



UNIVERSIDAD NACIONAL DE TRUJILLO  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
ESCUELA PROFESIONAL DE INFORMÁTICA



# APLICACIÓN MÓVIL DE REALIDAD AUMENTADA PARA LA VISUALIZACIÓN DE ANIMALES CLASIFICADOS

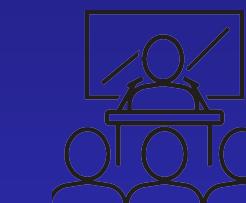


## DOCENTE:

ING. MARTIN SALCEDO QUIÑONES



CURSO:  
TÓPICOS EN TECNOLOGÍAS  
INMERSIVAS



## INTEGRANTES:

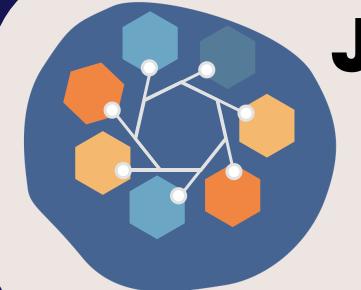
- GUTIERREZ PRETELL, ALDAHIR KENYI NOE
- PADILLA LEYVA LESLIE VALENTINA
- REBAZA AGUILAR, MARCO JAVIER
- RECALDE MONZÓN ANGIE TATIANA

# INDICE



## REALIDAD PROBLEMÁTICA

SOCIAL  
TECNOLÓGICA



## JUSTIFICACIÓN



## OBJETIVOS

GENERAL  
ESPECÍFICOS



## METODOLOGIA



## CRONOGRAMA



## ANÁLISIS DE REQUERIMIENTOS

FUNCIONALES  
NO FUNCIONALES



## HERRAMIENTAS DE DESARROLLO

Unity  
Google ARCore  
AR Foundation  
Blender  
Dispositivo móvil



## DESARROLLO DEL TRABAJO



## CONCLUSIONES Y RECOMENDACIONES

# REALIDAD PROBLEMÁTICA

El cuidado de la vida silvestre y la concientización a los ciudadanos sobre los múltiples desafíos que existen en la actualidad se ven limitados por las pocas oportunidades para que un niño interactúe con animales de manera segura en la ciudad de Trujillo.



Se propone el desarrollo de una app que permita a los niños interactuar con la fauna. Mediante esta experiencia, podrán interactuar de manera segura y fomentando al mismo tiempo una conexión y concientización por el cuidado de los animales.

- Los animales deben mantenerse en áreas restringidas para garantizar la seguridad, limitando la experiencia a una simple observación.
- La interacción directa con animales representa riesgos inherentes



# JUSTIFICACIÓN

## SOCIAL

La aplicación puede dar un frente a la necesidad de educación ambiental y de concienciación sobre la biodiversidad sin que los usuarios, haciendo énfasis en los niños, puedan interactuar con animales en cautiverio, ya que hoy en día tienen acceso limitado a la naturaleza. Por la cual el proyecto brinda una alternativa educativa y accesible para aprender sobre la fauna. Esto fomenta la educación ambiental y también promueve el interés por la conservación de especies, que es un tema importante en el actual contexto del cambio climático y la pérdida de biodiversidad.



## TECNOLÓGICA

La aplicación aprovecha el avance de las tecnologías en realidad virtual como Google ARCore que hace posible proyectar elementos virtuales y de esta manera transformar el aprendizaje en una experiencia inmersiva y atractiva permitiendo fusionar el mundo digital y físico. En este sentido, la aplicación no solo resuelve una necesidad educativa, sino que también impulsa el progreso y la adopción de tecnologías emergentes en el ámbito educativo y social.



WHY?



# OBJETIVOS



Desarrollar una aplicación móvil de realidad aumentada utilizando Google ARCore que permita a los usuarios visualizar y aprender sobre animales clasificados en diferentes categorías (mamíferos, acuáticos y reptiles).

1

Elegir e implementar las tecnologías de realidad virtual

2

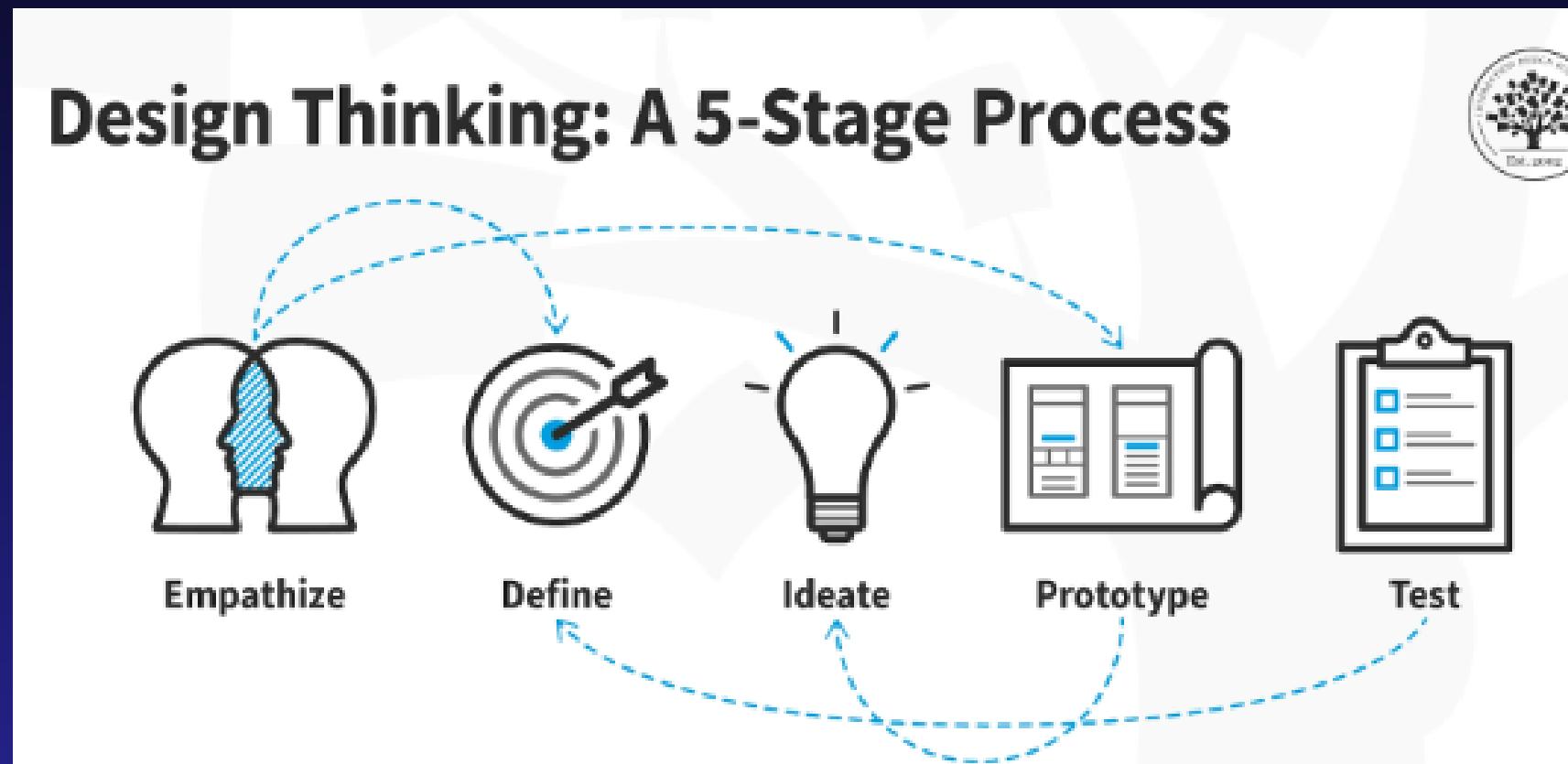
Implementar un sistema de portales en realidad aumentada para clasificar a los animales en diferentes hábitats virtuales

3

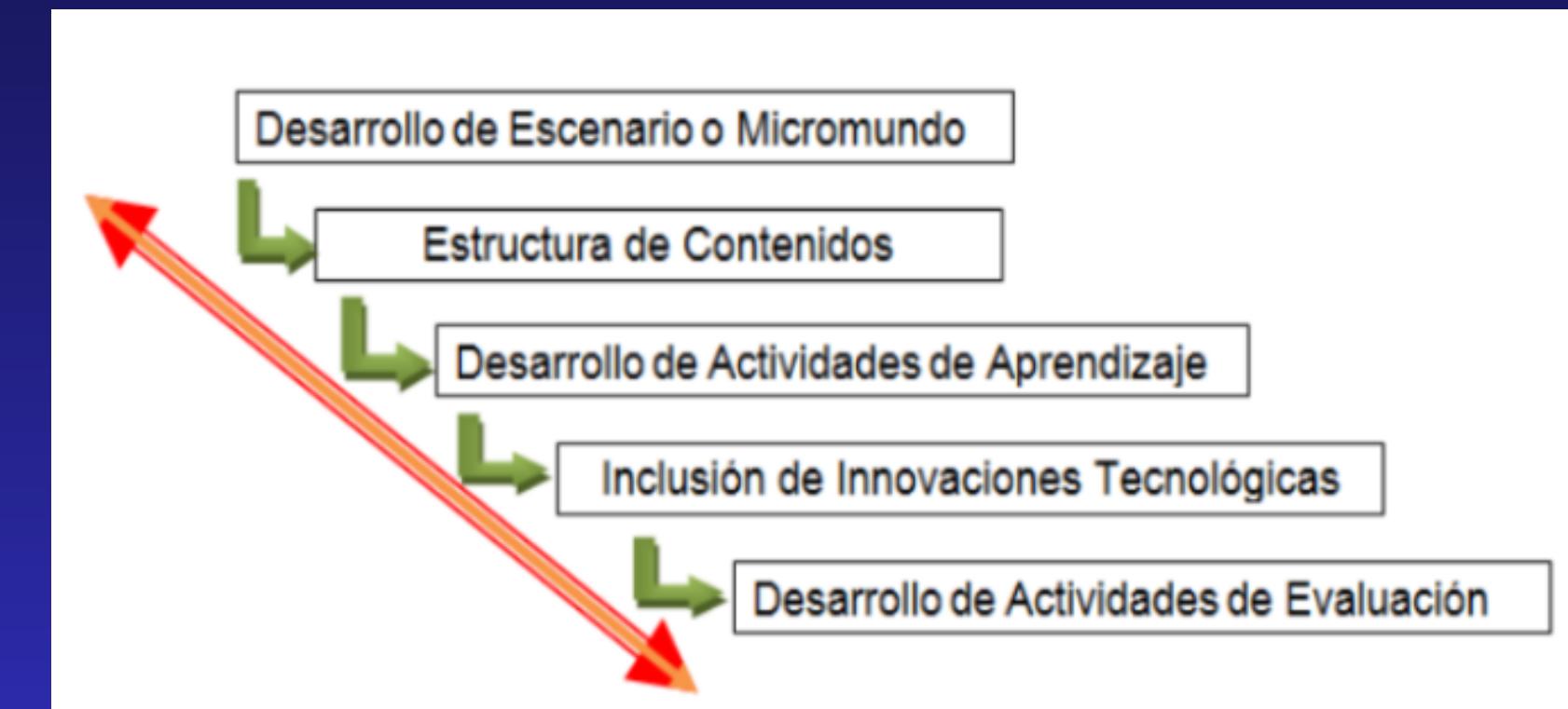
Desarrollar modelos 3D realistas y animados de los animales para ofrecer una experiencia visual atractiva y educativa.

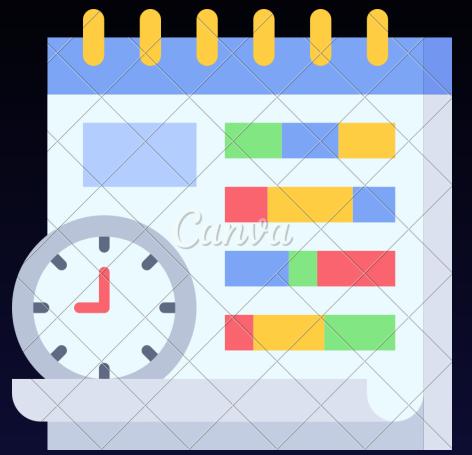
# METODOLOGÍA

Design Thinking, metodología centrada en el usuario

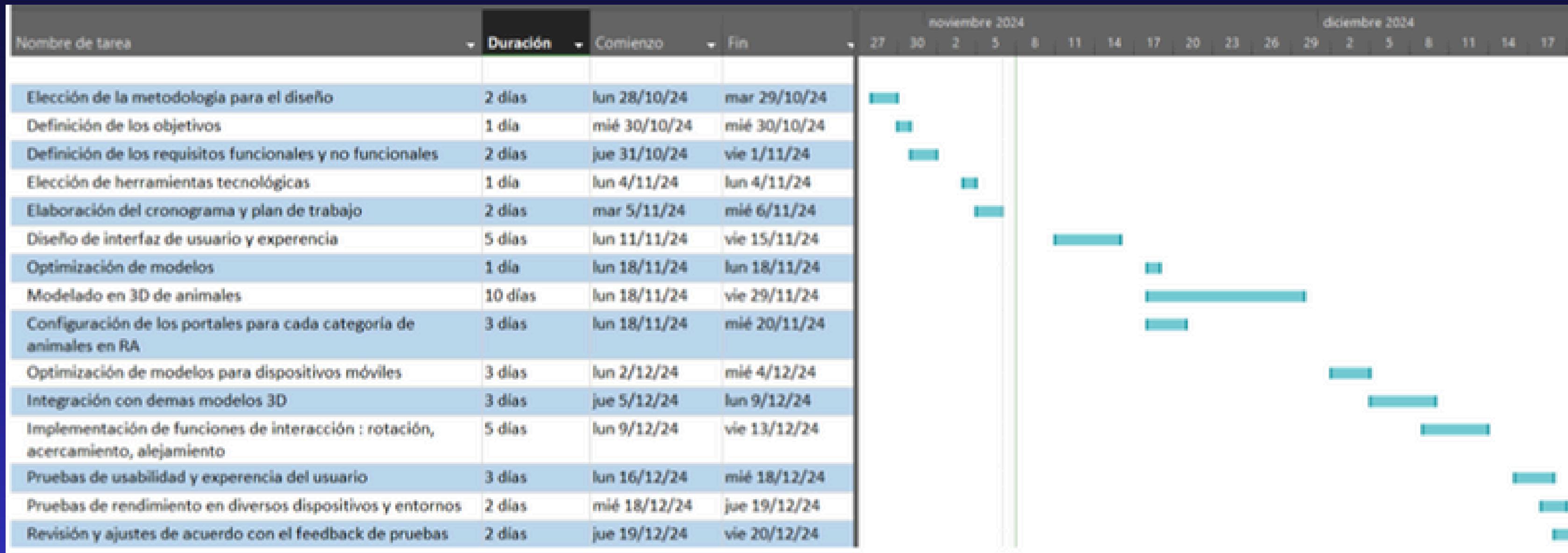


Producción de un OA de la metodología UBoa.





# CRONOGRAMA DE TRABAJO



# ANÁLISIS DE REQUERIMIENTOS

## REQUERIMIENTOS FUNCIONALES

**RF1.** El sistema debe permitir al usuario acceder a la visualización en realidad **aumentada de animales clasificados en tres categorías: mamíferos, acuáticos y reptiles.**

**RF2.** El sistema debe habilitar portales específicos para cada categoría, los cuales permitirán al usuario acceder y visualizar en RA únicamente los animales correspondientes a la categoría seleccionada.

**RF3.** El sistema debe permitir al usuario desplazarse libremente y observar el objeto en todos sus ángulos dentro del escenario.

**RF4.** El sistema debe permitir desplazarse de portal en portal por las tres categorías.

## REQUERIMIENTOS NO FUNCIONALES

**RNF1. Rendimiento:** La aplicación de RA debe operar de forma rápida y eficiente.

**RNF2. Usabilidad:** La interfaz de la aplicación debe ser intuitiva y fácil de usar para niños.

**RNF3. Compatibilidad:** La aplicación debe ser compatible con dispositivos Android que soporten Google ARCore.

**RNF5. Pruebas y Calidad:** Llevar a cabo pruebas completas de usabilidad y calidad



# HERRAMIENTAS DE DESARROLLO

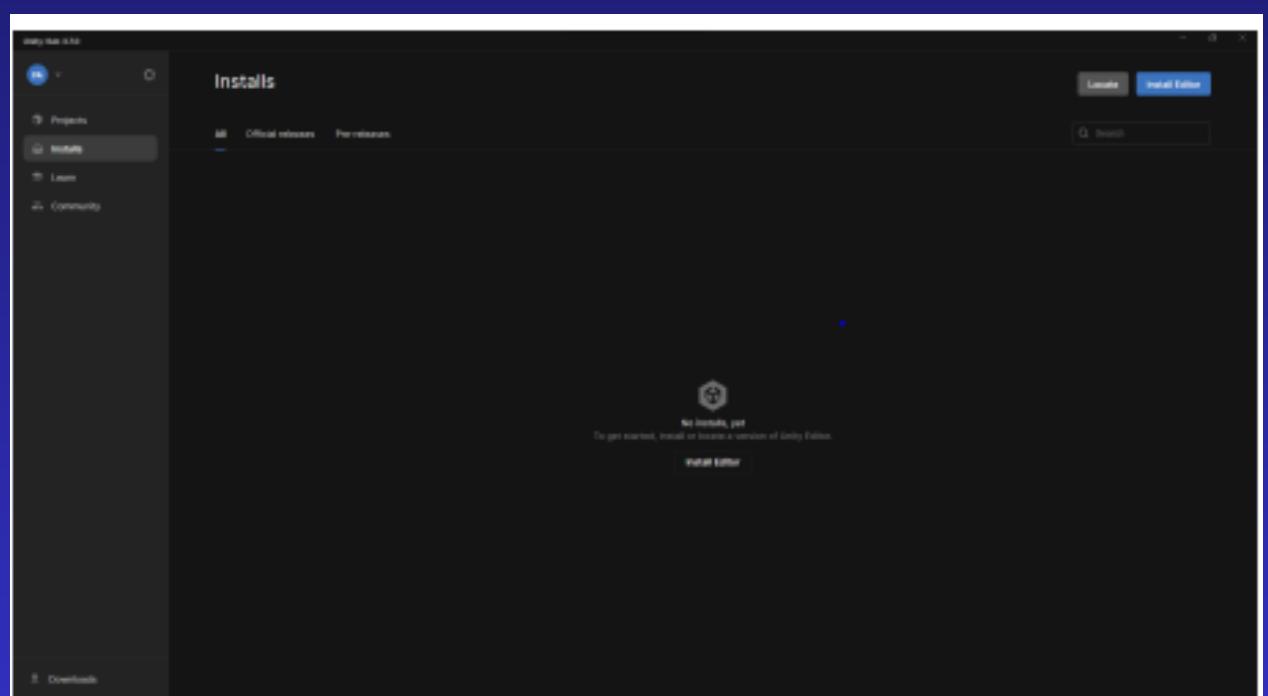
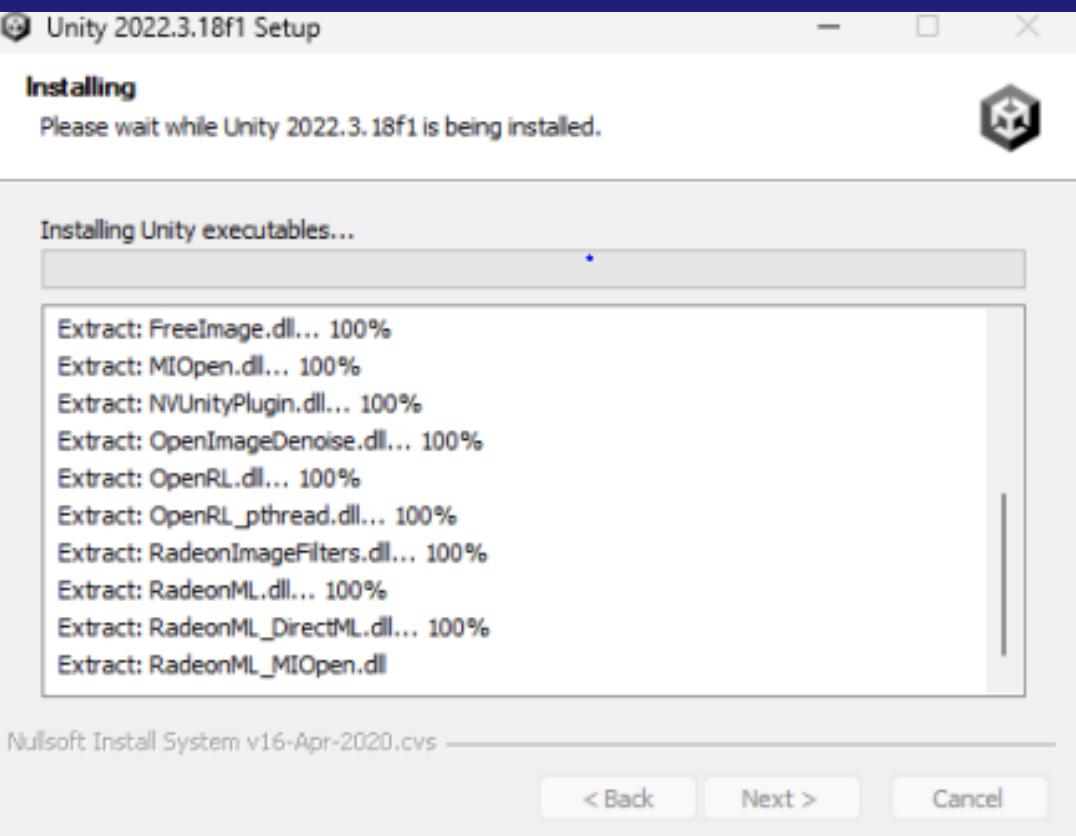
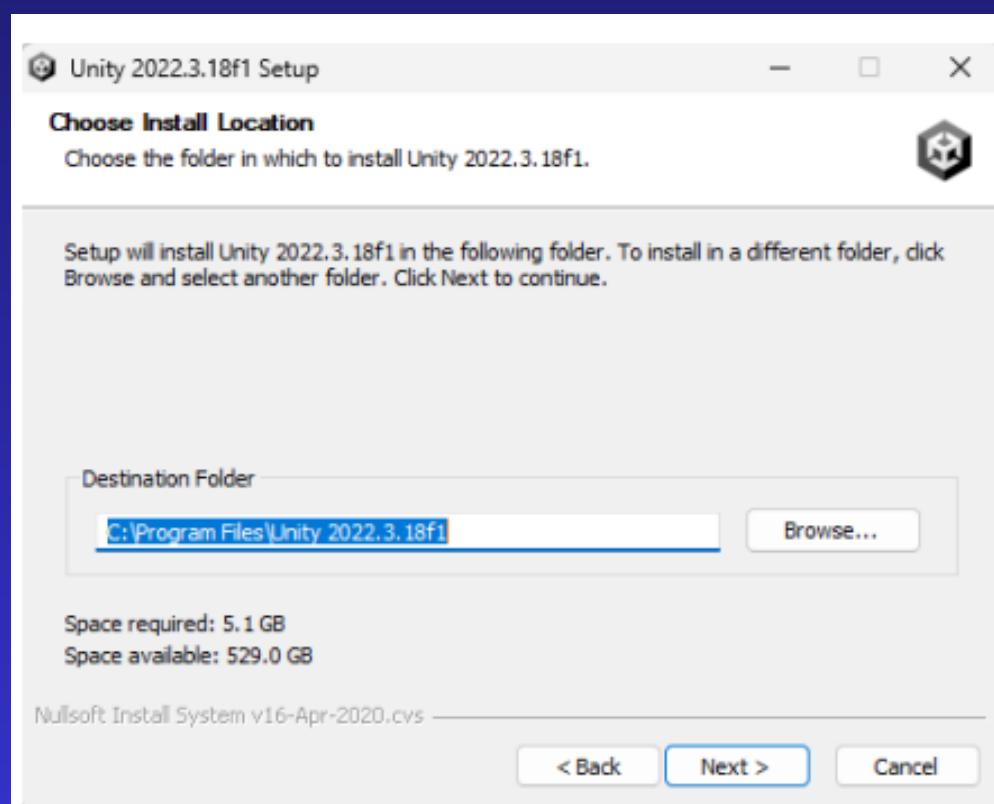
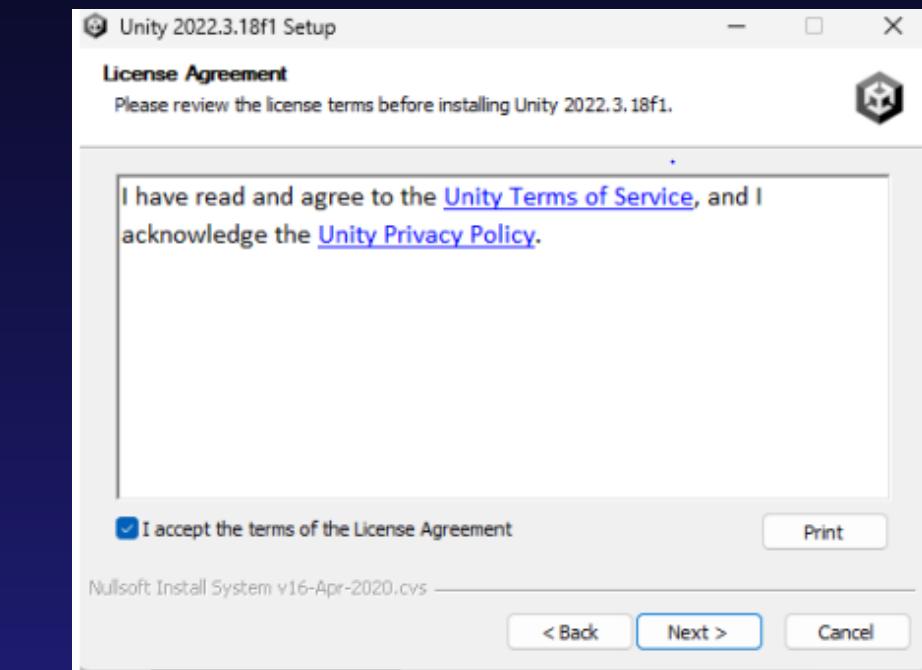
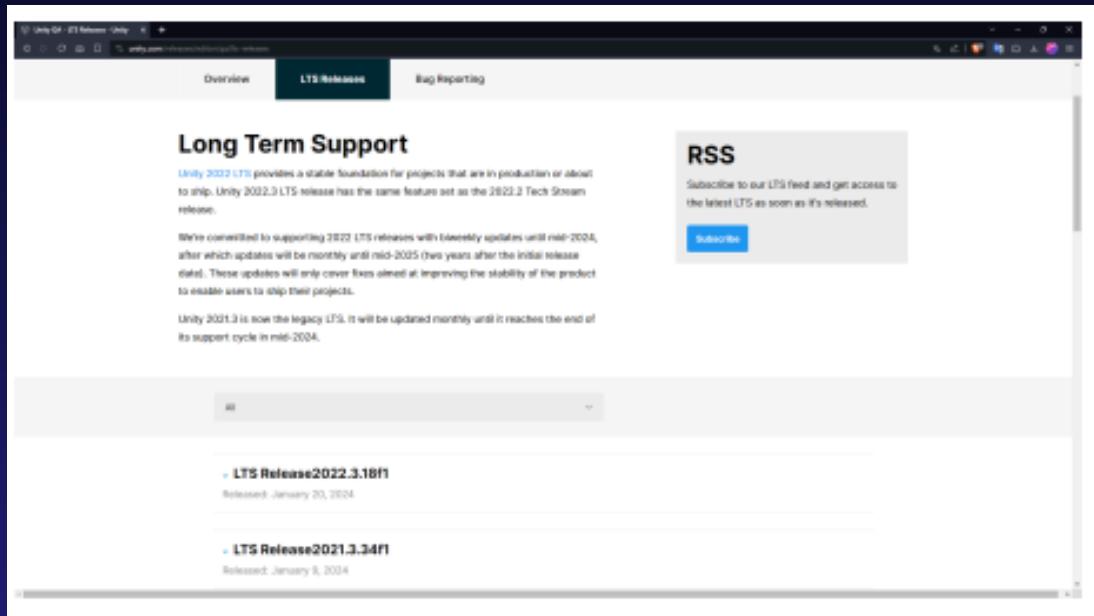
## SOFTWARE





# DESARROLLO DEL ESCENARIO O MICROMUNDO

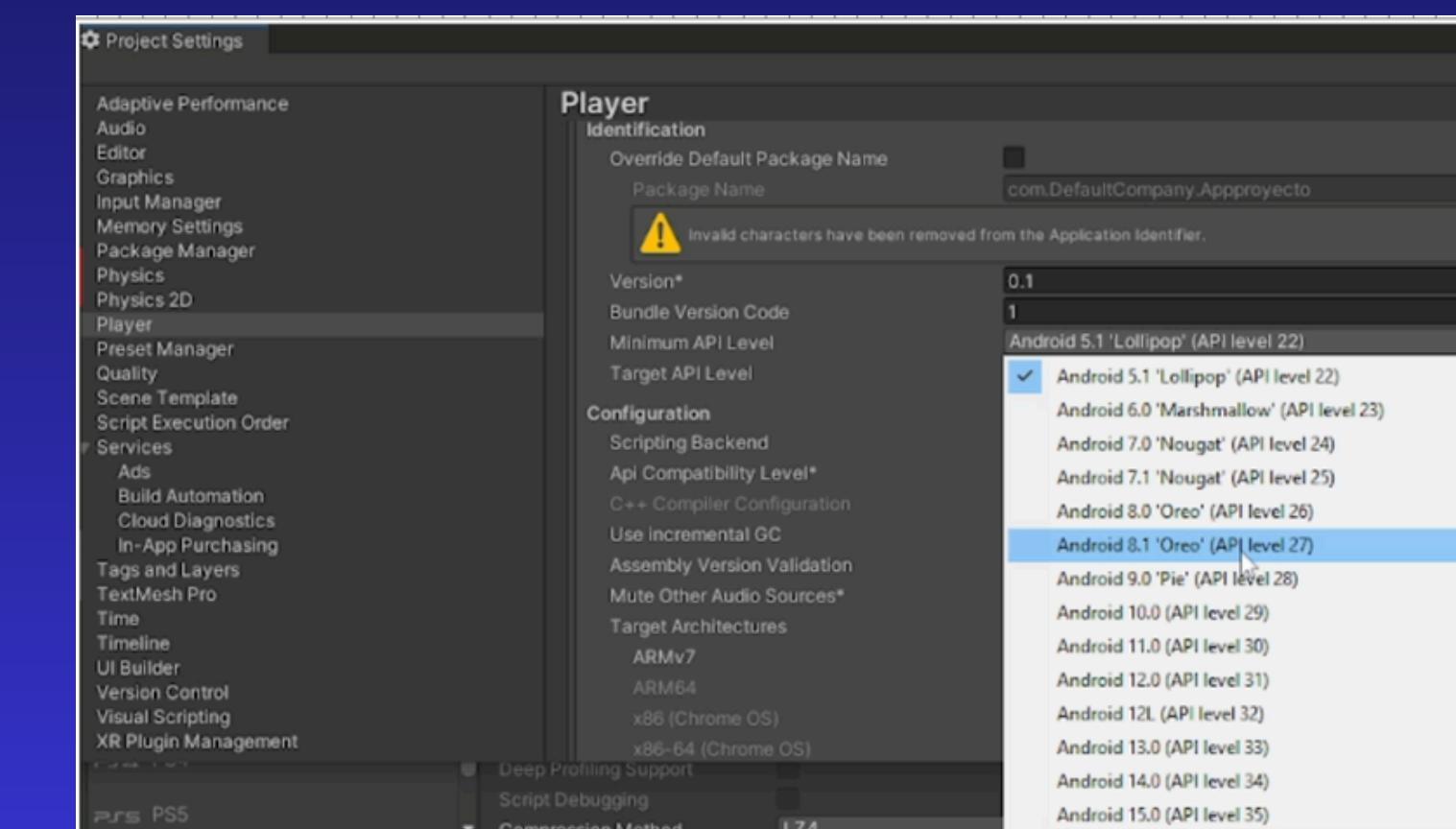
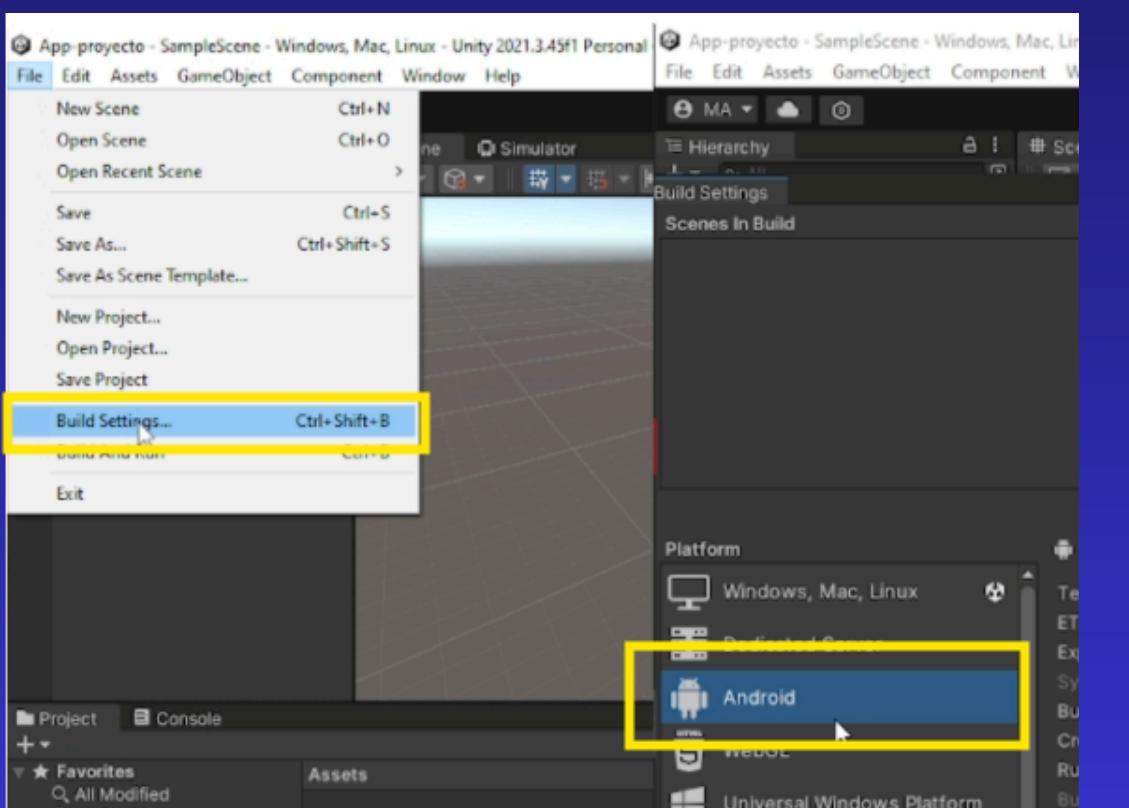
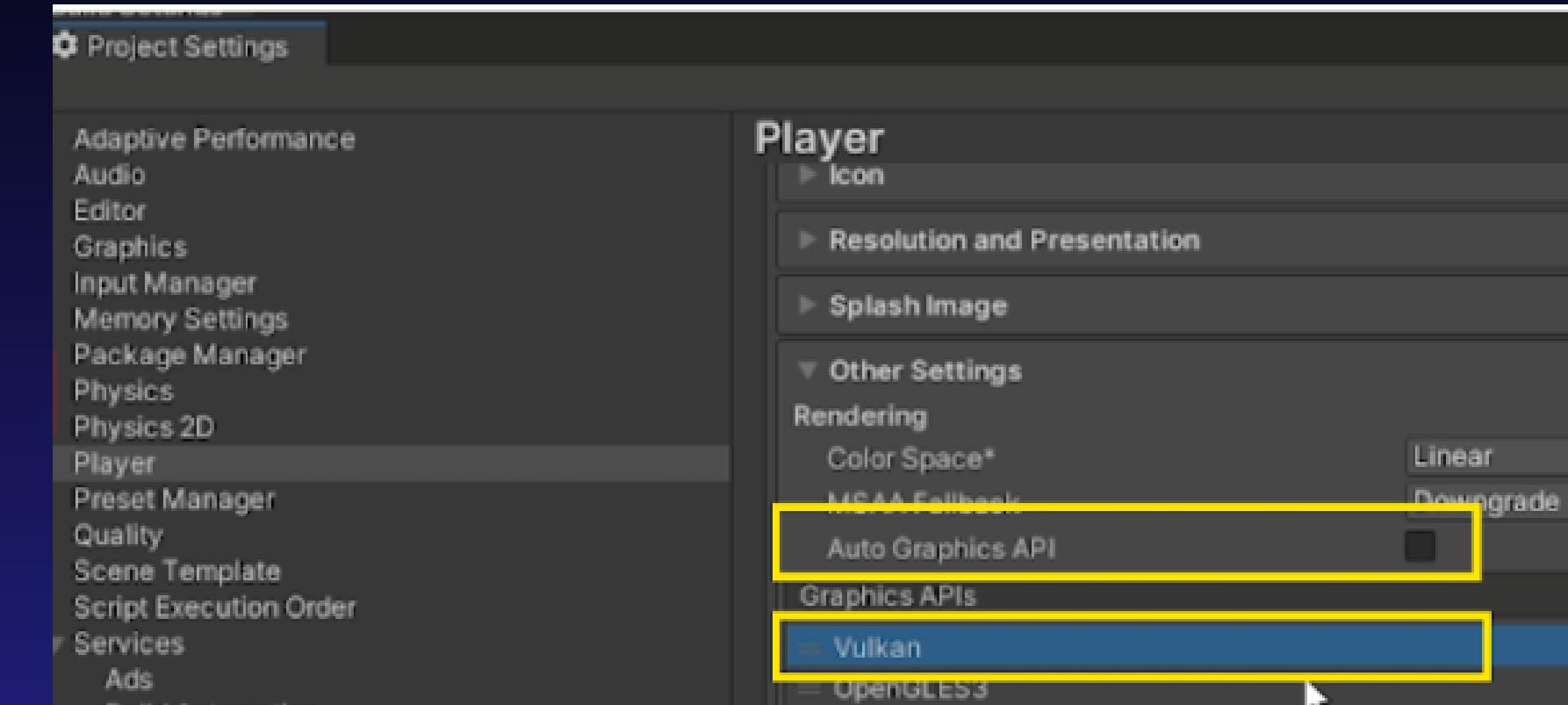
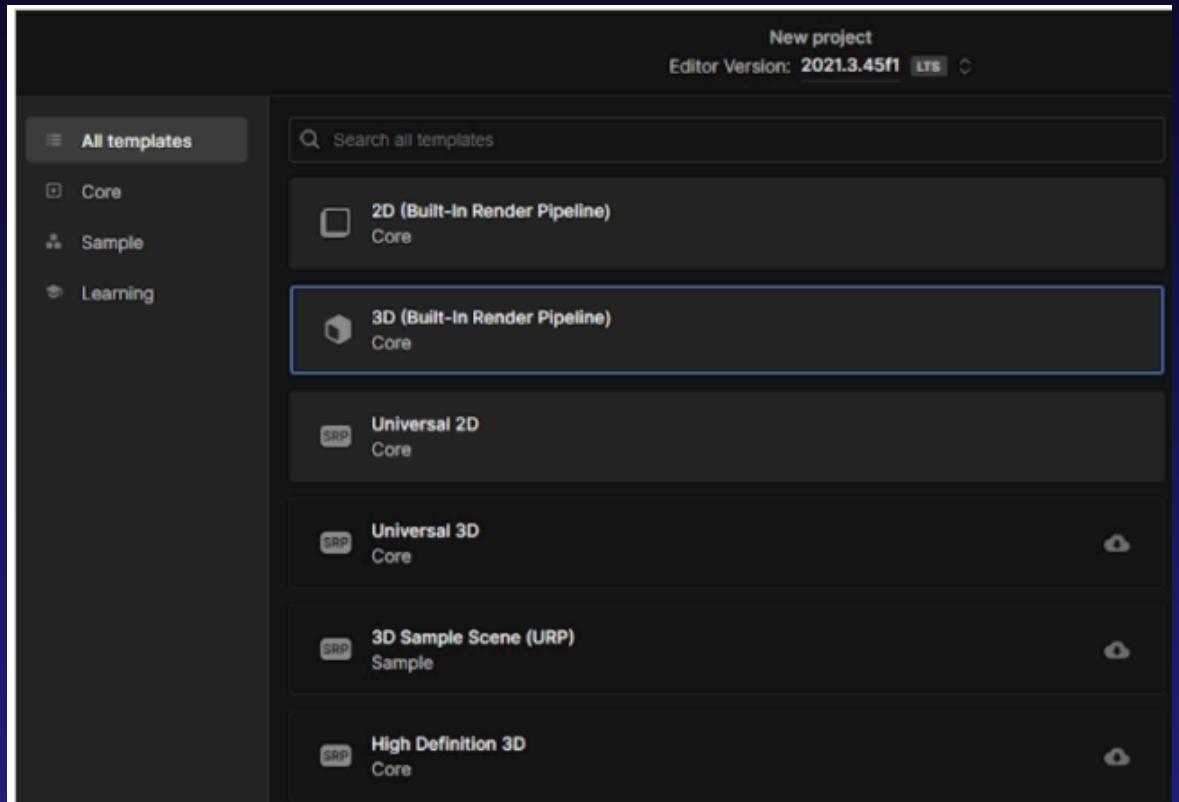
## Instalación de Unity





# DESARROLLO DEL ESCENARIO O MICROMUNDO

## Configuración del entorno de desarrollo





# DESARROLLO DEL ESCENARIO O MICROMUNDO

## Configuración del entorno de desarrollo

The screenshot shows the Unity Editor interface. The top bar displays the version: 2021.3.4f1 Personal <DX11>. The left sidebar has a 'Window' tab selected, showing options like Next Window, Previous Window, Layouts, Search, Unity Version Control, Asset Store, and Package Manager, which is currently active. The main area shows the Package Manager window with a list of packages under 'Features' and 'Packages'. The 'AR Foundation' package is selected and highlighted with a yellow box. The 'File' menu is open at the bottom, showing options like New Scene, Open Scene, Save, and Build Settings..., with 'Build Settings...' also highlighted with a yellow box. To the right, there's a 'XR Plug-in Management' panel.

File Edit Assets GameObject Component Window Help

Panels

Next Window Ctrl+Tab

Previous Window Ctrl+Shift+Tab

Layouts

Search

Unity Version Control

Asset Store

Package Manager

Asset Management

Tools

2021.3.4f1 Personal <DX11>

Package Manager

+ Packages: Unity Registry Sort: Name ↓

Features

- 3D Characters and Animation 4 packages
- AR 6 packages

Packages

- AR Foundation 4.2.10
- ARCore XR Plugin 4.2.10
- ARKit Face Tracking 4.2.10
- ARKit XR Plugin 4.2.10
- Core RP Library

App-proyecto - SampleScene - Andriod Project Settings

New Scene

Open Scene

Open Recent Scene

Save

Save As...

Save As Scene Template...

New Project...

Open Project...

Save Project

Build Settings...

Build And Run

Exit

Editor Graphics Input Manager Memory Settings Package Manager Physics Physics 2D Player Preset Manager Quality Scene Template Script Execution Order Services Ads Build Automation Cloud Diagnostics In-App Purchasing Tags and Layers TextMeshPro Time Timeline UI Builder Version Control Visual Scripting XR Plug-in Management ARCore ARKit

Initialize XR on Startup

Editor Play mode uses Desktop Platform Settings regardless of Active Build Target.

Plug-in Providers

ARCore

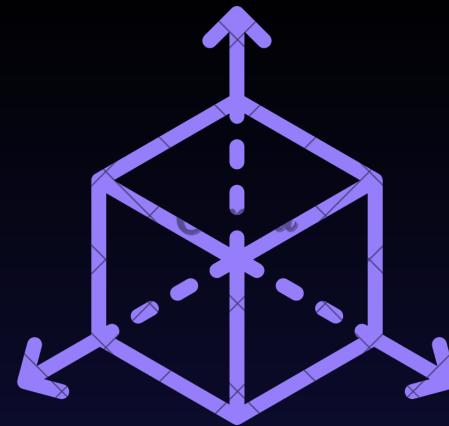
Oculus

Open XR

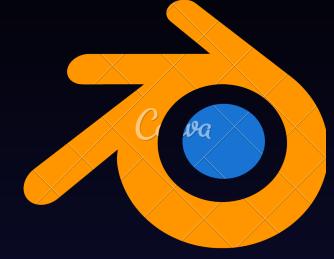
Unity Mock HMD

Information about configuration, tracking and migration can be found below.

[View Documentation](#)

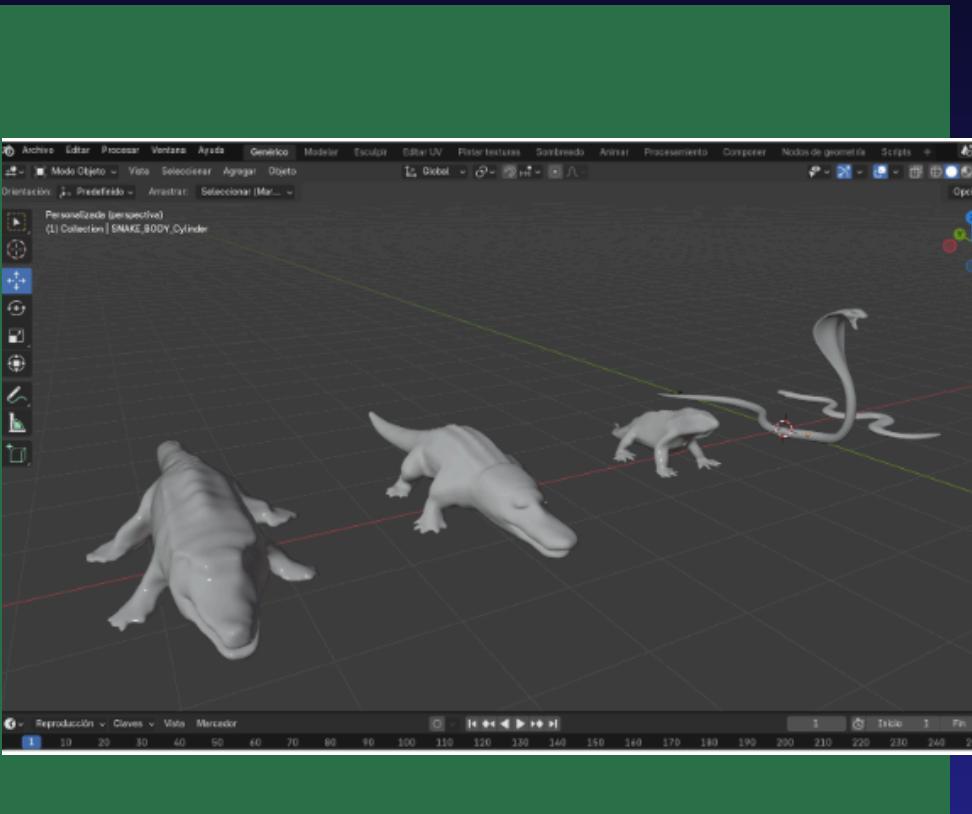


# Modelado de objetos



## A. reptiles

- Lagartija
- Iguana
- Cocodrilo
- Cobra
- Serpiente
- Caiman



## A. mamíferos

- Mapache
- Conejo
- Canguro
- Oso



## A. acuáticos

Se realizó la búsqueda de modelos tridimensionales y de sus texturas representativas asegurando un diseño realista.

- Tortuga Marina
- Tiburón
- Pulpo
- Medusa
- Tiburón nodriza
- Ballena jorobada
- Delfín
- Cangrejo

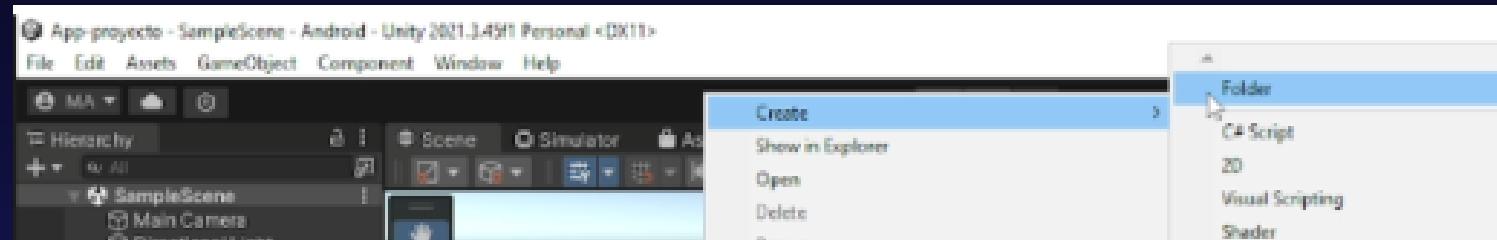
Se incluyeron mas elementos tales como su habitat natural para crear un entorno visualmente interactivo



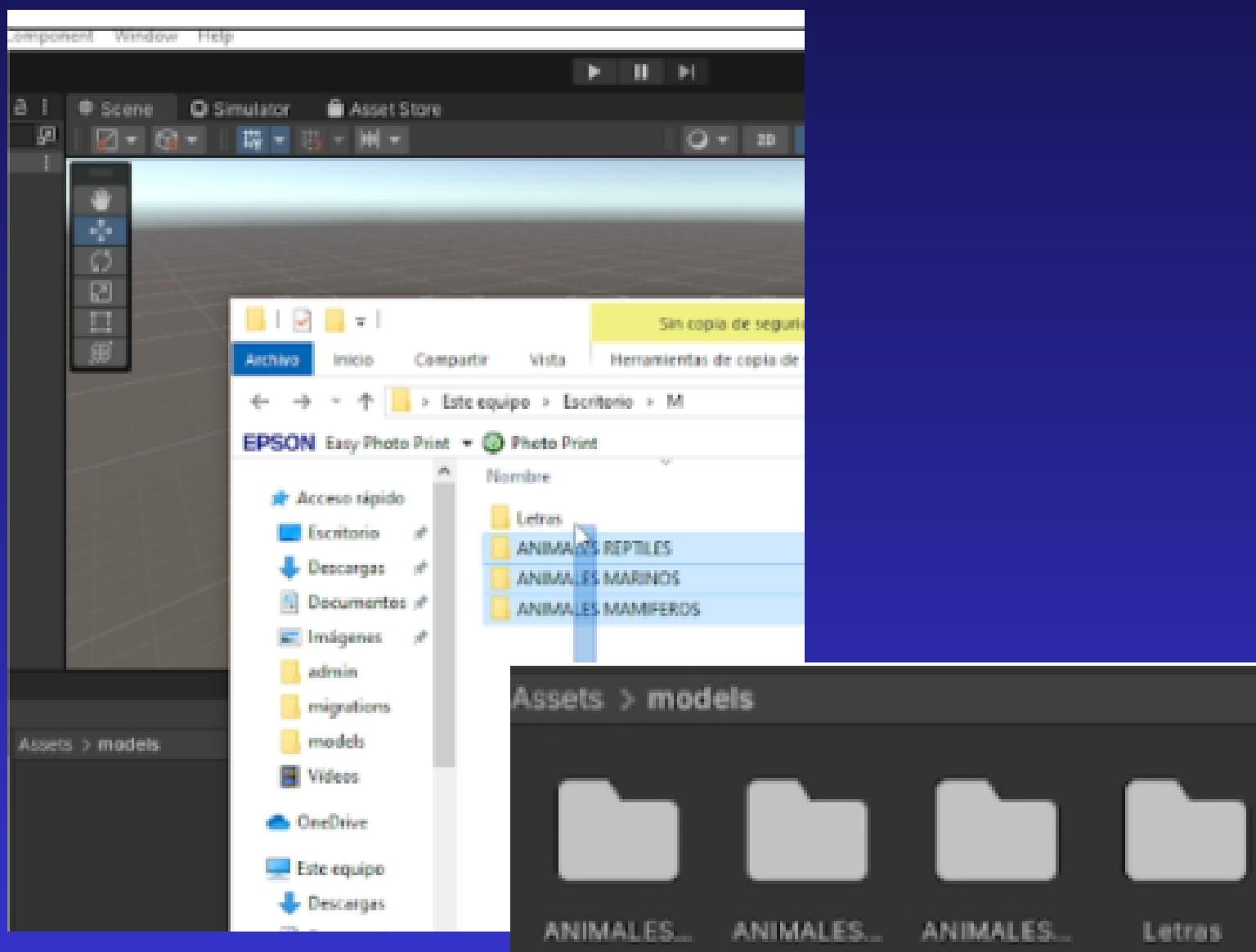


# Carga de modelos al proyecto

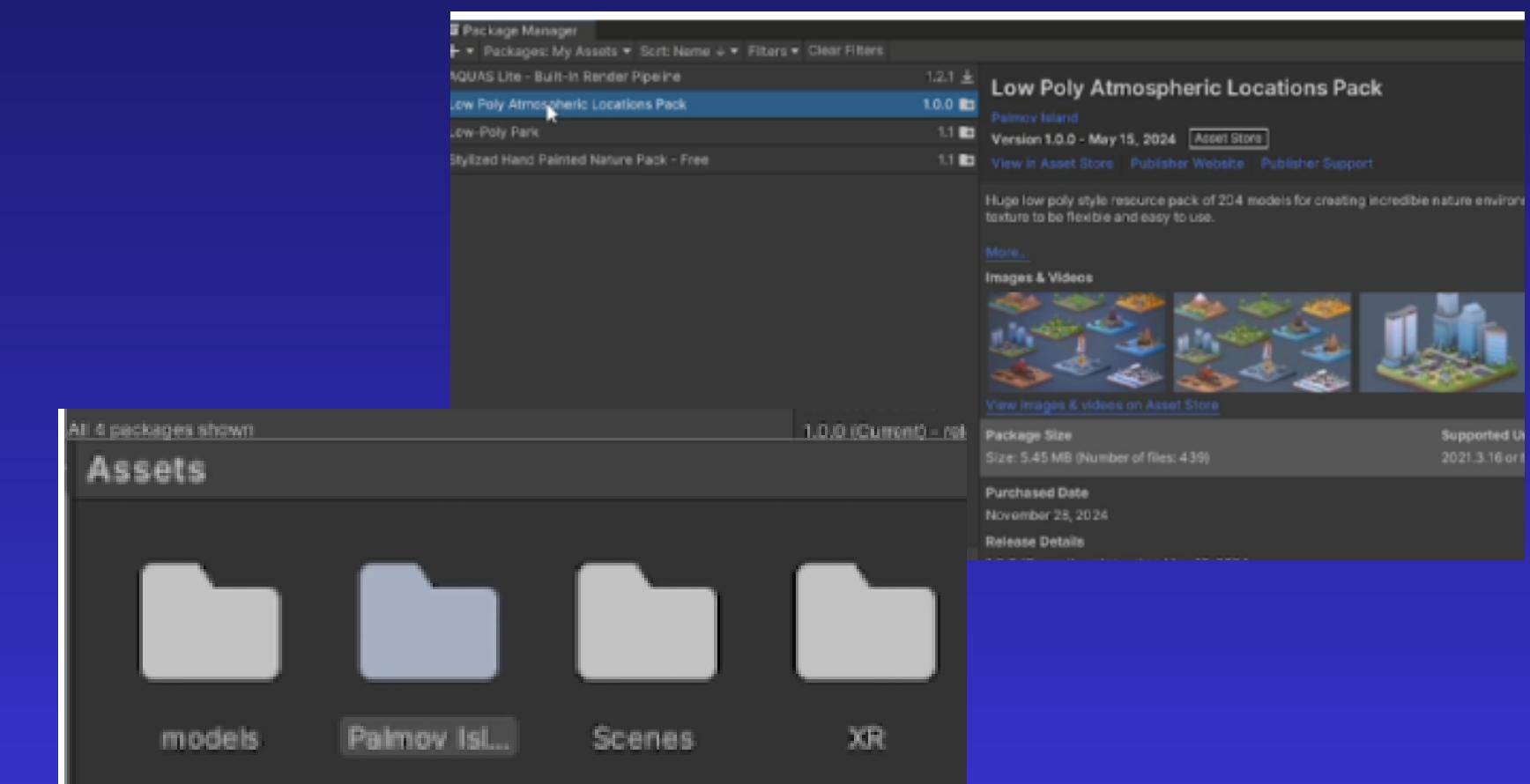
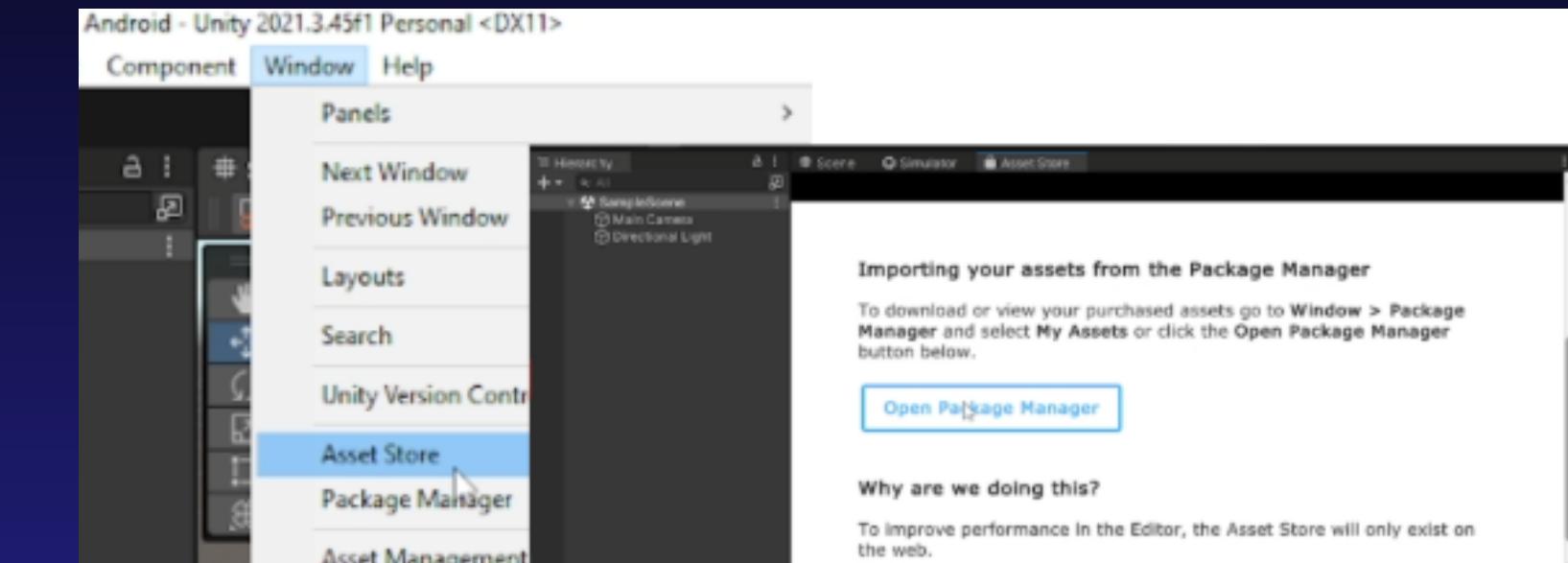
CREACIÓN DE UN NUEVO FOLDER LLAMADO MODELS



COPIAMOS TODOS LOS ELEMENTOS A LA CARPETA CREADA



Entramos a Asset Store



# Scripts

## CameraControllers.cs

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class CameraController : MonoBehaviour
6  {
7      // Start is called before the first frame update
8      void Start()
9      {
10      }
11
12      // Update is called once per frame
13      void Update()
14      {
15          transform.Translate(Input.GetAxis("Horizontal"), 0, Input.GetAxis("Vertical"));
16      }
17  }
```

## Portal.cs

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.Rendering;
5  using UnityEngine.XR.ARFoundation;
6
7  public class Portal : MonoBehaviour
8  {
9      public Material[] materials;
10     public Transform device; // Referencia al dispositivo o cámara
11     public ARAnchorManager anchorManager; // Referencia al AnchorManager de AR Foundation
12
13     private ARAnchor anchor;
14     private bool wasInFront;
15     private bool inOtherWorld;
16
17     void Start()
18     {
19         // Ancla el portal en la posición inicial
20         anchor = anchorManager.AddAnchor(new Pose(transform.position, transform.rotation));
21         if (anchor == null)
22         {
23             Debug.LogWarning("Failed to create anchor for the portal.");
24         }
25     }
26
27     void SetMaterials(bool fullRender)
28     {
29         var stencilTest = fullRender ? CompareFunction.NotEqual : CompareFunction.Equal;
30
31         foreach (var mat in materials)
32         {
33             mat.SetInt("_StencilTest", (int)stencilTest); // Configura correctamente el stencil
34         }
35     }
36
37     bool GetIsFront()
38     {
39         // Calcula si el dispositivo está frente al portal
40         Vector3 direction = device.position - transform.position;
41         return Vector3.Dot(transform.forward, direction) > 0;
42     }
43
44     void OnTriggerEnter(Collider other)
45     {
46         if (other.transform != device)
47             return;
48
49         wasInFront = GetIsFront();
50
51         // Mantén la posición del portal fija
52         if (anchor != null)
53         {
54             transform.position = anchor.transform.position;
55             transform.rotation = anchor.transform.rotation;
56         }
57     }
58
59
60     void OnTriggerStay(Collider other)
61     {
62         if (other.transform != device)
63             return;
64
65         bool isInFront = GetIsFront();
66
67         // Cambia de mundo si se cruza el portal
68         if (isInFront != wasInFront)
69         {
70             inOtherWorld = !inOtherWorld;
71             SetMaterials(inOtherWorld);
72         }
73
74         wasInFront = isInFront;
75
76     }
77
78     void OnDestroy()
79     {
80         // Restaura los materiales cuando el portal se destruye
81         SetMaterials(true);
82
83         // Elimina el anclaje
84         if (anchor != null)
85         {
86             anchorManager.RemoveAnchor(anchor);
87         }
88
89     }
90
91     void Update()
92     {
93         // Mantén el portal fijo al anclaje
94         if (anchor != null)
95         {
96             transform.position = anchor.transform.position;
97             transform.rotation = anchor.transform.rotation;
98         }
99     }
100 }
```

## Interdimensionalatransport:

```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.Rendering;
5
6  public class InterdimensionTransport : MonoBehaviour
7  {
8      // Start is called before the first frame update
9      public Material[] materials;
10     void Start()
11     {
12         foreach(var mat in materials){
13             mat.SetInt("_StencilTest", (int)CompareFunction.Equal);
14         }
15     }
16
17     void OnTriggerStay(Collider other){
18         if(other.name != "AR Camera"){
19             return;
20         }
21
22         if(transform.position.z > other.transform.position.z){
23             Debug.Log("Outside of other world");
24             foreach(var mat in materials){
25                 mat.SetInt("_StencilTest", (int)CompareFunction.Equal);
26             }
27         }else{
28             Debug.Log("Inside Other World");
29             foreach(var mat in materials){
30                 mat.SetInt("_StencilTest", (int)CompareFunction.NotEqual);
31             }
32         }
33     }
34 }
```

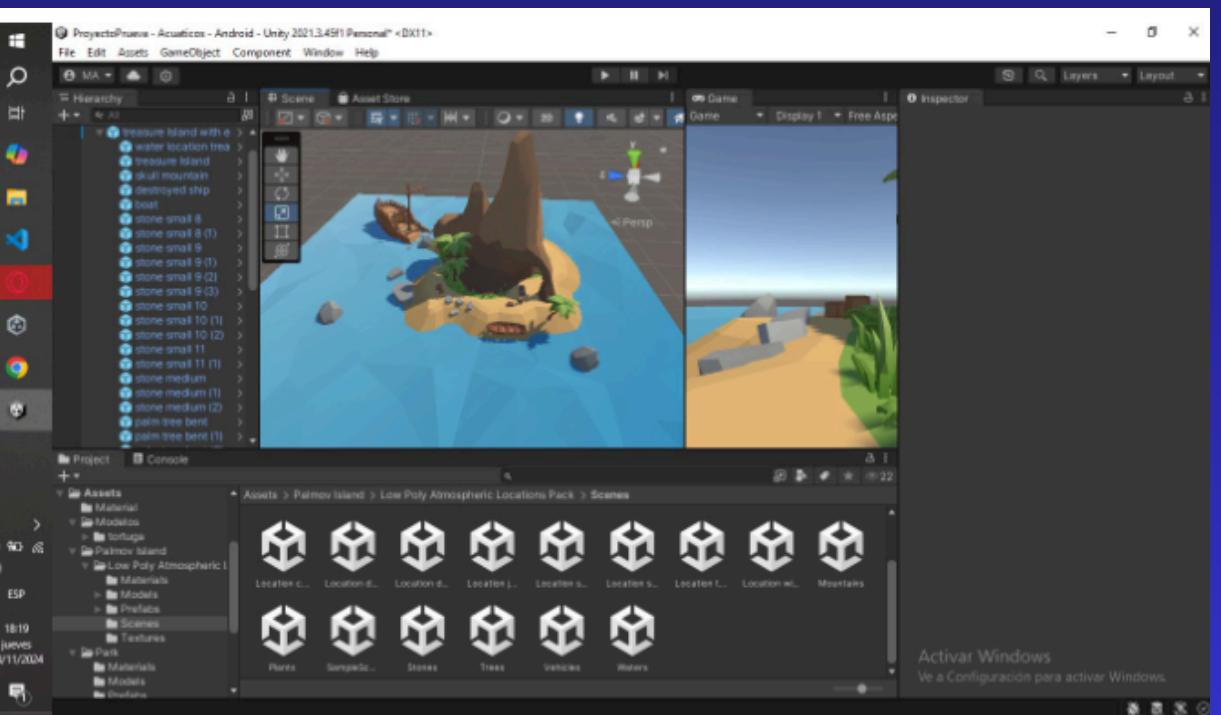


## StandarStencilFiltro.shader

```
Shader "Custom/Standard Stencil Filter"
{
    Properties
    {
        _Color("Color", Color) = (1,1,1,1)
        _MainTex("Albedo", 2D) = "white"
        _Cutoff("Alpha Cutoff", Range(0.0, 1.0)) = 0.5
        _Glossiness("Smoothness", Range(0.0, 1.0)) = 0.5
        _GlossMapScale("Smoothness Scale", Range(0.0, 1.0)) = 1.0
        [Enum(Metallic Alpha,0,Albedo Alpha,1)] _SmoothnessTextureChannel ("Smoothness texture channel", Float) = 0
        [Gamma] _Metallic("Metallic", Range(0.0, 1.0)) = 0.8
        _MetallicGlossMap("Metallic", 2D) = "white"
        [ToggleOff] _SpecularHighlights("Specular Highlights", Float) = 1.0
        [ToggleOff] _GlossyReflections("Glossy Reflections", Float) = 1.0
        _BumpScale("Scale", Float) = 1.0
        [Normal] _BumpMap("Normal Map", 2D) = "bump"
        _Parallax ("Height Scale", Range (0.005, 0.00)) = 0.02
        _ParallaxMap ("Height Map", 2D) = "black"
        _OcclusionStrength("Strength", Range(0.0, 1.0)) = 1.0
        _OcclusionMap("Occlusion", 2D) = "white"
        _EmissionColor("Color", Color) = (0,0,0)
        _EmissionMap("Emission", 2D) = "white"
        _DetailMask("Detail Mask", 2D) = "white"
        _DetailAlbedoMap("Detail Albedo x2", 2D) = "grey"
        _DetailNormalMapScale("Scale", Float) = 1.0
        [Normal] _DetailNormalMap("Normal Map", 2D) = "bump"
        [Enum(UV0,0,UV1,1)] _UVSet ("UV Set for secondary textures", Float) = 0
        [Enum(Equal,3,NotEqual,6)] _StencilTest("Stencil Test", Int) = 6
        // Blending state
        [HideInInspector] _Mode ("__mode", Float) = 0.0
        [HideInInspector] _SrcBlend ("__src", Float) = 1.0
        [HideInInspector] _DstBlend ("__dst", Float) = 0.0
        [HideInInspector] _ZWrite ("__zw", Float) = 1.0
    }

    CGINCLUDE
        #define UNITY_SETUP_BRDF_INPUT MetallicSetup
    ENDCG
}
```

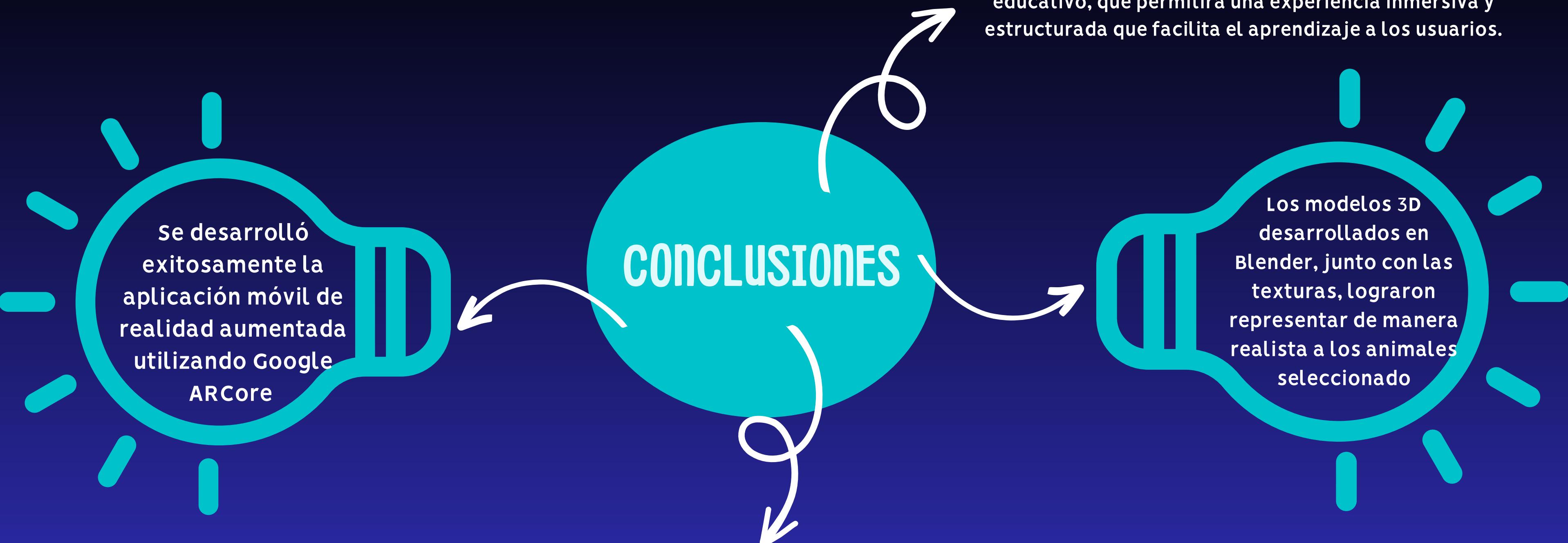
Implementación:



# Filtro



La implementación de portales de realidad aumentada para cada categoría de animales) resultó ser una estrategia efectiva para organizar el contenido educativo, que permitirá una experiencia inmersiva y estructurada que facilita el aprendizaje a los usuarios.



La aplicación ofrece una alternativa segura y efectiva para la educación ambiental infantil, promoviendo la concientización sobre la biodiversidad mediante realidad aumentada (RA), sin requerir acceso físico a zoológicos o reservas, destacándose como una herramienta educativa potencial.

¡muchas gracias!