A MINI PROJECT REPORT ON

**“Guess Number: Randomly generate a number unknown to the user. The user needs to guess what that number is. If the user’s guess is wrong, the program should return some sort of indication as to how wrong (e.g. the number is too high or too low). If the user guesses correctly, a positive indication should appear. Write functions to check if the user input is an actual number, to see the difference between the inputted number and the randomly generated numbers, and to then compare the numbers.”**

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

OF

**FIRST YEAR OF ENGINEERING.**

SUBMITTED BY

**( DIVISION - R , BATCH - 3 )**

UNDER THE GUIDENCE OF

**Prof. A.A. Jamgaonkar**



#### FIRST YEAR ENGINEERING

#### P.E.S MODERN COLLEGE OF ENGINEERING PUNE - 411005.

### SAVITRIBAI PHULE PUNE UNIVERSITY

#### [2022 - 23]



Progressive Education Society’s **Modern College of Engineering**  First Engineering Shivajinagar, Pune - 411005.

**CERTIFICATE**

This is to certify that the following students of First Year Engineering of PES’s, Modern College of Engineering have successfully completed their mini project and design of project entitled “Title” under the guidance of the course instructor name.

The Group Members are:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 2547 | PORJE VEDANT BHASKAR  (Team Leader) | 14 | 2560 | SUTAR PRANIT ADHIKRAO |
| 2 | 2548 | PUNJABI VEERSINGH PANKAJ | 15 | 2561 | TAYADE SIDDHESHWAR HEMANT |
| 3 | 2549 | RAJURKAR AMEYA SHAILESH | 16 | 2562 | THAKARE HIMANSHU HARISH |
| 4 | 2550 | RATHOD SHIKHAR VIJAY | 17 | 2563 | THORAT SAMRUDDHI SOMNATH |
| 5 | 2551 | SABLE MANASVI MAHENDRA | 18 | 2564 | UNHALE PRUTHU RAM |
| 6 | 2552 | SALUNKE VEDANT SAYAJI | 19 | 2565 | VETAL JAY SHANTARAM |
| 7 | 2553 | SATAV SHREYAS SUBHASH | 20 | 2566 | WALVE PRANJALI DADASAHEB |
| 8 | 2554 | SHAH HARSH JAYESH | 21 | 2567 | WARUDKAR ABHIJEET GOPAL |
| 9 | 2555 | SHAIKH SAHEEL SHAJAHAN | 22 | 2568 | YELE SAMARTH AMOL |
| 10 | 2556 | SHINDE NIKHIL ANAND | 23 | 2569 | PAWAR ANURAG JITENDRA |
| 11 | 2557 | SHINDE VIRAJ VIKAS | 24 | 2570 | SHARMA ANSHU |
| 12 | 2558 | SINGH YUVRAJ HOSHIYAR | 25 | 2571 | SHARMA MEGHA |
| 13 | 2559 | SONKUSARE MOHIT ANOJ |  |  |  |

This is in partial fulfillment of the award of the degree First Year of Engineering of Savitribai Phule Pune University.

Date: 1/12/2023

**Prof. A.A. Jamgaonkar Dr. Mrs. Rajopadhye Prof. Dr. V.U. Edlabadkar**

Course Instructor H.O.D. F.E. Co-ordinator

. Dept. of First Year Engineering

**Acknowledgement**

It gives us pleasure in presenting the partial project report on **‘Guess Number’**.

Firstly, we would like to express our indebtedness appreciation to our course instructor name. Her constant guidance and advice played very important role in making the execution of the re- port. She always gave us her suggestions that were crucial in making this report as flawless as possible.

We would like to express our gratitude towards **Dr. Mrs. Rajopadhye** Head of First Engineering Department, PES Modern College of Engineering and **Prof. Dr. V.U. Edlabadkar** F.E. Co-ordinator, PES Modern College of Engineering for kind co-operation and encouragement which helped us during the completion of this report.

Also we wish to thank our Principal, **Prof. Dr. Mrs. K. R. Joshi** and all faculty members for their whole hearted co-operation for completion of this report. We also thank our laboratory assistants for their valuable help in laboratory.

Last but not the least, the backbone of our success and confidence lies solely on blessings of dear parents and lovely friends.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 2547 | PORJE VEDANT BHASKAR  (Team Leader) | 14 | 2560 | SUTAR PRANIT ADHIKRAO |
| 2 | 2548 | PUNJABI VEERSINGH PANKAJ | 15 | 2561 | TAYADE SIDDHESHWAR HEMANT |
| 3 | 2549 | RAJURKAR AMEYA SHAILESH | 16 | 2562 | THAKARE HIMANSHU HARISH |
| 4 | 2550 | RATHOD SHIKHAR VIJAY | 17 | 2563 | THORAT SAMRUDDHI SOMNATH |
| 5 | 2551 | SABLE MANASVI MAHENDRA | 18 | 2564 | UNHALE PRUTHU RAM |
| 6 | 2552 | SALUNKE VEDANT SAYAJI | 19 | 2565 | VETAL JAY SHANTARAM |
| 7 | 2553 | SATAV SHREYAS SUBHASH | 20 | 2566 | WALVE PRANJALI DADASAHEB |
| 8 | 2554 | SHAH HARSH JAYESH | 21 | 2567 | WARUDKAR ABHIJEET GOPAL |
| 9 | 2555 | SHAIKH SAHEEL SHAJAHAN | 22 | 2568 | YELE SAMARTH AMOL |
| 10 | 2556 | SHINDE NIKHIL ANAND | 23 | 2569 | PAWAR ANURAG JITENDRA |
| 11 | 2557 | SHINDE VIRAJ VIKAS | 24 | 2570 | SHARMA ANSHU |
| 12 | 2558 | SINGH YUVRAJ HOSHIYAR | 25 | 2571 | SHARMA MEGHA |
| 13 | 2559 | SONKUSARE MOHIT ANOJ |  |  |  |

**Contents**

|  |  |
| --- | --- |
| [**Abstract**](#_3znysh7) |  |
|  |  |
| [**1**](#_2et92p0) **Problem Statement** | **1** |
|  |  |
|  |  |
| [**2**](#_2s8eyo1) **Scope** | **2** |
|  |  |
| [**3**](#_lnxbz9) **Need** | **3** |
|  |  |
|  |  |
| [**4 Software and Hardware Requirement**](#_2jxsxqh) | **4** |
|  |  |
| [**5 Introduction**](#_2jxsxqh)  **6 Outcome**  **7. Conclusion**  **8. Reference** | **5** |
|  |  |

**Abstract**

The "Guess Number" project is an interactive and creatively designed guessing game where players attempt to identify a randomly generated number within a range of 1-100. The game features engaging elements such as ASCII art for visual appeal and text-to-speech functionality for auditory feedback. Users can choose from Easy, Medium, or Difficult difficulty levels, each offering a different number of chances. The project emphasizes user experience with feedback like "Too High" or "Too Low." Potential future enhancements include implementing a scoring system, graphical user interface elements, and additional features for an even more captivating user experience. References include the use of Patorjk's Text to ASCII Art Generator for ASCII art, pyttsx3 library for text-to-speech, and the random module in Python for number generation.

**1.**

**Introduction**

##### Guess Number

* 1. **Introduction**

The "Guess Number" project introduces an engaging and interactive game designed to challenge users' ability to predict a randomly generated number within a defined range (1-100). Rooted in the spirit of classic guessing games, this project aims to provide users with a fun and entertaining experience, incorporating elements such as ASCII art, text-to-speech functionality, and a dynamic difficulty level mechanism.

Project Overview:

Guessing games have long been a source of entertainment, combining chance and strategy to captivate players. In the "Guess Number" project, we leverage the versatility of Python programming to create a modern and interactive version of this timeless concept. The game not only challenges users to make accurate guesses but also enhances the experience through creative visual and auditory elements.

Key Objectives:

* Random Number Generation: The project utilizes Python's random module to generate unpredictable numbers, ensuring each game session is unique.
* User Input Validation: Input validation mechanisms guarantee that users provide legitimate guesses within the specified range, contributing to the robustness of the program.
* Dynamic Feedback Mechanism: Users receive real-time feedback on the correctness of their guesses, guiding them with indications such as "Too High," "Too Low," "High," or "Low."
* Creative Elements: ASCII art is employed to create an eye-catching introduction, and text-to-speech functionality adds an auditory dimension to the user experience.

Significance of the Project:

Beyond its entertaining nature, the "Guess Number" project serves as a platform to explore various programming concepts, including random number generation, input validation, and the integration of creative elements. The combination of technical implementation and user engagement makes this project both educational and enjoyable.

Structure of the Report:

This project report is structured to provide a comprehensive overview of the "Guess Number" game, detailing its implementation, creative features, testing, and potential future improvements. Each section contributes to a thorough understanding of the project's design and execution.

Project Rationale:

The allure of guessing games lies in their simplicity and the universal appeal of the challenge they present. By developing the "Guess Number" game, we aim to deliver an immersive and enjoyable experience that not only entertains users but also provides a practical application of programming concepts. The project serves as a conduit for users to engage with concepts such as random number generation, input validation, and creative integration within a user-friendly and visually appealing interface.

Motivation:

The motivation behind this project is rooted in fostering a hands-on, interactive learning experience. Traditional programming concepts often come to life through real-world applications, and the "Guess Number" game serves as a prime example. Through the exploration of random number generation and user input validation, users can grasp these concepts in a practical context while enjoying a game that challenges their problem-solving skills.

User-Centric Design:

Understanding the importance of user experience, the project incorporates creative elements to enhance engagement. ASCII art is strategically used to create a visually striking introduction, setting the tone for an interactive and visually stimulating experience. Furthermore, the integration of text-to-speech functionality adds an auditory layer, making the game more inclusive and accessible.

Educational Value:

Beyond its entertaining nature, the "Guess Number" project is designed with educational objectives in mind. It provides a structured environment for users to apply their programming knowledge, experiment with Python's capabilities, and gain insights into the logic behind random number generation algorithms. The project's simplicity ensures that even users with basic programming skills can actively participate and derive educational benefits.

Scope of the Report:

This project report unfolds to explore various facets of the "Guess Number" game, from its creative aspects and technical implementation to testing procedures and potential future enhancements. Each section is crafted to contribute to a holistic understanding of the project's design, execution, and educational significance.

As we embark on this exploration, we delve into the intricacies of creative game design, discuss the technical architecture behind the random number generation, and outline the steps taken to validate user inputs. Additionally, we examine the testing methodologies employed to ensure the robustness of the game and conclude with reflections on potential future improvements and expansions.

PESMCOE, Department of Computer Engineering 2022 - 23

**2.**

**Scope**

##### Guess Number

**2.1 Scope**

1. Project Objectives:

* Develop an interactive and entertaining "Guess Number" game.
* Challenge users to guess a randomly generated number within the range of 1 to 100.
* Integrate creative elements, including ASCII art and text-to-speech functionality.
* Implement a dynamic difficulty level mechanism to tailor the user experience.

1. Deliverables:

* Fully functional "Guess Number" game software.
* Project documentation, including this project report.
* Source code repository containing the game code.

1. Features and Functionalities:

* Random number generation within the specified range.
* User input validation to ensure guesses are within the range.
* Dynamic feedback on user guesses (e.g., "Too High," "Too Low," "High," "Low").
* ASCII art for visually appealing introduction. • Text-to-speech functionality for auditory feedback.
* Difficulty level options: Easy (10 chances), Medium (7 chances), Difficult (5 chances).

1. Tasks and Activities:

* Design and implement the random number generation algorithm.
* Develop user input validation mechanisms.
* Create ASCII art for the game introduction.
* Integrate text-to-speech functionality.
* Implement dynamic feedback based on user guesses.
* Design and implement the difficulty level mechanism.
* Conduct testing to ensure the robustness of the game.

1. Constraints:

* Limited development time: The project must be completed within a specified timeframe.
* Resource constraints: Development resources, including personnel and software tools, are limited.

1. Assumptions:

* Users have basic knowledge of Python programming.
* The target audience is open to a text-based game without a graphical user interface.

1. Exclusions:

* Graphical user interface (GUI): The game will be text-based without a graphical interface.
* Scoring system: The initial version will focus on guessing mechanics without a scoring system.

1. Acceptance Criteria:

* The game successfully generates random numbers within the specified range.
* User inputs are validated, ensuring they fall within the range.
* Dynamic feedback provides accurate indications based on user guesses.
* ASCII art and text-to-speech elements enhance the user experience.
* The game operates smoothly under different difficulty levels.
* The project documentation is comprehensive and clear.

PESMCOE, Department of Computer Engineering 2022 - 23

**3.**

**Need**

##### Guess Number

1. Engaging Learning Experience:

* Need: To provide an engaging and interactive learning experience for individuals interested in Python programming fundamentals.
* Rationale: Learning programming concepts is often more effective when applied in a practical and enjoyable context. The game format offers users a dynamic platform to practice their programming skills while having fun.

1. Application of Programming Concepts:

* Need: To offer a practical application for key programming concepts, including random number generation, user input validation, and conditional statements.
* Rationale: The project serves as a hands-on opportunity for users to implement and reinforce fundamental programming principles, bridging the gap between theoretical knowledge and practical application.

1. Creativity and Aesthetics:

* Need: To integrate creative elements such as ASCII art and text-to-speech functionality, enhancing the visual and auditory appeal of the game.
* Rationale: Incorporating creative elements not only adds visual interest but also exposes users to the expressive and artistic side of programming, expanding their understanding of what can be achieved.

1. User Engagement and Enjoyment:

* Need: To create an entertaining and challenging game that captures the user's attention and encourages repeated interaction.
* Rationale: User engagement is essential for effective learning. The game's interactive nature, coupled with dynamic feedback and adjustable difficulty levels, ensures that users find the experience enjoyable and rewarding.

1. Practical Problem-Solving:

* Need: To provide a platform for users to engage in practical problem-solving by interpreting feedback and iteratively refining their guessing strategy.
* Rationale: The game simulates real-world problem-solving scenarios where users must analyze feedback, adjust their approach, and enhance their critical thinking skills—an invaluable skill applicable across various domains.

1. Inclusive and Accessible Design:

* Need: To incorporate features like text-to-speech for inclusive and accessible design, catering to users with diverse preferences and abilities.
* Rationale: Ensuring inclusivity in design promotes accessibility and usability, making the game enjoyable and accessible to a broader audience.



PESMCOE, Department of Computer Engineering 2022 - 23

**4.**

**Software and Hardware Specifications**

##### Guess Number

* 1. **Software Specifications**
     1. Python:
* Version: Python 3.x
* The project is developed using the Python programming language.
  + 1. Operating System:
* The project is designed to run on various operating systems that support Python, including Windows, macOS, and Linux.
  + 1. Pyttsx3 Library:
    - Version: Latest stable version
    - Pyttsx3 is used for text-to-speech functionality.
    - Type following line in the Python Terminal to install the library.
* **pip install pyttsx3**
  1. **Hardware Specifications**

1. Processor:

* Any modern processor capable of running Python 3.x.

1. RAM:

* Minimal requirements for running Python and the game logic. A standard amount of RAM for basic programming tasks.

1. Storage:

* Minimal storage requirements. The project is text-based and doesn't involve large datasets or resource-intensive files.

1. Input Device:

* Standard keyboard for user interaction with the game.

1. Output Device:

* A display device (monitor or screen) for displaying the game interface.

1. Audio Output Device:

* Speakers or headphones for the text-to-speech functionality.



PESMCOE, Department of Computer Engineering 2022 - 23

**5.**

**Conclusion**

##### title

## Conclusion:

In the culmination of the "Guess Number" project, we have successfully crafted a dynamic and engaging text-based game that not only entertains users but also serves as a practical application of fundamental programming concepts. The project's journey has traversed through the realms of creativity, learning, and user interaction, resulting in a game that encapsulates the essence of both entertainment and education.

Achievements:

1. Interactive Learning Experience:

* The project has provided users with an interactive and enjoyable learning experience, allowing them to apply programming principles in a gamified context.

1. Practical Application of Concepts:

* Key programming concepts such as random number generation, user input validation, and conditional statements have found practical expression in the development of the game.

1. Creative Elements:

* The incorporation of ASCII art and text-to-speech functionality has added creative dimensions to the project, making it visually and auditorily appealing.

1. User Engagement:

* The game's adjustable difficulty levels, dynamic feedback mechanisms, and overall design contribute to a high level of user engagement, fostering a positive and enjoyable experience.

1. Inclusive Design:

* Features like text-to-speech have been integrated to make the game more inclusive, ensuring accessibility for users with diverse preferences and abilities.

Learning Opportunities:

The project has offered valuable learning opportunities, emphasizing the importance of creative design, user interaction, and the practical application of programming skills. As users engage with the game, they not only refine their programming proficiency but also enhance problem-solving abilities and critical thinking skills.

Future Possibilities:

While the current version of the "Guess Number" game stands as a robust and entertaining application, there are possibilities for future enhancements. These may include the introduction of additional game elements, integration with graphical interfaces, or the development of a scoring system to further enrich the user experience.

In conclusion, the "Guess Number" project stands as a testament to the synergy between programming and creativity. It not only showcases the technical capabilities of Python but also underscores the potential of programming to create enjoyable and educational experiences. As we reflect on the journey of this project, the door remains open for further exploration and innovation, inviting users and developers alike to delve deeper into the world of programming through engaging and interactive applications.



PESMCOE, Department of Computer Engineering 2022 – 23

**REFERENCE**

* ASCII art generated using Patorjk's Text to ASCII Art Generator.
* Text-to-speech functionality implemented with the pyttsx3 library.
* Random number generation facilitated by the random module in Python.

Code

##################################===PROJECT===############################################

"""Guess Number: Randomly generate a number unknown to the user. The user needs to

guess what that number is. If the user's guess is wrong, the program should return some

sort of indication as to how wrong (e.g. the number is too high or too low). If the user

guesses correctly, a positive indication should appear. Write functions to check if the user

input is an actual number, to see the difference between the inputted number and the

randomly generated numbers, and to then compare the numbers."""

##################################===PREREQUISITES===############################################

# install pyttsx3 for TEXT TO SPEECH

# imported random to get a random number

# use of ascii characters

##################################===CREATIVITY===############################################

# ascii characters

# text to speech

# difficulty level mechanism according to the player

# comparison between Guessed number and the Number Generated by The Program.

# (eg. Too High, Too Low, High, Low)

from random import randint

import pyttsx3

text\_speech= pyttsx3.init()

game= """

██████╗ ██╗ ██╗███████╗███████╗███████╗ ███╗ ██╗██╗ ██╗███╗ ███╗██████╗ ███████╗██████╗

██╔════╝ ██║ ██║██╔════╝██╔════╝██╔════╝ ████╗ ██║██║ ██║████╗ ████║██╔══██╗██╔════╝██╔══██╗

██║ ███╗██║ ██║█████╗ ███████╗███████╗ ██╔██╗ ██║██║ ██║██╔████╔██║██████╔╝█████╗ ██████╔╝

██║ ██║██║ ██║██╔══╝ ╚════██║╚════██║ ██║╚██╗██║██║ ██║██║╚██╔╝██║██╔══██╗██╔══╝ ██╔══██╗

╚██████╔╝╚██████╔╝███████╗███████║███████║ ██║ ╚████║╚██████╔╝██║ ╚═╝ ██║██████╔╝███████╗██║ ██║

╚═════╝ ╚═════╝ ╚══════╝╚══════╝╚══════╝ ╚═╝ ╚═══╝ ╚═════╝ ╚═╝ ╚═╝╚═════╝ ╚══════╝╚═╝ ╚═╝

"""

number= randint(1,100)

# print(number)

def level():

print(game)

welcome= "Welcome To The Game of GUESS NUMBER!"

select="Select a Difficulty Level:"

print(select)

text\_speech.say(welcome)

text\_speech.runAndWait()

level\_ = int(input("1. Easy (10 Chances)\n2. Medium (7 Chances)\n3. Difficult (5 Chances)\n"))

if level\_ == 1:

chance= 10

elif level\_ == 2:

chance= 7

elif level\_ == 3:

chance= 5

return chance

def get\_user\_guess():

while True:

try:

user\_guess = int(input("Guess a number (1-100): "))

if 1 <= user\_guess <= 100:

return user\_guess

else:

print("Please enter a number within the specified range (1-100).")

except ValueError:

print("Invalid input. Please enter a valid number.")

def play(chance):

guess= get\_user\_guess()

if number == guess:

greet= "Hurray! You Guessed it Correct. Thank-You"

print(greet)

text\_speech.say(greet)

text\_speech.runAndWait()

elif number > guess:

if ((number - guess) > 30):

print("The Number Guessed is TOO LOW!")

else:

print("The Number Guessed is LOW!")

elif guess > number:

if ((guess - number) > 30):

print("The Number Guessed is TOO HIGH!")

else:

print("The Number Guessed is HIGH!")

if number != guess:

chance -= 1

if chance > 0:

text= f"You have {chance} Chances left."

print(text)

text\_speech.say(text)

text\_speech.runAndWait()

play(chance)

else:

text= f"The CORRECT guess was {number}"

print(text)

end= "You Lose. Better Luck Next Time."

print(end)

text\_speech.say(text)

text\_speech.say(end)

text\_speech.runAndWait()

play(level())

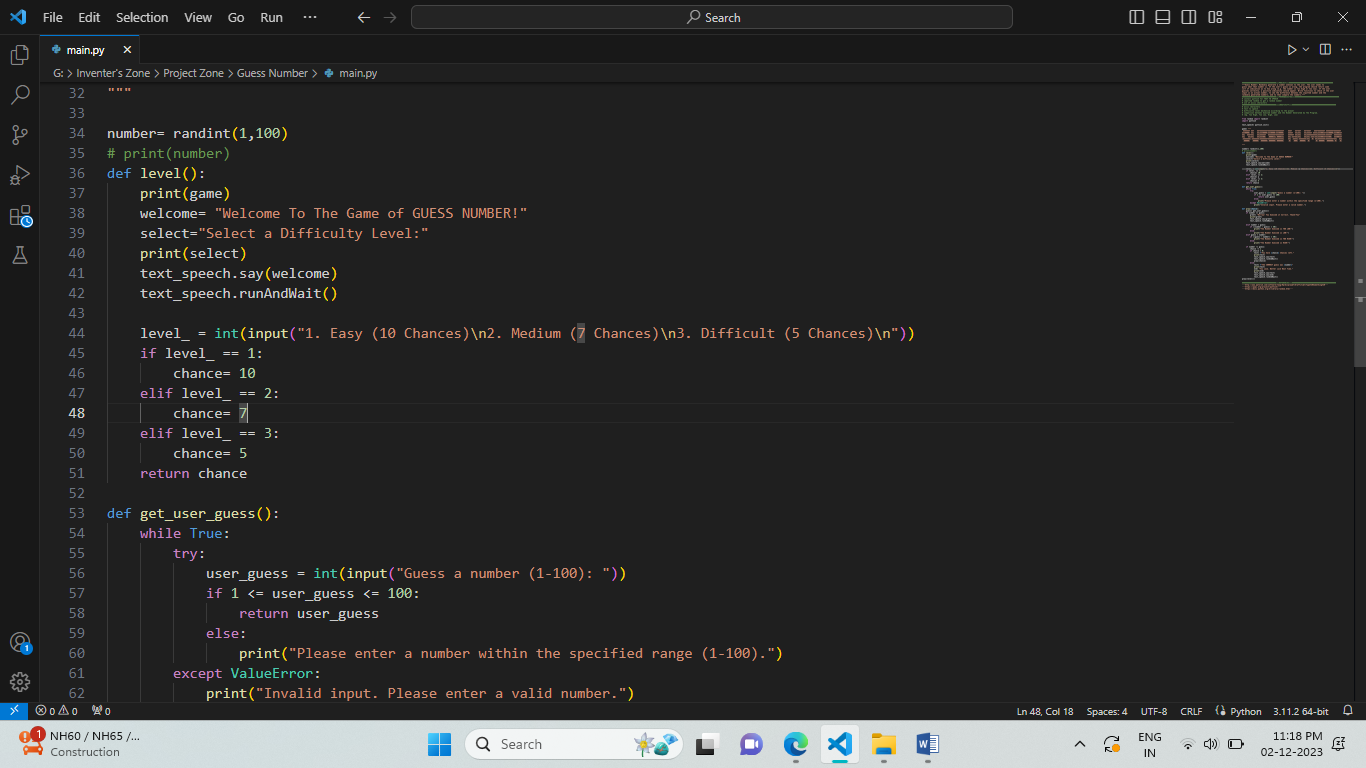
##################################===REFENENCES===############################################

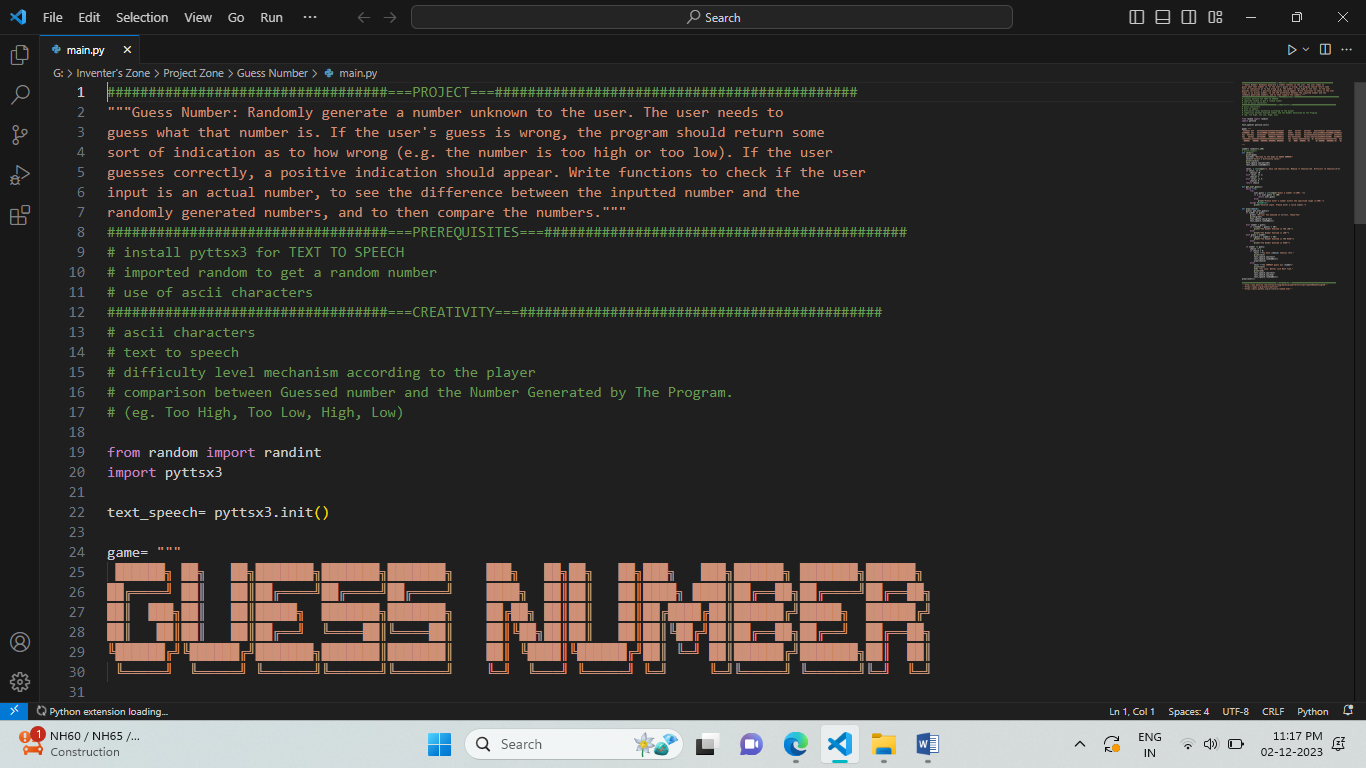
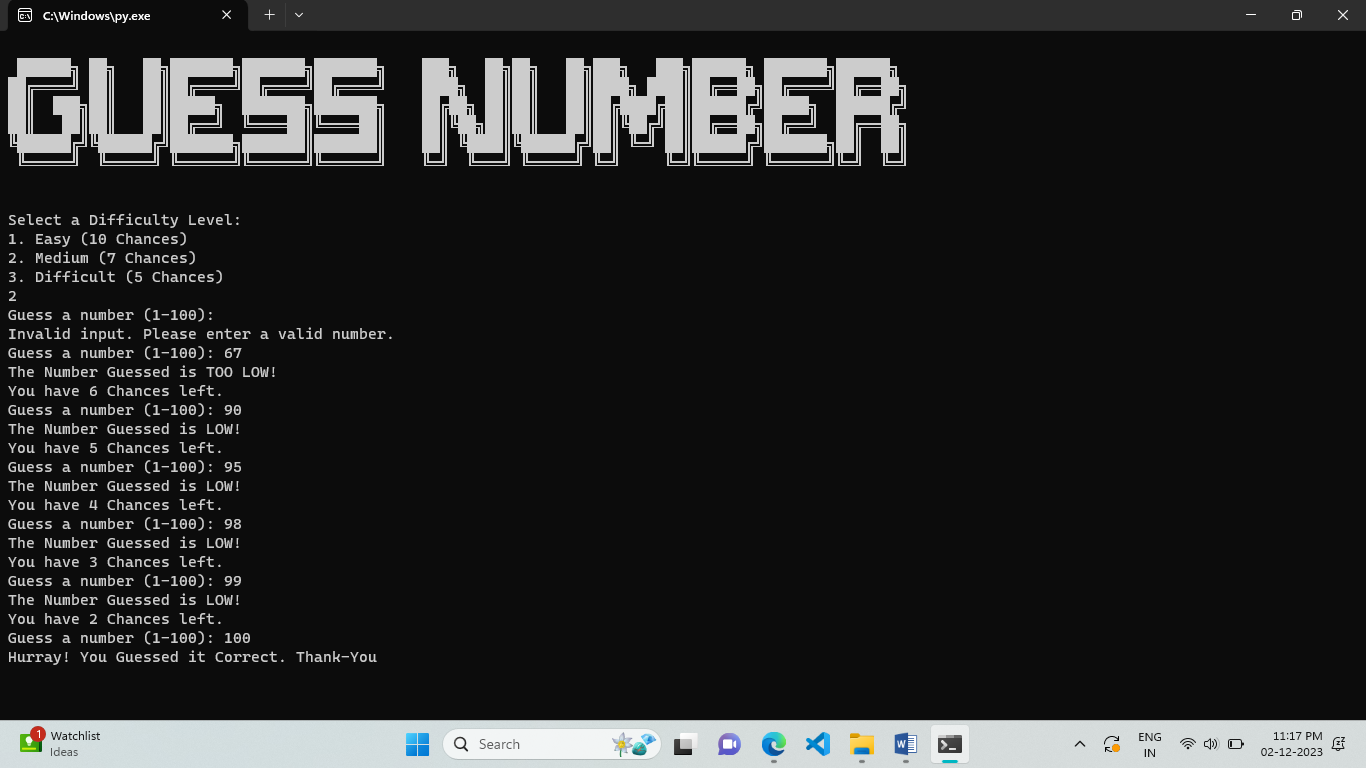
'''http://www.patorjk.com/software/taag/#p=display&f=Graffiti&t=Type%20Something%20'''

'''https://pypi.org/project/pyttsx3/'''

'''https://docs.python.org/3/library/random.html'''

Github link:



**SCREENSHOTS**

