

박종찬 데브캣 포트폴리오

1. 바운티볼

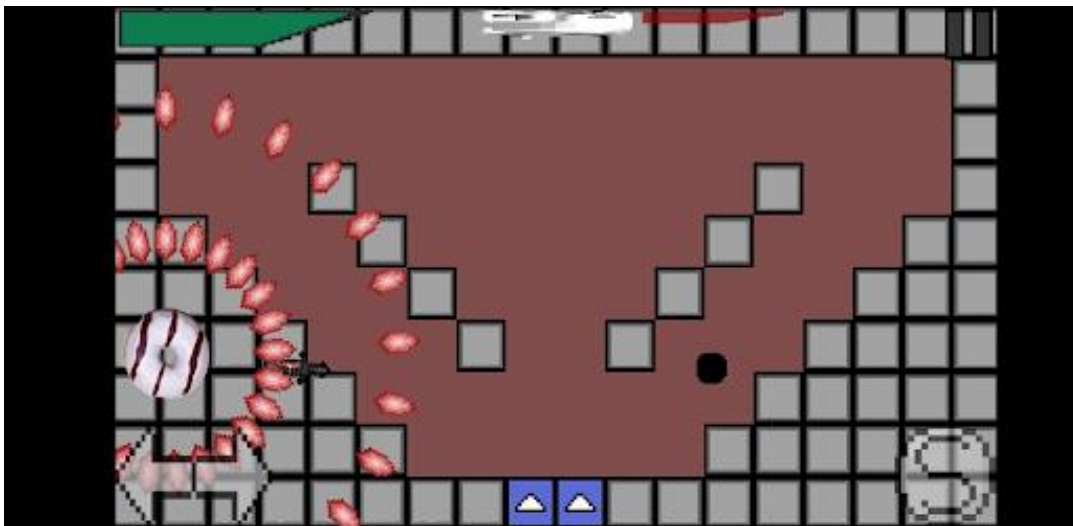
플랫폼: Android

설명: 공 튀기기와 비행 슈팅을 결합해, 공이 튀길 때마다 투사체로 적을 공격하는 아케이드 게임

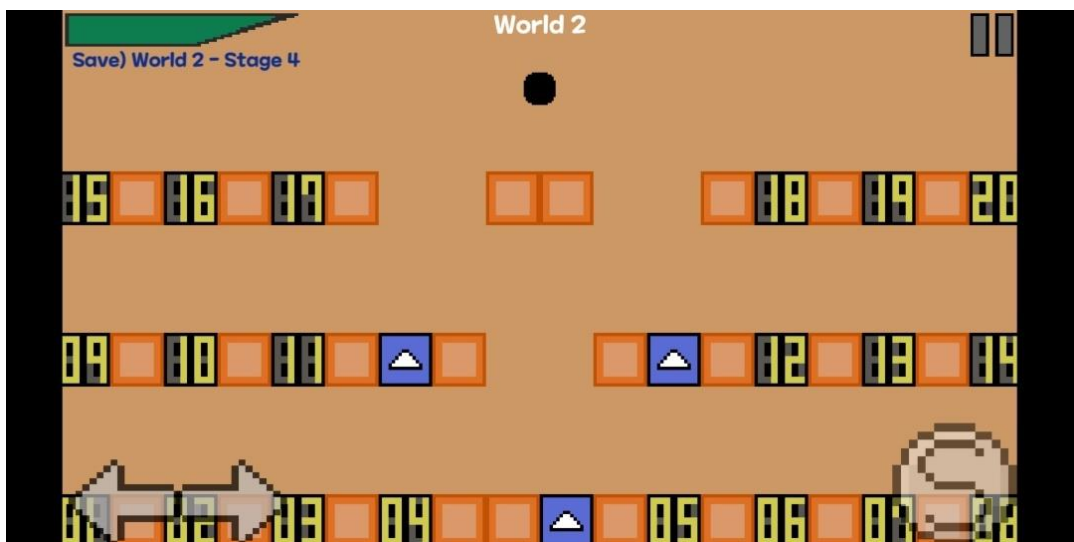
블로그에 정리한 내용: <https://blog.naver.com/whdcks522/223197881025>

동영상: <https://www.youtube.com/watch?v=UX2JkhX3EJI>

플레이 스토어 링크: <https://play.google.com/store/apps/details?id=com.StudyCompany.BountyBall>



전투 화면(공이 튀길 때마다 투사체로 적을 공격)



선택 화면(플레이할 수 있는 메인 화면)

2. Project D

플랫폼: Windows

설명: 최대 2명이 함께 하며 업적을 달성하는 멀티 게임

블로그에 정리한 내용: <https://blog.naver.com/whdcks522/223187126544>

동영상:

https://www.youtube.com/watch?v=vduoog1lksM&list=PLz_DIf14tApmJ3Plu2MOWivJIGn8n0zRG&index=2



로비(점차 나타나는 셰이더)



로비(마우스를 바라봄, 파이어베이스로 계정 연동, 업적 관리)



실행 화면(채팅 구현)

3. 우리들의 폰게임

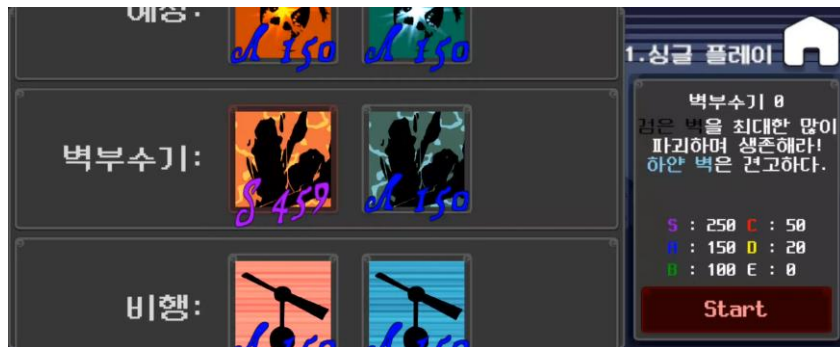
플랫폼: Android

설명: 컴투스의 '슈퍼액션히어로'와 유사하게 다수의 미니게임으로 구성된 게임

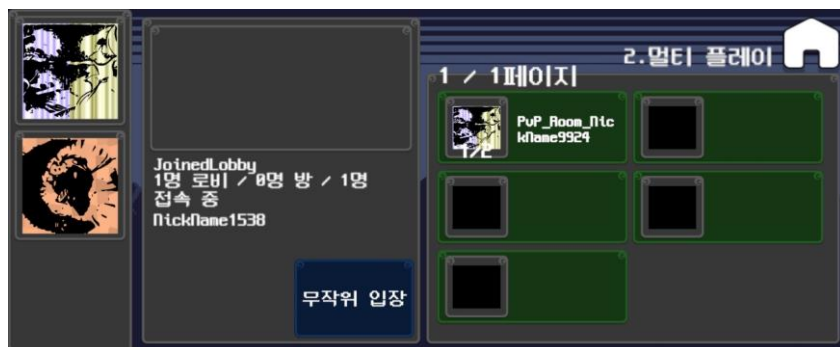
블로그에 정리한 내용: <https://blog.naver.com/whdcks522/223368806181>

동영상: <https://www.youtube.com/watch?v=TFsxWrOTyIU>

플레이 스토어 링크: <https://play.google.com/store/apps/details?id=com.IssacCompany.OurPhoneGame>



싱글 선택 화면(아이콘 별로 세이더 배경 적용)



멀티 선택 화면(들어갈 미니게임의 아이콘을 미리 보여줌)



개싸움 화면(미니게임 중 하나, 머신러닝 적용)

```

public override void CollectObservations(VectorSensor sensor)//관찰할 정보, 5번 당 한번 호출
{
    //1. 수치형, 받아오는 데이터가 적을 수록 좋음
    if (gameObject.activeSelf) //쪽으면 정보 인식이 필요 없음
    {
        sensor.AddObservation(transform.position.x); //state size = 1    x,y,z를 모두 받아오면 size가 3이 됨
        sensor.AddObservation(transform.position.y);

        //가속을 더하기도 함
        sensor.AddObservation(rigid.velocity.x); //state size = 1
        sensor.AddObservation(rigid.velocity.y);

        if (player != null) //시작 한 순간, 빈 취급됨
        {
            //플레이어의 정보
            sensor.AddObservation(player.transform.position.x);
            sensor.AddObservation(player.transform.position.y);
            //각각의 거리
            sensor.AddObservation(curRange);
        }

        sensor.AddObservation(StepCount / (float)MaxStep); //진행한 스텝 비율    //state size = 1
    }
}

```

머신러닝 코드(센서를 통해 주변 물체 정보 확인)

```

private void OnTriggerEnter2D(Collider2D other)
{
    if ( enemy.isML)
    {
        if (other.transform.CompareTag("Outline")) //맵 밖으로 나가면 감점
        {
            AddReward(-1f);
            EndEpisode();
        }
    }
}

```

머신러닝 코드(낙사하면 감점)

```

private void OnCollisionEnter2D(Collision2D collision)//플레이어와 충돌하면 득점
{
    if (collision.gameObject == player && enemy.isML)
    {
        AddReward(10f); //점프했다 거기
        EndEpisode(); //이것만으로 초기화가 되진 않음
    }
}

```

머신러닝 코드(대상에게 닿으면 득점)

```

public override void OnEpisodeBegin()//EndEpisode가 호출됐을 때 사용됨(씬을 호출할 때는 통째로 삭제)
{
    //다음 학습을 위해 초기화하는 함수

    if (enemy.isML) //자신에 의해서만 발동
    {
        //Debug.Log("0rc");
        //자신의 코드에 의해서만 발동됨(플레이어가 초기화해도 이 코드는 호출 x, 충돌의 경우 둘다 적용됨)

        int enemyIndex = Random.Range(0, points.Length);

        while (true)//무작위 위치에서 다시 소환됨(다양한 위치에서 학습을 위한)
        {
            int playerIndex = Random.Range(0, points.Length);
            if (enemyIndex != playerIndex)
            {
                transform.position = points[enemyIndex].position;
                rigid.velocity = Vector2.zero;

                player.transform.position = points[playerIndex].position;
                break;
            }
        }
    }
}

```

머신러닝 코드(대상에게 닿거나, 낙사하는 경우 초기화)

4. 시즈 크로니클

플랫폼: Android

설명: 서로의 성을 공격하는 공성전 게임

블로그에 정리한 내용: <https://blog.naver.com/whdcks522/223495148084>

동영상: <https://www.youtube.com/watch?v=ENFspGrUqPA>

플레이 스토어 링크: https://play.google.com/store/apps/details?id=com.IssacCompany.Siege_Chronicle



선택 화면



버튼에서 드래그해서 사용(블루아카이브 따라함, 세이더 적용)



리더보드(파이어베이스 이용)