소프트웨어 개발자 포트폴리오

박진원 Jinwon Park

문서화와 소통을 중요하게 생각합니다.

C++·C를 주로 사용하며, 팀 프로젝트에서 C# 저장·조회 API를 개발했습니다. 개인 앱은 AI와 함께 만들고 동작을 확인합니다.

nowjinpark@gmail.com · 010-9617-3114
https://jinwon-park-portfolio.pages.dev/sw | GitHub

기술

주력 언어 C++ · C
프로젝트 경험 C# · Unreal Engine · Blueprint · Unity
서버 연동 경험 ASP.NET Core · HTTP / JSON · Azure Cosmos DB
기초 학습 SQL·PostgreSQL 초급, 수업 실습 **도구** Git · Perforce

프로젝트 안내

02 LectureScribe
맥 전용 STT 앱 · 개인 프로젝트 · AI 활용 개발

03 Triad of Valor
C++ 저장 요청 · C# API · Azure Cosmos DB

04 Personal Knowledge OS
자료 검색 · 가져오기 검사 · AI 개발 프로젝트

05 시스템 과제와 팀 협업
C++ 진단 도구 · 통신 · SQL 실습 · 팀 협업

학력·교육

DigiPen Institute of Technology · 2024.09 - 2026.04
B.S. in Computer Science in Real-Time Interactive Simulation · 학사 졸업

계명대학교 · 2017.03 - 2026.08 · 디지펜게임공학과 학사 졸업

SKALA 4기 · SK AX · AI 서비스 개발 교육 수강 중
2026.07 - 현재 · 2026.12.11 수료 예정

LectureScribe

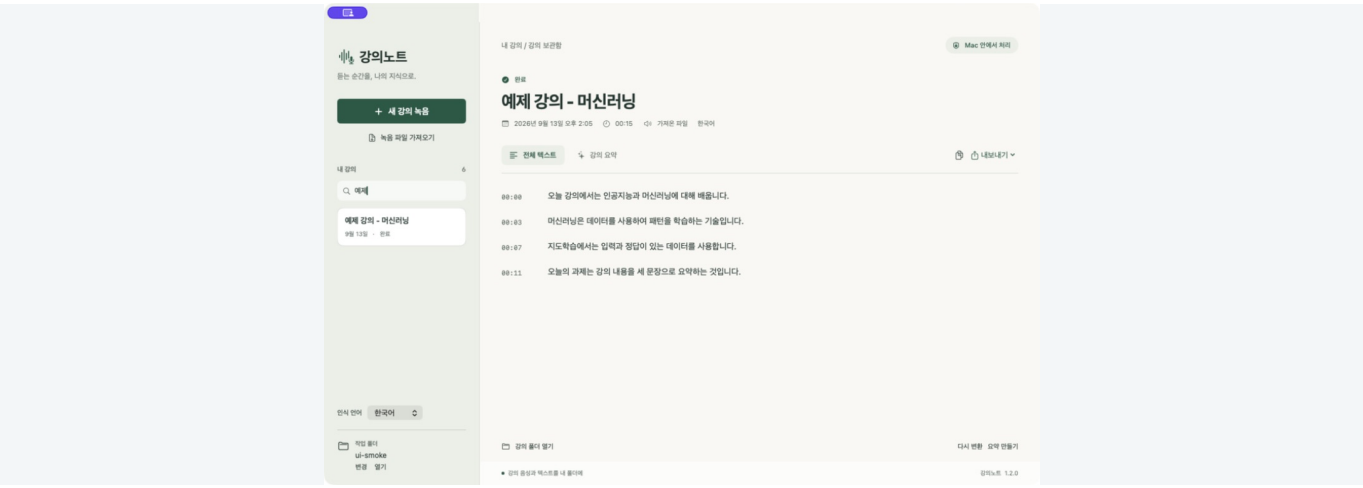
2026.09

개인 프로젝트 | 사용 기술: Swift / SwiftUI / WhisperKit / Foundation Models

담당 기획 · AI 활용 개발

프로젝트 상세 | GitHub · 공개 소스와 개발 문서

수업 중 놓친 내용을 다시 확인하려고 만든 맥 전용 STT 앱입니다. 강의를 녹음해 텍스트로 바꾸고, 필요한 내용은 요약해서 볼 수 있습니다.



녹음한 내용을 시간대별로 확인하고 텍스트 파일로 저장할 수 있습니다.

Whisper로 음성을 텍스트로 변환

기존 음성 인식에서 문장이 빠지는 문제를 줄이기 위해 WhisperKit으로 변경했습니다. 긴 무음 구간은 분석 대상에서 제외하되 원본 녹음과 문장의 시간 정보는 유지하도록 했습니다.

다시 변환해도 이전 기록 유지

녹음 파일과 텍스트를 강의별 폴더에 저장합니다. 같은 녹음을 다시 변환할 때는 새 결과가 완성된 뒤 파일을 바꾸도록 해, 작업이 취소되거나 실패해도 이전 기록이 남도록 했습니다.

필요한 강의만 요약

녹음이 끝날 때마다 자동으로 요약하던 기능을 버튼을 눌러 실행하는 방식으로 바꿨습니다. Apple Intelligence를 사용할 수 없으면 원문에서 핵심 문장을 골라 보여 줍니다. 창을 닫아도 메뉴 막대에서 녹음을 제어할 수 있습니다.

담당 범위 필요한 기능을 직접 정하고 AI와 함께 개발·테스트했습니다.

Triad of Valor

2025.09-2026.04 · DigiPen GAM400/450 팀 프로젝트

Kimbap Games · 6인 팀 | C++ / C# / ASP.NET Core / HTTP·JSON / Azure Cosmos DB

담당 게임플레이 프로그래머 · 전투·계정 연동·클라우드 저장

프로젝트 상세 | GitHub · 소스 발췌

여럿이 함께 보스에 도전하는 액션 RPG입니다. 보스 공격·패링·체력 처리와 로그인을 맡고, 팀의 캐릭터 코드에 클라우드 저장을 연동했습니다.



눈 덮인 항구의 플레이 화면입니다. 체력·스태미나·인벤토리 HUD가 보입니다. 출처: 팀원의 공개 튜토리얼 영상.

Unreal과 C# 저장 API 연동

CloudSaveComponent가 ASP.NET Core API를 호출해 Azure Cosmos DB에 저장하도록 구현했습니다. 중첩 객체와 배열을 유지하고 누락된 필드에만 기본값을 넣었습니다. 조회 응답의 JSON 형식을 맞춰 게임에서 인벤토리를 다시 읽을 수 있게 했습니다.

로그인 화면과 비동기 요청

Slate 로그인 화면과 AuthSubsystem의 로그인·회원가입 요청을 구현했습니다. 응답 시간 초과, 네트워크 실패, HTTP 오류를 처리하고 콜백은 Game Thread에서 실행했습니다. 요청 전 서버 상태를 확인하는 기능도 추가했습니다.

저장 데이터 집계와 CSV 다운로드

캐릭터의 현재 저장 데이터를 집계하는 C# API를 만들었습니다. 돈·사망 횟수·플레이 시간을 요약하고, 지역별 분포를 조회하거나 집계 결과를 CSV로 내려받을 수 있게 했습니다.

담당 범위 보스 공격·체력·패링, 계정 연동과 클라우드 저장을 담당했습니다. 캐릭터와 레벨 이동은 팀 공동 코드이며 제가 만든 저장 컴포넌트와 API를 사용합니다. 저장 API는 수업 예제에서 시작해 교수와 AI의 도움을 받아 개발했습니다.

Personal Knowledge OS

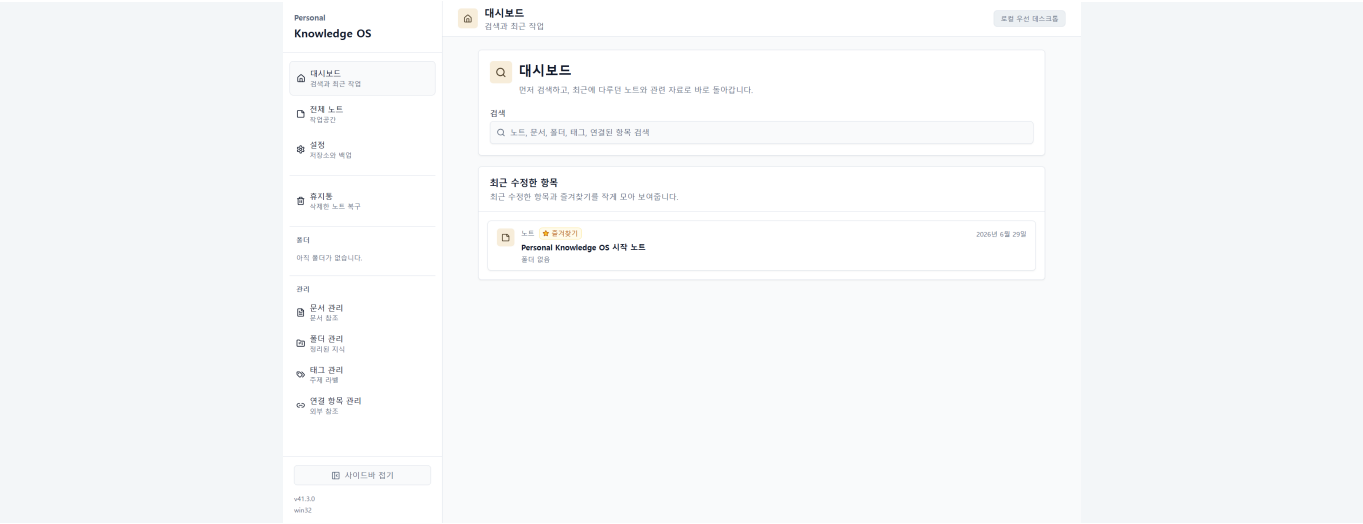
2026.04-07

개인 프로젝트 | 사용 기술: Electron / React / TypeScript / SQLite · AI 개발

담당 개인 기획·기능 설계 · AI 보조 개발·검증

프로젝트 상세 | GitHub · 공개 소스

흩어진 노트·문서·링크를 모아 검색하는 Windows 앱입니다. 필요한 기능을 기획하고 AI와 함께 개발했으며, 폴더·태그 분류와 관련 자료 연결을 지원합니다.



노트를 검색하고 최근 수정한 항목을 확인하는 대시보드입니다.

노트·문서·링크 통합 검색

세 종류의 자료를 같은 검색 결과에 표시합니다. NFKC 정규화와 대소문자 정리로 문자열 표기를 맞추고, 휴지통 항목은 제외한 뒤 수정일순으로 정렬합니다.

폴더·태그와 관련 자료 연결

본문과 분류 정보를 나눠 저장하고 폴더·태그·즐겨찾기, 자료 간 연결 관계를 관리합니다. 분류 메타데이터가 맞지 않으면 다시 정리하는 기능도 만들었습니다.

JSON 검사와 초기화 전 백업

가져올 JSON의 스키마 버전·ID·문서 소유권·첨부 파일 상대 경로를 검사합니다. 초기화 전에는 자료와 첨부 메타데이터를 JSON으로 저장하고, 삭제 대상이 앱의 첨부 폴더인지 확인합니다.

담당 범위 AI와 함께 개발한 Windows 앱이며 데이터는 PC에 저장합니다. JSON 백업에는 자료와 첨부 메타데이터가 들어가지만, 첨부 파일 자체가 모두 포함되는 것은 아닙니다.

시스템 과제와 팀 협업

C++ 개인 과제 · SQL 기초 실습 · 게임 팀 프로젝트

C++ Debugging Tools

크래시 덤프, 메모리 할당·해제 기록, 실행 위치 샘플을 남기는 C++ 도구를 만들었습니다. 오류 원인과 해제되지 않은 메모리를 확인하는 수업 과제입니다.

전역 new/delete에 할당 기록을 연결했습니다. 이미 해제된 주소를 다시 해제하거나 new와 delete의 종류가 맞지 않으면 CSV에 기록합니다. 종료 시에는 해제되지 않은 할당 내역도 남깁니다.

프로젝트 상세 | GitHub · 소스 발췌

WinSock Networking

WinSock으로 TCP·UDP 송수신과 논블로킹 I/O를 구현한 과제입니다. 수업에서 제공한 Lockstep 게임에 UDP 통신과 턴 번호·이동 좌표 직렬화를 추가했습니다.

제공된 게임 상태 코드에 논블로킹 UDP 통신을 추가했습니다. 턴 번호와 이동 좌표를 직렬화해 보내고, 받은 메시지는 역직렬화해 게임 진행 코드에 전달했습니다.

수업에서 제공한 Lockstep 게임에 통신 기능을 추가한 과제입니다. 게임 상태 구성은 제공 코드이며, 로컬 주소로 연결했습니다.

프로젝트 상세 | GitHub · 소스 발췌

PostgreSQL Lab · SQL 초급

SQL 기초를 익히며 인덱스와 페이지네이션에 따른 실행계획을 비교했습니다. PostgreSQL 수업 실습에서 격리 수준의 차이를 확인하고 데드락을 재현했습니다.

READ COMMITTED와 REPEATABLE READ를 비교했습니다. 두 세션의 UPDATE 순서를 바꿔 40P01 데드락을 재현하고, 같은 순서로 잠금을 획득하면 피할 수 있는 이유를 정리했습니다.

프로젝트 상세 | GitHub · SQL 실습

An Omen in the Mirror

거울을 통해 두 시대의 저택을 오가는 퍼즐 게임입니다. 피아노 퍼즐과 문·창문 상호작용을 맡고, 플레이테스트 의견에 따라 정답 판정과 조작 방식을 수정했습니다.

마일스톤마다 개인 계획과 회고에 목표·작업·위험 요소·결과를 기록했습니다. ClickUp으로 서로의 진행 상황을 확인하고, 구현 방식이 바뀌면 프로그래머 팀 Discord에 공유했습니다. 정기 회의에서는 다음 작업을 논의하고 오브젝트 사용법을 설명했습니다.

프로젝트 상세 | GitHub · Blueprint 소스