KillDoctorLuckyGameDesign README
Author: Zack-Haowen Xue
Project Github: https://github.com/xue59/KillDoctorLuckyGameDesign/tree/Milestone3

Milestone3_GamePlay

1. **Instruction on how to run:**
   1.1 Once you clone or download the project Repo, unzip the Repo.
   1.2 Open the terminal under the folder of /Milestone3_GamePlay/res
   1.3 Find the jar file called ms3_GamePlay.jar.
   1.4 In the terminal under the res folder, run the command: java -jar ms2_Controller.jar [map txt file directory] [totalPlayersNumber] [totalMaxTurnNumber]
   1.5: Please provide the map txt file, total player and max turn exact as above pattern, for example: java -jar ms3_GamePlay.jar mansion2023Pet.txt 2 99 this would start with 2 players and 99 turns.
   1.6: Once the game starts, you will see the Welcome Screen, and you can execute the following commands: 0-create map graph, 1-setup all players, 2-Find info on a specific room, 3-Find info on a specific player, 4-Start the game turns, 66-quit the program.
   1.7: Once in the game, follow the instructions to take turns and play the game.
   1.8: Currently, the game will over when the maximum number of turns is reached or target, Dr. Lucky, got completely killed. Once the game is over, you can use 'm' or 'M' to return to the main menu.
   1.9: To exit the game program, in main menu execute 66 to kill the program. Enjoy the game!

2. **Example map txt file:**
   a. Example Map Files .txt can be found in the /res/ folder. There are 3 sample .txt map files you can choose to start and run the Jar file:
   b. mansion2023Pet.txt - the standard Milestone 1 provided Dr. Lucky's Mansion file with Pet added and modified item damage, a few item damage are enhanced to make it easy on killing Dr. Lucky.
   c. customerWorld.txt - slightly changed room names based on Dr. Lucky's Mansion.
   d. myOwnWorldMapV1.txt - my designed map.txt file.
   e. After each run, check the generated pictures in the same /res folder.

3. **Example run with explanations:**

Ms3ExampleRunHumanWin.txt

1. This example run logged that the Human players win the game by successfully killing Dr Lucky and reduced target HP to 0.
2. Line 16 – 35 shows two Human players are added into the game called p1 and p2.
3. Line 66: the game officially started.
4. Line 76: turn #1, the p1 player moved the pet to Drawing Room. (Pet move success confirmed in line 182)
5. Line 93: turn #2, the p2 picked Letter Opener(Damage=77).

6. Line 182: turn #5, the Pet **effected the visibility** of a Look around command carried out by p1. The P1 was trying to look around for room Dining Hall, but the Pet is the Drawing Room, thus the outsit players (p1) cannot see inside conditions. Thus, players and items in Drawing Room in this turn cannot be seen.
7. Line 207-209: turn #6, the p2 player encountered Dr. Lucky, target, in the same room in room #5 Foyer and the p2 player successfully Poking attacked the target. It confirmed the attack and Dr. Lucky recued HP by 1.
8. Line 348: turn #14, p2 is in room Parlor with Pet, but Dr. Lucky not in the same room, thus cannot attack.
9. Line 443: turn #18, player p2 is in room #17 Tennessee Room with Dr. Lucky, (cannot be seen by others, since no players around room) and player p2 successfully attacked the Dr. Lucky with Letter Opener(Damage=77). This caused damage and 0 HP for the target. Thus game is over and p2 win.(confirmed by the system message line 452 & 451)
10. Line 451-452: Congratulation!!!!!!!!! Game Winner: p2
11. Line 468: user confirmed game result again: Last game already finished, back to main menu to re-start a New Game. Last game winner: p2!!!
12. Line 482: Line 235: quit game successfully with order 66. :)

Ms3ExampleRunComputerWin.txt

1. This example run logged that the Computer players win the game by successfully killing Dr Lucky and reduced target HP to 0.
2. Line 16 – 35 shows two Computer players are added into the game called com1 and com2.
3. Line 61: the game officially started.
4. Line 97: turn #1, Player(com1) attacked Dr.Lucky SUCCESS with item: Poking(Damage=1) and causing damage of hp reduced by 1. It is confirmed that the targeted hp reduced to 49 in line 99. This shows that the computer attack logic is working as expected, as it will always trying to attack to Dr. Lucky whenever it’s possible. Even the computer player does not have items, it will use the basic attack, poking, to attack. If they have items, they will always try to use the highest damage item to attack. (This can be shown in turn #9, line 203).
5. Line 125: turn #3, **Computer player**: com1 PICK up Revolver with 99 damage. This shows the computer pick logic is working correctly, since it will always trying to pick item when they cannot attack and also having capacity. It also will try to pick the highest damage item in the room floor.
6. Line 138: turn #4, **Computer player**: com2 MOVE to room: Trophy Room. This shows the computer move logic is correct. When the computer player cannot attack or pick it will try to move randomly in the game world. In this case the com2 player moved to Trophy Room.
7. Line 145: turn #5, Current Room: Armory(**Pet**(Fortune Cat Pet) in this room.) The computer player com1 encountered the Pet, the Pet is taking effect, but there is no Dr. Lucky in the room thus can not attack.
8. Line 203: turn#9, **Computer player**: Player(com1) attack Dr.Lucky SUCCESS with item: Revolver(Damage=99). This shows that the computer attack logic worked correctly, and it

used the highest item to attack Dr. Lucky when there is a chance to attack. And the Dr. Lucky is successfully killed reduced HP to 0, thus, computer player com1 wins.

9. Line 222: user can confirm that the computer player com1 won the game.
10. Line 235: quit game successfully with order 66. :)

Ms3ExampleRunEscaped.txt

1. This example run logged that the max turns reached thus no winner for the game, and Dr. Lucky, the target escaped successfully with positive HP.
2. Line 33– 67: shows two Computer players are added into the game called com1 and com2, they are both computer players in this case just to speed up the process and improve automation. Also setting the computer players with less capacity for items, for example with capacity, this ensured Dr. Lucky can have higher escape rate.
3. Line 92: turn #1, since com1 are initially in the same with Dr. Lucky so it attacked the target with only Poking, it caused 1 damage, can be confirmed by: **Computer player**: Player(com1) attack Dr.Lucky SUCCESS with item: Poking(Damage=1) Dr.Lucky(Doctor Lucky) HP: -1. Dr.Lucky(Doctor Lucky) Current HP=49
4. Line 95 – line 198: shows some of the players movement in turn#2 – turn#9, since there are total 10 turns in this Demo run. After next turn#10, the target will escape.
5. Line 213: turn #10, Game Over: Max 10 turns finished! Whoops, Dr.Lucky(Doctor Lucky HP=49) escaped!!! This shows, the player failed kill Dr. Lucky within 10 turns of games and the Dr. Lucky go escaped.
6. Line 226: quit game successfully with order 66. :)

Ms3ExampleRunHumanRunAttackFailed.txt

1. This example run logged that the human player tried an attempt on attacking Dr. Lucky, but failed due to be seen by other players in the neighbor room
2. This can be found in log line 130: Player(hum222) Attack failed due to be seen!
3. The player hum222 try attack Dr. Lucky using poking in room #1 Billiard Room, but it failed.

4. **Assumptions for the MS2 and MS3:**

4.1: I assume that the visibility of the game for different players would be their neighboring rooms. Thus, players in neighboring room can see each other with out pet effect. If players can be seen each other they can not execute the attack or the attack will be failed.

4.2: Assume that the total number of players and total turns are set when starting to run the game program, and those settings cannot be changed during the game.

4.3: Assume the player can only choose from one of 5 commands: [Move, Pick, Look, PetMove, Attack].

4.4: Assume each command would consume 1 turn, and each player can only take 1 command during their turn.

4.5: Assume the turn takes place in the order when players were added.

4.6: Assume that Dr. Lucky must move after each player's turn.

4.7: Other intuitive and necessary assumptions that are needed to run my game of killing Dr. Lucky.

4.8: No duplicated item names, player names, room names, and other necessary objects in the game.

4.9: In real game play the Pet will not move in a DFS pattern, due to hard to play and validation. But DFS for pet moving feature is programmed for extra credit and testing purpose. The DFS implementation can be found in the source code.

4.10 Assume that when Dr. Lucky in are not in the room with players, the player cannot execute attempt of attack Dr. Lucky. The system will not give the option for customers to attempt kill if target is not in the same room with the player.

4.11 Assume the user can properly read and parse the English word game description on the command line and make rational choices base on the game rule.

4.12 Assume the items in the game can all be consumed once, once used it will remove from the game. Reuse the same item will cause error/warnings to the user.

4.13 Assume the Pet effect will be acting as a cloud in a single room, if a pet present in a room this room's item and players will not be seen from outside, but inside players can see each other.

4.14. Assume if the Pet in neighboring room, and player execute look around command will still see the room name as its neighbor, but will not be able seen its inside, the player inside will not be seen then, the player inside room with pet can still kill Dr. Lucky if that is possible.

5. Limitations for MS2 and MS3:

5.1: Include all assumptions from above.

5.2: The game will only end when the maximum number of turns is reached.

5.3: The players cannot stop the game during the gameplay turn. Players can quit the program in the Main Menu.

5.4: Once the Game Over, the user cannot play anymore and must quit and start the program again to play.

5.5: Computer players will move randomly in the world and always pick items in the room starting with the highest damage on the item as long as it has carrying capacity.

5.6: The world map file will be generated at a fixed path with a fixed file name base on given.

5.7: Any other necessary limitations that are needed to run my game design of killing Dr. Lucky.

5.8: Computer player will be able to attempt attack target, and if it know it will been seen it will not attack the target. When computer attacks, it will always try to use its highest damage item, it no items available, it will just use the basic poking attack.

5.9: Computer player will act command with following priority in order: attempt attack, pick, move, and look around. The computer player will not move pet, as it's not smart enough yet.

5.10: At beginning of each player's turn, the system will remind the current player their room info but will not show neighbor room info(such as items, players, those neighbor info need to be get by look around command). But the customer will tell the user if the Dr. Lucky is in their current room or not. This will make the game more playable and user friendly.

5.11: If the attack failed due to item name input error or used item, the system will remind the player try to use another item and try attack again.

5.12Pet remain stationary for most of the time unless move, or it can be activated for DFS move, this will increase the game difficulty, and hard to valid on command line interface thus not recommended in game play. But can be tested with test cases and unit test.
5.13: If the attack failed due to be seen by others, the system will count this turn and pass, let player know attack failed but will not remove item since the item did not take effect. The player may use the item again next time.

6. Change and Version History v2.0 - v3.0:

6.1: From v1.0 the milestone 1, players models are introduced.
6.2: Synchronous Controller Controller introduced along with MockWorld testing method.
6.3: Necessary changes in World and World Implementation to accommodate the Synchronous
Controller introduced in this version.
6.3: The controller would run the program and play the game in a command line interface.
6.4: All other necessary reversion and changes to accommodate the new introduced feature and
requirements. For more details please refer to the Project GitHub Repository for detailed commit
history: https://github.com/xue59/KillDoctorLuckyGameDesign/tree/Milestone3
6.5: Command instruction feature introduced to accommodate game play, players can give Move, Look, Pick, Attack, PetMove command during the game play, as well as enquiry a player or room specific information.
6.6: Detailed design change please refer to Milestone3_PreliminaryDesignV3.pdf in
the resource folder.
6.7: Mock World Model also introduced and designed for testing purpose.
6.8: In v3.0 game play is completed, the pet feature is added into the game.
6.9: Pet will affect the visibility, more detail see above assumption and limitation.
6.10: Pet can be moved by players, this count as a turn.
6.11: Pet remain stationary for most of the time unless move, or it can be activated for DFS move, this will increase the game difficulty, and hard to valid on command line interface thus not recommended in game play. But can be tested with test cases and unit test.
6.12: Attack command introduced in the game, see rule in above assumption and limitation.

References:

- https://www.youtube.com/watch?v=9hsPLAB5a9s&ab_channel=fenirob
- https://docs.oracle.com/en/java/javase/11/docs/api/java.desktop/java/awt/Graphics.html
- https://docs.oracle.com/en/java/javase/11/docs/api/java.desktop/java/awt/image/BufferedImage.html
- https://www.jetbrains.com/idea/guide/tutorials/hello-world/packaging-the-application/
- https://www.jetbrains.com/help/idea/compiling-applications.html#run_packaged_jar
- https://steamcommunity.com/sharedfiles/filedetails/?id=261218852
- https://docs.oracle.com/javase/8/docs/api/java/util/Random.html