

# **OSSP 제안서**

**8조 3인분**

**2021-1 OSSPC 3PEOPLE**

**2015110510 통계학과 김태환**

**2015112583 산업시스템공학과 박재민**

**2018111711 의생명공학과 배유진**

# 1. 기존 프로젝트 소개

파이썬으로 만들어진 간단한 슈팅게임이다. sprite.py, shooting\_game.py, load.py, database.py 4 개의 소스코드로 이루어져 있다.



그림 [1] 메뉴와 실행 화면

A(좌측 이동), D(우측 이동)로 ship 을 조작하고, Space 를 눌러 미사일을 발사해 무작위로 내려오는 Alien 을 격추할 수 있다. Bomb 과 Shield 아이템이 일정 시간마다 드랍 되며, Bomb 은 시작 시 기본적으로 3 개 주어진다.

Bomb	Shield
	
Ship 을 기준으로 원을 그리며 적을 격추함. b 키로 사용할 수 있다.	적과 충돌해도 게임이 종료되지 않음. 적과 1 회 충돌하면 shield 는 사라짐.

표 [1] 아이템

Alien 과 1 회 충돌하면 게임이 종료된다. 하이스코어를 달성하면 이름을 기록하라는 창이 뜨고, 이름과 점수, accuracy 가 database.py 에 저장된다. Accuracy 는 점수를 미사일이 발사된 횟수로 나눈 값이다. 점수는 Alien 의 색과 관계없이 모두 1 점으로 동일하다.

일정 숫자의 Alien 을 격추하면 다음 레벨(wave)로 올라갈 수 있다. 레벨마다 추가되는 alien 의 종류는 다음과 같다.





Wave 1	Wave2	Wave3	Wave4
siney(green)&spikey(blue)	Fasty(white)	Roundy(red)	Crawly(yellow)
			

표 [2] Alien

각 Alien 들은 random 과 math 함수로 구성된 서로 다른 패턴을 가지고 있다. 화면의 Aliens Left 는 다음 wave 로 가기 위해서 격추해야 하는 남은 alien 의 숫자를 나타낸다. 레벨이 올라갈 때 마다 다음레벨로 가기 위해 격추해야 하는 alien 의 수는 2 배가 되고, 속도가 0.5 빨라진다.

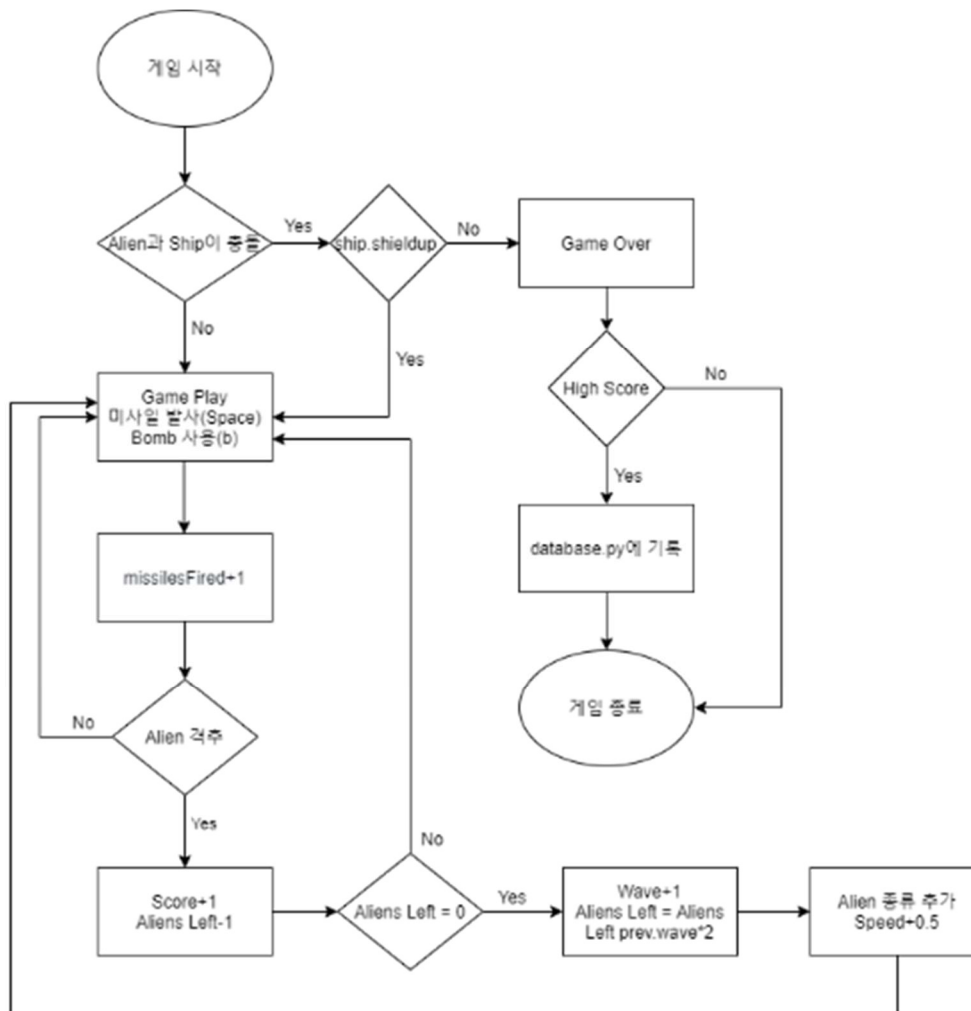


그림 [2] 프로세스 다이어그램



### 그림 [3] MIT 라이선스

본 프로젝트는 MIT라이선스를 따른다. 상업적 이용, 수정, 배포, 사적인 이용이 가능하고 저자는 소프트웨어에 관해서 아무런 책임을 지지 않는다. 단, 저작권 표시를 소프트웨어의 모든 복제물에 기재해야 한다.

## 2. 기존 프로젝트 장단점

장점	단점
<ul style="list-style-type: none"> <li>- 간결한 코드, 4 개의 코드가 기능별로 나누어져 있어 코드를 수정하기 용이하다.</li> <li>- 다양한 Alien 의 움직임 패턴이 있어 긴장감 있게 플레이 할 수 있다.</li> <li>- wave 시스템이 잘 구현 되어있다.</li> </ul>	<ul style="list-style-type: none"> <li>- Alien 이 화면 밖으로 나가거나, 미사일로 격추할 수 없는 위치로 이동하는 문제점이 있다.</li> <li>- 중복 아이디로 저장이 가능하다.</li> <li>- 색과 관계없이 Alien 의 점수가 동일하다.</li> </ul>

표 [3] 기존 코드의 장단점

## 3. 라이브러리 분석

### - pygame

License : LGPL 2.1

- GPL License 에서 저작권 규칙을 완화한 버전으로, 라이브러리를 직접 변경하지 않고 단순히 사용하는 경우 큰 제약 없이 사용할 수 있다.
- 이 라이브러리를 이용하여 만든 프로그램을 상업적으로 이용하거나 특허 신청을 할 수 있으며, 다른 라이브러리와 함께 사용할 수 있고, 소스 코드의 공개 의무가 없다. 단, LGPL License 인 코드(pygame)를 사용했음을 명시해야 한다.

## 주요 기능

- pygame 은 게임 제작에 필요할 만한 기능들을 모아놓은 라이브러리이다. 그렇기 때문에 게임에 필요한 출력(화면, 음악 등), 입력(키보드, 마우스, 조이스틱 등), 구조물 객체, 이벤트 객체, 이미지 등에 관한 모듈들을 포함하고 있다.
- 화면 출력은 구조물의 위치가 조금씩 변하는 이미지를 빠르게 넘겨서, 구조물이 움직이는 것처럼 보이게 하는 방식을 사용한다. 따라서 2D 게임을 만드는 경우, 움직임 대부분을 수학적함수를 이용해서 구현할 수 있다. 프로젝트 게임에 있는 Siney 라는 외계인은 좌우로 왔다 갔다 하면서 내려오는데, 실제로 sin 함수로 구현되어있다. 그 외에도 중력은 중력가속도를 계산하여 구현하는 등 다양한 움직임을 구현할 수 있다.
- pygame 자체가 3D 기능을 제공하지는 않지만, 3D 도 라이브러리를 이용하는 등의 방법으로 이미지 출력값을 계산하면 같은 방식으로 출력할 수 있다.
- pygame 을 이용한 게임은 일반적으로 아래와 같은 구조를 갖는다.

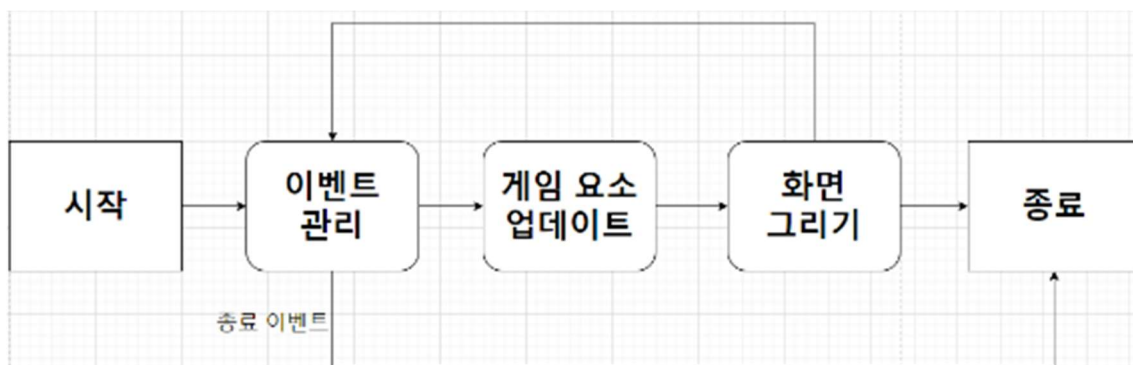


그림 [4] pygame을 이용한 게임 구조

## 오픈소스에 사용된 주요 모듈들

- sprite : 비행기, 미사일, 적 등의 속성, 위치, 속도, 충돌 상태 등의 정보를 효율적으로 관리할 수 있다.
- sprite.Group : 다양한 sprite 들을 관리하기 위해 Group 을 사용할 수 있다. 객체가 어떤 그룹에 있는지를 통해 어떤 상태인지 파악하거나 그룹 객체들의 정보를 함께 업데이트하는 용도로 사용할 수 있다.
- K\_(입력 키) : 각 입력문자에 대응하는 함수가 있어서 입력한 키를 알 수 있다.
- event : 입력장치를 통한 입력이나 콘솔 종료 등의 이벤트가 있는지 확인할 수 있다.

- rect : 기본적으로는 사각형을 만들 수 있는 함수이다. 또한, rect 와 관련된 좌표값을 불러올 수 있으므로 객체를 생성하는 경우, 객체 범위를 지정해서 함수 조건을 짜는 경우 등에서 유용하게 사용할 수 있다.
- display : 화면 크기, 업데이트 등을 하기 위해 사용된다.
- mixer : 배경음악, 효과음을 출력하기 위해 사용된다.

## - sqlite

License : Public Domain, BSD License

- C 로 개발되었고, 제작자가 저작권을 포기하고 배포하여 Public Domain 라이선스를 갖는다. 따라서 어떠한 의무사항 없이 자유롭게 사용할 수 있다.
- python 에서는 sqlite 를 이용할 수 있도록 표준 라이브러리로 지원하고 있고, python 은 BSD License 를 따른다.
- BSD License 또한 특별한 제약 없이 자유롭게 사용할 수 있다.

## 주요 기능

- 임베디드 SQL 데이터베이스 엔진으로, 서버의 연결 없이 파일을 직접 읽고 쓴다. 또한, 서버 관련 모듈을 지원하지 않고 표준 SQL 문법을 따른다.
- 작고 빠른 DBMS 로, 간단한 데이터베이스가 필요할 때 유리하다.

## 오픈소스에 사용된 주요 모듈들

- connect : DB 파일과 연결한다.
- execute : SQL 문을 이용하여 DB 를 처리할 수 있다.
- fetchall : 해당 행의 리스트를 불러온다.
- commit : 변경을 저장한다.

## 4. 개선 부분

### - UI 부분

1. 배경이 검은색에 파란 글씨 변경
2. 마우스 커서도 사용할 수 있게 변경

### - User 설정

1. 사용자 설정으로 방향키 설정하게 변경
2. 랭킹에 아이디 중복저장 안되게 변경
3. Pause 구현

### - 비행기

1. 방패, 총알 2 배 같은 여러 아이템 추가

### - 적

1. 적들도 공격을 하게 변경

### - 레벨(단계)

1. 맞으면 Life 감소
2. 일정시간동안 난이도 상승 (적들이 더 많이 생성되거나 적이 미사일을 발사함)

### - 그 외 추가 기능

1. 랭킹에 아이디 중복으로 저장 안되게 만들기
2. 동국대 마스코드 '아코'를 이용한 캐릭터 커스터마이징 고려

## 5. 실현 가능성

### 1. 키설정변경 (wasd 에서 화살표키로)

- pygame.event 값을 기존에 설정해둔 값이 아니라 pygame.K\_LEFT, pygame.K\_RIGHT 등으로 변경하면 가능

### 2. 사용자 설정으로 방향키 설정하게 만들기

- 사용자가 입력할 수 있는 UI 를 만들고 사용자 입력에 맞춰 키를 바꾸면 가능 할 것

### 3. 맞으면 죽는게 아니라 Life 감소

- sprite.py 의 ship 클래스에 life 라는 변수를 추가하고 pygame.sprite.collide\_rect 부분의 if else 문에서 if 문에 life 감소로 로직으로 바꾸면 가능

### 4. 레벨에 따라 공격을 해오는 적이 등장

- 비행기의 미사일처럼 적에게도 미사일 클래스 생성하면 가능

### 5. 일정 시간 총알을 2 배로 늘려주는 아이템 등 아이템 추가

- sprite.py 의 powerup 클래스에 함수 추가하면 가능

### 6. 일정시간동안 난이도 상승(적들이 더 많이 생성됨)

- 적들이 생성되는 주기인 alienPeriod 를 수정하여 빨리 나오게 하고, 현재 적들의 수인 aliensLeftThisWave 를 수정하여 구현 가능

### 7. 랭킹에 아이디 중복저장 안되게 변경

- database.py 의 setscore 함수의 쿼리문을 수정하면 중복이름 저장이 안되고 중복이름 이었을 때 try except 문으로 중복 오류 안내 구현 가능 예상

### 8. 레벨이 증가하면 적의 속도가 빨라짐

- alien sprite 객체에서 유닛 이동 값을 수정하여 구현 가능

### 9. 마우스 커서 사용 가능

- pygame.mouse 함수 사용 및 클릭 범위 지정, 클릭 시 동작 구현하여 가능



## 10. Pause 버튼 만들기

- state 라는 변수와 running, pause 변수를 만들어서 pygame.event.get() for 문에 p 키를 누르면 pause, s 키를 누르면 running 할 수 있는 로직 구현 가능

## 6. 예상결과 및 기대성과

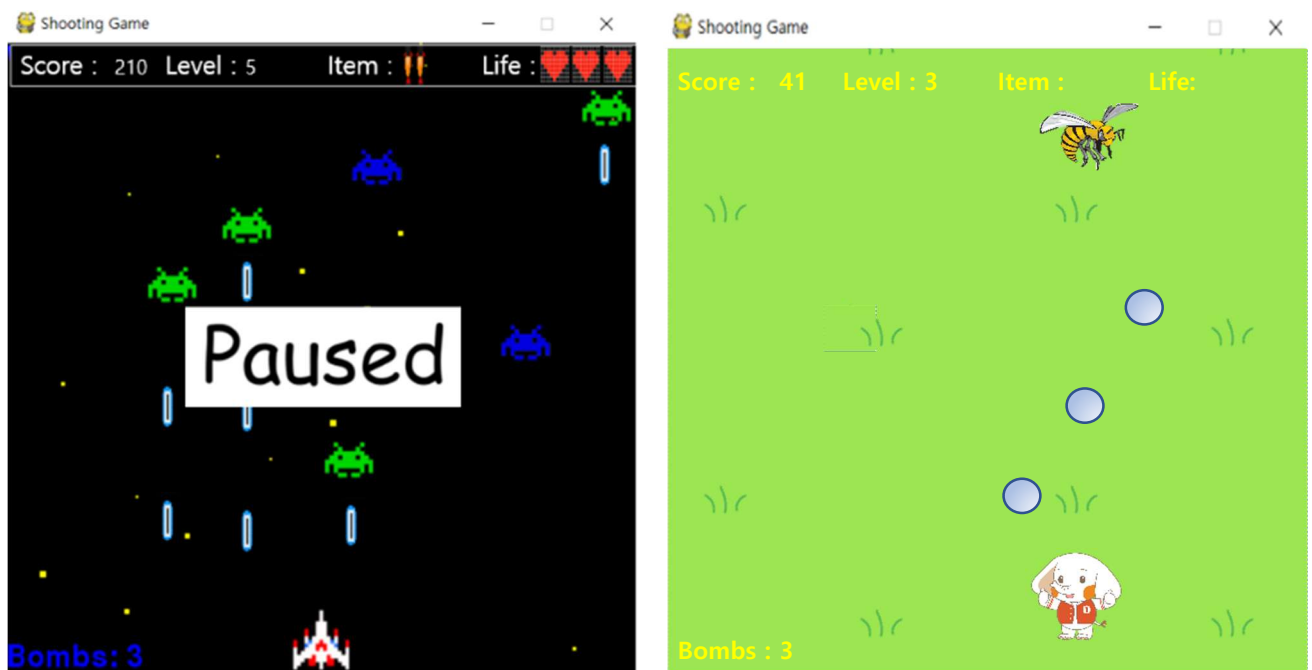


그림 [5] 프로젝트 예상 결과물

### 1. 게임성 증대

- 선택 가능한 난이도를 만들고, 게임 중간에도 난이도가 변화(적이 빨라지거나 미사일 발사)하여 게임의 긴장감을 끌어올림
- 레벨, 아이템, 미사일, 라이프시스템 등을 추가 도입하여 즐길 수 있는 게임 요소 증대

### 2. 사용자 경험 개선

- 메뉴 선택에서 마우스 사용 가능하도록 변경, 익숙한 화살표키로 변경하여 익숙한 환경 제공
- 폰트 디자인 등을 명시적으로 변경
- pause 키 추가, 랭킹 이름 중복 방지 등의 세부적인 기능 추가
- 추후 개발 진행 상황에 따라 전체 이미지를 동국대 마스코트 '아코'를 기반으로 한 UI 고려

## 7. 프로젝트 일정

	1주차	2주차	3주차	4주차	5주차	6주차	7주차	8주차
⇒ 메인 코드 분석 및 발표 준비	■							
⇒ UI		■						
⇒ User 설정			■	■				
⇒ 게임 모드			■	■				
⇒ 비행기, 적 객체					■			
⇒ 레벨(단계)					■	■		
⇒ 부가적 기능						■		
⇒ 기능 마무리 및 최종 발표준비							■	■

그림 [6] 프로젝트 일정 계획

- 분리 구현 가능한 기능들을 정리하고, 먼저 만들어야 하는 기능부터 순차적으로 구현
- 주기적인 진행 상황 공유 및 git 을 이용한 코드 관리

## 8. 참고 자료

- \* <메인 코드> : <https://github.com/jpritcha3-14/shooting-game>
- \* GPL 라이선스 : [https://ko.wikipedia.org/wiki/MIT\\_%ED%97%88%EA%B0%80%EC%84%9C](https://ko.wikipedia.org/wiki/MIT_%ED%97%88%EA%B0%80%EC%84%9C)
- \* Pygame 라이선스 : <https://www.pygame.org/docs/LGPL.txt>
- \* sqlite 라이선스 : <https://www.sqlite.org/copyright.html>
- \* python sqlite 라이선스 : <https://docs.python.org/3/license.html>
- \* pause 구현 : <https://stackoverflow.com/questions/30744237/how-to-create-a-pause-button-in-pygame>
- \* 키보드 화살표로 비행기 움직이기 구현 : <https://kkamikoon.tistory.com/132>
- \* 적공격구현:  
<https://m.blog.naver.com/PostView.nhn?blogId=samsjang&logNo=220710524226&proxyReferer=https:%2F%2Fwww.google.com%2F>
- \* 적과 충돌 시 life 감소 구현: <https://blog.dalso.org/language/python/14188>
- \* 마우스사용 구현 : [https://www.pygame.org/docs/ref/mouse.html#pygame.mouse.set\\_visible](https://www.pygame.org/docs/ref/mouse.html#pygame.mouse.set_visible)