**ESCUELA COLOMBIANA DE INGENIERÍA**

**PROGRAMACIÓN ORIENTADA A OBJETOS**

**Parcial tercer tercio.**

**S17: 2020-1**

Nikolás Bernal Giraldo

Nota Esperada: 2.8

1. REFACTORIZANDO: BBD-MDD XP

Seleccione un comportamiento del proyecto que requiera refactorización.

1. Explique su selección, detallando las ventajas y desventajas de la implementación actual.

Colocar los elementos del juego como una herencia, ya que con esto podriamos implementar polimorfismo, y ahorrarnos bastantes lineas de codigo, ya que los elementos comparten algunas caracteristicas.

1. Incluya en el documento los componentes BDD-MDD del comportamiento seleccionado: requisitos, diseño (estructural y de comportamiento), código (presentación y aplicación) y pruebas (unidad y aceptación). [NOTA: Consultar contenido al final del documento]

Requisitos: creacion de una nueva clase en la capa de aplicación, de la cual hereden los elementos necesarios que compartan comportamiento.

1. ¿Cuáles patrones se usan en la implementación actual? ¿Cuál patrón podría aplicar para la refactorización?¿por qué?

Patron de comportamiento, el patron observador se podria llegar a utilizar, ya que podrian ocurrir cambios que serian necesarios de informar a otros objetos.

1. Refactorice este comportamiento. Explique e incluya los componentes que cambiaron.

II. EXTENDIENDO. Nuevo elemento.

Extienda el proyecto para poder incluir un nuevo elemento (sorpresa, blanco o pelota) con las siguientes características:

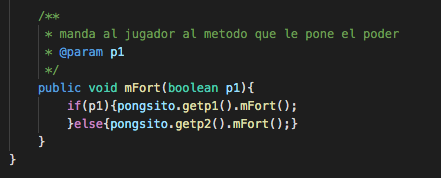
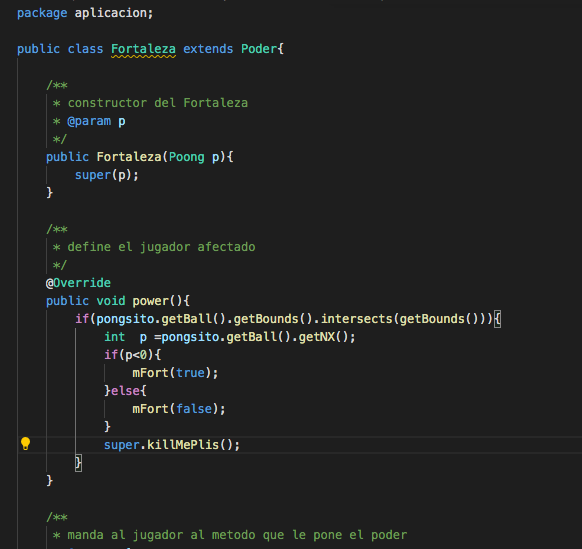
Si es sorpresa o blanco:

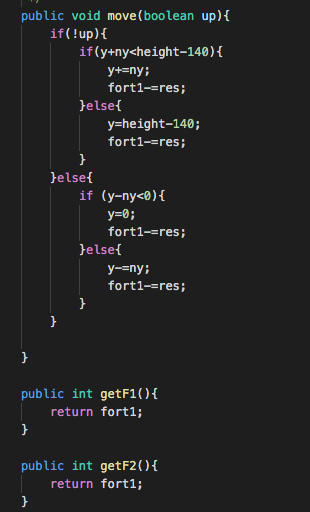
• Otorga al jugador que la capture, la habilidad de no perder el 50% de fortaleza si es golpeado por un bloque. Aplica solo una vez.

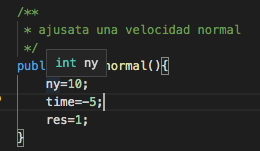
• Algunas veces (1 de cada 2) estos elementos buscan a la pelota.

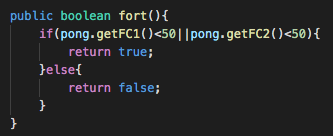
1. Incluya en el documento los siguientes componentes BDD-MDD: requisitos, diseño (estructural y de comportamiento), código (presentación y aplicación) y pruebas (unidad y aceptación).

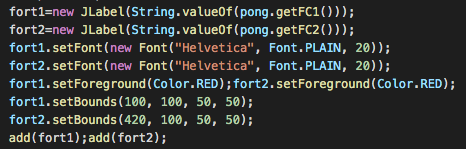
Requisitos: nueva clase en la capa de aplicación la cual herede de poder, ya que se hara como una sorpresa, controlar la fortaleza cuando se capture el poder con un atributo añadido a la clase personaje, el cual evita que se disminuya la fortaleza al chocar con un bloque.



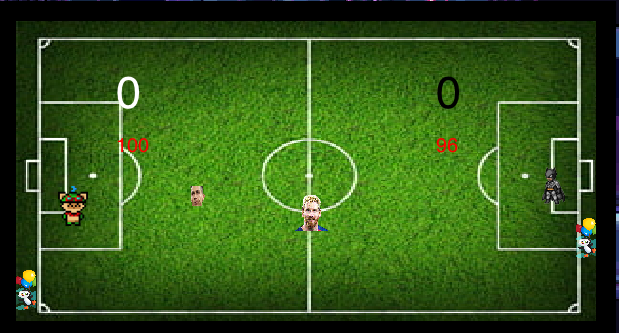




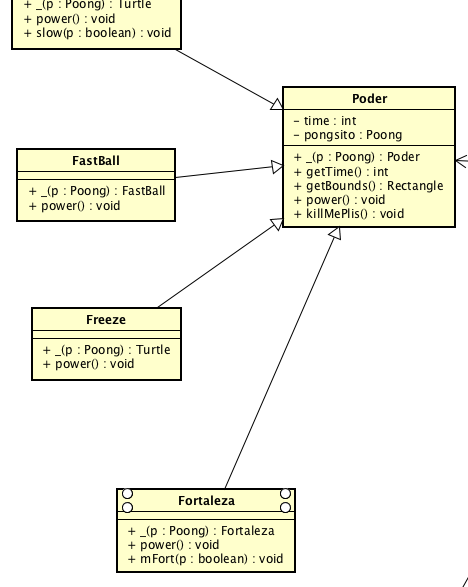
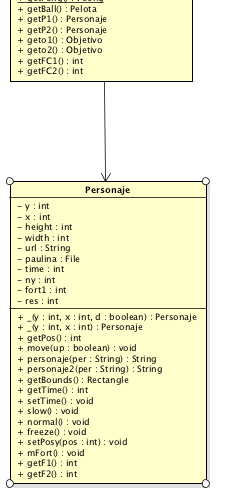


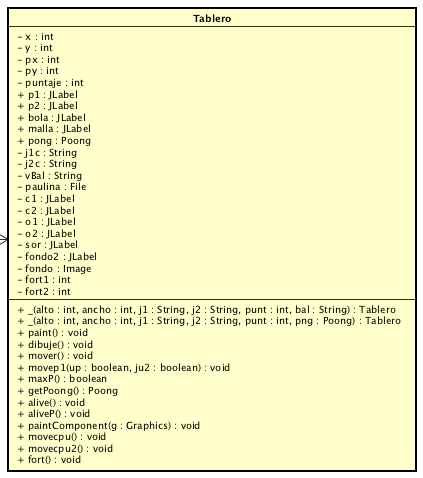


Pruebas de aceptacion: Al jugar, cuando el personaje obtiene el poder de la fortaleza, su fotaleza deja de disminuir hasta que se hace un gol, evitando asi que cuando un bloque lo choque, disminuya su fortaleza un 50%. (Los numeros en rojo son la fortaleza de cada personaje).



Astah:





1. ¿Cuáles patrones usa la implementación actual?

Patrón de diseño de creación, ya que este nos ayuda a la implementación de nuevos objetos en nuestro proyecto, el patrón prototipo podría ser útil, ya que podríamos crear y manipular copias de objetos ya creados, para así reducir código.