

BAB III

METODOLOGI

3.1. Alat dan Bahan Penelitian

Penelitian ini membutuhkan alat-alat untuk mendukung berjalannya perancangan dan implementasi aplikasi *web*, antara lain:

3.1.1 Alat

Alat yang digunakan berupa perangkat keras dan perangkat lunak

1. Perangkat Keras

Personal Computer (PC) / Laptop

- 32/64 bit Architecture Processor.
- 2GB Random Access Memory (RAM).

2. Perangkat Lunak

- a. *Browser Internet* (Google Chrome atau Mozilla Firefox).
- b. My SQL
- c. XAMPP
- d. Sistem Operasi Windows 7/8.
- e. PHP

3.1.2 Bahan

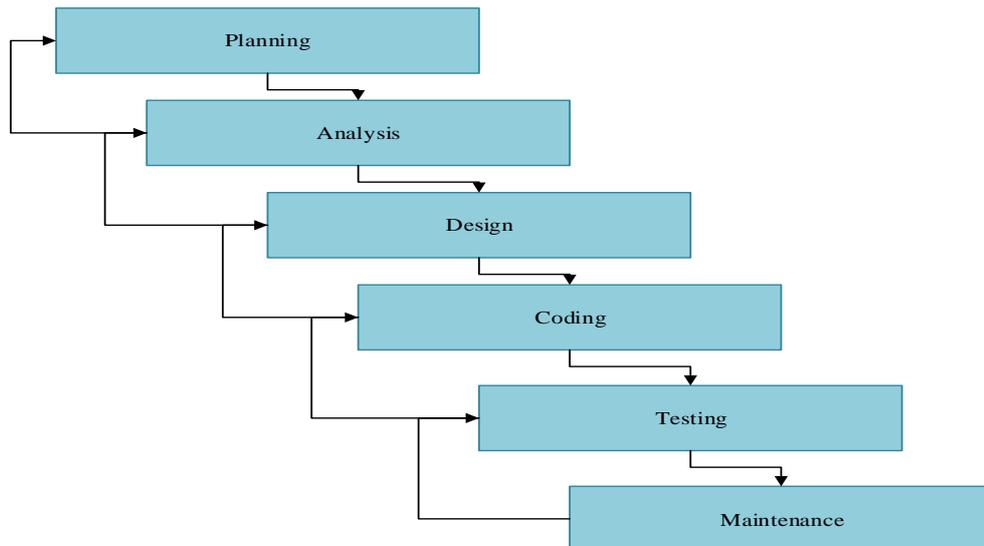
Bahan yang digunakan dalam penelitian ini adalah :

1. Data yang diperoleh melalui studi literatur berdasarkan penelitian sebelum yang masih memiliki kaitan dengan aplikasi *web* yang dikembangkan. Dari data tersebut dapat diperoleh pemahaman mengenai kebutuhan dari aplikasi *web* yang akan dikembangkan pada pembuatan aplikasi *web* ini.
2. Data dari TK Aba tegal gendu.

3.2. Metodologi Pengembangan Sistem

Untuk pengembangan sistem penelitian ini menggunakan model *Software Development Life Cycle (SDLC)*. Selain untuk proses produksi, SDLC juga penting untuk proses *maintenance software* itu sendiri.

Model SDLC yang dipakai dalam penelitian ini adalah model *Waterfall*. *Waterfall Model* atau *Classic Life Cycle* merupakan model yang paling banyak dipakai dalam *Software Engineering (SE)*. Disebut *waterfall* karena tahap demi tahap yang harus dilalui menunggu selesainya tahap sebelumnya dan berjalan berurutan. Gambaran untuk model SDLC dapat dilihat pada gambar 3.1.



Gambar 3.1 Model SDLC (*Software Development Life Cycle*)

Model SDLC diatas tersebut menjelaskan sebagai berikut:

- a. Tahapan Perencanaan yang sehubungan dengan kebutuhan pengguna, kelayakan baik secara teknik maupun secara teknologi. Pada tahapan ini dilakukan perancangan sistem yang akan dibangun dalam hal ini aplikasi *web* Sistem Informasi Managemen TK.
- b. Tahapan analisis, merupakan proses pendalaman mengenai permasalahan dan resiko pengguna.
- c. Tahap perancangan, Tahap ini menyangkut perancangan sistem dimana kita akan memberikan solusi dari masalah yang muncul pada tahap analisis.
- d. Tahapan implementasi, tahapan dimana perancangan sistem diimplementasikan ke situasi nyata dengan pemilihan perangkat keras penyusun design. Untuk implementasinya yaitu dengan membuat rancangan aplikasi *web* menggunakan PHP.

- e. Tahapan pengujian, tahapan yang menentukan design yang sudah dibuat sudah sesuai dengan kebutuhan pengguna atau belum. Tujuan pengujian ini adalah untuk meminimalisir kesalahan design aplikasi *web* sehingga sistem yang dikembangkan dapat berjalan dengan sebaik mungkin.
- f. Tahap pemeliharaan, adalah tahap dimana dilakukannya perawatan dan pemeliharaan aplikasi *web*, Jika diperlukan perbaikan kecil kemudian jika periode sistem sudah habis maka akan masuk kedalam tahap perencanaan.

3.4. Metodologi Pengumpulan Data

Jenis data yang dikumpulkan dalam penelitian ini adalah data primer dan data sekunder sebagai berikut:

3.4.1. Data Primer

Data primer yaitu data yang berasal dari perorangan, kelompok, panel atau data terselubung. Dalam memperoleh data primer perlu dilakukan wawancara dan observasi.

1. Observasi

Tujuan dari observasi adalah dengan mendeskripsikan setting yang diamati, tempat kegiatan orang-orang yang berpartisipasi dalam kegiatan tersebut dan makna apa yang diamati menurut perspektif pengamat (Patton, 1990). Pengamatan dapat diklasifikasikan atas pengamatan melalui cara berperan serta dan yang tidak berperan serta (Moeleong, 1998 : 126).

Pada pengamatan berperan serta, pengamat melakukan dua peran sekaligus, yaitu sebagai pengamat dan sekaligus menjadi anggota resmi dari kelompok yang diamati. Pada penelitian ini peneliti mendapatkan data dengan mengamati objek yang akan diteliti secara langsung di tempat penelitian yaitu TK Aba tegal gendu. Pengumpulan data meliputi sistem pembayaran, kehadiran dan nilai murid di TK Aba tegal gendu.

2. Wawancara

Wawancara merupakan percakapan dengan maksud tertentu. Percakapan dilakukan antara peneliti yang mengajukan pertanyaan dan yang diwawancarai memberikan jawaban atas pertanyaan itu (Moleong, 1998). Proses wawancara

ini dilakukan di TK Aba tegal gendu untuk mendapatkan data dengan melakukan dialog langsung dengan orang yang dianggap dapat memberikan informasi yang dibutuhkan oleh penulis yaitu Suryati ningsih Staff di TK Aba Tegal gendu. Tujuan melakukan wawancara adalah untuk mengetahui sistem yang dibutuhkan oleh TK Aba Tegal gendu dari segi arsitektur aplikasi, teknologi yang digunakan serta pemanfaatannya.

3.4.2. Data Sekunder

Data sekunder merupakan data yang diperoleh melalui perantara. Data sekunder umumnya berupa bukti, catatan ataupun laporan yang telah tersusun dalam arsip yang dipublikasikan dan yang tidak dipublikasikan. Dalam penelitian ini data sekunder bersumber dari tinjauan pustaka dan landasan teori melalui berbagai jurnal, laporan, artikel, referensi yang diambil dari perpustakaan maupun internet untuk mendapatkan data yang diperlukan.

3.4.3. Pengembangan Website

Pengembangan *website* dilakukan dengan membuat *website* untuk Sistem Informasi Managemen TK yang dimulai dari perencanaan lalu analisis kebutuhan sebagai penunjang, agar dapat mencapai tujuan penggunaan *website* untuk meningkatkan kinerja sistem managemen data yang ada di TK. *Website* yang dikembangkan untuk TK Aba Tegal gendu merupakan jenis *website* dinamis.

3.5.2 Analisis Kebutuhan Website

Dari hasil kebutuhan yang didapat dari berbagai sumber dan diskusi bersama dengan pihak TK Aba Tegal gendu, maka didapatkanlah beberapa masukan untuk mengembangkan aplikasi berbasis *web* ini agar sesuai dengan tujuan dan manfaatnya, seperti berikut:

- a. Halaman *Login* admin untuk masuk ke *dashboard* admin
- b. Halaman *Home* sebagai halaman utama *website* Sistem Informasi Managemen TK

- c. Halaman *kelola data* sebagai halaman untuk input data dan pengelolaan data.
- d. Halaman Menu Guru, Murid, Orang Tua, pencatan pembayaran SPP, daftar hadir murid, serta nilai murid.

3.5.3 Rancangan

Metode perancangan yang digunakan adalah *Unified Modeling Language (UML)*. Model UML yang dipakai dalam pengembangan adalah model *Use Case Diagram*, *Activity Diagram* dan *Class Diagram*. Selain itu dilakukan perancangan basis data dengan menggunakan ER Diagram.

3.5.3.1 Use Case Diagram

Berdasarkan analisis kebutuhan dibuatlah *use case* diagram untuk membantu perancangan aplikasi *website* ini. Terdapat 3 aktor, yaitu:

- A. Admin adalah yang berperan sebagai penanggung jawab penuh untuk mengurus operasional *website*. Admin juga memiliki otoritas untuk menambah, mengedit dan menghapus data baik guru maupun siswa.
- B. Guru adalah yang berperan sebagai menambah, mengedit dan menghapus data siswa, spp, absensi, dan raport.
- C. Siswa adalah yang berperan sebagai pengunjung yang hanya dapat melihat spp, absensi dan raport.

Berikut adalah *use case* yang telah teridentifikasi untuk dibuat pada *use case* diagram:

- *Login*: untuk autentikasi admin untuk dapat mengakses *dashboard* admin
- *Kelola data*: untuk menambahkan, memperbaharui atau menghapus data yang dimasukkan.

- Laporan : untuk menampilkan rangkuman dari spp, absensi, dan raport.

3.5 Rancangan Sistem

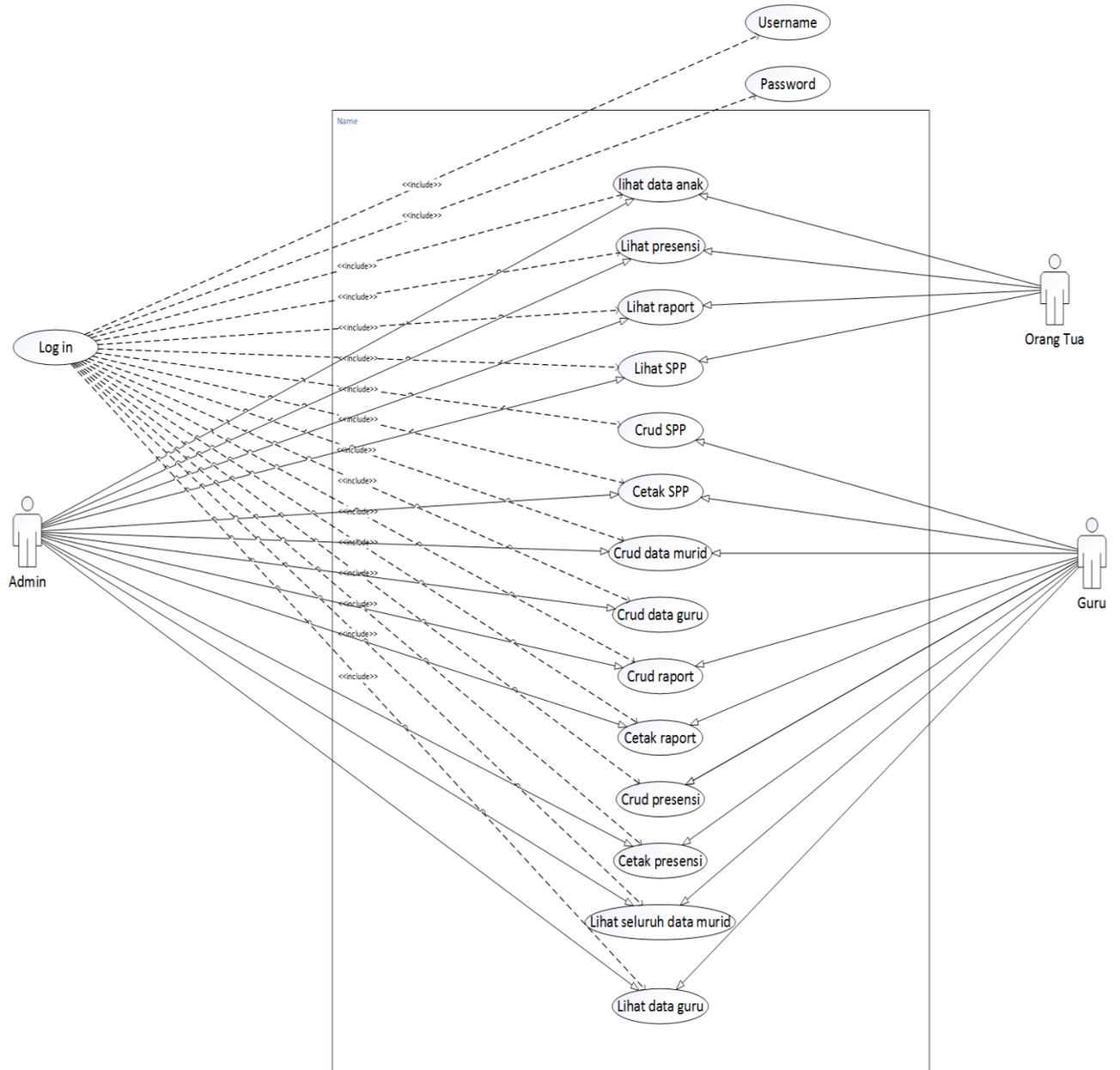
3.5.1 Rancangan Sistem Aplikasi Web

Perancangan sistem aplikasi *web* berupa perencanaan mengenai sistem aktivitas yang ada pada aplikasi *web* dan aplikasi *web* yang akan dibuat dengan metode *Unified Modelling Language (UML)* merupakan sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, membangun dan pendokumentasian dari sebuah sistem pembangunan perangkat lunak berbasis objek, diantaranya menggunakan *use case*, *class diagram*, *ER Diagram* dan *Activity Diagram*. Perancangan tersebut dijelaskan sebagai berikut:

1. Use Case Diagram

Use Case Diagram merupakan teknik pemodelan untuk menjelaskan setiap aktifitas yang mungkin terjadi dengan menggunakan sudut pandang dari aktor sebagai pengguna sistem dan sudut pandang administrator.

Gambaran *Use Case Diagram* yang digunakan pada aplikasi *web* dapat dilihat pada gambar 3.3.



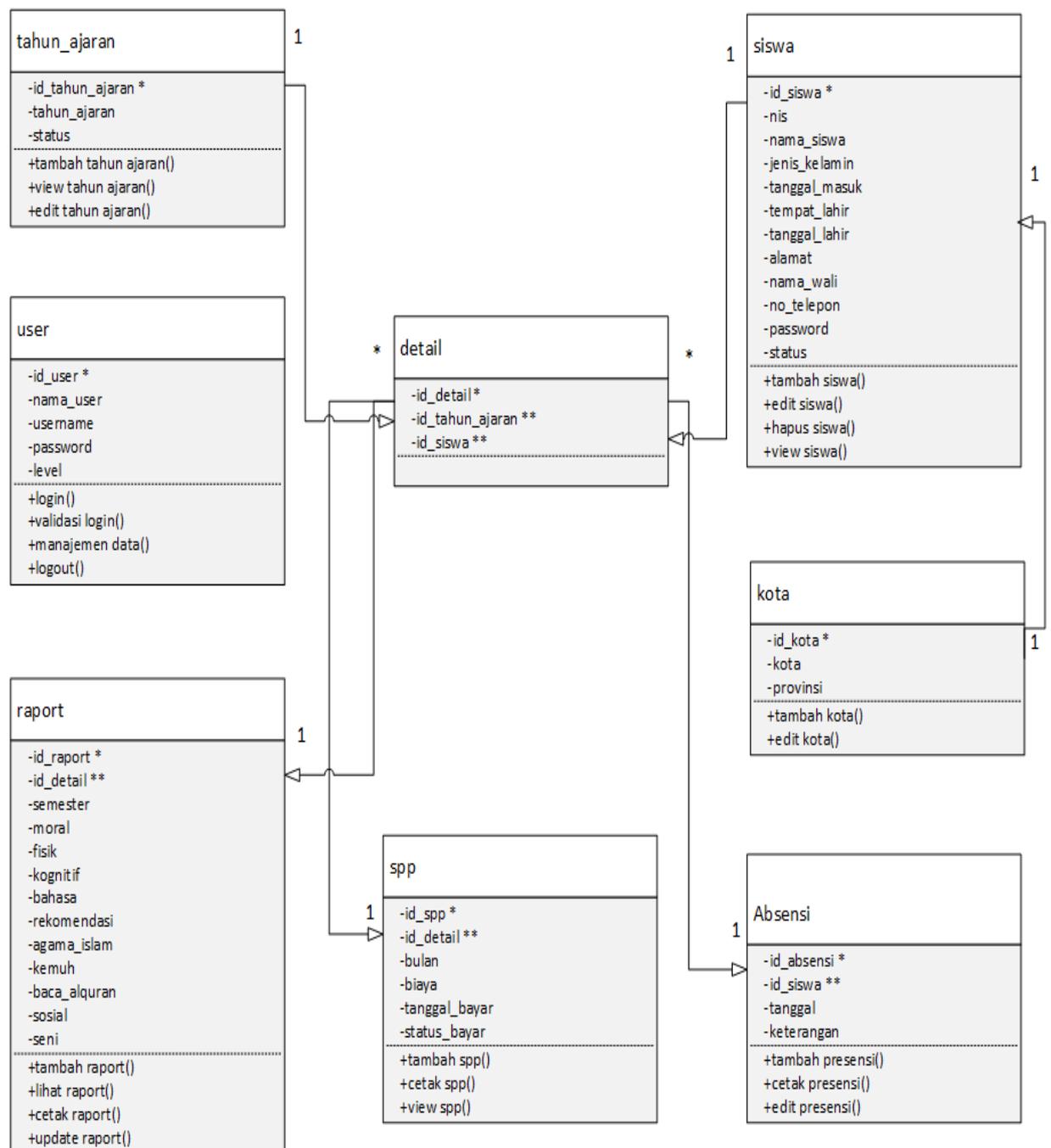
Gambar 3.3 Use Case Diagram Aplikasi Web

Berikut penjelasan tentang gambar 3.3 :

1. Terdapat 3 aktor pada use case diagram aplikasi admin, guru dan murid
2. Admin dapat mengakses semua menu aplikasi web.
3. Guru dapat mengakses data – data yang ada pada murid.
4. Murid hanya bisa melihat dan me-*print* data nya.

2. Class Diagram

Gambaran *Class Diagram* yang digunakan pada aplikasi *web* dapat dilihat pada gambar 3.4.



Gambar 3.4 *Class Diagram* Aplikasi

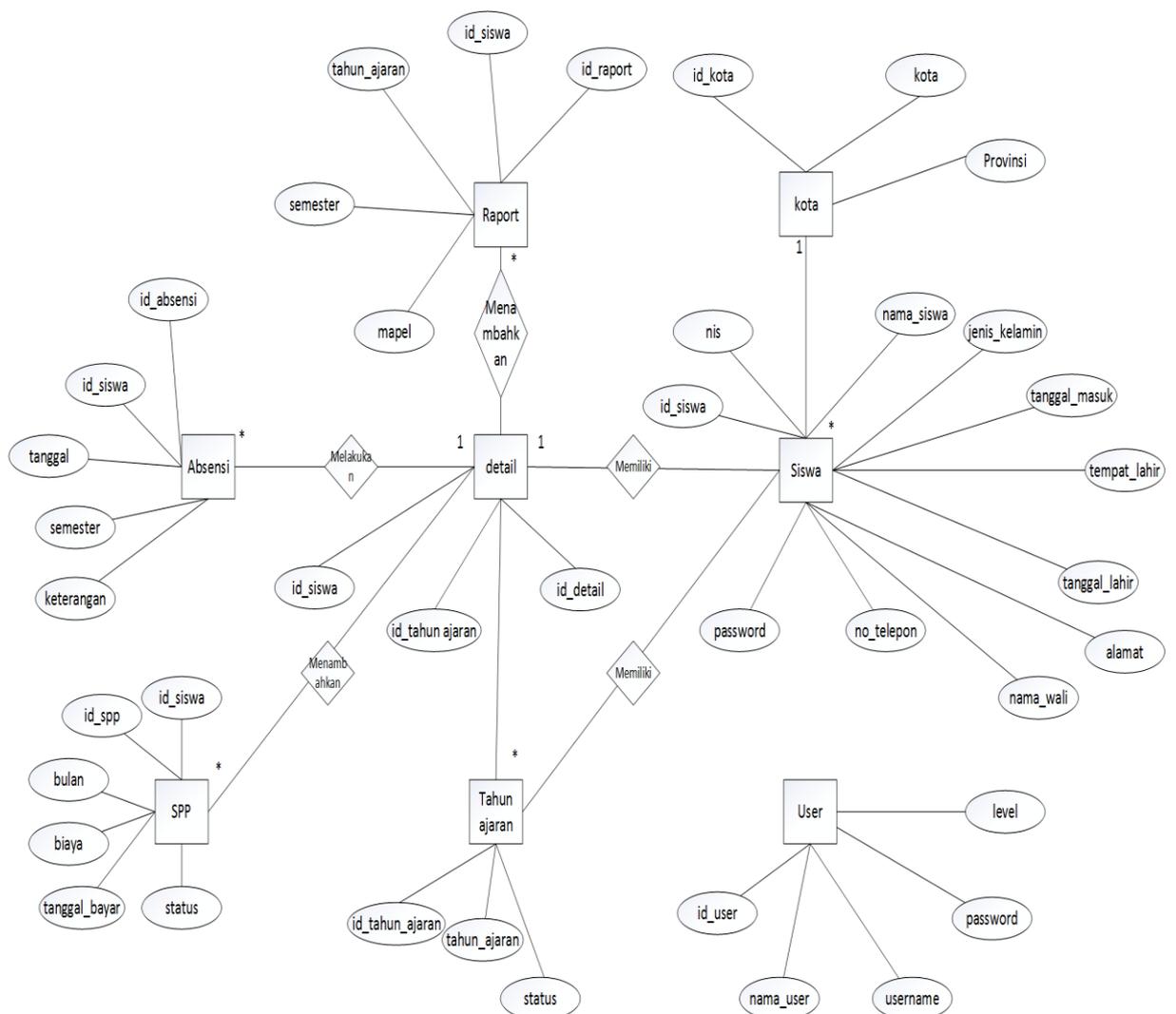
Berikut penjelasan tentang gambar 3.4:

Penjelasan fungsi

1. class absensi
berfungsi untuk memasukan dan menampilkan data absensi siswa yang digunakan dalam aplikasi web.
2. class raport
berfungsi untuk memasukan dan menampilkan data raport siswa dan untuk menampilkan perkembangan siswa tersebut.
3. Class siswa
berfungsi untuk memasukan dan menampilkan data siswa yang di gunakan untuk login dan juga biodata dari siswa.
4. Class SPP
berfungsi untuk memasukan dan menampilkan data pembayaran yang telah dibayarkan maupun belum di bayarkan.
5. Class detail
berfungsi untuk menyimpan data siswa dengan tahun ajaran.
6. Class kota
berfungsi untuk memasukkan dan menampilkan data kota untuk keperluan biodata dari user.
7. Class tahun ajaran
berfungsi untuk memasukkan dan menampilkan tahun ajaran yg terintegrasi dengan class detail dan juga mengkategorisasikan data siswa berdasarkan tahun ajaran.
8. Class user
berfungsi untuk memasukkan dan menampilkan data admin dan guru yang di gunakan untuk login.

3. Entity Relationship Diagram(ERD)

Untuk pengembangan aplikasi pemodelan basis data yang digunakan adalah *ER Diagram*. *ER Diagram* merupakan diagram yang digunakan untuk merancang basis data dan memperlihatkan relasi antar entitas atau object dengan atributnya. Tujuan *ER Diagram* ini adalah untuk memberikan gambaran umum tentang sistem yang akan dikembangkan sehingga mempermudah perancangan basis data.



Gambar 3.5 ER Diagram

Gambar 3.5 menunjukkan *Entity Relationship Diagram* dan memiliki 5 buah entitas yaitu:

1. User

Pada entitas user berfungsi untuk menyimpan data *administrator*, guru dan juga berfungsi untuk *login administrator* dan guru.

2. Raport

Pada entitas raport memiliki relasi dengan entitas detail. Yang artinya setiap detail memiliki detail yang berbeda sesuai dengan tahun ajaran.

3. Siswa

pada entitas siswa memiliki relasi dengan entitas user. Yang berfungsi untuk menyimpan data siswa saat *login*.

4. Absensi

pada entitas absensi memiliki relasi dengan entitas detail. Setiap siswa memiliki presensi yang mana di dalam nya terdapat hadir, izin, sakit dan alpa.

5. SPP

pada entitas SPP memiliki relasi dengan entitas detail. Relasi nya adalah *one to many* dimana satu data detail dapat disimpan satu atau lebih pada tabel SPP.

6. Tahun ajaran

pada entitas tahun ajaran memiliki relasi dengan entitas detail. Yang artinya setiap satu data tahun ajaran dapat di simpan satu atau lebih pada tabel detail.

7. Detail

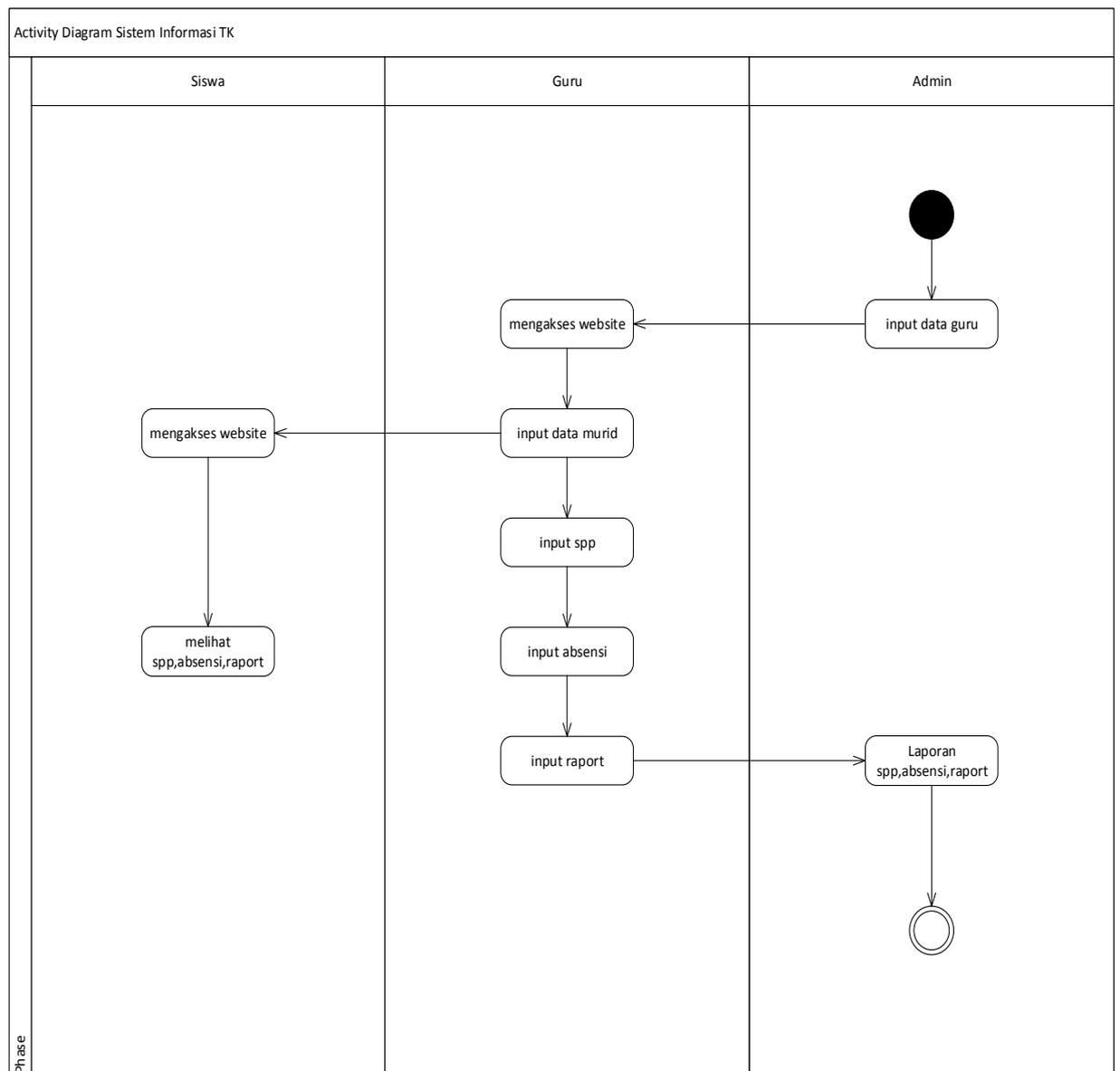
pada entitas detail berisikan id dari siswa, tahun ajaran, dan detail yang di gunakan sebagai kategorisasi siswa berdasarkan tahun ajaran.

8. Kota

pada entitas kota memiliki relasi dengan entitas siswa. Yaitu satu data kota dapat disimpan satu atau lebih pada tabel siswa.

4. Activity Diagram

Berdasarkan *use case* yang telah dibuat sebelumnya maka dapat diperoleh *activity diagram* berdasarkan actor yang terlibat dalam *use case diagram*. *Activity diagram* dalam aplikasi *web* dibagi menjadi tiga bagian yaitu *activity diagram* admin dan *activity diagram* guru dan *activity diagram* siswa.



Gambar 3.6 Activity Diagram

Berikut penjelasan tentang gambar 3.6 :

1. Terdapat alur yang di mulai dari Admin, yaitu admin guru ketika ingin membuat akun harus menghubungi admin terlebih dahulu.
2. Kemudian admin input data guru. Guru dapat mengakses *web*, kemudian guru menginput data siswa, input spp, input absensi, input raport.
3. Kemudian laporan di teruskan ke admin.
4. Siswa mengakses *website*, dan bisa melihat spp, absensi, raport.

3.5.1 Rancangan Antarmuka

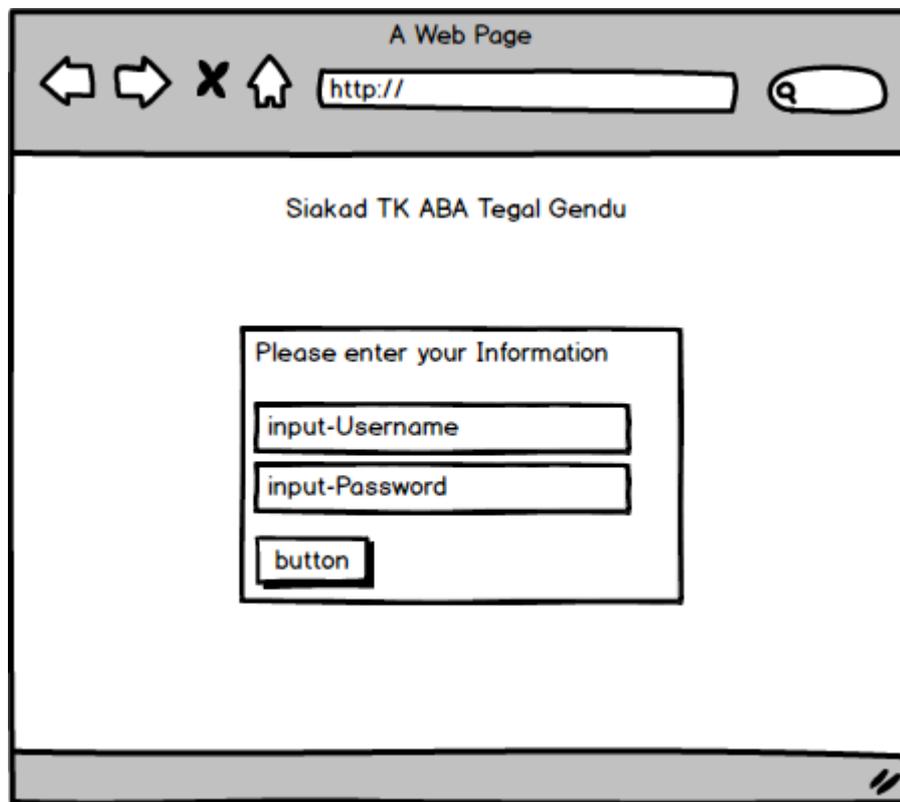
Rancangan antarmuka atau tampilan yang akan dirancang harus dapat memberikan gambaran dari setiap bagian dalam aplikasi *web*. Rancangan antarmuka ini menjelaskan keterkaitan setiap halaman menu dan penjelasan cara kerja dari setiap menu dari aplikasi *web*. Rancangan antarmuka ini menjadi dasar untuk membuat tampilan pada aplikasi *web* yang akan dibuat.

3.5.2.1 Rancangan Antarmuka Admin

Rancangan antarmuka Admin adalah rancangan antarmuka dari sisi admin yang mengunjungi *web* ini. Admin tetap melakukan proses *login*, kemudian admin dapat mengakses *web* ini.

1. Rancangan antarmuka halaman *login*.

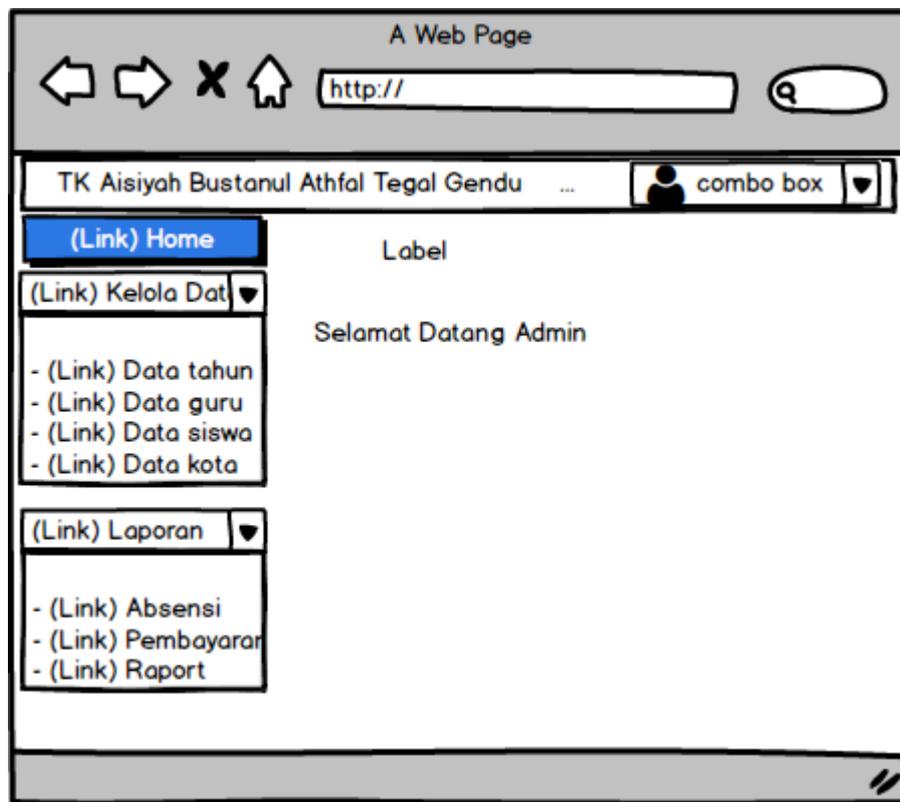
Rancangan antarmuka ini menampilkan *textbox* untuk mengisi *username* dan *password*. Fungsi halaman ini adalah melakukan login untuk admin. Gambaran untuk rancangan antarmuka halaman *login* dapat dilihat pada gambar 3.7



Gambar 3.7 Halaman *Login*

2. Rancangan antarmuka halaman utama.

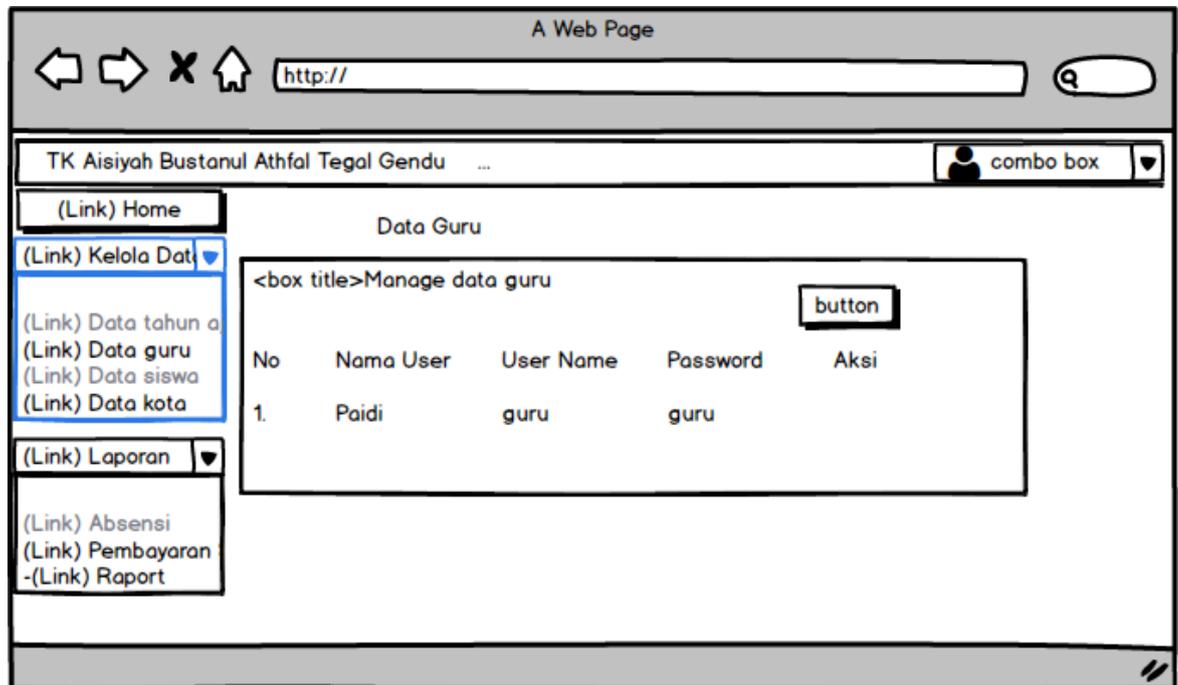
Rancangan antarmuka ini adalah rancangan aplikasi *web* setelah proses *login*. menu dari halaman ini antara lain home, kelola data, dan laporan. Gambaran rancangan antarmuka halaman ini dapat dilihat pada gambar 3.8



Gambar 3.8 Halaman Utama

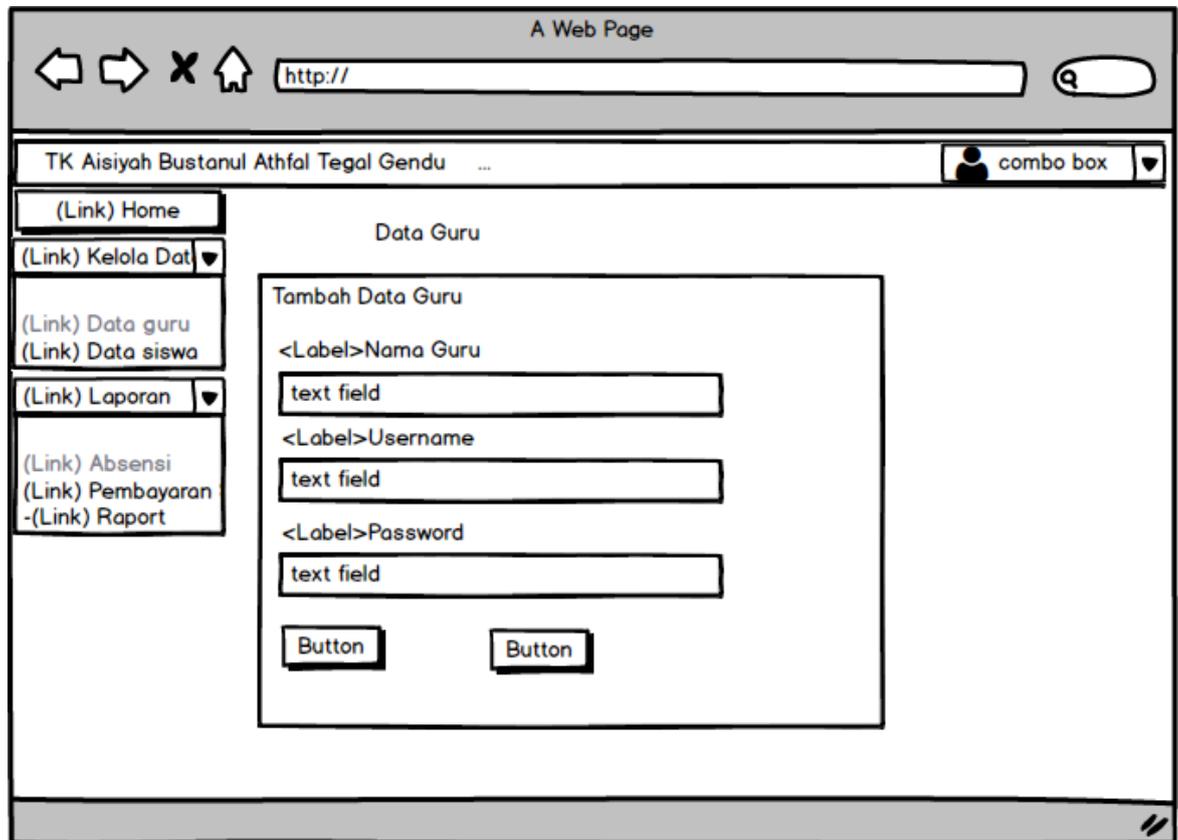
3. Rancangan antarmuka kelola data guru

Rancangan antarmuka ini menampilkan data dari guru yang ada dalam daftar guru, sehingga admin dapat melihat data dari guru yang terdapat dalam halaman ini. gambaran rancangan antarmuka ini dapat dilihat pada gambar 3.9



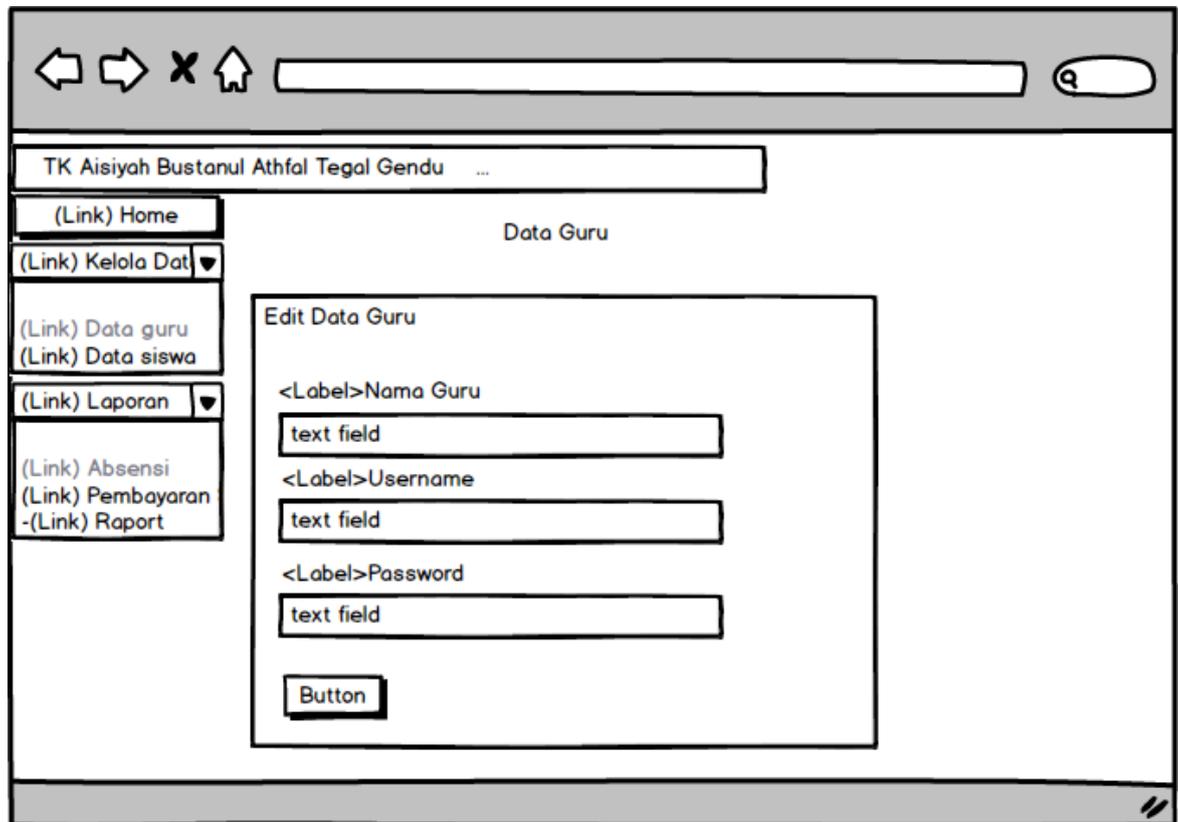
Gambar 3.9 Halaman Kelola Data Guru

Pada gambar ini terdapat data guru yang berisikan *username* dan *password* untuk *login*.



Gambar 3.10 Halaman Tambah Data Guru

Pada gambar ini terdapat menu untuk menambah data guru dan juga untuk proses *login* ke *website*.



Gambar 3.11 Halaman *Edit* Data Guru

Pada gambar ini adalah menu untuk *edit* data guru, yang merupakan untuk merubah nama guru, *username* dan *password*.