

프로젝트 제안서

팀장 경영학과 고명섭
팀원 산업시스템공학과 고다희
팀원 산업시스템공학과 김수빈

1. Outline



1.1 Why Tetris



1.2 Base Source



1.3 SWOT

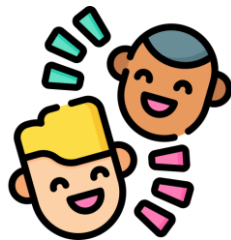


1.4 Goal

1. Outline

1.1 Tetris

- Game ?



재미를 추구

- Tetris & 기존 프로젝트

- 장) 대중성 있는 테트리스
- 단) 다소 단조로워질 수 있는 테트리스의 단점 → 적당한 난이도 설정이 중요
- 따라서, 기본적인 tool만을 가지고 있는 *기존 프로젝트의 문제점 개선*을 목표

» 오픈소스를 이용한 개발이 기존의 창작물을 기반으로 발전시킨다는 점에서 기존 tetris의 오픈소스를 비판적인 관점에서 접근하고 이를 Version Up하는 것은 본 프로젝트의 목적에 부합

1. Outline

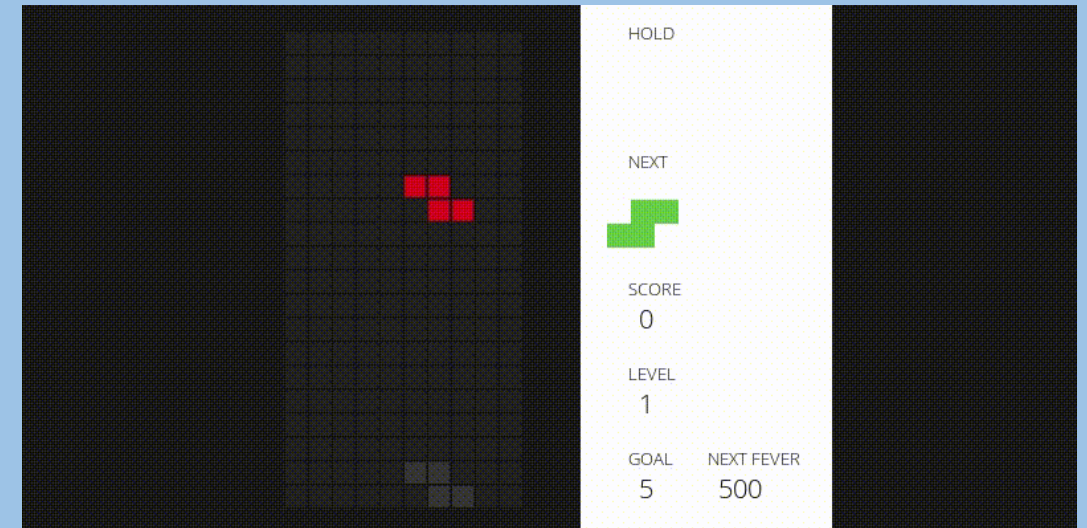
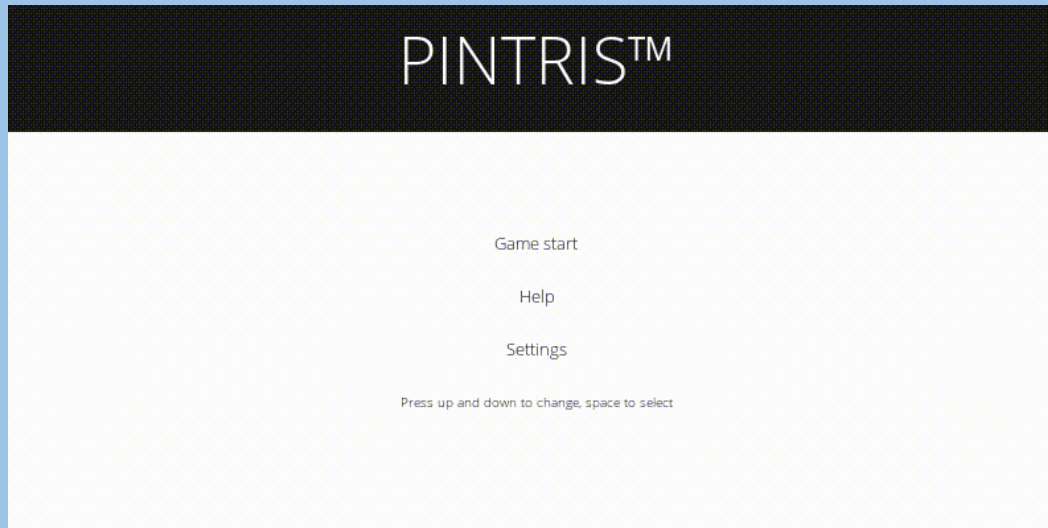
1.2 Base Source

오픈소스 이름: PINTRIS

오픈소스 URL: [CSID-DGU/2021-1-OSSPC-Pint-9 \(github.com\)](https://github.com/CSID-DGU/2021-1-OSSPC-Pint-9)

라이선스: MIT License

게임 구성: EASY, NORMAL, HARD, PvP, SPEED&MINI, REVERSE MODE



1. Outline

1.3 SWOT

Strength

- ✓ *Tetris experience* - 익숙함
- ✓ *Language* - Python

Weakness

- ✓ *Game* - 아쉬운 모듈화 기준
- ✓ 단조로운 게임 구성

Opportunity

- ✓ 단조로움 - 추가할 수 있는 요소 多
- ✓ 다양한 레퍼런스 보유 → 활용 용이

Threat

- ✓ 프로젝트 결과
➔ 사용자들의 만족도
- ✓ 고전 게임에 대한 낮은 흥미

1. Outline

1.4 Goal

- 게임의 근본적인 역할인 " 재미 "를 위한 개선을 목표로 기존 Tetris가 가지고 있는 문제점을 보완
- 기존의 모듈을 수정 + 새로운 Mode 추가
- 익숙함과 새로움이 공존하면서도 simple한 형식을 추구하고자 함



THROUGH PROJECT

- 1) 개발환경을 통한 협업에 경험이 부족한 팀원들에게 값진 경험이 될 것임
- 2) 리눅스 환경에 익숙해지고, 오픈소스에 대한 이해도와 오픈소스를 이용한 개발능력 향상을 궁극적인 목표로 설정

2. Development Environment



Language : python 3.9 이상



OS : Ubuntu



TOOL : Visual Studio Code



Collaborate : Github

3. Objectives



1 전체적인 모드 수정

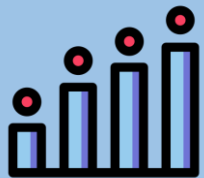
- 리버스모드 수정
- 이지모드, 스피드모드 & 미니모드 삭제
- 노멀모드에서 스피드 선택 기능 추가
- 피버타임 모드 수정
- 하드모드 수정



2 아이템모드 추가



3 블록 움직임 오류 개선



4 aws를 이용한 경쟁시스템구현



5 UI 개선

3. Objectives

- ① Mode 체계 수정 normal mode, hard mode, reverse mode, Item mode, PvP mode
- 기본+스피드 조절 기본+장애물 기본+방향키 반대 기본+아이템 사용 기본+2인

- normal mode - 스피드 조절 기능 추가

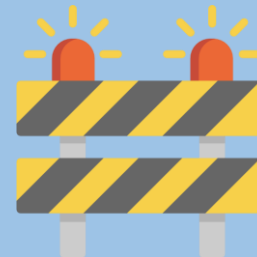


set_timer 설정, 속도 레벨 버튼 생성하기

- hard mode 변경



- 수정 전: 기존 *normal mode*에서 속도가 빠른 버전이 *hard mode*

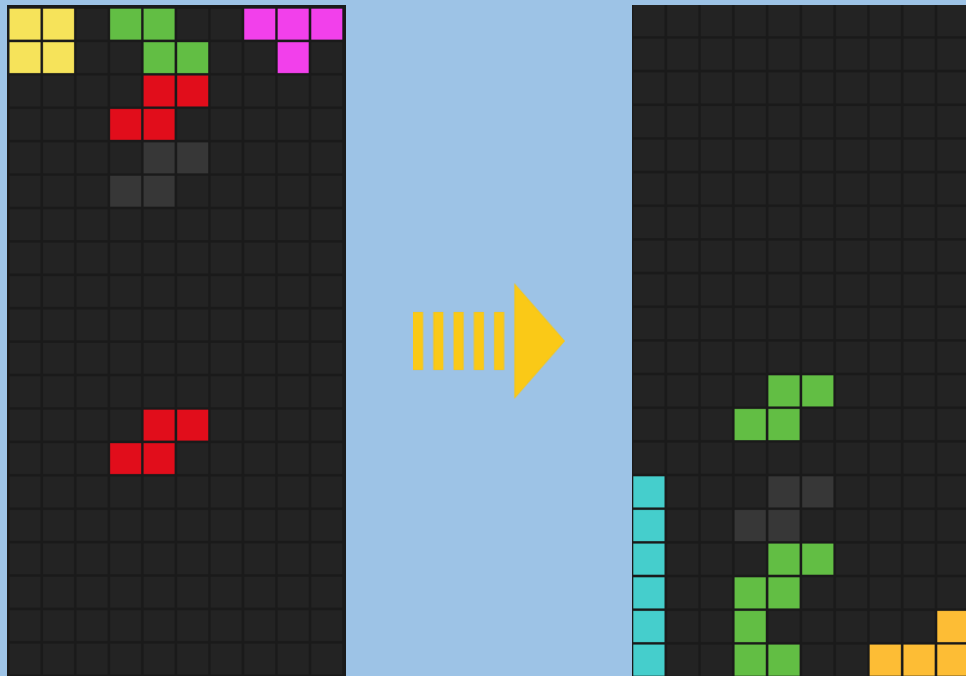


- 수정 후: 장애물이 있는 모드로 변경
→ 게임 중 랜덤으로 시야 방해물 공격

screen.blit(): 이미지를 보이게 하는 함수 사용하여 방해물 나오게 하기, PvP 모드에서 상대방 공격 기능 활용

3. Objectives

- 기존 reverse mode 수정



draw_reverse_board()함수에서 draw_block 부분 수정(방향키 설정 변경)

- 기존 피버타임 발생 조건 변경

- 피버스코어 달성 시 4*1 블록만 등장



- 수정 전: 피버타임 발동 시 무조건 500점 추가 획득 가능



- 수정 후: 정해진 점수에 도달 시 특정 피버타임 시간 획득

time 모듈을 사용하여 설정한 end time까지 4*1 도형만 나오도록 구현

3. Objectives

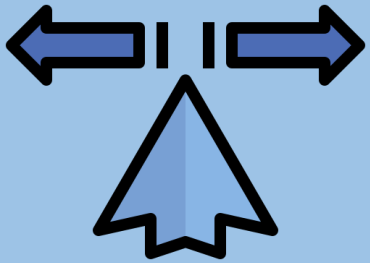
② 모드 추가 – Item Mode



- 500점 단위로 점수에 도달하면 아이템을 랜덤으로 부여
- 인벤토리에 최대 3개까지 보관 가능
: 1, 2, 3 버튼을 눌러 아이템 사용 가능
- 아이템 종류
: 맨 아래줄 삭제, 전체 블록 리셋, 현재 블록 중심으로 3*3크기 블록 삭제, 10초간 점수 2배

3. Objectives

③ 오류개선 - 블록 움직임 오류



수정 전: 블록이 바닥에 닿아도 일정 시간 동안 이동 가능함



수정 후: 블록이 바닥에 닿으면 바로 다음 블록이 생성 되도록

pygame.time.set_timer부분의 수치를 10보다 작은 수로 수정하여 블록이 생성되는 속도 빠르게 하기

3. Objectives

④ 시스템 수정 - AWS를 이용한 랭킹 연동 시스템



수정 전 : 최신 기록으로 랭킹에 남음



수정 후 : 유저별 가장 높게 나온 점수를 랭킹에 기록
(게임이 끝난 후 *aws*에서 최신 기록을 받아와
실시간은 아니지만 경쟁을 유도함)

AWS RDS상의 tetris라는 데이터베이스 생성 -> 이름 입력 -> 기록 저장 후 DB에서 fetch하여 랭킹 생성

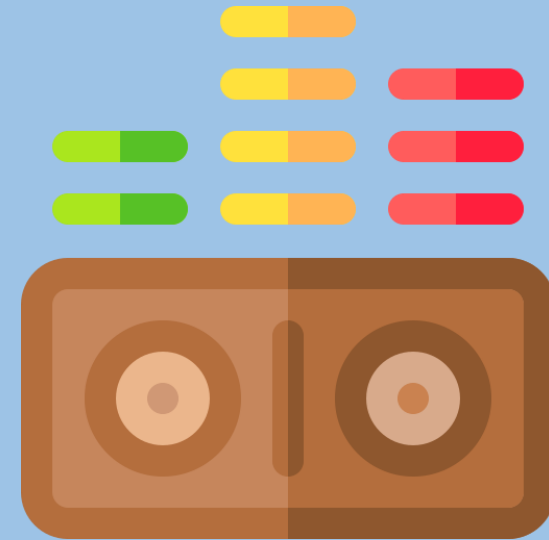
3. Objectives

⑤ UI 개선

- 글자 크기 확대



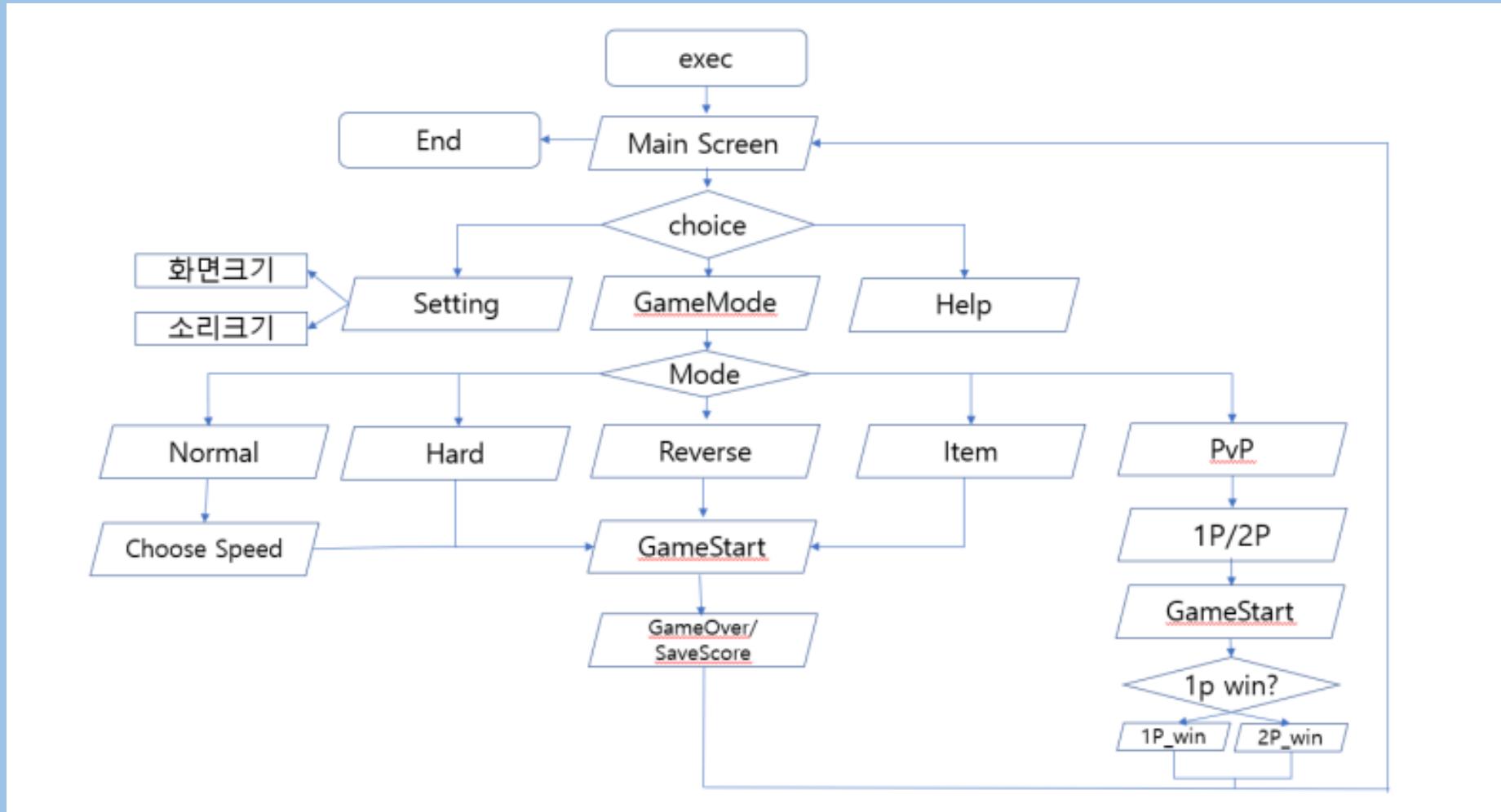
- 사운드 추가



ui_variables 클래스 수정

3. Objectives

- 예상 구조도 생성



4. Team

Role

고명섭
(팀장)

깃허브 총괄, 랭킹기능 수정, 리버스 모드 수정, 피버타임 수정, 보고서

고다희
(팀원)

아이템 모드 구현, 노말모드 난이도 조절, UI 개선, 오류개선, 보고서

김수빈
(팀원)

하드모드 구현, UI 개선, 오류 개선, 사운드 추가, PPT

4. Team

Team Rule

- Github Link : [CSID-DGU/2021-2-OSSPC-DoitDoit \(github.com\)](https://github.com/CSID-DGU/2021-2-OSSPC-DoitDoit)

- Github Rule

1) Commit Message

- 제목은 영문 기준 50자를 넘기지 않고, 마침표를 붙이지 않는다.
- 제목은 과거시제를 사용하지 않고 명령조로 작성한다.
- 제목의 첫 글자는 반드시 대문자로 쓴다.
- 제목과 본문은 한 줄 띄워 분리한다.
- 본문 내용은 선택사항, 한 줄에 72자 넘기지 않는다.
- 본문은 '어떻게'보다 '무엇을', '왜'에 맞춰 작성한다.

4. Team

Team Rule

- Github Rule

2) Branch

- 기본적으로 test 브랜치에 push를 원칙으로 한다.
- 매주 월요일 회의시간에 main으로 merge를 논의한다.

3) Merge

- 참고: <https://gmlwjd9405.github.io/2017/10/27/how-to-collaborate-on-GitHub-1.html>
- 항상 pull request를 거친 후 merge한다.
- Pull request가 있는 경우 해당 Issue를 보고 피드백을 남긴다.

4. Team

Team Rule

- 변수 및 함수 작성 규칙

- 변수명과 함수명은 자유롭게 정하되, 서로 같은 경우가 없도록 한다.
- 변수명은 항상 소문자로 시작, 다음 단어는 대문자로 구분한다.
- 함수명은 항상 대문자로 시작, 다음 단어는 대문자로 구분한다.
- 새롭게 선언한 경우 반드시 주석처리를 통해 목적을 밝힌다.

- 회의

- 카카오톡을 통해 실시간으로 논의하고, 오프라인 회의는 매주 월요일 6시에 진행한다.
- 오프라인 회의 날이 아닌 회의는 webex를 통해 진행한다.

5. Final

Expectation



- 도입부의 프로젝트 목표로 설정했던 기존 테트리스의 문제점을 해결하여 테트리스의 익숙하면서도 게임으로서의 역할을 하도록 발전시킴
- 기존 오픈소스 코드를 활용하고 협업하는 경험을 통해, 앞으로 좀 더 적극적으로 오픈소스에 참여할 수 있는 자신감을 얻음
- 팀 단위 협업하는 과정에서 소통규칙과 방법을 익힘
- 문제파악 – 해결방안 논의 – 제안 – 피드백 – 수행이란 소프트웨어개발 사이클을 경험함으로써 개발자로서 한걸음 더 나아갈 수 있는 계기

THE END

DoitDoit