Model::Main

Personaje
#velocidad: int
#poder: boolean
+Personaje(n: Nivel, c: Celda, v: int, p: boolean)
+getVelocidad(): int
+getPoder(): boolean
+getCelda(): Celda
+morir(): int
+setCelda(c: Celda)
+getNivel(): Nivel

Nivel
#puntaje: int
#tiempo: int
#cantDestruibles: int
+Nivel(j: Juego)
+getPuntaje(): int
+getTiempo(): int
+getCantDestruibles(): int
+getCelda(fila: int, columna int): Celda
+incrementarTiempo()
+incrementarPuntaje(n: int)
+disminuirCantDestruibles()
+finalizarJuego()
+ganarJuego()
+getAlto(): int
+getAncho(): int
+getJuego(): Juego
+getEnemigos(): coleccion de Enemigo
+setEnemigo(e: enemigo)
+eliminarEnemigo(e: Enemigo)

El constructor inicia el juego

disminuirCantDestruibles(): diminuye en 1 cantDestruibles

ganarJuego(): si cantDestruibles es cero

incrementarTiempo(): aumenta en 1 tiempo

Juego
+cantRugulos: int
+cantAltair: int
+cantSiruis: int
+Juego(GUI gui)
+mover()
+mover(dir: int)
+getGui(): Gui

Gui

#miNivel 1
#miNivel 1
#miJuego 1
#miGui 1
#miNivel 1

Bomberman
#cantBombas: int
#alcanceBomba: int
+Bomberman(n: Nivel, c: Celda)
+mover(direccion: int)
+duplicarVelocidad()
+setPoder(pd: boolean)
+ponerBomba()
+morir+(): int
+getNivel(): Nivel
+agregarBomba()
+duplicarAlcance()
+getCantBombas(): int
+getAlcanceBomba(): int
+setCantidadBombas(n: int)

morir(): int retornar cero

#miBomberman 1

Enemigo
#valor: int
+Enemigo(n: Nivel, c: Celda, v: int, p: boolean, v: int)
+mover()
+morir(): int
+getValor(): int

#misEnemigos 0..6
#misEnemigos 0..6

Rugulo
+Rugulo(n: Nivel, c: Celda)
+mover+()

Altair
+Altair(n: Nivel, c: Celda)
+mover+()

Siruis
+Siruis(n: Nivel, c: Celda)
+mover+()

Valor, velocidad y poder no se reciben por parámetro. Son fijos para cada Enemigo y se establecen en su

Celda
#fila: int
#columna: int
+Celda(f: int, c: int, nivel: Nivel, power: PowerUp, estado: Estado)
+getFila(): int
+getColumna(): int
+getNivel(): Nivel
+getPowerUp(): PowerUp
+getBomberman(): Bomberman
+getEnemigos(): colección de Enemigos
+explotar(): int
+getEstado(): Estado
+setEstado(e: Estado)
+avanzar(e: Enemigo, dir: int)
+avanzar(b: Bomberman, dir: int)
+setBomberman(b: Bomberman)
+setEnemigo(e: Enemigo)
+eliminarEnemigo(e: Enemigo)
+setPowerUp(p: PowerUp)

explotar() modifica el estado de la celda

#miCelda 1
#miGrilla[][] 403
#miNivel 1
#miCelda 1
#miCelda 1
#miCelda 1

Estado
#Estado(c: Celda)
+avanzar(e: Enemigo, dir: int, c: Celda)
+avanzar(b: Bobmerman)
+destruir()
+getCelda(): Celda

#miEstado 1

Libre
+Libre(c: Celda)
+avanzar+(e: Enemigo, dir: int, c: Celda)
+avanzar+(b: Bomberman, dir: int, c: Celda)
+destruir+(): int

Destruible
+Destruible(c: Celda)
+avanzar+(e: Enemigo, dir: int, c: Celda)
+avanzar+(b: Bomberman, dir: int, c: Celda)
+destruir+(): int

Indestructible
+Indestruible(c: Celda)
+avanzar+(e: Enemigo, dir: int, c: Celda)
+avanzar+(b: Bomberman, dir: int, c: Celda)
+destruir+(): int

PowerUp
#valor: int
#PowerUp(c: Celda, valor: int)
+getValor(): int
+comportamiento(b: Bomberman)
+getCelda(): Celda

comportamiento(Bomberman) aumenta puntaje en Nivel a traves de Bomberman

#miPowerUp 0..1

SpeedUp
+SpeedUp(c: Celda)
+comportamiento+(b: Bomberman)

Fatality
+Fatality(c: Celda)
+comportamiento+(b: Bomberman)

Bombality
+bombality(c: Celda)
+comportamiento+(b: Bomberman)

Masacrallity
+Masacrallity(c: Celda)
+comportamiento+(b: Bomberman)

Bomba
#alcance: int
+Bomba(c: Celda, alcance: int)
+getAlcance(): int
+explotar(celda: Celda): int

explotar(Celda): verifica en la Celda y sus vecinos que tenga un estado destruible, enemigo y bomberman. Llama a los metodos explotar() del Celda y morir() de los personajes y lleva suma de lo que devuelve cada uno. Si el Bomberman no muere, aumenta el puntaje en nivel

explotar(Celda) debe tardar unos segundos en realizar la accion de destruir las celdas segun su alcence, y su efecto debe durar también unos segundos

#miBomberman 0..1
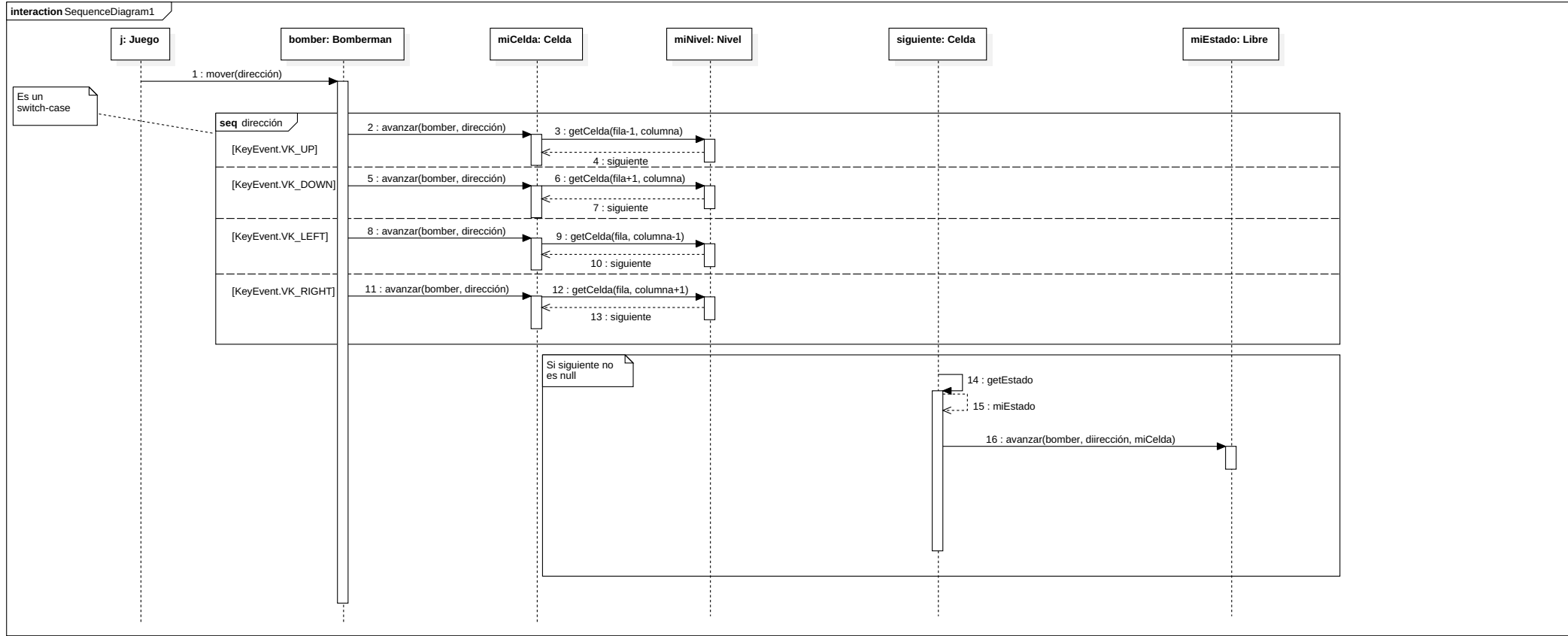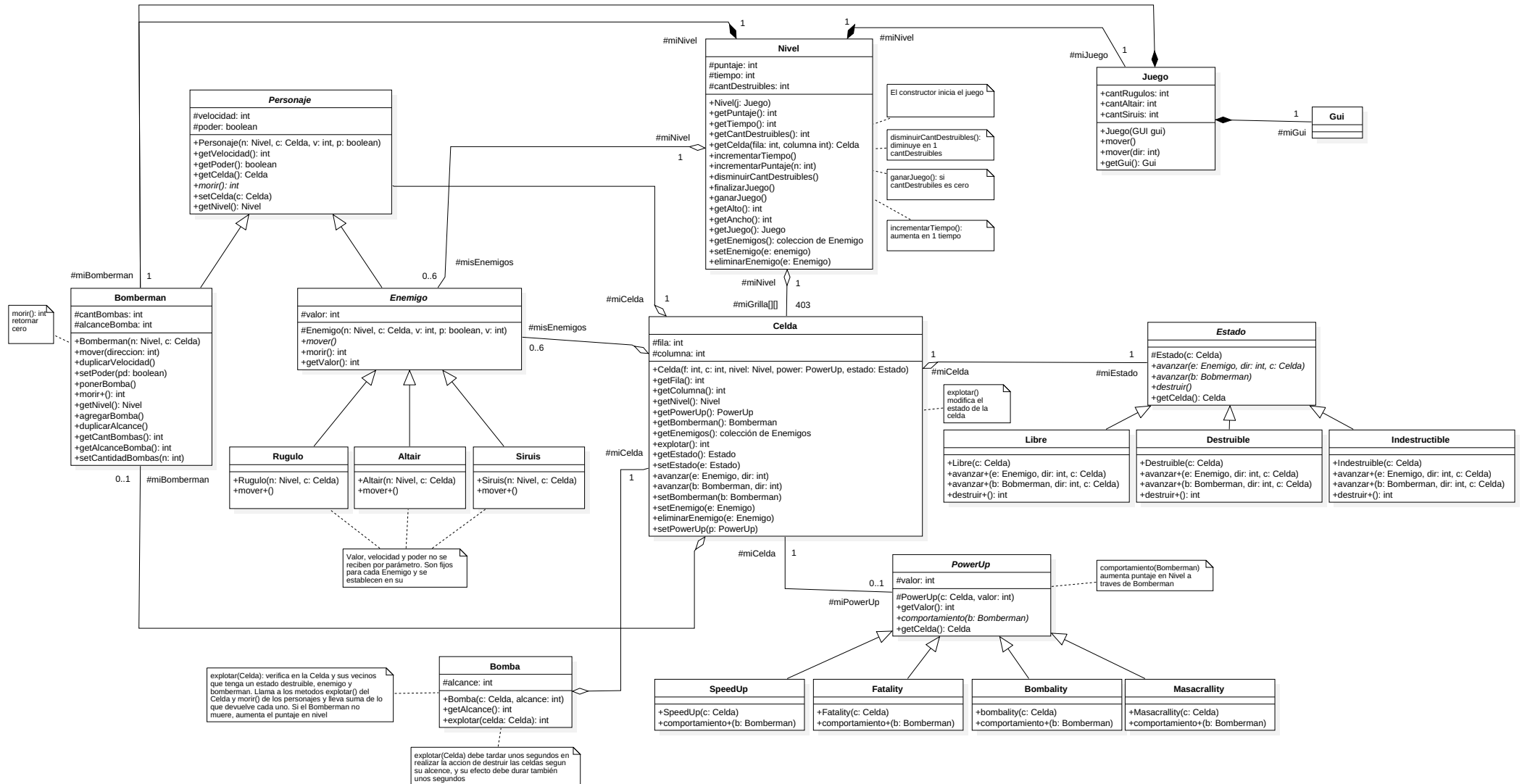
Model::Collaboration1::Interaction1::SequenceDiagram1

interaction SequenceDiagram1

j: Juego	bomber: Bomberman	miCelda: Celda	miNivel: Nivel	siguiente: Celda	miEstado: Libre

1 : mover(dirección)

Es un
switch-case

seq dirección

2 : avanzar(bomber, dirección)

3 : getCelda(fila-1, columna)

[KeyEvent.VK_UP]

4 : siguiente

5 : avanzar(bomber, dirección)

6 : getCelda(fila+1, columna)

[KeyEvent.VK_DOWN]

7 : siguiente

8 : avanzar(bomber, dirección)

9 : getCelda(fila, columna-1)

[KeyEvent.VK_LEFT]

10 : siguiente

11 : avanzar(bomber, dirección)

12 : getCelda(fila, columna+1)

[KeyEvent.VK_RIGHT]

13 : siguiente

Si siguiente no
es null

14 : getEstado

15 : miEstado

16 : avanzar(bomber, diirección, miCelda)

Personaje

#velocidad: int
#poder: boolean

+Personaje(n: Nivel, c: Celda, v: int, p: boolean)
+getVelocidad(): int
+getPoder(): boolean
+getCelda(): Celda
morir(): int
+setCelda(c: Celda)
+getNivel(): Nivel

Nivel

#puntaje: int
#tiempo: int
#cantDestruibles: int

+Nivel(j: Juego)
+getPuntaje(): int
+getTiempo(): int
+getCantDestruibles(): int
+getCelda(fila: int, columna int): Celda
+incrementarTiempo()
+incrementarPuntaje(n: int)
+disminuirCantDestruibles()
+finalizarJuego()
+ganarJuego()
+getAlto(): int
+getAncho(): int
+getJuego(): Juego
+getEnemigos(): coleccion de Enemigo
+setEnemigo(e: enemigo)
+eliminarEnemigo(e: Enemigo)

El constructor inicia el juego

disminuirCantDestruibles():
diminuye en 1 cantDestruibles

ganarJuego(): si
cantDestruibles es cero

incrementarTiempo():
aumenta en 1 tiempo

Juego

+cantRugulos: int
+cantAltair: int
+cantSiruis: int

+Juego(GUI gui)
+mover()
+mover(dir: int)
+getGui(): Gui

Gui

Bomberman

#cantBombas: int
#alcanceBomba: int

+Bomberman(n: Nivel, c: Celda)
+mover(direccion: int)
+duplicarVelocidad()
+setPoder(pd: boolean)
+ponerBomba()
+morir+(): int
+getNivel(): Nivel
+agregarBomba()
+duplicarAlcance()
+getCantBombas(): int
+getAlcanceBomba(): int
+setCantidadBombas(n: int)

morir(): int
retornar
cero

Enemigo

#valor: int

+Enemigo(n: Nivel, c: Celda, v: int, p: boolean, v: int)
mover()
+morir(): int
+getValor(): int

Celda

#fila: int
#columna: int

+Celda(f: int, c: int, nivel: Nivel, power: PowerUp, estado: Estado)
+getFila(): int
+getColumna(): int
+getNivel(): Nivel
+getPowerUp(): PowerUp
+getBomberman(): Bomberman
+getEnemigos(): colección de Enemigos
+explotar(): int
+getEstado(): Estado
+setEstado(e: Estado)
+avanzar(e: Enemigo, dir: int)
+avanzar(b: Bomberman, dir: int)
+setBomberman(b: Bomberman)
+setEnemigo(e: Enemigo)
+eliminarEnemigo(e: Enemigo)
+setPowerUp(p: PowerUp)

explotar()
modifica el
estado de la
celda

Estado

#Estado(c: Celda)
avanzar(e: Enemigo, dir: int, c: Celda)
avanzar(b: Bobmerman)
destruir()
+getCelda(): Celda

Rugulo

+Rugulo(n: Nivel, c: Celda)
+mover+()

Altair

+Altair(n: Nivel, c: Celda)
+mover+()

Siruis

+Siruis(n: Nivel, c: Celda)
+mover+()

Valor, velocidad y poder no se
reciben por parámetro. Son fijos
para cada Enemigo y se
establecen en su

Libre

+Libre(c: Celda)
+avanzar+(e: Enemigo, dir: int, c: Celda)
+avanzar+(b: Bomberman, dir: int, c: Celda)
+destruir+(): int

Destruible

+Destruible(c: Celda)
+avanzar+(e: Enemigo, dir: int, c: Celda)
+avanzar+(b: Bomberman, dir: int, c: Celda)
+destruir+(): int

Indestructible

+Indestruible(c: Celda)
+avanzar+(e: Enemigo, dir: int, c: Celda)
+avanzar+(b: Bomberman, dir: int, c: Celda)
+destruir+(): int

PowerUp

#valor: int

#PowerUp(c: Celda, valor: int)
+getValor(): int
comportamiento(b: Bomberman)
+getCelda(): Celda

comportamiento(Bomberman)
aumenta puntaje en Nivel a
traves de Bomberman

Bomba

#alcance: int

+Bomba(c: Celda, alcance: int)
+getAlcance(): int
+explotar(celda: Celda): int

explotar(Celda): verifica en la Celda y sus vecinos
que tenga un estado destruible, enemigo y
bomberman. Llama a los metodos explotar() del
Celda y morir() de los personajes y lleva suma de lo
que devuelve cada uno. Si el Bomberman no
muere, aumenta el puntaje en nivel

explotar(Celda) debe tardar unos segundos en
realizar la accion de destruir las celdas segun
su alcence, y su efecto debe durar también
unos segundos

SpeedUp

+SpeedUp(c: Celda)
+comportamiento+(b: Bomberman)

Fatality

+Fatality(c: Celda)
+comportamiento+(b: Bomberman)

Bombality

+bombality(c: Celda)
+comportamiento+(b: Bomberman)

Masacrallity

+Masacrallity(c: Celda)
+comportamiento+(b: Bomberman)

#miNivel 1
#miNivel
#miJuego 1
1 1
#miNivel 1
#miBomberman 1
#misEnemigos
0..6
#misEnemigos
0..6
#miCelda 1
#miNivel 1
#miGrilla[][] 403
#miCelda 1
#miEstado
1 1
#miCelda
1
#miCelda 1
#miPowerUp 0..1
0..1 #miBomberman
1 #miGui

Collaboration1::Interaction1::SequenceDiagram1

interaction SequenceDiagram1 — Avanzar en miEstado (continuación)

siguiente: Celda	**miEstado: Libre**	**miCelda: Celda**	**bomber: Bomberman**	**power: PowerUP**

1 : avanzar(bomber, dirección, miCelda)

2 : setBomberman(null)

3 : setBomberman(bomber)

4 : setCelda(siguiente)

5 : getPowerUp

6 : power

Si power no es null

7 : comportamiento(bomber)

8 : efecto

9 : getEnemigos.size

10 : cantEnemigos

Si cantEnemigos > 0

11 : getPoder

12 : tienePoder

Si tienePoder es falso

13 : morir