung *hardware*)
- Kamera, GPS, kompas dan *accelerometer* (tergantung *hardware*)
- Lingkungan *Development* yang lengkap dan kaya termasuk perangkat *emulator*, *tools* untuk *debugging*, profil dan kinerja memori, dan *plugin* untuk IDE *Eclipse*.

2.2.4.4 *ADT* (Android Development Tools)

Android Development Tools (ADT) adalah *plugin* yang didesain untuk IDE *Eclipse* yang memberikan kita kemudahan dalam mengembangkan aplikasi android dengan menggunakan IDE *Eclipse*. Dengan menggunakan ADT untuk *Eclipse* akan memudahkan kita dalam membuat aplikasi *project* android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya, begitu juga kita dapat melakukan *running* aplikasi menggunakan Android SDK melalui *Eclipse*. Dengan ADT juga kita dapat melakukan pembuatan *package* android (.apk) yang digunakan untuk distribusi aplikasi android yang kita rancang.

Mengembangkan aplikasi android dengan menggunakan ADT di *Eclipse* sangat dianjurkan dan sangat mudah untuk memulai mengembangkan aplikasi android[2].

2.2.4.5 Arsitektur Android

Secara garis besar Arsitektur Android dapat dijelaskan dan digambarkan sebagai berikut :

- *Applications* dan *Widgets*

Application dan *Widgets* ini adalah *layer* dimana kita berhubungan dengan aplikasi saja, dimana 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.

- *Applications Frameworks*

Android adalah "*Open Development Platform*" yaitu Android menawarkan kepada pengembang atau member 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 *notification*, dan sebagainya. Pengembang memliki 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 disimpulkan *Applications Frameworks* ini adalah *layer* dimana 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 Provider*

c) *Resource Manager*

d) *Notification Manager*

e) *Activity Manager*

- *Libraries*

Libraries ini adalah *layer* dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya. Berjalan diatas 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 Open GL untuk grafis 2D dan 3D

d) *Libraries SQLite* untuk dukungan *database.*

e) *Libraries SSL* dan *WebKit* terintegrasi dengan *web browser* dan *security.*

f) *Libraries LiveWebcore* mencakup modern *web browser* dengan *engine embedded web view.*

g) *Libraries* 3D yang mencakup implementasi OpenGL ES 1.0 API's.

- *Android Run Time*

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

a) *Core Libraries*

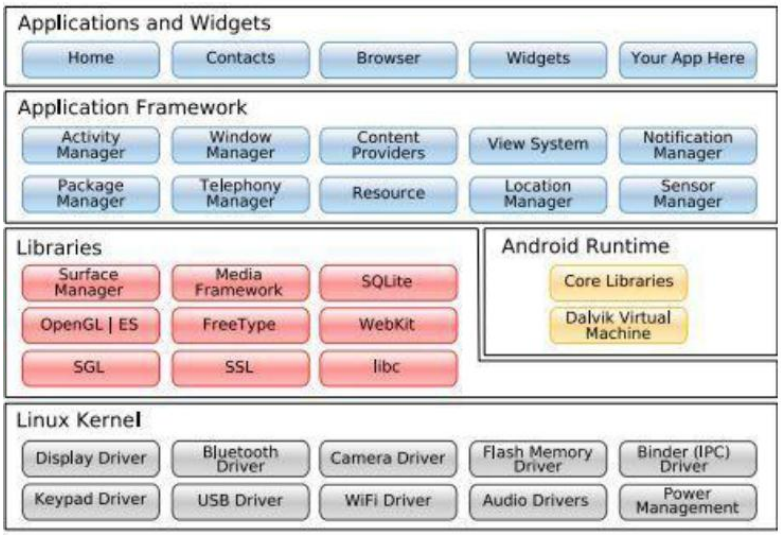
Aplikasi Android dibangun dalam bahasa Java, sementara *Dalvik* sebagai virtual mesinnya bukan Virtual Machine Java, sehingga diperlukan sebuah *libraries* yang berfungsi untuk menterjemahkan bahasa java/c yang ditangani oleh *Core Libraries.*

b) *Dalvik Virtual Machine*

Virtual mesin berbasis register yang dioptimalkan untuk menjalankan fungsi-fungsi secara efisien, dimana merupakan pengembangan yang mampu membuat linux kernel untuk melakukan *threading* dan manajemen tingkat rendah.

- *Linux Kernel*

Linux kernel adalah *layer* dimana inti dari *operating* sistem dari Android itu berada. Berisi *file-file system* yang mengatur sistem *processing, memory, resource, drivers,* dan sistem - sistem operasi android lainnya.



Gambar 2. 2 Arsitektur Android [2]

2.2.4.6 Versi Android

Telepon pertama yang memakai system operasi android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2010 diperkirakan hamper semua vendir seluler di dunia menggunakan anfrois sebagai *operating system*, adapun versi – versi android adalah sebagai berikut [2] :

1. **Android versi 1.1**

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam *alarm*, (pencarian suara), pengiriman pesan dengan *Gmail*, dan pemberitahuan *email*.

2. **Android versi 1.5 (Cupcake)**

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (*Software Development Kit*) dengan versi 1.5 (Cupcake). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton *video* dengan modus kamera, mengunggah *video* ke *Youtube* dan gambar ke *Picasa* langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke *headset* Bluetooth, animasi layar, dan *keyboard* pada layar yang dapat disesuaikan dengan sistem.

3. **Android versi 1.6 (Donut)**

Donut (versi 1.6) dirilis pada bulan September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol *applet* VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, *camcorder* dan galeri yang dintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan Text-to-*speech engine*; kemampuan dial kontak; teknologi *text to change speech* (tidak tersedia pada semua ponsel; pengadaan resolusi VWGA.

4. **Android versi 2.0/2.1 (Eclair)**

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah pengoptimalan *hardware*, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 MP, *digital Zoom*, dan Bluetooth 2.1. Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi *mobile* terbaik (*killer apps* - aplikasi unggulan). Kompetisi ini berhadiah $25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik. Dengan semakin berkembangnya dan semakin bertambahnya jumlah *handset* Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug. Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

5. **Android versi 2.2 (Froyo: Frozen Yoghurt)**

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, intergrasi V8 JavaScript *engine* yang dipakai Google Chrome yang mempercepat kemampuan *rendering* pada *browser*, pemasangan aplikasi dalam SD Card,

kemampuan WiFi Hotspot *portabel*, dan kemampuan *auto update* dalam aplikasi Android Market.

6. **Android versi 2.3 (Gingerbread)**

Pada 6 Desember 2010, Android versi 2.3 (Gingerbread) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (*gaming*), peningkatan fungsi *copy paste*, layar antar muka (*User Interface*) didesain ulang, dukungan format *video* VP8 dan WebM, efek *audio* baru (*reverb, equalization, headphone virtualization*, dan *bass boost*), dukungan kemampuan Near Field Communication (NFC), dan dukungan jumlah kamera yang lebih dari satu.

7. **Android versi 3.0/3.1 (Honeycomb)**

Android Honeycomb dirancang khusus untuk *tablet*. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada Honeycomb juga berbeda karena sudah didesain untuk *tablet*. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (*hardware*) untuk grafis. *Tablet* pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom. Perangkat *tablet* dengan *platform* Android 3.0 akan segera hadir di Indonesia. Perangkat tersebut bernama Eee Pad Transformer produksi dari Asus. Rencana masuk pasar Indonesia pada Mei 2011.

8. **Android versi 4.0 (ICS: Ice Cream Sandwich)**

Diumumkan pada tanggal 19 Oktober 2011, membawa fitur Honeycomb untuk *smartphone* dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari *email* secara *offline*, dan berbagi informasi dengan menggunakan NFC. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus.

9. **Android versi 4.1 (Jelly Bean)**

Android Jelly Bean yang diluncurkan pada acara Google I/O lalu membawa sejumlah keunggulan dan fitur baru. Penambahan baru diantaranya meningkatkan *input keyboard*, desain baru fitur pencarian, UI yang baru dan pencarian melalui *Voice Search* yang lebih cepat. Tak ketinggalan Google Now juga menjadi bagian

yang diperbarui. Google Now memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi Android Jelly Bean 4.1 muncul pertama kali dalam produk *tablet* Asus, yakni Google Nexus 7.

10. **Android versi 4.2 (Jelly Bean)**

Fitur *photo sphere* untuk panaroma, *daydream* sebagai *screensaver*, *power control, lock screen widget*, menjalankan banyak *user* (dalam *tablet* saja), *widget* terbaru. Android 4.2 Pertama kali dikenalkan melalui LG, yakni Google Nexus 4.

11. **Android 4.4 KitKat**

Android 4.4 KitKat API level 19. Google mengumumkan Android 4.4 KitKat (dinamai dengan izin dari Nestle dan Hershey) pada 3 September 2013, dengan tanggal rilis 31 Oktober 2013. Kitkat merupakan merk sebuah cokelat yang dikeluarkan Nestle. Rilis berikutnya setelah KitKat diperkirakan banyak pengamat akan diberi nomor 5.0 dan dinamai 'Key Lime Pie'.

2.2.5 Konsep Perancangan Berorientasi Objek

Teknologi objek menganalogikan sistem aplikasi seperti kehidupan nyata yang di dominasi oleh objek. Di dalam membangun sistem berorientasi objek akan menjadi lebih baik apabila langkah awalnya didahului dengan proses analisis dan perancangan yang berorientasi objek. Tujuannya adalah mempermudah programmer dalam mendesain program dalam bentuk objek-objek dan hubungan antar objek tersebut untuk kemudian dimodelkan dalam sistem nyata. Suatu perusahaan *software* yaitu *Rational Software*, telah membentuk konsorsium dengan berbagai organisasi untuk meresmikan pemakaian *Unifed Modelling Language* (UML) sebagai bahasa standar dalam *Object Oriented Analysist Design* (OOAD).

2.2.5.1 *UML (Unified Modeling Language)*

Unified Modelling Language (UML) adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual juga merupakan satu kumpulan

konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem *software* yang terkait dengan objek [4].

2.2.5.2 *Use Case Diagram*

Use Case Diagram menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (aktor). Diagram ini menunjukan fungsionalitas suatu sistem yang berinteraksi dengan dunia luar. Use Case Diagram dapat digunakan selama proses analisis untuk menangkap requirement sistem dan untuk memahami bagaimana sistem bekerja.

2.2.5.3 *Class Diagram*

Class Diagram menjelaskan dalam visualisasi struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Class Diagram memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas dalam model desain dari suatu sistem.

Selama proses analisis, class diagram memperlihatkan aturan- aturan dan tanggung jawab entitas yang menentukan prilaku sistem. Selama tahap desain, class diagram berperan dalam menagkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat.

2.2.5.4 *Behaviour Diagram*

Behaviour diagram dapat dikelompokkan menjadi tiga diagram, yaitu :

1. *Statechart Diagram*

Statechart Diagram berfungsi untuk memodelkan perilaku dinamis satu kelas satu objek.

2. *Activity Diagram*

Activity diagram memodelkan alur kerja (*workflow*) sebuah bisnis dan urutan aktifitas dalam suatu proses.

3. *Interaction Diagram*

Interaksi diagram berfokus pada gambaran aliran kendali interaksi dimana node adalah interaksi atau kejadian interaksi.

2.2.5.5 Implementasi Diagram

Implementation diagram dibagi menjadi dua diagram, yaitu :

1. *Component diagram*, menggambarkan alokasi semua kelas dan objek kedalam komponen-komponen dalam desain fisik sistem *software*. Diagram ini memperlihatkan pengaturan dan kebergantungan antara komponen-komponen *software*, seperti *source code*, *binary code*, dan komponen tereksekusi (execute components).
2. *Deployment diagram*, memperlihatkan pemetaan *software* kepada *hardware*. Dimana akan berjalan (di server/multitier, standalone, atau lainnya) dan menggambarkan model koneksi dan kemampuan jaringan dan hal lainnya yang bersifat fisik.

2.2.6 Analisis Basis Data

2.2.6.1 Pengertian *Database*

Database merupakan himpunan sekelompok data yang saling berkaitan, yang diolah dan diorganisasikan sehingga menjadi sebuah informasi yang berguna.

2.2.6.2 *Entity Relationship Diagram*

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol [11].

KomponenEntity Relationship Diagram:[11]

1. Entity

Entity adalah sesuatu yang terdapat di dalam sistem, baik nyata maupun abstrak yang tersimpan di dalam data. Entitas diberi nama dengan kata benda dan dapat dikelompokkan dalam empat jenis, yaitu : orang,benda, lokasi, dan kejadian.

2. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasikan isi elemen satu dengan yang lain.

3. Relationship

Relationship adalah hubungan alamiah yang terjadi antara entitas. Hubungan entitas ini akan menyangkut dua komponen yang menyatakan jalinan ikatan yang terjadi.

Komponen dasar model merupakan diagram entity-relationship yang digunakan untuk menyajikan objek data secara visual. Antar entitas selalu ada tiga hubungan biner, yaitu :[11]

1. Satu ke Satu (*One to One Relationship*)

Hubungan antara *entity* pertama dengan *entity* kedua adalah satu berbanding satu.



Gambar 2.3 ERD *One to One*

2. Satu ke Banyak (*One to Many*)

Hubungan *entity* pertama dapat berhubungan satu atau lebih pada himpunan kedua, tetapi tidak sebaliknya.



Gambar 2.4 ERD *One to Many*

3. Banyak ke Banyak (*Many to Many*)

Setiap *entity* pada himpunan *entity* pertama dapat berhubungan dengan banyakentitypada himpunan *entity* kedua, begitu pula sebaliknya.



Gambar 2.5 ERD *Many to Many*

2.2.6.3 Kamus Data

Kamus data (*data dictionary*) adalah daftar organisasi semua elemen yang ada dalam sistem secara lengkap dengan definisi yang baku sehingga Member dan analisis sistem akan memiliki pengertian yang sama untuk *input*, *output*, komponen penyimpanan dan perhitungannya. Kamus data dapat digunakan pada saat analisis sitem atau perancangan sistem, kamus data digunakan untuk mencatat terminologi bisnis, aturan standar (batasan panjang karakter, nilai,*system field*).

2.2.7 *Dreamweaver*

Macromedia Dreamweaver adalah sebuah HTML editor profesional untuk mendesain secara visual dan mengelola situs web maupun halaman web. Bilamana kita menyukai untuk berurusan dengan kode-kode HTML secara manual atau lebih menyukai bekerja dengan lingkungan secara visual dalam melakukan editing, Dreamweaver mambuatnya menjadi lebih mudah dengan menyediakan tool-tool yang sangat berguna dalam peningkatan kemampuan dan pengalaman dalam mendesain web.

2.2.8 *MySQL*

MySQL merupakan *multiuserdatabase* yang menggunakan bahasa *Structured Query Language* (SQL)[12]. Pada aplikasi yang dibangun, MySQL digunakan untuk menyimpan dan memproses data yang telah dilakukan oleh

aplikasi, sehingga dapat dilakukan penambahan, penghapusan, pengubahan, dan pencarian data yang telah tersimpant di dalam MySQL tersebut.

2.2.9 Konversi Sinyal Analog ke Digital

Sinyal yang terdapat pada kehidupan sehari-hari disebut dengan sinyal analog. Untuk dapat memproses sinyal analog tersebut, maka harus diubah terlebih dahulu menjadi sinyal digital atau sinyal diskrit. Sinyal analog memiliki sifat kontinu pada domain waktu dan amplitudo, sedangkan sinyal digital memiliki sifat diskrit pada domain waktu dan amplitudo. Untuk mengubah sinyal analog menjadi sinyal digital maka diperluas proses "digitalisasi". Proses ini terdiri dari 3 tahap, yaitu :

1. *Sampling*

Sampling mengambil bagian sinyal analog yang kontinu dan membagi menjadi beberapa bagian diskrit dalam domain waktu yang dilakukan dengan cara periodik menganalisis sinyal analog dengan menggunakan frekuensi sampling tertentu. Untuk mengetahui frekuensi sampling yang digunakan, sesuai dengan teori Nyquist, frekuensi sampling sebaiknya 2 kali dari frekuensi tertinggi dari sinyal yang akan di analisis. Perusahaan telepon sebagai contoh menggunakan frekuensi sampling sebesar 8kHz sebab suara manusia berkisar antara 3-4kHz. Dengan adanya frekuensi sampling ini maka sinyal diatas 4kHz akan dibuang.

Rumusnya adalah :

$$X = FS\ x\ dt(detik)\ x\ \left(\frac{bit}{sample}\right) x\ j \qquad \ldots(2.1)$$

Keterangan :

FS : Frekuensi sampling

bit : Jumlah bit yang ditentukan

j : Channel mono (bernilai 1) atau stereo (bernilai 2)

2. *Quantization*

Kuantisasi akan mengubah sinyal analog menjadi bentuk digital. Tahap ini akan mengambil suatu nilai tertentu untuk mengubah bentuk digital. Sebagai

contoh pada perusahaan telepon membagi sinyal analog menjadi 256 bagian. Hal ini diperlukan sebab sinyal analog memiliki kemungkinan nilai yang tidak terbatas.

3. Coding

Coding mengubah sinyal digital menjadi angka biner. Setelah sinyal disampling, misal menggunakan frekuensi sampling 8kHz, maka angka biner yang dihasilkan memiliki jangkauan antara 1 sampai dengan 256 (28 = 256). Dengan menggunakan frekuensi sampling yang semakin besar maka akan didapatkan tingkat akurasi atau ketelitian yang lebih tinggi.

2.2.10 Algoritma Fast Fourier Transform

Algoritma *fast fourier transform* merupakan suatu algoritma yang sering dipakai untuk menganalisa spektrum frekuensi sinyal analog ataupun data digital. Selain itu algoritma *fast fourier transform* merupakan salah satu metode untuk transformasi sinyal suara menjadi sinyal frekuensi [5].

FFT (*Fast Fourier Transform*) adalah teknik perhitungan cepat dari DFT (*Discrete Fourier Transform*). Transformasi fourier diskrit atau disebut dengan *Discrete Fourier Transform* (DFT) adalah model transformasi fourier yang dikenakan pada fungsi diskrit, dan hasilnya juga diskrit. FFT adalah DFT dengan teknik perhitungan yang cepat dan memanfaatkan sifat periodikal (konjugasi) dari transformasi *fourier*. Perhatikan rumus dari DFT :

$$F(u) = \frac{1}{N}\sum_{n=0}^{N-1} f(x) \exp[-\frac{2j\pi\mu x}{N}] \quad \ldots(2.2)$$

$$F(u) = \frac{1}{N}\sum_{n=0}^{N-1} f(x) \left(\cos\left(\frac{2\pi ux}{N}\right) - j \sin\ (\frac{2\pi ux}{N})\right) \quad \ldots(2.3)$$

Keterangan :

N : Jumlah data

$f(x)$: Nilai diskrit

π : Phi (3,14)

j : Channel mono (bernilai 1) atau stereo (bernilai 2)

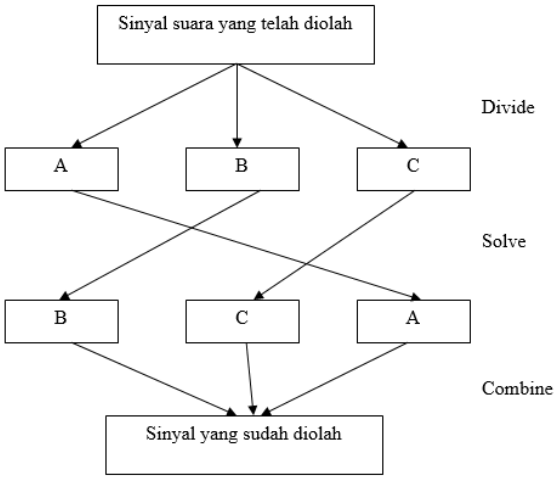
2.2.11 Algoritma Divide and Conquer

Algoritma *Divide and Conquer* merupakan algoritma yang sangat populer di dunia ilmu komputer. *Divide and Con*quer merupakan algoritma yang berprinsip memecah-mecah permasalahan yang terlalu besar menjadi beberapa bagian kecil sehingga mudah untuk diselesaikan. Langkah-langkah umum algoritma *Divide and Conquer* :

- *Divide* : membagi masalah menjadi beberapa sub-masalah yang memiliki kemiripan dengan masalah semula namun berukuran lebih kecil (idealnya berukuran hampir sama).

- *Conque* : memecahkan atau menyelesaikan masing-masing sub-masalah secara rekursif.

- *Combine* : menggabungkan solusi masing-masing sub-masalah sehingga membentuk solusi masalah semula.



Gambar 2. 3 Pengolahan Algoritma Divide and Conquer

Objek masalah yang dibagi adalah masukan (*input*) atau *instances* yang berukuran n: tabel (*array*), matriks, dan sebagainya, bergantung pada masalahnya. Tiap-tiap sub-masalah mempunyai karakteristik yang sama (*the same type*) dengan karakteristik masalah asal, sehingga metode *Divide and Conquer* lebih natural diungkapkan dalam skema rekursif. Sesuai dengan karakteristik pembagian dan pemecahan masalah tersebut, maka algoritma ini dapat berjalan

baik pada persoalan yang bertipe rekursif (perulangan dengan memanggil dirinya sendiri). Dengan demikian, algoritma ini dapat diimplementasikan dengan cara iteratif (perulangan biasa), karena pada prinsipnya iteratif hampir sama dengan rekursif. Salah satu penggunaan algoritma ini yang paling populer adalah dalam hal pengolahan data yang bertipe *array* (elemen larik). Karena pada umumnya pengolahan *array* pada umumnya selalu menggunakan prinsip rekursif atau iteratif.

Berikut skema umum algoritma *Divide And Conquer* :

```
procedure DIVIDE_n_CONQUER(input n : integer)
{Masukan: masukan yang berukuran n
Keluaran: solusi dari maslah semula
}
Deklarasi
r, k : integer
Algoritma
if n < n0 then {masalah sudah cukup kecil}
SOLVE sub-masalah yang berukuran n ini
else
Bagi menjadi r sub-masalah, masingmasing
berukuran n/k
for masing-masing dari r sub-masalah
do
DIVIDE_n_CONQUER(n/k)
endfor
COMBINE solusi dari r sub-masalah
menjadi solusi masalah semula
endif
```

Gambar 2. 4 Skema Umum Algoritma Divide and Conquer

Proses *Divide and Conquer* :

1. Bagi dua proses pecocokan menjadi beberapa sub-masalah.
2. Urutkan data fitur suara, bagi dua data fitur suara, ambil bagian yang terkecil atau terbesar dari data yang masuk.

3. Untuk setiap elemen data pada *template*, dicocokan dengan data baru menggunakan *binary search Divide and Conquer*.
4. Setiap sub-masalah disatukan kembali dan dianalisis, kecocokan dari segi data fitur dan apakah data yang diucapkan sesuai dengan kata yang tersedia pada *template* data.
5. Hitung nilai peramalan *error* terkecil, yang terkecil adalah solusi atau *output* yang dihasilkan.

Verifikasi data, jika sesuai maka akan masuk ke proses selanjutnya, sesuai dengan aplikasi yang mengimplementasikan algoritma ini.