Perfect — here's **everything**, structured so you can **build once and ship everywhere** without painting yourself into a corner.

I'll give you **4 deliverables** in one response:

1. **Project architecture & folder template**
2. **Unity setup checklist (step-by-step, no guesswork)**
3. **Feature parity matrix (Meta vs Pico vs SteamVR)**
4. **Store review & submission survival guide**

You can literally follow this top to bottom.

1️⃣ PROJECT ARCHITECTURE (THIS SAVES YOU MONTHS)

🎯 Core Principle

> **Gameplay must never know what headset it's running on.**

Everything platform-specific lives behind interfaces.

📁 Recommended Unity Folder Structure

```
Assets/
├── _Core/
│   ├── Gameplay/
│   ├── Combat/
│   ├── Movement/
│   ├── Interaction/
│   └── Systems/
│
├── _XR/
│   ├── OpenXR/
│   ├── Input/
│   ├── Locomotion/
│   └── HandsControllers/
│
```

```
├── _Platform/
│   ├── Meta/
│   │   └── MetaPlatformServices.cs
│   ├── Pico/
│   │   └── PicoPlatformServices.cs
│   ├── Steam/
│   │   └── SteamPlatformServices.cs
│   └── IPlatformServices.cs
│
├── _UI/
│   ├── WorldSpace/
│   ├── HUD/
│   └── Menus/
│
├── _Networking/
│   ├── Auth/
│   ├── Progression/
│   └── Matchmaking/
│
├── _Performance/
│   ├── LOD/
│   ├── Foveation/
│   └── QualityProfiles/
│
└── _ThirdParty/
```

🔌 Platform Interface Example

```
public interface IPlatformServices
{
   string GetUserId();
   void UnlockAchievement(string id);
   void PurchaseItem(string sku);
}
```

Each platform implements this **without touching gameplay code**.

2️⃣ UNITY SETUP CHECKLIST (OPENXR-FIRST)

✅ Unity Version

- **Unity 2022 LTS or newer**
- URP template

✅ Required Packages

Install via Package Manager:

- OpenXR Plugin
- XR Interaction Toolkit
- XR Hands (optional)
- Input System
- Universal Render Pipeline (URP)

❌ Do NOT install:

- Oculus Integration
- SteamVR Plugin (unless PC-only extras)

✅ OpenXR Settings (CRITICAL)

Project Settings → XR Plug-in Management

- Enable OpenXR for:
  - Android (Quest + Pico)
  - Windows (SteamVR)

OpenXR Features

- Controller profiles (Quest, Pico, Index)
- Eye Gaze (optional)
- Hand Tracking (optional)

✅ Input Actions (Abstracted)

Create actions like:

- `Grab`
- `PrimaryAction`
- `SecondaryAction`
- `Move`
- `Turn`
- `Menu`

Bind them via OpenXR — **never per-controller**.

✅ Rendering & Performance

- URP Forward Renderer
- MSAA: 4x (mobile), 8x (PC)
- Single Pass Instanced
- Fixed Foveated Rendering (mobile only)

3️⃣ FEATURE PARITY MATRIX (REALITY CHECK)

Feature	Quest	Pico	SteamVR
Standalone	✅	✅	❌
PC VR	❌	❌	✅
Hand Tracking	✅	⚠️ Limited	❌
Full Body Tracking	❌	❌	✅
High-End Graphics	⚠️	⚠️	✅

Mods	❌	❌	✅
Cross-play	✅	✅	✅

⚠️ = design fallback required

🎮 Design Rule

> **Design for standalone first, enhance for PC later.**

If it runs smoothly on Quest/Pico:

- It will scream on SteamVR

4️⃣ STORE REVIEW SURVIVAL GUIDE

🟢 Meta Quest Store

Biggest killers:

- Frame drops below 72 Hz
- No comfort options
- Missing boundary handling

Required:

- Teleport OR smooth locomotion toggle
- Snap turn option
- Seated + standing support
- Safety warning on launch

🔵 Pico Store (China + Global)

Biggest killers:

- Crashes on launch

- No Chinese language support (China store)
- Improper permission usage

Required:

- Performance stability > visuals
- Controller diagrams
- No Meta branding anywhere

🟣 SteamVR

Biggest killers:

- Bad controller mappings
- No rebinding support
- Poor default locomotion

Required:

- Steam Input bindings
- Room-scale + seated support
- Clear comfort warnings

5️⃣ MULTIPLAYER & ACCOUNTS (IF YOU USE THEM)

✅ Correct Way

- Supabase / PlayFab backend
- Your own user ID
- Optional platform ID linking

❌ Wrong Way

- Meta account as primary ID
- Pico ID as primary ID
- Steam ID-only multiplayer

This is **mandatory** if you want:

Quest ↔ Pico ↔ Steam cross-play

---

6 TESTING MATRIX (NON-NEGOTIABLE)

Minimum devices:

- Quest 2 or 3
- Pico 4
- SteamVR headset (Index or Vive)

Test:

- 30+ minute sessions
- Guardian boundary loss
- Low battery behavior
- Sleep / resume
- Controller disconnects

---

7 FINAL GOLDEN RULES

- OpenXR only
- No platform SDK in gameplay
- Mobile-first performance
- Interfaces everywhere
- Comfort > realism

This is the **exact architecture used by studios shipping globally**.

NEXT STEPS (YOUR CHOICE)

I can:

1. Give you a **starter Unity project (OpenXR + XR Toolkit pre-wired)**
2. Design your **movement/combat system for all headsets**

3. Map this directly onto your **AR + Mapbox + VR world roadmap**
4. Create a **store-ready checklist per platform**

Tell me which one you want first — **this is the way**.