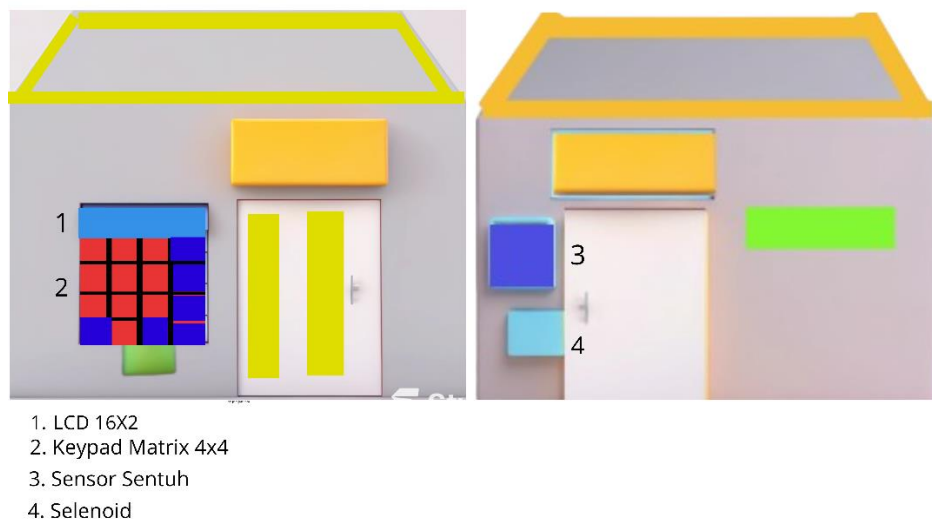


BAB IV

HASIL DAN PEMBAHASAN

Penelitian ini telah menghasilkan purwarupa kunci pintar. Kunci pintar tersebut telah di aplikasikan dilakukakn uji coba pada purwarupa yang telah dibangun berdasarkan skala kecil untuk dilakukan uji coba fungsi dari masing-masing perangkat-perangkat IoT untuk pengembangan sistem. Berikut adalah hasil perancangan layout purwarupa sebelum dilakukan pembuatan purwarupa asli.



Gambar 4.1 Layout Purwarupa

Pada Layout diatas perangkat-perangkat mempunyai alasan setiap penempatan semua komponen. Berikut alasan penempatan komponennya sebagai berikut :

1. Keypad berfungsi untuk menginput kata kunci, penempatan keypad ini diletakkan dikiri dekat dengan gagang pintu, dengan tujuan untuk memudahkan pengguna dalam memasukkan pin untuk masuk kerumah si pengguna.
2. Lcd berfungsi untuk melihat password, penempatan Lcd persis berada di atas keypad, dengan tujuan memberikan informasi apakah kata kunci yang dimasukkan sudah benar atau salah.
3. Sensor sentuh berfungsi sebagai sensor pendeteksi sentuhan, penempatan sensor sentuh tepat berada di dekat dengan gagang pintu bagian dalam rumah, dengan tujuan memebrikan kemudahan akses bagi si pengguna ketika mau keluar dari dalam rumah.
4. Selenoid berfungsi sebagai mekanisme pintu, penempatan solenoid tepat berada di dalam rumah dibagian bawahnya sensor, dengan tujuan untuk memastikan kinerja selenoid berjalan dengan lancar.

Setelah dilakukan perancangan layout maka hasil pembuatan purwarupa aslinya terlihat pada gambar berikut ini.

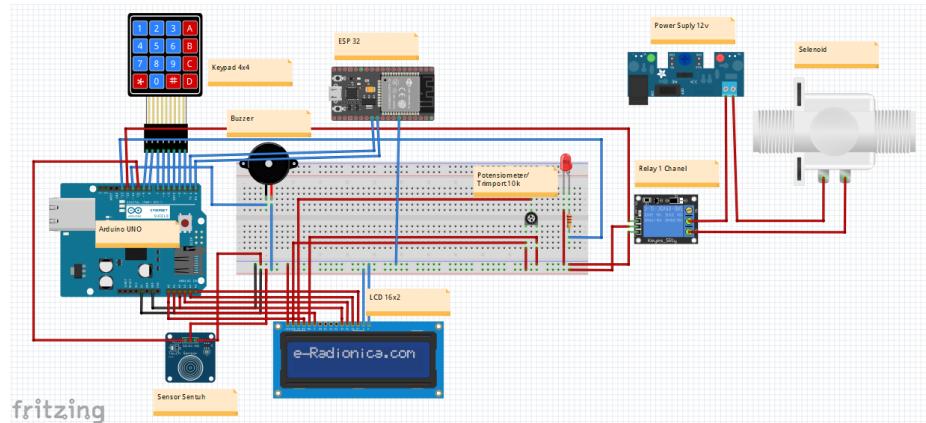


Gambar 4.2 purwarupa Smart Home

4.1. Hasil Perancangan Sistem

4.1.1. Hasil Perancangan Sistem Perangkat Keras

Perancangan Sistem yang digunakan dalam membangun sistem keamanan pintu menggunakan sensor sentuh dan verifikasi kata kunci berbasis IoT pada gambar 4.2 menjelaskan tentang perancangan sistem yang telah dibuat. Sensor sentuh yang diintegrasikan pada mikrokontroler Arduino akan membaca setiap akses yang dilakukan pada pintu. Sistem ini mengandalkan beberapa komponen seperti sensor sentuh, keypad untuk input kata kunci, dan solenoid untuk mekanisme kunci. Setelah pengguna menyentuh sensor dan memasukkan kata kunci yang benar, solenoid akan diaktifkan untuk membuka kunci pintu. Berikut adalah hasil perancangan sistem perangkat keras yang dirancang menggunakan fritzing seperti gambar dibawah ini.



Gambar 4.3 Skema Perancangan Perangkat keras

Pada perancangan skema perangkat keras diatas dapat dijelaskan fungsi dari masing-masing perangkat yang terdiri dari:

1. Arduino Uno, sebagai mikrokontroler sistemnya atau sebagai pemroses data dari sensor dan pengendalian perangkat seperti solenoid. Arduino Uno dipilih sebagai mikrokontroler memiliki keunggulan yang membuatnya ideal untuk berbagai proyek elektronik, termasuk pemrosesan data dari sensor dan pengendalian perangkat seperti solenoid. Salah satu keunggulan Arduino Uno adalah kemudahan penggunaannya dan ketersediaan banyak sumber daya serta komunitas yang mendukung, sehingga memudahkan dalam pengembangan dan troubleshooting proyek.
2. Sensor Sentuh, digunakan sebagai sensor untuk membuka pintu dari dalam rumah karena sangat mudah digunakan. Pengguna hanya perlu menyentuh permukaan sensor untuk mengaktifkannya, yang membuatnya sangat intuitif dan user-

friendly, terutama bagi semua penghuni rumah, termasuk anak-anak dan orang tua.

3. Keypad, digunakan sebagai tempat untuk menekan tombol dan memasukkan kata kunci dari luar pintu rumah dan keypad yang digunakan ukuran 4x4. Keypad memungkinkan penggunaan kombinasi angka yang panjang dan kompleks sebagai kata kunci, meningkatkan tingkat keamanan dibandingkan dengan metode akses yang lebih sederhana. Selain itu, keypad menawarkan kemudahan penggunaan dan fleksibilitas dalam pengaturan kode akses. Pengguna dapat dengan mudah mengubah kata kunci sesuai kebutuhan tanpa perlu mengganti perangkat keras.
4. Selenoid, digunakan sebagai mekanisme pintu yang terbuka secara otomatis ketika kata kunci yang benar dimasukkan dan sensor disentuh. Selenoid memungkinkan pintu terbuka secara otomatis, memberikan kenyamanan dan kemudahan bagi pengguna. Selain itu, selenoid adalah perangkat yang handal dan tahan lama, mampu beroperasi dengan efisien untuk jangka waktu yang lama tanpa memerlukan banyak perawatan.
5. LCD (Liquid Crystal Display), digunakan untuk menampilkan luaran sensor sentuh dan keypad. Dan yang di pakai adalah LCD 16x2 dipilih karena dapat menampilkan hingga 32 karakter (16 karakter per baris), yang cukup untuk memberikan

informasi penting seperti status sistem, instruksi, atau pesan kesalahan. Selain itu, LCD 16x2 mudah diintegrasikan dengan mikrokontroler seperti Arduino, dengan banyak library dan contoh kode yang memudahkan pengaturan dan pemrograman layar.

6. Relay, digunakan untuk menghidupkan dan mematikan selenoid, atau sebagai pengendali high dan low selenoid. Relay dipilih karena menyediakan isolasi elektrik yang penting antara rangkaian kontrol (misalnya, mikrokontroler) dan rangkaian yang dikendalikan (selenoid). Ini penting untuk melindungi mikrokontroler dari lonjakan tegangan atau arus tinggi yang bisa merusaknya. Selain itu, relay mampu mengendalikan perangkat yang membutuhkan arus tinggi, seperti selenoid. Mikrokontroler biasanya tidak dapat langsung memasok arus yang cukup untuk selenoid, sehingga relay menjadi perantara yang ideal.
7. Buzzer, digunakan untuk memberikan bunyi ketika pintu akan tertutup. Buzzer dipilih karena memberikan indikasi suara yang jelas, memastikan pengguna menyadari bahwa pintu sedang atau akan tertutup. Ini penting untuk keselamatan dan kenyamanan, terutama dalam situasi di mana pintu mungkin tidak terlihat langsung oleh pengguna. Dengan adanya buzzer, orang di sekitar pintu mendapatkan peringatan dini bahwa pintu akan bergerak, membantu mencegah cedera atau

kecelakaan yang mungkin terjadi akibat penutupan pintu yang tiba-tiba.

8. LED, menyala bersamaan dengan buzzer ketika pintu akan tertutup. LED dipilih karena memberikan indikasi visual yang jelas bahwa pintu akan tertutup. Hal ini sangat membantu bagi pengguna yang mungkin tidak mendengar bunyi buzzer, seperti mereka yang memiliki gangguan pendengaran. Kombinasi LED dan buzzer memberikan peringatan ganda, baik secara visual maupun audio, sehingga memastikan semua orang di sekitar pintu mendapatkan peringatan yang cukup sebelum pintu bergerak. Ini membantu mencegah cedera atau kecelakaan.
9. Potensiometer, digunakan untuk mengatur kontras LCD 16x2. Potensiometer dipilih karena memungkinkan pengguna untuk menyesuaikan tingkat kontras pada layar LCD 16x2 sesuai dengan preferensi mereka. Hal ini memungkinkan tampilan teks dan grafis menjadi lebih jelas dan mudah dibaca. Selain itu, potensiometer dapat diatur ulang sesuai kebutuhan, memungkinkan pengguna untuk mengatur kontras LCD secara real-time sesuai dengan kondisi pencahayaan lingkungan atau perubahan preferensi pengguna.
10. Power supply, digunakan untuk menggerakkan solenoid. Power supply yang digunakan mempunyai tegangan 12V dipilih karena solenoid umumnya memerlukan tegangan

operasi yang cukup tinggi untuk berfungsi dengan baik. Dengan menggunakan power supply 12V, pastikan bahwa solenoid mendapatkan tegangan yang sesuai untuk operasi yang optimal.

11. ESP32, adalah mikrokontroler yang sangat kuat dan serbaguna, sering digunakan sebagai modul Wifi untuk memperluas kemampuan jaringan dari perangkat lain seperti Arduino Uno. Menggunakan ESP32 sebagai modul Wifi untuk Arduino Uno memungkinkan proyek Arduino Anda untuk terhubung ke internet atau jaringan lokal. ESP32 memiliki kemampuan pemrosesan yang lebih tinggi dibandingkan dengan mikrokontroler tradisional dan dilengkapi dengan berbagai fitur seperti Bluetooth, Wifi, dan banyak GPIO, yang membuatnya ideal untuk berbagai aplikasi IoT (Internet of Things). Dengan menggunakan ESP32, Anda dapat mengembangkan proyek yang lebih kompleks, termasuk pengendalian jarak jauh, otomatisasi rumah, dan sistem pengumpulan data, semuanya dengan kemampuan untuk mengakses dan mengontrol perangkat dari mana saja.

12. Telegram, Menggunakan Telegram untuk notifikasi dari ESP32 memungkinkan Anda untuk menerima pesan real-time langsung di perangkat Anda. Ini mempermudah monitoring jarak jauh dan meningkatkan responsivitas sistem dengan memberikan pembaruan status dan notifikasi penting secara

otomatis. Selain itu, notifikasi Telegram dapat diakses dari berbagai perangkat seperti ponsel, tablet, dan komputer, memberikan fleksibilitas tinggi dalam memantau kondisi sistem di mana saja dan kapan saja. Sistem ini juga dapat dikonfigurasi untuk mengirim notifikasi berdasarkan berbagai kondisi dan parameter, memastikan bahwa informasi yang diterima selalu relevan dan up-to-date.

Setelah menjelaskan tentang peran setiap perangkat keras pada perancangan sistem perangkat keras selanjutnya akan dijelaskan perancangan pin komponen. Ini merupakan tahapan proses menyusun atau menyambungkan antar pin komponen (sensor) satu dengan yang lainnya. Perancangan pin komponen yang digunakan dalam bentuk tabel, sebagai berikut:

Tabel 4.1 Rangkaian Keypad

Keypad	Arduino Uno
Pin C4	Pin 2
Pin C3	Pin 3
Pin C2	Pin 4
Pin C2	Pin 5
Pin R4	Pin 6
Pin R3	Pin 7
Pin R2	Pin 8
Pin R1	Pin 9

Keteranganya :

1. Pin C4 Keypad dihubungkan ke Pin 2 Arduino Uno
2. Pin C3 Keypad dihubungkan ke Pin 3 Arduino Uno

3. Pin C2 Keypad dihubungkan ke Pin 4 Arduino Uno
4. Pin C1 Keypad dihubungkan ke Pin 5 Arduino Uno
5. Pin R4 Keypad dihubungkan ke pin 6 Arduino Uno
6. Pin R3 Keypad dihubungkan ke pin 7 Arduino Uno
7. Pin R2 Keypad dihubungkan ke pin 8 Arduino Uno
8. Pin R1 Keypad dihubungkan ke pin 9 Arduino Uno

Table 4.2 Rangkaian Sensor Sentuh

Sensor Sentuh	Arduino Uno
Pin IO (Input/Luaran)	Pin 10
Pin Gnd (-)	Pin Gnd (-)
Pin Vcc (+)	Pin 5v (+)

Keteranganya :

1. Pin IO (Input/Luaran) sensor sentuh dihubungkan ke Pin 10 Arduino Uno
2. Pin Gnd (-) sensor sentuh dihubungkan ke Pin Gnd (-) Arduino Uno
3. Pin Vcc(+) sensor sentuh dihubungkan ke Pin 5v (+) Arduino Uno

Tabel 4.3 Rangkaian Relay

Relay	Arduino Uno
Pin IN (Input)	Pin 12
Pin Gnd (-)	Pin Gnd (-)
Pin Vcc (+)	Pin 5v (+)

Keterangannya:

1. Pin IN Relay dihubungkan ke pin Pin 12 Arduino Uno
2. Pin Gnd (-) dihubungkan ke Pin Gnd (-) Arduino Uno
3. Pin Vcc (+) dihubungkan ke Pin 5v (+) Arduino Uno

Table 4.4 Rangkaian Led

Led	Arduino Uno
Pin Katoda (+)	Pin 13
Pin Anoda (-)	Pin Gnd (-)

Keterangannya:

1. Pin Katoda (+) Led dihubungkan ke Pin 13 Arduino Uno
2. Pin Anoda (-) Led dihubungkan ke Pin Gnd (-) Arduino Uno

Table 4.5 Rangkaian Buzzer

Buzzer	Arduino Uno
Pin Positif (+)	Pin 11
Pin Negatif (-)	Pin Gnd (-)

Keterangannya:

1. Pin positif (+) Buzzer dihubungkan ke Pin 11 Arduino Uno
2. Pin negatif (-) Buzzer dihubungkan ke Pin Gnd (-) Arduino Uno

Table 4.6 rangkaian LCD 16x2

LCD 16x2	Arduino Uno
Pin Gnd (-)	Pin Gnd (-)
Pin Vcc (+)	Terminal 1 Potensiometer 5v (+)
Pin Contrast	Terminal 2 Potensiometer
Pin Rs	Pin V0
Pin Rw	Terminal 3 Potensiometer Gnd (-)

Pin EN	Pin V1
Pin D4	Pin V2
Pin D5	Pin V3
Pin D6	Pin V4
Pin D7	Pin V5
Pin Blacklight (+)	Pin 5v (+)
Pin Blacklight (-)	Pin Gnd (-)

Keteranganya :

1. Pin Gnd Lcd dihubungkan ke Pin Gnd Arduino Uno
2. Pin Vcc Lcd dihubungkan ke Pin Terminal 1 Potensiometer yang terhubung ke 5v Arduino Uno
3. Pin Contrast Lcd dihubungkan ke Pin Terminal 2 Potensiometer
4. Pin RS Lcd dihubungkan ke pin V0 Arduino Uno
5. Pin RW Lcd dihubungkan ke Pin Terminal 3 Potensiometer yang terhubung ke pin Gnd Arduino Uno
6. Pin EN Lcd dihubungkan ke Pin V1 Arduino Uno
7. Pin D4 Lcd dihubungkan ke Pin V2 Arduino Uno
8. Pin D5 Lcd dihubungkan ke Pin V3 Arduino Uno
9. Pin D6 Lcd dihubungkan ke Pin V4 Arduino Uno
10. Pin D7 Lcd dihubungkan ke Pin V5 Arduino Uno
11. Pin Blacklight (+) dihubungkan ke Pin 5v Arduino Uno
12. Pin Blacklight (-) dihubungkan ke Pin Gnd Arduino Uno

Tabel 4.7 Rangkaian Esp 32

Esp 32	Arduino Uno
Pin Rx	Pin Tx
Pin Tx	Pin Rx
Pin Gnd	Pin Gnd

Penjelasnya :

1. Pin Rx Esp 32 dihubungkan ke Pin Tx Arduino Uno
2. Pin Tx Esp 32 dihubungkan ke Pin Rx Arduino Uno
3. Pin Gnd Esp 32 dihubungkan ke Pin Gnd Arduino Uno

4.1.2. Hasil Perancangan Perangkat Lunak

Hasil dari perancangan perangkat lunak pada *Arduino IDE* adalah program yang telah berhasil dikodekan dan dikompilasi dalam lingkungan *Integrated Development Environment (IDE)* *Arduino*. Program tersebut kemudian dapat diunggah (*upload*) ke papan *Arduino* atau *mikrokontroler* yang sesuai. Berikut merupakan gambar dari program Arduino IDE yang siap di upload.

```

#include <Keypad.h>
#include <LiquidCrystal.h>
#include <Password.h>

#define TouchSensor 10 // Pin digunakan untuk capacitive to
#define RELAY_ON 0
#define RELAY_OFF 1
#define RELAY_1 12 // Pin yang digunakan yaitu pin 12
#define BUZZER_PIN 11 // Definisikan pin untuk buzzer
#define BUZZER_ON HIGH
#define BUZZER_OFF LOW

byte currentLength = 0;
boolean currentState = LOW;
boolean lastState = LOW;
boolean RelayState = LOW;

Password password = Password("1234"); // Password default

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5); // Pin LCD yang

const byte ROWS = 4; // Baris pada keypad
const byte COLS = 4; // Kolom pada keypad

```

Gambar 4.4 Code Arduino IDE

Berikut adalah penjelasan mengenai kode yang disediakan dalam konteks judul skripsi "Pengembangan Prototipe Sistem Keamanan Pintu Menggunakan Sensor Sentuh Dan Verifikasi Kata kunci Berbasis IOT":

1. Library

Library yang Digunakan digunakan adalah:

#include <Keypad.h>: Menggunakan library untuk keypad matriks.

#include <LiquidCrystal.h>: Menggunakan library untuk LCD.

#include <Kata kunci.h>: Menggunakan library untuk manajemen kata kunci.

2. Definisi Pin dan Konstanta

Definisi setiap pin dan konstanta adalah sebagai berikut:

`#define TouchSensor 10``: Mendefinisikan pin 10 untuk sensor sentuh.

`#define RELAY_ON 0``, ``#define RELAY_OFF 1``: Mendefinisikan status relay.

`#define RELAY_1 12``: Mendefinisikan pin 12 untuk relay.

`#define BUZZER_PIN 11``: Mendefinisikan pin 11 untuk buzzer.

`#define BUZZER_ON HIGH``, ``#define BUZZER_OFF LOW``: Mendefinisikan status buzzer.

3. Variabel Global

Variable global atau variable umum yang digunakan sebagai berikut :

`byte currentLength``, ``boolean currentState``, ``boolean lastState``, ``boolean RelayState``: Variabel untuk mencatat status keypad, sensor, dan relay.

`Kata kunci kata kunci = Kata kunci("1234");``: Kata kunci default.

`LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);``: Mengatur pin untuk LCD.

`const byte ROWS = 4;``, ``const byte COLS = 4;``: Mengatur jumlah baris dan kolom pada keypad.

`const int red = 13;``: Mendefinisikan pin untuk LED merah.

`char keys[ROWS][COLS]`: Menyimpan peta tombol pada keypad.

`byte rowPins[ROWS]`, `byte colPins[COLS]`: Mendefinisikan pin baris dan kolom pada keypad.

`Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);`: Membuat objek keypad.

`boolean changeKata kunciMode`, `boolean oldKata kunciMode`: Status mode penggantian kata kunci.

`char newKata kunci[5]`, `char oldKata kunci[5]`: Buffer untuk kata kunci baru dan lama.

`byte newKata kunciIndex`, `byte oldKata kunciIndex`: Indeks untuk memasukkan kata kunci baru dan lama.

`int failedAttempts`: Menghitung jumlah usaha gagal memasukkan kata kunci.

`unsigned long lockoutTime`, `const unsigned long lockoutDuration = 60000;`: Variabel untuk waktu penguncian jika gagal berkali-kali.

4. Fungsi Void Setup

Void setup adalah Menyediakan tempat untuk kode inisialisasi yang dijalankan sekali saja saat program mulai berjalan, kode void setup yang digunakan sebagai berikut :

```
void setup() {  
  Serial.begin(9600);  
  
  pinMode(red, OUTPUT);  
  pinMode(RELAY_1, OUTPUT);  
  pinMode(TouchSensor, INPUT);  
  pinMode(BUZZER_PIN, OUTPUT);  
  
  digitalWrite(RELAY_1, RELAY_OFF);  
  
  keypad.addEventListener(keypadEvent);  
  
  lcd.begin(16, 2);  
  lcd.setCursor(4, 0);  
  lcd.print("SKRIPSI");  
  
  lcd.setCursor(0, 1);  
  lcd.print("anam 20TF026");  
  delay(5000);  
  lcd.clear();  
}
```

Gambar 4.5 Code Bagian Void Setup

5. Fungsi Void Loop

Menyediakan tempat untuk kode yang dijalankan berulang kali selama Arduino berjalan, kode void loop yang digunakan sebagai berikut :

```

void loop() {
  if (millis() - lockoutTime < lockoutDuration && failedAttempts >= 5) {
    lcd.setCursor(0, 0);
    lcd.print("Tunggu 1 Menit");
    lcd.setCursor(0, 1);
    lcd.print("Akses Ditolak");
    return;
  }

  if (!oldPasswordMode && !changePasswordMode) {
    lcd.setCursor(0, 0);
    lcd.print("Masukkan Password");
    lcd.setCursor(0, 1);
    lcd.print("Pass: ");
  }
  keypad.getKey();

  currentState = digitalRead(TouchSensor);
  if (currentState == HIGH && lastState == LOW) {
    Serial.println("Berhasil Buka Dari Dalam");
    delay(1);

    if (RelayState == HIGH) {
      digitalWrite(RELAY_1, RELAY_OFF);
      RelayState = LOW;
    }
  }
}

```

Gambar 4.6 Code Bagian Void Loop

```

    if (RelayState == HIGH) {
      digitalWrite(RELAY_1, RELAY_OFF);
      RelayState = LOW;
    } else {
      digitalWrite(RELAY_1, RELAY_ON);
      digitalWrite(BUZZER_PIN, BUZZER_ON); // H
      digitalWrite(red, HIGH); // Nyalakan LED
      RelayState = HIGH;
      lcd.clear();
      lcd.setCursor(2, 0);
      lcd.print("Berhasil Buka");
      lcd.setCursor(3, 1);
      lcd.print("dari Dalam");
      countdownDisplay(); // Tampilkan countdow
      digitalWrite(RELAY_1, RELAY_OFF); // Mati
      digitalWrite(BUZZER_PIN, BUZZER_OFF); // !
      digitalWrite(red, LOW); // Matikan LED
    }
  }
  lastState = currentState;
}

```

Gambar 4.7 Lanjutan Code Void Loop

6. Fungsi Tambahan

Fungsi tambahan mencakup logika-logika setiap kode sebagai berikut :

`void countdownDisplay()` : Menampilkan hitungan mundur pada LCD selama 10 detik.

`void checkKata kunci()` : Memeriksa apakah kata kunci yang dimasukkan benar atau salah, dan mengatur relay serta buzzer.

`void enterOldKata kunci(char key)` : Mengelola input kata kunci lama untuk perubahan kata kunci.

`void changeKata kunci(char key)` : Mengelola input kata kunci baru.

`void keypadEvent(KeypadEvent eKey)` : Fungsi callback untuk event keypad.

```

void changePassword(char key) {
  if (newPasswordIndex == 0) {
    lcd.clear();
    lcd.print("Masukkan Password");
    lcd.setCursor(0, 1);
    lcd.print("Baru: ");
  }

  if (key == '#') {
    if (newPasswordIndex > 0) {
      newPassword[newPasswordIndex] = '\0'; //
      password = Password(newPassword); // Upda
      lcd.clear();
      lcd.print("Pwd Updated");
      Serial.println("Password updated");
      delay(2000);
    }
    changePasswordMode = false;
    newPasswordIndex = 0;
    lcd.clear();
  } else if (newPasswordIndex < 4) { // Mengasu
    newPassword[newPasswordIndex++] = key;
    lcd.setCursor(newPasswordIndex + 6, 1); //
    lcd.print('*');
  }
}

```

Gambar 4.8 Code Tambahan Ganti Kata kunci

```

void keypadEvent(KeypadEvent eKey) {
  switch (keypad.getState()) {
    case PRESSED:
      if (changePasswordMode) {
        changePassword(eKey);
      } else if (oldPasswordMode) {
        enterOldPassword(eKey);
      } else {
        lcd.setCursor(currentLength + 6, 1);
        lcd.print('*');
        Serial.println(eKey);
        switch (eKey) {
          case '*':
            checkPassword();
            lcd.clear();
            currentLength = 0;
            break;
          case '#':
            password.reset();
            lcd.clear();
            currentLength = 0;
            break;
          case 'A':
            oldPasswordMode = true;

```

Gambar 4.9 Code Tambahan Lanjutan Diatas

Dengan kode ini, sistem keamanan pintu dikendalikan menggunakan keypad untuk memasukkan kata kunci, sensor sentuh untuk membuka pintu dari dalam, dan berbagai fitur seperti pembaruan kata kunci, penanganan usaha gagal, dan feedback pengguna melalui LCD dan buzzer.

4.2. Hasil Pengujian Sistem

Dalam skripsi ini, telah dikembangkan dan di uji purwarupa sistem keamanan pintu yang menggunakan sensor sentuh dan verifikasi password berbasis Internet of Things (IoT). Yang dapat meningkatkan keamanan rumah atau bangunan dengan memanfaatkan teknologi terkini. Pengujian dilakukan untuk menilai kinerja sensor sentuh, keakuratan verifikasi password. Hasil pengujian ini diharapkan dapat menunjukkan keandalan dan efektivitas sistem dalam meningkatkan keamanan pintu serta memberikan dasar untuk pengembangan lebih lanjut.

4.2.1. Hasil Pengujian Perangkat Keras

Sebelum melakukan uji coba sistem perangkat secara keseluruhan, terlebih dahulu komponen atau perangkat keras yang digunakan pada penelitian dan akan di uji coba sehingga bisa memastikan bahwa sensor dapat bekerja dengan benar atau tidak dan keypad Ketika ditekan apakah bisa atau tidak. Berikut ini adalah gambaran dari hasil uji coba sistem pada perangkat keras yaitu:

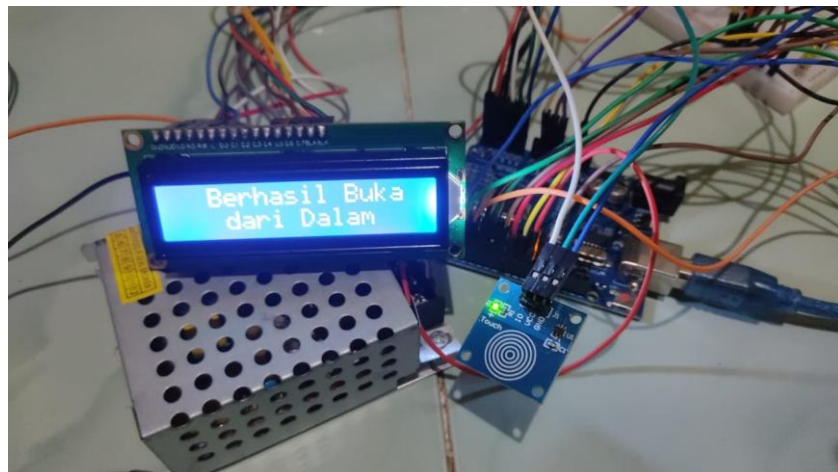
1. Rangkaian Keypad Dengan Arduino



Gambar 4.10 Arduino Dengan Keypad

Keypad digunakan sebagai input untuk verifikasi kata kunci ketika kita mau masuk lewat luar. Pengujian dilakukan dengan memastikan bahwa keypad berfungsi dengan baik ketika ditekan salah satu tombolnya dengan menampilkannya di Lcd 16x2.

2. Rangkaian Sensor Sentuh Dengan Arduino



Gambar 4.11 Sensor Sentuh dengan Arduino

Sensor sentuh digunakan sebagai input sentuhan ketika kita mau buka pintu lewat dalam. Pengujian dilakukan dengan

menyentuh sensor apakah berfungsi ketika disentuh dengan menampilkannya di Lcd 16x2.

3. Rangkaian Relay Dengan Arduino



Gambar 4.12 Relay Dengan Arduino

Relay digunakan sebagai high dan low untuk menggerakkan selenoid. Pengujian dilakukan dengan memastikan bahwa relay bisa menggerakkan selenoid ketika pintu terbuka.

4. Rangkaian Lcd 16x2 dengan Arduino



Gambar 4.13 Lcd 16x2 Dengan Arduino

Lcd 16x2 sebagai layar luaran untuk menampilkan tulisan keypad dan sensor sentuh. Pengujian dilakukan dengan memastikan bahwa Lcd 16x2 menampilkan tulisan yang benar sesuai yang sudah di program.

Setelah pengujian individu selanjutnya dilakukan pengujian secara menyeluruh berikut tabel pengujianya.

Tabel 4.8 Hasil Pengujian Perangkat

No	Nama Perangkat	Fungsi	Hasil	Keterangan
1	Keypad	menentukan data teks yang akan dipilih oleh pengguna	Keypad bekerja ketika ditekan tombol	Tidak ada eror yang ditemukan
2	Sensor Sentuh	mengubah suatu sentuhan menjadi besaran listrik sehingga dapat dianalisa dengan rangkaian listrik tertentu	Sensor sentuh bekerja Ketika disentuh	Tidak ada eror yang ditemukan
3	Relay	mengalirkan arus listrik dari saklar utama ke komponen listrik lainnya,	Relay mampu menggerakkan solenoid Ketika menunjukkan high dan menutupnya ketika low	Tidak ada eror yang ditemukan

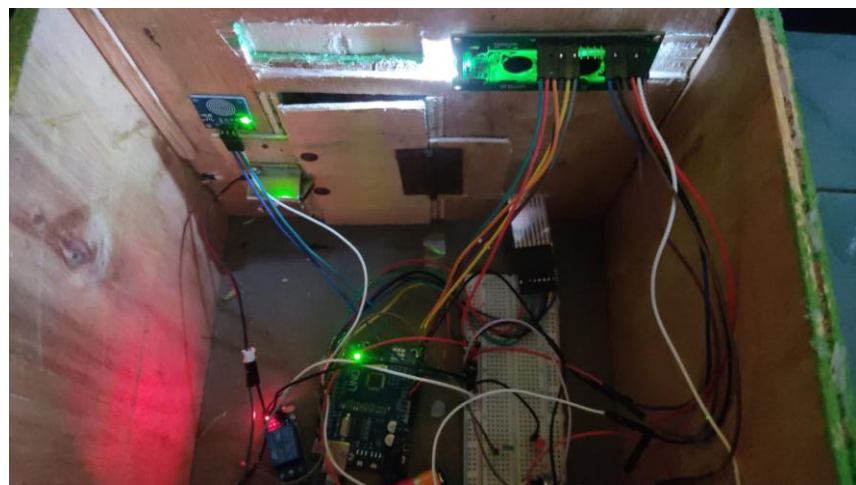
		seperti selenoid		
4	Lcd 16x2	sebagai tampilan,	Lcd menampilkan teks sesuai yang ditentukan	Kadang eror kadang normal
5	Buzzer	menghasilkan suara atau bip	Buzzer berbunyi Ketika pintu mau tertutup selama 10 detik	Tidak ada eror yang ditemukan
6	LED	mengeluarkan cahaya tampak apabila diberi tegangan bias maju dengan arah pancaran menyebar	LED menyala barengan dengan buzzer selama 10 detik	Tidak ada eror yang ditemukan
7	Esp 32	Sebagai wifi untuk Arduino biar bisa mengirim data ke telegram bot	Ketika pintu terbuka maka secara otomatis akan ada notifikasi di telegram bot	Kadang bisa kadang nggak nyambung ke telegram

Itulah hasil dari pengujian perangkat keras yang telah dilakukan. Dalam pengujian tersebut benar-benar diuji oleh peneliti untuk melihat hasil setiap rangkaian yang telah dilakukan.

Pengujian sistem pada sensor sentuh dan keypad ukuran 4x4 dilakukan untuk mengetahui kelayakan sensor dan keypad ini dalam penelitian. Pengujian sensor sentuh dan keypad 4x4 dilakukan untuk mengetahui apakah sensor sentuh bekerja ketika disentuh dan apakah keypad bisa memasukkan kata kunci Ketika ditekan tombolnya. Berikut ini adalah gambar pengujian dari sensor sentuh dan keypad yang terhubung dengan komponen-komponen yang digunakan.



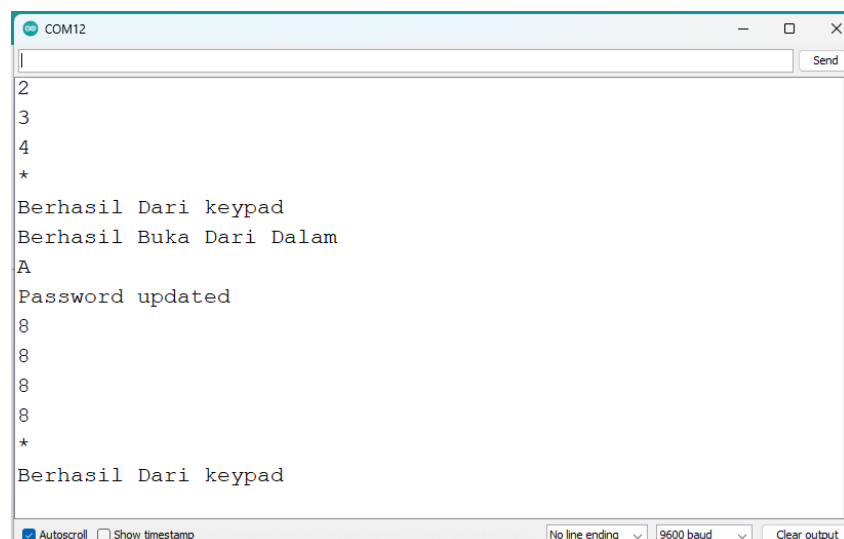
Gambar 4.14 Hasil Pengujian Sistem Dari Luar



Gambar 4.15 Hasil Pengujian Sistem Dari Dalam

4.2.2. Hasil Pengujian Perangkat Lunak

Hasil Pengujain perangkat lunak dari pengembangan prototipe sistem keamanan pintu menggunakan sensor sentuh dan verifikasi kata kunci berbasis IoT berhasil dikembangkan dan diuji dengan baik. Sistem ini memberikan solusi keamanan rumah yang aman dan juga praktis yang dimana bisa meningkatkan kenyamanan pengguna karena sistem ini begitu praktis yang dimana ketika kita mau masuk harus verifikasi kata kunci yang benar dan jika kita salah memasukkan kata kunci sebanyak lima kali berturut-turut maka sistem akan menolak akses dan menunggu selama satu menit dan ini bertujuan untuk menghindari dari serangan bruteforce. Dan ketika mau keluar dari dalam kita tinggal menyentuh sensor dan akan terbuka kuncinya ini bertujuan untuk mempermudah pengguna Ketika mau keluar tidak perlu memasukkan kata kunci lagi.



```
COM12
|
2
3
4
*
Berhasil Dari keypad
Berhasil Buka Dari Dalam
A
Password updated
8
8
8
8
*
Berhasil Dari keypad
```

Gambar 4.16 Hasil Pengujian Perangkat Lunak Pada Serial Monitor



Gambar 4.17 Hasil Pengujian Notifikasi Di Telegram

4.3. Pembahasan

4.3.1. Pembahasa Perangkat Lunak

Dalam pembahasan ini akan dibahas mengenai alasan setiap penggunaan perangkat berikut ini

1. Arduino Uno sangat mudah digunakan dari yang lain seperti esp atau rasbery, terutama bagi pemula, dengan lingkungan pengembangan sederhana dan bahasa pemrograman berbasis C/C++. Komunitas pengguna yang besar dan aktif menyediakan banyak sumber daya seperti tutorial dan proyek contoh. Didukung oleh berbagai macam shield dan modul tambahan yang mempermudah integrasi berbagai sensor dan aktuator.

Stabil dan andal untuk banyak aplikasi prototyping dengan performa konsisten. Keunggulan dari Arduino Uno Harga terjangkau dibandingkan dengan mikrokontroler lainnya. Sederhana dan user-friendly dengan antarmuka dan desain pin-out yang intuitif. Platform open-source, sehingga skematik dan desain dapat diakses dan dimodifikasi sesuai kebutuhan. Kekurangannya Memori dan kecepatan pemrosesan terbatas (2 KB SRAM, 32 KB flash, 16 MHz) yang mungkin tidak cukup untuk aplikasi yang membutuhkan pemrosesan data besar dan cepat. Jumlah pin I/O terbatas, yang dapat menjadi kendala jika proyek membutuhkan banyak input dan output. Tidak memiliki fitur jaringan bawaan seperti Wifi atau Ethernet, memerlukan modul tambahan untuk konektivitas jaringan.

2. Keypad sering digunakan dalam proyek mikrokontroler karena memungkinkan pengguna untuk memasukkan data atau memberikan input langsung ke sistem dengan mudah dan disini keypad yang digunakan adalah keypad matriks 4x4 alasanya dipilih dari keypad daripada keypad 4x3 atau sejenisnya karena keypad ini memiliki antarmuka pengguna yang sederhana dan intuitif, memudahkan pengguna untuk mengoperasikan sistem tanpa perlu perangkat input yang rumit. Kompatibel dengan berbagai mikrokontroler, termasuk Arduino Uno, dan mudah

diintegrasikan menggunakan pustaka yang tersedia di Arduino IDE. Keunggulannya memiliki berbagai konfigurasi tombol (misalnya 4x4) yang fleksibel sesuai kebutuhan proyek. Dapat digunakan untuk berbagai aplikasi, termasuk sistem keamanan, kalkulator, dan kontrol akses. Memudahkan interaksi pengguna dengan sistem melalui input numerik atau alfanumerik. Kekurangan dari keypad ini Jumlah input terbatas sesuai dengan jumlah tombol yang tersedia, yang mungkin tidak cukup untuk aplikasi yang membutuhkan input kompleks atau banyak. Kualitas dan daya tahan tombol bervariasi tergantung pada merek dan model, beberapa mungkin memerlukan penggantian lebih sering.

3. Sensor sentuh digunakan dalam proyek mikrokontroler karena memberikan metode input yang modern dan intuitif kenapa menggunakan sensor sentuh tidak sidik jari karena kemampuannya untuk mendeteksi sentuhan dengan akurasi tinggi, membuat antarmuka pengguna lebih menarik dan interaktif. Keunggulan dari sensor sentuh ini memiliki respon cepat dan akurat terhadap sentuhan, meningkatkan pengalaman pengguna. Memungkinkan desain antarmuka yang lebih bersih dan tahan lama karena tidak ada bagian mekanis yang bergerak. Mendukung berbagai aplikasi, termasuk kontrol

pencahayaannya, panel sentuh pada perangkat rumah pintar, dan sistem keamanan. Kekurangan dari sensor sentuh ini sensitivitas terhadap kondisi lingkungan seperti kelembaban dan suhu, yang dapat mempengaruhi performa. Memerlukan kalibrasi yang tepat untuk menghindari kesalahan deteksi atau sentuhan yang tidak diinginkan. Keterbatasan dalam mendeteksi multi-sentuhan (multitouch) pada beberapa model, yang mungkin tidak cukup untuk aplikasi yang lebih kompleks.

4. ESP32 digunakan sebagai modul wifi untuk Arduino Uno tidak menggunakan modul wifi yang lain karena kemampuannya yang sangat kuat dan serbaguna dalam menambah konektivitas jaringan ke proyek Arduino. Dipilih karena ESP32 memiliki kemampuan pemrosesan yang tinggi, konektivitas Wifi dan Bluetooth bawaan, serta banyak GPIO, yang membuatnya ideal untuk memperluas fungsionalitas jaringan pada Arduino Uno. Keunggulan esp 32 sebagai modul wifi karena memiliki kemampuan pemrosesan yang lebih tinggi dibandingkan dengan mikrokontroler tradisional, dengan prosesor dual-core dan kecepatan hingga 240 MHz. Dilengkapi dengan fitur Wifi dan Bluetooth bawaan, memungkinkan konektivitas nirkabel yang fleksibel dan hemat biaya. Kekurangan dari esp 32 yaitu kompleksitas yang lebih tinggi dibandingkan

dengan Arduino Uno, yang dapat memerlukan waktu dan usaha lebih untuk pemula dalam mempelajarinya. Memerlukan level shifter atau rangkaian pengatur level tegangan untuk interfacing dengan Arduino Uno, karena ESP32 bekerja pada level tegangan 3.3V sedangkan Arduino Uno pada 5V.

5. Solenoid digunakan dalam proyek mikrokontroler karena kemampuannya untuk mengubah energi listrik menjadi gerakan mekanis, memungkinkan kontrol fisik atas berbagai sistem. Dipilih karena kekuatannya dalam menghasilkan gerakan linear atau rotary dengan presisi tinggi. Keunggulan dari solenoid memiliki respon cepat dan kekuatan yang cukup besar untuk menggerakkan atau mengontrol objek mekanis. Struktur yang sederhana dan kokoh membuatnya tahan lama dan mudah dirawat. Konsumsi daya yang relatif rendah untuk operasi dasar, membuatnya efisien untuk berbagai aplikasi. Kekurangan solenoid memerlukan catu daya yang cukup kuat untuk operasi, terutama solenoid yang lebih besar, yang dapat menjadi kendala pada sistem yang bertenaga baterai. Biasanya membutuhkan driver tambahan atau rangkaian pengendali seperti transistor atau relay untuk dikendalikan oleh mikrokontroler.

6. Relay digunakan untuk menggerakkan solenoid dalam proyek mikrokontroler karena kemampuannya untuk mengendalikan arus listrik besar menggunakan sinyal listrik kecil dari mikrokontroler. Keunggulan relay dapat mengendalikan arus dan tegangan tinggi yang dibutuhkan oleh solenoid, memungkinkan kontrol yang efisien dan aman. Memberikan isolasi listrik antara rangkaian kontrol dan rangkaian beban, melindungi mikrokontroler dari lonjakan arus atau kerusakan. Mudah diintegrasikan dengan mikrokontroler seperti Arduino Uno menggunakan sinyal digital sederhana. Kekurangan dari relay memiliki kecepatan switching yang lebih lambat dibandingkan dengan komponen solid-state, yang mungkin menjadi kendala untuk aplikasi yang memerlukan respon cepat. Menghasilkan suara klik saat switching, yang mungkin tidak diinginkan dalam lingkungan yang memerlukan operasi senyap.

4.3.2. Pembahasan Perangkat Lunak

Dalam pembahasan ini akan dibahas tentang logika sistem yang dibuat berdasarkan code yang telah di upload ke papan Arduino Uno.

Dan pada kode kata kunci salah, kenapa memilih ketika salah memasukkan kata kunci sebanyak 5 kali maka sistem akan otomatis menyuruh menunggu selama 1 menit alasannya adalah

Membatasi jumlah percobaan yang salah dan memberikan periode penguncian adalah praktik umum dalam sistem keamanan untuk mencegah serangan brute-force. Serangan brute-force adalah upaya untuk mencoba setiap kombinasi kata kunci yang mungkin hingga menemukan yang benar. Dengan adanya periode penguncian, serangan ini menjadi jauh lebih sulit dan memakan waktu lebih lama. 1 menit dianggap waktu yang cukup lama untuk mencegah penyerang berusaha terus-menerus, tetapi masih dapat diterima oleh pengguna sah yang mungkin secara tidak sengaja salah memasukkan kata kunci beberapa kali. Waktu ini memberikan keseimbangan antara keamanan dan kegunaan.

```
int failedAttempts = 0; // Penghitung usaha ga
unsigned long lockoutTime = 0; // Waktu ketika
const unsigned long lockoutDuration = 60000; ,
```

Gambar 4.18 code Lama Waktu Penguncian

Dan alasan pemilihan 5 kali salah tidak 3 kali atau 10 kali karena Pengguna sah dapat salah memasukkan kata kunci beberapa kali karena berbagai alasan, seperti kesalahan mengetik atau lupa sementara. Dengan memberikan 5 kesempatan, sistem memberikan toleransi yang cukup bagi kesalahan manusia tanpa segera mengunci pengguna. Banyak sistem keamanan mengikuti kebijakan yang serupa dengan memberikan beberapa kesempatan sebelum menerapkan

penalti. karena jumlah tersebut dianggap cukup untuk mengidentifikasi upaya yang tidak sah sementara tetap memberikan kesempatan yang wajar kepada pengguna sah.

```
digitalPin100(100, LOW),  
if (failedAttempts >= 5) {  
    lockoutTime = millis(); // Mulai pe  
    lcd.clear();  
    lcd.print("Tunggu 1 Menit");  
    lcd.setCursor(0, 1);  
    lcd.print("Akses Ditolak");  
}
```

Gambar 4.19 code kata kunci salah 5x Mulai Penguncian

Dan pada digit kata kunci hanya menerapkan panjang kata kunci 4 digit dalam sistem keamanan pintu ini memiliki beberapa alasan yang berkaitan dengan keseimbangan antara keamanan, kegunaan, dan kompleksitas implementasi. Dengan panjang 4 digit, terdapat 10^4 atau 10,000 kombinasi kemungkinan (0000 hingga 9999). Ini memberikan tingkat keamanan yang memadai untuk banyak aplikasi, terutama jika dilengkapi dengan fitur seperti penguncian setelah sejumlah percobaan yang salah. Menggunakan 4 digit membuat sistem cukup tangguh terhadap percobaan tebak-tebakan secara acak dalam waktu singkat, terutama dengan adanya fitur penguncian selama 1 menit setelah 5 kali kesalahan. Mengingat dan memasukkan kata kunci 4 digit adalah hal yang relatif mudah bagi pengguna. Ini adalah panjang kata kunci yang sering digunakan dalam sistem seperti ATM atau

kunci smartphone, sehingga pengguna sudah terbiasa dengan panjang ini. Kata kunci yang lebih panjang mungkin memberikan keamanan lebih tinggi, tetapi bisa jadi sulit diingat dan lebih rentan terhadap kesalahan pengetikan.

```
boolean changePasswordMode = false;
boolean oldPasswordMode = false;
char newPassword[5]; // Mengasumsikan pa
byte newPasswordIndex = 0;
char oldPassword[5];
byte oldPasswordIndex = 0;
```

Gambar 4.20 Codngan Untuk Mengasumsikan Kata kunci 4 Digit

Agar lebih kompleks dibuat juga hitung mundur jeda waktu penutupan selama 10 detik karena alasan 10 detik umumnya cukup bagi pengguna untuk membuka pintu dan melewati sistem keamanan tanpa terburu-buru. Ini memberi mereka cukup waktu untuk menutup pintu dengan benar dan memastikan keamanan. Jika waktu hitung mundur terlalu pendek, pengguna mungkin merasa terburu-buru, yang dapat menyebabkan kesalahan atau ketidaknyamanan. Memberikan waktu tambahan seperti 20 atau 30 detik bisa meningkatkan risiko keamanan karena lebih banyak waktu yang tersedia untuk pintu tetap tidak terkunci. Ini bisa dimanfaatkan oleh pihak yang tidak berwenang untuk mencoba masuk. Pada perangkat dengan sumber daya terbatas, seperti mikrokontroler yang digunakan dalam proyek ini, menjaga komponen sistem aktif lebih lama dari yang diperlukan dapat

mengonsumsi daya tambahan. Hitung mundur 10 detik membantu menghemat daya dan menjaga kinerja sistem tetap optimal.

```
void countdownDisplay() {  
    for (int i = 10; i >= 1; i--) { // Perulangan 10 detik  
        lcd.clear();  
        lcd.print("Pnt Tutp Dalm:");  
        lcd.setCursor(0, 1);  
        lcd.print(i);  
        delay(1000); // Tunggu 1 detik  
    }  
}
```

Gambar 4.21 Code Hitung Mundur 10 Detik

Penelitian ini juga menerapkan Fungsi tombol penggantian kata kunci melalui keypad dalam sistem keamanan pintu ini memungkinkan pengguna untuk mengubah kata kunci yang ada dengan yang baru. Implementasi ini penting untuk meningkatkan keamanan dan fleksibilitas sistem, karena memungkinkan pengguna untuk mengubah kata kunci jika ada kekhawatiran bahwa kata kunci yang lama telah diketahui oleh pihak yang tidak berwenang. Saat pengguna menekan tombol 'A' pada keypad, sistem akan masuk ke mode penggantian kata kunci. Pada saat ini, variabel **oldKata kunciMode** diset menjadi true. Pengguna diminta untuk memasukkan kata kunci lama terlebih dahulu untuk memverifikasi identitas mereka. Kata kunci lama dimasukkan melalui keypad dan ditampilkan sebagai karakter '*' di LCD untuk keamanan. Jika kata kunci lama benar, sistem akan mengaktifkan mode penggantian kata

kunci baru dengan mengeset **changeKata kunciMode** menjadi true. Setelah kata kunci lama diverifikasi, pengguna diminta untuk memasukkan kata kunci baru. Kata kunci baru juga dimasukkan melalui keypad dan ditampilkan sebagai karakter '*' di LCD. Setelah memasukkan kata kunci baru dan menekan tombol '#', kata kunci baru akan disimpan dan digunakan untuk menggantikan kata kunci lama.

Penelitian ini juga menggunakan modul wifi esp 32 sebagai wifi internetnya agar Arduino bisa mengirim data ke telegram melalui esp 32 sebagai modulnya berikut gambar code serial antar Arduino dan esp 32.

```
digitalWrite(REL, HIGH); // Nyalakan REL  
lcd.clear();  
lcd.print("Berhasil Buka ");  
Serial.println("Berhasil Dari keypad");  
countdownDisplay(); // Tampilkan countdown di  
digitalWrite(RELAY_1, RELAY_OFF); // Matikan r
```

Gambar 4.22 Serial Di Arduino Uno

```
void setup() {  
  Serial.begin(9600); // Serial komuni  
  Serial2.begin(9600); // Serial2 untu
```

Gambar 4.23 Serial Di Esp 32

Itulah pembahasan perangkat keras dan perangkat lunak pada penelitian perancangan prototype sistem keamanan pintu dengan verifikasi password dan sensor sentuh berbasis iot, pada pembahasan diatas dijelaskan dari setiap komponen yang digunakan serta keunggulan dan kelebihan setiap komponen pada

perangkat keras dan juga pada perangkat lunak dijelaskan setiap logika kode nya.