

Proceduralne generowanie planet w czasie rzeczywistym  
w dynamicznie zmieniającej się skali

# Wstępna analiza wymagań

---

MATEUSZ CHECHLIŃSKI

*Wersja 1.0*

1 listopada 2014

Modyfikacje dokumentu			
Wersja	Data	Autor	Opis
1.0	1 listopada 2014	Mateusz Chechliński	Wstępna analiza wymagań

## Streszczenie

Celem projektu jest stworzenie aplikacji webowej służącej do dynamicznego generowania planet, a następnie prezentowania ich w formie grafiki trójwymiarowej.

Użytkownik otrzyma możliwość płynnej nawigacji pomiędzy planetami, oglądania planet z dowolnego kąta i z dowolnej odległości.

Tylko planety w danej chwili widoczne dla użytkownika będą przechowywane w pamięci, dla pozostałych zostanie zapisane jedynie ziarno (seed).

Aplikacja będzie działać w oparciu o bibliotekę Three.js bazującą na WebGL i będzie wspierać wszystkie przeglądarki, systemy operacyjne i karty graficzne wspierane przez w/w biblioteki.

Projekt zostanie zrealizowany metodą iteracyjną:

- **Iteracja 1 - do 28 listopada 2014** - Generowanie powierzchni pojedynczej planety
- **Iteracja 2 - do 12 grudnia 2014** - Generowanie wielu planet i gwiazd
- **Iteracja 3 - do 9 stycznia 2014** - Dynamiczne dostosowywanie poziomu szczegółów
- **Iteracja 4 - do 23 stycznia 2014** - Realistyczne efekty graficzne, w tym cieniowanie

# Spis treści

Streszczenie	3
Spis treści	4
1. Wstęp	5
1.1 Cel projektu	5
1.2 Wykorzystane technologie	5
2. User stories	6
2.1 Jako użytkownik...	6
3. Analiza funkcjonalna	6
4. Harmonogram	7
5. Podział prac	8
6. Słownik pojęć	8

# 1. Wstęp

Poniższy dokument stanowi wstępną analizę wymagań projektu, w tym technologii w oparciu o które powstanie. Wyróżniono 3 kamienie milowe, dokonano wstępnego podziału obowiązków między członków zespołu i zaplanowano podział prac na poszczególne etapy.

## 1.1 Cel projektu

Celem projektu jest stworzenie aplikacji webowej służącej do dynamicznego generowania planet, a następnie prezentowania ich w formie grafiki trójwymiarowej. Kluczowymi cechami aplikacji są:

1. Generowanie powierzchni planet na podstawie ziarna (Seed) z wykorzystaniem pewnej nielosowej funkcji.
2. Tak wygenerowane planety są następnie umieszczane w przestrzeni kosmicznej, wśród innych planet i krążąc wokół gwiazd.
3. Użytkownik aplikacji może w pewnym zakresie zbliżać się lub oddalać od powierzchni każdej planety. W zależności od odległości użytkownika od powierzchni, prezentowany będzie odpowiedni poziom szczegółów (Level of Details).
4. Prezentowane układy planetarne charakteryzują się realizmem.
5. Użytkownik może płynnie nawigować pomiędzy planetami/gwiazdami oraz regulować odległość od ich powierzchni - aplikacja działać będzie w czasie rzeczywistym, zarówno generowanie planet jak i regulowanie poziomu szczegółów ma się odbywać niezauważalnie dla użytkownika.
6. Aplikacja będzie wieloplatformowa, tj. będzie można ją uruchomić w każdej przeglądarce wspierającej WebGL na komputerze wyposażonym w kompatybilną kartę graficzną.

## 1.2 Wykorzystane technologie

1. **Three.js** - jest lekka biblioteka JavaScript, wspierająca wiele przeglądarek i działająca w oparciu o WebGL. Pozwala na tworzenie trójwymiarowych animacji dzięki wykorzystaniu w przeglądarce mocy obliczeniowej karty graficznej.
2. **qUnit** - framework służący do testów jednostkowych dla aplikacji napisanych w JavaScript

## 2. User stories

### 2.1 Jako użytkownik...

#	Chcę...	Żeby...	Kryteria akceptacji
1.	mieć możliwość nawigacji	przyglądać się różnym planetom / gwiazdom	Użytkownik może w sposób intuicyjny przemieszczać się pomiędzy planetami / gwiazdami.  Nawigacja odbywa się bez oczekiwania na wygenerowanie planety.
2.	móc zmieniać odległość kamery od powierzchni	przyglądać się im z dowolnej odległości	Użytkownik może płynnie przybliżać i oddalać kamerę.  W zależności od odległości prezentowany jest odpowiedni poziom szczegółów.
3.	móc krążyć wokół planet / gwiazd	przyglądać się im z każdej perspektywy	Użytkownik może w intuicyjny sposób zmienić położenie kamery względem osi planety lub gwiazdy.
4.	mieć do dyspozycji bardzo duży wszechświat		Planety niewidoczne dla użytkownika są usuwane z pamięci (oprócz ich ziarna).  Planety widoczne są dynamicznie generowane i wyświetlane
5.	mieć poczucie realizmu		Planety i gwiazdy są generowane w taki sposób, by naśladować cechy planet / gwiazd znanych naukowcom.

## 3. Analiza funkcjonalna

Aplikacja będzie działać na każdej przeglądarce wspieranej przez bibliotekę WebGL. Obecnie są to:

- Internet Explorer 11 lub nowszy
- Firefox w wersji 31 lub nowszej
- Chrome w wersji 31 lub nowszej
- Safari w wersji 5.1 lub nowszej
- Opera w wersji 25 lub nowszej
- Safari dla iOS w wersji 8.0 lub nowszej
- Chrome dla Android w wersji 38 lub nowszej

Dodatkowo, do poprawnego działania wymagana jest karta graficzna wspierająca WebGL. Wszystkie nowoczesne karty graficzne z najnowszymi sterownikami powinny wspierać w/w bibliotekę.

Obsługiwane systemy operacyjne to:

- Windows 7/Windows 8 (zalecany Windows 7 lub nowszy)
- Mac OS 10.6 lub nowszy (zalecany 10.8 lub nowszy)
- Linux
- System operacyjny Chrome

W zależności od wydajności konkretnego komputera, płynność działania aplikacji może się różnić, w szczególności może spaść poniżej 30 klatek/sek., nawet jeśli spełnione zostały minimalne wymagania określone powyżej.

## 4. Harmonogram

Projekt zostanie zrealizowany metodą iteracyjną.

### Iteracja 1 - do 28 listopada 2014

1. Generowanie powierzchni pojedynczej planety
2. Przemieszczanie kamery zawieszanej na orbicie planety

### Iteracja 2 - do 12 grudnia 2014

1. Generowanie wielu planet i gwiazd oraz ich relacji w przestrzeni kosmicznej
2. Nawigacja pomiędzy poszczególnymi ciałami niebieskimi

### Iteracja 3 - do 9 stycznia 2014

1. Dynamiczne dostosowywanie poziomu szczegółów (Level of Details)
2. Przybliżanie i oddalanie kamery od powierzchni planety

### Iteracja 4 - do 23 stycznia 2014

1. Realistyczne efekty graficzne, w tym cieniowanie
2. Maksymalna poprawa wydajności mierzonej liczbą klatek na sekundę

## 5. Podział prac

### Jakub Skąlecki

- Poziom szczegółów (Level of Details)
- Wykorzystanie Shaderów

### Piotr Leniec

- Ogólna architektura aplikacji
- Komunikacja CPU - GPU

### Mateusz Chechliński

- Procedury generujące planety
- Testy jednostkowe
- Dokumentacja

## 6. Słownik pojęć

**Kamień milowy** - istotny etap projektu. Aplikacja po osiągnięciu kamienia milowego powinna nadawać się do użytkowania, posiadając ograniczony, ale w pełni sprawny, zestaw funkcjonalności

**Ziarno (Seed)** - pewna (być może losowa) liczba, na podstawie której generowana będzie planeta. Dla danego ziarna, wygenerowana planeta zawsze będzie dokładnie taka sama.

**Shader** - krótki program komputerowy w typowych zastosowaniach opisujący właściwości pikseli i wierzchołków. W tym projekcie wykorzystywany do generowania powierzchni planet i gwiazd oraz do kontrolowania poziomu szczegółów.

**Poziom szczegółów (Level of Details)** - dokładność z jaką renderowana jest powierzchnia planety. Im mniejsza odległość obserwatora od powierzchni, tym dokładniejszy prezentowany obraz.