



Projekt Gry Strategicznej Statki

Dokumentacja Inżynierii Wymagań

Jan Ściga
Mateusz Zdyrski
Adrian Chmielowiec

AKADEMIA GÓRNICZO-HUTNICZA,
WYDZIAŁ INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI

4 Listopad 2021

Spis treści

1	Wprowadzenie	1
1.1	Przedmowa	1
1.2	Zasady gry	1
1.3	Cel dokumentu	1
1.4	Macierz kompetencji zespołu	1
2	Słownik pojęć technicznych	2
3	Definicja wymagań użytkownika	2
4	Architektura systemu (high-level)	2
5	Specyfikacja wymagań systemu (Technical)	2
6	Model Systemu	2
6.1	Model systemu w postaci tabeli	2
6.2	Model systemu w postaci diagramu sekwencyjnego	3
6.3	Model systemu w postaci diagramu przypadków użycia	4
7	Podsumowanie	4
7.1	Sugerowany język implementacji	4

1 Wprowadzenie

1.1 Przedmowa

Okręty (*ang. Battleship*) to gra strategiczna przeznaczona dla dwóch osób. Została wynaleziona przez Clifforda Von Vicklera na początku XX wieku, a opatentowana przez Milton Bradley Company w 1943 roku. Gra jest znana w wielu krajach pod nazwą: gra w Statki lub Salvo. W polskiej literaturze można się też spotkać z nazwą: wojna morska lub bitwa morska.

1.2 Zasady gry

Gra toczy się na specjalnej planszy, każdy z graczy posiada po dwie plansze o wielkości 10 x 10 pól. Pola oznaczane są przez specjalne współrzędne, dzięki którym lokalizowane są statki na planszy gry. Okręty są różnej wielkości. Trafienie okrętu przeciwnika polega na strzale, który jest odgadnięciem położenia jakiegoś statku. Strzały oddawane są naprzemiennie, poprzez podanie współrzędnych pola. W przypadku strzału trafionego, gracz kontynuuje strzelanie (czyli swój ruch) aż do momentu chybienia. Wygrywa ten, kto pierwszy zatopi wszystkie statki przeciwnika.

1.3 Cel dokumentu

Podstawową intencją powstania dokumentu jest umożliwienie spełnienia wymagań klienta w zakresie dostarczenia produktu - gry strategicznej „Statki”. W tym celu zespół tworzący niniejszy dokument wyróżnia następujące cele dokumentu adresowane do wszystkich osób związanych z realizacją tego projektu:

- Przedstawienie wymagań klienta
- Przedstawienie wymagań do planowania procesu tworzenia oprogramowania
- Oszacowanie dostępnych zasobów
- Zrozumienia co projektowany system powinien robić
- Przedstawienie punktu wyjściowego do walidacji oprogramowania
- Przedstawienie zależności pomiędzy poszczególnymi komponentami systemu

1.4 Macierz kompetencji zespołu

Zespół projektujący dokumentację inżynierii wymagań przedstawia macierz kompetencji członków zespołu.

2 Słownik pojęć technicznych

3 Definicja wymagań użytkownika

4 Architektura systemu (high-level)

5 Specyfikacja wymagań systemu (Technical)

6 Model Systemu

6.1 Model systemu w postaci tabeli

Poniższa tabela przedstawia zbiór elementów wraz z opisami oraz wymaganiami, które ich dotyczą. Elementy te stanowią części składowe modelowanego systemu gry w Statki.

Tablica 1: Modelowanie systemu w postaci tabeli

Modelowany element	Opis
Aktorzy	Komputer, Gracze
Opis	Celem działania modelowanego systemu jest umożliwienie rozegrania gry w statki zarówno w trybie Gracz vs Gracz, Komputer vs Komputer jak i Gracz vs Komputer, przechowywania rankingu oraz podstawowych statystyk dotyczących rozegranych gier.
Dane	Dane wprowadzane są do komputera za pomocą myszki (dot. Trybu Gracz vs Gracz oraz Gracz vs Komputer). Komputer wprowadza swoje dane za pomocą ustalonych algorytmów.
Statystyki	Statystyki dotyczą rozegranych w przeszłości gier danego gracza i obejmują następujące wielkości: <ul style="list-style-type: none">• Ilość rozegranych gier• Ilość wygranych gier• Celność (rozumiana jako stosunek ilości trafień do ilości strzałów ogólnie)• Powyższe statystyki powinny być widoczne z rozbiciem na trzy poziomy trudności.• Powyższe statystyki powinny być widoczne również dla gier Gracz vs Gracz
Plansza	W momencie rozgrywki każdy z Graczy powinien widzieć dwie plansze: <ul style="list-style-type: none">• Plansza gdzie zaznaczane będą strzały oddane przez gracza• Plansza gdzie widoczne będą statki gracza wraz z ewentualnymi trafieniami zadanymi przez przeciwnika Plansza powinna mieć standardowe wymiary (10x10)
Widoki	W przypadku rozgrywki Gracz vs Gracz, każdy z nich powinien dysponować osobnym oknem gdzie znajdowałyby się jego plansze.

Rysunek 1: Caption

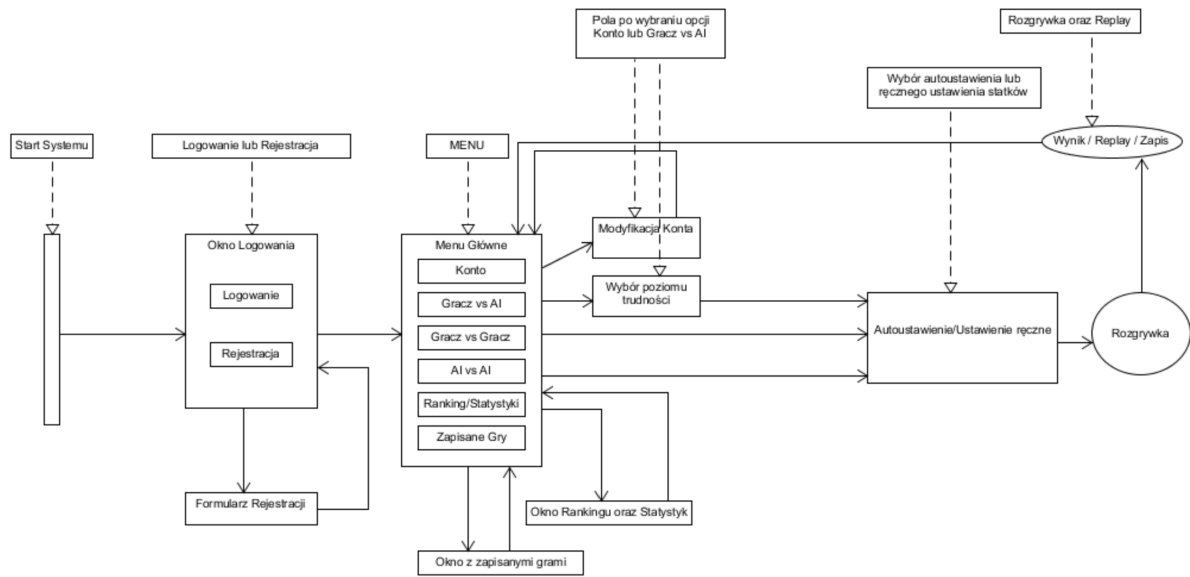
Tablica 1: Modelowanie systemu w postaci tabeli

Modelowany element	Opis
Statki	System powinien umożliwiać zarówno samodzielne jak i automatyczne ustawianie statków na planszy. Rodzaje dostępnych statków: <ul style="list-style-type: none"> • 1 X Statek 4 masztowy • 2 X Statek 3 masztowy • 3 X Statek 2 masztowy • 4 X Statek 1 masztowy
Wieloplatformowość	W przypadku realizacji aplikacji Webowej system powinien działać na przynajmniej dwóch z trzech wymienionych przeglądarek. <ul style="list-style-type: none"> • CHROME • SAFARI • FIREFOX W przypadku aplikacji desktopowej system powinien działać na systemach Windows oraz Linux.
Replay	System powinien umożliwiać odtworzenie dowolnej z zapisanych i rozegranych gier.
Hasło	System powinien umożliwiać logowanie gracza za pomocą nazwy oraz hasła z następującymi warunkami <ul style="list-style-type: none"> • Możliwość aktualizacji hasła • Hasło powinno zawierać co najmniej 8 znaków w tym co najmniej jedną cyfrę, znak specjalny oraz dużą literę
Konto	Każdy gracz powinien posiadać nazwę oraz avatar, który będzie mógł zaktualizować. System powinien umożliwiać założenie nowego konta.
Gra Komputera	Każdy ruch komputera powinien odbywać się co jedną sekundę
Poziomy trudności	W przypadku rozgrywki z komputerem, gracz powinien mieć możliwość wyboru jednego z trzech poziomu trudności (ŁATWY, ŚREDNI, TRUDNY)
Sygnalizacja trafienia	Sygnalizacja trafienia powinna być przekazywana przez krótkie wyświetlenie komunikatu np. Czerwonego napisu „TRAFIENIE !”
Wynik	Wynik (tzn. nazwa zwycięzcy powinien być widoczny po zatopieniu statków przeciwnika wraz z możliwością powrotu do menu głównego, odtworzenia zakończonej gry i zapisania gry)

6.2 Model systemu w postaci diagramu sekwencyjnego

Poniższy diagram w sposób ogólny przedstawia zarys modelowanego systemu, dostępne okna dla użytkownika oraz sekwencje działań które może wykonać w systemie.

Rysunek 2: Diagram sekwencyjny modelowanego systemu



6.3 Model systemu w postaci diagramu przypadków użycia

Poniższe diagramy pozwalają przedstawić różne rodzaje gier oraz widoki, które powinny być dostępne dla grających stron.

7 Podsumowanie

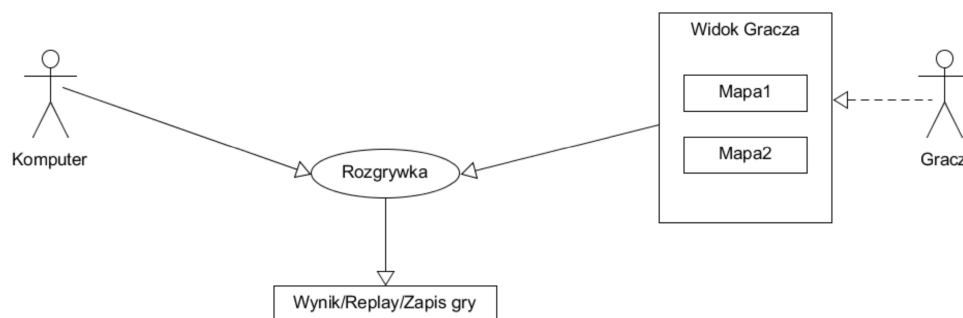
7.1 Sugerowany język implementacji

Sugerowanym językiem implementacji jest Python

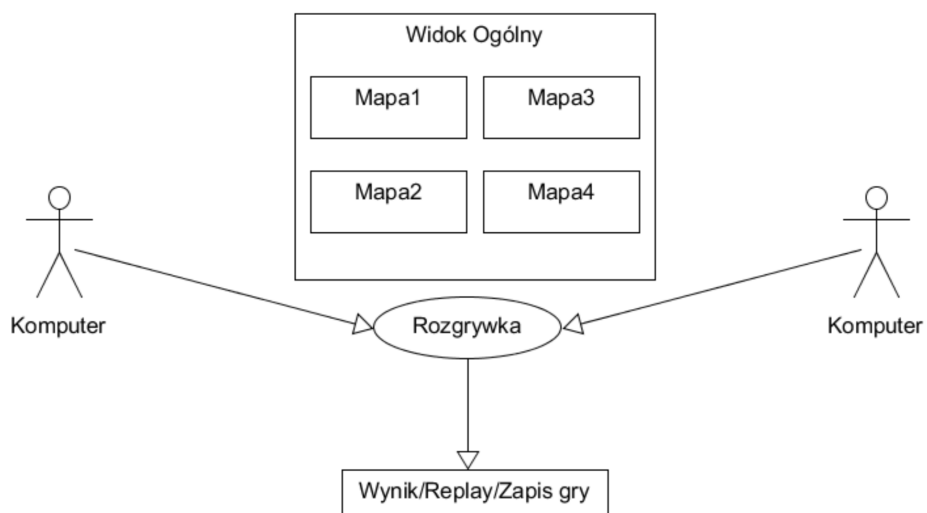
Python jest jednym z najprostszych a zarazem jednym z najbardziej wszechstronnych języków programowania, nie należy jednak błędnie interpretować jego pozornie łatwej składni i semantyki jako nieodpowiedniego do programowania gier. Python oferuje programistom w pełni rozwinięte techniki OOP (programowanie zorientowane obiektowo), podobnie jak C++ i Java.

Framework Pygame, oparty na tym popularnym języku, nabiera tempa każdego dnia, dzięki możliwości umożliwienia programistom szybszego prototypowania swoich gier. Python zyskuje sławę jako jeden z najlepszych języków programowania gier wideo.

Rysunek 3: Diagram przypadku dla gry Gracz vs Komputer



Rysunek 4: Diagram przypadku dla gry Komputer vs Komputer



Rysunek 5: Diagram przypadku dla gry Gracz vs Gracz

