

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Pengumpulan Data

Hasil pengumpulan data pada penelitian ini diperoleh dari kegiatan observasi dan wawancara kepada pihak yang bersangkutan dengan kantor Notaris dan PPAT Yoan Budiyanto, S.H., M.Kn.

4.1.1 Hasil observasi

Berdasarkan observasi yang dilakukan di Notaris dan PPAT Yoan Budiyanto, S.H., M.Kn menghasilkan bahwa arsip akta disimpan dalam bentuk *hardcopy* dan *softcopy*, sedangkan arsip laporan PPAT disimpan dalam bentuk *softcopy*. Arsip akta dalam bentuk *hardcopy* disimpan dalam lemari arsip yang dibendel menjadi sebuah buku, sedangkan arsip akta dalam bentuk *softcopy* disimpan pada komputer, dalam folder-folder sesuai dengan tahun dan bulan disahkannya akta. Laporan-laporan akta PPAT yang berupa *softcopy* disimpan pada komputer dalam bentuk folder pertahun. Selain observasi terhadap pengarsipan akta dan laporan, pengamatan proses pencatatan akta juga dilakukan. Pencatatan akta masuk dilakukan dengan ditulis ke dalam sebuah buku yang akan direkap menjadi laporan Notaris maupun laporan PPAT kepada BPN.

4.1.2 Hasil wawancara

Adapun hasil wawancara yang dilakukan pada tanggal 17 maret 2021 pada karyawan bagian PPAT menghasilkan bahwa pengarsipan dilakukan dengan menyimpan data akta dalam bentuk *hardcopy* ke dalam lemari arsip dan *softcopy* kedalam komputer. Media yang digunakan dalam pengarsipan yaitu buku untuk data *hardcopy* dan komputer sebagai penyimpanan data akta *softcopy*. Hal ini dapat menyebabkan kehilangan dan kerusakan data, selain itu pencarian data yang diperlukan relatif lebih lama. Dokumen yang diarsipkan berupa akta, data client, dan laporan Notaris dan PPAT. Pada proses pembuatan laporan surat akta, dilakukan dengan mengetik data-data yang didapatkan dari berkas persyaratan *client* ke dalam *Microsoft Word*, sedangkan laporan PPAT dilakukan dengan menulis ke dalam buku, selanjutnya diketik pada *Microsoft Excel* lalu diprint dan

discan, kemudian laporan dikirim ke Badan Pertanahan Nasional berupa *softcopy* dan *hardcopy*. Hal ini menyebabkan proses pembuatan laporan berlangsung lama dan sering terjadi kesalahan penulisan laporan. Pada kantor Notaris dan PPAT Yoan Budiyanto hanya membuat 5 jenis laporan dari 8 jenis laporan, yaitu Akta Jual Beli, Akta Hibah, Akta Pemberian Hak Tanggungan, Akta Pembagian Hak Bersama, dan Surat Kuasa Membebaskan Hak Tanggungan.

4.2 Hasil Analisis dan Desain

4.2.1 Analisis Sistem Lama

Pada proses pembuatan surat akta PPAT, karyawan membuat surat akta dengan mengetik data-data *client* yang didapatkan dari berkas persyaratan pengajuan pembuatan akta yang diberikan *client*. Jika akta sudah disahkan maka akta tersebut dimasukkan ke dalam catatan akta masuk. Akta masuk ditulis ke dalam sebuah buku Reportarium PPAT, pencatatan ini dilakukan oleh karyawan bagian PPAT. Hasil pencatatan tersebut akan direkap menjadi laporan yang ditujukan kepada Badan Pertanahan Nasional (BPN). Laporan kepada BPN dikirim dalam bentuk *hardcopy* dan *softcopy*. Akta masuk yang telah dicatat dalam buku Reportarium PPAT akan diketik pada *Microsoft Excel* kemudian diprint lalu discan. Laporan berupa *softcopy* dikirim ke BPN dengan menggunakan *email*. Sedangkan laporan yang berupa *hardcopy* dikirim langsung ke kantor BPN. Setelah akta selesai disahkan, akta-akta tersebut akan diarsipkan. Pengarsipan dilakukan dengan dua metode, yang pertama arsip akta berupa *hardcopy* yang disimpan dalam bentuk buku di lemari arsip, buku arsip tersebut berisi arsip akta dalam satu bulan. Metode yang kedua yaitu arsip akta disimpan dalam bentuk *softcopy* didalam komputer, arsip tersebut disimpan didalam folder arsip perbulan. Arsip akta tersebut, baik berupa *hardcopy* atau *softcopy* disimpan dan menjadi tanggung jawab Notaris dan PPAT selama 30 tahun.

4.2.2 Analisis Sistem Baru

Aplikasi ini mampu melakukan pencatatan akta masuk sehingga akta masuk tidak perlu ditulis ke dalam sebuah buku. Hal ini juga dapat memudahkan proses pembuatan laporan surat akta dan laporan akta masuk. Dimana karyawan dapat mengerjakan dua laporan tersebut dalam sekali proses. Surat akta yang telah

dibuat, secara otomatis akan masuk ke dalam catatan akta masuk, pada halaman akta masuk, terdapat fitur cetak laporan, sehingga untuk membuat laporan akta masuk, karyawan dapat menggunakan fitur tersebut. Selain itu, aplikasi ini mampu melakukan pengarsipan akta dengan mudah dan aman mengingat arsip akta akan menjadi tanggung jawab Notaris dan PPAT dalam waktu 30 tahun.

4.2.3 Desain Sistem

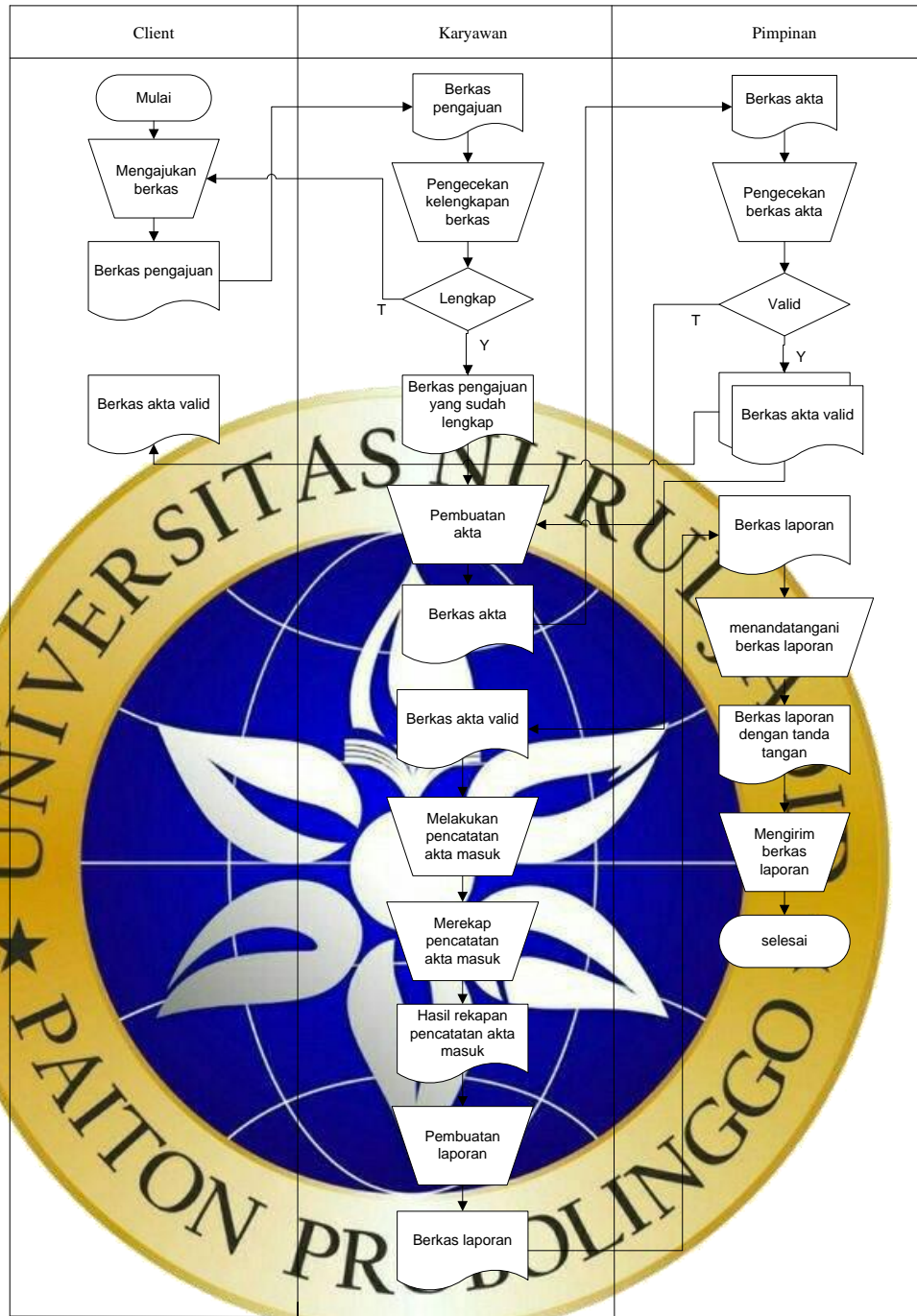
Pada penelitian ini desain sistem dilakukan dengan perancangan bagan alur (*flowchart*), *Data Flow Diagram* (DFD), Entity Relationship Diagram (ERD), serta dilengkapi dengan desain database dan desain program.

1. Bagan Alur (*flowchart*)

Flowchart sistem baru dan sistem lama pada pengarsipan akta dan pembuatan laporan PPAT adalah sebagai berikut:

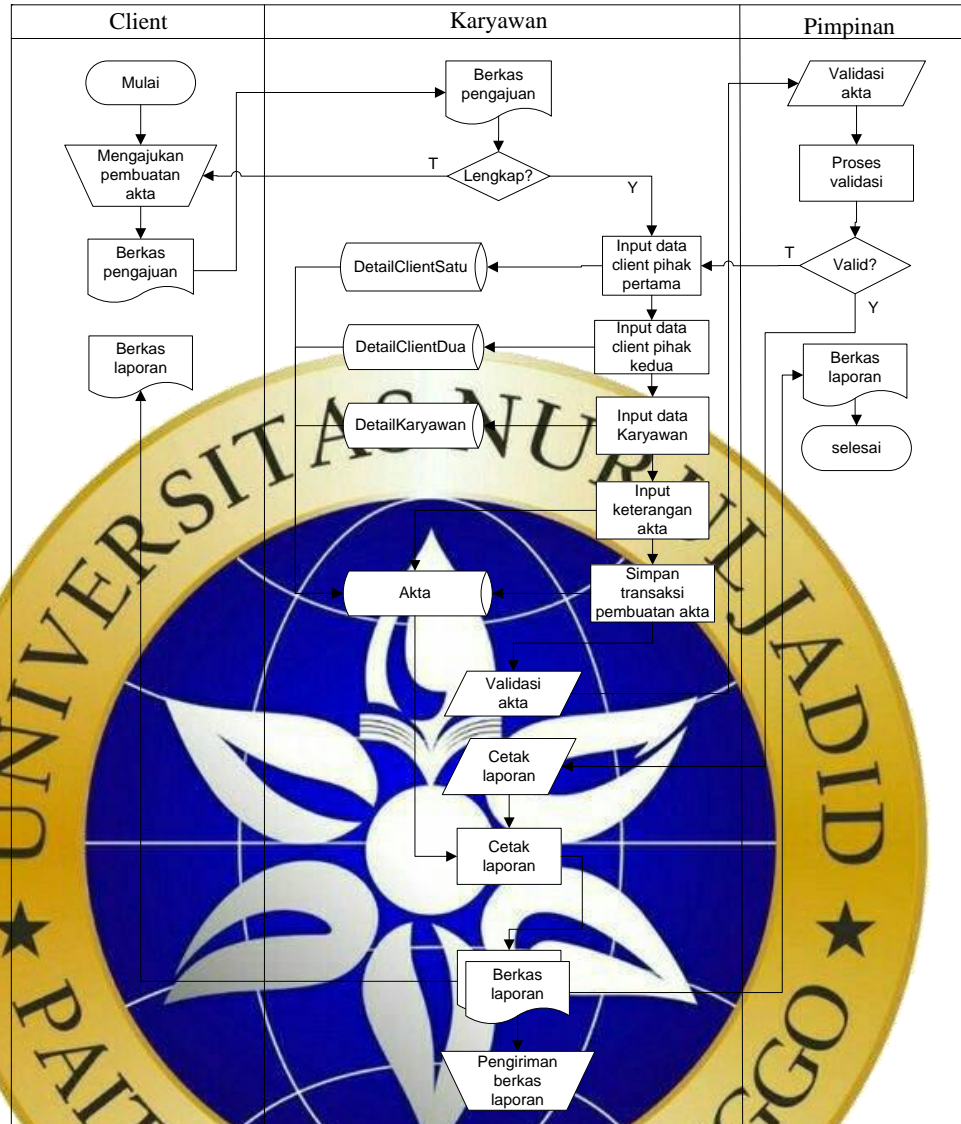
- a. *Flowchart* sistem lama





Gambar 4.1 *Flowchart* Sistem Lama

b. *Flowchart sistem baru*



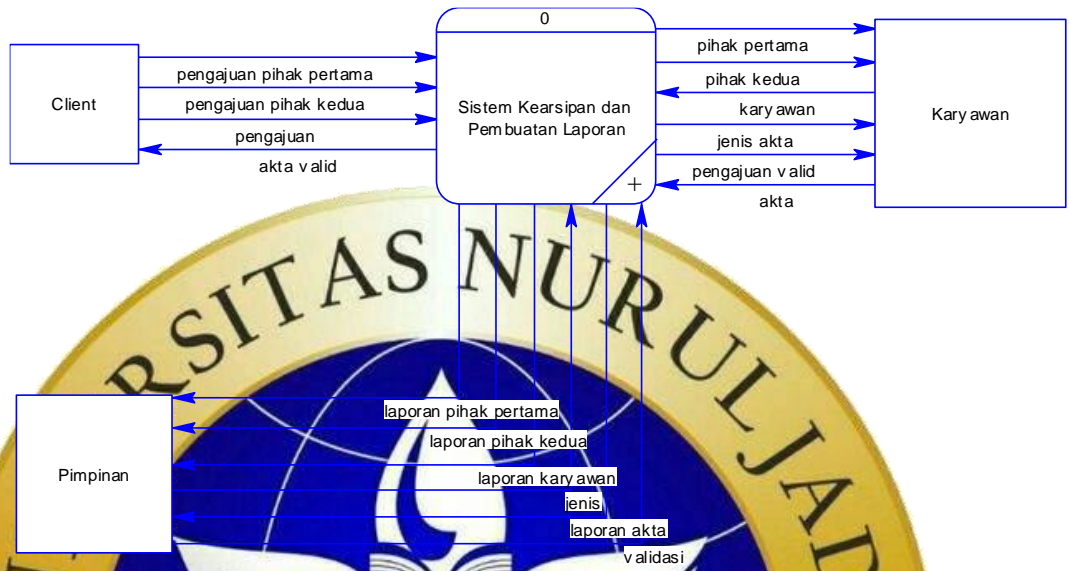
Gambar 4. 2 Flowchart Sistem Baru

2. *Data Flow Diagram (DFD)*

Berikut adalah gambaran DFD pada sistem kearsipan dan pembuatan laporan PPAT.

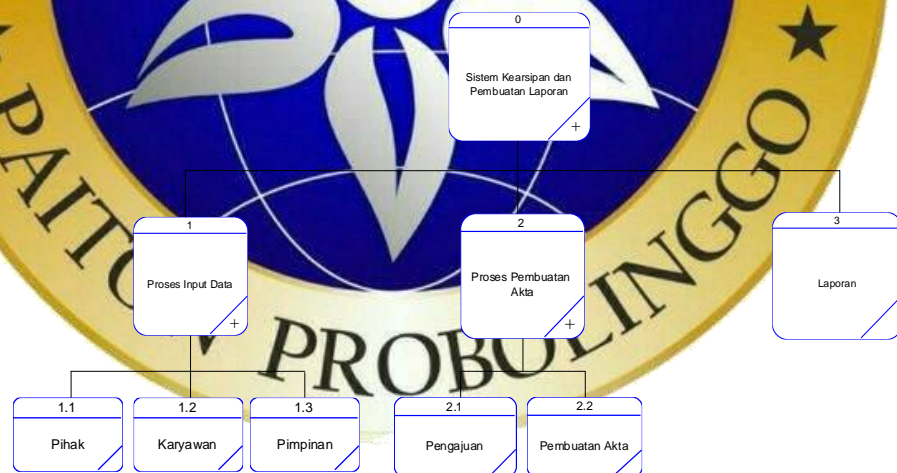
a. Diagram context

Berikut merupakan diagram context pada sistem kearsipan dan pembuatan laporan PPAT yang menghubungkan sistem informasi dengan entitas secara global.



Gambar 4. 3 Diagram Context

b. Bagan berjenjang

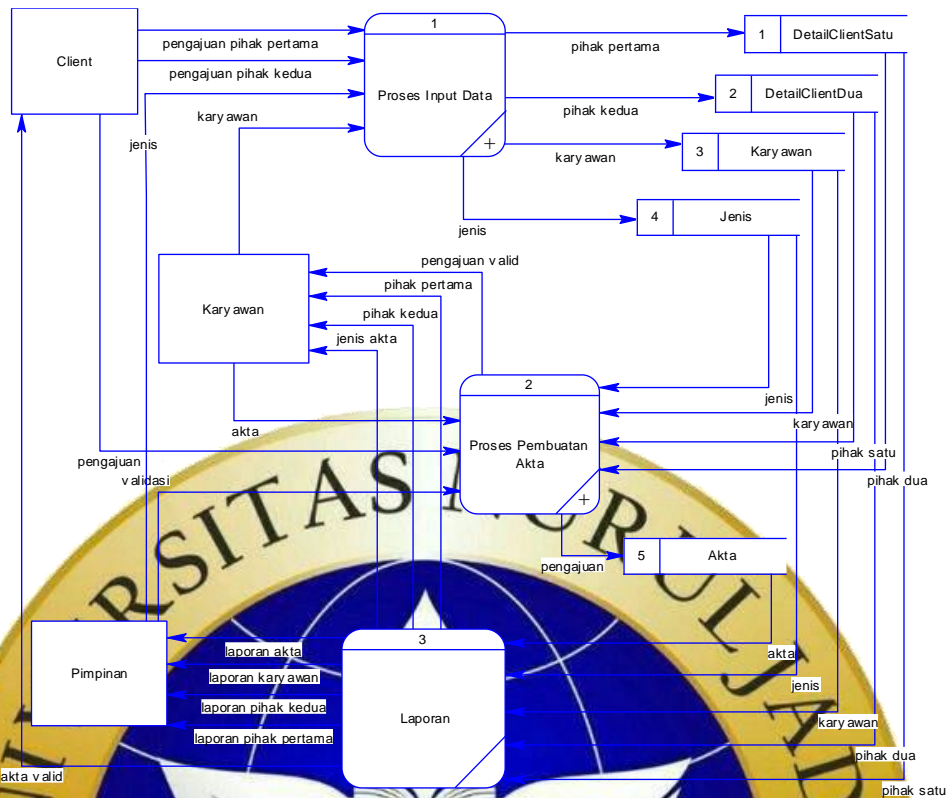


Gambar 4. 4 Bagan Berjenjang

c. Data Flow Diagram

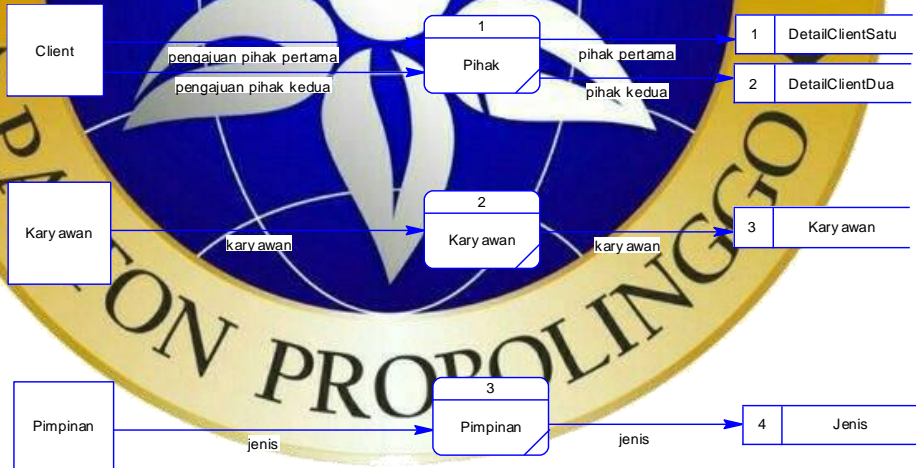
1) DFD level 1

DFD level 1 akan menjelaskan secara lebih rinci context diaram sistem kearsipan akta ke dalam beberapa proses sesuai dengan kebutuhan sistem.



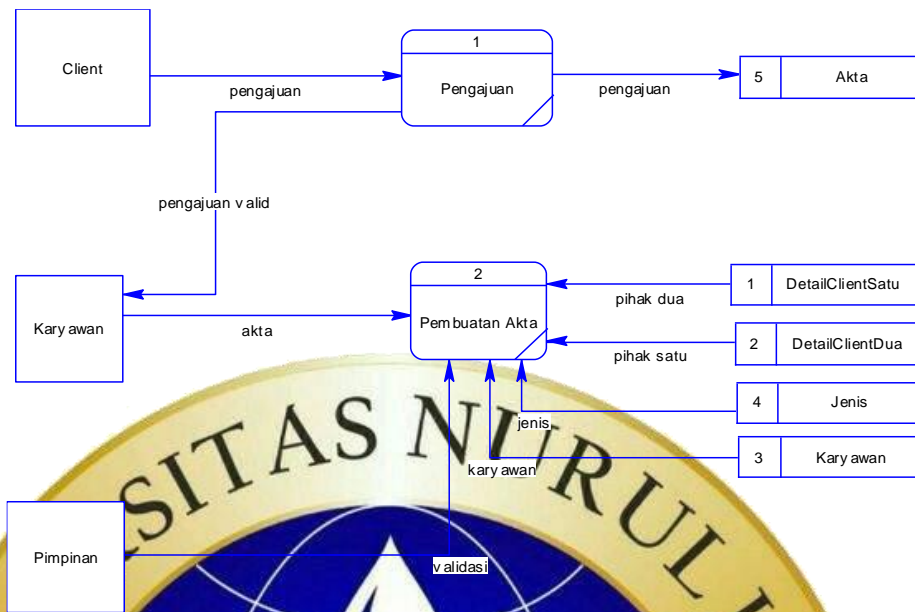
Gambar 4.5 DFD Level 1

2) DFD level 2 proses 1 input data



Gambar 4.6 DFD Level 2 Proses 1 Input Data

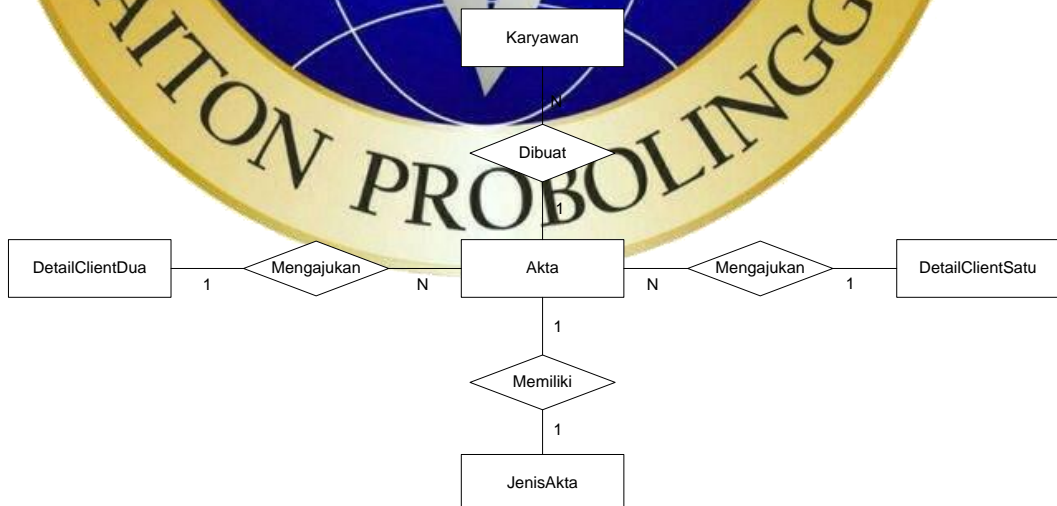
3) DFD level 2 proses 2 pembuatan akta



Gambar 4. 7 DFD Level 2 Proses 1 Pembuatan Akta Data

3. Entity Relationship Diagram (ERD)

Untuk menentukan hubungan antar entitas yang membentuk sistem ini diperlukan gambaran yang sistematis untuk memudahkan dalam bentuk tabel data. Dengan menggunakan ERD dapat dipahami dengan mudah relasi antar entitas one to one, one to many, dan many to many secara jelas.



Gambar 4. 8 ERD Sistem Kearsipan Akta

4. Kamus data

Karyawan : user, nik_karyawan, nama_karyawan, tempat_lahir_karyawan, tanggal_lahir_karyawan, uraian_tl_karyawan, status_kewarganegaraan_karyawan, prov_karyawan, kab_karyawan, almt_karyawan, rt_karyawan, desa_karyawan, kec_karyawan, telp_karyawan, jenkel_karyawan, jabatan, profil_pic.

DetailKaryawan : user, no_transaksi, nik_karyawan.

Jenis : id_jenis, jenis, keterangan_jenis.

DetailClientSatu : user, no_transaksi, nik_pertama, nama_pertama, tempat_lahir_pertama, tanggal_lahir_pertama, uraian_tl_pertama, pekerjaan_pertama, status_kewarganegaraan_pertama, prov_pertama, kab_pertama, alamat_pertama, rt_pertama, rw_pertama, desa_pertama, kec_pertama.

DetailClientDua : user, no_transaksi, nik_kedua, nama_kedua, tempat_lahir_kedua, tanggal_lahir_kedua, uraian_tl_kedua, pekerjaan_kedua, status_kewarganegaraan_kedua, pov_kedua, kab_kedua, alamat_kedua, rt_kedua, rw_kedua, desa_kedua, kec_kedua.

Akta : user, no_transaksi, nomor_akta, tgl_akta, jenis, nomor_hak, prov_tanah, kab_tanah, desa_tanah, luas_tanah, luas_bgn, harga_pengalihan, nop, njop, nib, tgl_ssp, harga_ssp, tgl_sob, harga_sbb, keterangan, pt, jabatan_pt, jenis_kuasa, keterangan_kuasa, jenis_bukti_perjanjian, no_bukti_ppat, jumlah_perjanjian, tanggal_perjanjian, no_perjanjian, nik_saksi, nama_saksi, tempat_lahir_saksi, tanggal_lahir_saksi, uraian_tl_saksi, pekerjaan_saksi, status_kewarganegaraan_saksi, prov_saksi, kab_saksi, alamat_saksi, rt_saksi, rw_saksi, desa_saksi, kec_saksi, status_saksi.

5. Desain Database

Tabel 4. 1 Karyawan

No	Nama	Jenis
1	user	OneToOneField
2	nik_karyawan	CharField(50)
3	nama_karyawan	CharField(200)
4	tempat_lahir_karyawan	CharField(200)
5	tanggal_lahir_karyawan	CharField(200)
6	uraian_tl_karyawan	CharField(200)
7	status_kewarganegaraan_karyawan	Char('Warga Negara Indonesia', 'Warga Negara Asing')
8	prov_karyawan	CharField(200)
9	kab_karyawan	CharField(200)
10	almt_karyawan	CharField(100)
11	rt_karyawan	CharField(50)
12	rw_karyawan	CharField(50)
13	desa_karyawan	CharField(200)
14	kec_karyawan	CharField(200)
15	telp_karyawan	CharField(100)
16	Jenkel_karyawan	Enum ('Laki-laki', 'Perempuan')
17	jabatan	Enum ('PPAT', 'Karyawan PPAT', 'Karyawan Notaris')
18	profil_pic	ImageField

Tabel 4. 2 DetailKaryawan

No	Nama	Jenis
1	No_transaksi	BigIntegerField
2	User	ForeignKey
3	Nik_karyawan	ForeignKey

Tabel 4. 3 Jenis

No	Nama	Jenis
1	id_jenis	CharField(50)
2	jenis	CharField(100)
3	keterangan_jenis	CharField(100)

Tabel 4. 4 DetailClientSatu

No	Nama	Jenis
1	user	ForeignKey
2	no_transaksi	BigIntegerField
3	nik_pertama	CharField(50)
4	nama_pertama	CharField(200)
5	tempat_lahir_pertama	CharField(200)
6	tanggal_lahir_pertama	CharField(200)
7	uraian_tl_pertama	CharField(200)
8	pekerjaan_pertama	CharField(200)
9	status_kewarganegaraan_pertama	Enum ('Warga Negara Indonesia', 'Warga Negara Asing')
10	prov_pertama	CharField(200)
11	kab_pertama	CharField(200)
12	alamat_pertama	CharField(200)
13	rt_pertama	CharField(200)
14	rw_pertama	CharField(200)
15	desa_pertama	CharField(200)
16	kec_pertama	CharField(200)

Tabel 4. 5 DetailClientDua

No	Nama	Jenis
1	user	ForeignKey
2	no_transaksi	BigIntegerField

Tabel 4.5 DetailClientDua (lanjutan)

No	Nama	Jenis
3	nik_kedua	CharField(50)
4	nama_kedua	CharField(200)
5	tempat_lahir_kedua	CharField(200)
6	tanggal_lahir_kedua	CharField(200)
7	uraian_tl_kedua	CharField(200)
8	pekerjaan_kedua	CharField(200)
9	status_kewarganegaraan_kedua	Enum ('Warga Negara Indonesia', 'Warga Negara Asing')
10	prov_kedua	CharField(200)

Tabel 4.6 Akta

No	Nama	Jenis
1	user	ForeignKey
2	No_transaksi	BigIntegerField
3	nomor_akta	CharField(200)
4	tgl_akta	CharField(200)
5	jenis	Enum ('AJB', 'APHT', 'SKMHT', 'AH', 'APHB')
6	nomor_hak	CharField(200)
7	prov_tanah	CharField(200)
8	kab_tanah	CharField(200)
9	kec_tanah	CharField(200)
10	desa_tanah	CharField(200)
11	luas_tanah	CharField(50)
12	luas_bgn	CharField(50)
13	harga_pengalihan	CharField(50)
14	nop	CharField(50)

Tabel 4.6 Akta (lanjutan)

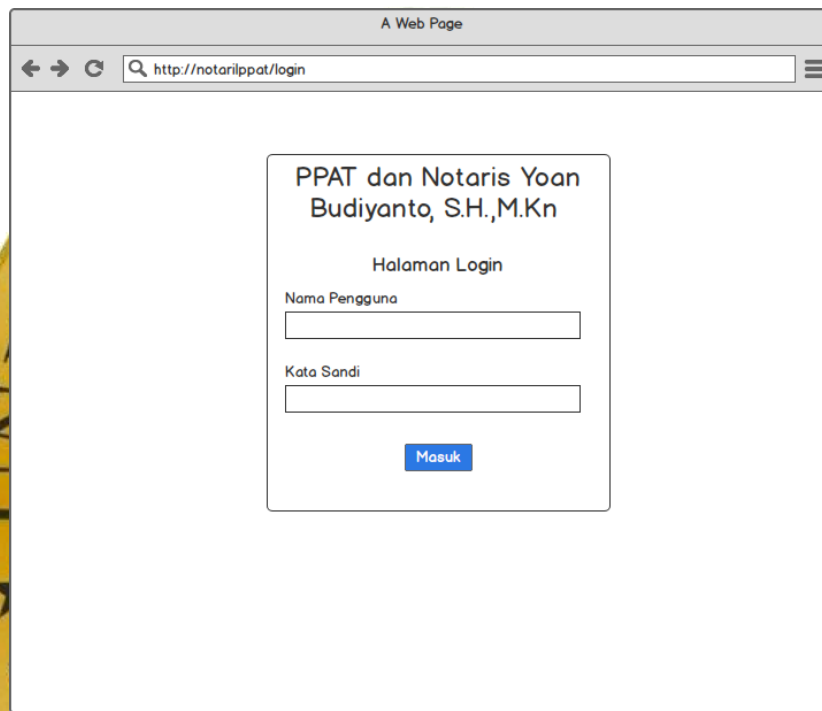
No	Nama	Jenis
15	njop	CharField(50)
16	nib	CharField(50)
17	tgl_ssp	CharField(50)
20	harga_sbb	CharField(50)
21	keterangan	CharField(200)
22	pt	CharField(200)
23	jabatan_pt	CharField(200)
24	jenis_kuasa	CharField(200)
25	keterangan_kuasa	CharField(500)
26	jenis_bukti_perjanjian	CharField(200)
27	no_bukti_ppat	CharField(200)
28	jumlah_perjanjian	CharField(200)
29	tanggal_perjanjian	CharField(50)
30	no_perjanjian	CharField(50)
31	nik_saksi	CharField(50)
32	nama_saksi	CharField(200)
33	tempat_lahir_saksi	CharField(200)
34	tanggal_lahir_saksi	CharField(200)
35	uraian_ft_saksi	CharField(200)
36	pekerjaan_saksi	CharField(200)
37	status_kewarganegaraan_saksi	Enum ('Warga Negara Indonesia', 'Warga Negara Asing')
38	prov_saksi	CharField(200)
39	kab_saksi	CharField(200)
40	alamat_saksi	CharField(200)
41	rt_saksi	CharField(50)
42	rw_saksi	CharField(50)
43	desa_saksi	CharField(200)

Tabel 4.6 Akta (Lanjutan)

No	Nama	Jenis
44	kec_saksi	CharField(200)
45	status_saksi	CharField(200)

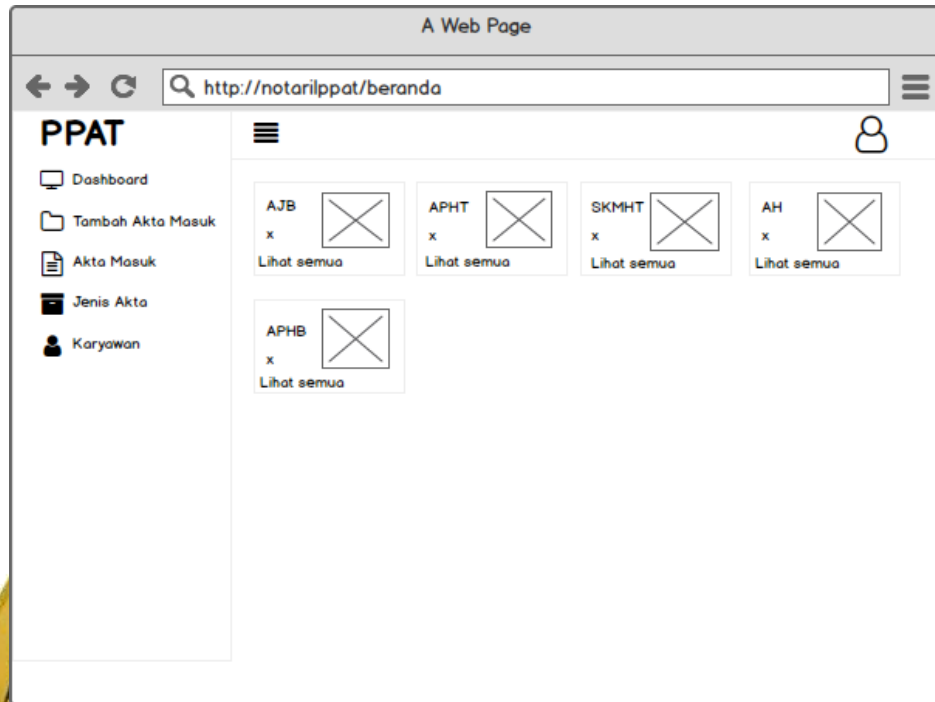
6. Perancangan User Interface

a. Desain halaman login



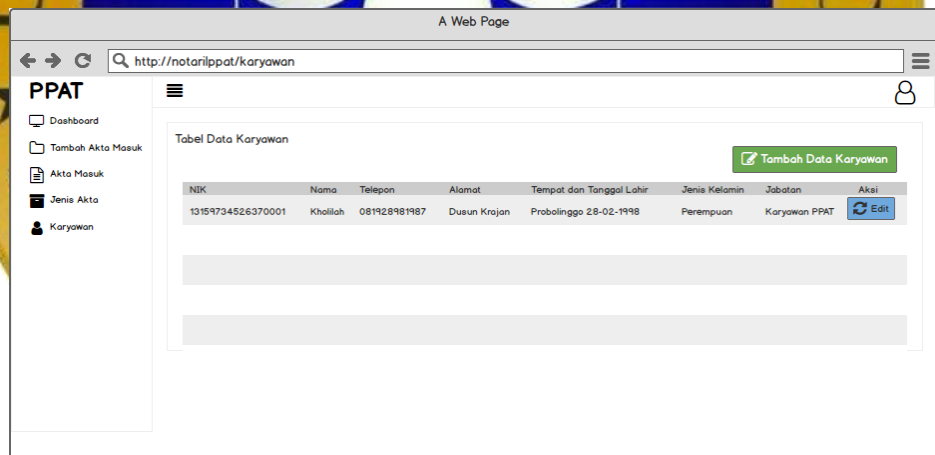
Gambar 4.9 Halaman Login

b. Halaman beranda



Gambar 4. 10 Halaman Beranda

c. Desain halaman karyawan



Gambar 4. 11 Halaman karyawan

d. Desain halaman tambah karyawan

A Web Page

http://notarilppat/tambahkaryawan

PPAT

- Dashboard
- Tambah Akta Masuk
- Akta Masuk
- Jenis Akta
- Karyawan

Tambah Karyawan

ID Karyawan

NIK Karyawan

Nama

Tempat Lahir

Tanggal Lahir

Uraian Tanggal Lahir

Status Kewarganegaraan

Provinsi

Kabupaten / Kota

Alamat

RT

RW

Desa / Kelurahan

Kecamatan

Telp Karyawan

Jenis Kelamin

Jabatan

Username

Password

Ulangi Password

Gambar 4. 12 Halaman Tambah Karyawan

e. Desain halaman edit karyawan

A Web Page

← → ↻ 🔍 http://notarilppat/editkaryawan

PPAT

- Dashboard
- Tambah Akta Masuk
- Akta Masuk
- Jenis Akta
- Karyawan

Edit Karyawan

ID Karyawan

NIK Karyawan

Nama

Tempat Lahir

Tanggal Lahir

Uraian Tanggal Lahir

Status Kewarganegaraan

Provinsi

Kabupaten / Kota

Alamat

RT

RW

Desa / Kelurahan

Kecamatan

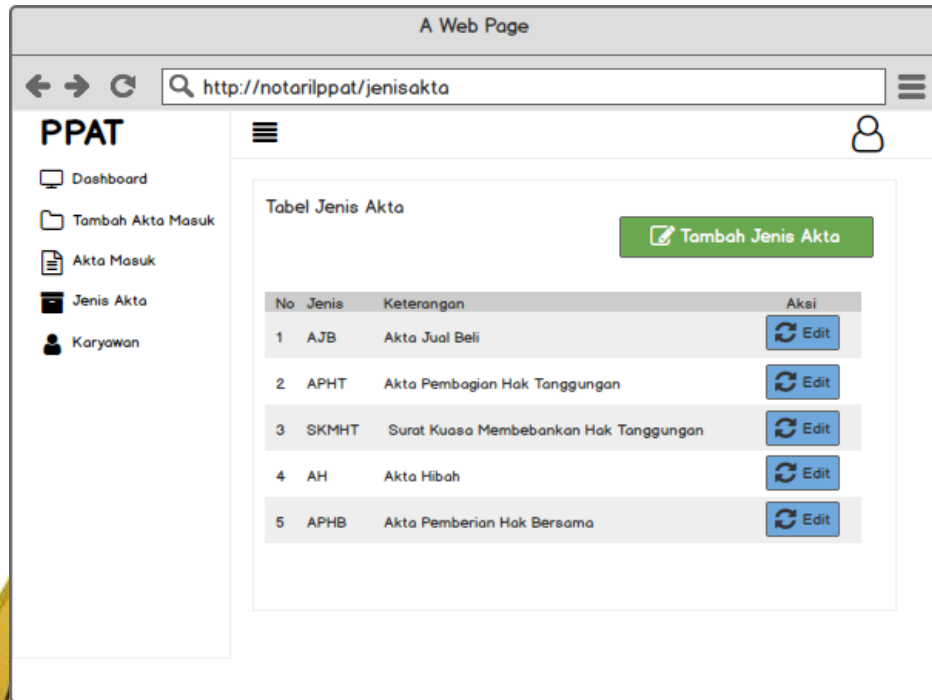
Telp Karyawan

Jenis Kelamin

Jabatan

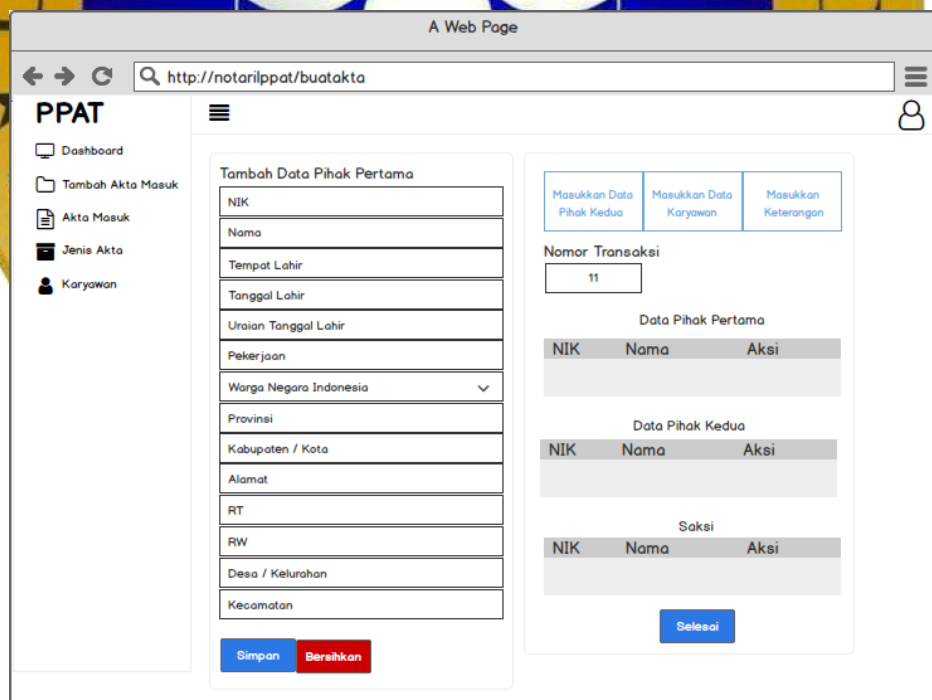
Gambar 4. 13 Halaman Edit Karyawan

f. Desain halaman jenis akta



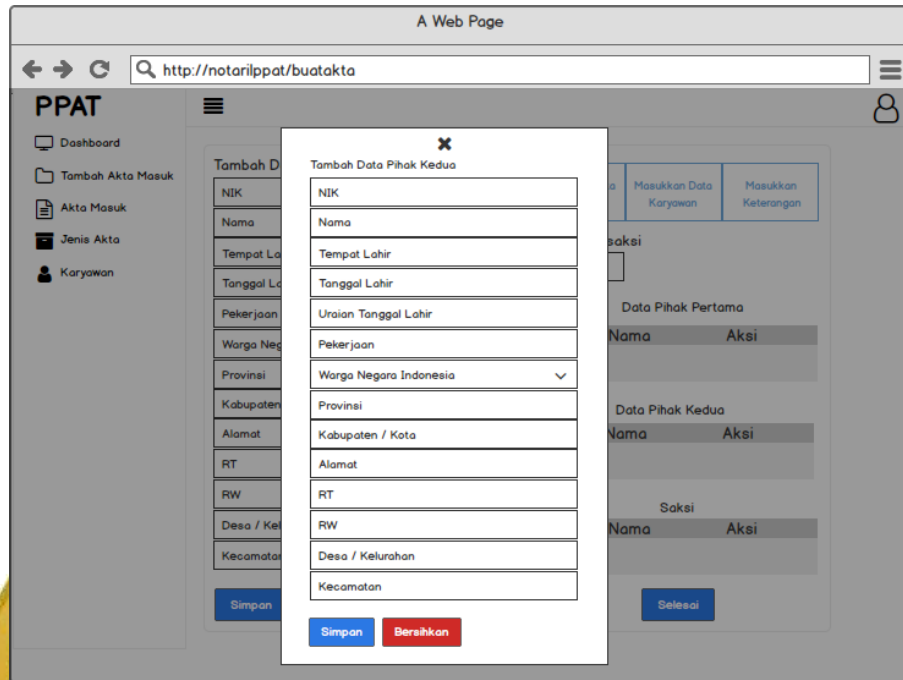
Gambar 4. 14 Halaman Jenis Akta

g. Desain halaman tambah data akta



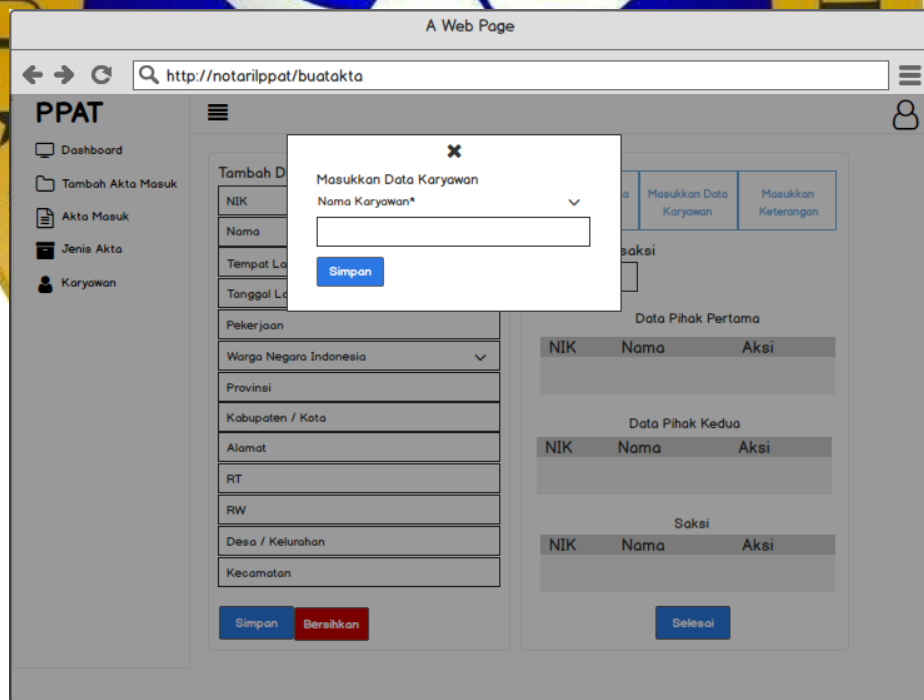
Gambar 4. 15 Halaman Tambah Data Akta

h. Desain halaman tambah data akta untuk menambah data pihak kedua



Gambar 4. 16 Halaman Tambah Data Pihak Kedua

i. Desain halaman tambah data akta untuk memilih data karyawan



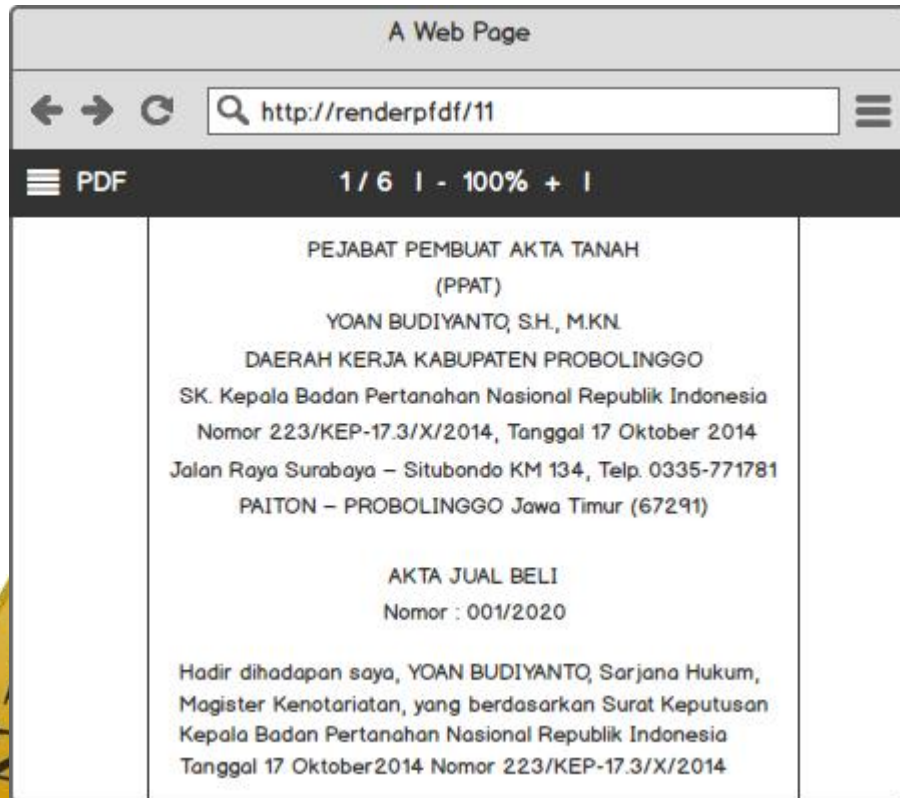
Gambar 4. 17 Halaman Tambah Data Karyawan

j. Desain halaman tambah data akta untuk menambah keterangan akta

The image shows a web application interface for adding deed details. The main page is titled "PPAT" and has a sidebar with navigation options: Dashboard, Tambah Akta Masuk, Akta Masuk, Jenis Akta, and Karyawan. The central area is titled "Tambah Data" and contains a form with fields for NIK, Nama, Tempat Lahir, Tanggal Lahir, Pekerjaan, Warga Negara, Provinsi, Kabupaten / Kota, Alamat, RT, RW, Desa / Kelurahan, and Kecamatan. A "Simpan" button is at the bottom of this form. A modal window titled "Masukkan Keterangan Akta" is open, containing a large number of input fields for detailed deed information, including: Nomor Akta, Tanggal Pembuatan Akta, AJB (dropdown), Nomor Hak, Provinsi (Letak Aset), Kabupaten / Kota (Letak Aset), Kecamatan (Letak Aset), Desa / Kelurahan (Letak Aset), Luas Tanah, Luas Bangunan, Harga Pengalihan, NOP, NJOP, Tanggal SSP, Harga SSP, Tanggal SBB, Harga SBB, Keterangan, Nama PT, Jabatan Pengalih, Jenis Surat Kuasa, Keterangan Surat Kuasa, Jenis Bukti Perjanjian, Jumlah Perjanjian (Rp), Tanggal Perjanjian (Rp), Nomor Perjanjian, NIK Saksi, Nama Saksi, Tempat Lahir Saksi, Tanggal Lahir Saksi, Pekerjaan Saksi, Warga Negara Indonesia, Provinsi (Alamat Saksi), Kabupaten (Alamat Saksi), Alamat Saksi, RT Saksi (Alamat Saksi), RW Saksi (Alamat Saksi), Desa Saksi (Alamat Saksi), Kecamatan Saksi, and Status Saksi. A "Simpan" button is at the bottom of the modal. The background shows a partial view of a yellow circular logo with the letters "PPAT".

Gambar 4. 18 Halaman Tambah Data Keterangan Akta

k. Halaman cetak PDF



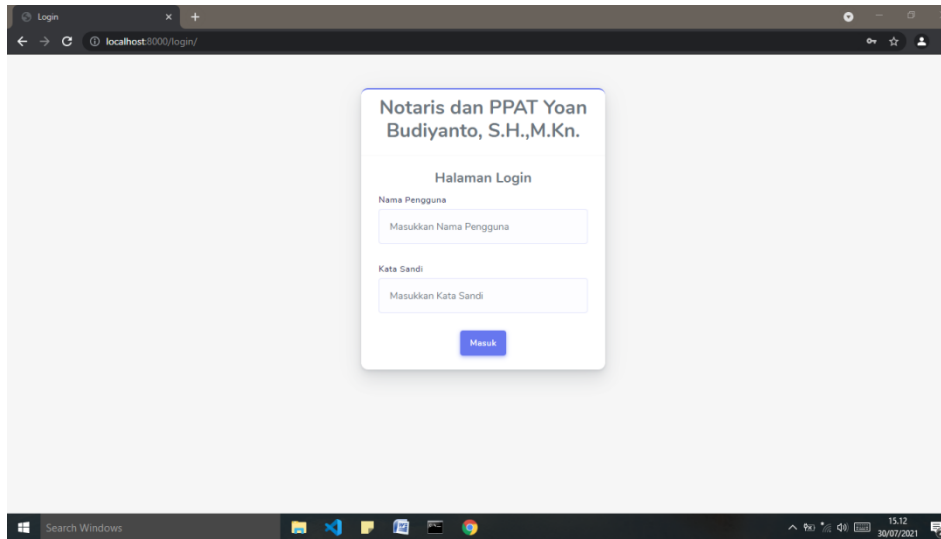
Gambar 4.19 Halaman Cetak Pdf

4.3 Implementasi

Setelah desain sistem selesai maka langkah selanjutnya adalah melakukan implementasi desain ke dalam bentuk kode program. Berikut merupakan hasil dari implementasi desain aplikasi kearsipan dan pembuatan laporan akta.

a. Halaman login

Halaman login digunakan oleh pengguna untuk mengakses aplikasi dengan memasukkan nama pengguna serta kata sandi. Jika pengguna berhasil masuk, maka pengguna akan diarahkan ke halaman utama aplikasi, jika pengguna gagal masuk maka pengguna tidak dapat menuju halaman utama. Di bawah ini merupakan gambar halaman login.



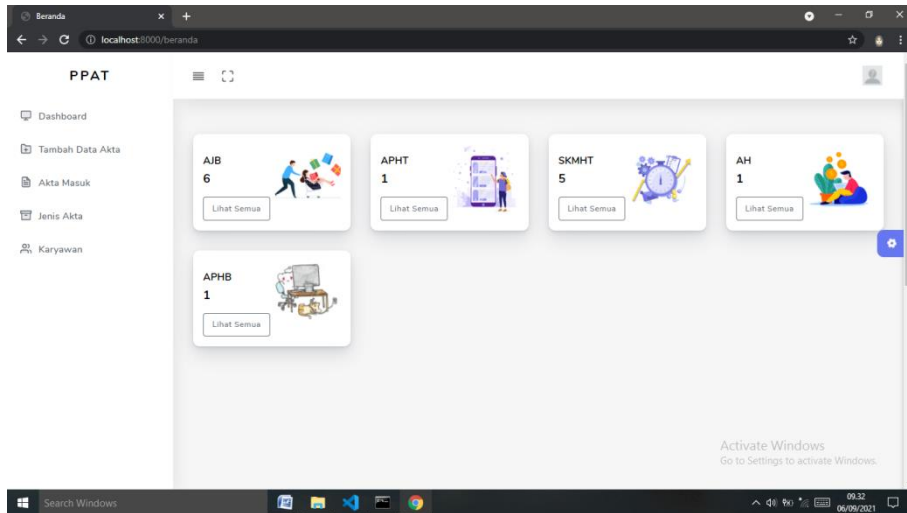
Gambar 4. 20 Halaman Login

Segmen Program 4.1 Views Halaman Login

```
@tolakhalaman_in
def loginPage(request):
    formlogin = AuthenticationForm
    if request.method == 'POST':
        username = request.POST.get('username')
        password = request.POST.get('password')
        cocokan = authenticate(request, username=username,
            password=password)
        if cocokan is not None:
            login(request, cocokan)
            return redirect('beranda')
        context = {
            'menu': 'Login',
            'page': 'Halaman login',
            'tampillogin': formlogin
        }
        return render(request, 'stap/login.html', context)
```

b. Halaman utama

Setelah pengguna berhasil masuk melalui login maka akan menampilkan halaman utama atau *dashboard*. Pada halaman utama akan menampilkan jumlah pembuatan akta yang dilakukan menggunakan aplikasi ini. Berikut merupakan gambar halaman utama atau dashboard.



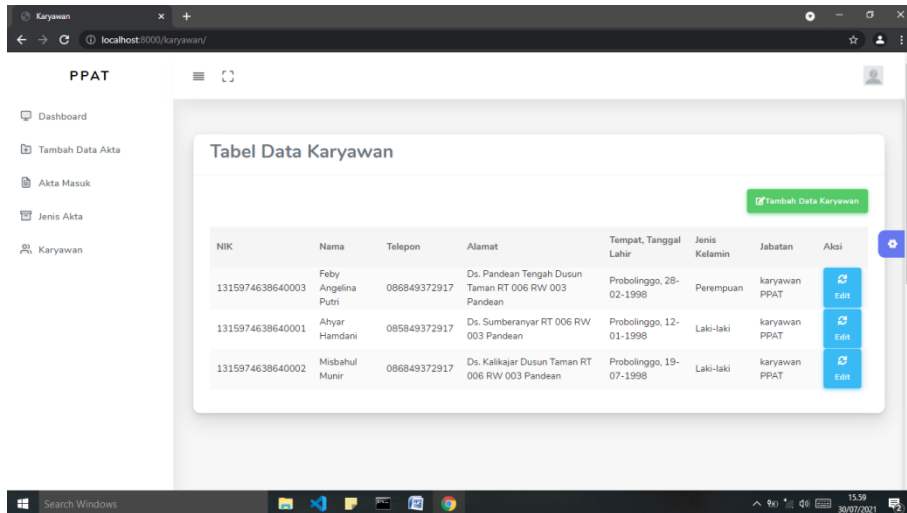
Gambar 4. 21 Halaman Utama

Segmen Program 4. 2 Views Halaman Utama

```
@login_required(login_url='login')
def home(request):
    akta = Akta.objects.order_by('-id')
    ajb = akta.filter(jenis = 'AIB').count()
    apht = akta.filter(jenis = 'APHT').count()
    skmht = akta.filter(jenis = 'SKMHT').count()
    ah = akta.filter(jenis = 'AH').count()
    apbh = akta.filter(jenis = 'APHB').count()
    context = {
        'menu': 'Beranda',
        'page': 'selamat datang di Notaris dan PPAT',
        'data_ajb': ajb,
        'data_apht': apht,
        'data_skmht': skmht,
        'data_ah': ah,
        'data_apbh': apbh,
    }
    return render(request, 'strap/home.html', context)
```

c. Halaman karyawan

Pada halaman karyawan akan ditampilkan data-data karyawan di kantor Notaris dan PPAT Yoan Budiyanto,S.H.,M.Kn. Terdapat firue berupa tambah data dan edit data karyawan. Berikut merupakan gambar tampilan halaman karyawan.



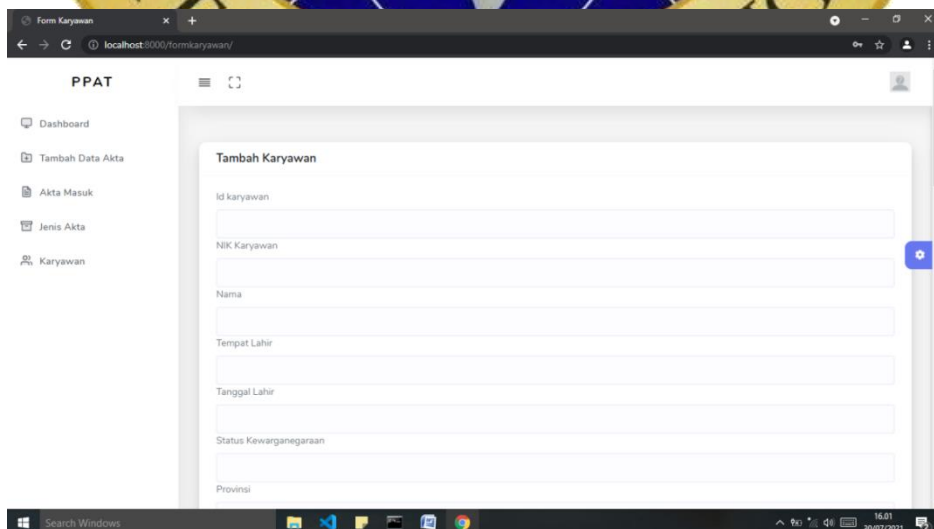
Gambar 4. 22 Halaman Karyawan

Segmen Program 4. 3 Views Halaman Karyawan

```
@login_required(login_url='login')
def karyawan(request):
    list_karyawan = Karyawan.objects.all()
    context = {
        'menu': 'Karyawan',
        'page': 'halaman karyawan',
        'karyawan': list_karyawan
    }
    return render(request, 'strap/karyawan.html', context)
```

d. Halaman Tambah Karyawan

Pada halaman ini digunakan pengguna untuk menambah data karyawan. Berikut merupakan halaman tambah data karyawan.



(a)

(b)

(c)

Gambar 4. 23 Halaman Tambah Karyawan

(a) Halaman Tambah Karyawan (b) Lanjutan (c) Lanjutan

Segmen Program 4. 4 Views Halaman Tambah Karyawan

```

@login_required(login_url='login')
def Tambahkaryawan(request):
    form = KaryawanForm()
    formkaryawan = KaryawanForm(request.POST)
    if request.method == 'POST':
        username = request.POST.get('username')
        password1 = request.POST.get('password')
        password2 = request.POST.get('password2')

        if User.objects.filter(username = username).first():

```

```

messages.success(request, 'Username sudah ada.')
return redirect('formkaryawan')

if password1 != password2:
    messages.success(request, 'Password Tidak sama')
    return redirect('formkaryawan')
# user
user = User.objects.create_user(username=username)
user.set_password(password1)
user.is_active = True
user.save()

# Group
strap_karyawan = Group.objects.get(name='karyawan')
user.groups.add(strap_karyawan)

# Karyawan
formsimpan_karyawan = formkaryawan.save()
formsimpan_karyawan.user = user
formsimpan_karyawan.save()
context = {
    'menu': 'Form Karyawan',
    'page': 'form karyawan',
    'form': form,
}
return render(request, 'strap/tambah_karyawan.html', context)

```

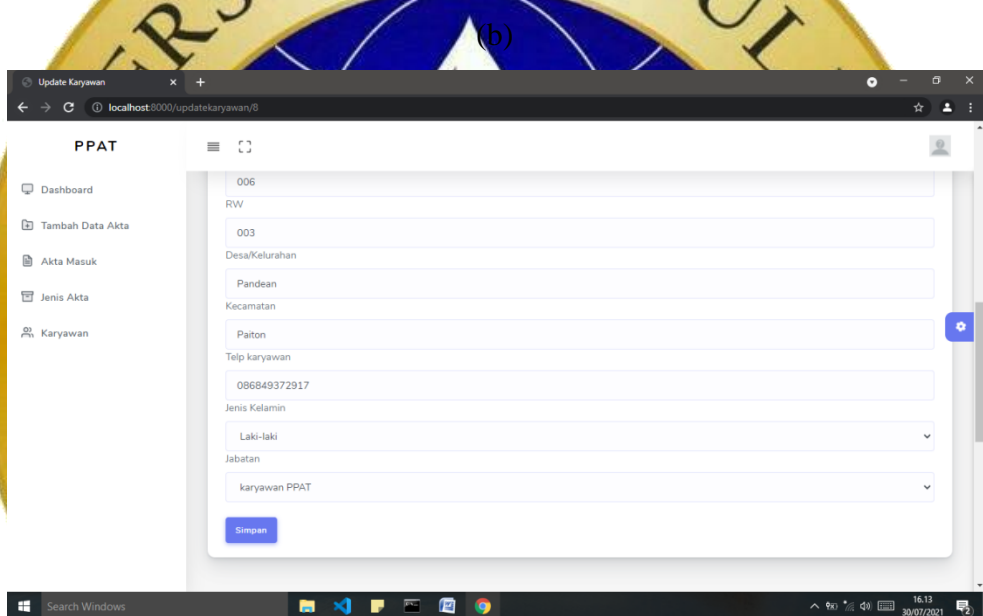
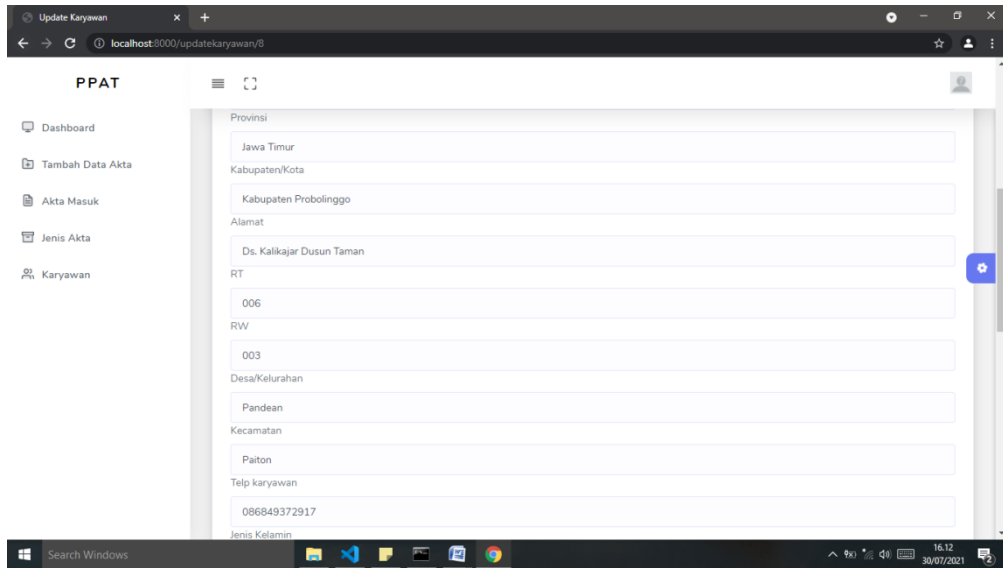
e. Halaman Edit Karyawan

Pada halaman ini pengguna dapat mengubah data karyawan. Berikut merupakan halaman edit karyawan.

The screenshot shows a web browser window with the URL `localhost:8000/updatekaryawan/8`. The page title is "Update Karyawan". On the left, there is a sidebar menu for "PPAT" with items: Dashboard, Tambah Data Akta, Akta Masuk, Jenis Akta, and Karyawan. The main content area is a form titled "Update Karyawan" with the following fields:

- Id karyawan: 05
- NIK Karyawan: 1315974638640002
- Nama: Misbahul Munir
- Tempat Lahir: Probolinggo
- Tanggal Lahir: 19-07-1998
- Status Kewarganegaraan: Warga Negara Indonesia
- Provinsi: (empty)

(a)



Gambar 4. 24 Halaman Edit Karyawan

(a) Halaman Edit Karyawan (b) Lanjutan (c) Lanjutan

Segmen Program 4. 5 Views Halaman Edit Karyawan

```
@login_required(login_url='login')
def updateKaryawan(request, pk):
    up_karyawan = Karyawan.objects.get(id=pk)
    formkaryawan = KaryawanForm(instance=up_karyawan)
    if request.method == 'POST':
        # print('Cetak POST:', request.POST)
        formedit = KaryawanForm(request.POST, instance=up_karyawan)
        if formedit.is_valid():
            formedit.save()
```

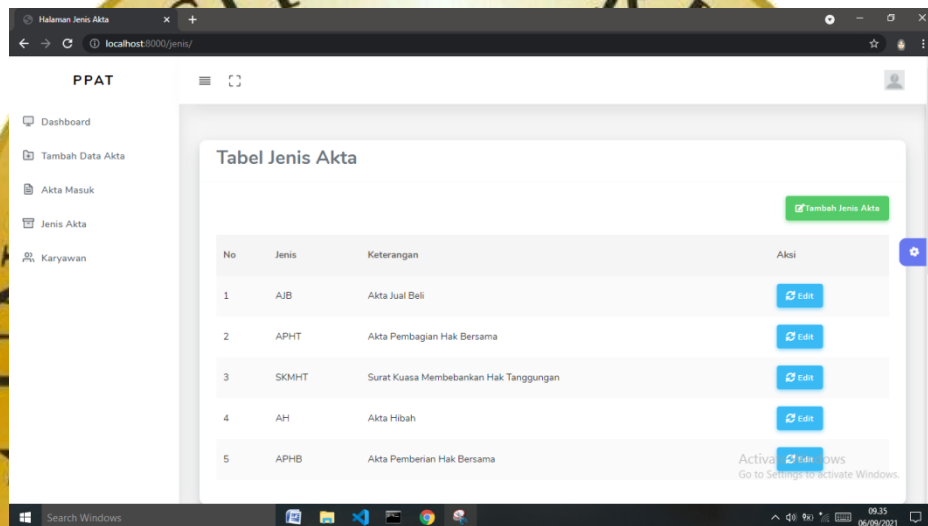
```

return redirect('karyawan')
context = {
    'menu' : 'Update Karyawan',
    'page' : 'Edit karyawan',
    'form' : formkaryawan,
}
return render(request, 'strap/update_karyawan.html', context)

```

f. Halaman Jenis Akta

Halaman ini menampilkan jenis-jenis akta yang dibuat oleh Notaris dan PPAT Yoan Budiyanto S.H.,M.Kn. Di bawah ini merupakan gambar tampilan jenis akta.



Gambar 4. 25 Halaman Jenis Akta

Segmen Program 4. 6 Views Halaman Jenis Akta

```

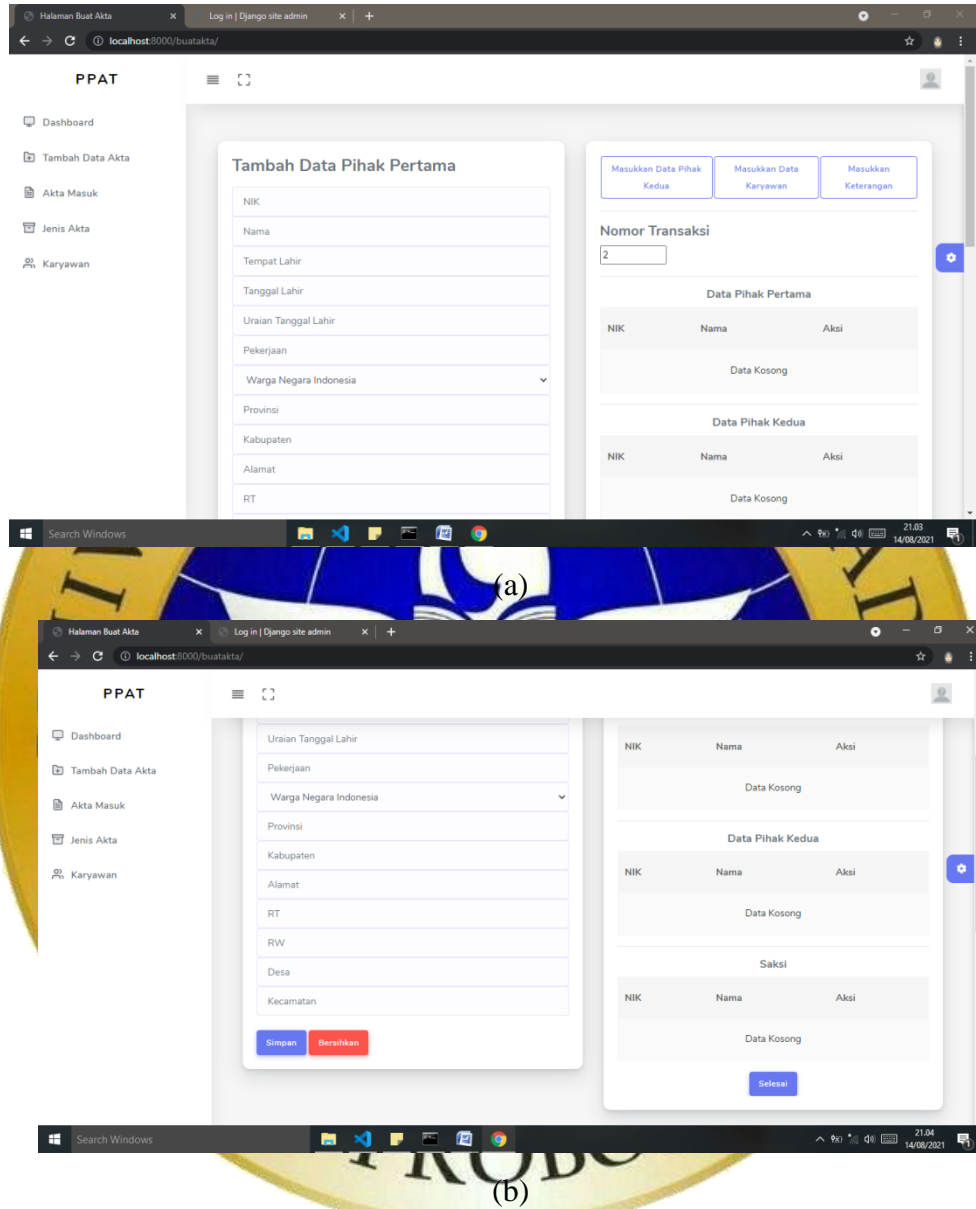
@login_required(login_url='login')
def jenis(request):
    list_jenis = Jenis.objects.all()
    context = {
        'menu' : 'Halaman Jenis Akta',
        'page' : 'jenis',
        'jenis' : list_jenis,
    }
    return render(request, 'strap/jenis.html', context)

```

g. Halaman Buat Akta

Halaman ini digunakan pengguna untuk menambah data akta. Pada halaman ini terdapat *form* untuk menambah data pihak pertama, fitur *modal*

untuk menambah data pihak kedua, karyawan, dan keterangan akta, serta tampilan hasil dari tambah data pihak pertama, pihak kedua, karyawan, dan keterangan akta. Berikut merupakan gambar tampilan halaman buat akta.



Gambar 4. 26 Halaman Buat Akta

(a) Halaman Buat Akta (b) Lanjutan (c) Lanjutan

Segmen Program 4. 7 Views Tambah Pihak Pertama

```
@login_required(login_url='login')
def simpanclientpertama(request):
    if request.method == 'POST':
        user = request.user
        nik_pertama = request.POST.get('nik_pertama')
        nama_pertama = request.POST.get('nama_pertama')
```

```

tempat_lahir_pertama = request.POST.get('tempat_lahir_pertama')
tanggal_lahir_pertama = request.POST.get('tanggal_lahir_pertama')
pekerjaan_pertama = request.POST.get('pekerjaan_pertama')
status_kewarganegaraan_pertama =
request.POST.get('status_kewarganegaraan_pertama')
prov_pertama = request.POST.get('prov_pertama')
kab_pertama = request.POST.get('kab_pertama')
alamat_pertama = request.POST.get('alamat_pertama')
rt_pertama = request.POST.get('rt_pertama')
rw_pertama = request.POST.get('rw_pertama')
desa_pertama = request.POST.get('desa_pertama')
kec_pertama = request.POST.get('kec_pertama')

cek_temp = TempClientSatu.objects.filter(user=user, nik_pertama =
nik_pertama, nama_pertama=nama_pertama,
tempat_lahir_pertama=tempat_lahir_pertama,
tanggal_lahir_pertama=tanggal_lahir_pertama,
pekerjaan_pertama=pekerjaan_pertama,
status_kewarganegaraan_pertama=status_kewarganegaraan_pertama,
prov_pertama=prov_pertama, kab_pertama=kab_pertama,
alamat_pertama=alamat_pertama, rt_pertama=rt_pertama,
rw_pertama=rw_pertama, desa_pertama=desa_pertama,
kec_pertama=kec_pertama).count()
if cek_temp > 0:
    temp = TempClientSatu.objects.get(user = user, nik_pertama =
nik_pertama, nama_pertama=nama_pertama,
tempat_lahir_pertama=tempat_lahir_pertama,
tanggal_lahir_pertama=tanggal_lahir_pertama,
pekerjaan_pertama=pekerjaan_pertama,
status_kewarganegaraan_pertama=status_kewarganegaraan_pertama,
prov_pertama=prov_pertama, kab_pertama=kab_pertama,
alamat_pertama=alamat_pertama, rt_pertama=rt_pertama,
rw_pertama=rw_pertama, desa_pertama=desa_pertama,
kec_pertama=kec_pertama)
    temp.save()

else:
    simpan = TempClientSatu.objects.create(user = user, nik_pertama
= nik_pertama, nama_pertama=nama_pertama,
tempat_lahir_pertama=tempat_lahir_pertama,
tanggal_lahir_pertama=tanggal_lahir_pertama,
pekerjaan_pertama=pekerjaan_pertama,
status_kewarganegaraan_pertama=status_kewarganegaraan_pertama,
prov_pertama=prov_pertama, kab_pertama=kab_pertama,
alamat_pertama=alamat_pertama, rt_pertama=rt_pertama,
rw_pertama=rw_pertama, desa_pertama=desa_pertama,
kec_pertama=kec_pertama)
    simpan.save()

```

```
return JsonResponse({'status': 0 })
```

Segmen Program 4. 8 Views Tampil Tambah Pihak Kedua

```
@login_required(login_url='login')
def TampilClientSatu(request):
    temp = TempClientSatu.objects.order_by('-id').filter(user=request.user)
    context = {
        'temp' : temp,
    }
    return render(request, 'strap/tampil_js/tampilclientsatujs.html', context)
```

Segmen Program 4. 9 Views Tampil Pihak Kedua

```
@login_required(login_url='login')
def TampilClientDua(request):
    tempdua = TempClientDua.objects.order_by('-id').filter(user=request.user)
    context = {
        'tempdua' : tempdua,
    }
    return render(request, 'strap/tampil_js/tampilclientduajs.html', context)
```

Segmen Program 4. 10 Views Tampil Karyawan

```
@login_required(login_url='login')
def TampilKaryawan(request):
    temp = TempKaryawan.objects.order_by('-id')
    context = {
        'temp' : temp,
    }
    return render(request, 'strap/tampil_js/tampilkaryawanjs.html', context)
```

Segmen Program 4. 11 Views Tombol Selesai

```
@login_required(login_url='login')
def simpantransaksi(request):
    if request.method == 'POST':
        no_transaksi = request.POST.get('no_transaksi')
        user = request.user
        trans = TempAkta.objects.order_by('-id').filter(user=request.user)
        for i in trans:
            instance_trans = Akta(
                no_transaksi = no_transaksi,
                user = user,
                nomor_akta= i.nomor_akta,
                tgl_akta=i.tgl_akta,
                jenis=i.jenis,
                nomor_hak=i.nomor_hak,
```

```

prov_tanah=i.prov_tanah,
kab_tanah=i.kab_tanah,
kec_tanah=i.kec_tanah,
desa_tanah=i.desa_tanah,
luas_tanah=i.luas_tanah,
luas_bgn=i.luas_bgn,
harga_pengalihan=i.harga_pengalihan,
nop=i.nop,
njop=i.njop,
nib=i.nib,
tgl_ssp=i.tgl_ssp,
harga_ssp=i.harga_ssp,
tgl_sbb=i.tgl_sbb,
harga_sbb=i.harga_sbb,
keterangan=i.keterangan,
pt=i.pt,
jabatan_pt=i.jabatan_pt,
jenis_kuasa=i.jenis_kuasa,
keterangan_kuasa=i.keterangan_kuasa,
jenis_bukti_perjanjian=i.jenis_bukti_perjanjian,
no_bukti_ppat=i.no_bukti_ppat,
jumlah_perjanjian=i.jumlah_perjanjian,
tanggal_perjanjian=i.tanggal_perjanjian,
no_perjanjian=i.no_perjanjian,
nik_saksi=i.nik_saksi,
nama_saksi=i.nama_saksi,
tempat_lahir_saksi=i.tempat_lahir_saksi,
tanggal_lahir_saksi=i.tanggal_lahir_saksi,
pekerjaan_saksi=i.pekerjaan_saksi,
status_kewarganegaraan_saksi=i.status_kewarganegaraan_saksi,
prov_saksi=i.prov_saksi,
kab_saksi=i.kab_saksi,
alamat_saksi=i.alamat_saksi,
rt_saksi=i.rt_saksi,
rw_saksi=i.rw_saksi,
desa_saksi=i.desa_saksi,
kec_saksi=i.kec_saksi,
status_saksi=i.status_saksi

```

```

)
instance_trans.save()
trans.delete()

```

```

temp = TempClientSatu.objects.order_by('-
id').filter(user=request.user)
for r in temp:
instance_detail= DetailClientSatu(
no_transaksi = no_transaksi,
user = request.user,

```



```

        nik_pertama = r.nik_pertama,
        nama_pertama= r.nama_pertama,
        tempat_lahir_pertama= r.tempat_lahir_pertama,
        tanggal_lahir_pertama= r.tanggal_lahir_pertama,
        pekerjaan_pertama= r.pekerjaan_pertama,
        status_kewarganegaraan_pertama=
r.status_kewarganegaraan_pertama,
        prov_pertama= r.prov_pertama,
        kab_pertama= r.kab_pertama,
        alamat_pertama= r.alamat_pertama,
        rt_pertama= r.rt_pertama,
        rw_pertama=r.rw_pertama,
        desa_pertama=r.desa_pertama,
        kec_pertama=r.kec_pertama
    )
    instance_detail.save()
    temp.delete()

    tempdua = TempClientDua.objects.order_by('-
id').filter(user=request.user)
    for a in tempdua:
        instance_detaildua= DetailClientDua(
            no_transaksi = no_transaksi,
            user = request.user,
            nik_kedua = a.nik_kedua,
            nama_kedua= a.nama_kedua,
            tempat_lahir_kedua= a.tempat_lahir_kedua,
            tanggal_lahir_kedua= a.tanggal_lahir_kedua,
            pekerjaan_kedua= a.pekerjaan_kedua,
            status_kewarganegaraan_kedua=
a.status_kewarganegaraan_kedua,
            prov_kedua= a.prov_kedua,
            kab_kedua= a.kab_kedua,
            alamat_kedua= a.alamat_kedua,
            rt_kedua= a.rt_kedua,
            rw_kedua=a.rw_kedua,
            desa_kedua=a.desa_kedua,
            kec_kedua=a.kec_kedua
        )
        instance_detaildua.save()
    tempdua.delete()

    tempk = TempKaryawan.objects.order_by('id')
    for b in tempk:
        instance_karyawan = DetailKaryawan(
            no_transaksi = no_transaksi,
            user = request.user,

```

```

        nik_karyawan = b.nik_karyawan
    )
    instance_karyawan.save()
    tempk.delete()

    return JsonResponse({'status': 0 })
else:
    return JsonResponse({'status': 0 })

```

h. Halaman Tambah Data Pihak Kedua

Pada halaman ini pengguna dapat menambah data pihak kedua. Form tambah pihak kedua merupakan modal dari tampilan halaman buat akta. Berikut merupakan tampilan tambah data pihak kedua

(a)

(b)

Gambar 4. 27 Halaman Tambah Pihak Kedua

(a) Halaman Tambah Pihak Kedua (b) Lanjutan

Segmen Program 4. 12 Views Tambah Data Pihak Kedua

```
@login_required(login_url='login')
def simpanclientkedua(request):
    if request.method == 'POST':
        user = request.user
        nik_kedua = request.POST.get('nik_kedua')
        nama_kedua = request.POST.get('nama_kedua')
        tempat_lahir_kedua = request.POST.get('tempat_lahir_kedua')
        tanggal_lahir_kedua = request.POST.get('tanggal_lahir_kedua')
        pekerjaan_kedua = request.POST.get('pekerjaan_kedua')
        status_kewarganegaraan_kedua =
request.POST.get('status_kewarganegaraan_kedua')
        prov_kedua = request.POST.get('prov_kedua')
        kab_kedua = request.POST.get('kab_kedua')
        alamat_kedua = request.POST.get('alamat_kedua')
        rt_kedua = request.POST.get('rt_kedua')
        rw_kedua = request.POST.get('rw_kedua')
        desa_kedua = request.POST.get('desa_kedua')
        kec_kedua = request.POST.get('kec_kedua')

        cek_temp = TempClientDua.objects.filter(user=user, nik_kedua =
nik_kedua, nama_kedua=nama_kedua,
tempat_lahir_kedua=tempat_lahir_kedua,
tanggal_lahir_kedua=tanggal_lahir_kedua,
pekerjaan_kedua=pekerjaan_kedua,
status_kewarganegaraan_kedua=status_kewarganegaraan_kedua,
prov_kedua=prov_kedua, kab_kedua=kab_kedua,
alamat_kedua=alamat_kedua, rt_kedua=rt_kedua, rw_kedua=rw_kedua,
desa_kedua=desa_kedua, kec_kedua=kec_kedua).count()
        if cek_temp > 0:
            temp = TempClientDua.objects.get(user = user, nik_kedua =
nik_kedua, nama_kedua=nama_kedua,
tempat_lahir_kedua=tempat_lahir_kedua,
tanggal_lahir_kedua=tanggal_lahir_kedua,
pekerjaan_kedua=pekerjaan_kedua,
status_kewarganegaraan_kedua=status_kewarganegaraan_kedua,
prov_kedua=prov_kedua, kab_kedua=kab_kedua,
alamat_kedua=alamat_kedua, rt_kedua=rt_kedua, rw_kedua=rw_kedua,
desa_kedua=desa_kedua, kec_kedua=kec_kedua)
            temp.save()

        else:
```

```

simpan = TempClientDua.objects.create(user = user, nik_kedua =
nik_kedua, nama_kedua=nama_kedua,
tempat_lahir_kedua=tempat_lahir_kedua,
tanggal_lahir_kedua=tanggal_lahir_kedua,
pekerjaan_kedua=pekerjaan_kedua,
status_kewarganegaraan_kedua=status_kewarganegaraan_kedua,
prov_kedua=prov_kedua, kab_kedua=kab_kedua,
alamat_kedua=alamat_kedua, rt_kedua=rt_kedua, rw_kedua=rw_kedua,
desa_kedua=desa_kedua, kec_kedua=kec_kedua)
simpan.save()

```

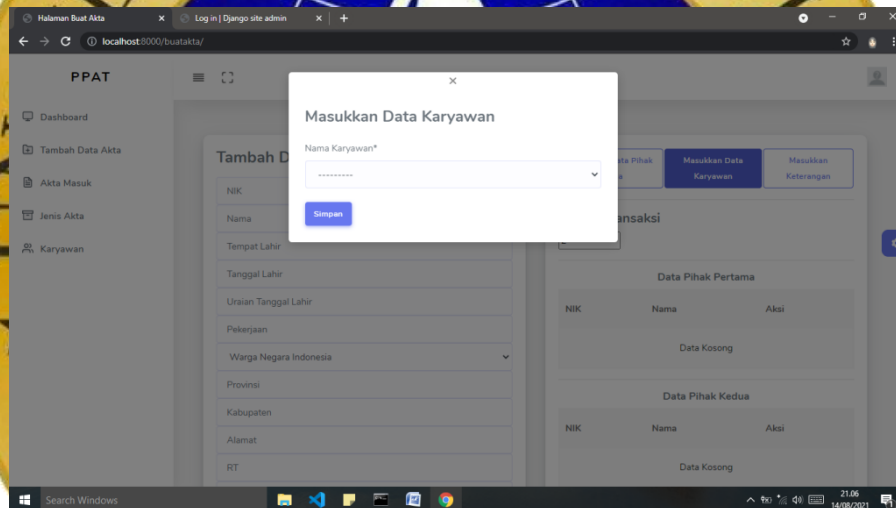
```

return JsonResponse({'status': 0 })

```

i. Halaman Tambah Karyawan

Halaman ini digunakan pengguna untuk memilih data karyawan sebagai pembuat akta. Berikut merupakan tampilan halaman tambah karyawan.



Gambar 4. 28 Halaman Tambah Data Karyawan

Segmen Program 4. 13 Views Tambah Data Karyawan Sebagai Saksi

```

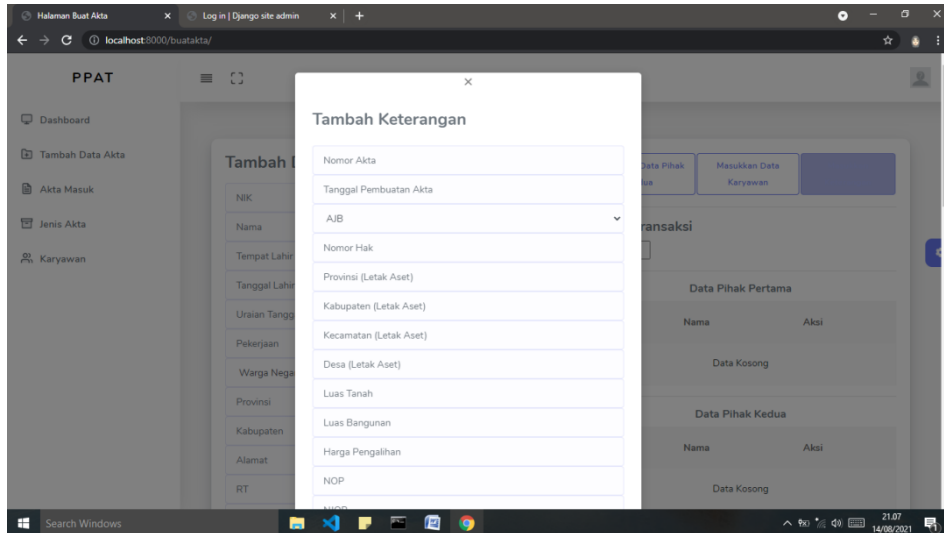
@login_required(login_url='login')
def SimpanFormK(request):
    if request.is_ajax and request.method == "POST":
        formk = SaksiForm(request.POST)
        if formk.is_valid():
            instance = formk.save()
            ser_instance = serializers.serialize('json',[instance,])
            return JsonResponse({"instance":ser_instance}, status=200)
        else:
            return JsonResponse({"error":formk.errors}, status=400)
    return JsonResponse({"error":""}, status=400)

```

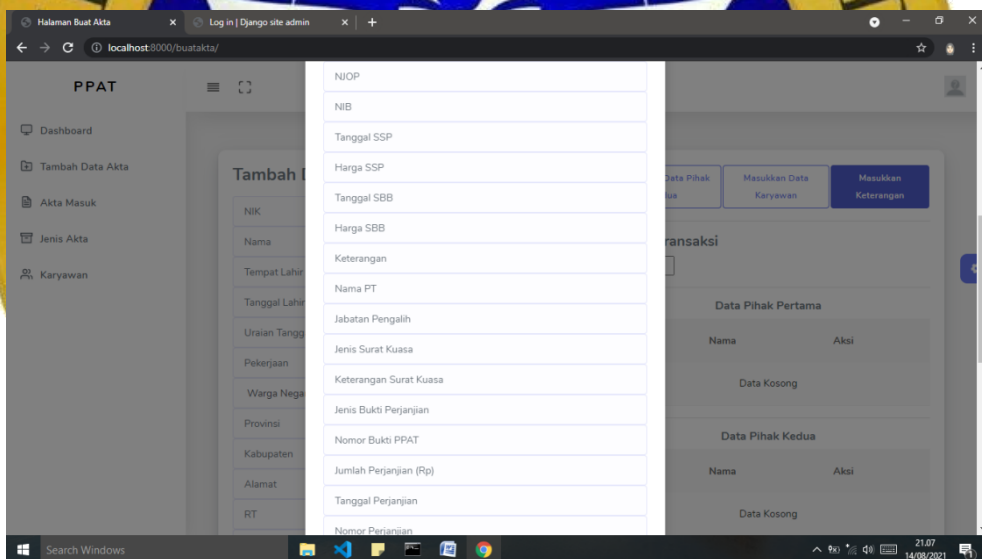
j. Halaman Tambah Keterangan Akta

Halaman ini digunakan pengguna untuk menambah data keterangan akta.

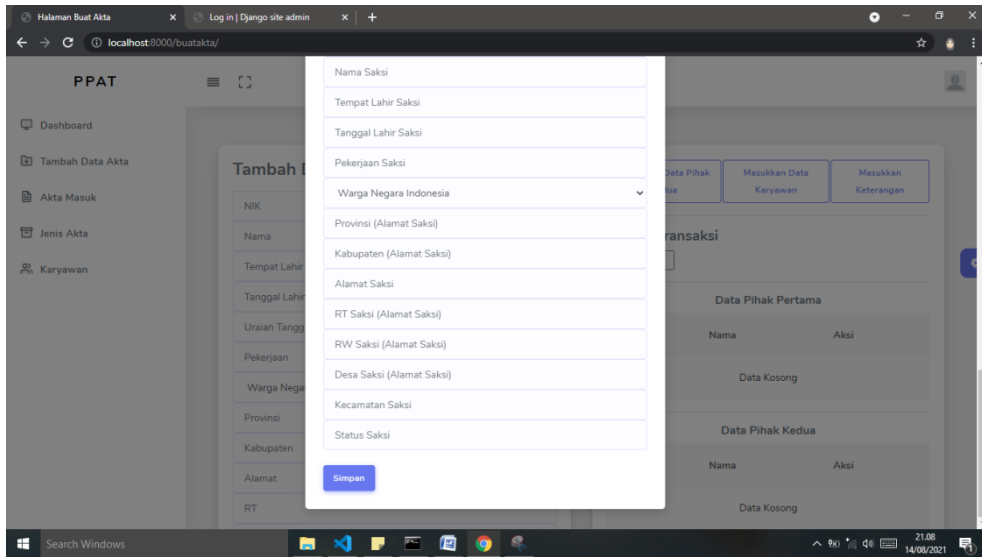
Berikut merupakan gambar tampilan halaman tambah keterangan akta.



(a)



(b)



(c)
Gambar 4. 29 Halaman Tambah Keterangan Akta

(a) Halaman Tambah Keterangan Akta (b) Lanjutan (c) Lanjutan
Segmen Program 4. 14 Views Tambah Keterangan Akta

```
@login_required(login_url='login')
def simpanketerangan(request):
    if request.method == 'POST':
        user = request.user
        nomor_akta = request.POST.get('nomor_akta')
        tgl_akta = request.POST.get('tgl_akta')
        jenis = request.POST.get('jenis')
        nomor_hak = request.POST.get('nomor_hak')
        prov_tanah = request.POST.get('prov_tanah')
        kab_tanah = request.POST.get('kab_tanah')
        kec_tanah = request.POST.get('kec_tanah')
        desa_tanah = request.POST.get('desa_tanah')
        luas_tanah = request.POST.get('luas_tanah')
        luas_bgn = request.POST.get('luas_bgn')
        harga_pengalihan = request.POST.get('harga_pengalihan')
        nop = request.POST.get('nop')
        njop = request.POST.get('njop')
        nib = request.POST.get('nib')
        tgl_ssp = request.POST.get('tgl_ssp')
        harga_ssp = request.POST.get('harga_ssp')
        tgl_sbb = request.POST.get('tgl_sbb')
        harga_sbb = request.POST.get('harga_sbb')
        keterangan = request.POST.get('keterangan')
        pt = request.POST.get('pt')
        jabatan_pt = request.POST.get('jabatan_pt')
        jenis_kuasa = request.POST.get('jenis_kuasa')
```

```

keterangan_kuasa = request.POST.get('keterangan_kuasa')
jenis_bukti_perjanjian = request.POST.get('jenis_bukti_perjanjian')
no_bukti_ppat = request.POST.get('no_bukti_ppat')
jumlah_perjanjian = request.POST.get('jumlah_perjanjian')
tanggal_perjanjian = request.POST.get('tanggal_perjanjian')
no_perjanjian = request.POST.get('no_perjanjian')
nik_saksi = request.POST.get('nik_saksi')
nama_saksi = request.POST.get('nama_saksi')
tempat_lahir_saksi = request.POST.get('tempat_lahir_saksi')
tanggal_lahir_saksi = request.POST.get('tanggal_lahir_saksi')
uraian_tl_saksi = request.POST.get('uraian_tl_saksi')
pekerjaan_saksi = request.POST.get('pekerjaan_saksi')
status_kewarganegaraan_saksi =
request.POST.get('status_kewarganegaraan_saksi')
prov_saksi = request.POST.get('prov_saksi')
kab_saksi = request.POST.get('kab_saksi')
alamat_saksi = request.POST.get('alamat_saksi')
rt_saksi = request.POST.get('rt_saksi')
rw_saksi = request.POST.get('rw_saksi')
desa_saksi = request.POST.get('desa_saksi')
kec_saksi = request.POST.get('kec_saksi')
status_saksi = request.POST.get('status_saksi')

simpan = TempAkta.objects.create(
    user=user,
    nomor_akta=nomor_akta,
    tgl_akta=tgl_akta,
    jenis=jenis,
    nomor_hak=nomor_hak,
    prov_tanah=prov_tanah,
    kab_tanah=kab_tanah,
    kec_tanah=kec_tanah,
    desa_tanah=desa_tanah,
    luas_tanah=luas_tanah,
    luas_bgn=luas_bgn,
    harga_pengalihan=harga_pengalihan,
    nop=nop,
    njop=njop,
    nib=nib,
    tgl_ssp=tgl_ssp,
    harga_ssp=harga_ssp,
    tgl_sbb=tgl_sbb,
    harga_sbb=harga_sbb,
    keterangan=keterangan,
    pt=pt,
    jabatan_pt=jabatan_pt,
    jenis_kuasa=jenis_kuasa,
    keterangan_kuasa=keterangan_kuasa,

```

```

jenis_bukti_perjanjian=jenis_bukti_perjanjian,
no_bukti_ppat = no_bukti_ppat,
jumlah_perjanjian=jumlah_perjanjian,
tanggal_perjanjian=tanggal_perjanjian,
no_perjanjian=no_perjanjian,
nik_saksi=nik_saksi,
nama_saksi=nama_saksi,
tempat_lahir_saksi=tempat_lahir_saksi,
tanggal_lahir_saksi=tanggal_lahir_saksi,
uraian_tl_saksi=uraian_tl_saksi,
pekerjaan_saksi=pekerjaan_saksi,
status_kewarganegaraan_saksi=status_kewarganegaraan_saksi,
prov_saksi=prov_saksi,
kab_saksi=kab_saksi,
alamat_saksi=alamat_saksi,
rt_saksi=rt_saksi,
rw_saksi=rw_saksi,
desa_saksi=desa_saksi,
kec_saksi=kec_saksi,
status_saksi=status_saksi)
simpan.save()

return JsonResponse({'status': 0 })

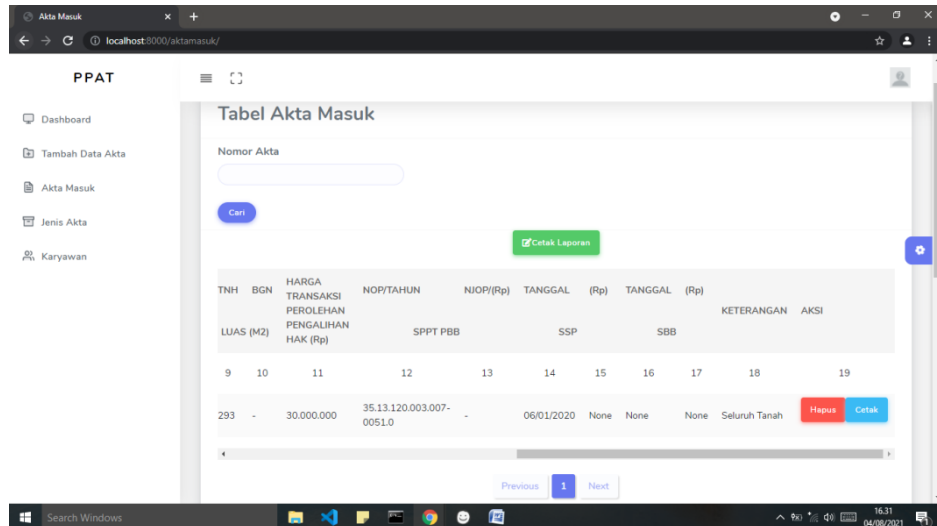
```

k. Halaman Akta Masuk

Halaman ini digunakan oleh pengguna untuk melihat catatan akta masuk, catatan akta masuk diperoleh dari hasil proses pembuatan akta yang dilakukan di halaman buat akta. Berikut merupakan gambar halaman akta masuk.

NO URUT	NOMOR AKTA	TANGGAL	BENTUK PEMBUATAN HUKUM	PIHAK YANG MENGALIHKAN NAMA, ALAMAT DAN NPWP	PIHAK YANG MENERIMA Putri, Dusun Pandan, Hasan Bulq, Dusun Pandan	JENIS DAN NOMOR HAK	LETAK TANAH DAN BANGUNAN	TNH	BGN	HARGA TRANSAKSI PEROLEHAN PENGALIHAN HAK (Rp)
1	002/2020	06/01/2020	AH	Fidayatul hasanah, Dusun Taman	Putri, Dusun Pandan	M.1479/ Sumberanyar		293	-	-
2	001/2020	02/01/2020	SKMHT	Fidayatul, Dusun Taman	Bulq, Dusun Pandan	M.293/ Sokaan		293	-	-

(a)



(b)
Gambar 4. 30 Halaman Akta Masuk

a) Halaman Akta Masuk (b) Lanjutan

Segmen Program 4. 15 Views Halaman Akta Masuk

```
@login_required(login_url='login')
def aktamasuk(request):
    list_akta = Akta.objects.order_by('-id')
    list_clientsatu = DetailClientSatu.objects.filter(no_transaksi='1')
    filter_akta = AktaFilter(request.GET, queryset=list_akta)
    list_akta = filter_akta.qs

    halaman_tampil = Paginator(list_akta, 5)
    halaman_url = request.GET.get('halaman',1)
    halaman_akta = halaman_tampil.get_page(halaman_url)

    if halaman_akta.has_previous():
        url_previous = f'?halaman={halaman_akta.previous_page_number()}'
    else:
        url_previous = ""

    if halaman_akta.has_next():
        url_next = f'?halaman={halaman_akta.next_page_number()}'
    else:
        url_next = ""

    context = {
        'menu': 'Akta Masuk',
        'page': 'halaman akta masuk',
        'aktamasuk': list_akta,
        'clientsatu': list_clientsatu,
        'filter_akta': filter_akta,
        'page_akta': halaman_akta,
```

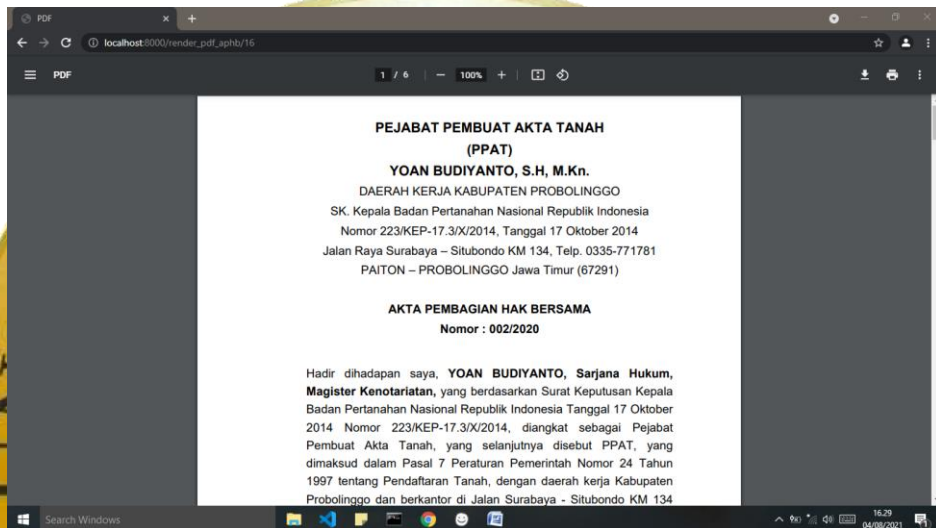
```

    'previous' : url_previous,
    'next' : url_next
}
return render(request, 'strap/aktamasuk.html', context)

```

I. Halaman Hasil Cetak PDF

Saat pengguna mencetak surat maka akan muncul tampilan dokumen berupa PDF. Surat yang dihasilkan berdasarkan jenis pembuatan hukum yang tertera pada halaman akta masuk.



Gambar 4. 31 Halaman Hasil Cetak PDF

Segmen Program 4. 16 Views Cetak PDF

```

@login_required(login_url='login')
def render_pdf_akta(request, pk):
    buakta = Akta.objects.get(id=pk)
    template_path = 'strap/pdf_ajb.html'
    context = {'buakta': buakta}
    response = HttpResponse(content_type='application/pdf')
    response['Content-Disposition'] = filename="report.pdf"
    template = get_template(template_path)
    html = template.render(context)
    pisa_status = pisa.CreatePDF(
        html, dest=response)
    if pisa_status.err:
        return HttpResponse('We had some errors <pre>' + html + '</pre>')
    return response

```

4.4 Pengujian

1. Pengujian Internal

Pengujian internal dilakukan dengan cara mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari sistem. Tujuan dari pengujian ini ialah untuk menemukan kesalahan dari fungsi program dan akurasi kinerja sistem sebelum dilakukannya pengujian eksternal kepada *user*. Pengujian dilakukan kepada dosen pengajar yang ahli dalam bidang pemrograman web di Fakultas Teknik Universitas Nurul Jadid yaitu Bapak Sukron, M.Kom.

Tabel 4. 7 Hasil Pengujian Internal

No	Pengujian	Target yang dicapai	Hasil yang diharapkan	Keterangan	
				Ya	Tidak
1	Login	Username dan password benar	Menuju dashboard	✓	
		Username dan password salah	Tidak dapat menuju dashboard	✓	
2	Halaman dashboard	-	Menampilkan perhitungan data akta yang sudah dibuat	✓	
3	Halaman Buat Akta	Tambah data akta	Menampilkan form tambah client pertama	✓	
		Tombol pihak kedua	Menampilkan form tambah client kedua	✓	
		Tombol karyawan	Menampilkan form pilih karyawan sebagai saksi	✓	
		Tombol keterangan	Menampilkan form tambah keterangan akta	✓	

		Tombol Selesai	Menyimpan data client pertama, kedua, karyawan, dan keterangan akta ke dalam database	✓	
		Tabel Tampil data	Menampilkan hasil input data client pertama, kedua, dan karyawan sebagai saksi	✓	
4	Akta Masuk	Tabel akta masuk	Menampilkan data akta dari database	✓	
		Pencarian data	Dapat melakukan pencarian data	✓	
		Tombol cetak	Dapat melakukan cetak pdf	✓	
5	Jenis Akta	Tampil jenis akta	Menampilkan data jenis akta dari database	✓	
6	Karyawan	Tampil Karyawan	Menampilkan data karyawan	✓	
		Tambah karyawan	Dapat menambah data karyawan dan data tersimpan ke dalam database	✓	
		Edit karyawan	Dapat mengubah data karyawan dan data tersimpan ke dalam database	✓	

2. Pengujian eksternal

Angket diberikan kepada 5 Responden, dimana responden adalah karyawan kantor Notaris dan PPAT Yoan Budiyo S.H., M.Kn. Berikut rincian perhitungan point dari pertanyaan 1 sampai dengan pertanyaan 5:

Tabel 4. 8 Hasil Pengujian Eksternal

No	Pernyataan	Pendapat				Keterangan
		SS (4)	S (3)	TS (2)	STS (1)	
1	Sistem ini mempermudah pengarsipan akta.	2	3	0	0	
2	Sistem ini mempercepat pencarian data yang diperlukan.	0	5	0	0	
3	Sistem ini meningkatkan efisiensi waktu dalam pembuatan laporan.	0	4	1	0	
4	Laporan yang dihasilkan oleh aplikasi ini telah sesuai dengan format laporan PPAT.	0	2	3	0	
5	Sistem ini memiliki fitur sesuai dengan kebutuhan pengguna.	0	5	0	0	

Berdasarkan pengumpulan hasil responden diatas, maka dapat ditentukan skor sebagai berikut :

a. Bobot setiap skala

SS (Sangat Setuju) = 4

S (Setuju) = 3

TS (Tidak Setuju) = 2

STS (Sangat Tidak Setuju) = 1

b. Nilai Interval

$I = 100 / \text{jumlah angket}$

$$I = 100 / 4$$

$$I = 25$$

Jadi persentase penilaian adalah sebagai berikut

Tabel 4. 9 Tabel Persentase Penilaian

Jawaban	Keterangan
0% - 24.99%	Sangat Tidak Setuju
25% - 49.99%	Tidak Setuju
50% - 74.99%	Setuju
75% - 100%	Sangat Setuju

c. Perhitungan hasil jawaban responden

$Y = \text{skor tertinggi likert} \times \text{jumlah responden}$

$X = \text{skor terendah likert} \times \text{jumlah responden}$

$$Y = 4 \times 5 = 20$$

$$X = 1 \times 5 = 5$$

1. Pertanyaan ke 1

$$\text{Responden yang menjawab SS} = 2 \times 4 = 8$$

$$\text{Responden yang menjawab S} = 3 \times 3 = 9$$

$$\text{Responden yang menjawab TS} = 0 \times 2 = 0$$

$$\text{Responden yang menjawab STS} = 0 \times 1 = 0$$

$$\text{Total skor} = 8 + 9 + 0 + 0 = 17$$

2. Pertanyaan ke 2

$$\text{Responden yang menjawab SS} = 0 \times 4 = 0$$

$$\text{Responden yang menjawab S} = 5 \times 3 = 15$$

$$\text{Responden yang menjawab TS} = 0 \times 2 = 0$$

$$\text{Responden yang menjawab STS} = 0 \times 1 = 0$$

$$\text{Total skor} = 0 + 15 + 0 + 0 = 15$$

3. Pertanyaan ke 3

$$\text{Responden yang menjawab SS} = 0 \times 4 = 0$$

$$\text{Responden yang menjawab S} = 4 \times 3 = 12$$

$$\text{Responden yang menjawab TS} = 1 \times 2 = 2$$

$$\text{Responden yang menjawab STS} = 0 \times 1 = 0$$

$$\text{Total skor} = 0 + 12 + 2 + 0 = 14$$

4. Pertanyaan ke 4

$$\text{Responden yang menjawab SS} = 0 \times 4 = 0$$

$$\text{Responden yang menjawab S} = 2 \times 3 = 6$$

$$\text{Responden yang menjawab TS} = 3 \times 2 = 6$$

$$\text{Responden yang menjawab STS} = 0 \times 1 = 0$$

$$\text{Total skor} = 0 + 6 + 6 + 0 = 12$$

5. Pertanyaan ke 5

$$\text{Responden yang menjawab SS} = 0 \times 4 = 0$$

$$\text{Responden yang menjawab S} = 5 \times 3 = 15$$

$$\text{Responden yang menjawab TS} = 0 \times 2 = 0$$

$$\text{Responden yang menjawab STS} = 0 \times 1 = 0$$

$$\text{Total skor} = 0 + 15 + 0 + 0 = 15$$

d. Persentase jawaban responden

1. Pertanyaan ke 1 :

$$\begin{aligned} \text{Index \%} &= \text{total skor} / Y \times 100 \\ &= 17 / 20 \times 100 \\ &= 85\% \text{ kategori sangat setuju} \end{aligned}$$

2. Pertanyaan ke 2 :

$$\begin{aligned} \text{Index \%} &= \text{total skor} / Y \times 100 \\ &= 15 / 20 \times 100 \\ &= 75\% \text{ kategori sangat setuju} \end{aligned}$$

3. Pertanyaan ke 3 :

$$\begin{aligned} \text{Index \%} &= \text{total skor} / Y \times 100 \\ &= 14 / 20 \times 100 \\ &= 70\% \text{ kategori setuju} \end{aligned}$$

4. Pertanyaan ke 4 :

$$\begin{aligned} \text{Index \%} &= \text{total skor} / Y \times 100 \\ &= 12 / 20 \times 100 \\ &= 60\% \text{ kategori setuju} \end{aligned}$$

5. Pertanyaan ke 5 :

$$\text{Index \%} = \text{total skor} / Y \times 100$$

$$= 15 / 20 \times 100$$
$$= 75\% \text{ kategori sangat setuju}$$

e. Persentase total jawaban responden

$$\text{Rata-rata} = \text{Total Index \%} / \text{jumlah pertanyaan}$$

$$\text{Rata-rata} = 365 / 5$$

$$\text{Rata-rata} = 73\%$$

Kesimpulan dari perhitungan ke lima pertanyaan di atas dengan 5 responden menunjukkan bahwa aplikasi yang telah dibuat mendapatkan penilaian 73%. Dari hasil tersebut menunjukkan bahwa dengan adanya aplikasi pengarsipan akta dan pembuatan laporan di kantor Notaris dan PPAT Yoan Budiyanto, S.H., M.Kn. berbasis web dengan menggunakan Django ini responden menyatakan setuju.

4.5 Pemeliharaan

Setelah melakukan pengujian terhadap sistem yang dibuat, maka tahap selanjutnya adalah melakukan perawatan (*Maintenance*) terhadap sistem tersebut. Ketika dijalankan mungkin saja masih ada *errors* yang tidak ditemukan dan kekurangan yang tidak dijumpai sebelumnya. Maka selain itu juga berguna dalam penambahan fitur baru nantinya sebagai bentuk pengembangan sistem.