

졸업프로젝트 요구분석 및 중간구현

2023.05

Class	Team	TEAM MEMBERS
졸업프로젝트2 (종합설계)	1팀	송호영 201711836
박능수 교수님		정은호 201710177
		손하빈 201811652

CONTENTS



01

프로젝트 개요

02

요구사항 분석

03

시스템 아키텍처

04

구현 방법
및 진행 상황

05

문제점
및 향후 계획

01 프로젝트 개요



모션 인식 댄스 리듬 게임

목표	<ul style="list-style-type: none">• User가 카메라를 켜고 춤을 추면 정확도에 따라 점수 부여
구현 범위	<ul style="list-style-type: none">• 한 곡으로 이루어진 하나의 스테이지
개발 환경	<ul style="list-style-type: none">• Unreal Engine• C++• Python
협업 Tool	<ul style="list-style-type: none">• SVN• Notion
레퍼런스	<ul style="list-style-type: none">• 저스트 댄스 : 컨트롤러 기반 동작 인식 댄스 리듬 게임

01 프로젝트 개요



프로젝트 사용 기술

MediaPipe Pose

실시간 주요 관절 위치
검출 및 추출

소켓 통신

파이썬 기반 MediaPipe와
언리얼 엔진 간 데이터 교환

+

코사인 유사도 알고리즘

코사인 각도를 통해
두 벡터 간 유사도를 계산

IK 및 FABRIK

인체 벡터 위치 변화에 따른
캐릭터 포즈 변화 적용

02 요구사항 분석



기능적 요구사항

1. 모션 인식 기술

목적	User의 움직임을 정확하게 감지하고 분석하기 위해 카메라 및 MediaPipe 사용
목표	<ul style="list-style-type: none">• 정확성 : User의 움직임을 정확하게 인식 및 분석할 수 있는 성능 보장• 실시간 처리 : 모션 인식과 분석 결과가 게임 플레이에 영향을 주지 않도록 최적화• 다양한 환경 대응 : 조명, 각도, 배경, 의상 등 다양한 환경에서 안정적인 인식 보장• 개인정보 보호 : 카메라를 활용하는 만큼 모션 데이터를 적절히 보호

02 요구사항 분석



2. 실시간 점수 피드백

목적	User의 움직임에 따른 실시간 점수 계산 및 피드백 제공 통해 게임성 및 사용자 경험 향상
목표	<ul style="list-style-type: none">• 정확성 : 모션을 정확히 인식하고 그에 따른 공정한 계산 시스템 필요• 실시간 피드백 : 실시간 점수 및 피드백을 통해 즉각적 반응 및 유저 경험 개선

3. 사용자 친화적 UI

목적	쉽고 직관적인 화면 구성을 통해 게임 플레이를 용이하게 함
목표	<ul style="list-style-type: none">• 직관성 : 쉽게 이해 및 조작할 수 있는 화면 구성• 디자인 요소 : 전체적인 분위기와 매력적인 비주얼 형성• 반응성 : 게임 속도 및 진행을 저하시키지 않는 빠른 반응• 도움말 : 게임 및 인터페이스 이해에 도움을 주는 기능

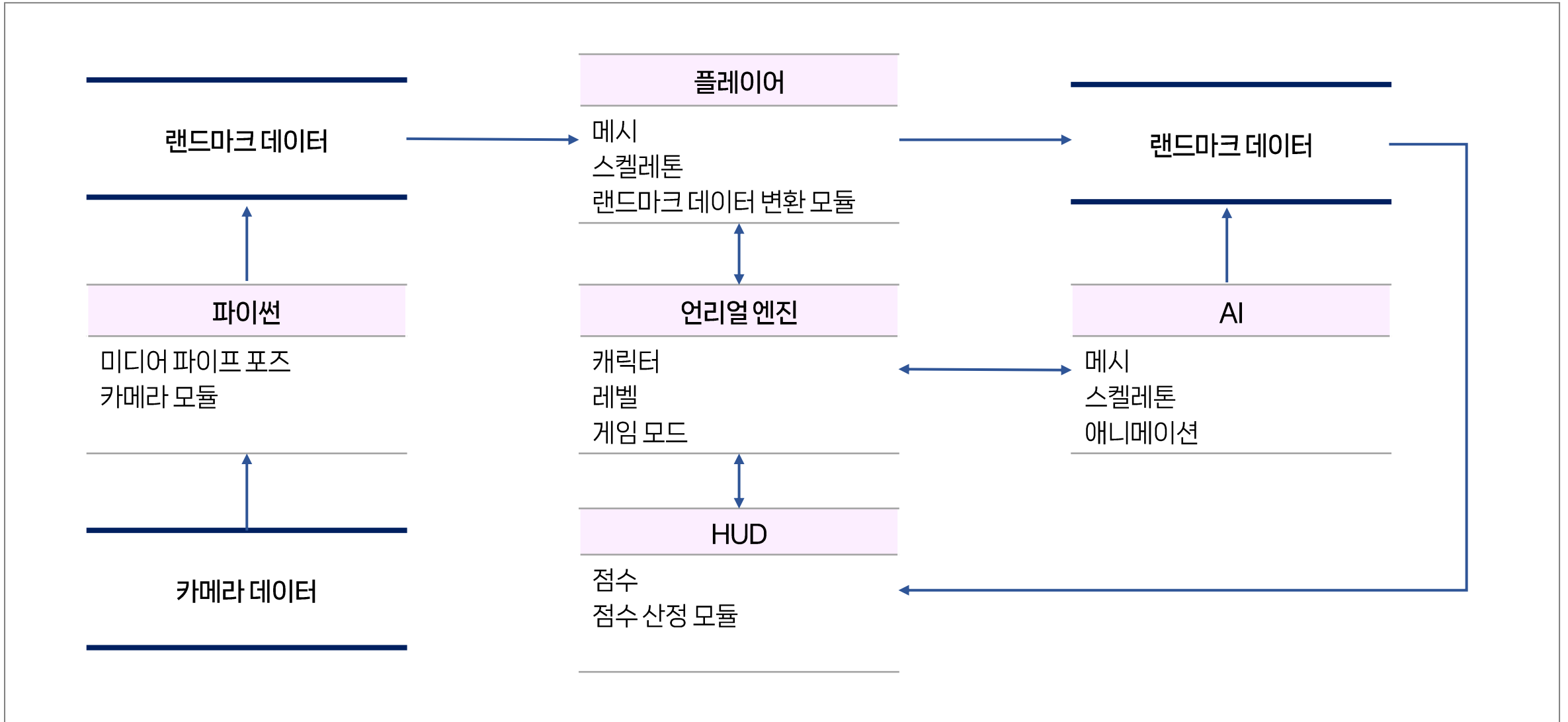
02 요구사항 분석



비기능적 요구사항

성능	<ul style="list-style-type: none">빠른 실시간 반응 및 애니메이션을 통해 원활한 진행 도모
안정성	<ul style="list-style-type: none">정상적으로 작동될 수 있도록 오류 최소화예기치 않은 상황도 대응할 수 있도록 설계
사용자 친화적 설계	<ul style="list-style-type: none">사용성 및 접근성 측면에서 사용자 요구에 맞도록 개선

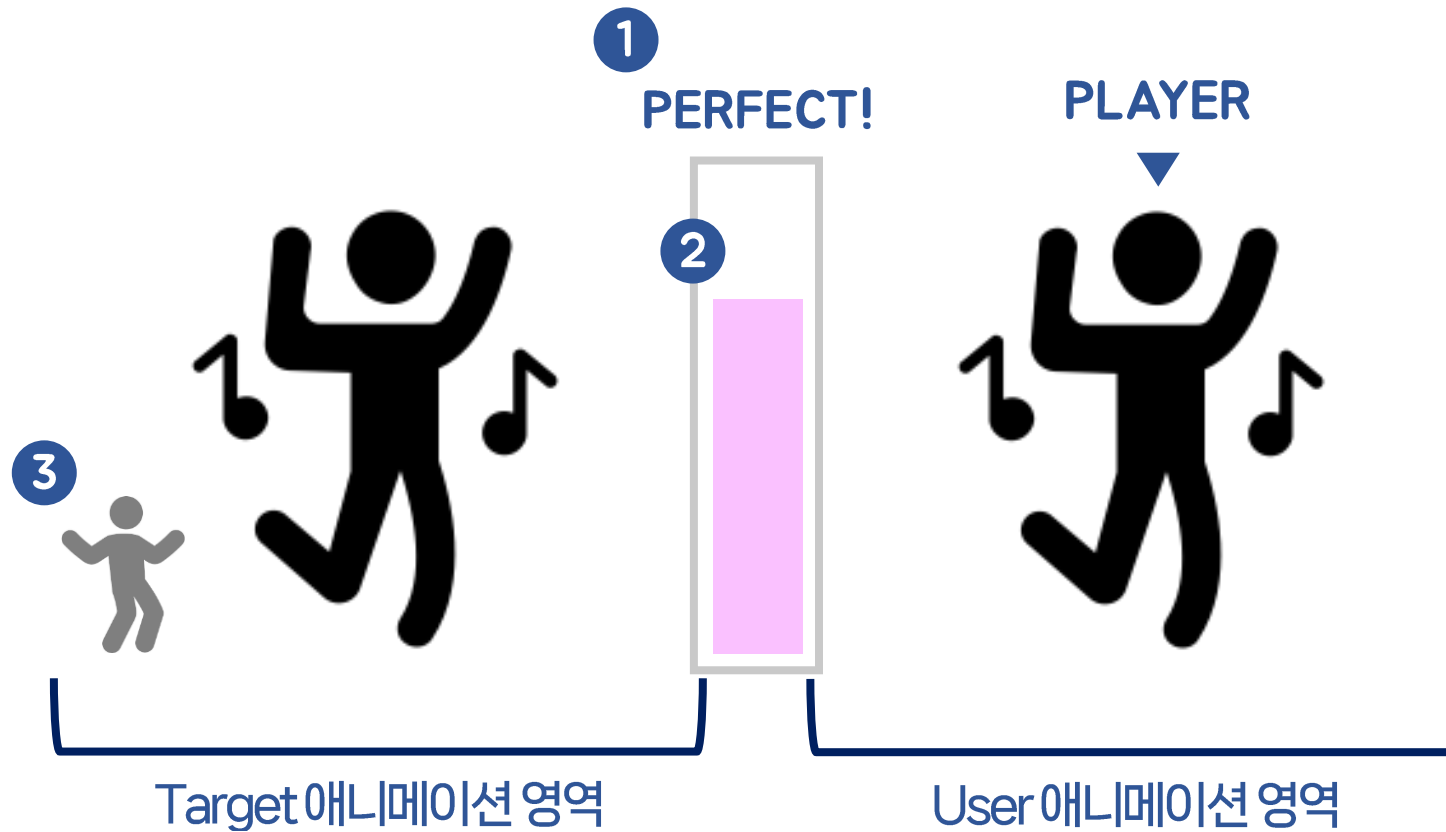
03 시스템 아키텍처



03 시스템 아키텍처



화면 구성 예시



- ① 실시간점수위젯
- ② 누적점수 bar 위젯
- ③ 다음동작미리보기위젯

04 구현 방법 및 진행 상황



ISSUE 01

데이터 교환

- 카메라 입력 및 미디어파이프를 통해 연산된 결과를
- 어떻게 언리얼 엔진에 전달할 것인가

ISSUE 02

좌표 값 변환

- MediaPipe의 33개의 관절 위치 벡터 값을
- 어떻게 언리얼 엔진에 적용 가능한 좌표 값으로 바꿀 것인가

04 구현 방법 및 진행 상황



ISSUE 03

애니메이션 적용

- 랜드마크 관절 위치 벡터 값을
- 어떻게 자연스러운 애니메이션을 구현할 것인가

04 구현 방법 및 진행 상황



ISSUE 01: 데이터 교환

언리얼 엔진
내장 파이썬
사용



언리얼에서 파이썬 사용이
아직 불완전하여 OUT

공용 메모리
사용



언리얼에서 공용 메모리
시스템 콜 함수 사용 불가하여 OUT

04 구현 방법 및 진행 상황



ISSUE 01: 데이터 교환



소켓 통신 사용

소켓 통신은 가용 데이터 통신량이 제한적이므로,
랜드마크 데이터만 전달하는 단방향 통신 모델 개발

04 구현 방법 및 진행 상황



ISSUE 02 :좌표 값 변환



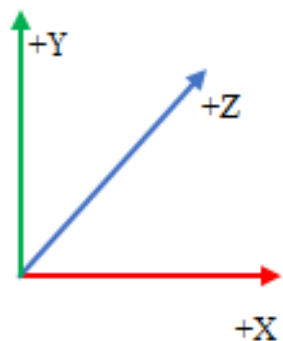
월드 좌표 값 + 랜드마크 벡터 값

신체 부위 별로 Location을 저장하는 변수를 두고,
실시간으로 월드 좌표 값에 랜드마크 벡터 값을 더하는 방식

04 구현 방법 및 진행 상황

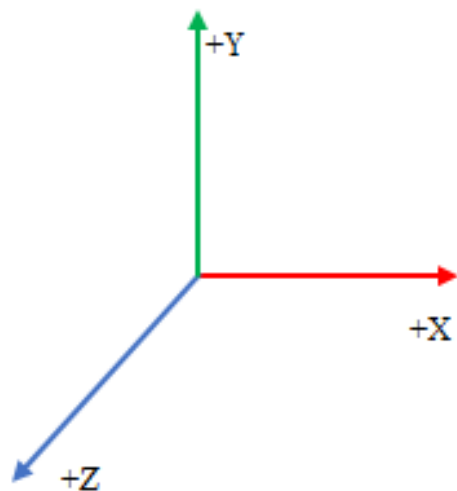


Left Handed



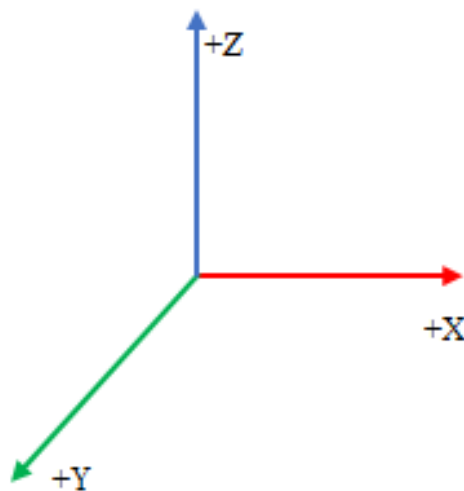
Unity, DirectX

Right Handed



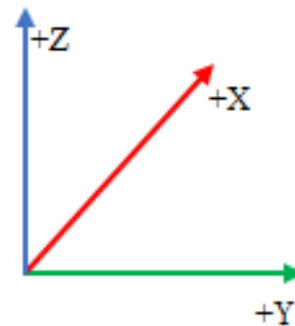
OpenGL

Right Handed



3DSMax

Left Handed



Unreal Engine

04 구현 방법 및 진행 상황



ISSUE 03 : 애니메이션 적용



IK 및 FABRIK 사용

캐릭터 본 구조에서 지정된 목표 위치에 맞게
뼈를 움직이고 회전시켜 원하는 모션을 구현하는 기술

언리얼 엔진 내 AnimGraph를 통해 사용

04 구현 방법 및 진행 상황



구현 완료 및 개선 목록

미디어파이프 파이썬 구동	완료
소켓 통신을 통한 언리얼 엔진과 데이터 교환	완료
언리얼 엔진 내 카메라 컴포넌트 추가	완료
언리얼 엔진 내 스켈레톤 캐릭터 메시 추가	완료
랜드마크 좌표계 변환	개선 중
캐릭터 애니메이션 적용	개선 중
코사인 유사도 알고리즘 구현	개선 중

04 구현 방법 및 진행 상황



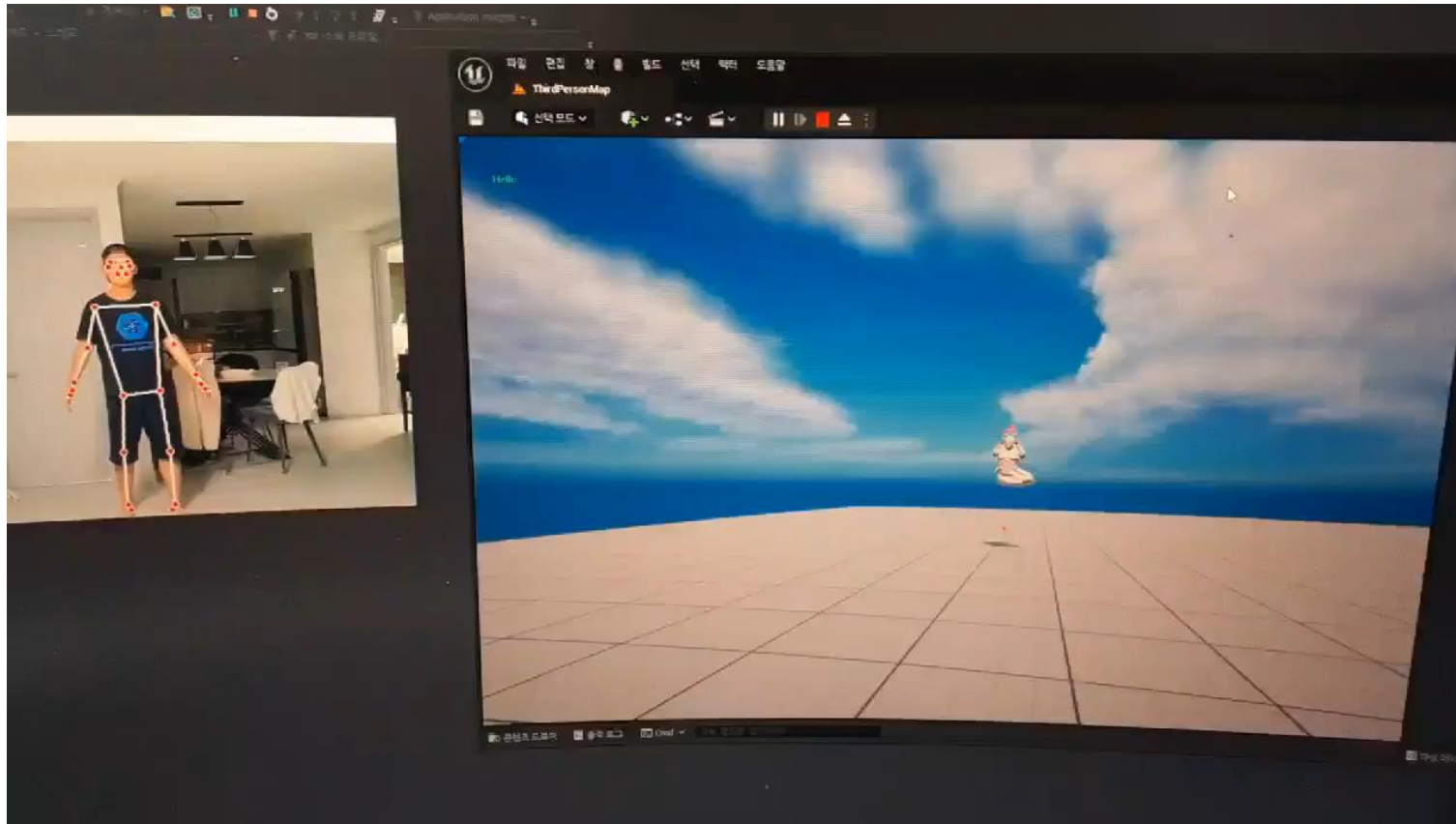
목표 대비 진행 상황

자료 조사, 작동 확인	개발 환경 구축	플러그인 제작	Camera 추가	움직임 벡터 추출	애니메이션 모델링	유사도 알고리즘	점수 시스템	UI 및 위젯
중간 구현 목표								
						최종 구현 목표		
<div>↓</div>								
★ 중간 구현 완료 ★								
						개선 중		
							최종 구현 목표	

04 구현 방법 및 진행 상황



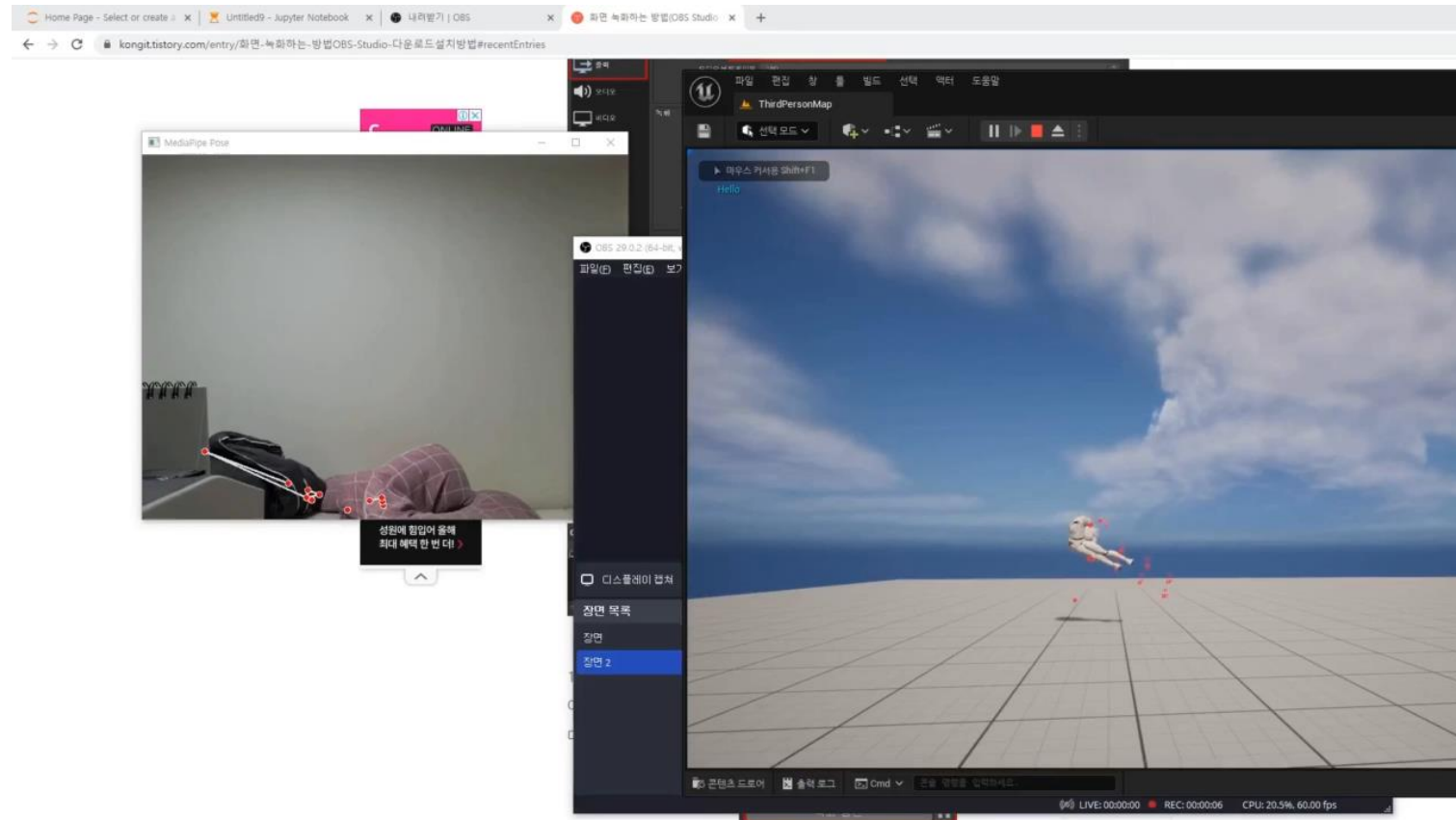
시연 영상 – 앉기



04 구현 방법 및 진행 상황



시연 영상 – 일어나기



05 문제점 및 향후 계획



Problem

카메라와의 거리 및 각도가 포즈 인식에 영향을 주는 문제

- 촬영 거리에 따른 피사체의 상대적인 크기를 반영하지 못함
- 촬영 각도에 따른 피사체의 상대적인 각도를 반영하지 못함
- 랜드마크를 좌표 값에 그대로 적용해서 생기는 문제

Solution

랜드마크 좌표 값 가공

- 랜드마크 좌표 간 각도 추출
- 캐릭터 본 거리에 맞는 로컬 좌표로 재조정
- FABRIK을 통해 루트 본을 기준으로 알맞은 표현 재현

05 문제점 및 향후 계획



ISSUE 01

점수 시스템 구현

- 언리얼 엔진에서 사용 가능하도록 코드 개선
- 유사도에 따른 점수 조건 및 공식 구현
- 테스트를 통한 점수 분배 및 밸런스 조정

ISSUE 02

위젯 및 UI 구현

- 실시간 점수 위젯
- 누적 점수 위젯
- 다음 동작 미리보기 위젯

THANK YOU
