Minor Technical information

Code styling and practices

This is somewhat informal. It is for me to write down some of my coding practices. Created this document late, and I do not intend to spend a lot of time on this document.

## Internal vs Public classes

All my classes, assuming they are not inside other classes or concrete implementations of interfaces, will be public by default. As of writing this, I only have one class-within-class – State Machine Transitions

Classes which are concrete implementations of interfaces however – A State Machine State/Condition for example, will most likely be internal. Some other examples are AI controllers

My reasoning for this is that I feel like classes which are public are build “to be derived from”. By making them internal I make a conscious effort to show to the reader – be that myself or someone else – that if you want to make a class that is based off of this, you should instead inherit from the same interface. I could of course use ‘public Sealed’, but that is restricting it unnecessarily – you *can* inherit from the internal class, but you *should not*.

There are however some things that theoretically fill these criteria but are still public. A good example for this is the ‘Always’ State Machine Condition.