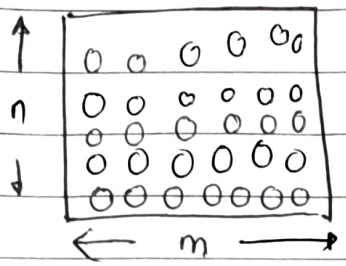
Design Connect Four



Characteristics

- 2 players - (player vs player)
- variable size $m \times n$
- 4 dots diagonally or straight
- Keep Score tracking i.e Target Score

High level

- Grid class - represents state of the grid.
  a slot can be · Empty
  · Red
  · Yellow
  } enum GridPosition.
  → you can also have a design where you don't have empty & just keep null/ Upto us.
- Place piece (Column , which player) piece
- Game class → Grid
  → player {Red or yellow}
  → track of how many rounds red player & yellow player have won.

Design Blackjack

- 52 cards
- Each card has value 2 - 10, J(10), K(10), Q(10), A(1, 11).
- You have to make a total of upto 21. Over it you loose (You will have 2 cards)
- Dealer can also have 2 cards
- Highest score wins. (upto 21)

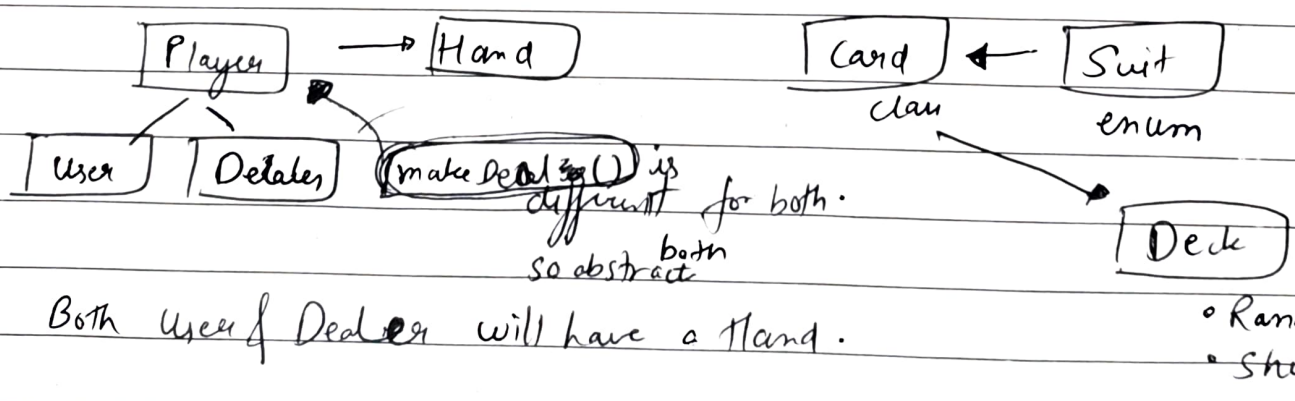
Questions you have to ask

1. How many cards are in a deck. (also shuffle the deck)?
2. What are their values
3. How may decks, What happens when we run out of cards in the deck
4. How many players
5. Score keeping? (Target score)

We are going to design for

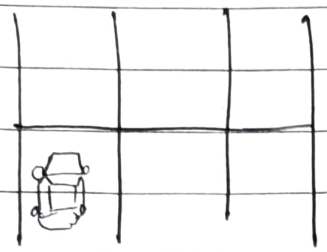
1. We will have to keep score
2. 52 Cards
3. We have 2 players. (Dealer & player)
4. Same as normal card values
5. We will have a single deck & we will refill before every round. and shuffle
6. Player can keep any amount they want
   if player wins, the player will get the same amount from the dealer that they gambled, if they loose they the money will go to dealer. If draw then they don't loose & any money.
7. Dealer will know player's card value. They will try to get more value by keep on drawing the cards.

With your own way for practice



Both User & Dealer will have a Hand.

Design a parking lot



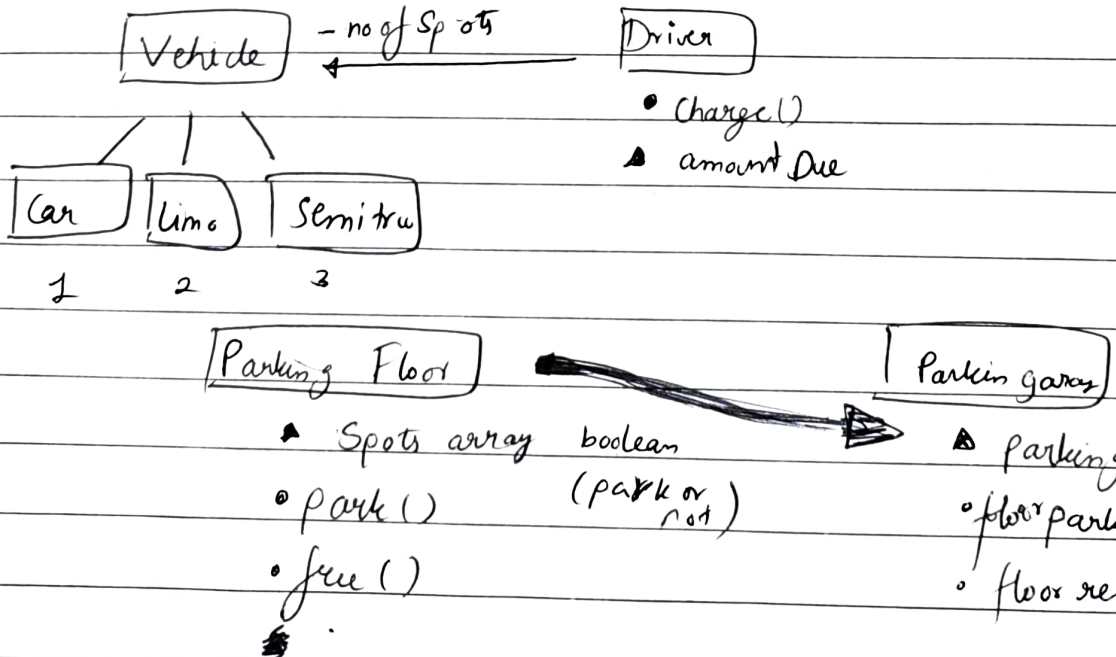
- Usually you will have cars, bikes, trucks (different sizes)
- Parking spots are numbered
- Parking garages are ~~numbered~~ multiple floors
- Parking can be free or charged.

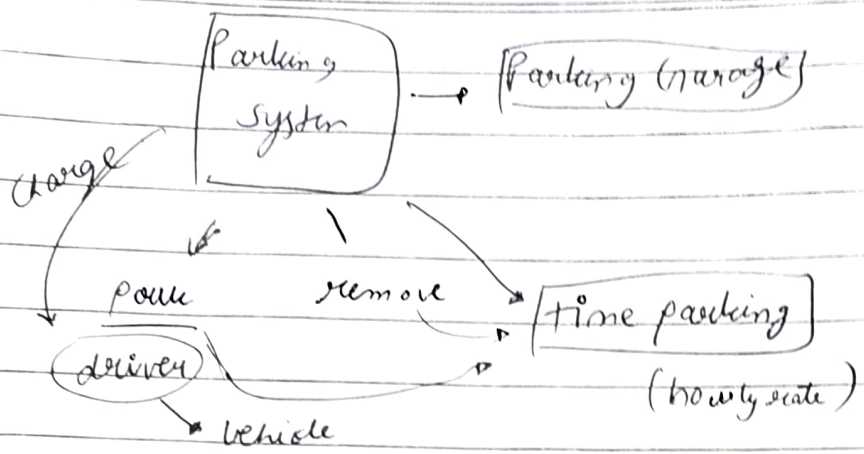
Questions

1. How many floors
2. What kind of vehicles? Designated slots for those
3. Payment - Does driver decide where they want to park or payment system gives.
4. ~~4~~

Ours requirement

1. Multiple floors
2. Multiple vehicle types - Cars, Limos & Semi-trucks
3. We will have normal parking spots but Car takes 1 ps., Limos - 2 ps, Semi - 3 ps (3 continuous ps)
4. Payment system which will take the amount at exit & it will assign the parking spot.





• park = will implement a sliding window to check for no of empty spaces needed for the vehicle.

Netcode design

a. Vehicle

b. Driver

  a. Car

  b. Limo

  c. Semitruck

- ParkingFloor — park, remove
- Parking Garage — array of floors
  — park
  — remove (check every floor)
- Parking System — parking gara
  — hourly Rate
  — time parked.
  - park
  - remove.

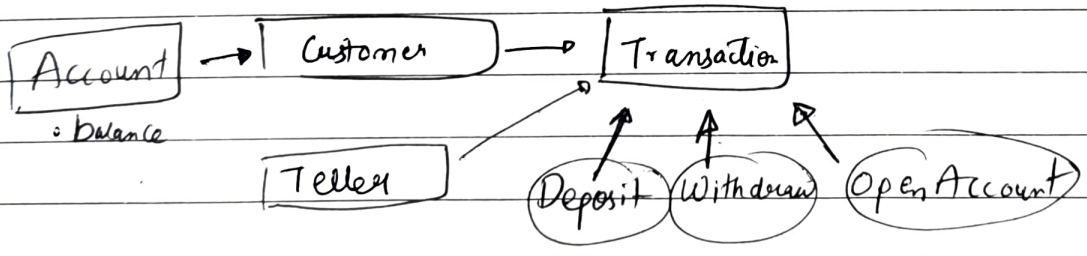
Bank

Questions

1. What kind of Services
2. Do customers have accounts / do they need accounts ? / what types of account
3. Location - physical or only online. / Single branch or multiple branches
4. if physical, will actual people be handling transactions.
5. Security

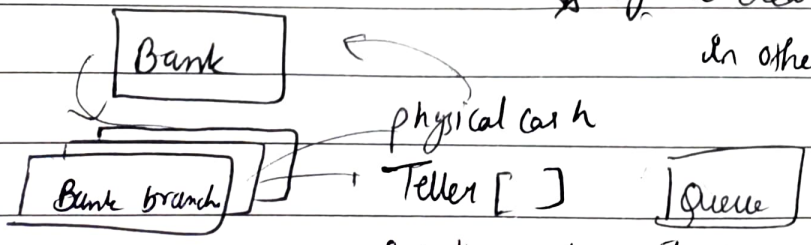
Answers of requirement:

1. Deposits & Withdrawals
2. Yes customers have accounts
3. Physical only with multiple branches. & every transactions has to go through a teller.
4. We have single HQ & branches.
5. Every day we will send a percentage of amount, that will be sent to head quaters.



In this case, Customer is nothing but an account for a bank, So, we don't need customer, as we can store the same data in Account.

If we create an account then that account exists in other branches as well. and Transactions also.



everytime a transaction is there we will take a teller at random to do the processing.

Design Movie Recommendation

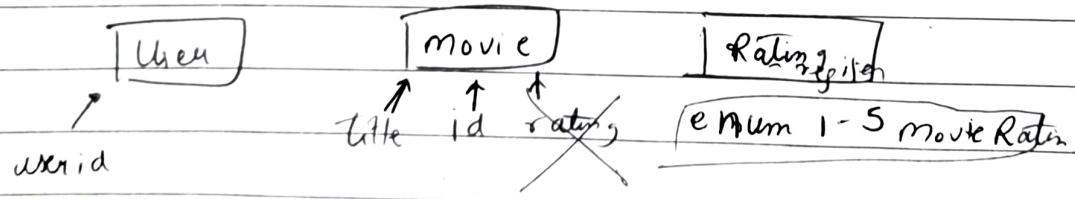
- Something like Netflix or Youtube.

Questions

1. How will we rate the movies.? User?
2. How to Recommend - ie, use the above rating or any other metric ? or any other algorithm.
3. How many users / movies.
4. What if there is a tie in the movie that we are recommending.
5. We recomnd single or multiple ? How to recommend to a user who hasn't seen anything yet on the platform

Answers

1. Star system 1-5.
2. If no data on user - go through all list at get the highest rated
   Known user - 1) Not watched
   2) Based on other user who is most similar to this user and recommend a movie the user 2 has seen to user 1 has not watched yet.



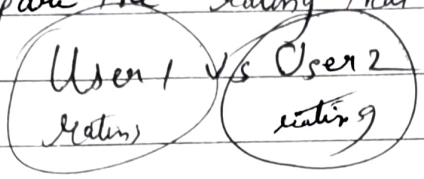
I had added rating of movie into movie class but Neetcode told we will have separate class Rating registration & the design of this will be based on how we design our recommdation algorithm.

Recommandation → Recommend a movie based on User

Rating Register | HashMap <Userid, List<MoviesSeen>>

HashMap <MovieId, HashMap <User, Rating>>

When we want to find for this user, other users with similar interest, for that user, we will iterate through every other user & find the intersection of movies they both have watched and then for that movies using the second HashMap <movieid / hash<user, ra
- we can compare the rating that the user provided



Neetcode has done
Using a similarity score, it is set to infinty and the lower the Similarity score the better matched the users similarities are.

Here it is basic recommendation.
- We can also add more weightage to the ratings of movies that user has watched recently.