

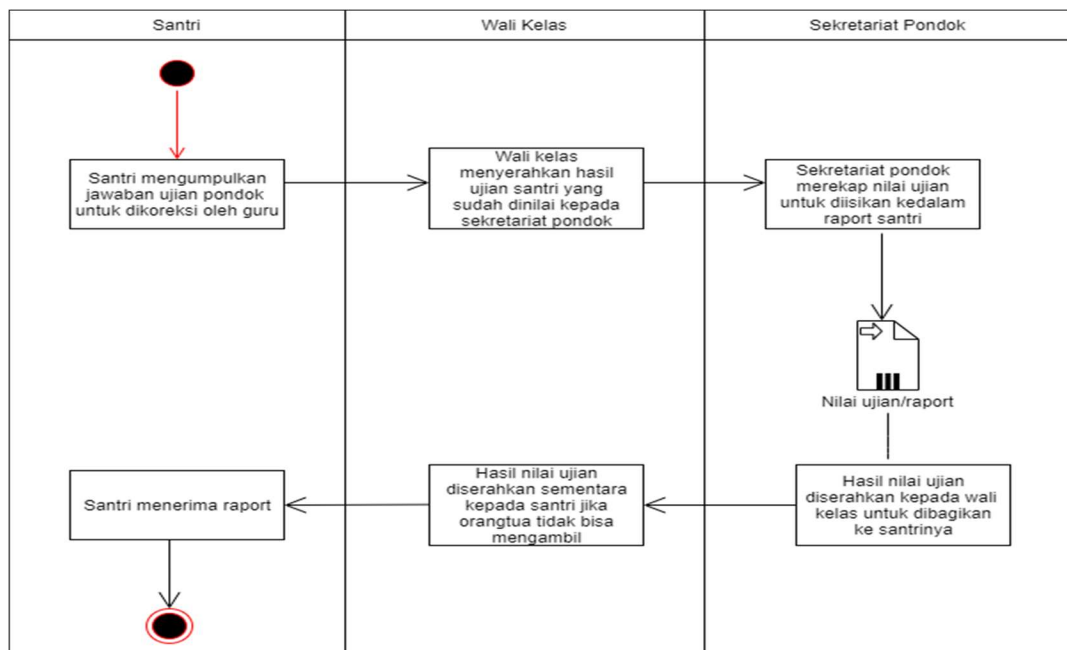
BAB 3

MODEL SISTEM

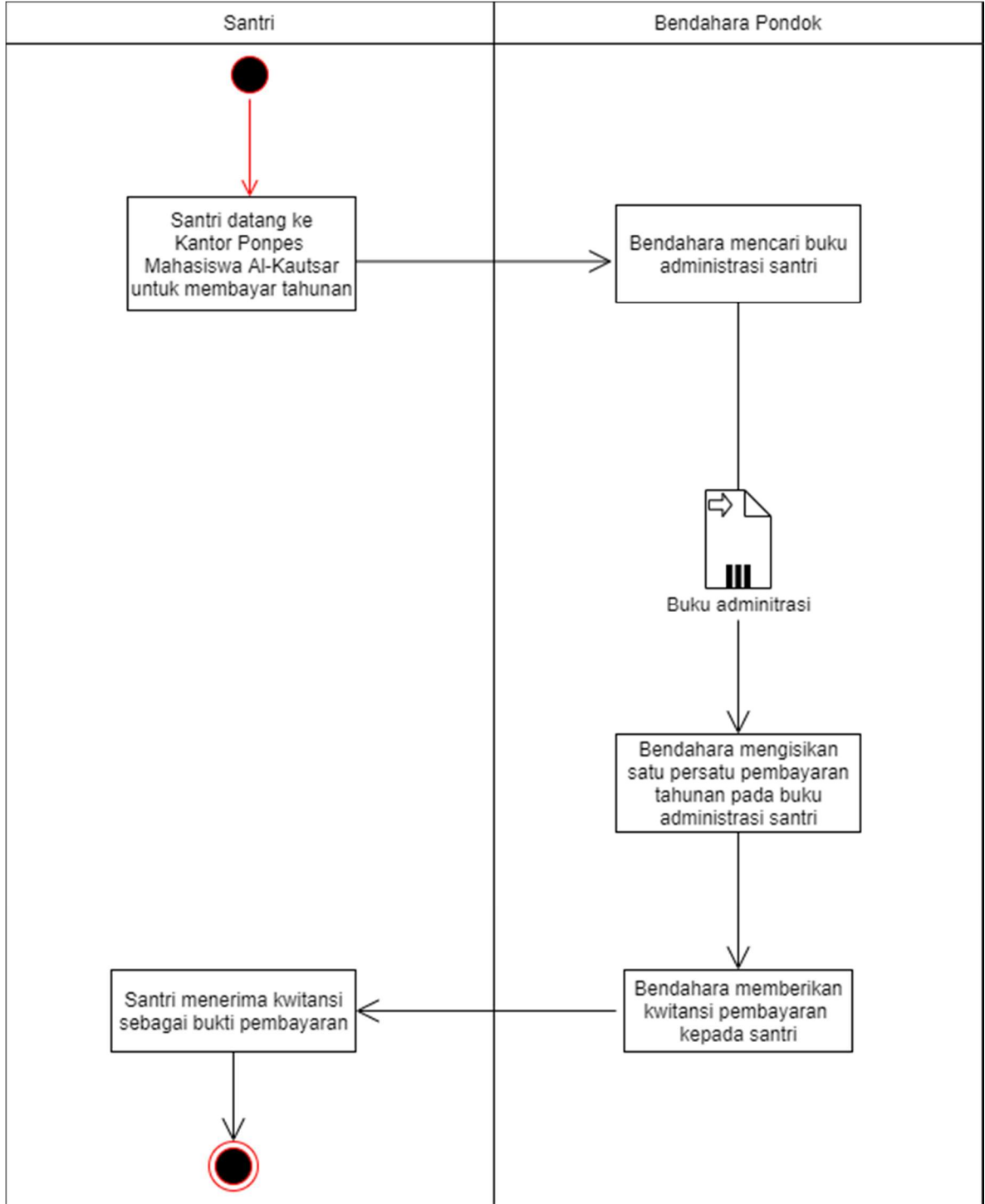
3.1. Proses Bisnis

3.1.1. Proses Bisnis Saat Ini

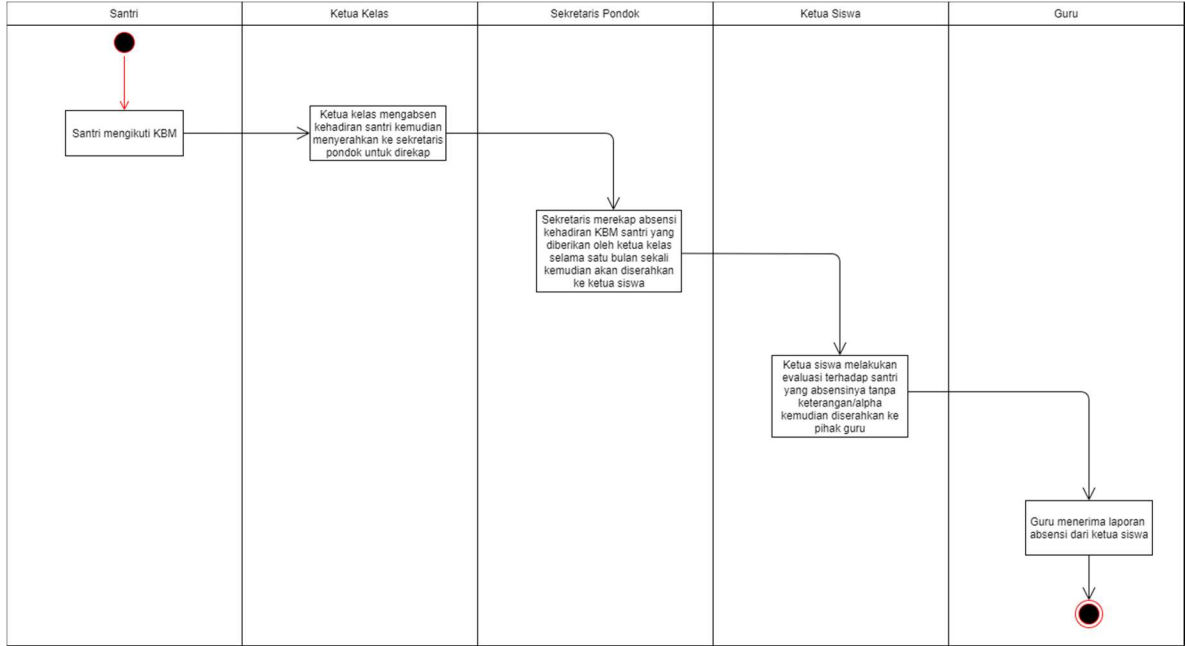
Pada saat ini, Pondok Pesantren Mahasiswa Al-Kautsar Kota Malang belum memiliki sistem informasi yang dapat mengelola data pondok. Selama ini juga untuk informasi pembayaran uang Pondok Pesantren Mahasiswa Al-Kautsar Kota Malang mencakup uang kamar dan uang KBM, kurikulum pembelajaran santri, biodata santri, semuanya didistribusikan via WhatsApp. Sehingga, *file* ataupun informasi berupa pengumuman terkadang ketumpuk dan hilang. Lalu presensi kehadiran dalam KBM dilakukan secara manual, akibatnya data rekapan sering terjadi kehilangan. Terlebih untuk pembayaran Pondok Pesantren Mahasiswa Al-Kautsar Kota Malang mengalami kendala, ketika pengecekan data santri yang sudah lunas dan yang belum lunas pembayarannya terjadi ketidakvalidan, dan akhirnya harus melakukan pengecekan manual pada santri satu per satu dengan melakukan pemanggilan untuk memastikan datanya sesuai atau tidak.



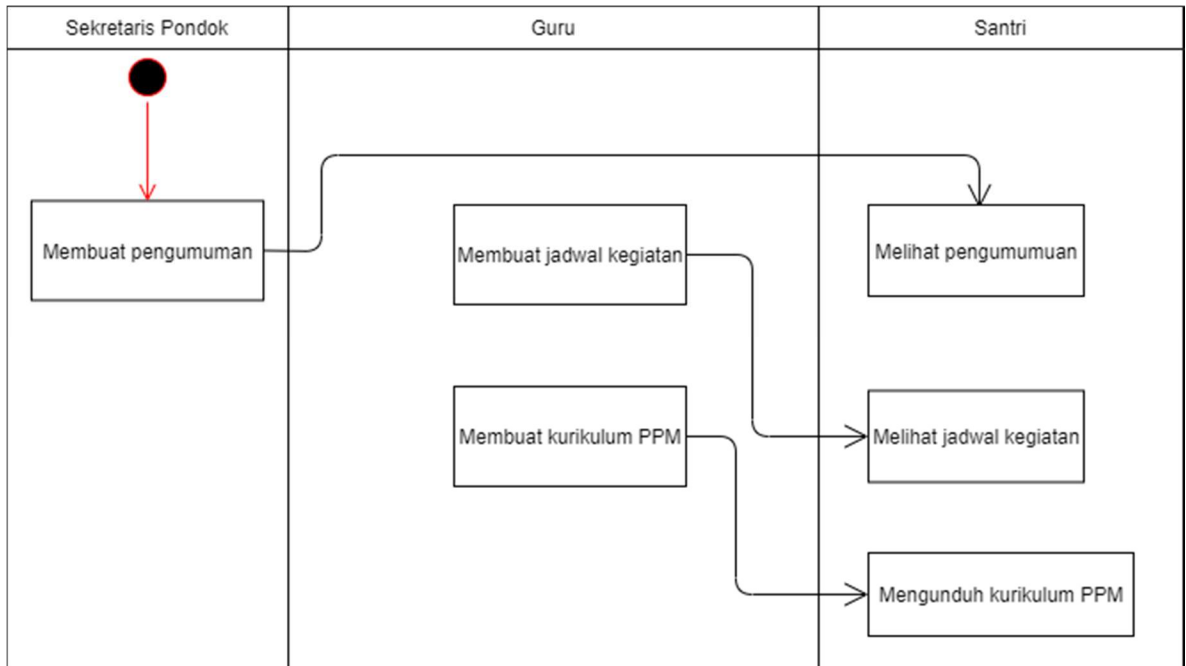
Gambar 3. 1 Bisnis Proses Penilaian Santri Saat Ini



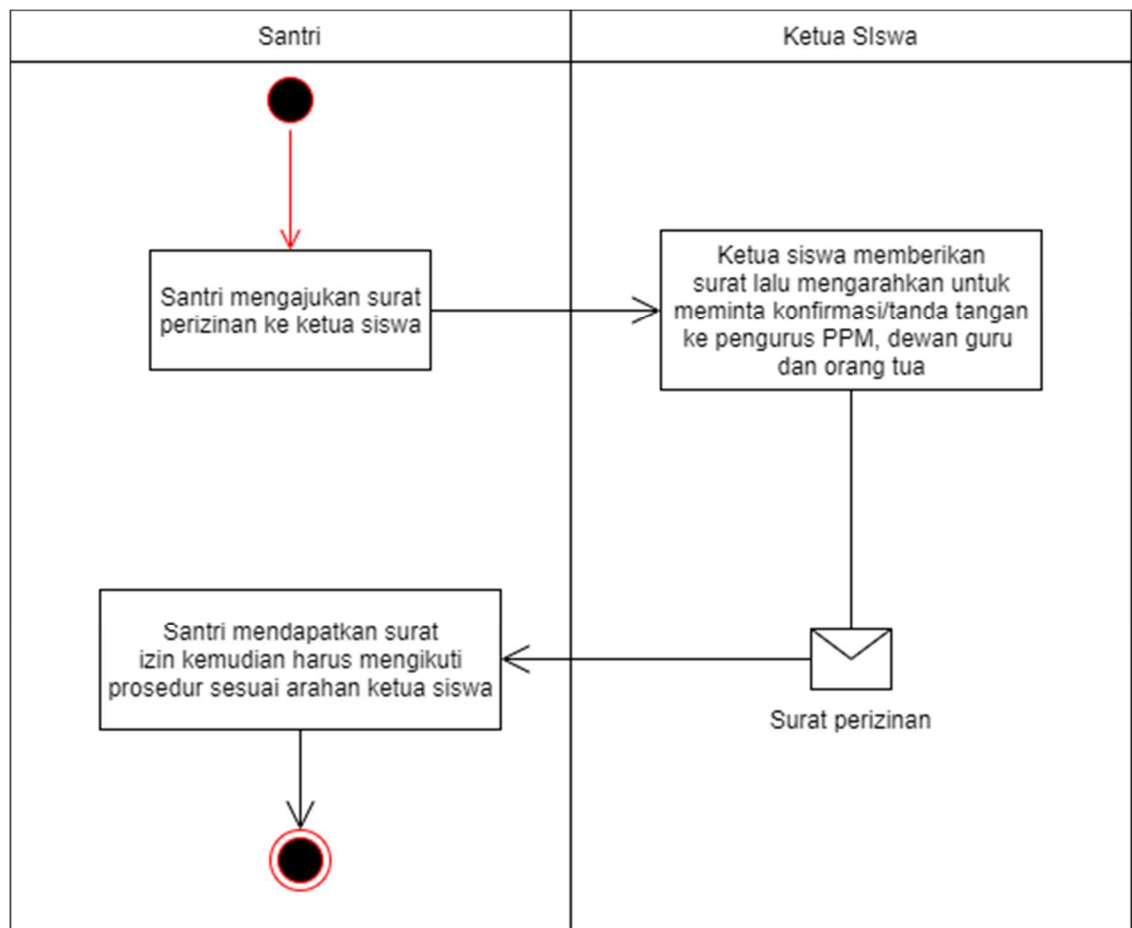
Gambar 3. 2 Bisnis Proses Pembayaran Tahunan Saat Ini



Gambar 3. 3 Bisnis Proses Sistem Kehadiran Santri Saat Ini



Gambar 3. 4 Bisnis Proses Informasi Santri Saat Ini



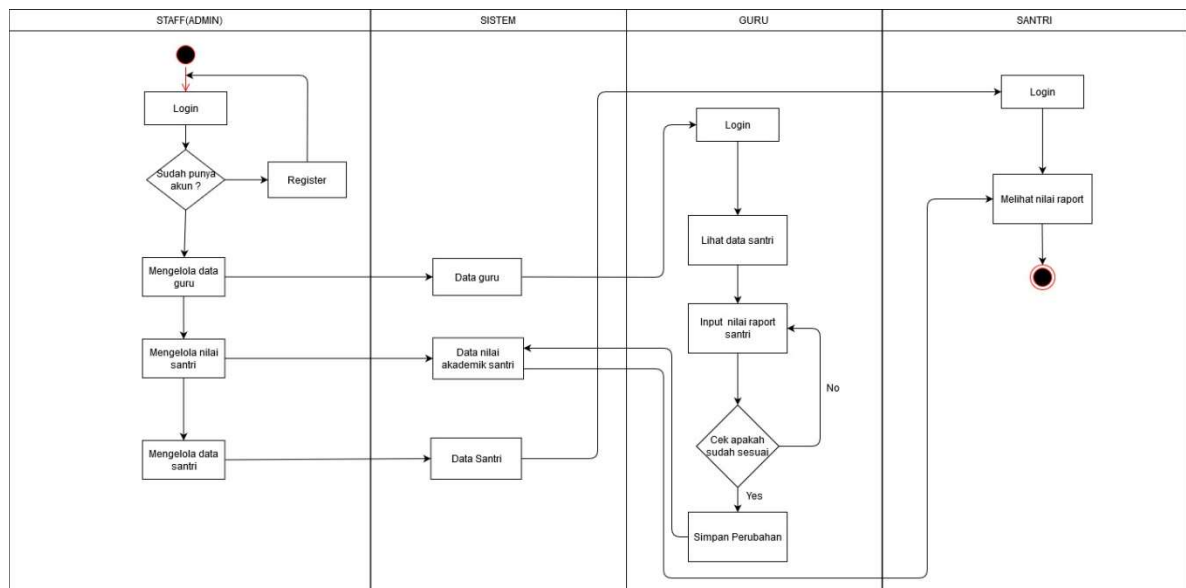
Gambar 3. 5 Bisnis Proses Pengajuan Surat Izin Saat Ini

3.1.2. Proses Bisnis Usulan

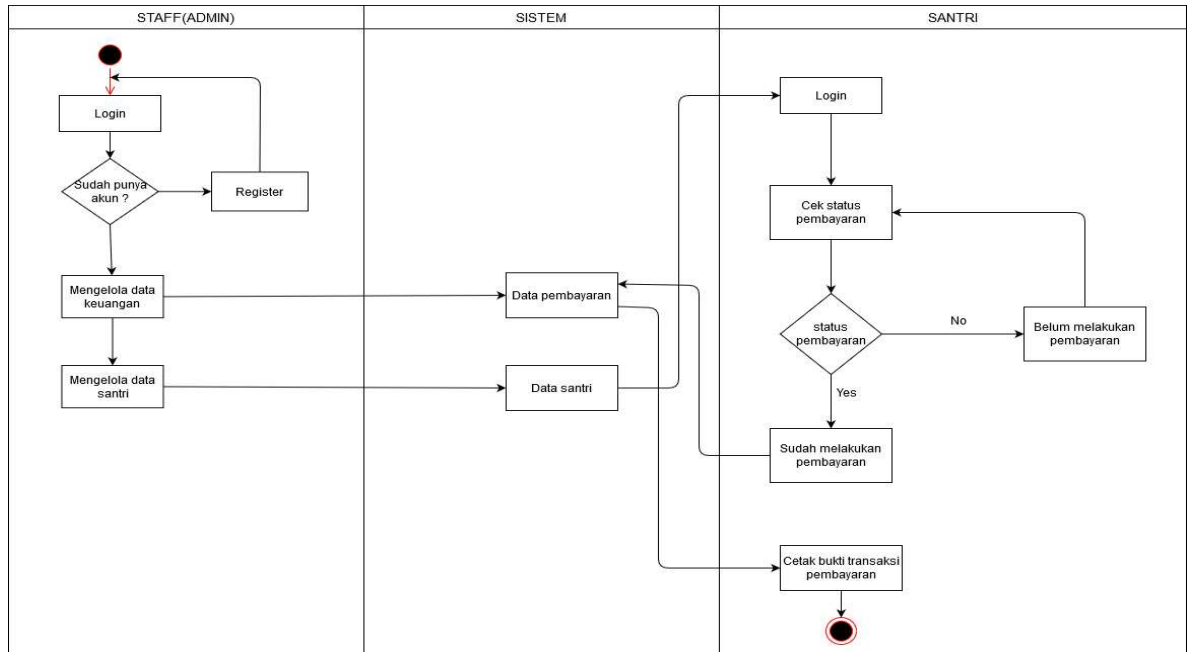
Dari hasil analisis permasalahan yang dilakukan, maka diusulkan pemecahan masalah sebagai berikut.

- Dibuat sistem terkomputerisasi untuk mengelola pendataan berbasis *website*.
- Terdapat *database* yang dapat menyimpan semua data admin dan dapat digunakan sebagai media pengolahan data.
- Diterapkan sistem keamanan *username/email* dan *password* pada *website* tersebut, agar tidak sembarang orang yang dapat membuka *website* tersebut.

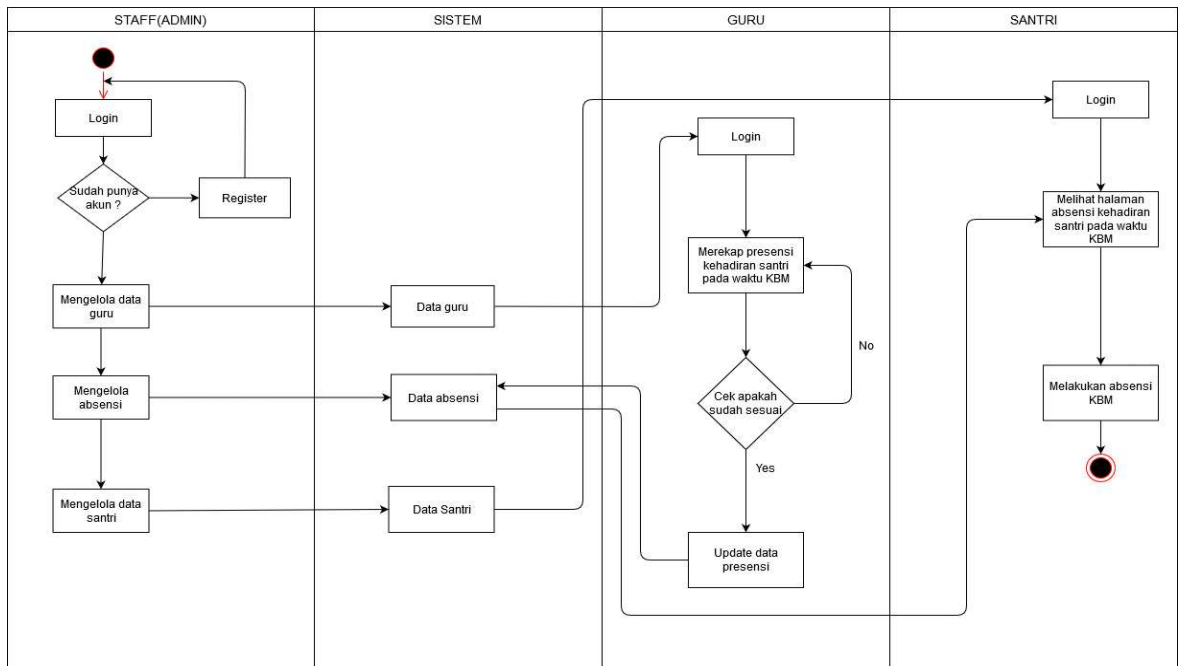
Secara umum usulan sistem dalam proses perancangan sistem yaitu akan membuat *website* pendataan pada Pondok Pesantren Mahasiswa Al-Kautsar Kota Malang dengan harapan mampu menangani permasalahan yang ada. Semoga *website* ini dapat membantu dan mempermudah dalam mengatasi masalah yang ada.



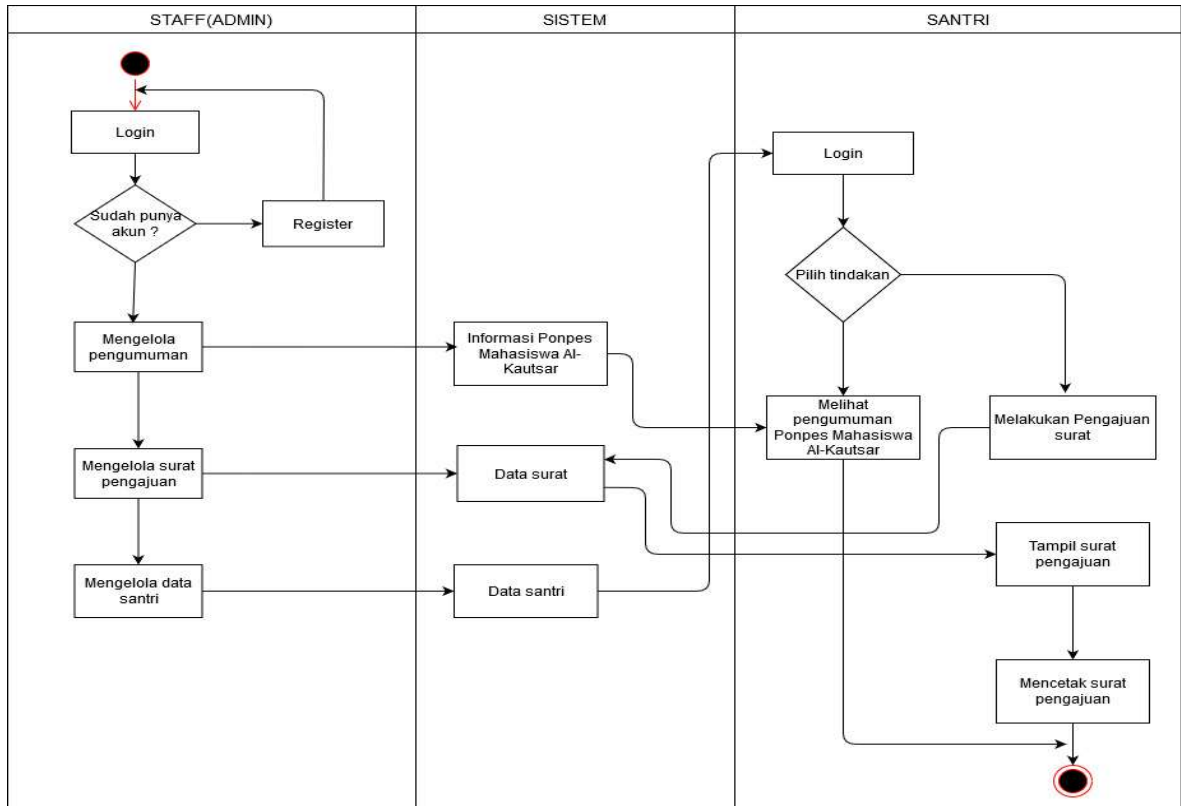
Gambar 3. 6 Bisnis Proses Usulan Penilaian Santri



Gambar 3. 7 Bisnis Proses Usulan Pembayaran Tahunan



Gambar 3. 8 Bisnis Proses Usulan Sistem Kehadiran Santri

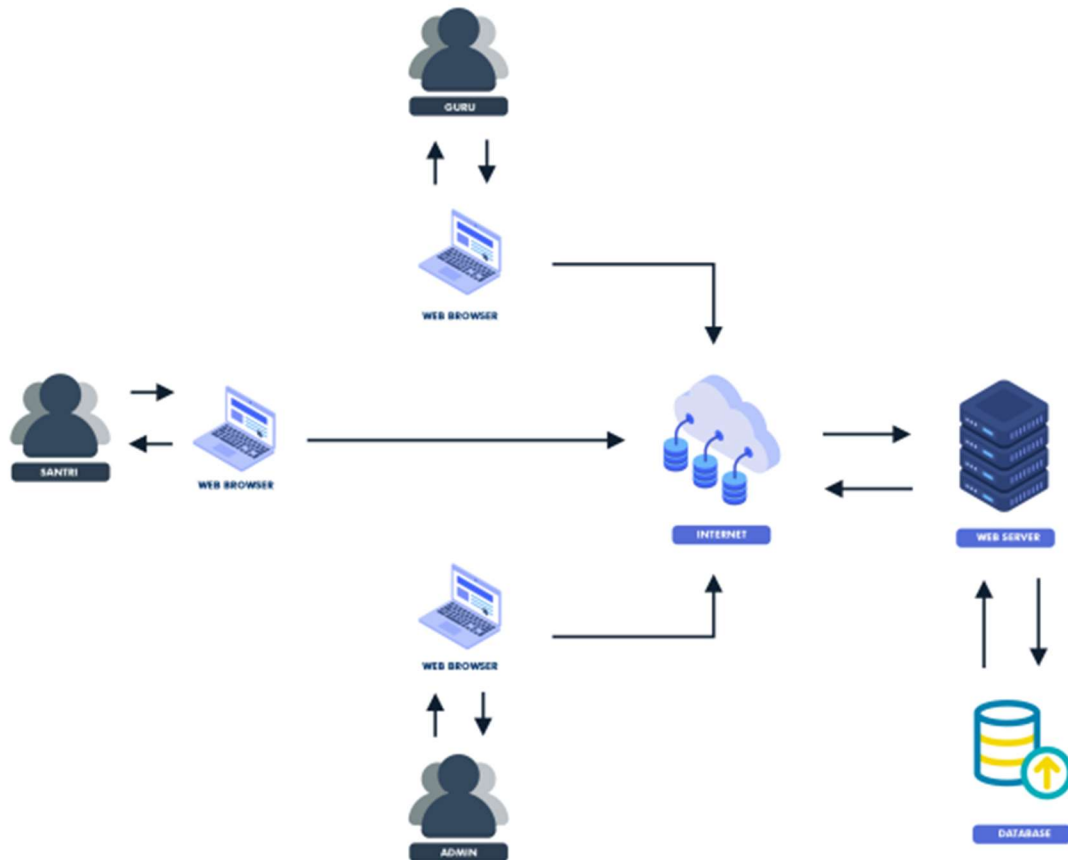


Gambar 3. 9 Bisnis Proses Usulan Pengumuman Dan Surat Pengajuan

3.2. Arsitektur dan Desain Usulan

Pada sub bab ini menjelaskan tentang segala sesuatu yang berkaitan dengan sistem yang diusulkan dimulai dari Arsitektur Sistem, *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan Diagram Alir.

3.2.1. Arsitektur Sistem



Gambar 3. 10 Arsitektur Sistem

Pengguna memerlukan sebuah *device* yang terhubung ke jaringan internet untuk mengakses *website*. Setelah terhubung dengan internet, *web browser* akan meminta ke *web server*, yaitu perangkat lunak yang berfungsi sebagai penerima permintaan yang dikirimkan melalui *browser* kemudian memberikan tanggapan permintaan dalam bentuk halaman situs *web* atau lebih umumnya dalam dokumen *HTML*. Semua yang berhubungan dengan *website* biasanya juga berhubungan dengan *web server*, karena tugas *web server* adalah

mengatur semua komunikasi yang terjadi antara *browser* dengan *server* untuk memproses sebuah *website*.

Kemudian *web server* meminta dari *database*. Jika *database* telah memberikan respon pada *web server*, maka *web server* akan memberikan respon pada *web browser*, sehingga pengguna akan mendapatkan informasi. Pengguna dapat menggunakan *web browser* seperti Mozilla Firefox, Google Chrome, Microsoft Edge, dan sebagainya.

3.2.2. Use Case Diagram

Pada sub sub bab ini, menjelaskan tentang fitur-fitur yang terdapat dalam sistem yang diusulkan dan disertai dengan keterkaitan dengan aktor yang menjalankan fitur tersebut. Secara umum, penyusunan *use case* diagram melalui beberapa tahapan, seperti:

a) Definisi Aktor

Tabel 3. 1 Definisi Aktor *Use Case* Diagram

No.	Aktor	Deskripsi
1	Admin	Admin adalah orang yang memiliki hak akses untuk melakukan operasi pengelolaan data pondok, guru, dan santri.
2	Guru	Guru adalah orang yang bertugas untuk mengelola materi, kelas, kehadiran santri, nilai santri, dan rapor.
3	Santri	Santri dapat mendapatkan informasi seperti pengumuman kegiatan pondok, biodata, nilai per semester, status pembayaran, dan pengajuan surat.

b) Definisi Use Case

Tabel 3. 2 Definisi *Use Case* Diagram

No	Use Case	Deskripsi
1	<i>Login</i>	Merupakan proses untuk masuk ke halaman <i>website</i> .
2	Mengelola hak akses	merupakan proses untuk membuat akun yang nantinya akan diberikan kepada guru dan santri.
3	Mengelola data <i>user</i>	Merupakan proses penambahan <i>user</i> yang dapat mengakses sistem.
4	Mengelola pengumuman	Merupakan proses yang dilakukan oleh admin untuk memberikan pengumuman atau informasi seputar kegiatan pondok kepada santri.
5	Melihat kelas	Merupakan proses untuk melihat daftar kelas yang ada pada ponpes mahasiswa al-kautsar kota malang.
6	Melihat anggota kelas	Merupakan proses untuk melihat daftar santri berdasarkan kelasnya masing-masing.
7	Melihat materi	Merupakan proses untuk melihat daftar materi yang ada pada ponpes

		mahasiswa al-kautsar kota malang.
8	Mengelola data keuangan	Merupakan proses yang nantinya akan menampilkan status pembayaran per semester berupa keterangan lunas dan belum lunas pada halaman santri sesuai akun.
9	Mengelola data santri	Merupakan proses yang dilakukan oleh admin untuk mengisi data santri berupa biodata pada sistem.
10	Mengelola data guru	Mengelola data guru merupakan proses yang dilakukan oleh admin untuk mengisi data guru berupa biodata pada sistem.
11	Mengelola pengajuan surat dari santri	Merupakan proses yang dilakukan oleh admin untuk mengelola surat pengajuan dari santri.
12	Mengelola pesan	Merupakan proses yang dilakukan oleh admin untuk memberikan <i>feedback</i> berupa pesan kepada santri yang mengajukan surat.
13	Melihat rekapan kehadiran	Merupakan proses yang dilakukan oleh admin untuk melihat data kehadiran santri yang telah dimasukan oleh guru pada sistem.
14	Melihat rapor	Merupakan proses yang dilakukan oleh admin untuk melihat rapor yang telah dikelola oleh guru dan admin juga dapat mencetak rapor tersebut.
15	Mengelola tahun akademik	Merupakan proses yang dilakukan oleh admin untuk mengaktifkan tahun akademik yang ingin dipilih.
16	Mengelola jadwal mengaji	Merupakan proses yang dilakukan oleh admin untuk menambahkan jadwal mengaji.
17	Melihat data guru	Merupakan proses melihat biodata guru.
18	Mengelola kelas	Merupakan proses untuk menambahkan daftar kelas.
19	Mengelola anggota kelas	Merupakan proses untuk untuk melihat daftar santri berdasarkan kelasnya masing-masing.
20	Mengelola materi	Merupakan proses untuk menambahkan daftar materi pelajaran santri.
21	Mengelola nilai santri	Mengelola nilai rapor merupakan proses yang dilakukan oleh guru untuk mengelola nilai santri.
22	Mengelola daftar kehadiran	Merupakan proses yang dilakukan oleh admin untuk merekap data kehadiran santri.
23	Mengelola data rapor	Merupakan proses yang dilakukan oleh guru untuk mengelola data rapor.
24	Melihat jadwal mengajar	Merupakan proses yang dilakukan oleh guru untuk melihat jadwal mengajar.
25	Melihat data santri	Merupakan proses melihat biodata santri.
26	Melihat pengumuman	Merupakan proses melihat pengumuman kegiatan pondok.
27	Melihat tagihan pembayaran	Merupakan proses melihat status pembayaran per semester.
28	Melakukan pengajuan surat	Merupakan proses yang dilakukan santri untuk mengajukan surat
29	Melihat transkrip nilai	Merupakan proses menampilkan data nilai per semester yang ada di sistem.
30	Melihat pesan masuk	Merupakan proses menerima pemberitahuan yang dikirimkan oleh admin berupa pesan.

c) Skenario *Use Case*Nama *Use Case*: *Login*

Skenario:

Tabel 3. 3 Skenario *Use Case Login*

Aksi Aktor	Reaksi Sistem
Skenario Normal	
Memasukkan <i>username</i> , dan <i>password</i>	Memeriksa valid tidaknya data masukan
	Masuk ke dalam sistem informasi akademik dan pengelolaan pondok
Skenario Alternatif	
Memasukkan <i>username</i> , dan <i>password</i>	Memeriksa valid tidaknya data masukan
	Menampilkan pesan login tidak valid
Memasukkan <i>username</i> , dan <i>password</i> yang valid	Memeriksa valid tidaknya data masukan
	Masuk ke dalam sistem informasi akademik dan pengelolaan pondok

Nama *Use Case*: Mengelola Data *User* (Admin)

Skenario:

Tabel 3. 4 Skenario *Use Case Mengelola Data User*

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu <i>user management</i>	Menampilkan menu <i>user management</i> dan tiga sub menunya, yaitu admin, guru, dan santri
Memilih sub menu <i>user</i>	Menampilkan tabel <i>CRUD user</i>
Menambahkan <i>user</i> dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data <i>user</i> baru
Menekan tombol “Simpan”	Menyimpan data <i>user</i> baru ke basis data
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu <i>user management</i>	Menampilkan menu <i>user management</i> dan tiga sub menunya, yaitu admin, guru, dan santri
Memilih sub menu <i>user</i>	Menampilkan tabel <i>CRUD user</i>
Menambahkan <i>user</i> dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data <i>user</i> baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data <i>user</i> baru ke basis data

	Menampilkan pesan bahwa data sukses ditambahkan
--	---

Nama *Use Case*: Mengelola Pengumuman (Admin)

Skenario:

Tabel 3. 5 Skenario *Use Case* Mengelola Pengumuman

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu pengumuman	Menampilkan menu pengumuman
Memilih sub menu pengumuman	Menampilkan tabel <i>CRUD</i> pengumuman
Menambahkan pengumuman dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data pengumuman baru
Menekan tombol “Simpan”	Menyimpan data pengumuman baru ke basis data dan memilih pengumuman yang ingin ditampilkan pada halaman santri
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu pengumuman	Menampilkan menu pengumuman
Memilih sub menu pengumuman	Menampilkan tabel <i>CRUD</i> pengumuman
Menambahkan pengumuman dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data pengumuman baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data pengumuman baru ke basis data dan memilih pengumuman yang ingin ditampilkan pada halaman santri
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Data Keuangan (Admin)

Skenario:

Tabel 3. 6 Skenario *Use Case* Mengelola Data Keuangan

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu pembayaran	Menampilkan menu pembayaran
Memilih sub menu pembayaran	Menampilkan tabel <i>CRUD</i> pembayaran

Menambahkan pembayaran dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data pembayaran baru
Menekan tombol “Simpan”	Menyimpan data pembayaran baru ke basis data dan menampilkan status pembayaran pada halaman santri berdasarkan akun
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu pembayaran	Menampilkan menu pembayaran
Memilih sub menu pembayaran	Menampilkan tabel <i>CRUD</i> pembayaran
Menambahkan pembayaran dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data pembayaran baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisikan
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data pembayaran baru ke basis data dan menampilkan status pembayaran pada halaman santri berdasarkan akun
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Data Guru (Admin)

Skenario:

Tabel 3. 7 Skenario *Use Case* Mengelola Data Guru

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu dewan guru	Menampilkan menu dewan guru
Memilih sub menu dewan guru	Menampilkan tabel <i>CRUD</i> dewan guru
Menambahkan biodata guru dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data biodata guru baru
Menekan tombol “Simpan”	Menyimpan data biodata guru baru ke basis data dan menampilkan biodata guru pada halaman guru sesuai akun
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu dewan guru	Menampilkan menu dewan guru
Memilih sub menu dewan guru	Menampilkan tabel <i>CRUD</i> dewan guru
Menambahkan biodata guru dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data biodata guru baru

	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data biodata guru baru ke basis data dan menampilkan biodata guru pada halaman guru sesuai akun
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Data Santri (Admin)

Skenario:

Tabel 3. 8 Skenario *Use Case* Mengelola Data Santri

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu santri	Menampilkan menu dewan santri
Memilih sub menu santri	Menampilkan tabel <i>CRUD</i> santri
Menambahkan biodata santri dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data biodata santri baru
Menekan tombol “Simpan”	Menyimpan data biodata santri baru ke basis data dan menampilkan biodata santri pada halaman santri sesuai akun
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu santri	Menampilkan menu santri
Memilih sub menu santri	Menampilkan tabel <i>CRUD</i> santri
Menambahkan biodata santri dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data biodata santri baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data biodata santri baru ke basis data dan menampilkan biodata santri pada halaman santri sesuai akun
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Pengajuan Surat Santri (Admin)

Skenario:

Tabel 3. 9 Skenario *Use Case* Mengelola Pengajuan Surat Santri

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu pengajuan surat santri	Menampilkan menu pengajuan surat santri
Memilih sub menu pengajuan surat santri	Menampilkan daftar nama santri yang telah mengajukan surat
Menekan tombol “Edit” pada menu pengajuan surat santri	Mengedit status pengajuan surat santri dari diproses menjadi bisa diambil
Menambahkan pesan dengan menekan tombol “Tambah”	Mengirimkan pesan baru kepada yang mengajukan surat sebagai <i>feedback</i>

Nama *Use Case*: Mengelola Tahun Akademik (Admin)

Skenario:

Tabel 3. 10 Skenario *Use Case* Mengelola Tahun Akademik

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu tahun akademik	Menampilkan menu tahun akademik
Memilih sub menu tahun akademik	Menampilkan tabel <i>CRUD</i> tahun akademik
Menambahkan tahun akademik dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data tahun akademik baru
Menekan tombol “Simpan”	Menyimpan data tahun akademik baru ke basis data dan menampilkan tahun akademik pada halaman <i>settings</i> akademik yang bertujuan untuk mengaktifkan tahun akademik yang ingin dipilih
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu tahun akademik	Menampilkan menu akademik
Memilih sub menu akademik	Menampilkan tabel <i>CRUD</i> tahun akademik
Menambahkan tahun akademik dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data tahun akademik baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data tahun akademik baru ke basis data dan menampilkan tahun akademik pada halaman <i>settings</i> akademik yang bertujuan untuk

	mengaktifkan tahun akademik yang ingin dipilih
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Jadwal Mengaji (Admin)

Skenario:

Tabel 3. 11 Skenario *Use Case* Mengelola Jadwal Mengaji

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu jadwal mengaji	Menampilkan menu jadwal mengaji
Memilih sub menu mengaji	Menampilkan tabel <i>CRUD</i> jadwal mengaji
Menambahkan jadwal mengaji dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data jadwal mengaji baru
Menekan tombol “Simpan”	Menyimpan data jadwal mengaji baru ke basis data dan menampilkan jadwal tersebut pada halaman guru sesuai akun
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu jadwal mengaji	Menampilkan menu jadwal mengaji
Memilih sub menu jadwal mengaji	Menampilkan tabel <i>CRUD</i> jadwal mengaji
Menambahkan jadwal mengaji dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data jadwal mengaji baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data jadwal mengaji baru ke basis data dan menampilkan jadwal tersebut pada halaman guru sesuai akun
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Kelas (Guru)

Skenario:

Tabel 3. 12 Skenario *Use Case* Mengelola Kelas

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu kelas	Menampilkan menu kelas
Memilih sub menu kelas	Menampilkan tabel <i>CRUD</i> kelas
Menambahkan kelas dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data kelas baru
Menekan tombol “Simpan”	Menyimpan data kelas baru ke basis data
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu kelas	Menampilkan menu kelas
Memilih sub menu kelas	Menampilkan tabel <i>CRUD</i> kelas
Menambahkan kelas dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data kelas baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data kelas baru ke basis data
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Materi (Guru)

Skenario:

Tabel 3. 13 Skenario *Use Case* Mengelola Materi

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu materi	Menampilkan menu materi
Memilih sub menu materi	Menampilkan tabel <i>CRUD</i> materi
Menambahkan materi dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data materi baru
Menekan tombol “Simpan”	Menyimpan data materi baru ke basis data
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu materi	Menampilkan menu materi
Memilih sub menu materi	Menampilkan tabel <i>CRUD</i> materi

Menambahkan materi dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data materi baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data materi baru ke basis data
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Nilai Santri (Guru)

Skenario:

Tabel 3. 14 Skenario *Use Case* Menambahkan Nilai Santri

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu nilai	Menampilkan menu nilai santri
Memilih sub menu nilai	Menampilkan tabel <i>CRUD</i> nilai santri
Menambahkan nilai santri dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data nilai santri baru
Menekan tombol “Simpan”	Menyimpan data nilai santri baru ke basis data, nilai yang telah dikelola oleh guru akan masuk ke data rapor, dan menampilkan nilai tersebut ke halaman santri sesuai akun
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu nilai	Menampilkan menu nilai santri
Memilih sub menu akademik	Menampilkan tabel <i>CRUD</i> nilai santri
Menambahkan nilai santri dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data nilai santri baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data nilai santri baru ke basis data, nilai yang telah dikelola oleh guru akan masuk ke data rapor, dan menampilkan nilai tersebut ke halaman santri sesuai akun
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Daftar Kehadiran (Guru)

Skenario:

Tabel 3. 15 Skenario *Use Case* Mengelola Daftar Kehadiran

Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu absensi	Menampilkan menu absensi
Memilih sub menu absensi	Menampilkan tabel <i>CRUD</i> absensi
Menambahkan absensi dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data absensi baru
Menekan tombol “Simpan”	Menyimpan data absensi baru ke basis data
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu absensi	Menampilkan menu absensi
Memilih sub menu absensi	Menampilkan tabel <i>CRUD</i> absensi
Menambahkan pengumuman dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data absensi baru
	Menampilkan pesan bahwa ada <i>field</i> yang belum diisi dan harus diisi
Mengisi <i>field</i> yang kosong dan menekan tombol “Simpan”	Menyimpan data absensi baru ke basis data
	Menampilkan pesan bahwa data sukses ditambahkan

Nama *Use Case*: Mengelola Rapor (Guru)

Skenario:

Tabel 3. 16 Skenario *Use Case* Mengelola Data Rapor

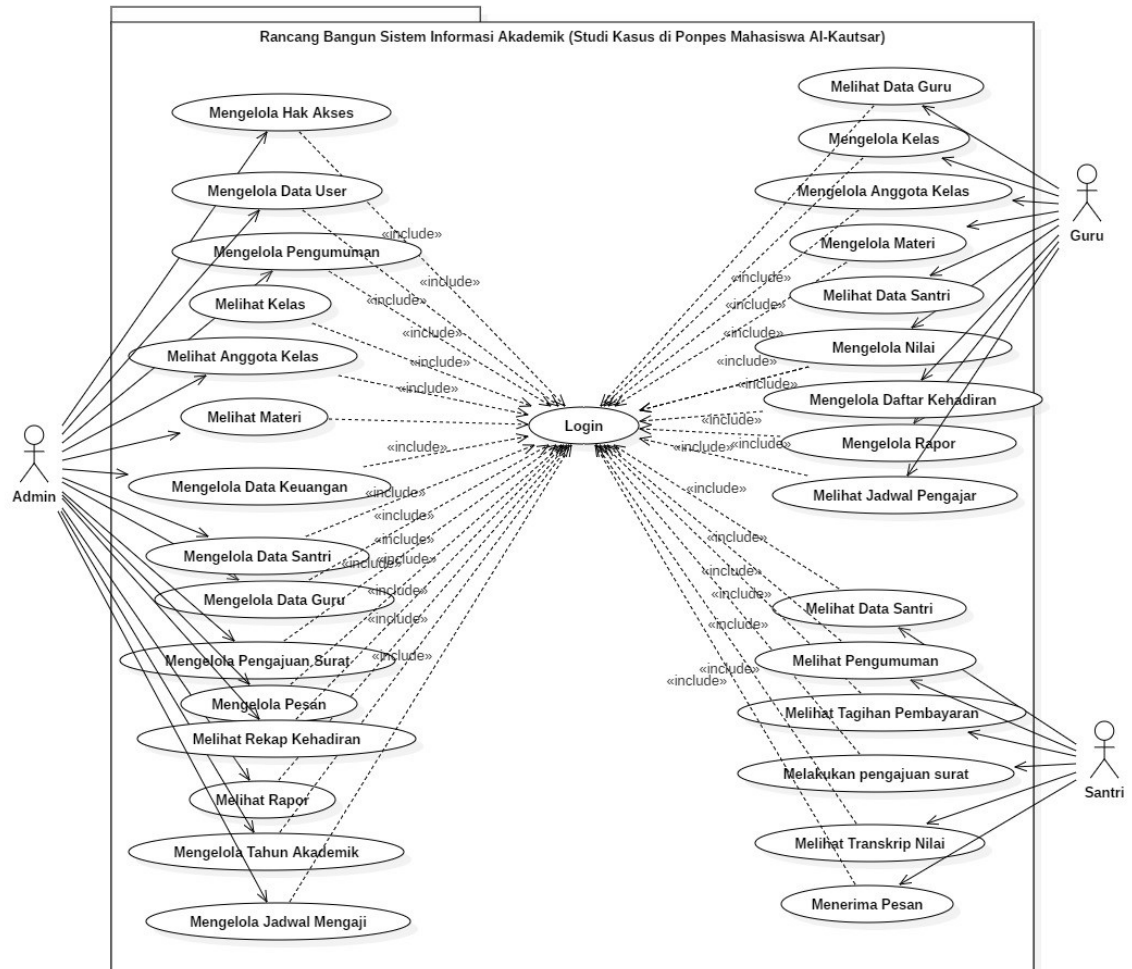
Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu rapor	Menampilkan menu rapor
Memilih sub menu rapor	Menampilkan daftar nama santri yang telah dikelola nilainya
Menekan tombol “Tambah” pada menu rapor	Menampilkan modal <i>form</i> untuk pengisian data nama santri yang sudah selesai dikelola nilainya
Menekan tombol “Rapor”	Melihat apakah data nilai yang dikelola sudah sesuai

Nama *Use Case*: Pengajuan Surat Santri (Santri)

Skenario:

Tabel 3. 17 Skenario *Use Case* Pengajuan Surat Santri

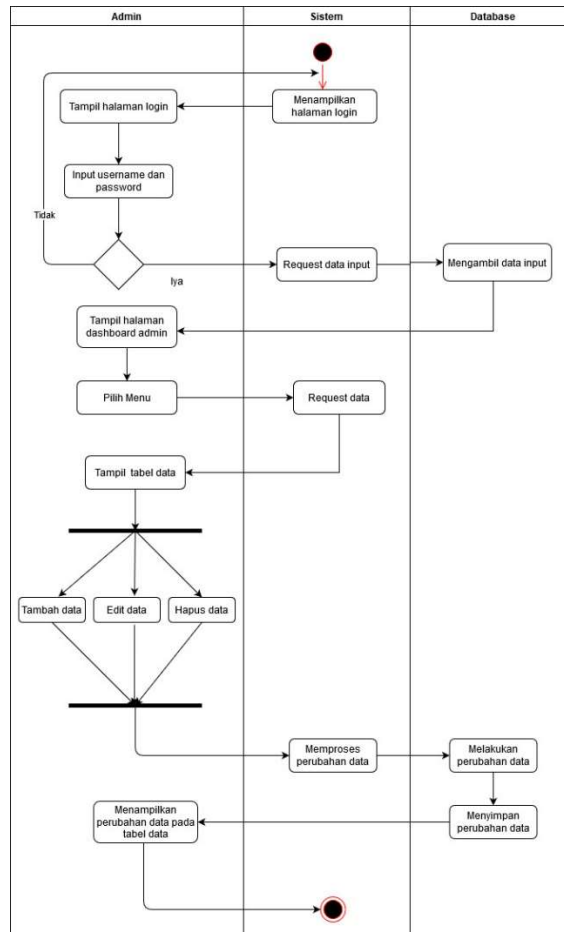
Aksi Aktor	Reaksi Sistem
Skenario Normal	
	Memeriksa status <i>login</i>
Memilih menu pengajuan surat	Menampilkan menu pengajuan surat
Memilih sub menu pengajuan surat	Menampilkan tabel <i>CRUD</i> pengajuan surat
Menambahkan pengajuan surat dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk memilih jenis surat yang ingin diajukan
Menekan tombol “Proses Permintaan”	Menyimpan data pengajuan surat baru ke basis data dan harus menunggu <i>feedback</i> dari admin dalam pengajuan surat tersebut
Skenario Alternatif	
	Memeriksa status <i>login</i>
Memilih menu pengajuan surat	Menampilkan menu pengajuan surat
Memilih sub menu pengajuan surat	Menampilkan tabel <i>CRUD</i> pengajuan surat
Menambahkan nilai santri dengan menekan tombol “Tambah”	Menampilkan modal <i>form</i> untuk pengisian data nilai santri baru
	Menampilkan pesan bahwa ada <i>field</i> yang harus memilih jenis surat
Memilih <i>field</i> jenis surat yang harus dipilih dan menekan tombol “Proses Permintaan”	Menyimpan data pengajuan surat baru ke basis data dan harus menunggu <i>feedback</i> dari admin dalam pengajuan surat tersebut
	Menampilkan pesan bahwa data sukses ditambahkan

d) Diagram *Use Case*Gambar 3. 11 Diagram *Use Case*

Pada *use case* ini kami menggambarkan hubungan antara pengguna atau aktor dengan sistem secara keseluruhan.

3.2.3. Activity Diagram

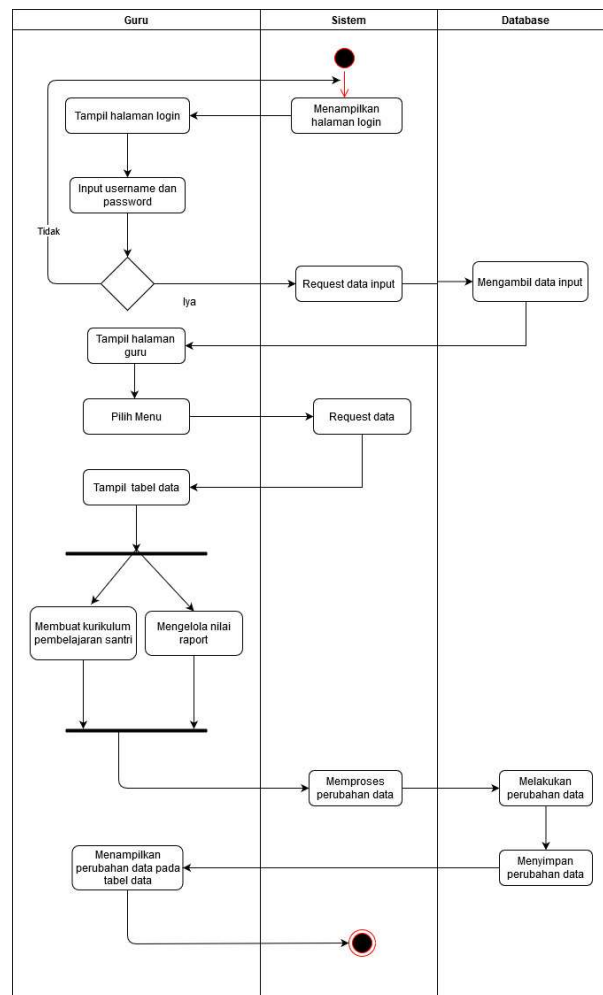
Activity diagram digunakan untuk menggambarkan rangkaian aliran aktivitas baik proses bisnis maupun *use case*. *Activity* diagram dapat juga digunakan untuk memodelkan *action* yang akan dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari *action* tersebut. (Vilela, 2013).



Gambar 3. 12 Activity Diagram Admin

Pada *activity* diagram ini menunjukkan alur aktifitas antara aktor admin dan sistem pada segala proses *create*, *read*, *update*, dan *delete*. Setelah *login* pada sistem, aktor memilih menu yang di inginkan, kemudian sistem akan merequest data dengan *controller* dan model ke *database* dan menampilkan data ke layar aktor menggunakan *view*. Setelah data tampil, aktor dapat melihat data dan memilih aksi apa yang di inginkan pada data tersebut tambah, ubah, atau hapus. Untuk tambah, sistem akan menampilkan *form* untuk

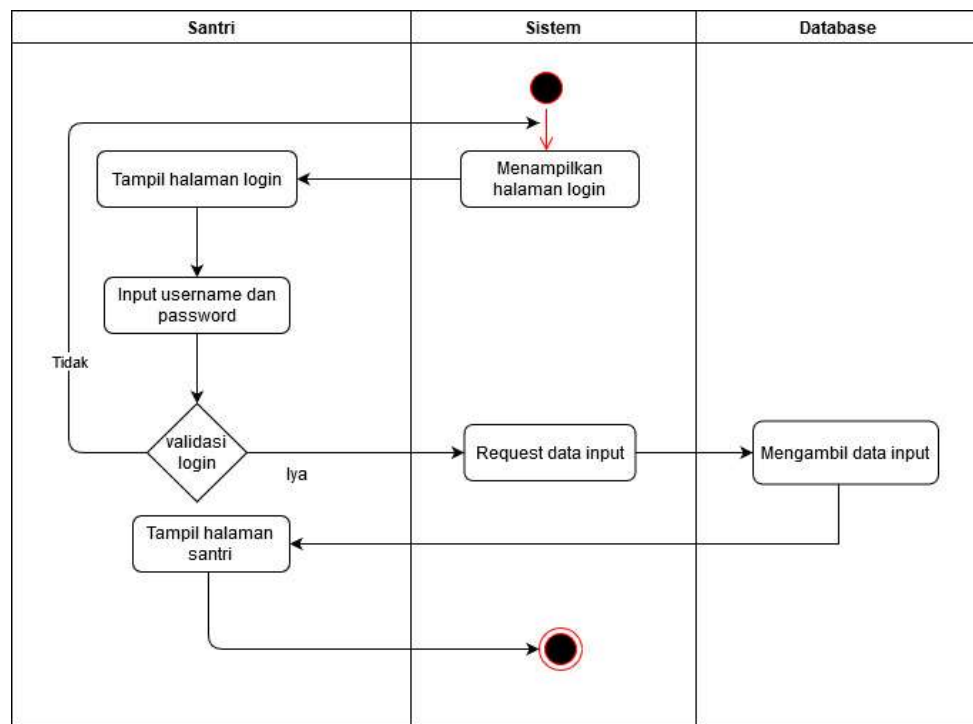
mengisi data, untuk *edit*, sistem akan menampilkan *form* untuk mengubah data, untuk menghapus data, sistem akan menampilkan *confirm box*, sesuai dengan pilihan aktor, aktor di haruskan untuk mengisi data pada *form* dan memilih tombol *submit*, atau memilih *yes* pada *confirm box*. Setelah itu sistem akan mengeksekusi *SQL* ke tabel *database*, dan tabel *database* akan berubah sesuai dengan aksi aktor.



Gambar 3. 13 Activity Diagram Guru

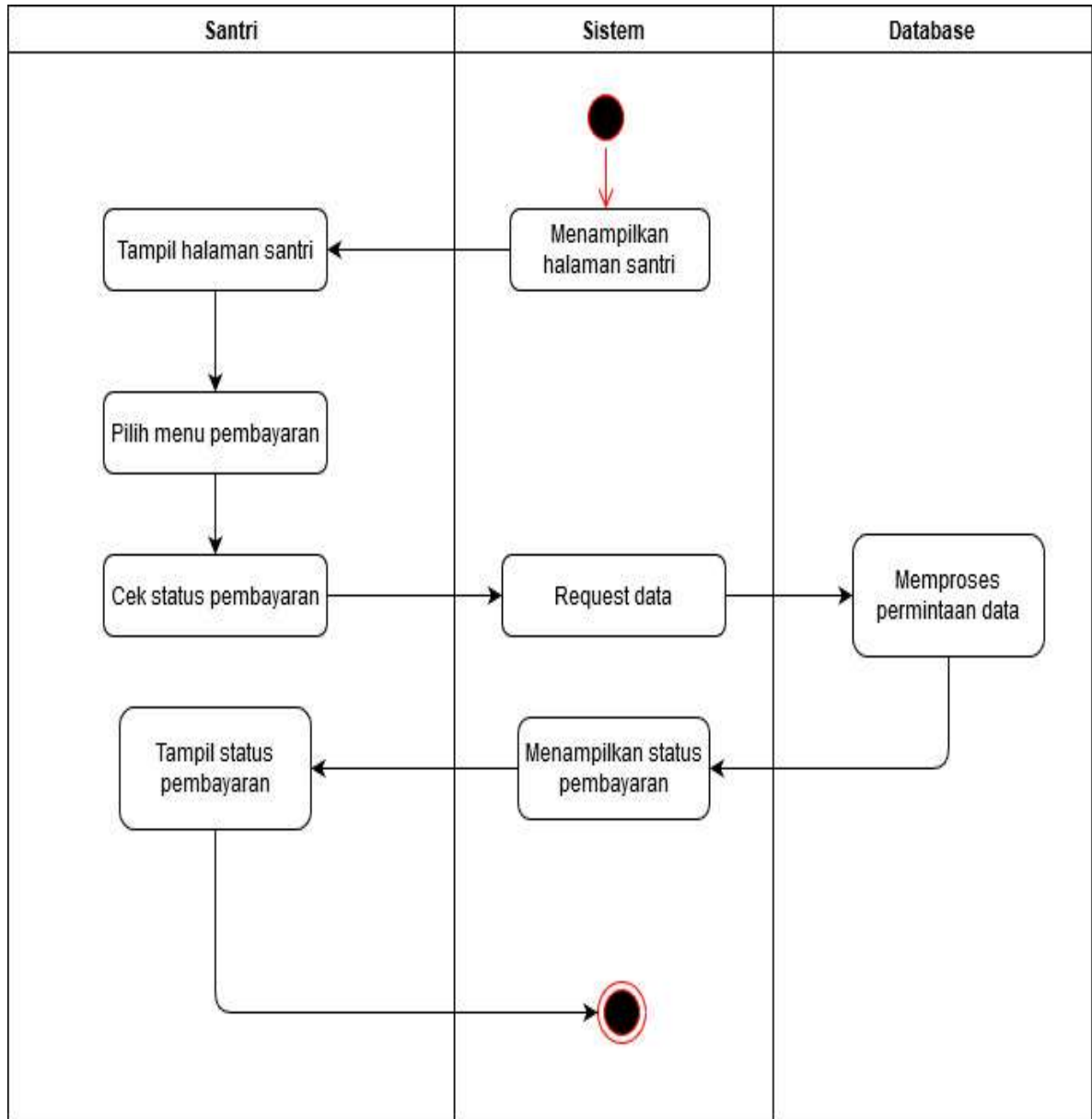
Pada *activity* diagram ini menunjukkan alur aktifitas antara aktor guru dan sistem. Setelah *login* pada sistem, sistem akan melakukan verifikasi, apakah sudah sesuai datanya, jika sudah maka oleh sistem akan disimpan, jika salah maka akan dilakukan verifikasi ulang. Setelah itu, aktor guru juga bisa menentukan materi pelajaran yang akan dikaji di

Pondok Pesantren Mahasiswa Al-Kautsar Kota Malang sesuai dengan kurikulum yang ada. Lalu aktor guru dapat melakukan entri data nilai santri, maka oleh sistem akan dilakukan verifikasi pengiriman data nilai apakah sudah sesuai, jika sudah maka oleh sistem akan diarahkan untuk mencetak nilai ke leger, jika dalam verifikasi gagal maka akan dilakukan pengisian entri nilai lagi sampai data sesuai. Aktor guru juga bertugas untuk membuat rapor hasil belajar santri. Setelah itu sistem akan mengeksekusi *SQL* ke tabel *database*, dan tabel *database* akan berubah sesuai dengan aksi aktor.



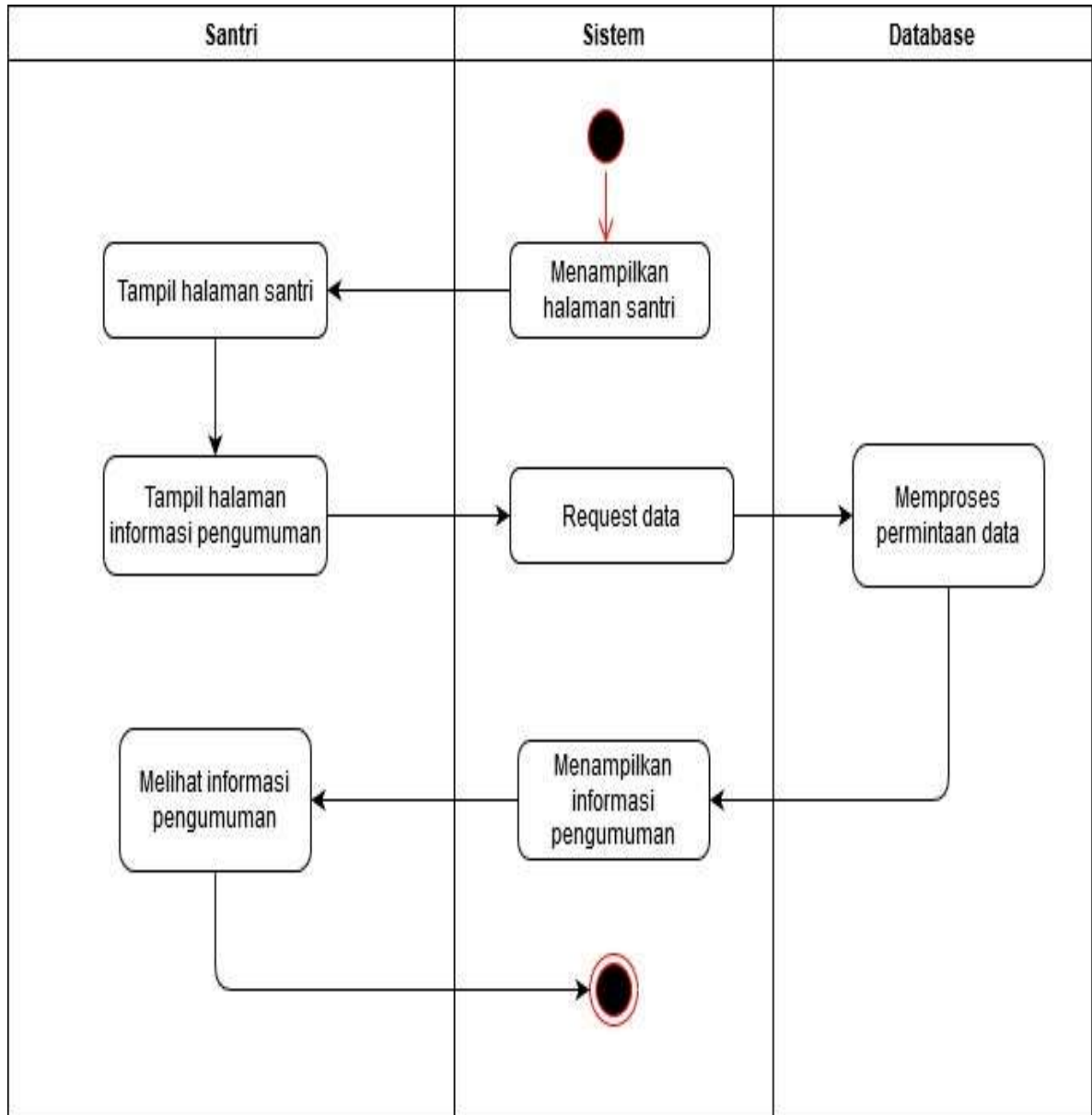
Gambar 3. 14 Activity Diagram Login

Pada *activity* diagram *login* ini digunakan untuk masuk ke halaman *website* dan menampilkan halaman santri.



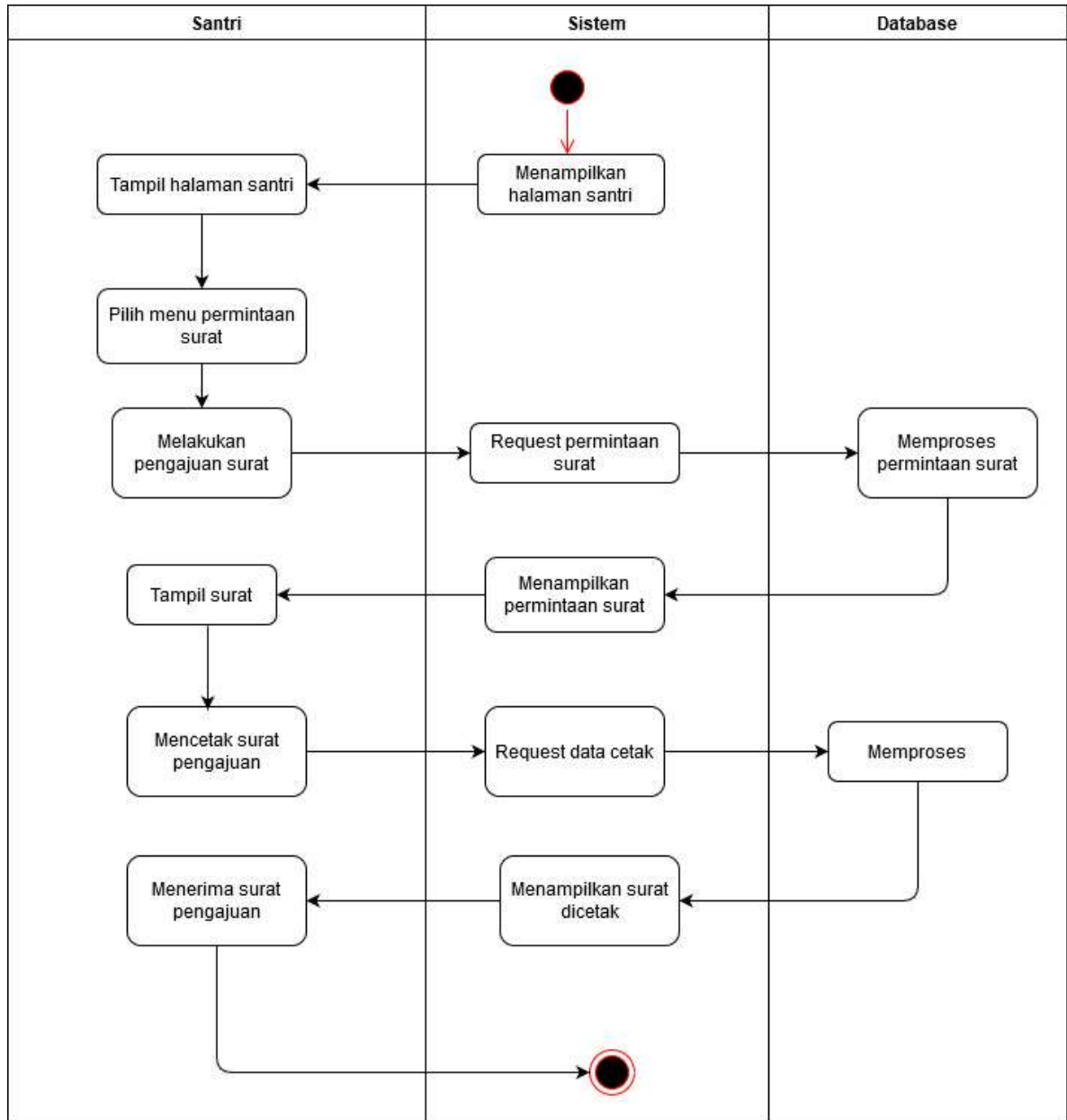
Gambar 3. 15 *Activity* Diagram Melihat Pembayaran

Pada *activity* diagram ini digunakan untuk melihat status pembayaran yang dilakukan oleh aktor santri apakah sudah membayar atau belum.



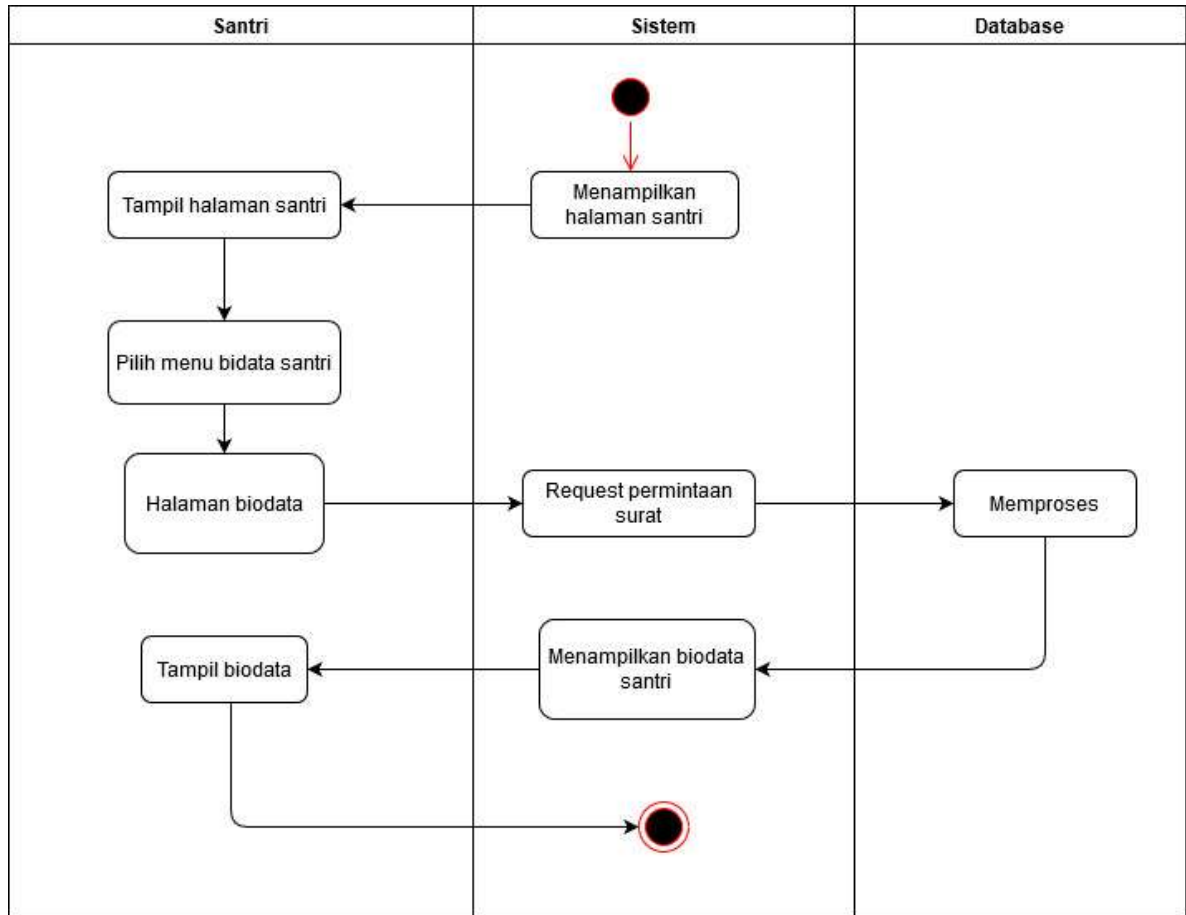
Gambar 3. 16 *Activity* Diagram Melihat Pengumuman

Pada *activity* diagram ini digunakan untuk melihat pengumuman yang dilakukan oleh aktor santri dan menampilkannya pada halaman awal setelah *login*.



Gambar 3. 17 *Activity* Diagram Mengajukan Surat

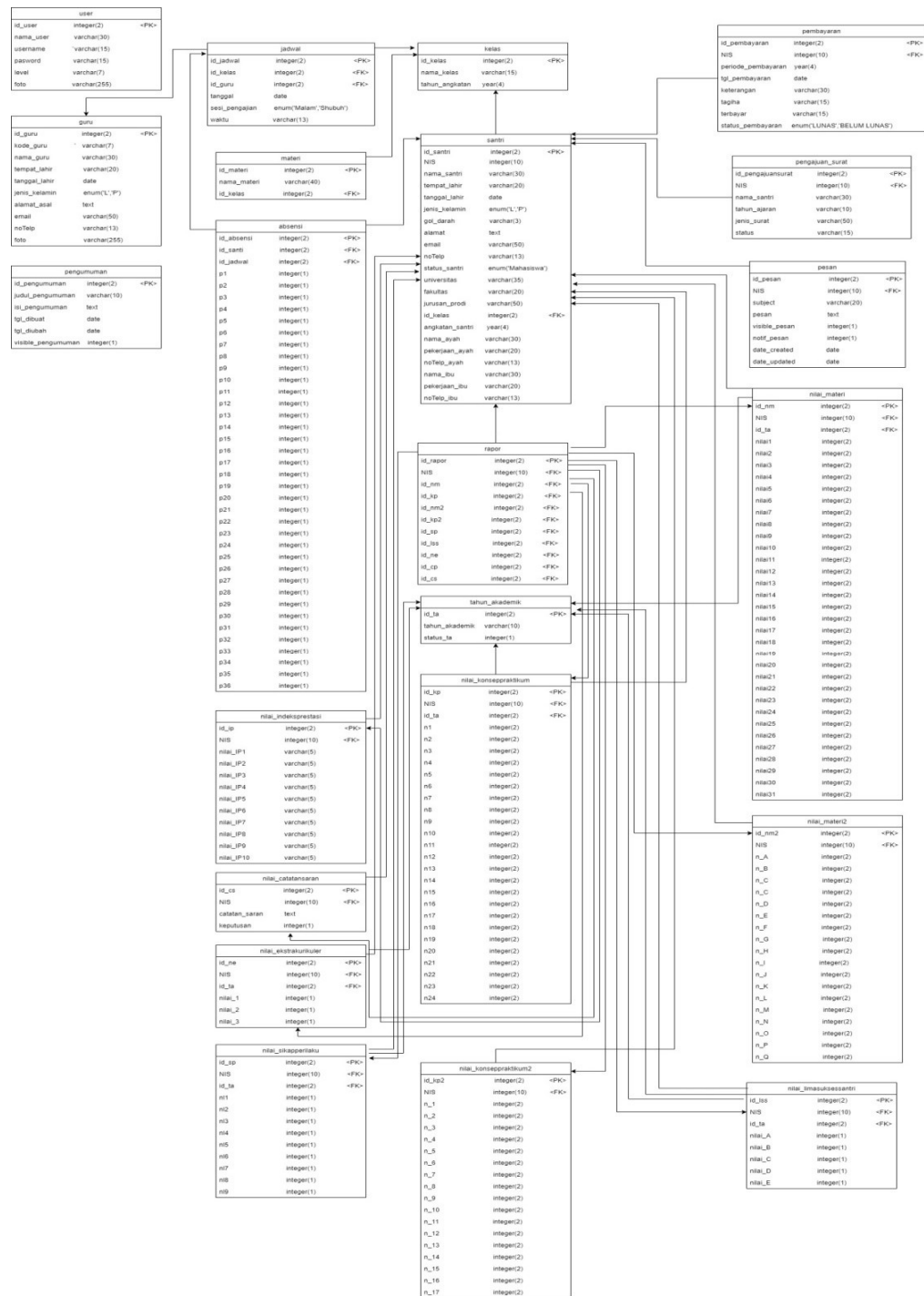
Pada *activity* diagram ini digunakan untuk mengajukan surat yang dilakukan oleh aktor santri dan santri juga dapat memilih jenis surat.



Gambar 3. 18 *Acitivity* Diagram Melihat Biodata

Pada *activity* diagram ini digunakan untuk melihat biodata yang dilakukan oleh aktor santri dan menampilkan biodata pada halaman santri.

3.2.4. Class Diagram



Gambar 3. 19 Class Diagram

Pada *class* diagram ini adalah umum bagi sistem, yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

a) Kamus Data

Sebelum membuat *database* pada *phpmyadmin* dibutuhkan kamus data agar mengetahui tabel beserta kolom yang dibutuhkan menurut rancangan sistem. Kamus data merupakan penjelasan tertulis atau gambaran suatu data secara fisik yang digunakan pada *database* sistem yang dibuat. Kamus data ini dibuat berdasarkan *class diagram* yang digunakan pada sistem. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, merancang laporan-laporan dan *database* (Hartono, 2001). Tabel-tabel yang dibuat adalah sebagai berikut.

- Tabel *User*

Isi: Data *User*

Primary key: *id_user*

Tabel 3. 18 Tabel *User*

Nama Field	Tipe Data	Keterangan
<i>id_user</i>	int(2)	PK
<i>nama_user</i>	varchar(30)	
<i>username</i>	varchar(15)	
<i>password</i>	varchar(15)	
<i>level</i>	varchar(7)	
<i>foto</i>	varchar(255)	

Pada tabel *user* digunakan untuk menambahkan *user*, dimana data yang ditampilkan adalah data dari tiap *user* yang dapat mengakses sistem. Data yang disajikan berupa nama, *username*, *password*, *level* masing-masing *user*, dan foto.

- Tabel Santri
Isi: Data Santri
Primary key: id_santri
Foreign key: id_kelas

Tabel 3. 19 Tabel Santri

Nama Field	Tipe Data	Keterangan
id_santri	int(2)	PK
NIS	varchar(10)	
nama_santri	varchar(30)	
tempat_lahir	varchar(20)	
tanggal_lahir	date	
jenis_kelamin	enum('L', 'P')	
gol_darah	varchar(3)	
alamat	text	
email	varchar(50)	
noTelp	varchar(13)	
status_santri	enum('Mahasiswa')	
universitas	varchar(35)	
fakultas	varchar(20)	
jurusan_prodi	varchar(50)	
id_kelas	int(2)	FK
angkatan_santri	year(4)	
nama_ayah	varchar(30)	
pekerjaan_ayah	varchar(20)	
noTelp_ayah	varchar(13)	
nama_ibu	varchar(30)	
pekerjaan_ibu	varchar(20)	
noTelp_ibu	varchar(13)	

Pada tabel santri digunakan untuk menyimpan data-data santri selama mondok di Pondok Pesantren Al-Kautsar Kota Malang. Di dalam tabel santri terdapat 22 kolom.

- Tabel Guru
Isi: Data Guru
Primary key: id_guru

Tabel 3. 20 Tabel Guru

Nama Field	Tipe Data	Keterangan
id_guru	int(2)	PK
kode_guru	varchar(7)	
nama_guru	varchar(30)	
tempat_lahir	varchar(20)	
tanggal_lahir	date	
jenis_kelamin	enum('L', 'P')	
alamat_asal	text	
email	varchar(50)	
noTelp	varchar(13)	
foto	varchar(255)	

Pada tabel guru digunakan untuk menyimpan data-data guru yang ada di Pondok Pesantren Al-Kautsar Kota Malang. Di dalam tabel guru terdapat 10 kolom.

- Tabel Kelas
Isi: Data Kelas
Primary key: id_kelas

Tabel 3. 21 Tabel Kelas

Nama Field	Tipe Data	Keterangan
id_kelas	int(2)	PK
kelas	varchar(15)	
tahun_angkatan	year(4)	

Pada tabel kelas digunakan untuk keterangan siswa berada di kelas apa.

- Tabel Materi
Isi: Data Materi
Primary key: id_materi
Foreign key: id_kelas

Tabel 3. 22 Tabel Materi

Nama Field	Tipe Data	Keterangan
id_materi	int(2)	PK
nama_materi	varchar(40)	
id_kelas	int(2)	FK

Pada tabel materi digunakan untuk menyimpan data materi untuk dimasukkan ke dalam rapor untuk dicetak.

- Tabel Absensi
Isi: Data Absensi
Primary key: id_absensi
Foreign key: id_santri
Foreign key: id_jadwal

Tabel 3. 23 Tabel Absensi

Nama Field	Tipe Data	Keterangan
id_absensi	int(2)	PK
id_santri	int(2)	FK
id_jadwal	int(2)	FK
p1	int(1)	
p2	int(1)	
p3	int(1)	
p4	int(1)	
p5	int(1)	
p6	int(1)	
p7	int(1)	
p8	int(1)	

p9	int(1)	
p10	int(1)	
p11	int(1)	
p12	int(1)	
p13	int(1)	
p14	int(1)	
p15	int(1)	
p16	int(1)	
p17	int(1)	
p18	int(1)	
p19	int(1)	
p20	int(1)	
p21	int(1)	
p22	int(1)	
p23	int(1)	
p24	int(1)	
p25	int(1)	
p26	int(1)	
p27	int(1)	
p28	int(1)	
p29	int(1)	
p30	int(1)	
p31	int(1)	
p32	int(1)	
p33	int(1)	
p34	int(1)	
p35	int(1)	
p36	int(1)	

Pada tabel absensi digunakan sebagai penyimpan data absensi santri dalam sehari-hari yang akan diinputkan oleh guru nantinya. Status yang ada di dalam tabel absensi adalah status kehadiran santri sakit, izin, alpha, dan hadir. Nantinya perhitungan absensi hanya dari status sakit, alpha, dan izin saja.

- Tabel Jadwal
Isi: Data Jadwal
Primary key: id_jadwal
Foreign key: id_kelas
Foreign key: id_guru

Tabel 3. 24 Tabel Jadwal

Nama Field	Tipe Data	Keterangan
id_jadwal	int(2)	PK
id_kelas	int(2)	FK
id_guru	int(2)	FK
tanggal	date	
sesi_pengajian	enum('Malam', 'Shubuh')	
waktu	varchar(13)	

Pada tabel jadwal digunakan untuk menyimpan data guru mengajar mata pelajaran apa saja.

- Tabel Pengumuman
Isi: Data Pengumuman
Primary key: id_pengumuman

Tabel 3. 25 Tabel Pengumuman

Nama Field	Tipe Data	Keterangan
id_pengumuman	int(2)	PK
judul_pengumuman	varchar(10)	
isi_pengumuman	text	
tgl_dibuat	date	
tgl_diubah	date	

visible_pengumuman	int(1)	
--------------------	--------	--

Pada tabel pengumuman digunakan untuk menyimpan data pengumuman kemudian ditampilkan pada halaman santri.

- Tabel Pembayaran
Isi: Data Pembayaran
Primary key: id_pembayaran
Foreign key: NIS

Tabel 3. 26 Tabel Pembayaran

Nama Field	Tipe Data	Keterangan
id_pembayaran	int(2)	PK
NIS	varchar(10)	FK
periode_pembayaran	year(4)	
tgl_pembayaran	date	
keterangan	varchar(30)	
tagihan	varchar(15)	
terbayar	varchar(15)	
status_pembayaran	enum('LUNAS','BELUM LUNAS')	

Pada tabel pembayaran digunakan untuk menyimpan data pembayaran kemudian ditampilkan pada halaman santri sesuai akun.

- Tabel Pengajuan Surat
Isi: Data Pengajuan Surat
Primary key: id_pengajuansurat
Foreign key: NIS

Tabel 3. 27 Tabel Pengajuan Surat

Nama Field	Tipe Data	Keterangan
id_pengajuansurat	int(2)	PK
NIS	int(10)	FK
nama_santri	varchar(30)	

tahun_ajaran	varchar(10)	
jenis_surat	varchar(50)	
status	varchar(15)	

Pada tabel pengajuan surat digunakan untuk menyimpan data pengajuan surat yang telah diajukan oleh santri.

- Tabel Pesan

Isi: Data Pesan

Primary key: id_pesan

Foreign key: NIS

Tabel 3. 28 Tabel Pesan

Nama Field	Tipe Data	Keterangan
id_pesan	int(2)	PK
NIS	int(10)	FK
subject	varchar(20)	
pesan	text	
visible_pesan	int(1)	
notif_pesan	int(1)	
date_created	date	
date_updated	date	

Pada tabel pesan digunakan untuk menyimpan data pesan, yang nantinya pesan tersebut akan terkirim pada santri.

- Tabel Tahun Akademik

Isi: Data Tahun Akademik

Primary key: id_ta

Tabel 3. 29 Tabel Tahun Akademik

Nama Field	Tipe Data	Keterangan
id_ta	int(2)	PK
tahun_akademik	varchar(10)	

status_ta	int(1)	
-----------	--------	--

Pada tabel tahun akademik digunakan untuk menyimpan data tahun akademik, yang nantinya tahun akademik bisa di *settings* untuk diaktifkan dan dinonaktifkan.

- Tabel Nilai Materi
Isi: Data Nilai Materi
Primary key: id_nm
Foreign key: NIS
Foreign key: id_ta

Tabel 3. 30 Tabel Nilai Materi

Nama Field	Tipe Data	Keterangan
id_nm	int(2)	PK
NIS	int(10)	FK
id_ta	int(2)	FK
nilai1	int(2)	
nilai2	int(2)	
nilai3	int(2)	
nilai4	int(2)	
nilai5	int(2)	
nilai6	int(2)	
nilai7	int(2)	
nilai8	int(2)	
nilai9	int(2)	
nilai10	int(2)	
nilai11	int(2)	
nilai12	int(2)	
nilai13	int(2)	
nilai14	int(2)	
nilai15	int(2)	

nilai16	int(2)	
nilai17	int(2)	
nilai18	int(2)	
nilai19	int(2)	
nilai20	int(2)	
nilai21	int(2)	
nilai22	int(2)	
nilai23	int(2)	
nilai24	int(2)	
nilai25	int(2)	
nilai26	int(2)	
nilai27	int(2)	
nilai28	int(2)	
nilai29	int(2)	
nilai30	int(2)	
nilai31	int(2)	

Pada tabel nilai materi digunakan untuk menyimpan nilai materi santri yang berada pada kelas reguler yang nantinya akan dimasukkan ke rapor dan dicetak.

- Tabel Nilai Materi 2
Isi: Data Nilai Materi 2
Primary key: id_nm2
Foreign key: NIS

Tabel 3. 31 Tabel Nilai Materi 2

Nama Field	Type Data	Keterangan
id_nm2	int(2)	PK
NIS	int(10)	FK
n_A	int(2)	
n_B	int(2)	

n_C	int(2)	
n_D	int(2)	
n_E	int(2)	
n_F	int(2)	
n_G	int(2)	
n_H	int(2)	
n_I	int(2)	
n_J	int(2)	
n_K	int(2)	
n_L	int(2)	
n_M	int(2)	
n_N	int(2)	
n_O	int(2)	
n_P	int(2)	
n_Q	int(2)	

Pada tabel nilai materi 2 digunakan untuk menyimpan nilai materi santri yang berada pada kelas hadits besar yang nantinya akan dimasukkan ke rapor dan dicetak.

- Tabel Nilai Konsep Praktikum
Isi: Data Nilai Konsep Praktikum
Primary key: id_kp
Foreign key: NIS
Foreign key: id_ta

Tabel 3. 32 Tabel Nilai Konsep Praktikum

Nama Field	Tipe Data	Keterangan
id_kp	int(2)	PK
NIS	int(10)	FK
id_ta	int(2)	FK
n1	int(2)	

n2	int(2)	
n3	int(2)	
n4	int(2)	
n5	int(2)	
n6	int(2)	
n7	int(2)	
n8	int(2)	
n9	int(2)	
n10	int(2)	
n11	int(2)	
n12	int(2)	
n13	int(2)	
n14	int(2)	
n15	int(2)	
n16	int(2)	
n17	int(2)	
n18	int(2)	
n19	int(2)	
n20	int(2)	
n21	int(2)	
n22	int(2)	
n23	int(2)	
n24	int(2)	

Pada tabel nilai konsep praktikum digunakan untuk menyimpan nilai praktikum santri yang berada pada kelas reguler yang nantinya akan dimasukkan ke rapor dan dicetak.

- Tabel Nilai Konsep Praktikum 2
Isi: Data Nilai Konsep Praktikum 2
Primary key: id_kp2
Foreign key: NIS

Tabel 3. 33 Tabel Nilai Konsep Praktikum 2

Nama <i>Field</i>	Tipe Data	Keterangan
id_kp2	int(2)	PK
NIS	int(10)	FK
n_1	int(2)	
n_2	int(2)	
n_3	int(2)	
n_4	int(2)	
n_5	int(2)	
n_6	int(2)	
n_7	int(2)	
n_8	int(2)	
n_9	int(2)	
n_10	int(2)	
n_11	int(2)	
n_12	int(2)	
n_13	int(2)	
n_14	int(2)	
n_15	int(2)	
n_16	int(2)	
n_17	int(2)	

Pada tabel nilai konsep praktikum 2 digunakan untuk menyimpan nilai praktikum santri yang berada pada hadits besar yang nantinya akan dimasukkan ke rapor dan dicetak.

- Tabel Nilai Sikap Perilaku
Isi: Data Nilai Sikap Perilaku
Primary key: id_sp
Foreign key: NIS
Foreign key: id_ta

Tabel 3. 34 Tabel Nilai Sikap Perilaku

Nama Field	Tipe Data	Keterangan
id_sp	int(2)	PK
NIS	int(10)	FK
id_ta	int(2)	FK
n11	int(1)	
n12	int(1)	
n13	int(1)	
n14	int(1)	
n15	int(1)	
n16	int(1)	
n17	int(1)	
n18	int(1)	
n19	int(1)	

Pada tabel nilai sikap perilaku digunakan untuk menyimpan nilai sikap perilaku santri untuk dimasukkan ke rapor dan dicetak.

- Tabel Nilai Lima Sukses Santri
Isi: Data Nilai Lima Sukses Santri
Primary key: id_iss
Foreign key: NIS
Foreign key: id_ta

Tabel 3. 35 Tabel Nilai Lima Sukses Santri

Nama Field	Tipe Data	Keterangan
id_iss	int(2)	PK

NIS	int(10)	FK
id_ta	int(2)	FK
nilai_A	int(1)	
nilai_B	int(1)	
nilai_C	int(1)	
nilai_D	int(1)	
nilai_E	int(1)	

Pada tabel nilai lima sukses santri digunakan untuk menyimpan nilai lima sukses santri untuk dimasukkan ke rapor dan dicetak.

- Tabel Nilai Ekstrakurikuler
Isi: Data Nilai Ekstrakurikuler
Primary key: id_ne
Foreign key: NIS
Foreign key: id_ta

Tabel 3. 36 Tabel Nilai Ekstrakurikuler

Nama Field	Tipe Data	Keterangan
id_ne	int(2)	PK
NIS	int(10)	FK
id_ta	int(2)	FK
nilai_1	int(1)	
nilai_2	int(1)	
nilai_3	int(1)	

Pada tabel nilai ekstrakurikuler digunakan untuk menyimpan nilai ekstrakurikuler santri untuk dimasukkan ke rapor dan dicetak.

- Tabel Nilai Indeks Prestasi

Isi: Data Nilai Indeks Prestasi

Primary key: id_ip

Foreign key: NIS

Tabel 3. 37 Tabel Nilai Indeks Prestasi

Nama Field	Tipe Data	Keterangan
id_ip	int(2)	PK
NIS	int(10)	FK
nilai_IP1	varchar(5)	
nilai_IP2	varchar(5)	
nilai_IP3	varchar(5)	
nilai_IP4	varchar(5)	
nilai_IP5	varchar(5)	
nilai_IP6	varchar(5)	
nilai_IP7	varchar(5)	
nilai_IP8	varchar(5)	
nilai_IP9	varchar(5)	
nilai_IP10	varchar(5)	

Pada tabel nilai indeks prestasi digunakan untuk menyimpan nilai indeks prestasi santri untuk dimasukkan ke rapor dan dicetak.

- Tabel Catatan Saran Pengurus

Isi: Data Catatan Saran Pengurus

Primary key: id_cs

Foreign key: NIS

Tabel 3. 38 Tabel Catatan Saran Pengurus

Nama Field	Tipe Data	Keterangan
id_cs	int(2)	PK
NIS	int(10)	FK

catatan_saran	text	
keputusan	int(1)	

Pada tabel catatan saran pengurus digunakan untuk menyimpan data catatan saran pengurus untuk dimasukkan ke rapor dan dicetak.

- Tabel Rapor

Isi: Data Rapor

Primary key: id_rapor

Foreign key: NIS

Foreign key: id_nm

Foreign key: id_nm2

Foreign key: id_kp

Foreign key: id_kp2

Foreign key: id_sp

Foreign key: id_lss

Foreign key: id_ne

Foreign key: id_cp

Foreign key: id_cs

Tabel 3. 39 Tabel Rapor

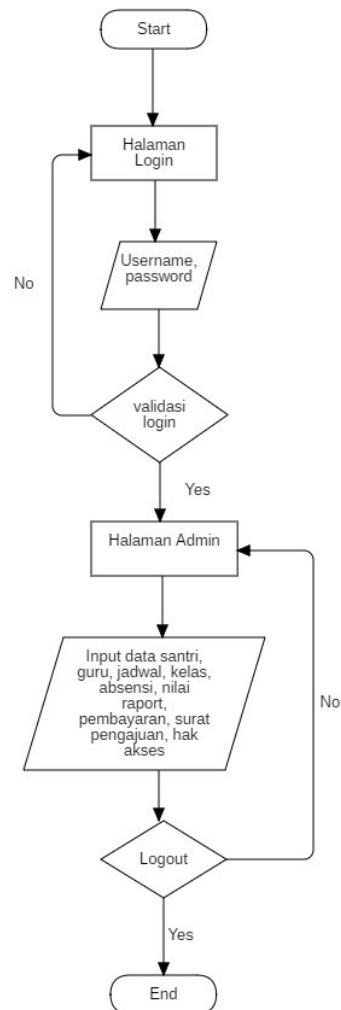
Nama Field	Tipe Data	Keterangan
id_rapor	int(2)	PK
NIS	int(10)	FK
id_nm	int(2)	FK
id_nm2	int(2)	FK
id_kp	int(2)	FK
id_kp2	int(2)	FK
id_sp	int(2)	FK
id_lss	int(2)	FK
id_ne	int(2)	FK

id_cp	int(2)	FK
id_cs	int(2)	FK

Pada tabel rapor digunakan untuk menampilkan semua nilai santri dan juga dapat mencetak.

3.2.5. Diagram Alir

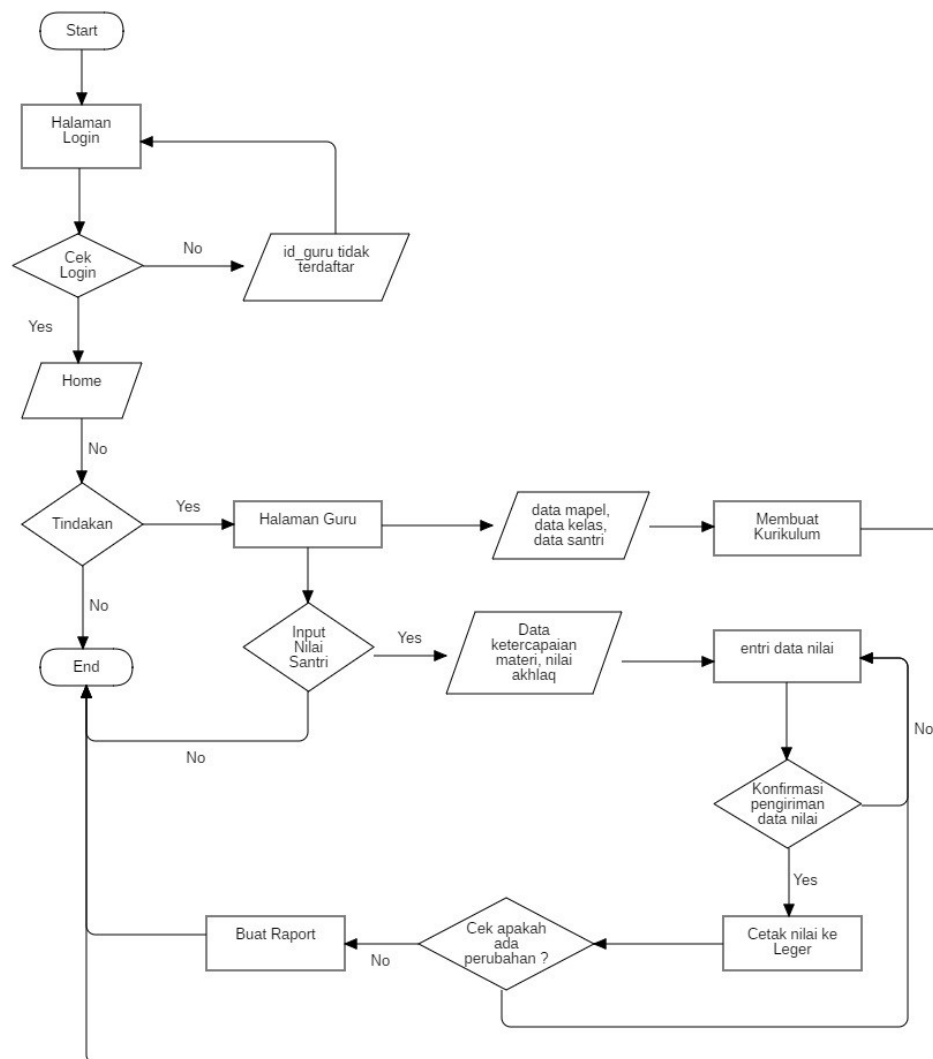
Diagram Alir merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Biasanya mempengaruhi penyelesaian masalah yang khususnya perlu dipelajari dan dievaluasi lebih lanjut. (Indrajani, 2011).



Gambar 3. 20 Diagram Alir Admin

Keterangan:

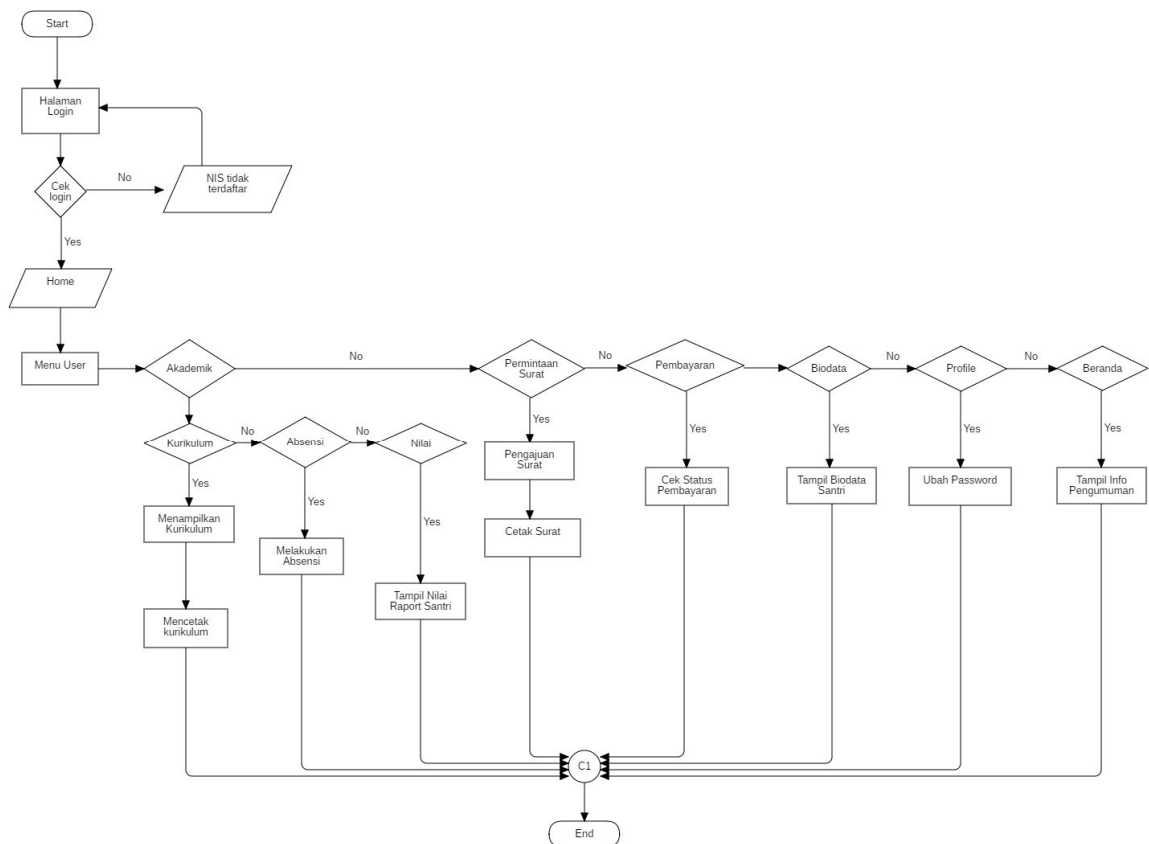
1. Admin masuk ke halaman *login*.
2. Admin akan masuk ke halaman admin jika berhasil *login*, jika gagal maka tidak bisa masuk ke halaman admin.
3. Admin dapat mengelola data pondok yang berhubungan dengan data pengajar, data santri, data nilai santri, data pengumuman santri, data pengajuan surat santri, pembayaran biaya pondok, dan sistem kehadiran kegiatan belajar mengajar.



Gambar 3. 21 Diagram Alir Guru

Keterangan:

1. Guru melakukan *login* terlebih dahulu.
2. Jika tidak bisa masuk ke halaman guru berarti id_guru belum terdaftar.
3. Jika berhasil *login* maka akan masuk ke halaman guru.
4. Guru dapat mengelola data nilai santri, dan rapor.



Gambar 3. 22 Diagram Alir Santri

Keterangan:

1. Santri *login* menggunakan nomer induk santri sebagai *username* dan *password*.
2. Jika santri berhasil *login* maka akan masuk ke tampilan menu.
3. Santri dapat melihat pengumuman, biodata, nilai materi, nilai praktikum, status pembayaran, pesan masuk, dan santri juga dapat mengajukan surat.