

## Лабораторна робота №7. Робота з VR-контролерами: механіки захоплення об'єктів та Haptic Feedback

### 1. Мета роботи

- Ознайомитися з архітектурою введення даних **WebXR Input Sources**.
- Реалізувати інерційну механіку захоплення (**grabbing**) та маніпуляції об'єктами.
- Опанувати програмування тактильного відгуку (**Haptic Feedback**) через Gamepad API.

### 2. Теоретична частина

#### WebXR Input Profiles

Браузер не знає заздалегідь, який пристрій підключено (Oculus Touch, Index Knuckles чи Vive Wand). Замість цього він використовує **Input Profiles**. Кожен контролер має список профілів (від найбільш специфічного до загального), що дозволяє розробнику мапити кнопки згідно з офіційним реєстром [WebXR Input Profiles](#).

#### Події введення: Select vs Squeeze

В WebXR існують дві основні абстракції для подій:

1. **Select**: Зазвичай мапиться на вказівний тригер. Використовується для активації (постріл, натискання кнопки UI).
2. **Squeeze**: Мапиться на бокові кнопки (grip). Це стандарт для механіки «взяття в руку».

#### Математика захоплення: .attach() vs .add()

У Three.js при спробі зробити об'єкт дитиною контролера:

- `group.add(object)`: Об'єкт змінить свою позицію у світі, оскільки його координати тепер будуть відносними до нового батька.
- `group.attach(object)`: Об'єкт збереже свою світову трансформацію. Three.js автоматично перерахує локальну матрицю об'єкта так, щоб він візуально залишився на тому ж місці в момент «прилипання» до руки.

#### Haptic Actuators

Тактильний відгук реалізується через `inputSource.gamepad.hapticActuators`. Основний метод: `pulse(value, duration)`, де:

- `value`: інтенсивність (від 0.0 до 1.0).
- `duration`: час у мілісекундах.

### 3. Практичне завдання: Проєкт «VR-Пісочниця»

#### Крок 1: Ініціалізація

Використовуйте `XRControllerModelFactory` для візуалізації контролерів. Це дозволить студенту бачити реальні моделі пристроїв.

#### Крок 2: Детекція колізій

Створіть функцію, яка перевіряє дистанцію між центром контролера та цільовим об'єктом. Якщо дистанція менша за радіус об'єкта — він стає «доступним для захоплення».

#### Крок 3: Реалізація Grabbing

Налаштуйте слухачі подій `squeezestart` та `squeezeend`. При старті — викликайте `controller.attach(object)`, при завершенні — повертайте об'єкт у `scene.attach(object)`.

#### Крок 4: Тактильний відгук

Додайте `pulse(0.3, 50)` при перетині контролера з об'єктом (`hover`) та `pulse(1.0, 100)` при успішному захопленні.

### 4. Завдання для самостійного виконання

1. **Механіка кидка:** В момент `squeezeend` обчисліть вектор швидкості: . Присвойте цей вектор об'єкту (якщо використовуєте фізичний рушій) або реалізуйте просту інерцію.
2. **Візуальний фідбек:** Змінюйте `emissive` колір матеріалу об'єкта на 0.2, коли рука знаходиться в зоні дотику.
3. **Творче завдання:** Створіть масив об'єктів з різною масою. «Важкі» об'єкти повинні викликати пульсуючу вібрацію кожні 500 мс, поки ви їх тримаєте.

### 5. Контрольні запитання

1. Чому при захопленні об'єкта краще використовувати `group.attach(object)` замість `group.add(object)`?
2. Які властивості об'єкта `GamepadHapticActuator` дозволяють керувати вібрацією?

3. Як визначити, чи підтримує підключений контролер відстеження ступенів свободи (6DoF)?
4. Що таке «Hand Tracking» і чим він відрізняється від роботи з контролерами в контексті WebXR?

**Корисні посилання:**

- [WebXR Emulator Extension](#) — незамінний інструмент для тестування squeeze-подій без шолома.