

## BAB 4

### IMPLEMENTASI DAN PENGUJIAN

#### 4.1 Implementasi

Tahap implementasi merupakan tahap yang dilakukan setelah tahap analisis dan perancangan system selesai dilakukan. Hasil analisis dan perancangan akan diimplementasikan kedalam bentuk program, pengimplementasian terdiri dari penjelasan mengenai lingkungan implementasi, yaitu lingkungan perangkat keras, perangkat lunak dan antarmuka.

##### 4.1.1. Implementasi Perangkat Keras

Spesifikasi perangkat keras (*hardware*) yang digunakan pada implementasi dalam penelitian ini dapat dilihat pada Tabel 4.1 Perangkat keras yang digunakan.

**Tabel 4.1 Perangkat keras yang digunakan**

<i>Hardware</i>	<b>Spesifikasi</b>
<i>Processor</i>	Intel i5-4200U 1.60 GHz ~ 2.3 GHz (4 CPUs)
<b>RAM</b>	4 GB
<b>Harddisk</b>	500 GB
<i>Display</i>	LED 14" (366x768)
<i>Keyboard dan Mouse</i>	<i>Standard</i>

##### 4.1.2. Implementasi Perangkat Lunak

Spesifikasi dari perangkat lunak (*software*) yang digunakan pada implementasi dalam penelitian ini dapat dilihat pada Tabel 4.2 Perangkat lunak yang digunakan.

**Tabel 4.2 Perangkat lunak yang digunakan**

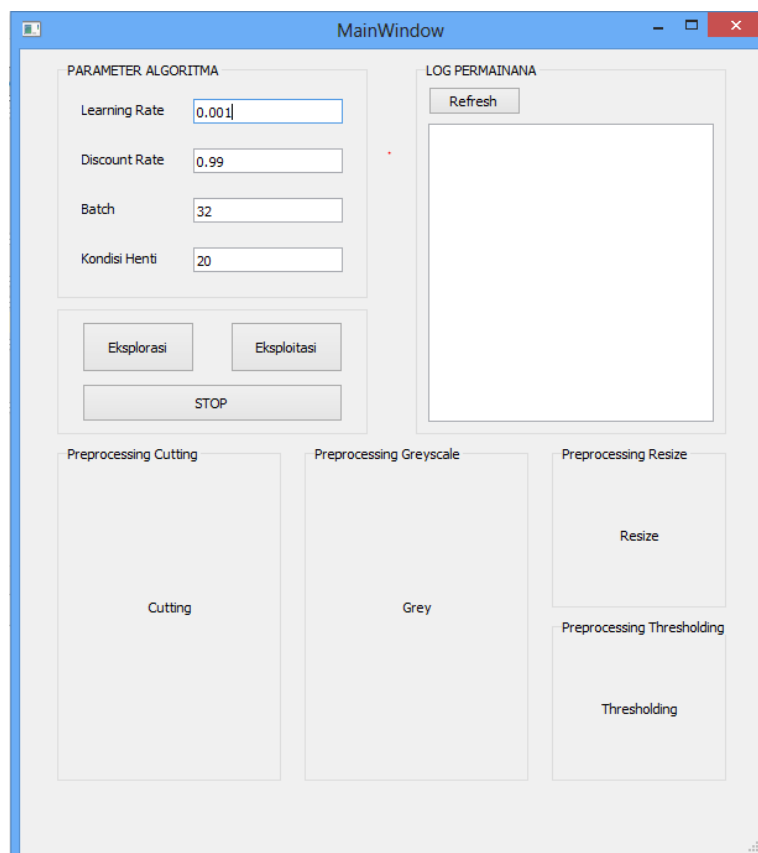
<i>Software</i>	<b>Spesifikasi</b>
<i>Operating System</i>	Windows 8
<i>Bahasa Pemrograman</i>	Python 3.6
<i>Libraries</i>	Keras 2.0, TensorFlow 2.0, PyQt5, OpenCV 4.1.0 dan PyGame
<i>Code Editor</i>	PyCharm

### 4.1.3. Implementasi Antarmuka

Adapun implementasi dari antarmuka pada aplikasi yang dibangun adalah sebagai berikut:

#### 1. Antarmuka Menu Utama

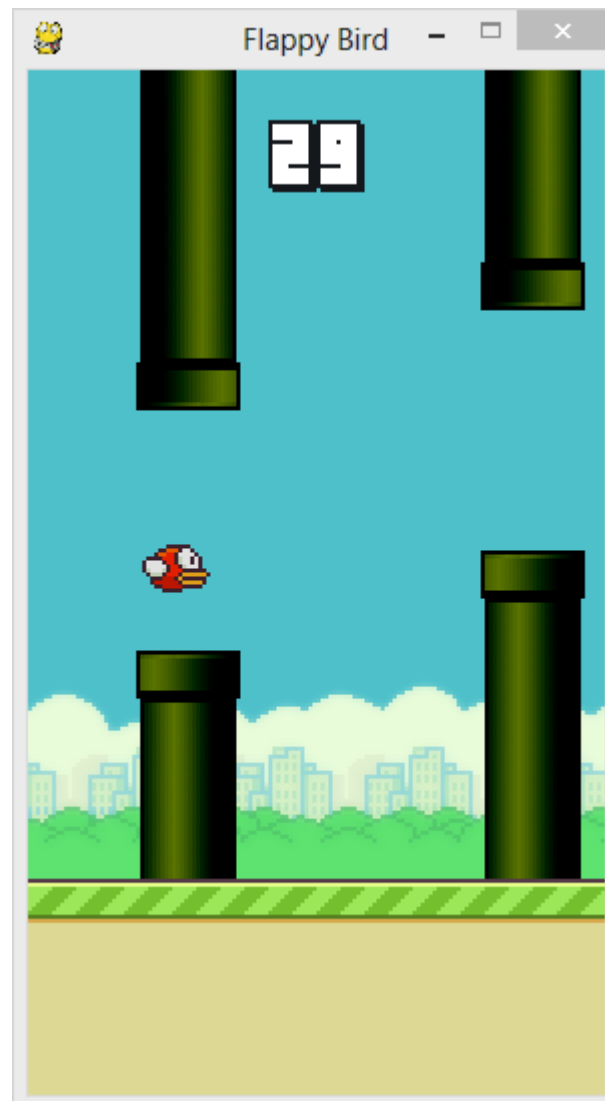
Antarmuka menu utama merupakan tampilan awal dari system yang dibangun, antarmuka menu utama dapat dilihat pada Gambar 4.1 Antarmuka Menu Utama.



**Gambar 4.1 Antarmuka Menu Utama**

## 2. Antarmuka Permainan

Antarmuka permainan merupakan tampilan permainan *flappybird* yang dimainkan secara otomatis oleh *agent*, antarmuka permainan dapat dilihat pada Gambar 4.2 Antarmuka Permainan.



**Gambar 4.2 Antarmuka Permainan**

## 4.2 Pengujian

Tahap pengujian ini bermaksud untuk mengetahui hasil dari implementasi yang dilakukan untuk selanjutnya ditarik sebuah kesimpulan dari penelitian yang dilakukan. Pengujian yang akan dilakukan terdiri dari dua tahap, yaitu:

### 1. Pengujian Fungsional

Pengujian fungsional dilakukan untuk memastikan bahwa aplikasi yang dibangun dapat berjalan dengan benar sesuai dengan tujuan fungsional yang diharapkan.

### 2. Pengujian Performansi

Pengujian performansi dilakukan untuk mengetahui peforma dari algoritma yang diterapkan dengan menguji setiap parameter yang digunakan pada algoritma untuk mencari kombinasi parameter yang optimal sehingga dapat diketahui rata-rata jumlah percobaan proses eksplorasi terkecil dan skor terbesar yang dapat diraih *agent*. Pada pengujian parameter algoritma, yang akan diujikan antara lain *learning rate*, *discount rate*, ukuran *batch* dan kondisi henti dengan parameter ujinya, yaitu jumlah percobaan pada proses *eksplorasi* dan skor yang didapatkan *agent*.

#### 4.2.1. Pengujian Fungsional

Pengujian ini dilakukan untuk mengetahui fungsionalitas dari perangkat lunak apakah sudah berjalan sesuai yang diharapkan atau belum. Skenario pengujian fungsional perangkat lunak dapat dilihat pada Tabel 4.3 Skenario Pengujian.

**Tabel 4.3 Skenario Pengujian**

No	Komponen Uji	Skenario Pengujian	Jenis Pengujian
1	Eksplorasi	Mengatur parameter algoritma dan menekan tombol eksplorasi	<i>Black-Box</i>
2	Eksplorasi	Mengatur parameter algoritma dan menekan tombol eksploitasi	<i>Black-Box</i>

Pengujian dilakukan dengan menguji setiap proses dengan kemungkinan kesalahan-kesalahan yang terjadi dan diamati respon dari perangkat lunak sebagai hasil pengujian.

#### 4.2.2.1. Hasil Pengujian

Pengujian dilakukan dengan menguji setiap proses untuk kemungkinan kesalahan yang terjadi. Kesalahan yang terjadi bisa saja kesalahan format inputan, inputan kosong atau nilai yang inputkan tidak sesuai dengan aturan atau batas yang telah ditentukan. Pada komponen uji eksplorasi hasil pengujian dapat dilihat pada Tabel 4.4 Hasil Pengujian Eksplorasi.

**Tabel 4.4 Hasil Pengujian Eksplorasi**

<b>Hasil Pengujian (Data Masukan BENAR)</b>			
<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate : 0.001	Sistem menampilkan permainan flappy bird	Sistem menampilkan permainan flappy bird	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
<b>Hasil Pengujian (Data Masukan SALAH)</b>			
<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate : - 0.001	Sistem menampilkan pesan 'Nilai Learning Rate Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Learning Rate Tidak Boleh Kurang Dari 0'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Kurang Dari 0'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : - 0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Lebih Dari 1'	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Lebih Dari 1'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 1.1			
Batch : 32			
Kondisi Henti : 20			

<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Batch Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Batch Tidak Boleh Kurang Dari 0'	[√] Diterima [ ] Ditolak
Discount Rate : 0.99			
Batch : -32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Kurang Dari 0'	[√] Diterima [ ] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : -1			
Learning Rate: 0.001	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Lebih Dari 1000'	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Lebih Dari 1000'	[√] Diterima [ ] Ditolak
Discount Rate: 0.99			
Batch: 32			
Kondisi Henti: 1001			
<b>Hasil Pengujian (Data Masukan kosong atau tidak sesuai format)</b>			
<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate :	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate :			
Batch :			
Kondisi Henti :			
Learning Rate :	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate :			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate : 0.99			
Batch :			
Kondisi Henti : 20			

<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti :			
Learning Rate : Aa%\$	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : Aa%\$			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : 0.99			
Batch : Aa%\$			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : Aa%\$			
Learning Rate : Aa%\$	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : Aa%\$			
Batch : Aa%\$			
Kondisi Henti : Aa%\$			

Sedangkan untuk komponen uji eksploitasi hasil pengujian dapat dilihat pada Tabel 4.5 Hasil Pengujian Eksploitasi.

Tabel 4.5 Hasil Pengujian Eksploitasi

<b>Hasil Pengujian (Data Masukan BENAR)</b>			
<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate : 0.001	Sistem menampilkan permainan flappy bird	Sistem menampilkan permainan flappy bird	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
<b>Hasil Pengujian (Data Masukan SALAH)</b>			
<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate : -0.001	Sistem menampilkan pesan 'Nilai Learning Rate Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Learning Rate Tidak Boleh Kurang Dari 0'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Kurang Dari 0'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : -0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Lebih Dari 1'	Sistem menampilkan pesan 'Nilai Discount Rate Tidak Boleh Lebih Dari 1'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 1.1			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Batch Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Batch Tidak Boleh Kurang Dari 0'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 0.99			
Batch : -32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Kurang Dari 0'	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Kurang Dari 0'	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : -1			



<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate: 0.001	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Lebih Dari 1000'	Sistem menampilkan pesan 'Nilai Kondisi Henti Tidak Boleh Lebih Dari 1000'	[√] Diterima [] Ditolak
Discount Rate: 0.99			
Batch: 32			
Kondisi Henti: 1001			
<b>Hasil Pengujian (Data Masukan kosong atau tidak sesuai format)</b>			
<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate :	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate :			
Batch :			
Kondisi Henti :			
Learning Rate :	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate :			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : 0.99			
Batch :			
Kondisi Henti : 20			
<b>Data Masukan</b>	<b>Hasil yang Diharapkan</b>	<b>Hasil Pengamatan</b>	<b>Kesimpulan</b>
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti :			

Data Masukan	Hasil yang Diharapkan	Hasil Pengamatan	Kesimpulan
Learning Rate : Aa%\$	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate : Aa%\$			
Batch : 32			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate : 0.99			
Batch : Aa%\$			
Kondisi Henti : 20			
Learning Rate : 0.001	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate : 0.99			
Batch : 32			
Kondisi Henti : Aa%\$			
Learning Rate : Aa%\$	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	Sistem menampilkan pesan 'Kolom Inputan Tidak Boleh Kosong dan Harus Berupa Angka'	[√] Diterima [ ] Ditolak
Discount Rate : Aa%\$			
Batch : Aa%\$			
Kondisi Henti : Aa%\$			

#### 4.2.2.2. Kesimpulan Hasil Pengujian Fungsional

Berdasarkan hasil pengujian yang telah dilakukan terhadap komponen-komponen uji, berkesimpulan bahwa pada proses penyaringan kesalahan input dalam bentuk pesan yang ditampilkan sistem telah sesuai dengan apa yang diharapkan dan secara fungsional sistem yang telah dibangun bekerja dengan baik dan sesuai dengan yang diharapkan.

#### 4.2.2. Pengujian Performansi

Pengujian performansi dilakukan untuk mengukur perform dari algoritma yang diterapkan dan melihat pengaruh dari setiap perubahan parameter algoritma untuk mencari kombinasi parameter yang optimal sehingga dapat diketahui rata-rata jumlah percobaan terkecil yang dibutuhkan pada proses eksplorasi dan skor terbesar yang dapat diraih *agent*. Parameter algoritma yang akan diujikan adalah *learning rate*, *discount rate*, ukuran *batch* dan kondisi henti. Sebagai catatan pada proses pengujian posisi pipa random dan tidak di atur sama antara satu percobaan dengan percobaan lainnya. Adapun skenario dari pengujian performansi dari algoritma yang digunakan dapat dilihat pada Tabel 4.3 Skenario Pengujian.

**Tabel 4.6 Skenario Pengujian**

No	Komponen Uji	Nilai Pengujian	Parameter yang Dilihat
1	Learning Rate	0,001; 0,01; 0,1	Jumlah percobaan pada proses eksplorasi dan Skor yang didapat
2	Discount Rate	0,1; 0,5; 0,99	Jumlah percobaan pada proses eksplorasi dan Skor yang didapat
3	Ukuran Batch	8 , 16 , 32	Jumlah percobaan pada proses eksplorasi dan Skor yang didapat
4	Kondisi Henti	10 , 15 , 20	Jumlah percobaan pada proses eksplorasi dan Skor yang didapat

Pengujian dilakukan dengan menguji setiap skenario pengujian dengan berbagai nilai uji. Setelah itu akan dilihat pengaruh perubahan nilai tersebut terhadap jumlah percobaan yang dibutuhkan *agent* untuk menyelesaikan proses eksplorasi dan skor yang didapatkan *agent* saat memainkan permainan

#### 4.2.2.1. Hasil Pengujian Learning Rate

*Learning rate* adalah parameter untuk menghitung koreksi bobot pada proses *backpropagation*. Pada pengujian ini akan menggunakan tiga nilai yaitu 0,001, 0,01 dan 0,1, sedangkan untuk parameter algoritma lainnya menggunakan *discount rate* 0,99, ukuran *batch* 32 dan kondisi henti di skor 20. Adapun hasil pengujian *learning rate* berupa pasangan nilai jumlah percobaan pada proses eksplorasi dan skor yang didapat saat memasuki proses eksploitasi dapat dilihat Tabel 4.7 Pengujian *Learning Rate*. Percobaan yang gagal memenuhi kondisi henti ditandai bintang(\*) pada isi sel table dengan nama kolom jumlah percobaan proses eksplorasi.

**Tabel 4.7 Pengujian *Learning Rate***

No	0,001		0,01		0,1	
	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat
1	323 percobaan	892	*7420 percobaan	11	*6279 percobaan	4
2	222 percobaan	1000	*7558 percobaan	22	*7005 percobaan	2
3	*4797 percobaan	5	*4336 percobaan	14	348 percobaan	351
4	*5417 percobaan	5	576 percobaan	724	*4843 percobaan	13
5	924 percobaan	974	*4851 percobaan	15	*6418 percobaan	8
<b>Rata-rata</b>	2336,6 percobaan	575,2	4948,2 percobaan	157,2	4978,6 percobaan	75,6

Berdasarkan hasil dari pengujian yang dilakukan sebanyak 5 kali, menunjukkan bahwa penggunaan *learning rate* 0,001 memiliki hasil paling baik dimana proses eksplorasi diselesaikan dengan rata-rata 2336,6 percobaan dan rata-rata skor yang didapat 575,2 meskipun 2 dari 5 percobaan gagal memenuhi kondisi henti. Hal menarik lainnya adalah ketika proses eksplorasi tidak berhasil memenuhi kondisi henti (ditandai \*) skor yang didapat *agent* akan jauh lebih kecil dibandingkan percobaan yang berhasil memenuhi kondisi henti. Selain itu, semakin besar *learning rate* jumlah percobaan yang dibutuhkan untuk menyelesaikan proses

eksplorasi cenderung semakin besar begitu pula skor yang didapat juga semakin kecil. Hal tersebut menunjukkan penggunaan *learning rate* yang sangat kecil memperbesar jaminan proses pembelajaran akan menemukan hasil yang optimal sedangkan *learning rate* yang terlalu besar akan memperbesar kemungkinan hasil pembelajaran tidak konvergen atau bahkan divergen dan memberikan hasil yang buruk [24].

#### 4.2.2.2. Hasil Pengujian Discount Rate

Komponen uji selanjutnya adalah *discount rate*. Pada pengujian ini akan menggunakan tiga nilai yaitu 0,1; 0,5 dan 0,99, sedangkan untuk parameter algoritma lainnya menggunakan *learning rate* 0,001, ukuran *batch* 32 dan kondisi henti di skor 20. Adapun hasil pengujian *discount rate* berupa pasangan nilai jumlah percobaan pada proses eksplorasi dan skor yang didapat saat memasuki proses eksploitasi dapat dilihat pada Tabel 4.8 Pengujian *Discount Rate*. Percobaan yang gagal memenuhi kondisi henti ditandai bintang(\*) pada isi sel table dengan nama kolom jumlah percobaan proses eksplorasi.

**Tabel 4.8 Pengujian *Discount Rate***

No	0,1		0,5		0,99	
	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat
1	*11203 percobaan	1	*4342 percobaan	4	323 percobaan	892
2	*8422 percobaan	0	*5717 percobaan	12	222 percobaan	1000
3	*9257 percobaan	2	*4372 percobaan	5	*4797 percobaan	5
4	*9510 percobaan	1	*4290 percobaan	5	*5417 percobaan	5
5	*11048 percobaan	1	*4088 percobaan	7	924 percobaan	974
<b>Rata-rata</b>	9888 percobaan	1	4561,8 percobaan	6,6	2336,6 percobaan	575,2

Berdasarkan hasil pengujian yang telah dilakukan terlihat bahwa penggunaan *discount rate* 0,99 memiliki hasil paling baik dimana 3 dari 5 percobaan berhasil memenuhi kondisi henti dengan rata-rata 2336,6 percobaan dan rata-rata skor yang didapat 575,2. Selain itu asumsi pada pengujian sebelumnya terjadi, yaitu percobaan yang gagal memenuhi kondisi henti akan memberikan skor yang lebih buruk dibandingkan percobaan yang berhasil memenuhi kondisi henti. Jika dilihat keseluruhan, penggunaan *discount rate* semakin kecil cenderung memberikan hasil yang semakin buruk bahkan untuk parameter *discount rate* 0,1 dan 0,5 tidak ada satupun percobaan yang berhasil memenuhi kondisi henti. Hal ini menunjukkan bahwa penggunaan nilai *discount rate* yang semakin mendekati 1 akan memberikan hasil semakin optimal dilihat dari skor yang didapatkan *agent* [25]. Semakin mendekati 0, maka kemampuan *agent* akan semakin menurun dan memeberikan hasil yang kurang optimal dan dapat memperlambat proses eksplorasi atau pembelajaran [25].

#### 4.2.2.3. Hasil Pengujian Ukuran *Batch*

Parameter uji selanjutnya adalah ukuran *batch*. Pada pengujian ini akan menggunakan tiga nilai berbeda, yaitu 8, 16 dan 32, sedangkan untuk parameter algoritma lainnya menggunakan *learning rate* 0,001, *discount rate* 0,99 dan kondisi henti di skor 20. Adapun hasil pengujian ukuran *batch* berupa pasangan nilai jumlah percobaan pada proses eksplorasi dan skor yang didapat saat memasuki proses eksploitasi dapat dilihat pada Tabel 4.9 Pengujian Ukuran *Batch*. Percobaan yang gagal memenuhi kondisi henti ditandai bintang(\*) pada isi sel table dengan nama kolom jumlah percobaan proses eksplorasi.

**Tabel 4.9 Pengujian Ukuran *Batch***

No	8		16		32	
	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat
1	*4187 percobaan	0	873 percobaan	83	323 percobaan	892
2	*5968 percobaan	6	1053 percobaan	937	222 percobaan	1000
3	*5244 percobaan	7	*6580 percobaan	6	*4797 percobaan	5
4	*7513 percobaan	3	*8478 percobaan	0	*5417 percobaan	5
5	*6378 percobaan	21	*7069 percobaan	13	924 percobaan	974
<b>Rata-rata</b>	5858 percobaan	7,4	4810,6 percobaan	207,8	2336,6 percobaan	575,2

Berdasarkan hasil pengujian yang telah dilakukan sebanyak 5 kali, terlihat bahwa penggunaan ukuran *batch* 32 memiliki hasil paling baik dimana 3 dari 5 percobaan berhasil memenuhi kondisi henti dengan rata-rata 2336,6 percobaan dengan rata-rata skor yang didapat 575,2 dan semakin kecil ukuran *batch* yang digunakan hasil yang didapat cenderung semakin buruk. Asumsi Hpengujian sebelumnya masih berlanjut, yaitu ketika proses eksplorasi tidak berhasil memenuhi kondisi henti (ditandai \*) skor yang didapat *agent* akan jauh lebih kecil dibandingkan percobaan yang berhasil memenuhi kondisi henti. Selain itu, ada hal menarik lainnya di percobaan 1 dengan ukuran *batch* 16 dimana *agent* berhasil memenuhi kondisi henti di percobaan 873 namun skor yang didapat cukup jauh dibandingkan dengan percobaan lain yang berhasil memenuhi kondisi henti, hasil tersebut menunjukkan ukuran *batch* semakin kecil maka akan menurunkan stabilitas proses pembelajaran atau eksplorasi yang adapat berakibat menurunnya kemampuan *agent* meskipun kondisi henti terpenuhi dan hal tersebut sesuai dengan penelitian J Lin menjelaskan bahwa penggunaan ukuran *batch* pada *reply memory* yang semakin besar maka akan meningkatkan *stabilitas* pada saat proses pembelajaran atau eksplorasi dan memberikan hasil yang optimum [26] dan penelitian Liu yang menjelaskan bahwa penggunaan ukuran *batch* yang terlalu kecil

akan memperlambat proses eksplorasi dan memperbesar kemungkinan hasil dari proses eksplorasi yang buruk dan *agent* cenderung melakukan *action* tertentu [27].

#### 4.2.2.4. Hasil Pengujian Kondisi Henti Proses Eksplorasi

Kondisi henti yang digunakan pada proses eksplorasi adalah skor yang didapat *agent*. Jika dalam 1000000 *frame agent* gagal memenuhi kondisi henti maka proses eksplorasi akan secara paksa dihentikan. Pada pengujian ini akan menggunakan tiga nilai berbeda, yaitu 10, 15 dan 20. Parameter algoritma lain yang digunakan antara lain *learning rate* 0,001, *discount rate* 0,99 dan ukuran *batch* 32. Hasil pengujian berupa pasangan nilai jumlah percobaan pada proses eksplorasi dan skor yang didapat saat memasuki proses eksploitasi dapat dilihat pada Tabel 4.10 Pengujian Kondisi Henti. Percobaan yang gagal memenuhi kondisi henti ditandai bintang(\*) pada isi sel table dengan nama kolom jumlah percobaan proses eksplorasi.

**Tabel 4.10 Pengujian Kondisi Henti**

No	10		15		20	
	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat	Jumlah Percobaan Proses Eksplorasi	Skor yang Didapat
1	76 percobaan	11	202 percobaan	1000	323 percobaan	892
2	1014 percobaan	2	701 percobaan	61	222 percobaan	1000
3	129 percobaan	1	165 percobaan	1000	*4797 percobaan	5
4	1212 percobaan	20	*4914 percobaan	5	*5417 percobaan	5
5	373 percobaan	21	*7408 percobaan	8	924 percobaan	974
<b>Rata-rata</b>	560,8 percobaan	11	2678 percobaan	414,8	2336,6 percobaan	575,2

Berdasarkan hasil pengujian yang telah dilakukan sebanyak 5 kali, jika dilihat dari jumlah percobaan untuk menyelesaikan eksplorasi maka penggunaan kondisi henti 10 menunjukkan hasil yang paling baik dengan rata-rata 560,8 kali percobaan dan selalu berhasil memenuhi kondisi henti. Namun, kondisi henti 10 justru



menunjukkan hasil yang paling buruk jika dilihat dari skor yang didapat yaitu rata-rata 11 dan paling kecil diantara 2 kondisi henti lainnya. Hal tersebut menunjukkan kondisi henti yang terlalu dini bisa mempercepat proses eksplorasi namun dapat berakibat pada skor yang didapat tidak terlalu baik dan menunjukkan bahwa pemilihan kondisi henti proses eksplorasi akan sangat mempengaruhi terhadap kemampuan *agent* dalam mendapatkan skor saat memainkan permainan [28]. Kondisi henti skor yang terlalu kecil atau terlalu dini akan memberikan hasil yang kurang baik, sedangkan kondisi henti dengan skor yang lebih besar akan memberikan hasil semakin baik, hal tersebut dilihat dari skor yang didapat *agent* saat memainkan permainan [28].

#### **4.2.2.5. Kesimpulan Hasil Pengujian Performansi**

Berdasarkan hasil pengujian yang telah dilakukan, setiap parameter memiliki pengaruh terhadap rata-rata jumlah percobaan yang dibutuhkan untuk menyelesaikan proses eksplorasi dan skor yang didapatkan *agent*. Penggunaan kombinasi parameter *learning rate* 0.001, *discount rate* 0.99, ukuran batch 32 dan kondisi henti di skor 20 menunjukkan hasil paling baik dimana *agent* menyelesaikan proses eksplorasi dengan rata-rata 2336,6 percobaan dan mendapat skor rata-rata 575,2 bahkan 1 dari 5 percobaan berhasil menyentuh skor 1000. Namun, proses eskplorasi tidak selalu berjalan baik dimana 2 dari 5 percobaan gagal memenuhi kondisi henti dan mendapat skor yang buruk saat memasuki proses eksploitasi. Selain itu, banyaknya percobaan pada proses eksplorasi cenderung tidak berbanding lurus dengan kemampuan *agent* dalam mendapatkan skor.

