

PROYECTO FINAL DE PROGRAMACIÓN ORIENTADA A OBJETOS “JUEGO DE OBSTÁCULO: FELIX VS SONIC”



**HÉCTOR AUGUSTO DAZA ROA
ANDRÉS MIGUEL IRIARTE PADILLA**

CONTENIDO

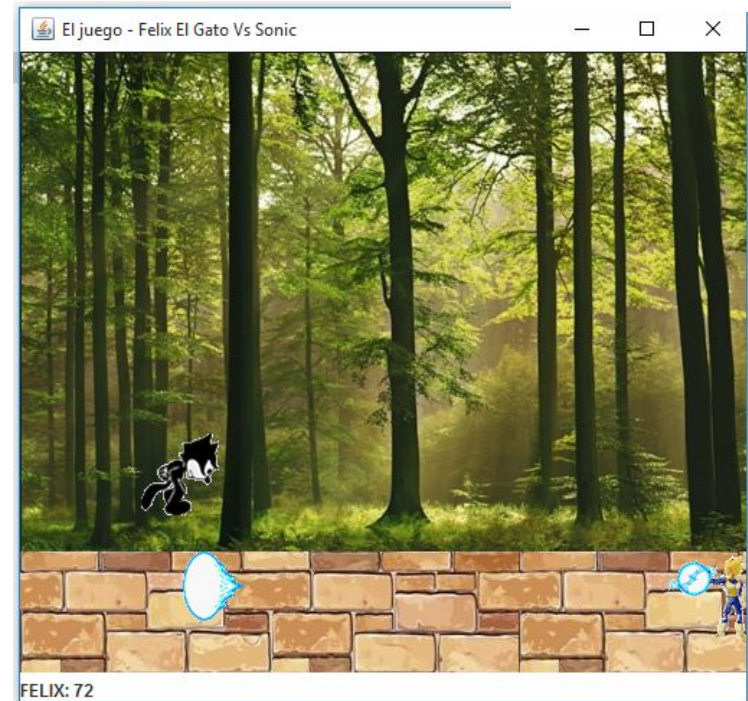


1. ¿En qué consiste el proyecto?
 - 1.1 Elementos del juego
2. Estrategias de diseño
3. Aplicación de conceptos
 - 3.1 Programación Orientada a Objetos
 - 3.2 Interfaz gráfica de usuario (Java Swing)
 - 3.3 Persistencia (almacenamiento)
 - 3.4 Programación multihilo
4. Conclusiones
5. Bibliografía

1. ¿En qué consiste el proyecto?



- Juego de obstáculo
- El puntaje depende de la cantidad de bombas esquivadas
- Multijugador por turnos (Félix o Sonic)
- Se puede variar el nivel de dificultad
- El puntaje es guardado en el Disco Local (C:)
- El jugador que toque a Vegeta gana.



1.1 Elementos del juego



- **Personajes**

Felix:



Sonic:



Vegeta:



Bomba:



- **Escenario**



2. Estrategias de diseño



1. Concepción de una idea viable
2. Ajustes en relación a los conceptos aprendidos
3. Búsqueda de recursos gráficos
4. Diseño de diagrama de clases. **ARCHIVO PDF**
5. Implementación del diseño en código fuente

3. Aplicación de conceptos



3.1 Programación Orientada a Objetos

3.2 Interfaz gráfica de usuario (Java Swing)

3.3 Persistencia (almacenamiento)

3.4 Programación multihilo

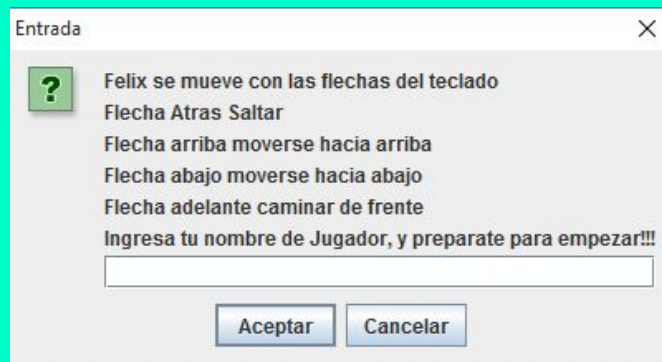
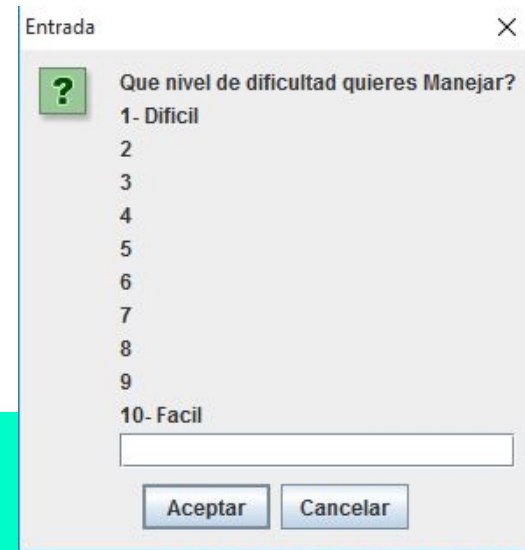
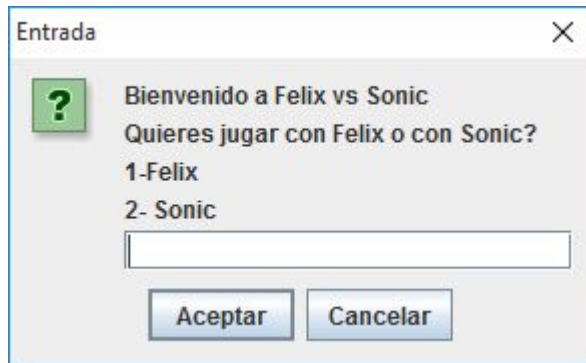
3.1 Programación Orientada a Objetos (POO)



La estructura de clases y objetos es aplicada en todo el programa, se construyeron dos proyectos, uno para el jugador (cliente) y otro para el observador (servidor), cada uno de los cuales posee los paquetes dto, hilos, image, juegoFelix y vista, además de las clases Personaje, MoverPersonaje, JuegoFelix, PuntajeRegistro, PanelDibujar, Tablero e Hilo respectivamente.

3.2 Interfaz gráfica de usuario (Java Swing)

Se hace uso de herramientas gráficas como las clases JPanel (panel), JFrame (ventana), JLabel, JOptionPane y Graphics

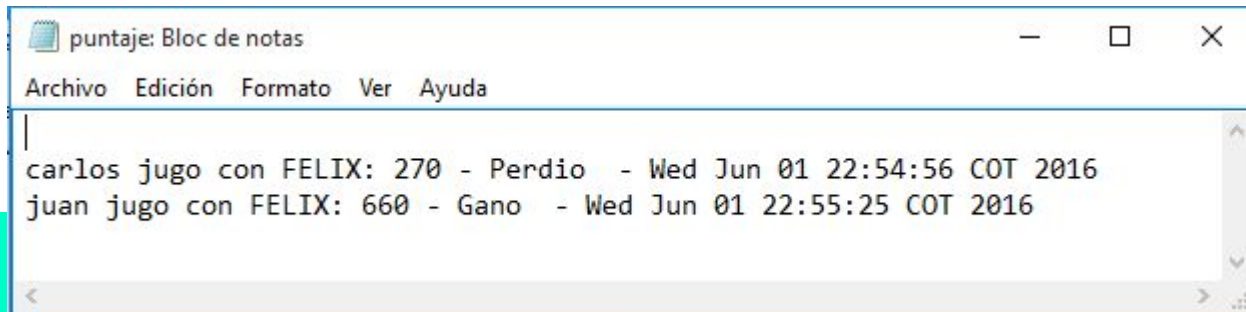


3.3 Persistencia (almacenamiento)



Los puntajes son almacenados en un documento de texto en la dirección “C:\\FelixVsSonic/puntaje.txt”. El formato con el que se almacena la información es el siguiente:

Nombre del jugador “jugo con” personaje : “perdio” o “gano”
- fecha



3.4 Programación Multihilo



El proyecto cliente posee la clase moverPersonaje que hereda de la clase Thread (hilo). El proyecto servidor posee la clase Hilo que implementa la interfaz runnable. Cada clase manipula los distintos movimientos que los personajes pueden ejecutar, esto incluye recibir datos desde el teclado para que el usuario controle al personaje principal (cliente) o en vez de eso recibir los datos del cliente para hacerlo (servidor).

4. Conclusiones



- El paradigma de la programación orientada a objetos permite abordar mejor los problemas que el paradigma de la programación estructurada.
- Una correcta planificación y diseño del proyecto facilitó su elaboración y la adecuada aplicación de los conceptos.

5. Bibliografía



- Barker J., ***Beginning Java Objects: From Concepts to Code***, 2nd Edition. Apress, 2005.
- Niemeyer P., Leuck D., ***Learning Java***, 4th Edition. O'Reilly, 2013.
- Sierra K., and Bates, B. ***Head First Java***, 2nd Edition, O'Reilly Media, 2005.
- Herramientas informáticas: Netbeans y Creately.



¡GRACIAS!

