Take Home Project: Blackjack

If you're not familiar with the game, here are the simplified rules we will be going by for this project:

1. The game consists of two players: You vs The House (the computer), where the goal is to beat the House's hand, without going over 21.
2. A card contains a "point" value equivalent to it's number (the 3 of club is worth 3 points...the 9 of spades is worth 9 points...etc etc). Face cards (Jack, Queen, King) are worth TEN points, and the Ace card is either worth 1 or 11, whichever is most helpful for the player's hand. For example:
    a. If the player has a Jack and a Queen, and then draws an Ace, the Ace will be worth 1 point to add up to 21
    b. If the player has a Queen and an Ace, the Ace will be worth 11 points to add up to 21
    c. If the player has a 2 and an Ace, the Ace will be worth 11 points to get closer to 21
    d. If the player has a 2 and a 5, and then draws an Ace, the Ace will be worth 11 points to get closer to 21. If the player then draws a 10, the Ace will now be worth 1 point
3. The House is initially dealt TWO face up cards and no more! This isn't part of the regular rules for Blackjack, but it is for us. In other words, the House will always only have 2 cards.
4. You are also initially dealt two face up cards, but you have one of the following options:
    a. Hit: You are dealt one more card to add to your point value. For this project, the player may hit as many times as they like, until their card value exceeds 21, at which point the game ends in an automatic loss
    b. Stand: Ends the round (for the purposes of this project, this will end the game)
5. Once you end the round, the game is over, and there should be a display of whether you won or lost.
6. Don't deal with a new deck every hand or rely on a refresh. You should be able to run through the deck and shuffle at the end when needed.

Conditions

You Win if:

- The House's total is > 21 and your total is < 21 (for the purposes of this project, you can ignore this condition, since the House will only have two cards and cannot get a total > 21)
- your current total is < 21 but higher than the House's total
- Your current total is 21 and the House's total is not 21

You lose if:

- Your current total totals over 21 (don't forget to factor in the different edge cases of the Ace card!)
- You current total is < 21 but lower than the House's total
- You tie with the House

To implement this, you'll use this API for card management: http://deckofcardsapi.com/. You should be able to initialize one deck and deal out cards from the deck using this API.

What we are looking for:

- Setting up a React project
- Code organization within your project. Although, it's a small scale. We're looking to see how you'd write code in production.
- Your interaction working with third party clients.
- UI/UX Layout decisions
- Following the spec.

Optional

A very optional additional element to this project is writing it in our current stack, if you're familiar. React, typescript, & NextJS. Don't feel the need to learn these technologies if you're not familiar with them already, but if you are, it's helpful to understand how you work with them.

Deliverable

Please deliver this as a cloneable git repo