

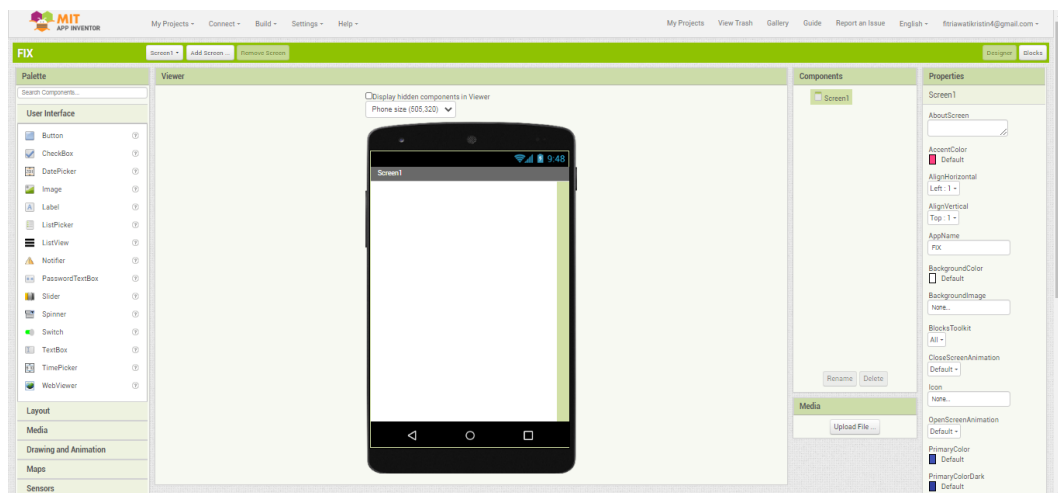
BAB II

LANDASAN TEORI

A. App Inventor

App Inventor adalah program yang dibuat oleh Google dan sekarang dikembangkan oleh MIT (Hamdi, G., Krisnawati, 2011). Program *App Inventor* dapat diakses melalui ai2.appinventor.mit.edu dan digunakan untuk membuat dan mendesain aplikasi android yang berbasis *web page* dan *java interface*. *App Inventor* ini menggunakan antarmuka grafis, sama seperti menggunakan *Scratch*, pengguna hanya melakukan *drag and drop* objek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android. Begitu dengan *coding*, dalam menggunakan program ini, tidak perlu menulis kode program yang amat sangat panjang, cukup dengan *drag and drop* seperti halnya menyusun *puzzle* (Komputer, Wahana, 2013).

Tampilan perancangan aplikasi Android menggunakan App Inventor ditunjukkan pada Gambar 2.1



Gambar 2.1 Tampilan Halaman Desain Aplikasi App Inventor

Pada lingkungan kerja App Inventor ini terdapat beberapa komponen yang terdiri dari :

1. Komponen Desainer

Komponen desainer berjalan pada browser yang digunakan untuk memilih komponen yang dibutuhkan dan mengatur propertinya. Pada komponen desainer sendiri terdapat 5 bagian, yaitu *palette*, *viewer*, *component*, *media* dan *properties*.

- a. *Palette* : list komponen yang bisa digunakan.
- b. *Viewer* : menempatkan komponen dan mengaturnya sesuai tampilan yang diinginkan.
- c. *Component* : tempat list komponen yang dipakai pada project.
- d. *Media* : mengambil media audio atau gambar untuk project.
- e. *Properties* : mengatur properti komponen yang digunakan, seperti *width*, *height*, *name*, dll.

2. Blok Editor

Blok editor berjalan di luar browser dan digunakan untuk membuat dan mengatur *behaviour* dari komponen-komponen yang dipilih dari komponen desainer.

3. Emulator

Emulator digunakan untuk menjalankan dan mencoba *project* yang dibuat. Jadi sebelum mengunduh aplikasi tersebut untuk diinstal di *Smartphone*, pengguna dapat mencoba hasil dari aplikasi tersebut terlebih dahulu.

B. Modul *Bluetooth* HC-05

Bluetooth adalah suatu peralatan media komunikasi yang dapat digunakan untuk menghubungkan sebuah perangkat komunikasi dengan perangkat komunikasi lainnya. *Bluetooth* umumnya digunakan di handphone, komputer atau pc, tablet, dan lain-lain. *Bluetooth* beroperasi dalam pita frekuensi 2,4 GHz dengan menggunakan sebuah *frequency hopping tranceiver* yang mampu menyediakan layanan komunikasi data dan suara secara *real time* antara *host-host Bluetooth* dengan jarak terbatas. Beberapa modul rangkaian *Bluetooth* sudah diproduksi ditunjukkan pada Gambar 2.2



Gambar 2.2 Modul *Bluetooth* HC-05

Menurut Budiharto (2010:83) modul *Bluetooth* adalah suatu perangkat yang berfungsi sebagai media penghubung antara *Smartphone Android* dengan Mikrokontroler yang sudah tertanam modul *Bluetooth* tersebut. Modul *Bluetooth* seri HC memiliki banyak jenis atau varian, yang secara garis besar terbagi menjadi dua yaitu jenis “*industrial series*” yaitu HC-03 dan HC-04 serta “*civil series*” yaitu HC-05 dan HC-06. Modul *Bluetooth* serial yang selanjutnya disebut dengan modul BT saja digunakan untuk mengirimkan data serial TTL via *Bluetooth*.

Bluetooth HC-05 merupakan module *Bluetooth* yang bisa menjadi *slave* ataupun *master* hal ini dibuktikan dengan bisa memberikan notifikasi untuk melakukan penyandingan ke perangkat lain, maupun perangkat lain tersebut yang melakukan penyandingan ke modul *Bluetooth HC-05*. Untuk mengatur perangkat

Bluetooth dibutuhkan perintah-perintah *AT Command* yang mana *perintah AT Command* tersebut akan di respon oleh perangkat *Bluetooth* jika modul *Bluetooth* tidak dalam keadaan terkoneksi dengan perangkat lain. Tabel 2.1 dibawah adalah tabel *AT Command Modul Bluetooth HC-05*.

Tabel 2.1 AT Command Modul Bluetooth HC-05

NO	Perintah	Kirim	Terima	Keterangan
1	Tes Koneksi	AT	OK	-
2	Ganti Nama <i>Bluetooth</i>	AT+NAME=Nama	OK	-
3	Ubah Password	AT+PSWD=Password	OK	+PIN:’’xxxx’’
4	Ubah Baud rate	AT+UART?	OK	+UART:9600
5	Reset Modul	AT+RESET	OK	-

Pada dasarnya teknologi *Bluetooth* ini diciptakan bukan hanya untuk menggantikan atau menghilangkan penggunaan media kabel dalam melakukan pertukaran data atau informasi, tetapi juga mampu menawarkan fitur yang bagus atau baik untuk teknologi *mobile wireless* atau tanpa kabel, dengan biaya yang relatif rendah, konsumsi daya rendah, serta mudah dalam pengoperasiannya. Saat ini sudah banyak perangkat yang menggunakan teknologi *Bluetooth*.

C. *Arduino Nano*

Arduino Nano adalah salah satu papan pengembangan mikrokontroler yang berukuran kecil, lengkap dan mendukung penggunaan *breadboard*. *Arduino Nano* diciptakan dengan basis mikrokontroler ATmega328 (untuk *Arduino Nano* versi 3.x) atau ATmega 168 (untuk *Arduino* versi 2.x). *Arduino Nano* kurang lebih memiliki fungsi yang sama dengan *Arduino Duemilanove*, tetapi dalam paket yang berbeda. *Arduino Nano* tidak menyertakan colokan DC berjenis Barrel Jack, dan dihubungkan ke komputer menggunakan port USB Mini-B. *Arduino Nano*

dirancang dan diproduksi oleh perusahaan Gravitech. (Ihsan: 2016, 1). Berikut dapat dilihat bentuk *Arduino Nano* pada Gambar 2.3



Gambar 2.3 *Arduino Nano*

Arduino Nano memiliki beberapa fasilitas untuk berkomunikasi dengan komputer, berkomunikasi dengan *Arduino* lainnya, atau dengan mikrokontroler lainnya. *Chip Atmega328* menyediakan komunikasi serial UART TTL (5V) yang tersedia pada pin 0 (RX) dan pin 1 (TX). Sebuah *chip FDTI* yang terdapat pada *board* berfungsi menterjemahkan bentuk komunikasi ini melalui *USB* dan akan tampil *virtual port* di komputer.

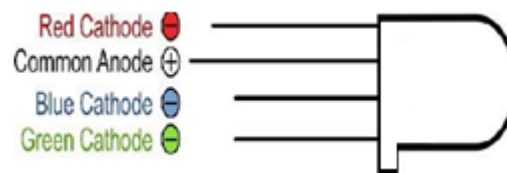
Pada *Arduino software* (IDE) terdapat monitor serial yang memudahkan data textual untuk dikirim menuju *Arduino* atau keluar dari *Arduino*. Lampu *LED TX* dan *RX* akan menyala berkedip-kedip ketika ada data yang ditransmisikan melalui *chip FDTI USB to serial* melalui kabel *USB* ke komputer. Untuk menggunakan komunikasi serial dari digital pin, maka *Software Serial library* yang digunakan pada pemrograman.

Tabel 2.2 Spesifikasi *Arduino Nano*

Mikrokontroler	Atmel ATmega168 atau ATmega328
Tegangan Operasi	5V
Input Voltage	7-12V
Input Voltage (limit)	6-20V
Pin Digital I/O	14 (6 pin digunakan sebagai output PWM)
Pins Input Analog	8
Arus DC per pin I/O	40 mA
Flash Memory	16KB (ATmega168) atau 32KB (ATmega328) 2KB digunakan oleh Bootloader
SRAM	1 KB (ATmega168) 2 KB (ATmega328)
EEPROM	512 byte (ATmega168) atau 1KB (ATmega328)
Clock Speed	16 MHz
Ukuran	1.85cm x 4.3cm

D. Light Emitting Diode RGB

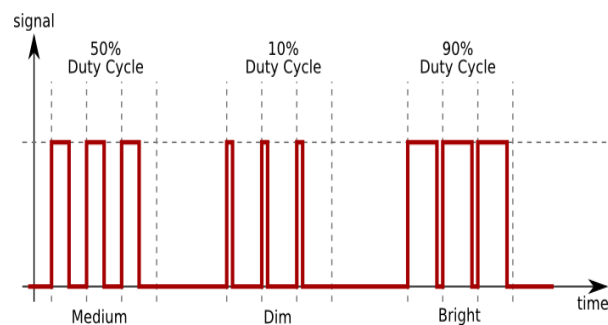
LED RGB adalah *LED* yang terdiri dari tiga warna yang diintegrasikan menjadi satu kemasan. Konstruksi *LED RGB* ditunjukkan pada Gambar 2.4 yaitu terdiri dari *LED* warna merah, hijau, dan biru. *LED RGB* pada gambar tersebut memiliki jenis *common anode* yang mana terminal anoda dari *LED* merah, hijau dan biru dihubungkan, sedangkan katodanya dalam kondisi terbuka. (Supegina, F., Imam, 2014).

Gambar 2.4 Konstruksi *LED RGB*

Pengaturan warna dari nyala *LED* dapat dilakukan dengan mengatur arus yang diberikan pada *LED*, sehingga akan mempengaruhi kecerahan *LED*. Jadi dengan memberikan arus yang berbeda-beda pada tiap-tiap terminal katoda *LED RGB* akan menghasilkan nyala *LED* dengan warna yang berbeda-beda.

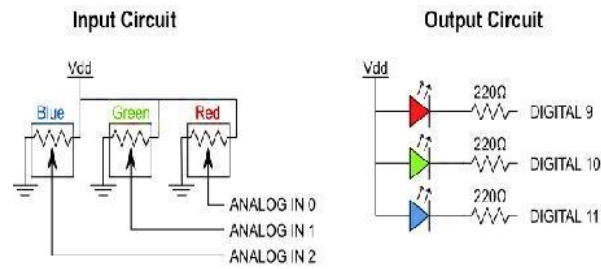
Pengaturan arus ini dapat dilakukan dengan metoda PWM (*pulse width modulation*). PWM merupakan metoda modulasi pulsa yang dilakukan dengan cara mengatur *duty cycle* dari gelombang kotak (*square wave*). Sehingga dengan metoda ini dapat digunakan untuk mengontrol daya rata-rata yang dikirim ke setiap *LED RGB*.

Dalam Gambar 2.5, ditunjukkan tiga bentuk sinyal dengan *duty cycle* yang berbeda, pertama yaitu *duty cycle* 50%, kemudian 10% dan 90%. Pada *duty cycle* 10%, sinyal logika *high* mempunyai waktu yang singkat setiap siklusnya, tetapi dengan siklus 90%, sebagian besar periode sinyal pada kondisi logika *high*. Jika frekuensi sinyal cukup cepat, maka tidak akan ada kedip (*flicker*) terlihat, dan kecerahan *LED* akan sebanding dengan *duty cycle* sinyal ini.



Gambar 2.5 Duty Cycle Sinyal PWM

Dalam realisasinya sinyal PWM ini akan dibangkitkan dari pin *Arduino*, yaitu pin 9, pin 10, dan pin 11. Pin-pin tersebut memang dirancang untuk membangkitkan sinyal PWM yang dikendalikan melalui program. Skematik pemasangan *LED RGB* yang dihubungkan dengan pin *Arduino* ditunjukkan pada Gambar 2.6



Gambar 2.6 Skematik Pemasangan LED RGB

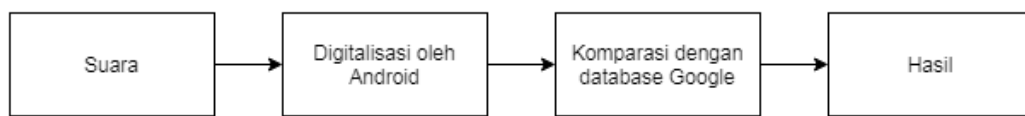
E. Google Speech Recognition

Menurut Prasojo (2015), *speech recognition* merupakan konversi sebuah sinyal akustik, yang ditangkap oleh *microphone* atau telepon, untuk merangkai kata kata. Sejalan dengan pendapat Prasojo, Washani dan Sharma (2015) menambahkan jika *speech recognition* tidak hanya melibatkan proses mengkonversi sinyal akustik menjadi teks tetapi juga proses mengidentifikasi apa yang dikatakan pembicara/pengguna. Oleh karena itu, kata-kata yang diucapkan pengguna akan dikenali dan merupakan hasil akhir untuk sebuah aplikasi seperti *command* dan *control*, masukan data, dan persiapan dokumen.

Speech recognition itu sendiri memiliki beberapa jenis berdasarkan tipe atau bentuk ucapan yang dapat dikenali oleh sistem *speech recognition* itu sendiri. Santosh, Gaikwad, Gawali, dan Yannawar (2010:16) menjelaskan empat jenis *speech recognition* tersebut. Jenis pertama disebut *isolated word*. Jenis ini akan mengenali informasi berupa satu kata atau satu ucapan dalam satu waktu pemberian perintah atau informasi. Jenis kedua yaitu *connected word*. Jenis kedua ini akan mengenali informasi berupa beberapa ucapan dalam satu waktu dengan catatan terdapat spasi atau jeda yang sebentar antar ucapan tersebut. Jenis ketiga dari *speech recognition* yaitu *continuous speech*. Jenis ini akan mengenali informasi atau perintah berupa ucapan-ucapan natural yang terdiri dari beberapa kalimat. Sehingga

pada jenis ini pengguna atau pemberi perintah dapat berbicara secara natural. Yang keempat yaitu jenis *spontaneous speech*. Pada jenis ini mesin akan mengenali informasi berupa ucapan atau informasi yang spontan dari pengguna.

Speech recognition juga memiliki sistem sampling atau digitizing suara. Sistem ini bekerja dengan cara mengambil ukuran yang pas dari gelombang suara yang diproduksi oleh pengguna. Sistem sampling ini akan menyaring suara yang telah didigitalkan tersebut dan membuang gangguan berupa kebisingan. Sistem sampling ini juga berfungsi untuk menormalkan suara dengan volume yang tetap dan mendatarkan suara. Adapun untuk sistem *speech recognition* itu sendiri mulai dari pengenalan suara sampai kepada hasil berupa teks digambarkan dalam diagram alir di bawah ini (Irawan, 2014).



Gambar 2.7 Diagram Alir Speech Recognition

(Sumber: Irawan, 2014:22)

Pada Gambar 2.7 terlihat diagram alir sistem *speech recognition* dari proses masukan berupa suara. Suara yang diucapkan selanjutnya diproses oleh android melalui digitalisasi. Setelah suara tadi didigitalisasikan selanjutnya akan dikomparasi dengan *database* google melalui template kata yang dimiliki google. Hasil dari komparasi berupa kata yang ditampilkan pada layar *Smartphone* atau yang dapat dibaca oleh perangkat.