

Proyecto Semestral:

Colisión entre objetos móviles

Segunda Iteración

Profesor: Jorge López.

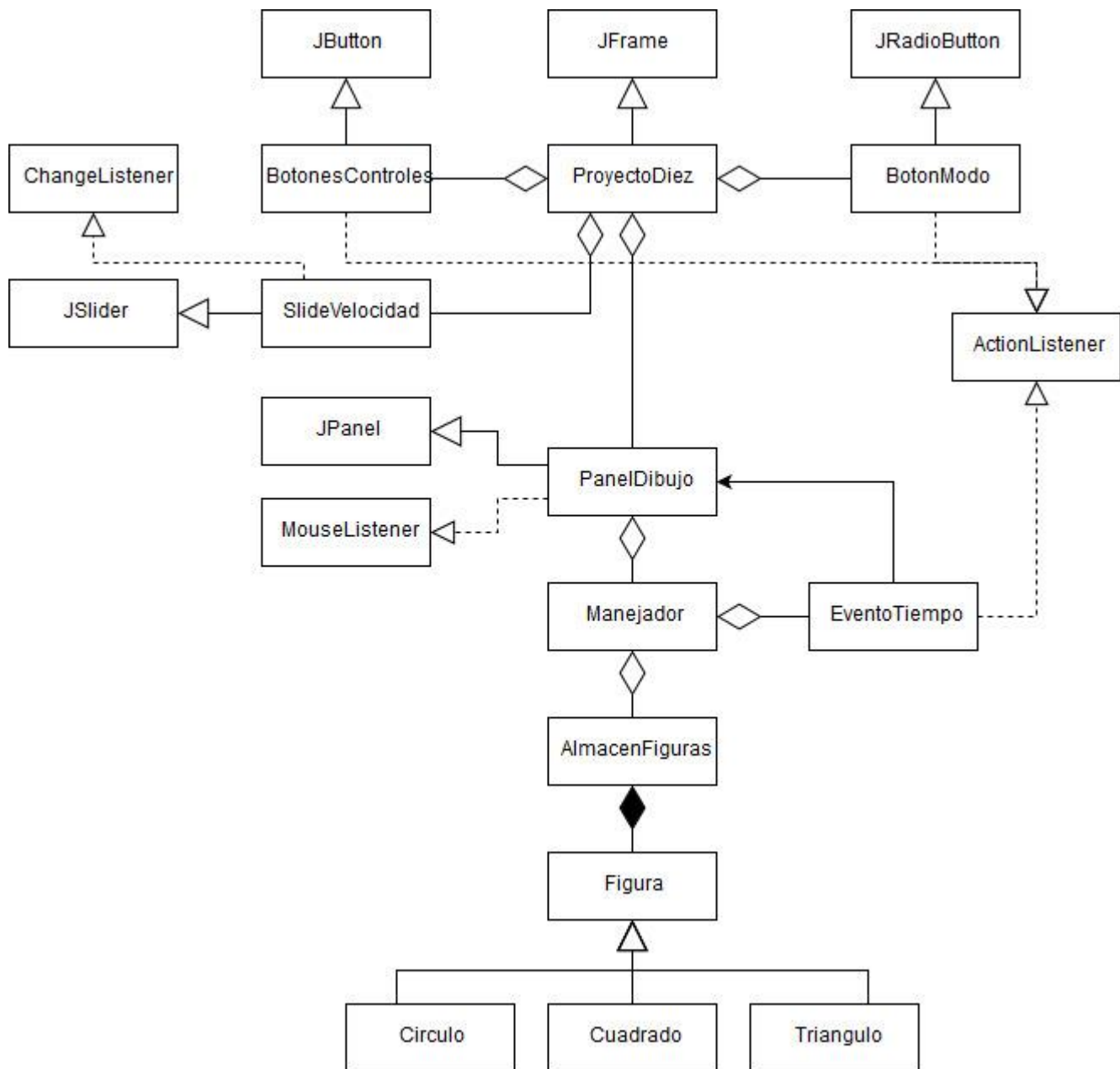
Ayudantes: Diego Gatica y Rodrigo Córdova.

Grupo diez: Rodrigo Estay

El proyecto trata de un programa que maneje las colisiones de figuras a velocidades constantes, se detectarán las colisiones entre figuras y se llevará un contador con respecto a cuantas colisiones han ocurrido. Las figuras serán círculos, cuadrados y triángulos equiláteros rojos y se considerará una colisión entre dos figuras cuando sus píxeles se toquen, una vez colisionado se aumentará el contador y las figuras rebotarán una con la otra y se pondrán de color azul, y luego desaparecerán. Las posiciones y velocidades iniciales de las figuras se definirán de maneras distintas dependiendo del modo escogido por el usuario, pero siempre se moverán en filas y columnas con movimientos horizontales y verticales respectivamente. Los modos son:

- **Modo Uniforme:** Las posiciones iniciales son definidas en filas y columnas predefinidas, las cuales estarán separadas a 100 píxeles de sus vecinas, mientras que las velocidades son definidas de manera aleatoria.
- **Modo Aleatorio:** Las posiciones iniciales serán completamente aleatorias pero siempre en los bordes de la ventana, mientras que las velocidades son definidas, al igual que en el modo anterior, de manera aleatoria.
- **Definir Posiciones:** Las posiciones son definidas por el usuario, el usuario podrá hacer click en el panel para crear figuras las cuales se moverán de manera horizontal o vertical dependiendo de cuál haya escogido el usuario mediante sus respectivos botones. También el usuario dispondrá de un slider que le permitirá definir la velocidad de la figura, en este slider tendrá la velocidad mínima a la izquierda y la máxima a la derecha, ya que a la velocidad mínima las figuras no se moverán, si se selecciona se definirán las velocidades de manera aleatoria y no será cero.

Diagrama UML



Propósito de las clases principales

Proyecto Diez: Esta es la clase principal la cual, como se observa en el diagrama, cumple la función de la creación de la ventana y de los controles con los cuales interactúa el usuario, dichos controles están conformados de JPanels y de botones y sliders creados a través de las clases internas: BotonesControles, BotonModo, SlideVelocidad.

PanelDibujo: su utilidad es dibujar las figuras, es el panel de fondo blanco que está arriba de los controles y que cubre la mayoría de la interfaz del programa. Esta clase además cumple el rol de

observar donde hace click el usuario en su panel para poder ubicar figuras en el modo “Definir Posiciones”

Manejador: Se encarga de la definición de velocidades y posiciones iniciales de cada figura creada, ya que esta clase maneja los valores aleatorios, y también detecta si es que ha ocurrido una colisión entre dos figuras. Además maneja el reloj interno del programa, por lo tanto cuando se hace click en “play” o “pause”, esta es la clase que empieza o detiene la simulación, mientras que su clase interna EventoTiempo se encarga de los eventos que están ligados al tiempo, como iniciar la creación de las figuras en los dos primeros modos, el indicar que deben moverse y también llama a que se vuelvan a dibujar las figuras, ya que EventoTiempo tiene una referencia a PanelDibujo.

AlmacenFiguras: Como lo dice su nombre, esta clase se dedica a almacenar cada figura creada, de esta manera Manejador no se complica con tratar las figuras de manera específica tan seguido, ya que con AlmacenFiguras las trata de una manera general. Esta clase crea las figuras, pero no define sus propiedades, ya que estas las define Manejador, y también puede eliminarlas.

Figura: Son las distintas figuras que se ven en el programa, esta clase solo se encarga de moverse a sí mismo respecto a su velocidad y de rebotar y ponerse azul una vez que haya colisionado. Figura, por otro lado, es capaz además de auto-eliminarse, ya que una vez que colisionó tiene una cierta cantidad restante de vida, la cual se va agotando a medida que se mueve, una vez agotado informa de que debe eliminarse. Las subclases solo definen las distintas formas de pintarse de momento.

Cambios respecto a la primera iteración:

Además del gran avance con respecto a la primera iteración, como la implementación de funcionamiento de métodos declarados anteriormente, ventanas, paneles y botones, también se hicieron cambios de capacidades de los objetos, ya que se agregaron capacidades que no se habían declarado anteriormente, porque a medida que se avanzaba en el proyecto se notó la necesidad de éstas, de la misma manera se eliminaron capacidades que se consideraron innecesarias. Pero había métodos declarados en algunas clases que convenían que estuvieran presentes en clases diferentes, así que algunos métodos se reubicaron a distintas clase.

Por otro lado se agregaron nuevas clases, como por ejemplo antes se había definido Manejador como la clase que se encargaba del movimiento de las figuras y además era el panel donde se dibujaban, ahora se le asigna la labor de panel a una nueva clase llamada PanelDibujo para separar la parte de interfaz con la lógica. Otras clases nuevas son las subclases de Figura, que de momento solo dibujan distintas figuras (Anteriormente se tenía previsto que iban a ser sólo círculos).