

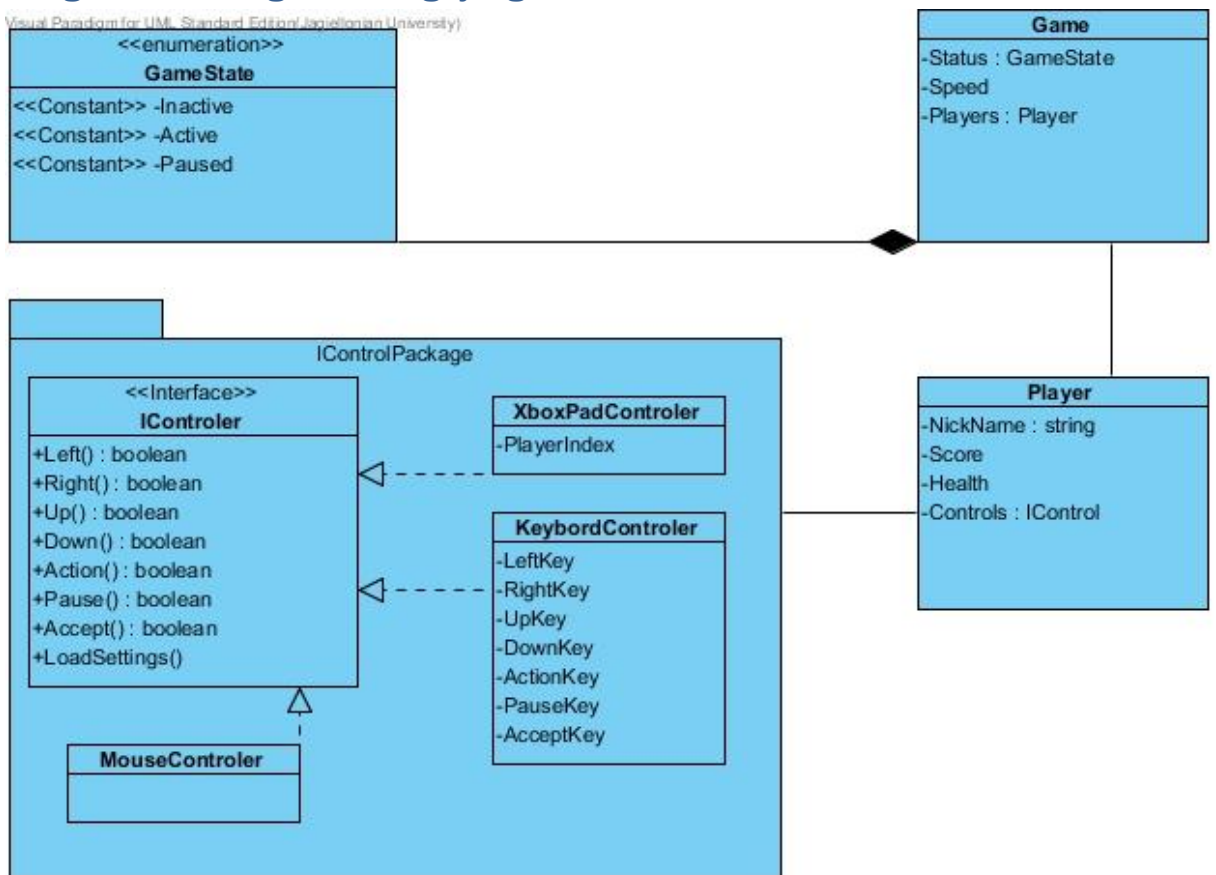
# Podróż Życia

## Diagramy UML

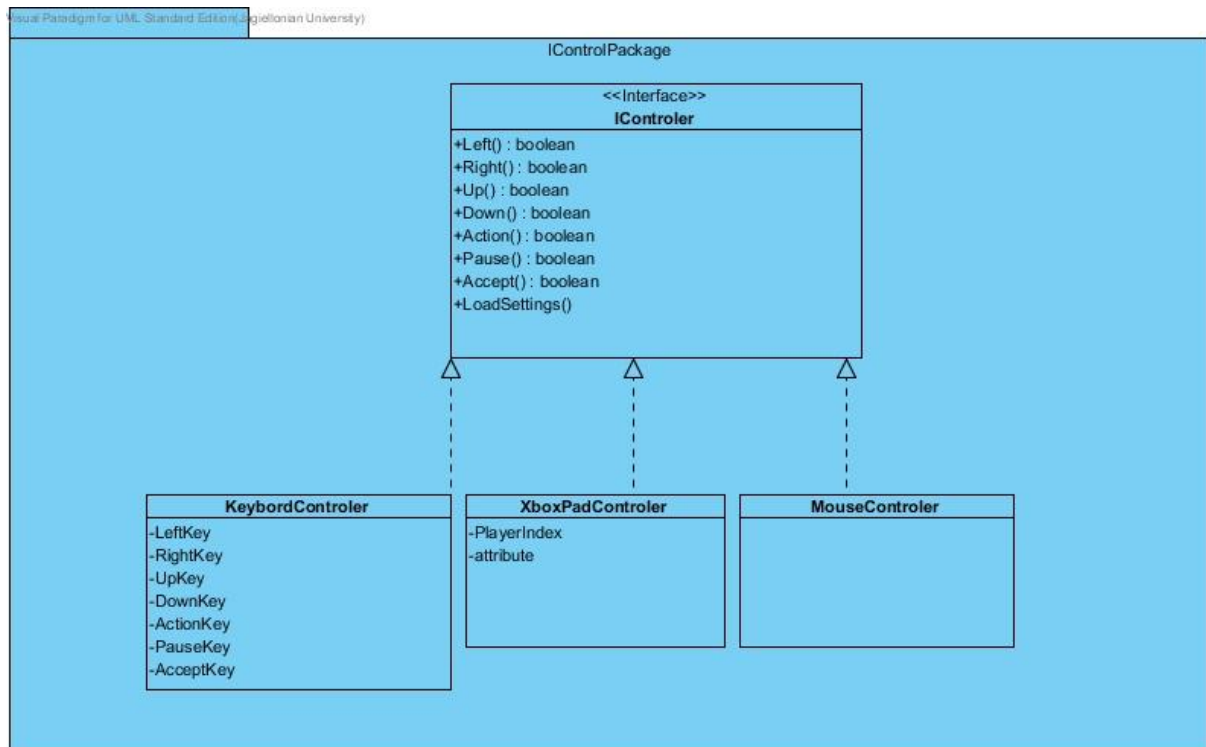
Mateusz Nowak  
Piotr Leniartek

## 1. Diagram klas – gra, stan gry i gracze

Visual Paradigm for UML, Standard Edition ( Jagiellonian University )

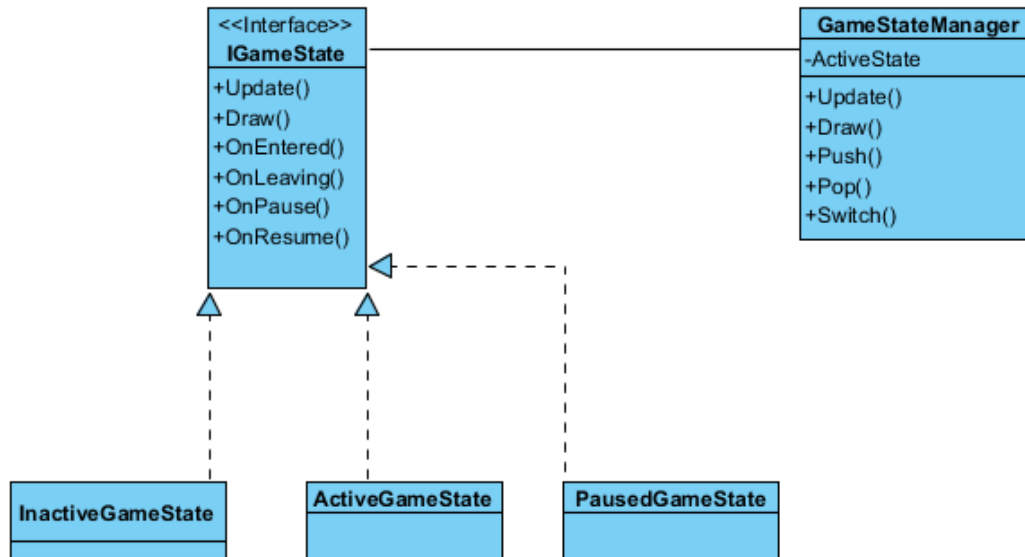


## 2. IControlPackage – klasy odpowiedzialne za „input” w grze



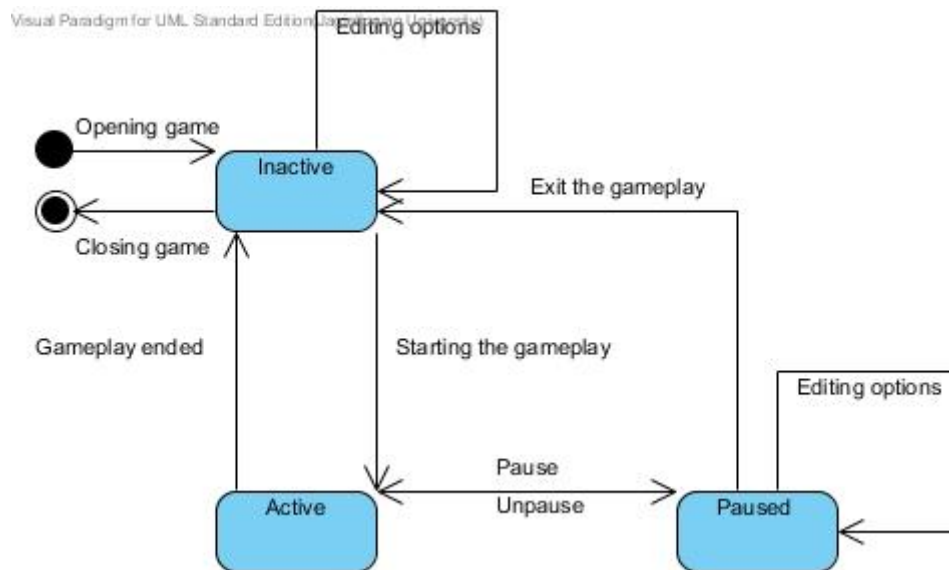
Klasy komunikujące się z użytkownikiem implementują interfejs IControler, który wykrywa akcje. Każda z tych klas definiuje akcję na swój sposób. Dodanie nowego kontrolera do gry to tylko dodanie jednej klasy do systemu.

### 3. Game State – klasy odpowiedzialne za obsługę stanów gry



Interfejs **IGameState** reprezentuje stan gry. **GameStateManager** jest odpowiedzialny za zarządzanie stanami w grze. Każdy stan musi implementować metody wywoływane w odpowiednich momentach przez **GameStateManager**.

#### 4. GameState Diagram – diagram maszyny stanowej prezentujący stany, w których może znajdować się rozgrywka.



Schemat pokazujący w jaki sposób przebiega cykl życia naszej gry, są to stany do których odwołujemy się w czasie wykonywania metod Update i Draw.

## 5. PlayerState Diagram – diagram maszyny stanowej reprezentujący w jakich stanach może znajdować się gracz.

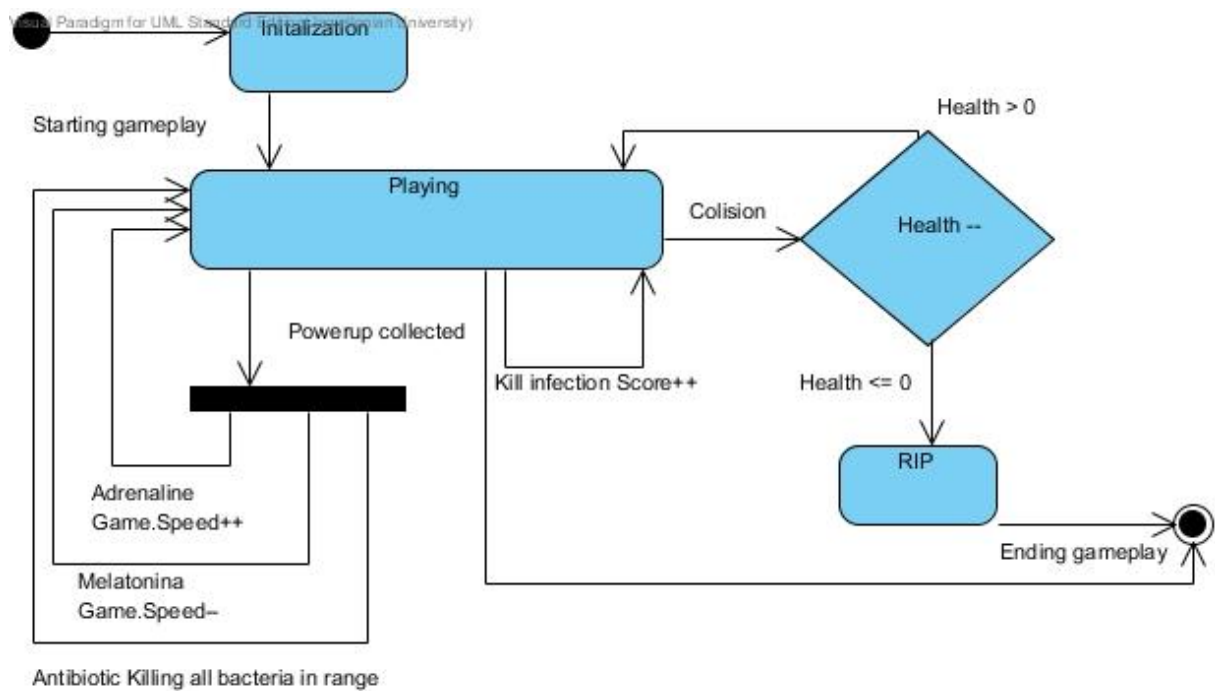


Diagram przedstawia interakcje jakie mogą zachodzić między graczem a otoczeniem w czasie rozgrywki. Po lewej u dołu znajdują się dostępne powerupy jakie można będzie znaleźć w grze.

**6. Menu State – diagram maszyny stanowej prezentujący jak można poruszać się po menu gry.**

