

# **SPECYFIKACJA WYMAGAŃ GRY**

## ***„Flappy Man”***

**Opracowanie : Adam Artur Gliszczyński**  
**Utworzenie dokumentu : 13.11.2016**  
**Ostatnia modyfikacja : 8.01.2016**  
**Aktualna wersja dokumentu : v 1.1**

## **SPIS TREŚCI :**

### **Rozdział 1 : OPIS OGÓLNY**

- 1.1 Nazwa programu
- 1.2 Poruszany problem
- 1.3 Użytkownik docelowy

### **Rozdział 2 : OPIS FUNKCJONALNOŚCI**

- 2.1 Jak korzystać z programu
- 2.2 Uruchamianie programu
- 2.3 Możliwości programu
- 2.4 Wymagania нефункционалне

### **Rozdział 3 : FORMAT DANYCH I STRUKTURY PLIKÓW**

- 3.1 Przechowywane dane w programie
- 3.2 Dane wejściowe
- 3.3 Dane wyjściowe
- 3.4 System operacyjny
- 3.5 Przykłady rozwiązań graficznych

### **Rozdział 4 : SCENARIUSZ DZIAŁANIA PROGRAMU**

- 4.1 Scenariusz szczegółowy
- 4.2 Ekrany działania programu
  - 4.2.1 Menu główne
  - 4.2.2 Okno z wynikami
  - 4.2.3 Exit

### **Rozdział 5 : TESTOWANIE**

- 5.1 Przebieg testowania

### **Rozdział 6 : DODATKI**

- 6.1 Harmonogram
- 6.2 Opis zmian w projekcie

## **Rozdział 1 : OPIS OGÓLNY**

### **1.1 Nazwa programu**

Nazwa programu to „Flappy Man”. Nazwa inspirowana pierwotnym, oryginalnym programem „Flappy Bird”.

### **1.2 Poruszany problem**

Program, jest prostą grą zręcznościową. Ma na celu dostarczanie rozrywki oraz możliwie rozwijanie koordynacji wzrokowo - ruchowej.

### **1.3 Użytkownik docelowy**

Program przeznaczony jest dla każdej grupy wiekowej, narodowościowej, etnicznej i religijnej.

## **Rozdział 2 : OPIS FUNKCJONALIŚCI**

### **2.1 Jak korzystać z programu**

Po uruchomieniu programu, wybraniu polecenia „START” z menu głównego następuje uruchomienie rozrywki. Użytkownik porusza postacią poprzez naciśnięcie klawisza „spacja”. Po każdym naciśnięciu klawisza „spacja” postać wznosi się na określoną wysokość oraz opada zgodnie z uproszczonymi prawami fizycznymi. Zadaniem gracza jest takie sterowanie postacią aby przeszła między przeszkodami nie powodując kolizji z nią ( przeszkodą ) oraz z gruntem. Do prawidłowego działania programu wymagana jest klawiatura, musz/ touchpad oraz głośniki / słuchawki bądź inny sprzęt audio umożliwiający odtwarzanie dźwięku i podłączenie do wyjścia audio w komputerze.

### **2.2 Uruchamianie programu**

Program uruchamia się poprzez podwójne kliknięcie lewym przyciskiem myszy lub touch-padem na plik z nazwą „Flappy Man”.

### **2.3 Możliwości programu**

Program ma za zadanie dostarczanie rozrywki użytkownikowi oraz rozwijanie koordynacji wzrokowo - ruchowej.

### **2.4 Wymagania нефunkcjonalne**

- Cały interfejs programu jest w języku angielskim
- Program napisany w języku C/C++ przy użyciu biblioteki zewnętrznej ALLEGRO 5 oraz IDE Xcode 10.

## Rozdział 3 : FORMAT DANYCH I STRUKTURY PLIKÓW

### 3.1 Przechowywane dane w programie

Program nie będzie przechowywał żadnych danych wprowadzony z zewnątrz przez użytkownika ani pobieranych z sieci zewnętrznych. Program przechowuje tylko dane niezbędne do poprawnego działania dostarczone razem z programem.

**Założenie dodatkowe** : Przechowywanie 5 ostatnich wyników w tabeli wraz z datą oraz godziną osiągnięcia danego rezultatu.

### 3.2 Dane wejściowe

Program pobiera dane w czasie rzeczywistym z klawiatury. Podczas rozgrywki pobierane dane są tylko z klawisza „spacja”. Do obsługi menu głównego oraz okien nie będącym oknem rozgrywki używane są dane myszy lub touchpada.

### 3.3 Dane wyjściowe

Wyjściem programu jest obraz oraz ruch postaci generowany na ekranie. Postacią jest obiekt będący superbohaterem przypominający „supermana”. Tłem jest panorama otwartej przestrzeni naturalnej - lasu oraz łąki. Przeszkoda to dwa obiekty przypominające drewniane bale ustawione pionowo z wolną przestrzenią pomiędzy nimi ( obiektami ) umożliwiające przejście postaci.

**Założenie dodatkowe** : Efekty dźwiękowe

### 3.4 System operacyjny

Program działa na systemie operacyjnym OS X Yosemite i wyższe.

**Założenie dodatkowe** : wersja programu działająca na Windows 7 i wyższe z późniejszymi SP.

### 3.5 Przykłady rozwiązań graficznych



przykładowy  
model postaci



przykładowy  
model przeszkody



przykładowy  
model tła

## Rozdział 4 : SCENARIUSZ DZIAŁANIA PROGRAMU

### 4.1 Scenariusz szczegółowy

Po rozpoczęciu rozgrywki użytkownik porusza postacią za pomocą klawisza „spacja”. Jego zadanie jest takie manipulowanie postacią aby zmieściła się ona (postać) w otworze przeszkody o stałej szerokości ale zmiennym położeniu. Przeszkody pojawiają się w stałej odległości od siebie ale z względnie losowym położeniem otworu umożliwiającemu przejście. Przesuwają się w lewo z prędkością umożliwiającą użytkownikowi swobodną rozgrywkę. Przeszkodą są imitacje drewnianych bali. Użytkownik otrzymuje za każdorazowe przejście przez otwór 1 (jeden) punkt. Licznik punktów znajduje się w lewym górnym rogu okna rozgrywki i z każdorazowym rozpoczęciem nowej rozgrywki jest zerowany. Postać nie porusza się w żadnym kierunku względem tła z wyłączeniem ruchu pionowego. W przypadku kolizji z przeszkodą lub gruntem (podłożem) rozgrywka zostaje przerwana, wyświetla się okno z wynikiem oraz wynik zapisywany jest to tabeli wyników. W przypadku wybrania przez użytkownika przycisku „New Game” rozpoczyna się nowa rozgrywka, w przypadku wybrania „Menu” otwiera się okno menu głównego. Program można zakończyć poprzez wybranie „Exit” w menu główne lub poprzez zamknięcie okna krzyżykiem w prawym górnym rogu okna.

### 4.2 Ekrany działania programu

#### 4.2.1 Menu główne

Menu główne składa się z poleceń :

- „New Game” - rozpoczyna rozgrywkę
- „Score” - otwiera okno z ostatnimi wynikami (**zał. dodatkowe**)
- „Credits” - lista osób, które przyczyniły się do powstania programu (**zał. dod.**)
- „Exit” - wyjście z programu

#### 4.2.2 Okno z wynikami (**zał. dodatkowe**)

Okno z wynikami ma za zadanie wyświetlać ostatnie wyniki razem z data oraz godziną uzyskania rezultatu.

#### 4.2.3 Okno z rozgrywką

W tym oknie jest prowadzona rozgrywka.

#### 4.2.4 Okno z wynikiem

Po zakończeniu rozgrywki wyświetla się napis „Game Over”, wynik użytkownika oraz przyciski :

- „New Game” - rozpoczyna nową rozgrywkę
- „Score” - otwiera okno z ostatnimi wynikami (**zał. dodatkowe**)
- „Credits” - lista osób, które przyczyniły się do powstania programu (**zał. dod.**)
- „Exit” - wyjście z programu

## Rozdział 5 : TESTOWANIE

### 5.1 Przebieg testowania

Testowanie będzie polegać na testach jednostkowych poszczególnych funkcji oraz modułów.

## Rozdział 6 : DODATKI

### 6.1 Harmonogram

tydzień	zadania do wykonania
1	otwarcie projektu, utworzenie niezbędnej grafiki, wczytanie tła do programu
2	wstawienie przeszkód oraz ruch (bez losowości)
3	ruch postaci (pionowy),
4	implementacja losowości przeszkód, kolizje z ziemią oraz przeszkodami
5	licznik punktów, utworzenie menu głównego part 1
6	drobne poprawki, utworzenie menu głównego part 2
7	testy, ewentualne realizowanie założeń dodatkowych
8	o ewentualne realizowanie założeń dodatkowych ,stateczne testy programu, oddanie projektu

### 6.2 Opis zmian w projekcie

1. Zmiana w harmonogramie. Zamiana pomiędzy tygodniem 2 i 4 . (wstawianie przeszkód, ruch postaci)
2. Zmiana w harmonogramie. Zamiana pomiędzy tygodniem 4 i 5 (licznik punktów , losowość przeszkód)
3. Uwzględnienie w harmonogramie implementacji kolizji.
4. Zmiany w projekcie menu - większa przejrzystość i intuicyjność.