3 Use Cases:

Group 5A (Connie Urie, Cloette Owensby, and Bolun Zhang)

CASE 1: Starting Solitaire

1. *Variation 1*
   1. User Goal: To begin a game of Solitaire
   2. User Action:
      1. User clicks on the application icon, then the start menu interface appears
      2. User clicks on the button labelled “Start”, which becomes highlighted once they click on it.
   3. Software:
      1. Loads the menu interface
      2. Upon the user clicking “Start”, highlights the button and initializes the selected game type. (In this case, the default, since the user does not select a type or acknowledge the available drop-down menu.)
2. *Variation 2*
3. User Goal: To begin a game of Solitaire (Klondike Type)
4. User Action:
   1. User clicks on the application icon, then the start menu interface appears
   2. User moves the mouse to click on the drop-down menu and hovers over the item “Klondike” and clicks, selecting the item (the item becomes highlighted).
   3. User clicks on the button labelled “Start”, which becomes highlighted once they click on it.
5. Software:
   1. Loads the menu interface
   2. Displays the contents of the drop-down menu upon the user’s click.
   3. Updates the gameType variable with the most recent selection of the type of game by the user and highlights the latest selection.
   4. Upon the user clicking “Start”, highlights the button and initializes the selected game type. (In this case, “Klondike”.)
6. *Variation 3*
   1. User Goal: To begin a game of Solitaire (Unavailable Type), then to close the application.
   2. User Action:
      1. User clicks on the application icon, then the start menu interface appears
      2. User moves the mouse to click on the drop-down menu and hovers over the items, clicking on several (the latest choice is highlighted).
      3. The user decides the type of game they want to play is not there, and clicks away from the drop down menu. The drop down menu collapses back into its original size, with a default type highlighted.
      4. User clicks the “x” icon at the top of the window to close the application
   3. Software:
      1. Loads the menu interface
      2. Displays the contents of the drop-down menu upon the user’s click.
      3. Updates the gameType variable with the most recent selection of the type of game by the user and highlights the latest selection.
      4. When the user clicks in a blank area (other than the “Start” button), the full list of game types closes so only the default option is visible and highlighted.
      5. Upon the user clicking “x”, the application stops running and closes the interface window.

CASE 2: Playing Solitaire

1. *Variation 1*
2. User Goal: To move a card from one stack of cards to another. (Klondike Type)
3. User Action:
   * 1. User clicks on the card they would like to move. The card becomes highlighted.
     2. User clicks on the stack they would like to move the card to, but the move is incorrect.
     3. User clicks on the same card again (it highlights) and then clicks on a different stack, the card appears at the desired stack and is no longer highlighted.
4. Software:
   * 1. Highlights the face-up card the user clicks on
     2. Tests the value of the card and the values in the selected/clicked stack against the organizing rule and rejects or accepts the move (in this case the move is rejected).
     3. Since this first move is invalid, it is rejected, and the card does not move. It becomes un-highlighted as well.
     4. Highlights the same face-up card the user clicks on
     5. Tests the value of the card and the values in the selected/clicked stack against the organizing rule and rejects or accepts the move (in this case the move is accepted).
     6. Moves the card to appear on the selected stack in the front, face-up and un-highlighted. Saves the move to the moveRecord (so it can be undone/redone).
     7. Checks to see if the user has won the game by checking the value in the deck and in each of the suit collections. If the deck is empty and the suit collections are full, the winning sequence is executed.
5. *Variation 2*
6. User Goal: To undo a move they made and then redo that move. (Klondike Type)
7. User Action:
   1. User clicks on the “Undo” button.
   2. The user observes other possible choices, concludes that they did make the best choice and clicks the”Redo” button.
8. Software:
   1. Highlights the “Undo” button upon the user’s click
   2. Takes the latest record of moves committed by the user (in moveRecord) and copies the latest move in a separate location, then reverses the last move made by the user. Also deletes that move from the moveRecord. Un-highlights the “Undo” button.
   3. Highlights the “Redo” button upon the user’s click
   4. Goes to the separate location, retrieves the movement information, and performs the last move that was undone. Un-highlights the “Redo” button. Clears the separate location memory once the move is completed.

CASE 3: Finishing Solitaire

1. *Variation 1*
2. User Goal: To finish the game of Solitaire (Klondike Type) (loses the game)
3. User Action:
   * 1. User moves the last available card
     2. The program displays a “Sorry. Maybe next time?” message over the game interface with a “Restart” button and a “Quit” button
4. Software:
5. Checks the value in the deck and in each of the suit collections. If the deck is empty and the suit collections are full, the winning sequence is executed. If the deck is empty and there are no more moves available, but the suit collections are not full, then the loss sequence is executed.
6. The loss sequence activates, displaying a “Sorry. Maybe next time?” message over the game interface with the “Restart” button and the “Quit” button. (no timer)
7. *Variation 2*
8. User Goal: To quit the game of Solitaire (Klondike Type) before they are out of cards because they know they are losing and are frustrated.
9. User Action:
   1. User clicks on the “Quit” button, which becomes highlighted
   2. The program displays an “Are you sure?” message, and the user does nothing for 5 seconds, then the application closes.
10. Software:
    1. Highlights the “Quit” button upon the user’s click
    2. Displays the “Are you sure?” message
    3. Begins the timer for 5 seconds
    4. If no key is pressed in those 5 seconds, the application closes
11. *Variation 3*
12. User Goal: To quit this game and begin a new game of Solitaire
13. User Action:
    * 1. User clicks on the “Restart” button, which becomes highlighted
      2. The user’s interface switches from the gameType interface to the Menu interface.
14. Software:
    * 1. Highlights the “Restart” button upon the user’s click
      2. Closes the gameType interface, resets all variable values to their defaults, and replaces the game interface with the Start Menu interface.