

BAB III

METODE PENELITIAN

3.1. Analisis kebutuhan Sistem

Sistem informasi akedemik ini dibangun untuk membantu mempermudah guru maupun orang tua yang utamanya di SMA Negeri 1 Karangnomo dalam mengelola dan melakukan pengawasan terhadap siswa, mengingat fitur-fitur yang akan dibuat agar tidak menyimpang dari konteks latar belakang masalah, serta dalam untuk mencegah pembahasan dan perluasan masalah maka penulis membatasi fitur sistem sebagai berikut:

1. Sistem dibangun menggunakan bahasa pemrograman PHP dan *database* MySQL, berbasis *website*.
2. Sistem yang dibangun memberikan fitur *role user* atau dengan kata lain setiap pengguna dari masing-masing tingkatan *user* diberikan hak akses masing-masing.
3. Sistem dapat melakukan olah data pada data master meliputi:
 - a. Data master mata pelajaran
 - b. Data master kelas
 - c. Data master *role*
 - d. Data master nilai
4. Sistem yang dibuat dapat melakukan olah data pengguna, data diri dan login masing dari pengguna, berikut daftar *user* yang dapat mengakses ke sistem
 - a. Administrator
 - b. Guru
 - c. Wali Siswa
 - d. Siswa

5. Sistem yang dibuat dapat melakukan olah data kelas angkatan, data kelas angkatan berisi data kelas pada tahun ajaran tertentu.
6. Sistem yang dibuat dapat melakukan olah data nilai, mulai dari nilai ulangan harian hingga nilai hasil semester akhir. Yang pada akhirnya nilai dapat dilihat oleh siswa dan orang tua siswa.
7. Sistem dapat mengolah data materi, menyimpan materi yang di *upload* oleh guru, sehingga siswa dapat di *download* oleh siswa.
8. Sistem dapat megolah data absensi perhari, sehingga wali siswa dapat melihat siapa saja siswa yang tidak masuk.
9. Sistem dapat mengolah data pengumuman yang di inputkan oleh guru atau operator, yang nantinya pengumuman tersebut dapat dilihat oleh siswa dan orang tua pada halaman publik *website*.
10. Sistem dapat megolah data galeri, galeri ini berisi foto atau video dokumentasi kegiatan sekolah. Data galeri nantinya akan ditampilkan pada halaman publik *website*.
11. Sistem dapat mengolah berita harian, berita yang terkait dengan perkembangan dunia pendidikan saat ini, sehingga bisa memberikan informasi kepada orang tua dan siswa yang mengakses *website*.
12. Sistem dapat mengolah data profil sekolah, kontak dan halaman visi misi, dan event. Sehingga sistem nantinya akan menjadi lebih informatif.

3.2. Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan adalah *Personal Computer / Laptop* dengan spesifikasi sebagai berikut:

1. Processor 3,5 GHz atau lebih
2. Ram dengan kapasitas 4 Gb atau lebih
3. Space *Hardisk* kapasitas minimal 10 Gb atau lebih.

3.3. Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam pengembangan sistem adalah sebagai berikut:

1. Sistem Operasi Minimal Microsoft Windows 7
2. XAMPP 5.6.3 , PHP 5.6.3
3. Sublime Text Editor
4. Firefox Web Browser Versi 38.0
5. Chrome Web Browser
6. Adobe PhotoShop CS

3.4. Kebutuhan Fungsional

Analisis kebutuhan sistem, menjelaskan apa saja yang dibutuhkan sistem informasi akademik. Analisis ini digunakan dengan tujuan memahami kebutuhan dari penerapan sistem, apakah sistem tersebut sudah sesuai dengan kebutuhan yang diperlukan atau belum, serta apakah sudah sesuai dengan tujuan yang ingin dicapai atau belum.

Dengan pengimplementasian sistem yang akan peneliti buat diharapkan dapat bermanfaat bagi para orang tua siswa guru serta murid dalam memantau dan mengawasi hasil dari kegiatan belajar mengajar:

1. Sistem ini harus dapat merekap nilai siswa, mulai dari nilai ulangan harian maupun semester.
2. Sistem harus dapat menampung data pengumuman dari sekolah terkait dengan kegiatan sekolah, dan pengumuman khusus daftar siswa yang tidak masuk sekolah setiap harinya dengan keterangan maupun tanpa keterangan.
3. Sistem harus dapat memberikan akses khusus kepada operator sistem, guru, orangtua dan siswa didalam aplikasi untuk membedakan hak akses di setiap penggunaanya.

3.5. Kebutuhan Non Fungsional

Kebutuhan non fungsional mengidentifikasi batasan dari fasilitas yang disediakan oleh sistem. Kebutuhan non fungsional meliputi kebutuhan yang diperlukan sistem untuk berjalan dengan baik dari segi operasional, dan kinerja.

1. Operasional

a. Sistem

- *Web hosting minimum space* 5 GB, lebih besar dan lebih bagus, hal ini dimaksudkan agar file materi yang di *upload* ke dalam sistem bisa memiliki *space* yang besar, atau tidak cepat penuh
- *Bandwith unlimited*, lebar akses data ke dalam *hosting* jika memiliki *space* yang besar/*unlimited* maka *hosting* dapat menampung banyak akses data secara bersamaan.
- Mendukung php versi 5.6, karena sistem yang di implemmentasikan menggunakan Laravel 4.1
- Mendukung *database* MySQL
- Semua paket *hosting* merujuk dari www.domainesia.com

b. Pengguna

- Sistem Operasi Microsoft Windows 7, atau Ubuntu 12.04
- Browser Mozilla Firefox 30 keatas, atau Google Chrome
- Koneksi internet Minimum 512 Kbps

2. Keamanan

- a. Sistem aplikasi maupun database dilengkapi *password*
- b. Hak akses untuk setiap penggunanya

3. Informasi

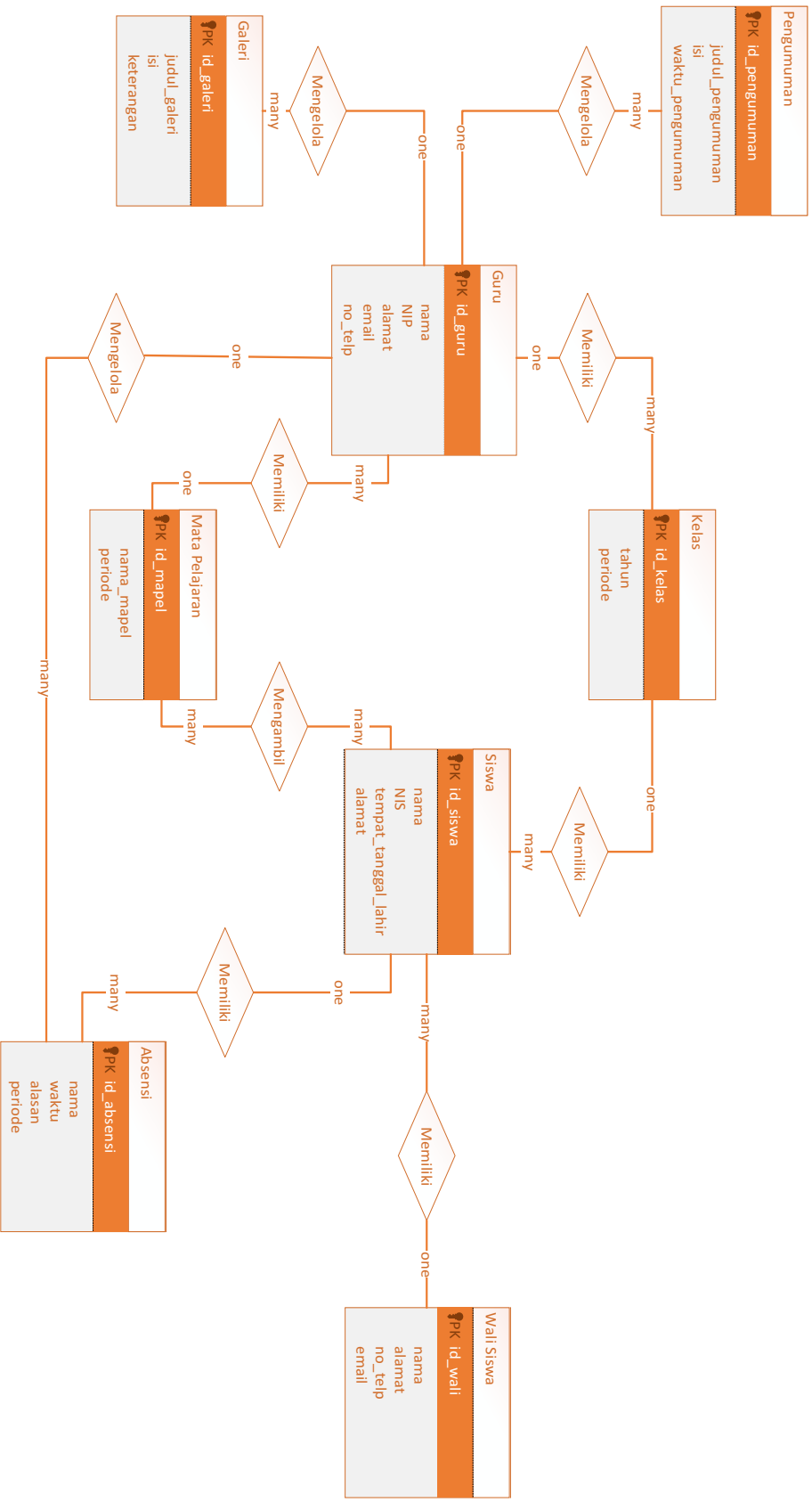
- a. Sistem harus dapat menginformasikan kesalahan ketika *user* memasukan data yang salah ketika *login*
- b. Sistem harus mampu menampilkan rekap nilai yang informatif sehingga mudah diterima oleh pengguna sistem.

3.6. Perancangan Sistem

Perancangan sistem adalah gambaran proses-proses yang terlibat dalam pengembangan sistem. Dalam perancangan sistem informasi akademik SMA N 1 Karangnom Klaten digambarkan menggunakan *Class Diagram*, *Use Case Diagram*, *Activity Diagram*, dan *ERD*.

3.6.1. Perancangan ERD

Menurut salah satu para ahli, Brady dan Loonam (2010), Entity Relationship diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh *System Analysts* dalam tahap analisis persyaratan proyek pengembangan sistem. Berikut adalah ERD yang dirancang berdasarkan sistem yang di bangun.



Gambar 3.1 ERD (Entity Relationship Diagram)

Berikut merupakan penjelasan ERD (*Entity Relationship Diagram*) diatas. Masing-masing entitas memiliki atribut dan relasi sebagai berikut:

1. Entitas guru

Attribute: id_guru (PK), nama, NIP, alamat, email, no_telp

Penjelasan:

Pada entitas guru memiliki 3 relasi, yaitu antara mata pelajaran, kelas dan absensi. Dimana relasi antara entitas guru dengan mata pelajaran akan menghasilkan tugas guru.

2. Entitas siswa

Attribute: id_siswa (PK), nama, NIS, tempat_tanggal_lahir, alamat

Penjelasan:

Pada entitas siswa memiliki 3 relasi, yaitu antara mata pelajaran, kelas dan absensi. Dimana relasi antara entitas siswa dengan mata pelajaran akan menghasilkan nilai yang diperoleh siswa.

3. Entitas wali siswa

Attribute: id_wali (PK), nama, alamat, no_telp, email

Penjelasan:

Pada entitas wali siswa memiliki relasi dengan siswa. Dimana seorang wali siswa berelasi dengan siswa agar dapat melihat absensi dan nilai siswa.

4. Entitas kelas

Attribute: id_kelas (PK), tahun, periode

Penjelasan:

Pada entitas kelas memiliki relasi dengan guru dan siswa, dimana seorang guru dapat mengampu pada banyak kelas dan satu kelas dimiliki oleh banyak siswa.

5. Mata pelajaran

Attribute: id_mapel (PK), nama_mapel, periode

Penjelasan:

Pada entitas mata pelajaran memiliki relasi dengan guru dan siswa, dimana beberapa guru memiliki banyak mata pelajaran yang harus diampu dan

semua siswa wajib mengambil semua mata pelajaran yang diberikan sesuai dengan tahun ajaran.

6. Absensi

Attribute: id_absensi (PK), nama, waktu, alasan, periode

Penjelasan:

Pada entitas absensi memiliki relasi dengan siswa dan guru, dimana seorang siswa dapat memiliki banyak absensi pada periode dan waktu tertentu dan seorang guru dapat mengelola banyak absensi siswa dalam setiap periode.

7. Pengumuman

Attribute: id_pengumuman (PK), judul_pengumuman, isi, waktu_pengumuman

Penjelasan:

Pada entitas pengumuman memiliki relasi dengan guru, dimana seorang guru dapat memberikan banyak pengumuman pada setiap waktu.

8. Galeri

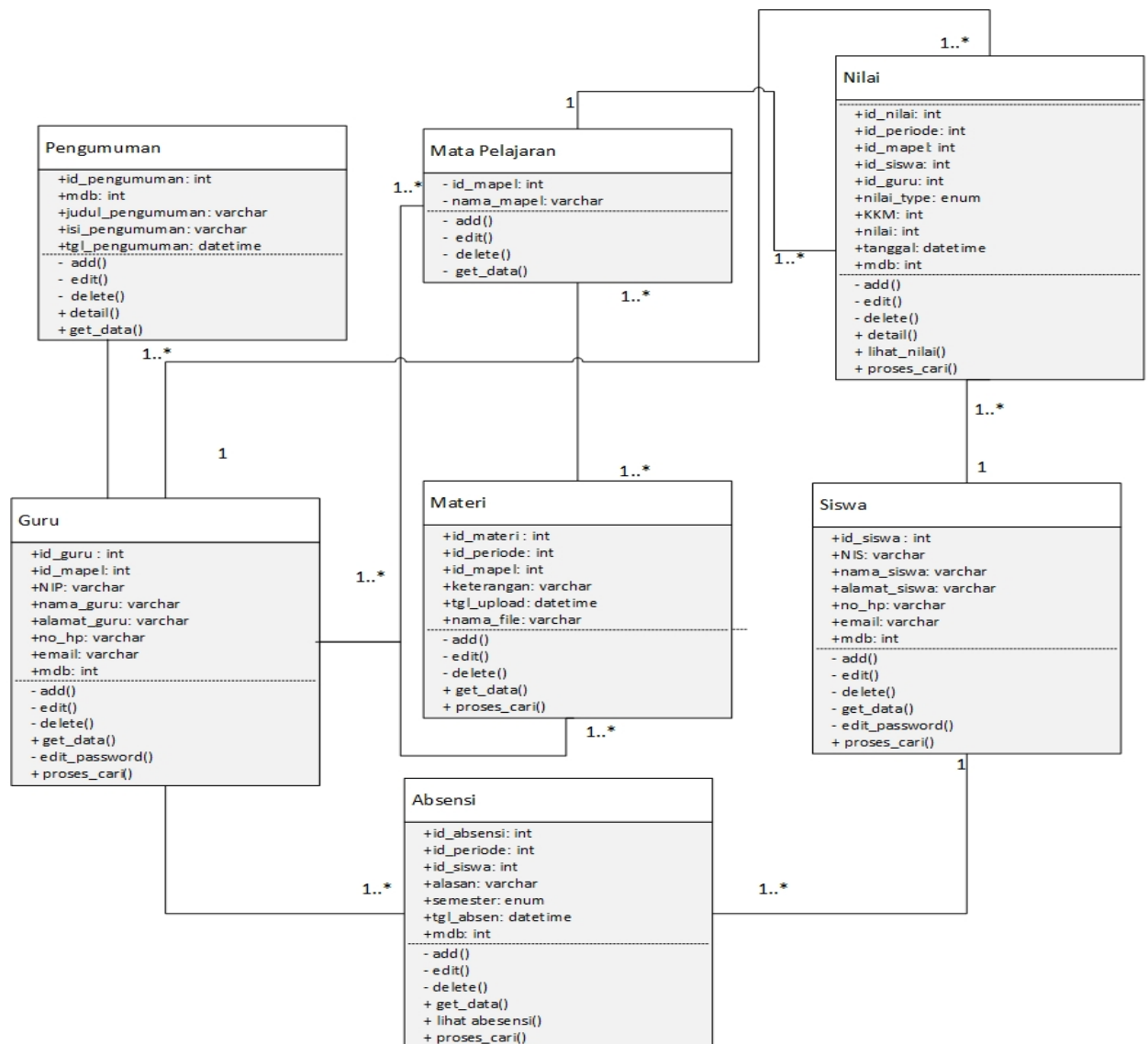
Attribute: id_galeri (PK), judul_galeri, isi, keterangan

Penjelasan:

Pada entitas galeri memiliki relasi dengan guru, dimana seorang guru dapat menambahkan banyak foto atau video pada setiap waktu.

3.6.2. Perancangan *Class Diagram*

Class diagram adalah model statis yang menggambarkan struktur dan deskripsi *class* serta hubungannya antara *class*, sehingga bisa di ketahui bagaimana relasi antar kelas pada sistem. Berikut adalah beberapa kelas yang masuk dalam *Class diagram* sistem.



Gambar 3.2 *Class Diagram*

Class Diagram memiliki fungsi masing-masing kelas, antara lain sebagai berikut:

a. Class Pengumuman

Class pengumuman adalah *class* pembantu dimana guru dapat memberikan pengumuman yang berupa perubahan jadwal, pengumuman jam kosong atau pengumuman lainnya yang dapat dilihat secara *online* oleh guru

dan siswa dimanapun dan kapanpun. Pada *class* pengumuman terdapat beberapa *method* yaitu sebagai berikut:

1. *Method add()*, berfungsi untuk menambahkan data pengumuman yang dapat diakses oleh admin dan guru.
2. *Method edit()*, berfungsi untuk mengubah data pengumuman yang telah ditambahkan sebelumnya oleh admin dan guru.
3. *Method delete()*, berfungsi untuk menghapus data pengumuman yang telah ditambahkan sebelumnya oleh admin dan guru.
4. *Method detail()*, berfungsi untuk menampilkan detail data pengumuman yang telah ditambahkan sebelumnya. Detail data pengumuman dapat dilihat oleh admin, guru, wali siswa dan siswa.
5. *Method get_data()*, berfungsi untuk mengambil data yang telah tersimpan pada *database*.

b. Class Mata Pelajaran

Class mata pelajaran adalah *class* tambahan dimana guru maupun admin dapat menambahkan jadwal sesuai kebutuhan atau merubah jadwal mata pelajaran agar dapat dilihat siswa secara online tanpa harus melihat di *madding* sekolah. Pada *class* mata pelajaran terdapat beberapa *method* yaitu sebagai berikut:

1. *Method add()*, berfungsi untuk operasi penambahan data master mata pelajaran yang dilakukan oleh admin.
2. *Method edit()*, berfungsi untuk melakukan operasi perubahan data master matapelajaran yang dilakukan oleh admin
3. *Method delete()*, berfungsi untuk melakukan operasi penghapusan master mata pelajaran yang dilakuakn oleh admin
4. *Method get_data()*, berfungsi untuk melakukan penarikan data mata pelajaran untuk di gunakan pada class lain nya

c. Class Nilai

Fungsi dari *class* nilai adalah dimana seorang guru dapat menambahkan, merubah, melihat dan menghapus nilai siswa secara *online* dimana dan

kapanpun untuk kemudian nilai yang telah dibuat segera di *upload* agar siswa dan wali siswa dapat melihat secara *online* perkembangan nilai siswa tersebut.

Pada *class* nilai terdapat beberapa *method* yaitu sebagai berikut:

1. *Method add()*, berfungsi untuk melakukan operasi penambahan nilai untuk siswa, yang bisa dilakukan oleh guru ataupun admin
2. *Method edit()*, berfungsi untuk melakukan operasi perubahan data nilai yang bisa dilakukan oleh admin ataupun guru.
3. *Method delete()*, berfungsi untuk melakukan operasi penghapusan data nilai, yang dapat dilakukan oleh admin ataupun guru.
4. *Method detail()*, berfungsi untuk melihat detail nilai dari setiap siswa yang bisa diakses oleh *admin*, guru, wali maupun siswa
5. *Method lihat_nilai()*, berfungsi untuk melihat daftar nilai yang bisa diakses oleh wali dan siswa
6. *Method proses_cari*, berfungsi untuk menyimpan daftar parameter pencarian yang berguna untuk memilah nilai berdasarkan keinginan pengguna.

d. *Class* Siswa

Fungsi dari *class* siswa adalah dimana di kumpulkan nya method-method yang berhubungan dengan pengolahan data siswa. Berikut method yang terdapat pada *class* siswa.

1. *Method add()*, berfungsi untuk melakukan operasi penambahan data siswa beserta login yang bisa dilakukan oleh admin.
2. *Method edit()*, berfungsi untuk melakukan operasi pengeditan data siswa yang bisa dilakukan oleh admin.
3. *Method delete()*, berfungsi untuk melakukan operasi penghapusan data siswa yang bisa dilakukan oleh admin.
4. *Method get_data()*, berfungsi untuk mengambil list data siswa yang ada.
5. *Method edit_password()*, berfungsi untuk melakukan operasi perubahan atau reset password pada akun login siswa, yang bisa dilakukan oleh admin
6. *Method proses_cari* berfungsi untuk menyimpan daftar parameter pencarian yang berguna untuk memilah data siswa berdasarkan keinginan pengguna.

e. **Class Materi**

Fungsi dari *class* materi adalah dimana guru dapat menambahkan, merubah, melihat dan menghapus *file* materi mata pelajaran yang diampu secara *online* di halaman *website* yang telah disediakan agar siswa dapat *download* materi tersebut untuk segera dipelajari. Berikut method yang terdapat pada *class* siswa.

1. *Method add()*, berfungsi untuk melakukan operasi penambahan data materi yang bisa di lakukan oleh guru dan admin.
2. *Method edit()*, berfungsi untuk melakukan operasi pengeditan data materi yang bisa dilakukan oleh guru dan admin.
3. *Method delete()*, berfungsi untuk melakukan operasi penghapusan data materi yang bisa lakukan oleh admin dan guru.
4. *Method get_data()*, berfungsi untuk mengambil *list* data materi guna di tampilkan kepada pengguna , agar bisa di dilihat dan di download
5. *Method proses_cari* berfungsi untuk menyimpan daftar parameter pencarian yang berguna untuk memilah data siswa berdasarkan keinginan perngguna

f. **Class Guru**

Fungsi dari *class* guru adalah dimana di kumpulan nya *method-method* yang berhubungan dengan pengolahan data guru. Berikut *method* yang terdapat pada *class* siswa. Berikut method yang terdapat pada *class* siswa.

1. *Method add()*, berfungsi untuk melakukan operasi penambahan data guru beserta login yang bisa di lakukan oleh admin.
2. *Method edit()*, berfungsi untuk melakukan operasi pengeditan data guru yang bisa dilakukan oleh admin.
3. *Method delete()*, berfungsi untuk melakukan operasi penghapusan data guru yang bisa lakukan oleh admin.
4. *Method get_data()*, berfungsi untuk mengambil *list* data guru yang ada.
5. *Method edit_password()*, berfungsi untuk melakukan operasi perubahan atau *reset password* pada akun *login* guru, yang bisa di lakukan oleh admin
6. *Method proses_cari* berfungsi untuk menyimpan daftar parameter pencarian yang berguna untuk memilah data guru berdasarkan keinginan perngguna.

g. **Class Absensi**

Fungsi dari *class* absensi adalah dimana guru yang sedang mengajar di satu kelas dapat mengecek kehadiran siswa dengan menambahkan absensi siswa yang tidak hadir dengan keterangan sesuai fakta agar kemudian data absensi hari itu disimpan untuk dapat dilihat oleh siswa atau wali siswa secara *online*.

Berikut method yang terdapat pada *class* absensi

1. *Method add()*, berfungsi untuk melakukan operasi penambahan data absensi siswa yang tidak hadir, penambahan data bisa dilakukan oleh admin dan guru
2. *Method edit()*, berfungsi untuk melakukan operasi perubahan data absensi siswa, perubahan data bisa dilakukan oleh admin dan guru
3. *Method delete()*, berfungsi untuk melakukan operasi penghapusan data absensi siswa, perubahan data bisa dilakukan oleh admin dan guru.
4. *Method get_data()*, berfungsi untuk melakukan operasi pengambilan *list* data absensi dari *database*.
5. *Method lihat_absensi*, berfungsi untuk melakukan operasi pengambilan data absensi untuk wali dan siswa.
6. *Method proses_cari* berfungsi untuk menyimpan daftar parameter pencarian yang berguna untuk memilah data absensi berdasarkan keinginan pengguna.

Berikut adalah penjelasan relasi antar *class* pada gambar *class* diagram diatas:

a. **Relasi *class* guru dengan *class* pengumuman**

Class guru dan pengumuman dihubungkan dengan atribut *id_guru*, hal ini dikarenakan setiap guru bisa mengisi data pengumuman. Dengan adanya *id_guru* pada *class* pengumuman maka setiap data pengumuman yang disimpan dapat diketahui siapa yang menulis pengumuman tersebut.

b. **Relasi *class* guru dengan *class* nilai**

Class guru dan nilai dihubungkan dengan atribut *id_guru*, hal ini dikarenakan setiap guru bisa mengisi data nilai untuk masing-masing siswa yang di ampu oleh guru. Dengan relasi ini dimungkinkan data dapat nilai dapat di pilah berdasarkan siapa guru yang menambah data nilai.

c. Relasi *class* guru dengan *class* mata pelajaran

Class guru dan pelajaran dihubungkan dengan atribut *id_guru*, hal ini dikarenakan setiap guru bisa memiliki satu atau beberapa mata pelajaran yang diampu. Jadi data-data mata pelajaran dapat di tampilkan juga siapa saja guru yang mengampu mata pelajaran tersebut.

d. Relasi *class* guru dengan *class* materi

Class guru dan materi dihubungkan dengan atribut *id_guru*, hal ini dikarenakan setiap guru bisa menyimpan banyak data materi, sehingga pada saat siswa ingin melihat data materi, data bisa di pilah berdasarkan siapa guru yang *upload* materi tersebut.

e. Relasi *class* guru dengan *class* absensi

Class guru dan absensi dihubungkan dengan atribut *id_guru*, hal ini dimaksudkan data absensi pada hari tertentu bisa dipilah berdasarkan siapa guru yang menyimpan data absensi.

f. Relasi *class* nilai dengan *class* mata pelajaran

Class nilai dan mata pelajaran dihubungkan dengan atribut *id_mapel*, hal ini dimaksudkan ketika data diakses dapat di pilah berdasarkan mata pelajaran tertentu, sehingga lebih mudah dalam penyajian datanya

g. Relasi *class* nilai dengan *class* siswa

Class nilai dengan siswa dihubungkan dengan atribut *id_siswa*, sehingga ketika siswa ingin melihat data siswa bisa di pilah berdasarkan dirinya sendiri atau siswa lainnya.

h. Relasi *class* materi dengan *class* mata pelajaran

Class materi dan mata mata pelajaran dihubungkan dengan atribut *id_mapel*. Hal ini dikarenakan pada saat data diakses oleh siswa, data materi dapat di pilah berdasarkan mata pelajaran tertentu.

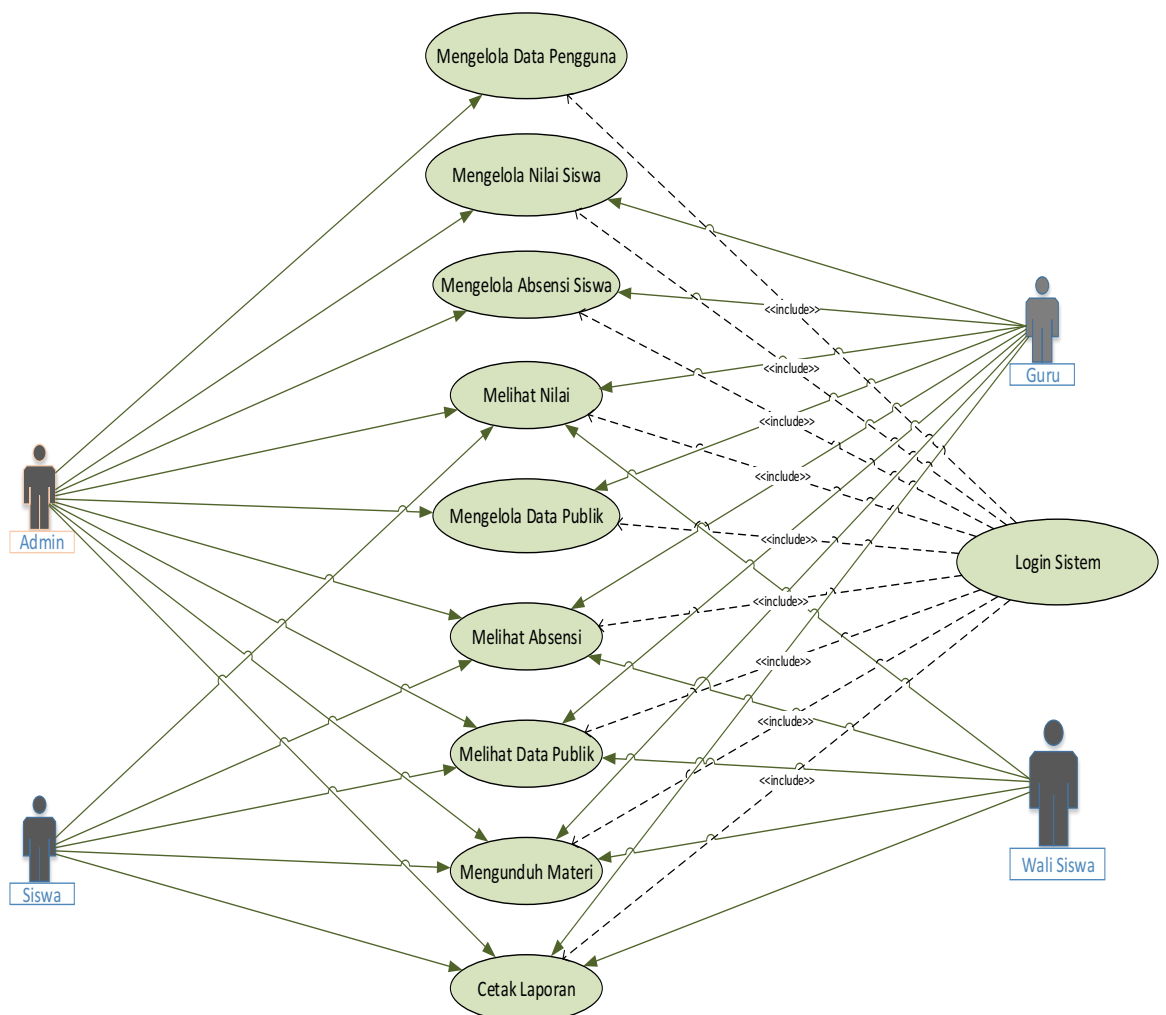
i. Relasi *class* absensi dengan *class* siswa

Class absensi dengan siswa dihubungkan dengan atribut *id_siswa*, hal ini dimaksudkan agar data absensi bisa di pilah berdasarkan siswa tertentu, dan

pada rentan waktu tertentu, sehingga bisa di kalkulasi jumlah absensi pada siswa tertentu.

3.6.3. Perancangan *Use Case Diagram*

Perancangan menggunakan *Use Case Diagram* adalah suatu model yang dapat fungsional dalam sebuah sistem yang menggunakan *actor* dan *use case*. Pada fase ini digram akan menjelaskan bagaimana user dalam melakukan tugasnya ketika ber interaksi dengan sistem. Berikut adalah gambar dari *use case* untuk website Sistem Informasi Akademik SMA N 1 Karangnom Klaten:



Gambar 3.3 Use Case Diagram

Berikut penjelasan dari gambar *use case* diagram diatas:

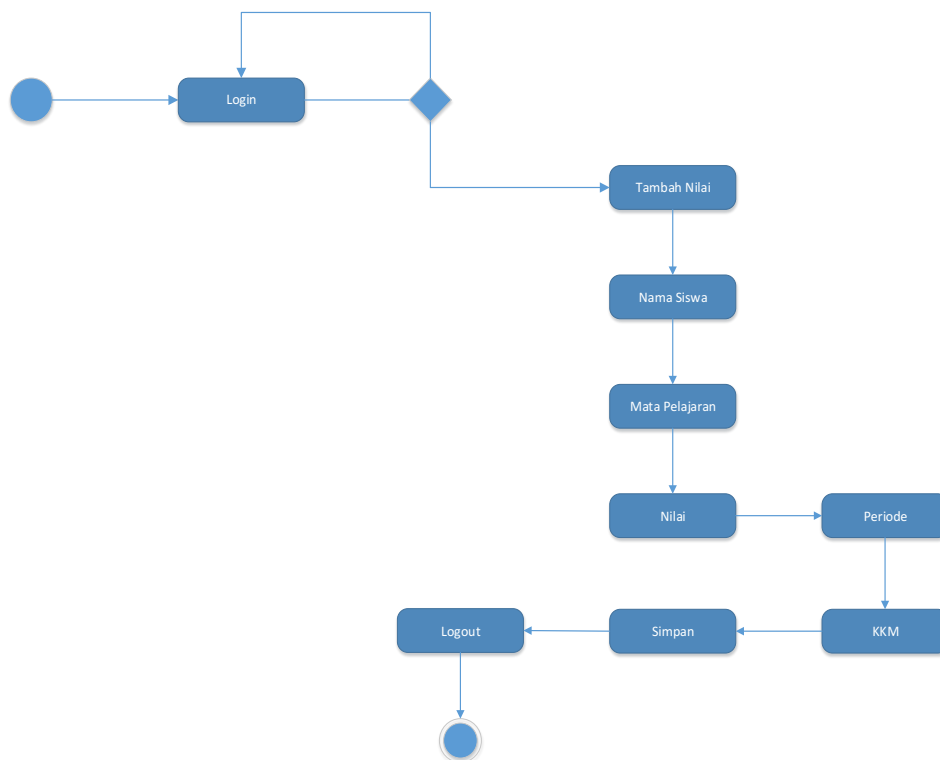
1. Terdapat 4 aktor pada *use case* diagram website SMA N 1 Karangnom Klaten yaitu *admin*, guru, siswa dan wali siswa.
2. *Admin* mengelola semua informasi seperti nilai siswa, absensi siswa, pengumuman, galeri, berita, profil sekolah dan materi yang berkaitan dengan sekolah.
3. Guru dapat menambahkan informasi seperti nilai siswa, absensi siswa, pengumuman, galeri, berita, profil sekolah dan materi yang berkaitan dengan sekolah.
4. Siswa dapat melihat nilai siswa, absensi siswa pada hari itu, pengumuman, galeri, berita, profil sekolah dan siswa dapat mengunduh materi yang telah di *upload* oleh guru atau *admin*.
5. Wali siswa hanya dapat melihat nilai anak didiknya sendiri, wali siswa juga dapat melihat absensi anak didiknya pada hari itu, wali siswa diperbolehkan melihat pengumuman sekolah, galeri, berita, profil sekolah dan mengunduh materi untuk siswa yang telah di *upload* oleh guru mata pelajaran atau *admin*.

3.6.4. Perancangan *Activity* Diagram

Perancangan dengan *activity* diagram adalah diagram yang menggambarkan *workflow* (alur kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

1. *Activity* diagram untuk modul nilai

Pada diagram dibawah digambarkan bagaimana alur ketika *user* ingin melihat atau menambah data nilai. Aktifitas apa yang dijalankan ketika modul ini di akses. Yang tentunya akses yang di berikan didasarkan pada hak akses yang di berikan pada *user* itu sendiri.



Gambar 3.4 Activity Diagram Nilai

Penjelasan *Activity Diagram* Nilai diatas akan dijelaskan dibawah ini:

1. *Admin*

- *Admin* melakukan *login* sebagai *admin*
- Setelah berhasil melakukan *login*, *admin* dapat langsung melihat data nilai, dan dapat menambahkan nilai baru atau batal menambahkan nilai baru kemudian masuk ke tampilan *logout*.
- Jika menambahkan nilai baru, kemudian *admin* akan memilih mata pelajaran yang akan ditambahkan nilai.
- Setelah memilih mata pelajaran, *admin* dapat memasukkan nilai siswa.
- Pilih periode nilai sesuai tahun ajaran.
- Kemudian nilai siswa disesuaikan dengan standar KKM yang sedang berlaku, apakah nilai siswa dapat melampaui standar KKM atau tidak.

- Kemudian nilai siswa disimpan dan otomatis tersimpan ke dalam *database*.
- Setelah selesai menambahkan nilai, *admin* dapat *logout*.

2. Guru

- Melakukan *login* guru
- Setelah berhasil melakukan *login*, guru dapat langsung melihat data nilai, dan dapat menambahkan nilai baru atau batal menambahkan nilai baru kemudian masuk ke tampilan *logout*.
- Jika menambahkan nilai baru, kemudian guru akan memilih mata pelajaran yang akan ditambahkan nilai.
- Setelah memilih mata pelajaran, guru dapat memasukkan nilai siswa.
- Pilih periode nilai sesuai tahun ajaran
- Kemudian nilai siswa disesuaikan dengan standar KKM yang sedang berlaku, apakah nilai siswa dapat melampaui standar KKM atau tidak.
- Setelah nilai siswa dimasukkan, nilai siswa disimpan dan otomatis tersimpan ke dalam *database*.
- Kemudian guru dapat *logout*

3. Siswa

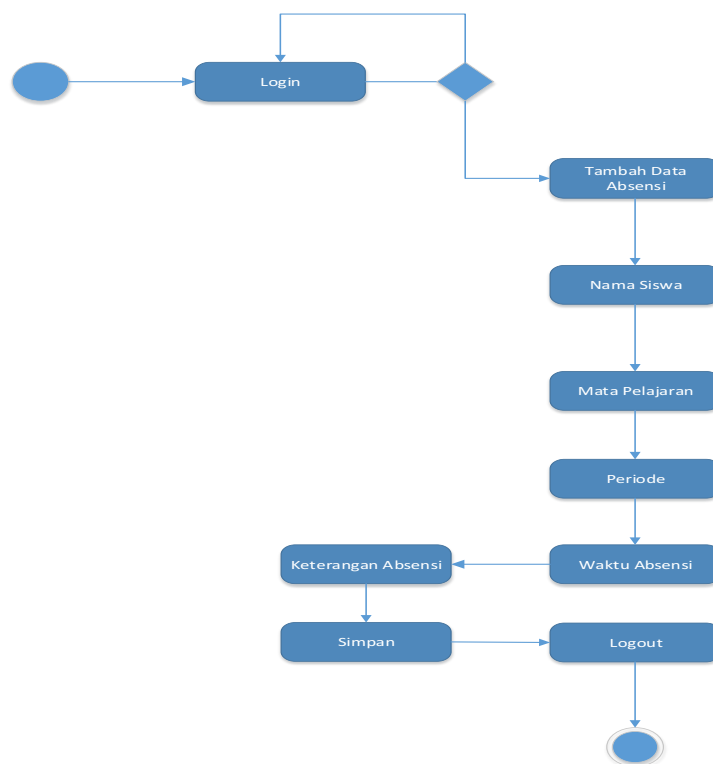
- Melakukan *login* siswa
- Setelah berhasil melakukan *login*, siswa dapat langsung melihat data nilai siswa.
- Kemudian siswa dapat *logout*

4. Orang Tua

- Melakukan *login* orang tua
- Setelah berhasil melakukan *login*, orang tua dapat langsung melihat data nilai siswa.
- Kemudian orang tua dapat *logout*

2. Activity diagram untuk modul Absensi

Pada diagram dibawah digambarkan bagaimana alur ketika *user* ingin melihat atau menambah data absensi. Aktifitas apa yang dijalankan ketika modul ini di akses.



Gambar 3. 5 *Activity Diagram Absensi*

Penjelasan *Activity Diagram Absensi* akan dijelaskan dibawah ini:

1. *Admin*

- Melakukan *login admin*
- Setelah berhasil melakukan *login*, *admin* dapat langsung melihat data absensi siswa, dan dapat menambahkan absensi terbaru sesuai daftar hadir siswa atau batal menambahkan absensi baru kemudian masuk ke tampilan *logout*.
- Jika menambahkan absensi baru, kemudian *admin* akan memilih mata pelajaran dimana siswa tersebut tidak hadir pada jam belajar mengajar.
- Setelah memilih mata pelajaran, *admin* dapat menambahkan periode absensi dan waktu absensi, dimana periode berisi tahun ajaran dan semester dan waktu tersebut berisi tanggal absensi, bulan absensi, tahun absensi.

- Pada tahap ini admin melakukan pengisian keterangan absensi. Apakah siswa tersebut absen karena bolos, izin, sakit atau tanpa keterangan.
- Kemudian data absensi siswa disimpan dan otomatis tersimpan ke dalam *database*.
- Setelah itu *admin* melakukan *logout*

2. Guru

- Melakukan *login* guru
- Setelah berhasil melakukan *login*, guru dapat langsung melihat data absensi siswa, dan dapat menambahkan absensi terbaru sesuai daftar hadir siswa atau batal menambahkan absensi baru kemudian masuk ke tampilan *logout*.
- Jika menambahkan absensi baru, kemudian guru akan memilih mata pelajaran dimana siswa tersebut tidak hadir pada jam belajar mengajar.
- Setelah memilih mata pelajaran, guru dapat menambahkan periode absensi dan waktu absensi, dimana periode berisi tahun ajaran dan semester, dimana waktu tersebut berisi tanggal absensi, bulan absensi, tahun absensi.
- Pada tahap ini guru melakukan pengisian keterangan absensi. Apakah siswa tersebut absen karena bolos, izin, sakit atau tanpa keterangan.
- Kemudian data presensi siswa disimpan dan otomatis tersimpan ke dalam *database*.
- Setelah itu, guru melakukan *logout*

3. Siswa

- Melakukan *login* siswa
- Jika berhasil melakukan *login*, siswa dapat langsung melihat data absensi siswa.
- Setelah melihat data absensi, siswa dapat *logout*

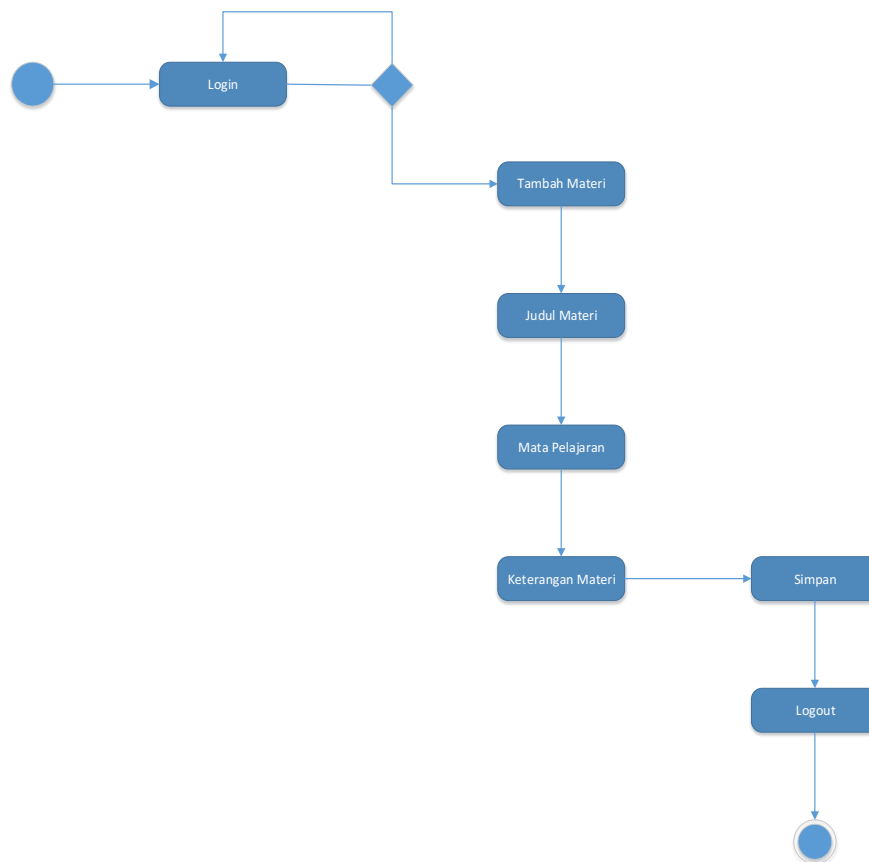
4. Wali Siswa

- Melakukan *login* wali siswa
- Jika berhasil melakukan *login*, wali siswa dapat langsung melihat data absensi siswa.

- Setelah melihat data absensi siswa, wali siswa dapat logout

3. *Activity diagram* untuk modul materi

Pada diagram dibawah digambarkan bagaimana alur ketika *user* ingin melihat atau menambah data materi Aktifitas apa yang dijalankan ketika modul ini di akses.



Gambar 3. 6 *Activity Diagram Materi*

Penjelasan *Activity Diagram Materi* akan dijelaskan dibawah ini:

1. *Admin*

- Pada tahap pertama, *admin* melakukan *login* sebagai *admin* terlebih dahulu.
- Setelah berhasil *login*, *admin* dapat melihat materi yang telah di *upload* sebelumnya.

- *Admin* juga dapat menambahkan materi baru atau batal menambahkan materi baru kemudian *logout*.
- Jika *admin* menambahkan materi baru, *admin* harus mengisi judul dari materi yang di *upload*.
- Setelah mengisi judul materi, *admin* mengisi mata pelajaran sesuai dengan materi.
- Pada tahap ini *admin* dapat menambahkan keterangan materi yang di *upload*.
- Setelah materi di *upload* dan disimpan, otomatis materi tersimpan di database.
- *Admin* dapat *logout*

2. **Guru**

- Pada tahap pertama, guru melakukan *login* sebagai guru terlebih dahulu.
- Setelah berhasil *login*, guru dapat melihat materi yang telah di *upload* sebelumnya.
- Guru juga dapat menambahkan materi baru atau batal menambahkan materi baru kemudian *logout*.
- Jika guru menambahkan materi baru, *admin* harus mengisi judul dari materi yang di *upload*.
- Setelah mengisi judul materi, guru mengisi mata pelajaran sesuai dengan materi.
- Pada tahap ini guru dapat menambahkan keterangan materi yang di *upload*.
- Setelah materi di *upload* dan disimpan, otomatis materi tersimpan di *database*.
- Guru dapat *logout*

3. **Siswa**

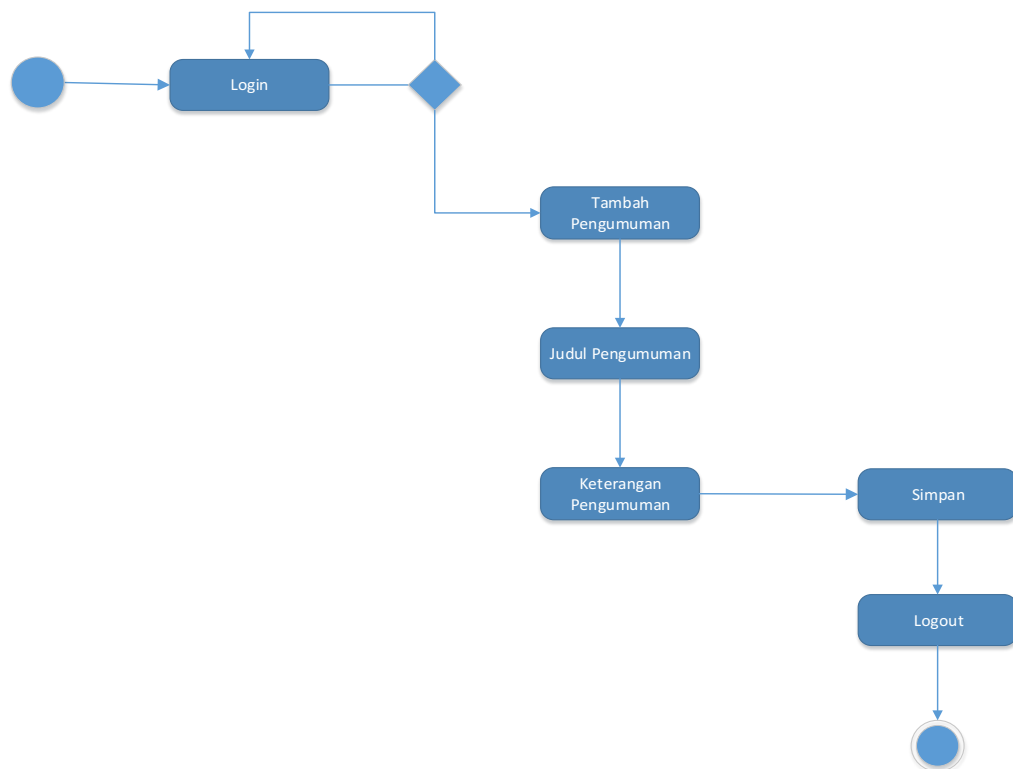
- Siswa melakukan *login* sebagai siswa terlebih dahulu
- Setelah berhasil *login*, siswa dapat melihat materi yang telah di *upload*
- Kemudian siswa dapat *logout* setelah melihat materi

4. Wali Siswa

- Wali Siswa melakukan *login* sebagai wali siswa terlebih dahulu
- Setelah melakukan *login*, wali siswa dapat langsung melihat materi yang telah di *upload*
- Wali siswa dapat *logout* setelah melihat materi

4. *Activity diagram* untuk modul pengumuman

Pada diagram dibawah digambarkan bagaimana alur ketika *user* ingin melihat atau menambah data pengumuman aktifitas apa yang dijalankan ketika modul ini di akses.



Gambar 3. 7 *Activity Diagram* Pengumuman

Penjelasan *Activity Diagram* Pengumuman akan dijelaskan dibawah ini:

1. *Admin*

- *Admin* melakukan *login* sebagai *admin* terlebih dahulu

- Kemudian *admin* dapat melihat data pengumuman yang telah di *upload*
- *Admin* juga dapat menambahkan pengumuman terbaru, dan *admin* dapat membatalkan tambah pengumuman untuk kemudian *logout*.
- Jika *admin* ingin menambahkan pengumuman, maka selanjutnya *admin* mengisi judul pengumuman
- Kemudian menambahkan keterangan pengumuman
- Setelah selesai menambahkan pengumuman, selanjutnya simpan pengumuman dan otomatis pengumuman akan tersimpan di *database*.
- *Admin* dapat melakukan *logout*

2. Guru

- Guru melakukan *login* sebagai guru terlebih dahulu
- Kemudian guru dapat melihat data pengumuman yang telah di *upload*
- Guru juga dapat menambahkan pengumuman terbaru, dan guru dapat membatalkan tambah pengumuman untuk kemudian *logout*.
- Jika guru ingin menambahkan pengumuman, maka selanjutnya guru mengisi judul pengumuman
- Kemudian menambahkan keterangan pengumuman
- Setelah selesai menambahkan pengumuman, selanjutnya simpan pengumuman dan otomatis pengumuman akan tersimpan di *database*.
- Guru dapat melakukan *logout*

3. Siswa

- Siswa melakukan *login* sebagai siswa terlebih dahulu
- Siswa dapat langsung melihat pengumuman yang telah di *upload*
- Siswa melakukan *logout*

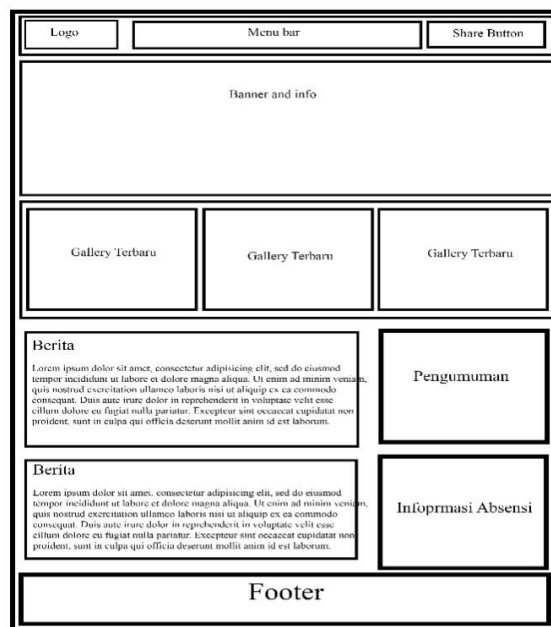
4. Orang Tua

- Siswa melakukan *login* sebagai siswa terlebih dahulu
- Siswa dapat langsung melihat pengumuman yang telah di *upload*
- Siswa melakukan *logout*

3.6.5. Rancangan Antarmuka (*User Interface*)

Dalam sistem informasi akademik SMA Negeri 1 Karangnom Klaten terdapat rancangan desain tampilan utama dari sistem nantinya. Rancangan desain ini yang nantinya akan menjadi patokan dasar tampilan jadi dari sistem nantinya. Adapun rancangan desain dari tampilan sistem yang akan digunakan pengguna nantinya sebagai berikut:

1. Desain *home_page*



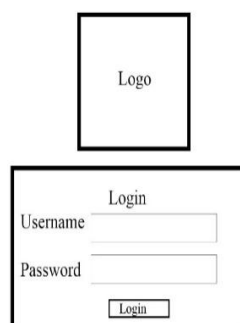
Gambar 3.8 Desain *home_page*

Dibawah ini penjelasan untuk *home_page*:

- Pada bagian kiri atas terdapat Logo SMA Negeri 1 Karangnom Klaten
- Pada bagian menu bar berfungsi untuk menampilkan menu-menu yang dapat dibuka.
- Pada bagian kanan terdapat *Share Button* yang berfungsi untuk membagikan galeri, berita atau pengumuman.
- Pada bagian *banner and Info* berisikan gambar profil dan sekilas info SMA Negeri 1 Karangnom Klaten.

- Pada galeri berisi foto siswa, guru ataupun *staff* SMA Negeri 1 Karanganom Klaten. Galeri juga dapat berisikan foto atau video ketika siswa atau guru sedang berkunjung ke suatu tempat.
- Pada bagian menu pengumuman berisi informasi terbaru kepada siswa tentang proses belajar mengajar maupun umum.
- Menu absensi berfungsi untuk mengetahui nama siswa yang tidak hadir pada jam belajar mengajar.
- Menu berita berisi berita-berita terhangat dan terbaru yang dapat menginspirasi siswa, guru maupun *staff* SMA Negeri 1 Karanganom Klaten.
- Pada bagian *footer* menampilkan kapan *website* dibuat.

2. Desain Halaman *Login*



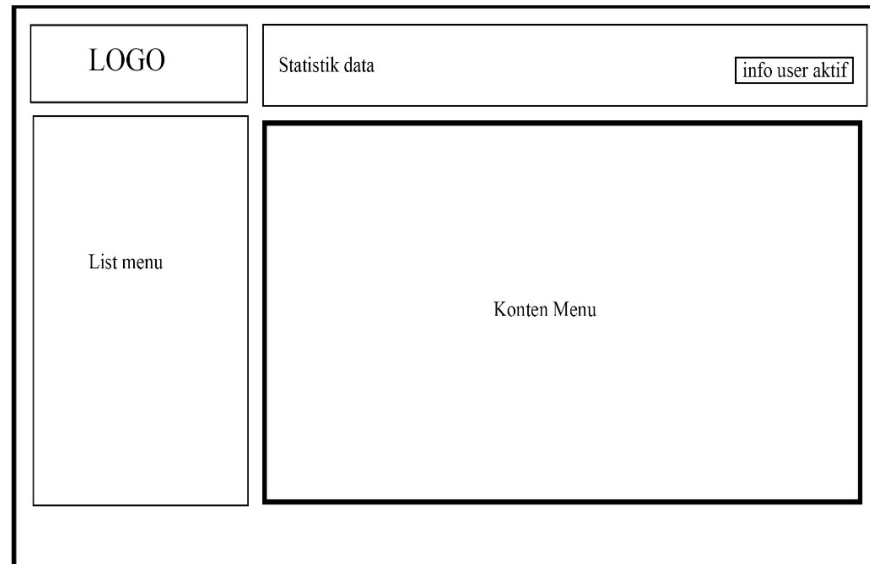
The diagram illustrates the layout of a login page. It consists of two main components: a logo box and a login form box. The logo box is a square with the word 'Logo' centered inside. The login form box is a larger rectangle containing the word 'Login' at the top, followed by two input fields labeled 'Username' and 'Password', and a 'Login' button at the bottom.

Gambar 3. 9 *Desain Halaman Login*

Dibawah ini penjelasan untuk desain halaman *login*:

- Pada bagian halaman *login* antar pengguna sama menggunakan *username* dan *password*, hanya berbeda pada hak pengguna.
- Pada bagian form *login* terdapat gambar yang nantinya menggunakan logo SMA N 1 Karanganom Klaten

3. Desain Halaman Utama



Gambar 3. 10 Desain Halaman Utama

Berikut merupakan penjelasan dari desain halaman utama:

- Pada bagian kiri atas terdapat logo yang nantinya menggunakan logo SMA N 1 Karanganom Klaten.
- Pada bagian statistik data berfungsi untuk menampilkan jumlah data yang tersimpan selama *website* berjalan seperti pengumuman, materi, absensi siswa yang tidak masuk.
- Pada bagian kanan atas terdapat info *user* aktif dimana fungsinya untuk menampilkan info *user* yang sedang login.
- Pada bagian kiri terdapat *list* menu yang akan menampilkan menu-menu yang dapat dibuka ketika sedang membuka *website*.
- Pada bagian tengah terdapat konten menu dimana nantinya menampilkan profil SMA N 1 Karanganom Klaten.

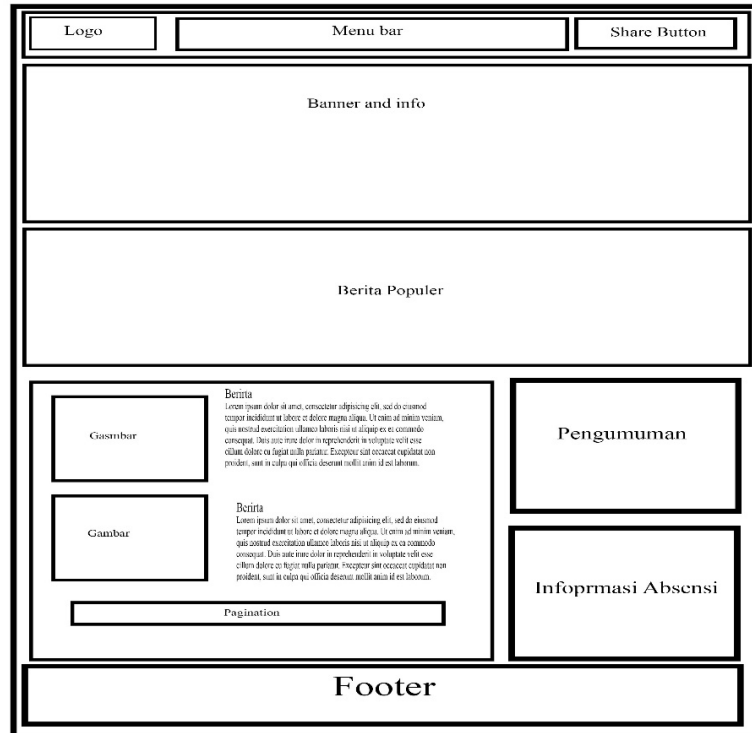
4. Desain Halaman *add_data*

Gambar 3. 11 Desain Halaman *Add_data*

Dibawah ini adalah penjelasan dari desain halaman *add_data*:

- Pada bagian kiri atas terdapat gambar logo SMA Negeri 1 Karangnom Klaten.
- Pada bagian tengah atas terdapat statistik data yang akan menampilkan jumlah data yang tersimpan selama *website* berjalan seperti pengumuman, materi, absensi siswa yang tidak masuk.
- Pada bagian kanan atas terdapat info *user* aktif dimana fungsinya untuk menampilkan info *user* yang sedang *login*.
- Pada bagian kiri terdapat *list* menu yang akan menampilkan menu-menu yang dapat dibuka ketika sedang membuka *website*.
- Pada bagian tengah halaman terdapat form untuk menambah data dengan label sebagai tempat untuk memasukkan data, *button* simpan untuk menyimpan data yang telah di isi dan *button* kembali untuk membatalkan pengisian data dan kembali ke halaman sebelumnya.
- Pada bagian *footer* menampilkan kapan *website* dibuat.

5. Desain Halaman Berita



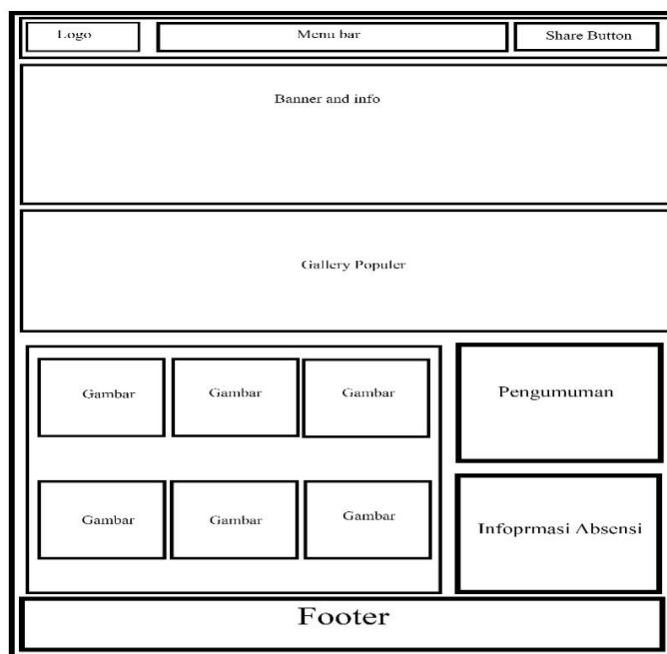
Gambar 3.12 Desain Halaman Berita

Dibawah ini adalah penjelasan dari desain halaman berita:

- Pada bagian kiri atas terdapat gambar logo SMA Negeri 1 Karangnom Klaten.
- Pada bagian menu bar berfungsi untuk menampilkan menu-menu yang dapat dibuka.
- Pada bagian kanan terdapat *share button* yang berfungsi untuk membagikan galeri, berita atau pengumuman.
- Pada bagian *banner and info* berisikan gambar profil dan sekilas info SMA Negeri 1 Karangnom Klaten.
- Pada bagian tengah terdapat berita terpopuler yang sedang hangat dibicarakan.
- Pada bagian kolom berita berisi gambar dari berita tersebut beserta sedikit keterangan dari berita. Bagian *Pagination* berfungsi untuk melihat berita yang lain.

- Pada bagian menu pengumuman berisi informasi terbaru kepada siswa tentang proses belajar mengajar maupun umum.
- Pada bagian menu absensi berfungsi untuk mengetahui nama siswa yang tidak hadir pada jam belajar mengajar.
- Pada bagian menu berita berisi berita-berita terhangat dan terbaru yang dapat menginspirasi siswa, guru maupun *staff* SMA Negeri 1 Karangnom Klaten.
- Pada bagian *footer* menampilkan kapan *website* dibuat.

6. Desain Halaman Galeri



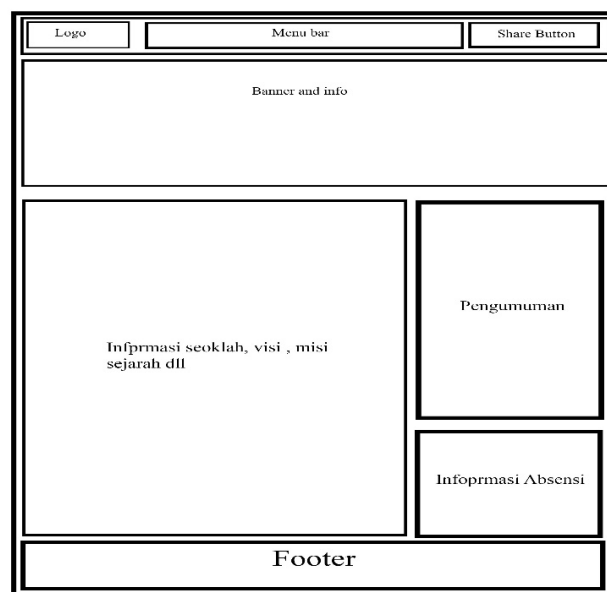
Gambar 3.13 Desain Halaman Galeri

Dibawah ini adalah penjelasan dari desain halaman galeri:

- Pada bagian kiri atas terdapat gambar logo SMA Negeri 1 Karangnom Klaten.
- Pada bagian menu bar berfungsi untuk menampilkan menu-menu yang dapat dibuka.
- Pada bagian kanan terdapat *share button* yang berfungsi untuk membagikan galeri, berita atau pengumuman.

- Pada bagian *banner and info* berisikan gambar profil dan sekilas info SMA Negeri 1 Karangnom Klaten.
- Pada bagian tengah terdapat galeri terpopuler yang sering dilihat.
- Pada bagian kiri bawah terdapat *list* dari galeri SMA Negeri 1 Karangnom Klaten yang berisikan gambar-gambar kegiatan proses belajar mengajar ataupun kunjungan ke suatu lokasi.
- Pada bagian menu pengumuman berisi informasi terbaru kepada siswa tentang proses belajar mengajar maupun umum.
- Pada bagian menu absensi berfungsi untuk mengetahui nama siswa yang tidak hadir pada jam belajar mengajar.
- Pada bagian *footer* menampilkan kapan *website* dibuat.

7. Desain Halaman Profil



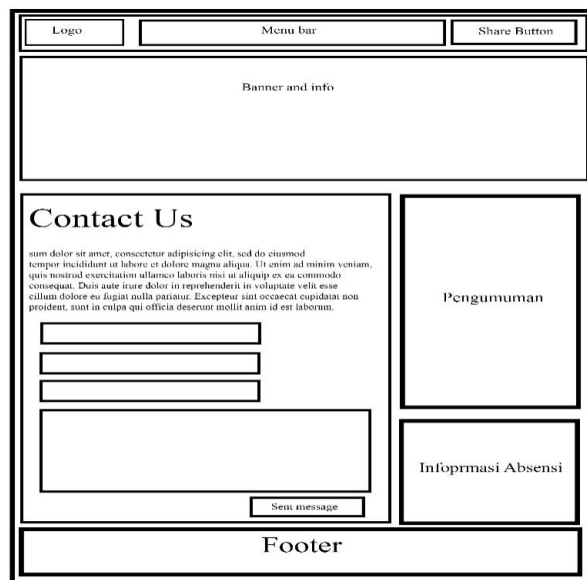
Gambar 3. 14 Desain Halaman Profil

Dibawah ini adalah penjelasan dari desain halaman profil:

- Pada bagian kiri atas terdapat gambar logo SMA Negeri 1 Karangnom Klaten.

- Pada bagian menu bar berfungsi untuk menampilkan menu-menu yang dapat dibuka.
- Pada bagian kanan terdapat *Share Button* yang berfungsi untuk membagikan galeri, berita atau pengumuman.
- Pada bagian *banner and info* berisikan gambar profil dan sekilas info SMA Negeri 1 Karangnom Klaten.
- Pada bagian kiri terdapat profil lengkap SMA Negeri 1 Karangnom Klaten yang berisi visi, misi, sejarah SMA Negeri 1 Karangnom Klaten, dan informasi sekolah.
- Pada bagian menu pengumuman berisi informasi-informasi terbaru kepada siswa tentang proses belajar mengajar maupun umum.
- Pada bagian menu absensi berfungsi untuk mengetahui nama siswa yang tidak hadir pada jam belajar mengajar.
- Pada bagian *footer* menampilkan kapan *website* dibuat.

8. Desain Halaman *Contact_Us*



Gambar 3. 15 Desain Halaman *Contact_Us*

Dibawah ini adalah penjelasan dari Desain Halaman *Contact_Us*:

- Pada bagian kiri atas terdapat gambar logo SMA Negeri 1 Karangnom Klaten.

- Pada bagian menu bar berfungsi untuk menampilkan menu-menu yang dapat dibuka.
- Pada bagian kanan terdapat *Share Button* yang berfungsi untuk membagikan galeri, berita atau pengumuman agar diketahui lebih banyak orang.
- Pada bagian *Banner* and Info berisikan gambar profil dan sekilas info SMA Negeri 1 Karangnom Klaten.
- Pada bagian kiri terdapat form untuk mengirimkan pesan atau saran kepada pihak sekolah dengan mengisi data dan keterangan pada label yang telah disediakan. Kemudian terdapat *button send message* untuk mengirimkan pesan.
- Pada menu pengumuman berisi informasi-informasi terbaru kepada siswa tentang proses belajar mengajar maupun umum.
- Menu absensi berfungsi untuk mengetahui nama siswa yang tidak hadir pada jam belajar mengajar.
- Pada bagian *footer* menampilkan kapan *website* dibuat.