

DAFTAR PUSTAKA

- Adress, M. A. C. (t.t.). Sistem Cerdas Keamanan Ruang Berbasis. *vol, 1*, 132–136.
- Adhitama, M. R., & Putra, I. G. E. W. (2019). Analisis Pembacaan dan Penulisan Data Buku Menggunakan RFID Mifare RC-522 untuk Perpustakaan. *Jurnal Sistem dan Informatika (JSI)*, 14(1), 23–34.
- Ahmad, A. (2012). Perkembangan teknologi komunikasi dan informasi: akar revolusi dan berbagai standarnya. *Jurnal Dakwah Tabligh*, 13(1), 137–149.
- Amrullah, M. A., Lhaksmana, K. M., & Adytia, D. (2018). Pembangunan dan pengujian protokol MQTT & WebSocket untuk Aplikasi IoT Rumah Cerdas berbasis Android. *eProceedings of Engineering*, 5(2).
- Aprianty, H., Rudiyanto, G., Dani, R., Purnawan, H., & Yanto, H. N. (2023). Peran Ditsamapta dalam Meningkatkan Pelayanan Publik untuk Mewujudkan Polisi Republik Indonesia yang Presisi. *Jurnal Publisitas*, 9(2).
- Areni, I. S., Palantei, E., Suyuti, A., Baharuddin, M., Samman, F. A., & Umraeni, A. E. (2019). Pengembangan dan Implementasi Smart Campus Berbasis Smart Card Di Institut Agama Islam Negeri Bone. *JURNAL TEPAT: Teknologi Terapan untuk Pengabdian Masyarakat*, 2(1), 1–7.
- Bahri, S., & Suhardiyanto, S. (2018). Sistem Keamanan Ruang Server Menggunakan Teknologi RFID Dan Password. *eLEKTUM*, 15(1), 11–18.
- Darmawan, A. (2010). Sistem Pengamanan Ganda Pada Kendaraan Bermotor Menggunakan Kartu Secara Elektronik Berbasis Mikrokontroler. *Jurnal Informatika*, 10(2), 44–53.
- Gunarjati, A. S. (2019). *Teknologi Iot Pada Monitoring Dan Otomasi Kolam Pembesaran Ikan Lele Berbasis Mikrokontroler*.
- Hadis, M. S., Palantei, E., Ilham, A. A., & Hendra, A. (2018). Design of smart lock system for doors with special features using bluetooth technology. *2018 international conference on information and communications technology (ICOIACT)*, 396–400.
- Hanafiah, N., Kariman, C. P., Fandino, N., Halim, E., Jingga, F., & Atmadja, W. (2021). Digital door-lock using authentication code based on ANN encryption. *Procedia Computer Science*, 179, 894–901.
- Iman, F. F., & Alfi, I. (2018). *Purwarupa smart door lock menggunakan multi sensor berbasis Arduino* (Doctoral dissertation, University of Technology Yogyakarta).
- , Sumendap, S. S., & Koagouw, F. (2016). Pentingnya Penggunaan Jaringan Wi-alam Memenuhi Kebutuhan Informasi Pemustaka Pada Kantor Perpustakaan



- Dan Kearsipan Daerah Kota Tidore Kepulauan. *ACTA DIURNA KOMUNIKASI*, 5(2).
- Lagan, M. D., & Ary, M. (2021). Sistem Kendali Kunci Pintu Menggunakan Voice Command Berbasis Internet of Things (Iot). *eProsiding Teknik Informatika (PROTEKTIF)*, 2(1), 1–12.
- Lee, I., & Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business horizons*, 58(4), 431–440.
- Nataliana, D., Syamsu, I., & Giantara, G. (2014). Sistem Monitoring Parkir Mobil menggunakan Sensor Infrared berbasis RASPBERRY PI. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 2(1), 68.
- Ometov, A., Bezzateev, S., Mäkitalo, N., Andreev, S., Mikkonen, T., & Koucheryavy, Y. (2018). Multi-factor authentication: A survey. *Cryptography*, 2(1), 1.
- Rahayu, A., & Masdi, H. (2020). Sistem Kendali Rumah Pintar Menggunakan Voice Recognition Module V3 Berbasis Mikrokontroler dan IOT. *JTEV: Jurnal Teknik Elektro Dan Vokasional*, 6(2), 19–32.
- Rankl, W., & Effing, W. (2010). Smart card security. *Smart Card Handbook*, 667–734.
- Riyadi, S., & Heriyanto, R. (2017). Aplikasi Smart Card E-KTP Berbasis RFID Untuk Sistem Keamanan Ruangan. *Jurnal Elektra*, 2(2), 83–92.
- Siswanto, S., Anif, M., Hayati, D. N., & Yuhefizar, Y. (2019). Pengamanan pintu ruangan menggunakan arduino mega 2560, mq-2, dht-11 berbasis android. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 3(1), 66–72.
- Sulaiman, O. K., & Widarma, A. (2017). *Sistem internet of things (IoT) berbasis cloud computing dalam campus area network*.
- Tarigan, J. (2004). Biometric Security: Alternatif Pengendalian Dalam Sistem Informasi Akuntansi Terkomputerisasi. *Jurnal akuntansi dan keuangan*, 6(2), 90-105.
- Tawakal, M. I., & Ramdhani, Y. (2021). Smart Lock Door menggunakan Akses E-KTP Berbasis Internet of Things. *Jurnal Responsif: Riset Sains dan Informatika*, 3(1), 83–91.
- Ulum, S., & Budiyanto, M. (2020). Prototipe Pengaman Pintu Rumah Menggunakan Voice Recognition dengan EasyVR Berbasis Mikrokontroler. *Jurnal Listrik, Instrumentasi dan Elektronika Terapan (JuLIET)*, 1(2).
- Verma, G. K., & Tripathi, P. (2010). A digital security system with door lock system using RFID technology. *International Journal of Computer Applications*, 5(11), 6–8.
- . N. A., Noertjahjono, S., & Pranoto, Y. A. (2020). Rancang Bangun Sistem nanan Pada Sepeda Motor Menggunakan Sms Gateway Berbasis okontroller. *Jati (Jurnal Mahasiswa Teknik Informatika)*, 4(2), 113–119.



Zuhri, K., & Ikhwan, A. (2020). Perancangan Sistem Keamanan Ganda Brangkas Berbasis Telegram Menggunakan Mikrokontroler ESP32-CAM. *Jurnal Teknologi dan Informatika (JEDA)*, 1(2).



Lampiran 1 Tabel Jarak Baca *Reader*

1. Tabel Jarak *Tapping* Tanpa Penghalang

UID CARD	Jarak Terjauh Tapping Tanpa Penghalang (mm)									
	1	2	3	4	5	6	7	8	9	10
0461558AE85C80	15	15	15	15	15	15	15	15	15	15
04424DDA7C7880	15	15	15	15	15	15	15	15	15	15
058F1DB0A8D100	15	15	15	15	15	15	15	15	15	15
047F58E2C66180	15	15	15	15	15	15	15	15	15	15
E09BA419	15	15	15	15	15	15	15	15	15	15
12E59B22	15	15	15	15	15	15	15	15	15	15
E05DEB19	15	15	15	15	15	15	15	15	15	15
9000E521	15	15	15	15	15	15	15	15	15	15
C116C519	15	15	15	15	15	15	15	15	15	15
12E59B22	15	15	15	15	15	15	15	15	15	15

2. Tabel Jarak *Tapping* dengan Penghalang

UID CARD	Jarak Terjauh Tapping dengan Penghalang (mm)									
	1	2	3	4	5	6	7	8	9	10
0461558AE85C80	15	15	15	15	15	15	15	15	15	15
04424DDA7C7880	15	15	15	15	15	15	15	15	15	15
058F1DB0A8D100	15	15	15	15	15	15	15	15	15	15
047F58E2C66180	15	15	15	15	15	15	15	15	15	15
E09BA419	15	15	15	15	15	15	15	15	15	15
12E59B22	15	15	15	15	15	15	15	15	15	15
E05DEB19	15	15	15	15	15	15	15	15	15	15
9000E521	15	15	15	15	15	15	15	15	15	15
C116C519	15	15	15	15	15	15	15	15	15	15
12E59B22	15	15	15	15	15	15	15	15	15	15



Lampiran 2 Tabel Delay RFID

UID CARD	Delay (ms)									
	1	2	3	4	5	6	7	8	9	10
0461558AE85C80	0	0	0	0	0	0	0	0	0	0
04424DDA7C7880	0	0	0	0	0	0	0	0	0	0
058F1DB0A8D100	0	0	0	0	0	0	0	0	0	0
047F58E2C66180	0	0	0	0	0	0	0	0	0	0
E09BA419	0	0	0	0	0	0	0	0	0	0
12E59B22	0	0	0	0	0	0	0	0	0	0
E05DEB19	0	0	0	0	0	0	0	0	0	0
9000E521	0	0	0	0	0	0	0	0	0	0
C116C519	0	0	0	0	0	0	0	0	0	0
12E59B22	0	0	0	0	0	0	0	0	0	0



Lampiran 3 Tabel *Delay Fingerprint*

Finger ID	Delay(ms)										Rata-Rata
	1	2	3	4	5	6	7	8	9	10	
#101	81	240	253	79	137	279	59	157	282	63	163
#102	63	143	111	291	243	120	200	185	85	298	173,9
#103	142	233	98	70	298	156	64	64	235	116	147,6
#104	295	244	78	110	242	66	132	132	158	998	245,5
#105	64	67	79	173	281	110	79	345	109	77	138,4
#106	169	320	240	144	125	122	74	178	105	380	185,7
#107	164	161	176	376	160	166	125	88	160	125	170,1
#108	384	323	276	163	276	164	104	145	80	242	215,7
#109	230	242	182	163	110	266	179	179	189	154	189,4
#110	187	106	120	99	74	125	102	165	168	111	125,7



Lampiran 4 Tabel *Delay Voice Recognition*

User	Delay										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
Yusuf	2,29	2	1,94	1,89	1,66	1,94	1,76	1,55	1,79	2	1,882
Alhan	1	1,06	1,77	1,85	1,73	2,36	1,67	2,31	2,33	3,94	2,002
Padli	1,5	2,02	1,6	2,08	2,56	2,38	1,01	1,93	3,54	2,94	2,156
Dzulfadli	2,41	2,28	1,79	2,54	1,93	2,71	1,51	1,34	1,78	2,22	2,051
Fauzan	1,76	2,39	2,18	2,81	3,81	2,83	2,44	2,14	1,46	2,35	2,417
Pati	2,22	3,15	1,68	1,51	1,72	2,12	2,81	1,96	1,97	2,16	2,13
Wanda	2,14	2,75	2,16	3,02	3,55	4	3	3,28	2,19	2,18	2,827
Aulia	1,74	2,29	2,119	1,49	2,48	1,82	2,49	1,73	1,65	2,82	2,0629
Tria	2	2,18	2,41	1,93	2,62	1,79	2,55	1,83	2,14	1,58	2,103
Ade	2,33	2,13	2,83	1,61	3,57	1,81	2,73	1,93	3,82	2,93	2,569



Lampiran 5 Tabel *Delay Face Recognition*

1. Tabel *face recognition* dengan mata terbuka

USER	MATA TERBUKA										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
Yusuf	1,7	1,43	1,66	1,29	1,77	1,74	1,59	1,87	2,76	1,74	1,755
Alhan	2	2,06	1,58	1,75	1,92	2,67	2,31	2,1	2,2	2,07	2,066
Padli	2,5	4,27	2,35	2,06	1,26	3,88	2,45	2,59	3,19	3,22	2,777
Dzulfadli	2,45	2,1	1,48	3,15	1,98	3,48	1,94	2,38	2,52	2,18	2,366
Fauzan	2,32	1,93	2,18	2,16	2,76	2,61	2,47	2,78	2,83	2,5	2,454
Pati	2,72	3,78	1,34	2,37	1,92	2,29	1,99	1,72	2,87	2,69	2,369
Wanda	2,96	3,23	3,31	2,86	4,11	2,55	2,72	3,09	2,58	3,6	3,101
Aulia	1,86	1,47	2,52	1,52	2,51	2,9	2,26	1,67	1,76	1,73	2,02
Tria	2,49	2,19	2,29	3,93	3,76	2,31	4,06	2,12	2,67	1,69	2,751
Ade	3,02	2,16	2,75	2,96	4,1	1,75	1,66	1,8	3,03	2,56	2,579

2. Tabel *face recognition* dengan mata tertutup

USER	MATA TERTUTUP										Raya-rata
	1	2	3	4	5	6	7	8	9	10	
Yusuf	1,45	2	1,94	1,34	2,21	2,14	2,29	1,5	1,7	2	1,857
Alhan	2,37	1,72	2,12	2,73	1,74	1,7	1,53	1,62	2,5	2,49	2,052
Padli	1,42	2,63	2,2	2,26	2,82	3,41	2,64	4,14	2,15	2,84	2,651
Dzulfadli	2,5	3,16	3,09	2,49	2,73	1,65	2,6	2,19	1,82	2,51	2,474
Fauzan	2,58	2,42	2,53	2,77	2,68	2,62	2,77	2,66	2,55	3	2,658
Pati	2,31	2,26	1,74	3,02	2,19	1,86	2,19	2,66	1,21	4,14	2,358
Wanda	3,16	2,74	2,56	4,73	1,69	2,33	3,08	2,95	1,31	1,76	2,631
Aulia	1,86	2,09	2,16	2,32	4,54	1,92	2,31	1,9	2,48	1,98	2,356
Tria	1,85	2,56	2,74	3,16	2,18	2,57	3,09	2,72	2,55	3,41	2,683
Ade	2,19	3,75	2,49	3,28	1,7	4,16	2,87	1,96	3,81	2,17	2,838



Lampiran 6 Tabel *Delay Biometric Reader*

1. Tabel akses Reader melalui Web Browser

USER	WEB BROWSER									
	1	2	3	4	5	6	7	8	9	10
Yusuf	6,78	7,04	6,43	6,2	6,67	5,58	6,85	5,97	7,03	6,88
Alhan	4,8	5,98	6,91	6,82	7,7	6,6	7,56	4,71	7,34	7,47
Padli	7,77	7,72	7,07	9,56	5,64	11,87	5,59	6,29	12,27	11,3
Dzulfadli	6,67	6,08	4,22	9,69	8,29	8,32	9,37	8,49	8,98	10
Fauzan	5,1	4,93	5,91	6,41	10,5	6,16	5,23	5,26	6,74	12,3
Pati	9,37	10,6	9,54	15,1	7,46	7,11	9,12	11,4	8,25	9,22
Wanda	10,2	10,8	9,69	6,59	12,5	7,52	10,7	10,8	8,07	16,9
Aulia	6,12	15,4	9,38	8,37	10,2	8,67	9,54	10,7	7,32	8,66
Tria	8,37	9,38	15,7	6,12	8,47	9,92	7,83	13,1	11,42	7,32
Ade	9,56	9,69	10,5	10,2	8,76	7,56	11,5	9,76	8,96	10,8

2. Tabel akses reader menggunakan Smartcard

USER	RFID										Rata-Rata
	1	2	3	4	5	6	7	8	9	10	
Yusuf	5,57	6,07	5,41	6,91	6,11	6,21	5,85	5,82	5,73	5,54	5,922
Alhan	10,1	5,47	7,35	8,16	10,4	7	5,83	5,14	6,62	8,06	7,413
Padli	9,37	8,64	12,07	7,28	14,04	4,27	9,08	6,8	8,48	6,84	8,687
Dzulfadli	7,6	11,3	8,97	8,4	4,86	7,32	6,9	9,54	8,32	6,22	7,943
Fauzan	8,99	5,24	7,27	8,02	6,29	7,6	7,38	5,45	9,79	9,01	7,504
Pati	10,6	7,51	10,04	11,5	10,57	8,96	6,97	11,5	7,56	8,76	9,397
Wanda	8,67	9,76	7,81	11,3	6,55	7,43	11,3	7,83	9,76	7,84	8,829
Aulia	8,89	8,07	11,11	7,92	10,02	8,98	8,49	9,01	8,89	8,37	8,975
Tria	11,4	7,81	9,76	8,67	16,8	8,07	10,8	10,7	7,97	8,74	10,08
Ade	11,5	10	7,58	10,6	9,22	8,25	11,4	9,12	7,46	7,11	9,226



Lampiran 7 Program Index

```
#include "variable.h"

void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    /* Initialize the display and other peripherals */
    Serial.println("Setting up WiFi");
    connectWifi();
    delay(500);
    Serial.println("Setting up lockweb");
    int_webblock();
    delay(500);
    Serial.println("Fetching Time");
    initNTP();
    delay(250);
    Serial.println("Setting up RELay");
    int_relaylock();
    Serial.println("Setting up RFID");
    initRFID();
    delay(500);
    Serial.println("Setting up FingerPrint Sensor");
    initFingerSensor();
    delay(500);
    Serial.println("Setting up voice Sensor");
    int_voicerec();
    delay(500);

    Serial.println("Setting up facerecogintion Sensor");
    int_facerecognition();
```



```
ln("Setting up LCD Display");
```

```
GetDateTime();  
//DateTimeH.setText(RealT.c_str()); //PRINT TIMEEEEEEEEEEEEEEEEEEE  
prntime2lcd(RealT.c_str());  
delay(1000);  
printNormalModeMessage(); // Default initial screen  
}
```



Lampiran 8 Program *RFID*

```
#include <SPI.h>           // Library required for RC522 RFID Reader
#include <MFRC522.h>       // Library of RC522 RFID Reader http://192.168.189.98/
#define SS_PIN 5 /*Slave Select Pin*/
#define RST_PIN 4 /*Reset Pin for RC522*/
#define LED_G 12 /*Pin 8 for LED*/
MFRC522 mfrc522(SS_PIN, RST_PIN); /*Create MFRC522 initialized*/
void initRFID(){
    SPI.begin(); // Init SPI bus
    mfrc522.PCD_Init(); // Init MFRC522 card
}
void ScanRFID(){
    digitalWrite(LockerPin, HIGH);
    if (weblock == 0){
        if ( ! mfrc522.PICC_IsNewCardPresent() ) { //If a new PICC placed to RFID reader continue
            return;
        }
        if ( ! mfrc522.PICC_ReadCardSerial() ) { //Since a PICC placed get Serial and continue
            return;
        }
    }
    String content= "";
    user="";
    for (byte i = 0; i < mfrc522.uid.size; i++)
    {
        content.concat(String(mfrc522.uid.uidByte[i] < 0x10 ? " 0" : " "));
        content.concat(String(mfrc522.uid.uidByte[i], HEX));
    }
    content.toUpperCase();
    mfrc522.PICC_HaltA(); // Stop reading

    intln(content.substring(1));
    ock==1){
        'user3";
        ssword();
```



```

GetDateTime();

//DateTimeV.setText(RealT.c_str());

prntime2lcd(RealT.c_str());

//me delay(1000);

printsettext("You Open Weblock !");

//me delay(3000);

fingershow();

//prntime2lcd(RealT.c_str());

//me delay(2000);

int fingerID = getFingerprintID();

if((fingerID == 1) || (fingerID == 2) || (fingerID == 3) || (fingerID == 4) || (fingerID == 5) ||
(fingerID == 6) || (fingerID == 7) || (fingerID == 8) || (fingerID == 9) || (fingerID == 10) ||
(fingerID == 11) || (fingerID == 12) || (fingerID == 13) || (fingerID == 14) || (fingerID == 15) ||
(fingerID == 16) || (fingerID == 17) || (fingerID == 18) || (fingerID == 19) || (fingerID == 20) ||
(fingerID == 21) || (fingerID == 22) || (fingerID == 23) || (fingerID == 24) || (fingerID == 25) ||
(fingerID == 26) || (fingerID == 27) || (fingerID == 28)){

    ValidFinger();

    prntime2lcd(RealT.c_str());

    //me delay(2000);

    voiceshow();

    loop_voiceMODULE();

    if((voiceflag == 0) || (voiceflag == 1)||(voiceflag == 2)||(voiceflag == 3)||(voiceflag ==
4)||(voiceflag == 5)||(voiceflag == 6)||(voiceflag == 7)||(voiceflag == 8)||(voiceflag ==
9)||(voiceflag == 10)||(voiceflag == 11)||(voiceflag == 12)||(voiceflag == 13)||(voiceflag ==
14)||(voiceflag == 15)||(voiceflag == 16)||(voiceflag == 17)||(voiceflag == 18)||(voiceflag ==
19)||(voiceflag == 20))

    {

        ValidVoice();

        //me delay(1000);

//=====face=====
==

        faceshow();

        //me delay(2000);

        loop_facerecognition ();

```



```
recog == 0 )
```

```
alWrite(LockerPin, LOW);
```

```

Serial.println("unlock");
ValidFace();

}
else {
    digitalWrite(LockerPin, HIGH);
    Serial.println("lock");
    inValidFace();}
facerecog = 1;
}
else {inValidVoice();}
voiceflag = -1;
//PinEnter.show();
printtime2lcd(RealT.c_str());
weblock=0;
printNormalModeMessage();
}else{
weblock=0;
inValidFinger();
printtime2lcd(RealT.c_str());
//me delay(2000);
printNormalModeMessage();
}
}

//User 1

else if (content.substring(1) == "04 61 55 8A E8 5C 80") //change here the UID of the
card/cards that you want to give access
{
    user = "user1";
    ValidRFID();
    GetDateTime();
    imeV.setText(RealT.c_str());
    ie2lcd(RealT.c_str());
    1000); //me
    text("Welcome Yusuf !");

```



```

//delay(3000); //me
fingershow();
//prntime2lcd(RealT.c_str());
//delay(2000);//me
int fingerID = getFingerprintID();
Serial.println(fingerID);
if((fingerID == 1) || (fingerID == 2) || (fingerID == 3) || (fingerID == 4)){
  ValidFinger();
  prntime2lcd(RealT.c_str());
  //delay(2000);//me
  voiceshow();
  loop_voiceMODULE();
  if((voiceflag == 0) || (voiceflag == 1)||(voiceflag == 2))
  {
    ValidVoice();
    //delay(1000);//me

//=====face=====
==

    faceshow();
    loop_facerecognition ();
    if(facerecog == 0 )
    {
      digitalWrite(LockerPin, LOW);
      Serial.println("unlock");
      ValidFace();
    }
    else {
      digitalWrite(LockerPin, HIGH);
      Serial.println("lock");
      inValidFace();}
    facerecog = 1;

    {inValidVoice();}
    lag = -1;

```



```

//PinEnter.show();

printtime2lcd(RealT.c_str());

weblock=0;

printNormalModeMessage();
}else{

weblock=0;

invalidFinger();

//DateTimeF.setText(RealT.c_str());

printtime2lcd(RealT.c_str());

//me delay(2000);

printNormalModeMessage();

}

}

// user 2

else if(content.substring(1) == "04 78 4C FA C4 61 80"){

user = "user2";

ValidRFID();

GetDateTime();

//DateTimeV.setText(RealT.c_str());

printtime2lcd(RealT.c_str());

//me delay(1000);

printsettext("Welcome Ibni !");

//me delay(3000);

fingershow();

//printtime2lcd(RealT.c_str());

//me delay(2000);

int fingerID = getFingerprintID();

if((fingerID == 5) || (fingerID == 6) || (fingerID == 7) || (fingerID == 8)){

ValidFinger();

printtime2lcd(RealT.c_str());

//me delay(2000);

how();

voiceMODULE();

ceflag == 3) || (voiceflag == 4)|| (voiceflag == 5))

```



```

{
    ValidVoice();
    //me delay(1000);

//=====face=====
==

    faceshow();
    loop_facerecognition ();
    if(facerecog == 0 )
    { ValidFace();}
    else {inValidFace();}
    facerecog = -1;
}

    else {inValidVoice();}
    //PinEnter.show();
    printtime2lcd(RealT.c_str());
    weblock=0;
    printNormalModeMessage();
} else{
    weblock=0;
    inValidFinger();
    printtime2lcd(RealT.c_str());
    //me delay(2000);
    printNormalModeMessage();
}
}

// user 3
else if(content.substring(1) == "04 5F 8D 5A 1A 5B 80"){
    user = "user3";
    ValidRFID();
    GetDateTime();
    //DateTimeV.setText(RealT.c_str());
    ie2lcd(RealT.c_str());
    lay(1000);
    text("Welcome Ochang !");

```



```

//me delay(3000);
fingershow();
//prntime2lcd(RealT.c_str());
//me delay(2000);
int fingerID = getFingerprintID();

if((fingerID == 9) || (fingerID == 10) || (fingerID == 11) || (fingerID == 12)){
    ValidFinger();
    prntime2lcd(RealT.c_str());
    //me delay(2000);
    voiceshow();
    loop_voiceMODULE();
    if((voiceflag == 6) || (voiceflag == 7)|| (voiceflag == 8))
    {
        ValidVoice();
        //me delay(1000);

//=====face=====
==

        faceshow();
        loop_facerecognition ();
        if(facerecog == 0 )
        { ValidFace();}
        else {inValidFace();}
        facerecog = -1;
    }
    else {inValidVoice();}
    //PinEnter.show();
    prntime2lcd(RealT.c_str());
    weblock=0;
    printNormalModeMessage();
} else{
    ck=0;
    dFinger();
    me2lcd(RealT.c_str());

```



```

//me delay(2000);

printNormalModeMessage();
}
}

// user 4
else if(content.substring(1) == "04 7F 58 E2 C6 61 80"){
    user = "user4";
    ValidRFID();
    GetDateTime();
    //DateTimeV.setText(RealT.c_str());
    printtime2lcd(RealT.c_str());
    //me delay(1000);
    printsettext("Welcome Gabriel !");
    delay(100);
    fingershow();
    //printtime2lcd(RealT.c_str());
    //me delay(2000);
    int fingerID = getFingerprintID();

    if((fingerID == 13) || (fingerID == 14) || (fingerID == 15) || (fingerID == 16)){
        ValidFinger();
        printtime2lcd(RealT.c_str());
        //me delay(2000);
        voiceshow();
        loop_voiceMODULE();
        if((voiceflag == 9) || (voiceflag == 10) || (voiceflag == 11))
        {
            ValidVoice();
            //me delay(1000);

```



=====face=====

```

ow();
faceRecognition ();

```

```

if(facerecog == 0 )
{ValidFace();}
else {inValidFace();}
facerecog = -1;
}
else {inValidVoice();}
//PinEnter.show();
printtime2lcd(RealT.c_str());
weblock=0;
printNormalModeMessage();
}else{
weblock=0;
inValidFinger();
printtime2lcd(RealT.c_str());
//me delay(2000);
printNormalModeMessage();
}
}

// user 5
else if(content.substring(1) == "E0 9B A4 19"){
user = "user5";
ValidRFID();
GetDateTime();
//DateTimeV.setText(RealT.c_str());
printtime2lcd(RealT.c_str());
//me delay(1000);
printsettext("Welcome Fadil !");
delay(100);
fingershow();
//printtime2lcd(RealT.c_str());
//me delay(2000);
erID = getFingerprintID();

erID == 17) || (fingerID == 18) || (fingerID == 19) || (fingerID == 20)){

```



```

ValidFinger();

printtime2lcd(RealT.c_str());

//me delay(2000);

voiceshow();

loop_voiceMODULE();

if((voiceflag == 12) || (voiceflag == 13)||(voiceflag == 14))
{
    ValidVoice();

    //me delay(1000);

//=====face=====
==

    faceshow();

    loop_facerecognition ();

    if(facerecog == 0 )
    { ValidFace();}

    else {inValidFace();}

    facerecog = -1;

}

    else {inValidVoice();}

//PinEnter.show();

printtime2lcd(RealT.c_str());

weblock=0;

printNormalModeMessage();

}else{

    weblock=0;

    inValidFinger();

    printtime2lcd(RealT.c_str());

    //me delay(2000);

    printNormalModeMessage();

}
}

```



6

```

content.substring(1) == "04 78 2C EA FF 5C 80"){
    'user6';
}

```

```

ValidRFID();
GetDateTime();
//DateTimeV.setText(RealT.c_str());
prntime2lcd(RealT.c_str());
//me delay(1000);
printsettext("Welcome Dhedy !");
delay(100);
fingershow();
//prntime2lcd(RealT.c_str());
//me delay(2000);
int fingerID = getFingerprintID();

if((fingerID == 21) || (fingerID == 22) || (fingerID == 23) || (fingerID == 24)){
    ValidFinger();
    prntime2lcd(RealT.c_str());
    //me delay(2000);
    voiceshow();
    loop_voiceMODULE();
    if((voiceflag == 15) || (voiceflag == 16)|| (voiceflag == 17))
    {
        ValidVoice();
        //me delay(1000);

//=====face=====
==

        faceshow();
        loop_facerecognition ();
        if(facerecog == 0 )
        { ValidFace();}
        else {inValidFace();}
        facerecog = -1;
    }

    {inValidVoice();}
    nter.show();
    me2lcd(RealT.c_str());

```



```
    weblock=0;
    printNormalModeMessage();
} else {
    weblock=0;
    inValidFinger();
    printtime2lcd(RealT.c_str());
    //me delay(2000);
    printNormalModeMessage();
}
}
else {
    weblock=0;
    inValidRFID;
    printtime2lcd(RealT.c_str());
    printsettext("Access Denied !!");
    //me delay(3000);
    printNormalModeMessage();
}
}
```



Lampiran 9 Program *Fingerprint*

```
#include <Adafruit_Fingerprint.h> // Library of Fingerprint Sensor

#define SerialAT Serial1

#define MODEM_TX      26
#define MODEM_RX      27

#define TX1 26
#define RX1 27

Adafruit_Fingerprint finger = Adafruit_Fingerprint(&Serial1); //Fingerprint sensor is
connected to Serial port 2

void initFingerSensor(){

    // SerialAT.begin(57600, SERIAL_8N1, MODEM_RX, MODEM_TX);

    // set the data rate for the sensor serial port
    finger.begin(57600);
    delay(5);
    if (finger.verifyPassword()) {
        Serial.println("Found fingerprint sensor!");
    } else {
        Serial.println("Did not find fingerprint sensor :(");
        while (1) { delay(1); }
    }
    finger.getTemplateCount();
    if (finger.templateCount == 0) {
        Serial.print("Sensor doesn't contain any fingerprint data. Please run the 'enroll' example.");
    }
    else {
        Serial.println("Waiting for valid finger...");
        Serial.print("Sensor contains "); Serial.print(finger.templateCount); Serial.println("
templates");
    }
}

//*****Get the Fingerprint ID*****
erprintID() {
    rtln("cek fingerprint");
    in:
    = finger.getImage();
```



```

switch (p) {
    case FINGERPRINT_OK:
        //Serial.println("Image taken");
        GetDateTime();
        scanfinger();
        //DateTimeFS.setText(RealT.c_str());
        printtime2lcd(RealT.c_str());
        break;
    case FINGERPRINT_NOFINGER:
        //Serial.println("No finger detected");
        goto checkAgain;
        return 0;
    case FINGERPRINT_PACKETRECEIVEERR:
        //Serial.println("Communication error");
        return -2;
    case FINGERPRINT_IMAGEFAIL:
        //Serial.println("Imaging error");
        return -2;
    default:
        //Serial.println("Unknown error");
        return -2;
}

// OK success!
p = finger.image2Tz();
switch (p) {
    case FINGERPRINT_OK:
        //Serial.println("Image converted");
        break;
    case FINGERPRINT_IMAGEMESS:
        //Serial.println("Image too messy");
        return -1;
    case FINGERPRINT_PACKETRECEIVEERR:
        //Serial.println("Communication error");
        return -2;
}

```



```

case FINGERPRINT_FEATUREFAIL:
    //Serial.println("Could not find fingerprint features");
    return -2;
case FINGERPRINT_INVALIDIMAGE:
    //Serial.println("Could not find fingerprint features");
    return -2;
default:
    //Serial.println("Unknown error");
    return -2;
}
// OK converted!
p = finger.fingerFastSearch();
if (p == FINGERPRINT_OK) {
    //Serial.println("Found a print match!");
} else if (p == FINGERPRINT_PACKETRECEIVEERR) {
    //Serial.println("Communication error");
    return -2;
} else if (p == FINGERPRINT_NOTFOUND) {
    //Serial.println("Did not find a match");
    return -1;
} else {
    //Serial.println("Unknown error");
    return -2;
}
// found a match!
Serial.print("Found ID #"); Serial.print(finger.fingerID);
Serial.print(" with confidence of "); Serial.println(finger.confidence);

return finger.fingerID;
}

```



Lampiran 10 Program *Voice Recognition*

```
#include "VoiceRecognitionV3.h"

VR myVR(2, 3); // 2:RX 3:TX, you can choose your favourite pins.

uint8_t records[7]; // save record

uint8_t buf[64];

void printSignature(uint8_t *buf, int len)
{
    int i;
    for (i = 0; i < len; i++) {
        if (buf[i] > 0x19 && buf[i] < 0x7F) {
            Serial.write(buf[i]);
        }
        else {
            Serial.print("[");
            Serial.print(buf[i], HEX);
            Serial.print("]");
        }
    }
}

void printVR(uint8_t *buf)
{
    Serial.println("VR Index\ Group\ RecordNum\ Signature");
    Serial.print(buf[2], DEC);
    Serial.print("\ \ ");
    if (buf[0] == 0xFF) {
        Serial.print("NONE");
    }
    else if (buf[0] & 0x80) {
        Serial.print("UG ");
        Serial.print(buf[0] & (~0x80), DEC);
        Serial.print("SG ");
        Serial.print(buf[0], DEC);
    }
}
```



```
int("SG ");
int(buf[0], DEC);
```

```

    }
    Serial.print("\ ");
    Serial.print(buf[1], DEC);
    Serial.print("\ \ ");
    if (buf[3] > 0) {
        printSignature(buf + 4, buf[3]);
    }
    else {
        Serial.print("NONE");
    }
    Serial.println("\
\
");
}

void int_voicerec()
{
    /** initialize */
    myVR.begin(9600);
    Serial.println("Elechouse Voice Recognition V3 Module\
\
Control LED sample");
    if (myVR.load((uint8_t)(0)) >= 0) {
        Serial.println("open loaded");
    }
    if (myVR.load((uint8_t)(1)) >= 0) {
        Serial.println("door loaded");
    }
    if (myVR.load((uint8_t)(2)) >= 0) {
        Serial.println("unlock loaded");
    }
}

```



voiceMODULE()

```
{  
  oldtimer = millis();  
  Serial.println("voice active");  
  Serial.println("voice Taken");  
  while((millis() - oldtimer) <= 10000){  
    int ret;  
    ret = myVR.recognize(buf, 50);  
    // Serial.print("lockkk:");  
    // Serial.println(ret);  
    if (ret > 0) {  
      Serial.print("Found A Match Voice!:"");  
      Serial.println(buf[1]);  
      voiceflag = buf[1];  
      break;  
      /** voice recognized */  
      //printVR(buf);  
    }  
  }  
}
```



Lampiran 11 Program *Face Recognition*

```

#define pinface 33
#define trigerface 25
void int_facerecognition()
{
    pinMode(pinface,INPUT);
    pinMode(trigerface,OUTPUT);
    digitalWrite(trigerface,HIGH);
}
void loop_facerecognition ()
{
    oldtimer = millis();
    while((millis() - oldtimer) <= 15000){
        digitalWrite(trigerface,LOW);
        facerecog=digitalRead(pinface);
        Serial.print("pinface:");
        Serial.print(facerecog);
        Serial.print(",time:");
        Serial.println(millis() - oldtimer);
        if(facerecog ==0){
            break;
        }
        delay(10);
    }
    digitalWrite(trigerface,HIGH);
}

```



Lampiran 12 Program LCD

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
// Initialize the LCD library with I2C address and LCD size
LiquidCrystal_I2C lcd (0x27, 16,2);
void int_lcd()
{
  lcd. init ();
  lcd. backlight ();
}
void printNormalModeMessage() {
  lcd.clear();
  lcd.setCursor (0, 0);
  lcd.print ("PLACE RFID CARD");
  lcd.setCursor (0, 1);
  lcd.print (WiFi.localIP());
// Home.show();
// DateTimeH.setText(RealT.c_str());
  delay(150);
}
void ValidRFID()
{
  lcd.clear();
  lcd.setCursor (0, 0);
  lcd.print (" RFID DETECTED ");
  lcd.setCursor (0, 1);
  lcd.print ("==PLEASE WAIT==");
//me delay(2000);
}
void ValidPassword()
```



```
!;
rsor (0, 0);
" WEBLOCK DETECTED ");
```

```

    lcd.setCursor (0, 1);
    lcd.print ("==PLEASE WAIT==");
    //me delay(2000);
}

void inValidRFID()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print (" INVALID RFID ");
    lcd.setCursor (0, 1);
    lcd.print ("PLEASE TRY AGAIN");
    //me delay(1000);
}

void printtime2lcd(String datetme)
{
    lcd.setCursor (0, 1);
    lcd.print (datetme);
    // me delay(500);
}

void printsettext(String texts)
{
    lcd.setCursor (0, 1);
    lcd.print (texts);
    // me delay(500);
}

void fingershow()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print ("PLACE YO FINGER");

```



```

    rsor (0, 1);
    " ON SCANNER");
    r(500);

```

```

}

void scanfinger()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print ("Scanning");
    lcd.setCursor (0, 1);
    lcd.print ("finger");
    delay(100);
}

void ValidFinger()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print (" VALID FINGER ");
    lcd.setCursor (0, 1);
    lcd.print ("HELLO BOS");
    //me delay(2000);
}

void inValidFinger()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print (" INVALID FINGER ");
    lcd.setCursor (0, 1);
    lcd.print ("PLEASE TRY GAIN");
    delay(1000);
}

void voiceshow()
{
    ;
    rsor (0, 0);
    "Speak to Voice");

```



```

    lcd.setCursor (0, 1);
    lcd.print (" to Microphone");
    delay(100);
}

void ValidVoice()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print (" VALID VOICE ");
    lcd.setCursor (0, 1);
    lcd.print ("HELLO BOS");
    delay(100);
}

void inValidVoice()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print (" INVALID VOICE ");
    lcd.setCursor (0, 1);
    lcd.print ("PLEASE TRY GAIN");
    delay(1000);
}

void faceshow()
{
    lcd.clear();
    lcd.setCursor (0, 0);
    lcd.print ("Please In front");
    lcd.setCursor (0, 1);
    lcd.print (" of Camera ");
    //me delay(1000);
}

```



face()

;

```
    lcd.setCursor (0, 0);  
    lcd.print (" VALID FACE ");  
    lcd.setCursor (0, 1);  
    lcd.print (" WELCOME");  
    delay(5000);  
}  
void inValidFace()  
{  
    lcd.clear();  
    lcd.setCursor (0, 0);  
    lcd.print (" INVALID FACE ");  
    lcd.setCursor (0, 1);  
    lcd.print ("PLEASE TRY GAIN");  
    delay(3000);  
}
```



Lampiran 13 Program *Relay*

```
#define LockerPin 32          // Relay driving Solenoid lock is connected to this pin (Active Low)

void int_relaylock()
{
    pinMode(LockerPin, OUTPUT);    // Seting Relay PIN as Output
    digitalWrite(LockerPin, HIGH); // Relay PIN is active LOW

}
```



Lampiran 14 Program *Web Lock door*

```
// Import required libraries

#ifndef ESP32

#include <WiFi.h>

#include <AsyncTCP.h>

#else

#include <ESP8266WiFi.h>

#include <ESPAsyncTCP.h>

#endif

#include <ESPAsyncWebServer.h>

String outputState();

const char* PARAM_INPUT_1 = "state";

//const int output = 2;

// Create AsyncWebServer object on port 80

AsyncWebServer server(80);

const char index_html[] PROGMEM = R"rawliteral(

<!DOCTYPE HTML><html>

<head>

<title>smart biometric authentication</title>

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

html {font-family: 'Times New Roman', Times, serif; display: inline-block; text-align: center;}

h2 {font-size: 2.6rem;}

body {max-width: 600px; margin:0px auto; padding-bottom: 10px; background-color: #ccc;}

.switch {position: relative; display: inline-block; width: 120px; height: 68px}

.switch input {display: none}

.slider {position: absolute; top: 0; left: 0; right: 0; bottom: 0; background-color: #ccc; border-
radius: 34px}

.slider:before {position: absolute; content: ""; height: 52px; width: 52px; left: 8px; bottom: 8px;
background-color: #fff; -webkit-transition: .4s; transition: .4s; border-radius: 68px}

input:checked+.slider {background-color: #2196F3}

input:checked+.slider:before {-webkit-transform: translateX(52px); -ms-transform:
52px); transform: translateX(52px)}
```



```

<body>
  <h3>WELCOME TO MULTI BIOMETRIC AUTHENTICATION WEB</h3>
  <p>Open Access Reader <span id="state"> %STATE%</span></p>
  %BUTTONPLACEHOLDER% <br>
  <button onclick="logoutButton()">Logout</button>
  <script>function toggleCheckbox(element) {
    var xhr = new XMLHttpRequest();
    if(element.checked){
      xhr.open("GET", "/update?state=1", true);
      document.getElementById("state").innerHTML = "ON";
    }
    else {
      xhr.open("GET", "/update?state=0", true);
      document.getElementById("state").innerHTML = "OFF";
    }
    xhr.send();
  }
  function logoutButton() {
    var xhr = new XMLHttpRequest();
    xhr.open("GET", "/logout", true);
    xhr.send();
    setTimeout(function(){ window.open("/logged-out","_self"); }, 1000);
  }
</script>

</body>
</html>
)rawliteral";
const char logout_html[] PROGMEM = R"rawliteral(
<!DOCTYPE HTML><html>

<head>
  ne="viewport" content="width=device-width, initial-scale=1">

```



```

<p>Logged out or <a href="/">return to homepage</a>.</p>

<p><strong>Note:</strong> close all web browser tabs to complete the logout process.</p>
</body>
</html>

)rawliteral";

// Replaces placeholder with button section in your web page
String processor(const String& var){
    //Serial.println(var);
    if(var == "BUTTONPLACEHOLDER"){
        String buttons = "";
        String outputStateValue = outputState();
        buttons+= "<p><label class=\"switch\"><input type=\"checkbox\"
onchange=\"toggleCheckbox(this)\" id=\"output\" \" + outputStateValue + "><span
class=\"slider\"></span></label></p>";
        return buttons;
    }
    if (var == "STATE"){
        if(webblock){
            return "ON";
        }
        else {
            return "OFF";
        }
    }
    return String();
}

String outputState(){
    if(webblock){
        return "checked";
    }
    else {

```



```

}

void int_weblock(){
  // pinMode(output, OUTPUT);
  // digitalWrite(output, LOW);
  // // Connect to Wi-Fi
  // WiFi.begin(ssid, password);
  // while (WiFi.status() != WL_CONNECTED) {
  //   delay(1000);
  //   Serial.println("Connecting to WiFi..");
  // }

  // Print ESP Local IP Address
  Serial.println(WiFi.localIP());
  // Route for root / web page
  server.on("/", HTTP_GET, [](AsyncWebServerRequest *request){
    if(!request->authenticate(http_username, http_password))
      return request->requestAuthentication();
    request->send_P(200, "text/html", index_html, processor);
  });
  server.on("/logout", HTTP_GET, [](AsyncWebServerRequest *request){
    request->send(401);
  });
  server.on("/logged-out", HTTP_GET, [](AsyncWebServerRequest *request){
    request->send_P(200, "text/html", logout_html, processor);
  });
  // Send a GET request to <ESP_IP>/update?state=<inputMessage>
  server.on("/update", HTTP_GET, [] (AsyncWebServerRequest *request) {
    if(!request->authenticate(http_username, http_password))
      return request->requestAuthentication();
    String inputMessage;
    String inputParam;
    input1 value on <ESP_IP>/update?state=<inputMessage>
    st->hasParam(PARAM_INPUT_1)) {
    essage = request->getParam(PARAM_INPUT_1)->value();

```



```
inputParam = PARAM_INPUT_1;
webblock=inputMessage.toInt();
//digitalWrite(output, inputMessage.toInt());
}
else {
    inputMessage = "No message sent";
    inputParam = "none";
}
Serial.println(inputMessage);
request->send(200, "text/plain", "OK");
});
// Start server
server.begin();
}
```



Lampiran 15 Program *Variable*

```
String user="";           // String variable For Username
String data="";           // String variable For PIN data from Nextion
String branchName = "Secure Co-Op Bank";
String userName = "";
int weblock=0;
#define open  (0)
#define door  (1)
#define three (2)
int voiceflag=-1;
unsigned long oldtimer;
int facerecog =1 ;
// Replace with your network credentials
// Replace with your network credentials
const char* ssid = "123456789";
const char* password = "123456789"; rumah ada gak ?
const char* http_username = "admin";
const char* http_password = "admin";
#include "rtconline.h"
#include "lcd.h"
#include "relay_solenoid.h"
#include "facerecogniton.h"
#include "voice_recognition.h"
#include "fingersensors.h"
#include "RFID.h"
#include "weblockdoor.h"
```

