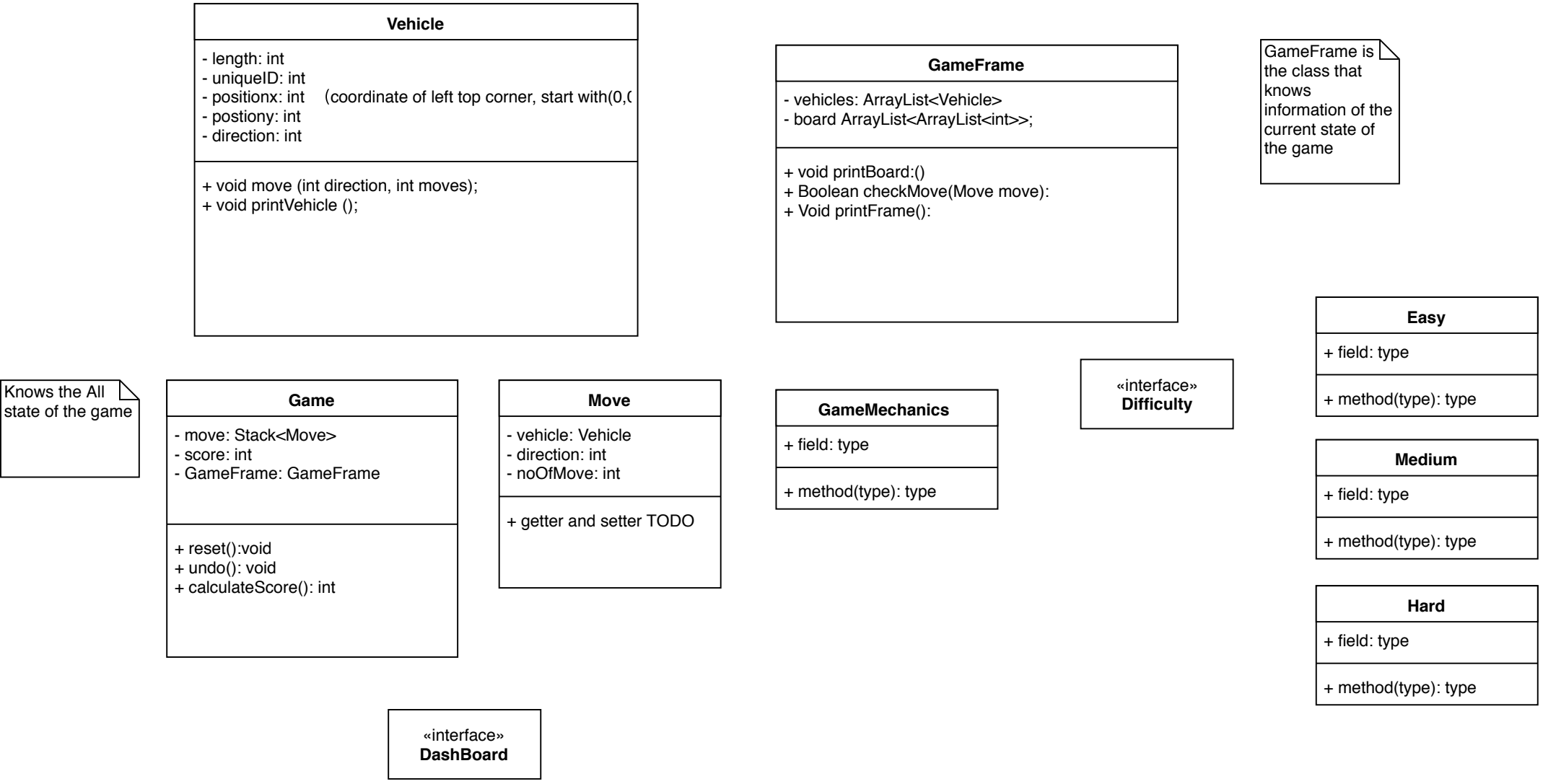
Vehicle
- length: int
- uniqueID: int
- positionx: int (coordinate of left top corner, start with(0,(
- postiony: int
- direction: int
+ void move (int direction, int moves);
+ void printVehicle ();
GameFrame
- vehicles: ArrayList<Vehicle>
- board ArrayList<ArrayList<int>>;
+ void printBoard:()
+ Boolean checkMove(Move move):
+ Void printFrame():
GameFrame is the class that knows information of the current state of the game
Easy
+ field: type
+ method(type): type
Knows the All state of the game
Game
- move: Stack<Move>
- score: int
- GameFrame: GameFrame
+ reset():void
+ undo(): void
+ calculateScore(): int
Move
- vehicle: Vehicle
- direction: int
- noOfMove: int
+ getter and setter TODO
GameMechanics
+ field: type
+ method(type): type
«interface»
Difficulty
Medium
+ field: type
+ method(type): type
Hard
+ field: type
+ method(type): type
«interface»
DashBoard