



USAC

TRICENTENARIA

Universidad de San Carlos de Guatemala

MANUAL TECNICO:

Proyecto: Practica No. 1

INTRODUCCION A LA PROGRAMACION Y COMPUTACION 1 SECCION "A"

Universidad De San Carlos De Guatemala
Centro Universitario De Occidente
División De Ciencias De La Ingeniería

The background of the page is decorated with large, flowing, pinkish-red swirls that originate from the top left and bottom right corners, framing the central text.

JUEGO RPG

Manual Técnico

Astrid Gabriela Martínez Castillo

Carne: 201731318

INDICE

1. INTRODUCCION	1
2. OBJETIVOS	2
2.1 Objetivos Específicos	2
2.2 Objetivos Generales	2
3. ALCANCE	2
4. REQUERIMIENTOS TECNICOS	2
4.1 Requerimientos Mínimos de Hardware	2
4.2 Requerimientos Mínimos de Software	2
5. HERRAMIENTAS PARA EL DESARROLLO	3
6. ALGORITMOS	3-17
7. DIAGRAMAS DE CLASES	18-27

INTRODUCCION

Este documento describe cada una de las herramientas que se utilizaron para el desarrollo de un juego RPG en consola, tanto aspectos relacionados con el hardware como el software esperando sirva de referencia para especificar la creación de la práctica, así como también los algoritmos y diagramas de flujo de cada uno de los programas y del programa en general.

2. OBJETIVOS

2.1. Objetivos Específicos

- Construcción de algoritmos para los requerimientos de la actividad.
- Ampliar el conocimiento de estructuras del control de JAVA.
- Ampliar el conocimiento de procedimientos en JAVA.
- Implementación de ciclos, sentencias de control y vectores.
- Construcción de aplicaciones simples en consola.

2.2. Objetivos Generales

- Familiarizar al estudiante con el lenguaje JAVA.
- Aplicar conceptos recibidos en clase magistral y laboratorio.
- Elaborar la lógica para la solución del problema planteado.

3. ALCANCE

Este documento está dirigido a: programadores.
Conocimientos básicos en: programación.

4. REQUERIMIENTOS TECNICOS

➤ Software

- Máquina virtual de JAVA (tenerlo instalado).

➤ Hardware

- Una computadora con sus respectivos accesorios (mouse y teclado).

4.1. Requerimientos Mínimos de Software y Hardware

- **Windows:** Windows 10 (8u51 y superiores) Windows 8.x (escritorio)
Windows 7 SP1 Windows Vista SP2 Windows Server 2008 R2 SP1 (64 bits)
Windows Server 2012 y 2012 R2 (64 bits).
- **Memoria RAM:** 128 MB.
- **Espacio en disco:** 124 MB para JRE; 2 MB para Java Update.
- **Procesador Mínimo:** Pentium 2 a 266 MHz
- **Linux**
Oracle Linux 5.5+1
Oracle Linux 6.x (32 bits), 6.x (64 bits)2
Oracle Linux 7.x (64 bits)2 (8u20 y superiores)
Ubuntu Linux 12.04 LTS, 13.x

Ubuntu Linux 14.x (8u25 y superiores)
Ubuntu Linux 15.04 (8u45 y superiores)
Ubuntu Linux 15.10 (8u65 y superiores)
Exploradores: Firefox

5. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO

- Java JDK. 8
- Netbeans 8.2
- Linux Ubuntu 18.04

6. CONFIGURACION

No hay una configuración general la aplicación en si viene configurada y los comandos son intuitivos para el usuario o programador que quiera manejar el sistema.

ALGORITMOS

MENU_PRINCIPAL

var opcion

inicio menuPrincipal

```
escribir "hola + llamar nombre Jugador"
escribir "nivel + llamar nivel Jugador"
escribir "oprime 1 para iniciar Juego"
escribir "oprime 2 para ver tu perfil y tus vehiculos"
escribir "oprime 3 para ir a la tienda"
escribir ""en la tienda puedes utilizar el oro ganado en batallas para adquirir distintos
    artefactos que te ayudaran a ser mas poderoso y a restaurar tus vehiculos que han
    sufrido daños en batalla"
escribir "oprime 4 para crear campo de Batalla"
escribir "oprime 5 para crear tus propias armas"
escribir "oprime 6 para salir del juego"
leer opcion
segun(opcion)
    caso, = 1
        escribir inicio del juego
        llamar a Juego
    Caso, = 2
        escribir "PERFIL DE + llamar JUgador.nombre"
```

```

        mostrar datos de jugDOR
        mostrar datos de Vehiculo
    Caso, = 3
        escribir"BIENVENIDO A TU TIENDA VIRTUAL"
        llamar a tienda
    Caso, = 4
        escribir"CREA TU CAMPO DE BATALLA"
        llamar CrearCampo
    Caso, = 5
        escribir "crea tus propias armas"
        llamar a crear arma
    SiNo
        llamar menuPrincipal;
fin

```

```

inicio regresarAlMenu
    escribir"Desea regresar al menuPrincipal?"
    escribir"s para regresar al menu principal"
    escribir"n para regresar a la tienda"
    leer opcion1
    segun(opcion1)
        caso, = 1
            llamar menuPrincipal
        Caso, = 2
            llamar Tienda
        SiNo
            llamar regresar al menu;
fin

```

*****8

Vehiculo

```

var ataque;
var defensa;
var punteria;
var int HP = (50 * Jugador.nivel);
var PP = (3 + (2 * Jugador.nivel));

```

```

inicio disparar
    var disparo = (aleatorio * 100 + 1)
    si (Arma.municiones > 0)
        escribir"ha acertado"
        si (disparo <= punteria)
            Arma.municiones---
        siNo
            escribir"ha fallado"
    finsi

```

fin

inicio mostrarDatos

 retornar escribir"vehiculo " +" nombre:"+nombre +"\\n"+ mostrarDatos()

fin

nombreA)

 super(7, 3, 7

Tanque

 var nombreT;

 var municiones;

 inicio Tanque(var nombreT)

 super(10, 6, 60,HP,PP);

 nombreT = nombreT;

 fin

 inicio mostrarDatos

 retornar escribir"tanque " +" nombre:"+nombreT +"\\n"+mostrarDatos()

 fin

Avion

 var nombreA;

 inicioAvion(var nombreA)

 super(7, 3, 70,H10P,PP);

 nombreA = nombreA;

 fin

 inicio mostrarDatos

 retornar escribir"Avion " +" nombre:"+nombreA +super.mostrarDatos;

 fin

CrearVehiculo

 Vehiculo[] Vehiculos =new Vehiculo[3];

 var vehiculo;

 var i=0;

 inicio crearVehiculos()

 escribir "puede crear 3 vehiculos");

 mientras (i<=2){

 hacer:

 escribir "-----Que tipo de vehiculo desea Crear?-----"


```

        escribir "1 para crear un Avion"
        escribir "este vehiculo no tiene mucha potencia de disparo \npero tiene excelente
punteria y mucha velocidad de ataque "
        escribir "2 para crear un Tanque"
        escribir "vehiculo pesado con mucha defensa, es capaz de disparar armas con
mucha potencia \n posee defensa alta pero tiene poca punteria "
        escribir "opcion:"
        vehiculo = Integer.valueOf(miScanner.nextLine());
    } mientras (vehiculo!=1 && vehiculo!=2);
    segun(vehiculo){
        case 1://escogio avion
            escribir "\n***CREAR UN AVION***");
            llamar a nombrarAvion();

```

```

        case 2://escogio tanque
            escribir "\n***CREAR UN TANQUE***"
            nombrarTanque();
    }
    fin

```

```

inicio nombrarAvion()
    var nombre;
    escribir "Que nombre desea ponerle a su nuevo avion?";
    leer nombre
    avion = nuevo Avion(nombre);
    //guardamos un avion a nuestro arreglo de Vehiculos
    Vehiculos[i] = avion
    i++
fin

```

```

inicio nombrarTanque()
    var nombre;

    escribir "Que nombre desea ponerle a su nuevo Tanque?";
    leer nombre
    tanque = new Tanque(nombre);
    //guardamos un Tanque dentro de nuestro arreglo de Vehiculos
    Vehiculos[i]= tanque
    i++
fin

```

```

inicio mostrarDatos()

for(int j=0;j<4;j++)
    if(Vehiculos[j] == null){
        System.out.println("desea regresar al menu S/N");
        leer var res
        switch(res){

```

```

        caso,= "s":
            llamar MenuPrincipal.menuPrincipal();

        casp,= "S":
            llamamr MenuPrincipal.menuPrincipal()

        siNo:
            mostrarDatos();

    else{
        escribir "Vehiculos[j].toString()"
    fin

```

Enemigo

```

    var nombre = "\u001B[0m    %%!!%% "
    var muento= "\u001B[0m    X    "
    var ATAQUE = 10;
    var DEFENSA =5;
    var = 25;
    var punteria = (int) (Math.random() * 15 + 1);

```

getters y setters

Arma

```

    var municiones;
    var nombreArma;
    var punteriaArma;
    var velocidadDisparoA;
    var ataqueA;

```

```

    inicio agregarMuniciones()
    fin

```

```

    inicio restarDisparos
    fin

```

Armas hereda de Arma

```

    public class Armas extends Arma
    protected String nombreAr
    protected int municionesN

```

```

    constructor()

```

Cañon

var municionesC

inicio Cañon(var municiones, var punteria, var ataque, var velocidadDisparo, var municiones)
super(nombreArma, 15, 15, 10)
municiones = municionesC
fin

inicio agregarMuniciones(var opcion)

segun(opcion)

caso, = 1

si(Jugador.myOro >= 50)

escribir "se han agregado 15 municiones a tu cañon"

cambiar miOro(Jugador.miOro - 50)

retornar: municionesC = municionesC + 15

siNO

escribir "no tienes dinero suficiente, gana partidas para adquirir mas

dinero"

MenuPrincipal.regresarAlMenu()

retornar: municionesC = municionesC + 0

finSi

caso, = 2

si(Jugador.myOro >= 100)

escribir "se han agregado 30 municiones a tu cañon"

cambiar miOro(Jugador.miOro - 100)

retornar: municionesC = municionesC + 30

siNO

escribir "no tienes dinero suficiente, gana partidas para adquirir mas

dinero"

MenuPrincipal.regresarAlMenu()

retornar: municionesC = municionesC + 0

finSi

siNo

escribir "numeri incorrecto"

retornar: municionesC = municionesC + 0

fin

inicio restarDisparos();

retornar municionesC = (municionesC - 1)

fin

Ametralladora

var municionesA

```

    inicio Ametralladora(var municiones, var punteria, var ataque, var velocidadDisparo, var
municiones)
    super(nombreArma, 8,15,20)
    municiones = municionesA
    fin

```

```

    inicio agregarMuniciones(var opcion)
        segun(opcion)
            caso, = 1
                si(Jugador.myOro>=50)
                    escribir"se hanint velocidadDisparo agregado 15 municiones a tu
Ametralladora"
                    cambiar miOro(JUgador.miOro-50)
                    retornar: municionesA= municionesA +15
                siNO
                    escribir"no tienes dinero suficiente, gana partidas para adquirir mas
dinero"
                    MenuPrincipal.regresarAlMenu()
                    retornar: municionesA= municionesA + 0
                finSi
            caso, = 2
                si(Jugador.myOro>=100)
                    escribir"se han agregado 30 municiones a tu Ametralladora"
                    cambiar miOro(JUgador.miOro-100)
                    retornar: municionesA= municionesA +30
                siNO
                    escribir"no tienes dinero suficiente, gana partidas para adquirir mas
dinero"
                    MenuPrincipal.regresarAlMenu()
                    retornar: municionesA= municionesA + 0
                finSi
            siNo
                escribir"numeri incorrecto"
                retornar: municionesA= municionesA + 0
        fin
    fin

```

```

    inicio restarDisparos();
        retornar municionesA = (municionesA -1)
    fin

```

```

-----
Misil
var municionesM

```

```

    inicio Misil(var municiones, var punteria, var ataque, var velocidadDisparo, var municiones)

```

```
super(nombreArma, 8,15,20)
municiones = municionesM
fin
```

```
inicio agregarMuniciones(var opcion)
```

```
    segun(opcion)
```

```
        caso, = 1
```

```
            si(Jugador.myOro>=50)
```

```
                escribir"se hanint velocidadDisparo agregado 15 municiones a tu Misil"
```

```
                cambiar miOro(JUgador.miOro-50)
```

```
                retornar: municionesM= municioneAM +15
```

```
            siNO
```

```
                escribir"no tienes dinero suficiente, gana partidas para adquirir mas
```

```
dinero"
```

```
                MenuPrincipal.regresarAlMenu()
```

```
                retornar: municionesM= municionesM + 0
```

```
            finSi
```

```
        caso, = 2
```

```
            si(Jugador.myOro>=100)
```

```
                escribir"se han agregado 30 municiones a tu Misil"
```

```
                cambiar miOro(JUgador.miOro-100)
```

```
                retornar: municionesM= municionesM +30
```

```
            siNO
```

```
                escribir"no tienes dinero suficiente, gana partidas para adquirir mas
```

```
dinero"
```

```
                MenuPrincipal.regresarAlMenu()
```

```
                retornar: municionesM= municionesM + 0
```

```
            finSi
```

```
        siNo
```

```
            escribir"numeri incorrecto"
```

```
            retornar: municionesM= municionesM + 0
```

```
fin
```

```
inicio restarDisparos();
```

```
    retornar municionesM = (municionesM -1)
```

```
fin
```

```
-----  
CrearArma
```

```
    var i,j;
```

```
    var punteriaArma;
```

```

Armas[] armas1 = nueva Armas[5];
var PRECIOA = 100;
var ataqueA;
var opcion

inicio crearArma
  hacer{
    escribir puede crear armas1.tamaño-i Armas
    llamar a nuevaArma()
    escribir "su nueva arma ha sido creada y se le han iniciado sus atributos"
    i++;
    si(i=Armas.tamaño)
      break
    siNo
      escribir"oprima s para seguir agregandp armas a la TIENDA "
      leer opcion;
  mientras("s" = opcion o "S"=opcion)
    escribir "TU ARCENAL ESTA LLENO"
    llamar menuPrincipal()

fin

inicio nuevaArma()
  var nombreAr
  escribir"ingrese el nombre de su nueva Arma"
  leer nombreAr
  punteriaArma = numero ALEATORIO
  ataqueA = numero ALEATORIO
  armaArmas1[j].precioAs1[i] = nueva
Armas(nombreAr,punttamañoeriaArma,5,ataqueA,precioA)
fin

inicio mostrarArmas()
  var numeroArma=3
  para(j=0;j<Armas1.tamaño;j++)
    si (armas[j] == null)
      escribir"regresar al menu S"
      escribir"oprima cualquier letra para ver el arcenal"
      leer redultado
      segun(resultado)
        caso "s":
          MenuPrincipal()
        caso "S":
          MenuPrincipal()
      siNo
        mostrarArmas()
    siNo si(j==Armas1.tamaño)

```

```

        escribir "desea regresar al arcenal oprima s"
        escribir "oprima cualquier letra para ver el arcenal"
    leer resultado
    segun(resultado)
        caso "s":
            MenuPrincipal()
        caso "S":
            MenuPrincipal()
        siNo
            mostrarArmas()
    siNo
        escribir "# numeroArma Armas1[j].datos)"
    numeroArma++
fin

```

Arcenal

```

    Arma[] miArcenal = new Arma[8]
    var i=0
    Misil MISIL = nuevo Misil()
    private static final Ametralladora AMETRALLADORA = new Ametralladora();

    public static void comprarArma(){
        escribir " Puede escoger entre las armas existentes y las que ha creado"
        escribir "ARMAS EXISTENTES:"
        escribir "#1 "+MISIL.datos PRECIO Q.MISIL.PrecioA"
        System.out.println("#2 AMETRALLADORA.datos PRECIO Q.
    AMETRALLADORA.getPrecioA"
        escribir "LAS ARMAS CREADAS POR TI SON: "
        mostrarArmas();
        obtenerPrecio();
        escribir "opreme el numero de arma que deseas comprar:"
        var opcion = Integer.valueOf(scaner1.nextLine());
        segun(opcion){
            caso 1:
                si(Jugador.Oro < MISIL.PrecioA)
                    oroInsuficiente(); //metodo que nos avisa si puede comprar el arma
                siNo
                    escribir "se ha agregado a su arcenal personal el arma Misil"
                    miArcenal[i]=MISIL;
                    Jugador.Oro= Jugador.oro-MISIL.PrecioA
                    i++

            case 2:
                si(Jugador.Oro<AMETRALLADORA.PrecioA)

```

```

        oroInsuficiente(); //metodo que nos avisa si puede comprar el arma
    siNo{
        escribir "se ha agregado a su arcenal personal el arma Ametralladora"
        miArcenal[i]=AMETRALLADORA;
        Jugador.Oro=Jugador.Oro-AMETRALLADORA.PrecioA
        i++

case 3:
    si(Armas1[0]!=null){
        si(Jugador.Oro<CrearArma.precioA)
            oroInsuficiente(); //metodo que nos avisa si puede comprar el arma
        siNo{
            escribir "se ha agregado a su arcenal personal el arma"+
CrearArma.getArmas1()[0].getNombreArma()
            miArcenal[i]= CrearArma.getArmas1()[0];
            Jugador.Oro(Jugador.MyOro-CrearArma.precioA
            i++

        siNo
            escribir "no has creado esta arma"
            comprarArma();}

caso 4:
    si(CrearArma.getArmas1()[1]!=null){
        si(Jugador.getMyOro()<CrearArma.precioA){
            oroInsuficiente(); //metodo que nos avisa si puede comprar el arma
        siNo{
            eswitchescribir "se ha agregado a su arcenal personal el arma"+
CrearArma.rmas1[1].getNombreArma)
            miArcenal[i]= CrearArma.Armas1[1];
            Jugador.MyOro=Jugador.MyOro()-CrearArma.precioA
            i++

        siNo
            escribir "no has creado esta arma"
            comprarArma();}

caso 5:
    si(CrearArma.Armas1[2]!=null)
        si(Jugador.getMyOro()<CrearArma.precioA)
            oroInsuficiente(); //metodo que nos avisa si puede comprar el arma
        siNo
            escribir "se ha agregado a su arcenal personal el arma"+
CrearArma.Armas1[2].NombreArma"
            miArcenal[i]= CrearArma.Armas1[2]
            Jugador.MyOro=Jugador.MyOro()-CrearArma.precioA
            i++

        siNo

```



```

        escribir "no has creado esta arma")
        comprarArma();
    caso 6:
        si(CrearArma.Armas1[3]!=null)
            si(Jugador.Oro()<CrearArma.precioA
                oroInsuficiente(); //metodo que nos avisa si puede comprar el arma
            siNo
                escribir "se ha agregado a su arcenal personal el arma"+
                CrearArma.getArmas1()[3].NombreArma());
                miArcenal[i]= CrearArma.getArmas1()[3]
                Jugador.sMyOro=Jugador.getMyOro()-CrearArma.precioA
                i++

```

```

        siNo
            escribir "no has creado esta arma");
            comprarArma()

```

```

    case 7:
        si(CrearArma.Armas1[4]!=null){
            si(Jugador.Oro()<CrearArma.precioA)
                oroInsuficiente(); //metodo que nos avisa si puede comprar el arma
            siNo
                System.out.println("se ha agregado a su arcenal personal el arma"+
                CrearArma.getArmas1()[4].getNombreArma())
                miArcenal[i]= CrearArma.Armas[4]
                Jugador.setMyOro(Jugador.getMyOro()-CrearArma.precioA);
                i++

```

```

        siNO
            escribir "no has creado esta arma"
            llamar comprarArma()

```

```

    case 8;
        escribir"ha decidido regresar a la tienda"
        llamar Tienda()
    siNo;
        escribir"numero incorrecto intente de Nuevo"
        llamar comprarArma()

```

```

    fin

```

```

    inicio miArcenal()
        escribir"ESTAS SON LAS ARMAS QUE TIENES DOSPONIBLES PARA TUS
        VEHICULOS"
        para(Arma miArcenal; Arcenla.miArcenal)
            si()
                escribir"miArcenal[]"

```

```

        siNo
            escribir "no has agregado armas aun"
    fin

```

Tienda

```

    inicio tienda
        var compra;
        var miOpcion= "s";
    mientras(miOpcion.equals("s")){
        escribir "\n Tu oro disponible es: "+ Jugador.getMyOro()+ " monedas"
        escribir "\n selecciona el producto que desees adquirir"
        escribir "1.*****KITS:*****"
        escribir ""
        escribir "2.*****MUNICIONES:*****"
        escribir ""
        escribir "3.*****ARCENAL*****"
        escribir ""
        escribir "4. REGRESAR AL MENU PRINCIPAL"
        escribir "\nQUE PRODUCTO DESEAS ADQUIRIR?"
        leer compra

        segun(compra)
        caso 1:
            escribir "COMPRA UN KIT PARA MEJORAR TU VEHICULO "
            ComprarKit.comprarKits()
        caso 2:
            llamr comprarMuniciones()
        caso 3:
            escribir "TU ARCENAL"
        caso 4:
            llamar menuPrincipal()
        siNo:
            escribir "numero incorrecto"
            llamr tienda()
    fin

```

ComprarMuniciones

```

    var MUNICIONES_MENOR= 50;
    var MUNICIONES_MAYOR= 100;
    var opcion;

```

```

inicio comprarMuniciones()

escribir "Tu oro disponible es: "+ Jugador.getMyOro()+ " monedas"
escribir "ADQUIERE TUS MUNICIONES"
escribir "1.- municiones para CAÑON"
escribir "2.- municiones para Metralleta"
escribir "3.- municiones para MISIL"
leer municion
    segun (municion)
        caso 1:
            hacer
                escribir "MUNICIONES CAÑON"
                escribir "1.- paquete con 15 municiones para CAÑON precio: "+
MUNICIONES_MENOR+ " monedas"
                escribir "2.- paquete con 30 municiones para CAÑON precio: "+
MUNICIONES_MAYOR+ " monedas"
                leer opcion
                llamar agregarMuniciones()
            mientras (opcion !=1&& opcion!=2)
                caso 2:
                    hacer
                        escribir "MUNICIONES AMETRALLADORA"
                        escribir "1.- paquete con 15 municiones para AMETRALLADORA precio: "+
MUNICIONES_MENOR+ " monedas"
                        escribir "2.- paquete con 30 municiones para AMETRALLADORA precio: "+
MUNICIONES_MAYOR+ " monedas\n"
                        leer opcion
                        llamar agregarMuniciones()
                    mientras(opcion !=1&& opcion!=2)
                        caso 3:
                            hacer
                                escribir "MUNICIONES MISIL"
                                escribir "1.- paquete con 15 municiones para ametralladora precio: "+
MUNICIONES_MENOR+ " monedas"
                                escribir "2.- paquete con 30 municiones para CAÑON precio: "+
MUNICIONES_MAYOR+ " monedas"
                                leer opcio
                                llamar agregarMuniciones()
                            mientras(opcion !=1 && opcion!=2)
                                siNo:
                                    escribir "la opcion ingresada es invalida intente de nuevo\n"
                                    llamar comprarMuniciones()

```

Kit <<Abstracta>>

var nombre

```
restaurarAuto()
```

KitMayor hereda Kit

```
var vida =50
```

```
restaurarAuto()  
Vehiculo miVehiculo= nuevo Vehiculo();  
miVehiculo.cambiarHP(miVehiculo.obtenerHP()+vida);
```

KitMenor

```
var vida = 25
```

```
restaurarAuto()  
Vehiculo miVehiculo= nuevo Vehiculo();  
miVehiculo.cambiarHP(miVehiculo.obtenerHP()+vida);
```

```
comprarKit
```

```
var PRECIO_KMENOR = 40  
var PRECIO_KMAYOR=60  
var PRECIO_ARMA=200  
var k; //contador  
var tamaño=0  
Kit[] kits = new Kit[tamaño]
```

```
gatters y setters
```

```
comprarKits()  
escribir "1.- para comprar kit MENOR    precio: "+ PRECIO_KMENOR + " monedas")  
escribir "    este kit repara tu vehiculo 25 puntos de vida")  
escribir "2.- para comprar kit MAYOR    precio: "+ PRECIO_KMAYOR + " monedas ")  
escribir "    este kit repara tu vehiculo 50 puntos de vida")  
leer opcion  
segun (opcion){  
    caso 1:  
        si (Jugador.getMyOro())>= PRECIO_KMENOR)  
            tamaño ++  
            System.out.println("se ha agregado un Kit menor"  
            KitMenor kitMenor = nuevo KitMenor("Kit Menor"  
            kits[k]= kitMenor  
            k++
```

```

        Jugador.myOro= (Jugador.myOro - PRECIO_KMENOR
siNo
        System.out.println("no tienes dinero suficiente "
        MenuPrincipal.regresarAlMenu());

```

caso 2:

```

        si (Jugador.getMyOro())>= PRECIO_KMAYOR)
            tamaño ++
            System.out.println("se ha agregado un kit mayor"
            Jugador.myOro= (Jugador.myOro - PRECIO_KMAYOR)
            KitMayor kitMayor = new KitMayor("Kit Mayor"
            kits[k]= kitMayor
            k++
        siNo
            System.out.println("no tienes dinero suficiente "
            MenuPrincipal.regresarAlMenu()
    siNo:
        llamar comprarKits()

```

fin

```

mostrarMisKits(){
for(var j=0;j<tamaño;j++){
    if(kits[j] == null){
        System.out.println("desea regresar al menu S/N")
        String res= scanner.nextLine();
        segun (res){
            caso, "s":
                MenuPrincipal.menuPrincipal()
                break;
            caso, "S":
                MenuPrincipal.menuPrincipal()
                break;
            siNo
                mostrarMisKits()
            else
                System.out.println(kits[j].mostrar)
        }
    }
}

```

fin

Elemento

```

var ANSI_RESET = "\u001B[0m      " // resetear fondos
var vida=0
var cont = 0
var elemento

```

getter y setters

Agua hereda Elemento

```
var ANSI_AGUA = "\u001B[46;30m ~~~~AGUA~~~~"; //color celeste
```

getters y setters

TerrenoNormal

```
var ANSI_TNORMAL = "\u001B[42;30m Terreno Normal" // color verde
```

getters y setters

Montañas

```
protected String ANSI_MONT = "\u001B[43;30m ...Montañas..."; //color amarillo
```

getters y setters

CampoDeJuego

```
var enemigos;  
var elemento;  
var nombre;  
Elemento campo [][];  
var int miCampo;
```

constructor()

```
escribir()  
retornar "CampoDeJuego{" + " nombre=" + nombre)
```

inicio listadoCampos

```
escribir "ANTES DE INICIAIR DEBES ESCOGER TU CAMPO DE BATALLA"  
escribir "1. Campos Preestablecidos"  
escribir "2. campos CREADOS"
```

```

leer miCampo
segun(miCampo)
    caso 1:
        escribir "escoge un campo preestablecido"

    caso 2:
        escribir "escoge uno de los campos que has creado"

    siNo:
        escribir "has ingresado una opcion incorrecta!!"
fin

```

Montañoso hereda de CampoDeJuego

```

var Elemento[][] montañas
var static int i
var static int j
var terreno

```

getter y setters

inicio Crearmontañas

```

montañas = new Elemento [8][8];
Elemento uno= new Elemento();
int terc = uno.getCont();
int sec = uno.getCont();

```

```

para(i=0;i<montañas.length;i++)
    para(j=0;j<montañas.length;j++)
        si(i==0 && j%2==0 && j != montañas.tamaño -1)
            montañas[i][j]= nuevo; //enemigos
        siNo si (i==5 && j%2!=0 && i!=7&&j!=montañas.tamaño -1)
            montañas[i][j]= nuevo;
        siNo si (i==5 && j%2==0 && i!=7 && j!=montañas.tamaño -1)
            Elemento nuevo = new Elemento(terreno)
            montañas[i][j]= nuevo;}
        siNo si (i==6 && i!=7&&j!=montañas.tamaño -1 )
            Elemento nuevo = new Elemento(terreno)
            montañas[i][j]= nuevo;
            siNo si (i!=7&&j!=montañas.tamaño -1)
                Elemento nuevo = new Elemento(terreno)
                montañas[i][j]= nuevo;
        siNo si (i==montañas.tamaño -1){           //enumerar filas
            Elemento numero = new Elemento(unos.getANSI_RESET()+sec)
            montañas[i][j]= (numero);
            sec = sec +1;
        siNo si (j==montañas.tamaño -1){           //enumerar columnas

```

```

        Elemento numero = new Elemento(unos.getANSI_RESET()+terc )
        montañas[i][j]= (numero );
        terc = terc +1;
    fin

    inicio mostrarMontañoso
    new Montañoso("Montañoso",montañas);
    Crearmontañas();

    escribir myMontañoso.Nombre"

    para (Elemento[] montaña1 : montañas)
        para (Elemento montaña2 : montaña1)
            escribir" montaña2.datos"
    fin

```

 Playa hereda de CampoDeJuegotamaño

```

    Elemento[][] playa;
    vatr i;
    var j;
    var terreno;

    inicio crearPlaya(){
    playa = new Elemento [8][8]
    Elemento uno= nuevo Elemento
    vat terc = uno.getCont
    var sec = uno.Cont

    para(i=0;i<playa.length;i++){
        para(j=0;j<playa.length;j++){
            ai(i==0 && j%2==0 && j != playa.tamaño -1)
                Elemento nuevo = new Elemento(terreno)
                playa[i][j]= nuevo; //enemigos
            siNo si (i==5 && j%2!=0 && i!=7&&j!=playa.tamaño -1)
                Elemento nuevo = new Elemento(terreno)
                playa[i][j]= nuevo;
            siNo si (i==5 && j%2==0 && i!=7 && j!=playa.tamaño -1)
                Elemento nuevo = new Elemento(terreno)
                playa[i][j]= nuevo;}
            siNo si (i==6 && i!=7&&j!=playa.tamaño -1 )
                Elemento nuevo = new Elemento(terreno)
                playa[i][j]= nuevo;
            siNo si (i!=7&&j!=playa.tamaño -1)
                Elemento nuevo = new Elemento(terreno)
                playa[i][j]= nuevo;
        }
    }

```



```

        siNo si (i==playa.tamaño -1){    //enumerar filas
            Elemento numero = new Elemento(ANSI_RESET()+sec);
            playa[i][j]= (numero);
            sec = sec +1;tamaño
        siNo si (j==playa.tamaño -1){    //enumerar columnas
            Elemento numero = new Elemento(unos.getANSI_RESET()+terc );
            playa[i][j]= (numero );
            terc = terc +1;
        fin
        inicio mostrarMontañoso
        nueva Playa("Playa",playa);
        CrearPlaya();

        escribir myMontañoso.Nombre"

        para (Elemento[] playa1 : Playa)
            para (Elemento playa2 : playa1)
                escribir" playa2.datos"
        fin

```

 Jungla hereda CampoDeJuego

```

        Elemento [][] jungla ;
        var enemy;
        var i;
        var j;
        var terreno;

```

INicio crearJungla

```

jungla = nuevo Elemento [8][8];
Elemento uno= nuevo Elemento
int terc = Cont
int sec = Cont

```

```

for(i=0;i<jungla.tamaño;i++){
    for(j=0;j<jungla.tamaño;j++){
        if(i==0 && j%2!=0 && j != jungla.tamaño)
            terreno = miEnemigo.getNombre()
            Elemento nuevo = new Elemento(terreno)
            jungla[i][j]= nuevo
        siNo si (i%2!=0&& j%2==0&&i!=7&&j!=jungla.tamaño)
            terreno = terrenos.ANSI_MONT
            Elemento nuevo = new Elemento(terreno)
            jungla[i][j]= nuevo
        siNo si (i%2==0&&i!=7&&j!=jungla.tamaño)
            terreno = terrenos.ANSI_TNORMAL
    }
}

```

```

        Elemento nuevo = new Elemento(terreno);
        jungla[i][j]= nuevo;
    siNo si (i!=7&&j!=7)
        terreno = terrenos.ANSI_AGUA;
        Elemento nuevo = new Elemento(terreno);
        jungla[i][j]= nuevo;
    siNo si (i==7){
        Elemento numero = new Elemento(ANSI_RESET()+sec);
        jungla[i][j]= (numero);
        sec = sec +1;
    siNo si (j==7){
        Elemento numero = new Elemento(ANSI_RESET()+terc );
        jungla[i][j]= (numero );
        terc = terc +1;
        finsi
    fin

```

```

mostrarJungla
Jungla miJungla = nueva Jungla("Jungla", jungla);
crearJungla();
escribir "miJungla.Nombre"

```

```

para (Elemento[] jungla1 : jungla)
    for (Elemento jungla2: jungla1)
        escribir "jungla2.datos"

```

CrearCampo

```

var int contador
var nombre
var eleccion
var camp
var c=0
var i
var j
var k
var terreno
var numeroFilas
var numeroColumnas
var enemigos = 3 //numero de enemigos que se ppueden agregar
Elemento [][]campo;
CampoDeJuego[] campos = new CampoDeJuego[5]

```

```

inicio pedirTerreno
    escribir " para Agua(facilidad de destruccion media)"

```

```

escribir "2 para terreno Normal(facilidad de destruccion baja)"
escribir "3 para Montañas (faciliudad de destruccion alta)"
escribir "4 para agregar un enemigo a la cuadrícula"
leer var opcion
    segun(opcion)
    caso 1:
        Agua terrenos = nueva Agua()
        escribir "se ha agregado agua a la cuadrícula"
        terreno = terrenos.ANSI_AGUA
    caso 2:
        TerrenoNormal normal = new TerrenoNormal();
        escribir "se ha agregado terreno normal a lacuadrícula"
        terreno = normal.ANSI_TNORMAL
    caso 3:
        Montaña mont = new Montaña()
        escribir "\nse ha agregado Montañas a la cuadrícula"
        terreno=mont.ANSI_MONT;

    caso 4:
        si(enemigos==0)
            escribir "no puedes agregar mas enemigos"
        siNo si
            escribir "ha agregado un enemigo le quedan:"+(enemigos-1)+"por agregar"
            Enemigo miEnemigo = nuevo Enemigo()
            terreno=miEnemigo.getNombre()
            enemigos = (enemigos-1)
        siNo
            escribir "elemento de Terreno invalido"
            llamar a pedirTerreno()
fin

```

Inicio crearTerreno

```

Scanner Nuevo =nuevo Scanner(System.in)
Elemento uno= nuevo Elemento()
int terc = Cont()
int sec = Cont()

```

```

escribir "ingrese numero de filas de obstaculos entre ud y su oponente V"
leer numeroFilas +1
escribir "ingrese la amplitud del terreno que desea ----->"
leer numeroColumnas +1
campo = Elemento[numeroFilas][numeroColumnas];
para ( i=0;i<campo.tamaño;i++){

```

```

        para( j=0;j<campo.length;j++)
para(i==campo.length-1)
    Elemento numero = nuevo Elemento(ANSI_RESET+sec)
    campo[i][j]= (numero)
    sec = sec +1
siNo si (j==campo.tamaño-1)
    Elemento numero = new Elemento(unos.getANSI_RESET()+terc )
    campo[i][j]= (numero )
    terc = terc +1
else {
    escribir "\nque tipo de terreno tendra su cuadrícula? "
    escribir "posicion de campo de juego ["+i+"]["+j+"]\n"
    llamar pedirTerreno();
    Elemento nuevo = nuevo Elemento(terreno);
    campo[i][j]= nuevo;
}
fin

```

inicio agregarCampo

```

hacer
    escribir"Que nombre desea ponerle a su campo?"
    nombre = Nuevo.nextLine();
    crearTerreno();
    CampoDeJuego miCampo = new CampoDeJuego(nombre,campo)
    campos[contador] = miCampo;
    escribir"su nuevo Campo de Juego se ha creado "
    contador++;
    if(i == campos.length)
        fin
siNo si
    escribir"Oprima s para seguir Creando Terrenos"
    leer eleccion
mientras("S".equals(eleccion) || "s".equals(eleccion));
    escribir "\nTU ESPACIO PARA TERRENOS ESTA LLENO \n"
    llamar menuPrincipal
fin

```

inicio listarCampoCreado

```

    escribir "Tus campos creados son:"
    int s=0;

    mientras(campos!= null && s<5)
        escribir "(s+1)+campos[s]nombre "
        s++
    finmientras

```

```

    escribir "ingrese el numero de campo que desea");
    leer camp
    si(campo[camp]==null)
        escribir "ha introducido un numero incorrecto");
        listarCampoCreado();
    siNo
        campo[camp]
        retornar camp

```

Jugador

```

    var nivel = 1;
    var myOro = 100;
    var nombre;
    var experiencia =0;
    var necesario = 50; //experiencia necesaria para subir de nivel

```

inicio pedirNombre

```

    System.out.println("Cual es tu nombre Jugador?");
    nombre= miScanner.nextLine()

```

fin

inicio mostrarDatos

```

    escribir "Hola " + Jugador.nombre + " estos son tus Datos:\n");
    escribir "tu oro " + Oro+ " monedas"+" tu nivel " + Nivel+ " tu experiencia
    "+
        Experiencia+
    escribir "TUS VEHICULOS"
    llamar mostrarDatos

```

```

    escribir " Tus ARMAS: "
    llamar miArcenal();
    escribir " Tus KITS de REPARACION comprados "
    ComprarKit.mostrarMisKits
    llamr regresarAlMenu

```

fin

inicio subirNivel

```

    if (necesario ==experiencia){
        nivel= nivel+1;
        return necesario = (necesario*(3/2))
    siNo
        return necesario = necesario +0;

```

fin

DIAGRAMAS DE CLASE:

Diagrama de Clases RPG en consola

