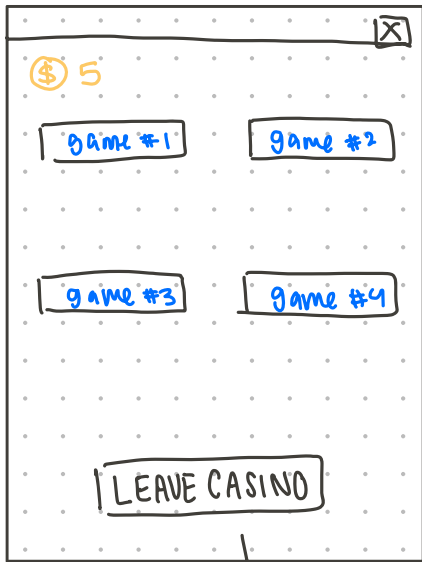
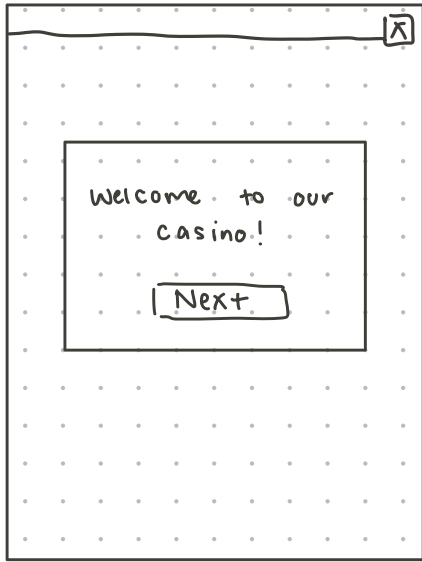


NOTE: Overall idea! probably exists better components

FINAL = 11th project

Due: 8th!

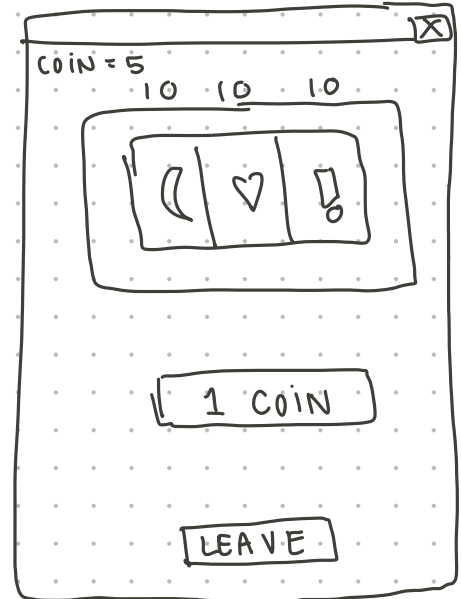
→ Straight forward maybe tutorial  
If we have time?



How many games do we have?

- roulette
- slots
- 21
- dice

→ opens to game

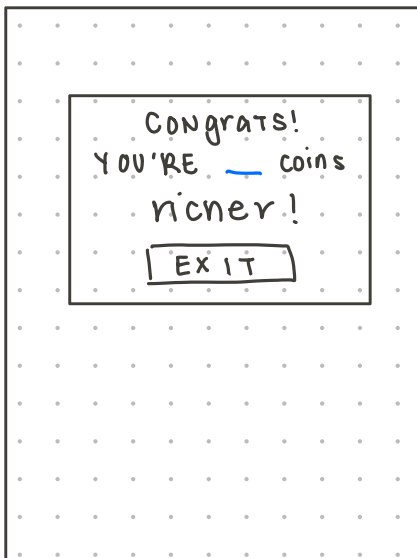


slots

♥ =

- press button  
- subtract coin total  
WINS  
- NOTHING = 0  
- 2 alike = 2  
- 3 alike = 3

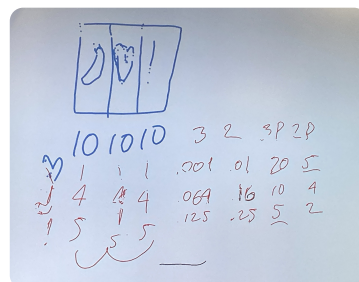
exits back to welcome

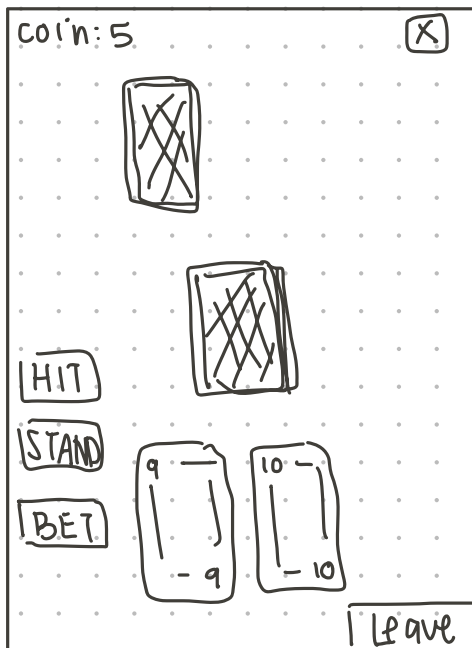


if coins >= 5  
else {  
— coins?  
LUCK NEXT TIME!  
EXIT

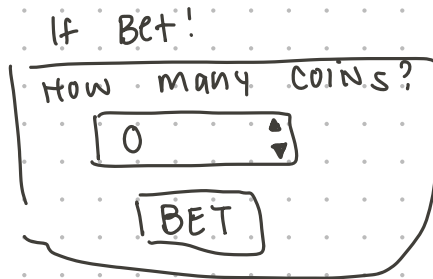
= ♥ ♥ ♥ =  
= ♣ ♣ ♣ =  
= ♦ ♦ ♦ =

|       |       |       |
|-------|-------|-------|
| - 10  | - 10  | - 10  |
| ♥ = 1 | ♥ = 1 | ♥ = 1 |
| ♣ = 4 | ♣ = 4 | ♣ = 4 |
| ♦ = 5 | ♦ = 5 | ♦ = 5 |





- hit, another card
- Bet, disabled until stand

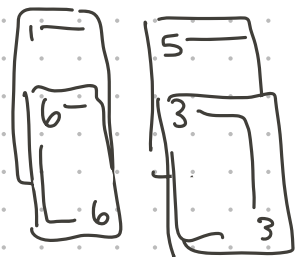


if stand, reveal results

if hit, give player another card

YOU WON 1 COIN!

YOU LOST 3 COINS!



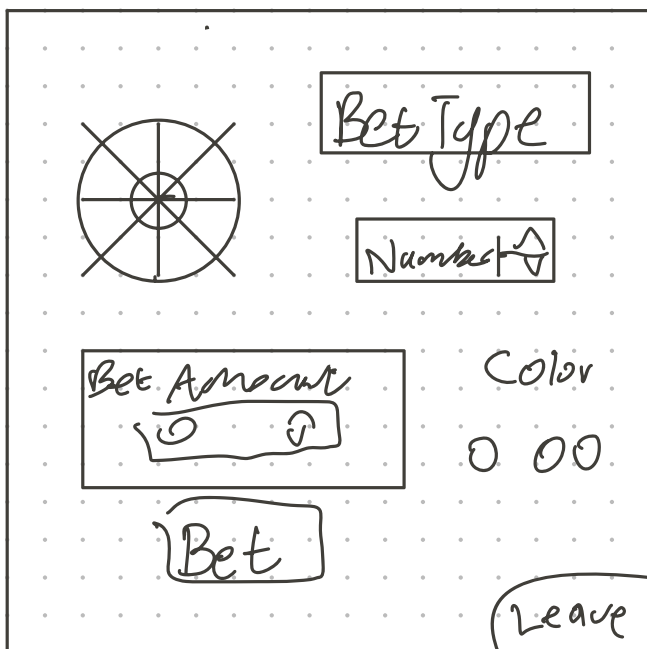
- 21, 2 cards x2.5
- Computer =
- if sum >= 15, hit!

- > each card has value?
- > stored value
- > keep track of value of cards and compare with comp stored value?
- > some sort of alg. for comp.

```
if (comp >= 21) {
  if (comp > player) {
    blackjacklost();
  }
}
```

## Roulette!

Will say Odds



- 1 number: 35:1
- Split (2 #s): 17:1
- Street (3 Row): 11:1
- Corner (4 #s): 8:1
- Basket (1,2,3,0,00): 6:1
- Line (6 #s 2 Rows): 5:1
- Column (12 #s): 2:1
- Dozen (12 #s): 2:1
- Low Third (1-12): 1:1
- Middle Third (13-24): 1:1
- Upper Third (25-36): 1:1
- Red/Black: 1:1
- Odd/Even: 1:1
- Zero Split: 17:1