**Design Idea For Functional Paradigm:**

Array of level’s words

Function – choose random word

Function – check and display letters/ \_

Function – hangman game

Function – check if game finished

Main – selection of levels

**Beginning Of Development:**

A screenshot of a computer

Description automatically generated

I started by creating switch cases for the level chosen by the user. Tested switch case and it works successfully.

A screenshot of a computer

Description automatically generated

I started by creating the word arrays holding the words for each level.

A screenshot of a computer

Description automatically generated

I was unsure of how to create a random number generator, So I asked replit AI for help.

A screen shot of a computer code

Description automatically generated

I was then able to make a choose random word function. I was able to make a function that takes in the parameter of the list and uses it to choose a random word. This makes the code more efficient as I am not writing the code multiple times.

A screenshot of a computer screen

Description automatically generated

I then added call function to each case with the parameter of the level’s words array and then tested. As shown in the console, they all work and outputted different words for both tests.

A screen shot of a computer program

Description automatically generated

I attempted to write my own function of checking if a letter is in a word and then displaying it.

A screenshot of a computer

Description automatically generated

There were some errors, so I got help from replit AI to try and fix them so that my function works.

A screenshot of a computer screen

Description automatically generated

After using AI to help fix the errors on line 25 and 26, it just kept creating more and more errors, so I decided to redo the function.

A screen shot of a computer program

Description automatically generated

This new function loops through every letter in the word and then loops through every letter in guessed letter and compares and outputs letter if they are the same, otherwise outputs \_.

A screenshot of a computer game

Description automatically generated

There is an issue with this. When I started testing, I counted the \_ outputted and it is not the same as the amount of letters in the word. Machine has 7 letter and 7 \_ were outputted, serene has 6 letters and 5 \_ were outputted, green has 5 letters and 7 \_ were outputted.

A screenshot of a computer

Description automatically generated

I added a print statement within the function to see what the word being outputted was, and the 2 words were completely different.

A screen shot of a computer program

Description automatically generated

The first print calls the chooseWord function which chooses a random word, and the second print also calls the chooseWord function to choose another random word to be used to output the display. This is where the confusion came from as I didn’t know which word the display function was using hence why there was a different number of letters outputted compared to the word.

A screen shot of a computer program

Description automatically generated

I am writing a function to play the hangman game that takes in the random word and max tries for now.

A screenshot of a computer

Description automatically generated

I tested the play hangman function and the test was successful, it calls the check and display and it outputs the number of attempts.

A screenshot of a computer

Description automatically generated

I have created a for loop within the hangman function, so the game is continuous unless broken out of. I took in inputs from user, and I tested to check that the input was being scanned and stores.

A screenshot of a computer screen

Description automatically generated

I added a for loop that loops through the letters in guessedLetters and compares it to the guess, if letter is in the array then message outputted otherwise it is added onto the array, as shown in the console, I inputted a twice and an error message was outputted the second time.

A computer screen shot of a code

Description automatically generated

I added a function that checks if the letter is present and outputs a Boolean value if found or not.

A screen shot of a computer program

Description automatically generated

I added the guessCorrect function into the playHangman function and I tested this. On the console the correct works and the incorrect works. I found a problem, which is that the game still took in the previously guessed letters and if they are wrong the tries decrease. I inputted d twice and both time a try was taken off.

A screenshot of a computer screen

Description automatically generated

I solved the previous issue by moving the code into the else section of the if statement checking if a letter was already guessed. I tested this and it works, it stops the user from double guessing a wrong letter and losing another try.

A screenshot of a computer

Description automatically generated

I added an if statement to check if the tries has reached the maxTries, If yes then an output message is shown and then loop is broken.

A screenshot of a computer

Description automatically generated

I created an if statement that compares the display function value and the word, if they are the same then the word has been correctly guess so a message it outputted and the loop is broken.

A screenshot of a computer

Description automatically generated

I wasn’t sure how to add a timer so I asked AI for some help. I will be using the parameter shown as a parameter for my play hangman function. I will also use the call function within the main.

A screenshot of a computer

Description automatically generated

I am going to use the time.Since(startTime) to find the difference between the start time and current time to work out if the user has reached their time limit.

A screen shot of a computer program

Description automatically generated

I used the timer as a parameter for the function. I also have a variable to store the initial beginning time. I then create and assign timeRemaining variable with the difference between the timer (max amount of time for game) and the time since the initial timer. I then compare time remaining with 0 to see if the user has ran out of time, if yes then a message is outputted and the loop is broken.

A computer screen shot of a program

Description automatically generated

I call the play hangman function and pass the appropriate word, number of tries and the length of time for the levels.

A screenshot of a computer program

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generated

I tested the time for all the levels, and it works, when the time is over and the user inputs, a game over message is outputted.