

SHOOTING GAME 프로젝트

최종보고서

8조 렛츠기린

전자전기공학부 2017111924 정태호
산업시스템공학과 2019112466 문지윤
산업시스템공학과 2019112476 하지수

목차

1. 프로젝트 소개

- 1) 기본 정보
- 2) 코드 비교

2. 게임 요소

- 1) UI
- 2) 인게임
- 3) 데이터베이스

3. 팀 프로젝트 체계

- 1) 프로젝트 일정
- 2) 깃 협업규칙
- 3) 역할 분담
- 4) 활동 로그

1. 프로젝트 소개

1. 기본 정보

기존 프로젝트

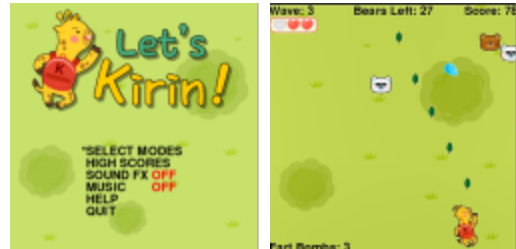
게임 컨셉



- 우주에서 외계인을 격추하는 우주선의 모험
<https://github.com/jpritcha3-14/shooting-game>

개선된 프로젝트

게임 컨셉



- 들판에서 곰들을 물리치는 기린이의 모험
https://github.com/CSID-DGU/2021-2-OSSProj-Lets_Kirin-8

게임 장르 shooting game

라이선스 MIT

사용 언어 및 모듈 python, pygame

2. 코드 비교

	베이스코드	렛츠기린	설명
main (shooting_game.py / Lets-kyrin.py)	461	110	게임을 실행하는 메인 코드
menu.py	-	646	메뉴 화면 코드
mode_single.py	-	690	싱글 모드 코드
mode_time.py	-	684	타임 모드 코드
mode_pvp.py	-	728	pvp 모드 코드
load.py	39	39	게임 내의 이미지와 음악을 로드하는 함수를 선언하는 코드
sprites.py	292	441	게임 내에서 사용되는 객체를 정의하는 코드
database.py	61	151	(기존) 하이스코어를 저장하는 hiScore.db를 생성하고, db에서 인게임으로 점수를 불러오는 코드 (추가) 유저 아이디, 비밀번호 및 모드별 점수 저장 코드
TOTAL	853	3,489	-

2. 게임 요소

1. UI

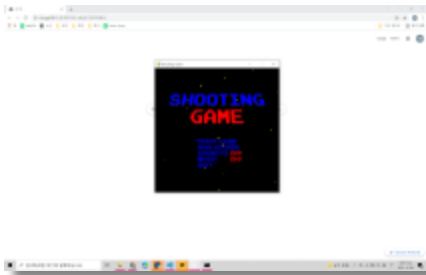
기존 프로젝트

메뉴 구성



창크기

창크기 조절이 되지 않음



로그인/회원가입

- 로그인, 회원가입 기능 존재 x
- sqlite 데이터베이스를 이용해 개인의 하이스코어만 저장 가능

PAUSE

- 게임 중 일시정지 기능 존재 x
- 게임을 한번 실행하면 게임이 끝날 때까지 나갈 수 없음

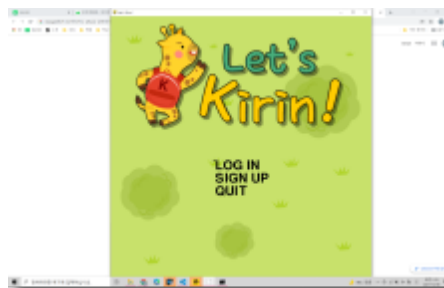
개선 된 프로젝트

메뉴 구성



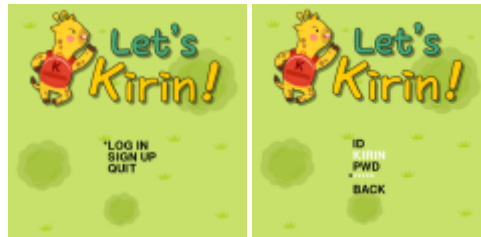
- 로그인 / 메인 / 모드 선택 등 목적에 따라 메뉴 구성을 추가
- 게임 컨셉에 맞게 이미지, 폰트 색깔 변화

창크기



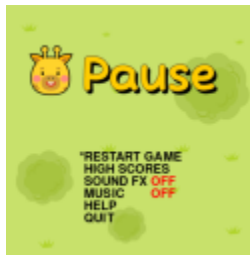
- 마우스를 이용해서 창크기 조절 가능

로그인/회원가입



- 로그인/ 회원가입 화면, 기능 추가
- AWS RDS 데이터베이스를 이용해 아이디, 비밀번호 저장
- bcrypt 모듈을 사용해서 아이디를 저장하고 hashing, salt를 이용하여 비밀번호 안전하게 저장

PAUSE



- 게임 중 키보드 'p'를 누르면 일시정지
- 다시 'p'를 누르면 게임 시작
- RESTART를 누르면 처음부터 시작

2. 인게임

기존 프로젝트

장애물

5종류의 장애물 존재
장애물별 획득 점수 동일

아이템

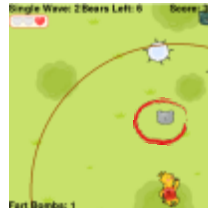
2종류 아이템 존재
(폭탄, 쉴드)

개선 된 프로젝트

획득 점수 조정

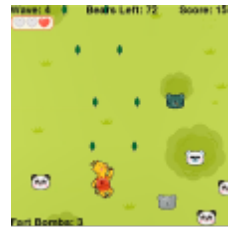


새로운 장애물 추가



→ 장애물별 획득 점수 조정
→ 죽지 않는 장애물 추가

더블 미사일 아이템



라이프 아이템



→ 아이템이 드랍되게 함
1) 일정시간 동안 미사일 두 줄기 사용 가능
2) 일정시간 동안 헬퍼가 유저를 따라다니며 도움
3) 남은 목숨 1증가

싱글모드

한 명의 유저가 살아남을 때 까지 게임을 하고 획득한 점수로 랭킹을 기록하는 모드

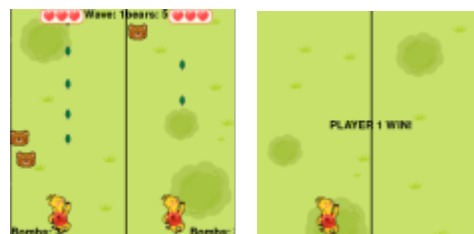


타임모드 추가



→ 제한시간 60초 동안 획득한 점수로 랭킹을 기록하는 모드 구현
→ 3초의 준비 시간
→ 기존 싱글모드에 비해 난이도를 상향 조정
→ 기존 싱글모드에 비해 아이템 출현 빈도수를 높임

PVP모드 추가



→ 두 명의 플레이어가 경쟁하는 PVP 모드 구현
→ 플레이어1은 w/a/s/d, 플레이어2는 화살표키 사용
→ 화면 분할에 따른 객체의 이동 범위 제한
→ 객체별 아이템, 목숨 등을 분리하여 적용
→ 오래 살아남는 플레이어가 승

3. 기타

기존 프로젝트

HELP

→ 따로 HELP 페이지가 없어서 게임을 하면서 조작법을 익혀야 함

LIFE

→ 기본 목숨 1개로 한번 죽으면 게임이 끝남

배경 플로팅

→ 검은 화면 위 별들이 내려오면서 배경이 흐르는 느낌을 줌

사운드

→ 우주 배경에 맞는 배경 음악

개선 된 프로젝트

HELP



→ 게임 요소가 다양해짐에 따라 간단한 게임 설명이 담긴 Help 페이지 구현
→ 메인 메뉴와 게임 중 pause 메뉴에서 HELP 확인 가능

LIFE



→ 기본 목숨 3개로 수정
→ 아이템으로도 출현

배경 플로팅 & 사운드

→ 풀숲 이미지가 위에서 아래로 흐르는 느낌 추가
→ 게임 컨셉에 맞게 배경 음악 수정

4. 데이터베이스

기존 프로젝트

데이터베이스

→ sqlite 데이터베이스를 이용해 개인의 하이스코어만 저장 가능
→ 기기마다 점수 기록이 다름

개선 된 프로젝트

분산 관계형 데이터베이스 구축



→ AWS RDS(MariaDB) 구축, 연동
→ 기기 간의 동일한 하이스코어 공유 가능
→ 스코어, 유저 정보 저장 가능

모드 별 랭킹 기록 구분

→ 모드 존재 x
→ score 테이블에 게임 기록 모두 저장

모드 별 랭킹 기록 구분

NAME	SCORE	ACCURACY
GEO	525	0.3463
A	167	0.5156
Z1900	161	0.4215
B	159	0.4173
TAEHO	150	0.4610
LSAI	142	0.6762
Z1900	84	0.3397
D	81	0.3389
D	79	0.3004
JLE	65	0.3009
AAA	50	0.5743
D	31	0.2582
E	2	0.0

NAME	SCORE	ACCURACY
TAEHOA	236	0.5742
JUL	192	0.4
GRUNE	161	0.5612
WHAT	161	0.4613
TAEHOT	133	0.6425
TAEHO	110	0.6644
GOOPAL	96	1.2695
PALPAL	80	0.4706
TIMEEST	72	1.5319
TAEHOB	72	1.3583
TAEHOTES	61	1.0702
TEST	44	1.4687
WHY	18	3.0
GUIN	16	0.6164
AHO	16	1.3333

→ 모드가 다양해짐에 따라 모드 별 점수 테이블 추가
→ 모드 별 하이스코어 목록 확인 가능

3. 팀 프로젝트 체계

1. 팀 프로젝트 일정

1) 진행 계획

정태호	
문지윤	
하지수	
팀원전체	

	9주차(1027~1101)	10주차(1103~1108)	11주차(1110~1115)	12주차(1117~1122)	13주차(1124~1129)	14주차(1201~1207)
더블슈팅 기능 추가						
일시정지 기능						
메인 메뉴 화면 개선						
기본 폭숨 추가						
게임 배경 플로팅 기능 추가						
기존 장애물 개선 및 새로운 장애물 추가						
AWS EC2 서버 구축 및 데이터베이스 연동						
회원가입/로그인 기능 구현						
장애물 별 획득 점수 추가						
창 크기 조절						
무기 아이템 개선 및 추가						
타임모드						
업적 기능(아이템 및 ui) 추가						
PVP 모드						
랭킹 기능 구현						
사운드 수정 및 추가						
전체적 ui 수정						
전체적 버그 수정 및 마무리						

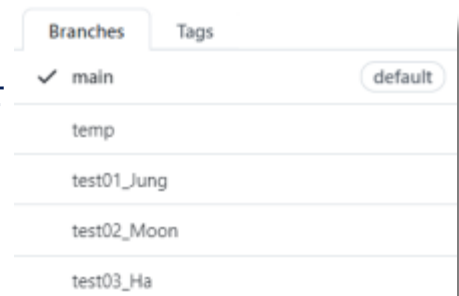
2) 진행 결과

	9주차(1027~1101)	10주차(1103~1108)	11주차(1110~1115)	12주차(1117~1122)	13주차(1124~1129)	14주차(1201~1207)
일시정지 기능 추가						
메인 메뉴 항목 추가						
더블슈팅 기능 추가						
기본폭숨 추가						
다시시작 기능 추가						
AWS RDS 구축 및 연동						
기존 장애물 개선 및 새로운 장애물 추가						
플로팅 배경이미지 추가						
로그인/회원가입 ui 추가						
더블슈팅 아이템 추가						
헬퍼 아이템 추가						
TIME모드 추가						
로그인/회원가입 기능 추가 및 서버 구현						
PVP모드 추가						
목숨 아이템 추가						
TIME모드 개선						
모드별 랭킹기록 저장 및 ui 구현						
헬퍼 아이템 개선						
창크기조절 - 배경 및 폰트						
help 메뉴 추가						
사운드 수정						
pvp모드 개선						
창크기조절 - 객체						
전체적 버그 수정 및 마무리						

- 10주차 이후 일부 계획 수정
- 창크기조절 기능은 마지막에 구현하는 것이 효율적이라고 판단
- 필요한 기능을 구현하는 몇 일정 추가
- 팀원전체 작업으로 예정되어있던 일을 효율을 위해 분담함

2. 깃 협업규칙

1. 팀 저장소를 fork한 개인 저장소에서 팀원들이 각자 작업을 한 뒤, 팀 저장소의 '각 팀원 branch'로 pull request
2. 팀장이 확인 후 merge하고, 'temp branch'에서 팀원들의 작업물을 병합
3. 'temp branch'에서 병합한 이후에 실행에 이상이 없으면 'main branch'에서 해당 작업 내용을 pull
4. 해당 주차의 작업이 모두 끝나면 팀 저장소의 개인 branch로 해당 내용을 동기화
5. 매주 주말에는 팀 저장소의 'main branch'의 내용을 개인 저장소의 'main branch'에 업데이트하고, '새로운 branch'에서 다음 작업을 진행



3. 역할분담



정태호

→ 일시정지 기능 추가, 기본목숨 추가, 다시시작 기능 추가, 인게임 배경 개선, TIME모드 추가, 목숨 아이템 추가, 창크기조절 기능 추가(배경 및 폰트 적용)



문지윤

→ 메인 메뉴 항목 추가, AWS RDS 구축 및 연동, 로그인/회원가입 UI 및 기능 추가 (서버구현), 모드별 랭킹기록 저장 및 UI 구현, help 메뉴 추가, 배경음악 수정

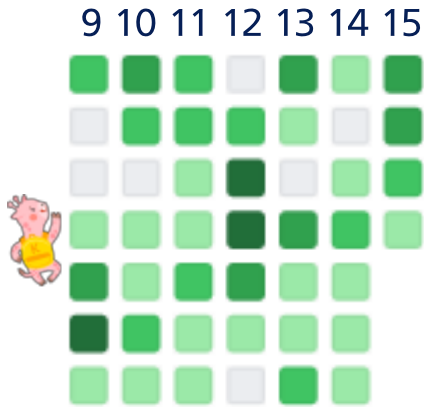


하지수

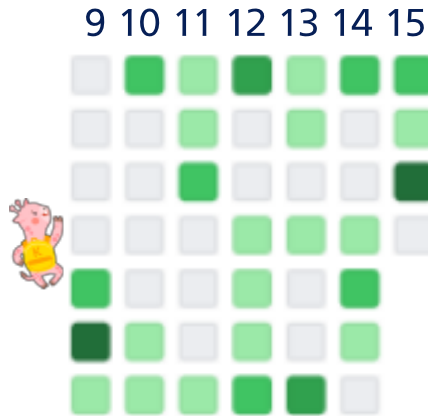
더블슈팅 아이템 추가, 기존 장애물 개선 및 새로운 장애물 추가, 헬퍼 아이템 추가, pvp모드 추가, 창크기조절 기능 추가(객체 적용)

4. 활동 로그

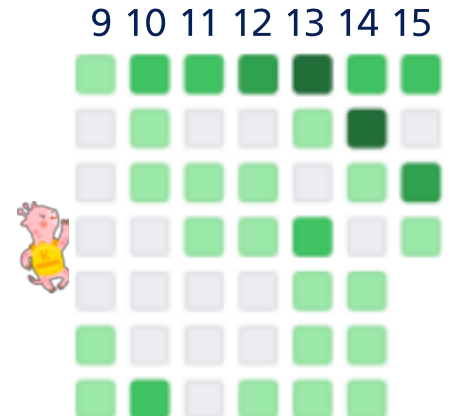
정태호



문지윤



하지수



5. 총평



정태호

→ 단순히 오픈소스를 활용해서 새로운 프로젝트를 해볼 수 있다는 점이 매력적으로 다가와서 수강했던 과목이었지만, 이 과목의 이론 수업을 통해서 가상환경, Linux, git & github, docker, aws 등 생각했던 것 이상의 지식에 대해 공부할 수 있어서 보람있었다. 특히 프로젝트를 통해서 팀원들과 협력하여 맡은 업무를 해내고, 집단지성을 통해 높은 관문을 넘어서며, 나아가 소프트웨어 분야와 더 친해질 수 있는 계기가 되어 정말 값진 경험이 되었다.



문지윤

→ linux, docker, aws 등의 소프트웨어 이론, 실습뿐 아니라 프로젝트 제안서, 일정 관리, 구성, 회의록, 피드백을 통해 팀 프로세스의 유용한 방향을 익힐 수 있었다. 특히 git과 github을 이용한 프로젝트는 깔끔한 포트폴리오 정리와 더 효율적인 진행이 가능함을 배웠다. 이번 수업에서 쌓은 팀 협업 경험은 앞으로 있을 프로젝트에서도 도움이 되리라 확신한다.



하지수

→ 오픈소스 사용의 중요성을 배웠고, 높은 활용성을 직접 경험하였다. 오픈소스를 우리의 프로젝트에 활용하기 위해서 타인이 작성한 코드를 분석하는 경험과 git을 통해 협업하는 경험이 처음인만큼 많이 힘들었지만 이번 프로젝트를 통해 프로그래밍 실력과 협업 능력을 향상시킬 수 있었다. 멋진 팀원들과 꾸준히 소통하며 각자가 맡은 업무는 책임감 있게 수행하고, 협업 규칙을 지키기 위해 다함께 많이 노력하였기 때문에 만족스러운 결과물을 얻을 수 있었다고 생각한다.