**GuessTheWord**

Done By: Sabah Mawani, Class: 13-4

**Index**

1. Synopsis ………………………………………………………….…. 3
2. Problem Definition……………………………………………….…. 4
3. Software Specification ………………………………………….…... 6
4. System Design …………………………………………………..….. 8
5. Testing ………………………………………………………….…..10
6. Implementation ……………………………………………………. 13
7. Conclusion ………………………………………………………… 22

**Synopsis**

The program on the ‘GuessTheWord’ is a computerized version of the game Hang Man, which has been popular among kids since 1984. This project has been initiated with the object of bringing joy to children when isolated, such as the global lock down in the year 2020.

If the user chooses to play the game, he/she will be asked their name and be given a choice of categories. Once the user has chosen a category the program will randomly choose a word for that specific topic, and the game will begin.

The program can be used on Windows, Linux and Mac-OS. It has been developed using Python 3.7.3 with WPS excel as background.

**Problem Definition**

The program on the ‘GuessTheWord’ is a computerized adaptation of the children’s game Hangman, which was invented in 1984. The user is first asked to enter his/her name after which they are given the choice of choosing a category from the following:

1. Movies
2. Books
3. Animals
4. Colors
5. Countries
6. Rock Bands

The user enters their choice by entering the corresponding number of their choice. Their choice is validated by first checking if the choice entered is within the range of one to six.

The program then generates a random number from one to twenty and stores it as the ‘index’. The program then accesses the background WPS excel file and chooses the word corresponding to the value stored in the ‘index’. The user is then shown a set of blank dashes that match the word or phrase. He/she then has to guess these letters by inputting them using their keyboard.

The user is asked to enter their guess, which is validated by first checking if the player has entered only one letter or a couple of letters by checking the length of the input. If the length exceeds one, then the user is told that they should enter only one letter and they are asked to enter their guess again. The second level of validation is if the value entered by the user is actually an alphabet or not, if the value entered is not an alphabet, then the player is told to enter an alphabet again. The third level of validation is whether or not the letter guessed has been entered before, if it has then the user is told to enter a different letter. After these three levels of validation the guess is then sent forward.

If the player enters a correct letter, it is revealed and the game continues, however if the player guesses the wrong letter, a message will appear telling them that their guess is wrong and their number of chances, which are set as ten at the beginning of the game, is reduced each time the user guesses the wrong letter.

If the player guesses the word before their chances run out a victory message will appear and the user will be given a choice, whether or not they want to continue playing the game. If the player does not guess before the chances reach zero, a message of defeat will appear along with the revelation of the word they were guessing.

They will then be given the choice whether or not they want to continue playing. When the user enters their answer if it is anything other than ‘y’ or ‘n’, they will be told that it is an invalid input and they will be asked to input their answer again. If they answer ‘y’ then they are given the choices of topics again and they can continue playing. If they answer ‘n’ then the program ends and the game is closed.

**Software Specification**

Python is an easy to learn, dynamic programming language. It has efficient high-level data structures and a simple but efficient approach to object-oriented programming. Python’s refined syntax and dynamic typing, together with its interpreted nature, makes it an ideal language for scripting and rapid application development in many areas on most platforms. It also has a comprehensive library.

IDLE is Python’s Integrated Development and Learning Environment.

It has many advantageous features from which a few of them are listed below:

* It is comprehensively coded in Python, using the tkinter GUI(Graphical User Interface) toolkit
* It also has the cross-platform component which particularly works the same on Windows, Unix, and Mac-OS.
* Furthermore it has an interactive interpreter with many convenient features such as: color coded input, output, and error messages.
* Moreover it has multi-window text editor with:
* Multiple undo
* Python colorizing: IDLE defaults to black on white text, but colors text with special meanings such as built-in class and function names, strings, comments, and names following ‘class’ and ‘def’.
* Smart indent: after a block-opening statement, the next line is indented by 4 spaces. The Tab key inserts indentation, of which the width is currently limited to four spaces in accordance with the Tk limitations.
* Call tips: is shown when one types ‘ (c’ after the name of an accessible function.
* Auto completion: these are supplied for functions, classes, and attributes of classes, both user-defined and built-in.

IDLE has two fundamental window types, one being the Shell window, while the other one being the Editor window. It is possible to have many Editor windows working simultaneously. This feature has its own top menu on both Windows and Linux.The Python Shell window enables one to enter, edit, and recall complete statements. IDLE may open editor windows when it starts, depending on the settings. Even though many Editor shells can be opened simultaneously, there can only be one open editor window for a given file. The title bar consists of the name of the file, the full path, and the version of Python and IDLE running the window. IDLE considers all files with a known .py extension contain Python code, while the rest of the files do not.

The tkinter package is the standard Python interface to the Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, as well as on Windows systems. Tk itself is not part of Python, it is maintained by ActiveState.

Pandas is a Python package providing swift, flexible, and expressive data structures designed to make working with data easy. It aims to be the essential high-level building block for doing practical, real life data analysis in Python. When working with tabular data, such as data stored in spreadsheets or databases, Pandas helps to explore, clean and process data. Moreover it aids the integration with many file formats or data sources. The two principal data structures of pandas, Series (1D) and DataFrame (2D), handle the vast majority of typical uses in Finance, Statistics,and many fields of Engineering.

The openpyxl is a Python library created to help users to read and write Excel files from 2010 and onwards. It is often used to extract valuable information from spreadsheets using Python, which helps create programs to solve real life database problems. Furthermore, it helps create personalized and complex spreadsheets, with the ability to add on charts and conditional formatting. This library is not only helpful on the school level but also in the business sector. It has the ability to read, write, append, and edit spreadsheets.

The time module in Python helps in creating realistic programs and is very helpful when creating a code for a game in Python. It helps to cause time delays, and use real-time in programs, which is effective and extremely desirable in some cases.

The random module implements pseudo-random number generators for various distributions. This module can be used to generate random integers, sequences, etc. The limits/range can be set when using the function. This is very useful when creating games which involve guessing.

**System Design**

A Gantt chart is a horizontal bar chart developed as a production control tool, it is frequently used in project management, a Gantt chart provides a graphical illustration of a schedule that helps to plan, coordinate, and track specific tasks in a project.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Identify Requirements |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Produce design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Write Program code |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Add Data Files |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Test Modules |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Write Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Day Number | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |

Below is the Gantt chart created for the program on “GuessTheWord”, where the cells shaded in green show the day number.

A Jackson Structured Program (JSP) diagram to represent the program’s modules and how they are linked to each other. JSP uses a diagramming notation to describe the structure of inputs, outputs and programs, with diagram elements for each of the fundamental component types. A simple operation is usually represented by a box, while a sequence of operations are represented by boxes connected with lines.

The sequence followed by such diagrams are up to down and left to right. The ‘\*’ sign represents iteration, where the module with the ‘\*’ sign will keep on iterating until the program is exited by the user. The ‘°’ represents selection, this helps in selecting only one option from various options that are given.

In the diagram below the program on the ‘GuessTheWord’ is represented, the sequencing is as followed:

1. The game is run on the system.
2. The module named ‘main( )’ is called.
3. The module named ‘choice( )’ lets the user enter their choice.
4. The choice entered selects one of the sub-modules, which then sends back a parameter.
5. The parameter received from the sub-module is sent to the ‘main( )’ module.
6. The game is played until the user either wins or loses.
7. The ‘continuation( )’ module then receives input from the user.
8. This input is validated and is then sent back to the ‘main( )’ module.
9. The program ends when the user exits in the ‘continuation( )’ module.

GuessTheWord

Continuation( )

Main( )

Choice( )

Bands( )

Country( )

Colors( )

Animals ( )

Books( )

Movies( )

**Testing**

There are two main types of testings that have been carried out on the program on ‘GuessTheWord’. These testings are the alpha testing and the beta testing.

Alpha testing is a form of software testing run on the program to identify bugs before the actual software is released to the the real users or the public. Alpha testing is one form of user acceptance testing.

The Alpha testing has four main objectives. They consist of refining the software product by finding and fixing bugs that were not initially found. One of the objectives is to involve the consumer deep into the process of the development. Lastly, it gives better insight into the program’s reliability earlier in the development stages.

The testing process followed is to first review the design specification and functional requirements, then a comprehensive test case and plan are to be developed. After that the test plan is executed, the defects are logged and a retest takes place as soon has the identified bugs are fixed.There are usually two phases of the Alpha testing, they are:

First Phase:

The first phase of the testing is usually done by the software engineers. A debugging software is used, with the aim being to catch bugs quickly. More often than not, while alpha testing, a tester will find lots of bugs, crashes, and missing features and documents.

Second Phase:

The second phase of the testing usually consists of black box and white box testing, these are done by the quality assurance staff.

Black-box testing:

|  |  |  |  |
| --- | --- | --- | --- |
| Module:Main( ) | | | |
| Types of input Data | Input | Expected output | Actual output |
| Normal | b | “Wrong”/Right | “Wrong”/Right |
| Normal | B | “Wrong”/Right | “Wrong”/Right |
| Extreme | a | “Wrong”/Right | “Wrong”/Right |
| Extreme | z | “Wrong”/Right | “Wrong”/Right |
| Abnormal | 1 | “Please enter an alphabet” | “Please enter an alphabet” |
| Abnormal | @ | “Please enter an alphabet” | “Please enter an alphabet” |

Beta testing is performed by the real users of the software application in a suitable environment. It is also a form of user acceptance testing. Beta testing consists of releasing a beta version of the software to a number of targeted users, with the requirement of feedback on the product quality. Beta testing helps minimize the chance of product failure and it provides increased quality of the program through customer validation.

The attributes of beta testing are as follows:

1. Beta testing is always performed by clients or users who are not employees of the company.
2. Endurance, reliability and security are checked during beta testing.
3. Black-box testing is usually used for beta testing.
4. It is carried out in the user’s location.
5. It does not require any lab or testing environment.

There are five main types of beta testing, they are:

1. Focused beta testing: this is when the software is released into the market for collecting feedback for specific features.
2. Public beta testing: is is when the product is released online and feedback is received. After adjustments are done the product is released officially.
3. Post-release beta testing: the product is released into the market and data is collected for future release of product.
4. Traditional beta testing: the product is officially released and data is collected for future improvement of the software.
5. Technical beta testing: the software is released to a group of employees of an organization and feedback is collected.

The beta testing criteria consists of a sign off document from alpha testing, a beta version of the software is also required. There should be an environment ready to release the software application to the public. A tool should also be prepared to capture real time faults.

Black-box testing:

|  |  |  |  |
| --- | --- | --- | --- |
| Module:Continue( ) | | | |
| Types of input Data | Input | Expected output | Actual output |
| Normal | n | exit( ) | exit( ) |
| Extreme | N | exit( ) | exit( ) |
| Abnormal | o | “Invalid input” | “Invalid input” |

Advantages of alpha testing:

* Gives better insight of the software’s reliability at its early stages.
* Delivery time to introduce the software to the market is reduced.
* The program quality is increased due to the early feedback.

Advantages of beta testing:

* It is cost effective in comparison to other data collection methods.
* It allows a company to test post-launch infrastructure.
* Beta testing creates goodwill with customers and increases customer satisfaction.
* It reduces software failure risk through customer validation and feedback.
* It also increases product quality through customer feedback.

**Implementation**

Code:

import time

import random

import openpyxl as px

import pandas as pd

Option = pd.Series(['1','2','3','4','5','6'])

Letters = pd.Series(['A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'])

def Country():

index = random.randint(1,20)

index = str(index)

book = px.load\_workbook('HangMan.xlsx')

sheet = book['Country']

cell='A'+index

word = sheet[cell].value

return (word)

def Bands():

index = random.randint(1,20)

index = str(index)

book = px.load\_workbook('HangMan.xlsx')

sheet = book['Bands']

cell='A'+index

word = sheet[cell].value

return (word)

def Movies():

index = random.randint(1,20)

index = str(index)

book = px.load\_workbook('HangMan.xlsx')

sheet = book['Movies']

cell='A'+index

word = sheet[cell].value

return (word)

def Books():

index = random.randint(1,20)

index = str(index)

book = px.load\_workbook('HangMan.xlsx')

sheet = book['Books']

cell='A'+index

word = sheet[cell].value

return (word)

def Animals():

index = random.randint(1,20)

index = str(index)

book = px.load\_workbook('HangMan.xlsx')

sheet = book['Animals']

cell='A'+index

word = sheet[cell].value

return (word)

def Colors():

index = random.randint(1,20)

index = str(index)

book = px.load\_workbook('HangMan.xlsx')

sheet = book['Colors']

cell='A'+index

word = sheet[cell].value

return (word)

def choice():

print("Which topic would you like to choose?")

print("1. Movies")

print("2. Books")

print("3. Animals")

print("4. Colors")

print("5. Countries")

print("6. Rock Bands")

time.sleep(0.75)

print("Enter the number corresponding to your choice.")

time.sleep(0.75)

Choice = input("Enter your choice here: ")

while Choice not in Option.values:

print("Please enter a valid choice.")

Choice = input("Enter your choice here: ")

if Choice == '1':

word = Movies()

elif Choice == '2':

word = Books()

elif Choice == '3':

word = Animals()

elif Choice == '4':

word = Colors()

elif Choice == '5':

word = Country()

elif Choice == '6':

word = Bands()

return (word)

def Continue():

option = ""

print ("Would you like to play again? (y/n)")

option = input("Enter your choice here: ")

option = option.upper()

Flag3 = False

while Flag3 == False:

if option == "N":

Flag = False

Flag3 = True

elif option == "Y":

Flag = True

Flag3 = True

else:

print("Invalid input, please enter again.")

option = input("Enter your choice here: ")

option = option.upper()

return Flag

def main():

print(" ")

time.sleep(1)

word = choice()

word = word.upper()

print("Start guessing...")

time.sleep(0.5)

guesses = ' :'

turns = 10

while turns >0:

failed = 0

for char in word:

if char in guesses:

print (char, end=" ")

else:

print("\_", end=" ")

failed = failed + 1

if failed == 0:

print('\n')

print("Congratulations")

print("You Won,",name)

break

print ('\n')

print ("You have ",turns,"more guesses.")

print ("Alphabets that have already been entered",guesses)

Flag1 = False

Flag2 = False

while Flag1== False or Flag2 == False:

guess = input("Guess an alphabet: ")

guess = guess.upper()

if len(guess) != 1:

print("Please enter one character.")

else:

Flag2 = True

if guess not in Letters.values:

print("Please enter an alphabet.")

elif guess in guesses:

print("This alphabet has already been entered before.")

else:

Flag1 = True

guesses +=guess

if guess not in word:

turns = turns-1

print ("Wrong")

if turns == 0:

print("You Lose!")

print("Better luck next time,",name)

print("The word was, ",word)

Flag = True

print("What is your name?")

name = input("Enter your name here: ")

print("Hello, ",name,", Time to play hangman.")

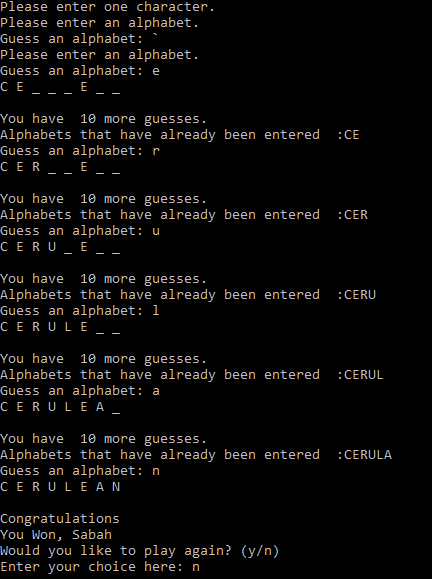
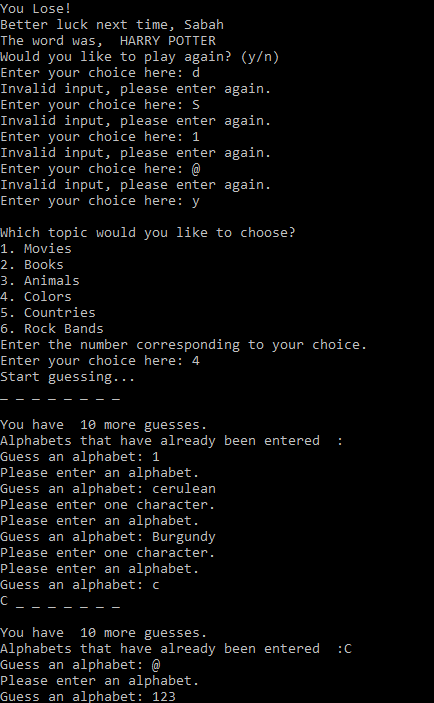
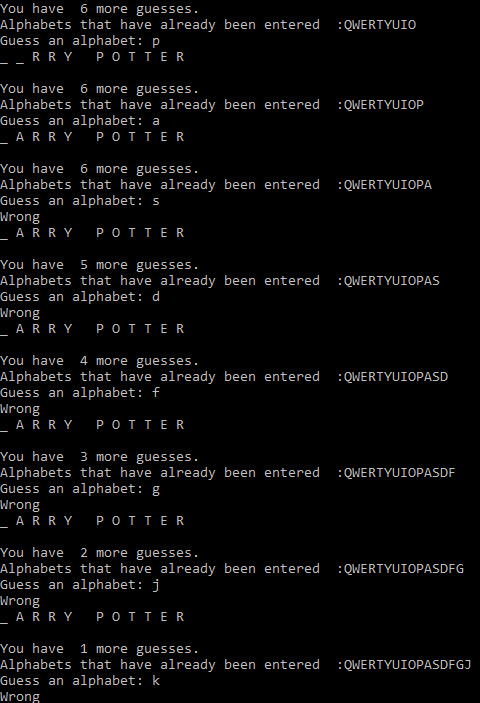
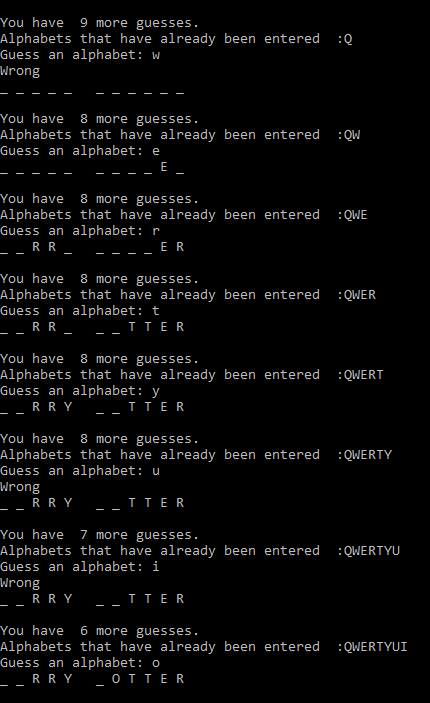
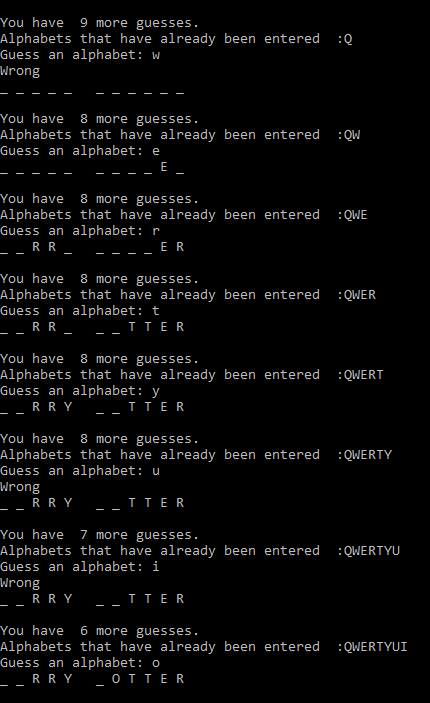
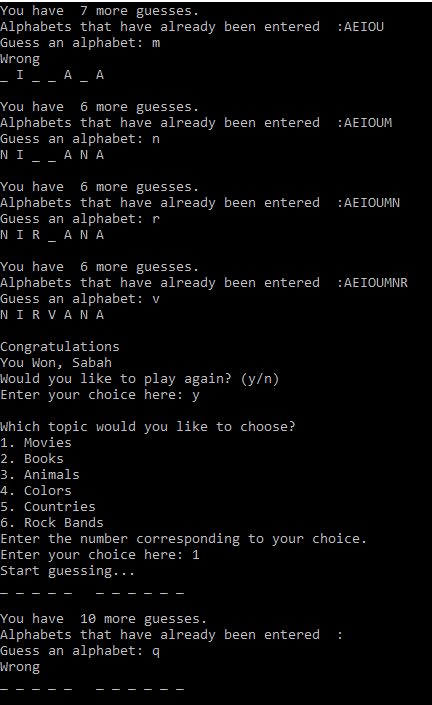
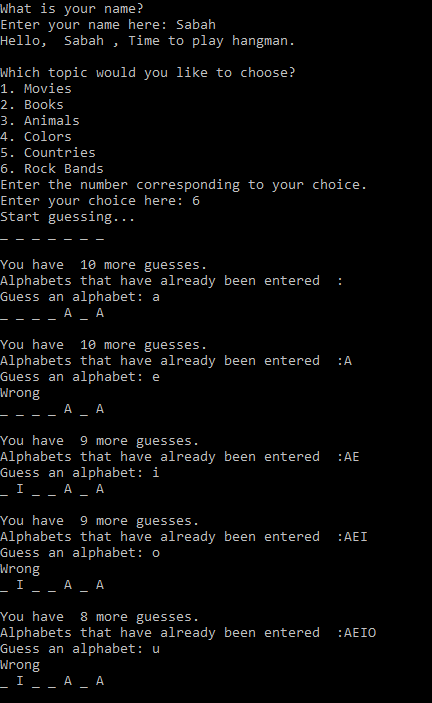
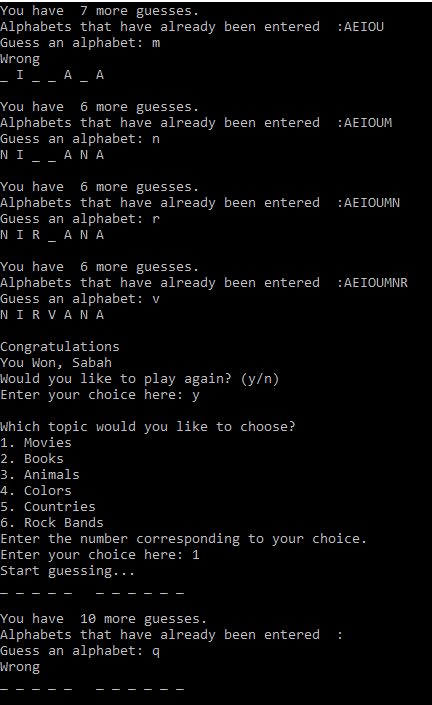
while Flag == True:

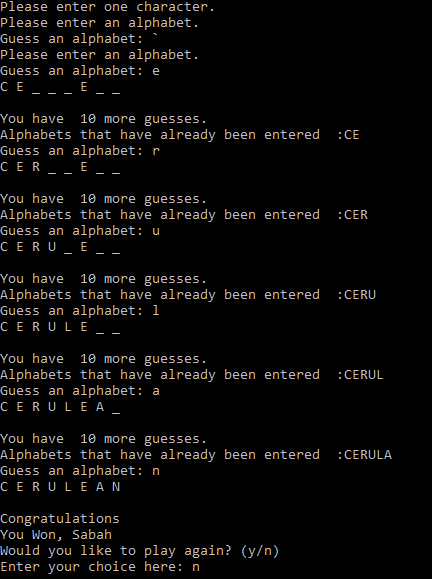
main()

time.sleep(0.5)

Flag = Continue()

Output:





**Conclusion**

The program on “GuessTheWord” is aimed on the younger age group, since this is the time where most of the human development takes place. This program helps children not only to enjoy their leisure time but can also be used to learn spellings of words. The game can also be used in schools to teach children the spellings of different colors, countries, etc.

The advantages of this game are that even though it is aimed for the younger generation even the older generation can take joy in playing the game which is a huge part of their childhood. The option of guessing Rock bands has been kept for the enjoyment of adults, this game can help them relax and unwind after a day of hard work.

This game can also be used in schools as part of their Geography and English lessons, since there are options such as Countries, colors, animals, etc. The Countries option can be used for an interactive lesson in Geography, while the other options can be used as an interactive method to learn spellings.

The drawbacks of this game are that it is very plain and does not have any colors that would attract the children to it. There options are also limited to six options with each holding only twenty words.

Although this game has a few drawbacks it can help with a child’s development and it is also a free software.

This project was done as part of the A-Level coursework for Computer Science (9608), and will be included in the grading process due to the pandemic of the year 2020.