

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ЛЬВІВСЬКА ПОЛІТЕХНІКА  
КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

## **МЕТОДИЧНІ ВКАЗІВКИ**

---

**до лабораторної роботи № 1 з дисципліни «Віртуальна реальність»  
для студентів напряму підготовки “Інженерія програмного забезпечення”**

**Львів 2025**

# Завдання

---

Налаштувати WebXR середовище для роботи з веб API доповненої реальності.

## 1. Необхідні інструменти розробки

---

1.1. Установити середовище розробки Cursor: <https://www.cursor.com/>

1.2. Для управління версіями Node.js -- Установити NVM: <https://github.com/nvm-sh/nvm>

1.2.1 Версія Node.js для виконання лабораторних робіт - LTS 22.13.1.

1.2.2 У випадку проблеми зі зміною версії Node.js за допомогою NVM, наприклад неможливості перейти з system на LTS, реініціалізацію можна виконати наступним чином:

```
nvm deactivate && nvm unload && source ~/.nvm/nvm.sh && nvm use --lts
```

1.3. Установити Node.js, останню LTS версію, за допомогою NVM.

```
nvm install --lts  
nvm use --lts
```

1.4. Для управління Node.js залежностями в проекті -- Установити PNPM: <https://www.npmjs.com/get-npm>

1.4.1. Варіант 1. Використовувати corepack для встановлення PNPM.

```
npm install --global corepack@latest  
corepack enable pnpm
```

1.4.2. Варіант 2. Використовувати npm для встановлення PNPM.

```
npm install -g pnpm
```

1.5. Для створення HTTPS ендпойнту онлайн -- зареєструватись в сервісі ngrok та встановити ngrok CLI: <https://ngrok.com/>

```
curl -sSL https://ngrok-agent.s3.amazonaws.com/ngrok.asc \  
| sudo tee /etc/apt/trusted.gpg.d/ngrok.asc >/dev/null \  
&& echo "deb https://ngrok-agent.s3.amazonaws.com buster main" \  
| sudo tee /etc/apt/sources.list.d/ngrok.list \  
&& sudo apt update \  
&& sudo apt install ngrok
```

```
ngrok config add-authtoken <YOUR_AUTH_TOKEN>  
# Test ngrok  
ngrok http localhost:3000
```

1.6 Склонувати локально цей репозиторій і відкрити в Cursor перше завдання ЛР-01-01 `ar-practice-2025/lab-2025-01-01` .

## 2. Нотатки до виконання.

2.1. NVM - Node Version Manager, підсистема для управління версіями Node.js. NVM необхідно для того, щоб можна було легко перемикатися між різними версіями Node.js. В рамках цього та інших курсів та лабораторних робіт може виникнути потреба мати доступ до різних версій Node.js. NVM дозволяє легко перемикатися між різними версіями рантайму. Наша робоча версія Node.js в рамках цього курсу - LTS 22.13.1.

2.2. PNPM - це аналог npm, ефективний пакет-менеджер, який використовується для управління залежностями в проектах. PNPM є альтернативою npm, яка має більш швидкі та ефективні алгоритми для керування залежностями. За бажанням можна використовувати інші пакет-менеджери, такі як Yarn, npm, vite тощо, але методичні вказівки та завдання будуть розроблені з урахуванням PNPM.

2.3. Cursor - це IDE, яка інтегрує VSCode та AI, і дозволяє використовувати промпти штучного інтелекту для написання коду. Cursor є альтернативою VSCode, методичні вказівки не роблять особливих вимог до використання Cursor, але він може бути корисним для написання коду. Як альтернатива може використовуватись VSCode.

## 3. Виконання завдання.

3.1. Відкрити в Cursor проект `ar-practice-2025/lab-2025-01-01` .

Ця директорія має бути робочою для наступних команд, якщо не вказано інше.

3.2. Переконатись, що версії інструментів встановлені і відповідають очікуваням. В терміналі виконати команди:

```
# Node.js
node --version
# v22.13.1
# У випадку проблеми з версією:
npm install --lts
npm use --lts

# PNPM
pnpm --version
# v10.2.1
# У випадку проблеми з версією:
corepack enable pnpm
corepack use pnpm@latest-10
```

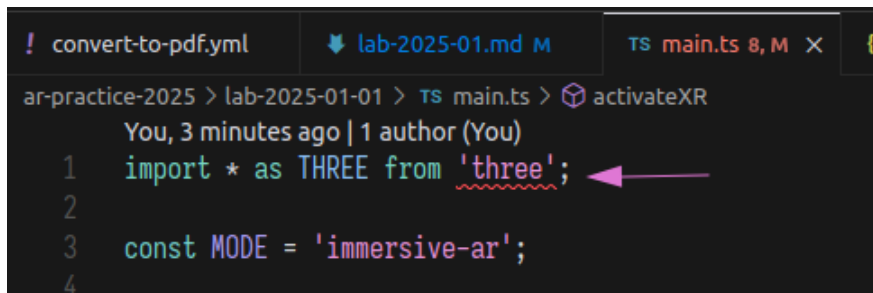
3.3. Відкрити файл `package.json` і переконатись, що `corepack` коректно встановив `"packageManager"` та `"devDependencies" -> "typescript"` .

### 3.4. Імплементувати скрипти для запуску проекту.

```
"scripts": {  
  "start": "ts-node main.ts"  
}
```

### 3.5. Відкрити файл `main.ts` і переконатись, що він не містить помилок.

3.5.1. Перша помилка, яку ви маєте побачити -- це помилка з імпортом Three.js:



```
! convert-to-pdf.yml lab-2025-01.md M TS main.ts 8, M X {}  
ar-practice-2025 > lab-2025-01-01 > ts main.ts > activateXR  
You, 3 minutes ago | 1 author (You)  
1 import * as THREE from 'three';  
2  
3 const MODE = 'immersive-ar';  
4
```

Вирішимо проблеми з імпортом наступним чином:

```
pnpm install three@0.172.0  
pnpm install @types/three@0.172.0  
pnpm install @types/webxr
```