Instalacja Unity i konfiguracja nowego środowiska

Windows / Ubuntu; nowy projekt Universal 3D (URP); pakiet WiRR

Odbiorca: student pracujący dobrowolnie na własnym PC lub administrator przygotowujący stanowisko. Nie jest to zadanie oceniane.

Najkrótsza droga na zajęciach: jeżeli na komputerze jest już gotowy projekt WiRR, **otwórz go, nie instaluj Unity ponownie**. Instrukcja poniżej służy tylko utworzeniu nowego środowiska lub odtworzeniu go na własnym komputerze. Do wszystkich obowiązkowych zadań wystarcza przygotowany komputer Ubuntu lub Windows.

1. Instalacja Unity Hub i edytora

1. **Windows 10/11 (64-bit):** pobierz *Unity Hub* wyłącznie z https://unity.com/download, uruchom instalator i zaloguj się do swojego Unity ID. Na własnym komputerze aktywuj dostępną, odpowiednią licencję (np. Unity Personal); na stanowisku uczelni korzystaj z licencji przygotowanej przez administratora.
2. **Ubuntu (64-bit, x86_64):** skorzystaj z oficjalnej instrukcji https://docs.unity.com/en-us/hub/install-hub-linux. Dodaj tam opisane **podpisane repozytorium Unity Hub**, następnie zainstaluj Hub przez menedżer pakietów: `sudo apt update` i `sudo apt install unityhub`. Nie uruchamiaj Unity jako root. Unity Editor na Linux ARM64 nie jest obsługiwany.
3. W **Unity Hub** > **Installs** > **Install Editor** wybierz **dokładną wersję Unity 6000.6.x wskazaną przez prowadzącego**. Bieżąca macierz kursu wymienia `6000.6.5f1`, ale przed instalacją porównaj ją z wersją projektu stanowiskowego. Nie wybieraj po prostu najnowszej wersji.
4. Przy standardowym projekcie PC wystarcza sam edytor. **Nie instaluj na zapas** Android SDK, Meta XR, ROS 2, Gazebo, Dockera ani Android Studio. Android Build Support, SDK/NDK i OpenJDK dodaje się w Hub tylko na komputerze, na którym rzeczywiście buduje się APK.

2. Utworzenie nowego projektu Unity

1. W **Unity Hub** > **Projects** > **New project** wybierz szablon **Universal 3D (URP)** dla zatwierdzonego edytora. Nie używaj szablonu Built-in 3D ani HDRP.
2. Nazwij projekt np. `WiRR-Labs` i zapisz go w lokalnym katalogu, w którym masz prawo tworzyć pliki. Na komputerze współdzielonym trzymaj projekty i raporty różnych osób osobno.
3. Poczekaj na zakończenie importu. W **Window** > **General** > **Console** skontroluj, czy nie ma błędów kompilacji; w **Edit** > **Project Settings** > **Player** sprawdź, że aktywne jest **Input System Package (New)** według ustawień pakietu (Unity może zaproponować restart edytora). Pozostałe ustawienia URP pozostaw zgodne z szablonem.

3. Dodanie pakietu WiRR do nowego projektu

1. Zainstaluj klienta **Git** tylko jeśli nowy projekt będzie pobierał pakiet z Git URL, a następnie sprawdź w terminalu `git --version`. Na Windows Git musi znajdować się w `PATH`. Na przygotowanym PC uczelnianym ta czynność należy do administratora.
2. W Unity otwórz **Window** > **Package Manager** > + > **Install package from git URL** i wpisz wyłącznie publiczny adres dla studentów:

   https://github.com/KIA-students/wirr.git#upm

3. Wybierz **Install**. Poczekaj na rozpoznanie `pl.prz.kia.wirr` w pakietach projektu. W menu **WiRR** > **Narzędzia kursu** sprawdź brakujące zależności **tylko dla aktualnego laboratorium**. Jeśli nie widzisz sceny, użyj **WiRR** > **Misje i odkrycia** albo importu odpowiedniej próbki w **Package Manager** > **Samples**, zgodnie z wersją pakietu.
4. Otwórz przygotowaną scenę Lab 01, uruchom **Play**, porusz kamerą i sprawdź podstawową interakcję. Zapisz scenę w `Assets/WiRR/`, a nie w `Library/PackageCache`.

4. Kontrola gotowości i raportów

Sprawdź	Oczekiwany rezultat
Wersja edytora i URP	Projekt otwarty bez błędów; poprawny szablon Universal 3D.
Pakiet WiRR i scena	Menu WiRR działa; Lab 01 uruchamia się w Play.
Profil i zapis	**WiRR** > **Student** > **Profil i grupa**; możesz zapisać własną scenę oraz lokalny raport.
Tryb urządzenia	Bez gogli: tryb desktop/symulacja. Nie oznaczaj go jako realne VR lub AR.

Co oddajesz? Ta instrukcja **nie wymaga sprawozdania ani oceny**. Raport dotyczący eksperymentu oddaje się według właściwej instrukcji Lab 01–08. Gdy nie działa Git URL, sprawdź `git --version` i *pierwszy* błąd w Unity Console; na zajęciach zamiast naprawiać pół systemu skorzystaj z przygotowanego projektu.

Dokumentacja: https://docs.unity.com/en-us/hub/install-hub-linux; https://github.com/KIA-students/wirr.

5. Diagnostyka i kryterium odbioru

- Zapisz wersję edytora (Help > About Unity), nazwę szablonu URP i numer wersji pakietu WiRR. Sprawdź, czy wskazany kanał `#upm` jest rzeczywiście opublikowany i dostępny; brak publikacji oznacza blokadę instalacji, a nie błąd studenta.
- W Console zapisz pierwszy błąd, nie całą kaskadę późniejszych komunikatów. Jeśli pakiet nie jest dostępny, użyj zatwierdzonego projektu stanowiskowego.
- Odbiór: czysty start projektu, brak błędów kompilacji, uruchomienie przykładu Lab01, zapis sceny i poprawne oznaczenie trybu PC/SIM.