

**AKSES KONTROL PINTU GARASI OTOMATIS  
MENGUNAKAN RASPBERRY PI BERBASIS ANDROID**

**NASKAH PUBLIKASI TUGAS AKHIR**



**HERMAWAN ARIF BUDIANTORO**

**3115211041**

**PROGRAM STUDI TEKNIK KOMPUTER  
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO  
UNIVERSITAS TEKNOLOGI YOGYAKARTA  
2018**

**AKSES KONTROL PINTU GARASI OTOMATIS  
MENGUNAKAN RASPBERRY PI BERBASIS ANDROID**

**NASKAH PUBLIKASI TUGAS AKHIR**

Diajukan untuk memenuhi salah satu syarat  
Mencapai derajat Sarjana S-1 Program Studi Teknik Komputer



**HERMAWAN ARIF BUDIANTORO**

**3115211041**

**PROGRAM STUDI TEKNIK KOMPUTER  
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO  
UNIVERSITAS TEKNOLOGI YOGYAKARTA  
2018**

**HALAMAN PENGESAHAN  
NASKAH PUBLIKASI TUGAS AKHIR MAHASISWA**

Judul Tugas Akhir :  
**AKSES KONTROL PINTU GARASI OTOMATIS  
MENGUNAKAN RASPBERRY PI BERBASIS ANDROID**

Judul Naskah Publikasi :  
**AKSES KONTROL PINTU GARASI OTOMATIS  
MENGUNAKAN RASPBERRY PI BERBASIS ANDROID**

Disusun oleh:  
**Hermawan Arif Budiantoro**  
**3115211041**

Mengetahui,

**DEWAN PENGUJI**

Nama

Jabatan

Tanda Tangan

Tanggal

**Dr. Enny Itje Sela, S.Si., M.Kom.**  
NIK. 121116059

Pembimbing

Naskah Publikasi Tugas akhir ini telah diterima sebagai salah satu syarat untuk mencapai derajat Sarjana S-1 Program Studi Teknik Komputer

Yogyakarta, .....  
Ketua Program Studi Teknik Komputer  
Fakultas Teknologi Informasi dan Elektro, Universitas Teknologi Yogyakarta

**Ikrima Alfi, S.T., M.Eng.**  
NIK. 120909013

## **PERNYATAAN PUBLIKASI**

Yang bertanda tangan di bawah ini, saya:

Nama : Hermawan Arif Budiantoro  
NIM : 3115211041  
Program Studi : Teknik Komputer  
Fakultas : Teknologi Informasi Dan Elektro

menyatakan bahwa Naskah Publikasi ini hanya akan dipublikasikan di JURNAL Tekno SAINS FST, UTY, dan tidak dipublikasikan di jurnal yang lain. Demikian surat pernyataan ini dibuat dengan sebenar-benarnya.

Yogyakarta, 9 Maret 2018  
Penulis,

Hermawan Arif Budiantoro  
3115211041

# AKSES KONTROL PINTU GARASI OTOMATIS MENGUNAKAN RASPBERRY PI BERBASIS ANDROID

Hermawan Arif Budiantoro<sup>[1]</sup>  
Enny Itje Sela<sup>[2]</sup>

Program Studi Teknik Komputer  
Fakultas Teknologi Informasi dan  
Elektro  
Universitas Teknologi Yogyakarta

<sup>[1]</sup>arifhermawan009@gmail.com  
<sup>[2]</sup>ennyjsela@uty.ac.id

## Abstrak

Garasi adalah suatu tempat atau ruangan untuk menyimpan mobil dan perlengkapan lainnya, agar terhindar dari pencurian dan untuk melindungi mobil dari cuaca terik matahari ataupun air hujan yang dapat merusak atau memperpendek usia kendaraan. Banyaknya aktifitas yang tidak lepas dari keberadaan pintu garasi di mana harus membuka atau menutup pintu garasi dengan menarik atau mendorong berulang-ulang kali membuat terasa enggan untuk melakukannya. Pintu garasi memiliki bermacam model dalam jenis kunci seperti diketahui pada zaman dahulu kunci bentuknya lobang kunci cukup besar dan juga didukung oleh kuncinya yang cukup besar hal tersebut tidak luput juga dari tindak kriminal seperti pengandaan kunci yang marak terjadi. Berdasarkan permasalahan tersebut dibuatlah sebuah akses kontrol pintu garasi otomatis menggunakan Raspberry Pi berbasis android. Sistem kontrol pintu garasi yang dibuat dikendalikan melalui aplikasi android menggunakan username dan password dengan media wireless sebagai penghubung antara smartphone android dengan sistem kendali garasi. Sistem kendali garasi yang dibuat mempunyai beberapa fitur yaitu, kendali pintu garasi, kendali lampu garasi, kendali kipas sirkulasi udara garasi, dan kamera keamanan pemantau garasi.

**Kata Kunci:** Kendali Garasi, Android, Raspberry Pi, Wireless, Smart Garage.

## 1. Pendahuluan

Perkembangan teknologi di bidang pengontrolan sudah semakin maju dan berkembang, sehingga memungkinkan manusia untuk menciptakan sistem kontrol otomatis untuk mengerjakan pekerjaan sehari-hari. Perkembangan itu telah membawa suatu perubahan pola hidup manusia untuk bekerja dengan cepat, efektif dan efisien. Garasi adalah suatu tempat atau ruangan untuk menyimpan mobil dan perlengkapan lainnya, agar terhindar dari pencurian dan untuk melindungi mobil dari cuaca terik matahari ataupun air hujan yang dapat merusak atau memperpendek usia kendaraan. Banyaknya aktifitas yang tidak lepas dari keberadaan pintu garasi di mana harus membuka atau menutup pintu garasi dengan menarik atau mendorong berulang-ulang kali membuat terasa enggan untuk melakukannya.

Menurut Aris Triwiyanto (2015), Sistem kendali merupakan sistem pengontrol suatu objek tertentu yang dalam hal ini akan diterapkan pada *switching* yang terdapat pada garasi, sistem kendali ini mempelajari pola atau kebiasaan dari aktivitas manusia. Penelitian yang dikembangkan tentang sistem kendali otomatisasi pintu garasi saat ini masih belum efektif. Penelitian yang dilakukan oleh Dadan Nurdin Bagenda dan Wildhan Adityoso (2014), tentang keamanan pintu garasi menggunakan *keypad* dan sensor berbasis mikrokontroler, sistem keamanan yang dapat bekerja secara otomatis membuka pintu garasi dengan memasukan *password* melalui *keypad* dan sensor yang berfungsi mendeteksi objek manusia yang berada di garis pintu garasi. Penelitian selanjutnya yang dilakukan Erizki Ridatullah (2017), menggunakan mikrokontroler Arduino sebagai pusat kendali dan bluetooth sebagai media komunikasi untuk pengendalian sistem pintu garasi. Kelemahan dari beberapa penelitian sebelumnya yaitu dari segi jarak untuk mengakses atau mengendalikan garasi.

Melihat situasi tersebut penulis mempunyai ide untuk merancang alat yang memiliki fungsi yang sama yaitu untuk mengakses pintu garasi akan tetapi menggunakan teknologi yang memiliki nilai tepat guna tinggi. Alat yang akan dibuat menggunakan sistem *single board computer* (SBC) Raspberry Pi yang berfungsi sebagai pusat pengontrolan kendali garasi otomatis yang dapat dikendalikan melalui jarak kurang lebih 50 meter melalui media *wireless*.

Berdasarkan latar belakang di atas, penulis tertarik untuk merancang sebuah alat dengan judul "Akses Kontrol Pintu Garasi Otomatis Menggunakan Raspberry Pi Berbasis Android".

## 2. Tinjauan Pustaka

Dalam memperoleh data sekunder, penulis menggunakan studi pustaka, studi *online*. Studi pustaka dilakukan dengan membaca, literatur dan mempelajari dokumen - dokumen yang berkaitan dengan permasalahan yang diteliti. Dari metode tersebut diperoleh keterangan dan konsep teoritis dengan cara menganalisis data pada pustaka dan bacaan lain yang dapat membantu dalam pemecahan masalah.

Penelitian yang dilakukan oleh Astri Anggraini (2016) dengan judul "Rancang Bangun *Open/Close* Pintu Ruang Menggunakan *Voice Reconition* Berbasis Raspberry Pi", sistem ini berisi tentang sebuah alat pembuka pintu ruangan otomatis menggunakan sistem *single board computer* (SBC) Raspberry Pi yang berfungsi sebagai pusat pengontrol sistem pembuka dan penutup pintu ruangan. Sebagai input menggunakan sistem pengenalan suara (*voice reconition*) oleh mikrofon yang kemudian akan diterjemahkan oleh Raspberry Pi untuk menjalankan motor DC sebagai penggerak pintu.

Penelitian yang dilakukan oleh Haryati (2015) dengan judul "Sistem Kontrol Gerbang Garasi berbasis *Web* Menggunakan Raspberry Pi", sistem ini berisi tentang sebuah sistem kontrol pintu yang dapat diremote melalui web menggunakan Raspberry Pi, yang mana Raspberry Pi merupakan *single board computer* (SBC) yang dijadikan sebagai *web server* sekaligus pengendali dan terkoneksi ke jaringan *wireless* sehingga user dapat mengaksesnya melalui *smartphone*.

Penelitian yang dilakukan Sunu Hasta Wibowo (2014) dengan judul "Simulasi Pengontrolan Pintu Garasi Otomatis", Sistem pengontrolan ini bekerja apabila kendaraan yang ingin masuk menekan polisi tidur teras garasi dan mengenai sensor maka pintu garasi akan membuka, selanjutnya setelah mengenai polisi tidur yang ada di dalam maka pintu garasi akan menutup.

Dengan adanya tinjauan pustaka, metode dan sebagian alat yang digunakan pada umumnya sama, tetapi yang membedakan dari segi *input* (masukan) melalui perintah dari aplikasi *android* dikirim melalui media *wireless* ke *single board computer* (SBC) Raspberry Pi selanjutnya diproses untuk menjalankan motor DC sebagai penggerak pintu garasi.

### 2.1 Landasan Teori

#### 2.2.1 Akses Kontrol

Akses kontrol adalah pusat keamanan dalam sebuah komputer dengan cara membatasi pengguna untuk mengakses sumber daya, dan dimotivasi oleh kebutuhan untuk membocorkan akses dalam memperoleh informasi yang tersedia di sumber daya.

Mekanisme keamanan sistem komputer yang paling penting berusaha untuk menjamin arus informasi tetap aman. Berarti tidak ada aliran yang tidak sah bekerja di dalam sistem komputer sehingga informasi yang berjalan di dalam sistem tersebut harus benar-benar aman. Seperti dalam pemerintahan ataupun sistem kemiliteran, keamanan akan mensyaratkan bahwa proses dalam melakukan pentransferan data pada sistem keamanan yang mempunyai proteksi tingkat tinggi ke para pengguna sampai ketinggian yang lebih rendah, dapat melakukan pentransferan serta melakukan pencegahan untuk membaca data secara langsung sehingga file yang mempunyai klasifikasi rahasia akan tetap terjaga dan juga dihambat secara tidak langsung dalam mengakses informasi tersebut. Pencegahan juga dilakukan untuk mencegah para pengguna yang cerdik dengan cara berkolaborasi dengan pengguna yang memiliki wewenang untuk mengakses informasi tersebut (Rotenberg, 1974).

Menurut Bagas Bayu (2014), akses kontrol pintu (*Door Acces Control*), adalah sebuah sistem yang dapat atau untuk membatasi pengguna untuk mengakses suatu ruangan dengan menempatkan sistem perangkat kontrol pada pintu. Dalam Akses Kontrol Pintu, kontrol akses merujuk pada praktek membatasi pintu masuk ke properti, bangunan, atau ruang untuk orang yang berwenang.

#### 2.2.2 Garasi

Garasi adalah suatu tempat atau ruangan untuk menyimpan mobil dan perlengkapan lainnya, agar terhindar dari pencurian dan untuk melindungi mobil dari cuaca terik matahari ataupun air hujan yang dapat merusak atau memperpendek usia kendaraan. Pintu garasi yang ada sekarang ini kebanyakan pengoperasiannya masih secara manual. Seiring perkembangan teknologi, banyak cara yang dapat dilakukan untuk mempermudah manusia dalam melakukan berbagai aktivitas (Sunu Hasta Wibowo, 2014).

#### 2.2.3 Raspberry Pi

Raspberry Pi adalah komputer berukuran kartu kredit yang dikembangkan di Inggris oleh yayasan Raspberry Pi dengan tujuan untuk mempromosikan pengajaran ilmu pengetahuan dasar komputer di

sekolah. Raspberry Pi diproduksi melalui lisensi manufaktur yang berkaitan dengan Elemen 14 atau Premier Farnell dan RS komponen.

Raspberry Pi bersifat *open source* berbasis Linux, Raspberry Pi dapat dimodifikasi sesuai kebutuhan penggunaannya. Sistem operasi utama Raspberry Pi menggunakan Debian GNU/Linux dan bahasa pemrograman Python. Salah satu pengembang OS untuk Raspberry Pi telah meluncurkan sistem operasi yang dinamai Raspbian. Raspbian diklaim mampu memaksimalkan perangkat Raspberry Pi. Sistem operasi tersebut dibuat berbasis Debian yang merupakan salah satu distribusi Linux OS. (Rahman, E., 2014.). Raspberry Pi pada penelitian ini digunakan sebagai pusat sistem kendali garasi.

#### 2.2.4 Database

Menurut Waljiyanto (2003), *database* atau memiliki istilah basis data merupakan suatu kumpulan data yang saling berhubungan dan berkaitan dengan subjek tertentu pada tujuan tertentu pula, hubungan antardata ini dapat dilihat oleh adanya *field* ataupun kolom.

Menurut Prahasta (2002), *database* itu didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil, dan dicari secara cepat.

Menurut Kusri (2007), basis data adalah kumpulan data yang saling berelasi. Data merupakan fakta mengenai obyek, orang, dan lain-lain. Data dinyatakan dengan nilai (angka, deretan karakter, atau simbol).

*Database* digunakan dalam penelitian ini digunakan untuk ruang penyimpanan data pengguna atau pemilik garasi berupa histori pengakses garasi, *username* dan *password* pengguna garasi.

#### 2.2.5 Android

Android adalah sistem operasi *mobile* yang dikembangkan oleh Google.. Sistem operasi android didasarkan pada kernel Linux terbuka (*open source*), yang berarti pengembang dapat memodifikasi dan menyesuaikan OS untuk setiap *smartphone*. *Smartphone* android biasanya datang dengan beberapa aplikasi bawaan dan juga mendukung program pihak ketiga. Pengembang dapat membuat program untuk android menggunakan SDK (*Software Developer Kit*) android secara gratis. Program android ditulis di Java dan dijalankan melalui Google "Davlik" *virtual machine*, yang dioptimalkan untuk perangkat mobile. (Siahaan, V., 2015). Penggunaan android dalam penelitian ini sebagai sistem operasi aplikasi untuk mengendalikan garasi melalui *smartphone*.

#### 2.2.6 Wireless

*Wireless* yakni sebuah jaringan nirkabel atau tanpa kabel yang menggunakan udara sebagai media penghubung transmisinya guna menghantarkan gelombang elektromagnetik maupun data. *Wireless* mempunyai cara kerja karena komputer mempunyai *built transreceiver* seperti *wakly-talky*. *Transreceiver* yang disebut dengan adapter *wireless*. Adapter *wireless* melakukan sejumlah pekerjaan, yang pertama, mendeteksi apakah terdapat jaringan *wireless* di sekitar komputer melalui radio dan juga tuning menghubungkan penerima untuk mendeteksi setiap ada sinyal yang masuk.

Setelah ada sinyal terdeteksi, untuk menghubungkannya yaitu melalui *signin* dan otentikasi pengguna. Apapun data yang dikirimkan dari komputer diubah melalui adaptor *wireless*, dari bentuk digital (0s & 1s) menjadi sinyal radio (bentuk analog). Sinyal radio *wireless* yang disalurkan biasanya mempunyai frekuensi lebih dari 2,4 GHz diterima oleh sebuah router *wireless* ataupun sebuah *wireless* adapter. (Zam, E. 2012. *Wireless Hacking: Cari dan eksploitasi kelemahan jaringan wireless*. PT. Elexmedia Komputindo: Jakarta.). Penggunaan *wireless* pada penelitian ini sebagai komunikasi tanpa kabel antara *smartphone* android dengan sistem kendali garasi.

#### 2.2.7 Python

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional.

Fitur yang tersedia pada python salah satunya adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Seperti halnya pada bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai bahasa skrip meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa skrip. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat

berjalan di berbagai *platform* sistem operasi. (Gunadi, H., 2016). Fungsi bahasa pemrograman python pada penelitian ini untuk membuat *script* program keseluruhan sistem kendali garasi pada Raspberry Pi.

### 3. Metode Penelitian

#### 3.1 Obyek Penelitian

##### 3.1.1 Perangkat Keras

Perangkat keras (*hardware*) merupakan sistem komputer yang digunakan dalam pembuatan prototipe kendali garasi, alat yang digunakan dalam Tugas Akhir ini adalah:

##### a. Laptop Gateway NV49C62n

Dalam penelitian ini, perangkat keras yang digunakan untuk menyusun laporan, *database server* dan membuat antarmuka adalah komputer portabel dengan spesifikasi sebagai berikut:

Nama	: Gateway NV49C62n
Operating System	: Windows 8.1 32 Bit
Prosesor	: Intel® Core™ i3-380M Processor (2.53 GHz)
RAM	: 2 GB DDR3 SDRAM PC-8500
Tipe Memory	: DDR3
Hard Drive	: 500 GB
Graphic Processor	: Intel® Graphics Media Accelerator HD 729MB

##### b. Raspberry Pi 3 Model B

Sebagai pusat kontrol kendali garasi otomatis ini menggunakan *single board computer* (SBC) Raspberry Pi 3 Model B dengan spesifikasi sebagai berikut:

Chipset	: Broadcom BCM2837
CPU	: 4 x ARM Cortex – A53, 1.2GHz
GPU	: Broadcom Cideocore IV
Memori	: 1GB LPDDR2 (900MHz)
Sistem Operasi	: Boots fro microSD <i>card</i> , berjalan di versi Linux
Jaringan	: 10/100 Ethernet, 2.4GHz 802.11n <i>wireless</i>
Penyimpanan	: Micro SD
GPIO	: 40 – pin header, populated
Ports	: HDMI, 3.5mm jack audio analog, 4 x usb2.0, Ethernet, <i>Camera Serial Interface</i> (CSI), <i>Display Serial Interface</i> (DSI).

##### 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 tidak menyertakan colokan DC berjenis Barrel Jack, dan dihubungkan ke komputer menggunakan *port* USB Mini-B. Arduino Nano dirancang dan diproduksi oleh perusahaan Gravitech. Arduino NANO pada penelitian ini dipakai untuk komunikasi serial dengan Raspberry Pi untuk membantu menggerakkan motor servo.

##### d. Modul Stepdown LM2596S

Modul konverter DC ke DC (DC-DC *Converter*) ini menggunakan IC LM2596S yang merupakan *Integrated Circuit* (IC) untuk mengubah tingkatan tegangan (*voltage level*) arus searah / Direct Current (DC) menjadi lebih rendah dibanding tegangan masukannya. Tegangan masukan (*input voltage*) dapat dialiri tegangan berapa pun antara 3 Volt hingga 40 Volt DC, yang akan diubah menjadi tegangan yang lebih rendah di antara 1,5 Volt hingga 35 Volt DC. Tegangan keluaran yang diinginkan dapat diatur dengan memutar sekrup pada potensiometer (sekrup kuningan pada komponen elektro yang berwarna biru), dengan catatan perbedaan tegangan antara tegangan masukan dengan tegangan keluaran minimal 1,5 Volt (contoh: dari 12V bisa ke tegangan berapapun antara 1,5 Volt hingga 10,5 Volt). Modul *stepdown* LM2596S yang digunakan dalam penelitian ini sebagai penurun tegangan dari 19 Volt DC menjadi 5 Volt DC sesuai kebutuhan konsumsi daya prototipe kendali garasi.

##### e. Motor Servo

Motor servo sebuah motor DC dengan sistem umpan balik tertutup, di mana posisi rotor-nya akan

diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor DC, serangkaian gear, potensiometer, dan rangkaian kontrol. Motor servo digunakan dalam penelitian ini sebagai prototipe penggerak pintu garasi

#### f. Solenoid Door lock

Solenoid *door lock* adalah sensor kunci otomatis yang dapat dikontrol oleh Arduino. Mudah cara penggunaannya, sensor ini dapat diaplikasikan di pintu otomatis atau pintu listrik. Penggunaan solenoid *door lock* pada penelitian ini sebagai kunci otomatis pintu garasi. Berikut di bawah ini gambar Solenoid *door lock*.

#### g. Webcam

Webcam atau *camera web* adalah nama sebutan untuk kamera yang dihubungkan pada komputer agar dapat dilihat dan melihat melalui aplikasi pemanggilan video. Webcam yang dipakai dalam penelitian ini mempunyai resolusi 0.48 Megapixel (640x480). Penggunaan webcam pada penelitian ini berfungsi sebagai prototipe kamera keamanan CCTV.

#### h. Modem Huawei HG8245A

Modem Huawei HG8245A pada penelitian ini digunakan sebagai komunikasi jaringan tanpa kabel atau *wireless* untuk menghubungkan antara *smartphone* dengan sistem kendali garasi. Modem Huawei HG8245A mempunyai spesifikasi sebagai berikut:

- 1 Port Buat USB
- 2 Port Telepon / WAN
- Port Ethernet / RJ45 / LAN
- 1 Port Konektor SC /
- 1 WAN Bisa Beberapa SSID Wi-Fi
- Support IPv4 dan IPv6 Dual Stack
- 802.11n, 300Mbps WLAN , Routing , SOHO

#### i. Smartphone Fujitsu Arrows NX F-01F

*Smartphone* yang dipakai untuk menguji aplikasi android yang telah dibuat, menggunakan *smartphone* merk Fujitsu Arrows NX F-01F dengan spesifikasi sebagai berikut:

- OS : Android OS v4.4.2 (Android KitKat)
- Processor : Quad Core 2.2 GHz, Krait 400
- RAM : 2 GB
- GPU : Adreno 330 @ 450 MHz
- Chipset : Qualcomm MSM8974-AA Snapdragon 800

#### j. Adaptor Charger Laptop Hipro

Sistem manajemen *power* untuk kendali garasi menggunakan adaptor *charger* laptop merk Hipro dengan tegangan 19 Volt DC dengan daya 3.42 Amper. Modul *stepdown* LM2596 digunakan sebagai penurun tegangan dari 19 Volt DC menjadi 5 Volt DC sebagai konsumsi daya yang tepat untuk prototipe kendali garasi.

### 3.1.2 Perangkat Lunak

Perangkat lunak adalah aplikasi komputer yang digunakan untuk membuat prototipe kendali garasi, perangkat lunak tersebut yaitu:

#### a. Sistem Operasi Windows 8.1 32-bit

Sistem operasi ini berfungsi sebagai antarmuka pengguna dengan perangkat keras komputer dalam melakukan kendali.

#### b. NOOBS

Sistem operasi yang digunakan Raspberry Pi untuk membuat program Python.

#### c. Embercadero Delphi XE8

RAD Studio Delphi XE8 digunakan untuk menulis dan mendesain antarmuka aplikasi Android.

#### d. Xampp-win32

Xampp-win32 digunakan sebagai *database server* untuk menyimpan *history* akses garasi, *username* dan *password* pengguna.

#### e. Arduino IDE

Arduino IDE adalah *software compiler* khusus yang dipakai untuk melakukan pemrograman Arduino.

#### f. PuTTY

PuTTY digunakan untuk mengakses terminal sistem operasi di dalam Raspberry Pi dari melalui Windows dengan memanfaatkan protokol jaringan seperti SSH dan Telnet untuk mengaktifkan sesi *remote* pada komputer.

### 3.2 Metode Penelitian

Beberapa tahap metode penelitian yang dilakukan dalam pembuatan prototipe kendali garasi ini adalah sebagai berikut:

#### 3.2.1 Pengumpulan Data

##### a. Observasi

Pengumpulan data dengan metode observasi dilakukan dengan cara mengamati perangkat keras dan perangkat lunak pendukung penelitian pembuatan akses kontrol pintu garasi otomatis menggunakan Raspberry Pi berbasis android.

##### b. Studi pustaka

Studi Pustaka dilakukan dengan cara mengkaji buku-buku yang memiliki relasi dengan pelaksanaan Penelitian Tugas Akhir, pencarian jurnal-jurnal dari internet dan mengunduh materi-materi dari internet.

#### 3.2.2 Perancangan alat dan aplikasi

Perancangan ini menjelaskan tentang tahapan atau langkah – langkah untuk merancang sistem dengan perangkat keras maupun perangkat lunak yang dibutuhkan untuk penelitian tersebut. Pada perancangan alat dilakukan analisis kebutuhan untuk menentukan kebutuhan penggunaan komponen elektronika dalam menunjang sistem akses kontrol pintu garasi yang akan dibuat.

Perancangan diagram alir dibuat untuk merancang aplikasi kontrol pintu garasi berbasis android yang berguna untuk menyusun tahapan-tahapan tentang bagaimana jalannya aplikasi tersebut.

Perancangan antarmuka bertujuan untuk merancang sistem antarmuka perangkat lunak yang efektif, siap digunakan, dan hasilnya sesuai dengan kebutuhan pengguna dalam menggunakan aplikasi yang dibuat. Perancangan aplikasi android menggunakan aplikasi Delphi XE8.

Pembuatan basis data digunakan untuk mempermudah pengolahan data yang ada pada sistem. Data yang ada mencakup data pengakses pintu garasi berupa histori pengakses garasi, *username* dan *password*.

#### 3.2.3 Implementasi Alat dan Aplikasi

Implementasi alat dan aplikasi akses kontrol pintu garasi menggunakan Raspberry Pi berbasis android yang sebelumnya hanya suatu rancangan dibuat menjadi sebuah prototipe kendali garasi. Implementasi dilakukan bukan hanya pembuatan alat dan aplikasi namun juga penyusunan basis data dan penerapan sistem.

#### 3.2.4 Pengujian Alat dan Aplikasi

Pada tahap ini akan dilakukan pengujian apakah alat dapat membaca dan dapat berkomunikasi dengan *server* serta basis data yang telah dibuat. Apabila sistem dapat membaca maka sistem akan memberikan data akses kepada pengguna yang akan masuk ke garasi.

### 3.3 Kerangka Kerja Penelitian

#### 3.3.1 Studi Literatur

Studi literatur dilaksanakan dengan melakukan studi kepustakaan untuk memahami teori yang berasal dari buku, *paper*, jurnal, maupun artikel. Kegiatan ini dilakukan untuk mempelajari hal-hal yang berkaitan dengan proses penelitian secara umum, meliputi literatur tentang kontrol pintu garasi, android dan Raspberry Pi.

#### 3.3.2 Pengumpulan Data

Pengumpulan data-data melalui media cetak maupun elektronik yang dibutuhkan dalam perancangan dan pembuatan akses kontrol pintu garasi Raspberry Pi dan android. Data yang dikumpulkan dikelola sedemikian rupa sehingga tersusun dengan baik dan dapat digunakan dengan semestinya.

### 3.3.3 Rancangan Sistem dan Gambaran Umum Sistem

Sistem akses kendali pintu garasi yang akan dibuat diterapkan arsitektur atau konsep kontrol mulai dari segi *input* (masukan) melalui perintah di *smartphone*. *Smartphone* dengan aplikasi android yang akan dibuat merupakan *input-an* perintah kendali garasi. Media pengantar perintah dari *smartphone* ke sistem kendali garasi menggunakan komunikasi tanpa kabel (*wireless*) menggunakan perangkat modem Huawei HG8245A.

### 3.3.4 Studi Pustaka

Dilakukan dengan cara mengkaji dari buku-buku yang memiliki relasi dengan pelaksanaan Tugas Akhir, mencari jurnal-jurnal dari internet dan mengunduh materi-materi dari internet kemudian dijadikan sebagai bahan referensi dalam penelitian.

### 3.3.5 Implementasi

Implementasi dilakukan bukan hanya pembuatan alat dan aplikasi namun juga penerapan sistem. Dalam rencana penerapan sistem yang terkomputerisasi yakni agar sistem siap dioperasikan, maka perlu diadakan kegiatan-kegiatan dari penerapannya. Implementasi bertujuan untuk meninjau apakah program dan alat bisa disinkronisasikan dengan baik.

### 3.3.6 Pengujian

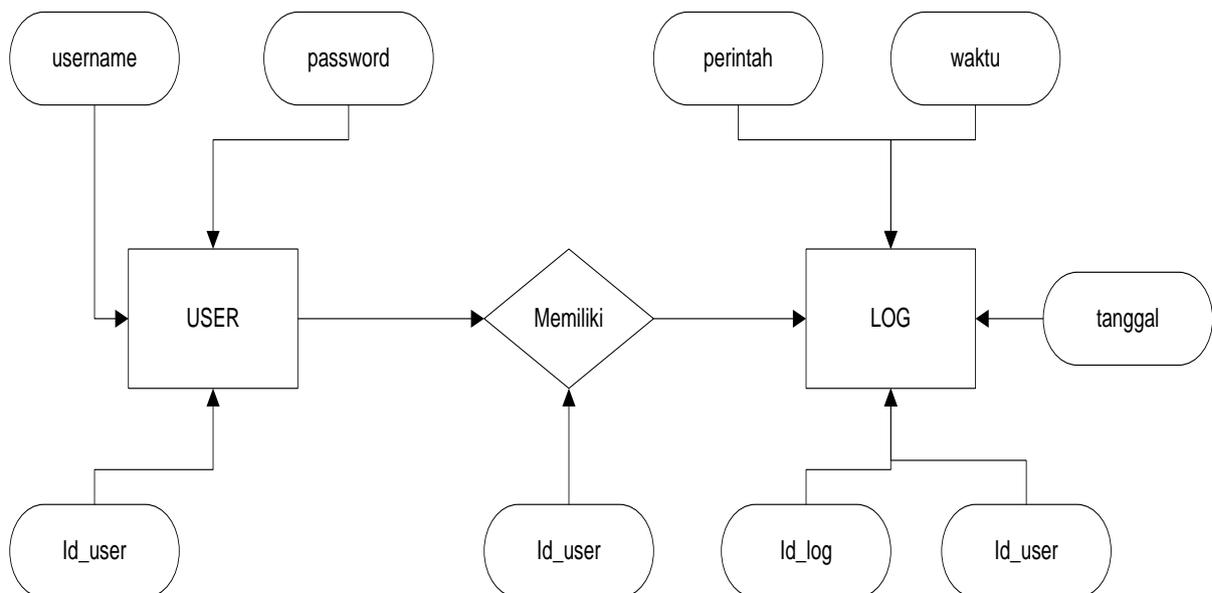
Pengujian sistem dilakukan setelah implementasi akses kontrol pintu garasi otomatis menggunakan Raspberry Pi berbasis android selesai, sebagai evaluasi sistem tentang keamanan, akurasi dan efisiensi kendali pintu garasi otomatis menggunakan Raspberry Pi berbasis android sehingga dapat menambah keamanan, efisien waktu dan menghemat tenaga.

## 4. Perancangan Sistem

### 4.1 Rancangan Database Server

Rancangan database *server* merupakan perancangan basis data sebagai media penyimpanan data yang digunakan dalam penelitian ini. Data yang disimpan adalah data *user* atau pemilik garasi dan data *loger* perintah kontrol pintu garasi. Kedua data tersebut disimpan dalam komputer *server* menggunakan aplikasi *Database Management Server* (DBMS) MySQL.

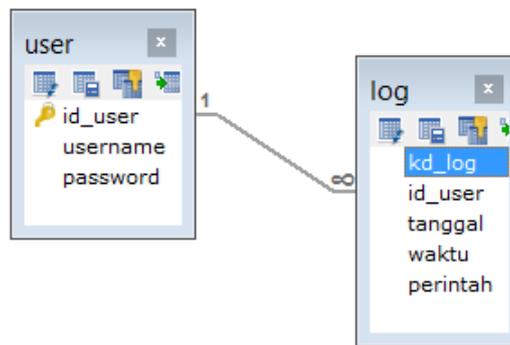
#### 4.1.1 Entity Relationship Diagram (ERD) sistem kendali garasi



Gambar 1. ERD sistem kendali garasi

Gambar 1 menjelaskan bahwa keterhubungan antar entitas untuk mengatur alir data sistem. Entitas *user* memiliki atribut *id\_user* sebagai key atau kode unik yang digunakan untuk merealisasikan dengan entitas *log*. Dalam satu entitas *user* akan memiliki banyak perintah yang akan disimpan di entitas *log*.

### 4.1.2 Relasi Antar Tabel *Database* Sistem Kendali Garasi



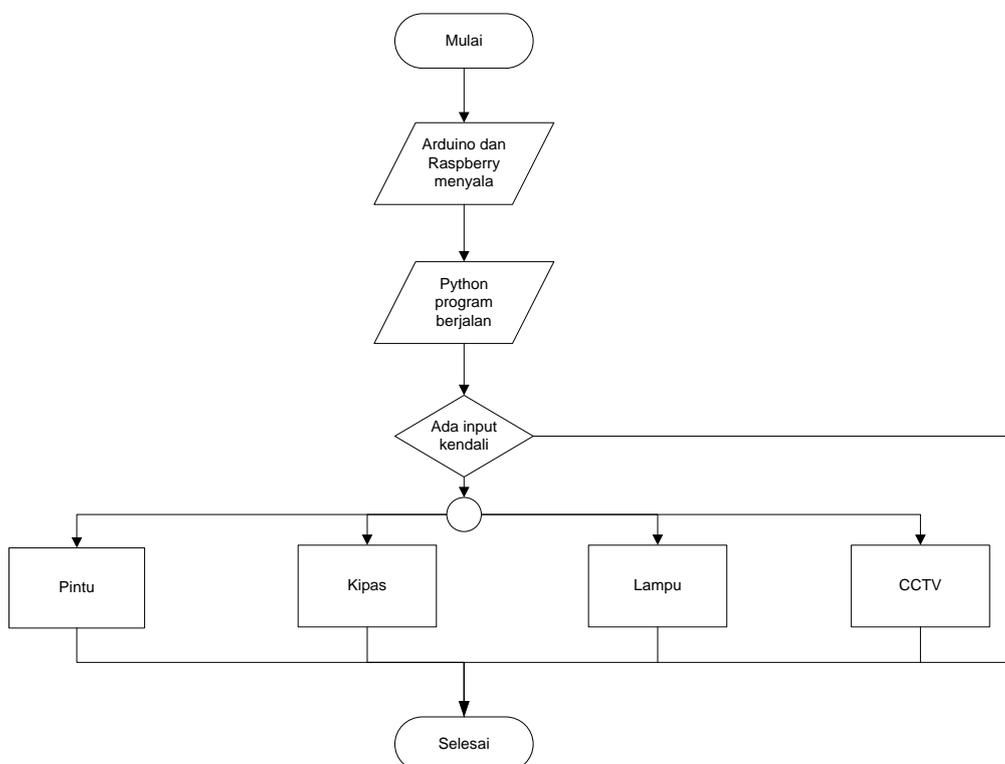
**Gambar 2.** relasi antar table *database* sistem kendali garasi

Gambar 2 menjelaskan bahwa relasi antara tabel *user* dan tabel *log* yang terhubung dengan kolom *id\_user*. Pada tabel *user* yang menjadi unik key atau kunci unik adalah kolom *id\_user* yang merupakan kunci primer dari tabel tersebut. Sedangkan tabel *log* mempunyai kunci primer kode log, tetapi tabel *log* juga mempunyai kolom *id\_user* (*foreign key*) untuk dapat terhubung dengan tabel *user*.

### 4.1.3 Rancangan Perangkat Keras

Dalam membangun pintu garasi otomatis diperlukan beberapa perangkat keras yang dipakai untuk memproses *input* kendali menjadi *output* kontrol pintu garasi. Selain itu, penulis menambahkan beberapa fitur untuk sistem kendali garasi yaitu, kendali lampu garasi, kipas garasi, kamera keamanan. Berikut beberapa perancangan perangkat keras yang akan dilakukan dalam penelitian ini.

#### 4.1.3.1 Flowchart Perangkat Keras



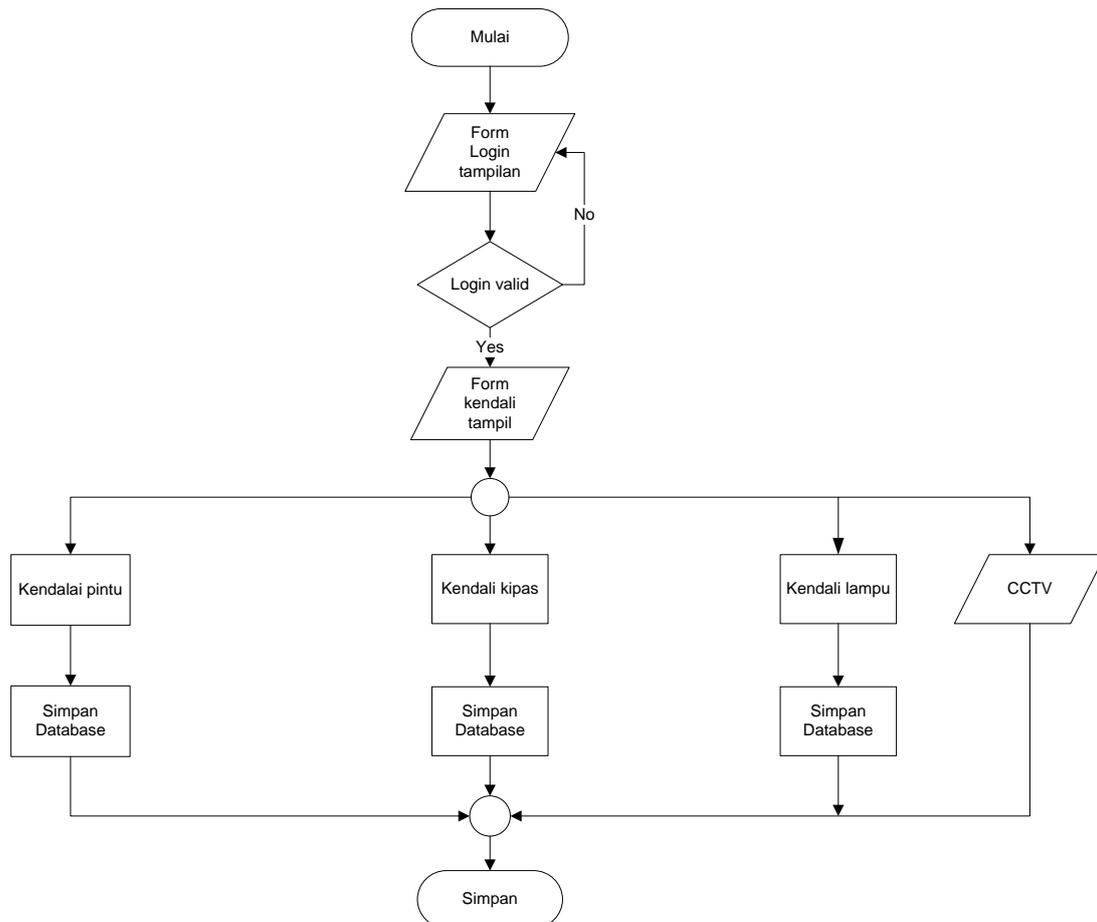
**Gambar 3.** flowchart perangkat keras

Diagram *flowchart* pada gambar 3 mulai dengan menyalakan perangkat *single board computer* (SBC) Raspberry Pi 3 dan menjalankan program python. Program python merupakan pemrograman inti proses kendali *input* diteruskan kontrol pintu garasi dan beberapa fitur lainnya. Sistem akan selalau menunggu kendali dari luar atau *smartphone*, ketika ada *input*-an sistem langsung memproses dan meneruskan perintah ke aktuator. Aktuator tersebut adalah pintu, kipas dan lampu sesuai dengan perintah yang dimasukan.

#### 4.1.4 Rancangan Perangkat Lunak

Perancangan perangkat lunak merupakan perancangan pembuatan aplikasi android pada *smartphone*. Aplikasi yang akan dibuat berfungsi sebagai antarmuka *user* untuk mengendalikan pintu garasi dan fitur lainnya. Berikut beberapa perancangan perangkat lunak yang akan dibuat dalam penelitian ini.

##### 4.1.4.1 Flowchart Perangkat Lunak

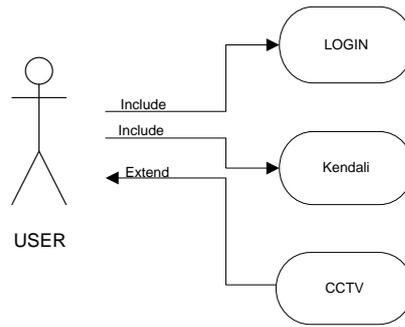


Gambar 4. *flowchart* perangkat lunak

Diagram *flowchart* pada gambar 6 menjelaskan alur kerja sistem kendali pintu garasi, dimulai dari membuka aplikasi sampai ditutupnya aplikasi atau selesai kendali. Aplikasi pertama kali dibuka akan menampilkan *form login*, setelah itu *user* akan memasukan data akun berupa *username* dan *password*. Kemudian sistem akan memeriksa apakah data tersebut cocok dengan data sistem. Ketika cocok maka diteruskan tampil *form kendali*, sedangkan ketika tidak cocok akan tampil pesan *error* “Login gagal”. Tampilan *form kendali* digunakan untuk antarmuka kontrol utama pengendalian pintu garasi dan fitur lainnya.

Sistem akan meneruskan perintah ke Raspberry Pi sesuai dengan perintah yang diinginkan. Perintah tersebut yaitu kontrol pintu, kontrol lampu, kontrol kipas dan pemantauan kamera keamanan. Kendali pintu, lampu dan kipas setiap aktifitas pengendaliannya akan disimpan pada database server untuk keperluan yang lain. Setelah selesai melakukan pengendalian *user* dapat keluar aplikasi menggunakan tombol *home* atau *back* pada *smartphone* pengguna.

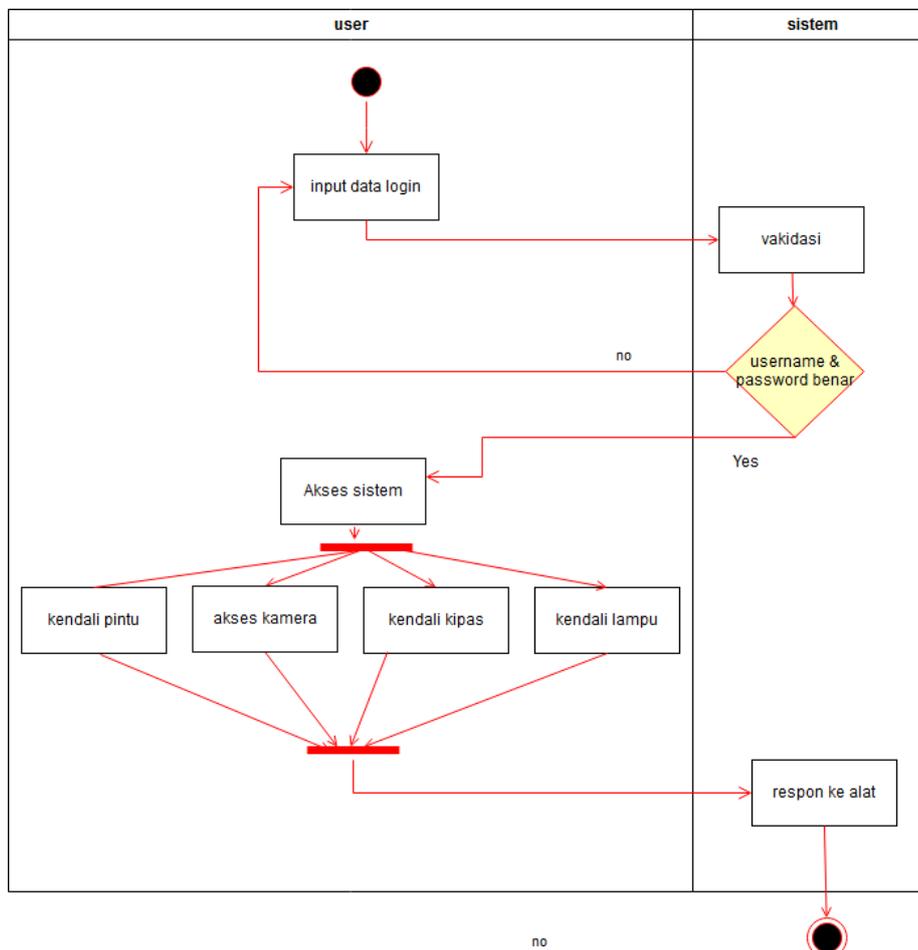
#### 4.1.4.2 Use Case Perangkat Lunak



**Gambar 5.** use case perangkat lunak

Use case merupakan diagram yang menjelaskan tentang penggunaan aplikasi terhadap penggunanya. Pada diagram yang ditunjukkan gambar 7 pengguna dapat melakukan *login* ke aplikasi dan melakukan pengendalian atau kontrol ke aplikasi. Fitur kamera memberikan informasi berupa gambar bergerak *real time* kepada *user*.

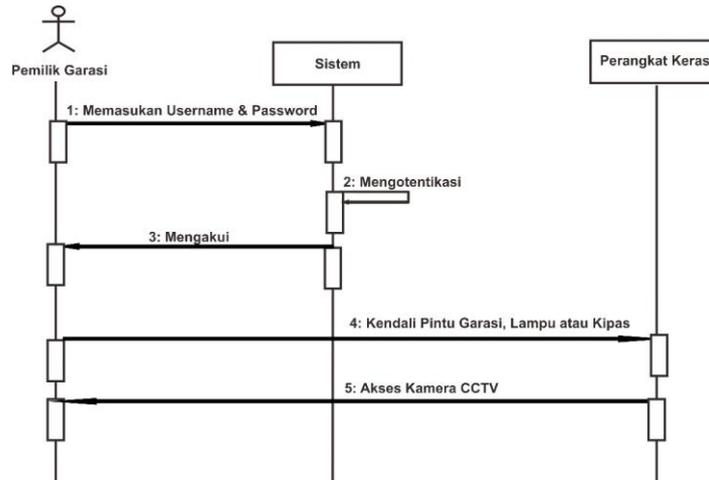
#### 4.1.4.3 Diagram Activity Perangkat Lunak



**Gambar 6.** diagram *activity* perangkat lunak

Diagram *activity* pada gambar 8 merupakan diagram aktifitas sistem aplikasi kendali garasi menggunakan *smartphone*. Diagram menjelaskan alur data yang akan masuk ke sistem basis data. Mulai dari data *login* untuk mevalidasi pengguna aplikasi yang dapat mengontrol kendali pintu garasi. Setelah aplikasi memfilter *user* yang dapat mengendalikan sistem maka akan diteruskan alir data kontrol ke sistem perangkat keras dan akan selalu disimpan di basis data setiap perintah yang dilakukan.

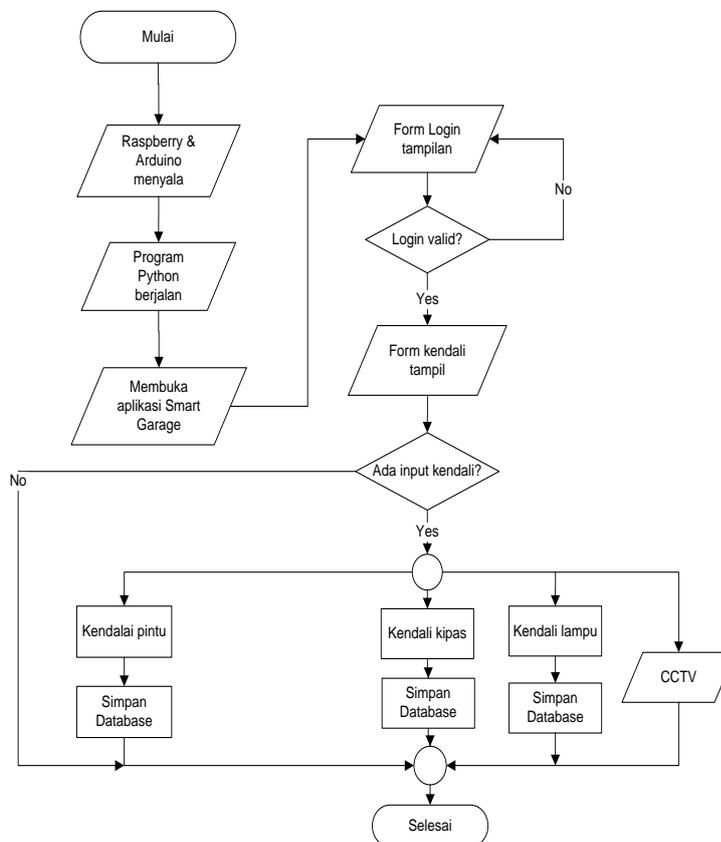
#### 4.1.4.4 Sequence Diagram Sistem Kendali Garasi



Gambar 7. *sequence* diagram sistem kendali garasi

Diagram *sequence* merupakan diagram akses keseluruhan sistem baik perangkat keras maupun perangkat lunak. Diagram pada gambar 9 menjelaskan bahwa *user* mengakses atau masuk ke sistem dengan memasukkan *username* dan *password*, kemudian data tersebut juga di kirimkan ke *hardware* untuk di lakukan pemeriksaan kecocokan data. *User* juga melakukan kontrol langsung dari sistem aplikasi dari *smartphone* yang langsung di teruskan ke perangkat lunak. Selain *user* melakukan kendali, *user* juga mendapat informasi gambar bergerak dari perangkat kamera yang ditampilkan pada aplikasi Smart Garage di *smartphone*.

#### 4.1.5 Flowchart Keseluruhan Sistem



Gambar 8. *flowchart* keseluruhan sistem

Diagram *flowchart* pada gambar 10 mulai dengan menyalakan perangkat *single board computer* (SBC) Raspberry Pi 3 dan menjalankan program python. Program python merupakan pemrograman inti proses kendali *input* diteruskan kontrol pintu garasi dan beberapa fitur lainnya. Selanjutnya pengguna membuka aplikasi smart garage, aplikasi smart garage berjalan dimulai menampilkan form login tampil user menginputkan data login.

Aplikasi Smart Garage selanjutnya akan memferifikasi data login tersebut, ketika login gagal aplikasi akan menampilkan pesan login gagal dan ketika login berhasil aplikasi menampilkan form kendali. Sistem akan selalau menunggu kendali dari *smartphone*, ketika ada *input*-an sistem langsung memproses dan meneruskan perintah ke aktuator. Aktuator tersebut adalah pintu, kipas dan lampu sesuai dengan perintah yang dimasukan. Kendali pintu, lampu dan kipas setiap aktifitas pengendaliannya akan disimpan pada database server untuk keperluan yang lain. Setelah selesai melakukan pengendalian *user* dapat keluar aplikasi menggunakan tombol *home* atau *back* pada *smartphone* pengguna.

## 5. Implementasi

### 5.1 Implementasi *Database*

Database atau lebih dikenal dengan istilah basis data pada penelitian ini digunakan untuk ruang penyimpanan data pengguna atau pemilik garasi dan data log kendali. Database yang di implementasikan dengan menggunakan DBMS MySQL yang sudah sangat populer. Berikut tabel yang dibutuhkan dalam sistem kendali pintu garasi berbasis mobile.

#### 5.1.1 Tabel Log

Tabel ini digunakan untuk menyimpan data setiap kendali sistem garasi.

Tabel 1. Tabel Log

Column Name	Data Type	Length
kd_log	int	5
id_user	int	5
tanggal	date	
waktu	time	
perintah	varchar	20

#### 5.1.2 Tabel *User*

Tabel ini digunakan untuk menyimpan data *user* atau pemilik sistem garasi.

Tabel 2. Tabel *User*

Column Name	Data Type	Length
id user	int	5
username	varchar	50
password	varchar	50

### 5.2 Implmentasi Perangkat Keras

Implementasi perangkat keras dalam penelitian ini dibuat dengan sistem prototipe atau miniatur garasi kendaraan. Prototipe dibuat menggunakan bahan multyplex dibalut dengan plastik stiker untuk mempercantik tampilan garasi. Prototipe dibuat sedemikian rupa sehingga mirip dengan implementasi garasi seperti posisi pintu garasi, peletakan kamera CCTV, kunci pintu, mesin aktuator penggerak pintu, lampu hingga kipas sirkulasi udara.

#### 5.2.1 Prototipe garasi

Prototipe garasi dengan dipasang langsung dengan mesin utama sistem yang dibuat serapi-rapinya. Peletakan mesin kendali dibawah prototipe sehingga tidak mengganggu kerapian.



**Gambar 9.** prototipe garasi

### 5.2.2 Prototipe pintu garasi

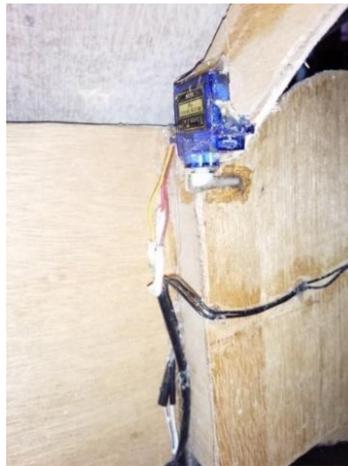
Prototipe pintu garasi menggunakan dua buah daun pintu dengan arah membuka keluar, masing-masing menggunakan mesin penggerak pintu dan satu kunci berada di tengah.



**Gambar 10.** prototipe pintu garasi

### 5.2.3 Prototipe penggerak pintu garasi

Prototipe penggerak pintu garasi disimulasikan menggunakan dua buah servo yang diletakan pada masing-masing daun pintu.



**Gambar 11.** prototipe penggerak pintu garasi

### 5.2.4 Prototipe kunci pintu garasi

Prototipe kunci penggerak pintu garasi disimulasikan menggunakan dua buah selenoid *door lock* yang diletakan pada salah satu pintu daun pintu yang nantinya dapat mengunci ketika kedua pintu garasi tertutup.



**Gambar 12.** prototipe kunci pintu garasi

### 5.2.5 Prototipe kipas garasi

Prototipe kipas garasi diletakan pada belakang bagian atas dinding garasi, berfungsi sebagai sirkulasi pembuangan udara di dalam garasi agar suhu ruangan di dalam garasi tidak terlalu lembab atau terlalu panas.



**Gambar 13.** prototipe kipas sirkulasi udara garasi

### 5.2.6 Prototipe lampu garasi

Prototipe lampu menggunakan jenis lampu LED dengan konsumsi daya 12V DC, diletakan di dalam bagian atas atap garasi. berfungsi sebagai penerangan di dalam garasi.



**Gambar 14.** prototipe lampu garasi

### 5.2.7 Prototipe CCTV

Prototipe CCTV menggunakan kamera jenis webcam diletakan di luar bagian atas pintu garasi bagian. Berfungsi sebagai kamera keamanan untuk memantau keadaan luar garasi.



Gambar 15. prototipe CCTV

### 5.2.8 Implementasi single board computer (SBC)

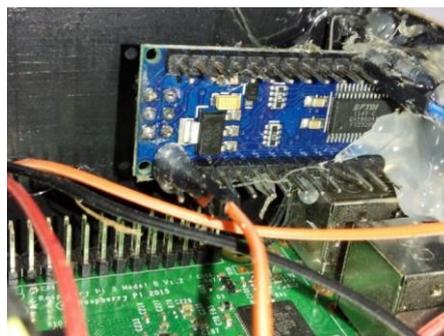
Implementasi *single board computer* (SBC) menggunakan Raspberry Pi 3 model B diletakan di bagian bawah prototipe garasi dan berada di dalam kotak. *Single board computer* (SBC) Raspberry Pi berfungsi sebagai sistem kendali utama garasi.



Gambar 16. Implementasi *single board computer* (SBC) raspberry pi

### 5.2.9 Implementasi mikrokontroler

Implementasi mikrokontroler menggunakan Arduino NANO diletakan di bagian bawah prototipe garasi dan berada di dalam kotak jadi satu dengan *single board computer* (SBC) Raspberry Pi.

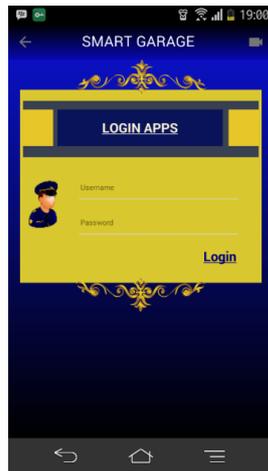


Gambar 17. implementasi mikrokontroler

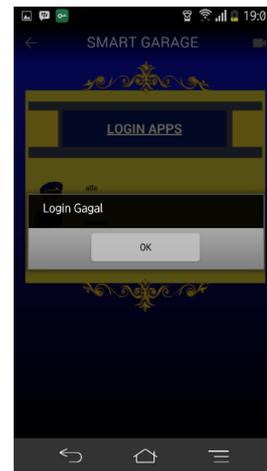
## 5.3 Implementasi Perangkat Lunak

### 5.3.1 Tampilan form login

*Form login* pada tampilan *screen* terdapat *input-an* *Username* dan *Password*, di mana data *username* dan *password* sudah tersimpan dalam basis data. Untuk login tinggal ketikkan *username* dan *password* kemudian tekan tombol *login*.



Gambar 18. tampilan sistem login



Gambar 19. tampilan login gagal

### 5.3.2 Tampilan form kendali

Ketika login berhasil maka tampilan screen di *smartphone* terdapat beberapa menu pilihan untuk kendali pintu, lapu, kipas, dan kamera pemantau.



Gambar 20. menu kendali aplikasi android

### 5.3.3 Tombol kendali dua pintu

Menu tombol kendali dua pintu ini berfungsi untuk membuka dan menutup kedua pintu garasi secara bersamaan.



Gambar 21. tombol kendali dua pintu aplikasi android

### 5.3.4 Tombol kendali pintu kanan dan kiri

Tombol kendali pintu kanan berfungsi untuk membuka atau menutup pintu kanan garasi dan tombol kendali pintu kiri berfungsi untuk membuka atau menutup pintu kanan garasi



Gambar 22. tombol kendali pintu kanan



Gambar 23. tombol kendali pintu kiri

### 5.3.5 Tombol kendali lampu garasi

Selanjutnya menu tombol kendali lampu garasi yang berfungsi untuk menghidupkan atau mematikan lampu di dalam garasi.



Gambar 24. tombol kendali lampu garasi aplikasi android

### 5.3.6 Tombol kendali kipas garasi

Berikutnya menu tombol kendali kipas garasi yang berfungsi untuk menghidupkan atau mematikan kipas sirkulasi udara di dalam garasi.



Gambar 25. tombol kendali kipas garasi aplikasi android

### 5.3.7 Tombol kendali kamera keamanan garasi

Pada menu tombol yang terakhir yaitu kendali kamera keamanan garasi yang terletak di pojok kanan atas, berfungsi untuk memantau keamanan di luar garasi secara *real time*.



Gambar 26. tampilan kamera keamanan aplikasi android

## 6. Kesimpulan

Berdasarkan dari pembahasan dan hasil pengujian sistem yang dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Pembuatan prototipe akses kontrol pintu garasi otomatis menggunakan Raspberry Pi dapat dikendalikan menggunakan aplikasi android dengan nama Smart Garage melalui media *wireless*.
2. Optimalisasi kinerja Raspberry Pi sebagai pusat kontrol pemrosesan sistem selain buka tutup pintu garasi prototipe garasi otomatis ini juga dilengkapi dengan lampu, kipas dan webcam, yang semua itu dikendalikan melalui aplikasi android Smart Garage.
3. Penambahan mikrokontroler Arduino NANO dalam penelitian ini berfungsi sebagai kontrol sekunder pintu garasi, membantu Raspberry Pi menggerakkan motor servo dengan cara komunikasi serial.
4. Keamanan akses masuk ke menu kendali aplikasi Smart Garage menggunakan *username* dan *password*, yang disediakan dan tersimpan di *database* komputer *server* akses kontrol pintu garasi otomatis. Pengguna yang mempunyai aplikasi Smart Garage tetapi tidak mempunyai *username* dan *password* maka tidak dapat mengakses kendali garasi.
5. Jangkauan jaringan *wireless* untuk komunikasi antara *smartphone* dengan sistem akses kontrol garasi bisa mencapai jarak 50 meter dengan keadaan di ruang terbuka bebas tanpa terhalang benda apapun.

## Daftar Pustaka

- Anggraini, A., (2016), *Rancang Bangun Open/Close Pintu Ruangan Menggunakan Voice Reconition Berbasis Raspberry Pi*, Tugas Akhir, Teknik Komputer, Politeknik Negeri Sriwijaya, Palembang.
- Haryati, (2015), *Sistem Kontrol Gerbang Garasi berbasis Web Menggunakan Raspberry Pi*. S1 thesis, Teknik Elektro, Universitas Mercu Buana, Jakarta.
- Wibowo, S.H., (2014), *Simulasi Pengontrolan Pintu Garasi Otomatis*, Jurnal INTEKNA, Vol 14(2), 102-209.
- Shadiq, H.M., Sudjadi, dan Darjat., (2014), *Perancangan Kamera Pemantau Nirkabel Menggunakan Raspberry Pi Model B*. Jurnal Elektronik, Vol 3(4), 546-551.
- Yuwono, B., Nugroho, S.N., dan Heriyanto., (2015), *Pengembangan Model Public Monitoring System Menggunakan Raspberry Pi*, Jurnal Telematika. Vol 12(2), 123 – 133.
- Baskoro, I.T., Darjat, D., dan Sudjadi, S., (2015). *Perancangan Pengontrolan Nyala Lampu Dan Kipas Angin Pada Sebuah Ruangan Menggunakan Raspberry PI Model B Dengan WEB GUI*. Jurnal TRANSIENT, vol. 3(4), 567–571.
- Sela, E.I., (2008). *Rekayasa Perangkat Lunak Untuk Aplikasi Multimedia*, (<https://scholar.google.co.id/citations?user=xspSq4MAAAAJ&hl=en>) akses 7 Maret 2018
- Pressman, R. S., (2012). *Software Engineering: A Practitioner's Approach*, Seventh. New York: McGraww-Hill.
- Kadir, A., (2015), *From Zero to A Pro Arduino*, Yogyakarta: Penerbit Andi
- Monk, S., (2015), *Raspberry Cookbook Software and Hardware Problems and Solution*, Sebastopol: O'Reilly Media, Inc.
- Matthes, E., (2016), *Python Crash Course*, San Francisco: No Starch Press.
- Kadir, A., (2017), *Dasar Raspberry Pi*, Yogyakarta: Penerbit Andi.
- Dinata, A., (2017), *Physical Computing dengan Raspberry Pi*, Jakarta: Penerbit PT Elex Media Komputindo.