

**Implementacja systemu gier  
do rehabilitacji osób  
z zaburzeniami błędnika  
za pomocą posturografu**

(Implementation of the game system  
for the rehabilitation of people  
with labyrinthitis)

Izabela Strumecka

Dominik Hawryluk

Praca inżynierska

**Promotor:** dr Łukasz Piwowar

Uniwersytet Wrocławski  
Wydział Matematyki i Informatyki  
Instytut Informatyki

03.09.2020



## Streszczenie

Projekt opisany w niniejszej pracy powstał dzięki współpracy z prof. dr hab. Małgorzatą Mraz z Akademii Wychowania Fizycznego we Wrocławiu oraz dr hab. Leszkiem Czerwoszem z Instytutu Medycyny Doświadczalnej i Klinicznej Polskiej Akademii Nauk. Celem projektu było uatrakcyjnienie systemu do rehabilitacji pacjentów zmagających się z problemami zaburzenia równowagi. Zastane oprogramowanie pozbawione było dźwięku oraz używało bardzo uproszczonej grafiki. Celem ćwiczenia było umieszczenie wskaźnika w ustalonym obszarze pojawiającym się w różnych częściach ekranu. Tak realizowane zadanie powtarzane przez około 30 minut było dla pacjenta nużące i nie zachęcało do aktywnego udziału w treningu. Stworzony przez nas system gier, składający się z pięciu mini gier, poprzez swą różnorodność, bogate efekty wizualne, słuchowe oraz nietrywialne cele, stara się rozwiązać ten problem.

Sterowanie w grach, polega na balansowaniu pacjenta na platformie posturograficznej. Wraz ze zmianą punktu rzutowanego środka ciężkości przemieszcza się wskaźnik na ekranie. Każda gra ma indywidualnie dostosowywane ustawienia, pozwalające na skonstruowanie wydajnego i efektywnego treningu, w zależności od potrzeb i możliwości pacjenta. System posiada także bazę osób biorących udział w ćwiczeniach, w której zapamiętywane są indywidualne wyniki oraz parametry dostosowane do schorzenia pacjenta. Ważnym elementem jest także punktacja, umożliwiająca śledzenie postępów oraz zachęcająca do rywalizacji. System wykorzystuje jeden z najpopularniejszych silników do tworzenia gier - Unity.

Od roku nasz projekt jest wdrożony praktycznie. U pacjentów widoczna jest szybsza poprawa stanu zdrowia związana prawdopodobnie z większą motywacją oraz zaangażowaniem podczas treningów. Rehabilitacja stała się dla nich przyjemnością i kojarzy się z mile spędzonym czasem.

---



# Spis treści

<b>1. Wprowadzenie</b>	<b>7</b>
<b>2. Posturografia</b>	<b>9</b>
<b>3. Treningi posturograficzne</b>	<b>13</b>
<b>4. Nowy system gier</b>	<b>15</b>
4.1. Cechy wspólne gier . . . . .	15
4.2. Puzzle . . . . .	17
4.3. Labirynt . . . . .	18
4.4. Kaczki . . . . .	19
4.5. Samoloty . . . . .	20
4.6. Arkanoid . . . . .	21
<b>5. Menu gier</b>	<b>23</b>
<b>6. Klient - Serwer</b>	<b>27</b>
<b>7. Silnik Unity oraz narzędzia wykorzystane w projekcie</b>	<b>29</b>
<b>8. Baza pacjentów i ustawień</b>	<b>37</b>
<b>9. Podsumowanie</b>	<b>39</b>
<b>Bibliografia</b>	<b>41</b>



# Rozdział 1.

## Wprowadzenie

Zaburzenia układu równowagi są związane z dysfunkcją co najmniej jednego układu biorącego udział w utrzymaniu równowagi, po stronie receptorów - w układzie czucia głębokiego, błędniaka, a także narządu wzroku, po stronie efektorów - tj w układzie nerwowo mięśniowym a także kostno-stawowym. Wielką rolę w koordynacji mięśniowo-ruchowej, w sterowaniu odruchami zapewniającymi stabilność, czyli przywracania równowagi ma mózdzek. Ubytki i zaburzenia mózdzku wpływają znacząco na niestabilność ciała. Zaburzenia równowagi mogą być konsekwencją wielu chorób, takich jak udar mózgu, choroba Addisona-Biermera, zapalenie błędniaka, choroba Parkinsona, w stwardnieniu rozsianym (SM), ataksjach rdzeniowo-mózdkowych. Pogorszenie działania układu równowagi obserwuje się w również procesie starzenia.

Problemy z kontrolą równowagi niosą ze sobą ryzyko upadku, który może spowodować złamanie, zwichnięcie a także "prowadzić do pogorszenia jakości życia" [1]. Niedogodnością są również częste zawroty głowy. Wczesne wykrycie schorzeń oraz odpowiednia rehabilitacja są w stanie zapobiec lub odwrócić negatywne skutki, może być to szczególnie istotne dla osób starszych. "Co drugi polski senior powyżej 65 roku życia ulega przynajmniej raz w roku upadkowi, który dla 41% osób z omawianej grupy związany jest z urazem o różnym stopniu nasilenia." [2]

Na terenie Akademii Wychowania Fizycznego we Wrocławiu w zakładzie Fizjoterapii w neurologii i pediatrii pod kierunkiem prof. dr hab. Małgorzaty Mraz, od lat prowadzone są rehabilitacje osób dotkniętych problemem zaburzeń układu równowagi.

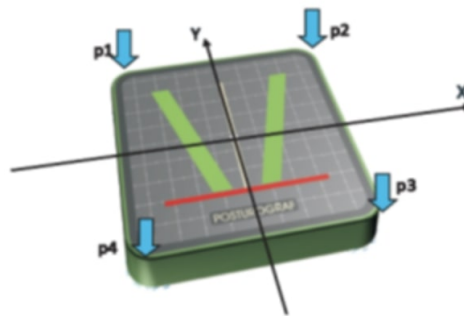




## Rozdział 2.

# Posturografia

Posturografia jest metodą badawczą, pozwalającą obiektywnie ocenić równowagę ciała. Do badań wykorzystywana jest platforma sił z czujnikami tensometrycznymi w narożnikach, które rejestrują naciski na nogi platformy. Na podstawie tych czterech sił możliwe jest obliczenie położenia w osi X i osi Y punktu zaczepienia siły ciężkości i jednocześnie punkt przyłożenia siły nacisku na stopy (center of foot pressure - COP).



Źródło: praca Czerwosz i inni 2012 – za zgodą autora [3]



W trakcie pomiaru punkt COP przemieszcza się zgodnie z:

1. ruchami rzutu pionowego punktu środka ciężkości osoby badanej (center of gravity - COG) na poziomą płaszczyznę podparcia,
2. wpływem sił bezwładności powstających przy zmianie prędkości ruchu ciała, czyli przyspieszeniem, które również stale zmienia się co do kierunku i wartości. [4]

Ruchy COP wynikają z chwilowej utraty równowagi i jej przywrócenia przez uruchamianie mięśni posturalnych. Przywrócenie równowagi może być nieświadome (automatyczne) lub świadome. Położenie chwilowe punktu COP jest w przybliżeniu zgodne z rzutem pionowym COG przy bardzo łagodnych ruchach ciała, tj. gdy przyspieszenia i związane z nimi siły bezwładności są małe w porównaniu z ciężarem ciała. Ruchy z większą intensywnością generują większe przyspieszenia i w konse-

kwencji siły bezwładności, mające wpływ na położenie COP.

Posturografia mierzy wielkość tzw. wychwiał osoby stojącej w pozycji wyprostowanej na platformie posturograficznej. Pomiar może odbywać się z oczami otwartymi lub zamkniętymi. Pomiary odbywają się przez około pół minuty. Po zakończeniu każdego pomiaru trajektoria ruchów COP jest oceniana parametrycznie. Obliczone parametry krzywej mają wartość diagnostyczną.

Jedną z metod oceny jest pomiar posturograficzny ze wzrokowo ruchowym sprzężeniem zwrotnym. Sprzężenie polega na obserwacji przez osobę badaną (pacjenta) ruchomego znacznika na ekranie komputera. Znacznik ten reprezentuje chwilowe położenie COP. [5]



## Rozdział 3.

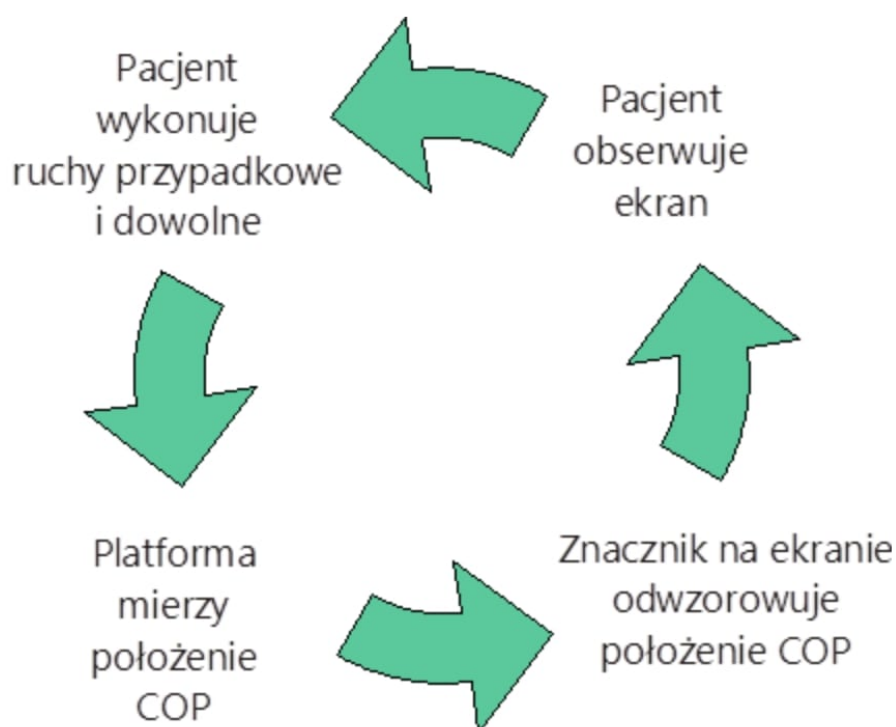
# Treningi posturograficzne

Ta sama zasada sprzężenia zwrotnego wykorzystywana jest w treningach posturograficznych, gdy osoba badana oprócz utrzymania równowagi w pozycji pionowej, ma za zadanie wychylać się od pionu w sposób zamierzony, zgodnie z obrazem na ekranie, na którym pojawiają się figury geometryczne (kwadraty lub koła) oddalone od środka (punktu równowagi chwiejnej) o kilka centymetrów (w skali płaszczyzny podparcia stóp).

Zadaniem pacjenta w trakcie treningu jest osiągnięcie znacznikiem reprezentującym punkt COP szeregu punktów na ekranie w obrębie koła lub kwadratu oraz utrzymanie pozycji przez jakiś czas (kilka-kilkanaście sekund). “Zadanie to przypomina sterowanie kursorem myszki na ekranie, z tym że zaangażowane są całkowicie inne mięśnie i dodatkowo kontrolowana równowaga posturalna.” [5] Trening jest z reguły dłuższy niż pomiar diagnostyczny i trwa od kilku do kilkunastu minut. Utrzymanie znacznika wymaga napięcia mięśni posturalnych, przede wszystkim mięśni stawów kończyn dolnych, brzucha. Jednocześnie następuje stymulacja obwodowego i centralnego układu nerwowego biorącego udział w procesie odczuwania pozycji ciała, jego ruchu i koordynacji. Po wychyleniu następuje powrót do położenia COP w przybliżeniu do punktu startowego. Treningi posturograficzne stanowią znakomite rozszerzenie procesu rehabilitacji układu równowagi. W trakcie ćwiczeń ma nastąpić poprawa koordynacji wzrokowo-ruchowej oraz opanowanie symetrycznego obciążenia obu nóg.

W zależności od potrzeb i możliwości pacjenta można zmienić warunki przeprowadzanego treningu, utrudniając utrzymanie równowagi osoby rehabilitowanej poprzez np. położenie gąbki na urządzeniu, czy zmieniając jego pozycję, np. ćwiczenie może być przeprowadzone w pozycji siedzącej.

Oprogramowanie wykorzystywane do tej pory w rehabilitacji jest w rzeczywistości grą, w której zadaniem pacjenta jest przemieszczanie znacznika na ekranie za pomocą balansu ciała. Gra jest bardzo prosta w założeniu, uboga w efekty wizualne i dźwiękowe, przez co sesje terapii są monotonne i nieangażujące. Oryginalne oprogramowanie zostało wykonane przez firmę Pro-Med. Zostało ono następnie rozszerzone



Schemat wzrokowo-ruchowego sprzężenia zwrotnego. Źródło: adaptacja ilustracji z Czerwosz i inni 2009, za zgodą autorów [6]

w celu równoległej pracy wielu zespołów badawczych i dla dokonywania nowych analiz sygnału, z zachowaniem zgodności formatu danych. Nowa wersja oprogramowania wysyła za pośrednictwem lokalnej sieci komputerowej sygnał o chwilowym położeniu  $xy(t)$  punktu COP. Wysyłanie danych odbywa się tylko w trakcie pomiaru posturograficznego lub treningu.

**Celem naszej pracy jest stworzenie systemu komputerowego, składającego się z pięciu bogatych graficznie, muzycznie, różnorodnych pod względem zawartości gier. Oczekiwany wynik projektu jest polepszenie warunków leczenia, poprzez dostarczenie pacjentowi zupełnie nowych wrażeń oraz wyzwań, a przez to, jak sądzimy, możliwe będzie osiągnięcie lepszych efektów leczenia.**

## Rozdział 4.

# Nowy system gier

Nowy system gier na platformę posturograficzną składa się z pięciu mini-gier: puzzle, kaczki, samoloty, labirynt i arkanoid. Celem projektu jest urozmaicenie czasu pacjenta podczas rehabilitacji oraz zachęcenie do większej uwagi i skupienia w trakcie ćwiczeń. Oprogramowanie posiada animacje, kolorową grafikę 3D, muzykę oraz dźwięk co zdecydowanie odróżnia je od zastanego rozwiązania.

Dodatkowym aspektem jest również element grywalizacji. System punktacji umożliwia pacjentowi śledzenie swoich postępów oraz porównywanie się z innymi. Po ukończeniu każdej z gier wyświetlany jest ranking pięciu najlepszych graczy oraz wynik aktualnie grającego. Ten element rywalizacji motywuje do ciągłej poprawy.

### 4.1. Cechy wspólne gier

Częścią wspólną gier jest system ustawień. W większości gier występuje tzw. "szachownica". Służy ona do wyboru części ekranu, na której ma się skupiać ćwiczenie pacjenta. Umożliwia to dostosowanie rehabilitacji do typu schorzenia.

Do wyboru mamy dziewięć pól dzielących ekran, które możemy zaznaczyć, bądź nie w zależności od potrzeb. Środek ciężkości człowieka oscyluje w granicach stabilności, po przekroczeniu których może dojść do upadku.

Granice stabilności wyznaczane przez obwódnie stóp dzielimy na przednio - tylną i w płaszczyźnie bocznej. Granica stabilności w płaszczyźnie przednio - tylnej jest większa od granicy w płaszczyźnie bocznej. W celach diagnostycznych i terapeutycznych przenosimy te granice na układ współrzędnych dzieląc go na cztery ćwiartki. Osoby cierpiące na schorzenia neurologiczne np. choroba Parkinsona, zawroty głowy w wyniku uszkodzenia narządu przedsionkowego, udar mózgu, stwardnienie rozsiane, są narażone na zwiększone ryzyko upadku z powodu gorszej kontroli wychyleń środka ciężkości. Wykorzystując platformę posturograficzną można dokładnie ocenić, w której ćwiartce układu współrzędnych występuje problem z kontrolą wychyleń środka ciężkości i odpowiednio dostosować do tego ćwiczenia.

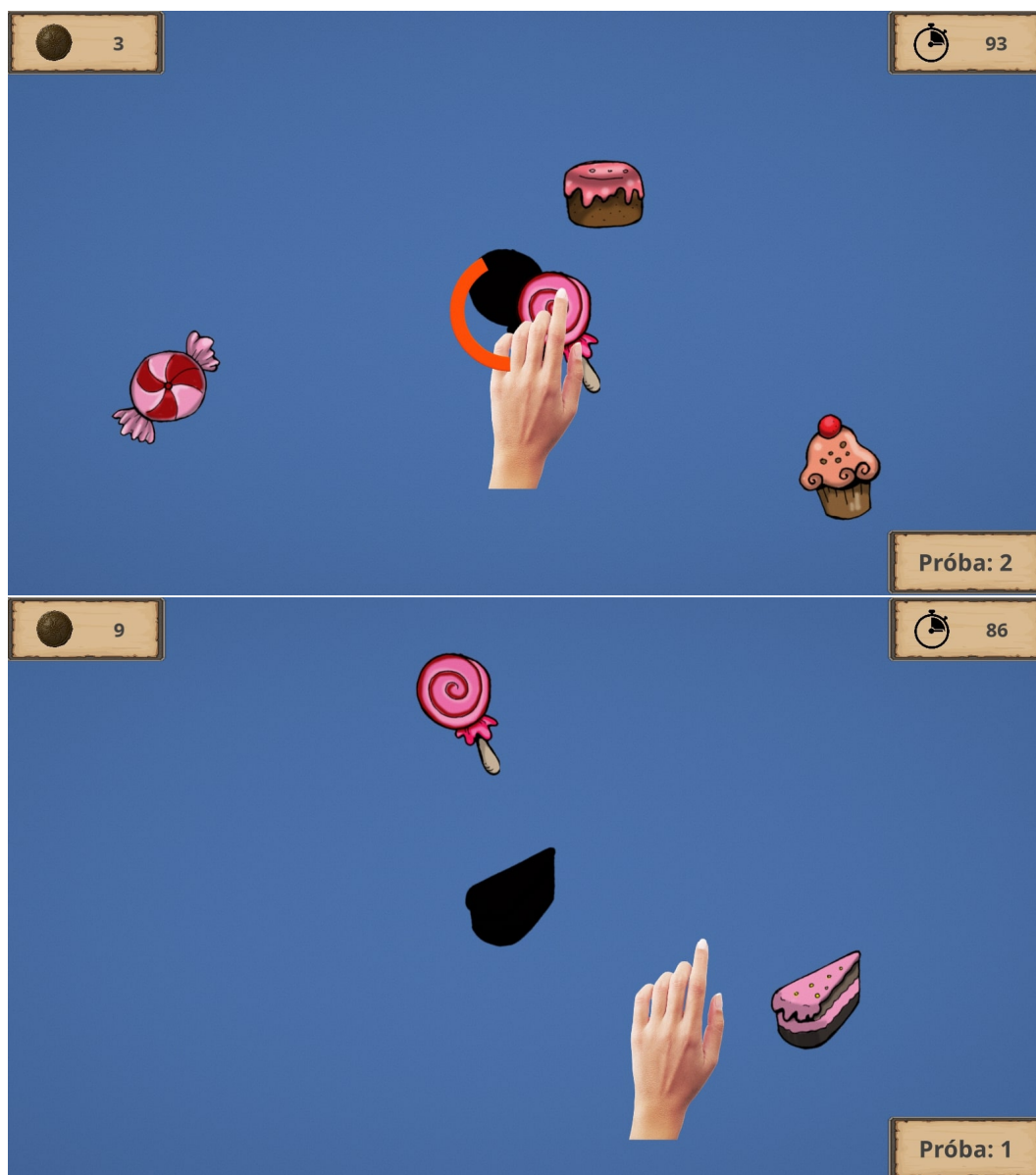
Wprowadzona w grach opcja pozwala precyzyjnie ukierunkować leczenie defi-

cytów pacjenta. Na przykład: pacjent po udarze mózgu ma zaburzoną kontrolę nad środkiem ciężkości w pierwszej ćwiartce układu współrzędnych (problemy z wychyleniem się do przodu i w lewo). Dzięki "szachownicy" można dokładnie określić tę ćwiartkę, uruchomić grę, której akcja skupia się na wybranej części i tym samym poprawić kontrolę nad środkiem ciężkości zmniejszając ryzyko upadku (krótko mówiąc, dostosować rodzaj ćwiczeń do dolegliwości pacjenta).

Każda z gier umożliwia wybór prędkości oraz czasu jej trwania. Te kustomizacje pozwalają na dostosowanie poziomu trudności do stanu pacjenta.



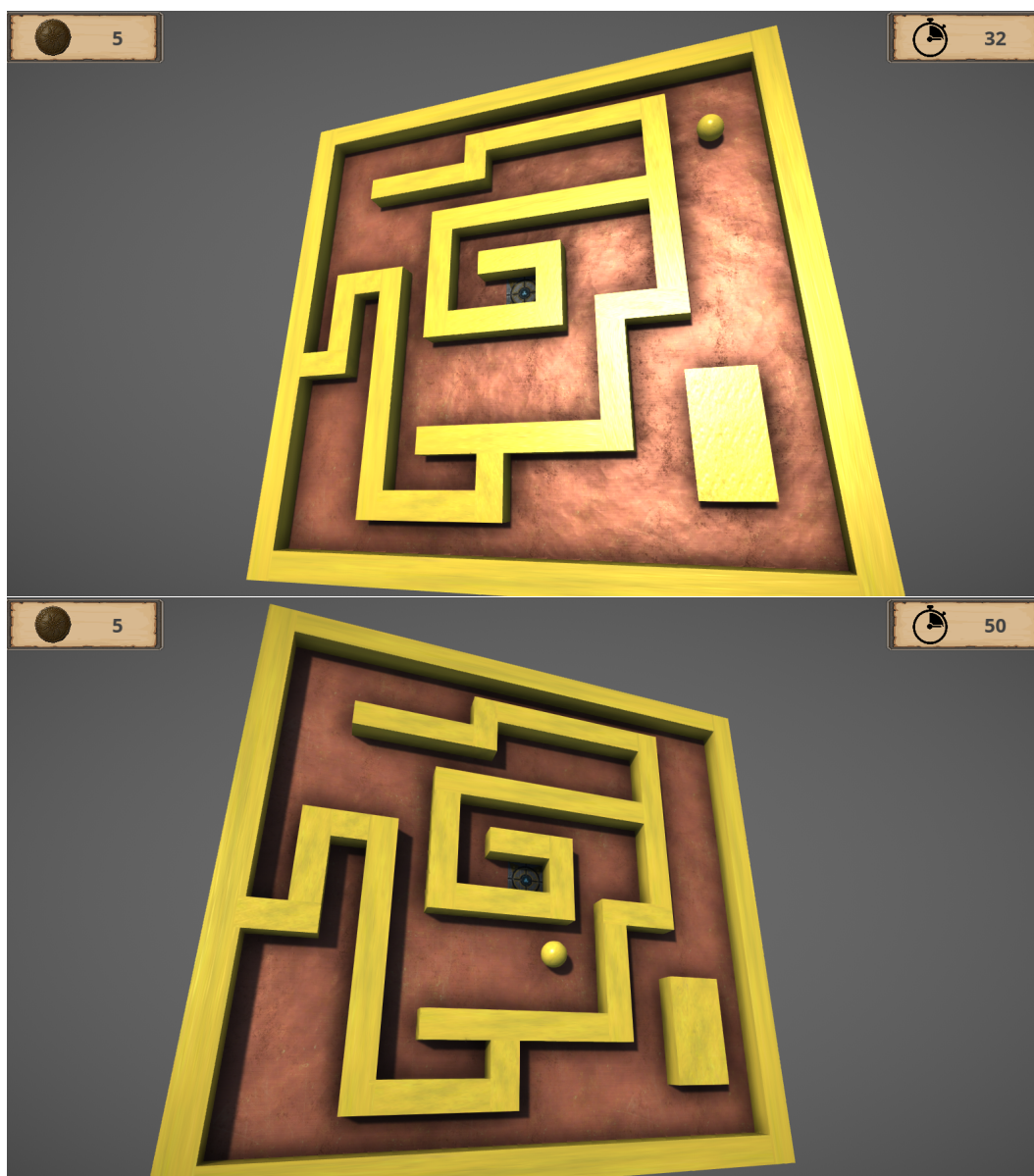
## 4.2. Puzzle



Gra polega na przemieszczeniu puzzla do odpowiadającego mu obrysu znajdującego się na środku ekranu. Po dobrym dopasowaniu losowany jest kolejny kształt. Całą procedurę powtarzamy do momentu pozbycia się wszystkich puzzli, bądź do zakończenia czasu gry.

Sterowanie polega na balansowaniu ciałem na platformie posturograficznej, któremu odpowiada poruszany na ekranie kursor. Aby podnieść element należy odpowiednią ilość czasu utrzymać wskaźnik w jego obrębie. Ta sama zasada obowiązuje przy jego odkładaniu. Czas wymagany do aktywacji podniesienia lub odłożenia puzzla jest jedną z opcji gry. Powierzchnię aktywności ćwiczenia pacjenta możemy ograniczyć wykorzystując "szachownicę". Spowoduje to pojawianie się puzzli tylko w wyznaczonych częściach ekranu. Punkty przydzielane są na podstawie ilości prób dopasowania elementu oraz sumarycznego czasu.

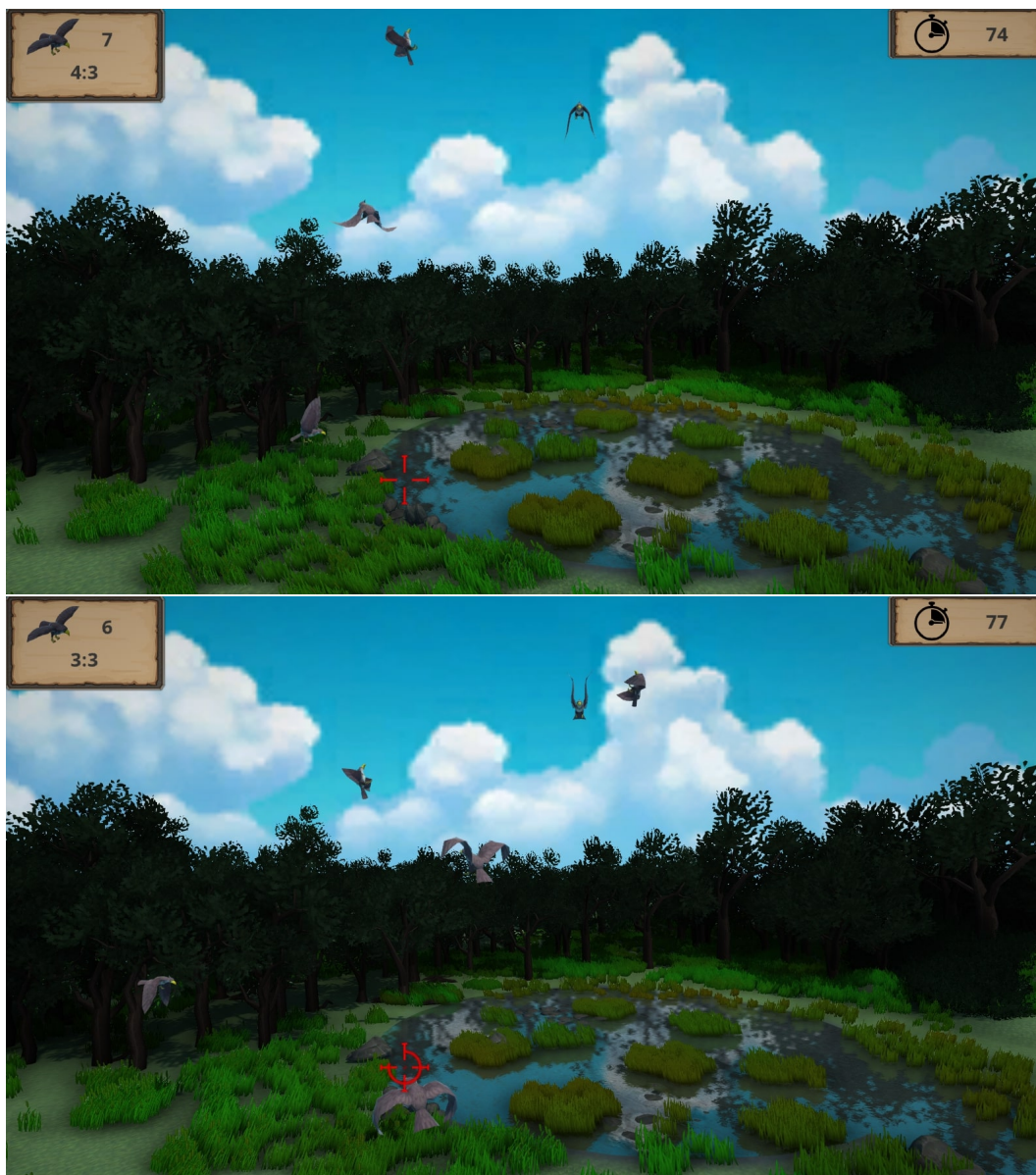
### 4.3. Labirynt



Celem gry jest przejście labiryntu. Wchylenie planszy w osiach X i Z odpowiada bezpośrednio balansowi pacjenta na platformie posturograficznej. Odpowiedni kąt nachylenia planszy, poprzez grawitację powoduje przemieszczanie się kulki. Celem jest dotarcie do mety, gra składa się z trzech poziomów.

Indywidualnym ustawieniem skorelowanym ze stanem chorego jest możliwość zmiany masy kulki. Efektem jest zmiana prędkości poruszania się kulki, co wyraźnie wpływa na trudność gry. Finalną ewaluacją postępów w grze jest suma sekund pozostałych po ukończeniu każdego z poziomów.

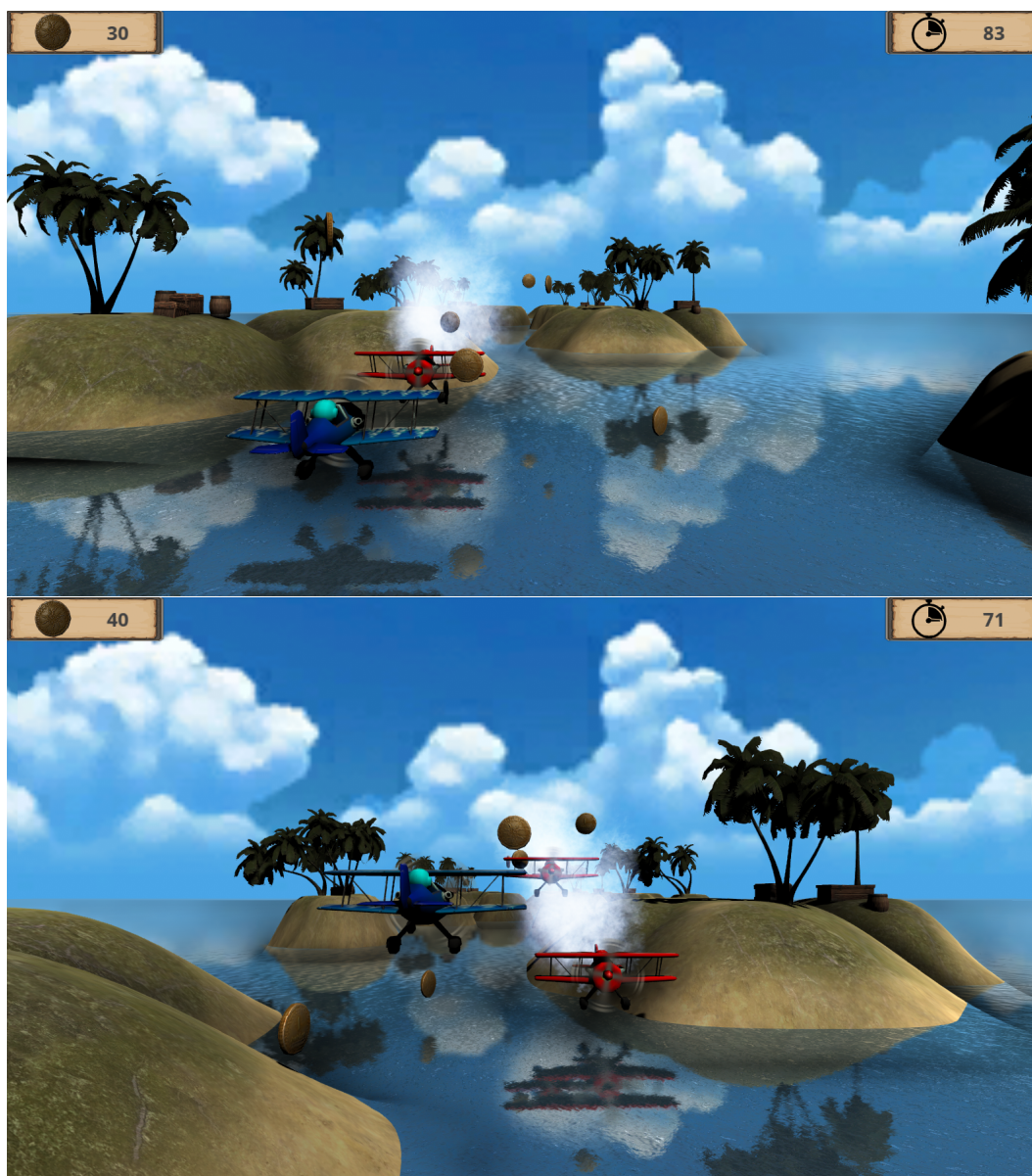
#### 4.4. Kaczki



Celem kolejnej gry jest zestrzelenie jak największej ilości przelatujących kaczek. Gracz, balansując ciałem przesuwa celownik po ekranie. Strzał następuje po utrzymaniu wskaźnika w obrębie celu przez ustaloną liczbę sekund.

Domyślnie kaczki nadlatują z każdej strony i mogą lecieć w stronę dowolnego punktu. Dzięki "szachownicy" możemy ustalić kierunek i miejsce nadlatujących kaczek, tak aby dostosować ćwiczenie do wymaganej rehabilitacji pacjenta. Dodatkowo możemy ustawić prędkość kaczek oraz celownika w zależności od potrzeb.

## 4.5. Samoloty



W kolejnej grze - Samoloty, pacjent przemieszcza się na ekranie awatarem samolotu wykorzystując balans ciała i jednocześnie ze stałą prędkością porusza się do przodu. Dodatkowym celem jest zbieranie monet, unikanie terenu oraz wrogich samolotów. Sceneria gry to kilka rodzajów generatywnie tworzonych grup wysp, które pojawiają się losowo w oddali przed graczem.

W ustawieniach możemy zmienić prędkość samolotu gracza, oraz, wykorzystując "szachownice" ustawić możliwe obszary pojawiania się monet i przeciwników.

## 4.6. Arkanoid



Celem ostatniej gry - Arkanoid - jest taki balans ciałem, aby za pomocą piłki odbijanej ruchomą paletką rozbić jak największą ilość punktowanych kafelek. Poza standardowymi, występują dwa rodzaje specjalnych klocków – zabicie pierwszego rodzaju wymaga kilku trafień, natomiast zabicie drugiego rodzaju sprawia, że otrzymujemy dodatkową piłkę. W przeciwieństwie do oryginalnego Arkanoida gra toczy się w zamkniętym obszarze. Zmiana ta powstała wskutek testów oryginalnych zasad gry, z udziałem pacjentów. Poza ułatwieniem wprowadziliśmy także pewne utrudnienie polegające na tym, że gdy piłka dotknie podłogi to zmienia kolor i przestaje być aktywna, aż do momentu ponownego odbicia przez gracza. Nie dotyczy to piłek dodatkowych, które po dotknięciu dolnej krawędzi znikają.

W ustawieniach oprócz czasu gry oraz prędkości zarówno piłek jak i paletki mo-

żemy ustawić wielkość obszaru, w którym toczy się rozgrywka. Po wybraniu liczby wierszy i kolumn, gra losuje, w których miejscach pojawią się specjalne pola.

## Rozdział 5.

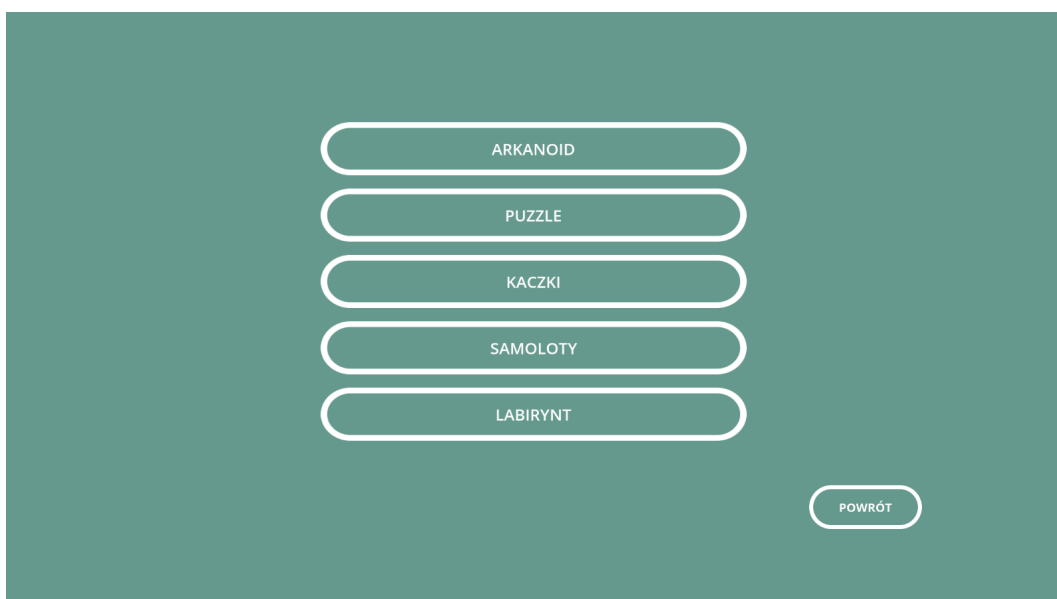
# Menu gier

Pierwszą rzeczą po uruchomieniu aplikacji jest menu główne. W założeniu jest ono obsługiwane przez osobę prowadzącą sesje rehabilitacyjną. Składa się z czterech ekranów: startowego, wyboru gier, ustawień ogólnych i szczegółowych:

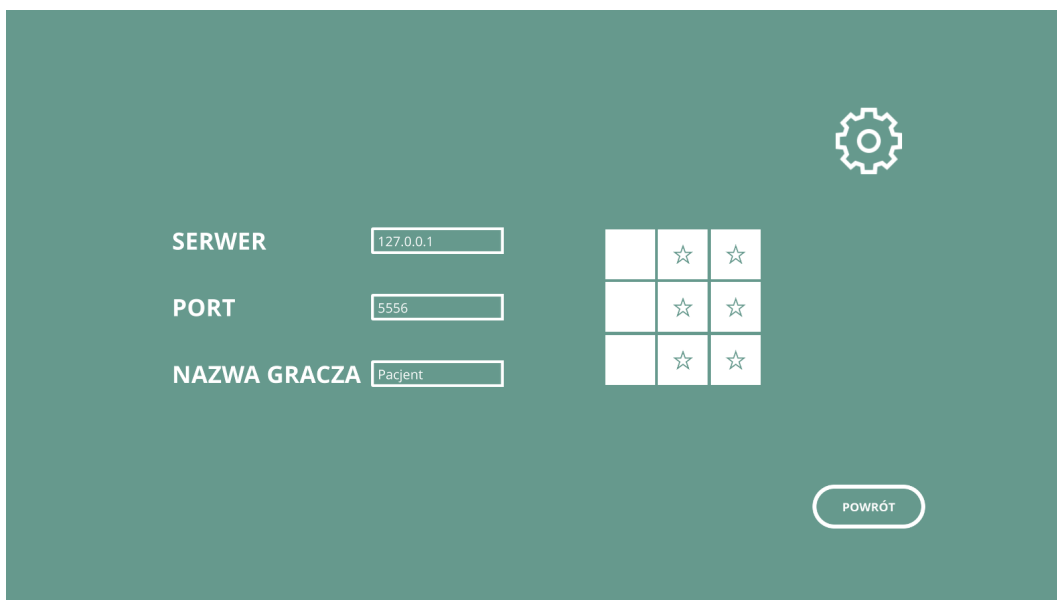
- Ekran startowy – służy jako łącznik pomiędzy ustawieniami, a listą gier.



- Ekran wyboru gier - umożliwia powrót lub wybranie konkretnej gry.

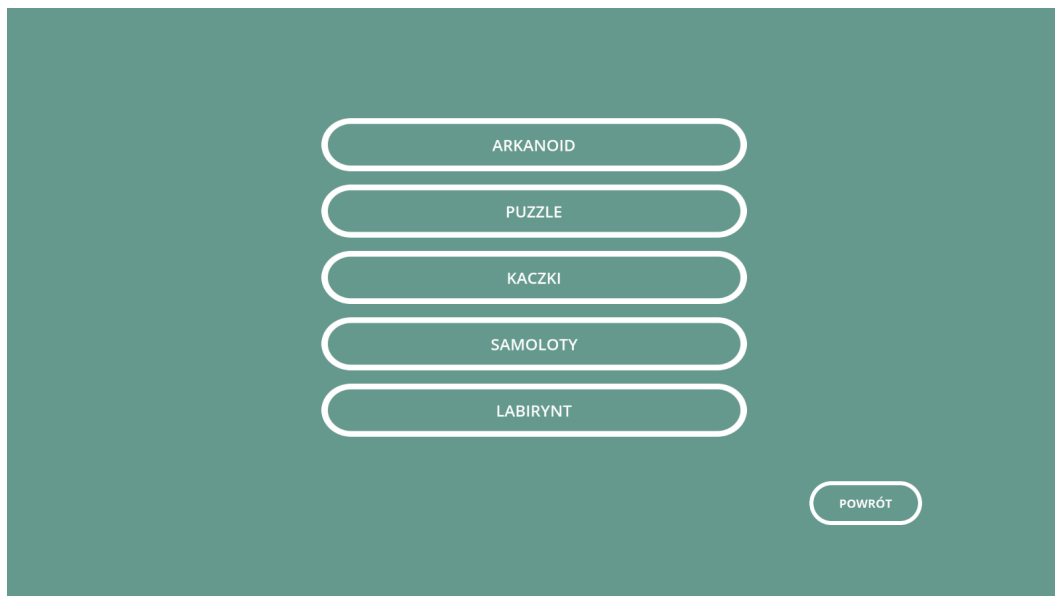


- Ekran ustawień ogólnych – umożliwia wprowadzenie pseudonimu identyfikującego danego pacjenta, aby odczytać lub zapisać indywidualne dla niego ustawienia. Wybranie odpowiednich pól szachownicy dostosowuje aktywne obszary poszczególnych gier do stanu pacjenta. Ostatnią z opcji jest możliwość wprowadzenia adresu i portu, po którym aplikacja łączy się z częścią serwerową.





- Ekran ustawień szczegółowych – umożliwia ustawienie indywidualnych opcji w poszczególnych grach dostosowanych do stanu zdrowia i możliwości motorycznych pacjenta. Poprzez szereg eksperymentów, zostały dobrane wartości domyślne, które powinny być optymalne dla większości graczy.





## Rozdział 6.

# Klient - Serwer

Niezbędną częścią naszego projektu było stworzenie klienta łączącego się z serwerem obsługującym platformę posturograficzną. Do połączenia wymagany jest adres IP oraz numer portu. Te wartości przypisujemy w ustawieniach. Klient łączy się przy pomocy gniazda z wykorzystaniem biblioteki NetMQ. Połączenie nawiązywane jest po włączeniu gry, a jego działanie w osobnym wątku, powoduje, że główny wątek gry nie jest w żaden sposób blokowany. Dane przekazywane są za pomocą ciągu znaków, z którego możemy wydzielić dwie interesujące nas części – aktualne współrzędne X i Y, odpowiadające położeniu punktu COP. Dla zapewnienia maksymalnej możliwej płynności rozrywki, dane pobierane są tak szybko jak tylko pojawią się po stronie serwera (klient aktywnie czeka na przesyłane dane, które serwer wysyła z częstotliwością 32 Hz). W przypadku utraty synchronizacji z serwerem, gra korzysta z ostatnich odczytanych wartości, aż do momentu przywrócenia połączenia, co podnosi stabilność działania gier. W pierwszej wersji systemu, dla każdej gry z osobna nawiązaliśmy indywidualne połączenie z serwerem. Seria testów praktycznych wykazała, że jedno połączenie z serwerem jest stabilniejszym rozwiązaniem.



## Rozdział 7.

# Silnik Unity oraz narzędzia wykorzystane w projekcie

W projekcie wykorzystany został jeden z najpopularniejszych w świecie twórców gier silnik - Unity. [7]

Technologia ta dostarcza wiele gotowych komponentów takich jak silnik graficzny, fizyczny, muzyczny, zunifikowany model elementów gry, posiadających wspólny komponent “transform”, oraz umożliwiających dodawanie dowolnych komponentów sterujących lub agregujących dane. Głównym językiem używanym do programowania w silniku jest język C#.

Za symulowanie fizyki w grze zarówno 2D jak i 3D odpowiada silnik fizyczny. Trudność jego użycia uproszczono do minimum, aby obiekt podlegał prawom fizyki wystarczy dodać do niego komponent Rigidbody oraz ustawić parametry takie jak masa, opór, opór obrotowy. Dodatkowo możemy ograniczyć ruch obiektu, blokując jego pozycję lub obrót w zadanych osiach, a także nałożyć materiał fizyczny, posiadający między innymi parametr, określający stopień sprężystości obiektu. Za interakcje fizyczne pomiędzy obiektami odpowiadają tzw. “collidery”, które mogą przyjmować w kolejności złożoności obliczeń kształt prostopadłościanu, sfery, kapsuły lub wypukłej siatki 3D.

Praca w Unity odbywa się w dwóch trybach – edycji (Edit mode) i gry (Play mode). W części edycji wyróżniamy:

- widok projektu - zawartość folderu "Assets", gdzie umieszczone są wszystkie części składowe gry,
- widok hierarchii - zbiór wszystkich obiektów używanych w aktualnej scenie,
- widok sceny – pozwala na ustalenie współrzędnych obiektów w danej scenie,
- widok gry – pokazuje obraz renderowany z kamer przedstawiający końcowy wygląd aplikacji,

- inspektor – wyświetla wszystkie właściwości, komponenty aktualnie wybranego obiektu. Pozwala na ich dodawanie i edycję.



W trybie gry możemy przetestować działanie i wygląd aplikacji. Przyciski Play, Pause, Stop pozwalają na płynne przechodzenie między trybami w celu dostosowywania parametrów gry.

Sercem silnika gry jest jego główna pętla, na najwyższym poziomie składająca się z:

- Odczytu danych wejściowych użytkownika
- Aktualizacji wewnętrznego stanu gry (np. fizyki)
- Renderingu obrazu

Liczba iteracji pętli w ciągu sekundy nazywana jest ilością klatek na sekundę (FPS - frames per second). Wartość ta powinna być wyższa niż 25, aby uzyskać efekt animacji. W współczesnych grach wartość ta wynosi 30 lub 60. W naszej aplikacji ustaliliśmy maksymalną ilość klatek na 60.

Silnik Unity pracuje w trybie jednowątkowym. Pozwala to zachować ustaloną kolejność działania poszczególnych funkcji, tak zwanej pętli gry (“game loop”) [8], w której wyróżniamy etapy i zdarzenia takie jak:

1. Wczytywanie sceny

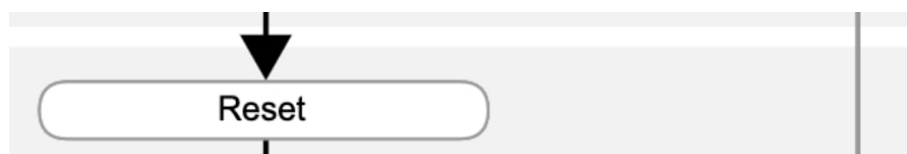


Następujące funkcje wywoływane są raz dla każdego obiektu, gdy wczytujemy nową scenę:

Funkcja **Awake()** wywoływana jest jako pierwsza, przed funkcją **Start()**, nawet gdy obiekt jest nieaktywny. Służy do inicjalizowania zmiennych oraz do tworzenia nowych instancji obiektów w trakcie działania aplikacji.

**OnEnable()** jest wywoływana podczas aktywowania obiektu (obiekt może być w dwóch stanach - aktywny lub nieaktywny).

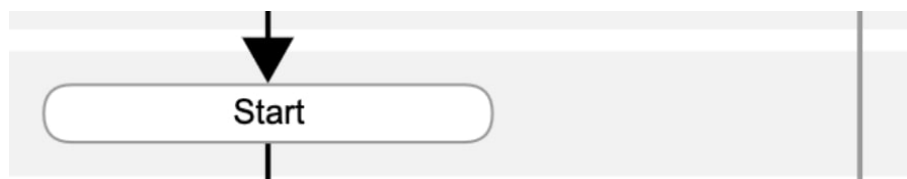
## 2. Edytor



[9]

Funkcja **Reset()** wywoływana jest w momencie wyboru elementu menu Reset wywoływanego ikoną zębatki. Służy do domyślnej inicjalizacji parametrów podczas dodawania komponentu lub przywrócenia jego wartości domyślnych.

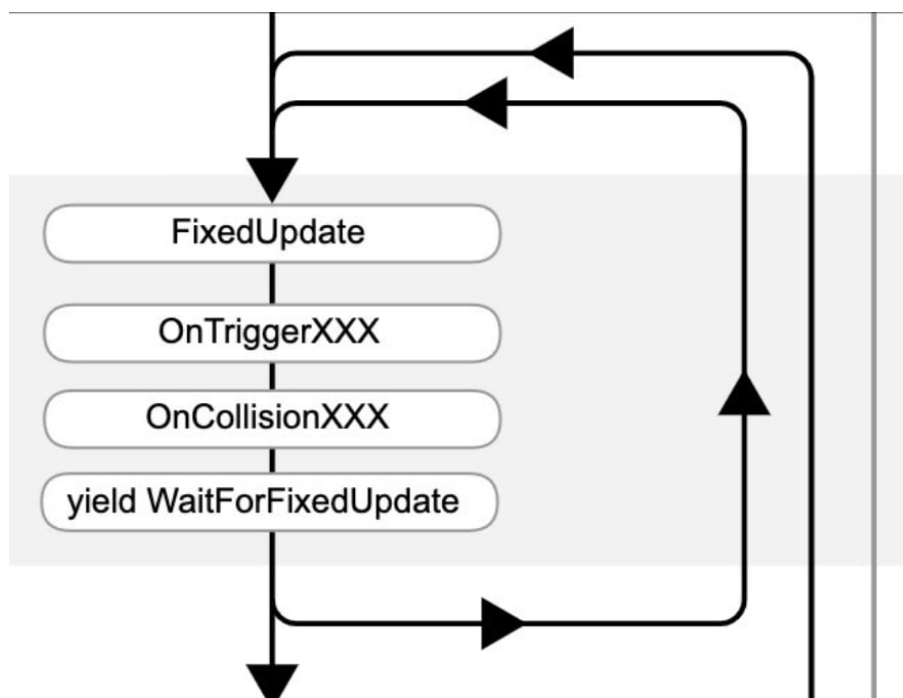
## 3. Inicjalizacja



[9]

Funkcja wywoływana jest tylko wtedy, gdy instancja skryptu jest aktywna. Uruchamiana jest dla każdego skryptu, przed pierwszą klatką działania aplikacji. W przypadku obiektów instancjonowanych w trakcie gry, **Start()** zostanie wykonane przed metodą **Update()**.

## 4. Fizyka gry



[9]

Funkcja **FixedUpdate()** w przeciwieństwie do **Update()** i **LateUpdate()** jest wywoływana stałą (ustaloną w edytorze) ilością razy na sekundę. Dzięki uniezależnieniu się od liczby wyświetlanych klatek nadaje się idealnie do obliczeń związanych z fizyką gry (przy tego typu obliczeniach kluczowy jest stały krok czasu).

**OnTriggerXXX()** i **OnCollisionXXX()** wywoływane są podczas zderzeń obiektów. Wszystkie obiekty biorące udział w zderzeniu muszą mieć przypisany “collider”. Wyróżniamy dwa rodzaje zderzeń - Trigger i Collision, które wysyłają odpowiednie informacje gdy nastąpi kolizja z innym obiektem (**Enter**, **Stay**, **Exit**). Różnica polega na ignorowaniu obiektów oznaczonych jako **isTrigger** przez silnik fizyczny gry – otrzymujemy dane obiektów, które wywołały zdarzenie – w przeciwieństwie do Collision, gdzie dostajemy pełne informacje o kolizji i w zależności od właściwości komponentu Rigidbody, następują odpowiednie przemieszczenia w wyniku zderzenia.

Każdy “collider” ma ustawioną jedną z trzech właściwości, które określają, jakiego rodzaju kolizję mogą wystąpić:

- **Static**: nie posiadają komponentu Rigidbody, silnik fizyczny na nie nie oddziałuje.
- **Dynamic**: typ domyślny, zderzenia tych obiektów są rozpatrywane przez silnik fizyczny gry.
- **Kinematic**: informują silnik fizyczny, że obiekt porusza się, ale za jego ruch odpowiada programista. Oddziałuje na inne obiekty fizyczne, natomiast



na niego nie oddziałuje fizyka.

Collision detection occurs and messages are sent upon collision						
	Static Collider	Rigidbody Collider	Kinematic Rigidbody Collider	Static Trigger Collider	Rigidbody Trigger Collider	Kinematic Rigidbody Trigger Collider
Static Collider		Y				
Rigidbody Collider	Y	Y	Y			
Kinematic Rigidbody Collider		Y				
Static Trigger Collider						
Rigidbody Trigger Collider						
Kinematic Rigidbody Trigger Collider						

Trigger messages are sent upon collision						
	Static Collider	Rigidbody Collider	Kinematic Rigidbody Collider	Static Trigger Collider	Rigidbody Trigger Collider	Kinematic Rigidbody Trigger Collider
Static Collider					Y	Y
Rigidbody Collider				Y	Y	Y
Kinematic Rigidbody Collider				Y	Y	Y
Static Trigger Collider		Y	Y		Y	Y
Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y
Kinematic Rigidbody Trigger Collider	Y	Y	Y	Y	Y	Y

Tabela zdarzeń jakie mogą wystąpić [10]

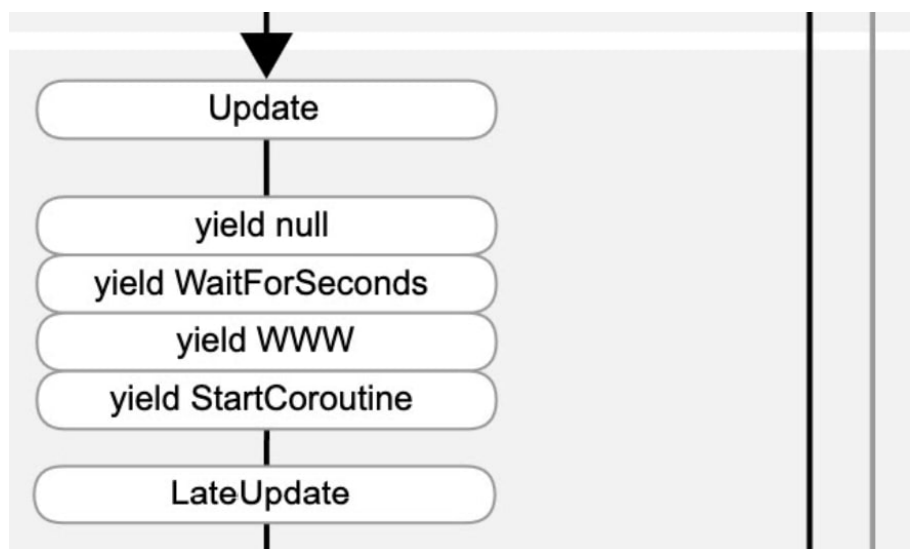
## 5. Wejście



[9]

Zbiór funkcji umożliwiający wykrycie m.in. zdarzeń dotyczących interakcji z “colliderem” obiektu takich jak: poruszenie kursora, najechanie, zabranie kursora, naciśnięcie lub zwolnienie przycisku myszy.

## 6. Logika gry

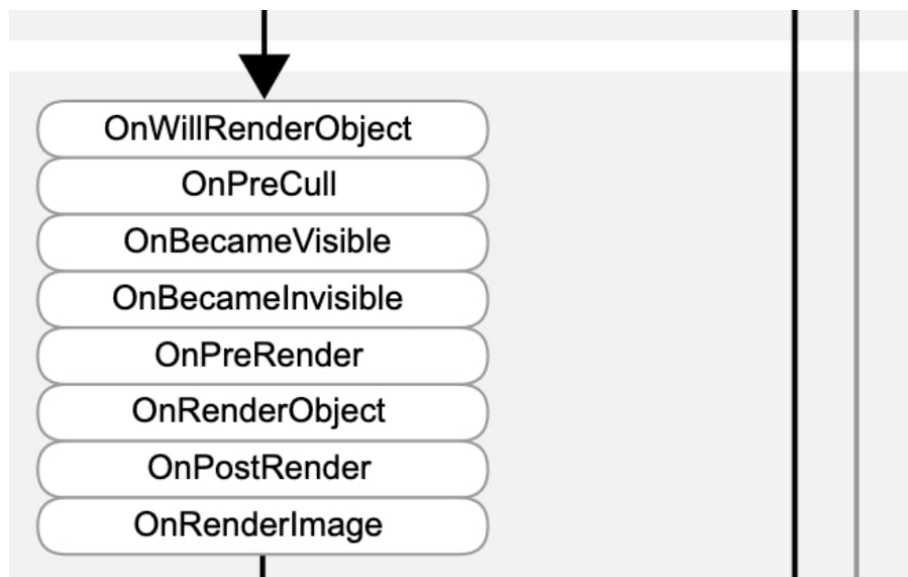


[9]

**Update()** jest najczęściej wywoływaną funkcją w większości skryptów. Znajduje się w niej większość kodu sterującego. Wywoływana jest raz na klatkę w głównej pętli gry.

**LateUpdate()** działa analogicznie, natomiast dzięki temu, że wywołuje się po wykonaniu wszystkich funkcji **Update()** mamy możliwość umieszczenia tam tzw. późnych obliczeń.

## 7. Renderowanie sceny



[9]

Moment wywołania funkcji na tym etapie jest uzależniony od renderowanej sceny lub obiektu.

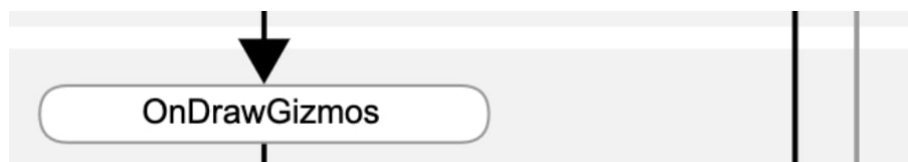
**OnBecameVisible()/OnBecameInvisible()** wywoływana jest, gdy obiekt staje się widoczny/niewidoczny w ostrosłupie widzenia aktywnej kamery.

**OnPreRender()** wywoływana przed rozpoczęciem renderowania sceny.

**OnPostRender()** wywoływana po wyrenderowaniu sceny.

**OnRenderImage()** często wykorzystywana do efektów post processingowych na ramce obrazu.

## 8. Gizmo

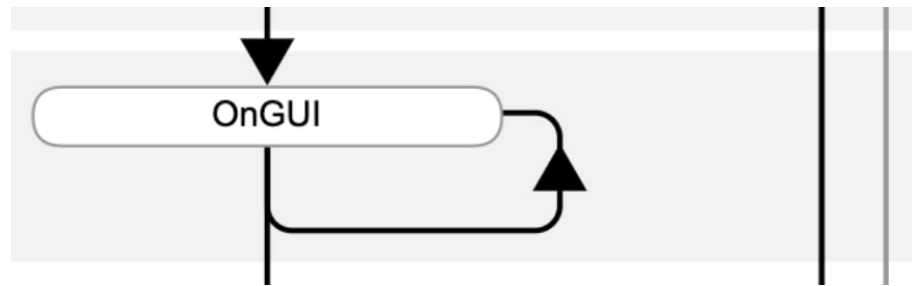


[9]

Obiekty typu Gizmo służą do rysowania kształtów, linii widocznych tylko podczas debugowania (działają tylko w trybie edytora). Pomagają w określaniu odległości, obszaru, kierunku światła, kamery, zasięgu rozchodzenia się dźwięku i tym podobnych.

Funkcja **OnDrawGizmos()** jest używana w każdej klatce działania aplikacji, a **OnDrawnGizmosSelected()** tylko dla wybranych obiektów.

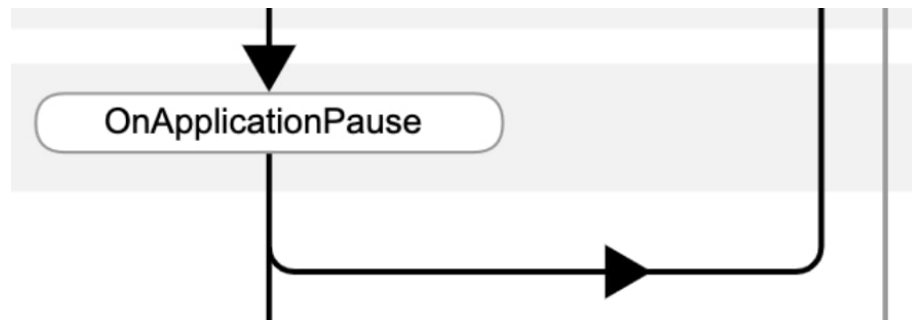
## 9. GUI - graficzny interfejs użytkownika



[9]

Funkcja odpowiadająca za renderowanie i obsługę zdarzeń GUI, takich jak wciśnięcie przycisku, aktualizacja interfejsu. Może być wywoływana wielokrotnie podczas każdej klatki.

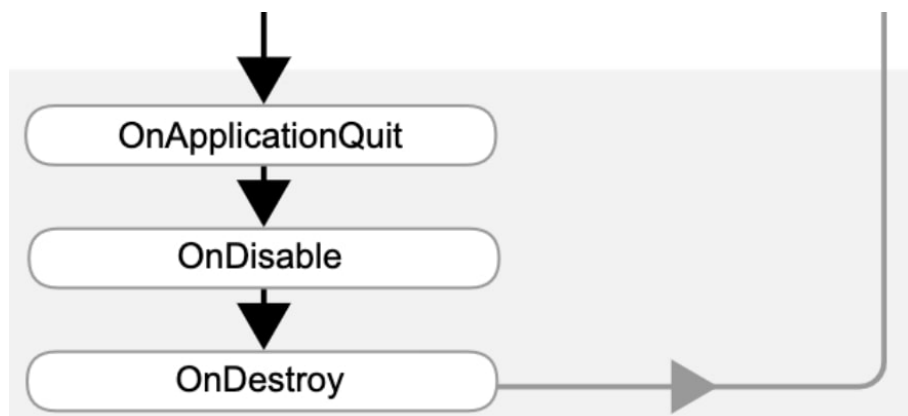
## 10. Pomiędzy klatkami



[9]

W momencie użycia przycisku Pauza w edytorze funkcja **OnApplicationPause()** jest wywołana na końcu aktualnej klatki. Po jej wywołaniu zostanie wygenerowana jeszcze jedna klatka, pozwalająca wyświetlić grafikę przedstawiającą stan wstrzymania.

11. Wyłączanie - funkcje ostatniego etapu pętli wywoływane są dla wszystkich aktywnych obiektów w scenie



[9]

Funkcja **OnApplicationQuit()** jest wywoływana przed zamknięciem aplikacji.

**OnDisable()** zostanie wywołana, gdy obiekt zostanie dezaktywowany bądź wyłączony.

**OnDestroy()** wykorzystywana po zniszczeniu obiektu przy pomocy funkcji **Destroy()** albo przy zamykaniu danej sceny. Wykonywana w ostatniej klatce istnienia obiektu.

## Rozdział 8.

# Baza pacjentów i ustawień

Ostatnim elementem naszej aplikacji jest system zapisywania ustawień oraz wyników pacjentów. Skorzystaliśmy z zaimplementowanej w Unity klasy PlayerPrefs, która działaniem przypomina słownik, w którym możemy przechowywać liczby całkowite, zmiennoprzecinkowe i ciągi znaków. Kluczem tej struktury jest krotka składająca się z pseudonimu identyfikującego pacjenta oraz nazwy zmiennej, której wartość chcemy zapisać lub odczytać. Zmiana nicku pacjenta powoduje odczytanie odpowiadających mu ustawień lub pobranie wartości domyślnych. Za pomocą PlayerPrefs pamiętamy również ustawienie "szachownicy" – kodując jej stan za pomocą bitów w liczbie 32 bitowej całkowitoliczbowej. [11]



## Rozdział 9.

# Podsumowanie

Projekt jest testowany w praktyce od ponad roku. Widać zaangażowanie i skupienie pacjentów podczas rehabilitacji. Główny problem zastanego oprogramowania - zmęczenie ćwiczeniami, został rozwiązany. Zmotywowani do osiągania coraz lepszego wyniku gracze nie traktują treningów jako obowiązku, tylko jako rozrywkę. Opinia wykładowców i studentów z AWF we Wrocławiu podczas zebrania katedry była pozytywna, ocenili gry jako bardzo atrakcyjne. Projekt okazał się sukcesem. Mamy nadzieję, że nasz projekt skłoni innych do podjęcia podobnych działań, łączących informatykę z fizjoterapią dla dobra zdrowia pacjentów.





# Bibliografia

- [1] Aktualne Problemy Biomechaniki, Wydawnictwo Katedry Biomechatroniki, Tom z. 1 (2007) s. 145-152
- [2] Rafał Halik, Jakub Stokwiszewski, Wojciech Seroka, Bogdan Wojtyniak, Urazy u osób powyżej 60 roku życia w Polsce, Narodowy Instytut Zdrowia Publicznego - Państwowy Zakład Higieny Narodowy Program Zdrowia na lata 2016-2020, Warszawa 2018, s. 17
- [3] Czerwosz L. Szczepek E., Sokołowska B., Jurkiewicz J., Czernicki Z. (2012). Recognition of Posture and Gait Disturbances in Patients with Normal Pressure Hydrocephalus Using a Posturography and Computer Dynography Systems. Rozdział do książki p.t. Hydrocephalus. Dr Sadip Pant (Ed.), Wydawnictwo. Intech.
- [4] Leszek Czerwosz – Autoreferat rozprawy habilitacyjnej pt. ”Badanie przemieszczeń środka ciężkości – analiza i opracowanie oprogramowania.”[http://ibib.pl/images/ibib/grupy/AUTOREFERAT\\_habilitacja\\_L\\_Czerwosz.pdf](http://ibib.pl/images/ibib/grupy/AUTOREFERAT_habilitacja_L_Czerwosz.pdf) (widziane 2020.09.19)
- [5] Błaszczyk JW, Czerwosz L. Stabilność posturalna w procesie starzenia. Gerontol Pol. 2005; 13(1): 25-36.
- [6] Czerwosz L., Błaszczyk W.J., Mraz M., Curzytek M. (2009) „Application of Virtual Reality In Postural Stability Rehabilitation”. Konferencja „Virtual Rehabilitation” Haifa p. 214. DOI: 10.1109/ICVR.2009.5174252.
- [7] Dokumentacja Unity, wersji 2019.4 z dnia 25.08.2020: <https://docs.unity3d.com/Manual/>
- [8] <https://mwin.pl/poznac-funkcje-unity/>, 25.08.2020
- [9] Dokumentacja Unity, wersji 2019.4 z dnia 25.08.2020: <https://docs.unity3d.com/Manual/ExecutionOrder.html>
- [10] Dokumentacja Unity, wersji 2019.4 z dnia 25.08.2020: <https://docs.unity3d.com/Manual/CollidersOverview.html>
- [11] Dokumentacja Unity, wersji 2019.4 z dnia 25.08.2020: <https://docs.unity3d.com/ScriptReference/PlayerPrefs.html>