

Dokumentacja inżynierii wymagań

Temat: Gra w oczko

Piotr Węgrzyn, Marek Kwak, Tymoteusz Dudek

1. Macierz kompetencji zespołu

Kompetencje	Węgrzyn	Kwak	Dudek
Programowanie Python	+++	+++	+
Programowanie Java	+++	++	+++
Programowanie C/C++	++	+	+
Praca z bazami danych/SQL	++	+++	+++
Znajomość UML	++	+	-
Programowanie języki WEB'owe	-	+++	-
Testowanie oprogramowania	+	-	-
Komunikatywność	++	+	+
Zaangażowanie	-	++	+++

2. Opis ogólny projektu

Tematem projektu jest [gra w oczko](#).

Na początku gry rozdane zostaną po dwie karty na każdego z graczy. W początkowej chwili każdy z graczy jest *graczem aktywnym*, to znaczy takim, który ma decyzję co do podjęcia dalszego działania. Gra polega na kolejnym dobieraniu kart przez kolejnych aktywnych graczy aż do momentu, w którym każdy z graczy uzna, że ma najlepszą sytuację na wygraną (każdy

z graczy staje się *graczem pasywnym*). Na zakończenie porównywane są wyniki i zostaje wybrany zwycięzca/zwycięzcy.

Klient życzy sobie, aby gra była możliwa z komputerem oraz innymi graczami (2-4 osób), była w formie graficznej, każdy z graczy miał swoje konto z rankingami i statystykami oraz aby takie statystyki pojawiały się po każdej rozgrywce (aby pokazywały rankingi graczy oraz statystyki kart - częstotliwości wystąpień etc.).

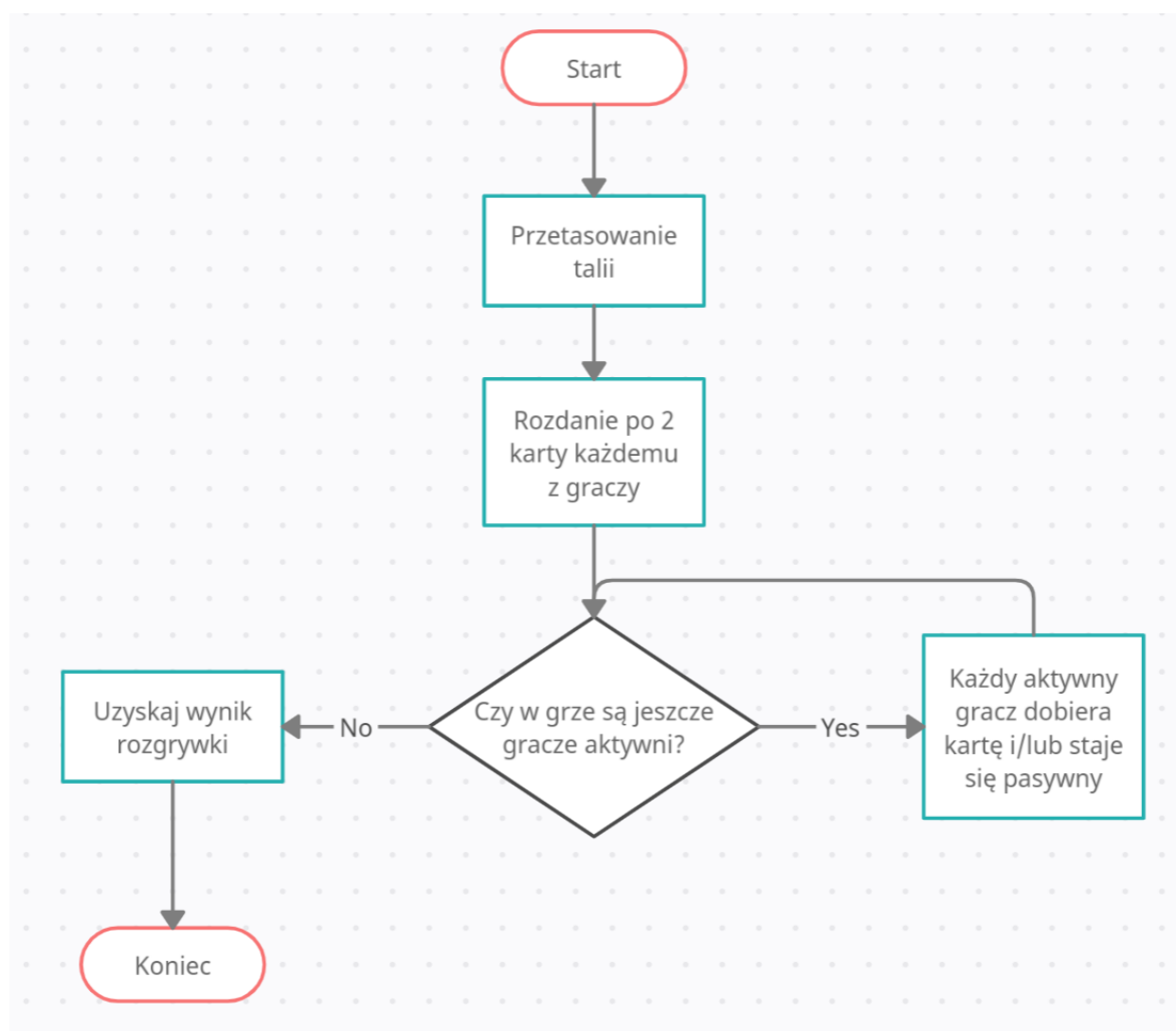
Klient jest zainteresowany kilkoma poziomami trudności oraz animacją pokazującą przebieg rozgrywki już po zakończeniu gry (animated replay).

3. Dodatkowe pytania odnośnie projektu:

Pytanie	Odpowiedź	Uwagi
Na jaką platformę ma zostać napisana gra?	Desktop lub WEB	WIN / LINUX Gra powinna działać na przynajmniej 2ch przeglądarkach
Czy gra ma posiadać interfejs graficzny?	Tak	Wystarczy prosty interfejs
Czy dołączyć instrukcję do gry?	Tak	Możliwość otworzenia instrukcji z menu
Czy gra ma być w wersji online?	Rozgrywka lokalna (Hot Seat)	W przypadku zmiany osoby wykonującej ruch dać czas na zmianę miejsca
Dla ilu graczy ma być dostępna rozgrywka?	Rozgrywka dla 2-4 graczy	
Czy użytkownik aplikacji powinien mieć swoje konto?	Tak	Użytkownik posiada nazwę użytkownika i hasło (z możliwością zmiany hasła)
Czy należy zbierać statystyki dla gracza?	Tak, dodatkowo ranking graczy	Statystyki powinny zawierać: Procent wygranych; Liczba rozegranych gier; Liczba użytych kart; Czas gry;
Zbierać statystykę częstotliwości pojawiających	Tak	

się kart?		
Z ilu talii kart ma się składać rozgrywka?	1-3 talie	Od '2' do 'As' - wartości z wikipedii
Ile wyglązków kart ma posiadać gra?	Przynajmniej dwa	Do wyboru z menu opcji
Jakie tryby rozgrywki mają być dostępne?	Gracz - Gracz Gracz - Komputer Komputer - Komputer	Dla różnych poziomów AI również zbieramy statystyki
Czy zapisywać powtórkę gry po jej zakończeniu?	Tak	Zapisywać wszystkie powtórki ze wszystkich trybów
Co ile symulować rozgrywkę Komputer - Komputer?	Ruch co jedną sekundę	
Czy dodać poziomy trudności dla gracza?	Tak, trzy poziomy trudności	W zależności od poziomu trudności zmienić jedynie czas na wykonanie ruchu (15/10/5 s)
Jakie mają być poziomy zaawansowania komputera?	Easy; Medium; Hard	Easy - Dobiera karty losowo; Medium - Posługuje się logicznymi decyzjami; Hard - Podgląda karty
Dodać możliwość grania 'na pieniądze'?	Gracz może wejść obstawiając	Ustalona kwota minimalna - X

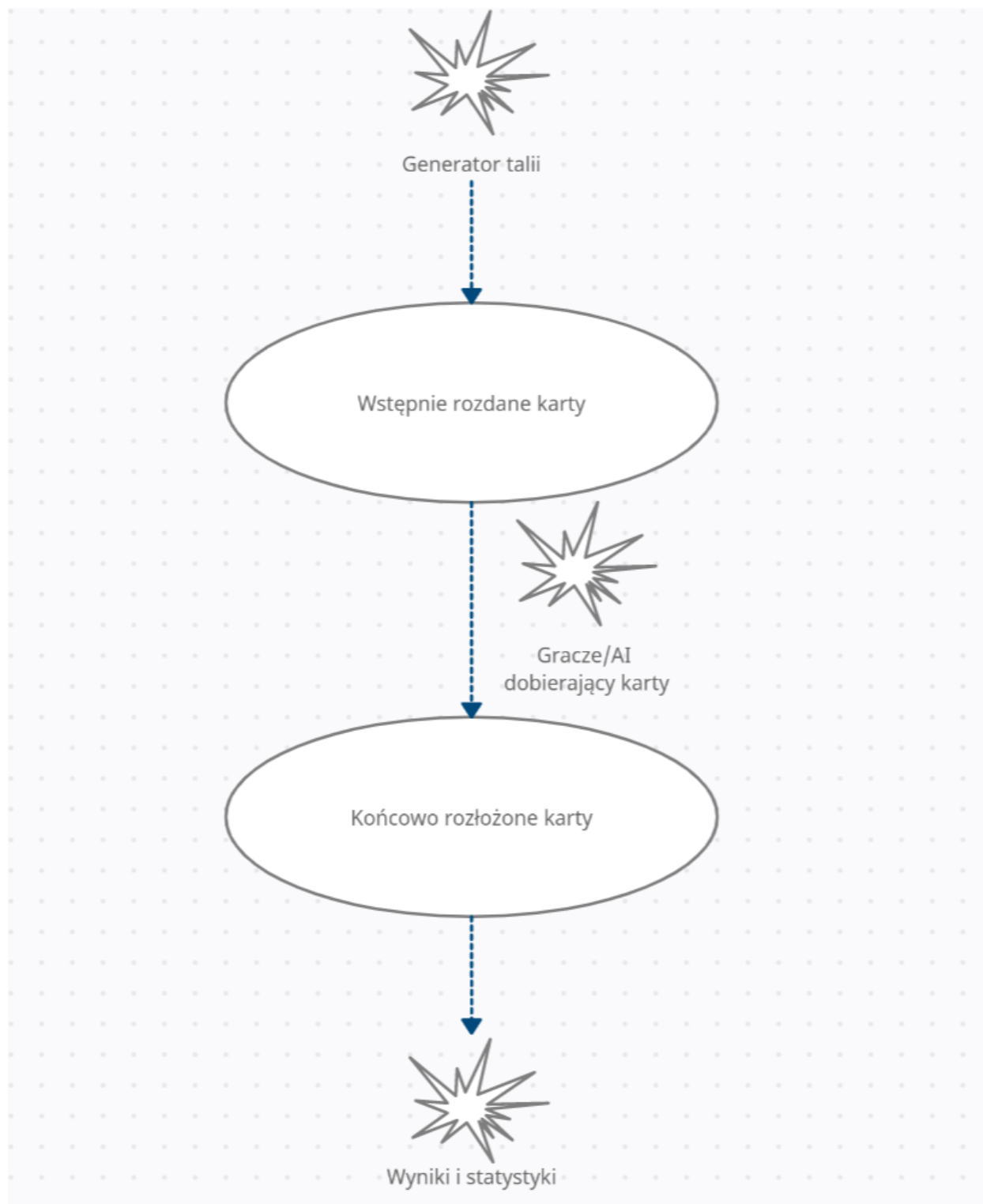
4. Algorytm rozgrywki jako schemat blokowy



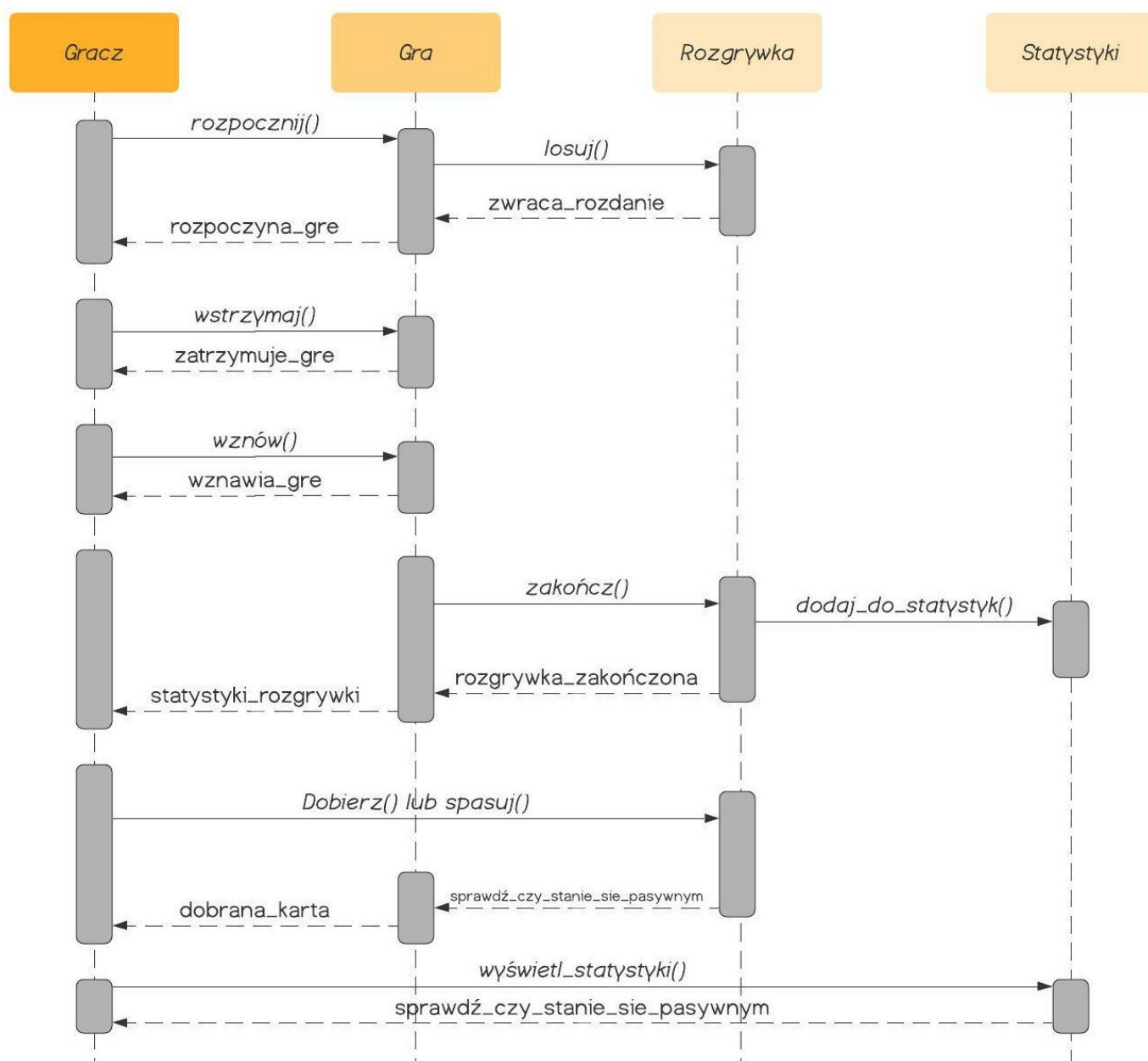
5. Algorytm rozgrywki zamodelowany tabelą

Aktorzy	Generator talii, gracze/AI, Analizator rozgrywki
Opis	Celem gry jest osiągnięcie najwyższej sumy kart przez każdego z graczy, lecz nieprzekraczającej 21 (wyjątkiem jest <i>Perskie oczko</i> , czyli dwa asy, które wygrywa grę). W każdej turze gracze aktywni kolejno dobierają karty i/lub przechodzą w stan pasywny i czekają na odkrycie kart i rozstrzygnięcie.
Dane	Rozłożenie kart
Wyzwalacz	Po przejściu wszystkich graczy w stan pasywny porównujemy wyniki graczy
Odpowiedź	Raport: wyniki rozgrywki, statystyki generowanych kart oraz animacja rozegranej rozgrywki
Uwagi	Gracze z sumą przekraczającą 21 automatycznie przechodzą w stan pasywny i ostatecznie przegrywają

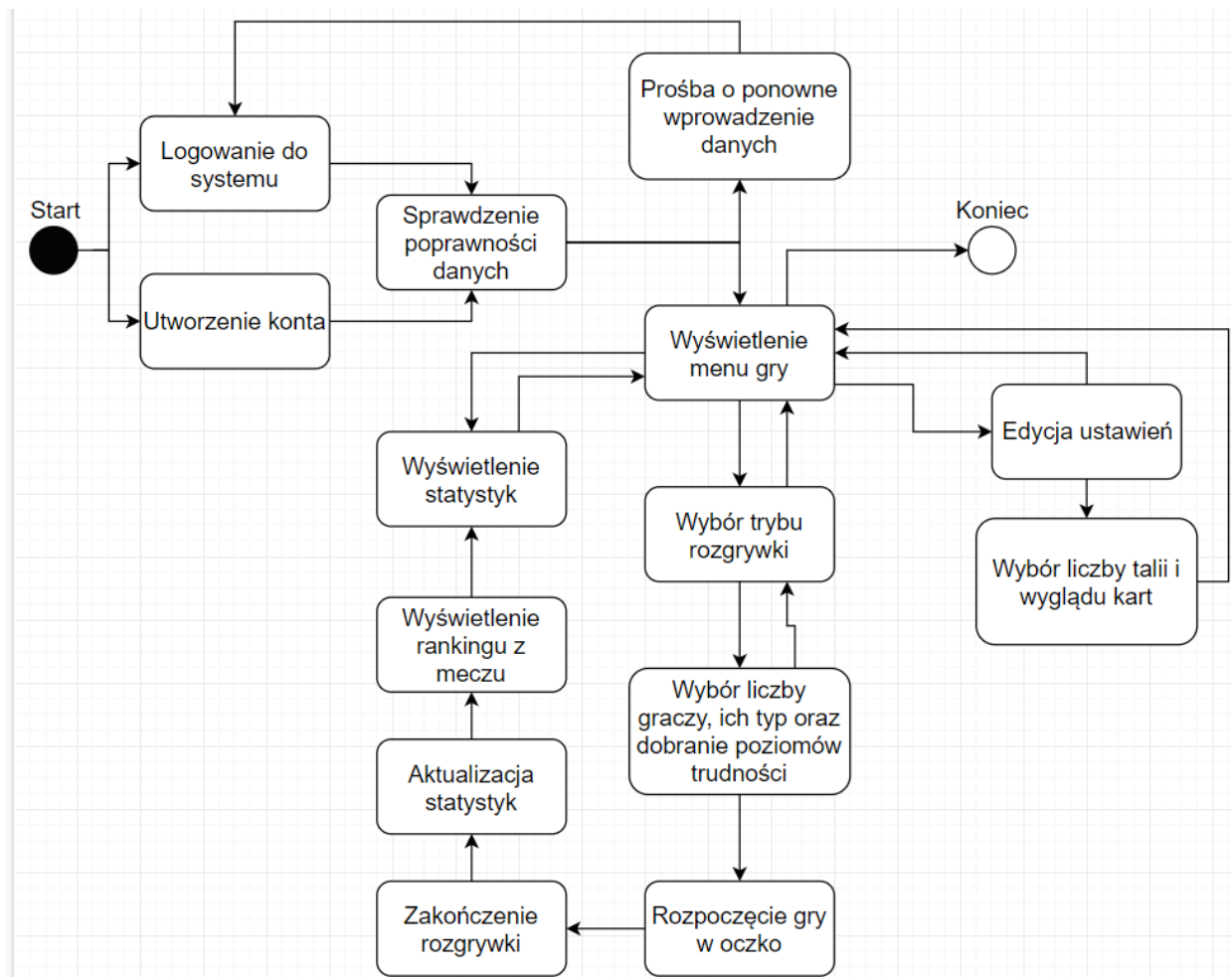
6. Algorytm rozgrywki zamodelowany diagramami UML



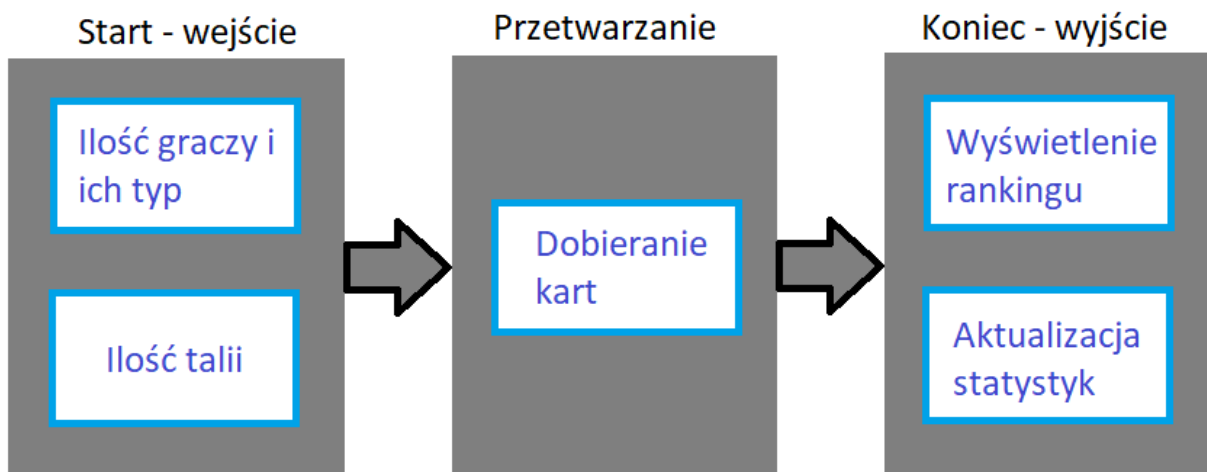
7. Diagram sekwencyjny UML



8. Diagram Aktywności



9. Projekt architektury opracowywanego systemu



Zadaniem systemu jest przyjąć dane od użytkownika mówiące o ilości talii oraz liczbie i typie graczy potrzebne do stworzenia odpowiednich warunków do działania, aby następnie umożliwić rozgrywkę. Po jej zakończeniu system generuje raport w celu podsumowania

rozgrywki. Pierwszym elementem systemu jest odpowiednie wygenerowanie uczestników gry oraz zbioru kart, które mają umożliwić rozpoczęcie gry. Drugim elementem jest przeprowadzenie rozgrywki zgodnie z ustalonymi zasadami. Wymienione elementy są podstawą do wygenerowania raportu, który ma na celu zapewnić poprawność działania programu.