

## **BAB II KAJIAN PUSTAKA**

### **2.1 Teori Dasar**

#### **2.1.1 Sistem Kendali Cerdas**

Dalam perkembangan ilmu dan teknologi dewasa ini sistem kendali manual maupun otomatis memiliki peran yang sangat penting. Peranan sistem kendali otomatis adalah yang paling menonjol dalam berbagai keperluan manusia atau bangsa yang telah maju peradabannya. Kemajuan sistem kendali otomatis akan memberikan kemudahan dalam melakukan suatu aktifitas, mempertinggi kualitas, menurunkan biaya produksi dan penghematan energi.

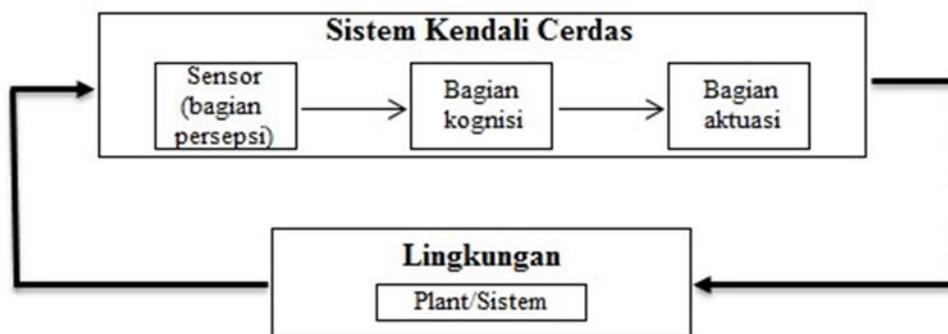
Menurut Wati, (2011: 1-2) Seiring perkembangan teknologi komputer, sistem kendali juga mengalami perkembangan yang cukup pesat karena kecepatan komputasi yang tinggi memungkinkan algoritma kendali yang rumit dapat diimplementasikan secara real time. Berbagai metode kendali baru yang awalnya baru sebatas penelitian karena terhalang oleh daya dukung komputasi, saat ini sudah banyak diimplementasikan dengan unjuk kerja yang baik. Salah satunya adalah sistem kendali cerdas.

Sistem kendali cerdas adalah sistem kendali yang menggunakan sistem kecerdasan buatan dalam perancangan pengendali maupun sebagai pengendali itu sendiri. Sistem kecerdasan buatan adalah produk dari bidang teknologi informasi

yang berusaha meniru cara berpikir manusia, bertujuan untuk menggantikan operator dengan mesin cerdas.

Sistem kendali cerdas memiliki kemampuan menangani ketidakpastian sistem, antara lain:

1. Ketidakpastian dari model sistem/plant yang dikendalikan
2. Perubahan kondisi lingkungan di luar plant yang tidak terprediksi
3. Informasi sensor yang tidak lengkap atau tidak konsisten
4. Perubahan fungsi aktuator.



Gambar 2.1 Struktur Sistem Kendali Cerdas

Sumber: (Wati, 2011)

Bagian persepsi dapat berupa sensor yang menangkap informasi dari plant atau dari lingkungan plant. Bagian kognisi merupakan bagian pengambil keputusan yang dapat berupa pemikiran, perencanaan, atau pembelajaran. Bagian aktuari berupa aktuator yang akan mengolah sinyal kendali dari sistem kendali cerdas untuk menggerakkan plant ke kondisi yang diinginkan. Dalam kondisi aktuator tidak dapat bekerja dengan baik, maka sistem kendali cerdas harus punya kemampuan untuk mengatasinya.

### 2.1.2 Pengenalan Suara

Menurut Saifuddin & Winardi, (2015) Pengenalan suara atau *voice/speech recognition* adalah suatu pengembangan teknik dan sistem yang memungkinkan komputer untuk menerima masukan berupa kata atau yang diucapkan. Teknologi ini memungkinkan suatu perangkat untuk mengenali dan memahami kata-kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan dalam suatu perangkat. Kata-kata yang diucapkan diubah bentuknya menjadi sinyal digital dengan cara mengubah gelombang suara menjadi sekumpulan angka yang kemudian disesuaikan dengan kode-kode tertentu untuk mengidentifikasi kata-kata tersebut. *Voice recognition* dibagi menjadi dua jenis, yaitu:

1. *Speech recognition*: merupakan proses yang dilakukan komputer untuk identifikasi suara yang diucapkan oleh seseorang tanpa mempedulikan identitas orang terkait. Implementasi *speech recognition* misalnya perintah suara untuk menjalankan aplikasi komputer. Parameter yang dibandingkan ialah tingkat penekanan suara yang kemudian akan dicocokkan dengan template database yang tersedia.
2. *Speaker recognition*: merupakan sistem pengenalan identitas yang diklaim oleh seseorang dari suaranya atau berdasarkan orang yang berbicara. Misalnya berupa intonasi suara, tingkat kedalaman suara, dan sebagainya. *Speech recognition* juga dikenal sebagai *automatic speech recognition* atau *computer speech recognition* yaitu penerjemah perkataan yang diucapkan menjadi *text*.

Teknologi *speech recognition* ini sudah ada sejak lama dan sekarang banyak sekali jenis aplikasi yang dikembangkan menggunakan teknologi ini.

### **2.1.3 Smartphone Android**

Berdasarkan penelitian Rizkidinia, Yamin, & Muchlis, (2016) *Smartphone* secara harfiah artinya telepon pintar, yakni telepon seluler yang memiliki kemampuan seperti PC walaupun terbatas. Selain itu, *smartphone* juga mendukung email dan *organizer*. Fitur lainnya adalah kemampuannya untuk ditambah aplikasi-aplikasi baru. *Smartphone* adalah telepon genggam yang mempunyai kemampuan tingkat tinggi, kadang-kadang dengan fungsi yang menyerupai komputer. Belum ada standar pabrik yang menentukan arti telepon cerdas. Bagi beberapa orang, *smartphone* merupakan telepon yang bekerja menggunakan seluruh perangkat lunak sistem operasi yang menyediakan hubungan standar dan mendasar bagi pengembang aplikasi. Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android merupakan generasi baru *platform mobile* yang memberikan kesempatan kepada pengembang untuk melakukan pengembangan sesuai dengan yang diharapkan. Sistem operasi yang mendasari Android merupakan lisensi dibawah naungan GNU, *General Public License Versi 2 (GPLv2)*, yang biasa dikenal dengan istilah *Copyleft*. Istilah *copyleft* ini merupakan lisensi yang setiap perbaikan oleh pihak ketiga harus terus jatuh dibawah terms.

Berdasarkan penelitian Saifuddin & Winardi, (2015) Secara umum pengertian Android adalah suatu *software* (perangkat lunak) yang berbasis Linux untuk telepon seluler dan komputer tablet yang meliputi sistem operasi, *middleware* dan aplikasi inti. Android SDK menyediakan alat dan API yang diperlukan untuk memulai pengembangan aplikasi pada platform. Android menggunakan bahasa pemrograman Java, yaitu kode Java yang terkompilasi dengan data dan *file resources* yang dibutuhkan aplikasi dan digabungkan oleh aapt tools menjadi paket Android. File tersebut ditandai dengan ekstensi .apk. *File* inilah yang didistribusikan sebagai aplikasi dan diinstall pada perangkat *mobile*. Telepon seluler atau HP pertama yang memakai sistem operasi Android adalah HTC Dream, yang dirilis pada tanggal 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis telepon seluler yang menggunakan Android.

Secara umum, kelebihan Android dibanding sistem operasi lain adalah :

1. *User Friendly*, dalam artian Android sangat mudah dioperasikan.
2. Bersifat *Open Source*, karena Android dibangun di atas kernel Linux maka siapapun dapat mengembangkan dan memodifikasi Android tanpa harus membayar.
3. Merakyat, sangat cocok untuk berbagai kalangan. Dari kelas bawah hingga kelas atas, kalangan orang tua maupun anak muda.
4. Dukungan berbagai aplikasi, terdapat jutaan aplikasi yang tersedia untuk menunjang kinerja Android.

#### **2.1.4 Internet of Things (IoT)**

*Internet of Things* (IoT) sudah mulai merasuki segala lini kehidupan kita. Tak cuma di rumah, IoT telah menginvasi tempat kerja, pabrik, rumah sakit, dan lain sebagainya. Bukan hanya menjanjikan kemudahan dan kenyamanan hidup, tetapi IoT juga menyajikan potensi ekonomi dan bisnis. Konsep IoT dapat digambarkan sebagai terhubungnya objek fisik ke jaringan internet. Contoh objek fisik ini adalah telepon mobile (*smart device*), pintu garasi, mesin cuci, lampu, dan lain-lain yang berbentuk peralatan elektronik.

Menurut Ashton, (2017: 9) *Internet dari Things* adalah sensor-sensor yang terhubung ke Internet, berperilaku seperti internet dengan membuka koneksi-koneksi setiap saat,serta berbagi data dengan bebas, dan membiarkan aplikasi tak terduga, sehingga komputer dapat memahami dunia di sekitar mereka dan menjadi bagian dari kehidupan manusia.

## **2.2 Teori Khusus**

### **2.2.1 Mikrokontroler Arduino**

Menurut July Tjindrawan, (2015: 134) Mikrokontroler adalah sebuah rangkaian terpadu yang berisi memori untuk menyimpan program data, prosesor atau CPU untuk mengolah program dan informasi, serta *input/output* yang dapat

dihubungkan dengan sensor dan aktuator. Untuk mengendalikan sebuah aktuator, output mikrokontroler perlu dihubungkan dengan sebuah *driver*, yang umumnya berupa penguat arus. Dengan demikian, arus output mikrokontroler yang kecil dapat mengendalikan beban yang besar.

Mikrokontroler bekerja berdasarkan program yang dimasukkan ke dalam memori, yakni memori program yang biasanya berupa *memory flash*. Dengan merubah program didalamnya, maka mikrokontroler yang sama akan menjalankan fungsi yang berbeda.

Menurut Syahwill, (2013: 53) Mikrokontroler adalah sebuah sistem komputer fungsional dalam sebuah chip. Di dalamnya terkandung sebuah inti prosesor, memori (sejumlah kecil RAM, memori program, atau keduanya), dan perlengkapan input-output. Mikrokontroler adalah alat yang mengerjakan instruksi-instruksi yang diberikan kepadanya. Artinya, bagian terpenting dan utama dari suatu sistem terkomputerisasi adalah program itu sendiri yang dibuat oleh seorang programmer.

Dengan demikian Mikrokontroler (pengendali mikro) pada suatu rangkaian elektronik berfungsi sebagai pengendali yang mengatur jalannya proses kerja dari rangkaian elektronik digital yang mempunyai masukan dan keluaran serta dapat dikendalikan dengan sebuah program.

Menurut Kholilah, Rafi, & Tahtawi, (2016) Arduino adalah kit elektronik atau papan rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama yaitu sebuah chip mikrokontroler dengan jenis AVR dari perusahaan ATMEL. Arduino sendiri terbagi menjadi beberapa jenis, diantaranya

Arduino Uno, Arduino Duemilanove, Arduino Mega, Arduino Nano, Arduino Romeo, dll. Penggunaan jenis Arduino tersebut tentunya disesuaikan dengan kebutuhan dan masing-masing memiliki kekurangan dan kelebihan.

Menurut Oroh & Kendekallo, (2014) Arduino adalah sebuah board mikrokontroler yang didasarkan pada IC Atmega328. Salah satu contoh yang akan dibahas kali ini yaitu arduino Uno. Arduino UNO mempunyai 14 pin digital *input/output* (6 di antaranya dapat digunakan sebagai *output PWM*), 6 *input analog*, sebuah osilator Kristal 16 MHz, sebuah koneksi USB, sebuah *power jack*, sebuah *ICSP header*, dan sebuah tombol reset. Arduino UNO memuat semua yang dibutuhkan untuk menunjang mikrokontroler.

Arduino dikembangkan dari thesis Hernando Barragan pada tahun 2004, seorang mahasiswa asal Kolombia. Judul thesisnya yaitu “*Arduino-Revolusi Open Hardware*”. Arduino diawali di ruang kelas *Interactive Design Institute* di Ivrea (IDII), pada tahun 2005 di Ivrea, Italia. Arduino ditemukan oleh Massimo Banzi dan David Cuartielles dengan tujuan awal yaitu untuk membantu para siswa membuat perangkat desain dan interaksi dengan harga yang murah dibandingkan dengan perangkat lain yang tersedia saat itu, seperti *BASIC Stamp* yang harganya cukup mahal bagi pelajar pada saat itu. Arduino berasal dari bahasa Italia yang berarti teman yang berani. Pada bulan Mei 2011, Arduino sudah terjual lebih dari 300.000 unit. Arduino saat ini sudah menjadi platform OSHW (*Open Source Hardware*).

Dengan demikian Arduino bisa disimpulkan sebagai kit elektronik yang open source yang bisa diatur fungsi dan tujuan penggunaannya dengan memasukkan sebuah coding program untuk menghasilkan sebuah sistem kendali.

Pada penelitian ini menggunakan arduino UNO R3 karena harganya yang murah serta memiliki fungsi yang memadai untuk membuat sistem pengendali motor ini.



Gambar 2.2 Arduino UNO

Sumber: Data Olahan Sendiri

### 2.2.2 Sepeda Motor

Sepeda motor adalah kendaraan beroda dua yang digerakkan oleh sebuah mesin. Letak kedua roda sebaris lurus dan pada kecepatan tinggi sepeda motor tetap stabil disebabkan oleh gaya giroskopik. Sedangkan pada kecepatan rendah, kestabilan atau keseimbangan sepeda motor bergantung kepada pengaturan setang oleh pengendara. Penggunaan sepeda motor di Indonesia sangat populer karena

harganya yang relatif murah, terjangkau untuk sebagian besar kalangan dan penggunaan bahan bakarnya serta biaya operasionalnya cukup hemat.

Sepeda motor memiliki tipe atau jenis masing- masing, contohnya yang sering kita jumpai diantaranya yakni: motor skuter matik, motor bebek, motor manual/*naked*, motor *trail*, motor *sport* dll.

Pada penelitian ini menggunakan sebuah motor Sport karena motor yang digunakan oleh peneliti adalah motor sport jenis Yamaha R15. Namun sebenarnya sistem pengendali ini dapat diterapkan ke berbagai jenis motor lainnya.



Gambar 2.3 Motor Sport

Sumber: Data Olahan Sendiri.

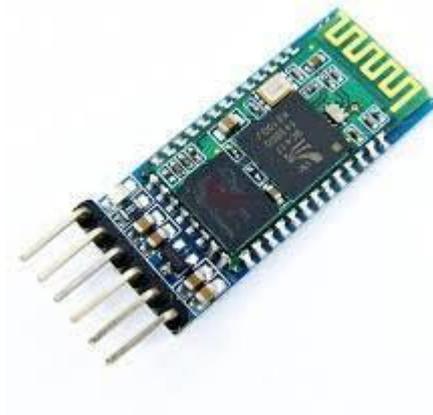
### 2.2.3 Bluetooth HC-05

Bluetooth adalah Sebuah teknologi *wireless* yang mampu menyediakan layanan komunikasi data dan suara dengan jarak jangkauan terbatas. Bluetooth adalah sebuah teknologi komunikasi *wireless* (tanpa kabel) yang beroperasi dalam pita frekuensi 2,4 GHz *unlicensed ISM (Industrial, Scientific and Medical)* dengan menggunakan sebuah *frequency hopping tranceiver* yang mampu menyediakan

layanan komunikasi data dan suara secara *real time* antar host-host bluetooth dengan jarak jangkauan layanan yang terbatas.

Sistem bluetooth terdiri dari sebuah *radio transceiver*, *baseband link management dan control*, *baseband (processor core, SRAM, UART, PCM USB interface)*, *flash* dan *voice code*, sebuah *link manager*, *baseband link controller* menghubungkan perangkat keras ke *radio baseband processing* dan layer protokol fisik. *Link manager* melakukan aktivitas-aktivitas protokol tingkat tinggi seperti melakukan *link setup*, *autentifikasi dan konfigurasi*. (Saifuddin & Winardi, 2015).

Module bluetooth HC serial digunakan untuk mengirimkan atau menerima data serial TTL via bluetooth. Modul bluetooth seri HC memiliki banyak jenis atau varian, yang secara garis besar terbagi menjadi dua yaitu jenis Industrial series (HC-03/HC-04) dan Civil series (HC-05/HC-06). Untuk melihat mode kerja module bluetooth master atau slave dapat dikenali dari nomor serinya, jika nomor seri genap maka module bluetooth tersebut telah diset oleh pabrik dengan bekerja sebagai slave mode atau master mode dan tidak dapat diubah mode kerjanya. Sedangkan bluetooth HC dengan nomor seri ganjil, kondisi default diset sebagai *slave mode*, tetapi dapat di ubah menjadi mode master dengan AT Command. (Doni & Widiyanto, 2015).



Gambar 2.4 Bluetooth HC-05

Sumber: Data Olahan Sendiri.

#### **2.2.4 Modul *Wi-Fi* Wemos D1 Mini Pro**

Wemos merupakan salah satu modul board yang dapat berfungsi dengan arduino khususnya untuk project yang mengusung konsep IOT. Wemos dapat *running standalone* atau dapat berjalan sendiri tanpa perlu dihubungkan dengan mikrokontroler, berbeda dengan modul wifi lain yang masih membutuhkan mikrokontroler sebagai pengontrol atau otak dari rangkaian tersebut, wemos dapat running stand-alone karena didalamnya sudah terdapat CPU yang dapat memprogram melalui serial port atau via OTA serta transfer program secara *wireless*.

Wemos memiliki Chipset ESP8266 merupakan sebuah chip yang memiliki fitur Wifi dan mendukung stack TCP/IP. Modul kecil ini memungkinkan sebuah mikrokontroler terhubung kedalam jaringan *Wi-Fi* dan membuat koneksi TCP/IP hanya dengan menggunakan command yang sederhana, clock 80 MHz, 4MB

eksternal RAM serta mendukung format IEEE 802.11 b/g/n sehingga tidak menyebabkan gangguan bagi yang lain. Selain itu terdapat Chipset CH340 yang mengubah USB serial menjadi serial interface, seperti aplikasi converter to IrDA atau aplikasi USB *converter* to Printer. Dalam *mode serial interface*, CH340 mengirimkan sinyal penghubung yang umum digunakan pada modem. CH340 digunakan untuk mengubah perangkat serial interface umum untuk berhubungan dengan bus USB secara langsung. (Limantara, A. D., Candra, A. I., & Mudjanarko, S. W. (2017).



Gambar 2.5 Wemos D1 Mini Pro

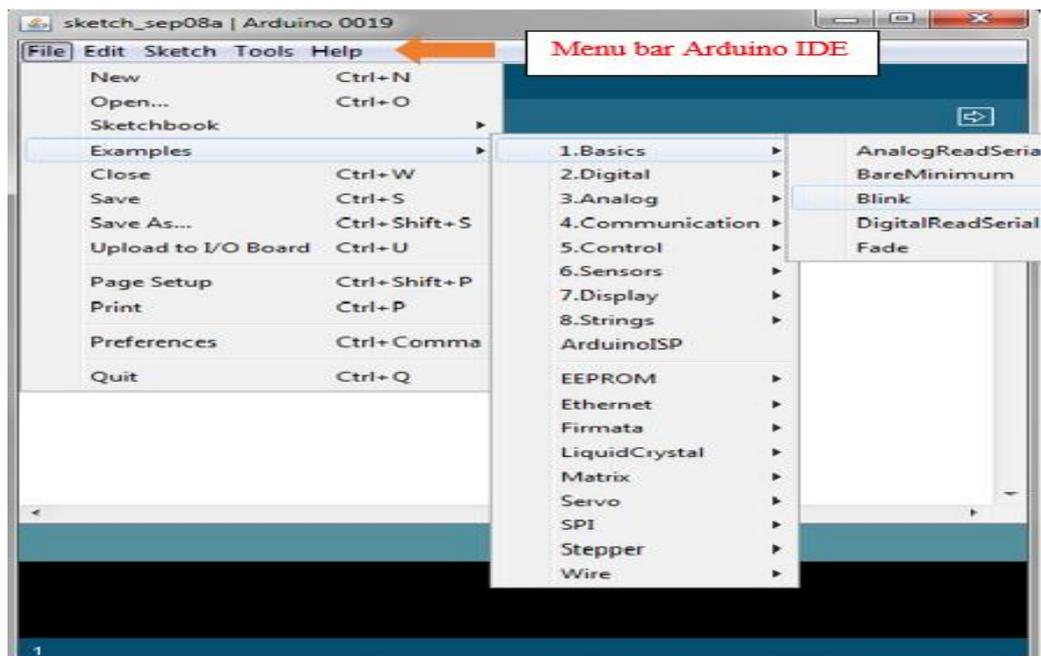
Sumber: Data Olahan Sendiri

## 2.3 Tools/Software/Aplikasi/System

### 2.3.1 Arduino IDE

Pemrograman modul Arduino dilakukan dengan menggunakan bantuan *software* Arduino IDE, yaitu *software* terpadu yang sudah menyediakan fitur-fitur yang diperlukan pengguna untuk menciptakan kode program yang akan dimasukkan ke dalam sebuah modul yang sering disebut juga dengan sketch.

Dalam Arduino IDE terdapat perintah tambahan dalam lima menu: *File*, *Edit*, *Sketch*, *Tools*, *Help*. Menu konteks sensitif, yang berarti hanya barang-barang yang relevan dengan pekerjaan yang sedang dilakukan tersedia.



Gambar 2.6 Menu Bar Arduino IDE

Sumber: Data olahan sendiri.

## 1. *File*

### a. *New*

Membuat sebuah contoh baru dari *editor*, dengan struktur minimal *sketch* yang sudah ada.

### b. *Open*

Memungkinkan untuk memuat *file sketch browsing* melalui *drive* komputer dan folder.

### c. *Open Recent*

Menyediakan daftar singkat *sketch* terbaru, siap dibuka.

### d. *Sketchbook*

Menunjukkan *sketch* saat ini dalam struktur folder *sketch*; mengklik nama apapun akan membuka *sketch* yang sesuai dalam contoh editor baru.

### e. *Examples*

Contoh apa pun yang disediakan oleh Arduino Software (IDE) atau perpustakaan muncul di item menu ini. Semua contoh terstruktur di pohon yang memungkinkan akses mudah berdasarkan topik atau perpustakaan.

### f. *Close*

Menutup contoh Perangkat Lunak Arduino dari mana ia diklik.

### g. *Save*

Menyimpan sketsa dengan nama saat ini. Jika file belum diberi nama sebelumnya, sebuah nama akan diberikan di jendela "*Save as ..*".

### h. *Save as...*

Memungkinkan untuk menyimpan *sketch* saat ini dengan nama yang berbeda.

i. *Page Setup*

Ini menunjukkan jendela *Page Setup* untuk dicetak.

j. *Print*

Mengirimkan sketsa saat ini ke printer sesuai dengan pengaturan yang didefinisikan di *Page Setup*.

k. *Preference*

Membuka jendela *Preferences* dimana beberapa pengaturan IDE dapat disesuaikan, seperti bahasa antarmuka IDE.

l. *Quit*

Tutup semua jendela IDE. *Sketch* yang sama terbuka saat *Quit* terpilih akan dibuka kembali secara otomatis saat Anda menjalankan IDE.

2. *Edit*

a. *Undo/Redo*

Melangkah mundur dari satu atau lebih langkah yang Anda lakukan saat mengedit; Saat Anda kembali, Anda bisa maju dengan *Redo*.

b. *Cut*

Menghapus teks yang dipilih dari editor dan memasukkannya ke clipboard.

c. *Copy*

Duplikat teks yang dipilih di editor dan tempatkan ke clipboard.

d. *Copy for Forum*

Salin kode sketsa Anda ke clipboard dalam bentuk yang sesuai untuk posting ke forum, lengkap dengan pewarnaan sintaks.

e. *Copy as HTML*

Salin kode sketsa Anda ke clipboard sebagai HTML, cocok untuk disematkan di halaman web.

f. *Paste*

Letakkan isi clipboard pada posisi kursor, di editor.

g. *Select all*

Memilih dan menyoroti keseluruhan isi editor.

h. *Comment/Uncomment*

Menempatkan atau menghapus // comment marker di awal setiap baris yang dipilih.

i. *Increase / Decrease Indent*

Menambahkan atau mengurangi ruang pada awal setiap baris yang dipilih, memindahkan teks satu spasi di sebelah kanan atau menghilangkan spasi di awal.

j. *Find*

Membuka jendela *Find* di mana Anda dapat menentukan teks untuk dicari di dalam sketch saat ini sesuai dengan beberapa pilihan.

k. *Find Next*

Sorot kejadian berikutnya - jika ada - dari string yang ditentukan sebagai item pencarian di jendela *Find*, relatif terhadap posisi kursor.

l. *Find Previous*

Menyoroti kejadian sebelumnya - jika ada - dari string yang ditentukan sebagai item pencarian di jendela *Find* relatif terhadap posisi kursor.

3. *Sketch*

a. *Verify / Compile*

Periksa sketsa Anda untuk kesalahan yang mengelompokannya; akan melaporkan penggunaan memori untuk kode dan variabel di area konsol.

b. *Upload*

Kompilasi dan muat *file* biner ke papan yang dikonfigurasi melalui *Port* yang dikonfigurasi.

c. *Upload Using Programmer*

Ini akan menimpa *bootloader* di papan tulis; Anda perlu menggunakan *Tools > Burn Bootloader* untuk mengembalikannya dan bisa mengunggah ke port USB lagi. Namun, ini memungkinkan Anda menggunakan kapasitas penuh memori Flash untuk *sketch* Anda. Harap dicatat bahwa perintah ini TIDAK akan *burn* *sekering*. Untuk melakukannya, perintah *Tools -> Burn Bootloader* harus dijalankan.

d. *Export Compiled Binary*

Menyimpan *file .hex* yang dapat disimpan sebagai arsip atau dikirim ke board dengan menggunakan alat lainnya.

e. *Show Sketch Folder*

Membuka folder sketsa saat ini.

f. *Include Library*

Menambahkan perpustakaan ke *sketch* anda dengan memasukkan *# include* pernyataan di awal kode Anda. Untuk lebih jelasnya, lihat perpustakaan di bawah ini. Selain itu, dari item menu ini Anda dapat mengakses *Library Manager* dan mengimpor perpustakaan baru dari *file .zip*.

g. *Add File...*

Menambahkan *file* sumber ke *sketch* (akan disalin dari lokasinya saat ini). *File* baru muncul di tab baru di jendela *sketch*. *File* dapat dihapus dari sketsa dengan menggunakan menu tab yang dapat diakses dengan mengklik ikon segitiga kecil di bawah monitor serial satu di sebelah kanan toolbar.

4. *Tools*

a. *Auto Format*

Ini memformat kode Anda dengan baik: yaitu indentasi sehingga membuka dan menutup kurung kurawal sejajar, dan bahwa pernyataan di dalam kurung kurawal semakin menjorok.

b. *Archive Sketch*

Arsipkan salinan sketsa saat ini dalam format *.zip*. Arsip ditempatkan di direktori yang sama dengan sketsa.

c. *Fix Encoding & Reload*

Memperbaiki kemungkinan perbedaan antara pengkodean peta *char editor* dan peta *char* sistem operasi lainnya.

d. *Serial Monitor*

Membuka jendela monitor serial dan memulai pertukaran data dengan papan terhubung pada Port yang saat ini dipilih. Ini biasanya me-reset papan, jika papan mendukung *Reset over port serial*.

e. *Board*

Pilih papan yang Anda gunakan. Lihat di bawah untuk deskripsi berbagai papan.

#### f. *Port*

Menu ini berisi semua perangkat serial (nyata atau virtual) pada komputer Anda. Ini harus secara otomatis refresh setiap kali Anda membuka menu alat tingkat atas.

#### g. *Programmer*

Untuk memilih programmer hardware saat memprogram board atau chip dan tidak menggunakan koneksi *USB-serial onboard*. Biasanya Anda tidak memerlukan ini, tapi jika Anda *burn bootloader* ke mikrokontroler baru, Anda akan menggunakan ini.

#### h. *Burn Bootloader*

Item dalam menu ini memungkinkan Anda *burn bootloader* ke mikrokontroler pada papan Arduino. Ini tidak diperlukan untuk pemakaian papan Arduino atau Genuino secara normal namun berguna jika Anda membeli mikrokontroler ATmega baru (yang biasanya tidak dilengkapi *bootloader*). Pastikan Anda telah memilih papan yang benar dari menu *Board* sebelum *burn bootloader* pada papan target. Perintah ini juga mengatur sekering yang tepat.

#### 4. *Help*

Di sini Anda menemukan akses mudah ke sejumlah dokumen yang disertakan dengan Arduino Software (IDE). Anda memiliki akses ke *Getting Started*, *Reference*, panduan ini ke IDE dan dokumen lainnya secara lokal, tanpa koneksi internet. Dokumen-dokumen itu adalah salinan lokal yang online dan mungkin akan dikirim kembali ke situs online kami.

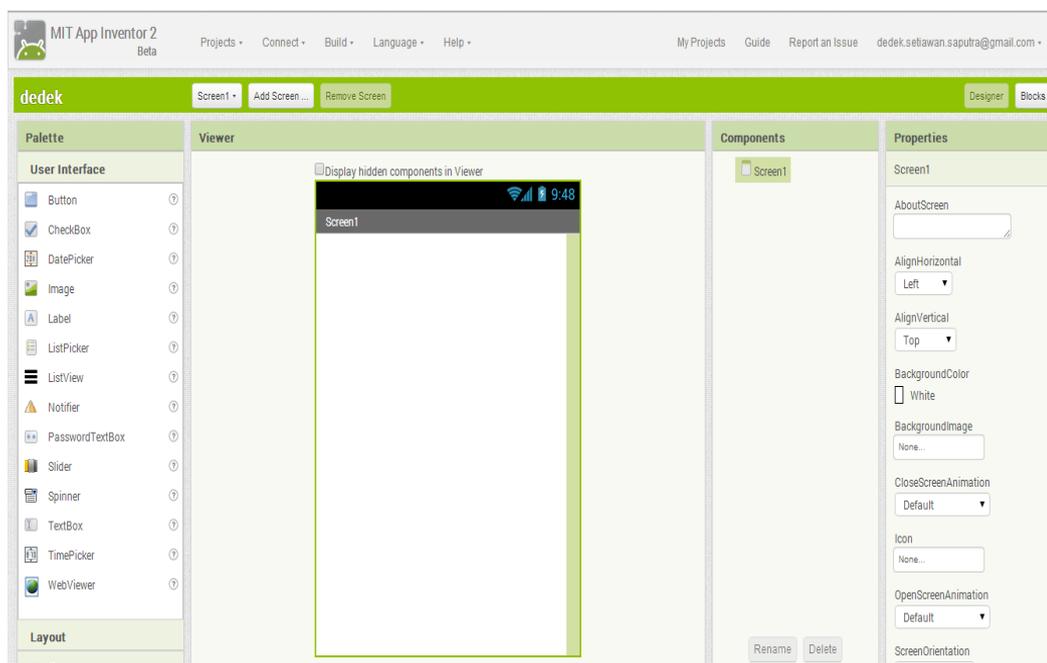
#### a. *Find in Reference*

Ini adalah satu-satunya fungsi interaktif dari menu Bantuan: ia secara langsung memilih halaman yang relevan di salinan lokal Referensi untuk fungsi atau perintah di bawah kursor. (<https://www.arduino.cc/en/Guide/Environment>, n.d.)

### 2.3.2 App Inventor

App inventor adalah salah satu sarana untuk membuat sebuah aplikasi mobile tanpa menggunakan kode program yang rumit. Hal ini karena pemrograman akan dilakukan secara visual. Jadi kita hanya menyusun template visualisasi sesuai dengan algoritma yang diinginkan. Begitu pula dengan tampilan desain yang dibuat, cukup menggunakan metode “click and drag”. App Inventor adalah aplikasi yang pada dasarnya disediakan oleh Google dan sekarang di *maintenance* oleh Massachusetts Institute of Technology (MIT). App Inventor memungkinkan semua orang untuk membuat *software* aplikasi untuk sistem operasi Android. Pengguna dapat menggunakan tampilan grafis GUI dan fitur *drag drop* visual objek untuk membuat sebuah aplikasi dapat berjalan pada sistem operasi Android. Aplikasi App Inventor selesai dibuat pada 12 juli 2010 dan dirilis untuk publik pada 15 desember 2010. Google menghentikan *project* App Inventor pada 31 desember 2011. App Inventor sekarang dipegang oleh MIT *Centre for Mobile Learning* dengan nama MIT App Inventor. (Andi, 2013).

Terlihat dibawah, tampilan awal *project* App Inventor.



Gambar 2.7 Tampilan Awal App Inventor

Sumber: Data olahan sendiri

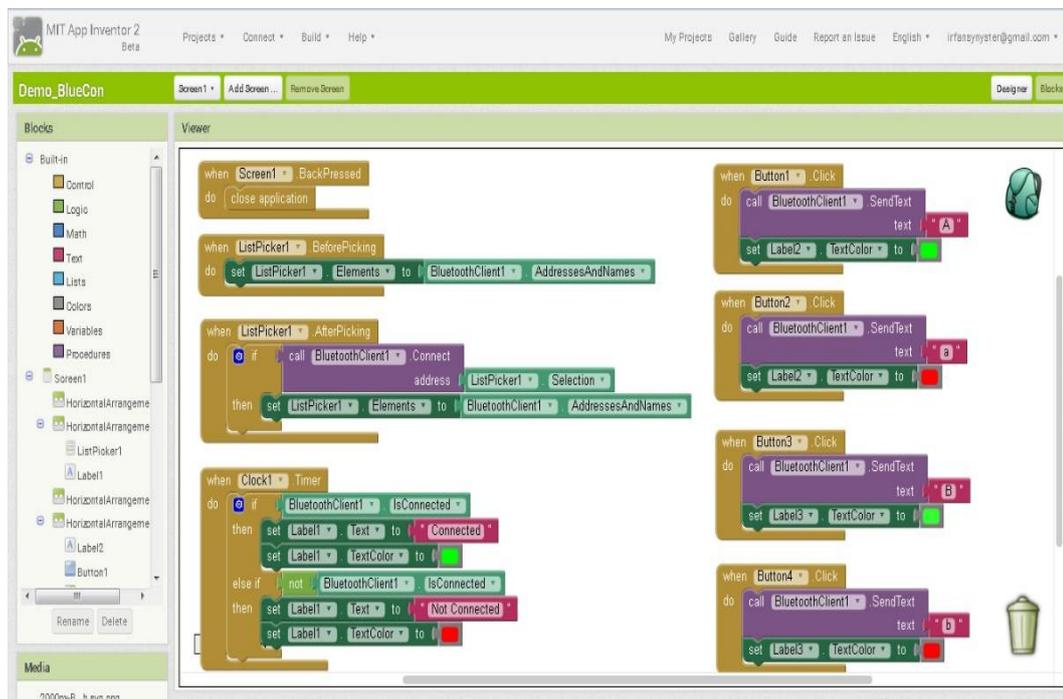
Dalam Andi, (2013: 4-5) Kemampuan App Inventor diantaranya yaitu:

1. Membuat sebuah Aplikasi diperangkat *mobile atau smartphone*
2. Berguna dalam pembuatan prototype
3. Membangun sebuah aplikasi sesuai kepribadian kita atau sesuai dengan keinginan kita.
4. Mengembangkan aplikasi secara lengkap tidak hanya dalam sebuah prototype.
5. Media pembelajaran saat membuat sebuah aplikasi

Aplikasi yang dapat dibuat dengan App Inventor diantaranya yaitu:

1. Aplikasi games.
2. Aplikasi edukasi.

3. Aplikasi berbasis tracking lokasi.
4. Aplikasi sms.
5. Aplikasi kompleks.
6. Aplikasi berbasis web.



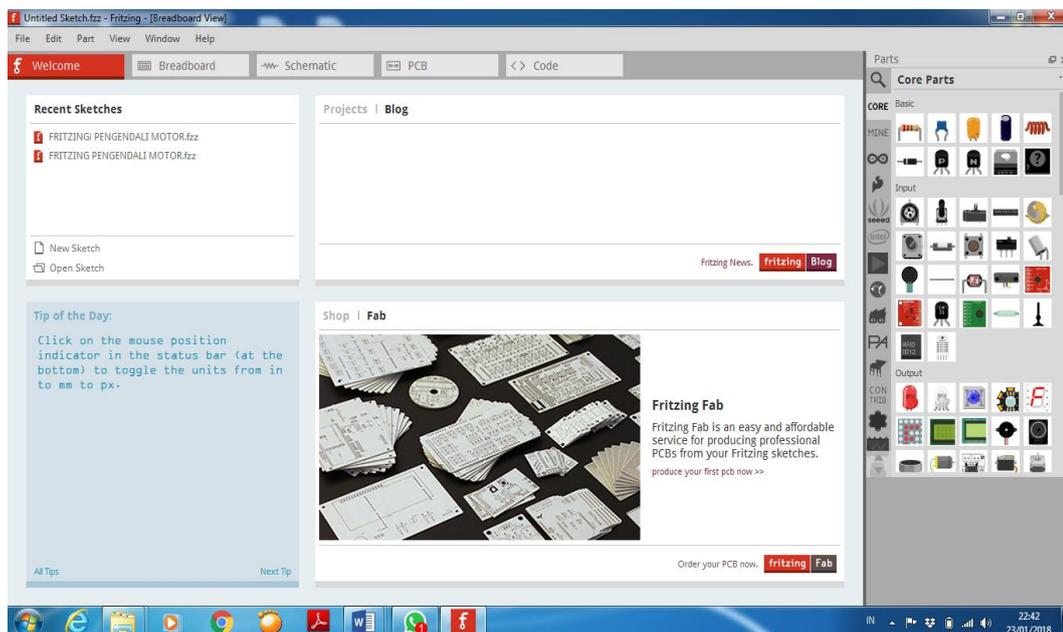
Gambar 2.8 Tampilan menu Blocks pada App Inventor

Sumber: Data Olahan Sendiri

Gambar 2.8 menunjukkan Tampilan pada menu *blocks* yang berisi blok-blok yang sudah tersusun dengan beberapa perintah untuk membuat sebuah aplikasi.

### 2.3.3 Fritzing

Fritzing adalah inisiatif perangkat keras *open source* yang membuat elektronika dapat diakses sebagai bahan kreatif bagi siapa saja. Yang menawarkan perangkat lunak, situs komunitas dan layanan dalam semangat Pengolahan dan Arduino, mendorong ekosistem kreatif yang memungkinkan pengguna mendokumentasikan prototipe mereka, membaginya dengan orang lain, mengajar elektronik di kelas, dan tata letak dan pembuatan PCB profesional. ([Http://fritzing.org/](http://fritzing.org/), n.d.).



Gambar 2.9 Tampilan Awal Fritzing

Sumber: Data Olahan Sendiri

## 2.4 Penelitian Terdahulu

Terdapat beberapa penelitian yang menerapkan sistem pengendali sepeda motor yang telah dikembangkan sebelumnya, perancangan yang berfokus pada penerapan yang berbeda melalui berbagai macam metode yang digunakan.

1. **Oroh, Joyner R Kendekallo, Elia** (2014). Rancang Bangun Sistem Keamanan Motor Dengan Pengenalan Sidik Jari. Perkembangan ilmu pengetahuan dan teknologi semakin meningkat terutama di bidang elektronika, banyak keuntungan yang dapat diperoleh dari perkembangan elektronika tersebut akan tetapi makin berkembangnya teknologi, makin banyak pula tindak kriminal, diantaranya pencurian motor. Dari permasalahan ini, dibuat system keamanan motor melalui pengenalan sidik jari. Sistem yang dibuat menggunakan sensor sidik jari Sm630 sebagai input untuk mendeteksi sidik jari dari pengguna sepeda motor. System ini juga didukung oleh kit arduino uno dengan mikrokontroler ATmega328 sebagai otak untuk mengolah data dari sensor sidik jari ke LCD, sepeda motor dan alarm. Dari pembuatan system ini, dapat disimpulkan bahwa, Hanya akan ada lima pengguna yang dapat mengakses sepeda motor dan system akan menghidupkan alarm saat ada sidik jari yang tidak sesuai menempel pada sensor karena sensor hanya akan berkomunikasi dengan sidik jari yang tersimpan dalam database sensor.

2. **Doni, Fahlepi Roma Widiyanto, Triadi.** (2015). Rancangan Pengaman Sepeda Motor Berbasis Mikrokontroler Atmega16 Dengan Kontrol Android. Langkah dalam meningkatkan keamanan motor selalu dikembangkan oleh perusahaan

pabrikasi motor, seperti menambahkan kunci stang ganda, penutup gerai, dan rem cakram kunci ganda, namun dengan keamanan berlapis seperti itu yang tidak mengikis rasa kendaraan pencuri terlihat. Dari media massa tentang pencurian kendaraan bermotor setiap saat selalu muncul, jadi kita butuh sistem keamanan yang kompleks agar bisa menyulitkan aksi para pencuri. Smartphone Android bisa membuat aplikasi sendiri yang bisa diintegrasikan dengan mikrokontroler dan modul Bluetooth sehingga bisa digunakan untuk pengamanan motor menggunakan gelombang bluetooth. Aplikasi Android pada smartphone akan memberikan perintah kepada mikrokontroler, dan mikrokontroler akan memproses sesuai program dan menghasilkan output yang akan bekerja dengan perintah relay untuk melepas kabel CDI, sehingga saat stop kontak dihidupkan, motor akan sulit dinyalakan. dengan starter listrik atau starter kick, dan juga tanduk dan belokan sinyal akan aktif dalam kondisi ini. Untuk sistem kontrol perangkat pengamanan berkisar antara 30 meter hingga mencapai titik terjauh.

**3. Kholilah, Ika Rafi, Adnan Tahtawi, Al.** (2016). Aplikasi Arduino-Android untuk Sistem Keamanan Sepeda Motor. Sistem keamanan sepeda motor diperlukan untuk mengatasi peningkatan pencurian sepeda motor. Saat ini, solusi yang biasa dilakukan oleh pemilik sepeda motor hanya dengan memakai kunci ganda saja dimana pencuri sudah sangat menguasainya. Untuk itu diperlukan suatu sistem keamanan yang lebih baik. Dalam makalah ini, akan dipaparkan suatu sistem keamanan sepeda motor berbasis Arduino-Android. Sistem keamanan ini berbasis relay dan akan dikendalikan melalui smartphone dengan sistem

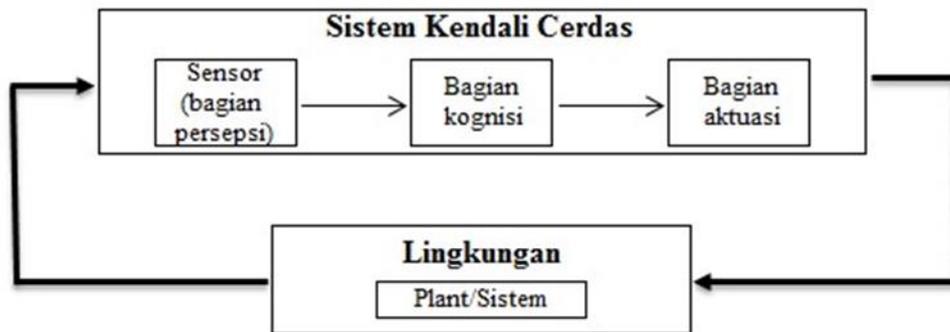
operasi Android v4.4 (KitKat). Sistem komunikasi dirancang dengan menggunakan modul bluetooth HC-06 yang dapat diintegrasikan dengan papan mikrokontroler Arduino Uno. Detail perancangan sistem dijelaskan pada makalah ini. Hasil pengujian menunjukkan jarak maksimal komunikasi bluetooth antara pengendali (*smartphone*) dengan sistem pada sepeda motor yaitu 30 meter.

4. **Rizkidiniah, Fatmah Yamin, Muh Muchlis, Nur Fajriah** (2016). Perancangan Dan Implementasi Prototipe Sistem Gps (*Global Positioning System*) Dan *Sms Gateway* Pada Pencarian Kendaraan Bermotor Berbasis Arduino Uno. Saat ini semakin marak terjadi pencurian kendaraan bermotor. Hal ini dikarenakan oleh minimnya tingkat keamanan yang ada pada kendaraan bermotor, khususnya kendaraan bermotor roda dua. Tujuan utama pembuatan alat ini adalah untuk melacak posisi kendaraan sehingga pemilik dapat mengetahui lokasi kendaraan tersebut dan secara bersamaan memberhentikan mesin motor menggunakan relay dalam bentuk simulasi. Alat ini memanfaatkan teknologi GPS (*Global Positioning System*), Arduino dan Smartphone Android. GPS berfungsi untuk memberikan posisi dimana kendaraan berada. Arduino berfungsi sebagai pusat pengontrolan dari alat ini dimana Arduino Uno mengontrol dan menyimpan segala perintah yang akan dijalankan oleh GPS Shield dan GSM Shield. Masukan dari sistem ini adalah GPS Shield yang berfungsi menangkap titik koordinat letak kendaraan berada dan GSM Shield yang berfungsi untuk mengirimkan dan menerima SMS yang berisikan titik koordinat dan kemudian hasil input-an dari kedua komponen tersebut diolah dalam mikrokontroler. Kemudian mikrokontroler memerintahkan

dinamo untuk mematikan mesin. Alat ini diharapkan dapat membantu mengurangi tingkat pencurian motor.

5. **Sumardi** (2017). Perancangan Sistem Starter Sepeda Motor Menggunakan Aplikasi Android Berbasis Arduino Uno. Penelitian ini untuk merancang sistem starter sepeda motor menggunakan aplikasi berbasis arduino uno yang dapat memudahkan pengguna untuk melakukan pemanasan mesin sepeda motor guna menghindari kerusakan pada mesin. Metodologi menggunakan teknik pengumpulan data yang dilakukan dengan cara bertanya kepada pihak – pihak yang lebih mengerti serta mengumpulkan bahan-bahan berupa buku pustaka yang berkaitan dengan ATmega328, Arduino Uno, serta prosedur menyalakan mesin sepeda motor. Proses dalam pembuatan sistem starter sepeda motor menggunakan metode waterfall dimana tahapan yang dilakukan adalah analisa, desain sistem, penulisan kode program, pengujian program, dan penerapan program. Kesimpulan bahwa sistem starter sepeda motor ini dapat membantu pengguna melakukan perawatan mesin dengan cara melakukan pemanasan mesin dan sistem ini menggunakan sistem starter elektrik.

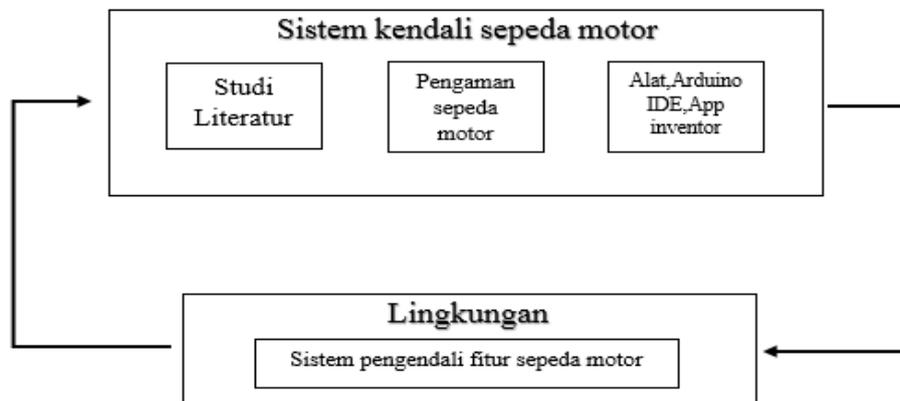
## 2.5 Kerangka Berfikir



Gambar 2.10 Sistem Kendali Cerdas

Sumber: (Wati, 2011)

Berdasarkan kerangka berfikir dari sistem kendali cerdas dalam Wati, (2011: 3) maka dibuatlah kerangka berfikir sebagai berikut:



Gambar 2.11 Kerangka Berfikir Sistem Pengendali Sepeda Motor

Sumber: Data Olahan Sendiri

Berdasarkan Gambar 2.11 menunjukkan Kerangka befikir yang berisi Langkah penelitian diantaranya yaitu

1. Studi literatur tentang referensi yang berhubungan dengan topik penelitian ini. Referensi diperoleh dari buku teks, *e-book*, jurnal penelitian, dan datasheet komponen elektronika yang digunakan.
2. Selanjutnya didapat sebuah pemikiran untuk membuat sebuah sistem pengendali fitur sepeda motor dengan merancang sebuah sistem kendali menggunakan mikrokontroler Arduino, Wemos D1 mini pro, Bluetooth HC-05 dan alat penunjang lainnya.
3. Kemudian merakit sebuah alat sistem kendali, selanjutnya yakni Arduino diprogram menggunakan software Arduino IDE, dan membuat sebuah aplikasi Android untuk *remote control* yang dirancang menggunakan *software* App Inventor.
4. Menghasilkan sebuah sistem pengendali sepeda motor yang bisa digunakan untuk pengaman sepeda motor serta menambahkan fungsi sepeda motor yang masih manual menjadi fungsi yang otomatis.