

Sistem Keamanan Pintu dengan Android Menggunakan NODEMCU

Agus Muhaimin¹, Ahmad Bagus Setiawan², Ardi Sanjaya³

^{1,2,3} Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri

E-mail: *¹agusmuhaimin30@gmail.com, ²bagus.este@gmail.com, ³dersky@gmail.com

Abstrak – Saat ini SDN 2 Tapan masih menggunakan kunci pintu yang biasa jadi ketika seorang pegawai lupa tidak mengunci pintu sedangkan pegawainya sudah pulang kerja maka harus kembali ketempat kerja untuk mengunci pintu. Hal tersebut membutuhkan waktu cukup lama karena harus bolak-balik dari rumah ke SDN 2 Tapan. Melihat permasalahan diatas membuat penyusun tertarik untuk membuat sebuah sistem keamanan pintu dengan Telegram menggunakan NodeMCU yang mampu meningkatkan kinerja pegawai SDN 2 Tapan sehingga lebih mudah karena dengan adanya sistem keamanan berbasis Telegram akan menyingkat waktu yang dibutuhkan pegawai dalam bekerja. Sistem keamanan pintu dengan Telegram menggunakan NodeMCU tersebut akan diimplementasikan kedalam pintu kantor di SDN 2 Tapan. Pendekatan dan model penelitian yang digunakan penulis adalah model Waterfall. Dan dari hasil uji coba dengan 7 kali buka dan 7 kali kunci pintu waktu rata-rata delay saat memberi perintah ke alat waktu buka pintu yaitu : 1.72 detik dan 1.68 detik untuk menutup pintu.

Kata Kunci — NodeMCU, SDN 2 Tapan, Sistem Keamanan Pintu

1. PENDAHULUAN

Sistem keamanan adalah suatu sekumpulan elemen atau unsur yang saling berkaitan dan memiliki tujuan yang sama untuk terbebas dari resiko. Sistem keamanan ini dapat dimanfaatkan dalam berbagai hal salah satunya sistem keamanan pintu. Sistem keamanan pintu dengan Android adalah sebuah alat yang digunakan untuk mengamankan pintu dengan menggunakan NodeMCU yang bisa diakses lewat Telegram. Sekolah Dasar Negeri 2 Tapan Tulungagung atau biasa disebut SDN 2 Tapan merupakan sekolah tingkat dasar yang berbasis pada pembelajaran sekolah dasar. SDN 2 Tapan terletak di Jalan Jayeng Kusuma Nomor 14 Tulungagung.

Sistem keamanan sangat penting bagi setiap instansi pendidikan. Karena dapat mengamankan uang sekolah, dokumen sekolah, peralatan sekolah. Saat ini SDN 2 Tapan masih menggunakan kunci pintu yang biasa jadi ketika seorang pegawai lupa tidak mengunci pintu sedangkan pegawainya sudah pulang kerja maka harus kembali ketempat kerja untuk mengunci pintu. Hal tersebut membutuhkan waktu cukup lama karena harus bolak-balik dari rumah ke SDN 2 Tapan. Melihat permasalahan diatas membuat penyusun tertarik untuk membuat sebuah sistem keamanan pintu dengan Android menggunakan NodeMCU yang mampu meningkatkan kinerja pegawai SDN 2 Tapan sehingga lebih mudah karena dengan adanya sistem keamanan pintu dengan Android akan menyingkat waktu yang dibutuhkan pegawai dalam bekerja. Sistem keamanan pintu dengan Android

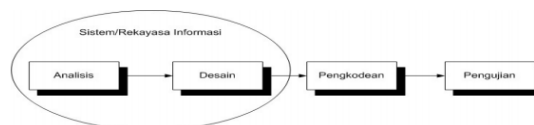
menggunakan NodeMCU tersebut akan diimplementasikan kedalam pintu di SDN 2 Tapan.

Dari hasil penelitian sebelumnya dengan judul “Rancang Bangun Sistem Otomasi Dan Keamanan Rumah Pintar Menggunakan Raspberry Pi 3 Dengan Pusat Kendali Telegram” Dapat menjalankan pintu otomatis dengan memberi perintah melalui aplikasi telegram. Waktu rata-rata delay saat memberi perintah ke alat yaitu : 1,783 detik untuk pintu otomatis [1].

2. METODE PENELITIAN

2.1. Metode Penelitian

Pendekatan dan model penelitian yang digunakan penulis adalah model waterfall. Disebut dengan waterfall karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan [2].



Gambar 1. Metode Waterfall

1) Analisis kebutuhan (*analyzing*)

Setelah melakukan pengamatan kemudian dilakukan analisa kebutuhan meliputi fungsional dan non fungsional. Fungsional adalah sistem dapat membuka dan mengunci pintu dengan Telegram. Non fungsional adalah peralatan yang diperlukan seperti laptop, NodeMCU, Arduino ide.

2) Desain (*design*)

Dari hasil analisa kebutuhan fungsional dan non fungsional dilakukan perancangan sistem keamanan yang mau dibangun.

3) Pembuatan kode (*coding*)

Setelah perancangan selesai selanjutnya melakukan coding dengan menggunakan Arduino IDE.

4) Pengujian sistem (*testing*)

Pengujian dilakukan dengan menjalankan sistem keamanan apakah berfungsi dengan baik atau tidak.

5) Implementasi (*implementation*)

Implementasi adalah tahap pemasangan sistem keamanan..

6) Perawatan (*maintenance*)

Tahap ini perlu dilakukan untuk terus menjaga sistem keamanan berfungsi dan tidak ada masalah yang timbul.

2.2. Teori Penunjang

1) NodeMCU esp8266



Gambar 2. NodeMCU

NodeMCU ESP8266 adalah chip terintegrasi yang dirancang untuk menghubungkan mikrokontroler dengan internet melalui Wi-Fi. Ia menawarkan solusi jaringan Wi-Fi yang lengkap dan mandiri, yang memungkinkan untuk menjadi host ataupun sebagai Wi-Fi client [3].

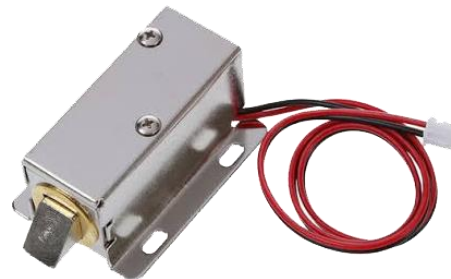
2) Relay



Gambar 3. Relay

Relay adalah Saklar (Switch) yang dioperasikan secara listrik dan merupakan komponen Electromechanical (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (Coil) dan Mekanikal (seperangkat Kontak Saklar/Switch).[3]

3) Solenoid Door Lock



Gambar 4. Solenoid Door Lock

Solenoid Door Lock atau Solenoid Kunci Pintu adalah alat elektronik yang dibuat khusus untuk pengunci pintu. Alat ini sering digunakan pada kunci pintu otomatis. Solenoid ini akan bergerak/bekerja apabila diberi tegangan. Tegangan solenoid kunci pintu ini rata-rata yang di jual dipasaran 12 volt tapi ada juga yang 6 volt dan 24 volt. [4]

4) Tampilan Alat



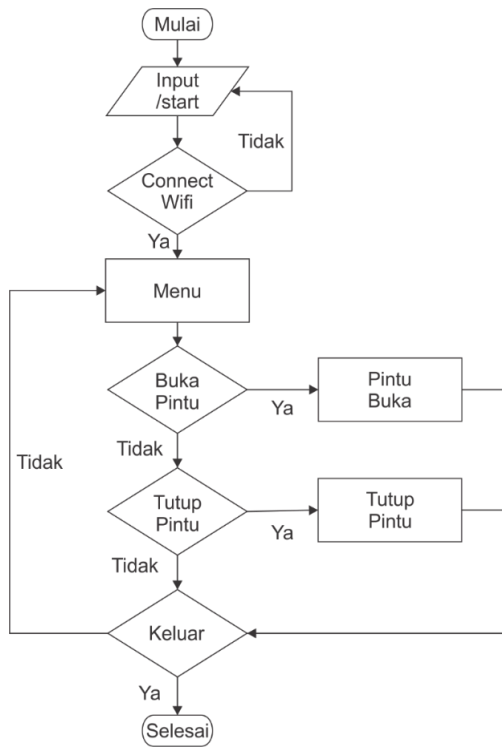
Gambar 5. Tampilan Alat

3. HASIL DAN PEMBAHASAN

3.1. Desain Sistem

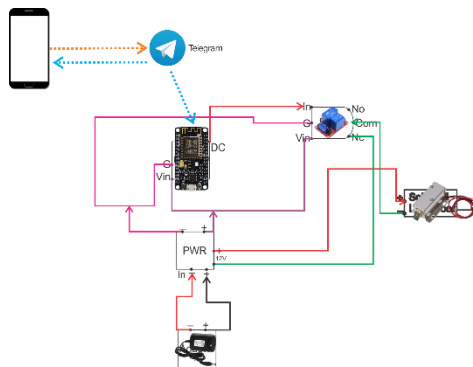
1) Flowchart Sistem

NodeMCU dihidupkan agar tersambung ke Wi-fi yang terhubung di internet. Lalu buka Aplikasi telegram di smartphone klik /start jika Berhasil akan dapat balasan dari bot telegram berupa menu bukapintu dan tutuppintu jika ingin buka pintu ketuk menu bukapintu jika Berhasil akan dapat balas dari bot telegram berupa pesan pintu terbuka jika ingin mengunci pintu ketuk menu tutuppintu jika Berhasil akan dapat balasan dari bot telegram berupa pesan tutup pintu lalu ingin keluar tutup Aplikasi telegramnya selesai. Alur system ditunjukkan pada diagram flowchart gambar 6.



Gambar 6. Flowchart Sistem

2) Perancangan Alat



Gambar 7. Perancangan Alat

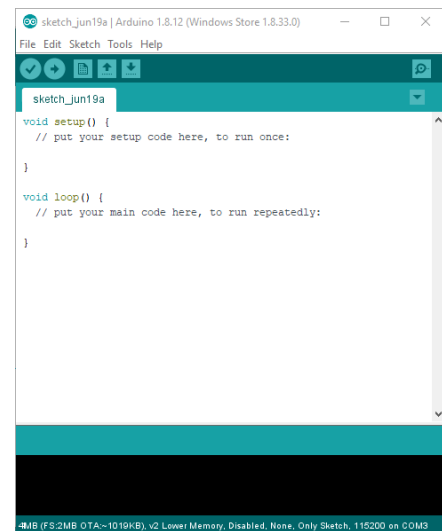
Diskripsi Perancangan Alat

- Smartphone mengirim perintah ke telegrambot, telegrambot memberi perintah ke NodeMCU, telegrambot mengirim balasan ke smartphone.
- Koneksi Modul Relay dengan kabel jumper warna merah dari Node MCU ke pin Modul Relay In, Koneksi Modul Relay dengan kabel jumper warna Pink dari Node MCU ke pin Modul Relay G. Koneksi Modul Relay dengan kabel jumper warna Ungu dari Node MCU ke pin Modul Relay VIn, Koneksi Modul Relay dengan kabel jumper warna hijau dari Board PWR ke pin Modul Relay Nc.
- Koneksi NodeMCU dengan kabel jumper warna ungu dari Board PWR ke pin

- NodeMCU Vin, Koneksi NodeMCU dengan kabel jumper warna Pink dari Board PWR ke pin NodeMCU G.
- Koneksi Selenoid *Lock Door* dengan kabel warna merah dari Board PWR ke pin Selenoid *Lock Door* (+), Koneksi Selenoid *Lock Door* dengan kabel warna hijau dari Modul Relay ke pin Selenoid *Lock Door* (-)
- Koneksi Board PWR dengan kabel warna merah dari Power Listrik ke Board PWR (-), Koneksi Board PWR dengan kabel warna hitam dari Power Listrik ke Board PWR (+).

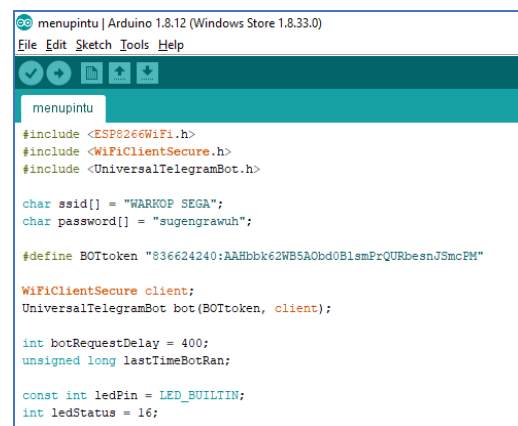
3) Perancangan Program NodeMCU

Pertama, melakukan instalasi arduino IDE pada perangkat PC atau laptop dengan OS *windows*.



Gambar 8. Home Arduino IDE

Dalam arduino terdapat 2 perintah utama yaitu void setup dan void loop. Void setup merupakan perintah awal yang berjalan hanya satu kali ketika mikrokontroler dinyalakan. Void loop merupakan perintah yang berjalan terus menerus atau berulang-ulang hingga daya pada mikrokontroler hilang.



Gambar 9. Program 1

Include merupakan perintah untuk memasukan *library* alat yang kita gunakan. Dalam alat ini, kita menggunakan *library* ESP8266 WiFi.h, WiFi Client Secure.h, Universal Telegram Both Sedangkan *doorlock* tidak perlu menggunakan *library* karena hanya menggunakan perintah *low* (mati) dan *high* (hidup). Sebelum menulis program, kita terlebih dahulu mendeklarasikan variable yang digunakan.

Char ssid Nama *Wifi* yang akan disambungkan, *Char password* adalah *password* dari *wifi* tersebut, *#define BOTtoken* adalah *token* dari *Bot telegram*, *int ledStatus = 16* adalah pin ke 16 dari NodeMCU.

```
void handleNewMessages(int numNewMessages) {
  Serial.println("handleNewMessages");
  Serial.println(String(numNewMessages));

  for (int i=0; i<numNewMessages; i++) {
    String chat_id = String(bot.messages[i].chat_id);
    String text = bot.messages[i].text;

    String from_name = bot.messages[i].from_name;
    if (from_name == "") from_name = "Guest";
  }
}
```

Gambar 10. Program 2

Pada *void handleNewMessages* untuk menyatukan pesan

```
for (int i=0; i<numNewMessages; i++) {
  String chat_id = String(bot.messages[i].chat_id);
  String text = bot.messages[i].text;
  String from_name = bot.messages[i].from_name;
  if (from_name == "") from_name = "Guest";
}
```

koding diatas berfungsi untuk mengirim pesan ke telegram

```
String from_name = bot.messages[i].from_name;
if (from_name == "") from_name = "Guest";

if (text == "/bukapintu") {
  digitalWrite(ledPin, LOW);
  ledStatus = 1;
  bot.sendMessage(chat_id, "Pintu Terbuka", "");
} else if (text == "/tutuppintu") {
  ledStatus = 0;
  digitalWrite(ledPin, HIGH);
  bot.sendMessage(chat_id, "Pintu Tertutup", "");
} else if (text == "/status") {
  if (ledStatus) {
    bot.sendMessage(chat_id, "Pintu sedang Terbuka", "");
  } else {
    bot.sendMessage(chat_id, "Pintu sedang Tertutup", "");
  }
} else if (text == "/start") {
  String welcome = "Selamat datang di Telegram Bot, " + from_name + ".\n";
  welcome += "Silahkan ketuk salah satu menu dibawah\n\n";
  welcome += "/start : Untuk memunculkan Menu.\n";
  welcome += "/bukapintu : Untuk membuka Pintu.\n";
  welcome += "/tutuppintu : Untuk menutup Pintu.\n";
  welcome += "/status : Cari tahu pintu sedang terbuka atau tertutup.\n";
  bot.sendMessage(chat_id, welcome, "Markdown");
}
```

Gambar 11. Program 3

```
if (text == "/bukapintu") {
  digitalWrite(ledPin, LOW);
}
```

```
ledStatus = 1;
bot.sendMessage(chat_id, "Pintu Terbuka", "");
}
```

Perintah ini digunakan untuk membuka Kunci pintu, jadi Ketika kita mengetuk menu /bukapintu maka *doorlock* akan membuka dan akan memberi balasan bahwa Pintu terbuka

```
else if (text == "/tutuppintu") {
  ledStatus = 0;
  digitalWrite(ledPin, HIGH);
  bot.sendMessage(chat_id, "Pintu Tertutup", "");
}
```

Perintah ini digunakan untuk mengunci pintu, jadi Ketika kita mengetuk menu /tutuppintu maka *doorlock* akan mengunci dan akan memberi balasan bahwa Pintu tertutup.

```
else if (text == "/status") {
  if (ledStatus) {
    bot.sendMessage(chat_id, "Pintu sedang Terbuka", "");
  } else {
    bot.sendMessage(chat_id, "Pintu sedang Tertutup", "");
  }
}
```

Perintah ini digunakan untuk mengecek apakah pintu sudah Ditutup atau dikunci/terbuka. Jadi Ketika mengetuk menu /status maka akan dapat balasan bahwa pintu terbuka / tertutup.

```
else if (text == "/start") {
  String welcome = "Selamat datang di Telegram Bot, " + from_name + ".\n";
  welcome += "Silahkan ketuk salah satu menu dibawah\n\n";
  welcome += "/start : Untuk memunculkan Menu.\n";
  welcome += "/bukapintu : Untuk membuka Pintu.\n";
  welcome += "/tutuppintu : Untuk menutup Pintu.\n";
  welcome += "/status : Cari tahu pintu sedang terbuka atau tertutup.\n";
  bot.sendMessage(chat_id, welcome, "Markdown");
}
```

Perintah ini akan muncul ketika kita pertama kali menjalankan telegram bot di smartpone.

```

menupintu | Arduino 1.8.12 (Windows Store 1.8.33.0)
File Edit Sketch Tools Help

menupintu

void setup() {
  Serial.begin(115200);
  client.setInsecure();

  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  delay(100);

  Serial.print("Connecting Wifi: ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(300);
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());

  pinMode(ledPin, OUTPUT);
  delay(10);
  digitalWrite(ledPin, HIGH);
}
    
```

Gambar 12. Program 4

Koding ini berfungsi sebagai mengoneksikan ke *Wi-Fi* dan menunjukkan ke serial monitor bahwa koneksi *Wi-Fi* telah terhubung dan menunjukkan *Ip Address*

```

void loop() {
  if (millis() > lastTimeBotRan + botRequestDelay) {
    int numNewMessages = bot.getUpdates(bot.last_message_received + 1);

    while(numNewMessages) {
      Serial.println("got response");

      handleNewMessages(numNewMessages);
      numNewMessages = bot.getUpdates(bot.last_message_received + 1);
    }

    lastTimeBotRan = millis();
  }
}
    
```

Gambar 13. Program 5

Koding ini berfungsi untuk mengecek apakah telegrambot sudah terhubung dengan nodeMCU di serial monitor IDE Arduino.

3.2. Hasil

Pengujian Sistem Pintu

Setelah mengupload kode program ke NodeMCU dan berhasil check di Serial Monitor di IDE Arduino muncul tulisan *WiFi connected* dan keluar *IP Address* nya, lalu buka Aplikasi telegram lalu ketik “ /start ” jika berhasil keluar menu buka dan Kunci pintu setelah itu ketuk menu “ /bukapintu “ untuk membuka pintu dari situ kita hitung berapa waktu yang dibutuhkan sampai telegrambot mengirim balasan ke perangkat smartphone.

Dari 7 kali pengujian diperoleh waktu rata-rata satu kali *door lock* membuka adalah 1.72

Tabel 1. Satu kali buka

Percobaan	Waktu Membuka (detik)
1	1.6
2	1.68
3	1.7
4	1.8
5	1.68
6	1.9
7	1.7

Dari 7 kali pengujian diperoleh waktu rata-rata satu kali *door lock* menutup adalah 1.68

Tabel 2. Satu kali tutup

Percobaan	Waktu Menutup (detik)
1	1.8
2	1.7
3	1.9
4	1.6
5	1.5
6	1.57
7	1.7

Dari data-data pada tabel diatas hanya 2 kali percobaan yang sesuai dengan yang diharapkan yaitu 1.5 detik untuk waktu membuka atau menutup pintu sementara sisanya ada yang kurang atau lebih dari 1.5 detik tapi mendekati. mungkin ada kesalahan saat penghitungan waktu dan kecepatan internet yang membuat selisih waktu menjadi tidak 1.5 detik..

4. SIMPULAN

Berdasarkan hasil pengujian dan analisa dapat disimpulkan bahwa

- 1) Tingkat keberhasilan alat dan aplikasi telegram berinteraksi dan saling terintegrasi dengan baik yaitu 100% karena semua perintah dapat dijalankan dengan menggunakan NodeMCU sebagai pusat dan aplikasi telegram sebagai pusat kendali.
- 2) Dapat menjalankan *Lock Door* dengan memberi perintah melalui aplikasi telegram. Waktu rata-rata delay saat memberi perintah ke alat waktu buka pintu yaitu : 1.72 detik dan 1.68 detik untuk menutup pintu.
- 3) Jarak tidak mempengaruhi perintah yang diberikan oleh user ke *Lock Door*. Dengan demikian user dapat menjalankan buka atau kunci pintu selama user dan *Lock Door* terhubung dengan internet

5. SARAN

Berdasarkan pengujian yang telah dilakukan pada tugas akhir kali ini, terdapat beberapa saran yang dapat membantu untuk menjadi panutan untuk penelitian berikutnya.

- 1) Menguji coba sistem dengan menggunakan sensor untuk mendeteksi terbukanya pintu selain limit switch seperti sensor ultrasonik, sensor magnet, sensor gerak, dan lain sebagainya.
- 2) Mengaplikasikan sistem pada jenis pintu lain seperti pintu geser, rolling door, atau diaplikasikan pada jendela yang terdapat dirumah.

DAFTAR PUSTAKA

- [1] Simbolon, Henric Sahala Teofilus Dkk 2018. *Rancang Bangun Sistem Otomasi Dan Keamanan Rumah Pintar Menggunakan Raspberry Pi 3 Dengan Pusat Kendali Telegram*. Bandung.
- [2] Hidayati, Nur. 2019. *Penggunaan Metode Waterfall Dalam Rancang Bangun Sistem Informasi Penjualan*. Kediri
- [3] Hidayat, M Reza. Dkk 2018. *Perancangan Sistem Keamanan Rumah Berbasis Iot Dengan Nodemcu Esp8266 Menggunakan Sensor Pir Hc-Sr501 Dan Sensor Smoke Detector*.
- [4] Iskandar, Akbar. Dkk 2017. *Sistem Keamanan Pintu Berbasis Arduino Mega*. Makassar.