:Player → **:Pacman**: currentSquare = getLocation() : Square

:Player → **:Pacman**: direction = getDirection() : Direction

:Player → **:Board**: newSquare = getSquare(currentSquare, direction)

:Player → **:Pacman**: setSquare(newSquare)

:Player → **newSquare : Square**: landedOn(player)

takeTurn()

We are applying the Expert Pattern here:
By Expert the Pacman class knows its current
square and the board is the expert for all
squares, so it knows which is the next square
for a given square and a direction.
The Player is the expert for taking a turn since
it owns the Pacman and can assign it a new location.

With this design the direction of the pacman
can be set asynchronously and during each turn the
player just pulls the current direction and
places the pacman accordingly. This way it makes
no difference if the direction of a pacman is set
by the server or the user interface.

landedOn(player)

alt [currentPlayer == null]

points = getPoints()

[currentPlayer = null] addPoints(points)

setPlayer(player)

[else]

[color condition] eat(player)