

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Program

Dalam penyusunan Tugas Akhir ini yang berkaitan dengan pembuatan suatu aplikasi sistem pakar di dalam suatu *smartphone* berbasis android yang berguna untuk membantu dokter dalam mengatasi sebuah penyakit yang terkena akibat paparan radiasi pada *handphone* pada kesehatan. Untuk memudahkan dalam memahami, penulis menggunakan referensi berbeda yang ditemukan di daftar pustaka. Berikut adalah teori pendukung yang dapat memperkuat penulisan Tugas Akhir ini.



2.1.1. Sistem

Sistem adalah kumpulan dari banyak elemen yang berinteraksi untuk mencapai suatu tujuan tertentu pada komponennya. Pada prosedurnya sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan dan berkumpul bersama melakukan suatu kegiatan untuk menyelesaikan sasaran tertentu, menurut Andalia dan Budi Setiawan dalam (Kiswoyo, 2015), “sistem adalah himpunan suatu komponen atau *variable* yang terdiri dari beberapa bagian yang saling terkait, terhubung, ketergantungan dan saling mendukung secara keseluruhan bersatu dalam suatu kesatuan untuk mencapai tujuan tertentu”. Menurut Lubis dalam (Kiswoyo, 2015), “sistem adalah suatu kesatuan yang terdiri dari suatu komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi atau materi untuk mencapai suatu tujuan”.

Untuk dapat mengetahui apakah segala sesuatu itu dianggap sistem maka harus mencakup lima unsur utama yakni sebagai berikut:

- a. Adanya kumpulan objek (*The existence of a collection of objects*).
- b. Berada pada suatu lingkungan yang utuh dan kompleks (*Being in an intact and complex environment*).
- c. Terdapat tujuan bersama (*Output*) sebagai hasil akhirnya.

2.1.2. Pakar

Menurut Purnia dalam (Suleman, Fitriana, & Putra, 2018), pengertian pakar adalah “individu yang memiliki pengetahuan khusus, pengalaman, dan metode-metode untuk menyelesaikan masalah dalam bidang tertentu”.

2.1.3. Sistem Pakar

Menurut Hayadi dalam (Sulistiyono, 2018), “Sistem pakar adalah suatu aplikasi komputer yang ditujukan untuk membantu pengambilan keputusan atau pemecahan persoalan dalam bidang yang spesifik”. Sistem pakar memanfaatkan secara maksimal pengetahuan khusus selayaknya seorang pakar untuk memecahkan masalah. Pakar atau ahli (*ekspert*) didefinisikan sebagai seorang yang memiliki pengetahuan atau keahlian khusus yang tidak dimiliki oleh kebanyakan orang. Seorang pakar dapat memecahkan masalah yang tidak mampu dipecahkan kebanyakan orang. Dengan kata lain dapat memecahkan masalah dengan efisien. Pengetahuan yang dimuat kedalam sistem pakar dapat berasal dari seorang pakar atau pengetahuan yang berasal dari buku, jurnal, majalah dan dokumentasi yang dipublikasikan lainnya, serta orang yang memiliki pengetahuan meskipun bukan ahli. Menurut Widayanto dalam (Sulistiyono, 2018), “istilah sistem pakar (*expert*

system), sering disinonimkan dengan sistem berbasis pengetahuan (*Knowledge based system*) atau sistem pakar berbasis pengetahuan (*Knowledge based expert system*)”.

Tabel.II.1

Perbedaan Pakar Manusia Dengan Sistem Pakar

Pakar Manusia	Sistem Pakar
Terbatas waktu karena manusia membutuhkan istirahat	Tidak terbatas karena dapat digunakan kapanpun juga
Tempat akses bersifat lokal pada suatu tempat saja dimana pakar berada	Dapat digunakan di berbagai tempat
Pengetahuan bersifat variable dan dapat berubah-ubah tergantung situasi	Pengetahuan bersifat konsisten
Kecepatan untuk menemukan solusi sifatnya bervariasi	Kecepatan untuk memberikan solusi konsisten dan lebih cepat daripada manusia
Biaya yang harus dibayar untuk konsultasi biasanya sangat mahal	Biaya yang dikeluarkan lebih murah

Sumber: (Ma'rifati & Kesuma, 2018)

2.1.4. Aplikasi

Menurut Kamus Besar Bahasa Indonesia dalam Juansyah (Widayanto & Refianti, 2018) ”Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu”. Dengan adanya

aplikasi pengguna dapat melakukan berbagai aktifitas seperti, melakukan pengolahan data, hiburan, belajar dan aktifitas yang lain.

Menurut Abdurrahman dan Riswaya dalam (Riyowati & Fadlilah, 2019) ”Aplikasi merupakan program yang siap dipakai dan digunakan untuk menjalankan perintah *user* dengan tujuan untuk mendapatkan hasil yang tepat dan akurat sesuai dengan tujuan dari pembuatan aplikasi tersebut”. Aplikasi merupakan salah satu proses pemecahan salah satu masalah yang menggunakan teknik komputasi yang diinginkan dalam pemrosesan data.

2.1.5. Android

Menurut Safaat dalam (Ariyanto, 2018), “*Android* adalah sebuah kumpulan perangkat lunak untuk perangkat *mobile* yang mencakup sistem operasi, *middleware* dan aplikasi utama *mobile*”.

Menurut Ii dalam (Ariyanto, 2018), “*Android* merupakan sebuah sistem operasi telepon seluler dan komputer tablet layar sentuh (*touch screen*) yang berbasis linux”. Namun seiring perkembangannya, *android* berubah menjadi *platform* yang begitu cepat dalam melakukan inovasi. Hal ini tidak lepas dari pengembangan utama dibelakangnya yaitu google. *Google* lah yang mengakuisi *android*, yang kemudian membuatkan sebuah *platform*.

1. Sejarah Android

Pada awal mulanya, *android Inc* melakukan sebuah perusahaan *software* kecil yang didirikan pada bulan Oktober 2003 di Palo Alto, California, USA. Didirikan oleh beberapa senior perusahaan yang berbasis IT & Communication; Andy Rubun, Rich Miner, Nick Sears, dan Chirs White. *Android INC* didirikan untuk

mewujudkan *mobile device* yang lebih peka terhadap lokasi dan prefensi pemilik. Dengan kata lain, *android* INC ingin mewujudkan *mobile device* yang lebih mengerti pemiliknya.

2. Versi Android Android selalu mengeluarkan versi terbaru dari *Operating System* ini. Berikut ini merupakan versi-versi android menurut API:

a. Android versi 1.1

Pada 9 maret 2009, untuk pertama kalinya Google merilis android versi 1.1 sebagai versi yang dapat digunakan. Android versi ini dilengkapi dengan pembaruan statis pada aplikasi, jam alarm, *voice search* (pencarian suara), pengiriman pesan dengan gmail, dan pemberitahuan email.

b. 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 untuk 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.

c. Android versi 1.6 (*Donut*)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibandingkan sebelumnya, menggunakan baterai indikator dan control *applet* VPN. Fitur lainnya adalah galeri yang

memungkinkan pengguna untuk memilih foto yang dihapus, kamera, *camrecorder* dan galeri yang diintegrasikan, CDMA atau EVDO, 802.1x, VPN, Gestures, dan *Text-to-speech engine*, kemampuan dual kontak, teknologi *textto change sppech* (tidak tersedia pada semua ponsel, pengadaan resolusi VWGA).

d. Android versi 2.0 atau 2.1 (*eclair*)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0 atau 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.

e. 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 0.1*, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi *V8 JavaScriptengine* yang dipakai *Google Crome* yang mempercepat kemampuan *rendering* pada *browser*, pemasangan dalam aplikasi SD Card, kemampuan *Wifi Hotspot Portable*, dan kemampuan *auto update* dalam aplikasi *Android Market*.

f. 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*,

Headphonevirtualization, dan *bassboost*), dukungan kemampuan *Near Field Communication* (NFC), dan dukungan jumlah kamera yang lebih dari satu.

g. Android versi 3.0 atau 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-processor* dan juga akselerasi perangkat keras (*Hardware*) untuk grafis. *Tablet* pertama yang dibuat dengan menjalankan *Honeycomb* adalah *Motorola Xoom*.

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

Diumumkan pada tanggal 10 Oktober 2011, membawa *future Honeycomb* untuk *Smartphone* dan menambahkan fitur baru termasuk membukaunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan *fitografi*, mencari *email* secara *offline*, dan berbagi informasi dengan menggunakan NFC.



Fitur yang tersedia di Android adalah:

1. Kerangka Aplikasi: itu memungkinkan penggunaan dan penghapusan komponen yang tersedia.
2. *Dalvik Mesin Virtual*: mesin *virtual* dioptimalkan untuk perangkat mobile.
3. Grafik: grafik di 2D dan grafis 3D berdasarkan pustaka OpenGL.
4. Mendukung Media: *audio*, *video*, dan berbagai format gambar (MPEGA, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
5. GSM, Bluetooth, EDGE, 3G, Wifi (*Hardware dependent*).

6. Kamera, *Global positioning System* (GPS), Kompas, dan *accelerometer* (tergantung *Hardware*).

i. Android versi 4.1 atau 4.2 (*Jelly bean*)

Di tahun 2012 Android versi 4.1 atau 4.2 diluncurkan dengan nama Android *Jelly bean*. Pada versi ini terdapat peningkatan kemampuan diantaranya kemampuan *on-screen keyboard* yang lebih cepat dan *responsive*, penambahan *fitur voice search* untuk pencarian kontak serta memungkinkan pengguna mengulir (*scroll*) cepat *home screen* ke bawah untuk melihat informasi penting.

j. Android versi 4.4 (*Kitkat*)

Butuh satu tahun lagi Google untuk mengkadirkan versi lainnya yaitu versi 4.4 atau *kitkat*. Versi *android* ini memberikan suasana segar dengan pembaharuan antarmuka beraksen putih dan biru muda. Pada *kitkat*, Google menghadirkan pencarian menggunakan suara atau disebut “OK, Google”. Disaat bersamaan, *google* juga meluncurkan aplikasi pesan singkat *hangouts* untuk pertama kalinya.

k. Android versi 5.0 (*Lolipop*)

Pada tanggal 15 Oktober 2014 hadirnya versi *Android* 5.0 atau versi *Lolipop*. Pembaharuan yang mencolok pada *Lolipop* tampak dari sisi desainnya yang diperhalus dan disesuaikan dengan zaman. Inovasi kurang terasa pada versi ini. Satu-satunya yang lumayan baru adalah dukungan untuk gambar berformat RAW.

l. Android versi 6.0 (*Marsmallow*)

Pada tanggal 19 Agustus 2015 *google* meluncurkan versi *android* lainnya yaitu versi 6.0 atau versi Android *Marsmallow*. Menu aplikasi pada Android

Marshmallow benar-benar dibuat baru. Desainnya membuat pengguna merasa naik kelas dari versi sebelumnya karena lebih dinamis. Selain itu ada juga *fitur memory manager* yang memungkinkan pengguna mengecek penggunaan setiap memori pada tiap aplikasi. Pembaharuan kedua dari pengaturan *volume* yang dapat dikontrol berbeda-beda pada panggilan, media, alarm, dan juga keamanan pada versi ini sudah menyelamatkan sensor pemindai sidik jari.

m. Android versi 7.0 (*Nougat*)

Setelah melakukan banyak update versi-versi Android, tanggal 22 Agustus 2016 diluncurkan versi yang paling terbaru dari Android versi 7.0 dengan nama versi *Nougat*. Pembaharuan yang paling mendasar pada versi ini adalah kehadiran *Google asisten* yang menggantikan *Google Now*. Fitur-fitur baru lainnya mencakup layar *layer Split-screen* saat dipakai *multitasking*, serta dukungan terhadap *platform virtual reality* terbaru *Google*.



n. Android versi 8.0 (*Oreo*)

Pada bulan Agustus 2017, *google* merilis sistem operasi seluler *Android 8.0 Oreo*. Hanya sedikit perangkat yang mendapat pembaruan dan yang lain akan segera menyusul. *Android Oreo* adalah versi utama kedelapan dari sistem operasi *Android*. Ini pertama kali dirilis pada Maret 2017 dan dirilis secara publik pada 21 Agustus 2017. Tepat sebelum *android 8.0 Oreo*, *Android 7.1.2 Nougat* adalah versi terakhir dan masih sedikit perangkat yang mendukungnya. Setelah meluncurkan pembaruan *Android 8*, *google* juga merilis versi 8.1, yang menjanjikan untuk meningkatkan kinerja pada perangkat dengan RAM 1GB atau kurang. Selain itu, *API Neural Network*

terbaru juga ditambahkan untuk mempercepat kecerdasan perangkat. Di perangkat *pixel 2*, anda juga akan mengalami peningkatan kualitas fotografi HDR.

o. *Android* versi 9.0 (Pie)

OS *android* versi kesembilan ini secara resmi dirilis pada 6 Agustus 2018, *android P* dikenal dengan nama “*Android Pie*”. OS *android* terbaru ini pertama kali dirilis untuk perangkat atau *smartphone google pixel*. Fitur menarik dari *android p* ini adalah *Adaptive Brightness*, fitur ini akan secara otomatis menurunkan kecerahan layar di ruang yang memiliki minim cahaya, berbeda dengan *auto adaptive* yang mengatur otomatis kecerahan layar, fitur *Adaptive Brightness* ini bisa kita atur kecerahan layar setinggi kita inginkan, ketika di ruang tertentu atau di waktu malam, fitur *Adaptive Brightness* ini akan otomatis menurunkan kecerahan layar hp *android* kita gunakan. Sudah ada fitur *Edge-to-edge screens* yang support ponsel layar 18.9. Seperti saya jelaskan di awal, versi *android* ini dikembangkan lebih focus untuk ponsel yang memiliki desain *baseless*.

3. Android SDK

Android SDK adalah tools bagi para programmer yang ingin mengembangkan aplikasi berbasis Android. Android SDK mencakup seperangkat alat pengembangan yang kompherensif. Android SDK terdiri dari *debugger*, *libraris*, *handset emulator*, dokumentasi, contoh kode, dan tutorial.

4. Arsitektur Android

Menurut Anwar, Nugroho, dan Lestariningsih dalam (Sulistiyono, 2018), secara garis besar arsitektur Android dapat dijelaskan sebagai berikut:

a. Application dan Widgets

Layer yang berhubungan dengan aplikasi saja, dimana biasanya aplikasi yang didownload kemudian diinstalasi dan jalankan aplikasi tersebut.

b. Application Frameworks

Layer dimana para pembuat aplikasi melakukan pengembangan 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.

c. Libraries

Layer dimana fitur-fitur android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasinya.



d. Android Run Time

Layer yang membuat aplikasi Android dapat dijalankan dimana dalam prosesnya menggunakan Implementasi linux.

e. Linux Kernel

Layer dimana inti dari sistem operasi Android itu berada . Berisi file-file sistem yang mengatur sistem *processing*, *memory*, *resource*, *drivers*, dan sistem-sistem operasi Android lainnya.

2.2. Teori Pendukung (*tools system*)

Dalam penulisan Tugas Akhir ini penulis juga menggunakan beberapa teori untuk melengkapi penulisan Tugas Akhir ini. Berikut adalah teori pendukung yang penulis gunakan:

2.2.1. SQLite Database

Menurut Wahana Komputer dalam Vadlya (Rahmanto, 2019) “SQLite adalah sebuah sistem manajemen basis data rasional yang berbasis data rasional yang bersifat *ACID-complaint* dan memiliki ukuran *library* yang *relative* kecil. *Software* ini ditulis dengan menggunakan bahasa C. Pengguna tidak memerlukan *setup database* dalam menggunakan *database SQLite*. Pengguna hanya perlu mendefinisikan *SQL statement* untuk membuat dan mengupdate *database*. Setelah itu *database* otomatis diatur oleh *platformAndroid* untuk pengguna.

2.2.2. Java

Menurut (Barri, Lumenta, & Wowor, 2015), Pengertian *java* adalah “bahasa pemrograman yang dapat dijalankan berbagai komputer termasuk telepon genggam”.

Bahasa pemrograman *java* ini awalnya dibuat oleh James Gosling saat masih bergabung di *Sun Microsystem* saat ini merupakan bagian dari *Oracle* dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin atas bawah yang minimal.

Java adalah bahasa pemrograman yang multi *platform* dan multi *device*. Sekali anda menuliskan sebuah program dengan *java*, anda dapat menjalankan hampir semua komputer dan perangkat lain yang mendukung *java*, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodenya. Aplikasi dengan

berbasis *java* ini dikompilasikan ke dalam *p-code* dan bisa dijalankan dengan *java virtual machine*. Fungsionalitas dari *java* ini dapat berjalan dengan *platform* sistem operasi yang berbeda karena sifatnya yang umum dan non-spesifik.

2.2.3. HTML

Menurut (Josi, 2017), pengertian HTML adalah “singkatan dari *Hyper Text Markup Language*, yaitu skrip yang berupa tag-tag untuk membuat dan mengatur struktur *website*”.

Beberapa tugas HTML, dalam membangun *website* diantaranya sebagai berikut:

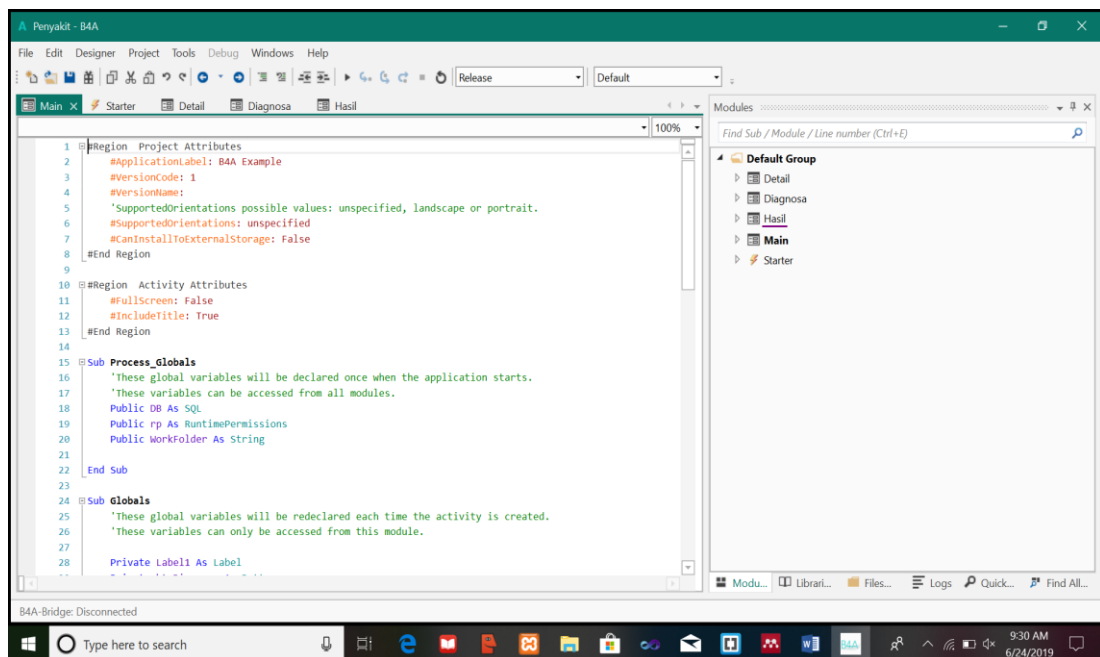
- a. Menentukan *layout website*.
- b. Memformat text dasar seperti pengaturan paragraf, dan format *font*.
- c. Membuat *list*.
- d. Membuat tabel.
- e. Menyisipkan gambar, *video*, dan *audio*.
- f. Membuat *link*.
- g. Membuat formulir.



2.2.4. B4A (Basic4 Android)

Bahasa B4A mirip dengan bahasa *Visual Basic*, namun perbedaannya aplikasi B4A ini didukung tambahan untuk objek. Aplikasi B4A mencakup semua fitur yang dibutuhkan untuk mengembangkan jenis aplikasi Android dengan cepat. Aplikasi B4A digunakan oleh puluhan ribu pengembang dari seluruh dunia, termasuk perusahaan seperti NASA, HP, IBM dan lainnya. Menurut Vadliya Maarif dkk dalam

(Sulistiyono, 2018), “B4A adalah *Development tool* sederhana yang kuat untuk membangun aplikasi *Android*”. B4A mirip dengan bahasa *visual basic* dengan tambahan dukungan objek.



Sumber:Aplikasi B4A

Gambar. II.1. Tampilan Aplikasi B4A

Komponen-komponen yang terdapat pada *basic-Android* antara lain:

1. Menu dan Toolbar

Terdiri dari beberapa menu dan tombol IDE. Untuk *Toolbar*, fungsi dari masing-masing tombol adalah:

- a *New* : membuat *project* baru
- b *Open* : membuka *project* yang sebelumnya disimpan dalam penyimpanan

- c *Save* : menyimpan *project* yang sedang dikerjakan
- d *Export* : mengekspor *project* yang sedang dibuka kedalam bentuk *file* berekstensi zip
- e *Copy* : melakukan *copy code* ke *clipboard*
- f *Cut* : memotong *code* ke *clipboard*
- g *Paste* : paste *code* ke *clipboard*
- h *Undo* : mengembalikan perubahan
- i *Run* : menjalankan aplikasi yang telah dibuka
- j *Stop* : berhenti dari *code debug*
- k *Refresh* : menjalankan ulang pemrograman

2. Code area

Tempat menulis *code* (bahasa pemrograman) dalam sebuah modul. Area ini merupakan tempat programmer melakukan *layout-layout* yang terkait sehingga membentuk sebuah desain aplikasi yang akan dibuat.



3. Abstract Designer

Tempat untuk merancang *layout* yang nantinya akan tampak saat di-run ke Android.

2.2.5. Brackets

Menurut Riyadi dalam (Kiswoyo, 2015), “*Brackets* adalah *code editor* yang secara khusus dikembangkan untuk tujuan *web design* dan *front-end development*”. Project *Brackets* ini diusung oleh *Adobe* secara *open source* dan dikembangkan secara aktif oleh komunitas *web developer* dan benar-benar dibuat untuk kebutuhan *web development*. *Brackets* merupakan aplikasi *code editor* yang berfokus pada HTML, CSS, dan *javascript*.

Berikut kelebihan *Brackets*:

1. *Live HTML Development*

Dengan menggunakan *Brackets*, setiap mau menulis *code*, efeknya akan langsung dikirim ke *web browser* tanpa harus menyimpan atau *me-reload* halaman. Catatan: *Install* Google Chrome untuk menikmati fitur ini.

2. *Javascript Debugging* dengan *TheseusBrackets* memanfaatkan *Theseus* untuk melakukan inspeksi dan *debugging javascripts*. Kalian akan dimudahkan dalam menginspeksi variabel dan *control flow* bahkan secara *asynchronous*.

3. Aktif dan dikembangkan khusus untuk *web defeloper*

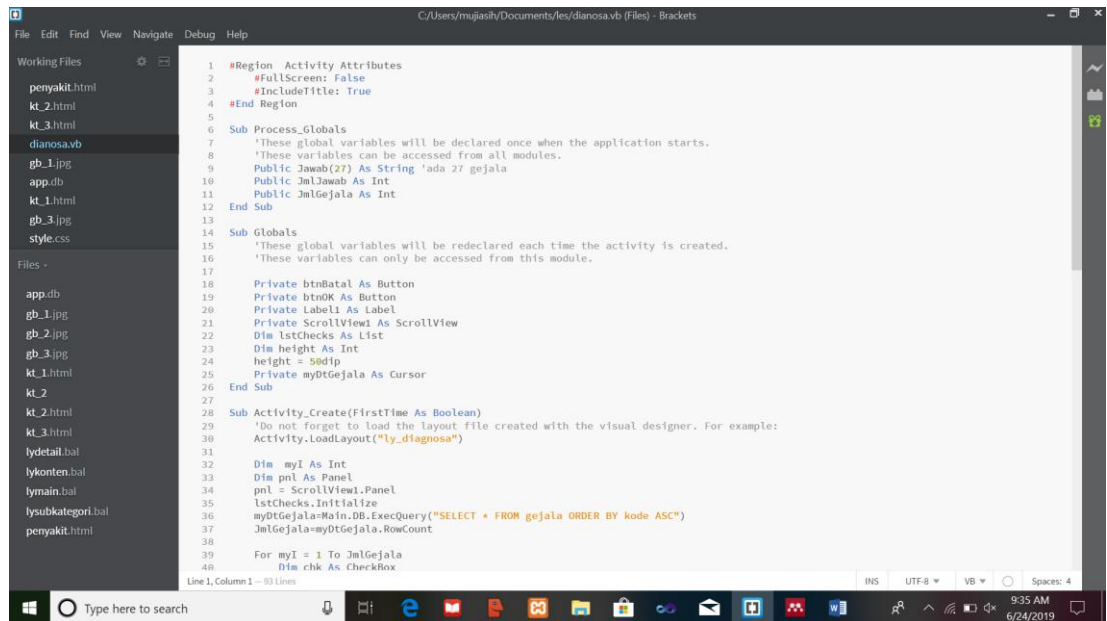
Brackets dikembangkan dengan menggunakan teknologi HTML, CSS, dan *Javascript* oleh para pengembang *web*. Sehingga arah pengembangan *Brackets* tentu saja unntuk memenuhi kebutuhan para *web developer*. Selain itu, *Brackets* aktif dikembangkan sehingga fitur dan ekstensi baru setiap 2.5 pekan.

4. Fitur “*Quick Edit*” yang mempercepat kerja

Fitur “*Quick Edit*” merupakan salah satu fitur andalan dari *Brackets*. Dengan fitur ini, bisa mengedit rule CSS langsung dari element HTML yang menggunakan rule tersebut tanpa harus berpindah-pindah *file*, atau dokumen.

Hal ini belaku juga untuk *javascript*.





Sumber: Aplikasi Brackets

Gambar. II.2. Tampilan Coding Aplikasi *Brackets*

2.2.6. Flowchart

Menurut (Iswandy, 2015), Pengertian *flowchart* adalah “urutan-urutan langkah kerja suatu proses yang digambarkan dengan menggunakan symbol-simbol yang disusun secara sistematis”

Menurut (Cahyono, 2018) , aturan-aturan dalam pembuatan diagram alir adalah sebagai berikut:

1. Diagram alir adalah diagram yang berorientasi dari atas ke bawah serta dari sebelah kiri ke kanan.
2. Semua proses atau kegiatan pada diagram alir wajib dinyatakan secara eksplisit.
3. Setiap diagram alir harus dimulai dari satu *start state* dan berakhir pada satu atau lebih terminal akhir.

4. Pakai *off-page connector state* dan *connector* dengan label yang sama untuk membentuk keterhubungan antar *path* algoritma yang terpotong atau terputus, misalnya sebagai akibat pindah atau ganti halaman.

Berikut ini merupakan simbol-simbol yang dipakai dan menggambarkan algoritma ke dalam bentuk diagram alir serta kegunaan dari symbol-simbol yang dimaksud.



	Arus atau <i>flow</i>	Penghubung antara prosedur atau proses
	<i>Connector</i>	Simbol keluar atau masuk prosedur atau proses dalam halaman yang sama
	<i>Off-line Connector</i>	Simbol keluar atau masuk prosedur
	<i>Process</i>	Pengolahan yang dapat dilakukan komputer
	<i>Decision</i>	Kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau aksi
	<i>Predefined Process</i>	Mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam <i>storage</i>
	<i>Terminal</i>	Permulaan atau akhir dari suatu program
	<i>Manual Input</i>	Pemasukan data secara manual <i>on-line keyboard</i>

Sumber : (Cahyono, 2018)

