



# **Grafika komputerowa**

## **Wyścigi samochodowe 3D**

### **Dokumentacja**

**Szymon Adach**

**Wersja 1.2**

**16.01.2017**

# Spis treści

1.Specyfikacja.....	3
1.1 Opis biznesowy.....	3
1.2 Wymagania funkcjonalne.....	4
1.3 Wymagania нефункционалне.....	6
1.4 Harmonogram projektu.....	7
1.5 Architektura rozwiązania.....	8
2.Dokumentacja końcowa (powykonawcza).....	9
2.1 Wymagania systemowe.....	9
2.2 Biblioteki wraz z określeniem licencji.....	9
2.3 Instrukcja instalacji.....	9
2.4 Instrukcja uruchomienia.....	9
2.5 Instrukcja użycia.....	9
2.6 Instrukcja utrzymania.....	10
2.7 Raport odstępstw od specyfikacji wymagań:.....	10

Data	Autor	Opis zmian	Wersja
14.12.2016	Szymon Adach	Pierwsza wersja dokumentu.	1.0
16.01.2017	Szymon Adach	Dokumentacja końcowa.	1.1

# 1. Specyfikacja

## 1.1 Opis biznesowy

Niniejsza aplikacja zapewnia trójwymiarową symulację wyścigów samochodowych, która odbywa się na torze. Symulacja przedstawiona jest użytkownikowi jako scena renderowana w trzech wymiarach.

Użytkownik aplikacji ma do wyboru kilka kamer, z czego podstawowe trzy to:

- Kamera nieruchomo obserwująca scenę
- Kamera nieruchoma śledząca samochód
- Kamera umieszczona za samochodem i poruszająca się wraz z nim

Kamery związane z obiektem ruchomym w realistyczny sposób odwzorowują zmiany kierunku ruchu, objawiające się występowaniem przyspieszenia dośrodkowego (występuje bezwładność kamery). Ponadto możliwa jest zmiana wychylenia kamery w trakcie działania symulacji za pomocą wybranego klawisza. Po puszczeniu tego klawisza kamera powraca do pierwotnego ustawienia.

Oświetlenie sceny realizowane jest w sposób określony przez użytkownika, któremu pozostawiono wybór modelu oświetlenia spośród poniższych:

- Model Phong
- Model Blinna

Ponadto doużytkownika należy wybór ustawienia aktualnego cieniowania sceny. Do wyboru ma on następujące opcje:

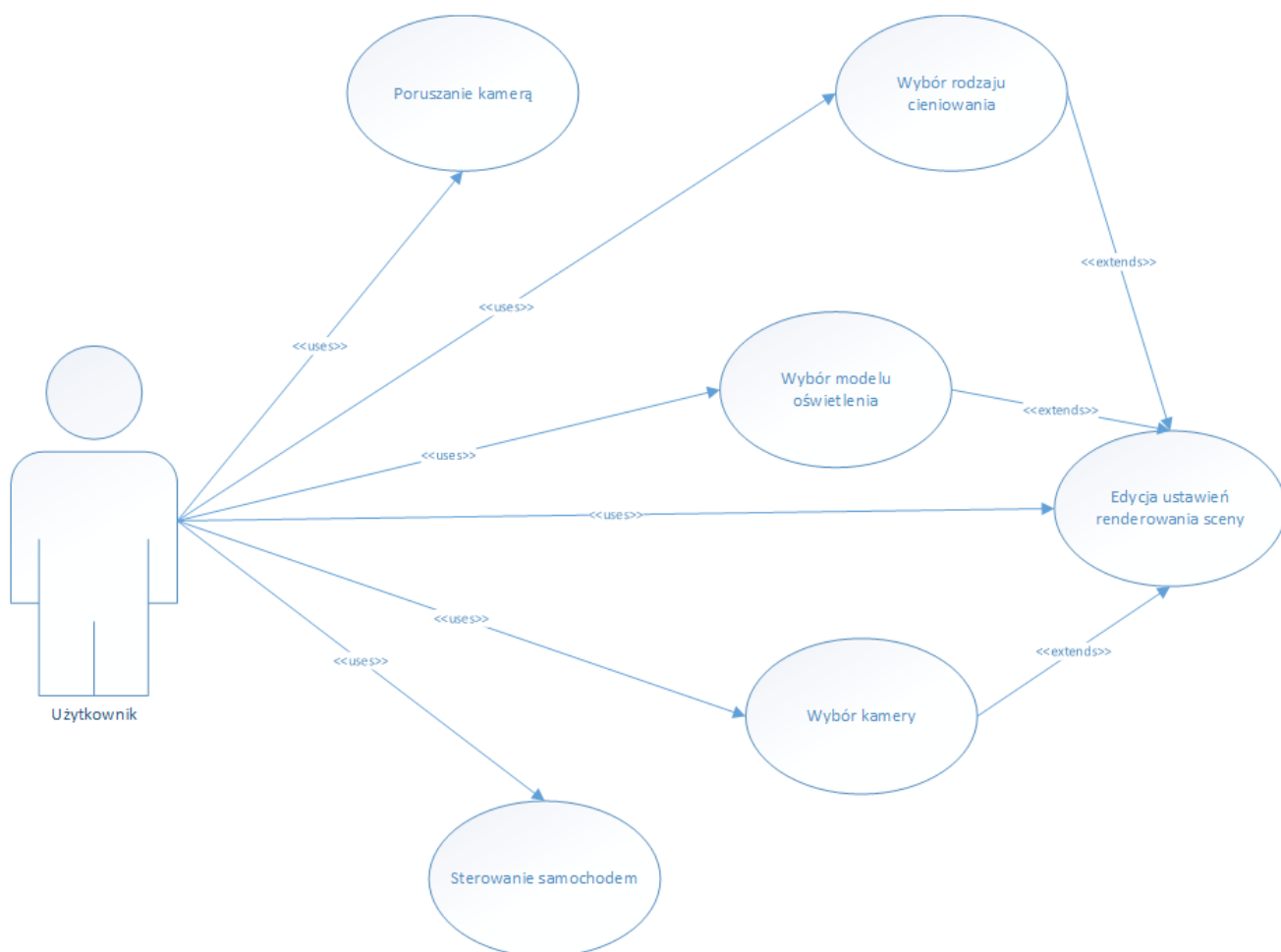
- cieniowanie stałe
- cieniowanie Gourauda
- cieniowanie Phong

Zmiana trybu spotyka się z natychmiastowym skutkiem w postaci zmiany wyglądu renderowanej sceny.

Poza powyższymi opcjami, przy pomocy klawiatury użytkownik może sterować jednym z pojazdów na torze. Wyświetlana scena zawiera oprócz niego kilka samochodów-rywali, nie została jednakże zaimplementowana interakcja między poszczególnymi samochodami, tj. przyjęto, że nie zachodzą między nimi zdarzenia wpływające na tor ruchu czy odkształcenia modeli.

## 1.2 Wymagania funkcjonalne

Na poniższym rysunku przedstawiono w postaci diagramu UML zbiór przypadków użycia aplikacji dla dowolnego użytkownika aplikacji:



Rysunek 1. Diagram przypadków użycia

User stories:

Interfejs użytkownika:

**1. Start symulacji**

Użytkownik w momencie, gdy symulacja jest w stanie pauzy, wciska przycisk Start, aby obejrzeć ruchomą trójwymiarową scenę – wyścig samochodów.

**2. Pauza symulacji**

Użytkownik w momencie, gdy symulacja jest w stanie działania, wciska przycisk Pauza, aby zatrzymać renderowanie.

**3. Wybór kamery**

Użytkownik z rozwijanego menu wybiera kamer dla sceny..

**4. Wybór modelu oświetlenia**

Użytkownik z rozwijanego menu wybiera model oświetlenia sceny.

**5. Wybór trybu cieniowania**

Użytkownik z rozwijanego menu wybiera tryb cieniowania sceny.

**6. Sterowanie samochodem**

Użytkownik w czasie, gdy symulacja jest w stanie działania, może sterować ruchem samochodu za pomocą strzałek na klawiaturze.

**Tabela 1 Opisy przypadków użycia dla aktora – sędziego**

ID	Aktor	Nazwa	Opis	Odpowiedź systemu
USR	Dowolny użytkownik	Wybór kamery	Wybór rodzaju kamery z odpowiedniego wpisu w menu aplikacji.	Natychmiastowa zmiana kamery renderowanej sceny.
		Wybór modelu oświetlenia	Wybór modelu oświetlenia (Blinna albo Phong) z odpowiedniego wpisu w menu aplikacji.	Natychmiastowa zmiana modelu oświetlenia renderowanej sceny.
		Wybór trybu cieniowania	Wybór trybu cieniowania (stałe/Gourauda/Phonga) z odpowiedniego wpisu w menu aplikacji.	Natychmiastowa zmiana trybu cieniowania renderowanej sceny.
		Start/Pauza symulacji	Przyciśnięcie przycisku start/pauza w menu aplikacji.	Natychmiastowe wystartowanie/pauzowanie renderowania sceny.
		Sterowanie samochodem	Sterowanie za pomocą strzałek klawiatury wpływa na ruch pojazdu.	Natychmiastowa reakcja samochodu odpowiadająca wciśniętymu klawiszowi.

Sumaryczna lista ról użytkowników występujących w historiach użytkownika:

<b>Dowolny użytkownik</b>	Każdy kto posiada dostęp do aplikacji.
---------------------------	--

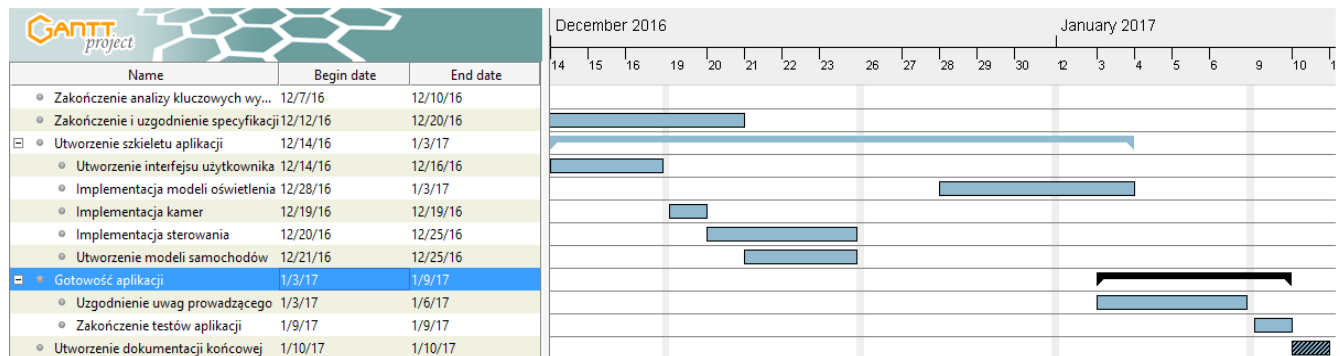
## 1.3 Wymagania нефункционалне

**Tabela 2 Lista wymagań нефункционалных**

Obszar wymagań	Nr wymagania	Opis
Użyteczność	1	Wszystkie funkcjonalności aplikacji dostępne dla użytkownika muszą mieścić się na pojedynczym ekranie przy rozdzielczości 1920x1080.
	2	Interfejs użytkownika musi być w języku angielskim.
Niezawodność	3	Aplikacja nie powinna wyświetlać artefaktów graficznych w przypadku błędów.
Wydajność	4	Aplikacja powinna renderować scenę z wydajnością powyżej 10 FPS (klatek na sekundę).
	5	Aplikacja powinna reagować na sterowanie użytkownika z opóźnieniem nieprzekraczającym 200ms.
Utrzymanie	6	Aplikacja powinna zachować zgodność z kartami graficznymi najpopularniejszych producentów (nVidia, AMD, Intel).
	7	Aplikacja powinna działać na systemach Windows 7 i nowszych.

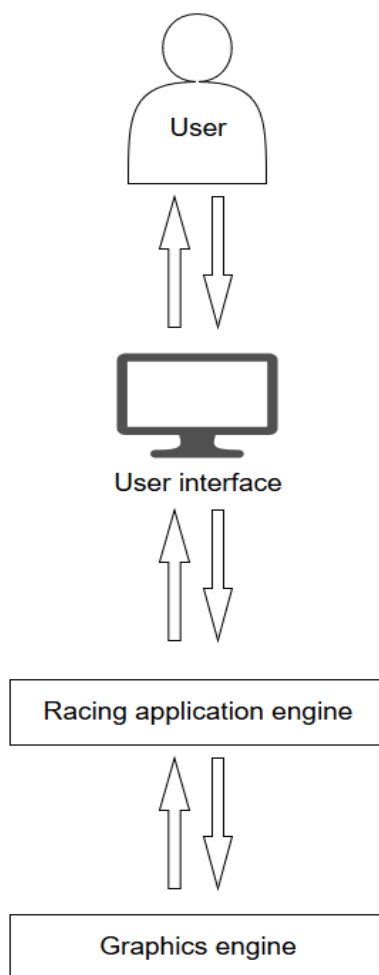
## 1.4 Harmonogram projektu

Planowana jest realizacja projektu zgodnie z następującym harmonogramem:



Rysunek 2. Harmonogram projektu

## 1.5 Architektura rozwiązania



**Rysunek 3. Schemat architektury rozwiązania.**

Użytkownik za pomocą interfejsu graficznego oraz klawiatury wpływa na wygląd sceny, która wyświetlana jest na monitorze. Zmiany wprowadzane przez użytkownika przetwarzane są przez silnik aplikacji i na ich podstawie generowany jest obraz. Renderowany jest on z ustaloną w specyfikacji biblioteką.



## 2. Dokumentacja końcowa (powykonawcza)

### 2.1 Wymagania systemowe

Na maszynie klienta wymagany jest system Windows w wersji 7 lub nowszej oraz instalacja DirectX w najnowszej dostępnej dla systemu wersji. Instalator dostarczany jest przez producenta wraz ze wszystkimi komponentami wymaganymi do uruchomienia aplikacji.

Wymagana jest nowoczesna karta graficzna, obsługująca PixelShader w wersji co najmniej 3.0 oraz VertexShader w wersji co najmniej 3.0.

### 2.2 Biblioteki wraz z określeniem licencji

nr.	Komponent i wersja	Opis	Licencja
1	Microsoft XNA 4.0	Biblioteka graficzna wykorzystana do generowania obrazu.	EULA

### 2.3 Instrukcja instalacji

Aby zainstalować aplikację należy uruchomić dostarczony instalator i stosować się do instrukcji na ekranie.

### 2.4 Instrukcja uruchomienia

Po zainstalowaniu aplikacji należy uruchomić odpowiedni plik wykonywalny.

### 2.5 Instrukcja użycia

Aplikacja uruchamia się w trybie pełnoekranowym w rozdzielczości 1920x1080 w trybie pauzy. Tryb pauzy wstrzymuje ruch wszystkich pojazdów na scenie. Do włączania pauzy służy klawisz **F1**, do wyłączania klawisz **F2**. Samochód użytkownika poruszać się może po przygotowanym obszarze, który ograniczony jest specjalnym płotem. Próba wyjechania poza obszar powoduje przeniesienie samochodu użytkownika na start.

Ponadto w lewym dolnym rogu wyświetla jest informacja o obecnie używanym modelu oświetlenia oraz cieniowania.

Domyślnym trybem cieniowania jest cieniowanie płaskie, a domyślnym trybem oświetlenia jest model Phong. W dowolnym momencie użytkownik może zmienić wybrany model. Służą do tego następujące klawisze:

**1** – model oświetlenia Phong

**2** – model oświetlenia Blinna-Phong

**Q** – model cieniowania płaskiego

**W** - model cieniowania Gouraud

**E** – model cieniowania Phong

Zgodnie z wymaganiami, użytkownik może obserwować scenę za pomocą trzech kamer:

1. Kamera poruszająca się za samochodem – istnieje możliwość wychylania jej w płaszczyźnie poziomej przy pomocy klawiszy **J** (wychylenie w lewo), **L** (wychylenie w prawo), **K** (wychylenie zerowe – powrót kamery do domyślnej pozycji). Użytkownik wybiera ją poprzez wciśnięcie klawisza **I**.
2. Kamera nieruchoma, obserwująca pozycję startową samochodu na linii startu. Wybierana klawiszem **P**.
3. Kamera nieruchoma, śledząca ruch samochodu. Wybierana klawiszem **O**.

Sterowanie samochodem odbywa się za pomocą przycisków strzałek:

**Strzałka w lewo** – skręcanie w lewo

**Strzałka w prawo** – skręcanie w prawo

**Strzałka w górę** – przyspieszanie samochodu

**Strzałka w dół** – hamowanie samochodu

Wyłączenie aplikacji następuje po wciśnięciu kombinacji przycisków **Alt+F4**.

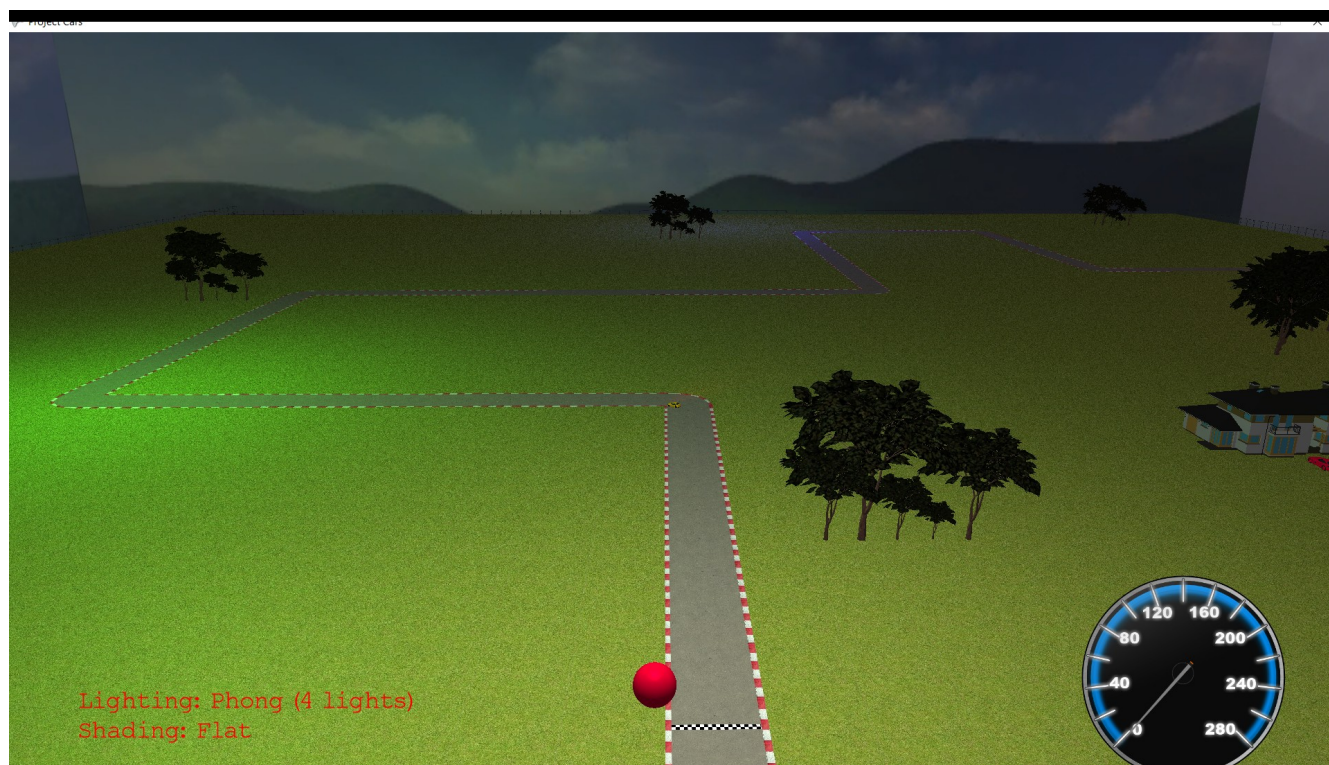
Ponadto dodano możliwość włączania i wyłączania punktowych źródeł światła umieszczonych nad sceną. Ich liczba może zmieniać się w zakresie od 1 do 4 świateł.

Domyślna wartość to 0. Do zmiany liczby świateł służą przyciski **+** (zwiększ o jeden) oraz **-** (zmniejsz o jeden).



**Rysunek 4. Pozycja startowa samochodów**

Powyższy rysunek przedstawia pozycję startową (samochód gracza po lewej, samochód AI po prawej). Widoczny jest również HUD w postaci prędkościomierza w prawym dolnym rogu oraz w informacji o aktualnym modelu oświetlenia i cieniowania.



**Rysunek 5. Kamera śledząca samochód gracza (przycisk O)**





Rysunek 6. Ogrodzenie tworzące granicę sceny



Rysunek 7. Obiekty na scenie przy długiej prostej toru

## 2.6 Instrukcja utrzymania

Nie jest wymagane podejmowanie żadnych kroków przez odbiorcę w celu utrzymywania aplikacji.

## 2.7 Raport odstępstw od specyfikacji wymagań

Zmiana sposobu przełączania między wyspecyfikowanymi kamerami oraz trybami oświetlenia i cieniowania.

### **Zmiana:**

Powyższe tryby zmieniane są przy pomocy klawiatury (w sposób opisany w rozdziale poświęconym użytkowaniu aplikacji) zamiast menu w postaci graficznego interfejsu użytkownika.

### **Uzasadnienie:**

Implementacja menu wymagałaby od producenta stosowania zewnętrznych bibliotek. Po zapoznaniu się z ich ofertą, producent uznał, że wiązałoby się to z niewspółmiernym do efektu końcowego rozrostem kodu aplikacji, pojawieniem się dodatkowych błędów oraz negatywnym wpływem na wydajność (w postaci spadku liczby klatek na sekundę). Rozsądnym wydało się pozostawienie funkcjonalności ze zmienionym sposobem nawigacji.