

# CRC Cards

Simplified Monopoly

# CmdLineUI

+ runPlayerSetUp()

GameController

# GameController

- id
  - filepath
  - netWorthGoal
  - maxRounds
  - BANKRUPT\_VALUE
  - boardManager
  - propertyManager
  - auctionManager
  - moneyManager
- 
- + getPath()
  - + getId()
  - + save()
  - + load()
  - + runPlayerSetUp()
  - + runGame() *// initialize all managers*

BoardManager  
PropertyManager  
AuctionManager  
MoneyManager

# BoardManager

- board //an instance of Board
- players //a list of Players with pieces on the Board
  
- + getPlayerList()
- + getPlayerLocations()
- + addPlayer()
- + updatePos() *//update player position by moving player up x tiles and determining which tile the player is on*
- + updateBankruptcy()
- + turn() *// a single turn in which a player rolls a die, moves, buys/sells*
- + generateOrder() *// generates the order that the players take turns in*
- + toString()

Player  
Board

# PropertyManager

- properties
- propertiesOwned //hashmap of all properties in the game  
and the players that own them
- + resetProperties() // re-make properties available of a  
player that was eliminated
- + buyProperty()
- + sellProperty()
- + propertyAction(player: Player) // determines whether the  
player pays the bank, can buy a property or pays rent to  
another player

City  
PublicProperty  
Building

# AuctionManager

+ tradeProperties() //trade properties between players if  
they are auctionable

Player  
City

# MoneyManager

- payRent(renter: Player, payee: Player) \*\*// could be moved to property manager for simplicity but there is high coupling accessing player attributes
- deductCostOfProperty(buyer: Player, cost: int)
- addSellbackOfProperty(seller: Player, cost: int)
- calculateNetWorth()

Player

# Player

- username
- cash
- netWorth
- token

- + roll()
- + getCash()
- + setCash()
- + addCash() *//more useful than calling setCash when just giving a player money*
- + getNetWorth()
- + setNetWorth()
- + isBankrupt()
- + setBankrupt()
- + getUsername()

Token



# Token

- id
- colour
- location
- inJail

- + getColour()
- + getLocation()
- + setLocation()
- + isInJail()
- + move()

Player

# Board

- tiles ArrayList<Tile>
- + getTileAt() // returns Tile at a particular location

Tile

Enterprise Business Rules

abstract

# Tile

PropertyTile  
SpecialTile

- interact()

# PropertyTile

- name
- rent
- price
- salePrice // for when a player sells a property, calculated as 80% of the original price
- owned // boolean storing whether or not someone owns this
- owner (Player object or null as bank)
- getRent()
- setRent()
- getPrice()
- getSalePrice()
- isOwned()
- setOwned()
- setOwner()
- getOwner()
- purchase()
- sell()

# PublicProperty

- setRent() [Override] // uses PropertyManager

Player  
Token

# City

PropertyTile  
Buildable  
Auctionable  
Buyable

- colour
- buildings // a list of Building objects
- interact() [Override]
- setRent() [Override] // PropertyManager uses this
- getColour()
- sameColourBlock() // returns boolean representing whether this city is the same colour as another parameter city
- addBuilding()
- removeBuilding()
- getBuildings() // return a list of buildings on a city property

Player  
Token  
Building

Enterprise Business Rules

abstract

# SpecialTile

JailTile, AuctionTile,  
SurpriseTile, StartTile

# StartTile

- interact() [Override] //interacts with player to give them cash

Player



# JailTile

- interact()

Token

# SurpriseTile

- randomEffects
- interact() [Override]

Token

# AuctionTile

- interact() [Override]

# Building

- cost
- rentIncrease
- + getCost()
- + getRentIncrease()

City

Interface

# Buyable

Implemented by PropertyTile

- getPrice()
- getSalePrice()
- purchase()
- sell()

Interface

# Buildable

Implemented by City

- addBuilding()
- removeBuilding()

Interface

# Auctionable

Implemented by City,  
PublicProperty

- //marker interface (meaning it is empty it simply exists to exhibit a property of a class)

Interface

# Movable

Implemented by Token

- move()



# Main

---

- `main()` // creates instances of GameController and CmdLineUI and initiates game