# PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

***(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)***

**PRN No.:** 2017016400173074 **Roll No.:** 36

## Name of the Student

Sachin Chandrakant Yelange

## Title of the Project

Hangman Game

## Name of the Guide

Prof. Sameer Kamble

## Teaching experience of the Guide

1. **Is this your first submission? Yes No**

## Signature of the Student Signature of the Guide

**Date: Date:**

**Signature of the Co-ordinater Date:**

# HANGMAN GAME

## A Project Report Submitted in partial fulfillment of the

**Requirements for the award of the Degree of MASTER OF SCIENCE (INFORMATION TECHNOLOGY)**

## By

**Name: Sachin Chandrakant Yelange Roll No.: 36**

## Under the esteemed guide of Prof. Sameer Kamble

**DEPARTMENT OF INFORMATION TECHNOLOGY CHIKITSAK SAMUHA’S**

## S.S & L.S PATKAR COLLEGE OF ARTS & SCIENCE & V. P. VARDE COLLEGE OFCOMMERCE & ECONOMICS.

**An Autonomous College Affiliated to University of Mumbai Goregaon (West), Mumbai – 400 062**

## 2022-2023

**CHIKITSAK SAMUHA’S**

## S.S & L.S PATKAR COLLEGE OF ARTS & SCIENCE & V. P. VARDE COLLEGE OFCOMMERCE & ECONOMICS.

**An Autonomous College DEPARTMENT OF INFORMATION TECHNOLOGY**

**CERTIFICATE**

This is to certify that the project entitled, **"Hangman Game"**, is bona fide work of **Sachin Chandrakant Yelange** bearing Seat No: **36** submitted in partial fulfillment of the requirements for the award of degree of MASTER’S OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

## Internal Guide Coordinator

**External Examiner**

**Date: College Seal**

# ABSTRACT

This is a simple Hangman game using Python programming language. We can use this as a small project to boost their programming skills and understanding logic. The Hangman program randomly selects a secret word from a list of secret words. The random module will provide this ability, so line 1 in program imports it. Hangman is a popular word game in whichone player the "chooser") chooses a secret word and another player (the "guesser") attempts.

Going back to our old school days, some of the pen-paper games were always atop for our leisure time. Hangman was one, other than some chit games, to guess words according to theguesses determined and as soon as they lost all their wrong guesses, they were hanged (not really, but on paper. That is an old way, now to play Hangman. The new advancement in technologies allows us to play hangman using our own computer also without any other player.

How?

Let’s find out further!!

# ACKNOWLEDGEMENT

I would like to express my sincere gratitude towards the Information Technology Department ofPatkar-Varde College for helping me in the accomplishment of my project.

After months of hard work finally, I am very happy to present my final year project. The projectmaking was full of new experiences along with lots of learning and with some difficulties too. Though a difficult job was made simpler, by timely guidance received, which helped me in converting my thought into reality. So, I would like to take full advantage of this opportunity to thank each and every person who has helped me throughout the completion of the project.

I am obliged to my parents & family members who always supported me greatly and encouragedme in each and every step. I would like to give my special thanks and express sincere gratitude towards our Chief Educational Officer Mr. Rajesh Kanakal, In-Charge principal Dr. Pratibha Gaikwad and the Co-ordinator of IT Department, Mrs. Namrata Kawale Shinde.

I owe my sincere thanks to our project guide Mr. Sameer Kamble for constantly supporting andencouraging me in the process of developing the project, without which the successful completion of this project would have been impossible. They have been instrumental in makingme concentrate and focus on the implementation of this project. And last but not least. I would like to thank the entire teaching and non-teaching staff of the college.

DECLARATION

I hereby declare that the project entitled, HANGMAN GAME done at Patkar-Varde College, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of MASTER’S OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted as

final semester project as part of our curriculum.

## Sachin Chandrakant Yelange

**Signature..................................**

# TABLE OF CONTENTS

## Chapter 1: Introduction 1

* 1. Introduction
  2. Objective
  3. Purpose, Scope and Availability
  4. Achievements

## Chapter 2: Survey of Technologie 5

* 1. Problem Statement
  2. Methodology
     1. Research Methodology
     2. Database
     3. Web-App Front-End
     4. Web-App Back-End
     5. Containerization
     6. Machine Learning
  3. Proposed System
  4. Benefits
  5. Summarized Definition

## Chapter 3: Research Knowledge and Literature Review 8

* 1. Primary Research
  2. Secondary Research
  3. Technology

## Chapter 4: Description of Methodology 21

* 1. Benefits
  2. Future Strategy
  3. Planning and Scheduling
  4. Preliminary Product Description
  5. Concept Design

## Chapter 5: Implementation & Result. 36

* 1. Implementation approach..
  2. Code Efficiency
  3. Testing
  4. Test Case
  5. Test Reports
  6. Modification
  7. User Documentation

## Chapter 6: Conclusions and proposals for the future work 90

* 1. Conclusion
  2. Limitations
  3. Future Scope

## Chapter 7: Bibliography 95

* 1. **Introduction:**

# CHAPTER 01: INTRODUCTION

Hangman is a popular word-guessing game where the player attempts to build a missing word by guessing one letter at a time. After a certain number of incorrect guesses, the game ends and the player loosest game ends and the player loses. The game also ends if the player correctly identifiesall the letters of the missing word. The code is written in python language. Visual Studio is used to compile the code. The code carries out all the functions that we see in a hangman game.

Going back to our old school days, some of the pen-paper games were always a top for our leisure time. Hangman was one, other than some chit games, to guess words according to the guesses determ–ined and as soon as they lost all their wrong guesses, they were hanged (not really, but on paper. That is an old way, now to play Hangman. The new advancement in technology allows us to play hangman using our own computer also without any other player. How? Let us find out further!!

## Objective:

The idea for the game was to make this game easy to operate without the problem of thinking a new word every time. You just click a button, and a random word is guessed for you. And every time you win you level- up and a new word is waiting for you to be guessed. With every win youget a score of 5 added. And when you lose a game, your name is asked, and your score is saved with your name in a text file. You can view your high score in main menu where we have given an option for high score.

Gaining students’ motivation is one of tasks that teacher should do in order the students have willingness to learn, but to create highly motivating environment in the classroom is not easy (Meece and McCuskey, 2001: 33). For Example, when the students learn English. English is an international language. And in this global era, it is necessary for people to be able to speak English.But all students have a different motivation to learn, and it is the teacher’s task to create a motivating environment in the learning process.

The objective of our project is to implement the hangman game using Python. It does not require any specific modules other than random and time. Python loops and functions are enough to buildthis game here.

## Purpose, Scope and Applicability:

Language games is a general term used to cover a variety of language activities, and the function of language games is to practice specific language items such as grammar, sentence structures, vocabulary, and spelling, and it is important to develop language skill (Rezapanah and Hamidi, 2013). Students can get a lot of knowledge and new experience by playing language games, because the students can learn language with different and fun ways. Language games will give them much benefit which is very important for their language skills.

There are many factors in games such as employing rules, fostering cooperation while making learning fun and games also enjoyable (Yolageldili and Arikan, 2011). It means that the teaching learning activity will be more interesting by using game. Game can also develop the teamwork between students. One of the effective techniques is by using games so students can improve their vocabulary knowledge and ability to communicate (Chirandon, Laohawiriyanon and Rakthong, 2010).

In other words, game is a useful technique that can be used by teacher so that the students can geta lot of vocabulary and the students can also improve their courage to communicate with their friends, especially in the lesson. Because communication skill is very important for the students. Their skill to communicate with others can be used when they should present something in their class, or when the students should give their opinion during discussion.

A classic way to practice the alphabet and vocabulary is by using Hangman game (Rauschenberg, 2005). Hangman game can be called as one of language game which can be used by teacher to teach the students about alphabet and vocabulary. Kartikawati (2014) also states that the students can express their idea in trying to guess the secret word by Hangman game and the researcher also can create fun atmosphere. So, the teaching learning activity will be not boring for the students by playing Hangman game, students will think about the letter which is in the word. They will also think about what the right word is. Hangman game also can be one way to help studentspractice their confidence to express their thought.

## Purpose

The player keeps guessing until seven wrong guesses have been made or until the word is completed, whichever happens first. At the end of a round, the user is prompted to indicate whetherhe/she wants to play again. The game ends when the player indicates that he/she wants to quit.

These are all some of the hardest-to-guess hangman words. Most words are obscure, contain unusual letters, or contain repeating letters that make guessing a wrong letter very likely. The length of these words is 4+ letters long. It is too easy to slump guessing with real short words. If you are interested in the thought behind selecting these words, here are some strategies for picking hangman words, and here's a list of hard hangman words to just choose from.

One of those classic word-guessing games played between two or more players. One player sets or defines a word or list of words, and the other player(s) must guess correctly, given a certain number of trials. It is quick, easy, and educational and most times requires only a piece of paper and the ability to spell correctly. It could also be played with a computer (like the one which will be built in this article). For every wrongly guessed word, a life or trial in the game is lost and a “hanging man” begins to appear, piece by piece. The aim is to solve the puzzle and guess the correct word(s)/phrase before the hangman dies.

The idea for the game was to make this game easy to operate without the problem of thinking of a new word every time. You just click a button, and a random word is guessed for you. And every time you win you level up and a new word is waiting for you to be guessed. With every win, youget a score of 5 added. And when you lose a game, your name is asked, and your score is saved with your name in a text file. You can view your high score in the main menu where we have givenan option for a high score.

## Scope

We are designing the world’s popular game hangman. Its purpose is to turn this paper game into a computerized version. In it, User has the option to play a game. The hangman will be drawn automatically if your chose an incorrect word.

If you finish the first level, you will be taken to the next level with a new word to be guessed. And every time you clear a level a score of 5 five will be added to your overall score.

This project requires good knowledge of Python which includes defining functions and managing for/while loops. The functions that we use here contain arguments that are defined in a global scope which can be further used in other functions to improve game quality. It can also be used toprovide different steps when required to execute upon conditions by the for and while loops.

## Applicability

A Hangman Game on Python is about guessing letters (A-Z) to form words. If the player guessesthe right letter that is within the word, the letter appears at its correct position. The user must guessthe correct word until a man is hung, then the game is over. The video Player is loading.

A hangman game was given to teach students of the experimental group. Students were divided into four groups, to make it easier to play the game. Then the writer wrote some dashes which represent the number of words After that, students tried to guess what letter is in the word. If the letter was not in the word, then the writer started to make the hanged man. After they knew the answered, they should raise their hand and guess the word. Students from the group who answeredthe word would get a score if the answer was correct. The words which were given to the studentswere related to the material of the lesson. The material of the lesson was the Recount text. These words can help the students to know the material that will be explained to them. And these wordscan increase their vocabulary.

It means that by using Hangman game, the students can practice the alphabet and vocabulary whichis important for their English skill, and it can also create fun atmosphere in teaching-learning activity.

## Achievements:

So, the idea in this series is to first create a very simple version and then in the upcoming series update and improve the functionality. Just as with the previous series. We would not be using the image display here, just a simple logic controlon how the game works. This classic game has gained so much popularity, that just like most games, there are a couple of tips, tricks, and strategies used to win.

At the start of the challenge, you're assigned a random phrase and you need to try and guess the letters by earning achievements. Can you work out the phrasebefore you run out of lives.

This Project perform all actions that we see in a typical hangman game. Thisproject is developed in visual studio and the program works well. It performsfollowing functions:

1. Welcome
   * Play Game
   * Rules and Instructions
2. High Score
3. A terminal with the game and the picture of a hangman
4. Auto level-up and added score
5. Store high score with name if reached

**We also did file handling in that project so that the data can be saved in atext file.**

# CHAPTER 02: SURVEY OF TECHNOLOGIES

## Problem statement

* + 1. The fourth study talked about the relationship between motivation and students’ English learning achievement. The purpose of this study was to find out the levels of motivate students with high and low achievement, to investigate the differences and similarities motivation of students with high and lowachievement, and to investigate relationship between motivation and students’achievement. This study used a questionnaire surveys an in-depth interview.
    2. Turn the normal hangman game into a computerized version for compatibility.
    3. This is a simple Hangman game using Python programming language.
* The Hangman program randomly selects a secret word from a list of secret words. The random module will provide this ability, so line 1 in program imports it.
* The Game: Here, a random word (a fruit name) is picked up from our collection and the player gets limited chances to win the game.

## Methodology

#### Research Methodology

* + - 1. The students were bored and sleepy in English lesson. They looked like they did not have interest in learning English. To do this research the writer found some journals about Hangman game and motivation. And the result of the journal is language through games can encourage the operation of certain psychological and intellectual factors, one of them is motivation (Pisheh and Bagheri, 2012).
      2. To teach the students there should be a right method or technique to make the teaching learning activity more fun, especially in learning English.
      3. Because English is not our mother tongue, so students will have some difficulties in learning English. And sometimes, the students have low motivation to learn the other language.
      4. And based on the observation result, the students’ activities when they learned English were different between experimental group and control group. In experimental group, the students had a lot of enthusiastic towards learning English.
      5. They gave a positive response in teaching learning activities. While in control group, the students did not have a lot of enthusiastic to learn English.
      6. The students were bored and sleepy in English lesson. They looked like they did not have interest in learning English.

#### Web-app Frontend

* + - 1. In this project, we are using Flask as a front end, which is a web-based framework writtenin python implemented on Werkzeug and Jinja2.
      2. Flask comes with many tools, libraries, and technologies that building builds a good web application.
      3. Flask is classified into a micro-framework which means it has little to no dependencies on external libraries. The front end of a website is the area with which the user immediately interacts.
      4. It contains everything that users see and interacts with text colors and styles, images and videos, graphs and tables, the navigation menu, buttons, and colors.
    1. **Web-app Backend**
       1. As a backend here we are using Python. Python is a general-purpose interpreted, high-level programming language, which can be used for the server-side application, system scripting, and many more.
       2. It saves and organizes data and ensures that everything on the client side of the website functions properly.
       3. It is the section of the website that you are unable to interact with or access. It is part of the software that has no direct communication with the users.
       4. Back End development is done using Python.

## Proposed System

* + 1. Hangman game was given to teach students the of experimental group. Students were divided into four groups, to make it easier to play the game. Then the writer wrote some dashes which representing the number word.
    2. After that, students tried to guess what which is in the word. If the letter was not in the word, then the writer started to make the hanged man. After they knew the answered, they should raise their hand and guess the word.
    3. Students from the group who answered the word would get a score if the answer were correct. The words which were given to the students were related to the material of the lesson.
    4. The material of the lesson was the Recount text. These words can help the students to know the material that will be explained to them. And these words can increase their vocabularies.

## Benefits

* + 1. Hangman game, students will think about the letter which is in the word. They will also think about what the right word is. Hangman game also can be one way to help students practice their confidence to express their thought.
    2. It means that by using Hangman game, the students can practice the alphabet and vocabulary which is really important for their English skills and it can also create fun atmosphere in teaching learning activity.
    3. Based on the result of questionnaire, the mean of experimental group was higher than control group. And after made some calculation from the data, the writer found that the result of account was higher than table.
    4. It means that the students in experimental group had high motivation in learning English than control group. Hangman game which is a special treatment for experimental group gave some effect to the students.

## Summarized Definition

* + 1. And based on the observation result, the students’ activities when they learned English were different between experimental group and control group. In experimental group, the students had a lot of enthusiastic towards learning English.
    2. They gave a positive response in teaching learning activities. While in control group, the students did not have a lot of enthusiastic to learn English. The students were bored and sleepyin English lesson. They looked like they did not have an interest in learning English.
    3. With this project in Python, we have successfully developed the Hangman game. We used the popular time and random modules to render our program. Executing different functions and using loops helped us with a better understanding of python basics.

# CHAPTER 03:

**RESEARCH KNOWLEDGE LITERATURE REVIEW**

## PRIMARY RESEARCH

We have employed a variety of techniques for primary research and have created numerous research questionnaires. By analyzing the responses, we may further analyze the concepts.

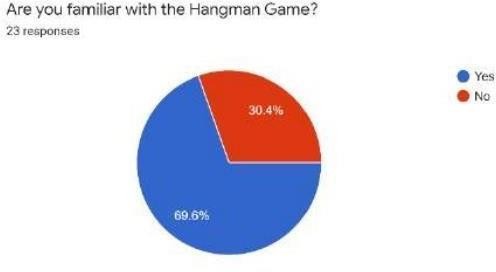
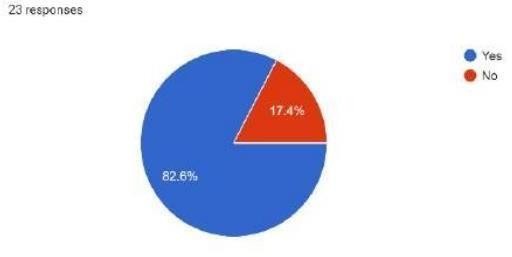
For performing primary research, we used the following techniques:

#### Surveys:

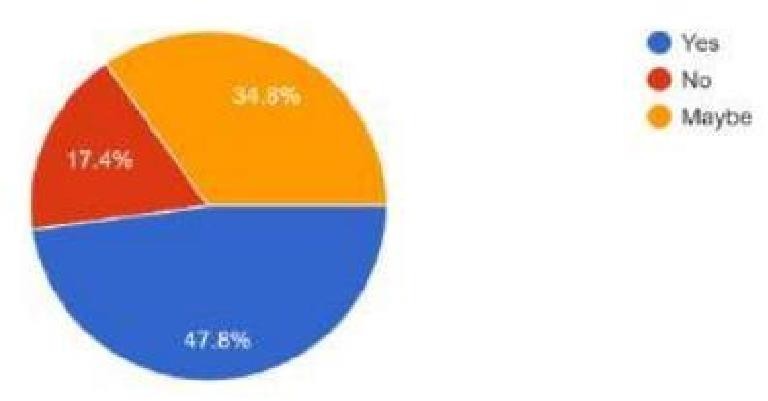
This is an excellent data gathering method because it allows us to learn about people's opinionsregarding the survey topic by asking them to respond to specific questions about their feelings,beliefs, attitudes, and behavior.

After ensuring that the application functions flawlessly on a variety of devices, the author created a brief questionnaire that allows users to rate the app's efficacy. There are 23respondents to the survey as of right now. 69.6% of the 23 respondents are familiar with the Hangman Game, and 82.6% of them expressed an interest in learning Mandarin, as seen in Interest chart in Hangman User Review Chart 15 JIKO (Jurnal Informatika dan Komputer), Vol. 6, No. 1, February 2022, hal. 6-17, e-ISSN: 2477-3964, p-ISSN: 2477-4413 Application Interest Chart, Figure 18. Assessing consumers' interest in utilising the application is the nextstep.

Out of 23 responses, almost half of them show interest in using the application, while 34.8% of the responses still indicate inconclusiveness. While 17.4% of the answers show disinterest in using the application, as seen in Figure 19, Figure 19.19: User Interface Rating Chart Additionally, the text also asked the user to rate the UI of this application. The result is leaningtowards the positive side, with scores of 4 and 5 currently tied at 8 votes. As seen in Figure 20,3 respondents rated it average with a score of 3, and 1 respondent rated it as poor at 1. Figure 20: Application Efficiency Chart Next, the author also asked if the user thinks this applicationcan help them prepare for HSK efficiently. Out of 23 respondents, 87% of them think this application can help them be more prepared for HSK. Meanwhile, 13% of them think that thisapplication would not be any help in HSK preparation. Finally, the author also asked for someconstructive criticism for this application. Most of the answers draw attention to the difficultyof the game, noting that it may be too hard for beginners. Respondents suggested that adding a translation feature or hint system would be better. Additionally, some respondents also request that pinyin be added to the application. All the suggestions are duly noted and will beconsidered to be added as improvements in the future.

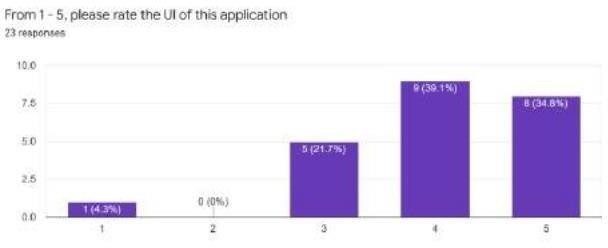


#### User Review Chart

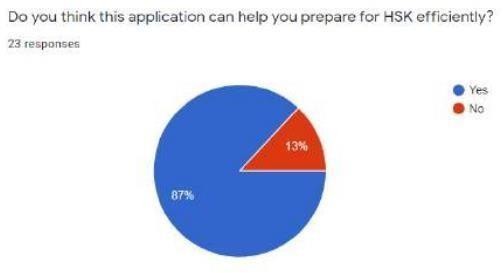


**Application Interest Chart**

The next question is to gauge the interest of users in using the application. Out of 23 responses, almost half of them show interest in using the application, while 34.8% of the responses still indicate inconclusiveness. Mean-while, 17.4% of the answer shows disinterest in using the application.



The user was also requested to rate the application's user interface in the essay. With scores 4 and5 tied at 8 votes each, the outcome is leaning toward being favorable. According to Figure, 3 people gave it an average rating of 3, while 1 said that it was poor.



#### Application Efficiency Chart

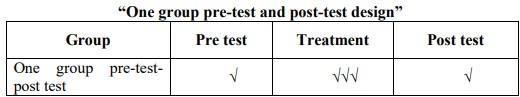
* + - 1. **Interviews:**

We have developed a simple method for gathering data from single individuals or small groups of people. We chose to do the interview digitally because we were unable to organize a physical place because to COVID-19.

To find out their opinions on cryptocurrencies, we spoke with some thesis advisors, PhD candidates, professors, and some cryptocurrency owners.

#### Observation

This research is conducted based on the pre-experimental design which is one grouppre-test and post- test. The pre-experimental design can be presented as follow:



1. Choosing the single group on one group
2. Giving pre test
3. Giving the treatment to group
4. Giving the post test

#### Focus groups:

We assembled a group of 10–12 individuals who were familiar with the hangman game. People from the group provided their perceptions as well as advice to comprehend nature and obtain a deeper understanding of the hangman game, which made this qualitative data gathering method very helpful.

#### Data Collecting Technique

In collecting the data, the researcher follows the step below;

* + - * 1. Find out the vocabulary relate to the subject in this case the researcher uses the hangman game as media.
        2. Giving children pre- test. The pre- test carried out to get the data of children score, especially vocabulary.
        3. Giving children treatment. this treatment is the teacher teach vocabulary in learning descriptive text using hangman game.
        4. Giving children post-test. The post-test is given to know how far the children master the vocabulary after teaching vocabulary using hangman game.
        5. Giving children questionnaire the question is given in order to know students’ responses toward hangman game technique.

## SECONDARY RESEARCH

This project requires good knowledge of Python which includes defining functions and managing for/while loops. The functions that we use here contain arguments that are defined in a global scope which can be further used in other functions to improve game quality. It can also be used to provide different steps when required to execute upon conditions by the for and while loops.

While conducting primary research we also continued the secondary research as it was really important to know the other side as well.

While conducting secondary research we have following sources which can assist us.

* Academic peer-reviewed journals
* Published books and articles
* Educational institutions
* Commercial information sources

We have read at least 5-6 paper for finding the useful information and qualitative data. Below is the same information which we used during our secondary research.

* Adachi, Rie (2015). Motivation and Communicative Attitudes Among japanese EFL Pupils. Indonesian Journal of Applied Linguistics, Vol. 5 No. 1, July 2015
* Alam, M. T., and Farid, S. (2011). Factors Affecting Teachers Motivation. International Journal of Business and Social Science Vol. 2 No. 1, January 2011.
* Brophy, Jere (2004). Motivating Students to Learn Second Edition.
* London: Lawrence Erlbaum Associates. Broughton G. et. al. (2003). Teaching English as a Foreign Language Second Edition. USA and Canada: Routlede.
* Brown, H. Douglas (2010). Language Assessment Principles and Classroom Practices Second Edition. United States: Pearson Longman.
* Brumfit C., Mitchell R. (1989). Research in the Language Classroom. Modern English Publication in association with The British Council.
* Chirandon A, Laohawiriyanon Cholanda, Rakthong A (2010). The Effects of Teaching English through Games. The 2nd International Conference on
* Humanities and Social Sciences, April 10th, 2010 Faculty of Liberal Arts, Prince of Songkla University.

## TECHNOLOGY:

#### Python

Python is one of the many open sources object oriented programming application software available in the market. Python is developed by Guido van Rossum. Guido van Rossum started implementing Python in 1989. Python is a very simple programming language so even if you are new to programming, you can learn python without facing any issues. Some of the many uses of Python are application development, implementation of automation testing process,

Allows multiple programming build, fully constructed programming library, can be used in all the major operating systems and platforms, database system accessibility, simple and readable code, easy to apply on complex software development processes, aids in test driven software application development approach, machine learning/ data analytics, helps pattern recognitions, supported in multiple tools, permitted by many of the provisioned frameworks, etc.

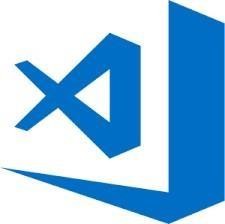


* + 1. **Visual studio code**

Visual Studio Code is a very powerful and easy-to-use code editor. It comes with broad programming language support, is highly customizable with various extensions and it is for free -a great package for beginners and more advanced programmers in my opinion.

What makes VS Code special? Microsoft’s Visual Studio Code is a text editor with powerful IDE- like features. Side note: VS Code is a totally separate product from Visual Studio the IDE. In termsof what is available right now, it seems like VS Code is a “best of both worlds” option between IDEs and Text Editors. Here are some more reasons why we’re using VS Code:

* + - * VS Code is simple to download and install on any OS.
      * It is completely free.
      * It is non-opinionated and open source, with non-proprietary standards.
      * You can bring your favorite key bindings from Sublime, Vim, etc. with you.
      * It’s easy to customize and has many useful extensions.
      * It has powerful IDE-like features including a built-in debugger and terminal.



#### Applications of Python programming language

Python can be used to develop different applications like web applications, graphic user interface- based applications, software development applications, scientific and numeric applications, network programming, Games and 3D applications and other business applications. It makes an interactive interface and easy development of applications. You may be wondering what all the applications of Python are. There are so many applications of Python, here are some of them.

1. Web development – Web framework like Django and Flask are based on Python. They help you write server-side code which helps you manage database, write backend programming logic, mapping URLs etc.
2. Machine learning – There are many machine learning applications written in Python. Machine learning is a way to write a logic so that a machine can learn and solve a particular problem on its own. For example, products recommendation in websites like Amazon, Flipkart, eBay etc. is a machine learning algorithm that recognizes user’s interest. Face recognition and Voice recognition in your phone is another example of machine learning.
3. Data Analysis – Data analysis and data visualization in form of charts can also be developed using Python.
4. Scripting – Scripting is writing small programs to automate simple tasks such as sending automated response emails etc. Such type of applications can also be written in Python programming language.
5. Game development – You can develop games using Python.
6. You can develop Embedded applications in Python.
7. Desktop applications – You can develop desktop application in Python using library like TKinter or QT.

#### Visual studio code

PyCharm is a hybrid platform developed by JetBrains as an IDE for Python. It is commonly usedfor Python application development. Some of the unicorn organizations such as Twitter, Facebook, Amazon, and Pinterest use PyCharm as their Python IDE!

It supports two versions: v2.x and v3.x.

We can run PyCharm on Windows, Linux, or Mac OS. Additionally, it contains modules and packages that help programmers develop software using Python in less time and with minimal effort. Further, it can also be customized according to the requirements of developers.

#### Features of PyCharm:

1. **Intelligent Code Editor:**
   * + - It helps us write high-quality codes!
       - It consists of color schemes for keywords, classes, and functions. This helps increase the readability and understanding of the code.
       - It helps identify errors easily.
       - It provides the autocomplete feature and instructions for the completion of the code.

#### Code Navigation:

* + - * It helps developers in editing and enhancing the code with less effort and time.
      * With code navigation, a developer can easily navigate to a function, class, or file.
      * A programmer can locate an element, a symbol, or a variable in the source code within no time.

1. **Refactoring**
   * + - It has the advantage of making efficient and quick changes to both local and global variables.
       - Refactoring in PyCharm enables developers to improve the internal structure without changingthe external performance of the code.
       - It also helps split up more extended classes and functions with the help of the extract method.
2. **Assistance for Many Other Web Technologies:**
   * + - It helps developers create web applications in Python.
       - Developers have the choice of live editing with this IDE. At the same time, they can previewthe created/updated web page.

#### Assistance for Python Scientific Libraries

* + - * PyCharm supports Python’s scientific libraries such as Matplotlib, NumPy, and Anaconda.
      * These scientific libraries help in building projects of Data Science and MachineLearning.
      * It consists of interactive graphs that help developers understand data.

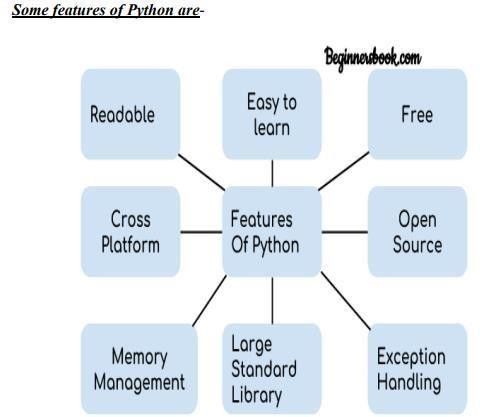


## Features:

This Project perform all actions that we see in a typical hangman game. This project is developedin visual studio and the program works well. It performs following functions:

* + 1. Welcome
* Play Game
* Rules and Instructions
* High Score
  + 1. A terminal with the game and the picture of hangman
    2. Auto level-up and added score
    3. Store high score with name if reached

We also did file handling in that project so that the data can be saved in a text file.



1. **Readable**: Python is a very readable language.
2. **Easy to Learn**: Learning python is easy as this is a expressive and high level programming language, which means it is easy to understand the language and thus easy to learn.
3. **Cross platform**: Python is available and can run on various operating systems such as Mac, Windows, Linux, Unix etc. This makes it a cross platform and portable language.
4. **Open Source:** Python is a open source programming language.
5. **Large standard library:** Python comes with a large standard library that has some handy codes and functions which we can use while writing code in Python.
6. **Free:** Python is free to download and use. This means you can download it for free and useit in your application. See: Open-Source Python License. Python is an example of a FLOSS (Free/Libre Open-Source Software), which means you can freely distribute copies of this software, read its source code and modify it.
7. **Supports exception handling:** If you are new, you may wonder what is an exception? Anexception is an event that can occur during program exception and can disrupt the normal flow of program. Python supports exception handling which means we can write less errorprone code and can test various scenarios that can cause an exception later on.
8. **Advanced features:** Supports generators and list comprehensions. We will cover these features later.
9. **Automatic memory management**: Python supports automatic memory managementwhich means the memory is cleared and freed automatically.

#### Pros:

* Thes are all some of the hardest to guess hangman words. Most words are obscure, contain unusual letters, or contain repeating letters that make guessing a wrong letter very likely. Thelength of these words are 4+ letters long. Its too easy to slump guessing with real short words.
* If you're interested in the thought behind selecting these words, here are some strategies for picking hangman words and here's a list of hard hangman words to just choose from.

#### Cons:

There are very few newly introduced concepts here. I will explain them below: “Pause, take a moment, think about the game, think about how it would proceed without code, then think about the flow. Remember that coding is only a tool.”

* Python string method.isalpha(): This is a string method which returns True if the all the characters in a string/variable is actually a string, and returns False otherwise, i.e. if any of thestring contains a numerical character alongside a string such as ‘num12’, it would return False.In the program, it was used to make sure the name entered by the user should only containedstring characters and not numbers as a game name.
* Docstrings: Docstrings are literals used in python inside a function, a class or a block of code, as some sort of information about what that particular sets of code does. For example, in the program, it was used to provide information on the functions and their purposes.

Generally, the logic of the game coded here goes as follows:

* A welcome function is defined, where a user enters a game name and the game nameis saved in a variable. A condition is set to ensure the entry by the user contains only letters and not numbers. The welcome message displayed is personalized and the useris informed that the word to be guessed will be set by the computer.
* A new function is defined where options are made available if the user wishes to playthe game again. In this function, if the response is yes, the game is initiated again, with a random word.
* A function is defined for generating the word which the user is to guess, and this is done using the random module imported at the beginning of the program, from a list of predefined words.

Finally, a the function controlling the game is defined, with certain variables such as the user’s guesses and number of tries, and using a set of conditions anddecisions, taking into consideration possible occurrences, such as:

* The user enters a non-alphabet
* The user enters a word twice or has guessed that letter before
* The user guesses a letter that is not in the proposed word.
* The user tries to give a shot at guessing the entire word at a go but fails.
* The user guesses correctly
* The user runs out of tries

# CHAPTER 04 DESCRIPTION OF METHODOLOGY

#### PROBLEM DEFINITION:

Python is an extremely versatile programming language employed by many large companies.It is a simple and easy-to-understand syntax, which makes it perfect for those trying to mastercomputer programming for the first time. It is a high-level programming language. Its fundamental design principle is about understanding code and the syntax that lets programmers communicate concepts within a few lines of code.

Sometimes, we need the computer to choose the random number from the specified range, choose an element randomly from a set, choose the random card from an assortment of decks,flip a coin, and so on. The random module allows the ability to access functions that can support these kinds of operations. One of these operations is the random. Choice () technique(returns an unspecified item from a tuple, list, or string.) which we'll make use of to choose the random words from a set of terms we've generated.

The game involves an array of words from which our interpreter will select one random word.The player must first input their names and is challenged to guess the alphabet of their choiceafter that. If the random word is comprised of the alphabet, it will be displayed in the output (with proper placement); otherwise, the program will prompt you to choose an alternative alphabet. Users will be given 12 turns (can be modified according to) to determine the full word.

One of those classic word guessing games played between two or more players. One player sets or defines a word or list of words, and the other player(s) have to guess correctly, givena certain number of trials. It is quick, easy, educational and most times requires only a piece of paper and the ability to spell correctly. It could also be played with a computer (as the onewhich will be built in this article). For every wrongly guessed word, a life or trial in the gameis lost and a “hanging-man” begins to appear, piece by piece. The aim is to solve the puzzle and guess the correct word(s)/phrase before the hangman dies.

* 1. **FUTURE STRATEGY**

The fact that the twelve most commonly occurring letters in the English language are e-t-a-o-i-n-s- h-r-d-l-u (from most to least), along with other letter frequency lists, are used by the guessing player to increase the odds when it is their turn to guess. On the other hand, the samelists can be used by the puzzle setter to stump their opponent by choosing a word that deliberately avoids common letters (e.g., rhythm or zephyr) or one that contains rare letters (e.g. jazz). The word to guess is represented by a row of dashes representing each letter of the word. Rules may permit or forbid proper nouns, such as names, places, brands, or slang. If theguessing player suggests a letter which occurs in the word, the other player writes it in all itscorrect positions. If the suggested letter does not occur in the word, the other player draws oneelement of a hanged stick figure as a tally mark.

The player guessing the word may, at any time, attempt to guess the whole word. If the wordis correct, the game is over and the guesser wins. Otherwise, the other player may choose to penalize the guesser by adding an element to the diagram. On the other hand, if the guesser makes enough incorrect guesses to allow the other player to complete the diagram, the guesserloses. However, the guesser can also win by guessing all the letters that appear in the word, thereby completing the word, before the diagram is completed.

Another common strategy is to guess vowels first, as English only has five vowels

(a, e, i, o, and u, while y may sometimes, but rarely, be used as a vowel) and almost everyword has at least one.

According to a 2010 study conducted by Jon McLoone for Wolfram Research, the most difficult words to guess include jazz, buzz, hajj, faff, fizz, fuzz and variations of these.

As the name of the game suggests, the diagram is designed to look like a hanging man. Although debates have arisen about the game, it is still in use today. A common alternative for teachers is to draw an apple tree with ten apples, erasing or crossing out the apples as theguesses are used up.

Some modifications to game play (house rules) to increase the difficulty level are sometimes implemented, such as limiting guesses on high frequency consonants and vowels. Another alternative is to give the definition of the word; this can be used to facilitate the learning of aforeign language.

## PLANNING AND SCHEDULING

To develop a system that detects potholes, we must first imagine how the system will function.Once you start imagining, you'll have a good idea of what you'll need to build the system.

* + - Once you begin to imagine, you will have a good idea of what is needed to build the system to your specifications.
    - Make a list of your project's needs or requirements.
    - Make a list of your ideas and consider what kind of software services you will need to achieve the best results.
    - Learn basic coding and conduct background research for your chosen software.
    - Once you've decided on the software, begin by implementing the basic modules, whichwill include multiple tabs.
    - Choose open source and public software instead of paid software, which may increaseyour costs.
    - Perform all the necessary testing for your product to determine the expected vs. actual results.
    - A blueprint or prototype will aid you in the design of your system.

## PERT CHART

PERT charts are project management tools that provide a graphical representation of a project's timeline. PERT, or Program Evaluation Review Technique, was developed in the 1950s by the United States Navy for the Polaris submarine missile programmer. PERT chartsallow you to analyze the tasks in a specific project, paying special attention to the time required to complete each task and the minimum time required to complete the entire project. The greatest strength of PERT is its ability to incorporate uncertainty in project time estimatesinto its methodology. It makes numerous assumptions that can either accelerate or delay the project's progress. Project managers can use PERT to estimate possible timelines.

A PERT chart, also known as a PERT diagram, is a project management tool used to schedule, organize, and coordinate tasks within a project. It provides a graphical representation of a project's timeline, allowing project managers to break down each individual task in the projectfor analysis. The PERT chart template uses nodes (drawn as rectangles or circles) to represent events and milestones throughout the project. The nodes are linked by vectors (drawn as lines)that represent the various tasks that must be completed.

PERT charts estimate the amount of time required to complete a project for project managers. Managers can also examine the work breakdown and task connections, as well as evaluate the project's risk. By visualizing the dependencies between each step of the process, the breakdown structure makes it simple to organize a complex project with numerous moving parts.

PERT is an acronym for Program Evaluation Review Technique, which was developed in the1950s by the United States Navy to manage the Polaris submarine missile programmed. At around the same time, a similar methodology, the critical path method (CPM), was developedfor project management in the private sector.

## Advantages of PERT charts

Project managers use PERT charts to gain these benefits:

* + - The PERT technique provides the ability to evaluate the time and resources necessaryto a project by tracking required assets at each stage of the process, as well as throughout the course of the project.
    - PERT charts are useful in what-if analyses, helping companies understand all possible workflows and choose the most efficient and beneficial path.
    - The analysis of the PERT chart includes data from various departments within an organization. Combining all of the information helps identify each responsible team within the company, while facilitating an environment where each department takes responsibility for its work.

## Disadvantages of PERT charts

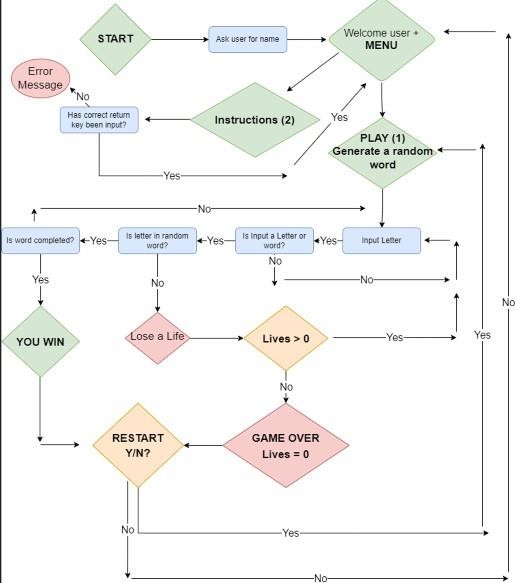
Disadvantages of the PERT chart include the following:

* + - A strict focus on deadlines may not enable managers to see the full financialpositioning of the project.
    - PERT charts lack the flexibility to adapt to small changes that occur when confrontedwith a roadblock.
    - If any calculations are inaccurate in the creation of the chart, delays could occur, causing bottlenecks and negatively impacting the final delivery date.

## PRELIMINARY PRODUCT DESCRIPTION

PPDs are documents that describe a Hangman project in written, graphic, and code form so that owners, estimators, and lenders may understand the design purpose. PPDs document the effectiveness and quality of the functional components of software requirements. PPDs are organized by system and assembly, unlike specifications. From the outside, the hangman game development process appears to be quite innocent. Well, no big deal. A group of peoplegot together and created a game. Many uninitiated people continue to criticize developers forbeing too slow in creating games. If we reject nonprofessionals, the gaze reveals a harsh reality. The steps to creating a hangman game are complex and multi-faceted. Here you will find:

There is a lot of preliminary planning and conceptual work, which is exciting but takes a longtime and drains the lifeblood out of game designers; It is work on design and art that both inspires and breaks the nerves of artists who accept and implement endless edits.



## Hangman Functional Requirements

1. **Starting the Application**

Allow the user to enter on the command line the IP address of a remote computer that is runninga Hangman Word Server. If the user doesn't provide an IP address, use as a default the address "localhost." At the start of each game the application will request a word from the Hangman Word Server specified by IP address.

The word obtained from the Word Server will be designated as the "hidden" word the player tries to guess. If no connection can be made to a Word Server on the given IP address displayan error message on the console "Apparently no hangman word server is running at

<IP address>." If a word is obtained from the Word Server, game play begins by offering the player their first turn.

## Playing a game

At each turn the application will display a visual indicator of how many letters are in the hiddenword and if any of the letters have been correctly guessed they are shown in the proper positionin which they appear in the word. The application will display a "guess count" which shows how many incorrect guesses the player has made. An incorrect guess is guessing a letter whichis not in the hidden word.

The application will allow the player to enter a letter. If the letter entered is not between A andZ display a message "Invalid move" and allow the player to guess again (without penalty). When the player enters a valid letter, the application will check to see if the game is over and if not will continue to the next turn.

## Ending a game

The player wins by correctly guessing all the letters in the hidden word. The player loses if he/she makes seven incorrect guesses.

If the player wins the application will display a congratulatory message.

If the player loses the application will display a consolation message and will reveal the hiddenword.

When the game is over (either win or loss) the application will offer the player an opportunityto begin another game.

If the player indicates they want to play again, a new game is started.

If the player indicates they do not want to play again, the application is terminate

## Word Server Functional Requirements

Hangman Word Server runs as a standalone application independently from the Hangman Solitaire game.

When a client program makes a network connection to the server's port the server will respondby sending a character string then closing the connection.

## Hangman Non-Functional Requirements

General Guidelines: Performance and reliability are not very important. Priority should begiven to adaptability, maintainability, and usability.

## Operating Constraints

The program requires the JRE v1.4 with Swing from Sun Microsystems.

## Platform constraints

The program requires a 486 or higher processor with 16 Megabytes of RAM and 5 Megabytesof available hard drive space.

## Modifiability

If it is desired to change the number of turns in a game, the developer will be able to make the required changes in < 1 person-hours.

## Adaptability

The user must be able to specify an alternate user interface at execution time on the commandline.

Any alternate user interface must be able to "plug in" at run time without recompiling the application. Any alternate user interface must implement the Hangman UI interface provided by Dr. Dalbey (see External Interfaces).

## Documentation

A short (< 200 words) message explaining command line syntax should be displayed if theuser enters no command line parameters.

## Portability

The program will run on Win 95/98/XP and Red Hat Linux 9.

## Reliability

Since the program is purely for recreation and involves no user data, reliability is of low importance.

## Security

The program will not access any user data files or programs. The program will not alter or replace any system files.

## Usability

A new user should be able to play a complete game of hangman in less than ten minutes.

A new user should commit less than one error in use of the game (e.g., selecting the wrongletter) every ten minutes.

A user who is familiar with the rules of Hangman be able to correctly operate the programwithout any written documentation.

## Testability

(optional) If the user provides an IP address of zero then skip getting a word from the Word Server and use a default word "calpoly" as the hidden word.

## Deployment

The program is to be deployed as executable java bytecode files (".class" files) that must fit ona 1.2M flexible disk. They must use no specific java package.

## Performance

Desired Response Times (not critical): At game start: less than five seconds

## Word Server Non-Functional Requirements Performance

After receiving a request, the word server must respond in less than half a second.

## Maintainability

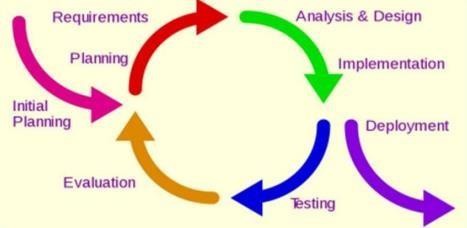
It is desirable that the game administrator be able to modify the words data file using a simpletext editor.

Adding a new word to the word data file should take less than ten minutes.

## Conceptual Model

1. **Incremental Model:**

Incremental Model is a process of software development where requirements are broken downinto multiple standalone modules of software development cycle. Incremental development isdone in steps from analysis design, implementation, testing/verification, maintenance.



Each iteration passes through the requirements, design, coding and testing phases. And each subsequent release of the system adds function to the previous release until all designed functionality has been implemented.

## Advantages*:*

* + The software will be generated quickly during the software life cycle.
  + It is flexible and less expensive to change requirements and scope.
  + Thought the development stages changes can be done.
  + This model is less costly compared to others.
  + A customer can respond to each building.
  + Errors are easy to be identified.

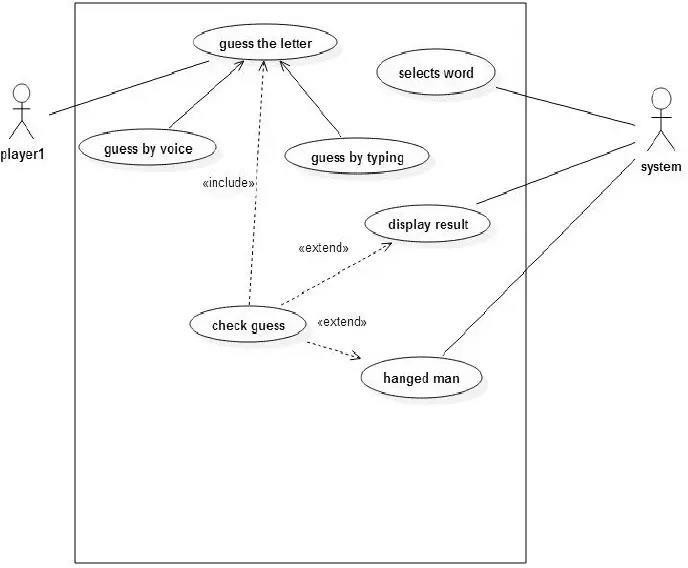
## Disadvantages*:*

* + It requires a good planning designing.
  + Problems might cause due to system architecture as such not all requirements collected upfront for the entire software lifecycle.
  + Each iteration phase is rigid and does not overlap each other.
  + Rectifying a problem in one unit requires correction in all the units and consumes a lot of ti**me.**

## Use Case Diagram

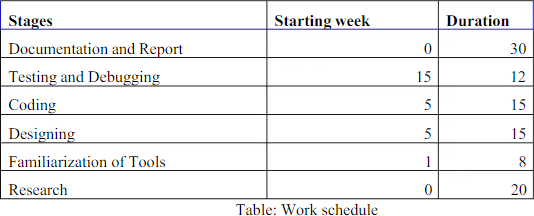
A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The usecases are represented by either circles or ellipses.

## single player use case diagram:



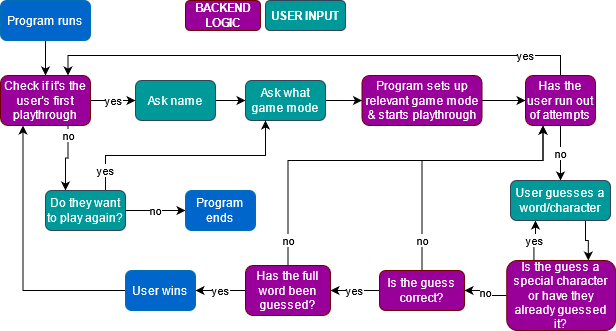
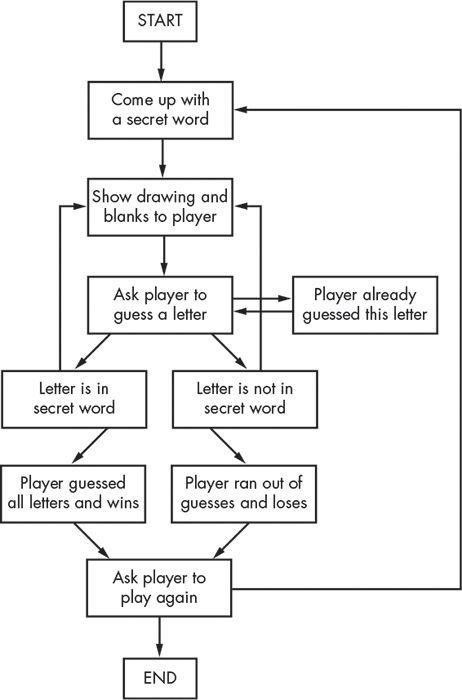
1. **Project Schedule**

The estimated time period of this project is six months. The work is divided into several phases as shown in Gantt chart. Research will be conducted within 1st twenty days. We are trying to get familiarized with tools for development of this project in the second week. The designing, coding and testing and debugging will be done in following weeks. The Documentation and Report will extend throughout the project period.



## Game Flowchart

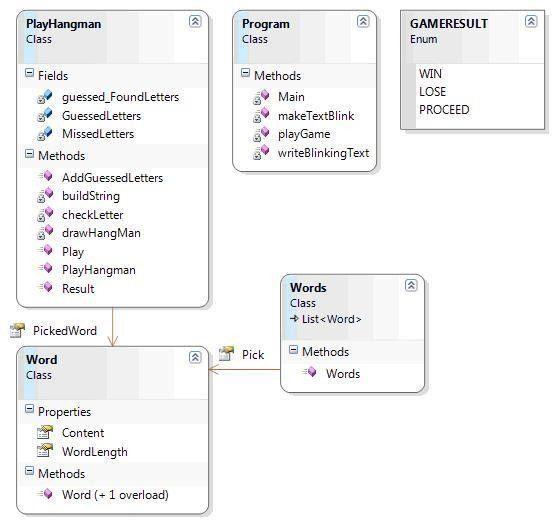
A flowchart is a picture of the separate steps of a process in sequential order. It is a generic tool that can be adapted for a wide variety of purposes, and can be used to describe various processes,such as a manufacturing process, an administrative or service process, or a project plan.



## F. Class diagram

UML CLASS DIAGRAM gives an overview of a software system by displaying classes, attributes, operations, and their relationships. This Diagram includes the class name, attributes, and operationin separate designated compartments.

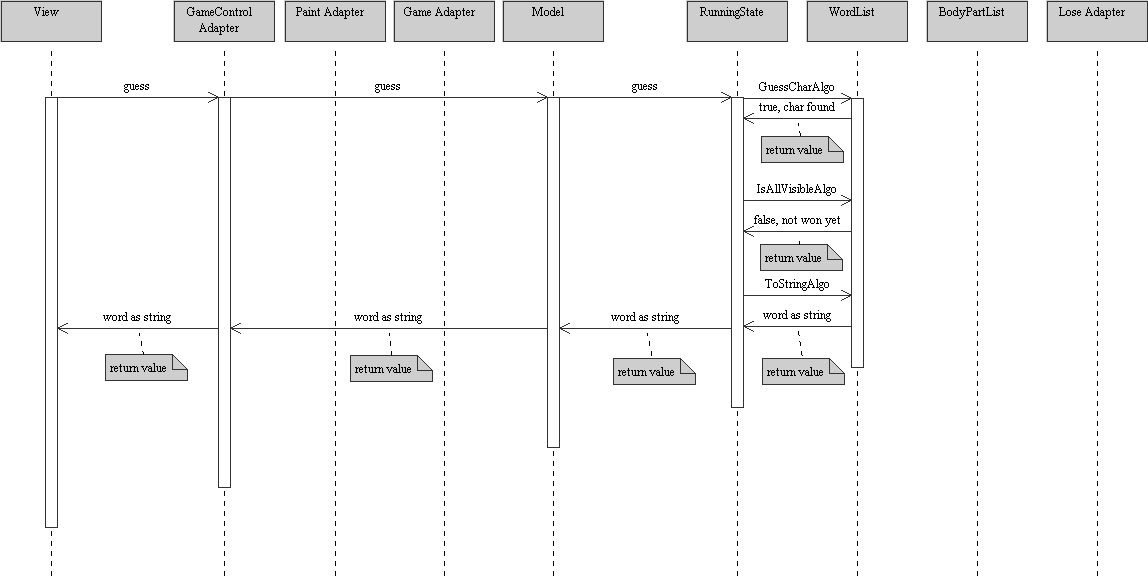
Class Diagram defines the types of objects in the system and the different types of relationships that exist among them. It gives a high-level view of an application. This modeling method can runwith almost all Object-Oriented Methods. A class can refer to another class. A class can have its objects or may inherit from other classes.



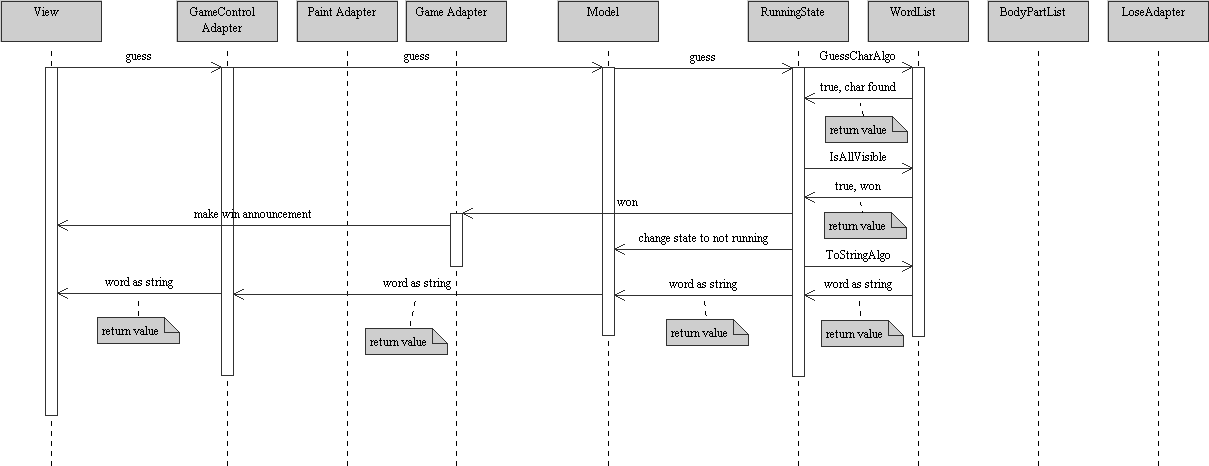
## G. Sequence Diagram

The UML class diagrams for the design of Hangman can only describe the static relationships between the various classes. To illustrate the dynamic interactions between the various objects inthe system at runtime, we use UML sequence diagrams. Below are UML sequence diagrams depicting different scenarios in playing the game.

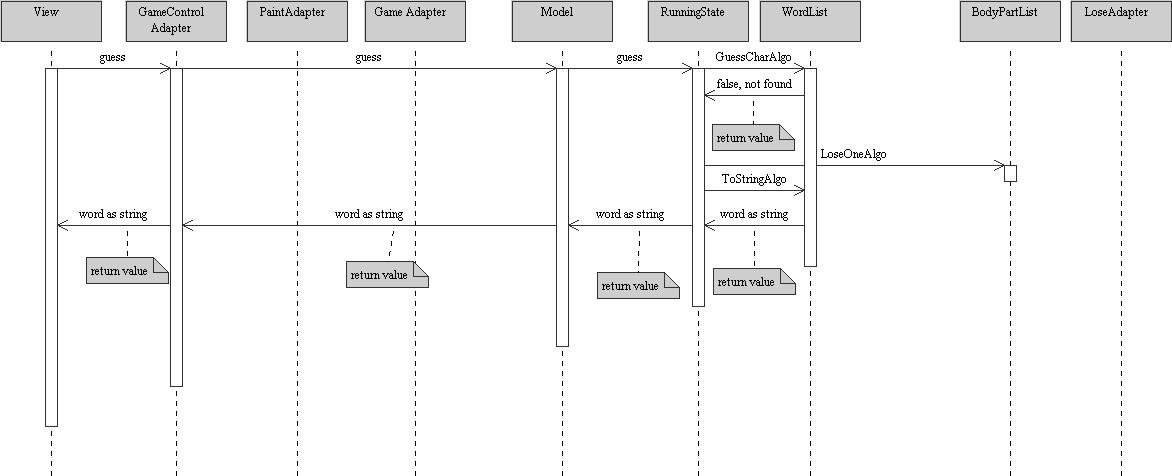
1. Correctly guess a character but still need to continue playing



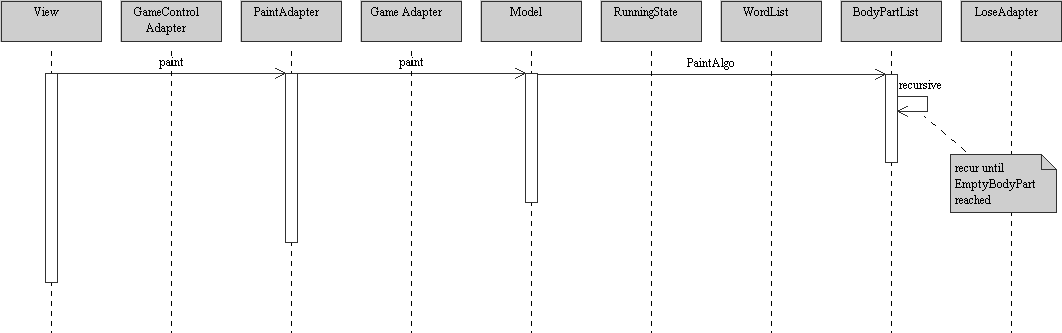
1. Correctly guess a character and win



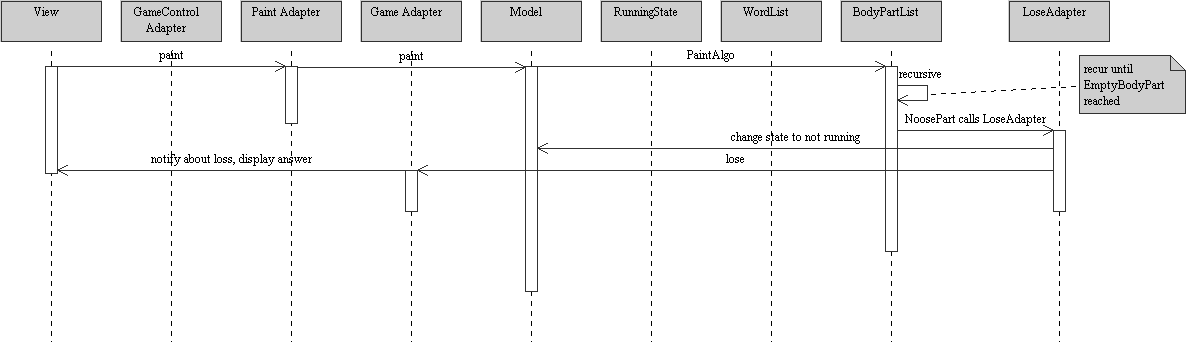
1. Incorrectly guess a character but still need to continue playing



1. Painting the body parts



1. The game is lost when the noose is drawn



# CHAPTER 05: IMPLEMENTATION & RESULTS

## Implementation Approach

This is a simple Hangman game using Python programming language. Beginners can use thisas a small project to boost their programming skills and understanding logic.

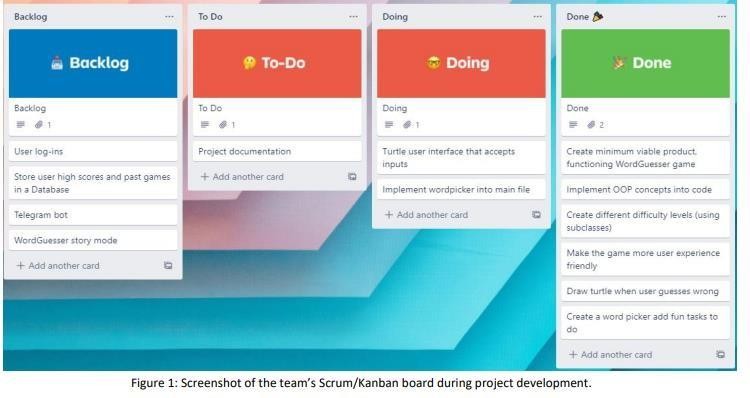
* + - The Hangman program randomly selects a secret word from a list of secret words. The random module will provide this ability, so line 1 in program imports it.
    - The Game: Here, a random word (a fruit name) is picked up from our collection and theplayer gets limited chances to win the game.
    - When a letter in that word is guessed correctly, that letter position in the word is made visible. In this way, all letters of the word are to be guessed before all the chances are over.
    - For convenience, we have given length of word + 2 chances. For example, word to be guessed is mango, then user gets 5 + 2 = 7 chances, as mango is a five-letter word.

The workload was distributed based on our individual skills and strengths. In addition, abreakdown of the project requirements specified by CFG helped to guide our approach and structure of the project.

The project approach revolved around an Agile mindset. Agile was decided upon rather than the other popular software development framework, Waterfall, because it allowed us to keep the project plan more fluid and susceptible to change. Waterfall requires a solid plan on system design and features right at the start of the project which is not feasible as there was not much time to come up with a plan, and not enough Python experience to estimate a timeframe.

Adopting Agile allowed the team to have a more dynamic approach to development, where a Minimum Viable Product (MVP) is established, and features can be gradually extended along with our increasing knowledge of Python. These features are taken from the product backlog, decided by the team, and implementedon a rolling basis until sufficient time before the before the project deadline. After which, the teamworked on finalizing the project document and submitting it along with the code.

Our main project management technique and the one which stood out the most was Kanban. Although this is not the traditional Scrum framework which was taught during the course, it still adopts an agile mindset. It was decided that using entirely Scrum would not be feasible because of the extreme focus on communication such as having daily stand-ups which did not fit around the schedules of team members who had other commitments such as full-time jobs. Hence, elements of both Scrum and Kanban were used.



An example of adopting an Agile framework is engaging in peer code reviews This was very important to the team’s success as it allowed every member to see the changes made by the other team members and keep up to speed with the latest code. We mainly reviewed code through pull requests on GitHub and organizing regular meetings in Teams. Another example of an Agile practice would be the Kanban/Scrum board we used.

## TOOLS & MODULES

The following tools were used to create the project:

* Project deployment and testing: The programme was written in the Python programming language and utilised various add-on packages.
* random – This module implements pseudo-random number generators for various distributions.Itwas used in the levels.py file to generate a random word from a given list.
* turtle – The turtle module provides turtle graphics primitives, in both object-oriented andprocedure- oriented ways. It allows shapes and pictures to be drawn easily. This was used to provide a type of user interface for a more enjoyable experience.
* nltk – NLTK is a Python library for symbolic and statistical natural language processing (NLP) to work with human language data. It was used to generate a list of words for the game.
* collections – This module implements specialized container datatypes providing alternatives to Python’s general purpose built-in containers, dict, list, set, and tuple. Counter from Collections was used in the word\_picker.py file to count the number of instances of unique letters in a word.
* json– JSON is a syntax for storing and exchanging data and the package can be used to work withJSON data. In this case, the word list was stored as JSON data.
* unittest – It supports test automation, sharing of setup and shutdown code for tests, aggregation oftests into collections, and independence of the tests from the reporting framework. It was used towrite unit tests for the different classes and methods.

Team collaboration and project storage:

* Git – A version control system used to facilitate collaborative work.
* GitHub – A cloud-based version control management system that uses Git to enable collaborationwithin a development team.

Communication:

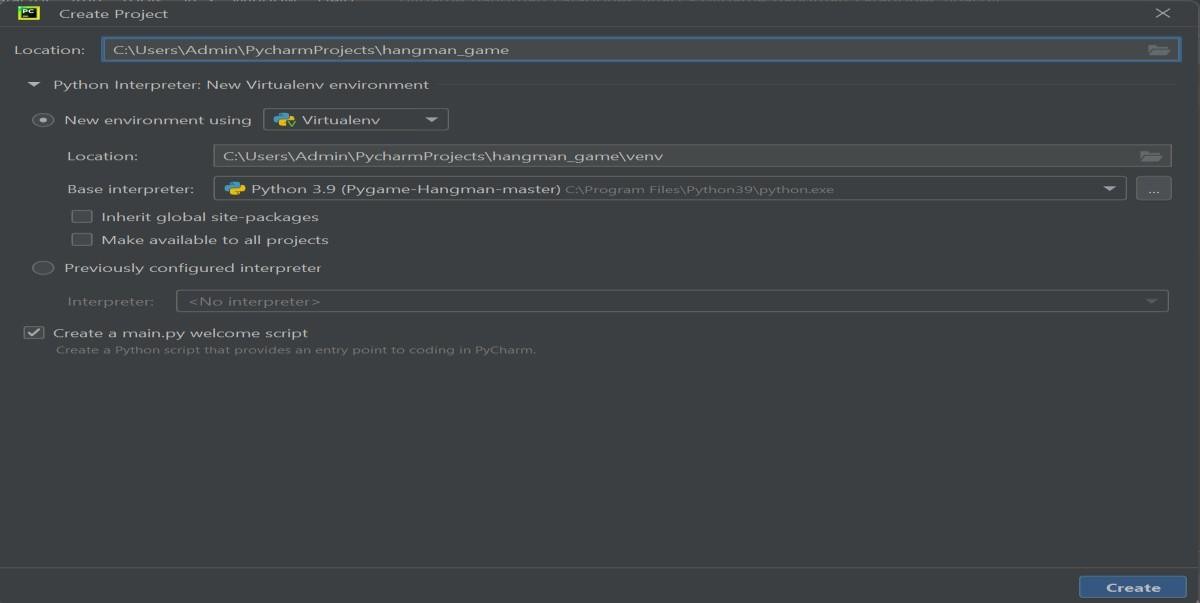
* Microsoft Teams – Used for scheduled meetings/debugging sessions and storing files (such as theproject report) to allow the team to access and work on the documents concurrently.
* Slack – Used for frequent chat communication and impromptu voice calls within the team andbetween the team and their designated instructor.
* Zoom – Used during schedule CFG class sessions with designated time allotted to project work.

## Code Efficiency

A Hangman Game in Python is about guessing letters (A-Z) to form the words. If the player guesses the right letter that is within the word, the letter appears at its correct position. The user has to guess the correct word until a man is hung, then the game is over.

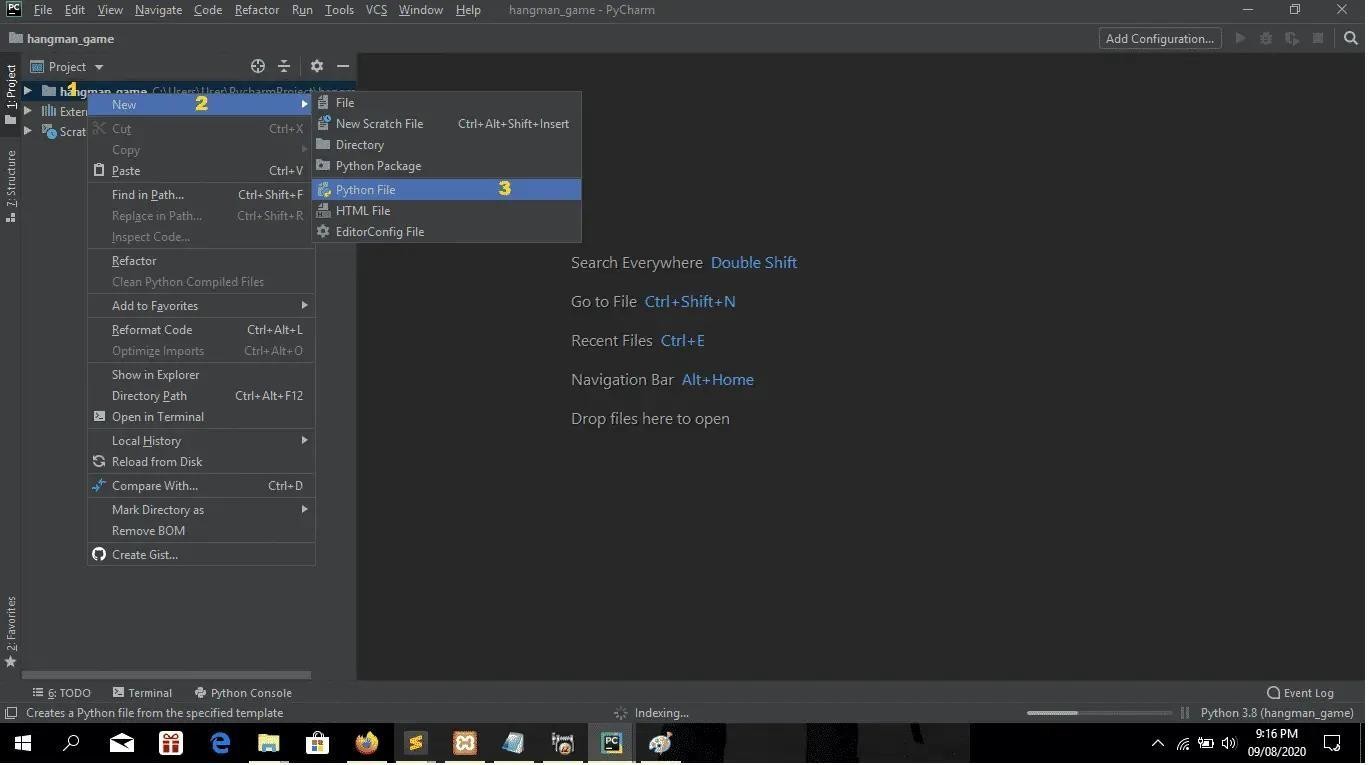
### Step 1: Create a project name.

First open **Pycharm IDE** and then create a “**project name**” after creating a project name click the “**create**” button.



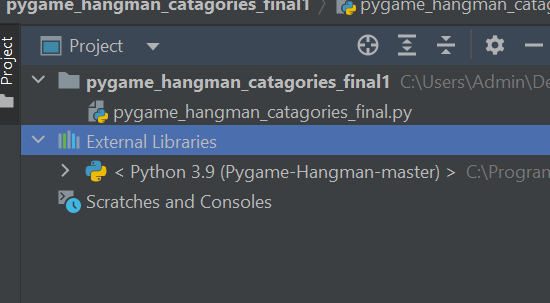
### Step 2: Create a python file.

Second after creating a project name, “**right click**” your project name and then click “**new**” after that click the “**python file**“.



### Step 3: Name your python file.

Third after creating a python file, Name your python file after that click “**enter**“.



**Step 4: The actual code.**

#ruya kumru-holroyd #pygame hangman

import pygame, sys, random from time import sleep

from pygame.locals import \*

from timeit import default\_timer as timer

fps = 30 pygame.init() width = 800

height = 700

black = (0,0,0)

white = (255,255,255)

lightred = (255, 165, 145)

darklightred = (255, 97, 81)

lightblue = (126,178,255)

darklightblue = (42, 129, 255)

lightgrey = (192, 192, 192)

darkred = (139,26,26)

darkgreen = (2, 48, 32)

textBoxSpace = 5

textBoxNumber = 0

def button(word,x,y,w,h,ic,ac,action=None): mouse = pygame.mouse.get\_pos()

click = pygame.mouse.get\_pressed()

if x+w > mouse[0] > x and y+h > mouse[1] > y: pygame.draw.rect(screen,ac,(x,y,w,h))

if click[0] == 1 and action != None: action()

else:

pygame.draw.rect(screen,ic,(x,y,w,h))

buttonText = pygame.font.Font("freesansbold.ttf",20) buttonTextSurf = buttonText.render(word, True, white) buttonTextRect = buttonTextSurf.get\_rect() buttonTextRect.center = ((x+(w/2)), (y+(h/2))) screen.blit(buttonTextSurf, buttonTextRect)

def endGame(pick):

global textBoxSpace, textBoxNumber, end, start end = timer()

print("Time it took: ",end - start)

timeTaken = (end - start) textBoxSpace = 5

textBoxNumber = 0

timeTakenMessage = "Time it took: " + str(round(timeTaken)) + "s" message = "Your word was: " + pick

while True:

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

screen.fill(darklightblue)

button("Yes",(width/2)-50,420,100,50,darkred,lightred,quitGame) button("Continue",(width/2)-50,500,100,50,darkred,lightred,selectLevel)

largeText = pygame.font.SysFont("comicsansms",50) TextSurf = largeText.render("End Game?",True,darkred) TextRect = TextSurf.get\_rect()

TextRect.center = (width / 2, 375) screen.blit(TextSurf, TextRect)

largeText = pygame.font.SysFont("comicsansms",65) TextSurf = largeText.render("Play Again!!!",True,black) TextRect = TextSurf.get\_rect()

TextRect.center = (width / 2, 50) screen.blit(TextSurf, TextRect)

textSurf = largeText.render(message,True,darkred) textRect = textSurf.get\_rect()

textRect.center = (width/2,150) screen.blit(textSurf, textRect)

textSurf = largeText.render(timeTakenMessage,True,darkred) textRect = textSurf.get\_rect()

textRect.center = (width/2,200) screen.blit(textSurf, textRect) pygame.display.update() clock.tick(fps)

def quitGame(): pygame.quit()

sys.exit()

def unpause(): global pause pause = False

def pause():

largeText = pygame.font.SysFont("comicsansms",115) TextSurf = largeText.render("Paused",True,black) TextRect = TextSurf.get\_rect()

TextRect.center = (width / 2, height / 2) screen.blit(TextSurf, TextRect)

while pause:

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

screen.fill(white)

button("Continue",150,450,100,50,darklightred,lightred,unpause) button("Quit",550,450,100,50,darklightblue,lightblue,quitgame)

pygame.display.update() clock.tick(fps)

def textObjects(text, font):

textSurface = font.render(text, True, black) return textSurface, textSurface.get\_rect()

def main():

global clock, screen, play play = True

clock = pygame.time.Clock()

screen = pygame.display.set\_mode((width, height)) pygame.display.set\_caption("Hangman!")

while True:

welcomeScreen()

def placeLetter(letter): global pick, pickSplit

space = 10

wordSpace = 0

while wordSpace < len(pick):

text = pygame.font.Font('freesansbold.ttf',40) if letter in pickSplit[wordSpace]:

textSurf = text.render(letter,True,black) textRect = textSurf.get\_rect() textRect.center = (((150)+space),(200)) screen.blit(textSurf, textRect)

wordSpace += 1

space += 60

pygame.display.update() clock.tick(fps)

def textBoxLetter(letter):

global textBoxSpace, textBoxNumber if textBoxNumber <= 5:

text = pygame.font.Font("freesansbold.ttf",40) textSurf = text.render(letter,True,black) textRect = textSurf.get\_rect()

textRect.center = (((105)+textBoxSpace),(350)) screen.blit(textSurf, textRect)

elif textBoxNumber <= 10:

text = pygame.font.Font("freesansbold.ttf",40) textSurf = text.render(letter,True,black) textRect = textSurf.get\_rect()

textRect.center = (((105)+textBoxSpace),(400)) screen.blit(textSurf, textRect)

elif textBoxNumber <= 15:

text = pygame.font.Font("freesansbold.ttf",40) textSurf = text.render(letter,True,black) textRect = textSurf.get\_rect()

textRect.center = (((105)+textBoxSpace),(450)) screen.blit(textSurf, textRect)

elif textBoxNumber <= 20:

text = pygame.font.Font("freesansbold.ttf",40) textSurf = text.render(letter,True,black) textRect = textSurf.get\_rect()

textRect.center = (((105)+textBoxSpace),(500)) screen.blit(textSurf, textRect)

pygame.display.update() clock.tick(fps)

def welcomeScreen():

global textBoxSpace, textBoxNumber textBoxSpace = 5

textBoxNumber = 0 while play == True:

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

screen.fill(darkgreen) space = 10

textBoxSpace = 5

text = pygame.font.Font("freesansbold.ttf",70) textSurf = text.render("HangMan GAME!!",True,white) textRect = textSurf.get\_rect()

textRect.center = ((width/2),(height/4)) screen.blit(textSurf, textRect)

# button("Play",325,350,150,100,black,lightgrey,selectLevel) button("Play",350,250,100,50,darkred,lightgrey,selectLevel)

pygame.display.update() clock.tick(fps)

def selectLevel():

global textBoxSpace, textBoxNumber textBoxSpace = 5

textBoxNumber = 0 while play == True:

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

screen.fill(lightblue) space = 10

textBoxSpace = 5

text = pygame.font.Font("freesansbold.ttf",50)

textSurf = text.render("Select level",True,darklightblue) textRect = textSurf.get\_rect()

textRect.center = ((width/2),(height/8)) screen.blit(textSurf, textRect)

button("Easy",325,150,150,100,black,lightgrey,Easy) button("Medium",325,350,150,100,black,lightgrey,Medium) button("Hard",325,550,150,100,black,lightgrey,Hard)

# button("Medium",550,450,150,100,black,lightgrey,hangman) # button("Hard",150,50,150,100,black,lightgrey,hangman)

pygame.display.update() clock.tick(fps)

def Easy():

global textBoxSpace, textBoxNumber textBoxSpace = 5

textBoxNumber = 0 while play == True:

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

screen.fill(lightblue) space = 10

textBoxSpace = 5

text = pygame.font.Font("freesansbold.ttf",50)

textSurf = text.render("Choose a catagory",True,darklightblue) textRect = textSurf.get\_rect()

textRect.center = ((width/2),(height/2)) screen.blit(textSurf, textRect)

button("Animals",350,450,170,70,black,lightgrey,Animals) button("Food", 350, 120, 160, 70, black, lightgrey, Foods)

button("Back", 50, 50, 100, 50, darkred, lightgrey, selectLevel)

pygame.display.update() clock.tick(fps)

def Medium():

global textBoxSpace, textBoxNumber textBoxSpace = 5

textBoxNumber = 0 while play == True:

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

screen.fill(lightblue) space = 10

textBoxSpace = 5

text = pygame.font.Font("freesansbold.ttf", 50)

textSurf = text.render("Choose a catagory", True, darklightblue) textRect = textSurf.get\_rect()

textRect.center = ((width / 2), (height / 2)) screen.blit(textSurf, textRect)

button("Cars", 550, 150, 150, 70, black, lightgrey, Cars)

button("Brands", 560, 550, 150, 80, black, lightgrey, Brands)

button("Trees", 100, 550, 150, 80, black, lightgrey, Trees)

button("Fruits", 80, 150, 150, 70, black, lightgrey, Fruits)

button("Back", 50, 50, 100, 50, darkred, lightgrey, selectLevel)

pygame.display.update() clock.tick(fps)

def Hard():

global textBoxSpace, textBoxNumber textBoxSpace = 5

textBoxNumber = 0 while play == True:

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

screen.fill(lightblue) space = 10

textBoxSpace = 5

text = pygame.font.Font("freesansbold.ttf", 20)

textSurf = text.render("Choose a catagory", True, darklightblue)

textRect = textSurf.get\_rect() textRect.center = ((width / 2), (height / 2)) screen.blit(textSurf, textRect)

button("Temples", 110, 150, 200, 60, black, lightgrey, Temples)

button("Sports", 500, 150, 150, 60, black, lightgrey, Sports)

button("Movies", 100, 550, 150, 60, black, lightgrey, Movies)

button("Countries", 350, 370, 150, 60, black, lightgrey, Countries)

button("Colours", 550, 550, 150, 60, black, lightgrey, Colours)

button("Back", 50, 50, 100, 50, darkred, lightgrey, selectLevel) pygame.display.update()

clock.tick(fps)

def hangmanGame(catagory,title):

global pause, pick, pickSplit, textBoxSpace, textBoxNumber, start start = timer()

chances = 20

pick = random.choice(catagory)

pickSplit = [pick[i:i+1] for i in range(0, len(pick), 1)] screen.fill(darklightblue)

wordSpace = 0

space = 10

while wordSpace < len(pick):

text = pygame.font.Font("freesansbold.ttf",40) textSurf1 = text.render("\_",True,black) textRect1 = textSurf1.get\_rect() textRect1.center = (((150)+space),(200)) screen.blit(textSurf1, textRect1)

space = space + 60 wordSpace += 1

guesses = '' gamePlay = True

while gamePlay == True: guessLett = ''

if textBoxNumber == 5: textBoxSpace = 5

if textBoxNumber == 10: textBoxSpace = 5

if textBoxNumber == 15: textBoxSpace = 5

pygame.draw.rect(screen, white, [550,20,200,20]) text = pygame.font.Font("freesansbold.ttf",20)

textSurf = text.render(("Chances: %s" % chances),False,black) textRect = textSurf.get\_rect()

textRect.topright = (700,20) screen.blit(textSurf, textRect)

textTitle = pygame.font.Font("freesansbold.ttf",40) textTitleSurf = textTitle.render(title,True,black) textTitleRect = textTitleSurf.get\_rect() textTitleRect.center = ((width/2),50) screen.blit(textTitleSurf, textTitleRect)

pygame.draw.rect(screen, black, [100,300,250,250],2) if chances == 19:

pygame.draw.rect(screen,black,[450,550,100,10])

elif chances == 18: pygame.draw.rect(screen,black,[550,550,100,10])

elif chances == 17: pygame.draw.rect(screen,black,[650,550,100,10])

elif chances == 16: pygame.draw.rect(screen,black,[500,450,10,100])

elif chances == 15: pygame.draw.rect(screen,black,[500,350,10,100])

elif chances == 14: pygame.draw.rect(screen,black,[500,250,10,100])

elif chances == 13:

pygame.draw.rect(screen,black,[500,250,150,10]) elif chances == 12:

pygame.draw.rect(screen,black,[600,250,100,10]) elif chances == 11:

pygame.draw.rect(screen,black,[600,250,10,50]) elif chances == 10:

pygame.draw.line(screen,black,[505,505],[550,550],10) elif chances == 9:

pygame.draw.line(screen,black,[550,250],[505,295],10) elif chances == 8:

pygame.draw.line(screen,black,[505,505],[460,550],10) elif chances == 7:

pygame.draw.circle(screen,black,[605,325],30) elif chances == 6:

pygame.draw.rect(screen,black,[600,350,10,60])

elif chances == 5: pygame.draw.rect(screen,black,[600,410,10,60])

elif chances == 4: pygame.draw.line(screen,black,[605,375],[550,395],10)

elif chances == 3: pygame.draw.line(screen,black,[605,375],[650,395],10)

elif chances == 2: pygame.draw.line(screen,black,[605,465],[550,485],10)

elif chances == 1: pygame.draw.line(screen,black,[605,465],[650,485],10)

button("Back",50,50,100,50,black,lightgrey,selectLevel) button("Quit",650,50,100,50,darkred,darklightred,quitGame)

for event in pygame.event.get(): if event.type == pygame.QUIT:

pygame.quit() sys.exit()

if event.type == pygame.KEYDOWN: failed = 0 print("Failed",failed) print("Chance", chances)

if event.key == pygame.K\_SPACE: pause()

if event.key == pygame.K\_ESCAPE: gamePlay = False

#letter a

if event.key == pygame.K\_a:

guessLett = guessLett + 'a'

guesses += guessLett print("letter a guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick:

placeLetter('a')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('a')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter b

if event.key == pygame.K\_b:

guessLett = guessLett + 'b' guesses += guessLett print("letter b guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('b')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("")

print("That letter is not in the word") textBoxLetter('b')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter c

if event.key == pygame.K\_c:

guessLett = guessLett + 'c' guesses += guessLett print("letter c guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('c')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('c')

if chances == 0:

print("Sorry you have lost") print("The word was",pick)

endGame(pick)

#letter d

#letter e

if event.key == pygame.K\_d:

guessLett = guessLett + 'd' guesses += guessLett print("letter d guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('d')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('d')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

if event.key == pygame.K\_e:

guessLett = guessLett + 'e' guesses += guessLett print("letter e guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('e')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('e')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter f

if event.key == pygame.K\_f:

guessLett = guessLett + 'f' guesses += guessLett print("letter f guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('f')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('f')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter g

if event.key == pygame.K\_g:

guessLett = guessLett + 'g' guesses += guessLett print("letter g guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('g')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('g')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter h

if event.key == pygame.K\_h:

guessLett = guessLett + 'h' guesses += guessLett print("letter h guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('h')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('h')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter i

#letter j

if event.key == pygame.K\_i:

guessLett = guessLett + 'i' guesses += guessLett print("letter i guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('i')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('i')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

if event.key == pygame.K\_j:

guessLett = guessLett + 'j' guesses += guessLett

print("letter j guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('j')

if failed == 0:

print("You got the word") print(pick)

#gamePlay = False endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('j')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) #gamePlay = False endGame(pick)

#letter k

if event.key == pygame.K\_k:

guessLett = guessLett + 'k' guesses += guessLett print("letter k guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('k')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('k')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter l

if event.key == pygame.K\_l:

guessLett = guessLett + 'l' guesses += guessLett print("letter l guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('l')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('l')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter m

if event.key == pygame.K\_m:

guessLett = guessLett + 'm' guesses += guessLett print("letter m guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('m')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber)

print("")

print("That letter is not in the word") textBoxLetter('m')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter n

if event.key == pygame.K\_n:

guessLett = guessLett + 'n' guesses += guessLett print("letter n guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('n')

if failed == 0:

print("You got the word") print(pick)

#gamePlay = False endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('n')

if chances == 0:

print("Sorry you have lost") print("The word was",pick)

#gamePlay = False endGame(pick)

#letter o

#letter p

if event.key == pygame.K\_o:

guessLett = guessLett + 'o' guesses += guessLett print("letter o guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('o')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('o')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

if event.key == pygame.K\_p:

guessLett = guessLett + 'p' guesses += guessLett print("letter p guessed")

print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('p')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('p')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter q

if event.key == pygame.K\_q:

guessLett = guessLett + 'q' guesses += guessLett print("letter q guessed")

print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('a')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('q')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter r

if event.key == pygame.K\_r:

guessLett = guessLett + 'r' guesses += guessLett print("letter r guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('r')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('r')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter s

if event.key == pygame.K\_s:

guessLett = guessLett + 's' guesses += guessLett print("letter s guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('s')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word")

textBoxLetter('s')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter t

#letter u

if event.key == pygame.K\_t:

guessLett = guessLett + 't' guesses += guessLett print("letter t guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('t')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('t')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

if event.key == pygame.K\_u:

guessLett = guessLett + 'u' guesses += guessLett print("letter u guessed")

print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('u')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('u')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter v

if event.key == pygame.K\_v:

guessLett = guessLett + 'v' guesses += guessLett print("letter v guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_")

failed += 1

if guessLett in pick: placeLetter('v')

if failed == 0:

print("You got the word")

print(pick) endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('v')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

if event.key == pygame.K\_w:

#letter w

guessLett = guessLett + 'w' guesses += guessLett print("letter w guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('w')

if failed == 0:

print("You got the word")

print(pick) endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('w')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter x

if event.key == pygame.K\_x:

guessLett = guessLett + 'x' guesses += guessLett print("letter x guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('x')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber)

print("")

print("That letter is not in the word") textBoxLetter('x')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter y

if event.key == pygame.K\_y:

guessLett = guessLett + 'y' guesses += guessLett print("letter y guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('y')

if failed == 0:

print("You got the word") print(pick)

endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('y')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

#letter z

if event.key == pygame.K\_z:

guessLett = guessLett + 'z' guesses += guessLett print("letter z guessed") print("")

for char in pick:

if char in guesses: print(char)

else:

print("\_") failed += 1

if guessLett in pick: placeLetter('z')

if failed == 0:

print("You got the word")

print(pick) endGame(pick)

if guessLett not in pick: textBoxSpace += 40

textBoxNumber += 1 chances = chances - 1 print("") print(textBoxNumber) print("")

print("That letter is not in the word") textBoxLetter('z')

if chances == 0:

print("Sorry you have lost") print("The word was",pick) endGame(pick)

pygame.display.update() clock.tick(fps)

pygame.display.update() clock.tick(fps)

def Animals():

Animals = ['cow','dog','cat','pig','zebra','bird','giraffe','lion','tiger','penguin','hamster','

fox','panda','bear','cheetah','ostrich','meerkat','whale','shark','horse','monkey','oc topus','kitten','kangaroo','chicken','fish','rabbit','sheep']

print("animal") title = "Animals"

hangmanGame(Animals,title)

def Foods():

Foods = ['apple','banana','orange','peach','pizza','donut','chips','sandwich','cookie','cucumb er','carrot','sweetcorn','ice cream','pancake','bread','potato','tomato','nuts','yogurt','pasta','rice','cheese','so up','fish','egg','meat','ham','sausage']

print("food") title = "Foods"

hangmanGame(Foods,title) def Cars():

Cars = ['Bonbon','lovebug','buffie','zippy','itsy','lamborghini ','ferrari ','carrera ','sparky','nano','honda','bmw','audi','toyoto','alfa romeo','renault','mercedes benz','volkswagen beetle','camry','dodge viper','chevy','pip']

print("Cars")

title = "Cars" hangmanGame(Cars,title)

def Brands():

Brands = ['adidas','nike','microsoft','google','apple','calvin klein','walmart','samsung','disney','coca cola','mcdonalds','ibm','cisco','marlboro','nescafe','gillette','hermes','verizon','am azon','tata','infosys','reliance','lic','nike','louis vuitton','hermes','toyoto','hdfc']

print("Brands") title = "Brands"

hangmanGame(Brands,title)

def Trees():

Trees = ['banyan','nutmeg','rosewood','neem','babul','peepal','sandal','mangrove','lindens','a cacia','cedar','willow','tamarind','jackfruit','myrtle','eucalyptus','redwood','giant sequio','baobab','coconut palm','red maple','rhododendron','sycamore','yucca','hyperion','greatbasin bristlecone']

print("Trees") title = "Trees"

hangmanGame(Trees,title)

def Fruits():

Fruits = ['mulberry','pineapple','watermelon','raspberry','satsuma','nectarine','sapodilla','po megranate','papya','olive','jackfruit','peaches','mango','strawberries','limes','lemon s','oranges','jujube','tangerine','turnip','lychee','dragon fruit','musk melon','carambola','sweet potato','avacoda','custard apple','sweet lime']

print("Fruits") title = "Fruits"

hangmanGame(Fruits,title) def Temples():

Temples = ['badrinath','kedarnath','jagannath','golden

temple','akshardham','somnath','siddhivinayak','dwarkadish','sri ranganathaswamy','brihadeeswara','kasi vishwanath','yamnotri','meenakshi','kanchipuram','shridi sai baba','triupati balaji','shri padmanabhaswamy','virupaksha','lingaraja','mundeshvari','gobekli tepe','yamunotri','kornak sun','vaishno devi']

print("Temples") title = "Temples"

hangmanGame(Temples,title)

def Sports():

Sports = ['rugby', 'football', 'netball', 'basketball', 'swimming', 'hockey', 'curling', 'running', 'golf', 'tennis',

'badmington', 'archery', 'volleyball', 'bowling', 'dancing', 'gym', 'skating', 'baseball', 'rounders',

'boxing', 'climbing', 'canoe', 'cycling', 'fencing', 'karate', 'shooting', 'cricket']

print("Sports") title = "Sports"

hangmanGame(Sports, title)

def Movies():

Movies = ['pathaan','dil se','ram setu','brahmastra','gully boy','sooryavanshi','bajrangi bhaijaan','goodbye','phuspa','kedarnath','kati patang','jeevan mrityu','bajirao mastani','samrat prithviraj','heropanti','devdas','sholay','parvarish','shiddat','bahubali','kgf','octo pus','kitten','raja harishchandra','dangal','avengers','avatar','kisan kanya']

print("Movies") title = "Movies"

hangmanGame(Movies,title)

def Countries():

Countries = ['united states','australia','germany','france','canada','argentina','china','czechia','afghani stan','cambodia','burkina faso','costa rica','guinea bissau','india','indonesia','kazakhstan','mauritius','new zealand','philippines','south korea','sri lanka','switzerland','united kingdom','vietnam','zimbabwe','thailand','russia','nigeria']

print("Countries") title = "Countries"

hangmanGame(Countries,title)

def Colours(): Colours =

['corel','gold','moccasin','lavendar','violet','orchid','mageta','indigo','macron','fu chsia','crimson','chartreuse','emrald','beige','ivory','sapphire','burgundy','medallio n','mahogany','fushcia','boysenberry','amethyst','purple','black','green','white']

print("Colours") title = "Colours"

hangmanGame(Colours,title)

if name == ' main ': main()

* 1. **Testing**

The rules of hangman have changed with respect to words that contain the same letter multipletimes. Instead of all instances of the letter being found when the letter is played, only a single instanceneeds to be displayed.

For example, in the word banana, if the player plays the letter ‘a’ as the first move of the hangmangame then the game could display either:

* --------a or

November 2011 CSC7302: Testing & Metrics AdvancedTestingTechniques.1

* --------a or
* ---a-- or
* a

Your task is to analyses your design and identify which classes/methods need to be updated in order to correctly implement these new rules.

**The hangman Implementation**

Now, take the hangman code of a colleague:

* Implement the new rules by changing their code
* Implement the new tests by changing their code
* Execute the tests
* How good was their design? November 2011 CSC7302: Testing & Metrics AdvancedTesting Techniques.2
* How good was their design?
* Did you have to change something that was not anticipated by the original developer?
* Did you change something that you did not have to change?
* Which metrics have changed as a result of this update**?**
  1. **Test Case**

import unittest

from hangman import HangmanGame

class TestHangmanGame(unittest.TestCase)

def setUp(self):

self.game = HangmanGame('banana') def test\_guess\_correct\_letter(self): self.game.guess('a')

self.assertEqual(self.game.get\_display\_word(), '\_a\_a\_a')

def test\_guess\_incorrect\_letter(self): self.game.guess('z')

self.assertEqual(self.game.get\_remaining\_guesses(), 5) def test\_guess\_same\_letter\_twice(self):

self.game.guess('a') self.game.guess('a')

self.assertEqual(self.game.get\_remaining\_guesses(), 6) def test\_win\_game(self):

self.game.guess('b') self.game.guess('a')

self.game.guess('n') self.assertEqual(self.game.check\_game\_status(), 'win'

def test\_lose\_game(self):

self.game.guess('z') self.game.guess('y') self.game.guess('x') self.game.guess('w') self.game.guess('v') self.game.guess('u')

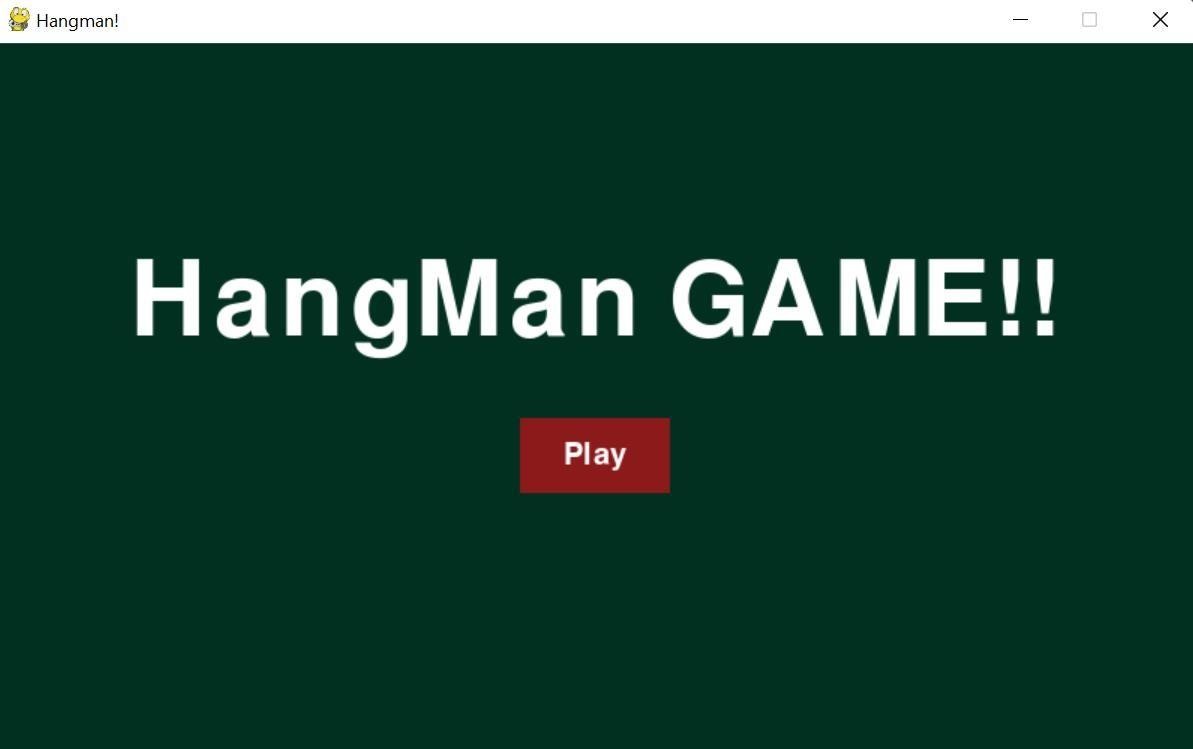
self.assertEqual(self.game.check\_game\_status(), 'lose')

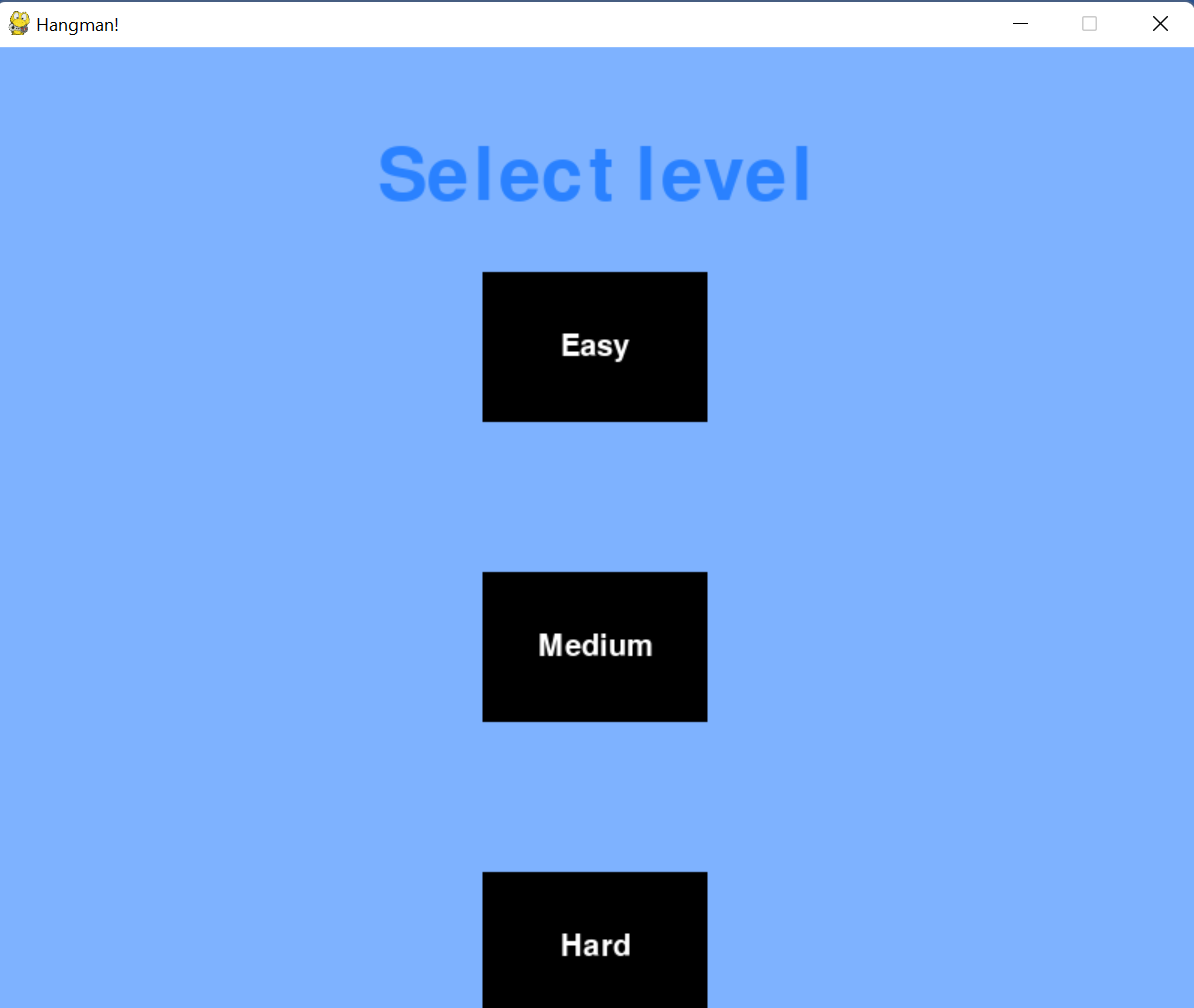
we are testing a HangmanGame class, which takes a word as input and allows the user to guess letters until they either win or lose the game.

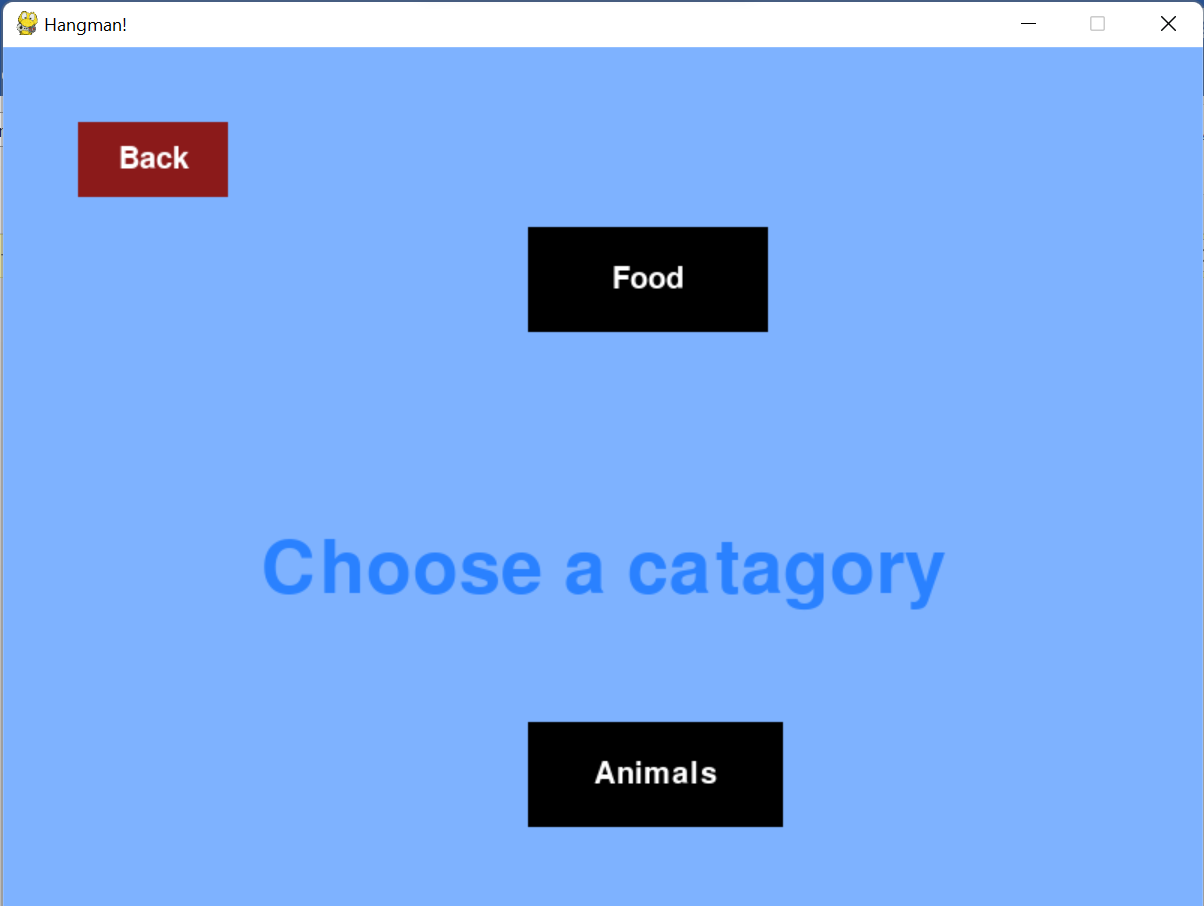
The set Up method creates a new game instance with the word 'banana'. The remaining methods are individual tests that check various aspects of the game's behavior, such as correctly guessing aletter, incorrectly guessing a letter, guessing the same letter twice, winning the game, and losing the game.

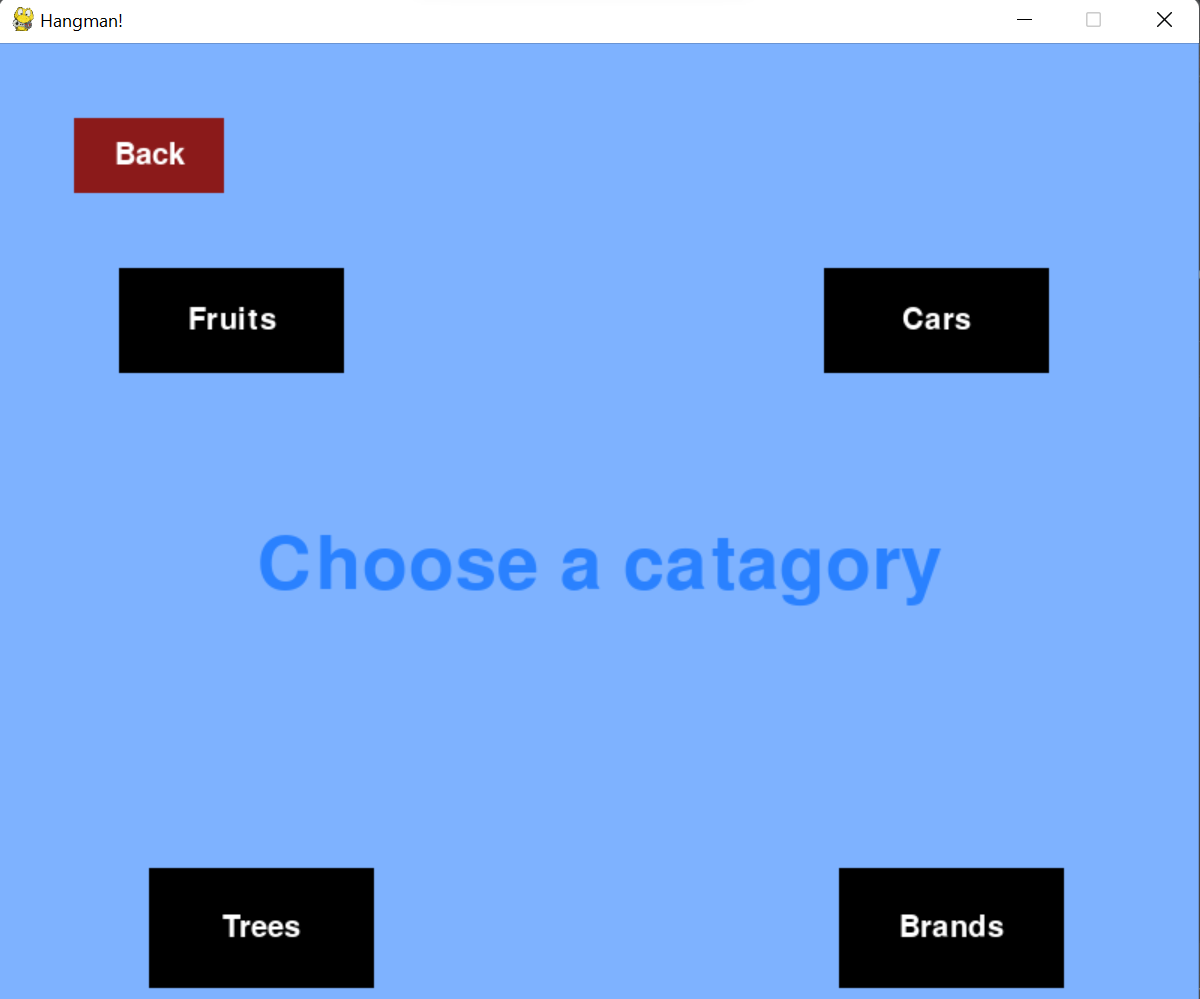
Each test uses assertions to ensure that the game behaves as expected. For example, in the test\_win\_game method, we guess the letters 'b', 'a', and 'n', which should reveal the entire word and result in a win. We then use assertEqual to check that the game's check\_game\_status methodreturns 'win'. Similarly, in the test\_lose\_game method, we guess six incorrect letters, which shouldresult in a loss, and use assertEqual to check that the game's check\_game\_status method returns 'lose'.

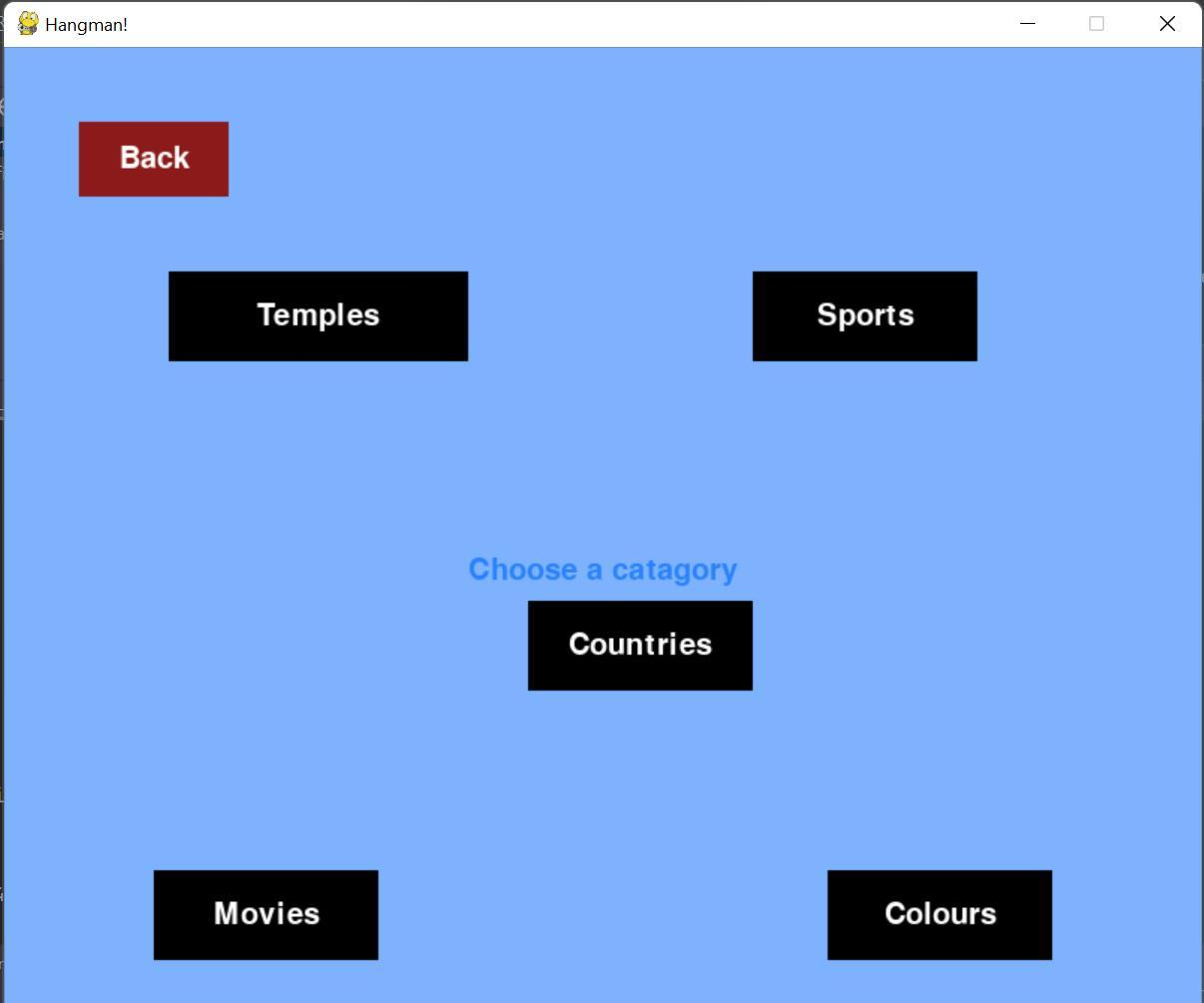
## Test Reports

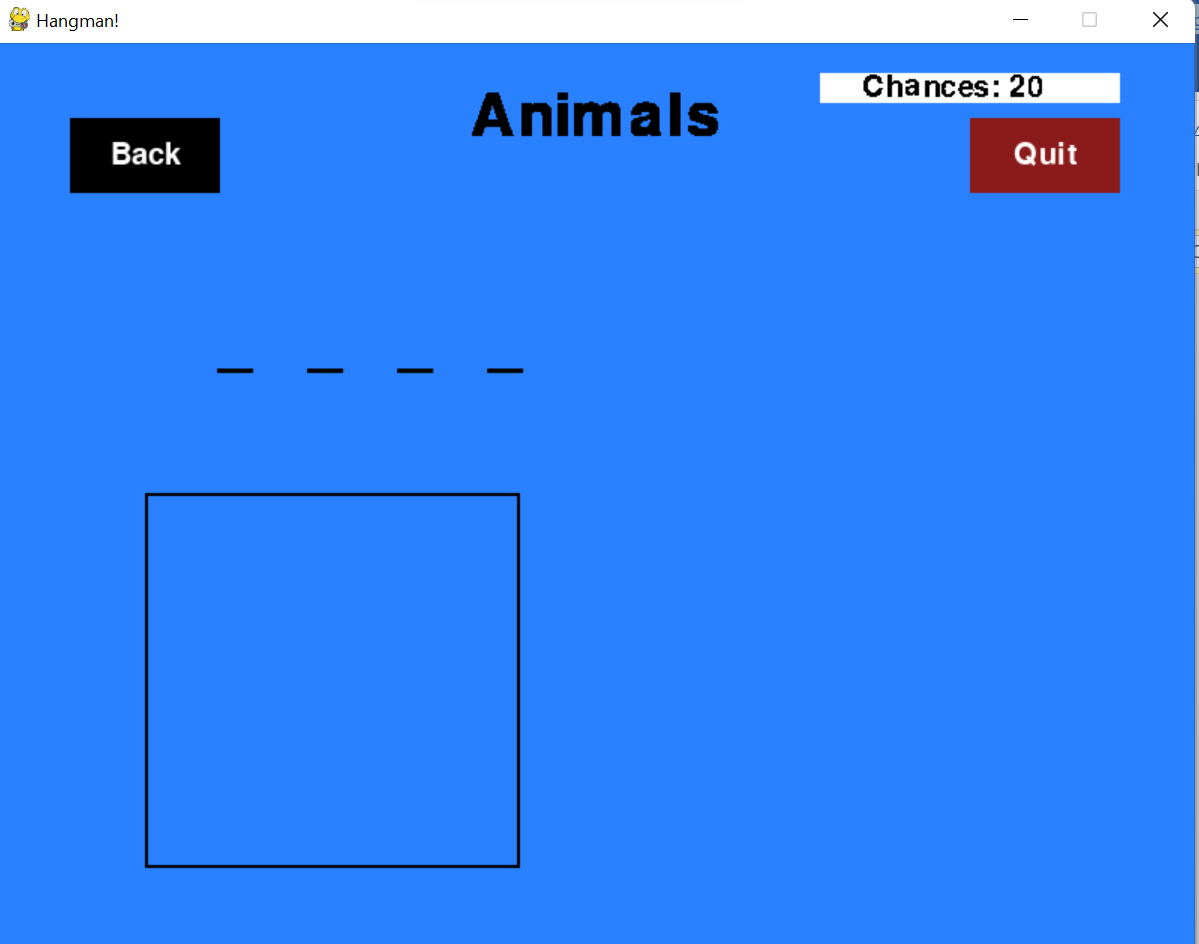
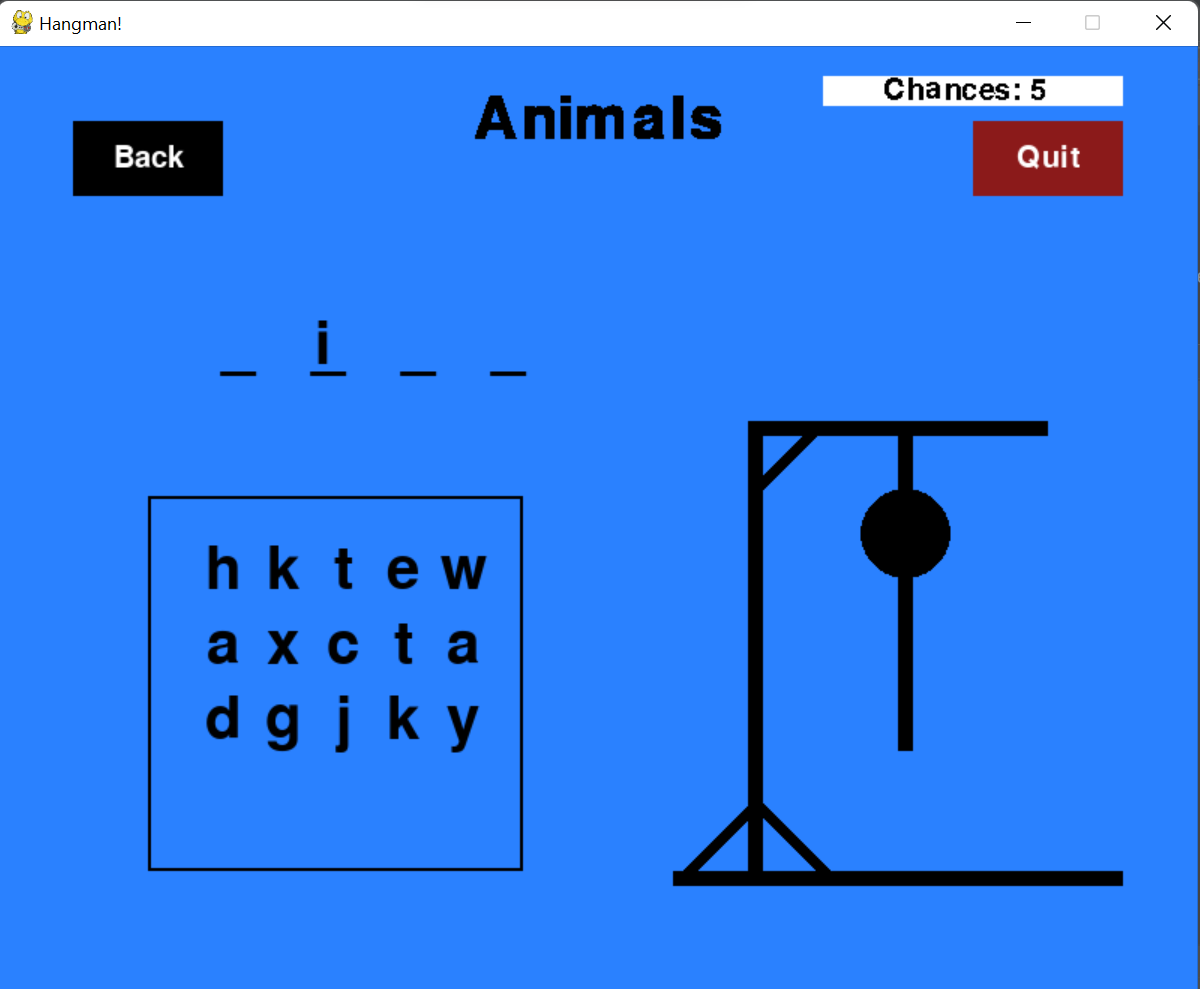


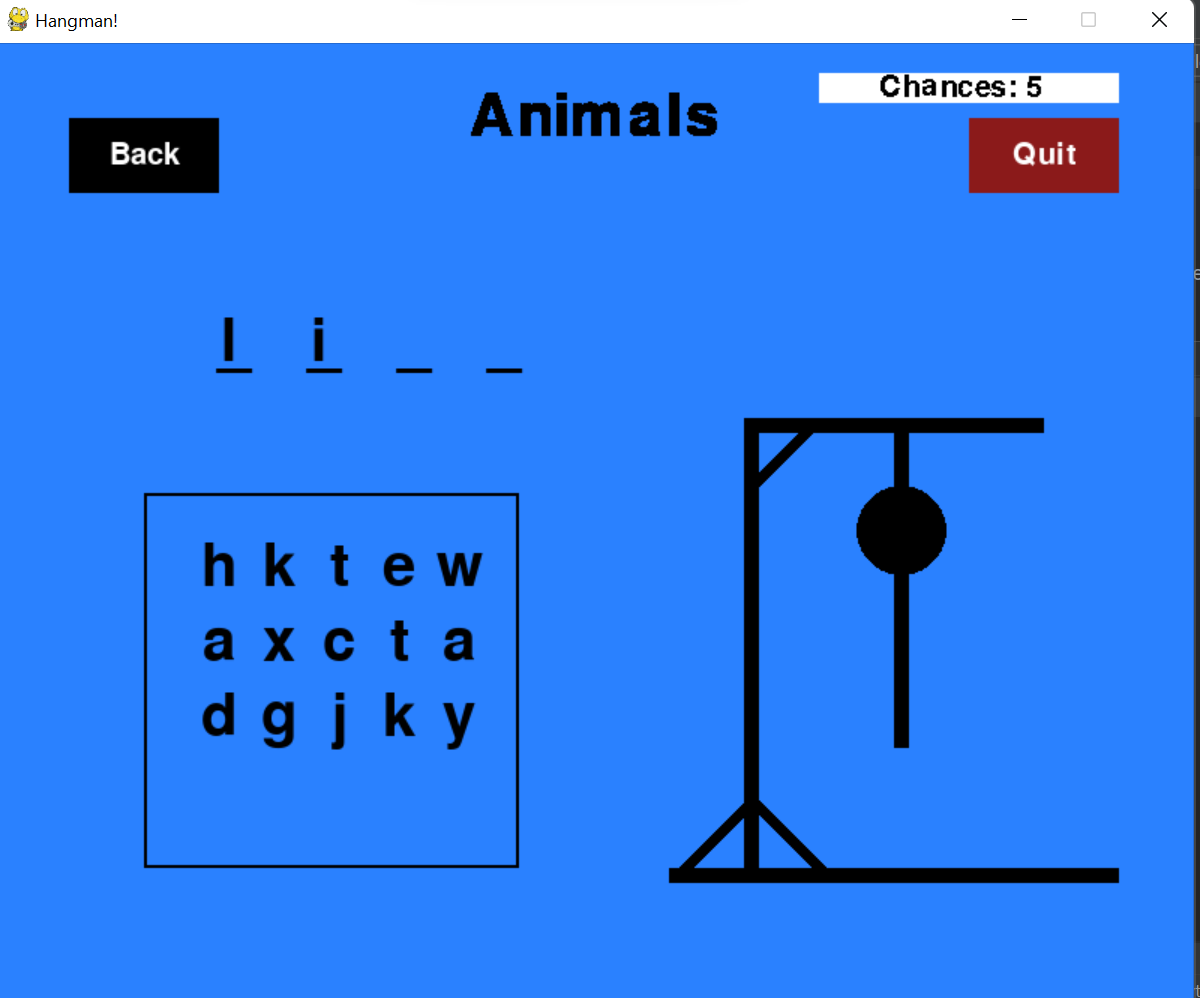


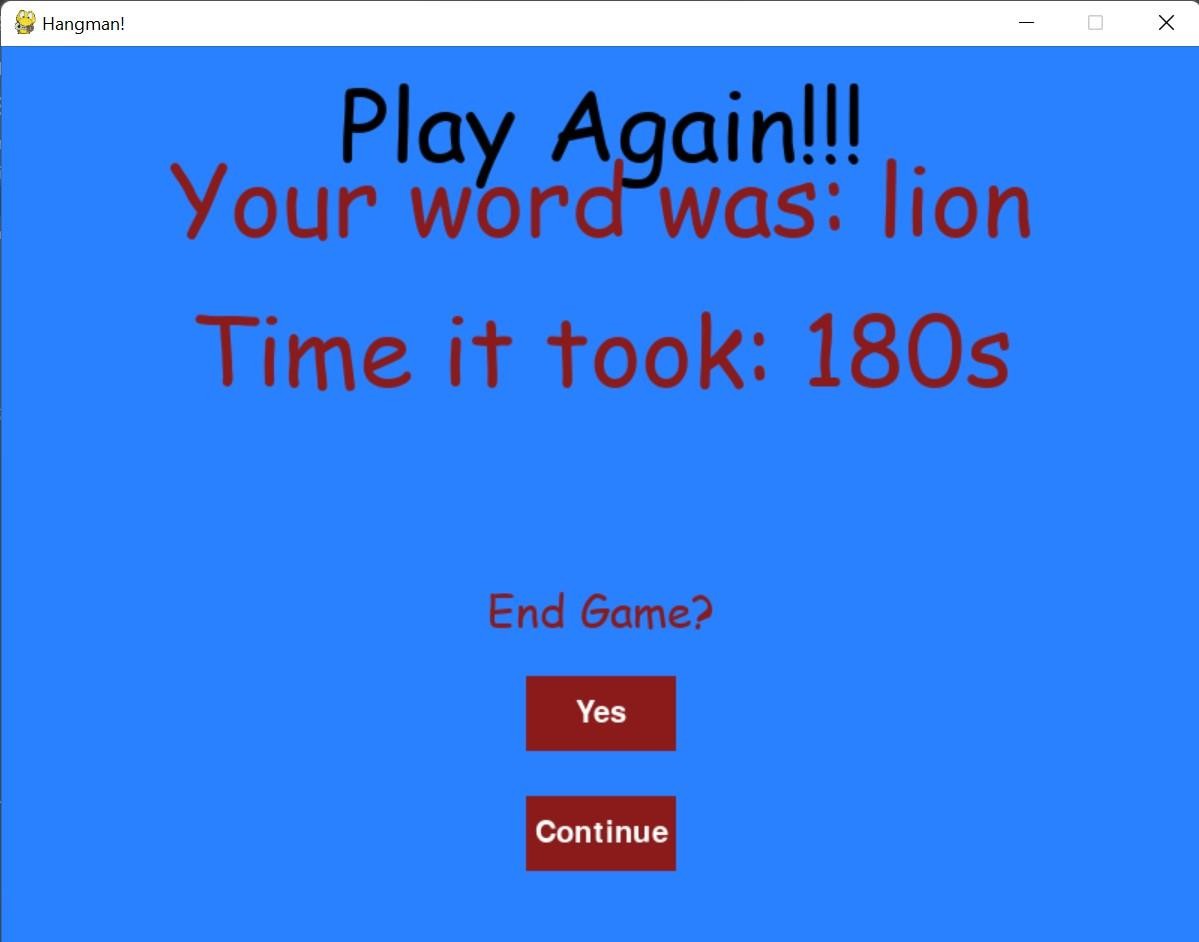
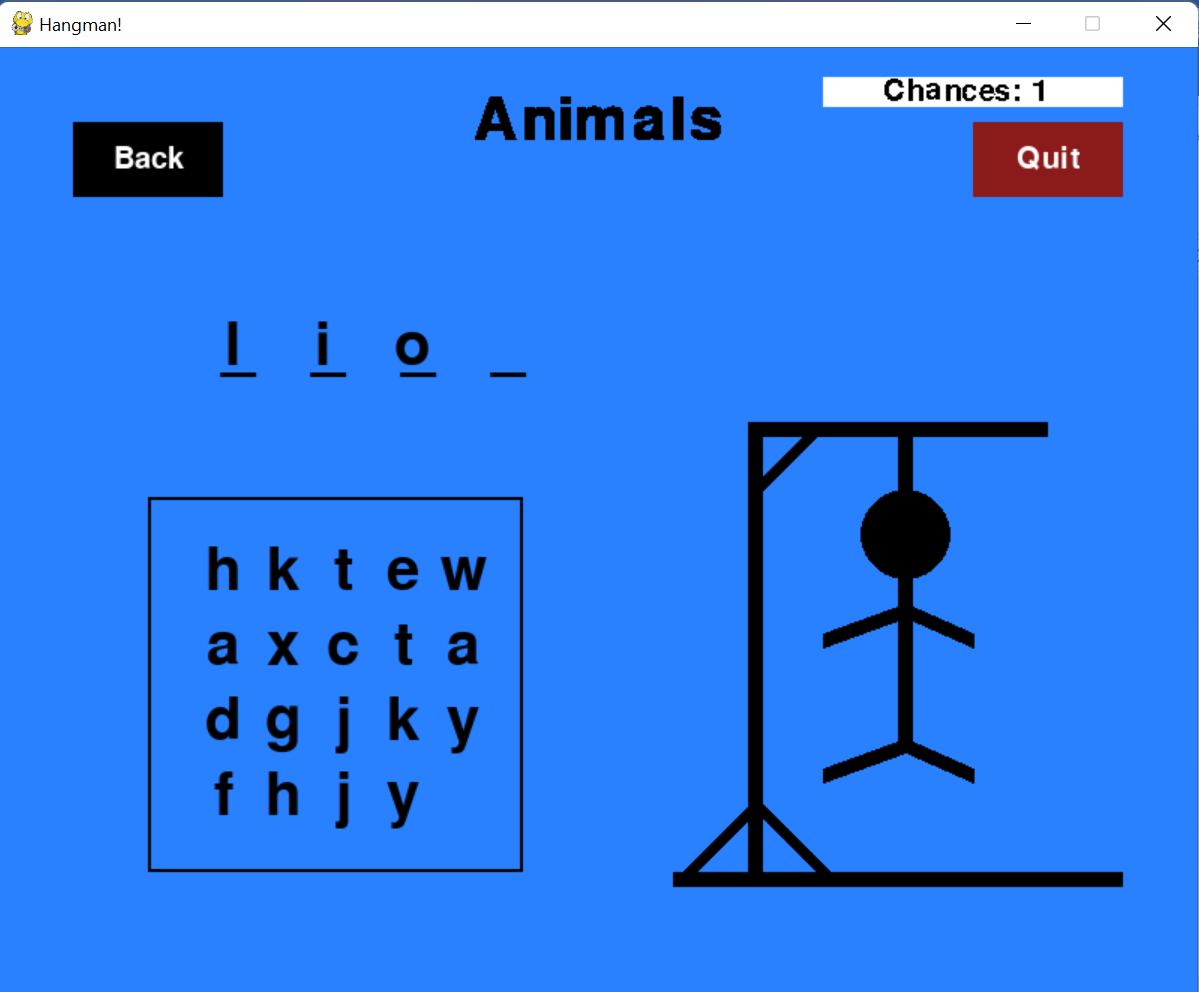


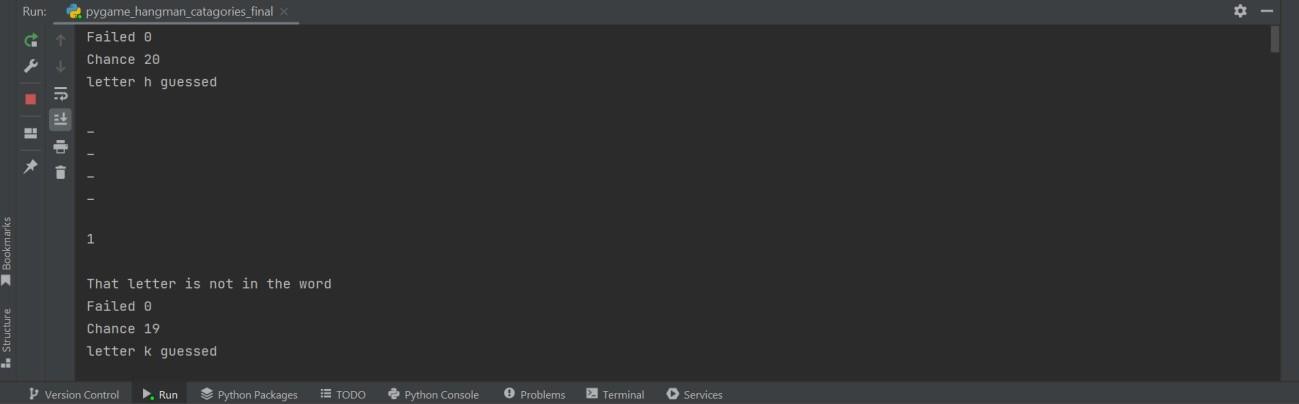


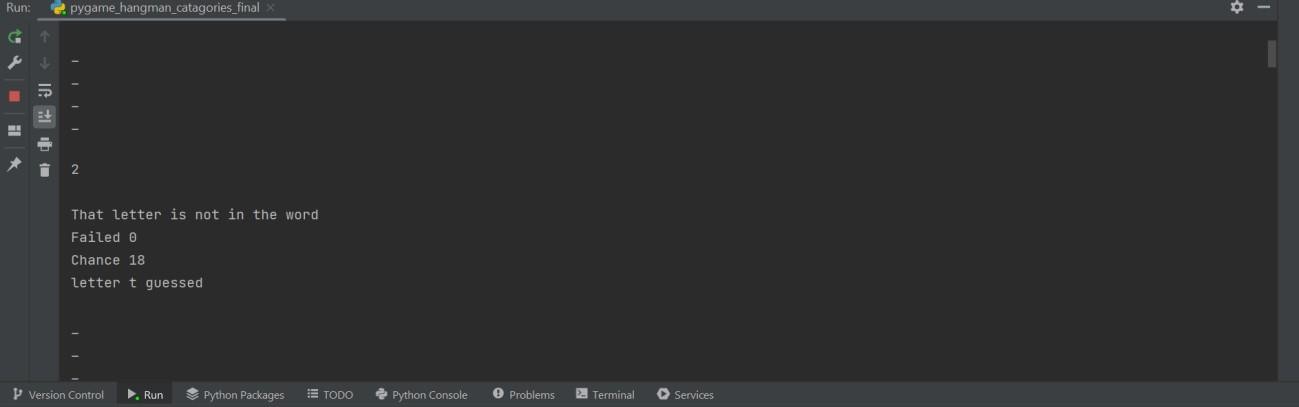


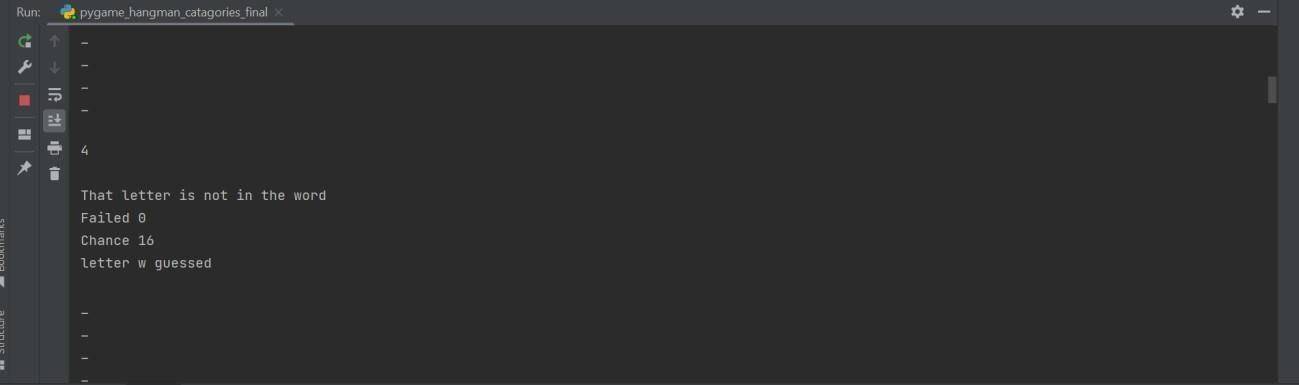
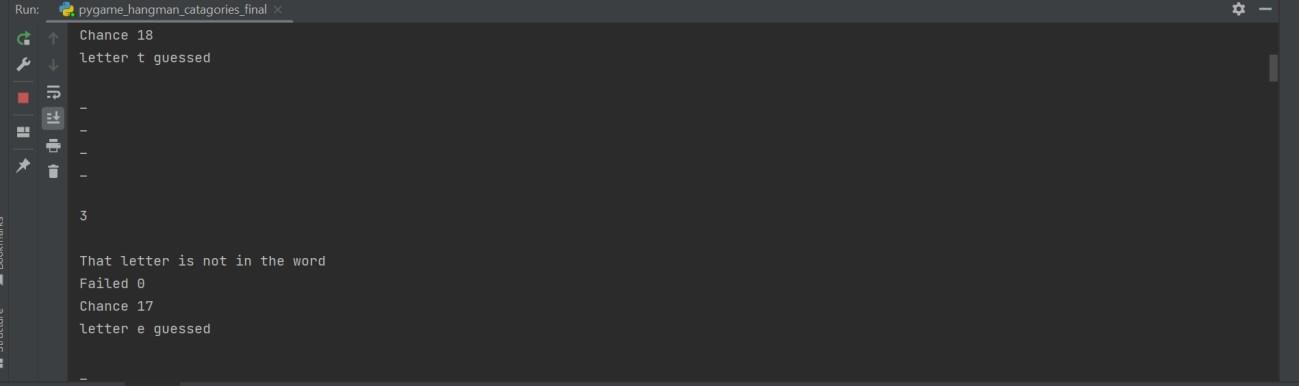


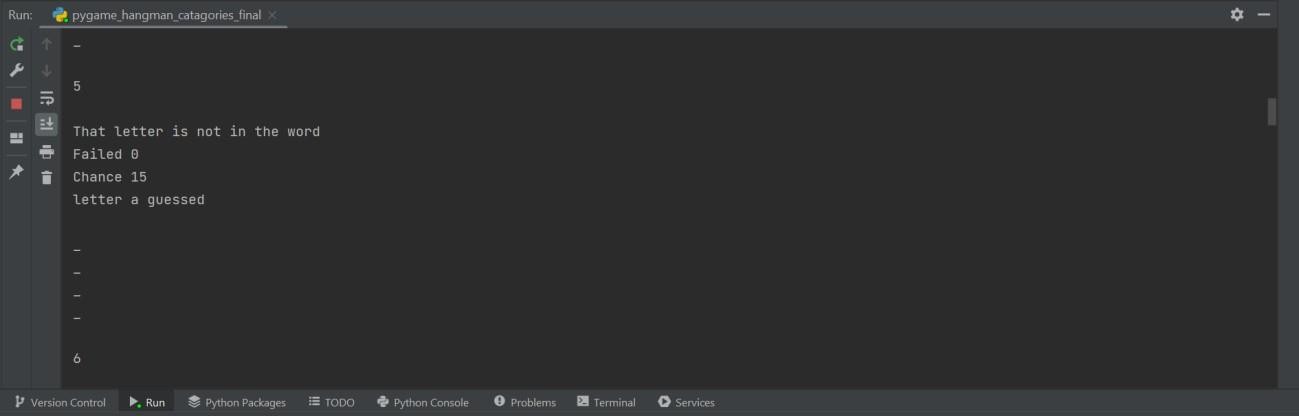


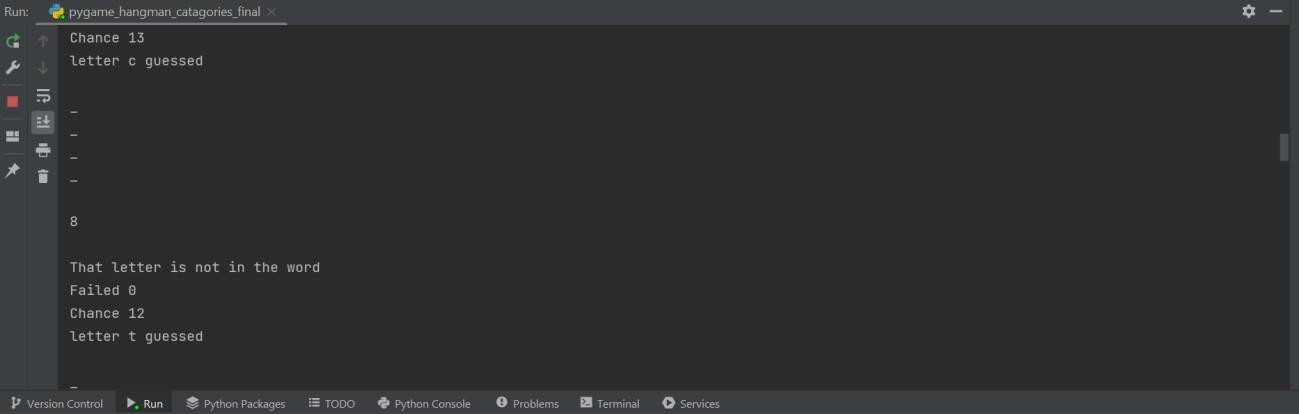
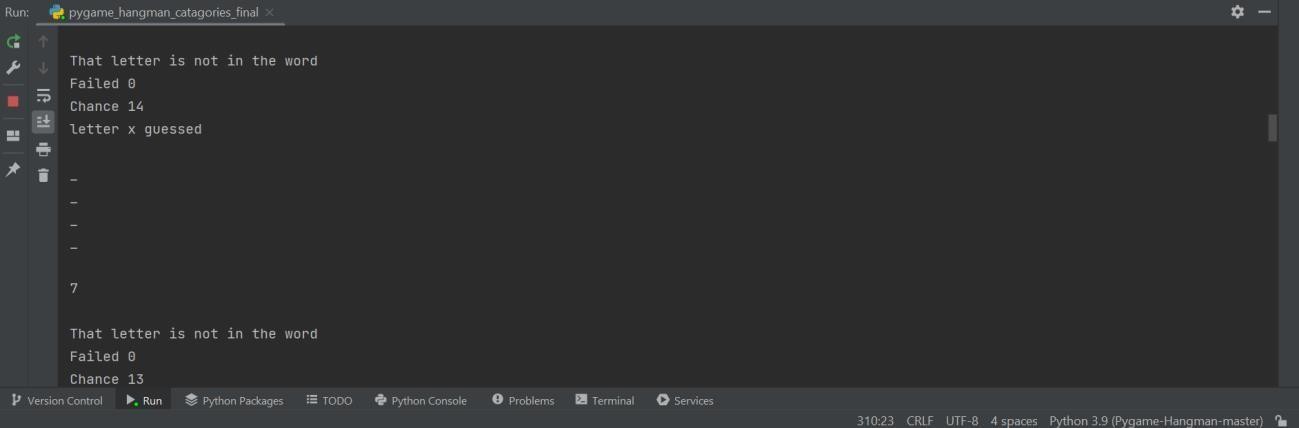


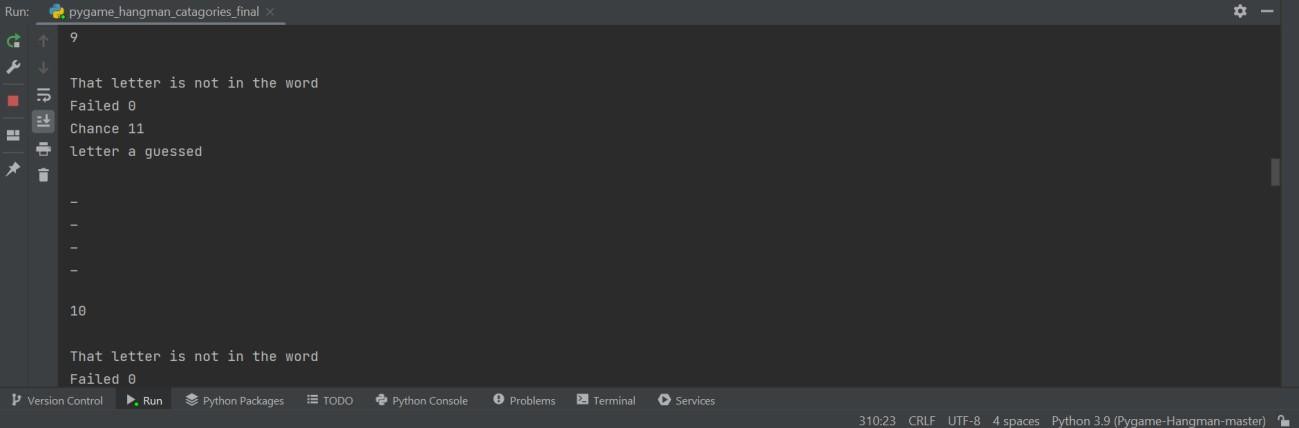


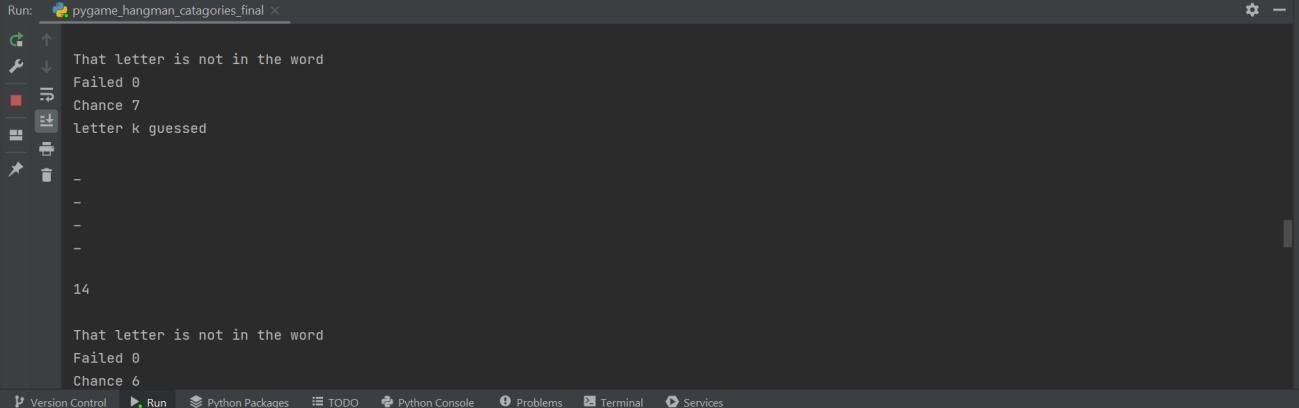


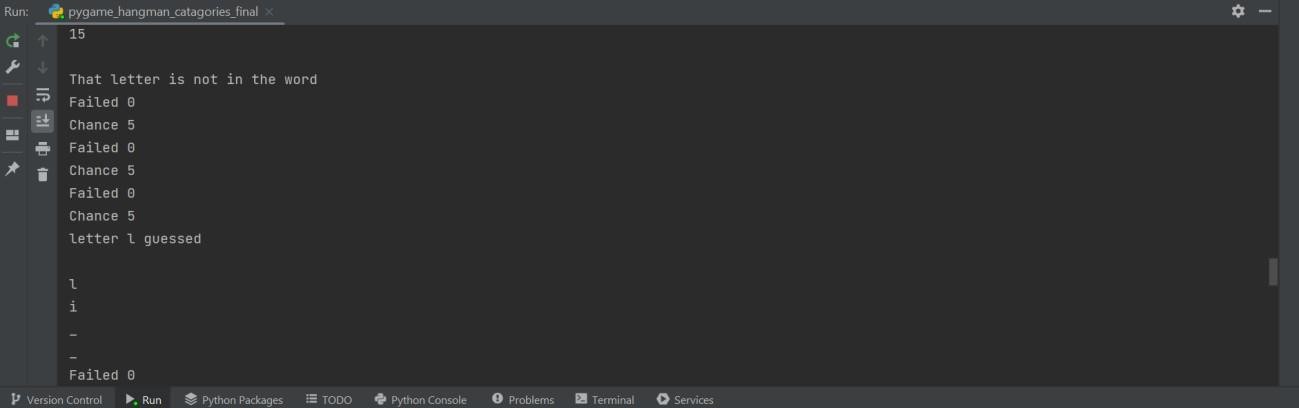
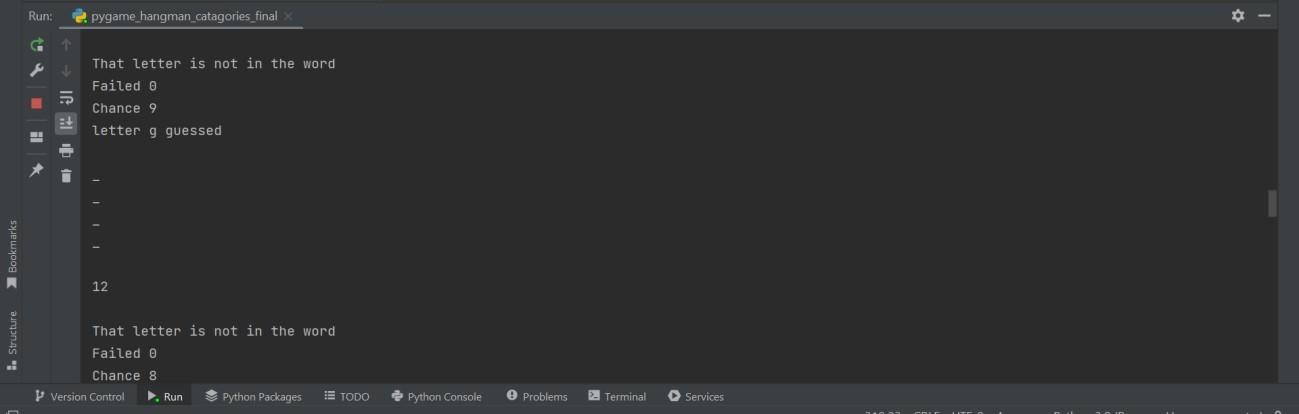
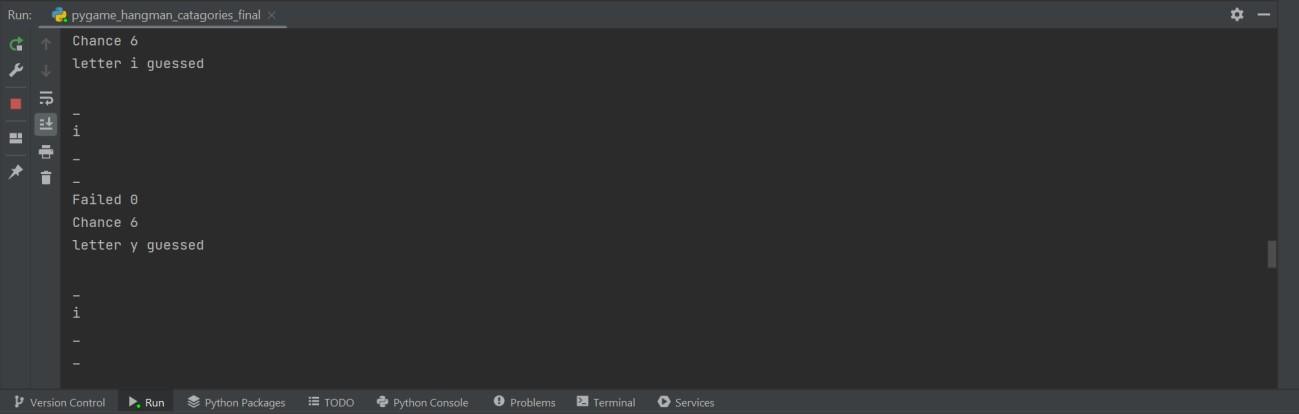


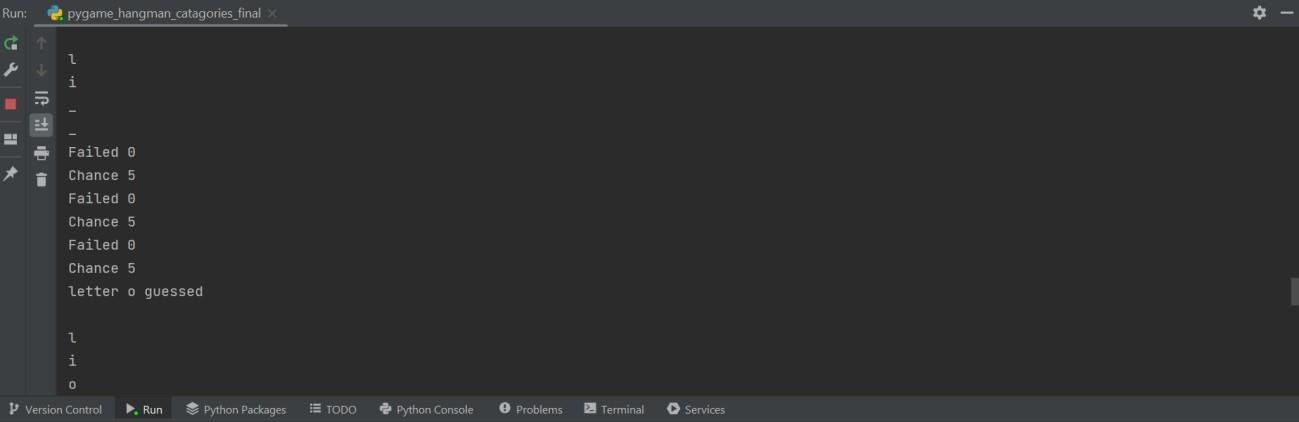


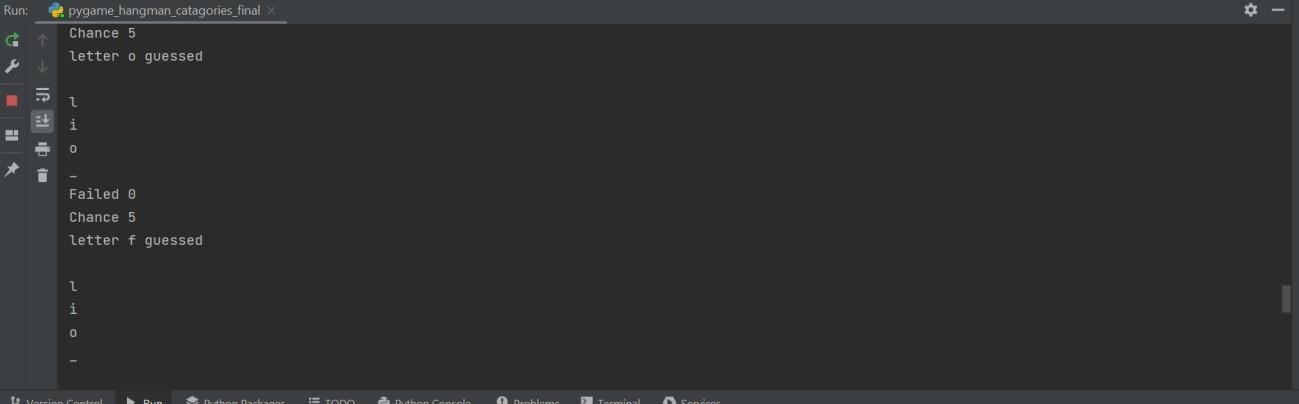


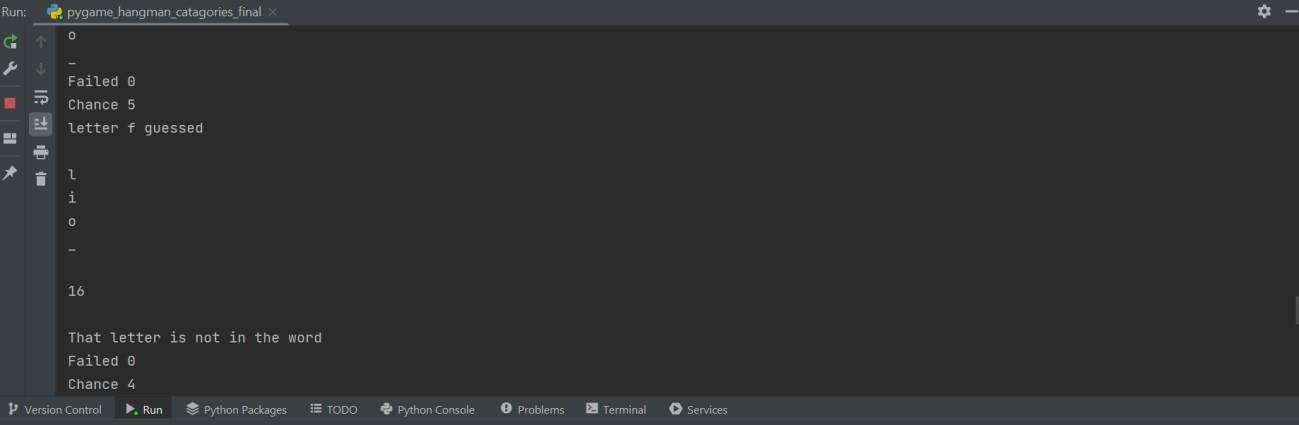


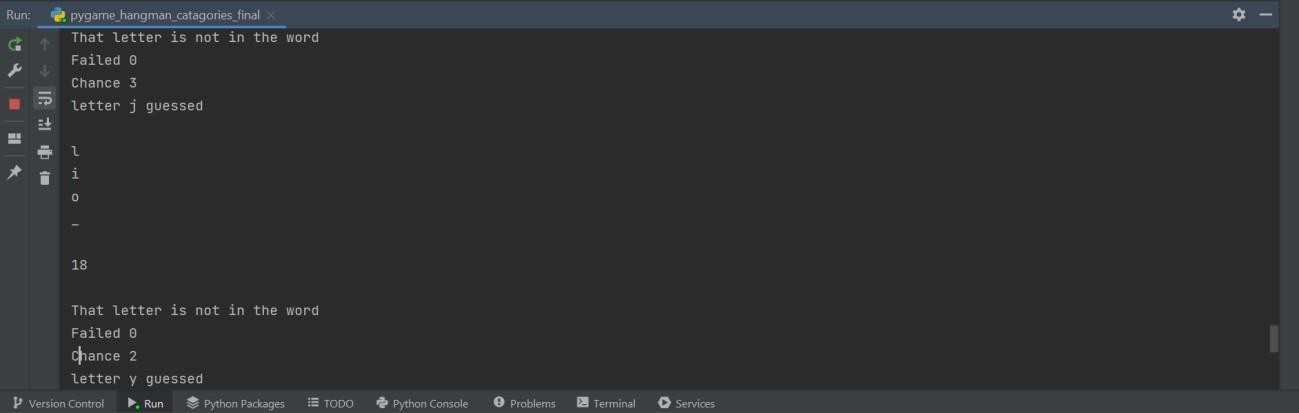
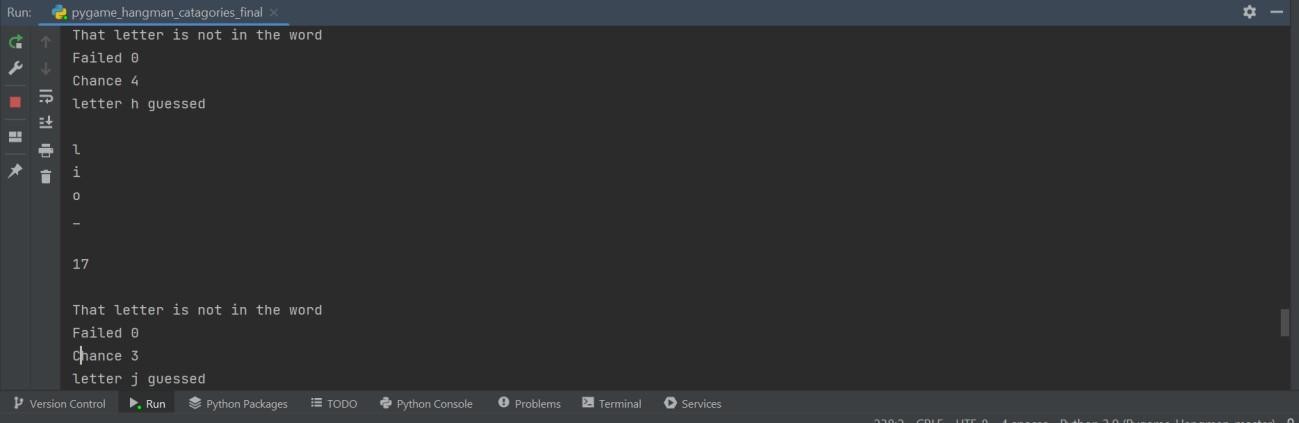


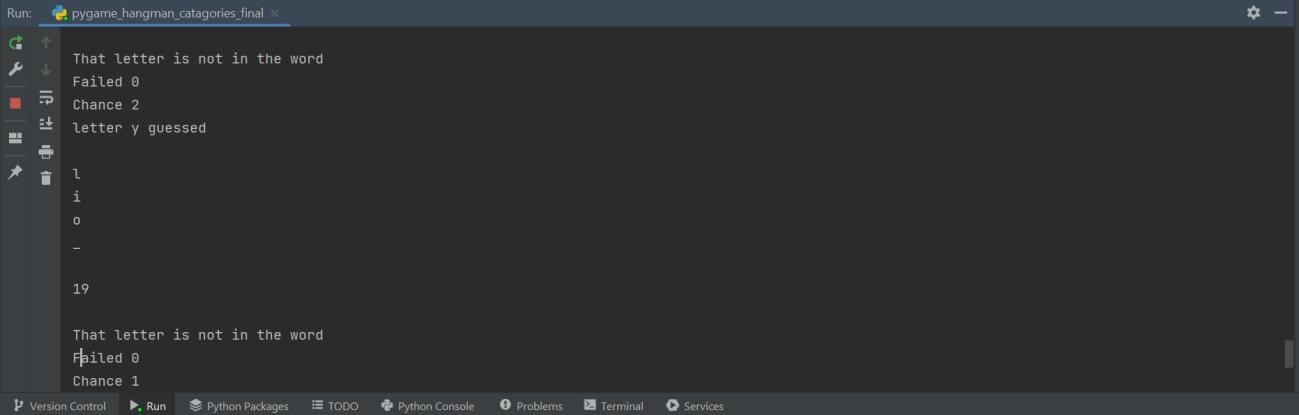


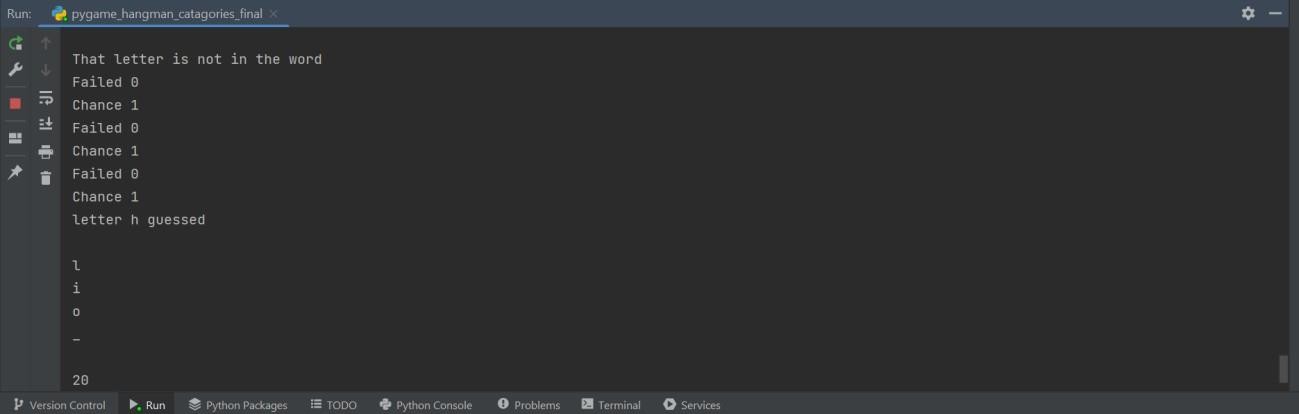


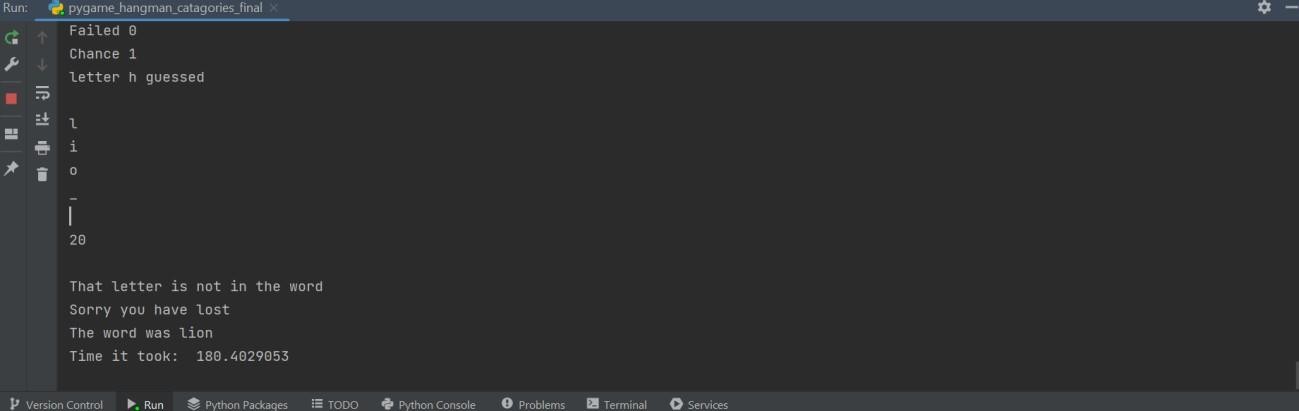






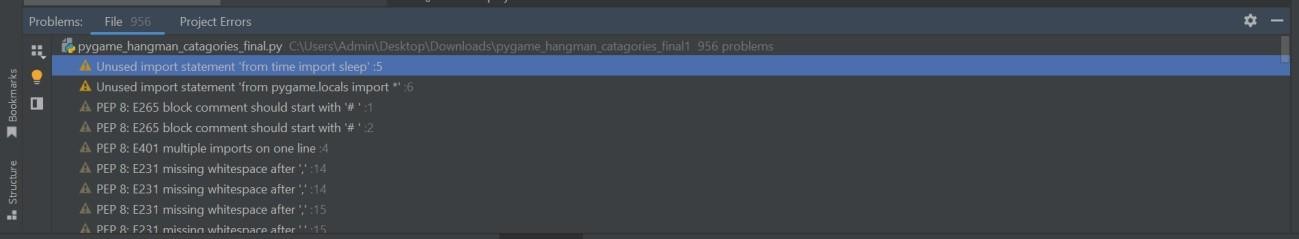




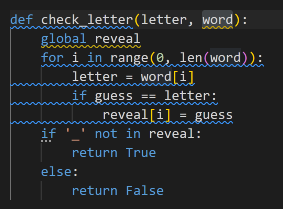


* 1. **Bugs**

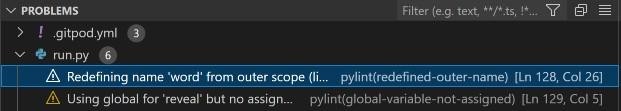
Throughout the project I encountered several bugs, some of which were solved and some not which are stated below:



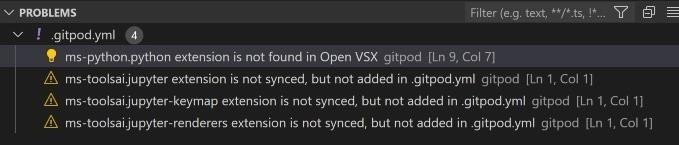
While building my diagrams, I had numerous errors stating invalid escape sequences, this waseasily resolved by adding in an extra \ or / where necessary.



I had a couple of errors with my check\_letter function and Python tutor was handy in resolvingthis. At first, I had reveal as a global but after doing some more research I found out that this was not needed in my code.



After completing my project, I was left with a few problems in the terminal. Gitpod pre-installssome bits of code which aren't necessary for certain projects. As this code causes no problemsto my work and is not needed, I have left these warnings as they are.



* 1. **User Documentation**

Allow user to guess a letter

The program should accept a single character from the user (followed by ENTER or RETURN). If multiple characters are provided then the first should be accepted and all others up to the ENTER/RETURN should be thrown away**.**

* + - If the user specifies an upper case letter, it should be converted into lower case.
    - If the user specifies an upper or lower case letter that is still available, then this is a guess andshould be handled as specified below (see “Matching a guess”).
    - If the user specifies a letter which is not currently available then tell the user, ignore the inputand prompt the user again for another letter.
    - The following special characters (!?\*$) should also be accepted. If the user enters one of these then it is NOT a guess (i.e. do not record it in the list of guesses). Instead handle the character in the following way:

! Exit the program now.

? Cheat. Display the current secret word to the user.

# Chapter 06:

**CONCLUSIONS AND PROPOSALS FOR THE FUTURE WORK**

* 1. **Conclusions**

In the conclusion of this project, Hangman is a traditional game, typically played withwords. It’s possible, however, to play Category Hangman rather than guessing words the player might guess names of cities, or athletes, or fictional characters, or Duke professors, or top forty song titles the list is endless. You’ll be writing a program to play a “guess a word letter-by-letter” version of hangman as shown above. You’ll alsobe doing some statistical analysis of the words used in the Hangman game.

Throughout the creation of Hangman Proteus Edition, Qt Creator was used to create the code for the game that could be adapted and understood by the Proteus. The program worked and performed efficiently. The code having to be adjusted to includeFEH libraries and function on the Proteus was just one of the limitations of the software.The creation of the words was limited since the words weren’t easily recognized by theProteus and each letter had to be individually programmed. Although the program tooklonger than expected to work, the coordinates of each alphabetical letter, the Hangmanstructure, and the blank spaces for each of the letters were easily plotted on the Proteusand appeared with the visually appealing look that was intended. However, thecoordinates for each letter varied from the actual coordinates of the letters when touching the screen to choose a letter. The coordinates had to be printed off on the screen to determine their actual location on the Proteus, which caused the x and y coordinates for each word’s letters to differ. Due to the time constraint, not all of the additional features were able to be fulfilled and added to the game, causing the game to only have one difficulty level and for the option to play with Hangwoman to not beavailable. If given more time to complete the project, more words and difficulty levelswould have been added. Hangwoman would become an option instead of just playing with Hangman. Since the code was significantly long, logical operators could have been used to replace the extensive loops, while less variables could have been used with the repetition and selection structures to complete the program.

The Hangman Game In Python project requires good knowledge of Python which includes defining functions and managing for/while loops. The functions that we use here contain arguments that are defined in a global scope which can be further used inother functions to improve game quality. It can also be used to provide different steps when required to execute upon conditions by the for and while loops.

The objective of our project is to implement the hangman game using Python. It doesn’trequire any specific modules other than random and time. Python loops and functions are enough to build this game here.

* 1. **Limitations and Solutions**

Hangman is a game in which students try to guess what letters are in the word before acomplete picture of their man getting hanged appears, with one element of the man andscaffold being added for each time they choose a letter that isn’t in the word. It is popular with native speaker children, […]

Hangman is a game in which students try to guess what letters are in the word before acomplete picture of their man getting hanged appears, with one element of the man andscaffold being added for each time they choose a letter that isn’t in the word. It is popular with native speaker children, and many teachers who remember from their ownschool days try to use in their EFL classes. It has recently become even more popular due to online versions being available (there are links to EFL versions of hangman at the bottom of this article). Despite the game being fun enough that some kids carry onplaying it outside class, I generally disagree with its use in its original form with EFL classes for the reasons given below. With a little imagination put it into how it is usedand what language is used with it, however, it is possible to make this game a worthwhile addition to your toolkit of classroom games.

1. It only tests spelling/ They don’t need to know how to pronounce it

The simplest variation is to ask them to pronounce the word when they have finished spelling it, either in order to get the point or for extra points. A more complex way is to ask them to say the sound they think is in the word rather than the letters, e.g. /u:/ rather than U. The teacher can then either write in the letters that make up that part of the word (e.g. “ough” if the students say “uf” and the word is “enough”) or to write theword out in phonemic symbols. If the students are just learning the phonemic script, you could write the whole word out in phonemics at the beginning of the game and getthem to guess the spelling, or give a few phonemic symbols (e.g. just the vowel sounds)as clues if they get stuck. Alternatively, you could give the phonemic transcriptions ofall the words you are going to use in the game mixed up at the top of the board.

1. They can win without knowing the meaning

For example, they might still think that “embarrassed” means “pregnant” due to them being false friends in Spanish but still get a point because all they needed was the spelling to win. You could ask the winning team to define the word as suggested for pronunciation above. A way with more speaking is to ask the student who set the challenge to give hints, either after each guess or whenever the other students get stuck.Alternatively, the clues can be pictures, perhaps ones that are revealed segment by segment.

1. There isn’t much speaking in it

In the original game the only thing the students say is the letters of that alphabet. One way of adding language is to get them to use full sentences when guessing and giving

feedback, e.g. “We’d like to choose A” and “Well done, A is in the second and ninth spaces”. (see the Useful Classroom Language section at the bottom for more possible language). Another method is to ask them to give hints as suggested above.

1. Hanging a man is a bit gory

A slightly less depressing version is to have an island or raft that disappears with eachwrong guess (without the circling sharks if you have sensitive students). You can alsoadd parts to different things as they guess wrong, but it is best if this adds up to something that shows clearly that their go is over such as adding one line or letter to TIME UP each time until the words are complete. Another possibility is to tie it in withthe picture clue that is suggested above- each wrong guess means that another segmentof the picture is revealed and if the whole picture is revealed before the word is spelt the person or team loses. You could also use this as a way of brainstorming vocabulary,

e.g. asking them to add a shop to a street every time they guess wrong, meaning that they lose the game if they can’t think of any more to add. This could work with furniturein one room, parts of a car or bicycle, etc.

1. It doesn’t tie in with any particular language points/ It’s random

You can make it tie in with a vocabulary point by limiting the words to one type of vocabulary, e.g. personality words. You can also make it into practice of body part vocabulary by asking them to name the part of the body that should go next, and similargames are possible for the other things they can build that are mentioned above. Thereare also possible connections to grammar such as there is/ there are with some and any(“Are there any Bs in it?” “Yes, there is one B”), modals of probability and possibility(“It must have an E it”, with more plus or minus points if they used language that shows they are sure like “must”), or can for ability (“I can get this word with ten guesses” with the person who bids lowest being the one to try). You can also use it to teach spelling rules as suggested below.

1. Some students shout out random letters without thinking about it

You can make the word more challenging by making it a short one with letters like Q,give hints so that they think about the meaning, give points for each time a letter is guessed correctly and take points away for each wrong guess, or let people who guessedcorrectly keep on guessing until they make a mistake.

1. It’s unfair

Some words are much more difficult to guess- generally short words with unusual letters. You can even this up by giving more hints for such words, rank the words beforethe game so that they start out with easy ones and get more difficult, or take away the competitive aspect by making it teacher against the class.

1. It’s just isolated vocabulary

Do a whole example sentence in a similar way (guessing words to make up a sentencein the same as you would usually guess letters to make up a word), or give a gapped sentence or even a whole text as a hint to the missing word.

1. It makes spelling appear random

Give a spelling rule as a hint, or only use words that represent one spelling rule (such as magic E)

1. The students start winning or losing every time

You can rank the words before you start the game and choose ones that are easier or more difficult depending on their success last time. Alternatively, you can change the number of pieces in the hangman by adding or eliminating body parts like fingers andeyebrows. You can give them even more chances by letting them continue as long as they can name more body parts to add (belly button etc).

* 1. **Future Scope:**

This game can have varied applications in the context of word formations and puzzles.Its knowledge can be valuable to many other games like CROSSWORD PUZZLES, WHEEL OF FORTUNE,SCRABBLE.We can also have an investigation of very popular and commonly used letters in most of the words. Make a frequency distribution in graph out of it. The underlying mathematical concepts are Data Collection and Analysis,Presentation and Interpretation which can have lot of implications in language processing and study of graphs and testing conjectures.Also,we can find out that the most popular letter in the English language is "e". The letter frequency of all letters in the English language is: e t a o i n s r h l d c u m f p g w y b v k x j q z.

Throughout the creation of Hangman Proteus Edition, Qt Creator was used to create the code for the game that could be adapted and understood by the Proteus. The program worked and performed efficiently. The code having to be adjusted to includeFEH libraries and function on the Proteus was just one of the limitations of the software.The creation of the words was limited since the words weren’t easily recognized by theProteus and each letter had to be individually programmed. Although the program tooklonger than expected to work, the coordinates of each alphabetical letter, the Hangmanstructure, and the blank spaces for each of the letters were easily plotted on the Proteusand appeared with the visually appealing look that was intended. However, thecoordinates for each letter varied from the actual coordinates of the letters when touching the screen to choose a letter. The coordinates had to be printed off on the screen to determine their actual location on the Proteus, which caused the x and y coordinates for each word’s letters to differ. Due to the time constraint, not all of the additional features were able to be fulfilled and added to the game, causing the game to only have one difficulty level and for the option to play with Hangwoman to not beavailable. If given more time to complete the project, more words and difficulty levels would have been added. Hangwoman would become an option instead of just playing with Hangman. Since the code was significantly long, logical operators could have been used to replace the extensive loops, while less variables could have been used with the repetition and selection structures to complete the program.

# CHAPTER 07: BIBLIOGRAPHY

1. ["Traditional Games of England, Scotland, and Ireland (Vol. I), by Alice Bertha Gomme"](https://www.gutenberg.org/files/41727/41727-h/41727-h.htm#GameI_50)., via [Project Gutenberg](https://en.wikipedia.org/wiki/Project_Gutenberg)
2. [https://www.cse.iitb.ac.in/~anwesha/cs699/mpStage1.html#:~:text=The%20game%20is%20ove](https://www.cse.iitb.ac.in/~anwesha/cs699/mpStage1.html#%3A~%3Atext%3DThe%20game%20is%20over%20when%2Cguesses%20the%20whole%20word%20correctly) [r%20when,guesses%20the%20whole%20word%20correctly](https://www.cse.iitb.ac.in/~anwesha/cs699/mpStage1.html#%3A~%3Atext%3DThe%20game%20is%20over%20when%2Cguesses%20the%20whole%20word%20correctly)
3. https://[www.cs.usask.ca/documents/Python\_Advanced.pdf](http://www.cs.usask.ca/documents/Python_Advanced.pdf)
4. [**^** "The Philadelphia Inquirer 16 Feb 1902, page Page](https://en.wikipedia.org/wiki/Hangman_(game)#cite_ref-AWCP_2-0)[39".](https://www.newspapers.com/image/168213335) Newspapers.com. Retrieved 2023-02-12.
5. [**^**](https://en.wikipedia.org/wiki/Hangman_(game)#cite_ref-3) Kapoor, Leena; Prakashan, Goyal Brothers (2021-01-01). [Solutionsto GET Smart Book for](https://books.google.com/books?id=PnA-EAAAQBAJ&dq=hangman%2Bgame%2BThe%2Bplayer%2Bguessing%2Bthe%2Bword%2Bmay%2C%2Bat%2Bany%2Btime%2C%2Battempt%2Bto%2Bguess%2Bthe%2Bwhole%2Bword&pg=PA46) [Class 4](https://books.google.com/books?id=PnA-EAAAQBAJ&dq=hangman%2Bgame%2BThe%2Bplayer%2Bguessing%2Bthe%2Bword%2Bmay%2C%2Bat%2Bany%2Btime%2C%2Battempt%2Bto%2Bguess%2Bthe%2Bwhole%2Bword&pg=PA46). Goyal Brothers Prakashan.
6. [**^**](https://en.wikipedia.org/wiki/Hangman_(game)#cite_ref-4) Sahoo, Reeta Sahoo, Gagan. [Computer Science with Python.](https://books.google.com/books?id=FOVBDAAAQBAJ&dq=hangman%2Bgame%2Bguesser%2Bcan%2Balso%2Bwin%2Bby%2Bguessing%2Ball%2Bthe%2Bletters%2Bthat%2Bappear%2Bin%2Bthe%2Bword&pg=RA1-PA13) NewSaraswati House India Pvt Ltd. [ISBN](https://en.wikipedia.org/wiki/ISBN_(identifier)) 978-93-5199-980-5.
7. [**^**](https://en.wikipedia.org/wiki/Hangman_(game)#cite_ref-5)Foreign English Teacher Under Attack For Playing Hangman Game] "Foreign English Teacher Under Attack for Playing HangmanGame | Japan Probe". Archived from [the original](http://www.japanprobe.com/2010/07/23/foreign-english-teacher-under-attack-for-playing-hangman-game/) on 2010-12-25.Retrieved 2011-03-11., Japan Probe, July 23, 2010
8. [**^**](https://en.wikipedia.org/wiki/Hangman_(game)#cite_ref-6) McLoone, Jon (13 August 2010). ["25 Best Hangman Words"](https://blog.wolfram.com/2010/08/13/25-best-hangman-words/). [Wolfram Research.](https://en.wikipedia.org/wiki/Wolfram_Research) Archived from the original on 15 August 2010. Retrieved 14 March 2015.
9. [**^**](https://en.wikipedia.org/wiki/Hangman_(game)#cite_ref-7) Griffin, Merv & Bender, David (2007). Merv: Making the Good LifeLast. New York: Simon and Schuster. pp. 99–100. [ISBN](https://en.wikipedia.org/wiki/ISBN_(identifier)) 978-0-7434- 5696-8.