Neil Quinn

[Email address]

Abstract

This document outlines Models for individual Objects inside the Kings N’ Things Game. It includes general descriptions, fields, methods, and any other relevant information for each.

kings n’ things

Object Models

Contents

[Hex Tiles 2](#_Toc377481632)

[Player 3](#_Toc377481633)

[Game Board 4](#_Toc377481634)

[Player Rack 5](#_Toc377481635)

[Things 6](#_Toc377481636)

[Fort 7](#_Toc377481637)

[Settlement 8](#_Toc377481638)

[Special Income Thing 9](#_Toc377481639)

[Incomable (Interface) 10](#_Toc377481640)

[Magic Thing 11](#_Toc377481641)

[Treasure Thing 12](#_Toc377481642)

[Event Thing 13](#_Toc377481643)

[Creature Thing 14](#_Toc377481644)

[Special Characters 15](#_Toc377481645)

[Game 16](#_Toc377481646)

[Game Model 17](#_Toc377481647)

[Game View 19](#_Toc377481648)

[Combatant 20](#_Toc377481649)

[Dice 21](#_Toc377481650)

[Terrain Lords 22](#_Toc377481651)

# Hex Tiles

Implement incomable

Fields:

enum terrain : {Unrevealed, Jungle, Frozen Waste, Forest, Plains, Swamp, Mountain, Desert, Sea}  
(Note: all but “Sea” and “Unrevealed”are considered “land”)  
enum controlledBy : {Neutral, Player1, Player2, Player3, Player4}  
Vector<Thing> player1Things  
Vector<Thing> player2Things  
Vector<Thing> player3Things  
Vector<Thing> player4Things  
Fort fort  
SpecialIncome specialIncome

# Player

Fields:

PlayerRack playerRack  
int playerNum  
int gold  
int playerOrder  
Vector<Thing> heldThings

# Game Board

Fields:

HexTile [7][7] hexTiles

# Player Rack

Fields:

Vector<Thing> things

# Things

Fields:

int thingID  
bool isFlipped  
enum thingType : {…}  
Texture image? (Shouldn’t this be in the view?)  
String name

# Fort

Is a Building

Implements incomable

Fields:

enum level : {Tower, Keep, Castle Citadel}

# Settlement

Is a Building

Implements Incomable

Fields:

# Special Income Thing

Is a Thing

Implements Incomable

Fields:

int income  
enum terrain : {Unrevealed, Jungle, Frozen Waste, Forest, Plains, Swamp, Mountain, Desert, Sea}

# Incomable (Interface)

# Magic Thing

Is a Thing

# Treasure Thing

Is a Thing

# Event Thing

Is a Thing

# Creature Thing

Is a Combatant

Fields:

enum terrain : {Unrevealed, Jungle, Frozen Waste, Forest, Plains, Swamp, Mountain, Desert, Sea}

# Special Characters

Is a Combatant

Implements incomable

(need method for special ability)

# Game

Fields:

GameModel gameModel  
GameView gameView

# Game Model

Fields:

GameBoard gameBoard  
Player player1, player2, player3, player4  
int numPlayers  
Thing[234] playingCup  
SpecialCharacter[<# of Special Characters>] unownedCharacters  
SpecialCharacter[<# of Special Characters>] ownedCharacters  
Dice dice

# Game View

# Building

Is a Combatant

Fields:  
 int hits

# Combatant

Is a Thing

Fields:

int combatValue  
bool isFlying  
bool isMagic  
bool canCharge  
bool hasSpecial

# Dice

# Terrain Lords

Is a Special Character

Fields:

enum terrain: