

## BAB III METODOLOGI PENELITIAN

### 3.1 Waktu dan Tempat Penelitian

Waktu dan tempat penelitian merupakan serangkaian gambaran umum yang menjelaskan lokasi serta waktu dalam mengumpulkan data dalam sebuah penelitian atau riset.

#### 3.1.1 Waktu Penelitian

Penelitian dilaksanakan pada semester ganjil dan genap tahun akademik 2021/2022, tabel waktu penelitian sebagai berikut :

Tabel 3.1 Jadwal Penelitian

No	Kegiatan	Bulan Ke-					
		1	2	3	4	5	6
1	Pengumpulan Data	■					
2	Perancangan Alat		■				
3	Penentuan Alat dan Bahan		■				
4	Perancangan Perangkat Lunak			■	■		
5	Uji Coba Alat					■	
6	Pembuatan Laporan						■

#### 3.1.2 Tempat Penelitian

Dalam penelitian ini, penulis mengambil tempat di perusahaan SHATOMEDIA Yogyakarta, dan observasi di SMA Negeri 1 Bantarsari beralamat di JL. KH. Syarbini Hasan No. 2, Bulaksari, Kec. Bantarsari, Kab. Cilacap Prov. Jawa Tengah.

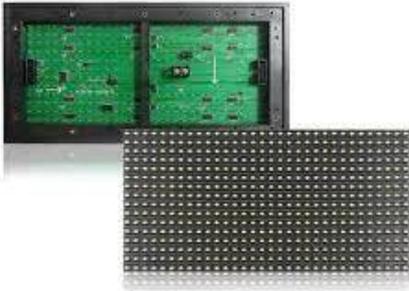
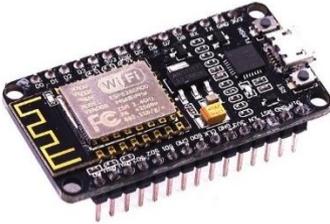
### 3.2 Alat dan Bahan

Pada perancangan dan pembuatan sistem ini dibutuhkan beberapa alat dan bahan untuk membuat sistem monitoring ini guna mengurangi penumpukan antrian. Daftar alat dan bahan yang digunakan dalam melakukan perancangan dan pembuatan sistem ini sebagai berikut :

### 3.2.1 Kebutuhan Perangkat Keras

Kebutuhan perangkat keras dalam penelitian ini ditunjukkan pada Tabel 3.2.

Tabel 3.2 Perangkat Keras

No	Kebutuhan Perangkat Keras	Spesifikasi
1.		Laptop Ram 4GB, processor AMD A9-9425 RADEON R5, 5 COMPUTE CORES 2C+3G 3.10GHz
2.		Led matrix modul P10 Resolusi 16 LED (T) x 32 LED (L), tegangan <i>input</i> DC 5V, jarak antar LED 10mm, jumlah LED 512per modul
3.		<i>Power supply</i> V input = 220-230 VAC V output = 5 VDC I output = 10 A
4.		NodeMCU ESP8266 Tegangan <i>input</i> 3.3 – 5V, GPIO 13 PIN, frekuensi 2.4 GHz – 22.5 Ghz, USB port <i>micro</i> USB

### 3.2.2 Kebutuhan Perangkat Lunak

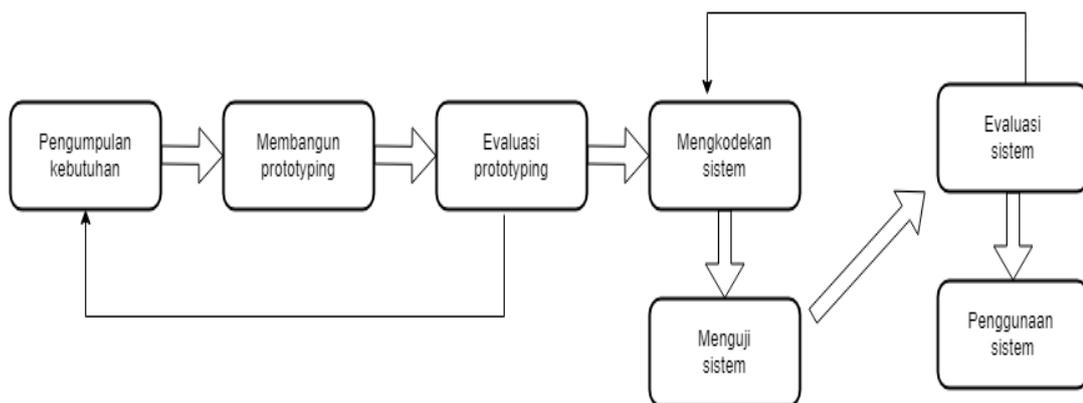
Kebutuhan perangkat keras dalam penelitian ini ditunjukkan pada Tabel 3.2.

Tabel 3.3 Perangkat Lunak

No	Kebutuhan Perangkat Lunak	Spesifikasi
1.	Windows	Versi windows 10 pro
2.	Arduino IDE	Versi Arduino 1.8.15, Bahasa C++
3.	Visual Studio code	Versi Visual Studio code 1.6, Bahasa HTML
4.	Fritzing	Versi Fritzing 0.9.3 (b04.199)64, Bahasa C++

### 3.3 Alur Penelitian

Dalam melakukan pembuatan sistem ini dilakukan perancangan setelah mengetahui latar belakang dari sistem yang dibuat. Setelah itu menentukan alat dan bahan apa saja yang dibutuhkan untuk membuat sistem untuk menapilakan waktu dan running teks. Dalam menyusun tugas akhir ini, terdapat beberapa tahapan yang dapat dilihat pada diagram alir dibawah ini:



Gambar 3. 1 Diagram Alur Penelitian

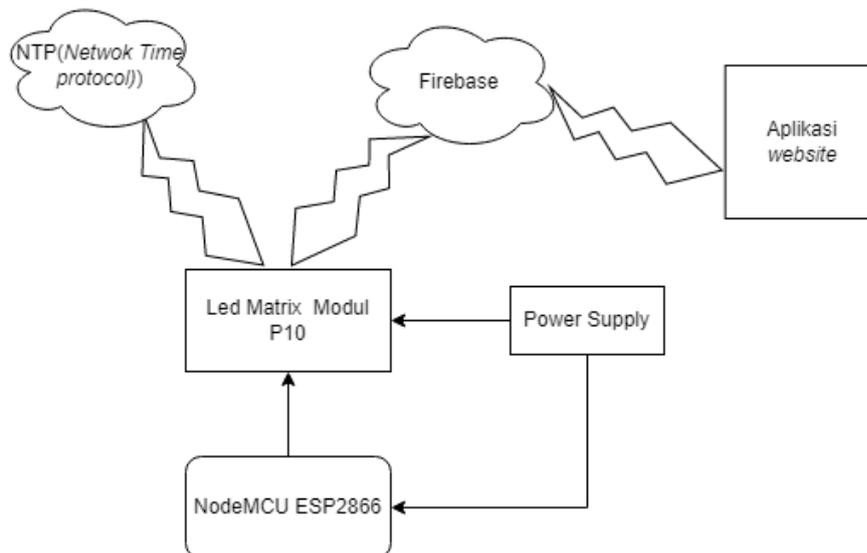
#### 3.3.1 Perancangan Sistem

Setelah menentukan alat dan bahan yang dibutuhkan pada penelitian ini, maka langkah selanjutnya adalah merancang sistem. Tujuan perancangan sistem yaitu memberikan gambaran yang jelas.

### 3.3.2 Perancangan Perangkat Keras

Pada perancangan ini digunakan NodeMCU ESP8266 sebagai pengendali dari perangkat keras. NodeMCU ESP8266 dipilih karena sudah terdapat *wifi* modul didalamnya, sehingga bisa bertugas sebagai *client* maupun *access point*. NodeMCU ESP8266 ini akan membaca apakah waktu sinkron atau tidak dan apakah mau mengganti papan informasi pada setiap ruang sekolah. NodeMCU ESP8266 disini memiliki dua tugas pengambilan data yaitu pengambilan data pesan yang tersimpan di *database* Firebase dan pengambilan waktu NTP (*network time prtocol*)

Ketika NodeMCU ESP8266 mengambil data dari NTP (*network time prtocol*) berupa data waktu, maka data tersebut akan ditampilkan ke LED Matrix (Modul P10) untuk mengeksekusi program yang telah diatur. Setelah NodeMCU ESP8266 mendapatkan dua data tersebut, LED Matrix (Modul P10) akan menampilkan hasil dari pengambilan data NTP (*network time prtocol*) dan data pesan yang sudah dirubah atau dikendalikan oleh komputer kendali berupa aplikasi *website* tersebut.

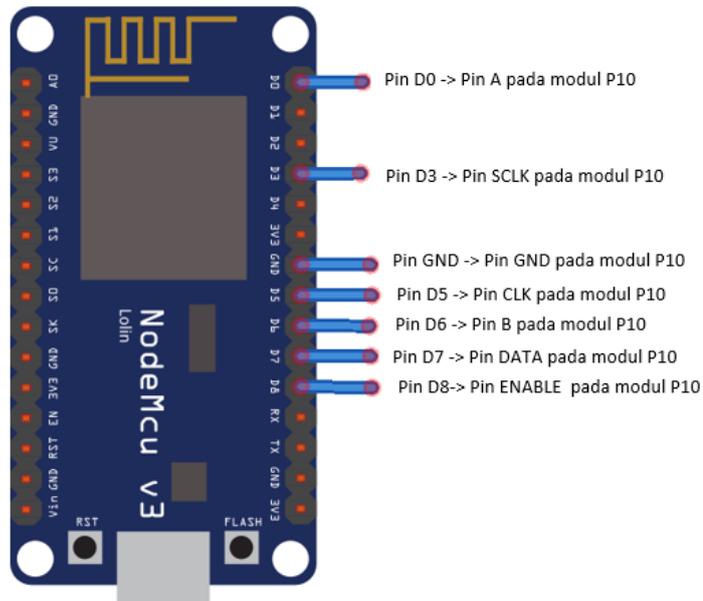


Gambar 3. 2 Diagram Blok Sistem

*Power supply* di gunakan untuk mengaliri listrik di setiap komponen, dan NodeMCU ESP8266 akan selalu membaca data dari *database* Firebase secara *realtime*. Ketika terdapat perubahan pada Firebase yang di *input* dari *website*, NodeMCU akan langsung membaca dan memberikan perintah ke LED Matrix (Modul P10) untuk menampilkan. Fungsi dari alat dan bahan yang digunakan yaitu:

## 1. NodeMCU ESP8266

NodeMCU ESP8266 disini bertugas sebagai mikrokontroler yang akan mengambil data pesan yang ada di *database* Firebase pesan tersebut ada dua tipe yaitu *database pesan secara private* dan pesan secara *global*, selanjutnya NodeMCU ESP8266 mengirimkan perintah kepada LED Matrix (Modul P10) untuk menampilkan data waktu yang sudah di ambil dari NTP (*Network Time Protocol*) dan teks pesan secara *private* atau pesan secara *global*, ketika admin sudah menginputkan teks yang di inginkan maka data tersebut akan di kirim ke Firebase melalui aplikasi *website* . NodeMCU ESP8266 dipilih karena sudah terdapat modul *wifi*.



Gambar 3. 3 konfigurasi NodeMCU ESP8266

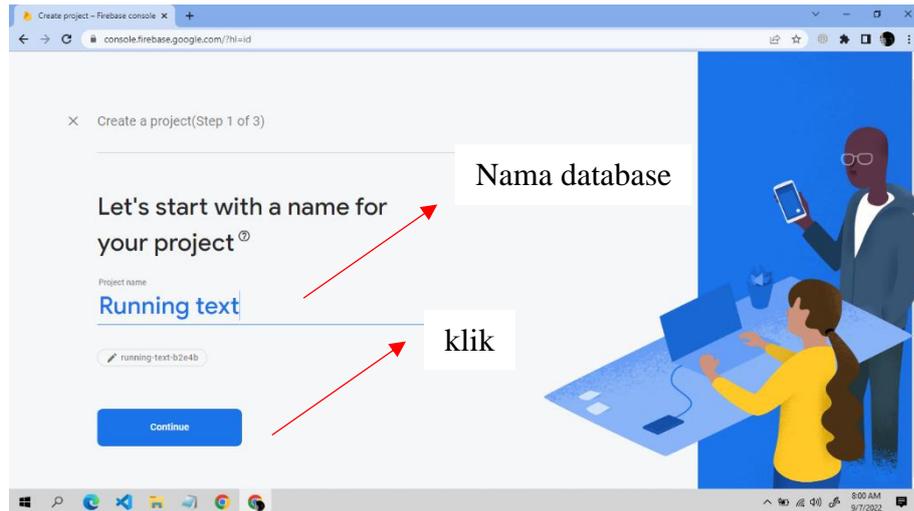
Tabel 3.4 Konfigurasi Pin NodeMCU ESP8266

No	Pin NodeMCU ESP8266	DMD P10	Power Supply
1	D0	A	-
2	D1	-	-
3	D2	-	-
4	D3	SCLK	-
5	D4	-	-
6	3V	-	-

7	GND	GND	GND
8	D5	CLK	-
9	D6	B	-
10	D7	R/DATA	-
11	D8	ENABLE	-
12	RX	-	-
13	TX	-	-
14	GND	GND	GND
15	3V	-	-
16	VIN	-	V+(5V)
17	GND	GND	GND
18	RST	-	-
19	EN	-	-
20	3V	-	-
21	GND	GND	GND
22	SK	-	-
23	SO	-	-
24	SC	-	-
25	S1	-	-
26	S2	-	-

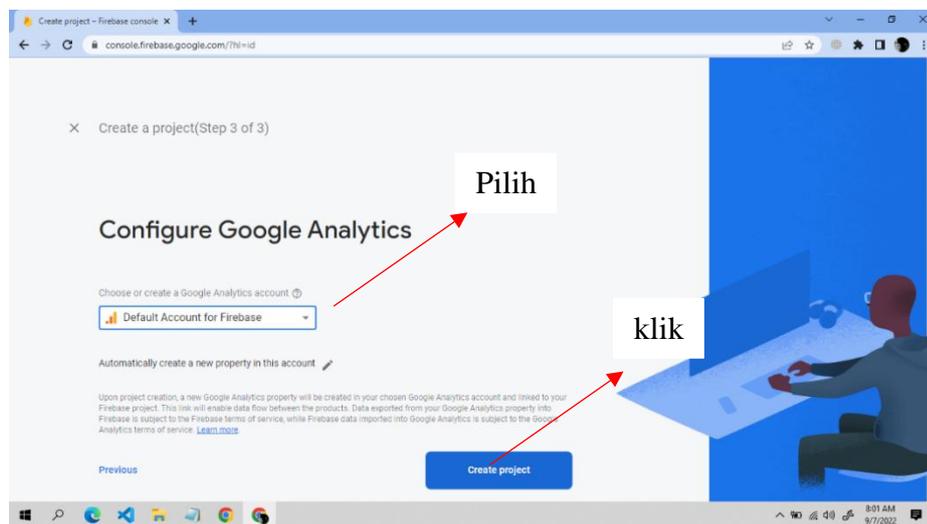
## 1 Firebase

Firebase digunakan sebagai penyimpan data informasi berupa *database* yang nantinya akan ditampilkan ke LED Matrix (Modul P10), *database* tersebut akan diproses oleh Node MCU ESP8266 kemudian di tampilkan di LED Matrix (Modul P10). Disaat data *running text* diubah melalui apalikasi *webiste*, maka Firebase akan memperbarui data tersebut dan ditampilkan ke LED Matrix (Modul P10). *Database* Firebase perlu dibuat agar Node MCU ESP8266 bisa membaca data informasi berupa pesan yang disimpan *didatabase* Firebase. Berikut langkah-langkah membuat *database* Firebase.



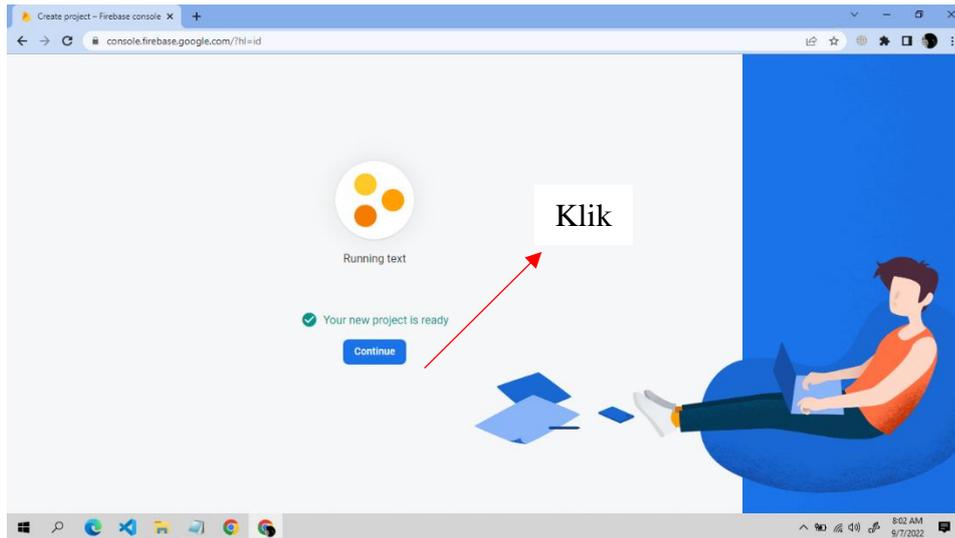
Gambar 3. 4 nama *database*

Langkah pertama yang perlu dilakukan untuk membuat *database* Firebase yaitu membuat nama *database* di Firebase yang ingin digunakan. Ketikkan nama *database* di *your project* kemudian klik *continue*.



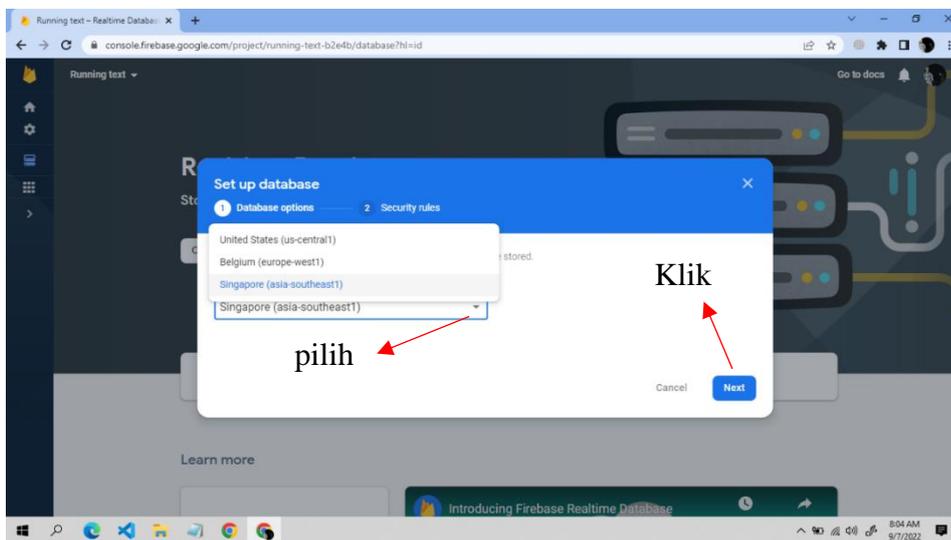
Gambar 3. 5 *default database*

Langkah kedua setelah membuat nama *database* yang akan dibuat untuk nama proyek kemudian ditampilkan *Configure Google Analytics* dimana perlu memilih *default account for Firebase database* yang ingin dituju selanjutnya klik *create project*.



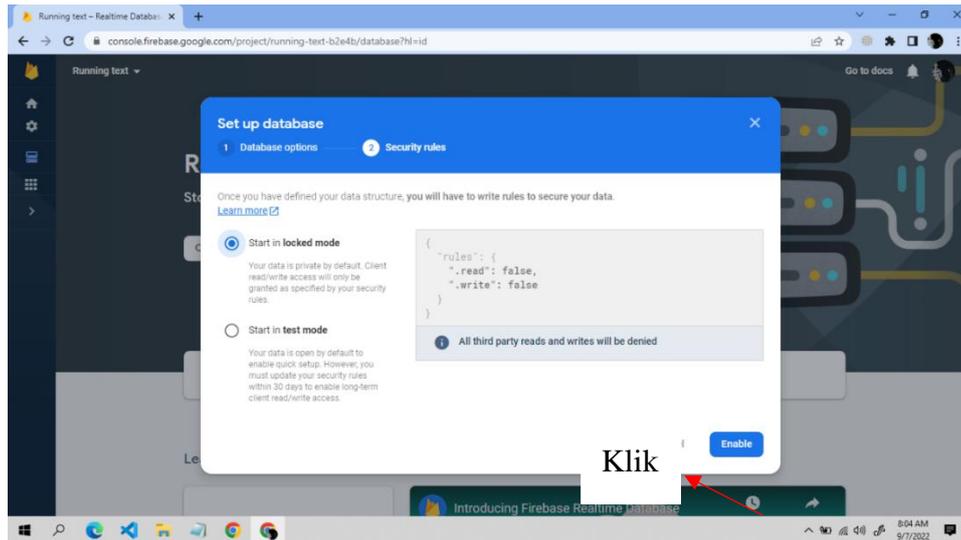
Gambar 3. 6 selesai *continue database*

Setelah memilih *default account for Firebase database*, tunggu beberapa menit sesuai jaringan internet proses *creat project* kemudian klik *continue*.



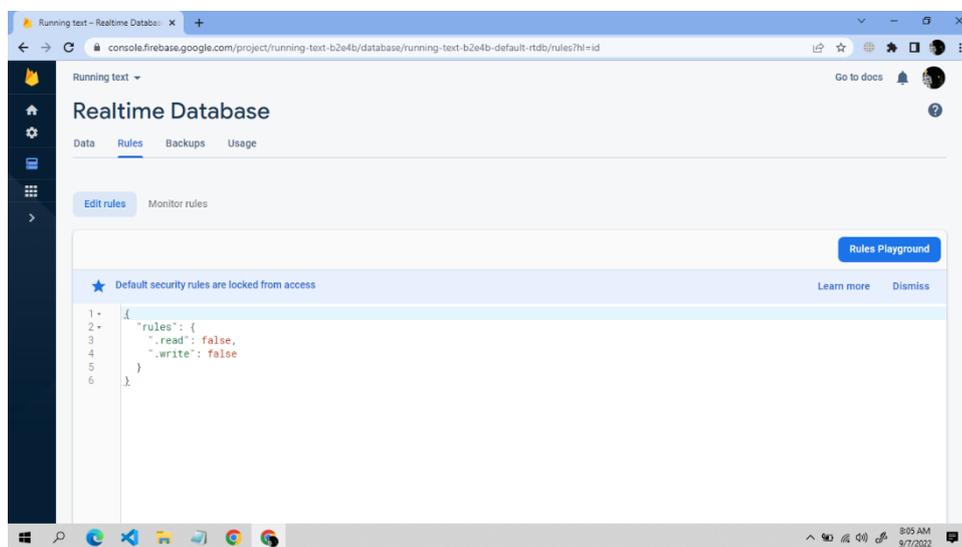
Gambar 3. 7 *set up database options*

Setelah proses pembuatan *database* selanjutnya perlu konfigurasi memilih *default database*, kemudian ada menu *realtime database* didalamnya ada menu *set up options databasa* dengan tiga pilihan, pilih sesuai keinginan dan klik next.



Gambar 3. 8 *set up database security rules*

Kemudian, dilanjutkan ke *security rules* pilih *start in locked mode* dan klik *enable*.



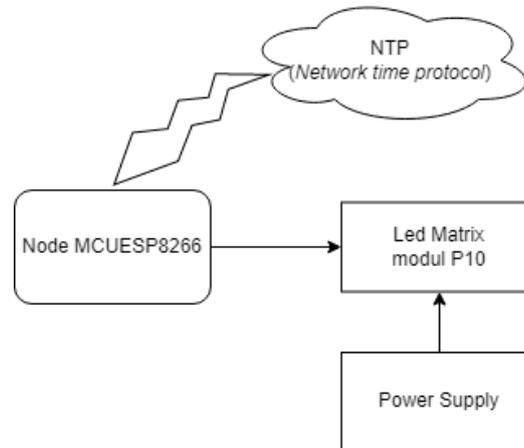
Gambar 3. 9 *database*

Kemudian, dilanjutkan ke *security rules* pilih *start in locked mode* dan klik *enable*. Kemudian tampilan masuk ke *database*. *Database* pun siap di gunakan.

#### 1. NTP (*Network Time Protocol*)

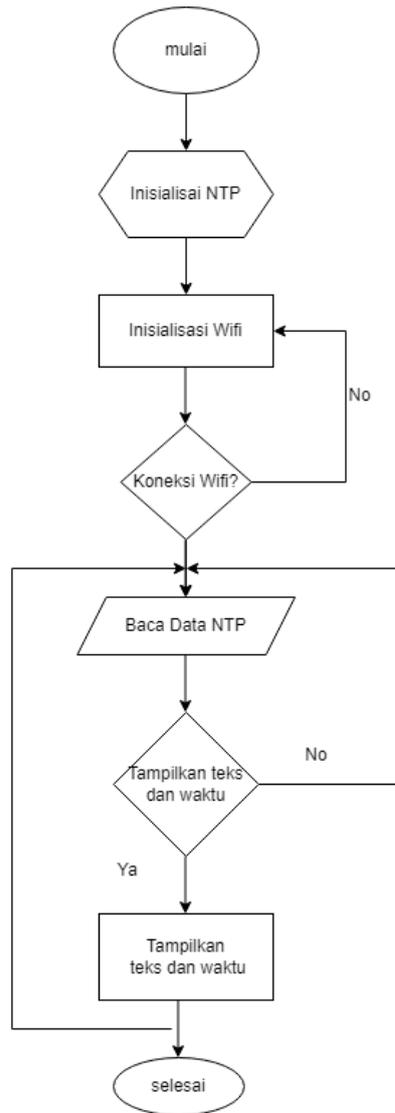
NTP (*Network Time Protocol*) pada penelitian ini berfungsi sebagai sumber data waktu yang akan diambil oleh NodeMCU ESP8266. Dengan catatan setiap komputer yang terhubung dengan jaringan yang stabil. Di dalam jaringan perangkat

pada setiap sekolah terhubung dengan modem atau jaringan internet dengan koneksi yang stabil. Setelah komputer terhubung dengan modem atau jaringan internet NodeMCU ESP8266 akan mengambil data NTP (*Network Time Protocol*) tersebut dan ditampilkan di LED Matrix (Modul P10). Tidak lupa *Display School* membutuhkan arus listrik dengan adanya *power supply* ini alat bisa lebih kuat untuk menampilkan. berikut diagram blok NTP (*Network Time Protocol*).



Gambar 3. 10 Diagram Blok NTP (*network time protocol*)

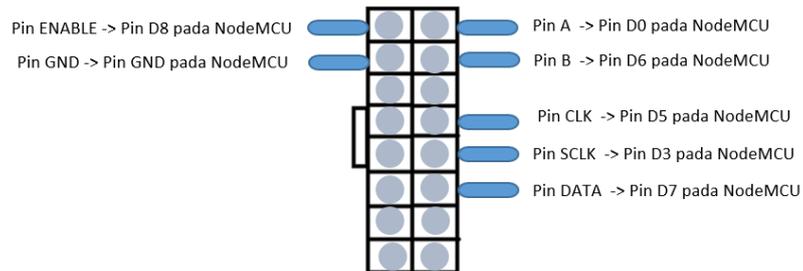
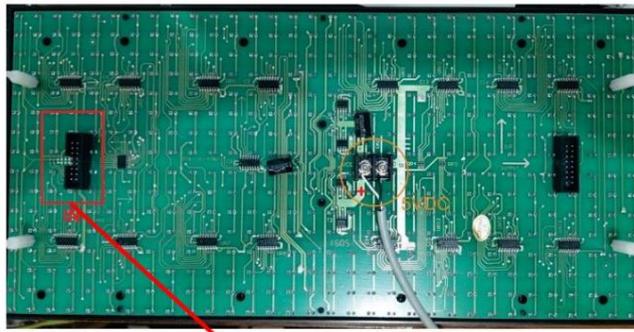
Setelah mengetahui diagram blok tersebut, di dalam diagram blok terdapat proses pengambilan data NTP (*Network Time Protocol*) yang meliputi jam, menit, dan detik yang di baca oleh NodeMCU ESP8266. Di awali dengan ambil data, kemudian Node MCUESP8266 akan meng-inisialisasikan data NTP (*Network Time Protocol*), jika sudah akan langsung meng-inisialisasikan Wifi jika Wifi sudah bisa terkoneksi dengan Node MCUESP8266, dan sudah mendapatkan data waktu dari NTP (*Network Time Protocol*) kemudian langsung tampilkan data NTP (*Network Time Protocol*) dan data pesan yang ada di Firebase. Proses ini akan terus berulang ulang apabila proses ambil data Firebase di lakukan melalui input di aplikasi *website*. Berikut di atas *flowchart* ambil data NTP (*Network Time Protocol*).



Gambar 3. 11 Flowchart ambil data NTP (*Network Time Protocol*)

## 2. LED Matrix (Modul P10)

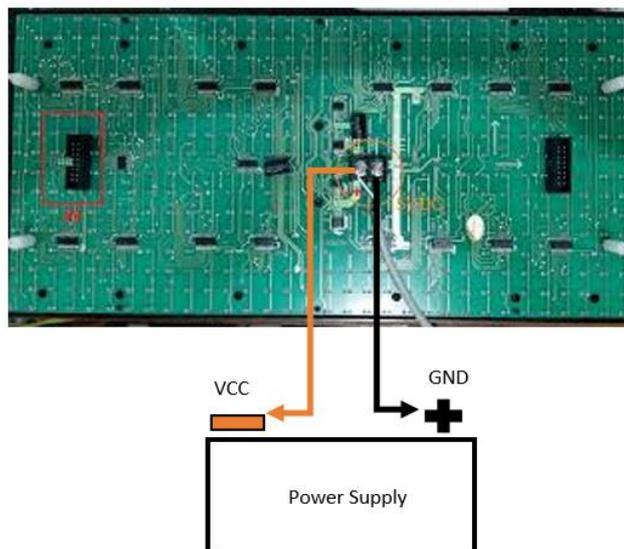
LED Matrix (Modul P10) merupakan komponen yang dapat menampilkan LED dalam jumlah banyak sehingga dapat di atur sesuai keinginan pengguna. LED Matrix (Modul P10) pada sistem ini berfungsi sebagai papan informasi yang menampilkan waktu dan teks yang sedang berlangsung secara *realtime*. LED Matrix (Modul P10) akan menerima perintah dari NodeMCU ESP8266.



Gambar 3. 12 Koneksi Mikrokontroller dengan modul P10

### 3. *Power supply*

*Power supply* dalam sistem ini berfungsi untuk penyedia sumber daya listrik untuk komponen-komponen atau *hardware* pada LED Matrix (Modul P10) dengan arus DC menjadi arus searah. Dengan komponen  $-VCC$  dan  $+GND$  dengan ini arus listrik akan mengalir dengan sistem yang ada untuk bisa menghidupkan alat



Gambar 3. 13 power supply

### 3.3.3 Perancangan Perangkat Lunak A lat

Setelah perancangan perangkat keras maka selanjutnya dilanjutkan dengan perancangan perangkat lunak agar sensor dapat bekerja. Kode ditulis dengan menggunakan Bahasa C, Penulisan Kode dilakukan pada aplikasi Arduino IDE. Kode dibuat untuk dapat mengfungsikan LED Matrix (Modul P10), dan keseluruhan alat agar alat dapat selalu membaca data NTP (*Network Time Protocol*) dan data Firebase secara *realtime*. Sistem perangkat lunak berbentuk aplikasi *website* yang bisa diakses oleh komputer kendali, untuk bisa melakukan *update* teks yang ada di halaman *website* ini perlu memasukan *password* yang sudah dirahasiakan untuk mengurangi hal-hal yang tidak diinginkan. Adapun komputer kendali sebagai admin yang bisa dan tahu isi *password* aplikasi website tersebut.

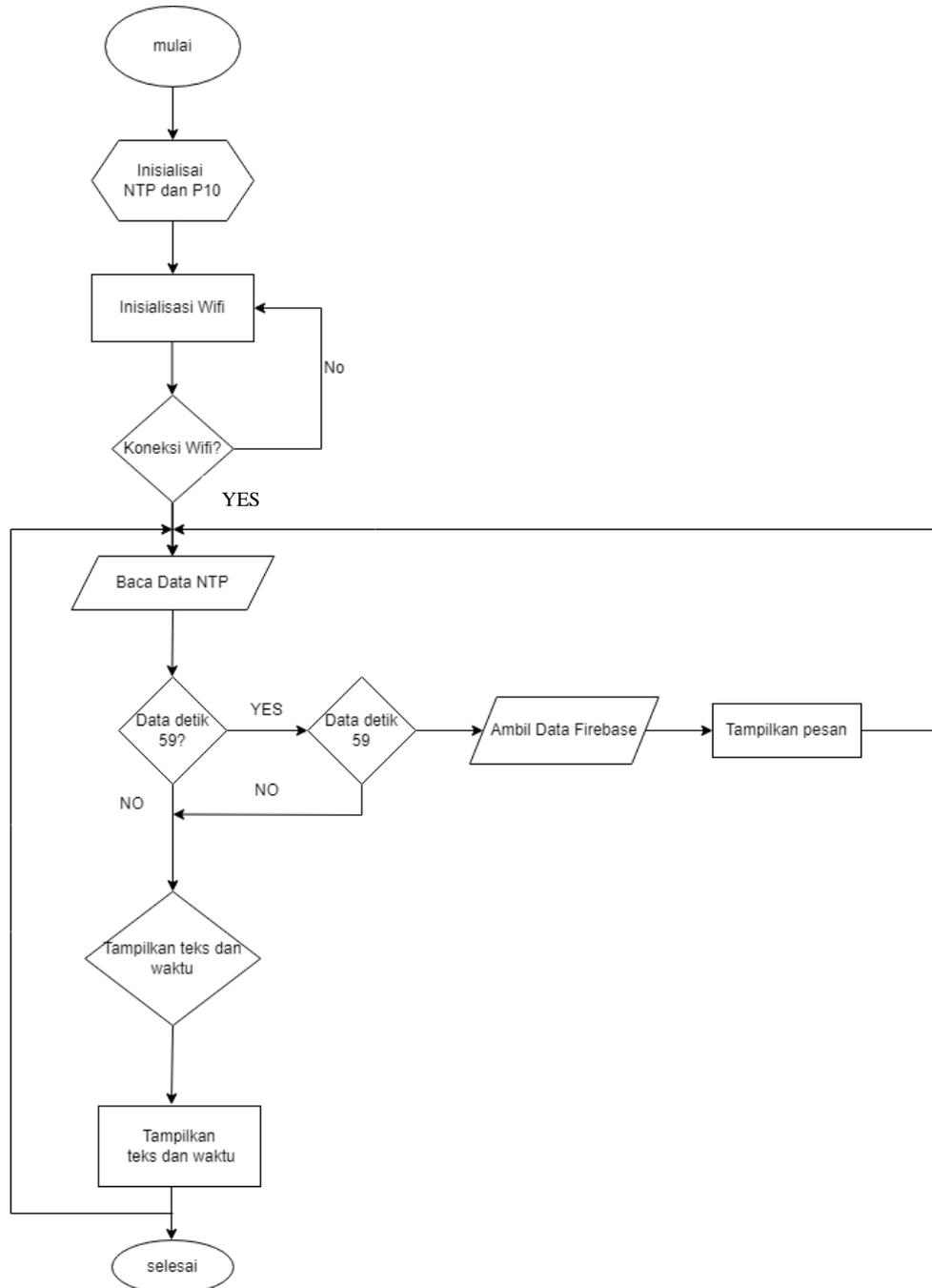
#### 1. Perancangan perangkat lunak pada setiap sekolah

Perancangan perangkat lunak pada sekolah berbentuk aplikasi *website* yang bisa diakses oleh admin atau komputer kendali. Komputer kendali ini yang bisa mengubah teks atau pesan sebagai papan informasi.

Sistem berjalan diawali pada saat arus listrik sudah tersambung seperti yang dilampirkan pada gambar 3.14, kemudian NodeMCU ESP8266 akan menginisialisasikan terlebih dahulu komponen yang akan menjadi *ouput* LED Matrix (Modul P10) disini LED Matrix (Modul P10) sebagai komponen untuk menampilkan hasil *ouput* berupa tampilan teks dan waktu, setelah LED Matrix (Modul P10) ter-inisialisasi, selanjutnya NodeMCU ESP8266 akan menginisialisasi NTP (*Network Time-Protocol*) server dan LED Matrix (Modul P10) selanjutnya NodeMCU ESP8266 akan mengoneksikan Wifi, jika tidak terkoneksi akan dikembali ke pertanyaan awal jika sudah koneksi Wifi maka NodeMCU ESP8266 akan membaca data NTP (*Network Time-Protocol*) jika sudah membaca data NTP (*Network Time-Protocol*), maka langsung menampilkan pada LED Matrix (Modul P10), kemudian ada pertanyaan apakah data waktu sudah menunjukkan detik ke 59, jika sudah menunjukkan detik ke 59 NodeMCU ESP8266 akan langsung mengambil data pesan yang ada di Firebase dan tampilah data pesan. Jika detik tidak menunjukkan detik 59, maka kondisi ini teks akan tetap berjalan dan tidak akan memperbaharui teks atau pesan yang di ubah oleh admin. Teks yang ingin di ubah sesuai kebutuhan (*private*), maka komputer akan menginput teks tersebut akan di kirim ke Firebase, dan jika komputer kendali ingin mengubah teks

semua secara bersamaan (*global*), maka komputer kendali menginput teks yang ingin di ubah di kirim ke Firebase dan akan tampil di P10. Berikut gambar *flowchart* program sistem.

Gambar 3. 14 Diagram *Flowchart* Sistem



## 2. Perancang perangkat lunak pada aplikasi *website* komputer

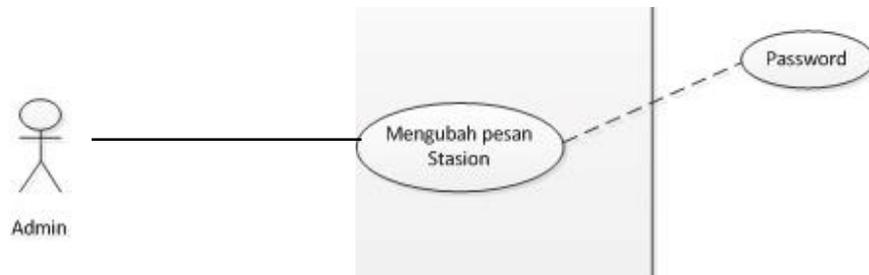
Perancangan perangkat lunak pada aplikasi *website* komputer kendali berbentuk *website* yang bisa di akses oleh admin atau komputer kendali.

Komputer kendali ini yang bisa mengubah teks secara *privat* dan secara *global* admin sendiri tahu *password*. Ada pun *use case*, diagram *sequensial*, diagram *class*, dan *prototype* komputer kendali.

a) Use case diagram

*Use case* diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Hendini, 2016)

Dalam penelitian ini penulis membuat *use case diagram* pada aplikasi *website display school* pada gambar berikut:

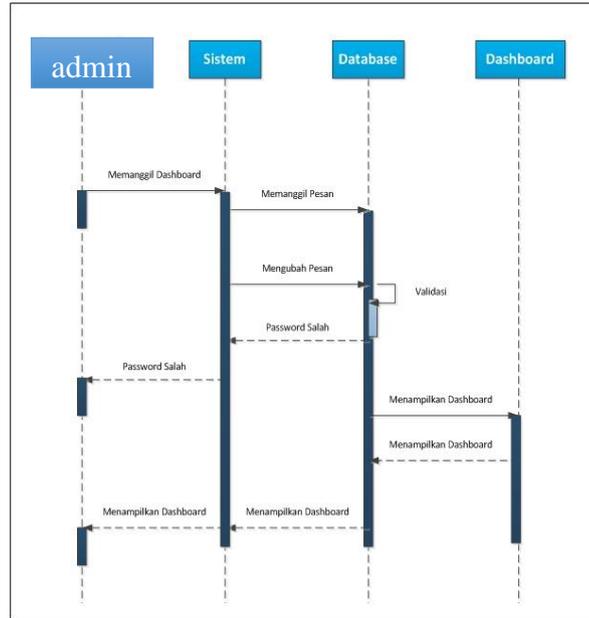


Gambar 3. 15 Use Case Diagram Aplikasi Website display School

b) Sequence Diagram

*Sequence Diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan antar objek. Berikut merupakan *sequence diagram* pada aplikasi *display school* :

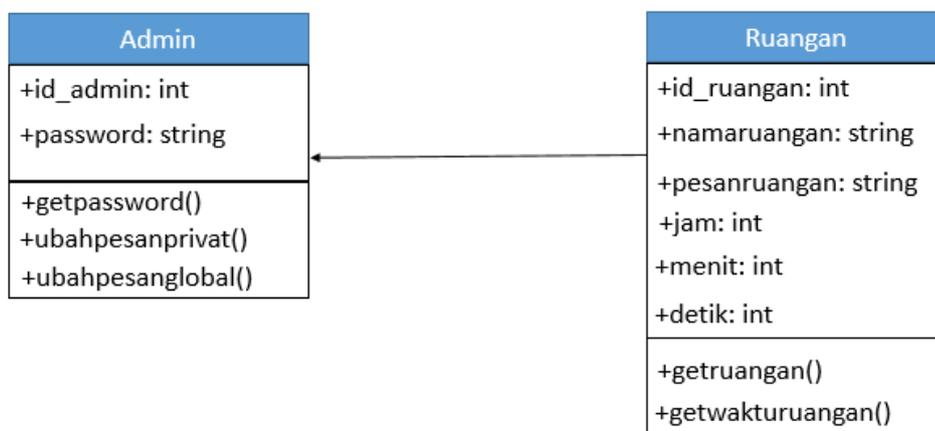
Pada diagram terdapat satu aktor dan empat objek, yaitu admin, sistem, *database*, dan *dashboard*. Pertama-tama admin akan masuk ke tampilan *website* dengan tampilan tombol *send* masih *disable*. Sebelum admin akan mengubah teks di tampilan *website* terdapat kolom untuk mengisi *password* yang akan mehidupkan tombol menjadi *enable* agar bisa merubah teks yang di inginkan. Jika *password* salah akan ada pesan *password* salah, dan jika *password* benar tombol *send* akan langsung *enable* dan admin sudah bisa mengubah teks .



Gambar 3. 16 *Sequence Diagram*

c) *Class diagram*

*Class diagram* merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Berikut untuk *class diagram* yang ada pada penelitian ini :



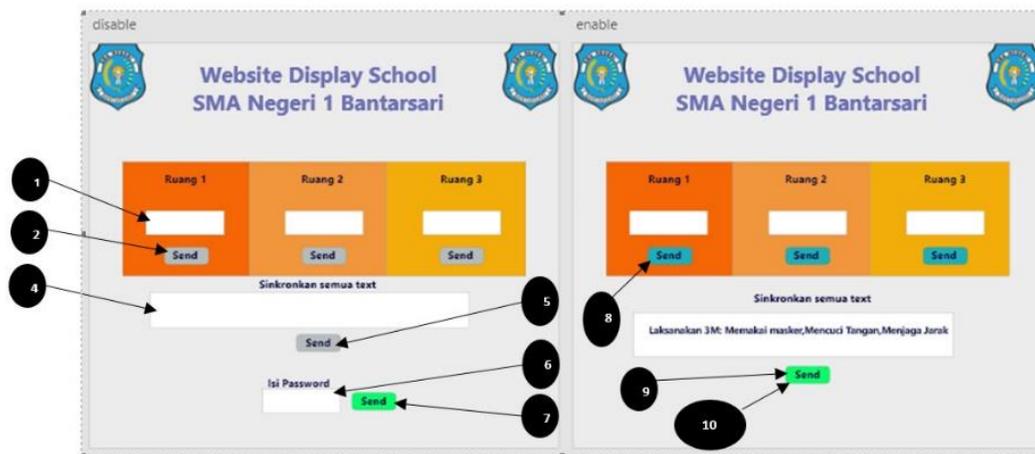
Gambar 3. 17 *class Diagram*

Pada *class diagram* tersebut terdapat dua kelas, yaitu kelas admin dan kelas Ruangan. Kelas admin digunakan untuk menyimpan *password* yang dapat digunakan oleh aktor admin. Sedangkan untuk kelas ruangan

digunakan untuk mengelola data dari setiap ruangan seperti nama, pesan pada papan ruangan dan juga waktu pada ruangan tersebut. Pada dua kelas tersebut memiliki relasi asosiasi yang *multiplicity* nya memiliki arti bahwa satu admin dapat mengelola beberapa ruangan.

d) *Prototype*

*Prototype* ini perancangan antarmuka sistem yang digunakan sebagai landasan pengembangan sistem. *Prototype* dapat memudahkan pengembang untuk mengetahui apa saja fitur dan detail dari setiap komponen yang ada di dalam antarmuka sistem. Maka dengan adanya *prototype* ini, diharapkan dapat lebih memudahkan dalam tahap pengembangan dengan membuat *prototype* yang sudah dilakukan pengkajian dan evaluasi, serta sudah siap di buat berbentuk *website*, *website* ini akan di simpan menjadi aplikasi *website* yang di rahasiakan oleh admin itu sendiri Berikut untuk *prototype* yang ada pada penelitian ini :



Gambar 3. 18 *Prototype* aplikasi *website*

Tabel 3.5 Deskripsi *Prototype*

No	Nama	Deskripsi
1	Label sekolah	Tampilan nama dari sekolah
2	<i>Teks Field</i> Pesan Stasion	Inputan pesan untuk stasion tersebut yang akan tampil di papan sekolah
3	<i>Disable Button</i> Pesan ruangan	<i>Disable Button</i> untuk mengirim pesan pada ruangan tersebut
4	<i>Teks Field</i> Sinkronisasi Semua Pesan	Inputan pesan yang digunakan untuk semua ruangan
5	<i>Disable Button</i> Sinkronisasi Pesan	<i>Disable Button</i> untuk mengirim pesan pada semua ruangan
6	<i>Teks Field Password</i>	Inputan <i>password</i> yang digunakan untuk enable semua <i>button</i>
7	<i>Button Kirim Password</i>	<i>Button</i> untuk mengirim <i>password</i> yang diinputkan
8	<i>Enable Button</i> Pesan ruangan	<i>Enable Button</i> untuk mengirim pesan pada ruangan tersebut
9	<i>Enable Button</i> Sinkronisasi Pesan	<i>Enable Button</i> akan aktif
10	<i>Enable Button</i> Sinkronisasi Pesan	<i>Enable Button</i> untuk mengirim pesan pada semua ruangan

### 3.3.4 Pengujian Alat *Display School*

Pengujian alat dimaksud untuk mengetahui apakah sistem yang telah dibuat sudah sesuai apa yang telah dirancang. Pengujian alat pada penelitian ini menggunakan metode *balck box* yang menguji dari sisi fungsionalitas. Berikut poin-poin yang akan di uji pada penelitian ini:

- a. Pengujian teknis alat.
- b. Pengujian komponen alat.
- c. Pengujian kecepatan respon *update* pesan *running text*.
- d. Pengujian waktu tampil.

### **3.3.5 Pengujian Aplikasi Website**

Pengujian aplikasi *website* dimaksud untuk mengetahui apakah sistem yang telah dibuat sudah sesuai apa yang telah dirancang. Pengujian aplikasi *website* pada penelitian ini menggunakan metode *black box* yang menguji dari sisi fungsionalitas. Berikut poin-poin yang akan di uji pada penelitian ini:

- a. Pengujian fungsionalitas aplikasi *website*.

Pengujian kecepatan mengirim data pesan.