

Escuela Superior de Ingeniería

PROYECTO FIN DE CARRERA

Dominous: simulador libre de dominó

Dominous: simulador libre de dominó

Ignacio Palomo Duarte, ⁽¹⁾Inmaculada Medina Bulo y ⁽²⁾Manuel Palomo Duarte

Avenida Andalucía, nº 57, código postal 11407, Jerez de la Frontera (Cádiz). Teléfono 654907976 – correo electrónico: ignacio.palomoduarte@alum.uca.es

⁽¹⁾ ⁽²⁾ ESI I – Escuela Superior de Ingeniería C/ Chile 1, 11002 Cádiz (Spain). Teléfono 8570, +34 956 015784. Fax +34 956 0151392

Dominous es un simulador de dominó internacional, que incluye tanto el desarrollo del videojuego con una atractiva interfaz gráfica como la integración del mismo con un módulo de Sistemas Expertos que simula el comportamiento de los jugadores. Este proyecto aporta un nuevo punto de vista sobre el desarrollo de Sistemas Expertos aplicados a juegos en los que se tenga un conocimiento parcial de la situación.

Palabras Clave: Sistemas expertos, Inteligencia Artificial, dominó, videojuegos

1. Introducción

¿Por qué se ha decidido hacer un simulador de dominó? Las razones por las que se optó por desarrollar este videojuego fueron múltiples, entre las que se destacan:

1. El apartado de **Inteligencia Artificial** es muy complejo, con lo cual se puede abordar de diferentes maneras. Es un problema de elevada complejidad computacional si intentamos resolverlo mediante simples árboles de decisión: como el juego se desarrolla dentro de un marco de conocimiento limitado (no conocemos las fichas de los demás jugadores) se produce una explosión combinatoria que nos obliga a buscar otros métodos y herramientas, como técnicas de sistemas expertos. Esta búsqueda de nuevas técnicas para la resolución de un problema concreto es la base misma de la Ingeniería Informática, y es un clarísimo ejemplo de Proyecto Fin de Carrera.
2. Por otro lado el **desarrollo de videojuegos** relaciona multitud de aspectos que resultan interesantes a la hora de ser abordados, como pueden ser:

1. **Programación gráfica**, un asunto complejo que no se suele abordar dentro del plan de estudios de una Ingeniería Técnica en Informática de Gestión, y que presenta un gran número de dificultades, como resoluciones de pantalla, velocidad y optimización del sistema, diferencias sustanciales entre sistemas operativos, tratamiento de excepciones, entre otros.
 2. **Diseño de interfaces**, haciendo que la aplicación sea fácil de usar, divertida, sencilla y atractiva para el usuario final, y controlando diferentes opciones y dispositivos de entrada.
 3. **Sistema de audio**, ya que nuestra aplicación debe sincronizar actividad gráfica y ejecución de música y efectos de sonido.
 4. **Aspecto visual**, manteniendo una coherencia en cuanto a diseño gráfico de todas y cada una de las páginas, secciones y menús de toda la aplicación, guardando una uniformidad y buscando que la belleza de la aplicación se apoye en requerimientos orientados al usuario.
3. Por último, tras una concienzuda búsqueda en la red, no se ha encontrado ningún simulador de dominó de código abierto, lo que creemos que es esencial para difundir el conocimiento sobre este juego [Sta04].

2. Conceptos básicos

Dentro de los conceptos básicos a tener en cuenta para el desarrollo del proyecto se destacan tres apartados principales como son la historia de este juego, las normas básicas del mismo y unas breves nociones de Inteligencia Artificial y Sistemas Expertos.

2.1. Historia del dominó

Breve repaso a los comienzos del dominó de la mano de grandes historiadores y estudiosos de este juego, con diferentes citas como las de **Miguel Lugo** en su libro **Dominó Competitivo** [Lug08] “es entretenido y fácil de aprender. Ya desde pequeños

comenzamos a jugar al dominó con frutas o con animales en lugar de hacerlo con puntos”, o referencias a **Benito Ruipérez** [RM90] quien señala que se trata de un juego muy antiguo, de unos 400 años de historia, aunque dice que se desconoce su origen y etimología (1990:7).

2.2. Reglas básicas

El objetivo del juego es alcanzar una determinada puntuación previamente fijada, jugando para ello las manos o rondas que sean precisas. En el caso del Dominó Internacional, el **número de puntos son 200**. En esta modalidad se enfrentan dos equipos, cada uno formado por una pareja de jugadores dispuestos en la mesa de forma alternativa.

Antes de empezar, las fichas se colocan boca abajo sobre la mesa y se revuelven para que los jugadores las recojan al azar en igual número cada uno. Cada jugador cogerá 7 fichas. La primera ronda la comenzará el jugador que posea el seis doble. En las siguientes rondas, empezará el jugador a la derecha del que empezó la ronda anterior. Podrá comenzar usando cualquier ficha, no tiene porqué ser doble.

En su turno cada jugador colocará una de sus piezas con la restricción de que dos piezas sólo pueden colocarse juntas cuando los cuadrados adyacentes sean del mismo valor. Si un jugador no puede colocar ninguna ficha en su turno tendrá que pasar el turno al siguiente jugador.

La mano continúa hasta que se da alguna de las dos situaciones:

- Alguno de los jugadores se queda **sin fichas** por colocar en la mesa. En este caso el jugador se dice que dominó la partida y la pareja ganadora sumará la totalidad de los puntos no jugados, es decir, la suma de los puntos en las fichas que resten por jugar a ambas parejas.
- En caso de **cierre** – es decir, cuando a pesar de quedar fichas en juego ninguna pueda colocarse – ganará la pareja cuyas fichas sumen menos puntos. Esta situación solamente ocurre cuando el mismo número está en ambos extremos del juego, y las siete fichas de ese número ya han sido jugadas. En este caso gana la

pareja/jugador que menos puntos tenga en sus fichas, y se le suman los puntos del perdedor al ganador. Al igual que en el anterior caso, la pareja ganadora sumará la totalidad de los puntos no jugados, es decir, la suma de todos los puntos en las fichas que resten por jugar a ambos equipos.

La partida finaliza una vez que un equipo ha alcanzado o superado los 200 puntos.

2.3. Inteligencia Artificial

Para implementar la inteligencia de los contrincantes de Dominous se utilizará un **sistema experto** [Gia89]. El desarrollo de sistemas expertos es una rama de la Inteligencia Artificial, que imita los mecanismos y la forma de pensar de un experto en cierta materia para resolver problemas de su campo de aplicación. Esto lo hace adecuado para un juego como el dominó, cuyas estrategias ganadoras son conocidas. Estos sistemas tiene una gran implantación en diversas ramas de la ciencia, como medicina, ingeniería o sistemas de decisiones para negocios.

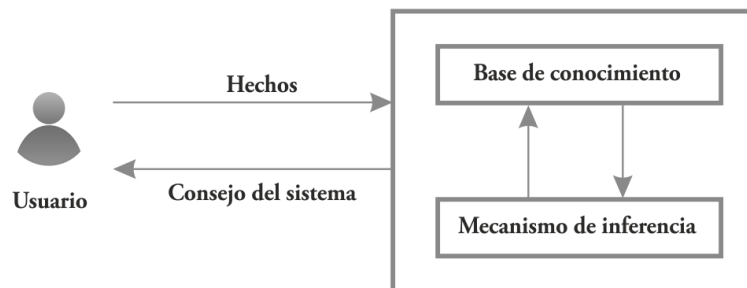


Figura 1. Estructura básica de un Sistema Experto

La memoria detalla luego los seis apartados principales en los que se divide un sistema experto basado en reglas, que son la base de conocimiento, la base de hechos, los mecanismos de inferencia, el módulo de justificación, el módulo para la adquisición de conocimiento y la interfaz de usuario

3. Planificación

La planificación del proyecto dividió al mismo en tres apartados principales (subsistema de gestión de la partida, subsistema gráfico y subsistema de Inteligencia Artificial), para más tarde elaborar un diagrama de Gantt con todas las secciones y pasos inherentes a un proyecto de ingeniería del software. En resumen, los tiempos estimados para el proyecto fueron los siguientes:

- Toma de requisitos — 27d
- Investigación preliminar — 80d
- Planificación — 27d
- Análisis del sistema — 28d
- Diseño del sistema — 265d
- Implementación — 236d
- Pruebas — 20d
- Redacción de la memoria — 62d

4. Análisis

El análisis de la aplicación ha seguido la estructura clásica de **toma de requisitos**, **diseño del modelo de casos de uso** y por último **diseño del modelo conceptual de datos**.

La toma de requisitos se desarrolló mediante reuniones con los tutores del proyecto, simulando ser clientes potenciales de la aplicación, estableciéndose requisitos de interfaces externas, funcionales y de rendimiento. Después de este proceso se pudo obtener el diagrama de casos de uso del sistema – ver **Figura 2** – para más tarde describir el modelo conceptual de datos.

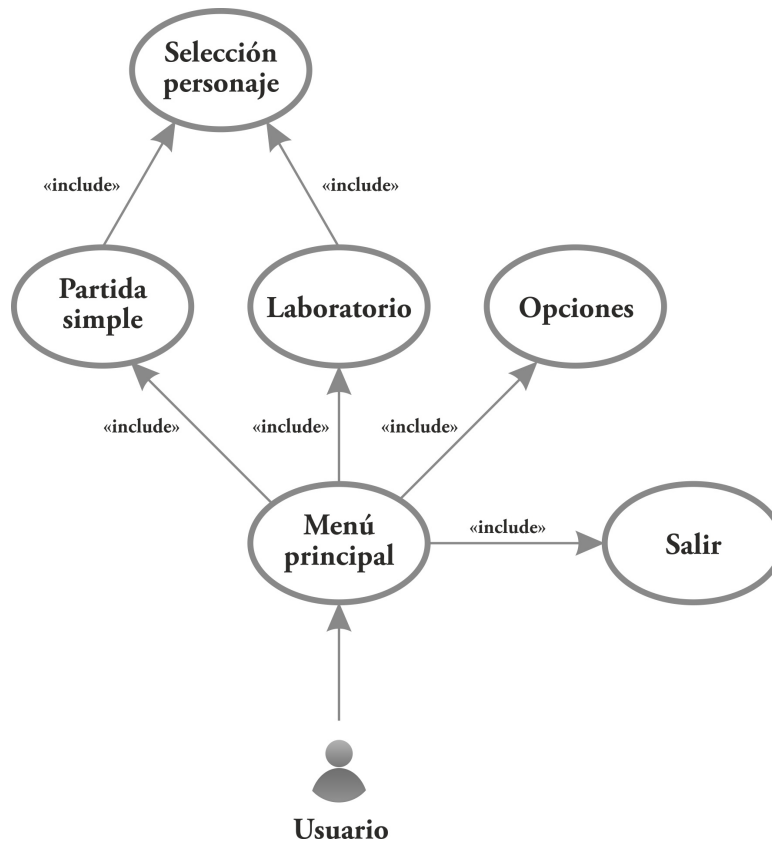


Figura 2. *Diagrama de casos de uso del sistema*

5. Diseño

5.1. Herramientas utilizadas

Para el desarrollo de Dominous se han utilizado las siguientes herramientas:

- **SDL** y **PyGame** como bibliotecas gráficas, proporcionando estabilidad, comodidad para generar gráficos 2D y disponible multiplataforma.
- **Python** como lenguaje de programación por su facilidad de escritura, ser multiparadigma e interpretado (lo cual nos facilitará la labor de crear nuevos jugadores sin necesidad de compilar) y también estar disponible para multitud de plataformas.
- **LaTeX** y **Doxygen** para documentación de código y realización de la memoria.
- **Subversion** como sistema de control de versiones.

- Otras herramientas como **vim** para escribir código, **GIMP** y **Photoshop** para edición y retoque de imágenes o **Audacity** para mejorar y trabajar con audio.

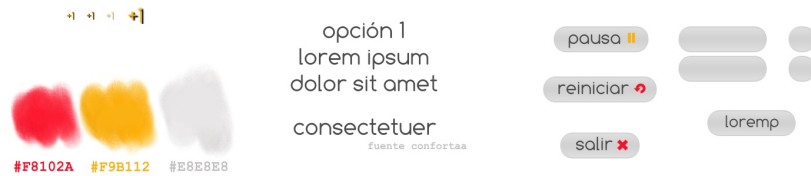


Figura 3. Diferentes recursos gráficos creados específicamente para la aplicación.

5.2. Estructura principal de la aplicación

La estructura principal de Dominous se sustenta sobre dos clases principales, una que se encarga de gestionar los flujos y transiciones entre partes de la aplicación y otra que controla la partida completa de dominó.

5.2.1 Clase dominous

La clase **dominous** genera el objeto principal que soporta todo el peso de toda la aplicación. Al inicio de la ejecución se crea un objeto de esta clase; este objeto va creando – según las necesidades del flujo que tome la ejecución de la misma – los diferentes objetos, controlando y dando paso a la interfaz activa, llamando a las principales funciones y permitiendo la captura de los eventos de entrada.

Al igual que la gran mayoría de videojuegos, para el desarrollo de Dominous se han utilizado dos métodos principales que permiten controlar la aplicación en tiempo real. Estos dos métodos son:

- **draw** – el primer método se encarga de dibujar en pantalla todos los elementos de la interfaz. El funcionamiento básico es recorrer todos los objetos activos y pintarlos según la posición, rotación y escala que tengan en ese preciso momento.
- **update** – el segundo método se ocupa de realizar los cálculos para el movimiento de los elementos gráficos. En cada iteración cambiará el estado de

los objetos activos, modificando la imagen o los valores posición, rotación y/o escala, estudiando las posibles colisiones, eventos y cualquier otra circunstancia que cambie el estado de la aplicación, para que en el siguiente ciclo se dibujen los elementos en su posición correcta.

Una vez que este objeto principal toma el control de la aplicación, su función es llamar a los métodos draw y update del objeto encargado de la interfaz que estemos mostrando actualmente, de forma alternativa.

5.2.2 Clase *dominoes_game*

Por otro lado tenemos la clase **dominoes_game**. La labor de esta clase es llevar el control de una partida completa de dominó, siguiendo las reglas del dominó internacional. Dependiendo de la configuración de la partida, creará los jugadores según la selección que se haya realizado en la pantalla de selección de personajes y comenzará con el bucle principal. Repartirá las fichas entre los diferentes participantes del juego e irá pidiendo fichas a los jugadores de forma consecutiva, hasta que se de alguna circunstancia de finalización de la mano. Este bucle se repetirá hasta que alguno de los dos equipos alcance o supere los 200 puntos, momento en el cual la partida se dará por finalizada.

Esta clase también mantendrá activo y actualizado un fichero de registro (log) con toda la información generada en la partida, que será de vital importancia para el sistema de Inteligencia Artificial.

5.3. Sistemas Expertos

El dominó es un juego muy profundo, con normas, técnicas y muchas dosis de psicología, un deporte que obliga al jugador a estar concentrado al cien por cien durante todo el desarrollo del juego, que permite que jugadores nóveles puedan adentrarse en el juego, pero que presenta mucha complicación convertirse en un gran jugador de dominó.

Ya que los Sistemas Expertos requieren utilizar el conocimiento de personas experimentadas, el desarrollo de dominous contó con la base de conocimiento de:

- **El Libro del dominó de Benito Ruipérez [RM90]**, un libro ameno y profundo sobre el mundo del dominó, con multitud de partidas explicadas movimiento a movimiento, trucos, ejemplos, técnicas básicas y avanzadas y mucha información.
- **Don Manuel Palomo Fernández de Bobadilla** – jugador amateur de dominó y participante en torneos durante más de cuarenta años, con un gran conocimiento de técnicas y métodos de juego, mucha experiencia y una gran facilidad para comunicar toda esa sabiduría y conocimiento.

Una vez analizado el juego y estudiado las habilidades que ha de tener el sistema experto, se obtuvo un subconjunto de requisitos mínimos que deben presentar los jugadores controlados por el ordenador:

- **Estrategia según la posición, reglas de obligado cumplimiento, pesos de las diferentes reglas** – estas herramientas permiten crear jugadores complejos, definiendo reglas o nuevos subconjuntos de ellas, añadiendo pesos a esas reglas según su importancia.
- **Aleatoriedad, generación de errores o sistemas imperfectos** – se debe simular en mayor o menor medida un comportamiento humano, con aciertos y errores.
- **Diferentes jugadores** – el sistema debe poder modelar diferentes tipos de jugadores, cada uno con unas habilidades concretas y diferenciadas.

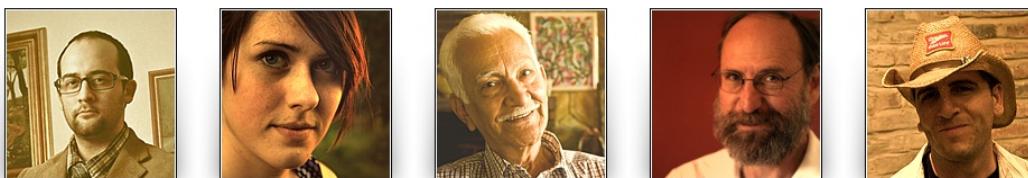


Figura 3. Los distintos jugadores añaden más interés a dominous.

6. Implementación

La implementación del proyecto se ha dividido en cinco apartados generales:

- **Entorno gráfico** – cómo se colocan las fichas unas con otras, control del flujo general entre apartados de la aplicación, máquina de estados en la partida, y otros elementos de interés.
- **Interfaz de sonido**, llamadas al objeto sound para que ejecute de forma paralela e independiente los efectos de sonido de fichas.
- **Configuración de la aplicación** mediante ficheros INI.
- **Inteligencia Artificial** incluyendo:
 - Utilización de la base de conocimiento generada por el módulo de Gestión de Partida de Dominó.
 - Base de hechos propia por cada jugador
 - Motor de inferencia con pesos, aleatoriedad y reglas por defecto.
- **Música y efectos** para dotar de cierta homogeneidad al arte del juego.

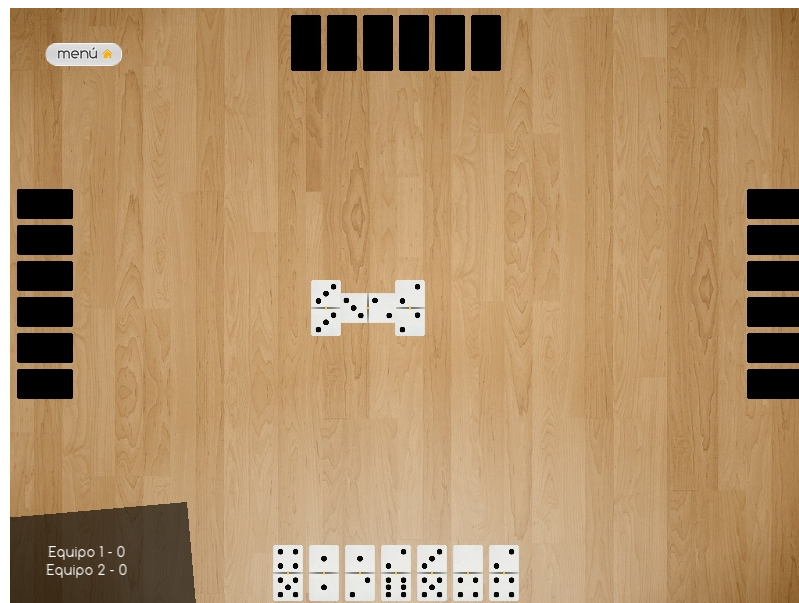


Figura 4. Captura de pantalla de una partida de dominó.

7. Pruebas

El apartado de pruebas del proyecto se completó con **pruebas unitarias** durante la fase de implementación, **pruebas de integración** según se iban desarrollando los tres principales módulos que componen dominous y por último **pruebas de jugabilidad, usabilidad y experiencia de usuario**, para las cuales se contó con cuatro usuarios reales que proporcionaron suficiente feedback para detectar diferentes errores o mejoras al producto final.

8. Conclusiones

8.1. Resultados obtenidos

Una vez terminado el proyecto es hora de realizar una valoración de los resultados obtenidos en diferentes planos, tanto profesionales como personales.

- **Obtención de un simulador libre de dominó**, licenciado como Software libre, con la posibilidad de que sea mejorado por la comunidad.
- **Aplicación de todos los conocimientos adquiridos en la carrera**, como Inteligencia Artificial, Programación Orientada a Objetos o Estructura de Datos, entre otros.
- **Aprendizaje del juego del dominó.**
- **Realización de un proyecto completo de forma íntegra.**

8.2. Trabajos futuros

Una vez completados con todos los requisitos exigidos en el pliego de especificaciones técnicas, es interesante divagar y estudiar nuevas funcionalidades o mejoras que podrían desarrollarse para la aplicación, como:

- Posibilidad de jugar en red.
- Servidor web para compartir sistemas expertos.

- Analizador de partidas con aprendizaje automático o semiautomático.
- Migración a otros sistemas.

9. Referencias

1. [Sta04] R.M. Stallman. Software libre para una sociedad libre. Traficantes de Sueños, 2004.
2. [Lug08] Miguel Lugo. Dominó Competitivo. AuthorHouse, 2008.
3. [RM90] B. Ruipérez Moral. Libro del dominó. Otero Ediciones, 1990.
4. [Gia89] Joseph C. Giarratano y Gary Riley. Expert Systems: Principles and Programming. Brooks/Cole Publishing Co., Pacific Grove, CA, USA, 1989. ISBN 0878353356.

10. Agradecimientos

El autor agradece a tutores del proyecto, familia y compañeros el apoyo, los ánimos y la paciencia que han tenido durante todo el desarrollo de la aplicación y el proyecto.