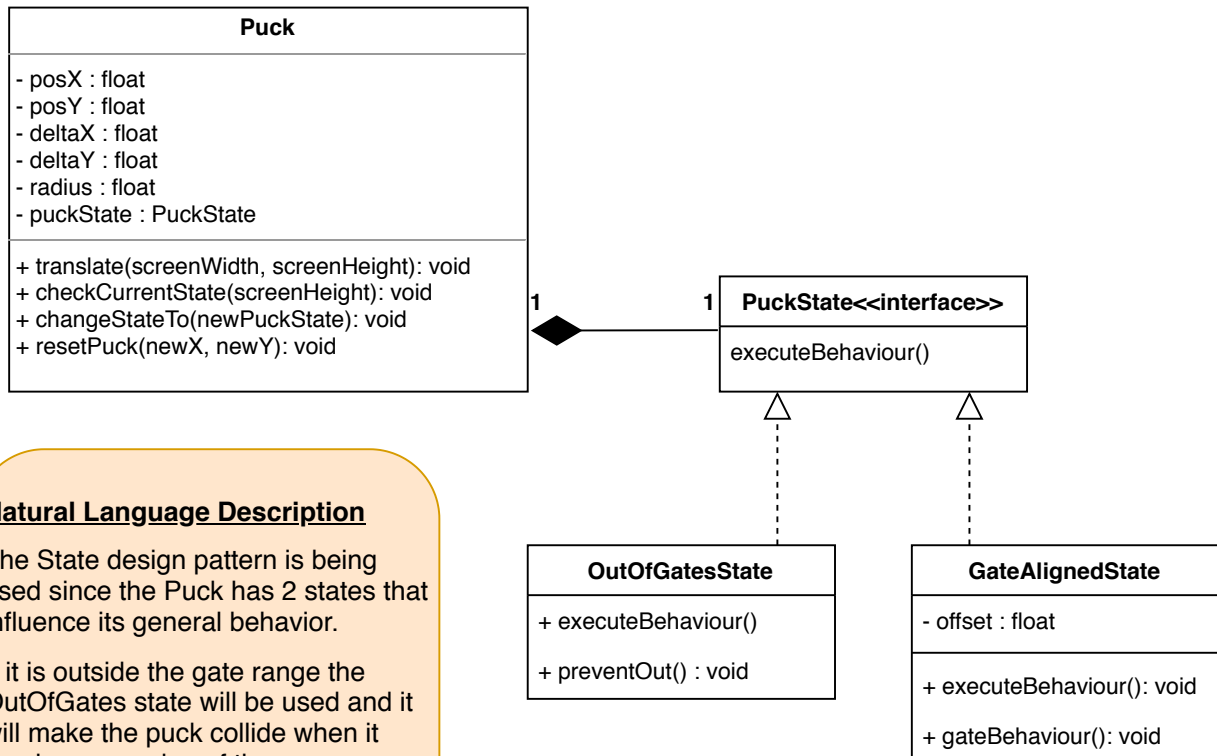


## State Design Pattern



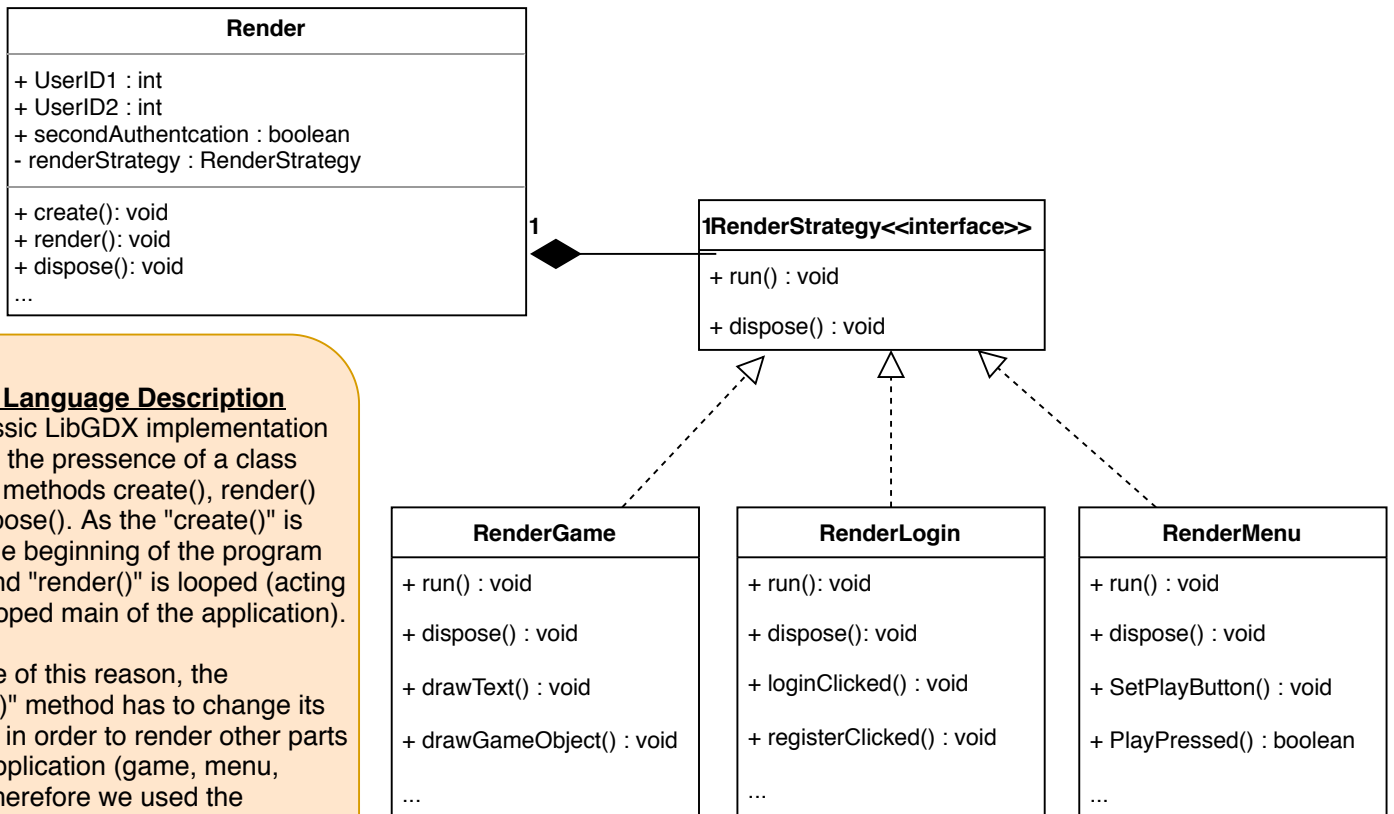
### Natural Language Description

The State design pattern is being used since the Puck has 2 states that influence its general behavior.

If it is outside the gate range the OutOfGates state will be used and it will make the puck collide when it reaches any edge of the screen.

When the puck is Gate Aligned it means that it can go past the edge of the screen in order to score a goal, so the behaviour of the GateAlignedState is used.

## Strategy Design Pattern



### Natural Language Description

The classic LibGDX implementation requires the presence of a class with the methods create(), render() and dispose(). As the "create()" is run at the beginning of the program once, and "render()" is looped (acting like a looped main of the application).

Because of this reason, the "render()" method has to change its strategy in order to render other parts of the application (game, menu, login), therefore we used the Strategy to specify what the render method has to do, depending on the situation and the strategy that must be used for that specific situation