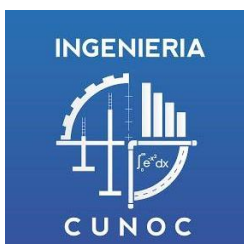


PROYECTO FINAL



“Juego RPG con UI”

LUIS EDUARDO GUILLÉN RUIZ

UNIVERSIDAD SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE OCCIDENTE “CUNOC”
DIVISIÓN DE CIENCIAS DE LA INGENIERÍA
INGENIERÍA EN CIENCIAS Y SISTEMAS
PRIMER SEMESTRE
2,019
Ing. OLIVER ERNESTO SIERRA

“ID Y ENSEÑAD A TODOS”

Contenido del Manual

Contenido del Manual	2
Presentación	3
Objetivos del Programa	3
Generales	3
Específicos	3
Instrucciones de Programa	4
Diagrama de Clases	4
Requerimientos Técnicos	5
Herramientas de Construcción	5
Como Acceder al Programa	5
Descripción del Programa	6
JavaDoc	6
Ver registros	6
Juego	7
Glosario (tecnicismos)	7
Reporte de Programa	8

“Manual Técnico, Juego RPG con UI”

Presentación

El presente documento es el manual Técnico del Juego de Combate en consola que se presenta como parte de la Practica 1. Este juego es un RPG (role playing game) de combate en el que el jugador cuenta con 3 vehículos para atacar a enemigos sobre un escenario. Crear Vehículos o escenarios según le parezca al jugador siempre y cuando vaya desbloqueando estas opciones a través de la experiencia y puntaje en el juego la cual se obtendrá de acuerdo a sus capacidades y tiempo de uso de la consola.

El juego está diseñado en lenguaje JAVA y es reproducible (.jar) por medio de cualquier IDEA con la capacidad de interpretar este lenguaje de programación, así mismo se incluye el funcionamiento básico del mismo y una guía para su primera ejecución y posteriormente el uso del mismo.

Objetivos del Programa

Generales

- Familiarizar al estudiante con el lenguaje Java.
- Aplicar conceptos de programación orientada a objetos recibidos en clase magistral y laboratorio.
- Practicar con la interfaz gráfica (GUI) de una aplicación de escritorio de java implementando los conceptos aprendido durante el curso.
- Elaborar la lógica para la solución del problema planteado.

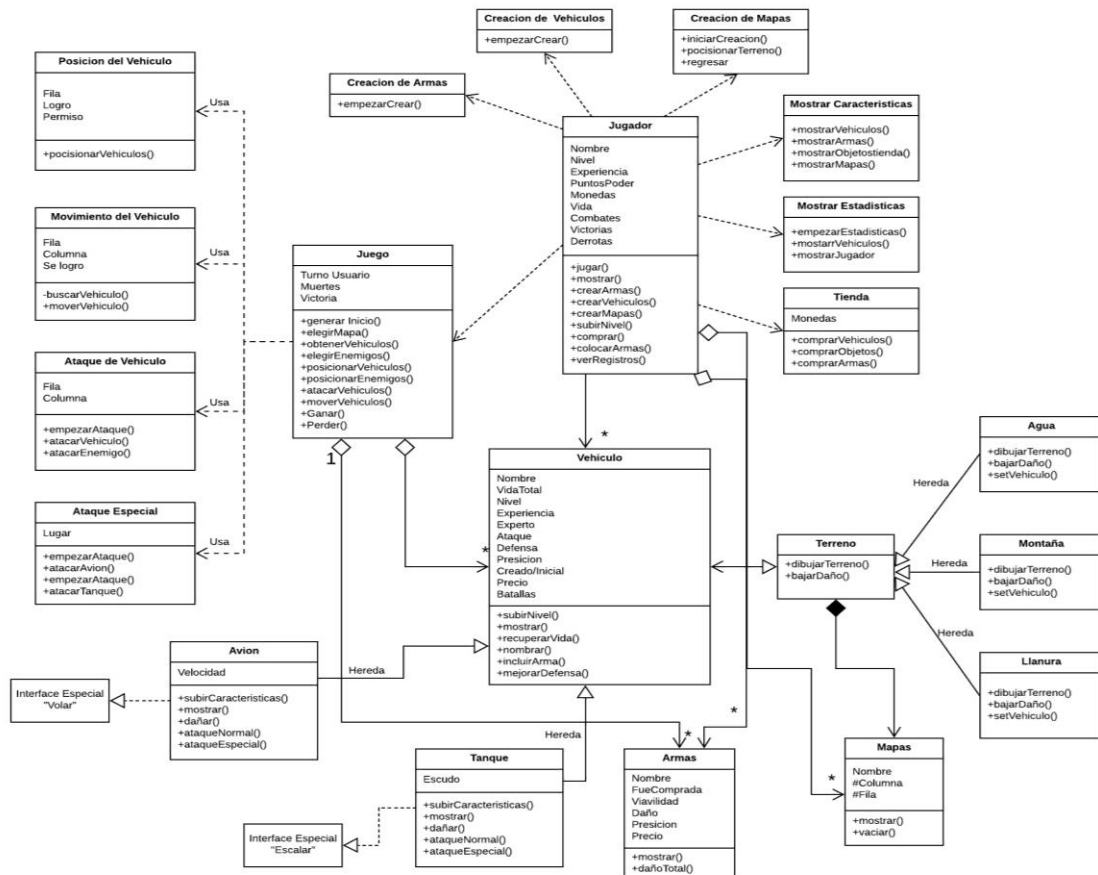
Específicos

- Construcción de algoritmos para los requerimientos de la actividad.
- Ampliar el conocimiento de Programación orientada a objetos en JAVA.
- Implementación de ciclos, sentencias de control y arreglos.
- Implementación de clases, herencia y polimorfismo.
- Usar de interfaz gráfica en java, utilizando frames, buttons, entre otros.
- Usar de Recursividad.
- Usar de Threads (hilos).
- Ampliar el conocimiento del lenguaje JAVA.
- Desarrollar diagramas de clase como parte del análisis del problema.
- Desarrollar de manual técnico.

Instrucciones de Programa

1. Ejecutar Inicio
2. Elegir Jugar, Reportes, Exit
3. Tienda
4. Nueva Partida o Cargar Partida
5. Crear Nickname
6. Elegir Componentes
7. Registro
8. Elegir Escenario
9. Juego
10. Tirar Dado
11. Continuar nivel
12. Finalizar Partida
13. Guardar
14. Reportes
15. Regresar
16. Salir

Diagrama de Clases



Requerimientos Técnicos

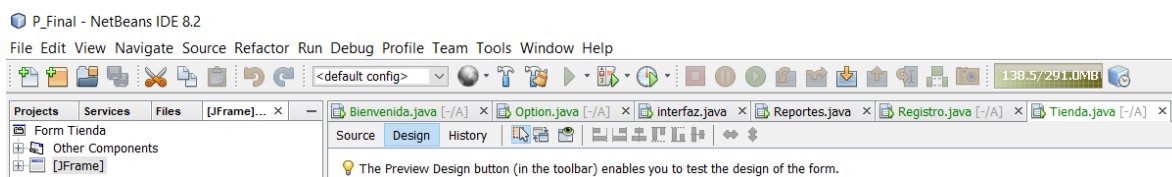
- SO que soporte Java (Ejemp.: Linux, Windows, IOs, Android).
- Java 1.8.0_201 o compatibles.
- NetBeans IDEA (creación: v. 8.2) o cualquier editor de lenguaje JAVA.
- .Jar ejecutable o proyecto completo.
- Entender lenguaje JAVA (para ver creación de programa).
- Experiencia previa en Interfaz Gráfica

Herramientas de Construcción

- NetBeans IDEA
- Package de JAVA
- Clases de JAVA
- Métodos creados en JAVA
- Interfaz de JAVA
- PC funcional
- Conocimientos de Programación

Como Acceder al Programa

Se debe tener el proyecto completo luego se carga a la terminal por medio de NetBeans Idea o cualquier IDEA con la capacidad de interpretar el lenguaje JAVA, siempre verificando la compatibilidad del mismo. Una vez cargado el código correspondiente al Juego RPG con GUI, se ubica la carpeta Inicio y se compila el código a través de la opción de NetBeans para acceder a la consola y a su vez ejecutar la Interfaz Gráfica por medio de JAVA Jar.



Cargar y Ejecutar Proyecto

Descripción del Programa

Al iniciar se tendrá la opción de elegir entre las opciones de Nueva Partida y Cargar Partida posteriormente se le solicitara un Nickname que debe ser el cual use siempre que desee jugar (esto para las estadísticas), además de elegir cuales son los vehículos iniciales que se utilizaran desde el primer nivel, la experiencia es de 0, se poseerá un arma básica por vehículo seleccionado.

Un turno se compone de una acción del jugador y acciones de los enemigos, ejecutando primero la acción de quien inició el combate (random), la arena donde se pelee está compuesta por valles, montañas y agua esto con el fin de poner restricciones a los vehículos. Se permite atacar, cambiar de vehículo, Utilizar Items (comprados en la tienda antes de iniciar el juego) y Rendirse. El juego se dará por finalizado al perder todos los vehículos o salir de manera rápida (rendirse). Está disponibilidad es en el combate vs CPU como también en el P1 vs P2.

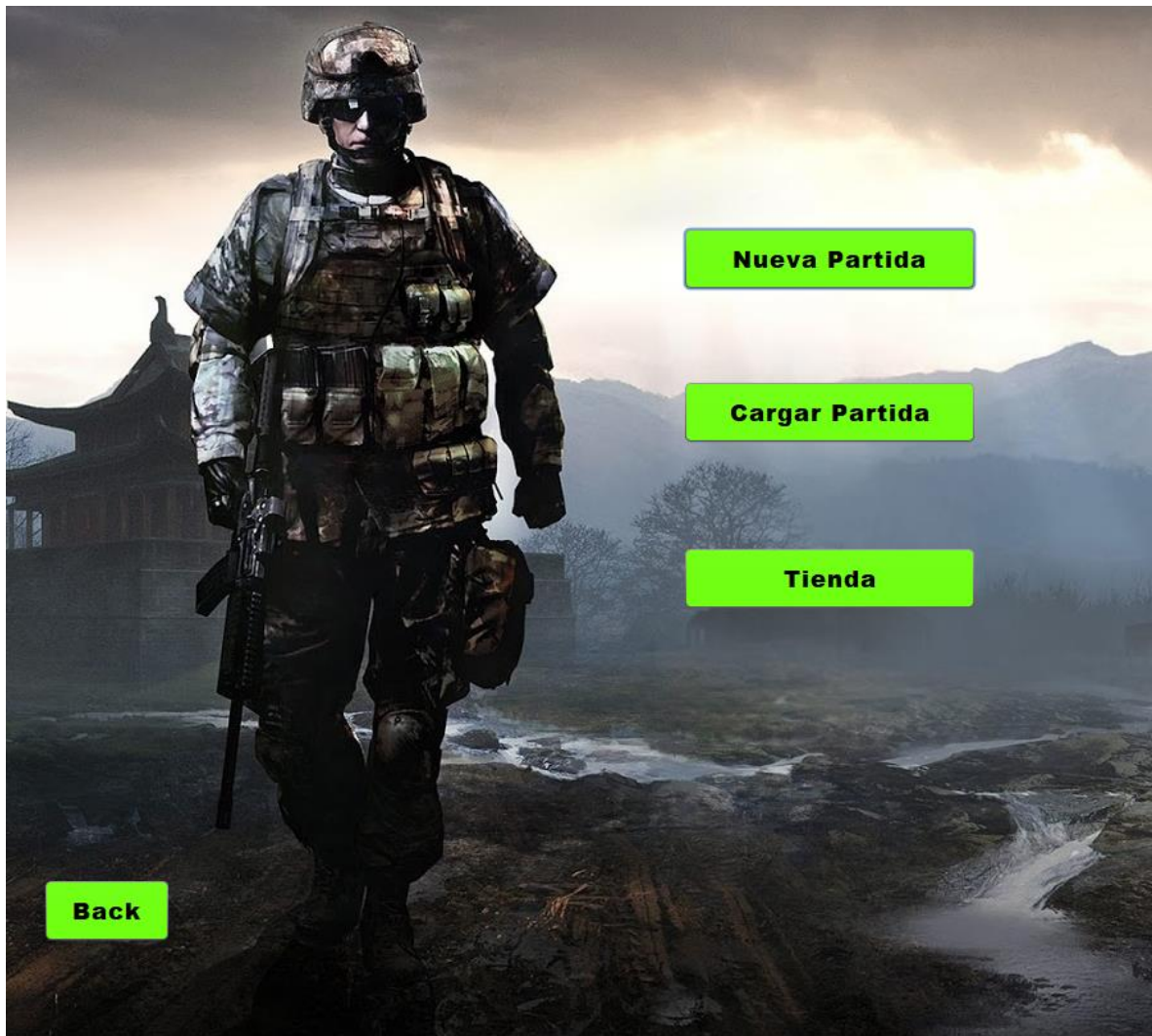
JavaDoc

file:///C:/Users/luedu/Documents/NetBeansProjects/P_Final/dist/javadoc/index.html

Ver registros

- Listado de vehículos: Por cada vehículo desplegar en pantalla el nombre, imagen y el estado (activo o destruido). Incluir también la cantidad de enemigos destruidos por el vehículo y la cantidad de veces que el vehículo ha sido destruido. El listado debe poder ordenarse de forma ascendente o descendente basado en la cantidad de enemigos destruidos por cada vehículo.
- Reporte de batallas: Muestra la cantidad total de batallas, indicando la cantidad de batallas perdidas y ganadas.
- Reporte de mejor vehículo: Muestra el vehículo que más enemigos a destruido, mostrar también el listado de todos los enemigos, incluyendo el escenario donde se destruyó al enemigo.
- Reporte de peor vehículo: Muestra el vehículo que más se ha destruido por enemigos, mostrar también el listado de todos los enemigos, incluyendo el escenario donde se ha destruido al vehículo.

Juego



Glosario (tecnicismos)

Java.Util: Clases de utilidades para el lenguaje, como acceso a recursos del sistema.

Import: Para importar clases de un paquete o una librería se usa el comando import.

Public static void: Clase estática para usarla durante el código (lo que lee primero).

Boolean: Boolean es un tipo de variable que sólo tiene dos valores posibles: "true" (verdadero, 1) y "false" (falso, 0). (la cual es en base a una pregunta).

JPanel: Los JPanel en Java son objetos contenedores, la finalidad de estos objetos es la agrupación de otros objetos tales como botones, campos de texto, etiquetas, selectores, etc;

JFrame: JFrame es una clase utilizada en Swing para generar ventanas sobre las cuales añadir distintos objetos con los que podrá interactuar o no el usuario.

JLabel: Un Label o etiqueta puede mostrar texto plano, una imagen o una imagen con un texto.

TextField: Un JTextField o campo de texto es un componente utilizado para la captura de datos, estos son casi indispensables en una aplicación.

Button: En informática un botón es una metáfora común, utilizada en interfaces gráficas con objetivo similar al de un botón corriente. Los botones suelen ser representados como rectángulos con una leyenda o icono dentro, generalmente con efecto de relieve.

Int: para declarar una variable y luego darle valor (se asigna la letra y un entero).

Break: Se usa para salir rápidamente de un ciclo. Mientras se ejecuta una iteración de un ciclo, si se llega a ejecutar la instrucción break; se abandona de inmediato esa iteración y se sale del ciclo.

Scanner: buscar lo que devuelve el usuario.

Reporte de Programa

El objetivo final del juego RPG con GUI es la utilización de herencias y polimorfismo mediante las cuales se pretende hacer superclases que permitan utilizar sus atributos en clases generadas a partir de una clase primaria, además de la utilización de la Graphical User Interface (GUI) la cual permite el entorno perfecto entre el desarrollador y el usuario. El juego deberá heredar datos de una Persona la cual se convertirá en un jugador a la hora de utilizar el programa, también se utiliza de la clase vehículos especificaciones generales para los vehículos del juego (generado por el programa o por el usuario). Se posee una tienda al iniciar un juego en la cual se comprarán boots e Items de ser necesarios para su posterior uso durante el combate. De esta manera poder tener un conocimiento más exhaustivo de la programación orientada a objetos (POO) y el uso adecuado del GUI debido a que el fin primordial de todo programa es el facilitar las rutinas diarias de las personas.

El juego pondrá a prueba la capacidad de generar interfaces amigables para que el uso y la jugabilidad sean lo más eficiente posible.