

# Projekt I

## Zasady

1. Termin obrony projektów: **20.05.2018 r.**
2. Zabrania się wykorzystywania narzędzi WYSIWYG do generowania okien (tzw. Window Builderów).
3. Brak znajomości dowolnego fragmentu kodu lub plagiat skutkować będzie wyzerowaniem punktacji za ten projekt.
4. W ocenie projektu poza praktyczną i merytoryczną poprawnością będzie brana również pod uwagę optymalność, jakość i czytelność napisanego przez Państwa kodu.
5. Ważną częścią projektu jest wykorzystanie między innymi: dziedziczenia, kolekcji, interfejsów lub klas abstrakcyjnych, lambda-wyrażeń, typów generycznych, wątków, dodatkowych funkcjonalności lub struktur oraz innych elementów charakterystycznych (ale tylko w naturalny sposób, nic na siłę)

## Zadanie

Napisz program będący grą w stylu "memory". Gra polega na odnajdywaniu par takich samych kart. Gracz odsłania 2 karty. Jeśli są to takie same karty, zostają one widoczne na planszy, jeśli nie, karty będą zakrywane z powrotem (po **2 sekundach**). Celem gracza jest zdjęcie wszystkich kart przy możliwie najszybszym czasie. Program ten po uruchomieniu powinien wyświetlać menu główne składające się z opcji:

- **new game** - nowa gra
- **high scores** - tabela wyników
- **exit** - wyjście

Po uruchomieniu nowej gry, gracz zostanie zapytany w osobnym oknie jakiej wielkości pole chce rozwiązać. Należy zauważyć, że sumaryczna ilość kart musi być parzysta, aby każda karta miała swoją parę (na przykład w siatce 3x3 znajduje się 9 kart, więc po odkryciu 4 par jedna zostanie bez).

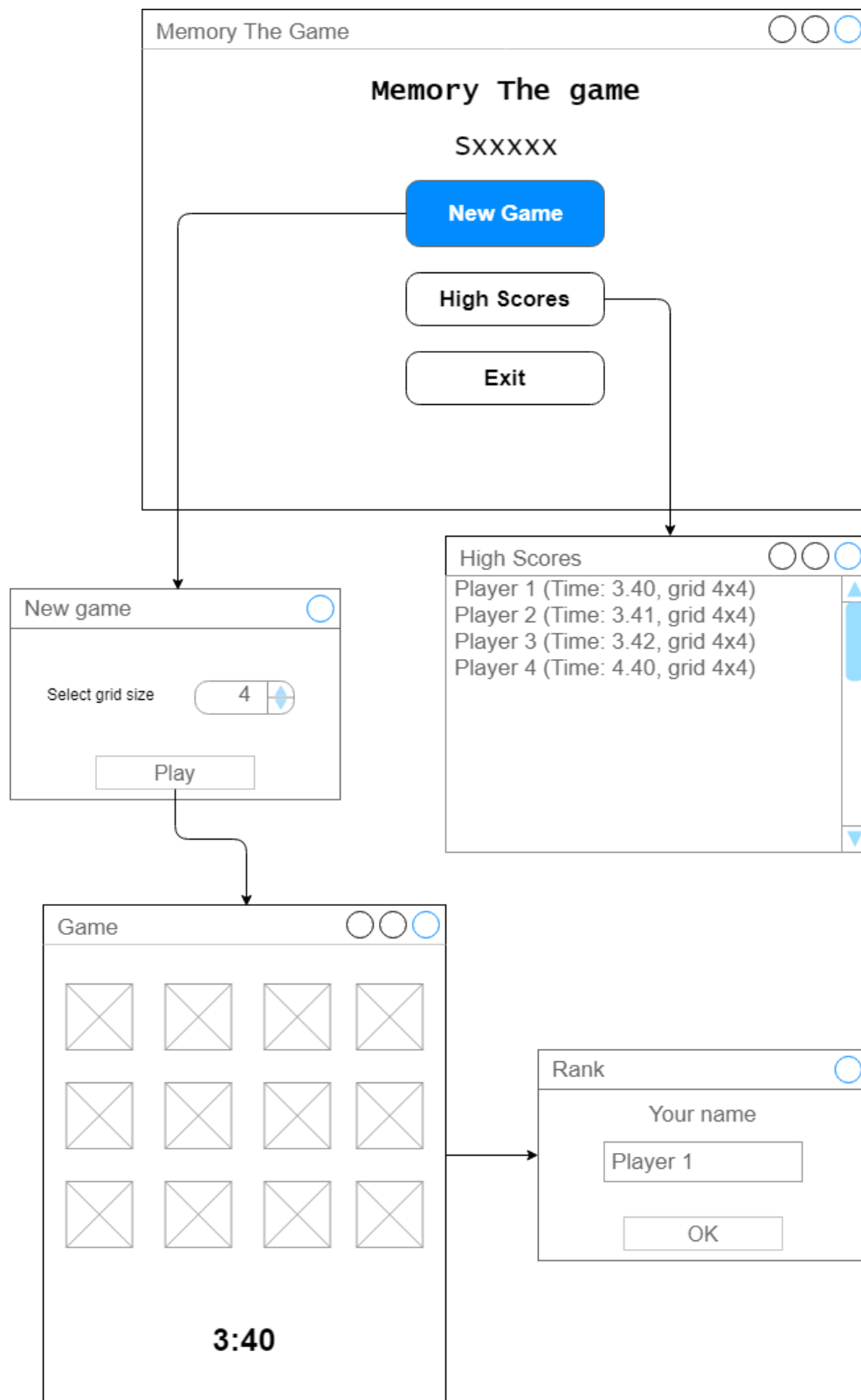
Po uruchomieniu gry w nowym oknie wyświetlane są zakryte karty, a licznik czasu rusza (warto zauważyć, że licznik powinien być realizowany w osobnym wątku, żeby nie blokować interakcji z oknem). Gra toczy się według wyżej wymienionych zasad do momentu odkrycia wszystkich kart.

Po zakończeniu gry, w nowym oknie gracz proszony jest o swoją nazwę pod jaką ma być zapisywany w rankingu.

Ranking liczony jest na podstawie czasu i wielkości planszy (np: punkty = wielkość planszy/czas gry). Należy zapewnić trwałość rankingu po ponownym uruchomieniu aplikacji, czyli należy go przechowywać w pliku na dysku. Postać przechowywanych danych jest nieistotna i nie musi być czytelna dla gracza (można wykorzystać np. interfejs serializable).

Po wybraniu opcji rankingu z menu głównego powinien zostać on wyświetlony użytkownikowi. Należy zauważyć, że może być on relatywnie duży, dlatego należy zadbać o paski przewijania, w razie gdyby nie mieścił się on w oknie.

Przykładowa wizualizacja gry:



Wskazówki:

Należy zadbać o wyjątki w programie. Jeśli jakiś wystąpi należy wyświetlić jego komunikat użytkownikowi.

Ranking można realizować przy pomocy komponentu JList, ale nie trzeba. Może to być zwykły tekst, w którym każdy wpis będzie w osobnej linii.

Karty można realizować przy pomocy przycisków, ale można też zaprojektować własny komponent. Na kartach nie trzeba umieszczać obrazków, jeśli będą to litery alfabetu lub symbole unicode też będzie dobrze.

Nie wszystkie okna muszą być realizowane poprzez klasę JFrame, przy mniejszych oknach można wykorzystać okna dialogowe.