











































Diagrama de Flujo - Tetris



Bizagi Modeler

Tabla de Contenidos

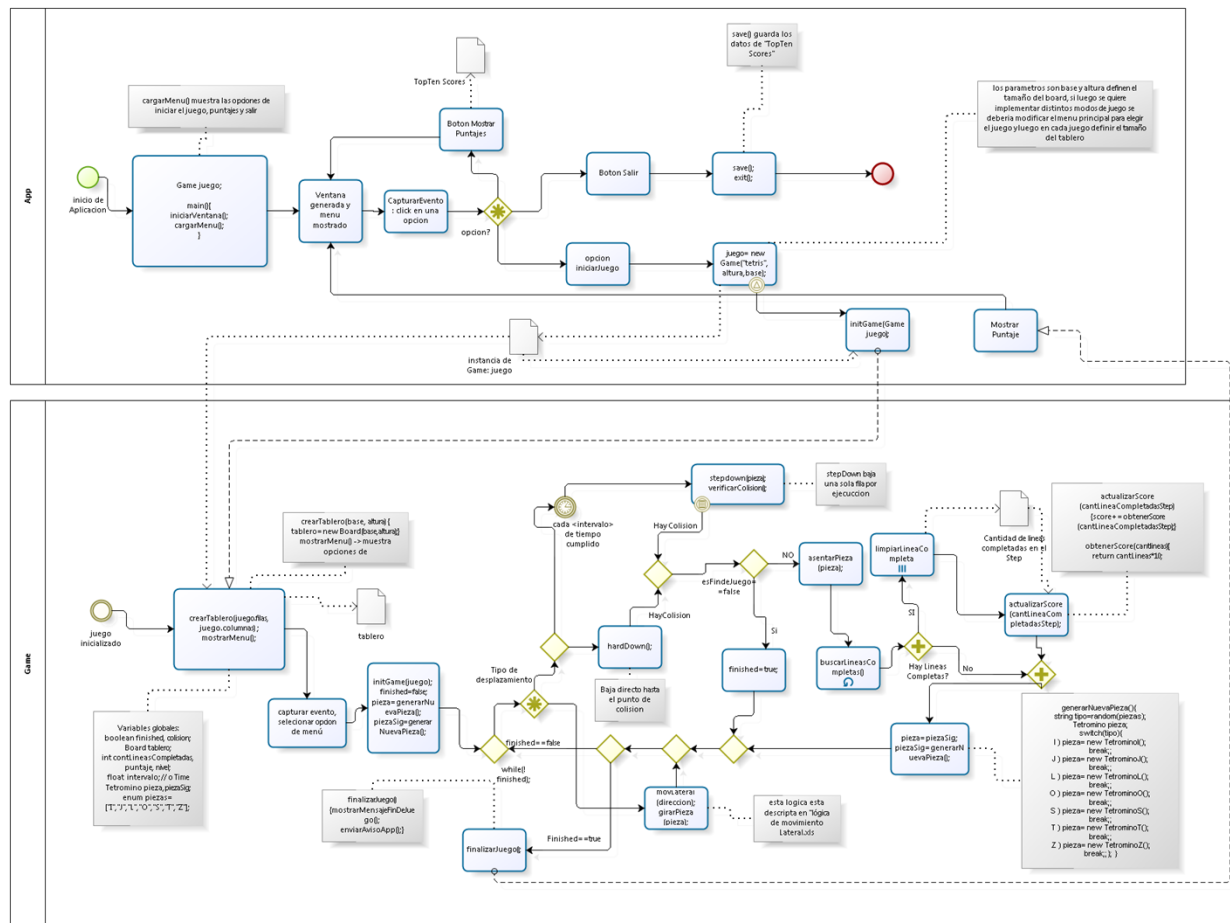
DIAGRAMA DE FLUJO - TETRIS	1
BIZAGI MODELER	1
1 LOGICA DE APP Y GAME	7
1.1 APP.....	8
1.1.1 Elementos del proceso	8
1.1.1.1  inicio de Aplicacion	8
1.1.1.3  Game juego;	8
1.1.1.5 main(){	8
1.1.1.6 iniciarVentana();.....	8
1.1.1.7 cargarMenu();	8
1.1.1.8 }	8
1.1.1.9  Ventana generada y menu mostrado	8
1.1.1.10  CapturarEvento: click en una opcion	8
1.1.1.11  opcion?	8
1.1.1.12  Boton Mostrar Puntajes	8
1.1.1.13  Boton Salir	9
1.1.1.14  save();	9
1.1.1.15 exit();.....	9
1.1.1.16  Event.....	9
1.1.1.17  opcion iniciarJuego	9
1.1.1.18  juego= new Game("tetris", altura,base);	9
1.1.1.19  initGame(Game juego);.....	9
1.1.1.20  Event.....	9
1.1.1.21  Mostrar Puntaje.....	9
1.1.1.22  TopTen Scores	9
1.1.1.23  instancia de Game: juego	9
1.2 GAME	9
1.2.1 Elementos del proceso	9

1.2.1.1		juego inicializado	9
1.2.1.2		crearTablero(juego.filas, juego.columnas) ;	10
1.2.1.3		mostrarMenu();	10
1.2.1.4		capturar evento, seleccionar opcion de menú	10
1.2.1.5		initGame(juego);	10
1.2.1.6		finished=false;	10
1.2.1.7		pieza=generarNuevaPieza();	10
1.2.1.8		piezaSig=generarNuevaPieza();	10
1.2.1.9		while(!finished);.....	10
1.2.1.10		Tipo de desplazamiento	10
1.2.1.11		Gateway.....	10
1.2.1.12		cada <intervalo> de tiempo cumplido.....	10
1.2.1.13		stepdown(pieza);.....	10
1.2.1.14		verificarColision();	10
1.2.1.15		hardDown();	11
1.2.1.16		Gateway.....	11
1.2.1.17		esFindeJuego==false	11
1.2.1.18		asentarPieza(pieza);	11
1.2.1.19		buscarLineasCompletas()	11
1.2.1.20		Hay Lineas Completas?.....	11
1.2.1.21		limpiarLineaCompleta	11
1.2.1.22		actualizarScore(cantLineaCompletadasStep);	11
1.2.1.23		Gateway.....	11
1.2.1.24		pieza=piezaSig;	11
1.2.1.25		piezaSig=generarNuevaPieza());.....	11
1.2.1.26		Gateway.....	11

1.2.1.27	 Gateway	12
1.2.1.28	 Gateway	12
1.2.1.29	 finalizarJuego();	12
1.2.1.30	 finished=true;	12
1.2.1.31	 movLateral(direccion);	12
1.2.1.32	girarPieza(pieza);	12
1.2.1.33	 Event	12
1.2.1.34	 Cantidad de lineas completadas en el Step	12
1.2.1.35	 tablero	12
2	LÓGICA DE BOARD	13
2.1	BOARD	14
2.1.1	Elementos del proceso	14
2.1.1.1	 new Board.....	14
2.1.1.2	 tiene parametros?	14
2.1.1.3	 Slot tablero[10][20];	14
2.1.1.4	 Gateway	14
2.1.1.5	 foreach(punto in tablero){punto=new Slot();}	14
2.1.1.6	 return tablero;	14
2.1.1.7	 Event	14
2.1.1.8	 Slot tablero[filas][columnas].....	14
2.1.1.9	 Slots inicializados.....	14
2.2	TETROMINO //ABSTRACT	15
2.2.1	Elementos del proceso	15
2.2.1.1	 new TetrominoX();	15
2.2.1.2	 Funciones	15
2.2.1.3	 girar()	15
2.2.1.4	 Event	15

2.2.1.5	 draw();.....	15
2.2.1.6	 asentar	15

1 LOGICA DE APP Y GAME



Versión: 1.0

Autor: Nico

1 . 1 A P P

1.1.1 ELEMENTOS DEL PROCESO

1.1.1.1  inicio de Aplicacion

1.1.1.2

1.1.1.3  Game juego;

1.1.1.4

1.1.1.5 main(){

1.1.1.6 iniciarVentana();

1.1.1.7 cargarMenu();

1.1.1.8 }

1.1.1.9  Ventana generada y menu mostrado

1.1.1.10  CapturarEvento: click en una opcion

1.1.1.11  opcion?

Flujos

Boton Mostrar Puntajes

Boton Salir

opcion iniciarJuego

1.1.1.12  Boton Mostrar Puntajes

1.1.1.13  Boton Salir

1.1.1.14  save();

1.1.1.15 exit();

1.1.1.16  Event

1.1.1.17  opcion iniciarJuego

1.1.1.18  juego= new Game("tetris", altura,base);

1.1.1.19  initGame(Game juego);

1.1.1.20  Event

1.1.1.21  Mostrar Puntaje

1.1.1.22  TopTen Scores

1.1.1.23  instancia de Game: juego

1 . 2 G A M E

1.2.1 ELEMENTOS DEL PROCESO

1.2.1.1  juego inicializado

1.2.1.2  crearTablero(juego.filas, juego.columnas) ;

1.2.1.3 mostrarMenu();

1.2.1.4  capturar evento, seleccionar opcion de menú

1.2.1.5  initGame(juego);

1.2.1.6 finished=false;

1.2.1.7 pieza=generarNuevaPieza();

1.2.1.8 piezaSig=generarNuevaPieza();

1.2.1.9  while(!finished);

Flujos

finished==false

1.2.1.10  Tipo de desplazamiento

Flujos

Flujo

movLateral(direccion);
girarPieza(pieza);

1.2.1.11  Gateway

Flujos

cada de tiempo cumplido

hardDown();

1.2.1.12  cada <intervalo> de tiempo cumplido

1.2.1.13  stepdown(pieza);

1.2.1.14 verificarColision();

1.2.1.15  hardDown();

1.2.1.16  Gateway

Flujos

esFindeJuego==false

1.2.1.17  esFindeJuego==false

Flujos

NO

Si

1.2.1.18  asentarPieza(pieza);

1.2.1.19  buscarLineasCompletas()

1.2.1.20  Hay Lineas Completas?

1.2.1.21  limpiarLineaCompleta

1.2.1.22  actualizarScore(cantLineaCompletadasStep);

1.2.1.23  Gateway

1.2.1.24  pieza=piezaSig;

1.2.1.25  piezaSig=generarNuevaPieza();

1.2.1.26  Gateway

Flujos

Flujo

1.2.1.27  Gateway

Flujos

Flujo

1.2.1.28  Gateway

Flujos

Finished==true

while(!finished);

1.2.1.29  finalizarJuego();

1.2.1.30  finished=true;

1.2.1.31  movLateral(direccion);

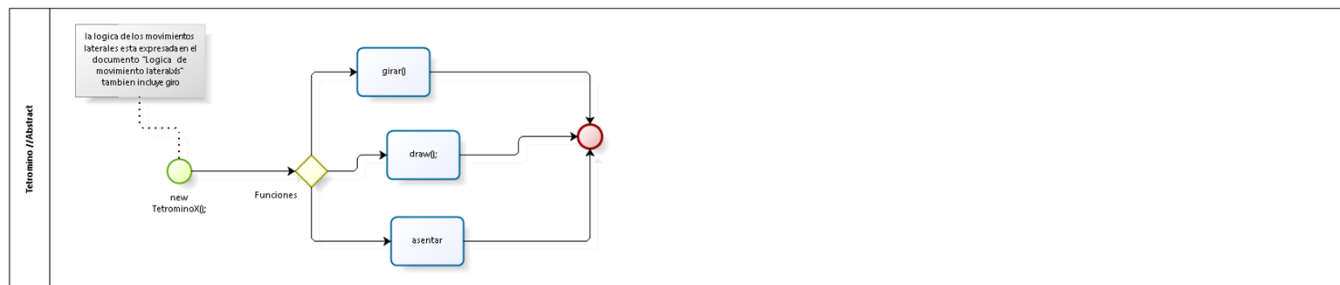
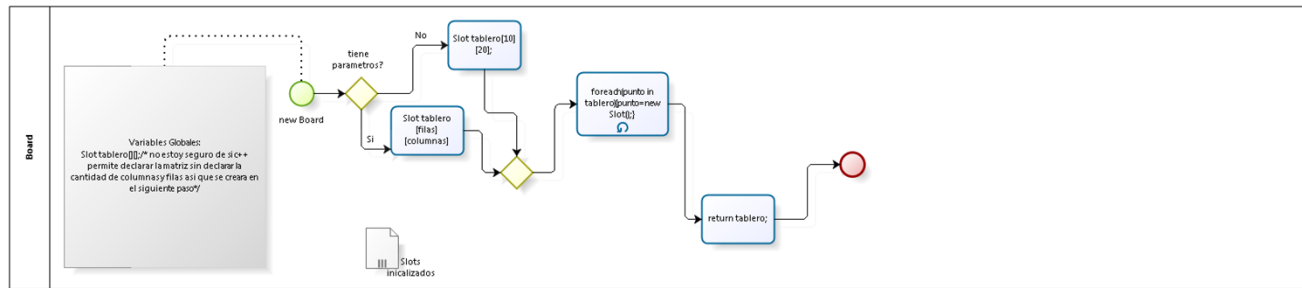
1.2.1.32 girarPieza(pieza);

1.2.1.33  Event

1.2.1.34  Cantidad de lineas completadas en el Step

1.2.1.35  tablero

2 LÓGICA DE BOARD



Versión: 1.0

Autor: Nico

2 . 1 B O A R D

2.1.1 ELEMENTOS DEL PROCESO

2.1.1.1  new Board

2.1.1.2  tiene parametros?

Flujos

Si

No

2.1.1.3  Slot tablero[10][20];

2.1.1.4  Gateway

Flujos

foreach(punto in tablero){punto=new Slot();}

2.1.1.5  foreach(punto in tablero){punto=new Slot();}

2.1.1.6  return tablero;

2.1.1.7  Event

2.1.1.8  Slot tablero[filas][columnas]

2.1.1.9  Slots inicializados

2.2 TETROMINO // ABSTRACT

2.2.1 ELEMENTOS DEL PROCESO

2.2.1.1  `new TetrominoX();`

2.2.1.2  **Funciones**

Flujos

girar()


draw();

asentar

2.2.1.3  `girar()`

2.2.1.4  `Event`

2.2.1.5  `draw();`

2.2.1.6  `asentar`