Paul Jackson III and Thomas Braun
Computer Science 112-Final Project Updates
April 8th, 2019

Updates

Generally speaking the game of cup pong stayed the same. The user did still play against the computer and took turns shooting. The user was able to shoot the selected cup of their choice. We felt the gameplay was one of the most important things to keep the same so these elements of the design stayed the same. However, a significant difference between the difference in design was the implementation of a sliding bar that determined the accuracy of the shot instead of the shot being determined by guessing a random number. We figured this would make the gameplay a lot more fun, easy to pick up and aesthetically pleasing. We also decided it would be fun to throw in some different game over screens and a splash screen that would give the game a more game-like feel. However, to shoot and choose a shot we decided to ask the user for a cup based on a number system we printed on the screen instead of a click. This was something that we felt would just work better with our project and the user would not have problems clicking on a smaller surface area of a cup to choose.

As far as the classes the only significant difference occurred in the prototypes game.java better known as our main.java. The cup class stayed the same and had two instances of the rack class, one rack instance for the user and one rack instance for the computer. The cup class was also similar to our prototypes we had 20 instances of the cup class and each cup is given a boolean deeming it alive or dead. However, we strayed from our original idea of a linked list for the cups because we figured it would be less problematic to implement the linked list for cups. We moved the focus of a linked list to other areas of the code specifically creating graphics and aesthetic appeal of the game because for the we didn't need to store anything more than an int. The game class was not implemented at all instead we did a Main-Class which had similarities. Some of these include; generating a number for the computer to guess to make a shot, KeyListener, MouseListener, paintComponent, methods that control turns, and recording made shots. Other than that we were required to create many other methods that would have to handle certain functionalities within the game to help it run smoothly. Some of these include our newly implemented sliding bar speed select and spee methods. Others were our methods that determined when the game was over telling us who won or who lost.