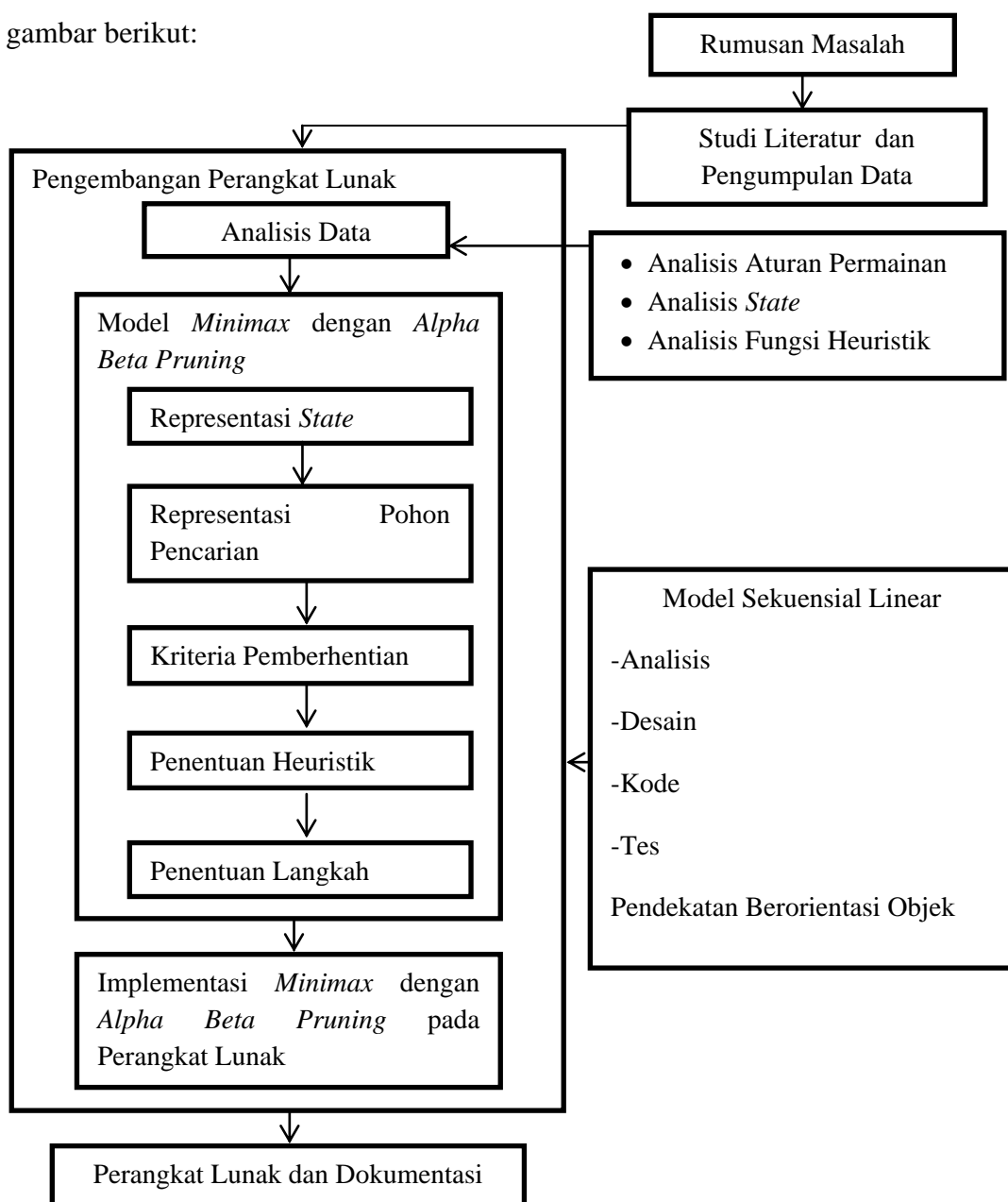


BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Desain dan tahapan yang dilakukan dalam penelitian ini dapat dilihat pada gambar berikut:



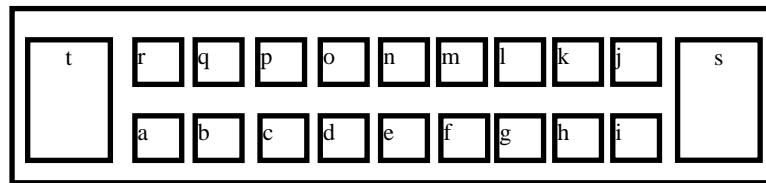
Gambar 3. 1 Desain Penelitian

Penjelasan dari gambar diatas adalah sebagai berikut:

1. Perumusan masalah merupakan dasar dari penelitian ini. Dalam penelitian ini, permasalahan yang dirumuskan meliputi rumusan aturan permainan, rumusan penentuan nilai *heuristik*, dan penentuan fitur yang akan diterapkan.
2. Data penelitian, merupakan acuan untuk melakukan perancangan dan pembangunan perangkat lunak.
3. Studi Literatur dilakukan dalam rangka pengumpulan data. Studi literatur digunakan untuk mempelajari tentang *Minimax* dengan *Alpha Beta Pruning* dan mempelajari tentang *ottjin*.
4. Pengembangan perangkat lunak, yaitu merupakan proses merancang dan membangun aplikasi. Perancangan dan pembangunan aplikasi ini menggunakan pendekatan berorientasi objek dengan model sekuensial linier sebagai siklus pengembangannya. Tahapan ini berisi analisis data, pembuatan model *Minimax* dengan *Alpha Beta Pruning*, dan proses implementasi kedalam perangkat lunak.
5. Tahapan analisis data berisi analisis – analisis yang dibutuhkan untuk pengembangan perangkat lunak. Tahapan analisis ini terdiri dari analisis aturan permainan, analisis *state*, dan analisis fungsi heuristik. Dari tahapan analisis ini kemudian menghasilkan desain yang kemudian digunakan untuk proses selanjutnya.

6. Pada tahap ini juga dilakukan penerapan *Minimax* dengan *Alpha Beta Pruning*. Untuk tahapan – tahapan lebih jelasnya:

1. Representasi *state* merupakan representasi atau penggambaran kondisi papan permainan saat ini. Representasi *state* digambarkan sebagai berikut:

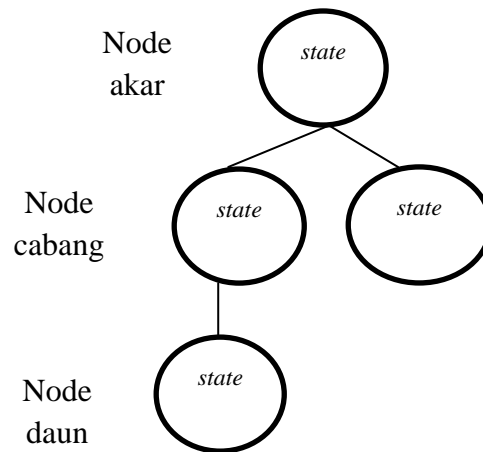


Gambar 3. 2 Representasi State

Untuk penjelasan gambar diatas:

1. posisi a sampai i merupakan lubang biji milik pemain komputer.
 2. posisi j sampai r merupakan lubang biji milik pemain manusia.
 3. Posisi s merupakan kantung biji milik pemain komputer.
 4. Posisi t merupakan kantung biji milik pemain manusia.
2. Representasi pohon pencarian merupakan penggambaran *state – state* ke dalam sebuah pohon pencarian. Setiap *state* dimasukkan ke dalam *node* pada pohon pencarian. Di dalam pohon pencarian ini kemudian *node – node* dikategorikan menjadi 3 bagian, yaitu *node* akar, *node* cabang, dan *node* daun. *Node* akar yang merupakan tingkatan paling awal dalam pohon pencarian, *node* cabang yang merupakan tingkatan antara *node* akar dan *node* daun, dan *node* daun yang merupakan

tingkatan paling akhir pada pohon pencarian. Sebagai ilustrasi dapat dilihat dari gambar dibawah ini.



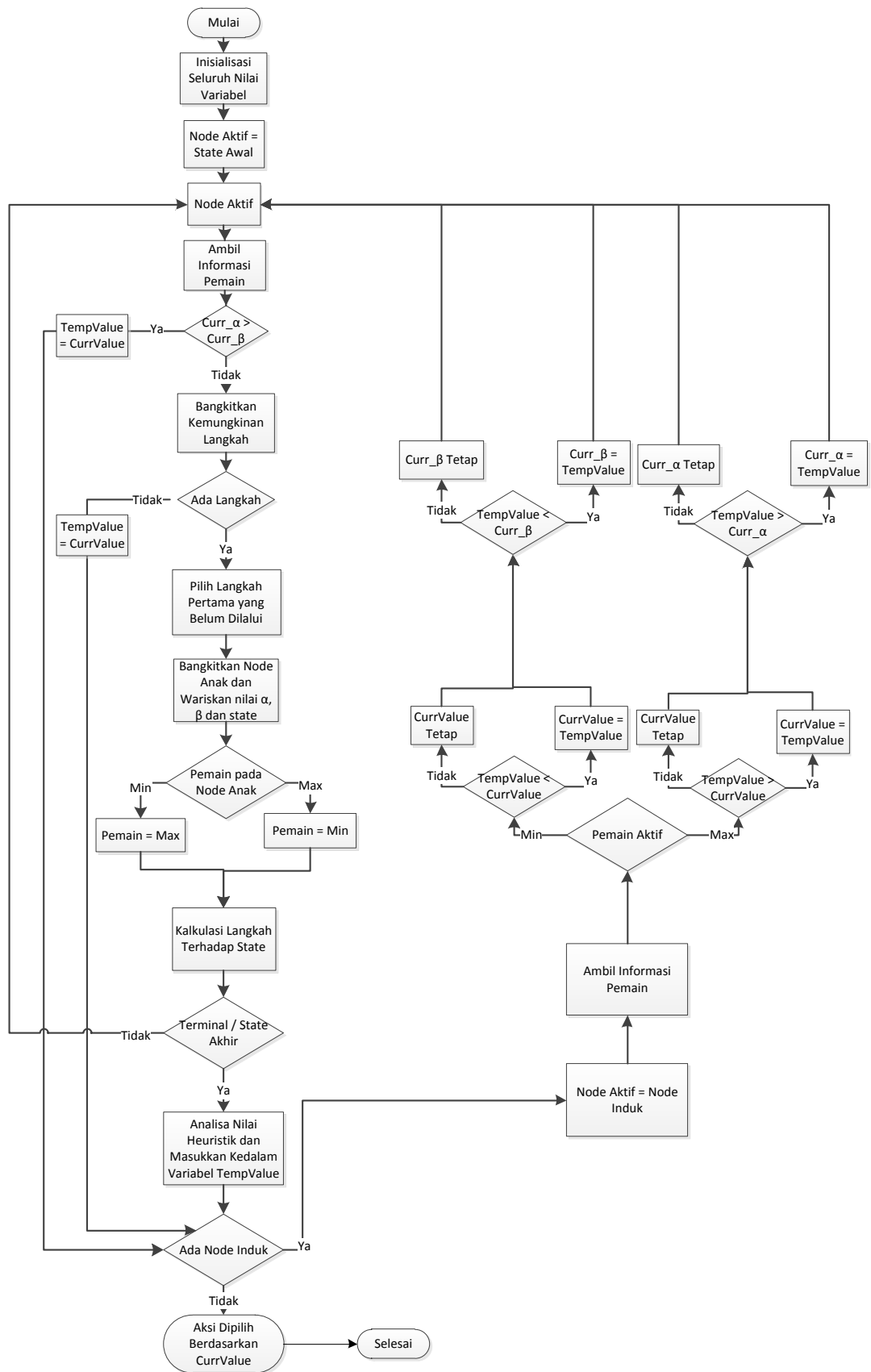
Gambar 3. 3Ilustrasi Pohon Pencarian

3. Kriteria pemberhentian berisi kriteria – kriteria yang dibutuhkan untuk menilai apakah giliran atau permainan sudah berakhir. Kriteria – kriteria pemberhentian apakah giliran sudah berakhir, dan kriteria – kriteria pemberhentian apakah permainan sudah berakhir pada permainan ottjin akan dibahas lebih lanjut pada bab 4.
4. Penentuan heuristik merupakan penentuan pemberian nilai terhadap *state* pada *node* daun. Fungsi heuristik yang dianalisis pada tahapan analisis kemudian dikalkulasikan terhadap *state* pada *node* daun yang kemudian dihasilkan nilai heuristik. Nilai heuristik ini berfungsi sebagai bobot untuk menentukan penilaian pada *node* – *node* di atasnya pada proses penentuan langkah menggunakan *Minimax* dengan *Alpha Beta Pruning*. Sebagai ilustrasi fungsi heuristik pada permainan papan bisa dilihat sebagai berikut:

Nilai Heuristik = Skor Sementara + Bobot strategi

Di mana bobot strategi spesifik dari permainan dan analisisnya.

5. Penentuan langkah merupakan proses inti dari *Minimax* dengan *Alpha Beta Pruning*. Proses ini menghasilkan langkah terbaik dari langkah – langkah yang ada. Langkah terbaik inilah yang dipilih sebagai langkah selanjutnya untuk pemain komputer. Untuk mengetahui lebih jelas bagaimana proses penentuan langkah ini berjalan, dapat dilihat dari flowchart berikut ini:



Gambar 3. 4 Flowchart Minimax dengan Alpha Beta Pruning

Penjelasan dari flowchart diatas adalah sebagai berikut:

1. *State* awal yang merupakan *state* yang akan dicari langkah terbaiknya menjadi input.
2. Seluruh variabel pendukung seperti CurrValue, TempValue, Curr_α, Curr_β diinisialisasi dengan nilai awal.
 - CurrValue : nilai heuristik pada *node* tertentu.
 - TempValue : kontainer nilai heuristik sementara.
 - Curr_α : nilai α pada *node* tertentu. Dimana α adalah nilai heuristik terbesar yang ada pada pemain max.
 - Curr_β : nilai β pada *node* tertentu. Dimana β adalah nilai heuristik terkecil yang ada pada pemain min.
3. *State* awal kemudian menjadi *node* aktif yang memiliki atribut Curr_α dan Curr_β yang telah diinisialisasi.
4. *Node* aktif terpilih.
5. Ambil informasi pemain pada *node* aktif.
6. Lakukan pengecekan nilai α dan β pada *node* aktif. Apabila $\alpha > \beta$ maka dilakukan *Pruning* dengan tidak membangkitkan *node* anak dan kemudian melanjutkan ke proses pemilihan keputusan pada proses no. 14 dan TempValue = CurrValue.
7. Apabila kondisi $\alpha > \beta$ tidak terpenuhi, maka dilanjutkan ke proses pembangkitan langkah yang tersedia pada *node* aktif.

8. Apabila langkah tersedia maka langkah pertama yang belum dikunjungi dipilih. Apabila langkah tidak ada maka dilanjutkan ke proses pemilihan keputusan pada proses no. 14 dan $TempValue = CurrValue$.
9. Pada proses ini dibangkitkan *node* anak yang berisi *state* dan atribut α dan β yang diwariskan dari *node* induk. *Node* yang aktif sekarang adalah *node* anak.
10. Info pemain pada *node* anak diberikan. Apabila pemain *node* induk adalah min, maka *node* anak adalah max. Begitu pula sebaliknya.
11. Langkah yang sudah dipilih pada proses no. 7 dikalkulasi pada *node* anak.
12. Pengecekan kondisi terminal / *state* akhir. Kondisi terminal adalah kondisi di mana permainan berakhir atau langkah pemain habis. Apabila kondisi terminal terpenuhi, maka dilanjutkan ke proses analisa heuristik. Apabila kondisi tidak terpenuhi, maka dilakukan kembali proses no. 4 dan seterusnya.
13. Analisa heuristik terhadap *node* aktif. Nilai heuristik yang didapat kemudian dimasukkan ke dalam variabel $TempValue$.
14. Dilanjutkan ke proses pemilihan keputusan apakah *node* aktif sekarang memiliki *node* induk. Apabila ada maka *node* yang aktif sekarang adalah *node* induk.

15. Ambil informasi pemain pada *node* aktif. Dan lakukan keputusan berikut.

- Pemain min: apabila $\text{TempValue} < \text{CurrValue}$ maka $\text{CurrValue} = \text{Temp Value}$, kalau tidak maka CurrValue tidak diubah. Apabila $\text{TempValue} < \beta$ maka $\beta = \text{CurrValue}$, kalau tidak maka α tidak diubah.
- Pemain max: apabila $\text{TempValue} > \text{CurrValue}$ maka $\text{CurrValue} = \text{Temp Value}$, kalau tidak maka CurrValue tidak diubah. Apabila $\text{TempValue} > \alpha$ maka $\alpha = \text{CurrValue}$, kalau tidak maka α tidak diubah.

Kemudian kembali ke proses no. 4 dan seterusnya.

16. Apabila seluruh langkah yang mungkin sudah dikalkulasi maka langkah dipilih berdasarkan nilai CurrValue yang didapat pada *node* induk.

Proses penentuan langkah dengan membangkitkan langkah – langkah yang memungkinkan ini menggunakan pohon pencarian *Depth First Search* (DFS).

7. Implementasi perangkat lunak dilakukan setelah proses analisis dan pemodelan selesai. Proses implementasi perangkat lunak dilakukan berdasarkan model yang dibuat sebelumnya. Hasil dari implementasi ini adalah permainan ottjin yang dapat dimainkan oleh dua pemain, satu manusia dan satu komputer.

8. Hasil akhir berupa perangkat lunak dan dokumentasi. Hasil penelitian kemudian dimasukkan ke dalam dokumentasi yang diantaranya dokumen skripsi, dokumen teknis perangkat lunak, dan paper.

3.2 Metode Penelitian

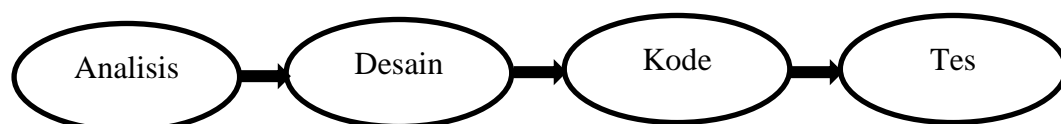
Dalam penelitian ini metode yang digunakan dalam penelitian meliputi metode pengumpulan data dan pengembangan perangkat lunak.

3.2.1 Metode Pengumpulan Data

Metode pengumpulan data dalam penelitian ini adalah studi literatur, yang bertujuan sebagai acuan dalam melakukan penelitian seperti diantaranya paper, buku, website, blog, dan artikel - artikel yang berhubungan dalam penelitian ini.

3.2.2 Metode Pengembangan Perangkat Lunak

Metode pengembangan perangkat lunak yang digunakan adalah model sekuensial linear. Model sekuensial linear adalah proses pengembangan perangkat lunak yang mengalir secara bertahap melalui analisis, desain, kode, dan tes (Pressman, 2001). Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.



Gambar 3. 5 Model Sekuensial Linear (Pressman, 2001).

Penjelasan untuk gambar diatas adalah sebagai berikut:

1. Analisis

Tahap ini dilakukan untuk mencari seluruh kebutuhan dalam membangun perangkat lunak. Diantaranya data – data yang dibutuhkan, fungsi – fungsi yang dibutuhkan, tingkah laku yang akan dimiliki, dan *interface*. Dalam penelitian ini analisis yang dilakukan antara lain analisis aturan permainan, analisis *state*, dan analisis fungsi heuristik.

2. Desain

Tahap ini dilakukan untuk membuat rancangan perangkat lunak berdasarkan kebutuhan yang didapat saat tahap analisis. Dalam penelitian ini dibuat desain *Minimax* dengan *Alpha Beta Pruning*.

3. Kode

Pada tahap ini dilakukan implementasi dari tahap – tahap sebelumnya ke dalam bahasa pemrograman yang menghasilkan sebuah perangkat lunak yang dapat digunakan. Dalam penelitian ini dilakukan implementasi dari desain *Minimax* dengan *Alpha Beta Pruning* ke dalam bahasa pemrograman java.

4. Tes

Tahap ini dilakukan untuk proses pengujian terhadap perangkat lunak yang telah dibuat. Pada tahap ini dilakukan pengujian eksternal atau yang biasa disebut dengan blackbox testing. Pada blackbox dilakukan pengujian terhadap fungsi – fungsi interaksi dengan pengguna, seperti interaksi terhadap tombol – tombol yang fungsional serta apakah output sesuai dengan yang diharapkan.

Pada proses pengembangan perangkat lunak ini dilakukan pendekatan berorientasi objek yang artinya segala permasalahan digambarkan ke dalam objek yang memiliki perilaku tertentu. Untuk penggambaran dalam pendekatan berorientasi objek digunakan *Unified Modelling Language* (UML). Diagram – diagram UML yang digunakan diantaranya *use case diagram*, *activity diagram*, dan *class diagram*. *Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem atau dengan kata lain menggambarkan apa saja yang dapat dilakukan oleh sistem dari sudut pandang aktor. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain. *Activity diagram* menggambarkan aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, keputusan yang mungkin terjadi, dan bagaimana mereka berakhir.

3.3 Alat dan Bahan Penelitian

Berikut ini merupakan penjelasan mengenai alat – alat dan bahan yang digunakan untuk penelitian.

3.3.1 Alat

Spesifikasi komputer yang digunakan dalam penelitian ini adalah sebagai berikut:

- Prosesor *Intel Pentium 2,2 Ghz*
- RAM 4 GB
- HDD 320 GB
- Monitor dengan resolusi 1366 x 768 *pixel 32 bit*
- *Mouse dan Keyboard*
- Sistem Operasi menggunakan *Microsoft Windows 7*

Sedangkan perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut:

- Java Development Kit dan Java Runtime Environment 7
- Eclipse IDE
- Sistem Operasi Microsoft Windows 7

3.3.2 Bahan

Bahan yang digunakan dalam penelitian ini antara lain data – data mengenai ottjin seperti peraturan – peraturan dan cara bermain dan algoritma *Minimax* dengan *Alpha Beta Pruning* yang berasal dari literatur.