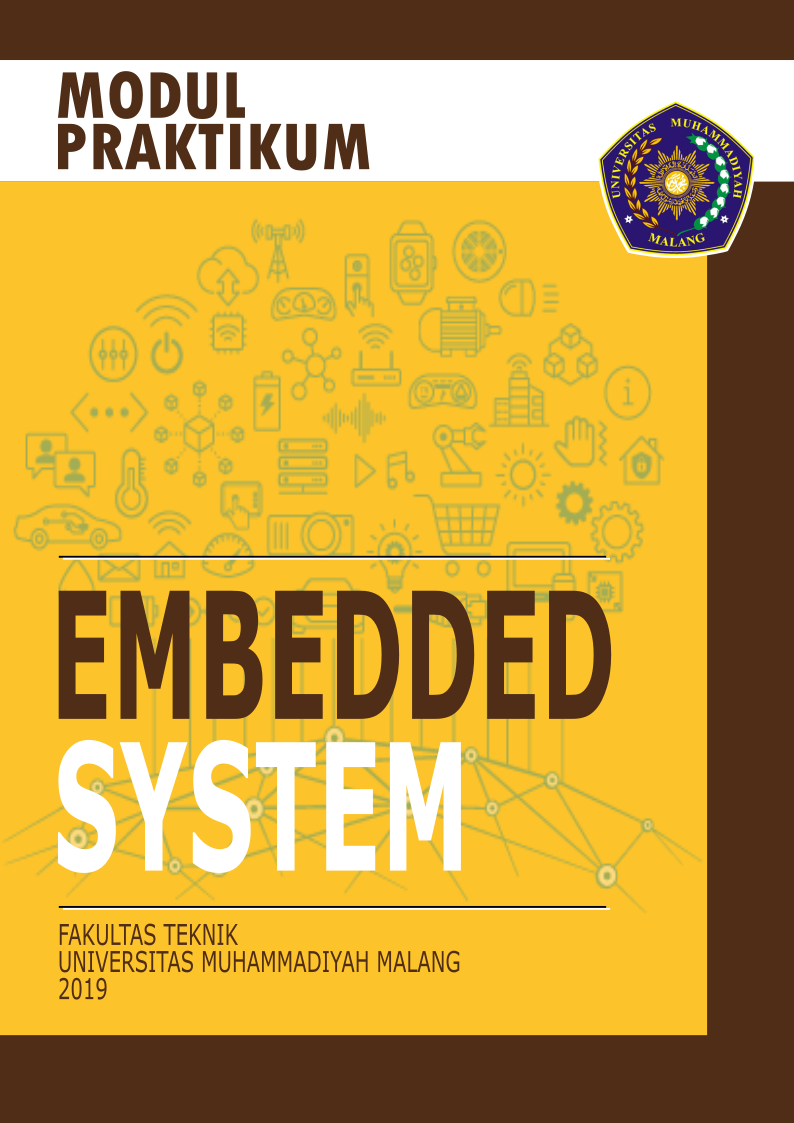
**COVER**

NAMA PRAKTIKAN

NIM

*Halaman ini sengaja dikosongkan*

**STANDAR OPERASIONAL PROSEDUR**

**LABORATORIUM TEKNIK ELEKTRO**

**UNIVERSITAS MUHAMMADIYAH MALANG**

1. **PRA PRAKTIKUM**
2. Ka. Laboratorium bersama Ketua Prodi menetapkan daftar Mata Praktikum yang akan dilaksanakan pada semester berjalan
3. Laboran atau Staf mengumumkan daftar Mata Praktikum dan pengumuman lainnya via web lab-elektro.umm.ac.id
4. Staf / Laboran menerima pendaftaran calon praktikan yang mengulang
5. Staf / Laboran mengumumkan daftar peserta Mata Praktikum berdasarkan data peserta mata kuliah dan peserta mengulang di web lab-elektro.umm.ac.id
6. Kepala lab dan wakil kepala lab menetapkan daftar Instruktur dan Asisten Mata Praktikum dan diusulkan untuk ditetapkan SK Dekan
7. Ka. Lab mengundang Peserta Mata Praktikum untuk mengikuti pertemuan persiapan dan pembagian jadwal peserta mengikuti praktikum dan peraturan serta prosedur praktikum dan K3
8. Instruktur dan Asisten mengundang peserta Mata Praktikum untuk mengikuti Ujian Pra Praktikum (Memberikan Tugas Pra Praktikum)

1. **PRA PELAKSANAAN PERCOBAAN PRAKTIKUM**
   * + 1. Asisten dan Praktikan hadir 15 menit sebelum dimulai jam praktikum
       2. Asisten mempersiapkan instrumen ukur serta modul praktikum dan peralatan pendukung seperti kabel, jumper dan lain lain
       3. Praktikan membaca petunjuk praktikum dan mempersiapkan kebutuhan peralatan sebelum masuk ruang/lab
       4. Asisten memberikan salam dan ucapan selamat datang dengan senyum serta memberikan arahan kepada kelompok Praktikan tentang prosedur pelaksanaan praktikum dan penjelasan daftar peralatan dan modul
       5. Asisten menunjuk peserta yang menjadi petugas pencatat, melakukan pengukuran dan pembantu pelaksanaan
       6. Asisten meminta kelompok Praktikum untuk membaca doa/Basmalah sebelum dimulai pemasangan dan instalasi praktikum dan dipandu oleh Asisten

1. **PRAKTIKUM BERLANGSUNG**
2. Asisten memberikan instruksi kepada kelompok praktikan pemasangan atau instalasi modul dan mengawasi dan mengevaluasi serta memeriksa hasil pemasangan dan memastikan kebenaran instalasi
3. Praktikan dan asisten saling menjaga kenyamanan dan ketertiban praktikum sesuai tata tertib yang berlaku serta menjaga keamanan perangkat lab selama pelaksanaan praktikum dari satu percobaan ke percobaan berikutnya.
4. Asisten berhak menegur dan menindak praktikan apabila ketahuan merusak, mengubah atau memindahkan perlengkapan lab tanpa ijin.
5. Asisten melakukan penilaian dan pengawasan tiap praktikan melakukan pengukuran selama percobaan.
6. Asisten dan kelompok praktikan mengakhiri praktikum dengan membaca hamdallah dan mengucap salam serta meminta praktikan untuk merapikan peralatan dan modul serta kursi dan membuang sampah di sekitarnya.

1. **PRAKTIKUM BERAKHIR**
2. Praktikan meninggalkan ruangan dengan rapi dan teratur.
3. Asisten Mengkondisikan ruangan kembali,
   1. Mengembalikan/mengatur kursi kembali.
   2. Merapikan sampah yang ditemukan berserakan dalam ruangan.
   3. Mengembalikan peralatan dan modul ke Lemari Alat dan Modul sesuai nama jenis Mata Praktikum
   4. Mengunci pintu
   5. Mematikan lampu apabila tidak ada praktikum berikutnya.
4. Asisten menandatangani presensi kelompok dan memberikan daftar penilaian kerja percobaan kelompok ke ruang administrasi (Laboran).
5. Instruktur dan atau asisten melakukan evaluasi reguler praktikum jika diperlukan.

1. **PASCA PRAKTIKUM**

Praktikan menyusun laporan semua percobaan

Praktikan melakukan asistensi laporan ke Asisten Praktikum min 4 kali

Setelah laporan praktikum ditandatangani oleh Asisten, Tiap Praktikum menghadap Instruktur sesuai jadwal yang ditetapkan Instruktur

Instruktur menguji praktikum mengenai proses pelaksanaan praktikum

Instruktur memberikan nilai akhir praktikan

Nilai akhir prakatikum diserahkan ke Lab untuk proses administrasi

1. **SANKSI**
2. Keterlambatan asistensi pertama kali sanksi point 1
3. Tidak memenuhi minimal 4 kali asistensi sanksi point 2
4. Datang terlambat 15 menit dari waktu yang telah ditentukan sanksi point 3
5. \* Tidak mengikuti proses praktikum tanpa adanya konfirmasi sanksi point 4
6. \* Tidak mengikuti ujian koordinator tanpa adanya konfirmasi sanksi point 5
7. Keterlambatan pengumpulan laporan resmi sanki point 6
8. \* Tidak mengikuti ujian instruktur sesuai dengan jadwal yang ditentukan instruktur sanksi point 7
9. Pemalsuan tanda tangan selama proses praktikum berlangsung sanksi point 8
10. Merusakkan perlatan Lab. Teknik Elektro sanksi point 9

\* Maksimal konfirmasi 2 x 24 jam sejak jadwal resmi diumumkan untuk penggantian jadwal ujian

|  |  |
| --- | --- |
| Point 1 | Menulis materi modul bab 1 |
| Point 2 | Menulis materi modul bab 1-3 & Pengurangan Nilai |
| Point 3 | Menulis materi 1 bab & Pengurangan Nilai |
| Point 4 | Mengulang (tidak konfirmasi sesuai waktu yang telah ditentukan) atau Pengurangan Nilai |
| Point 5 | Mengulang (tidak konfirmasi sesuai waktu yang telah ditentukan) atau Pengurangan Nilai |
| Point 6 | Membeli buku berkaitan dengan bidang Teknik elektro |
| Point 7 | Pengurangan Nilai Instruktur |
| Point 8 | Mengulang Praktikum atau mendapat Nilai E |
| Point 9 | Mengganti peralatan tersebut sesuai dengan spesifikasi atau mirip dan memiliki fungsi yang sama |

1. **KESELAMATAN DAN KESEHATAN KERJA (K3)**
2. Sebelum memulai praktikum, praktikan memahami tata tertib dan keselamatan di Laboratorium
3. Mengetahui tempat dan cara penggunaan perlatan Laboratorium
4. Memperhatikan dan waspada terhadap tempat-tempat sumber listrik ( stop kontak dan circuit breaker)
5. Praktikan harus memperhatikan dan menaati peringatan (warning) yang biasa tertera pada badan perlatan praktikum maupun rambu peringatan yang terdapat di ruangan Laboratorium
6. Jika melihat ada kerusakan yang berpotensi menimbulkan bahaya, segera laporkan ke asisten terkait atau dapat langsung melapor ke laboran.
7. Hindari daerah atau benda yang berpotensi menimbulkan bahaya listrik ( sengatan listrik) secara tidak sengaja, missal seperti jala-jala kabel yang terkelupas
8. Keringkan bagian tubuh yang basah, seperti keringat atau sisa air wudhu
9. Selalu waspada terhadap bahaya listrik pada setiap aktifitas praktikum.
10. Jika terjadi kecelakaan akibat bahaya listrik, berikut ini adalah hal-hal yang harus diikuti praktikan:
11. Jangan panik
12. Matikan semua peralatan elektronik dan sumber listrik di meja masing-masing dan di meja praktikum yang tersengat arus listrik.
13. Bantu praktikan yang tersengat arus listrik untuk melepaskan diri dari sumber listrik
14. Beritahukan dan minta bantuan kepada laboran, praktikan lain dan orang di sekitar anda tentang terjadinya kecelakaan akibat bahaya listrik.
15. Jangan membawa benda-benda mudah terbakar (korek api, gas, dll) ke dalam ruangan laboratorium bila tidak disyaratkan dalam modul praktikum.
16. Jangan melakukan sesuatu yang menimbulkan api, percikan api, atau panas yang berlebihan.
17. Jangan melakukan sesuatu yang menimbulkan bahaya api atau panas berlebih pada diri sendiri atau orang lain.
18. Selalu waspada terhadap bahaya api atau panas berlebih pada setiap aktivitas di laboratorium.
19. Jika terjadi kecelakaan akibat bahaya listrik, berikut ini adalah hal-hal yang harus diikuti praktikan:
20. Jangan panik
21. Matikan semua peralatan elektronik dan sumber listrik di meja masing-masing.
22. Beritahukan dan minta bantuan laboran, praktikan lain dan orang di sekitar anda tentang terjadinya bahaya api atau panas berlebih
23. Menjauh dari ruang praktikum
24. Dilarang membawa benda tajam (pisau, gunting dan sejenisnya) ke ruang praktikum bila tidak diperlukan untuk pelaksanaan percobaan
25. Dilarang memakai perhiasan dari logam misalnya cincin, kalung, gelang, dll
26. Hindari daerah, benda atau logam yang memiliki bagian tajam dan dapat melukai.
27. Tidak melakukan sesuatu yang dapat menimbulkan luka pada diri sendiri atau orang lain.

*Halaman ini sengaja dikosongkan*

# BAB I AKSES I/O dan ADC ESP8266

## Tujuan

Praktikan diharapkan dapat mengakses I/O pada ESP8266

## Dasar Teori

ESP8266 merupakan modul wifi yang berfungsi sebagai perangkat tambahan mikrokontroler seperti Arduino agar dapat terhubung langsung dengan wifi dan membuat koneksi TCP/IP.

Modul ini membutuhkan daya sekitar 3.3v dengan memiliki tiga mode wifi yaitu Station, Access Point dan Both (Keduanya). Modul ini juga dilengkapi dengan prosesor, memori dan GPIO dimana jumlah pin bergantung dengan jenis ESP8266 yang kita gunakan. Sehingga modul ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler.

Firmware default yang digunakan oleh perangkat ini menggunakan AT Command, selain itu ada beberapa Firmware SDK yang digunakan oleh perangkat ini berbasis opensource yang diantaranya adalah sebagai berikut:

* NodeMCU dengan menggunakan basic programming lua
* MicroPython dengan menggunakan basic programming python
* AT Command dengan menggunakan perintah perintah AT command

Untuk pemrogramannya sendiri kita bisa menggunakan ESPlorer untuk Firmware berbasis NodeMCU dan menggunakan putty sebagai terminal control untuk AT Command.

Selain itu kita bisa memprogram perangkat ini menggunakan Arduino IDE. Dengan menambahkan library ESP8266 pada board manager kita dapat dengan mudah memprogram dengan basic program arduino.

## Bahan Yang Diperlukan

* Laptop / PC
* Nodemcu/ESP8266
* Software Arduino
* Notepad++
* Koneksi Internet
* Tombol/buton yang terhubung dengan D4

## Langkah Percobaan

### 1.4.1 Akses Input

1. Buka Software Arduino IDE

Pilih file-> NewProject, lalu Pilih tools -> board -> pilih Nodemcu 1.0

1. Salin program Arduino berikut ke Arduino IDE

|  |
| --- |
| #include <ESP8266WiFi.h>  #define BUTTON D4 //On board LED  void setup(){  pinMode(BUTTON, INPUT);  Serial.begin(115200);  }  void loop(){  Serial.println(BUTTON);  delay(300);  } |

1. Buka Serial Monitor dan atur baudrate ke 115200 lalu tekan tombol pada D4

### 1.4.2 Akses Output

1. Buka Software Arduino IDE

Pilih file-> NewProject, lalu Pilih tools -> board -> pilih Nodemcu 1.0

1. Salin program Arduino berikut ke Arduino IDE

|  |
| --- |
| #include <ESP8266WiFi.h>  #define LED 2 //On board LED  void setup(){  pinMode(LED, OUTPUT);  }  void loop(){  digitalWrite(LED, HIGH);  delay(1000);  digitalWrite(LED, LOW);  delay(1000);  } |

### 1.4.3 Akses ADC

1. Buka Software Arduino IDE

Pilih file-> NewProject, lalu Pilih tools -> board -> pilih Nodemcu 1.0

1. Salin program Arduino berikut ke Arduino IDE

|  |
| --- |
| #include <ESP8266WiFi.h>  #define LED 2 //On board LED  void setup(){  Serial.begin(115200);  }  void loop(){  analogRead(A0);  Serial.println(A0);  delay(500);  } |

## Analisa Data

## Kesimpulan

# BAB II REALTIME GRAFIK WEB SERVER MENGGUNAKAN ESP8266

## Tujuan

Praktikan diharapkan dapat membuat menampilkan grafik melalui ESP8266 Secara Online

## Dasar Teori

Server atau Web server adalah sebuah software yang memberikan layanan berbasis data dan berfungsi menerima permintaan dari HTTP atau HTTPS pada klien yang dikenal dan biasanya kita kenal dengan nama web browser (Mozilla Firefox, Google Chrome) dan untuk mengirimkan kembali yang hasilnya dalam bentuk beberapa halaman web dan pada umumnya akan berbentuk dokumen HTML.

Fungsi utama Server atau Web server adalah untuk melakukan atau akan mentransfer berkas permintaan pengguna melalui protokol komunikasi yang telah ditentukan sedemikian rupa. Pemanfaatan web server berfungsi untuk mentransfer seluruh aspek pemberkasan dalam sebuah halaman web. Halaman web yang diminta terdiri dari berkas teks, video, gambar, file dan banyak lagi..

## Bahan Yang Diperlukan

* Laptop / PC
* Nodemcu/ESP8266
* Software Arduino
* Notepad++
* Koneksi Internet

## Langkah Percobaan

1. Buka Software Arduino IDE

Pilih file-> NewProject, lalu Pilih tools -> board -> pilih Nodemcu 1.0

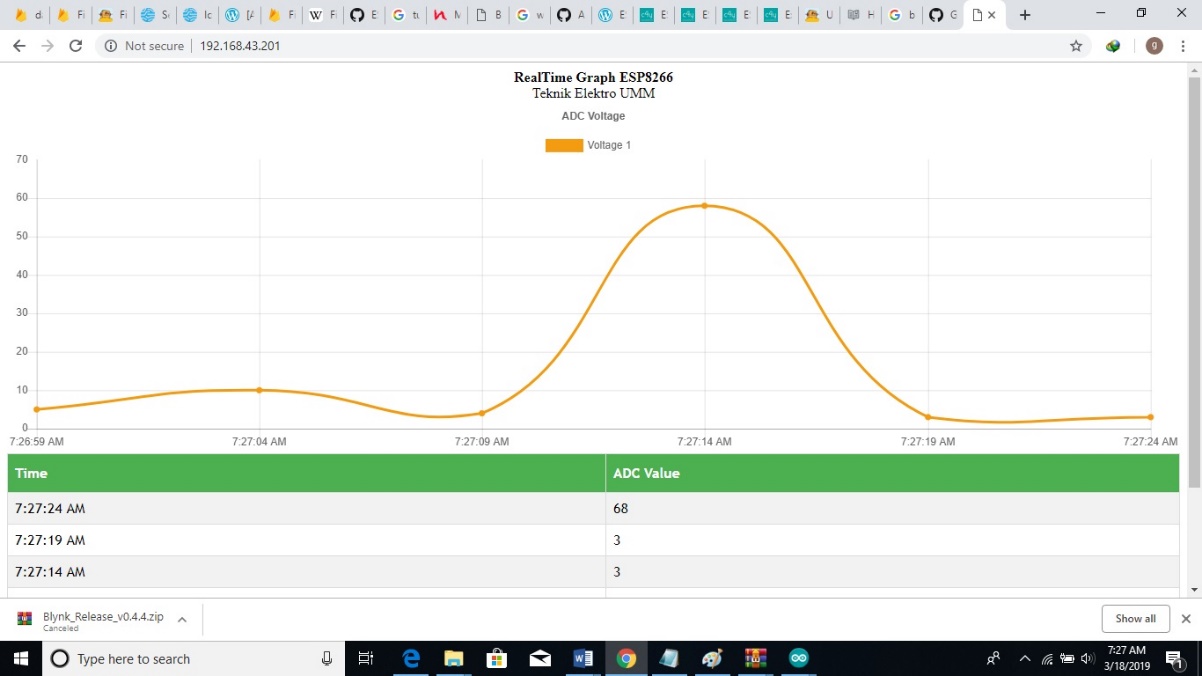
1. Salin program Arduino berikut ke Arduino IDE

|  |
| --- |
| #include <ESP8266WiFi.h>  #include <WiFiClient.h>  #include <ESP8266WebServer.h>  #include "index.h"  #define LED 2 //On board LED  const char\* ssid = "asus";  const char\* password = "15091997";  ESP8266WebServer server(80); //Server on port 80  void handleRoot() {  String s = MAIN\_page;  server.send(100, "text/html", s);  }  void handleADC() {  int a = analogRead(A0);  String adcValue = String(a);  digitalWrite(LED,!digitalRead(LED));  server.send(100, "text/plane", adcValue);  }  void setup(void){  Serial.begin(115200);    WiFi.begin(ssid, password);  Serial.println("");  pinMode(LED,OUTPUT);    while (WiFi.status() != WL\_CONNECTED) {  delay(500);  Serial.print(".");  }  Serial.println("");  Serial.print("Connected to ");  Serial.println(ssid);  Serial.print("IP address: ");  Serial.println(WiFi.localIP());    server.on("/", handleRoot);  server.on("/readADC", handleADC);  server.begin(); //Start server  Serial.println("HTTP server started");  }  void loop(void){  server.handleClient();  }  void loop()  {  Blynk.run();  } |

1. Replace SSID dan Password sesuai SSID dan Pasword router yang akan dihubungkan ke Nodemcu.
2. Salin program berikut ke Notepad++

|  |
| --- |
| const char MAIN\_page[] PROGMEM = R"=====(  <!doctype html>  <html>  <head>  <title>Line Chart | Nodemcu Real Time Graph</title>  <script src = "https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script>  <style>  canvas{  -moz-user-select: none;  -webkit-user-select: none;  -ms-user-select: none;  }  /\* Data Table Styling \*/  #dataTable {  font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;  border-collapse: collapse;  width: 100%;  }  #dataTable td, #dataTable th {  border: 1px solid #ddd;  padding: 8px;  }  #dataTable tr:nth-child(even){background-color: #f2f2f2;}  #dataTable tr:hover {background-color: #ddd;}  #dataTable th {  padding-top: 12px;  padding-bottom: 12px;  text-align: left;  background-color: #4CAF50;  color: white;  }  </style>  </head>  <body>  <div style="text-align:center;"><b>RealTime Graph ESP8266</b><br>Teknik Elektro UMM</div>  <div class="chart-container" position: relative; height:350px; width:100%">  <canvas id="Chart" width="400" height="400"></canvas>  </div>  <div>  <table id="dataTable">  <tr><th>Time</th><th>ADC Value</th></tr>  </table>  </div>  <br>  <br>  <script>  var values = [];  var timeStamp = [];  function showGraph()  {  for (i = 0; i < arguments.length; i++) {  values.push(arguments[i]);  }  var ctx = document.getElementById("Chart").getContext('2d');  var Chart2 = new Chart(ctx, {  type: 'line',  data: {  labels: timeStamp,  datasets: [{  label: "Voltage 1",  fill: false, //Try with true  backgroundColor: 'rgba( 243, 156, 18 , 1)',  borderColor: 'rgba( 243, 156, 18 , 1)', //Graph Line Color  data: values,  }],  },  options: {  title: {  display: true,  text: "ADC Voltage"  },  maintainAspectRatio: false,  elements: {  line: {  tension: 0.5 //Smoothening (Curved) of data lines  }  },  scales: {  yAxes: [{  ticks: {  beginAtZero:true  }  }]  }  }  });  }  //On Page load show graphs  window.onload = function() {  console.log(new Date().toLocaleTimeString());  showGraph(5,10,4,58);  };  //Ajax script to get ADC voltage at every 5 Seconds  //Read This tutorial https://circuits4you.com/2018/02/04/esp8266-ajax-update-part-of-web-page-without-refreshing/  setInterval(function() {  // Call a function repetatively with 5 Second interval  getData();  }, 5000); //5000mSeconds update rate    function getData() {  var xhttp = new XMLHttpRequest();  xhttp.onreadystatechange = function() {  if (this.readyState == 4 && this.status == 200) {  //Push the data in array  var time = new Date().toLocaleTimeString();  var ADCValue = this.responseText;  values.push(ADCValue);  timeStamp.push(time);  showGraph(); //Update Graphs  //Update Data Table  var table = document.getElementById("dataTable");  var row = table.insertRow(1); //Add after headings  var cell1 = row.insertCell(0);  var cell2 = row.insertCell(1);  cell1.innerHTML = time;  cell2.innerHTML = ADCValue;  }  };  xhttp.open("GET", "readADC", true); //Handle readADC server on ESP8266  xhttp.send();  }    </script>  </body>  </html>  )====="; |

1. Simpan file dengan nama index.h dan taruh dalam 1 folder dengan program Arduino ide yang telah dibuat,
2. Klik upload pada program Arduino ide
3. Klik serial monitor dan setting baudrate ke 115200, lalu tekan tombol reset pada nodemcu,
4. Setelah ip address muncul di serial monitor copy ip address dan paste di web browser sehingga akan muncul tampilan seperti dibawah ini



## Analisa Data

## Kesimpulan

# BAB III KONTROL LED ESP8266 MENGGUNAKAN APLIKASI BLYNK

## Tujuan

Praktikan diharapkan dapat membuat UI sederhana untuk mengontrol led pada esp8266 menggunakan aplikasi Blynk

## Dasar Teori

**BLYNK** adalah platform untuk aplikasi OS Mobile (iOS dan Android) yang bertujuan untuk kendali module Arduino, Raspberry Pi, ESP8266, WEMOS D1, dan module sejenisnya melalui Internet.

Aplikasi ini merupakan wadah kreatifitas untuk membuat antarmuka grafis untuk proyek yang akan diimplementasikan hanya dengan metode drag and drop widget.

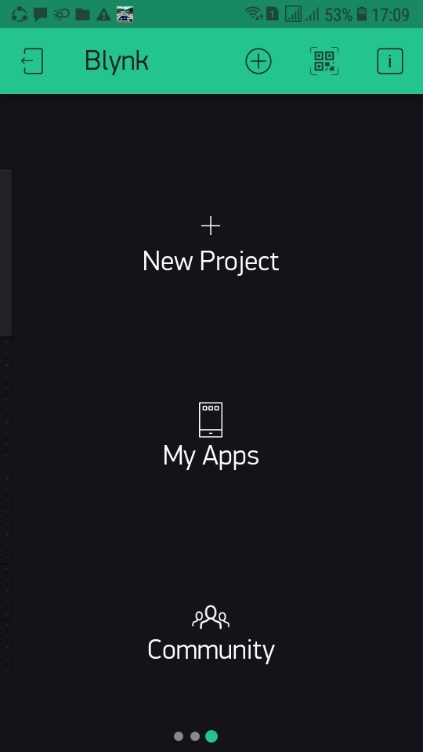
Penggunaannya sangat mudah untuk mengatur semuanya dan dapat dikerjakan dalam waktu kurang dari 5 menit. **Blynk** tidak terikat pada papan atau module tertentu. Dari platform aplikasi inilah dapat mengontrol apapun dari jarak jauh, dimanapun kita berada dan waktu kapanpun. Dengan catatan terhubung dengan internet dengan koneksi yang stabil dan inilah yang dinamakan dengan sistem Internet of Things (**IOT**)

## Bahan Yang Diperlukan

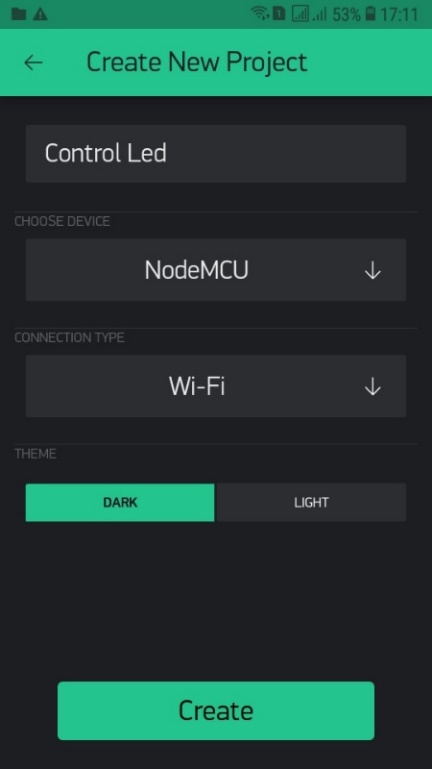
* Laptop / PC
* Nodemcu/ESP8266
* Software Arduino
* Koneksi Internet

## Langkah Percobaan

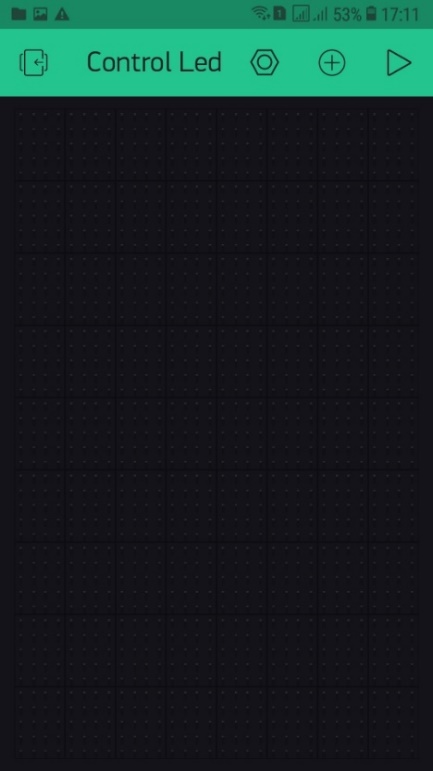
1. Download dan Install aplikasi blynk pada playstore
2. Buka Aplikasi Blynk dan klik New Project



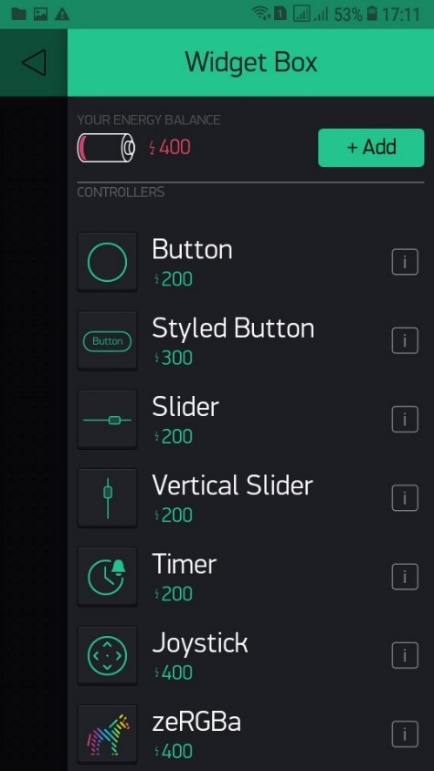
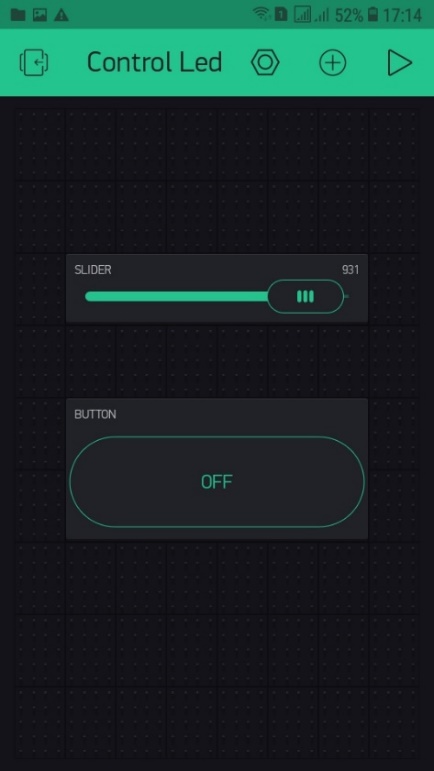
1. Buat Nama Project, pada menu choose device pilih NodeMCU, dan connection tipe pilih WIFI, lalu klik create



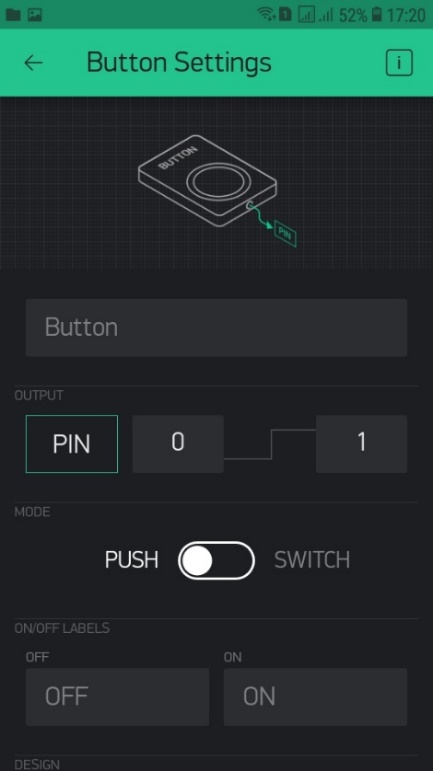
1. Setelah Project berhasil dibuat, klik gambar dengan logo + atau widgetBox



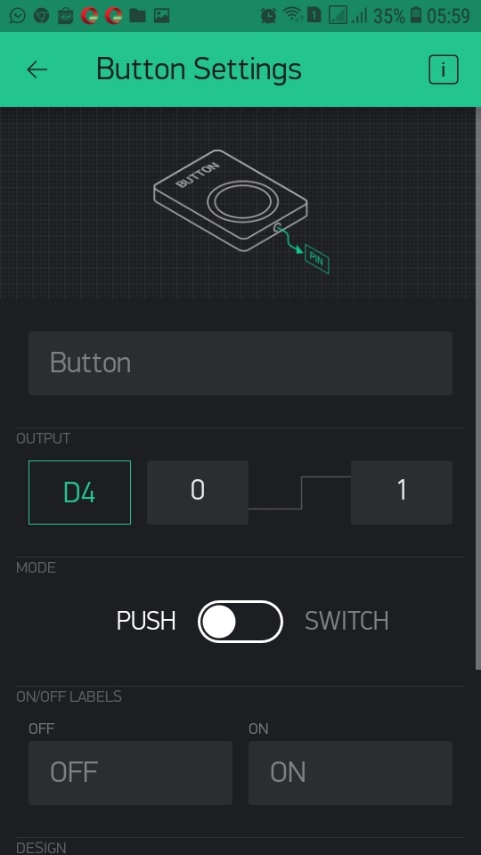
1. Pilih Button dan Slider, kemudian tempatkan sesuai keinginan

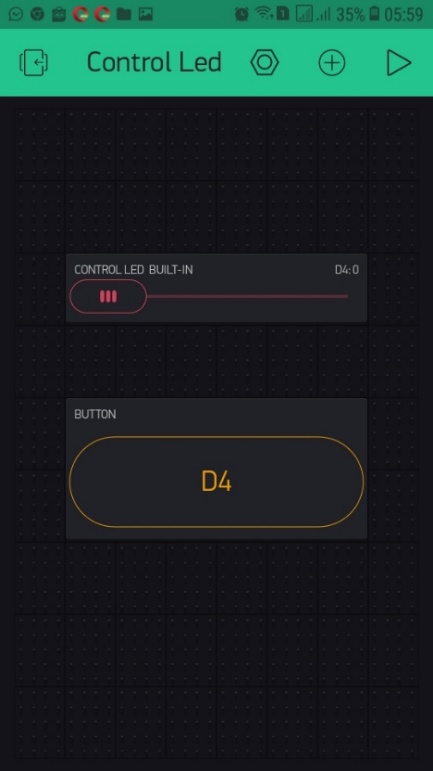
1. Tekan gambar button sehingga muncul menu seperti dibawah ini



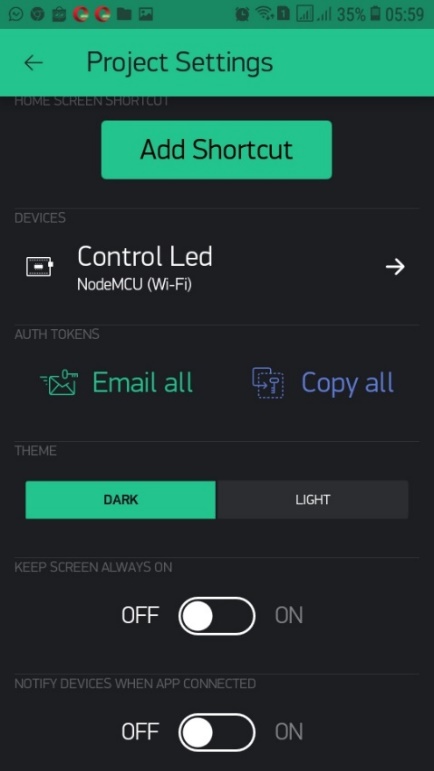
1. Ganti PIN menjadi D4



1. Lakukan hal yang sama untuk slider (tekan slider, ganti PIN menjadi D4 lalu kembali ke main project
2. Klik/tekan project setting (gambar segi 6)



1. Lalu tekan “Copy All” pada menu Auth Token



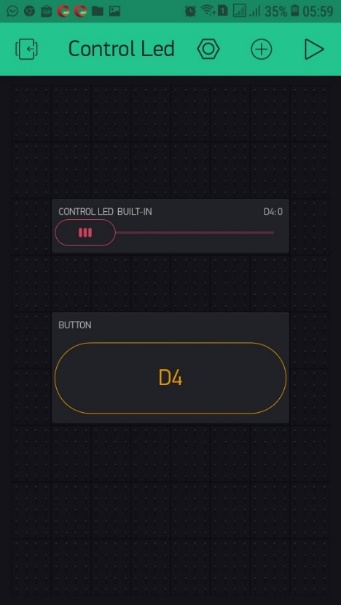
1. Buka Software Arduino IDE

Pilih file-> NewProject, lalu Pilih tools -> board -> pilih Nodemcu 1.0

1. Salin program Arduino berikut

|  |
| --- |
| #include <ESP8266WiFi.h>  #include <BlynkSimpleEsp8266.h>  char auth[] = "YourAuthToken";  char ssid[] = "YourNetworkName";  char pass[] = "YourPassword";  void setup()  {  Blynk.begin(auth, ssid, pass);  }  void loop()  {  Blynk.run();  } |

1. Replace Auntefikasi Auth sesuai autentifikasi yang telah dicopy sebelumnya.
2. Replace SSID dan Password sesuai SSID dan Pasword router yang akan dihubungkan ke Nodemcu lalu klik Upload, dan geser slider dan button untuk mengatur lampu led pada Nodemcu.



## Analisa Data

## Kesimpulan

# BAB IV KIRIM DAN TERIMA DATA ESP8266 MELALUI GOOGLE FIREBASE

## Tujuan

Praktikan diharapkan dapat mengerti cara mengirim dan menerima data dari firebase google

## Dasar Teori

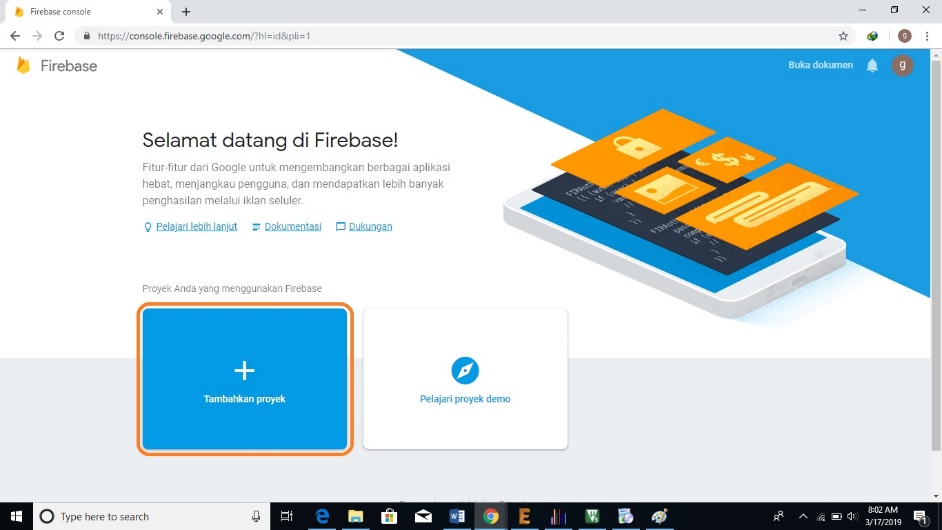
Firebase adalah suatu layanan dari [Google](https://id.wikipedia.org/wiki/Google) yang digunakan untuk mempermudah para pengembang aplikasi dalam mengembangkan aplikasi. Dengan adanya Firebase, pengembang aplikasi bisa fokus mengembangkan aplikasi tanpa harus memberikan usaha yang besar. Dua fitur yang menarik dari Firebase yaitu Firebase Remote Config dan Firebase Realtime Database. Selain itu terdapat fitur pendukung untuk aplikasi yang membutuhkan pemberitahuan yaitu Firebase Notification.

## Bahan Yang Diperlukan

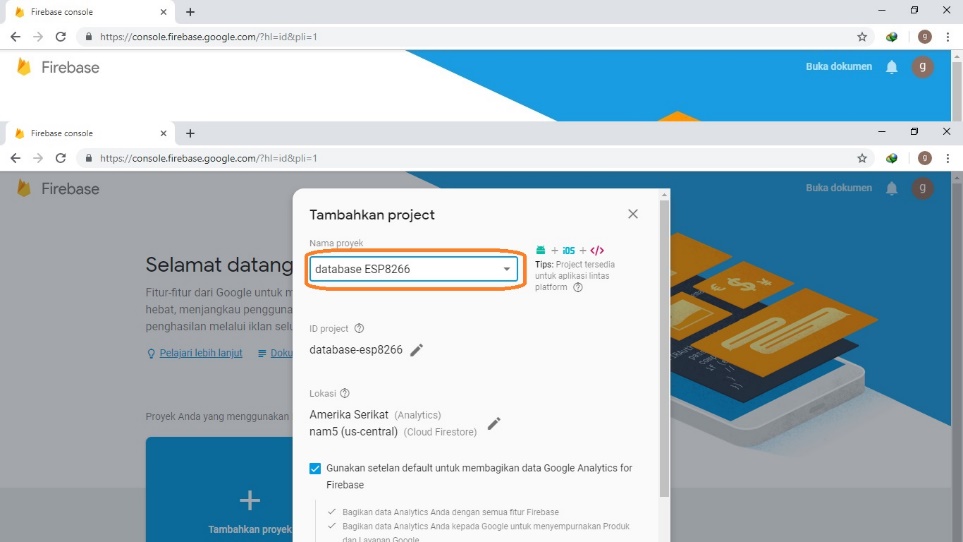
* Laptop / PC
* Nodemcu/ESP8266
* Software Arduino
* Koneksi Internet

## Langkah Percobaan

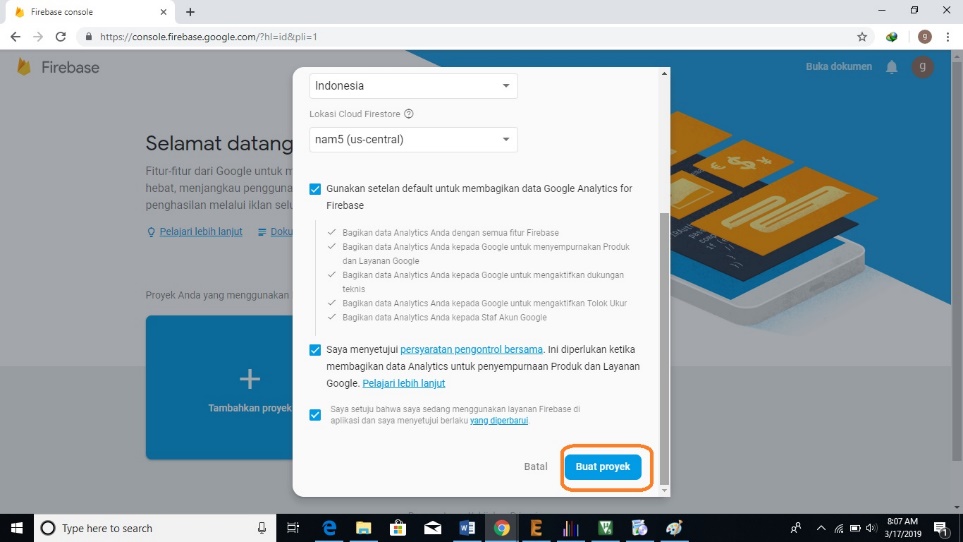
1. Buat akun Firebase di <https://console.firebase.google.com>
2. Pilih Tambahkan Proyek



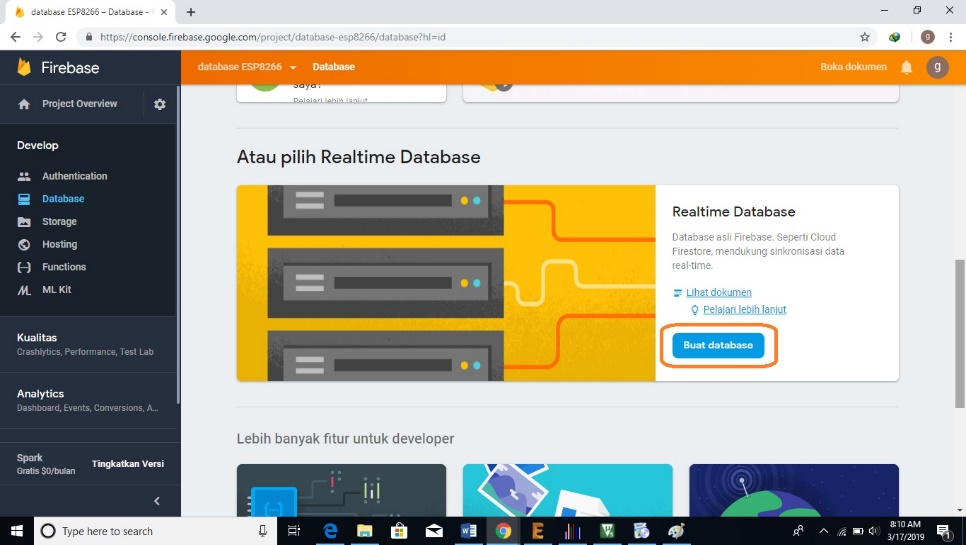
1. Beri nama proyek sesuai keinginan



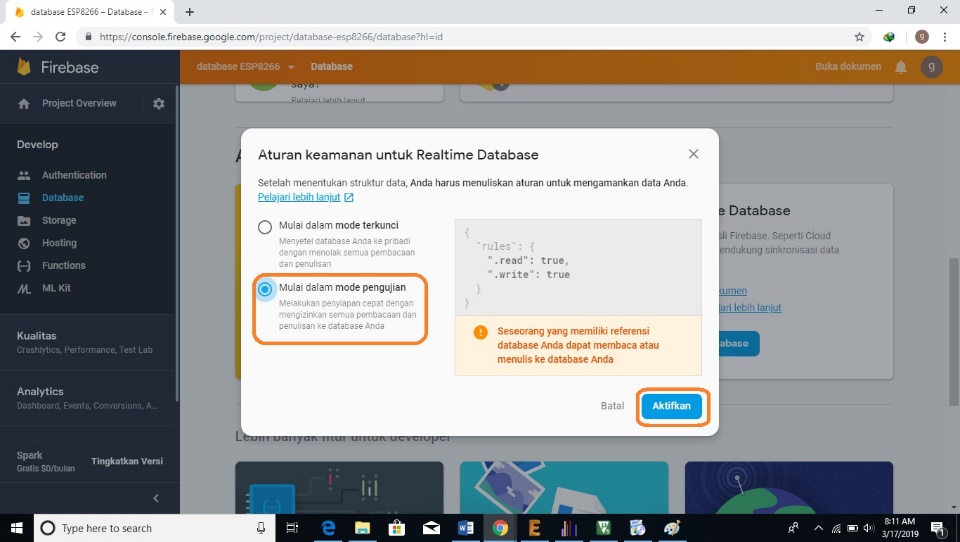
1. Pada Menu lokasi, ganti dari default (united states) ke Indonesia lalu pilih buat proyek



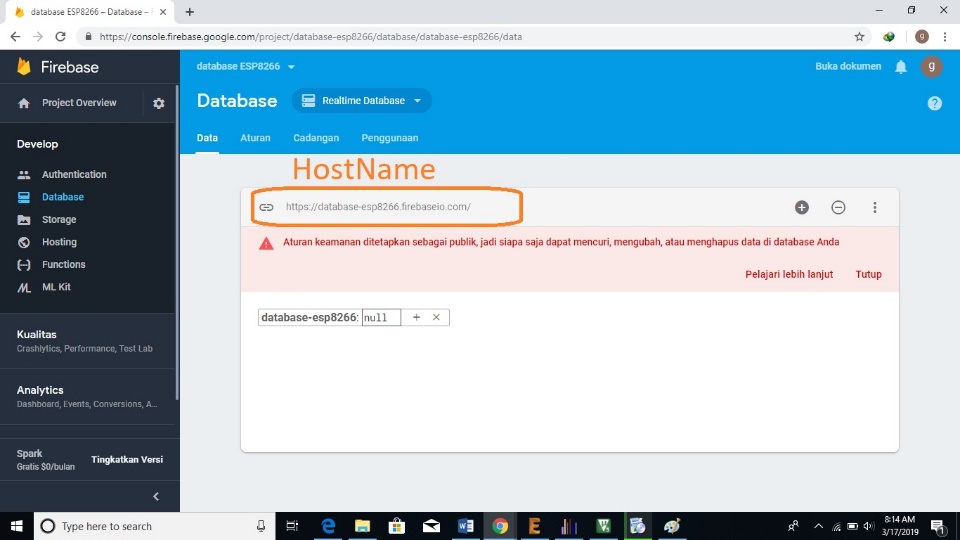
1. Setelah control page FireBase muncul, pilih menu develop lalu klik database
2. Pada menu realtime database klik buat database



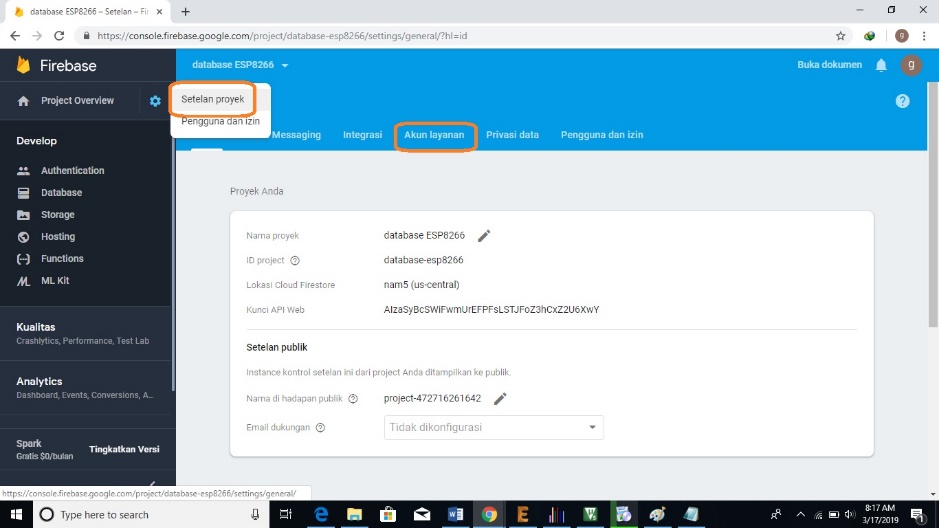
1. Lalu pilih “mulai dalam mode pengujian” dan klik “aktifkan”



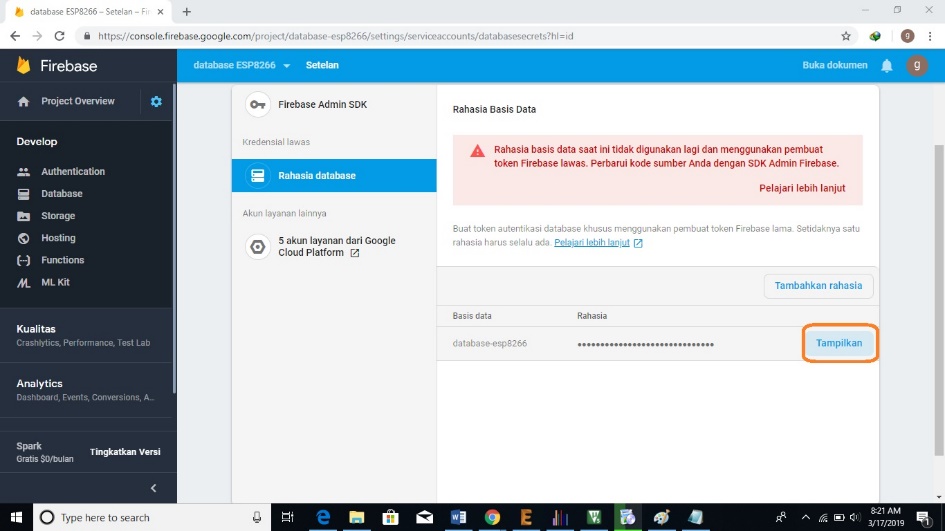
1. Setelah database selasi dibuat, Salin host name untuk url push nodemcu



1. Pada Project Overview klik gambar gear lalu pilih setelan proyek kemudian pilih akun layanan



1. Pada menu TampilkanRahasia klik “tampilkan“



1. Salin secret key



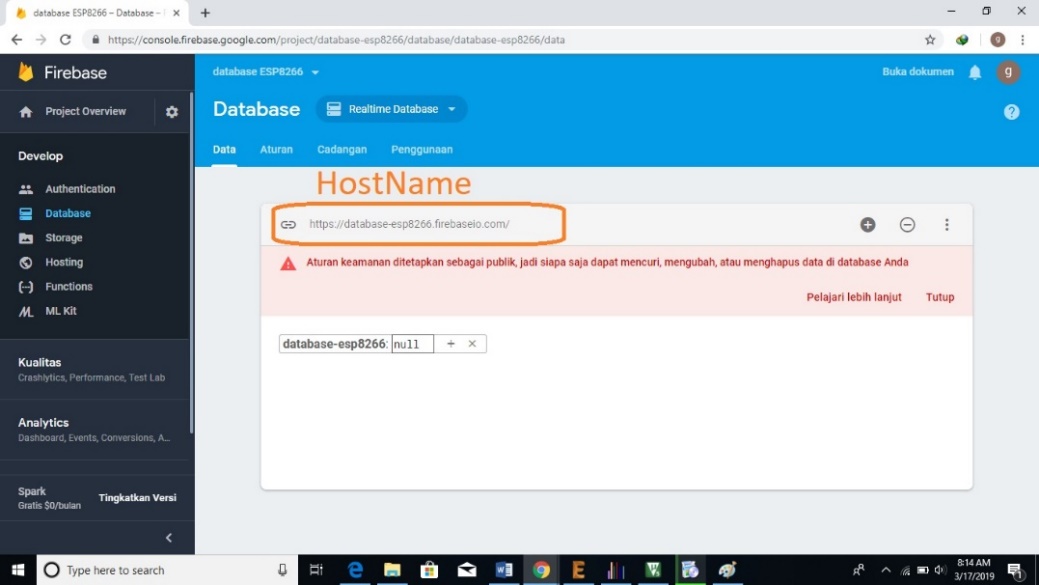
1. Buka Software Arduino IDE

Pilih file-> NewProject, lalu Pilih tools -> board -> pilih Nodemcu 1.0

1. Salin Program berikut ke Arduino IDE

|  |
| --- |
| #include <ESP8266WiFi.h> // esp8266 library  #include <FirebaseArduino.h> // firebase library  #define FIREBASE\_HOST "database-esp8266.firebaseio.com" //Paste Host Name Proyek FireBase Disini  #define FIREBASE\_AUTH "JMwmfmopfklS72MQLlqJnu8zvcxCDy643OcYuE2W" //Paste kode rahasia/autentifikasi auth Proyek FireBase Disini  #define WIFI\_SSID "Rumahku\_Istanaku" // input SSID wifi yang akan disambungkan ke FireBase  #define WIFI\_PASSWORD "15091997" // input password wifi yang akan disambungkan ke FireBase  String fireStatus = "";  void setup() {  Serial.begin(9600);  delay(1000);  pinMode(D4, OUTPUT);  WiFi.begin(WIFI\_SSID, WIFI\_PASSWORD);  Serial.print("Connecting to ");  Serial.print(WIFI\_SSID);  while (WiFi.status() != WL\_CONNECTED) {  Serial.print(".");  delay(500);  }  Serial.println();  Serial.print("Connected to ");  Serial.println(WIFI\_SSID);  Serial.print("IP Address is : ");  Serial.println(WiFi.localIP()); //print local IP address  Firebase.begin(FIREBASE\_HOST, FIREBASE\_AUTH); // connect to firebase  Firebase.setString("LED\_STATUS", "OFF"); // kirim string LED\_STATUS dengan nilai OFF ke FireBase  }  void loop() {  fireStatus = Firebase.getString("LED\_STATUS"); // terima/get string kondisi LED\_STATUS dari FireBase  if (fireStatus == "ON") {  Serial.println("Led Turned ON");  digitalWrite(D4, LOW);  }  else if (fireStatus == "OFF") {  Serial.println("Led Turned OFF");  digitalWrite(D4, HIGH);  }  } |

1. Replace Host “Name” dan “kode rahasia/autentifikasi auth” sesuai pada proyek firebase masing masing
2. Replace SSID dan password sesuai wifi/akses point yang akan dihubungkan ke nodemcu
3. Lalu Buka Kembali database Firebase yang telah dibuat dan tambahkan judul LED\_STATUS dengan menekan tombjol +, dan ganti status menjadi ON/OFF untuk menyalakan dan Mematikan Led pada Nodemcu



## Analisa Data

## Kesimpulan

*Halaman ini sengaja dikosongkan*