

## PERANCANGAN SISTEM IDENTIFIKASI KECANDUAN GAME BERDASARKAN JENIS PERILAKU KECANDUAN GAME MENGGUNAKAN PENDEKATAN BERBASIS OBJECT

Anastasya Latubessy<sup>1\*</sup>, Ahmad Jazuli<sup>1</sup>

<sup>1</sup>Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muria Kudus  
Gondangmanis, PO Box 53, Bae, Kudus 59352

\*Email: anastasya.latubessy@umk.ac.id

### Abstrak

*Kecenderungan seseorang dalam bermain game dengan intensitas dan pola yang tidak biasa, dapat dilihat berdasarkan perilakunya. Terdapat enam jenis perilaku kecanduan game yaitu Salience, Euphoria, Conflict, Tolerance, Withdrawal, Relapse and Reinstatement. Dimana, seseorang dapat dikatakan mengalami kecanduan game apabila memiliki paling sedikit tiga dari enam jenis perilaku kecanduan game yang ada. Penelitian ini menggunakan diagram-diagram UML(Unified Modelling Language) untuk merancang sistem identifikasi kecanduan game. Sementara model yang digunakan dalam penelusuran gejala adalah backward chaining. Sehingga menghasilkan rancangan yang dapat digunakan sebagai dasar dalam membangun startup aplikasi identifikasi kecanduan game.*

**Kata kunci:** perancangan, aplikasi, identifikasi, kecanduan, game

### 1. PENDAHULUAN

Manusia merupakan makhluk sosial yang membutuhkan hiburan. Salah satu hiburan yang dapat diperoleh adalah dengan bermain *game*. Namun, kecenderungan seseorang dalam bermain *game* terkadang tidak dapat dikontrol. Seseorang dapat melupakan waktu dan aktifitas yang seharusnya dilakukan karena keasyikan bermain *game*. Hal tersebut dapat memicu terjadinya kecanduan *game*. Menurut Brown, seorang ahli psikologi, terdapat enam jenis perilaku kecanduan *game* antara lain *Salience, Euphoria, Conflict, Tolerance, Withdrawal, Relapse and Reinstatement*. Seseorang dikatakan kecanduan *game* apabila memenuhi minimal tiga dari enam jenis yang diungkapkan oleh Brown (Faried, 2012).

Kecanduan *game* saat ini juga mempengaruhi anak-anak yang dapat menjadi sasaran empuk dalam mempromosikan *game*. Seperti yang diungkapkan dalam penelitian Latubessy, A dan Ahsin, M.N(2016) yang menyatakan bahwa tingkat keserangan bermain *game* sangat berpengaruh pada keaktifan anak dalam proses pembelajaran. Sehingga penting sekali mendeteksi tingkat kecanduan terhadap *game*.

Untuk membuat sebuah sistem yang dapat digunakan untuk mendeteksi tingkat kecanduan *game* dibutuhkan perancangan yang baik. Konsep perancangan yang baik dapat menunjang dalam pembangunan sistem. Perancangan yang digunakan dalam penelitian ini menggunakan pendekatan berorientasi objek. Sehingga diagram-diagram yang digunakan adalah Diagram UML yang meliputi *usecase diagram, activity diagram, sequence diagram, dan class diagram*.

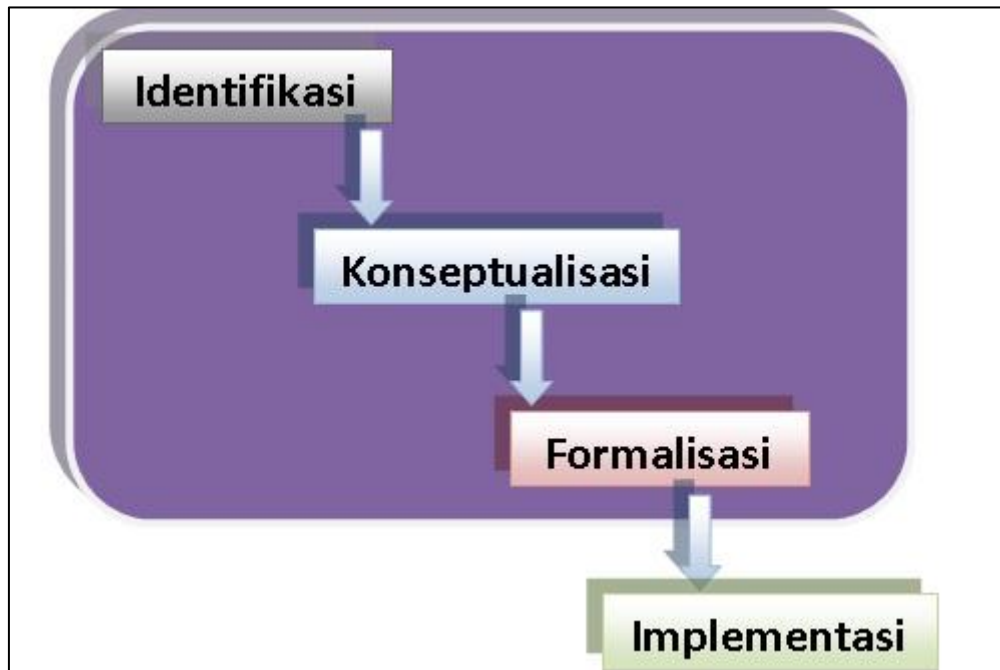
Sementara model yang digunakan dalam penelusuran identifikasi kecanduan *game* adalah model *backward chaining* yang telah dimodelkan sebelumnya dalam penelitian Latubessy, A dan Wijayanti, E(2017). *Backward chaining* merupakan salah satu model penelusuran dalam sistem pakar. Menurut T.Sutojo dkk, (2011) *Backward chaining* adalah metode yang inferensi yang bekerja mundur kearah kondisi awal. Proses diawali dari *goal* (yang berada dibagian *THEN* dari *rule IF-THEN*), kemudian pencari mulai dijalankan untuk mencocokkan apakah fakta-fakta yang ada cocok dengan premis-premis dibagian *IF*. Jika cocok, *rule* dieksekusi, kemudian hipotesis dibagian *THEN* ditempatkan di basis data sebagai fakta baru. Jika tidak cocok, simpan premis dibagian *IF* kedalam *stack* sebagai *SubGoal*. Proses berakhir jika *goal* ditemukan atau tidak ada *rule* yang bisa membuktikan kebenaran dari *SubGoal* atau *Goal*.

Untuk membangun sebuah sistem perangkat lunak yang baik diperlukan perancangan yang matang. Oleh sebab itu dalam penelitian ini dilakukan perancangan menggunakan diagram UML karena pendekatan yang digunakan untuk implementasi sistem menggunakan pendekatan berbasis objek. Diagram yang dihasilkan seperti *usecase diagram, activity diagram, sequence diagram dan class diagram*. Menurut Kendal, K.E dan Kendal, J.E(2011) Pemrograman berorientasi objek

berbeda dari pemrograman prosedural, dengan memeriksa objek-objek yang merupakan bagian dari sistem. Setiap objek pada sistem mewakili beberapa hal atau peristiwa yang sebenarnya.

## 2. METODOLOGI

Metodologi yang digunakan dalam pengembangan sistem adalah metode pengembangan sistem pakar. Metode pengembangan sistem pakar dimulai dari melakukan **identifikasi** masalah yang berhubungan dengan kasus kecanduan *game*. Pencarian fakta dilakukan pada tahap identifikasi. Setelah itu masuk pada tahap kedua dalam pengembangan sistem pakar yaitu **konseptualisasi** dimana dilakukan penyusunan relasi data serta penentuan model yang akan digunakan. Model sistem pakar yang dipilih adalah *backward chaining*. Tahap terakhir yang dilakukan dalam penelitian ini adalah **formalisasi**.

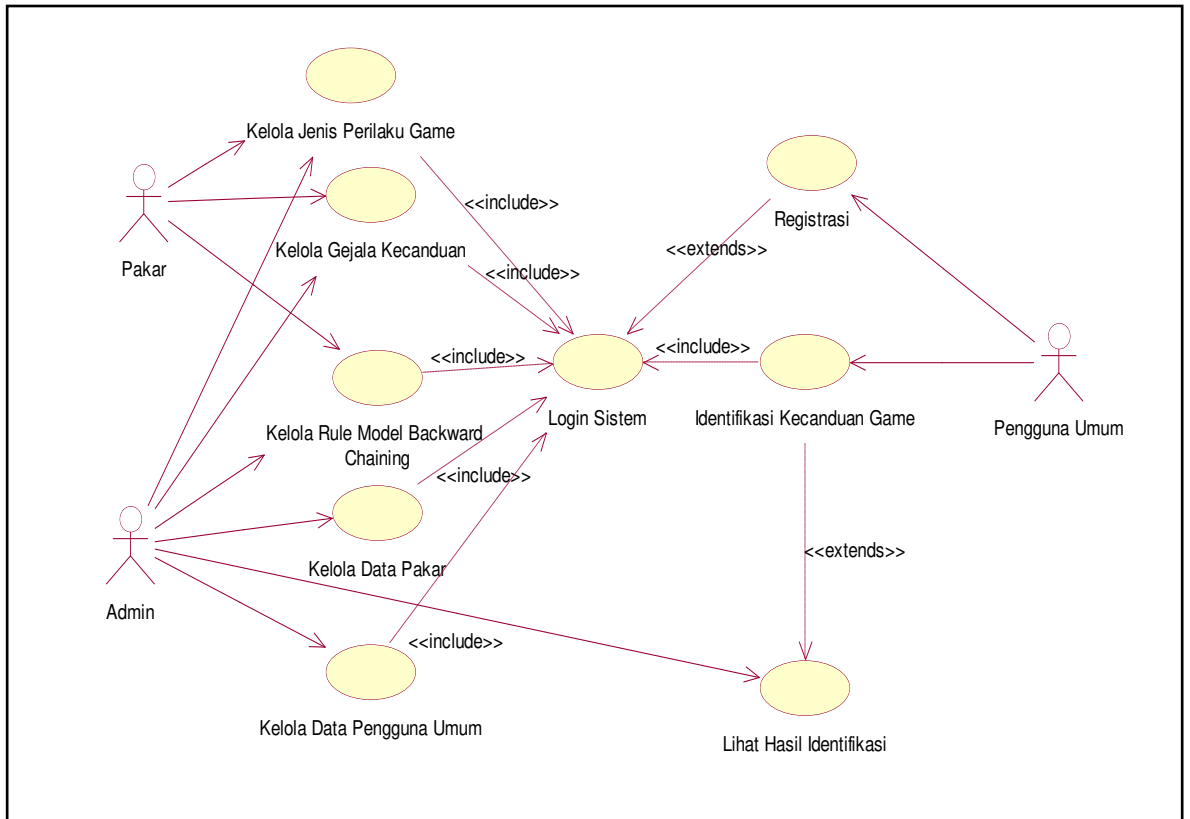


**Gambar 1. Tahapan Sistem Pakar yang Digunakan**

Penelitian ini berhenti pada tahap ketiga dikarenakan *output* penelitian yang dihasilkan adalah berupa rancangan diagram sistem. Seperti yang ditunjukkan pada Gambar 1, tahap yang dikerjakan dalam penelitian ini adalah bagian yang berada pada kotak berwarna ungu.

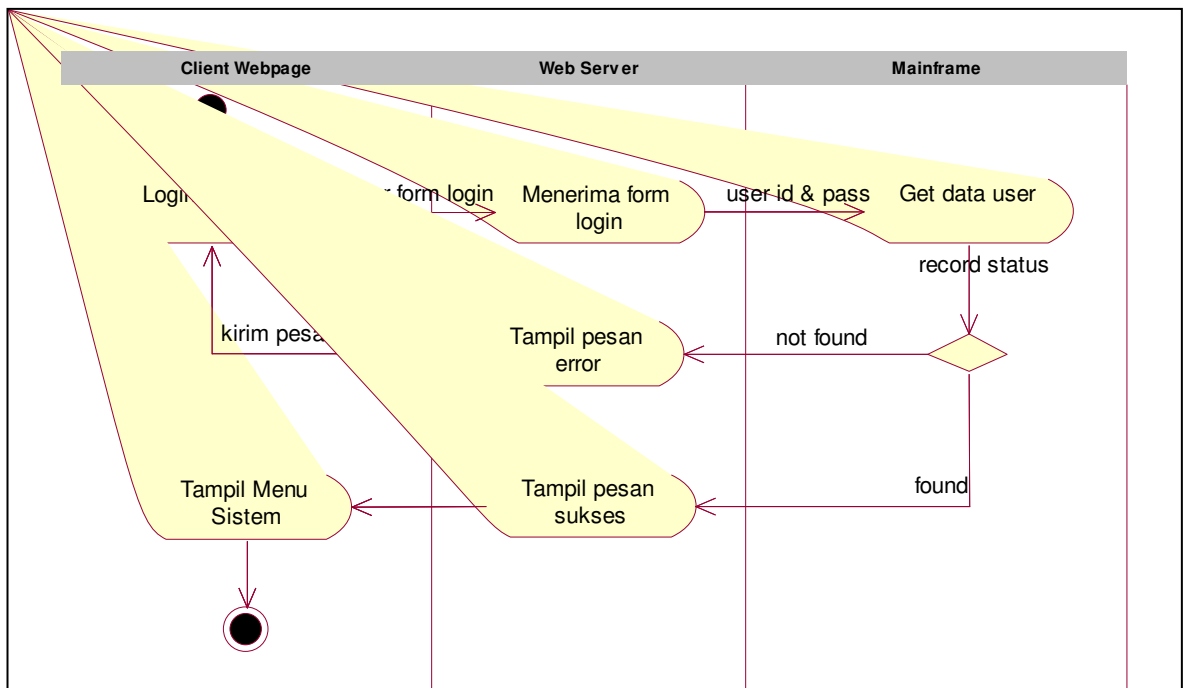
## 3. HASIL DAN PEMBAHASAN

Dilakukan perancangan *use case diagram* untuk mendeskripsikan bagaimana sistem akan digunakan. Terdapat tiga aktor yang dapat menggunakan sistem yaitu pakar, pengguna umum, dan admin. Pengguna umum dapat melakukan proses registrasi, proses identifikasi kecanduan *game* dan melihat hasil identifikasinya. Untuk *usecase* registrasi *extends usecase* login. Artinya, setelah pengguna umum melakukan registrasi pengguna umum tersebut dapat langsung melakukan proses *login* atau dapat melakukan *login* pada waktu yang lain. Setelah login berdasarkan *password* dan *username* maka pengguna umum dapat melakukan proses identifikasi dan melihat hasilnya. Sementara Pakar disini adalah seorang ahli psikologi yang bertugas melakukan kelola jenis perilaku kecanduan *game*, kelola gejala kecanduan, dan kelola *rule* model kecanduan. Sama halnya seperti pengguna umum, seorang pakar juga harus melakukan *login* sebagai pakar untuk dapat melakukan hal tersebut. Disamping itu, Admin yang memiliki level paling tinggi diantara ketiga pengguna ini dapat melakukan semua hal yang dilakukan dua aktor yang lain, ditambah kelola data pakar dan kelola data pengguna. Gambaran lebih jelasnya dapat dilihat pada Gambar 2.



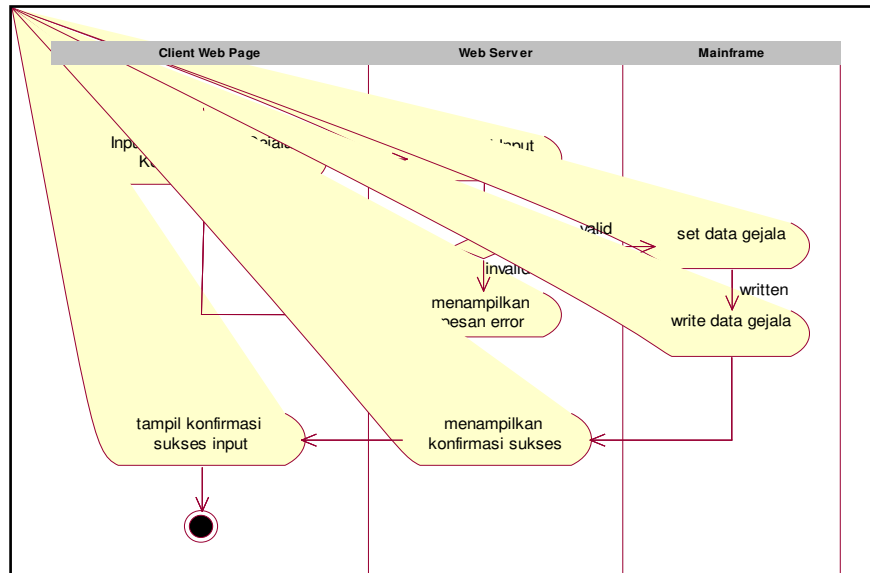
**Gambar 2. Usecase Diagram Sistem Identifikasi Kecanduan Game**

Langkah selanjutnya adalah merancang *activity diagram*. Setiap *usecase* akan menghasilkan setidaknya satu *activity diagram*. Jumlah *activity diagram* yang dihasilkan pada penelitian ini adalah sebanyak jumlah *usecase* yang ada pada *usecase diagram* yang ditunjukkan pada Gambar 2.

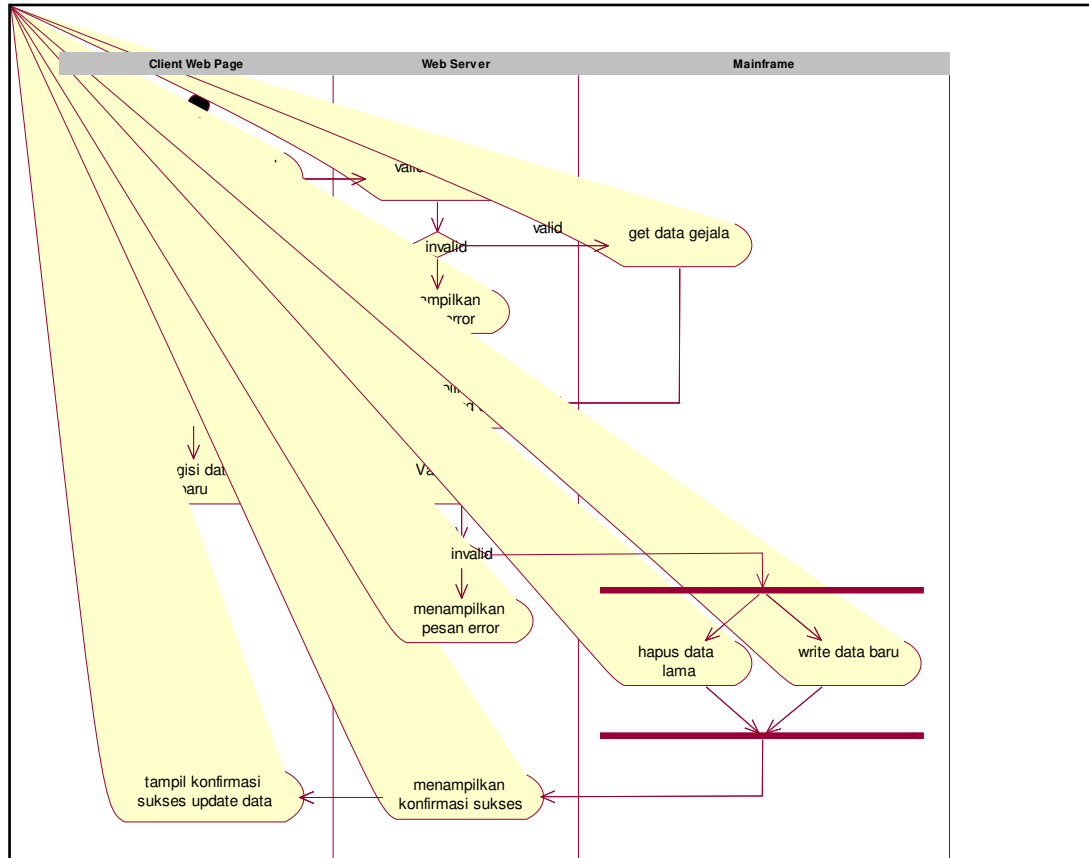


**Gambar 3. Activity Diagram Login Sistem Identifikasi Kecanduan Game**

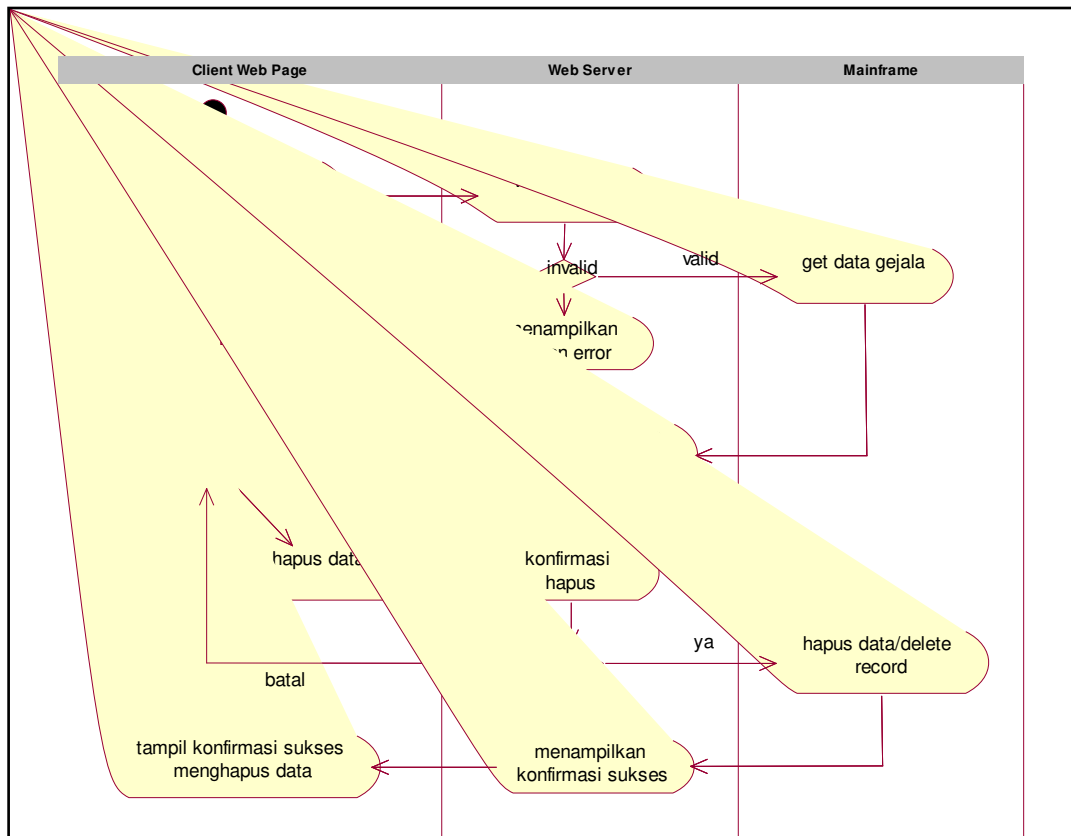
Terdapat lima *usecase* yang digunakan untuk kelola data seperti yang digambarkan pada Gambar 2. Sebagai contoh dibuat salah satu *activity diagram* yaitu *activity diagram* kelola data gejala kecanduan *game* seperti yang ditunjukkan pada Gambar 4, Gambar 5 dan Gambar 6. Kelola data meliputi proses *input*, *update*, dan *delete*. Untuk kelola data pakar, kelola data jenis perilaku, kelola data pengguna dan kelola *rule* pada dasarnya sama dengan aktifitas pada aktifitas yang ada pada kelola data gejala.



Gambar 4. Activity Diagram Input Gejala Kecanduan Game

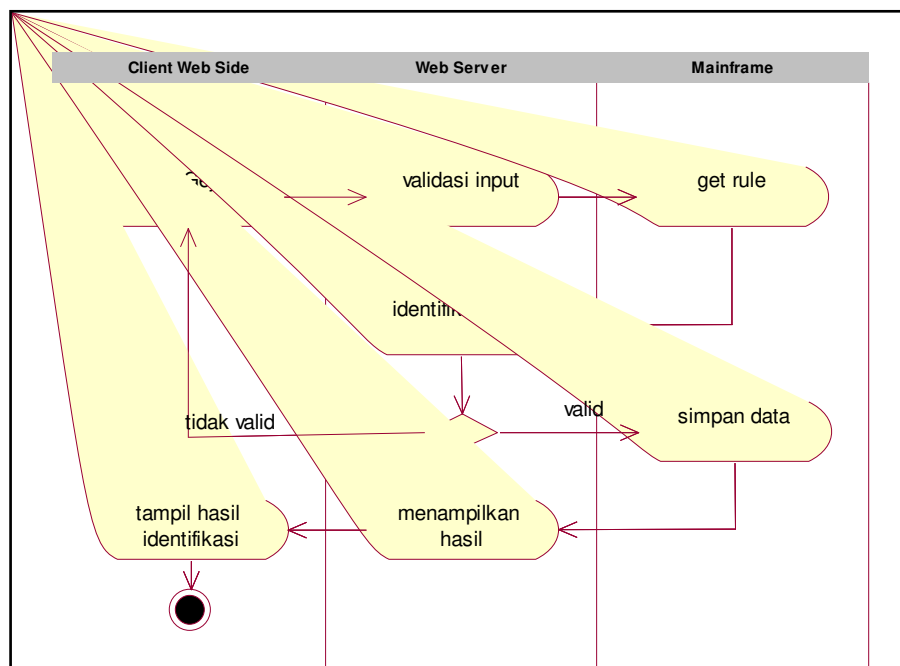


Gambar 5. Activity Diagram Update Gejala Kecanduan Game



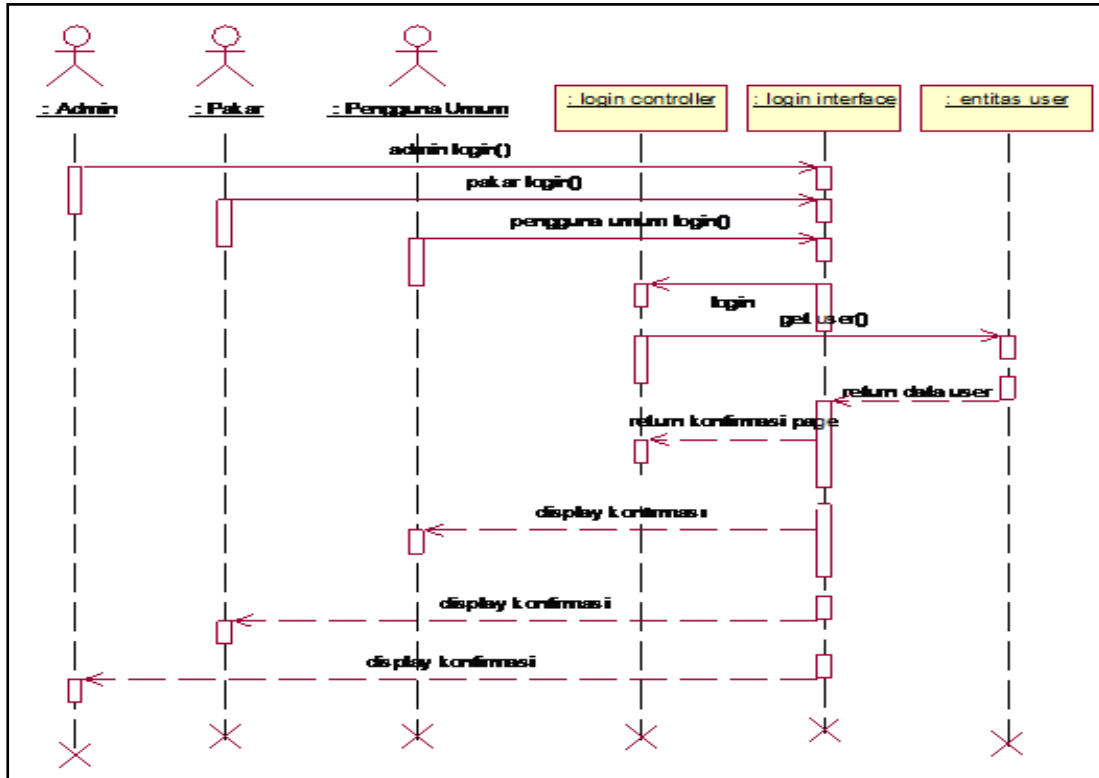
Gambar 6. Activity Diagram Hapus Gejala Kecanduan Game

Aktifitas lainnya adalah identifikasi aturan. Untuk aktifitas identifikasi aturan dapat dilihat pada Gambar 7. Proses identifikasi aturan disesuaikan dengan *rule* yang telah dibuat pada model identifikasi perilaku kecanduan *game* pada penelitian sebelumnya (Latubessy, A. dan Wijayanti, E., 2017).



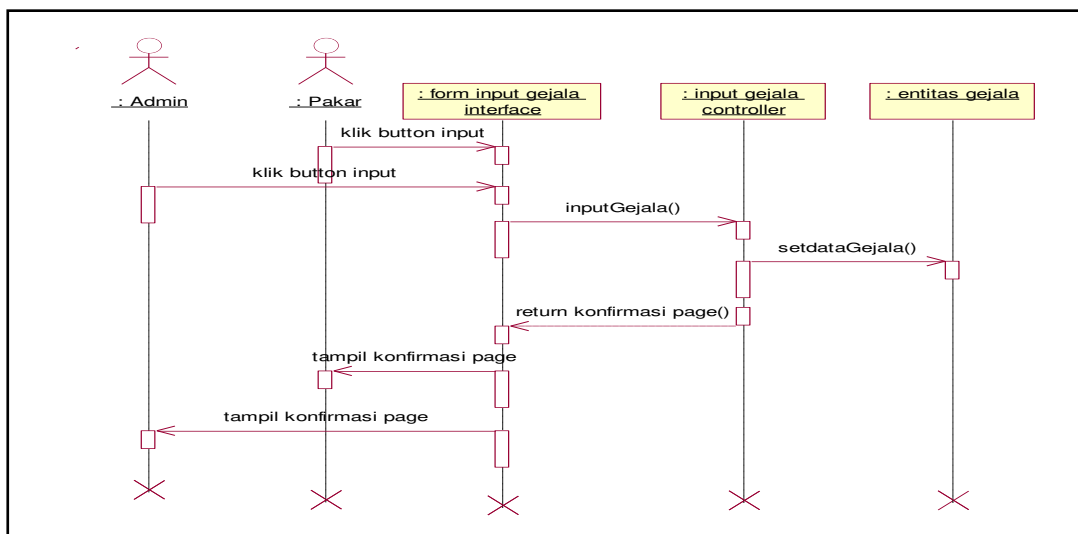
Gambar 7. Activity Diagram Identifikasi Aturan

Setelah membuat *activity diagram* maka langkah selanjutnya adalah membuat *sequence diagram*. Setiap *usecase* juga akan menghasilkan paling sedikit satu *sequence diagram*. Gambar 8 merupakan *sequence diagram* dari *usecase login*. Dimana aktor yang terlibat dalam *sequence* ini adalah admin, pakar, dan pengguna umum. Sementara *class* yang dihasilkan adalah *class login interface*, *login controller* dan entitas *user*.



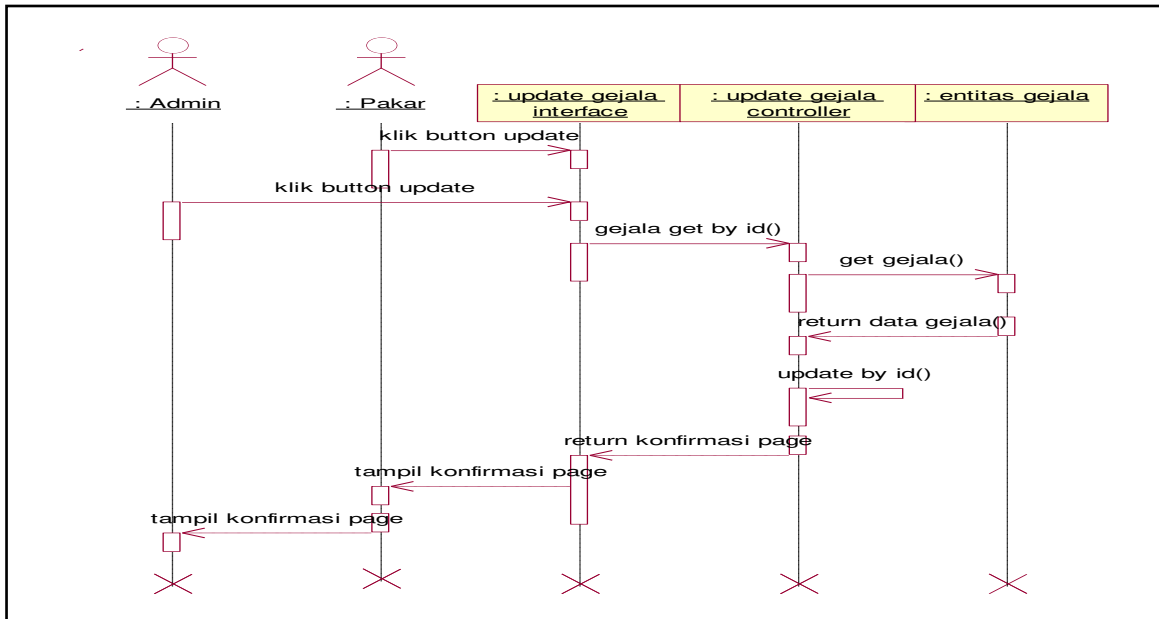
Gambar 8. Sequence Diagram Login Sistem

*Usecase* kelola gejala menghasilkan tiga *sequence diagram* yaitu input, hapus, dan *update* gejala. *Sequence diagram* input gejala ditunjukkan pada Gambar 9. Aktor yang terlibat adalah admin dan pakar. Sementara *class* yang dihasilkan adalah *class input gejala interface*, *input gejala controller*, dan entitas *gejala*.



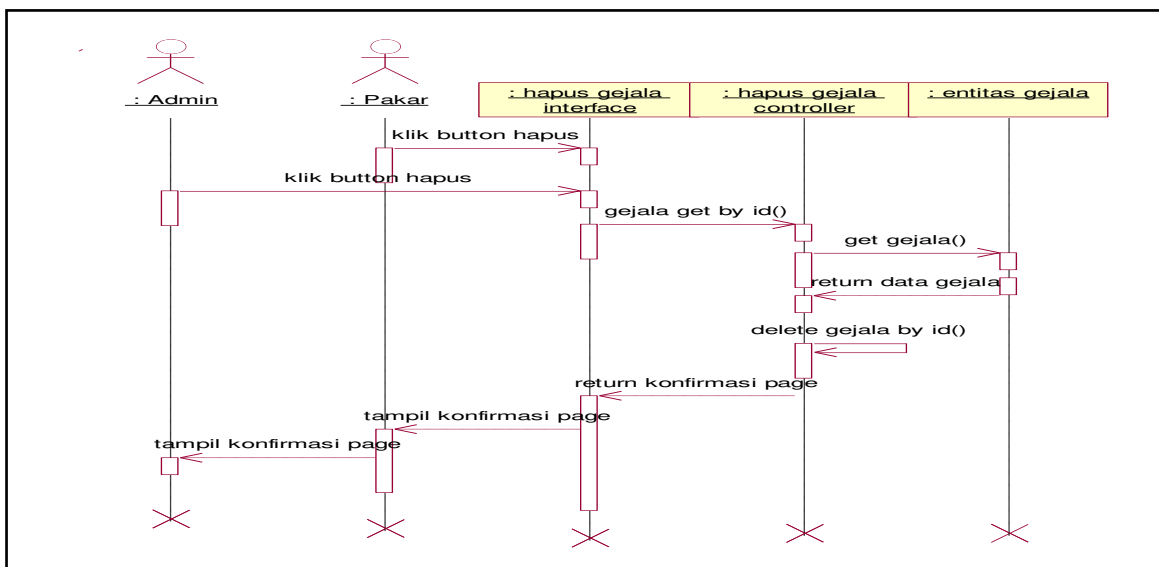
Gambar 9. Sequence Diagram Input Gejala

Sequence diagram yang kedua dari usecase gejala adalah sequence diagram update gejala. Melibatkan dua aktor yaitu admin dan pakar. Class yang dihasilkan adalah class update gejala interface, update gejala controller dan entitas gejala. Ditunjukkan pada Gambar 10.



Gambar 10. Sequence Diagram Update Gejala

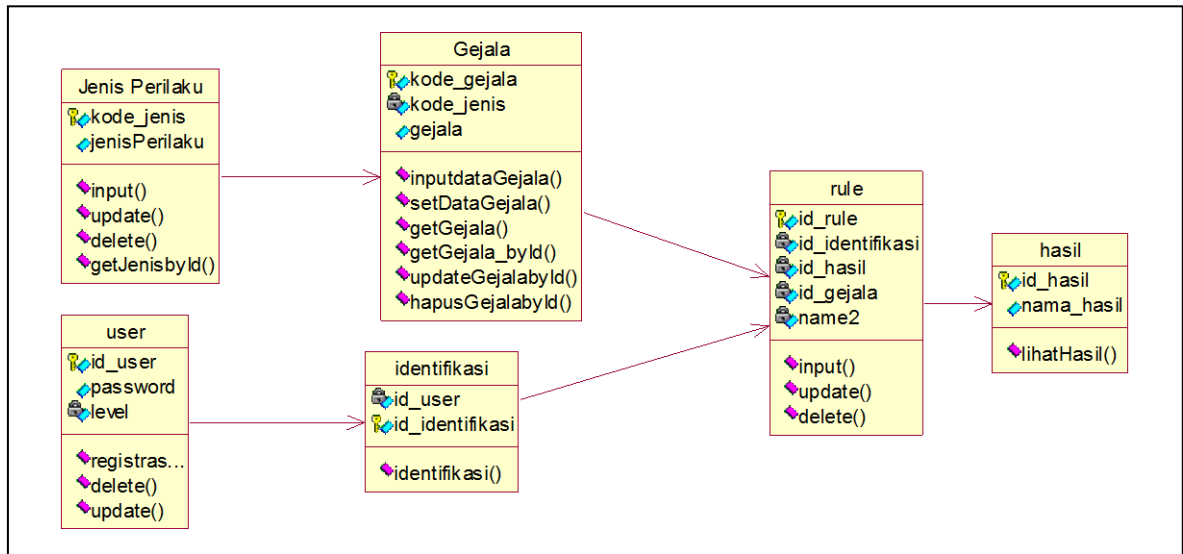
Sequence diagram yang ketiga dari usecase gejala adalah sequence diagram hapus gejala. Melibatkan dua aktor yaitu admin dan pakar. Class yang dihasilkan adalah class hapus gejala interface, hapus gejala controller dan entitas gejala. Ditunjukkan pada Gambar 11.



Gambar 11. Sequence Diagram Hapus Gejala

Gambar 3 sampai dengan Gambar 11 merupakan contoh dari beberapa activity dan sequence diagram yang dikembangkan. Namun, perlu diingat bahwa setiap usecase yang telah dirancang pada usecase diagram akan menghasilkan setidaknya satu activity diagram dan satu sequence diagram.

Langkah selanjutnya adalah membuat *Class diagram*. Gambar 12 menunjukkan *class diagram* yang dirancang untuk membangun sistem identifikasi kecanduan *game*. Sehingga diperoleh setidaknya enam class yang saling berhubungan seperti yang ada pada Gambar 12.



**Gambar 12. Class Diagram Sistem Identifikasi Kecanduan Game**

Perancangan yang telah dibuat dengan pendekatan objek ini telah menghasilkan beberapa diagram yang dapat digunakan sebagai dasar atau acuan bagi pembuatan sistem identifikasi kecanduan *game*. Pentingnya dokumentasi rancangan seperti ini dilakukan agar memudahkan proses pembuatan sistem. Dokumentasi yang rapi dapat mempengaruhi kinerja pengembangan sistem. Apabila nanti terjadi pengembangan sistem dan perubahan sistem maka dokumen perancangan sangat diperlukan.

#### 4. KESIMPULAN

Berdasarkan hasil dan pembahasan maka diperoleh beberapa kesimpulan sebagai berikut:

- (1) Proses identifikasi terhadap gejala-gejala jenis perilaku kecanduan *game* menghasilkan sembilan *usecase* dan tiga *actor* yang dapat menggunakan sistem.
- (2) Setiap *usecase* akan menghasilkan setidaknya satu *activity diagram* dan satu *sequence diagram*.
- (3) Dihasilkan rancangan sistem *berorientasi object* menggunakan diagram-diagram UML (*Unified Modeling Language*) untuk identifikasi jenis perilaku kecanduan *game*.

#### DAFTAR PUSTAKA

- Faried., (2012), Perilaku Remaja Pecandu Game Online. Skripsi, IKIP PGRI, Semarang.
- Latubessy, A. dan Ahsin, M.N., (2016), Hubungan Antara Adiksi Game Terhadap Keaktifan Pembelajaran Anak Usia 9-11 Tahun, *Jurnal Simetris*, 7. 2, 687-692.
- Latubessy, A. dan Wijayanti, E., (2017), Model Identifikasi Kecanduan Game Menggunakan Backward Chaining, *Jurnal Simetris*, 8. 1, 9-14.
- T. Sutojo, dkk, 2011, Kecerdasaan Buatan, Andi, Yogyakarta.