

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Pengumpulan Data

Pengumpulan data pada penelitian ini diperoleh dengan tiga tahap yaitu observasi, wawancara dan studi literatur. Observasi dilakukan di Kantor Madrasah Aliyah Nurul Qadim dengan bertemu bagian lembaga. Dalam observasi dilakukan pencatatan dan pengamatan mengenai sistem pengolahan dan penyimpanan data buku induk santri. Dari observasi ini dihasilkan suatu masalah dari pihak lembaga dan kantor pusat yaitu saling membutuhkan pembaharuan sistem yang lebih efisien.

Sedangkan, wawancara dilakukan secara langsung dengan pihak-pihak terkait dalam sistem buku induk santri, yaitu Bapak Yasin selaku bagian IT pesantren dan Bapak Gozi selaku bagian Kepala Pesantren. Beberapa masalah yang ada di sistem berjalan dapat disimpulkan dari hasil wawancara seperti, pertama selama ini seringkali terjadi tidak sinkron data antara data kantor pusat dan data lembaga. Kedua, belum adanya *database* yang bisa digunakan secara bersamaan. Dan ketiga belum adanya laporan cetak data santri. Studi literatur juga dilakukan untuk melengkapi data yang butuhkan seperti menghasilkan metode yang digunakan dan mengetahui secara teoritis permasalahan yang sedang dihadapi.

4.2 Hasil Analisis dan Desain

4.2.1 Analisis Sistem Lama

Dalam pengolahan data buku induk santri, data santri ditulis dan disimpan menggunakan komputer dengan *software Microsoft Excel* yang menjadi alat bantu. Penulisan data tersebut dilakukan oleh petugas lembaga. Data yang sudah ditulis kemudian disimpan dalam bentuk *softcopy* didalam komputer. Data buku induk yang sudah tersimpan pada lembaga harus memiliki data salinan ke kantor pusat. Pada kantor pusat terdapat buku induk santri pusat, dimana buku induk dari seluruh lembaga tersimpan. Buku induk santri yang dibuat dengan sistem ini sering kali terjadi tidak sinkron data antara data yang terdapat di kantor pusat

dengan data yang ada di lembaga. Selain itu juga, data buku induk santri yang disimpan di Microsoft Excel tidak memiliki laporan data santri yang siap dicetak.

4.2.2 Analisis Sistem Baru

Aplikasi ini mampu melakukan pengolahan data santri secara terpusat dan disimpan didalam satu *database* yang dapat digunakan secara bersamaan. Hal ini untuk meminimalisir terjadinya tidak sinkron data antara data yang terdapat di kantor pusat dan data yang ada di lembaga. Dalam pengolahan data buku induk santri, aplikasi ini memiliki hak akses yang berbeda, pertama adalah hak akses admin pusat, hak akses ini hanya dilakukan oleh admin kantor pusat dimana seluruh pengolahan data seperti proses penginputan dan perubahan data dilakukan. Kedua adalah hak akses admin lembaga, karena dalam ruang lingkup pesantren sehingga terdapat tiga kategori yaitu pertama, pendidikan lembaga atau yang disebut lembaga terdiri dari PAUD Nurul Qadim, TK Nurul Hasan, MI Nurul Qadim, MTS Nurul Qadim dan MA Nurul Qadim. Kedua, pendidikan diniyah atau disebut diniyah terdiri dari Madrasah Idadiyah Nurul Qadim, TPQ An-Nuriyah, PDF Wustha Nurul Qadim, PDF Ulya Nurul Qadim dan Mahad Aly Nurul Qadim, dan ketiga adalah wilayah yang terdiri dari wilayah pondok putra, Banat 1, Banat 2, Pondok Anak-Anak dan Khadim. Dari ketiga kategori tersebut memiliki hak akses untuk melihat data, melakukan pencarian data, melakukan *export* data dan mencetak laporan data santri lembaga dalam bentuk pdf. Apabila admin lembaga menemukan kesalahan penulisan pada data santri maka admin lembaga dapat mengirimkan pesan ke admin pusat untuk melakukan perubahan pada data santri tersebut, serta akan mendapat konfirmasi pesan balasan jika data telah diperbaiki.

4.2.3 Desain Sistem

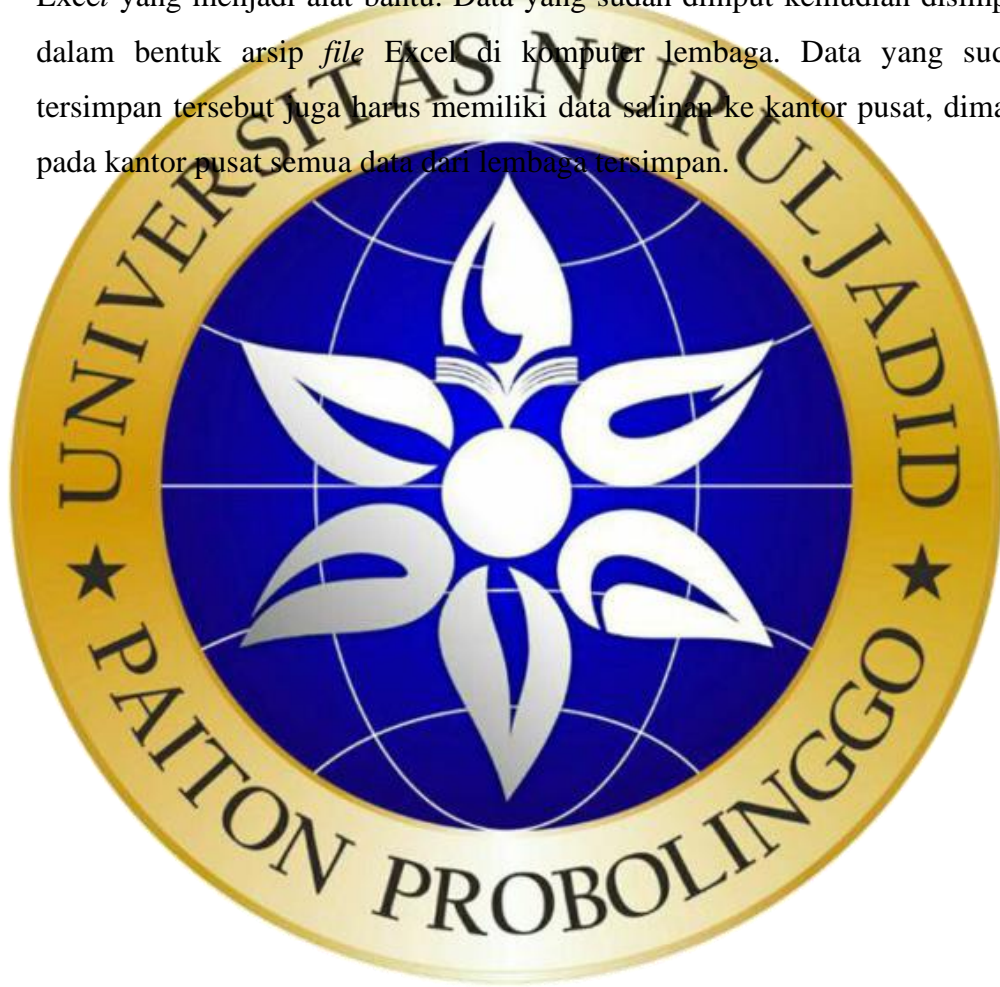
Desain sistem dimaksudkan untuk mengetahui dan memberikan gambaran sistem secara menyeluruh baik sistem yang sedang digunakan maupun yang akan diimplementasikan. Desain yang baru digambarkan dengan bagan alur (*flowchart*), *Data Flow Diagram* (DFD) dan *Entity Relationship Diagram* (ERD).

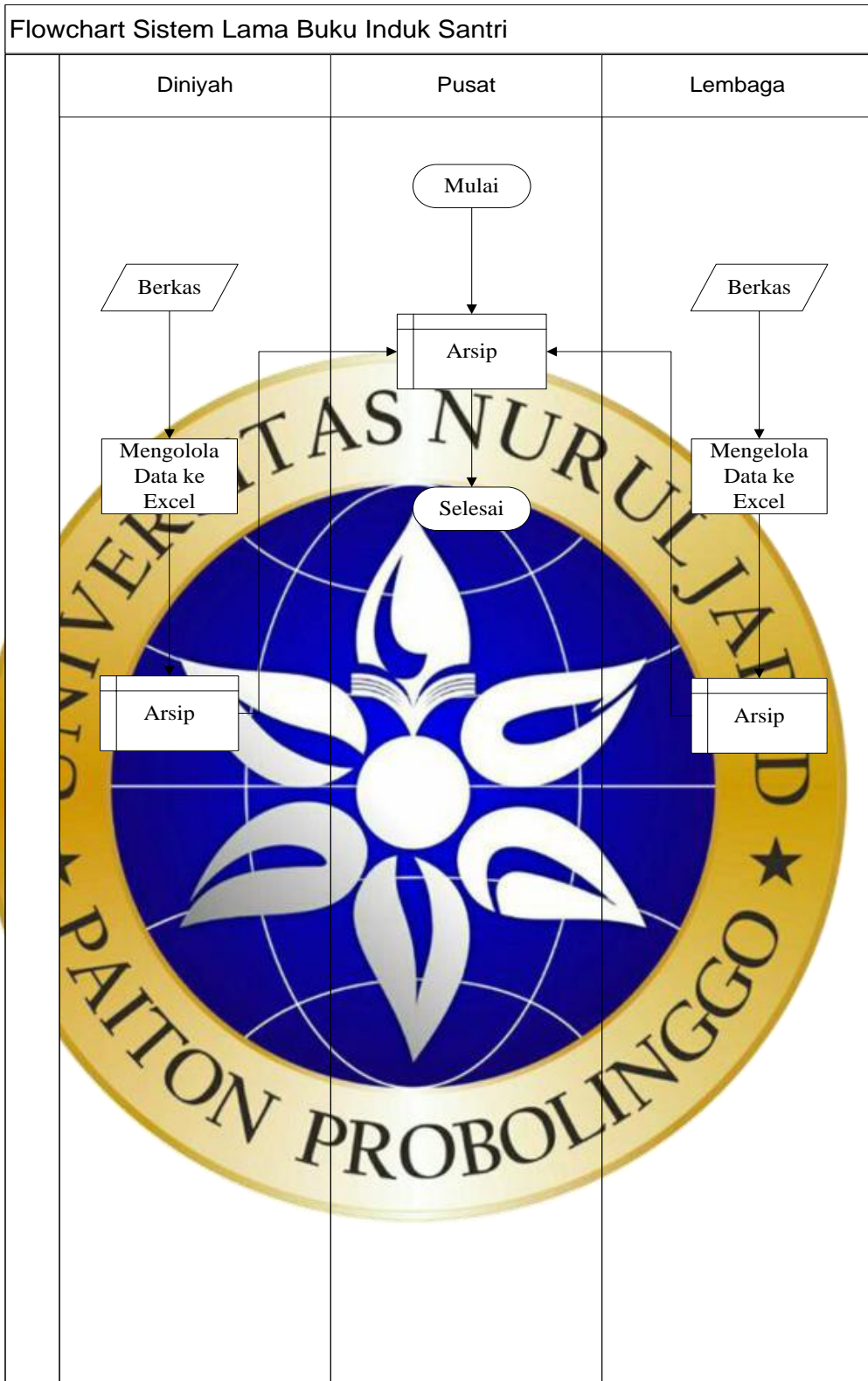
a. Bagan Alur (Flowchart)

Flowchart sistem baru dan sistem lama pada pengolahan data buku induk santri adalah sebagai berikut:

1. Flowchart sistem lama

Dari pengumpulan data yang dilakukan di pondok pesantren Nurul Qadim, adapun sistem yang berjalan selama ini yaitu penginputan data santri dilakukan oleh petugas lembaga dengan menggunakan *software* Microsoft Excel yang menjadi alat bantu. Data yang sudah diinput kemudian disimpan dalam bentuk arsip *file* Excel di komputer lembaga. Data yang sudah tersimpan tersebut juga harus memiliki data salinan ke kantor pusat, dimana pada kantor pusat semua data dari lembaga tersimpan.





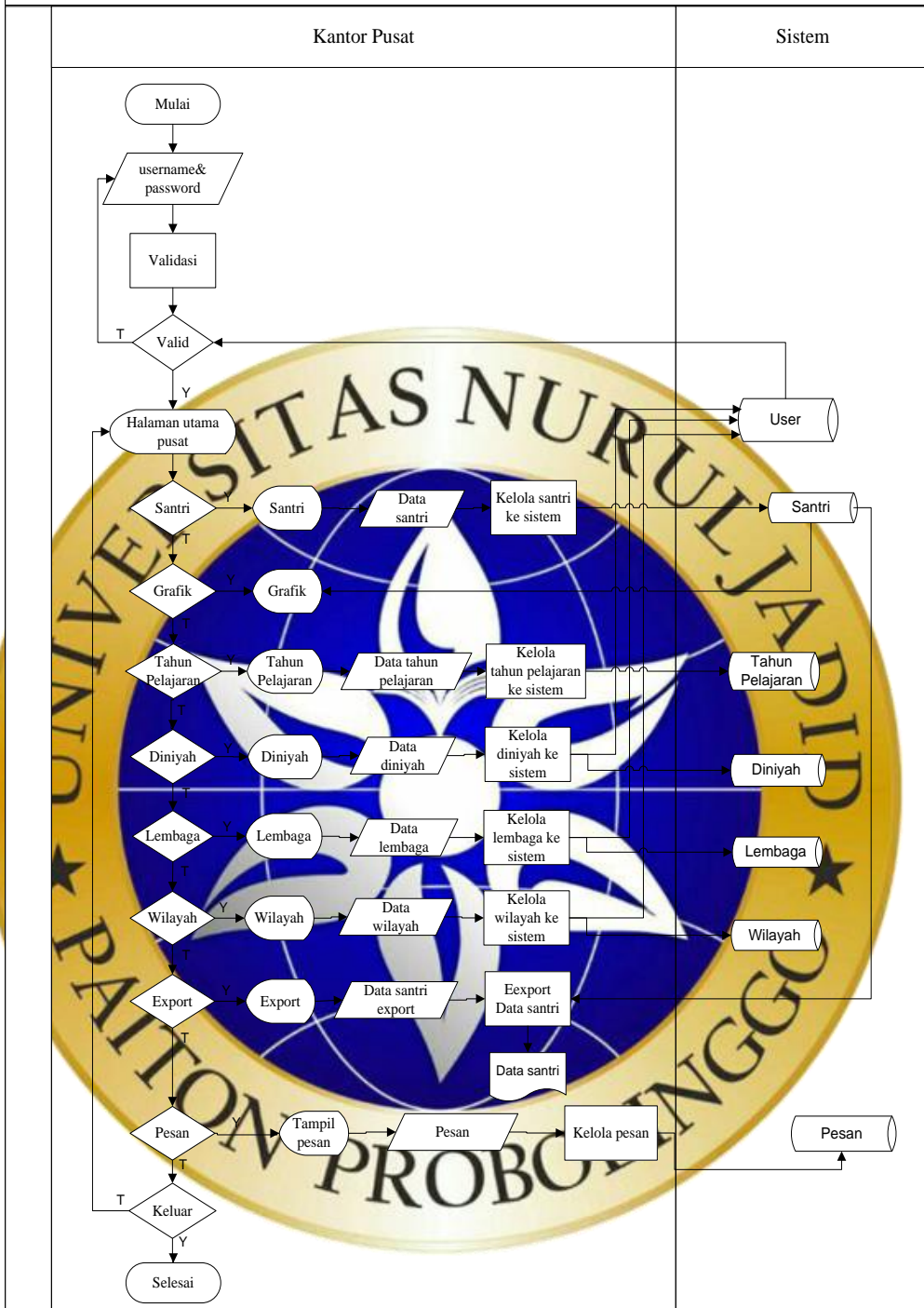
Gambar 4.1. Flowchart Sistem Lama

2. Flowchart sistem baru

Dari hasil pengumpulan data yang dilakukan di Pondok Pesantren Nurul Qadim pada penelitian ini maka akan dibuat aplikasi buku induk santri terpusat berbasis web yang diharapkan untuk membantu pengolahan data santri yang sedang berjalan. Seluruh pengolahan data seperti proses penginputan dan perubahan data dilakukan secara terpusat yang tersimpan dalam satu *database* sehingga dapat menghindari tidak sinkron data antara data kantor pusat dengan data lembaga.

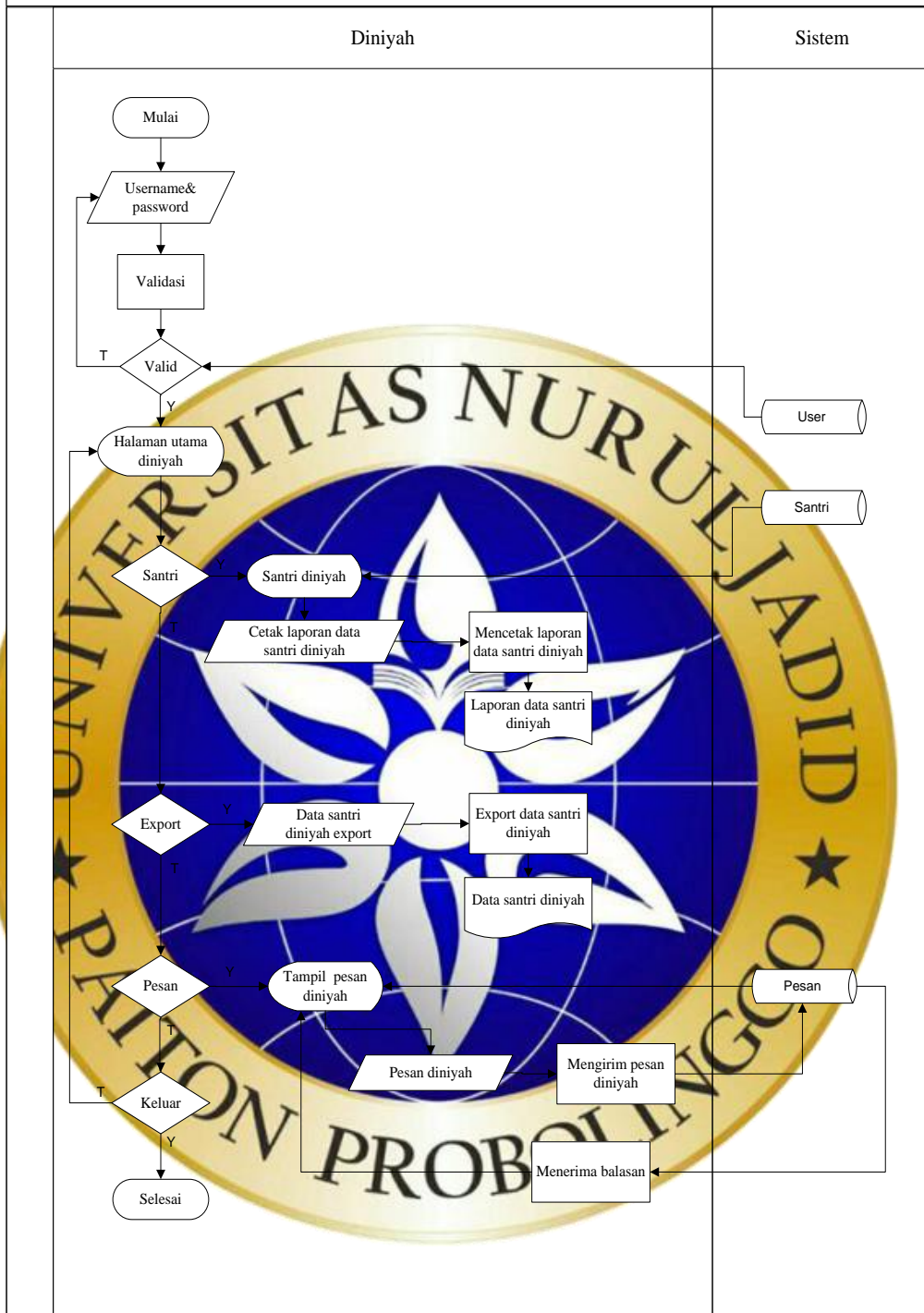


Flowchart Baru Aplikasi Buku Induk Santri Terpusat



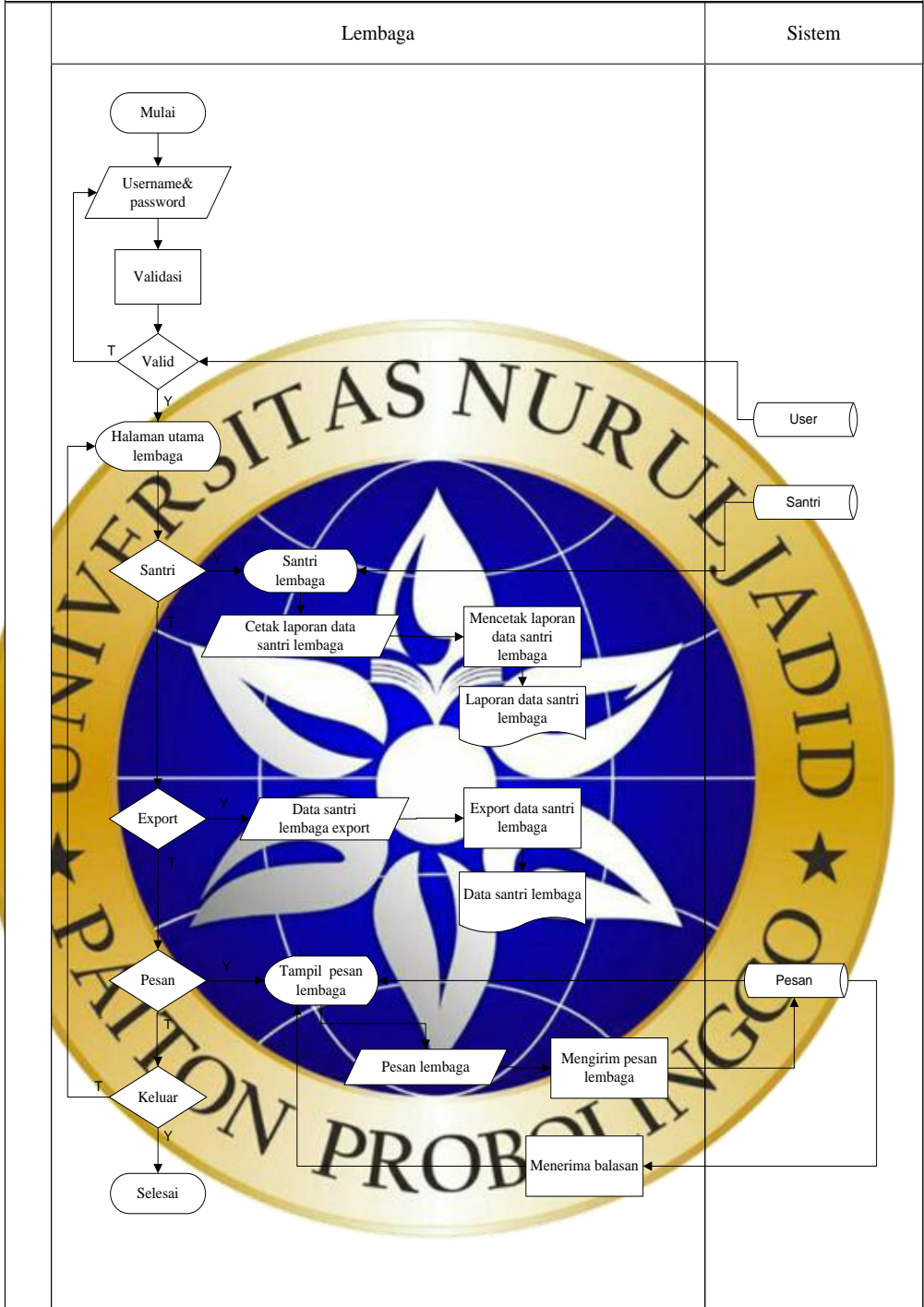
Gambar 4.2 Flowchart Baru Aplikasi Buku Induk 1

Flowchart Baru Aplikasi Buku Induk Santri Terpusat



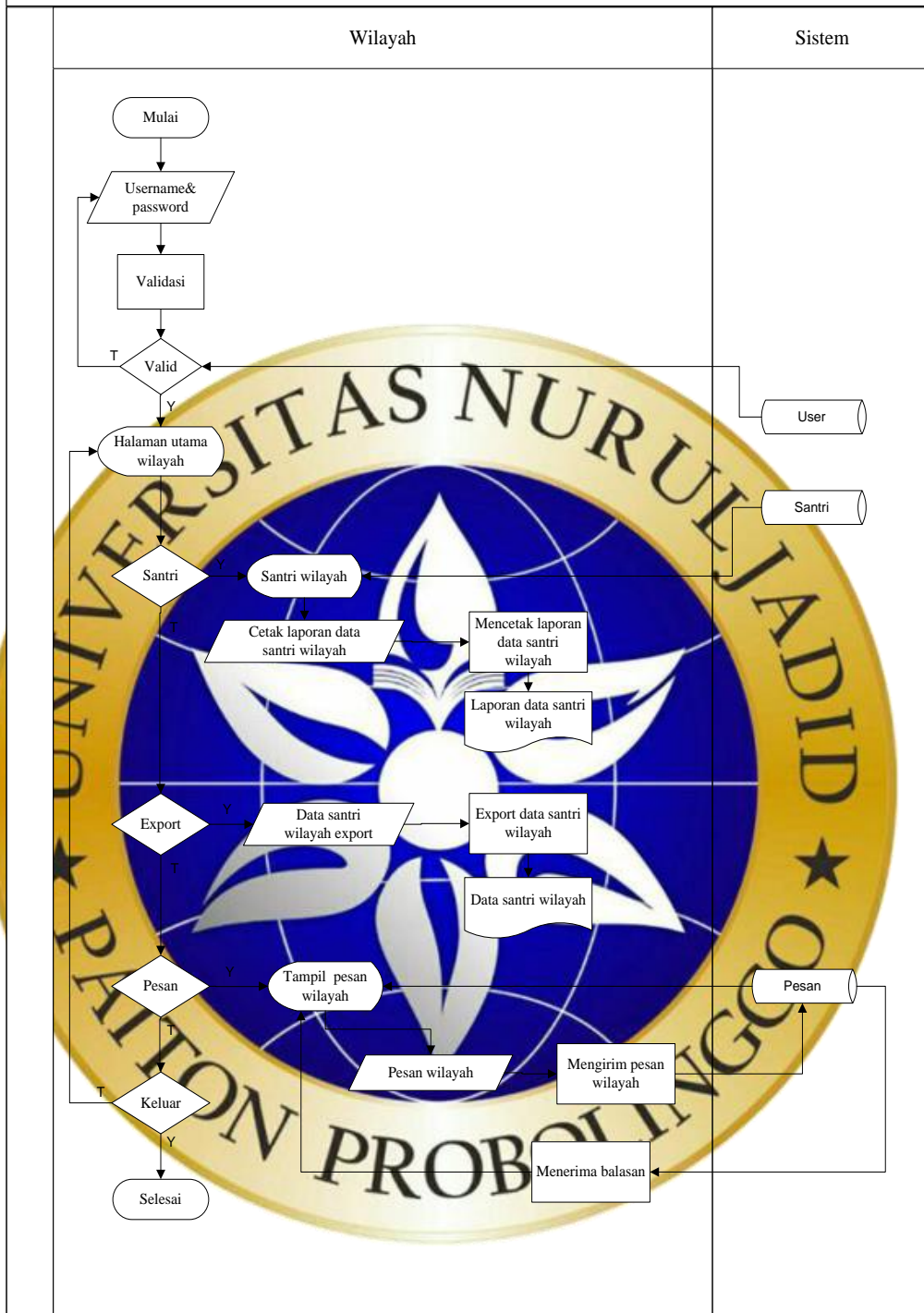
Gambar 4.3 Flowchart Baru Aplikasi Buku Induk Santri 2

Flowchart Baru Aplikasi Buku Induk Santri Terpusat



Gambar 4.4. Flowchart Baru Aplikasi Buku Induk 3

Flowchart Baru Aplikasi Buku Induk Santri Terpusat



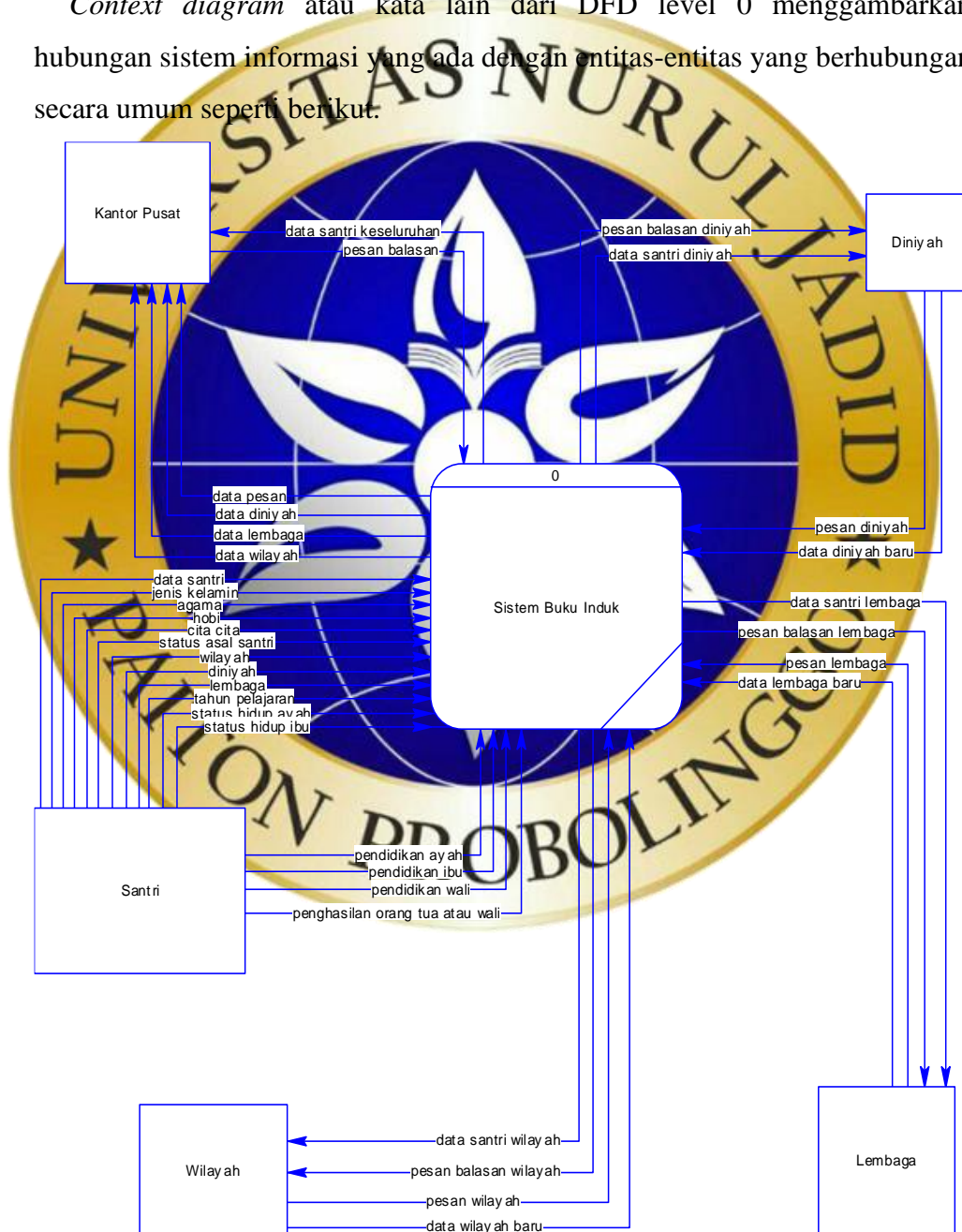
Gambar 4.5. Flowchart Baru Aplikasi Buku Induk 4

b. Data Flow Diagram (DFD)

Data Flow Diagram (DFD) adalah suatu bagan alur yang menunjukkan bagian alur proses pengolahan data dalam suatu sistem. *Data flow diagram* adalah representasi dari keterkaitan entitas, jenis data, penyimpanan dan proses dalam bentuk sistem yang menggambarkan suatu sistem mulai dari gambaran sistem yang paling umum sampai gambaran sistem yang terperinci.

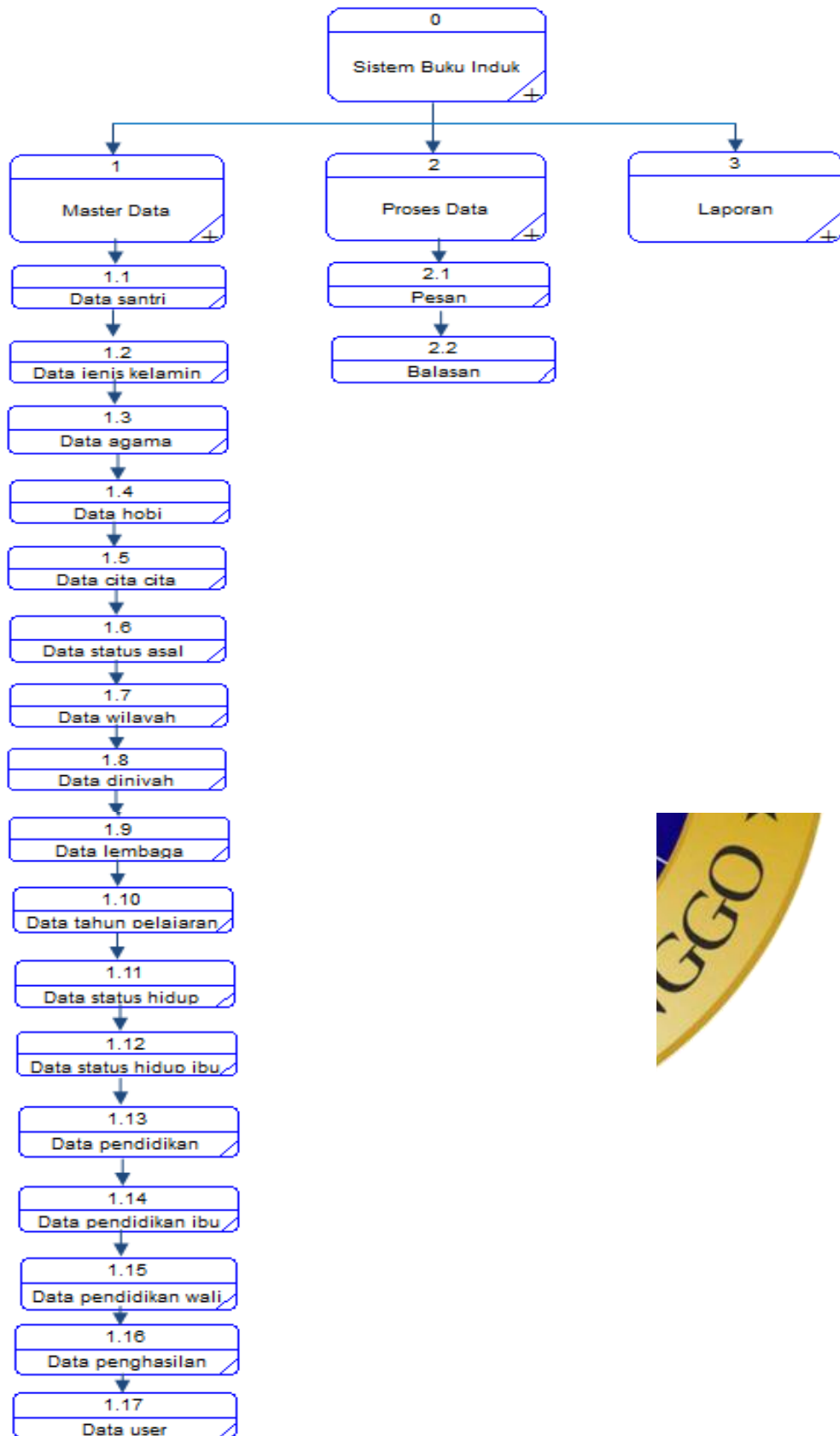
1. Context Diagram

Context diagram atau kata lain dari DFD level 0 menggambarkan hubungan sistem informasi yang ada dengan entitas-entitas yang berhubungan secara umum seperti berikut.



Gambar 4.6. Context Diagram

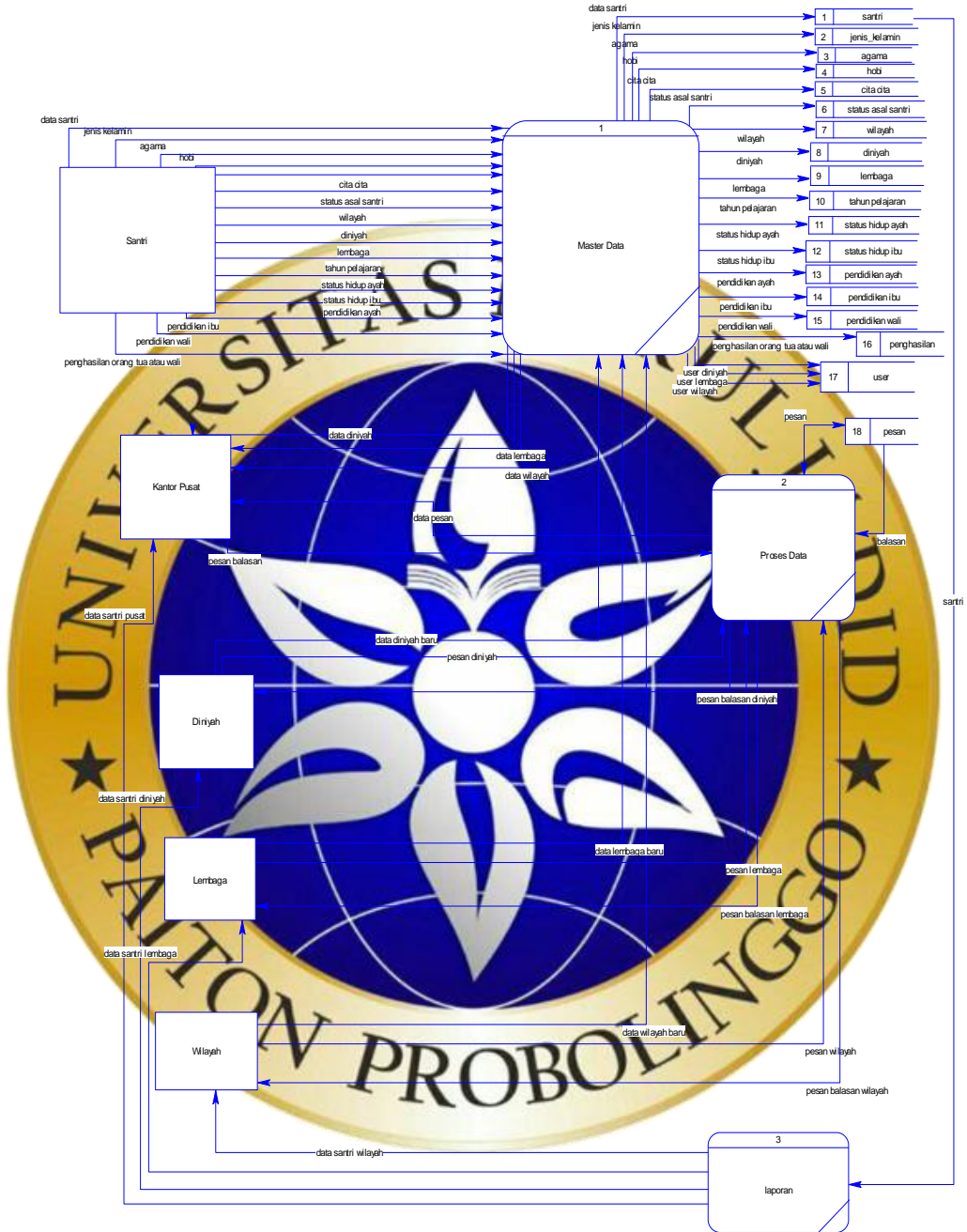
2. Bagan Berjenjang



Gambar 4.7. Bagan Berjenjang

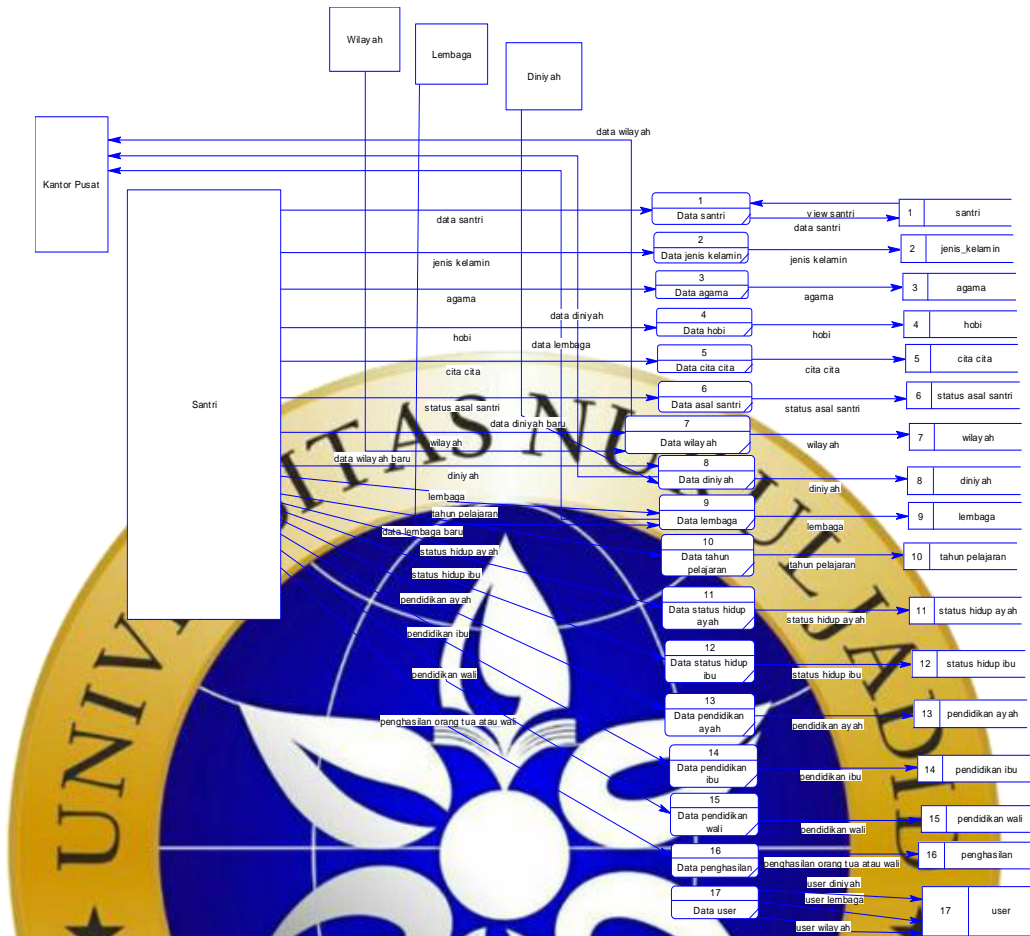
3. DFD level 1

DFD level 1 akan menjelaskan secara lebih rinci *context diagram* sistem buku induk santri kedalam beberapa proses sesuai dengan kebutuhan sistem.



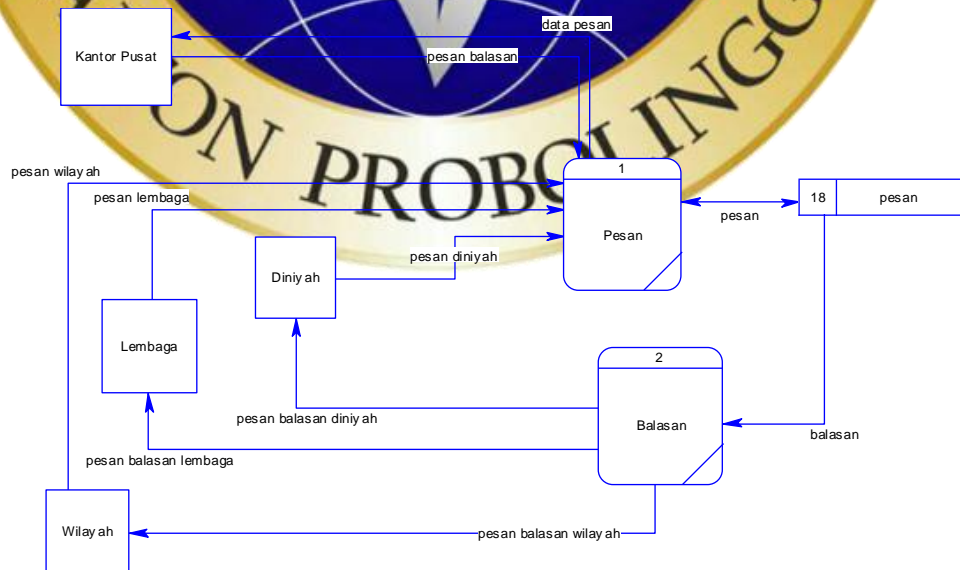
Gambar 4.8. DFD Level 1

4. DFD level 2 master data



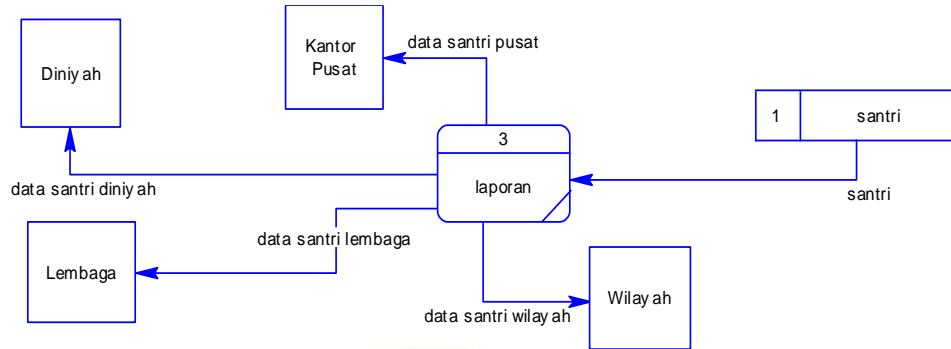
Gambar 4.9. DFD Level 2 Master Data

5. DFD level 2 proses data



Gambar 4.10. DFD Level 2 Proses Data

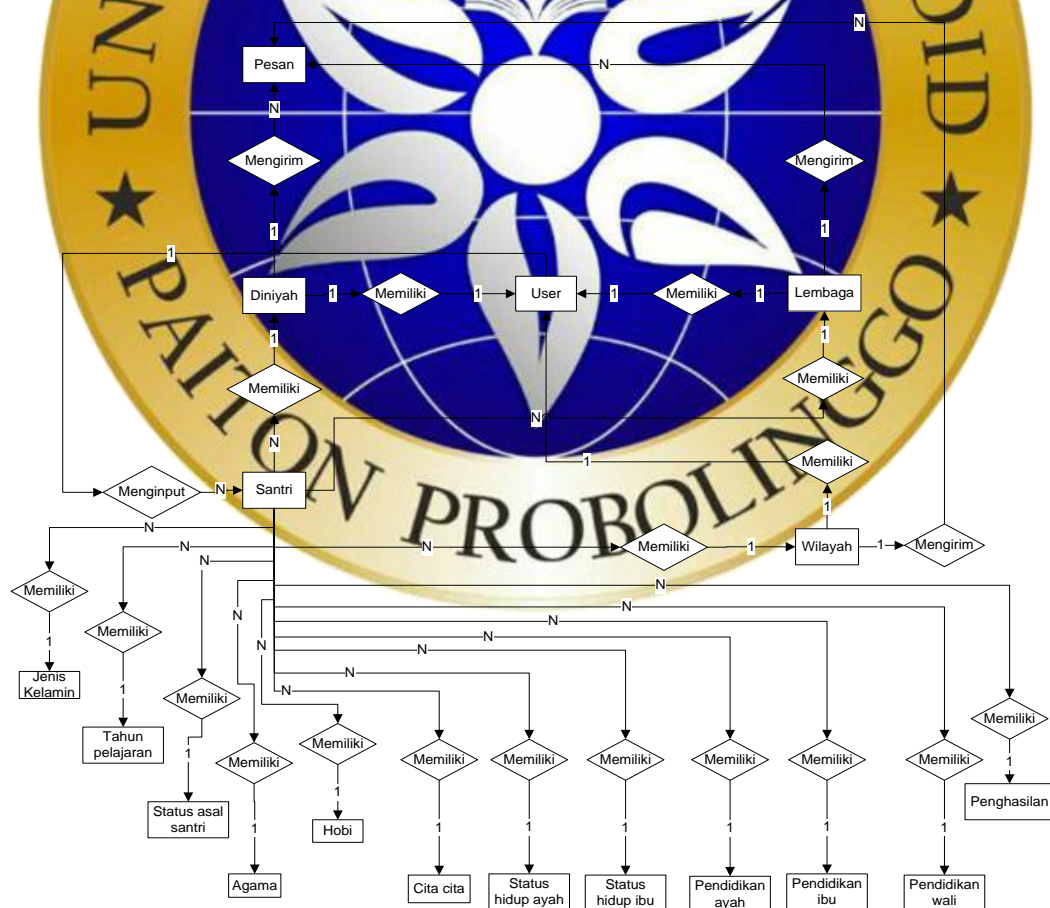
6. DFD level 2 laporan



Gambar 4.11. DFD Level 2 Laporan

c. Entity Relationship Diagram (ERD)

Tahapan *Entity Relationship Diagram* (ERD) adalah tahapan pemodelan data yang dibutuhkan oleh sistem yang digambarkan dengan entitas-entitas yang saling berhubungan. Setiap entitas memiliki kumpulan atribut yang mempresentasikan informasi dari sebuah record dari entitas tersebut.



Gambar 4.12. ERD

Kamus data:

1. Santri : id*, nis, nisl, nik, no_kk, nama, tempat_lahir, tanggal_lahir, jenis_kelamin**, agama**, alamat, kecamatan, kabupaten, propinsi, hobi**, cita_cita**, anak_ke, jumlah_saudara, tanggal_masuk_pesantren, status_asal_santri**, sekolah_asal, alamat_sekolah_asal, santri_aktif, wilayah**, lembaga**, diniyah**, tahun_pelajaran**, foto, nama_ayah, status_hidup_ayah**, pendidikan_ayah**, pekerjaan_ayah, no_hp_ayah, nama_ibu, status_hidup_ibu**, pendidikan_ibu**, pekerjaan_ibu, no_hp_ibu, nama_wali, pendidikan_wali**, pekerjaan_wali, no_hp_wali, penghasilan_orang_tua_atau_wali**.
2. Pesan : id*, lembaga**, diniyah**, wilayah**, keterangan, status, pesan_balasan, create_at.
3. Wilayah : id*, user, nama, keterangan.
4. Lembaga : id*, user, nama, keterangan.
5. diniyah : id*, user, nama, keterangan.
6. Tahunpelajaran : id*, tahun_pelajaran.
7. Jeniskelamin : id*, jenis.
8. Hobi : id*, hobi.
9. Agama : id*, agama.
10. Citacita : id*, cita.
11. Statusasalsantri : id*, status_asal_santri.
12. Statushidupayah : id*, status_hidup_ayah.
13. Statushidupibu : id*, status_hidup_ibu.
14. Pendidikanayah : id*, pendidikan_ayah.
15. Pendidikanayah : id*, pendidikan_ibu.
16. Pendidikanwali : id*, pendidikan_wali.
17. Penghasilan : id*, pennghasilan.
18. User : id*, username, password.

d. Struktur Database

Database adalah pusat data yang akan diolah dalam sistem informasi. Dalam *database* akan menampung seluruh jenis data sehingga *database* harus memiliki pola struktur yang baik agar tidak terjadi redundansi dan inkonsistensi data.

Database yang akan dipakai dibuat dengan menggunakan MySQL yang terdiri dari beberapa tabel. Berikut struktur tabel *database* yang akan dipakai sebagai pusat penyimpanan data yang akan dijadikan sumber data.

Tabel 4.1. Tabel Santri

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Nis	varchar(50)	
3	Nisn	varchar(50)	
4	Nik	varchar(50)	
5	no_kk	varchar(50)	
6	Nama	varchar(200)	
7	tempat_lahir	varchar(50)	
8	tanggal_lahir	Date	
9	jenis_kelamin	int(11)	Foreign key
10	Agama	int(11)	Foreign key
11	Alamat	Longtext	
12	Desa	varchar(50)	
13	Kecamatan	varchar(30)	
14	Kabupaten	varchar(30)	
15	Propinsi	varchar(30)	
16	Hobi	int(11)	Foreign key
17	cita_cita	int(11)	Foreign key
18	anak_ke	int(10)	
19	jumlah_saudara	int(10)	
20	tanggal_masuk_pesantren	Date	
21	status_asal_santri	int(11)	Foreign key
22	sekolah_asal	varchar(100)	
23	alamat_sekolah_asal	Longtext	
24	santri_aktif	tinyint(1)	
25	Wilayah	int(11)	Foreign key
26	Lembaga	int(11)	Foreign key

Tabel 4.2. Tabel Santri (Lanjutan)

27	Diniyah	int(11)	Foreign key
28	tahun_pelajaran	int(11)	Foreign key
29	Foto	varchar(100)	
30	nama_ayah	varchar(200)	
31	status_hidup_ayah	int(11)	Foreign key
32	pendidikan_ayah	int(11)	Foreign key
33	pekerjaan_ayah	varchar(50)	
34	no_hp_ayah	varchar(20)	
35	nama_ibu	varchar(200)	
36	status_hidup_ibu	int(11)	Foreign key
37	pendidikan_ibu	int(11)	Foreign key
38	pekerjaan_ibu	varchar(50)	
39	no_hp_ibu	varchar(20)	
40	nama_wali	varchar(200)	
41	pendidikan_wali	int(11)	Foreign key
42	pekerjaan_wali	varchar(50)	
43	no_hp_wali	varchar(20)	
44	penghasilan_orang_tua_atau_wali_perbulan	int(11)	Foreign key

Tabel 4.3. Tabel Pesan

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Lembaga	int(11)	Foreign key
3	Diniyah	int(11)	Foreign key
4	Wilayah	int(11)	Foreign key
5	Keterangan	varchar(500)	
6	Status	int(11)	
7	pesan_balasan	varchar(500)	
8	created_at	Date	

Tabel 4.4. Tabel Wilayah

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	User	int(11)	
3	Nama	varchar(200)	
4	Keterangan	varchar(500)	

Tabel 4.5. Tabel Lembaga

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	User	int(11)	
3	Nama	varchar(200)	
4	Keterangan	varchar(500)	

Tabel 4.6. Tabel Diniyah

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	User	int(11)	
3	Nama	varchar(200)	
4	Keterangan	varchar(500)	

Tabel 4.7. Tabel Tahun Pelajaran

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	tahun_pelajaran	varchar(50)	

Tabel 4.8. Tabel Jenis Kelamin

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Jenis	varchar(50)	

Tabel 4.9. Tabel Hobi

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Hobi	varchar(100)	

Tabel 4.10. Tabel Agama

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Agama	varchar(100)	

Tabel 4.11. Tabel Cita-cita

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Cita	varchar(100)	

Tabel 4.12. Tabel Status Asal Santri

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	status_asal_santri	varchar(200)	

Tabel 4.13. Tabel Status Hidup Ibu

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	status_hidup_ibu	varchar(100)	

Tabel 4.14. Tabel Status Hidup Ayah

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	status_hidup_ayah	varchar(100)	

Tabel 4.15. Tabel Pendidikan Ayah

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	pendidikan_ayah	varchar(100)	

Tabel 4.16. Tabel Pendidikan Ibu

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	pendidikan_ibu	varchar(100)	

Tabel 4.17. Tabel Pendidikan Wali

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	pendidikan_wali	varchar(100)	

Tabel 4.18. Tabel Penghasilan

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Penghasilan	varchar(100)	

Tabel 4.19. Tabel User

No	Nama	Jenis	Key
1	Id	int(11)	Primary key
2	Username	varchar(150)	
3	Password	Varchar(128)	

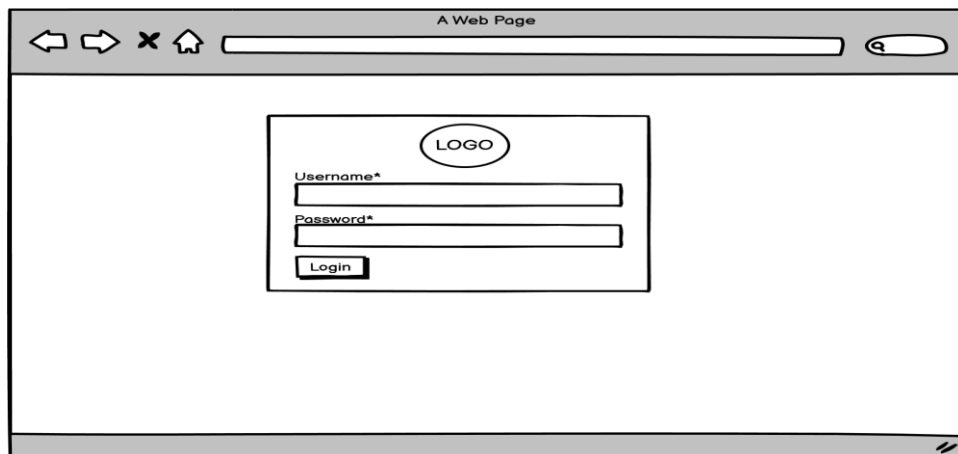
e. Desain Interface

Desain *interface* adalah desain tampilan antar muka program aplikasi. Karena masih berupa desain maka bisa jadi dalam program sebenarnya terdapat penambahan atau pengurangan, namun penambahan dan pengurangan tersebut tidak akan terlepas dari fungsi pokok masing-masing prosedur. Desain tersebut adalah sebagai berikut:

1. Login

Login adalah halaman utama saat sistem web pertama kali dijalankan. Pada halaman ini pengguna harus memasukkan *username* dan *password* untuk melakukan verifikasi data. Terdapat empat kategori pengguna yang dapat melakukan login pada sistem, yaitu admin kantor

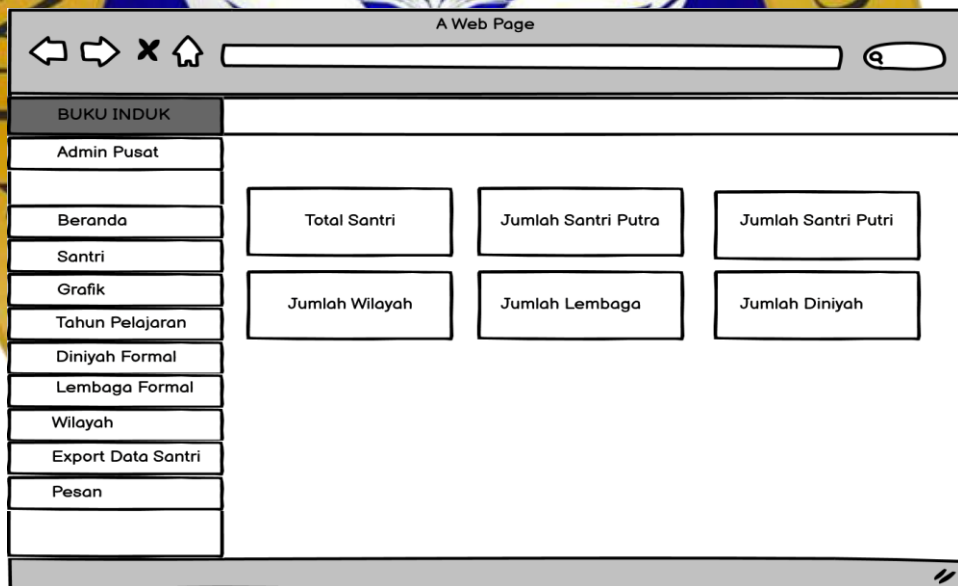
pusat, diniyah, lembaga dan wilayah. Tampilan desain halaman masuk sistem dapat dilihat pada gambar 4.13.



Gambar 4.13. Desain Login

2. Menu utama admin pusat

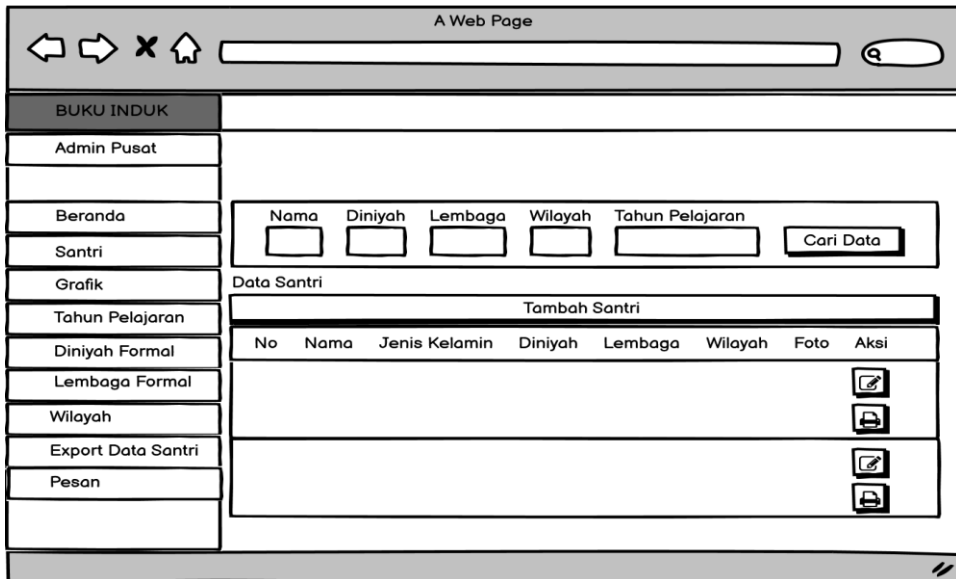
Setelah kantor pusat berhasil melakukan login maka menu utama yang akan tampil seperti pada gambar 4.14.



Gambar 4.14. Desain Menu Utama Admin Pusat

3. Form menu santri admin pusat

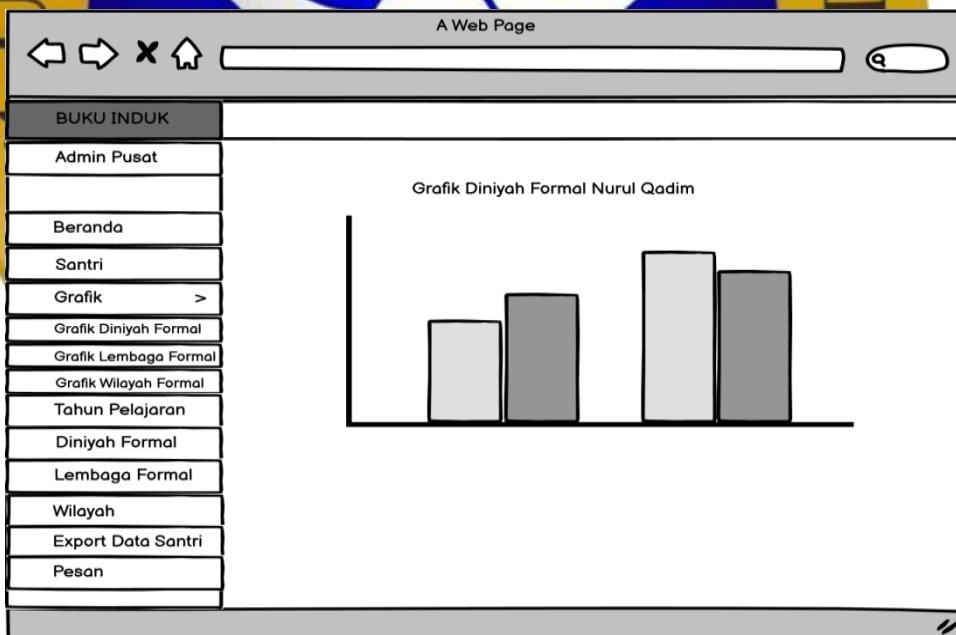
Desain form menu santri admin pusat adalah halaman yang akan menampilkan data santri yang sudah diinputkan. Selain itu juga, dapat melakukan pencarian santri berdasarkan kategori (nama, diniyah, lembaga, wilayah dan tahun pelajaran), mencetak laporan data santri dan melakukan edit data santri jika terdapat kesalahan dalam data santri tersebut.



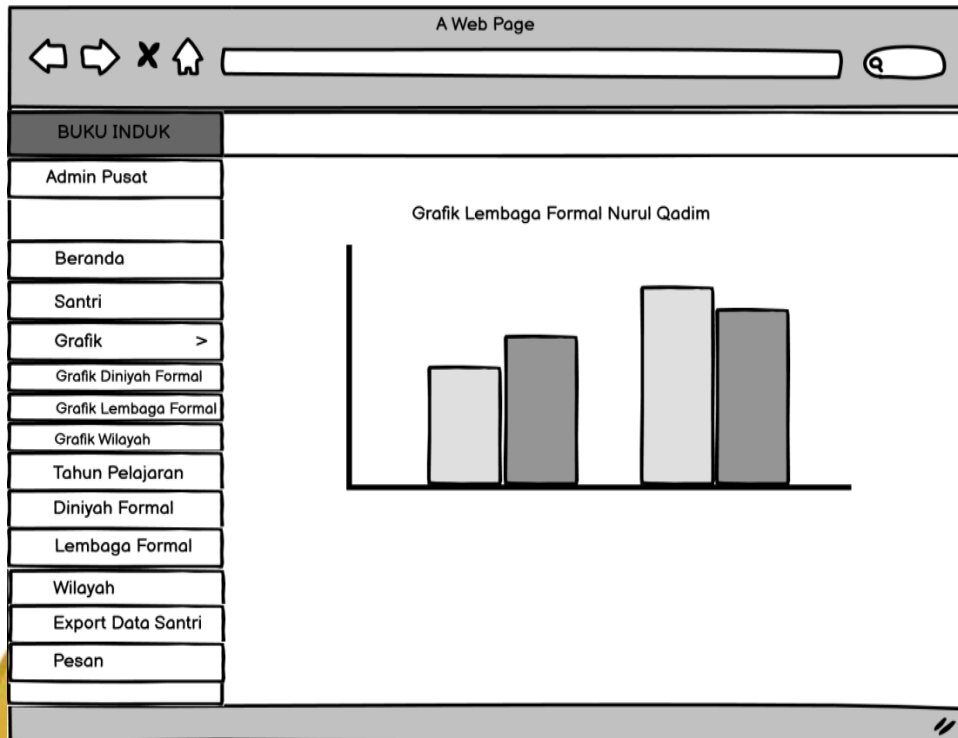
Gambar 4.15 Desain Menu Santri Admin Pusat

4. Form menu grafik

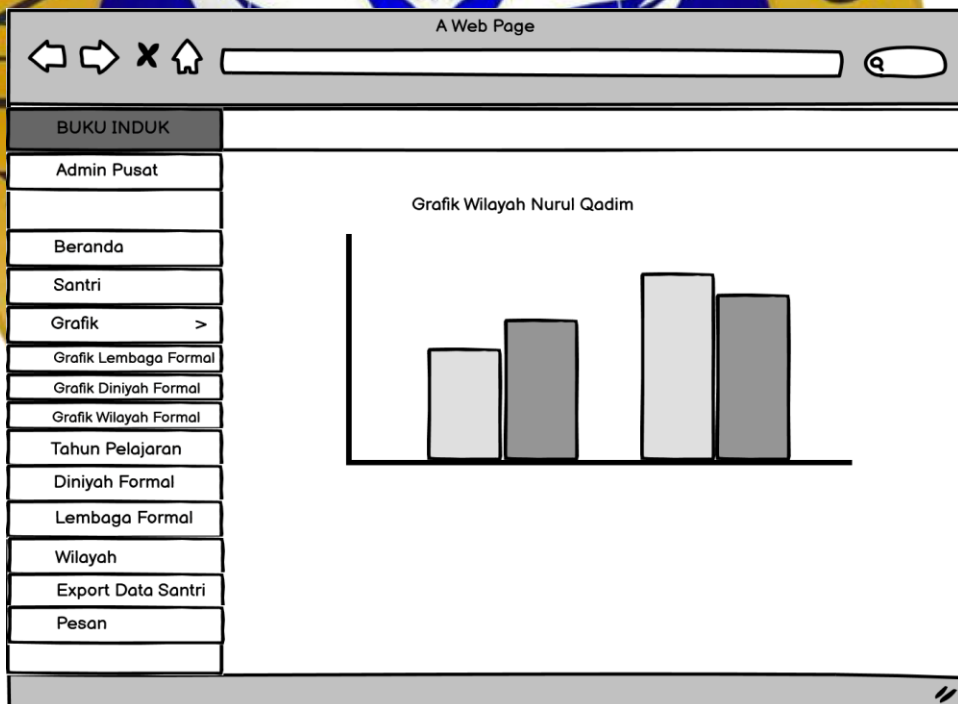
Desain form menu grafik adalah halaman admin pusat yang akan menampilkan grafik dari jumlah data santri berdasarkan lembaga, diniyah dan wilayah.



Gambar 4.16 Desain Menu Grafik Diniyah



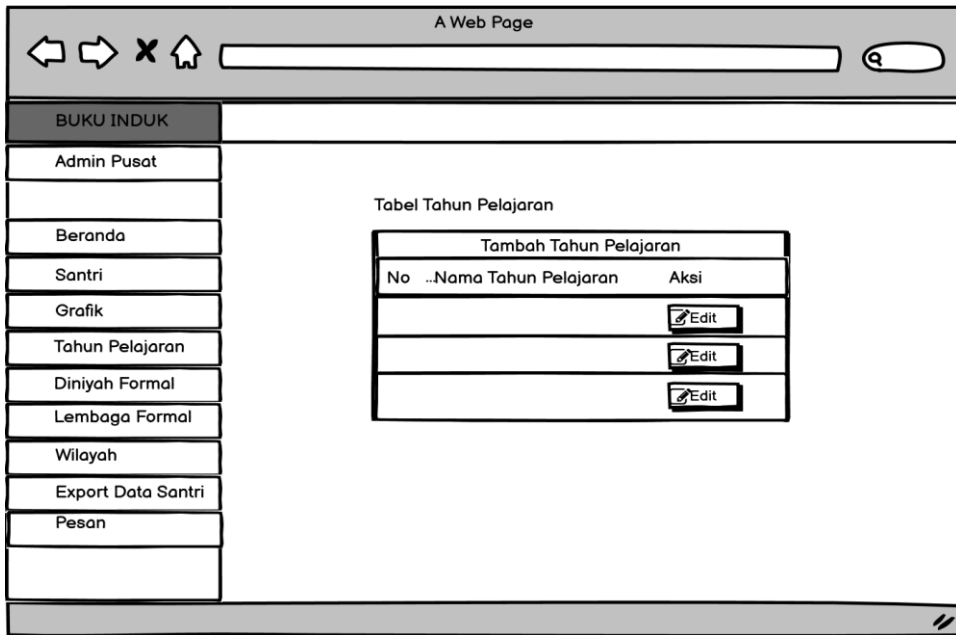
Gambar 4.17. Desain Menu Grafik Lembaga



Gambar 4.18. Desain Menu Grafik Wilayah

5. Form menu tahun pelajaran admin

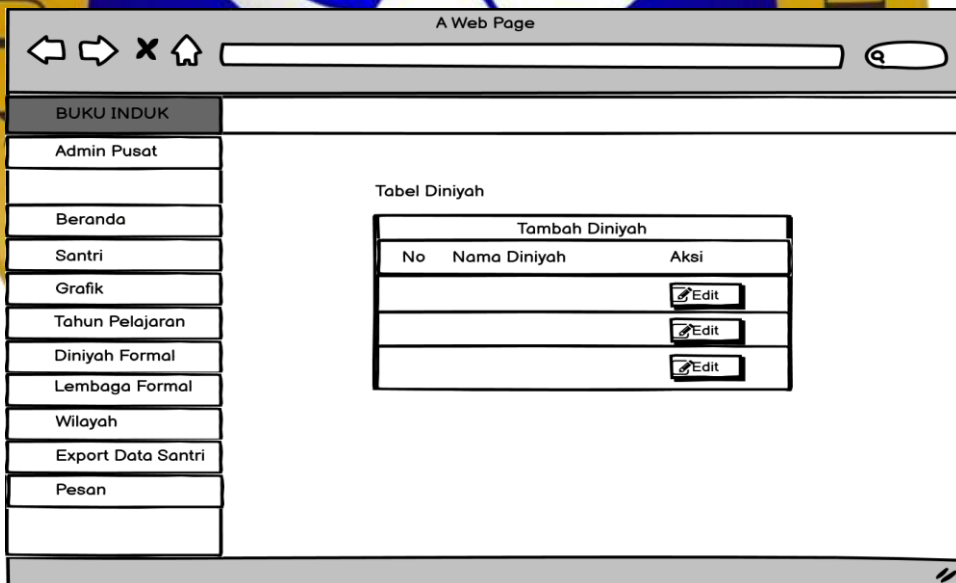
Desain form menu tahun pelajaran adalah halaman admin pusat yang akan menampilkan data tahun pelajaran yang sudah diinputkan.



Gambar 4.19. Desain Menu Tahun Pelajaran

6. Form menu diniyah

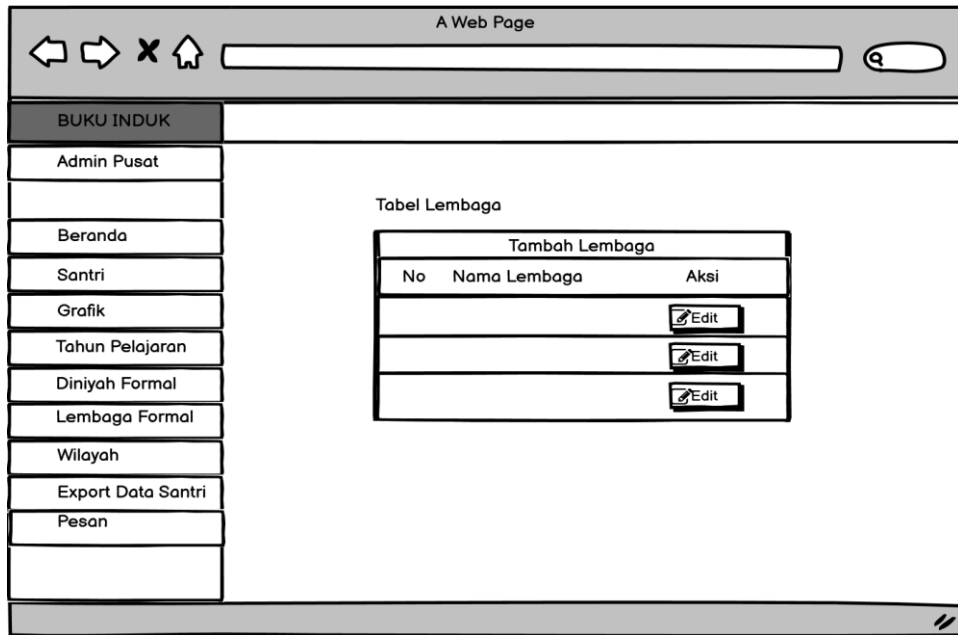
Desain form menu diniyah adalah halaman admin pusat yang akan menampilkan data diniyah yang sudah diinputkan.



Gambar 4.20. Desain Menu Diniyah

7. Form menu lembaga

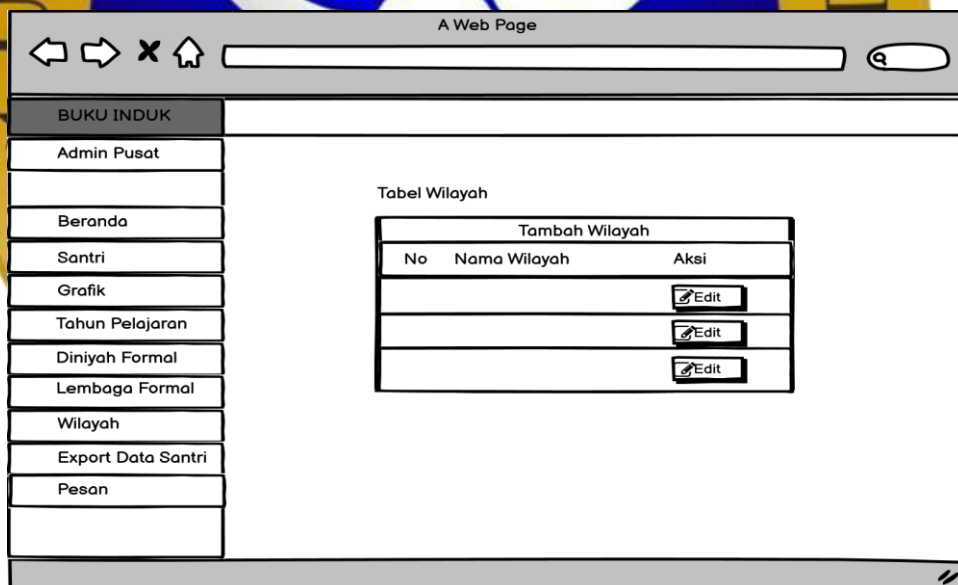
Desain form menu lembaga adalah halaman admin pusat yang akan menampilkan data lembaga yang sudah diinputkan.



Gambar 4.21. Desain Menu Lembaga

8. Form menu wilayah

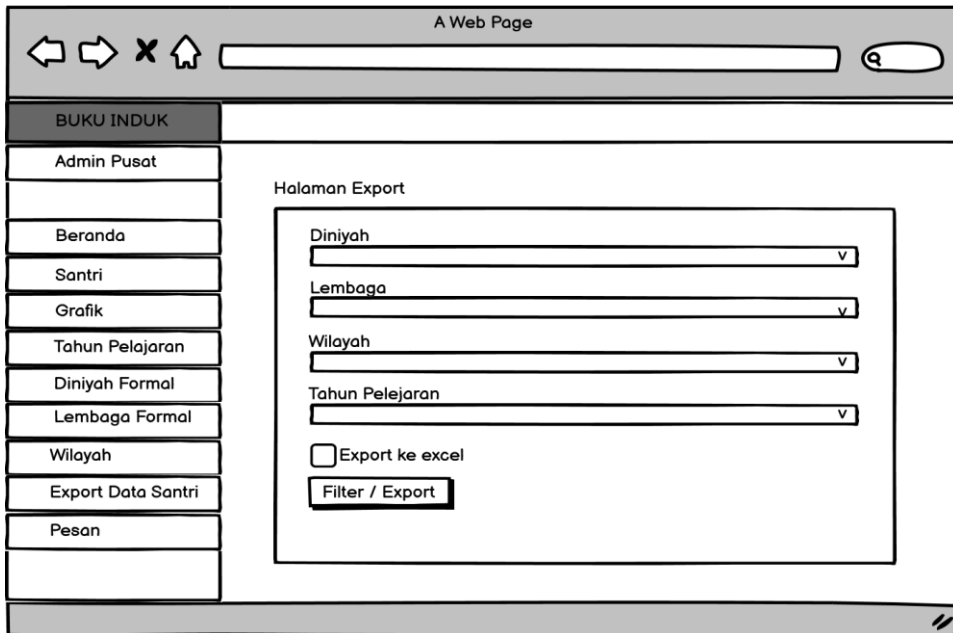
Desain form menu wilayah adalah halaman admin pusat yang akan menampilkan data wilayah yang sudah diinputkan.



Gambar 4.22. Desain Menu Wilayah

9. Form export data santri admin pusat

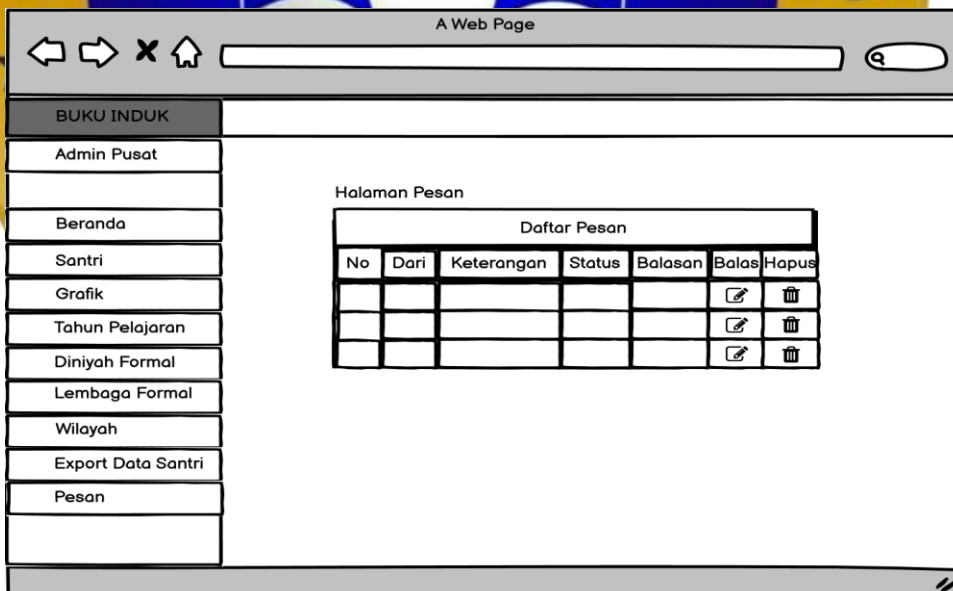
Desain form menu *export* data santri admin pusat adalah halaman admin pusat untuk melakukan *export* data santri ke bentuk *excel*.



Gambar 4.23. Desain Menu *Export Data Santri* Admin Pusat

10. Form menu pesan admin pusat

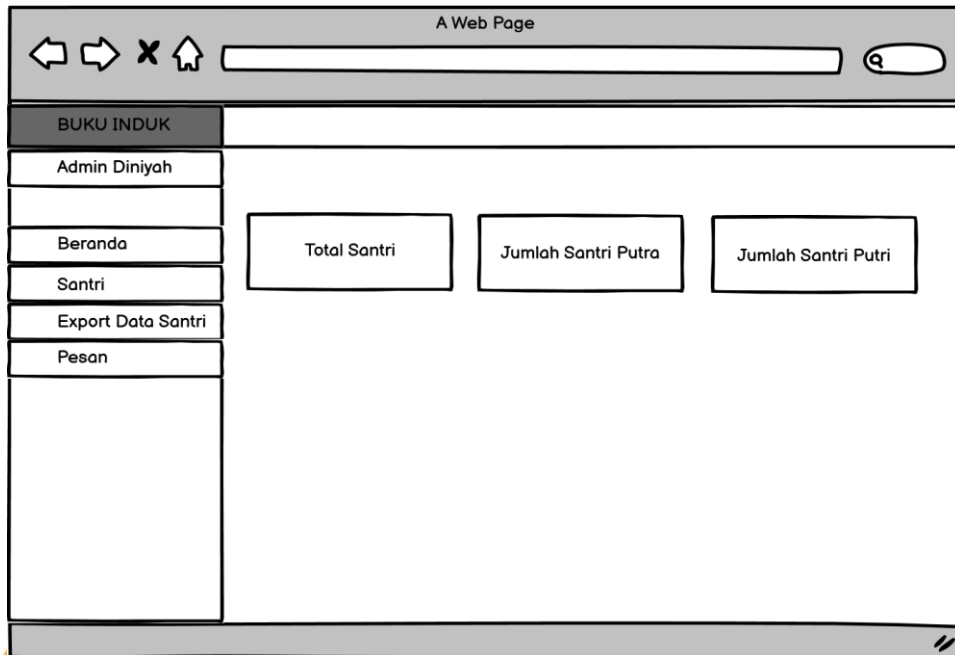
Desain form menu pesan admin pusat adalah halaman admin pusat yang akan menampilkan data pesan dari diniyah, lembaga, dan wilayah, serta melakukan pengolahan dan konfirmasi pada pesan tersebut.



Gambar 4.24. Desain Menu Pesan Admin Pusat

11. Menu utama admin diniyah

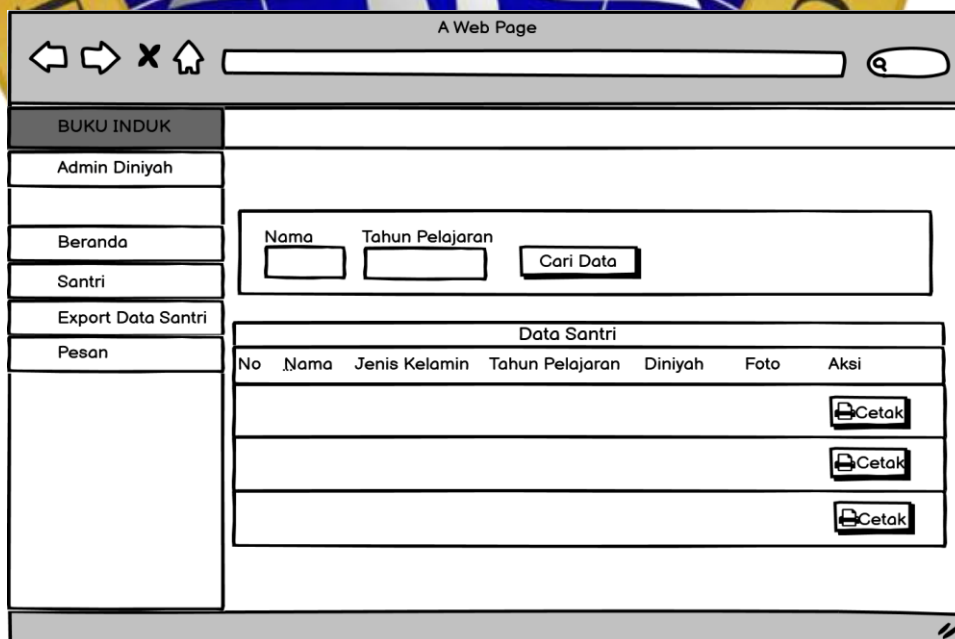
Setelah admin diniyah berhasil melakukan login maka menu utama yang akan tampil seperti pada gambar 4.25.



Gambar 4.25. Desain Menu Utama Admin Diniyah

12. Form menu santri admin diniyah

Desain form menu santri admin diniyah adalah halaman admin diniyah yang akan menampilkan data santri diniyah yang sudah diinputkan oleh admin pusat. Selain itu juga, dapat melakukan pencarian santri diniyah sesuai kategori (nama dan tahun pelajaran) dan mencetak laporan data santri diniyah.



Gambar 4.26. Desain Menu Santri Admin Diniyah

13. Form export data santri admin diniyah

Desain form menu *export* data santri admin diniyah adalah halaman admin diniyah untuk melakukan *export* data santri diniyah ke bentuk *excel*.

The screenshot shows a web browser window titled 'A Web Page'. On the left is a sidebar menu with the following items: 'Admin Diniyah', 'Beranda', 'Santri', 'Export Data Santri' (highlighted), and 'Pesan'. The main content area is titled 'Halaman Export' and contains a form with a dropdown menu labeled 'Tahun pelajaran', a checkbox labeled 'Export ke excel', and a button labeled 'Filter / Export'.

Gambar 4.27. Desain Menu *Export* Data Santri Admin Diniyah

14. Form menu pesan admin diniyah

Desain form menu pesan diniyah adalah halaman admin diniyah untuk mengirimkan pesan diniyah ke admin pusat jika terdapat kesalahan penulisan pada data santri diniyah. Selain itu juga, dapat melihat status pesan beserta konfirmasi balasan dari admin pusat.

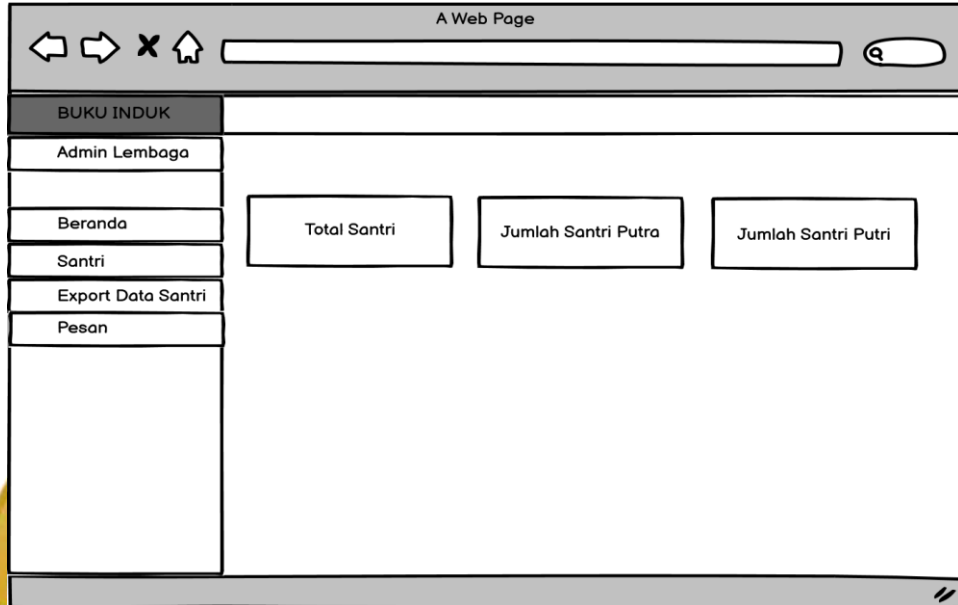
The screenshot shows a web browser window titled 'A Web Page'. On the left is a sidebar menu with the following items: 'Admin Diniyah', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan' (highlighted). The main content area is titled 'Halaman Pesan' and contains a table with the following structure:

Tambah atau kirim pesan			
No	Pesan	Status	Balasan

Gambar 4.28. Desain Menu Pesan Admin Diniyah

15. Menu utama admin lembaga

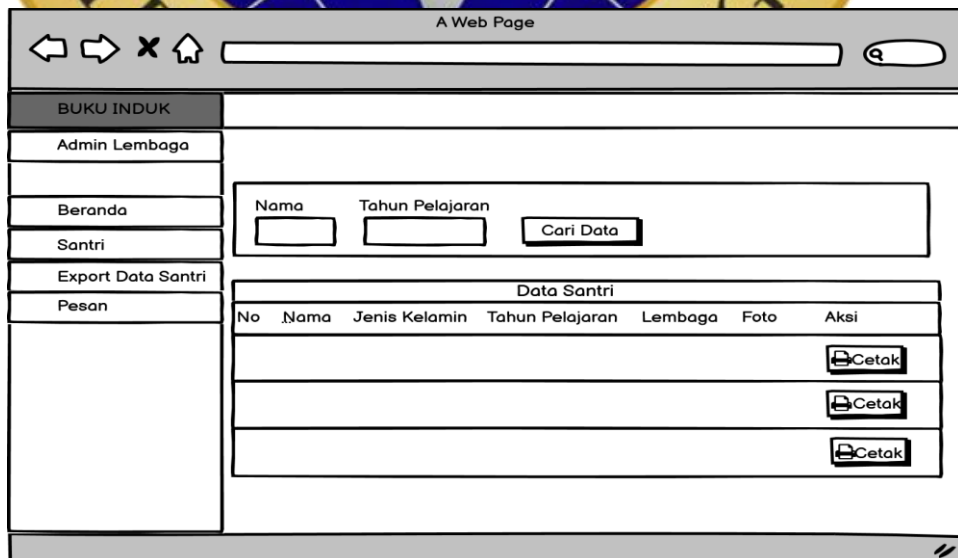
Setelah admin lembaga berhasil melakukan login maka menu utama yang akan tampil seperti pada gambar 4.29.



Gambar 4.29. Desain Menu Utama Admin Lembaga

16. Form menu santri admin lembaga

Desain form menu santri admin lembaga adalah halaman admin lembaga yang akan menampilkan data santri lembaga yang sudah diinputkan oleh admin pusat. Selain itu juga, dapat melakukan pencarian santri lembaga sesuai kategori (nama dan tahun pelajaran) dan mencetak laporan data santri lembaga.



Gambar 4.30. Desain Menu Santri Admin Lembaga

17. Form export data santri admin lembaga

Desain form menu *export* data santri admin lembaga adalah halaman admin lembaga untuk melakukan *export* data santri lembaga ke bentuk *excel*.

The screenshot shows a web browser window titled 'A Web Page'. On the left is a sidebar menu with the following items: 'Admin Lembaga', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan'. The main content area is titled 'Halaman Export' and contains a form with a dropdown menu labeled 'Tahun pelajaran', a checkbox labeled 'Export ke excel', and a button labeled 'Filter / Export'.

Gambar 4.31. Desain Menu *Export* Data Santri Admin Lembaga

18. Form menu pesan admin lembaga

Desain form menu pesan admin lembaga adalah halaman admin lembaga untuk mengirimkan pesan lembaga ke admin pusat jika terdapat kesalahan penulisan pada data santri lembaga. Selain itu juga, dapat melihat status pesan beserta pesan balasan dari admin pusat.

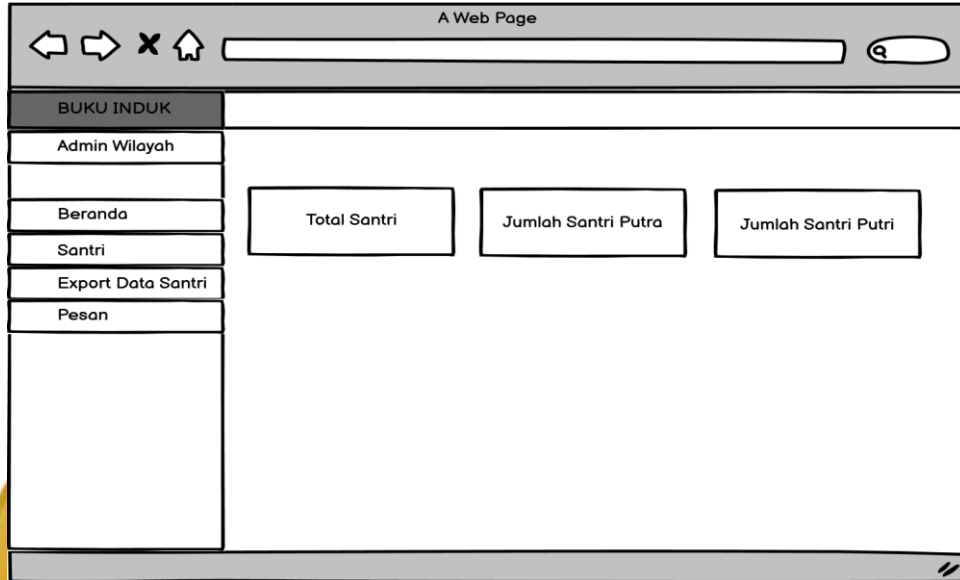
The screenshot shows a web browser window titled 'A Web Page'. On the left is a sidebar menu with the following items: 'Admin Lembaga', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan'. The main content area is titled 'Halaman Pesan' and contains a table with the following structure:

Tambah atau kirim pesan			
No	Pesan	Status	Balasan

Gambar 4.32. Desain Menu Pesan Admin Lembaga

19. Menu utama admin wilayah

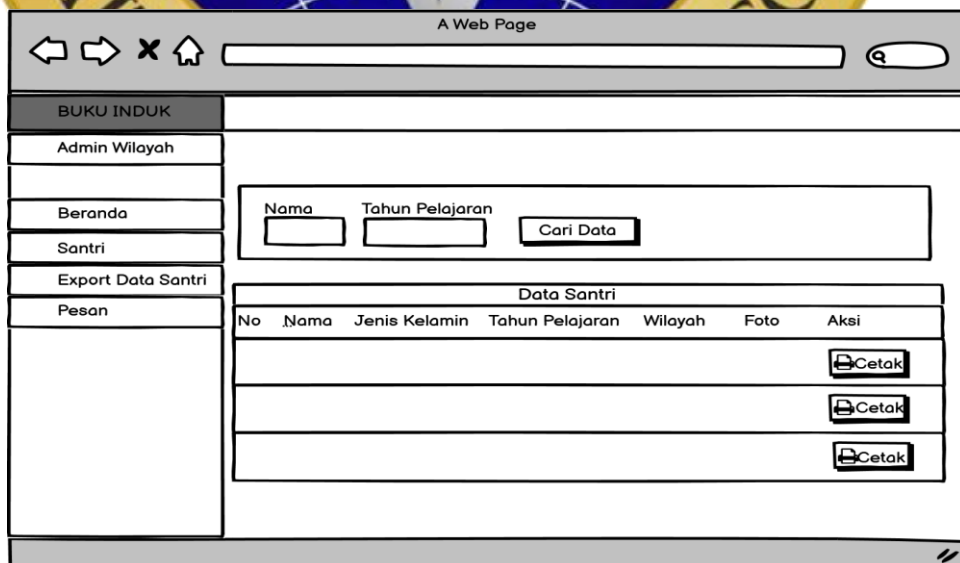
Setelah admin wilayah berhasil melakukan login maka menu utama yang akan tampil seperti pada Gambar 4.33.



Gambar 4.33. Desain Menu Utama Admin Wilayah

20. Form menu santri admin wilayah

Desain form menu santri admin wilayah adalah halaman admin wilayah yang akan menampilkan data santri wilayah yang sudah dimputkan oleh admin pusat. Selain itu juga, dapat melakukan pencarian santri wilayah sesuai kategori (nama dan tahun pelajaran) dan mencetak laporan data santri wilayah.



Gambar 4.34. Desain Menu Santri Admin Wilayah

21. Form export data santri admin wilayah

Desain form menu *export* data santri admin wilayah adalah halaman admin wilayah untuk melakukan *export* data santri wilayah ke bentuk *excel*.

The screenshot shows a web browser window titled 'A Web Page'. On the left is a sidebar menu with the following items: 'Admin Wilayah', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan'. The main content area is titled 'Halaman Export' and contains a dropdown menu for 'Tahun pelajaran' with a 'v' icon, an unchecked checkbox for 'Export ke excel', and a 'Filter / Export' button.

Gambar 4.35. Desain Menu *Export* Data Santri Admin Wilayah

22. Form menu pesan admin wilayah

Desain form menu pesan admin wilayah adalah halaman admin wilayah untuk mengirimkan pesan wilayah ke admin pusat jika terdapat kesalahan penulisan pada data santri wilayah. Selain itu juga, dapat melihat status pesan beserta konfirmasi balasan dari admin pusat.

The screenshot shows a web browser window titled 'A Web Page'. On the left is a sidebar menu with the following items: 'Admin Wilayah', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan'. The main content area is titled 'Halaman Pesan' and contains a table with the following structure:

Tambah atau kirim pesan			
No	Pesan	Status	Balasan

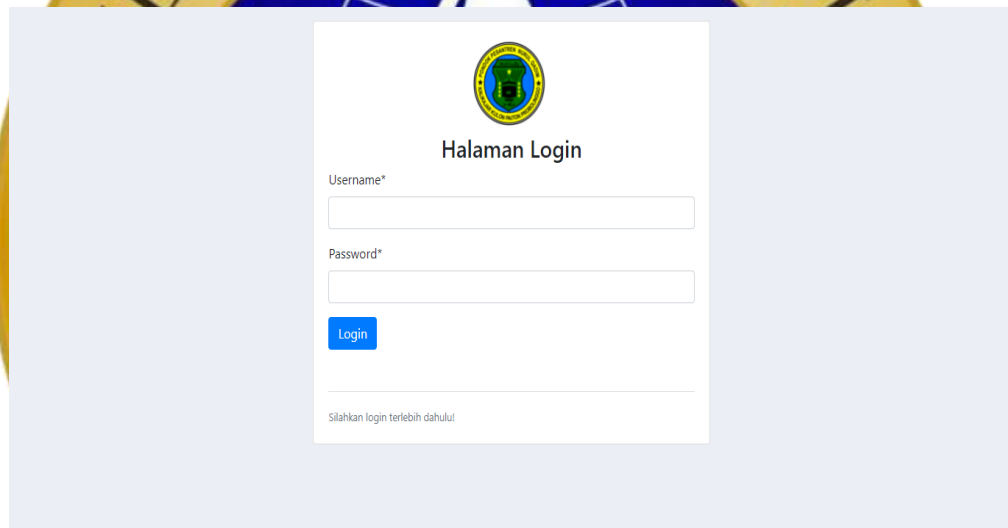
Gambar 4.36. Desain Menu Pesan Admin Wilayah

4.3 Implementasi

Dalam tahap ini desain sistem yang sudah disepakati selanjutnya diterjemahkan ke dalam bahasa pemrograman yang sesuai. Sistem yang dibangun mengacu pada hasil analisis dan perancangan yang telah dilakukan. Berikut gambaran awal sistem buku induk santri terpusat Pondok Pesantren Nurul Qadim:

1. Form login

Form login adalah tampilan awal saat sistem dijalankan. Admin pusat melakukan login dengan menginputkan *username* serta *password* sesuai *database*. Begitu pula dengan admin di wilayah, lembaga dan wilayah melakukan login dengan menginputkan *username* serta *password* sesuai *database*. Form login dapat dilihat pada gambar 4.37. Untuk mengaksesnya dengan alamat url datainduk.nurulqadim.com



Gambar 4.37. Form login

Adapun segmen program pada menu login dapat dilihat pada segmen program berikut.

Segmen Program 4.1. Login

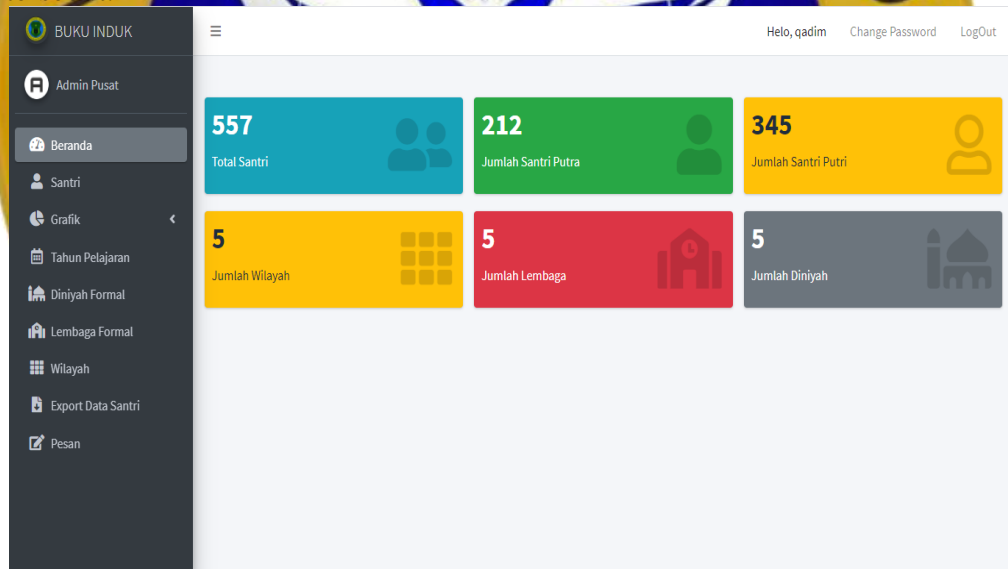
```
def loginPage (request):  
    if request.user.is_authenticated:  
        return render(request, 'data/beranda.html')  
    if request.method == 'POST':  
        username = request.POST.get('username')  
        password = request.POST.get('password')  
        cocokan = authenticate(request,username=username,password=password)  
        if cocokan is not None:  
            login(request, cocokan)
```

Segmen Program 4.2. Login (Lanjutan)

```
        return redirect('beranda')
    else:
        msg = 'Failed Login! Pastikan username dan pasword benar'
        form = AuthenticationForm(request.POST)
        return render(request, 'data/login.html', {'form': form, 'msg': msg})
    else:
        form = AuthenticationForm()
        return render(request, 'data/login.html', {'form': form})
    context = {
        'judul': 'Halaman Login',
        'menu': 'login',
        'tampillogin': formlogin
    }
    return render(request, 'data/login.html', context)
```

2. Form menu utama admin pusat

Menu utama yang akan tampil saat admin pusat login terlihat pada gambar 4.38. Terdapat menu santri, grafik, tahun pelajaran, *export* data santri, diniyah, lembaga, wilayah dan pesan. Admin pusat dapat mengelola data pada menu tersebut.



Gambar 4.38. Form Menu Utama Admin Pusat

Adapun segmen program pada menu utama admin pusat dapat dilihat pada segmen program berikut.

Segmen Program 4.3. Menu Utama Admin Pusat

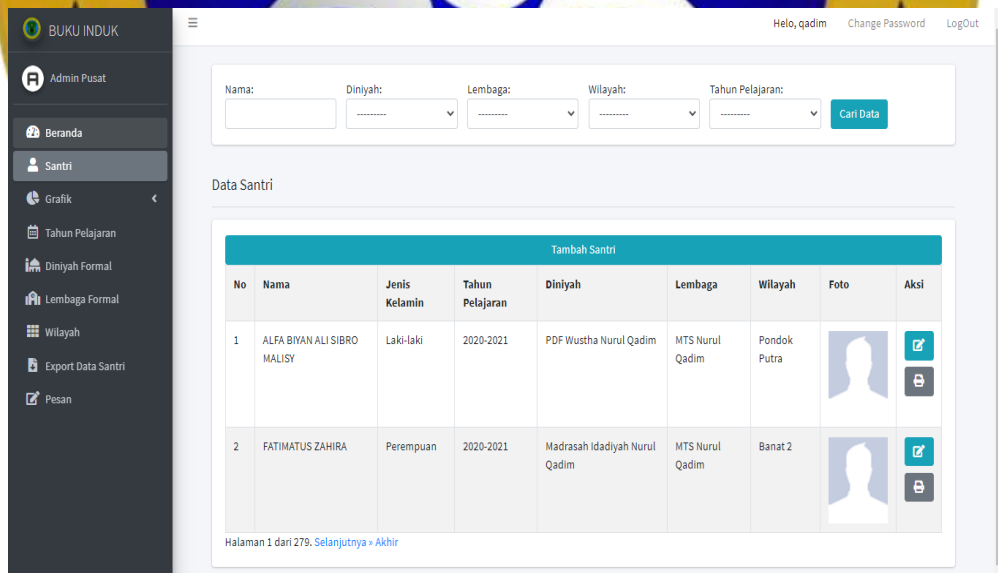
```
@login_required(login_url='login')
@pilihan_login
def beranda(request):
    list_santri = Santri.objects.all()
    list_wilayah = Wilayah.objects.all()
```

Segmen Program 4.4. Menu Utama Admin Pusat (Lanjutan)

```
list_lembaga = Lembaga.objects.all()
list_diniyah = Diniyah.objects.all()
total_santri = list_santri.count()
total_santri_putra = list_santri.filter(jenis_kelamin_id = '1').count()
total_santri_putri = list_santri.filter(jenis_kelamin_id = '2').count()
total_wilayah = list_wilayah.count()
total_lembaga = list_lembaga.count()
total_diniyah = list_diniyah.count()
context={
    'judul': 'Halaman Beranda',
    'menu': 'beranda',
    'total_data_santri': total_santri,
    'total_data_santri_putra': total_santri_putra,
    'total_data_santri_putri': total_santri_putri,
    'total_data_wilayah': total_wilayah,
    'total_data_lembaga': total_lembaga,
    'total_data_diniyah': total_diniyah,
}
return render(request, 'data/beranda.html', context)
```

3. Form menu santri admin pusat

Menu santri admin pusat menampilkan data santri Pondok Pesantren Nurul Qadim, terlihat pada gambar 4.39.



The screenshot shows the 'Admin Pusat' interface. On the left is a sidebar menu with options: Admin Pusat, Beranda, Santri, Grafik, Tahun Pelajaran, Diniyah Formal, Lembaga Formal, Wilayah, Export Data Santri, and Pesan. The main content area has a search form with fields for Nama, Diniyah, Lembaga, Wilayah, and Tahun Pelajaran, and a 'Cari Data' button. Below the search form is a table titled 'Data Santri' with a 'Tambah Santri' button. The table has columns: No, Nama, Jenis Kelamin, Tahun Pelajaran, Diniyah, Lembaga, Wilayah, Foto, and Aksi. Two rows of data are visible.

No	Nama	Jenis Kelamin	Tahun Pelajaran	Diniyah	Lembaga	Wilayah	Foto	Aksi
1	ALFA BIYAN ALI SIBRO MALISY	Laki-laki	2020-2021	PDF Wustha Nurul Qadim	MTS Nurul Qadim	Pondok Putra		
2	FATIMATUS ZAHIRA	Perempuan	2020-2021	Madrasah Ibtidaiyah Nurul Qadim	MTS Nurul Qadim	Banat 2		

Gambar 4.39. Form Menu Santri Admin Pusat

Adapun segmen program pada menu santri admin pusat dapat dilihat pada segmen program 4.5.

Segmen Program 4.5. Menu Santri Admin Pusat

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
def santri(request):
    list_santri = Santri.objects.all()
    filter_santri = SantriFilter(request.GET, queryset=list_santri)
    list_santri_cari=filter_santri.qs
    # pagination
    paginated_filter_santri = Paginator(filter_santri.qs, 2)
    page_number = request.GET.get('page')
    santri_page_obj = paginated_filter_santri.get_page(page_number)
    context={
        'judul': 'Halaman Santri',
        'menu': 'santri',
        'list': list_santri,
        'filter_data_santri':filter_santri,
        'santri_page_obj':santri_page_obj,
    }
    return render(request, 'appsantri/santri.html', context)
```

4. Form menu grafik

Menu grafik menampilkan grafik jumlah data santri berdasarkan diniyah, lembaga dan wilayah. Menu grafik hanya dapat diakses oleh admin pusat, terlihat pada gambar berikut.



Gambar 4.40. Form Menu Grafik Diniyah

Adapun segmen program pada menu grafik diniyah dapat dilihat pada segmen program berikut.

Segmen Program 4.6. Grafik Diniyah

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
```

Segmen Program 4.7. Grafik Diniyah (Lanjutan)

```
def grafik2(request):
    labels = []
    data = []
    appsantri = Santri.objects.values('diniyah__nama').annotate(total=Count('d
    iniyah')).order_by('-total').filter(~Q(diniyah__nama=None))
    for r in appsantri:
        labels.append(r['diniyah__nama'])
        data.append(r['total'])
    context={
        'judul': 'Halaman Grafik'
        'menu': 'grafik2'
        'labels':labels,
        'data' : data
    }
    return render(request, 'appsantri/grafik2.html', context)
```



Gambar 4.41. Form Menu Grafik Lembaga

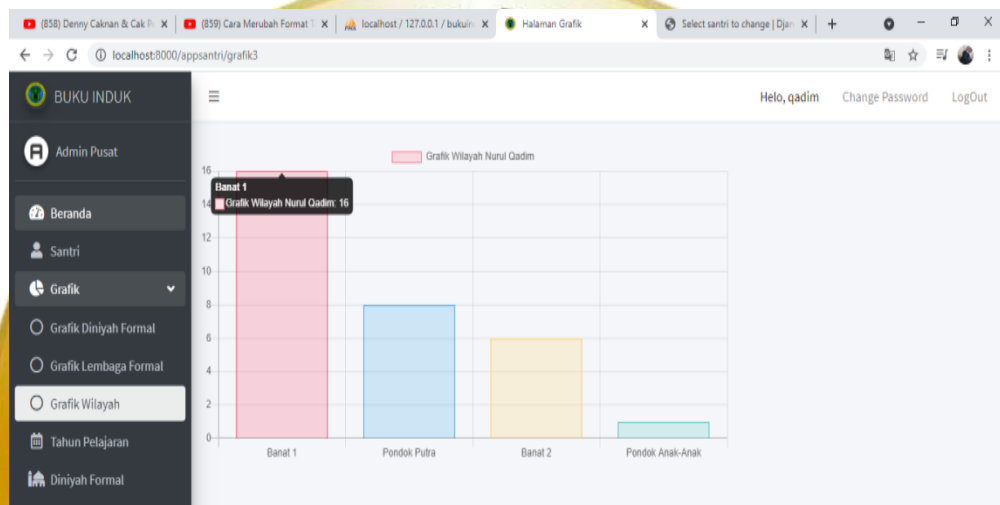
Adapun segmen program pada menu grafik lembaga dapat dapat dilihat pada segmen program berikut.

Segmen Program 4.8. Grafik Lembaga

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
def grafik(request):
    labels = []
    data = []
    appsantri = Santri.objects.values('lembaga__nama').annotate(total=Count('l
    embaga')).order_by('-total')
    for r in appsantri:
```

Segmen Program 4.9. Grafik Lembaga (Lanjutan)

```
labels.append(r['lembaga__nama'])
data.append(r['total'])
context={
    'judul': 'Halaman Grafik',
    'menu': 'grafik',
    'labels':labels,
    'data' : data,
}
return render(request, 'appsantri/grafik.html', context)
```



Gambar 4.42. Form Menu Grafik Wilayah

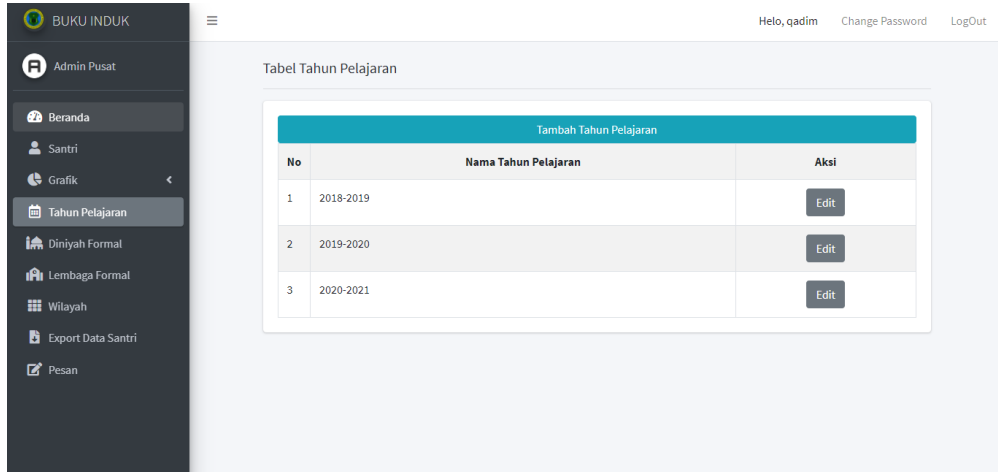
Adapun segmen program pada menu grafik wilayah dapat dilihat pada segmen program berikut.

Segmen Program 4.10. Grafik Wilayah

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
def grafik3(request):
    labels = []
    data = []
    appsantri = Santri.objects.all().values('wilayah__nama').annotate(total=Count('wilayah')).order_by('-total').filter(~Q(wilayah__nama=None))
    for r in appsantri:
        labels.append(r['wilayah__nama'])
        data.append(r['total'])
    context={
        'judul': 'Halaman Grafik',
        'menu': 'grafik3',
        'labels':labels,
        'data' : data, }
    return render(request, 'appsantri/grafik3.html', context)
```

5. Form menu tahun pelajaran

Menu tahun pelajaran menampilkan dan menambah data tahun pelajaran. Menu tahun pelajaran hanya dapat diakses oleh admin pusat terlihat pada gambar 4.43.



Gambar 4.43. Form Menu Tahun Pelajaran

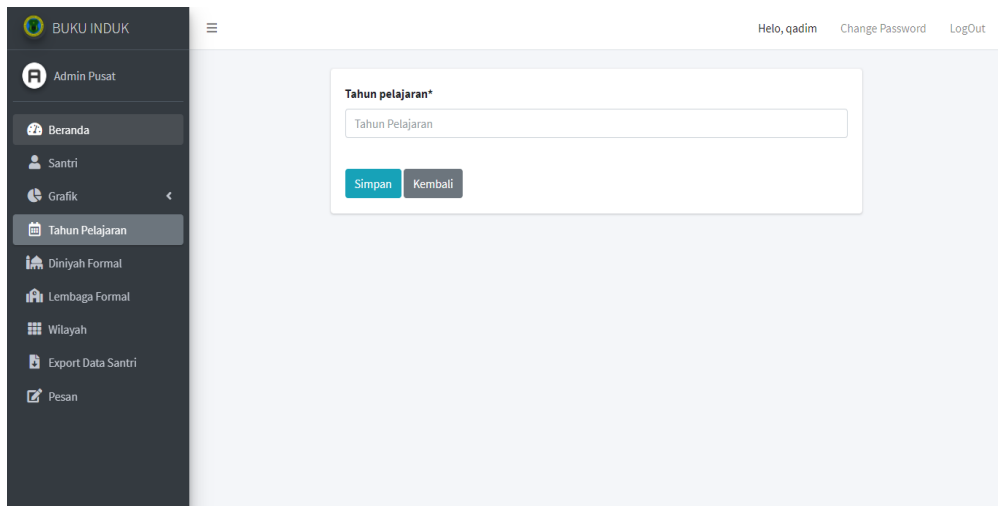
Adapun segmen program pada menu tahun pelajaran dapat dilihat pada segmen program 4.11.

Segmen Program 4.11. Tahun Pelajaran

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_dizinkan=['admin'])
def tahunpelajaran(request):
    list_th = Tahunpelajaran.objects.all()
    context={
        'judul': 'Halaman Tahun Pelajaran',
        'menu': 'tahunpelajaran',
        'list': list_th
    }
    return render(request, 'data/tahunpelajaran.htm', context)
```

6. Form input tahun pelajaran

Form *input* tahun pelajaran adalah halaman untuk menambah data tahun pelajaran dengan mengisi *textfield* yang kosong, yaitu tahun pelajaran. Form *input* tahun pelajaran hanya dapat diakses oleh admin pusat, terlihat pada gambar 4.44.



Gambar 4.44. Form *Input* Tahun Pelajaran

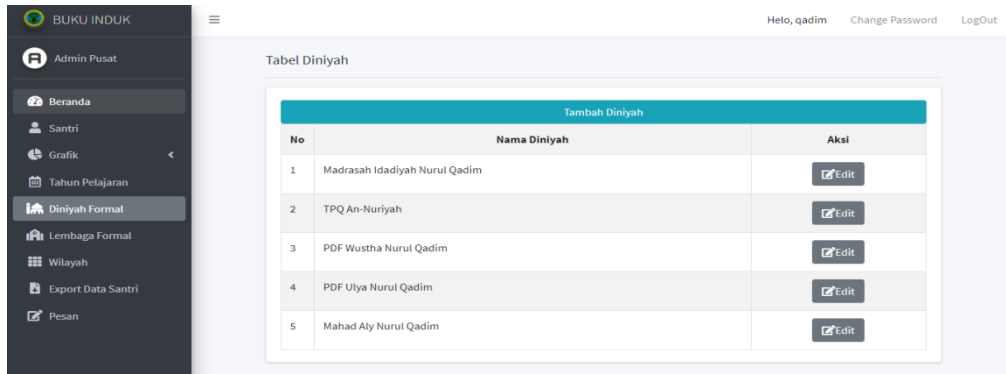
Adapun segmen program pada menu input tahun pelajaran dapat dilihat pada segmen program 4.12.

Segmen Program 4.12. Input Tahun Pelajaran

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
def form_tahunpelajaran(request):
    form_th = FormTahunPelajaran()
    if request.method == 'POST':
        formsimpan = FormTahunPelajaran(request.POST)
        if formsimpan.is_valid():
            formsimpan.save()
            return redirect('tahunpelajaran')
    context={
        'judul': 'Form Tahun Pelajaran',
        'menu': 'tahunpelajaran',
        'form': form_th
    }
    return render(request, 'data/form_th.html', context)
```

7. Form menu diniyah

Menu diniyah menampilkan dan menambah data diniyah Pondok Pesantren Nurul Qadim. Menu diniyah hanya dapat diakses oleh admin pusat, terlihat pada gambar 4.45.



Gambar 4.45. Form Menu Diniyah

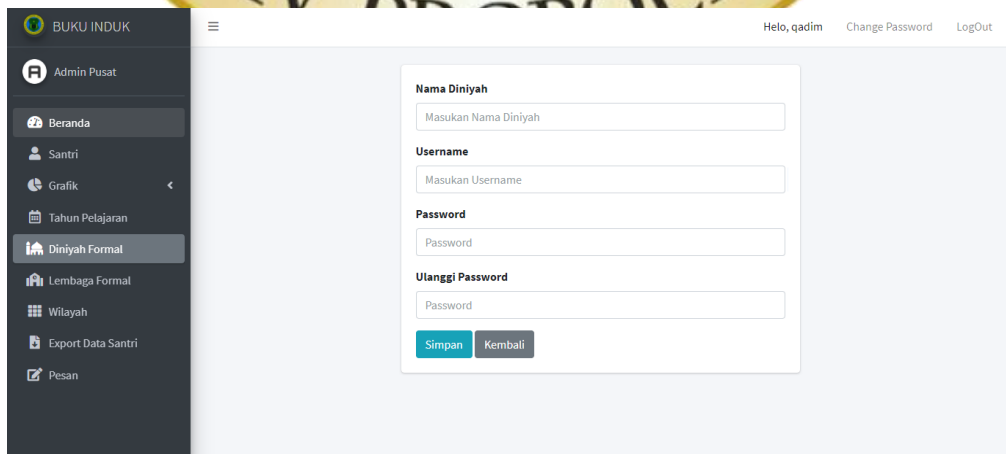
Adapun segmen program pada menu diniyah dapat dilihat pada segmen program 4.13.

Segmen Program 4.13. Menu Diniyah

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_dizinkan=['admin'])
def diniyah(request):
    list_diniyah = Diniyah.objects.all()
    context={
        'judul': 'Halaman Diniyah',
        'menu': 'diniyah',
        'list': list_diniyah,
    }
    return render(request, 'data/diniyah.html', context)
```

8. Form input data diniyah

Form *input* diniyah adalah halaman untuk menambah data beserta *register* admin diniyah dengan mengisi semua *textfield*, yaitu nama diniyah, *username*, *password* dan ulangi *password*. Form *input* data diniyah hanya dapat diakses oleh admin, terlihat pada gambar 4.46.



Gambar 4.46. Form *Input* Data Diniyah

Adapun segmen program pada menu input data diniyah dapat dilihat pada segmen program berikut.

Segmen Program 4.14. Input Data Diniyah

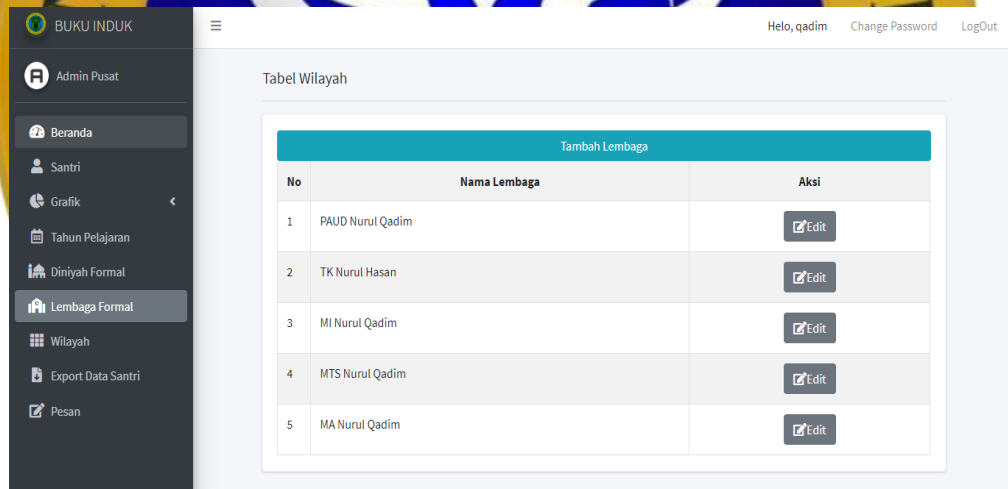
```
class FormDiniyahView(View):
    def get(self, request):
        context={
            'judul': 'Form Diniyah',
            'menu': 'diniyah',
        }
        return render(request, 'data/form_diniyah.html', context)
    def post(self, request):
        context={
            'judul': 'Form Diniyah',
            'menu': 'diniyah',
            'has_error': False,
            'data': request.POST}
        nama = request.POST.get('nama')
        email = request.POST.get('email')
        username = request.POST.get('username')
        password = request.POST.get('pass')
        password2 = request.POST.get('pass2')
        if not nama:
            messages.add_message(request, messages.ERROR, 'Maaf nama masih kosong')
            context['has_error'] = True
        if not validate_email(email):
            messages.add_message(request, messages.ERROR, 'Maaf Email Anda tidak valid')
            context['has_error'] = True
        if len(password) < 6:
            messages.add_message(request, messages.ERROR, 'Password minimal harus 6 karakter')
            context['has_error'] = True
        if password != password2:
            messages.add_message(request, messages.ERROR, 'Password tidak sama')
            context['has_error'] = True
        try:
            if User.objects.get(username=username):
                messages.add_message(
                    request, messages.ERROR, 'Username Sudah ada')
                context['has_error'] = True
        except Exception as identifier:
            pass
        try:
            if User.objects.get(email=email):
                messages.add_message(request, messages.ERROR, 'Email Sudah ada')
                context['has_error'] = True
```

Segmen Program 4.15. Input Data Diniyah (Lanjutan)

```
except Exception as identifier:
    pass
if context['has_error']:
    return render(request, 'data/form_diniyah.html', context, status=400)
user = User.objects.create_user(username=username, email=email)
user.set_password(password)
user.is_active = True
user.save()
diniyah_obj = Diniyah.objects.create(user = user, nama = nama)
diniyah_obj.save()
akses_diniyah = Group.objects.get(name='diniyah')
user.groups.add(akses_diniyah)
messages.success(request, Data diniyah berhasil tersimpan.)
return redirect('diniyah')
```

9. Form menu lembaga

Menu lembaga menampilkan dan menambah data lembaga Pondok Pesantren Nurul Qadim. Menu lembaga hanya dapat diakses oleh admin pusat, terlihat pada gambar 4.47.



Gambar 4.47. Form Menu Lembaga

Adapun segmen program pada menu lembaga dapat dilihat pada segmen program berikut.

Segmen Program 4.16. Menu Lembaga

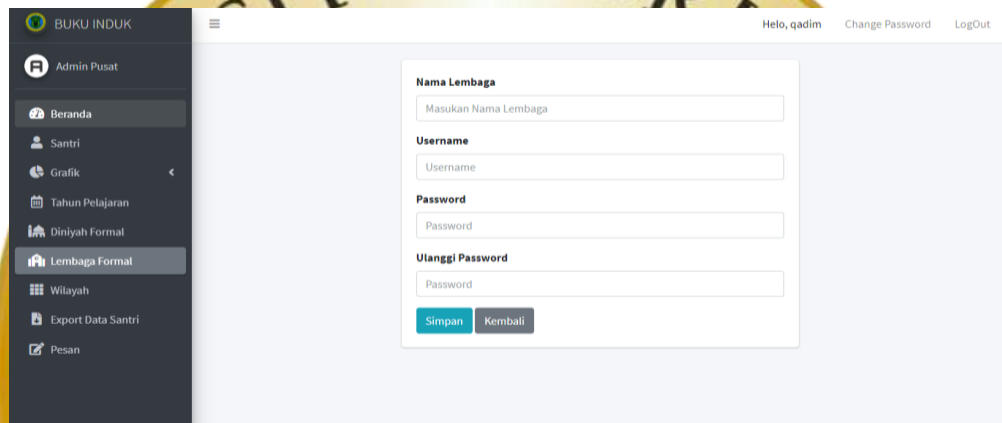
```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
def lembaga(request):
    list_lembaga = Lembaga.objects.all()
    context={
        'judul': 'Halaman Lembaga',
```

Segmen Program 4.17. Menu Lembaga (Lanjutan)

```
'menu': 'lembaga',
'list': list_lembaga,
}
return render(request, 'data/lembaga.html', context)
```

10. Form input data lembaga

Form *input* lembaga adalah halaman untuk menambah data beserta *register* admin lembaga dengan mengisi semua *textfield*, yaitu nama lembaga, *username*, *password* dan ulangi *password*. Form *input* data lembaga hanya dapat diakses oleh admin pusat, terlihat pada gambar 4.48.



Gambar 4.48. Form *Input* Data Lembaga

Adapun segmen program pada menu input data lembaga dapat dilihat pada segmen program berikut.

Segmen Program 4.18. Input Data Lembaga

```
class FormLembagaView(View):
    def get(self, request):
        context={
            'judul': 'Form Lembaga',
            'menu': 'lembaga',
        }
        return render(request, 'data/form_lembaga.html', context)

    def post(self, request):
        context={
            'judul': 'Form Lembaga',
            'menu': 'lembaga',
            'has_error': False,
            'data': request.POST
        }
        nama = request.POST.get('nama')
```

Segmen Program 4.19. Input Data Lembaga (Lanjutan)

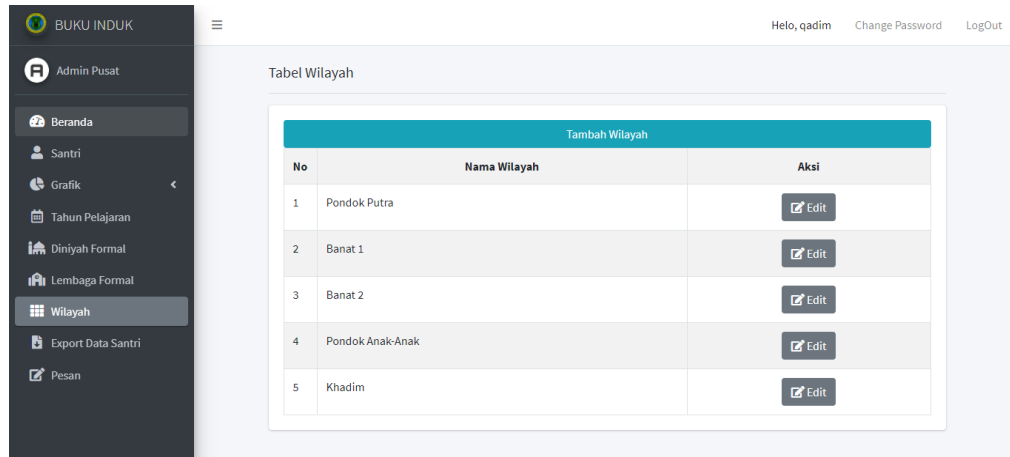
```
email = request.POST.get('email')
username = request.POST.get('username')
password = request.POST.get('pass')
password2 = request.POST.get('pass2')
if not nama:
    messages.add_message(request, messages.ERROR, 'Maaf nama masih
        kosong')
    context['has_error'] = True
if not validate_email(email):
    messages.add_message(request, messages.ERROR, 'Maaf Email Anda t
        idak valid')
    context['has_error'] = True
if len(password) < 6:
    messages.add_message(request, messages.ERROR, 'Password minimal
        harus 6 karekter')
    context['has_error'] = True
if password != password2:
    messages.add_message(request, messages.ERROR, 'Password tidak sama')
    context['has_error'] = True
try:
    if User.objects.get(username=username):
        messages.add_message(
            request, messages.ERROR, 'Username Sudah ada')
        context['has_error'] = True
except Exception as identifier:
    pass
try:
    if User.objects.get(email=email):
        messages.add_message(request, messages.ERROR, 'Email Sudah ada')
        context['has_error'] = True
except Exception as identifier:
    pass
if context['has_error']:
    return render(request, 'data/form_lembaga.html', context, status=400)

user = User.objects.create_user(username=username, email=email)
user.set_password(password)
user.is_active = True
user.save()

lembaga_obj = Lembaga.objects.create(user = user, nama = nama)
lembaga_obj.save()
akses_lembaga = Group.objects.get(name='lembaga')
user.groups.add(akses_lembaga)
messages.success(request, 'Data lembaga berhasil tersimpan.')
return redirect('lembaga')
```

11. Form menu wilayah

Menu wilayah menampilkan dan menambah data wilayah Pondok Pesantren Nurul Qadim. Menu wilayah hanya dapat diakses oleh admin pusat terlihat pada gambar 4.49.



Tambah Wilayah		
No	Nama Wilayah	Aksi
1	Pondok Putra	Edit
2	Banat 1	Edit
3	Banat 2	Edit
4	Pondok Anak-Anak	Edit
5	Khadim	Edit

Gambar 4.49. Form Menu Wilayah

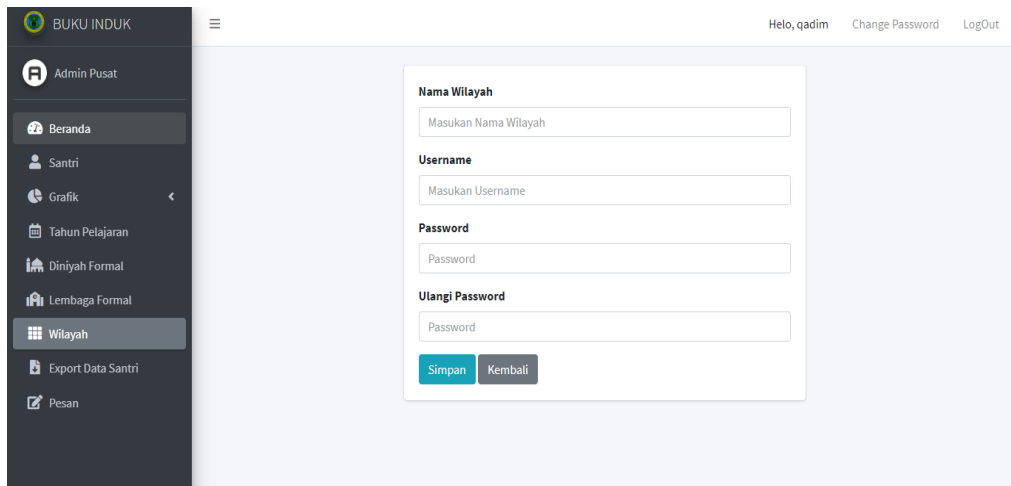
Adapun segmen program pada menu wilayah dapat dilihat pada segmen program 4.20.

Segmen Program 4.20. Menu Wilayah

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
def wilayah(request):
    list_wilayah = Wilayah.objects.all()
    context={
        'judul': 'Halaman Wilayah',
        'menu': 'wilayah',
        'list':list_wilayah,
    }
    return render(request, 'data/wilayah.html', context)
```

12. Form input data wilayah

Form *input* wilayah adalah halaman untuk menambah data beserta *register* admin wilayah dengan mengisi semua *textfield*, yaitu nama wilayah, *username*, *password* dan ulangi *password*. Form *input* data wilayah hanya dapat diakses oleh admin pusat, terlihat pada gambar 4.50.



Gambar 4.50. Form Input Data Wilayah

Adapun segmen program pada menu input data wilayah dapat dilihat pada segmen program berikut.

Segmen Program 4.21. Input Data Wilayah

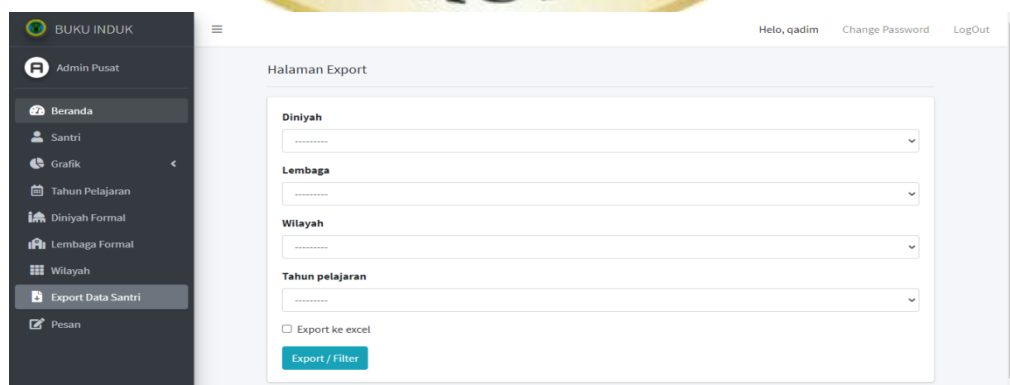
```
class Form WilayahView(View):
    def get(self, request):
        context={
            'judul': 'Form Wilayah',
            'menu': 'wilayah',
        }
        return render(request, 'data/form_wilayah.html', context)
    def post(self, request):
        context={
            'judul': 'Form Wilayah',
            'menu': 'wilayah',
            'has_error': False,
            'data': request.POST
        }
        nama = request.POST.get('nama')
        email = request.POST.get('email')
        username = request.POST.get('username')
        password = request.POST.get('pass')
        password2 = request.POST.get('pass2')
        if not nama:
            messages.add_message(request, messages.ERROR, 'Maaf nama masih kosong')
            context['has_error'] = True
        if not validate_email(email):
            messages.add_message(request, messages.ERROR, 'Maaf Email Anda tidak valid')
            context['has_error'] = True
        if len(password) < 6:
            messages.add_message(request, messages.ERROR, 'Password minimal harus 6 karakter')
```

Segmen Program 4.22. Input Data Wilayah (Lanjutan)

```
context['has_error'] = True
if password!=password2:
    messages.add_message(request, messages.ERROR,'Password tidak sama')
    context['has_error'] = True
try:
    if User.objects.get(username=username):
        messages.add_message(
            request, messages.ERROR, 'Username Sudah ada')
        context['has_error'] = True
except Exception as identifier:
    pass
try:
    if User.objects.get(email=email):
        messages.add_message(request, messages.ERROR, 'Email Sudah ada')
        context['has_error'] = True
except Exception as identifier:
    pass
if context['has_error']:
    return render(request, 'data/form_wilayah.html', context, status=400)
user = User.objects.create_user(username=username, email=email)
user.set_password(password)
user.is_active = True
user.save()
wilayah_obj = Wilayah.objects.create(user = user, nama = nama)
wilayah_obj.save()
akses_wilayah = Group.objects.get(name='wilayah')
user.groups.add(akses_wilayah)
messages.success(request,'Data wilayah berhasil tersimpan.')
return redirect('wilayah')
```

13. Form export data santri admin pusat

Form *export* data santri admin pusat adalah halaman untuk *export* data santri ke bentuk *excel*. Form *export* data santri admin pusat hanya dapat diakses oleh admin pusat, terlihat pada gambar 4.51.



Gambar 4.51. Form *Export* Data Santri Admin Pusat

Adapun segmen program pada menu export data santri admin pusat dapat dilihat pada segmen program berikut.

Segmen Program 4.23. Export Data Santri Admin Pusat

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['admin'])
def export(request):
    header = 'Halaman Export Data'
    queryset = Santri.objects.all()
    form = SantriSearchForm(request.POST or None)
    context = {
        "judul": header,
        "queryset": queryset,
        "menu": "export",
        "form": form,
    }
    if request.method == 'POST':
        lembaga = form['lembaga'].value()
        diniyah = form['diniyah'].value()
        wilayah = form['wilayah'].value()
        tahun_pelajaran = form['tahun_pelajaran'].value()
        if (lembaga != ""):
            queryset = queryset.filter(lembaga_id=lembaga)
        if (diniyah != ""):
            queryset = queryset.filter(diniyah_id=diniyah)
        if (wilayah != ""):
            queryset = queryset.filter(wilayah_id=wilayah)
        if (tahun_pelajaran != ""):
            queryset = queryset.filter(tahun_pelajaran_id=tahun_pelajaran)
        if form['export_ke_Excel'].value() == True:
            response = HttpResponse(content_type='application/ms-excel')
            response['Content-Disposition'] = 'attachment; filename="Data
            santri.xls"'
            wb = xlwt.Workbook(encoding='utf-8')
            ws = wb.add_sheet('Santri')
            row_num = 0
            font_style = xlwt.XFStyle()
            font_style.font.bold = True
            columns = ['NIS', 'NISN', 'NIK', 'NO KK', 'NAMA', 'TEMPAT LAHIR',
            'TANGGAL LAHIR', 'JENIS KELAMIN', 'AGAMA', 'ALAMAT',
            'KECAMATAN', 'KABUPATEN', 'PROPINSI', 'HOBI', 'CITA CITA',
            'ANAK KE', 'JUMLAH SAUDARA', 'TGL MASUK PESANTREN',
            'STATUS ASAL SANTRI', 'SEKOLAH ASAL', 'ALAMAT SEKOLAH ASAL',
            'WILAYAH', 'LEMBAGA', 'DINIYAH', 'TAHUN PELAJARAN',
            'NAMA AYAH', 'STATUS HIDUP AYAH', 'PENDIDIKAN AYAH',
            'PEKERJAAN AYAH', 'NO HP AYAH', 'NAMA IBU', 'STATUS HIDUP IBU',
            'PENDIDIKAN IBU', 'PEKERJAAN IBU', 'NO HP IBU', 'NAMA WALI',
            'PENDIDIKAN WALI', 'PEKERJAAN
```

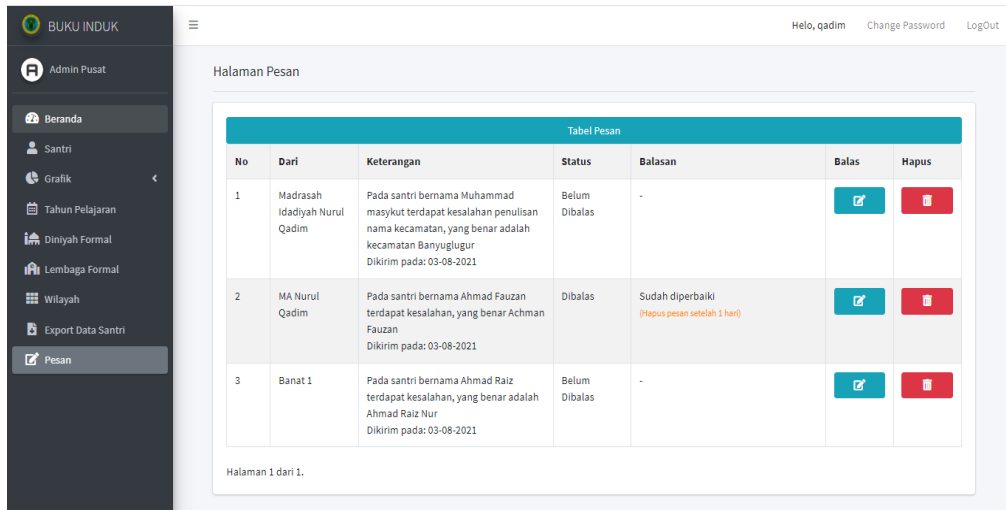
Segmen Program 4.24. Export Data Santri Admin Pusat (Lanjutan)

```
WALI', 'NO HP WALI', 'PENGHASILAN ORANG TUA ATAU WA  
LI PERBULAN' ]
```

```
for col_num in range(len(columns)):  
    ws.write(row_num, col_num, columns[col_num], font_style)  
font_style = xlwt.XFStyle()  
rows = queryset.values_list('nis', 'nisn', 'nik', 'no_kk', 'nama', 'tempat_l  
ahir', 'tanggal_lahir', 'jenis_kelamin', 'agama', 'alamat', 'kecamatan', 'ka  
bupaten', 'propinsi', 'hobi', 'cita_cita', 'anak_ke', 'jumlah_saudara', 'tang  
gal_masuk_pesantren', 'status_asal_santri', 'sekolah_asal', 'alamat_sek  
olah_asal', 'wilayah', 'lembaga', 'diniyah', 'tahun_pelajaran', 'nama_aya  
h', 'status_hidup_ayah', 'pendidikan_ayah', 'pekerjaan_ayah', 'no_hp_a  
yah', 'nama_ibu', 'status_hidup_ibu', 'pendidikan_ibu', 'pekerjaan_ibu', '  
no_hp_ibu', 'nama_wali', 'pendidikan_wali', 'pekerjaan_wali', 'no_hp_  
wali', 'penghasilan_orang_tua_atau_wali_perbulan' )  
for row in rows:  
    row_num += 1  
    for col_num in range(len(row)):  
        ws.write(row_num, col_num, row[col_num], font_style)  
wb.save(response)  
return response  
context = {  
    "form": form,  
    "header": header,  
    "queryset": queryset,  
    "menu": "export",  
}  
return render(request, 'appsantri/export.html', context)
```

14. Form pesan admin pusat

Form pesan admin pusat adalah halaman yang menampilkan pesan dari diniyah, lembaga dan wilayah, serta melakukan pengolahan data pada pesan tersebut. Form pesan admin pusat hanya dapat diakses oleh admin pusat, terlihat pada gambar 4.52.



Gambar 4.52. Form Pesan Admin Pusat

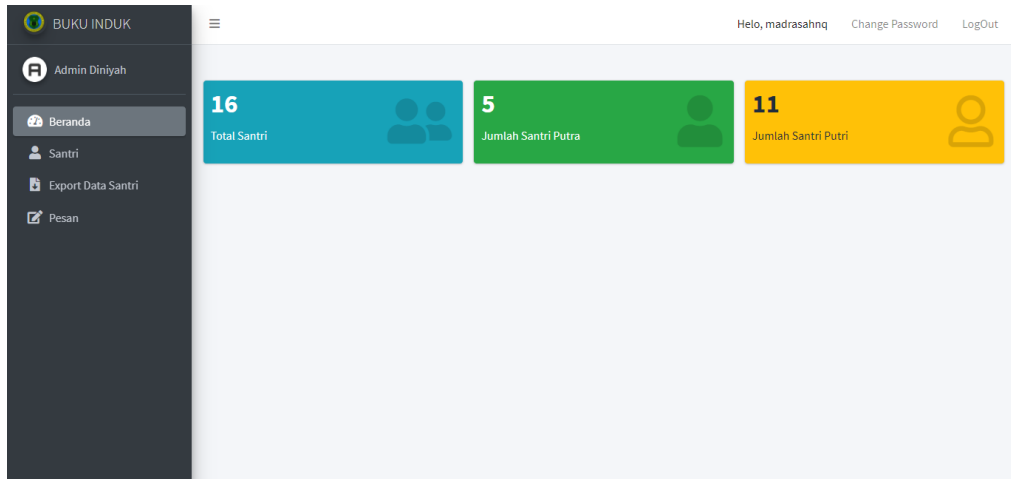
Adapun segmen program pada menu utama admin pusat dapat dilihat pada segmen program 4.25.

Segmen Program 4.25. Halaman Pesan Admin Pusat

```
@login_required(login_url='login')
@tjinkan_pengguna(yang_diizinkan=['admin'])
def pesan_admin(request):
    list_pesan = Pesan.objects.all()
    paginated_pesan = Paginator(list_pesan, 10)
    page_number = request.GET.get('page')
    pesan_page_obj = paginated_pesan.get_page(page_number)
    context={
        'judul': 'Halaman Pesan',
        'menu': 'pesan',
        'list': list_pesan,
        'pesan_page_obj': pesan_page_obj,
    }
    return render(request, 'data/pesan_admin.html', context)
```

15. Form menu utama admin diniyah

Menu utama yang akan tampil saat admin diniyah login terlihat pada gambar 4.53. Terdapat menu santri, *export* data santri dan pesan. Admin diniyah dapat mengirimkan pesan jika terdapat kesalahan pada data santri diniyah tersebut dan akan mendapat konfirmasi balasan jika data telah diperbaiki.



Gambar 4.53. Form Menu Utama Admin Diniyah

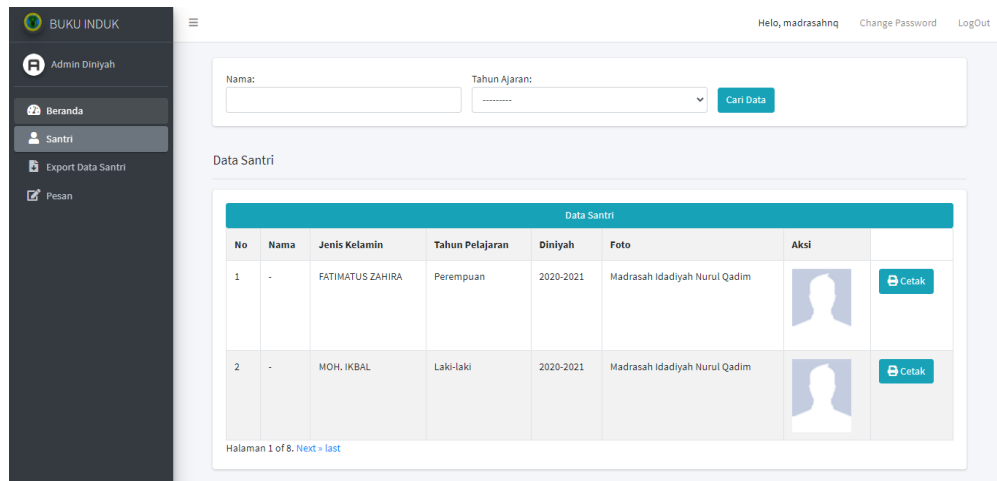
Adapun segmen program pada menu utama admin diniyah dapat dilihat pada segmen program 4.26.

Segmen Program 4.26. Menu Utama Admin Diniyah

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['diniyah'])
def userDiniyah(request):
    list_santridiniyah = Santri.objects.all().filter(diniyah=request.user.diniyah)
    total_santridiniyah = list_santridiniyah.count()
    total_santridiniyah_putra = list_santridiniyah.filter(jenis_kelamin_id = '1').count()
    total_santridiniyah_putri = list_santridiniyah.filter(jenis_kelamin_id = '2').count()
    context={
        'judul': 'Halaman Beranda',
        'menu': 'userdiniyah',
        'total_data_santridiniyah': total_santridiniyah,
        'total_data_santridiniyah_putra': total_santridiniyah_putra,
        'total_data_santridiniyah_putri': total_santridiniyah_putri,
    }
    return render(request, 'data/userdiniyah.html', context)
```

16. Form menu santri admin diniyah

Menu santri admin diniyah menampilkan data santri diniyah Pondok Pesantren Nurul Qadim. Admin diniyah dapat melakukan pencarian data santri diniyah berdasarkan kategori (nama dan tahun pelajaran) serta mencetak laporan laporan data santri diniyah, terlihat pada gambar 4.54.



Gambar 4.54. Form Menu Santri Admin Diniyah

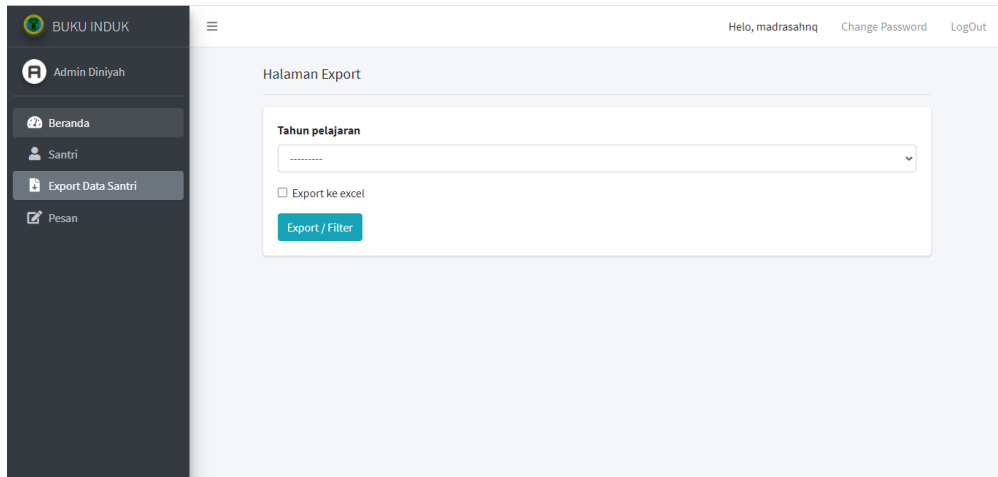
Adapun segmen program pada menu santri admin diniyah dapat dilihat pada segmen program 4.27

Segmen Program 4.27. Menu Santri Admin Diniyah

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['diniyah'])
def santridiniyah(request):
    list_santri = Santri.objects.all().filter(diniyah=request.user.diniyah)
    filter_santri = SantriFilter(request.GET, queryset=list_santri)
    list_santri=filter_santri.qs
    paginated_filter_santri = Paginator(filter_santri.qs, 2)
    page_number = request.GET.get('page')
    santri_page_obj = paginated_filter_santri.get_page(page_number)
    context={
        'judul': 'Halaman Santri',
        'menu': santridiniyah,
        'list': list_santri,
        'filter_data_santri':filter_santri,
        'santri_page_obj':santri_page_obj,
    }
    return render(request, 'appsantri/santridiniyah.html', context)
```

17. Form menu export data santri admin diniyah

Menu *export* data santri admin diniyah adalah halaman untuk *export* data santri diniyah ke bentuk *excel*. Form *export* data santri admin diniyah terlihat pada gambar 4.55.



Gambar 4.55. Form Menu *Export* Data Santri Admin Diniyah

Adapun segmen program pada menu *export* data santri admin diniyah dapat dilihat pada segmen program berikut.

Segmen Program 4.28. *Export* Data Santri Diniyah

```
login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['diniyah'])
def export_santridiniyah(request):
    header = 'Halaman Export Data'
    queryset = Santri.objects.all()
    form = SantrisSearchForm(request.POST or None)
    context = {
        "judul": header,
        "queryset": queryset,
        "menu": "export_santridiniyah",
        "form": form,
    }
    if request.method == 'POST':
        tahun_pelajaran = form['tahun_pelajaran'].value()
        if (tahun_pelajaran != ""):
            queryset = Santri.objects.filter(tahun_pelajaran_id=tahun_pelajaran)
            if form['export_ke_Excel'].value() == True:
                response = HttpResponse(content_type='application/ms-excel')
                response['Content-Disposition'] = 'attachment; filename="santri.xls"'
                wb = xlwt.Workbook(encoding='utf-8')
                ws = wb.add_sheet('Santri')
                row_num = 0
                font_style = xlwt.XFStyle()
                font_style.font.bold = True
                columns = ['NIS', 'NISN', 'NIK', 'NO KK', 'NAMA', 'TEMPAT LAHIR', 'TANGGAL LAHIR', 'JENIS KELAMIN', 'AGAMA', 'ALAMAT', 'KECAMATAN', 'KABUPATEN', 'PROPINSI', 'HOBI', 'CITA CITA', 'ANAK KE', 'JUMLAH SAUDARA', 'TGL MASUK PESANTREN', 'STATUS ASAL SANTRI', 'SEKOLAH ASAL', 'ALAMAT SEKOLA
```

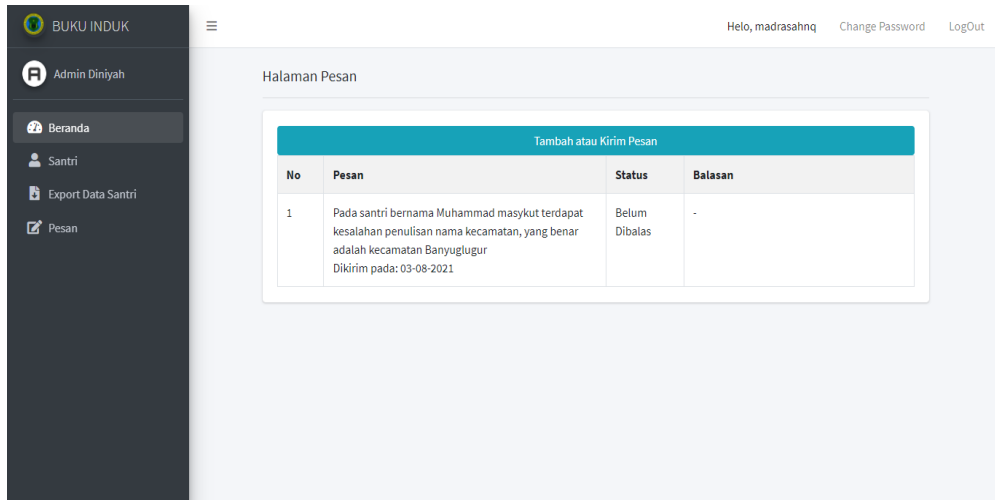
Segmen Program 4.29. Export Data Santri Diniyah (Lanjutan)

H ASAL', 'WILAYAH', 'LEMBAGA', 'DINIYAH', 'TAHUN PELAJARAN', 'NAMA AYAH', 'STATUS HIDUP AYAH', 'PENDIDIKAN AYAH', 'PEKERJAAN AYAH', 'NO HP AYAH', 'NAMA IBU', 'STATUS HIDUP IBU', 'PENDIDIKAN IBU', 'PEKERJAAN IBU', 'NO HP IBU', 'NAMA WALI', 'PENDIDIKAN WALI', 'PEKERJAAN WALI', 'NO HP WALI', 'PENGHASILAN ORANG TUA ATAU WALI PERBULAN']

```
for col_num in range(len(columns)):
    ws.write(row_num, col_num, columns[col_num], font_style)
font_style = xlwt.XFStyle()
rows = queryset.filter(diniyah=request.user.diniyah).values_list('nis', 'n
isn', 'nik', 'no_kk', 'nama', 'tempat_lahir', 'tanggal_lahir', 'jenis_kelamin'
, 'agama', 'alamat', 'kecamatan', 'kabupaten', 'propinsi', 'hobi', 'cita_cita',
'anak_ke', 'jumlah_saudara', 'tanggal_masuk_pesantren', 'status_asal_sa
ntri', 'sekolah_asal', 'alamat_sekolah_asal', 'wilayah', 'lembaga', 'diniya
h', 'tahun_pelajaran', 'nama_ayah', 'status_hidup_ayah', 'pendidikan_ayah',
'pekerjaan_ayah', 'no_hp_ayah', 'nama_ibu', 'status_hidup_ibu', 'pe
ndidikan_ibu', 'pekerjaan_ibu', 'no_hp_ibu', 'nama_wali', 'pendidikan_
wali', 'pekerjaan_wali', 'no_hp_wali', 'penghasilan_orang_tua_atau_wa
li_perbulan')
for row in rows:
    row_num += 1
    for col_num in range(len(row)):
        ws.write(row_num, col_num, row[col_num], font_style)
wb.save(response)
return response
context = {
    "form": form,
    "header": header,
    "queryset": queryset,
    "menu": "export_santridiniyah",
}
return render(request, 'appsantri/export_santridiniyah.html', context)
```

18. Form menu pesan admin diniyah

Menu pesan admin diniyah adalah halaman untuk mengimkan pesan diniyah ke pusat jika terdapat kesalahan pada data santri diniyah dan akan mendapat konfirmasi balasan jika data telah diperbaiki, terlihat pada gambar 4.56.



Gambar 4.56. Form Menu Pesan Admin Diniyah

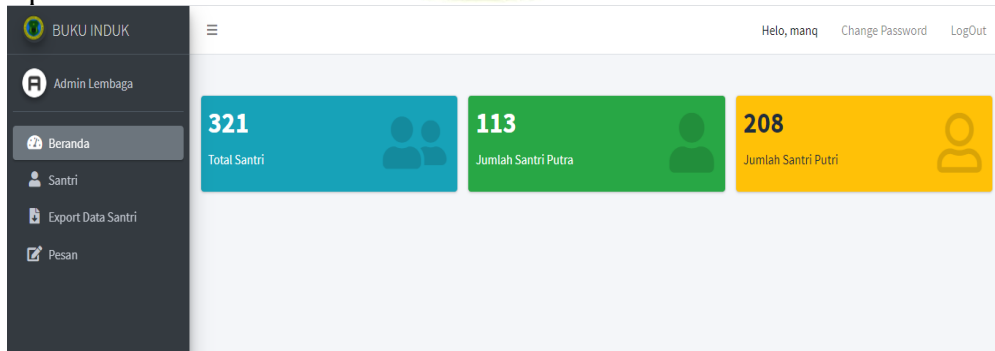
Adapun segmen program pada menu pesan admin diniyah dapat dilihat pada segmen program 4.30.

Segmen Program 4.30. Menu Pesan Admin Diniyah

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['diniyah'])
def pesandiniyah(request):
    list_pesan = Pesan.objects.all().filter(diniyah=request.user.diniyah)
    context={
        'judul': 'Halaman Pesan',
        'menu': 'pesanadmindiniyah',
        'list': list_pesan,
    }
    return render(request, 'data/pesandiniyah.html', context)
```

19. Form menu utama admin lembaga

Menu utama yang akan tampil saat admin lembaga login terlihat pada gambar 4.57. Terdapat menu santri, *export* data santri dan pesan. Admin lembaga dapat mengirimkan pesan jika terdapat kesalahan pada data santri lembaga tersebut dan akan mendapat konfirmasi balasan jika data telah diperbaiki.



Gambar 4.57. Form Menu Utama Admin Lembaga

Adapun segmen program pada menu utama admin lembaga dapat dilihat pada segmen program 4.31.

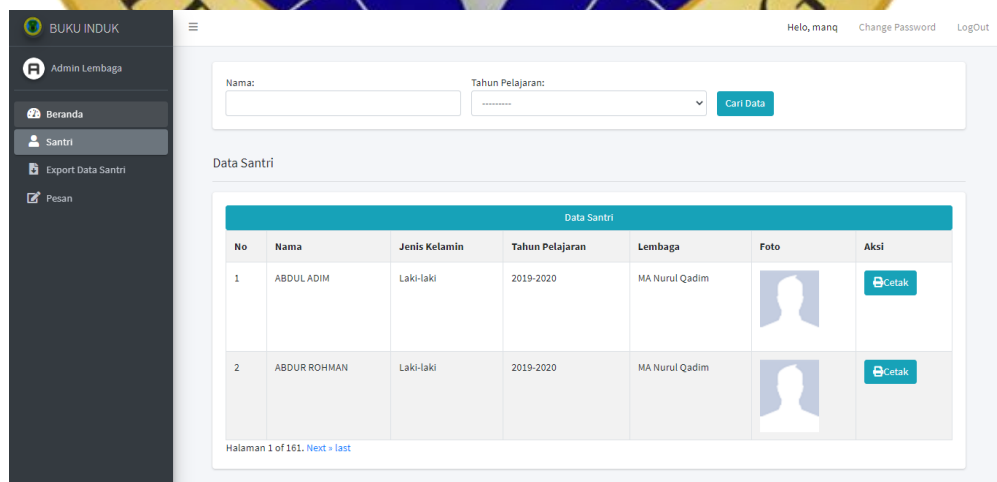
Segmen Program 4.31. Menu Utama Admin Lembaga

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['lembaga'])
def userLembaga(request):
    list_santrilembaga = Santri.objects.all().filter(lembaga=request.user.lembaga)
    total_santrilembaga = list_santrilembaga.count()
    total_santrilembaga_putra = list_santrilembaga.filter(jenis_kelamin_id = '1').count()
    total_santrilembaga_putri = list_santrilembaga.filter(jenis_kelamin_id = '2').count()



    context={
        'judul': 'Halaman Beranda',
        'menu': 'userlembaga',
        'total_data_santrilembaga': total_santrilembaga,
        'total_data_santrilembaga_putra': total_santrilembaga_putra,
        'total_data_santrilembaga_putri': total_santrilembaga_putri,
    }
    return render(request, 'data/userlembaga.html', context)
```

20. Form menu santri admin lembaga

Menu santri admin lembaga menampilkan data santri lembaga Pondok Pesantren Nurul Qadim. Admin lembaga dapat melakukan pencarian data santri lembaga berdasarkan kategori (nama dan tahun pelajaran) dan mencetak laporan data santri lembaga terlihat pada gambar 4.58



The screenshot shows a web application interface for an admin. On the left is a dark sidebar with a menu containing 'Admin Lembaga', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan'. The main content area has a header with 'BUKU INDUK' and user options 'Helo, maq', 'Change Password', and 'LogOut'. Below the header is a search form with two input fields: 'Nama:' and 'Tahun Pelajaran:', followed by a 'Cari Data' button. Underneath the search form is a table titled 'Data Santri' with the following data:

No	Nama	Jenis Kelamin	Tahun Pelajaran	Lembaga	Foto	Aksi
1	ABDUL ADIM	Laki-laki	2019-2020	MA Nurul Qadim		Cetak
2	ABDUR ROHMAN	Laki-laki	2019-2020	MA Nurul Qadim		Cetak

At the bottom of the table, it says 'Halaman 1 of 161. Next > last'.

Gambar 4.58. Form Menu Santri Admin Lembaga

Adapun segmen program pada menu santri admin lembaga dapat dilihat pada segmen program 4.32.

Segmen Program 4.32. Data Santri Admin Lembaga

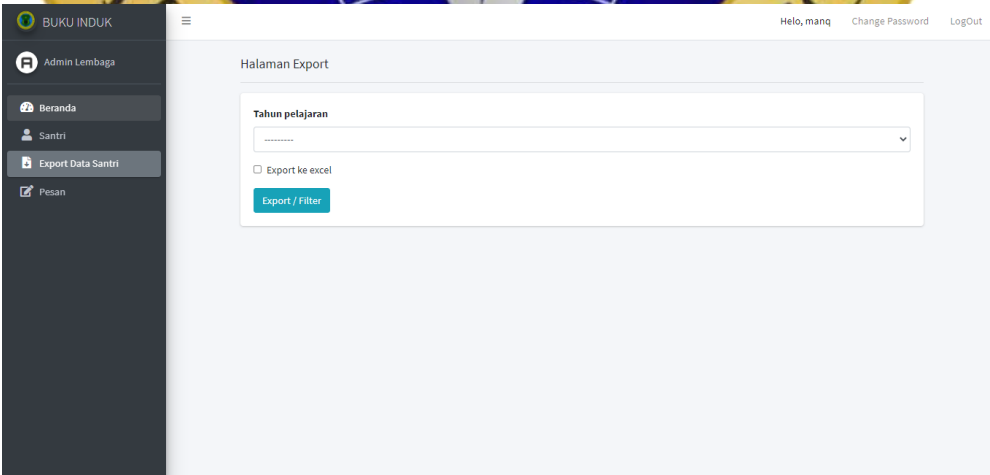
```
login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['lembaga'])
def santrilembaga(request):
    list_santri = Santri.objects.all().filter(lembaga=request.user.lembaga)
    filter_santri = SantriFilter(request.GET, queryset=list_santri)
    list_santri=filter_santri.qs

    paginated_filter_santri = Paginator(filter_santri.qs, 2)
    page_number = request.GET.get('page')
    santri_page_obj = paginated_filter_santri.get_page(page_number)

    context={
        'judul': 'Halaman Santri',
        'menu': 'santrilembaga',
        'list': list_santri,
        'filter_data_santri':filter_santri,
        'santri_page_obj':santri_page_obj,
    }
    return render(request, 'appsantri/santrilembaga.html', context)
```

21. Form menu export data santri admin lembaga

Menu *export* data santri admin lembaga adalah halaman untuk *export* data santri lembaga ke bentuk *excel*. Form *export* data santri admin lembaga terlihat pada gambar 4.59.



The screenshot shows a web application interface. On the left is a dark sidebar with a menu containing 'Admin Lembaga', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan'. The main content area is titled 'Halaman Export' and features a dropdown menu labeled 'Tahun pelajaran' with a downward arrow. Below the dropdown is a checkbox labeled 'Export ke excel'. At the bottom of the form is a blue button labeled 'Export / Filter'. The top right of the page shows the user's name 'Helo, manq', a 'Change Password' link, and a 'LogOut' link.

Gambar 4.59. Form Menu *Export* Data Santri Admin Lembaga

Adapun segmen program pada menu *export* data santri admin lembaga dapat dilihat pada segmen program berikut.

Segmen Program 4.33. *Export* Data Santri Admin Lembaga

```
login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['lembaga'])
def export_santrilembaga(request):
    header = 'Halaman Export Data'
    queryset = Santri.objects.all()
    form = SantrisSearchForm(request.POST or None)
    context = {
        "judul": header,
        "queryset": queryset,
        "menu": "export_santrilembaga",
        "form": form,
    }
    if request.method == 'POST':
        tahun_pelajaran = form['tahun_pelajaran'].value()
        if (tahun_pelajaran != ""):
            queryset = Santri.objects.filter(tahun_pelajaran_id=tahun_pelajaran)
            if form['export_ke_Excel'].value() == True:
                response = HttpResponse(content_type='application/ms-excel')
                response['Content-Disposition'] = 'attachment; filename="santri.xls"'
                wb = xlwt.Workbook(encoding='utf-8')
                ws = wb.add_sheet('Santri')
                row_num = 0
                font_style = xlwt.XFStyle()
                font_style.font.bold = True
                columns = ['NIS', 'NISN', 'NIK', 'NO KK', 'NAMA', 'TEMPAT LAHIR',
                    'TANGGAL LAHIR', 'JENIS KELAMIN', 'AGAMA', 'ALAMAT', 'KECAMATAN', 'KABUPATEN', 'PROPINSI', 'HOBI', 'CITA CITA', 'ANAK KE', 'JUMLAH SAUDARA', 'TGL MASUK PESANTREN', 'STATUS ASAL SANTRI', 'SEKOLAH ASAL', 'ALAMAT SEKOLAH ASAL', 'WILAYAH', 'LEMBAGA', 'DENIYAH', 'TAHUN PELAJARAN', 'NAMA AYAH', 'STATUS HIDUP AYAH', 'PENDIDIKAN AYAH', 'PEKERJAAN AYAH', 'NO HP AYAH', 'NAMA IBU', 'STAU S HIDUP IBU', 'PENDIDIKAN IBU', 'PEKERJAAN IBU', 'NO HP IBU', 'NAMA WALI', 'PENDIDIKAN WALI', 'PEKERJAAN WALI', 'NO HP WALI', 'PENGHASILAN ORANG TUA ATAU WALI PERBULAN']
                for col_num in range(len(columns)):
                    ws.write(row_num, col_num, columns[col_num], font_style)
                    font_style = xlwt.XFStyle()
                rows = queryset.filter(lembaga=request.user.lembaga).values_list('nis', 'nisn', 'nik', 'no_kk', 'nama', 'tempat_lahir', 'tanggal_lahir', 'jenis_kelamin', 'agama', 'alamat', 'kecamatan', 'kabupaten', 'propinsi', 'hobi', 'cita_cita', 'anak_ke', 'jumlah_saudara', 'tanggal_masuk_pesantren', 'status_asal
```

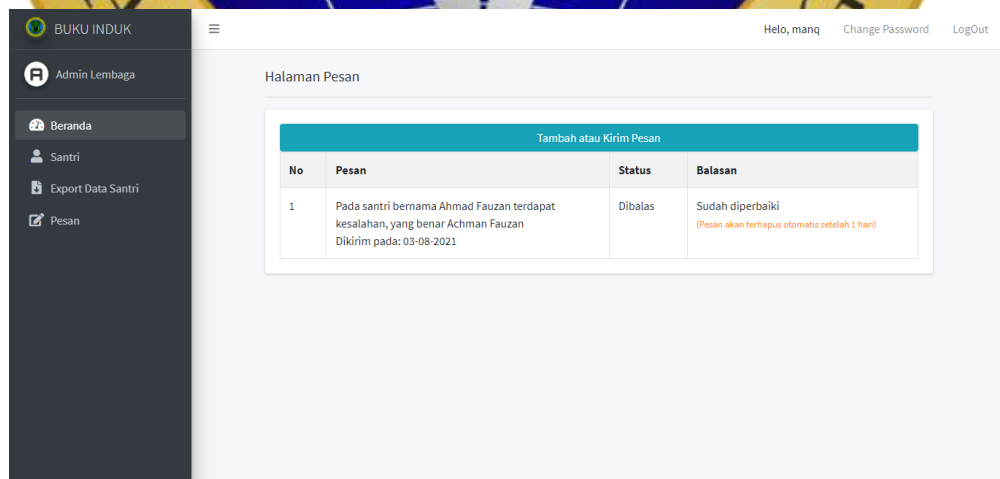
Segmen Program 4.34. Export Data Santri Lembaga (Lanjutan)

```
_santri', 'sekolah_asal', 'alamat_sekolah_asal', 'wilayah', 'lembaga', 'din  
iyah', 'tahun_pelajaran', 'nama_ayah', 'status_hidup_ayah', 'pendidikan  
_ayah', 'pekerjaan_ayah', 'no_hp_ayah', 'nama_ibu', 'status_hidup_ibu',  
'pendidikan_ibu', 'pekerjaan_ibu', 'no_hp_ibu', 'nama_wali', 'pendidika  
n_wali', 'pekerjaan_wali', 'no_hp_wali', 'penghasilan_orang_tua_atau_  
wali_perbulan' )
```

```
for row in rows:  
    row_num += 1  
    for col_num in range(len(row)):  
        ws.write(row_num, col_num, row[col_num], font_style)  
wb.save(response)  
return response  
context = {  
    "form": form,  
    "header": header,  
    "queryset": queryset,  
    "menu": "export_santrilembaga",  
}  
return render(request, 'appsantri/export_santrilembaga.html', context)
```

22. Form menu pesan admin lembaga

Menu pesan admin lembaga adalah halaman untuk mengirim pesan lembaga ke pusat jika terdapat kesalahan pada data santri lembaga dan akan mendapat konfirmasi balasan jika data telah diperbaiki, terlihat pada gambar 4.60.



Gambar 4.60. Form Menu Pesan Admin Lembaga

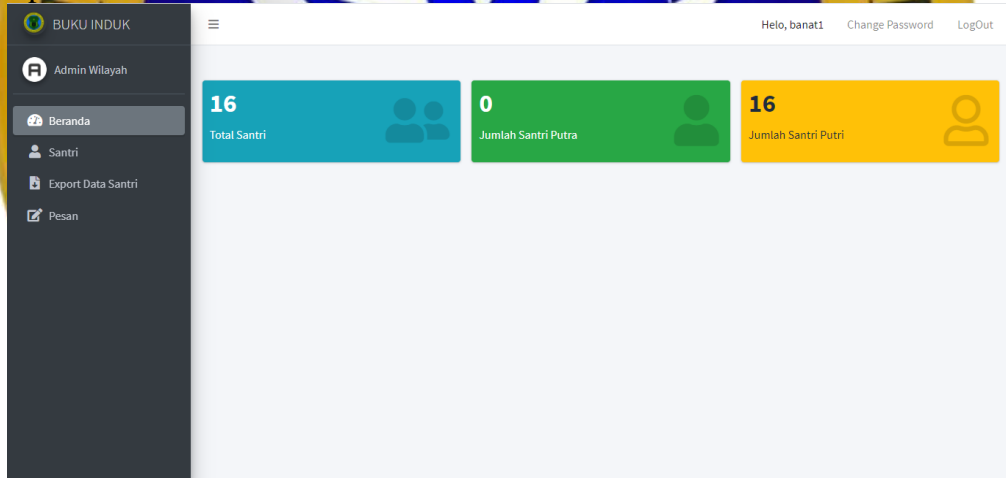
Adapun segmen program pada menu pesan admin lembaga dapat dilihat pada segmen program berikut.

Segmen Program 4.35. Menu Pesan Admin Lembaga

```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['lembaga'])
def pesanlembaga(request):
    list_pesan = Pesan.objects.all().filter(lembaga=request.user.lembaga)
    context={
        'judul': 'Halaman Pesan',
        'menu': 'pesanadminlembaga',
        'list': list_pesan,
    }
    return render(request, 'data/pesanlembaga.html', context)
```

23. Form menu utama admin wilayah

Menu utama yang akan tampil saat admin wilayah login terlihat pada gambar 4.61. Terdapat menu santri, *export* data dan pesan. Admin wilayah dapat mengirimkan pesan wilayah jika terdapat kesalahan pada data santri wilayah tersebut dan akan mendapat konfirmasi balasan jika data telah diperbaiki.



Gambar 4.61. Form Menu Utama Admin Wilayah

Adapun segmen program pada menu utama admin wilayah dapat dilihat pada segmen program berikut.

Segmen Program 4.36. Menu Utama Admin Wilayah

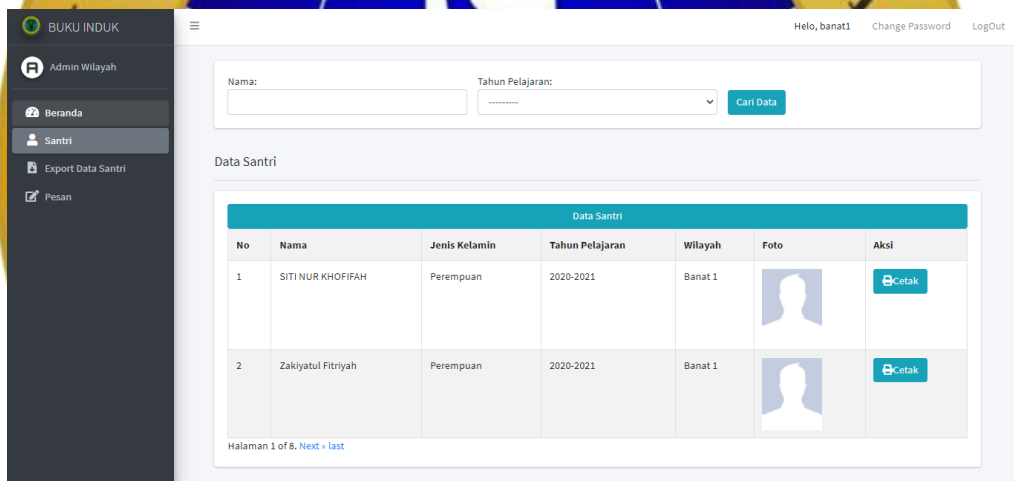
```
@login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['wilayah'])
def userWilayah(request):
    list_santriwilayah = Santri.objects.all().filter(wilayah=request.user.wilayah)
    total_santriwilayah = list_santriwilayah.count()
    total_santriwilayah_putra = list_santriwilayah.filter(jenis_kelamin_id = '1').
    count()
```

Segmen Program 4.37. Menu Utama Admin Wilayah (Lanjutan)

```
total_santriwilayah_putri = list_santriwilayah.filter(jenis_kelamin_id = '2').  
count()  
context={  
    'judul': 'Halaman Beranda',  
    'menu': 'userwilayah',  
    'total_data_santriwilayah': total_santriwilayah,  
    'total_data_santriwilayah_putra': total_santriwilayah_putra,  
    'total_data_santriwilayah_putri': total_santriwilayah_putri,  
}  
return render(request, 'data/userwilayah.html', context)
```

24. Form menu santri admin wilayah

Menu santri admin wilayah menampilkan data santri wilayah Pondok Pesantren Nurul Qadim. Admin wilayah dapat melakukan pencarian data santri wilayah berdasarkan kategori (nama dan tahun pelajaran) dan mencetak laporan data santri wilayah, terlihat pada gambar 4.62.



The screenshot shows a web application interface for 'Admin Wilayah'. On the left is a dark sidebar with navigation items: 'Admin Wilayah', 'Beranda', 'Santri', 'Export Data Santri', and 'Pesan'. The main content area has a search form with 'Nama:' and 'Tahun Pelajaran:' fields, a 'Cari Data' button, and a 'Data Santri' table. The table has columns for No, Nama, Jenis Kelamin, Tahun Pelajaran, Wilayah, Foto, and Aksi. Two rows of data are visible, each with a 'Cetak' button. The footer of the table indicates 'Halaman 1 of 8. Next > last'.

No	Nama	Jenis Kelamin	Tahun Pelajaran	Wilayah	Foto	Aksi
1	SITI NUR KHOIFAH	Perempuan	2020-2021	Banat 1		Cetak
2	Zakiyatul Fitriyah	Perempuan	2020-2021	Banat 1		Cetak

Gambar 4.62. Form Menu Santri Admin Wilayah

Adapun segmen program pada menu santri admin wilayah dapat dilihat pada segmen program berikut.

Segmen Program 4.38. Data Santri Admin Wilayah

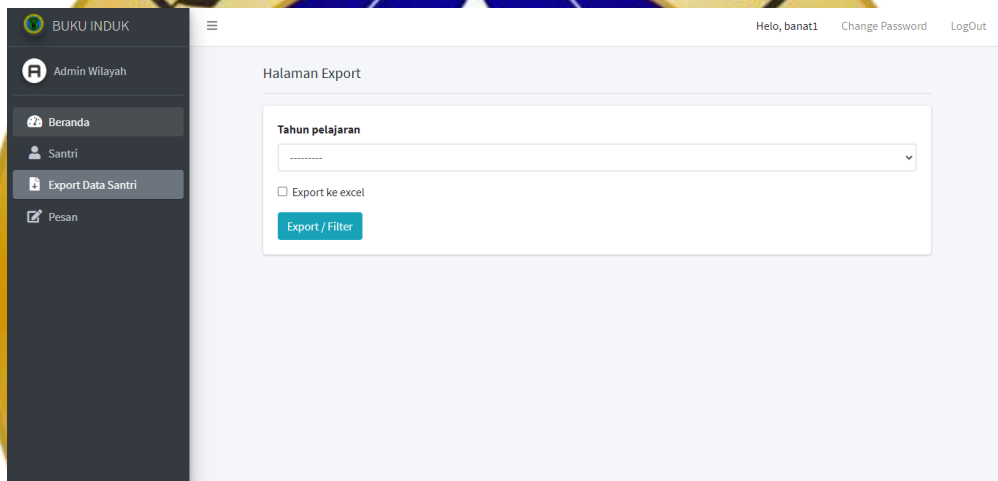
```
login_required(login_url='login')  
@ijinkan_pengguna(yang_diizinkan=['wilayah'])  
def santriwilayah(request):  
    list_santri = Santri.objects.all().filter(wilayah=request.user.wilayah)  
    filter_santri = SantriFilter(request.GET, queryset=list_santri)  
    list_santri=filter_santri.qs  
    paginated_filter_santri = Paginator(filter_santri.qs, 2)  
    page_number = request.GET.get('page')  
    santri_page_obj = paginated_filter_santri.get_page(page_number)
```

Segmen Program 4.39. Data Santri Admin Wilayah (Lanjutan)

```
context={
    'judul': 'Halaman Santri',
    'menu': 'santriwilayah',
    'list': list_santri,
    'filter_data_santri':filter_santri,
    'santri_page_obj':santri_page_obj,
}
return render(request, 'appsantri/santriwilayah.html', context)
```

25. Form menu export data santri admin wilayah

Menu *export* data santri wilayah adalah halaman untuk *export* data santri wilayah ke bentuk *excel*. Form *export* data santri admin wilayah terlihat pada gambar 4.63.



Gambar 4.63. Form Menu *Export* Data Santri Admin Wilayah

Adapun segmen program pada menu *export* data santri admin lembaga dapat dilihat pada segmen program berikut.

Segmen Program 4.40. *Export* Data Santri Wilayah

```
login_required(login_url='login')
@ijinkan_pengguna(yang_diizinkan=['wilayah'])
def export_santriwilayah (request):
    header = 'Halaman Export Data'
    queryset = Santri.objects.all()
    form = SantrisSearchForm(request.POST or None)
    context = {
        "judul": header,
        "queryset": queryset,
        "menu": "export_santriwilayah ",
        "form": form,
    }
}
```

Segmen Program 4.41. Export Data Santri Wilayah (Lanjutan)

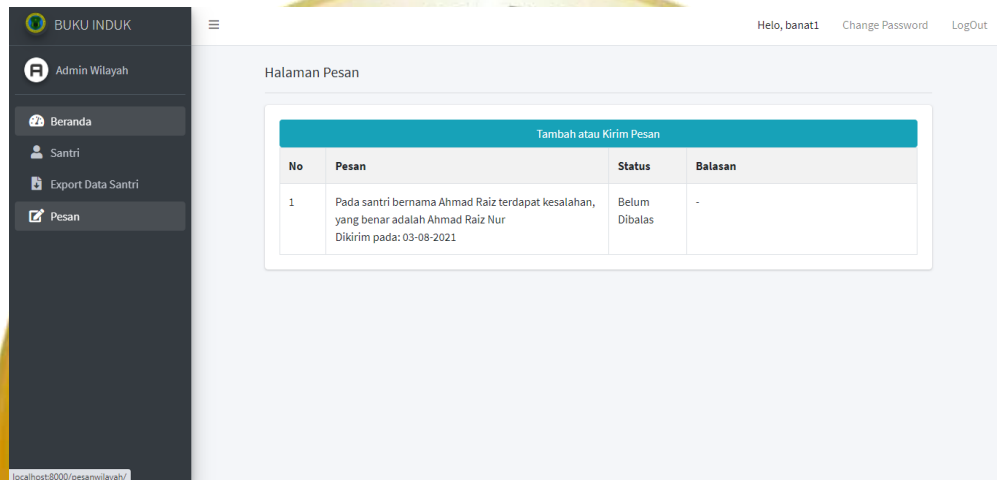
```
if request.method == 'POST':
    tahun_pelajaran = form['tahun_pelajaran'].value()
    if (tahun_pelajaran != ""):
        queryset = Santri.objects.filter(tahun_pelajaran_id=tahun_pelajaran)
    if form['export_ke_Excel'].value() == True:
        response = HttpResponse(content_type='application/ms-excel')
        response['Content-Disposition'] = 'attachment; filename="santri.xls"'
        wb = xlwt.Workbook(encoding='utf-8')
        ws = wb.add_sheet('Santri')
        row_num = 0
        font_style = xlwt.XFStyle()
        font_style.font.bold = True
        columns = ['NIS', 'NISN', 'NIK', 'NO KK', 'NAMA', 'TEMPAT LAHIR', 'TANGGAL LAHIR', 'JENIS KELAMIN', 'AGAMA', 'ALAMAT', 'KECAMATAN', 'KABUPATEN', 'PROPINSI', 'HOBI', 'CITA CITA', 'ANAK KE', 'JUMLAH SAUDARA', 'TGL MASUK PESANTREN', 'STATUS ASAL SANTRI', 'SEKOLAH ASAL', 'ALAMAT SEKOLAH ASAL', 'WILAYAH', 'LEMBAGA', 'DINIYAH', 'TAHUN PELAJARAN', 'NAMA AYAH', 'STATUS HIDUP AYAH', 'PENDIDIKAN AYAH', 'PEKERJAAN AYAH', 'NO HP AYAH', 'NAMA IBU', 'STATUS HIDUP IBU', 'PENDIDIKAN IBU', 'PEKERJAAN IBU', 'NO HP IBU', 'NAMA WALI', 'PENDIDIKAN WALI', 'PEKERJAAN WALI', 'NO HP WALI', 'PENGHASILAN ORANG TUA ATAU WALI PERBULAN']
        for col_num in range(len(columns)):
            ws.write(row_num, col_num, columns[col_num], font_style)
            font_style = xlwt.XFStyle()
            rows = queryset.filter(wilayah=request.user.wilayah).values_list('nis', 'nisl', 'nik', 'no_kk', 'nama', 'tempat_lahir', 'tanggal_lahir', 'jenis_kelamin', 'agama', 'alamat', 'kecamatan', 'kabupaten', 'propinsi', 'hobi', 'cita_cita', 'anak_ke', 'jumlah_saudara', 'tanggal_masuk_pesantren', 'status_asal_santri', 'sekolah_asal', 'alamat_sekolah_asal', 'wilayah', 'lembaga', 'diniyah', 'tahun_pelajaran', 'nama_ayah', 'status_hidup_ayah', 'pendidikan_ayah', 'pekerjaan_ayah', 'no_hp_ayah', 'nama_ibu', 'status_hidup_ibu', 'pendidikan_ibu', 'pekerjaan_ibu', 'no_hp_ibu', 'nama_wali', 'pendidikan_wali', 'pekerjaan_wali', 'no_hp_wali', 'penghasilan_orang_tua_atau_wali_perbulan')
            for row in rows:
                row_num += 1
                for col_num in range(len(row)):
                    ws.write(row_num, col_num, row[col_num], font_style)
        wb.save(response)
        return response
context = {
    "form": form,
    "header": header,
    "queryset": queryset,
```


Segmen Program 4.42. *Export Data Santri Wilayah* (Lanjutan)

```
"menu": "export_santriwilayah",}  
return render(request, 'appsantri/export_santriwilayah.html', context)
```

26. Form menu pesan admin wilayah

Menu pesan admin wilayah adalah halaman untuk mengimkan pesan wilayah ke pusat jika terdapat kesalahan pada data santri wilayah dan akan mendapat konfirmasi balasan jika data telah diperbaiki, terlihat pada gambar 4.64.



Gambar 4.64. Form Menu Pesan Admin Wilayah

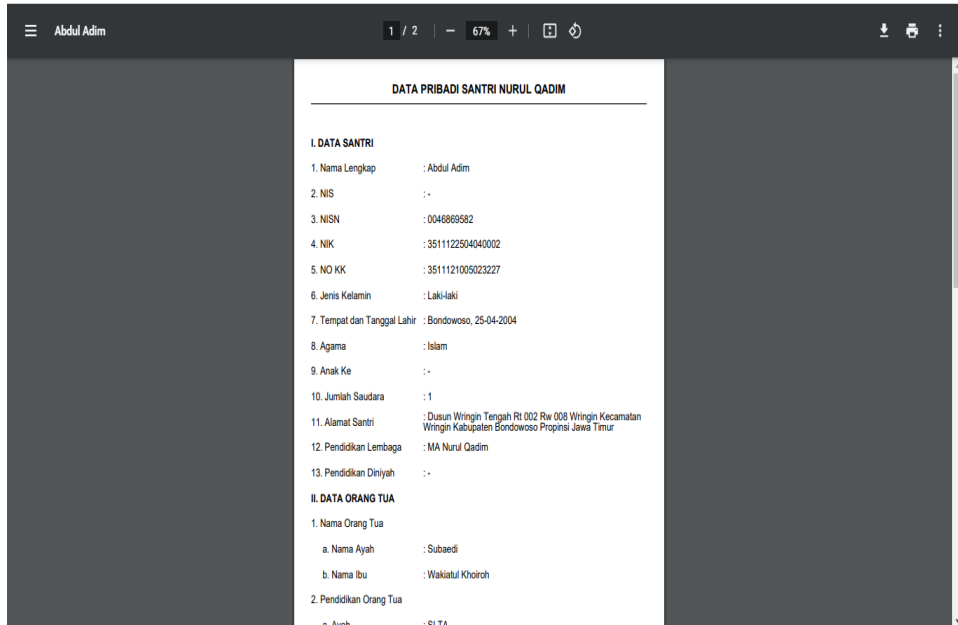
Adapun segmen program pada menu pesan admin wilayah dapat dilihat pada segmen program berikut.

Segmen Program 4.43. Menu Pesan Admin Wilayah

```
@login_required(login_url='login')  
@ijinkan_pengguna(yang_ditizinkan=['wilayah'])  
def pesanwilayah(request):  
    list_pesan = Pesan.objects.all().filter(wilayah=request.user.wilayah)  
    context={  
        'judul': 'Halaman Pesan',  
        'menu': 'pesanadminwilayah',  
        'list': list_pesan,  
    }  
    return render(request, 'data/pesanwilayah.html', context)
```

27. Laporan data santri pdf

Laporan data santri pdf terlihat pada Gambar 4.65. Untuk mengunduh atau mencetak laporan data santri tersebut caranya adalah dengan klik *icon* yang sudah ada seperti yang terlihat pada gambar 4.65.



Gambar 4.65. Laporan Data Santri Pdf

28. Export data santri

Export data santri ke bentuk excel terlihat pada gambar 4.66.

A	B	C	D	E	F	G	H	I	J
NIS	NISN	NIK	NO KK	NAMA	TEMPAT LAHIR	TANGGAL LAHIR	JENIS KELAMIN	AGAMA	ALAM
0046869582	3511122504040002	3511121005023227	3511121005023227	Abdul Adim	Bondowoso	38102	1	1	Dusun
0049897571	3513082503040001	3513081211056254	3513081211056254	Abdur Rohman	Probolinggo	38071	1	1	Dusun
0046382315	3513100906040002	3513102805080024	3513102805080024	Ahmad Zaini	Probolinggo	38147	1	1	Dusun
0059805693	2901072301050002	3201071203080100	3201071203080100	Aji Khairiansyah Astery	Sorong	38375	1	1	Jl. Piss
0036418460	3513120412030003	3513122512100002	3513122512100002	Alan Nuri Hidayatullah	Probolinggo	37959	1	1	Dusun
0037259953	3512032610030001	3512030411052844	3512030411052844	Eka Octa Andriansyah	Stubondo	37920	1	1	Kp. Tar
0045151855	3305251408040002	3305251801110030	3305251801110030	Erik Gus Stefani	Kabupaten	38213	1	1	Tegok
003288176	3513131803030002	3513131511053473	3513131511053473	Hafidul Ahkam	Probolinggo	37698	1	1	Dusun
0041818404	1505041803040002	1505041103082888	1505041103082888	M. Alfa Rizki	Muaru Jambi	38064	1	1	Danau
0035682038	3511112508030003	3511111005028106	3511111005028106	Muhammad Ikram Amin Darmawan	Bondowoso	37858	1	1	Sumbe
0042992776	3513123112040001	3513121411052030	3513121411052030	M. Sholihin Zahni	Probolinggo	38352	1	1	Dusun
0047040398	35131020940001	3513102785130005	3513102785130005	Maulana Kholidir Ridho	Probolinggo	38250	1	1	Dusun
0044410556	3513111004040001	3513111311056187	3513111311056187	Moh. Babur Rizki	Probolinggo	38087	1	1	Dusun
0039912167	3513100405030003	3513102605080054	3513102605080054	Moh. Karim	Probolinggo	37745	1	1	Dusun
0033186116	351312040512030003	3512040511052299	3512040511052299	Muhammad Muzeyyin	Stubondo	37960	1	1	Kp. Tre
0034424283	3528081609030002	3528082504069674	3528082504069674	Rahmat Hidayat	Pamekasan	37880	1	1	Dn Ka
0042274938	3513091411040001	3513090310130004	3513090310130004	Sawaluddin	Probolinggo	38305	1	1	Dusun
0041736371	1806132412040001	1806132402100027	1806132402100027	Taufiqur Rahman	Probolinggo	38028	1	1	Dusun
0037376536	3512051506030001	3512050511053268	3512050511053268	Uman Sayyid Abdullah	Margoyoso	38345	1	1	Dusun
0042643191	351313408040001	3513131511055896	3513131511055896	Rizky Tanjung Putra	Indramayu	37257	1	1	Cakung
0037376544	3512051506030001	3512050511053268	3512050511053268	Abdillah Aidi Makmun	Stubondo	37787	1	1	Kp. Krc
0036417979	3513122510030004	3513121411052844	3513121411052844	Achmad Allanzi	Probolinggo	38213	1	1	Dusun
0042920910	3513100906040002	3513102805080024	3513102805080024	Ahmad Syaifulah Yusuf	Stubondo	37885	1	1	Kp. Ka
0041667353	3512010704040001	3512010311052793	3512010311052793	Ahmad Fudholi	Probolinggo	37919	1	1	Dusun
00115209894	3601122207010005	3601121003140002	3601121003140002	Ahmad Zaini	Probolinggo	38249	1	1	Dusun
0046140630	3513090608040002	3513091305080012	3513091305080012	Amin Fauzan	Stubondo	38084	1	1	Kampu
002783171	35131212101040001	3513121211120020	3513121211120020	Fauzi Azima	Pandeglang	37459	1	1	Kp. Pa
				Muhammad Daniel Maulana	Probolinggo	38204	1	1	Dusun
				M. Hasan Annon Maulana	Probolinggo	38007	1	1	Dusun

Gambar 4.66. Hasil Export Data Santri Ke Bentuk Excel

4.4 Uji Coba

Uji coba atau pengujian sistem dilakukan dengan metode *black-box testing*. Pengujian ini dimaksudkan untuk mengetahui apakah *output* yang dihasilkan dari pengolahan sistem benar-benar sesuai dengan *output* yang diharapkan oleh

pengguna sistem buku induk santri terpusat Pondok Pesantren Nurul Qadim. Untuk hasil pengujian terhadap terhadap beberapa proses memberikan hasil sebagai berikut:

a. Pengujian internal

Pengujian internal dilakukan dengan cara mengamati hasil eksekusi melalui data uji dan memeriksa fungsionalitas dari perangkat lunak. Dari pengujian internal yang dilakukan bersama Bapak Abu Tholib, M.Kom dengan sembilan pertanyaan berkaitan dengan penggunaan fungsi aplikasi didapatkan jawaban sembilan sesuai dan nol tidak sesuai. Adapun bukti pengujian terdapat dilampiran pengujian internal.

b. Pengujian eksternal

Pengujian eksternal dilakukan dengan cara menguji program langsung pada *user*. Pengujian ini bertujuan agar dapat mengetahui apakah sistem sudah berjalan dengan baik atau belum, termasuk kelebihan dan kekurangan pada sistem tersebut. Diantaranya sebagai berikut:

Tabel 4.20. Hasil Pengujian Eksternal

★ No	User	Jawaban Pertanyaan ke					★ Total
		Q1	Q2	Q3	Q4	Q5	
1	Responden 1	4	4	4	4	4	20
2	Responden 2	4	4	4	4	4	20
3	Responden 3	4	3	3	4	4	18
4	Responden 4	4	4	3	4	3	18
5	Responden 5	4	4	3	4	3	18
6	Responden 6	4	4	4	3	3	18
7	Responden 7	4	3	4	4	3	18
8	Responden 8	4	3	3	4	3	17

Hasil pengujian dari delapan *user* dan lima pertanyaan didapatkan data sebagai berikut:

- a) Jumlah responden pada setiap titik respon dan total skor
- Responden yang menjawab Sangat Setuju (27) = $27 \times 4 = 108$
 - Responden yang menjawab Setuju (13) = $13 \times 3 = 39$
 - Responden yang menjawab Cukup (0) = $0 \times 2 = 0$
 - Responden yang menjawab Tidak Setuju (0) = $0 \times 1 = 0$
- Total Skor = $108 + 39 + 0 + 0 = 147$
- b) Jumlah skor tertinggi (Y)
 $Y = 4 \times 5 \times 8 = 160$
- c) Total skor responden diperoleh angka 147, maka penilaian interpretasi responden adalah:

$$\begin{aligned} \text{Index\%} &= \text{Total Skor} / Y \times 100 \\ &= \left(\frac{147}{160}\right) \times 100 \\ &= 91,8\% \text{ Kategori } \mathbf{Sangat\ Baik} \end{aligned}$$

Dari hasil diatas maka dapat disimpulkan bahwa responden **Sangat Baik** aplikasi tersebut sudah layak digunakan.

4.5 Perawatan

Setelah melakukan pengujian terhadap sistem baru yang dibuat, tahap berikutnya adalah perawatan sistem. Ketika dijalankan mungkin saja masih ada *errors* yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur pada *software* yang dibuat. Maka dari itu perlu adanya perawatan dan pengembangan sistem, validasi data dan updating data.