

### 1. Opis gry

Gra polega na zbieraniu postacią świecących grzybów rozmieszczonych losowo na określonym terenie. Celem gry jest zebranie jak największej ilości grzybów w czasie trzech minut. Postać porusza się między drzewami, których należy jednak unikać, ponieważ w momencie zderzenia z niektórymi z nich postać jest odpychana do tyłu. Nieświecących grzybów nie można zbierać.

### 2. Specyfikacja wewnętrzna

Projekt złożony jest z trzech folderów:

- Assets, w którym przechowywane są wszystkie elementy wizualne takie jak modele czy tekstury, oraz skrypty określające działanie obiektów.
- Packages i ProjectSettings – zawierają one pliki wygenerowane przez środowisko Unity niezbędne do poprawnego uruchomienia projektu.

W folderze Assets/MyAssets znajdują się utworzone w ramach pracy nad projektem pliki. Pozostałe pliki zostały zaimportowane z sklepu Asset Store.

Przegląd utworzonych skryptów:

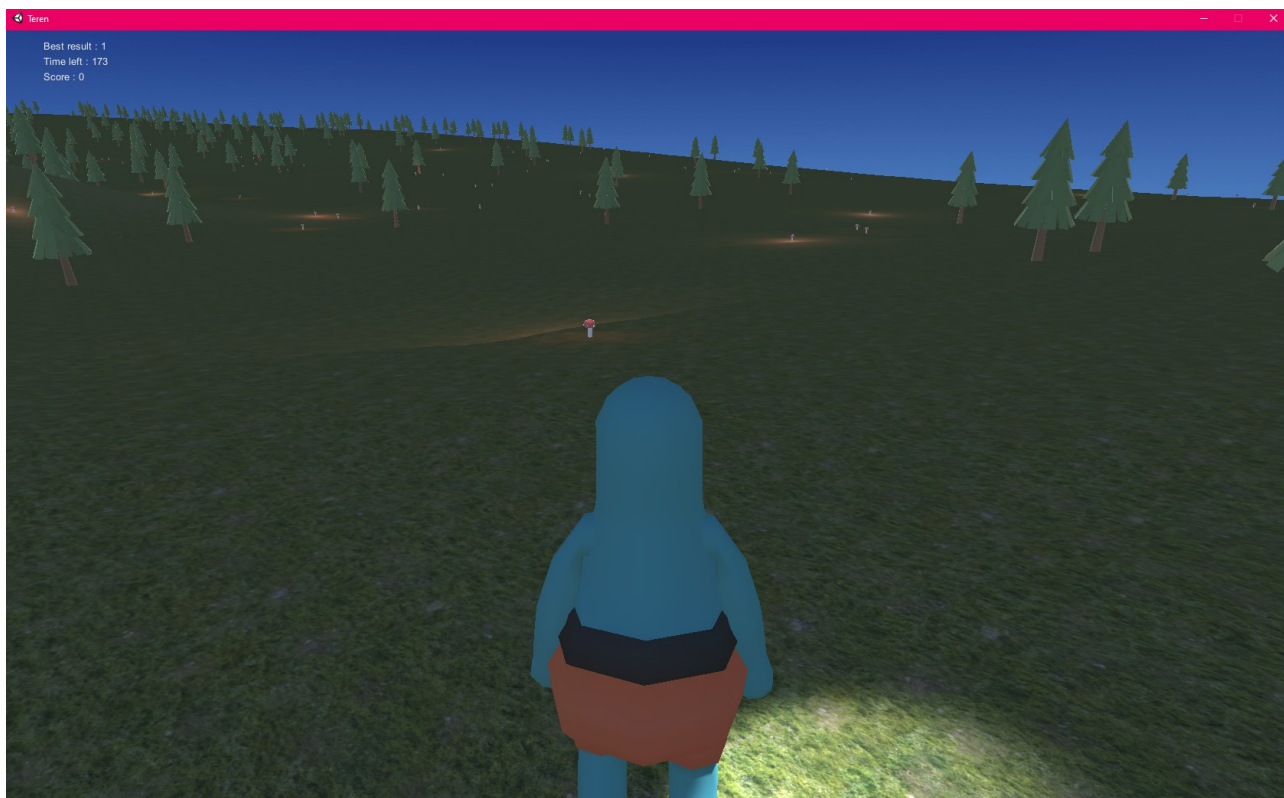
- CharacterAnimation – odpowiada on za ustawianie właściwych wartości parametrów: BlendJumpWalk, BlendJumpRun (typu float), isWalking, isWalkingBackwards, isRunning, isJumping w zależności od tego który klawisz jest wciśnięty przez użytkownika. Parametry te są wykorzystywane w kontrolerze animacji.
- GameManager – zarządza grą poprzez kontrolę jej stanu, wyświetlanie aktualnego wyniku, uruchamianie i kończenie jej poprzez:
  - void StartGame()
  - void GenerateMushrooms() generującą świecące grzyby które gracz będzie zbierać, poprzez utworzenie w losowych miejscach na powierzchni terenu obiektów na podstawie przygotowanego wcześniej prefaba (gotowego już modelu obiektu zawierającego wszystkie niezbędne komponenty).
  - void StopGame(),
  - OnGUI().
- LightManager – odpowiada za włączanie lub wyłączanie głównego światła (przycisk Enter), oraz włączanie lub wyłączanie latarki (światła typu Spotlight połączonego z prawą ręką postaci gracza) za pomocą przycisku L.
- MushroomCollision – skrypt ten odpowiada za reakcję na zetknięcie postaci gracza ze świecącym grzybem (zwiększenie liczby punktu oraz zniszczenie obiektu grzyba). Odbywa się to za pomocą metody OnTriggerEnter.
- TreeCollision – odpowiada za reakcję na zetknięcie postaci gracza z drzewem. Ponownie odbywa się to za pomocą metody OnTriggerEnter, korzystającej z Rigidbody.AddForce.

### 3. Specyfikacja zewnętrzna

Gra rozpoczyna się od razu po uruchomieniu. Scena składa się z terenu, drzew, grzybów i pni (pełniących tylko funkcje ozdobne) oraz świecących grzybów, które gracz powinien zbierać poprzez zetknięcie postaci gracza z takim obiektem.



W górnym lewym rogu znajduje się wynik gracza, jego najwyższy dotychczasowy wynik oraz informacja ile czasu pozostało do końca gry. Poruszanie odbywa się za pomocą przycisków W (przód), A (lewo), S (tył), D (lewo). Dodatkowo poruszając się do przodu przytrzymując lewy klawisz Shift postać może biec lub za pomocą spacji skakać.



Za pomocą klawisza Enter użytkownik może włączyć lub wyłączyć główne światło, oraz analogicznie za pomocą klawisza L włączyć lub wyłączyć latarkę.



Gdy czas gry upłynie wyświetlany jest komunikat, a grzyby przeznaczone do zbierania są niszczone. Za pomocą klawisza R można uruchomić grę ponownie, co spowoduje wygenerowanie nowych losowo rozmieszczonych grzybów i rozpoczęcie odliczania czasu.

#### 4. Testowanie

Projekt został zbudowany na komputery PC i na takim sprzęcie przetestowany. Ze względu na bardzo prostą grafikę i nieskomplikowane działanie nie zaobserwowano problemów z wydajnością. Problemem zaobserwowanym podczas testowania było nieprawidłowe rozmieszczanie grzybów na powierzchni terenu. Przez niepoprawnie obliczoną składową Y znajdowały się one nieznacznie pod powierzchnią. W finalnej wersji podczas tworzenia pojawiają się one nad powierzchnią i dzięki komponentowi RigidBody zgodnie z grawitacją spadają i osiadają na powierzchni (zablokowana rotacja we wszystkich trzech osiach powoduje że nie przewracają się i osiadają prosto).

#### 5. Podsumowanie

Silnik Unity stanowi bardzo dobre narzędzie do tworzenia aplikacji zarówno dla początkujących jak i zaawansowanych użytkowników. Ponadto w niezwykle prosty sposób można osiągnąć ciekawe efekty. Wykonany projekt może być tego przykładem – nawet bez użycia kodu można wprowadzić niektóre elementy fizyki – większość własności można nadać poprzez dodanie odpowiednich komponentów, a współczynniki edytować z interfejsu graficznego edytora.