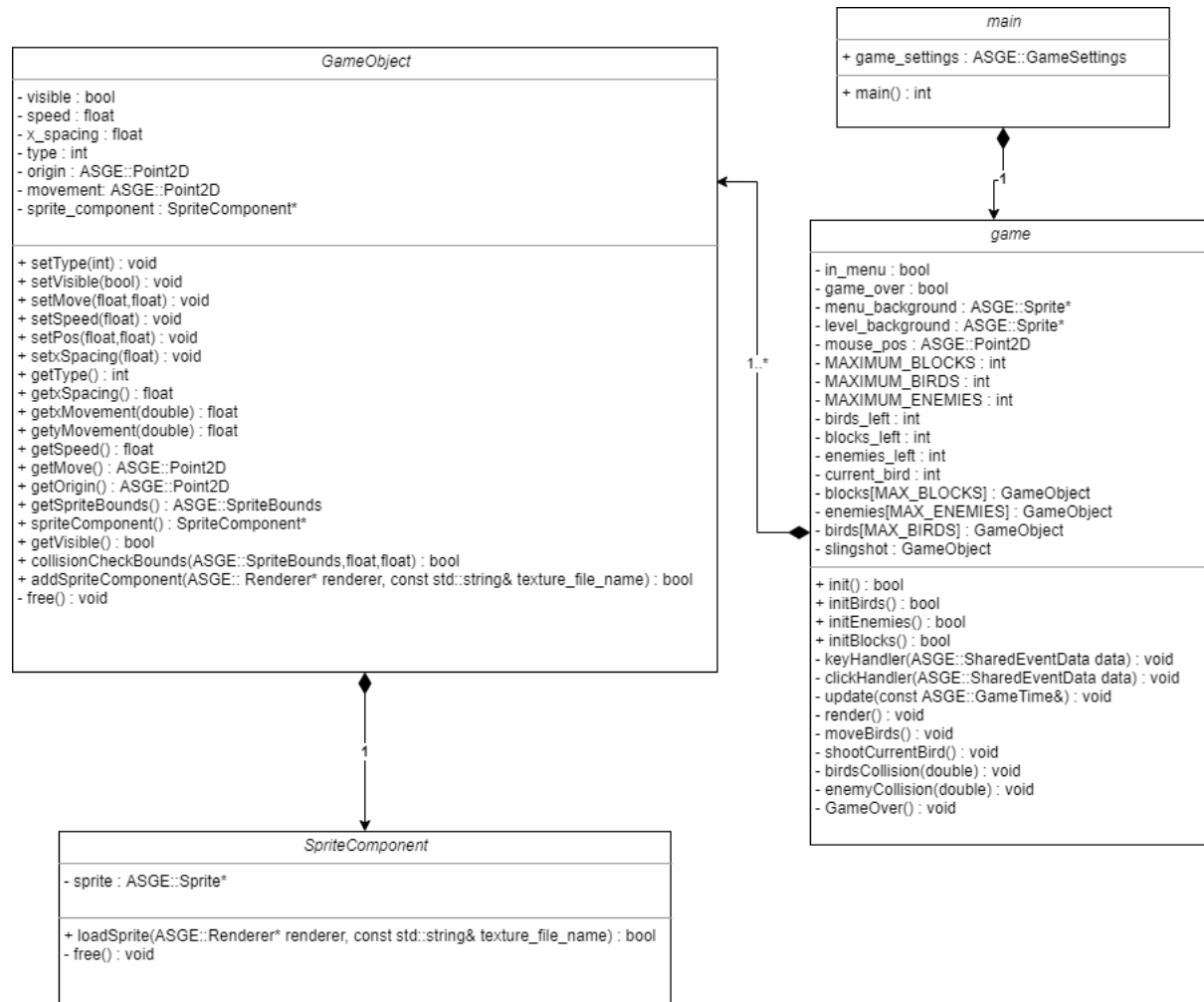
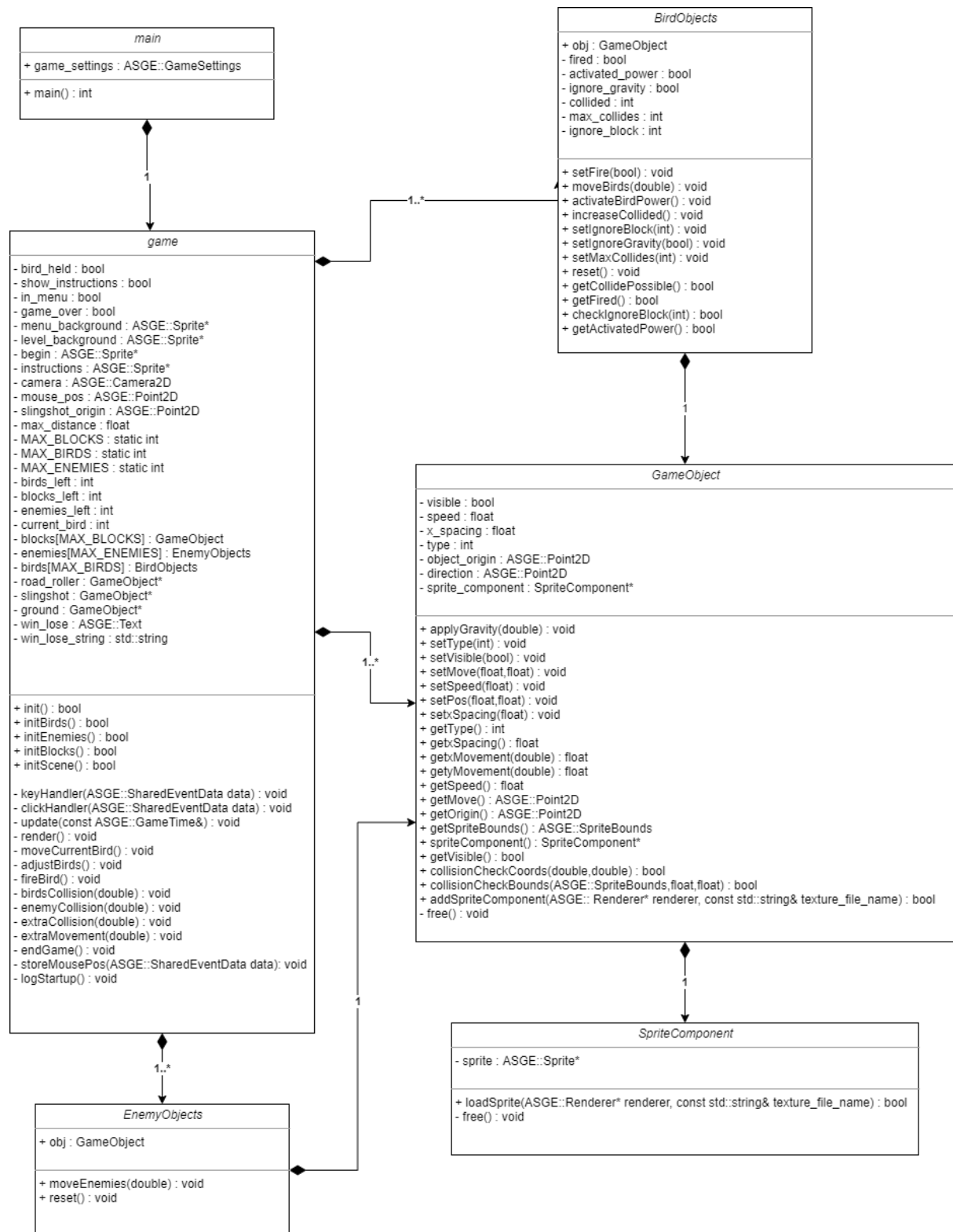


## Pre-Project UML Diagram



**Post-Project UML Diagram**

## **UML Diagrams Comparison**

### **GameObjects:**

Whilst planning the project there were a few areas that were not covered when making the first plan of the UML diagram. Most apparent is that I did not consider that objects would need a collision check that did not involve a sprite, so had to create a new function to cover mouse-based co-ordinate collision.

### **Additional Classes:**

Another area that I hadn't considered in my planning was the use of additional classes for both bird objects and enemy objects, as both would require specific functions. I feel that by implementing these classes in my project I made both more manageable and easier to understand, as I would otherwise have had to include a series of object specific functions in my GameObject class.

By giving the BirdObjects and EnemyObjects a GameObject component I can still utilise the foundations I set up with the GameObject whilst adding in the necessary functions for each object type without impacting other object types.

One other advantage is that this allowed me to easily add extra functions to the BirdObjects class without having to worry about setting default values for any new variables that were required for the additional features, such as activating individual powers and ignoring collisions for a specific number of objects.

### **Game Logic:**

When starting the project, I had planned for the bulk of my game's logic to take place in either the Update function or the MoveBird and BirdCollision functions in the game.cpp file. This would have been a terrible option as the logic behind the game would be very difficult to follow within those functions.

In the end I split the game logic into multiple functions across the different classes of my game. This made it a lot easier to pinpoint any bugs as I could work on an isolated feature with very little overlap into other features / components of the game.

### **Conclusion:**

Overall, I feel that my initial planning was not enough for the scope of the project, however this lesson has been learned and I will plan out individual features into greater depth going forwards.

As far as the finished project goes, I would say that I learned a lot throughout creating the game, such as correctly utilising individual classes for objects and setting up a more modular design to my code in general.