Requirements Part Two

Use Case Name: User Playing the game

Actors: Player

Flow of Events:

1.1 -Preconditions: this use case begins when the user goes into the game and presses start game, the user is given options to PLAY, QUIT, or RESTART. Pressing QUIT will end the game, pressing PLAY will continue the game

2.1- Main Flow:

- If the user selects PLAY , the game will proceed and the user will be able to place cards upon other cards , playing solitaire
- If the user selects QUIT after PLAY, the game will exit out after a final score is given based on their placement at the end of the activity.
- If the user selects RESTART , they can restart the game at any point to refresh the score and loop the program back to PLAY
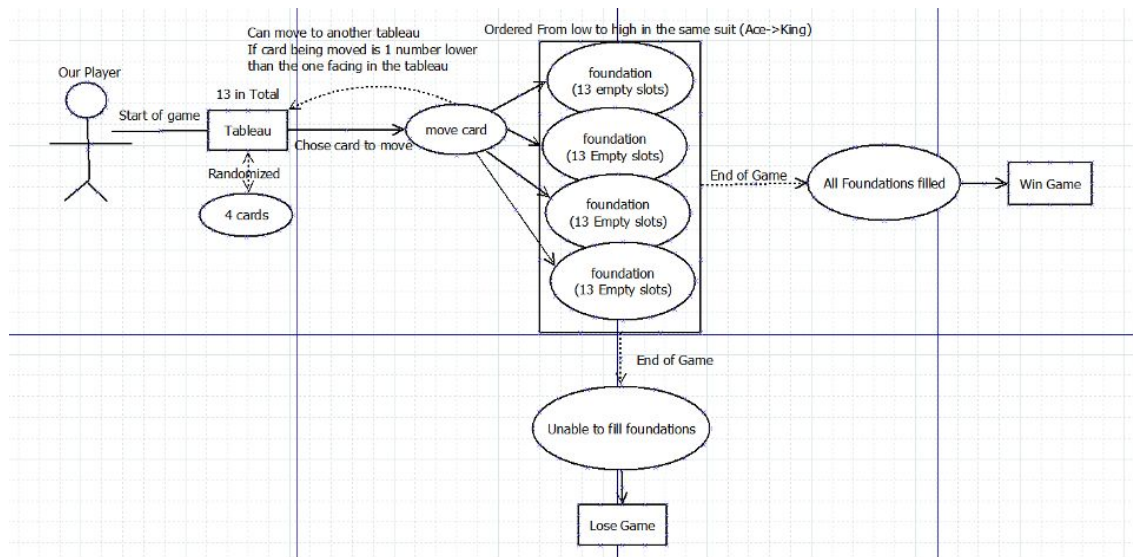
3. Subflows

S.3.1 - Win Game: If the user beats the game of solitaire by reaching the required score, the game will flow into a win message from PLAY screen. WIN message displays a congratulatory message along with the score and the option to RESTART

4. Exceptional Flows

E.4.1 - Invalid move: this exceptional flow will be implemented if the user performs a solitaire action that isn't allowed in the game, the game will reset to before the invalid move was made

Case Diagram

Our Player

Can move to another tableau
If card being moved is 1 number lower
than the one facing in the tableau

Ordered From low to high in the same suit (Ace->King)

Start of game

13 in Total

Tableau

Chose card to move

Randomized

4 cards

move card

foundation
(13 empty slots)

foundation
(13 Empty slots)

foundation
(13 Empty slots)

foundation
(13 Empty slots)

End of Game

All Foundations filled

Win Game

End of Game

Unable to fill foundations

Lose Game

Class Diagram

Main

Style

Implimentation

Rules

Sound

Placement

Design

Implementation
Logic

Rules Logic