



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2233 Programación Avanzada (II/2014)

Tarea 6

1. Objetivos

- Aplicar conceptos de Networking para sincronización de programas en red.
- Profundizar contenidos de WPF.
- Terminar con la tiranía del General Zod-To.

2. Introducción

Este será el final para mí, pero aun hay esperanza para este planeta y sus habitantes. Tras la aparición del Elegido, llamado Timmy por el resto de los ermitaños, los habitantes de la ciudad Vichitus vivían en perfecta armonía... pero todo cambió cuando el General Zod-To atacó. Con las pocas tropas que le quedaban, atrapó a Timmy y al resto de los ermitaños, incluyéndome a mí, y nos arrojó a las cavernas labirínticas que conforman las Mazmorras de Jamezkaban.

Cualquiera diría que el resurgimiento de nuestra civilización se habría acabado y Zod-To habría finalmente ganado, pero Timmy puede leer la mente de la población, y muchos de ellos conocen la forma del calabozo. Algunos, como los ermitaños, quieren ayudarlo. Los partidarios de Zod-To en cambio, quieren que se quede en Jamezkaban por siempre. Timmy no puede distinguir entre las intenciones de estas voces, y obedece a todas esperando escapar.

Yo no podré escapar, pero las fuerzas del General Zod-To ya están débiles. Si Timmy lograra escapar, él podría salvar Adrianus LXIX. Me despido, con esperanzas de un futuro mejor para nuestra civilización.

Conde Halepi Ment-Hel



3. Problema

- Su tarea consistirá en desarrollar un programa que permita a un gran número de usuarios interactuar simultáneamente en un juego por medio de un chat. Todo esto deberá poder visualizarse con una interfaz gráfica desarrollada usando WPF.
- Se espera que se respeten las reglas del juego especificadas a continuación, además de manejar apropiadamente la conectividad entre usuarios.

- Es de especial importancia el buen manejo de conexiones y desconexiones en su código. Se espera que su programa funcione correctamente independiente de que un usuario se conecte o desconecte repentinamente.

4. IIC2233 Plays *Pokémon Rigomon X: Rise of the Timmys* HD¹

El juego consistirá en un personaje que se moverá en un mapa bidimensional con el fin de alcanzar un punto específico del mapa. El objetivo es llegar a dicho punto. El personaje tendrá la facultad de moverse en cuatro direcciones, además de poder interactuar con los objetos que lo rodean.

4.1. Interfaz

La interfaz gráfica deberá contener los siguientes tres bloques:

- **Situación actual del juego:** Deberá poder visualizarse en todo momento al personaje y su entorno.
- **Chat:** En todo momento, cada usuario deberá poder escribir en el chat para comunicarse con los demás jugadores. Adicionalmente, existirán palabras reservadas que al ser enviadas al chat ejecutarán distintas acciones dentro del juego:
 - **Right, Left, Up, Down:** Cada uno de estos 4 comandos permiten que el jugador se mueva en la dirección que se indica.
 - **A:** Permite interactuar con lo que esté frente al personaje, en caso de existir algo. Por ejemplo, recoger un objeto o abrir una puerta.
 - **Democracy:** Permite agregar un voto a favor de la Democracia.
 - **Anarchy:** Permite agregar un voto a favor de la Anarquía.
 - **Boom/Antiboom (Secuencia de autodestrucción):** Alimento para los trolls. Su funcionamiento se detallará más adelante.
- **Pantalla de estado:** Se debe poder visualizar la duración de la partida actual, una lista de los últimos comandos que se han llevado a cabo y la cantidad de votos a favor de los modos Democracia y Anarquía, además de mostrar un recuento de votos en el caso de que se esté en modo Democracia. Adicionalmente se debe poder visualizar la situación del botón explosivo que explota².

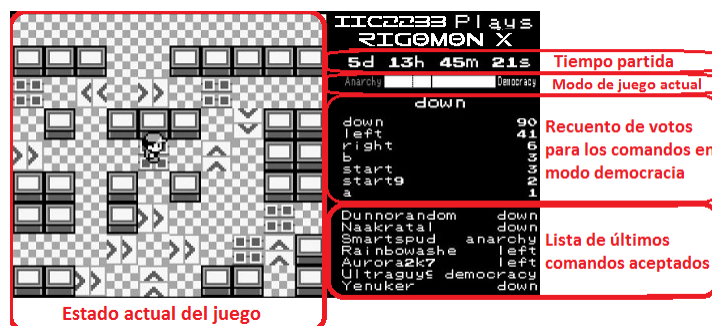


Figura 1: Ejemplo de interfaz (sin chat)

¹Algunos dirán que *Rigomon IX: The Depths of Jamezkaban* es el juego superior de la serie.

²Sí, efectivamente explota

4.2. Dinámica del Chat

El chat constituye la vía que tendrán para interactuar los usuarios, enviando mensajes directamente, o a través de comandos que serán utilizados para avanzar (o hacer cualquier cosa contraria a avanzar) en el juego. Cualquier usuario puede enviar cualquier mensaje en cualquier momento. Sin embargo, si el mensaje enviado corresponde a alguno de los comandos descritos anteriormente (*Right, Left, Up, Down, A, Democracy, Anarchy, Boom, Antiboom*), éste deberá aparecer en el chat en **negritas**.

4.3. Reglas del juego

- **Movimiento y acción:** El personaje solo podrá recibir comandos de acción cuando esté estacionario. Al recibir un comando, se tomará un instante en realizarlo, durante el cual no podrá recibir nuevos comandos. El tiempo que se demora en llevar a cabo un comando queda a juicio del programador, pero el movimiento a través del mapa debe verse fluido sin ser inapreciablemente rápido. Además, cuando un comando se lleva a cabo, debe imprimirse en la interfaz, en una lista de comandos recientes. Tras volver al modo estacionario, el primer comando escrito en el chat será el ejecutado y el personaje nuevamente no podrá aceptar nuevos comandos hasta que el actual se haya realizado.

En el caso de recibir un comando de movimiento, el personaje primero cambia su sentido de vista hacia la correspondiente. Si el personaje ya estaba mirando hacia el sentido del comando, éste se moverá un espacio en esa dirección. En el caso de que no pueda moverse en ese sentido (a causa de alguno de los obstáculos que se describirán a continuación), el personaje no se mueve y mantiene su sentido de vista.

- **Mazmorra de Jamezkaban:** Nuestro héroe está encerrado en uno de los calabozos más infames de todo el planeta Adrianus. Todos los adrianos temen de los peligros y obstáculos que es posible hallar en este lugar.
 - **Espacios vacíos:** El personaje puede moverse sobre ellos sin problemas.
 - **Entrada:** Se comporta igual que un espacio vacío. Si el personaje muere, aparecerá un nuevo Elegido en este espacio.
 - **Pared:** Obstáculo imposible de atravesar.
 - **Bordes:** Al caminar sobre ellos en una dirección específica es posible saltarlas y atravesarlas, sin embargo, son consideradas paredes en el sentido contrario.
 - **Pozo:** Si se cae sobre este espacio, nuestro héroe muere y un nuevo Timmy aparece en el espacio correspondiente a la entrada.
 - **Puerta:** Requiere que se ejecute un comando de acción (*A*) frente a ellas para poder abrirse. Una vez abiertas, permanecerán abiertas por toda la partida. Si no están abiertas, se comportan como una pared.
 - **Llave:** Hay una por mapa. Se requiere para abrir la puerta de la salida. Para recogerla el personaje debe ubicarse frente a ella y ejecutar un comando de acción. Una vez recogida, debe indicarse en la interfaz.
 - **Salida:** Todos saben que es posible escapar vivo del laberinto, sin embargo no muchos logran encontrar la puerta final antes de morir cruelmente por el botón ³. Al mirar hacia la puerta y efectuar una acción *A* de interacción, con la llave en mano, nuestro héroe será capaz de escapar con vida.

El mapa corresponde a una grilla generada de manera aleatoria que contenga los elementos descritos anteriormente. La manera en que se realice esta generación aleatoria queda a criterio del programador, pero el mapa generado debe ser coherente⁴, además de contener solo una llave, que debe poder alcanzarse por el personaje. El tamaño del mapa también es aleatorio, con dimensiones entre 20x20 y 40x40. Sin embargo, en la interfaz solo se mostrará un cuadrado de 10x10 alrededor del personaje. Al moverse éste, la sección del mapa que se muestra en la interfaz debe moverse también, con la condición que no se muestren espacios que queden fuera de los límites del mapa generado.

³Sí, el que explota.

⁴No deben haber puertas rodeadas de espacios vacíos, ni espacios vacíos rodeados por paredes, etc.

Para representar los elementos anteriores puede usarse cualquier dibujo, imagen, etc., siempre y cuando quede claro el tipo y sentido de cada uno⁵.

- **Anarquía y Democracia:** Existen 2 modos de participación colectiva en el juego.
 - **Anarquía:** Cada comando del chat es ejecutado inmediatamente en el juego (Solo si el personaje no está realizando otra acción en ese momento, en dicho caso, la instrucción se pierde).
 - **Democracia:** En este modo de juego, existirán intervalos de tiempo en el que cada usuario podrá enviar una cantidad indefinida de acciones mediante el chat, y se contabilizarán como votos. Cuando termine el período de toma de votos, se realizará la acción que fue mayoría y se iniciará un nuevo ciclo de votación.

El estado de participación se definirá por votación colectiva por medio del chat. Cada usuario puede utilizar los comandos *Anarchy* o *Democracy* del chat cuando quiera con el fin de añadir votos a su modo de participación predilecto. Se llevará un conteo los últimos 100 comandos *Anarchy/Democracy*. El juego permanecerá en el modo actual al menos que un 80 % de los comandos del conteo correspondan a votos para el modo contrario, en cuyo caso se invierte el modo de juego.

- **Secuencia de autodestrucción:** Además de todos los peligros a los que uno se enfrenta en Jamezkaban, cada prisionero es equipado con dispositivo de autodestrucción⁶. Increíblemente, muchos prisioneros han muerto a causa de este dispositivo.

La secuencia de autodestrucción tiene tres etapas, cuyo orden es: **OFF**, **READY** y **SET**. En todo momento debe mostrarse en la interfaz en qué etapa uno se encuentra. Para pasar a la etapa siguiente (y estar más cerca de la destrucción), debe usarse el comando *Boom*. Para pasar a la etapa anterior, debe usarse el comando *Antiboom*. Si se ejecuta un comando *Boom* cuando la secuencia esté en **SET**, Timmy explotará y aparecerá un nuevo Timmy a la entrada del calabozo. Esta explosión debe mostrarse en la interfaz, para herirle el alma a aquellos jugadores que pretendían avanzar con el juego (U MAD??).

5. Conectividad

5.1. LogIn

Al momento de iniciar el programa lo primero que debe hacerse es pedir al usuario un nombre de usuario, el cual debe ser única para cada persona conectada. Junto con esto se debe pedir la IP y puerto de conexión⁷. Cada vez que un usuario se conecte éste debe tener la opción de crear una nueva sesión de *IIC2233 Plays Rigomon X: Rise of the Timmys HD* o conectarse a una existente. Una vez dentro, si alguno de los usuarios se desconecta, el resto no debe perder la conexión bajo ningún caso. Es decir, debe manejarse la caída de un cliente o del servidor. La forma de implementar esto queda a su criterio.

5.2. Apariencia

Cada usuario además de tener un ID único debe también contar con un color específico de texto (no necesariamente a su elección) de modo que cada vez que este escriba quede diferenciado del resto de los usuarios mediante el color del texto.

⁵Debe a primera vista quedar claro cuáles son las puertas y hacia dónde están orientadas, por ejemplo.

⁶Que explota.

⁷Será muy penalizado no hacer esto y asumir la IP o puerto.

6. Bonus

6.1. Animaciones (5 %)

La gloria de Timmy y su aventura por las Mazmorras de Jamezkaban es infinita, y debe representarse de esa manera.

Para obtener este bonus, todos los movimientos y acciones ocurridas deben estar animados. Esto incluye que pueda distinguirse que Timmy camina al moverse, que Timmy no desaparezca instantáneamente al caer a un pozo, que el abrir de puertas esté animado, etc. Entre más completa sea la animación de todo los aspectos del juego, mayor será el porcentaje de obtención de este bonus⁸.

6.2. Emotes (5 %)

Timmy tiene una imaginación bastante viva. Al escuchar ciertos pensamientos de una persona, en lugar de palabras su mente se repleta de imágenes que no hacen mucho sentido.

Para obtener este bonus, se deberá implementar un sistemas de emotes para el chat. Si en algún momento alguien escribe *Kappa*, *Kreygasm*, *FrankerZ*, *BibleThump* o *FailFish*, en lugar de imprimirse el texto correspondiente en el cuadro de chat deberá imprimirse una de las siguientes imagenes (con un tamaño razonable que no desordene el chat).



Figura 2: Emotes que deben implementarse

7. Restricciones y alcances

- Tu programa debe ser desarrollado en el lenguaje C#, sobre .Net Framework 4.5.
- El entorno de desarrollo a utilizar debe ser Microsoft Visual Studio 2013.
- El ayudante puede bonificar o castigar tu puntaje⁹ de la tarea, si le parece adecuado. Se recomienda ordenar el código.
- No subas las carpetas `bin`, `obj` ni el archivo `.suo`¹⁰, hay un descuento de un punto en tu nota por hacerlo.

8. Entrega

- **Fecha/hora:** Jueves 4 de Diciembre de 2014 / 23:59.
- **Lugar:** En repositorios SVN personales, al interior del directorio `Tarea6`.

Debe subir la solución de Visual Studio completa en código, sin ejecutables. Tareas que no compilen o no cumplan con las restricciones señaladas en este enunciado tendrán la calificación mínima (1.0).

⁸A criterio del ayudante.

⁹ ± 5 décimas.

¹⁰Es un archivo oculto.

9. Últimas indicaciones

