

BAB III

METODOLOGI PENELITIAN

3.1 Alat dan Bahan Penelitian

Dalam penelitian ini, dibutuhkan beberapa alat dan bahan untuk mendukung berjalannya perancangan serta proses implementasi aplikasi web sebagai berikut:

3.1.1 Alat

Alat yang harus digunakan harus berupa perangkat keras (*Hardware*) dan perangkat lunak (*Software*). Berikut ini adalah alat – alat yang digunakan untuk proses pembangunan aplikasi web:

- a. Interl® Core™ i5 *processor*,
- b. 8.00 GB RAM,
- c. 64-bit *Operating System*,
- d. Windows 10 *pro*. *Printer* dokumen untuk mencetak laporan.
- e. *Microsoft Visio* 2013
- f. Xampp
- g. MySql
- h. *Sublime Text*
- i. *Jquery*

3.1.2 Bahan

Dalam penelitian ini, bahan yang dibutuhkan adalah:

- a. Data yang didapatkan melalui studi literatur berdasarkan penelitian sebelumnya yang masih memiliki keterkaitan dengan *website* yang dikembangkan. Dari data yang diperoleh, maka didapatkan dari aplikasi pada saat pengembangan aplikasi.
- b. Model dan Konsep perekapan keuangan Program Studi Teknik Informatika Universitas Muhammadiyah Yogyakarta sebelumnya.

3.2 Teknik Pengumpulan Data

3.2.1 Wawancara

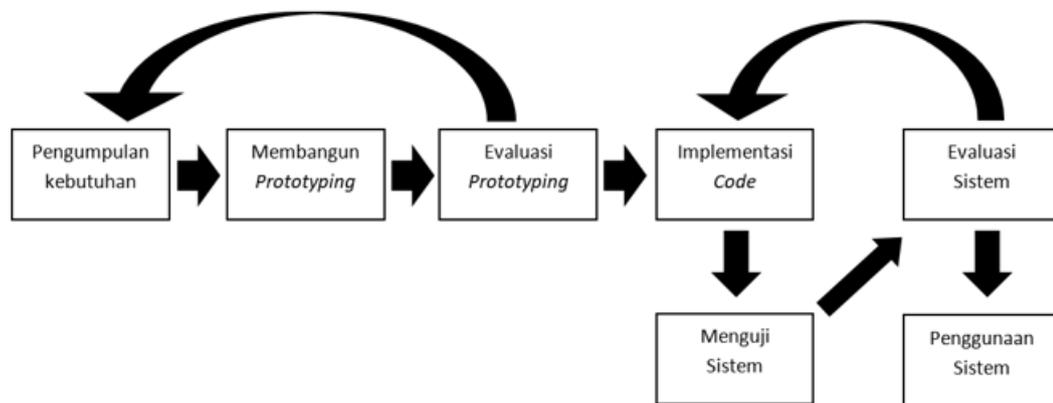
Wawancara adalah proses percakapan dengan tujuan untuk mendapatkan gambaran atau konsep tentang penelitian ini. Percakapan dilakukan oleh penulis yang mengajukan beberapa pertanyaan dan narasumber akan memberikan jawaban atas pertanyaan tersebut.

Proses wawancara ini dilakukan secara langsung kepada dosen pembimbing 1, yaitu dengan Bapak Asroni, S.T., M.Eng. Topik yang dibahas adalah mengenai konsep serta data manipulasi untuk proses pembangunan pada aplikasi web rekap keuangan tersebut.

3.2.2 Alur Penelitian

Untuk perancangan dan pembuatan sistem, penulis menggunakan metode *SDLC (Software Development Life Cycle)*. Metode tersebut dipilih untuk melakukan proses pembangunan sistem serta proses *maintenance software* tersebut. Metode *SDLC* yang digunakan oleh penulis adalah model *Prototyping*. *Prototyping* perangkat lunak (*software prototyping*) atau siklus hidup menggunakan *prototyping (life cycle using prototyping)* adalah salah satu metode siklus hidup sistem yang didasarkan pada konsep model bekerja (*working model*). Model pengembangan *prototyping* mempunyai beberapa keunggulan, diantaranya:

- a.** Adanya komunikasi antara *developer* dan *user*. Peran *user* meningkat, karena *user* dapat memberikan masukan setiap saat.
- b.** Sistem dapat dibangun sampai beberapa kali di evaluasi sehingga perubahan *requirement* dapat ditambah maupun diubah seiring pengembangan dilakukan tanpa harus mengulanginya dari awal.
- c.** *user* dapat mengenal aplikasi yang akan dikembangkan dan tidak merasa asing saat *user* menggunakannya dikarenakan metode *SDLC* bersifat *user friendly*.



Gambar 3.1 Konsep *Prototype*

Seperti pada gambar 3.1 dijelaskan ada beberapa tahapan - tahapan dalam konsep *prototyping*. Penjelasan dari tahapan-tahapan tersebut adalah sebagai berikut:

a. Pengumpulan Kebutuhan

Tahapan ini adalah tahap pertama dimana *developer* serta *user* bekerja sama mendefinisikan format perangkat lunak, mengidentifikasi semua kebutuhan, dan alur sistem yang akan dibuat.

b. Membangun *Prototype*

Tahapan ini adalah proses perancangan sistem meliputi perancangan *database* dan *use case* serta design program sementara yang berfokus pada *interface* (contohnya seperti tampilan).

c. Evaluasi *Prototype*

Tahapan ini hanya boleh dilakukan oleh *user* apakah *prototype* yang sudah dibuat telah sesuai dengan apa yang dibutuhkan. Jika sudah sesuai maka dilanjutkan pada langkah berikutnya. Dan jika belum sesuai, maka akan dilakukan revisi dengan mengulang pada langkah **a**, **b**, dan **c**.

d. Impelementasi *code*

Jika telah disepakati *prototype* yang sudah dibuat, maka kemudian diimplementasi dalam bentuk *source code* atau bahasa pemrograman.

e. Menguji sistem

Saat aplikasi sudah selesai dibangun, untuk mengetahui bahwa aplikasi tersebut sudah layak digunakan atau tidak, maka terlebih dahulu harus dilakukan pengujian agar aplikasi terhindar dari *error* atau *bug*.

f. Evaluasi sistem

User akan mengevaluasi apakah sistem tersebut yang sudah selesai dibangun dan sesuai dengan yang diharapkan. Jika iya lanjut ke langkah selanjutnya, namun jika tidak harus mengulangi pada langkah **d** dan **e**.

g. Menggunakan sistem

Aplikasi yang telah lulus standar pengujian dan disetujui oleh *user* maka aplikasi sudah siap untuk digunakan.

3.3 Analisis Penelitian

3.3.1 Analisis Kebutuhan

Analisis kebutuhan dapat melalui diskusi bersama dosen pembimbing penelitian ini. Berikut adalah hasil analisis kebutuhan pada aplikasi web rekap keuangan Program Studi Teknik Informatika Universitas Muhammadiyah Yogyakarta:

- a.** Terdapat fitur *login* sebagai standar keamanan dan hanya admin yang boleh mengakses aplikasi tersebut.
- b.** Terdapat fitur *register* dan ketika user ataupun orang lain ingin melakukan registrasi melalui *register*, mereka harus mendapatkan *approvalment* atau persetujuan dari pihak admin dikarenakan fitur *register* telah dilindungi oleh fitur *login*.
- c.** Terdapat menu *profile* yang berisi submenu *About*, *History*, serta *Vision and Mision*.
- d.** Terdapat menu *Akademics* yang berisi submenu *Lecturer* dan *Organization*.

- e. Terdapat data master *entry* kategori. Dalam menu ini terdapat fitur - fitur yang memungkinkan admin untuk melakukan tambah data, ubah data, hapusa data, cari data, serta cetak data sesuai kebutuhan.
- f. Terdapat data master *entry* kegiatan. Dalam menu ini terdapat fitur - fitur yang memungkinkan admin untuk melakukan tambah data, ubah data, hapusa data, cari data, serta cetak data sesuai kebutuhan.
- g. Terdapat data master *entry* tahun ajaran. Dalam menu ini terdapat fitur - fitur yang memungkinkan admin untuk melakukan tambah data, ubah data, hapusa data, cari data, serta cetak data sesuai kebutuhan
- h. Terdapat data master *entry* laporan. Dalam menu ini terdapat fitur-fitur yang memungkinkan admin untuk melakukan tambah data, ubah data, hapus data, cari data, cetak data, serta unduh data berupa foto atau gambar sesuai kebutuhan.
- i. Terdapat fitur *logout* yang membantu admin untuk keluar dari aplikasi tersebut.

3.3.2 Analisis Pengguna

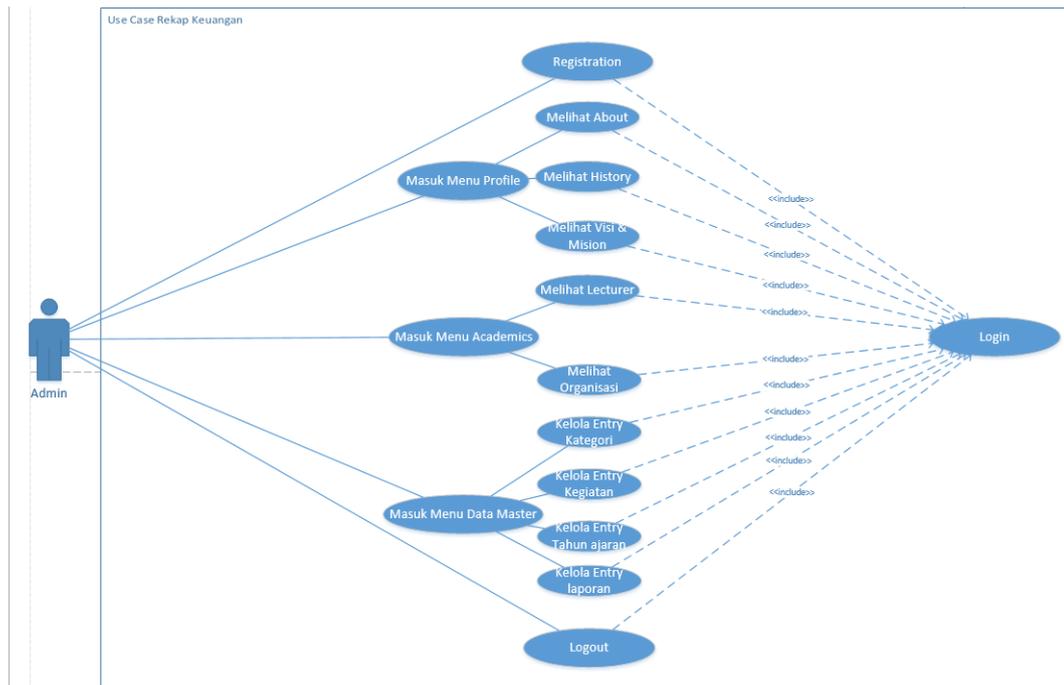
Aplikasi web yang ditelah layak pakai hanya boleh digunakan oleh satu pengguna, yaitu admin. Admin tersebut memiliki hak penuh untuk mengakses aplikasi tersebut. Admin dapat melakukan perekapan data keuangan serta membuat laporan melalui aplikasi tersebut.

3.4 Rancangan Sistem dan Basis data

Sebelum membangun aplikasi web Rekapitulasi Laporan Keuangan Program Studi Teknik Informatika Universitas Muhammadiyah Yogyakarta, ada kalanya perlu dibuat perencanaan dalam bentuk rancangan diagram. Diagram ini akan dijadikan sebuah acuan sebagai alur kerjanya aplikasi, serta *database*. Metode yang digunakan dalam perancangan website rekap keuangan adalah *United Markup Language* (UML). Metode UML yang dipakai dalam pengembangan aplikasi web, yaitu model diagram *Use Case Diagram*, *ER Diagram*, *Class Diagram*, dan *Activity Diagram*. Berikut ini adalah penjelasan dari rancangan diagram aplikasi web Rekapitulasi Laporan Keuangan Program Studi Teknik Informatika Universitas Muhammadiyah Yogyakarta.

3.4.1 Use Case Diagram

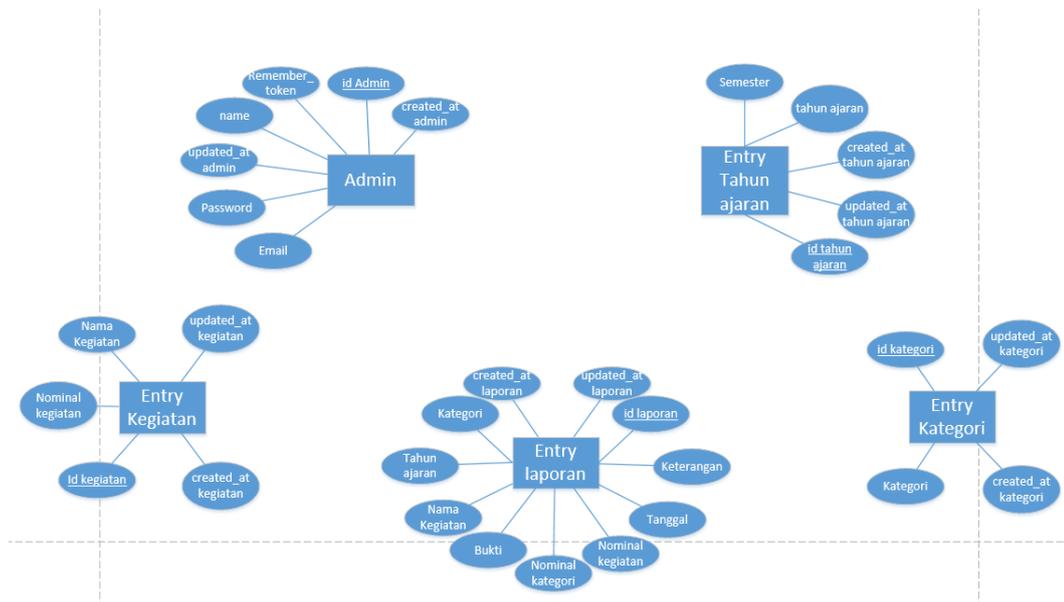
Berikut ini adalah gambaran model *Use Case Diagram* yang akan digunakan pada aplikasi web Rekapitulasi Laporan Keuangan Program Studi Teknik Informatika Universitas Muhammadiyah Yogyakarta.



Gambar 3.2 Use Case Diagram

Seperti yang kita lihat pada Gambar 3.2 memperlihatkan bahwa bagaimana relasi antara *actor* dengan *use case*. Pada gambar diatas menjelaskan bahwa *actor* tersebut hanyalah admin yang mempunyai hak akses penuh terhadap aplikasi tersebut. Tetapi, sebelum bisa melakukan tugas untuk mengelola data yang ada didalam aplikasi web tersebut, ada *include* yang mengharuskan *actor* untuk *login* kedalam system. Setelah *actor* berhasil melakukan akses baik itu *login*,. Maka *actor* akan diarahkan sebuah isi dari aplikasi web tersebut seperti *Profile*, *Academics*, *Data Master*, dan lain-lain.

3.4.2 ER Diagram

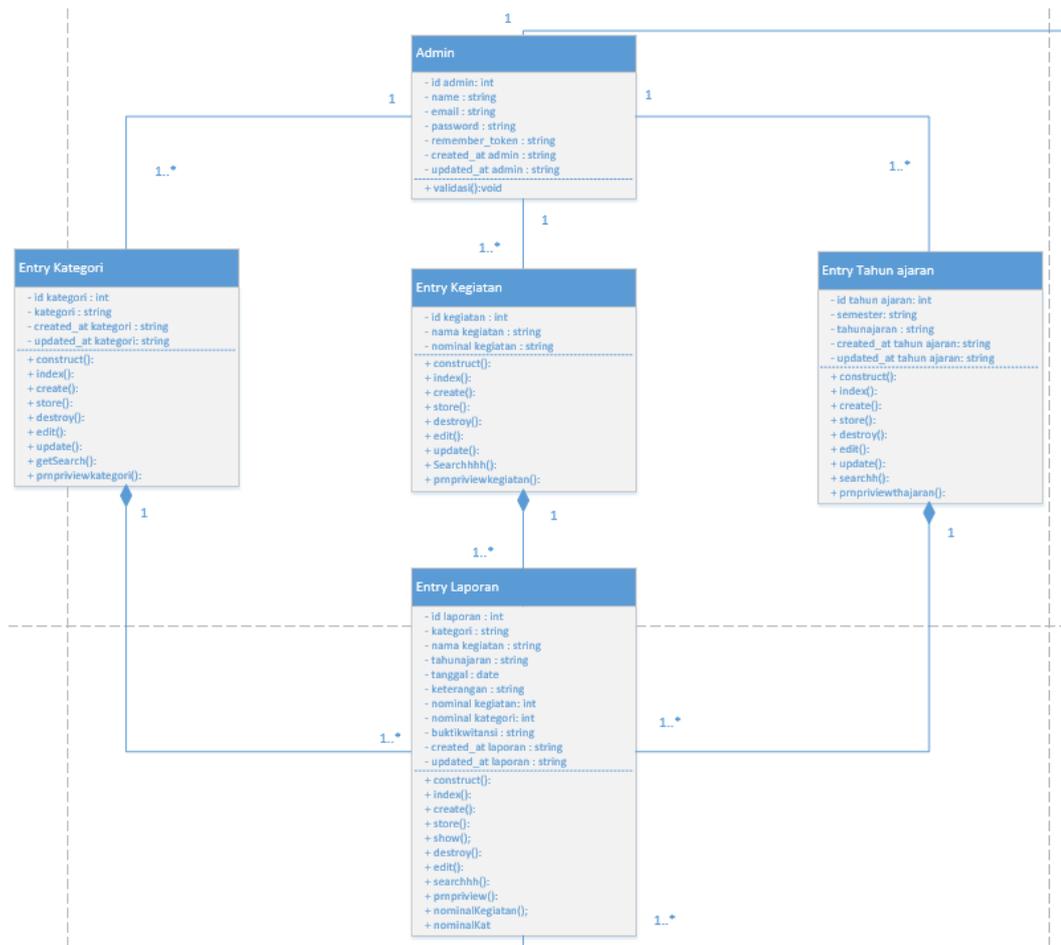


Gambar 3.3 ER Diagram

Gambar 3.3 diatas adalah model *ER Diagram* yang menunjukkan *entity* atau entitas, atribut, serta dengan relasinya. Entitas ini yang akan selanjutnya menjadi rancangan tabel pada *database* yang digunakan. Entitas diatas memiliki relasi antar satu dengan yang lainnya. Berikut adalah penjelasan relasi yang ada pada gambar diatas:

- Entitas Admin dapat berdiri sendiri dikarenakan tidak memiliki atribut pendukung dari entitas yang lain.
- Entitas *Entry* Kategori dapat berdiri sendiri dikarenakan tidak memiliki atribut pendukung dari entitas yang lain.
- Entitas *Entry* Kegiatan dapat berdiri sendiri dikarenakan tidak memiliki atribut pendukung dari entitas yang lain.
- Entitas *Entry* Tahun ajaran dapat berdiri sendiri dikarenakan tidak memiliki atribut pendukung dari entitas yang lain.
- Entitas *Entry* Laporan dapat berdiri sendiri dikarenakan tidak memiliki atribut pendukung dari entitas yang lain.

3.4.3 Class Diagram



Gambar 3.4 Class Diagram

Gambar 3.4 diatas adalah model *class diagram* yang menunjukkan class, atribut, *operation* atau *method*, serta dengan relasinya. *Class diagram* yang ada pada sistem dibuat berupa sebuah *controller* dari aplikasi web yang dibuat, dimana setiap *controller* memiliki fungsi yang sama dengan *operation* atau *method* pada *class diagram*, yaitu:

- a. *construct()*: dimaksudkan untuk *authentication* pada saat masuk ke page tersebut.
- b. *index()*: dimaksudkan untuk menampilkan data dari *database*.
- c. *create()*: dimaksudkan untuk menambahkan data.
- d. *store()*: dimaksudkan untuk mengkonfirmasi data yang akan ditambahkan
- e. *destroy()*: dimaksudkan untuk menghapus data.

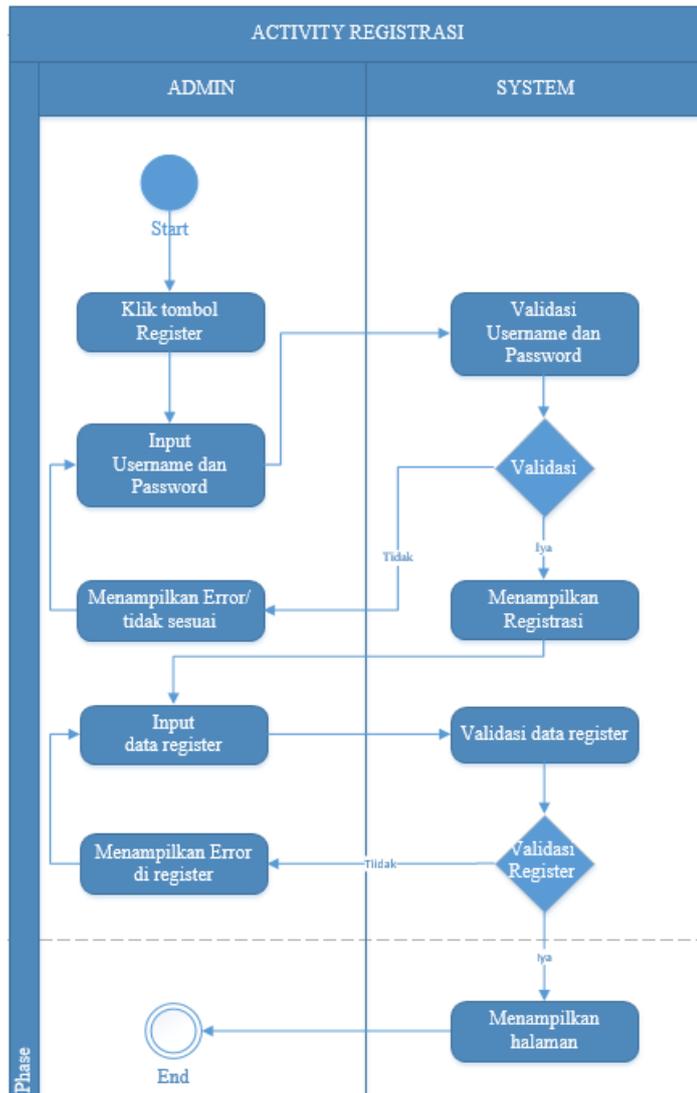
- f. *edit()*: dimaksudkan untuk mengubah data.
- g. *update()*: dimaksudkan untuk mengkonfirmasi data yang akan dibuat.
- h. *getSearch()*: atau *searchh()*: atau *searchhh()*: atau *searchhhh()*: dimaksudkan untuk mencari data berdasarkan atribut.
- i. *prnpriview()*: atau *prnpriviewkategori()*: atau *prnpriviewkegiatan()*: atau *prnpriviewthajaran()*: dimaksudkan untuk mencetak data.
- j. *Show()*: dimaksudkan untuk menampilkan rincian laporan.
- k. *nominalKegiatan()*: dimaksudkan untuk memanggil nilai nominalkegiatan.
- l. *nominalKat()*: dimaksudkan untuk memanggil nilai nominalkategori.

berikut ini adalah penjelasan terhadap relasi yang ada pada gambar 3.4

- a. Pada *class entry* laporan terdapat relasi *composition* yang menghubungkan antar *class entry* kategori dan *class entry* tahun ajaran dikarenakan *class entry* laporan mempunyai atribut yang bergantung pada *class entry* kategori serta *class entry* tahun ajaran , yaitu atribut kategori dan atribut tahun ajaran.
- b. Pada *class admin* terdapat relasi *association* dengan nilai *multiplicity one only* yang menghubungkan ke *class entry* kategori dengan nilai *multiplicity one or more* yang artinya satu admin dapat mengolah satu atau banyak data kategori.
- c. Pada *class admin* terdapat relasi *association* dengan nilai *multiplicity one only* yang menghubungkan ke *class entry* kegiatan dengan nilai *multiplicity one or more* yang artinya satu admin dapat mengolah satu atau banyak data kegiatan.
- d. Pada *class admin* terdapat relasi *association* dengan nilai *multiplicity one only* yang menghubungkan ke *class entry* tahun ajaran dengan nilai *multiplicity one or more* yang artinya satu admin dapat mengolah satu atau banyak data tahun ajaran.
- e. Pada *class entry* kategori terdapat relasi *composition* dengan nilai *multiplicity one only* yang menghubungkan antar *class entry* laporan dengan nilai *multiplicity one or more* yang artinya satu data kategori dapat diakses oleh satu atau banyak untuk data laporan.
- f. Pada *class entry* kegiatan terdapat relasi *composition* dengan nilai *multiplicity one only* yang menghubungkan antar *class entry* laporan dengan nilai

halaman *home*. Dan jika validasi salah, maka admin ada diberikan berupa notifikasi bahwa autentikasi salah.

3.4.5 Activity Diagram Register

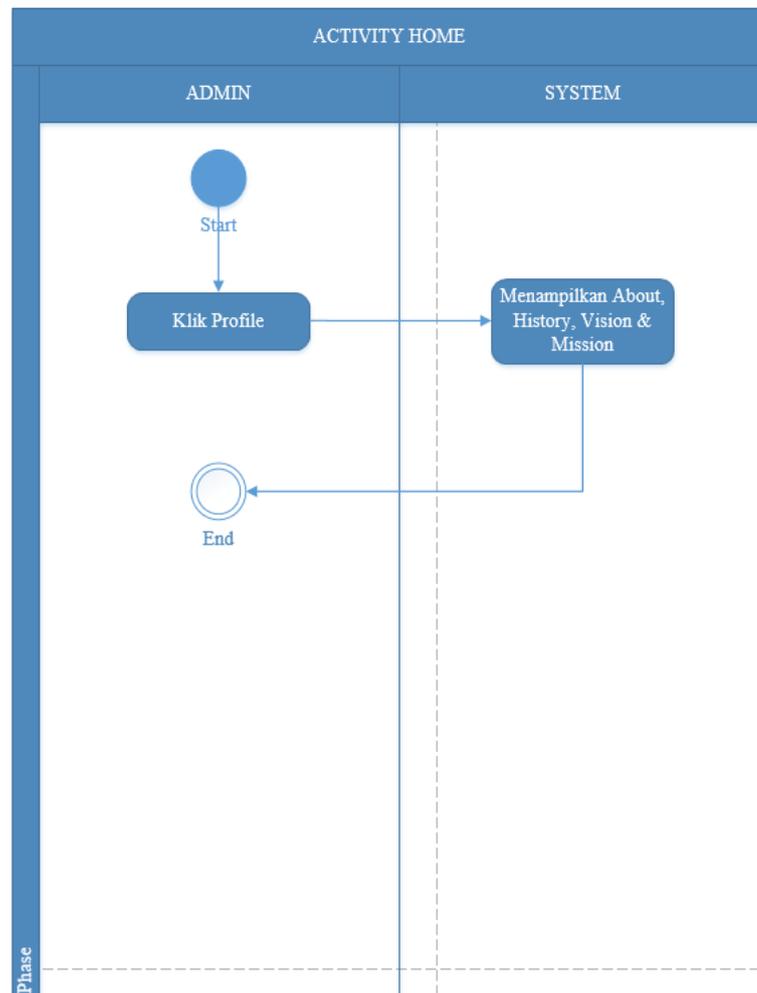


Gambar 3.6 Activity Diagram Register

Pada gambar 3.6 merupakan model dari *Activity Diagram* untuk *register*. Pada proses awal, sama seperti halnya *Activity Diagram login*, admin akan diarahkan untuk melakukan *login* terlebih dahulu dikarenakan harus mendapatkan *approval* dari pihak admin yang sudah terdaftar. Untuk itu, sistem akan mengidentifikasi admin dengan meminta email serta password untuk proses

otentikasi. Setelah admin mengisi autentikasi dengan email dan password, selanjutnya sistem akan membaca hasil autentikasi tersebut. Pada diagram ini, terdapat simbol *decision* sebagai validasi yang akan memberikan *feedback* kepada admin. Jika validasi benar, maka admin akan diarahkan ke halaman *register*. Dan jika validasi salah, maka admin akan diberikan berupa notifikasi bahwa autentikasi salah. Setelah berhasil masuk ke halaman *register*, admin akan diminta mengisi data untuk proses registrasi. Setelah admin mengisi data, selanjutnya sistem akan membaca hasil data tersebut. Pada diagram ini, terdapat simbol *decision* sebagai validasi yang akan memberikan *feedback* kepada admin. Jika validasi benar, maka admin akan diarahkan ke halaman *home* sebagai tanda bahwa proses registrasi telah berhasil. Dan jika validasi salah, maka admin akan diberikan berupa notifikasi bahwa autentikasi salah dan admin akan tetap berada di halaman *register*.

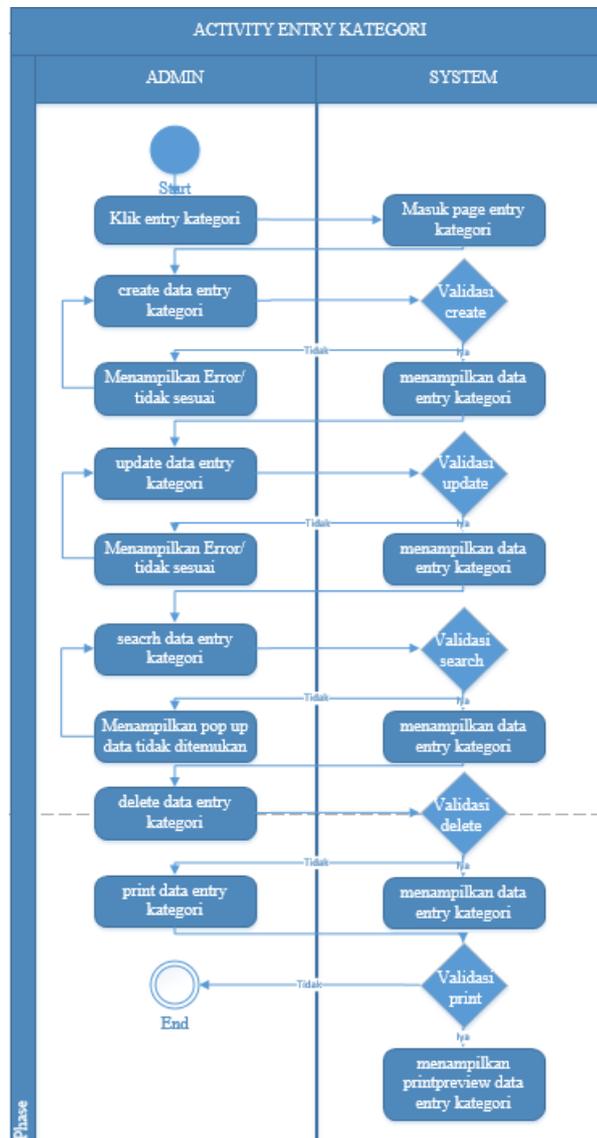
3.4.6 Activity Diagram Home



Gambar 3.7 Activity Diagram Home

Pada gambar 3.7 merupakan model dari *Activity Diagram* untuk *home*. Pada proses awal, admin akan memulai dengan mengklik tombol atau *Dropdown profile* terlebih dahulu dan selanjutnya sistem akan menampilkan *list dropdown* dari beberapa *home page* seperti *About, History, dan Vision & Mission*.

3.4.7 Activity Diagram Entry Kategori



Gambar 3.8 Activity Diagram Entry Kategori

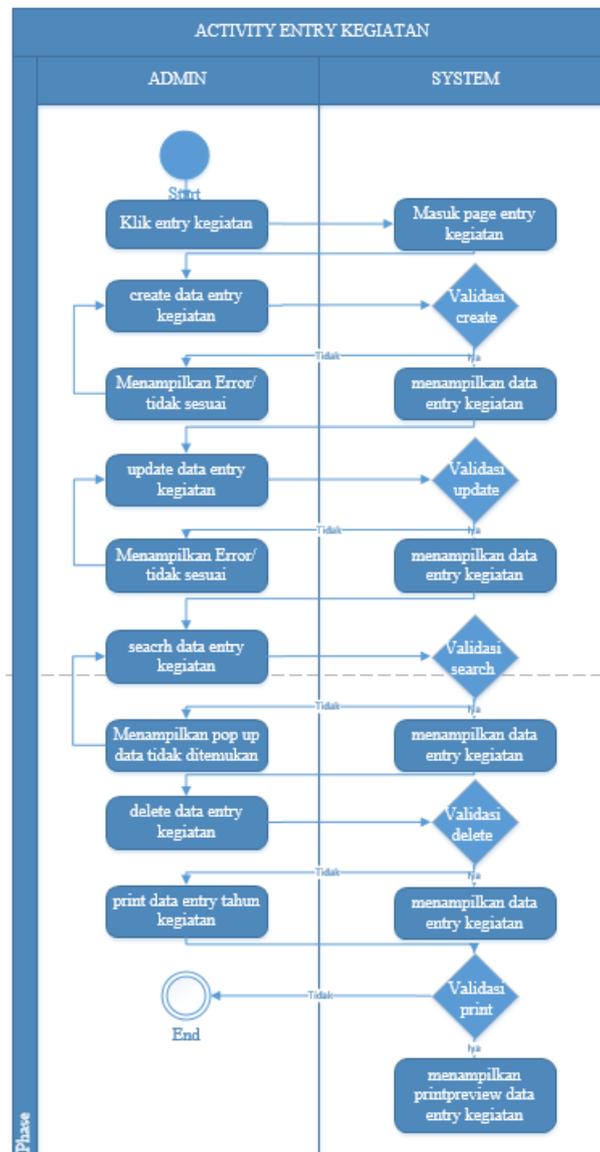
Pada gambar 3.8 merupakan model dari *Activity Diagram* untuk *entry kategori*. Pada proses awal, admin akan memulai dengan mengklik tombol *entry kategori* terlebih dahulu dan selanjutnya sistem akan menampilkan halaman *entry kategori*. Pada diagram ini admin akan melakukan aktivitas pengolahan data sebagai berikut:

- a. Admin akan melakukan proses *create data* dengan *decision iya* dan *tidak* yang artinya jika admin berhasil (*iya*) melakukan proses *create data*, maka sistem

akan mengarahkan admin ke halaman *entry* kategori untuk melihat data yang berhasil ditambahkan, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.

- b.** Admin akan melakukan proses *update data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *update data*, maka sistem akan mengarahkan admin ke halaman *entry* kategori untuk melihat data yang berhasil diubah, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.
- c.** Admin akan melakukan proses *search data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *search data*, maka sistem akan mengarahkan admin ke halaman *entry* kategori untuk melihat data yang berhasil dicari, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan tidak ditemukan.
- d.** Admin akan melakukan proses *delete data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *delete data*, maka sistem akan mengarahkan admin ke halaman *entry* kategori, namun jika admin (tidak) melakukan proses delete data, admin bisa langsung melakukan proses print data untuk membuat laporan rekap keuangan.
- e.** Admin akan melakukan proses *print data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *print data*, maka sistem akan mengarahkan admin ke halaman *print priview* untuk melihat data yang akan dicetak, namun jika admin (tidak) melakukan proses *print data*, maka admin dapat mengakhiri atau menyelesaikan proses pengolahan data pada model *activity diagram* dengan simbol *end*.

3.4.8 Activity Diagram Entry Kegiatan

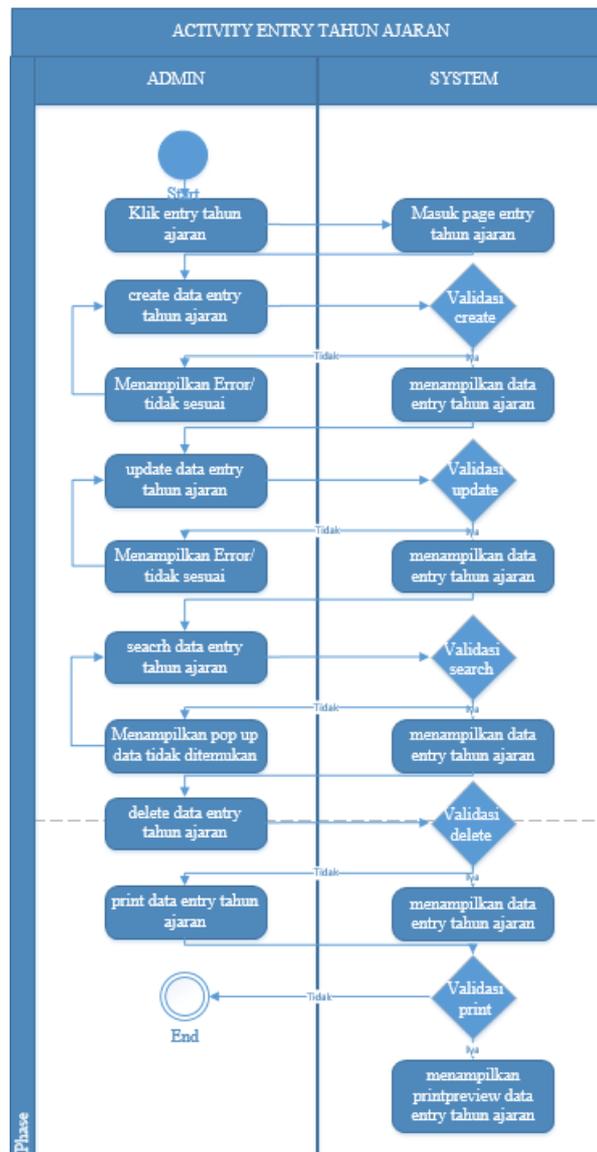


Gambar 3.9 Activity Diagram Entry Kegiatan

Pada gambar 3.9 merupakan model dari *Activity Diagram* untuk *entry* kegiatan. Pada proses awal, admin akan memulai dengan mengklik tombol *entry* kegiatan terlebih dahulu dan selanjutnya sistem akan menampilkan halaman *entry* kegiatan. Pada diagram ini admin akan melakukan aktivitas pengolahan data sebagai berikut:

- f. Admin akan melakukan proses *create data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *create data*, maka sistem akan mengarahkan admin ke halaman *entry* kegiatan untuk melihat data yang berhasil ditambahkan, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.
- g. Admin akan melakukan proses *update data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *update data*, maka sistem akan mengarahkan admin ke halaman *entry* kegiatan untuk melihat data yang berhasil diubah, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.
- h. Admin akan melakukan proses *search data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *search data*, maka sistem akan mengarahkan admin ke halaman *entry* kegiatan untuk melihat data yang berhasil dicari, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan tidak ditemukan.
- i. Admin akan melakukan proses *delete data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *delete data*, maka sistem akan mengarahkan admin ke halaman *entry* kegiatan, namun jika admin (tidak) melakukan proses delete data, admin bisa langsung melakukan proses print data untuk membuat laporan rekap keuangan.
- j. Admin akan melakukan proses *print data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *print data*, maka sistem akan mengarahkan admin ke halaman *print preview* untuk melihat data yang akan dicetak, namun jika admin (tidak) melakukan proses *print data*, maka admin dapat mengakhiri atau menyelesaikan proses pengolahan data pada model *activity diagram* dengan simbol *end*.

3.4.9 Activity Diagram Entry Tahun ajaran

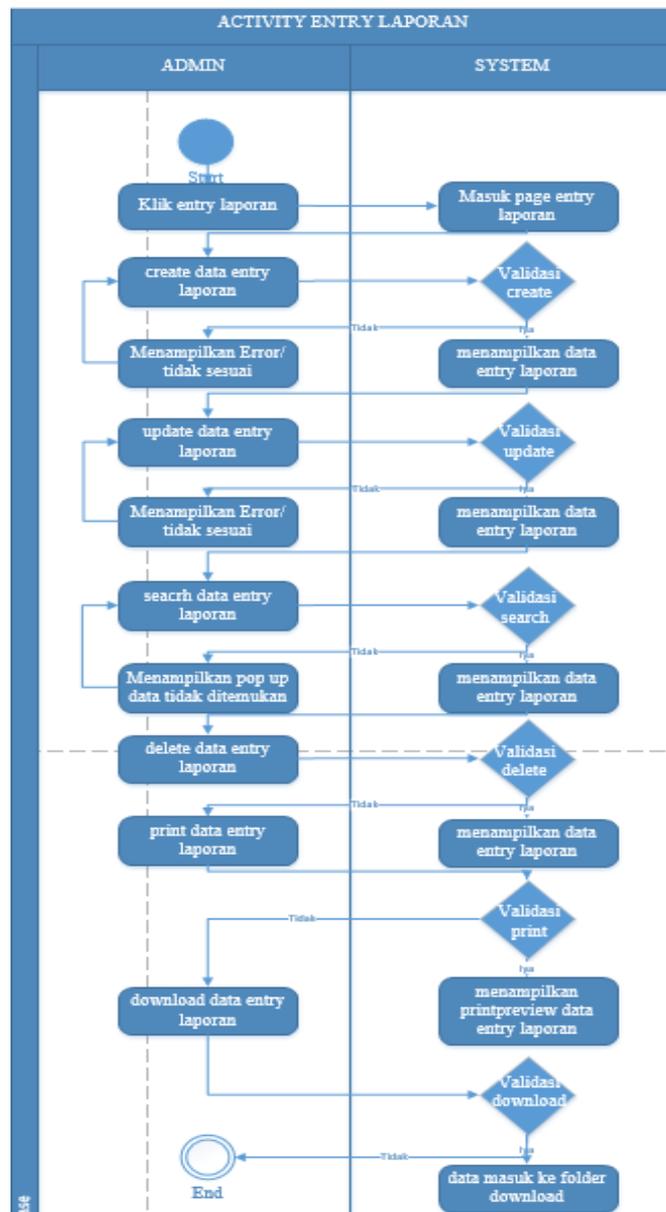


Gambar 3.10 Activity Diagram Entry Tahun Ajaran

Pada gambar 3.10 merupakan model dari *Activity Diagram* untuk *entry* tahun ajaran. Pada proses awal, admin akan memulai dengan mengklik tombol *entry* tahun ajaran terlebih dahulu dan selanjutnya sistem akan menampilkan halaman *entry* tahun ajaran. Pada diagram ini admin akan melakukan aktivitas pengolahan data sebagai berikut:

- a. Admin akan melakukan proses *create data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *create data*, maka sistem akan mengarahkan admin ke halaman *entry* tahun ajaran untuk melihat data yang berhasil ditambahkan, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.
- b. Admin akan melakukan proses *update data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *update data*, maka sistem akan mengarahkan admin ke halaman *entry* tahun ajaran untuk melihat data yang berhasil diubah, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.
- c. Admin akan melakukan proses *search data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *search data*, maka sistem akan mengarahkan admin ke halaman *entry* tahun ajaran untuk melihat data yang berhasil dicari, namun jika proses gagal (tidak), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan tidak ditemukan.
- d. Admin akan melakukan proses *delete data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *delete data*, maka sistem akan mengarahkan admin ke halaman *entry* tahun ajaran, namun jika admin (tidak) melakukan proses delete data, admin bisa langsung melakukan proses print data untuk membuat laporan rekap keuangan.
- e. Admin akan melakukan proses *print data* dengan *decision* **iya** dan **tidak** yang artinya jika admin berhasil (iya) melakukan proses *print data*, maka sistem akan mengarahkan admin ke halaman *print preview* untuk melihat data yang akan dicetak, namun jika admin (tidak) melakukan proses *print data*, maka admin dapat mengakhiri atau menyelesaikan proses pengolahan data pada model *activity diagram* dengan simbol *end*.

3.4.10 Activity Diagram Entry Laporan



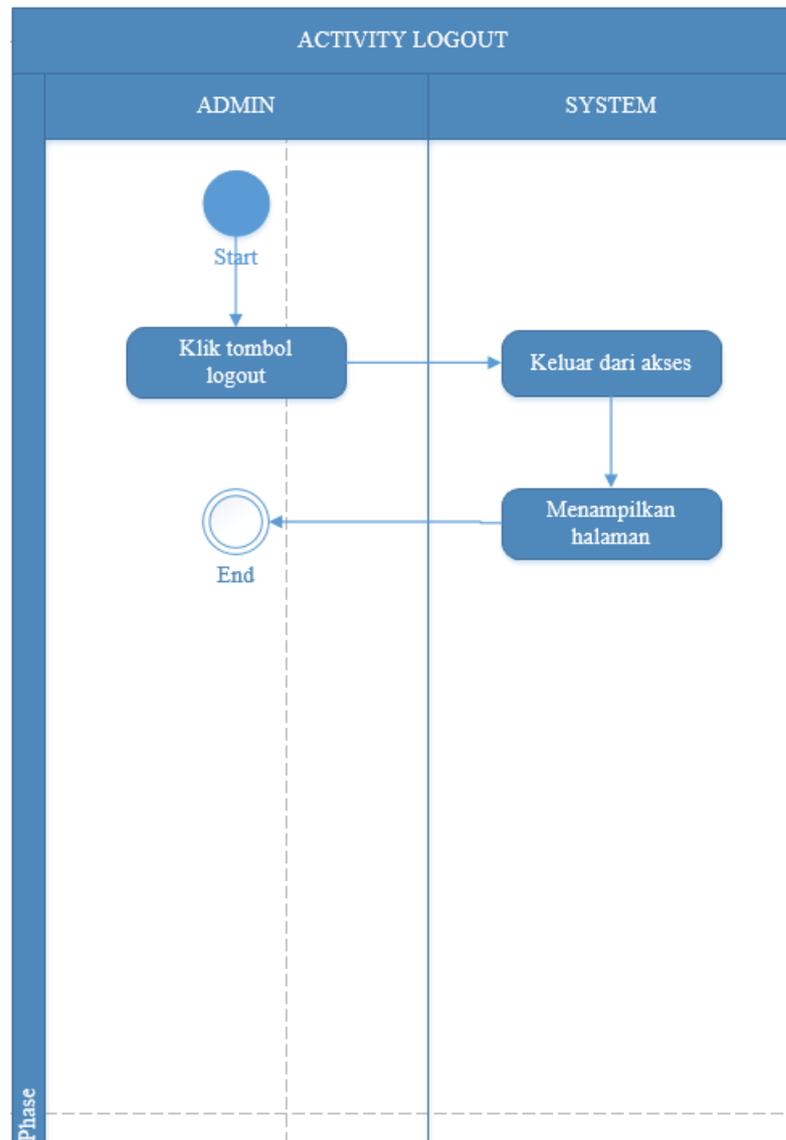
Gambar 3.11 Activity Diagram Entry Laporan

Pada gambar 3.11 merupakan model dari *Activity Diagram* untuk *entry Laporan*. Pada proses awal, admin akan memulai dengan mengklik tombol *entry laporan* terlebih dahulu dan selanjutnya sistem akan menampilkan halaman *entry laporan*. Pada diagram ini admin akan melakukan aktivitas pengolahan data sebagai berikut:

- a. Admin akan melakukan proses *create data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *create data*, maka sistem akan mengarahkan admin ke halaman *entry* laporan untuk melihat data yang berhasil ditambahkan, namun jika proses gagal (*tidak*), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.
- b. Admin akan melakukan proses *update data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *update data*, maka sistem akan mengarahkan admin ke halaman *entry* laporan untuk melihat data yang berhasil diubah, namun jika proses gagal (*tidak*), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan *error* atau tidak sesuai.
- c. Admin akan melakukan proses *search data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *search data*, maka sistem akan mengarahkan admin ke halaman *entry* laporan untuk melihat data yang berhasil dicari, namun jika proses gagal (*tidak*), maka admin akan mendapatkan notifikasi yang berisi data yang dimasukkan tidak ditemukan.
- d. Admin akan melakukan proses *delete data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *delete data*, maka sistem akan mengarahkan admin ke halaman *entry* laporan, namun jika admin (*tidak*) melakukan proses *delete data*, admin bisa langsung melakukan proses *print data* untuk membuat laporan rekap keuangan.
- e. Admin akan melakukan proses *print data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *print data*, maka sistem akan mengarahkan admin ke halaman *print preview* untuk melihat data yang akan dicetak, namun jika admin (*tidak*) melakukan proses *print data*, maka admin bisa langsung melakukan proses *download data* berupa foto atau gambar dibagian bukti kwitansi.
- f. Admin akan melakukan proses *download data* dengan *decision iya* dan **tidak** yang artinya jika admin berhasil (*iya*) melakukan proses *download data*, maka sistem akan mengarahkan admin ke folder *download* untuk melihat data berupa

foto atau gambar yang telah di unduh, namun jika admin (tidak) melakukan proses *download data*, maka admin dapat mengakhiri atau menyelesaikan proses pengolahan data pada model *activity diagram* dengan simbol *end*.

3.4.11 Activity Diagram Logout



Gambar 3.12 Activity Diagram Logout

Pada gambar 3.12 merupakan model dari *Activity Diagram* untuk *logout*. Pada proses ini, ketika admin mengklik tombol *logout*, sistem akan mengarahkan admin ke halaman awal aplikasi.

3.4.12 Perancangan Database

Berikut ini adalah rancangan *database* beserta tabel pada aplikasi web rekapitulasi laporan keuangan Program Studi Universitas Muhammadiyah Yogyakarta:

Tabel 3.1 Data Tabel Users

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
Id	Int	10	<i>Primary Key</i>	Id Admin
Name	Varchar	255	-	Nama Admin
Email	Varchar	255	-	Email Admin
email_verified_at	timestamp	-	-	Verifikasi Email Admin
Password	Varchar	191	-	Password Admin
remember_token	Varchar	100	-	-
created_at	timestamp	-	-	Waktu pembuatan
updated_at	timestamp	-	-	Waktu perubahan

Tabel 3.2 Data Tabel Kategoriinput

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
id	int	10	<i>Primary Key</i>	Id Kategori
kategori	varchar	255	-	Nama Kategori
created_at	timestamp	-	-	Waktu pembuatan
updated_at	timestamp	-	-	Waktu perubahan

Tabel 3.3 Data Tabel Kegiataninput

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
id	int	10	<i>Primary Key</i>	Id Kegiatan
namakegiatan	varchar	255	-	Nama Kegiatan
nominalkegiatan	int	255		Nomonal Kegiatan
created_at	timestamp	-	-	Waktu pembuatan
updated_at	timestamp	-	-	Waktu perubahan

Tabel 3.4 Data Tabel Thajaraninput

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
id	int	10	<i>Primary Key</i>	Id Tahun ajaran
semester	varchar	255	-	Semester
tahunajaran	varchar	255	-	Tahun ajaran
created_at	timestamp	-	-	Waktu pembuatan
updated_at	timestamp	-	-	Waktu perubahan

Tabel 3.5 Data Tabel Laporanann

Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
id	int	10	<i>Primary Key</i>	Id Laporan
kategori	varchar	191	-	Nama Kategori
namakegiatan	varchar	191		Nama Kegiatan
tahunajaran	varchar	191	-	Tahun ajaran
tanggal	date	-	-	Tanggal Laporan
keterangan	varchar	191	-	Keterangan Laporan
nominalkegiatan	int	20	-	Nominal Kegiatan
nominalkategori	int	20	-	Nominal Kategori

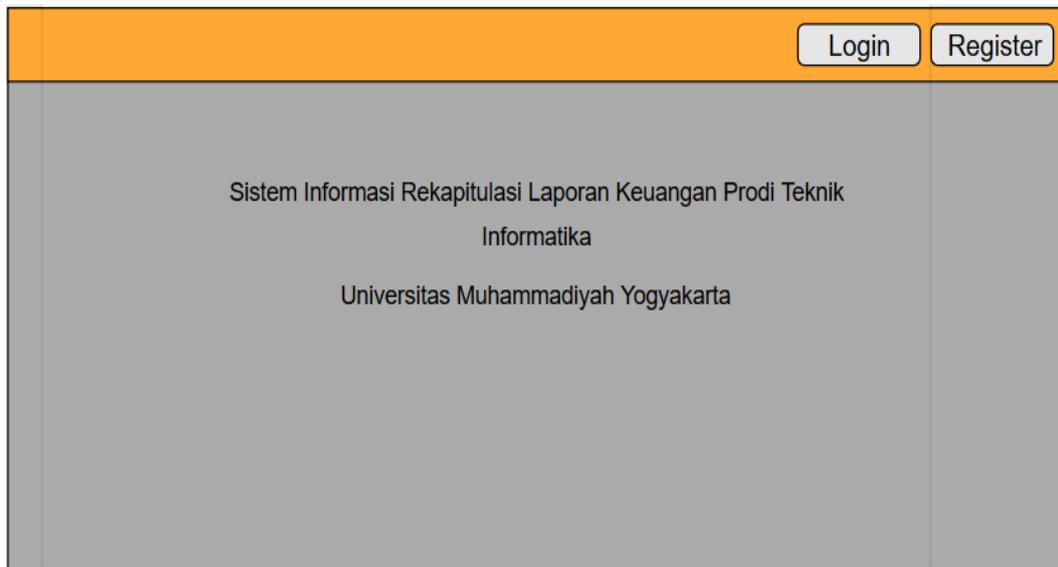
Nama Kolom	Tipe Data	Panjang Data	Key	Keterangan
photo	varchar	191	-	Gambar Bukti Laporan
created_at	timestamp	-	-	Waktu pembuatan
updated_at	timestamp	-	-	Waktu perubahan

3.5 Rancangan *Interface*

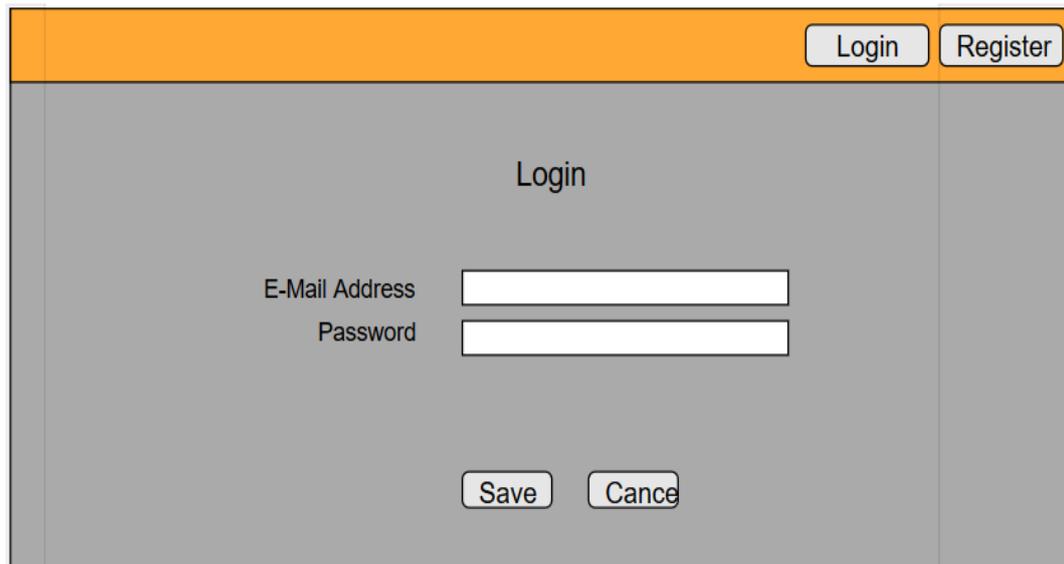
Dalam pembuatan website, perancangan *interface* sangat penting. Hal tersebut dikarenakan interface merupakan hal yang berhubungan langsung dengan pengguna. Dengan merancang *interface* memberikan kemudahan pengguna dalam mengakses website. *interface* digunakan penghubung antarmuka pengguna dengan website dan memberikan *feedback* pada setiap kegiatan pengguna. Oleh karena itu, Semakin menarik *Interface* yang dibuat, maka pengguna akan merasa nyaman dengan website tersebut.

3.5.1 Rancangan *Interface* halaman awal, *login*, dan *register*

Halaman awal adalah tampilan awal saat pengguna mengakses website rekapitulasi laporan Program Studi Teknik Informatika Universitas Muhammadiyah Yogyakarta. Untuk semua pengguna atau admin akan mendapatkan tampilan yang sama pada halaman awal yang terdapat pada gambar 3.13. Kemudian setelah halaman awal terdapat rancangan halaman *login* beserta *register*. Pada gambar 3.14 terdapat *interface login* yang berfungsi sebagai autentikasi pengguna ketika masuk ke halaman website. Sedangkan pada gambar 3.15 terdapat *interface register* berfungsi sebagai pembuatan data admin baru.



Gambar 3.13 *Interface* Halaman Awal



Gambar 3.14 *Interface login*

Login Register

Register

Name

E-Mail Address

Password

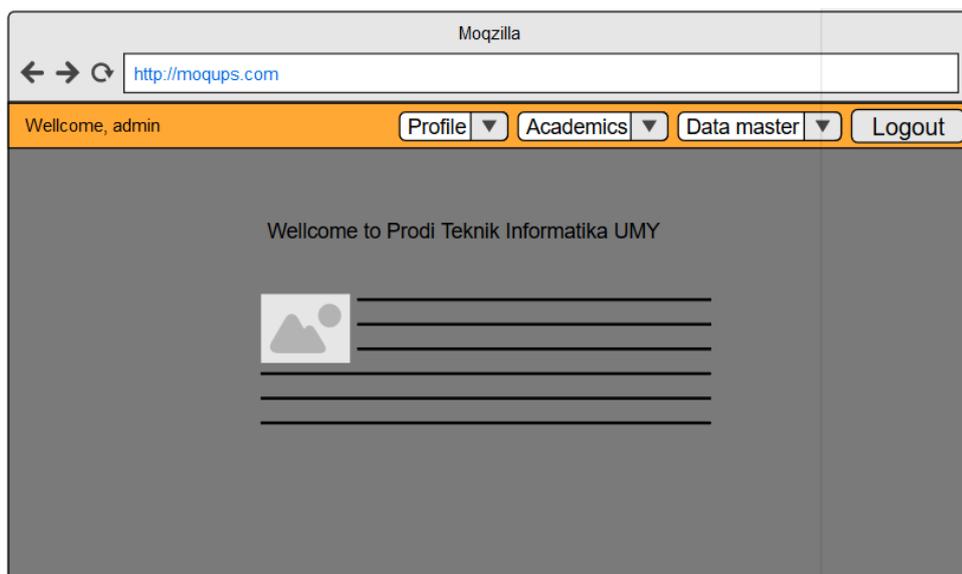
Confirm Password

Save Cancel

Gambar 3.15 *Interface Register*

3.5.2 Rancangan *Interface home*

Setelah login, admin akan masuk ke dalam halaman utama yang disebut *home*. Seperti yang dilihat pada gambar 3.15, admin akan masuk ke halaman *About* yang berisikan konten tentang Program Studi Teknik Informatika Universitas Muhammadiyah Yogyakarta. Di website ini, halaman *home* terdiri dari beberapa *sub menu* yang diakses melalui *dropdown*. Selain dapat melihat *sub menu* yang ada di halaman utama admin dapat melihat halaman lainnya, seperti *Academics* maupun *Data Master* yang tentunya diakses melalui *dropdown*.



Gambar 3.16 *Interface Home*

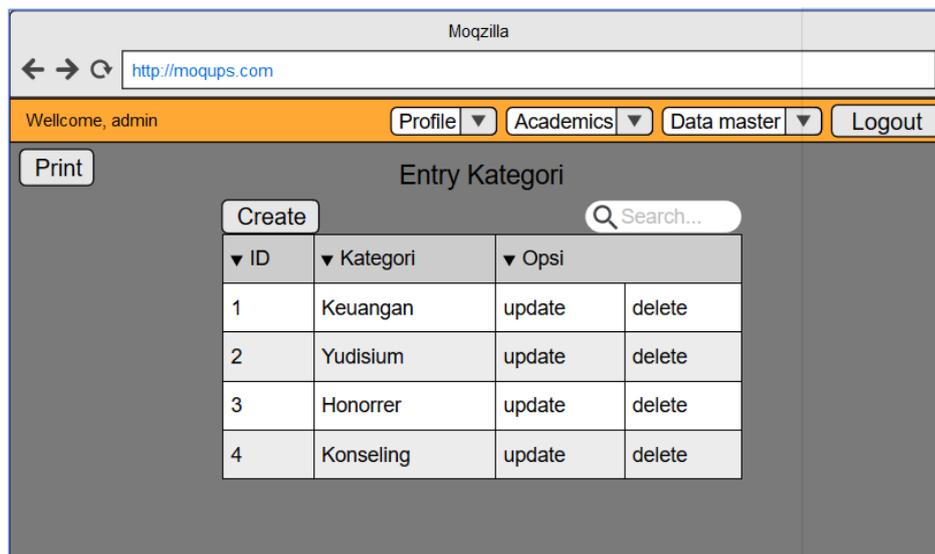
Berikut ini adalah keterangan yang ada didalam *interface home*:

Tabel 3.6 Menu Home

No	Menu	Sub Menu	Keterangan
1	Profile	About	Sub menu yang berisi konten Program Studi Teknik Informatika UMY
		History	Sub menu yang berisi konten sejarah berdirinya Program Studi Teknik Informatika UMY
		Vision & Mission	Sub menu yang berisi konten visi dan misi Program Studi Teknik Informatika UMY
2	Academics	Lecturer	Sub menu yang berisi konten profil dosen beserta staff Program Studi Teknik Informatika UMY
		Organization	Sub menu yang berisi konten organisasi yang dibawah naungan Program Studi Teknik Informatika UMY
3	Data Master	Entry Kategori	Sub menu yang berisi data kategori rekap keuangan Prodi Teknik Informatika UMY
		Entry Tahun ajaran	Sub menu yang berisi data tahun ajaran rekap keuangan Program Studi Teknik Informatika UMY

No	Menu	Sub Menu	Keterangan
		Entry Laporan	Sub menu yang berisi data lapora rekap keuangan Program Studi Teknik Informatika

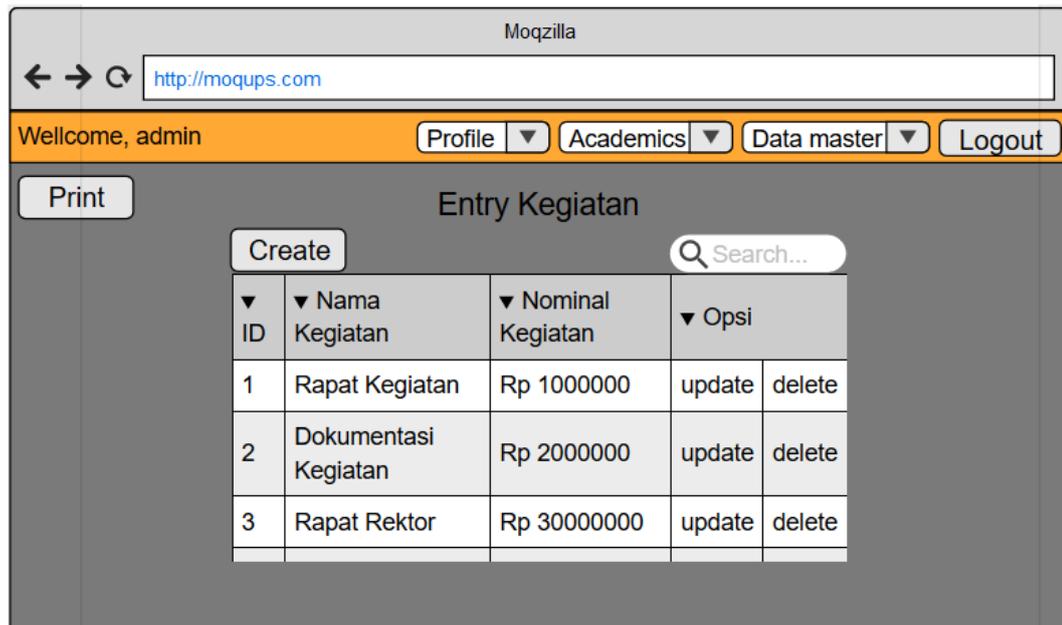
3.5.3 Rancangan *Interface Entry Kategori*



Gambar 3.17 *Interface Entry Kategori*

Pada gambar 3.17 merupakan rancangan *interface* dari halaman *entry kategori*. Halaman ini adalah halaman dimana admin dapat mengelola data kategori rekap keuangan baik itu *create* data, *update* data, *delete* data, *search* data, serta *print* data yang akan nantinya digunakan dalam pembuatan laporan.

3.5.4 Rancangan *Interface Entry Kegiatan*

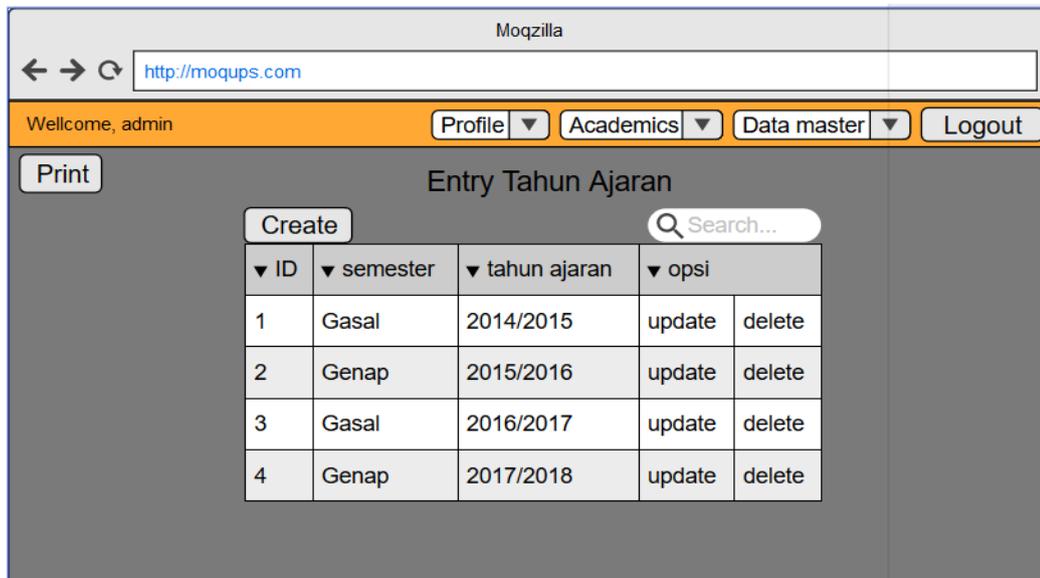


Gambar 3.18 *Interface Entry Kegiatan*

Pada gambar 3.18 merupakan rancangan *interface* dari halaman *entry kategori*. Halaman ini adalah halaman dimana admin dapat mengelola data kategori

rekap keuangan baik itu *create* data, *update* data, *delete* data, *search* data, serta *print* data yang akan nantinya digunakan dalam pembuatan laporan.

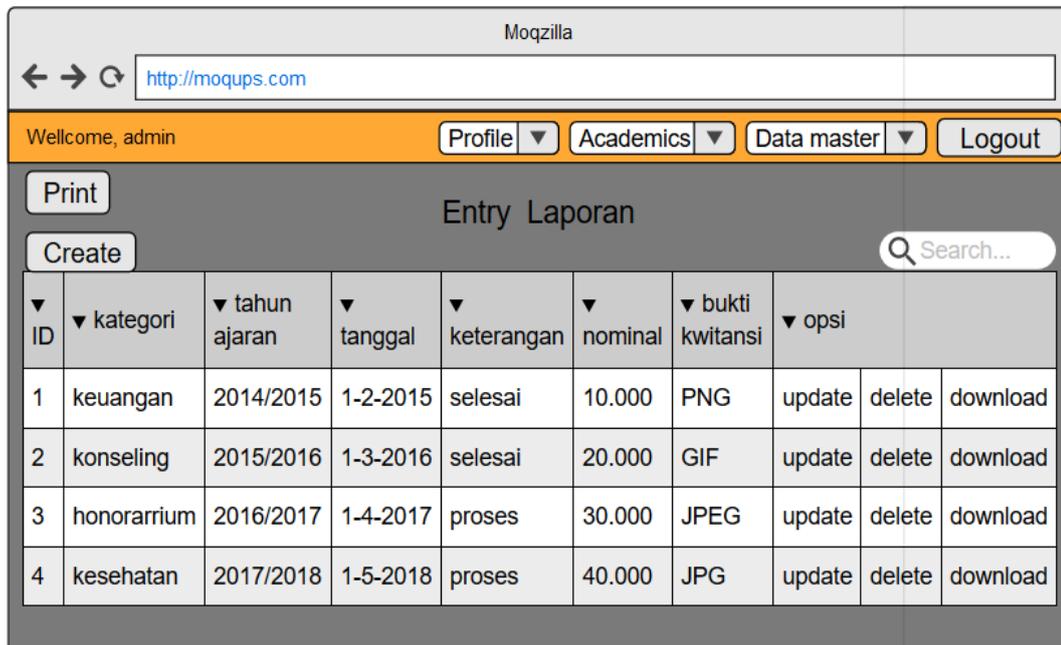
3.5.5 Rancangan *Interface Entry Tahun ajaran*



Gambar 3.19 *Interface Entry Tahun Ajaran*

Pada gambar 3.19 merupakan rancangan *interface* dari halaman *entry* tahun ajaran. Sama halnya seperti *entry* kategori, Halaman ini adalah halaman dimana admin dapat mengelola data tahun ajaran rekap keuangan baik itu *create* data, *update* data, *delete* data, *search* data, serta *print* data yang akan nantinya digunakan dalam pembuatan laporan.

3.5.6 Rancangan Interface Entry Laporan



Gambar 3.20 *Interface Entry Laporan*

Pada gambar 3.20 merupakan rancangan *interface* dari halaman *entry* laporan. halaman ini merupakan fitur utama dari aplikasi ini. Sama halnya seperti *entry* kategori ataupun *entry* tahun ajaran, Halaman ini adalah halaman dimana admin dapat mengelola data laporan rekap keuangan baik itu *create* data, *update* data, *delete* data, *search* data, serta *print* data yang akan nantinya digunakan dalam pembuatan laporan.