****

2021-2 오픈소스소프트웨어프로젝트 2조 DoitDoit

팀장 고명섭 2014111472

팀원 고다희 2018112498

팀원 김수빈 2018112453

팀장 고명섭 2014111472

팀원 고다희 2018112498

팀원 김수빈 2018112453

팀장 고명섭 2014111472

팀원 고다희 2018112498

팀원 김수빈 2018112453

**팀장 고명섭 2014111472**

**팀원 고다희 2018112498**

**팀원 김수빈 2018112453**

**목차**

**1. 프로젝트 개요**

1.1 기존 프로젝트

1.2 DoitDoit 프로젝트

**2. DoitDoit Tetris**

2.1 기존 기능 개선

1) 하드 모드 장애물 구현

2) 피버 타임 방식 변경

3) 오류 개선

4) PvP 모드 플레이어별 soft drop 구현

5) UI 개선

2.2 새로운 기능 추가

1) 로그인 기능

2) 기록 AWS 연동

3) 모드별 시작 속도 조절 기능

4) 아이템 모드 구현

**3. 기존 프로젝트와 비교**

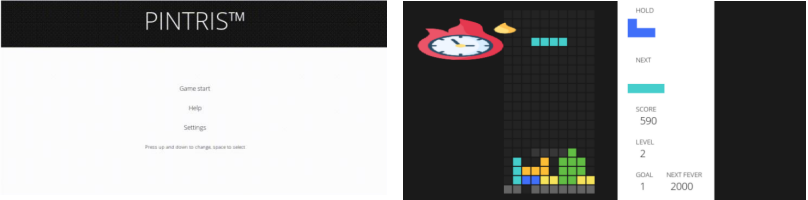
**4. 프로젝트 일정**

**5. 역할 분담**

**1. 프로젝트 소개**

* 1. **기존 프로젝트**

파이썬 모듈 중 pygame을 활용해 제작된 ‘PYTRIS’를 수정한 프로젝트로, 싱글 모드를 개선한 Easy, Normal, Hard 모드, PvP 모드, Speed&Mini 모드와 리버스 모드로 이루어져 있으며 직관적 UI, 사운드, 그래픽 개선 등을 구현하였다.



- https://github.com/CSID-DGU/2021-1-OSSPC-Pint-9

■ 개발 환경 및 라이선스

라이선스: MIT-License

OS: Ubuntu

언어: Python

구성: pintris.py (PINTRIS 게임 구현)/Mino.py (테트리스 블록 구현)/assets (폰트, 이미지, 사운드 파일 저장/leaderboard.txt (유저 테트리스 점수 저장)

* 1. **DoitDoit 프로젝트**

오픈소스를 이용하여 테트리스 게임을 개선하는 것을 목표로, 2021학년도 1학기 진행된 파인트 팀의 PINTRIS를 기반으로 기존 모드들을 수정하고 새로운 모드를 추가하여 게임의 재미 요소를 높였고 기존 게임이 가지고 있던 오류와 UI를 개선하여 게임을 발전시켰다.

■ 개발 환경 및 라이선스

Language: Python 3.9 이상

OS: Ubuntu

Tool: Visual Studio Code

License: MIT

Collaborate: GitHub

■ 프로젝트 개발 사항

1) 모드 수정

: Easy 모드와 Speed&Mini 모드 삭제 & Normal모드에서 속도 조절 버튼 생성

2) 랭킹 입력 방식 수정 및 AWS 연동

- 동일 이름으로 등록 시 최고점만 기록 & AWS를 이용한 랭킹 연동 시스템 구축

3) Fever 기능 수정

- 점수에 따라 Fever Time 발동 시간에 차등을 둠

4) Item 모드 추가

- 500점 단위로 점수에 도달하면 랜덤 아이템 생성 → 인벤토리에 저장 후 사용

5) Hard 모드 변경

- Hard 모드를 게임 중 방해 요소가 있는 모드로 변경

6) UI 개선 & 버그 수정

- 전체적인 UI 개선, 블록 생성 오류 개선, 리버스 모드 수정

**2. DoitDoit Tetris**

**2.1 기존 기능 개선**

**1) 하드 모드 장애물 구현**

\* 기존: 노말 모드에서 속도만 빠른 하드모드

\* 변경: 방해 요소가 있는 모드로 재구성하여 난이도 향상을 위해 next 블록을 알 수 없도록 하고 레벨업 시 장애물 블록을 생성하는 하드모드

<변경된 화면1>



: 하드모드에서는 next 블록을 알 수 없음

<변경된 화면2>



: 노말 모드에서는 레벨업 시 한 칸이 빈 장애물 블록이 생성되어 게임을 하면서 지울 수 있지만, 하드모드는 레벨업 시 아래에서부터 지울 수 없는 장애물 블록이 한 줄씩 생성되도록 구현

**2) 피버 타임 방식 변경**

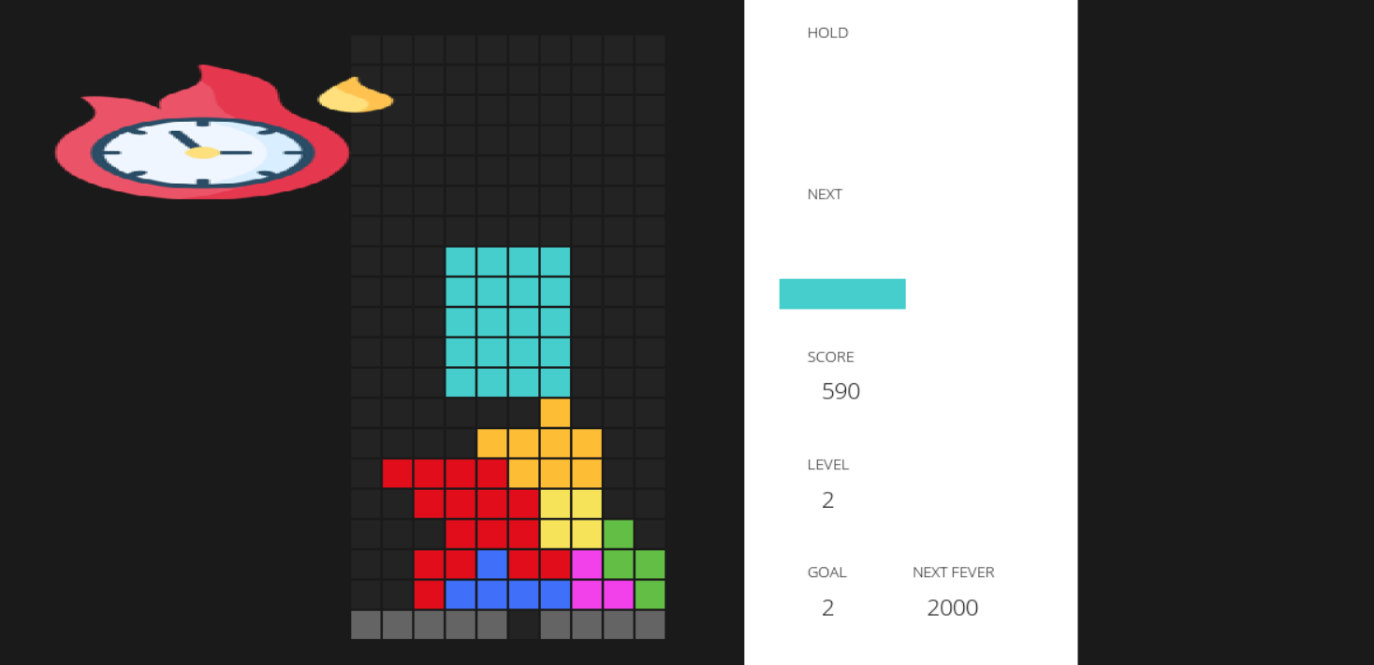
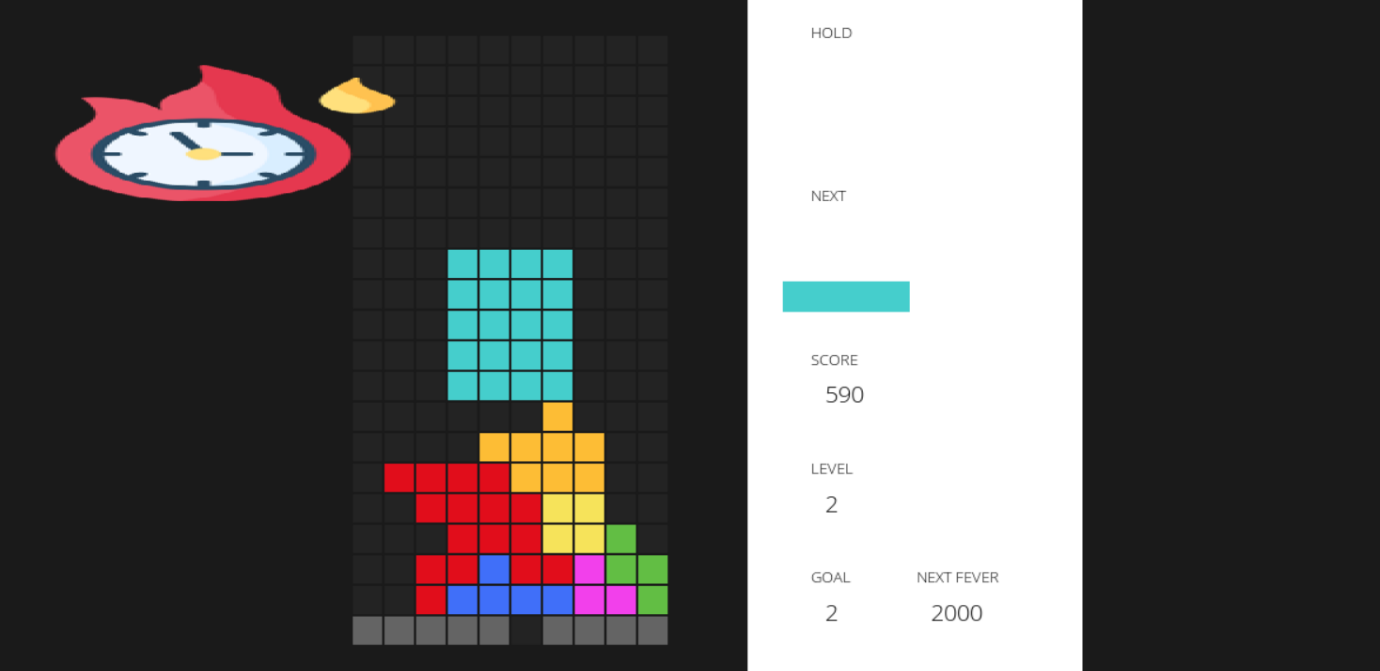
\* 기존: 피버 타임 발동 시 500점 추가 획득하면 피버 타임 종료

\* 변경: 점수가 높을수록 피버 타임을 줄여 낮은 점수에서 점수 획득을 많이 할 수 있도록 변경하고, 유저가 남은 피버 타임을 직관적으로 알 수 있도록 타이머 바 추가

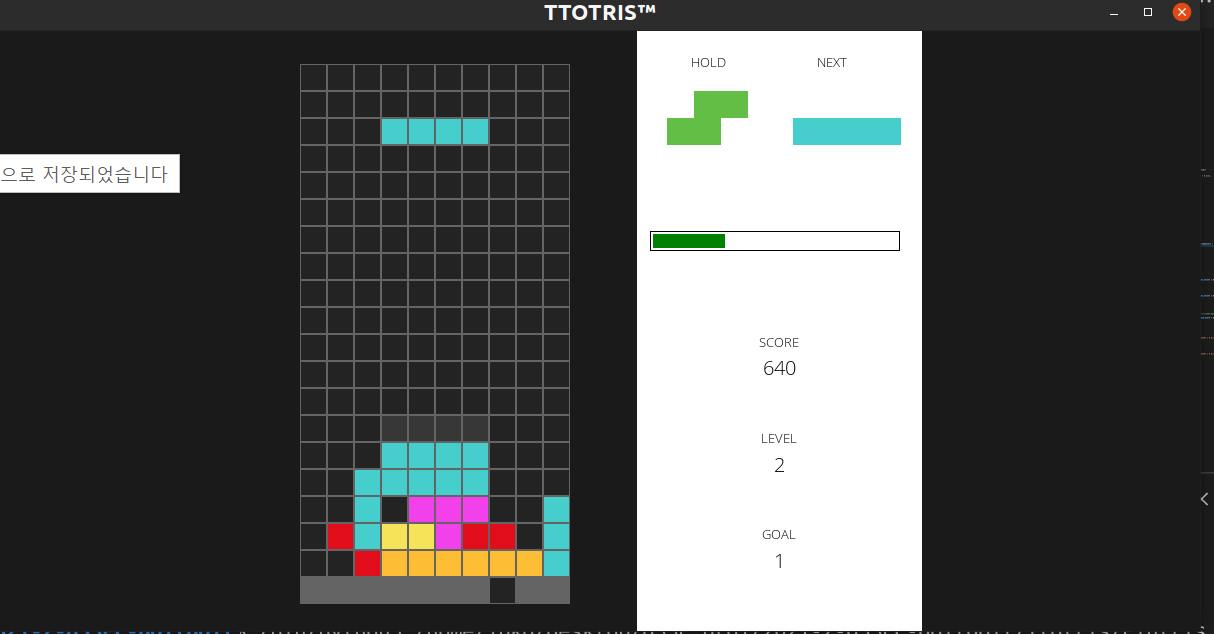
<기존 화면>

**불필요한 정보 삭제**

**게임에 방해가 되는 이미지 삭제**



<변경된 화면>



**남은 피버 타임을 알려주는 타이머 바 추가**

**3) 오류 개선**

■ 블록 생성 오류 개선

\* 기존: 블록이 바닥에 닿은 후 오랜 시간 움직이기가 가능한 버그 발견

\* 변경: 게임 초기 set speed에서 블록이 생성되는 간격을 감소시켜 해결

■ 리버스 모드에서 블록이 떨어지는 방향 수정

\* 기존: 화면 상하 반전

\* 변경: 화면은 그대로, 방향키만 reverse 되도록 수정

**4) PvP 모드의 플레이어별 soft drop 구현**

\* 기존: 아래쪽 화살표를 길게 누르면 화면 출력 속도를 높이는데, single 모드들에서는 문제가 없었으나 PvP 모드에서 각 플레이어 별로 하강 속도 조절이 불가능하다는 문제점 발견

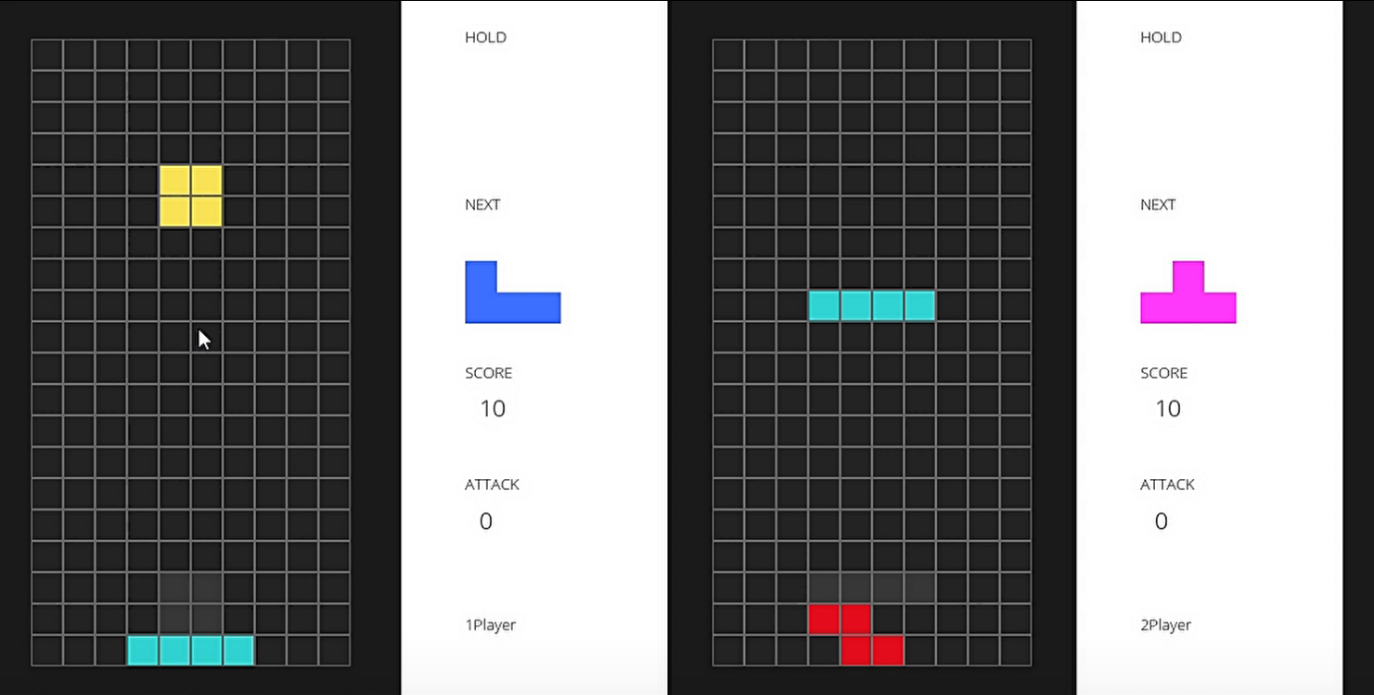
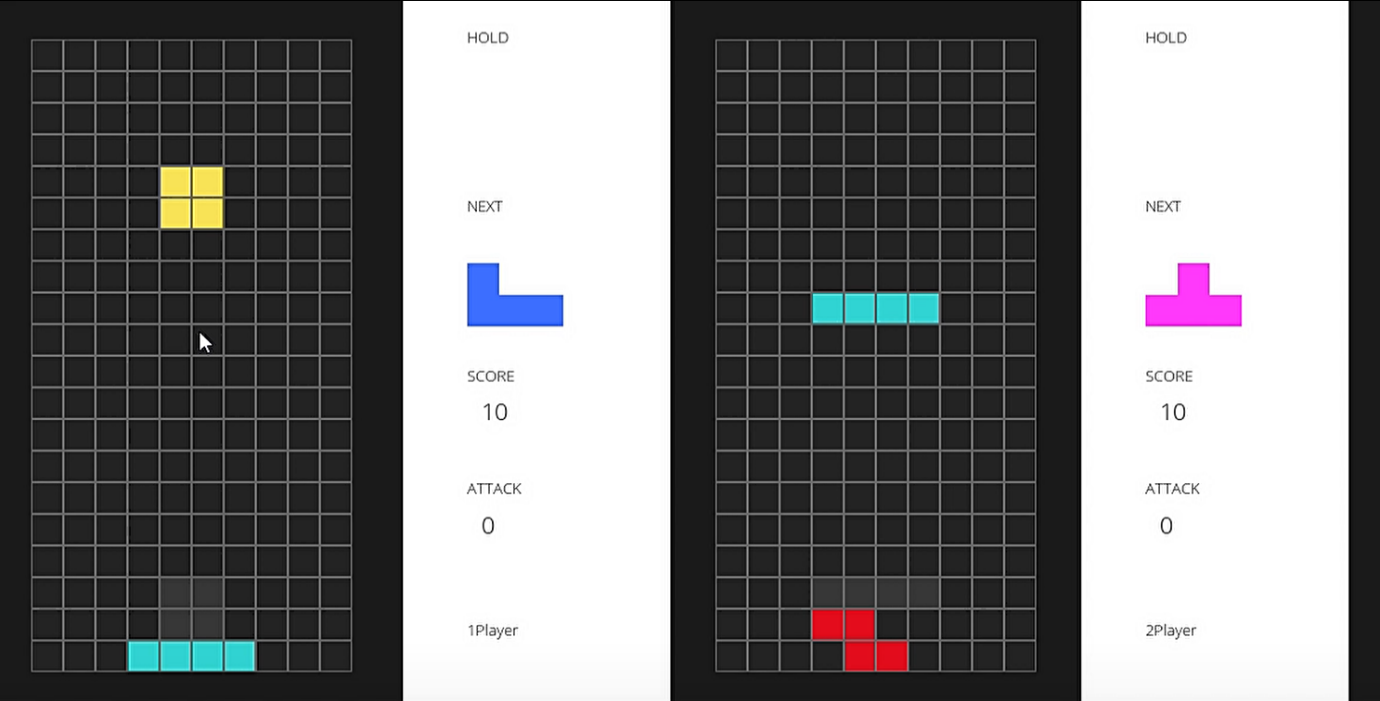
\* 변경: 아래 화살표 또는 S 키를 눌렀을 때 블록의 높이에 해당하는 값을 하나씩 증가시켜 하강 속도가 빨라지는 효과를 내는 soft drop 기능 구현

<변경된 화면>

**soft drop 키 조작 시**

**빠르게 떨어지는 효과**

**키 조작 x**



: 2p(화면 상 오른쪽)만 아래 화살표로 soft drop 했을 때의 게임 화면으로, 2p의 블록만 더 빨리 내려와 있는 것 확인

**5) UI 개선**

■ HELP 페이지 제작

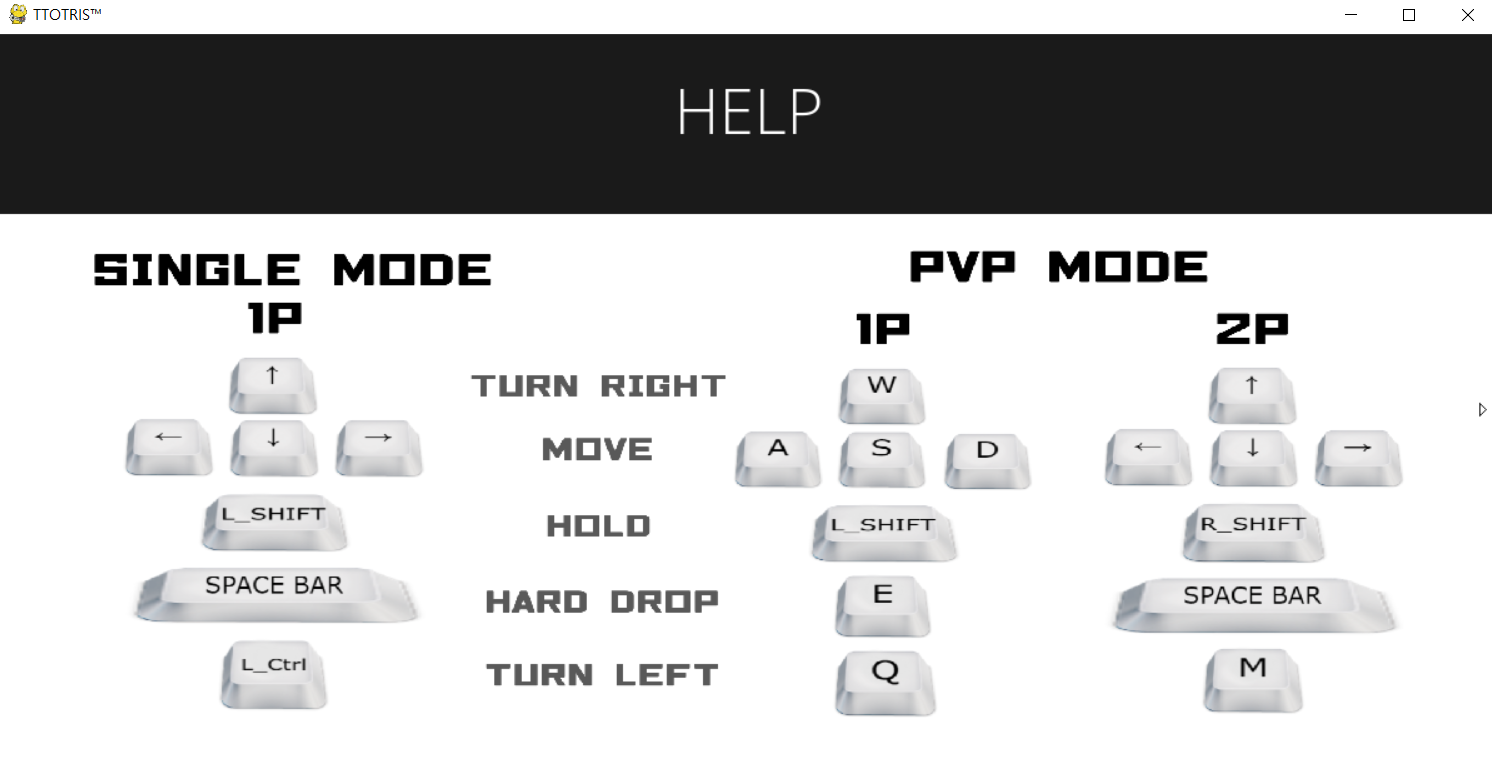
: 기존의 HELP 페이지의 설명 부족 + 아이템 모드로 인한 키 조작의 추가로 HELP 페이지를 수정

<기존 화면>

테이블이(가) 표시된 사진

자동 생성된 설명

<변경된 화면>

 텍스트이(가) 표시된 사진

자동 생성된 설명

**아이템 모드 키 조작법**

**싱글 모드와 PvP 모드 키 조작법**

■ 사운드 추가

: 배경음 추가, 아이템 사용 시 효과음 추가

* 1. **새로운 기능 추가**

**1) 로그인 기능 생성**

-> 명섭님 14주차 회의록 보고 추가

**2) 기록 AWS 연동**

- 모드별 AWS 테이블을 구성하여 ID당 최고 기록만 업데이트 되도록 설정하고 모드별 상위 3명의 기록을 각 모드 페이지에 출력하도록 구현

<기존 화면>

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

: 모드별 구분 없이 초기 화면에 상위 랭킹이 표시됨

<변경된 화면>

텍스트이(가) 표시된 사진

자동 생성된 설명

: 모드별로 상위 랭킹이 표시됨

**3) 모드별 시작 속도 조절 기능 생성**

- 기존 게임에서 속도만으로 구별하는 싱글 모드들의 개성이 부족하다고 판단하여 무의미한 구분인 Easy 모드 및 Speed&Mini 모드 삭제 후 Normal 모드 안에서 속도 조절이 가능하도록 구현

- 모드를 선택하고 게임이 시작되기 전 속도를 조절하는 페이지 구현

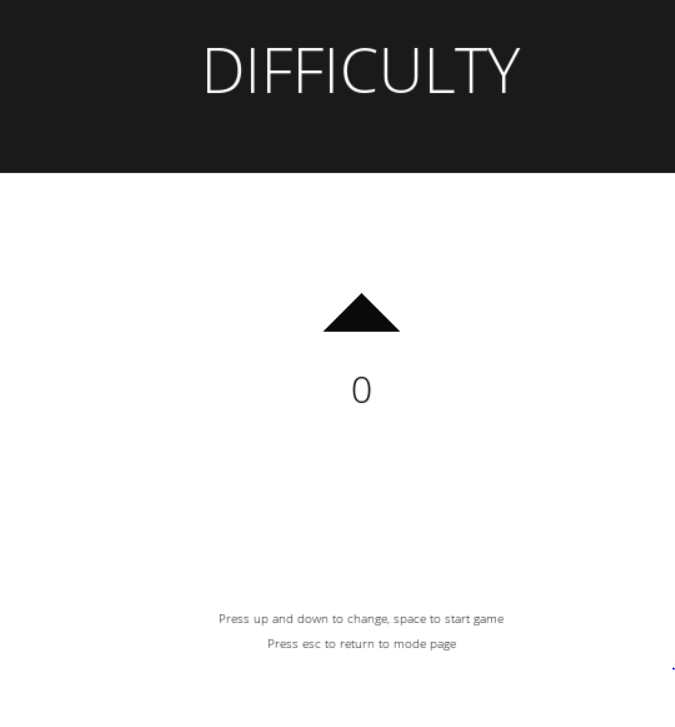
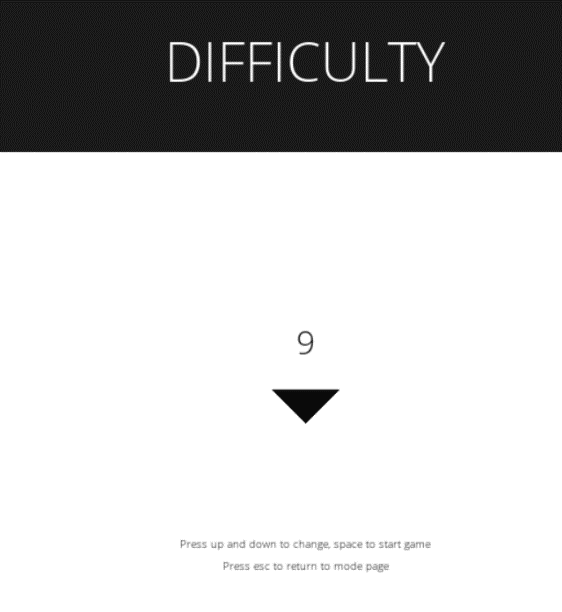
<구현된 화면1>



**모드 선택 후 게임의 초기 속도 설정 가능**

: 위, 아래 방향키를 통해 0~9까지의 범위에서 초기 속도를 조절하여 선택 가능

<구현된 화면2>

**최대 속도=9**

**최소 속도=0**

: 최소 및 최대 난이도에서는 각각 아래, 위 화살표가 표시되지 않도록 하여 유저가 난이도 조절 범위를 인지할 수 있도록 함

**4) 아이템 모드 구현**

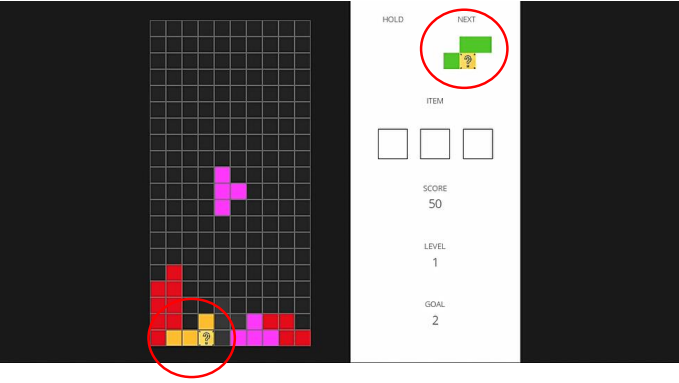
(1) 조건: 랜덤으로 발생하는 물음표 블록을 지우면 아이템을 얻을 수 있음

(2) 아이템 종류: 가로줄 삭제, 세로줄 삭제, 3X3 블록 삭제, 전체 블록 리셋, 맨 아래 줄 삭제

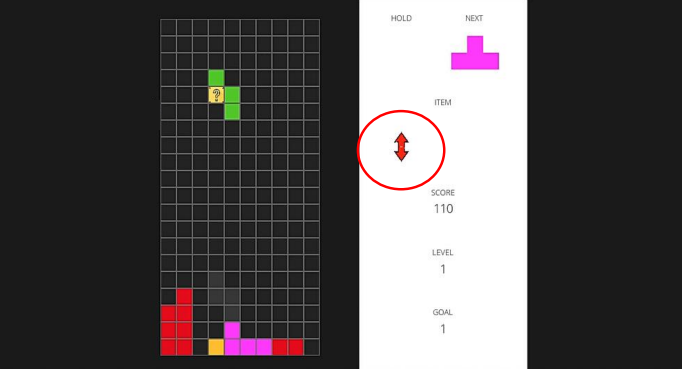
⇒ 이 중 가로줄 삭제, 세로줄 삭제, 3x3 블록 삭제의 경우 아이템 사용 키를 누르면 해당 아이템 블록이 생성돼 원하는 위치로 이동시켜 사용할 수 있음

(3) 아이템 사용키: 순서대로 Z, X, C → 인벤토리의 1, 2, 3번째 아이템

<물음표 블록 생성 화면>

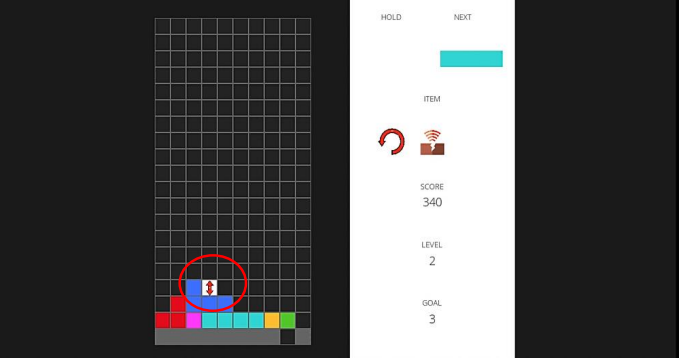


<아이템 생성 화면>



: 물음표 블록이 삭제되면 인벤토리에 아이템이 저장되고, 지워진 블록에 물음표 블록이 있다면 개수를 세고 그 개수만큼 아이템을 얻는 방식

<아이템 사용 화면>

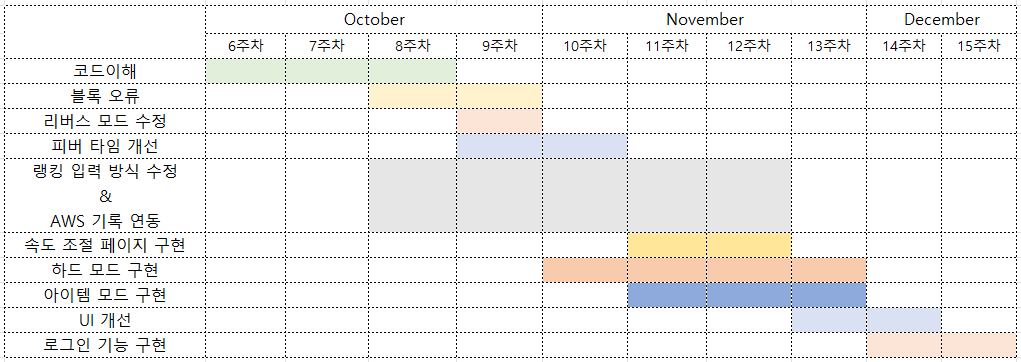


: Z, X, C 키를 눌러 원하는 아이템을 사용

**3. 기존 프로젝트와 DoitDoit 프로젝트 비교**

|  |  |  |
| --- | --- | --- |
|  | Pintris | DoitDoit |
| 파일 수 |  |  |
| 소스코드 라인 수 | 1. Pintris   * 2522라인   2. mino   * 192라인 |  |
| 클래스 | 1. Pintris  ui\_variables   * 1개   2. mino  is\_stackable\_2P   * 1개 | 1. ui  ui\_variables   * 1개   2. mino  Tetrimino   * 1개 |
| 함수 | 1. Pintris  set\_volume()  draw\_block()  draw\_board()  draw\_reverse\_board()  draw\_1Pboard()  draw\_2Pboard()  draw\_multiboard()  draw\_mino()  draw\_mino\_2P()  erase\_mino()  erase\_mino\_2P()  is\_bottom()  is\_bottom\_2P()  is\_leftedge()  is\_leftedge\_2P()  is\_rightedge()  is\_rightedge\_2P()  is\_turnable\_r()  is\_turnable\_r\_2P()  is\_turnable\_l()  is\_turnable\_l\_2P()  is\_stackable()  is\_stackable\_2P()  def init\_game()   * 24개 | 1. 파일이름  set\_volume()  draw\_block()  draw\_image()  draw\_block\_image()  draw\_board()  draw\_hardboard()  draw\_reverse\_board()  draw\_1Pboard()  draw\_2Pboard()  draw\_multiboard()  draw\_itemboard()  draw\_mino()  draw\_mino\_2P()  erase\_mino()  erase\_mino\_2P()  is\_bottom()  is\_bottom\_2P()  is\_leftedge()  is\_leftedge\_2P()  is\_rightedge()  is\_rightedge\_2P()  is\_turnable\_r()  is\_turnable\_r\_2P()  is\_turnable\_l()  is\_turnable\_l\_2P()  is\_stackable()  is\_stackable\_2P()  isthereID()  istheresaved()  LoginPass()  LoginID()  LoginCom()  DrawBar()  get\_item()  use\_item()  earthquake()  board\_reset()  erase\_row()  erase\_col()  bomb()  init\_game()   * 41개 |

**4. 프로젝트 일정**



**5. 역할 분담**

|  |  |
| --- | --- |
| 고명섭(팀장) | AWS 연동, 로그인 기능 구현, 피버 기능 수정, 리버스 모드 수정, 버그 수정, UI 수정 |
| 고다희 | 모드별 난이도 기능, 아이템 모드 구현, Soft drop, 버그 수정, UI 수정 |
| 김수빈 | 하드 모드 구현, 버그 수정, UI 수정 |