

회의록 - OSSProj LetsGitIt

- 15주차 회의록

회의일시	2022년 12월 8일, 12월 10일	작성자	송영철
참석자	정재혁, 김재홍, 송영철		
회의장소	대면(8일), Webex(10일)		

회의안건	1. 12월 1주차 개발 사항 공유 및 완료사항 보고 2. 12월 2주차 계획
------	--

내용

1. 11월 5주차 개발 사항 공유 및 완료사항 보고

분류	목록	담당	타임테이블				
			11월 3주차	11월 4주차	11월 5주차	12월 1주차	12월 2주차
오류 수정	코드 수정	모두			완료		
	Option 인터페이스	송영철			완료		
	Play 인터페이스	정재혁			완료		
	보스 오류 수정	김재홍			완료		
	뒤로가기 버튼 생성	정재혁			완료		
기능 추가	hard 모드 coin 관리 수정	김재홍			완료		
	High-Score 관리	송영철			완료		
	PVP모드 Battle 개선(resize)	송영철 (정)			완료		
		정재혁 (부)			완료		
	PVP모드 Running 구현	김재홍 (정)			완료		
		송영철 (부)			완료		
	아이템 연결 (Running Mode)	정재혁 (정)			완료		
		김재홍 (부)			완료		
	단축키 연결	김재홍 (부)			완료		
		송영철 (정)			완료		
	화면 리사이즈 수정	정재혁			완료		
	최종 수정 및 발표	모두			완료		

✓ [정재혁]

(1) 코드 정리 및 최적화:

: 각종 키 역할 수정

: 상수화 진행

```
#obstacle.py
DEFAULT_SPEED = 4
MOVING_WIDTH = 0.5
SPEED_UP_DOWN = 2

#intro
INTRO_BTN_X = 150
INTRO_BTN_Y = 50
OPTION_BTN = 80
BACK_BTN_X = 75
BACK_BTN_Y = 30

INTRO_RESIZE_X = 0.8
INTRO_RESIZE_Y = 0.35
INTRO_RESIZE_INTERVER = 0.75

BACK_BTN_SIZE = 0.025
BACK_BTN_RECT = 0.055
SINGLE_BTN_SIZE_X = 0.66
MULTI_BTN_SIZE_X = 0.33
SELECT_BTN_H = 0.25

STORE_ITEM_Y = 0.37
STORE_ITEM_X = 0.35
STORE_ITEM2_Y = 0.08
STORE_ITEM2_X = 0.64

#배치
(life_rect.centerx, life_rect.centery) = (width * (STORE_ITEM_X), height * STORE_ITEM_Y)
(coin2_rect.centerx, coin2_rect.centery) = (width * (STORE_ITEM_X-0.02), height * (STORE_ITEM_Y + item_
life_price_rect = life_price.get_rect(center=(width * (STORE_ITEM_X+0.03), height * (STORE_ITEM_Y + item_
(buy_btn2_rect.centerx, buy_btn2_rect.centery) = (width * (STORE_ITEM_X), height * (STORE_ITEM_Y + item_
(no_money2_rect.centerx, no_money2_rect.centery) = (width * (STORE_ITEM_X), height * (STORE_ITEM_Y + ite

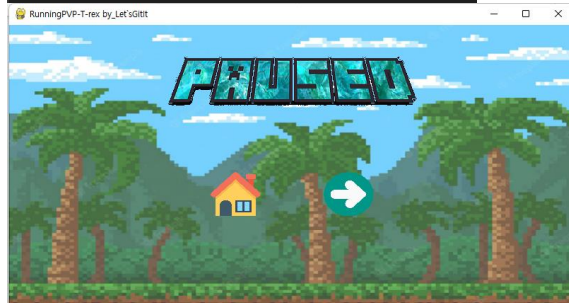
(time_rect.centerx, time_rect.centery) = (width * (STORE_ITEM_X + btn_offset), height * STORE_ITEM_Y)
(coin3_rect.centerx, coin3_rect.centery) = (width * (STORE_ITEM_X -0.02 + btn_offset), height * (STORE_I
time_price_rect = time_price.get_rect(center=(width * (STORE_ITEM_X + 0.03 + btn_offset), height * (STOR
(buy_btn3_rect.centerx, buy_btn3_rect.centery) = (
width * (STORE_ITEM_X + btn_offset), height * (STORE_IT
(no_money3_rect.centerx, no_money3_rect.centery) = (
width * (STORE_ITEM_X + btn_offset), height * (STORE_IT
#
user_count_offset = 0.04
user_btn_offset = 0.09
(user_life_rect.centerx, user_life_rect.centery) = (width * (STORE_ITEM2_X + user_btn_offset), height *
user_l_rect = user_life.get_rect(center=(width * (STORE_ITEM2_X + user_btn_offset + user_count_offset),
(user_time_rect.centerx, user_time_rect.centery) = (width * (STORE_ITEM2_X + 2 * user_btn_offset), heigh
user_t_rect = user_time.get_rect(center=(width * (STORE_ITEM2_X + 2 * user_btn_offset) + user_count_off
(user_coin_rect.centerx, user_coin_rect.centery) = (width * (STORE_ITEM2_X + 3 * user_btn_offset), heigh
user_c_rect = user_coin.get_rect(center=(width * (STORE_ITEM2_X + (3 * user_btn_offset) + user_count_off
```

: 게임 내 존재하는 각종 에러 해결

pausint()함수를 통해 게임을 종료하고 home을 이동한 경우 다시 esc 버튼을 누르면 기존 진행하던 게임의 pausing()모드로 화면이 전환되는 오류 해결

intro_screen().함수 내에서 ESC키가 눌렸을 때 아래 코드와 같이 다시 인트로 화면을 호출하도록 설정

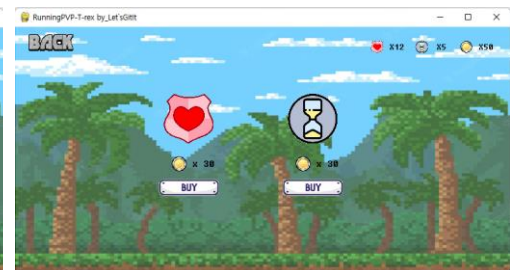
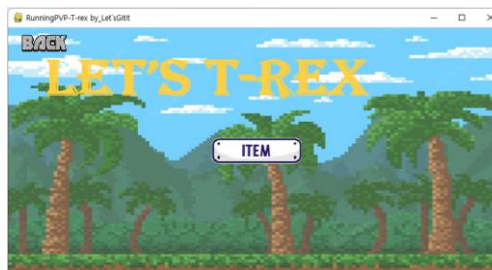
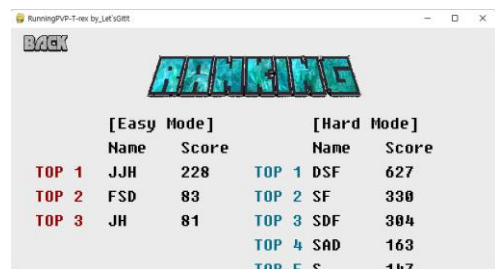
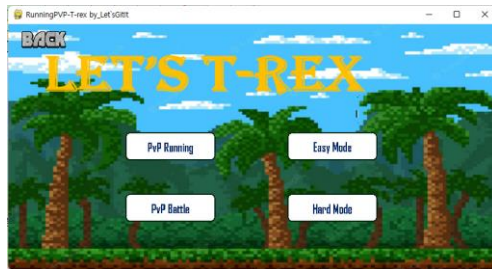
```
if event.type == pygame.KEYDOWN:
    if event.key == pygame.K_ESCAPE:
        game_quit = True
        intro_screen()
```



(2) 게임 내 UI 개선:

: 버튼 사이즈 및 위치 통일

- 화면마다 사이즈가 다르고 위치가 다르게 있던 BACK버튼 위치와 크기를 상수화를 통해 지정

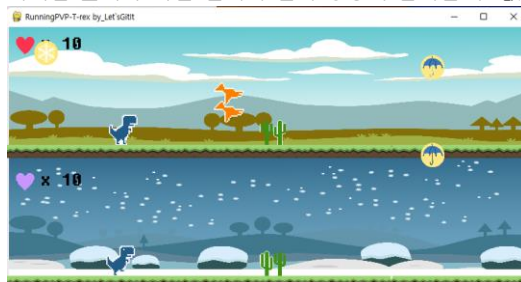


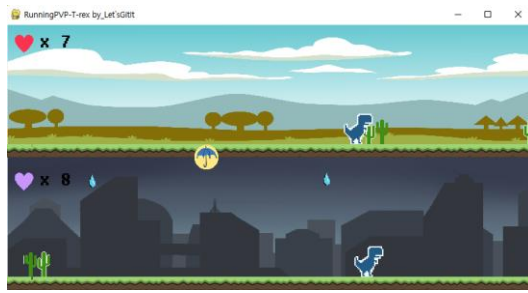
: PVP_Running 모드에서 아이템 분리해서 등장하도록 수정

- 기존에는 아이템들을 하나의 클래스로 생성해 화면 내 높이 관계없이 등장하도록 구현돼 있는데 이를 두개의 클래스로 분할해 각 공룡별로 아이템 등장 시기와 높낮이를 다르게 구현.

<기존>

아이템 높이가 화면 전체에 걸쳐 공룡이 습득할 수 없는 곳에 위치하는 오류 존재





<현재>

각 공룡에 대한 아이템을 클래스를 분할하여 따로 생성하고 높이를 조절하여 등장하도록 설정



<코드>

클래스 분리하고 아이템 별로 다르게 그룹화 하여 따로 등장하도록 구현

```
> class Dust_item(pygame.sprite.Sprite):...
You, 6시간 전 | 2 authors (MealWithoutSoup and others)
> class Dust_item_pvp(pygame.sprite.Sprite): ...
You, 6시간 전 | 2 authors (MealWithoutSoup and others)
> class Water_item(pygame.sprite.Sprite):...
You, 6시간 전 | 1 author (You)
> class Water_item_pvp(pygame.sprite.Sprite):...
You, 6시간 전 | 2 authors (You and others)
> class Ice_item(pygame.sprite.Sprite): ...
You, 6시간 전 | 1 author (You)
> class Ice_item_pvp(pygame.sprite.Sprite): ...
You, 6시간 전 | 2 authors (You and others)
> class Potion_item(pygame.sprite.Sprite): ...
You, 6시간 전 | 2 authors (You and others)
> class Potion_item_pvp(pygame.sprite.Sprite): ...

if len(dust_item) < 2:
    for l in last_obstacle:
        if l.rect.right < OBJECT_REFRESH_LINE and random.randrange(DUST_INTENSITY) < 1:
            last_obstacle.empty()
            last_obstacle.add(Dust_item(RUN_GAME_SPEED, object_size[0], object_size[1]))

if len(water_item) < 2:
    for l in last_obstacle:
        if l.rect.right < OBJECT_REFRESH_LINE and random.randrange(WATER_INTENSITY) < 1:
            last_obstacle.empty()
            last_obstacle.add(Water_item(RUN_GAME_SPEED, object_size[0], object_size[1]))

if len(dust_item2) < 2:
    for l in last_obstacle:
        if l.rect.right < OBJECT_REFRESH_LINE and random.randrange(DUST_INTENSITY) < 1:
            last_obstacle.empty()
            last_obstacle.add(Dust_item_pvp(RUN_GAME_SPEED, object_size[0], object_size[1]))

if len(water_item2) < 2:
    for l in last_obstacle:
        if l.rect.right < OBJECT_REFRESH_LINE and random.randrange(WATER_INTENSITY) < 1:
            last_obstacle.empty()
            last_obstacle.add(Water_item_pvp(RUN_GAME_SPEED, object_size[0], object_size[1]))
```

각 공룡에 따라 아이템이 등장하는 높이 범위를 다르게 설정

```
if type == 1:
    self.rect.bottom = random.choice([int(0.45*height), int(height*0.35)])
elif type == 2:
    self.rect.bottom = random.choice([int(0.95*height), int(height*0.85)])
```

(3) 최종 보고서 초안 작성

: 프로젝트에 대한 정리 및 최종 보고서에 들어갈 내용 정리 및 보고서 작성

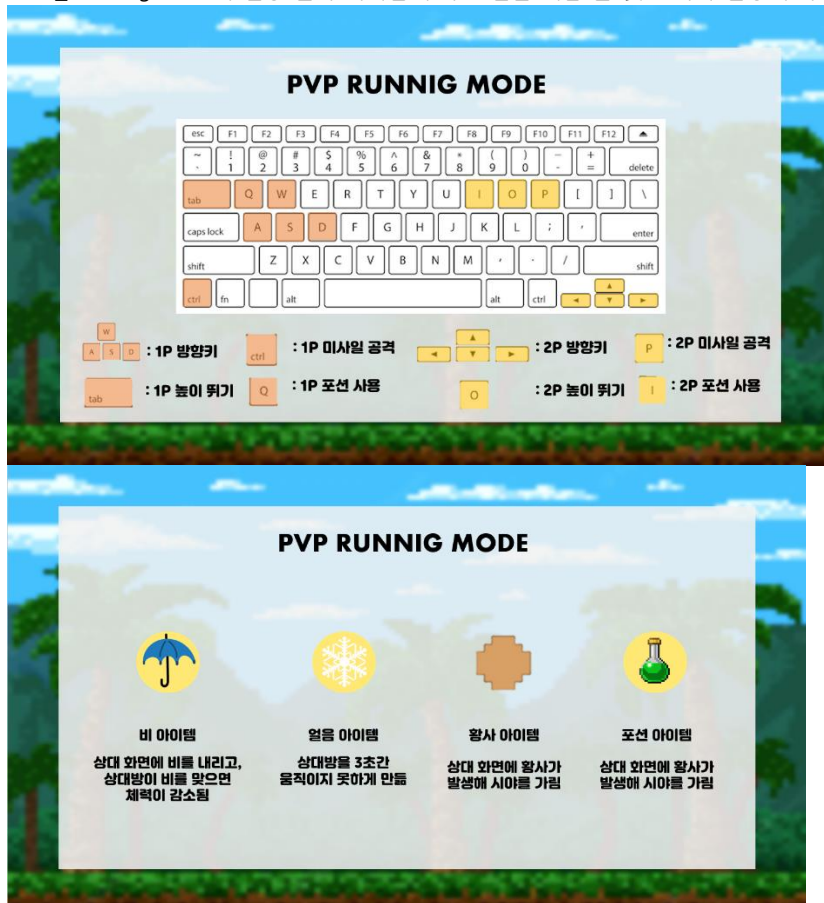
목차

- 1 프로젝트 개요
 - 1.1 프로젝트 목표
 - 1.2 선정 오픈소스 분석
- 2 프로젝트 개발 환경
 - 2.1 개발 환경 및 라이선스
 - 2.2 협업 툴
 - 2.2.1 Github
 - 2.2.2 Slack
 - 2.2.3 Notion
- 3 개선사항
 - 3.1 코드
 - 3.2 게임 내 오류 수정
 - 3.3 상점 기능 개선
 - 3.4 UI 개선
 - 3.5 Hard mode 개선
- 4 추가 기능 구현
 - 4.1 화면 Resize 기능
 - 4.2 하드 모드 내 아이템
 - 4.3 PVP_Running 모드
- 5 UI 변경
- 6 제안서 및 기존 오픈소스와 비교
- 7 배운 점

✓ [송영철]

(1) Game rules 페이지 변경사항 추가

: PVP_Running 모드 내 만능 물약 아이템 추가로 인한 게임 룰 및 조작키 설명 추가



(2) Readme 파일 작성

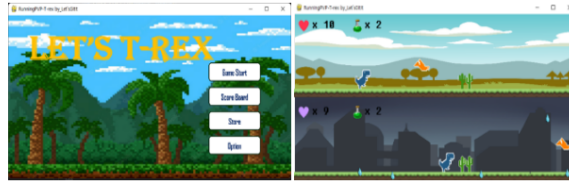
: 오픈소스 코드 출처 및 라이선스 명시. 개발사항 및 개선 사항 정리. 팀원 정보 등을 정리하여 파일 작성

license MIT pygame 2.7.5 OS ubuntu IDE VSCode

2022-2-OSSProj-Let-sGitIt-6

팀원 : 정재혁 송영철 김재홍

프로젝트 제목 : 오픈소스를 활용한 T-Rex Rush 게임 개발:



프로젝트 내용 :

Chrome 이스터에그 게임 T-rex Rush에 다양한 모션, 공격기능, PVP BATTLE모드, PVP RUNNING모드 설정 기능을 추가하여 흥미를 더한 게임

Major Change

- Intro Screen 디자인 변경 및 GUI 버그 수정
- PVP battle모드 추가 및 개선
- PVP running모드 개발
- 상점 기능 추가
- 죽음 시스템을 count로 변경
- HARD모드 아이템 및 오류 수정
- 리사이즈 개선
- 신기록 페이지 추가

SETTING TO PLAY GAME

Python version 3.7
Git

(3) 최종 발표 자료 초안 작성

최종 보고서를 기반으로 발표 내용 정리 및 발표 역할 정리



(4) 코드 정리 및 상수화 (gamerule 관련 파일 정리)

```
def gamerule():
    global resized_screen
    done = False
    btnpush_interval = 500

    largeText = pygame.font.Font('freesansbold.ttf', 60)
    TextSurf, TextRect = load_image('control_word.png', 350, 75, -1)

    TextRect.center = (width * 0.5, height * 0.2)

    # 버튼 이미지
    #easy mode button
    easymoderule_btn_image, easymoderule_btn_rect = load_image('easy_mode.png', INTRO_BTN_X, INTRO_BTN_Y, -1)
    r_easymoderule_btn_image, r_easymoderule_btn_rect = load_image(*resize('easy_mode.png', INTRO_BTN_X, INTRO_BTN_Y, -1))
    # hardmode button
    btn_hardmoderule, btn_hardmoderule_rect = load_image('hard_mode.png', INTRO_BTN_X, INTRO_BTN_Y, -1)
    r_btn_hardmoderule, r_btn_hardmoderule_rect = load_image(*resize('hard_mode.png', INTRO_BTN_X, INTRO_BTN_Y, -1))
    # runningmode button, 임시로 hardmode 이미지로 진행
    btn_runningmoderule, btn_runningmoderule_rect = load_image('pvp_running.png', INTRO_BTN_X, INTRO_BTN_Y, -1)
    r_btn_runningmoderule, r_btn_runningmoderule_rect = load_image(*resize('pvp_running.png', INTRO_BTN_X, INTRO_BTN_Y, -1))
    # battlemode button, 임시로 hardmode 이미지로 진행
    btn_battlemoderule, btn_battlemoderule_rect = load_image('pvp_battle.png', INTRO_BTN_X, INTRO_BTN_Y, -1)
    r_btn_battlemoderule, r_btn_battlemoderule_rect = load_image(*resize('pvp_battle.png', INTRO_BTN_X, INTRO_BTN_Y, -1))
    #back 버튼
    btn_home, btn_home_rect = load_image('back.png', BACK_BTN_X, BACK_BTN_Y, -1)
    r_btn_home, r_btn_home_rect = load_image(*resize('back.png', BACK_BTN_X, BACK_BTN_Y, -1))

    # 배경 이미지, 일단 임시로 사진으로 대체
    Background, Background_rect = load_image('no_name_background.png', width, height)

    # 이지, 하드모드 버튼
    easymoderule_btn_rect.center = (width * SINGLE_BTN_SIZE_X, height * SELECT_BTN_H*2)
    btn_hardmoderule_rect.center = (width * SINGLE_BTN_SIZE_X, height * SELECT_BTN_H*3)
    # 러닝, 배틀모드 버튼
    btn_runningmoderule_rect.center = (width * MULTI_BTN_SIZE_X, height * SELECT_BTN_H*2)
    btn_battlemoderule_rect.center = (width * MULTI_BTN_SIZE_X, height * SELECT_BTN_H*3)
    btn_home_rect = (width * BACK_BTN_SIZE, height * BACK_BTN_SIZE)
```

✓ [김재홍]

(1) PvP 러닝모드 만능 물약 아이템 추가

: 기존 방해 아이템을 무력화시키는 아이템을 추가.

- 만능 물약 아이템 설명

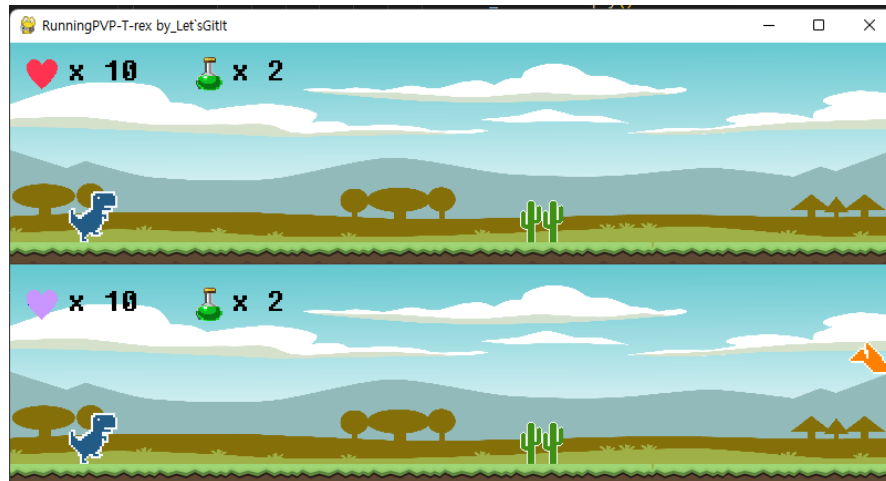
게임 시작 시 플레이어마다 2개씩 가지고 게임을 시작

아이템 개수는 게임 내 인터페이스에 배치하여 사용 가능한 개수 표시

일반 기타 아이템과 동일하게 게임 진행 중에도 등장하도록 하여 추가로 습득 가능하도록 구현

1P => q 버튼을 통해 사용, 2P => i 버튼을 통해 사용

<아이템 추가한 인터페이스>



<코드>

: 방해 아이템을 통해 방해가 시작됐을 경우에만 사용 가능하도록 구현

```
if event.type == pygame.KEYDOWN:
    # 1p dino
    if (is_dust_time_2p or is_ice_time or is_water_time):
        if event.key == pygame.K_q: # 물약 아이템
            if potion_item_count > 0: # 물약 개수가 1개 이상 일 때만 작동
                potion_item_count -= 1
                water_rest_time = 0
                dust_rest_time_2p = 0
                ice_rest_time = 0
                background.update('spring', 1)
                background_2p.update(background_2p.name, 2)

    if not is_ice_time: # 아이스 아이템 먹으면 키 못움직임
        if event.key == pygame.K_w:
            # 스페이스 누르는 시점에 공룡이 땅에 닿아있으면 점프한다.
            if player1_dino.rect.bottom == int(DEFAULT_HEIGHT * height):
                player1_dino.is_jumping = True
                if pygame.mixer.get_init() is not None:
                    jump_sound.play()
                player1_dino.movement[1] = -1 * player1_dino.jump_speed_running

        if event.key == pygame.K_s:
            # 아래방향키를 누르는 시점에 공룡이 점프중이지 않으면 숙인다.
            if not (player1_dino.is_jumping and player1_dino.is_dead):
                player1_dino.is_ducking = True
```

(2) PvP 및 PvP 러닝모드 상수화 및 코드 최적화 진행

: 코드 or 이미지


```

@@ -47,16 +47,13 @@
DUST_INTERVAL = 100
WATER_INTERVAL = 100
ICE_INTERVAL = 100

MAX_WATER = 5
WATER_SIZE_X = 40
WATER_SIZE_Y = 40
ITEM_TIME = 7
ICE_TIME = 3
ONE_SECOND = 1

- dust_rest_time = 0
- water_rest_time = 0
- DUST_APPEARANCE_TIME = 0
- WATER_APPEARANCE_TIME = 0

#store.py
ALPHA_MOVE = 20

47 DUST_INTERVAL = 100
48 WATER_INTERVAL = 100
49 ICE_INTERVAL = 100
50 + POTION_INTERVAL = 100
51 MAX_WATER = 5
52 WATER_SIZE_X = 40
53 WATER_SIZE_Y = 40
54 ITEM_TIME = 7
55 ICE_TIME = 3
56 ONE_SECOND = 1

57
58 #store.py
59 ALPHA_MOVE = 20

124 # 게임 후 버튼
125 - r_btn_restart, r_btn_restart_rect =
126     load_image(*resize('btn_restart.png', 150, 80, -1))
127
128 - btn_restart, btn_restart_rect =
129     load_image('btn_restart.png', 150, 80, -1)
130 - r_btn_exit, r_btn_exit_rect =
131     load_image(*resize('btn_exit.png', 150, 80, -1))
132 - btn_exit, btn_exit_rect = load_image('btn_exit.png', 150,
133     80, -1)

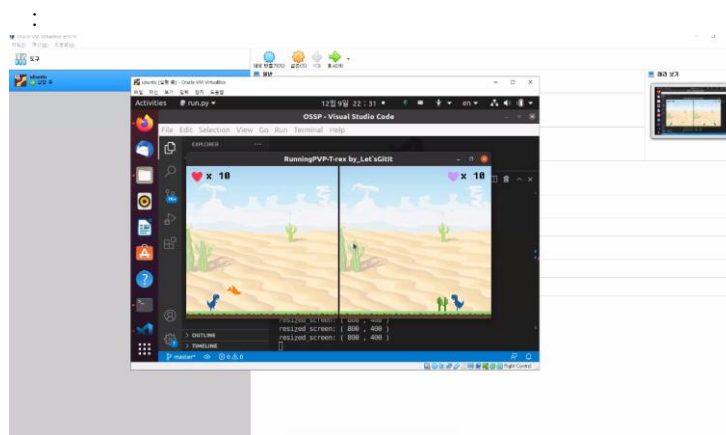
765 - pm.x = random.randrange(48, width-48)
766 - pm.y = 210
767 - pm.move = 3
768 - if len(pm_list) < MAX_WATER:

130 # 게임 후 버튼
131 + r_btn_restart, r_btn_restart_rect =
132     load_image(*resize('btn_restart.png', btn_size_w, btn_size_h,
133     -1))
134 + btn_restart, btn_restart_rect =
135     load_image('btn_restart.png', btn_size_w, btn_size_h, -1)
136 + r_btn_exit, r_btn_exit_rect =
137     load_image(*resize('btn_exit.png', btn_size_w, btn_size_h, -1))
138 + btn_exit, btn_exit_rect = load_image('btn_exit.png',
139     btn_size_w, btn_size_h, -1)

784 + pm.x = random.randrange(WATER_SIZE_X, width
785     - WATER_SIZE_X)
786 + pm.y = 210 # 2P에게 물방울 등장 높이
787 + pm.move = 3 # 물방울 속도
788 + if len(pm_list) < MAX_WATER: # 물방울 개수 제한
789     continue

```

(3) 게임 데모 영상 제작



+ 개발 사항 PR 및 merge 완료 현황

2. 12월 2주차 개발 계획

- [재혁]: 최종 보고서 및 발표 자료 제작, 코드 최적화 및 상수화
- [재홍]: 데모 영상 제작 및 발표 자료 제작, 코드 최적화 및 상수화
- [영철]: 최종 보고서 및 발표 자료 제작, 상수화, 라이선스 및 필요 자료 조사

	내용	진행일정
결정사항	대면회의 진행 - 최종 발표 준비 회의	12월 13일
	최종 발표	12월 14일

특이사항	
------	--
