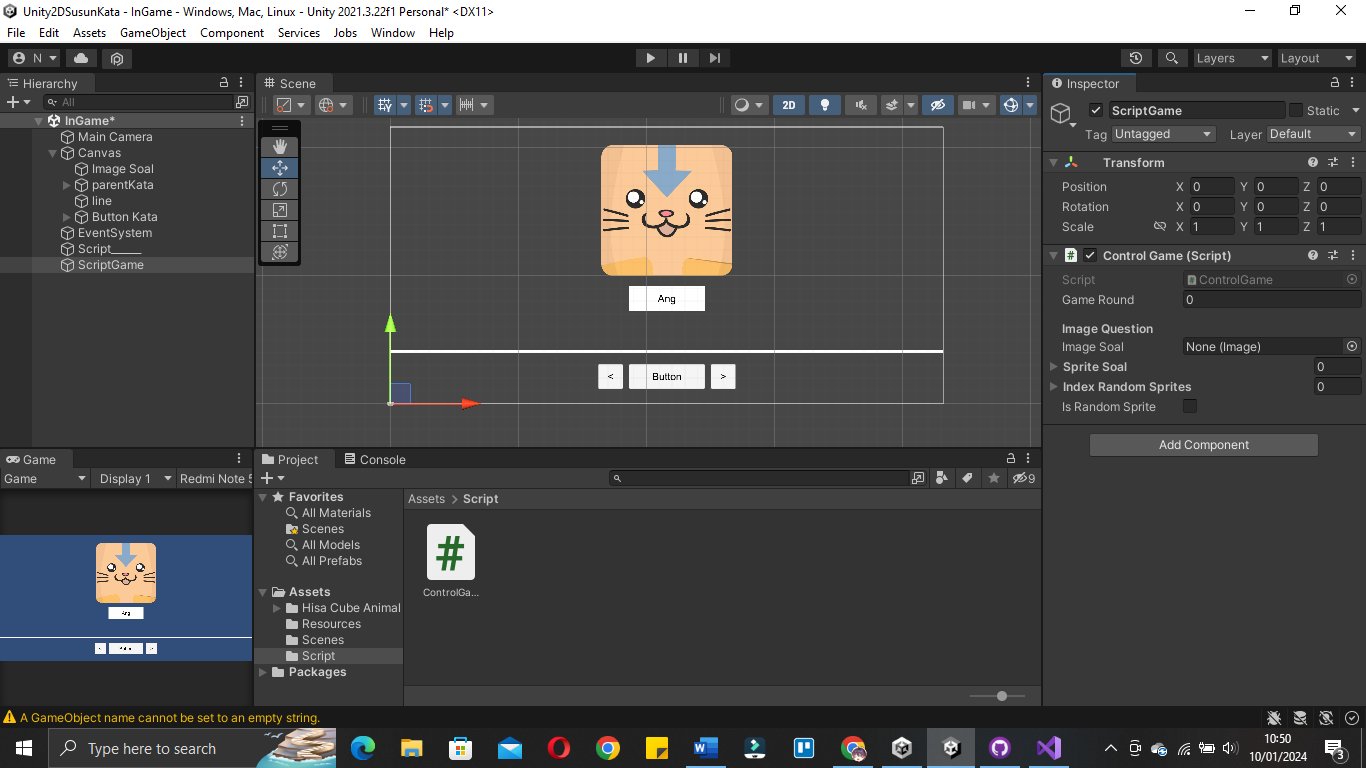
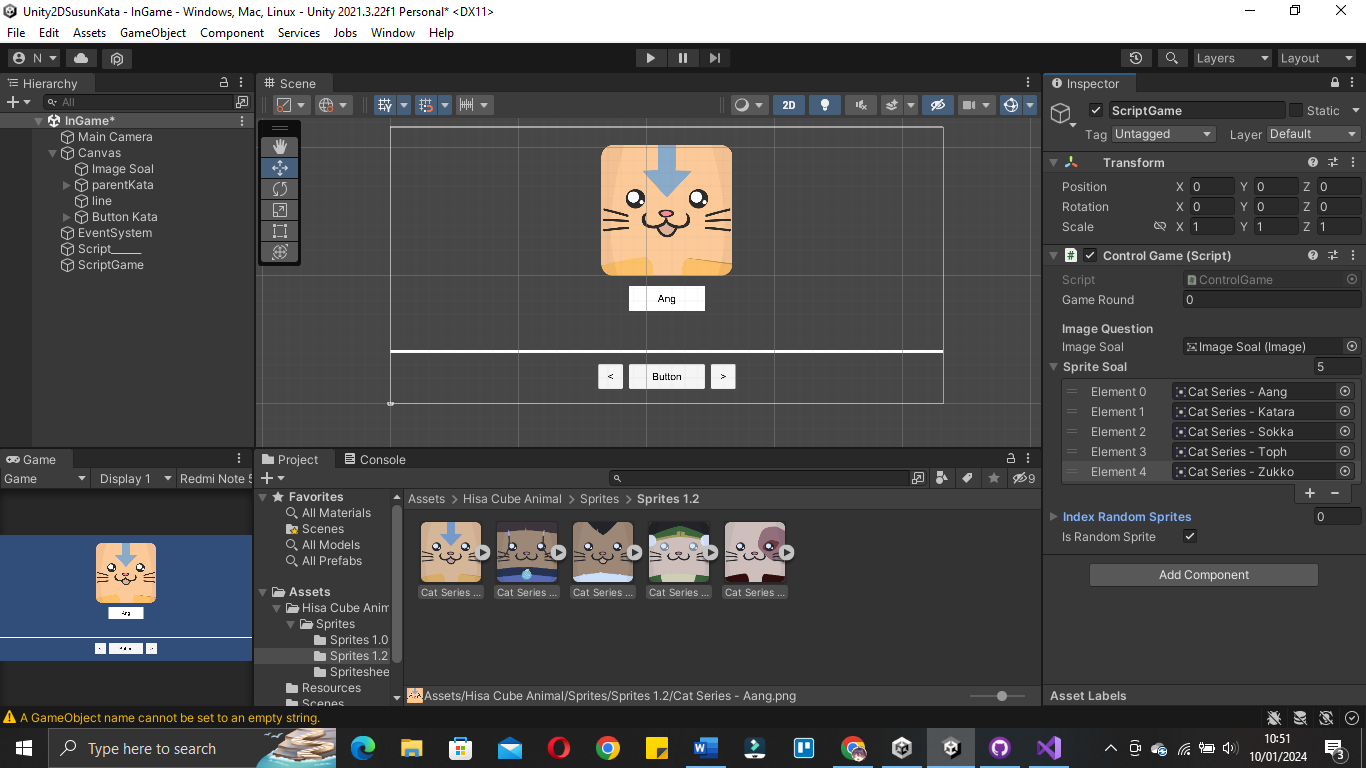
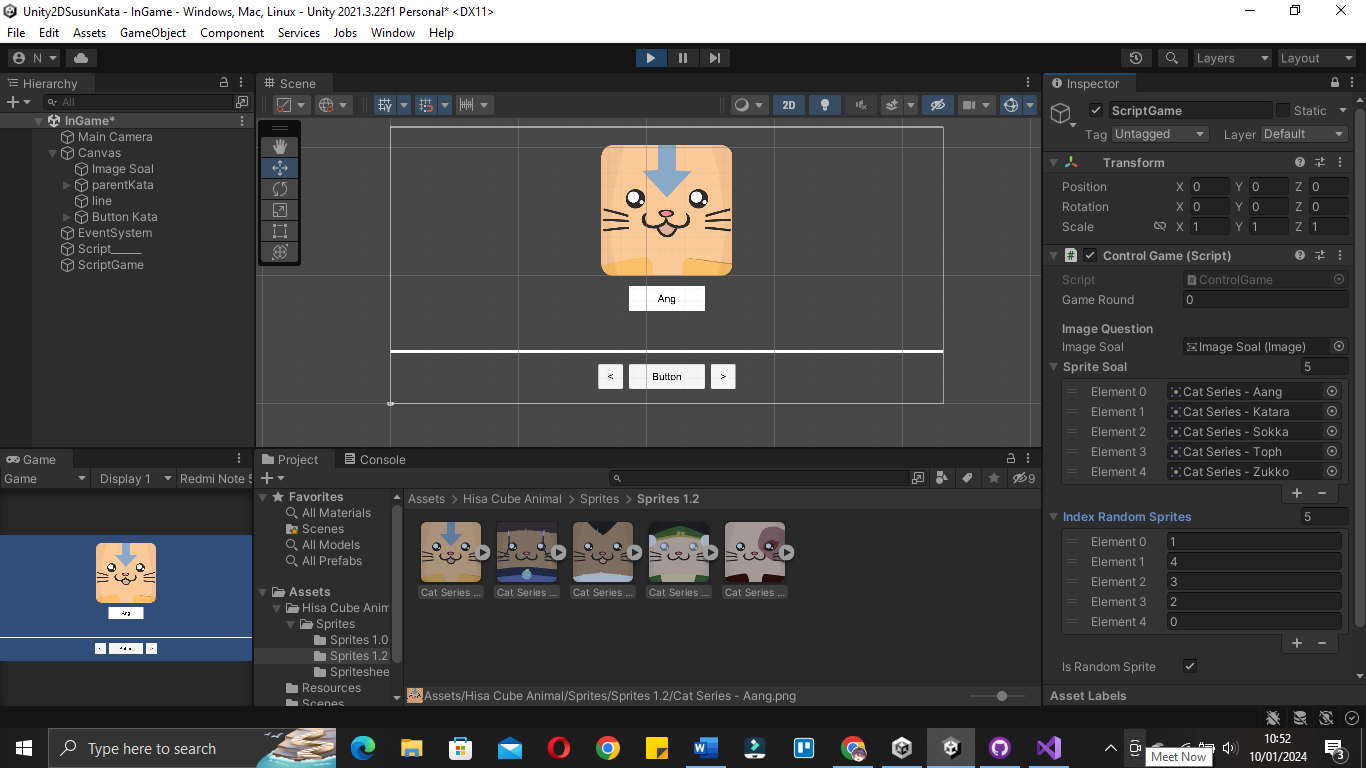
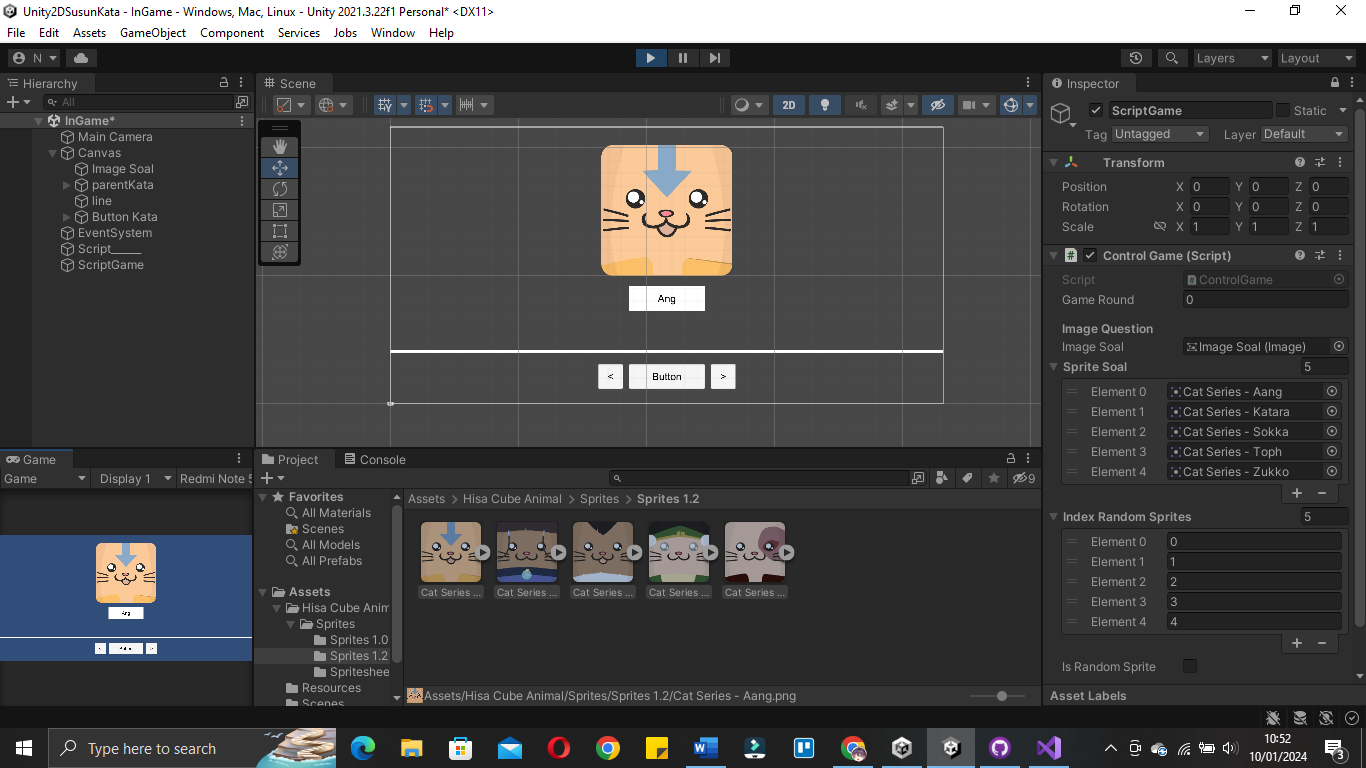


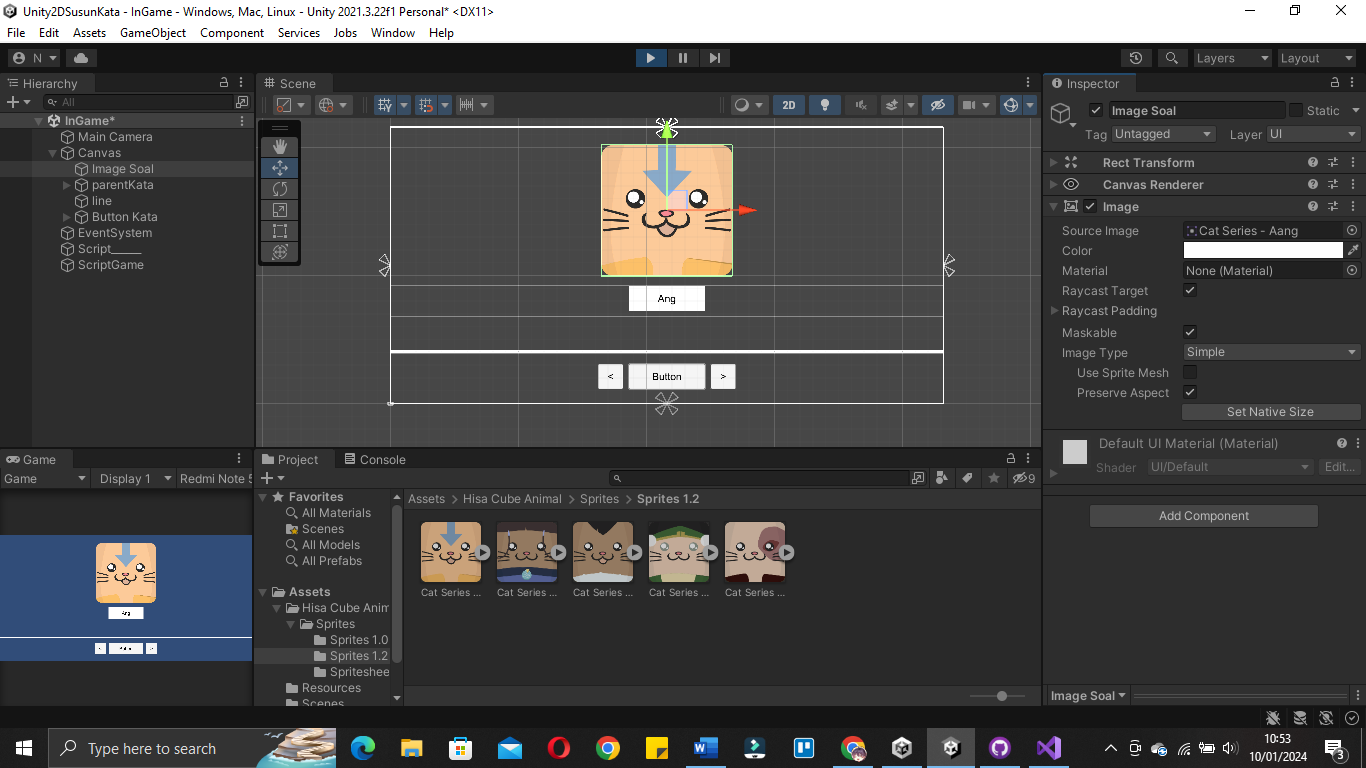
|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites);  }  }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |



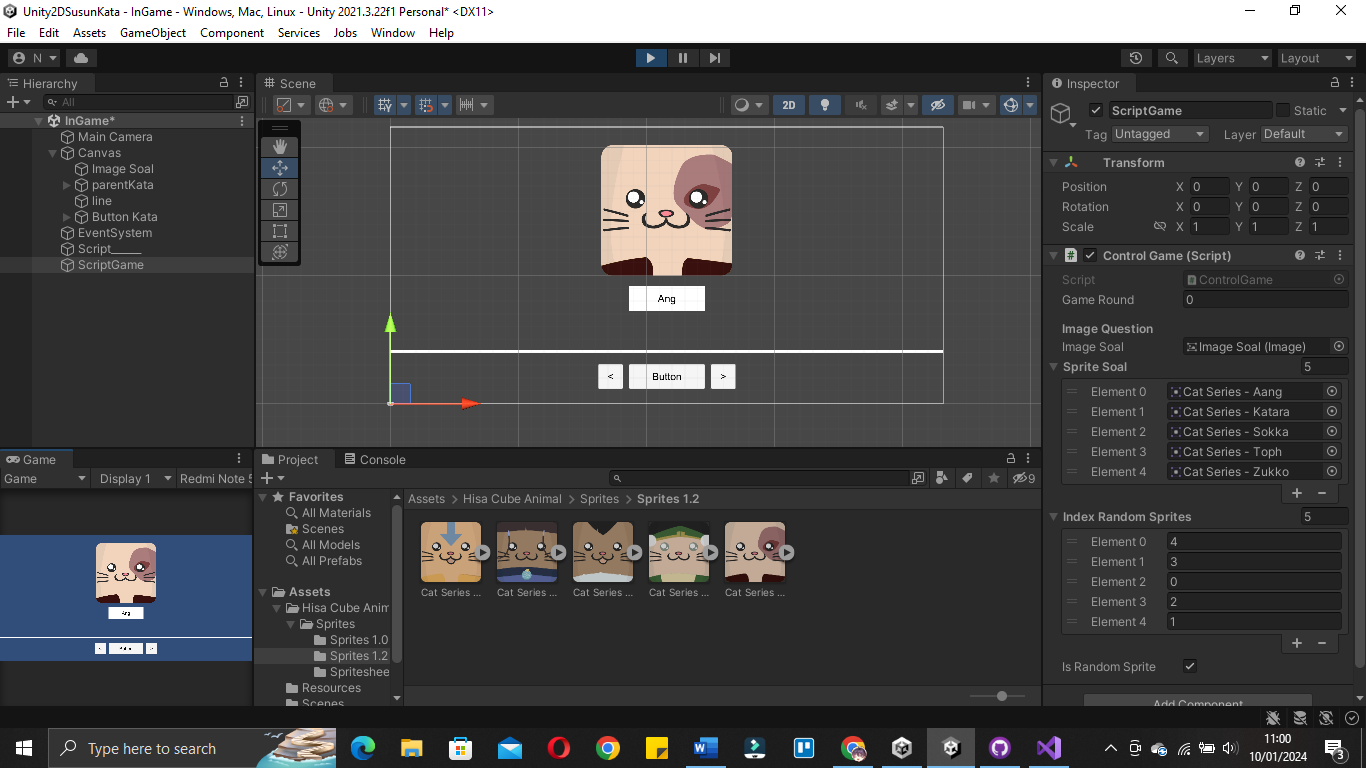




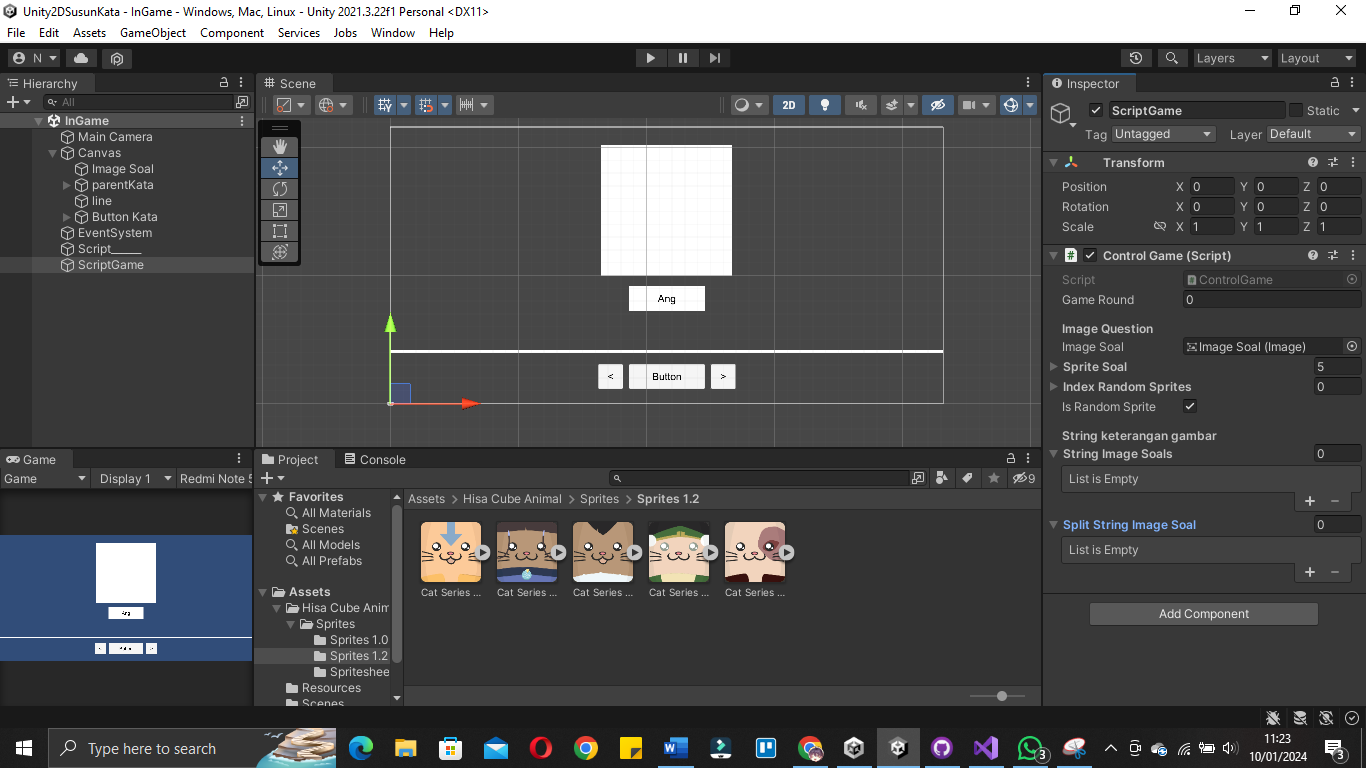


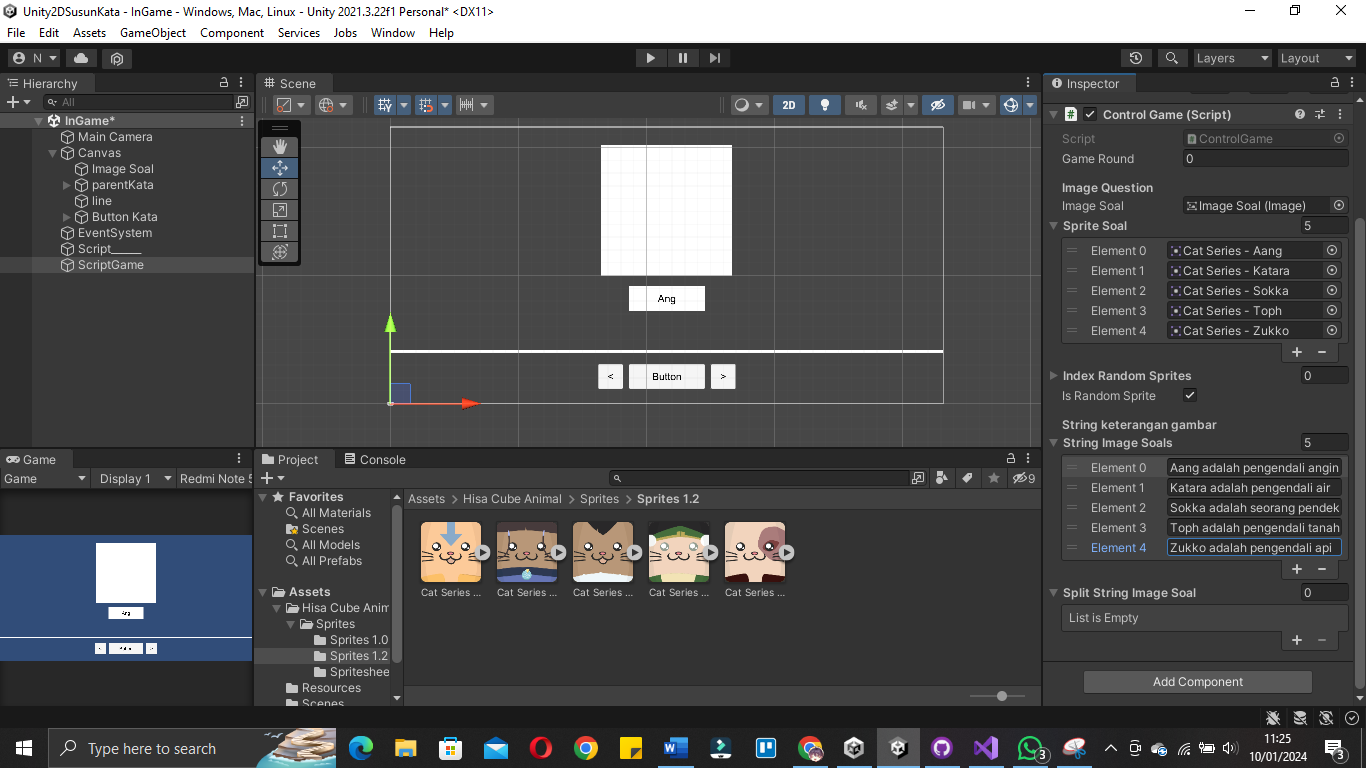


|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak value  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |

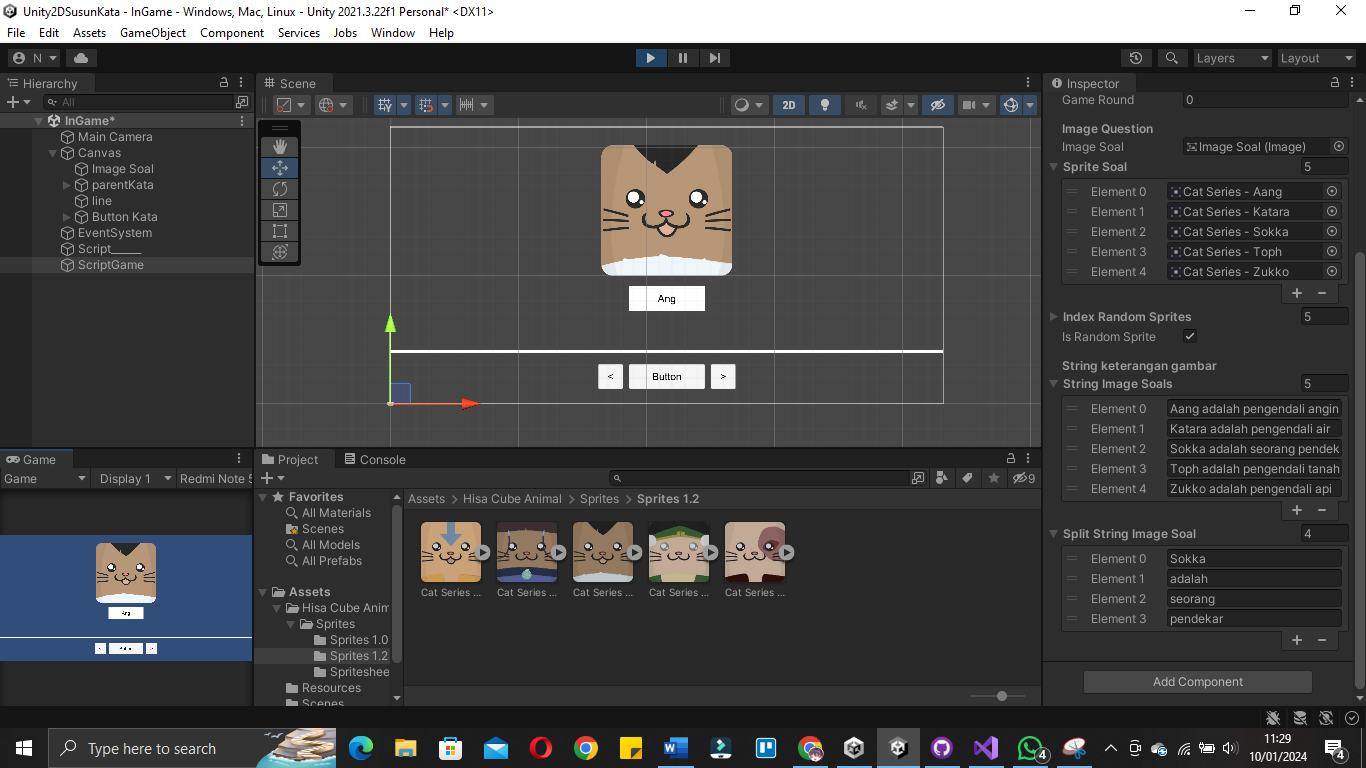


|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |

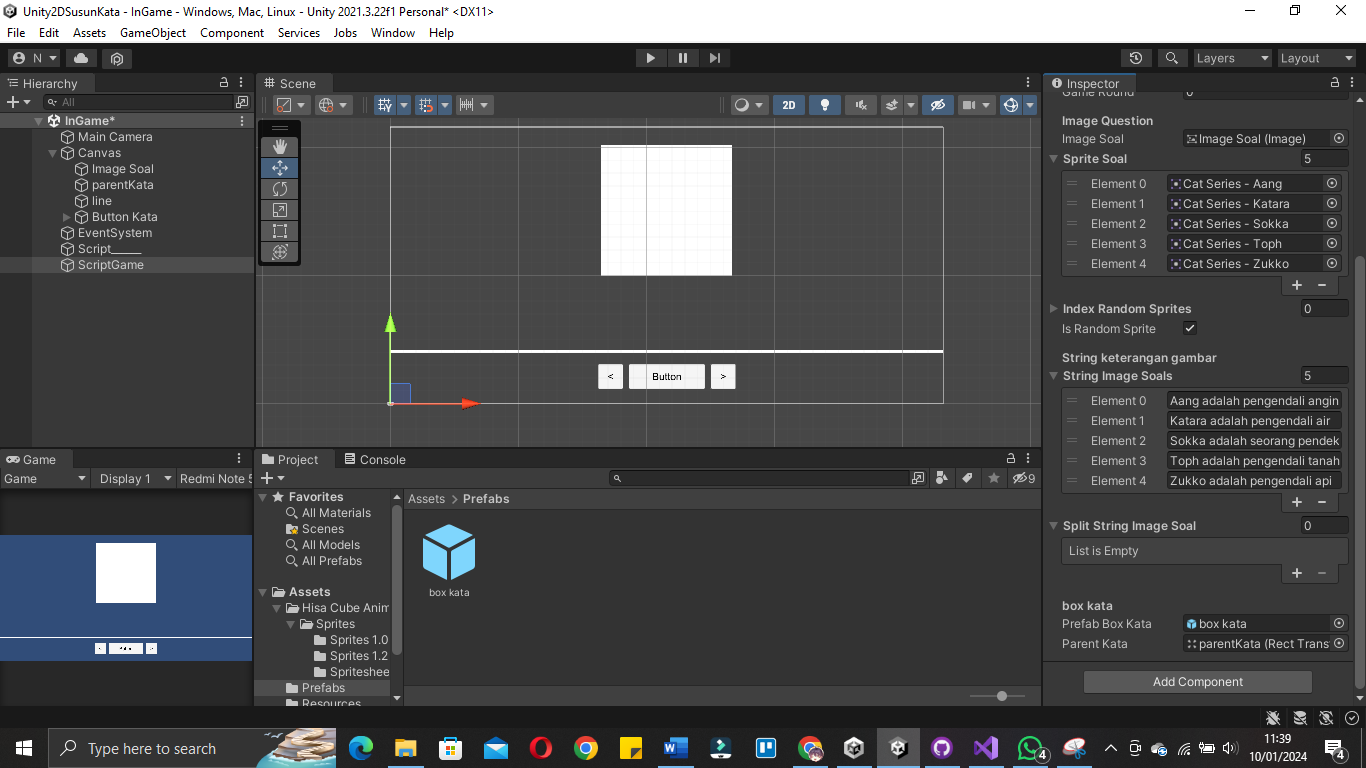


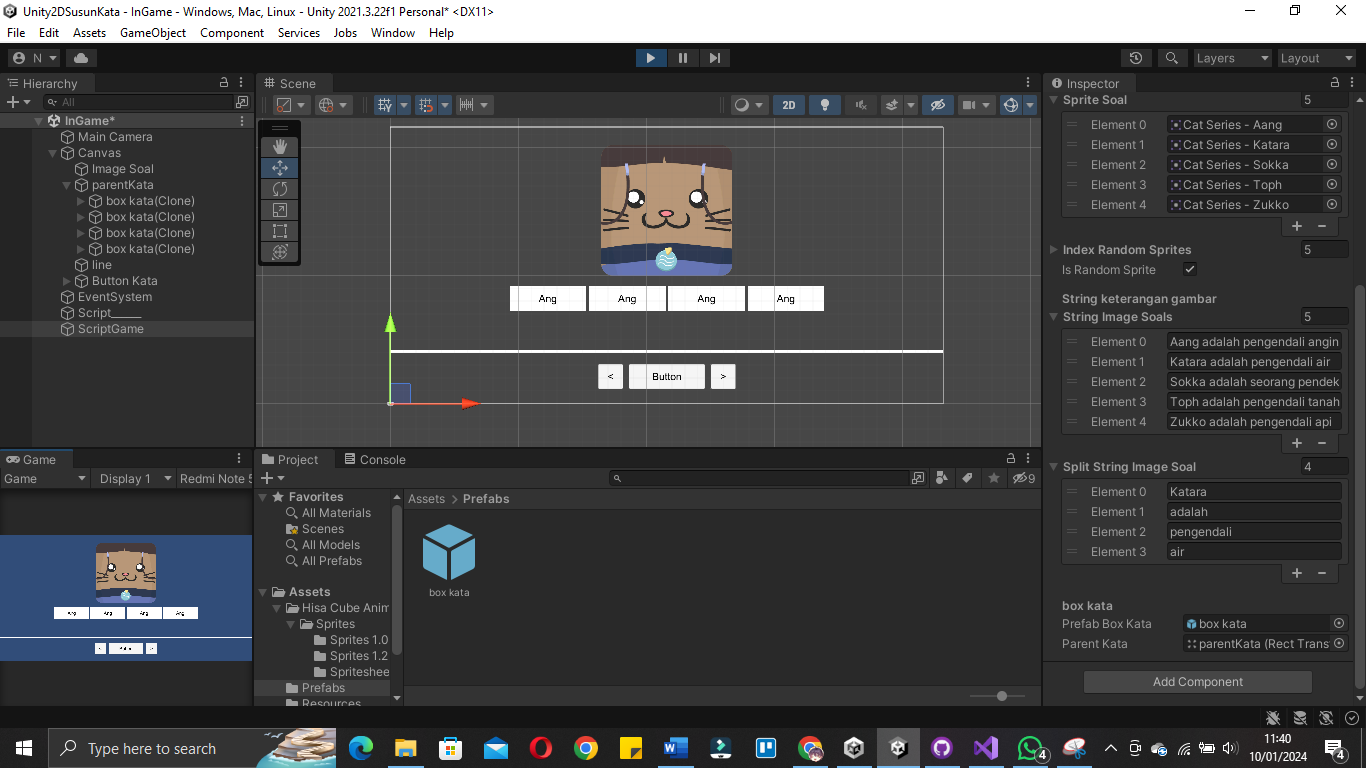


|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  //Implementasi keterangan gambar  splitStringImageSoal = stringImageSoals[indexRandomSprites[gameRound]].Split(' ');//dipotong dengan acuan spasi  }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |

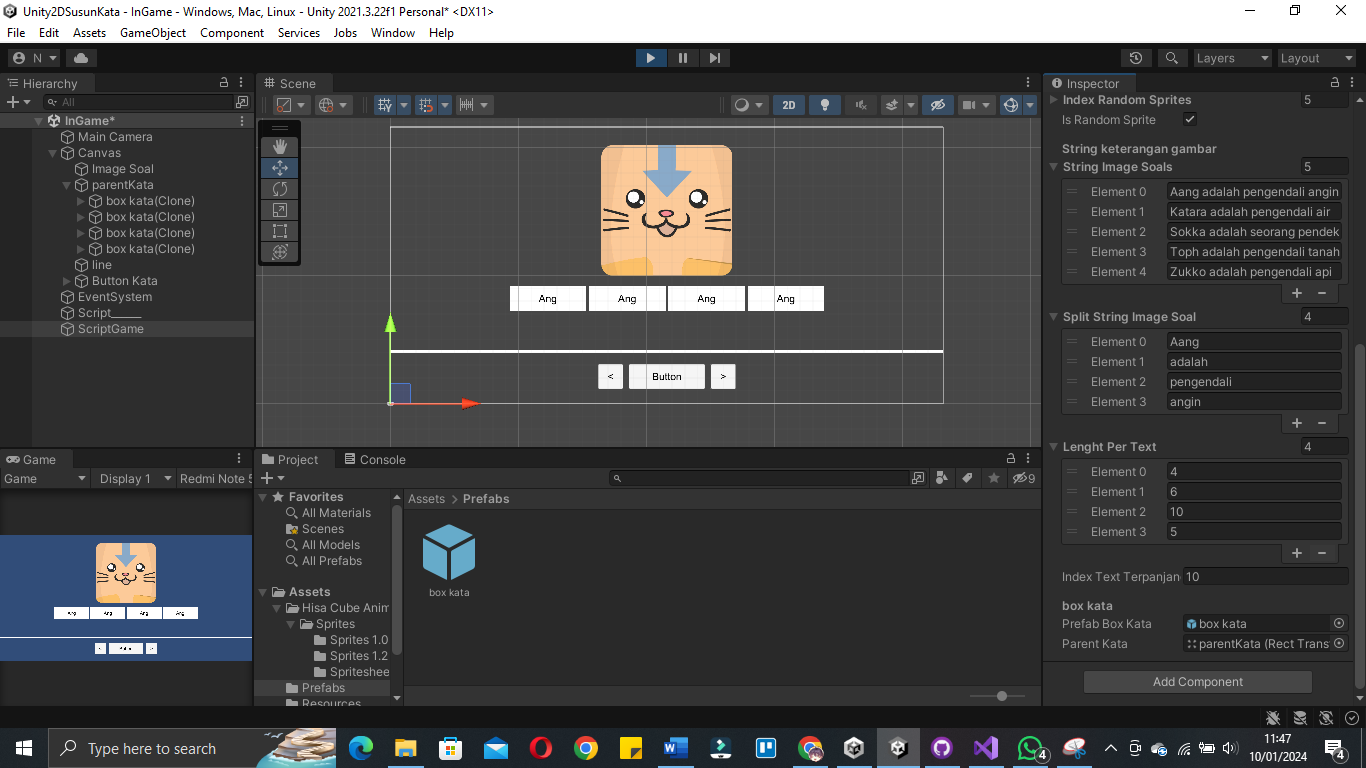


|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  [Header("box kata")]  public GameObject prefabBoxKata;  public Transform parentKata;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  //Implementasi keterangan gambar  splitStringImageSoal = stringImageSoals[indexRandomSprites[gameRound]].Split(' ');//dipotong dengan acuan spasi  //respon box  for(int i=0; i<splitStringImageSoal.Length; i++)  {  GameObject cloneBoxKata = Instantiate(prefabBoxKata);//respawn  cloneBoxKata.transform.SetParent(parentKata);//set parent  }    }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |



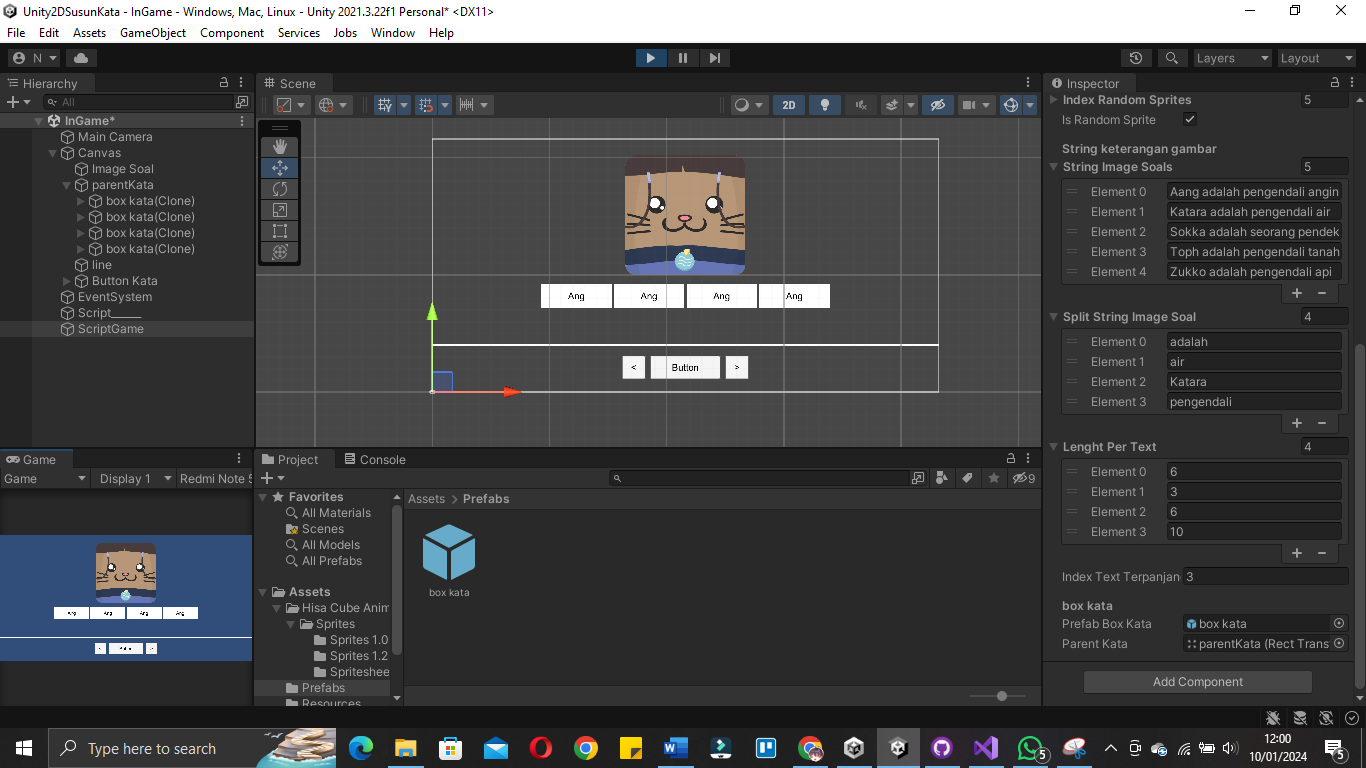


|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  public int[] lenghtPerText;  public int indexTextTerpanjang;  [Header("box kata")]  public GameObject prefabBoxKata;  public Transform parentKata;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  //Implementasi keterangan gambar  splitStringImageSoal = stringImageSoals[indexRandomSprites[gameRound]].Split(' ');//dipotong dengan acuan spasi  lenghtPerText = new int[splitStringImageSoal.Length];  for (int i = 0; i <splitStringImageSoal.Length; i++)  {  lenghtPerText[i] = splitStringImageSoal[i].Length; //di isi dari lenght text  }  indexTextTerpanjang = Mathf.Max(lenghtPerText);  //respon box  for(int i=0; i<splitStringImageSoal.Length; i++)  {  GameObject cloneBoxKata = Instantiate(prefabBoxKata);//respawn  cloneBoxKata.transform.SetParent(parentKata);//set parent  }    }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |

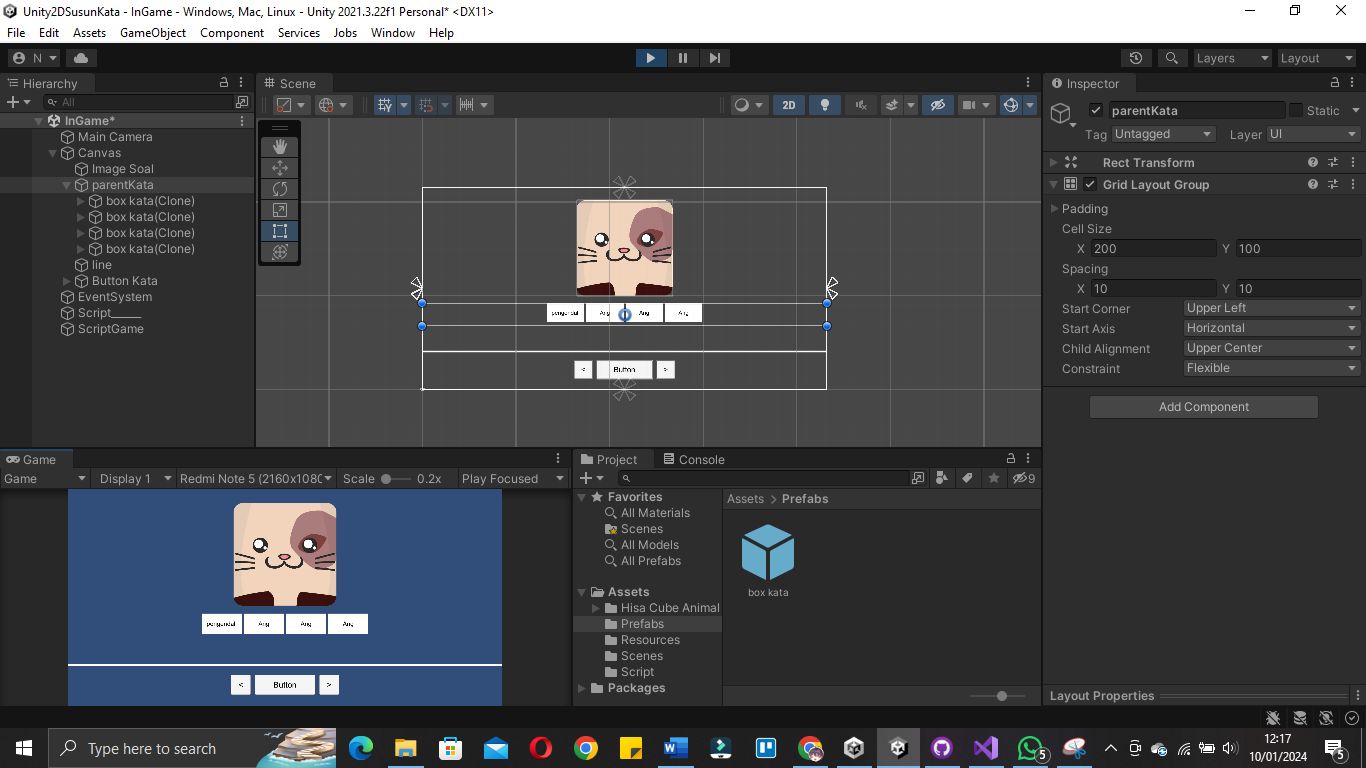


|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  public int[] lenghtPerText;  public int indexTextTerpanjang;  [Header("box kata")]  public GameObject prefabBoxKata;  public Transform parentKata;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  //Implementasi keterangan gambar  splitStringImageSoal = stringImageSoals[indexRandomSprites[gameRound]].Split(' ');//dipotong dengan acuan spasi  lenghtPerText = new int[splitStringImageSoal.Length];  for (int i = 0; i <lenghtPerText.Length; i++)  {  lenghtPerText[i] = splitStringImageSoal[i].Length; //di isi dari lenght text  }  for (int i = 0; i < lenghtPerText.Length; i++)  {  if (lenghtPerText[i] == Mathf.Max(lenghtPerText))  {  indexTextTerpanjang = i; //take index terpanjang  }  }  //respon box  for(int i=0; i<splitStringImageSoal.Length; i++)  {  GameObject cloneBoxKata = Instantiate(prefabBoxKata);//respawn  cloneBoxKata.transform.SetParent(parentKata);//set parent  }    }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |

|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  public int[] lenghtPerText;  public int indexTextTerpanjang;  [Header("box kata")]  public GameObject prefabBoxKata;  public Transform parentKata;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  //Implementasi keterangan gambar  splitStringImageSoal = stringImageSoals[indexRandomSprites[gameRound]].Split(' ');//dipotong dengan acuan spasi  RandomValueString(splitStringImageSoal);//random string  lenghtPerText = new int[splitStringImageSoal.Length];  for (int i = 0; i <lenghtPerText.Length; i++)  {  lenghtPerText[i] = splitStringImageSoal[i].Length; //di isi dari lenght text  }  for (int i = 0; i < lenghtPerText.Length; i++)  {  if (lenghtPerText[i] == Mathf.Max(lenghtPerText))  {  indexTextTerpanjang = i; //take index terpanjang  }  }  //respon box  for(int i=0; i<splitStringImageSoal.Length; i++)  {  GameObject cloneBoxKata = Instantiate(prefabBoxKata);//respawn  cloneBoxKata.transform.SetParent(parentKata);//set parent  if(i == 0)  {  //cloneBoxKata.transform.GetChild(0).GetComponent<Text>().text =  }  }    }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  void RandomValueString(string[] indexRandoms)  {  for (int i = 0; i < indexRandoms.Length; i++)  {  string a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |



|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  public int[] lenghtPerText;  public int indexTextTerpanjang;  [Header("box kata")]  public GameObject prefabBoxKata;  public Transform parentKata;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  //Implementasi keterangan gambar  splitStringImageSoal = stringImageSoals[indexRandomSprites[gameRound]].Split(' ');//dipotong dengan acuan spasi  RandomValueString(splitStringImageSoal);//random string  lenghtPerText = new int[splitStringImageSoal.Length];  for (int i = 0; i <lenghtPerText.Length; i++)  {  lenghtPerText[i] = splitStringImageSoal[i].Length; //di isi dari lenght text  }  for (int i = 0; i < lenghtPerText.Length; i++)  {  if (lenghtPerText[i] == Mathf.Max(lenghtPerText))  {  indexTextTerpanjang = i; //take index terpanjang  }  }  //respon box  for(int i=0; i<splitStringImageSoal.Length; i++)  {  GameObject cloneBoxKata = Instantiate(prefabBoxKata);//respawn  cloneBoxKata.transform.SetParent(parentKata);//set parent  if(i == 0)  {  Text textTerpanjang = cloneBoxKata.transform.GetChild(0).GetComponent<Text>();  textTerpanjang.text = splitStringImageSoal[indexTextTerpanjang];//get text  parentKata.GetComponent<GridLayoutGroup>().cellSize = new Vector2(textTerpanjang.preferredWidth + 50f,parentKata.GetComponent<GridLayoutGroup>().cellSize.y);  }  }    }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  void RandomValueString(string[] indexRandoms)  {  for (int i = 0; i < indexRandoms.Length; i++)  {  string a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |



|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  using UnityEngine.UI;  public class ControlGame : MonoBehaviour  {  public int gameRound;  [Header("Image Question")]    public Image ImageSoal;    public Sprite[] spriteSoal;  public int[] indexRandomSprites;  [Tooltip("Jika ingin random tekan ini")]  public bool isRandomSprite;  [Header("String keterangan gambar")]    public string[] stringImageSoals;  public string[] splitStringImageSoal;  public int[] lenghtPerText;  public int indexTextTerpanjang;  [Header("box kata")]  public GameObject prefabBoxKata;  public Transform parentKata;  public float extraSpaceBoxKata;  void Start()//3  {  RandomImageSoal();  }  void RandomImageSoal()//2  {  indexRandomSprites = new int[spriteSoal.Length];//membuat slot secara otomatis sesuai prite yang digunakan  for(int i=0; i<indexRandomSprites.Length; i++)  {  indexRandomSprites[i] = i;//fill element array  }  if(isRandomSprite == true)  {  RandomValue(indexRandomSprites); //acak index  }  ImageSoal.sprite = spriteSoal[indexRandomSprites[gameRound]];//implementasi sprite stelah di acak  //Implementasi keterangan gambar  splitStringImageSoal = stringImageSoals[indexRandomSprites[gameRound]].Split(' ');//dipotong dengan acuan spasi  RandomValueString(splitStringImageSoal);//random string  lenghtPerText = new int[splitStringImageSoal.Length];  for (int i = 0; i <lenghtPerText.Length; i++)  {  lenghtPerText[i] = splitStringImageSoal[i].Length; //di isi dari lenght text  }  for (int i = 0; i < lenghtPerText.Length; i++)  {  if (lenghtPerText[i] == Mathf.Max(lenghtPerText))  {  indexTextTerpanjang = i; //take index terpanjang  }  }  //respon box  for(int i=0; i<splitStringImageSoal.Length; i++)  {  GameObject cloneBoxKata = Instantiate(prefabBoxKata);//respawn  cloneBoxKata.transform.SetParent(parentKata);//set parent  if(i == 0) //for change size x  {  Text textTerpanjang = cloneBoxKata.transform.GetChild(0).GetComponent<Text>();  textTerpanjang.text = splitStringImageSoal[indexTextTerpanjang];//get text  parentKata.GetComponent<GridLayoutGroup>().cellSize = new Vector2(textTerpanjang.preferredWidth + extraSpaceBoxKata,parentKata.GetComponent<GridLayoutGroup>().cellSize.y);  }  Text textCloneBoxKata = cloneBoxKata.transform.GetChild(0).GetComponent<Text>();//set text  textCloneBoxKata.text = "";  for (int j = 0; j <splitStringImageSoal[i].Length; j++)  {  textCloneBoxKata.text += "\_";//change text dengan \_  }  }    }  void RandomValue(int[] indexRandoms)//1  {  for(int i=0; i<indexRandoms.Length; i++)  {  int a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  void RandomValueString(string[] indexRandoms)  {  for (int i = 0; i < indexRandoms.Length; i++)  {  string a = indexRandoms[i];  int b = Random.Range(0, indexRandoms.Length);  indexRandoms[i] = indexRandoms[b];  indexRandoms[b] = a;  }  }  } |

