**BAB III**

**METODOLOGI PENELITIAN**

**3.1 Tahap Penelitian**

Dalam kegiatan penelitian, dibuat langkah-langkah melaksanakan penelitian sebagai berikut:

1. Studi Literatur

Studi literatur dilakukan dengan mengambil dan mempelajari materi dari buku, jurnal, *paper* maupun skripsi yang telah ada sebelumnya untuk dijadikan sebagai acuan, referensi atau pengembangan dalam melakukan penelitian yang akan dilaksanakan.

1. Perancangan Mekanik

Perancangan mekanik alat dilakukan setelah mempelajari materi dan komponen-komponen yang dibutuhkan untuk proyek yang akan dibuat.

1. Perancangan Rangkaian dan Program

Membuat rangkaian serta program pada proyek akhir yang akan dibuat.

1. Pembuatan Prototype

Pembuatan prototype ini dilakukan pada PCB dalam keadaan terbuka dan tidak memasang beban penggerak gerbang otomatis.

1. Pembuatan Alat

Merupakan kelanjutan pembuatan prototype, alat diberi kotak pelindung agar terlihat rapi, serta dipasang beban berupa penggerak gerbang otomatis sehingga alat siap digunakan.

1. Pengujian Alat dan Pengambilan Data

Pengujian alat akan dilakukan supaya benar dan sesuai dengan kebutuhan proyek. Setelah sesuai kemudian dapat diambil data.

1. Pengolahan Data

Data yang telah didapat kemudian diolah sebagai hasil untuk bahan penyusunan laporan.

1. Penyusunan Laporan

Penyusunan laporan dilakukan setelah semua tahapan selesai dilaksanakan sesuai dengan prosedur.

25

**3.2 Waktu dan Tempat Kegiatan**

Perancangan dan penelitian proyek akhir ini dilaksanakan pada:

Waktu : Juni 2019 - Agustus 2019

Tempat: Jl.Cipaku 1 No.18, Kelurahan Ledeng, Kecamatan Cidadap, Kota Bandung.

**3.3 Ruang Lingkup Kegiatan**

Untuk menghindari meluasnya topik permasalahan, maka perlu adanya batasan masalah sebagai berikut:

1. Mikrokontrol yang digunakan untuk memproses *output* adalah NodeMCU V3 ESP8266 untuk komunikasi *wireless.*
2. Komunikasi dari mikrokontroler ke *smartphone* dilakukan menggunakan jaringan *Wi-Fi.*
3. Pemograman menggunakan *software* Arduino IDE.

**3.4 Alat dan Bahan**

Komponen yang digunakan pada proyek akhir ini terdiri dari beberapa bagian antara lain:

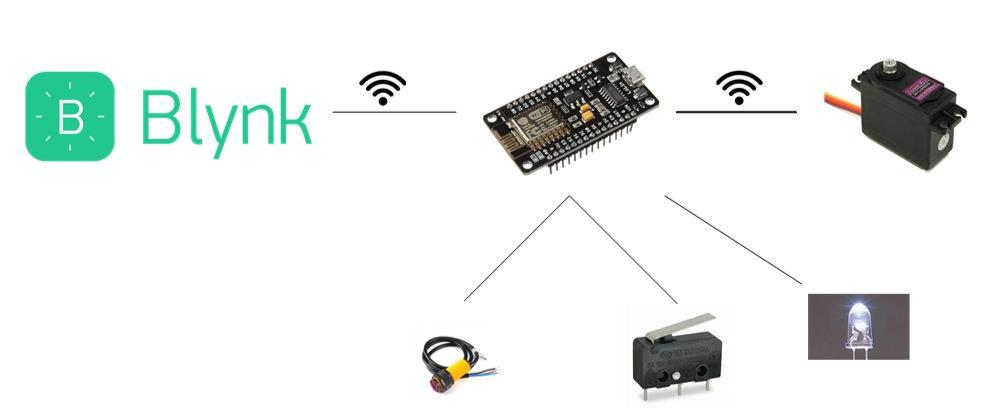
1. *Hardware:*
   1. NodeMCU V3
   2. *Proximity sensor*
   3. PCB dot matrix lubang
   4. Motor servo MG966R
   5. *Pin header male*
   6. Kabel
   7. *Limit switch*
   8. *Power supply*
2. *Software:*
   1. Arduino IDE
   2. Blynk
3. Alat dan bahan:
   1. Laptop
   2. *Smartphone*
   3. Box elektronik

26

1. Bor
2. Palu
3. Paku
4. Solder
5. Gergaji
6. Duplex
7. Kayu
8. Akrilik
9. Kabel USB

**3.5** **Perancangan Alat**

**3.5.1 Perancangan Sistem**

****

**Gambar 3.1** Perancangan sistem.

Gambar 3.1 ini merupakan perancangan dari suatu rangkaian sistem secara keseluruhan dengan NodeMCU sebagai mikrokontrol. NodeMCU dapat terhubung secara *wireless* yang terintegrasi dengan Wi-Fi ESP8266. Rancangan ini menggunakan sensor jarak (*proximity sensor)* sebagai pendeteksi pada saat gerbang akan terbuka dan tertutup. Jika sensor mendeteksi adanya objek pada saat gerbang akan terbuka maka gerbang tersebut akan terhenti dan akan melanjutkan kembali membuka setelah waktu yang sudah ditentukan, dan pada saat gerbang akan tertutup kemudian sensor mendeteksi adanya objek maka gerbang akan kembali terbuka. Data yang didapat dari sensor akan dikirim melalui internet ke Blynk *cloud.* Data yang dikirim dapat dimonitor pada aplikasi Blynk yang telah terpasangpada *smartphone*. Pengiriman data ke Blynk akan dilakukan terus menerus selama

27

terkoneksi pada jaringan internet. Gerbang ini akan terbuka dan tertutup menggunakan motor servo MG996R dan akan terhenti ketika gerbang menekan *limit switch* yang akan termonitor pada aplikasi Blynk*.* Sedangkan untuk lampuLED tersebut digunakan sebagai indikator monitoring gerbang ketika gerbang tertutup dan terbuka.

**3.5.2 Perancangan Mekanik**

Perancangan ini merupakan desain mekanik sistem gerbang otomatis, desain mekanik tersebut merupakan gabungan dari beberapa komponen maupun alat dan bahan. *Input* sensor *proximity*, motor servo dan *limit switch* terhubung dengan mikrokontroler NodeMCU. Sensor tersebut sebagai pendeteksi yang disimpan didepan gerbang agar dapat mendeteksi adanya objek yang akan masuk maupun mengganggu proses beroperasinya gerbang. Motor servo sebagai penggerak maju dan mundurnya gerbang kemudian *limit switch* yang dipasang didepan dan dibelakang maju maupun mundurnya gerbang akan menghentikan putaran pada motor servo tersebut. Gambar 3.2 dibawah ini merupakan perancangan mekanik yang telah selesai dibuat.



**Gambar 3.2** Perancangan mekanik.

**3.5.3 Perancangan Box**

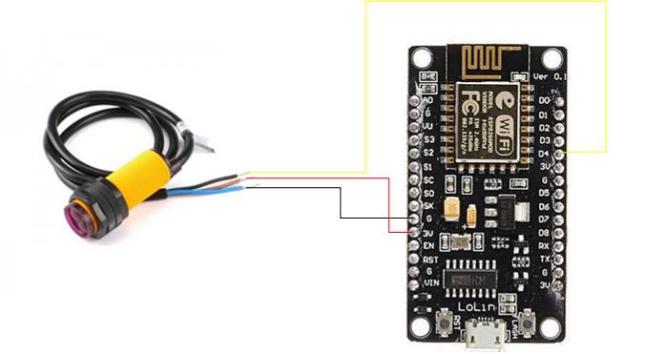
****

**Gambar 3.3** Perancangan box.

28

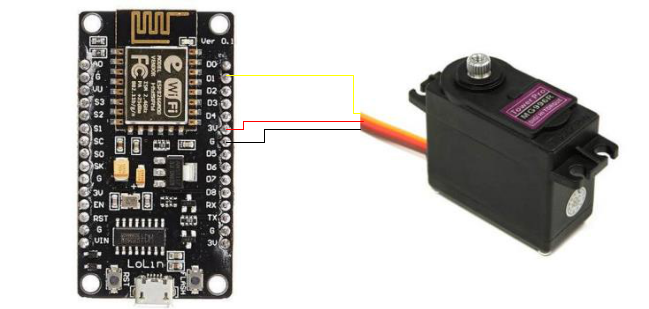
Gambar 3.3 merupakan tampilan dari perancangan box dengan menggunakan box elektronik yang akan di gunakan untuk menyimpan rangkaian elektronik pada proyek akhir ini, penggunakan box ini bertujuan agar rangkaian elektronik terlindungi dan terlihat rapi. Material yang digunakan oleh box ini adalah plastik abs, box ini berukuran 140 x 90 x 50mm, rangkaian akan dimasukan kedalam box kemudian box akan dilubangi untuk keluarnya masuk kabel pada komponen-komponennya.

**3.5.4 Perancangan Elektronika**

****

Gambar 3.4 Rangkaian sensor.

Gambar 3.4 merupakan rangkaian sensor *proximity* dengan mikrokontroler NodeMCU. Kabel sensor warna merah dihubungkan dengan pin sumber 3V pada mikrokontroler kemudian kabel warna hitam pada sensor dihubungkan ke pin digital D4 pada mikrokontroler dan kaber warna biru pada sensor adalah kabel *ground* dihubungkan ke *ground* pada mikrokontroler.



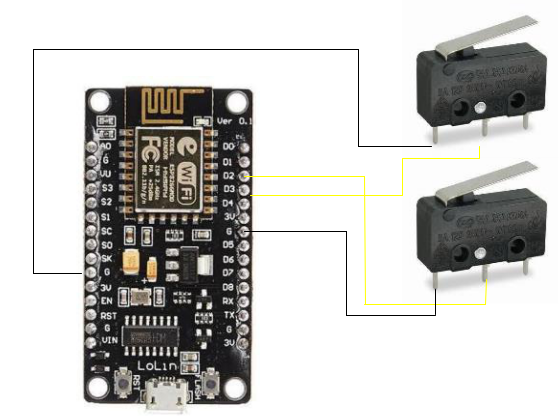
**Gambar 3.5** Rangkaian motor servo MG966R.

Gambar 3.5 merupakan rangkaian motor servo MG966R yang terkoneksi ke

NodeMCU. Kabel pin pada motor servo dihubungkan pada pin digital D1 di

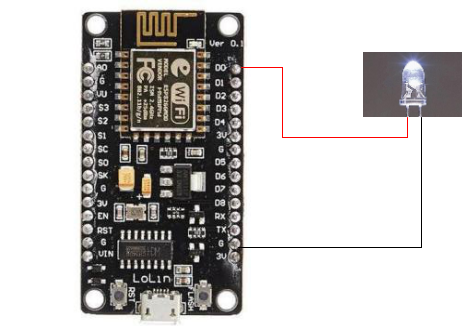
NodeMCU, kemudian kabel yang lainnya dihubungkan pada *power supply.*

29



**Gambar 3.6** Rangkaian*limit switch*.

Gambar 3.6 merupakan rangkaian *limit switch.* Pada proyek ini *limit siwtch* berfungsi untuk menghentikan motor yang bergerak memutarkan gerbang dan akan berhenti ketika gerbang menyentuh/menekan *limit switch*. Proyek ini menggunakan 2 buah *limit switch* yang terkoneksi pada NodeMCU, dengan menghubungkan kabel pin ke D2 dan D3 kemudian kabel *ground* dihubungkan pada *ground* yang ada di NodeMCU.

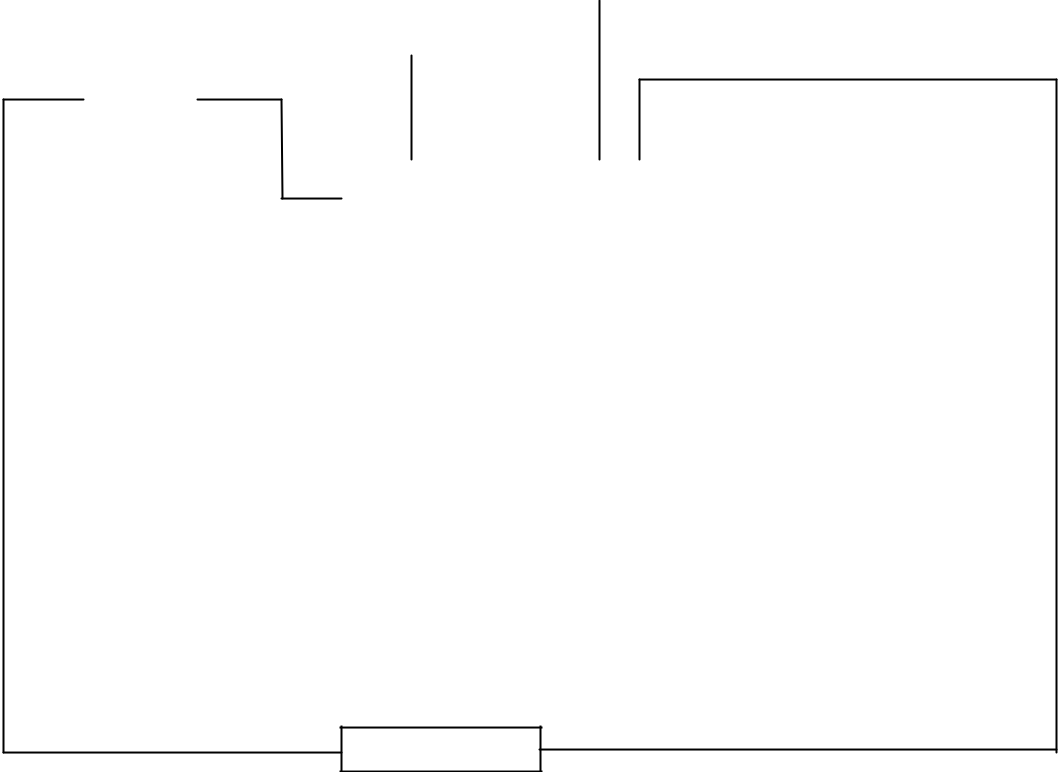


**Gambar 3.7** Rangkaian lampu LED.

Gambar 3.7 merupakan rangkaian lampu LED yang bekerja sebagai indikator monitoring, kabel positif dihubungkan pada pin D0 dan kabel negatif dihubungkan pada *ground* yang ada di NodeMCU tersebut.

30

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | Power Supply | |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
| Proximity |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Sensor |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  | D4 | |  |  | D1 | |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  | Servo | |  | Gerbang |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
| Limit | D3 | |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Switch 1 |  |  |  | NodeMCU |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  | |  |  |  |  |  |  |  |  |  |
| Limit | D2 | |  |  | D0 | |  |  |  |  |  |
|  |  | LED | |  |  |  |
| Switch 2 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

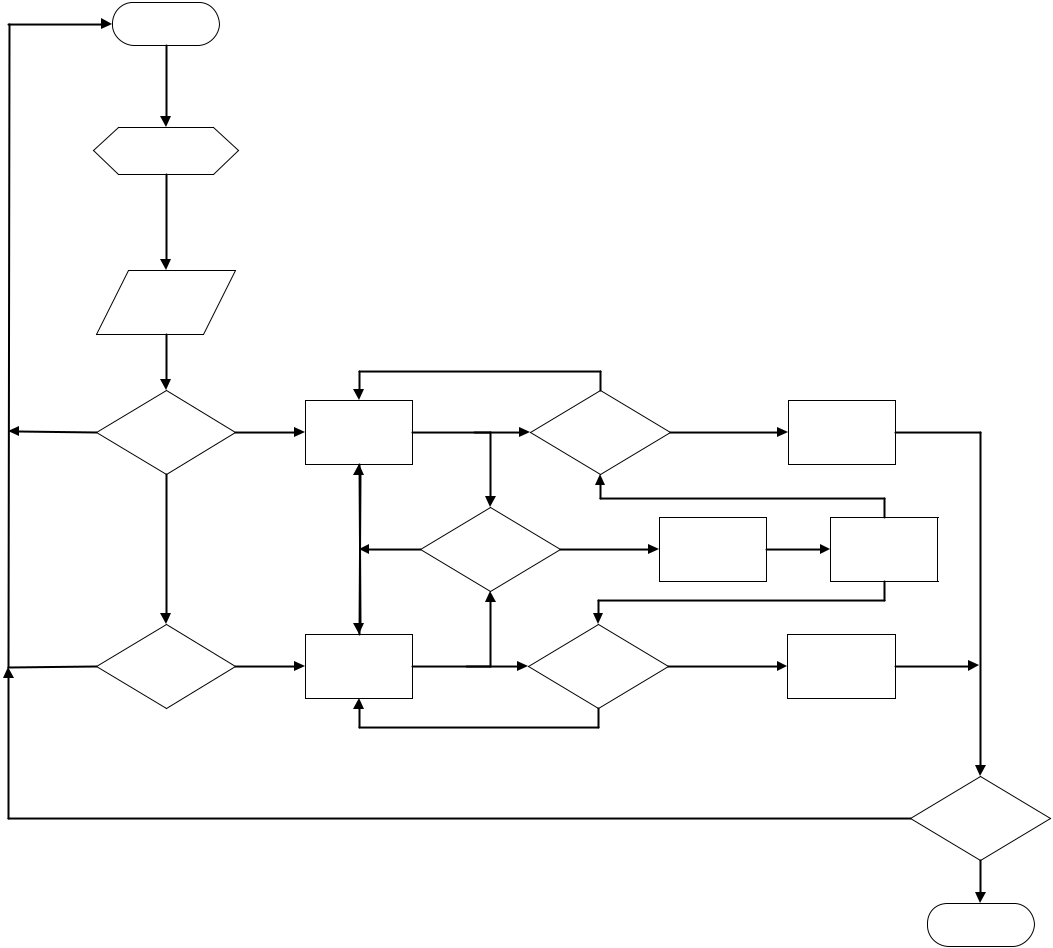


Smartphone

**Gambar 3.8** Diagram blok sistem.

Gambar 3.8 merupakan diagram blok alat dengan *power supply* sebagai sumber daya untuk mikrokontroler dan motor servo. Motor servo digunakan sebagai penggerak maju dan mundurnya gerbang, sensor *proximity* berfungsi sebagai pendeteksi objek pada saat gerbang beroperasi, *limit switch* disimpan ditiap ujung gerbang untuk menghentikan gerbang ketika *limit switch* tertekan, kemudian lampu LED berfungsi sebagai indikator monitoring terbuka dan tertutupnya gerbang. Dengan ESP8266 memungkinkan NodeMCU terhubung dengan AP *(access point)* yang terkoneksi jaringan internet. Dalam jaringan internet data olehsensor dapat dikirim ke *cloud* sebagai server. Pada *smartphone* telah terpasang aplikasi yang digunakan sebagai kontrol dan monitoring gerbang.

31



Start

Inisialisasi

Blynk

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Input |  |  |  |  |  |
|  | data |  |  |  |  |  |
|  |  |  | Tidak |  |  |  |
| Tidak | Ya | Gerbang |  |  | Ya |  |
|  |  |  |  |
|  | Tombol on |  | LS1 ditekan? |  | Stop |
|  | terbuka |  |  |
|  |  |  |  |  |  |
|  | Tidak | Tidak | IR mendeteksi | Ya | Stop | Terbuka/ |
|  |  |  | tertutup |
|  |  |  |  |  |  |
| Tidak | Ya | Gerbang |  |  | Ya |  |
|  |  | LS2 ditekan? | Stop |
|  | Tombol off | tertutup |  |  |
|  |  |  |  |  |  |
|  |  |  | Tidak |  |  |  |

EOF?

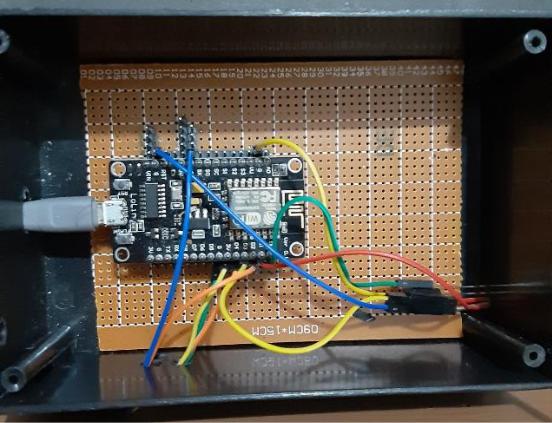
End

**Gambar 3.9** Diagram alir.

Gambar 3.9 merupakan diagram alir alat dimulai dengan inialisasi pada aplikasi Blynk yang ada di *smartphone* ke mikrokontroler untuk dijadikan perintah bagi pengguna yang akan menutup dan membuka gerbang, mikrokontroler akan membaca keadaan gerbang terbuka atau tetutup kemudian mikrokontroler akan mengirim kondisi gerbang ke aplikasi *Blynk* yang ada di *smartphone,* jika kondisi yang diterima pada tombol *on* maka gerbang akan terbuka dan jika tombol *off* maka gerbang akan tertutup kemudian akan terhenti ketika gerbang menekan *limit switch*. Pada saat bagian sensor *proximity* mendeteksi adanya objek ketika gerbang akan terbuka maka gerbang akan terhenti dan ketika sensor mendeteksi pada saat gerbang akan menutup lalu gerbang akan otomatis kembali pada posisi terbuka.

32

**3.5.5 Perancangan Prototype**

****

**Gambar 3.10** Perancangan prototype.

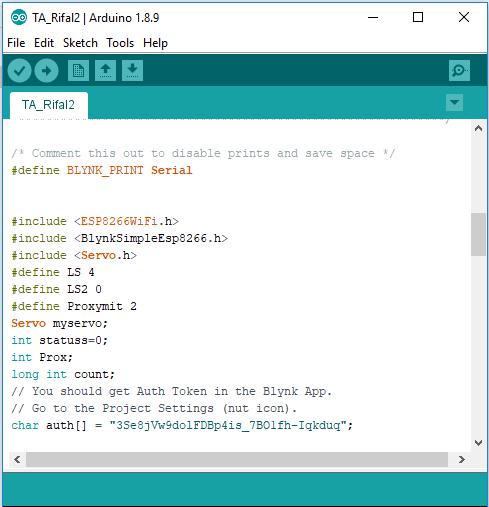
Gambar 3.10 merupakan tampilan dari perancangan prototype, prototype alat ini adalah perancangan alat sementara atau prototype yang bertujuan untuk memastikan semua komponen dan program yang telah dibuat berjalan dengan baik, perancangan dibuat pada papan PCB dan tidak menggunakan beban gerbang otomatis. Pada proses ini semua komponen dan program berjalan dengan baik.

**3.5.6 Perancangan Software**

1. Arduino IDE

Pemrograman NodeMCU dibuat untuk mengolah data-data komponen yang digunakan dan dikirimkan ke server Blynk melalui NodeMCU. Pemrograman dilakukan menggunakan format Blynk. Data yang dikirim ke Blynk *cloud* dapat dimonitor melalui *smartphone* menggunakan aplikasi Blynk. Pertama pada Arduino IDE dilakukan dengan membuat inisialisasi program data komponen yang dipakai.

33



**Gambar 3.11** Program data komponen.

Gambar 3.11 merupakan program data komponen, setelah melakukan pemrograman pada komponen-komponen yang dipakai, program tersebut digabungkan dan perlu dipastikan bahwa sensor bekerja sesuai dengan program. Hal terpenting yang perlu diperhatikan saat melakukan pemrograman adalah melakukan secara bertahap, supaya saat terjadi *error* bisa dapat segera ditemukan dan diselesaikan. Setelah program padakomponen yang dipakai berjalan dengan baik. Lakukan penambahan program untuk memprogram sensor *proximity* saat mendeteksi objek agar gerbang dapat berhenti ketika akan terbuka dan kembali mundur saat akan tertutup secara otomatis dengan fungsi IF-ELSE *statement.*

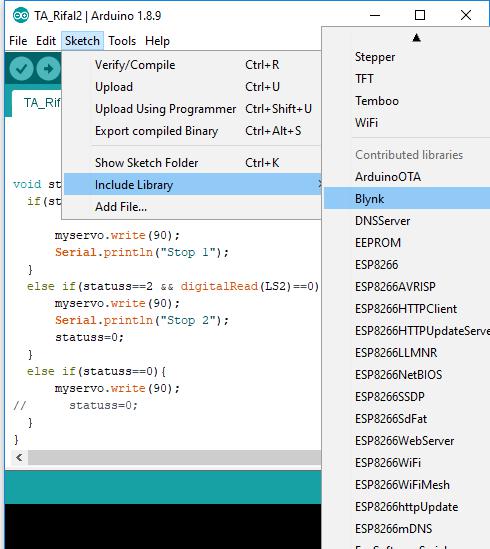
34



**Gambar 3.12** Program fungsi IF.

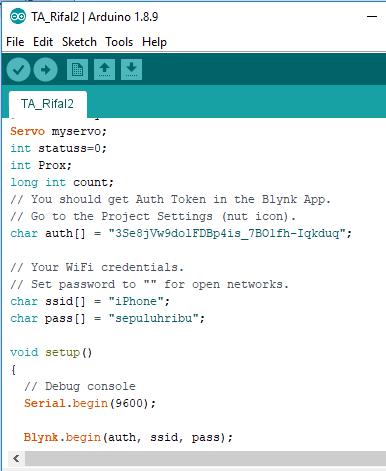
Setelah selesai membuat pemrograman untuk mengontrol dan monitoring gerbang otomatis, pastikan respon pada sensor dengan sesuai program. Selanjutnya menambahkan program untuk akses ke aplikasi Blynk*,* pastikan *Library Blynk* telah terpasang pada *software* Arduino IDE. Seperti yang ada pada gambar 3.12 diatas tersebut.

35



**Gambar 3.13** *Library Blynk.*

Setelah *library* terpasang kemudian cari di *example* pada arduino IDE, cari *Blynk* kemudian *board Wi-Fi* dan pilih ESP8266, gabungkan programdengan program sensor *proximity* yang telah dibuat. Pada program tersebut masukan kode *auth, ssid* (nama *access point*) dan *password*. *Auth token* didapatkan setelah memasang aplikasi Blynk pada *smartphone*. *Compile* program dan pastikan tidak terjadi *error.*

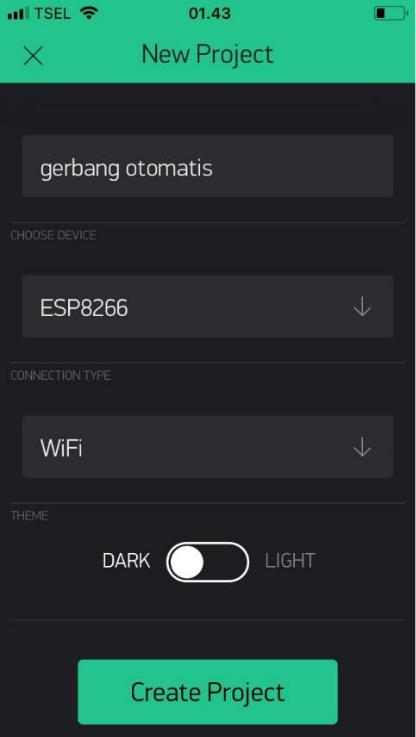
**

**Gambar 3.14** Program pengkoneksian mikrokontroler.

36

1. Blynk

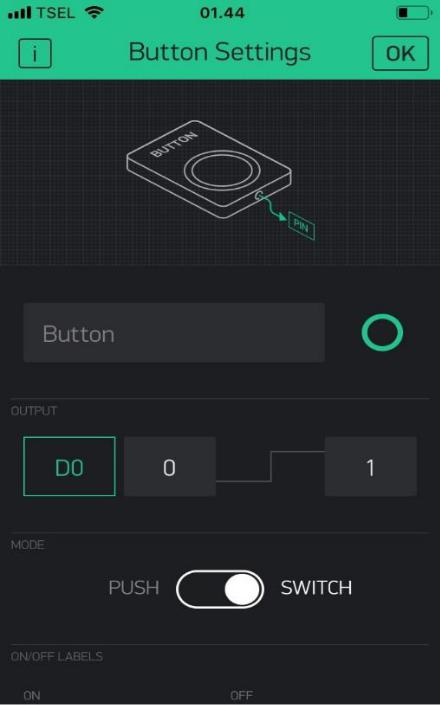
Aplikasi Blynk digunakan untuk membuka dan menutup gerbang serta memonitoring tertutup atau terbuka. Antarmuka pada aplikasi Blynk dapat diatur dengan menambahkan widget seperti grafik dan *button.*

**

**Gambar 3.15** Membuat project baru Blynk*.*

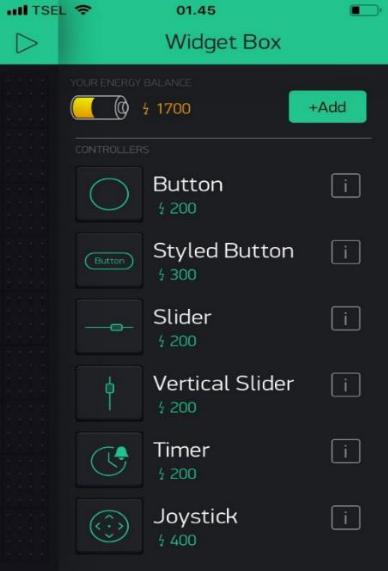
Gambar 3.15 merupakan tampilan menu membuat proyek baru Blynk di *smartphone* seperti pada gambar 3.15. Pertama membuat nama proyek lalumemilih *device* yang dipakai. Kemudian memilih tipe koneksi yang tersedia seperti *Wi-Fi, Ethernet,* GSM dan lain sebagainya. Setelah memilih tipe koneksi, langsung *create* atau memilih tema yang tersedia. Masuk pada tampilan awal proyek yang telah dibuat terdapat beberapa menu yang disediakan seperti widget box, play button dan setting. Pada menu widget *box* disediakan beberapa widget yang dapat digunakan sebagai monitoring,kontrol, notifikasi dan lain sebagainya. Widget yang digunakan dapat langsung didrag dan drop pada media yang disediakan. Widget dikonfigurasi supaya dapat menerima data dari alat yang telah dibuat. Konfigurasi widget cukup mudah dengan menyamakan pin yang diprogram pada alat yang dibuat.

37



**Gambar 3.16** Mengatur widget.

[Pada menu widget box terdapat banyak widget yang dapat langsung](http://repository.unej.ac.id/) [digunakan dengan *drag* dan *drop* pada proyek yang telah dibuat. Supaya](http://repository.unej.ac.id/) [dapat terhubung dengan *device* atau alat yang dibuat, diperlukan *auth*](http://repository.unej.ac.id/) [*token* dari aplikasi yang telah terkirim di *email*. *Auth token* yang telah](http://repository.unej.ac.id/)[diperoleh dimasukkan pada NodeMCU supaya dapat terkoneksi.](http://repository.unej.ac.id/)



**Gambar 3.17** Macam-macam widget Blynk