

**2016113565 심석인**

**2017111573 최시운**

**2017111405 유연아**

**목차**

1. **Bases**
2. **구현 내용 및 결과**
3. **역할분담**
4. **최종 변경사항**

**1. Bases**

**I . 선정 게임 정보**

게임 이름 : Shooting game

사용 언어 : python

추가 모듈 : pygame

**II. 개발환경**

언어 : Python

편집기 : VS code, Atom etc..

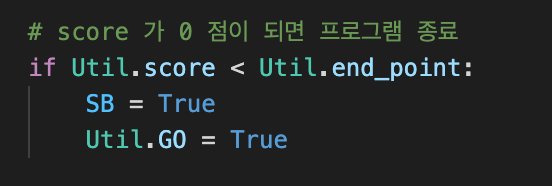
OS : Window & MacOs

**2. 구현 내용 및 결과**

* **배경 이미지에 맞게 자동차 위치 개선**

자동차의 왼쪽과 오른쪽의 움직임의 경계를 조절하여 자동차가 배경의 도로와 맞게 이동할 수 있도록 개선하였습니다. 도로 밖으로 이동하지 못하는 자동차로 레이싱효과를 더욱 극대화하였습니다.

* **장애물을 지나쳐도 게임 지속**

****

장애물을 많이 지나치게 되면 게임이 끝나는 설정으로 초반 점수가 0일때 쉽게 게임이 끝나는 것을 수정하고자 하였습니다. 따라서 장애물에 충돌하지 않아도 제거하지 않은 장애물이 늘어나면 게임이 끝나는 기존의 설정인 위의 코드를 제거하여 장애물 충돌 시에만 게임이 종료되도록 합니다.

* **음향효과 개선**



미사일을 많이 발사하는 경우 배경음악이 들리지 않는 문제를 개선하고자 전체적인 음향을 조절하였습니다.

* **재시작 기능**

Restart() 함수 생성 후 게임 시작을 위한 정보와 Main 루프를 삽입했습니다. 코드를 추가하여 기존 750줄에서 약 1400줄로 증가하였습니다.

Game Over후 R버튼을 우르면 Restart함수를 불러오는 방식입니다.

* **미사일 속도**

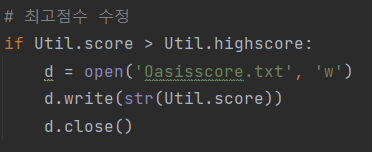
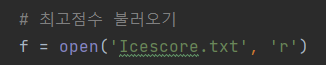
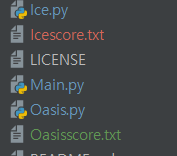
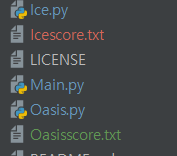
이전에는 장애물 등장 횟수에 비해 미사일 속도가 너무 느려서 초반에 게임이 종료되는 경우가 많았습니다. 이를 방지하고자 미사일 속도를 기존 15에서 25로 변경했습니다.

* **최고점수 기록추가**



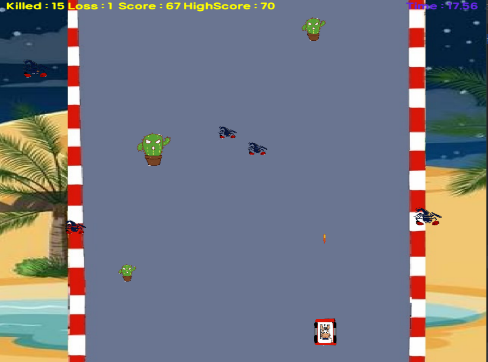
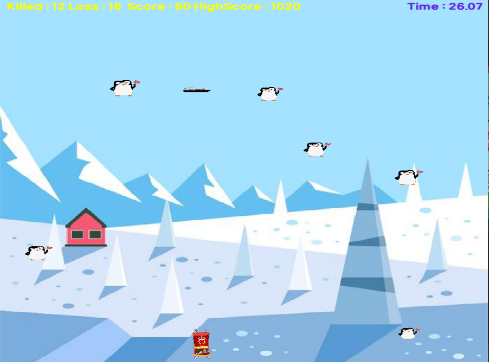
저희 조는 개인의 최고 점수를 저장하여 유저에게 보여줌으로써, 지속적으로 게임을 할 수 있는 동기인 도전심을 부여하였습니다.

구현은 각 모드마다 점수를 저장하는 txt파일을 생성하여 해당 파일에 최고 점수를 저장하고, 게임이 시작될 때 불러오는 방식을 통해 구현하였습니다. 먼저 파이썬의 open()을 활용하고, ‘r’의 속성을 부여하여 읽기 전용으로 점수를 불러온 후, 화면에 띄워주게 됩니다. 게임이 마무리되면 조건문을 이용해 현재 점수가 저장되어 있는 최고 점수보다 높을 때, 파이썬의 open()으로 각 모드의 txt의 내용을 부르고, ‘w’의 속성, 쓰기 속성을 이용해 저장된 점수를 수정하는 방식으로 점수를 부여하는 형태로 구현하였습니다.

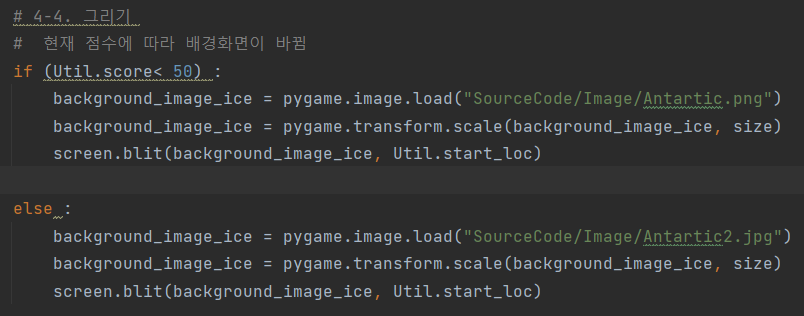


* **배경화면 변경**

점수에 따른 배경화면 변경을 도입하여 스토리가 진행되는 느낌을 줌과, 다음에 어떠한 배경이 나올지에 대한 궁금증을 유도하여 게임에 대한 훅적인 요소를 부여하였습니다.

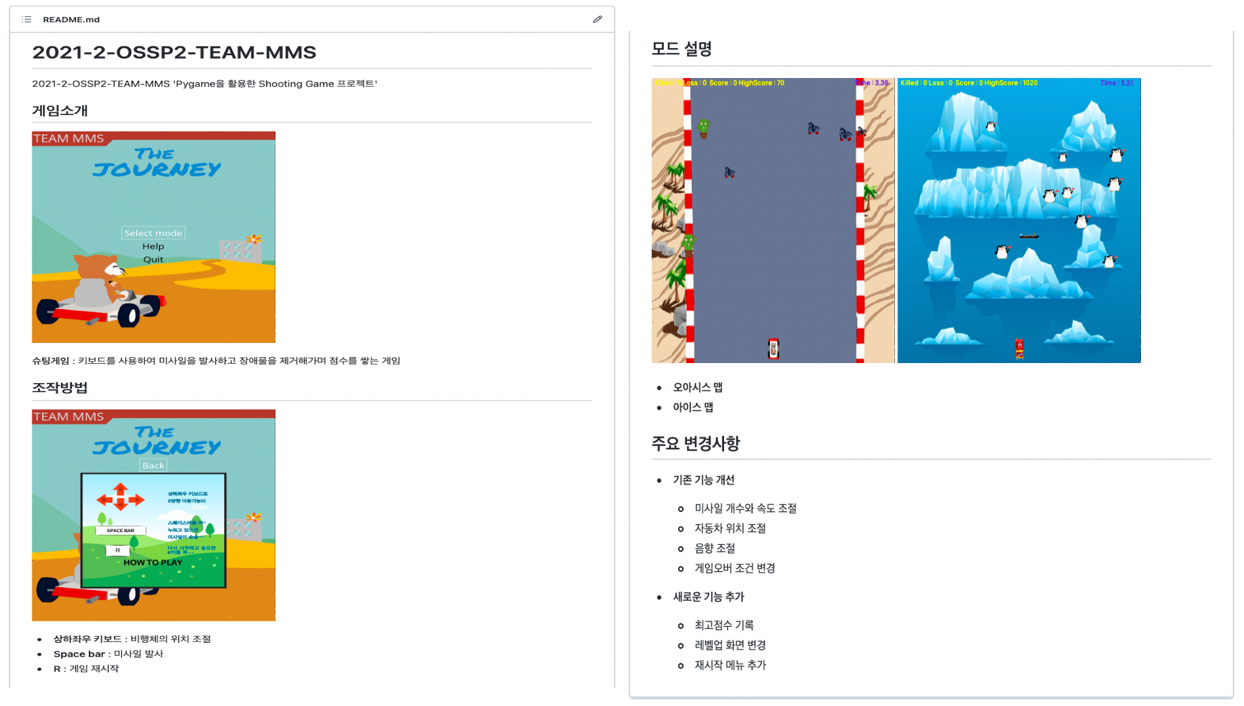


현재 점수 50점을 기준으로 배경이 바뀌게 되며, 구현 방식은 조건문을 통해 게임이 진행되는 동안 현재 스코어가 기준 점수를 넘었는지 넘지 않았는지를 판단하며, 점수를 초과하게 될 시, pygame의 image.load기능을 이용하여 이미지를 배경으로 불러오게 됩니다.



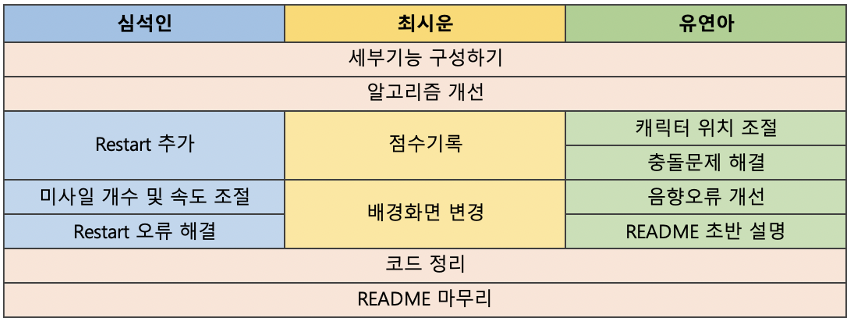


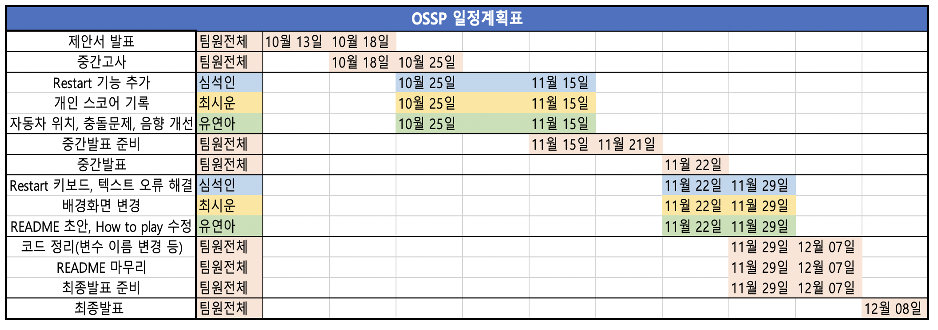
* **README 변경**



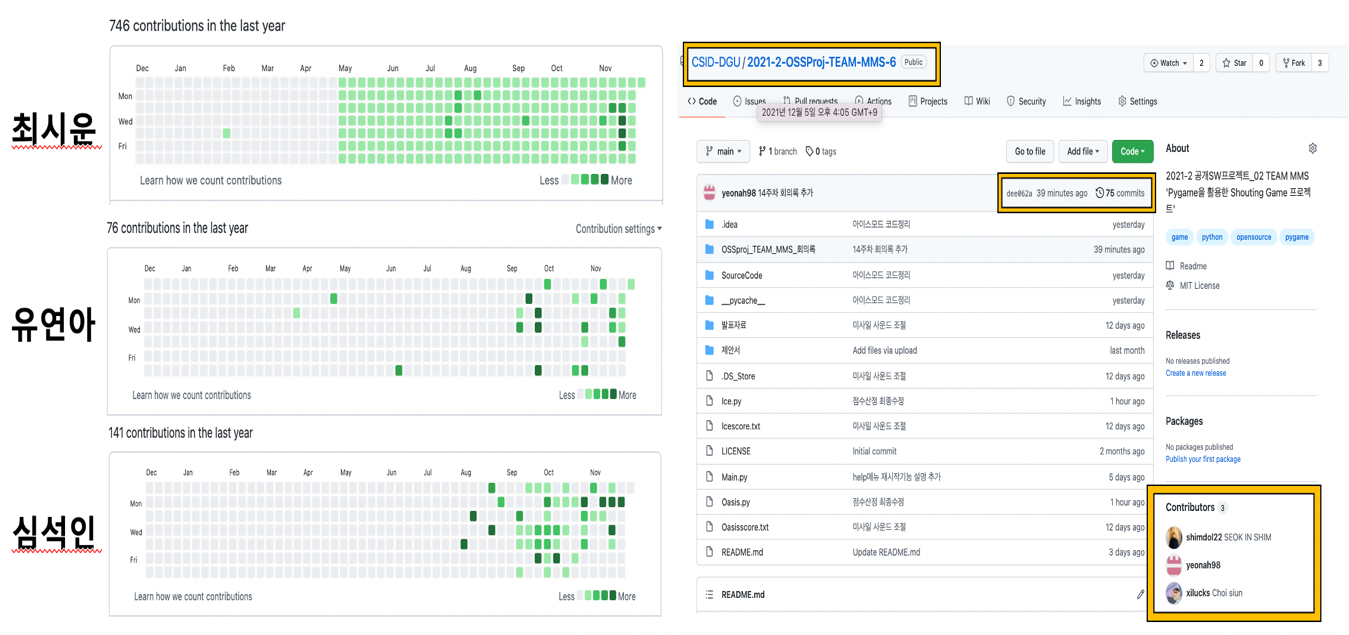
1. **역할분담과 일정계획표 및 깃허브**

**Ⅰ. 역할분담**

****

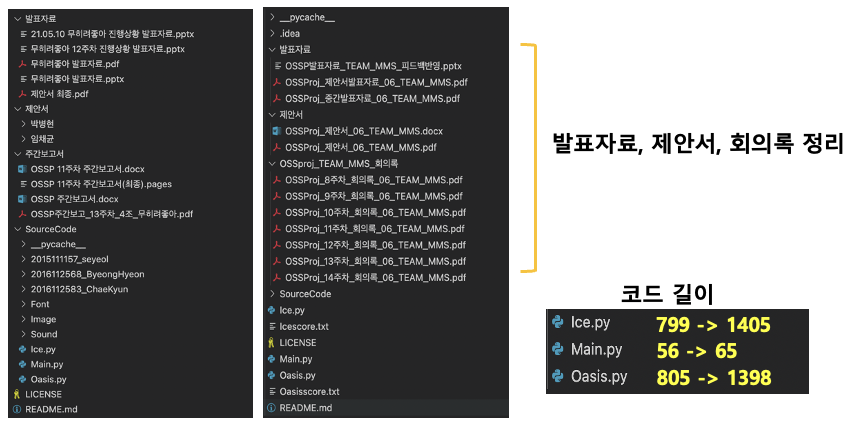
**Ⅱ.일정계획표**

**Ⅲ. 깃허브**

****

팀원 모두 일정계획표에 맞게 최소 일주일에 한 번씩 커밋을 완료했으며 총 75회의 커밋을 통해 프로젝트를 완성하였습니다.

**4. 최종 변경사항**

****