



객체지향 프로그래밍 팀프로젝트

Zombie-EXIT

김도형 김혁민 윤영배 이우성



I N D E X



Zombie EXIT

- 01 게임 소개
 - 02 게임 줄거리
 - 03 핵심 코드
 - 04 게임 시연
-



1. 게임소개



01 게임 소개



〈FPS〉



〈RPG〉



〈스토리〉



〈전략〉



〈퍼즐〉

+ 방탈출



〈배틀〉



2. 게임 줄거리



좀비를 피해 들어간 건물...

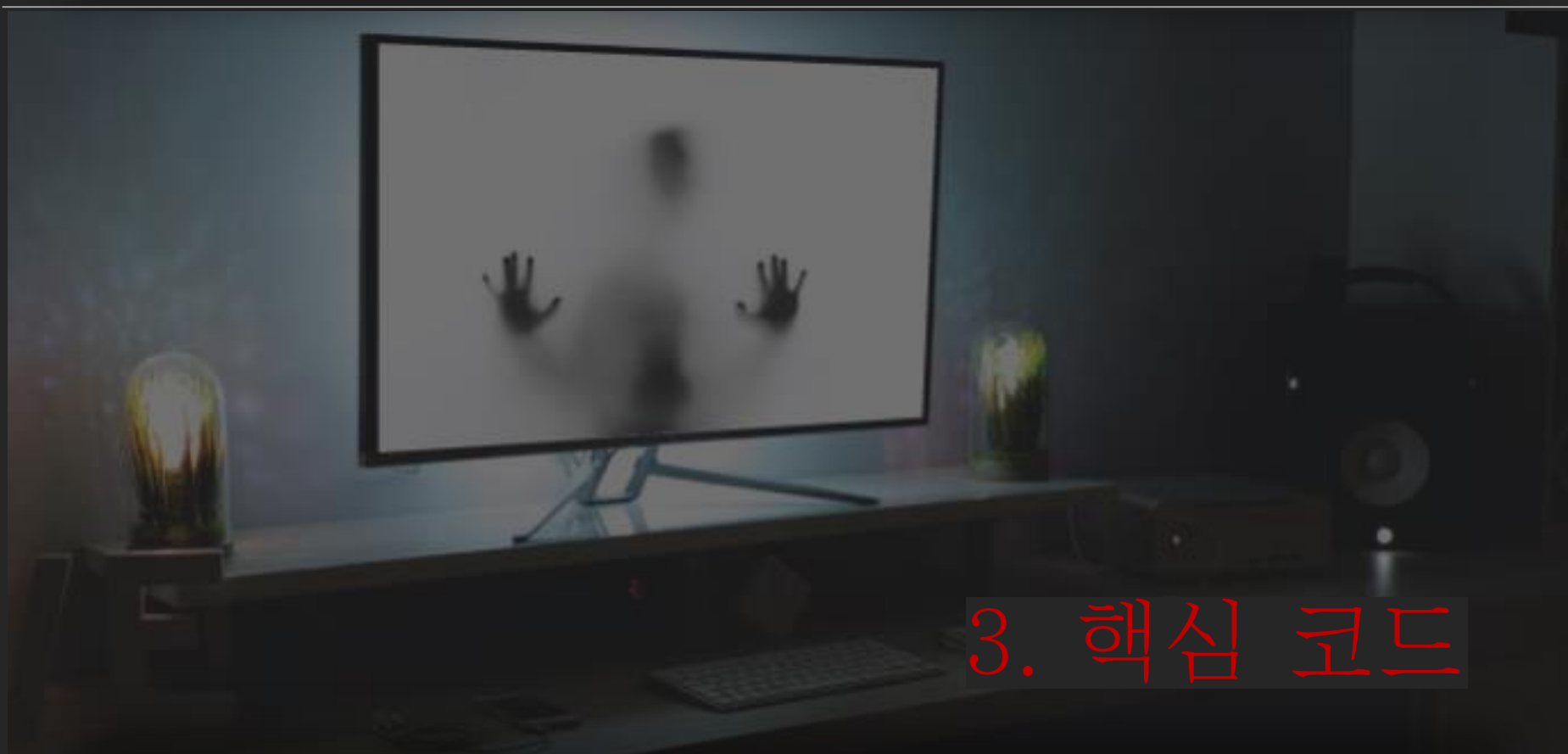
그곳에서 자신을 '상점 아줌마'라 소개하는 조력자를 만나게 되는데...

아주머니의 도움을 받으며 성장 해 나가는 와중

건물에 얹힌 비밀과 흑막에 대해 조금씩 알아가게 되는 플레이어!

목숨을 위협하는 좀비로부터 생존하여

안전하게 건물을 탈출하라!!



3. 핵심 코드



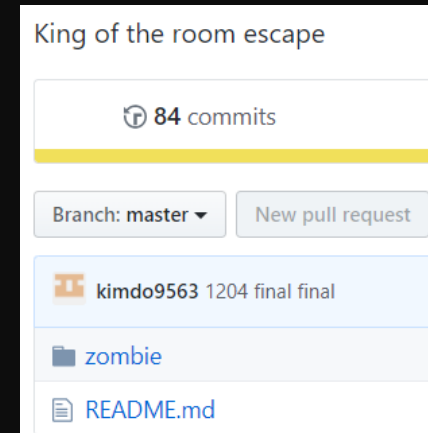
1. 모든 기능의 객체화를 통한 코드 최적화
2. 115여개의 이미지와 15종류의 사운드로 만족감 선사
3. 곳곳에 산재한 게임오버 요소
4. 플레이어의 전략적인 체력, 스테미너, 금액 운용
5. 퀘스트를 바탕으로 한 탄탄한 스토리라인
6. 보스 전투 시스템 구현
7. 방탈출의 한계를 뛰어넘는 퍼즐들
8. 엘리베이터를 통한 이동과 오픈 월드 시스템



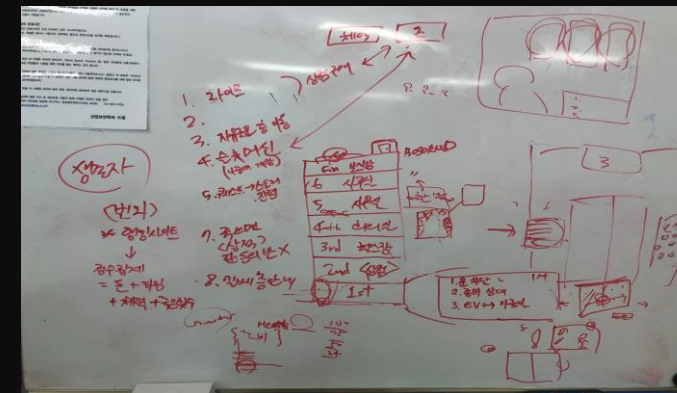
03 핵심 코드



모든 멤버의 적극적 개발



최소 80번의 commit



탄탄한 설계 과정



03 핵심 코드 - 화살표

```
function arrow(room, name, go_to_room, width, x_loc, y_loc) {
  if (name == "left_arrow") {
    obj.call(this, room, name, "left_arrow.png", width, x_loc, y_loc) //150, 100, 360
  } else if (name == "right_arrow") {
    obj.call(this, room, name, "right_arrow.png", width, x_loc, y_loc) //150, 1200, 360
  } else if (name == "down_arrow") {
    obj.call(this, room, name, "down_arrow.png", width, x_loc, y_loc) //200, 640, 650
  } else if (name == "up_arrow") {
    obj.call(this, room, name, "up_arrow.png", width, x_loc, y_loc) //200, 640, 650
  }
  this.go_to_room = typeof go_to_room !== 'undefined' ? go_to_room: this.room;
}
arrow.prototype = Object.create(obj.prototype);
arrow.prototype.constructor = arrow;
arrow.prototype.onClick = function () {game.move(this.go_to_room)}
```

```
_2nd_floor_one.down_arrow = new arrow(_2nd_floor_one, "down_arrow", _elevator, 100, 1200, 650)
```



03 핵심 코드 - 배틀필드

```
function battle(come_to_room, enemy){
    original_zombie = enemy;
    _battle_field.button_exit.onClick = function() {game.move(come_to_room); }

    _battle_field.zombie.obj.setSprite(enemy.image)
    _battle_field.zombie.life = enemy.life;
    _battle_field.zombie.damage = enemy.damage;

    game.move(_battle_field)
}

function zombie(room, name, image, width, x_loc, y_loc, life, damage) {
    obj.call(this, room, name, image, width, x_loc, y_loc);
    this.name = name;
    this.image = image;
    this.width = width;
    this.x_loc = x_loc;
    this.y_loc = y_loc;
    this.life = life;
    this.damage = damage;
}

zombie.prototype = Object.create(obj.prototype)
zombie.prototype.constructor = zombie;
zombie.prototype.onClick = function() {
    battle(this.room, this)
}
```



03 핵심 코드 - 배틀필드

```
var original_zombie; //좀비를 처치 시, 원래 위치의 좀비를 hide 하기 위한 용도

_battle_field.button_attack = new empty_box(_battle_field, "button_attack", 150, 370, 510)
_battle_field.button_attack.obj.setSprite("empty_box3.png")
_battle_field.button_skill = new empty_box(_battle_field, "button_skill", 150, 410, 610)
_battle_field.button_skill.obj.setSprite("empty_box3.png")
_battle_field.button_exit = new empty_box(_battle_field, "button_exit", 100, 1000, 620)

_battle_field.button_attack.onClick = function () {
    try {
        playSound("zombie_hit.wav")
        Player.weapon();
        _battle_field.zombie.life -= Player.damage;
        Player.life_change(0-_battle_field.zombie.damage);
        printMessage("아다다다다 !\n"+"남은 HP("+Player.life+") 좀비 HP("+_battle_field.zombie.life+)");

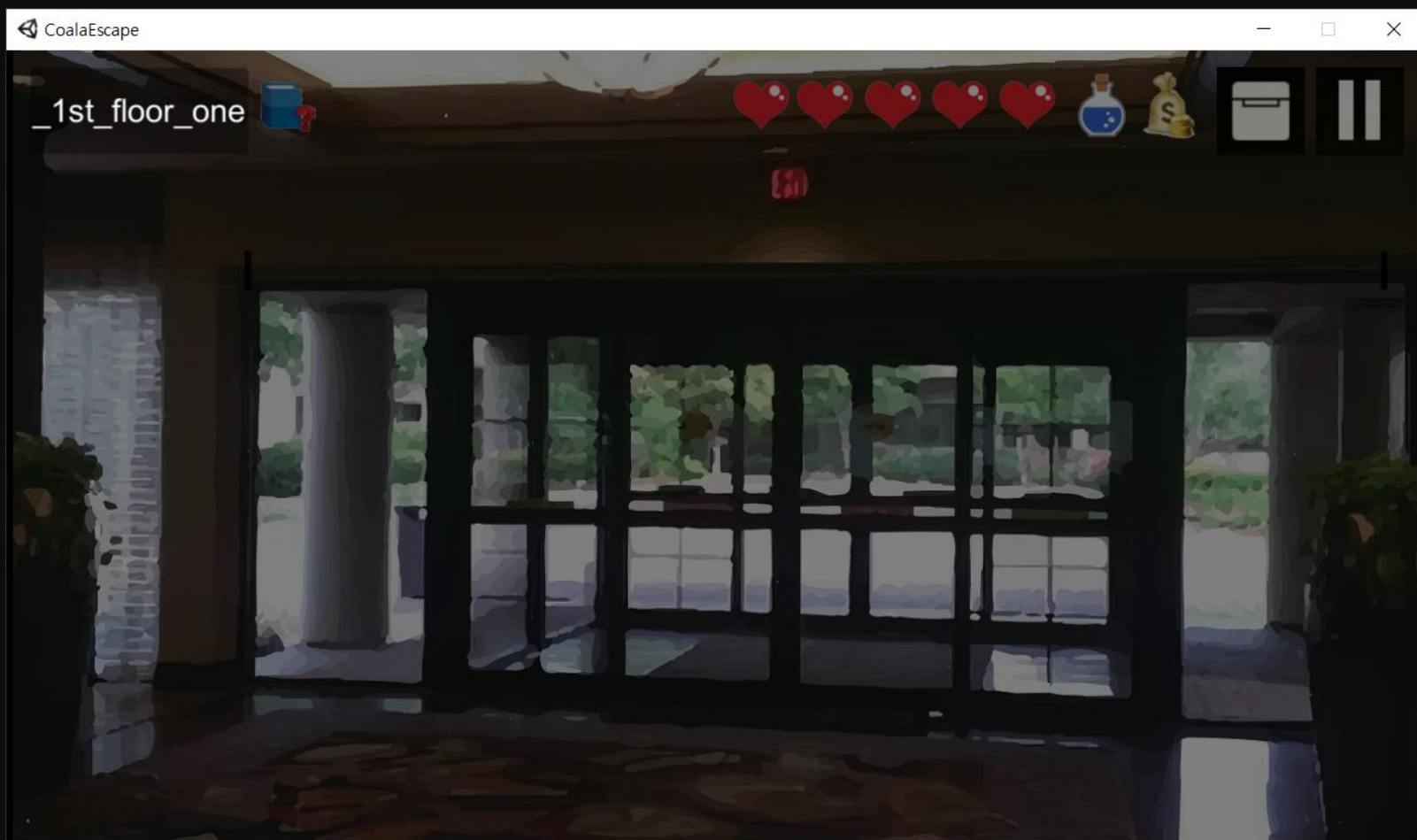
        if(_battle_field.zombie.life <= 0) {
            playSound("zombie_die.wav")
            printMessage("좀비를 무찔렀다!")
            _battle_field.button_exit.onClick()
            original_zombie.obj.hide();
            original_zombie.obj.lock();
        }
    } catch(e) {
        printMessage(e)
    }
}

// 문지기 좀비
_4th_floor_one.zombie1 = new zombie(_4th_floor_one, "zombie1", "3층좀비_4.png", 100, 640, 450, 30, 2);
```



03

핵심 코드 - HP, MP, 퀘스트, 소지금 구현





03

핵심 코드 - HP, MP, 퀘스트, 소지금 구현

```
function player_control() {  
    // 반드시 할당되는 객체의 이름을 "Player"로 해야합니다!  
    // var Player = player_control();  
  
    this.life = 0;  
    this.stamina = 0;  
    this.money = 0;  
  
    this.quest_index = 0;  
    this.quest_message = " ";  
  
    this.damage = 0;  
    this.skill_name = "";  
    this.skill_damage = 0;  
}  
  
/*  
// Prototype Description  
life_create      : UI에 하트 생성  
life_change      : 라이프 변경(eg.공격을 받거나, 회복 아이템 사용 시)  
life_change_image : change 메소드에 따른, 라이프 이미지 변경(Do NOT use directly)  
  
stamina_create   : UI에 기력 생성  
stamina_change   : 기력 변경(eg. 스킬 사용, 회복 아이템 사용 시)  
  
money_create     : UI에 소지금 생성  
money_change     : 소지금 변경(eg.상점 이용, 플레이어 사망 시)  
  
weapon           : 호출 시마다, 손에 장착한 무기를 바탕으로, 플레이어 스탯 정보 갱신  
  
quest_create     : UI에 퀘스트 생성  
*/
```




03

핵심 코드 - HP, MP, 퀘스트, 소지금 구현

```
player_control.prototype.life_create = function() {  
    for (var i = 0; i < room_list.length; i++){  
        room = room_list[i];  
        room.life = new obj(room, "life", "life_5.png", 300, 800, 50);  
        room.life.onClick = function(){printMessage("HP : "+Player.life)}  
    }  
}
```

```
player_control.prototype.life_change = function(change_life_amount) {  
    var changed_life = this.life + change_life_amount;  
  
    if (changed_life <= 0) {game.gameover()} // 라이프가 0이하면 게임오버  
    if (changed_life > 100) {throw "체력이 최대이므로 회복될 수 없습니다."}  
  
    if (Math.floor(this.life/10) != Math.floor(changed_life/10)) {  
        //자릿수가 바뀔 경우에만 실행  
        if(changed_life > 0 && changed_life < 20) {this.life_change_image("life_1.png");}  
        if(changed_life >= 20 && changed_life < 40) {this.life_change_image("life_2.png");}  
        if(changed_life >= 40 && changed_life < 60) {this.life_change_image("life_3.png");}  
        if(changed_life >= 60 && changed_life < 80) {this.life_change_image("life_4.png");}  
        if(changed_life >= 80 && changed_life < 100) {this.life_change_image("life_5.png");}  
    }  
    this.life = changed_life;  
}  
  
player_control.prototype.life_change_image = function(change_to_life_image) {  
    for(var i = 0; i < room_list.length; i++) {  
        room = room_list[i];  
        room.life.obj.setSprite(change_to_life_image)}  
}
```



```
//💡 라이프, 소지금, 퀘스트, 기력 등이 보이길  
//원하는 방을 생성하면, room_list 배열에 동기화 필수!  
var room_list = new Array(  
    _battle_field,  
    _battle_field_boss,  
    _1st_floor_one,  
    _1st_floor_two,  
    _1st_floor_three,  
    _2nd_floor_one,  
    _3rd_floor_one,  
    _4th_floor_one,  
    _4th_floor_two,  
    _4th_floor_three,  
    _5th_floor_one,  
    _5th_floor_two,  
    _5th_floor_three,  
    _roof_top_one,  
    _roof_top_two,  
    _boss_room_1,  
    _boss_room_2,  
    _boss_room_3,  
    _boss_room_4,  
    _boss_room_5);
```

```
//Player Initialize  
var Player = new player_control();  
  
Player.life_create()  
Player.life_change(100)  
  
Player.stamina_create()  
Player.stamina_change(50)  
  
Player.money_create()  
Player.money_change(500)  
  
Player.quest_create()
```



4. 게임 시연



팀원 소개

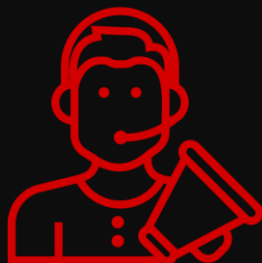


김도형(24)

게임 제작 총괄

헤더 코드 작성

상점, 배틀필드 담당



김혁민(25)

음향 총괄

디자인 검수

깃허브 담당



윤영배(25)

게임 스토리 기획

보스 스테이지 총괄

미니게임 담당



이우성(24)

디자인 제작

코드 작성

전략 요소 담당



감사합니다