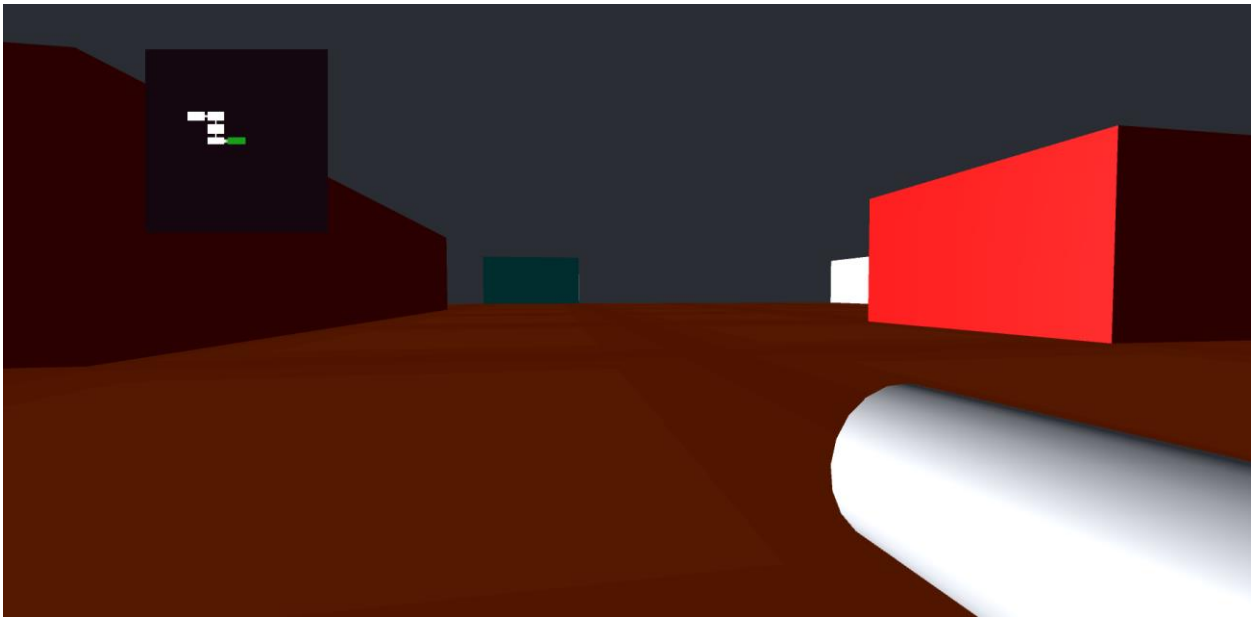


Just van der Linde

Klas - GDV2

Datum – 17-06-2018

Game – AI and Procedural Generation

Document – Technical document

Table of contents

The Game

Introduction

This project is divided into three sections: The first one being the observation of an AI played in a game, the second one building an AI system with two AI's interacting with each other, and the third one the creation of a randomly generated dungeon in which the AI's take place.

Everything was working out great. The AI was doing its thing and the dungeons were created, only one more thing left to do now; putting the AI into the dungeons.
All I have to do is create NavMesh at runtime, that couldn't be so hard right?
Turned out my 'at runtime generated NavMesh' had a will of it's own, creating heaps of NavMesh floors everywhere, causing my Unity to crash over and over again.
I failed to fix this before the end of the deadline, so I put it on hold for a while.

Coming back to the project I kind of grew out of the idea that I previously had, and really wanted to make a roguelike game. So, I decided to let the other project be was it was and start from the beginning.

Gameplay

As mentioned above, I actually have two games to present, so I'll explain the first one first:
This game was intended to be a 3D binding of Isaac, allowing you to travel from randomly generated room to room, killing the AI's present.
The project contains two scenes: Dungeon and AI. The AI scene demonstrates the AI since I couldn't get it to work in the dungeon. You can press space to fire projectiles at the target.
The second one is a simple roguelike in which you have to survive for as long as you can, making sure you eat enough food and avoiding the enemy.

Improvement

There is a lot of room for improvement on this subject, mainly because it's such a large and interesting subject. There are quite some flaws on the first project, which I'm willing to fix in the summer.
The second game is meant for me to build further on, expanding features and just being a base for me to start off from.

Task 1: Study AI

Play a game in which AI plays a prominent role, analyse it by doing the following:

- Describe the AI in the game
- Describe the different states of the AI, and what triggers it
- Write down what you like about the AI
- Write down what you don't like about the AI

For this assignment I have chosen to analyze the game Metal Gear Solid V: The Phantom Pain.
I bought this game fairly recently and given the AI has had an important role, I think this is a good game to investigate!

The AI that I am going to investigate are your human enemies.
I've chosen the human enemies because I think they're somewhat more interesting to study compared to animals / friendly AI.
The purpose of the AI in this game is to catch you and kill you.
The game mainly focuses on stealth, so it is important that you are not seen.
I have found a few different stages the AI can be in:

Calm
The enemy is calm and walks around quietly.
The enemy can also walk a predefined path to patrol.
If the enemy hears something in this condition, thinking that he sees something, he becomes suspicious. This can be suppressed by buying other armor that camouflages you better and dampens your sound.

Curious
At this moment there is a white curved line on the screen that indicates which enemy landed in named state. If the enemy is in this state, it is not directly bad news. The AI will even keep looking in the area where he thinks he heard or saw something

Suspicious
This is a bit like the state above, but then the AI heard something more from you, provision he came to the place where he thought he saw something. The enemy seems to be more on guard.

Reflex mode
This is the last condition before the enemy starts attacking you.
The enemy has spotted and is shook when he saw you.
Time is moving in slow motion, which gives you a last chance to neutralize the enemy.

Attack mode
The AI has seen you and is now trying to kill you. Besides that, he also warns others via radio. If the alarm is sound, your mission score will decrease and more enemies will come to the location.

Outside these states there are also certain elements that influence it, as indicated earlier: The kind of armor you wear, the sound you make, but also the behavior of fellow enemies around him.

I like this AI because it's simple and feels realistic. Keeping the states in consideration allows for you to anticipate on the actions made by the enemy. When the enemies become aware of you, you emerge in a panicked state of mind knowing you screwed up, continuing trying to escape the area.

The parts I don't like about this AI include the predefined paths. I know that it's also used in real life situations, but watching the guards walk their paths like simpletons, walking right into your trap sometimes feels to easy. I think it would be better if they had more then one path, or maybe strafe a little off their paths sometimes.

Task 2: Creating the AI

As mentioned in the introduction, I used the Panda Behavior Tree to set up my AI.

The AI has a few different states:

Wander
The AI walks around the walkable area randomly.

Look Around
The AI stays stationary and looks in different directions.

Patrol
The AI walks around a predefined path, patrolling the area in search of another player or enemy

Attack
When another AI has been spotted, the AI targets the player by getting into the correct rotation.
When it has reached its correct rotation, it starts firing.

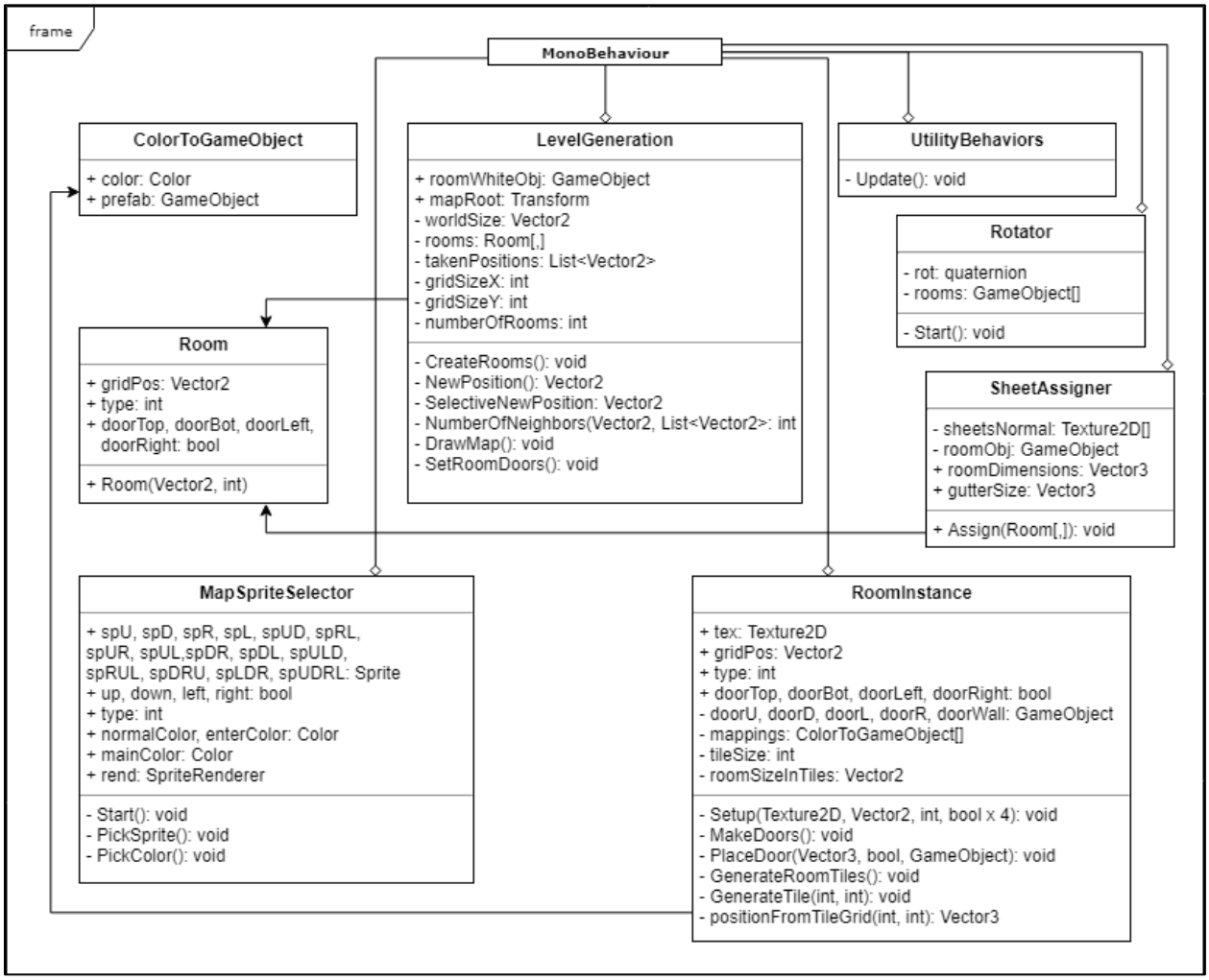
Flee
When an AI reaches a certain point of health, in my case less then 40, it gives up fighting and searches for a place to take cover.

Dead
When the AI has no more health left, it dies (explodes).

Which actions are triggered can be viewed live while playing, which makes for easy troubleshooting and ideating.

The AI makes use of a NavMesh agent to move around in the level.

Task 3: Creating the dungeon

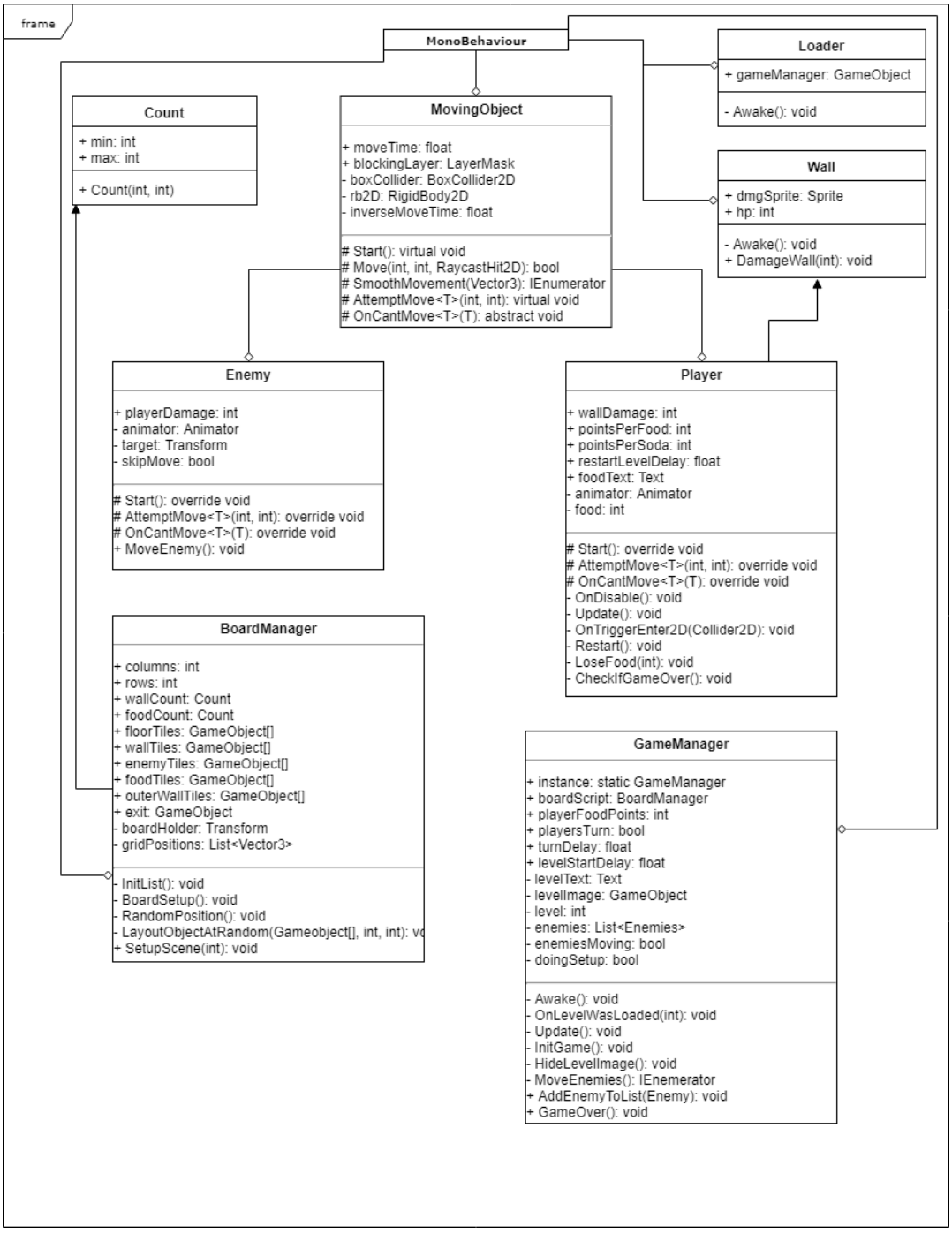


I had many ideas for this part but the one that stood out most was a 3D version of binding of Isaac.

The rooms are made in a two-dimensional array. The rooms are made randomly next to each other, doors are then created between the rooms. The interior of the room is made by reading the colors of the pixels of predefined sprites, which can be walls, floors, obstacles or enemies.

By doing this, a two-dimensional map of the rooms is created, from which the 3D area is created.

Task 3.5: Creating the dungeon (again)



Explanation on next page

The Enemy AI and player both inherit from MovingObject, which takes care of the turns and movement. The AI in this game is not as advanced as in the previous one, but that's an objective for in the summer. The BoardManager creates a few arrays of different tiles, which are placed randomly. The GameManager takes care of the rest of the flow of the game.