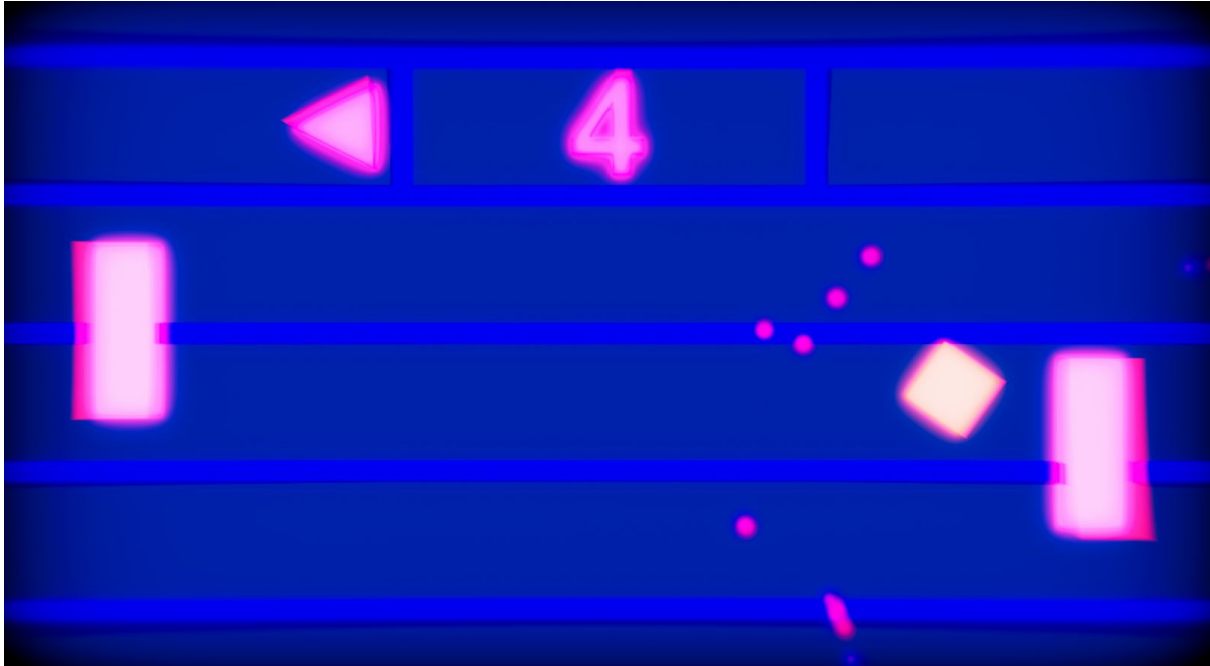


DOCUMENTATIE PINGPONGBADINGBADONG

ROEL ZWAKMAN - 20 JUNI 2018



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. Daarnaast zit er een Object Pooling-systeem in voor de particle effects, die overigens ook een Singleton is om deze makkelijk aanspreekbaar te maken in code, en om zo een wat flexibelere structuur te hebben.

Ik ben door het project heen van mening gewisseld over Object Pooling. Eerst wilde ik het, en toen vond ik het eigenlijk niet nodig. Nu bleek bij de feedback dat dit absoluut wel nodig was, dus dat is nu geïmplementeerd. Ik snap overigens inmiddels wel waarom, omdat je als oefening soms nou eenmaal niet 100% een real-world scenario kan volgen. Het object-pooling systeem is ook een singleton, om het makkelijk aanspreekbaar te maken voor verschillende dingen.

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. Ook zou ik tijdens een volgend project wat minder koppig willen zijn in de manier waarop ik programmeer, om zo mezelf niet te saboteren qua schoolwerk.