

BAB IV

IMPLEMENTASI DAN PEMBAHASAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi bertujuan untuk menyiapkan kegiatan penerapan sistem sesuai dengan rancangan yang telah ditentukan. Selanjutnya akan dilakukan tahapan pengujian sistem guna meminimalisir kemungkinan kesalahan yang terjadi.

4.1.1 Implementasi Basis Data

Pembuatan basis data dan tabel merupakan tahap awal dalam mengimplementasikan sebuah sistem. Digitalisasi Arsip Pindah Datang Di Dinas Kependudukan dan Pencatatan Sipil Kabupaten Bantul Berbasis Webs terdiri dari 5 (lima) tabel :

1. Tabel Biodata Adminduk (tbl_bioadm)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	bioadmId	int(11)		Tidak	Tidak ada			AUTO_INCREMENT	Ubah Hapus Lainnya
2	NIK	varchar(16)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
3	NAMA_LGKP	varchar(50)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
4	JENIS_KLMIN	varchar(10)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
5	TMPT_LHR	varchar(100)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
6	TGL_LHR	date		Tidak	Tidak ada				Ubah Hapus Lainnya
7	NO_AKTA_LHR	varchar(16)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
8	GOL_DRH	varchar(10)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
9	AGAMA	varchar(20)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
10	STAT_KWN	varchar(25)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
11	NO_AKTA_KWN	varchar(16)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
12	TGL_KWN	date		Tidak	Tidak ada				Ubah Hapus Lainnya
13	NO_AKTA_CRAI	varchar(16)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
14	TGL_CRAI	date		Tidak	Tidak ada				Ubah Hapus Lainnya
15	STAT_HBKEL	varchar(25)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
16	PDDK_AKH	varchar(10)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
17	JENIS_PKRJN	varchar(25)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
18	NAMA_LGKP_IBU	varchar(50)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
19	NAMA_LGKP_AYAH	varchar(50)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
20	NO_KK	varchar(16)	latin1_swedish_ci	Tidak	Tidak ada				Ubah Hapus Lainnya
21	terhapus	tinyint(1)		Tidak	Tidak ada				Ubah Hapus Lainnya
22	dibuatOleh	int(11)		Tidak	Tidak ada				Ubah Hapus Lainnya
23	waktuDibuat	datetime		Tidak	Tidak ada				Ubah Hapus Lainnya
24	diubahOleh	int(11)		Tidak	Tidak ada				Ubah Hapus Lainnya
25	waktuDiubah	datetime		Tidak	Tidak ada				Ubah Hapus Lainnya

Gambar 4.1 Implementasi Tabel Biodata Adminduk (tbl_bioadm)

Gambar 4.1 merupakan implementasi tabel biodata adminduk (tbl_bioadm) yang digunakan untuk menyimpan data biodata adminduk.

2. Tabel Digitalisasi (tbl_digitalisasi)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/> 2	nik	varchar(16)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 3	namafile	varchar(225)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 4	filesize	varchar(225)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 5	userid	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 6	jenisdokid	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 7	jenismutasi	varchar(100)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 8	lastupload	datetime			Tidak	Tidak ada			Ubah Hapus Lainnya

Gambar 4.2 Implementasi Tabel tbl_digitalisasi

Gambar 4.2 merupakan hasil implementasi dari tabel digitalisasi (tbl_digitalisasi) yang digunakan untuk menyimpan data digitalisasi arsip yang sudah discan.

3. Tabel Jenis Dokumen (tbl_jenisdok)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	jenisdokid	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/> 2	kodeDok	varchar(10)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 3	nama	varchar(50)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 4	terhapus	tinyint(1)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 5	dibuatOleh	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 6	waktuDibuat	datetime			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 7	diubahOleh	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 8	waktuDiubah	datetime			Tidak	Tidak ada			Ubah Hapus Lainnya

Gambar 4.3 Implementasi Tabel Jenis Dokumen (tbl_jenisdok)

Gambar 4.3 merupakan implementasi dari tabel jenis dokumen (tbl_jenisdok) yang digunakan untuk menyimpan data jenis dokumen.

4. Tabel Role (tbl_roles)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	<u>roleid</u> 🔑	tinyint(4)			Tidak	Tidak ada	role id	AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/> 2	<u>role</u>	varchar(50)	utf8_general_ci		Tidak	Tidak ada	role text		Ubah Hapus Lainnya

Gambar 4.4 Implementasi Tabel Role (tbl_roles)

Gambar 4.4 merupakan implementasi dari tabel role (tbl_roles) yang digunakan untuk menyimpan status user, sebagai admin atau sebagai petugas.

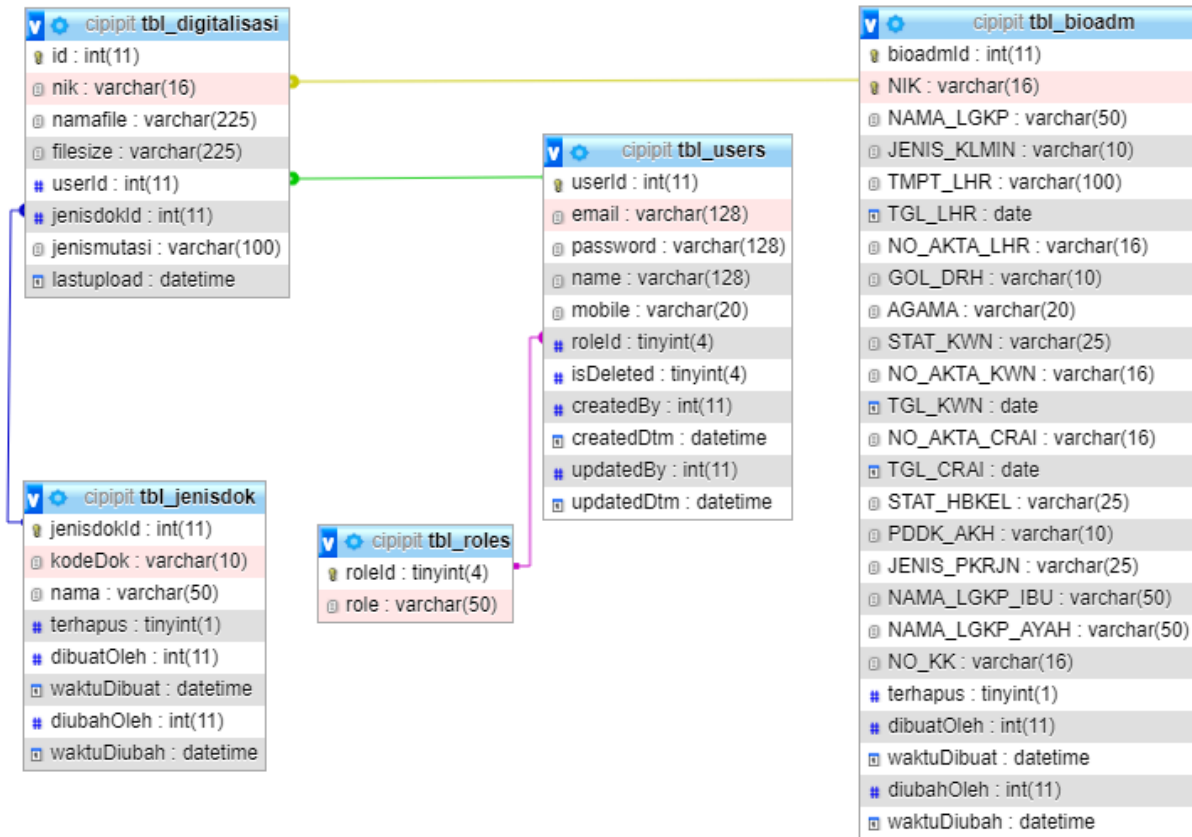
5. Tabel User (tbl_users)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	<u>userid</u> 🔑	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/> 2	<u>email</u>	varchar(128)	utf8_general_ci		Tidak	Tidak ada	login email		Ubah Hapus Lainnya
<input type="checkbox"/> 3	<u>password</u>	varchar(128)	utf8_general_ci		Tidak	Tidak ada	hashed login password		Ubah Hapus Lainnya
<input type="checkbox"/> 4	<u>name</u>	varchar(128)	utf8_general_ci	Ya	NULL		full name of user		Ubah Hapus Lainnya
<input type="checkbox"/> 5	<u>mobile</u>	varchar(20)	utf8_general_ci	Ya	NULL				Ubah Hapus Lainnya
<input type="checkbox"/> 6	<u>roleid</u> 🔑	tinyint(4)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 7	<u>isDeleted</u>	tinyint(4)			Tidak	0			Ubah Hapus Lainnya
<input type="checkbox"/> 8	<u>createdBy</u>	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 9	<u>createdDtm</u>	datetime			Tidak	Tidak ada			Ubah Hapus Lainnya
<input type="checkbox"/> 10	<u>updatedBy</u>	int(11)		Ya	NULL				Ubah Hapus Lainnya
<input type="checkbox"/> 11	<u>updatedDtm</u>	datetime		Ya	NULL				Ubah Hapus Lainnya

Gambar 4.5 Implementasi User (tbl_users)

Gambar 4.5 merupakan implementasi dari tabel user (tbl_users) yang digunakan untuk menyimpan data user.

4.1.2 Relasi Antar Tabel



Gambar 4.6 Implementasi Relasi Tabel

Gambar 4.6 merupakan implementasi relasi antar tabel pada sistem bantu Digitalisasi Arsip Pindah Datang Di Dinas Kependudukan dan Pencatatan Sipil Kabupaten Bantul Berbasis Web.

4.1.3 Fungsi Menambah Data Petugas

Fungsi berikut digunakan untuk menambahkan data user. Data user hanya dapat ditambahkan oleh admin yang merupakan super user. Fungsi ini digunakan pada halaman Users.

(Controller : user.php)

```
function addUser()
{
    if($this->isAdmin() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        $this->load->model('user_model');
        $data['roles'] = $this->user_model->getUserRoles();

        $this->global['pageTitle'] = 'Adminduk : Add New User';

        $this->loadViews("addUser", $this->global, $data,
NULL);
    }
}
```

Gambar 4.7 Fungsi Menampilkan Form Tambah User

Gambar 4.7 merupakan fungsi untuk menampilkan form tambah user. Untuk menambahkan data user baru admin harus mengisi terlebih dahulu form tambah user pada halaman Users.

(Application/controllers/user.php)

```
function addNewUser()
{
    if($this->isAdmin() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        $this->load->library('form_validation');

        $this->form_validation->set_rules('fname', 'Full
Name', 'trim|required|max_length[128]|xss_clean');
        $this->form_validation-
>set_rules('email', 'Email', 'trim|required|valid_email|xss_clean|max
_length[128]');
```

```

        $this->form_validation->
>set_rules('password','Password','required|max_length[20]');
        $this->form_validation->set_rules('cpassword','Confirm
Password','trim|required|matches[password]|max_length[20]');
        $this->form_validation->
>set_rules('role','Role','trim|required|numeric');
        $this->form_validation->set_rules('mobile','Mobile
Number','required|min_length[10]|xss_clean');

        if($this->form_validation->run() == FALSE)
        {
            $this->addUser();
        }
        else
        {
            $name = ucwords(strtolower($this->input->
>post('fname')));
            $email = $this->input->post('email');
            $password = $this->input->post('password');
            $roleId = $this->input->post('role');
            $mobile = $this->input->post('mobile');

            $userInfo = array('email'=>$email,
'password'=>getHashedPassword($password), 'roleId'=>$roleId,
'name'=> $name,
                                'mobile'=>$mobile,
'createdBy'=>$this->vendorId, 'createdDtm'=>date('Y-m-d H:i:s'));

            $this->load->model('user_model');
            $result = $this->user_model->addNewUser($userInfo);

            if($result > 0)
            {
                $this->session->set_flashdata('success', 'New
User created successfully');
            }
            else
            {
                $this->session->set_flashdata('error', 'User
creation failed');
            }
            redirect('addUser');
        }
    }
}

```

Gambar 4.8 Fungsi Validasi dan Inisialisasi Input Data User

Gambar 4.8 merupakan fungsi untuk validasi data input dan inisialisasi data untuk dikirim ke database. Fungsi ini akan melakukan validasi input data dan inisialisasi data yang

petugas yang telah diisikan. Jika data yang diisikan lengkap data akan tersimpan namun apabila data yang diisikan kurang valid maka data penyimpanan data tidak dapat dilanjutkan.

(Model : user_model.php)

```
function addNewUser($userInfo)
{
    $this->db->trans_start();
    $this->db->insert('tbl_users', $userInfo);

    $insert_id = $this->db->insert_id();

    $this->db->trans_complete();

    return $insert_id;
}
```

Gambar 4.9 Fungsi Menambahkan Data User ke Database

Gambar 4.9 merupakan fungsi untuk menambahkan data user baru ke dalam database. Setelah data yang diisikan pada form tambah data sudah divalidasi dan diinisialisasi, selanjutnya data akan diproses simpan data dengan fungsi di atas.

4.1.4 Fungsi Mengedit Data Petugas

Fungsi berikut merupakan fungsi yang digunakan untuk mengedit data petugas. Untuk mengedit data petugas menggunakan akses dari admin. Fungsi ini digunakan pada menu Users.

(Controller user.php)

```
function editOldUsr($userId = NULL)
{
    if($this->isAdmin() == TRUE || $userId == 1)
    {
        $this->loadThis();
    }
    else
    {

```

```

        if($userId == null)
        {
            redirect('userListing');
        }

        $data['roles'] = $this->user_model->getUserRoles();
        $data['userInfo'] = $this->user_model->getUserInfo($userId);

        $this->global['pageTitle'] = 'Adminduk : Edit User';

        $this->loadViews("editOldUsr", $this->global, $data,
        NULL);
    }
}

```

Gambar 4.10 Fungsi Menampilkan Form Ubah Data User

Gambar 4.10 merupakan fungsi untuk menampilkan form ubah data user lama.

Fungsi ini hanya dapat diakses oleh admin pada halaman Users.

(application/controllers/user.php)

```

function editUser()
{
    if($this->isAdmin() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        $this->load->library('form_validation');

        $userId = $this->input->post('userId');

        $this->form_validation->set_rules('fname', 'Full
Name', 'trim|required|max_length[128]|xss_clean');
        $this->form_validation->set_rules('email', 'Email', 'trim|required|valid_email|xss_clean|max_
length[128]');
        $this->form_validation->set_rules('password', 'Password', 'matches[cpassword]|max_length[20] '
);
        $this->form_validation->set_rules('cpassword', 'Confirm
Password', 'matches[password]|max_length[20]');
        $this->form_validation->set_rules('role', 'Role', 'trim|required|numeric');
    }
}

```



```

        $this->form_validation->set_rules('mobile','Mobile
Number','required|min_length[10]|xss_clean');

        if($this->form_validation->run() == FALSE)
        {
            $this->editOldUsr($userId);
        }
        else
        {
            $name = ucwords(strtolower($this->input-
>post('fname')));
            $email = $this->input->post('email');
            $password = $this->input->post('password');
            $roleId = $this->input->post('role');
            $mobile = $this->input->post('mobile');

            $userInfo = array();

            if(empty($password))
            {
                $userInfo = array('email'=>$email,
'roleId'=>$roleId, 'name'=>$name,
                                'mobile'=>$mobile,
'updatedBy'=>$this->vendorId, 'updatedDtm'=>date('Y-m-d H:i:s'));
            }
            else
            {
                $userInfo = array('email'=>$email,
'password'=>getHashedPassword($password), 'roleId'=>$roleId,
                                'name'=>ucwords($name), 'mobile'=>$mobile,
'updatedBy'=>$this->vendorId,
                                'updatedDtm'=>date('Y-m-d H:i:s'));
            }

            $result = $this->user_model->editUser($userInfo,
$userId);

            if($result == true)
            {
                $this->session->set_flashdata('success', 'User
updated successfully');
            }
            else
            {
                $this->session->set_flashdata('error', 'User
updation failed');
            }

            redirect('userListing');
        }
    }
}

```

```
}
```

Gambar 4.11 Fungsi Validasi Data Input dan Inisialisasi Data

Gambar 4.11 merupakan fungsi untuk validasi data input dan inisialisasi data untuk dikirim ke database. Fungsi ini digunakan pada akses admin di halaman Users.

(Models user_model.php)

```
function editUser($userInfo, $userId)
{
    $this->db->where('userId', $userId);
    $this->db->update('tbl_users', $userInfo);

    return TRUE;
}
```

Gambar 4.12 Fungsi Menyimpan Data Input ke Database

Gambar 4.12 merupakan fungsi untuk menyimpan data input user ke database. Data yang sudah berhasil divalidasi dan diinisialisasi kemudian disimpan ke dalam database menggunakan fungsi di atas.

4.1.5 Fungsi Menghapus Data Petugas

Fungsi berikut merupakan fungsi yang digunakan untuk menghapus data user. Hapus data users hanya dapat dilakukan oleh admin. Fungsi hapus ini hanya mengubah status users dari aktif (status 0) menjadi terhapus (status 1)

(Controllers user.php)

```
function deleteUser()
{
    if($this->isAdmin() == TRUE)
    {
        echo(json_encode(array('status'=>'access')));
    }
    else
    {
        $userId = $this->input->post('userId');
```

```

        $userInfo = array('isDeleted'=>1, 'updatedBy'=>$this->vendorId, 'updatedDtm'=>date('Y-m-d H:i:s'));

        $result = $this->user_model->deleteUser($userId, $userInfo);

        if ($result > 0) {
            echo(json_encode(array('status'=>TRUE))); }
        else { echo(json_encode(array('status'=>FALSE))); }
    }
}

```

Gambar 4.13 Fungsi Untuk Menghapus Data User

Gambar 4.13 merupakan fungsi untuk menghapus data users terpilih. Fungsi ini digunakan pada halaman users dan hanya dapat diakses oleh admin.

(Models user_model.php)

```

function deleteUser($userId, $userInfo)
{
    $this->db->where('userId', $userId);
    $this->db->update('tbl_users', $userInfo);

    return $this->db->affected_rows();
}

```

Gambar 4.14 Fungsi Untuk Menghapus Data User di Database

Gambar 4.14 merupakan fungsi untuk menghapus data user di database. Fungsi ini digunakan pada halaman users dan hanya dapat diakses oleh admin.

4.1.6 Fungsi Melihat Laporan dan Mencetak Laporan

Fungsi berikut adalah fungsi yang digunakan untuk melihat laporan dan mencetak laporan. Laporan dapat dilihat berdasarkan jenis pindah atau disajikan semua data pindah datang yang sudah dilakukan digitalisasi arsipnya. Untuk fungsi ini dapat diakses oleh admin dan petugas.

(Models BioAdm_model.php)

```
function DaftarBioAdmJoinDigital($searchText = '', $dt1 , $dt2,
$jenismutasi, $page, $segment)
{
    $this->db->select('*');
    $this->db->from('tbl_bioadm as bio');
    $this->db->join('tbl_digitalisasi as dgt', 'dgt.nik =
bio.NIK');
    $this->db->group_by("dgt.nik");

    if(!empty($searchText)) {
        $likeCriteria = "(bio.NIK LIKE '%" . $searchText . "%' or
bio.NAMA_LGKP LIKE '%" . $searchText . "%' or bio.TMPT_LHR LIKE
 '%" . $searchText . "%')";
        $this->db->where($likeCriteria);
    }
    if(!empty($dt1) && !empty($dt2)){
        $dt1 = date('Y-m-d',strtotime($dt1));
        $dt2 = date('Y-m-d',strtotime($dt2));

        $likeCriteria = "dgt.lastupload between '" . $dt1 . "' and
'" . $dt2 . "' ";
        $this->db->where($likeCriteria);
    }
    if(!empty($jenismutasi)){
        $likeCriteria = "dgt.jenismutasi='" . $jenismutasi . "'";
        $this->db->where($likeCriteria);
    }

    $this->db->where('bio.terhapus', 0);
    $this->db->limit($page, $segment);
    $query = $this->db->get();

    $result = $query->result();

    return $result;
}
```

Gambar 4. 15 Fngsi Untuk Mengambil Data Laporan

Gambar 4.15 merupakan fungsi untuk mengambil data laporan dari database. Fungsi ini akan menampilkan laporan berdasarkan tanggal dan jenis pindah.

(Controllers Laporan.php)

```
function lapcetakpdf(){
    //2 (tglawal) 3 (tglakhir) 4 (searchText)

    $this->load->model('laporan_model');

    $tgl1 = $tgl2 = $searchText = '';

    if($this->uri->segment('2')!='-') $tgl1 = date('d-m-
Y',$this->uri->segment('2'));
    else $tgl1 = '';

    if($this->uri->segment('3')!='-') $tgl2 = date('d-m-
Y',$this->uri->segment('3'));
    else $tgl2 = '';

    if($this->uri->segment('4')!='-') $jmutasi = $this->uri-
>segment('4');
    else $jmutasi = '';

    if($this->uri->segment('5')!='-') $searchText = $this->uri-
>segment('5');
    else $searchText = '';

    $pdf = new FPDF('l','mm','A5');
    $pdf->AddPage();
    $pdf->SetFont('Arial','B',16);
    $pdf->Cell(190,7,'LAPORAN MUTASI',0,1,'C');
    $pdf->SetFont('Arial','B',12);
    if($tgl1!='' && $tgl2!=''){
        $pdf->Cell(190,6,'Tanggal '.$tgl1.' s/d
'.$tgl2,0,1,'C');
    }
    if($jmutasi!=''){
        $pdf->Cell(190,6,'Jenis Mutasi :
'.$ucwords($jmutasi),0,1,'');
    }

    $pdf->Cell(10,7,'',0,1);
    $pdf->SetFont('Arial','B',10);
    $pdf->Cell(10,6,'No.',1,0);
    $pdf->Cell(35,6,'NIK',1,0);
    $pdf->Cell(45,6,'Nama',1,0);
    $pdf->Cell(35,6,'Jenis Kelamin',1,0);
    $pdf->Cell(25,6,'Tempat Lahir',1,0);
    $pdf->Cell(25,6,'Tanggal Lahir',1,1);
    $pdf->SetFont('Arial','',10);
```

```

        $laporanData = $this->laporan_model-
>cetakBasedPost($searchText,$tgl1,$tgl2,$jmutasi);

        $i=1;
        foreach ($laporanData as $row){
            if($row->JENIS_KLMIN=='L') $jk = 'Laki-Laki'; else
if($row->JENIS_KLMIN=='P') $jk = 'Perempuan';
            if($row->TGL_LHR!='0000-00-00') $dtx= date('d-m-
Y',strtotime($row->TGL_LHR)); else $dtx= '-';

            $pdf->Cell(10,6,$i,1,0);
            $pdf->Cell(35,6,$row->NIK,1,0);
            $pdf->Cell(45,6,$row->NAMA_LGKP,1,0);
            $pdf->Cell(35,6,$jk,1,0);
            $pdf->Cell(25,6,$row->TMPT_LHR,1,0);
            $pdf->Cell(25,6,$dtx,1,1);
            $i++;
        }

        $pdf->Output();
    }

```

Gambar 4.16 Fungsi Untuk Mengatur Tampilan Laporan

Gambar 4.16 merupakan fungsi untuk mengatur tampilan laporan, kemudian laporan akan ditampilkan dalam bentuk pdf.

(Models laporan_model.php)

```

function cetakBasedPost($searchText = '', $dt1 , $dt2,
$jenismutasi)
{
    $this->db->select('*');
    $this->db->from('tbl_bioadm as bio');
    $this->db->join('tbl_digitalisasi as dgt', 'dgt.nik =
bio.NIK');
    $this->db->group_by("dgt.nik");

    if(!empty($searchText)) {
        $likeCriteria = "(bio.NIK LIKE '%".$searchText."%' or
bio.NAMA_LGKP LIKE '%".$searchText."%' or bio.TMPT_LHR LIKE
'%".$searchText."%')";
        $this->db->where($likeCriteria);
    }
    if(!empty($dt1) && !empty($dt2)){
        $dt1 = date('Y-m-d',strtotime($dt1));
        $dt2 = date('Y-m-d',strtotime($dt2));
    }
}

```

```

        $likeCriteria = "dgt.lastupload between '". $dt1.'" and
        '". $dt2.'" ";
        $this->db->where($likeCriteria);
    }
    if(!empty($jenismutasi)){
        $likeCriteria = "dgt.jenismutasi='". $jenismutasi.'" ";
        $this->db->where($likeCriteria);
    }

    $this->db->where('bio.terhapus', 0);
    $query = $this->db->get();

    $result = $query->result();

    return $result;
}

```

Gambar 4.17 Fungsi Untuk Mengambil Data Dari Database Untuk Fungsi Cetak

Gambar 4.17 merupakan fungsi untuk mengambil data dari database untuk fungsi cetak.

4.1.7 Fungsi Menambah Jenis Dokumen

Fungsi berikut digunakan untuk menambahkan data jenis dokumen. Data jenis dokumen merupakan data yang digunakan untuk mengupload dokumen kelengkapan berkas pindah datang. Data jenis dokumen dapat ditambah, diedit dan dihapus sesuai kebutuhan dari kelengkapan berkas.

(Controllers JenisDok.php)

```

/**
 * Fungsi ini digunakan untuk menampilkan form tambah jenis
 * dokumen
 */
function addJenisDok()
{
    if($this->isForAll() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        $this->load->model('jenisdok_model');
    }
}

```

```

        $this->global['pageTitle'] = 'Adminduk : Buat Jenis
Dokumen Baru';

        $this->loadViews("jenisdok/addJenisDok", $this->global,
'', NULL);
    }
}

```

Gambar 4.18 Fungsi Untuk Menampilkan Form Tambah Jenis Dokumen

Gambar 4.18 merupakan fungsi untuk menampilkan form tambah jenis. Fungsi tersebut akan menampilkan form untuk menambahkan jenis dokumen yang digunakan untuk mengupload dokumen.

```

function addNewJenisDok()
{
    if($this->isForAll() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        $this->load->library('form_validation');

        $this->form_validation->
set_rules('kodeDok','Kode','trim|required|max_length[10]|xss_clean
');
        $this->form_validation->set_rules('nama','Nama Jenis
Dokumen','trim|required|max_length[50]|xss_clean');

        if($this->form_validation->run() == FALSE)
        {
            $this->addJenisDok();
        }
        else
        {
            $kode = $this->input->post('kodeDok');
            $nama = $this->input->post('nama');

            $jenisdokInfo =
array('kodeDok'=>$kode,'nama'=>$nama, 'dibuatOleh'=>$this->
vendorId, 'waktuDibuat'=>date('Y-m-d H:i:s'));

            $this->load->model('jenisdok_model');

```



```

        $result = $this->jenisdok_model->addNewJenisDok($jenisdokInfo);

        if($result > 0)
            $this->session->set_flashdata('success', 'Jenis
Dokumen Baru Berhasil Dibuat');
        else
            $this->session->set_flashdata('error', 'Jenis
Dokumen Baru Gagal Dibuat');

        redirect('jenisdok/addJenisDok');
    }
}

```

Gambar 4.19 Fungsi Untuk Validasi Data Input Dan Inisialisasi Data Untuk Dikirim ke Database

(Models jenisdok_model.php)

```

function addNewJenisDok($jenisdokInfo)
{
    $this->db->trans_start();
    $this->db->insert('tbl_jenisdok', $jenisdokInfo);

    $insert_id = $this->db->insert_id();

    $this->db->trans_complete();

    return $insert_id;
}

```

Gambar 4.20 Fungsi Untuk Menambahkan Jenis Dokumen di Database

Gambar 4.20 merupakan Fungsi untuk validasi data input dan inisialisasi data untuk dikirim ke database

4.1.8 Fungsi Mengedit Jenis Dokumen

Fungsi berikut digunakan untuk mengedit jenis dokumen. Fungsi ini digunakan pada halaman menu Jenis Dokumen.

(Controllers JenisDok.php)

```
function editOldJdk($jenisdokId = NULL)
{
    if($this->isForAll() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        if($jenisdokId == null)
        {
            redirect('DaftarJenisDok');
        }

        $data['jenisdokInfo'] = $this->jenisdok_model->getJenisDokInfo($jenisdokId);

        $this->global['pageTitle'] = 'Adminduk : Ubah Jenisd
Dokumen';

        $this->loadViews("jenisdok/editOldJdk", $this->global,
        $data, NULL);
    }
}
```

Gambar 4.21 Fungsi Untuk Menampilkan Form Edit dan Data Jenis Dokumen Lama

Gambar 4.21 merupakan fungsi untuk menampilkan form edit jenis dokumen dan menampilkan informasi jenis dokumen lama.

(Controllers JenisDok.php)

```
function editJenisDok()
{
    if($this->isForAll() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        $this->load->library('form_validation');

        $jenisdokId = $this->input->post('jenisdokId');
```

```

        $this->form_validation->
>set_rules('kodeDok','Kode','trim|required|max_length[10]|xss_clean
');
        $this->form_validation->set_rules('nama','Nama Jenis
Dokumen','trim|required|max_length[50]|xss_clean');

        if($this->form_validation->run() == FALSE)
        {
            $this->editOldJdk($jenisdokId);
        }
        else
        {
            $kode = $this->input->post('kodeDok');
            $nama = $this->input->post('nama');

            $jenisdokInfo = array('kodeDok'=>$kode,
'nama'=>$nama, 'diubahOleh'=>$this->vendorId,
'waktuDiubah'=>date('Y-m-d H:i:s'));

            $result = $this->jenisdok_model->
>editJenisDok($jenisdokInfo, $jenisdokId);

            if($result == true)
            {
                $this->session->set_flashdata('success', 'Data
Jenis Dokumen Berhasil Diubah !');
            }
            else
            {
                $this->session->set_flashdata('error', 'Data
Jenis Dokumen Gagal Diubah');
            }

            redirect('DaftarJenisDok');
        }
    }
}

```

Gambar 4.22 Fungsi Untuk Validasi Data dan Inisialisasi Data

Gambar 4.22 merupakan Fungsi untuk validasi data jenis dokumen yang telah diinput dan inisialisasi data untuk dikirim ke database. Jika sudah lengkap maka akan dapat dikirim ke database untuk disimpan, jika gagal akan menampilkan pesan gagal simpan.

(Models jenisdok_model)

```
function editJenisDok($jenisdokInfo, $jenisdokId)
{
    $this->db->where('jenisdokId', $jenisdokId);
    $this->db->update('tbl_jenisdok', $jenisdokInfo);

    return TRUE;
}
```

Gambar 4.23 Fungsi Untuk Menyimpan Data Perubahan Jenis Dokumen

Gambar 4.23 merupakan fungsi untuk menyimpan data perubahan jenis dokumen ke dalam database. Setelah selesai berhasil divalidasi dan diinisialisasi data akan dikirim ke database untuk disimpan.

4.1.9 Fungsi Menghapus Jenis Dokumen

Fungsi berikut merupakan fungsi yang digunakan untuk menghapus jenis dokumen dimana fungsi ini digunakan pada halaman menu jenis dokumen. Status jenis dokumen yang aktif adalah berstatus nol (status 0) dan untuk jenis dokumen yang sudah dihapus akan berstatus satu (status 1).

(Controllers JenisDok.php)

```
function deleteJenisDok()
{
    if($this->isForAll() == TRUE)
    {
        echo(json_encode(array('status'=>'access')));
    }
    else
    {
        $jenisdokId = $this->input->post('jenisdokId');
        $jenisdokInfo =
array('terhapus'=>1, 'diubahOleh'=>$this->vendorId,
'waktuDiubah'=>date('Y-m-d H:i:s'));

        $result = $this->jenisdok_model-
>deleteJenisDok($jenisdokId, $jenisdokInfo);
    }
}
```

```

        if ($result > 0) {
            echo(json_encode(array('status'=>TRUE))); }
        else { echo(json_encode(array('status'=>FALSE))); }
    }
}

```

Gambar 4.24 Fungsi Untuk Menghapus Jenis Dokumen

Gambar 4.24 merupakan fungsi untuk menghapus jenis dokumen pada halaman menu Jenis Dokumen.

(Models jenisdok_model.php)

```

function deleteJenisDok($jenisdokId, $jenisdokInfo)
{
    $this->db->where('jenisdokId', $jenisdokId);
    $this->db->update('tbl_jenisdok', $jenisdokInfo);

    return $this->db->affected_rows();
}

```

Gambar 4.25 Fungsi Untuk Menghapus Jenis Dokumen di Database

Gambar 4.25 merupakan fungsi yang digunakan untuk menghapus jenis dokumen pada database

4.1.10 Fungsi Menampilkan Biodata Adminduk

Fungsi berikut merupakan fungsi yang digunakan untuk menampilkan biodata adminduk. Fungsi ini digunakan pada halaman menu Biodata Adminduk. Biodata adminduk merupakan data penduduk Kabupaten Bantul yang berisi informasi tentang data kependudukan.

(Controllers BioAdm.php)

```

function DaftarBioAdm()
{
    if($this->isForAll() == TRUE)
    {
        $this->loadThis();
    }
}

```

```

    }
    else
    {
        $this->load->model('bioadm_model');

        $searchText = $this->input->post('searchText');
        $data['searchText'] = $searchText;

        $this->load->library('pagination');

        $count = $this->bioadm_model->DaftarBioAdmCount($searchText);

        $returns = $this->paginationCompress (
            "DaftarBioAdm/", $count, 5 );

        $data['bioadmRecords'] = $this->bioadm_model->DaftarBioAdm($searchText, $returns["page"], $returns["segment"]);

        $this->global['pageTitle'] = 'Adminduk : Daftar Biodata Adminduk';

        $this->loadViews("bioadm/viewList", $this->global,
            $data, NULL);
    }
}

```

Gambar 4.26 Fungsi Untuk Menampilkan Daftar Biodata Adminduk

Gambar 4.26 merupakan fungsi untuk menampilkan daftar biodata adminduk. Fungsi ini akan menampilkan halaman Biodata Adminduk

(Models bioadm_model.php)

```

function DaftarBioAdm($searchText = '', $page, $segment)
{
    $this->db->select('*');
    $this->db->from('tbl_bioadm');
    if(!empty($searchText)) {
        $likeCriteria = "(NIK LIKE '%".$searchText."%'
                        OR NAMA_LGKP LIKE '%".$searchText."%'
                        OR JENIS_KLMIN LIKE
'%".$searchText."%'
                        OR TMPT_LHR LIKE '%".$searchText."%'
                        OR NO_AKTA_LHR LIKE
'%".$searchText."%'
                        OR AGAMA LIKE '%".$searchText."%'

```

```

        OR STAT_KWN LIKE '%".$searchText."%'
        OR NO_AKTA_KWN LIKE
'%".$searchText."%'
        OR NO_AKTA_CRAI LIKE
'%".$searchText."%'
        OR STAT_HBKEL LIKE
'%".$searchText."%'
        OR JENIS_PKRJN LIKE
'%".$searchText."%'
        OR NAMA_LGKP_IBU LIKE
'%".$searchText."%'
        OR NAMA_LGKP_AYAH LIKE
'%".$searchText."%'
        OR NO_KK LIKE '%".$searchText."%'
    );
    $this->db->where($likeCriteria);
}
$this->db->where('terhapus', 0);
$this->db->limit($page, $segment);
$query = $this->db->get();

$result = $query->result();

return $result;
}

```

Gambar 4.27 Fungsi Mengambil Data Dari Database

Gambar 4.27 merupakan fungsi mengambil data dari database untuk ditampilkan. Fungsi ini mengambil data dari database untuk ditampilkan pada halaman Biodata Adminduk.

4.1.11 Fungsi Menampilkan Detail Biodata Adminduk

Fungsi berikut merupakan fungsi yang digunakan untuk menampilkan detail biodata adminduk pada halaman menu Biodata Adminduk. Data detail biodata adminduk akan ditampilkan pada halaman Biodata Adminduk, data tidak dapat diubah, ditambah maupun dihapus.

(Controllers BioAdm.php)

```
function detailBam($bioadmId = NULL)
{
    if($this->isForAll() == TRUE)
    {
        $this->loadThis();
    }
    else
    {
        if($bioadmId == null)
        {
            redirect('DaftarBioAdm');
        }
        $data['bioadmInfo'] = $this->bioadm_model->getBioAdmInfo($bioadmId);

        $this->global['pageTitle'] = 'Adminduk : Detail Biodata Adminduk';
        $this->loadViews("bioadm/detailBam", $this->global, $data, NULL);
    }
}
```

Gambar 4.28 Fungsi Untuk Informasi Detail Biodata

Gambar 4.28 merupakan fungsi yang digunakan untuk informasi detail biodata adminduk.

(Models bioadm_model)

```
function getBioAdmInfo($bioadmId)
{
    $this->db->select('*');
    $this->db->from('tbl_bioadm');
    $this->db->where('bioadmId', $bioadmId);
    $query = $this->db->get();

    return $query->result();
}
```

Gambar 4.29 Fungsi Mengambil Data Dari Database

Gambar 4.29 merupakan fungsi yang digunakan untuk mengambil data dari database dan selanjutnya data tersebut ditampilkan pada halaman menu Biodata Adminduk.

4.1.12 Fungsi Upload Dokumen Pindah Datang

Fungsi berikut merupakan fungsi yang digunakan untuk mengupload dokumen pindah datang. Fungsi ini digunakan pada halaman menu Mutasi Datang dan Mutasi Keluar.

(Controlles Upload.php)

```
public function set_upload_options($nik,$nmfile,$stype){
    $config = array();
    $nmfile = str_replace(' ', '_', $nmfile);

    if(!is_dir('./dokumen')) mkdir('./dokumen');
    if(!is_dir('./dokumen/mutasi')) mkdir('./dokumen/mutasi');
    if(!is_dir('./dokumen/mutasi/'.$stype))
mkdir('./dokumen/mutasi/'.$stype);
    if(!is_dir('./dokumen/mutasi/'.$stype.'/'.$nik))
mkdir('./dokumen/mutasi/'.$stype.'/'.$nik);

    $config['upload_path']           =
'./dokumen/mutasi/'.$stype.'/'.$nik;
    $config['allowed_types']         = 'pdf';
    $config['max_size']               = 0;
    $config['max_width']              = 0;
    $config['max_height']             = 0;
    $config['file_name']              = $nmfile;

    $arrfile = array();
    if(is_dir('./dokumen/mutasi/'.$stype.'/'.$nik)){
        if($handle =
opendir('./dokumen/mutasi/'.$stype.'/'.$nik))
        {
            while(($file = readdir($handle)) !== false)
            {
                if($file != '.' && $file != '..')
                {
                    $arrfile[] = $file;
                }
            }
            closedir($handle);
        }
    }

    foreach($arrfile as $k=>$v){
        if(preg_match("/".$nmfile."/i", $v)){
```

```

if(file_exists('./dokumen/mutasi/'.$type.'/'.$nik.'/'.$v))
unlink('./dokumen/mutasi/'.$type.'/'.$nik.'/'.$v);
    }
}

    return $config;
}

public function aksi_upload(){
    date_default_timezone_set("Asia/Jakarta");

    $nik = $this->input->post('nik');
    $type = $this->uri->segment('2');
    $this->load->model('upload_model');
    $this->load->model('jenisdok_model');
    $this->load->model('config_model');

    $this->global['pageTitle'] = 'Adminduk : Upload Dokumen';
    $this->load->model('bioadm_model');
    $databioadm = $this->bioadm_model->
>getBioAdmInfoByNIK($nik);
    $dataconfig = $this->config_model->getConfigInfo();
    $datajenis = $this->jenisdok_model->ListJenisDok();

    $valx = array();
    foreach($dataconfig as $k=>$v){
        if($type=='datang'){
            if($v->conf_key=='mutasidatang'){
                $valx[] = $v->conf_val;
            }
        }
        if($type=='keluar'){
            if($v->conf_key=='mutasikeluar'){
                $valx[] = $v->conf_val;
            }
        }
    }

    $newvalx = json_decode($valx[0],true);

    $this->load->library('upload');

    $files = $_FILES;

    $error= array();
    if(is_array($files) && count($files)>0){
        $nmfile = $this->input->post('nmfile');

        $errx = array();
        foreach($newvalx as $k=>$v){

```

```

        if($files['berkas']['name'][$k]!=''){
            if($v==1){
                $errx[$k] = 0;
            }
            else{
                $errx[$k] = 0;
            }
        }
        else{
            if($v==1){
                $errx[$k] = 1;
            }
            else{
                $errx[$k] = 0;
            }
        }
    }

    if(in_array('1',$errx)){
        ?>
        <script>
            alert('Maaf, mohon perhatikan wajib isi pada
setiap field');
        </script>
        <?php
    }
    else{
        foreach($_FILES['berkas']['name'] as $k=>$v)
        {
            $_FILES['berkas']['name']=
$files['berkas']['name'][$k];
            $_FILES['berkas']['type']=
$files['berkas']['type'][$k];
            $_FILES['berkas']['tmp_name']=
$files['berkas']['tmp_name'][$k];
            $_FILES['berkas']['error']=
$files['berkas']['error'][$k];
            $_FILES['berkas']['size']=
$files['berkas']['size'][$k];

            //            echo $_FILES['berkas']['name'].'<br>';
            if($_FILES['berkas']['name']!=''){
                // $this->upload-
>deletedatafile($databioadm[0]->NIK,$_FILES['berkas']['name']);
                $dr_[$k]['filename'] = $nmfile[$k];
                $dr_[$k]['filesize'] =
$_FILES['berkas']['size'];
                $dr_[$k]['jenisdokId'] = $k;

                $this->upload->initialize($this-
>set_upload_options($databioadm[0]->NIK,$nmfile[$k],$type));
            }
        }
    }
}

```

```

        $tes = $this->upload->do_upload('berkas');

        if(!$tes){
            //echo $_FILES['berkas']['name'].'no
upload <br>';
            $error[] = $this->upload-
>display_errors();
        }
    }

    $lct = './dokumen/mutasi/'.$type.'/'.$nik;
    if(!file_exists($lct.'/waktu.txt')){
        $filex = fopen($lct."/waktu.txt", "w");
        fwrite($filex, strtotime(date('d-m-Y H:i:s')));
        fclose($filex);
    }
    $this->upload_model->FileDBproses($databioadm[0]-
>NIK,$dr_,$type,$this->vendorId);

    ?>
    <script>
        alert('Berhasil Upload Data !');
    </script>
    <?php
    }
}

$data['config'] = $dataconfig;
$data['jenis'] = $datajenis;
$data['error'] = $error;
$data['bioadm'] = $databioadm;
$data['type'] = $type;
$this->loadViews("upload/v_upload", $this->global, $data,
NULL);
}

```

Gambar 4.30 Fungsi Menampilkan Form dan Mengupload Dokumen

Gambar 4.30 merupakan fungsi untuk menampilkan form upload dan untuk mengupload dokumen ke server.

(Models upload_model.php)

```

function FileDBproses($nik,$data,$type,$user)
{

```

```

date_default_timezone_set("Asia/Jakarta");
foreach($data as $k=>$v){
    $this->db->where('nik', $nik);
    $this->db->where('namafile', $v['filename']);
    $this->db->delete('tbl_digitalisasi');

    $dt = array(
        'nik' => $nik,
        'namafile' => $v['filename'],
        'filesize' => $v['filesize'],
        'jenismutasi' => $type,
        'jenisdokId' => $v['jenisdokId'],
        'userId' => $user,
        'lastupload' => date('Y-m-d H:i:s')
    );
    $this->db->insert('tbl_digitalisasi', $dt);
}
return TRUE;
}

```

Gambar 4.31 Fungsi Menyimpan Informasi File Upload ke Database

Gambar 4.31 merupakan fungsi untuk menyimpan informasi file upload ke database.

Fungsi ini digunakan pada halaman menu Mutasi Datang dan Mutasi Keluar.

4.1.13 Koneksi Basis Data

Fungsi berikut merupakan fungsi yang digunakan untuk mengkoneksikan dengan database yang akan digunakan.

```

$active_group = 'default';
$active_record = TRUE;

$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'root';
$db['default']['password'] = '';
$db['default']['database'] = 'cipipit';
$db['default']['dbdriver'] = 'mysqli';
$db['default']['dbprefix'] = '';
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = '';
$db['default']['char_set'] = 'utf8';

```

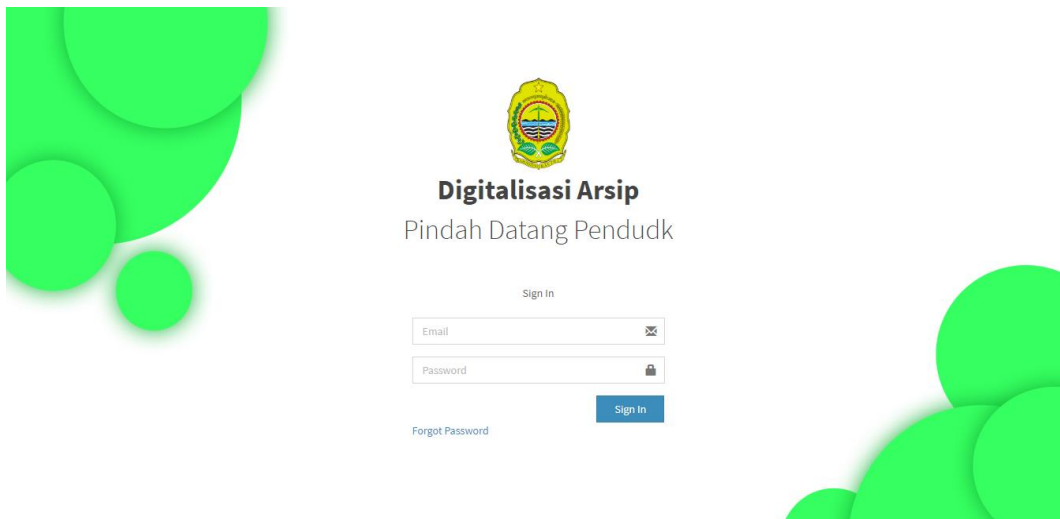
```
$db['default']['dbcollat'] = 'utf8_general_ci';  
$db['default']['swap_pre'] = '';  
$db['default']['autoinit'] = TRUE;  
$db['default']['stricton'] = FALSE;
```

Gambar 4.32 Koneksi Database

4.2 Pembahasan Sistem

Sistem bantu digitalisasi arsip ini memiliki 2 (dua) bagian sistem, untuk sistem yang pertama adalah admin dan yang kedua adalah petugas. Untuk hak akses admin dapat melakukan seluruh rangkaian proses. Sedangkan untuk hak akses petugas terbatas.

4.2.1 Halaman Login



Gambar 4.34 Tampilan Halaman Login

Gambar 4.34 merupakan tampilan halaman login, baik yang digunakan oleh admin maupun petugas tampilan halaman loginnya sama.

4.2.2 Halaman Utama Admin



Gambar 4.35 Tampilan Halaman Utama Admin

Gambar 4.35 merupakan tampilan halaman utama admin dengan login admin dan memiliki hak akses penuh atas sistem. Admin memiliki beberapa menu yang dapat diakses, antara lain menu jenis dokumen, biodata admindak, mutase datang, muasi keluar, dan users.

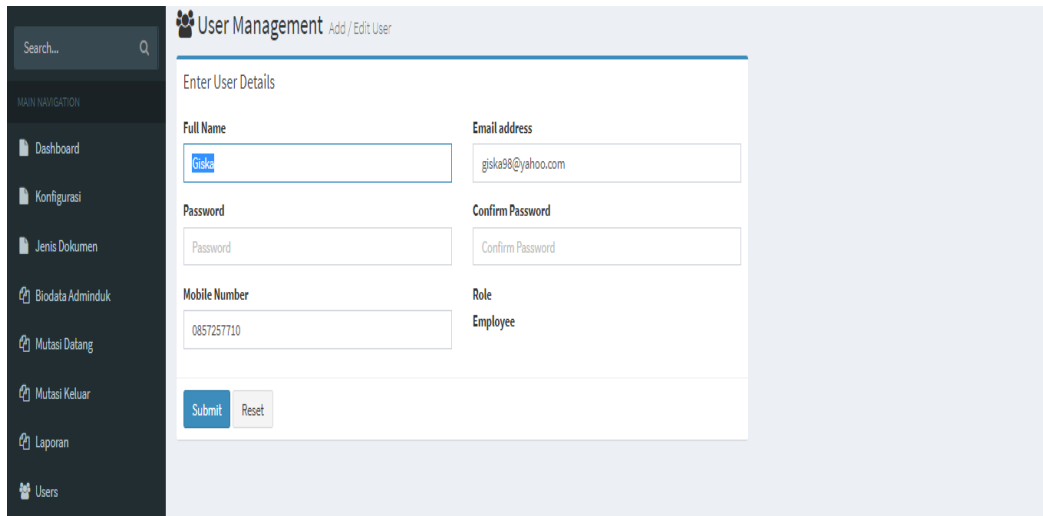
4.2.3 Halaman Tambah User

Gambar 4.36 Tampilan Halaman Tambah User

Gambar 4.36 merupakan tampilan halaman tambah user pada hak akses admin. Untuk menambahkan data user hanya dapat dilakukan oleh admin dengan login admin, kemudian

mengisikan data user pada form tambah user. Untuk role tidak dapat diubah karena pada sistem bantu ini admin hanya 1 (satu).

4.2.4 Halaman Edit User

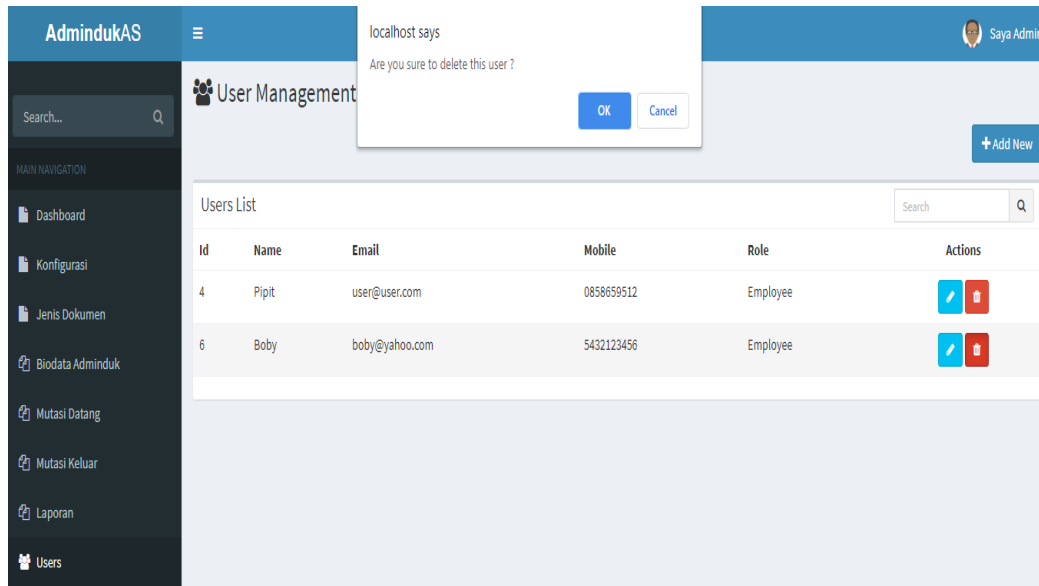


The screenshot shows a web application interface for 'User Management'. On the left is a dark sidebar with a search bar and a 'MAIN NAVIGATION' menu containing links for Dashboard, Konfigurasi, Jenis Dokumen, Biodata Adminduk, Mutasi Datang, Mutasi Keluar, Laporan, and Users. The main content area is titled 'User Management Add / Edit User'. It features a form titled 'Enter User Details' with the following fields: 'Full Name' (containing 'giska'), 'Email address' (containing 'giska98@yahoo.com'), 'Password' (containing 'Password'), 'Confirm Password' (containing 'Confirm Password'), 'Mobile Number' (containing '0857257710'), and 'Role' (set to 'Employee'). At the bottom of the form are 'Submit' and 'Reset' buttons.

Gambar 4.37 Tampilan Halaman Edit User

Gambar 4.37 merupakan tampilan halaman edit user, dimana user yang sudah ada diedit datanya sesuai dengan data yang diinginkan. Setelah selesai melakukan edit user klik tombol submit untuk menyimpan.

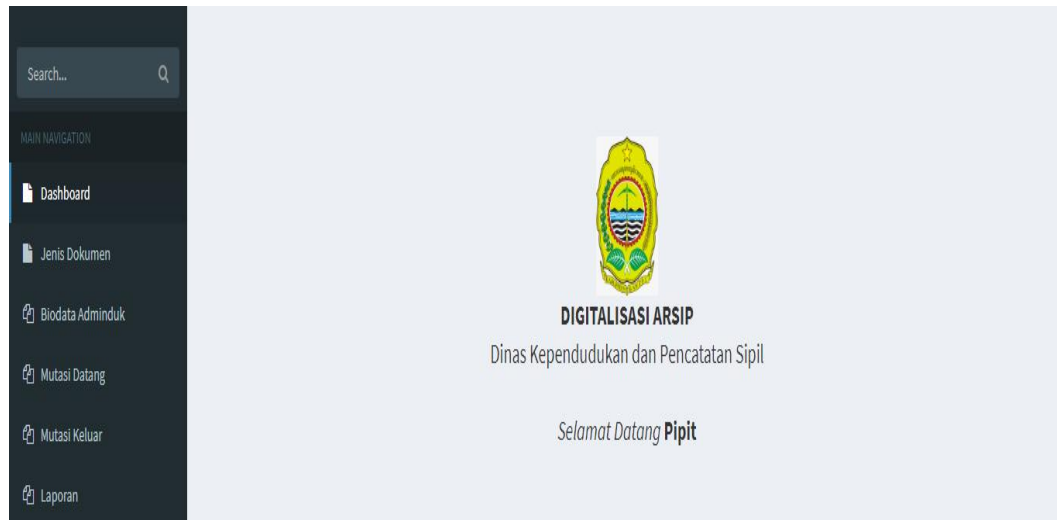
4.2.5 Halaman Hapus User



Gambar 4.38 Tampilan Halaman Hapus User

Gambar 4.38 merupakan tampilan halaman hapus user. Untuk menghapus user harus dengan hak akses sebagai admin. Sebelum menghapus user terlebih dahulu masuk pada halaman user kemudian memilih user yang akan dihapus dan klik tombol sampah atau hapus, maka akan tampil pesan untuk menghapus user kemudian klik ok.

4.2.6 Halaman Utama User



Gambar 4.39 Tampilan Halaman Utama User (Petugas)

Gambar 4.39 merupakan tampilan halaman utama user (petugas). Hak akses user (petugas) terdapat 5 (lima) menu yang dapat diakses, yaitu menu jenis dokumen, biodata admonduk, mutase datang, mutasi keluar dan laporan.

4.2.7 Halaman Upload Mutasi Datang

The screenshot displays the 'Dokumen Mutasi Datang' (Incoming Mutation Documents) upload page. The page title is 'Dokumen Mutasi Datang' with a subtitle 'Kelola Dokumen'. It features a form for uploading documents, including fields for NIK, Nama Lengkap, Jenis Kelamin, Tempat Lahir, and Tanggal Lahir. Below the form is a table with columns: No, Jenis, Upload, Dokumen, and Status. The table lists six document types, each with a 'Choose File' button and a 'Status' column. A note at the bottom indicates that documents marked with an asterisk (*) are mandatory.

No	Jenis	Upload	Dokumen	Status
1	Surat Permohonan (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
2	Surat Pindah Daerah Asal (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
3	Biodata (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
4	Kartu Tanda Penduduk (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
5	Kartu Keluarga	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
6	Surat Kedatangan	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak

(*) wajib diisi

Gambar 4.39 Tampilan Halaman Upload Mutasi Dtang

Gambar 4.39 merupakan tampilan halaman upload mutasi datang. Pada menu ini dapat diakses pada login admin maupun user (petugas). Menu ini digunakan untuk mengupload

berkas atau dokumen mutasi datang. Pada tampilan menu ini petugas menginputkan NIK terlebih dahulu kemudian klik tombol cari untuk menampilkan datanya. Petugas memastikan data tersebut benar kemudian langkah berikutnya adalah mengupload dokumen pada menu upload yang sudah disediakan.

Upload dokumen sudah dibedakan berdasarkan jenis dokumen sehingga dokumen tidak akan tertukar dan dipastikan lengkap dokumen tersebut. Setelah upload berkas selesai, data tersebut akan ditandai sudah dilakukan digitalisasi arsip untuk mutasi datang.

4.2.8 Halaman Upload Mutasi Keluar

No	Jenis	Upload	Dokumen	Status
1	Surat Permohonan (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
2	Surat Pindah Daerah Asal (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
3	Biodata (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
4	Kartu Tanda Penduduk (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
5	Kartu Keluarga (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak
6	Surat Pindah Keluar (*)	<input type="button" value="Choose File"/> No file chosen	Kosong	Tidak

(*) wajib diisi

Gambar 4.40 Tampilan Halaman Upload Mutasi Keluar

Gambar 4.40 merupakan tampilan halaman upload mutasi keluar. Pada menu ini dapat diakses pada login admin maupun user (petugas). Menu ini digunakan untuk mengupload berkas atau dokumen mutasi keluar. Pada tampilan menu ini petugas menginputkan NIK

terlebih dahulu kemudian klik tombol cari untuk menampilkan datanya. Petugas memastikan data tersebut benar kemudian langkah berikutnya adalah mengupload dokumen pada menu upload yang sudah disediakan.

Upload dokumen sudah dibedakan berdasarkan jenis dokumen sehingga dokumen tidak akan tertukar dan dipastikan lengkap dokumen tersebut. Setelah upload berkas selesai, data tersebut akan ditandai sudah dilakukan digitalisasi arsip untuk mutasi keluar.

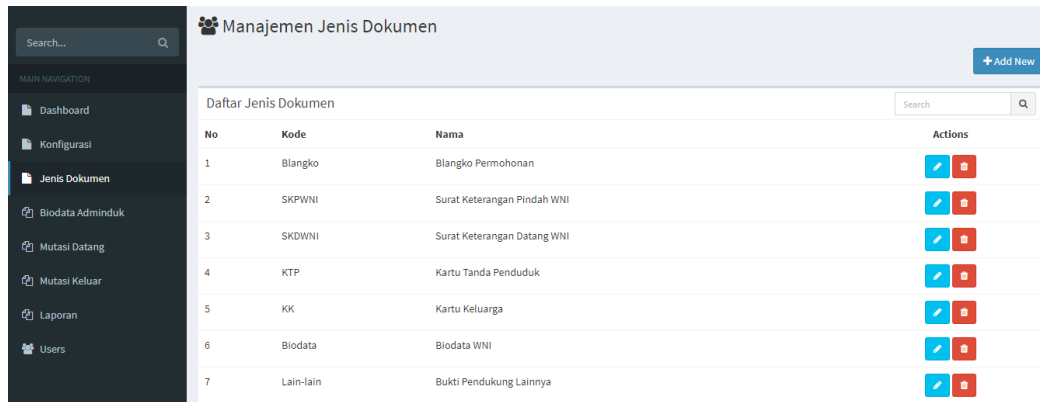
4.2.9 Halaman Menu Laporan

No	NIK	Nama	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Actions
1	99728129	Giska9	Perempuan	Yogyakarta9	18-02-2020	Q













Gambar 4.41 Tampilan Halaman Menu Laporan

Gambar 4.41 merupakan tampilan halaman menu laporan. Halaman menu laporan dapat diakses dengan login admin maupun user (petugas). Halaman ini akan menampilkan data penduduk yang sudah dilakukan digitalisasi arsipnya. Data yang disajikan oleh menu laporan dengan rentan 1 (satu) bulan. Tampilan laporan adalah berupa file pdf.

4.2.10 Halaman Jenis Dokumen



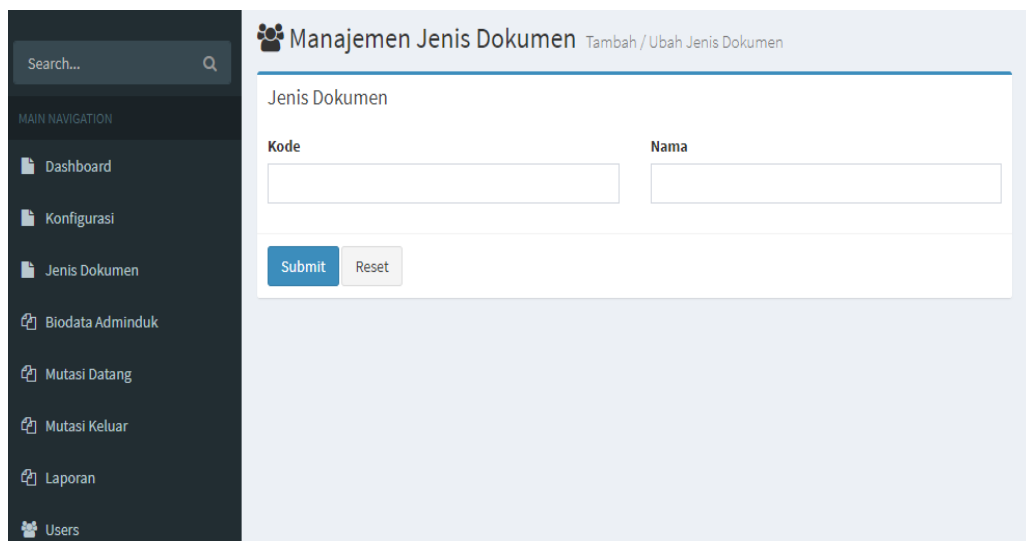
The screenshot shows the 'Manajemen Jenis Dokumen' page. On the left is a dark sidebar with a search bar and a 'MAIN NAVIGATION' menu containing: Dashboard, Konfigurasi, Jenis Dokumen (highlighted), Biodata Adminduk, Mutasi Datang, Mutasi Keluar, Laporan, and Users. The main content area has a title 'Manajemen Jenis Dokumen' and a '+ Add New' button. Below is a table titled 'Daftar Jenis Dokumen' with a search bar. The table has columns: No, Kode, Nama, and Actions. It lists 7 document types.

No	Kode	Nama	Actions
1	Blangko	Blangko Permohonan	 
2	SKPWNI	Surat Keterangan Pindah WNI	 
3	SKDWN	Surat Keterangan Datang WNI	 
4	KTP	Kartu Tanda Penduduk	 
5	KK	Kartu Keluarga	 
6	Biodata	Biodata WNI	 
7	Lain-lain	Bukti Pendukung Lainnya	 

Gambar 4.42 Tampilan Halaman Jenis Dokumen

Gambar 4.42 merupakan tampilan halaman jenis dokumen. Menu ini dapat diakses dengan login admin maupun login user (petugas). Kedua hak akses tersebut dapat melakukan aksi tambah, ubah dan hapus pada data jenis dokumen,

4.2.11 Halaman Tambah Jenis Dokumen



The screenshot shows the 'Manajemen Jenis Dokumen' page with the 'Tambah / Ubah Jenis Dokumen' form. The sidebar is the same as in the previous image. The main content area has a title 'Manajemen Jenis Dokumen' and a subtitle 'Tambah / Ubah Jenis Dokumen'. Below is a form titled 'Jenis Dokumen' with two input fields: 'Kode' and 'Nama'. At the bottom of the form are two buttons: 'Submit' and 'Reset'.

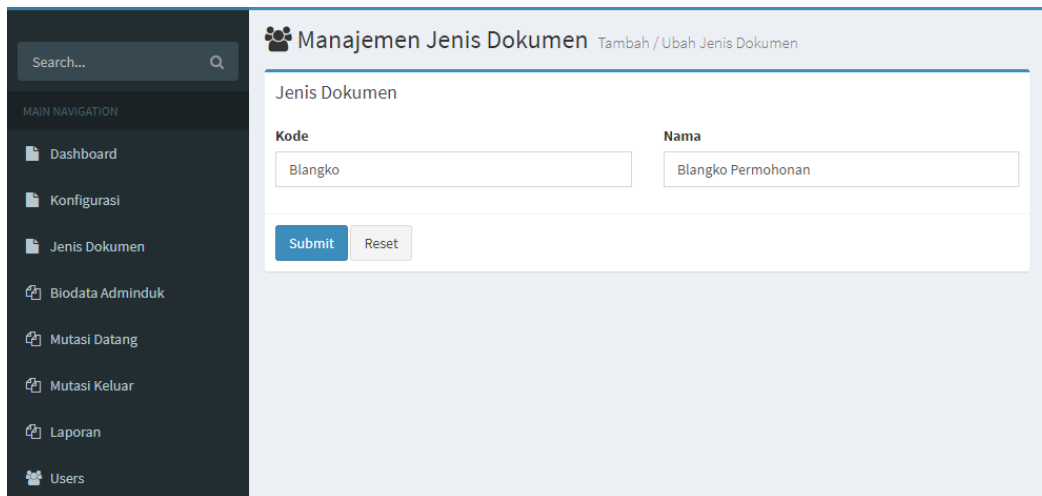
Gambar 4.43 Tampilan Halaman Tambah Jenis Dokumen

Gambar 4.43 merupakan tampilan halaman tambah jenis dokumen. Apabila ada tambahan jenis dokumen dalam proses upload dokumen pindah datang, admin maupun user

(petugas) dapat melakukan tambah jenis dokumen. Jenis dokumen dapat ditambahkan sesuai dengan kebutuhan dari berkas permohonan pindah datang.

Untuk menambahkan jenis dokumen isikan kode dan nama dari jenis dokumen tersebut. Selanjutnya klik submit untuk menyimpan data jenis dokumen.

4.2.12 Halaman Edit Jenis Dokumen

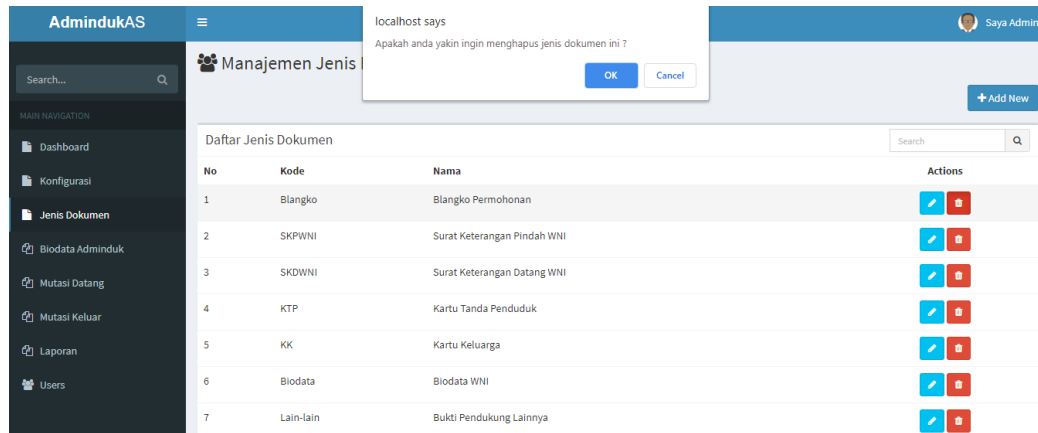


The screenshot displays a web application interface for managing document types. On the left is a dark sidebar with a search bar and a list of navigation items: Dashboard, Konfigurasi, Jenis Dokumen, Biodata Adminduk, Mutasi Datang, Mutasi Keluar, Laporan, and Users. The main content area has a light blue header with the title 'Manajemen Jenis Dokumen' and a subtitle 'Tambah / Ubah Jenis Dokumen'. Below the header, the form is titled 'Jenis Dokumen'. It contains two input fields: 'Kode' with the value 'Blangko' and 'Nama' with the value 'Blangko Permohonan'. At the bottom of the form are two buttons: 'Submit' (in blue) and 'Reset' (in light gray).

Gambar 4.44 Tampilan Halaman Edit Jenis Dokumen

Gambar 4.44 merupakan tampilan halaman edit jenis dokumen. Jika jenis dokumen dirasa kurang sesuai dengan dokumen pindah datang, maka jenis dokumen dapat diupdate sesuai dengan yang diinginkan. Admin maupun user (petugas) hanya tinggal mengubah kode dan nama sesuai data yang baru.

4.2.13 Halaman Hapus Jenis Dokumen



Gambar 4.45 Tampilan Halaman Hapus Jenis Dokumen

Gambar 4.45 merupakan tampilan halaman hapus jenis dokumen. Apabila disasa jenis dokumen tersebut sudah tidak digunakan ataupun sudah tidak relevan dengan berkas pindah datang, maka jenis dokumen dapat dihapus oleh admin maupun user (petugas). Sebelum menghapus data pastikan terlebih dahulu data jenis dokumen sudah tidak digunakan, kemudian klik tombol hapus. Maka, akan tampil pesan hapus data klik tombol ok dan data jenis dokumen sudah terhapus.