IMPLEMENTASI ALGORITMA SMA* PADA GAME TPS MONSTER NEST BERBASIS MOBILE

SIDANG SKRIPSI

FEBRIAN DONI

Penguji 1: Eko Budi Setiawan, S.Kom.

Penguji 2: Nelly Indriani W, S.Si., M.T.

Penguji 3: Ednawati Rainarli, S.Si., M.Si.

Latar Belakang

Algoritma A* menghabiskan banyak resources.

SMA* (Simplified Memory-Bounded A*) mengatasi masalah storage problem

Rumusan Masalah

Bagaimana cara mengatasi konsumsi memori pada A* yang menghabiskan banyak resources

Tujuan

- Menggunakan algoritma SMA*
- Mengimplementasikan SMA* pada pathfinding game TPS

Batasan Masalah

- Memori = jumlah CLOSED list dan OPEN list.
- Memiliki 3 level
- Berbasis mobile
- Implementasi SMA* pada NPC
- Bergenre TPS
- Single player

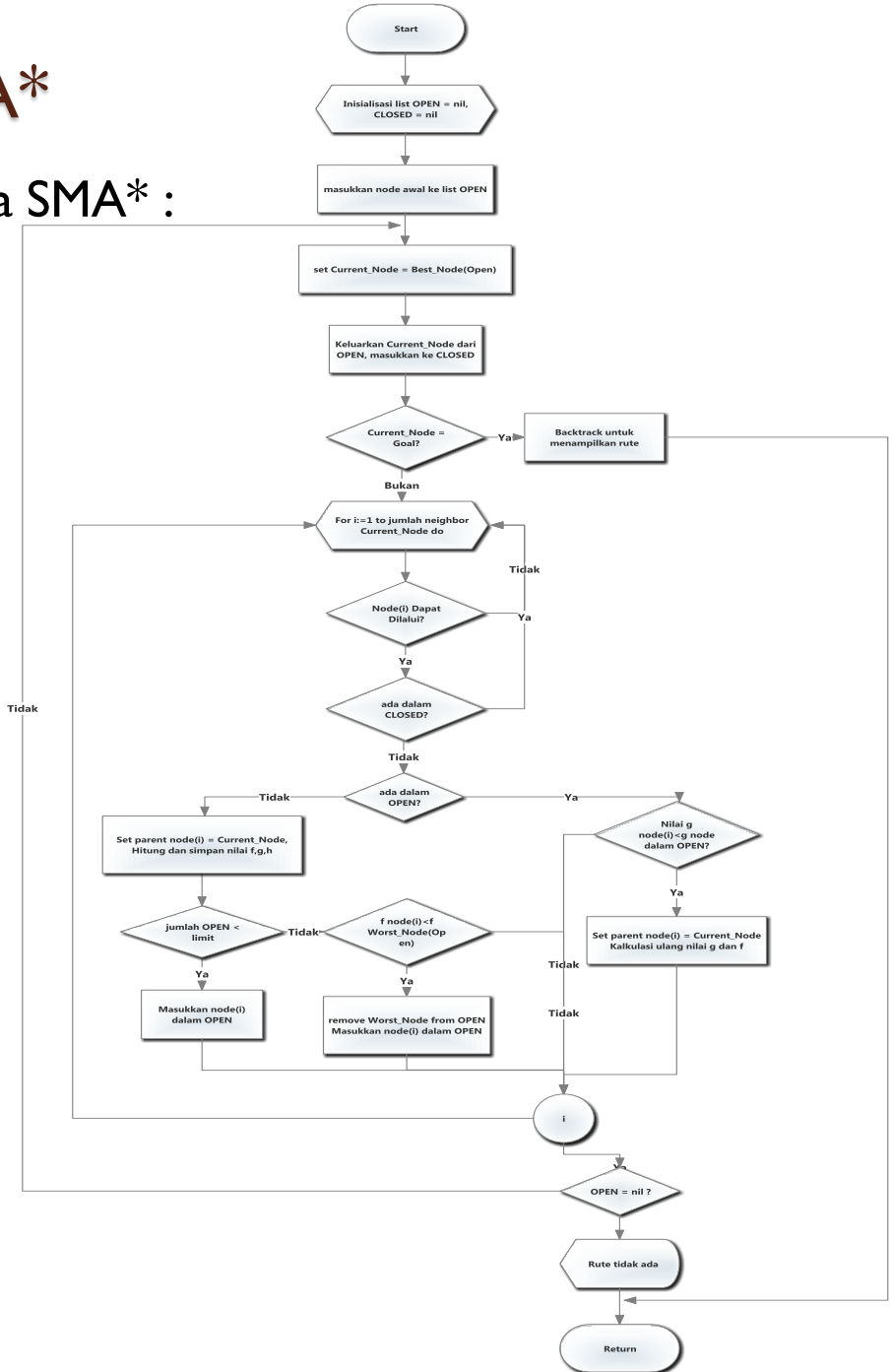
Algoritma SMA* (Simplified Memory Bounded A Star)

- SMA* memiliki beberapa sifat sebagai berikut :
  - F-cost digunakan untuk mendapatkan node dengan nilai terkecil diantara semua node yang berada di open list.
  - SMA* selalu memperluas node yang memiliki nilai terbaik pada keturunannya yang belum diperiksa.
  - SMA* dijamin mendapatkan solusi yang optimal, jika disediakan MAX setidaknya sama besar dengan jumlah node pada jalur solusi optimal.
  - SMA* berperilaku identik dengan A* bila MAX lebih besar dari jumlah node yang dihasilkan oleh A*.

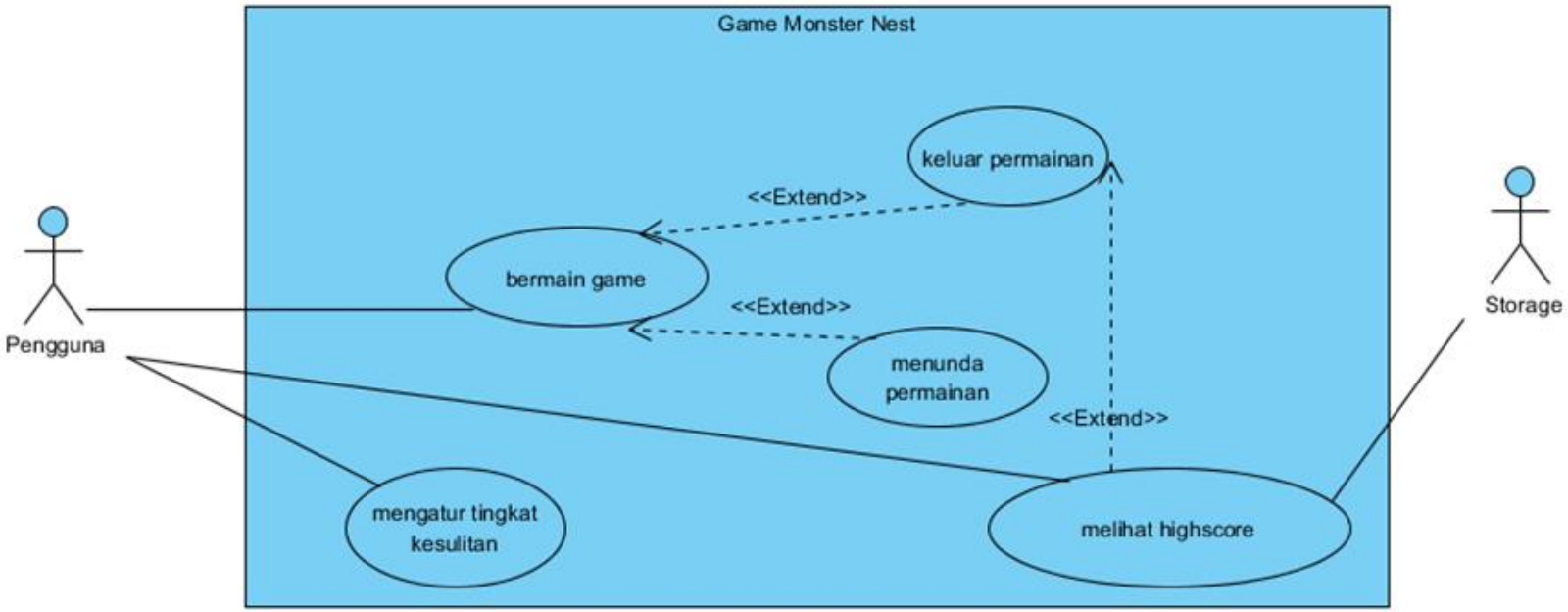
.

Algoritma SMA*

Cara kerja algoritma SMA* :

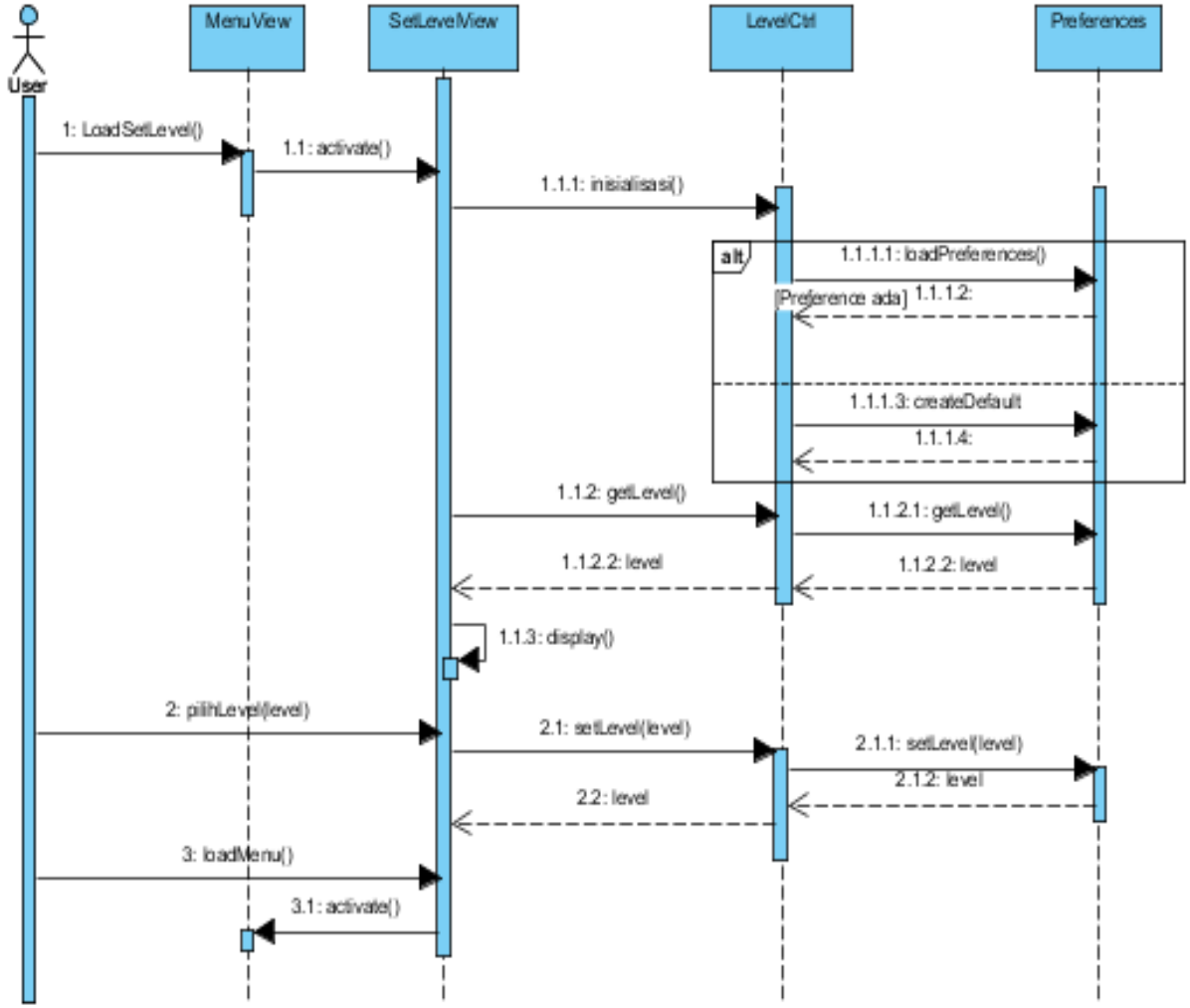


Use Case Diagram



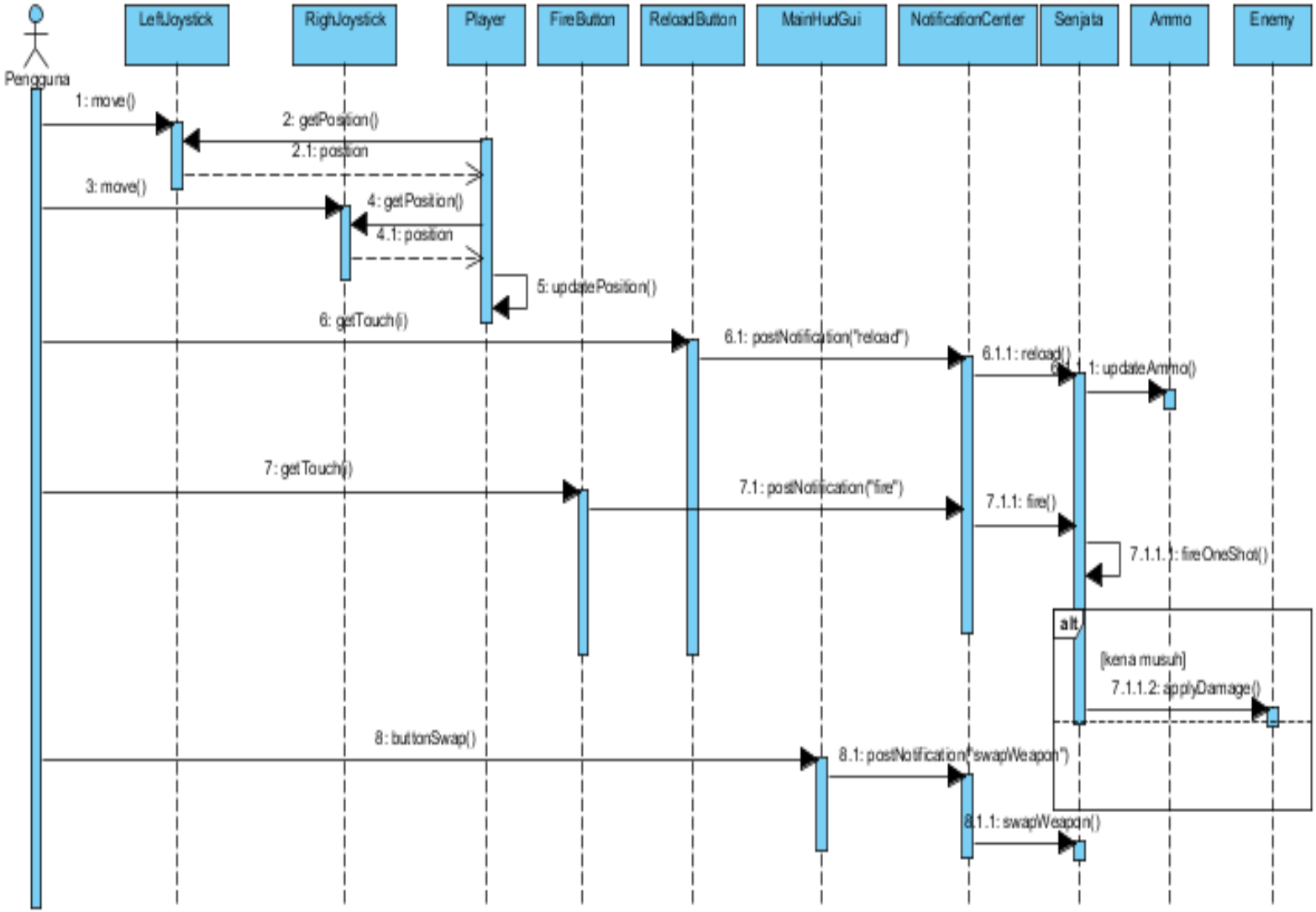
Sequencial Diagram (2)

- Sequencial Diagram Mengatur Tingkat Kesulitan



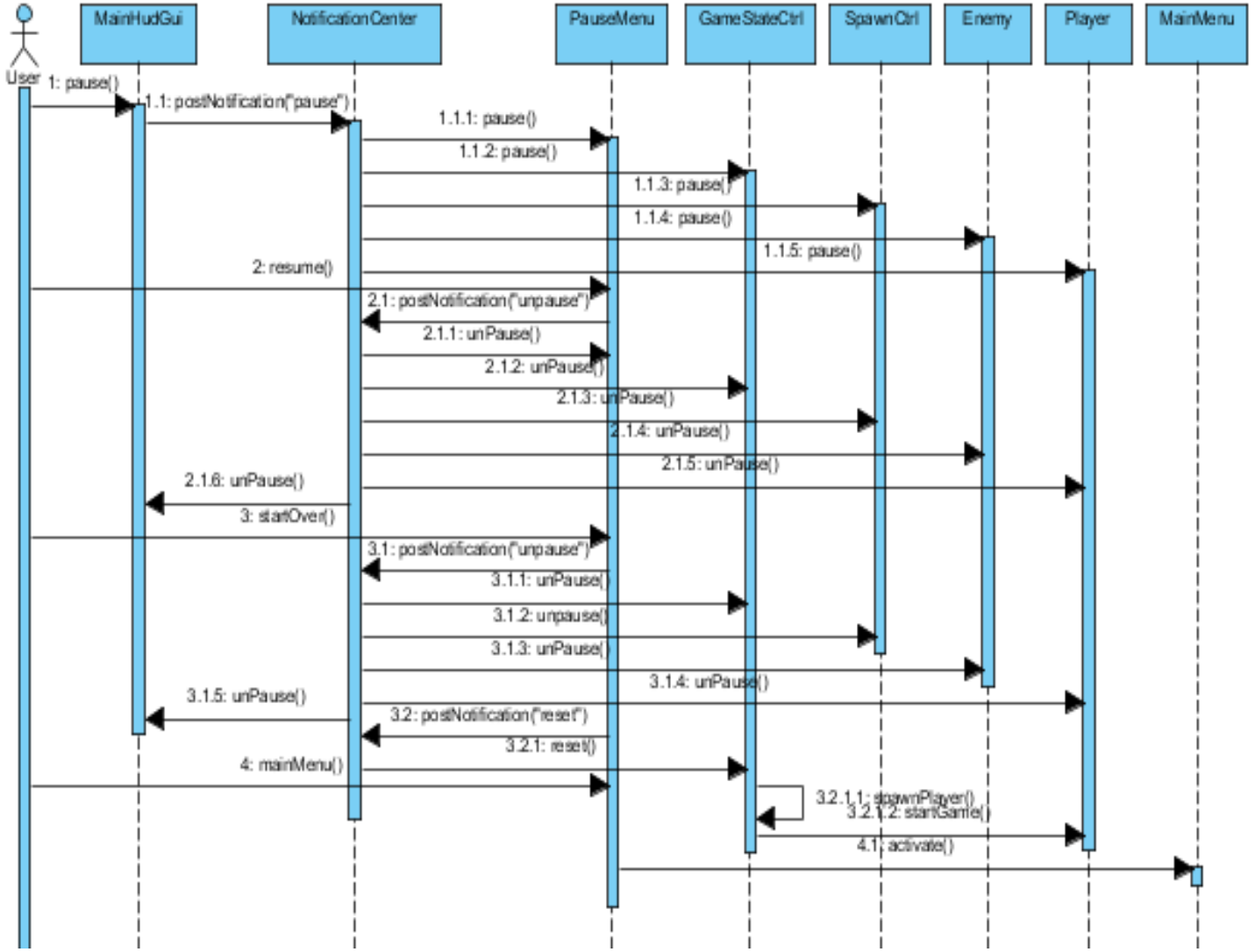
Sequencial Diagram (3)

- Sequencial Diagram Bermain Game



Sequencial Diagram (4)

- Sequencial Diagram Menunda Permainan



Implementasi Antarmuka





Implementasi antarmuka



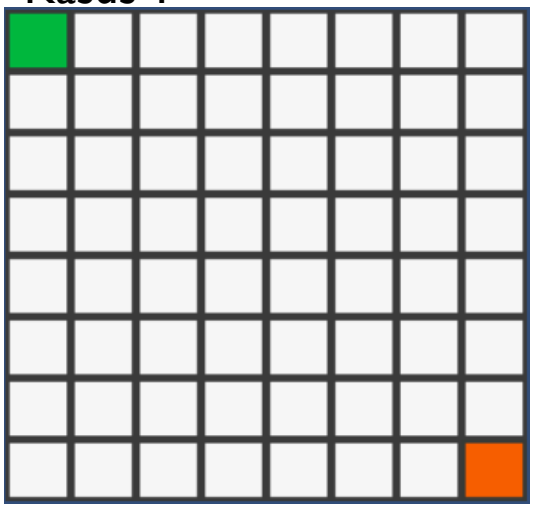


Pengujian program

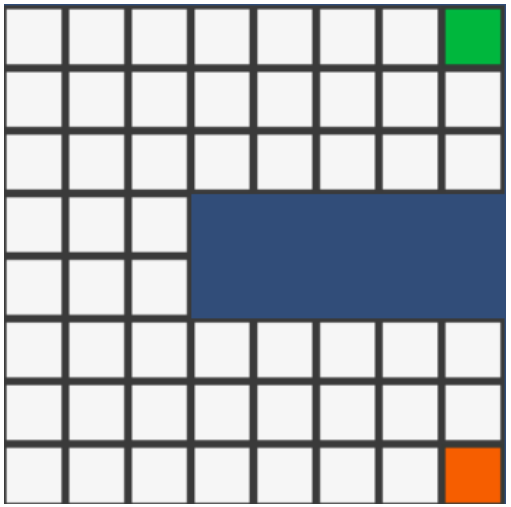
No	Komponen yang di uji	Skenario Pengujian	Jenis Pengujian
1	Menu Utama	Klik Menu *Play game*	*Black Box*
		Klik Menu set level	*Black Box*
		Klik Menu *highscore*	*Black Box*
2	Set level	Klik tombol *easy*	*Black Box*
		Klik tombol *medium*	*Black Box*
		Klik tombol *hard*	*Black Box*
3	Highscore	Klik Menu *highscore*	*Black Box*
		Mengisi nama pemain	*Black Box*
		Klik tombol *back*	*Black Box*
4	Layar permainan	Klik tombol pause	*Black Box*
		Klik tombol ganti senjata	*Black Box*
		Klik tombol reload	*Black Box*
		Klik tombol tembak	*Black Box*
		Menggerakkan joystick kiri	*Black Box*
		Menggerakkan joystick kanan	*Black Box*
5	Menu *pause*	Klik tombol *resume*	*Black Box*
		Klik tombol *start over*	*Black Box*
		Klik tombol *main menu*	*Black Box*
6	Bermain	Menyentuh ammo pack	*Black Box*
		Menyentuh health pack	*Black Box*
		Menembak musuh	*Black Box*
		Berpindah ronde	*Black Box*
		Player mati	*Black Box*
7	Algoritma SMA* (Simplified Memory Bounded A star)	Pencarian jalan	*White Box*
		Pengecekan node terdekat	*White Box*

Pengujian Algoritma

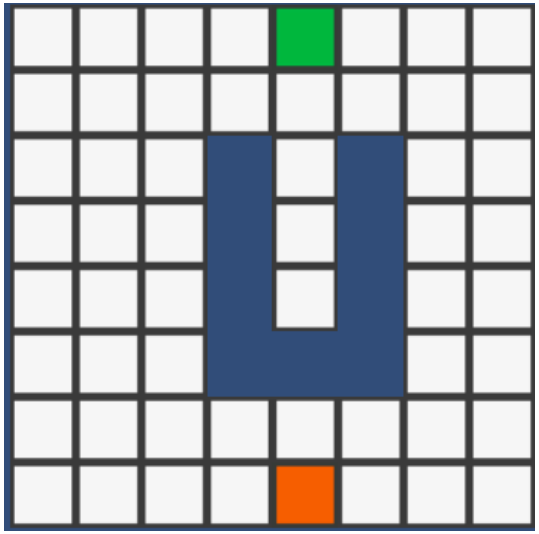
Kasus 1



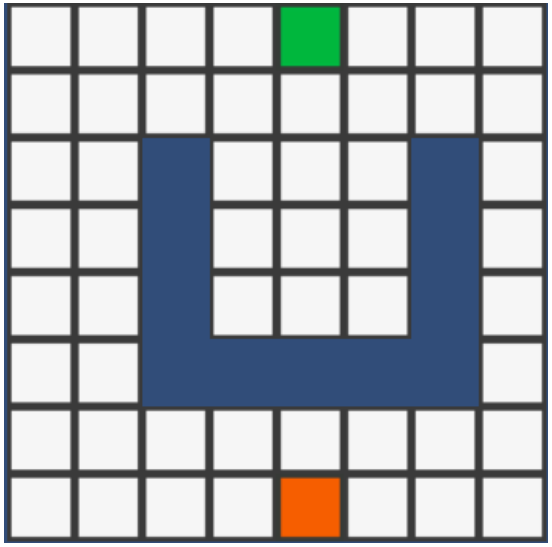
Kasus 2



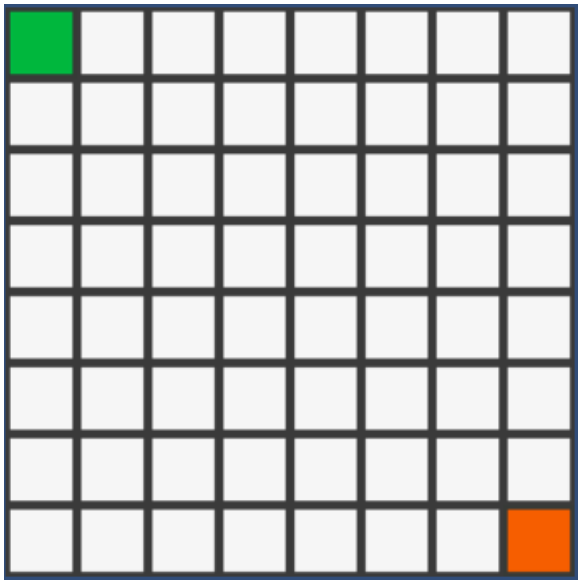
Kasus 3



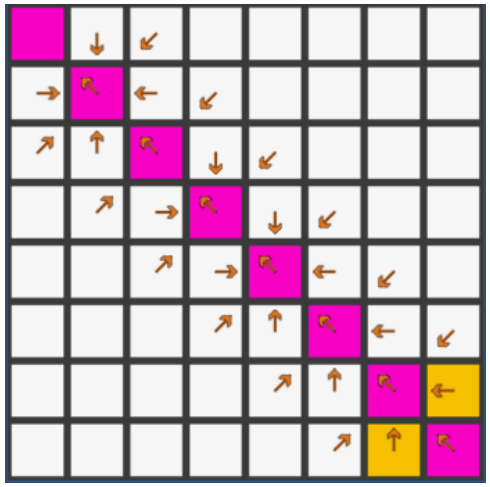
Kasus 4



Kasus 1

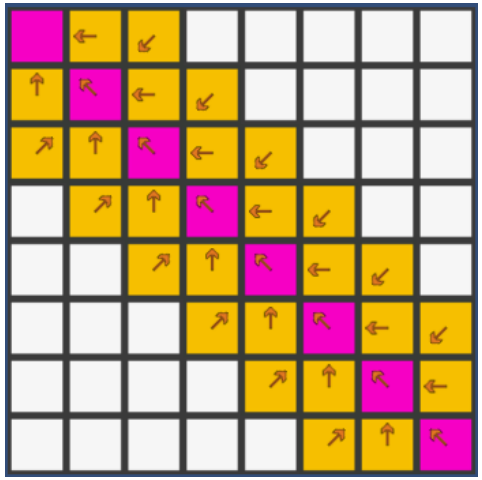


SMA* limit 3



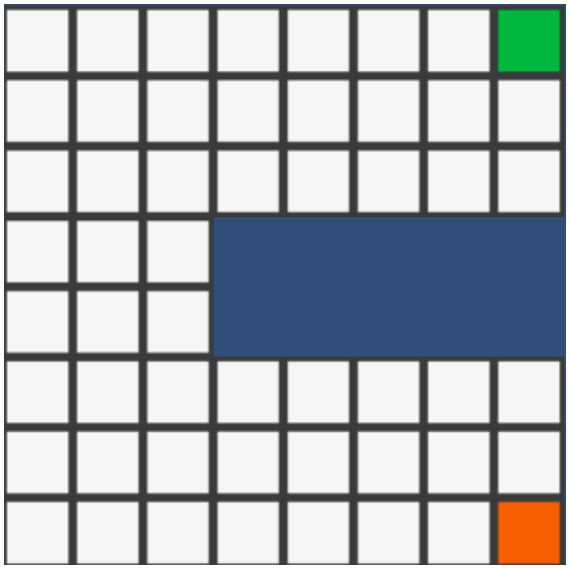
Open = 2
Closed = 8

A*

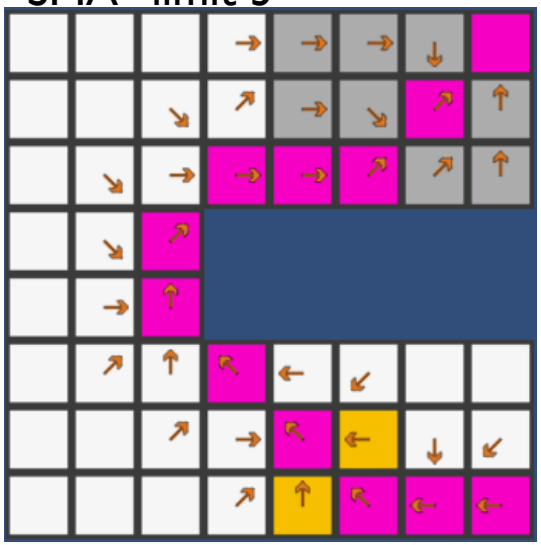


Open = 26
Closed = 8

Kasus 2

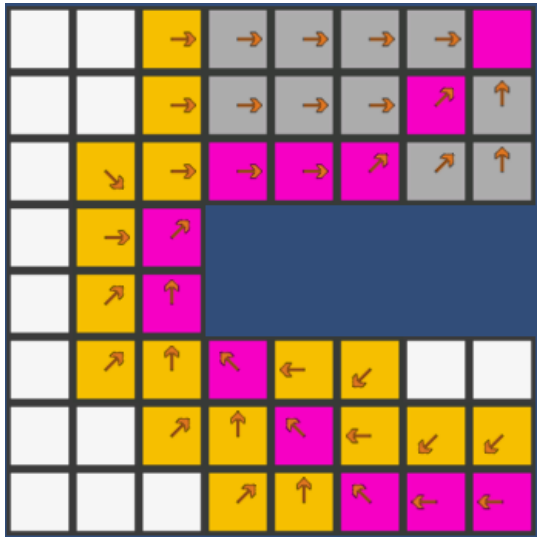


SMA* limit 3



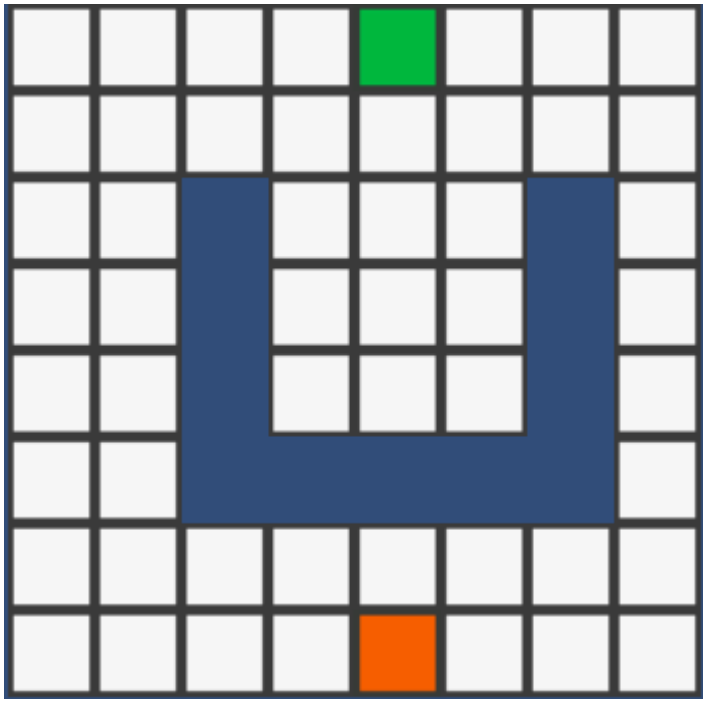
Open = 2
Closed = 20

A*

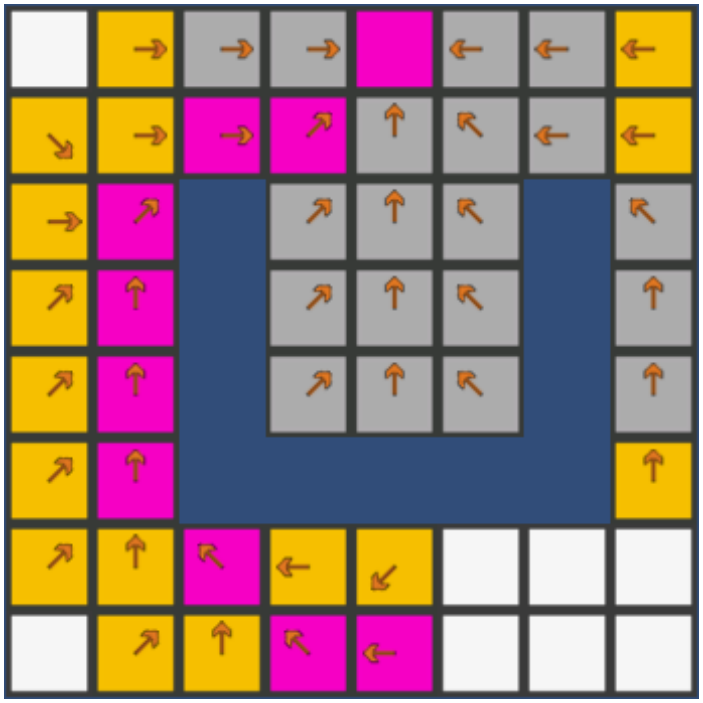


Open = 17
Closed = 22

Kasus 4



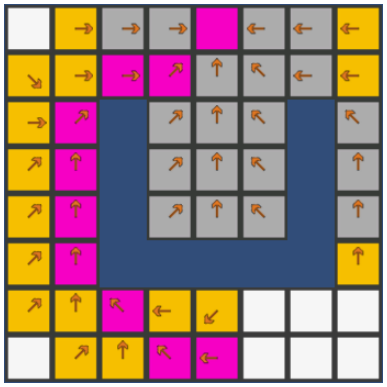
A*



Open = 16
Closed = 29

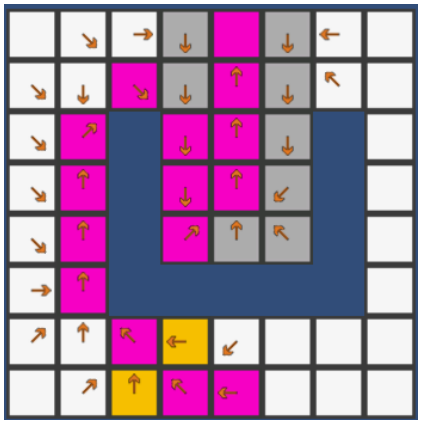
Kasus 4(lanjutan)

A*



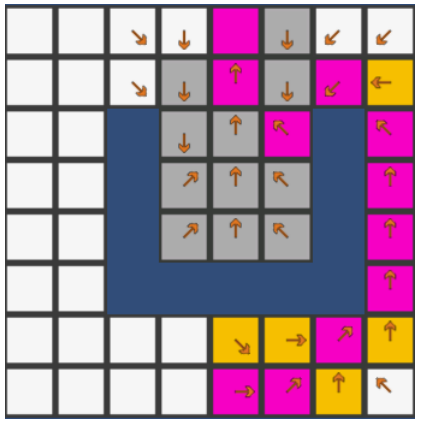
Open = 16
Closed = 29

SMA* limit 3



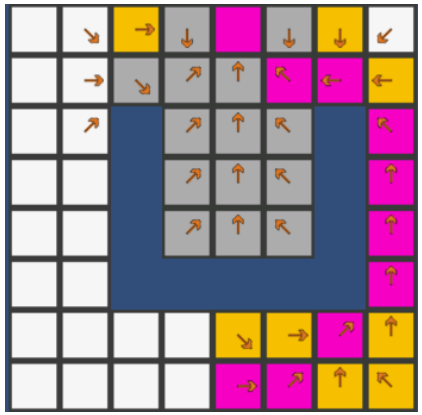
Open = 2
Closed = 23

SMA* limit 6



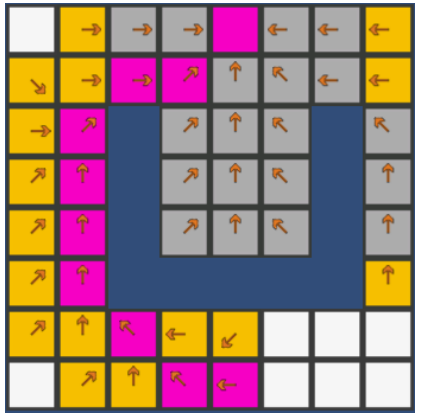
Open = 5
Closed = 22

SMA* limit 9



Open = 8
Closed = 24

SMA* limit 17



Open = 16
Closed = 29

Kesimpulan

- Algoritma SMA* (Simplified Memory Bounded A *) dapat diimplementasikan pada game third person shooter monster nest.
- Algoritma SMA* menggunakan resource atau memori yang lebih sedikit dibandingkan dengan algoritma A*.

Terima Kasih