

Wydział Matematyki i Nauk Informatycznych Politechniki
Warszawskiej



Symulacja wyścigów samochodowych w 3D

Grafika Komputerowa I

Autor: **Maciej Grzeszczak**

v1.1

10 grudnia 2016r.

Spis treści

1	Specyfikacja	2
1.1	Opis biznesowy	2
1.2	Wymagania funkcjonalne	2
1.3	Wymagania нефункционалне	4
1.4	Harmonogram projektu	5
1.5	Architektura rozwiązania	6
1.6	Dokumentacja powykonawcza	7
1.6.1	Wymagania systemowe	7
1.6.2	Biblioteki i licencje	7
1.6.3	Instrukcja instalacji	7
1.6.4	Instrukcja uruchomienia	7
1.6.5	Instrukcja użycia	8
1.6.6	Instrukcja utrzymania	8
1.6.7	Raport odstępstw od specyfikacji wymagań	8
1.7	Lista użytych skrótów	9
1.8	Bibliografia	9

Tablica 1: Lista zmian

Data	Autor	Opis zmiany	Wersja
17.12.2016	Maciej Grzeszczak	Opis architektury oraz rozwinięcie opisu biznesowego	1.1
10.12.2016	Maciej Grzeszczak	Pierwsza wersja dokumentu	1.0

1. Specyfikacja

1.1. Opis biznesowy

Niniejszy program to aplikacja przeglądarkowa, wykorzystująca technologię WebGL w postaci biblioteki three.js. Będzie ona zoptymalizowana głównie pod przeglądarkę Mozilla Firefox.

Aplikacja jest symulacją wyścigów samochodowych w 3D. Umożliwia użytkownikowi prowadzenie pojazdu za pomocą odpowiednich klawiszy i ściganie się z samochodami sterowanymi przez komputer. Dany będzie jeden, z góry określony tor po którym będzie można jeździć, sam gracz będzie mógł również go opuścić i poruszać się po całej mapie. Samochód będzie przyspieszał, hamował i skręcał w zależności od wciśniętych klawiszy, będzie można również włączyć bieg wsteczny.

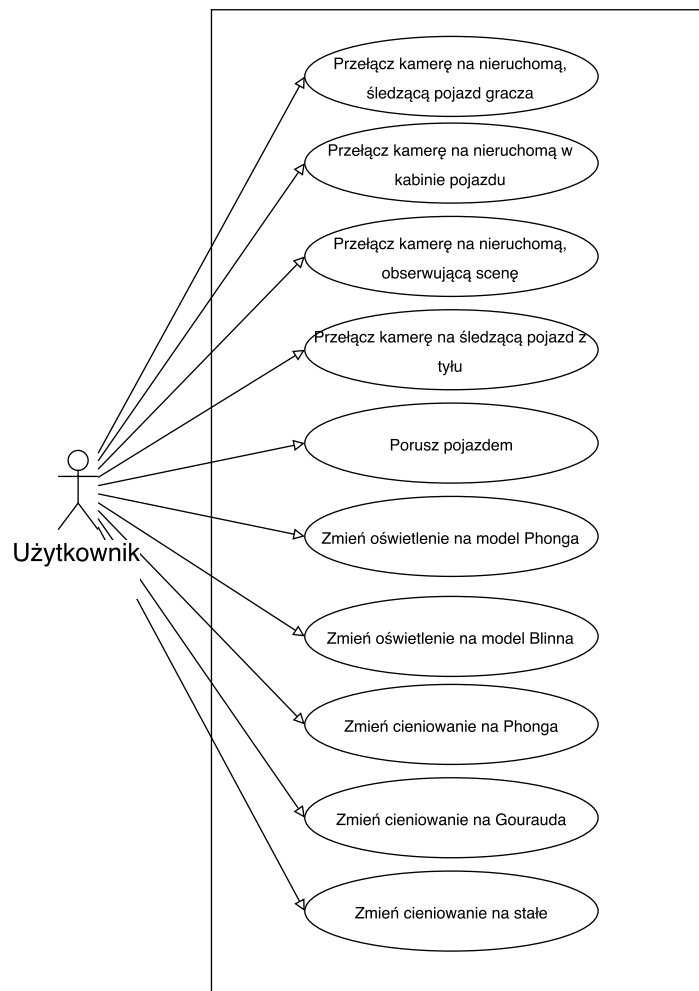
Aplikacja służy również jako pokaz różnych modeli oświetlenia oraz cieniowania. Gracz będzie mógł za pomocą odpowiednich przycisków zmienić obecny model oświetlenia na model Phong'a lub Blinn'a, jak również wybrać jeden z trzech trybów cieniowania (Phong'a, Gouraud'a, stałe).

Oprócz tego, dostępne będą cztery kamery, umożliwiające obserwowanie wyścigów z różnych perspektyw, między innymi ze środka kabiny oraz zza pojazdu, która będzie płynnie poruszać się z pojazdem, wykorzystując do tego interpolację.

W planach jest również wprowadzenie większej ilości torów, możliwości ich wczytywania z tekstur, wprowadzenie nierówności terenu i dostosowywania się auta do nich, oraz użycie oddzielnych modeli do kół pojazdów, co umożliwiłoby ich kręcenie się oraz skręcanie.

1.2. Wymagania funkcjonalne

Poniższy rysunek w postaci diagramu UML przedstawia możliwe przypadki użycia systemu przez użytkownika:



Tablica 2: Opisy przypadków użycia dla użytkownika

Aktor	Nazwa	Opis	Odpowiedź systemu
Użytkownik	Przełącz kamerę na nieruchomą, z kabiny pojazdu.	Zmiana położenia kamery na kabinę pojazdu, skierowaną na drogę przed pojazdem.	Natychmiastowa zmiana pozycji kamery.
	Przełącz kamerę na nieruchomą, obserwującą całą scenę.	Zmiana położenia kamery na pozycję umożliwiającą obserwowanie całej sceny z oddali.	Natychmiastowa zmiana pozycji kamery.
	Przełącz kamerę na nieruchomą, śledzącą pojazd gracza.	Zmiana położenia kamery na pozycję, która jest nieruchoma i śledzi pojazd gracza.	Natychmiastowa zmiana pozycji kamery.
	Przełącz kamerę na śledzącą pojazd z tyłu.	Zmiana położenia kamery na pozycję za pojazdem, która porusza się za pojazdem płynnie i wykorzystuje do tego interpolację	Natychmiastowa zmiana pozycji kamery.
	Porusz pojazdem.	Przemieszczenie się pojazdu pod wpływem wciśnięcia odpowiednich klawiszy.	Przemieszczenie się pojazdu.
	Zmień oświetlenie na model Phong.	Zmiana obecnego modelu oświetlenia na model Phong.	Natychmiastowa zmiana modelu oświetlenia na model Phong.
	Zmień oświetlenie na model Blinn.	Zmiana obecnego modelu oświetlenia na model Blinn.	Natychmiastowa zmiana modelu oświetlenia na model Blinn.
	Zmień cieniowanie na Phong.	Zmiana obecnego trybu cieniowania na cieniowanie Phong.	Natychmiastowa zmiana trybu cieniowania na cieniowanie Phong.
	Zmień cieniowanie na Gouraud.	Zmiana obecnego trybu cieniowania na cieniowanie Gouraud.	Natychmiastowa zmiana trybu cieniowania na cieniowanie Gouraud.
	Zmień cieniowanie na stałe.	Zmiana obecnego trybu cieniowania na stałe.	Natychmiastowa zmiana trybu cieniowania na cieniowanie stałe.

1.3. Wymagania нефункционалне

Poniżej przykładowe wymagania нефункционалне pogrupowane w poszczególne kategorie URPS.

Tablica 3: Lista wymagań niefunkcjonalnych

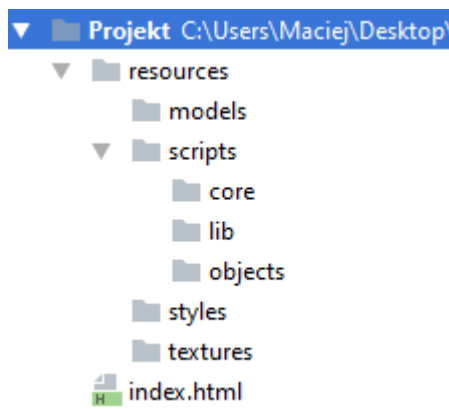
Obszar wymagań	Lp	Opis
Użyteczność	1	Aplikacja będzie działała na przeglądarce Mozilla Firefox dla każdej rozdzielczości powyżej 800x600.
Niezawodność	2	Aplikacja będzie dostępna 24/7 pod podanym adresem.
Wydajność	3	Aplikacja będzie utrzymywać minimalny poziom 15 FPS (klatek na sekundę).
Utrzymanie	4	Wraz z aplikacją zostaje dostarczona instrukcja użytkownika.

1.4. Harmonogram projektu

Implementacja projektu zostanie podzielona na dwie fazy:

1. **Faza tworzenia sceny (14 dni)** - stworzenie świata wraz z obiektami (pojazdami), implementacja poruszania się pojazdem oraz poruszania się i zmiany pozycji kamery.
2. **Faza implementacji poszczególnych modeli oświetlenia oraz cieniowania (7 dni)** - implementacja modeli oświetlenia Phong'a i Blinn'a oraz cieniowań: stałego, Phong'a i Gouraud'a.

1.5. Architektura rozwiązania



Powyższe zdjęcie przedstawia szkielet architektury projektu. Poniżej znajduje się opis kolejnych elementów.

1. **index.html** - główny i jedyny plik html, w którym zagnieżdżony będzie HTML5 Canvas.
2. **resources** - folder, w którym znajdują się wszystkie zasoby wykorzystywane przez aplikację:
 - (a) **models** - folder z modelami 3D używanymi przez aplikację
 - (b) **scripts** - folder ze skryptami Javascript
 - i. **core** - folder zawierający pliki .js dotyczące szkieletu działania aplikacji, czyli pętli gry, liczenia fizyki, renderowania, obsługi interakcji użytkownika.
 - ii. **objects** - folder w którym znajdują się pliki .js definiujące wszystkie obiekty wykorzystywane w aplikacji, między innymi pojazd, kamery, mapy.
 - iii. **lib** - folder w którym znajdują się pliki .js zewnętrznych bibliotek.
 - (c) **styles** - folder z wszystkimi plikami .css
 - (d) **textures** - folder z teksturami wykorzystywanymi przez aplikację

Oprócz tego projekt będzie oparty na wzorcu modułów, który pomaga w organizacji całości kodu oraz uzyskania tzw. 'loose coupling', czyli jak najmniejszej zależności pomiędzy poszczególnymi częściami kodu.

1.6. Dokumentacja powykonawcza

1.6.1. Wymagania systemowe

Do używania aplikacji lokalnie wymagane są

1. przeglądarka internetowa z obsługą HTML5 oraz WebGL (np. Mozilla Firefox, Chrome)
2. oprogramowanie umożliwiające postawienie serwera HTTP (np. Python 2.7.x/3.x.x albo PHP)

1.6.2. Biblioteki i licencje

Tablica 4: Tabela bibliotek zastosowanych w projekcie

Lp	Komponent i wersja	Opis	Licencja
1	glMatrix 2.0	Biblioteka do operacji na wektorach i macierzach	Custom

1.6.3. Instrukcja instalacji

Aplikacja nie wymaga instalacji. W przypadku gdy nie posiada się oprogramowania do postawienia lokalnego serwera HTTP można pobrać i zainstalować zgodnie z instrukcjami np. PHP ze strony <http://windows.php.net/download>.

1.6.4. Instrukcja uruchomienia

Aby uruchomić aplikację należy postawić lokalny serwer HTTP w folderze 'Projekt', gdzie znajduje się plik index.html. Aby to zrobić używając PHP, należy:

1. Uruchomić wiersz poleceń.
2. Przejść do folderu 'Projekt' aplikacji.
3. Wykonać komendę 'php -S localhost:8080'.

Następnie należy uruchomić swoją ulubioną przeglądarkę internetową i otworzyć stronę 'localhost:8080'.

1.6.5. Instrukcja użycia

Poniżej znajdują się opis wszystkich klawiszy i ich funkcjonalności w aplikacji:

1. W - gaz
2. S - hamulec
3. A - skręcenie kół w lewo
4. D - skręcenie kół w prawo
5. R - zmiana pomiędzy biegiem wstecznym a normalnym
6. L - włączenie/wyłączenie reflektorów
7. P - włączenie modelu oświetlenia Phong
8. B - włączenie modelu oświetlenia Blinn
9. 8 - włączenie cieniowania stałego
10. 9 - włączenie cieniowania Gourauda
11. 0 - włączenie cieniowania Phong
12. M - stwórz przeciwnika jeżdżącego po całym torze
13. Enter - rozpocznij/zakończ nagrywanie trasy samochodu - po zakończeniu stworzony zostanie przeciwnik naśladujący ruch gracza po nagrany torze.

1.6.6. Instrukcja utrzymania

Aby wyłączyć aplikację należy zamknąć przeglądarkę oraz wyłączyć serwer HTTP.

Uruchamianie aplikacji zostało przedstawione we wcześniejszym punkcie dokumentacji.

1.6.7. Raport odstępstw od specyfikacji wymagań

Aplikacja spełnia wszystkie wymagania przedstawione wcześniej w specyfikacji. W trakcie realizacji projektu zostały dodane następujące elementy:

1. Ruchome koła
2. Kierownica
3. Lusterka
4. Reflektory samochodu
5. Skybox

1.7. Lista użytych skrótów

Tablica 5: Lista użytych skrótów

Lp	Skrót	Opis
1	HTML	HyperText Markup Language
2	HTTP	HyperText Transfer Protocol
3	PHP	Hypertext Preprocessor

1.8. Bibliografia

1. Strona biblioteki glmatrix: <http://glmatrix.net/>