Version1.2

Feng Jitao 1063750

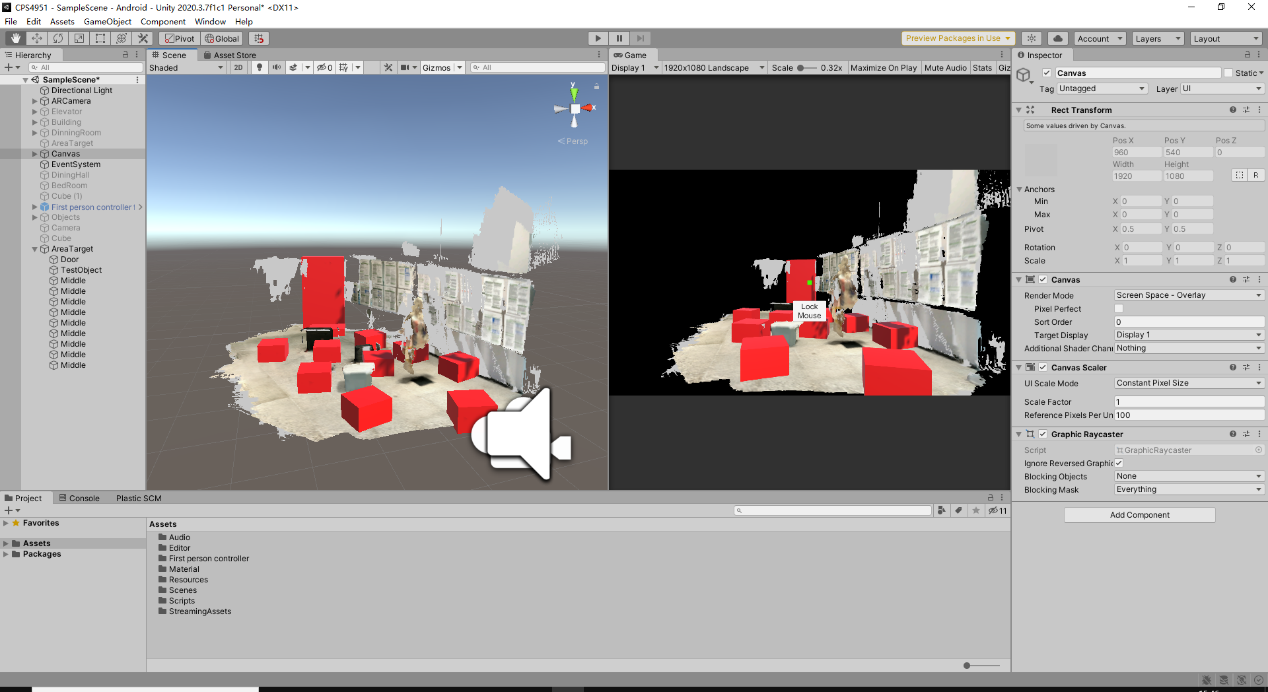
Chen Shiyi 1063701

Wang Yudong 1064076

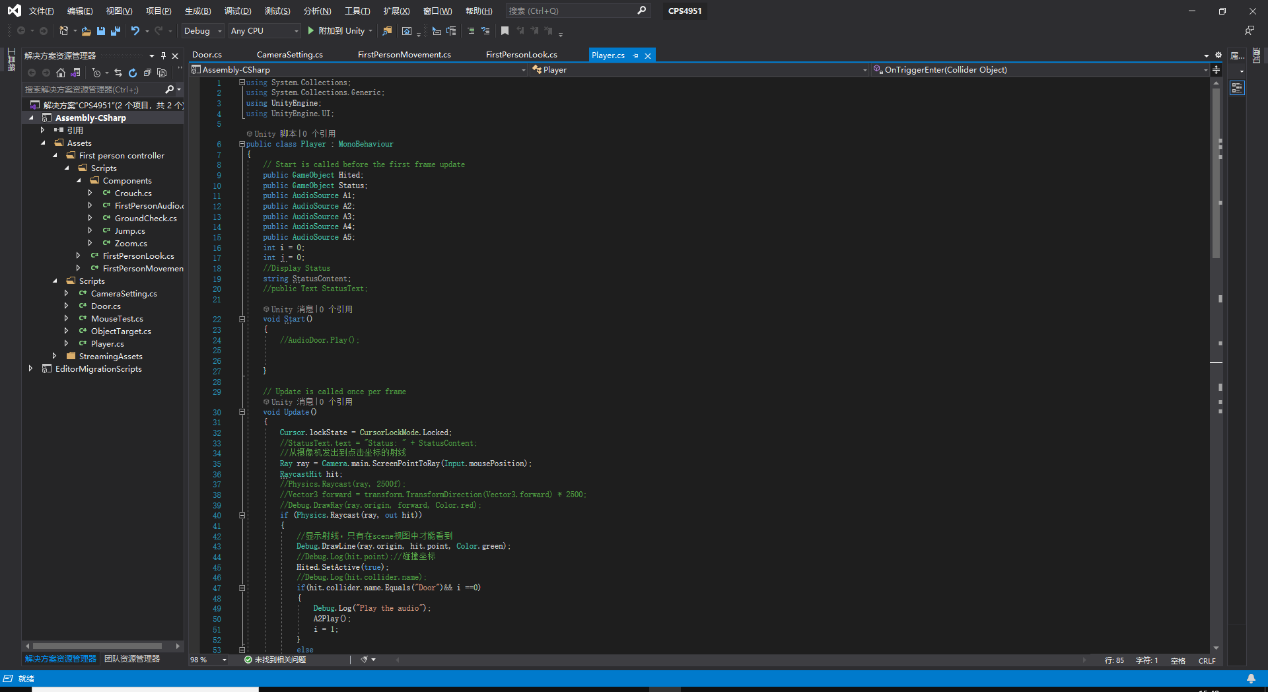
Yuan Huajun 1064178

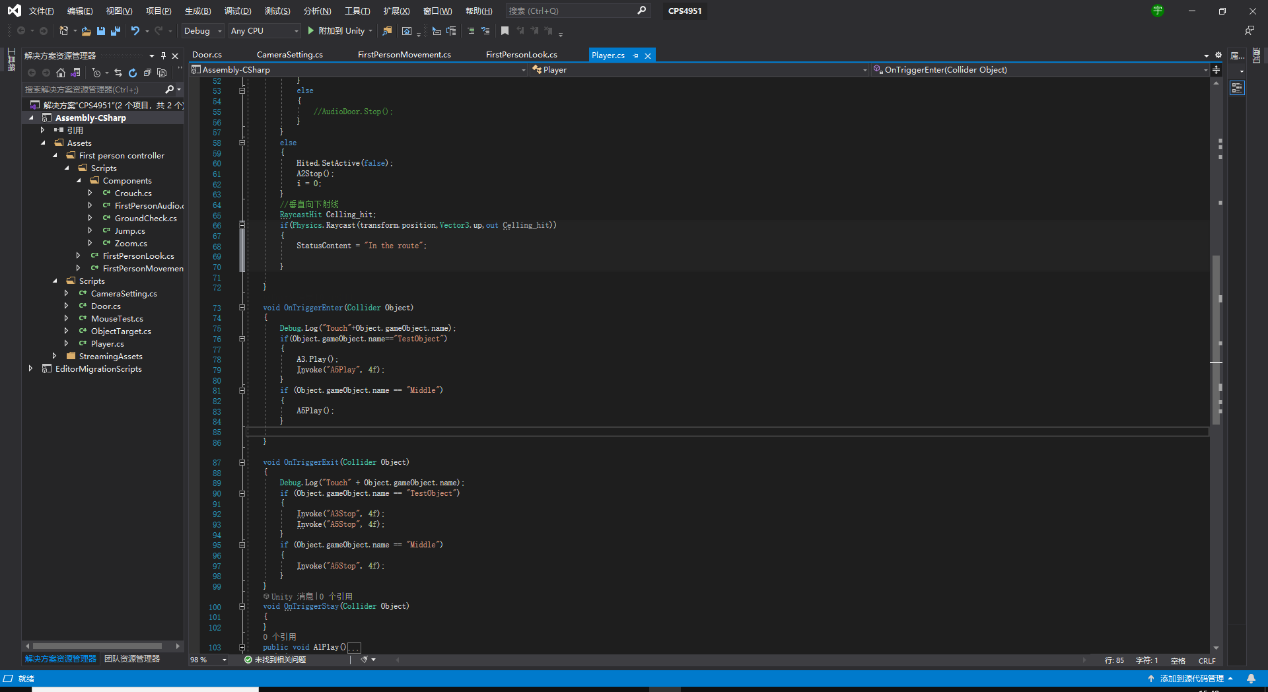
## Unity3D code

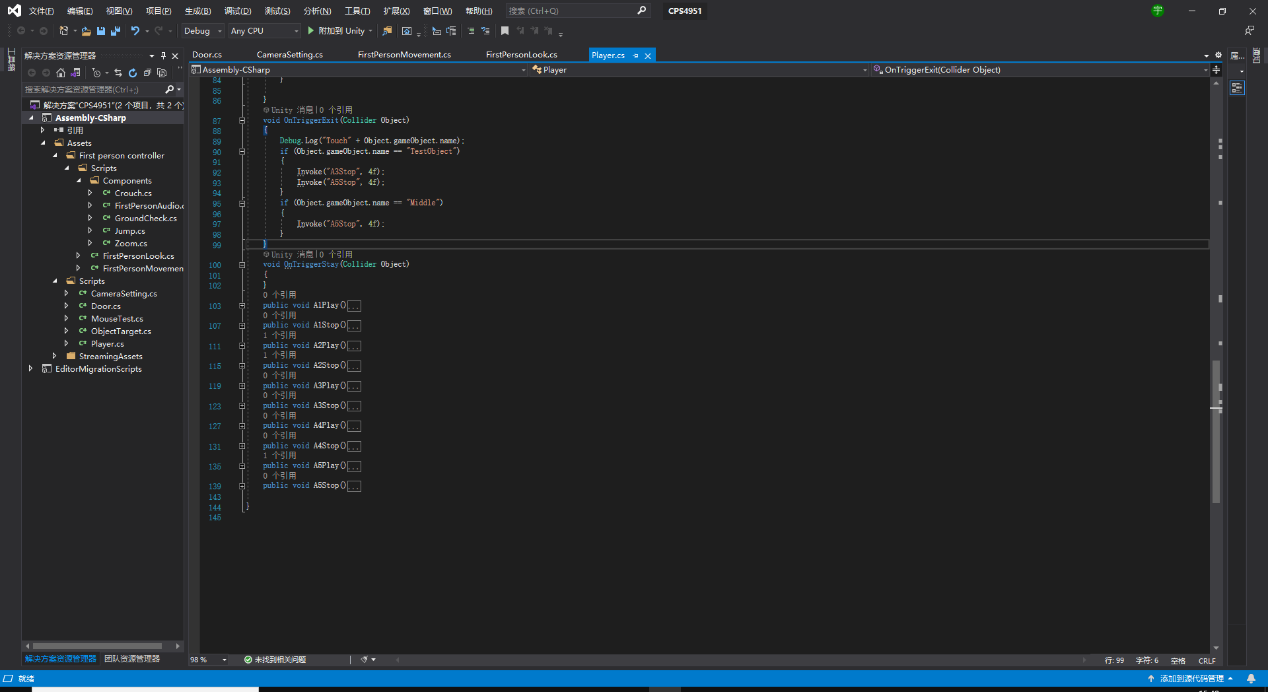
This application aims to simulated the data of user in specific scenario like bedroom, living room, bathroom, etc; The strategy of this application is using the camera of user to detect the environment, the application will recognize the specific scenario and set the location data of users into the unity local scene as a player, then it is able to implement some functions like vioce assistant and tactility assistant to navigate the player to some place in unity scene. Since the scenario of user is simulated in the game scene, the same assistant function is also helpful for the user in real environment.



Here is C# code of event response:







## Arduino code

In this part, the code mainly have the ultrasonic sounding function and alarm function.

Here is the code:

#include <SoftwareSerial.h>

#include <Arduino.h>

#define Trig 2 //引脚Tring 连接 IO D2

#define Echo 3 //引脚Echo 连接 IO D3

float cm; //距离变量

float temp; //

char serialData;

//SoftwareSerial WIFI(8, 9);

bool stringComplete = false;

String str = "";

String sig = "";

const static int KEY\_ON PROGMEM = 1;

const static int KEY\_OFF PROGMEM = 0;

int buzzerPin = 4;

int mode = 1;

int count = 0;

void setup() {

Serial.begin(9600);

//WIFI.begin(9600);

pinMode(buzzerPin, OUTPUT);

digitalWrite(buzzerPin, HIGH);

// pinMode(WIFIPin, OUTPUT);

// digitalWrite(WIFIPin, HIGH);

pinMode(Trig, OUTPUT);

pinMode(Echo, INPUT);

}

void loop() {

//声纳测距

//给Trig发送一个低高低的短时间脉冲,触发测距

digitalWrite(Trig, LOW); //给Trig发送一个低电平

delayMicroseconds(2); //等待 2微妙

digitalWrite(Trig,HIGH); //给Trig发送一个高电平

delayMicroseconds(10); //等待 10微妙

digitalWrite(Trig, LOW); //给Trig发送一个低电平

temp = float(pulseIn(Echo, HIGH)); //存储回波等待时间,

//pulseIn函数会等待引脚变为HIGH,开始计算时间,再等待变为LOW并停止计时

//返回脉冲的长度

//声速是:340m/1s 换算成 34000cm / 1000000μs => 34 / 1000

//因为发送到接收,实际是相同距离走了2回,所以要除以2

//距离(厘米) = (回波时间 \* (34 / 1000)) / 2

//简化后的计算公式为 (回波时间 \* 17)/ 1000

cm = (temp \* 17 )/1000; //把回波时间换算成cm

Serial.print("Echo =");

Serial.print(temp);//串口输出等待时间的原始数据

Serial.print(" | | Distance = ");

Serial.print(cm);//串口输出距离换算成cm的结果

Serial.println("cm");

delay(100);

//蜂鸣器

if(cm <= 60 && cm >= 40){

digitalWrite(buzzerPin, LOW);

delay(1000);

digitalWrite(buzzerPin, HIGH);

delay(1000);

}

else if(cm <39 && cm >=20){

digitalWrite(buzzerPin, LOW);

delay(500);

digitalWrite(buzzerPin, HIGH);

delay(500);

}

else if(cm <19){

digitalWrite(buzzerPin, LOW);

delay(100);

digitalWrite(buzzerPin, HIGH);

delay(100);

}

}

In the Arduino module, the main functions we have implemented at present are ultrasonic ranging and warning. These two functions are mainly used to detect obstacles at close range. When the user holds the guide stick in his hand, he can judge the distance from the obstacle by the frequency of the beep of the buzzer. But the disadvantage is that it can't detect some obstacles with too small surface areas. And it can only be used to detect a relatively small area (about 30°).

## Device design



Figure1. Guidance device



Figure2. Intelligent obstacle avoidance bar