LAMPIRAN

- **Berkas**

```
<?php
if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Berkas extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Berkas_model');
        $this->load->model('Pengaturan_model');
        $this->load->library('form_validation');
          $this->load->library('datatables');
    }

    public function index()
    {
        $data['title'] = 'Berkas';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Berkas' => '',
        ];

        $data['code_js'] = 'berkas/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'berkas/Berkas_list';
        $this->load->view('template/backend', $data);
    }
```

```
public function json() {
    header('Content-Type: application/json');
    echo $this->Berkas_model->json();
}

public function read($id)
{
    $row = $this->Berkas_model->get_by_id($id);
    if ($row) {
        $data = array(
                          'id_berkas' => $row->id_berkas,
                          'nama_berkas' => $row->nama_berkas,
                          'keterangan_berkas' => $row->keterangan_berkas,
                          'tipe_berkas' => $row->tipe_berkas,
                          'ukuran_berkas' => $row->ukuran_berkas,
                          'id_peserta' => $row->id_peserta,
            'nama_peserta' => $row->nama_peserta,
          );

    $data['title'] = 'Berkas';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['page'] = 'berkas/Berkas_read';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('berkas'));
    }
    }

    public function delete($id)
    {
        $row = $this->Berkas_model->get_by_id($id);

        if ($row) {
            $this->Berkas_model->delete($id);
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus berkas ".$row->keterangan_berkas." ".$row-
>nama_peserta);
            redirect(site_url('berkas'));
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('berkas'));
        }
    }

    public function deletebulk(){
        $delete = $this->Berkas_model->deletebulk();
        if($delete){
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
        }else{
            $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
        }
        echo $delete;
    }
    public function _rules()
    {
      $this->form_validation->set_rules('nama_berkas', 'nama berkas', 'trim|required');
      $this->form_validation->set_rules('keterangan_berkas', 'keterangan berkas',
'trim|required');
      $this->form_validation->set_rules('tipe_berkas', 'tipe berkas', 'trim|required');
      $this->form_validation->set_rules('ukuran_berkas', 'ukuran berkas', 'trim|required');
      $this->form_validation->set_rules('id_peserta', 'id peserta', 'trim|required');


      $this->form_validation->set_rules('id_berkas', 'id_berkas', 'trim');
      $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');
    }
}
/* End of file Berkas.php */
```

- Bobot

```
<?php
if (!defined('BASEPATH'))
    exit('No direct script access allowed');
class Bobot extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Bobot_model');
        $this->load->model('Pengaturan_model');
        $this->load->model('Jalur_model');
        $this->load->library('form_validation');
          $this->load->library('datatables');
    }

    public function index()
    {
        $data = array(
            'button' => 'Tambah',
            'action' => site_url('bobot/create_action'),
            'id_bobot' => set_value('id_bobot'),
            'id_jalur' => set_value('id_jalur'),
            'jalur' => set_value('jalur'),
            'bobot_jarak' => set_value('bobot_jarak'),
            'bobot_nilai' => set_value('bobot_nilai'),
            'bobot_prestasi' => set_value('bobot_prestasi'),
        );

        $data['title'] = 'Bobot';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Bobot' => '',
        ];
        $data['code_js'] = 'bobot/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['jalur'] =$this->Jalur_model->get_all();
        $data['page'] = 'bobot/Bobot_list';
        $this->load->view('template/backend', $data);
    }
    public function json() {
        header('Content-Type: application/json');
        echo $this->Bobot_model->json();
    }

    public function read($id)
    {
        $row = $this->Bobot_model->get_by_id($id);
        if ($row) {
        $data = array(
                'id_bobot' => $row->id_bobot,
                  'id_jalur' => $row->id_jalur,
            'jalur' => $row->jalur,
                  'bobot_jarak' => $row->bobot_jarak,
                  'bobot_nilai' => $row->bobot_nilai,
                  'bobot_prestasi' => $row->bobot_prestasi,
        );

        $data['title'] = 'Bobot';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'bobot/Bobot_read';
        $this->load->view('template/backend', $data);
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('bobot'));
        }
    }
    public function create()
    {
        $data = array(
            'button' => 'Tambah',
            'action' => site_url('bobot/create_action'),
          'id_bobot' => set_value('id_bobot'),
          'id_jalur' => set_value('id_jalur'),
          'bobot_jarak' => set_value('bobot_jarak'),
          'bobot_nilai' => set_value('bobot_nilai'),
          'bobot_prestasi' => set_value('bobot_prestasi'),
      );
```

```
        $data['title'] = 'Bobot';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];

        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['jalur'] =$this->Jalur_model->get_all();
        $data['page'] = 'bobot/Bobot_form';
        $this->load->view('template/backend', $data);
    }
    public function create_action()
    {
        $this->_rules();

        if ($this->form_validation->run() == FALSE) {
            // $this->create();
            $this->session->set_flashdata('message',
                form_error('id_jalur').
                form_error('bobot_jarak').
                form_error('bobot_nilai').
                form_error('bobot_prestasi')
            );
            redirect(site_url('bobot'));
        } else {
            $data = array(
                              'id_jalur' => $this->input->post('id_jalur',TRUE),
                              'bobot_jarak' => $this->input->post('bobot_jarak',TRUE),
                              'bobot_nilai' => $this->input->post('bobot_nilai',TRUE),
                              'bobot_prestasi' => $this->input->post('bobot_prestasi',TRUE),
                      );
        $this->Bobot_model->insert($data);
        $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
        helper_log("add", "Menambah data bobot jalur pendaftaran");
        redirect(site_url('bobot'));}
}
public function update($id)
{
    $row = $this->Bobot_model->get_by_id($id);

    if ($row) {
    $data = array(
        'button' => 'Ubah',
        'action' => site_url('bobot/update_action'),
              'id_bobot' => set_value('id_bobot', $row->id_bobot),
              'id_jalur' => set_value('id_jalur', $row->id_jalur),
        'jalur' => set_value('jalur', $row->jalur),
              'bobot_jarak' => set_value('bobot_jarak', $row->bobot_jarak),
              'bobot_nilai' => set_value('bobot_nilai', $row->bobot_nilai),
              'bobot_prestasi' => set_value('bobot_prestasi', $row->bobot_prestasi),
    );

    $data['title'] = 'Bobot';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'bobot/Bobot_form';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
```

```
            redirect(site_url('bobot'));
        }
    }

    public function update_action()
    {
        // $this->_rules();
        $this->form_validation->set_rules('bobot_jarak', 'bobot jarak',
'trim|required|numeric',
            array(
                'required'      => 'Bobot Jarak tidak boleh kosong ',
                'numeric'       => 'Bobot Jarak hanya angka '
            ));
        $this->form_validation->set_rules('bobot_nilai', 'bobot nilai', 'trim|required|numeric',
            array(
                'required'      => 'Bobot Nilai tidak boleh kosong ',
                'numeric'       => 'Bobot Nilai hanya angka '
            ));
        $this->form_validation->set_rules('bobot_prestasi', 'bobot prestasi',
'trim|required|numeric',
            array(
                'required'      => 'Bobot Prestasi tidak boleh kosong ',
                'numeric'       => 'Bobot Prestasi hanya angka '
            ));

        $this->form_validation->set_rules('id_bobot', 'id_bobot', 'trim');
        $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');

        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_bobot', TRUE));
        } else {
            $data = array(
                    // 'id_jalur' => $this->input->post('id_jalur',TRUE),
                    'bobot_jarak' => $this->input->post('bobot_jarak',TRUE),
                    'bobot_nilai' => $this->input->post('bobot_nilai',TRUE),
                    'bobot_prestasi' => $this->input->post('bobot_prestasi',TRUE),
                );

            $this->Bobot_model->update($this->input->post('id_bobot', TRUE), $data);
            $this->session->set_flashdata('message', 'Data Berhasil diubah');
            helper_log("edit", "Update data bobot jalur pendaftaran");
            redirect(site_url('bobot'));
        }
    }

    public function delete($id)
    {
        $row = $this->Bobot_model->get_by_id($id);

        if ($row->id_bobot=='1' or $row->id_bobot=='2' or $row->id_bobot=='3' or $row-
>id_bobot=='4') {
            $this->session->set_flashdata('message', 'Data Gagal dihapus');
            redirect(site_url('bobot'));
        } else {
            $this->Bobot_model->delete($id);
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus data bobot jalur ".$row->jalur);
            redirect(site_url('bobot'));
        }
    }

    public function deletebulk(){
        $delete = $this->Bobot_model->deletebulk();
        if($delete){
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Hapus multi data bobot jalur pendaftaran");
        }else{
            $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
        }
        echo $delete;
    }

    public function _rules()
    {
    $this->form_validation->set_rules('id_jalur', 'id jalur',
'trim|required|is_unique[bobot.id_jalur]',
        array(
                'required'      => 'Jalur Pendaftaran tidak boleh kosong ',
                'is_unique'     => 'Jalur Pendaftaran sudah ada '
        ));
    $this->form_validation->set_rules('bobot_jarak', 'bobot jarak', 'trim|required|numeric',
        array(
                'required'      => 'Bobot Jarak tidak boleh kosong ',
                'numeric'       => 'Bobot Jarak hanya angka '
        ));
    $this->form_validation->set_rules('bobot_nilai', 'bobot nilai', 'trim|required|numeric',
        array(
                'required'      => 'Bobot Nilai tidak boleh kosong ',
                'numeric'       => 'Bobot Nilai hanya angka '
        ));
    $this->form_validation->set_rules('bobot_prestasi', 'bobot prestasi',
'trim|required|numeric',
        array(
                'required'      => 'Bobot Prestasi tidak boleh kosong ',
                'numeric'       => 'Bobot Prestasi hanya angka '
        ));
```

```
  $this->form_validation->set_rules('id_bobot', 'id_bobot', 'trim');
  $this->form_validation->set_error_delimiters('<span class="text">', '</span>');
}

public function excel()
{
    $this->load->helper('exportexcel');
    $namaFile = "bobot.xls";
    $judul = "bobot";
    $tablehead = 0;
    $tablebody = 1;
    $nourut = 1;
    //penulisan header
    header("Pragma: public");
    header("Expires: 0");
    header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
    header("Content-Type: application/force-download");
    header("Content-Type: application/octet-stream");
    header("Content-Type: application/download");
    header("Content-Disposition: attachment;filename=" . $namaFile . "");
    header("Content-Transfer-Encoding: binary ");

    xlsBOF();

    $kolomhead = 0;
    xlsWriteLabel($tablehead, $kolomhead++, "No");
  xlsWriteLabel($tablehead, $kolomhead++, "Jalur");
  xlsWriteLabel($tablehead, $kolomhead++, "Bobot Jarak");
  xlsWriteLabel($tablehead, $kolomhead++, "Bobot Nilai");
  xlsWriteLabel($tablehead, $kolomhead++, "Bobot Prestasi");

  foreach ($this->Bobot_model->get_all() as $data) {
    $kolombody = 0;

    //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
    xlsWriteNumber($tablebody, $kolombody++, $nourut);
      xlsWriteLabel($tablebody, $kolombody++, $data->jalur);
      xlsWriteNumber($tablebody, $kolombody++, $data->bobot_jarak);
      xlsWriteNumber($tablebody, $kolombody++, $data->bobot_nilai);
      xlsWriteNumber($tablebody, $kolombody++, $data->bobot_prestasi);

      $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}

public function word()
{
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=bobot.doc");

    $data = array(
        'bobot_data' => $this->Bobot_model->get_all(),
        'start' => 0
    );
    $this->load->view('bobot/Bobot_doc',$data);
}

public function printdoc(){
    $data = array(
        'bobot_data' => $this->Bobot_model->get_all(),
            'start' => 0
        );
        $this->load->view('bobot/Bobot_print', $data);
    }
}
/* End of file Bobot.php */
```

- Dashboard

```
?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Dashboard extends CI_Controller {

      public function __construct()
      {
                  parent::__construct();
                  $this->layout->auth();
        $this->load->library(array('googlemaps'));
                  $this->load->model('Users_model');
        $this->load->model('Users_groups_model');
        $this->load->model('Peserta_model');
        $this->load->model('Sekolah_model');
        $this->load->model('Jalur_model');
        $this->load->model('Jarak_model');
        $this->load->model('Jurusan_model');
        $this->load->model('Tahunpelajaran_model');
                  $this->load->model('Formulir_model');
        $this->load->model('Pengumuman_model');
        $this->load->model('Pengaturan_model');
        $this->load->model('Jurusan_model');
        $this->load->model('Berkas_model');
        $this->load->model('Log_model');
```

```
}

public function index()
{
            $data['title'] = 'Dashboard';
            $data['subtitle'] = '';
  $data['crumb'] = [
      'Dashboard' => '',
  ];

  //$this->layout->set_privilege(1);
  $data['code_js'] = 'Dashboard/codejs';
  $data['log'] = $this->Log_model->get_log();
  $data['peserta'] = $this->Peserta_model->get_all();
  $data['countjurusan'] = $this->Peserta_model->get_countjurusan();
  $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();
  $data['formulir'] =  $this->Formulir_model->get_by_id_1();
  $data['nomer'] = $this->Peserta_model->nodaftar();

  if ($data['nomer']) {
      $id=$data['nomer']->id_peserta;
      $data['berkas'] = $this->Berkas_model->get_id($id);
  }

  $data['group']=$this->Users_groups_model->get_id();
  $data['infomember']=$this->Pengumuman_model->get_by_member();
  $data['contact'] = $this->Users_model->get_panitia();

  //total peserta tahun aktif
  $data['totalpeserta'] =  $this->Peserta_model->hitungDataPeserta();
  $data['totaljalur1'] =  $this->Peserta_model->hitungDataJalur1();
  $data['totaljalur2'] =  $this->Peserta_model->hitungDataJalur2();
```

```
$data['totaljalur3'] =  $this->Peserta_model->hitungDataJalur3();
$data['totaljalur4'] =  $this->Peserta_model->hitungDataJalur4();
$data['totaldiverifikasi'] =  $this->Peserta_model->hitungDataVerifikasi();
$data['totalbaru'] =  $this->Peserta_model->hitungDataBaru();
$data['totalberkaskurang'] =  $this->Peserta_model->hitungDataKurang();

$data['jalur'] =  $this->Jalur_model->get_all();

//kuota peserta tahun aktif
$data['kuotax'] =  $this->Tahunpelajaran_model->get_tahun_aktif();
if (empty($data['kuotax'])) {
  $data['kuota']='0';
} else {
  $data['kuota']=$data['kuotax']->kuota;
}

$data['jalur1'] =  $this->Jalur_model->KuotaJalur1();
$data['jalur2'] =  $this->Jalur_model->KuotaJalur2();
$data['jalur3'] =  $this->Jalur_model->KuotaJalur3();
$data['jalur4'] =  $this->Jalur_model->KuotaJalur4();

$data['KuotaJalur1']=$data['kuota']*$data['jalur1']->persentase/100;
$data['KuotaJalur2']=$data['kuota']*$data['jalur2']->persentase/100;
$data['KuotaJalur3']=$data['kuota']*$data['jalur3']->persentase/100;
$data['KuotaJalur4']=$data['kuota']*$data['jalur4']->persentase/100;

// progress bar tahun aktif
// bar jalur 1
if (empty($data['KuotaJalur1'])) {
    $data['barjalur1']='0';
} else {
```

```
$data['barjalur1']=($data['totaljalur1']!=0)?($data['totaljalur1']/$data['KuotaJalur1'])*100:
0;
        }
        // bar jalur 2
        if (empty($data['KuotaJalur2'])) {
            $data['barjalur2']='0';
        } else {

$data['barjalur2']=($data['totaljalur2']!=0)?($data['totaljalur2']/$data['KuotaJalur2'])*100:
0;
        }
        // bar jalur 3
        if (empty($data['KuotaJalur3'])) {
            $data['barjalur3']='0';
        } else {

$data['barjalur3']=($data['totaljalur3']!=0)?($data['totaljalur3']/$data['KuotaJalur3'])*100:
0;
        }
        // bar jalur 4
        if (empty($data['KuotaJalur4'])) {
            $data['barjalur4']='0';
        } else {
$data['barjalur4']=($data['totaljalur4']!=0)?($data['totaljalur4']/$data['KuotaJalur4'])*100:
0;
        }
        $data['page'] = 'Dashboard/Index';
      $this->load->view('template/backend', $data);
      }
}
```

- Formulir

```
<?php
```

```
if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Formulir extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Formulir_model');
        $this->load->model('Pengaturan_model');
        $this->load->model('Pengumuman_model');
        $this->load->model('Tahunpelajaran_model');
        $this->load->model('Jalur_model');
        $this->load->library('form_validation');
    }

    public function index()
    {
        $data['title'] = 'Formulir';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Formulir' => '',
        ];

        $data['code_js'] = 'formulir/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
                  $data['formulir'] =  $this->Formulir_model->get_by_id_1();
        $data['page'] = 'formulir/Formulir_atur';
        $this->load->view('template/backend', $data);
    }


    public function update_action()
    {
        $this->_rules();


        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_formulir', TRUE));
        } else {
            $data = array(
                                    // 'tahun_pelajaran' => $this->input-
>post('tahun_pelajaran',TRUE),
                                    // 'jalur_pendaftaran' => $this->input-
>post('jalur_pendaftaran',TRUE),
                                    // 'nama_peserta' => $this->input->post('nama_peserta',TRUE),
                                    // 'jenis_kelamin' => $this->input-
>post('jenis_kelamin',TRUE),
                                    // 'nisn' => $this->input->post('nisn',TRUE),
                                    'nik' => $this->input->post('nik',TRUE),
                                    // 'tempat_lahir' => $this->input->post('tempat_lahir',TRUE),
                                    // 'tanggal_lahir' => $this->input->post('tanggal_lahir',TRUE),
                                    'no_registrasi_akta_lahir' => $this->input-
>post('no_registrasi_akta_lahir',TRUE),
                                    // 'agama' => $this->input->post('agama',TRUE),
                                    'kewarganegaraan' => $this->input-
>post('kewarganegaraan',TRUE),
                                    'berkebutuhan_khusus' => $this->input-
>post('berkebutuhan_khusus',TRUE),
                                    // 'alamat' => $this->input->post('alamat',TRUE),
                                    'rt' => $this->input->post('rt',TRUE),
                                    'rw' => $this->input->post('rw',TRUE),
                                    'nama_dusun' => $this->input->post('nama_dusun',TRUE),
                                    'nama_kelurahan' => $this->input-
>post('nama_kelurahan',TRUE),
						'kecamatan' => $this->input->post('kecamatan',TRUE),
						'kode_pos' => $this->input->post('kode_pos',TRUE),
						'latitude' => $this->input->post('latitude',TRUE),
						'longitude' => $this->input->post('longitude',TRUE),
						'tempat_tinggal' => $this->input->post('tempat_tinggal',TRUE),
						'moda_transportasi' => $this->input->post('moda_transportasi',TRUE),
						'no_kks' => $this->input->post('no_kks',TRUE),
						'anak_ke' => $this->input->post('anak_ke',TRUE),
						'penerima_kps_pkh' => $this->input->post('penerima_kps_pkh',TRUE),
						'no_kps_pkh' => $this->input->post('no_kps_pkh',TRUE),
						'penerima_kip' => $this->input->post('penerima_kip',TRUE),
						'no_kip' => $this->input->post('no_kip',TRUE),
						'nama_tertera_di_kip' => $this->input->post('nama_tertera_di_kip',TRUE),
						'terima_fisik_kartu_kip' => $this->input->post('terima_fisik_kartu_kip',TRUE),
						// 'nama_ayah' => $this->input->post('nama_ayah',TRUE),
						'nik_ayah' => $this->input->post('nik_ayah',TRUE),
						'tahun_lahir_ayah' => $this->input->post('tahun_lahir_ayah',TRUE),
						'pendidikan_ayah' => $this->input->post('pendidikan_ayah',TRUE),
						'pekerjaan_ayah' => $this->input->post('pekerjaan_ayah',TRUE),
						'penghasilan_bulanan_ayah' => $this->input->post('penghasilan_bulanan_ayah',TRUE),
						'berkebutuhan_khusus_ayah' => $this->input->post('berkebutuhan_khusus_ayah',TRUE),
						// 'nama_ibu' => $this->input->post('nama_ibu',TRUE),
						'nik_ibu' => $this->input->post('nik_ibu',TRUE),
						'tahun_lahir_ibu' => $this->input->post('tahun_lahir_ibu',TRUE),
						'pendidikan_ibu' => $this->input->post('pendidikan_ibu',TRUE),
                    'pekerjaan_ibu' => $this->input->post('pekerjaan_ibu',TRUE),
                    'penghasilan_bulanan_ibu' => $this->input-
>post('penghasilan_bulanan_ibu',TRUE),
                    'berkebutuhan_khusus_ibu' => $this->input-
>post('berkebutuhan_khusus_ibu',TRUE),
                    'nama_wali' => $this->input->post('nama_wali',TRUE),
                    'nik_wali' => $this->input->post('nik_wali',TRUE),
                    'tahun_lahir_wali' => $this->input-
>post('tahun_lahir_wali',TRUE),
                    'pendidikan_wali' => $this->input-
>post('pendidikan_wali',TRUE),
                    'pekerjaan_wali' => $this->input-
>post('pekerjaan_wali',TRUE),
                    'penghasilan_bulanan_wali' => $this->input-
>post('penghasilan_bulanan_wali',TRUE),
                    'no_telepon_rumah' => $this->input-
>post('no_telepon_rumah',TRUE),
                    // 'nomor_hp' => $this->input->post('nomor_hp',TRUE),
                    'email' => $this->input->post('email',TRUE),
                    'jenis_ekstrakurikuler' => $this->input-
>post('jenis_ekstrakurikuler',TRUE),
                    'tinggi_badan' => $this->input->post('tinggi_badan',TRUE),
                    'berat_badan' => $this->input->post('berat_badan',TRUE),
                    // 'jarak_ke_sekolah' => $this->input-
>post('jarak_ke_sekolah',TRUE),
                    'jumlah_saudara_kandung' => $this->input-
>post('jumlah_saudara_kandung',TRUE),
                    'jurusan' => $this->input->post('jurusan',TRUE),
                    // 'asal_sekolah' => $this->input->post('asal_sekolah',TRUE),
                    'no_peserta_ujian' => $this->input-
>post('no_peserta_ujian',TRUE),
                    'no_seri_ijazah' => $this->input->post('no_seri_ijazah',TRUE),
                    'no_seri_skhu' => $this->input->post('no_seri_skhu',TRUE),
                    'nilai_usbn' => $this->input->post('nilai_usbn',TRUE),
                    'nilai_rapor' => $this->input->post('nilai_rapor',TRUE),
                              'nilai_unbk_unkp' => $this->input-
>post('nilai_unbk_unkp',TRUE),
                              'ketentuan' => $this->input->post('ketentuan',TRUE),
                              'foto' => $this->input->post('foto',TRUE),
                              'akte_kelahiran' => $this->input->post('akte_kelahiran',TRUE),
                              'kartu_keluarga' => $this->input->post('kartu_keluarga',TRUE),
                              'skl_skhu' => $this->input->post('skl_skhu',TRUE),
                              'skd' => $this->input->post('skd',TRUE),
                              'berkaslain' => $this->input->post('berkaslain',TRUE),
                              'tipe' => $this->input->post('tipe',TRUE),
                              'kode_daring' => $this->input->post('kode_daring',TRUE),
                              'kode_luring' => $this->input->post('kode_luring',TRUE),
                              'kode_formulir' => $this->input->post('kode_formulir',TRUE),
                              'foto_full' => $this->input->post('foto_full',TRUE),
                              'rapor' => $this->input->post('rapor',TRUE),
                              'sktm' => $this->input->post('sktm',TRUE),
                              'ktp_ortu' => $this->input->post('ktp_ortu',TRUE),
                              'kartu_bantuan' => $this->input->post('kartu_bantuan',TRUE),
                              'sptjm' => $this->input->post('sptjm',TRUE),
                              'sp' => $this->input->post('sp',TRUE),

              );

          $this->Formulir_model->update($this->input->post('id_formulir', TRUE), $data);
          $this->session->set_flashdata('message', 'Data Berhasil diubah');
          helper_log("edit", "Update inputan formulir pendaftaran");
          redirect(site_url('formulir'));
        }
    }

    public function _rules()
    {
      $this->form_validation->set_rules('tahun_pelajaran', 'tahun pelajaran', 'trim');
      $this->form_validation->set_rules('jalur_pendaftaran', 'jalur pendaftaran', 'trim');
      $this->form_validation->set_rules('nama_peserta', 'nama peserta', 'trim');
      $this->form_validation->set_rules('jenis_kelamin', 'jenis kelamin', 'trim');
      $this->form_validation->set_rules('nisn', 'nisn', 'trim');
      $this->form_validation->set_rules('nik', 'nik', 'trim');
      $this->form_validation->set_rules('tempat_lahir', 'tempat lahir', 'trim');
      $this->form_validation->set_rules('tanggal_lahir', 'tanggal lahir', 'trim');
      $this->form_validation->set_rules('no_registrasi_akta_lahir', 'no registrasi akta lahir', 'trim');
      $this->form_validation->set_rules('agama', 'agama', 'trim');
      $this->form_validation->set_rules('kewarganegaraan', 'kewarganegaraan', 'trim');
      $this->form_validation->set_rules('berkebutuhan_khusus', 'berkebutuhan khusus', 'trim');
      $this->form_validation->set_rules('alamat', 'alamat', 'trim');
      $this->form_validation->set_rules('rt', 'rt', 'trim');
      $this->form_validation->set_rules('rw', 'rw', 'trim');
      $this->form_validation->set_rules('nama_dusun', 'nama dusun', 'trim');
      $this->form_validation->set_rules('nama_kelurahan', 'nama kelurahan', 'trim');
      $this->form_validation->set_rules('kecamatan', 'kecamatan', 'trim');
      $this->form_validation->set_rules('kode_pos', 'kode pos', 'trim');
      $this->form_validation->set_rules('latitude', 'latitude', 'trim');
      $this->form_validation->set_rules('longitude', 'longitude', 'trim');
      $this->form_validation->set_rules('tempat_tinggal', 'tempat tinggal', 'trim');
      $this->form_validation->set_rules('moda_transportasi', 'moda transportasi', 'trim');
      $this->form_validation->set_rules('no_kks', 'no kks', 'trim');
      $this->form_validation->set_rules('anak_ke', 'anak ke', 'trim');
      $this->form_validation->set_rules('penerima_kps_pkh', 'penerima kps pkh', 'trim');
      $this->form_validation->set_rules('no_kps_pkh', 'no kps pkh', 'trim');
      $this->form_validation->set_rules('penerima_kip', 'penerima kip', 'trim');
      $this->form_validation->set_rules('no_kip', 'no kip', 'trim');
      $this->form_validation->set_rules('nama_tertera_di_kip', 'nama tertera di kip', 'trim');
      $this->form_validation->set_rules('terima_fisik_kartu_kip', 'terima fisik kartu kip', 'trim');
```

```
$this->form_validation->set_rules('nama_ayah', 'nama ayah', 'trim');
$this->form_validation->set_rules('nik_ayah', 'nik ayah', 'trim');
$this->form_validation->set_rules('tahun_lahir_ayah', 'tahun lahir ayah', 'trim');
$this->form_validation->set_rules('pendidikan_ayah', 'pendidikan ayah', 'trim');
$this->form_validation->set_rules('pekerjaan_ayah', 'pekerjaan ayah', 'trim');
$this->form_validation->set_rules('penghasilan_bulanan_ayah', 'penghasilan bulanan ayah', 'trim');
$this->form_validation->set_rules('berkebutuhan_khusus_ayah', 'berkebutuhan khusus ayah', 'trim');
$this->form_validation->set_rules('nama_ibu', 'nama ibu', 'trim');
$this->form_validation->set_rules('nik_ibu', 'nik ibu', 'trim');
$this->form_validation->set_rules('tahun_lahir_ibu', 'tahun lahir ibu', 'trim');
$this->form_validation->set_rules('pendidikan_ibu', 'pendidikan ibu', 'trim');
$this->form_validation->set_rules('pekerjaan_ibu', 'pekerjaan ibu', 'trim');
$this->form_validation->set_rules('penghasilan_bulanan_ibu', 'penghasilan bulanan ibu', 'trim');
$this->form_validation->set_rules('berkebutuhan_khusus_ibu', 'berkebutuhan khusus ibu', 'trim');
$this->form_validation->set_rules('nama_wali', 'nama wali', 'trim');
$this->form_validation->set_rules('nik_wali', 'nik wali', 'trim');
$this->form_validation->set_rules('tahun_lahir_wali', 'tahun lahir wali', 'trim');
$this->form_validation->set_rules('pendidikan_wali', 'pendidikan wali', 'trim');
$this->form_validation->set_rules('pekerjaan_wali', 'pekerjaan wali', 'trim');
$this->form_validation->set_rules('penghasilan_bulanan_wali', 'penghasilan bulanan wali', 'trim');
$this->form_validation->set_rules('no_telepon_rumah', 'no telepon rumah', 'trim');
$this->form_validation->set_rules('nomor_hp', 'nomor hp', 'trim');
$this->form_validation->set_rules('email', 'email', 'trim');
$this->form_validation->set_rules('jenis_ekstrakurikuler', 'jenis ekstrakurikuler', 'trim');
$this->form_validation->set_rules('tinggi_badan', 'tinggi badan', 'trim');
$this->form_validation->set_rules('berat_badan', 'berat badan', 'trim');
$this->form_validation->set_rules('jarak_ke_sekolah', 'jarak ke sekolah', 'trim');
```

```
        $this->form_validation->set_rules('jumlah_saudara_kandung', 'jumlah saudara kandung', 'trim');
        $this->form_validation->set_rules('jurusan', 'jurusan', 'trim');
        $this->form_validation->set_rules('asal_sekolah', 'asal sekolah', 'trim');
        $this->form_validation->set_rules('no_peserta_ujian', 'no peserta ujian', 'trim');
        $this->form_validation->set_rules('no_seri_ijazah', 'no seri ijazah', 'trim');
        $this->form_validation->set_rules('no_seri_skhu', 'no seri skhu', 'trim');
        $this->form_validation->set_rules('nilai_usbn', 'nilai usbn', 'trim');
        $this->form_validation->set_rules('nilai_rapor', 'nilai rapor', 'trim');
        $this->form_validation->set_rules('nilai_unbk_unkp', 'nilai unbk/unkp', 'trim');
        $this->form_validation->set_rules('ketentuan', 'ketentuan', 'trim');
        $this->form_validation->set_rules('foto', 'foto', 'trim');
        $this->form_validation->set_rules('akte_kelahiran', 'akte_kelahiran', 'trim');
        $this->form_validation->set_rules('kartu_keluarga', 'kartu_keluarga', 'trim');
        $this->form_validation->set_rules('skl_skhu', 'skl_skhu', 'trim');
        $this->form_validation->set_rules('skd', 'skd', 'trim');
        $this->form_validation->set_rules('berkaslain', 'berkaslain', 'trim');
        $this->form_validation->set_rules('tipe', 'tipe', 'trim');
        $this->form_validation->set_rules('kode_daring', 'kode daring', 'trim');
        $this->form_validation->set_rules('kode_luring', 'kode luring', 'trim');
        $this->form_validation->set_rules('kode_formulir', 'kode formulir', 'trim');
        $this->form_validation->set_rules('foto_full', 'foto_full', 'trim');
        $this->form_validation->set_rules('rapor', 'rapor', 'trim');
        $this->form_validation->set_rules('sktm', 'sktm', 'trim');
        $this->form_validation->set_rules('ktp_ortu', 'ktp_ortu', 'trim');
        $this->form_validation->set_rules('kartu_bantuan', 'kartu_bantuan', 'trim');
        $this->form_validation->set_rules('sptjm', 'sptjm', 'trim');
        $this->form_validation->set_rules('sp', 'sp', 'trim');

        $this->form_validation->set_rules('id_formulir', 'id_formulir', 'trim');
        $this->form_validation->set_error_delimiters('<span class="text-danger">', '</span>');
    }
```

```
   public function cetak(){
     // $id_peserta = $id;
     // $data['berkas']=$this->Berkas_model->get_foto($id_peserta);
     // $data['prestasipeserta']=$this->Prestasipeserta_model-
>get_all_prestasi($id_peserta);
     $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();
     $data['pengumuman'] = $this->Pengumuman_model->get_by_formulir();
     $data['tp'] =  $this->Tahunpelajaran_model->get_tahun_aktif();
     $data['jalur'] =  $this->Jalur_model->get_all();

     $data['formulir'] =  $this->Formulir_model->get_by_id_1();
     helper_log("print", "Cetak Formulir Kosong ");

          $mpdf = new \Mpdf\Mpdf(['format' => 'Legal']);
          $html = $this->load->view('formulir/Print_formulir', $data,true);
          $mpdf->WriteHTML($html);
          $mpdf->Output('formulir.pdf','I');
  }
}
/* End of file Formulir.php */
```

- Groups

```
<?php
if (!defined('BASEPATH')) exit('No direct script access allowed');

class Groups extends CI_Controller
{
  function __construct()
  {
     parent::__construct();
     $this->layout->auth();
```

```
    $c_url = $this->router->fetch_class();
    $this->layout->auth_privilege($c_url);
    $this->load->model('Groups_model');
    $this->load->model('Pengaturan_model');
    $this->load->library('form_validation');
      $this->load->library('datatables');
}

public function index()
{
    $data['title'] = 'Groups';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['code_js'] = 'groups/codejs';
    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'groups/groups_list';
    $this->load->view('template/backend', $data);
}

public function json() {
    header('Content-Type: application/json');
    echo $this->Groups_model->json();
}

public function read($id)
{
    $row = $this->Groups_model->get_by_id($id);
    if ($row) {
    $data = array(
                'id' => $row->id,
                'name' => $row->name,
                'description' => $row->description,
        );

    $data['title'] = 'Groups';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'groups/groups_read';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('groups'));
    }
}

public function create()
{
    $data = array(
        'button' => 'Tambah',
        'action' => site_url('groups/create_action'),
      'id' => set_value('id'),
      'name' => set_value('name'),
      'description' => set_value('description'),
      );

    $data['title'] = 'Groups';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'groups/groups_form';
    $this->load->view('template/backend', $data);
}

public function create_action()
{
    $this->_rules();

    if ($this->form_validation->run() == FALSE) {
        $this->create();
    } else {
    $data = array(
              'name' => $this->input->post('name',TRUE),
              'description' => $this->input->post('description',TRUE),
      );

        $this->Groups_model->insert($data);
        $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
        helper_log("add", "Menambah data group");
        redirect(site_url('groups'));
    }
}

public function update($id)
{
    $row = $this->Groups_model->get_by_id($id);
    if ($row) {
    $data = array(
        'button' => 'Ubah',
        'action' => site_url('groups/update_action'),
              'id' => set_value('id', $row->id),
              'name' => set_value('name', $row->name),
              'description' => set_value('description', $row->description),
      );

    $data['title'] = 'Groups';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'groups/groups_form';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('groups'));
    }
}

public function update_action()
{
    $this->_rules();

    if ($this->form_validation->run() == FALSE) {
        $this->update($this->input->post('id', TRUE));
    } else {
    $data = array(
            'name' => $this->input->post('name',TRUE),
            'description' => $this->input->post('description',TRUE),
        );

        $this->Groups_model->update($this->input->post('id', TRUE), $data);
        $this->session->set_flashdata('message', 'Data Berhasil diubah');
        helper_log("edit", "Update data group");
        redirect(site_url('groups'));
    }
}

public function delete($id)
{
    $row = $this->Groups_model->get_by_id($id);

    if ($row->id=='1' or $row->id=='2' or $row->id=='3') {
        $this->session->set_flashdata('message', 'Data Gagal dihapus');
        redirect(site_url('groups'));
    } else {
        $this->Groups_model->delete($id);
        $this->session->set_flashdata('message', 'Data Berhasil dihapus');
        helper_log("delete", "Menghapus data group");
        redirect(site_url('groups'));
    }
}

public function deletebulk(){
    $data = $_POST['msg_'];
    $dataid = explode(',', $data);
    foreach ($dataid as $key => $value) {
       $this->Groups_model->delete($value);
       $this->session->set_flashdata('message', 'Data Berhasil dihapus');
```

```
        }
        echo true;
    }
    public function printdoc(){
        $data = array(
            'groups_data' => $this->Groups_model->get_all(),
            'start' => 0
        );
        $this->load->view('groups/groups_print', $data);
    }
    public function _rules()
    {
      $this->form_validation->set_rules('name', 'name', 'trim|required');
      $this->form_validation->set_rules('description', 'description', 'trim|required');
      $this->form_validation->set_rules('id', 'id', 'trim');
      $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');
    }

    public function excel()
    {
        $this->load->helper('exportexcel');
        $namaFile = "groups.xls";
        $judul = "groups";
        $tablehead = 0;
        $tablebody = 1;
        $nourut = 1;
        //penulisan header
        header("Pragma: public");
        header("Expires: 0");
        header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
        header("Content-Type: application/force-download");
    header("Content-Type: application/octet-stream");
    header("Content-Type: application/download");
    header("Content-Disposition: attachment;filename=" . $namaFile . "");
    header("Content-Transfer-Encoding: binary ");

    xlsBOF();

    $kolomhead = 0;
    xlsWriteLabel($tablehead, $kolomhead++, "No");
  xlsWriteLabel($tablehead, $kolomhead++, "Name");
  xlsWriteLabel($tablehead, $kolomhead++, "Description");

  foreach ($this->Groups_model->get_all() as $data) {
    $kolombody = 0;

    //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
    xlsWriteNumber($tablebody, $kolombody++, $nourut);
      xlsWriteLabel($tablebody, $kolombody++, $data->name);
      xlsWriteLabel($tablebody, $kolombody++, $data->description);

      $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}

public function word()
{
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=groups.doc");
```

```
        $data = array(
            'groups_data' => $this->Groups_model->get_all(),
            'start' => 0
        );
        $this->load->view('groups/groups_doc',$data);
    }
}
/* End of file Groups.php */
```

- Jalur dan Jarak

```
?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Jalur extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Jalur_model');
        $this->load->model('Pengaturan_model');
        $this->load->library('form_validation');
          $this->load->library('datatables');
    }

    public function index()
    {
        $data = array(
        'button' => 'Tambah',
        'action' => site_url('jalur/create_action'),
        'id_jalur' => set_value('id_jalur'),
        'jalur' => set_value('jalur'),
        'persentase' => set_value('persentase'),
        'status_jalur' => set_value('status_jalur'),
    );

    $data['title'] = 'Jalur';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Jalur' => '',
    ];

    $data['code_js'] = 'jalur/codejs';
    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jalur/Jalur_list';
    $this->load->view('template/backend', $data);
}

public function json() {
    header('Content-Type: application/json');
    echo $this->Jalur_model->json();
}

public function read($id)
{
    $row = $this->Jalur_model->get_by_id($id);
    if ($row) {
    $data = array(
            'id_jalur' => $row->id_jalur,
            'jalur' => $row->jalur,
              'persentase' => $row->persentase,
        'status_jalur' => $row->status_jalur,
      );

    $data['title'] = 'Jalur';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jalur/Jalur_read';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('jalur'));
    }
}

public function create()
{
    $data = array(
        'button' => 'Tambah',
        'action' => site_url('jalur/create_action'),
      'id_jalur' => set_value('id_jalur'),
      'jalur' => set_value('jalur'),
      'persentase' => set_value('persentase'),
        'status_jalur' => set_value('status_jalur'),
      );

    $data['title'] = 'Jalur';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jalur/Jalur_form';
    $this->load->view('template/backend', $data);
}

public function create_action()
{
    $this->_rules();

    if ($this->form_validation->run() == FALSE) {
        // $this->create();
        $this->session->set_flashdata('message',
            form_error('jalur').
            form_error('persentase').
            form_error('status_jalur')
        );
        redirect(site_url('jalur'));
    } else {
    $data = array(
              'jalur' => $this->input->post('jalur',TRUE),
              'persentase' => $this->input->post('persentase',TRUE),
        'status_jalur' => $this->input->post('status_jalur',TRUE),
      );

    $this->Jalur_model->insert($data);
    $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
    helper_log("add", "Menambah data jalur ".$data['jalur']);
    redirect(site_url('jalur'));
    }
}

public function update($id)
{
    $row = $this->Jalur_model->get_by_id($id);

    if ($row) {
    $data = array(
        'button' => 'Ubah',
        'action' => site_url('jalur/update_action'),
              'id_jalur' => set_value('id_jalur', $row->id_jalur),
              'jalur' => set_value('jalur', $row->jalur),
              'persentase' => set_value('persentase', $row->persentase),
        'status_jalur' => set_value('persentase', $row->status_jalur),
      );

    $data['title'] = 'Jalur';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jalur/Jalur_form';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('jalur'));
    }
}
```

```
    public function update_action()
    {
        // $this->_rules();
        $this->form_validation->set_rules('jalur', 'jalur', 'trim|required',
            array(
                    'required'      => 'Jalur Pendaftaran tidak boleh kosong '
            ));
        $this->form_validation->set_rules('persentase', 'persentase', 'trim|required|numeric',
            array(
                    'required'      => 'Persentase tidak boleh kosong ',
                    'numeric'       => 'Persentase hanya angka '
            ));
        $this->form_validation->set_rules('status_jalur', 'status jalur', 'trim|required',
            array(
                    'required'      => 'Status Jalur tidak boleh kosong '
            ));

        $this->form_validation->set_rules('id_jalur', 'id_jalur', 'trim');
        $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');

        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_jalur', TRUE));
        } else {
        $data = array(
                  'jalur' => $this->input->post('jalur',TRUE),
                  'persentase' => $this->input->post('persentase',TRUE),
            'status_jalur' => $this->input->post('status_jalur',TRUE),
          );

        $this->Jalur_model->update($this->input->post('id_jalur', TRUE), $data);
        $this->session->set_flashdata('message', 'Data Berhasil diubah');
        helper_log("edit", "Update data jalur ".$data['jalur']);
        redirect(site_url('jalur'));
        }
    }

    public function delete($id)
    {
        $row = $this->Jalur_model->get_by_id($id);

        if ($row->id_jalur=='1' or $row->id_jalur=='2' or $row->id_jalur=='3' or $row->id_jalur=='4') {
            $this->session->set_flashdata('message', 'Data Gagal dihapus');
            redirect(site_url('jalur'));
        } else {
            $this->Jalur_model->delete($id);
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus data jalur ".$row->jalur);
            redirect(site_url('jalur'));
        }
    }

    public function deletebulk(){
        $delete = $this->Jalur_model->deletebulk();
        if($delete){
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus multi data jalur pendaftaran");
        }else{
            $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
        }
        echo $delete;
    }
```

```
public function _rules()
{
  $this->form_validation->set_rules('jalur', 'jalur', 'trim|required|is_unique[jalur.jalur]',
    array(
            'required'      => 'Jalur Pendaftaran tidak boleh kosong ',
            'is_unique'     => 'Jalur Pendaftaran sudah ada '
    ));
  $this->form_validation->set_rules('persentase', 'persentase', 'trim|required|numeric',
    array(
            'required'      => 'Persentase tidak boleh kosong ',
            'numeric'       => 'Persentase hanya angka '
    ));
$this->form_validation->set_rules('status_jalur', 'status jalur', 'trim|required',
    array(
            'required'      => 'Status Jalur tidak boleh kosong '
    ));

  $this->form_validation->set_rules('id_jalur', 'id_jalur', 'trim');
  $this->form_validation->set_error_delimiters('<span class="text">', '</span>');
}

public function excel()
{
    $this->load->helper('exportexcel');
    $namaFile = "jalur.xls";
    $judul = "jalur";
    $tablehead = 0;
    $tablebody = 1;
    $nourut = 1;
    //penulisan header
    header("Pragma: public");
    header("Expires: 0");
    header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
    header("Content-Type: application/force-download");
    header("Content-Type: application/octet-stream");
    header("Content-Type: application/download");
    header("Content-Disposition: attachment;filename=" . $namaFile . "");
    header("Content-Transfer-Encoding: binary ");

    xlsBOF();

    $kolomhead = 0;
    xlsWriteLabel($tablehead, $kolomhead++, "No");
  xlsWriteLabel($tablehead, $kolomhead++, "Jalur");
  xlsWriteLabel($tablehead, $kolomhead++, "Persentase");
    xlsWriteLabel($tablehead, $kolomhead++, "Status");

  foreach ($this->Jalur_model->get_all() as $data) {
    $kolombody = 0;

    //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
    xlsWriteNumber($tablebody, $kolombody++, $nourut);
     xlsWriteLabel($tablebody, $kolombody++, $data->jalur);
     xlsWriteNumber($tablebody, $kolombody++, $data->persentase);
    xlsWriteLabel($tablebody, $kolombody++, $data->status_jalur);

     $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}
```

```
    public function word()
    {
        header("Content-type: application/vnd.ms-word");
        header("Content-Disposition: attachment;Filename=jalur.doc");

        $data = array(
            'jalur_data' => $this->Jalur_model->get_all(),
            'start' => 0
        );
        $this->load->view('jalur/Jalur_doc',$data);
    }

    public function printdoc(){
        $data = array(
            'jalur_data' => $this->Jalur_model->get_all(),
            'start' => 0
        );
        $this->load->view('jalur/Jalur_print', $data);
    }
}
/* End of file Jalur.php */
<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Jarak extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
    $this->layout->auth();
    $this->layout->auth_privilege($c_url);
    $this->load->model('Jarak_model');
    $this->load->model('Pengaturan_model');
    $this->load->library('form_validation');
      $this->load->library('datatables');
}

public function index()
{
    $data = array(
        'button' => 'Tambah',
        'action' => site_url('jarak/create_action'),
        'id_jarak' => set_value('id_jarak'),
        'jarak' => set_value('jarak'),
        'skor_jarak' => set_value('skor_jarak'),
    );

    $data['title'] = 'Jarak';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Jarak' => '',
    ];

    $data['code_js'] = 'jarak/codejs';
    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jarak/Jarak_list';
    $this->load->view('template/backend', $data);
}

public function json() {
    header('Content-Type: application/json');
    echo $this->Jarak_model->json();
}

public function read($id)
{
    $row = $this->Jarak_model->get_by_id($id);
    if ($row) {
        $data = array(
                          'id_jarak' => $row->id_jarak,
                          'jarak' => $row->jarak,
                          'skor_jarak' => $row->skor_jarak,
         );

    $data['title'] = 'Jarak';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jarak/Jarak_read';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('jarak'));
    }
}

public function create()
{
    $data = array(
        'button' => 'Tambah',
        'action' => site_url('jarak/create_action'),
      'id_jarak' => set_value('id_jarak'),
      'jarak' => set_value('jarak'),
      'skor_jarak' => set_value('skor_jarak'),
  );

    $data['title'] = 'Jarak';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jarak/Jarak_form';
    $this->load->view('template/backend', $data);
}

public function create_action()
{
    $this->_rules();

    if ($this->form_validation->run() == FALSE) {
        // $this->create();
        $this->session->set_flashdata('message',
            form_error('jarak').
            form_error('skor_jarak')
        );
        redirect(site_url('jarak'));
    } else {
        $data = array(
                          'jarak' => $this->input->post('jarak',TRUE),
                          'skor_jarak' => $this->input->post('skor_jarak',TRUE),
          );

        $this->Jarak_model->insert($data);
        $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
        helper_log("add", "Menambah data jarak ".$data['jarak']);
        redirect(site_url('jarak'));
    }
}

public function update($id)
{
    $row = $this->Jarak_model->get_by_id($id);

    if ($row) {
        $data = array(
            'button' => 'Ubah',
            'action' => site_url('jarak/update_action'),
                          'id_jarak' => set_value('id_jarak', $row->id_jarak),
                          'jarak' => set_value('jarak', $row->jarak),
                          'skor_jarak' => set_value('skor_jarak', $row->skor_jarak),
         );

    $data['title'] = 'Jarak';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jarak/Jarak_form';
    $this->load->view('template/backend', $data);
    } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('jarak'));
        }
    }

    public function update_action()
    {
        // $this->_rules();
        $this->form_validation->set_rules('skor_jarak', 'skor jarak', 'trim|required|numeric',
            array(
                    'required'      => 'Poin Jarak tidak boleh kosong ',
                    'numeric'       => 'Poin Jarak hanya angka '
            ));

        $this->form_validation->set_rules('id_jarak', 'id_jarak', 'trim');
        $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');

        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_jarak', TRUE));
        } else {
            $data = array(
                  'jarak' => $this->input->post('jarak',TRUE),
                  'skor_jarak' => $this->input->post('skor_jarak',TRUE),
          );

            $this->Jarak_model->update($this->input->post('id_jarak', TRUE), $data);
            $this->session->set_flashdata('message', 'Data Berhasil diubah');
            helper_log("edit", "Update data poin jarak ".$data['jarak']);
            redirect(site_url('jarak'));
        }
    }
```

```
public function delete($id)
{
    $row = $this->Jarak_model->get_by_id($id);

    if ($row) {
        $this->Jarak_model->delete($id);
        $this->session->set_flashdata('message', 'Data Berhasil dihapus');
        helper_log("delete", "Menghapus data jarak ".$row->jarak);
        redirect(site_url('jarak'));
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('jarak'));
    }
}

public function deletebulk(){
    $delete = $this->Jarak_model->deletebulk();
    if($delete){
        $this->session->set_flashdata('message', 'Data Berhasil dihapus');
        helper_log("delete", "Menghapus multi data poin jarak");
    }else{
        $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
    }
    echo $delete;
}

public function _rules()
{
  $this->form_validation->set_rules('jarak', 'jarak', 'trim|required|is_unique[jarak.jarak]',
    array(
            'required'      => 'Jarak ke sekolah tidak boleh kosong ',
            'is_unique'     => 'Jarak ke sekolah sudah ada '
    ));
  $this->form_validation->set_rules('skor_jarak', 'skor jarak', 'trim|required|numeric',
    array(
            'required'      => 'Poin Jarak tidak boleh kosong ',
            'numeric'       => 'Poin Jarak hanya angka '
    ));

  $this->form_validation->set_rules('id_jarak', 'id_jarak', 'trim');
  $this->form_validation->set_error_delimiters('<span class="text">', '</span>');
}

public function excel()
{
    $this->load->helper('exportexcel');
    $namaFile = "jarak.xls";
    $judul = "jarak";
    $tablehead = 0;
    $tablebody = 1;
    $nourut = 1;
    //penulisan header
    header("Pragma: public");
    header("Expires: 0");
    header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
    header("Content-Type: application/force-download");
    header("Content-Type: application/octet-stream");
    header("Content-Type: application/download");
    header("Content-Disposition: attachment;filename=" . $namaFile . "");
    header("Content-Transfer-Encoding: binary ");

    xlsBOF();
    $kolomhead = 0;
    xlsWriteLabel($tablehead, $kolomhead++, "No");
  xlsWriteLabel($tablehead, $kolomhead++, "Jarak");
  xlsWriteLabel($tablehead, $kolomhead++, "Skor Jarak");

  foreach ($this->Jarak_model->get_all() as $data) {
    $kolombody = 0;

    //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
    xlsWriteNumber($tablebody, $kolombody++, $nourut);
      xlsWriteLabel($tablebody, $kolombody++, $data->jarak);
      xlsWriteNumber($tablebody, $kolombody++, $data->skor_jarak);

      $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}

public function word()
{
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=jarak.doc");

    $data = array(
        'jarak_data' => $this->Jarak_model->get_all(),
        'start' => 0
    );
    $this->load->view('jarak/Jarak_doc',$data);
}
```

```
    public function printdoc(){
        $data = array(
            'jarak_data' => $this->Jarak_model->get_all(),
            'start' => 0
        );
        $this->load->view('jarak/Jarak_print', $data);
    }
}
/* End of file Jarak.php */
```

- Jurusan

```
<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Jurusan extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Jurusan_model');
        $this->load->model('Pengaturan_model');
        $this->load->library('form_validation');
          $this->load->library('datatables');
    }

    public function index()
    {
        $data = array(
            'button' => 'Tambah',
            'action' => site_url('jurusan/create_action'),
            'id_jurusan' => set_value('id_jurusan'),
            'bidang_keahlian' => set_value('bidang_keahlian'),
            'nama_jurusan' => set_value('nama_jurusan'),
            'kuota_jurusan' => set_value('kuota_jurusan'),
        );

        $data['title'] = 'Jurusan';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Jurusan' => '',
        ];

        $data['code_js'] = 'jurusan/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'jurusan/Jurusan_list';
        $this->load->view('template/backend', $data);
    }

    public function json() {
        header('Content-Type: application/json');
        echo $this->Jurusan_model->json();
    }

    public function read($id)
    {
        $row = $this->Jurusan_model->get_by_id($id);
        if ($row) {
        $data = array(
                'id_jurusan' => $row->id_jurusan,
            'bidang_keahlian' => $row->bidang_keahlian,
       'nama_jurusan' => $row->nama_jurusan,
    'kuota_jurusan' => $row->kuota_jurusan,
 );

  $data['title'] = 'Jurusan';
  $data['subtitle'] = '';
  $data['crumb'] = [
    'Dashboard' => '',
  ];

  $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
  $data['page'] = 'jurusan/Jurusan_read';
  $this->load->view('template/backend', $data);
  } else {
    $this->session->set_flashdata('message', 'Data tidak ditemukan');
    redirect(site_url('jurusan'));
  }
}

public function create()
{
  $data = array(
    'button' => 'Tambah',
    'action' => site_url('jurusan/create_action'),
   'id_jurusan' => set_value('id_jurusan'),
    'bidang_keahlian' => set_value('bidang_keahlian'),
   'nama_jurusan' => set_value('nama_jurusan'),
    'kuota_jurusan' => set_value('kuota_jurusan'),
 );

  $data['title'] = 'Jurusan';
  $data['subtitle'] = '';
```

```
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'jurusan/Jurusan_form';
    $this->load->view('template/backend', $data);
}

public function create_action()
{
    $this->_rules();

    if ($this->form_validation->run() == FALSE) {
        // $this->create();
        $this->session->set_flashdata('message',
            form_error('bidang_keahlian').
            form_error('nama_jurusan').
            form_error('kuota_jurusan')
        );
        redirect(site_url('jurusan'));
    } else {
        $data = array(
                          'bidang_keahlian' => $this->input->post('bidang_keahlian',TRUE),
            'nama_jurusan' => $this->input->post('nama_jurusan',TRUE),
            'kuota_jurusan' => $this->input->post('kuota_jurusan',TRUE),
      );

        $this->Jurusan_model->insert($data);
        $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
        helper_log("add", "Menambah data jurusan ".$data['nama_jurusan']);
        redirect(site_url('jurusan'));}
```

```
}

public function update($id)
{
   $row = $this->Jurusan_model->get_by_id($id);

   if ($row) {
   $data = array(
      'button' => 'Ubah',
      'action' => site_url('jurusan/update_action'),
           'id_jurusan' => set_value('id_jurusan', $row->id_jurusan),
           'bidang_keahlian' => set_value('bidang_keahlian', $row->bidang_keahlian),
      'nama_jurusan' => set_value('nama_jurusan', $row->nama_jurusan),
      'kuota_jurusan' => set_value('kuota_jurusan', $row->kuota_jurusan),
 );

   $data['title'] = 'Jurusan';
   $data['subtitle'] = '';
   $data['crumb'] = [
      'Dashboard' => '',
   ];

   $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
   $data['page'] = 'jurusan/Jurusan_form';
   $this->load->view('template/backend', $data);
   } else {
      $this->session->set_flashdata('message', 'Data tidak ditemukan');
      redirect(site_url('jurusan'));
   }
}

public function update_action()
```

```
{
    $this->_rules();

    if ($this->form_validation->run() == FALSE) {
        $this->update($this->input->post('id_jurusan', TRUE));
    } else {
    $data = array(
                  'bidang_keahlian' => $this->input->post('bidang_keahlian',TRUE),
        'nama_jurusan' => $this->input->post('nama_jurusan',TRUE),
        'kuota_jurusan' => $this->input->post('kuota_jurusan',TRUE),
      );
        $this->Jurusan_model->update($this->input->post('id_jurusan', TRUE), $data);
        $this->session->set_flashdata('message', 'Data Berhasil diubah');
        helper_log("edit", "Update data jurusan ".$data['nama_jurusan']);
        redirect(site_url('jurusan'));
    }
}

public function delete($id)
{
    $row = $this->Jurusan_model->get_by_id($id);

    if ($row->id_jurusan=='1') {
        $this->session->set_flashdata('message', 'Data Gagal dihapus');
        redirect(site_url('jurusan'));
    } else {
        $this->Jurusan_model->delete($id);
        $this->session->set_flashdata('message', 'Data Berhasil dihapus');
        helper_log("delete", "Menghapus data jurusan ".$row->nama_jurusan);
        redirect(site_url('jurusan'));
    }
}
```

```
    public function deletebulk(){
        $delete = $this->Jurusan_model->deletebulk();
        if($delete){
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus multi data jurusan");
        }else{
            $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
        }
        echo $delete;
    }

    public function _rules()
    {
    $this->form_validation->set_rules('bidang_keahlian', 'bidang keahlian', 'trim|required',
        array(
                'required'      => 'Bidang Keahlian tidak boleh kosong '
        ));
      $this->form_validation->set_rules('nama_jurusan', 'nama jurusan', 'trim|required',
        array(
                'required'      => 'Nama Jurusan tidak boleh kosong '
        ));
    $this->form_validation->set_rules('kuota_jurusan', 'kuota jurusan',
'trim|required|numeric',
        array(
                'required'      => 'Kuota Jurusan tidak boleh kosong ',
                'numeric'     => 'Kuota hanya angka '
        ));

      $this->form_validation->set_rules('id_jurusan', 'id_jurusan', 'trim');
      $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');
    }
```

```
public function excel()
{
    $this->load->helper('exportexcel');
    $namaFile = "jurusan.xls";
    $judul = "jurusan";
    $tablehead = 0;
    $tablebody = 1;
    $nourut = 1;
    //penulisan header
    header("Pragma: public");
    header("Expires: 0");
    header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
    header("Content-Type: application/force-download");
    header("Content-Type: application/octet-stream");
    header("Content-Type: application/download");
    header("Content-Disposition: attachment;filename=" . $namaFile . "");
    header("Content-Transfer-Encoding: binary ");

    xlsBOF();

    $kolomhead = 0;
    xlsWriteLabel($tablehead, $kolomhead++, "No");
      xlsWriteLabel($tablehead, $kolomhead++, "Bidang");
    xlsWriteLabel($tablehead, $kolomhead++, "Nama Jurusan");
    xlsWriteLabel($tablehead, $kolomhead++, "Kuota Jurusan");

  foreach ($this->Jurusan_model->get_all() as $data) {
    $kolombody = 0;

    //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
    xlsWriteNumber($tablebody, $kolombody++, $nourut);
    xlsWriteLabel($tablebody, $kolombody++, $data->bidang_keahlian);
      xlsWriteLabel($tablebody, $kolombody++, $data->nama_jurusan);
    xlsWriteNumber($tablebody, $kolombody++, $data->kuota_jurusan);

      $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}

public function word()
{
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=jurusan.doc");

    $data = array(
        'jurusan_data' => $this->Jurusan_model->get_all(),
        'start' => 0
    );
    $this->load->view('jurusan/Jurusan_doc',$data);
}

public function printdoc(){
    $data = array(
        'jurusan_data' => $this->Jurusan_model->get_all(),
        'start' => 0
    );
    $this->load->view('jurusan/Jurusan_print', $data);
}
```

```
}
/* End of file Jurusan.php */
```

- Pengumuman

```
<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Pengumuman extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Pengumuman_model');
        $this->load->model('Pengaturan_model');
        $this->load->library('form_validation');
          $this->load->library('datatables');
    }

    public function index()
    {
        $data['title'] = 'Pengumuman';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Pengumuman' => '',
        ];

        $data['code_js'] = 'pengumuman/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'pengumuman/Pengumuman_list';
    $this->load->view('template/backend', $data);
}

public function json() {
    header('Content-Type: application/json');
    echo $this->Pengumuman_model->json();
}

public function read($id)
{
    $row = $this->Pengumuman_model->get_by_id($id);
    if ($row) {
        $data = array(
              'id_pengumuman' => $row->id_pengumuman,
              'type' => $row->type,
              'judul' => $row->judul,
              'text' => $row->text,
              'date' => $row->date,
      );

    $data['title'] = 'Pengumuman';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'pengumuman/Pengumuman_read';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('pengumuman'));
    }
}

public function create()
{
    $data = array(
        'button' => 'Tambah',
        'action' => site_url('pengumuman/create_action'),
      'id_pengumuman' => set_value('id_pengumuman'),
      'type' => set_value('type'),
      'judul' => set_value('judul'),
      'text' => set_value('text'),
      'date' => set_value('date'),
  );

    $data['title'] = 'Pengumuman';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['code_js'] = 'pengumuman/codejs';
    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'pengumuman/Pengumuman_form';
    $this->load->view('template/backend', $data);
}

public function create_action()
{
    $this->_rules();
```

```
        if ($this->form_validation->run() == FALSE) {
            $this->create();
        } else {
            $data = array(
                             'type' => $this->input->post('type',TRUE),
                             'judul' => $this->input->post('judul',TRUE),
                             'text' => $this->input->post('text',TRUE),
                             // 'date' => date('Y-m-d H:i:s'),
          );
            $this->Pengumuman_model->insert($data);
            $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
            helper_log("add", "Menambah data pengumuman ".$data['type']);
            redirect(site_url('pengumuman'));}
    }

    public function update($id)
    {
        $row = $this->Pengumuman_model->get_by_id($id);

        if ($row) {
            $data = array(
                'button' => 'Ubah',
                'action' => site_url('pengumuman/update_action'),
                             'id_pengumuman' => set_value('id_pengumuman', $row-
>id_pengumuman),
                             'type' => set_value('type', $row->type),
                             'judul' => set_value('judul', $row->judul),
                             'text' => set_value('text', $row->text),
                             'date' => set_value('date', $row->date),
          );

        $data['title'] = 'Pengumuman';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];

        $data['code_js'] = 'pengumuman/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'pengumuman/Pengumuman_form';
        $this->load->view('template/backend', $data);
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('pengumuman'));
        }
    }

    public function update_action()
    {
        $this->_rules();

        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_pengumuman', TRUE));
        } else {
            $data = array(
                              'type' => $this->input->post('type',TRUE),
                              'judul' => $this->input->post('judul',TRUE),
                              'text' => $this->input->post('text',TRUE),
                              // 'date' => date('Y-m-d H:i:s'),
          );
            $this->Pengumuman_model->update($this->input->post('id_pengumuman',
TRUE), $data);
            $this->session->set_flashdata('message', 'Data Berhasil diubah');
            helper_log("edit", "Update data pengumuman ".$data['type']);
            redirect(site_url('pengumuman'));
        }
    }

    public function delete($id)
    {
        $row = $this->Pengumuman_model->get_by_id($id);

        if ($row) {
            $this->Pengumuman_model->delete($id);
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus data pengumuman ".$row->type);
            redirect(site_url('pengumuman'));
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('pengumuman'));
        }
    }

    public function deletebulk(){
        $delete = $this->Pengumuman_model->deletebulk();
        if($delete){
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus multi data pengumuman");
        }else{
            $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
        }
        echo $delete;
    }

    public function _rules()
    {
        $this->form_validation->set_rules('type', 'type', 'trim|required');
        $this->form_validation->set_rules('judul', 'judul', 'trim|required');
        $this->form_validation->set_rules('text', 'text', 'trim|required');
        $this->form_validation->set_rules('date', 'date', 'trim');

        $this->form_validation->set_rules('id_pengumuman', 'id_pengumuman', 'trim');
        $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');
    }

    public function excel()
    {
        $this->load->helper('exportexcel');
        $namaFile = "pengumuman.xls";
        $judul = "pengumuman";
        $tablehead = 0;
        $tablebody = 1;
        $nourut = 1;
        //penulisan header
        header("Pragma: public");
        header("Expires: 0");
        header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
        header("Content-Type: application/force-download");
        header("Content-Type: application/octet-stream");
        header("Content-Type: application/download");
        header("Content-Disposition: attachment;filename=" . $namaFile . "");
        header("Content-Transfer-Encoding: binary ");

        xlsBOF();

        $kolomhead = 0;
        xlsWriteLabel($tablehead, $kolomhead++, "No");
```

```
  xlsWriteLabel($tablehead, $kolomhead++, "Type");
  xlsWriteLabel($tablehead, $kolomhead++, "Judul");
  // xlsWriteLabel($tablehead, $kolomhead++, "Informasi Pengumuman");
  xlsWriteLabel($tablehead, $kolomhead++, "Date");

  foreach ($this->Pengumuman_model->get_all() as $data) {
    $kolombody = 0;

    //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
    xlsWriteNumber($tablebody, $kolombody++, $nourut);
      xlsWriteLabel($tablebody, $kolombody++, $data->type);
      xlsWriteLabel($tablebody, $kolombody++, $data->judul);
      // xlsWriteLabel($tablebody, $kolombody++, $data->text);
      xlsWriteLabel($tablebody, $kolombody++, $data->date);

      $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}

public function word()
{
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=pengumuman.doc");

    $data = array(
        'pengumuman_data' => $this->Pengumuman_model->get_all(),
        'start' => 0
    );
```

```
        $this->load->view('pengumuman/Pengumuman_doc',$data);
    }

    public function printdoc(){
        $data = array(
            'pengumuman_data' => $this->Pengumuman_model->get_all(),
            'start' => 0
        );
        $this->load->view('pengumuman/Pengumuman_print', $data);
    }
}
/* End of file Pengumuman.php */
```

- Peserta

```
<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Peserta extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Pengaturan_model');
        $this->load->model('Pengumuman_model');
        $this->load->model('Peserta_model');
        $this->load->model('Prestasipeserta_model');
        $this->load->model('Sekolah_model');
    $this->load->model('Jalur_model');
    $this->load->model('Jarak_model');
    $this->load->model('Jurusan_model');
    $this->load->model('Tahunpelajaran_model');
    $this->load->model('Formulir_model');
    $this->load->model('Berkas_model');
    $this->load->library('form_validation');
      $this->load->library('datatables');
}

public function index()
{
      $data['title'] = 'Peserta';
      $data['subtitle'] = '';
      $data['crumb'] = [
              'Peserta' => '',
              ];

      $data['code_js'] = 'peserta/codejs';
    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
      $data['page'] = 'peserta/Peserta_list';
      $this->load->view('template/backend', $data);
}

public function json() {
    header('Content-Type: application/json');
    echo $this->Peserta_model->json();
}

public function read($id)
{
      $row = $this->Peserta_model->get_by_id($id);
```

```
if ($row) {
    $data = array(
                                'id_peserta' => $row->id_peserta,
                                'no_pendaftaran' => $row->no_pendaftaran,
                                'tanggal_daftar' => $row->tanggal_daftar,
                                'tahun_pelajaran' => $row->tahun_pelajaran,
                                'jalur' => $row->jalur,
                                'nama_peserta' => $row->nama_peserta,
                                'jenis_kelamin' => $row->jenis_kelamin,
                                'nisn' => $row->nisn,
                                'nik' => $row->nik,
                                'tempat_lahir' => $row->tempat_lahir,
                                'tanggal_lahir' => $row->tanggal_lahir,
                                'no_registrasi_akta_lahir' => $row->no_registrasi_akta_lahir,
                                'agama' => $row->agama,
                                'kewarganegaraan' => $row->kewarganegaraan,
                                'berkebutuhan_khusus' => $row->berkebutuhan_khusus,
                                'alamat' => $row->alamat,
                                'rt' => $row->rt,
                                'rw' => $row->rw,
                                'nama_dusun' => $row->nama_dusun,
                                'nama_kelurahan' => $row->nama_kelurahan,
                                'kecamatan' => $row->kecamatan,
                                'kode_pos' => $row->kode_pos,
                                'latitude' => $row->latitude,
                                'longitude' => $row->longitude,
                                'tempat_tinggal' => $row->tempat_tinggal,
                                'moda_transportasi' => $row->moda_transportasi,
                                'no_kks' => $row->no_kks,
                                'anak_ke' => $row->anak_ke,
                                'penerima_kps_pkh' => $row->penerima_kps_pkh,
                                'no_kps_pkh' => $row->no_kps_pkh,
                                'penerima_kip' => $row->penerima_kip,
                                'no_kip' => $row->no_kip,
                                'nama_tertera_di_kip' => $row->nama_tertera_di_kip,
                                'terima_fisik_kartu_kip' => $row->terima_fisik_kartu_kip,
                                'nama_ayah' => $row->nama_ayah,
                                'nik_ayah' => $row->nik_ayah,
                                'tahun_lahir_ayah' => $row->tahun_lahir_ayah,
                                'pendidikan_ayah' => $row->pendidikan_ayah,
                                'pekerjaan_ayah' => $row->pekerjaan_ayah,
                                'penghasilan_bulanan_ayah' => $row->penghasilan_bulanan_ayah,
                                'berkebutuhan_khusus_ayah' => $row->berkebutuhan_khusus_ayah,
                                'nama_ibu' => $row->nama_ibu,
                                'nik_ibu' => $row->nik_ibu,
                                'tahun_lahir_ibu' => $row->tahun_lahir_ibu,
                                'pendidikan_ibu' => $row->pendidikan_ibu,
                                'pekerjaan_ibu' => $row->pekerjaan_ibu,
                                'penghasilan_bulanan_ibu' => $row->penghasilan_bulanan_ibu,
                                'berkebutuhan_khusus_ibu' => $row->berkebutuhan_khusus_ibu,
                                'nama_wali' => $row->nama_wali,
                                'nik_wali' => $row->nik_wali,
                                'tahun_lahir_wali' => $row->tahun_lahir_wali,
                                'pendidikan_wali' => $row->pendidikan_wali,
                                'pekerjaan_wali' => $row->pekerjaan_wali,
                                'penghasilan_bulanan_wali' => $row->penghasilan_bulanan_wali,
                                'no_telepon_rumah' => $row->no_telepon_rumah,
                                'nomor_hp' => $row->nomor_hp,
                                'email' => $row->email,
                                'jenis_ekstrakurikuler' => $row->jenis_ekstrakurikuler,
                                'tinggi_badan' => $row->tinggi_badan,
                                'berat_badan' => $row->berat_badan,
                                'jarak' => $row->jarak,
                                'jumlah_saudara_kandung' => $row-
>jumlah_saudara_kandung,
                                'nama_jurusan' => $row->nama_jurusan,
                                'pilihan_dua' => $row->pilihan_dua,
                                'asal_sekolah' => $row->asal_sekolah,
                                'no_un' => $row->no_un,
                                'no_seri_ijazah' => $row->no_seri_ijazah,
                                'no_seri_skhu' => $row->no_seri_skhu,
                                'nilai_rapor' => $row->nilai_rapor,
                                'nilai_usbn' => $row->nilai_usbn,
                                'nilai_unbk_unkp' => $row->nilai_unbk_unkp,
                                // 'skor_jarak' => $row->skor_jarak,
                                'status' => $row->status,
                                'status_hasil' => $row->status_hasil,
                                'status_daftar_ulang' => $row->status_daftar_ulang,
                                'id_users' => $row->id_users,
                                'pilihan_sekolah_lain' => $row->pilihan_sekolah_lain,
                                'catatan' => $row->catatan,
                        );

     $data['title'] = 'Peserta';
     $data['subtitle'] = '';
     $data['crumb'] = [
          'Dashboard' => '',
       ];

     $id=$row->id_peserta;
    $data['berkas_limit']=$this->Berkas_model->get_berkas($id);
    $data['berkas']=$this->Berkas_model->get_id($id);
    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
```

```
        $data['page'] = 'peserta/Peserta_read';
        $this->load->view('template/backend', $data);
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('peserta'));
        }
    }

    public function create()
    {
        $data = array(
            'button' => 'Tambah',
            'action' => site_url('peserta/create_action'),
                            'id_peserta' => set_value('id_peserta'),
                            'no_pendaftaran' => set_value('no_pendaftaran'),
                            'tanggal_daftar' => set_value('tanggal_daftar'),
                            'id_tahun' => set_value('id_tahun'),
                            'id_jalur' => set_value('id_jalur'),
                            'nama_peserta' => set_value('nama_peserta'),
                            'jenis_kelamin' => set_value('jenis_kelamin'),
                            'nisn' => set_value('nisn'),
                            'nik' => set_value('nik'),
                            'tempat_lahir' => set_value('tempat_lahir'),
                            'tanggal_lahir' => set_value('tanggal_lahir'),
                            'no_registrasi_akta_lahir' => set_value('no_registrasi_akta_lahir'),
                            'agama' => set_value('agama'),
                            'kewarganegaraan' => set_value('kewarganegaraan'),
                            'berkebutuhan_khusus' => set_value('berkebutuhan_khusus'),
                            'alamat' => set_value('alamat'),
                            'rt' => set_value('rt'),
                            'rw' => set_value('rw'),
                            'nama_dusun' => set_value('nama_dusun'),
                    'nama_kelurahan' => set_value('nama_kelurahan'),
                    'kecamatan' => set_value('kecamatan'),
                    'kode_pos' => set_value('kode_pos'),
                    'latitude' => set_value('latitude'),
                    'longitude' => set_value('longitude'),
                    'tempat_tinggal' => set_value('tempat_tinggal'),
                    'moda_transportasi' => set_value('moda_transportasi'),
                    'no_kks' => set_value('no_kks'),
                    'anak_ke' => set_value('anak_ke'),
                    'penerima_kps_pkh' => set_value('penerima_kps_pkh'),
                    'no_kps_pkh' => set_value('no_kps_pkh'),
                    'penerima_kip' => set_value('penerima_kip'),
                    'no_kip' => set_value('no_kip'),
                    'nama_tertera_di_kip' => set_value('nama_tertera_di_kip'),
                    'terima_fisik_kartu_kip' => set_value('terima_fisik_kartu_kip'),
                    'nama_ayah' => set_value('nama_ayah'),
                    'nik_ayah' => set_value('nik_ayah'),
                    'tahun_lahir_ayah' => set_value('tahun_lahir_ayah'),
                    'pendidikan_ayah' => set_value('pendidikan_ayah'),
                    'pekerjaan_ayah' => set_value('pekerjaan_ayah'),
                    'penghasilan_bulanan_ayah' =>
set_value('penghasilan_bulanan_ayah'),
                    'berkebutuhan_khusus_ayah' =>
set_value('berkebutuhan_khusus_ayah'),
                    'nama_ibu' => set_value('nama_ibu'),
                    'nik_ibu' => set_value('nik_ibu'),
                    'tahun_lahir_ibu' => set_value('tahun_lahir_ibu'),
                    'pendidikan_ibu' => set_value('pendidikan_ibu'),
                    'pekerjaan_ibu' => set_value('pekerjaan_ibu'),
                    'penghasilan_bulanan_ibu' => set_value('penghasilan_bulanan_ibu'),
                    'berkebutuhan_khusus_ibu' => set_value('berkebutuhan_khusus_ibu'),
                    'nama_wali' => set_value('nama_wali'),
                    'nik_wali' => set_value('nik_wali'),
                    'tahun_lahir_wali' => set_value('tahun_lahir_wali'),
                    'pendidikan_wali' => set_value('pendidikan_wali'),
                    'pekerjaan_wali' => set_value('pekerjaan_wali'),
                    'penghasilan_bulanan_wali' => set_value('penghasilan_bulanan_wali'),
                    'no_telepon_rumah' => set_value('no_telepon_rumah'),
                    'nomor_hp' => set_value('nomor_hp'),
                    'email' => set_value('email'),
                    'jenis_ekstrakurikuler' => set_value('jenis_ekstrakurikuler'),
                    'tinggi_badan' => set_value('tinggi_badan'),
                    'berat_badan' => set_value('berat_badan'),
                    'id_jarak' => set_value('id_jarak'),
                    'jumlah_saudara_kandung' => set_value('jumlah_saudara_kandung'),
                    'id_jurusan' => set_value('id_jurusan'),
                    'pilihan_dua' => set_value('pilihan_dua'),
                    'id_sekolah' => set_value('id_sekolah'),
                    'no_un' => set_value('no_un'),
                    'no_seri_ijazah' => set_value('no_seri_ijazah'),
                    'no_seri_skhu' => set_value('no_seri_skhu'),
                    'nilai_rapor' => set_value('nilai_rapor'),
                    'nilai_usbn' => set_value('nilai_usbn'),
                    'nilai_unbk_unkp' => set_value('nilai_unbk_unkp'),
                    'status' => set_value('status'),
                    'status_hasil' => set_value('status_hasil'),
                    'status_daftar_ulang' => set_value('status_daftar_ulang'),
                    'id_users' => set_value('id_users'),
                    'pilihan_sekolah_lain' => set_value('pilihan_sekolah_lain'),
        );

$data['title'] = 'Peserta';
$data['subtitle'] = '';
$data['crumb'] = [
    'Dashboard' => '',
        ];


        $data['code_js'] = 'peserta/codejs';
        $data['sekolah'] =$this->Sekolah_model->get_all();
        $data['jalur'] =$this->Jalur_model->get_all();
        $data['jarak'] =$this->Jarak_model->get_all();
        $data['jurusan'] =$this->Jurusan_model->get_all();
        $data['tahunpelajaran'] =$this->Tahunpelajaran_model->get_tahun_aktif();
        $data['formulir'] =  $this->Formulir_model->get_by_id_1();
        $data['user'] = $this->ion_auth->user()->row();
       $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();


        $data['page'] = 'peserta/Peserta_form';
        $this->load->view('template/backend', $data);
    }


    public function create_action()
    {
       // no pendaftaran
       $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();
        $data['formulir'] =  $this->Formulir_model->get_by_id_1();
        if ($data['formulir']->kode_formulir=="Ya") {
            if ($data['pengaturan']->penomoran=="otomatis") {
                    $data['no_pendaftaran'] = $this->Peserta_model-
>no_pendaftaran_luring();
                                $data['nodaftar'] = $data['formulir']->kode_luring."-
".$data['no_pendaftaran']."-L";
                        $no_pendaftaran = $data['nodaftar'];
                 } else if ($data['pengaturan']->penomoran=="manual") {
                    $data['no_pendaftaran'] = $this->input->post('no_pendaftaran',TRUE);
                                $data['nodaftar'] = $data['formulir']->kode_luring."-
".$data['no_pendaftaran']."-L";
                        $no_pendaftaran = $data['nodaftar'];
            }
          } else {
        if ($data['pengaturan']->penomoran=="otomatis") {
              $data['no_pendaftaran'] = $this->Peserta_model->no_pendaftaran();
                      $data['nodaftar'] = $data['formulir']->kode_luring."-
".$data['no_pendaftaran'];
                $no_pendaftaran = $data['nodaftar'];
            } else if ($data['pengaturan']->penomoran=="manual") {
              $data['no_pendaftaran'] = $this->input->post('no_pendaftaran',TRUE);
                      $data['nodaftar'] = $data['formulir']->kode_luring."-
".$data['no_pendaftaran'];
                $no_pendaftaran = $data['nodaftar'];
            }
          }

   $nama_peserta=$this->input->post('nama_peserta',TRUE);
    $this->load->library('ciqrcode'); //pemanggilan library QR CODE

    $config['cacheable']    = true; //boolean, the default is true
    $config['cachedir']     = './assets/'; //string, the default is application/cache/
    $config['errorlog']     = './assets/'; //string, the default is application/logs/
    $config['imagedir']     = './assets/uploads/image/grcode/'; //direktori penyimpanan qr
code
    $config['quality']      = true; //boolean, the default is true
    $config['size']         = '1024'; //interger, the default is 1024
    $config['black']        = array(224,255,255); // array, default is array(255,255,255)
    $config['white']        = array(70,130,180); // array, default is array(0,0,0)
    $this->ciqrcode->initialize($config);

    $qrcode=$no_pendaftaran.'.png'; //buat name dari qr code sesuai dengan no
pendaftaran

    $params['data'] = $no_pendaftaran.'-'.$nama_peserta; //data yang akan di jadikan QR
CODE
```

```
        $params['level'] = 'H'; //H=High
        $params['size'] = 10;
        $params['savename'] = FCPATH.$config['imagedir'].$qrcode; //simpan image QR CODE ke folder assets/uploads/image/grcode/
        $this->ciqrcode->generate($params); // fungsi untuk generate QR CODE


        $this->_rules();


        if ($this->form_validation->run() == FALSE) {
            $this->create();
        } else {
            $data = array(
                                'no_pendaftaran' => $no_pendaftaran,
                        'tanggal_daftar' => date('Y-m-d'),
                                'id_tahun' => $this->input->post('id_tahun',TRUE),
                                'id_jalur' => $this->input->post('id_jalur',TRUE),
                                'nama_peserta' => $this->input->post('nama_peserta',TRUE),
                                'jenis_kelamin' => $this->input->post('jenis_kelamin',TRUE),
                                'nisn' => $this->input->post('nisn',TRUE),
                                'nik' => $this->input->post('nik',TRUE),
                                'tempat_lahir' => $this->input->post('tempat_lahir',TRUE),
                        'tanggal_lahir' => date('Y-m-d', strtotime($this->input->post('tanggal_lahir',TRUE))),
                                'no_registrasi_akta_lahir' => $this->input->post('no_registrasi_akta_lahir',TRUE),
                                'agama' => $this->input->post('agama',TRUE),
                                'kewarganegaraan' => $this->input->post('kewarganegaraan',TRUE),
                                'berkebutuhan_khusus' => $this->input->post('berkebutuhan_khusus',TRUE),
                                'alamat' => $this->input->post('alamat',TRUE),
                                'rt' => $this->input->post('rt',TRUE),
                                'rw' => $this->input->post('rw',TRUE),
                                'nama_dusun' => $this->input->post('nama_dusun',TRUE),
                                'nama_kelurahan' => $this->input-
        >post('nama_kelurahan',TRUE),
                                'kecamatan' => $this->input->post('kecamatan',TRUE),
                                'kode_pos' => $this->input->post('kode_pos',TRUE),
                                'latitude' => $this->input->post('latitude',TRUE),
                                'longitude' => $this->input->post('longitude',TRUE),
                                'tempat_tinggal' => $this->input->post('tempat_tinggal',TRUE),
                                'moda_transportasi' => $this->input-
        >post('moda_transportasi',TRUE),
                                'no_kks' => $this->input->post('no_kks',TRUE),
                                'anak_ke' => $this->input->post('anak_ke',TRUE),
                                'penerima_kps_pkh' => $this->input-
        >post('penerima_kps_pkh',TRUE),
                                'no_kps_pkh' => $this->input->post('no_kps_pkh',TRUE),
                                'penerima_kip' => $this->input->post('penerima_kip',TRUE),
                                'no_kip' => $this->input->post('no_kip',TRUE),
                                'nama_tertera_di_kip' => $this->input-
        >post('nama_tertera_di_kip',TRUE),
                                'terima_fisik_kartu_kip' => $this->input-
        >post('terima_fisik_kartu_kip',TRUE),
                                'nama_ayah' => $this->input->post('nama_ayah',TRUE),
                                'nik_ayah' => $this->input->post('nik_ayah',TRUE),
                                'tahun_lahir_ayah' => $this->input-
        >post('tahun_lahir_ayah',TRUE),
                                'pendidikan_ayah' => $this->input-
        >post('pendidikan_ayah',TRUE),
                                'pekerjaan_ayah' => $this->input-
        >post('pekerjaan_ayah',TRUE),
                                'penghasilan_bulanan_ayah' => $this->input-
        >post('penghasilan_bulanan_ayah',TRUE),
                                'berkebutuhan_khusus_ayah' => $this->input-
        >post('berkebutuhan_khusus_ayah',TRUE),
                                'nama_ibu' => $this->input->post('nama_ibu',TRUE),
                                'nik_ibu' => $this->input->post('nik_ibu',TRUE),
                                'tahun_lahir_ibu' => $this->input-
        >post('tahun_lahir_ibu',TRUE),
        			'pendidikan_ibu' => $this->input->post('pendidikan_ibu',TRUE),
        			'pekerjaan_ibu' => $this->input->post('pekerjaan_ibu',TRUE),
        			'penghasilan_bulanan_ibu' => $this->input->post('penghasilan_bulanan_ibu',TRUE),
        			'berkebutuhan_khusus_ibu' => $this->input->post('berkebutuhan_khusus_ibu',TRUE),
        			'nama_wali' => $this->input->post('nama_wali',TRUE),
        			'nik_wali' => $this->input->post('nik_wali',TRUE),
        			'tahun_lahir_wali' => $this->input->post('tahun_lahir_wali',TRUE),
        			'pendidikan_wali' => $this->input->post('pendidikan_wali',TRUE),
        			'pekerjaan_wali' => $this->input->post('pekerjaan_wali',TRUE),
        			'penghasilan_bulanan_wali' => $this->input->post('penghasilan_bulanan_wali',TRUE),
        			'no_telepon_rumah' => $this->input->post('no_telepon_rumah',TRUE),
        			'nomor_hp' => $this->input->post('nomor_hp',TRUE),
        			'email' => $this->input->post('email',TRUE),
        			'jenis_ekstrakurikuler' => $this->input->post('jenis_ekstrakurikuler',TRUE),
        			'tinggi_badan' => $this->input->post('tinggi_badan',TRUE),
        			'berat_badan' => $this->input->post('berat_badan',TRUE),
        			'id_jarak' => $this->input->post('id_jarak',TRUE),
        			'jumlah_saudara_kandung' => $this->input->post('jumlah_saudara_kandung',TRUE),
        			'id_jurusan' => $this->input->post('id_jurusan',TRUE),
        			'pilihan_dua' => $this->input->post('pilihan_dua',TRUE),
        			'id_sekolah' => $this->input->post('id_sekolah',TRUE),
        			'no_un' => $this->input->post('no_un',TRUE),
        			'no_seri_ijazah' => $this->input->post('no_seri_ijazah',TRUE),
        			'no_seri_skhu' => $this->input->post('no_seri_skhu',TRUE),
        			'nilai_rapor' => $this->input->post('nilai_rapor',TRUE),
        			'nilai_usbn' => $this->input->post('nilai_usbn',TRUE),
                                        'nilai_unbk_unkp' => $this->input-
        >post('nilai_unbk_unkp',TRUE),
                                        'status' => $this->input->post('status',TRUE),
                                        'status_hasil' => $this->input->post('status_hasil',TRUE),
                                        'status_daftar_ulang' => $this->input-
        >post('status_daftar_ulang',TRUE),
                                        'id_users' => $this->input->post('id_users',TRUE),
                                        'qrcode' => $qrcode,
                                        'pilihan_sekolah_lain' => $this->input-
        >post('pilihan_sekolah_lain',TRUE),
                        ); 

                                $cekno = $this->Peserta_model->get_cek($no_pendaftaran);

                                if ($cekno) {
                    $this->session->set_flashdata('message', 'No Pendaftaran sudah ada');
                    $this->create();
                                } else {
                    $this->Peserta_model->insert($data);
                    $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
                    helper_log("add", "Menambah data peserta ".$data['nama_peserta']);
                    redirect(site_url('peserta'));
                  }
                }
            }

            public function update($id) 
            {
                $row = $this->Peserta_model->get_by_id($id);
                if ($row) {
                    $data = array(
                        'button' => 'Ubah',
                        'action' => site_url('peserta/update_action'),
                                        'id_peserta' => set_value('id_peserta', $row->id_peserta),
                                        // 'no_pendaftaran' => set_value('no_pendaftaran', $row-
        >no_pendaftaran),
                                        // 'tanggal_daftar' => set_value('tanggal_daftar', $row-
        >tanggal_daftar),
                                        // 'id_tahun' => set_value('id_tahun', $row->id_tahun),
                                        'id_jalur' => set_value('id_jalur', $row->id_jalur),
                                        'nama_peserta' => set_value('nama_peserta', $row-
        >nama_peserta),
                                        'jenis_kelamin' => set_value('jenis_kelamin', $row-
        >jenis_kelamin),
                                        'nisn' => set_value('nisn', $row->nisn),
                                        'nik' => set_value('nik', $row->nik),
                                        'tempat_lahir' => set_value('tempat_lahir', $row-
        >tempat_lahir),
                                        'tanggal_lahir' => set_value('tanggal_lahir', $row-
        >tanggal_lahir),
                                        'no_registrasi_akta_lahir' =>
        set_value('no_registrasi_akta_lahir', $row->no_registrasi_akta_lahir),
                                        'agama' => set_value('agama', $row->agama),
                                        'kewarganegaraan' => set_value('kewarganegaraan', $row-
        >kewarganegaraan),
                                        'berkebutuhan_khusus' => set_value('berkebutuhan_khusus',
        $row->berkebutuhan_khusus),
                                        'alamat' => set_value('alamat', $row->alamat),
                                        'rt' => set_value('rt', $row->rt),
                                        'rw' => set_value('rw', $row->rw),
                                        'nama_dusun' => set_value('nama_dusun', $row-
        >nama_dusun),
                                        'nama_kelurahan' => set_value('nama_kelurahan', $row-
        >nama_kelurahan),
                                        'kecamatan' => set_value('kecamatan', $row->kecamatan),
                                        'kode_pos' => set_value('kode_pos', $row->kode_pos),
                                        'latitude' => set_value('latitude', $row->latitude),
                                        'longitude' => set_value('longitude', $row->longitude),
                                        'tempat_tinggal' => set_value('tempat_tinggal', $row-
        >tempat_tinggal),
                                        'moda_transportasi' => set_value('moda_transportasi', $row-
        >moda_transportasi),
                                        'no_kks' => set_value('no_kks', $row->no_kks),
                                        'anak_ke' => set_value('anak_ke', $row->anak_ke),
                                        'penerima_kps_pkh' => set_value('penerima_kps_pkh', $row-
        >penerima_kps_pkh),
                                        'no_kps_pkh' => set_value('no_kps_pkh', $row->no_kps_pkh),
                                        'penerima_kip' => set_value('penerima_kip', $row-
        >penerima_kip),
                                        'no_kip' => set_value('no_kip', $row->no_kip),
                                        'nama_tertera_di_kip' => set_value('nama_tertera_di_kip',
        $row->nama_tertera_di_kip),
                                        'terima_fisik_kartu_kip' => set_value('terima_fisik_kartu_kip',
        $row->terima_fisik_kartu_kip),
                                        'nama_ayah' => set_value('nama_ayah', $row->nama_ayah),
                                        'nik_ayah' => set_value('nik_ayah', $row->nik_ayah),
                                        'tahun_lahir_ayah' => set_value('tahun_lahir_ayah', $row-
        >tahun_lahir_ayah),
                                        'pendidikan_ayah' => set_value('pendidikan_ayah', $row-
        >pendidikan_ayah),
                                        'pekerjaan_ayah' => set_value('pekerjaan_ayah', $row-
        >pekerjaan_ayah),
                                        'penghasilan_bulanan_ayah' =>
        set_value('penghasilan_bulanan_ayah', $row->penghasilan_bulanan_ayah),
                                        'berkebutuhan_khusus_ayah' =>
        set_value('berkebutuhan_khusus_ayah', $row->berkebutuhan_khusus_ayah),
                                        'nama_ibu' => set_value('nama_ibu', $row->nama_ibu),
                                        'nik_ibu' => set_value('nik_ibu', $row->nik_ibu),
                                        'tahun_lahir_ibu' => set_value('tahun_lahir_ibu', $row-
        >tahun_lahir_ibu),
                                        'pendidikan_ibu' => set_value('pendidikan_ibu', $row-
        >pendidikan_ibu),
                                        'pekerjaan_ibu' => set_value('pekerjaan_ibu', $row-
        >pekerjaan_ibu),
                                        'penghasilan_bulanan_ibu' =>
set_value('penghasilan_bulanan_ibu', $row->penghasilan_bulanan_ibu),
                                        'berkebutuhan_khusus_ibu' =>
set_value('berkebutuhan_khusus_ibu', $row->berkebutuhan_khusus_ibu),
                                        'nama_wali' => set_value('nama_wali', $row->nama_wali),
                                        'nik_wali' => set_value('nik_wali', $row->nik_wali),
                                        'tahun_lahir_wali' => set_value('tahun_lahir_wali', $row-
>tahun_lahir_wali),
                                        'pendidikan_wali' => set_value('pendidikan_wali', $row-
>pendidikan_wali),
                                        'pekerjaan_wali' => set_value('pekerjaan_wali', $row-
>pekerjaan_wali),
                                        'penghasilan_bulanan_wali' =>
set_value('penghasilan_bulanan_wali', $row->penghasilan_bulanan_wali),
                                        'no_telepon_rumah' => set_value('no_telepon_rumah', $row-
>no_telepon_rumah),
                                        'nomor_hp' => set_value('nomor_hp', $row->nomor_hp),
                                        'email' => set_value('email', $row->email),
                                        'jenis_ekstrakurikuler' => set_value('jenis_ekstrakurikuler',
$row->jenis_ekstrakurikuler),
                                        'tinggi_badan' => set_value('tinggi_badan', $row-
>tinggi_badan),
                                        'berat_badan' => set_value('berat_badan', $row->berat_badan),
                                        'id_jarak' => set_value('id_jarak', $row->id_jarak),
                                        'jumlah_saudara_kandung' =>
set_value('jumlah_saudara_kandung', $row->jumlah_saudara_kandung),
                                        'id_jurusan' => set_value('id_jurusan', $row->id_jurusan),
                                        'pilihan_dua' => set_value('pilihan_dua', $row->pilihan_dua),
                                        'id_sekolah' => set_value('id_sekolah', $row->id_sekolah),
                                        'no_un' => set_value('no_un', $row->no_un),
                                        'no_seri_ijazah' => set_value('no_seri_ijazah', $row-
>no_seri_ijazah),
                                        'no_seri_skhu' => set_value('no_seri_skhu', $row-
>no_seri_skhu),
                                        'nilai_rapor' => set_value('nilai_rapor', $row->nilai_rapor),
                                        'nilai_usbn' => set_value('nilai_usbn', $row->nilai_usbn),
                                        'nilai_unbk_unkp' => set_value('nilai_unbk_unkp', $row-
>nilai_unbk_unkp),
                                        'status' => set_value('status', $row->status),
                                        'status_hasil' => set_value('status_hasil', $row->status_hasil),
                                        'status_daftar_ulang' => set_value('status_daftar_ulang', $row-
>status_daftar_ulang),
                                        // 'id_users' => set_value('id_users', $row->id_users),
                                        'pilihan_sekolah_lain' => set_value('pilihan_sekolah_lain',
$row->pilihan_sekolah_lain),
                            );

          $data['title'] = 'Peserta';
          $data['subtitle'] = '';
          $data['crumb'] = [
              'Dashboard' => '',
          ];

          $data['code_js'] = 'peserta/codejs';
          $data['sekolah'] =$this->Sekolah_model->get_all();
          $data['jalur'] =$this->Jalur_model->get_all();
          $data['jarak'] =$this->Jarak_model->get_all();
          $data['jurusan'] =$this->Jurusan_model->get_all();
          $data['tahunpelajaran'] =$this->Tahunpelajaran_model->get_tahun_aktif();
          $data['peserta'] = $this->Peserta_model->get_by_id($id);
          $data['formulir'] =  $this->Formulir_model->get_by_id_1();
          $data['user'] = $this->ion_auth->user()->row();
        $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();

          $data['page'] = 'peserta/Peserta_form';
          $this->load->view('template/backend', $data);
          } else {
              $this->session->set_flashdata('message', 'Data tidak ditemukan');
              redirect(site_url('peserta'));
        }
    }

    public function update_action()
    {
        $this->_rules();

        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_peserta', TRUE));
        } else {
            $data = array(
                                        // 'no_pendaftaran' => $this->input-
>post('no_pendaftaran',TRUE),
                                // 'tanggal_daftar' => date('Y-m-d'),
                                        // 'id_tahun' => $this->input->post('id_tahun',TRUE),
                                        'id_jalur' => $this->input->post('id_jalur',TRUE),
                                        'nama_peserta' => $this->input->post('nama_peserta',TRUE),
                                        'jenis_kelamin' => $this->input->post('jenis_kelamin',TRUE),
                                        'nisn' => $this->input->post('nisn',TRUE),
                                        'nik' => $this->input->post('nik',TRUE),
                                        'tempat_lahir' => $this->input->post('tempat_lahir',TRUE),
                                'tanggal_lahir' => date('Y-m-d', strtotime($this->input-
>post('tanggal_lahir',TRUE))),
                                        'no_registrasi_akta_lahir' => $this->input-
>post('no_registrasi_akta_lahir',TRUE),
                                        'agama' => $this->input->post('agama',TRUE),
                                        'kewarganegaraan' => $this->input-
>post('kewarganegaraan',TRUE),
                                        'berkebutuhan_khusus' => $this->input-
>post('berkebutuhan_khusus',TRUE),
                                        'alamat' => $this->input->post('alamat',TRUE),
                                        'rt' => $this->input->post('rt',TRUE),
                                        'rw' => $this->input->post('rw',TRUE),
                                        'nama_dusun' => $this->input->post('nama_dusun',TRUE),
                                        'nama_kelurahan' => $this->input-
>post('nama_kelurahan',TRUE),
                                        'kecamatan' => $this->input->post('kecamatan',TRUE),
                                        'kode_pos' => $this->input->post('kode_pos',TRUE),
                                        'latitude' => $this->input->post('latitude',TRUE),
                                        'longitude' => $this->input->post('longitude',TRUE),
                                        'tempat_tinggal' => $this->input->post('tempat_tinggal',TRUE),
                                        'moda_transportasi' => $this->input-
>post('moda_transportasi',TRUE),
                                        'no_kks' => $this->input->post('no_kks',TRUE),
                                        'anak_ke' => $this->input->post('anak_ke',TRUE),
                                        'penerima_kps_pkh' => $this->input-
>post('penerima_kps_pkh',TRUE),
                                        'no_kps_pkh' => $this->input->post('no_kps_pkh',TRUE),
                                        'penerima_kip' => $this->input->post('penerima_kip',TRUE),
                                        'no_kip' => $this->input->post('no_kip',TRUE),
                                        'nama_tertera_di_kip' => $this->input-
>post('nama_tertera_di_kip',TRUE),
                                        'terima_fisik_kartu_kip' => $this->input-
>post('terima_fisik_kartu_kip',TRUE),
                                        'nama_ayah' => $this->input->post('nama_ayah',TRUE),
                                        'nik_ayah' => $this->input->post('nik_ayah',TRUE),
                                        'tahun_lahir_ayah' => $this->input-
>post('tahun_lahir_ayah',TRUE),
                                        'pendidikan_ayah' => $this->input-
>post('pendidikan_ayah',TRUE),
                                        'pekerjaan_ayah' => $this->input-
>post('pekerjaan_ayah',TRUE),
                                        'penghasilan_bulanan_ayah' => $this->input-
>post('penghasilan_bulanan_ayah',TRUE),
                                        'berkebutuhan_khusus_ayah' => $this->input-
>post('berkebutuhan_khusus_ayah',TRUE),
                                        'nama_ibu' => $this->input->post('nama_ibu',TRUE),
                                        'nik_ibu' => $this->input->post('nik_ibu',TRUE),
                                        'tahun_lahir_ibu' => $this->input-
>post('tahun_lahir_ibu',TRUE),
                                        'pendidikan_ibu' => $this->input-
>post('pendidikan_ibu',TRUE),
                                        'pekerjaan_ibu' => $this->input->post('pekerjaan_ibu',TRUE),
                                        'penghasilan_bulanan_ibu' => $this->input-
>post('penghasilan_bulanan_ibu',TRUE),
                                        'berkebutuhan_khusus_ibu' => $this->input-
>post('berkebutuhan_khusus_ibu',TRUE),
                                        'nama_wali' => $this->input->post('nama_wali',TRUE),
                                        'nik_wali' => $this->input->post('nik_wali',TRUE),
                                        'tahun_lahir_wali' => $this->input-
>post('tahun_lahir_wali',TRUE),
                                        'pendidikan_wali' => $this->input-
>post('pendidikan_wali',TRUE),
                                        'pekerjaan_wali' => $this->input-
>post('pekerjaan_wali',TRUE),
                                        'penghasilan_bulanan_wali' => $this->input-
>post('penghasilan_bulanan_wali',TRUE),
                                        'no_telepon_rumah' => $this->input-
>post('no_telepon_rumah',TRUE),
                                        'nomor_hp' => $this->input->post('nomor_hp',TRUE),
                                        'email' => $this->input->post('email',TRUE),
                                        'jenis_ekstrakurikuler' => $this->input-
>post('jenis_ekstrakurikuler',TRUE),
                                        'tinggi_badan' => $this->input->post('tinggi_badan',TRUE),
                                        'berat_badan' => $this->input->post('berat_badan',TRUE),
                                        'id_jarak' => $this->input->post('id_jarak',TRUE),
                                        'jumlah_saudara_kandung' => $this->input-
>post('jumlah_saudara_kandung',TRUE),
                                        'id_jurusan' => $this->input->post('id_jurusan',TRUE),
                                        'pilihan_dua' => $this->input->post('pilihan_dua',TRUE),
                                        'id_sekolah' => $this->input->post('id_sekolah',TRUE),
                                        'no_un' => $this->input->post('no_un',TRUE),
                                        'no_seri_ijazah' => $this->input->post('no_seri_ijazah',TRUE),
                                        'no_seri_skhu' => $this->input->post('no_seri_skhu',TRUE),
                                        'nilai_rapor' => $this->input->post('nilai_rapor',TRUE),
                                        'nilai_usbn' => $this->input->post('nilai_usbn',TRUE),
                                        'nilai_unbk_unkp' => $this->input-
>post('nilai_unbk_unkp',TRUE),
                                        'status' => $this->input->post('status',TRUE),
                                        'status_hasil' => $this->input->post('status_hasil',TRUE),
                                        'status_daftar_ulang' => $this->input-
>post('status_daftar_ulang',TRUE),
                                        'pilihan_sekolah_lain' => $this->input-
>post('pilihan_sekolah_lain',TRUE),
                        );

          $this->Peserta_model->update($this->input->post('id_peserta', TRUE), $data);
          $this->session->set_flashdata('message', 'Data Berhasil diubah');
          helper_log("edit", "Update data peserta ".$data['nama_peserta']);
          redirect(site_url('peserta'));
       }
    }

    public function status($id)
    {

        $row = $this->Peserta_model->get_id($id);

        if ($row) {
            $data = array(
                'button' => 'Ubah Status',
                'action' => site_url('peserta/status_action'),
                                        'id_peserta' => set_value('id_peserta', $row->id_peserta),
                                        'nama_peserta' => set_value('nama_peserta', $row-
>nama_peserta),
                                        'status' => set_value('status', $row->status),
                                        'status_hasil' => set_value('status_hasil', $row->status_hasil),
                                        'status_daftar_ulang' => set_value('status_daftar_ulang', $row-
>status_daftar_ulang),
                                        'catatan' => set_value('catatan', $row->catatan),
                                        'id_users' => set_value('id_users', $row->id_users),
                        );

        $data['title'] = 'Peserta';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];

        $data['page'] = 'peserta/Peserta_status';
        $this->load->view('template/backend', $data);
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('peserta'));
        }
    }

    public function status_action()
    {
       $data = array(
                'nama_peserta' => $this->input->post('nama_peserta',TRUE),
                        'status' => $this->input->post('status',TRUE),
                        'status_hasil' => $this->input->post('status_hasil',TRUE),
                        'status_daftar_ulang' => $this->input-
>post('status_daftar_ulang',TRUE),
                        'catatan' => $this->input->post('catatan',TRUE),
                );

       $this->Peserta_model->update($this->input->post('id_peserta', TRUE), $data);
       $this->session->set_flashdata('message', 'Data Berhasil diubah');
       helper_log("edit", "Update data status peserta ".$data['nama_peserta']);
       redirect(site_url('peserta'));
```

```
}



public function delete($id)
{
    $row = $this->Peserta_model->get_by_id($id);

    if ($row) {
        $this->Peserta_model->delete($id);
        $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus data peserta ".$row->nama_peserta);
        redirect(site_url('peserta'));
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('peserta'));
    }
}

public function deletebulk(){
   $delete = $this->Peserta_model->deletebulk();
    if($delete){
        $this->session->set_flashdata('message', 'Data Berhasil dihapus');
      helper_log("delete", "Menghapus multi data peserta");
    }else{
        $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
    }
    echo $delete;
}

public function _rules()
{
    $this->form_validation->set_rules('no_pendaftaran', 'no pendaftaran',
'trim|exact_length[4]|is_unique[peserta.no_pendaftaran]');
    $this->form_validation->set_rules('tanggal_daftar', 'tanggal daftar', 'trim');
    $this->form_validation->set_rules('id_tahun', 'id tahun', 'trim');
    $this->form_validation->set_rules('id_jalur', 'id jalur', 'trim|required');
    $this->form_validation->set_rules('nama_peserta', 'nama peserta', 'trim|required');
    $this->form_validation->set_rules('jenis_kelamin', 'jenis kelamin', 'trim|required');
    $this->form_validation->set_rules('nisn', 'nisn',
'trim|required|numeric|exact_length[10]');
    $this->form_validation->set_rules('nik', 'nik', 'trim|numeric|exact_length[16]');
    $this->form_validation->set_rules('tempat_lahir', 'tempat lahir', 'trim|required');
    $this->form_validation->set_rules('tanggal_lahir', 'tanggal lahir', 'trim|required');
    $this->form_validation->set_rules('no_registrasi_akta_lahir', 'no registrasi akta lahir',
'trim');
    $this->form_validation->set_rules('agama', 'agama', 'trim|required');
    $this->form_validation->set_rules('kewarganegaraan', 'kewarganegaraan', 'trim');
    $this->form_validation->set_rules('berkebutuhan_khusus', 'berkebutuhan khusus',
'trim');
    $this->form_validation->set_rules('alamat', 'alamat', 'trim|required');
    $this->form_validation->set_rules('rt', 'rt', 'trim|numeric');
    $this->form_validation->set_rules('rw', 'rw', 'trim|numeric');
    $this->form_validation->set_rules('nama_dusun', 'nama dusun', 'trim');
    $this->form_validation->set_rules('nama_kelurahan', 'nama kelurahan', 'trim');
    $this->form_validation->set_rules('kecamatan', 'kecamatan', 'trim');
    $this->form_validation->set_rules('kode_pos', 'kode pos', 'trim|numeric');
    $this->form_validation->set_rules('latitude', 'latitude', 'trim');
    $this->form_validation->set_rules('longitude', 'longitude', 'trim');
    $this->form_validation->set_rules('tempat_tinggal', 'tempat tinggal', 'trim');
    $this->form_validation->set_rules('moda_transportasi', 'moda transportasi', 'trim');
    $this->form_validation->set_rules('no_kks', 'no kks', 'trim|exact_length[6]');
    $this->form_validation->set_rules('anak_ke', 'anak ke', 'trim|numeric');
    $this->form_validation->set_rules('penerima_kps_pkh', 'penerima kps pkh', 'trim');
    $this->form_validation->set_rules('no_kps_pkh', 'no kps pkh', 'trim');
    $this->form_validation->set_rules('penerima_kip', 'penerima kip', 'trim');

    $this->form_validation->set_rules('no_kip', 'no kip', 'trim|exact_length[6]');

    $this->form_validation->set_rules('nama_tertera_di_kip', 'nama tertera di kip', 'trim');

    $this->form_validation->set_rules('terima_fisik_kartu_kip', 'terima fisik kartu kip',
'trim');

    $this->form_validation->set_rules('nama_ayah', 'nama ayah', 'trim|required');

    $this->form_validation->set_rules('nik_ayah', 'nik ayah',
'trim|numeric|exact_length[16]');

    $this->form_validation->set_rules('tahun_lahir_ayah', 'tahun lahir ayah',
'trim|numeric|exact_length[4]');

    $this->form_validation->set_rules('pendidikan_ayah', 'pendidikan ayah', 'trim');

    $this->form_validation->set_rules('pekerjaan_ayah', 'pekerjaan ayah', 'trim');

    $this->form_validation->set_rules('penghasilan_bulanan_ayah', 'penghasilan bulanan
ayah', 'trim');

    $this->form_validation->set_rules('berkebutuhan_khusus_ayah', 'berkebutuhan
khusus ayah', 'trim');

    $this->form_validation->set_rules('nama_ibu', 'nama ibu', 'trim|required');

    $this->form_validation->set_rules('nik_ibu', 'nik ibu',
'trim|numeric|exact_length[16]');

    $this->form_validation->set_rules('tahun_lahir_ibu', 'tahun lahir ibu',
'trim|numeric|exact_length[4]');

    $this->form_validation->set_rules('pendidikan_ibu', 'pendidikan ibu', 'trim');

    $this->form_validation->set_rules('pekerjaan_ibu', 'pekerjaan ibu', 'trim');

    $this->form_validation->set_rules('penghasilan_bulanan_ibu', 'penghasilan bulanan
ibu', 'trim');

    $this->form_validation->set_rules('berkebutuhan_khusus_ibu', 'berkebutuhan khusus
ibu', 'trim');

    $this->form_validation->set_rules('nama_wali', 'nama wali', 'trim');

    $this->form_validation->set_rules('nik_wali', 'nik wali',
'trim|numeric|exact_length[16]');

    $this->form_validation->set_rules('tahun_lahir_wali', 'tahun lahir wali',
'trim|numeric|exact_length[4]');

    $this->form_validation->set_rules('pendidikan_wali', 'pendidikan wali', 'trim');

    $this->form_validation->set_rules('pekerjaan_wali', 'pekerjaan wali', 'trim');

    $this->form_validation->set_rules('penghasilan_bulanan_wali', 'penghasilan bulanan
wali', 'trim');
```

```
    $this->form_validation->set_rules('no_telepon_rumah', 'no telepon rumah', 
'trim|numeric');
    $this->form_validation->set_rules('nomor_hp', 'nomor hp', 'trim|numeric');
    $this->form_validation->set_rules('email', 'email', 'trim|valid_email');
    $this->form_validation->set_rules('jenis_ekstrakurikuler', 'jenis ekstrakurikuler', 
'trim');
    $this->form_validation->set_rules('tinggi_badan', 'tinggi badan', 'trim|numeric');
    $this->form_validation->set_rules('berat_badan', 'berat badan', 'trim|numeric');
    $this->form_validation->set_rules('id_jarak', 'id jarak', 'trim|required');
    $this->form_validation->set_rules('jumlah_saudara_kandung', 'jumlah saudara 
kandung', 'trim|numeric');
    $this->form_validation->set_rules('id_jurusan', 'jurusan pilihan satu', 'trim');
    $this->form_validation->set_rules('pilihan_dua', 'jurusan pilihan dua', 'trim');
    $this->form_validation->set_rules('id_sekolah', 'id sekolah', 'trim|required');
    $this->form_validation->set_rules('no_un', 'no un', 'trim');
    $this->form_validation->set_rules('no_seri_ijazah', 'no seri ijazah', 'trim');
    $this->form_validation->set_rules('no_seri_skhu', 'no seri skhu', 'trim');
    $this->form_validation->set_rules('nilai_rapor', 'nilai rapor', 'trim|numeric');
    $this->form_validation->set_rules('nilai_usbn', 'nilai usbn', 'trim|numeric');
    $this->form_validation->set_rules('nilai_unbk_unkp', 'nilai un', 'trim|numeric');
    $this->form_validation->set_rules('status', 'status', 'trim');
    $this->form_validation->set_rules('status_hasil', 'status hasil', 'trim');
    $this->form_validation->set_rules('status_daftar_ulang', 'status daftar ulang', 'trim');
    $this->form_validation->set_rules('id_users', 'id users', 'trim');
    $this->form_validation->set_rules('pilihan_sekolah_lain', 'pilihan sekolah lain', 'trim');

    $this->form_validation->set_rules('id_peserta', 'id_peserta', 'trim');
    $this->form_validation->set_error_delimiters('<span class="text-danger">', 
'</span>');
  }

  public function excel()
  {
```

```
$this->load->helper('exportexcel','date');
$namaFile = "peserta.xls";
$judul = "peserta";
$tablehead = 0;
$tablebody = 1;
$nourut = 1;
//penulisan header
header("Pragma: public");
header("Expires: 0");
header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
header("Content-Type: application/force-download");
header("Content-Type: application/octet-stream");
header("Content-Type: application/download");
header("Content-Disposition: attachment;filename=" . $namaFile . "");
header("Content-Transfer-Encoding: binary ");

xlsBOF();

$kolomhead = 0;
xlsWriteLabel($tablehead, $kolomhead++, "No");
        xlsWriteLabel($tablehead, $kolomhead++, "No Pendaftaran");
        xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Daftar");
        xlsWriteLabel($tablehead, $kolomhead++, "Tahun");
        xlsWriteLabel($tablehead, $kolomhead++, "Nama Peserta");
        xlsWriteLabel($tablehead, $kolomhead++, "JK");
        xlsWriteLabel($tablehead, $kolomhead++, "NISN");
        xlsWriteLabel($tablehead, $kolomhead++, "NIK");
        xlsWriteLabel($tablehead, $kolomhead++, "Tempat Lahir");
        xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Lahir");
        xlsWriteLabel($tablehead, $kolomhead++, "No Registrasi Akta Lahir");
        xlsWriteLabel($tablehead, $kolomhead++, "Agama");
        xlsWriteLabel($tablehead, $kolomhead++, "Kewarganegaraan");
xlsWriteLabel($tablehead, $kolomhead++, "Berkebutuhan Khusus");
xlsWriteLabel($tablehead, $kolomhead++, "Alamat");
xlsWriteLabel($tablehead, $kolomhead++, "RT");
xlsWriteLabel($tablehead, $kolomhead++, "RW");
xlsWriteLabel($tablehead, $kolomhead++, "Nama Dusun");
xlsWriteLabel($tablehead, $kolomhead++, "Nama Kelurahan");
xlsWriteLabel($tablehead, $kolomhead++, "Kecamatan");
xlsWriteLabel($tablehead, $kolomhead++, "Kode Pos");
xlsWriteLabel($tablehead, $kolomhead++, "Latitude");
xlsWriteLabel($tablehead, $kolomhead++, "Longitude");
xlsWriteLabel($tablehead, $kolomhead++, "Tempat Tinggal");
xlsWriteLabel($tablehead, $kolomhead++, "Moda Transportasi");
xlsWriteLabel($tablehead, $kolomhead++, "No KKS");
xlsWriteLabel($tablehead, $kolomhead++, "Anak Ke");
xlsWriteLabel($tablehead, $kolomhead++, "Penerima KPS/PKH");
xlsWriteLabel($tablehead, $kolomhead++, "No KPS/PKH");
xlsWriteLabel($tablehead, $kolomhead++, "Penerima KIP`");
xlsWriteLabel($tablehead, $kolomhead++, "No KIP");
xlsWriteLabel($tablehead, $kolomhead++, "Nama Tertera Di KIP");
xlsWriteLabel($tablehead, $kolomhead++, "Terima Fisik Kartu KIP");
xlsWriteLabel($tablehead, $kolomhead++, "Hobi");
xlsWriteLabel($tablehead, $kolomhead++, "Nama Ayah");
xlsWriteLabel($tablehead, $kolomhead++, "NIK Ayah");
xlsWriteLabel($tablehead, $kolomhead++, "Tahun Lahir Ayah");
xlsWriteLabel($tablehead, $kolomhead++, "Pendidikan Ayah");
xlsWriteLabel($tablehead, $kolomhead++, "Pekerjaan Ayah");
xlsWriteLabel($tablehead, $kolomhead++, "Penghasilan Bulanan Ayah");
xlsWriteLabel($tablehead, $kolomhead++, "Berkebutuhan Khusus Ayah");
xlsWriteLabel($tablehead, $kolomhead++, "Nama Ibu");
xlsWriteLabel($tablehead, $kolomhead++, "NIK Ibu");
xlsWriteLabel($tablehead, $kolomhead++, "Tahun Lahir Ibu");
xlsWriteLabel($tablehead, $kolomhead++, "Pendidikan Ibu");
xlsWriteLabel($tablehead, $kolomhead++, "Pekerjaan Ibu");
xlsWriteLabel($tablehead, $kolomhead++, "Penghasilan Bulanan Ibu");
xlsWriteLabel($tablehead, $kolomhead++, "Berkebutuhan Khusus Ibu");
xlsWriteLabel($tablehead, $kolomhead++, "Nama Wali");
xlsWriteLabel($tablehead, $kolomhead++, "NIK Wali");
xlsWriteLabel($tablehead, $kolomhead++, "Tahun Lahir Wali");
xlsWriteLabel($tablehead, $kolomhead++, "Pendidikan Wali");
xlsWriteLabel($tablehead, $kolomhead++, "Pekerjaan Wali");
xlsWriteLabel($tablehead, $kolomhead++, "Penghasilan Bulanan Wali");
xlsWriteLabel($tablehead, $kolomhead++, "No Telepon Rumah");
xlsWriteLabel($tablehead, $kolomhead++, "Nomor HP");
xlsWriteLabel($tablehead, $kolomhead++, "Email");
xlsWriteLabel($tablehead, $kolomhead++, "Tinggi Badan");
xlsWriteLabel($tablehead, $kolomhead++, "Berat Badan");
xlsWriteLabel($tablehead, $kolomhead++, "Jarak ke sekolah");
xlsWriteLabel($tablehead, $kolomhead++, "Jumlah Saudara Kandung");
xlsWriteLabel($tablehead, $kolomhead++, "Jurusan Pilihan 1");
xlsWriteLabel($tablehead, $kolomhead++, "Jurusan Pilihan 2");
xlsWriteLabel($tablehead, $kolomhead++, "Asal Sekolah");
xlsWriteLabel($tablehead, $kolomhead++, "No UN");
xlsWriteLabel($tablehead, $kolomhead++, "No Seri Ijazah");
xlsWriteLabel($tablehead, $kolomhead++, "No Seri SKHU");
xlsWriteLabel($tablehead, $kolomhead++, "Sekolah Pilihan 2");
xlsWriteLabel($tablehead, $kolomhead++, "Jalur");
xlsWriteLabel($tablehead, $kolomhead++, "Nilai Rapor");
xlsWriteLabel($tablehead, $kolomhead++, "Nilai USBN");
xlsWriteLabel($tablehead, $kolomhead++, "Nilai UN");
xlsWriteLabel($tablehead, $kolomhead++, "Jumlah Nilai");
xlsWriteLabel($tablehead, $kolomhead++, "Poin Jarak");
xlsWriteLabel($tablehead, $kolomhead++, "Poin Prestasi");
xlsWriteLabel($tablehead, $kolomhead++, "Nilai Akhir");
xlsWriteLabel($tablehead, $kolomhead++, "Usia");
```

```
            xlsWriteLabel($tablehead, $kolomhead++, "Status");
            xlsWriteLabel($tablehead, $kolomhead++, "Status Hasil");
            xlsWriteLabel($tablehead, $kolomhead++, "Status Daftar Ulang");
            xlsWriteLabel($tablehead, $kolomhead++, "Catatan");
            // xlsWriteLabel($tablehead, $kolomhead++, "Id Users");

    foreach ($this->Peserta_model->get_all() as $data) {
      $kolombody = 0;
      $id=$data->id_peserta;
      $id_peserta=$data->id_peserta;
      // total nilai
      $totalnilai=$data->nilai_rapor+$data->nilai_usbn+$data->nilai_unbk_unkp;
      // total poin
      foreach ($this->Prestasipeserta_model->sumpoin($id_peserta) as $poin) {
            $totalpoin=$poin->totalpoin;
      }
      // bobot by id_peserta
      foreach ($this->Peserta_model->bobot($id) as $bobot) {
            $bobotjarak=($bobot->bobot_jarak!=0)?($bobot->bobot_jarak*$data->skor_jarak)/100:0;
            $bobotnilai=($bobot->bobot_nilai!=0)?($bobot->bobot_nilai*$totalnilai)/100:0;
            $bobotprestasi=($bobot->bobot_prestasi!=0)?($bobot->bobot_prestasi*$totalpoin)/100:0;
      }
      // nilai akhir
      if ($this->Peserta_model->bobot($id)) {
          $nilaiakhir=$bobotjarak+$bobotnilai+$bobotprestasi;
      } else {
          $nilaiakhir='bobot jalur blm ada';
      }

      // umur by id_peserta
```

```
foreach ($this->Peserta_model->tgl_lhr($id) as $tgl) {
        $tanggal_lahir=new DateTime($tgl->tanggal_lahir);
}
        $today = new DateTime('today');
        $y = $today->diff($tanggal_lahir)->y;
        $m = $today->diff($tanggal_lahir)->m;
        $d = $today->diff($tanggal_lahir)->d;
        $usia = $y . " tahun " . $m . " bulan " . $d . " hari";

//ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
xlsWriteNumber($tablebody, $kolombody++, $nourut);
  xlsWriteLabel($tablebody, $kolombody++, $data->no_pendaftaran);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_daftar);
  xlsWriteLabel($tablebody, $kolombody++, $data->tahun_pelajaran);
  xlsWriteLabel($tablebody, $kolombody++, $data->nama_peserta);
  xlsWriteLabel($tablebody, $kolombody++, $data->jenis_kelamin);
  xlsWriteLabel($tablebody, $kolombody++, $data->nisn);
  xlsWriteLabel($tablebody, $kolombody++, $data->nik);
  xlsWriteLabel($tablebody, $kolombody++, $data->tempat_lahir);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_lahir);
  xlsWriteLabel($tablebody, $kolombody++, $data->no_registrasi_akta_lahir);
  xlsWriteLabel($tablebody, $kolombody++, $data->agama);
  xlsWriteLabel($tablebody, $kolombody++, $data->kewarganegaraan);
  xlsWriteLabel($tablebody, $kolombody++, $data->berkebutuhan_khusus);
  xlsWriteLabel($tablebody, $kolombody++, $data->alamat);
  xlsWriteLabel($tablebody, $kolombody++, $data->rt);
  xlsWriteLabel($tablebody, $kolombody++, $data->rw);
  xlsWriteLabel($tablebody, $kolombody++, $data->nama_dusun);
  xlsWriteLabel($tablebody, $kolombody++, $data->nama_kelurahan);
  xlsWriteLabel($tablebody, $kolombody++, $data->kecamatan);
  xlsWriteLabel($tablebody, $kolombody++, $data->kode_pos);
  xlsWriteLabel($tablebody, $kolombody++, $data->latitude);
xlsWriteLabel($tablebody, $kolombody++, $data->longitude);
xlsWriteLabel($tablebody, $kolombody++, $data->tempat_tinggal);
xlsWriteLabel($tablebody, $kolombody++, $data->moda_transportasi);
xlsWriteLabel($tablebody, $kolombody++, $data->no_kks);
xlsWriteLabel($tablebody, $kolombody++, $data->anak_ke);
xlsWriteLabel($tablebody, $kolombody++, $data->penerima_kps_pkh);
xlsWriteLabel($tablebody, $kolombody++, $data->no_kps_pkh);
xlsWriteLabel($tablebody, $kolombody++, $data->penerima_kip);
xlsWriteLabel($tablebody, $kolombody++, $data->no_kip);
xlsWriteLabel($tablebody, $kolombody++, $data->nama_tertera_di_kip);
xlsWriteLabel($tablebody, $kolombody++, $data->terima_fisik_kartu_kip);
xlsWriteLabel($tablebody, $kolombody++, $data->jenis_ekstrakurikuler);
xlsWriteLabel($tablebody, $kolombody++, $data->nama_ayah);
xlsWriteLabel($tablebody, $kolombody++, $data->nik_ayah);
xlsWriteLabel($tablebody, $kolombody++, $data->tahun_lahir_ayah);
xlsWriteLabel($tablebody, $kolombody++, $data->pendidikan_ayah);
xlsWriteLabel($tablebody, $kolombody++, $data->pekerjaan_ayah);
xlsWriteLabel($tablebody, $kolombody++, $data->penghasilan_bulanan_ayah);
xlsWriteLabel($tablebody, $kolombody++, $data->berkebutuhan_khusus_ayah);
xlsWriteLabel($tablebody, $kolombody++, $data->nama_ibu);
xlsWriteLabel($tablebody, $kolombody++, $data->nik_ibu);
xlsWriteLabel($tablebody, $kolombody++, $data->tahun_lahir_ibu);
xlsWriteLabel($tablebody, $kolombody++, $data->pendidikan_ibu);
xlsWriteLabel($tablebody, $kolombody++, $data->pekerjaan_ibu);
xlsWriteLabel($tablebody, $kolombody++, $data->penghasilan_bulanan_ibu);
xlsWriteLabel($tablebody, $kolombody++, $data->berkebutuhan_khusus_ibu);
xlsWriteLabel($tablebody, $kolombody++, $data->nama_wali);
xlsWriteLabel($tablebody, $kolombody++, $data->nik_wali);
xlsWriteLabel($tablebody, $kolombody++, $data->tahun_lahir_wali);
xlsWriteLabel($tablebody, $kolombody++, $data->pendidikan_wali);
xlsWriteLabel($tablebody, $kolombody++, $data->pekerjaan_wali);
xlsWriteLabel($tablebody, $kolombody++, $data->penghasilan_bulanan_wali);
```

```
  xlsWriteLabel($tablebody, $kolombody++, $data->no_telepon_rumah);
  xlsWriteLabel($tablebody, $kolombody++, $data->nomor_hp);
  xlsWriteLabel($tablebody, $kolombody++, $data->email);
  xlsWriteLabel($tablebody, $kolombody++, $data->tinggi_badan);
  xlsWriteLabel($tablebody, $kolombody++, $data->berat_badan);
  xlsWriteLabel($tablebody, $kolombody++, $data->jarak);
  xlsWriteLabel($tablebody, $kolombody++, $data->jumlah_saudara_kandung);
  xlsWriteLabel($tablebody, $kolombody++, $data->nama_jurusan);
  xlsWriteLabel($tablebody, $kolombody++, $data->pilihan_dua);
  xlsWriteLabel($tablebody, $kolombody++, $data->asal_sekolah);
  xlsWriteLabel($tablebody, $kolombody++, $data->no_un);
  xlsWriteLabel($tablebody, $kolombody++, $data->no_seri_ijazah);
  xlsWriteLabel($tablebody, $kolombody++, $data->no_seri_skhu);
  xlsWriteLabel($tablebody, $kolombody++, $data->pilihan_sekolah_lain);
  xlsWriteLabel($tablebody, $kolombody++, $data->jalur);
  xlsWriteLabel($tablebody, $kolombody++, $data->nilai_rapor);
  xlsWriteLabel($tablebody, $kolombody++, $data->nilai_usbn);
  xlsWriteLabel($tablebody, $kolombody++, $data->nilai_unbk_unkp);
  xlsWriteLabel($tablebody, $kolombody++, $totalnilai);
  xlsWriteLabel($tablebody, $kolombody++, $data->skor_jarak);
  xlsWriteNumber($tablebody, $kolombody++, $totalpoin);
  xlsWriteNumber($tablebody, $kolombody++, $nilaiakhir);
  xlsWriteLabel($tablebody, $kolombody++, $usia);
  xlsWriteLabel($tablebody, $kolombody++, $data->status);
  xlsWriteLabel($tablebody, $kolombody++, $data->status_hasil);
  xlsWriteLabel($tablebody, $kolombody++, $data->status_daftar_ulang);
  xlsWriteLabel($tablebody, $kolombody++, $data->catatan);
  // xlsWriteNumber($tablebody, $kolombody++, $data->id_users);

  $tablebody++;
$nourut++;
}
```

```
    xlsEOF();
    exit();
}

public function word()
{
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=peserta.doc");

    $data = array(
        'peserta_data' => $this->Peserta_model->get_all(),
        'start' => 0
    );
    $this->load->view('peserta/Peserta_doc',$data);
}

  public function printdoc(){
    $data = array(
        'peserta_data' => $this->Peserta_model->get_all(),
        'start' => 0
    );
    $this->load->view('peserta/Peserta_print', $data);
}

  public function printform($id){
            $data = array(
        'peserta' => $this->Peserta_model->get_by_id($id),
        'start' => 0
    );

    $id_peserta = $id;
```

```
        $data['berkas']=$this->Berkas_model->get_foto($id_peserta);
        $data['prestasipeserta']=$this->Prestasipeserta_model-
>get_all_prestasi($id_peserta);
        $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();
        $data['pengumuman'] = $this->Pengumuman_model->get_by_formulir();

        $data['formulir'] =  $this->Formulir_model->get_by_id_1();
        helper_log("print", "Cetak bukti pendaftaran ".$data['peserta']->nama_peserta);

            $mpdf = new \Mpdf\Mpdf(['format' => 'Legal']);
            $html = $this->load->view('peserta/Print_formulir', $data,true);
            $mpdf->WriteHTML($html);
            $mpdf->Output('formulir.pdf','I');

    }

     public function printSKL($id)
     {
            $data = array(
          'peserta' => $this->Peserta_model->get_by_id($id),
          'start' => 0
        );

        $id_peserta = $id;
        $data['prestasipeserta']=$this->Prestasipeserta_model-
>get_all_prestasi($id_peserta);
        $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();
        $data['pengumuman'] = $this->Pengumuman_model->get_by_skl();

        $data['formulir'] =  $this->Formulir_model->get_by_id_1();
        helper_log("print", "Cetak bukti SKL ".$data['peserta']->nama_peserta);

            $mpdf = new \Mpdf\Mpdf(['format' => 'A4']);
        $html = $this->load->view('peserta/Print_SKL', $data,true);
        $mpdf->WriteHTML($html);
        $mpdf->Output('SKL.pdf','I');

}

 public function terima_all(){
        $hasil="Di Terima";
        $this->Peserta_model->get_hasil($hasil);
        helper_log("edit", "Update data status hasil semua peserta");
   redirect('peserta');
}

 public function reset_all(){
        $hasil="Belum ada";
        $this->Peserta_model->get_reset($hasil);
        helper_log("edit", "Update data status hasil semua peserta");
   redirect('peserta');
}

public function nilai_peserta()
{
    $data['title'] = 'Nilai Peserta';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Peserta' => '',
        ];

    // $data['code_js'] = 'peserta/codejs';
    $data['peserta']=$this->Peserta_model->get_all();
   $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'peserta/Peserta_nilai';
```

```
     $this->load->view('template/backend', $data);
}

public function rekap_nilai()
{
    $this->load->helper('exportexcel','date');
    $namaFile = "rekap_nilai.xls";
    $judul = "rekap";
    $tablehead = 0;
    $tablebody = 1;
    $nourut = 1;
    //penulisan header
    header("Pragma: public");
    header("Expires: 0");
    header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
    header("Content-Type: application/force-download");
    header("Content-Type: application/octet-stream");
    header("Content-Type: application/download");
    header("Content-Disposition: attachment;filename=" . $namaFile . "");
    header("Content-Transfer-Encoding: binary ");

    xlsBOF();

    $kolomhead = 0;
    xlsWriteLabel($tablehead, $kolomhead++, "No");
            xlsWriteLabel($tablehead, $kolomhead++, "No Pendaftaran");
            xlsWriteLabel($tablehead, $kolomhead++, "Nama Peserta");
            xlsWriteLabel($tablehead, $kolomhead++, "JK");
            xlsWriteLabel($tablehead, $kolomhead++, "NISN");
            xlsWriteLabel($tablehead, $kolomhead++, "Asal Sekolah");
            xlsWriteLabel($tablehead, $kolomhead++, "Sekolah Pilihan 2");
            xlsWriteLabel($tablehead, $kolomhead++, "Jurusan Pil 1");
```

```
                xlsWriteLabel($tablehead, $kolomhead++, "Jurusan Pil 2");
                xlsWriteLabel($tablehead, $kolomhead++, "Jalur");
                xlsWriteLabel($tablehead, $kolomhead++, "Nilai Rapor");
                xlsWriteLabel($tablehead, $kolomhead++, "Nilai USBN");
                xlsWriteLabel($tablehead, $kolomhead++, "Nilai UN");
                xlsWriteLabel($tablehead, $kolomhead++, "Jumlah Nilai");
                xlsWriteLabel($tablehead, $kolomhead++, "Poin Jarak");
                xlsWriteLabel($tablehead, $kolomhead++, "Poin Prestasi");
                xlsWriteLabel($tablehead, $kolomhead++, "Nilai Akhir");
                xlsWriteLabel($tablehead, $kolomhead++, "Usia");

                foreach ($this->Peserta_model->get_all() as $data) {
        $kolombody = 0;
        $id=$data->id_peserta;
        $id_peserta=$data->id_peserta;
        // total nilai
        $totalnilai=$data->nilai_rapor+$data->nilai_usbn+$data->nilai_unbk_unkp;
        // total poin
        foreach ($this->Prestasipeserta_model->sumpoin($id_peserta) as $poin) {
                $totalpoin=$poin->totalpoin;
        }
        // bobot by id_peserta
        foreach ($this->Peserta_model->bobot($id) as $bobot) {
                $bobotjarak=($bobot->bobot_jarak!=0)?($bobot->bobot_jarak*$data->skor_jarak)/100:0;
                $bobotnilai=($bobot->bobot_nilai!=0)?($bobot->bobot_nilai*$totalnilai)/100:0;
                $bobotprestasi=($bobot->bobot_prestasi!=0)?($bobot->bobot_prestasi*$totalpoin)/100:0;
        }
        // nilai akhir
        if ($this->Peserta_model->bobot($id)) {
            $nilaiakhir=$bobotjarak+$bobotnilai+$bobotprestasi;
```

```
} else {
    $nilaiakhir='bobot jalur blm ada';
}

// umur by id_peserta
foreach ($this->Peserta_model->tgl_lhr($id) as $tgl) {
          $tanggal_lahir=new DateTime($tgl->tanggal_lahir);
}
          $today = new DateTime('today');
          $y = $today->diff($tanggal_lahir)->y;
          $m = $today->diff($tanggal_lahir)->m;
          $d = $today->diff($tanggal_lahir)->d;
          $usia = $y . " tahun " . $m . " bulan " . $d . " hari";

//ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
xlsWriteNumber($tablebody, $kolombody++, $nourut);
  xlsWriteLabel($tablebody, $kolombody++, $data->no_pendaftaran);
  xlsWriteLabel($tablebody, $kolombody++, $data->nama_peserta);
  xlsWriteLabel($tablebody, $kolombody++, $data->jenis_kelamin);
  xlsWriteLabel($tablebody, $kolombody++, $data->nisn);
  xlsWriteLabel($tablebody, $kolombody++, $data->asal_sekolah);
  xlsWriteLabel($tablebody, $kolombody++, $data->pilihan_sekolah_lain);
  xlsWriteLabel($tablebody, $kolombody++, $data->nama_jurusan);
          xlsWriteLabel($tablebody, $kolombody++, $data->pilihan_dua);
  xlsWriteLabel($tablebody, $kolombody++, $data->jalur);
  xlsWriteLabel($tablebody, $kolombody++, $data->nilai_rapor);
  xlsWriteLabel($tablebody, $kolombody++, $data->nilai_usbn);
  xlsWriteLabel($tablebody, $kolombody++, $data->nilai_unbk_unkp);
  xlsWriteLabel($tablebody, $kolombody++, $totalnilai);
  xlsWriteLabel($tablebody, $kolombody++, $data->skor_jarak);
  xlsWriteNumber($tablebody, $kolombody++, $totalpoin);
  xlsWriteNumber($tablebody, $kolombody++, $nilaiakhir);
```

```
      xlsWriteLabel($tablebody, $kolombody++, $usia);

      $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}

public function tambah_sekolah()
{
    $this->_rules_sekolah();

    if ($this->form_validation->run() == FALSE) {
        // $this->create();
        $this->session->set_flashdata('message',
            form_error('npsn_sekolah').
            form_error('asal_sekolah').
            form_error('status_sekolah').
            form_error('kecamatan')
        );
        redirect(site_url('peserta/create'));
    } else {
    $data = array(
              'npsn_sekolah' => $this->input->post('npsn_sekolah',TRUE),
              'asal_sekolah' => $this->input->post('asal_sekolah',TRUE),
              'alamat_sekolah' => $this->input->post('alamat_sekolah',TRUE),
              'kelurahan' => $this->input->post('kelurahan',TRUE),
              'status_sekolah' => $this->input->post('status_sekolah',TRUE),
              'kecamatan' => $this->input->post('kecamatan',TRUE),
      );
```

```
        $this->Sekolah_model->insert($data);
        $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
        helper_log("add", "Menambah data sekolah ".$data['asal_sekolah']);
        redirect(site_url('peserta/create'));
        }
    }

    public function _rules_sekolah()
    {
      $this->form_validation->set_rules('npsn_sekolah', 'npsn sekolah',
'trim|required|numeric|exact_length[8]|is_unique[sekolah.npsn_sekolah]',
        array(
                'required'      => 'NPSN Sekolah tidak boleh kosong ',
                'numeric'       => 'NPSN Sekolah hanya angka ',
                'is_unique'     => 'NPSN Sekolah sudah ada '
        ));
      $this->form_validation->set_rules('id_sekolah', 'id_sekolah', 'trim');
      $this->form_validation->set_error_delimiters('<span class="text">', '</span>');
    }

    public function update_penomoran()
    {
        $data = array(
                'penomoran' => $this->input->post('penomoran',TRUE),
            );
        $this->Peserta_model->update_penomoran($this->input->post('id_identitas',
TRUE), $data);
        helper_log("edit", "Update data pengaturan");
        redirect(site_url('peserta/create'));
    }
}
/* End of file Peserta.php */
```

- Prestasi

```
<?php

if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Prestasi extends CI_Controller
{
    function __construct()
    {
        parent::__construct();
        $c_url = $this->router->fetch_class();
        $this->layout->auth();
        $this->layout->auth_privilege($c_url);
        $this->load->model('Prestasi_model');
        $this->load->model('Pengaturan_model');
        $this->load->library('form_validation');
          $this->load->library('datatables');
    }

    public function index()
    {
        $data = array(
            'button' => 'Tambah',
            'action' => site_url('prestasi/create_action'),
            'id_prestasi' => set_value('id_prestasi'),
            'tingkat' => set_value('tingkat'),
            'kategori' => set_value('kategori'),
            'juara' => set_value('juara'),
            'skor_prestasi' => set_value('skor_prestasi'),
        );
        $data['title'] = 'Prestasi';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Prestasi' => '',
        ];

        $data['code_js'] = 'prestasi/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'prestasi/Prestasi_list';
        $this->load->view('template/backend', $data);
    }

    public function json() {
        header('Content-Type: application/json');
        echo $this->Prestasi_model->json();
    }

    public function read($id)
    {
        $row = $this->Prestasi_model->get_by_id($id);
        if ($row) {
            $data = array(
                              'id_prestasi' => $row->id_prestasi,
                              'tingkat' => $row->tingkat,
                              'kategori' => $row->kategori,
                              'juara' => $row->juara,
                              'skor_prestasi' => $row->skor_prestasi,
          );

        $data['title'] = 'Prestasi';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];

        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'prestasi/Prestasi_read';
        $this->load->view('template/backend', $data);
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('prestasi'));
        }
    }

    public function create()
    {
        $data = array(
            'button' => 'Tambah',
            'action' => site_url('prestasi/create_action'),
          'id_prestasi' => set_value('id_prestasi'),
          'tingkat' => set_value('tingkat'),
          'kategori' => set_value('kategori'),
          'juara' => set_value('juara'),
          'skor_prestasi' => set_value('skor_prestasi'),
      );

        $data['title'] = 'Prestasi';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];

        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'prestasi/Prestasi_form';
        $this->load->view('template/backend', $data);
    }

    public function create_action()
    {
        $this->_rules();

        if ($this->form_validation->run() == FALSE) {
            $this->create();
        } else {
            $data = array(
                              'tingkat' => $this->input->post('tingkat',TRUE),
                              'kategori' => $this->input->post('kategori',TRUE),
                              'juara' => $this->input->post('juara',TRUE),
                              'skor_prestasi' => $this->input->post('skor_prestasi',TRUE),
          );
            $this->Prestasi_model->insert($data);
            $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');
            helper_log("add", "Menambah data prestasi ".$data['tingkat']);
            redirect(site_url('prestasi'));}
    }

    public function update($id)
    {
        $row = $this->Prestasi_model->get_by_id($id);

        if ($row) {
            $data = array(
                'button' => 'Ubah',
                'action' => site_url('prestasi/update_action'),
                              'id_prestasi' => set_value('id_prestasi', $row->id_prestasi),
                              'tingkat' => set_value('tingkat', $row->tingkat),
                              'kategori' => set_value('kategori', $row->kategori),
                              'juara' => set_value('juara', $row->juara),
                              'skor_prestasi' => set_value('skor_prestasi', $row->skor_prestasi),
                   );

        $data['title'] = 'Prestasi';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];

        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'prestasi/Prestasi_form';
        $this->load->view('template/backend', $data);
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('prestasi'));
        }
    }

    public function update_action()
    {
        $this->_rules();

        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_prestasi', TRUE));
        } else {
            $data = array(
                              'tingkat' => $this->input->post('tingkat',TRUE),
                              'kategori' => $this->input->post('kategori',TRUE),
                              'juara' => $this->input->post('juara',TRUE),
                              'skor_prestasi' => $this->input->post('skor_prestasi',TRUE),
            );
            $this->Prestasi_model->update($this->input->post('id_prestasi', TRUE), $data);
            $this->session->set_flashdata('message', 'Data Berhasil diubah');
            helper_log("edit", "Update data prestasi ".$data['tingkat']);
            redirect(site_url('prestasi'));
        }
    }

    public function delete($id)
    {
        $row = $this->Prestasi_model->get_by_id($id);

        if ($row) {
            $this->Prestasi_model->delete($id);
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus data prestasi ".$row->tingkat);
            redirect(site_url('prestasi'));
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('prestasi'));
        }
    }

    public function deletebulk(){
        $delete = $this->Prestasi_model->deletebulk();
        if($delete){
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus multi data prestasi");
        }else{
            $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
        }
```

```
        echo $delete;
    }

    public function _rules()
    {
      $this->form_validation->set_rules('tingkat', 'tingkat', 'trim|required');
      $this->form_validation->set_rules('kategori', 'kategori', 'trim|required');
      $this->form_validation->set_rules('juara', 'juara', 'trim|required');
      $this->form_validation->set_rules('skor_prestasi', 'skor prestasi',
'trim|required|numeric');

      $this->form_validation->set_rules('id_prestasi', 'id_prestasi', 'trim');
      $this->form_validation->set_error_delimiters('<span class="text-danger">',
'</span>');
    }

    public function excel()
    {
        $this->load->helper('exportexcel');
        $namaFile = "prestasi.xls";
        $judul = "prestasi";
        $tablehead = 0;
        $tablebody = 1;
        $nourut = 1;
        //penulisan header
        header("Pragma: public");
        header("Expires: 0");
        header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
        header("Content-Type: application/force-download");
        header("Content-Type: application/octet-stream");
        header("Content-Type: application/download");
        header("Content-Disposition: attachment;filename=" . $namaFile . "");
        header("Content-Transfer-Encoding: binary ");
    xlsBOF();

    $kolomhead = 0;
    xlsWriteLabel($tablehead, $kolomhead++, "No");
  xlsWriteLabel($tablehead, $kolomhead++, "Tingkat");
  xlsWriteLabel($tablehead, $kolomhead++, "Kategori");
  xlsWriteLabel($tablehead, $kolomhead++, "Peringkat");
  xlsWriteLabel($tablehead, $kolomhead++, "Poin Prestasi");

  foreach ($this->Prestasi_model->get_all() as $data) {
    $kolombody = 0;

    //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
    xlsWriteNumber($tablebody, $kolombody++, $nourut);
      xlsWriteLabel($tablebody, $kolombody++, $data->tingkat);
      xlsWriteLabel($tablebody, $kolombody++, $data->kategori);
      xlsWriteLabel($tablebody, $kolombody++, $data->juara);
      xlsWriteNumber($tablebody, $kolombody++, $data->skor_prestasi);

      $tablebody++;
    $nourut++;
    }

    xlsEOF();
    exit();
}

public function word()
{
    header("Content-type: application/vnd.ms-word");
    header("Content-Disposition: attachment;Filename=prestasi.doc");
```

```
        $data = array(
            'prestasi_data' => $this->Prestasi_model->get_all(),
            'start' => 0
        );
        $this->load->view('prestasi/Prestasi_doc',$data);
    }
    public function printdoc(){
        $data = array(
            'prestasi_data' => $this->Prestasi_model->get_all(),
            'start' => 0
        );
        $this->load->view('prestasi/Prestasi_print', $data);
    }
}

/* End of file Prestasi.php */
```

- Statistik

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Statistik extends CI_Controller {

      public function __construct()
      {
                  parent::__construct();
                  $this->layout->auth();
        $this->load->model('Peserta_model');
        $this->load->model('Pengaturan_model');
      }

      public function index()
```

```
    {
            $data['title'] = 'Statistik';
            $data['subtitle'] = '';
     $data['crumb'] = [
        'Statistik' => '',
     ];
     // $data['code_js'] = 'Statistik/codejs';
     $data['count_jk']=$this->Peserta_model->get_chart_jk();
     $data['count_agama']=$this->Peserta_model->get_chart_agama();
     $data['count_sekolah']=$this->Peserta_model->get_chart_sekolah();
     $data['page'] = 'statistik/Statistik_list';
    $this->load->view('template/backend', $data);
    }
}
```

- Tahun Pelajaran

```
<?php

if (!defined('BASEPATH'))
   exit('No direct script access allowed');

class Tahunpelajaran extends CI_Controller
{
   function __construct()
   {
      parent::__construct();
      $c_url = $this->router->fetch_class();
      $this->layout->auth();
      $this->layout->auth_privilege($c_url);
      $this->load->model('Tahunpelajaran_model');
      $this->load->model('Pengaturan_model');
      $this->load->library('form_validation');
      $this->load->library('datatables');
}

public function index()
{
    $data = array(
        'button' => 'Tambah',
        'action' => site_url('tahunpelajaran/create_action'),
        'id_tahun' => set_value('id_tahun'),
        'tahun_pelajaran' => set_value('tahun_pelajaran'),
        'kuota' => set_value('kuota'),
        'tanggal_mulai_pendaftaran' => set_value('tanggal_mulai_pendaftaran'),
        'tanggal_selesai_pendaftaran' => set_value('tanggal_selesai_pendaftaran'),
        'tanggal_mulai_seleksi' => set_value('tanggal_mulai_seleksi'),
        'tanggal_selesai_seleksi' => set_value('tanggal_selesai_seleksi'),
        'tanggal_pengumuman' => set_value('tanggal_pengumuman'),
        'tanggal_mulai_daftar_ulang' => set_value('tanggal_mulai_daftar_ulang'),
        'tanggal_selesai_daftar_ulang' => set_value('tanggal_selesai_daftar_ulang'),
        'status_tahun' => set_value('status_tahun'),
    );

    $data['title'] = 'Tahun Pelajaran';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Tahunpelajaran' => '',
    ];

    $data['code_js'] = 'tahunpelajaran/codejs';
    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'tahunpelajaran/Tahunpelajaran_list';
    $this->load->view('template/backend', $data);
}
```

```
public function json() {
    header('Content-Type: application/json');
    echo $this->Tahunpelajaran_model->json();
}

public function read($id)
{
    $row = $this->Tahunpelajaran_model->get_by_id($id);
    if ($row) {
    $data = array(
            'id_tahun' => $row->id_tahun,
            'tahun_pelajaran' => $row->tahun_pelajaran,
            'kuota' => $row->kuota,
            'tanggal_mulai_pendaftaran' => $row->tanggal_mulai_pendaftaran,
            'tanggal_selesai_pendaftaran' => $row->tanggal_selesai_pendaftaran,
            'tanggal_mulai_seleksi' => $row->tanggal_mulai_seleksi,
            'tanggal_selesai_seleksi' => $row->tanggal_selesai_seleksi,
            'tanggal_pengumuman' => $row->tanggal_pengumuman,
            'tanggal_mulai_daftar_ulang' => $row->tanggal_mulai_daftar_ulang,
            'tanggal_selesai_daftar_ulang' => $row->tanggal_selesai_daftar_ulang,
            'status_tahun' => $row->status_tahun,
     );

    $data['title'] = 'Tahun Pelajaran';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
    $data['page'] = 'tahunpelajaran/Tahunpelajaran_read';
    $this->load->view('template/backend', $data);
    } else {
        $this->session->set_flashdata('message', 'Data tidak ditemukan');
        redirect(site_url('tahunpelajaran'));
    }
}

public function create()
{
    $data = array(
        'button' => 'Tambah',
        'action' => site_url('tahunpelajaran/create_action'),
      'id_tahun' => set_value('id_tahun'),
      'tahun_pelajaran' => set_value('tahun_pelajaran'),
      'kuota' => set_value('kuota'),
      'tanggal_mulai_pendaftaran' => set_value('tanggal_mulai_pendaftaran'),
      'tanggal_selesai_pendaftaran' => set_value('tanggal_selesai_pendaftaran'),
      'tanggal_mulai_seleksi' => set_value('tanggal_mulai_seleksi'),
      'tanggal_selesai_seleksi' => set_value('tanggal_selesai_seleksi'),
      'tanggal_pengumuman' => set_value('tanggal_pengumuman'),
      'tanggal_mulai_daftar_ulang' => set_value('tanggal_mulai_daftar_ulang'),
      'tanggal_selesai_daftar_ulang' => set_value('tanggal_selesai_daftar_ulang'),
      'status_tahun' => set_value('status_tahun'),
      );

    $data['title'] = 'Tahun Pelajaran';
    $data['subtitle'] = '';
    $data['crumb'] = [
        'Dashboard' => '',
    ];

    $data['code_js'] = 'tahunpelajaran/codejs';
```

```
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'tahunpelajaran/Tahunpelajaran_form';
        $this->load->view('template/backend', $data);
    }

    public function create_action()
    {
        $this->_rules();

        if ($this->form_validation->run() == FALSE) {
            $this->create();
            $this->session->set_flashdata('message',
                form_error('tahun_pelajaran').
                form_error('kuota').
                form_error('tanggal_mulai_pendaftaran').
                form_error('tanggal_selesai_pendaftaran').
                form_error('tanggal_mulai_seleksi').
                form_error('tanggal_selesai_seleksi').
                form_error('pengumuman').
                form_error('tanggal_mulai_daftar_ulang').
                form_error('tanggal_selesai_daftar_ulang').
                form_error('status_tahun')
            );
            redirect(site_url('tahunpelajaran'));
        } else {
        $data = array(
                  'tahun_pelajaran' => $this->input->post('tahun_pelajaran',TRUE),
                  'kuota' => $this->input->post('kuota',TRUE),
            'tanggal_mulai_pendaftaran' => date('Y-m-d', strtotime($this->input-
>post('tanggal_mulai_pendaftaran',TRUE))),
            'tanggal_selesai_pendaftaran' => date('Y-m-d', strtotime($this->input-
>post('tanggal_selesai_pendaftaran',TRUE))),
            'tanggal_mulai_seleksi' => date('Y-m-d', strtotime($this->input->post('tanggal_mulai_seleksi',TRUE))),

            'tanggal_selesai_seleksi' => date('Y-m-d', strtotime($this->input->post('tanggal_selesai_seleksi',TRUE))),

            'tanggal_pengumuman' => date('Y-m-d', strtotime($this->input->post('tanggal_pengumuman',TRUE))),

            'tanggal_mulai_daftar_ulang' => date('Y-m-d', strtotime($this->input->post('tanggal_mulai_daftar_ulang',TRUE))),

            'tanggal_selesai_daftar_ulang' => date('Y-m-d', strtotime($this->input->post('tanggal_selesai_daftar_ulang',TRUE))),

            'status_tahun' => $this->input->post('status_tahun',TRUE),

          );


        $this->Tahunpelajaran_model->insert($data);

        $this->session->set_flashdata('message', 'Data Berhasil ditambahkan');

        helper_log("add", "Menambah data tahun pelajaran ".$data['tahun_pelajaran']);

        redirect(site_url('tahunpelajaran'));

        }

    }


    public function update($id)

    {

        $row = $this->Tahunpelajaran_model->get_by_id($id);


        if ($row) {

        $data = array(

            'button' => 'Ubah',

            'action' => site_url('tahunpelajaran/update_action'),

            'id_tahun' => set_value('id_tahun', $row->id_tahun),

            'tahun_pelajaran' => set_value('tahun_pelajaran', $row->tahun_pelajaran),

            'kuota' => set_value('kuota', $row->kuota),

            'tanggal_mulai_pendaftaran' => set_value('tanggal_mulai_pendaftaran', $row->tanggal_mulai_pendaftaran),
            'tanggal_selesai_pendaftaran' => set_value('tanggal_selesai_pendaftaran', $row->tanggal_selesai_pendaftaran),
            'tanggal_mulai_seleksi' => set_value('tanggal_mulai_seleksi', $row->tanggal_mulai_seleksi),
            'tanggal_selesai_seleksi' => set_value('tanggal_selesai_seleksi', $row->tanggal_selesai_seleksi),
            'tanggal_pengumuman' => set_value('tanggal_pengumuman', $row->tanggal_pengumuman),
            'tanggal_mulai_daftar_ulang' => set_value('tanggal_mulai_daftar_ulang', $row->tanggal_mulai_daftar_ulang),
            'tanggal_selesai_daftar_ulang' => set_value('tanggal_selesai_daftar_ulang', $row->tanggal_selesai_daftar_ulang),
            'status_tahun' => set_value('status_tahun', $row->status_tahun),
        );

        $data['title'] = 'Tahun Pelajaran';
        $data['subtitle'] = '';
        $data['crumb'] = [
            'Dashboard' => '',
        ];

        $data['code_js'] = 'tahunpelajaran/codejs';
        $data['pengaturan']=$this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'tahunpelajaran/Tahunpelajaran_form';
        $this->load->view('template/backend', $data);
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('tahunpelajaran'));
        }
    }

    public function update_action()
    {
        $this->_rules();
        if ($this->form_validation->run() == FALSE) {
            $this->update($this->input->post('id_tahun', TRUE));
        } else {
        $data = array(
                'tahun_pelajaran' => $this->input->post('tahun_pelajaran',TRUE),
                'kuota' => $this->input->post('kuota',TRUE),
            'tanggal_mulai_pendaftaran' => date('Y-m-d', strtotime($this->input->post('tanggal_mulai_pendaftaran',TRUE))),
            'tanggal_selesai_pendaftaran' => date('Y-m-d', strtotime($this->input->post('tanggal_selesai_pendaftaran',TRUE))),
            'tanggal_mulai_seleksi' => date('Y-m-d', strtotime($this->input->post('tanggal_mulai_seleksi',TRUE))),
            'tanggal_selesai_seleksi' => date('Y-m-d', strtotime($this->input->post('tanggal_selesai_seleksi',TRUE))),
            'tanggal_pengumuman' => date('Y-m-d', strtotime($this->input->post('tanggal_pengumuman',TRUE))),
            'tanggal_mulai_daftar_ulang' => date('Y-m-d', strtotime($this->input->post('tanggal_mulai_daftar_ulang',TRUE))),
            'tanggal_selesai_daftar_ulang' => date('Y-m-d', strtotime($this->input->post('tanggal_selesai_daftar_ulang',TRUE))),
            'status_tahun' => $this->input->post('status_tahun',TRUE),
          );

            $this->Tahunpelajaran_model->update($this->input->post('id_tahun', TRUE), $data);
            $this->session->set_flashdata('message', 'Data Berhasil diubah');
            helper_log("edit", "Update data tahun pelajaran ".$data['tahun_pelajaran']);
            redirect(site_url('tahunpelajaran'));
        }
    }

    public function delete($id)
    {
        $row = $this->Tahunpelajaran_model->get_by_id($id);
        if ($row) {
            $this->Tahunpelajaran_model->delete($id);
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus data tahun pelajaran ".$row->tahun_pelajaran);
            redirect(site_url('tahunpelajaran'));
        } else {
            $this->session->set_flashdata('message', 'Data tidak ditemukan');
            redirect(site_url('tahunpelajaran'));
        }
    }

    public function deletebulk(){
        $delete = $this->Tahunpelajaran_model->deletebulk();
        if($delete){
            $this->session->set_flashdata('message', 'Data Berhasil dihapus');
            helper_log("delete", "Menghapus multi data tahun pelajaran");
        }else{
            $this->session->set_flashdata('message_error', 'Data Gagal dihapus');
        }
        echo $delete;
    }

    public function _rules()
    {
      $this->form_validation->set_rules('tahun_pelajaran', 'tahun pelajaran',
'trim|required|numeric|exact_length[4]',
        array(
                'required'      => 'Tahun Pelajaran tidak boleh kosong ',
                'numeric'       => 'Tahun Pelajaran hanya angka ',
                'exact_length[4]' => 'Tahun Pelajaran '
        ));
    $this->form_validation->set_rules('kuota', 'kuota', 'trim|required|numeric',
     array(
          'required'     => 'Kuota tidak boleh kosong ',
          'numeric'      => 'Kuota hanya angka '
     ));
    $this->form_validation->set_rules('tanggal_mulai_pendaftaran', 'tanggal mulai
pendaftaran', 'trim|required',
     array(
          'required'     => 'Mulai Pendafataran tidak boleh kosong '
     ));
    $this->form_validation->set_rules('tanggal_selesai_pendaftaran', 'tanggal selesai
pendaftaran', 'trim|required',
     array(
          'required'     => 'Selesai Pendafataran tidak boleh kosong '
     ));
    $this->form_validation->set_rules('tanggal_mulai_seleksi', 'tanggal mulai seleksi',
'trim|required',
     array(
          'required'     => 'Mulai Seleksi tidak boleh kosong '
     ));
    $this->form_validation->set_rules('tanggal_selesai_seleksi', 'tanggal selesai seleksi',
'trim|required',
     array(
          'required'     => 'Selesai tidak boleh kosong '
     ));
    $this->form_validation->set_rules('tanggal_pengumuman', 'tanggal pengumuman',
'trim|required',
     array(
          'required'     => 'Tanggal Pengumuman tidak boleh kosong '
     ));
    $this->form_validation->set_rules('tanggal_mulai_daftar_ulang', 'tanggal mulai daftar
ulang', 'trim|required',
     array(
          'required'     => 'Mulai Daftar Ulang tidak boleh kosong '
```

```
        ));
      $this->form_validation->set_rules('tanggal_selesai_daftar_ulang', 'tanggal selesai daftar ulang', 'trim|required',
        array(
            'required'      => 'Selesai Daftar Ulang tidak boleh kosong '
        ));
      $this->form_validation->set_rules('status_tahun', 'status tahun', 'trim|required',
        array(
            'required'      => 'Status Tahun tidak boleh kosong '
        ));

      $this->form_validation->set_rules('id_tahun', 'id_tahun', 'trim');
      $this->form_validation->set_error_delimiters('<span class="text-danger">', '</span>');
    }

    public function excel()
    {
        $this->load->helper('exportexcel');
        $namaFile = "tahunpelajaran.xls";
        $judul = "tahunpelajaran";
        $tablehead = 0;
        $tablebody = 1;
        $nourut = 1;
        //penulisan header
        header("Pragma: public");
        header("Expires: 0");
        header("Cache-Control: must-revalidate, post-check=0,pre-check=0");
        header("Content-Type: application/force-download");
        header("Content-Type: application/octet-stream");
        header("Content-Type: application/download");
        header("Content-Disposition: attachment;filename=" . $namaFile . "");
        header("Content-Transfer-Encoding: binary ");
```

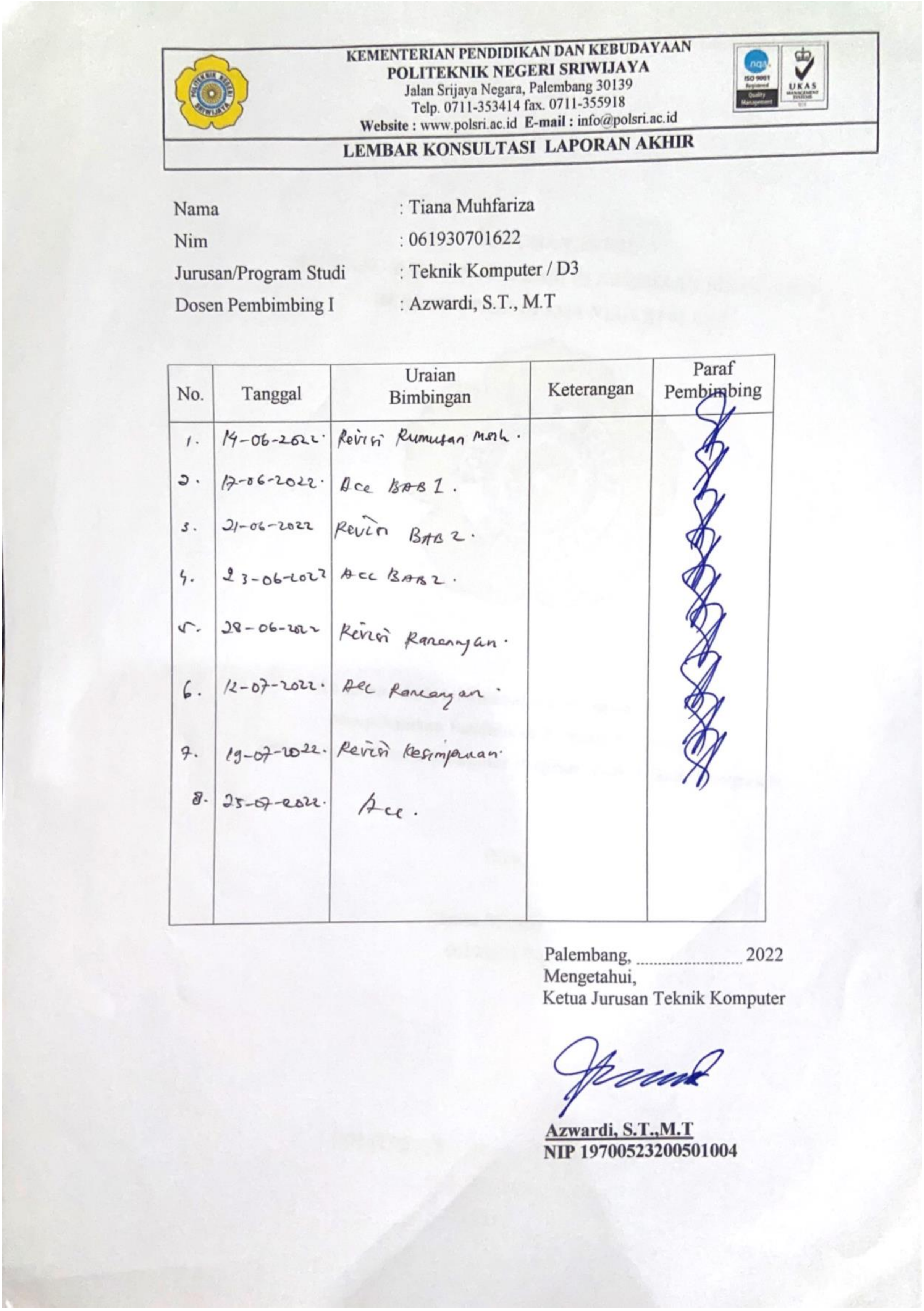
```
 xlsBOF();

 $kolomhead = 0;
 xlsWriteLabel($tablehead, $kolomhead++, "No");
xlsWriteLabel($tablehead, $kolomhead++, "Tahun Pelajaran");
xlsWriteLabel($tablehead, $kolomhead++, "Kuota");
xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Mulai Pendaftaran");
xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Selesai Pendaftaran");
xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Mulai Seleksi");
xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Selesai Seleksi");
xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Pengumuman");
xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Mulai Daftar Ulang");
xlsWriteLabel($tablehead, $kolomhead++, "Tanggal Selesai Daftar Ulang");
xlsWriteLabel($tablehead, $kolomhead++, "Status Tahun");

foreach ($this->Tahunpelajaran_model->get_all() as $data) {
 $kolombody = 0;

 //ubah xlsWriteLabel menjadi xlsWriteNumber untuk kolom numeric
 xlsWriteNumber($tablebody, $kolombody++, $nourut);
  xlsWriteLabel($tablebody, $kolombody++, $data->tahun_pelajaran);
  xlsWriteLabel($tablebody, $kolombody++, $data->kuota);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_mulai_pendaftaran);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_selesai_pendaftaran);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_mulai_seleksi);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_selesai_seleksi);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_pengumuman);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_mulai_daftar_ulang);
  xlsWriteLabel($tablebody, $kolombody++, $data->tanggal_selesai_daftar_ulang);
  xlsWriteLabel($tablebody, $kolombody++, $data->status_tahun);
```

```
         $tablebody++;
          $nourut++;
        }

        xlsEOF();
        exit();
    }

    public function word()
    {
        header("Content-type: application/vnd.ms-word");
        header("Content-Disposition: attachment;Filename=tahunpelajaran.doc");

        $data = array(
            'tahunpelajaran_data' => $this->Tahunpelajaran_model->get_all(),
            'start' => 0
        );
        $this->load->view('tahunpelajaran/Tahunpelajaran_doc',$data);
    }

    public function printdoc(){
        $data = array(
            'tahunpelajaran_data' => $this->Tahunpelajaran_model->get_all(),
            'start' => 0
        );
        $this->load->view('tahunpelajaran/Tahunpelajaran_print', $data);
    }
}
/* End of file Tahunpelajaran.php */
```

- Profil

```
<?php
```

```
if (!defined('BASEPATH'))
    exit('No direct script access allowed');

class Profile extends CI_Controller {

      public function __construct()
      {
        parent::__construct();
        $this->layout->auth();

        //add
        $this->load->database();
        $this->load->library(array('ion_auth', 'form_validation'));
        $this->load->helper(array('url', 'language'));
        $this->load->model('Pengaturan_model');

        $this->lang->load('auth');
        //end add
      }

      public function index()
      {
                  $data['title'] = 'Profile';
                  $data['subtitle'] = '';
        $data['crumb'] = [
            'Profile' => '',
        ];
        //$this->layout->set_privilege(1);
        $data['pengaturan'] = $this->Pengaturan_model->get_by_id_1();
        $data['page'] = 'profile';
                  $this->load->view('template/backend', $data);
```



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 fax. 0711-355918
Website : www.polsri.ac.id **E-mail :** info@polsri.ac.id

LEMBAR KONSULTASI LAPORAN AKHIR

Nama	: Tiana Muhfariza
Nim	: 061930701622
Jurusan/Program Studi	: Teknik Komputer / D3
Dosen Pembimbing I	: Azwardi, S.T., M.T

No.	Tanggal	Uraian Bimbingan	Keterangan	Paraf Pembimbing
1.	14-06-2022.	Revisi Rumusan Mslh.		
2.	17-06-2022.	Acc BAB 1.		
3.	21-06-2022	Revisi BAB 2.		
4.	23-06-2022	Acc BAB 2.		
5.	28-06-2022	Revisi Rancangan.		
6.	12-07-2022.	Acc Rancangan.		
7.	19-07-2022.	Revisi Kesimpulan.		
8.	25-07-2022.	Acc.		

Palembang, ... 2022
Mengetahui,
Ketua Jurusan Teknik Komputer

Azwardi, S.T.,M.T
NIP 197005232005010004

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 fax. 0711-355918
Website : www.polsri.ac.id E-mail : info@polsri.ac.id

LEMBAR KONSULTASI LAPORAN AKHIR

Nama	: Tiana Muhfariza
Nim	: 061930701622
Jurusan/Program Studi	: Teknik Komputer / D3
Dosen Pembimbing II	: Isnainy Azro, S.Kom, M.Kom

No.	Tanggal	Uraian Bimbingan	Keterangan	Paraf Pembimbing
1.	16-06-2022	Revisi Bab 1.		
2.	20-06-2022	Acc Bab 1.		
3.	23-06-2022.	Revisi Bab 2.		
4.	27-06-2022	Acc Bab 2.		
5.	04-07-2022.	Revisi Flowchart.		
6.	11-07-2022.	Acc flowchart.		
7.	18-07-2022.	Revisi Rancangan.		
8.	22-07-2022.	Acc.		

Palembang, ... 2022
Mengetahui,
Ketua Jurusan Teknik Komputer

Azwardi, S.T.,M.T
NIP 197005232005011004

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 fax. 0711-355918
Website : www.polsri.ac.id **E-mail :** info@polsri.ac.id

REKOMENDASI UJIAN LAPORAN AKHIR (LA)

Pembimbing Laporan Akhir memberikan rekomendasi kepada,

Nama	:	Tiana Muhfariza
NIM	:	061930701622
Jurusan/Program Studi	:	Teknik Komputer/ D III Teknik Komputer
Judul Laporan Akhir	:	Rancang Bangun Aplikasi Penerimaan Siswa Baru Berbasis Web di SMA N 01 OKU

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Laporan Akhir (LA) pada Tahun Akademik 2022.

Palembang, Juli 2021

Pembimbing I,

Azwardi, S.T.,M.T

NIP 19700523200501004

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
POLITEKNIK NEGERI SRIWIJAYA
Jalan Srijaya Negara, Palembang 30139
Telp. 0711-353414 fax. 0711-355918
Website : www.polsri.ac.id **E-mail :** info@polsri.ac.id

REKOMENDASI UJIAN LAPORAN AKHIR (LA)

Pembimbing Laporan Akhir memberikan rekomendasi kepada,

Nama : Tiana Muhfariza

NIM : 061930701622

Jurusan/Program Studi : Teknik Komputer/ D III Teknik Komputer

Judul Laporan Akhir : Rancang Bangun Aplikasi Penerimaan Siswa Baru Berbasis Web di SMA N 01 OKU

Mahasiswa tersebut telah memenuhi persyaratan dan dapat mengikuti Ujian Laporan Akhir (LA) pada Tahun Akademik 2022.

Palembang, Juli 2021

Pembimbing II,

Isnainy Azro, S.Kom
NIP. 197310012002122007

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN, RISET DAN TEKNOLOGI **POLITEKNIK NEGERI SRIWIJAYA** Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji : Yulian Mirza,ST.,M.Kom

Nama Mahasiswa : Tiana Muhfariza

NIM : 061930701622

Jurusan /Program Studi : Teknik Komputer/ DIII Teknik Komputer

Judul LA/ Skripsi : Rancang Bangun Aplikasi Penerimaan Siswa Baru Berbasis Web di SMA N 01 OKU

No	Uraian Revisi	Paraf
	Tata tulis	

Palembang, ………………………………

Dosen Penguji,

(………………………………)

No. Dok. :	Tgl. Berlaku : KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN, RISET DAN TEKNOLOGI **POLITEKNIK NEGERI SRIWIJAYA** Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	No. Rev. :
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji : Meyi Darlies, M.Kom

Nama Mahasiswa : Tiana Muhfariza

NIM : 061930701622

Jurusan /Program Studi : Teknik Komputer/ DIII Teknik Komputer

Judul LA/ Skripsi : Rancang Bangun Aplikasi Penerimaan Siswa Baru Berbasis Web di SMA N 01 OKU

No	Uraian Revisi	Paraf
1.	✓ Latar Belakang tambahan pola penyelesaian Permasalahan (Jabarkan)	
2	✓ Sebaiknya di tambahkan Arsitektur aplikasi yg dibuat.	
3	✓ Perancangan ~~Ada~~ (Beberapa) tidak sesuai dengan hasil.	
4.	Tambahkan pembahasan program	
5	✓ Perbaiki Rumusan Masalah, Tujuan, dan Kesimpulan	

Palembang, 28 July 2022.
Dosen Penguji,

(Meiyi Darlies, S.Kom, M.Kom.)

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN, RISET DAN TEKNOLOGI **POLITEKNIK NEGERI SRIWIJAYA** Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji	:	Alan Novi Tompunu,S.T., M.T
Nama Mahasiswa	:	Tiana Muhfariza
NIM	:	061930701622
Jurusan /Program Studi	:	Teknik Komputer/ DIII Teknik Komputer
Judul LA/ Skripsi	:	Rancang Bangun Aplikasi Penerimaan Siswa Baru Berbasis Web di SMA N 01 OKU

No	Uraian Revisi	Paraf
1.	latar belakang	
2.	Hasil & pembahasan	
3.	kesimpulan & saran	
4.	Surat pernyataan dari SMAN 1 OKU tentang Aplikasi Web yg dibuat.	
	alvito.porni@gmail.com. word.	

Palembang, ……………………………

Dosen Penguji,

(Alan N.T. ………)

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN, RISET DAN TEKNOLOGI **POLITEKNIK NEGERI SRIWIJAYA** Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji	:	Hartati Deviana,S.T., M.Kom
Nama Mahasiswa	:	Tiana Muhfariza
NIM	:	061930701622
Jurusan /Program Studi	:	Teknik Komputer/ DIII Teknik Komputer
Judul LA/ Skripsi	:	Rancang Bangun Aplikasi Penerimaan Siswa Baru Berbasis Web di SMA N 01 OKU

No	Uraian Revisi	Paraf
-	Perbaiki bab Pendahuluan sesuai kondisi tempat.	
-	Perbaiki kesimpulan	

Palembang, ……………………………
Dosen Penguji,

(…………………………………)

No. Dok. :	Tgl. Berlaku :	No. Rev. :
	KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN, RISET DAN TEKNOLOGI **POLITEKNIK NEGERI SRIWIJAYA** Jalan Srijaya Negara, Palembang 30139. Telp. 0711-353414 **Website :** www.polsri.ac.id **E-mail :** info@polsri.ac.id	
	REVISI TUGAS AKHIR (TA)	

Dosen Penguji	:	Rian Rahmanda Putra, M.Kom
Nama Mahasiswa	:	Tiana Muhfariza
NIM	:	061930701622
Jurusan /Program Studi	:	Teknik Komputer/ DIII Teknik Komputer
Judul LA/ Skripsi	:	Rancang Bangun Aplikasi Penerimaan Siswa Baru Berbasis Web di SMA N 01 OKU

No	Uraian Revisi	Paraf
1	Jelaskan pada bab I bagian latar belakang bagaimana alur penerimaan mahasiswa baru saat ini dan aplikasi yang kamu buat apa dampak yang diberikannya.	
2	Buat Arsitektur aplikasi yang kamu buat ini. pada bab III.	
3.	Daptar pustaka harus sesuai dengan kutipan.	

Palembang, ……………………………

Dosen Penguji,

(RIAN RAHMANDA PUTRA, M.KOM.)

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI SRIWIJAYA
JalanSrijaya Negara, Palembang 30139
Telp. 0711-353414 Fax. 0711-355918
Website : www.polisriwijaya.ac.id E-mail :info@polsri.ac.id

PELAKSANAAN REVISI LAPORAN AKHIR (LA)

Mahasiswa berikut,

Nama	:	Tiana Muhfariza
NIM	:	061930701622
Jurusan/Program Studi	:	Teknik Komputer/DIII-Teknik Komputer
Judul Laporan KP	:	Aplikasi Perpustakaan Berbasis Web di SMP Negeri 54 Palembang

Telah melaksanakan revisi terhadap Laporan Akhir (LA) yang diseminarkan pada hari .... tanggal .... bulan .... tahun .... Pelaksanaan revisi terhadap Laporan Akhir tersebut telah disetujui oleh Dosen Penilai yang memberikan revisi:

No.	Komentar	Nama Dosen Penilai *)	Tanggal	Tanda Tangan
1	Acc	Yulian Mirza,S.T.,M.Kom	16/8 2022.	
2	Acc	Meiyi Darlies,M.Kom	18/8 2022.	
3	Acc	Alan Novi Tompunu,S.T.,M.T	24/8 22	
4	Acc	Hartati Deviana,S.T.,M.Kom	21/8 22.	
5	Acc.	Rian Rahmanda Putra,M.Kom	18/8 22	

Palembang, ...

Ketua Penilai **)

Yulian Mirza, ST., M.Kom

NIP 196607121990031003