

# Landscape Generator

## Inżyniera Oprogramowania

Artur Bednarczyk, Dawid Grajewski, Tomasz Januszek  
Politechnika Śląska  
Wydział Matematyki Stosowanej  
Informatyka, semestr V

18 listopada 2018

# Spis treści

<b>1</b>	<b>O projekcie</b>	<b>3</b>
1.1	Zespół . . . . .	3
1.2	Temat . . . . .	3
1.3	Cel . . . . .	3
<b>2</b>	<b>Projekt</b>	<b>3</b>
2.1	Plany i pomysły . . . . .	3
2.2	UI/UX . . . . .	3
2.2.1	Zawartość . . . . .	3
2.2.2	Projekty UI . . . . .	3
<b>3</b>	<b>Teoria</b>	<b>3</b>
3.1	Losowość . . . . .	3
3.2	Algorytmy . . . . .	3
3.2.1	Szum Perlina . . . . .	3
<b>4</b>	<b>Narzędzia</b>	<b>4</b>
4.1	Kontrola wersji . . . . .	4
4.2	Zarządzanie zespołem . . . . .	4
4.3	Środowisko . . . . .	4
<b>5</b>	<b>Aplikacja</b>	<b>4</b>
5.1	Architektura . . . . .	4
5.2	Struktury danych . . . . .	4
5.3	Schemat graficzny struktury systemu . . . . .	5
5.4	Komunikacja między modułami . . . . .	5
<b>6</b>	<b>API</b>	<b>5</b>
6.1	Perlin . . . . .	5

# **1 O projekcie**

## **1.1 Zespół**

Artur Bednarczyk, Dawid Grajewski, Tomasz Januszek.

## **1.2 Temat**

TREŚĆ ZADANIA

## **1.3 Cel**

Opis treści.

# **2 Projekt**

## **2.1 Plany i pomysły**

Co chcemy zrobić

## **2.2 UI/UX**

### **2.2.1 Zawartość**

### **2.2.2 Projekty UI**

# **3 Teoria**

## **3.1 Losowość**

Kilka słów o losowości w naszej aplikacji.

## **3.2 Algorytmy**

### **3.2.1 Szum Perlina**

Szum Perlina - co to jest, opis algorytmu itp.

## **4 Narzędzia**

### **4.1 Kontrola wersji**

Do zarządzania kodem i wersjami projektu wykorzystujemy narzędzie Git. Korzystamy z platformy GitHub jako repozytorium dostępnego online. Wybór narzędzi służących do korzystania z repozytorium to sprawa indywidualna każdego członka zespołu, ponieważ nie ma ona wpływu na sam projekt.

### **4.2 Zarządzanie zespołem**

Trello - Kanban Board

### **4.3 Środowisko**

Visual Studio

## **5 Aplikacja**

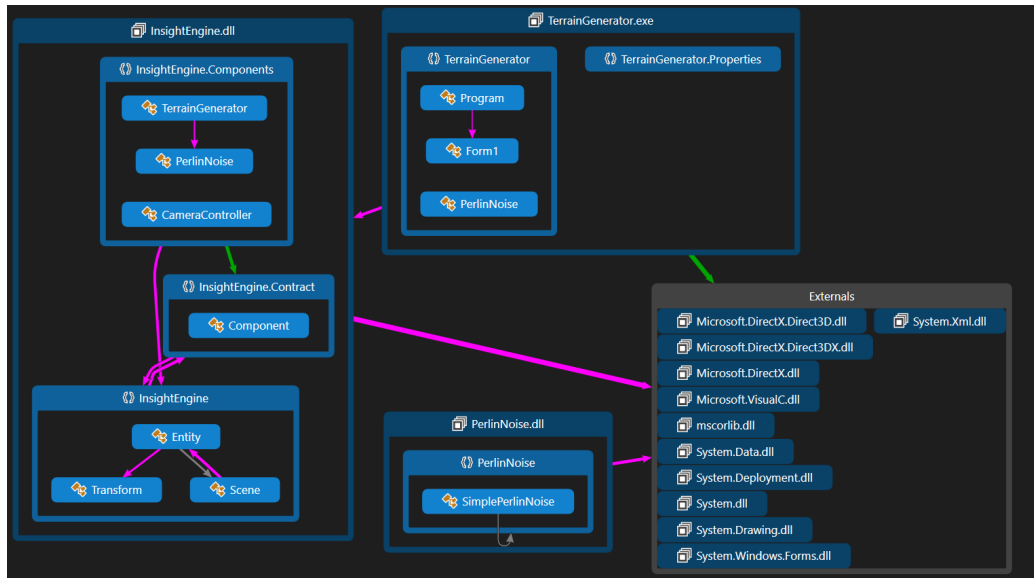
### **5.1 Architektura**

Wzorce architektoniczne

### **5.2 Struktury danych**

Dane przechowujemy ...

## 5.3 Schemat graficzny struktury systemu



## 5.4 Komunikacja między modułami

Jak to wszystko się komunikuje

# 6 API

## 6.1 Perlin

Tworzymy obiekt `PerlinNoise`, który wymaga podania rozmiaru i wymiarów. Pobieranie wartości za pomocą metody: `CalculatePerlin(x,y)`;