TO DO

* Context menus using clickcount which is not synchronized, could create sync problems.
* Grids using division and rounding could create sync problems.

PLAN

* Every element has an owner, either a specific player or the activePlayer (prevents collisions).
* All the mouse events from 1 second or 0.25 second gets bundled into a packet and sent. The packet is an array of say 50 fractions of that time period. Upon receiving a packet, all the fractions are dumped into a buffer. The buffer plays back from the buffer, playing up to 3x speed to catch up or down to 1/3rd speed to smooth out incoming information.
* Packets are numbered and never used out of order, just stay in buffer till its time.
* Event types: mouseMove, mouseLeftDown, mouseRightDown, mouseLeftUp, mouseRightUp.
* Events all have mouse coordinates in absolute board position (not the pan and zoom position)
* Someone is the activePlayer. Buttons to pass control to each of the other players.
* Your screen has a white border if you are the activePlayer, black border if not.
* Random numbers are synced.
* Element types: buttons, decks, cards, dice, tokens.
  + GameObject->MoveableObject
    - Token
    - Card
    - Die
  + GameObject->StaticObject
    - Button
    - Deck
    - Board
    - Grid
* To figure out grid locations: load game with coordinates displayed, move mouse to see coordinates of that location.
* Grids: each grid has a name. Elements can snap to 1 or more grids. Grid is just an array of x,y points. Snap option: snap to centers, or left corners.
* You can see everyone’s mouse all the time.
* Elements being held are wobbly and have shadow cast.
* Game elements push on each other till they are not on top of each other.
* Draw objects currently held above other objects
* Flipable, rollable, snapable
* left drag to pick up and move one object.
* ctrl+left drag to pick up and move anything you drag over.
* shift+left to pan
* shift+scroll to zoom
* right click to flip card
* right click deck for menu (shuffle)
* right click die for menu (set to value: 1,2,3,4,5,6)