

Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI I TECHNIK INFORMACYJNYCH

*Nazwa przedmiotu: Programowanie strukturalne*

*Kod przedmiotu: 103A-ELxx-ISP-PROO*

---

DOKUMENTACJA

*Autorzy:*

*Robert Zawadzki 325102*

*Karol Franczuk 325001*

*Prowadzący:*

*dr inż. Dominik Kasprowicz*

2 czerwca 2024



WYDZIAŁ ELEKTRONIKI  
I TECHNIK INFORMACYJNYCH

Oświadczamy, że niniejsza praca, stanowiąca podstawę do uznania osiągnięcia efektów uczenia się z przedmiotu Programowanie strukturalne została wykonana przez nas samodzielnie.

Robert Zawadzki 325102 Karol Franczuk 325001

## Spis treści

1	Krótki opis Projektu	2
2	Sposób działania	2
3	Potrzebne pliki	2
4	Krótki opis przeznaczenia poszczególnych klas	2
5	Wymagane elementy projektu	3
6	Procedura uruchomienia	3
7	Diagram UML	4

# 1 Krótki opis Projektu

Projekt jest prostą aplikacją komputerową w Javie, która umożliwia użytkownikowi interakcję z graficzną planszą do gry. Użytkownik może wybierać spośród różnych trybów gry, w tym trybu standardowego oraz trybu rosnącej planszy, w którym plansza powiększa się po wypełnieniu.

Aplikacja ma na celu zapewnienie rozrywki poprzez prostą grę planszową, w której użytkownik klika na pola planszy, zmieniając ich stan. Gra zawiera różne tryby gry, które zwiększają poziom trudności oraz umożliwiają użytkownikowi zapisywanie i przeglądanie wyników.

# 2 Sposób działania

1. Po uruchomieniu aplikacji użytkownik widzi główne okno z pięcioma przyciskami:

- "Zagraj" - rozpoczyna standardową grę na domyślnej planszy.
- "Zwiększająca się plansza" - uruchamia tryb gry, w którym plansza powiększa się po wypełnieniu.
- "Ustaw rozmiar planszy" - pozwala użytkownikowi ustawić rozmiar planszy.
- "Wyświetl wyniki" - wyświetla zapisane wyniki gier.
- "Wyjście z gry" - zamyka aplikację.

2. W trybie standardowym użytkownik klika na pola planszy, zmieniając ich kolor i stan. Celem jest wypełnienie całej planszy.

3. W trybie rosnącej planszy plansza powiększa się po każdym pełnym wypełnieniu, aż do maksymalnego rozmiaru 9x9.

4. Wyniki gier są zapisywane i mogą być wyświetlane użytkownikowi.

# 3 Potrzebne pliki

Pliki źródłowe Java: Main.java, MainFrame.java, TrybStandardowy.java, TrybRosnacejPlanszy.java, ZarzadzaniePlansza.java, TablicaWynikow.java, Projekt.java, Plansza.java. Plik wyników: wyniki.txt (zapis i odczyt wyników gry).

# 4 Krótki opis przeznaczenia poszczególnych klas

Main: Klasa główna uruchamiająca aplikację.

MainFrame: Klasa głównego okna aplikacji zawierająca przyciski do wyboru trybu gry.

Projekt: Klasa zarządzająca stanem gry, w tym planszą i wynikami.

Plansza: Klasa reprezentująca planszę do gry.

TrybStandardowy: Klasa obsługująca standardowy tryb gry.

TrybRosnacejPlanszy: Klasa obsługująca tryb rosnącej planszy, dziedziczy po TrybStandardowy.

ZarzadzaniePlansza: Klasa zawierająca metodę wypełniającą planszę zgodnie z logiką gry.

TablicaWynikow: Klasa zarządzająca wynikami gier.

Wynik: Klasa reprezentująca pojedynczy wynik gry.

## 5 Wymagane elementy projektu

Kolekcje: Mapa (HashMap) w klasie TablicaWynikow do przechowywania wyników.

Dziedziczenie: TrybRosnacejPlanszy dziedziczy po TrybStandardowy.

GUI: Graficzny interfejs użytkownika w osobnym wątku (klasa MainFrame).

Strumienie: Zapis i odczyt wyników do/z pliku wyniki.txt w klasie TablicaWynikow.

Wielowątkowość: GUI uruchamiane w osobnym wątku (SwingUtilities.invokeLater).

Wyjątki: Obsługa wyjątków podczas wczytywania i zapisywania wyników (klasa TablicaWynikow).

Używanie wyrażeń lambda tam, gdzie to ma sens.

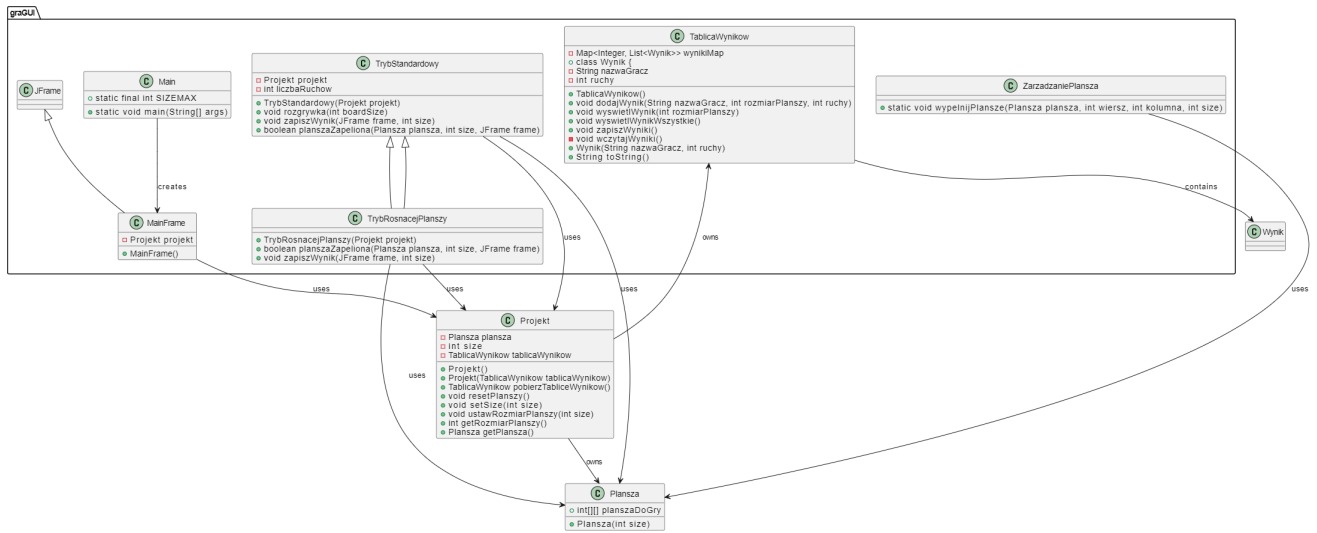
Obsługa wyjątków w sposób czytelny dla użytkownika (okna dialogowe).

## 6 Procedura uruchomienia

Uruchomić plik Main.java jako główną klasę projektu. Za pomocą komend:

```
- javac -d ./graGUI/* *.java  
- java graGUI.Main
```

## 7 Diagram UML



Rysunek 1: Diagram UML GUI java