Just Say the Word Game - Technical Guide

INDEX NUMBER: AS2020978
NAME: L.K.I. SILVA

Introduction

This is a game which the user can gain fun with the knowledge. Simply What happen in this game is user can guess the word by going through multiple hints.

So first the system will ask for a random number to enter by the user. User can enter a number within the range of 1-15(Developed using 15 words only. The no. of words can be updated later).

When he enters a random number, user can see the count of letters in the word which the user has to guess.

After that, it will show a hint to guess the correct word. The User can enter the word after every hint.

First user will be given three hints to guess the correct word. The score of the user will be 40 for the first three guesses. But if the user failed to enter the correct word after first three hints, he will be given another three hints to guess the correct word.

But for each extra hint the score of the user will be deducted by 10 marks.

If the user fails enter the correct word even after all the extra hint, the score of the user will be shown with the correct word.

If the user entered more letters than expected number of letters of the word it will be informed to the user.

Similarly, if the user entered less than the expected number of letters of the word it will be informed to the user.

If the user entered same no. of letters as the correct word but with different letters it will also be informed to the user with the count of the wrong letters which the user entered.

Architecture

There is a flow chart attached to the folder So that you can get clear idea about the system how it going to work.

Logics used in the game

In this game I have used multiple one-dimensional arrays to store the words, hints, and extra hints. First, I defined a function called ‘StartGame’.

Then starts the main function. In the main function I initialized the variable ‘intRandomNumber’

Then give an output to the user to enter a random number and scan and store it for the future purposes.

Line 22: Then function calling takes place. In the function 'StartGame' first I initialized the variables which will use in the future.

Line 36-41: Then I initialized three string arrays named strWords, strFirstHints, and strExtraHints.

Line 82: To get the no. of letters in the word which the user has to guess, I used the strlen() function. I equalize it to the variable intWordLength to use it later.

Line 83: Next I got intHintProgress as (intRandomNumber-1)*3, because for a one word I use three hints in the array strFirstHints. So, I must skip some hints to get the exact hint. Here I used intRandomNumber -1, because in the array the index starts from 0.

Line 89: After that user can see the no. of letters of the word. Here I used previously calculated intWordLength variable.

For loop.

Line 97: Then it start to show the hints to the user. For that I used a for loop. I used the count variable for the number of occurrences of the loop. Here count starts from the intHintProgress variable because the hint must show relevant to the index of the word which the user enters to intRandomNumber.

Then starts to show the hint to the user. Here I used strFirstHints[count]. Because count is the iterator of this loop. After the first occurrence the count will be intHintProgress plus 1. So, after first hint it will show the next hint stored in the array. Like that It would show three hints if the user failed to enter the word in first two chances.

After each hint user has a chance to guess the correct word. When he enters the word it will store in the variable called strUserWord. And also, the length of the user entered word also calculated and store in the variable intUserWordLength.

If else-if ladder.

Line 106: Then there is an if condition to check whether the word guessed by the user correct or not. First, it checks whether the entered word is exactly the correct word. If yes, then the user can see a message that "You Guessed the Correct word" with the score of the user. The score of the user is stored in the variable intFirstScore. First It is initialized to 40.

If the word entered by the user is not correct, then it will go to the second if condition. In the second if condition it checks whether the word entered by the user has more letters than the expected no. of letters. If yes then the user can see the message "You have Entered more letters than the expected No. of letters".

If the word entered by the user has less than the expected no. of letters, then it will be informed to the users as "You have Entered less than the expected No. of letters".

If the no. of letters entered by the user is equal to the expected no. of letters but letters are incorrect, it will be also informed to the user as "You Guessed it wrong".

For loop.

Line 134: Here I used a for loop to check the word letter by letter. I used check as no. of occurrences in the loop. And for the condition test I used the check is less than the length of the correct word. Here you can see that the check variable is starting from 0.

If condition.

Line 137: Inside for loop I used an if condition to separate the wrong letters from the correct letters. Here if the letter in the strUserWord does not equal to the letter in the strWords(correct word), it will goes inside the block. I used the check for the letter as it will iterate within the for loop. Inside this if condition I assigned intWrongLetters to increase by one in each time. So, when the loop terminated, we can get the no. of wrong letters in the word guessed by the user.

After loop termination the user can see the no. of letters wrong in the guessed word as "***There are "xx" letters wrong in your guessed word***". And he can again try again with this idea.

If the user failed to enter the correct word even after the first three hints, he can get three extra hints to guess the word.

If condition.

Line 150: For that purpose, I used an if condition. In this case if the user already entered the correct word in the previous chances, I don't have to show the extra hints. And also, it should start after third hint. So, I used logical "AND" operator to satisfy t both conditions here. First, I check whether the count has iterated three occurrences.

Then I check whether the word guessed by the user at the last occurrence is correct. So, if both conditions satisfied, the user could have extra hints to guess the word.

Line 155: Inside this condition first, the user will get informed that his score will be deducted 10 marks for each extra hint.

For loop.

Line 160: Then I used for loop to give Extra Hints for the user. Here I used the counter for the no. of occurrences in the loop. Here also the counter starts from intHintProgress variable because the hint must show relevant to the index of the word which the user enters to the intRandomNumber. Here it does not need to change the initialization, condition test and the increment because I used it for a different array called strExtraHints.

Line 163: Inside this loop first intFirstScore will be decreased 10 from its original value.

Then the user can have other three extra hints from the strExtraHints array. Here the index used here is counter variable. Counter is equal to the intHintProgress and it will incremented one by one in each occurrences.

Then user will be given the chance to guess the correct word. It will scan and store in the variable strUserWord.

Then the length of the user entered word will be stored in the intUserWordLength.

If else- if Ladder.

Then the word guessed by the user will be checked in the if condition ladder. First check whether the entered word is exactly same as the correct word. If yes it will be informed to the user with the score.

Line 179: If the no. of letters entered by the user is more than the expected no. of letters it will also be informed to the user in the next if condition.

Line 185: If the no. of letters entered by the user is less than the expected no. of letters it will be also informed to the user in the next if condition.

Line 191: If the no. of letters entered by the user is same as the expected no. of letters in the correct word but the letters are incorrect then it will also be informed to the user.

For loop.

Line 201: To separate the no. of wrong letters from no. of correct letters entered by the user, this for loop will be used. As I explained before the initialization, condition test and increment counter will be same as in the previous loop.

If the letter in the strUserWord does equal to the letter in the strWords of the same position, then it will go inside the loop and intWrongLetters will be incremented by one. After termination of the loop, it will show the incorrect letters in the guessed word.

Line 222-224: If the user fails to enter the correct word after those extra hints, then his score and message that he failed to enter the correct word will be informed.

And the correct word will be informed to the user.

Limitations

- If the user entered word has some spaces, it will not be recognized by the system.
- If the user entered a character to the random number, it would not recognize, and the loop will be continued as the system is case sensitive.
- If the user entered capital letters it will not be recognized as a correct word because the system is case sensitive.

For further details

Contact me:

- Kushani Silva - Game Developer
- Email Address – as2020978@sci.sjp.ac.lk

Hope you enjoy the just say the word game…!