```
Utility
```

# <<enum>>

### Color

# <<enum>>

# Shade

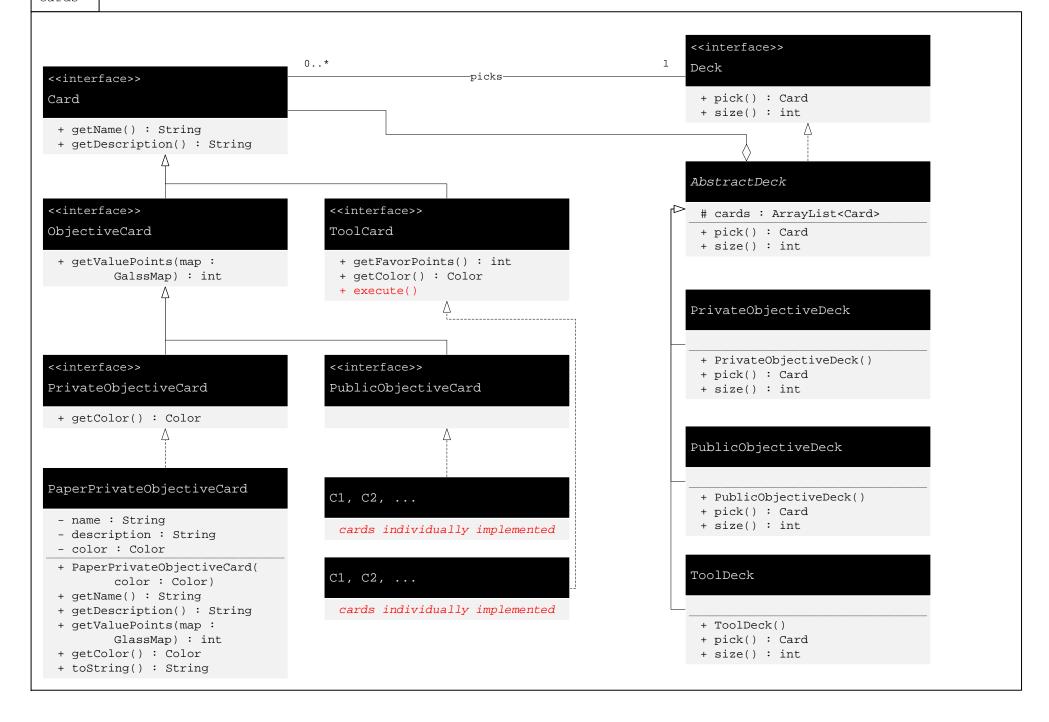
### Parameters

```
+ MAX_ROWS : int
+ MAX_COLUMNS : int
+ PUBLIC_OBJECTIVES : int
+ TOOL_CARDS : int
+ MAX_ROUNDS : int
```

Dice

```
<<interface>>
<<interface>>
Die
                                                   DiceBaq
                                      -uses-
 + throwDie()
                                                     + pick() : Die
 + getColor() : Color
                                                     + pick(numOfDice : int) :
 + getShade() : Shade
                                                             ArrayList<Die>
                                   0..*
 + setColor(color : Color)
 + setShade(shade : Shade)
 + flip()
 + increment()
 + decrement()
                                                    ClothDiceBag
                                                     - progressiveCounter : int
                                                     + ClothDiceBag()
PlasticDie
                                                     + pick() : Die
                                                     + pick(numOfDice : int) :
                                                             ArrayList<Die>
 - shade : Shade
 - color : Color
 - ID : int
 + PlasticDie(ID : int)
 + PlasticDie(die : Die)
 + getColor() : Color
 + getShade() : Shade
 + setColor(c : Color)
 + setShade(s : Shade)
 + throwDie()
 + flip()
 + increment()
 + decrement()
 + equals() : boolean
 + toString() : String
```

```
Coordinate
<<interface>>
                                           - row : int
                                                                                        PaperGlassMap
                                           - column : int
GlassMap
                                                                                 uses-
                                           + Coordinate(r : int, c : int)
                                                                                          - name : String
 + getDice() : Die[][]
                                           + getRow() : int
                                                                                          - dice : Map<Coordinate, Die>
 + getColorConstraints():
                                           + getColumn() : int
                                                                                          - colorConstraints : Map
         Color[][]
                                           + equals(Object o) : boolean
                                                                                                  <Coordinate, Color>
 + getShadeConstraints():
                                           + hashCode() : int
                                                                                          - shadeConstraints : Map
         Shade[][]
                                                                                                  <Coordinate, Shade>
 + getDie(r : int, c : int) :
                                                                                          - difficulty : int
         Die
                                                                                          + PaperGlassMap(generator :
 + isEmpty(r : int, c : int)
                                                                                                  MapGenerator)
         : boolean
                                                                                          + getDice() : Die[][]
 + getDifficulty() : int
                                                                                          + getColorConstraints():
 + put(r : int, c : int, die :
                                                                                                  Color[][]
                                           <<interface>>
         Die)
                                                                                          + getShadeConstraints() :
 + move(oldR : int, oldC : int,
                                          MapGenerator
                                                                                                  Shade[][]
         newR : int, newC : int)
                                                                                          + getDie(r : int, c : int) :
 + move(die : Die, newR :
                                            + getName() : String
                                                                                                  Die
        int, newC : int)
                                            + getColorConstraints() :
                                                                                          + isEmpty(r : int, c : int)
                                                                              1
 + remove(r : int, c : int):Die
                                                    Color[][]
                                                                                                  : boolean
                                            + getShadeConstraints() :
                                                                                          + getDifficulty() : int
                                                    Shade[][]
                                                                                          + put(r : int, c : int, die :
                                            + getDifficulty() : int
                                                                                                  Die)
                                                                                          + move(oldR : int, oldC : int,
                                                                                                  newR : int, newC : int)
                                                                                          + move(die : Die, newR :
                                          MapFromFileGenerator
                                                                                                  int, newC : int)
                                                                                          + remove(r : int, c : int):Die
                                                                              generates
                                                                                          + toString() : String
                                            - name : String
                                            - path : String
                                                                                                          0..*
                                            - Color[][] : colorConstraints
                                            - Shade[][] : shadeConstraints
                                            + MapFromFileGenerator(path :
                                                    String)
                                            + getName() : String
                                            + getColorConstraints():
                                                    Color[][]
                                            + getShadeConstraints():
                                                    Shade[][]
                                            + getDifficulty() : int
```



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<</pre>
        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

- diceBaq : DiceBaq - roundTrack : RoundTrack - dicePool : ArravList<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