

Documento de diseño proyecto estructuras de datos 2023-3

Juan Felipe Morales Espitia

Daniel Aleiandro Terán Escobar

Resumen—Este documento es una guía básica del diseño de la aplicación propuesta para realizar como proyecto de la asignatura de estructuras de datos de la Pontificia Universidad Javeriana, el proyecto tiene como objetivo realizar un sistema de apoyo para el juego RISK.

Abstract—This document is a basic guide for the design of the application proposed as a project of the data structures course of the Pontificia Universidad Javeriana, the project aims to create a support system for the game RISK.

I. INTRODUCCIÓN

El juego RISK es originalmente un juego de mesa basado en la guerra y toma de territorios, se juega en turnos y el ganador es aquel jugador que tome una cantidad de territorios del juego o tome todos los territorios.

El sistema de apoyo descrito en este documento va a ser un programa que recibe comandos escritos textualmente en la consola indicando al sistema las diferentes acciones que se desean realizar, los comandos se dividen en componentes dependiendo de la acción que el jugador quiera realizar.

Existen tres componentes dentro del sistema, los cuales son la configuración del juego, el almacenamiento de partidas y las estrategias de juego.

II. COMPONENTE 1 (CONFIGURACION DEL JUEGO)

En este componente estan definidos los comandos que se pueden ejecutar para configurar el juego e inicializarlo, cada commando cuenta con una serie de funciones que realizan diferentes instrucciones para el correcto funcionamiento del programa, el dividir las instrucciones del programa en funciones facilita la implementación, el mantenimiento y la comprensión de este.

Los diferentes comandos que incluye este componente son “inicializar”, “turno” seguido de el ID del jugador y el comando “salir”. Estos comandos tienen la siguiente estructura.

A. Inicializar

Este comando se utiliza para iniciar el juego, en caso de que el juego no esté inicializado lo va a iniciar, lo primero que hace el programa es llamar a la función leer jugadores, esta función

es preguntar por la cantidad de jugadores, este numero debe estar entre 3 y 6, en otro caso manda un mensaje de error indicando que el valor no es válido, posteriormente por cada jugador pregunta por su ID o su nombre, el programa tiene una lista de colores que los jugadores pueden elegir para representar su ejército, se le pregunta al jugador cual color desea, si el color no esta en la lista, manda un mensaje de error diciendo que escoja otro color, si el color está en la lista lo busca en esta y lo borra para que el otro jugador tenga que escoger un color diferente, esta función retorna el color asignado, después se llama a la función de asignar infantería, esta función retorna la cantidad de tropas dependiendo de la cantidad de jugadores, mientras mas jugadores hay se le asignas menos tropas, esta función retorna la cantidad de tropas de infantería que le pertenecen al jugador, por ultimo se crea el jugador teniendo como parámetros el nombre o id, el color que escogió y la cantidad de tropas que le pertenecen. Y por ultimo se añade a un vector de jugadores, cuando ya se añadieron al vector de jugadores todos los que están en el juego, luego se añaden los territorios que cada jugador va a ocupar inicialmente.

Primero se leen los continentes, para esto se tiene un archivo con los nombres de los continentes y los países que pertenecen a este, se llena un vector de continentes, este es el que se usa para todas las operaciones que se hacen con este, luego se verifica que un jugador no haya conquistado todos los continentes, y se asignan los países, para esto el jugador escoge el continente y luego el país al cual le desea asignar una de sus tropas, luego de que se haya verificado que el país si esta disponible se le resta una tropa al jugador, y se marca como ocupado el país. Y después de que cada jugador tenga desplegadas todas sus tropas, se muestran los territorios que cada jugador ha ocupado.

Y finalmente se pide llama a la función pedir cola jugadores, esta función lo que hace es hacer una cola con todos los jugadores en la partida.

B. Turno <id-jugador>

Este comando se utiliza para el turno de cada jugador, primero se llama a la función buscar jugador, esta función lo busca en el vector de jugadores y retorna la posición en la que este se encuentra, también se mira la cola de jugadores y si el id del jugador indicado no está al frente de la cola, esto significa que no es el turno de este jugador, luego se ejecuta el turno del jugador indicado, este jugador se elimina de la cola y se vuelve a añadir al final de la cola.

Para la ejecución del turno del jugador se necesita la

posición anteriormente obtenida, cada jugador en su turno tiene que realizar cuatro acciones, ubicar su ejército, agregar tropas, atacar y reforzar sus territorios, cada acción tiene su respectiva función dentro del programa y se ejecutan en secuencia para cada turno.

En primer lugar, se llama a la función ubicar ejército, para esto se busca cuantos territorios tiene el jugador, este numero se divide entre 3, estas son las tropas que se le añaden al jugador en su turno, también a este numero de tropas se le añade un numero de tropas si han conquistado algún continente, este numero de tropas depende del continente, y este numero de las tropas por territorio mas las tropas por continente se le añaden al jugador con la función sumar infantería.

Luego se llama a la función agregar tropas a territorios, esta función muestra los territorios actuales con los cuales cuenta el jugador, después le pide al jugador el país al que le desea agregar tropas, y se verifica si el jugador tiene el país recorriendo la lista de países y comparando el nombre, después busca la posición del país y del continente seleccionado que son números enteros guardados en un vector, después se le pide al jugador la cantidad de tropas que desea agregar y este número se le resta a la cantidad de tropas con las que el jugador cuenta, y con el numero de tropas se le suma al país teniendo en cuenta las posiciones obtenidas anteriormente, y se muestran con cuantas tropas quedó el país, y se le pregunta a que otro país desea agregar tropas, esto siempre y cuando el jugador tenga tropas disponibles.

Luego se llama a la función atacar, en esta función primero se muestran los territorios que el jugador tiene, se pide el nombre del país desde el cual desea atacar y el país al que desea atacar, cuando se tiene un país y este no lo posee el jugador inicia la batalla, estos países tienen que ser países vecinos, en la batalla, dependiendo de los números de los dados se decide si el jugador gana o pierde la batalla.

Finalmente se llama a la función fortalecer, en esta función se pregunta el país desde el cual desea movilizar tropas y al cual las desea mover.

C. Salir

Este comando termina la ejecución del programa.

III. COMPONENTE 2 (ALMACENAMIENTO DE PARTIDAS)

En este componente están definidos los comandos relacionados con el almacenamiento de las partidas, estos comandos permiten guardar el estado actual de la partida en un archivo o binario o de texto, de igual manera se pueden leer estos archivos para que se pueda continuar con una partida que ya había sido guardada.

A. Guardar <nombre_archivo>

Este comando se utiliza para poder guardar el estado actual de la partida para posteriormente continuar con esta mas adelante, en primer lugar se verifica si ya se ha inicializado la partida, de lo contrario no habrá ninguna información para guardar, si ya se inicializo la partida, se crea un archivo de

texto con el nombre indicado en el comando, luego la primera línea que va a aparecer en el archivo es "Turnos", esto le indica al programa de quien es el siguiente turno y que jugadores van después, después se escribe el nombre o id de cada jugador en el orden que le corresponde, después de la sección de turnos sigue la información de los jugadores y continentes, para esto se escribe en el archivo el carácter "#", para indicarle al programa se acaban los turnos, después se llama a la función archivotexto que guarda toda la información de la partida en una sola cadena de texto, en la cadena de texto primero se inserta la palabra "jugadores", luego un salto de línea y se inserta la información de cada jugador, en primer lugar se escribe el nombre y color de cada jugador separado por un "/", después las tarjetas que tiene cada jugador, cada tarjeta tiene el id de continente, el id del país, y el ejército, cada atributo separado por un "-", después se pone un "/" y se inserta la información de los ejércitos, cada ejercito tiene la cantidad, el tipo y el color, de igual manera se separa con un "-", por ultimo cada jugador tiene una serie de territorios, la información de sus territorios se guarda de la siguiente manera, se guarda el territorio y su numero separado por un "-", y así se hace por cada jugador, cuando la se tiene toda esta información almacenada en una sola cadena de caracteres esta se escribe en el archivo.

B. Guardar_comprimido <nombre_archivo>

Este comando se utiliza para guardar la partida utilizando la codificación Hoffman, para esto se guarda en un mapa cada carácter y su frecuencia, después se crea un árbol binario, en este árbol solo los nodos hoja tienen los datos, y el camino para llegar a estos es el código que se utiliza en el comprimido, el 0 es izquierda y el 1 es derecha, y son estos códigos los que se escriben en el archivo binario, este proceso se realiza para cada uno de los caracteres de la forma en la que se guardan los datos indicada en el anterior comando llamado "guardar".

C. Inicializar <nombre_archivo>

Este comando se utiliza para poder inicializar y continuar una partida ya antes guardada, este comando puede ser utilizado tanto para cargar partidas de un archivo de texto como de un binario, en primer lugar se mira la extensión del archivo indicado si es ".txt" o ".bin - .dat", si es .txt se realizan las mismas instrucciones que en guardar, pero a la inversa, es decir, se crea una cola de jugadores y por cada línea que se lea después de "Turnos", se va guardando en esa cola los nombres que aparezcan ahí, después se leen las tarjetas, ejércitos y territorios de cada jugador, por ultimo para el juego como tal se guarda la información de los continentes. El mismo proceso se hace para los archivos binarios solo que en este caso primero se decodifica cada carácter y ahí si se guarda en la memoria del programa.

IV. PLAN DE PRUEBAS GUARDAR_COMPRIMIDO

A. *Guardar sin nombre*

- Descripción: se prueba este comando sin indicarle ningún nombre para que guarde la información de la partida.
- Datos ingresados: el comando por consola seria únicamente “guardar_comprimido”.
- Retorno obtenido: se guarda los datos de la partida en un archivo binario llamado guardar_comprimido.bin.
- Retorno esperado: se espera que no se guarde los datos de la partida y que muestre un mensaje de error indicando que se necesita un nombre de archivo, esto con el fin de poder diferenciar las partidas.

B. *Guardar con nombre*

- Descripción: se prueba este comando indicándole algún nombre para que guarde la información de la partida.
- Datos ingresados: el comando por consola seria para esta prueba “guardar_comprimido prueba_comprimido”.
- Retorno obtenido: se guarda los datos de la partida en un archivo binario llamado prueba_comprimido.bin.
- Retorno esperado: se espera que se guarde la partida en un archivo binario llamado prueba_comprimido.bin.