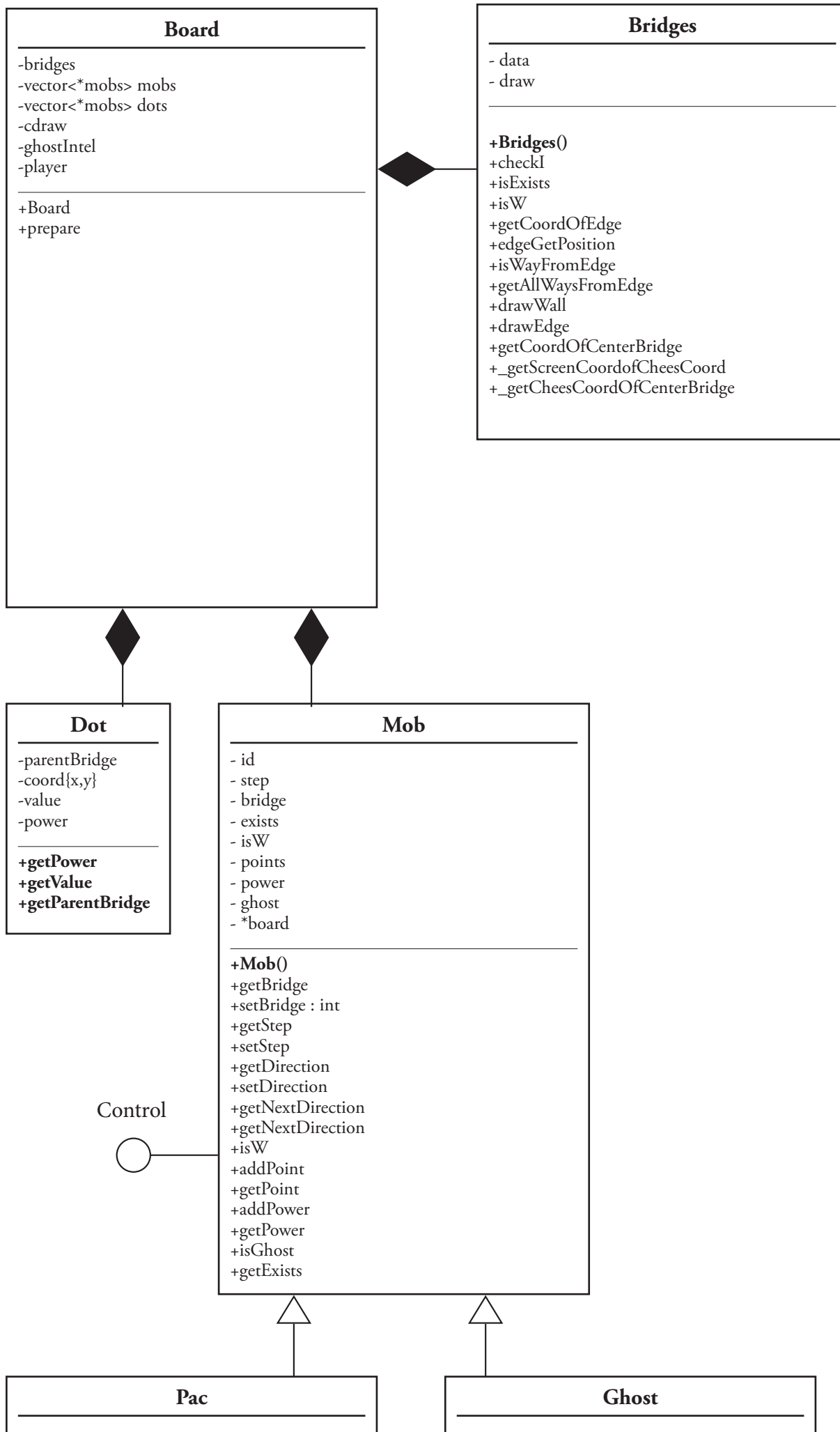


PACMAN 2023

`git@github.com:piotrHeinzelman/msPacMan.git`



Aktorzy (**Actor**) poruszają się po planszy (**Board**), w wybranym kierunku, o ile to możliwe. klient sieciowy może ustawić kierunek (Enum) tylko swojemu awatarowi.
 Jeśli kierunek nie zostanie zmieniony ruch jest kontynuowany.
 Jeśli aktor dotrze do środka pola (step=0) próbuje uciec następne pole, o ile takie istnieje.
 Jeśli aktor osiągnie granicę pola (step= squareDimension) zostaje przeniesiony na następne pole a (step=-squareDimension) i ruch jest kontynuowany do środka pola.
 Po każdym kroczku może zmniejszyć się siła aktora, w razie spotkania wygrywa silniejszy aktor.
 Aktor w konstruktorze dostaje wartość identyfikującą obiekt setrujący jego ruchem.
 Jeśli aktor może zebrać punkt z pola na które wkracza to zbiera je.

Plansza zawiera kolekcję aktorów (ze względu na wydajność raczej rozdzielona, bardziej obiektowo - połączona)
 Na początku rozgrywki przenosi aktorów na pola startowe. I uruchamia zegar gry.

Zegar gry pobudza obiekt Planszy do wywołania dla wszystkich aktorów z kolekcji planszy metodę moveNextStep(); a następnie Plansza sprawdza czy nastąpiły kolizje i ewentualnie reaguje.

Klasy pomocnicze DIRECT - kiedynek (enum)
 ConsoleDraw - klasa rysuje na ekranie konsoli
 UDPServer, UDPClient - wykorzystują Windows2 i tworzą kanał komunikacji sieciowej klienta z serwerem w celu ustanie-
 nia oczekiwanego przyszłego kierunku ruchu awatara.

