**Space Invaders Game**

*Marmoud Zakaria F5XWUZ*

*-*

*AbuKhatro Mahmoud D6UWHX*

*-*

Objectives:

1. Create a fun space invaders style game in WPF with C# programming

2. We will use nested loops in the timer objects to identify and instruct game objects

3. Create custom functions to enhance the efficiency of the program

4. Key board control along with moving left and right we can shoot bullets at enemies

5. Determine how to win or lose the game

6. Change enemy speed during run time

7. Shoot back at player object with random bullets to make the game more interesting and challenging

8. Use dispatcher timer to create the main game engine with all the rules and logic

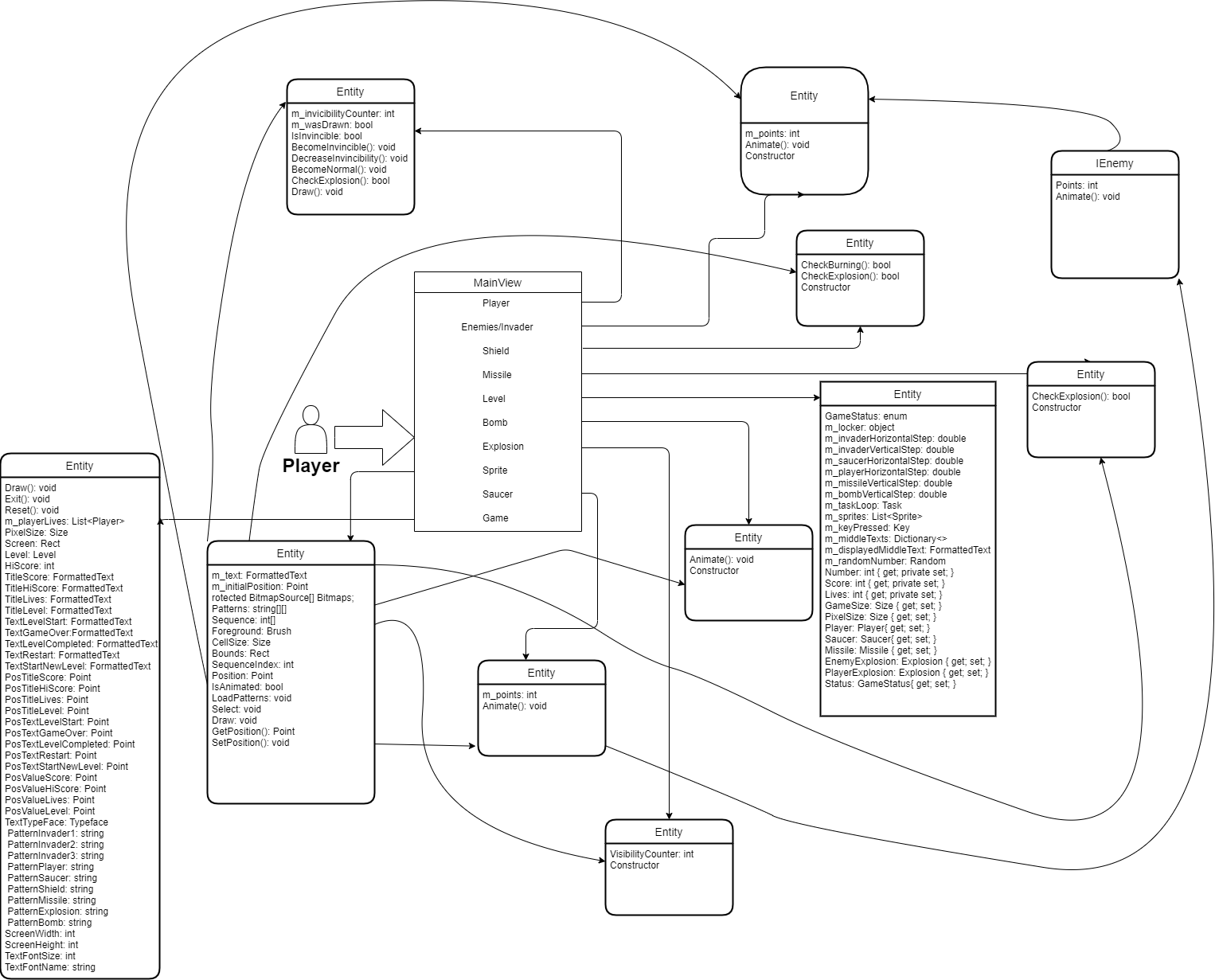
**Playing the Game**

This version is played using the left and right arrow keys to move the man and the space bar to fire. If you hit an alien, you get 10, 20, or 30 points, depending on which one you hit. If a bomb hits you, you die and you only have 3 lives. The shields will protect you from a bomb, so you can hide behind them while you wait for the bombs to pass. Occasionally, a saucer flies across the screen. If you hit it, you get a suprise score (up to 150 points).

A screenshot of a computer

Description automatically generated with medium confidence

**Design of the Game**The design for this game is shown below:



**What can a user do?**

* - Start the game
* - Full View Of The Game
* - To Start Press Enter
* Player on buttom mid
* - Shot Enemies With SpaceBar
* - Buttom Left: Lives
* Buttom Right: Level
* - Top Are HighScores And Score
* - Avoid Getting Hit
* - Shield To Cover and Decaying
* - Lives get to 0 Game Over!

Figure 2 - Space Invader UML Design Reverse engineered using WithClass 2000

The Design basically has a base class **GameObject** which knows how to draw the game pieces bitmap.  Subclassed under this object is all the component classes of the game: **Player**, **Invader**, **Bullet,** **Game**, **Explosion**,  **Bomb**,  **Level, Missile , Saucer, Sprite** and **Shield**. A row of Invader classes is controlled through the InvaderRow class. The form contains an array of  InvaderRows, an array of Shields, and a Player.

**The Timer Event Handler**

The entire game is played off of the timer component. After each tick of the timer, the game checks to see if their are bomb collisions, bullet collisions, and whether or not to move the invaders. It also handles the key presses from the user here so as not to change the timing of the game. If a key is pressed, it is remembered by the key press event, but nothing more happens in the key press event handler.

**Painting the Game Components**The drawing of all the components is triggered by the Invalidate.

**Spinning Bombs with Graphic Paths and Drawing the Bomb.**