CST236 – Assignment 2

Wilsonville, Spring 2020

# Goal

Create a pipeline in Azure DevOps that builds, unit tests, integration tests and deploys a web application to play a 2-person game of poker.

# Steps

## Step 1: Understand the rules of the game

1. Read about Five Card Draw: <https://en.wikipedia.org/wiki/Five-card_draw>
2. Read about poker hands: <https://en.wikipedia.org/wiki/List_of_poker_hands>

## Step 2: Implement a poker class library

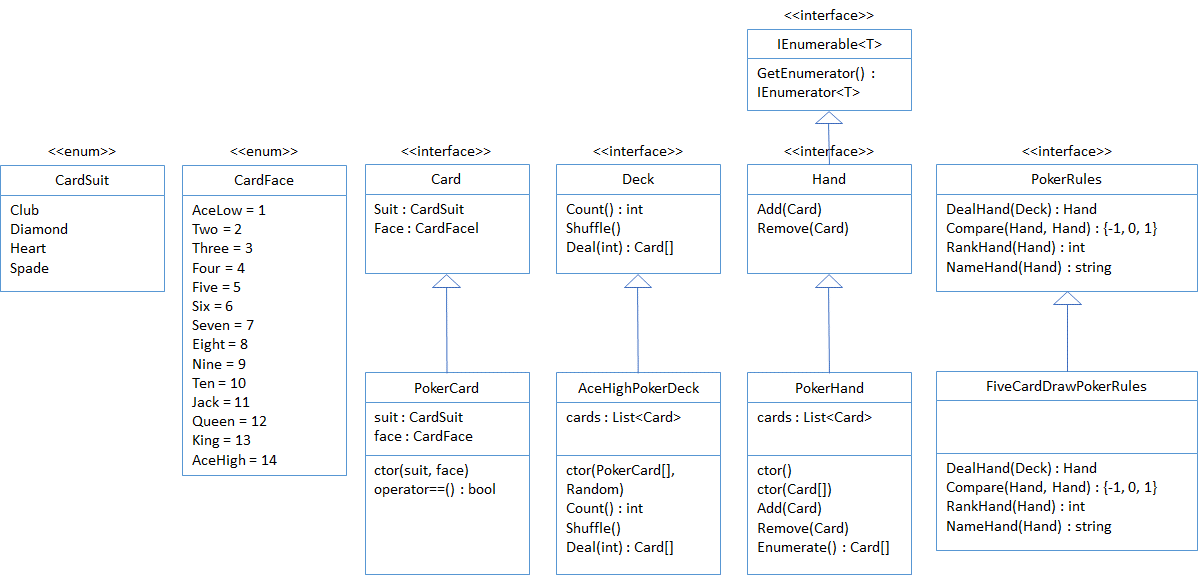
1. Create a class library called PokerLibrary and implement as follows…

PokerLibrary includes the following enums and interfaces:

* CardSuit
* CardFace
* Card
* Deck
* Hand
* PokerRules

And the following classes:

* PokerCard
* AceHighPokerDeck
* PokerHand
* FiveCardDrawPokerRules



1. Create unit and integration tests to find and remove bugs in the library

## Step 3: Create a web application to play Five Card Draw poker

1. Create a new ASP.NET Web Application called FiveCardDraw
2. Use the MVC template
3. Check the box for “Also create a project for unit tests”
4. Implement the simplest version of 2-play five card draw…
   1. Players connect, enter their name and are give $100
   2. Start a game…
      1. Choose Host New Game and wait for 2nd player, or
      2. Choose another player’s game who is waiting
   3. Player 1 always goes first and chooses Deal to start
   4. 5 cards are dealt to each player
   5. Cards are shown to the player, along with the name of the hand
   6. Bet some $ or fold
   7. Select zero to fie cards to turn in for new cards
   8. Receive replacement cards and see name of new hand
   9. Bet some $ or fold
   10. Showdown! Players see each others cards and hand name
   11. Game indicates winner and distributes $ winnings
5. Everything is in memory: no database or files
6. Make use of the

## Step 4: Create a pipeline in Azure DevOps

1. Use your FirstProject in Azure DevOps or create a new project
2. Make sure to add [peter.myers@oit.edu](mailto:peter.myers@oit.edu) to the project
3. Add your library, web site and test code to your code repository
4. Create a pipeline for a web application
5. Ensure the library and web app are…
   1. Built
   2. Tested (unit and integration)
   3. Deployed to a test environment

## Step 5: Turn in your work

1. Make sure all code is committed
2. Label your repo with an official build version
3. Create a text document (\*.txt) with the following info
   1. Your name and the date
   2. The name of your Azure DevOps project
   3. The name of your Visual Studio solution containing the library and web site
   4. The name of your pipeline
   5. The name of the label applied to the repo
4. Upload your txt document on Canvas to submit your assignment

Note: The instructor should be able to pull your code and see everything performed in your Azure DevOps project