

Taller de Programación I - 75.42

Trabajo práctico grupal: Z

Manual de Proyecto

Profesores:

- Roca, Pablo Daniel
- Lafroce, Matías

Grupo: 3

Integrantes:

- Lavandeira, Lucas - 98042
- Llauro, Manuel - 95736
- Rozanec, Matías - 97404

Facultad de Ingeniería, Universidad de Buenos Aires

1º Cuatrimestre, 2017

	1
Enunciado	3
División de tareas	4
Cronograma	4
Inconvenientes Encontrados	6
Análisis de puntos pendientes	7
Análisis del estado del servidor	8
Herramientas	9
Git	9
Glade	9
CLion, by JetBrains	9
Conclusiones	10

Enunciado

La consigna del proyecto es la implementación de Z, un juego de estrategia en tiempo real 2D, multijugador por red. El proyecto incluye dos programas principales, un servidor que maneje la lógica del juego y varios clientes, uno por jugador, con la representación gráfica del juego en pantalla, y sonidos. Tiene funcionalidad de equipos, de manejo de unidades y control de fábricas y territorios capturables y destruibles.

El proyecto debe ser desarrollado completamente en ISO C++, valiéndose de librerías gráficas para la representación del juego, y de librerías de interpretación de XML para manejo de archivos de datos persistentes.

Al final de la documentación se presenta el enunciado original completo.

División de tareas

Se dividieron las tareas de implementación en tres grandes bloques, uno por cada integrante del grupo:

- Servidor, con toda la lógica del juego, desde el movimiento y creación de unidades, los ataques y disparos, la destrucción de estructuras, hasta las condiciones de victoria, además de la comunicación con los jugadores.
- Cliente gráfico, con el dibujado de todo el área del juego, incluyendo la visualización del mapa de juego, las unidades, los edificios y estructuras.
- Modelo del cliente y generador de mapas, un tercer bloque que tiene como responsabilidad la comunicación del cliente con el servidor, el diseño de menús de interfaz gráfica, y una herramienta para el servidor que genera aleatoriamente mapas de juego.

La división de tareas fue respetada a lo largo del proyecto, aunque las cuestiones de diseño mayores fueron abarcadas grupalmente.

Cronograma

El proyecto fue desarrollado a lo largo de 8 semanas. Aproximadamente se dividieron las tareas en el siguiente orden:

- Semana 1: Análisis del enunciado y diseño preliminar del proyecto
- Semana 2: Comienzo de implementación del algoritmo A* en el servidor, instalación de librerías de parseo de XML, gráficos y sonidos.
- Semana 3: Interfaz gráfica base de la ventana de juego, generación de mapas básicos
- Semana 4: Dibujado del mapa entero en el cliente, modelo de unidades y fábricas en el servidor, generación de algunas estructuras en el mapa
- Semana 5: Carga de mapas en el servidor, ataque y movimiento de unidades en el servidor, funcionalidad de cámara en el cliente.
- Semana 6: Conexión entre cliente y servidor, envío del mapa a través de sockets, vistas previas a la del juego (inicial, lobby)

- Semana 7: Sincronización entre cliente y servidor, movimiento y ataque de punta a punta
- Semana 8: Creación de unidades de punta a punta, inicio y finalización de una partida entera, opción de seleccionar el mapa a jugar, captura de territorios

Inconvenientes Encontrados

Durante el desarrollo del proyecto se encontraron varios problemas a medida que se encaraban distintos aspectos del programa, que causaron que la implementación de toda la funcionalidad pedida no pudiera completarse. Los dos ejes más notables fueron la falta de organización grupal en determinar objetivos claros semana a semana, y la poca familiaridad con las librerías gráficas utilizadas (*gtkmm* y *cairo*). Todos estos factores llevan a un programa de envergadura más simple que el que fue propuesto al inicio del proyecto.

Análisis de puntos pendientes

Por motivos diversos, al final del desarrollo del proyecto quedaron los siguientes puntos pendientes:

- uso de sonidos: si bien se cumplió con el ítem del cronograma que indicaba que la primera semana había que *emitir un sonido*, lo que implicó buscar la librería más adecuada y entender cómo integrarla al proyecto, esto no se logró integrar a la totalidad del proyecto según solicitado.
- selección grupal de unidades: si bien el tema de selección fue encarado de una forma que permite la fácil expansión de selección individual a selección grupal de unidades, éste último tipo de selección no se logró implementar. La selección se realiza considerando los puntos en que el usuario baja el botón del mouse para hacer click y en que el usuario suelta el botón del mouse, es decir que se marca un 'rectángulo' dibujado con el mouse, procesando como seleccionadas las unidades que quedan en susodicho rectángulo. Actualmente se selecciona una sola unidad. Además la precisión del seleccionado parece no ser del todo precisa en algunos casos concretos, por ejemplo varias unidades cerca de una fábrica.
- generación de mapas más complejos: lo más notorio para destacar en lo que falta de generación de mapas es la generación de puentes, y la generación de climas distintos. Los mapas finales entregados sólo ofrecen terrenos de tipo llanuras, con ríos de lava y agua. Además, la generación de fábricas parece haber quedado incompleta, con algunas siendo ubicadas dentro de celdas de agua.
- manejo de la cámara del juego con el mouse: se optó por dejar la versión más simple de utilizar únicamente las flechas direccionales.
- Animaciones de los distintos tipos de ataques (balas, misiles, láser, etc)

Otro aspecto a destacar es la limpieza general del código escrito. Varios sectores del programa quedaron con una implementación mejorable, ya sea por problemas de orientación a objetos (responsabilidad múltiple, oportunidades a polimorfismo no aprovechadas), o por prolijidad general del código.

Análisis del estado del servidor

El servidor tiene un hilo aceptador de clientes, y otro hilo principal donde está el menú del juego. Dentro del menú existe un comando para crear un Lobby que es la instancia previa a crear una partida. Con un máximo de 4 jugadores, se puede iniciar la partida, comenzando a correr otro hilo con la misma.

Vale la pena aclarar que como las condiciones de la partida se crean en el lobby, del lobby al Game se producen por única vez algunos pasajes por copia de estructuras grandes, pero son necesarias ya que al salir del scope del Lobby se perderían los datos sino.

Herramientas

Durante el transcurso del desarrollo se utilizó un repositorio de *git* para el control de versiones entre los integrantes del grupo, *Glade* para editar las interfaces gráficas, y los tres integrantes usamos *CLion* de JetBrains como IDE para el desarrollo del programa entero, aprovechándose de sus herramientas de depuración.

Git

El uso de un repositorio hosteado en GitHub le otorgó al grupo mucha flexibilidad, pudiendo estar trabajando cada uno de los grandes aspectos del trabajo en *branches* y *mergeando* los avances a medida que éstos se encontraban completos y listos para unirlos con el resto del desarrollo.

Además se intentó hacer uso del feature *Issues* de GitHub. Si bien se utilizó, la poca familiaridad de los integrantes hizo que no haya sido la principal herramienta para guiar el desarrollo.

Glade

El uso de *Glade* favoreció especialmente el diseño general de las distintas ventanas. Así, una vez generadas las ventanas con sus distintos elementos estáticos, se pudieron importar en el código y programar sobre eso el comportamiento deseado.

CLion, by JetBrains

Para el desarrollo del proyecto se decidió usar el IDE *CLion*, haciendo uso de las distintas herramientas de desarrollo que brinda, destacando:

- las distintas herramientas de refactoring, detección temprana de errores, ayuda con los distintos imports.
- debugger.
- herramientas que ayudan en la navegación del código, cosa que a medida que fue creciendo el proyecto fue cada vez más valorado, como la búsqueda de términos en todo el proyecto y Ctrl + click para navegar a la declaración o definición de lo seleccionado.

Conclusiones

La falta de experiencia por parte de los integrantes en el desarrollo de proyectos de esta escala afectó negativamente el trabajo realizado. Los errores cometidos en la organización causaron, a pesar de tener un cronograma detallado de las metas semanales, un retraso notable en el progreso del proyecto, y consecuentemente del resultado final. El proyecto sirvió en parte como experiencia de administración eficiente de los tiempos.

También se pudo ver lo que es el desarrollo de una aplicación de una escala mayor a la típica vista en otras materias de la carrera hasta el momento, y de los problemas que surgen en los proyectos de magnitud considerable. Se puede destacar la importancia de tener un diseño simple y eficiente al principio, pero que sea a la vez escalable.