

De UML van de gameplay van mijn spel is opgedeeld in twee delen: de score en de gameplay. Dit heb ik bewust zo gescheiden mogelijk gehouden om ervoor te zorgen dat ik niet op een wazig moment tijdens het programmeren opeens de AI afhankelijk zie te maken van de score. Daarnaast heb ik de AI proberen te clusteren met hun manager. Deze manager heb ik een Singleton gemaakt om makkelijk vanuit de Ball aan de Manager door te geven dat de speler heeft geschoten, zodat de balls hun gedrag daarop kunnen aanpassen.

Ik dacht van te voren eigenlijk dat ik Object Pooling zou gaan gebruiken voor de particle effects die spawnen als de ball een paddle gebruikt. Dit heb ik uiteindelijk toch niet gedaan omdat dat nodeloos complex was voor een minieme snelheidsverbetering. Er zou namelijk op z'n meest maar 2 particle effect systems op hetzelfde moment actief kunnen zijn, en om voor 2 objects een objects pool te maken is niet echt nodig.

Helaas ben ik er niet aan toegekomen om een timer en rondes toe te voegen, omdat ik dit binnen de tijd niet op een uitbreidbare wijze kon ontwerpen. Dat zou ik de volgende keer graag anders zien.