

DAFTAR PUSTAKA

- Abiyyi, M. H. (2020). *Rancang Bangun Sistem Monitoring dan Reservasi Parkir Berbasis Online Lahan Parkir Mobil Fakultas Teknik Universitas Hasanuddin*. Makassar: Universitas Hasanuddin.
- Agustin, M., Mekongga, I., Admirani, I., & Azro, I. (2019). Desain Sistem Parkir Berbasis RFID. *Jurnal JUPITER*, 11(1), 21-28.
- Anas, N. (2019). *Implementasi Algoritma Genetika Untuk Pengendalian Perangkat Listrik Pada Miniatur Ruangan Cerdas*. Makassar: Universitas Hasanuddin.
- Anshar, M., Ejah, A. U., Zaenab, M., Ida, R., Emir, A., Alwi, . . . Akkas, M. (2017). Smart Room Design: A Pilot Project. *Proc. The 1st EPI Int. Conf. on Science and Engineering*, 1(4), 19-26.
- Astuti, Y. (2015). Radio Frequency Identification (RFID) Untuk Keamanan Parkir Sepeda Motor di SMK X. *Jurnal Teknologi Informasi*, X(29).
- Darat, D. P. (1996). *Pedoman Teknis Penyelenggaraan Fasilitas Parkir*. Jakarta: Departemen Perhubungan Direktur Jenderal Perhubungan Darat.
- Djamal, H. (2014). Radio Frequency Identification (RFID) Dan Aplikasinya. *TESLA*, 16(1).
- HID Corporation. (2006). *Understanding Card Data Formats*. Diambil kembali dari HID Global: https://www.hidglobal.com/sites/default/files/hid-understanding_card_data_formats-wp-en.pdf
- Mallawakkang, M. N. (2020). *ATM Beras Dengan Sistem Aktifasi RFID*. Makassar: Universitas Hasanuddin.

- Parkash, D., Kundu, T., & Kaur, P. (2012). THE RFID TECHNOLOGY AND ITS APPLICATIONS: A REVIEW. *International Journal of Electronics, Communication & Instrumentation Engineering Research and Development (IJECIERD)*, 2(3), 110-114.
- Sharon, D., Sapri, & Supardi, R. (2014). Membangun Jaringan Wireless Local Area Network (WLAN) Pada CV. BIQ Bengkulu. *Jurnal Media Infotama*, 10(1), 35-41.
- Warpani, S. (2002). *Pengelolaan Lalu Lintas dan Angkutan Jalan*. Bandung: ITB.

LAMPIRAN

Lampiran 1. Gambar tiap node dan palang

- Node satu (gate masuk motor)



- Node dua (gate masuk mobil)



- Node tiga (gate keluar motor)



- Node empat (gate keluar motor)



Lampiran 2. Program Raspberry Pi

- Gate_ardu.py

```
#!/usr/bin/env python3
import serial
from modules.repeatabletimer import TimerEx
from subprocess import call
from search_users import *

ser = serial.Serial('/dev/ttyUSB0', 9600, timeout=1)
ser.reset_input_buffer()

"""
Gate mechanism
"""

def close_gate():
    ser.write("close".encode('utf-8'))

close_gate_timer = TimerEx(interval_sec = 20, function = close_gate)
def pass_thru():
    global close_gate_timer
    if close_gate_timer.is_alive():
        close_gate_timer.cancel() # cancel active countdown
        ser.write("open".encode('utf-8'))
        close_gate_timer.start() # start countdown to closing gate

if __name__ == '__main__':
    masterCard = "7480045"
    shutDownCard = "15142909"
    rebootCard = "2742370"

    updateMode = False

    while True:
        if ser.in_waiting > 0:
            value = ser.readline().decode('utf-8').rstrip()
            value = str(value).lstrip("0")
            print(value)
            if str(value) == masterCard:
                ser.write("master_on".encode('utf-8'))
                updateMode = True
                print("master ON")
            elif str(value) == shutDownCard:
                ser.write("boot".encode('utf-8'))
                call("sudo shutdown -h now", shell=True)
```

```

elif str(value) == rebootCard:
    ser.write("boot".encode('utf-8'))
    call("sudo shutdown -r now", shell=True)
elif not updateMode:
    if user_exists(str(value)):
        pass_thru()
    else:
        ser.write("denied".encode('utf-8'))
else:
    if str(value) != masterCard:
        updateMode = False
    if user_exists(str(value)):
        ser.write("denied".encode('utf-8'))
        print("id already on database, denied")
    else:
        ser.write("accepted".encode('utf-8'))
        update_json(str(value))
        print("DB updated")

```

- **get_database.py**

```

import requests
from datetime import datetime

try:
    response = requests.get('http://192.168.0.3/sistempalang/get_json.php', {'Accept':
'application/json'})
    response.raise_for_status()
    if 'application/json' in response.headers.get('Content-Type'):
        with open('/home/pi/Programs/gate/database.json', 'wb') as outf1:
            outf1.write(response.content)
        pass
    else:
        print("Not a json file. Database not downloaded.")
except requests.exceptions.HTTPError as e:
    print("Failed to sync, please check connection")
    print(e)

```

- **search_users.py**

```

import json

def user_exists(rfid):
    json_data = open('/home/pi/Programs/gate/database.json')
    database = json.load(json_data)

    for user in database:
        if str(user["id"]).lstrip("0") == rfid:
            json_data.close()
            return True
    json_data.close()
    return False

```

```
def update_json(rfid):  
    json_data = open('/home/pi/Programs/gate/database.json')  
    database = json.load(json_data)  
    database.append({'id': rfid, 'nama': "", 'alamat': ""})  
  
    with open('/home/pi/Programs/gate/database.json', 'w') as f:  
        json_data.close()  
        json.dump(database, f)
```


Lampiran 3. Program Arduino Uno

```
#include <Wiegand.h>
#define data_0 2
#define data_1 3
#define relay_open 4
#define relay_close 5
#define buzzer 13
Wiegand wiegand;
String data = "";

void setup() {
  Serial.begin(9600);
  wiegand.onReceive(receivedData, "Card
  readed: ");
  wiegand.onReceiveError(receivedDataError,
  "Card read error: ");
  wiegand.onStateChange(stateChanged, "State
  changed: ");
  wiegand.begin(Wiegand::LENGTH_ANY,
  true);

  pinMode(data_0, INPUT);
  pinMode(data_1, INPUT);
  pinMode(relay_open, OUTPUT);
  pinMode(relay_close, OUTPUT);
  pinMode(buzzer, OUTPUT);
  digitalWrite(relay_open, HIGH);
  digitalWrite(relay_close, HIGH);

  attachInterrupt(digitalPinToInterrupt(data_0),
  pinStateChanged, CHANGE);
  attachInterrupt(digitalPinToInterrupt(data_1),
  pinStateChanged, CHANGE);
  pinStateChanged();
}

void pinStateChanged() {
  wiegand.setPin0State(digitalRead(data_0));
  wiegand.setPin1State(digitalRead(data_1));
}

void stateChanged(bool plugged, const char*
  message) {
  Serial.print(message);
  Serial.println(plugged ? "CONNECTED" :
  "DISCONNECTED");
}

void receivedData(uint8_t* data, uint8_t bits,
  const char* message) {
  uint8_t bytes = (bits + 7) / 8;
  String x = "";
  for (int i = 0; i < bytes; i++) {
    x += String(data[i] >> 4, HEX);
    x += String(data[i] & 0xF, HEX);
  }

  x.toUpperCase();
  Serial.println(hexToDec(x));
}

long hexToDec(String hexString){
  long decValue = 0;
  int nextInt;

  for (int i = 0; i < hexString.length(); i++)
  {
    nextInt = int(hexString.charAt(i));
    if (nextInt >= 48 && nextInt <= 57) nextInt =
    map(nextInt, 48, 57, 0, 9);
    if (nextInt >= 65 && nextInt <= 70) nextInt =
    map(nextInt, 65, 70, 10, 15);
    if (nextInt >= 97 && nextInt <= 102) nextInt =
    map(nextInt, 97, 102, 10, 15);
    nextInt = constrain(nextInt, 0, 15);
    decValue = (decValue * 16) + nextInt;
  }
  return decValue;
}

void receivedDataError(Wiegand::DataError
  error, uint8_t* rawData, uint8_t rawBits, const
  char* message) {
  Serial.print(message);
  Serial.print(Wiegand::DataErrorStr(error));
  Serial.print(" - Raw data: ");
  Serial.print(rawBits);
  Serial.print("bits / ");

  uint8_t bytes = (rawBits + 7) / 8;
  for (int i = 0; i < bytes; i++) {
    Serial.print(rawData[i] >> 4, 16);
    Serial.print(rawData[i] & 0xF, 16);
  }
  Serial.println();
}

void open_gate() {
  digitalWrite(relay_open, LOW); delay(2000);
  digitalWrite(relay_open, HIGH);
}

void close_gate() {
  digitalWrite(relay_close, LOW); delay(2000);
  digitalWrite(relay_close, HIGH);
}
```

```
void denied() {  
    digitalWrite(buzzer, HIGH); delay(300);  
    digitalWrite(buzzer, LOW); delay(300);  
    digitalWrite(buzzer, HIGH); delay(300);  
    digitalWrite(buzzer, LOW);  
}  
  
void boot() {  
    digitalWrite(buzzer, HIGH); delay(1500);  
    digitalWrite(buzzer, LOW); delay(800);  
    digitalWrite(buzzer, HIGH); delay(1500);  
    digitalWrite(buzzer, LOW);  
}  
  
void master_on() {  
    digitalWrite(buzzer, HIGH); delay(1000);  
    digitalWrite(buzzer, LOW); delay(300);  
    digitalWrite(buzzer, HIGH); delay(1000);  
    digitalWrite(buzzer, LOW);  
}  
  
void id_added() {  
    digitalWrite(buzzer, HIGH); delay(500);  
    digitalWrite(buzzer, LOW); delay(300);  
    digitalWrite(buzzer, HIGH); delay(1500);  
    digitalWrite(buzzer, LOW);  
}
```

Lampiran 3. Program Tampilan Website

```
<?php
session_start();
$db = new
PDO("mysql:host=localhost;dbname=sistemp
alang", "root", "");
$db->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
?>
<!DOCTYPE html>
<html>
<head>
<link rel="shortcut icon" href="favicon.ico">
<meta charset="utf-8">
<meta name="viewport"
content="width=device-width, initial-
scale=1">
<title>Sistem Palang | IASCR | Universitas
Hasanuddin</title>
<style type="text/css">
*,
*::before,
*::after { box-sizing: border-box;}
html,
body {margin: 0;padding: 0;}
body {font-family: "Roboto";font-size: 1rem;
background-color: #496591;
color: #333; }
h1 {font-family: 'Arial'; font-size: 2.2rem;
color: #efefef; margin: 40px 15px; display:
block; text-align: center; }

.tabs { margin: 0 auto; max-width: 80%; }

.content { background-color: #ffffff;
border-radius: 0 3px 3px 3px;
padding: 15px; min-height: 400px;
box-shadow: 0 10px 10px #111;}

ul {list-style: none; margin: 0; padding: 0;}
li {display: inline-block; }

a {display: block; padding: 10px 15px;
width: 150px; text-decoration: none;
background-color: #fff;border-radius: 3px 3px
0 0; background-color: #ccc; color: #555;
font-weight: bold; text-align: center; }

a.active {border-top: 3px solid #ffcc00;
position: relative; background-color: #ffffff;
color: #333;}
a.active::before {position: absolute;
content: "";height: 80px;width: 100%;top: 0;
left: 0; border-radius: 3px;background-color:
#efefef; display: block;z-index: -1;}

hr {border: 0; border-bottom: 1px solid #aaa;}

table, tr, th, td {border: 1px solid #aaa;}

table {border-collapse: collapse;width: 100%;}

.loginbox {margin: 25px auto 0;width: 320px;}

.loginbox input, .loginbox button {width:
100%;padding: 10px 15px;margin-bottom:
15px;}

.info {padding: 15px;border: 1px solid #aaa;
border-left: 5px solid #fc0; input, textarea
{border: 1px solid #ccc; border-radius:
none;font-family: inherit; font-size: 1rem; }

input:focus, textarea:focus {outline: 1px solid
#004d40;}

.submit {border: 0; height: 50px;
background-color: #fc0; font-family: inherit;
font-size: 1rem;}

th, td {padding: 7px 15px; text-align: left;}

th {background-color: #efefef;}

.error, .popup-error, .popup-success {width:
100%; border: 1px solid #aaa; padding: 15px;
border-left: 5px solid #f05; padding: 7px 15px;
display: none;}

.popup-success {border-left-color: #5f0;}
.popup-error, .popup-success {margin-bottom:
10px;}

.btn {border: 0; border-radius: 3px; font-
family: inherit; font-size: 1rem; padding: 7px
15px;}

.blue {background-color: #50f; color: #efefef;}

.red {background-color: #f05; color: #efefef;}

.popup {box-shadow: 2px 2px 5px black;
max-width: 500px; width: 100%;position:
absolute; z-index: 10; top: 50%;left:
50%;transform: translate(-50%, -
50%);background-color: #fff; padding: 25px
50px; }
```

```
.popup input, .popup textarea, .popup button
{width: 100%;margin-bottom: 10px; font-
family: inherit; font-size: 1rem; padding:
10px 15px;}
```

```
.popup-background { position: absolute; top:
0; left: 0; height: 0; width: 0; overflow:
hidden; transition: all .25s linear; transition-
property: height, width;}
```

```
.popup-background.active {z-index: 9; top: 0;
left: 0; width: 100%;height: 100%;transition:
all .25s linear; transition-property: height,
width; }
```

```
.popup-tambah {margin-top: 10px; border: 0;
background-color: #fc0; height: 50px; }
</style></head>
```

```
<body><h1>Sistem Palang<br />Universitas
Hasanuddin</h1>
<div class="tabs"> <div class="menu">
<ul><li><a class="item active" data-
target="content2">Admin</a></li> </ul>
</div>
```

```
<div class="content"><div id="content1">
<div class="titles">
<?php if(isset($_SESSION['loginsession'])): ?>
<h2 class="after-logout" style="float:
left;">Halaman Admin</h2>
<?php else: ?>
```

```
<h2 class="after-login" style="float:
left;">Halaman Login</h2>
<?php endif; ?>
 <div style="display:
table;clear: both;content: ""></div></div>
<hr />
<div id="r-content">
```

```
<?php if(isset($_SESSION['loginsession'])): ?>
<br /> <div style="text-align: right; Selamat
Datang, <b><?php echo
strtoupper($_SESSION['loginsession']);
?></b>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<button
class="btn blue" onclick="popup()">Tambah
Pegguna</button>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<button
class="btn red"
onclick="logout()">Keluar</button>
</div><br />
```

```
<table><tr>
```

```
<th>ID</th>
<th>Nama</th>
<th>Alamat</th>
</tr>
<?php $sql = "SELECT * FROM pelanggan";
$stmt = $db->query($sql); while ($data =
$stmt->fetch()) {
?>
<tr><td><?php echo $data['id'] ?></td>
<td><?php echo $data['nama'] ?></td>
<td><?php echo $data['alamat'] ?></td>
</tr> <?php } ?> </table>
<?php else: ?>
```

```
<div class="loginbox"><div
class="error"></div><p class="info">
Lakukan 'Login' terlebih dahulu untuk
melanjutkan. </p>
```

```
<form id="forms"><input type="text"
name="uname" id="uname"
placeholder="Nama Pengguna" />
<input type="password" name="password"
id="password" placeholder="Kata Sandi" />
<button class="submit"
type="submit">Login</button></form>
</div><?php endif; ?>
</div></div></div></div>
```

```
<div class="popup-background"> <form
class="popup" id="popup-form"
onsubmit="return processPopupForm(this)">
<h2>Tambah Pengguna Baru</h2><hr>
<br><div class="popup-error"></div><div
class="popup-success"></div><input
type="text" name="id" placeholder="ID"
autocomplete="off" id="x" />
```

```
<input type="text" name="nama"
placeholder="Nama Lengkap"
autocomplete="off" id="y" /> <textarea
name="alamat" placeholder="Alamat"
autocomplete="off" id="z"></textarea>
<button type="submit" class="popup-
tambah">Tambah</button></form>
</div>
<script type="text/javascript">
var ignoreClickOnMeElement =
document.getElementsByClassName('popup')[
0];
document.querySelector('.popup-
background').addEventListener('click',
function(event) {
```



```

document.querySelector(".popup-
error").style.display = "block";
document.querySelector(".popup-
error").textContent = "Field tidak boleh
kosong.";
break; }
else {var req = []for(var i = 0; i <
data.elements.length - 1; ++i)
req.push(data.elements[i].value); req =
req.join(",");var xml = new
XMLHttpRequest();
xml.onload = function() {if(this.status == 200) {
var res = this.responseText; var lerr =
document.querySelector(".popup-error");
var lsuc = document.querySelector(".popup-
success");

```

```

if(res == 204) {lerr.style.display = "block";
lsuc.style.display = "none"; lerr.textContent =
"Data gagal ditambahkan ulangi!"; }
else if(res == 205) {lerr.style.display =
"block";
lsuc.style.display = "none";lerr.textContent =
"ID sudah terdaftar, coba dengan ID
berbeda.";

else {lerr.style.display =
"none";lsuc.style.display =
"block";lsuc.textContent = "Data berhasil
ditambahkan";processTable(res);
document.querySelector("#x").value = "";
document.querySelector("#y").value = "";
document.querySelector("#z").value = "";
document.querySelector("#a").value = "";
document.querySelector("#b").value = "";
}}}

```

Lampiran 4. Program Menambah Data pada Website

```
<?php

session_start();

$db = new PDO("mysql:host=localhost;dbname=sistempalang", "root", "");
$db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

if ($_SERVER['REQUEST_METHOD'] == "POST") {
    $data = file_get_contents('php://input');
    $data = explode(",", $data);
    $x = trim($data[0]);
    $y = trim($data[1]);
    $z = trim($data[2]);
    $a = trim($data[3]);
    $b = trim($data[4]);

    $sql = "SELECT * FROM pelanggan WHERE id = ?";
    $stmt = $db->prepare($sql);
    $stmt->bindValue(1, $x, PDO::PARAM_STR);
    $stmt->execute();
    if(!empty($stmt->fetch(PDO::FETCH_ASSOC))) {
        echo 205;
    }
    else {
        try {
            $sql = "INSERT INTO pelanggan (id,nama,alamat) VALUES(?,?,?)";
            $stmt = $db->prepare($sql);
            $stmt->bindValue(1, $x, PDO::PARAM_STR);
            $stmt->bindValue(2, $y, PDO::PARAM_STR);
            $stmt->bindValue(3, $z, PDO::PARAM_STR);
            $stmt->execute();
            if($stmt->rowCount() > 0) {
                $sql = "SELECT * FROM pelanggan";
                $stmt = $db->query($sql);
                $ret = $stmt->fetchAll(PDO::FETCH_ASSOC);
                $strer = ["u" => strtoupper($_SESSION['loginsession']), "d" =>
                $ret];
                echo json_encode($strer);
            }
            else {
                echo 204;
            }
        } catch(Exception $e) {
            echo 204;
        }
    }
}

$db = null;

?>
```

Lampiran 5. Program Login pada Website

```
<?php

session_start();

$db = new PDO("mysql:host=localhost;dbname=sistempalang", "root", "");
$db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

if ($_SERVER['REQUEST_METHOD'] == "POST") {
    $data = file_get_contents('php://input');

    $data = explode(",", $data);

    $u = trim($data[0]);
    $p = trim($data[1]);

    $sql = "SELECT * FROM admin WHERE username = ?";
    $stmt = $db->prepare($sql);
    $stmt->bindValue(1, $u);
    $stmt->execute();

    if($stmt->rowCount() > 0) {
        $sdata = $stmt->fetch();
        if($p == $sdata['password']) {
            $_SESSION['loginsession'] = $sdata['fullname'];
            $sql = "SELECT * FROM pelanggan";
            $stmt = $db->query($sql);
            $ret = $stmt->fetchAll(PDO::FETCH_ASSOC);
            $strer = ["u" => strtoupper($_SESSION['loginsession']), "d" => $ret];
            echo json_encode($strer);
        }
        else {
            echo 204;
        }
    } else {
        echo 204;
    }
}

$db = null;

?>
```


Lampiran 6. Program Menampilkan Data Database pada Website

```
<?php

$db = new PDO("mysql:host=localhost;dbname=sistempalang", "root", "");
$db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

if($_SERVER['REQUEST_METHOD'] == "POST") {
    $data = $_POST['data'];
    $data = explode("//", $data);
    $id = $data[0];
    $pin = $data[1];
    $skurang = (int) $data[2];
    $sql = "SELECT sisa_saldo, pin FROM pelanggan WHERE id = ?";
    $stmt = $db->prepare($sql);
    $stmt->bindParam(1, $id);
    if($stmt->execute()) {
        if($stmt->rowCount() > 0) {
            $res = $stmt->fetch(PDO::FETCH_ASSOC);
            $jatah = $res['sisa_saldo'] - $skurang;
            if($jatah < 0) {
                echo "saldo_tidak_cukup";
            }
            else if($res['pin'] != $pin) {
                echo "pin_salah";
            }
            else {
                $sql = "UPDATE pelanggan SET sisa_saldo = ? WHERE id = ?";
                $stmt = $db->prepare($sql);
                $stmt->bindParam(1, $jatah);
                $stmt->bindParam(2, $id);
                $stmt->execute();
                echo "berhasil";
            }
        }
        else {
            echo "id_tidak_ditemukan";
        }
    }
}

$db = null;
```

Lampiran 7. Program mengubah database menjadi file JSON

```
<?php
    //open connection to mysql db
    $connection = mysqli_connect("localhost","root","","sistempalang") or die("Error " .
mysqli_error($connection));

    //fetch table rows from mysql db
    $sql = "SELECT * FROM pelanggan";
    $result = mysqli_query($connection, $sql) or die("Error in Selecting " . mysqli_error($connection));

    //create an array
    $smparray = array();
    while($row =mysqli_fetch_assoc($result))
    {
        $smparray[] = $row;
    }
    header('Content-Type: application/json; charset=utf-8');
    echo json_encode($smparray);
    mysqli_close($connection);
?>
```