Model::Main «interface» ModoDosFactory ModoDosFactory Ranking ManagerJuego ManagerPartida NivelBuilder ModoDosFactory +CrearLakitu(x: int, v: int): Lakitu +CrearLakitu(x: int, v: int): Lakitu #ranking: Ranking #ranking: Ranking #fabrica: GameFactory +CrearLakitu(x: int, y: int): Lakitu +CrearKoopa(x, y: int): Koopa +CrearKoopa(x, y: int): Koopa +esTop(int): boolean #partida: ManagerPartida #factory: GameFactory #imagenMapa: File +CrearKoopa(x, y: int): Koopa +CrearGoomba(x: int, y: int): Goomba +CrearGoomba(x: int, y: int): Goomba +ingresarTop(int, string) #factory: GameFactory #nivelCreado: Nive #nivelActual: int +CrearGoomba(x: int, y: int): Goomba +CrearSpiny(x: int, y: int): Spiny +CrearSpiny(x: int, y: int): Spiny +VerRanking() creadorNivel: NivelBuilder +CrearSpiny(x: int, y: int): Spiny +SeleccionarModo() +NivelBuilder(Factory: GameFactory, f: File) +CrearPiranha(x: int, v: int): Piranha +CrearPiranha(x; int. v; int); Piranha IniciarPartida(GameFactory) -leerArchivo() +CrearPiranha(x: int, y: int): Piranha +CrearBuzzy(x: int, y: int): Buzzy +CrearBuzzy(x: int, y: int): Buzzy +VerRanking() gameOver(puntaje final: int) +CrearBuzzy(x: int, y: int): Buzzy +CrearSuperChampi(x: int, y: int): SuperChampiñon +CrearSuperChampi(x: int, y: int): SuperChampiñon -procesarCaracter() -Ver si entró o no en el ranking (una vez terminada la siguienteNivel() +getNivel(): Nivel +CrearSuperChampi(x: int. v: int): SuperChampiñon +CrearEstrella(x: int, y: int): Estrella +CrearEstrella(x: int, y: int): Estrella +CrearEstrella(x: int, y: int): Estrella +CrearChampiVerde(x: int. v: int): ChampiñonVerde +CrearChampiVerde(x: int, v: int): ChampiñonVerde partida) seria responsabilidad +reiniciarNivelActual() +iniciarNivel(Nivel) +CrearChampiVerde(x: int, y: int): ChampiñonVerde +CrearFlorDeFuego(x: int, y: int): FlorDeFuego +CrearFlorDeFuego(x: int, y: int): FlorDeFuego +CrearFlorDeFuego(x: int. v: int): FlorDeFuego +CrearMoneda(x: int, y: int): Moneda +CrearMoneda(x: int, y: int): Moneda +CrearMoneda(x: int, y: int): Moneda Мара +CrearBolaDeFuego(x; int, v; int); BolaDeFuego +CrearBolaDeFuego(x; int, v; int); BolaDeFuego +CrearBolaDeFuego(x: int, y: int): BolaDeFuego +CrearBloqueDePreg(x: int, y: int): BloqueDePregunta +CrearBloqueDePreg(x: int, y: int): BloqueDePregunta +Attribute1 CollisionManager Nivel +CrearBloqueDePreg(x: int, y: int): BloqueDePregunta +CrearLadrilloSolido(x: int, y: int): LadrilloSolido +CrearLadrilloSolido(x: int, y: int): LadrilloSolido +Crearl adrilloSolido(x: int. v: int): LadrilloSolido MasterMind +CrearVacio(x: int, y: int): Vacio +CrearVacio(x: int, y: int): Vacio #nivel: Nivel #plataformas[]: Elemento +CrearVacio(x: int, y: int): Vacio +CrearTuberias(x: int, v: int): Tuberias +CrearTuberias(x; int. v; int); Tuberias +masterMind: MasterMind enemigosfl: Elemento +CrearTuberias(x: int, y: int): Tuberias +CrearBloqueSolido(x: int, y: int): BloqueSolido +CrearBloqueSolido(x: int, y: int): BloqueSolido +gestionarColision(Elemento, Elemento) #powerUps[]: Elemento -detectarColisiones() +CrearBloqueSolido(x: int, y: int): BloqueSolido +CrearJugador(x: int, y: int): Jugador +CrearJugador(x: int, y: int): Jugador #jugador: Jugador +CrearJugador(x: int. v: int): Jugador #tiempoPartida: float #managPartida: ManagerPartida Sonido +agregarPlataforma(Plataforma) mapeoSonido: Map<String,Clip> #x: int +agregarEnemigo(Enemigo) #volumen: float #y: int +agregarPowerUn(PowerUn) #ancho: int +CargarSonido(key: String, soundFilePath: String) +agregarJugador(Jugador) #alto: int +reproducirSonido(clave: String) +getPlataformas(): Elemento[*] #imagen: File +detenerSonido(clip: Clip) +getEnemigos(): Elemento[*] +reiniciarSonido(clip: Clip) +getPowerUps(): Elemento[*] +Sprite(x: int, y: int, ancho: int, Parameter3) +getSonido() +dibujar(imagen: File) +getJugador(): Jugador +getVolumen() +iniciarTimer() +actualizarPosicion(x: int. v: int) +setVolumen() +cambiarlmagen(imagen; File) +aplicarEfecto(pa: PowerUp) +resetear() +eliminar() Visitor (interfaz) +imagenPrevia(): File +visitar(ele: Elemento) InfoJugador Jugador State Invulnerable #timer: float Elemento #jugador: Jugador #posicionX: int #sprite: Image BolaDeFuego Enemigo PowerUp #monedas: int #posicionY: int #jugador: Jugador estadoAnterior: State Plataforma #posicionX: int #posicionX: int #posicionX: int #posicionX: int #puntaje: int #velocidad: float #managPartida: ManagerPartida -recibirDaño∩ #nosicionY: int #nosicionX: int #posicionY: int #posicionY: int #posicionY: int #sunerMario: hoolea #sprite: Image #sprite: Image +recibirDaño() +aumentarEstado(PowerUp) #posicionY: int #vida: int #info: InfoJugador #sprite: Image #sprite: Image #sprite: Image +aumentarEstado(Powerl In #sprite: Image #nivel: Nivel #estado: State #velocidad: float #velocidad: float #velocidad: float #managPartida: ManagerPartida FlorDeFuego #jugador: InfoJugado +Jugador(x: int, y: int, im: Image) visitar(Jugador) +visitar(Jugador) Movible +getPuntaje(): int +setState(State) -aceptarVisita(Visitor) +visitar(Enemigo) -moverse() Estatico +recibirDaño∩ +getVida(): int +getState(): State -moverse() +moverse() #posicionX: int +aumentarEstado(PowerUn #nosicionX: in recibirDaño() +BolaDeFuego(x: int, y: int, im: Image) +sumarVida() +moverse() #posicionY: int +saltar() +aumentarMoneda() #posicionY: int +nuntosQueResta(): int #sprite: Image SuperMario +actualizarPuntaje(int) +disparar() #sprite: Image +puntosOueDa(): int #velocidad: float +getJugador(): Jugador +aetPosicion() +morir() +recibirDaño() +moverse(+morir() +visitar(Fnemigo) +restarVida() +getInfo(): InfoJugador -aumentarEstado(PowerUp) recibirDaño() BloqueDePregunta Normal Lakitu SuperChampiñon

+recibirDaño∩

+aumentarEstado(PowerUp)

