# IBM – Poker Game Challenge Summary

## Challenge Description

IBM has given me a challenge as to build a web page that evaluate two poker hands and declares a winner among the two. Using any of the technology, development framework that is available. For instance, Groovy/Grail, Java, ASP.NET, PHP…
The goal of the solution was to give a user option to pick cards for 2 players from a standard deck of cards, omitting the jokers. When the user click on the Evaluate button, all of the information on the page will be sending back to the server for evaluation. Winner will be displayed below that evaluation button when available.

## Project development

The following will show that progress and development of the project over the course of two weeks, using Sprint as milestones/references:

### *Sprint 1*

For the first Sprint, I set a goal as to create created the UI view. That includes five drop down lists for each player, player text boxes and an Evaluate button. In the addition some back end set up for challenge. Using HTML and some PHP, I was able to set up the UI view in a short time. I had a little trouble coming up the functionalities behind the scene. I research a little more of what other options I could do this challenge that would support object oriented and web application. I saw ASP.NET was one of the options that I could do the challenge in. I did research on it as well as watching a lot of Youtube videos on what and how to do something in .NET framework. I downloaded and installed Visual Studio 2015 Community, and code the challenge in C# and HTML. My code from the prior project was coming in handy. Most of the code was transferable. HTML code for the UI view was exactly the same. Most of Java code was very similar to C#. It was easy to transfer them over. Visual Studio supports so much that I don't have to install a third party application to emulate the server. Visual Basic let me use me computer as a localhost. Most functionalities for the Challenge was completed in this Sprint.

### *Sprint 2/Final Sprint*

At sprint two the goal I set was to have the application fully working and giving the correct winner. After have most of everything setup in the previous sprint at this Sprint the tasks seem to be easy to manage. I implemented functions in each object to carry out a task. I used them to my advantage. Things went very smooth at this Sprint. I was able to have the application working sooner than predicted. I moved on and started working on the optional tasks. Validation players' cards was not very difficult to do, since the cards were objects and stored in each player object. Couple methods were able to do the trick. Randomizing a game was added in as well. Using JavaScript and randomizer function provided in the Math library and attached to a button, I, now, have button to randomizing players' hands. Scorekeeping was a little hiccup for me. Since I was not very familiar with .NET, MVC or C#, I had to do some research and asking people on how to save value throughout time, not lose it every time user click Evaluate button and increment it on player wins. I found a function/variable call Session that keep the player score throughout session. After that everything came easily. Mandatory features and optional features are done on this Sprint.

### Final Solution

I have been able to complete the challenge within the allowed time frame as well as the extra features that had in the requirement document. The web page now has five option boxes for each player, and two buttons for user to interact. Each option box has 52 options correspond to 52 cards in a standard deck. A Randomize Hand button will give both players random cards. The server now was able to determine what hand each player has, validate cards for each player and both players, determine who wins the game and keep score for each player. There are still some more functionalities to add to the application, but the time may not allow me to do so and that would go out of scope of this project. As a result, the application has more room to grow and expand with potential features in the future.

### Potential Future Features

- Display image of card in the UI when user pick a card
- Allow more people to be in one game
- Player can share up to 3 cards like a real poker game
- Better graphical presentation