# BAB 1
# PENDAHULUAN

## 1.1 Latar Belakang Masalah

Penelitian Subakti menyatakan bahwa implementasi algoritma untuk NPC (*Non-Player Characters*) yang menggunakan *pathfinding* dalam permainan *tower defense* dilakukan berdasarkan algoritma pencarian jalur A* secara *real time* pada peta permainan berbasis *grid*. Dan menyatakan bahwa peggunaan *pathfinding* dapat diubah, agar terdapat berbagai macam *enemy*, misalnya ada *enemy* yang berjalan dengan memperhitungkan bobot *tower* yang menghadang dalam peta permainan[1].

Algoritma A* memiliki keunggulan dalam pencarian *pathfinding* di suatu peta, namun A* mempunyai kelemahan pada perubahan keadaan peta serta kepadatan dari sebuah rintangan[2]. Sedangkan *game tower defense* merupakan *game* berjenis RTS (*Real-Time Strategy*) yang memungkinkan terjadinya perubahan keadaan peta serta memperhitungkan bobot *tower* sehingga penggunaan algoritma A* kurang tepat untuk menyelesaikan permasalahan tersebut.

Algoritma *neural network backpropagation* merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan *neuron-neuron* yang ada pada lapisan tersembunyinya[3]. Selain itu pada penelitian lain tentang algoritma *neural network backpropagation* dengan masalah perbandingan algoritma tersebut dengan yang lain berkesimpulan lebih unggul dalam akurasinya.

NPC *enemy* yang berjalan dengan memperhitungkan bobot *tower* dibutuhkan ketepatan dalam memilih setiap langkahnya dan untuk mengetahui apakah langkah tersebut sudah benar atau tidak harus dilakukan pengujian akurasi. Sehingga penelitian ini bertujuan untuk mengetahui akurasi tingkat keberhasilan algoritma pada penentuan langkah NPC *enemy*.

Berdasarkan hal tersebut diatas maka penelitian ini akan membahas bagaimana mengimplementasikan algoritma *neural network backpropagation* di NPC *enemy* sehingga dapat menentukan langkah yang tepat pada *game tower defense*.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah disebutkan di atas, permasalahan yang dapat di rumuskan adalah bagaimana cara menerapkan algoritma *neural network backpropagation* dalam menentukan langkah NPC *enemy*.

## 1.3 Maksud dan Tujuan

Berdasarkan permasalahan diatas maksud dari penelitian tugas akhir ini adalah pengimplementasian algoritma *neural network backpropagation* pada NPC *enemy* untuk menentukan langkah di *game tower defense*.

Sedangkan tujuan dari penelitian ini untuk mengetahui akurasi tingkat keberhasilan algoritma *neural network backpropagation* pada NPC *enemy* dalam menentukan langkahnya di *game tower defense*.

## 1.4 Batasan Masalah

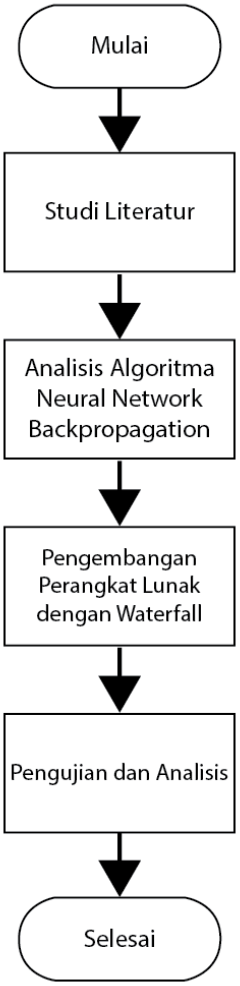
Untuk menghindari pembahasan yang meluas, maka permasalahan pada penelitian ini dibatasi sebagai berikut:

1. Algoritma yang digunakan adalah *artificial neural network.*
2. Metode *neural network* yang digunakan adalah *backpropagation.*
3. Implementasi *neural network backpropagation* pada NPC *enemy.*
4. Variabel yang digunakan untuk menentukan langkah adalah jarak *enemy* menuju ke tujuan, jarak *enemy* menuju ke *tower*, dan tipe *tower*.
5. *Enemy* melangkah dari titik awal ke titik tujuan, tanpa melakukan serangan.
6. *Tower* terdiri dari 4 tipe.
7. Pemodelan yang digunakan dalam laporan penelitian adalah *Object Oriented Programming* (OOP).
8. Sistem yang akan dibangun berbasis *desktop*.

9. Bahasa pemrograman yang digunakan adalah C#.

## 1.5 Metodologi Penelitian

Metode penelitian yang digunakan dalam tugas akhir ini adalah metode kuantitatif. Disebut juga metode *discovery*, karena dengan metode ini dapat ditemukan dan dikembangkan berbagai iptek baru. Disebut kuantitatif karena data-data berupa angka-angka dan menggunakan statistik[4].




**Gambar 1.1 Metode Penelitian**

### 1.5.1 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan adalah sebagai berikut:

1. Studi Literatur

Studi Literatur adalah mengumpulkan data melalui buku-buku tentang *artificial intelligence*, *artificial neural network*, situs internet tentang *tower defense*, jurnal dan artikel yang besangkutan dengan permasalan yang diteliti.

2. Analisis Algoritma

   Melakukan analisis algoritma *neural network* tentang bagaimana cara kerja metode *backpropagation*.

### 1.5.2 Metode Pengembangan Perangkat Lunak

Metode yang digunakan dalam pengembangan perangkat lunak adalah metode *waterfall* yang meliputi beberapa proses sebagai berikut[5]:

1. *Communication*

   Tahap *Communication* merupakan tahap analisis kebutuhan sistem serta tahap mengadakan pengumpulan data dengan melakukan pertemuan dengan pihak instansi yang terkait, maupun mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

2. *Planning*

   Tahap ini merupakan tahap lanjutan dari proses *communication*. Tahap ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan data yang berhubungan dengan keinginan pengguna dalam pembangunan *software*, termasuk dengan rencana yang akan dilakukan.

3. *Modeling*

   Pada tahap modeling ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum proses pengkodean. Proses ini terfokus pada rancangan struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*

4. *Construction*

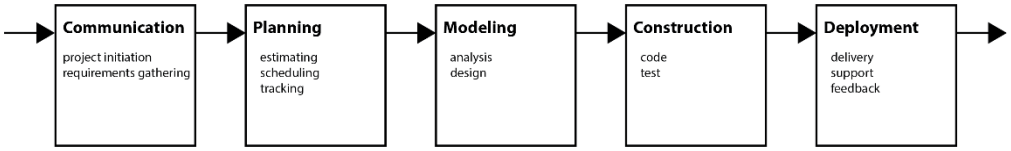
   Tahap ini merupakan tahap pengkodean yaitu tahap menerjemahkan desain dalam bahasa yang dikenali oleh komputer. *Programmer* akan menerjemahkan transaksi yang diminta oleh pengguna. Tahapan inilah yang merupakan tahap secara nyata dalam mengerjakan suatu *software*, artinya

penggunaan komputer akan dimaksimalkan dalam tahap ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibangun. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

5. *Deployment*

   Tahap ini bisa dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisis, desain dan pengkodean, maka sistem yang sudah jadi akan digunakan oleh pengguna. Kemudian *software* yang telah dibangun harus dilakukan pemeliharaan secara berkala.

Penggambaran model *waterfall* dapat dilihat pada gambar dibawah ini.




**Gambar 1.2 Metode *Waterfall***

### 1.5.3 Metode Pengujian

Metode yang digunakan untuk pengujian adalah *Confusion Matrix* diantaranya dapat mengukur hal berikut:

1. *Accuracy*

   Merupakan persentase dari prediksi yang benar.

2. *Precision*

   Adalah ukuran akurasi dari suatu kelas tertentu yang telah diprediksi.

3. *Recall*

   Merupakan persentase dari data dengan nilai positif yang nilai prediksinya juga positif.

4. *F-measure*

   Merupakan kombinasi rata-rata *harmonic* (*weighted harmonic mean*) dari *precision dan recall*.

## 1.6 Sistematika Penulisan

Sistematika penulisan ini untuk memberikan gambaran secara umum mengenai permasalahan serta pemecahannya. Sistematika penulisan ini dibagi dalam beberapa bab berikut:

**BAB 1 PENDAHULIAN**

Bab ini membahas mengenai latar belakang, rumusan masalah, menentukan maksud dan tujuan, batasan masalah, metodologi penelitian, serta sistematika penulisan untuk menjelaskan pokok pembahasannya.

**BAB 2 LANDASAN TEORI**

Bab ini akan menjelaskan mengenai objek dari penelitain, dan teori-teori pendukung yang berhubungan dengan pembangunan sistem.

**BAB 3 ANALISIS ALGORITMA**

Bab ini berisi tentang tahapan untuk menganalisis algoritma *neural network* metode *backpropagation* untuk menentukan langkah NPC *enemy* pada *game tower defense*.

**BAB 4 IMPLEMENTASI DAN PENGUJIAN**

Bab ini berisi mengenai implementasi dan analisis dari perancangan sistem yang dilakukan, uji coba dan hasil pengujian sistem.

**BAB 5 KESIMPULAN DAN SARAN**

Bab ini berisi tentang kesimpulan yang diperoleh dari hasil pengujian sistem serta saran untuk pengembangan sistem kedepan.