DARK FOREST ADVENTURE GAME

Submitted by Sufyan ali awan (reg no 22318076)

Introduction:

The "Dark Forest Adventure" stands as an engaging text-based game crafted using the Python programming language. Its purpose is to submerge players into an enigmatic forest teeming with challenges, puzzles, and captivating chambers. Within this report, I aim to present a comprehensive examination of the game, delving into its structure, features, and overall functionality.

The game is built around a network of interconnected rooms, each characterized by a distinct class with its own set of features. The primary classes encompass:

Room: Serving as the fundamental class, it represents a generic room with exits connecting to other rooms.

Equipment Room:
This specialized room enables players to acquire items essential for their adventure.

Trees Chamber:
A room featuring a math puzzle challenge with rewards awaiting those who successfully solve it.

River Room:
Designed for players equipped with specific items, this room allows them to swim across a river.

Cave Room:
Presents a math puzzle and requires players to discard items for energy.

Mountain Room:
Players must possess specific equipment to conquer the challenge of climbing a mountain.

Labyrinth Room:
This room introduces a geography challenge, tasking players with finding the correct path.

Secret Chamber:
A room with multiple paths, each leading to different outcomes.

Dark Secret Room:

Players face a riddle challenge in this room, unlocking further progression.

Exit Room:
The ultimate destination where players successfully conclude their adventure.

This game will start with the name of person who want to play this game in this game there are ten rooms which interconnected with each other and there are different kind of math games and puzzle game for complexity first I added floor game but it hard to play so I removed all games difficult game with easy game.

I added an energy level drink in this game for more fun. In this game player will pick different items from different rooms and will drop as well.

The things which used in this game are.

Move:
Use directional commands like north, south, east, and west to navigate between rooms.

Look:
Examine the description of the current room by using the look or inspect command.

Inventory:
View the items collected in the player's inventory by typing inventory or items.

Quit:
End the game by using the quit or exit command.

Games:

Math and puzzle game.

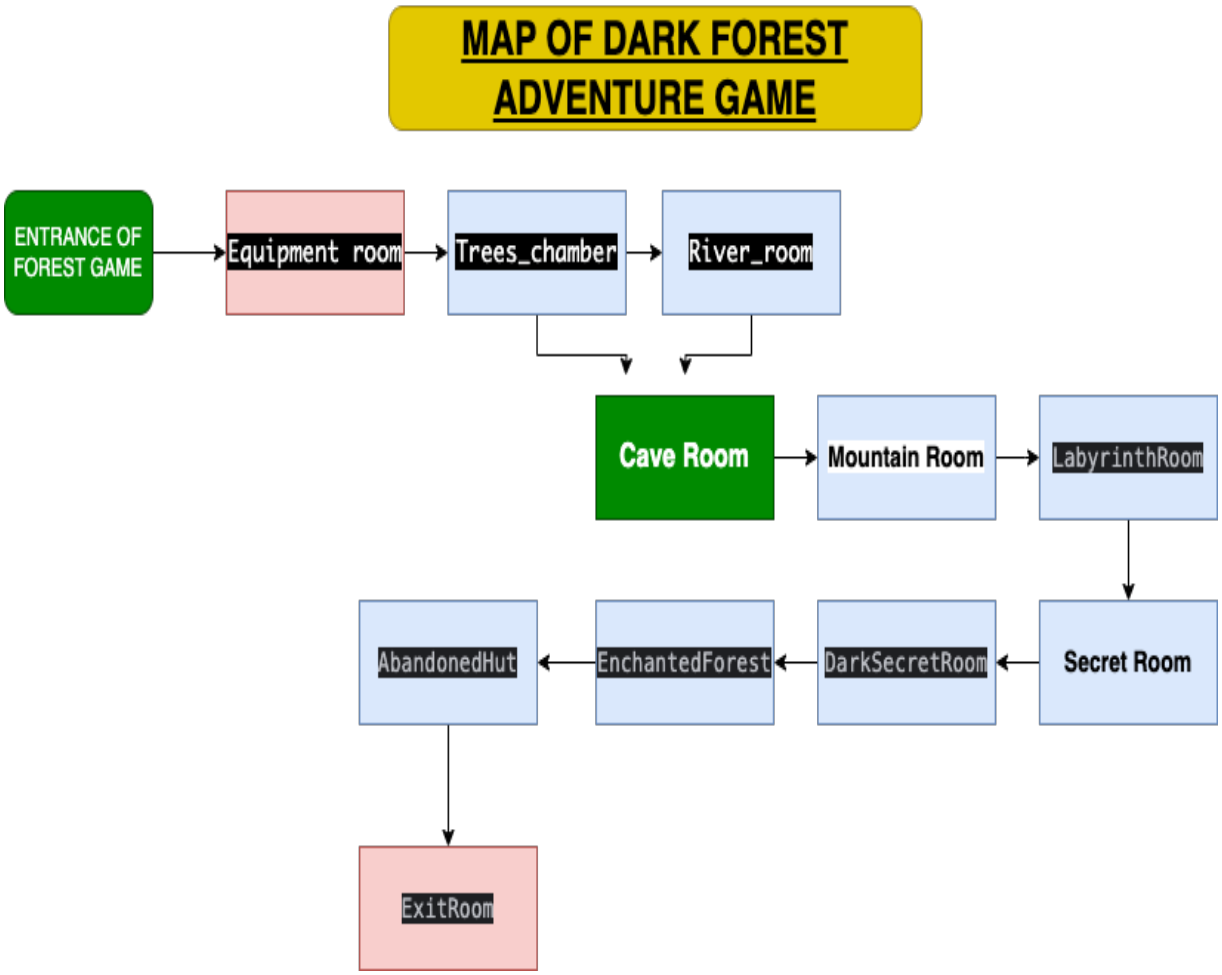
Help:
Display information about available commands or seek assistance by typing help.

Choose:
Utilized in the Labyrinth Room to pick a path.

Yes or No:
Respond to prompts, such as deciding whether to swim across the river.

Numeric Input:
Input numbers like 1, 2, 3, etc., to make selections, such as choosing an item in the Equipment Room

MAP OF DARK FOREST GAME:



PYTHON COMMANDS AND ERROR HANDLING IN THIS GAME:

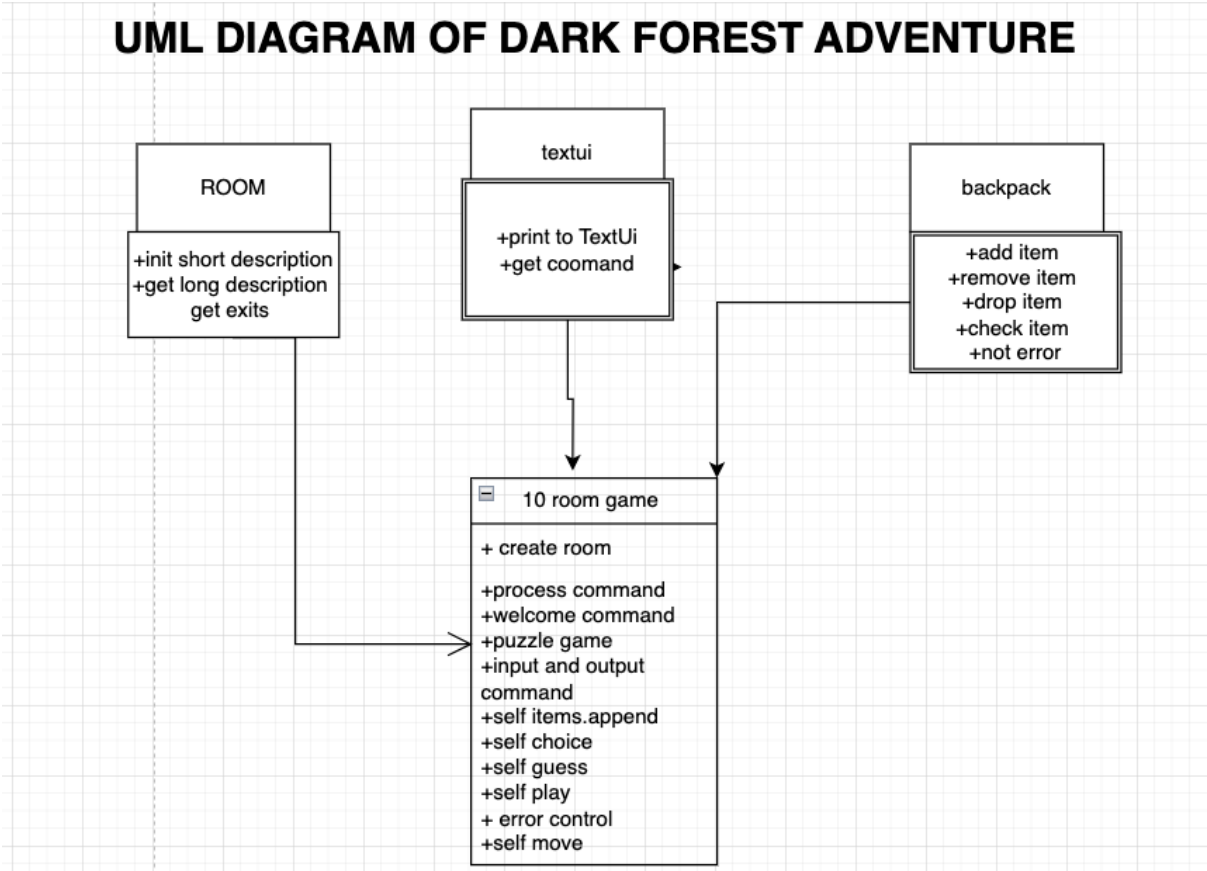
In this game I used basic game logic of python text-based game, I construct one class for complete forest in this one class there are many rooms. After base room I added subclasses like “trees chamber “etc.

After this I utilized class method and behaviour of all sub classes with the help of control full statements, loops, conditional statement, no over flow statement, self-generating ,random

module ,string formatting techniques for dynamic perfect output and error handling techniques.

n my "Dark Forest Adventure" game, I've made sure to include strong error handling to improve your overall experience and ensure everything runs smoothly. Specifically, when you're in the Equipment Room deciding which items to pick, I've set up a system to handle any potential issues if you accidentally input something that's not a number. So, if you happen to type a word instead of a number, don't worry! The game won't crash or act strangely. Instead, I will kindly let you know about the mistake and ask you to try entering a valid number again. This way, I hope to make your gaming journey more user-friendly and freer of unnecessary hiccups.

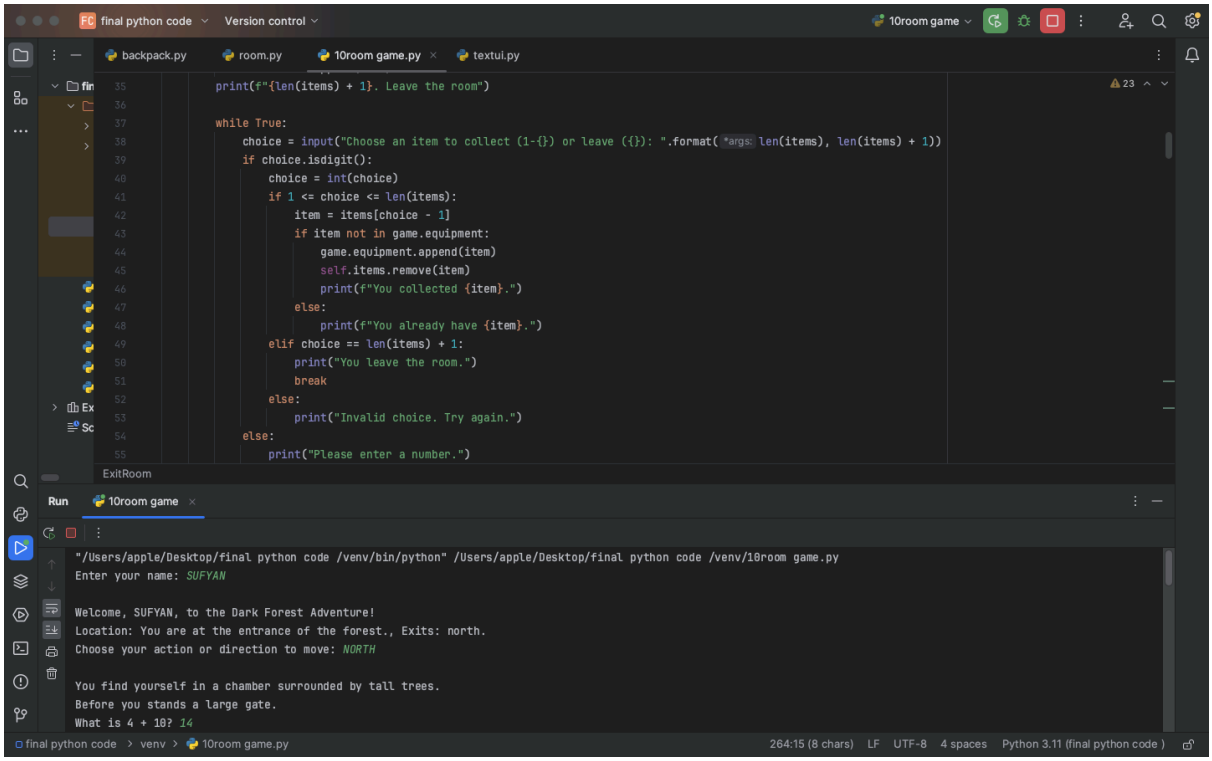
UML Diagram:



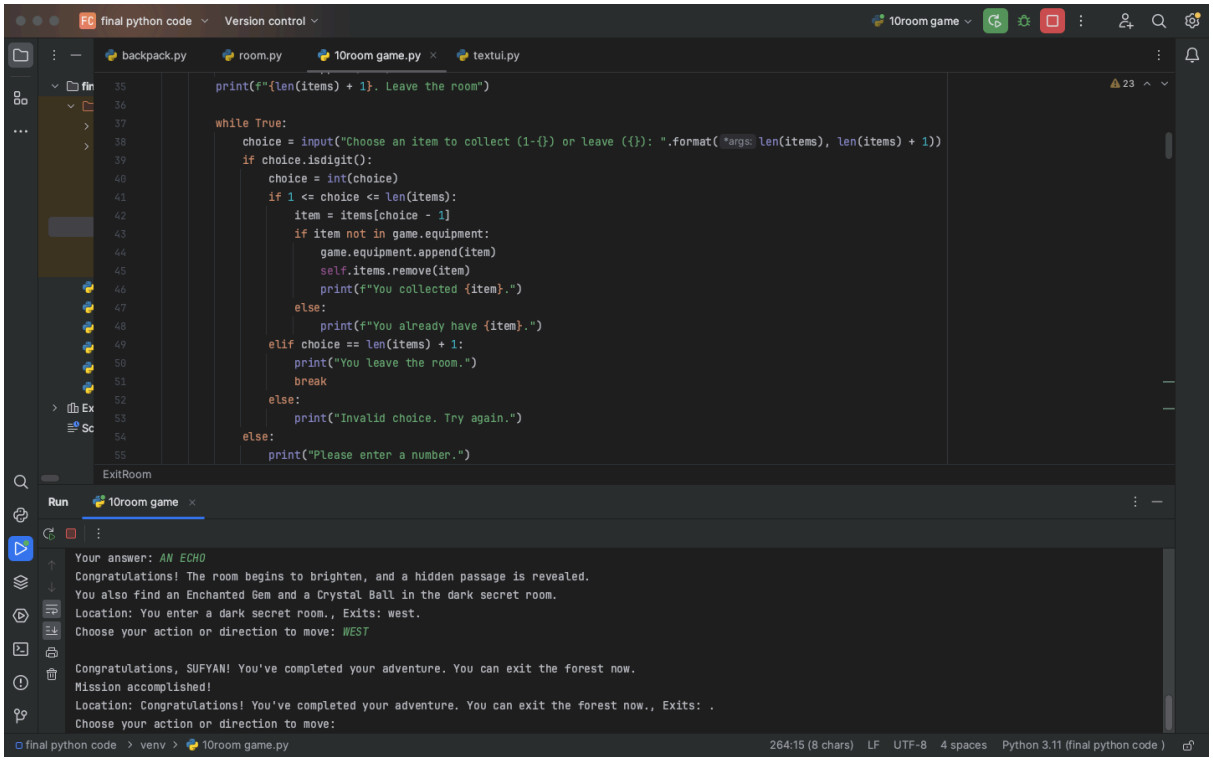
Result:

When someone enter in this game for each direction, he/she will go in different task according to their selection of direction and game result.
Starting and result of my game.

STARTING OF GAME:



END OF GAME:


final python code
Version control
10room game
backpack.py
room.py
10room game.py
textui.py
print(f"{len(items) + 1}. Leave the room")
while True:
choice = input("Choose an item to collect (1-{}) or leave ({}): ".format(*args: len(items), len(items) + 1))
if choice.isdigit():
choice = int(choice)
if 1 <= choice <= len(items):
item = items[choice - 1]
if item not in game.equipment:
game.equipment.append(item)
self.items.remove(item)
print(f"You collected {item}.")
else:
print(f"You already have {item}.")
elif choice == len(items) + 1:
print("You leave the room.")
break
else:
print("Invalid choice. Try again.")
else:
print("Please enter a number.")
ExitRoom
Run
10room game
Your answer: AN ECHO
Congratulations! The room begins to brighten, and a hidden passage is revealed.
You also find an Enchanted Gem and a Crystal Ball in the dark secret room.
Location: You enter a dark secret room., Exits: west.
Choose your action or direction to move: WEST
Congratulations, SUFYAN! You've completed your adventure. You can exit the forest now.
Mission accomplished!
Location: Congratulations! You've completed your adventure. You can exit the forest now., Exits: .
Choose your action or direction to move:
final python code
venv
10room game.py
264:15 (8 chars)
LF
UTF-8
4 spaces
Python 3.11 (final python code)