

UJIAN AKHIR SEMESTER (UAS)
PROYEK PERANGKAT LUNAK (PPL)



Disusun oleh :

Nabilla Nur Fadillah

201601053

SI B

Dosen Pengampu :

Purwono, S.Kom

PROGRAM STUDI S1 SISTEM INFORMASI
SEKOLAH TINGGI ILMU KOMPUTER “YOS SUDARSO”
PURWOKERTO

2020

Soal UAS PPL

1. Jelaskan secara detail latar belakang aplikasi yang telah anda bangun, sertakan alur aplikasi dalam bentuk flowchart sederhana.

Jawaban :

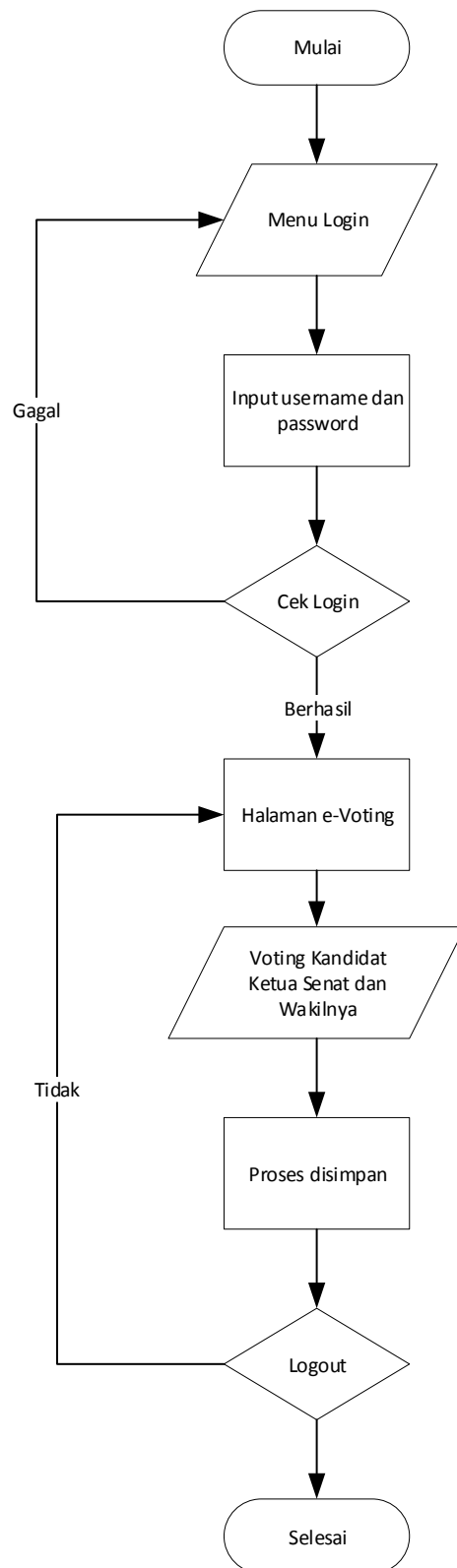
- Latar Belakang

Indonesia merupakan salah satu negara yang menganut demokrasi. Demokrasi itu sendiri ditandai dengan adanya pemilu untuk memilih figur calon seorang pemimpin. Pemilu dilakukan dengan cara memilih suara terbanyak dari suatu kelompok, yang nantinya hasil suara terbanyak tersebut dapat dinyatakan sebagai pemenang dan akan diangkat menjadi seorang pimpinan. Kampus adalah salah satu yang menganut sistem demokrasi. Contohnya dalam pemilihan calon ketua SEMA (Senat Mahasiswa), para mahasiswa akan memilih sepasang calon kandidat yang nantinya akan menjabat sebagai ketua SEMA.

Selama ini pemilu di Indonesia masih menggunakan sistim kertas suara. Para pemilih akan datang ke TPS (Tempat Pemungutan Suara) yang sudah disediakan sebelumnya, kemudian diberikan kertas suara. Pemilih memilih dengan cara mencoblos salah satu atau sepasang calon kandidat dengan alat bantu seperti paku, dan setelah itu memasukkannya dalam kotak suara yang telah disediakan. Menanggapi hal semacam itu dirasa kurangnya efisiensi dalam hal pemakaian kertas suara dan kotak suara dari setiap melakukan kegiatan pemilu. Juga kemungkinan adanya salah satu kertas suara yang terselip dalam melakukan perhitungan suara.

Kampus itu sendiri adalah salah satu lembaga yang juga dalam pemilihan kandidat ketua SEMA menggunakan sistem pemilu. Sistem ini masih tergolong kuno dan primitif bila masih menggunakan media kertas dan kotak suara didalam pengaplikasiannya. padahal kampus merupakan tempat bernaungnya para pelajar yang memiliki intelektual dan kreatifitas yang tinggi. Dengan demikian, melalui sistem e-pemilu ini diharapkan dapat meringankan dan membantu jalannya proses demokrasi khususnya pada pemilihan calon ketua SEMA di dalam kampus. Pemilu yang sebelumnya menggunakan kertas suara yang banyak dan pemakaian kotak suara, tetapi dengan adanya sistem pemilihan ini dapat dilakukan dengan cara komputerisasi tanpa menggunakan kertas suara (papperless). Serta dalam perhitungannyapun akan lebih akurat.

- Flowchart



2. Jelaskan secara detail teknologi apa saja yang digunakan dan alasan kenapa anda menggunakan itu.

Jawaban :

Teknologi yang digunakan adalah :

- a. Bootstrap** merupakan salah satu Front-End framework paling populer saat ini. Bootstrap telah mengalami beberapa perubahan, dan sekarang menjadi salah satu framework yang paling stabil dan responsif. Bahkan dengan pengetahuan dasar tentang HTML dan sedikit CSS, developer pemula dapat dengan mudah memulai dengan Bootstrap. Berikut ini 5 alasan mengapa menggunakan Bootstrap:

1. Sistem grid yang powerful

Bootstrap memiliki fitur responsif terbaik dan mudah digunakan. Ini sangat membantu dalam desain situs web dari perangkat seluler di berbagai ukuran. Bootstrap membagi layar menjadi 12 kolom, sehingga kamu dapat memutuskan berapa banyak kolom yang diambil untuk setiap elemen desain situs web.

2. Proses Pembuatan web yang cepat

Hanya perlu menggunakan beberapa HTML untuk memasukkannya ke dalam script template, tanpa perlu menghabiskan banyak waktu menulis CSS dan JavaScript yang kompleks. Hebatnya, semua komponen ini juga responsif.

3. Kompatibilitas Browser

Bootstrap kompatibel dengan versi terbaru dari semua browser.

4. Open Source

Bootstrap adalah open source. Lisensi Bootstrap memberikan kebebasan untuk sepenuhnya mengubah dan bereksperimen dengannya.

- b. CodeIgniter** adalah sebuah Application Development Framework yaitu sebuah toolkit yang sangat cocok bagi programmer pemula yang akan membangun sebuah website menggunakan PHP. Tujuan utamanya adalah memungkinkan untuk mengembangkan proyek-proyek jauh lebih cepat daripada jika menulis kode dari awal. CodeIgniter adalah sebuah Framework yang menurut saya berisi kumpulan fungsi yang lengkap sehingga relatif mampu membuat lebih fokus pada proyek dengan meminimalkan jumlah kode yang dibutuhkan untuk tugas tertentu. Alasan menggunakan codeigniter :

1. Performa sangat cepat.

Codeigniter sangat cepat bahkan mungkin bisa dibilang merupakan framework yang paling cepat dibanding framework yang lain.

2. Konfigurasi yang sangat minim.

CodeIgniter menggunakan setting yang standard, hanya perlu merubah sedikit saja file pada folder config.

3. Dukungan komunitas.

CI relatif punya dukungan komunitas pengguna yang kuat. Sehingga kita dapat lebih mudah berinteraksi dengan mereka yang lebih tahu.

4. Dokumentasi yang lengkap.

Setiap paket instalasi CodeIgniter sudah disertai user guide yang sangat bagus dan lengkap untuk dijadikan permulaan, bahasanya pun mudah dipahami.

5. Menggunakan pattern MVC.

Dengan menggunakan MVC (Model-View-Controller), struktur code menjadi lebih terstruktur dan memiliki standart yang jelas.

- c. **MySQL** adalah sistem manajemen basis data open source yang cepat, handal dan stabil. Berikut adalah alasan mengapa menggunakan MySQL :

1. Keamanan Data

MySQL adalah sistem manajemen basis data yang paling aman dan andal yang digunakan dalam aplikasi web populer seperti WordPress, Facebook dan Twitter.

2. Performa Yang Tinggi

MySQL dirancang untuk memenuhi aplikasi yang paling memerlukan kecepatan optimal.

3. Fleksibilitas Dari Open Source

MySQL adalah database open source sehingga Anda tidak perlu membayar satu rupiah pun untuk menggunakannya.

4. Didukung Oleh Banyak Sistem Operasi Dan Bahasa Pemrograman

MySQL bisa digunakan di banyak sistem operasi seperti Microsoft Windows. Serta didukung oleh banyak bahasa pemrograman seperti PHP.

5. Dukungan Dari Banyak Tool

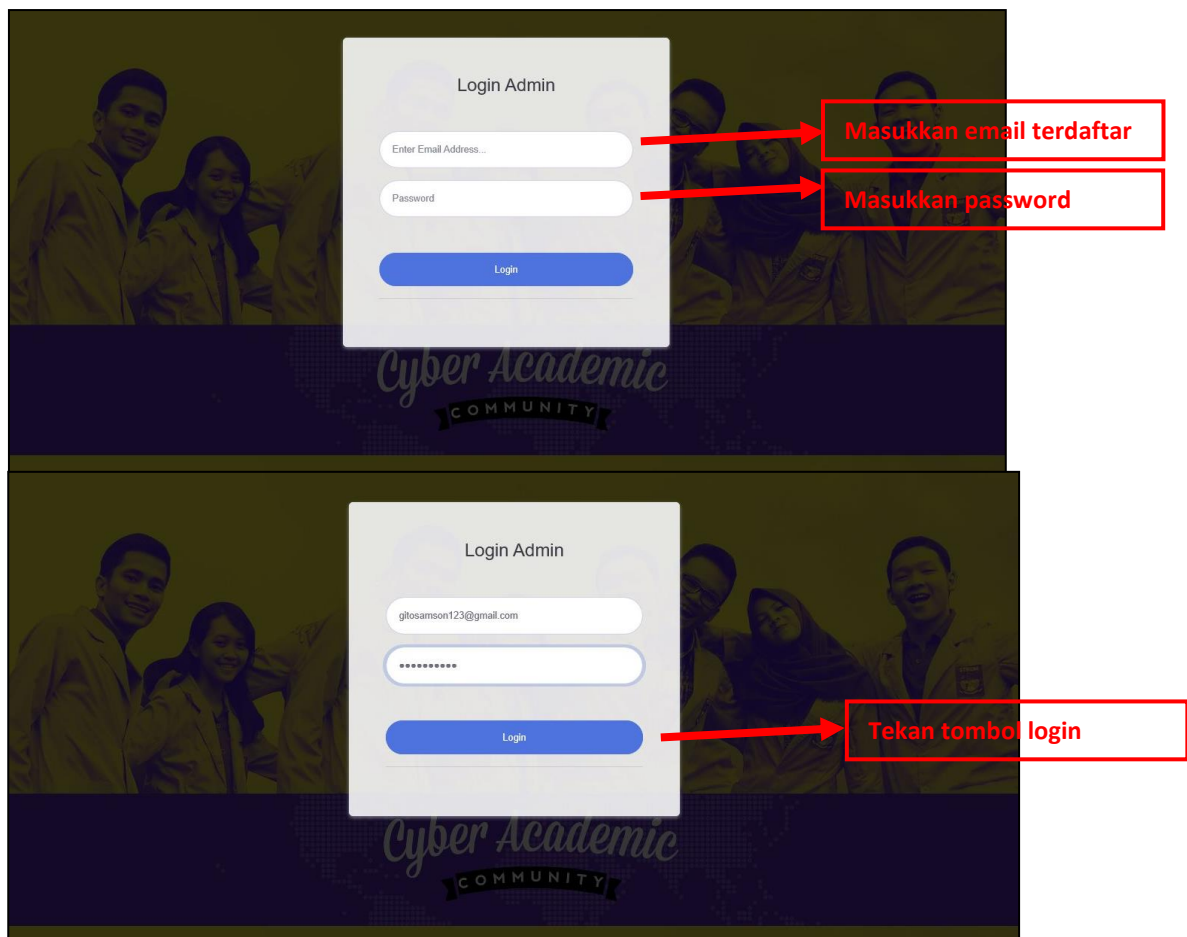
MySQL juga bisa dengan mudah dikelola dengan menggunakan alat visual seperti phpMyAdmin, SQLYog, Navicat, dll. Dan tentunya dengan berbagai tool ini akan sangat membantu para pemula.

d. Javascript digunakan untuk menyempurnakan tampilan dan sistem halaman web. Alasan menggunakan javascript karena mudah untuk mulai belajar karena sumber-sumber belajar JavaScript juga sudah tersedia cukup banyak dan lengkap.

3. Pada aplikasi tentunya membutuhkan alur Login. Silahkan berikan penjelasan alur logika login aplikasi anda, sertakan source code php yang memperlihatkan keterkaitan antara frontend, database dan backend pada tahap login.

Jawaban :

Tampilan halaman login pada website saya adalah seperti dibawah ini :



Alur login :

Pada halaman login diatas, user diminta untuk memasukkan email yang sudah terdaftar (untuk superadmin dan admin), NIDN (untuk pimpinan), NIDN/NIK (untuk dosen/karyawan), dan NIM (untuk mahasiswa). Kemudian user diminta untuk memasukkan password, lalu tekan tombol Login. Kemudian apa yang telah diinputkan oleh user tadi akan dicek oleh Controller. Berikut adalah source code controller :

Kode program :

Controller : Auth.php (fungsi index())

```
11     public function index()
12     {
13         if ($this->session->userdata('email')) {
14             redirect('user');
15         }
16
17         $this->form_validation->set_rules('email', 'Email', 'trim|required|valid_email');
18         $this->form_validation->set_rules('password', 'Password', 'trim|required');
19         if ($this->form_validation->run() == false) {
20             $data['title'] = 'Login Page';
21             $this->load->view('templates/auth_header', $data);
22             $this->load->view('auth/login');
23             $this->load->view('templates/auth_footer');
24         } else {
25             // validasinya sukses
26             $this->_login();
27         }
28     }
29 }
```

Keterangan program :

Kode program diatas adalah fungsi bernama index() yang terdapat pada Controller Auth.php untuk me-load view halaman login.

Baris 13-15 adalah kondisi ketika seorang user sudah login kemudian dia mengakses halaman home tanpa melakukan logout, maka ketika dia ingin masuk ke halaman user lagi, dia tidak perlu login lagi karena controller telah mengambil sesi ketika user melakukan login menggunakan email (untuk superadmin dan admin).

Baris 16-17 untuk memvalidasi apakah yang diisikan itu sesuai dengan rules(aturan) yang diinginkan. Contohnya trim berfungsi agar inputan tidak boleh mengandung spasi, required berfungsi untuk form input harus diisi, valid_email berfungsi untuk mengecek apakah isian pada form input berupa format email yang benar atau tidak.

Pada baris 18-27 terdapat kondisi apabila yang diinputkan tidak sesuai dengan validasi yang diinginkan maka akan me-load halaman yang sama yaitu halaman login beserta dengan form errornya, namun apabila sesuai dengan validasinya maka akan masuk ke method _login().

Berikut adalah function _login() :

Controller Auth.php (fungsi _login())

Ketika user sudah menginputkan sesuai dengan rules yang ada pada fungsi index() diatas, maka selanjutnya akan masuk ke fungsi _login().

```
219     private function _login()
220     {
221         $email = $this->input->post('email');
222         $password = $this->input->post('password');
223
224         $user = $this->db->get_where('user', ['email' => $email])->row_array();
225
226         $id_role = $user['role_id'];
227         $role = $this->db->get_where('user_role', ['id' => $id_role])->row_array();
228
229         // jika usernya ada
230         if ($user) {
231             //jika usernya aktif
232             if ($user['is_active'] == 1) {
233                 //cek password
234                 if (password_verify($password, $user['password'])) {
235                     $data = [
236                         'email' => $user['email'],
237                         'role_id' => $user['role_id'],
238                         'id' => $role['id']
239                     ];
240                     $this->session->set_userdata($data);
241                     if ($user['role_id'] == 1) {
242
243                         redirect('admin');
244                     } else if ($user['role_id'] == 2) {
245                         redirect('admin');
246                     } else {
247
248                         redirect('user');
249                     }
250                 } else {
251                     $this->session->set_flashdata('message', '<div class="alert
252                     alert-danger" role="alert"> Wrong password!</div>');
253                     redirect('auth');
254                 }
255             } else {
256                 $this->session->set_flashdata('message', '<div class="alert
257                 alert-danger" role="alert"> This email has not been activated!</div>');
258                 redirect('auth');
259             }
260         } else {
261             $this->session->set_flashdata('message', '<div class="alert
262             alert-danger" role="alert">Email is not registered!</div>');
263             redirect('auth');
264         }
265     }
```

Keterangan program :

Baris 221 dan 222 merupakan kode untuk mengambil value hasil post dari user yang berupa email dan password.

Baris 224 digunakan untuk mengambil data tabel user dari database yang berisi email sesuai dengan yang diinputkan.

Baris 226 digunakan untuk mengambil role_id dari tabel user berdasarkan email dari baris 224 diatas.

Baris 227 digunakan untuk mengambil id dari tabel user_role berdasarkan id_role dari baris 226 diatas.

Baris 230 untuk mengecek apakah user yang diinputkan (email) ada atau tidak di dalam database

Baris 232 untuk mengecek apakah user tersebut aktif atau tidak, jika aktif ditandai dengan 1, jika tidak aktif 0.

Baris 234 untuk mengecek password, jika benar maka akan menyimpan data session yang ditujukan pada baris 235-240.

Baris 235-239 digunakan untuk membungkus data email, role_id dan id ke dalam sebuah array.

Baris 240 digunakan untuk membuat session dari data yang telah dibungkus dalam array seperti yang dijelaskan pada baris 235-239 diatas.

Baris 241-249 untuk mengecek role, jika role id = 1 atau 2 maka akan diarahkan ke halaman admin, selain role id 1 dan 2 akan diarahkan ke dalam halaman user.

Baris 250-253 untuk mengecek jika password salah akan me-reload halaman login yang disertai dengan pemberitahuan password salah.

Baris 255-259 untuk mengecek jika email tidak aktif akan me-reload halaman login yang disertai dengan pemberitahuan email belum aktif.

Baris 260-264 untuk mengecek jika email belum terdaftar akan me-reload halaman login yang disertai dengan pemberitahuan email belum terdaftar

Source code dibawah ini adalah source dari front end view login.php yang telah di-load oleh controller.

View : login.php

```
1 <div class="container">
2   <!-- Outer Row -->
3   <div class="row justify-content-center">
4     <div class="col-lg-5">
5       <div class="card my-5">
6         <div class="card-body p-0">
7           <!-- Nested Row within Card Body -->
8           <div class="row">
9             <div class="col-lg">
10              <div class="p-5">
11                <div class="text-center">
12                  <h1 class="h4 text-gray-900 mb-5">Login Admin</h1>
13                </div>
14                <?=$this->session->flashdata('message'); ?>
15                <form class="user" method="post" action="{?}= base_url('auth'); ?>">
16                  <div class="form-group">
17                    <input type="text" class="form-control form-control-user" id="email" name="email" placeholder="Enter Email Address..." value="{?}= set_value('email'); ?>"
18                      <?=$form_error('email', ' <small class="text-danger pl-3">', ' </small>'); ?>
19                  </div>
20                  <div class="form-group mb-5">
21                    <input type="password" class="form-control form-control-user" id="password" name="password" placeholder="Password"
22                      <?=$form_error('password', ' <small class="text-danger pl-3">', ' </small>'); ?>
23                  </div>
24                  <button type="submit" class="btn btn-primary btn-user btn-block">
25                    Login
26                  </button>
27                </form>
28              </div>
29            </div>
30          </div>
31        </div>
32      </div>
33    </div>
34  </div>
35 </div>
36 </div>
37 </div>
```

View diatas adalah view untuk menampilkan halaman login.

4. Pada aplikasi tentunya memiliki alur Simpan Data (pilih salah satu). Silahkan berikan penjelasan alur logika simpan data aplikasi anda, sertakan source code php yang memperlihatkan keterkaitan antara frontend, database dan backend pada tahap simpan data.

Jawaban :

Simpan Data yang akan dijelaskan yaitu mengenai **simpan data dosen/karyawan**. Cara untuk menambah dosen yaitu seperti dibawah ini :

Dosen			
Tambah Dosen			
Tampilkan 10 Data			
No	NIK	Nama	
1	2005004	Oscar Adi	
2	2012132	F. Aribroto, M.Pd	

Untuk menambah dosen/karyawan baru maka tekan Tambah Dosen.

Tambah Dosen

NIK

Masukkan NIK

Nama Lengkap

Masukkan Nama Lengkap

Password

Masukkan Password

Repeat Password

Ulangi Password

Jenis Kelamin

☒ Laki-laki ☐ Perempuan

Mengajar

Mengajar

Email

Masukkan Email Valid


Handphone

Masukkan Nomor HP

Aktif

☒ Ya ☐ Tidak

Foto



Pilih foto

Browse

Tambah

Kembali

Alur simpan data :

Kemudian akan muncul halaman seperti gambar diatas, isikan nomor induk kependudukan (nik), nama lengkap, password, ulangi password, jenis kelamin, mata kuliah yang diampu, email, nomor handphone, status aktif, dan foto profil (jika foto tidak diisi maka akan otomatis menggunakan gambar default), kemudian tekan Tambah. Kemudian apa yang telah diinputkan oleh tadi akan dicek oleh Controller. Berikut adalah source code controller :

Kode program :

Controller : Operation.php (fungsi tambahdosen())

```
214 public function tambahdosen()
215 {
216     $data['title'] = 'Dosen';
217     $data['user'] = $this->db->get_where('user', ['email' =>
218     $this->session->userdata('email')])>->row_array();
219     $this->form_validation->set_rules('nik', 'NIK', 'required|trim|is_unique[dosen.nik]', [
220     'is_unique' => 'This nik has already registered!');
221     ];
222     $this->form_validation->set_rules('namalengkap', 'Nama Lengkap', 'required|trim');
223     $this->form_validation->set_rules('password', 'Password', 'required|trim|min_length[3]|matches[password2]', [
224     'matches' => 'password dont match!',
225     'min_length' => 'password too short!');
226     ];
227     $this->form_validation->set_rules('password2', 'Password', 'required|trim|matches[password]');
228     $this->form_validation->set_rules('mengajar', 'Mengajar', 'required|trim');
229     $this->form_validation->set_rules('email', 'Email', 'required|trim|valid_email|is_unique[dosen.email]', [
230     'is_unique' => 'This email has already registered!');
231     ];
232     $this->form_validation->set_rules('hp', 'Hp', 'required|trim');
233     $data['namarole'] = $this->db->get_where('user_role', ['id' =>
234     $this->session->userdata('id')])>->row_array();
235     if ($this->form_validation->run() == false) {
236         $this->load->view('templates/header', $data);
237         $this->load->view('templates/sidebar', $data);
238         $this->load->view('templates/topbar', $data);
239         $this->load->view('operation/dosen/tambahdosen', $data);
240         $this->load->view('templates/footer');
241     } else {
242         $this->Dosen_model->tambahDataDosen();
243         $this->session->set_flashdata('message', 'Ditambahkan!');
244         redirect('operation/dosen');
245     }
}
```

Keterangan program:

Kode program diatas adalah fungsi bernama tambahdosen() yang terdapat pada Controller Operation.php untuk me-load view halaman tambah dosen.

Baris 216 digunakan untuk memberikan sebuah judul pada halaman tambahdosen.php.

Baris 217-218 digunakan untuk mengambil data user berdasarkan session email.

Baris 219-221 digunakan untuk penanganan validasi terhadap form NIK yang diinputkan, apabila yang diinputkan pada form NIK tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (harus diisi), trim (tidak boleh ada spasi), is_unique (tidak boleh sama dengan NIK lain yang pernah diinput).

Baris 222 digunakan untuk penanganan validasi terhadap form nama lengkap yang diinputkan, apabila yang diinputkan pada form nama lengkap tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (harus diisi) dan trim (tidak boleh ada spasi dibelakang).

Baris 223-226 digunakan untuk penanganan validasi terhadap form password yang diinputkan, apabila yang diinputkan pada form password tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (harus diisi), trim (tidak boleh ada spasi), min_length[3] (minimal password berisi 3 karakter), mathes[password2] (password harus sama dengan password2).

Baris 227 digunakan untuk penanganan validasi terhadap form password 2 yang diinputkan, apabila yang diinputkan pada form password tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (tidak boleh kosong), trim (tidak boleh ada spasi), dan mathes[password] (password 2 harus sama dengan password).

Baris 228 digunakan untuk penanganan validasi terhadap form mengajar yang diinputkan, apabila yang diinputkan pada form mengajar tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (tidak boleh kosong) dan trim (tidak boleh ada spasi).

Baris 229-231 digunakan untuk penanganan validasi terhadap form email yang diinputkan, apabila yang diinputkan pada form password tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (tidak boleh kosong), trim (tidak boleh ada spasi), valid_email (email yang diinputkan harus sesuai dengan struktur penulisan email yang valid), dan is_unique (tidak boleh sama dengan email lain yang pernah diinput).

Baris 232 digunakan untuk penanganan validasi terhadap form hp yang diinputkan, apabila yang diinputkan pada form hp tidak sesuai rules yang ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (harus diisi) dan trim (tidak boleh ada spasi)

Baris 233-234 digunakan untuk mengambil nama_role pada data user_role berdasarkan session id.

Baris 235-240 digunakan apabila validasi form yang sudah dijelaskan diatas tidak sesuai dengan rules yang telah ditentukan maka akan meload view tambah dosen beserta pesan error pada form yang tidak sesuai rules.

Baris 241-245 digunakan apabila yang diinputkan benar maka akan masuk ke dalam Dosen_model pada fungsi tambahDataDosen() untuk penanganan lebih lanjut, kemudian akan memunculkan pesan data telah berhasil ditambahkan di halaman dosen.

Model : Dosen_model.php (fungsi upload() dan fungsi tambahDataDosen())

```
15 public function upload()
16 {
17     $upload_image = $_FILES['imagedosen']['name'];
18     if ($upload_image) {
19         $config['allowed_types'] = 'gif|jpg|png';
20         $config['max_size'] = '2048';
21         $config['upload_path'] = './assets/img/profile/dosen';
22         $this->load->library('upload', $config);
23         if ($this->upload->do_upload('imagedosen')) {
24             return $this->upload->data('file_name');
25         }
26     }
27     return "default.jpg";
28 }
```

```
33 public function tambahDataDosen()
34 {
35     $nik = $this->input->post('nik', true);
36     $data = [
37         'nik' => $nik,
38         'name' => $this->input->post('namalengkap', true),
39         'password' => password_hash($this->input->post('password'), PASSWORD_DEFAULT),
40         'jk' => $this->input->post('jk', true),
41         'mengajar' => $this->input->post('mengajar', true),
42         'email' => $this->input->post('email', true),
43         'hp' => $this->input->post('hp', true),
44         'is_active' => $this->input->post('aktif', true),
45         'image' => $this->upload(),
46         'role_id' => 4
47     ];
48     $this->db->insert('dosen', $data);
49 }
```

Keterangan program :

Pada Dosen_model.php, method upload(), digunakan untuk mengupload gambar beserta ketentuannya (akan dijelaskan lebih lanjut pada nomor selanjutnya yang khusus membahas fungsi upload()).

Pada method tambahdataDosen() baris 35-47 digunakan untuk membungkus data dosen (dibungkus oleh variabel \$data) yang telah diinputkan melalui form dan telah dicek oleh controller.

Baris 48 digunakan untuk meng-insert data yang telah dibungkus tadi ke dalam tabel dosen di database.

Source code dibawah ini adalah source dari front end tambahdosen.php yang telah di-load oleh controller.

View : tambahdosen.php

```
<!-- Begin Page Content -->
<div class="container-fluid">
    <!-- Page Heading -->
    <h1 class="h3 mb-4 text-gray-800">Tambah Dosen</h1>
    <div class="row">
        <div class="col-lg-8">
            <? = form_open_multipart('operation/tambahdosen'); ?>
            <div class="form-group row">
```

```

        <label for="nik" class="col-sm-3 col-form-label">NIK</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="nik" name="nik" placeholder="Masukkan NIK" value="<?= set_value('
nik'); ?>">
            <?= form_error('nik', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="namalengkap" class="col-sm-3 col-form-
label">Nama Lengkap</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="namalengkap" name="namalengkap" placeholder="Masukkan Nama Lengka
p" value="<?= set_value('namalengkap'); ?>">
            <?= form_error('namalengkap', ' <small class="text-
danger pl-3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="password" class="col-sm-3 col-form-
label">Password</label>
        <div class="col-sm-9">
            <input type="password" class="form-
control" id="password" name="password" placeholder="Masukkan Password">
            <?= form_error('password', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="password2" class="col-sm-3 col-form-
label">Repeat Password</label>
        <div class="col-sm-9">
            <input type="password" class="form-
control" id="password2" name="password2" placeholder="Ulangi Password">
            <?= form_error('password2', ' <small class="text-
danger pl-3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="jk" class="col-sm-3 col-form-
label">Jenis Kelamin</label>
        <div class="form-check form-check-inline pl-3">
            <input type="radio" name="jk" value="L" <?php echo set_ra
dio('jk', 'L', TRUE); ?> />
            <label class="form-check-label" for="jk">Laki-laki</label>
        </div>
    </div>

```



```

        <div class="form-check form-check-inline">
            <input type="radio" name="jk" value="P" <?php echo set_ra
dio('jk', 'P'); ?> />
            <label class="form-check-label" for="jk">Perempuan</label>
        </div>
    </div>
    <div class="form-group row">
        <label for="mengajar" class="col-sm-3 col-form-
label">Mengajar</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="mengajar" name="mengajar" placeholder="Mengajar" value="<?= set_v
alue('mengajar'); ?>">
            <?= form_error('mengajar', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="email" class="col-sm-3 col-form-
label">Email</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="email" name="email" placeholder="Masukkan Email Valid" value="<?=
set_value('email'); ?>">
            <?= form_error('email', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="hp" class="col-sm-3 col-form-
label">Handphone</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="hp" name="hp" placeholder="Masukkan Nomor HP" value="<?= set_valu
e('hp'); ?>">
            <?= form_error('hp', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label class="col-sm-3 col-form-label">Aktif</label>
        <div class="form-check form-check-inline pl-3">
            <input type="radio" name="aktif" value="1" <?php echo set
_radio('aktif', '1', TRUE); ?> />
            <label class="form-check-label">Ya</label>
        </div>
    <div class="form-check form-check-inline">

```

```

        <input type="radio" name="aktif" value="0" <?php echo set
_radio('aktif', '0'); ?> />
        <label class="form-check-label">Tidak</label>
    </div>
</div>
<div class="form-group row">
    <div class="col-sm-3">Foto</div>
    <div class="col-sm-9">
        <div class="row">
            <div class="col-sm-3">
                
            </div>
            <div class="col-sm-9">
                <div class="custom-file">
                    <input type="file" class="custom-file-
input" id="imagedosen" name="imagedosen">
                    <label class="custom-file-
label" for="image">Pilih foto</label>
                </div>
            </div>
        </div>
    </div>
</div>
<div class="form-group row justify-content-end">
    <div class="col-sm-9">
        <button type="submit" class="btn btn-
primary">Tambah</button>

        <a href="<?= base_url('operation/dosen'); ?>" class="btn b
tn-danger">Kembali</a>
    </div>
</div>
</form>
</div>
</div>
<!-- /.container-fluid -->
</div>
<!-- End of Main Content -->

```

View diatas adalah view untuk menampilkan halaman form tambah dosen.

5. Pada aplikasi tentunya memiliki alur Tampil Data (pilih salah satu). Silahkan berikan penjelasan alur logika tampil data aplikasi anda, sertakan source code php yang memperlihatkan keterkaitan antara frontend, database dan backend pada tahap tampil data.

Jawaban :

Tampil Data yang akan dijelaskan yaitu mengenai **tampil data dosen/karyawan**. Tampilan halaman **tampil data dosen/karyawan** pada website saya adalah seperti dibawah ini :

Dosen									
<div>Tambah Dosen</div>									
Tampilkan 10		Data		Pencarian: <input type="text"/>					
No	NIK	Nama	JK	Mengajar	Email	HP	Aktif	Opsi	
1	2005004	Oscar Adi	L	pemrogramman web	oscaradi@yahooo.com	089888777666	1	opsi	
2	2012132	F. Aribroto, M.Pd	L	-	aribroto@gmail.com	08xxxxxxxxxx	0	opsi	
3	201912001	Drs. Titus S., M.Pd.	L	Bahasa Indonesia	titus@stikomys.ac.id	08xxxxxxxxxx	1	opsi	
4	201912002	Nyoto Soeharmoko, S.Kom	L	Algoritma dan Pemrograman	moko@stikomys.ac.id	08xxxxxxxxxx	1	opsi	

Alur program :

Dosen/karyawan adalah sebuah menu yang menampilkan nomor induk kependudukan (nik), nama, jenis kelamin, mengajar, email, nomor handphone dan status aktif. Berikut adalah source code controller untuk menampilkan data dosen/karyawan dari database:

Kode program :

Controller : Operation.php (fungsi dosen())

```
200 public function dosen()
201 {
202     $data['title'] = 'Dosen';
203     $data['user'] = $this->db->get_where('user', ['email' =>
204     $this->session->userdata('email')])->row_array();
205     $data['dosen'] = $this->Dosen_model->getAllDosen();
206     $data['namarole'] = $this->db->get_where('user_role', ['id' =>
207     $this->session->userdata('id')])->row_array();
208     $this->load->view('templates/header', $data);
209     $this->load->view('templates/sidebar', $data);
210     $this->load->view('templates/topbar', $data);
211     $this->load->view('operation/dosen/dosen', $data);
212     $this->load->view('templates/footer');
213 }
```

Keterangan program :

Kode program diatas adalah fungsi bernama dosen() yang terdapat pada Controller Operation.php untuk me-load view halaman dosen.

```
<!-- Begin Page Content -->
<div class="container-fluid">
    <!-- Page Heading -->
    <h1 class="h3 mb-4 text-gray-800"><?= $title; ?></h1>
    <div class="row">
        <div class="col-lg">
            <?php if (validation_errors()) : ?>
                <div class="alert alert-danger" role="alert">
                    <?= validation_errors(); ?>
                </div>
            <?php endif; ?>
            <?php if ($this->session->flashdata('message')) : ?>
                <div class="flashdataadosen" data-flashdatadosen="<?= $this->session->flashdata('message'); ?>"></div>
            <?php endif; ?>
            <a href="<?= base_url('operation/tambahdosen'); ?>" class="btn btn-success mb-3">Tambah Dosen</a>
            <div class="table-responsive">
                <table class="table table-bordered table-hover" id="dataTable">
                    <thead class="thead-dark">
                        <tr>
                            <th scope="col">No</th>
                            <th scope="col">NIK</th>
                        </tr>
                    </thead>
                    <tbody>
                        <tr>
                            <td>1</td>
                            <td>1234567890</td>
                        </tr>
                        <tr>
                            <td>2</td>
                            <td>9876543210</td>
                        </tr>
                        <tr>
                            <td>3</td>
                            <td>0987654321</td>
                        </tr>
                        <tr>
                            <td>4</td>
                            <td>1098765432</td>
                        </tr>
                        <tr>
                            <td>5</td>
                            <td>2109876543</td>
                        </tr>
                        <tr>
                            <td>6</td>
                            <td>3210987654</td>
                        </tr>
                        <tr>
                            <td>7</td>
                            <td>4321098765</td>
                        </tr>
                        <tr>
                            <td>8</td>
                            <td>5432109876</td>
                        </tr>
                        <tr>
                            <td>9</td>
                            <td>6543210987</td>
                        </tr>
                        <tr>
                            <td>10</td>
                            <td>7654321098</td>
                        </tr>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>
```

```

        <th scope="col">Nama</th>
        <th scope="col">JK</th>
        <th scope="col">Mengajar</th>
        <th scope="col">Email</th>
        <th scope="col">HP</th>
        <th scope="col">Aktif</th>
        <th scope="col">Ops</th>
    </tr>
</thead>
<tbody>
    <?php $i = 1; ?>
    <?php foreach ($dosen as $dsn) : ?>
        <tr>
            <th scope="row"><? $i ?></th>
            <td><? $dsn['nik']; ?></td>
            <td><? $dsn['name']; ?></td>
            <td><? $dsn['jk']; ?></td>
            <td><? $dsn['mengajar']; ?></td>
            <td><? $dsn['email']; ?></td>
            <td><? $dsn['hp']; ?></td>
            <td><? $dsn['is_active']; ?></td>
            <td>
                <div class="dropdown ">
                    <button class="btn btn-primary btn-
sm dropdown-toggle" type="button" id="dropdownMenuButton" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
                        opsi
                    </button>
                    <div class="dropdown-menu animated--
fade-in" aria-labelledby="dropdownMenuButton">
                        <a class=" btn btn-success btn-
sm" href="<? $base_url() ?>operation/detaildosen/<? $dsn['nik']; ?>">detail</
a>
                        <a class="btn btn-warning btn-
sm" href="<? $base_url() ?>operation/editdosen/<? $dsn['nik']; ?>" class="bad
ge badge-warning">edit</a>
                        <a class=" btn btn-danger btn-
sm tombol-
hapusdosen" href="<? $base_url() ?>operation/hapusdosen/<? $dsn['nik']; ?>">h
apus</a>
                    </div>
                </div>
            </td>
        </tr>
    <?php $i++; ?>
    <?php endforeach; ?>
</tbody>
</table>

```

```

        </div>
    </div>
</div>
<!-- /.container-fluid -->
</div>
<!-- End of Main Content -->
<!-- Modal -->
<!-- Hapus Modal-->
<div class="modal fade" id="hapusModal" tabindex="-1" role="dialog" aria-
labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-
title" id="exampleModalLabel">Hapus Dosen</h5>
                <button class="close" type="button" data-dismiss="modal" aria-
label="Close">
                    <span aria-hidden="true">x</span>
                </button>
            </div>
            <div class="modal-
body">Apakah Anda yakin ingin menghapus dosen ini?</div>
            <div class="modal-footer">
                <button class="btn btn-danger" type="button" data-
dismiss="modal">Hapus</button>
                <a class="btn btn-
primary" href="<?= base_url('operation/dosen'); ?>">Batal</a>
            </div>
        </div>
    </div>
</div>

```

View diatas adalah view untuk menampilkan halaman tabel dosen/karyawan.

6. Pada aplikasi tentunya memiliki alur Ubah Data. Silahkan berikan penjelasan alur logika ubah data aplikasi anda, sertakan source code php yang memperlihatkan keterkaitan antara frontend, database dan backend pada tahap ubah data (pilih salah satu).

Jawaban :

Ubah Data yang akan dijelaskan yaitu mengenai **ubah data dosen/karyawan**. Tampilan halaman **ubah data dosen/karyawan** pada website saya adalah seperti dibawah ini :

oscaradi@yahoo.com	089888777666	1	opsi ▾
aribroto@gmail.com	08xxxxxxxxxx	detail	edit

Untuk mengedit dosen/karyawan maka pilih dosen/karyawan yang akan diedit lalu tekan tombol opsi kemudian pilih edit.

Form Edit Dosen

NIK

2005004

Nama Lengkap

Oscar Adi

Password

Repeat Password

Jenis Kelamin

☒ Laki-laki ☐ Perempuan

Mengajar

pemrogramman web

Email

oscaradi@yahoo.com


Handphone

089888777666

Aktif

☒ Ya ☐ Tidak

Foto



Ganti Foto

Pilih foto

Browse

Perbarui

Reset

Kembali

Alur ubah data :

Ketik apa yang ingin diubah (nik tidak bisa ubah) lalu tekan tombol Perbarui. . Kemudian apa yang telah diubah tadi akan dicek oleh Controller. Berikut adalah source code controller :

Kode program :

Controller : Operation.php

```
266 public function editdosen($nik)
267 {
268     $data['title'] = 'Dosen';
269     $data['user'] = $this->db->get_where('user', ['email' =>
270     $this->session->userdata('email')])>row_array();
271     $data['nama_role'] = $this->db->get_where('user_role', ['id' =>
272     $this->session->userdata('id')])>row_array();
273     $data['dosen'] = $this->Dosen_model->getDosenById($nik);
274     $dosen = $this->Dosen_model->getDosenById($nik);
275     $this->form_validation->set_rules('nama_lengkap', 'Nama Lengkap', 'required|trim');
276     $this->form_validation->set_rules('passwordddosen1', 'Password', 'trim|min_length[3]|matches[passwordddosen2]', [
277     'matches' => 'password dont match!',
278     'min_length' => 'password too short!');
279     ];
280     $this->form_validation->set_rules('passwordddosen2', 'Password', 'trim|matches[passwordddosen1]');
281     $this->form_validation->set_rules('mengajar', 'Mengajar', 'required');
282     $this->form_validation->set_rules('email', 'Email', 'required|trim|valid_email');
283     $this->form_validation->set_rules('hp', 'Hp', 'required');
284     if ($this->form_validation->run() == false) {
285         $this->load->view('templates/header', $data);
286         $this->load->view('templates/sidebar', $data);
287         $this->load->view('templates/topbar', $data);
288         $this->load->view('operation/dosen/editdosen', $data);
289         $this->load->view('templates/footer');
290     } else {
291         $this->Dosen_model->ubahDataDosen($dosen, $nik);
292         $this->session->set_flashdata('message', 'Diubah!');
293         redirect('operation/dosen');
294     }
295 }
```

Keterangan program:

Kode program diatas adalah fungsi bernama editdosen() yang terdapat pada Controller Operation.php untuk me-load view halaman edit dosen.

Baris 268 digunakan untuk memberikan sebuah judul pada halaman editdosen.php.

Baris 269-270 digunakan untuk mengambil data user berdasarkan session email.

Baris 271-272 digunakan untuk mengambil nama_role pada data user_role berdasarkan session id.

Baris 273-274 digunakan untuk mengambil data dosen berdasarkan nik melalui Dosen_model pada fungsi getDosenById().

Baris 275 digunakan untuk penanganan validasi terhadap form nama lengkap yang diinputkan, apabila yang diinputkan pada form nama lengkap tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (harus diisi) dan trim (tidak boleh ada spasi dibelakang).

Baris 276-279 digunakan untuk penanganan validasi terhadap form passwordddosen1 yang diinputkan, apabila yang diinputkan pada form passwordddosen1 tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (harus diisi), trim (tidak boleh ada spasi), min_length[3] (minimal password berisi 3 karakter), mathes[passwordddosen2] (passwordddosen1 harus sama dengan passwordddosen2).

Baris 280 digunakan untuk penanganan validasi terhadap form passworddosen2 yang diinputkan, apabila yang diinputkan pada form passworddosen2 tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (tidak boleh kosong), trim (tidak boleh ada spasi), dan mathes[passworddosen1] (passworddosen2 harus sama dengan password).

Baris 281 digunakan untuk penanganan validasi terhadap form mengajar yang diinputkan, apabila yang diinputkan pada form mengajar tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (tidak boleh kosong) dan trim (tidak boleh ada spasi).

Baris 282 digunakan untuk penanganan validasi terhadap form email yang diinputkan, apabila yang diinputkan pada form password tidak sesuai rules yang telah ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (tidak boleh kosong), trim (tidak boleh ada spasi), valid_email (email yang diinputkan harus sesuai dengan struktur penulisan email yang valid), dan is_unique (tidak boleh sama dengan email lain yang pernah diinput).

Baris 283 digunakan untuk penanganan validasi terhadap form hp yang diinputkan, apabila yang diinputkan pada form hp tidak sesuai rules yang ditentukan, maka akan memunculkan pesan error, rules nya yaitu required (harus diisi) dan trim (tidak boleh ada spasi).

Baris 284-289 digunakan apabila validasi form yang sudah dijelaskan diatas tidak sesuai dengan rules yang telah ditentukan maka akan meload view edit dosen beserta pesan error pada form yang tidak sesuai rules.

Baris 290-295 digunakan apabila yang diinputkan benar maka akan masuk ke dalam Dosen_model pada fungsi ubahDataDosen() untuk penanganan lebih lanjut, kemudian akan memunculkan pesan data telah berhasil diubah di halaman dosen.

Model : Dosen_model.php (fungsi getDosenById())

```
10 public function getDosenById($nik)
11 {
12     return $this->db->get_where('dosen', ['nik' => $nik])->row_array();
13 }
```

Keterangan program :

Fungsi diatas digunakan untuk mengambil serta menampilkan data dosen dari tabel dosen di database yang diambil berdasarkan nik.

Model : Dosen_model.php (fungsi ubahDataDosen())

```
49 public function ubahDataDosen($dosen, $nik)
50 {
51     $upload_image = $_FILES['imagedosen']['name'];
52     if ($upload_image) {
53         $config['allowed_types'] = 'gif|jpg|png';
54         $config['max_size'] = '2048';
55         $config['upload_path'] = './assets/img/profile/dosen';
56         $this->load->library('upload', $config);
57         if ($this->upload->do_upload('imagedosen')) {
58             $old_image = $dosen['image'];
59             if ($old_image != 'default.jpg') {
60                 unlink(FCPATH . 'assets/img/profile/dosen/' . $old_image);
61             }
62             $new_image = $this->upload->data('file_name');
63             $this->db->set('image', $new_image);
64         } else {
65             echo $this->upload->display_errors();
66         }
67     }
68     $name = $this->input->post('namalengkap', true);
69     $jk = $this->input->post('jk', true);
70     $mengajar = $this->input->post('mengajar', true);
71     $email = $this->input->post('email', true);
72     $hp = $this->input->post('hp', true);
73     $password = password_hash($this->input->post('passworddosen1'), PASSWORD_DEFAULT);
74     $is_active = $this->input->post('aktifdosen', true);
75     $data = [
76         'name' => $name,
77         'jk' => $jk,
78         'mengajar' => $mengajar,
79         'email' => $email,
80         'hp' => $hp,
81         'is_active' => $is_active
82     ];
83     $this->db->set($data);
84     if ($this->input->post('passworddosen1') != null) {
85         $this->db->set('password', $password);
86     }
87     $this->db->where('nik', $nik);
88     $this->db->update('dosen');
89 }
```

Keterangan program :

Baris 50 – 67 digunakan untuk penanganan apabila terdapat pengubahan gambar, dimana gambar baru akan di upload menggantikan gambar yang lama (akan dijelaskan lebih lanjut pada nomor selanjutnya yang membahas tentang upload).

Baris 68 – 82 digunakan untuk membungkus data dosen (dibungkus oleh variabel \$data) yang telah diubah melalui form dan telah dicek oleh controller.

Baris 83 digunakan untuk mengubah data sesuai variabel data yang telah dibungkus diatas.

Baris 84-86 digunakan apabila password diubah maka akan mengubah kolom password di database.

Baris 87-88 digunakan untuk mengupdate data dosen di database berdasarkan nik yang dipilih untuk diubah.

Source code dibawah ini adalah source dari front end ubahdosen.php yang telah di-load oleh controller.

View : ubahdosen.php

```
<!-- Begin Page Content -->
<div class="container-fluid">
  <!-- Page Heading -->
  <h1 class="h3 mb-4 text-gray-800">Form Edit Dosen</h1>
  <div class="row">
    <div class="col-lg-8">
      <?= form_open_multipart(); ?>
      <input type="hidden" name="nik" value="<?= $dosen['nik']; ?>">
      <div class="form-group row">
        <label for="nik" class="col-sm-3 col-form-label">NIK</label>
        <div class="col-sm-9">
          <input type="text" class="form-
control" id="nik" name="nik" value="<?= $dosen['nik']; ?>" readonly>
          <?= form_error('nik', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
      </div>
      <div class="form-group row">
        <label for="namalengkap" class="col-sm-3 col-form-
label">Nama Lengkap</label>
        <div class="col-sm-9">
          <input type="text" class="form-
control" id="namalengkap" name="namalengkap" value="<?= $dosen['name']; ?>">
          <?= form_error('namalengkap', ' <small class="text-
danger pl-3">', '</small>'); ?>
        </div>
      </div>
      <div class="form-group row">
        <label class="col-sm-3 col-form-label">Password</label>
        <div class="col-sm-9">
          <input type="password" class="form-
control" id="passworddosen1" name="passworddosen1">
          <?= form_error('passworddsn1', ' <small class="text-
danger pl-3">', '</small>'); ?>
        </div>
      </div>
      <div class="form-group row">
        <label class="col-sm-3 col-form-label">Repeat Password</label>
        <div class="col-sm-9">
          <input type="password" class="form-
control" id="passworddosen2" name="passworddosen2">
        </div>
      </div>
      <div class="form-group row">
```

```

        <?php
        $jk = $dosen['jk']; ?>
        <label for="jk" class="col-sm-3 col-form-
label">Jenis Kelamin</label>
        <div class="form-check form-check-inline pl-3">
            <input type="radio" name="jk" <?php if ($jk == 'L') {
                echo 'checked';
            } ?> value="L">
            <label class="form-check-label" for="jk">Laki-laki</label>
        </div>
        <div class="form-check form-check-inline">
            <input type="radio" name="jk" <?php if ($jk == 'P') {
                echo 'checked';
            } ?> value="P">
            <label class="form-check-label" for="jk">Perempuan</label>
        </div>
    </div>
    <div class="form-group row">
        <label for="mengajar" class="col-sm-3 col-form-
label">Mengajar</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="mengajar" name="mengajar" value="<?=$dosen['mengajar']; ?>"
            <?=$form_error('mengajar', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="email" class="col-sm-3 col-form-
label">Email</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="email" name="email" value="<?=$dosen['email']; ?>"
            <?=$form_error('email', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">
        <label for="hp" class="col-sm-3 col-form-
label">Handphone</label>
        <div class="col-sm-9">
            <input type="text" class="form-
control" id="hp" name="hp" value="<?=$dosen['hp']; ?>"
            <?=$form_error('hp', ' <small class="text-danger pl-
3">', '</small>'); ?>
        </div>
    </div>
    <div class="form-group row">

```

```

        <?php
        $aktif = $dosen['is_active']; ?>
        <label for="aktif" class="col-sm-3 col-form-
label">Aktif</label>
        <div class="form-check form-check-inline pl-3">
            <input type="radio" name="aktifdosen" <?php if ($aktif ==
'1') {
                                                    echo 'checked'
;
                                                    } ?> value="1">
            <label class="form-check-
label" for="aktifdosen">Ya</label>
        </div>
        <div class="form-check form-check-inline">
            <input type="radio" name="aktifdosen" <?php if ($aktif ==
'0') {
                                                    echo 'checked'
;
                                                    } ?> value="0">
            <label class="form-check-
label" for="aktifdosen">Tidak</label>
        </div>
    </div>
    <div class="form-group row">
        <div class="col-sm-3">Foto</div>
        <div class="col-sm-9">
            <div class="row">
                <div class="col-sm-3">
                    
                </div>
            </div>
        </div>
    </div>
    <div class="form-group row">
        <div class="col-sm-3">Ganti Foto</div>
        <div class="col-sm-9">
            <div class="custom-file">
                <input type="file" class="custom-file-
input" id="imagedosen" name="imagedosen">
                <label class="custom-file-
label" for="image">Pilih foto</label>
            </div>
        </div>
    </div>
    <div class="form-group row justify-content-end">
        <div class="col-sm-9">

```

```

        <button type="submit" class="btn btn-
primary">Perbarui</button>
        <button type="reset" class="btn btn-
warning">Reset</button>
        <a href="<?= base_url('operation/dosen'); ?>" class="btn b
tn-danger">Kembali</a>
    </div>
</div>
</form>
</div>
</div>
<!-- /.container-fluid -->
</div>
<!-- End of Main Content -->

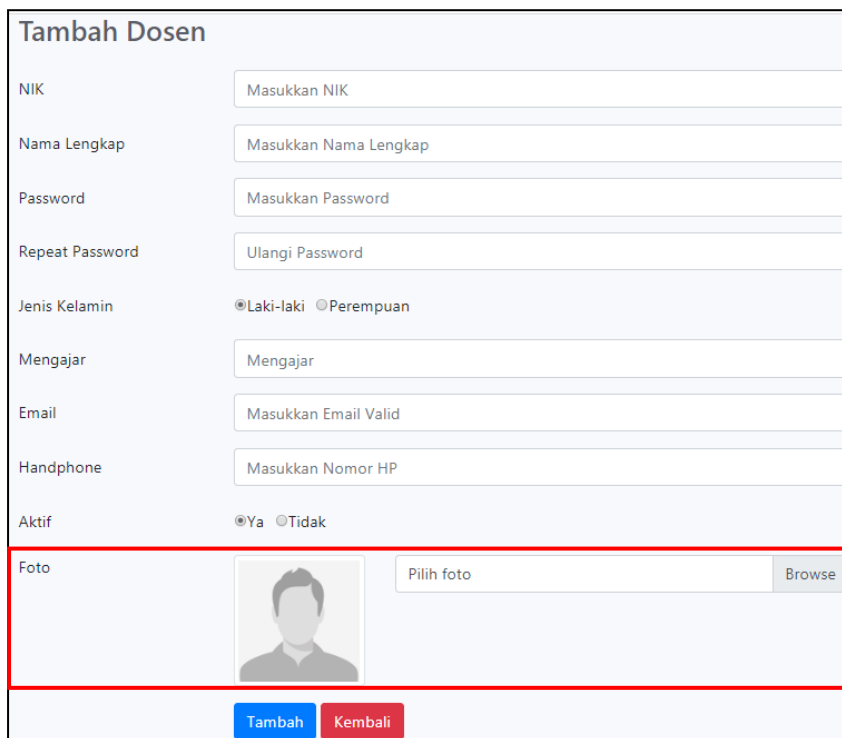
```

View diatas adalah view untuk menampilkan halaman ubah data dosen/karyawan.

7. Pada aplikasi tentunya memiliki alur Upload Gambar (pilih salah satu). Silahkan berikan penjelasan alur logika Upload Gambar aplikasi anda, sertakan source code php yang memperlihatkan keterkaitan antara frontend, database dan backend pada tahap Upload Gambar.

Jawaban :

Upload Gambar yang akan dijelaskan yaitu mengenai **upload gambar dosen/karyawan**. Tampilan halaman **upload gambar dosen/karyawan** pada website saya adalah seperti dibawah ini :



The screenshot shows a web form titled "Tambah Dosen". It contains several input fields: NIK (Masukkan NIK), Nama Lengkap (Masukkan Nama Lengkap), Password (Masukkan Password), Repeat Password (Ulangi Password), Jenis Kelamin (radio buttons for Laki-laki and Perempuan), Mengajar (Masukkan Mengajar), Email (Masukkan Email Valid), Handphone (Masukkan Nomor HP), and Aktif (radio buttons for Ya and Tidak). At the bottom, there is a "Foto" section with a placeholder image of a person and a "Pilih foto" button with a "Browse" link. This entire "Foto" section is enclosed in a red rectangular border. Below the form, there are two buttons: "Tambah" (blue) and "Kembali" (red).

Alur upload gambar :

Pilih foto yang diinginkan dari komputer (jika dibiarkan kosong maka foto akan menggunakan gambar default). Kemudian gambar yang telah dipiih tadi akan dicek oleh Controller. Berikut adalah source code controller :

Kode program :

Controller : Operation.php (fungsi tambahDataDosen())

```
33 public function tambahDataDosen()
34 {
35     $nik = $this->input->post('nik', true);
36     $data = [
37         'nik' => $nik,
38         'name' => $this->input->post('namalengkap', true),
39         'password' => password_hash($this->input->post('password'), PASSWORD_DEFAULT),
40         'jk' => $this->input->post('jk', true),
41         'mengajar' => $this->input->post('mengajar', true),
42         'email' => $this->input->post('email', true),
43         'hp' => $this->input->post('hp', true),
44         'is_active' => $this->input->post('aktif', true),
45         'image' => $this->upload(),
46         'role_id' => 4
47     ];
48     $this->db->insert('dosen', $data);
49 }
```

Fungsi pertama yang akan dijalankan adalah fungsi tambahDataDosen(), kemudian pada baris 45 gambar yang telah diinput kan dibungkus oleh variabel \$data yang selanjutnya akan menjalankan fungsi upload(). Setelah melewati fungsi upload maka akan menjalankan baris 48 yaitu melakukan insert ke tabel dosen di database.

Controller : Operation.php (fungsi upload())

```
15 public function upload()
16 {
17     $upload_image = $_FILES['imagedosen']['name'];
18     if ($upload_image) {
19         $config['allowed_types'] = 'gif|jpg|png';
20         $config['max_size'] = '2048';
21         $config['upload_path'] = './assets/img/profile/dosen';
22         $this->load->library('upload', $config);
23         if ($this->upload->do_upload('imagedosen')) {
24             return $this->upload->data('file_name');
25         }
26     }
27     return "default.jpg";
28 }
```

Pada Controller.php, fungsi upload() digunakan untuk mengupload gambar beserta ketentuannya yaitu :

Baris 17 digunakan untuk mengambil nama file gambar yang telah dipilih oleh user.

Baris 18 digunakan untuk pengkondisian yaitu jika file gambar yang dipilih memiliki ketentuan sebagai berikut :

Baris 19 digunakan untuk menentukan bahwa file gambar yang dipilih oleh user memiliki tipe gambar gif, jpg atau png.

Baris 20 digunakan untuk menentukan bahwa file gambar yang dipilih oleh user memiliki ukuran maksimal 2048 mb.

Baris 21 digunakan untuk menentukan bahwa file gambar yang dipilih oleh user akan ditempatkan ke dalam folder assets/img/profile/dosen.

Baris 22 digunakan untuk me-load library upload dengan ketentuan yang telah dijelaskan diatas.

Baris 23-24 digunakan untuk melakukan upload sesuai gambar yang dipilih user dan sudah sesuai dengan ketentuan yang telah ditentukan diatas.

Baris 27 digunakan untuk apabila user tidak memilih gambar dari komputer, maka gambar yang digunakan adalah gambar default.

8. Bagaimana cara membuat user level pada aplikasi anda?

Jawaban :

Pada sistem ini mempunyai 5 aktor, diantaranya Superadmin, Admin, Pimpinan, Dosen/Karyawan dan Mahasiswa.

a. Super Admin

Pada role ini, superadmin mempunyai peran untuk mengontrol penuh sistem serta merupakan tingkatan tertinggi yang dapat mengatur user lain seperti Admin, Pimpinan, Dosen/Karyawan dan Mahasiswa.

Role Access

Role : SuperAdmin

Tampilkan 10 ▾ Data

Pencarian:

#	Menu	Access
1	User	<input checked="" type="checkbox"/>
2	Menu	<input checked="" type="checkbox"/>
3	Operation	<input checked="" type="checkbox"/>
4	Super Admin	<input checked="" type="checkbox"/>
5	Coblos	<input type="checkbox"/>
6	(user) Komentar	<input type="checkbox"/>
7	Riwayat	<input checked="" type="checkbox"/>
8	Administrator	<input checked="" type="checkbox"/>

Total 8 Data ditampilkan (1 sampai 8)

Sebelumnya

1

Selanjutnya

Berdasarkan tabel Role Access diatas, superadmin memiliki akses yaitu pada menu user, menu, operation, super admin, riwayat, dan administrator. Superadmin dapat membuat menu dan submenu baru, membuat role baru serta memberikan akses pada tiap-tiap role, superadmin juga dapat membuatkan akun untuk admin.

b. Admin

Admin mempunyai peran tidak jauh berbeda dengan superadmin namun letak perbedaannya adalah admin tidak dapat mengontrol atau mengatur isi menu-menu user serta tidak dapat memberikan hak akses role. Akun admin dibuat oleh superadmin.

Role Access		
Role : Admin		
Tampilkan	10	Data
		Pencarian: <input type="text"/>
#	Menu	Access
1	User	<input checked="" type="checkbox"/>
2	Menu	<input type="checkbox"/>
3	Operation	<input checked="" type="checkbox"/>
4	Super Admin	<input type="checkbox"/>
5	Coblos	<input type="checkbox"/>
6	(user) Komentar	<input type="checkbox"/>
7	Riwayat	<input checked="" type="checkbox"/>
8	Administrator	<input type="checkbox"/>
Total 8 Data ditampilkan (1 sampai 8)		Sebelumnya 1 Selanjutnya

Berdasarkan tabel Role Access diatas, admin memiliki akses yaitu pada menu user, operation dan riwayat. Admin dapat melakukan CRUD pada menu operation. Admin adalah tingkatan kedua setelah superadmin. Admin diberikan hak akses oleh superadmin.

c. Pimpinan

User yang digunakan oleh pimpinan STIKOM.

Role Access			
Role : Pimpinan			
Tampilkan 10 Data		Pencarian:	
#	Menu	Access	
1	User	<input checked="" type="checkbox"/>	
2	Menu	<input type="checkbox"/>	
3	Operation	<input type="checkbox"/>	
4	Super Admin	<input type="checkbox"/>	
5	Coblos	<input checked="" type="checkbox"/>	
6	(user) Komentar	<input checked="" type="checkbox"/>	
7	Riwayat	<input type="checkbox"/>	
8	Administrator	<input type="checkbox"/>	
Total 8 Data ditampilkan (1 sampai 8)		Sebelumnya 1 Selanjutnya	

Berdasarkan tabel Role Access diatas, pimpinan memiliki akses yaitu pada menu user, coblos dan user (komentar). Pimpinan dibuat oleh admin/superadmin. Pimpinan dapat mecoblos calon kandidat senat, mengubah profil dan menambahkan komentar.

d. Dosen/karyawan

User yang digunakan oleh dosen dan karyawan STIKOM.

Role Access		
Role : Dosen		
Tampilkan	10	Data
Pencarian:		
#	Menu	Access
1	User	<input checked="" type="checkbox"/>
2	Menu	<input type="checkbox"/>
3	Operation	<input type="checkbox"/>
4	Super Admin	<input type="checkbox"/>
5	Coblos	<input checked="" type="checkbox"/>
6	(user) Komentar	<input checked="" type="checkbox"/>
7	Riwayat	<input type="checkbox"/>
8	Administrator	<input type="checkbox"/>
Total 8 Data ditampilkan (1 sampai 8)		Sebelumnya 1 Selanjutnya

Berdasarkan tabel Role Access diatas, dosen memiliki akses yaitu pada menu user, coblos dan user (komentar). Dosen/karyawan dibuat oleh admin/superadmin. Dosen/karyawan dapat mecoblos calon kandidat senat, mengubah profil dan menambahkan komentar.

e. Mahasiswa

User yang digunakan oleh mahasiswa STIKOM.

Role Access			
Role : Mahasiswa			
Tampilkan	10	Data	Pencarian: <input type="text"/>
#	Menu	Access	
1	User	<input checked="" type="checkbox"/>	
2	Menu	<input type="checkbox"/>	
3	Operation	<input type="checkbox"/>	
4	Super Admin	<input type="checkbox"/>	
5	Coblos	<input checked="" type="checkbox"/>	
6	(user) Komentar	<input checked="" type="checkbox"/>	
7	Riwayat	<input type="checkbox"/>	
8	Administrator	<input type="checkbox"/>	
Total 8 Data ditampilkan (1 sampai 8)			Sebelumnya 1 Selanjutnya

Berdasarkan tabel Role Access diatas, dosen memiliki akses yaitu pada menu user, coblos dan user (komentar). Mahasiswa dibuat oleh admin/superadmin. Mahasiswa dapat mecoblos calon kandidat senat, mengubah profil dan menambahkan komentar.

9. Berikan penjelasan langkah anda setelah lulus sebagai sarjana komputer dan jelaskan ilmu apa saja yang telah anda dapatkan untuk bersaing di industri IT secara nyata di STIKOM YOS Sudarso Purwokerto.

Jawaban :

Setelah lulus saya akan melamar pekerjaan dengan cara mencari lowongan kerja yang membutuhkan sarjana komputer terutama di bidang sistem informasi. Ilmu yang saya dapatkan selama kuliah di STIKOM Yos Sudarso yaitu :

- Algoritma dan pemrograman (tentang bahasa pemrograman C++ untuk dasar pengenalan pemrograman)
- Algoritma dan struktur data (tentang penggunaan array, queue, stuck, linked list pada bahasa pemrograman C++)
- Database (tentang DBMS yaitu MySQL agar dapat melakukan CRUD, Trigger, View, JOIN, dan fungsi-fungsi lain)

- Pemrograman I, II, dan III (tentang bahasa pemrograman C# dan Java untuk membuat aplikasi android)
- Pemrograman Web (tentang bahasa pemrograman PHP, javascript, codeigniter, css dan bootstrap untuk membuat website)
- Analisis dan Perancangan Sistem Informasi (tentang metode-metode pengembangam sistem serta perancangan design UML)
- Pengujian dan Implementasi Sistem Informasi (tentang berbagai pengujian yang dilakukan untuk kelayakan sebuah perangkat lunak)
- Jaringan Komputer (tentang pembuatan kabel LAN, sharing antar komputer dan lain lain)
- Aplikasi Komputer (tentang pengoperasian berbagai aplikasi yang sering digunakan dalam perusahaan seperti Microsoft Word, Microsoft Excel, Microsoft Powerpoint, CorelDraw, dan aplikasi-aplikasi lain.
- Interaksi manusia dan komputer (tentang User experience dan user interface).
- Dan mata kuliah lainnya seperti akuntansi, pengetahuan bisnis, manajemen sistem informasi, perpajakan dan lainnya.