

# Specyfikacja Funkcjonalna z Projektu *„Tanks”*

Daniel Ślusarczyk i Jakub Łaba

28.04.2021

# Spis treści

1	Informacje ogólne . . . . .	1
1.1	Przeznaczenie dokumentu . . . . .	1
1.2	Przeznaczenie programu . . . . .	1
1.3	Wymagania programu . . . . .	1
1.4	Problematyka – wprowadzenie teoretyczne . . . . .	1
1.5	Cel Projektu . . . . .	2
2	Strona funkcjonalna programu . . . . .	3
2.1	Uruchomienie programu . . . . .	3
2.2	Sterowanie . . . . .	3
2.3	Funkcjonalności programu . . . . .	3
2.4	Wynik działania programu . . . . .	3
3	Interfejs graficzny . . . . .	4
3.1	Okno Menu . . . . .	4
3.2	Okno Ustawień . . . . .	6
3.3	Okno Gry . . . . .	8
4	Parametry działania programu . . . . .	10
4.1	Plik konfiguracyjny . . . . .	10
4.2	Modyfikowalne parametry . . . . .	11
4.3	Struktura pliku konfiguracyjnego . . . . .	12
4.4	Przykładowy plik konfiguracyjny . . . . .	12
5	Przebieg działania programu . . . . .	13
5.1	Scenariusz powszechny . . . . .	13
5.2	Scenariusze ewentualne . . . . .	13

## 1 Informacje ogólne

### 1.1 Przeznaczenie dokumentu

Przeznaczeniem niniejszego dokumentu jest zawarcie szczegółów przedstawianego oprogramowania oraz omówienia zakresu prac, funkcjonalności produktu, a także interakcji zachodzących pomiędzy użytkownikiem a programem. Dokument ten pozwala na zapoznanie się z tematyką projektu i trudnościami z nim związanymi. Ponadto jest niejako instrukcją obsługi pozwalającą na zgłębienie problemu prawidłowego i świadomego korzystania z oprogramowania.

### 1.2 Przeznaczenie programu

Program służy do przeprowadzenia prostej gry opartej na rywalizacji dla dwóch graczy. Pozwala na indywidualne ustawienie konfiguracji przebiegu rozgrywki poprzez dobranie preferowanych parametrów. Niemniej jednak, oprogramowanie „Tanks” jest dedykowane średnio-zaawansowanym informatycznie użytkownikom.

### 1.3 Wymagania programu

Program „Tanks” jest wieloplatformowym oprogramowaniem przeznaczonym głównie dla systemu „Windows”. Prawidłowe działanie wymaga posiadania zainstalowanego środowiska uruchomieniowego Javy (JRE) służącego do uruchamiania aplikacji z rozszerzeniem .jar. Wszystkie potrzebne komponenty powinny zostać przygotowane przez dołączony instalator po przejściu przez proces instalacji.

### 1.4 Problematyka – wprowadzenie teoretyczne

Gra jest dedykowana dwóm graczom rozmieszczonym po dwóch stronach okna rozgrywki, których zadaniem jest strzelanie do komórek w środkowej części za pomocą przypisanych czołgów. Gracze mają możliwość poruszania się swoimi jednostkami w górę i w dół, oraz obracania lufą czołgu +/- 60 stopni względem osi OX (prostej poziomej). Każdy pojazd strzela okrągłymi pociskami lecącymi z określoną prędkością, ograniczonych do pewnej ilości pocisków wystrzelonych jednocześnie. Dodatkowo na planszy może znajdować się tylko ograniczona ilość pocisków jednego gracza. Kwadratowe komórki, które są obiektem przyznawania punktów graczom dzielą się na trzy rodzaje:

- **Zwykła komórka** –kwadratowy obiekt o wszystkich krawędziach wrażliwych na kontakt z pociskiem. Punkty za jej unicestwienie przyznawane są graczowi zadającemu końcowe obrażenia.
- **Bomba** – obiekt posiadający pancerne krawędzie z pominięciem górnej, która jest jedynym sposobem na zastrzelenie tej komórki. Dokonanie tego kończy grę.
- **Kolonia** – zbiór kwadratowych obiektów o wszystkich krawędziach wrażliwych na kontakt z pociskiem w losowej konfiguracji maksymalnie pięciu komórek. Gracz otrzymuje punkty o określonej wartości pomnożonej przez liczbę komórek należących do kolonii po zastrzeleniu ostatniej komórki w zbiorze.

Wszystkie wyżej wymienione komórki poruszają się od górnej krawędzi okna rozgrywki do dolnej z określoną prędkością i mogą zostać zastrzelone tylko w momencie ich widoczności na

polu bitwy. Ich aktualna wartość potrzebna do zastrzelenia znajdują się w centralnej części komórki.

Gra w sposób regularny zwiększa trudność rozgrywki poprzez zwiększanie wartości potrzebnej do unicestwienia komórki, szybkości komórek i wystrzeliwanych pocisków, oraz zmniejszanie promienia pocisku i długości boku komórki. Koniec gry przewidują dwa scenariusze: upływanie określonego czasu rozgrywki lub zastrzelenie komórki bomby. Grę wygrywa gracz o większej ilości punktów w momencie zakończenia.

## **1.5 Cel Projektu**

Celem projektu „Tanks” jest stworzenie prostej gry z interfejsem graficznym w języku Java. Założeniami projektu są:

- Zawarcie w interfejsie gry liczby punktów obu graczy, okna komunikatów, licznika pozostałego czasu do zakończenia rozgrywki, liczby pocisków wystrzelonych przez graczy oraz okna z menu,
- Zachowanie podejścia obiektowego w kodzie,
- Zapisanie wyniku w formie pliku graficznego,
- Przetestowanie kodu za pomocą automatycznych testów jednostkowych (JUnit),
- Zachowanie wieloplatformowości oprogramowania.

## 2 Strona funkcjonalna programu

### 2.1 Uruchomienie programu

Przy pierwszym uruchomieniu należy uruchomić instalator „*InstallTanks.exe*” i postępować zgodnie z instrukcjami na ekranie. Instalator utworzy skrót aplikacji w wybranej przez użytkownika lokalizacji. Przy następnych uruchomieniach należy jedynie wybrać utworzony w wyniku instalacji skrót – gra przed rozpoczęciem rozgrywki pozwala użytkownikowi na wprowadzenie własnych ustawień za pośrednictwem menu.

### 2.2 Sterowanie

Efekt	Przycisk
Ruch czołgu w górę	‘W’ dla gracza lewego, strzałka w górę dla gracza prawego
Ruch czołgu w dół	‘S’ dla gracza lewego, strzałka w dół dla gracza prawego
Obrót lufy w górę	‘A’ dla gracza lewego, strzałka w prawo dla gracza prawego
Obrót lufy w dół	‘D’ dla gracza lewego, strzałka w lewo dla gracza prawego
Oddanie strzału	Spacja dla gracza lewego, prawy Shift dla gracza prawego
Pauza	Escape

### 2.3 Funkcjonalności programu

Ekran ustawień, dostępny za pośrednictwem menu głównego, pozwala na dostosowywanie rozgrywki do potrzeb i wymagań użytkownika na wiele sposobów – pozwala wybrać jeden z trzech predefiniowanych poziomów trudności, lub dostosować wszystkie parametry rozgrywki ręcznie.

### 2.4 Wynik działania programu

Po zakończeniu rozgrywki – w wyniku upływu czasu przeznaczonego na rozgrywkę bądź w wyniku trafienia przez jednego z graczy komórki-bomby – aplikacja wykona zrzut ekranu. Ponadto, wspomniane wyżej ustawienia gry będą umożliwiać zapisanie pożądaných parametrów rozgrywki w formie pliku konfiguracyjnego. W wyniku działania programu mogą powstać więc 2 rodzaje plików – graficzne oraz konfiguracyjne.

### 3 Interfejs graficzny

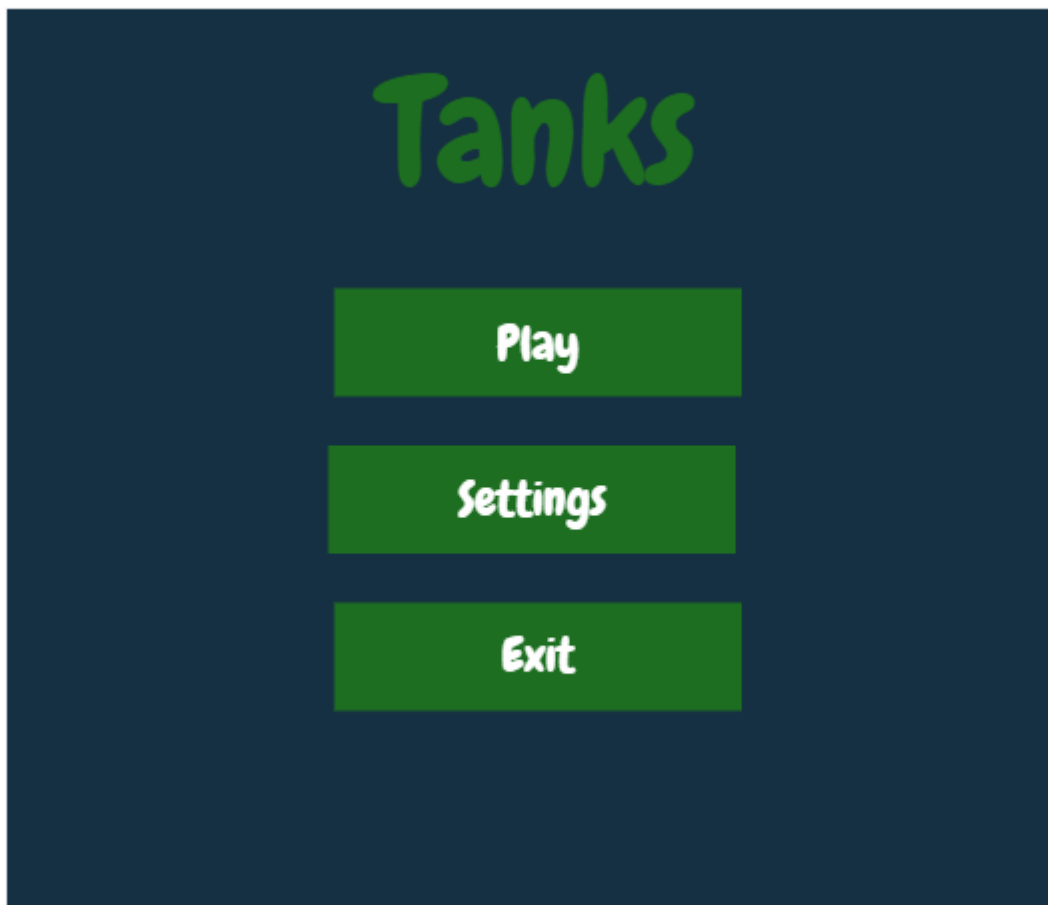
#### 3.1 Okno Menu

Elementami obowiązkowymi okna menu są trzy przyciski: jeden odpowiedzialny za początek gry, drugi odpowiedzialny za przejście do ustawień, trzeci umożliwiający wyjście z gry. Na samej górze powinna znajdować się nazwa gry w sposób maksymalnie ostentacyjny. Poniższe dwa rysunki przedstawiają możliwe rozmieszczenie komponentów i proponowany styl dla tego okna.



Rysunek 1: Szablon interfejsu okna menu

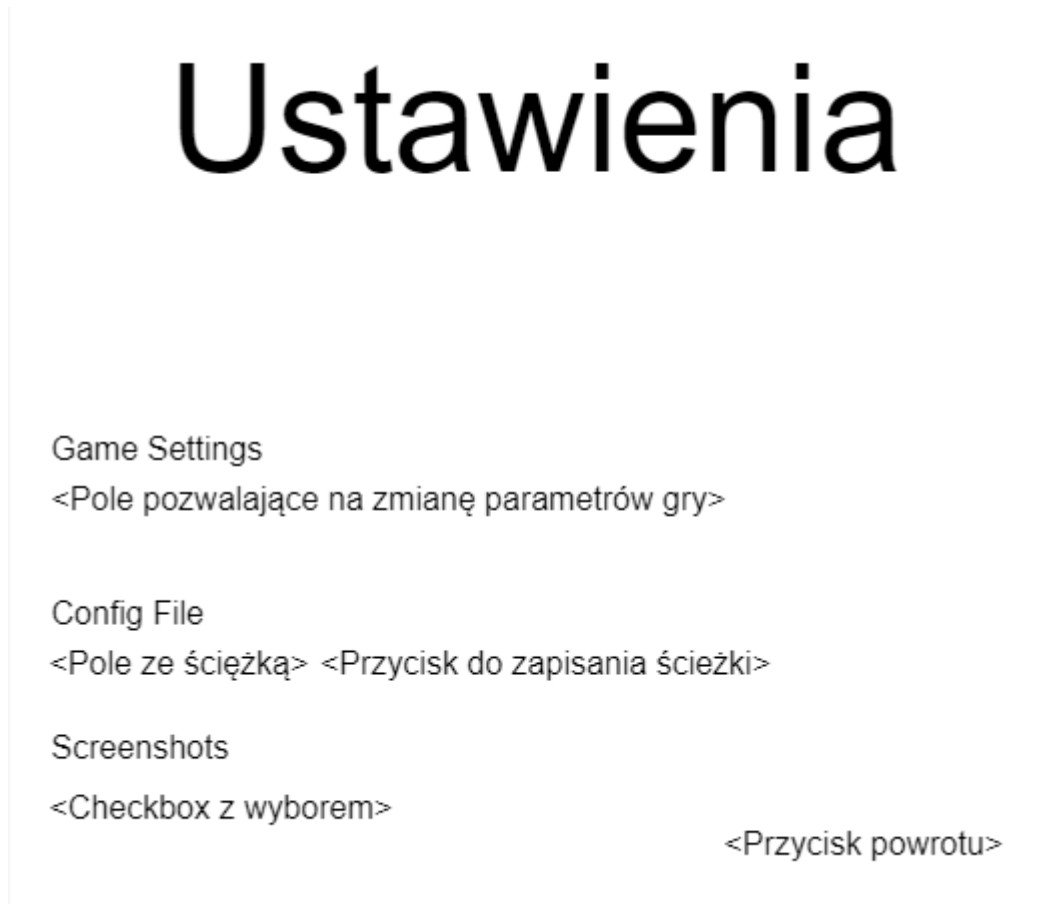
## Wersja pogładowa



Rysunek 2: Wersja pogładowa okna menu

### 3.2 Okno Ustawień

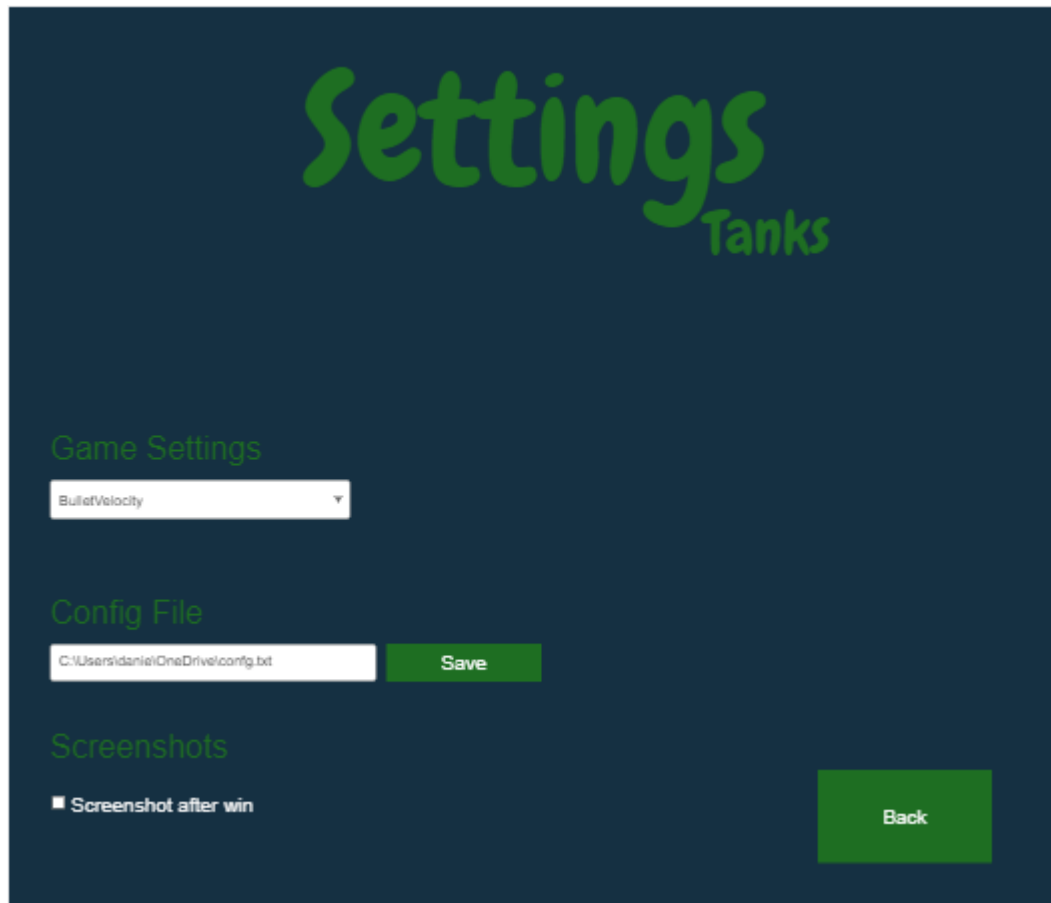
Elementami obowiązkowymi okna ustawień jest pole umożliwiające modyfikowanie parametrów domyślnych używanych tylko w przypadku problemów z plikiem konfiguracyjnym. Ważnymi aspektami tego okna są również pole umożliwiające podanie ścieżki do pliku konfiguracyjnego, *checkbox* z wyborem, czy po rozgrywce ma zostać wykonany zrzut ekranu i przycisk umożliwiający powrót do okna menu. Poniższe dwa rysunki przedstawiają możliwe rozmieszczenie komponentów i proponowany styl dla tego okna.



Rysunek 3: Szablon interfejsu okna ustawień



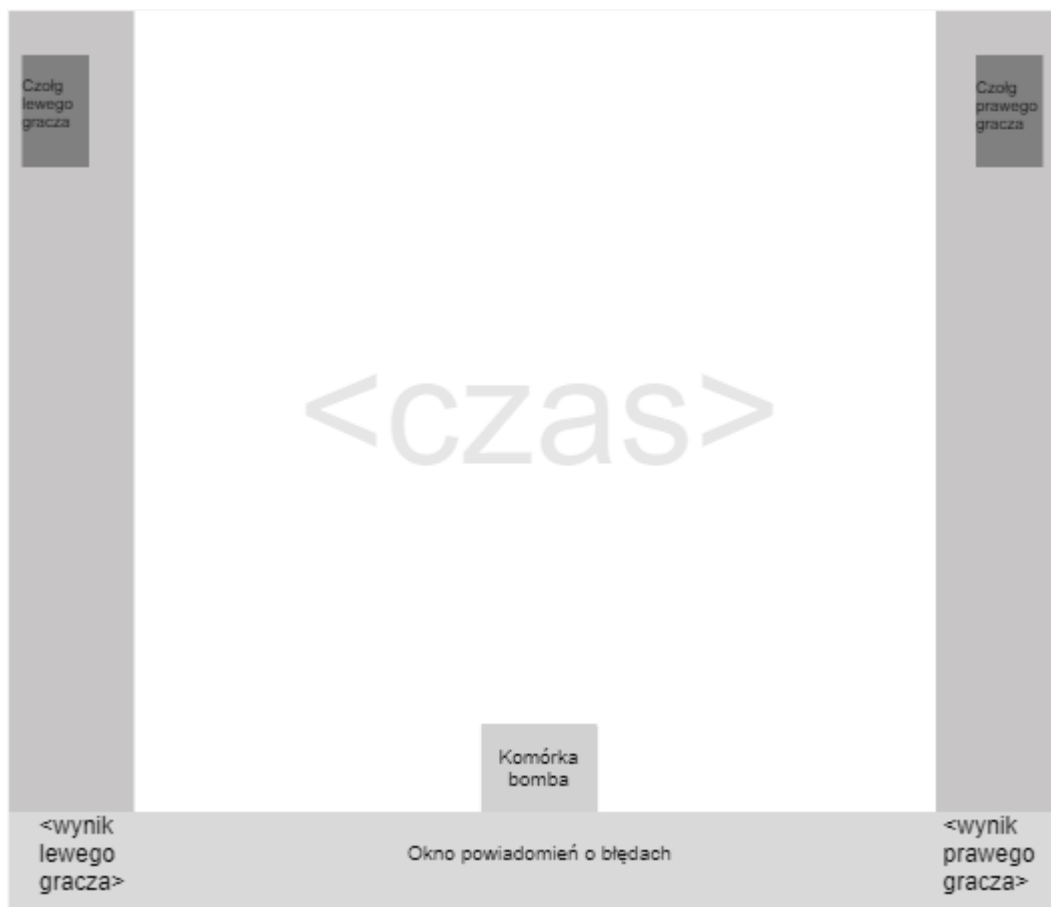
## Wersja pogładowa



Rysunek 4: Wersja pogładowa okna ustawień

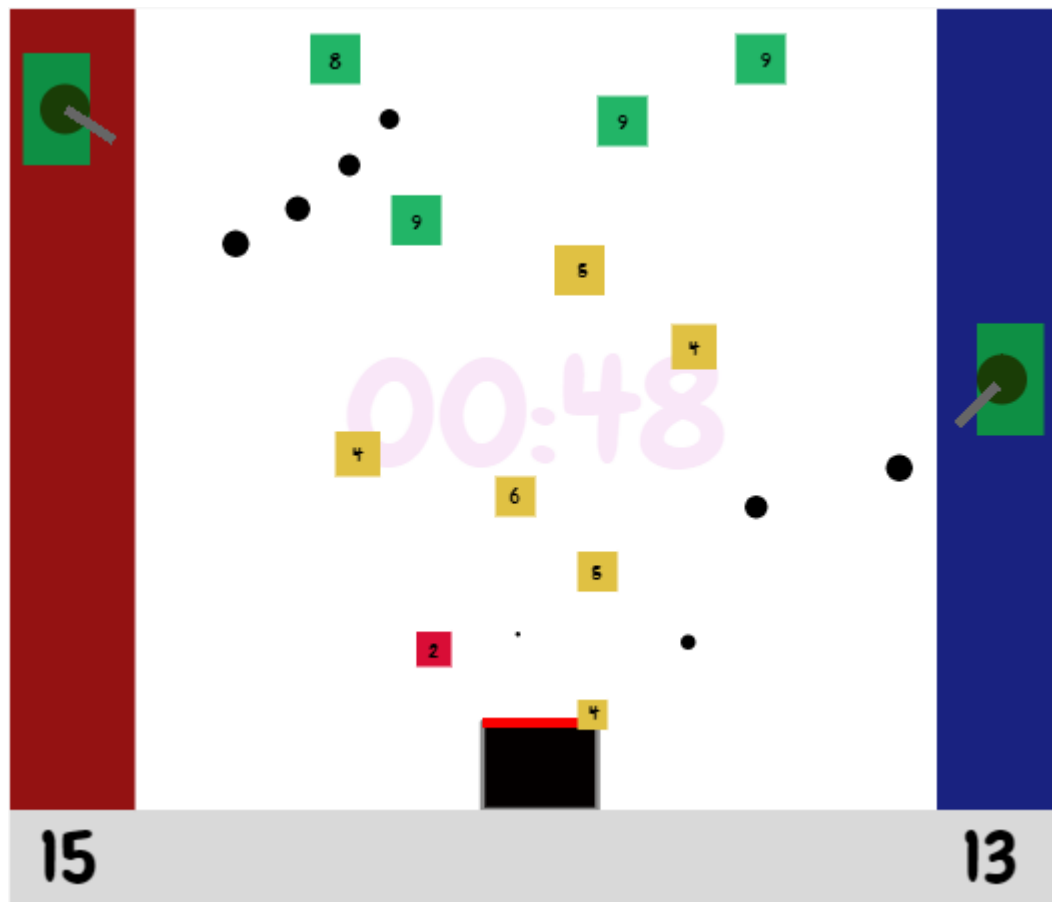
### 3.3 Okno Gry

Elementami obowiązkowymi okna gry są komponenty umożliwiające identyfikację zwycięzcy wśród obu graczy (pola wyświetlające wynik). Dodatkowo *pole bitwy* powinno być widocznie odznaczone od reszty pól, a sam środek wypełniać licznik czasu do końca rozgrywki. Lewy i prawy skraj planszy jest przeznaczony na poruszanie się czołgów, a dolna krawędź *pola bitwy* na samym środku dla bomby. Pod nią znajduje się pole wyświetlające komunikaty. Poniższe dwa rysunki przedstawiają możliwe rozmieszczenie komponentów i proponowany styl dla tego okna.



Rysunek 5: Szablon interfejsu okna rozgrywkowego

## Wersja pogładowa



Rysunek 6: Wersja pogładowa okna rozgrywki

## **4 Parametry działania programu**

### **4.1 Plik konfiguracyjny**

Plik konfiguracyjny jest zbiorem danych ustalającym szczegółowe zasady działania gry i pozwalającym na zastąpienie nimi wartości domyślnych przyjmowanych przez programu przy braku pliku konfiguracyjnego.

## 4.2 Modyfikowalne parametry

Indentyfikator	Nazwa	Opis
V1	Bullet Velocity	Szybkość pocisków wystrzeliwanych przez graczy. Jednostka: [piksel/s]
X1	NumberOfBullets	Maksymalna liczba pocisków gracza znajdująca się na planszy.
V2	CellVelocity	Szybkość przemieszczania się komórki od górnej do dolnej krawędzi. Jednostka [piksel/s]
H1	CellSize	Długość boku kwadratu komórek. Jednostka: [piksel]
P1	CellHealth	Wartość przypisana do każdej komórki symbolizująca liczbę kul potrzebną do unicestwienia mieszcząca się w przedziale od 1 do 9.
T2	CellRegeneartionInterval	Interwał czasowy pomiędzy kolejnymi wzrostami P1. Jednostka: [piksel]
T1	Interval	Interwał czasowy pomiędzy kolejnymi przyrostami. Jednostka: [s]
DV1	BulletVelocityIncrease	Przyrost szybkości wystrzeliwanego pocisku przez graczy dodawany co T1 sekund do V1. Jednostka: [piksel/s]
DV2	CellVelocityIncrease	Przyrost szybkości poruszania się komórki dodawany co T1 sekund do V2. Jednostka: [piksel/s]
DR1	BulletRadiusDecrease	Spadek długości promienia wystrzeliwanego pocisku odejmowany co T1 sekund od R1. Jednostka: [piksel]
DH1	CellSizeDecrease	Spadek długości boku kwadratu pojawiającej się komórki odejmowany co T1 sekund od H1. Jednostka: [piksel]
T3	GameTime	Czas trwania rozgrywki. Jednostka: [s]
IMG	ImageExtension	Format pliku graficznego tworzony przy zakończeniu gry. Jeśli nie jest podany żaden to plik nie jest tworzony.

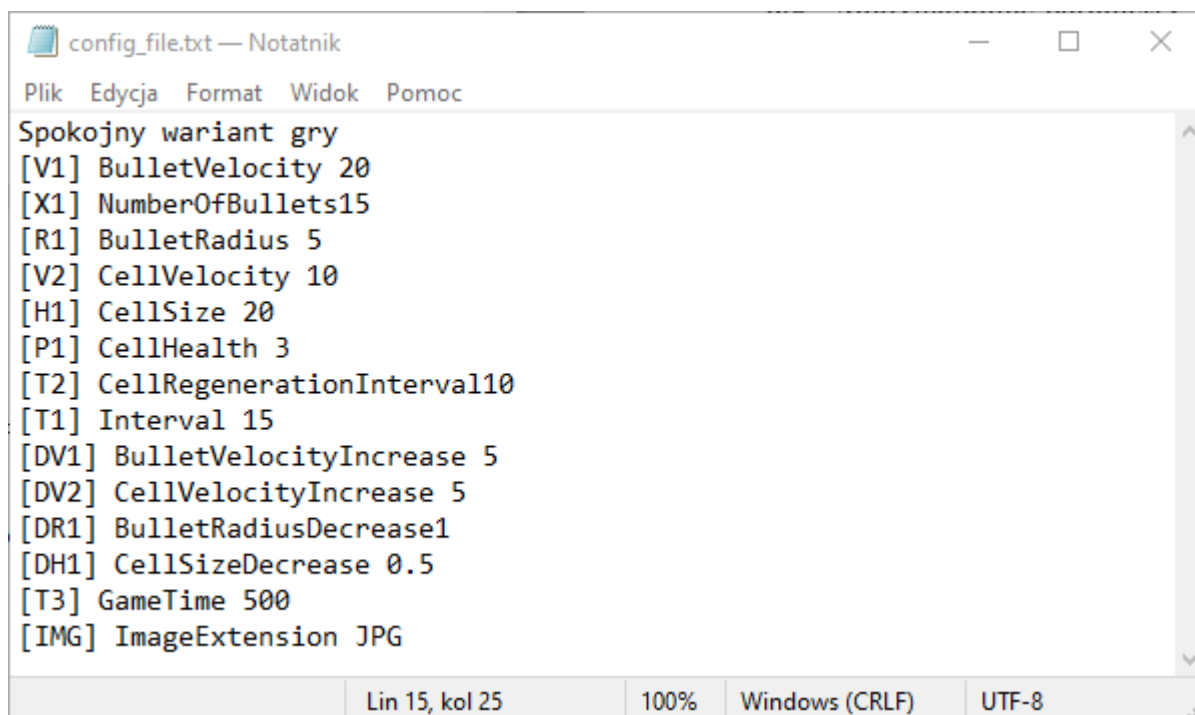
### 4.3 Struktura pliku konfiguracyjnego

Przedstawiona poniżej struktura pliku konfiguracyjnego przedstawia sposób prawidłowego przedstawienia elementów pliku. Miejsca ujęte w nawiasy trójkątne powinny zostać zastąpione oczekiwanymi wartościami zgodnie z wymaganiami przedstawionymi w nawiasach.

**Struktura:**

```
<Nazwa pliku konfiguracyjnego>
[V1] BulletVelocity <liczba rzeczywista większa od 0>
[X1] NumberOfBullets<liczba całkowita większa od 0>
[R1] BulletRadius <liczba rzeczywista większa od 0>
[V2] CellVelocity <liczba rzeczywista większa od 0>
[H1] CellSize <liczba rzeczywista większa od 0>
[P1] CellHealth <liczba całkowita większa od 0 i mniejsza od 9>
[T2] CellRegenerationInterval <liczba rzeczywista większa od 0>
[T1] Interval <liczba rzeczywista większa od 0>
[DV1] BulletVelocityIncrease <liczba rzeczywista większa od 0>
[DV2] CellVelocityIncrease <liczba rzeczywista większa od 0>
[DR1] BulletRadiusDecrease<liczba rzeczywista większa od 0>
[DH1] CellSizeDecrease <liczba rzeczywista większa od 0>
[T3] GameTime <liczba całkowita większa od 0>
[IMG] ImageExtension <PNG/ JPG /JPEG/ BMP>
```

### 4.4 Przykładowy plik konfiguracyjny



Rysunek 7: Przykładowy plik konfiguracyjny

## 5 Przebieg działania programu

### 5.1 Scenariusz powszechny

W domyślnym scenariuszu działania programu, użytkownik uruchamia rozgrywkę od razu lub najwypierw definiuje własne ustawienia zgodnie z instrukcją zawartą w podpunktach „Funkcjonalności programu” oraz „Plik konfiguracyjny”. Następnie, po zakończeniu rozgrywki zostaje zapisany zrzut ekranu. Po przeprowadzeniu jednej gry można zamknąć aplikację, od razu uruchomić kolejną rozgrywkę z identycznymi ustawieniami lub powrócić do menu głównego w celu ich modyfikacji przed kolejną rozgrywką.

### 5.2 Scenariusze ewentualne

#### Nieprawidłowy plik konfiguracyjny

Program wczytuje cały plik konfiguracyjny linia po linii sprawdzając poprawność każdego wiersza pod kątem formatu. W przypadku znalezienia jakichkolwiek nieprawidłowości zostanie wyświetlony komunikat informujący użytkownika o nieprawym pliku konfiguracyjnym z informacjami pozwalającymi na zdiagnozowanie problemu i zapytanie, czy kontynuować program z ustawieniami domyślnymi. Odpowiedź „Tak” pozwala na przeprowadzenie rozgrywki z ustawieniami domyślnymi. Odpowiedź „Nie” kończy działanie programu.

#### Brak pliku konfiguracyjnego

Program w przypadku braku pliku konfiguracyjnego wyświetla ostrzeżenie o braku pliku konfiguracyjnego, niezaburzające działania aplikacji. Dalszy przebieg programu odbywa się na podstawie domyślnych ustawień.

#### Nieprawidłowe działanie programu

W działaniu aplikacji „Tanks” mogą wystąpić błędy związane z nieprawidłowym przygotowaniem pliku konfiguracyjnego, poza tym nie przewiduje się żadnych błędów ponad standardowe wyjątki zgłaszane przez środowisko Javy.

#### Komunikowanie błędów

- **FILE\_INCOMPLETE** [nazwa skrócona danej] – oznacza wybrakowanie informacji w pliku konfiguracyjnym (któregoś z parametrów brakuje). Przykładowy kod tego błędu: FILE\_INCOMPLETE [T3] – oznacza, że w pliku brakuje informacji o czasie rozgrywki.
- **INVALID\_DATA** [nazwa skrócona danej] – oznacza wystąpienie nieprawidłowego typu lub wartości danej, odnoszącego się do podanego przez kod błędu parametru. Przykładowy kod tego błędu: INVALID\_DATA [DV1] – oznacza, że w pliku wystąpił niewłaściwy typ parametru opisującego przyrost prędkości pocisków, tj. niebędący liczbą rzeczywistą bądź będący liczbą rzeczywistą wykraczającą poza dozwolony zakres.