# LAPORAN HASIL PRAKTIKUM PEMPROGAMAN WEB & MOBILE



NAMA : YUDHA ARTHA NUGRAHA

NIM : 193030503045

KELAS : A

**MODUL** : III PHP MySQL Database

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021

#### **BAB I**

# TUJUAN DAN LANDASAN TEORI

### **TUJUAN**

- Mahasiswa mampu membuat program yang bisa menyimpan data dalamjumlah yang banyak.
- Mahasiswa mampu membuat program yang bisa mengolah data yang tersimpan dalam database.

Data dalam database MySQL disimpan dalam tabel-tabel. Sebuah tabel adalah koleksi dari data yang berelasi dan mengandung kolom dan baris. Database sangat bermanfaat untuk menyimpan informasi secara kategori. Contoh yang akan diberikan pada modul praktikum ini adalah tabel yang mengandung data Employees (Pekerja), Products (Produk), Customers (Pelanggan) dan Orders (Pesanan).

## Membuka Koneksi

Sebelum mengakses data dalam database MySQL, kita harus terhubung ke server database MySQL. Berikut adalah contoh kode program agar terhubung dengan server mySQL:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Membuat hubungan
$conn = new mysqli($servername, $username, $password);

?>
```

Jika kode diatas tidak berhasil, kemungkinan variabel \$connect\_error sudah tidaktersedia pada versi PHP yang kita gunakan, maka gunakan kode program berikut:

```
<?ph
р
     $servername = "localhost";
     $username = "username";
     $password = "password";
     // Membuat Hubungan
     $conn = mysqli_connect($servername, $username,
     $password);
     //
     Memeriks
     Hubungan
     if
     (!$conn){
            die("Connection failed: ".
            mysqli_connect_error());
     }
     echo "Connected successfully";
```

Ketika selesai menggunakan data dari database, sebaiknya koneksi atau hubungan ke server ditutup, caranya dengan menggukan kode program berikut:

```
mysqli_close($conn);
```

### **Membuat Database**

Database pada MySQL bisa juga dibuat menggunakan kode program PHP. Program tersebut akan berisi statement SQL "CREATE DATABASE". Berikut adalah contoh program yang digunakan untuk membuat database "myDB":

```
<?ph
    $servername = "localhost";

$username = "username";

$password = "password";

// Membuat Koneksi

$conn = mysqli_connect($servername, $username, $password);

//
Memeriksa
Kanakaiif
</pre>
```

```
echo "Database created successfully";
} else {

echo "Error creating database: " .

mysqli_error($conn);

?>
```

### **Membuat Tabel**

Pembuatan tabel pada bahasa pemrograman PHP juga menggunakan

statement SQL, yaitu statement "CREATE TABLE". Contoh yang akan diberikan adalah pembuatan tabel MyGuests. Statement pembuatan tabel MyGuests adalah sebagai berikut:

```
CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY

KEY, firstname VARCHAR(30) NOT NULL, lastnameVARCHAR(30) NOT

NULL,

email VARCHAR(50),

reg_date TIMESTAMP
)
```

Berikut adalah contoh program untuk membuat tabel diatas:

```
<?ph
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "myDB";

// Membuat Koneksi
$conn = mysqli_connect($servername, $username, $password, $dbname);
//</pre>
```

```
id INT(6) UNSIGNED AUTO_INCREMENT
PRIMARY

KEY, firstname VARCHAR(30) NOT NULL,
lastnameVARCHAR(30) NOT NULL,
email VARCHAR(50),
reg_date TIMESTAMP
)";

if (mysqli_query($conn, $sql)) {
?>
```

#### Memasukkan Data ke Database

Untuk memasukkan data kedalam tabel di database, terdapat beberapa aturan sintaks yang harus diikuti:

- Query SQL harus diberikan kutip dalam PHP.
- Nilai string didalam query SQL harus diberikan kutip.
- Nilai numeris tidak harus diberikan kutip.
- Nilai NULL tidak harus diberikan kutip.

Statement INSERT INTO digunakan untuk menambahkan baris pada tabelMySQL, query untuk menambahkan data bersebut adalah sebagai berikut:

```
INSERT INTO table_name (column1, column2, column3,...)VALUES (value1, value2, value3,...)
```

adapun contoh kode program PHP untuk menambahkan data tersebut adalah sebagai berikut:

```
<?ph
  $servername =
  "localhost";
  $username =</pre>
```

```
// Membuat Koneksi
      $conn = mysqli_connect($servername, $username, $password,
      $dbname);
      // Memeriksa Koneksiif (!$conn){
              die("Connection failed: " . mysqli_connect_error());
       }
      $sql = "INSERT INTO MyGuests (firstname, lastname, email)
       VALUES ('John', 'Doe', 'john@example.com')";
      if (mysqli_query($conn, $sql)) {
              echo "New record created successfully";
       } else {
              echo "Error: " . $sql . "<br>" . mysqli_error($conn);
       }
      mysqli_close($conn);
?>
```

Jika kita melakukan perintah INSERT pada tabel yang menggunakan field dengan opsi AUTO\_INCREMENT, kita bisa mendapatkan ID dari baris yang terakhir diinputkan. Caranya menggunakan fungsi mysqli\_insert\_id(\$conn); nilai kembalian dari fungsi ini akan memberikan id dari record atau baris terakhir yang diinputkan.

Untuk menambahkan record dengan jumlah yang banyak, kita bisa menggunakan fungsi mysqli\_multi\_query(\$conn, \$sql), berikut adalah contoh program untuk menambahkan record dengan jumlah yang banyak:

```
<?ph
    $servername = "localhost";

$username = "username";

$password = "password";

$dbname = "myDB";</pre>
```

### Mengambil Data dari Database

Untuk mengambil data, statement SQL yang digunakan adalah SELECT nama\_kolom() FROM nama\_tabel, atau kita bisa menggunakan karakter \* untuk memilih semua kolom yang ada pada tabel. Pengambilan data dengan kriteria tertentu bisa dilakukan dengan menggunakan statement WHERE setelah nama\_tabel. Adapun contoh kode program untuk mengambil data dari database adalah sebagai berikut:

```
<?ph
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "myDB";

// Membuat koneksi
    $conn = mysali_connect($servername_$username_$nassword)</pre>
```

```
$sql = "SELECT id, firstname,
lastname FROM MyGuests"; $result
= mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {

    // Menampilkan data pada
    setiap baris while($row =
    mysqli_fetch_assoc($result)) {

        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " "
        .

$row["lastname"]. "<br>";
```

# Menghapus Data dari Database

Statement DELETE digunakan untuk menghapus baris data atau records dari tabel. Clause WHERE digunakan untuk menspesifikasikan baris yang akan dihapus. Jika statement DELETE digunakan tanpa menggunakan clausa WHERE, maka semua recordyang ada pada tabel akan dihapus.

id	namadepan	namabelakang	email	reg_date
1	John	Doe	john@example.com	2014-10-22 14:26:15
2	Mary	Moe	mary@example.com	2014-10-23 10:22:30
3	Julie	Dooley	julie@example.com	2014-10-26 10:48:23

Jika kita memiliki data seperti tabel "MyGuest" seperti terlihat diatas, kemudian kita ingin menghapus data dengan nama depan Julie, maka kode program untuk menghapus data tersebut adalah sebagai berikut:

```
<?ph
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "myDB";</pre>
```

```
//
Memeriksa
koneksiif
(!$conn){
    die("Connection failed: " .
    mysqli_connect_error());
}

// sql untuk menghapus record
$sql = "DELETE FROM MyGuests WHERE id=3";
?>
```

# **Update Data dalam Database**

Untuk melakukan perubahan data di dalam database, statement UPDATE digunakan, yaitu sebagai berikut:

```
UPDATE nama_tabel

SET kolom1=nilai1, kolom2=nilai2, ... WHERE kolom_penentu=nilai_penentu ...
```

Jika statement ini digunakan untuk update data ke dua pada tabel "MyGuest" diatas untuk mengganti nama belakang dari "Moe" menjadi "Doe", maka kode program untuk update tersebut adalah sebagai berikut:

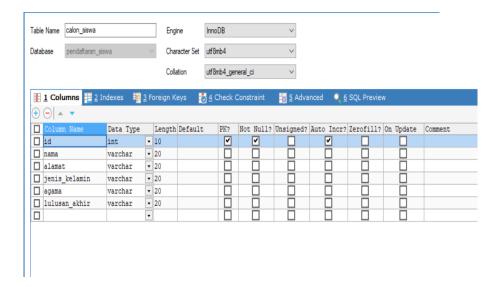
```
<?ph
    $servername = "localhost";
    $username = "username";
    $password = "password";
    $dbname = "myDB";</pre>
```

#### **BAB II**

#### **PEMBAHASAN**

# Keterangan:

- config.php untuk menyimpan konfigurasi database;
- index.php halaman utama;
- list-pegawai.php halaman untuk menampilkan data siswa;
- form-daftar.php halaman formulir pendaftaran;
- proses-pendaftaran.php skrip yang memproses pendaftaran;
- form-edit.php form untuk edit data siswa;
- proses-edit.php skrip untuk memproses edit/update;
- hapus.php skrip untuk menghapus data dari database;



Gambar 2.1 database

Pada bagian pertama diperlukan databse yanga akan dihubungkan dengan website yang akan dibuat, database akan digunakan sebagai penyimpanan data pada website yang akan terkoneksi dengan website.

```
1  <?php
2
3  $server = "localhost";
4  $user = "root";
5  $password = "";
6  $nama_database = "data_pegawai";
7
8  $db = mysqli_connect($server, $user, $password, $nama_database);
9
10  if( !$db ){
11     die("Gagal terhubung dengan database: " . mysqli_connect_error());
12  }
13
14  ?>
```

Gambar 2.2 config.php

Pada config.php ini digunakan untuk membuat koneksi dengan databse, dengan begitu data yang dirubah dalam website berpengaruh kepada database begitu pula sebaliknya. Pada baris 3-6 digunakan untuk membuat variable yang akan dipakai dalam baris ke 8 dan jika tidak berhasil akan menampilkan tulisan gagal terhubung dengan database fungsi gagal terdapat dalam baris ke 10-12

Gambar 2.3 index.php

Pada file index.php digunakan untuk menampilkan halaman awal yang kita anggap sebagai index. Pada bagian ini digunkana untuk menampilkan halaman yang berisi 2 pilihan option yaitu daftar pegawai dimana untuk hyperlink ke tambah data dan semua pegawai terdaftar dimana hyperlink ke list pegawai. Pada baris 4 digunakan untuk memberi nama halaman, pada baris 9-13 digunkana untuk menampilkan header tulisan, pada baris 16-17 digunakan untuk pilihan hyperlink untuk menambah data atau melihat data terdaftar, pada baris 20-36 digunakan untuk memberikan info bahwa data telah ditambahkan atau data gagal di tambahakan.

```
| Control | Cont
```

Gambar 2.4 form-daftar.php

Pada file form-daftar.php merupakan php untuk menambah data, dimana data yang ditambahkan menggunakan php ini juga akan menambahkan data di database. Pada bagian ini lebih tepatnya digunakan untuk menginputkan data yang akan durubah dan akan memberikan inputan ke file proses-pendaftaran.php di file selanjutnya. Pada baris 1-10 digunakan untuk memberi nama halaman, judul halaman dan header halaman. Pada baris ke 12 digunakan untuk memberikan hasil inputan ke file proses-pendaftaran.php, pada baris ke 14-48 digunakan untuk memunculkan tampilan dari data yang akan diinputkan dan data yang akan di tangkap dimana data yang akan di tanggkap yaitu nama, alamat, jenis kelamin, agama dan lulusan akhir dimana form yang dipakai menggunkaan fieldset yang mana pinggiran dari tampilan akan ada kotak, pada baris 52-54 digunakan untuk menambahkan hyperlink ke menu utama atau halaman index dan juga merupkana penutup dari program

Gambar 2.5 proses-pendaftaran.php

Pada file proses-pendaftaran.php ini digunakan untuk menambah data ke dalam database. Pada php ini digunakan untuk memenambah kan data berdasarkan data yang diterima di form-daftar.php jadi disimpulkan jika form-daftar.php digunakan untuk mengambil data yang di input dan prosespendaftaran.php digunakan untuk menambahkan data. Pada baris 1-6 digunakan

untuk menghubungkan koneksi dan cek apakah tombol telah di klik, pada baris 8-17 digunakan untuk merubah data yang telah diterima dari form-daftar dimana data yang diteriman nama, alamat, jenis kelamin, agama, dan lulusan setelah itu terdapat query yang digunakan untuk menambahkan data pada database, pada bari 19-32 digunakan untuk memberikan info jika tambah data berhasil, gagal dan akses dilarang

Gambar 2.6 hapus.php

Pada file hapus.php digunakan untuk menghapus data pada table yang ada di dalam database yang digunakan dimana data yang di hapus bisa dipilih berdasarkan variable id diatas. Pada baris 1-5 digunakan untuk menghubungkan konektivitas dan menerima query, pada baris 11-12 digunakan query untuk menghapus data berdasarkan id yang dipilih dan data yang dihapus diambil berdasarkan database yang dipilih, pada baris 15-25 digunakan untuk memberikan info bahwa proses penghapusan data gagal.

```
// kalau tidak ada id di query string
if( !isset($_GET['id']) ){
   header('Location: list-pegawai.php');
       //ambil id dari query string
$id = $_GET['id'];
        // jika data yang di-edit tidak dite
if( mysqli_num_rows($query) < 1 ){
    die("data tidak ditemukan...");</pre>
      <header>
     <h>> header>
     <h>> Formulir Edit pegawai</h3>
</header>
                     <form action="proses-edit.php" method="POST">
                                action="proses-edit.php" method="POST">
                          <abel for="nama">Nama: </label>
    <iapput type="text" name="nama" placeholder="nama lengkap" value="<?php echo $pegawai['nama'] ?>" />
                                      <label for="alamat">Alamat: </label>
                                                   out type="text" name="alamat" value= <?php echo $pegawai['alamat'] ?> />
                        checked":
checked:
chec
                                     <label for="lulusan_akhir">lulusan_akhir: </label>
                                    <input type="text" name="lulusan_akhir" placeholder="lulusan terakhir kali" value="<?php echo $pegawai['</pre>
                                                  lulusan_akhir'] ?>" />
                                      <input type="submit" value="Simpan" name="simpan" />
                          <a href="index.php">kembali ke menu</a>
```

Gambar 2.7 form-edit.php

Pada file form-edit.php ini digunkana untuk membuat perubahan pada isi table yang telah di dapatkan sehingga tidak perlu menghapus dan menambahkan berulang melainkan tinggal mengedit saja. Pada baris 1-23 digunakan untuk membuat konektivitas, mengambil data dari list-pegawai, dan memberikan info jika data gagal ditemukan, pada baris 26 -81 digunakan untuk menampilkan tampilan perubahan data yang di ambil dari array id yang dipilih dimana tampilan ini memiliki coding yang sama dengan tambah data namun dengan rekaman data yang sebelumnya telah tertulis segingga tinggal merubah saja. Pada baris 81-85 digunakan untuk membuat hyperlink ke halaman utama atau halaman index dan juga penanda berakhirnya program dari file form-edit.php

```
include("config.php");

include("config.php");

// cek apakah tombol simpan sudah diklik atau blum?

if(isset(s_POST['simpan'])){

// ambil data dari formulir

sid = $_POST['ald'];

Snama = $_POST['alman'];

slamant = $_POST['alman'];

slamant = $_POST['alman'];

slamant = $_POST['alman'];

// bust query update

saql = "UPDATE data_pagawai SET naman='Snama', alamat-'Salamat', jenis_kelamin-'Sjk', agama='Sagama', lulusan_akhir-
'Sulusan_akhir' MHERE id-$id-$id';

squery = mysqli_query(sdb, $sql);

// apakah query update berhasil?

if( $query );

// apakah query update berhasil?

if( $query );

// kalau berhasil alihkan ke halaman list-pegawai.php
header('Location: list-pegawai.php');
} else {

// kalau gagal tampilkan pesan
die('Gagal menyimpan perubahan...");
}

} else {

die("Akses dilarang...");
}

}

**Post['alman']

**
```

Gambar 2.8 proses-edit.php

Pada proses-edit.php ini digunakan untuk melakukan perubahan pada database data yang di ambil untuk merubah di ambil dari form-edit.php, jadi disimpulkan bahwa form-edit.php untuk mengambil data yang akan dirubah dan proses-edit.php digunkan untuk merubahnya. pada baris 1-6 digunakan untuk membuat konektifitas dan cek tombol telah terpicik, pada baris 8-14 digunakan untuk membuat variable yang dapat menerima input dari form-edit.php, pada baris ke 17-18 digunakan untuk memberi fungsi update atau query update

dimana variable yang di ambil dari baris di atasnya, pada baris 20-34 digunakan untuk memberikan info bahwa gagal atau jika berhasil akan di alihkan ke halaman list pegawai

```
de("config.php"); ?>
      tle>Pendaftaran pegawai | kantor Coding</title>
iy>
<header>
<h3>pegawai yang sudah terdaftar</h3>
-/baader>
      <a href="form-daftar.php">[+] Tambah Baru</a>
av>
      ble border="1">
            /
No
No
Nama
Alamat
Jenis Kelamin
             Agama
Agama
Sekolah Asal
Tindakan
      <?php
$sql = "SELECT * FROM data_pegawai";
$query = mysqli_query($db, $sql);</pre>
      while($pegawai = mysqli_fetch_array($query)){
    echo "";
            echo "".$pegawai['id']."";
echo "".$pegawai['nama']."";
echo "".$pegawai['alamat']."";
echo "".$pegawai['jenis_kelamin']."";
echo "".$pegawai['agama']."";
echo "".$pegawai['lulusan_akhir']."";";
            echo "";
echo "<a href='form-edit.php?id=".$pegawai['id']."'>Edit</a> | ";
echo "<a href='hapus.php?id=".$pegawai['id']."'>Hapus</a>";
echo "";
             echo "";
Total: <?php echo mysqli_num_rows($query) ?>
<a href="index.php">kembali ke menu</a>
```

Gambar 2.9 list-pegawai.php

Pada file list-pegawai.php ini digunkana untuk menampilkan data dari dari database, dimana data yang dipanggil adalah data data pegawai yang diambil dari

data\_pegawai. Pada baris 1-14 digunakan untuk memberi nama halaman dan header, pada baris ke 15 digunakan untuk hyperlink ke tambah data, pada baris 20-32 digunakan untuk memberi nama judul dari kolom dan baris nanti, pada baris 34-55 merupakan php yang digunakan untuk query menampilkan data dari isi table dan juga fungsi hyperlink ke edit data dan hapus data, pada baris 60 digunakan untuk menghitung jumlah data, pada baris 63-65 digunakan untuk hyperlink ke index.php dan penutup dari akhir program pada list-pegawai.php

# Pendaftaran pegawai baru

# kantor coding

#### Menu

- <u>Daftar pegawai Baru</u>
- semua pegawai terdaftar

### Gambar 2.10 halaman awal

Pada gambar di atas merupakan tampilan dari website yang ada di halaman awal dan di halaman ini ada dua hyperlink dimana untuk menambah data dan melihat isi data dari seluruh pegawai terdaftar



Gambar 2.11 halaman tambah data

gambar di atas ini merupakan halaman untuk menambah kan data pada database dimana data yang dimasukna adalah nama ,alamat ,jenis ,kelamin, agama dan lulusan akhir

# pegawai yang sudah terdaftar

[+] Tambah Baru

No	Nama	Alamat	Jenis Kelamin	Agama	Sekolah Asal	Tindakan
1	nama1	alamat1	laki-laki	Islam	universitas1	Edit   <u>Hapus</u>
2	nama2	alamat2	perempuan	Islam	universitas2	Edit   Hapus
4	nama4	alamat4	perempuan	khatolik	universitas4	Edit   Hapus
5	nama5	alamat5	perempuan	Kristen	universitas5	Edit   Hapus

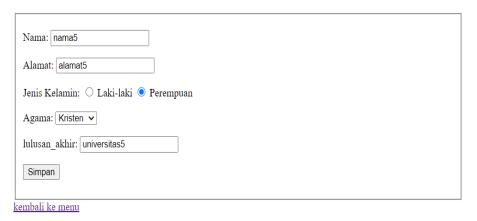
Total: 4

kembali ke menu

Gambar 2.12 halaman list pegawai terdaftar,edit,hapus

Pada gambar di atas merupakan halaman dari list pegawai yang dimana terdapat hyperling di table terakhir yang berisi edit dan tambah, pada halaman ini akan menampilkan semua data yang ada di database

# Formulir Edit pegawai



Gambar 2.13 halaman edit data

Pada gambar di atas merupakan halaman untuk merubah data yang ada pada list- pegawai dimana kita tidak perlu menambah dan menghapus secara manual melainkan dengan opsi edit

### **BAB III**

# **KESIMPULAN**

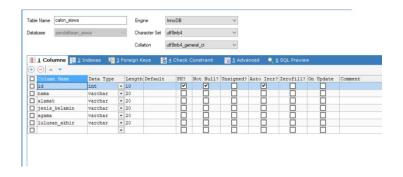
PHP sering dijodohkan dengan MySQL untuk membangun aplikasi web. MySQL berperan sebagai tempat menyimpan data (basis data) sedangkan PHP adalah program yang mengelola data tersebut.

Sebenarnya PHP tidak hanya bisa dijodohkan dengan MySQL saja. PHP juga bisa dijodohkan dengan SQLite, PostgreSQL, MongoDB, Ms. Access, dsb. Pada laporan di atas kita menggunakan untuk menghubungkan database dan webseite yang telah di buat

#### DAFTAR PUSTAKA

- Muhardian A. Tutorial PHP dan MySQL: Membuat Aplikasi CRUD [Studi Kasus Pendaftaran Siswa Baru]. Petani Kode. Published June 10, 2017. Accessed April 12, 2021. https://www.petanikode.com/tutorial-php-mysql/
- PHP Tutorial. W3schools.com. Published 2021. Accessed April 12, 2021. https://www.w3schools.com/php/default.asp
- Modul Praktikum Pemrograman Web & Mobile I. Jurusan Teknik Informatika. Fakultas Teknik. Universitas Palangka Raya. 2021.

# **LAMPIRAN**



Gambar 2.1 database

```
1  <?php
2
3  $server = "localhost";
4  $user = "root";
5  $password = "";
6  $nama_database = "data_pegawai";
7
8  $db = mysqli_connect($server, $user, $password, $nama_database);
9
10  if( !$db ){
11     die("Gagal terhubung dengan database: " . mysqli_connect_error());
12  }
13
14  ?>
```

Gambar 2.2 config.php

```
1 closervit html>
2 chad2
3 chad3
4 citial>Pendaftaran pagawal | kantor Coding(/title>
4 citial>Pendaftaran pagawal | kantor Coding(/title>
5 chad3
6 chad3
7 chad3
7 chad3
8 chad3
8 chad3
10 chad3
11 chad3
12 chad3
13 chad3
14 chad3
15 chad3
16 chad3
17 chad3
18 chad3
19 chad3
10 c
```

Gambar 2.3 index.php

```
| clockyre html> | clockyre html | clock
```

Gambar 2.4 form-daftar.php

Gambar 2.5 proses-pendaftaran.php

Gambar 2.6 hapus.php

```
| String | S
```

Gambar 2.7 form-edit.php

Gambar 2.8 proses-edit.php

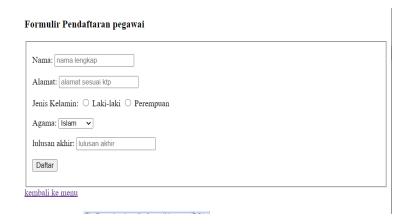
Gambar 2.9 list-pegawai.php

Pendaftaran pegawai baru

# kantor coding

- Daftar pegawai Baru
   semua pegawai terdaftar

Gambar 2.10 halaman awal

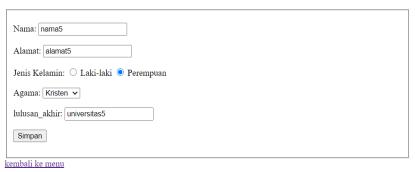


Gambar 2.11 halaman tambah data



Gambar 2.12 halaman list pegawai terdaftar,edit,hapus

# Formulir Edit pegawai



Gambar 2.13 halaman edit data