

## Criterion A: Planning

### Description of the Scenario

The yearly NCAA March Madness tournament pits the best college basketball teams across the nation in a 64-team single elimination competition. My family and I are huge basketball fans when it comes to this event, and every March we partake in a little competition of our own to celebrate the occasion. My father, mother, and I all are part of an online league on cbssports.com, where 30 of our close friends and family create brackets predicting the March Madness tournament results. It is a fun and friendly tradition in our household, and one filled with much excitement and smack talk.

While I was initially debating IA topic ideas, I had thought of designing a workout routine for my father, Jesse Lynch, as he was just getting back into running coming off of a heavy calf strain. My father indeed liked the idea; however, during my first meeting with him about potential coding needs, he also brought up the topic of the upcoming 2019 March Madness tournament. As mentioned previously, this family/friend competition is a fairly big deal in our social circle, and every year my father sends out daily entertaining emails updating the March Madness basketball status. He mentioned to me that creating these emails is quite a lot of work, as he must search up the details of every single game for each update. He also admitted that, while he loves to include random fun facts in the emails, it becomes tiresome trying to remember all of the information for each individual team. The first meeting concluded with my father stating that if I could design something to help organize information for his emails, that would be preferred.

Two days later, I spoke again with my father and agreed to help him with his proposed concept of organizing the information for his March Madness emails. He thus will be my client for this Internal Assessment, while my advisor for the actual coding of this project will be my computer science teacher, Mr. Donnelly. 334

### Rationale for Proposed Solution

Due to the fact that this proposed program will only really need to be accessed by my client, a basic Java application program should fit his needs. The March Madness game and team data can be found easily on the internet, and I will use File I/O to store this information before transferring it into the program. From a technical stance, (my advisor and I have agreed) that I will indeed have the programming skills to execute a program of this stature, without the need of much assistance. Ergo, this assessment topic is appropriate for my current coding abilities.

In addition to providing me with an opportunity to improve my programming, this Java program will be undoubtedly helpful to my client, which in turn is beneficial for me. The complex yet quirky emails that my father works so hard to create every year is part of

what makes this tradition so entertaining, and, for many of our friends who don't watch the games, these emails are the only source of information for the basketball tournament that they receive.

## **Success Criteria**

During the February 12<sup>th</sup> and 14<sup>th</sup> meetings, my father and I decided on the following success criteria for the March Madness program:

- Menu system that lets user choose either to update the March Madness game data or view the printed statistics
- Blank bracket that can be updated the game winners are decided
- Store NCAA game results as they are played
- Subset menu system that organizes the statistics to be printed in an organized fashion
- Printout NCAA game results
- Printout the number of upsets (lower seeded teams winning)
- Print in a table the quantity of number of wins based on team seed number
- Record whether a certain team has a humanoid or animal mascot
- Print out winning percentages of animal vs humanoid mascot games
- Store data regarding college tuition and college size
- Print out game results where smaller and/or less expensive schools won
- Store the latitude and longitude of each college
- Utilize File I/O so that all of the stored data can be re-accessed when the program is launched
- Option to delete entire database

Words: 512