## **Description & Features**

The goal of Helicopter Madness is to last as long as possible without crashing into obstacles or hitting the top or bottom of the game window. The speed at which you travel steadily increases as time progresses making it harder to dodge the obstacles. Score is based on the distance you are able to travel before crashing. While in play gravity is constantly trying to crash you into the ground and you have to counter the gravity by lifting the helicopter.

There is a high score system that keeps track of the top 5 scores. If the user beats one of these scores the high score screen will come up and they will be asked to enter a name after they crash. Once they have entered and confirmed their name they will have ability to jump right back into a new game.

## **Classes**

**SharedSettings**: Contains common values that are used throughout the game.

**InputHelper**: Static extension / helper methods for dealing with user input.

**TextDisplay**: Displays a message on screen.

**FlashingTextDisplay**: Displays text on screen that flashes on and off. It is hidden for half the amount of time it is shown to improve readability.

**Sprite**: Abstract class for any textured sprite. Simplifies creating new sprite classes by handling the SpriteBatch, texture, and position. Also has an implementation of draw which has the basic functionality required by most sprites.

**ScreenLoopSprite**: A sprite that continuously loops from left to right on the screen in a seamless fashion by drawing two copies of the texture front to end.

**AnimatedSprite**: An abstract class for a sprite that has animation frames. This class handles creating the rectangles for each frame of the animation, looping through the frames, and drawing the current frame.

**GameScene**: An abstract class inherited by all Scenes in the game. It handles the drawing and updating of scene components. It maintains two lists of the added components, one of drawable components and one of updateable components to reduce the work required when updating or drawing.

**HelicopterGame**: The game class. It creates the SceneManager and sets the resolution to 1280x720.

**SceneManager**: Creates all of the scenes and manages which scene is currently active. It updates and draws the active scene. It also responds to the state of certain scenes. For example if the ActionScene is in its game over state and it has a score higher than any of the high scores, it switches to the HighScoreScene which goes into a name entry state.

**MenuScene**: The menu scene for the game. Allows users to select a scene by pressing up or down to move the selection and enter to confirm the selection. Notified SceneManager when a menu selection is made.

**MenuComponent**: Handles the menu scene input and draws the menu entries.

**HowToPlayScene**: Displays information about the controls for the game. Draws a single sprite with an image showing the controls as the texture.

**HelpScene**: Displays a description of the game. Draws a single sprite with an image showing the description as the texture.

**CreditScene**: Displays the credits for the game including references to the sounds used. Draws a single sprite with an image showing the credits as the texture.

**HighScoreScene**: The high score scene for the game. Displays the top 5 high scores and allows users to  
 enter a 3 letter name if they get into the top 5 scores.

**HighScoreEntry**: Represents the score and player name for an entry on the high score scene.

**ActionScene**: The playable scene for the game. It creates the helicopter, the borders, and all of the obstacles. It repositions and re-enables the obstacles when they disable themselves by going off the left of the screen. The current games score, which is managed by the ActionScene, is displayed on the top left and the highest score is displayed on the top right.

**Obstacle**: A collidable sprite for the player to avoid that scrolls from right to left and disables itself when it goes off the left of the screen. It plays a sound when it is involved in a collision.

**Helicopter**: The helicopter that the player controls. It is always falling unless the player is using one of   
the input keys in which case it starts to rise. It can collide with obstacles and the borders. When it collides, it starts the explosions animation and stops playing the helicopter sound.

**Border**: A collidable sprite for the scrolling top and bottom borders of the action scene. It plays a sound when it is involved with a collision.

**Explosion**: An animated explosion that plays when the helicopter crashes. It inherits from AnimatedSprite but adds a delay to the frame change and sets the origin of the frame to the frames center.

**CollisionManager**: Checks for collisions between the helicopter and obstacles or borders. Notifies the   
colliding objects of their collision so they can react to it.

**ICollidable** – Interface: Interface with methods for getting an objects bounds and for notifying the object   
of a collision.

## **Class Diagram**



**Zoom in  
 or   
See the UMLModels project in the game solution for higher resolution class diagram**

**Resources**

### Textures

* Background Texture: <http://pdtextures.blogspot.ca/2008/03/twentieth-set.html#BLOGGER_PHOTO_ID_5177196950156455650>
* Border Texture: <http://pdtextures.blogspot.ca/2008/03/seventh-set.html#BLOGGER_PHOTO_ID_5177185259255474882>
* Obstacle Texture: <http://pdtextures.blogspot.ca/2008/03/seventh-set.html#BLOGGER_PHOTO_ID_5177185250665540258>

### Sound & Music

* Border Collision Sound: <http://www.freesound.org/people/severaltimes/sounds/242859/>
* Helicopter Sound: <http://www.freesound.org/people/lezaarth/sounds/232818/>
* Obstacle Collision Sound: <http://www.freesound.org/people/DJ%20Chronos/sounds/123226/>
* Invalid Key Press Sound: <http://www.freesound.org/people/Splashdust/sounds/67454/>
* Menu Selection Change Sound: [http://www.freesound.org/people/broumbroum/sounds/50561](http://www.freesound.org/people/broumbroum/sounds/50561/)
* Back To Menu Sound: <http://www.freesound.org/people/broumbroum/sounds/50557/>
* Menu Selection Sound: <http://www.freesound.org/people/broumbroum/sounds/50565/>
* Beat Highest Score Sound: <http://www.freesound.org/people/Kastenfrosch/sounds/162476/>
* New High Score Sound: <http://www.freesound.org/people/plasterbrain/sounds/242855/>
* Music: <http://incompetech.com/music/royalty-free/index.html?isrc=USUAN1400046>

### Control Images

* Keyboard Image: <http://www.informationq.com/wp-content/uploads/2013/10/computer-keyboard-layout.jpg>
* Mouse Image: <http://www.wpclipart.com/computer/mouse/computer_mouse.png>