Naomi Ruffalo

CSC 230

4/25/21.

Journal 8

The Game Room client wants a software developed for Draw it or Lose it. The A game consists of four rounds of play lasting one minute each. Drawings are rendered at a steady rate and are fully complete at the 30-second mark. If the team does not guess the puzzle before time expires, the remaining teams have an opportunity to offer one guess each to solve the puzzle with a 15-second time limit. Their software design consists of the following:

* A game will have the ability to have one or more teams involved.
* Each team will have multiple players assigned to it.
* Game and team names must be unique to allow users to check whether a name is in use when choosing a team name.
* Only one instance of the game can exist in memory at any given time.

What I did particularly well in developing this documentation was research outside of my given sources to find out what languages could be used for this project and see if they are secure and create a vivid image for the game.

Some process I found helpful when developing the code was organizing my materials. Keeping track of what I want to accomplish specifically, allows me to create the code in a much simpler way.

I would revisit the design constraints: One design constraint for developing the game application in a web-based distributed environment is security. For Android mobile and application there can be multiple languages that can be used to develop this application. If multiple languages are involved, it rases the risk of security. Specially, if different players are using different operating systems such as: Linux, Windows, or Unix. While developing this application the developers must make sure to use limited and secure languages. Java is a secure and fast programming language perfect for gaming. Along with C++. For the development of the application, the developers should keep the code neat and limited to fulfill all the necessary functions. From this, I would leave the secure languages to the scrum master.

I interpreted the users needs by placing them in a priority list. For example, the design was more important to the Game Room, instead of the security. I made sure design is implemented into the software. It is important to consider the needs of the user because that is what will make the software so successful. How many users play the game and come back to it.