



LensFrame by SwiftCode

MANUAL TECNICO v1.0

FECHA DE ENTREGA: 18/11/2023

TABLA DE CONTENIDO

1. INTRODUCCIÓN:	3
Descripción:	3
Propósito:	3
2. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO DEL SOFTWARE:	4
Software Utilizado:	4
Librerías utilizadas en todo el proyecto:	4
3. CREACIÓN DE LA APLICACIÓN:	4
LOGIN	5
Menu_Principal	7
Menu_CAT1	8
Menu_CAT2	9
Menu_Model1	10
Menu_Model2	11
Menu_Model3	12
Menu_Model4	13
Menu_Search	14
Mod_1,2,3,4	15
OPCIONES	16
4. SCRIPTS UTILIZADOS:	17
SCRIPT NÚMERO 1 (LENSMANAGER)	17
SCRIPT NÚMERO 2 (UIMANAGER)	18
SCRIPT NUMERO 3 (NETWORKSMANAGER)	19
SCRIPT NÚMERO 4 (USERSMANAGER)	20
5. SOPORTE TÉCNICO:	22
Información de contacto para el soporte técnico:	22
6. ACTUALIZACIONES DE LA APLICACIÓN:	22
Pasos para actualizar la aplicación:	22
7. TERMINOS Y CONDICIONES:	23
Términos y Condiciones de Uso para Aplicación Móvil: LensFrame	23
8. AGRADECIMIENTOS:	25

1. INTRODUCCIÓN:

Descripción:

Esta aplicación móvil es una herramienta diseñada para mejorar la experiencia del usuario al explorar y seleccionar productos ópticos, como de contacto. Utilizando la tecnología de realidad aumentada, la aplicación permite a los usuarios visualizar cómo lucirían diferentes modelos de gafas en su rostro antes de realizar una posible compra.

Para ello, esta aplicación es útil y eficiente cumpliendo así su objetivo y trayendo al mercado una experiencia diferente a lo que comúnmente la gente está acostumbrada con este tipo de situaciones.

Propósito:

El propósito principal de la aplicación es transformar la forma en que los usuarios eligen y adquieren productos al fusionar la experiencia con la tecnología de realidad aumentada. Algunos de los objetivos clave de la aplicación incluyen:

1. **Mejorar la experiencia del cliente:** La aplicación de AR proporcionará a los clientes una experiencia de compra única y personalizada al permitirles probar virtualmente una amplia variedad de gafas en tiempo real. Esto mejorará la satisfacción del cliente y fomentará la fidelización.
2. **Personalización:** Permitir a los usuarios encontrar el producto que se adapte mejor a su estilo y necesidades individuales al visualizar cómo se ven las opciones en su propio rostro.
3. **Facilitar la Elección de Productos:** Simplificar el proceso de selección de productos al proporcionar a los usuarios una representación visual precisa de cómo se verán con los productos elegidos.
4. **Promover la Confianza del Usuario:** Mejorar la confianza del usuario al brindar una experiencia de virtual que reduce la incertidumbre y las sorpresas desagradables al recibir el producto físico.
5. **Ampliar el alcance:** La aplicación de AR no solo servirá a los clientes que visitan la tienda física, sino que también se puede utilizar de forma remota, lo que ampliará el alcance de la tienda y atraerá a un público más amplio.

En resumen, la aplicación móvil de óptica con realidad aumentada busca redefinir y mejorar la experiencia de compra de productos ópticos al integrar tecnología innovadora para brindar a los usuarios una herramienta más visual, informativa y personalizada.

2. HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO DEL SOFTWARE:

Software Utilizado:

Unity Version 2022.3.11f1: [Descarga el Unity Hub para comenzar tus proyectos creativos | Unity](#)

Visual Studio 2022 (al descargar unity te instala como complemento visual studio)

Blender 2022: <https://www.blender.org/download/>

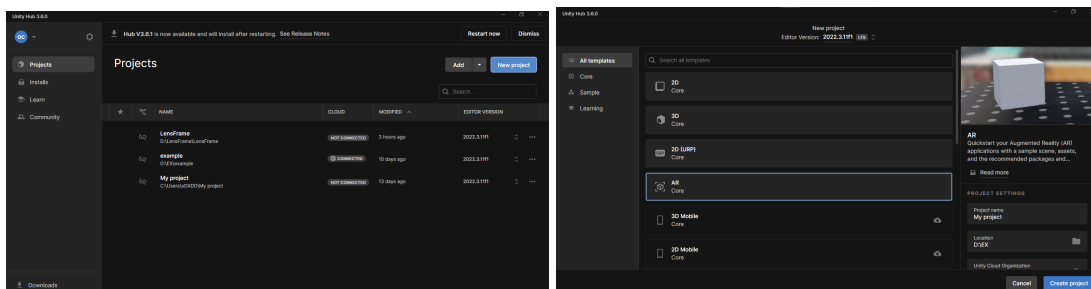
Adobe Photoshop 2021 (Para la creación de diseño de las interfaces, botones, etc)

Librerías utilizadas en todo el proyecto:

- Vuforia Engine AR (Por cuestiones técnicas en obsolescencia nuestras necesidades no eran cumplidas con esta librería, pero una mención sobre su uso como primera opción para el uso de AR): [SDK Download | Engine Developer Portal \(vuforia.com\)](#)
- Magic Leap XR
- Ar Foundation
- Google AR Core XR

3. CREACIÓN DE LA APLICACIÓN:

Con lo primero que se comenzó fue con ejecutar la aplicación de unity la cual por default nos asigna un hub en el cual el usuario crea un nuevo proyecto para ello se divide en categorías en nuestro caso que necesitamos el uso de AR seleccionamos dicha opción para la creación del proyecto:

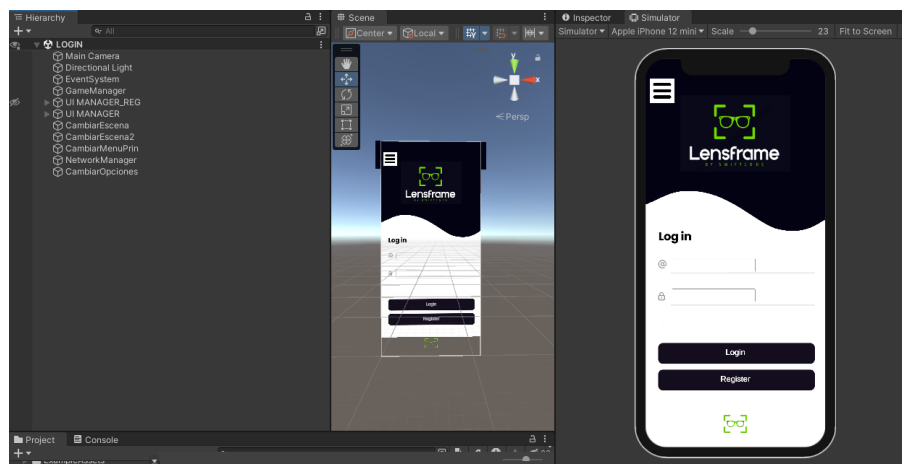


Una vez seleccionado el tipo de proyecto y crearlo unity empezará con la creación de dicho proyecto (nota esto tarda varios minutos, o su demora varía dependiendo el tipo de equipo con el que se está desarrollando la aplicación).



Una vez dentro del proyecto lo primero que se comenzó a trabajar fueron en las escenas de dicho proyecto para ello aquí se muestran las escenas con sus respectivos componentes utilizados:

LOGIN

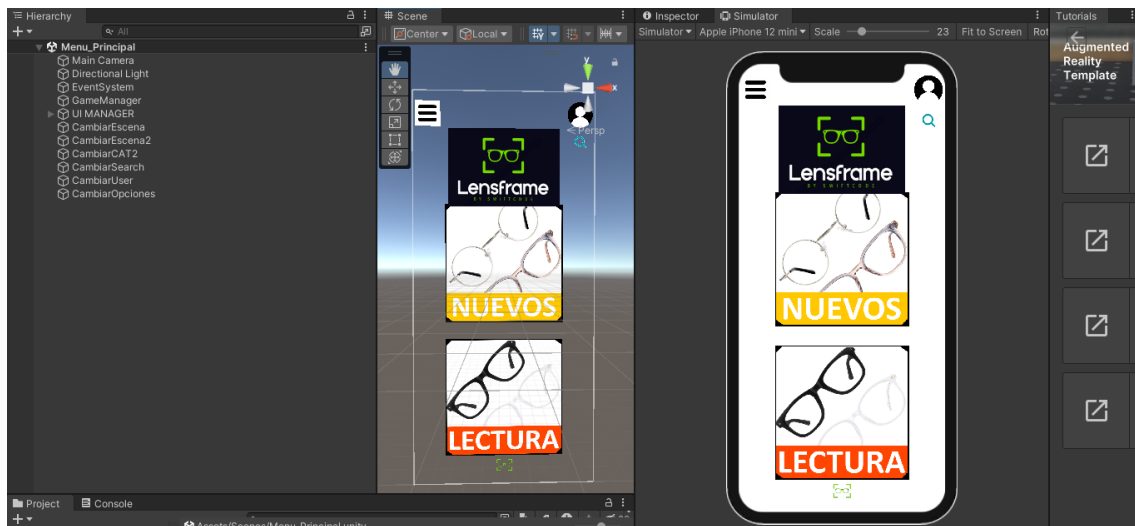


Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.

- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER_REG: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - FONDO: Contiene el color de fondo (030213)
 - CAMPOS DE REGISTRO: inputs text que serán para que el usuario ingrese su nombre y una contraseña para su cuenta.
 - TEXT: esta etiqueta hará uso para avisar al usuario si fue registrado correctamente o si tuvo algún error en los campos de texto.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - FONDO: Contiene el color de fondo (030213)
 - LOGIN: Campo para que el usuario ingrese su nombre y contraseña
 - TEXT: esta etiqueta hará uso para avisar al usuario si fue logueado correctamente o si tuvo algún error en los campos de texto.
 - BOTÓN LOGIN: Funcionamiento para loguear al usuario
 - BOTÓN REGISTER: Para que el usuario pueda registrar una cuenta.
- CambiarEscena: Metodo que cambia a la escena del gui register
- CambiarEscena: Metodo que cambia a la escena del gui login
- CambiarMenuPrin: Cambio de escena al menu principal de la aplicacion
- NetworkManager: Metodo que almacena lo necesario para el registro y login del usuario
- CambiarOpciones: Metodo que cambia a la escena de opciones

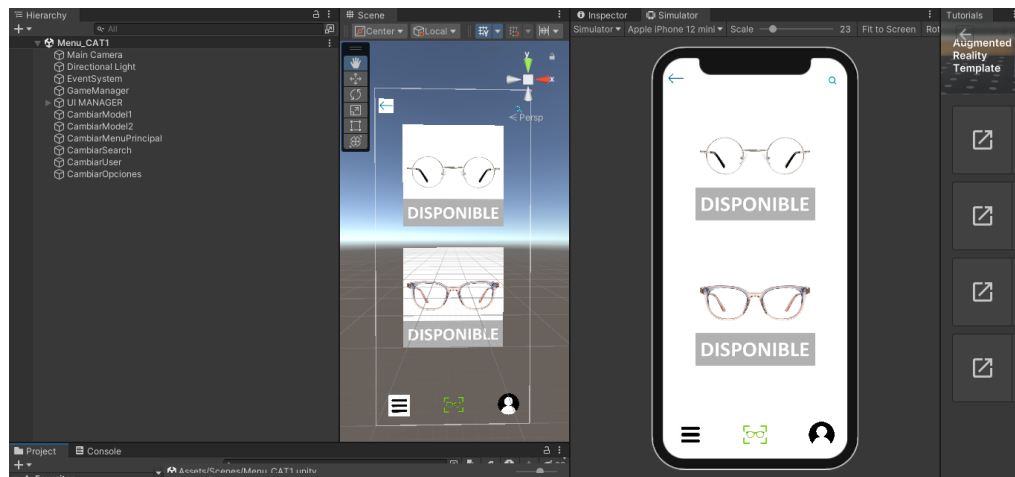
Menu_Principal



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - CAT_1: Contiene el boton para el menu numero 1
 - CAT_2: Contiene el boton para el menu numero 2
 - USER: Contiene el metodo para la interfaz de usuario de login/register
 - SEARCH: Menu de busqueda de todos los productos disponibles
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
- CambiarEscena: Metodo que cambia a la escena de la categoria 1 de lentes
- CambiarCAT2: Metodo que cambia a la escena de la categoria 2 de lentes
- CambiarUser: Cambio de escena al menu de login/register
- CambiarOpciones: Metodo que cambia a la escena de opciones
- CambiarSearch: Metodo que manda al menu de busqueda con todos los modelos

Menu_CAT1



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - Model1_1: Contiene el boton para el modelo numero 1
 - Model1_2: Contiene el boton para el modelo numero 2
 - USER: Contiene el metodo para la interfaz de usuario de login/register
 - SEARCH: Menu de busqueda de todos los productos disponibles
 - BACK: Boton Para regresar a la pantalla anterior
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
- CambiarModel1: Metodo que cambia a la escena del lente 1
- CambiarModel2: Metodo que cambia a la escena del lente 2
- CambiarUser: Cambio de escena al menu de login/register
- CambiarOpciones: Metodo que cambia a la escena de opciones
- CambiarSearch: Metodo que manda al menu de busqueda con todos los modelos

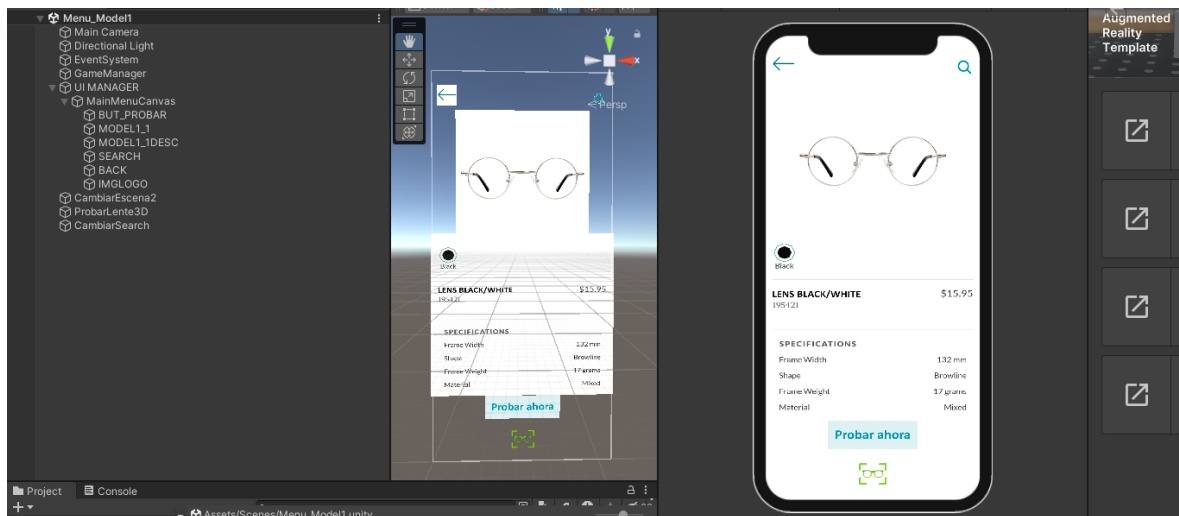
Menu_CAT2



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - Model1_1: Contiene el boton para el modelo numero 3
 - Model2_1: Contiene el boton para el modelo numero 4
 - USER: Contiene el metodo para la interfaz de usuario de login/register
 - SEARCH: Menu de busqueda de todos los productos disponibles
 - BACK: Boton Para regresar a la pantalla anterior
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
- CambiarModel3: Metodo que cambia a la escena del lente 3
- CambiarModel4: Metodo que cambia a la escena del lente 4
- CambiarUser: Cambio de escena al menu de login/register
- CambiarOpciones: Metodo que cambia a la escena de opciones
- CambiarSearch: Metodo que manda al menu de busqueda con todos los modelos

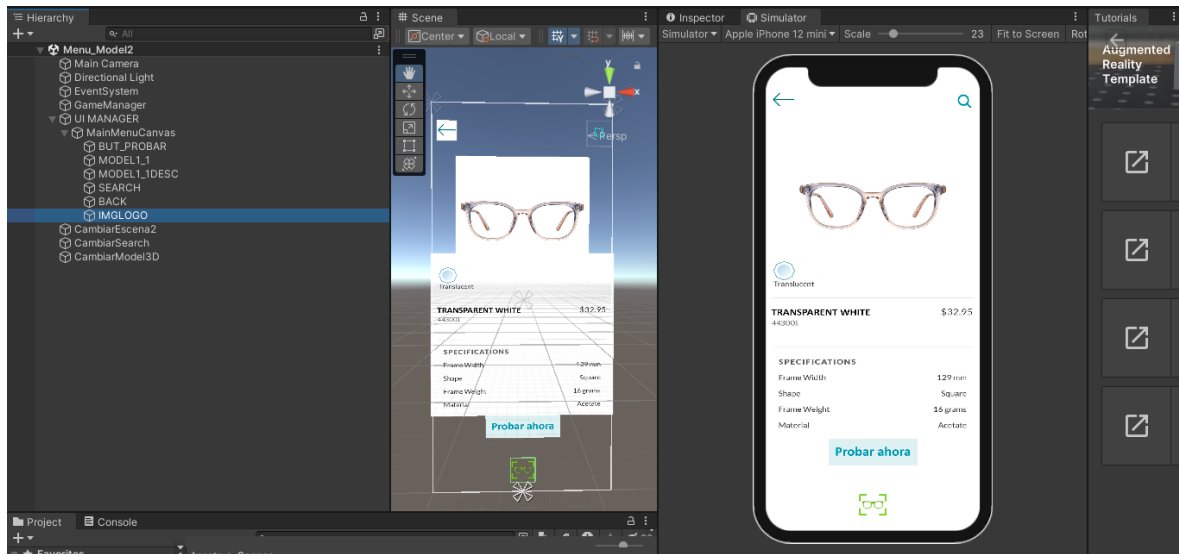
Menu_Model1



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - BUT_PROBAR: Contiene el boton para probar en realidad aumentada el modelo 1
 - Model1_1: Contiene imagen previa del modelo 1
 - Model1_1DESC: Contiene una descripcion del modelo 1
 - SEARCH: Menu de busqueda de todos los productos disponibles
 - BACK: Boton Para regresar a la pantalla anterior
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
- CambiarEscena: Regresa al menu anterior
- ProbarLente3D: Metodo que manda al usuario a probar el modelo 3D seleccionado
- CambiarSearch: Metodo que manda al menu de busqueda con todos los modelos

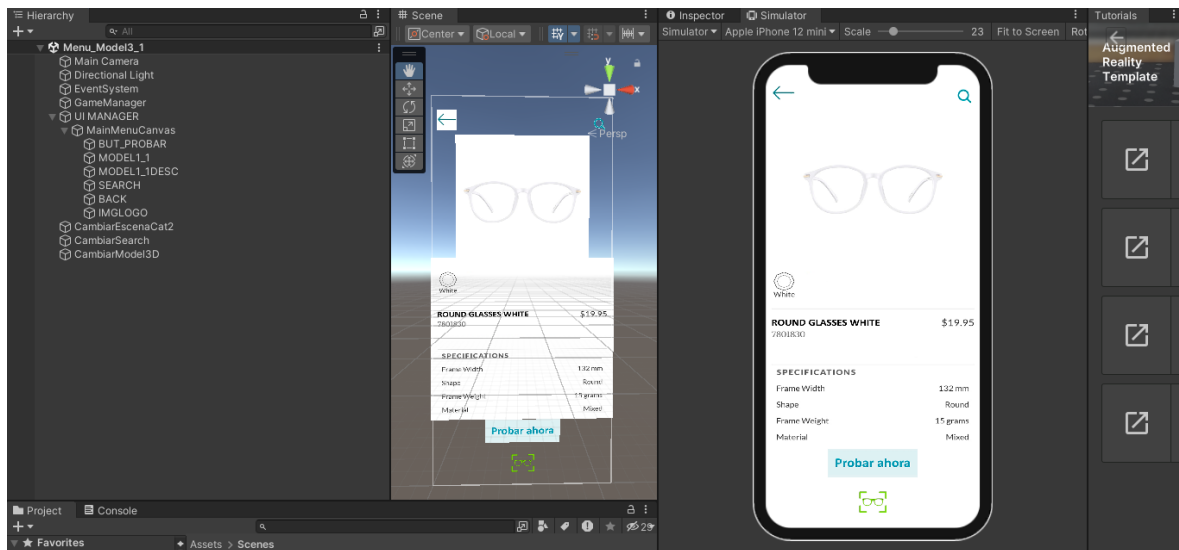
Menu_Model2



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - BUT_PROBAR: Contiene el boton para probar en realidad aumentada el modelo 2
 - Model1_1: Contiene imagen previa del modelo 2
 - Model1_1DESC: Contiene una descripcion del modelo 2
 - SEARCH: Menu de busqueda de todos los productos disponibles
 - BACK: Boton Para regresar a la pantalla anterior
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
- CambiarEscena: Regresa al menu anterior
- ProbarLente3D: Metodo que manda al usuario a probar el modelo 3D seleccionado
- CambiarSearch: Metodo que manda al menu de busqueda con todos los modelos

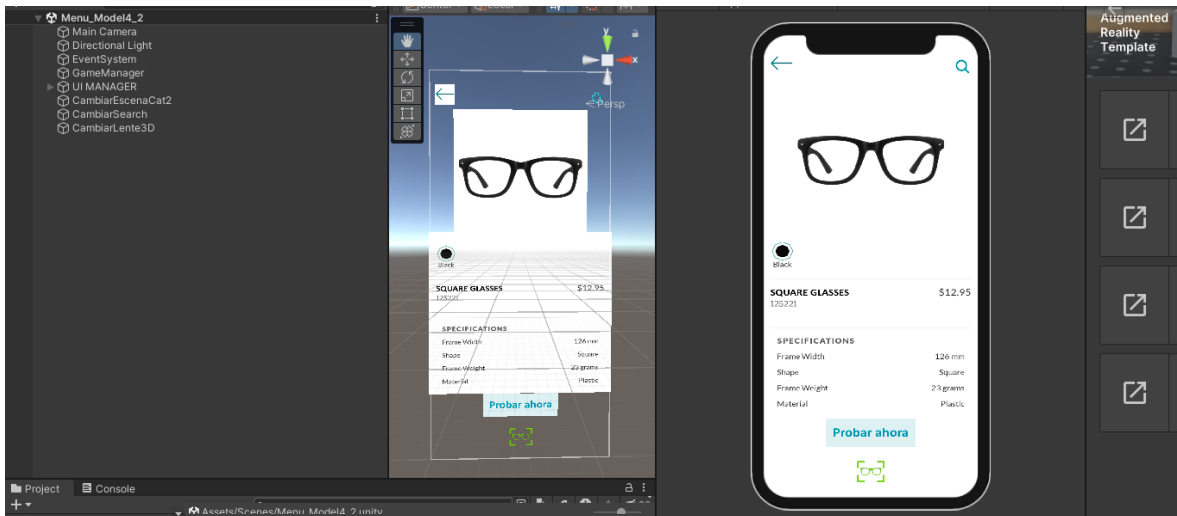
Menu_Model3



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - BUT_PROBAR: Contiene el boton para probar en realidad aumentada el modelo 3
 - Model2_1: Contiene imagen previa del modelo 3
 - Model2_1DESC: Contiene una descripcion del modelo 3
 - SEARCH: Menu de busqueda de todos los productos disponibles
 - BACK: Boton Para regresar a la pantalla anterior
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
- CambiarEscenaCat2: Regresa al menu anterior
- ProbarLente3D: Metodo que manda al usuario a probar el modelo 3D seleccionado
- CambiarSearch: Metodo que manda al menu de busqueda con todos los modelos

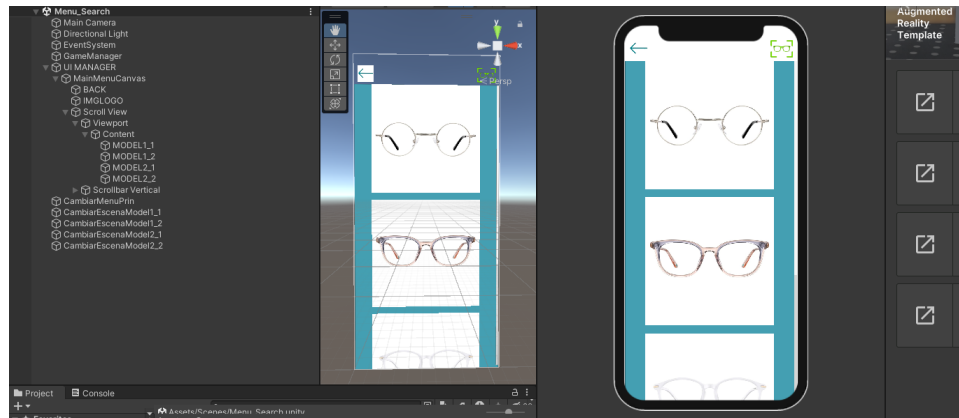
Menu_Model4



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - BUT_PROBAR: Contiene el boton para probar en realidad aumentada el modelo 4
 - Model2_2: Contiene imagen previa del modelo 4
 - Model2_2DESC: Contiene una descripcion del modelo 4
 - SEARCH: Menu de busqueda de todos los productos disponibles
 - BACK: Boton Para regresar a la pantalla anterior
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
- CambiarEscenaCat2: Regresa al menu anterior
- ProbarLente3D: Metodo que manda al usuario a probar el modelo 3D seleccionado
- CambiarSearch: Metodo que manda al menu de busqueda con todos los modelos

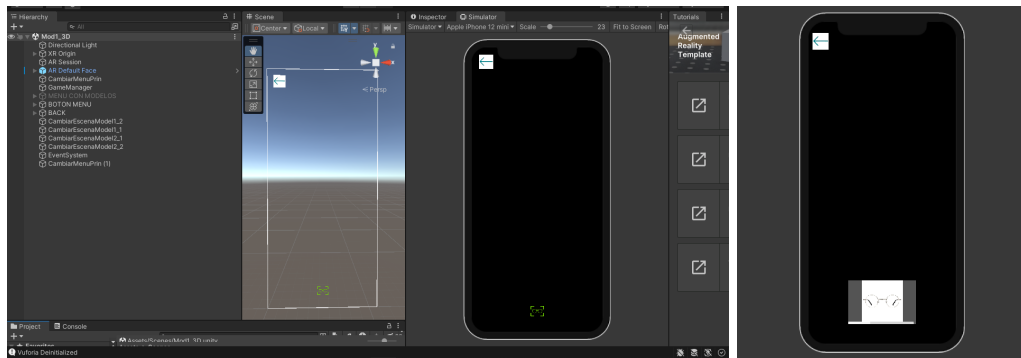
Menu_Search



Contenido de esta escena:

- MainCamera: Esta será la principal razón de que el usuario pueda ver perfectamente la interfaz en su dispositivo.
- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Compone todos los objetos que utiliza la interfaz grafica y aqui se presenta un listado de cada uno de ellos:
 - BACK: Boton Para regresar a la pantalla anterior
 - IMG LOGO/ LOGO MAIN: iconos del menu principal de la aplicacion
 - ScrollView: Este contiene un scroll de manera vertical en la cual contiene todos los modelos disponibles:
 - MODEL1_1: modelo 1
 - MODEL1_2: modelo 2
 - MODEL2_1: modelo 3
 - MODEL2_2: modelo 4
- CambiarMenuPrin: Regresa al menú principal
- CambiarEscenaModel1_1: Se redirige al usuario al modelo número 1 con sus respectivas características
- CambiarEscenaModel1_2: Se redirige al usuario al modelo número 2 con sus respectivas características
- CambiarEscenaModel2_1: Se redirige al usuario al modelo numero 3 con sus respectivas características
- CambiarEscenaModel2_2: Se redirige al usuario al modelo número 4 con sus respectivas características

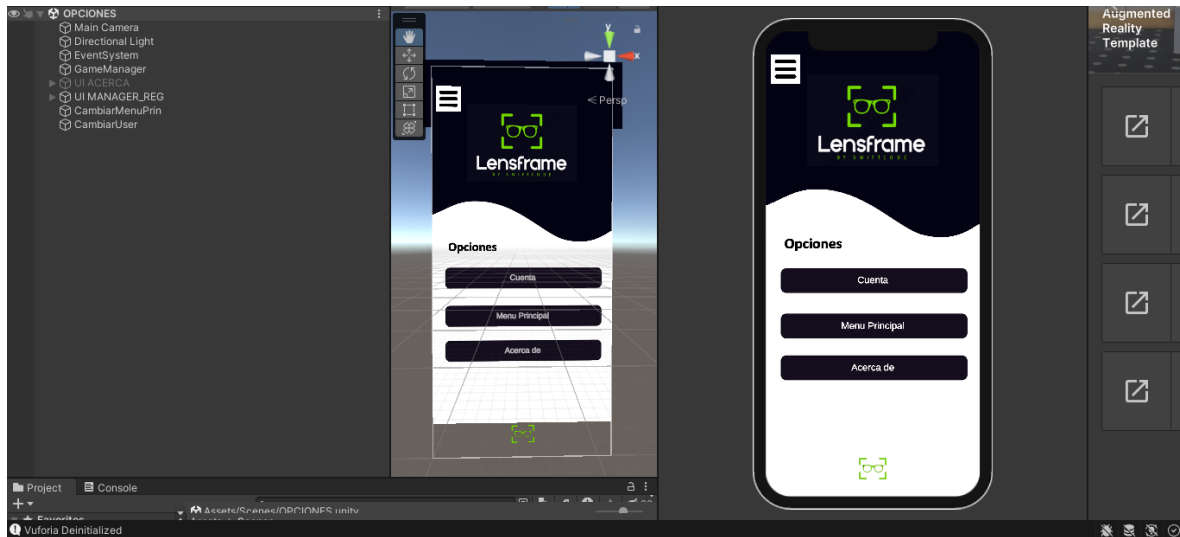
Mod_1,2,3,4



Contenido de esta escena:

- XR Origin: Contiene el punto de inicio de la AR ya que incluye el main camera.
- AR Default Face: Contiene el modelo de lente con su respectivo punto de inicio y detección de cámara
- EventSystem: Este se encargará de que cada evento o acción que haga el usuario lo ejecute sin ningún problema.
- MENU CON MODELOS: Este contiene un scroll de manera horizontal en la cual contiene todos los modelos disponibles:
 - MODEL1_1: modelo 1
 - MODEL1_2: modelo 2
 - MODEL2_1: modelo 3
 - MODEL2_2: modelo 4
- BOTON MENU: Boton que al pulsar el usuario le aparecerá el menu con el contenido de los modelos disponibles.
- BACK: Boton Para regresar a la pantalla anterior
- CambiarEscenaModel1_1: Se redirige al usuario al modelo número 1 con sus respectivas características
- CambiarEscenaModel1_2: Se redirige al usuario al modelo número 2 con sus respectivas características
- CambiarEscenaModel2_1: Se redirige al usuario al modelo número 3 con sus respectivas características
- CambiarEscenaModel2_2: Se redirige al usuario al modelo número 4 con sus respectivas características
- CambiarMenuPrin: Regresa al menú principal

OPCIONES



Contenido de esta escena:

- EventSystem: Este se encargara de que cada evento o acción que haga el usuario lo ejecute sin ningun problema.
- UI MANAGER: Este contiene metodos que te redirigen a la escena seleccionada en los cuales consta de 3:
 - BOTON CUENTA: Este te redirige al apartado de cuenta de usuario
 - BOTON ACERCA_DE: Te redirige a un apartado que incluye informacion del equipo de desarrollo de la aplicacion.
 - MENU PRIN: Te redirige al menu principal de la aplicacion.
- UI ACERCA: Contiene el GUI informativo con la informacion del equipo de desarrollo de la aplicacion
- CambiarMenuPrin: metodo que se utiliza para llamar a la escena del menú principal
- CambiarUser: metodo que se utiliza para llamar a la escena del menu de usuario

4. SCRIPTS UTILIZADOS:

SCRIPT NÚMERO 1 (LENSMANAGER)

LIBRERÍAS UTILIZADAS

```
using UnityEngine;  
using UnityEngine.UI;
```

```
public class ManagerLens : MonoBehaviour //SE DECLARA LA CLASE
```

```
{  
    [SerializeField] private GameObject m_lente1 = null; //SE ASIGNA EL OBJETO (EL  
    MODELO DEL LENTE)
```

```
    [SerializeField] private GameObject m_lente1_menu = null; //SE ASIGNA EL OBJETO  
    (EL MENU DE LA INTERFAZ GRAFICA)
```

```
    [SerializeField] private GameObject m_lente1_but = null; //SE ASIGNA EL OBJETO (EL  
    BOTON QUE HARA LA FUNCION DE MOSTRAR EL MENU)
```

```
public void showlente1() //METODO PARA HACER EL CAMBIO DEL MODELO  
SELECCIONADO
```

```
{  
    m_lente1.SetActive(true); //SE ACTIVA EL MODELO  
    m_lente1_menu.SetActive(false); //SE DESACTIVA EL MENU CUANDO SE CAMBIE  
    DE MODELO  
    m_lente1_but.SetActive(true); //VUELVE A ACTIVARSE EL BOTON PARA ABRIR EL  
    MENU  
}
```

```
public void showlente1menu() //METODO PARA EL MENÚ CON LOS MODELOS
```

```
{  
    m_lente1.SetActive(true); //EL MODELO QUEDA ACTIVADO  
    m_lente1_menu.SetActive(true); //SE MUESTRA EL MENU CON LOS MODELOS  
    DISPONIBLES  
    m_lente1_but.SetActive(false); //SE OCULTA EL BOTON QUE ABRE EL MENU DE  
    LOS MODELOS  
}  
}
```

SCRIPT NÚMERO 2 (UIMANAGER)

LIBRERÍAS UTILIZADAS

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System;
using UnityEngine.SceneManagement; //USO PARA EL CAMBIO DE ESCENAS Y
ADMINISTRARLAS
```

```
public class UIManager : MonoBehaviour //SE DECLARA LA CLASE
```

```
{
```

```
    public int cambioEscena; //SE DECLARA EL CAMBIO DE ESCENA EN TIPO INT YA QUE
    ES EN BASE AL NUMERO DE ESCENAS
```

```
void Start() //SE ACTIVA EL METODO
```

```
{
```

```
    // GameManager.instance.OnMainMenu += ActivateMainMenu;
    // GameManager.instance.OnItemsMenu += ActivateItemsMenu;
    // GameManager.instance.OnAirPosition += ActivateArPosition;
```

```
}
```

```
public void CambiarEscena(int indice) //METODO PARA EL CAMBIO DE ESCENA (TIPO
ENTERO CON INDICES ASIGNADOS POR EL MISMO UNITY)
```

```
{
```

```
    SceneManager.LoadScene(cambioEscena); //EL SCENEMANAGER CARGA LA
    ESCENA QUE SE LE ESPECIFIQUE
```

```
}
```

```
}
```

SCRIPT NUMERO 3 (NETWORKSMANAGER)

LIBRERIAS UTILIZADAS

```
using UnityEngine;
using System;
using System.Collections;
```

```
public class NetworkManager : MonoBehaviour //SE DECLARA LA CLASE
{
    public void createUser(String email, String pass, Action<Response> response) //SE DECLARA EL METODO PARA CREAR LOS USUARIOS
    {
        StartCoroutine(CO_CreateUser(email, pass, response)); //SE DECLARAN LOS CAMPOS QUE UTILIZAREMOS PARA EL REGISTRO
    }

    private IEnumerator CO_CreateUser(String email, String pass, Action<Response> response) //SE DECLARAN LOS TIPOS PARA LOS CAMPOS REQUERIDOS EN EL REGISTRO
    {
        WWWForm form = new WWWForm(); //METODO PARA UTILIZARSE EN FORMULARIO WEB
        form.AddField("email", email); //SE DECLARA EL CAMPO PARA EL CORREO O USUARIO
        form.AddField("pass", pass); //SE DECLARA EL CAMPO PARA LA CONTRASEÑA (SE DELCARA POR AUTOMATICO QUE ES FORMATO DE PASSWORD)

        WWW w = new WWW("http://localhost/lensuser/createUser.php", form); //ENLACE PARA LA BASE DE DATOS DONDE SE ALMACENARAN LOS USUARIOS REGISTRADOS

        yield return w;

        response(JsonUtility.FromJson<Response>(w.text)); //METODO RETURN QUE VERIFICA QUE EL USUARIO FUE REGISTRADO CORRECTAMENTE
    }
}

[Serializable]
public class Response //CLASE QUE SE ENCARGARA PARA EL EVENTO DEL FORMULARIO
{
    public bool done = false;
    public String messege = ""; //MENSAJE DE AVISO AL USUARIO
}
```

SCRIPT NÚMERO 4 (USERSMANAGER)

LIBRERIAS UTILIZADAS

```
using UnityEngine;  
using UnityEngine.UI;
```

```
public class UsersManager : MonoBehaviour //DECLARACION DE LA CLASE  
{  
    [SerializeField] private GameObject m_registerUI = null; //OBJETO QUE SERA LA  
INTERFAZ DE REGISTRO  
    [SerializeField] private GameObject m_loginUI = null; //OBJETO QUE SERA LA  
INTERFAZ DE LOGIN  
    [SerializeField] private InputField m_emailInput = null; //CAMPO DE ENTRADA DEL  
USUARIO  
    [SerializeField] private InputField m_passInput = null; //CAMPO DE ENTRADA DE LA  
PASSWORD DEL USUARIO  
    [SerializeField] private Text m_Text = null; //TEXTO DE AVISO AL USUARIO  
    [SerializeField] private Text m_Textt = null; //TEXTO DE AVISO AL USUARIO DE  
ALGUN ERROR  
  
    private NetworkManager m_NetworkManager = null; //SE DEDCLARA EL CAMPO  
PRIVADO DEL NETWORKMANAGER  
  
    private void Awake() //METODO DE AVISO  
    {  
        m_NetworkManager = GameObject.FindObjectOfType<NetworkManager>(); //SE  
DECLARA EL OBJETO A UTILIZAR PARA LLAMAR LOS METODOS  
CORRESPONDIENTES DEL SCRIPT  
    }  
  
    public void SubmitRegister() //METODO QUE CREA EL REGISTRO DEL USUARIO  
    {  
        if (m_emailInput.text == "" || m_passInput.text == "") //CONDICION DE QUE SI EL  
USUARIO O PASSWORD ESTA VACIO MANDE A LLAMAR UN MENSAJE  
        {  
            m_Text.text = "Favor de rellenar los campos"; //MENSAJE EN CASO DE QUE EL  
USUARIO OLVIDE LLENAR UN CAMPO  
            return;  
        }  
  
        m_NetworkManager.createUser(m_emailInput.text, m_passInput.text, delegate  
(Response response) //METODO QUE CREA EL USUARIO  
        {
```

```

        m_Text.text = "registrado"; //UNA VEZ QUE EL USUARIO INGRESE SU
USUARIO Y CONTRASEÑA DESEADA SALTARA UN MENSAJE DE REGISTRO
        return;
        m_Text.text = response.messege;
    });
}

public void SubmitLogin() //METODO PARA LOGEAR AL USUARIO
{
    if (m_emailInput.text == "" || m_passInput.text == "") //SI EL USUARIO OLVIDA
LLENAR ALGUN CAMPO SALTARA UN AVISO
    {
        m_Textt.text = "Favor de rellenar los campos"; //MENSAJE DE AVISO PARA QUE
EL USUARIO CORROBORE SUS DATOS CORRECTAMENTE
        return;
    }
}

public void showlogin() //METODO QUE CAMBIARA LA INTERFAZ A LA DE LOGIN Y
DESACTIVA LA DE REGISTRO
{
    m_registerUI.SetActive(false); //SE DESACTIVA LA UI DE REGISTRO
    m_loginUI.SetActive(true); //SE ACTIVA LA UI DE LOGIN
}

public void showRegister() //METODO QUE CAMBIARA LA INTERFAZ DE LOGIN A
LA DE REGISTRO
{
    m_registerUI.SetActive(true); //SE ACTIVA LA UI DE REGISTRO
    m_loginUI.SetActive(false); //SE DESACTIVA LA UI DE LOGIN
}
}

```

5. SOPORTE TÉCNICO:

Información de contacto para el soporte técnico:

Si el usuario contase con algún problema en la aplicación o inconveniente puede contar con los siguientes enlaces para cualquier respuesta o duda que sea necesaria:

Contacto numero 1 (Oswaldo Alfonso Hernandez Campos):

- Correo electrónico: alu.20130845@correo.itlalaguna.edu.mx
- Numero de telefono: +52 8713691299

Contacto numero 2 (Marcos Eduardo Solis):

- Correo electrónico: alu.20130844@correo.itlalaguna.edu.mx

Contacto numero 3 (Alejandro Garcia Alferez):

- Correo electrónico: alu.20130832@correo.itlalaguna.edu.mx

Contacto numero 4 (Ivan Osmar Vazquez Flores):

- Correo electrónico: alu.20130827@correo.itlalaguna.edu.mx

Contacto número 5 (Carlos Babun):

- Correo electrónico: alu.20130822@correo.itlalaguna.edu.mx

6. ACTUALIZACIONES DE LA APLICACIÓN:

Pasos para actualizar la aplicación:

El usuario si desea las últimas versiones de la aplicación lo único que tiene que hacer es dirigirse al sitio web <https://lensframe.000webhostapp.com/> en el cual constantemente se estará actualizando los links de dichas actualizaciones recientes, esto con la finalidad de que se arreglen problemas, añadan nuevas cosas, e incluso añadir funcionalidades nuevas para un futuro con la aplicación.

Una vez que el usuario descargue la actualización más reciente, solo deberá dirigirse a la ruta donde asigno la descarga, abrir el instalador y actualizar la aplicación, para ello previamente el usuario asignar permisos para acceder a archivos y multimedia de sitios externos para que no cuente con ningún error de instalación (en caso de necesitarlo, checar el punto número 7 de dicho manual).

7. TERMINOS Y CONDICIONES:

Términos y Condiciones de Uso para Aplicación Móvil: LensFrame

Por favor, lea detenidamente los siguientes términos y condiciones antes de utilizar la aplicación móvil LensFrame. Al descargar, instalar o utilizar esta aplicación, usted acepta cumplir con estos términos y condiciones.

1. Aceptación de Términos:

Al acceder y utilizar la aplicación móvil LensFrame, usted acepta y se compromete a cumplir con estos términos y condiciones. Si no está de acuerdo con alguno de estos términos, por favor, no utilice la aplicación.

2. Uso Autorizado:

La aplicación se proporciona para su uso personal y no comercial. No está permitido modificar, copiar, distribuir, transmitir, mostrar, realizar, reproducir, publicar, licenciar o crear trabajos derivados basados en la aplicación.

3. Privacidad:

El uso de la aplicación está sujeto a nuestra Política de Privacidad.

4. Actualizaciones y Cambios:

Nos reservamos el derecho de realizar cambios en la aplicación, incluyendo actualizaciones, modificaciones o la suspensión del servicio en cualquier momento sin previo aviso.

5. Responsabilidades del Usuario:

Usted acepta no utilizar la aplicación para actividades ilegales, difamatorias, abusivas, ofensivas o que violen los derechos de terceros. Además, se compromete a no intentar eludir la seguridad de la aplicación ni a realizar acciones que puedan dañar, desactivar o sobrecargar la aplicación.

6. Limitación de Responsabilidad:

La aplicación se proporciona "tal cual" y "según disponibilidad". No garantizamos la disponibilidad ininterrumpida o sin errores de la aplicación. En ningún caso seremos responsables de daños directos, indirectos, incidentales, especiales o consecuentes que resulten del uso o la imposibilidad de uso de la aplicación.

7. Cambios en los Términos:

Nos reservamos el derecho de modificar estos términos y condiciones en cualquier momento. Los cambios entrarán en vigencia inmediatamente después de su publicación en la aplicación. Se recomienda revisar periódicamente estos términos para estar al tanto de las actualizaciones.

Al utilizar la aplicación móvil LensFrame, usted acepta estos términos y condiciones. Si tiene alguna pregunta, por favor contáctenos a través de alu.20130845@correo.itlalaguna.edu.mx para más información.

8. AGRADECIMIENTOS:

¡Queridos usuarios de LensFrame!

Queremos expresar nuestro sincero agradecimiento por elegir nuestra aplicación para satisfacer sus necesidades. En LensFrame, nos esforzamos continuamente por ofrecer una experiencia excepcional y útil para cada uno de ustedes. Su elección de confiar en nosotros no pasa desapercibida, y estamos comprometidos a proporcionar servicios de alta calidad que superen sus expectativas.

Valoramos su opinión y estamos interesados en conocer sus experiencias con nuestra aplicación. Nos gustaría invitarlos a compartir sus comentarios, sugerencias o cualquier comentario que tengan sobre cómo podemos hacer que LensFrame sea aún mejor para ustedes.

Estamos emocionados de tenerlos a bordo y queremos recordarles que esta aplicación ha sido creada pensando en ustedes. Explorarán las comodidades con cada función que ofrece LensFrame. Los alentamos a sumergirse, experimentar y disfrutar de todo lo que tenemos preparado para ustedes.

Su satisfacción es nuestra principal prioridad, y estamos aquí para hacer que cada interacción con la aplicación sea memorable. Siéntanse libres de explorar, aprender y compartir sus experiencias con nosotros.

Su voz es fundamental para nuestro crecimiento y mejora continua. Entendemos que cada usuario tiene necesidades únicas, y estamos aquí para adaptarnos y evolucionar en consecuencia. Para proporcionar sus comentarios, puede ponerse en contacto con nuestro equipo de soporte a través de los medios de soporte (proporcionados en el punto número 8).

Gracias nuevamente por elegir LensFrame. Estamos emocionados de tenerlos a bordo y esperamos seguir siendo su elección preferida.

Atentamente, El Equipo de SwiftCode.