Ahmed Hassanin

Tasneem

Rotaj Tamer

1. First meeting Nov 13, 20228:00 PM on Zoom.

* I assigned us to watch a video explaining basic game-related concepts & libraries in QT and also assigned the project notes along with the video to be read & watched before the meeting.
* Assignments: Ahmed → all the maze mapping
* Assignments: Tasneem → Implementing the enemy class
* Assignment: Rotaj → Implement Franklin & Collectables
* The enemy class should include random movement of the enemies and their starting positions and other stuff mentioned in the project doc
* Franklin & collectables should also include everything mentioned in the doc including disappearing the bullet after collecting them etc
* We all are going to use the playlist vids I sent on Whatsapp & the lab doc related to the project on blackboard along with the project doc as references
* I will start the map with any pics for now and we could change the theme later

.

1. Second meeting Nov 17, 2022 11:30 Library

* Complete the classes
* Watch the playlist
* Use githup to collaborate and test
* Complete all the classes by Friday 8 pm

Ahmed Hassanin:

1. I wrote the board file and drew the map after reading the file. The map has different images in different locations such as the four corners, the horizontal road, and the vertical road, and all the intersections. Moreover, I tried to add on a cat theme, where the main character is a cat, the powerpellets are fish, and the enemy is a big evil cat.
2. I wrote most of the main character, bullets, and powerpellet classes. In main character, i wrote the main structure of the class, angrymeow function turns the cat into a different image for 10 seconds in the power mode, gobacktonormal function it brings the cat back to normal after taking a bullet or a powerpellet, controlled the movement of the main character while checking with every movement if the main character is colliding with an enemy or a collectible, in which case a different response happens depending on what type of item the main character collides with. Also wrote holdpistol function which changes the image of the cat once it collides with bullet for one second then we use the function gobakctonormal to go back again to the normal image of the cat (main character).
3. Wrote the bullets classes, gave them initial positions in the constructor, and controlled their disappearance once collided with main character
4. Wrote class for powerpellet and gave it positions in the constructor and also controlled its disappearance once colliding with main character.

Tasneem:

1. Wrote the enemy class structure, and inside it wrote that each enemy has two lives and when colliding with Franklin their position changes and then they continue their movement
2. Wrote an algorithm to control random movement for the two enemies and had their positions in the bottom middle of the map.
3. Wrote the function for not affecting the main character once colliding with a powerpellet, so when an enemy collides with Franklin while he is powerful, he does not lose a life, but then he goes back to his normal state after 10 seconds.
4. I wrote the code the controlls the winning process, as first a function to calculate the distance between the the enemies and the main character when he collides with the bullet, and the enemy that is closer to him loses a life, and when both the enemies lose their two lives a house appears in the middle of the map so that indicates that Franklin has won.
5. Also I changed some blocks of the map so it is not all linear and have some obstacles.

Rotaj:

1. I wrote the base of the class of the main character that Ahmed continued working on afterwards, I wrote the if conditions which are responsible for moving the main character, and testing beforehand if it's an obstacle or not, so if it is an obstacle the main character won't move in that direction, and I also connected that to when the user presses the keyboard button.
2. I also generally helped ahmed and Tasneem in implementing the classes of the project as we worked together sometimes and thought about the algorithm in which we will implement the classes and the functions