

Zadanie 3. – Gra Catan

Zespół Projektowy

November 19, 2025

1 Projekt systemu

Przykładowe diagramy UML znajdują się poniżej. Pełen diagram klas UML znajduje się poza dokumentem ze względu na jego rozmiar.

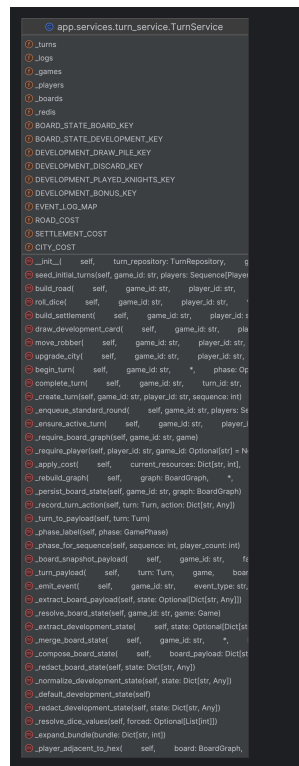


Figure 1: Diagram klas TurnService

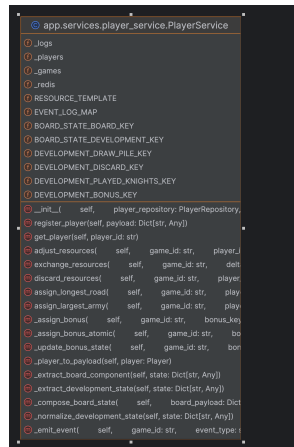


Figure 2: Diagram klas PlayerService

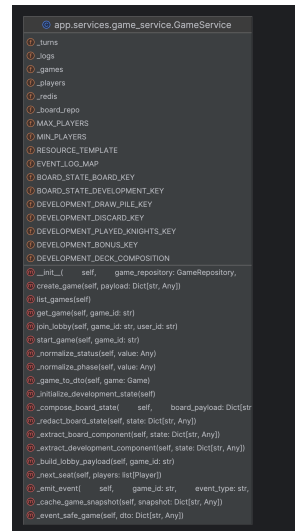


Figure 3: Diagram klas GameService

2 Ryzyka projektowe i plany naprawcze

2.1 Ryzyko 1: Opóźnienia w implementacji UI

Opis: UI gry planszowej jest złożony i może zajmować więcej czasu niż planowano.

Zapobieganie: Wczesne prototypowanie interfejsu.

Plan naprawczy: Ograniczenie zakresu animacji i elementów niemechanicznych.

2.2 Ryzyko 2: Integracja z backendem

Opis: API backendowe może się zmieniać podczas implementacji.

Zapobieganie: Ustalenie kontraktów API przed rozpoczęciem prac.

Plan naprawczy: Stworzenie warstwy adaptacyjnej w frontendzie.

2.3 Ryzyko 3: Problemy z synchronizacją stanu gry

Opis: Niespójności między stanem u klienta a serwerem.

Zapobieganie: Regularne odświeżanie stanu, unikanie logiki krytycznej po stronie klienta.

Plan naprawczy: Wymuszenie pełnej resynchronizacji po wykryciu niespójności.

3 Plan iteracji

3.1 Iteracja 1

- Działające lobby, tworzenie gry i dołączenie do istniejącej.
- Podstawowy layout i komponenty UI.
- Przydzielanie surowców i funkcjonowanie dróg, osad i miast.
- Budowanie, otrzymywanie i wydawanie surowców.

3.2 Iteracja 2

- Pełne działanie portów.
- Obsługa kart rozwoju i ich efektów.
- Implementacja zasad specjalnych (np. rycerze, największa armia).
- Negocjacja pomiędzy graczami.