Design Notebook

Nick Waggoner
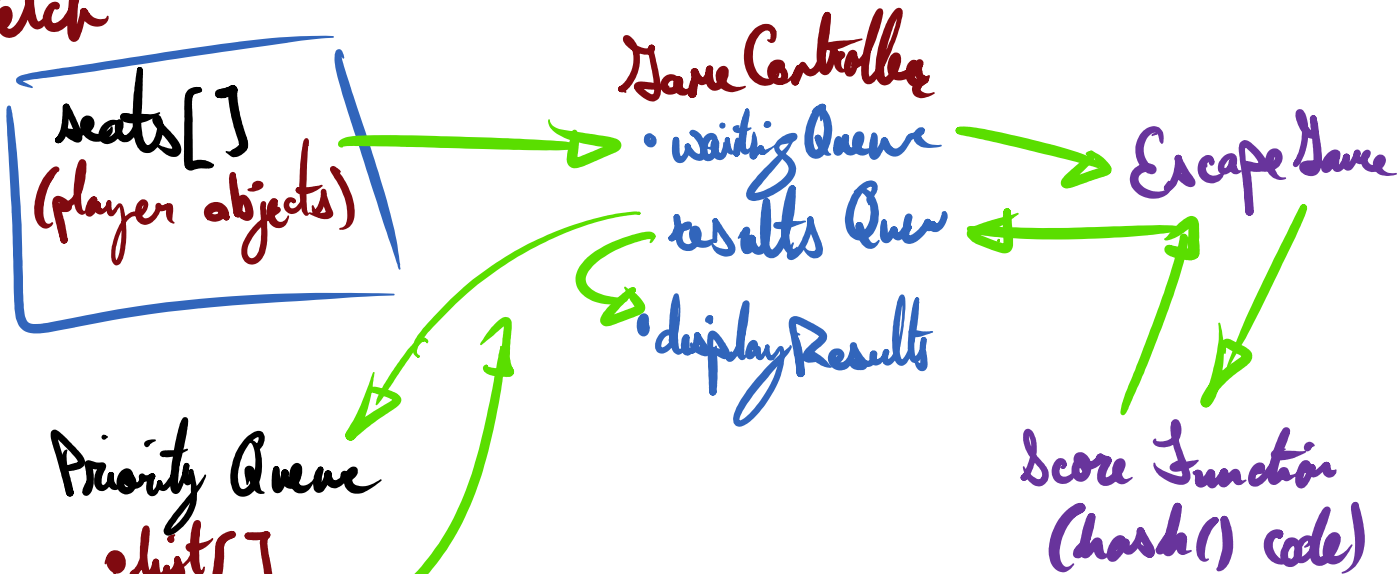CS1450 - Mickey Gonzalez

Problem:

Designed to work with Queues. Read players from a text file into an array. Send those player objects into a waiting list queue which are then taken one @ a time into the Escape Game where they earn a score created using a hash function. Players are then stored in a Priority Queue and sorted by their scores.

Understandings

What I know:
Arrays of Objects + Creating object classes. Stack + ArrayList implementation; very similar to Queues

What I don't:
Creating a Queue or Priority Queue from scratch. Selection Sort coding - "linear" sort.

Sketch

seats[ ]
(player objects)

Game Controller
• waiting Queue
• results Quen
• display Results

Escape Game

Score Function
(hash() code)

Priority Queue
• list[ ]

Priority Queue
• list[]

(hash() code)

selection sorts objects by score

PseudoCode:

Escape Room{
```
    public in hash(key){ //copy text
        creates a "random" integer value by combining player
        name and rank. for the score
    }
    public int tryToEscape(name, rank){
        combine name and rank in string
        hash(combined)
    }
}
Priority Queue{ //all queue methods are similar to ArrayList)
    private void selectionSort( player[] list, numPlayers){
        for( i < numPlayes){
            biggest = list[i]
            int biggestIndex = i;
            for(j = i+1; j < numPlayers){
                if (biggest.compareTo(list[j]) > 0){
                    biggest = list[i] =
```

```
      if ( biggest.compareTo ( list[j] ) > 0 ) {
              biggest    = list [j];
              biggestIndex = j ;
          }
        }
      if ( biggestIndex ?= i ) {
              list[biggestIndex] = list[i];
              list[i] = biggest ;
          }
      }
    }
  }
```

What I learned :

Queues cannot be explicitly initialized
Selection Sort coding
Priority Queues + How they work.