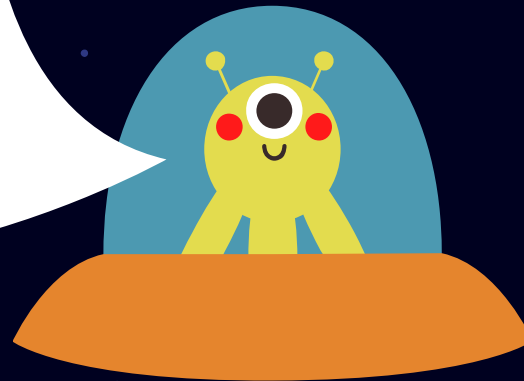


비보이 게임

# STAR WARS

이긴 어게인



# 목차.



게임배경

게임룰

제작과정

시연영상

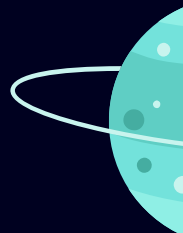
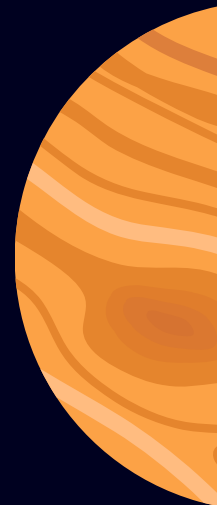
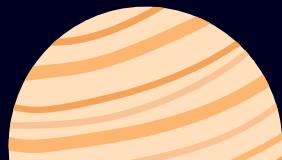
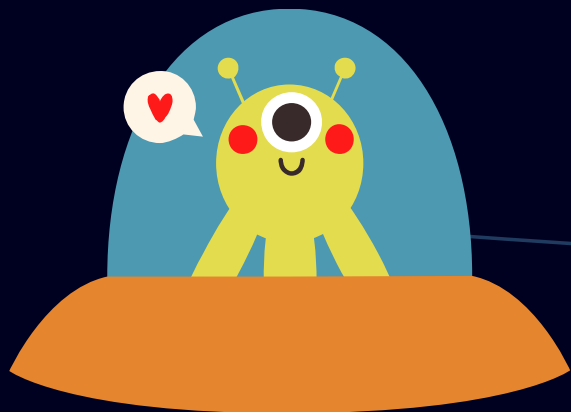
특징

팀원소개

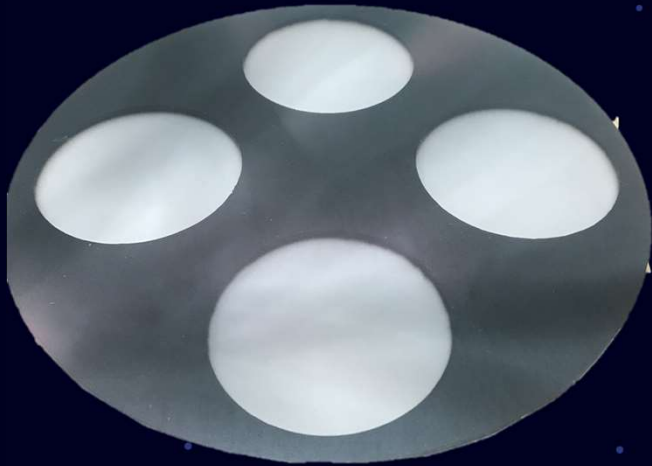
# | 게임배경

## 비보이, 우주자원을 구해라!

부족한 자원때문에 우주로 나온 비보이들!!  
UFO를 타고 반짝반짝 빛나는 광물들(빛)을 채취해 각자의 행성으로 돌아가자!



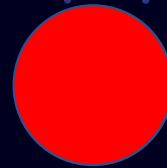
# | 게임룰



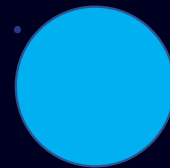
## 경기장 특징

- 경기장이 회전함
- 흰색부분에서 랜덤하게 빛이 나옴  
빛은 3초간 on  
1초간 off

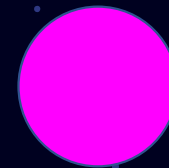
## EASY모드



-30점

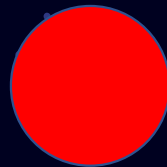


+10점

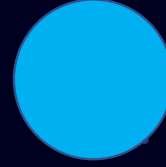


+20점

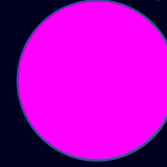
## HARD 모드



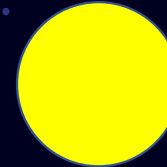
-30점



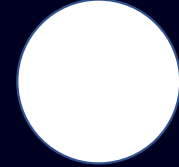
+10점



+20점



+30점

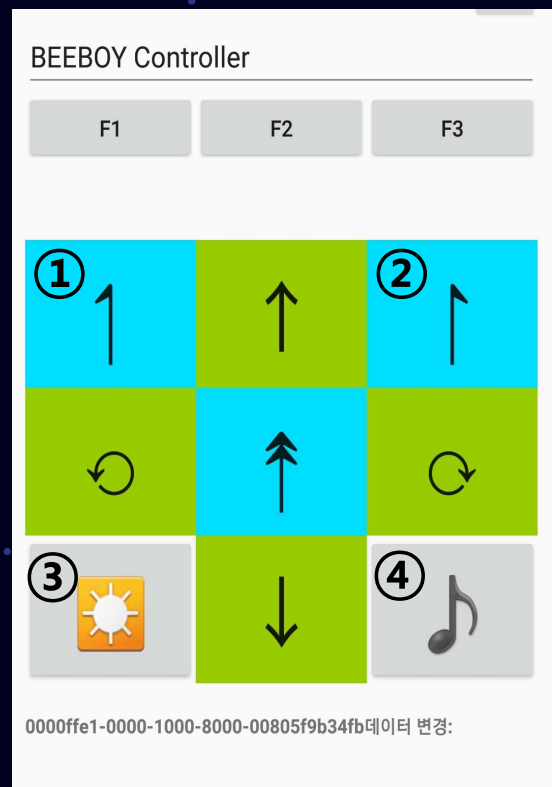


+50점



# | 게임룰

## 비보이 조작키



① 비보이에 ● 빛이  
나고 왼쪽으로 회전

② 비보이에 ● 빛이  
나고 오른쪽으로 회전

③ 비보이에 ● 빛이  
나옴

④ 비보이에 ● 빛이  
나옴

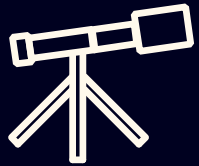


# I 게임룰

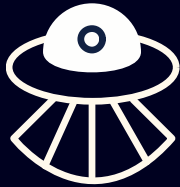
1. 제한시간은 3분, 개인전(1명씩 진행) 또는 배틀(2명씩 진행)으로 경기를 진행한다
2. 회전하는 무대 위 광물(빛)위에서 각각의 색에 맞는 버튼을 누른다. 색에 맞는 동작을 하였을 경우, 심판이 점수를 준다(=광물을 획득한다)
  - \* 비보이가 빨강색 빛에 50%이상 접촉 즉시 감점이다
  - \* 비보이가 해당 빛 안에 약 70%이상 들어와서 동작을 수행해야 점수를 준다
3. 제한시간 동안 많은 점수를 얻은(=광물을 획득한) 비보이가 승리한다
5. 단, 비보이가 경기장에서 떨어질 경우 4초 뒤에 올라올 수 있다



# | 제작과정\_구성품



턴테이블



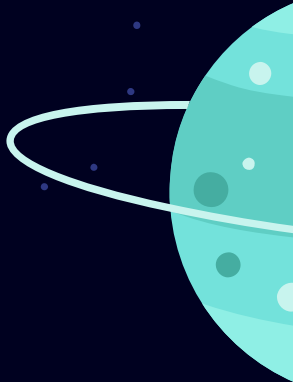
무대



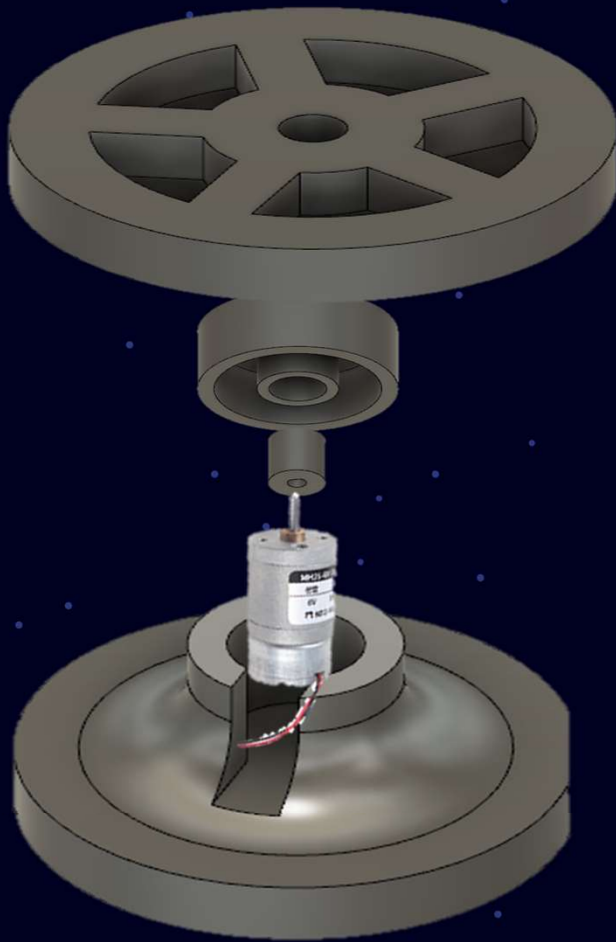
LED



비보이 로봇



# | 제작과정\_턴테이블



- 토크와 속도를 고려하여 모터를 선택

\* 6V 기어드 모터 사용

무부하 : RPM 20

최적부하 : RPM 16 토크 3 kg/cm

최대부하 : 토크 12 kg/cm

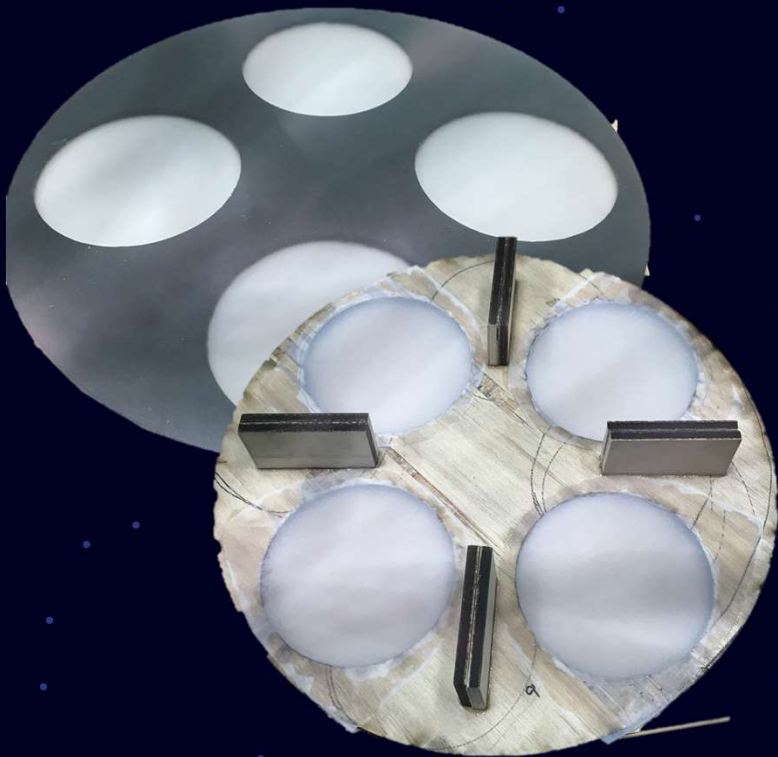
- 모터 크기를 고려하여 턴테이블 모델링을 함

\* 휴대 용이하도록 조립형식으로 설계

- 치수용 부품을 뺐은 뒤 본부품을 출력



# | 제작과정\_무대



- MDF와 나무합판 사용
- 레이저커팅기를 사용해 빛이 나올 구멍과 경기장 커팅
- LED 빛이 확산되도록 반투명 종이 부착
- 우주배경에 맞게 경기장을 검정색으로 도색
- 구멍에 걸리지 않도록 경기장위에 PVC필름 덮음

# | 제작과정\_LED

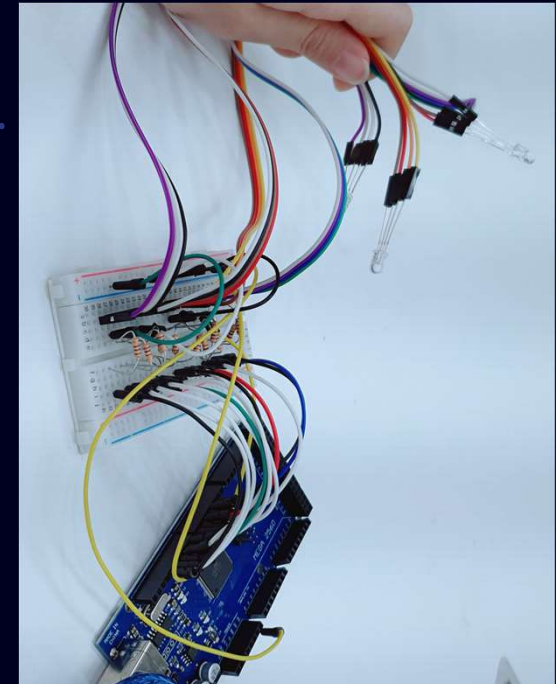
```
beeboy_game
void LED1(){
  int a;

  a=random(12); //0부터 11사이의 랜덤한 숫자 고름

  if(a==0){
    //경고사인
    analogWrite(ledRed1, 255);
    analogWrite(ledBlue1, 0);
    analogWrite(ledGreen1, 0); //빨
  }
  else if(a==1 || a==2 || a==3 || a==4 || a==5){
    analogWrite(ledRed1, 200);
    analogWrite(ledBlue1, 0);
    analogWrite(ledGreen1, 200); //노
  }
  else if(a==6 || a==7 || a==8){
    analogWrite(ledRed1, 255);
    analogWrite(ledBlue1, 127);
    analogWrite(ledGreen1, 0); //(핑크)
  }
  else if(a==9 || a== 10){
    analogWrite(ledRed1, 10);
    analogWrite(ledBlue1, 250);
    analogWrite(ledGreen1, 200); //파랑
  }
  else if(a==11){
```

```
    else if(a==11){
      analogWrite(ledRed4, 255);
      analogWrite(ledBlue4, 255);
      analogWrite(ledGreen4, 255);
    }
  }
  void loop() {
    // put your main code here, to run repeatedly:
    LED1(); LED2(); LED3(); LED4();
    delay(3000);

    analogWrite(ledRed1, 0);
    analogWrite(ledGreen1, 0);
    analogWrite(ledBlue1, 0);
    analogWrite(ledRed2, 0);
    analogWrite(ledGreen2, 0);
    analogWrite(ledBlue2, 0);
    analogWrite(ledRed3, 0);
    analogWrite(ledGreen3, 0);
    analogWrite(ledBlue3, 0);
    analogWrite(ledRed4, 0);
    analogWrite(ledGreen4, 0);
    analogWrite(ledBlue4, 0);
    delay(1000);
  }
}
```

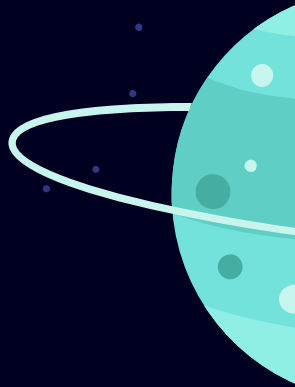


- 여러색을 표현할 수 있는 RGB LED 사용
- 많은 디지털 핀이 필요함, ARDUINO MEGA 사용

# | 제작과정\_BEEBOY



- FUSION 360 프로그램 사용
- UFO에 맞게 모델링 함
- 3D프린터로 출력 후 공차 조정



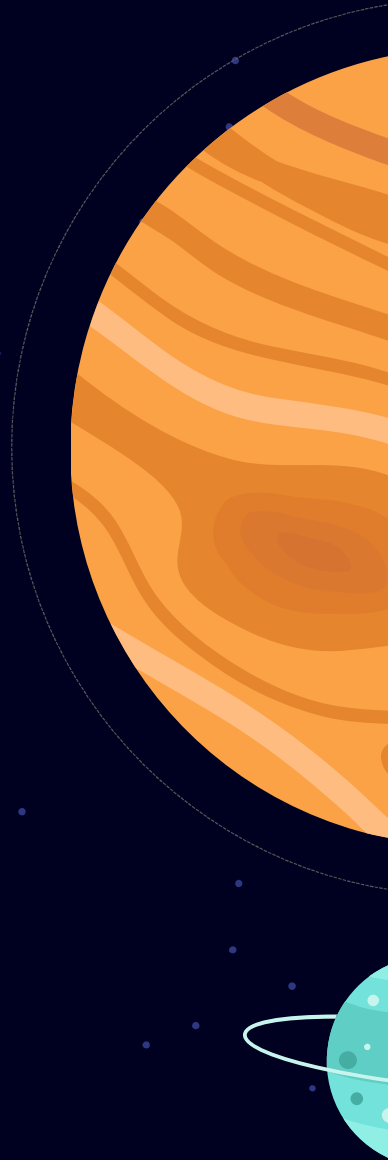
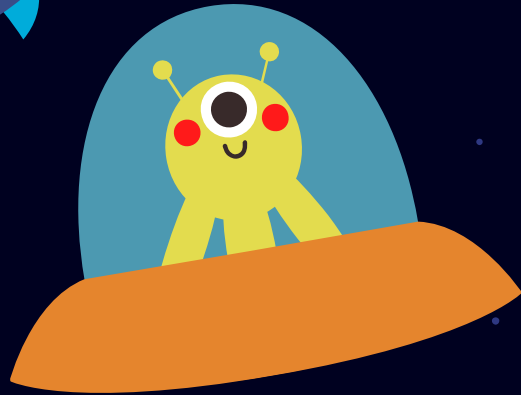
# | 제작과정\_BEEBOY

```
char btbuf;
float temp, volt;
int analogv;
void loop() {
  // put your main code here, to run repeatedly:
  if (mySerial.available())
  {
    digitalWrite(9, HIGH);
    digitalWrite(10, HIGH);
    digitalWrite(11, HIGH);
    btbuf = mySerial.read();
    Serial.println(btbuf);
    if (btbuf < 95) {
      if ('U' == btbuf)
      {
        Serial.println("forward");
        moving(HIGH, HIGH, 180, 180);
      }
      else if ('D' == btbuf)
      {
        Serial.println("backward");
        moving(LOW, LOW, 180, 180);
      }
      else if ('L' == btbuf)
      {
        moving(HIGH, HIGH, 60, 160);
      }
    }
  }
}
```

```
}
else if ('B' == btbuf)
{
  moving(LOW, HIGH, 180, 180); //파란색
  analogWrite(9, 0);
  analogWrite(10, 250);
  analogWrite(11, 00);
}
else if ('E' == btbuf) //노란색이 나옴
{
  moving(HIGH, LOW, 180, 180);
  analogWrite(9, 127);
  analogWrite(10, 0);
  analogWrite(11, 255);
}
else if ('S' == btbuf) //음악버튼 흰색
{
  analogWrite(9, 200);
  analogWrite(10, 200);
  analogWrite(11, 200);
}
else if ('A' == btbuf || 'a' == btbuf) //빛버튼 핑크색
{
  analogWrite(9, 230);
  analogWrite(10, 50);
  analogWrite(11, 10);
}
```

- 각각의 버튼을 누르면 얼굴에서 색이 나오도록 코딩

# | 시연영상



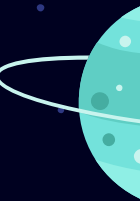
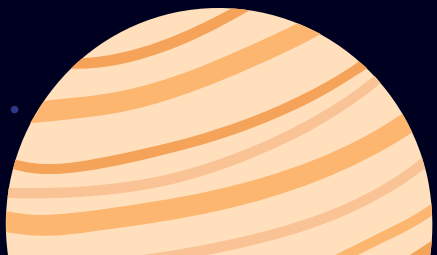
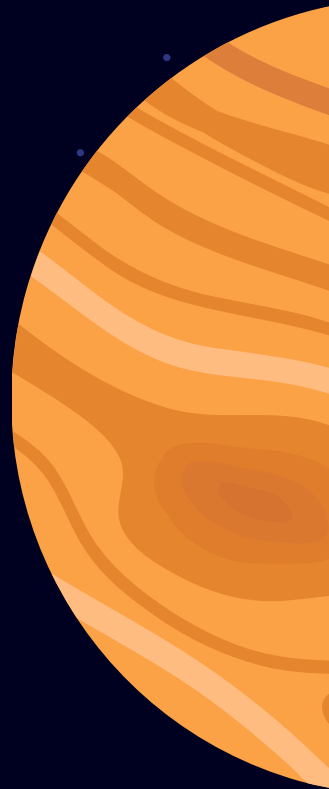
# |특징

간편한 조립의 3단 구성

난이도 조절 가능

좁은 공간에서도 게임 가능

1인~2인용 게임

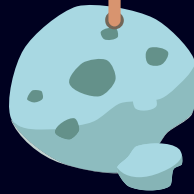


# | 팀원소개



이도원

- 기계공학부
- 무대 설계
- pt 발표



김찬우

- 기계공학부
- 턴테이블 설계
- 비보이 모델링



박성아

- 정보컴퓨터공학부
- LED 코딩
- 비보이 코딩



Q&A