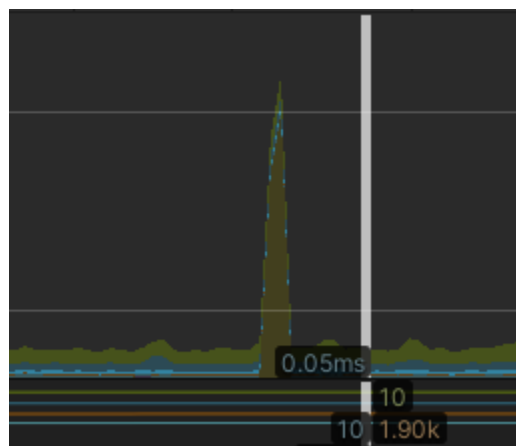
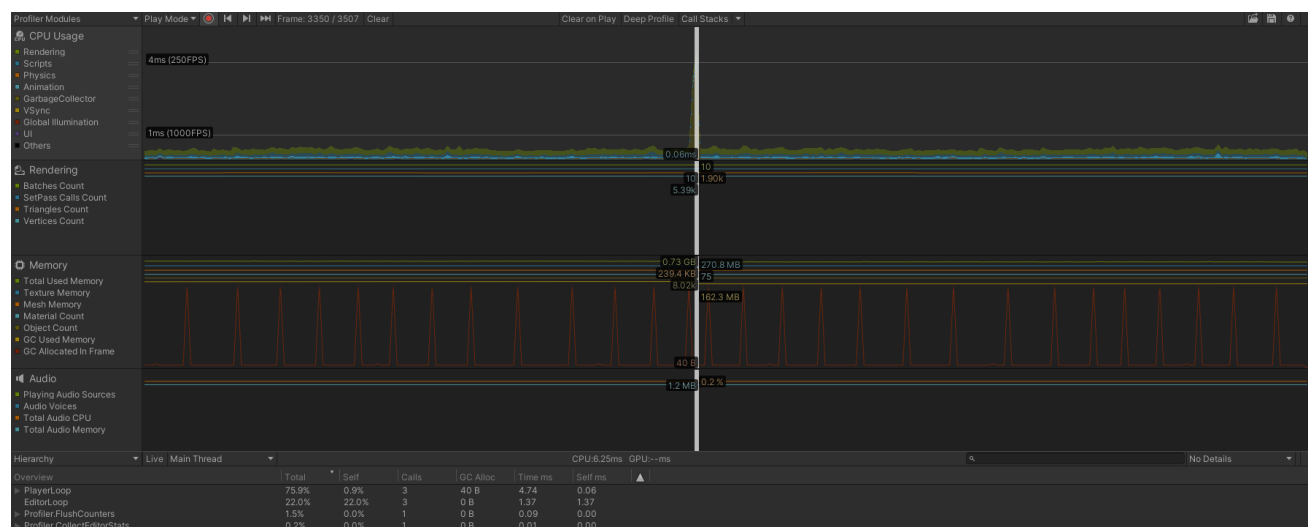


Gra która będzie sprawdzana pod kątem wydajności to gra multiplayer.

Testowanie gry w stanie spoczynku za pomocą profilera.

Połączenie z jednym graczem:

Jak można zauważyć co pewien czas występuje problem z dość wysokim spadkiem wydajności



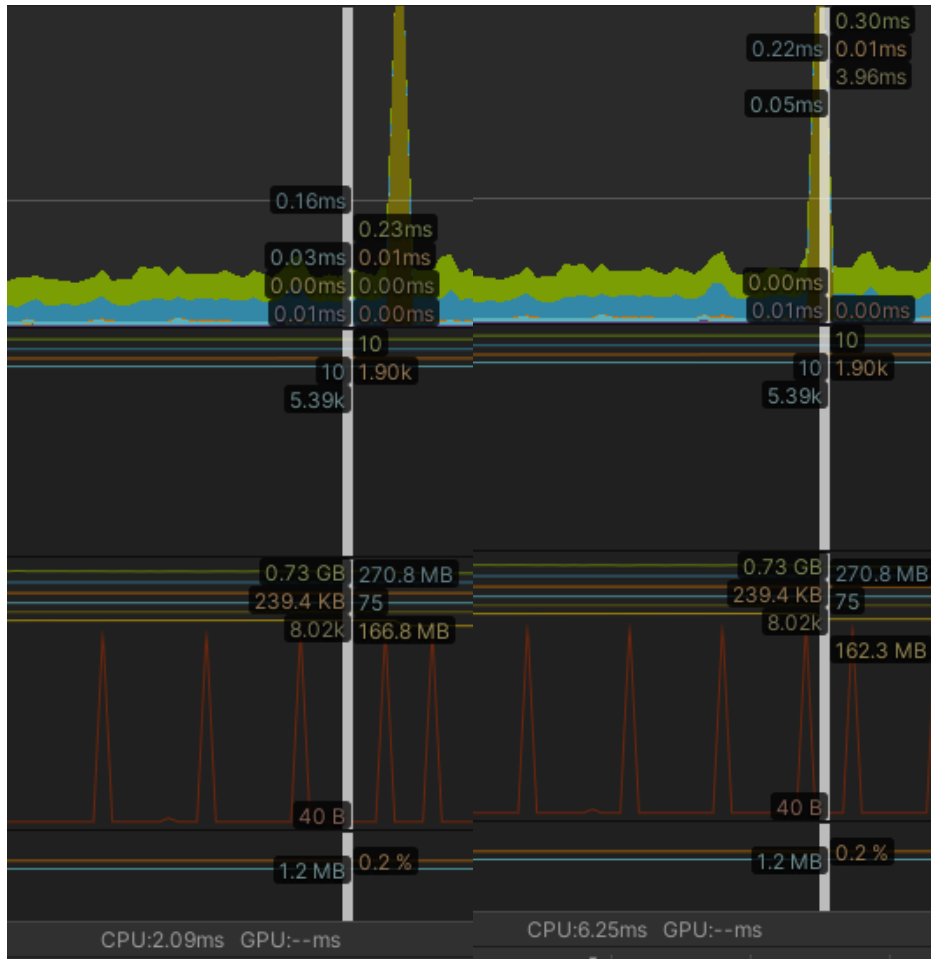
Widać zdecydowanie że błąd leży po stronie playerLoop czyli problem ze strony jakiegoś skryptu. W tym przypadku to nadużycie Gabage collector, który w tej klatce alokuje 40B.

Hierarchy	Live	Main Thread				CPU:6.25ms	GPU:-
Overview	Total	Self	Calls	GC Alloc	Time ms	Self ms	
▼ PlayerLoop	75.9%	0.9%	3	40 B	4.74	0.06	▲
▼ TimeUpdate.WaitForLastPresentationAndUpdateTime	63.4%	0.0%	1	0 B	3.96	0.00	
▼ GarbageCollector.CollectIncremental	63.4%	26.0%	1	0 B	3.95	1.62	
GC.Collect	37.3%	37.3%	1	0 B	2.33	2.33	

Zwiększa się również zużycie czasu wykonywania akcji procesora.

Zwykła klatka:

W miejscu problemu:



Spadek wynosi około 750FPS to dość sporo. Czas alokowania pamięci wynosi około 5ms.

Rozwiązanie: Niestety nie wiem

Procent wykonywania metod Update w danej klatce :

Wynosi 1.6%

Update.ScriptRunBehaviourUpdate	1.6%	0.0%	1	0 B	0.10	0.00
BehaviourUpdate	1.6%	0.1%	1	0 B	0.10	0.01

W FixedUpdate:

Brak

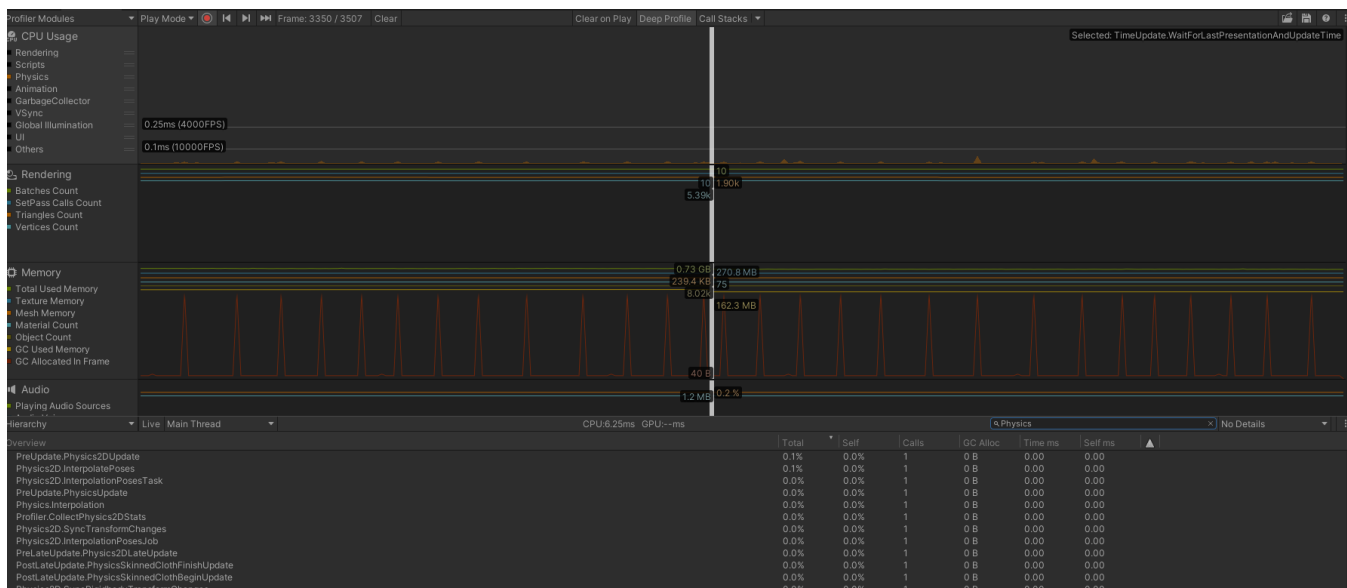
Render UI wynosi bardzo mało ponieważ występują tylko 2 teksty na scenie



Hierarchy	Live	Main Thread	CPU:6.25ms GPU:~ms				
Overview	Total	Self	Calls	GC Alloc	Time ms	Self ms	
UIEvents.WillRenderCanvases	0.2%	0.0%	1	0 B	0.01	0.00	
UGUI.Rendering.UpdateBatches	0.2%	0.0%	1	0 B	0.01	0.00	
Profiler.CollectUIStats	0.1%	0.1%	1	0 B	0.01	0.01	
UGUI.Rendering.RenderOverlays	0.1%	0.0%	1	0 B	0.01	0.00	
UGUI.Rendering.EmitWorldScreenspaceCameraGeometry	0.0%	0.0%	1	0 B	0.00	0.00	
GUI.Repaint	0.0%	0.0%	1	0 B	0.00	0.00	
UIEvents.UIElementsRegisterRenderers	0.0%	0.0%	1	0 B	0.00	0.00	
UIEvents.UpdateCanvasRectTransform	0.0%	0.0%	1	0 B	0.00	0.00	
GUIUtility.SetSkin() [Invoke]	0.0%	0.0%	1	0 B	0.00	0.00	
PostLateUpdate.GUI.ClearEvents	0.0%	0.0%	1	0 B	0.00	0.00	
UIElements.RuntimeUtility.Native.RepaintOverlayPanels() [Invok	0.0%	0.0%	1	0 B	0.00	0.00	
UIEvents.CanvasManager.EmitOffScreenGeometry	0.0%	0.0%	1	0 B	0.00	0.00	

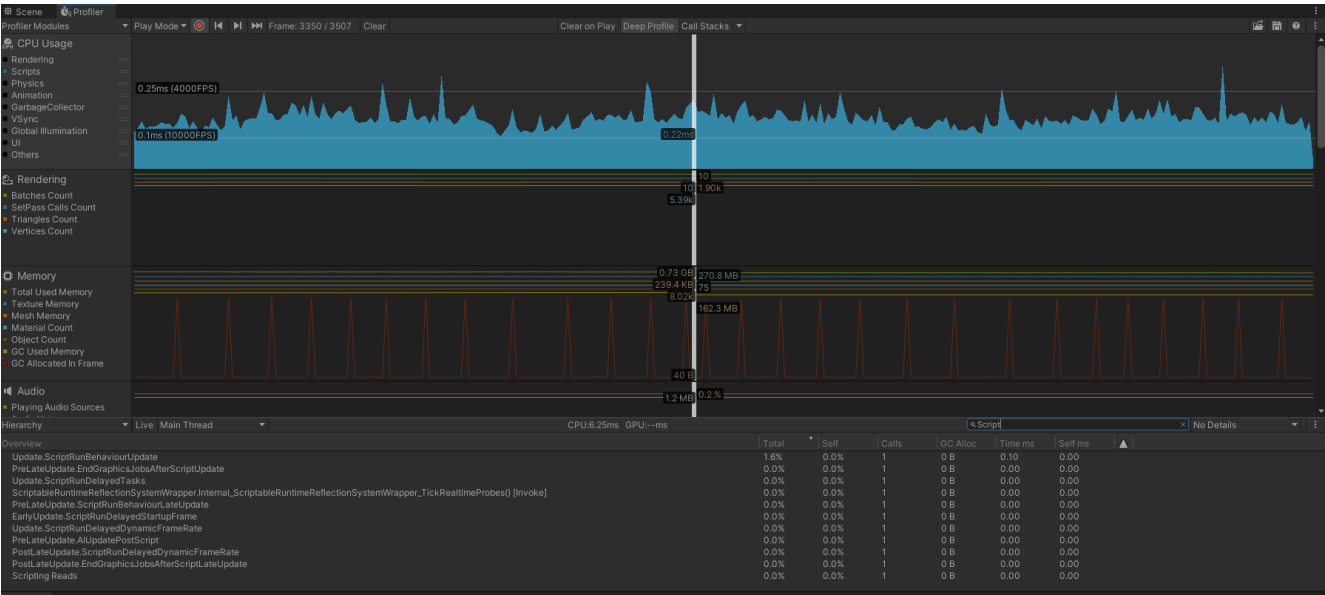
Fizyka:

Brak



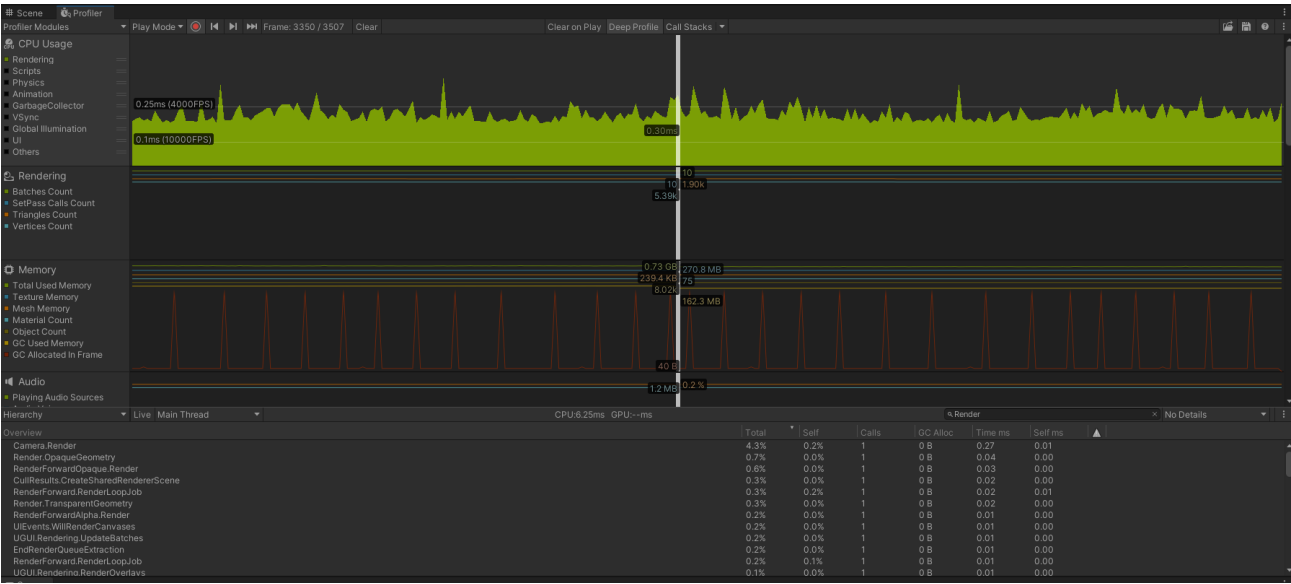
Skrypty:

Okolo 1.6%



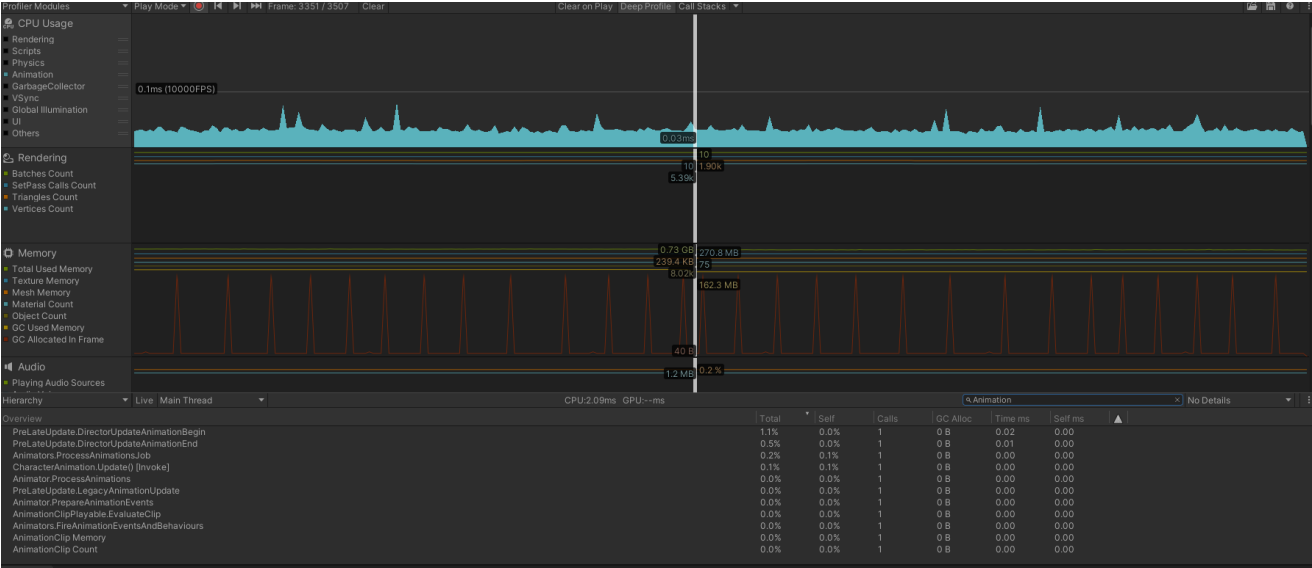
Render:

Wynosi w granicach 10 – 20%



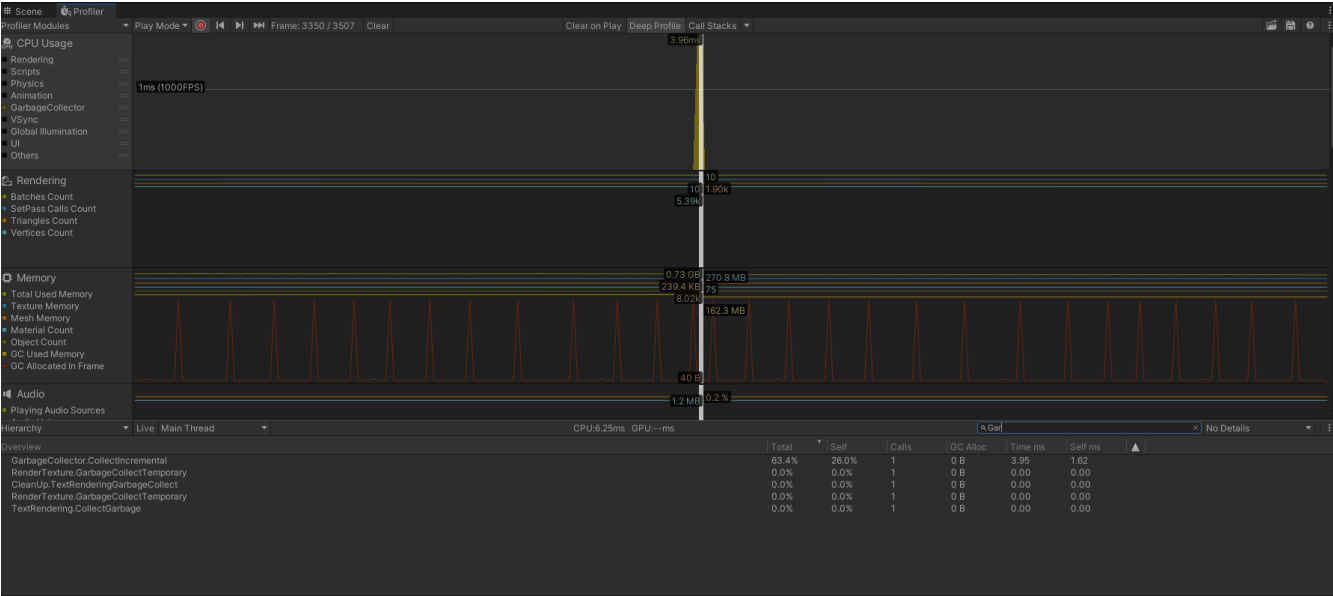
Animacje:

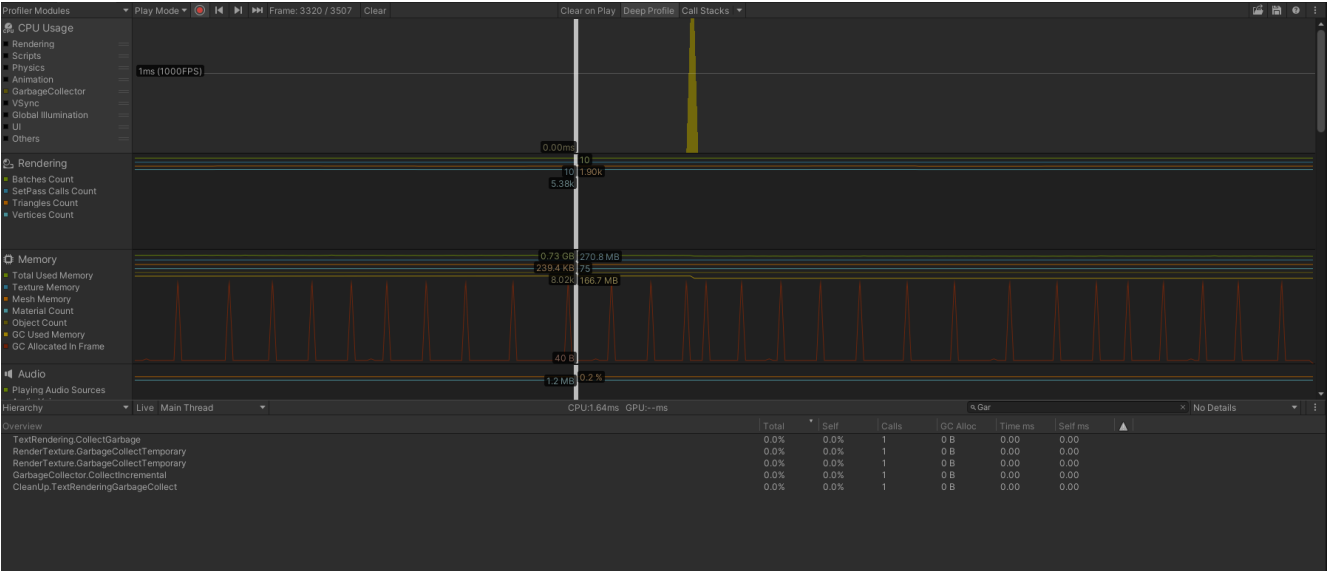
Animacje około 3%



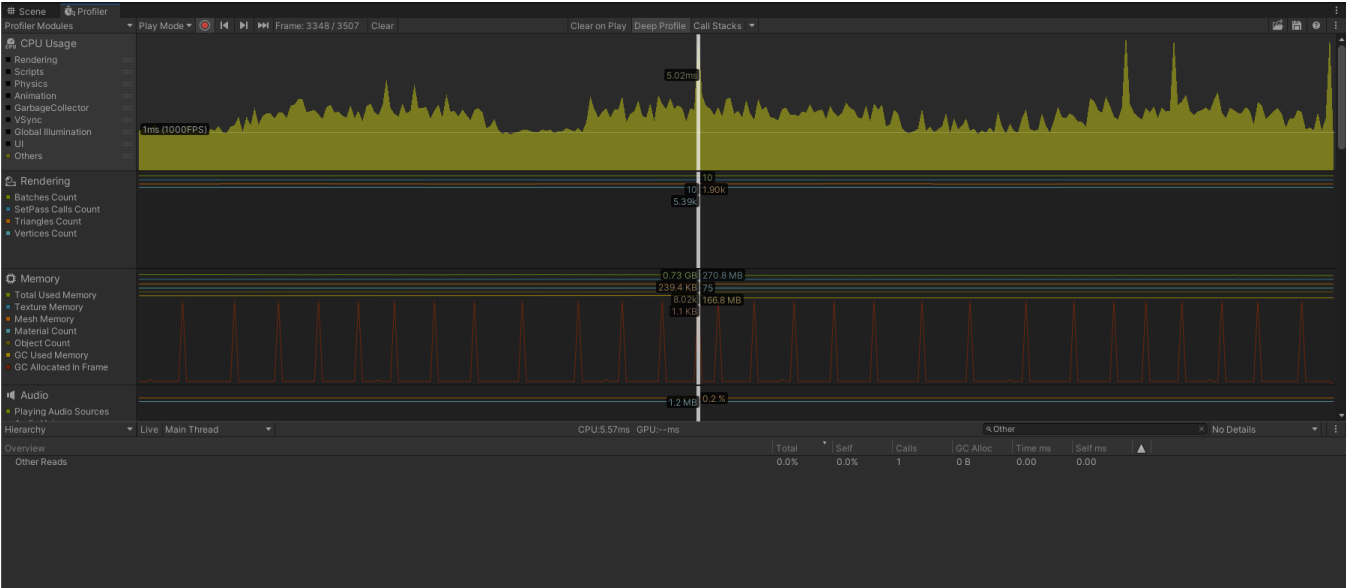
GC:

W danej klatce która powoduje spadek raz na jakiś czas około 60%. A tak to 0%.

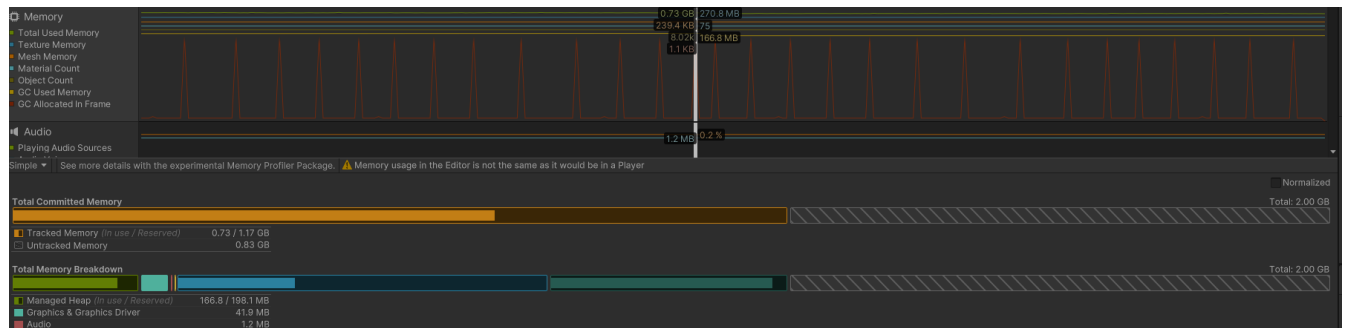




Others:



Pamięć:



Z racji że gra nie zawiera dużo elementów do renderowania tylko kilka testowych elementów, niema zbytnio co sprawdzać w frame debuggerze.

