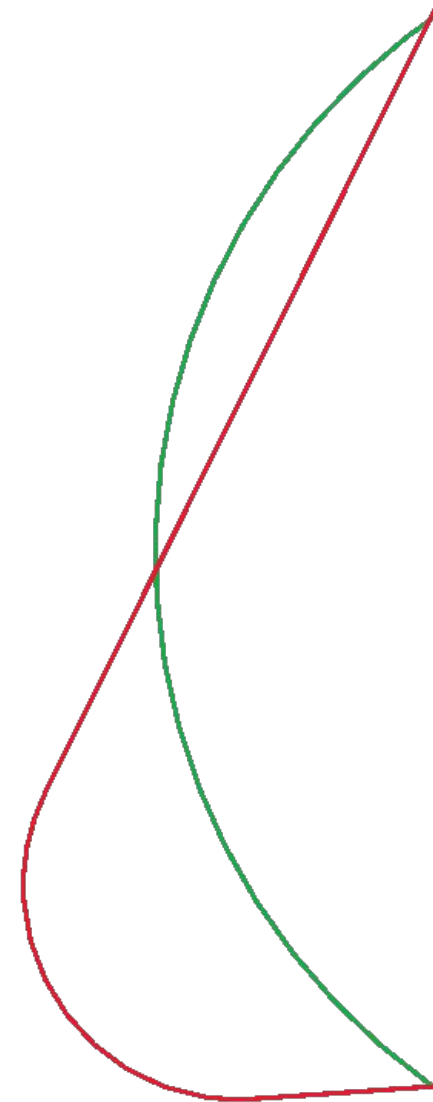


Unity 導入 第三方api chatgpt

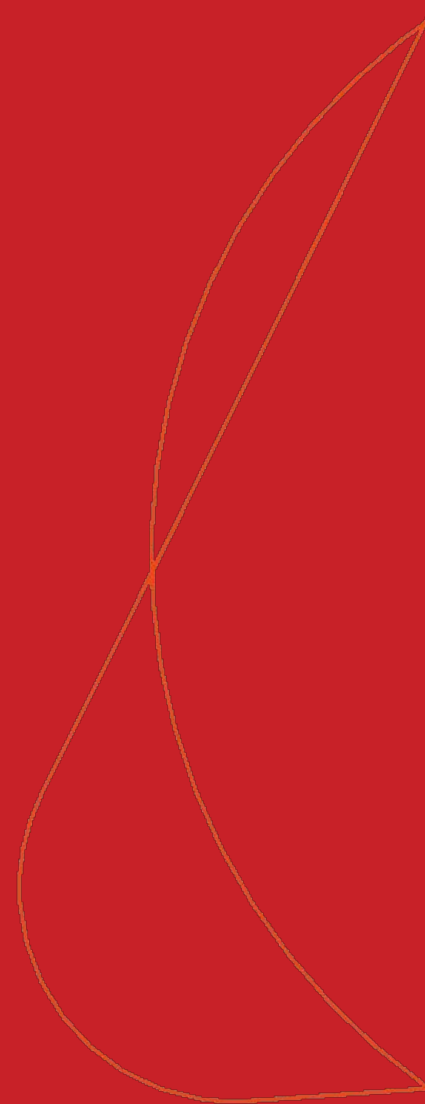
簡報人:林瑋琮

2023/06/02



OUTLINE

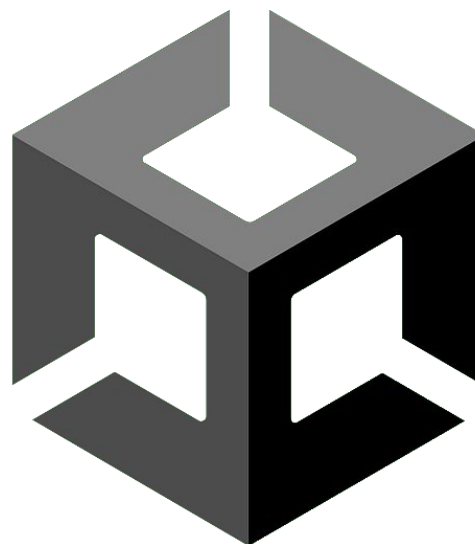
前言
實作流程
實作方法



前言

鑒於AI技術的崛起，而遊戲又是人們生活中重要的休閒娛樂，因此本計畫預計將ChatGpt導入unity遊戲引擎。

1. U3d 2022.2.1f1
2. OpenAI SDK
3. Oculus Voice SDK



實作流程

1

環境建置&建構介面

2

串接 Open AI API

3

串接 Oculus Voice Sdk

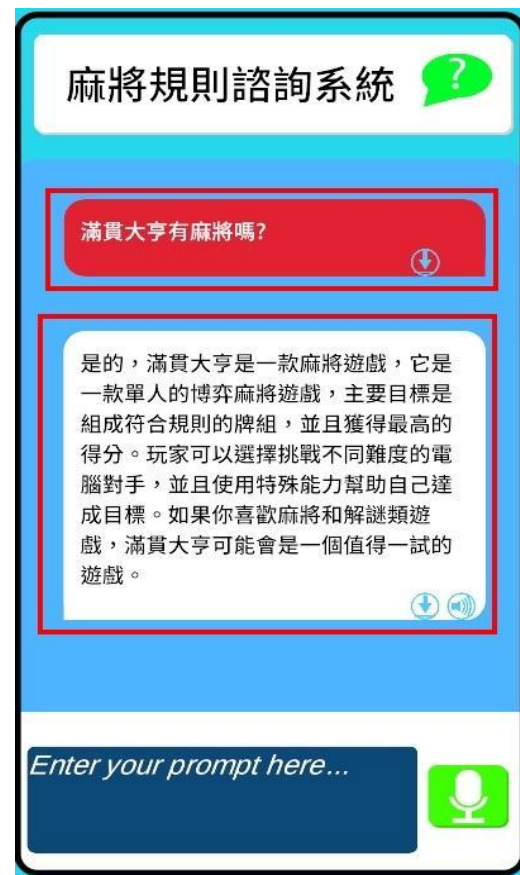
4

問題排除&簡報製作

實作方法

建立訊息泡泡物件

1. 我的訊息泡泡
2. 機器人泡泡

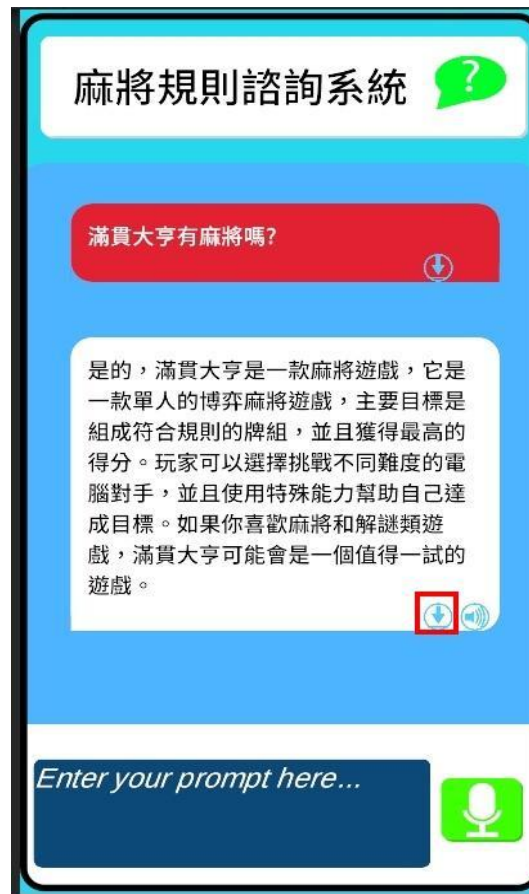


實作方法

複製訊息泡泡文字

1. 跨平台的剪貼簿功能安裝
2. 可將GPT回傳的訊息保存

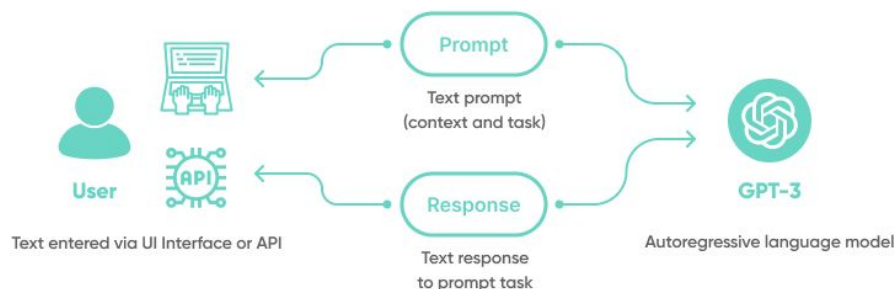
```
public void CopyToClipboardCallback()  
{  
    GUIUtility.systemCopyBuffer = messageText.text;  
}
```



實作方法

串接OpenAI API

1. 實作Clnet & OpenAI 交互
2. Clnet可指定訊息的角色: System or User
3. Gpt交互物件ChatRequest
 - a. 選擇使用的模型: GPT3_5_Turbo
 - b. 模型人性化設定: Temperature
4. Client發送要求後, 用非同步等待OpenAI的回應



實作方法

系統風格設定檔安裝

1. 設定系統規則在string.xml檔
2. 每次對GPT發送請求前，訊息的前綴放讀進來的規則

```
private void ReadXML()
{
    //檔案要放在Asset底下
    string filePath = Path.Combine(Application.dataPath, "string.xml");
    // 載入XML檔案
    XmlDocument xmlDoc = new XmlDocument();
    xmlDoc.Load(filePath);
    // 取得根元素
    XmlElement root = xmlDoc.DocumentElement;
    // 指定要讀取的key
    string keyToRead = "GameIntroduction";
    // 尋找對應的Setting元素
    XmlNodeList settings = root.SelectNodes($"Setting[@key='{keyToRead}']");
    if (settings.Count > 0){
        // 取得value值
        value = settings[0].Attributes["value"].Value;
    }
}
```



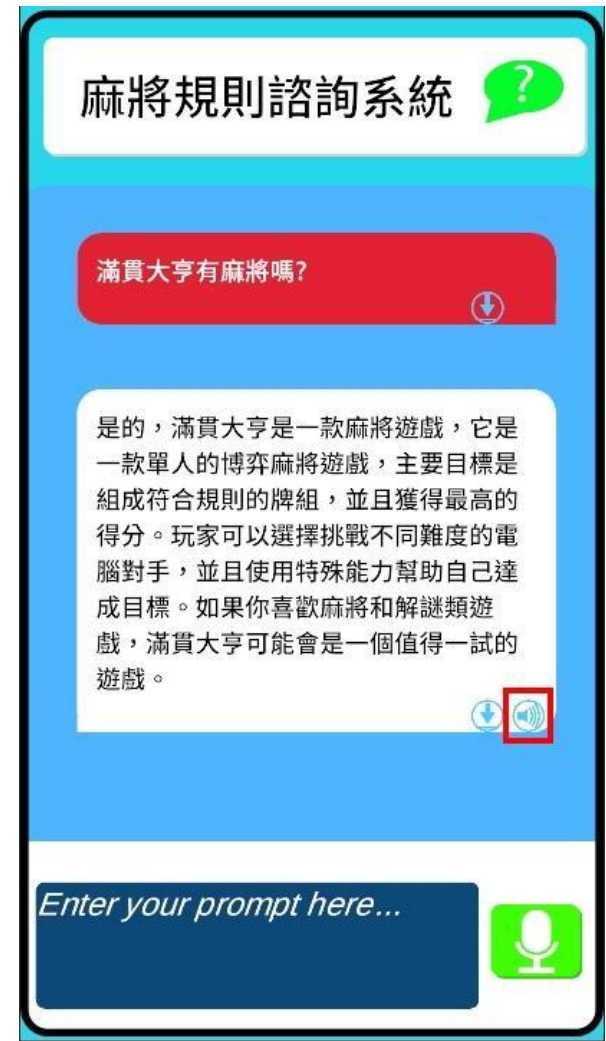
實作方法

語音朗讀文字功能安裝

1. 導入Oculus voice sdk
2. 建立TTS Speaker物件
3. 將訊息傳遞給TTS Speaker物件

```
void Start()  
{  
    DiscussionBubble.onVoiceButtonClicked += Speak;  
}
```

```
private void Speak( string message )  
{  
    string[] messages = message.Split( '.' );  
    speaker.StartCoroutine( speaker.SpeakQueuedAsync( messages ) );  
}
```



結論

GPT在遊戲中有一些幫助，例如輕鬆幽默的小笑話、劇情遊戲的NPC對話、引導學生學習等功能。然而，目前GPT的token限制是4096，若有大量的互動訊息可能會較不精準。因此，未來若能打破token的限制，gpt有機會能讓遊戲變得越來越智能。

Thank You

