J.

Portfollo

남 홍 근

|010-3254-5455|

ghd[illegible]s1220@n[illegible].co[illegible]|

J

I N D E X

1 . 학 부 생

한 이 음 프 로 젝 트

C h a s e r

마 리 오 의 미 로 찾 기

R o l l i n g B a l l

2 . 재 직 중

S & G

U N I T Y

D E V E L O P E R

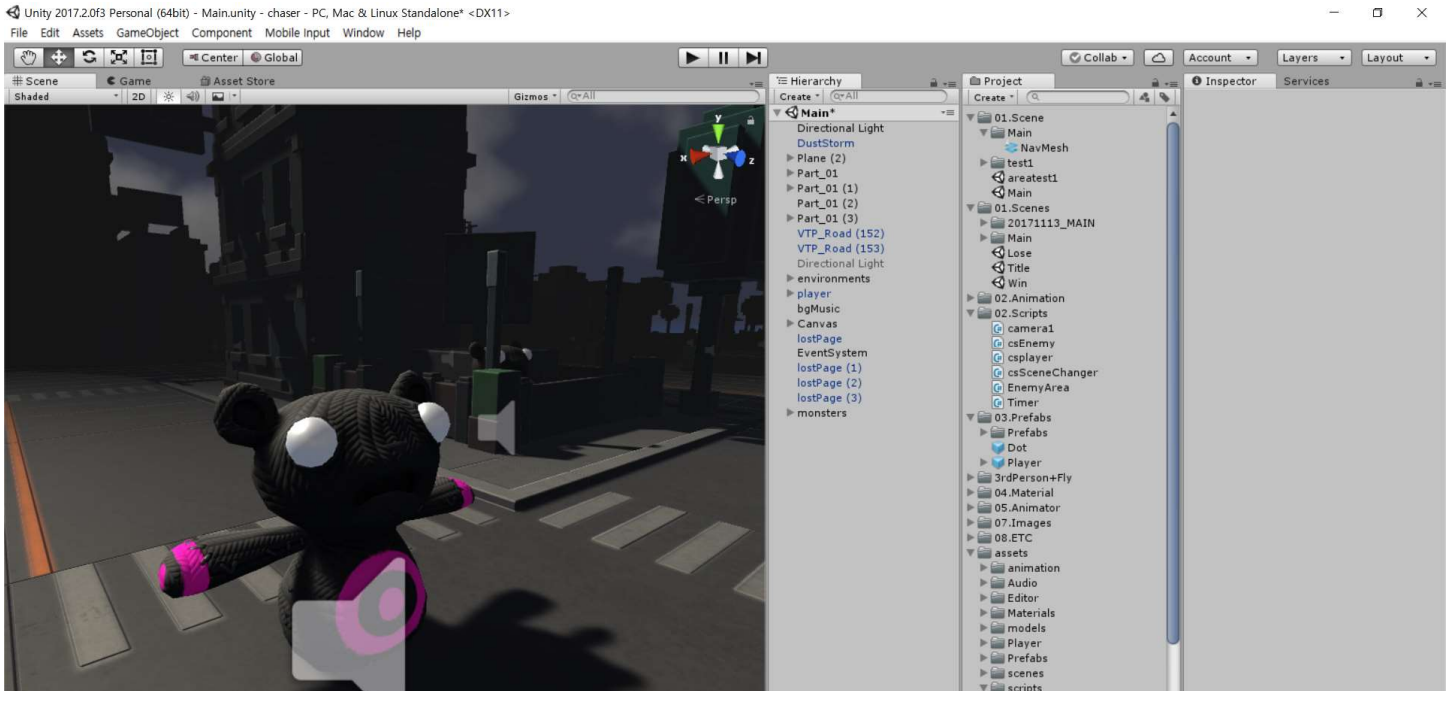

J
1
학 부 생

프로젝트명	Chaser
개발인원(총인원)	3명(8명)
개발기간	2018.03~2019.01
개발프로그램(개발언어)	Unity(C#)
진행 사유	창업 동아리 프로젝트
프로젝트 내용	다음 장 그림 참조
본인 역할	개발 서브 개발자 - UI/UX 개발 및 네브 매쉬 등 다수 개발
프로젝트 소개	1인칭 3D 러닝 게임 UI : UGUI와 스프라이트 네브매쉬 에이전트를 통해서 적들의 움직임을 자연스럽게 진행 상태머신을 만들어 최적화 진행 유니티 콜라보레이트를 통한 협업진행


Pages 0/4

게임화면


Pages 0/4


Unity 2017.2.0f3 Personal (64bit) - Main.unity - chaser - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Mobile Input Window Help
Hierarchy
Project
Inspector
Services
Main*
Directional Light
DustStorm
Plane (2)
Part_01
Part_01 (1)
Part_01 (2)
Part_01 (3)
VTP_Road (152)
VTP_Road (153)
environments
player
bgMusic
Canvas
lostPage
EventSystem
lostPage (1)
lostPage (2)
lostPage (3)
monsters
01.Scene
Main
NavMesh
test1
areatest1
01.Scenes
20171113_MAIN
Lose
Title
Win
02.Animation
02.Scripts
camera1
csEnemy
csplayer
csSceneChanger
EnemyArea
Timer
03.Prefabs
Prefabs
Dot
Player
3rdPerson+Fly
04.Material
05.Animator
07.Images
08.ETC
assets
animation
Audio
Editor
Materials
models
Player
Prefabs
scenes
scripts

```
    if(state == "idle")
    {
        //pick a random place to walk to//
        Vector3 randomPos = Random.insideUnitSphere*alertness;
        NavMeshHit navHit;
        NavMesh.SamplePosition(transform.position + randomPos, out navHit,20f,NavMesh.AllAreas);

        //go near the player//
        if(highAlert)
        {
            NavMesh.SamplePosition(player.transform.position + randomPos, out navHit,20f,NavMesh.AllAreas);
            //each time, lose awareness of player general position//
            alertness += 5f;

            if(alertness > 20f)
            {
                highAlert = false;
                nav.speed = 1.2f;
                anim.speed = 1.2f;
            }
        }


        nav.SetDestination(navHit.position);
        state = "walk";
    }
    //Walk//
    if(state == "walk")
    {
        if(nav.remainingDistance <= nav.stoppingDistance && !nav.pathPending)
        {
            state = "search";
            wait = 5f;
        }
    }
//check if we can see the player//
public void checkSight()
{
    if(alive)
    {
        RaycastHit rayHit;
        if(Physics.Linecast(eyes.position,player.transform.position, out rayHit))
        {
            //print("hit "+rayHit.collider.gameObject.name);
            if(rayHit.collider.gameObject.name == "player")
            {
                if(state != "kill")
                {
                    state = "chase";
                    nav.speed = 3.5f;
                    anim.speed = 3.5f;
                    growl.pitch = 1.2f;
                    growl.Play();
                }
            }
        }
    }
}
```

상황별 상태머신으로
캐릭터를 컨트롤

레이캐스트를 통한
적군의 주인공 식별

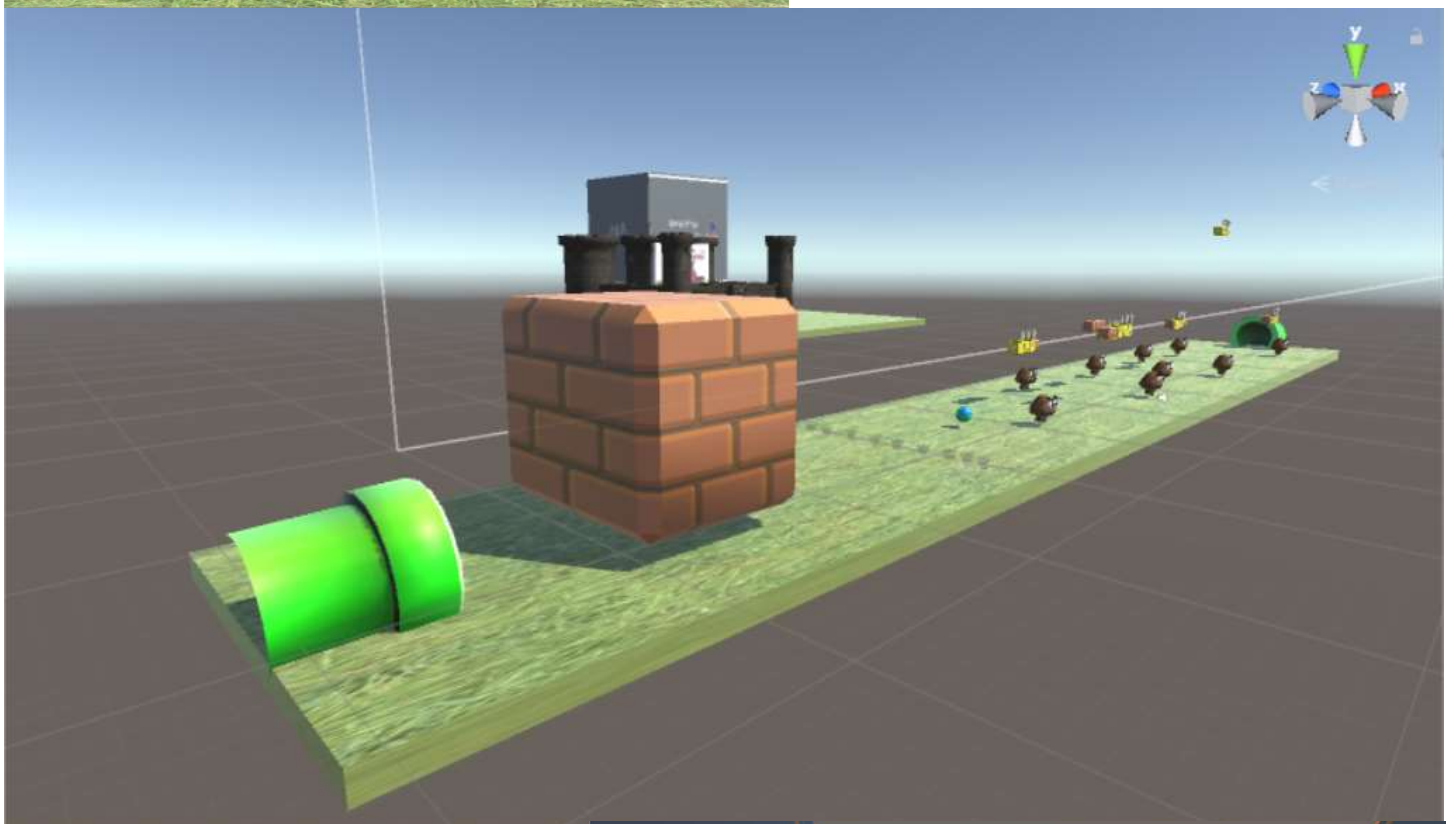
Death캠을 통해 죽었을
경우 카메라를 바꿔줌

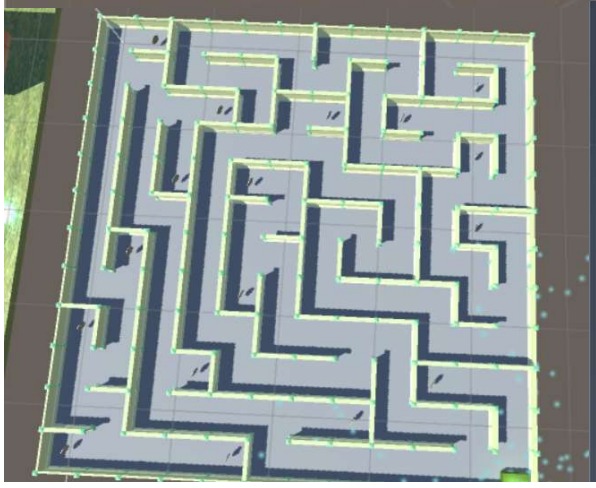
```
//Kill//
if(state == "kill")
{
    deathCam.transform.position = Vector3.Slerp(deathCam.transform.position,camPos.position,10f*Time.deltaTime);
    deathCam.transform.rotation = Quaternion.Slerp(deathCam.transform.rotation,camPos.rotation,10f*Time.deltaTime);
    anim.speed = 1f;
    nav.SetDestination(deathCam.transform.position);
}

//nav.SetDestination(player.transform.position);
```

프로젝트명	마리오의 미로찾기
개발인원(총인원)	3명
개발기간	2018.05.08~2018.06.23
개발프로그램(개발언어)	Unity(C#) + UDP
진행 사유	과제
프로젝트 내용	다음 장 그림 참조
본인 역할	개발 팀장 - 파티클 및 teleport 기능 구현 등 - 미로제작 및 디버깅
프로젝트 소개	마리오 게임을 모티브 삼아 VR게임으로 제작 매번 다른 미로 모양을 통해 새로운 게임의 재미를 선사 UDP통신을 통해 핸드폰을 컨트롤러 삼아 게임 가능 숨겨놓은 이스터에그를 통해 바로 게임의 끝으로 갈 수 있음

게임화면


UDPReceive
127.0.0.1 10000 #
shell> nc -u 127.0.0.1 : 10000
Last Packet:
All Messages:
Use SPACE to jump.
Collect coins!






THANKS FOR
PLAYING

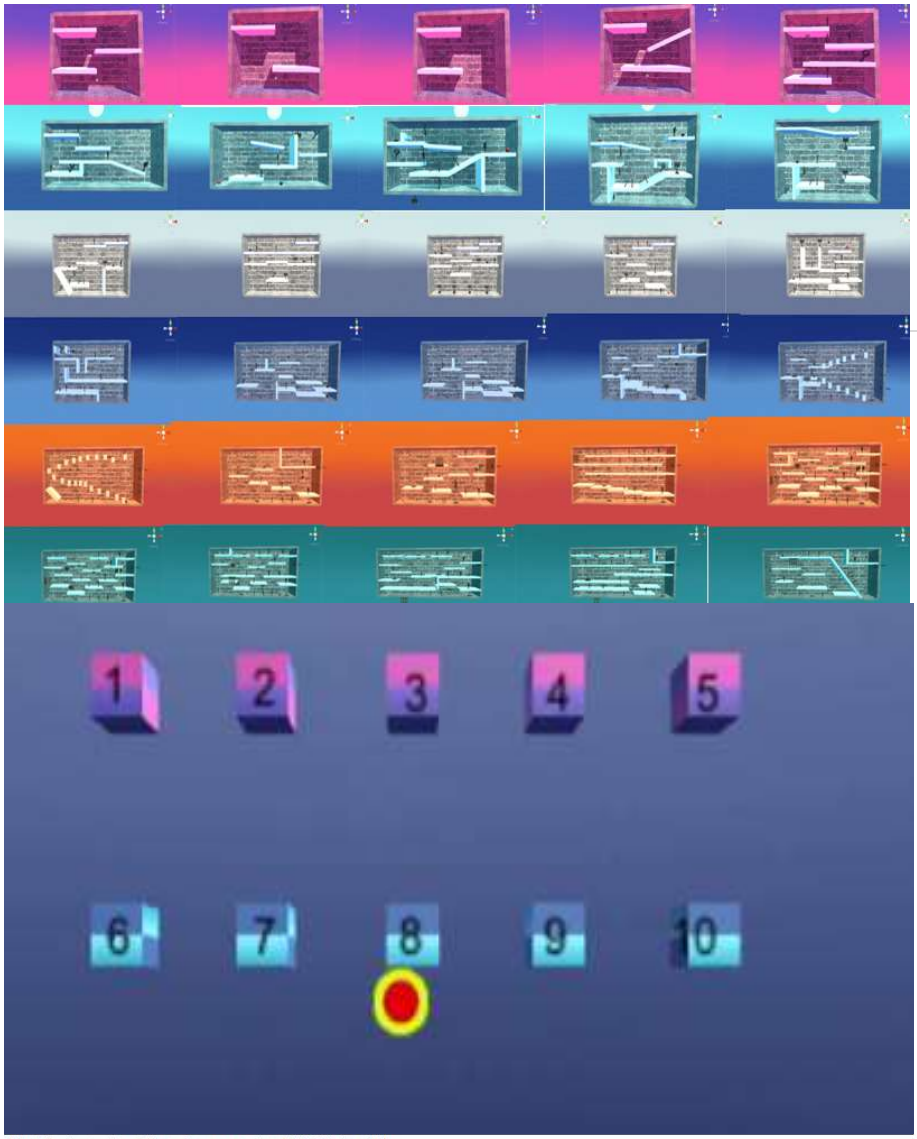
```
}

// init
private void init()
{
    rb = GetComponent<Rigidbody>();

    // Endpunkt definieren, von dem die Nachrichten gesendet werden.
    print("UDPSend.init()");

    // define port
    port = 10000;

    // status
    //print("Sending to 127.0.0.1 : " + port);
    //print("Test-Sending to this Port: nc -u 127.0.0.1  " + port + "");


    // ----------------------------
    // Abhören
    // ----------------------------
    // Lokalen Endpunkt definieren (wo Nachrichten empfangen werden).
    // Einen neuen Thread für den Empfang eingehender Nachrichten erstellen.
    receiveThread = new Thread(
        new ThreadStart(ReceiveData));
    receiveThread.IsBackground = true;
    receiveThread.Start();

}
```

```
    client = new UdpClient(port);
    while (true)
    {

        try
        {
            // Bytes empfangen.
            IPEndPoint anyIP = new IPEndPoint(IPAddress.Any, 0);
            byte[] data = client.Receive(ref anyIP);

            // Bytes mit der UTF8-Kodierung in das Textformat kodieren.
            string text = Encoding.UTF8.GetString(data);

            // Den abgerufenen Text anzeigen.
            print(">> " + text);

            if(text.Trim().Equals("a"))
            {
                Vector3 movement = new Vector3(-100, 0, 0);
                rb.AddForce(movement);
            }
            else if (text.Trim().Equals("d"))
            {
                Vector3 movement = new Vector3(100, 0, 0);
                rb.AddForce(movement);
            }
            else if(text.Trim().Equals("w"))
            {
                Vector3 movement = new Vector3(0, 0, 100);
                rb.AddForce(movement);
            }
            else if(text.Trim().Equals("s"))
            {
                Vector3 movement = new Vector3(0, 0, -100);
                rb.AddForce(movement);
            }
            else if(text.Trim().Equals("j"))
            {
                Vector3 movement = new Vector3(0, 100, 0);
```

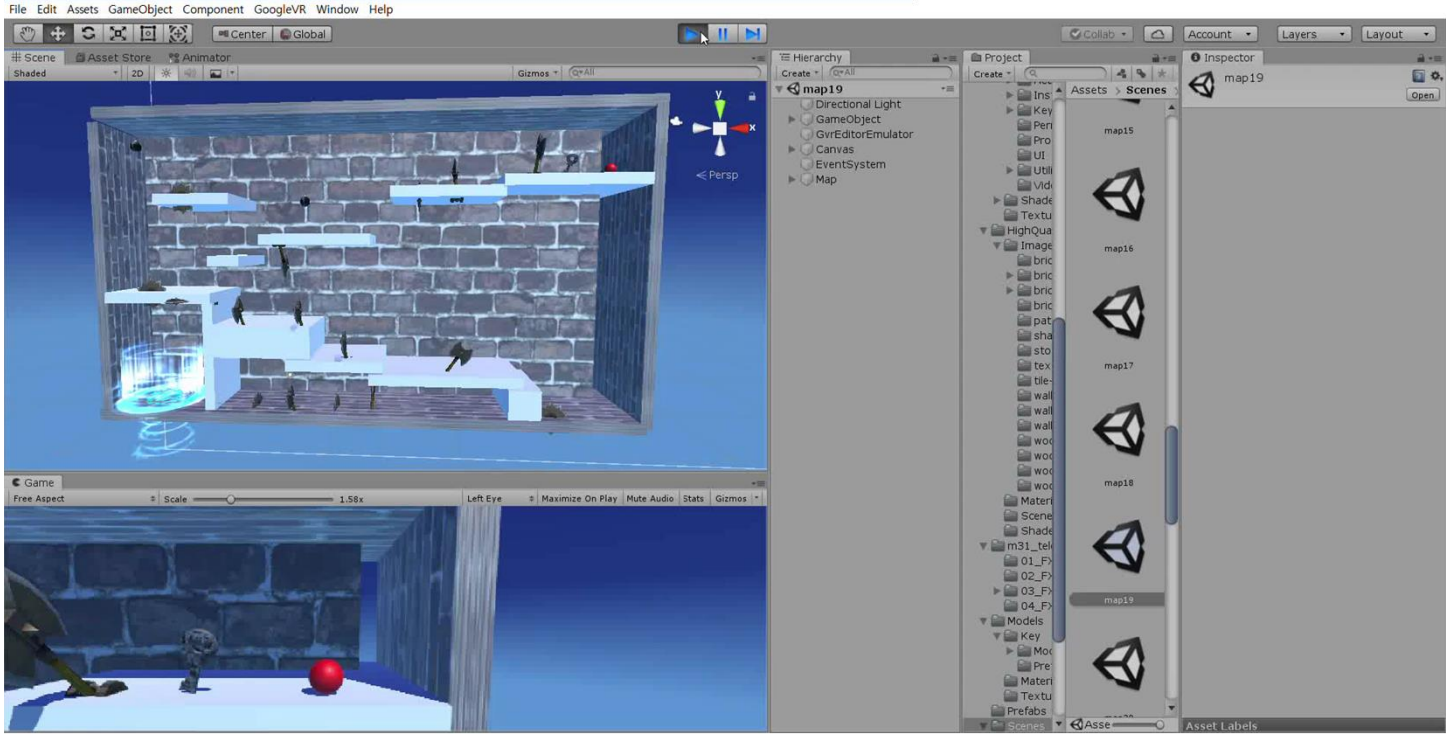
```
// getLatestUDPPacket
// cleans up the rest
public string getLatestUDPPacket()
{
    allReceivedUDPPackets = "";
    return lastReceivedUDPPacket;
}
```

UDP 통신 및
관련 콜백

프로젝트명	oll ng ll(dco)
개발인원(총인원)	1명
개발기간	2019.05~2019.06.05
개발프로그램(개발언어)	Un ty(#)
진행 사유	과제
프로젝트 내용	Un ty를 활용하여 v 게임 개발 다음 장 그림 참조
본인 역할	맵 디자인 및 개발 총괄
프로젝트 소개	스테이지 퍼즐 형식의 게임 총 30 스테이지의 레벨디자인 애니메이션을 통한 장애물의 움직임


1
2
3
4
5
6
7
8
9
10

게임화면 및
동영상


File Edit Assets GameObject Component GoogleVR Window Help
Center Global
Collab
Account
Layers
Layout
Scene
Asset Store
Animator
Persp
Hierarchy
map19
Directional Light
GameObject
GvrEditorEmulator
Canvas
EventSystem
Map
Project
Assets > Scenes
Inspector
map19
Open
Game
Free Aspect
Scale
1.58x
Left Eye
Maximize On Play
Mute Audio
Stats
Gizmos
Asset Labels

```
void Jump()
{
    if (Input.GetMouseButtonDown(0))
    {
        if (!IsJumping)
        {
            IsJumping = true;
            rigid.AddForce(Vector3.up * JumpPower, ForceMode.Impulse);
        }
        else
        {
            return;
        }

    }
}

 }
 void Update()
 {

     if (Input.GetMouseButtonDown(0))
         startTime = Time.time;
     if (Input.GetMouseButtonUp(0))
         endTime = Time.time;

     if(endTime - startTime > 0.5f)
     {
         map.SetActive(false);

         Continue.SetActive(true);
         Quit.SetActive(true);
         Cam.SetActive(true);

         startTime = 0f;
         endTime = 0f;
     }

}
private void OnTriggerEnter(Collider other)
{
    if (other.gameObject.CompareTag("Finish"))
    {
        if(Item == true)
        StartCoroutine(Clear());
    }
    if (other.gameObject.CompareTag("Key"))
    {
        Item = true;
        Destroy(ItemKey);



    }
    if (other.gameObject.CompareTag("Respawn"))
    {
        IsOpen = true;
        open.SetActive(false);
        ball.SetActive(false);
        IsEat = true;

    }
    if (other.gameObject.CompareTag("Finish"))
    {

        open1.SetActive(true);
        ball1.SetActive(false);
        IsEat = true;

    }


}
```

무한 점프를
막기위한 코드

시간을 체크하여
메뉴를 불러옴

태깅을 통한
게임 제어


J
2
재 직 중

프로젝트명	S & G
회사명	피비컴
재직기간	2019.12.16 ~ 재직중
개발프로그램(개발언어)	Unity(C#)

11개국 글로벌 작업

- 첫 로그인 시 국가정보 추출 및 DB 저장, 설정에서 언어 변경 가능

TCP/IP통신

- 서버에서 보내주는 JSON데이터를 파싱하여 유저 정보를 저장

게임 개발

- 룰렛, 농구게임, 우편함, 랭킹, 프로필 개발 및 기존 게임 리팩토링 작업

Admob, Pincrux, GPGS 등 SDK 작업

- 구글 플레이 로그인, 애드몹 광고 적용, 핀크럭스 광고 적용, 인앱 결제 적용

빌드 작업

- Jenkins 빌드 자동화 및 깃 플로우 사용

업데이트 작업

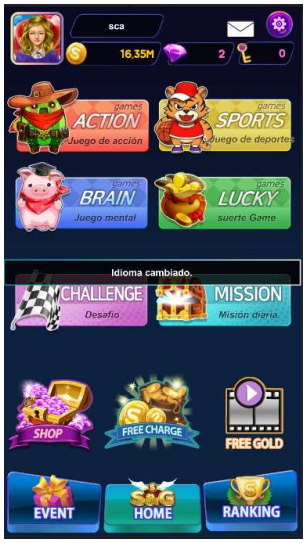
- 게임 리팩토링 및 게임 추가 등 작업 및 버그 수정
- 게임 출시 전부터 출시 후 작업까지 진행

```
IEnumerator APICalls_MyInfo_Real(string str_api)
{

    UnityWebRequest www = UnityWebRequest.Get(str_api);
    www.timeout = GameManager.API_TIMEOUT;
    yield return www.SendWebRequest();


    if (www.isNetworkError || www.isHttpError)
    {

        var lobby = Camera.main.GetComponent<Lobby>();
        if (lobby)
        {
            lobby.Loading_Start();
        }
        Debug.Log(www.error);
        if (GameManager.IS_GAME_LOBBY)
        {
            if (lobby)
            {
                yield return new WaitForSeconds(3.0f);
                lobby.Loading_Stop();
                Sounds.Play("MP3_ui_1step_01");
                lobby.mGameObject_ReStart.SetActive(true);

            }
        }
    }
```

Json 데이터 파싱

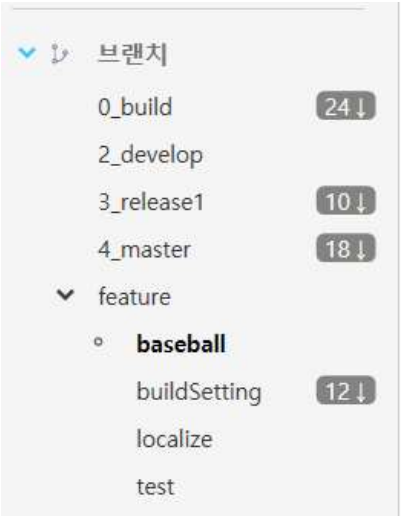
```
        }
        break;
    case GameManager.ENUM_RACE_MODE.SPINNING:
        {

            BrokenWall();
            LeftTime();
            // Calcu...Score...
            int nScore = mint_Myscore;
            m_nAnswerCurr = nScore;
            m_fScore_Total = nScore;
            mText_Score.text = ((int)(m_fScore_Total)).ToString();


            // refiesh...
            if (Time.time >= m_fTime_Broadcast)
            {
                m_fTime_Broadcast = Time.time + TIME_TICK_BROADCAST;
                Broadcast_ScoreInfo();
            }

            // Check...GameOver...
            if (Time.time >= m_fTime_GameStart + TIME_GAME_LIMIT)
            {
                if (m_bSweep == false)
                {
                    ChangeState(GameManager.ENUM_RACE_MODE.OVER);
                }
            }
        }
        break;
    case GameManager.ENUM_RACE_MODE.OVER:
        {
            if (m_fTime_GameCurr < m_fTime_GameOver)
            {
                // Do Nothing...
            }
            else
            if (GameManager.USER_IDX == 0)
            {
                if (m_nCurrRoomUsers == 1)
                {
                    if (!mbResultSend)
                    {
                        Broadcast_Result();
                    }
                }
                else if (Time.time >= m_fTime_GameStart + TIME_GAME_LIMIT)
                {
                    if (!mbResultSend)
                    {
                        Broadcast_Result();
                    }
```

상태이상 머신으로 농구 게임



리뉴얼 전

리뉴얼 후



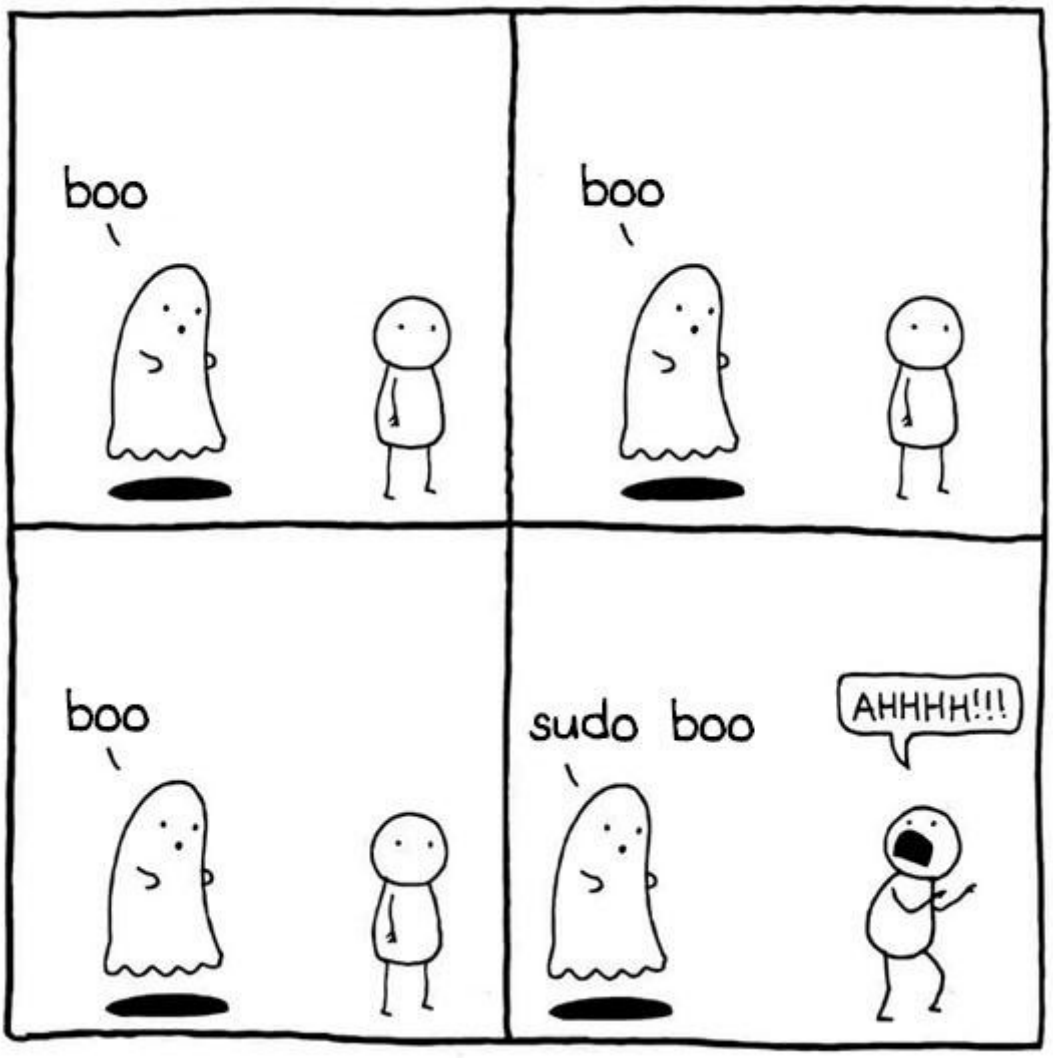
깃 플로우

젠킨스 빌드

포트폴리오 Git주소

https://github.com/Amborsia/portfolio

Thanks you



사람들과 함께 하면서 실력을 키우는 남홍근

비 전공생이지만 전공생만큼 잘한다고 이야기 듣는 남홍근

한 번 알아두면 어떤 일에서든 쓸모가 있는 남홍근

항상 학구열이 넘치며 열심히 하는 남홍근

이런 남홍근이 되도록 하겠습니다

사진 출처 : https://swiftcoding.org/sysadmin-humour