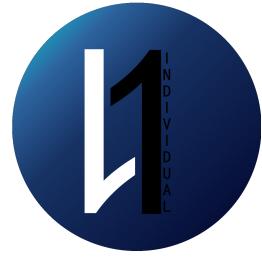


Cem Suzen - IndividualGames

suzen.cem@yandex.com

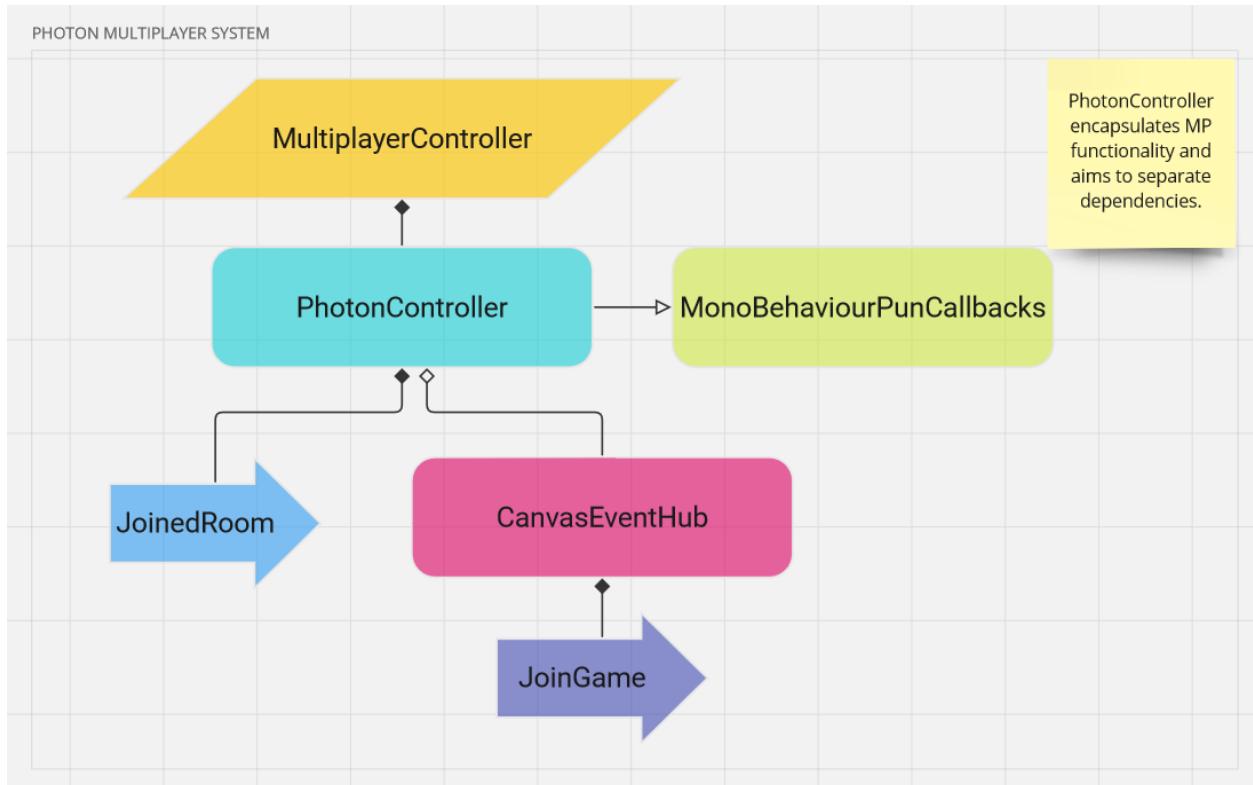


Happy Hour Strategy Case - 27/8/2023

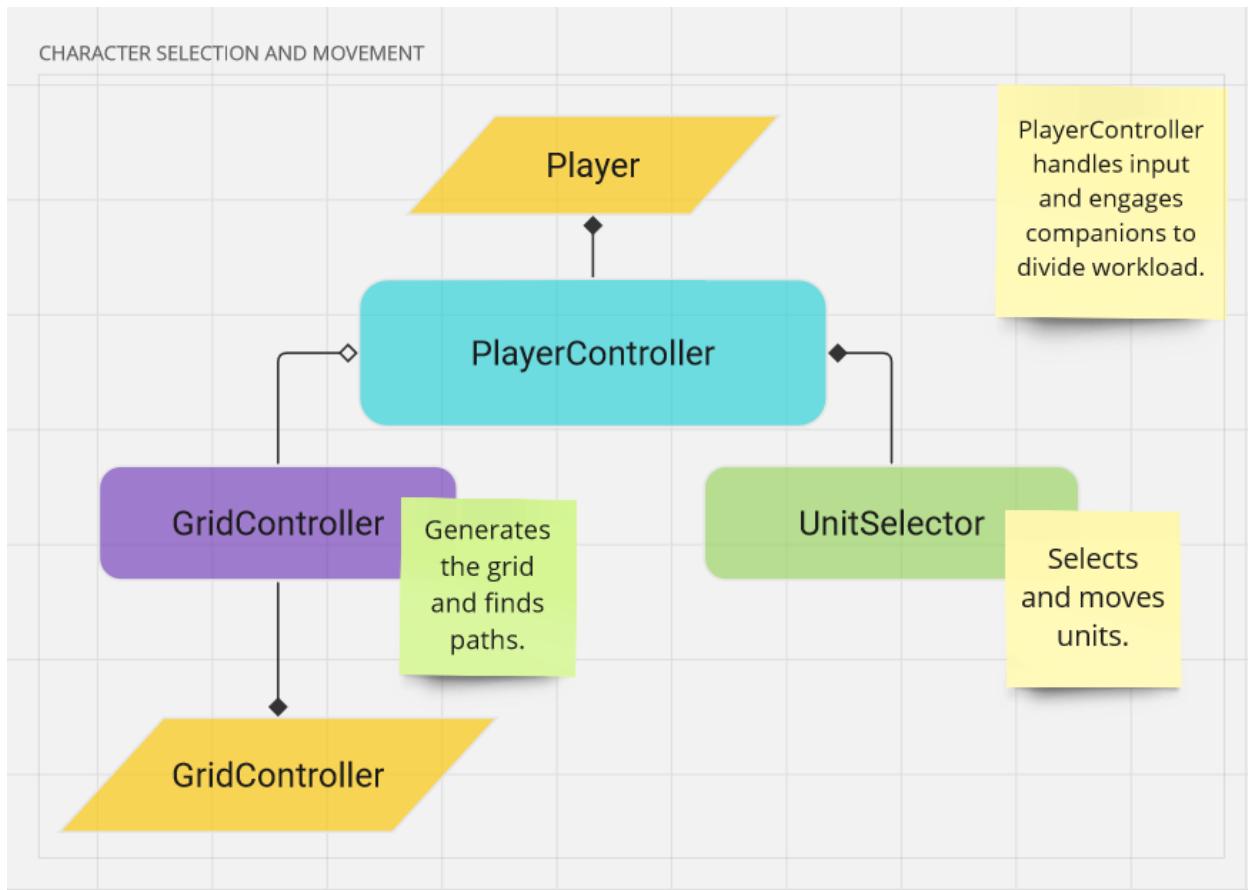
Code Structure:

*Meta of the diagrams provided below.

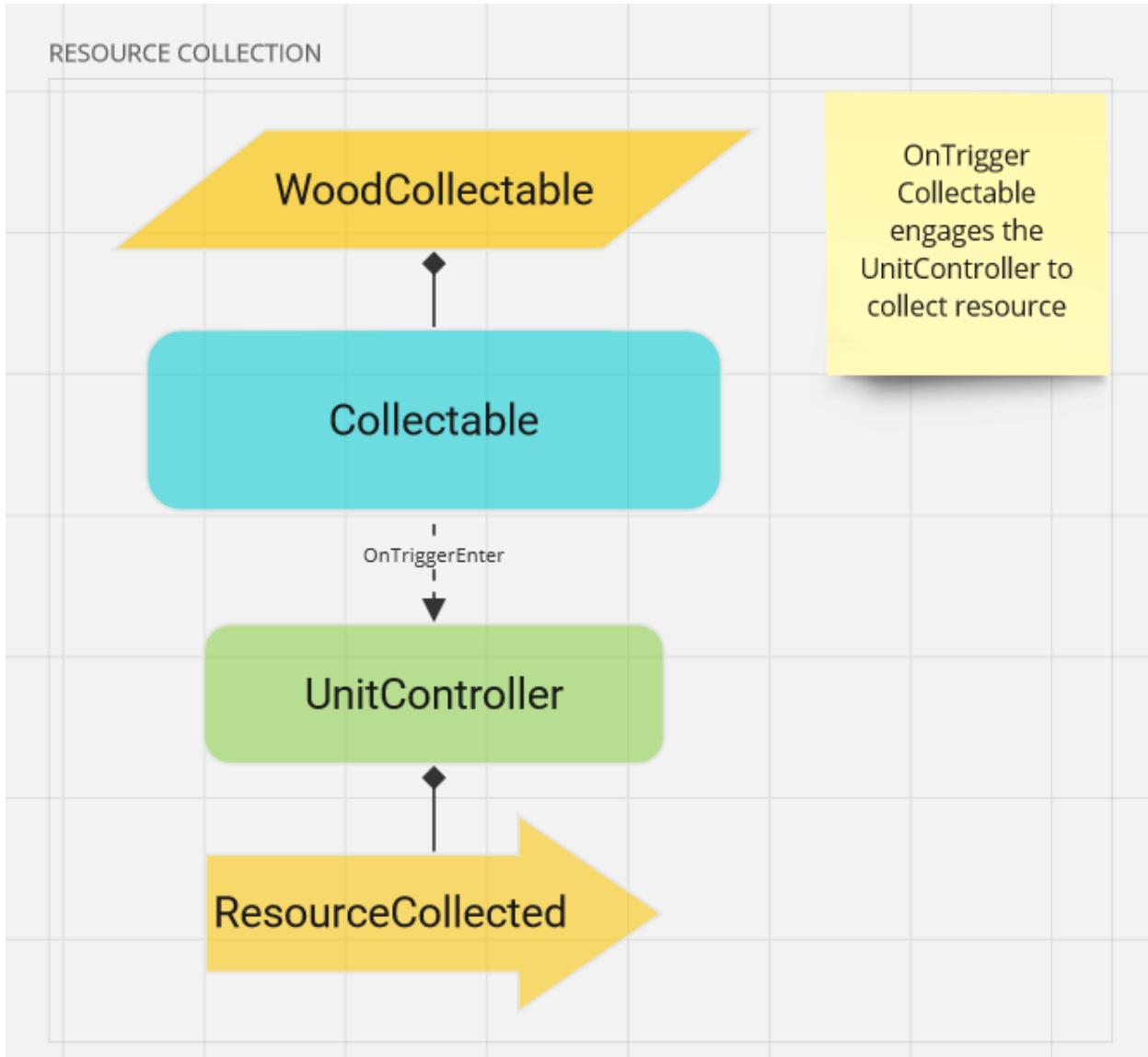
Multiplayer Functionality:



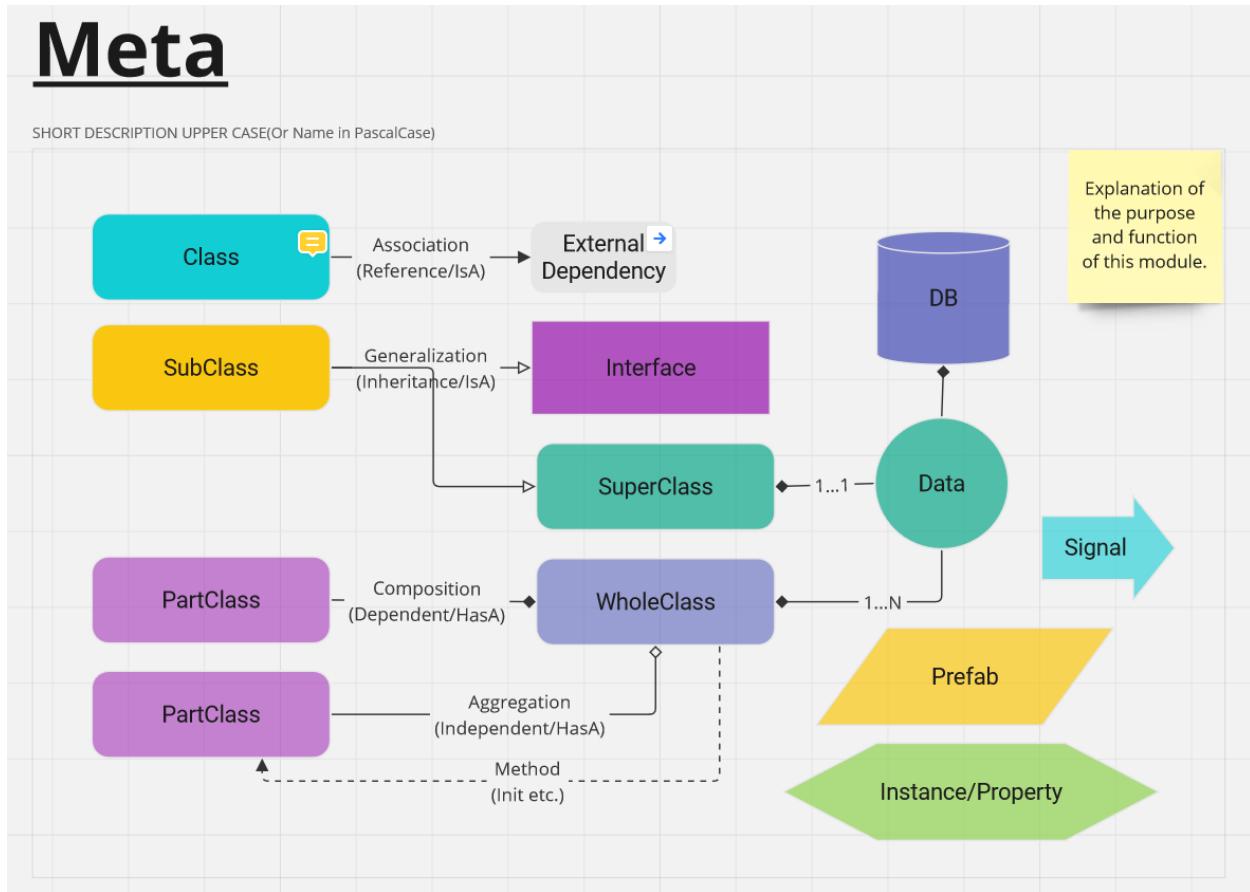
Character Selection and Movement:



Resource Collection:



Meta:

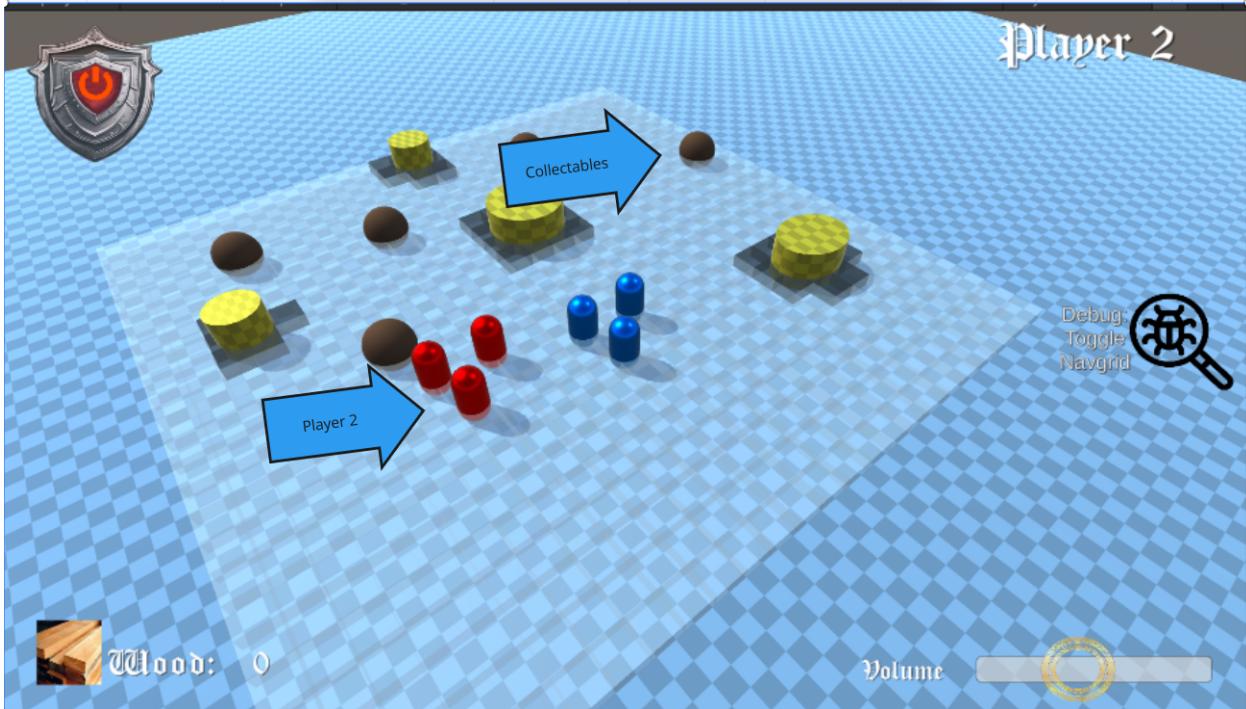
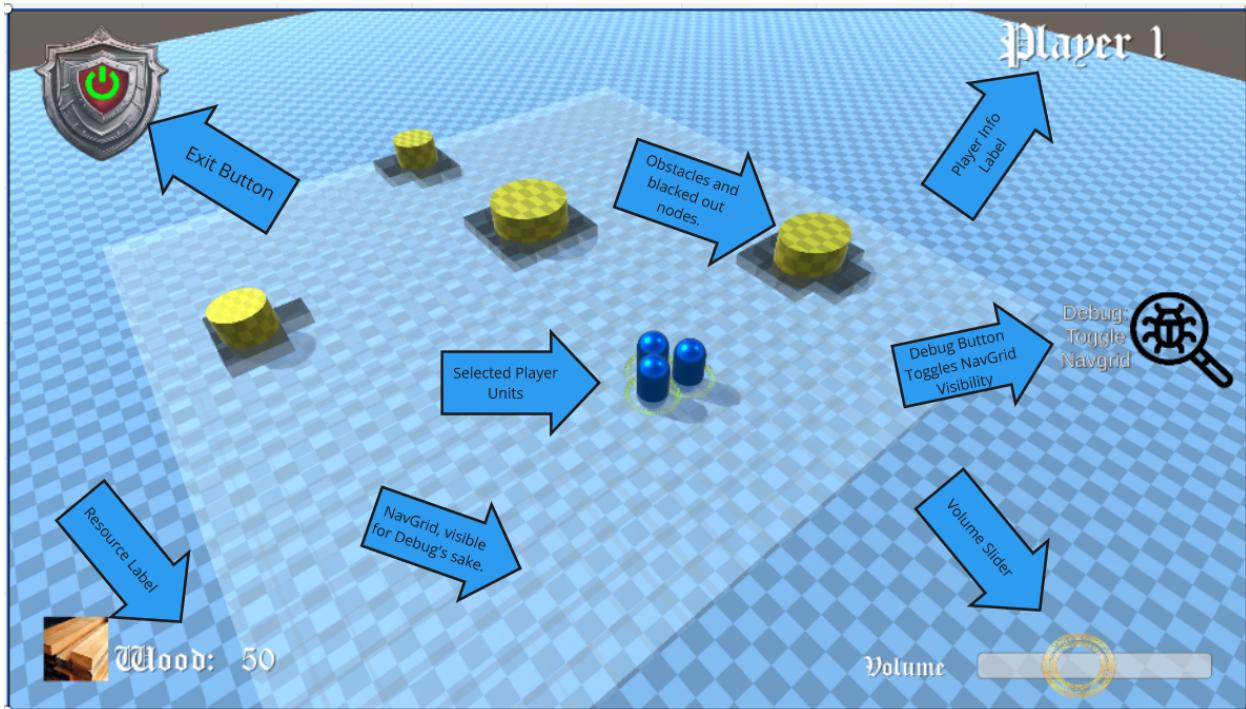


Interface:

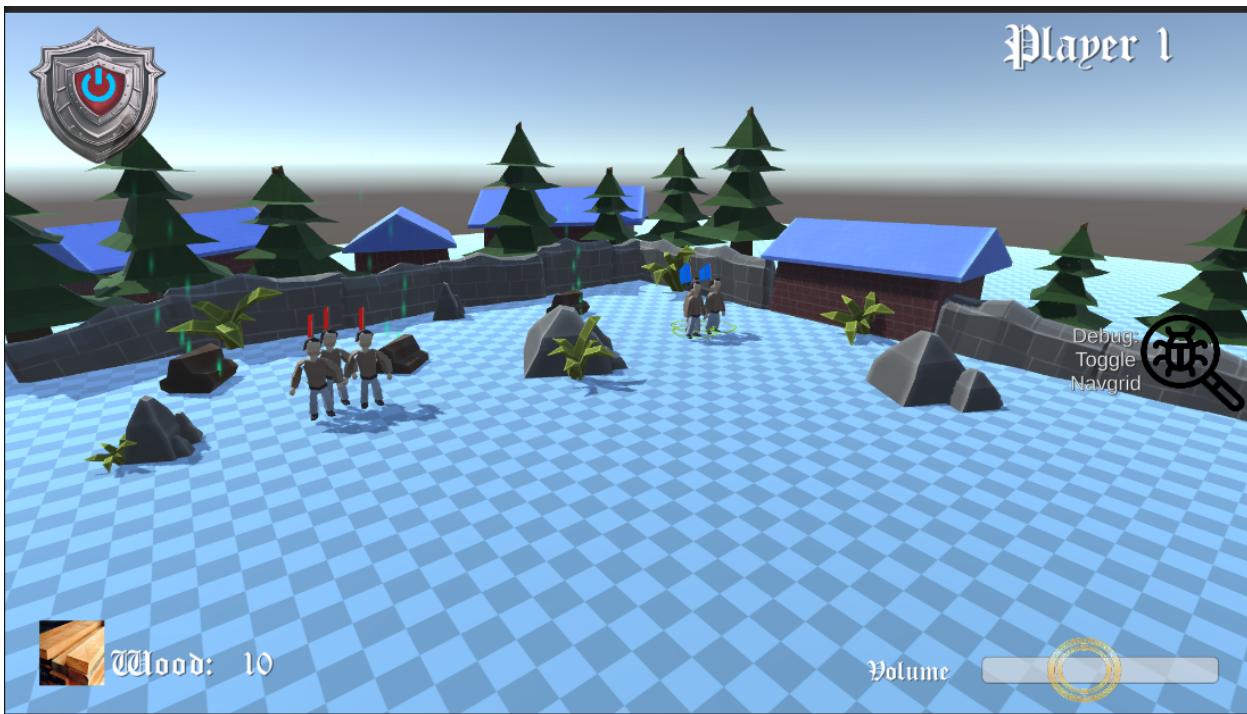
Main Menu



Level1 - Whiteboxing



Level1 - Final



Methodologies Used

SOLID, KISS, OOP, FP, DESIGN PATTERNS, Clean Code, Modern C#

SOLID is used to divide responsibilities, keep manageable interfaces, use DI where applicable, keep codes extensible and upgradeable.

KISS keeps things simple without clutter code and performs precise functionality.

OOP provides inner dependencies that are separating responsibilities, making code more reusable and modular.

FP(Functional Programming) is used to generate functional results rather than depending on other structures. This keeps OOP approaches simple and separated.

Design Patterns were used where applicable.

Clean coding is performed by logical coding, ideally we want to read the code without reading code but reading logic. Naming and structuring are the basis for it.

Using Modern C# wherever possible to clean the code with easier to read and better controlled code.

Credits

3D art made by me.

Music, 2D icons, fonts and other assets to their owners.

Video Demonstration:

https://youtu.be/h_FRdw3YtZ4

Thanks and feel free to ask any questions about the case.