The Forest – dokumentacja

# Ogólny opis projektu

Gra The Forest to prosta gra symulująca życie rozbitka, który w celu przetrwania musi wykonywać różne czynności. Celem gry jest pokonanie finałowego potwora. Aby być do tego zdolnym, należy zdobyć odpowiedni ekwipunek - broń i zbroję oraz schronienie.

# Interfejs użytkownika



## Silnik graficzny

Interfejs graficzny został zaprojektowany w oparciu o silnik graficzny WPF. Okno gry posiada możliwość skalowania (minimalne wymiary okna to 600 x 1067 px).

## Komunikaty

W centralnej części ekranu znajduje się pole z komunikatami tekstowymi. Znajdują się w nim opisy wszystkich wykonywanych na bieżąco akcji.

## Ekwipunek

W lewej części ekranu znajduje się ekwipunek gracza. Składają się na niego jedzenie oraz 4 podstawowe surowce: drewno, kamień, skóra i metal. Są one potrzebne do konstruowania przedmiotów, które można zakupić w sklepie. Poniżej znajduję się pole, w którym wyświetlona jest aktualnie posiadana broń i zbroja.

## Statystyki

Powyżej ekwipunku na żółtym polu znajdują się statystyki gracza. Pokazana jest ilość posiadanych pieniędzy, aktualna godzina, dzień gry oraz punkty doświadczenia. Czas gry jest czasem wirtualnym i jest on przyspieszony 200 razy.

## Zapis gry i pauzowanie

Poniżej ekwipunku znajduje pole, w którym możemy zapisać lub wczytać zapis gry. Jest też przycisk pauza, który umożliwia zapauzowanie oraz wznowienie gry. Wówczas zostanie wstrzymany czas i wszystkie aktywności zostaną zablokowane.

## Sklep

W górnej części ekranu znajduje się sklep. Wybierając odpowiednią rzecz z listy, możemy ją kupić lub sprzedać. Należy jednak zwrócić uwagę, żeby mieć odpowiednią ilość pieniędzy i surowców. Informacja o tym znajduje się tuż obok nazwy produktu. Oprócz tego przy broniach zbrojach i schronieniach znajdują się ich odpowiednio atak, obrona i wytrzymałość.

## Życie i energia

Na górze pośrodku znajduje się pole z informacją o punktach energii i życia. Należy zwracać szczególną uwagę na punkty życia, ponieważ gdy osiągniemy 0 punktów, gra się zakończy i trzeba będzie zacząć grać od nowa.

## Obrona i atak

Na prawo od życia i energii znajduje się wskaźnik punktów ataku i obrony. Mają one wpływ na łupy z polowania lub zbierania surowców.

## Las

W prawej części ekranu znajdują się przyciski, które umożliwiają interakcje z lasem. Możemy udać się na polowanie, gdzie zdobędziemy jedzenie. Wówczas jednak stracimy kilka punktów energii i życia - zależnie od tego, z jaką bronią i zbroją się wybierzemy. Oprócz tego możemy zbierać surowce. Wtedy tracimy tylko energię. Możemy również spróbować zawalczyć z finałowym potworem, aby ukończyć grę.

## Jezioro

Jezioro to miejsce rekreacji, możemy tutaj odpocząć i naładować punkty energii albo udać się na wędkowanie i zyskać jedzenie.

## Schronienie

W schronieniu można zjeść jedzenie albo udać się spać. Warto dbać o rozbudowanie schronienia, aby być przygotowanym na niespodziewany atak nocnych potworów.

## Czy zapisać stan gry?

Przy zamykaniu okna wyświetla się okno z zapytaniem o zapisanie stanu gry.

# Mechanika gry

## Początek gry

Na start gracz posiada 3 sztuki jedzenia i 10 monet. Jest w pełni wypoczęty i ma 100 punktów życia.

## Doświadczenie

W czasie gry warto dbać o zróżnicowane aktywności takie jak zbieranie surowców, polowania lub łowienie. Dodają one punkty doświadczenia, co wpływa na efektywność naszych akcji.

## Czas gry

Warto zwrócić uwagę na biegnący czas. Każda aktywność w grze zajmuje pewien czas. Warto zerkać na zegar w porach wieczornych i wykorzystać noc do spania.

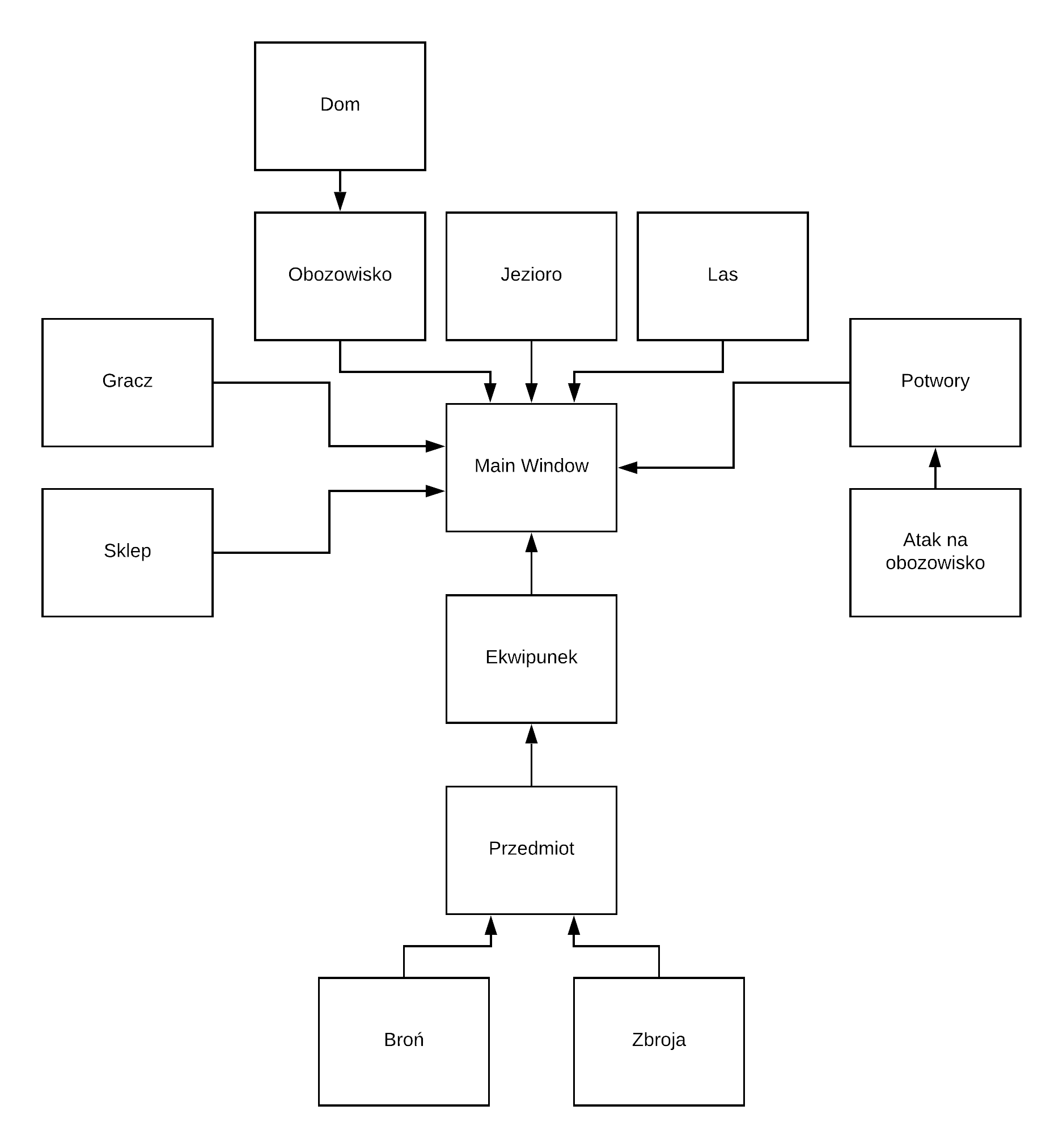
## Atak na obozowisko

W czasie nieprzespanej nocy, hałas zdradza potworom naszą lokalizację. Mogą się one pokusić na zaatakowanie naszego obozowiska. Dlatego warto ulepszać nasze schronienie, co zmniejszy szansę na zaatakowanie naszego domostwa.

## Finał gry

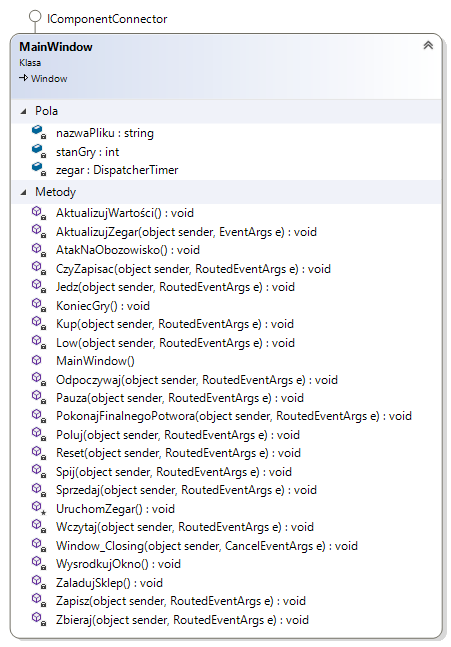
W celu ukończenia gry należy rozprawić się z finałowym potworem. Aby tego dokonać, wymagane są najlepsza broń i zbroja oraz najlepsze zbudowane schronienie. Ponadto trzeba mieć maksymalną ilość zdrowia i energii.

# Uproszczony diagram klas



# Opis klas

## MainWindow



Klasa jest klasą główną programu. Jest odpowiedzialna za inicjalizacje komponentów gry oraz reagowanie na wszystkie interakcje z użytkownikiem. Obsługuje również liczenie czasu w grze oraz zdarzenia losowe związane z porą dnia.

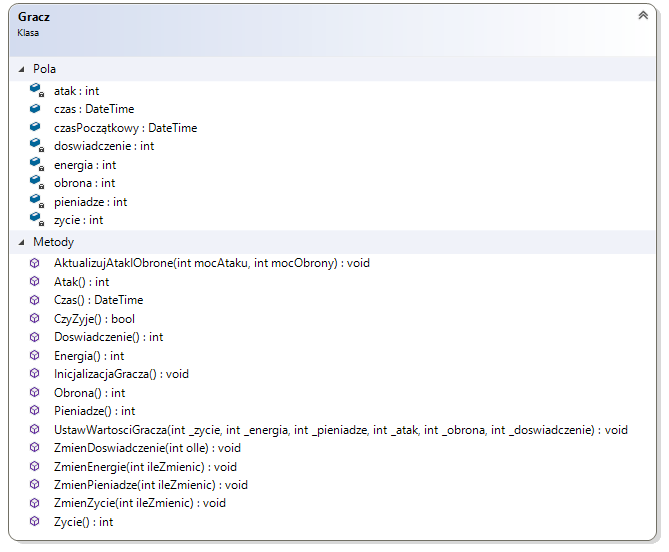
Pola klasy:

* nazwaPliku – przechowuje nazwę pliku, w którym jest zapisywany stan gry
* stanGry – przechowuje informacje o tym czy gra trwa, jest w stanie pauzy lub się zakończyła
* zegar – zmienna obsługująca bieg czasu w grze

Metody klasy:

* MainWindow – konstruktor klasy, inicjalizuje interfejs i pozostałe komponenty programu
* Kup – na podstawie wybranego elementu na w ListBoxie *sklep* wywołuje funkcję Sklep.Kup() i wyświetla komunikat tekstowy o zakupie. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* Sprzedaj - na podstawie wybranego elementu na w ListBoxie *sklep* wywołuje funkcję Sklep.Sprzedaj() i wyświetla komunikat tekstowy o sprzedaży. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* Poluj – wywołuje funkcję Las.Poluj() i wyświetla komunikat o polowaniu. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* Zbieraj – wywołuje funkcję Las.Zbieraj() i wyświetla komunikat o zebranych surowcach. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* Odpoczywaj – wywołuje funkcję Jezioro.Odpoczywaj() i wyświetla komunikat o zyskanej energii. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* Low – wywołuje funkcję Jezioro.Low() i wyświetla komunikat o złowionych rybach. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* Jedz – wywołuje funkcję Obozowisko.Jedz() i wyświetla komunikat o zyskanym zdrowiu i energii. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* Spij – wywołuje funkcję Jezioro.Low() i wyświetla komunikat o złowionych rybach. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* PokonajFinalnegoPotwora – symuluje pokonywanie potwora. Sprawdza czy gracz ma wystarczający ekwipunek do pokonania potwora i wyświetla komunikat o pokonaniu lub porażce. Następnie wywołuje funkcję AktualizujWartości(), która aktualizuje wszystkie pola tekstowe.
* AktualizujWartości – aktualizuje wartości we wszystkich polach tekstowych oraz sprawdzająca czy gracz żyje
* KoniecGry – wywoływana gry gracz umiera – wyświetla komunikat o końcu gry, blokuje interakcję z przyciskami i wyświetla pole końca gry
* ZaladujSklep – dodaje wszystkie elementu podlegające kupnie lub sprzedaży do ListBox *sklep*
* WysrodkujOkno – wywoływana podczas inicjalizacji gry – ustawia okno na środku ekranu w przypadku, gdy gra nie jest uruchamiana w trybie pełnoekranowym
* UruchomZegar – inicjalizuje działanie zegara liczącego czas w grze
* AktualizujZegar – wywoływana z każdym cyklem zegara, co 300 ms – w odpowiednich godzinach wywołuje funkcję AtakNaObozowisko(). Aktualizuje pola tekstowe godziny. Jeżeli gracz żyje – dodaje minutę do czasu gry.
* Zapisz – tworzy plik, w którym zapisuje wszystkie zmienne związane z aktualnym stanem gry i wyświetla komunikat o pomyślnym zapianiu.
* Wczytaj – wczytuje z pliku stan gry który został poprzednio zapisany i wyświetla komunikat o wczytaniu.
* Pauza – wprowadza lub wyprowadza grę ze stanu pauzy, który polega na zablokowaniu przycisków i zatrzymaniu czasu.
* Reset – resetuje stan gry, jest używana, gdy gracz zginie.
* CzyZapisac – otwiera okno i pyta użytkownika czy chce zapisać stan gry przy zamykaniu okna gry.
* AtakNaObozowisko - wywołuje funkcję ataku na obozowisko i wyświetla komunikat.
* PokonajFinalnegoPotwora – umożliwia przeprowadzenie ataku na końcowego potwora i na podstawie analizy ekwipunku wyświetla odpowiedni komunikat.

## Gracz

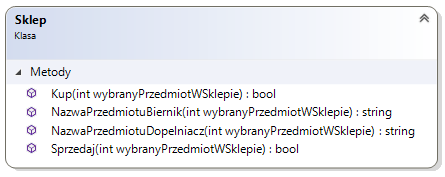


Klasa reprezentuje postać gracza i obsługuje jego parametry i zachowania.

Pola klasy:

* atak – moc ataku na podstawie posiadanej broni
* obrona – moc obrony na podstawie posiadanej zbroi
* energia – poziom energii gracza
* zycie – poziom życia gracza
* pieniądze – liczba pieniędzy gracza
* doświadczenie – liczba punktów doświadczenia gracza
* czas – aktualny czas w grze
* czasPoczątkowy – czas startu gry
* Metody klasy:
* InicjalizacjaGracza– ustawia wartości początkowe gracza
* UstawWartosciGracza – ustawia przekazane wartości gracza
* Zycie – zwraca życie gracza
* Enegia – zwraca energię gracza
* Pieniadze – zwraca liczbę pieniędzy gracza
* Atak – zwraca atak gracza
* Obrona – zwraca obronę gracza
* Doswiadczenie – zwraca doświadczenie gracza
* ZmienPieniadze – zmienia pieniądze o daną wartość
* ZmienZycie – zmienia zycie o daną wartość
* ZmienEnergie - zmienia energię o daną wartość
* AktualizujObroneIAtak – zmienia moc ataku o moc obrony o przekazaną wartość
* CzyZyje – zwraca wartość logiczną zależnie czy gracz żyje czy nie
* ZmienDoswiadczenie – zmienia doświadczenie gracza o daną wartość

## Sklep

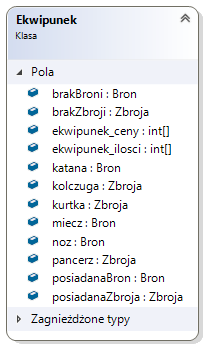


Klasa obsługuję zakupy i sprzedaż.

Metody klasy:

* Kup – dokonuje zakupu wskazanego przedmiotu, jeżeli gracz ma fundusze i zwraca informacje o udanym lub nieudanym zakupie
* Sprzedaj – dokonuje sprzedaż wskazanego przedmiotu, jeżeli gracz posiada go i zwraca informacje o udanej lub nieudanej sprzedaży
* NazwaPrzedmiotuBiernik – zwraca nazwę danego przedmiotu w bierniku
* NazwaPrzedmiotuDopelniacz – zwraca nazwę danego przedmiotu w dopełniaczu

## Ekwipunek



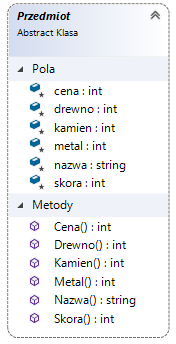
Klasa przechowuje informacje o ekwipunku gracza

Pola klasy:

* ekwipunek\_ceny – przechowuje ceny przedmiotów
* ekwipunek\_ilosci – przechowuje informacje o ilości przedmiotów w ekwipunku
* posadanaBron i posiadanaZbroja – przechowuje informacje o posiadanej broni i zbroi

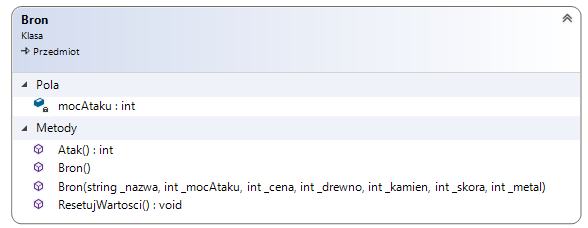
Pozostałe pola to obiekty przechowujące informacje o danych rodzajach broni lub zbroi

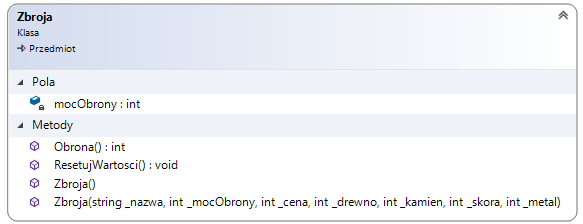
## Przedmiot



Klasa abstrakcyjna zawierająca pola informujące o cechach danego przedmiotu i metodach zwracających te wartości.

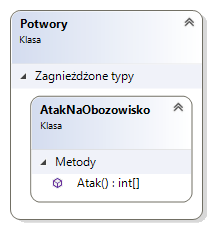
## Broń i Zbroja





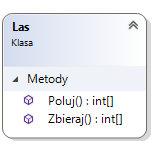
Klasy dziedziczące z klasy przedmiot. Posiadają one mocAtaku i mocObrony odpowiednio oraz metody zwracające te wartości. Dodatkowo posiadają metodę resetującą wartości oraz dwa konstruktory.

## Potwory i AtakNaObozowisko



Klasa AtakNaObozowisko to klasa zagnieżdżona w klasie Potwory. Metoda Atak odpowiada za przeprowadzenie losowego ataku w trakcje nocy w grze. Jest wtedy zabierany łup oraz gracz traci zdrowie.

## Las

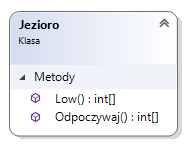


Klasa odpowiada za akcje wykonywane w lesie.

Metody klasy:

* Poluj – kosztem energii daje graczowi jedzenie na podstawie doświadczenia i posiadanej zbroi
* Zbieraj – kosztem energii daje graczowi surowce na podstawie doświadczenia

## Jezioro



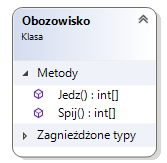
Klasa odpowiada za akcje wykonywane nad jeziorem

Metody klasy:

Low – kosztem energii daje graczowi jedzenie na podstawie doświadczenia

Odpoczywaj – kosztem czasu dodaje graczowi energię

## Obozowisko

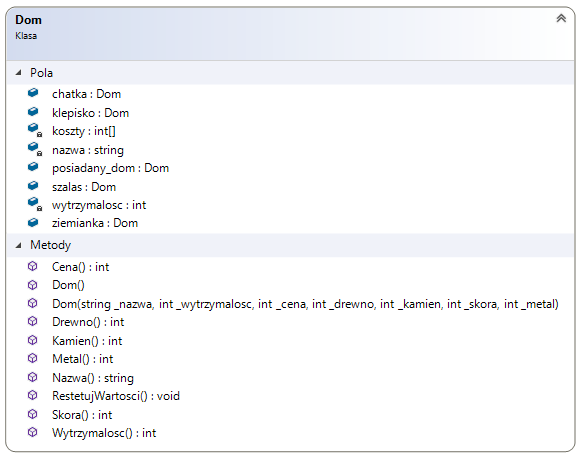


Klasa odpowiada za akcje wykonywane w domu.

Metody klasy:

* Jedz – dodaje życie i energię kosztem odjętego jedzenia
* Spij – dodaje życie i energię kosztem czasu. Umożliwia spanie tylko w konkretnych godzinach

## Dom



Klasa zagnieżdżona w klasie obozowisko, odpowiada za tworzenie różnych rodzajów schronień.

Pola klasy:

* nazwa – nazwa wybranego rodzaju schronienia
* koszty – tablica przechowująca kosztu wybudowania danego schronienia (pieniędzy i surowców)
* wytrzymalosc – wytrzymałość na atak danego schronienia

Pozostałe pola to zdefiniowane rodzaje schronień oraz aktualnie posiadany dom

Metody klasy:

Klasa zawiera dwa konstruktory, metody zwracające wartości pól prywatnych oraz metodę resetującą parametry obiektu klasy Dom.