

프로젝트 제안서

Open Source Software Project

2017112504 안석환

2018112451 정민주

2019113173 홍석주

목차

1. 프로젝트 개요
 - A. 프로젝트 주제
 - B. 선정 오픈소스 분석
 - i. 선정 오픈소스 정보
 - ii. SWOT 분석
 - iii. 선정이유
2. 프로젝트 목표
 - A. 개발환경 및 라이선스
 - B. 프로젝트 개선 및 개발 사항
 - i. 개선 사항
 - ii. 개발 사항
3. 예상 결과물
 - A. 프로젝트 시스템 구조
 - B. 예상 시스템 결과
 - C. 최종 결과물
 - D. 기대효과
4. 프로젝트 일정 및 규칙
 - A. 개발 역할 분담
 - B. 타임라인
 - C. Git 운영방식
 - D. 정기회의
 - E. Notion 활용방식

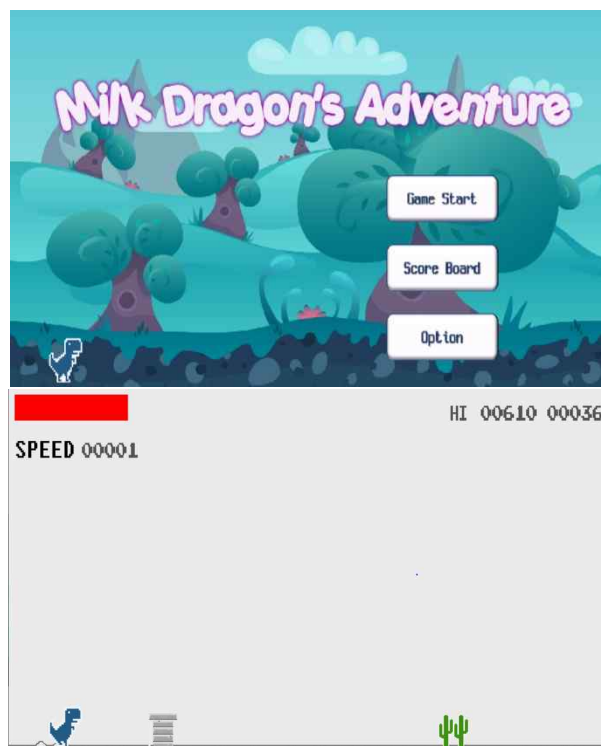
1. 프로젝트 개요

1.1 프로젝트 주제

이번 프로젝트의 목표는 오픈소스를 활용하여 'T-rex Rush' 게임을 개선하는 것이다. 이 프로젝트는 2020년 2학기 'OldKokiri' 팀에서 1차 가공하였고, 2021년 1학기 'MilkDragon' 팀에서 2차 가공하였다. 기존 OSSP 수업에서 진행한 2021-1-MilkDragon-6을 기반으로 하여, 기존 코드에서 발생하는 일부 기능적 오류와 UI 문제를 해결하고 이와 더불어 게임 진행을 수월하게 해주는 아이템을 살 수 있는 상점과 pvp 모드를 개발하여 재미요소를 더한 다채로운 T-rex Rush 게임을 개발할 것이다.

1.2 선정 오픈소스 분석

1.2.1 선정 오픈소스 정보



본 프로젝트에서 선정한 오픈소스는 2021-1-mikdragon-6팀의 'T-rex Rush'이다. Pygame을 통해 제작되었으며 게임모드는 easy mode와 hard mode 2가지

로 구성되어 있다. 부가적인 기능으로는 점수를 보여주는 Score Board와 게임에 대한 설명과 간단한 설정을 할 수 있는 option이 있다.

✓ License: MIT License

✓ 사용 언어 : python

✓ 추가 모듈 : pygame

✓ 주소(URL) : <https://github.com/CSID-DGU/2021-1-OSSPC-MilkDragon-6>

✓ 디렉토리 구조 :

1. db = 게임의 점수 및 사용자 이름을 저장하는 DB 관련 패키지를 포함하는 디렉토리

- a. __inti__.py
- b. db_interface.py = DB를 사용하는 인터페이스 모듈
- c. schema.sql = DB구조 선언 query 파일
- d. score.db = 게임 점수 정보 저장 DB 파일

2. sprites = 게임 내에서 사용되는 이미지와 배경음이 포함된 디렉토리

- a. 000.png
- b. XXX.wav ...

3. src = 게임 진행에 필요한 각 객체 및 기능을 포함한 디렉토리

- a. __inti__.py
- b. dino.py = 게임 내 공룡 캐릭터를 선언하는 파일
- c. game.py = 게임 초기화 파일
- d. interface.py = UI 표현 클래스를 선언하는 파일
- e. item.py = 아이템의 구조를 선언하는 파일
- f. obstacle.py = 장애물 및 보스를 선언하는 파일
- g. setting.py = 게임의 전반적인 설정값을 선언하는 파일

4. run.py = game.py의 intoscreen()을 호출해 게임을 시작해주는 파일

1.2.2 SWOT 분석


Strength	Weaknees
<ul style="list-style-type: none"> ✓ pygame 으로 작성됨 ✓ 게임이 낯선 사용자를 위한 일반 모드와 게임에 익숙한 사용자를 위한 하드 모드 존재 	<ul style="list-style-type: none"> ✓ 게임 내 버그 존재. ✓ 테트리스와 달리 1 인플레이 기능만 있음. ✓ UI 디자인 부족
Opportunity	Threat
<ul style="list-style-type: none"> ✓ 플레이 방식이 단순하여 누구나 쉽게 접근할 수 있는 게임 ✓ 다양한 응용 접목 가능 	<ul style="list-style-type: none"> ✓ 더 복잡하고 흥미요소가 많은 게임이 존재함 ✓ 정리 안된 코드

1.2.3 선정이유

- ✓ 팀원 모두가 사용가능한 python 언어로 이루어져 있음
- ✓ 2개의 모드(easy, hard)로만 구성되어 있어 다양한 모드를 추가할 수 있음
- ✓ 게임 내 버그 등 수정해야하는 부분을 개선하고자 하는 욕구
- ✓ MIT License가 사용되어 수정, 배포의 권한이 허용되어 있음. (다만, 저작권 표시 및 허가 표시를 SW의 모든 복제물 및 중요한 부분에 기재해야 한다.)

– *MIT License*

main ▾
2021-2-OSSPC-Coin-T-Rex / LICENSE
Go to file
...


JoungMinJu/2021-2-OSSPC-Coin-T-Rex is licensed under the MIT License
A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions

- ✓ Commercial use
- ✓ Modification
- ✓ Distribution
- ✓ Private use

Limitations

- ✗ Liability
- ✗ Warranty

Conditions

- ⓘ License and copyright notice

This is not legal advice. [Learn more about repository licenses.](#)

2. 프로젝트 목표

2.1 개발환경 및 라이선스

- ✓ OS = Ubuntu
- ✓ Code editor = VSCode
- ✓ Language = Python
- ✓ Collaboration Tool = notion, Github
- ✓ License = MIT
- ✓ References:
 - <http://www.pygame.org/docs>
 - <https://github.com/wayou/t-rex-runner>
 - <https://github.com/shivamshekhar/Chrome-T-Rex-Rush>
 - <https://github.com/CSID-DGU/2020-2-OSSP-CP-OldKokiri-6>
 - <https://github.com/CSID-DGU/2021-1-OSSPC-MilkDragon-6>

2.2 프로젝트 개선 및 개발 사항

2.2.1 개선

		개선할 점	개선 방향
1	전체	뒤로가기 버튼 부재	‘뒤로가기’ 기능 단축키 지정
2		normal 모드 단축키 설명 부재	gamerule.png 변경
3	메인화면	hard mode 버튼이 잘 작동하지 않음.	selectMode() 함수 수정
4		GameOver 시 선택권 없이 점수 입력, 랭킹판 노출, 메인화면으로 나감	game over 시 ‘restart’ 버튼과 ‘main’, ‘store’ 버튼 노출. restart 누르면 점수 입력 없이 게임 재시작.

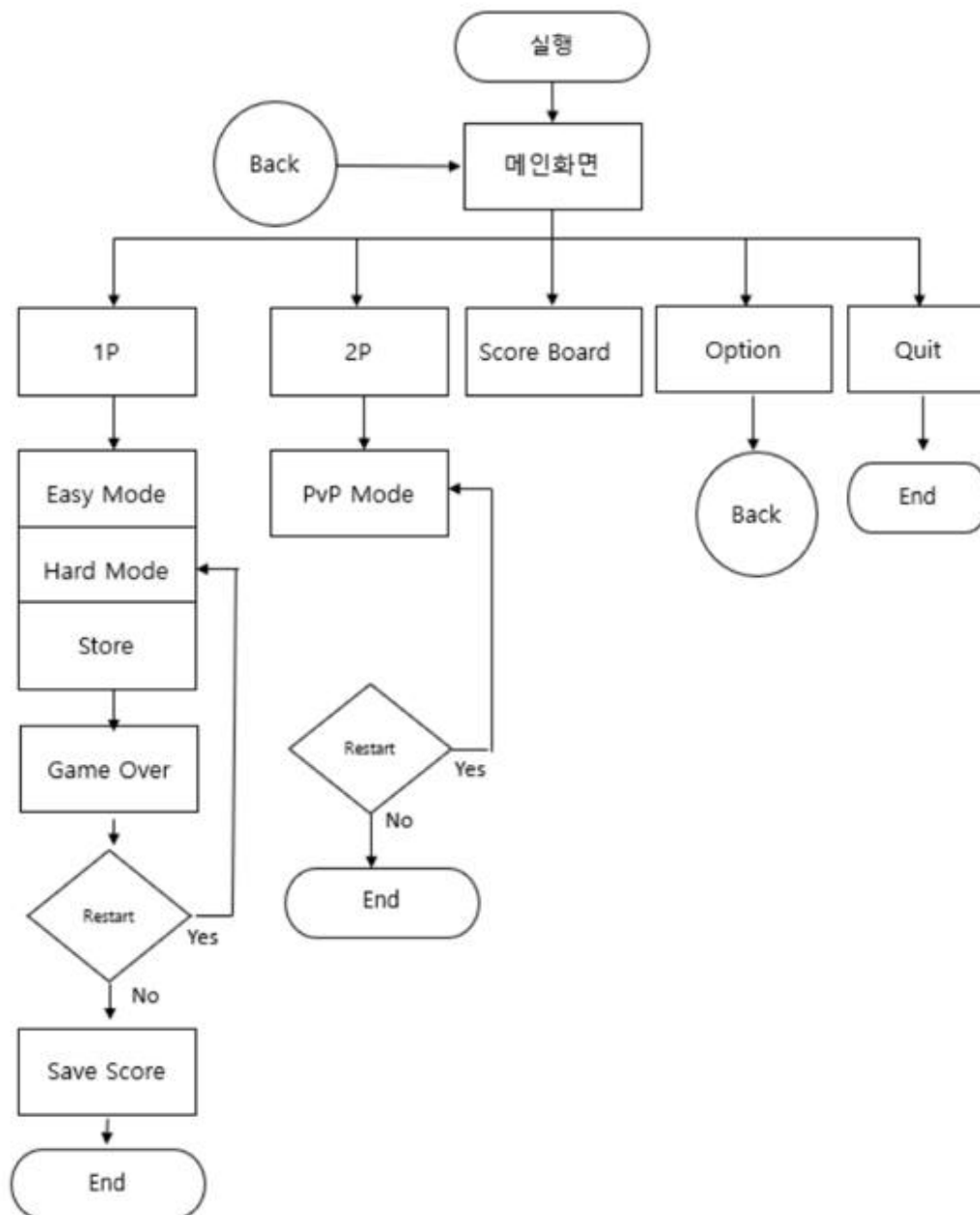
			main 누르면 점수 입력 -> 랭킹판 노출 -> 메인화면
5	랭킹판	normal 모드와 hard 모드 랭킹판의 통합	typescore 함수 수정, db 수정
6	Hard Mode	캐릭터가 화면 밖으로 나갈 수 있음 (점수와 스피드는 계속 올라감)	gameplay_hard 함수의 방향키 이동 부분에 screen_width, screen_height 를 벗어나면 안된다는 기준 추가
7		보스 공격이 잘 안 먹힘	gameplay_hard 함수 #7 라인 수정
8		보스 hp 바 표시가 안되어 있어서 공격하고자하는 의지 하락	PteraKing 클래스의 hp 속성 플레이 화면에 표시 interface.py HeartIndicator 클래스 활용
9		보스를 공격해야하는데, 스피드가 계속 올라가서 게임 진행에 차질	PteraKing 객체 플레이 화면에 등장할 때는 속도 유지하도록 코드 수정
10		보스가 한 번 밖에 나오지 않음	stage 마다 지정된 수의 보스가 등장하도록 코드 수정
11		플레이 화면에 아이템이 나오는 방식. 실질적으로 게임 도중 아이템 사용이 어려움	2.2.2 상점기능과 연결
12	코드	from 모듈 import * 형식으로서, 메소드 찾기가 어려움	python convention 에 맞게 기존 코드 수정
13		python convention 이 지켜지지 않은 코드가 많음.	#11 번과 동일

2.2.2 추가 기능 개발

	개발 내용	개발 내용
1	pvp 모드 구현	불꽃을 이용해서 서로 싸울 수 있도록 구현 or 점수가 높은 플레이어가 이기도록 구현
2	상점 기능 구현	플레이 화면에 노출되는 코인을 기반으로 상점의 아이템을 구매할 수 있도록 구현
2 -1	아이템 추가	상점에서 구매 가능한 아이템 추가

3. 예상 결과물

3.1 프로젝트 시스템 구조



3.2 예상 시스템 결과

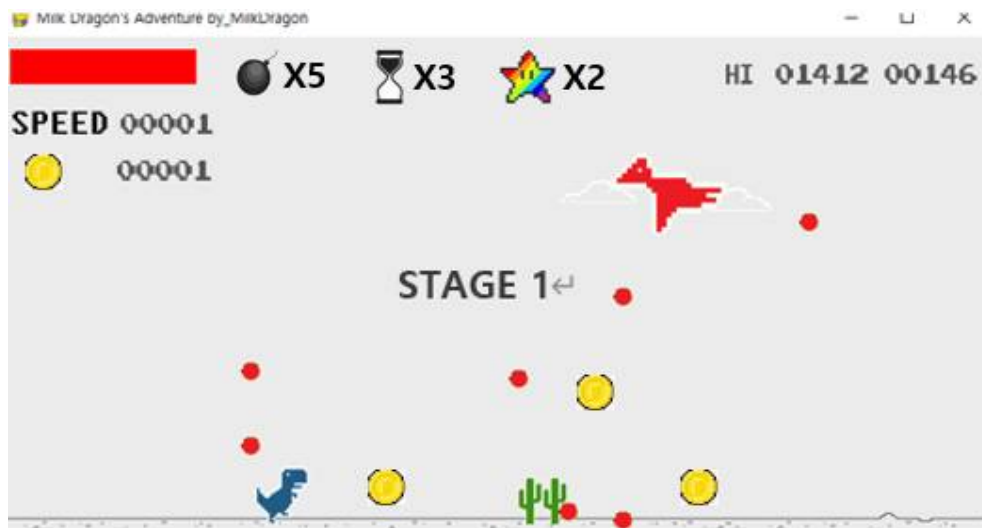
✓ 아이템 상점 개시

= 게임 내에서 얻은 코인으로 아이템을 구매하여 오락성을 향상



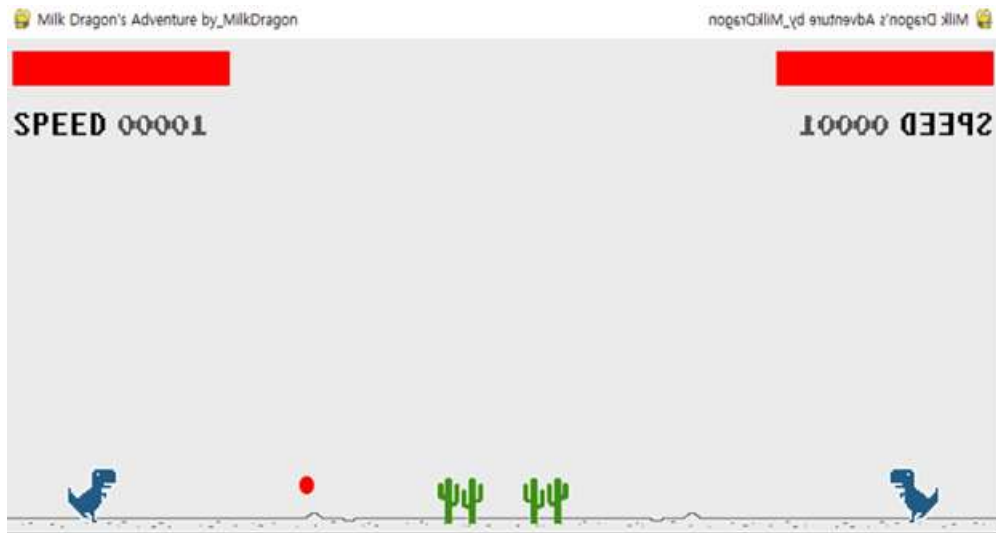
✓ Hard Mode 개선

= 스테이지 별로 보스가 등장하는 횟수를 다르게 하여 난이도 조절과 함께 게임에 흥미를 더함



✓ PVP 모드

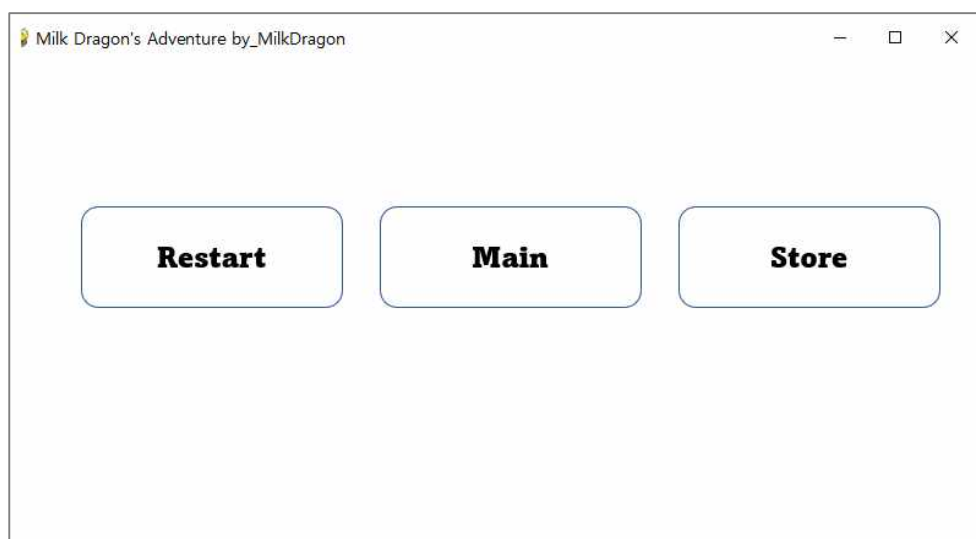
= 2인 플레이가 가능하도록 개발.



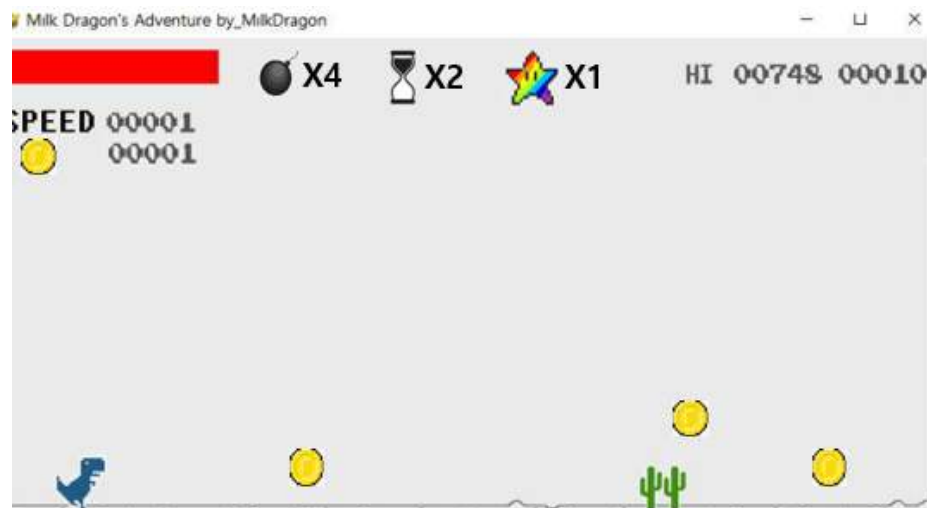
✓ Game Over시 알고리즘 흐름 변경

= 본래 Game Over시 바로 Score 입력 후, 메인화면으로 빠져나갔던 것을 Restart(점수 입력없이 재시작), Main(메인메뉴), Store(상점) 세 가지의 선택지를 제공하는 것으로 변경. 이를 통해 게임 진행이 보다 원활하게 이루어지도록 함.

(0) Game Over시 화면



(1) Restart 클릭



(2) - Store 클릭



(2) - 1 Restart 클릭



(3) main 클릭

Milk Dragon's Adventure by_MilkDragon

— □ ×

플레이어 이름을 입력해주세요

CURRENT SCORE: 92

(3)- 1

Milk Dragon's Adventure by_MilkDragon

— □ ×

RANKING

	Name	Score
TOP 1	AAA	748
TOP 2	ADF	92

(3)-3

Milk Dragon's Adventure by_MilkDragon

— □ ×

Milk Dragon's Adventure

Game Start

Score Board

Option

3.3 최종 결과물

- 제안서, PPT, 중간 보고서 및 최종 보고서
- 원본 오픈소스에서 수정 및 보완된 프로젝트
- 시연 영상
- 게임 사용법이 담긴 README 파일

3.4. 기대효과

- ✓ 오픈소스 개선 과정을 통해 이미 만들어진 코드를 이해하는 과정 학습 및 개선하는 능력 향상
- ✓ Python에 대한 이해도 증진
- ✓ Python Convention을 사용하여 코드의 가독성을 올리는 법 터득
- ✓ 협업 툴(Github & notion)을 통해 코드의 가독성 증진
- ✓ 오픈소스 라이선스 관련 지식 습득

4. 프로젝트 일정 및 규칙

4.1 개발 역할 분배

이름	내용
안석환	상점 기능 구현, 아이템 추가, 플레이화면 인터페이스 수정
정민주	pvp 구현, 상점 기능 구현 (보조), 랭킹판 분리
홍석주	코드 정리, Game Over 알고리즘 수정, 보스 스테이지 정비

4.2 타임라인

이름	역할	타임 테이블										
		11월 1주차	11월 2주차	11월 3주차	11월 4주차	11월 5주차	12월 1주차	12월 2주차				
안석환	코드 수정(보조)											
	play 인터페이스											
	상점 및 코인 기능											
	단축키 연결											
	최종 수정 및 발표											
정민주	코드 수정(보조)											
	뒤로가기 버튼											
	Option 인터페이스											
	아이템 추가											
	PVP 구현											
	진행상황 파악 및 보조											
홍석주	최종 수정 및 발표											
	코드 수정											
	보스 오류 수정											
	보스 스테이지 구현											
	랭킹판 통합											
	최종 수정 및 발표											

최종 발표

4.3 정기회의

- 매주 금요일 오후 6시, Webex로 진행.
- 코드 정리 후 본격적인 개발이 시작되면 오프라인으로 미팅 진행 예정.

4.4. Git 협업규칙

✓ PR 및 Commit 규칙

- Fork한 저장소를 각자의 local로 가져와 수정 후, add → commit → push 후 upstream에 Pull Request 수행
- 기능 단위의 branch 만들어 작업 수행.
- 코드 리뷰 후 문제가 없으면 merge한다.
- Commit 단위는 하나 이상의 의미를 가지지 않는다.
- commit 양식은 다음을 따른다. [날짜, 이름] (type) 설명

Ex) [21.10.15, 정민주] (Add) PterKing 클래스 추가

[Commit type]

	Type	설명
1	Add : (기능)	새로운 기능을 추가
2	Mod : (코드 부분)	기존 코드에서 오류 부분을 수정
3	Delete : (기능 및 코드 부분)	필요 없는 기능을 제거
4	Docs : (문서)	문서 수정

✓ 표기 규칙

- 변수는 스네이크 케이스 표기
- 클래스는 캐멀 케이스 표기

4.5 Notion 활용 방식

TODO

Not started	In progress	Completed
시간 남으면 할 것	제안서 초안 작성	주제 정하기
	합업 규칙	리포트
	제안서 제출자료 만들기	특정/동영 정리하기
	License 관련 정보 구체화	정답지/Growing T-MEX
		제안서 독자 정리
		문제집 정리
		우표 파악하기

일정

2021년 10월

일	월	화	수	목	금	토	일
26	27	28	29	30	10월 1일		
3	4	5	6	7	8	9	10월 1일
10	11	12	13	14	15	16	17

- 각자의 타임라인에 맞추어 작업 내용 ToDo에 기재
- 참고 자료들은 notion에 업로드하여 공유
- 회의날짜 및 일정 관리는 캘린더 활용
- 공유가 필요한 내용만 기재 (노션 용량 제한에 기반한 내용)