The Old Maid Card Game

In this assignment, you are required to use Java Multithreading to simulate the old maid card game, in which each player is represented as a thread that plays the game.

First, let us go over the game rules. The game can be played by two players or more using the standard 52-card deck, plus one additional card, the Joker. The game starts by dividing the cards equally between the players (it is okay if some players have one more card than the others). The objective of the game is to discard all cards. Specifically, a player can discard a "matching pair" of cards, which are two cards with the **same value** and the same color, i.e., Spades (♠) must match with Clubs (♣) and Diamonds (♦) must match with Hearts (♥). Because the Joker has no match, the player who has the Joker at the end of the game will lose.

Assuming we have four players, at the beginning of the game and after the cards are dealt, all players will throw all matching pairs in their hands. Then, the first player will pick one random card from the second player and add it to his or her hand. If the card results in having a matching pair, then the player may discard this pair. Similarly, the second player will take one random card from the third player and discard any matching pair. Then, the third player will take one random card from the fourth player and discard any matching pair. Finally, the fourth player will take one random card from the first player and discard any matching pair. This circle will continue till eventually all players have discarded their cards, except for the player who has the Joker, who will be considered the loser of the game.

What you need to do?

- Use Java Multithreading and OOP to implement the old maid game that is played automatically by the computer (no human players are included).
- Each player should be running as a separate thread in your program. Therefore, if there are N players, then there must be N threads in the game, plus the main thread. The number of players is an input to the game.
- The main thread role is to start all player threads, divide the cards between them, and then report the game results at the end. The main thread should NOT be used as one of the players.
- For efficiency, make sure all player threads that in the "wait" state when it is not their turn. In other words, you need to figure out how to use wait/notify mechanism appropriately between threads to simulate the game correctly while ensuring efficient use of the CPU.
- Note: the focus of the assignment is Java multithreading, but please do not ignore OOP design as it will also be taken into consideration during assignment grading.

What you need to submit:

1. Your source code, i.e., all of your *.java files.
2. A jar file that I can run to play your game.
3. A report that:
    a. Explains object-oriented design in your code
    b. Explain thread synchronization mechanisms used in your code
    c. Defend your code against clean code principles (Uncle Bob)
4. A video that shows a run demo of the game. Please restrict your video length to 5 minutes.