

WildXR Auto Launch Installation Guide

Purpose & Context

This guide installs the WPS Auto Launch application that automatically starts WildXR when a headset powers on. Auto Launch eliminates the need for hand controllers in public installations, creates a seamless user experience, and enables remote monitoring of headset uptime. This process transforms a standard VR headset into a kiosk-ready system that requires minimal user interaction to begin the conservation experience.

⚠ Third Party Software: Third Party hardware and software (Meta, ArborXR) are outside WPS control. Meta updates may cause unexpected functionality changes in VR systems. WPS monitors Meta and ArborXR releases to inform users of potential impacts and changes.

Prerequisites

- Meta Quest Developer Hub fully installed and configured with at least one headset connected and USB debugging authorized (*see MQDH Setup Guide*)
- VR headset with Developer mode enabled in both Oculus app and headset settings
- WildXR application already installed and successfully run at least once on target headset
- **autolaunchping_v.1.0.0.apk** file (*provided by WPS*)
- Meta account with developer privileges enabled by WPS
- USB cable for connecting headset during installation

Quick Overview (for experienced users)

1. Connect target headset to computer (*or verify existing MQDH connection*)
2. Access MQDH app management and install **autolaunchping** APK
3. Launch Auto Launch app and configure WildXR integration
4. Test automatic startup functionality

Detailed Steps

Preparing for Auto Launch Installation

1. Connect headset for file transfer

- Launch **Meta Quest Developer Hub**
- Power on Quest Headset
- Connect Headset to Computer with USB cable

2. Verify headset connection

- Ensure your target headset appears as "**Active**" in MQDH devices list
- *New headsets will typically require the same USB debugging authorization process completed during MQDH setup (see MQDH Setup Guide)*

- Confirm the headset shows no error indicators or connectivity issues

3. Confirm WildXR readiness

- Verify that WildXR is installed on the target headset
- Ensure WildXR has been launched successfully at least once
- *Auto Launch cannot register applications that have never run on the system*
- Test WildXR startup manually if uncertain about its operational status

Installing the Auto Launch Application

4. Access application management

- In MQDH, click the **folder icon** in the left sidebar
- *This opens the file and application management interface*
- Under **"On Device"** section, click the **"Apps"** folder icon
- *This displays currently installed applications on your headset*

5. Add the Auto Launch application

- Click **"Add Build"** button in the upper right corner
- *File browser opens to locate your APK installation file*
- Navigate to the location where you saved **autolaunchping_v.1.0.0.apk**
- Select the APK file and click **"Open"**

6. Monitor installation progress

- MQDH displays installation progress indicators
- Allow the file to install completely before proceeding
- *Installation typically takes 30-60 seconds depending on system performance*
- You should see **com.wps.autowildxrping** appear in your apps list when complete

Configuring Auto Launch Integration

7. Launch Auto Launch application

- Locate **com.wps.autowildxrping** in your MQDH apps list
- Click the **three-dot menu icon** to the right of the application name
- Select **"Launch App"** from the dropdown menu
- Put on the VR headset to complete configuration

8. Complete Auto Launch setup

- WildXR may launch immediately if Auto Launch detects existing configuration
- If WildXR launches automatically, remove headset and power cycle to test functionality
- If you see a **"WildXRAutoPing"** window instead, continue with manual configuration
- *Look for a tile resembling an alien or unusual icon on the Quest menu bar if the window is not immediately visible*

9. Finalize configuration

- In the **WildXRAutoPing** window, click "**Save URL**" button
- *This creates the permanent link between Auto Launch and WildXR*
- WildXR should launch immediately after clicking **Save URL**
- Remove headset and power cycle the device to verify automatic startup functionality

Understanding the Purpose of Auto Launch Technology

Before beginning installation, it helps to understand what Auto Launch accomplishes and why it requires this specialized installation process. Traditional VR applications require users to manually navigate menus and select applications using controllers. This works well for personal use but creates barriers in public conservation settings where visitors might be unfamiliar with VR interfaces or where controllers might be lost or damaged.

The WPS Auto Launch application operates at the Android system level, registering itself to trigger when the headset completes its startup sequence. Think of it like setting a default web browser on your computer, except instead of handling web links, Auto Launch handles the headset startup event. When the system boots, Auto Launch immediately signals WildXR to begin, creating a seamless transition from power-on to conservation content.

This system-level integration is why the installation requires Meta Quest Developer Hub rather than standard app installation methods. We are essentially modifying the headset's behavior at a deeper level than normal applications access.

Troubleshooting

MQDH shows no connected devices:

- Verify you completed the full MQDH setup including USB debugging authorization
- Check that your target headset is the same one configured during MQDH setup
- If using a different headset, it will need the same USB debugging authorization process
- Ensure the headset remains powered and connected throughout the installation

Auto Launch installation fails:

- Confirm that WildXR is properly installed and has been run successfully at least once
- Verify headset has sufficient storage space for additional applications
- Check that **com.wps.autowildxrping** appears in MQDH apps list before proceeding with configuration
- Ensure the **autolaunchping** APK file is not corrupted by trying installation on a different headset


Auto Launch functionality not working after installation:


- Verify **com.wps.autowildxrping** appears as running in MQDH three-dot menu
- Confirm WildXR launches manually to ensure application integrity
- Test complete power cycle rather than sleep/wake to verify startup behavior
- Check that "**Save URL**" was clicked during configuration and that WildXR launched afterward


WildXRAutoPing window not visible:


- Look for an unusual or alien-like icon on the Quest menu bar
- Try removing and putting the headset back on to refresh the interface
- Ensure Auto Launch app was actually launched from MQDH rather than just installed
- Check that WildXR has been run at least once before Auto Launch configuration

Important Notes

 **Developer Account Requirements:** Your Meta account must have verified developer status with Wildlife Protection Solutions before any custom application installation will succeed.

 **System Compatibility:** Auto Launch requires headsets with developer mode properly enabled in both the Oculus mobile application and the headset's internal settings under **Advanced > Developer**.

 **Account Matching:** The Meta account used for MQDH must exactly match the account used to configure the headset. Mismatched accounts will prevent device recognition and application installation.

 **WildXR Prerequisites:** WildXR must be successfully installed and launched at least once before Auto Launch installation. Auto Launch cannot register an application that the system does not recognize.

Verification and Testing

After completing installation, verify Auto Launch functionality through comprehensive testing. Power down the headset completely using the power button, wait 30 seconds, then power on again. Auto Launch should trigger WildXR startup without any manual intervention or controller use. If manual navigation is still required, the registration process was not completed successfully and should be repeated from step 5.