

## Wprowadzenie do silnika Unity

### Informacje organizacyjne.

Poniżej została przedstawiona instrukcja wystarczająca do realizacji laboratorium. Czas przewidziany do realizacji laboratorium wynosi 2 tygodnie od momentu umieszczenia instrukcji na platformie oraz wysłania na platformie zdalnej edukacji informacji przez prowadzącego. Każdy tydzień opóźnienia będzie skutkował odjęciem 0,5 punktu. Laboratorium może być realizowane przez maksymalnie dwie osoby. Jako rezultat wykonania laboratorium należy umieścić na platformie zdalnej edukacji filmik przedstawiający efekt końcowy prac i jeśli pozwoli na to platforma (w zależności od rozmiaru efektu finalnego) również spakowanego rozwiązania z silnika Unity (wersja opcjonalna). Każda osoba z podsekcji powinna umieścić rezultat wykonanej pracy na platformie. Film powinien być podpisany nazwiskami każdej z osób realizującej laboratorium np. Nazwisko1\_Nazwisko2.avi.

Składowe oceny:

- 3 – Stworzenie terenu (wzniesienia uniemożliwiające graczowi wypadnięcie z planszy, woda, dodanie przeszkód innych niż wbudowane prymitywy np. drzewa);
- 4 – Dodanie kontrolera postaci oraz dodanie punktu zakończenia gry;
- 5 – Poprawne podpięcie skryptów;

Istnieje pełna dowolność formy wizualnej budowanego świata, jednak podstawowe zasady funkcjonowania gry powinny być zachowane. W razie większych problemów w realizacji laboratorium proszę o kontakt mailowy. Jeśli pojawi się więcej pytań lub wątpliwości również proszę o kontakt mailowy w celu ustalenia terminu i formy współpracy zdalnej.

### Wprowadzenie do silnika Unity.

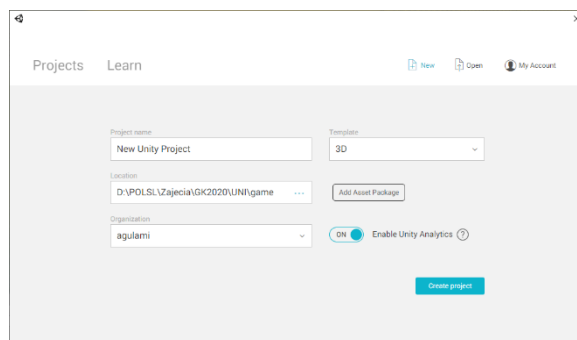
Silnik Unity jest przeznaczony do tworzenia gier 2D oraz 3D oraz interaktywnych aplikacji na różne platformy (Windows, UWP, android, mac i inne). Celem laboratorium jest zapoznanie się z podstawowymi elementami silnika. W tym celu należy zaprojektować prostą grę omówioną poniżej.

### Wymagania do realizacji laboratorium:

- Silnik Unity – instrukcja została opracowana na silniku Unity w wersji 2018.4.5 która dostępna jest na stronie (<https://unity3d.com/get-unity/download/archive>). Do realizacji laboratorium niekoniecznie jest wymagana wyżej wymieniona wersja, jednak dla innych wersji część elementów wprowadzenia może się nieznacznie różnić. W celu skorzystania z Unity należy założyć konto.
- Microsoft Visual Studio – na potrzeby tego laboratorium opcjonalne.

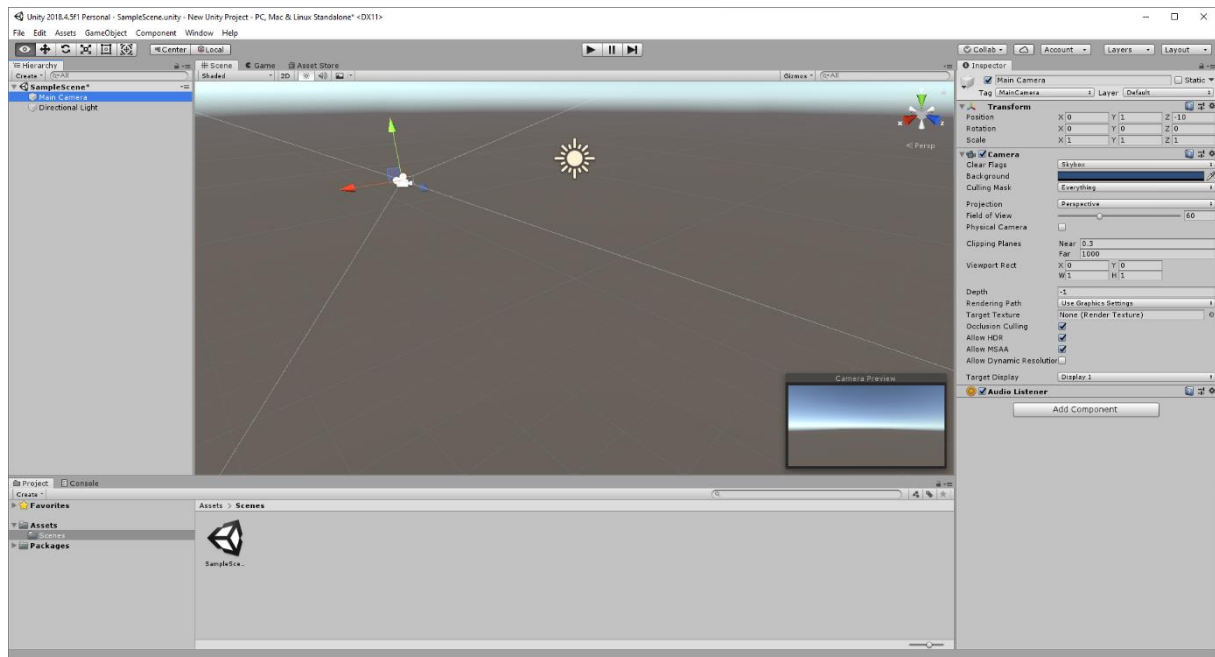
### Interfejs Unity:

Po uruchomieniu środowiska Unity pierwszym elementem jest *okno Projektu (Project)* przedstawiony na: Rysunek 1. Możliwe jest w nim utworzenie nowego projektu (*New*) lub projektu wcześniej utworzonego i zapisanego na dysku (*Open*).



Rysunek 1. Okno projektu.

Po utworzeniu nowego projektu wyświetlone zostanie okno główne środowiska Unity przedstawione na Rysunek 2.



Rysunek 2. Okno główne silnika unity. Widok domyślny po wczytaniu pustego projektu.

. Okno główne zawiera następujące panele:

- *Panel Project* – zawiera wszystkie elementy (zasoby) stworzone dla projektu np. skrypty, pliki, tekstury, modele itd. Odzwierciedla on strukturę plików na dysku. Głównym folderem, w którym zawarte są wszystkie potrzebne zasoby dla gry jest folder *Assets*. Zarządzanie elementami odbywa się za pomocą standardowej techniki przeciągnij i upuść.
- *Panel Hierarchy* – zawiera wszystkie elementy wyświetlane na bieżącej scenie. Po utworzeniu pustego projektu zawiera on dwa elementy dla sceny *SampleScene* (dostępnej w panelu *Project* -> *Assets* -> *SampleScene*). Elementami są:
  - *Main Camera* – kamera główna.
  - *Directional Light* – światło kierunkowe;

Aby dodać nowy element do sceny należy z menu głównego wybrać opcję *GameObject*, a następnie interesujący nas element np. wybranie *GameObject*->*3D Object*->*Cube* doda nam na scenę oraz w *panelu Hierarchy* sześcian. Alternatywnym sposobem jest dodanie elementu z menu kontekstowego dostępnego w *panelu Hierarchy* gdzie należy wybrać interesujący nas element. W celu dodaniu elementu już istniejącego w projekcie (stąd też dostępnego w *Panelu Project*) należy skorzystać z techniki przeciągnij (z *Panelu Project*) i upuść (w *Panelu Hierarchy*).

- *Panel Inspektor* – wyświetla wszystkie właściwości zaznaczonego obiektu (zaznaczonego w *Panel Hierarchy* lub na scenie). Przykładowo dla kamery głównej (*Main Kamera*) dostępne są następujące komponenty: transformacja obiektu (*transform*), właściwości kamery (*Camera*) oraz *Auto Listener*. Możliwe jest dodanie własnych komponentów za pomocą przycisku *Add Component*. Komponentami mogą być wbudowane komponenty z silnika Unity oraz własne skrypty.
- *Panel Scene* – pokazuje wygląd tworzonej gry.



Rysunek 3. Przyciski nawigacyjne.

opcje transformacji obiektów dostępne za pomocą przycisków nawigacyjnych przedstawionych na Rysunek 3.

- *Panel Game* – pozwala na uruchomienie gry w edytorze. Edytor automatycznie przechodzi w tryb gry po uruchomieniu gry.

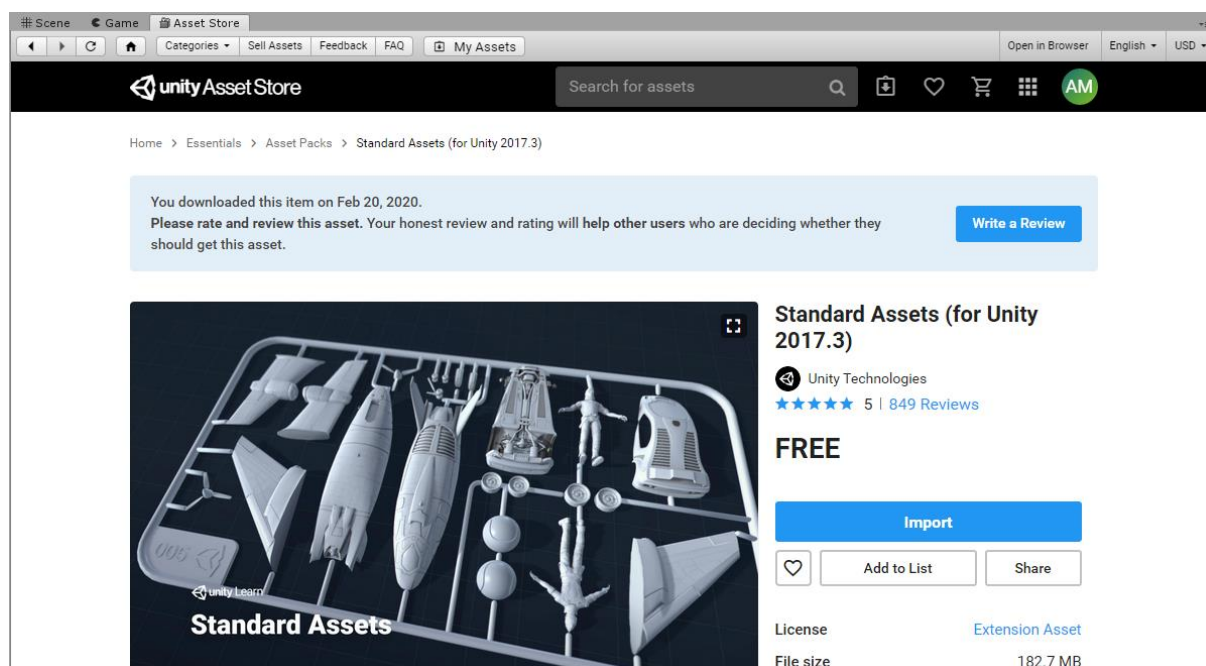
### Cel realizacji na laboratorium.

Na laboratorium należy zaprojektować prostą grę. W grze, gracz rozpoczyna rozgrywkę na początku pewnego obszaru i jego celem jest jak najszybsze dotarcie na drugi koniec obszaru. Po drodze gracz napotyka na przeszkody takie jak: wzgórza, drzewa i inne przeszkody.

### Realizacja laboratorium:

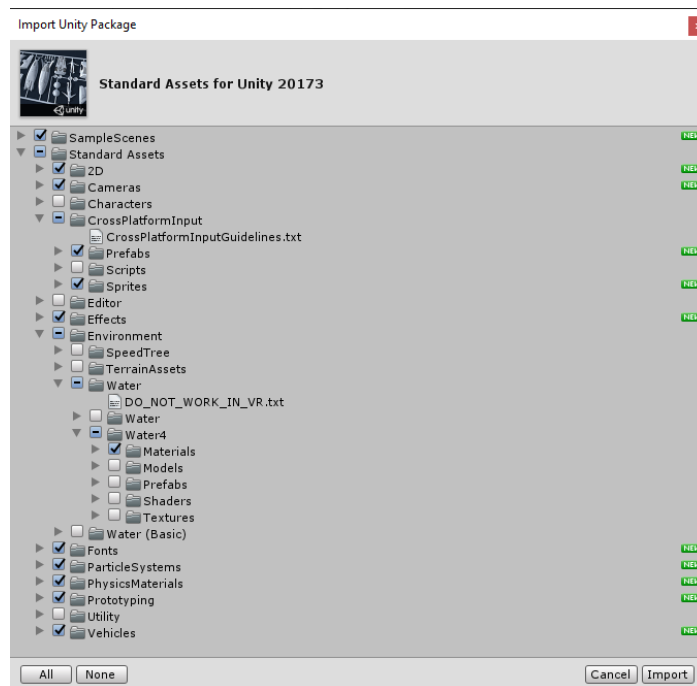
#### *Dodanie standardowych assetów do projektu.*

Tworząc gry w środowisku Unity istnieje wiele elementów wspólnych dla wszystkich gier, takich jak kontroler postaci. Dlatego twórcy Unity (poza wbudowanymi standardowymi assetami w środowisku) udostępnili również darmowe assety na stronie unity *Asset Store*. Assety możemy utożsamiać z modelami lub po prostu gotowymi obiektami, które można zaimportować do gry. Do realizacji laboratorium należy pobrać z *Asset Store* standardowe assety wydane przez *Unity Technologies*. Assety można pobrać bezpośrednio z środowiska Unity. Zakładka ze sklepem Asset Store powinna być widoczna w oknie głównym (jedna z zakładek obok *Panelu Scene* oraz *Panelu Game*). Jeśli nie jest widoczna można ją uruchomić z menu głównego *Window->Asset Store*. W oknie wyszukiwania należy wpisać *Standard Assets* gdzie powinien zostać znaleziony pakiet widoczny na Rysunek 4.



Rysunek 4. Przykładowy pakiet assetów dostępnych na Asset Store.

Po zaimportowaniu pakietu, możliwe jest pobranie wszystkich modeli bądź wybranych. Do realizacji laboratorium wystarczy pobrać następujące assety z paczki *Standard Assets: CrossPlatformInput, Editor, Utility, Character* oraz *Environment*. Na Rysunek 5 pokazane są wyżej wymienione assety (te które są odznaczone). W każdej chwili można dociągnąć brakujące elementy powtarzając powyższe kroki.



Rysunek 5. Wybrane elementy ze standardowych assetów.

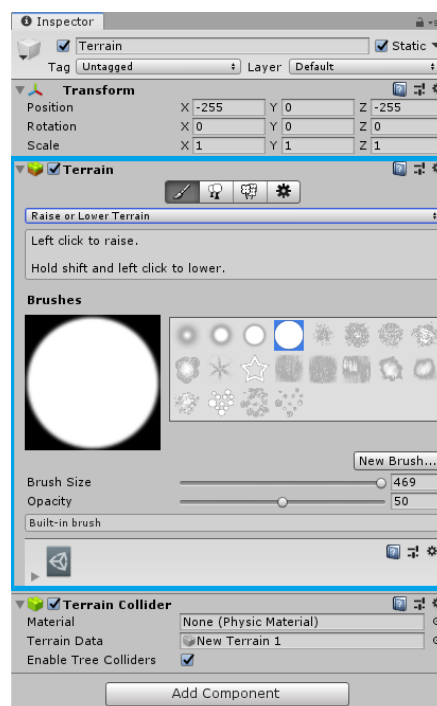
#### Budowanie terenu:

Pierwszym elementem który należy wykonać to zbudowanie świata w którym będzie rozgrywana rozgrywka, czyli stworzenie terenu. W tym celu w *panelu Hierarchy* należy z menu kontekstowego wybrać opcję *3D Object -> Terrain* ewentualnie z Menu Głównego *GameObject->3D Object ->Terrain*. Po wybraniu tej opcji w *panelu Scene* pojawi się płaszczyzna która będzie naszym światem. Po zaznaczeniu terenu w *Panelu Scene*, w *panelu Inspektor* pojawią się właściwości terenu. Można zmienić jego położenie (*Position*) w właściwościach Transformacji.

#### Rzeźbienie terenu

W kolejnym kroku należy zróżnicować rzeźbę terenu. W tym celu w *panelu Inspektor* we właściwości *Terrain* należy wybrać opcję *Paint Terrain* przedstawioną na Rysunek 6. Następnie należy wybrać opcję *Raise or Lower Terrain* pozwalającą za pomocą pędzla na rzeźbienie terenu, czyli tworzenie wzniesień. W przypadku tworzenia dolin lub tworzenia zbiorników wodnych (które będą tworzone w późniejszym etapie) zaleca się podnieść cały teren powyżej a następnie przytrzymując klawisz *shift* obniżyć teren w miejscach przeznaczonych na wgłębienia. Zaleca się również aby tworzony teren otoczony był wzgórzami uniemożliwiającymi opuszczenie terenu przez playera (przez wypadnięcie poza teren).

Istnieją oczywiście inne metody tworzenia terenu, np. na podstawie mapy wysokości. Przedstawiony powyżej sposób jest jednym z przykładowych i nie obowiązkowym dla celów realizacji laboratorium.



Rysunek 6 Rzeźbienie terenu.

### Dodanie tekstur do terenu

Po utworzeniu rzeźby terenu należy dodać tekstury. W tym celu w *Panelu Inspector* zamiast opcji *Raise or Lower Terrain* należy wybrać opcję *Paint Texture*. We właściwości *Terrain Layers* nacisnąć przycisk *Edit Terrain Layers...*, a następnie wybrać opcję *Create Layer*. Po wybraniu tej opcji pojawi się nowe okno z teksturami *SelectTexture2D*. Z dostępnych tekstur należy wybrać interesującą nas teksturę, która pokryje cały teren. W celu dodania nowej tekstury należy czynność powtórzyć. Dodanie kolejnych warstw pozwala nam za pomocą pędzla na malowanie po terenie. W tym celu należy wybrać interesującą nas teksturę, a następnie za pomocą pędzla nanieść teksturę na teren.

### Dodanie drzew do terenu

Po nałożeniu tekstur na teren możliwe jest dodanie innych obiektów 3D które będą pełniły rolę przeszkód. Przykładem mogą być drzewa. Silnik Unity posiada wbudowanie drzewa. W *panelu Inspector* we właściwościach *Terrain* możliwe jest dodawanie drzew (opcja *Paint Trees*). W sekcji *Trees* możliwe jest utworzenie drzew w sposób podobny jak to było przedstawione w malowaniu tekstur. Dodając nowe drzewa należy nacisnąć przycisk *Edit Trees...*, a następnie *Add Tree*. W nowym oknie pojawi się opcja dodania prefabrykatów (*prefab*) czyli gotowych obiektów 3D z określonymi parametrami. Z pobranych wcześniej zasobów możliwe jest zaimportowanie gotowych obiektów. W tym celu w *panelu Project* wybierany jest folder *Assets->Standard Assets->Environment->SpeedTree->Broadleaf*, a następnie należy zaznaczyć obiekt *Broadleaf\_Desktop* i przeciągnąć do okna *Add Tree* w miejsce gdzie powinien znaleźć się obiekt, a aktualnie jest puste (*None (Game Object)*). Po zatwierdzeniu wyboru (*Add*) nowy typ drzewa zostanie dodany w *panelu Inspector*. Powstałe w ten sposób drzewa dodawane są do terenu za pomocą pędzla w sposób analogiczny jak to było realizowane w przypadku tekstur.

### Dodanie wody do terenu.

W celu dodania wody do terenu możliwe jest również skorzystanie z wcześniej zaimportowanych prefabrykatów. Prefabrykat wody o nazwie *Water4Advanced* znajduje się w folderze *Assets->Standard Assets->Environment->Water->Water4->Prefabs*. Aby umieścić wodę na terenie należy przenieść go do *Panelu Hierarchy*. Po przeciągnięciu prefabrykatu wody zostanie on automatycznie umieszczony na scenie. Za pomocą transformacji należy umieścić wodę w odpowiednim miejscu, czyli w miejscu gdzie woda będzie widoczna we wszystkich wgłębieniach terenu przewidzianych na nią.

### Kontroler postaci

Kontroler postaci pozwala na poruszanie się użytkownika na scenie. Do sterowania w grze, twórcy Unity dostarczyli nam dwa kontrolery *Third Person Controller* i *First Person Controller*, które różnią się orientacją kamery. Są one dostępne w paczce z wcześniej pobranego pakietu standardowych pakietów. Kontroler *First Person Controller* znajduje w folderze *Assets->Standard Assets->Characters->FirstPersonCharacter->Prefabs*. Analogicznie jak w przypadku wody przeciągamy prefabrykat *FPSController* do *Panelu Hierarchy*. Kontroler postaci należy ustawić na jednym ze skajów terenu (zgodnie z założeniami przebiegu gry).

Ponieważ kontroler posiada własną kamerę, główną kamerę możemy usunąć z *Panelu Hierarchy*. W tym momencie możliwe jest przetestowanie gry i obejrzenie stworzonego świata z punktu widzenia gracza. Sterowanie odbywa się za pomocą strzałek lub klawiszy WASD.

### Tworzenie strefy zakończenia gry

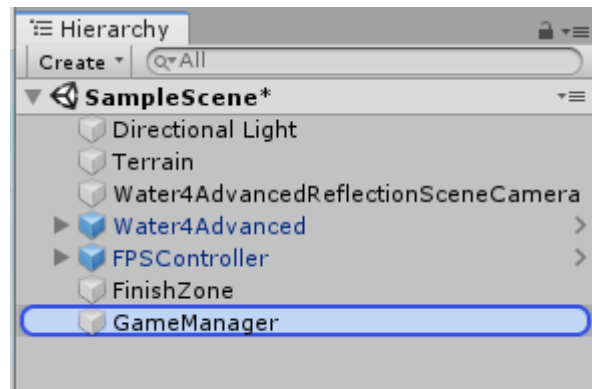
Celem zakończenia gry jest przebycie terenu w najszybszym czasie. W tym celu należy stworzyć miejsce do którego gracz będzie musiał się udać. Będzie to prosty obiekt z światłem punktowym, który pozwoli graczowi na rozpoznanie tego miejsca. Aby stworzyć strefę zakończenia gry należy w *panelu Hierarchy* dodać pusty obiekt (*create Empty*) i umieścić go na terenie, w przeciwnym miejscu w stosunku do



kontrolera postaci. Do obiektu należy dodać komponent światła. Komponenty dodajemy poprzez naciśnięcie przycisku *AddComponet* w *Panelu Inspector*. Z dostępnych opcji należy wybrać *Light*. W tym momencie komponent zostaje dodany do świata, można pozmieniac jego właściwości w celu lepszej widoczności strefy zakończenia. W kolejnym kroku należy dodać komponent kolajdera. Kolajder pozwala na wykrycie kolizji z obiektem, w tym przypadku wejście gracza w strefę zakończenia. Kolajder dodawany jest analogicznie jak komponent światła. Sugeruje się dodanie komponentu nazwanego *Capsule Collider* i zmianę wartości promienia. Dodatkowo należy zaznaczyć pole *Is Trigger*.

#### Dodanie skryptów do obiektów.

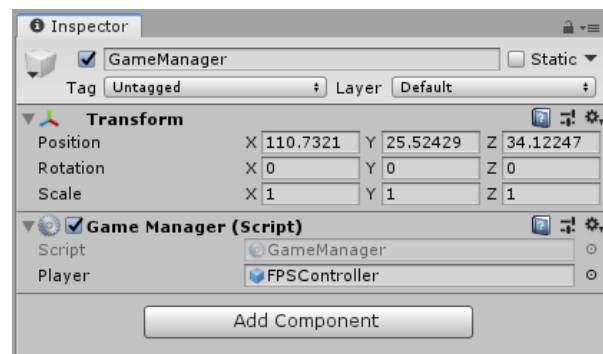
Jednym ze sposobów zarządzania obiektami w grze jest dodanie skryptów do obiektów. W tym przypadku należy podpiąć skrypt jako komponent. Do realizacji laboratorium przygotowano dwa skrypty (dostępne na platformie), pierwszy z nich *GameManager* tworzy GUI i zarządza ogólnym przebiegiem gry. Drugi *FinishZone* wywołuje metodę z *GameManagera* w momencie gdy gracz dotrze do miejsca docelowego. Oba skrypty powinny zostać umieszczone w folderze *Assets*, oczywiście można stworzyć podfolder w celu utrzymania przejrzystości projektu. W większości przypadków skrypty muszą być podpięte pod obiekty, dlatego aby dodać skrypt *GameManager* należy stworzyć pusty obiekt (analogicznie jak przypadku tworzenia obiektu będącego reprezentantem miejsca zakończenia gry) o nazwie *GameManager* a następnie przeciągnąć skrypt na ten obiekt. W momencie przeciągania skryptu z *Panelu Projekt* do obiektu w *Panelu Hierarchy*, obiekt na który zostaje przeciągnięty skrypt powinien być zaznaczony jak zostało to przedstawione na Rysunek 8. Po poprawnym dodaniu skryptu do obiektu jest on widoczny w *panelu Inspector* tego obiektu. Jedną z możliwości komunikacji pomiędzy obiektami za pomocą skryptów jest tworzenie pól publicznych w skryptach do których możemy z poziomu Unity przypisać inne obiekty. W przypadku skryptu *GameManager* zostało stworzone pole publiczne reprezentujące kontroler postaci:



Rysunek 8. Dodanie skryptu do obiektu na scenie.

`public` GameObject player;

dzięki czemu z poziomu Unity w *panelu Inspector* możemy dodać nasz kontroler postaci. W tym celu należy z *panelu Hierarchy* przeciągnąć obiekt *FPSController* w miejsce Player w panelu Inspector jak to zostało przedstawione na Rysunek 7. Analogicznie postępowanie należy wykonać w przypadku skryptu *FinishZone*. Skrypt umieszczamy w obiekcie *FinishZone* a następnie przeciągamy obiekt *GameManager* w puste pole w *panelu Inspector*.



Rysunek 7. Dodanie innego obiektu do skryptu.