REPORT

Github Repo: https://github.com/Rekha440/SME-Assignment/

Part 1: Setting up the Project
This took around 1 hour 30 mins, due to poor internet connectivity and the large download size of Visual Studio. The actual setting up of the project took at most 5 mins.

Part 2: Resolve Bug
This took around 1 hour. This included a surface level understanding of how the project worked, what each of the files and folders did, and how they connected with each other. This also included finding which files were responsible for the bug, and then finally finding out how to fix the bug.

There was a circular dependency between PlayerView.h and PlayerController.h
- PlayerView needed PlayerController, and PlayerController needed PlayerView.

To fix this,

- I removed #include "../../header/Player/PlayerController.h" from PlayerView.h
- Added forward declaration of PlayerController in PlayerView.h

```
namespace Player {
    class PlayerController;
}
```

(A forward declaration is basically just declaring a class, function, etc. before providing the actual body for it)

- PlayerView.cpp was already including PlayerController.h, so the circular dependency issue was resolved.

Part 3: Partially Added firing mechanism
This took around 3 hours, where most of the time was spent on digging through the code, identifying where to make changes and identifying which files were responsible for the feature that I was trying to implement.

I was unable to proceed further with Part 3. Currently, enemies in front of the player ship are getting destroyed when the user clicks the mouse. However, there are no bullets being fired visibly. I messed around with the code for a long time, however, I was still unable to find out how to get it done. Therefore, I could not proceed any further.

Made changes in header/Player/PlayerController.h:

- Added function declaration

  void processBulletFire();

Added implementation in source/Player/PlayerController.cpp:

- Added function implementation:

```
void PlayerController::processBulletFire() {
      sf::Vector2f currentPosition =
      player_model->getPlayerPosition();
      ServiceLocator::getInstance()->getEnemyService()->dest
royEnemyAlongLine(currentPosition.x);
}
```

- Also called function on player input, PlayerController::processPlayerInput():

```
if (event_service->pressedLeftMouseButton())
      processBulletFire();
```

Made changes in header/Enemy/EnemyService.h

- Added function declaration

  void destroyEnemyAlongLine(float x_coord);

Added implementation in source/Enemy/EnemyService.cpp

- Added function definition

```
void EnemyService::destroyEnemyAlongLine(float x_coord) {
      for (EnemyController* enemy : enemy_list) {
            float enemy_pos = enemy->getEnemyPosition().x;
            if ((enemy_pos > (x_coord - 20)) && (enemy_pos <
            (x_coord + 20))) {
                  destroyEnemy(enemy);
                  enemy->processScore();
            }
      }
}
```