Trading Card Game

About this Kata

In this Kata you will be implementing a rudimentary two-player trading card game. The rules are loosely based on Blizzard Hearthstone (http://us.battle.net/hearthstone/en/) which itself is an already much simpler and straight-forward game compared to other TCGs like *Magic: The Gathering* (http://www.wizards.com/magic/).

- Difficulty: Medium/Hard
- Good for teaching: Test Doubles, Mocking, TDD, Refactoring, Clean Code

Problem Description

Preparation

- Each player starts the game with 30 Health and 0 Mana slots
- Each player starts with a deck of 20 Damage cards with the following Mana costs: 0,0,1,1,2,2,2,3,3,3,3,4,4,4,5,5,6,6,7,8
- From the deck each player receives 3 random cards has his initial hand

Gameplay

- The active player receives 1 Mana slot up to a maximum of 10 total slots
- The active player's empty Mana slots are refilled
- The active player draws a random card from his deck
- The active player can play as many cards as he can afford. Any played card empties Mana slots and deals immediate damage to the opponent player equal to its Mana cost.
- If the opponent player's Health drops to or below zero the active player wins the game
- If the active player can't (by either having no cards left in his hand or lacking sufficient Mana to pay for any hand card) or simply doesn't want to play another card, the opponent player becomes active

Special Rules

- Bleeding Out: If a player's card deck is empty before the game is over he receives 1 damage instead of drawing a card when it's his turn.
- Overload: If a player draws a card that lets his hand size become >5 that card is discarded instead of being put into his hand.
- Dud Card: The 0 Mana cards can be played for free but don't do any damage either. They are just annoyingly taking up space in your hand.

Clues

When approached iteratively with TDD you can take different starting points, like the player state or the game loop. It is your own choice whether you implement the game for human or computer players - or both. Game visualization can be anything between System.out and a GUI. You can also increase the difficulty by adding more rules, like Healing cards, Damage independent from Mana cost or introducing individual Deck building. You will find some examples of Advanced Variations from the Kata's author at https://github.com/bkimminich/kata-tcg . Even without extra rules the toughest part of this Kata might be coming up with actually smart CPU player decision-making algorithms.

Example Solution

- Example solution in Java 8 from the Kata's author BjoernKimminich

Edit

fr **en** **pt**