

<<interface>>

GlassMap

```

+ getDice() : Die[][]
+ getColorConstraints() :
  Color[][]
+ getShadeConstraints() :
  Shade[][]
+ getDie(r : int, c : int) :
  Die
+ isEmpty(r : int, c : int)
  : boolean
+ getDifficulty() : int
+ put(r : int, c : int, die :
  Die)
+ move(oldR : int, oldC : int,
  newR : int, newC : int)
+ move(die : Die, newR :
  int, newC : int)
+ remove(r : int, c : int):Die

```

Coordinate

```

- row : int
- column : int
+ Coordinate(r : int, c : int)
+ getRow() : int
+ getColumn() : int
+ equals(Object o) : boolean
+ hashCode() : int

```

PaperGlassMap

```

- name : String
- dice : Map<Coordinate, Die>
- colorConstraints : Map
  <Coordinate, Color>
- shadeConstraints : Map
  <Coordinate, Shade>
- difficulty : int
+ PaperGlassMap(generator :
  MapGenerator)
+ getDice() : Die[][]
+ getColorConstraints() :
  Color[][]
+ getShadeConstraints() :
  Shade[][]
+ getDie(r : int, c : int) :
  Die
+ isEmpty(r : int, c : int)
  : boolean
+ getDifficulty() : int
+ put(r : int, c : int, die :
  Die)
+ move(oldR : int, oldC : int,
  newR : int, newC : int)
+ move(die : Die, newR :
  int, newC : int)
+ remove(r : int, c : int):Die
+ toString() : String

```

<<interface>>

MapGenerator

```

+ getName() : String
+ getColorConstraints() :
  Color[][]
+ getShadeConstraints() :
  Shade[][]
+ getDifficulty() : int

```

MapFromFileGenerator

```

- name : String
- path : String
- Color[][] : colorConstraints
- Shade[][] : shadeConstraints
+ MapFromFileGenerator(path :
  String)
+ getName() : String
+ getColorConstraints() :
  Color[][]
+ getShadeConstraints() :
  Shade[][]
+ getDifficulty() : int

```

1

uses

0..*

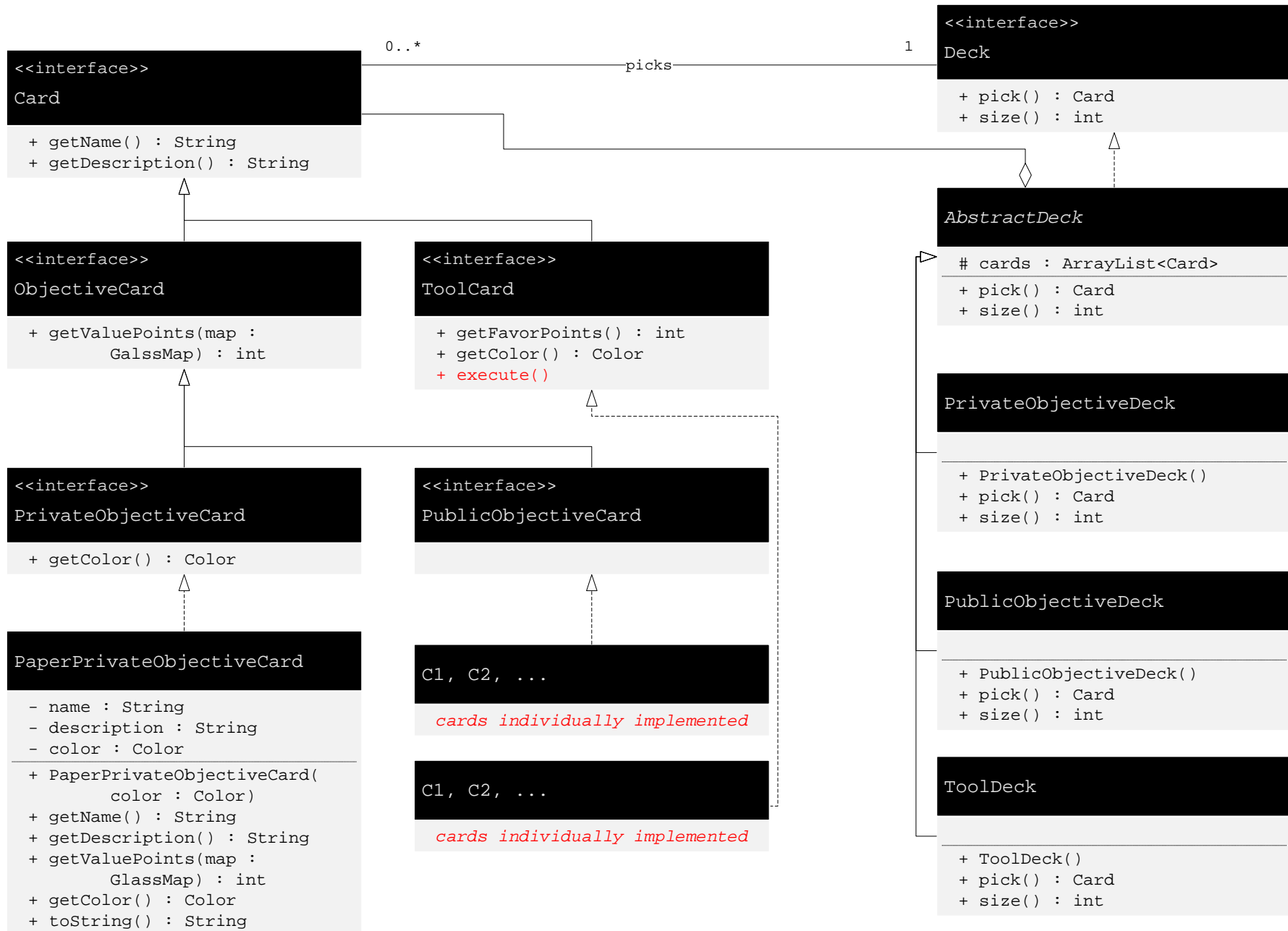


1

generates

0..*

Cards



RoundTrack

```
<<interface>>
RoundTrack

+ getDice() : ArrayList<Die>
+ getCurrentRoundNumber() :
  int
+ getTotalRoundsNumber() :
  int
+ isGameFinished() : boolean
+ getColors() : ArrayList
  <Color>
+ getSumOfShades() : int
+ put(dice : ArrayList<Die>)
+ put(die : Die)
+ swap(userDie : Die,
      roundTrackDie : Die)
```



```
PaperRoundTrack

- dice : ArrayList<ArrayList<
  Die>>
- currentRound : int
- totalRounds : int

+ PaperRoundTrack()
+ getDice() : ArrayList<Die>
+ getCurrentRoundNumber() :
  int
+ getTotalRoundsNumber() :
  int
+ isGameFinished() : boolean
+ getColors() : ArrayList
  <Color>
+ getSumOfShades() : int
+ put(dice : ArrayList<Die>)
+ put(die : Die)
+ swap(userDie : Die,
      roundTrackDie : Die)
+ toString() : String
```

GameBoard

```
GameBox

- instance : GameBox

- GameBox()
+ openBox() : GameBox
+ getDiceBag() : DiceBag
+ getRoundTrack() : RoundTrack
+ getPrivateObjectives(
    numOfPlayers : int) :
  ArrayList
  <PrivateObjectiveCard>
+ getPublicObjectives() :
  ArrayList
  <PublicObjectiveCard>
+ getToolCards() :
  ArrayList<ToolCard>
+ getMaps() : ArrayList
  <GlassMap>
```

Dice

GlassMaps

RoundTrack

Cards

GameModel

```
GameState

- diceBag : DiceBag
- roundTrack : RoundTrack
- dicePool : ArrayList<Die>
- publicObjectives : ArrayList
  <PublicObjectiveCard>
- toolCards : ArrayList
  <ToolCard>
- players : ArrayList<Player>

+ GameState(players :
  ArrayList<Player>)
+ getRoundTrack() : RoundTrack
+ getDicePool() : ArrayList<Die>
+ getPublicObjectives() :
  ArrayList
  <PublicObjectiveCard>
+ getToolCards() : ArrayList
  <ToolCard>
+ getPlayers() : ArrayList
  <Player>
```

Players

GameBoard