

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. Implementasi sistem dapat menjadi tolok ukur sistem tersebut dapat dipergunakan atau berjalan dengan baik pada Seksi Pindah Datang Penduduk di Dinas Kependudukan dan Pencatatan Sipil Kabupaten Bantul.

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 Web terdiri dari 5 (lima) tabel yang digunakan dalam sistem ini antara lain tabel users, jenis dokumen, digitalisasi, role, dan jenis dokumen. Penjelasan masing-masing tabel sebagai berikut :

a. Tabel Users (tbl_users)

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

Gambar 4.1 Implementasi User (tbl_users)

Gambar 4.1 merupakan implementasi dari tabel users (tbl_users) yang digunakan untuk menyimpan data user. Data users yang disimpan merupakan data dari petugas yang dapat digunakan sebagai hak akses masuk ke dalam sistem sebagai petugas.

b. 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 4.2 Implementasi Tabel Jenis Dokumen (tbl_jenisdok)

Gambar 4.2 merupakan implementasi dari tabel jenis dokumen (tbl_jenisdok) yang digunakan untuk menyimpan data jenis dokumen. Dokumen permohonan pindah dan datang penduduk memiliki beberapa jenis antara lain blangko permohonan, surat keterangan pindah atau datang, kartu keluarga, KTP

elektronik dan dokumen pendukung lainnya. Pada tabel ini menyimpan data jenis dokumen tersebut.

c. 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 4.3 Implementasi Tabel tbl_digitalisasi

Gambar 4.3 merupakan hasil implementasi dari tabel digitalisasi (tbl_digitalisasi) yang digunakan untuk menyimpan data digitalisasi arsip yang sudah discan. Tabel digitalisasi merupakan penghubung dari beberapa tabel yang nantinya dapat mengeluarkan informasi yang diinginkan. Setelah arsip digital diupload data tersebut disimpan pada tabel digitalisasi ini.

d. Tabel Role (tbl_roles)

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
<input type="checkbox"/> 1	roleid	tinyint(4)			Tidak	Tidak ada	role id	AUTO_INCREMENT	Ubah Hapus Lainnya
<input type="checkbox"/> 2	role	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, memberikan hak akses kepada user sebagai admin atau sebagai petugas.

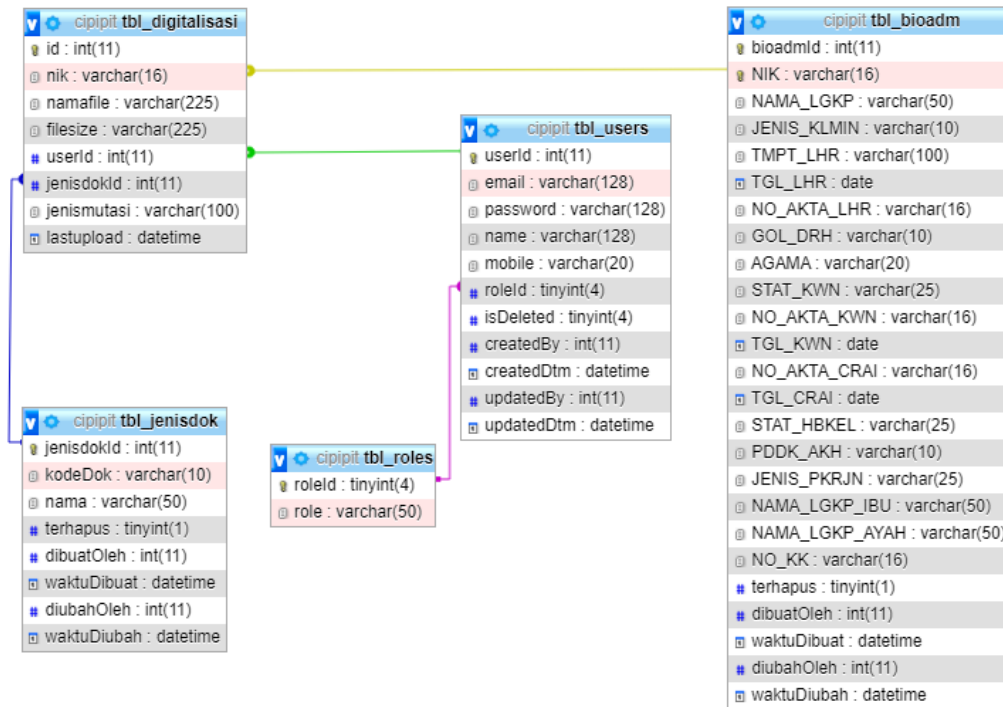
e. Tabel Biodata Adminduk (tbl_bioadm)

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

Gambar 4.5 Implementasi Tabel Biodata Adminduk (tbl_bioadm)

Gambar 4.5 merupakan implementasi tabel biodata adminduk (tbl_bioadm) yang digunakan untuk menyimpan data biodata adminduk. Struktur tabel pada tabel tersebut telah disamakan dengan tabel utama pada tabel biodata adminduk pada database kependudukan di Dinas Kependudukan dan Pencatatan Sipil Kabupaten Bantul. Tabel ini menyimpan informasi kependudukan di Kabupaten Bantul.

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. Untuk tabel users berelasi dengan tabel roles dan tabel digitalisasi. Tabel users mengambil status *user* dari tabel roles. Untuk relasi antara tabel users dan digitalisasi dimaksudkan petugas yang melakukan pengarsipan digital pindah datang dapat terlihat.

Relasi tabel berikutnya adalah relasi antara tabel jenis dokumen dengan tabel digitalisasi. Tabel jenis dokumen menampilkan jenis dokumen untuk mengunggah file arsip digital pada tabel digitalisasi arsip.

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_c
lean|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_cl
ean|max_length[128]');
        $this->form_validation-
>set_rules('password','Password','matches[cpassword]|max_len
gth[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. Data user yang dihapus adalah data petugas karena data admin merupakan superuser dan hanya ada 1 (satu) data saja.

(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)

```

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]|x
ss_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]|x
ss_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).

(Controlllers 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
                        )";

        $this->db->where($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,$type){
    $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/'.$type))
mkdir('./dokumen/mutasi/'.$type);
    if(!is_dir('./dokumen/mutasi/'.$type.'/'.$nik))
mkdir('./dokumen/mutasi/'.$type.'/'.$nik);

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

    $arrfile = array();
    if(is_dir('./dokumen/mutasi/'.$type.'/'.$nik)){
        if($handle =
opendir('./dokumen/mutasi/'.$type.'/'.$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]['jenisId'] = $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 terbagi menjadi 2 (dua) hak akses yaitu untuk sistem yang pertama adalah hak akses sebagai admin dan yang kedua adalah sebagai petugas. Untuk hak akses admin dapat melakukan proses sebagai berikut membuat akun petugas, mengedit akun petugas, menghapus akun petugas, mengubah pengaturan konfigurasi untuk mutasi datang maupun mutasi keluar, selain itu hak akses admin dapat melakukan menambah jenis dokumen, mengedit jenis dokumen, menghapus jenis dokumen, melihat laporan, melihat biodata adminduk, dan yang terakhir admin dapat melakukan import data penduduk.

Sedangkan untuk hak akses petugas dapat melakukan proses sebagai berikut menambah jenis dokumen, mengedit jenis dokumen, menghapus jenis dokumen, melihat biodata adminduk, melakukan upload berkas mutasi datang dan mutasi keluar yang sudah berbentuk pdf dan sudah discan oleh petugas. Petugas juga dapat melihat laporan.

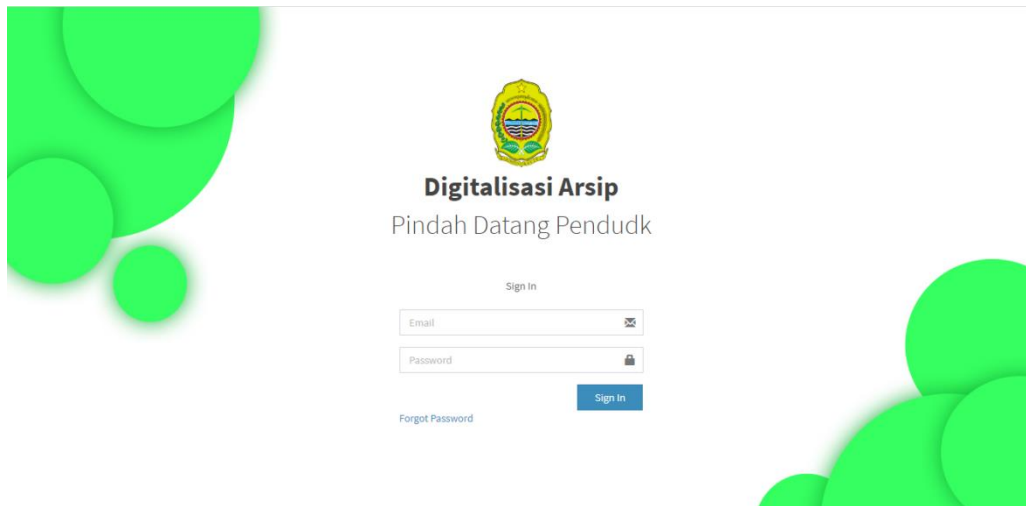
Sebelum menggunakan sistem bantu digital arsip ini terlebih dahulu petugas diberikan hak aksesnya untuk dapat masuk ke dalam sistem. Untuk pemberian hak akses petugas, admin akan membuatkan akun petugas yang dapat digunakan untuk melakukan proses upload arsip digital pindah datang.

Arsip pindah datang sebelum diupload ke dalam sistem ini terlebih dahulu dilakukan scan dokumen dengan menggunakan scanner. Selanjutnya file disimpan dengan format *namapemohon.pdf* yang kemudian data tersebut diupload ke dalam

sistem arsip digital. Nama penduduk yang telah dilakukan pengarsipan digital terhadap permohonan mutasi keluar maupun mutasi datangnya akan ditandai oleh sistem bahwa data penduduk tersebut memiliki arsip digital permohona mutasi.

Dengan demikian dapat dihasilkan laporan data pemohoan yang sudah ada arsip digita permohonan mutasi keluar atau mutasi datang. Sehingga ketika petugas mencari arsip permohonan mutasinya dapat dengan mudah menemukannya tanpa harus membongkar arsip fisiknya. Berikut pembahasan sistem bantu pengarsipan digital pindah datang.

4.2.1 Halaman Login

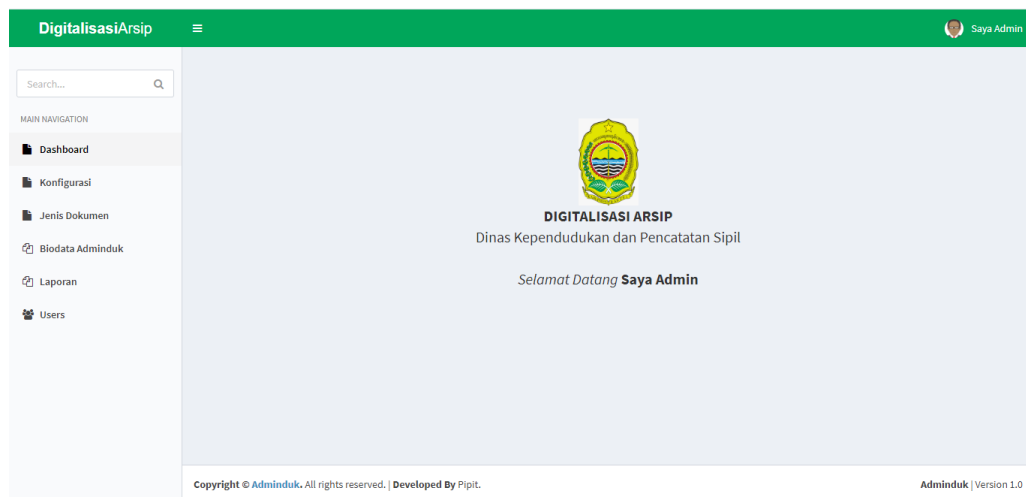


Gambar 4.33 Tampilan Halaman Login

Gambar 4.33 merupakan tampilan utama sistem ini pada halaman login, baik yang digunakan oleh admin maupun petugas tampilan halaman loginnya sama. Terdapat kotak isian untuk *username* yang berupa email dan *password*. Jika *login*

sebagai admin maka setelah login akan masuk pada tampilan menu untuk hak akses admin dan jika *login* sebagai petugas maka setelah login akan masuk pada tampilan menu untuk hak akses petugas.

4.2.2 Halaman Utama Admin



Gambar 4.34 Tampilan Halaman Utama Admin

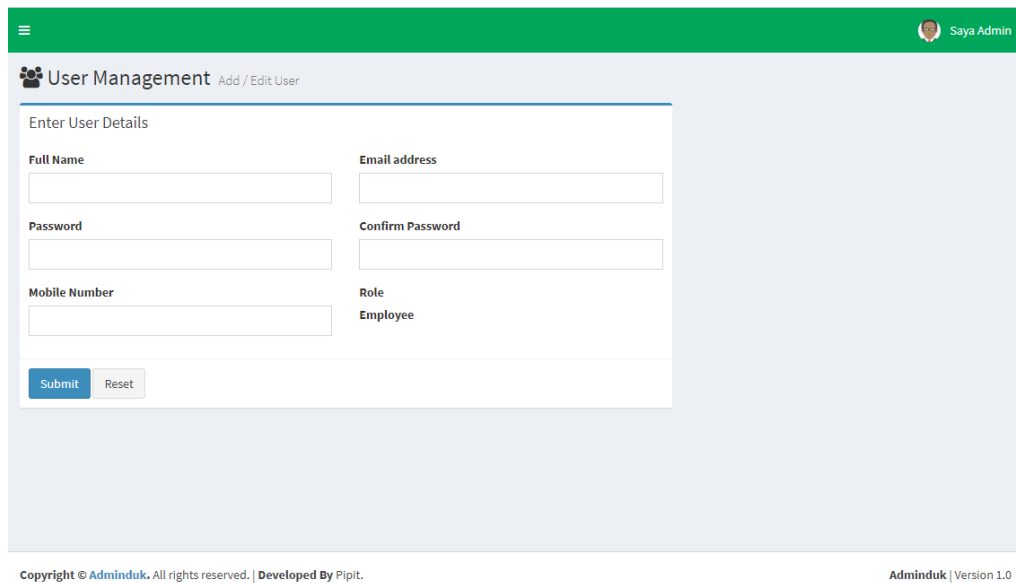
Gambar 4.34 merupakan tampilan halaman utama admin dengan login sebagai admin dan memiliki hak akses sebagai admin. Seperti yang terlihat pada gambar menu pada admin yang dapat diakses adalah menu konfigurasi yang digunakan untuk mengatur jenis-jenis dokumen yang dapat diupload. Menu kedua adalah jenis dokumen, pada menu ini admin dapat menambah, mengedit dan menghapus jenis dokumen sesuai dengan yang dibutuhkan.

Menu ketiga adalah menu Biodata Adminduk, pada menu ini baik admin maupun petugas hanya dapat melihat data karena sistem ini dibangun sebagai

sistem utama dimana data penduduk diambil dari database utama di Dinas Kependudukan dan Pencatatan Sipil Kabupaten Bantul sehingga data penduduk tidak dapat diubah maupun dihapus.

Menu keempat adalah menu laporan, menu ini digunakan untuk menampilkan laporan dari pengarsipan digital dari berkas permohonan mutasi keluar dan mutasi datang. Menu terakhir merupakan menu kelima adalah menu users, pada menu ini admin dapat menambah, mengedit dan menghapus data users. Menu ini hanya ada pada hak akses admin.

4.2.3 Halaman Tambah User



The screenshot displays a web application interface for user management. At the top, there is a green header bar with a hamburger menu icon on the left and a user profile icon labeled 'Saya Admin' on the right. Below the header, the main content area has a light blue background. A white box titled 'User Management' with a sub-link 'Add / Edit User' is centered. Inside this box, the section 'Enter User Details' contains several input fields: 'Full Name', 'Email address', 'Password', 'Confirm Password', 'Mobile Number', and 'Role'. The 'Role' field is currently set to 'Employee'. At the bottom of the form, there are two buttons: 'Submit' (in blue) and 'Reset' (in light gray). The footer of the page contains copyright information: 'Copyright © Adminduk. All rights reserved. | Developed By Pipit.' on the left and 'Adminduk | Version 1.0' on the right.

Gambar 4.35 Tampilan Halaman Tambah User

Gambar 4.35 merupakan tampilan halaman tambah *user* pada hak akses admin. Untuk menambahkan data *user* hanya dapat dilakukan oleh admin dengan

login admin, kemudian mengisi data *user* pada form tambah *user*. Untuk role tidak dapat diubah karena pada sistem bantu ini admin hanya 1 (satu). Data akun yang akan menjadi *username* adalah email, jika ada *user* yang memiliki email sama meskipun data yang diisikan pada kotak isian lainnya berbeda tetap dianggap ganda.

4.2.4 Halaman Edit User

The screenshot shows a web application interface for user management. At the top, there is a green header bar with a menu icon on the left and a user profile labeled 'Saya Admin' on the right. Below the header, the main content area has a title 'User Management' with a subtitle 'Add / Edit User'. The form itself is titled 'Enter User Details' and is divided into two columns. The left column contains fields for 'Full Name' (filled with 'Pipit'), 'Password', and 'Mobile Number' (filled with '0858659512'). The right column contains fields for 'Email address' (filled with 'user@user.com'), 'Confirm Password', and 'Role' (filled with 'Employee'). At the bottom of the form, there are two buttons: 'Submit' and 'Reset'. The footer of the page contains copyright information 'Copyright © Adminduk. All rights reserved. | Developed By Pipit.' and the version 'Adminduk | Version 1.0'.

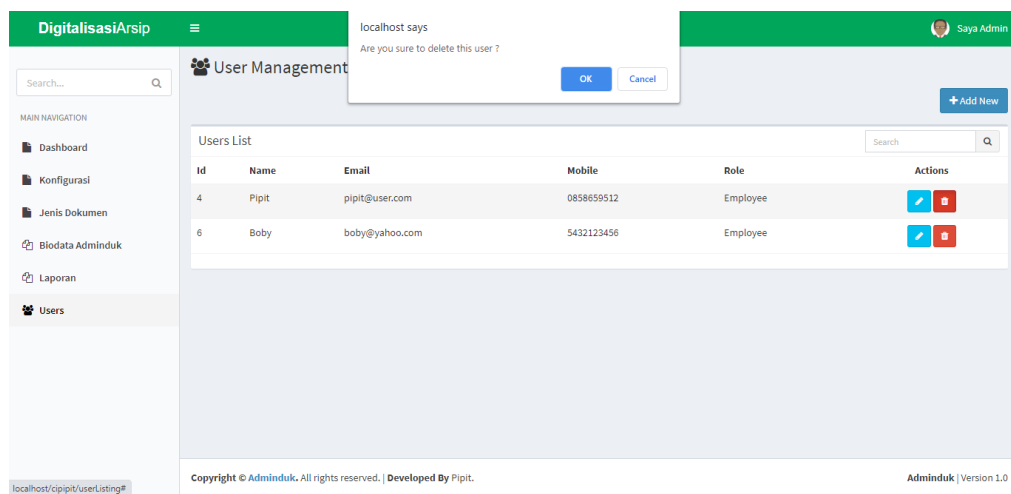
Gambar 4.36 Tampilan Halaman Edit *User*

Gambar 4.36 merupakan tampilan halaman edit *user*, dimana *user* yang sudah tersimpan akan diedit datanya sesuai dengan data yang diinginkan. Setelah selesai melakukan edit *user* klik tombol *submit* untuk menyimpan. Elemen data *user* dapat diedit terkecuali pada elemen data *role*.

Role merupakan status dari *user*, namun untuk status *user* tidak dibuat banyak karena sistem ini akan diolah oleh Bidang PIAK khususnya pada Seksi

Sistem Informasi Administrasi Kependudukan pada Dinas Kependudukan dan Pencatatan Sipil Kabupaten Bantul yang bertugas sebagai pengelola sistem. Pembatasan admin dimaksudkan untuk menghindari penyalahgunaan hak akses karena terlalu banyaknya akun admin.

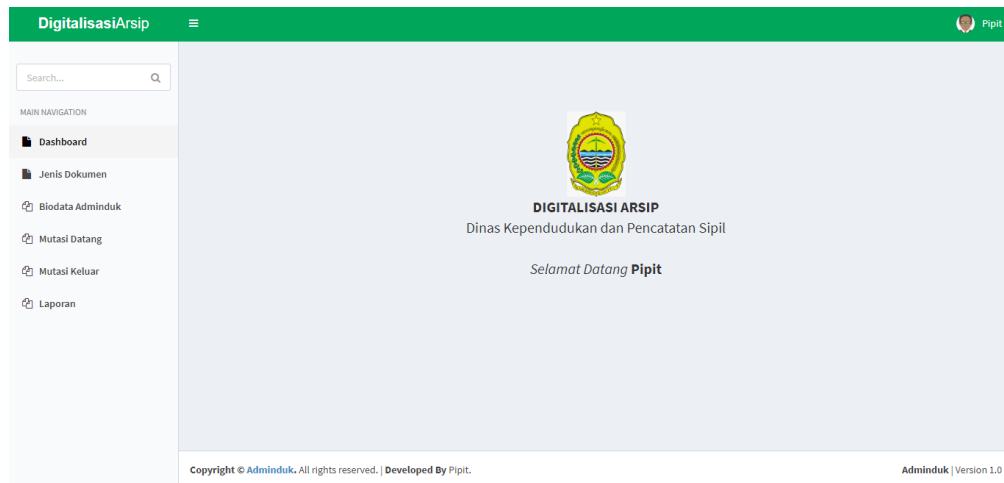
4.2.5 Halaman Hapus User



Gambar 4.37 Tampilan Halaman Hapus *User*

Gambar 4.37 merupakan tampilan halaman hapus *user*. Menghapus *user* dikhususkan pada hak akses sebagai admin. Sebelum menghapus *user* terlebih dahulu masuk pada halaman *user* kemudian memilih user yang akan dihapus dan klik tombol gambar sampah atau hapus, maka akan tampil pesan untuk menghapus *user* kemudian klik ok. Data user yang sudah dihapus akan hilang dari users list, namun data yang dihapus tidak hilang dari database. Pada data yang dihapus akan berstatus 1 (satu).

4.2.6 Halaman Utama User



Gambar 4.38 Tampilan Halaman Utama *User* (Petugas)

Gambar 4.38 merupakan tampilan halaman utama *user* (petugas). Hak akses *user* (petugas) terdapat 5 (lima) menu yang dapat diakses, yaitu menu pertama adalah jenis dokumen, pada menu ini petugas dapat menambah, mengedit dan menghapus jenis dokumen. Menu ini juga terdapat pada hak akses admin artinya menu ini diberikan juga pada hak akses petugas.

Menu kedua pada hak akses petugas adalah menu biodata adminduk yang hanya berfungsi untuk melihat data biodata penduduk Kabupaten Bantul. Menu ini juga terdapat pada hak akses admin.

Menu ketiga adalah menu mutasi datang, pada menu ini petugas dapat mengupload dokumen permohonan mutasi datang yang sudah discan terlebih

dahulu dan disimpan dengan format .pdf. Menu keempat adalah menu mutasi keluar, menu ini memiliki fungsi yang sama dengan menu mutasi datang yang digunakan untuk *upload* data dokumen mutasi datang yang sudah discan terlebih dahulu dan disimpan dengan format .pdf.

Menu kelima mpat adalah menu laporan, menu ini digunakan untuk menampilkan laporan dari pengarsipan digital dari berkas permohonan mutasi keluar dan mutasi datang. Menu ini juga terdapat pada hak akses admin.

4.2.7 Halaman Upload Mutasi Datang

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

Gambar 4.39 Tampilan Halaman Upload Mutasi Datang

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 memasukkan 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.

DigitalisasiArsip

Dokumen Mutasi Datang Kelola Dokumen

Upload Dokumen

NIK: 34029997777888

Data ditemukan 1

Nama Lengkap	Jenis Kelamin
BELLE AUBADINAH	Perempuan
Tempat Lahir	Tanggal Lahir
BANTUL	10-11-1976

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

(*) wajib diisi

Gambar 3.40 Tampilan Halaman *Upload* Mutasi Datang Kedua

Gambar 3.40 merupakan tampilan halaman *upload* dokumen permohonan mutasi datang yang kedua dimana perbedaan pada menu ini adalah setelah menginputkan NIK dan klik tombol cari untuk menampilkan data penduduk, kemudian tombol *submit* untuk menyimpan data akan tampil. Selanjutnya petugas

mengupload dokumen permohonan yang sudah discan. Data harus dengan format .pdf, jika diupload dengan format data *image* tidak dapat tersimpan.

4.2.8 Halaman Upload Mutasi Keluar

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

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.

Adminduk : Upload Dokumen x +

localhost/cipit/aksi_upload/keluar

Paused

DigitalsiArsip

Dokumen Mutasi Keluar Kelola Dokumen

Upload Dokumen

NIK 34029997777888 Q Cari

Data ditemukan 1

Nama Lengkap BELLE JUBADYAH Jenis Kelamin Perempuan

Tempat Lahir BANTUL Tanggal Lahir 10-11-1976

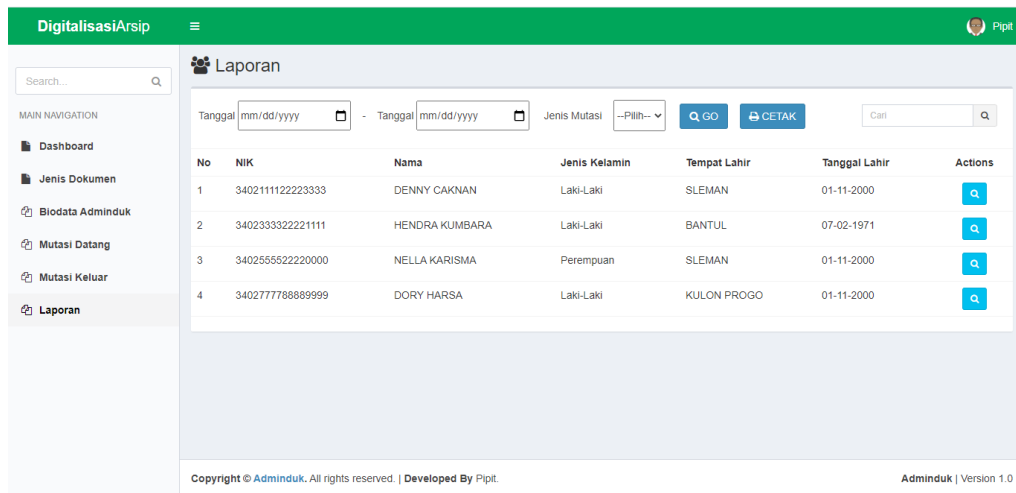
No	Jenis	Upload	Dokumen	Status
1	Blangko Permohonan (*)	Choose File No file chosen	Kosong	Tidak
2	Surat Keterangan Pindah WNI (*)	Choose File No file chosen	Kosong	Tidak
3	Kartu Tanda Penduduk (*)	Choose File No file chosen	Kosong	Tidak
4	Kartu Keluarga (*)	Choose File No file chosen	Kosong	Tidak
5	Biodata WNI (*)	Choose File No file chosen	Kosong	Tidak
6	Bukti Pendukung Lainnya (*) wajib diisi	Choose File No file chosen	Kosong	Tidak

Submit

Gambar 4.41 Tampilan Halaman *Upload* Mutasi Keluar Kedua

Gambar 4.41 merupakan tampilan halaman *upload* dokumen permohonan mutasi keluar yang kedua dimana perbedaan pada menu ini sama seperti yang sudah dijelaskan pada tampilan halaman *upload* mutasi keluar adalah setelah menginputkan NIK dan klik tombol cari untuk menampilkan data penduduk, maka tombol *submit* untuk menyimpan data akan tampil. Selanjutnya petugas mengupload dokumen permohonan yang sudah discan. Data harus dengan format .pdf, jika diupload dengan format data *image* tidak dapat tersimpan

4.2.9 Halaman Menu Laporan



Gambar 4.42 Tampilan Halaman Menu Laporan

Gambar 4.42 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.

Laporan dapat disajikan untuk semua jenis mutasi baik mutasi keluar maupun mutasi datang. Selain itu laporan juga dapat disajikan per jenis mutasi. Untuk tampilan laporan adalah senagai berikut :

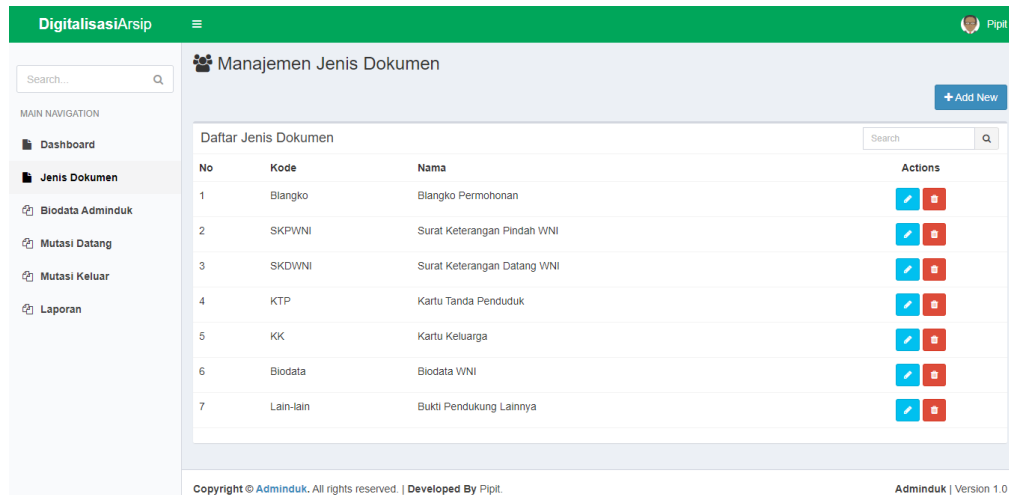
LAPORAN MUTASI Tanggal 01-07-2020 s/d 31-07-2020					
No.	NIK	Nama	Jenis Kelamin	Tempat Lahir	Tanggal Lahir
1	340211112223333	DENNY CAKNAN	Laki-Laki	SLEMAN	01-11-2000
2	340233332221111	HENDRA KUMBARA	Laki-Laki	BANTUL	07-02-1971
3	340255552220000	NELLA KARISMA	Perempuan	SLEMAN	01-11-2000
4	3402777788889999	DORY HARSA	Laki-Laki	KULON PROGO	01-11-2000

Gambar 4.43 Tampilan Laporan Data Mutasi Yang Sudah Terarsip Digital

LAPORAN MUTASI Tanggal 01-07-2020 s/d 31-07-2020					
Jenis Mutasi : Datang					
No.	NIK	Nama	Jenis Kelamin	Tempat Lahir	Tanggal Lahir
1	340211112223333	DENNY CAKNAN	Laki-Laki	SLEMAN	01-11-2000
2	340233332221111	HENDRA KUMBARA	Laki-Laki	BANTUL	07-02-1971

Gambar 4.44 Tampilan Laporan Per Jenis Mutasi

4.2.10 Halaman Jenis Dokumen



Gambar 4.45 Tampilan Halaman Jenis Dokumen

Gambar 4.45 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, Jenis dokumen ini menyesuaikan dengan kondisi sebenarnya di unit pelayanan Pindah Datang penduduk. Regulasi administrasi kependudukan kadang berubah menyesuaikan dengan keadaan yang sedang terjadi. Sehingga jenis dokumen juga harus dapat menyesuaikan dengan perubahan regulasi tersebut. Untuk mempercepat dalam penyesuaian regulasi untuk itu menu ini dapat diakses pada hak akses admin maupun hak akses petugas.

4.2.11 Halaman Tambah Jenis Dokumen

The screenshot shows a web application interface for managing document types. At the top, there is a green navigation bar with a user profile icon labeled 'Pipit'. Below this, the page title 'Manajemen Jenis Dokumen' is displayed, followed by the sub-header 'Tambah / Ubah Jenis Dokumen'. The main content area is a light blue box containing a white form titled 'Jenis Dokumen'. This form has two input fields: 'Kode' and 'Nama'. Below these fields are two buttons: 'Submit' (in blue) and 'Reset' (in grey). At the bottom of the page, there is a footer with the text 'Copyright © Adminduk. All rights reserved. | Developed By Pipit.' on the left and 'Adminduk | Version 1.0' on the right.

Gambar 4.46 Tampilan Halaman Tambah Jenis Dokumen

Gambar 4.46 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. Selanjutnya jenis dokumen yang baru akan ditambahkan pada daftar jenis dokumen dan akan tampil pada menu upload mutasi keluar maupun mutas datang.

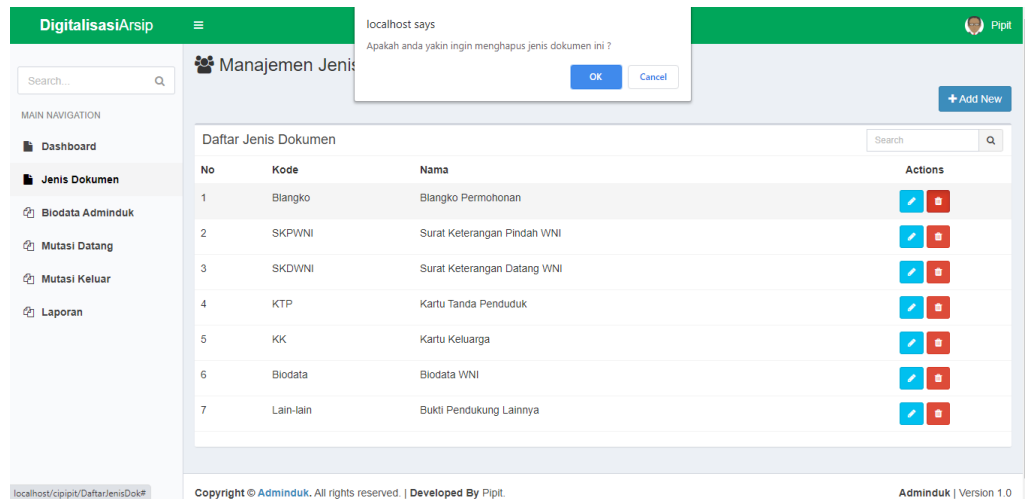
4.2.12 Halaman Edit Jenis Dokumen

The screenshot shows a web application interface for managing document types. The header is green and contains a menu icon on the left and a user profile 'Pipit' on the right. The main content area is light blue. A white form titled 'Jenis Dokumen' is centered. It has two input fields: 'Kode' with the value 'Blangko' and 'Nama' with the value 'Blangko Permohonan'. Below the fields are 'Submit' and 'Reset' buttons. The footer contains copyright information for Adminduk and version 1.0.

Gambar 4.47 Tampilan Halaman Edit Jenis Dokumen

Gambar 4.47 merupakan tampilan halaman edit jenis dokumen. Jika jenis dokumen dirasa kurang sesuai dengan dokumen pindah datang karena perubahan regulasi maupun perubahan nama dokumen, maka jenis dokumen dapat diupdate sesuai dengan regulasi atau keadaan pada saat itu. Admin maupun user (petugas) hanya tinggal mengubah kode dan nama sesuai data yang baru. Kemudian klik tombol *submit*, maka data jenis dokumen yang baru akan tersimpan

4.2.13 Halaman Hapus Jenis Dokumen



Gambar 4.48 Tampilan Halaman Hapus Jenis Dokumen

Gambar 4.48 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.

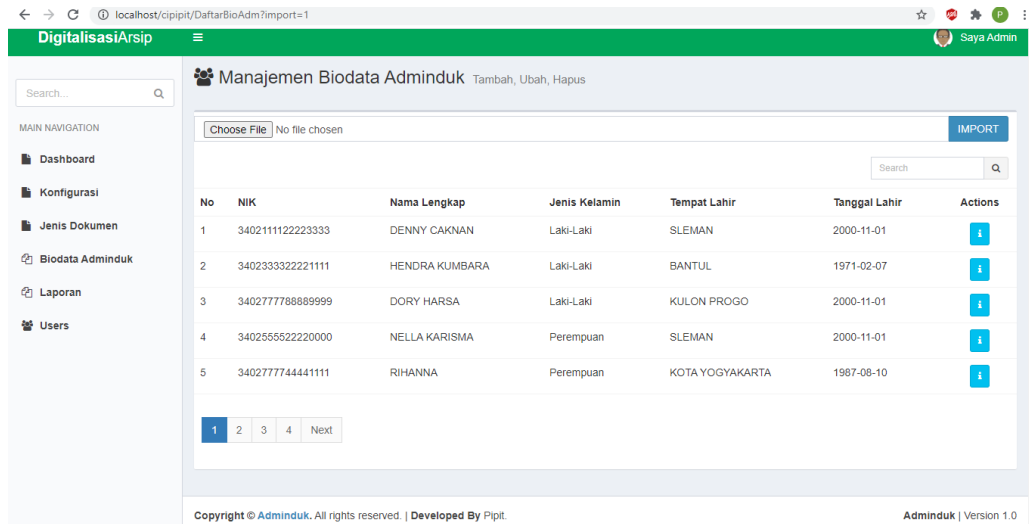
4.2.14 Halaman Menu Biodata Adminduk

No	NIK	Nama Lengkap	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Actions
1	3402111122223333	DENNY CAKNAN	Laki-Laki	SLEMAN	2000-11-01	i
2	3402333322221111	HENDRA KUMBARA	Laki-Laki	BANTUL	1971-02-07	i
3	3402777788889999	DORY HARSA	Laki-Laki	KULON PROGO	2000-11-01	i
4	3402555522220000	NELLA KARISMA	Perempuan	SLEMAN	2000-11-01	i
5	3402777744441111	RIHANNA	Perempuan	KOTA YOGYAKARTA	1987-08-10	i

Gambar 4.49 Tampilan Halaman Menu Biodata Adminduk

Gambar 4.49 merupakan tampilan halaman menu biodata adminduk yang digunakan untuk melihat data penduduk Kabupaten Bantul. Pada menu ini tidak ada menu tambah, hapus dan edit. Hal tersebut dikarenakan data penduduk tidak diperbolehkan ditambah, diubah maupun dihapus pada sistem bantu. Data penduduk sudah memiliki sistem resmi yang digunakan untuk menambah, mengubah dan menghapus sesuai aturan perundang-undangan yang berlaku.

Untuk itu pada sistem bantu ini hanya dapat mengimport data dari database penduduk di Dinas Kependudukan dan Pencatatan Sipil Kabupaten Bantul tanpa mengubah data sedikitpun. Format file yang dapat diimport dengan menggunakan format .xls. Cara untuk mengimport data pada sistem bantu ini adalah sebagai berikut:



Gambar 4.50 Tampilan *Impor* Data

Gambar 4.50 merupakan tampilan *impor* data kependudukan berupa .xls. Cara untuk mengimpor data adalah pada alamat url ditambah dengan ?impor=1, maka akan tampil menu untuk mengimpor data seperti gambar 4.50