

UNIVERSIDAD DE SANTIAGO DE COMPOSTELA
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA



Inteligencia Artificial aplicada a Videojuegos Top-Down en tiempo real

TRABAJO DE FIN DE GRADO

Autor:

Rubén Osorio López

Directores:

Manuel Mucientes Molina

Pablo Rodríguez Mier



D. Manuel Mucientes Molina, Profesor del Departamento de Electrónica y Computación de la Universidad de Santiago de Compostela, y **D. Pablo Rodríguez Mier**, Profesor del Departamento de Electrónica y Computación de la Universidad de Santiago de Compostela,

INFORMAN:

Que la presente memoria, titulada *Inteligencia Artificial aplicada a Videojuegos Top-Down en tiempo real*, presentada por **D. Rubén Osorio López** para superar los créditos correspondientes al Trabajo de Fin de Grado de la titulación de Grado en Ingeniería Informática, se ha realizado bajo nuestra dirección en el Departamento de Electrónica y Computación de la Universidad de Santiago de Compostela.

Y para que así conste a los efectos oportunos, expiden el presente informe en Santiago de Compostela a (Data):

El director,

El codirector,

El alumno,

Manuel Mucientes Molina Pablo Rodríguez Mier Rubén Osorio López

Agradecimientos

A mi director Manuel y a mi codirector Pablo por su inestimable e imprescindible ayuda durante el periodo de desarrollo de este trabajo.

A mis compañeros de clase que me han sabido animar cuando más lo necesitaba.

A mi familia por apoyarme durante todo este tiempo.

Resumen

Completar
el resu-
men una
ver ter-
minado
el docu-
mento

Índice general

1. Introducción	1
1.1. Objetivos	2
1.2. Organización del documento	3
2. Análisis de requisitos	5
3. Gestión del proyecto	7
3.1. Análisis de riesgos	7
3.2. Metodología de desarrollo	7
3.3. Gestión de la configuración	7
3.4. Planificación temporal	8
3.5. Estimación de costes	8
4. Arquitectura y herramientas	9
4.1. Arquitectura del sistema	9
4.2. Herramientas utilizadas	9
5. Diseño e implementación	11
6. Validación y pruebas	13
7. Valoraciones finales	15
8. Ejemplos	19
8.1. Un exemplo de sección	19
8.1.1. Un exemplo de subsección	19
8.1.2. Otro exemplo de subsección	19
8.2. Exemplos de figuras e cadros	20
8.3. Exemplos de referencias á bibliografía	20
8.4. Exemplos de enumeracións	20
9. Conclusiones y posibles ampliaciones	23
A. Manuais técnicos	25

B. Manuais de usuario	27
C. Licencia	29
Bibliografía	31

Índice de figuras

8.1. Esta é a figura de tal e cal.	20
--	----

Índice de cuadros

8.1. Esta é a táboa de tal e cal.	20
---	----

Capítulo 1

Introducción

Los juegos de lógica o estrategia han sido utilizados como *benchmark* en el campo de la inteligencia artificial desde sus inicios. Pese a que principalmente se han utilizado juegos de mesa o de cartas dadas las facilidades a la hora de representar su modelo, con el avance de la tecnología y la llegada de los videojuegos en ordenador se consideró la posibilidad de que los mismos podrían ser una nueva herramienta con la que experimentar y explorar diferentes técnicas e implementaciones de IA. En la actualidad prácticamente todos los videojuegos que incorporan elementos que simulan comportamiento humano lo hacen con lógica determinista y patrones definidos, sin incorporar ningún tipo de simulación de pensamiento complejo. Algunas de las razones causantes de este fenómeno son:

- La diferencia en la dificultad de implementación entre patrones simples y un agente complejo.
- El hecho de que un agente puede llegar a ser demasiado efectivo a la hora de competir contra un humano.
- Limitaciones en la capacidad de computación restante cuando el sistema está ejecutando un videojuego ya que suelen ser aplicaciones significativamente pesadas y complejas.

Sin embargo, estos problemas son solventables y debemos tener en cuenta los beneficios que se pueden obtener al trabajar con videojuegos reales. En la actualidad, somos capaces de simular grandes mundos con millones de entidades y jugadores reales. Estos entornos comúnmente utilizan simulaciones físicas realistas y trabajan con datos numéricos continuos en lugar de acciones discretas, es decir, una entidad que simule estar viva en dicho entorno se podría mover cara una posición concreta y realizar una determinada acción durante un periodo de tiempo sin necesidad de que ninguno de estos conceptos esté atado a una representación discreta. Nos encontramos el caso contrario con los juegos de mesa o

cartas tradicionales ya que en estos solo existen acciones discretas y simbólicas, no se trata con ningún tipo de estado continuo.

Por estas razones se podría considerar que un videojuego está mucho mas cerca de ser capaz de modelar el mundo real que un juego de mesa, lo que lo hace un banco de trabajo idóneo para investigar y experimentar con Inteligencia Artificial. Técnicas como redes neuronales, computación evolutiva o lógica difusa pueden funcionar en situaciones en tiempo real, continuas y complejas que presentan tanto nuestra realidad como los videojuegos. Por esto se pretende investigar y experimentar sobre cuales de estas técnicas pueden ser de utilidad en este tipo de situaciones para lo cual se implementará un agente que compita con un jugador real en un pequeño videojuego en el que ambos se moverán libremente en un espacio de posiciones continuo y realizarán una serie de acciones ofensivas y defensivas con el objetivo de derrotar a su oponente.

Con este fin, se pretende realizar una implementación de un juego que permita al agente tener acceso a los estados en los que se encuentran el y su competidor así como las acciones que puede realizar en un determinado momento. Todo esto con la finalidad de averiguar cuales de las posibles técnicas de IA pueden ser usadas para lograr un competidor no solo apto pero también justo. Además, esto nos permitirá realizar un entrenamiento en el que el propio agente podrá competir con él mismo para aprender los fundamentos del juego además de competir contra jugadores humanos. Esto nos brinda la posibilidad de determinar como reacciona ante estos y si es capaz de adaptarse a los posiblemente distintos estilos de juego contra los que se tendrá que enfrentar.

1.1. Objetivos

El objetivo global será implementar un agente que sea capaz de competir contra un jugador simulando un comportamiento humano para lo cual se realizará una pequeña demostración de un videojuego y la implementación del agente con las técnicas que resulten ser más eficaces a la hora de lograr el mencionado objetivo.

Los sub-objetivos que tendremos que abarcar son los siguientes:

1. **Implementar la demostración del videojuego:** Necesitaremos una plataforma que permita tanto a un jugador humano como a nuestro agente interactuar con el entorno del videojuego siguiendo ambos las reglas que este mismo define.
2. **Crear la librería que implemente nuestro agente con las técnicas escogidas:** Se requerirá realizar una implementación del agente utilizando

las técnicas de inteligencia artificial que hayamos escogido para simular un comportamiento aparentemente humano.


3. **Realizar el entrenamiento del agente:** Una vez realizada la implementación del agente necesitaremos ejecutar un proceso de entrenamiento para que este adquiera la información necesaria para comportarse adecuadamente en el entorno competitivo que el juego presenta.
4. **Obtener datos sobre el agente compitiendo contra jugadores reales:** Cuando el agente pueda presentar un desafío no insignificante para un jugador humano se deberán observar las reacciones de los jugadores al enfrentarse al mismo, así como obtener información sobre la dificultad que presenta o las características que ayudan a categorizarlo como un jugador virtual y no real.
5. **Analizar los resultados obtenidos** Haciendo una recopilación de la información obtenida durante las etapas de diseño, implementación y obtención de datos deberemos realizar un análisis que resuma lo que ha logrado el agente centrándonos en aspectos como el desafío que es capaz de presentar, si es capaz de simular a un humano y la posibilidad de usar implementaciones similares en videojuegos en un futuro.

1.2. Organización del documento

La finalidad de este documento es presentar como se han resuelto los objetivos definidos para el proyecto. Para ello se explicarán las diferentes partes que forman el producto final así como las tareas que han sido realizadas a lo largo del proyecto y que han dado lugar al mismo tal y como se presenta.

Capítulo 2

Análisis de requisitos



Completar
análisis
de requi-
sitos

Capítulo 3

Gestión del proyecto

Dentro de la ingeniería de software, una de las partes esenciales para la realización de proyectos considerados exitosos es la gestión de proyectos. La realización de una buena gestión no se puede considerar, ni mucho menos, una garantía de que el proyecto a gestionar vaya a resultar un éxito. Sin embargo, si elegimos ignorar o realizar una mala gestión de nuestro proyecto sí podremos considerar que nos encontraremos en una situación proclive para un proyecto fallido. Durante toda la extensión del ciclo de vida de nuestro proyecto se utilizará la gestión de proyectos como un método que nos ayudará a lograr la obtención de un producto final ajustado a todas las necesidades y restricciones presentes, sea cual sea la índole de las mismas (tiempo, costes, requisitos, etc.).

Dedicaremos este capítulo a definir y explicar la gestión de nuestro proyecto en todas sus partes. Se determinará y explicará el análisis de riesgos, la metodología de desarrollo empleada, la gestión de configuración, la planificación temporal y la estimación de costes.

3.1. Análisis de riesgos

3.2. Metodología de desarrollo

3.3. Gestión de la configuración

Completar
los ries-
gos
necesarios

Completar
la meto-
dología

Completar
la ges-
tión de
la confi-
guración

3.4. Planificación temporal

Completar
la plani-
ficación
temporal

3.5. Estimación de costes

Completar
la esti-
mación
de costes

Capítulo 4

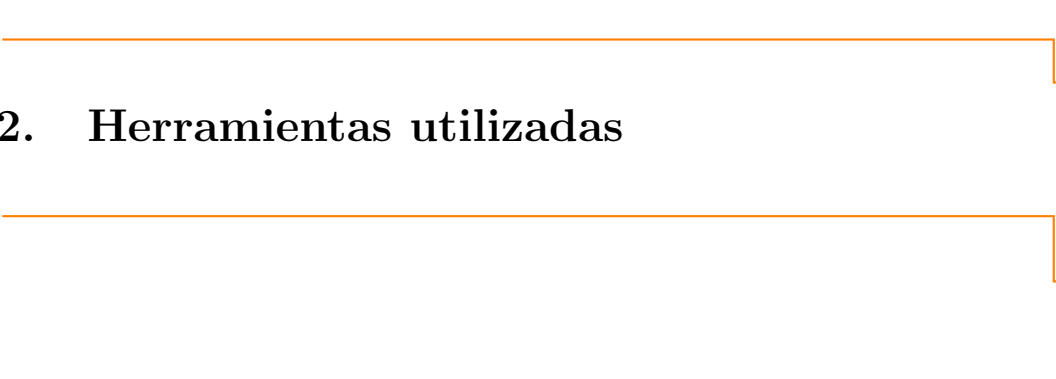
Arquitectura y herramientas

El IEEE define arquitectura como los conceptos fundamentales o propiedades de un sistema en su entorno que se encarnan en sus elementos, relaciones y los principios de su diseño y evolución [1]. Como referencia el propio IEEE, es el estándar ISO 12207 el que identifica un diseño arquitectónico y un diseño detallado y desgranado en lo que a un sistema se refiere. A partir de esto se obtiene que el diseño de la arquitectura de un sistema describe la estructura y la organización del mismo, es decir, se centra en los subsistemas o componentes que forman el sistema completo y las relaciones entre los mismos desde un punto de vista de alto nivel.

Contando ahora con una idea definida de cuales son los objetivos y requisitos de nuestro proyecto, nos centraremos en este capítulo en el diseño a alto nivel y en la arquitectura a la cual nos referimos en el párrafo anterior. Para ello utilizaremos representaciones gráficas a alto nivel del sistema y detallaremos los elementos arquitectónicos del producto y que tecnologías han sido usadas en los mismos así como las herramientas de las que hemos hecho uso para llevar a cabo el proyecto en su totalidad.

4.1. Arquitectura del sistema

4.2. Herramientas utilizadas




Completar
Arquitectu-
ra del
sistema

Completar
Herra-
mientas
utiliza-
das

Capítulo 5

Diseño e implementación



Completar
Diseño
e imple-
menta-
ción

Capítulo 6

Validación y pruebas


Completar
Validación y
pruebas

Capítulo 7

Valoraciones finales



Completar
Valora-
ciones
finales



Quitar
la sec-
ción de
ejemplos

Capítulo 8

Exemplos

8.1. Un exemplo de sección

Esta é *letra cursiva*, esta é **letra negrilla**, esta é letra subrallada, e esta é **letra curier**. Letra tiny, scriptsize, small, large, Large, LARGE e moitas más. Exemplo de fórmula: $a = \int_0^\infty f(t)dt$. E agora unha ecuación aparte:

$$S = \sum_{i=0}^{N-1} a_i^2. \quad (8.1)$$

As ecuaciones se poden referenciar: ecuación (8.1).

8.1.1. Un exemplo de subsección

O texto vai aquí.

8.1.2. Otro exemplo de subsección

O texto vai aquí.

Un exemplo de subsubsección

O texto vai aquí.

Un exemplo de subsubsección

O texto vai aquí.

Un exemplo de subsubsección

O texto vai aquí.

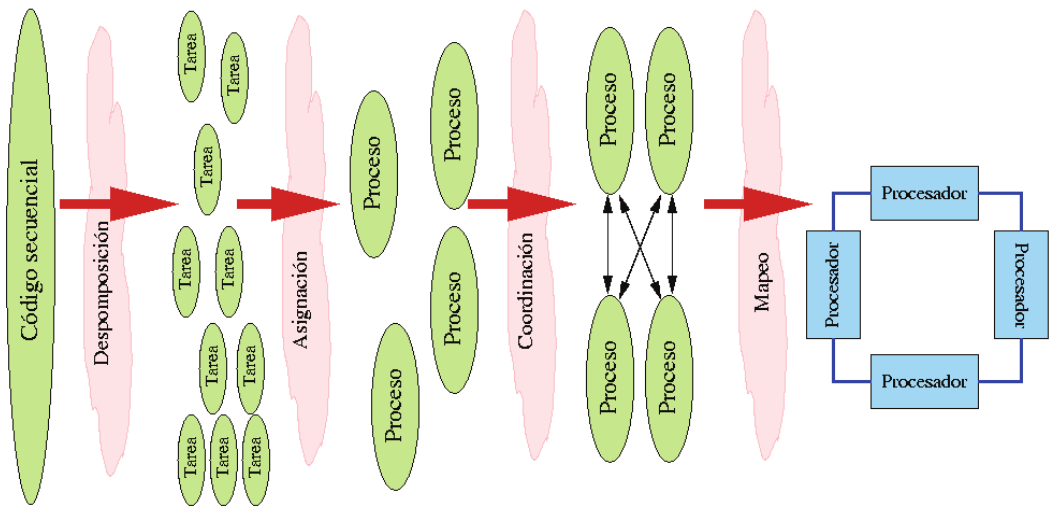


Figura 8.1: Esta é a figura de tal e cal.

Izquierda	Derecha	Centrado
ll	r	cccc
llll	rrr	c

Cuadro 8.1: Esta é a táboa de tal e cal.

8.2. Exemplos de figuras e cadros

A figura número 8.1.
O cadro (taboa) número 8.1.

8.3. Exemplos de referencias á bibliografía

Este é un exemplo de referencia a un documento descargado da web [2]. E este é un exemplo de referencia a unha páxina da wikipedia [3]. Agora un libro [4] e agora unha referencia a un artigo dunha revista [5]. Tamén se poden pór varias referencias á vez [2, 4].

8.4. Exemplos de enumeracións

Con puntos:

- Un.
- Dous.

- Tres.

Con números:

1. Catro.
2. Cinco.
3. Seis.

Exemplo de texto verbatim:

```
0 texto          verbatim
    se visualiza tal
        como se escribe
```

Exemplo de código C:

```
#include <math.h>
main()
{  int i, j, a[10];
   for(i=0;i<=10;i++) a[i]=i; // comentario 1
   if(a[1]==0) j=1; /* comentario 2 */
   else j=2;
}
```

Exemplo de código Java:

```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello_World!"); // Display the string.
    }
}
```


Capítulo 9

Conclusiones y posibles ampliaciones

Conclusiones e posibles ampliaciones

Completar
conclu-
siones y
posibles
amplia-
ciones

Apéndice A

Manuais técnicos

Manuais técnicos: en función do tipo de Traballo e metodoloxía empregada, o contido poderase dividir en varios documentos. En todo caso, neles incluírase toda a información precisa para aquelas persoas que se vaian a encargar do desenvolvemento e/ou modificación do Sistema (por exemplo código fonte, recursos necesarios, operacións necesarias para modificacións e probas, posibles problemas, etc.). O código fonte poderase entregar en soporte informático en formatos PDF ou postscript.

Realizar
manual
técnico

Apéndice B

Manuais de usuario

Manuais de usuario: incluírán toda a información precisa para aquelas persoas que utilicen o Sistema: instalación, utilización, configuración, mensaxes de erro, etc. A documentación do usuario debe ser autocontida, é dicir, para o seu entendemento o usuario final non debe precisar da lectura de outro manual técnico.

Realizar
manual
de usua-
rio

Apéndice C

Licencia

Insertar
licencia
apropia-
da

Completar
biblio-
grafía

Bibliografía

- [1] Definición de arquitectura según la ISO/IEC/IEEE 42010:2011. Artículo referenciado (<http://www.iso-architecture.org/ieee-1471/defining-architecture.html>). Consultado el 2 de junio de 2017.
- [2] Nvidia CUDA programming guide. Versión 2.0, 2010. Disponible en <http://www.nvidia.com>.
- [3] Acceso múltiple por división de código. Artigo da wikipedia (<http://es.wikipedia.org>). Consultado o 2 de xaneiro do 2010.
- [4] R.C. Gonzalez e R.E. Woods, *Digital image processing*, 3ª edición, Prentice Hall, New York, 2007.
- [5] P. González, J.C. Cartex e T.F. Pellas, “Parallel computation of wavelet transforms using the lifting scheme”, *Journal of Supercomputing*, vol. 18, no. 4, pp. 141-152, junio 2001.

Quitar
esta lista
de TO-
DOs

Todo list

Completar el resumen una vez terminado el documento	v
Completar análisis de requisitos	5
Completar los riesgos necesarios	7
Completar la metodología	7
Completar la gestión de la configuración	7
Completar la planificación temporal	8
Completar la estimación de costes	8
Completar Arquitectura del sistema	9
Completar Herramientas utilizadas	9
Completar Diseño e implementación	11
Completar Validación y pruebas	13
Completar Valoraciones finales	15
Quitar la sección de ejemplos	17
Completar conclusiones y posibles ampliaciones	23
Realizar manual técnico	25
Realizar manual de usuario	27
Insertar licencia apropiada	29
Completar bibliografía	31
Quitar la lista de TODOs al final del documento	35

