**Introduction**

This is the implementation of coding challenge, I took for a development job at a gaming company.

**Specification**

For your coding challenge, you will create an achievements system for a fictional online game.

Our fictional game is played online between two teams. Each team can have 3-5 players (but must be evenly matched – i.e., if one team has 4 players, the other must have 4 players). Statistics are compiled for each player during/after each game:

• Number of attempted attacks

•

• Number of hits

•

• Total Amount of damage done

•

• Number of kills

•

• Number of “first hit” kills

•

• Number of assists

•

• Total number of spells cast

•

• Total spell damage done

•

• Total time played

•

**Statistics**

In addition, historical player statistics are maintained as well. For instance, each player is credited with a win/loss based upon the performance of his/her team during the game. Other historical player-level stats can be tracked as well – such as total number of games played, total duration of games played, total number of kills, etc.

Achievements are awarded based upon any combination of these statistics (historical OR game-specific). Players collect achievements over time, and these will be displayed on their profile page (the display is outside the scope of this exercise). This profile page is visible to all other players, so achievements act as a kind of “badge of honor”. After each game has ended, the achievements logic is called and achievements are awarded to each of the players if applicable.

Your mission, if you choose to accept it, is to create this achievements system.

**Expectations**

So, here’s what’s expected from this coding exercise:

• Code should be written in Java.

•

• The achievements system that you are creating should correctly assign achievements to each player involved in the game that just ended, if applicable.

•

• Persistence, messaging, and other potential concerns are considered ancillary to this exercise, so don’t worry about implementing such concerns.

•

• Create all domain objects suggested in the above description, using the proper data structures where applicable.

•

• Create the achievements system. At a minimum, it should have an entry point that would be called at the end of a game. You must include the following achievements:

•

1. “Sharpshooter Award” – a user receives this for landing 75% of their attacks, assuming they have at least attacked once.

2. “Bruiser” Award – a user receives this for doing more than 500 points of damage in one game

3. “Veteran” Award – a user receives this for playing more than 1000 games in their lifetime.

4. “Big Winner” Award – a user receives this for having 200 wins

• Your system should be extensible in the following ways:

•

1. The system must easily handle the tracking of additional statistics at any level listed above (historical by player or game-specific by player)

2. The system must easily handle the addition of new achievements

• Add a new statistic, and create a new Achievement utilising this statistic in conjunction to one of the other statistics defined above.

•

• Create unit tests to demonstrate that your code is functional using JUnit .

•

• Create a driver class that sets up sample data and calls the achievements system, printing out achievements to the console.

•

**Persistence**

Don’t worry about persistence, and use any method of creating/loading sample data that you are familiar with. Also, you can combine/integrate this into your unit tests if you prefer – we’re just looking for you to demonstrate that the code runs.