

Monopoly Game CRC Cards

Class: Game
Responsibilities: <ul style="list-style-type: none">-Start Game-Resume Game-Quit Game
Collaborations: <ul style="list-style-type: none">- Board

Class: Board
Responsibilities: <ul style="list-style-type: none">-Store squares-Ensure the rules-Get dice values-Keep track of the turn
Collaborations: <ul style="list-style-type: none">-Square

Class: Square
Responsibilities: <ul style="list-style-type: none">-Represent each square on the board-Show piece
Collaborations: <ul style="list-style-type: none">-Piece

Class: Piece
Responsibilities: <ul style="list-style-type: none">-Get type-Move
Collaborations: <ul style="list-style-type: none">-Card-Building

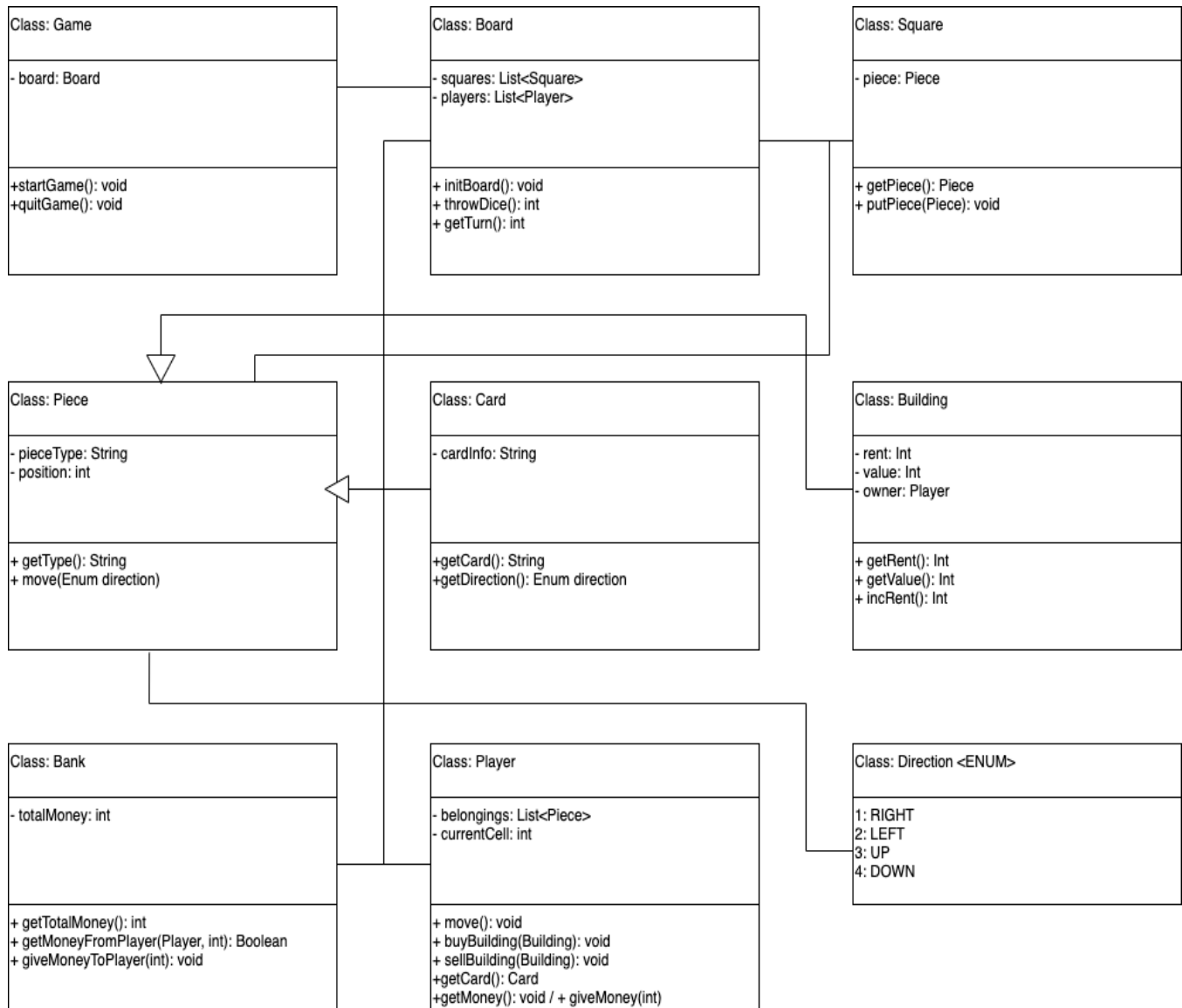
Class: Card
Responsibilities: <ul style="list-style-type: none">-Get information on the card-Move pieces with respect to its type
Collaborations: <ul style="list-style-type: none">-Piece-Player

Class: Building
Responsibilities: <ul style="list-style-type: none">-Get Rent-Get Value-Increment Rent
Collaborations: <ul style="list-style-type: none">-Player

Class: Bank
Responsibilities: <ul style="list-style-type: none">-Get Total Money-Get money from the players-Give money to the players
Collaborations: <ul style="list-style-type: none">-Player

Class: Player
Responsibilities: <ul style="list-style-type: none">-Move-Buy/Sell buildings-Get Card-Get/Give money from/to bank
Collaborations: <ul style="list-style-type: none">-Piece-Bank-Building

Monopoly Game Class Diagram



Alperen Ugus
CWID: 10864101

Why the specified classes are selected?

In my design, I tried to create the most fundamental classes and did not want to waste memory by creating for example a 'Dice' class; because a dice value can be generated by the Board class itself within a member method. Also, I did not create separate classes for houses, hotels etc. since those can be represented just by the building class itself.

I have a 'Piece' class which is the parent class of all pieces that can exist in the game. The Object-Oriented Design Pattern here is used to be able to have list of pieces stored in players since a player can have various pieces in hand.

In this class diagram, I did not specify the methods like getters, setters and draws since those are compulsory and I did not want my class diagram to seem complex.