

**WYDZIAŁ
ELEKTROTECHNIKI
I INFORMATYKI**
POLITECHNIKI RZESZOWSKIEJ

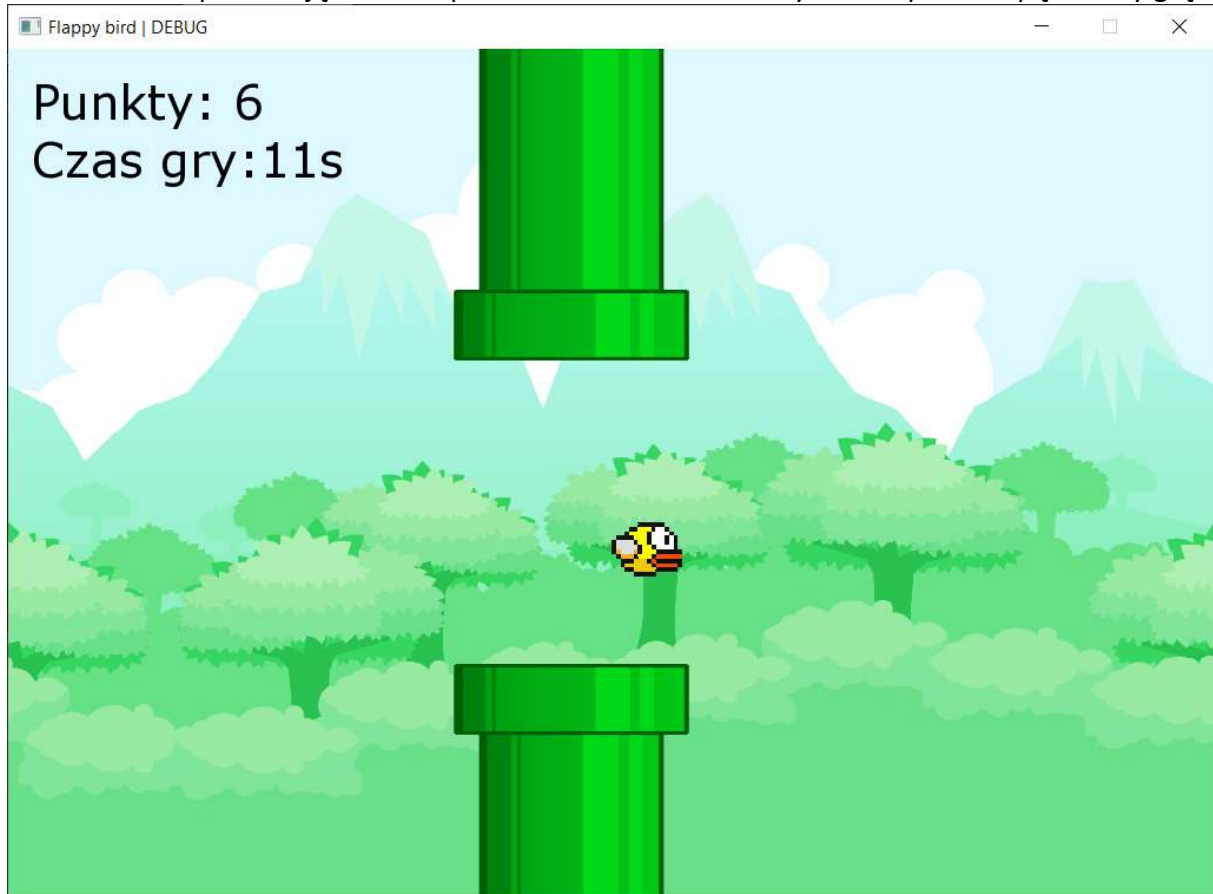
**Projekt
Programowanie w c++**

Gra „flappy bird”

Damian Bielecki 2EFZI L1
24.01.2021r

1. Temat projektu

Wybrany temat to własna wersja popularnej gry Flappy Bird, która polega na sterowaniu lecącym ptaszkiem tak aby ten zmieścił się pomiędzy pojawiającymi się przeszkodami. Każde przelecenie pomiędzy przeszkodami powodują dodanie punktu a zahaczenie lub wylecenie poza mapę kończy grę.



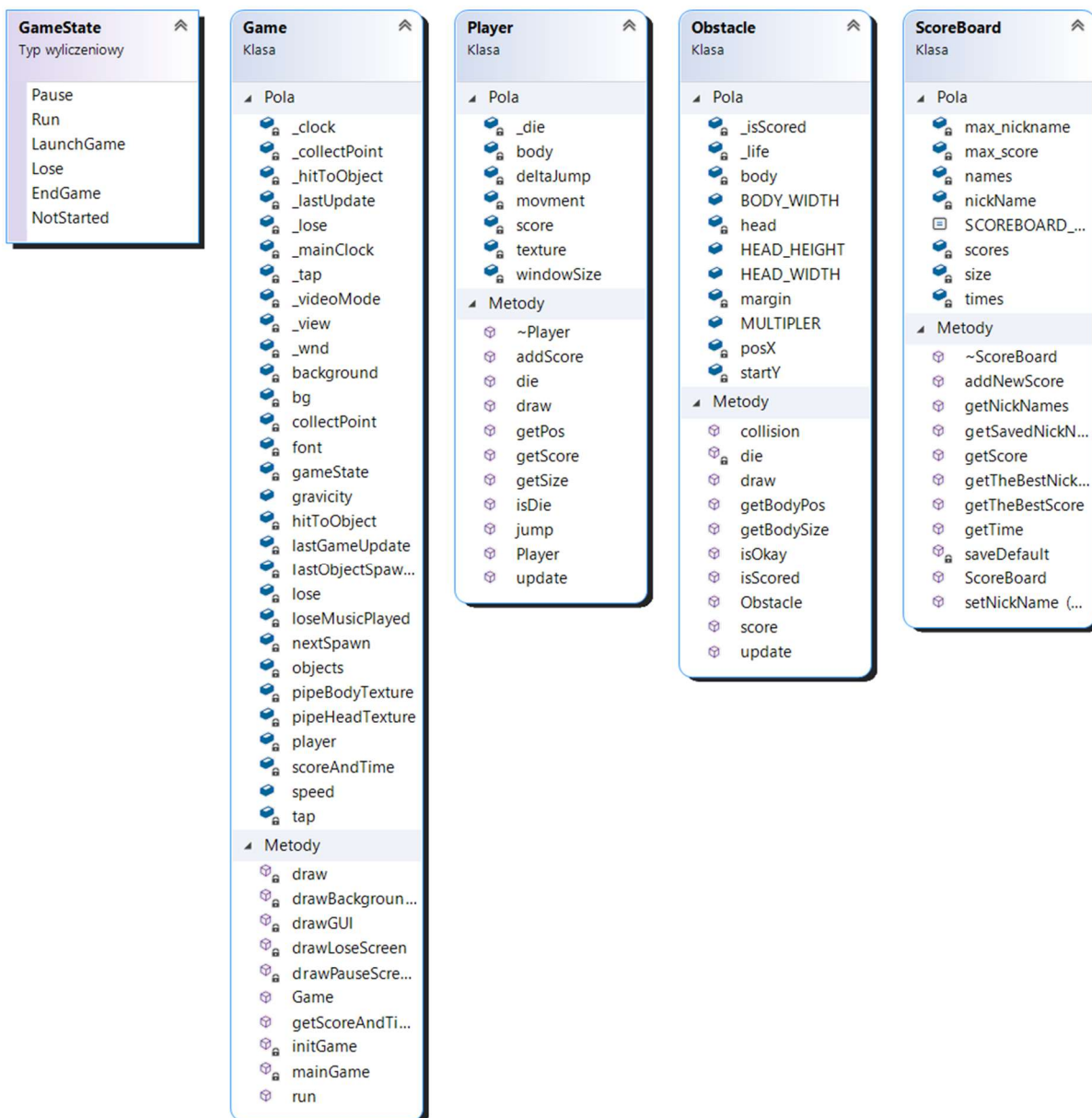
Zdjęcie 1. Zrzut obrazu z gry w trybie debug

2. Cechy projektu

Wstępnie projektując aplikację założyłem sobie że końcowy program ma spełniać następujące cechy:

- a. „nieskończona” mapa – mapa gry powinna być dynamicznie generowana tak aby możliwości zdobywania punktów nie były ograniczone przez grę.
- b. Obsługa dźwięków(dźwięk zdobycia punktu, przegranej itp.)
- c. Możliwość łatwego przeniesienia na inne systemy operacyjne
- d. Ustawienie własnego pseudonimu
- e. Tablica ostatnich wyników, najlepszy wynik
- f. Sterownie klawiaturą
- g. Generowanie nowego pliku zapisu jeżeli poprzedni został usunięty

3. Struktura



Zdjęcie 2. Struktura klas w projekcie

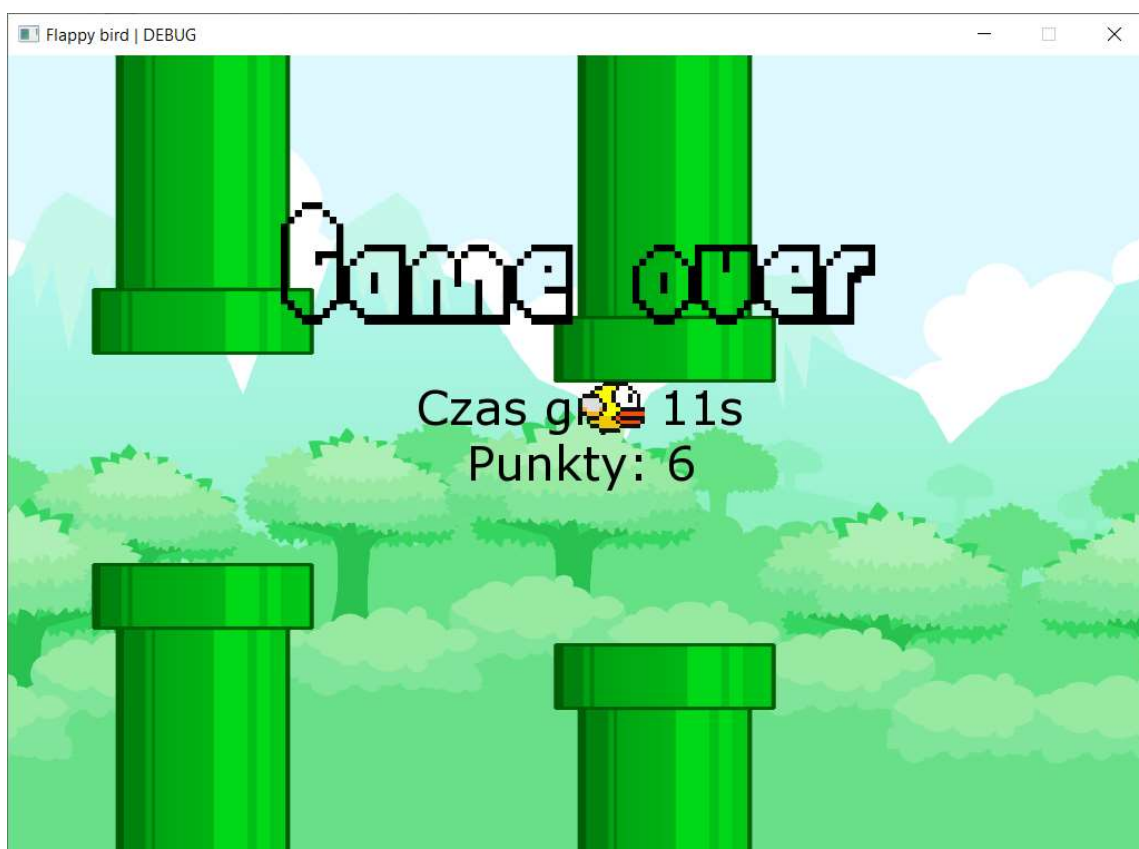
Ze względu na charakter aplikacji nie ma w niej klas dziedziczących po sobie.

4. Działanie gry

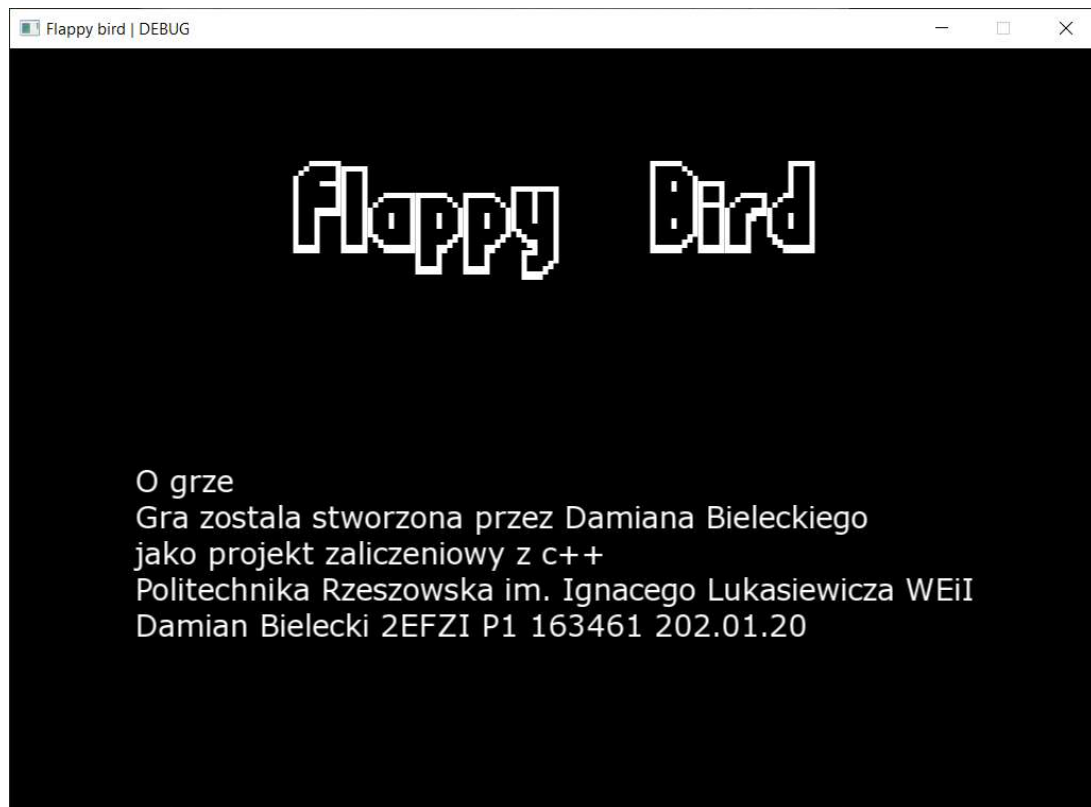
W projekcie znajdują się dwie wersje tej samej aplikacji. Wersja debug uruchamia się w małym oknie wraz z konsolą na którą trafiają wszystkie komunikaty przydatne podczas programowania. Gra w wersji „Release” uruchamia się w trybie okna bez ramki z automatycznie dobraną rozdzielczością, dodatkowo okno konsoli zostało ukryte. Ze względu na rozmiar tworzonego okna poniższe zrzuty są z trybu testowania.



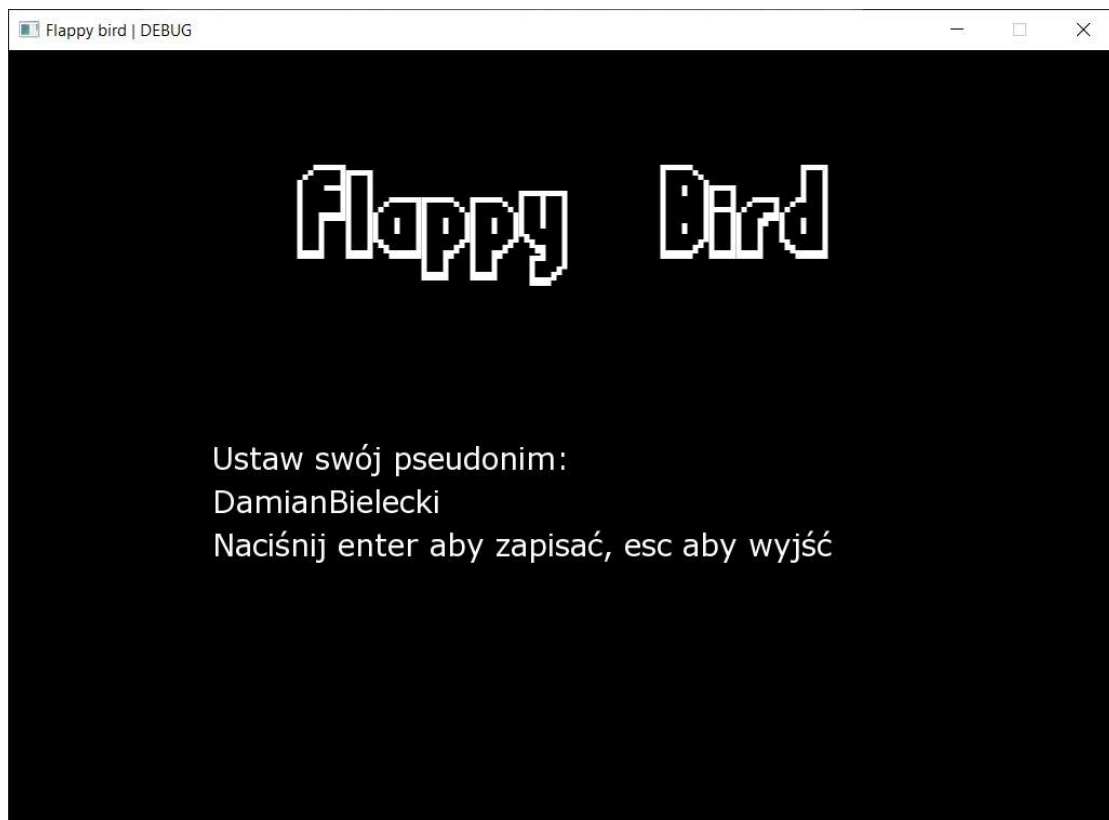
Zdjęcie 3. Menu główne



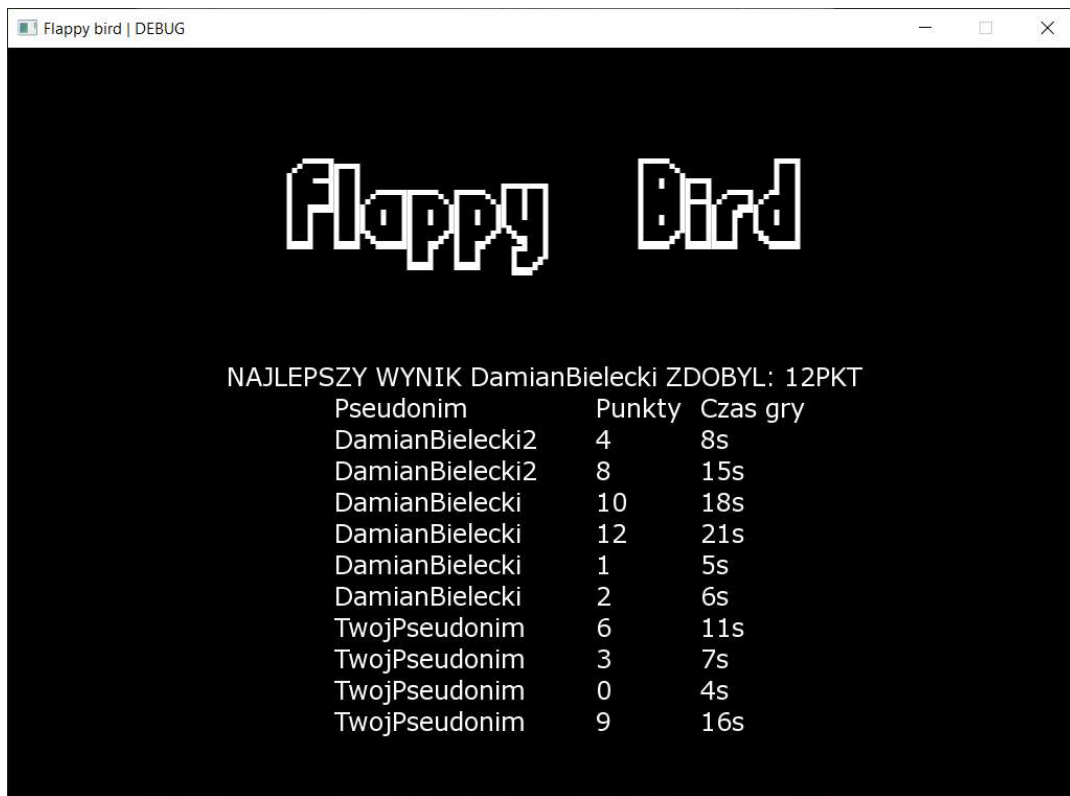
Zdjęcie 4. Działająca gra - ekran przegranej



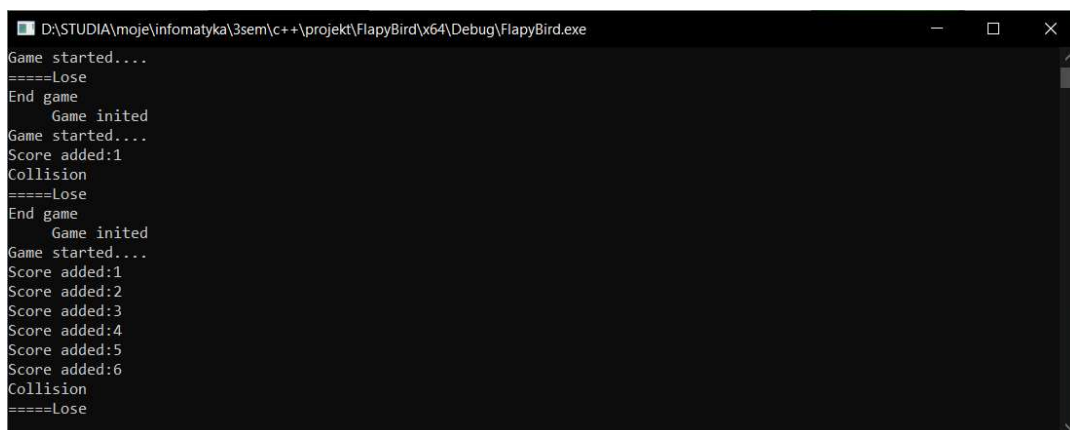
Zdjęcie 5. Okno o grze



Zdjęcie 6. Zmiana pseudonimu



Zdjęcie 7. Tabela wyników



Zdjęcie 8. Konsola z dziennikiem zdarzeń podczas gry

5. Wnioski

Program w raz z jego założonymi funkcjami został zrealizowany w całości. Struktura klas jest dosyć prosta jednak w przedstawione klasy są zdecydowanie bardziej złożone. Gra wykorzystuje kilka sztuczek aby poprawnie działać np. nieskończone tło wykonane jest przy pomocy pojedynczego obrazka który jest odpowiednio przewijany. Dzięki zastosowaniu biblioteki SFML do generowania obrazu i obsługi innych elementów okna graficznego(np. dźwięk, obsługa zdarzeń systemowych) i unikaniu funkcji charakterystycznych dla danego komputera gra może zostać przeniesiona na inny system(po odpowiednim skompilowaniu). Po wykonaniu projektu doszedłem do wniosku że taka aplikacja powinna być jak najbardziej rozbudowana pod względem ilości klas co znacznie ułatwiło by późniejsze modyfikacje. Przykładowymi zmianami może być dodanie klasy Gry po której

dziedziczyły by klasy obsługujące konkretne tryby(choć w moim programie jest tylko jeden tryb) lub dodanie klas odpowiadających za obsługę menu. Jak widać pisanie nawet tak prostej gry może pokazać jak ważnym elementem programowania jest samo odpowiednie zaprojektowanie architektury programu. Dobrze zaprojektowana aplikacja będzie wymagała więcej pracy na wstępie ale za to później będzie prostsza w modyfikacji.