

## 1. Użyta technologia

Nasz generator map 3D został stworzony na silniku Unity, który służy do tworzenia trójwymiarowych oraz dwuwymiarowych gier komputerowych lub innych materiałów interaktywnych. Silnik umożliwia nam pisanie skryptów w języku C#, który to wykorzystaliśmy w naszym projekcie. Mapa 3D tworzona jest z pomocą map wysokości, które są generowane przez nas za pomocą szumu Perlina.

## 2. Instrukcja uruchomienia

Uruchomienie naszego generatora wymaga od nas posiadania silnika Unity w wersji 2020.3.31f1 oraz plików projektowych. Gdy posiadamy obie te rzeczy, po wybraniu naszego projektu w menadżerze projektów Unity, możemy uruchomić generator map i zacząć generować własne mapy.

## 3. Instrukcja obsługi i opis działania

Nasz generator pozwala w prosty sposób generować mapy według własnych preferencji, które możemy dostosować przy pomocy obiektu „MapGenerator”, znajdującego się po lewej stronie ekranu. Po kliknięciu na niego po prawej stronie w inspektorze zobaczymy właściwości tego obiektu. Wewnątrz jego znajduje się skrypt, który jest menadżerem do zarządzania mapą.

Pierwsza opcja jest to „Draw Mode” pozwala ona na wybranie trybu rysowania. Daje nam możliwość wygenerowania np. samego szumu, czy też nałożenia na niego odpowiednich regionów, które są kolorowane, lub po prostu stworzenia z tego mapy 3D.

Kolejne opcje dają nam możliwość manipulacji szerokością i długością mapy.

Kolejna wartość jest to skala szumu, czyli jak duży ma być szum, im mniejsza wartość tym więcej mniejszych cząsteczek a im większa wartość to stają się one coraz większe co powoduje stworzenie terenu, która posiada mniejszą ilość gór i dołków.

Następna wartość, którą możemy wpływać na teren jest „Octaves”, określa to liczbę różnych warstw szumu nałożonych na siebie w celu uzyskania mapy.

Kolejna zmienna pozwalająca modyfikować mapę to „Persistence”, pozwala ona określić wpływ oktawy na kształt terenu, ponieważ pierwsze oktawy, które formują góry mają większy wpływ na teren niż kolejne formujące nierówności.

Następna wartość to „Lacunarity”, odpowiedzialna jest na zwiększanie detali, które są generowane przez kolejne oktawy.

Kolejna opcja to „Seed”, odpowiada ona za determinację stanu generatora liczb losowych, co pozwala na uzyskanie jednakowych efektów przy zastosowaniu tego samego ziarna.

„Offset” jest to przesunięcie mapy, czyli o ile chcemy ją przesunąć względem szerokości lub długości, w ten sposób możemy zobaczyć jak wygląda teren w innym miejscu.

Następnie mamy do wyboru opcję „Use Falloff”, która pozwala nam utworzyć wodę

wokół mapy. Działa to na zasadzie takiej że generuje mapę z wartością 1 blisko krawędzi przechodzących w wartości 0 do środka i miesza to z mapą szumu.

„Mesh Height Multiplier” jest to mnożnik dla wysokości modelu terenu, czyli pozwala na utworzenie wyższych gór bądź niższych.

„Mesh Hight Curve” jest to krzywa, która określa na jakie wartości ma wpływać bardziej bądź mniej „Mesh Height Multiplier”.

Opcja nazwana „Auto Update” pozwala nam określić, czy mapa ma być aktualizowana automatycznie po zmianie którejś z wyżej opisanych opcji czy też dopiero w momencie kliknięcia przez nas przycisku o nazwie „Generate”.

Ostatnia już możliwa opcja w naszym menadżerze służy do określenia liczby regionów i ich dostosowania. Pozwala nam to na dowolne kolorowanie mapy w zależności od wysokości terenu. Możemy określić na jakiej wysokości ma znajdować się woda a na jakiej zaczynać się trawa, czy też skały. Możemy utworzyć dowolną ilość regionów.

Na końcu menadżera znajduje się przycisk „Generate”, który po naciśnięciu generuje naszą mapę z zadeklarowanymi wartościami.

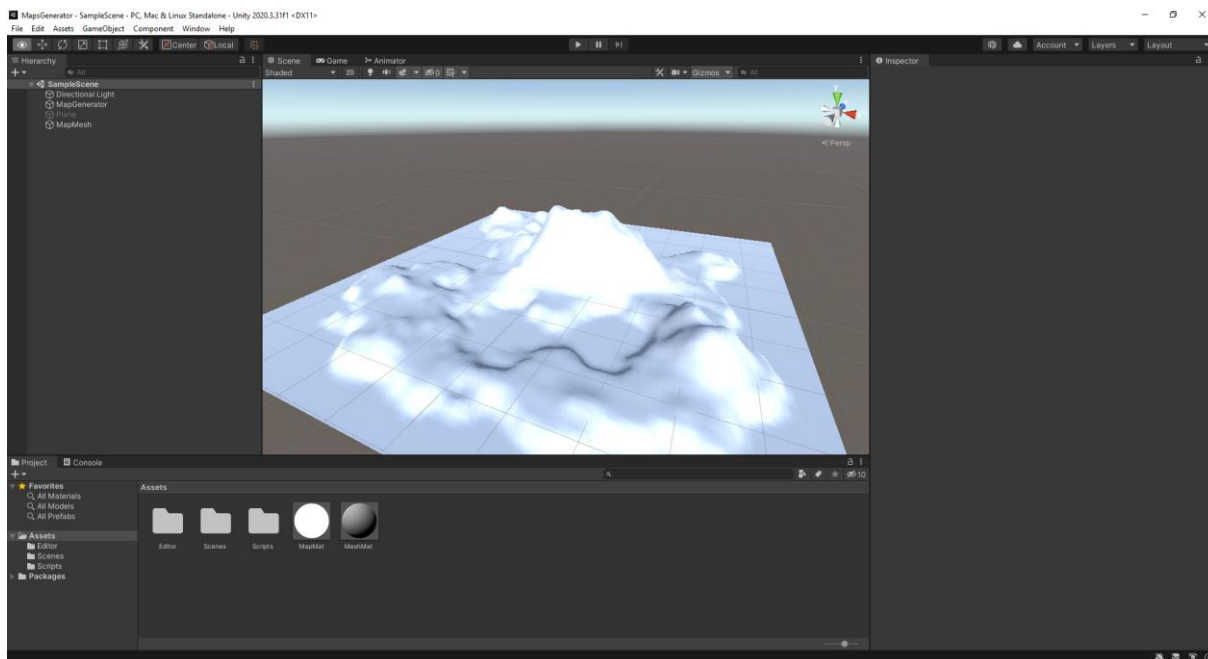
Prócz obiektu „MapGenerator”, posiadamy również „Plane”, który pokazuje nam jak wygląda aktualny obraz monochromatyczny dla którego generujemy teren. Domyślnie został on przez nas ukryty, lecz w każdej chwili możemy go wyświetlać na ekranie. Wystarczy kliknąć na obiekt „Plane” po prawej stronie a następnie po lewej w inspektorze na górze, nad napisem „Tag” zaznaczyć ten kwadracik. Gdy tak zrobimy ukaże nam się obraz, na podstawie którego generowany jest teren.

Kolejnym obiektem jest „MapMesh”, jest to po prostu mesh czyli siatka wierzchołków, wygenerowanego przez nas terenu.

Ostatni obiekt to „Directional Light” czyli światło dla naszego świata. Możemy zmienić jego pozycję, kolor, intensywność, typ świecenia i wiele innych rzeczy, które pozwala nam Unity.

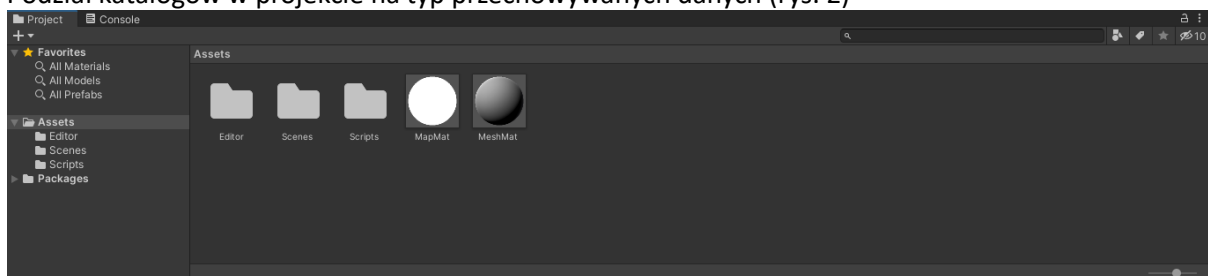
## 4. Zrzuty ekranu

Ekran po wczytaniu projektu (rys. 1)



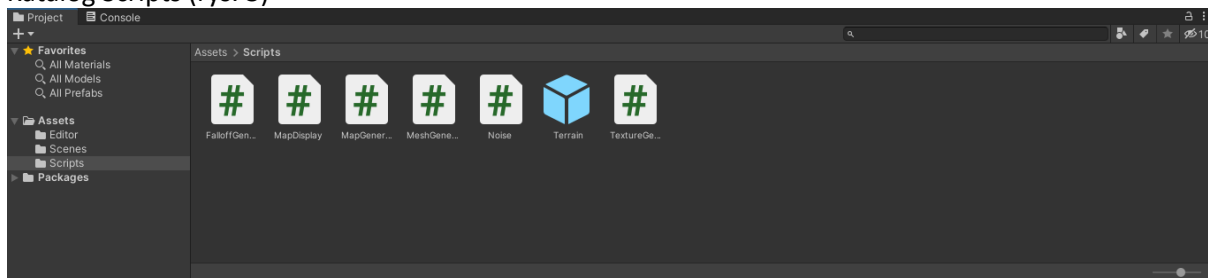
Rys. 1. Ekran po wczytaniu projektu.

Podział katalogów w projekcie na typ przechowywanych danych (rys. 2)



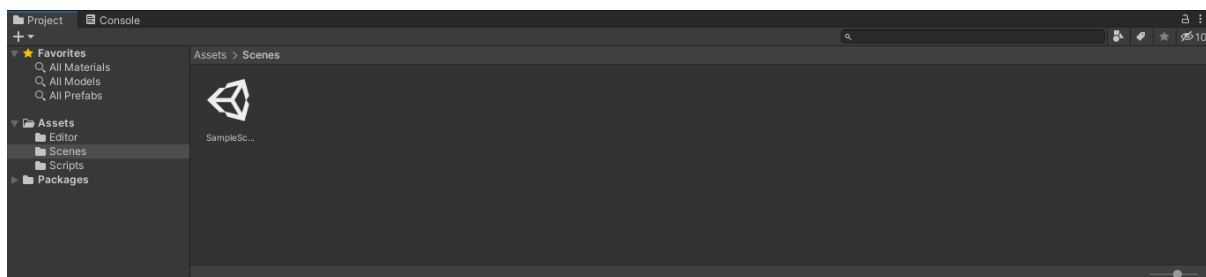
Rys. 2. Podział katalogów w projekcie na typ przechowywanych danych.

Katalog Scripts (rys. 3)



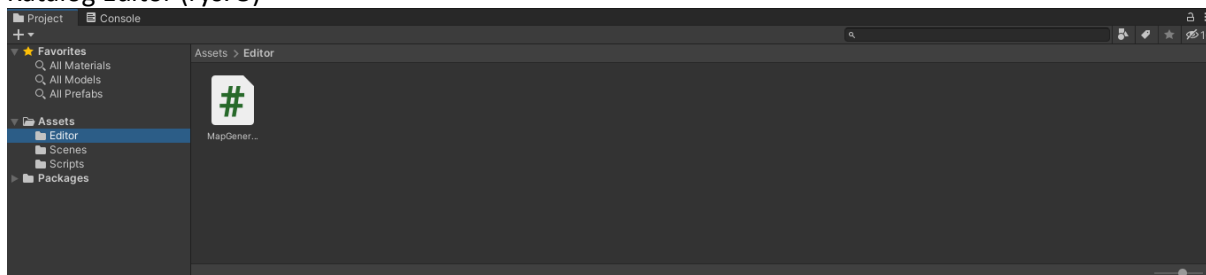
Rys. 3. Katalog Scripts.

Katalog Scenes (rys. 4)



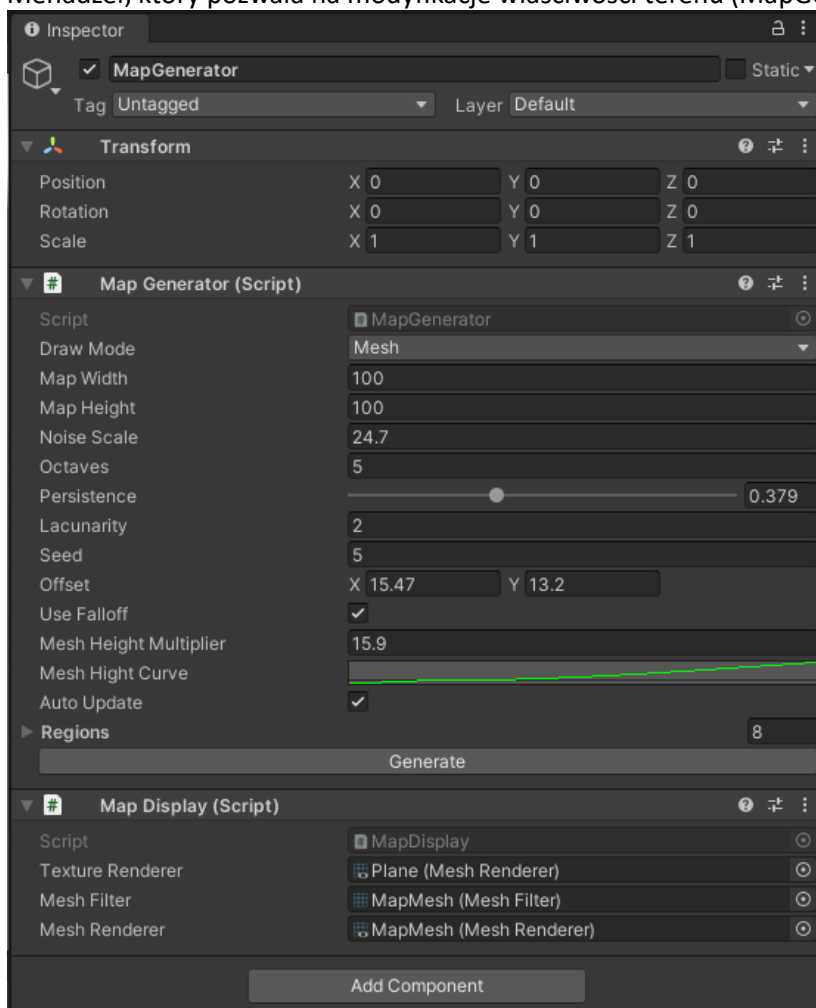
Rys. 4. Katalog Scenes.

Katalog Editor (rys. 5)



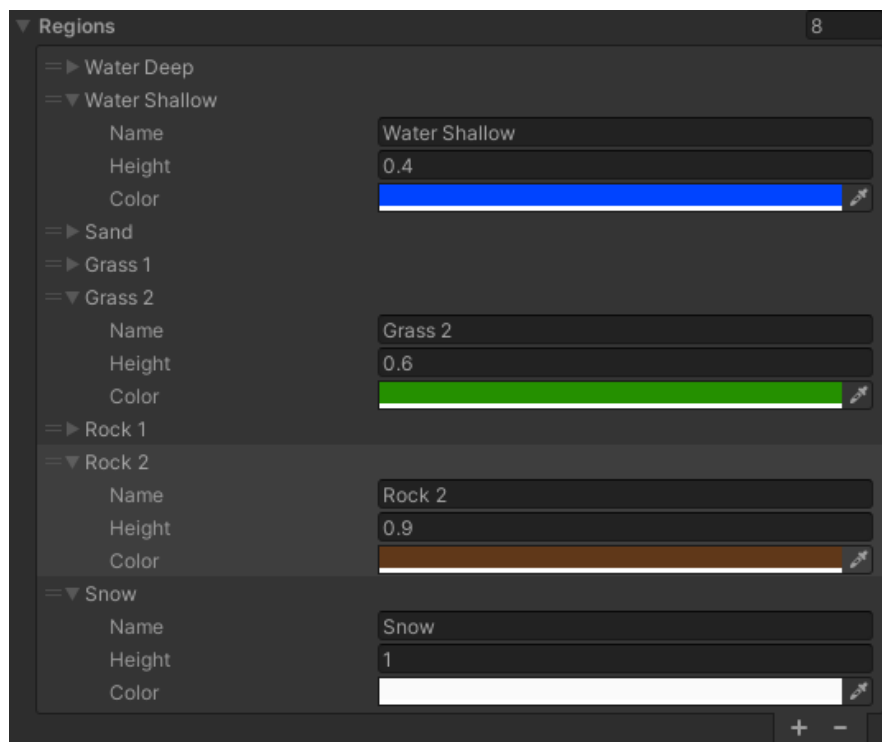
Rys. 5. Katalog Editor.

Menadżer, który pozwala na modyfikację właściwości terenu (MapGenerator) (rys. 6)



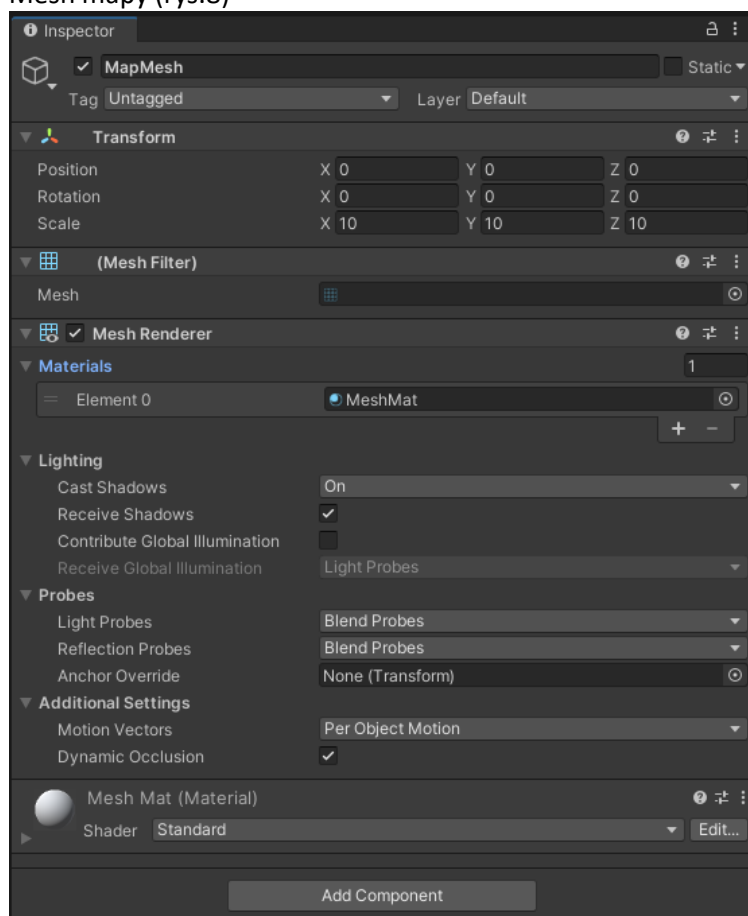
Rys. 6. Menadżer, który pozwala na modyfikację terenu (MapGenerator).

Dostępne regiony mapy (rys. 7)



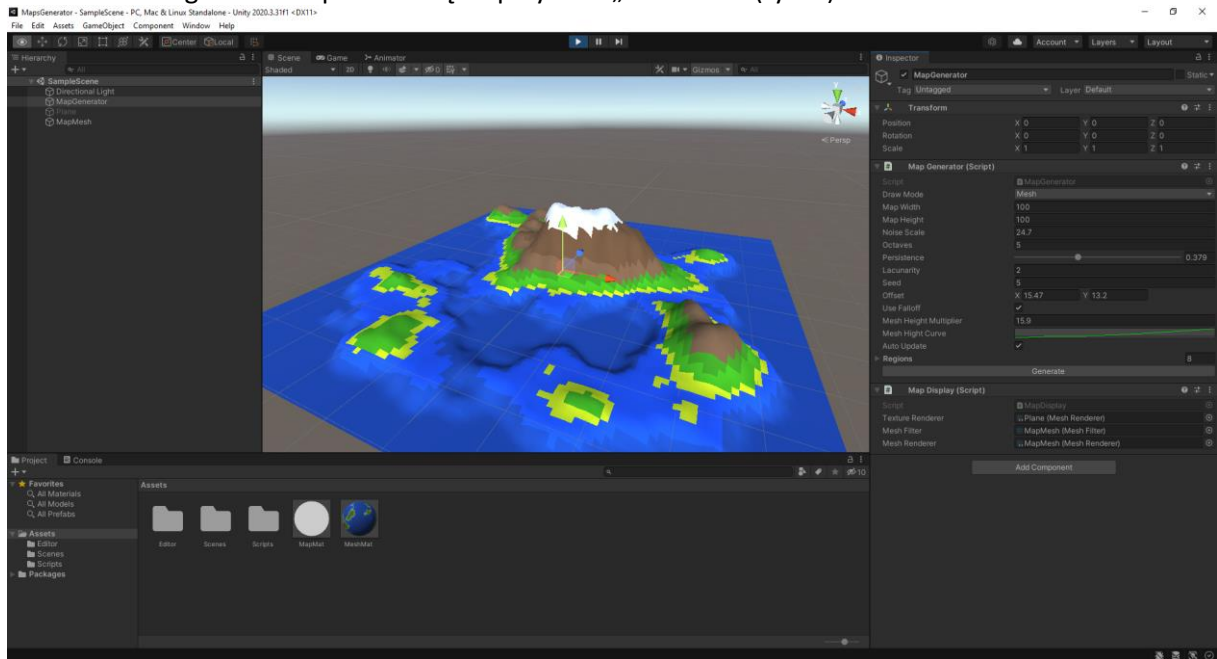
Rys. 7. Dostępne regiony mapy.

Mesh mapy (rys.8)



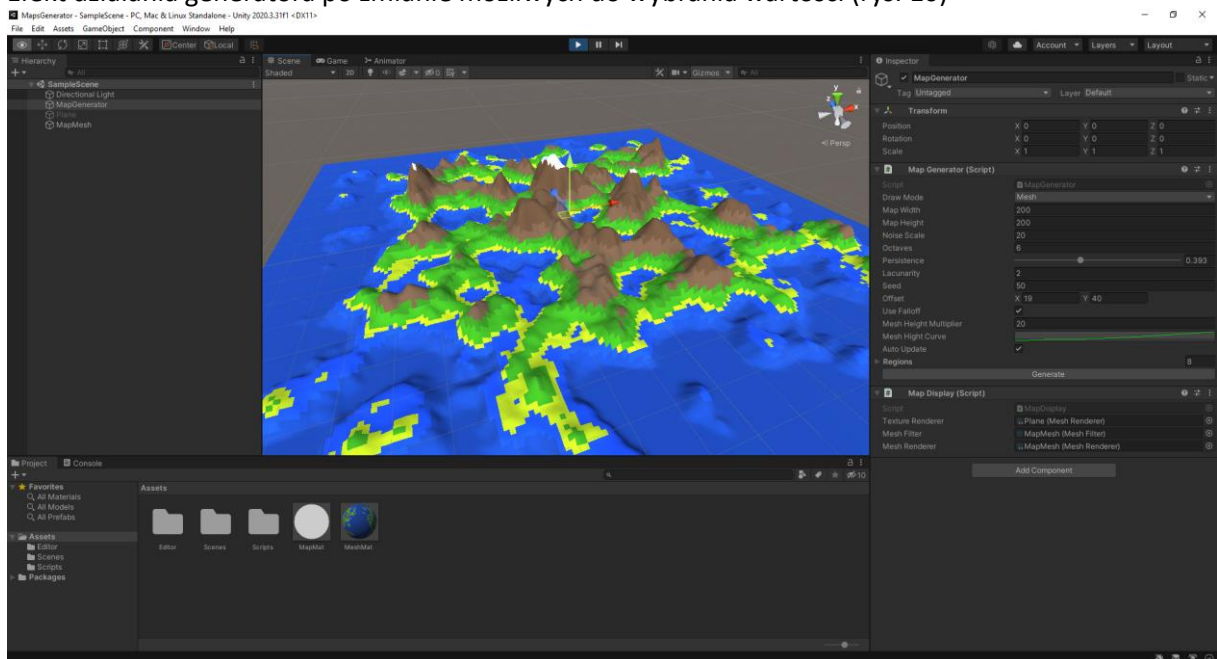
Rys. 8. Mesh mapy.

Efekt działania generatora po naciśnięciu przycisku „Generate” (rys. 9)



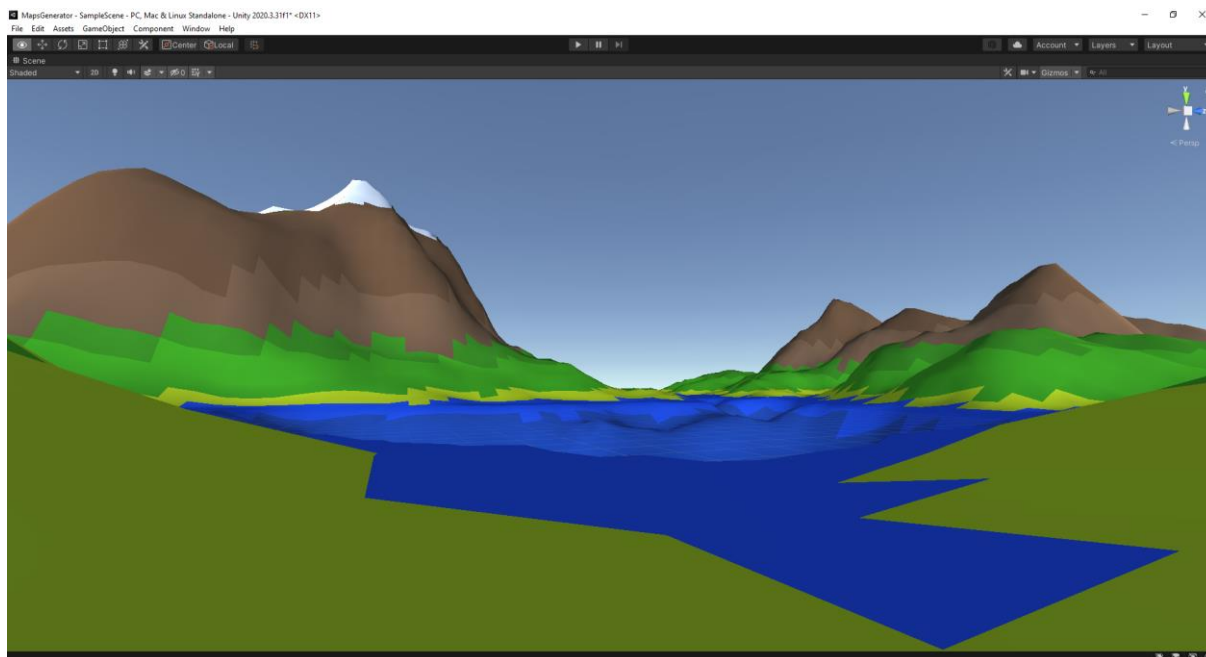
Rys. 9. Efekt działania generatora po naciśnięciu przycisku „Generate”.

Efekt działania generatora po zmianie możliwych do wybrania wartości (rys. 10)



Rys. 10. Efekt działania generatora po zmianie możliwych do wybrania wartości.

Prezentacja wygenerowanego terenu z bliska (rys. 11)



Rys. 11. Prezentacja wygenerowanego terenu z bliska.

## 5. Podsumowanie

Nasz projekt udało się w pełni zrealizować i jesteśmy zadowoleni z efektów naszej pracy. Stworzyliśmy generator, który z pomocą obrazu monochromatycznego, generowanego przy pomocy szumu Perlina tworzy w pełni użytkową mapę terenu 3D, którą możemy później wykorzystać np. w grze wideo. Ponad to, możemy tą mapę dynamicznie modyfikować, widząc na bieżąco efekty naszych zmian, które dokonujemy z pomocą menadżera w którym to mamy możliwość dostosowania każdej opcji. Nasz teren jest również kolorowany co sprawia że wygląda bardziej realistycznie. Możemy tworzyć własne regiony lub modyfikować już stworzone co pozwala stworzenie wielu wariantów tego samego terenu. Podczas tworzenia tego projektu natknęliśmy się na kilka problemów, które ostatecznie udało nam się rozwiązać i ostateczny efekt możemy zobaczyć na powyższych zrzutach ekranu. Wybrana przez nas technologia a głównie silnik Unity bardzo pomógł nam w stworzeniu tego projektu dostarczając już podstawowych funkcjonalności i funkcji z zakresu silnika graficznego. Skrypty, które napisaliśmy podczas tworzenia generatora są w języku C#, który to poznajemy coraz bardziej. Nasz kod staraliśmy się pisać w taki sposób aby był zrozumiały dla każdego i przy pomocy prostych właściwości języka C# oraz funkcji Unity stworzyć poszczególne funkcje. Staraliśmy się tworzyć także dokładne komentarze w skryptach, które pomogą w wyjaśnieniu i zrozumieniu zasady działania naszego programu. Ten projekt jest dla nas cennym doświadczeniem w tworzeniu terenu oraz map 3D, które w przyszłości mamy nadzieję wykorzystywać.