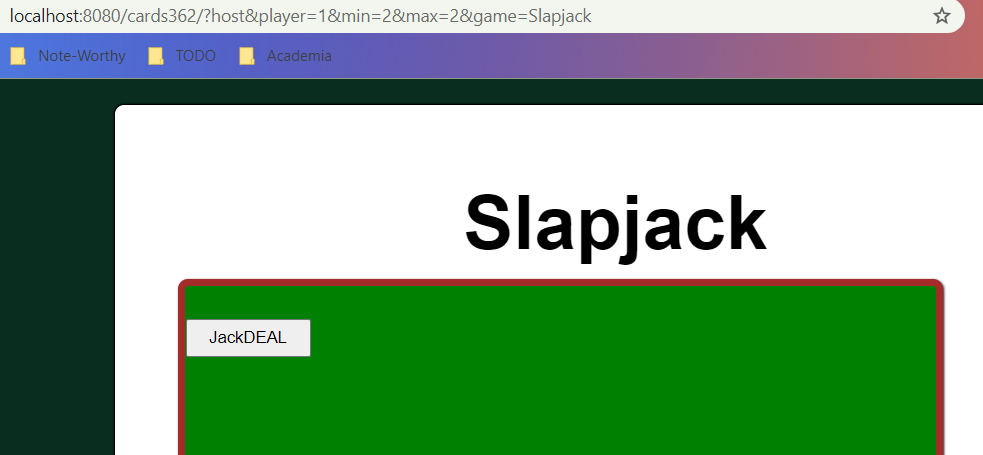
**Team 12: Plan**

**Features to Implement:**

* How does the GameController know that a game is selected and what does it need to do before it can call match.start()? Set up the required infrastructure for GameController to start the match
  + **Andrew Marek**
    - Done, I think. ✅
* How does MatchController know when the game can begin and what does it need to do before it can call mainloop.play()? Set up the required infrastructure for MatchController to start the match.
  + **Andrew Marek**
    - Done, I think. ✅
* At the start of play there should be a deal button and the title of the game should be set to “Slapjack”.
  + **Steven Sheets**
    - Done, I think. ✅
    - 
* Pressing the deal button results in two even piles of shuffled face down cards. There are many simple shuffle algorithms, any is fine.
  + **Steven Sheets**
* During play, the player’s alternate selecting the card at the top of their pile. Their card is placed face up on a center pile. Players are ignored if they select a card out of turn or from a pile that does not belong to them.
  + **Michael (Zhifan) Huang**
* When a player selects a Jack on the center pile all cards in the pile are transferred to the bottom of their pile. A player that selects a card incorrectly is ignored.
  + **Michael (Zhifan) Huang**
* The display of the player's score always represents the number of cards in their pile.
  + **Alexis Cordts**
* When a player runs out of cards the other player wins at the end of their turn. Set the title of the game to “Player X Wins”.
  + **Alexis Cordts**
* (optional) Extra game play rule: if a player improperly selects a center card that is not a Jack the other player wins all of the cards in the center pile.
  + **Jared Weiland**
* (optional) Extra game play rule: on each play the center pile is moved to a random location to prevent a player from hovering over the pile.
  + **Jared Weiland**
* (optional) When the game is finished show the deal button. This will be helpful. String remoteId = view.getRemoteId(DealButton.kSelector); view.send(new ShowButtonRemote(remoteId));
  + **Evan Christensen**
* (optional) When the deal button is selected for a new game, deal the existing cards (not new ones) to the two players.
  + **Evan Christensen**

*(Complete a short planning document detailing the tasks that need to be completed, dependencies between tasks and any other risks to the success of the project.)*

**Task Dependencies:**

* TODO. Analyze the features to implement and note the dependencies between these tasks.

**Risks:**

* Procrastination
* Lack of communication
* Not asking our TA questions if/when we get stuck

**Additional Notes:**

* There is a relatively high chance that some of these features will be far easier to implement than others. If you can finish your features with ease, potentially assist with the others.