awstreams
technology innovative solutions

Assessment Development

Test 1. Poker chance calculator

Description:

Web application should calculate chance of getting desired card based on the amount of cards left.

Use string notation to designate cards.

Example:

H2-H10 - cards 2-10 of hearts

HJ - jack of hearts

HA - ace of hearts

SJ - jack of spades

DJ - jack of diamonds

CJ - jack of clubs

[suit][value]

Step 1. User should select a suit and a card rank.

Step 2. User starts drafting cards, one by one.

Step 3. Website should display a chance of getting customer selected card on the next Draft.

If customer selected card is drafted website should display popup with a message "Got it, the chance was

(current chance of getting the card)%" and reset to step 1.

Guidance steps:

- create an MVC application. Follow SOLID principles in your work.

- create a model that will generate a proper poker deck in random order. (52 cards, excluding jokers).

- assignment doesn't require any javascript. Reload the page as much as you need.

- use a framework if needed. using composer is a plus

- write a phpunit test to test chance calculation and data generator logic. [plus]

Test 2: Phrase analyser

Description:

Create a web application that will analyses customer input and provide some statistics.

Run flow:

Step 1. Customer is asked to insert a string (no longer than 255 chars)

Step 2. Customer submits the data and receives a grid overview with character statistics.

column1 - character symbol

column2 - how many times character encountered.

column3 - sibling character info: character was seen standing before [list of chars], after [list of chars], longest

distance between chars is 10 (in case of 2 or more characters).

Guidance:

- use framework if needed. using composer is a plus

- Using Db is not required in this assignment

- Assignment should be implemented using objects arranged into graph

- Write a phpunit for function that will traverse the graph. [plus]

Example:

String 'football vs soccer' should output:

f : 1 : before: o after: none

o : 3 : before: o,t,c after: o,f,s max-distance: 10 chars

t : 1 : before: b, after: o