

LAMPIRAN LISTING PROGRAM

Calon.php

```
<?php
defined('BASEPATH') or exit('No direct script access allowed');
class Calon extends MX_Controller{
    public function __construct(){
        parent::__construct();
        is_logged_in_admin();
        $this->load->model('CalonModel');
        $this->load->model('ParpolModel');
        $this->load->model('PeriodeModel');
        $this->load->model('PemilihModel');
    }

    public function index(){
        $data['title'] = 'Data Calon Pilkada';
        $data['calon'] = $this->CalonModel->semuaCalon();
        adminView('calon/index', $data);
    }

    public function acak_nourut(){
        $calons = $this->CalonModel->calonPeriodeAktif();
        $i = 1;
        foreach ($calons as $calon) {
            $this->CalonModel->updateNoUrut($calon['id_calon'], $i);
            $i++;
        }
        $this->session->set_flashdata('success', 'No urut calon pilkada
berhasil diacak');
        redirect('admin/calon');
    }

    public function form_calon($id = null){
        $data['title'] = 'Tambah Calon Pilkada';
        $data['periode'] = $this->PeriodeModel->semuaPeriode();
        $data['pemilih'] = $this->PemilihModel->semuaPemilih();
        $data['parpol'] = $this->ParpolModel->semuaParpol();
        if ($id) {
            $data['title'] = 'Edit Calon Pilkada';
            $data['detail'] = $this->CalonModel->calonById($id);
        }
        adminView('calon/form_calon', $data);
    }

    public function simpan_calon(){
        $this->form_validation->set_rules('id_ketua', 'Ketua',
'trim|required');
        $this->form_validation->set_rules('id_wakil', 'Wakil Ketua',
'trim|required');
```

```

        $this->form_validation->set_rules('kategori', 'Kategori',
'trim|required');
        $this->form_validation->set_rules('visi_misi', 'Visi dan Misi',
'trim|required');
        if ($this->form_validation->run() == FALSE) {
            redirect('admin/calon');
        } else {
            if ($this->input->post('submit') == "Tambah") {
                if ($this->CalonModel->tambahCalon()) {
                    $this->session->set_flashdata('success', 'Data Calon Pilkada
Berhasil Ditambah');
                } else {
                    $this->session->set_flashdata('danger', 'Data Calon Pilkada
Gagal Ditambah');
                }
            } else {
                if ($this->CalonModel->editCalon()) {
                    $this->session->set_flashdata('success', 'Data Calon Pilkada
Berhasil Diedit');
                } else {
                    $this->session->set_flashdata('danger', 'Data Calon Pilkada
Gagal Diedit');
                }
            }
            redirect('admin/calon');
        }
    }
    public function hapus_calon($id_calon){
        $calon = $this->CalonModel->calonById($id_calon);
        if ($calon) {
            if ($this->CalonModel->hapusCalon($id_calon)) {
                if ($calon['photo'] != "defaultcalon.png") {
                    $img_file = "./assets/images/calon/" . $calon['photo'];
                    unlink($img_file);
                }
                $this->session->set_flashdata('success', 'Data Calon Pilkada
Berhasil Dihapus');
            } else {
                $this->session->set_flashdata('danger', 'Data Calon Pilkada
Gagal Dihapus');
            }
        }
        redirect('admin/calon');
    }
}

```

Pemilih.php

```
<?php
defined('BASEPATH') or exit('No direct script access allowed');
class Pemilih extends MX_Controller{
    public function __construct(){
        parent::__construct();
        is_logged_in_admin();
        $this->load->model('PemilihModel');
    }
    public function index(){
        $data['title'] = 'Data Pemilih';
        $data['pemilih'] = $this->PemilihModel->semuaPemilih();
        adminView('pemilih/index', $data);
    }
    public function simpan_pemilih()
    {
        if ($this->input->post('submit') == "Tambah") {
            $cekNIK = $this->PemilihModel->pemilihByNik($this->input->post('nik', true));
            if ($cekNIK) {
                $this->session->set_flashdata('danger', 'Data Pemilih Gagal Ditambah. NIK Sudah Terdaftar');
                redirect('admin/pemilih');
            }
            if ($this->PemilihModel->tambahPemilih()) {
                $this->session->set_flashdata('success', 'Data Pemilih Berhasil Ditambah');
            } else {
                $this->session->set_flashdata('danger', 'Data Pemilih Gagal Ditambah');
            }
        } else {
            if ($this->PemilihModel->editPemilih()) {
                $this->session->set_flashdata('success', 'Data Pemilih Berhasil Diedit');
            } else {
                $this->session->set_flashdata('danger', 'Data Pemilih Gagal Diedit');
            }
        }
        redirect('admin/pemilih');
    }
    public function hapus($id_pemilih){
        $pemilih = $this->PemilihModel->pemilihById($id_pemilih);
```

```

        if ($pemilih) {
            if ($this->PemilihModel->hapusPemilih($id_pemilih)) {
                $this->session->set_flashdata('success', 'Data Pemilih Berhasil
Dihapus');
            } else {
                $this->session->set_flashdata('success', 'Data Pemilih Gagal
Dihapus');
            }
        }
        redirect('admin/pemilih');;
    }
}

```

Parpol.php

```

<?php
defined('BASEPATH') or exit('No direct script access allowed');
class Parpol extends MX_Controller{
    public function __construct(){
        parent::__construct();
        is_logged_in_admin();
        $this->load->model('ParpolModel');
    }
    public function index($id = null){
        $data['title'] = 'Data Partai Politik';
        $data['parpol'] = $this->ParpolModel->semuaParpol();
        if ($id) {
            $data['detail'] = $this->ParpolModel->parpolById($id);
        }
        adminView('parpol/index', $data);
    }
    public function simpan_parpol(){
        $this->form_validation->set_rules('nama_parpol', 'Nama partai',
'trim|required');
        $this->form_validation->set_rules('singkatan', 'Singkatan',
'trim|required');
        if ($this->form_validation->run() == FALSE) {
            redirect('admin/parpol');
        } else {
            if ($this->input->post('submit') == "Tambah") {
                if ($this->ParpolModel->tambahParpol()) {
                    $this->session->set_flashdata('success', 'Data Partai Berhasil
Ditambah');
                } else {
                    $this->session->set_flashdata('danger', 'Data Partai Gagal
Ditambah');
                }
            } else {
                if ($this->ParpolModel->editParpol()) {

```

```

        $this->session->set_flashdata('success', 'Data Partai Berhasil
Diedit');
    } else {
        $this->session->set_flashdata('danger', 'Data Partai Gagal
Diedit');
    }
}
redirect('admin/parpol');
}
}
public function hapus_parpol($id_parpol){
    $parpol = $this->ParpolModel->parpolById($id_parpol);
    if ($parpol) {
        if ($this->ParpolModel->hapusParpol($id_parpol)) {
            if ($parpol['logo'] != "defaultlogo.png") {
                $img_file = "./assets/images/parpol/" . $parpol['logo'];
                unlink($img_file);
            }
            $this->session->set_flashdata('success', 'Data Partai Berhasil
Dihapus');
        } else {
            $this->session->set_flashdata('success', 'Data Partai Gagal
Dihapus');
        }
    }
    redirect('admin/parpol');;
}
}
}

```

Periode.php

```

<?php
defined('BASEPATH') or exit('No direct script access allowed');
class Periode extends MX_Controller{
    public function __construct(){
        parent::__construct();
        is_logged_in_admin();
        $this->load->model('PeriodeModel');
    }
    public function index($id = null){
        $data['title'] = 'Data Periode Pilkada';
        $data['periode'] = $this->PeriodeModel->semuaPeriode();
        if ($id) {
            $data['detail'] = $this->PeriodeModel->periodeById($id);
        }
        adminView('periode/index', $data);
    }
    public function simpan_periode(){
        $this->form_validation->set_rules('periode_jabatan', 'Periode
jabatan', 'trim|required');
    }
}

```

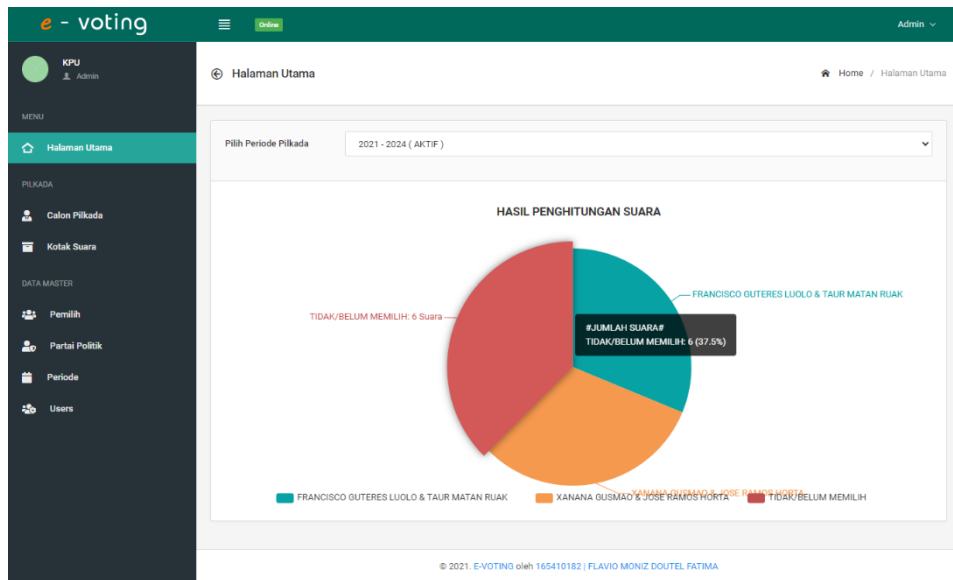
```

        $this->form_validation->set_rules('mulai_pilih', 'Waktu mulai',
'trim|required');
        $this->form_validation->set_rules('batas_pilih', 'Batas pilih',
'trim|required');
        if ($this->form_validation->run() == FALSE) {
            redirect('admin/periode');
        } else {
            $cekPeriode = $this->PeriodeModel->periodeAktif();
            if ($this->input->post('status') == 'aktif' &&
$cekPeriode['id_periode'] != $this->input->post('id_periode')) {
                $this->session->set_flashdata('danger', 'Gagal Disimpan.
Periode aktif pilkada maksimal 1 periode');
                redirect('admin/periode');
            }
            if ($this->input->post('submit') == "Tambah") {
                if ($this->PeriodeModel->tambahPeriode()) {
                    $this->session->set_flashdata('success', 'Data Periode
Berhasil Ditambah');
                } else {
                    $this->session->set_flashdata('danger', 'Data Periode Gagal
Ditambah');
                }
            } else {
                if ($this->PeriodeModel->editPeriode()) {
                    $this->session->set_flashdata('success', 'Data Periode
Berhasil Diedit');
                } else {
                    $this->session->set_flashdata('danger', 'Data Periode Gagal
Diedit');
                }
            }
            redirect('admin/periode');
        }
    }
    public function hapus_periode($id_periode){
        $periode = $this->PeriodeModel->periodeById($id_periode);
        if ($periode) {
            if ($this->PeriodeModel->hapusPeriode($id_periode)) {
                $this->session->set_flashdata('success', 'Data Periode Berhasil
Dihapus');
            } else {
                $this->session->set_flashdata('success', 'Data Periode Gagal
Dihapus');
            }
        } else {
            $this->session->set_flashdata('success', 'Data Periode Tidak
Ditemukan');
        }
        redirect('admin/periode');
    }
}

```

LAMPIRAN GAMBAR APLIKASI

1. Halaman Utama



2. Halaman Data Calon Pilkada

Data Calon Pilkada

Filter: Type to filter... Show: 10

NO. CALON	PHOTO	KETUA	WAKIL	VISI MISI	PENGUSUL	PERIODE
1		XANANA GUSMAO	JOSE RAMOS HORTA	Visi : 1. Bagian misi 2. Bagian mis...dst		2021 - 2024
2		FRANCISCO GUTERES LUOLO	TAUR MATAN RUAK	Visi : 1. Bagian mist 2. Bagian mis...dst		2021 - 2024

Showing 1 to 2 of 2 entries

© 2021. E-VOTING oleh 165410182 | FLAVIO MONIZ DOUDEL FATIMA

3. Halaman Data Kotak Suara

KPU

Admin

Menu

Halaman Utama

PLKADA

Calon Pilkada

Kotak Suara

DATA MASTER

Pemilih

Partai Politik

Periode

Users

Data Kotak Suara

Home / Data Kotak Suara

Filter:

Show: 10

NO	NIK	NAMA	PERIODE	TGL PILIH
1	55116666	NURI NAJMI WICAKSONO	2021 - 2024	27 Jun 2021 18:14
2	00700206	FLAVIO MONIZ DOUTEL FATIMA	2021 - 2024	04 Jun 2021 19:51
3	12345678	PUTRI AJU	2021 - 2024	29 Mei 2021 21:36
4	12345677	SANDIAGA UNO	2021 - 2024	29 Mei 2021 21:21
5	12345676	PRABOWO SUBIANTO	2021 - 2024	29 Mei 2021 21:21
6	12345675	MAARUF AMIN	2021 - 2024	29 Mei 2021 21:19
7	12345674	JOKO WIDODO	2021 - 2024	27 Mei 2021 16:16
8	12345673	SOFYAN GIBRAN	2021 - 2024	26 Mei 2021 20:20
9	12345672	PUTRI AYU	2021 - 2024	26 Mei 2021 14:02
10	12345671	ARIEF MUHAMAD	2021 - 2024	26 Mei 2021 14:00

Showing 1 to 10 of 10 entries

© 2021. E-VOTING oleh 165410182 | FLAVIO MONIZ DOUTEL FATIMA

4. Halaman Data Pemilih

KPU

Admin

Menu

Halaman Utama

PLKADA

Calon Pilkada

Kotak Suara

DATA MASTER

Pemilih

Partai Politik

Periode

Users

Data Pemilih

Home / Data Pemilih

+ Data Pemilih

Filter:

Show: 10

NO	NIK	NAMA	ALAMAT	JENIS KELAMIN	TGL LAHIR	AGAMA
1	12345671	ARIEF MUHAMAD	Isantul	Laki-laki	Rasuli, 28 Jan 1987	Islam
2	12345672	PUTRI AYU	Isantul	Perempuan	Kulon Progo, 10 Mei 1995	Hindu
3	12345673	SOFYAN GIBRAN	Isantul	Laki-laki	Sleman, 17 Feb 2000	Islam
4	12345674	JOKO WIDODO	solo	Laki-laki	Solo, 20 Nov 1960	Islam
5	12345675	MAARUF AMIN	Jawa Tengah	Laki-laki	Jawa Tengah, 30 Okt 1977	Islam
6	12345676	PRABOWO SUBIANTO	gosempao	Laki-laki	Gorontalo, 19 Jul 1989	Islam
7	12345677	SANDIAGA UNO	Kalimantan	Laki-laki	Kalimantan, 20 Jan 1988	Islam
8	12345678	PUTRI AJU	Jl. Raya Janti Karang Jember No. 143 Yogyakarta, Indonesia	Perempuan	Yogyakarta, 10 Mei 1996	
9	12345679	ALIMAG HUSEIN	Isantul	Laki-laki	Yogyakarta, 30 Mei 2001	Kristen
10	12345680	MUHAMMAD SAMUDERAN	Isantul	Laki-laki	Yogyakarta, 30 Mei 2001	Katolik

Showing 1 to 10 of 10 entries

© 2021. E-VOTING oleh 165410182 | FLAVIO MONIZ DOUTEL FATIMA

5. Halaman Data Partai Politik

e - voting Admin

Data Partai Politik

Filter: Type to filter... Show: 10

NO	LOGO	NAMA PARTAI	SINGKATAN
1		CONGRESSO NACIONAL DE RECONSTRUÇÃO DE TIMOR	CNRT
2		FRENTE REVOLUCIONARIO DE TIMOR LESTE INDEPENDENTE	FRETILIN
3		PARTIDO DEMOCRATICO	PD
4		KHMANEK OAN RAI KLARAN	KHUNTO
5		PARTIDO TIMOR FORTE	PKATFOR
6		PARTIDO UNIDADE DESENVOLVIMENTO DEMOCRATICO	PLDD

Showing 1 to 6 of 6 entries

Tambah Partai Politik

Nama Partai:

Singkatan:

Logo: no file chosen

© 2021. E-VOTING oleh 165410182 | FLAVIO MONIZ DOUTEL FATIMA

6. Tampilan Data Periode

e - voting Admin

Data Periode Pilkada

Filter: Type to filter... Show: 10

NO	PERIODE JABATAN	WAKTU PEMILU	STATUS
1	2021 - 2024	25 Mei 2021 s/d 01 Jul 2021	AKTIF
2	2018 - 2021	09 Apr 2018 s/d 10 Apr 2018	NONAKTIF

Showing 1 to 2 of 2 entries

Tambah Periode Pilkada

Periode Jabatan:

Tanggal Pilkada:

s/d:

Status:

© 2021. E-VOTING oleh 165410182 | FLAVIO MONIZ DOUTEL FATIMA

LAMPIRAN QUERY DATABASE

CalonModel.php

```
<?php
class CalonModel extends CI_Model{
    public function semuaCalon(){
        $this->db->select('c.*,pr.periode_jabatan,(SELECT nama from
pemilih WHERE id_pemilih=c.id_ketua) as nama_ketua, (SELECT nama
from pemilihan WHERE id_pemilih=c.id_wakil) as nama_wakil');
        $this->db->from('calon as c');
        $this->db->join('periode as pr', 'pr.id_periode=c.id_periode');
        return $this->db->get()->result_array();
    }
    public function calonPeriodeAktif(){
        $this->db->select('c.*');
        $this->db->from('calon as c');
        $this->db->join('periode as p', 'p.id_periode=c.id_periode');
        $this->db->where('p.status', 'aktif');
        $this->db->order_by('c.id_calon', 'RANDOM');
        return $this->db->get()->result_array();
    }
    public function updateNoUrut($id_calon, $no_urut){
        $this->db->where('id_calon', $id_calon);
        $this->db->update('calon', ['no urut' => $no_urut]);
    }
    public function calonById($id_calon){
        $query = $this->db->get_where('calon', ['id_calon' =>
$id_calon]);
        return $query->row_array();
    }
    public function tambahCalon(){
        try {
            $id_ketua = $this->input->post('id_ketua', true);
            $id_wakil = $this->input->post('id_wakil', true);
            $kategori = $this->input->post('kategori', true);
            $id_periode = $this->input->post('id_periode', true);
            $visi_misi = $this->input->post('visi_misi', true);
            $id_kpu = $this->session->userdata('admin-iduser');
            if ($kategori == "parpol") {
                $parpol = $this->input->post('parpol');
                $id_parpol = "";
                foreach ($parpol as $p) {
                    $id_parpol .= $p . ',';
                }
            } else {
                $id_parpol = NULL;
            }
            $photo = $this->_uploadImage(time());
```

```

        $data = [
            "id_ketua" => $id_ketua,
            "id_wakil" => $id_wakil,
            "photo" => $photo,
            "visi_misi" => $visi_misi,
            "kategori" => $kategori,
            "id_periode" => $id_periode,
            "id_parpol" => $id_parpol,
            "id_kpu" => $id_kpu,
        ];
        $this->db->insert('calon', $data);
        return true;
    } catch (\SQLException $e) {
        return false;
    }
}

public function editCalon(){
    try {
        $id_calon = $this->input->post('id_calon', true);
        $id_ketua = $this->input->post('id_ketua', true);
        $id_wakil = $this->input->post('id_wakil', true);
        $kategori = $this->input->post('kategori', true);
        $id_periode = $this->input->post('id_periode', true);
        $visi_misi = $this->input->post('visi_misi', true);
        $photo_lama = $this->input->post('photo_lama', true);
        $id_kpu = $this->session->userdata('admin-iduser');
        if ($kategori == "parpol") {
            $parpol = $this->input->post('parpol');
            $id_parpol = "";
            foreach ($parpol as $p) {
                $id_parpol .= $p . ',';
            }
        } else {
            $id_parpol = NULL;
        }
        if (!empty($_FILES["photo"]["name"])) {
            $photo = $this->uploadImage(time());
            if ($photo_lama != "defaultcalon.png") {
                $img_file = "./assets/images/calon/" . $photo_lama;
                unlink($img_file);
            }
        } else {
            $photo = $photo_lama;
        }
        $data = [
            "id_ketua" => $id_ketua,
            "id_wakil" => $id_wakil,
            "photo" => $photo,
            "visi_misi" => $visi_misi,
            "kategori" => $kategori,
            "id_periode" => $id_periode,
            "id_parpol" => $id_parpol,

```

```

        "id_kpu" => $id_kpu,
    ];
    $this->db->where('id_calon', $id_calon);
    $this->db->update('calon', $data);
    return true;
} catch (\SQLException $e) {
    return false;
}
}
public function hapusCalon($id_calon){
    try {
        $this->db->delete('calon', ['id_calon' => $id_calon]);
        return true;
    } catch (\SQLException $e) {
        return false;
    }
}
private function _uploadImage($filename){
    $config['upload_path']          = './assets/images/calon/';
    $config['allowed_types']        = 'jpeg|jpg|png';
    $config['file_name']             = $filename;
    $config['overwrite']            = true;
    $this->load->library('upload', $config);
    if ($this->upload->do_upload('photo')) {
        return $this->upload->data("file_name");
    }
    return "defaultcalon.png";
}
}
}

```

PemilihModel.php

```

<?php
class PemilihModel extends CI_Model{
    public function semuaPemilih(){
        $query = $this->db->get('pemilih');
        return $query->result_array();
    }
    public function pemilihById($id_pemilih){
        $query = $this->db->get_where('pemilih', ['id_pemilih' =>
$id_pemilih]);
        return $query->row_array();
    }
    public function pemilihByNik($nik){
        $query = $this->db->get_where('pemilih', ['nik' => $nik]);
        return $query->row_array();
    }
    public function tambahPemilih(){
        try {

```

```

$nik = $this->input->post('nik', true);
$nama = $this->input->post('nama', true);
$alamat = $this->input->post('alamat', true);
$jekel = $this->input->post('jekel', true);
$tempat_lahir = $this->input->post('tempat_lahir', true);
$ttl_lahir = $this->input->post('ttl_lahir', true);
$agama = $this->input->post('agama', true);
$data = [
    "nik" => $nik,
    "nama" => $nama,
    "alamat" => $alamat,
    "jekel" => $jekel,
    "tempat_lahir" => $tempat_lahir,
    "ttl_lahir" => $ttl_lahir,
    "agama" => $agama,
];
$this->db->insert('pemilih', $data);
return true;
} catch (\SQLException $e) {
    return false;
}
}

public function editPemilih(){
    try {
        $id_pemilih = $this->input->post('id_pemilih', true);
        $nama = $this->input->post('nama', true);
        $alamat = $this->input->post('alamat', true);
        $jekel = $this->input->post('jekel', true);
        $tempat_lahir = $this->input->post('tempat_lahir', true);
        $ttl_lahir = $this->input->post('ttl_lahir', true);
        $agama = $this->input->post('agama', true);
        $data = [
            "nama" => $nama,
            "alamat" => $alamat,
            "jekel" => $jekel,
            "tempat_lahir" => $tempat_lahir,
            "ttl_lahir" => $ttl_lahir,
            "agama" => $agama,
        ];
        $this->db->where('id_pemilih', $id_pemilih);
        $this->db->update('pemilih', $data);
        return true;
    } catch (\SQLException $e) {
        return false;
    }
}

public function hapusPemilih($id_pemilih){
    try {
        $this->db->delete('pemilih', ['id_pemilih' => $id_pemilih]);
        return true;
    } catch (\SQLException $e) {
        return false;
    }
}

```

ParpolModel.php

```
<?php
class ParpolModel extends CI_Model{
    public function semuaParpol(){
        $query = $this->db->get('parpol');
        return $query->result_array();
    }
    public function parpolById($id_parpol){
        $query = $this->db->get_where('parpol', ['id_parpol' =>
        $id_parpol]);
        return $query->row_array();
    }
    public function tambahParpol(){
        try {
            $nama_parpol = $this->input->post('nama_parpol', true);
            $singkatan = $this->input->post('singkatan', true);
            $slug = url_title($singkatan, 'dash', TRUE) . '-' . time();
            $logo = $this->uploadImage($slug);
            $data = [
                "nama_parpol" => $nama_parpol,
                "singkatan" => $singkatan,
                "logo" => $logo,
            ];
            $this->db->insert('parpol', $data);
            return true;
        } catch (\SQLException $e) {
            return false;
        }
    }
    public function editParpol(){
        try {
            $id_parpol = $this->input->post('id_parpol');
            $nama_parpol = $this->input->post('nama_parpol', true);
            $singkatan = $this->input->post('singkatan', true);
            $slug = url_title($singkatan, 'dash', TRUE) . '-' . time();
            $old_image = $this->input->post('logo_lama', true);
            if (!empty($_FILES["logo"]["name"])) {
                $logo = $this->uploadImage($slug);
                if ($old_image != "defaultlogo.png") {
                    $img_file = "./assets/images/parpol/" . $old_image;
                    unlink($img_file);
                }
            } else {
                $logo = $old_image;
            }
            $data = [
                "nama_parpol" => $nama_parpol,
```

```

        "singkatan" => $singkatan,
        "logo" => $logo,
    ];
    $this->db->where('id_parpol', $id_parpol);
    $this->db->update('parpol', $data);
    return true;
} catch (\SQLException $e) {
    return false;
}
}
public function hapusParpol($id_parpol){
    try {
        $this->db->delete('parpol', ['id_parpol' => $id_parpol]);
        return true;
    } catch (\SQLException $e) {
        return false;
    }
}
private function _uploadImage($parpol){
    $config['upload_path']           = './assets/images/parpol/';
    $config['allowed_types']         = 'jpeg|jpg|png';
    $config['file_name']              = $parpol;
    $config['overwrite']              = true;
    $this->load->library('upload', $config);
    if ($this->upload->do_upload('logo')) {
        return $this->upload->data("file_name");
    }
    return "defaultparpol.png";
}
}
}

```

SuaraModel.php

```

<?php
class SuaraModel extends CI_Model{
    public function semuaSuara(){
        $this->db->select('ks.*, p.nama as nama_pemilih, p.nik, c.*,
pr.*');
        $this->db->from('kotak_suara as ks');
        $this->db->join('pemilih as p', 'p.id_pemilih=ks.id_pemilih');
        $this->db->join('calon as c', 'c.id_calon=ks.id_calon');
        $this->db->join('periode as pr', 'pr.id_periode=ks.id_periode');
        $this->db->order_by('ks.tgl_pilih', 'DESC');
        return $this->db->get()->result_array();
    }
    public function jumlahSuara($id_periode){
        $this->db->select('ks.id_calon, (SELECT nama FROM pemilih WHERE
id_pemilih=c.id_ketua) AS nama_ketua,

```

```

        (SELECT nama FROM pemilih WHERE id_pemilih=c.id_wakil) AS
nama_wakil,
        COUNT(*) AS jml_suara');
$this->db->from('kotak_suara AS ks');
$this->db->join('periode AS pr', 'pr.id_periode=ks.id_periode');
$this->db->join('calon AS c', 'c.id_calon=ks.id_calon');
$this->db->where('pr.status', 'aktif');
$this->db->where('pr.id_periode', $id_periode);
$this->db->group_by('ks.id_calon');
$jml_memilih = $this->db->get()->result_array();

$jml_tidak_memilih = $this->tidakMemilih($id_periode);
$data = array_push($jml_memilih, $jml_tidak_memilih);
return $jml_memilih;
}
public function tidakMemilih($id_periode){
    $this->db->select('((SELECT COUNT(*) FROM pemilih) - (SELECT
COUNT(*) FROM kotak_suara WHERE id_periode=' . $id_periode . ')) AS
jml_suara');
    return $this->db->get()->row_array();
}
}

```


LAMPIRAN GAMBAR RELASI TABEL

