

# LAMPIRAN

## Lampiran 1 Surat Kesiediaan Dosen Pembimbing 1

### SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama : Mohammad Humam, M.Kom  
NIDN : 0618117901  
NIPY : 12.002.007  
Jabatan Struktural : Dosen Tetap  
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir mahasiswa berikut :

Nama : Irvan Aldi Susanto  
NIM : 21041033  
Program Studi : DIII – Teknik Komputer  
Judul TA : SISTEM PENDAFTARAN PELAYANAN KB  
DI KLINIK IBI TEGAL BERBASIS WEBSITE

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Tegal, 04 Maret 2023

Mengetahui  
Ka. Prodi DIII Teknik Komputer

Dosen Pembimbing I,



Ida Afrakana, ST, M.Kom  
NIPY 12.013.168

Mohammad Humam, M.Kom  
NIPY 12.002.007

## Lampiran 2 Surat Kesiediaan Dosen Pembimbing 2

### SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini :

Nama : Yerry Sabanise, M.Kom  
NIDN : 0613028602  
NIPY : 03.012.110  
Jabatan Struktural : Dosen Tetap  
Jabatan Fungsional : Asisten Ahli

Dengan ini menyatakan bersedia untuk menjadi pembimbing II pada Tugas Akhir mahasiswa berikut :

Nama : Irvan Aldi Susanto  
NIM : 21041033  
Program Studi : DIII – Teknik Komputer  
Judul TA : SISTEM PENDAFTARAN PELAYANAN KB  
DI KLINIK IBI TEGAL BERBASIS WEBSITE

Demikian pernyataan ini dibuat agar dapat dilaksanakan sebagaimana mestinya.

Mengetahui  
Ka. Prodi DIII Teknik Komputer



Ira Afriliana, ST, M.Kom  
NIPY 12.013.168

Tegal, 08 Mei 2024

Dosen Pembimbing II,

Yerry Sabanise, M.Kom  
NIPY 12.002.007

### Lampiran 3 Surat Selesai Observasi



## IKATAN BIDAN INDONESIA KOTA TEGAL

Jl. Nakula No.1, Slerok, Kec. Tegal Timur Kota Tegal 52124

### SURAT KETERANGAN SELESAI OBSERVASI

Yang bertanda tangan dibawah ini :

Nama : TARYULI, S.Tr  
Jabatan : Ketua IBI Tegal  
Alamat : Jl. Serayu Gg Jatimulyo 1 Kec Tegal Timur. Kota Tegal

Nama : IRVAN ALDI SUSANTO  
NIM : 21041033  
Program Studi : DIII Teknik Komputer

Telah selesai melakukan observasi di Klinik IBI Tegal Selama 1 (satu) minggu, terhitung mulai tanggal 27 Mei 2024 sampai dengan 31 Mei 2024 untuk memperoleh data dalam rangka penyusunan Tugas Akhir yang berjudul "**SISTEM PENDAFTARAN PELAYANAN KB DI KLINIK IBI TEGAL**".

Demikian surat keterangan ini dibuat dan diberikan kepada yang bersangkutan untuk dipergunakan seperlunya.

Tegal, 01 Juni 2024

Ketua IBI Kota Tegal



(taryuli, S.Tr)

## **Lampiran 4 Source Code *Dashboard***

### **Source code application/controllers/*Dashboard***

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

class Dashboard extends CI_Controller {

    public function __construct() {
        parent::__construct();
        check_not_login(); // Load model yang akan digunakan
    }

    public function index()
    {
        $data['title'] = 'Dashboard';
        $this->template->load('template','dashboard', $data);
    }

}
```

## Lampiran 5 Source Code Pelayanan

### Source code application/controllers/Pelayanan

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

use Twilio\Rest\Client;
class Pelayanan extends CI_Controller {

    public function __construct() {
        parent::__construct();
        check_not_login();
        $this->load->model(['pelayanan_m', 'kota_m',
'kecamatan_m', 'kelurahan_m', 'kader_m']); // Load model yang akan
digunakan
    }

    public function index()
    {
        $data['title'] = 'Pelayanan';
        $data['py'] = $this->pelayanan_m->get();
        $this->template->load('template', 'form/Pelayanan',
$data);
    }

    public function tambah()
    {
        $data['title'] = 'Input Pelayanan';
        $data['kota'] = $this->kota_m->getkota();
        $id_kota = (!empty($data['kota'])) ? $data['kota'][0]-
>id_kota : null;

        if ($id_kota !== null) {
            $data['kecamatan'] = $this->kecamatan_m-
>getdatakec($id_kota);
        } else {
            $data['kecamatan'] = array();
        }

        $id_kecamatan = (!empty($data['kecamatan'])) ?
$data['kecamatan'][0]->id_kecamatan : null;

        if ($id_kecamatan !== null) {
            $data['kelurahan'] = $this->kelurahan_m-
>getdatakel($id_kecamatan);
        } else {
            $data['kelurahan'] = array();
        }

        $data['kader'] = $this->kader_m->get_all_kader();

        $this->template->load('template', 'form/input_Pelayanan',
$data);
    }
}
```

```

public function tambahdata()
{
    $this->load->library('form_validation');

    // Set rules for form validation
    $this->form_validation->set_rules('namkader',      'Nama
Kader', 'required');
    $this->form_validation->set_rules('nampes',      'Nama
Peserta', 'required');
    $this->form_validation->set_rules('nohp',      'No   HP',
'required');
    $this->form_validation->set_rules('nik',      'NIK',
'required|exact_length[16]',
    array(
        'required' => 'NIK harus diisi.',
        'exact_length' => 'NIK harus terdiri dari 16 angka.'
    ));
    $this->form_validation->set_rules('nasuami',      'Nama
Suami', 'required');
    $this->form_validation->set_rules('alamat',      'Alamat',
'required');
    $this->form_validation->set_rules('kota',      'Kota',
'required');
    $this->form_validation->set_rules('kec',      'Kecamatan',
'required');
    $this->form_validation->set_rules('kel',      'Kelurahan',
'required');
    $this->form_validation->set_rules('tkun',      'Tanggal
Dikunjungi', 'required');
    $this->form_validation->set_rules('tlahir',      'Tanggal
Lahir', 'required');
    $this->form_validation->set_rules('temlahir',      'Tempat
Lahir', 'required');
    $this->form_validation->set_rules('usia',      'Usia',
'required');
    $this->form_validation->set_rules('asuransi',
'Asuransi', 'required');
    $this->form_validation->set_rules('alat',      'Alat',
'required');
    $this->form_validation->set_rules('tdil',      'Tanggal
Dilayani', 'required');
    $this->form_validation->set_rules('tdic',      'Tanggal
Dicabut', 'required');
    $this->form_validation->set_rules('pensuami',
'Pendidikan Suami', 'required');
    $this->form_validation->set_rules('penistri',
'Pendidikan Istri', 'required');
    $this->form_validation->set_rules('peksuami',
'Pekerjaan Suami', 'required');
    $this->form_validation->set_rules('pekistri',
'Pekerjaan Istri', 'required');
}

```

## Source Code application/controllers/Pelayanan

```
if ($this->form_validation->run() == TRUE) {
    // Manipulasi nomor HP untuk menambahkan kode
    negara
        $nohp = $this->input->post('nohp');
        $full_nohp = '62' . ltrim($nohp, '0'); //
    Tambahkan kode negara 62 dan hilangkan angka 0 di depan jika ada

    // Siapkan data untuk disimpan ke database
    $data = array(
        'id_kader' => $this->input-
>post('namkader'),
        'nama_pesKB' => $this->input-
>post('nampes'),
        'nik' => $this->input->post('nik'),
        'nohp' => $full_nohp,
        'nama_suami' => $this->input-
>post('nasuami'),
        'alamat_pes' => $this->input-
>post('alamat'),
        'id_kota' => $this->input->post('kota'),
        'id_kecamatan' => $this->input-
>post('kec'),
        'id_kelurahan' => $this->input-
>post('kel'),
        'tanggal_kunjungan' => $this->input-
>post('tkun'),
        'tanggal_lahir' => $this->input-
>post('tlahir'),
        'tempat_lahir' => $this->input-
>post('temlahir'),
        'usia' => $this->input->post('usia'),
        'asuransi' => $this->input-
>post('asuransi'),
        'alat' => $this->input->post('alat'),
        'tanggal_dilayani' => $this->input-
>post('tdil'),
        'tanggal_dicabut' => $this->input-
>post('tdic'),
        'pen_suami' => $this->input-
>post('pensuami'),
        'pen_istri' => $this->input-
>post('penistri'),
        'pek_suami' => $this->input-
>post('peksuami'),
        'pek_istri' => $this->input-
>post('pekistri'),
    );

    // Panggil model untuk menyimpan data ke database
    $this->load->model('pelayanan_m');
    $inserted_id = $this->pelayanan_m-
>insert($data);
    if ($inserted_id) {
```

## Source Code application/controllers/pelayanan



```

        $this->session->set_flashdata('success',      'Data
Berhasil Ditambahkan');
        redirect('pelayanan', 'refresh');
    } else {
        // Penanganan jika ada masalah saat
penyisipan data
        $this->session->set_flashdata('error',
'Terjadi kesalahan saat menambahkan data.');
```

```

        redirect('pelayanan', 'refresh');
    }
    } else {
        // Memuat ulang tampilan dengan data yang
diperlukan jika validasi gagal
        $this->session->set_flashdata('error',
validation_errors());
        $this->tambah();
    }
}

// public function tambahdata()
// {
//     $this->load->library('form_validation');

//     // Set rules for form validation
//     $this->form_validation->set_rules('namkader', 'Nama
Kader', 'required');
//     $this->form_validation->set_rules('nampes', 'Nama
Peserta', 'required');
//     $this->form_validation->set_rules('nohp', 'No HP',
'required');
//     $this->form_validation->set_rules('nik', 'NIK',
'required|exact_length[16]',
//     array(
//         'required' => 'NIK harus diisi.',
//         'exact_length' => 'NIK harus terdiri dari 16 angka.'
//     ));
//     $this->form_validation->set_rules('nasuami', 'Nama
Suami', 'required');
//     $this->form_validation->set_rules('alamat', 'Alamat',
'required');
//     $this->form_validation->set_rules('kota', 'Kota',
'required');
//     $this->form_validation->set_rules('kec', 'Kecamatan',
'required');
//     $this->form_validation->set_rules('kel', 'Kelurahan',
'required');
//     $this->form_validation->set_rules('tkun', 'Tanggal
Dikunjungi', 'required');
//     $this->form_validation->set_rules('tlahir', 'Tanggal
Lahir', 'required');
//     $this->form_validation->set_rules('temlahir', 'Tempat
Lahir', 'required');
```

Source Code applications/controllers/pelayanan

```

        //          $this->form_validation->set_rules('usia', 'Usia',
'required');
        //          $this->form_validation->set_rules('asuransi',
'Asuransi', 'required');
        //          $this->form_validation->set_rules('alat', 'Alat',
'required');
        //          $this->form_validation->set_rules('tdil', 'Tanggal
Dilayani', 'required');
        //          $this->form_validation->set_rules('tdic', 'Tanggal
Dicabut', 'required');
        //          $this->form_validation->set_rules('pensuami',
'Pendidikan Suami', 'required');
        //          $this->form_validation->set_rules('penistri',
'Pendidikan Istri', 'required');
        //          $this->form_validation->set_rules('peksuami', 'Pekerjaan
Suami', 'required');
        //          $this->form_validation->set_rules('pekistri', 'Pekerjaan
Istri', 'required');

        //          if ($this->form_validation->run() == TRUE) {
        //              // Manipulasi nomor HP untuk menambahkan kode negara
        //              $nohp = $this->input->post('nohp');
        //              $full_nohp = '62' . ltrim($nohp, '0'); // Tambahkan
kode negara 62 dan hilangkan angka 0 di depan jika ada

        //              // Siapkan data untuk disimpan ke database
        //              $data = array(
        //                  'id_kader' => $this->input->post('namkader'),
        //                  'nama_pesKB' => $this->input->post('nampes'),
        //                  'nik' => $this->input->post('nik'),
        //                  'nohp' => $full_nohp,
        //                  'nama_suami' => $this->input->post('nasuami'),
        //                  'alamat_pes' => $this->input->post('alamat'),
        //                  'id_kota' => $this->input->post('kota'),
        //                  'id_kecamatan' => $this->input->post('kec'),
        //                  'id_kelurahan' => $this->input->post('kel'),
        //                  'tanggal_kunjungan' => $this->input->post('tkun'),
        //                  'tanggal_lahir' => $this->input->post('t_lahir'),
        //                  'tempat_lahir' => $this->input->post('t_lahir'),
        //                  'usia' => $this->input->post('usia'),
        //                  'asuransi' => $this->input->post('asuransi'),
        //                  'alat' => $this->input->post('alat'),
        //                  'tanggal_dilayani' => $this->input->post('tdil'),
        //                  'tanggal_dicabut' => $this->input->post('tdic'),
        //                  'pen_suami' => $this->input->post('pensuami'),
        //                  'pen_istri' => $this->input->post('penistri'),
        //                  'pek_suami' => $this->input->post('peksuami'),
        //                  'pek_istri' => $this->input->post('pekistri'),
        //              );

        //              // Panggil model untuk menyimpan data ke database
        //              $inserted_id = $this->pelayanan_m->insert($data);

        //              if ($inserted_id) {

```

Source code application/controllers/pelayanan

```

//          // Siapkan pesan untuk dikirim ke WhatsApp
//          $message = "Pesan ini dibuat oleh IBI Tegal \n";
//          $message .= "Nama Peserta IBI Tegal: " .
$data['nama_pesKB'] . "\n";
//          $message .= "Nama Suami Peserta IBI Tegal:
" . $data['nama_suami'] . "\n";
//          $message .= "Nama Alat KB: " . $data['alat'] .
"\n";
//          $message .= "Tanggal Dilayani: " .
$data['tanggal_dilayani'] . "\n";
//          $message .= "Tanggal Dicabut: " .
$data['tanggal_dicabut'] . "\n";
//          $message .= "Terima kasih telah berkunjung di
IBI Tegal \n";
//          $message .= "Salam 2 Anak Cukup \n";

//          // Kirim pesan menggunakan Twilio API
//          $this->send_whatsapp_message($full_nohp,
$message);

//          $this->session->set_flashdata('success', 'Data
Berhasil Ditambahkan dan pesan WhatsApp telah dikirim');
//          redirect('pelayanan', 'refresh');
//          } else {
//          // Penanganan jika ada masalah saat penyisipan
data
//          $this->session->set_flashdata('error', 'Terjadi
kesalahan saat menambahkan data. ');
//          redirect('pelayanan', 'refresh');
//          }
//          } else {
//          // Memuat ulang tampilan dengan data yang diperlukan
jika validasi gagal
//          $this->session->set_flashdata('error',
validation_errors());
//          $this->tambah();
//          }
//          }

///akhir tambah data

///awal edit data

public function edit()
{
    $data['title'] = 'Edit';
    $data['kota'] = $this->kota_m->getkota();
    $id_kota = (!empty($data['kota'])) ? $data['kota'][0]-
>id_kota : null;

    if ($id_kota !== null) {
        $data['kecamatan'] = $this->kecamatan_m-
>getdatakec($id_kota);

```

Source code application/controllers/pelayanan

```

    } else {
        $data['kecamatan'] = array();
    }

    $id_kecamatan = (!empty($data['kecamatan'])) ?
    $data['kecamatan'][0]->id_kecamatan : null;

    if ($id_kecamatan !== null) {
        $data['kelurahan'] = $this->kelurahan_m-
>getdatakel($id_kecamatan);
    } else {
        $data['kelurahan'] = array();
    }

    $this->template->load('template', 'form/edit_Pelayanan',
    $data);
}

public function editdata()
{
    $this->load->library('form_validation');

    $this->form_validation->set_rules('nampes', 'Nama
Peserta', 'required');
    $this->form_validation->set_rules('nik', 'NIK',
'required');
    $this->form_validation->set_rules('nasuami', 'Nama
Suami', 'required');
    $this->form_validation->set_rules('alamat', 'Alamat',
'required');
    //$this->form_validation->set_rules('kota', 'Kota',
'required');
    //$this->form_validation->set_rules('kec',
'Kecamatan', 'required');
    //$this->form_validation->set_rules('kel',
'Kelurahan', 'required');
    $this->form_validation->set_rules('tkun', 'Tanggal
Dikunjungi', 'required');
    $this->form_validation->set_rules('tlahir', 'Tanggal
Lahir', 'required');
    $this->form_validation->set_rules('temlahir', 'Tempat
Lahir', 'required');
    $this->form_validation->set_rules('usia', 'Usia',
'required');
    $this->form_validation->set_rules('asuransi',
'Asuransi', 'required');
    $this->form_validation->set_rules('alat', 'Alat',
'required');
    $this->form_validation->set_rules('tdil', 'Tanggal
Dilayani', 'required');
    $this->form_validation->set_rules('tdic', 'Tanggal
Dicabut', 'required');
}

```

Source code application/controllers/pelayanan

```

        $this->form_validation->set_rules('pensuami',
'Pendidikan Suami', 'required');
        $this->form_validation->set_rules('penistri',
'Pendidikan Istri', 'required');
        $this->form_validation->set_rules('peksuami',
'Pekerjaan Suami', 'required');
        $this->form_validation->set_rules('pekistri',
'Pekerjaan Istri', 'required');

        if ($this->form_validation->run() == TRUE) {
            $id = $this->input->post('id');

            if (!empty($id)) {
                // Proses edit data
                $data = array(
                    'nama_pesKB' => $this->input-
>post('nampes'), 'nik' => $this->input->post('nik'),
                    'nama_suami' => $this->input-
>post('nasuami'),
                    'alamat_pes' => $this->input-
>post('alamat'),
                    //'id_kota' => $this->input-
>post('kota'),
                    //'id_kecamatan' => $this->input-
>post('kec'),
                    //'id_kelurahan' => $this->input-
>post('kel'),
                    'tanggal_kunjungan' => $this->input-
>post('tkun'),
                    'tanggal_lahir' => $this->input-
>post('tlahir'),
                    'tempat_lahir' => $this->input-
>post('temlahir'),
                    'usia' => $this->input-
>post('usia'),
                    'asuransi' => $this->input-
>post('asuransi'),
                    'alat' => $this->input-
>post('alat'),
                    'tanggal_dilayani' => $this->input-
>post('tdil'),
                    'tanggal_dicabut' => $this->input-
>post('tdic'),
                    'pen_suami' => $this->input-
>post('pensuami'),
                    'pen_istri' => $this->input-
>post('penistri'),
                    'pek_suami' => $this->input-
>post('peksuami'),
                    'pek_istri' => $this->input-
>post('pekistri'),
                );
            }
        }
    }
}

```

Source code application/controllers/pelayanan

```

        $result = $this->pelayanan_m->update($id,
$data);

        if ($result) {
            $this->session-
>set_flashdata('success', 'Data Berhasil Diupdate');
        } else {
            $this->session-
>set_flashdata('error', 'Terjadi kesalahan saat mengupdate data. ');
        }

        redirect('pelayanan', 'refresh');
    } else {
        // ID kosong, mungkin ada kesalahan
        $this->session->set_flashdata('error', 'ID
data tidak valid. ');
        redirect('pelayanan', 'refresh');
    }
} else {
    // Validasi tidak berhasil, kembali ke halaman
edit
    $id = $this->input->post('id');
    $data['title'] = 'Input Pelayanan';
    //$data['kota'] = $this->kota_m->getkota();
    //$data['kecamatan'] = $this->kecamatan_m-
>getdatakec($this->input->post('kota'));
    //$data['kelurahan'] = $this->kelurahan_m-
>getdatakel($this->input->post('kec'));

    $this->template->load('template',
'form/pelayanan', $data, $id);
}

function getkecamatan() {
    $id_kota = $this->input->post('kota');
    $getdatakec = $this->kecamatan_m->getdatakec($id_kota);
    echo json_encode($getdatakec);
}

function getkelurahan()
{
    $id_kecamatan = $this->input->post('kecamatan');
    $getdatakel = $this->kelurahan_m-
>getdatakel($id_kecamatan);
    echo json_encode($getdatakel);
}

public function delete(){
    $id = $this->input->post('id');
    $this->pelayanan_m->del($id);

    if ($this->db->affected_rows() > 0) {

```

Source Code application/controllers/pelayanan

```

$this->session->set_flashdata('success', 'Data Berhasil Di Hapus');
    }
    redirect('pelayanan');
}

// private function send_whatsapp_message($recipient,
$message)
// {
//     // Update the path below to your autoload.php,
//     // see https://getcomposer.org/doc/01-basic-usage.md
//     require_once FCPATH . 'vendor/autoload.php';

//     $sid    = "AC6f9bbe32f4185063feb03fe184d4a4ae";
//     $token  = "c09cf806f6fe58307a8bd807f9abef40";
//     $twilio = new Client($sid, $token);

//     $twilio->messages
//         ->create("whatsapp:".$recipient, // to
//             array(
//                 "from" => "whatsapp:+14155238886", // nomor
Twilio WhatsApp Anda
//                 "body" => $message
//             )
//         );
// }
}

```

## Lampiran 6 Source Code Laporan

### Source code application/controllers/Laporan

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

class Laporan extends CI_Controller {

    public function __construct() {
        parent::__construct();
        check_not_login();
        $this->load->model(['pelayanan_m', 'kota_m',
'kecamatan_m', 'kelurahan_m']);
        $this->load->library('dompdf_gen');
    }

    public function index()
    {
        $data['title'] = 'Laporan Pelayanan';
        $data['kota'] = $this->kota_m->getkota();
        $id_kota = (!empty($data['kota'])) ? $data['kota'][0]-
>id_kota : null;

        if ($id_kota !== null) {
            $data['kecamatan'] = $this->kecamatan_m-
>getdatakec($id_kota);
        } else {
            $data['kecamatan'] = array();
        }

        $id_kecamatan = (!empty($data['kecamatan'])) ?
$data['kecamatan'][0]->id_kecamatan : null;

        if ($id_kecamatan !== null) {
            $data['kelurahan'] = $this->kelurahan_m-
>getdatakel($id_kecamatan);
        } else {
            $data['kelurahan'] = array();
        }

        $this->template->load('template', 'laporan/laporan',
$data);
    }

    public function printPDF() {
        // Mendapatkan data dari form
        $data['title'] = 'Print Pelayanan';
        $id_kota = $this->input->get('kota');
        $id_kecamatan = $this->input->get('kec');
        $id_kelurahan = $this->input->get('kel');
        $tanggalawal = $this->input->get('dariTanggal');
        $tanggalakhir = $this->input->get('sampaiTanggal');

        // Mendapatkan data dari model
        $data['kota'] = $this->pelayanan_m->getKota();
    }
}
```



## Source code application/controllers/laporan

```
        $data['kecamatan'] = $this->pelayanan_m-
>getKecamatan($sid_kota);
        $data['kelurahan'] = $this->pelayanan_m-
>getKelurahan($sid_kecamatan);
        $data['tanggal'] = $this->pelayanan_m-
>filterByTanggal($sid_kecamatan, $sid_kelurahan, $tanggalawal,
$tanggalakhir);

        // Meneruskan informasi ke tampilan
        $data['tanggalawal'] = $tanggalawal;
        $data['tanggalakhir'] = $tanggalakhir;

        // Load library dan view Dompdf
        $this->load->library('dompdf_gen');
        $html = $this->load->view('laporan/print_pdf', $data,
true);

        $this->dompdf->loadHtml($html);
        $this->dompdf->setPaper('A4', 'landscape');
        $this->dompdf->render();
        $this->dompdf->stream("laporan.pdf",
array('Attachment' => 0));
    }

    public function get_kelurahan_by_kecamatan()
    {
        $kecamatanId = $this->input->post('kecamatan_id');
        $kelurahanData = $this->kelurahan_m-
>getdatakelu($kecamatanId);

        echo json_encode($kelurahanData);
    }

    public function load_data()
    {
        $kecamatanId = $this->input->post('kecamatan_id');
        $kelurahanId = $this->input->post('kelurahan_id');

        $data['results'] = $this->pelayanan_m-
>get_data_by_kecamatan_kelurahan($kecamatanId, $kelurahanId);

        $this->output->set_content_type('application/json')-
>set_output(json_encode($data['results']));
    }

    public function cetak_pdf()
    {
        $this->load_pdf_lib();

        $kotaId = $this->input->post('kota');
        $kecamatanId = $this->input->post('kec');
        $kelurahanId = $this->input->post('kel');
```

## Source code application/controllers/laporan

```
// Ambil data untuk ditampilkan di PDF
$data['py'] = $this->pelayanan_m->get();

// Ambil data untuk disertakan dalam PDF
$pdfData = $this->get_data_for_pdf($kotaId,
$kecamatanId, $kelurahanId);

// Load tampilan PDF
$html = $this->load->view('laporan/form_pdf', $data,
true);

// Set ukuran kertas dan orientasi
$ukuran_kertas = 'A4';
$orientasi = 'portrait';
$this->dompdf->set_paper($ukuran_kertas, $orientasi);

// Mengonversi HTML ke PDF
$this->dompdf->load_html($html);
$this->dompdf->render();

// Menampilkan atau mengunduh PDF
$this->dompdf->stream("laporan_data_anggota.pdf",
array('Attachment' => 0));
}

public function get_data_for_pdf($kotaId, $kecamatanId,
$kelurahanId)
{
    // Ambil data sesuai dengan kecamatan dan kelurahan
    $data = $this->pelayanan_m-
>get_data_by_kecamatan_kelurahan($kotaId, $kecamatanId,
$kelurahanId);

    // Format data sesuai kebutuhan
    $result = array();
    $no = 1;

    foreach ($data as $row) {
        $result[] = array(
            'No' => $no++,
            'Nama Peserta KB' => $row->nama_pesKB,
            'NIK' => $row->nik,
            'Nama Suami' => $row->nama_suami,
            'Alamat' => $row->alamat_pes,
            'Kota' => $row->kota,
            'Kecamatan' => $row->kecamatan,
            'Kelurahan' => $row->kelurahan,
            'Alat' => $row->alat,
            'Asuransi' => $row->asuransi,
            'Tanggal Kunjungan' => $row->tanggal_kunjungan,
            'Tanggal Dipasang' => $row->tanggal_dilayani,
            'Tanggal Dicabut' => $row->tanggal_dicabut,
            'Tanggal Lahir' => $row->tanggal_lahir,
        );
    }
}
```

```

        'Tempat Lahir' => $row->tempat_lahir,
        'Usia' => $row->usia,
        'Pendidikan Suami' => $row->pen_suami,
        'Pendidikan Istri' => $row->pen_istri,
        'Pekerjaan Suami' => $row->pek_suami,
        'Pekerjaan Istri' => $row->pek_istri,
    );
}

return $result;
}

public function generate_pdf() {
    $pdfData = $this->input->post('data');

    $this->load_pdf_lib();

    $this->dompdf->loadHtml($pdfData);
    $this->dompdf->render();
    $pdfContent = $this->dompdf->output();

    echo base64_encode($pdfContent);
}

private function load_pdf_lib() {
    if (!isset($this->dompdf)) {
        $this->load->library('dompdf_gen');
    }
}
}
}

```