

# Projekt Python - Minigolf Game

Zespół:

- Paweł Kajanek, 250097
- Adrian Kiełbowicz, 250081
- Krzysztof Małczak, 250118
- Patryk Rygiel, 250080

Prowadzący: dr inż. Marcin Zawada

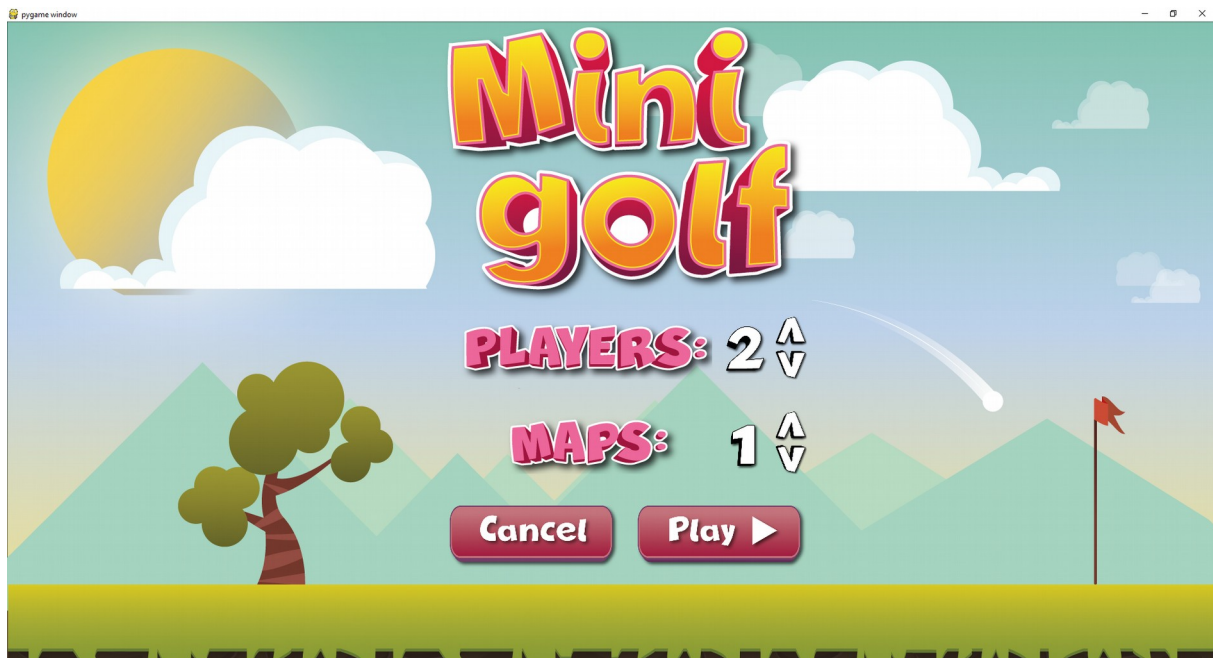
Repozytorium projektu: <https://gitlab.com/pythongbros/minigolf>

## Opis projektu

Gra Minigolf napisana przy użyciu biblioteki pygame oraz silnika fizycznego pymunk. Pierwsza z tych bibliotek odpowiada za główną pętlę gry: służy do renderowania grafiki oraz przechwytywania event'ów. Silnik pymunk natomiast odpowiada za fizyczną symulację gry, która odbywa się w tle. Na miejscu oddziałujących między sobą ciał w symulacji na ekranie renderowane są odpowiadające im ilustracje. Dzięki silnikowi pymunk możliwe jest także generowanie masek kolizji obiektu na podstawie ilustracji.



F.1. Menu główne



*F.2. Multiplayer launcher*



*F.3. Przykładowa plansza*



*F.1. Podsumowanie*

## Stan projektu według dat

### 1. 2020.04.26 po I tygodniu

#### **Ogólny opis:**

Zapoznanie się z biblioteką pygame oraz omówienie głównych oraz opcjonalnych elementów rozgrywki. Przygotowanie środowiska w condzie dla systemów linux i windows, w którym przechowywane są potrzebne dependency do odpalenia programu.

#### **Mój wkład:**

Research w kwestii tworzenia Graphical User Interface w języku python (wstępna implementacja szkieletu menu gry przy użyciu biblioteki Tkinter, jednak przez dużą ilość ograniczeń, jak brak obsługi przezroczystości, w późniejszym etapie całość GUI została zaimplementowana przy użyciu pygame).

### 2. 2020.05.03 po II tygodniu

#### **Ogólny opis:**

Zaimplementowanie wzorca zarządzania scenami i delegacji event'ów dla biblioteki pygame. Stworzenie prototypu podstawowej fizyki gry przy użyciu biblioteki pymunk i pygame. Stworzenie podstawowego wyglądu głównego menu wraz z grafikami.

**Mój wkład:**

Wstępne zaprogramowanie menu gry (teraz już przy użyciu pygame, ze względu na ograniczenia Tkinter): oprogramowanie animacji ruchomych komponentów (ruchome chmury w tle + implementacja odpowiednich metod, dzięki którym ruch chmur nie będzie zaburzany podczas zmiany rozmiaru okna), oprogramowanie animacji przycisków. Wstępna implementacja mechanizmów (dodanie odpowiednich pól oraz metod), dzięki którym możliwy będzie "resizing" okna menu gry. Utworzenie wszystkich grafik tła do menu gry oraz ruchomych komponentów występujących na tym tle. Projekt i realizacja grafik przycisków oraz napisu tytułowego.

**3. 2020.05.10 po III tygodniu****Ogólny opis**

Zaimplementowanie podstawowej rozgrywki singleplayer oraz multiplayer.

Dodanie przełączania między mapami, gdy rozgrywka zakończona (chwilowo tylko jedna mapa, więc się odświeża). Dodanie podstawowej wersji map loader'a. Dodanie grafik dla elementów mapy.

**Mój wkład:**

Wprowadzenie zmian w implementacji menu gry (od teraz menu zbudowane jest przy użyciu klasy "Scene") - implementacja klasy głównej sceny menu + usprawnienie implementacji już wcześniej implementowanych komponentów wchodzących w skład menu (np. fix bug'ów przycisków oraz ruchu chmur). Wstępna implementacja klasy typu surface (klasy powierzchni występujących na mapach gry) oraz mechanizmu tarcia w grze. Dodatkowo projekt oraz realizacja tekstur do wszystkich powierzchni w grze, rodzajów pilek oraz przeszkód (tekstury komponentów, od których piłka ma się odbijać). Podpięcie GUI do całości aplikacji (od teraz można uruchomić singleplayer za pomocą przycisku) + implementacja animacji podświetlania przycisków w momencie najechania na nie kursorem.

**4. 2020.05.17 po IV tygodniu****Ogólny opis**

Stworzenie masek kolizji dla obiektów. Refactor map loader'a oraz dodanie przykładowych map w celu testowania fizyki gry wraz z maskami. Dodanie animacji ognia dla piłki wpadającej do lawy. Dodanie nawierzchni.

### Mój wkład:

Implementacja klasy AboutScene oraz utworzenie jej designu (scena zawierająca podstawowe informacje na temat projektu). Dodatkowo utworzenie grafiki panelu pokazującego się po zakończonej rozgrywce (panel „Score”).



Wygląd sceny obsługiwanej przez klasę AboutScene

## 5. 2020.05.24 po V tygodniu

### Ogólny opis

Rozpoczęcie prac nad edytorem map. Refactor przechowywania typów obiektów jako template'y w yaml. Dodanie dodatkowych scen do menu w celu wyboru ilości graczy i map do multiplayer. Optymalizacja ilość fps'ów w grze poprzez renderowanie obiektów nie ruchomych jako statyczne tło.

### Mój wkład:

Implementacja klasy MultiSetupScene oraz utworzenie jej designu (scena odpowiedzialna za ustawianie preferencji gry multiplayer - liczba graczy, plansze)(pozycjonowanie elementów na scenie, oprogramowanie ich funkcjonalności np.update\_images() itd.). Implementacja „handler'ów” dla przycisków w tej klasie - przyciski umożliwiające ustawienie ilości graczy, ilości map, dodanie możliwości uruchomienia rozgrywki z danymi ustawieniami (+ preview ilości graczy oraz ilości map wraz z grafikami). Podłączenie nowej sceny do całości aplikacji (teraz po wciśnięciu "multiplayer" pytamy gracza o

odpowiednie preferencje przed uruchomieniem rozgrywki) + możliwość powrotu do menu (patrz *F.2. Multiplayer launcher*).

## **6. 2020.05.31 po VI tygodniu**

### **Ogólny opis**

Dokończenie edytora. Dodanie paska z wynikami dla gry. Dopracowywanie elementów rozgrywki i fizyki gry. Łatanie bugów. Design map.

### **Mój wkład:**

Refactor klasy ScoreScene – reimplementacja metod odpowiedzialnych za wyświetlanie końcowych wyników rozgrywki. Utworzenie grafik do wyświetlania wyników końcowych. Fix bug'u dotyczącego zmiany sceny po wcześniejszej zmianie rozmiaru okna. Naprawienie bug'u ruchu chmur w scenie ScoreScene. Tworzenie map do gry przy użyciu edytora. Testowanie rozgrywki oraz całości aplikacji w celu lokalizacji bug'ów.

## **7. 2020.06.07 (koniec projektu)**

### **Udało się zrobić**

- Tryb singleplayer i multiplayer,
- Fizyka gry przy użyciu silnika fizycznego pymunk,
- Generowanie masek kolizji z ilustracji,
- Przeszkody i powierzchnie z template'ów – w łatwy sposób możliwe jest dodanie nowych,
- Edytor map w celu ułatwienia ich tworzenia.

### **Nie udało się zrobić – potencjalne plany na rozwój**

- Tryb multiplayer online – podstawowy serwer i klient zostały utworzone, lecz sam tryb nie został zaimplementowany,
- Skalowanie mapy gry – skalowanie zostało dodane do menu głównego, lecz nie dla samej gry,
- Bot multiplayer – był jako opcjonalna funkcjonalność do dodania, jeżeli starczyłoby czasu.