

# Proyecto programación 2º Trimestre.

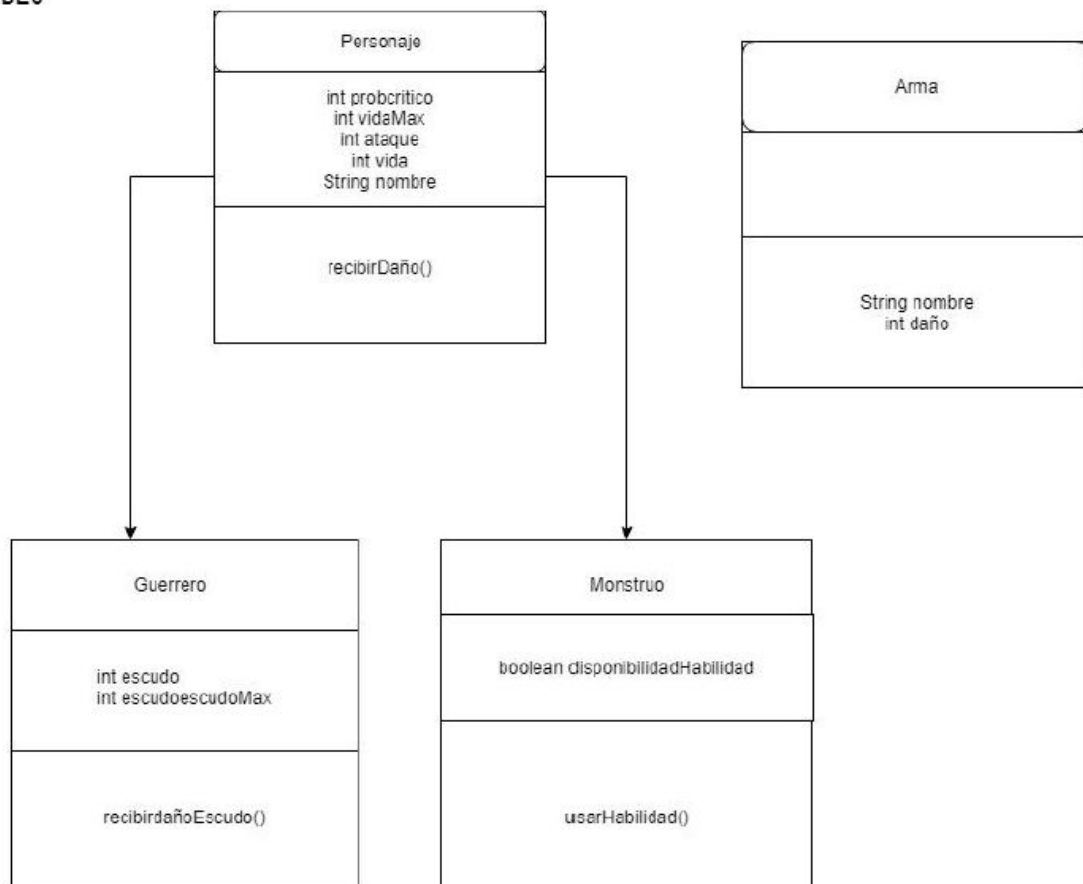
## POO avanzada

## Aspectos generales del programa

El programa es un juego basado en una lucha de un guerrero contra un monstruo, al principio del programa se lanzarán dos dados, quien gane podrá decidir si ser el guerrero o el monstruo, cada uno podrá elegir un arma y tendrá unas características distintas. El guerrero tendrá un escudo que le hará más resistente y el monstruo una habilidad especial que potenciará sus ataques. Una vez esté hecho eso comenzará el combate. Por turnos (Que comenzará el guerrero por defecto) se elegirán las acciones a realizar hasta que a uno de los dos le llegue la vida a 0.

## Diagrama UML del proyecto

### ENTIDADES



GESTORAS

GestoraDado
lanzardado() gestoraDados() paint() lanzar()

GestoraEntradaTeclado
introducirNombre()

GestoraGuerrero
int escudoMax int proCritico int vidaMax
Guerrero()

GestoraMenu
menu()

GestoraMonstruo
generarMonstruo()

GestoraPeleas
Pelear()

UTILIDADES

Utilidad
esperar() esperarFin() finalCombate()

DATOS

AlmacenArmas
ArrayList armas

MAIN

CombateMain
Clase Main

## Descripción de paquetes y clases

Paquete **Datos**: La clase AlmacénArmas es la encargada de almacenar el ArrayList de armas para que los jugadores puedan elegir una u otra.

Paquete **Entidades**: En este paquete encontramos la clase Personaje, que es la clase padre de Guerrero y Monstruo, ambas clases son las clases principales del programa puesto que son los objetos que “luchan” entre ellos. Y la clase Armas que es una clase destinada a crear armas y almacenarlas en la clase AlmacénArmas anteriormente descrita.

Paquete **Gestoras**: En este paquete encontramos la clase GestoraDado que es la encargada de, mediante una interfaz gráfica, adjudicar a los jugadores si pueden elegir personaje o no, la clase GestoraGuerrero y GestoraMonstruo son las encargadas de generar un monstruo con unos valores dados anteriormente en la misma clase, la GestoraMenu es la encargada de almacenar el menú principal del juego y los menús de lucha de ambos jugadores. En la GestoraEntradaTeclado tenemos un método que nos permite validar los nombres introducidos por los jugadores. Y por último GestoraPeleas que es la clase más importante del programa puesto que es la encargada de llevar el *thread* del combate entre los dos jugadores.

Paquete **Utilidades**: En este paquete encontramos la clase Utilidades, en esta encontramos tres métodos. Dos están destinados a parar el *thread* del programa durante unos milisegundos para dar mayor sensación de estar en videojuego y el otro método es el encargado de mostrar el mensaje *Fin del combate* cuando a uno de los dos jugadores le llega la vida a 0.

Clase **CombateMain**: El main del programa en el que tenemos un menú controlado por una estructura de control *do-while* que nos permite seguir el hilo principal del programa las veces que queramos y realizar las acciones que queramos en todo momento, ya sea lanzar el dado o pelear una y otra vez, siempre con la opción de salir del programa.

## Reparto de tareas

Andrés: Clase Armas y diagrama UML del proyecto.

Diana: Clase AlmacenArmas, Clase Monstruo, Clase Personaje, Clase GestoraEntradaTeclado, Clase GestoraMonstruo.

Antonio: Su propia documentación

Miguel: Clase Personaje, Clase GestoraDado, GestoraGuerrero, GestoraPeleas, CombateMain, Clase Utilidades, documentación interna del proyecto y supervisión y corrección del proyecto completo. Diseño de clases y paquetes y de lógica del programa  
+ lo que hablé contigo