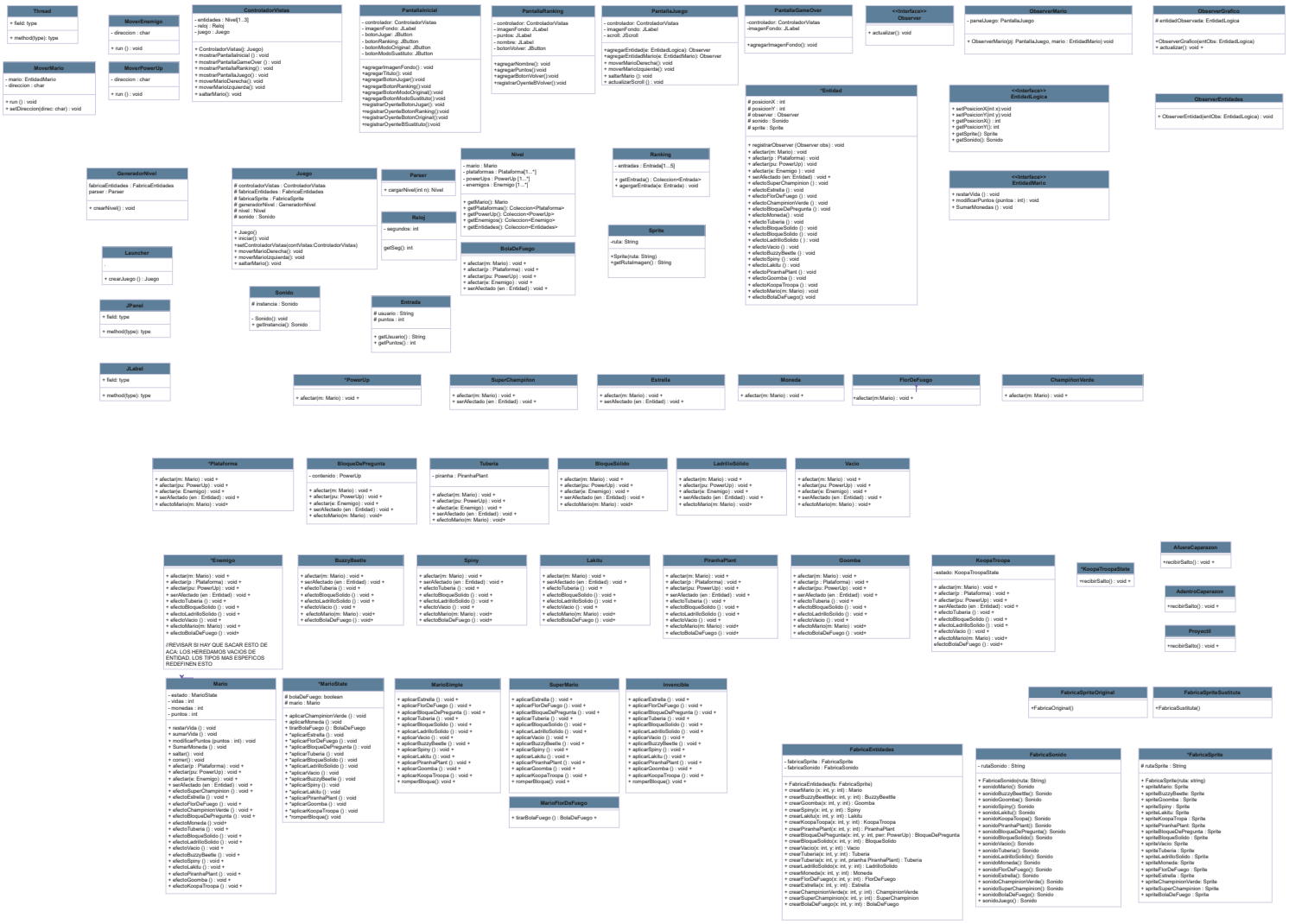
Thread
+ field: type
+ method(type): type
MoverMario
- mario: EntidadMario
- direccion : char
+ run () : void
+ setDireccion(direc: char) : void
MoverEnemigo
- direccion : char
+ run () : void
MoverPowerUp
- direccion : char
+ run () : void
ControladorVistas
- entidades : Nivel[1..3]
- reloj : Reloj
- juego : Juego
+ ControladorVistas(): Juego
+ mostrarPantallaInicial () : void
+ mostrarPantallaGameOver () : void
+ mostrarPantallaRanking() : void
+ mostrarPantallaJuego() : void
+ moverMarioDerecha(): void
+ moverMarioIzquierda(): void
+ saltarMario(): void
PantallaInicial
- controlador: ControladorVistas
- imagenFondo: JLabel
- botonJugar: JButton
- botonRanking: JButton
- botonModoOriginal: JButton
- botonModoSustituto: JButton
+agregarImagenFondo() : void
+agregarTitulo() : void
+agregarBotonJugar():void
+agregarBotonRanking():void
+agregarBotonModoOriginal():void
+agregarBotonModoSustituto():void
+registrarOyenteBotonJugar() :void
+registrarOyenteBotonRanking():void
+registrarOyenteBotonOriginal():void
+registrarOyenteBotoSustituto():void
PantallaRanking
- controlador: ControladorVistas
- imagenFondo: JLabel
- puntos: JLabel
- nombre: JLabel
- botonVolver: JButton
+agregarNombre(): void
+agregarPuntos():void
+agregarBotonVolver():void
+registrarOyenteBtnVolver():void
PantallaJuego
- controlador: ControladorVistas
- imagenFondo: JLabel
- scroll: JScroll
+agregarEntidad(a: EntidadLogica): Observer
+agregarEntidadMario(e: EntidadMario): Observer
+ moverMarioDerecha(): void
+ moverMarioIzquierda(): void
+ saltarMario () : void
+ actualizarScroll () : void
PantallaGameOver
-controlador: ControladorVistas
-imagenFondo: JLabel
+agregarImagenFondo(): void
<<Interface>>
Observer
+ actualizar(): void
ObserverMario
- panelJuego: PantallaJuego
+ ObserverMario(p: PantallaJuego, mario : EntidadMario) void
ObserverGrafico
entidadObservada: EntidadLogica
+ObserverGrafico(entOba: EntidadLogica)
+ actualizar(): void +
*Entidad
posicionX : int
posicionY : int
observer : Observer
sonido : Sonido
sprite : Sprite
+ registrarObserver (Observer oba) : void
+ afectar(m: Mario) : void
+ afectar(p : Plataforma) : void
+ afectar(pu: PowerUp) : void
+ afectar(e: Enemigo) : void
+ serAfectado (en: Entidad) : void +
+ efectoSuperChampinion () : void
+ efectoEstrella () : void
+ efectoFlorDeFuego () : void
+ efectoChampinionVerde () : void
+ efectoBloqueDePregunta () : void
+ efectoMoneda() : void
+ efectoTuberia () : void
+ efectoBloqueSolido () : void
+ efectoBloqueSolido () : void
+ efectoLadrilloSolido () : void
+ efectoVacio () : void
+ efectoBuzzyBeetle () : void
+ efectoSpiny () : void
+ efectoLakitu () : void
+ efectoPiranhaPlant () : void
+ efectoGoomba () : void
+ efectoKoopaTroopa () : void
+ efectoMario(m: Mario) : void
+ efectoBolaDeFuego() : void
<<Interface>>
EntidadLogica
+ setPosicionX(int x):void
+ setPosicionY(int y):void
+ getPosicionX() : int
+ getPosicionY() : int
+ getSprite(): Sprite
+ getSonido(): Sonido
ObserverEntidades
+ ObserverEntidad(entObs: EntidadLogica) : void
<<Interface>>
EntidadMario
+ restarVida () : void
+ modificarPuntos (puntos : int) : void
+ SumarMonedas () : void
GeneradorNivel
fabricaEntidades : FabricaEntidades
parser : Parser
+ crearNivel() : void
Juego
controladorVistas : ControladorVistas
fabricaEntidades : FabricaEntidades
fabricaSprite : FabricaSprite
generadorNivel : GeneradorNivel
nivel : Nivel
sonido : Sonido
+ Juego()
+ iniciar() :void
+setControladorVistas(conVistas:ControladorVistas)
+ moverMarioDerecha() :void
+ moverMarioIzquierda() :void
+ saltarMario() :void
Parser
+ cargarNivel(int n): Nivel
Reloj
- segundos: int
getSeg() :int
Nivel
- mario : Mario
- plataformas : Plataforma[1...*]
- powerUps : PowerUp [1...*]
- enemigos : Enemigo [1...*]
+ getMario() : Mario
+ getPlataformas() : Coleccion<Plataforma>
+ getPowerUp() : Coleccion<PowerUp>
+ getEnemigos() : Coleccion<Enemigo>
+ getEntidades() : Coleccion<Entidades>
BolaDeFuego
+ afectar(m: Mario) : void +
+ afectar(p : Plataforma) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
Ranking
- entradas : Entrada[1...5]
+ getEntrada() : Coleccion<Entrada>
+ agergarEntrada(e: Entrada) : void
Sprite
-ruta: String
+Sprite(ruta: String)
+getRutaImagen() : String
Launcher
-
+ crearJuego () : Juego
Sonido
instancia : Sonido
- Sonido():void
+ getInstancia():Sonido
Entrada
usuario : String
puntos : int
+ getUsuario() : String
+ getPuntos() : int
JPanel
+ field: type
+ method(type): type
JLabel
+ field: type
+ method(type): type
*PowerUp
+ afectar(m: Mario) : void +
SuperChampiion
+ afectar(m: Mario) : void +
+ serAfectado (en : Entidad) : void +
Estrella
+ afectar(m: Mario) : void +
+ serAfectado (en : Entidad) : void +
Moneda
+ afectar(m: Mario) : void +
FlorDeFuego
+afectar(m:Mario) : void +
ChampiionVerde
+ afectar(m: Mario) : void +
*Plataforma
+ afectar(m: Mario) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
+ efectoMario(m: Mario) : void+
BloqueDePregunta
- contenido : PowerUp
+ afectar(m: Mario) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
+ efectoMario(m: Mario) : void+
Tuberia
- piranha : PiranhaPlant
+ afectar(m: Mario) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
+ efectoMario(m: Mario) : void+
BloqueSolido
+ afectar(m: Mario) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
+ efectoMario(m: Mario) : void+
LadrilloSolido
+ afectar(m: Mario) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
+ efectoMario(m: Mario) : void+
Vacio
+ afectar(m: Mario) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
+ efectoMario(m: Mario) : void+
*Enemigo
+ afectar(m: Mario) : void +
+ afectar(p : Plataforma) : void +
+ afectar(pu: PowerUp) : void +
+ serAfectado (en : Entidad) : void +
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoMario(m: Mario) : void+
+ efectoBolaDeFuego () : void+
//REVISAR SI HAY QUE SACAR ESTO DE ACA. LOS HEREDAMOS VACIOS DE ENTIDAD. LOS TIPOS MAS ESPECIFICOS REDEFINEN ESTO
BuzzyBeetle
+ afectar(m: Mario) : void +
+ serAfectado (en : Entidad) : void +
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoMario(m: Mario) : void+
+ efectoBolaDeFuego () : void+
Spiny
+ afectar(m: Mario) : void +
+ serAfectado (en : Entidad) : void +
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoMario(m: Mario) : void+
+ efectoBolaDeFuego () : void+
Lakitu
+ afectar(m: Mario) : void +
+ serAfectado (en : Entidad) : void +
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoMario(m: Mario) : void+
+ efectoBolaDeFuego () : void+
PiranhaPlant
+ afectar(m: Mario) : void +
+ afectar(p : Plataforma) : void +
+ afectar(pu: PowerUp) : void +
+ serAfectado (en : Entidad) : void +
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoMario(m: Mario) : void+
+ efectoBolaDeFuego () : void+
Goomba
+ afectar(m: Mario) : void +
+ afectar(p : Plataforma) : void +
+ afectar(pu: PowerUp) : void +
+ serAfectado (en : Entidad) : void +
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoMario(m: Mario) : void+
+ efectoBolaDeFuego () : void+
KoopaTroopa
-estado: KoopaTroopaState
+ afectar(m: Mario) : void +
+ afectar(p : Plataforma) : void +
+ afectar(pu: PowerUp) : void +
+ serAfectado (en : Entidad) : void +
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoMario(m: Mario) : void+
efectoBolaDeFuego () : void+
*KoopaTroopaState
+recibirSalto() : void +
AfueraCaparazon
+recibirSalto() : void +
AdentroCaparazon
+recibirSalto() : void +
Proyectil
+recibirSalto() : void +
Mario
- estado : MarioState
- vidas : int
- monedas : int
- puntos : int
+ restarVida () : void
+ sumarVida () : void
+ modificarPuntos (puntos : int) : void
+ SumarMoneda () : void
+ saltar() : void
+ correr() : void
+ afectar(p : Plataforma) : void +
+ afectar(pu: PowerUp) : void +
+ afectar(e: Enemigo) : void +
+ serAfectado (en : Entidad) : void +
+ efectoSuperChampinion () : void +
+ efectoEstrella () : void +
+ efectoFlorDeFuego () : void +
+ efectoChampinionVerde () : void +
+ efectoBloqueDePregunta () : void +
+ efectoMoneda () :void+
+ efectoTuberia () : void +
+ efectoBloqueSolido () : void +
+ efectoLadrilloSolido () : void +
+ efectoVacio () : void +
+ efectoBuzzyBeetle () : void +
+ efectoSpiny () : void +
+ efectoLakitu () : void +
+ efectoPiranhaPlant () : void +
+ efectoGoomba () : void +
+ efectoKoopaTroopa () : void +
*MarioState
bolaDeFuego: boolean
mario : Mario
+ aplicarChampinionVerde () : void
+ aplicarMoneda () :void
+ tirarBolaFuego () : BolaDeFuego
+ *aplicarEstrella () : void
+ *aplicarFlorDeFuego () : void
+ *aplicarBloqueDePregunta () : void
+ *aplicarTuberia () : void
+ *aplicarBloqueSolido () : void
+ *aplicarLadrilloSolido () : void
+ *aplicarVacio () : void
+ *aplicarBuzzyBeetle () : void
+ *aplicarSpiny () : void
+ *aplicarLakitu () : void
+ *aplicarPiranhaPlant () : void
+ *aplicarGoomba () : void
+ *aplicarKoopaTroopa () : void
+ *romperBloque() : void
MarioSimple
+ aplicarEstrella () : void +
+ aplicarFlorDeFuego () : void +
+ aplicarBloqueDePregunta () : void +
+ aplicarTuberia () : void +
+ aplicarBloqueSolido () : void +
+ aplicarLadrilloSolido () : void +
+ aplicarVacio () : void +
+ aplicarBuzzyBeetle () : void +
+ aplicarSpiny () : void +
+ aplicarLakitu () : void +
+ aplicarPiranhaPlant () : void +
+ aplicarGoomba () : void +
+ aplicarKoopaTroopa () : void +
+ romperBloque() : void +
SuperMario
+ aplicarEstrella () : void +
+ aplicarFlorDeFuego () : void +
+ aplicarBloqueDePregunta () : void +
+ aplicarTuberia () : void +
+ aplicarBloqueSolido () : void +
+ aplicarLadrilloSolido () : void +
+ aplicarVacio () : void +
+ aplicarBuzzyBeetle () : void +
+ aplicarSpiny () : void +
+ aplicarLakitu () : void +
+ aplicarPiranhaPlant () : void +
+ aplicarGoomba () : void +
+ aplicarKoopaTroopa () : void +
+ romperBloque() : void +
MarioFlorDeFuego
+ tirarBolaFuego () : BolaDeFuego +
Invencible
+ aplicarEstrella () : void +
+ aplicarFlorDeFuego () : void +
+ aplicarBloqueDePregunta () : void +
+ aplicarTuberia () : void +
+ aplicarBloqueSolido () : void +
+ aplicarLadrilloSolido () : void +
+ aplicarVacio () : void +
+ aplicarBuzzyBeetle () : void +
+ aplicarSpiny () : void +
+ aplicarLakitu () : void +
+ aplicarPiranhaPlant () : void +
+ aplicarGoomba () : void +
+ aplicarKoopaTroopa () : void +
+ romperBloque() : void +
FabricaSpriteOriginal
+FabricaOriginal()
FabricaSpriteSustituta
+FabricaSustituta()
FabricaEntidades
- fabricaSprite : FabricaSprite
- fabricaSonido : FabricaSonido
+ FabricaEntidades(fs: FabricaSprite)
+ crearMario (x: int, y: int) : Mario
+ crearBuzzyBeetle(x: int, y: int) : BuzzyBeetle
+ crearGoomba(x: int, y: int) : Goomba
+ crearSpiny(x: int, y: int) : Spiny
+ crearLakitu(x: int, y: int) : Lakitu
+ crearKoopaTroopa(x: int, y: int) : KoopaTroopa
+ crearPiranhaPlant(x: int, y: int) : PiranhaPlant
+ crearBloqueDePregunta(x: int, y: int, pwr: PowerUp) : BloqueDePregunta
+ crearBloqueSolido(x: int, y: int) : BloqueSolido
+ crearVacio(x: int, y: int) : Vacio
+ crearTuberia(x: int, y: int) : Tuberia
+ crearTuberia(x: int, y: int, pirianha PiranhaPlant) : Tuberia
+ crearLadrilloSolido(x: int, y: int) : LadrilloSolido
+ crearMoneda(x: int, y: int) : Moneda
+ crearFlorDeFuego(x: int, y: int) : FlorDeFuego
+ crearEstrella(x: int, y: int) : Estrella
+ crearChampinionVerde(x: int, y: int) : ChampinionVerde
+ crearSuperChampinion(x: int, y: int) : SuperChampinion
+ crearBolaDeFuego(x: int, y: int) : BolaDeFuego
FabricaSonido
- rutaSonido : String
+ FabricaSonido(ruta: String)
+ sonidoMario(): Sonido
+ sonidoBuzzyBeetle(): Sonido
+ sonidoGoomba() : Sonido
+ sonidoSpiny() : Sonido
+ sonidoLakitu() : Sonido
+ sonidoKoopaTroopa() : Sonido
+ sonidoPiranhaPlant(): Sonido
+ sonidoBloqueDePregunta() :Sonido
+ sonidoBloqueSolido() : Sonido
+ sonidoVacio() : Sonido
+ sonidoTuberia() : Sonido
+ sonidoLadrilloSolido() : Sonido
+ sonidoMoneda() : Sonido
+ sonidoFlorDeFuego() : Sonido
+ sonidoEstrella() : Sonido
+ sonidoChampinionVerde() : Sonido
+ sonidoSuperChampinion() : Sonido
+ sonidoBolaDeFuego() : Sonido
+ sonidoJuego() : Sonido
*FabricaSprite
rutaSprite : String
+ FabricaSprite(ruta: string)
+ spriteMario: Sprite
+ spriteBuzzyBeetle: Sprite
+ spriteGoomba : Sprite
+ spriteSpiny : Sprite
+ spriteLakitu: Sprite
+ spriteKoopaTroopa : Sprite
+ spritePiranhaPlant: Sprite
+ spriteBloqueDePregunta: Sprite
+ spriteBloqueSolido: Sprite
+ spriteVacio: Sprite
+ spriteTuberia: Sprite
+ spriteLadrilloSolido: Sprite
+ spriteMoneda: Sprite
+ spriteFlorDeFuego : Sprite
+ spriteEstrella : Sprite
+ spriteChampinionVerde: Sprite
+ spriteSuperChampinion: Sprite
+ spriteBolaDeFuego : Sprite