

BAB III METODE PENELITIAN

3.1. Objek Penelitian

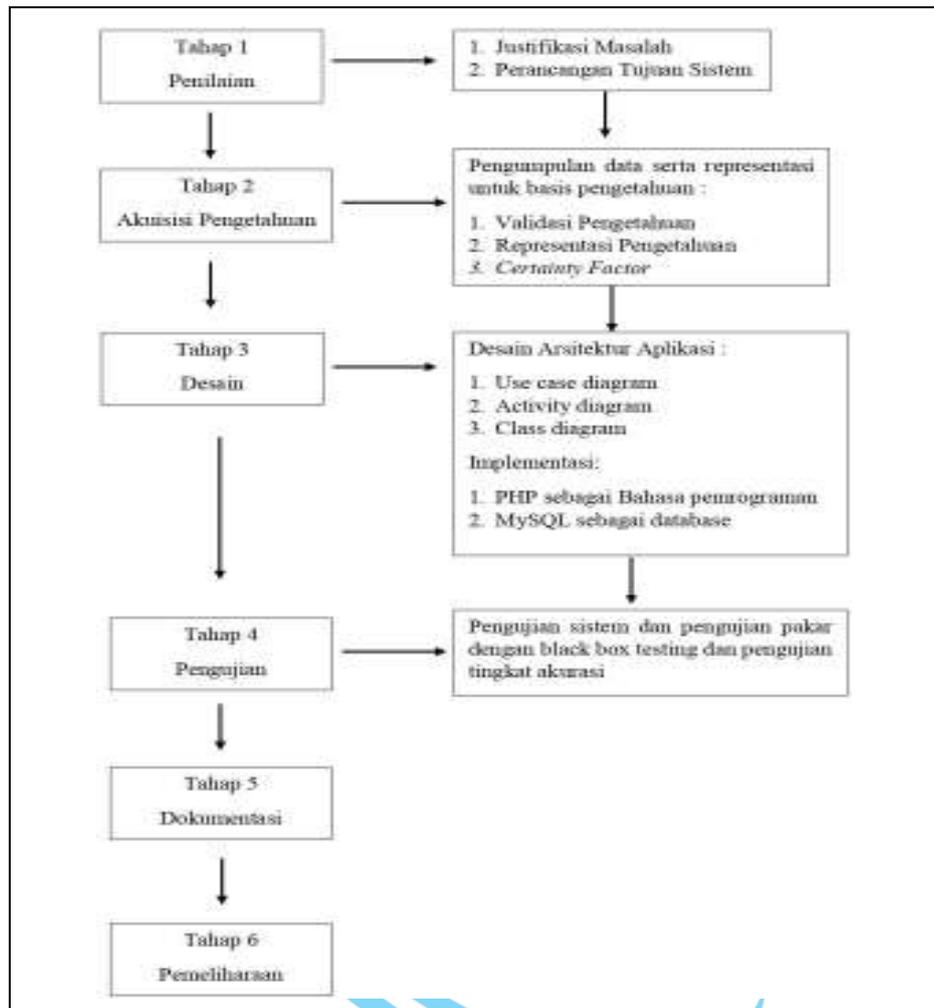
Pada penelitian ini objek penelitian yang digunakan merupakan pemuda pemudi karang taruna Desa Kiarapayung. Sampel yang akan diambil pada penelitian ini kurang lebih 30 orang. Penelitian ini diharapkan dapat memberikan informasi mengenai gejala kecanduan *game online* serta cara penanganannya dengan bantuan dari Ibu Syahrani Paramitha Kurnia Illahi., M.Psi sebagai seorang ahli psikologi. Lokasi penelitian dilakukan di karang taruna desa kiarapayung yang bertempat di Dusun Krajan Desa Kiarapayung Kec.Klari Kab.Karawang, Jawa Barat 41371. Waktu penelitian dilaksanakan pada bulan April – September 2022.

Tabel 3. 1 Tahapan dan Waktu penelitian

Kegiatan	Bulan					
	April	Mei	Juni	Juli	Agust	Sept
Penilaian						
Akuisisi Pengetahuan						
Desaian						
Implementasi						
Pengujian						
Dokumentasi						
Pemeliharaan						

3.2. Prosedur Penelitian

Tahapan dan prosedur penelitian yang dilakukan pada penelitian ini menggunakan metode *Expert System Development Life Cycle* (ESDLC) dapat dilihat pada Gambar 3.1. :



Gambar 3. 1 Gambar Prosedur Penelitian

Sesuai Gambar 3.1 diatas, secara rinci tahapan di uraikan sebagai berikut :

3.2.1 Penilaian

Dalam tahapan penilaian ini yang dilakukan yaitu mendefinisikan masalah dan juga mendefinisikan tujuan secara umum serta ruang lingkup dari sistem. Tujuannya adalah untuk menganalisa masalah dan kebutuhan terkait sistem pakar.

3.2.2 Akuisisi Pengetahuan

Pada tahap akuisisi pengetahuan adalah mengidentifikasi sumber pengetahuan tentang masalah yang akan ditangani serta arah pengembangannya. Pengetahuan ini digunakan untuk memberikan informasi mengenai permasalahan yang menjadi acuan dalam perancangan sistem pakar.

3.2.3 Desain

Pada tahap desain, desain antarmuka dan teknik pemecahan masalah dapat diimplementasikan pada sistem pakar. Pada tahap desain, desain sistem dibangun untuk lebih memahami masalah. Pada tahap ini ada beberapa langkah, yaitu:

1. Desain data

Tahapan ini menjelaskan detail dari komponen – komponen :

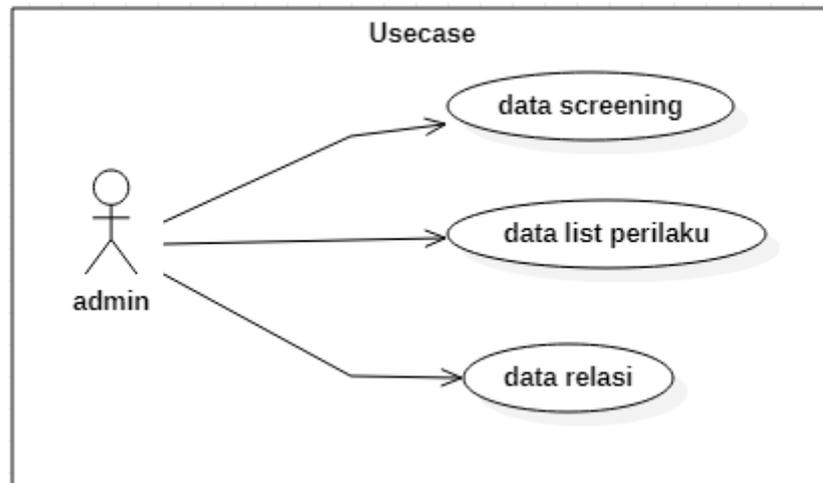
- a. *User* adalah *admin* dan pengguna
- b. *Hardware* merupakan perangkat keras yang digunakan untuk mengolah data : DESKTOP-DCD2M8H, RAM 10 GB, intel® core™ i3-2348M CPU @ 2.30GHz, *operating system windows 10 64-bit*.
- c. *Software* yang digunakan, *Microsoft Office Word 2016*, DBMS MySQL, PHP dan Menggunakan program *Unified Modeling Language (UML)*

2. Desain Arsitektur

Pada langkah ini, tujuan utamanya adalah mengembangkan struktur program dan struktur data menggunakan struktur algoritma, dengan metode koefisien kepastian. Metode koefisien kepastian adalah metode untuk membuktikan apakah suatu kejadian pasti atau tidak pasti. Pada antarmuka ini, perancangan dilakukan dengan menggunakan aplikasi *Unified Modeling Language (UML)* yang menampilkan pilihan kebutuhan pengguna. Dalam *desain* antarmuka, menu *admin* dan menu *pengguna* akan dibuat.

1. *Usecase* kelola *admin*

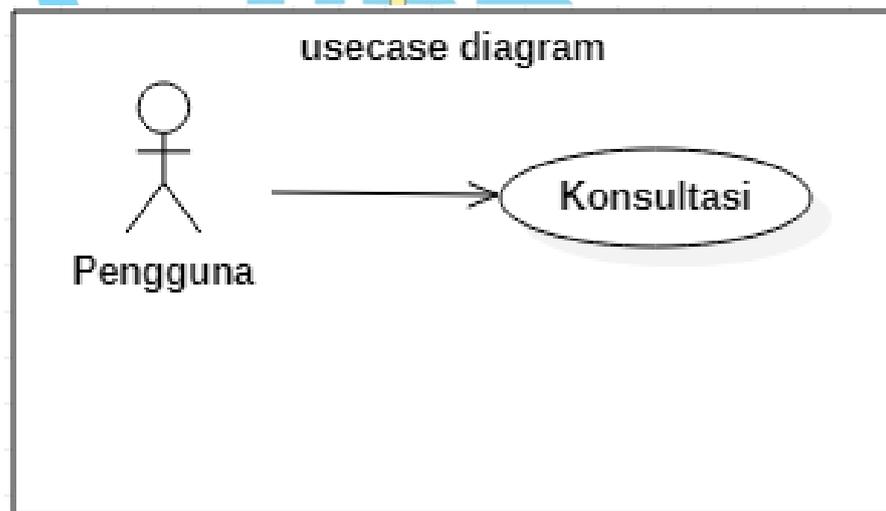
Usecase Sistem pakar Diagnosa Tingkat Kecanduan *Game online* pada halaman ini hanya *admin* yang bisa mengakses menu data *screening*, data *list* perilaku dan data relasi.



Gambar 3. 2 Usecase Kelola Data Admin

2. *Usecase kelola pengguna*

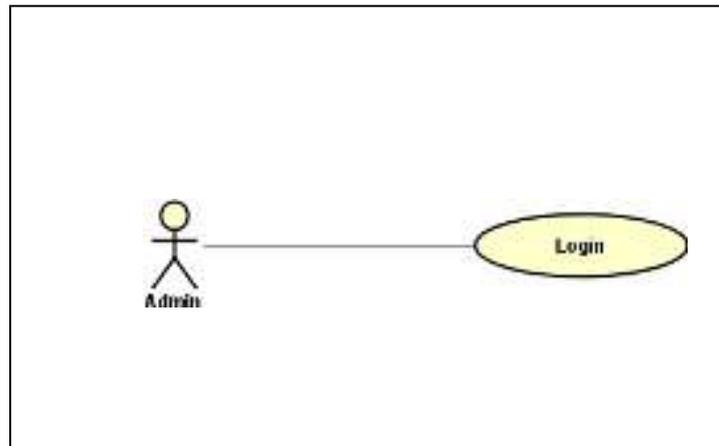
Adapun pengguna hanya bisa memilih menu konsultasi yang dimana pengguna juga dapat melihat hasilnya.



Gambar 3. 3 Usecase Kelola Data Pengguna

3. *Usecase Login*

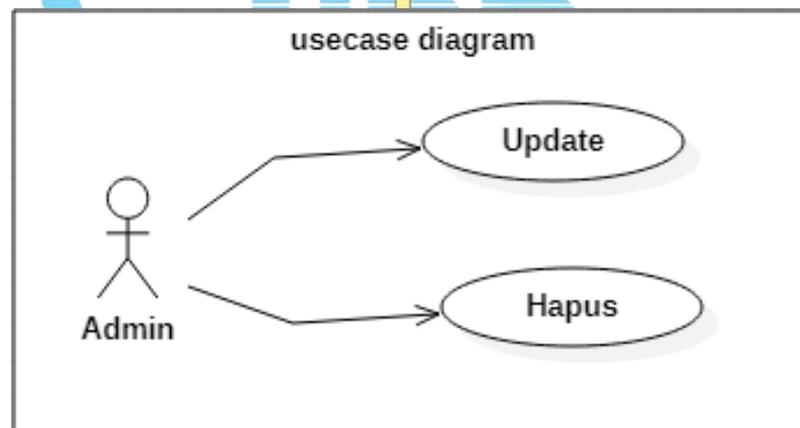
Admin memasuki halaman *login* dengan *username: admin* dan *password: admin123* lalu akan diarahkan ke menu *dashboard*.



Gambar 3. 4 Usecase Kelola Login

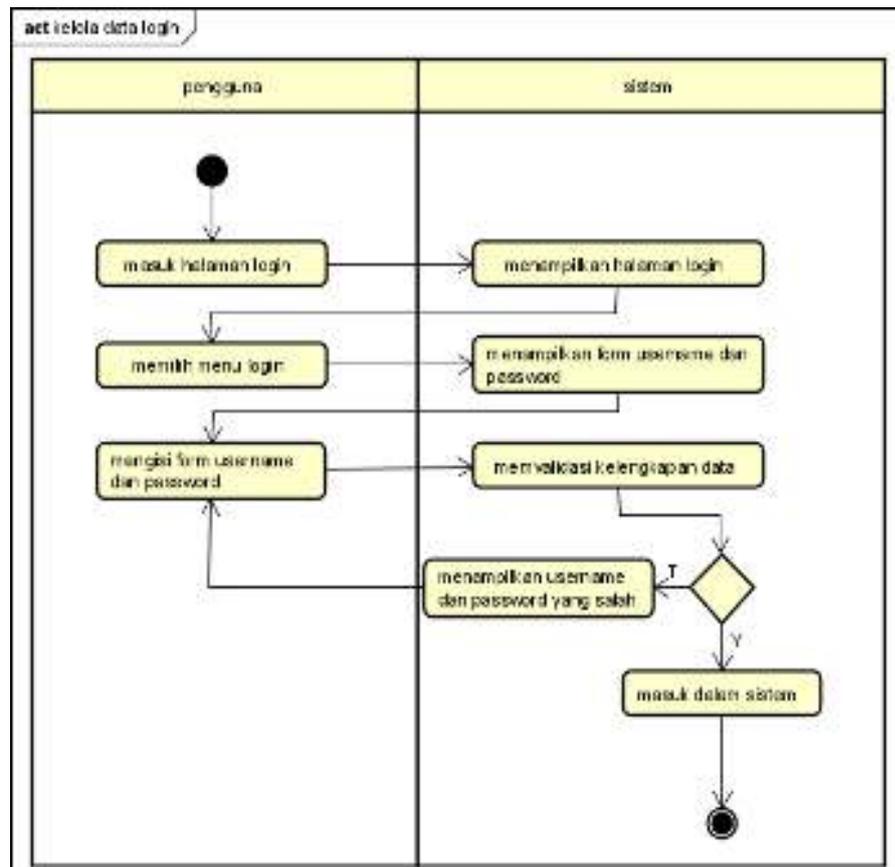
4. Usecase data screening, data list perilaku dan data relasi

Usecase data screening, data list perilaku dan data relasi yang mengakses hanya *admin*. *Admin* hanya bisa *update* dan hapus data pada menu tersebut.



Gambar 3. 5 Usecase Diagram

5. Activity Diagram Login

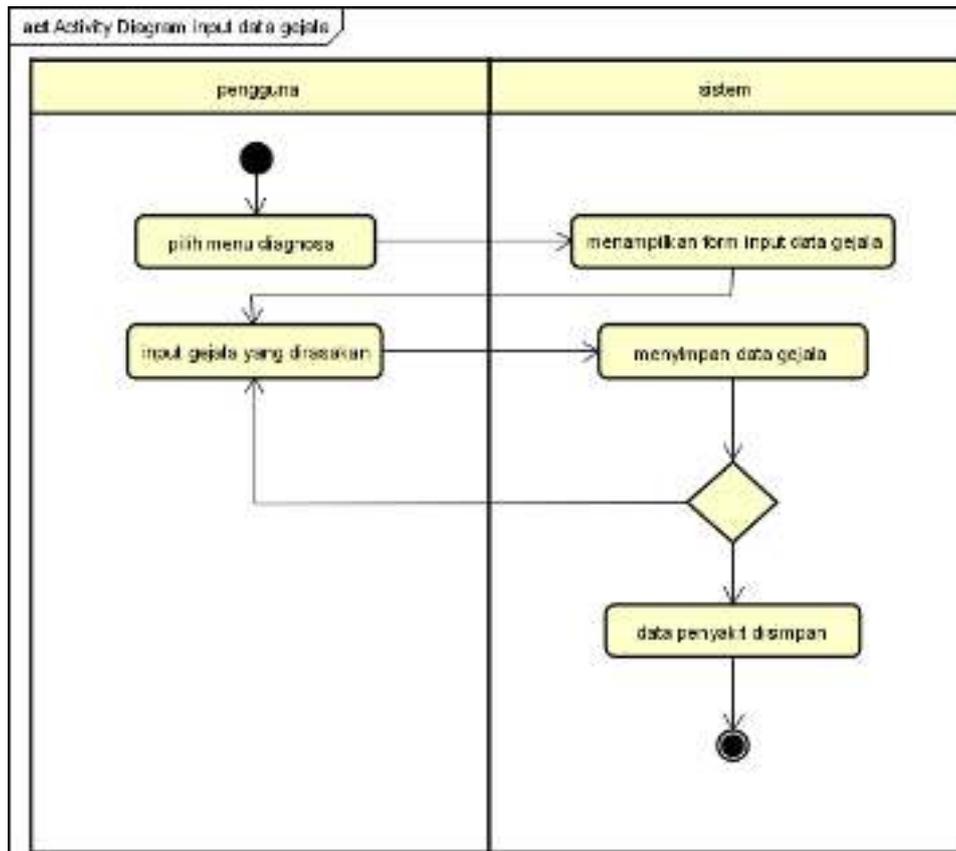


Gambar 3.6 Activity Diagram Login

Pada keterangan Gambar 3.6 maka dapat dijelaskan :

1. *Admin* membuka halaman *login*, tampil halaman menu *login*.
2. Pengguna memilih menu *login*, tampil menu form *username* dan *password*.
3. Pengguna mengisi form *username* dan *password*, memvalidasi *username* dan *password*.
4. Jika berhasil maka akan masuk kehalaman *dashboard*
5. Jika gagal tampil *username* dan *password* salah maka harus mengisi ulang form *username* dan *password*.

6. Activity Diagram Konsultasi

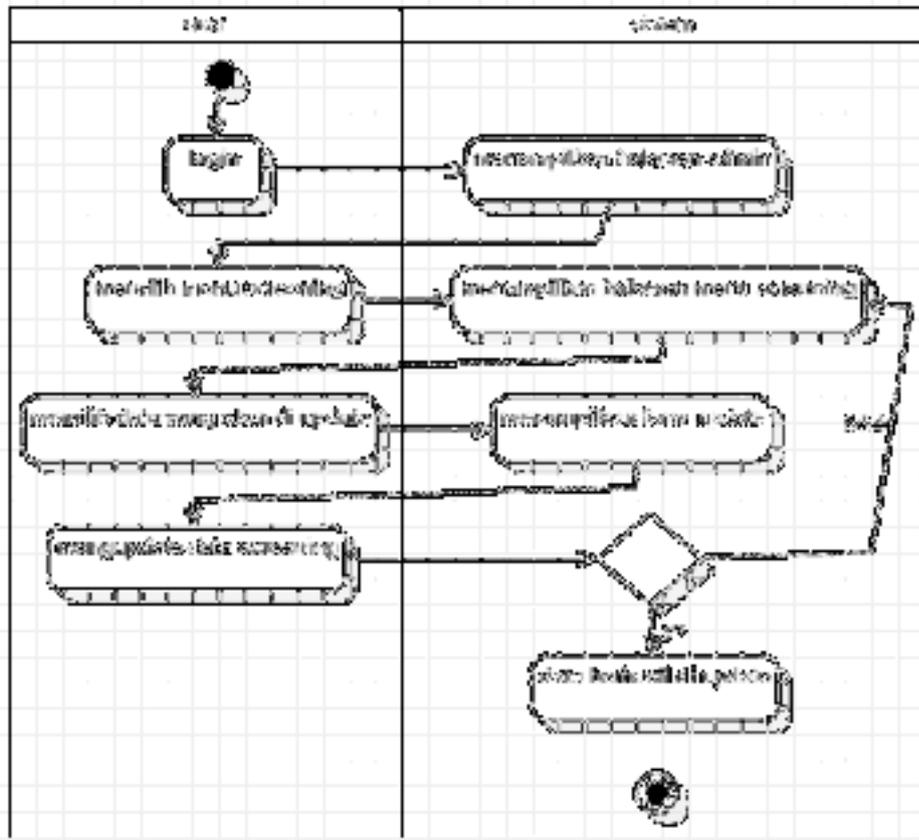


Gambar 3. 7 Activity Diagram Konsultasi

Dari keterangan Gambar 3.7 maka dapat dijelaskan :

1. Pengguna memilih menu Konsultasi, tampil halaman menu konsultasi.
2. Pengguna memasukan gejala yang di rasakan, sistem menyimpan data dan menghitung hasil konsultasi
3. Menyimpan data yang sudah pengguna masukan, tampil hasil konsultasi.

7. Activity diagram *update screening*

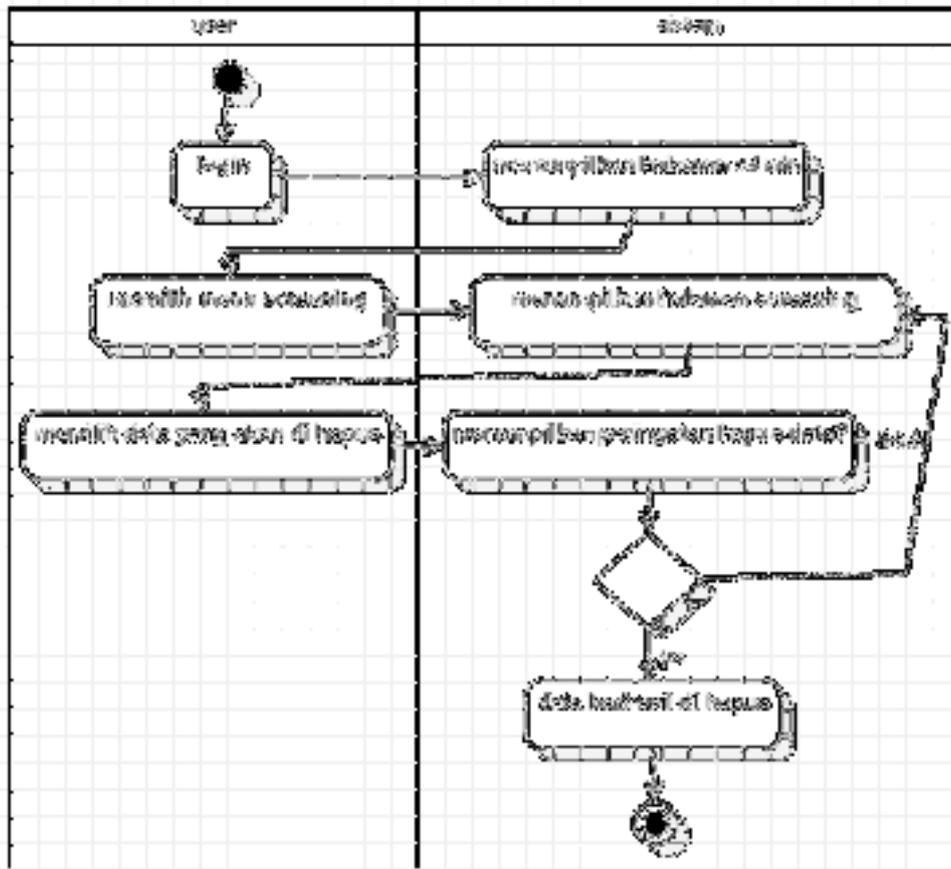


Gambar 3. 8 Activity Diagram *Update Data Screening*

Dari keterangan Gambar 3.8 maka dapat dijelaskan :

1. *Admin* sudah melakukan *login*, sistem menampilkan halaman *admin*.
2. *Admin* memilih menu *screening*, sistem menampilkan halaman *screening*.
3. *Admin* memilih data yang akan diupdate pada menu *screening*, sistem menampilkan form *update screening*.
4. *Admin* meng-update data *screening*, jika *admin* menekan tombol simpan maka data yang sudah di update akan tersimpan di sistem.
5. Jika *admin* memilih *cancel* maka akan kembali ke halaman *screening*.

8. Activity diagram hapus data screening

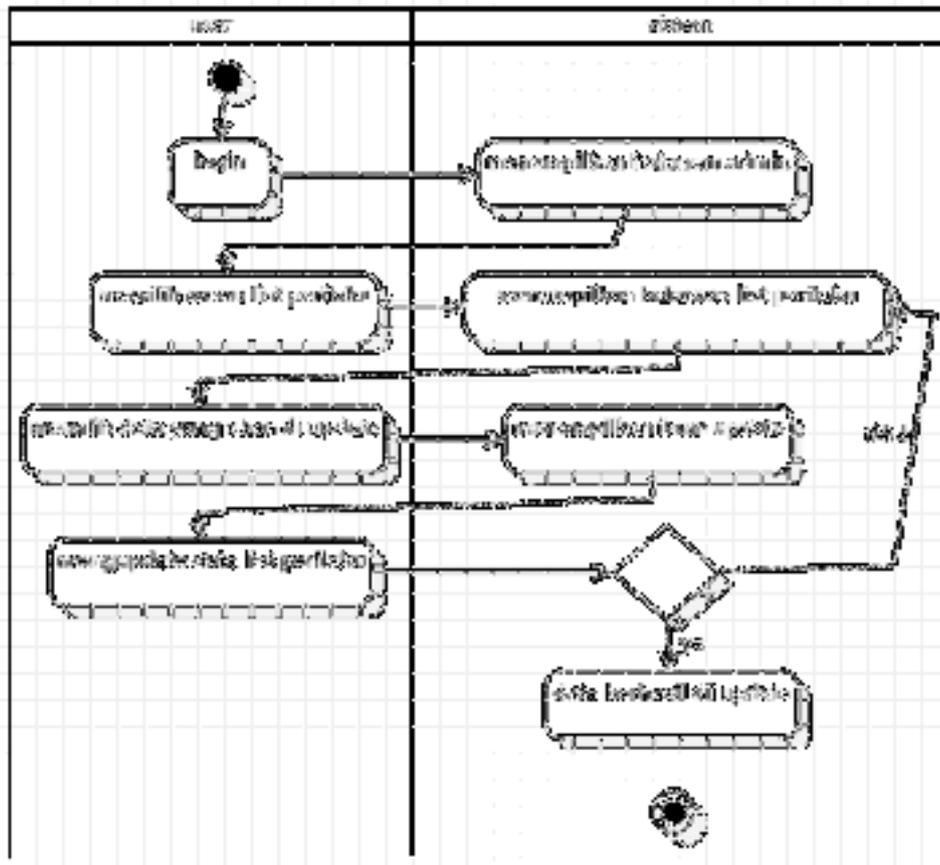


Gambar 3. 9 Activity Diagram Hapus Data Screening

Dari keterangan Gambar 3.9 maka dapat dijelaskan :

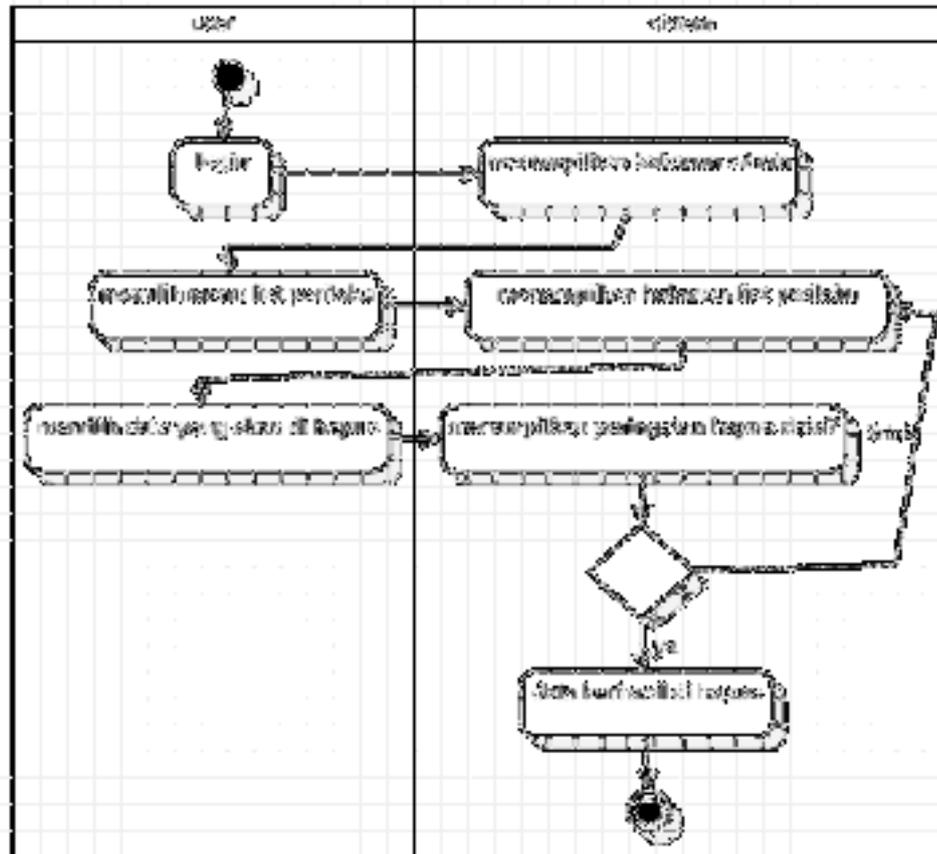
1. *Admin* sudah melakukan *login*, sistem menampilkan halaman *admin*.
2. *Admin* memilih menu halaman *screening*, sistem akan menampilkan halaman *screening*.
3. *Admin* memilih data yang akan dihapus pada data *screening*, lalu sistem akan menampilkan peringatan “hapus data?”
4. Jika *admin* memilih “ok” data akan dihapus jika *admin* memilih “cancel” maka data tidak akan dihapus tetapi sistem menampilkan halaman *screening*.

9. Activity diagram *update list* perilaku



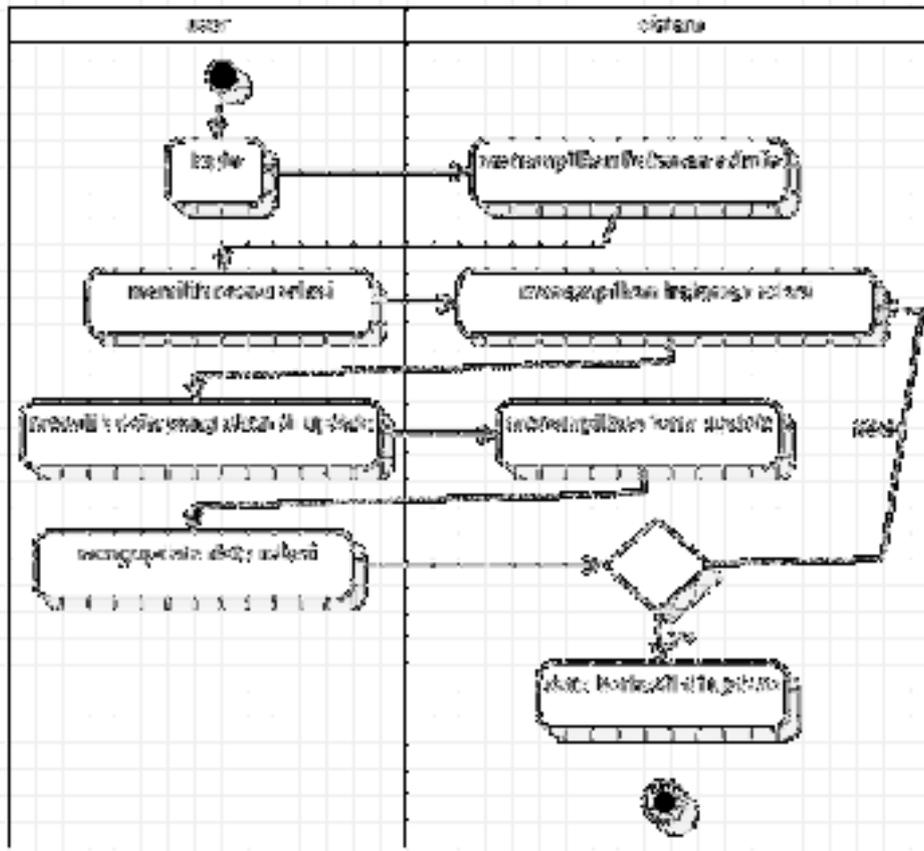
Gambar 3. 10 Activity Diagram *Update Data List* Perilaku
 Pada keterangan Gambar 3.10 dapat dijelaskan :

1. *Admin* sudah melakukan *login*, sistem akan menampilkan halaman *admin*
2. *Admin* memilih menu , sistem akan menampilkan halaman *list* perilaku.
3. *Admin* memilih data yang akan di-*update* pada menu *list* perilaku, sistem akan akan menampilkan *form update list* perilaku.
4. *Admin* melakukan *update* pada *list* perilaku, jika *admin* menekan tombol simpan maka data *list* perilaku yang dipilih akan di-*update* pada sistem.
5. Jika *admin* memilih *cancel* maka sistem akan kembali menampilkan halaman *list* perilaku.

10. Activity Diagram hapus data *list* perilakuGambar 3. 11 Activity Diagram Hapus Data *List* Perilaku

Pada keterangan Gambar 3.11 dapat dijelaskan :

1. *Admin* sudah melakukan *login*, sistem akan menampilkan halaman *admin*
2. *Admin* memilih menu *list* perilaku, sistem akan menampilkan halaman *list* perilaku
3. *Admin* memilih data *list* perilaku yang akan dihapus, sistem akan menampilkan peringatan “hapus data?”
4. Jika *admin* menekan tombol “ok” maka sistem akan menghapus data, tetapi jika *admin* memilih *cancel* maka sistem akan kembali pada halaman menu *list* perilaku

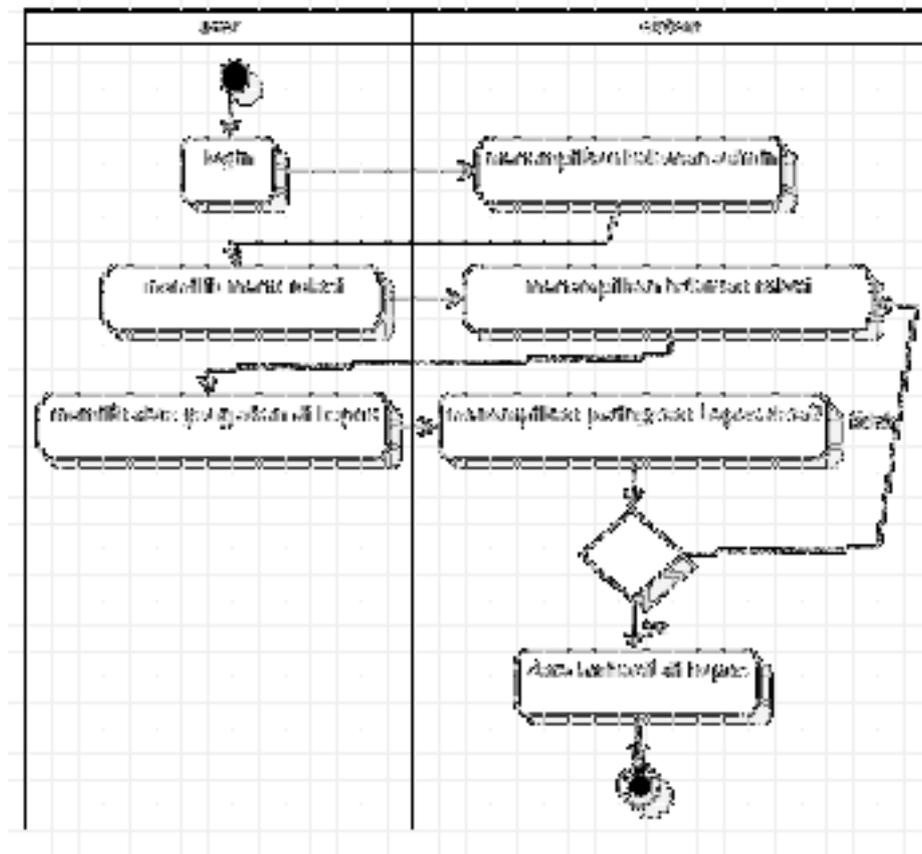
11. Activity diagram *update* data relasi

Gambar 3. 12 Activity Diagram Update Data Relasi

Pada keterangan Gambar 3.12 maka dapat dijelaskan sebagai berikut :

1. *Admin* sudah melakukan *login*, sistem menampilkan halaman *admin*.
2. *Admin* memilih menu basis relasi, sistem akan menampilkan halaman relasi
3. *Admin* memilih data yang akan di-*update* pada menu relasi, sitem menampilkan form *update* data relasi.
4. *Admin* meng-*update* data relasi, jika admin memilih tombol simpan maka data relasi yang telah di-*update* tersimpan. Jika admin memilih *cancel* maka sistem akan kembali ke menu halaman data relasi

12. Activity diagram hapus data relasi

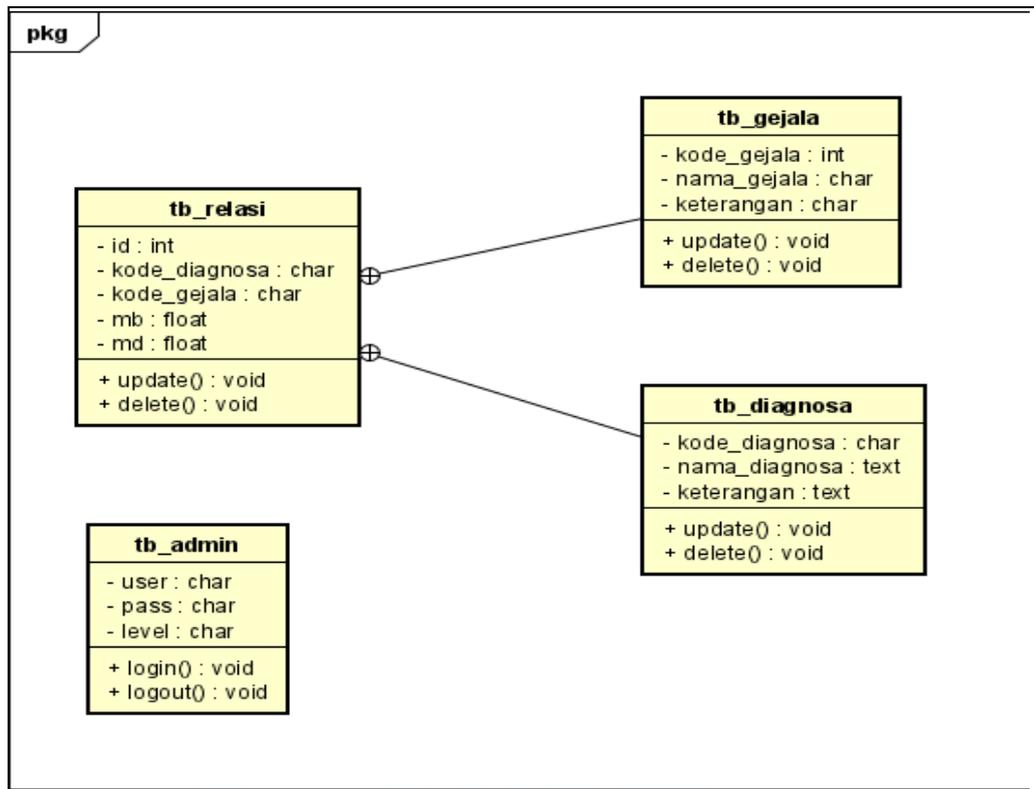


Gambar 3. 13 Activity Diagram Hapus Data Relasi

Pada keterangan Gambar 3.13 maka dapat dijelaskan :

1. *Admin* melakukan *login*, sistem menampilkan halaman *admin*.
2. *Admin* memilih menu relasi, sistem menampilkan halaman relasi.
3. *Admin* memilih data relasi yang akan dihapus, sistem menampilkan peringatan “hapus data?”
4. *Admin* memilih “ok” maka data relasi yang dipilih akan dihapus pada sistem, jika memilih “cancel” maka sistem akan kembali ke menu relasi.

13. Class diagram



Gambar 3. 14 Class Diagram

Dari keterangan Gambar 3.14 maka dapat dijelaskan :

1. Tabel tb_relasi memiliki hubungan satu sama lain pada tabel tb_gejala dan tabel tb_diagnosa.
2. Tabel tb_admin hanya digunakan oleh *admin* dimana hanya bisa *login* menggunakan *username* dan *password* khusus *admin*.
3. Pada tabel tb_relasi, tb_diagnosa dan tb_gejala pada sistem hanya bisa *diupdate* dan *dihapus*.

14. Implementasi

Tahapan ini merupakan sebuah proses mengubah desain menjadi kode program.

3.2.4. Pengujian

Tahap selanjutnya adalah tahap pengujian. Tahap pengujian ini dimaksudkan untuk memastikan bahwa fungsi dan fitur sistem sudah aktif dan berjalan sesuai kebutuhan. Pengujian dilakukan dengan metode *blackbox testing*.

a. *Blackbox Testing*

Blackbox Testing adalah pengujian yang melibatkan fungsionalitas perangkat lunak, yang dilakukan oleh pengguna untuk menemukan kesalahan dalam antarmuka dan kesalahan dalam struktur data atau mengakses database *eksternal*.

b. Pengujian Akurasi

Pengujian akurasi merupakan metode untuk memperoleh nilai ketepatan yang dinyatakan dalam persentase. Pengujian tingkat akurasi pada sistem dengan metode *certainty factor* dihitung menggunakan rumus:

$$\text{persentase keberhasilan} = \frac{\text{jumlah item uji berhasil}}{\text{jumlah seluruh item uji}} \times 100\% \quad (7)$$

3.2.5. Dokumentasi

Dokumentasi dapat berbentuk *source code* dan panduan penggunaan sistem. Dokumentasi merupakan hal yang bermanfaat bagi pengguna dan peneliti selanjutnya.

3.2.6. Pemeliharaan

Pemeliharaan adalah suatu proses memelihara dan merawat sistem yang telah di bangun.