Addison Correll

Computer Science (4)

12/1/30

Interactive Fiction Essay

My game brings the player on a journey to find the last cat or dog after the great war. The player has to maneuver their way through the wasteland caused by the feud between cats and dogs. They must make decisions that could bring them supplies and help, or danger and death. I was able to use methods, strings, booleans, scanners, if statements, and while loops to create this game, and I will explain how all of these come together to make the game and its components function.

One of the first requirements of the game was to have a protagonist that affects other objects in the game. I did this by allowing the player to create their own protagonist to control out in the wastes. To do this, I created a method that allows the player to insert the protagonist's name and gender, which is all that was needed to create the protagonist.

```
Scanner name = new Scanner(System.in);
	System.out.println("Player's Name Is...");
	playername = name.nextLine();
	//What's the player's name?
	Scanner sex = new Scanner(System.in);
	boolean goodanswer = false;
	while (!goodanswer) {
		System.out.println("Player's Gender Is...");
		gender = sex.nextLine();
		if (gender.contains("F") || gender.contains("f")) {
```

```
            goodanswer = true;
            mf = false;
            tpp = "she";
            tppp = "her";
            tpop = "her";
            } else if (gender.contains("M") || gender.contains("m")) {
            goodanswer = true;
            mf = true;
            tpp = "he";
            tppp = "his";
            tpop = "him";
            } else {
            System.out.println("I don't understand. \n");
            }
      }
      //Determines the player's gender depending on what you type
      #hero
      //this is the character creation of the game, making the protagonist to
come to life
```

The protagonist affects the world by making certain decisions that the player chooses them to do, and they are able to find and take objects in the world, such as a can of sardines.

```
Scanner first = new Scanner(System.in);
      boolean goodanswer = false;
      while (!goodanswer) {
```

```
	System.out.println("As " + playername + " drives along, " + tpp + "
hears a loud thump. Does " + tpp + " investigate the noise, or leave?");
	response = first.nextLine();
	if (response.contains("I") || response.contains("i")) {
		System.out.println(playername + " gets out of the car to
investigate, and finds a trash can on the side of the road, dented from where
" + tpp + " hit it. Inside, " + playername + " finds a can of sardines. " +
tpp.substring(0, 1).toUpperCase() + tpp.substring(1) + " gets back in the car
and continues driving. \n");
		goodanswer = true;
		sardines = true;
		//This is what you receive after taking this route
			#hero
			//this is an example of how the protagonist affects the
world
	} else if (response.contains("L") || response.contains("l")) {
		System.out.println(playername + " continues driving down the
street. \n");
		goodanswer = true;
	} else {
		System.out.println("I don't understand. \n");
	}
	}
```

This is how the player/protagonist affect the world and change their story based on what they decide.

The next requirement for the game was to have a working enemy that would affect the player in some way. I did this by having a few enemies in the game that could end the game if the protagonist did not make a good decision. I used multiple methods for these encounters, and booleans as to whether or not the protagonist has died/lost the game.

```
System.out.println("As " + playername + " drives down the street, the sound of a gunshot rings out across the open plains. Does " + tpp + " get out of the car, or will " + tpp + " drive away?");
	response = second.nextLine();
	if (response.contains("G") || response.contains("g")) {
		System.out.println("A woman with a revolver runs up from behind a bush and grazes " + playername + "'s shoulder. Before " + playername + " can retaliate, the woman sprints away. " + playername + " bleeds out on the dry grass.");
		youLose = true;
			System.exit(0);
			//Player loses the game
	} else if (response.contains("D") || response.contains("d")) {
		System.out.println(playername + " continues driving down the street. \n");
		goodanswer = true;
	} else {
		System.out.println("I don't understand. \n");
	}
	}
```

However, the enemies do not harm the player in that they lose health, but in that one little decision could end the player's journey. That is how I implemented an enemy into my game.

The next requirement was originality. There is not much code to support this, but I believe the story that I created my game around was very unique. After an old woman's cat was supposedly torn apart by her neighbor's doberman, the world began to try and end the feud between cats and dogs. However, the discussions quickly became arguments, and soon the arguments tuned into fights, and then a war. Nothing was left to fight about, as the animals had fled, and nobody to fight with, as people also went into hiding to avoid the war. The protagonist of the game is one of the few people left who are willing to go out and find the last cat or dog, and to see which pet truly came out on top. I believe that this story is very original, and very different than everyone else's stories, to make it truly stand out.

The next requirement was to have at least five working methods in the game. I included seven methods in my game, the first one being the intro and the character creation. The next five methods were five different scenarios that the player will encounter, therefore there will be at least five different decisions to be made by the player. The final method is the finale to my game, where the final scenario is shown and the final decision is made. This is where the player either wins or loses the game. The entirety of the game is separated into these seven methods, and I would not have a game without them.

The next two requirements are to have a clear win and lose goal, as well as having a win and lose state. I included this into my game by having an intro explaining what the protagonist's goal is, which will help influence the player to make decisions to get to this goal. This also made clear how the protagonist could fail in this game, and how the player could choose decisions that would influence their loss. I then used booleans to set if the win and lose states after the player

has completed a certain objective. For example, if the player makes a decision that will accomplish the protagonist's goal, then the player will win, and vice versa. I also used `System.exit(0);` to end the game completely, forcing the player to restart the game to play again.

```
            if (response.contains("Y") || response.contains("y") && sardines)
{
            System.out.println("The cat comes closer as " + playername + "
gives it the food, and " + playername + " sees the cat's collar. The cats
name is Walter. This is the end of " + playername + "'s quest.");
            youWin = true;
            System.exit(0);
            //Player wins the game
            } else if (response.contains("N") || response.contains("n") &&
sardines) {
            System.out.println(playername + " stares at the cat, refusing to
give up " + tpp + " food. The cat hisses and runs away.");
            youLose = true;
            System.exit(0);
            //Player loses the game
            } else if (!sardines) {
            System.out.println("The cat grows hungry, and dies of
starvation.");
            youLose = true;
            System.exit(0);
            //Player loses the game
```

```
		}
	}
```

This is how I implemented a win and lose goal/state into my game.

The next requirement for the game was to have at least one while loop. I implemented one while loop into my game by having a boolean that would check the response of the player after a question is asked to determine if it was a good answer or not. If the answer was acceptable and my game understood what they typed, then the game would continue. However, if the response did not follow the guidelines, then the game would tell the player that the answer was unacceptable, and the question would be repeated again.

```
Scanner sex = new Scanner(System.in);
	boolean goodanswer = false;
	while (!goodanswer) {
	#while
		System.out.println("Player's Gender Is...");
		gender = sex.nextLine();
		if (gender.contains("F") || gender.contains("f")) {
		goodanswer = true;
		mf = false;
		tpp = "she";
		tppp = "her";
		tpop = "her";
		} else if (gender.contains("M") || gender.contains("m")) {
		goodanswer = true;
```

```
        mf = true;
        tpp = "he";
        tppp = "his";
        tpop = "him";
        } else {
        System.out.println("I don't understand. \n");
        }
}
```

This is how I implemented a while loop into my game.

Overall, my game follows all of the requirements given as well as being original and unique to me. I have a protagonist that the player controls, as well as enemies that can affect the player's journey. I have included at least five methods that make the game what it is. I have clear win and lose goals, as well as win and lose states. Finally, I have one while loop and I included code comments explaining the components of my game.