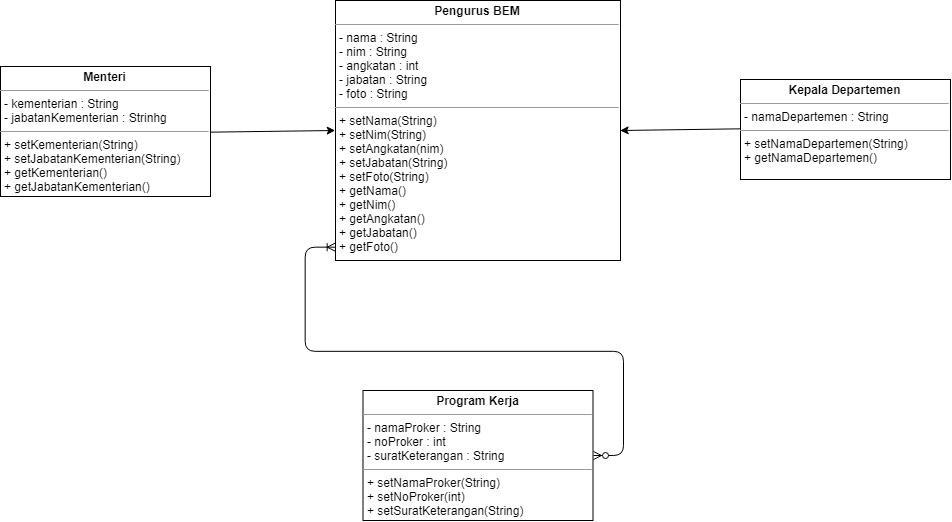
**LATIHAN 1**

1. **Soal**

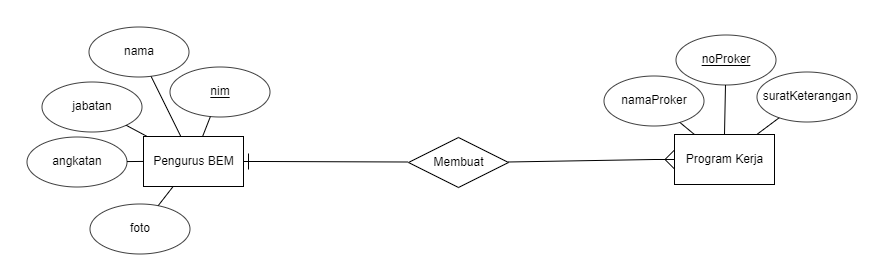
Buatlah pemodelan class diagram dan *entity relationship diagram* sesuai dengan analisis dan perancangan sistem yang sudah dibuat.

1. **Screenshoot**

* Class Diagram



* Entity Relationship Diagram



1. **Syntax**

|  |  |
| --- | --- |
|  |  |

1. **Penjelasan**

Class Pengurus BEM memiliki atribut dan method seperti pada diagram, kemudian class Pengurus BEM memiliki child class menteri dan Kepala Departemen yang memiliki atribut dan method seperti tersebut, kemudian terdapat juga class Program Kerja yang memiliki atribut dan method seperti tersebut.

Pada ER Diagram terdapat dua entitas yaitu Pengurus BEM dan juga Program Kerja, yang memiliki atribut seperti pada diagram, dengan relasi Pengurus BEM dapat membuat banyak Program Kerja.

**LATIHAN 2**

1. **Soal**

Buatlah folder baru dalam folder htdocs dan beri nama oop, seluruh kode Latihan 2 ini disimpan dalam folder oop.

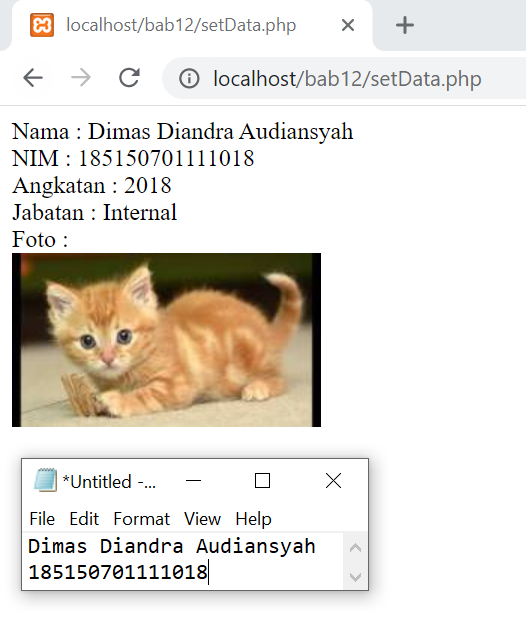
Membuat pemodelan struktur pengguna sistem dengan menuliskan kode di bawah ini dan simpan dalam file pengurusBEM.php

|  |  |
| --- | --- |
|  | <?php  class pengurusBEM {  public $nama;  public $nim;  public $angkatan;  private $jabatan;  private $foto;  public function setNama($nama) {  $this->nama = $nama;  }  public function getNama() {  echo ”$this->nama”;  }  private function setJabatan($jabatan) {  $this->jabatan = $jabatan;  }    public function getJabatan() {  echo ”$this->jabatan”;  }  } |

Tulis kode di bawah ini dan simpan dalam file menteri.php

|  |  |
| --- | --- |
|  | <?php  include ”pengurusBEM.php”  class menteri extends pengurusBEM {    public function \_\_construct($nama){  $this->nama = $nama;  }  function setJabatan($jabatan) { // method  $this->jabatan = $jabatan;  }  public function getJabatan() {  echo ”$this->jabatan”;  }  } |

1. Apa maksud dari penulisan public dan private pada *class* pengurusBEM?
2. Modifikasilah kode diatas sehingga kita dapat melakukan manipulasi attribut dari pengurus BEM dengan lengkap.
3. Apa hubungan antara method setJabatan pada class pengurusBEM dan class Menteri? Apakah method yang sama karena konsep turunan?
4. Apa yang terjadi saat dalam pembuatan class kita tidak menuliskan tingkat visibility? Apakah method atau attribut dengan semua tingkat *visibility* itu diturunkan?
5. **Screenshoot**



*tampilan halaman setelah melakukan modifikasi pada file pengurusBEM.php dan mengisikan datanya pada file setData.php*

1. **Syntax**

|  |  |
| --- | --- |
|  | pengurusBEM.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  2122  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50 | <?php  class pengurusBEM {  public $nama;  public $nim;  public $angkatan;  private $jabatan;  private $foto;  public function setNama($nama) {  $this->nama = $nama;  }  public function getNama() {  echo "Nama : $this->nama<br>";  }  public function setJabatan($jabatan) {  $this->jabatan = $jabatan;  }  public function getJabatan() {  echo "Jabatan : $this->jabatan<br>";  }  public function setNIM($nim) {  $this->nim = $nim;  }  public function getNIM() {  echo "NIM : $this->nim<br>";  }  public function setAngkatan($angkatan) {  $this->angkatan = $angkatan;  }  public function getAngkatan() {  echo "Angkatan : $this->angkatan<br>";  }  public function setFoto($foto) {  $this->foto = $foto;  }  public function getFoto() {  echo "Foto :<br><img src=$this->foto><br>";  }  } |

|  |  |
| --- | --- |
|  | menteri.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | <?php  include "pengurusBEM.php";  class menteri extends pengurusBEM {  public function \_\_construct($nama){  $this->nama = $nama;  }  function setJabatan($jabatan) { // method  $this->jabatan = $jabatan;  }  public function getJabatan() {  echo "$this->jabatan";  }  } |

|  |  |
| --- | --- |
|  | setData.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | <?php  include "pengurusBEM.php";  $x = new pengurusBEM;  $x->setNama("Dimas Diandra Audiansyah");  $x->getNama();  $x->setNIM("185150701111018");  $x->getNIM();  $x->setAngkatan("2018");  $x->getAngkatan();  $x->setJabatan("Internal");  $x->getJabatan();  $x->setFoto("kucing.jpg");  $x->getFoto(); |

1. **Penjelasan**
2. Maksud dari penulisan public pada class pengurusBEM adalah class tersebut dapat diakses oleh class manapun tanpa membutuhkan method tertentu. Sedangkan maksud dari penulisan private pada class pengurusBEM adalah class tersebut hanya dapat diakses di class itu sendiri dan membutuhkan sebuah method tertentu untuk mengaksesnya.
3. Untuk dapat melakukan manipulasi atribut dari pengurus BEM dengan lengkap maka yang perlu dilakukan adalah menambahkan method set dan get untuk semua atribut yang ada pada file ”pengurusBEM.php”. Kemudian membuat sebuah file dengan nama ”setData.php” yang digunakan untuk mengeksekusi method set dan get dan menampilkannya untuk membuktikan bahwa sintaks berhasil.
4. Hubungan antara method setJabatan pada class pengurusBEM dan pada class menteri adalah pada class menteri method tersebut melakukan override method setJabatan pada class pengurusBEM. Sehingga jika ada objek yang diinisialisasi dari class menteri, maka method tersebut akan di jalankan dan bukan merupakan konsep turunan.
5. Ketika pembuatan class jika tidak menuliskan tingkat visibility, maka class tersebut bersifat default dan mempunyai visibility public, sehingga semua method dan atributnya akan diturunkan ke anaknya.

**LATIHAN 3**

1. **Soal**

Buatlah tabel proker dalam *database* untuk menyimpan data program kerja, dengan struktur sebagai berikut.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Nama Field** | **Tipe Data** | **Ukuran** | **Keterangan** |
| 1 | nomorProgram | Integer | default | Primary key |
| 2 | namaProgram | Varchar | 255 |  |
| 3 | suratKeterangan | Varchar | 100 |  |

Buatlah folder baru dalam folder htdocs dan beri nama mvc, seluruh kode Latihan 3 ini disimpan dalam folder mvc.

Tulis kode dibawah ini dan simpan dalam *file* koneksiMVC.php

|  |  |
| --- | --- |
|  | <?php  // sesuaikan username dan password  $mysqli = new mysqli(’localhost’, ’username’, ’password’, ’praktikumWeb’); |

Tulis kode di bawah ini dan simpan dalam *file* m\_programKerja.php!

|  |  |
| --- | --- |
|  | <?php  require "koneksiMVC.php";  class m\_programKerja {  private $nomorProgram;  private $namaProgram;  private $suratKeterangan;  public $hasil = array();    public function \_\_construct($nomorProgram, $namaProgram, $suratKeterangan) {  $this->nomorProgram = $nomorProgram;  $this->namaProgram = $namaProgram;  $this->suratKeterangan = $suratKeterangan;  }  public function setPogramKerja($nomorProgram, $namaProgram, $suratKeterangan) {  $rs = $mysqli->query("INSERT INTO proker VALUES ('$this->nomorProgram’, '$this->namaProgram’, '$this->suratKeterangan')");  }    public function getSemuaPogramKerja() {  $rs = $mysqli->query("SELECT \* FROM proker");  $rows = array();    while($row = $rs->fetch\_assoc()) {  $rows[] = $row;  }  $this->hasil[] = $rows;  return $this->hasil;  }  } |

Tulis kode di bawah ini dan simpan dalam *file* ”c\_programKerja.php”

|  |  |
| --- | --- |
|  | <?php  include\_once("m\_programKerja");  class c\_programKerja {    public $model;  public function \_\_construct($nama){  $this->model = new m\_programKerja();  }  public function invoke()  $proker = $this->model->getSemuaPogramKerja();  include 'v\_programKerja.php';  }  } |

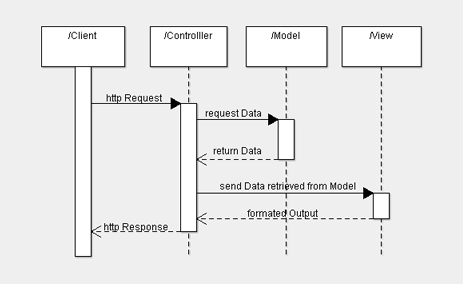
Tulis kode di bawah ini dan simpan dalam *file* ”v\_programKerja.php”

|  |  |
| --- | --- |
|  | <html>  <head></head>  <body>  <h2>Daftar Program Kerja BEM</h2>  <table>  <tbody>  <tr>  <td>No</td>  <td>Nama Program Kerja</td>  <td>Surat Keterangan</td>  </tr>  </tbody>  <?php  echo ‘<table><tr>’,  ‘<td>$proker[nomorProgram]</td>’,  ‘<td>$proker[namaProgram]</td>’,  ‘<td>$proker[suratKeterangan]</td>’,  ‘</tr></table>’;  ?>  </body>  </html> |

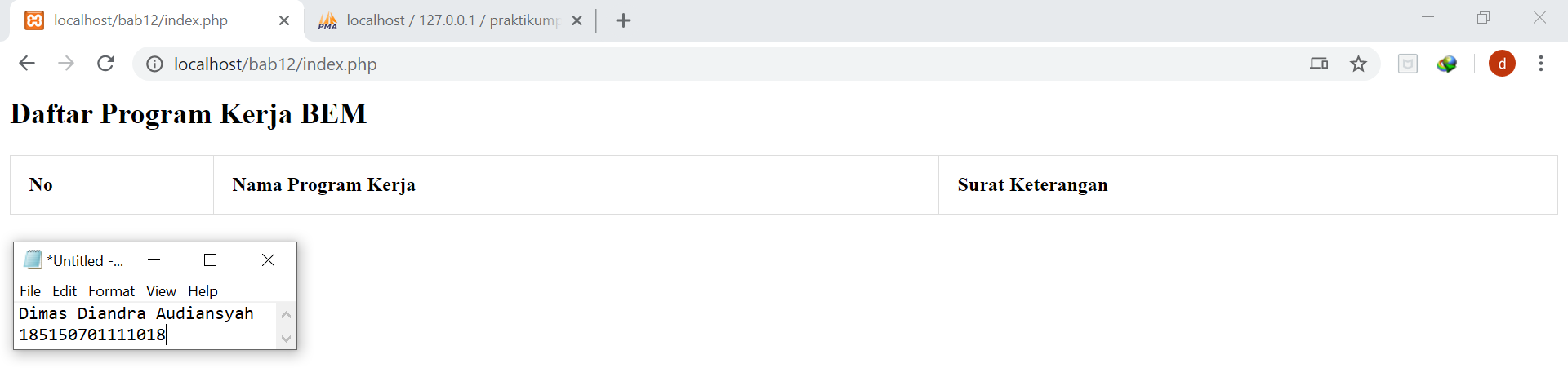
Tulis kode di bawah ini dan simpan dalam *file* ”index.php”

|  |  |
| --- | --- |
|  | <?php  include\_once("c\_programKerja.php");  $controller = new c\_programKerja();  $controller->invoke(); |

1. Jalankan file index.php dan amati apa yang terjadi. Apa hubungannya dengan *sequence diagram* di bawah ini



1. Apa maksud dari pembuatan kode ”m\_programKerja.php”?
2. Apa maksud dari pembuatan kode ”c\_programKerja.php”?
3. Apa maksud dari pembuatan kode ”v\_programKerja.php”?
4. Apa maksud dari pembuatan kode ”index.php”?
5. **Screenshoot**



*menjalankan file index.php*

1. **Syntax**

|  |  |
| --- | --- |
|  | koneksimvc2.php |
| 1  2 | <?php  $mysqli = mysqli\_connect('localhost', 'root', '', 'praktikumweb') or die("Fail to connect"); |

|  |  |
| --- | --- |
|  | m\_programKerja.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  2122  23  24  25  26  27  28  29  30  31  32  33  34  35 | <?php  class m\_programKerja {  private $nomorProgram;  private $namaProgram;  private $suratKeterangan;  public $hasil = array();  public function \_\_construct($nomorProgram, $namaProgram, $suratKeterangan) {  $this->nomorProgram = $nomorProgram;  $this->namaProgram = $namaProgram;  $this->suratKeterangan = $suratKeterangan;  }  public function setPogramKerja($nomorProgram, $namaProgram, $suratKeterangan) {  $rs = $mysqli->query("INSERT INTO proker VALUES ('$this->nomorProgram', '$this->namaProgram', '$this->suratKeterangan')");  }  public function getSemuaPogramKerja() {  require("koneksimvc2.php");  $rs = $mysqli->query("SELECT \* FROM proker");  $rows = array();  while($row = $rs->fetch\_assoc()) {  $rows[] = $row;  }  $this->hasil[] = $rows;  return $this->hasil;  }  } |

|  |  |
| --- | --- |
|  | c\_programKerja.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | <?php  include\_once("m\_programKerja.php");  class c\_programKerja {  public $model;  public function \_\_construct(){  $this->model = new m\_programKerja(3,"Makrab","belum selesai");  }  public function invoke(){  $proker = $this->model->getSemuaPogramKerja();  include 'v\_programKerja.php';  }  } |

|  |  |
| --- | --- |
|  | v\_programKerja.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  2122  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45 | <html>  <head>  <style>  table, td, th {  border: 1px solid #ddd;  text-align: left;  }  table {  border-collapse: collapse;  width: 100%;  }  th, td {  padding: 15px;  }  </style>  </head>  <body>  <h2>Daftar Program Kerja BEM</h2>  <table>  <tbody>  <tr>  <th>No</th>  <th>Nama Program Kerja</th>  <th>Surat Keterangan</th>  </tr>  </tbody>  <?php  $proker = $proker[0];  echo "<table>";  for ($i=0; $i < count($proker); $i++) {  echo "<tr><td>";  echo $proker[$i]["nomorProgram"];  echo "</td><td>";  echo $proker[$i]["namaProgram"];  echo "</td><td>";  echo $proker[$i]["suratKeterangan"];  echo "</td></tr>";  }  echo "</table>";  ?>  </body>  </html> |

|  |  |
| --- | --- |
|  | index.php |
| 1  2  3  4  5 | <?php  include\_once("c\_programKerja.php");  $controller = new c\_programKerja();  $controller->invoke(); |

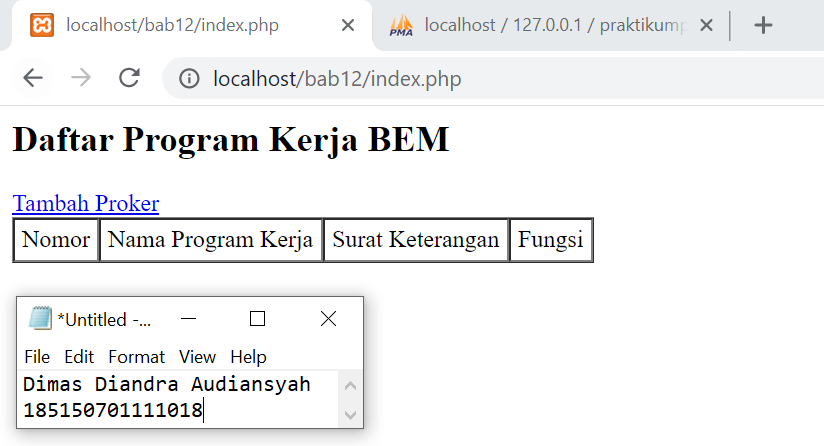
1. **Penjelasan**
2. Saat index.php dijalankan, pertama kali akan memuat controller dari class c\_programKerja.php. Kemudian dari controller tersebut akan memanggil method setProgramKerja untuk membuat model baru, setelah itu ajan memanggil method invoke yang berfungsi untuk memanggil getSemuaProgramKerja dari model m\_programKerja.php sehingga akan mengembalikan array yang berisi semua proker yang ada di database. Terkahir controller akan memanggil view v\_programKerja.php dan melempar array proker tersebut ke view untuk ditampilkan datanya. Flow tersebut sama dengan apa yang ditampilkan pada sequence diagram.
3. Kode m\_programKerja.php merupakan model yang berfungsi untuk membuat suatu objek data atau mengambil data dari database.
4. Kode c\_programKerja.php merupakan controller yang berfungsi untuk membuat model baru atau memanggil model yang sudah ada, kemudian data tersebut akan dilempar ke view tertentu.
5. Kode v\_programKerja.php merupakan view yang berisi kode html atau php yang akan menampilkan data yang telah diproses pada controller.
6. Kode index.php berfungsi sebagai halaman awal yang akan memanggil controller pertama kali.

**LATIHAN 4**

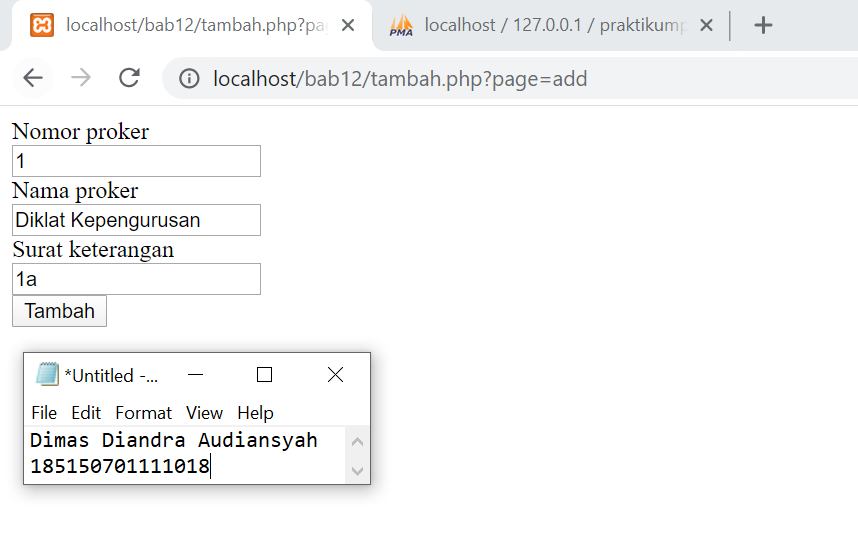
1. **Soal**

Modifikasilah Latihan 3 sehingga dapat dibuat program dengan kemampuan CRUD secara lengkap!

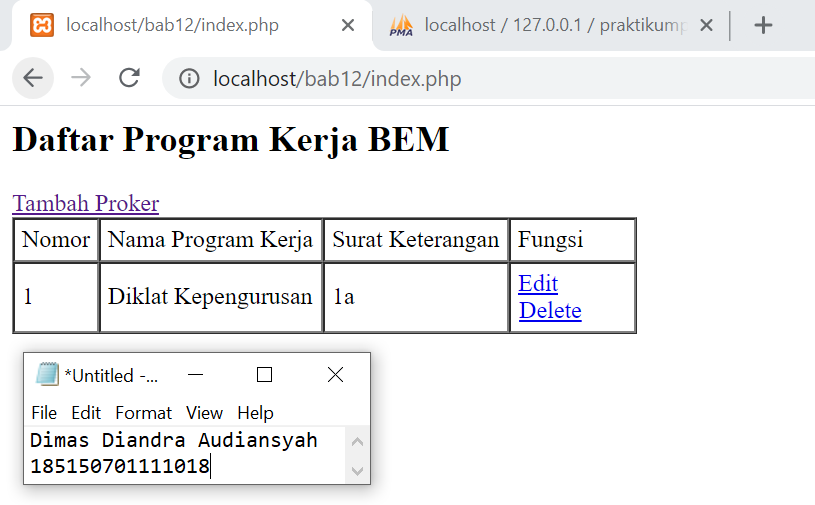
1. **Screenshoot**



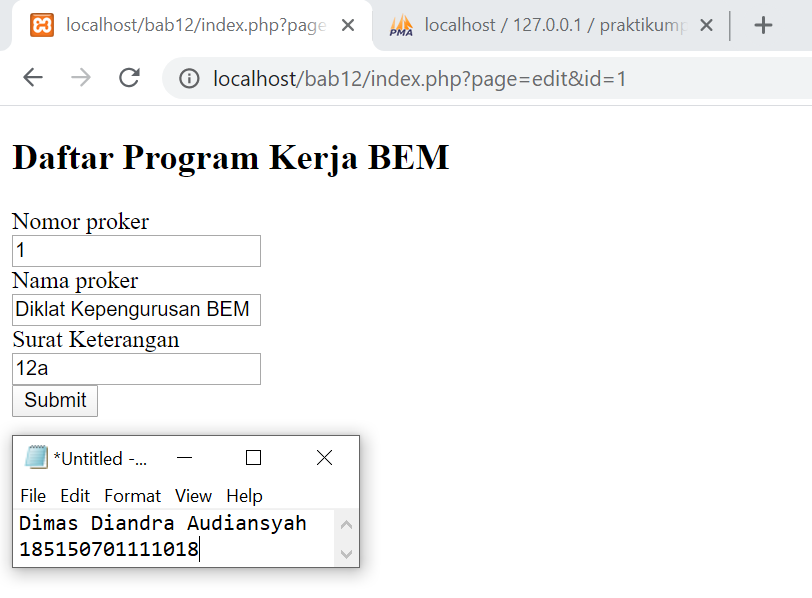
*tampilan ketika menjalankan index.php*



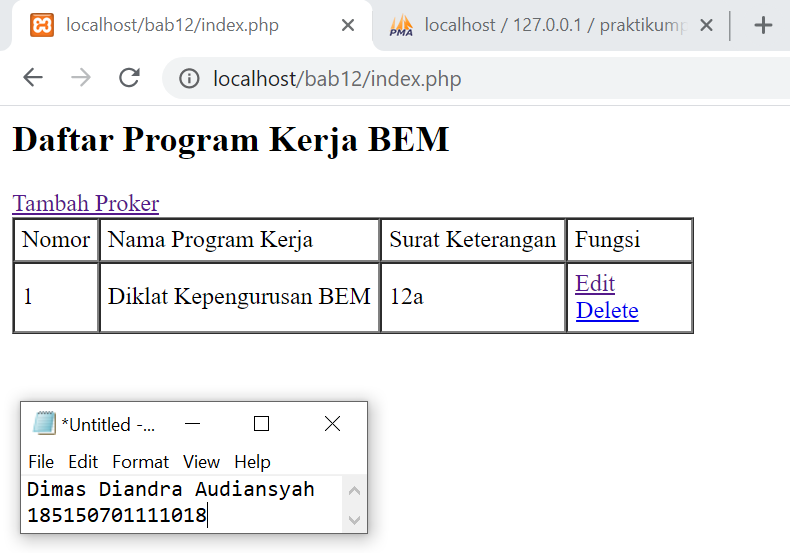
*menambahkan proker baru*



*setelah penambahan proker baru*



*melakukan edit data*



*setelah melakukan edit data*

  
*melakukan tambah data*

**

*melakukan hapus data*

1. **Syntax**

|  |  |
| --- | --- |
|  | koneksimvc2.php |
| 1  2 | <?php  $mysqli = mysqli\_connect('localhost', 'root', 'malang12', 'praktikumpemweb') or die("Fail to connect"); |

|  |  |
| --- | --- |
|  | m\_programKerja.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  2122  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52 | <?php  require "koneksimvc2.php";  class m\_programKerja  {  private $nomorProgram;  private $namaProgram;  private $suratKeterangan;  public $hasil = array();  public function setPogramKerja($nomorProgram, $namaProgram, $suratKeterangan)  {  global $mysqli;  $mysqli->query("INSERT INTO proker VALUES('$nomorProgram', '$namaProgram', '$suratKeterangan')");  }  public function updateProgramKerja($old, $nomorProgram, $namaProgram, $suratKeterangan)  {  global $mysqli;  $mysqli->query("UPDATE proker SET  nomorProgram = '$nomorProgram',  namaProgram = '$namaProgram',  suratKeterangan = '$suratKeterangan'  WHERE nomorProgram='$old'");  }  public function getProgramKerja($nomorProgram)  {  global $mysqli;  $rs = $mysqli->query("SELECT \* FROM proker WHERE nomorProgram = '$nomorProgram'");  return $rs->fetch\_assoc();  }  public function delete($nomorProgram)  {  global $mysqli;  $mysqli->query("DELETE FROM proker WHERE nomorProgram = '$nomorProgram'");  }  public function getSemuaPogramKerja()  {  global $mysqli;  $rs = $mysqli->query("SELECT \* FROM proker");  $rows = array();  while($row = $rs->fetch\_assoc()) {  $rows[] = $row;  }  $this->hasil = $rows;  return $this->hasil;  }  } |

|  |  |
| --- | --- |
|  | c\_programKerja.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  2122  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56 | <?php  include\_once("m\_programKerja.php");  class c\_programKerja  {  public $model;  public function \_\_construct()  {  $this->model = new m\_programKerja();  }  public function add()  {  include 'tambah.php';  }  public function post($post)  {  $nomor = $post['nomor'];  $nama = $post['nama'];  $surat = $post['surat'];  $this->model->setPogramKerja($nomor, $nama, $surat);  header('Location: index.php');  }  public function edit($nomor)  {  $data = $this->model->getProgramKerja($nomor);  include('edit.php');  }  public function update($post)  {  $old = $post['nomor\_old'];  $nomor = $post['nomor'];  $nama = $post['nama'];  $surat = $post['surat'];  $this->model->updateProgramKerja($old, $nomor, $nama, $surat);  header('Location: index.php');  }  public function delete($nomor)  {  $this->model->delete($nomor);  header('Location: index.php');  }  public function index()  {  $proker = $this->model->getSemuaPogramKerja();  include 'v\_programKerja.php';  }  } |

|  |  |
| --- | --- |
|  | v\_programKerja.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32 | <html>  <body>  <h2>Daftar Program Kerja BEM</h2>  <table>  <a href="tambah.php?page=add">Tambah Proker</a>  <table border="1" cellpadding="5" cellspacing="0">  <thead>  <tr>  <td>Nomor</td>  <td>Nama Program Kerja</td>  <td>Surat Keterangan</td>  <td>Fungsi</td>  </tr>  <tbody>  <?php  foreach($proker as $p){  ?>  <tr>  <td><?php echo $p['nomorProgram']; ?></td>  <td><?php echo $p['namaProgram']; ?></td>  <td><?php echo $p['suratKeterangan']; ?></td>  <td>  <a href="?page=edit&id=<?php echo $p['nomorProgram']; ?>" class="ui button">Edit</a>  <a href="?page=delete&id=<?php echo $p['nomorProgram']; ?>" class="ui button" style="margin-left: 1%">Delete</a>  </td>  </tr>  <?php };?>  </tbody>  </thead>  </table>  </body>  </html> |

|  |  |
| --- | --- |
|  | index.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  2122  23  24  25  26  27  28  29 | <?php  include\_once("c\_programKerja.php");  $controller = new c\_programKerja();  if (!isset($\_GET['page'])) {  $controller->index();  exit();  } else {  $page = $\_GET['page'];  }  switch ($page) {  case 'add':  $controller->add();  break;  case 'edit':  $controller->edit($\_GET['id']);  break;  case 'post':  $controller->post($\_POST);  break;  case 'update':  $controller->update($\_POST);  break;  case 'delete':  $controller->delete($\_GET['id']);  break;  } |

|  |  |
| --- | --- |
|  | tambah.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20 | <!DOCTYPE html>  <html>  <body>  <form action="index.php?page=post" method="post">  <label>  Nomor proker<br>  </label>  <input type="number" name="nomor" placeholder="Nomor proker"><br>  <label>  Nama proker<br>  </label>  <input type="text" name="nama" placeholder="Nama proker"><br>  <label>  Surat keterangan<br>  </label>  <input type="text" name="surat" placeholder="Surat keterangan"><br>  <input type="submit" name="tambah" value="Tambah">  </form>  </body>  </html> |

|  |  |
| --- | --- |
|  | edit.php |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26 | <!DOCTYPE html>  <html>  <head>  <title></title>  </head>  <h2>Daftar Program Kerja BEM</h2>  <table>  <body>  <form action="index.php?page=update" method="post">  <input type="hidden" name="nomor\_old" value="<?php echo $data['nomorProgram']?>">  <label>  Nomor proker<br>  </label>  <input type="number" name="nomor" value="<?php echo $data['nomorProgram']?>" readonly><br>  <label>  Nama proker<br>  </label>  <input type="text" name="nama" value="<?php echo $data['namaProgram'] ?>"><br>  <label>  Surat Keterangan<br>  </label>  <input type="text" name="surat" value="<?php echo $data['suratKeterangan'] ?>"><br>  <input type="submit" name="edit">  </form>  </body>  </html> |

1. **Penjelasan**

Pada index.php sebagai halaman awal yang pertama kali akan di eksekusi, akan membuat objek controller baru (c\_programKerja), dimana methodnya akan dijalankan sesuai dengan routing yang sudah ditentukan. Jika pada GET page tidak ada, maka akan menampilkan daftar proker. Jika page bernilai maka :

* add : memanggil method add pada controller
* edit : memanggil method edit dengan mempassing $\_GET[’id’] pada controller
* post : memanggil method post dengan mempassing $\_POST pada controller
* update : memanggil method update dengan mempassing $\_POST pada controller
* delete : memanggil method delete dengan mempassing $\_GET[’id’] pada controller

Pada controller sendiri (c\_programKerja) dibuat model( m\_prokerKerja). Dan juga terdapat method sebagai berikut:

* add : menampilkan view tambah.php
* edit : menampilkan view edit.php
* post : memanggil setProgramKerja pada model untuk menambah data proker
* update : memanggil updateProgramKerja pada model untuk mengupdate isi proker
* delete : memanggil delete pada model untuk menghapus data proker

Pada model (m\_programKerja) terdapat method sebagai berikut:

* setProgramKerja : berfungsi untuk menambahkan data proker baru ke database
* getProgramKerja : berfungsi untuk mendapatkan salah satu proker tergantung pada nomorProker yang diberikan
* updateProgramKerja : berfungsi untuk mengubah isi dari proker tergantung pada nomorProgram yang diberikan
* delete : berfungsi untuk menghapus proker dengan nomorProgram yang diberikan
* getSemuaProgramKerja : berfungsi untuk mendapatkan semua proker dari database

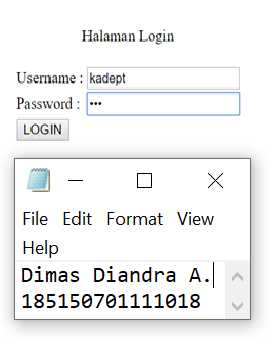
Pada view (v\_programKerja) akan menampilkan semua program kerja yang ada, tambah.php akan menampilkan form untuk menambahkan proker dan edit.php akan menampilkan form untuk mengedit proker.

**LATIHAN 5**

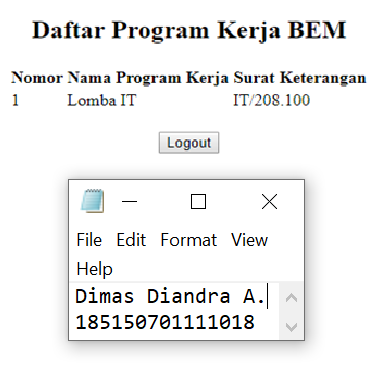
1. **Soal**

Modifikasilah sesuai kebutuhan terlebih dahulu, lengkapi dengan halaman login dan halaman administrasi dan gabungkan Latihan 2 dan Latihan 3 sehingga didapat skenario Menteri hanya dapat melihat daftar program kerja saja, sedangkan Kepala Departemen dapat melakukan proses manajemen program kerja (CRUD). Lakukan modifikasi pada *database* dan struktur class model, controller dan view jika diperlukan untuk mengakomodasi skenario tersebut.

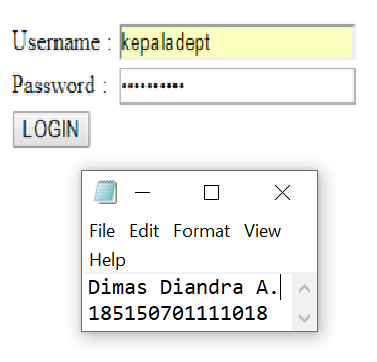
1. **Screenshoot**



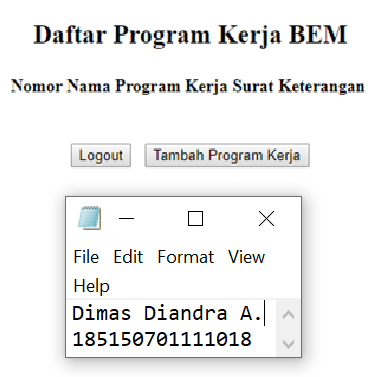
*Melakukan login sebagai menteri*



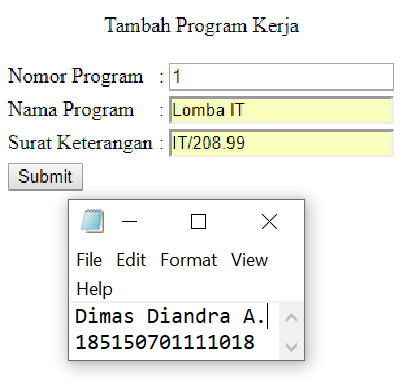
*Hasil apabila melakukan login sebagai menteri*



*Melakukan login sebagai kepala departemen*



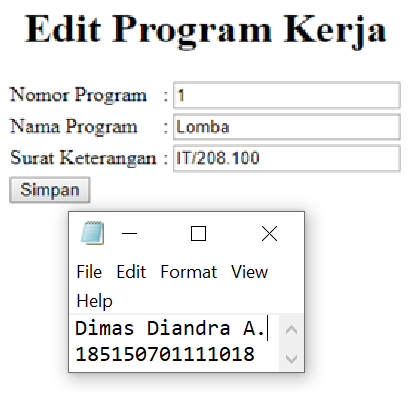
*Menambahkan data*



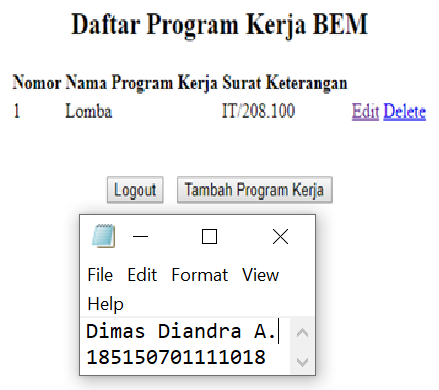
*Mengisi tambah data*



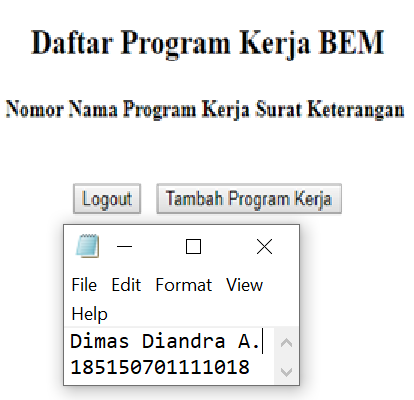
Apabila proses tambah data berhasil



*Melakukan edit*



*Jika edit berhasil*



*Menghapus program kerja*

1. **Syntax**

|  |  |
| --- | --- |
| Koneksimvc2.php | |
| 1  2  3 | <?php  $mysqli = mysqli\_connect('localhost', 'root', 'malang12', 'praktikumpemweb') or die("Fail to connect"); |

|  |  |
| --- | --- |
| login.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | <html>  <head>  </head>  <body>  <center>  <form method='post' action='index.php'>  <table>  <tr>  <p>Halaman Login</p>  </tr>  <tr>  <td>Username :</td>  <td><input type='text' name='id'></td>  </tr>  <tr>  <td>Password : </td>  <td><input type='password' name='pass'></td>  </tr>  <tr>  <td><input type='submit' name='Login' value='LOGIN'></td>  </tr>  </table>  </form>  </body>  </html> |

|  |  |
| --- | --- |
| c\_programKerja.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  6768  69  70  71  72  73  74  75 | <?php  include\_once("m\_programKerja.php");  class c\_programKerja  {  public $model;  public function \_\_construct()  {  $this->model = new m\_programKerja();  }  public function invoke(){  session\_start();  if (isset($\_SESSION['status']) && $\_SESSION['status']=='kadep'){  if(isset($\_GET['edit'])){  $data = $this->model->getPogramKerja($\_GET['nomorProgram']);  include 'edit.php';  if (isset($\_POST['Edit'])){  $add= $this->model->updatePogramKerja($\_GET['nomorProgram'], $\_POST['namaProgram'], $\_POST['suratKeterangan']);  header('Location:index.php');  }  }  else if(isset($\_GET['delete'])){  $del= $this->model->deletePogramKerja($\_GET['nomorProgram']);  header('Location:index.php');  }  else if(isset($\_GET['tambah'])){  include 'tambah.php';  if (isset($\_POST['Add'])){  $add= $this->model->setPogramKerja($\_POST['nomorProgram'], $\_POST['namaProgram'], $\_POST['suratKeterangan']);  header('Location:index.php');  }  }  else if(isset($\_GET['Logout'])){  $logout= $this->model->Logout();  header('Location:index.php');  }  else{  $proker = $this->model->getSemuaPogramKerja();  Include 'prokerCRUD.php';  }  }  else if(isset($\_SESSION['status']) && $\_SESSION['status']=='menteri'){  if(isset($\_GET['Logout'])){  $logout= $this->model->Logout();  header('Location:index.php');  }  else {  $proker = $this->model->getSemuaPogramKerja();  Include 'v\_programKerja.php';  }  }  else{  Include 'login.php';  if (isset($\_POST['Login'])){  $login = $this->model->Login($\_POST['id'], $\_POST['pass']);  header('Location:index.php');  }  }  }  } |

|  |  |
| --- | --- |
| m\_programKerja.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  6768  69  70  71 | <?php  include 'kepalaDepartemen.php';  include 'menteri.php';  class m\_programKerja {  public $hasil = array();  public $user;  public function setPogramKerja($nomorProgram, $namaProgram, $suratKeterangan){  include "koneksimvc2.php";  $rs = $mysqli->query(" INSERT INTO proker VALUES ('$nomorProgram','$namaProgram','$suratKeterangan')");  }  public function getSemuaPogramKerja(){  include "koneksimvc2.php";  $rs = $mysqli->query("SELECT \* FROM proker");  $rows = array();  while($row = $rs->fetch\_assoc()) {  $rows[] = $row;  }  return $rows;  }  public function getPogramKerja($nomorProgram){  include "koneksimvc2.php";  $rs = $mysqli->query("SELECT \* FROM proker WHERE nomorProgram='$nomorProgram'");  return mysqli\_fetch\_array($rs);  }  public function updatePogramKerja($nomorProgram, $namaProgram, $suratKeterangan){  include "koneksimvc2.php";  $rs = $mysqli->query(" UPDATE proker SET nomorProgram= '$nomorProgram', namaProgram = '$namaProgram', suratKeterangan = '$suratKeterangan' WHERE nomorProgram = '$nomorProgram'");  }  public function deletePogramKerja($nomorProgram){  include "koneksimvc2.php";  $rs = $mysqli->query(" DELETE from proker WHERE nomorProgram = '$nomorProgram'");  }  public function Login($id, $pass){  if($id=='kepaladept' && $pass=='kepaladept'){  $\_SESSION['status'] = 'kadep';  $user = new kepalaDepartment();  $user->setData('Cahyo Gusti','185150701111017','2018');  $user->setJabatan();  }  else if($id=='menteri' && $pass=='menteri'){  $\_SESSION['status'] = 'menteri';  $user = new menteri();  $user->setData('Samid Diandra','185150701111018','2018');  $user->setJabatan();  }  }  public function Logout(){  session\_start();  session\_destroy();  }  } |

|  |  |
| --- | --- |
| v\_programKerja.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38 | <html>  <body>  <h2>Daftar Program Kerja BEM</h2>  <table>  <a href="tambah.php?page=add">Tambah Proker</a>  <table border="1" cellpadding="5" cellspacing="0">  <thead>  <tr>  <td>Nomor</td>  <td>Nama Program Kerja</td>  <td>Surat Keterangan</td>  </tr>  <tbody>  <?php  foreach($proker as $p){  ?>  <tr>  <td><?php echo $p['nomorProgram']; ?></td>  <td><?php echo $p['namaProgram']; ?></td>  <td><?php echo $p['suratKeterangan']; ?></td>  <td>  <a href="?page=edit&id=<?php echo $p['nomorProgram']; ?>" class="ui button">Edit</a>  <a href="?page=delete&id=<?php echo $p['nomorProgram']; ?>" class="ui button" style="margin-left: 1%">Delete</a>  </td>  </tr>  <?php };?>  </tbody>  </thead>  </table>  </body>  </html> |

|  |  |
| --- | --- |
| tambah.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | <!DOCTYPE html>  <html>  <body>  <form action="index.php?page=post" method="post">  <label>  Nomor proker<br>  </label>  <input type="number" name="nomor" placeholder="Nomor proker"><br>  <label>  Nama proker<br>  </label>  <input type="text" name="nama" placeholder="Nama proker"><br>  <label>  Surat keterangan<br>  </label>  <input type="text" name="surat" placeholder="Surat keterangan"><br>  <input type="submit" name="tambah" value="Tambah">  </form>  </body>  </html> |

|  |  |
| --- | --- |
| edit.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | <!DOCTYPE html>  <html>  <head>  <title></title>  </head>  <h2>Daftar Program Kerja BEM</h2>  <table>  <body>  <form action="index.php?page=update" method="post">  <input type="hidden" name="nomor\_old" value="<?php echo $data['nomorProgram']?>">  <label>  Nomor proker<br>  </label>  <input type="number" name="nomor" value="<?php echo $data['nomorProgram']?>" readonly><br>  <label>  Nama proker<br>  </label>  <input type="text" name="nama" value="<?php echo $data['namaProgram'] ?>"><br>  <label>  Surat Keterangan<br>  </label>  <input type="text" name="surat" value="<?php echo $data['suratKeterangan'] ?>"><br>  <input type="submit" name="edit">  </form>  </body>  </html> |

|  |  |
| --- | --- |
| pengurusBEM.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 | <?php  class pengurusBEM {  public $nama;  public $nim;  public $angkatan;  private $jabatan;  private $foto;  public function setData($nama,$jabatan,$angkatan) {  $this->nama = $nama;  $this->jabatan = $jabatam;  $this->angkatan = $angkatan;  }  public function getNama() {  echo $this->nama;  }  public function getJabatan() {  echo $this->jabatan;  }  public function getAngkatan() {  echo $this->angkatan;  }  } |

|  |  |
| --- | --- |
| menteri.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 | <?php  include "pengurusBEM.php";  class menteri extends pengurusBEM {  public function \_\_construct($nama){  $this->nama = $nama;  }  function setJabatan($jabatan) { // method  $this->jabatan = $jabatan;  }  public function getJabatan() {  echo "$this->jabatan";  }  } |

|  |  |
| --- | --- |
| kepalaDepartemen.php | |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | <?php  include\_once 'pengurusBEM.php';  class kepalaDepartemen extends pengurusBEM {  public function \_\_construct(){  }  function setJabatan() {// method  $this->jabatan = 'Kepala Departemen';  }  public function getJabatan() {  echo $this->jabatan;  }  } |

|  |  |
| --- | --- |
| index.php | |
| 1  2  3  4  5 | <?php  include\_once("c\_programKerja.php");  $controller = new c\_programKerja();  $controller->invoke(); |

1. **Penjelasan**

Pada program di atas terdapat form login untuk menentukan apakah seseorang yang login tersebut adalah kepala department atau menteri dimana apabila kepala department mampu menambahkan, merubah, dan menghapus data tersebut, namun menteri hanya bisa melihat isi dari table saja.

Pada controller c\_programKerja ditambahkan method checkUser untuk mengecek apakah user bertipe menteri atau kepaladepartemen, jika bertipe menteri, maka fungsi add, edit, post, update, delete akan di blok aksesnya.

Pada view v\_programKerja dimodifikasi untuk tidak menampilkan tombol tambah program kerja, tombol edit dan delete jika user tidak bertipe kepaladepartemen. Dan juga ditambahkan tombol Logout.

Pada model User terdapat method sebagai berikut:

- get: berfungsi untuk mengambil data user dari username dan password yang diberikan, jika ada maka return object User dari data yang didapatkan, jika tidak ada maka return null

- loggedUser: berfungsi untuk mengambil object user yang sedang login dari session