
Mcp Manager :M&M

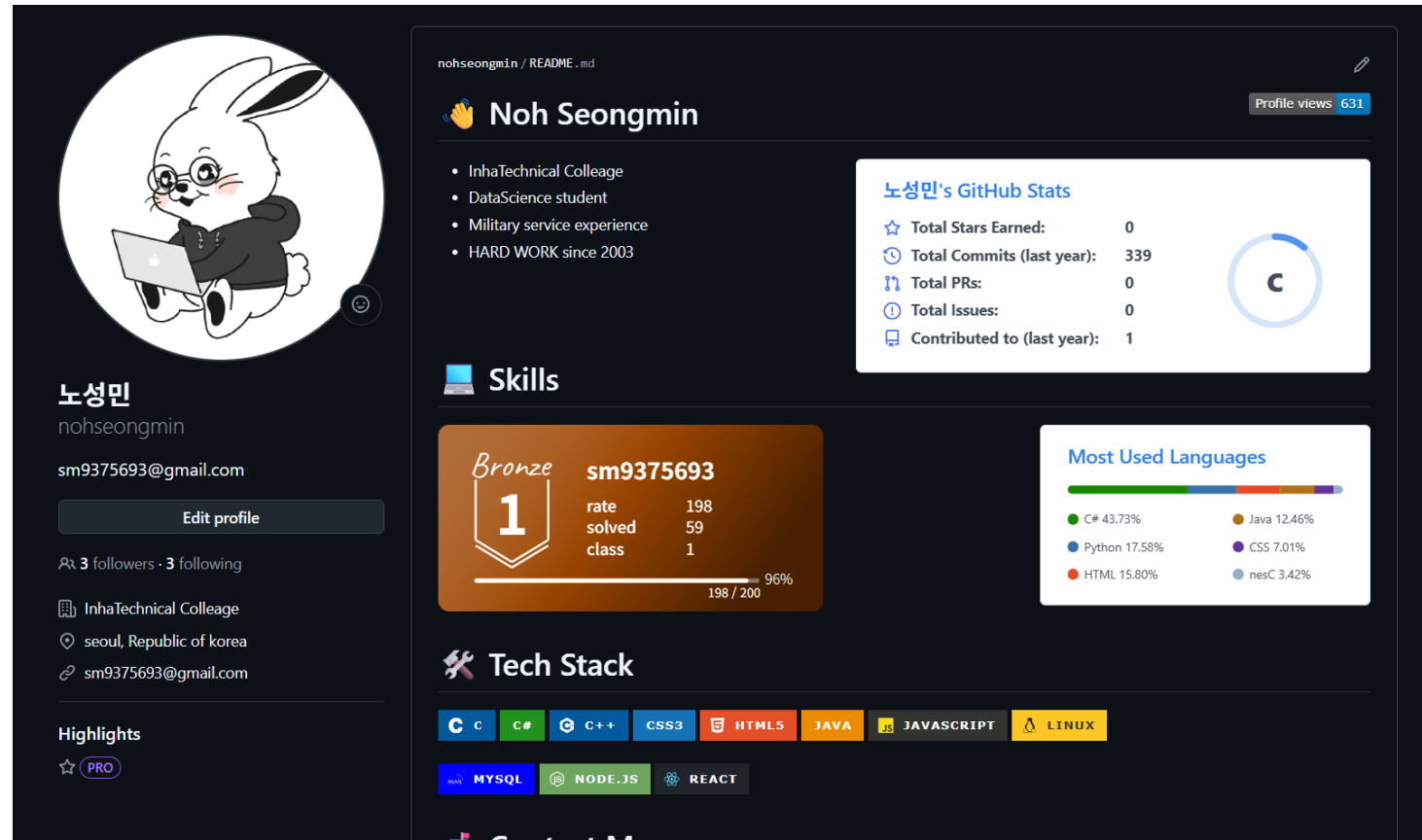
1주차 기획

s/w 프로젝트 노성민



기획의도

- 꾸미기 귀찮음
- 지루하고 현학적임



The image shows a GitHub profile for Noh Seongmin (nohseongmin). The profile includes a circular avatar of a white rabbit with glasses and a laptop. The bio lists: InhaTechnical College, DataScience student, Military service experience, and HARD WORK since 2003. The profile has 3 followers and is following 3 people. The location is Seoul, Republic of Korea. The email is sm9375693@gmail.com. The profile is highlighted with a 'PRO' badge. The 'Stats' section shows: Total Stars Earned: 0, Total Commits (last year): 339, Total PRs: 0, Total Issues: 0, and Contributed to (last year): 1. The 'Skills' section shows a Bronze 1st rank for sm9375693 with a rate of 198, solved 59, and class 1. The 'Most Used Languages' section shows: C# 43.73%, Python 17.58%, HTML 15.80%, Java 12.46%, CSS 7.01%, and nesC 3.42%. The 'Tech Stack' section shows: C, C#, C++, CSS3, HTML5, JAVA, JAVASCRIPT, LINUX, MYSQL, NODE.JS, and REACT.

nohseongmin / README.md

Noh Seongmin Profile views 631

- InhaTechnical College
- DataScience student
- Military service experience
- HARD WORK since 2003

노성민's GitHub Stats

- ☆ Total Stars Earned: 0
- 🕒 Total Commits (last year): 339
- 🔗 Total PRs: 0
- 🔔 Total Issues: 0
- 📦 Contributed to (last year): 1

Skills

Bronze 1 **sm9375693**

rate	198
solved	59
class	1

198 / 200 96%

Most Used Languages

- C# 43.73%
- Python 17.58%
- HTML 15.80%
- Java 12.46%
- CSS 7.01%
- nesC 3.42%

Tech Stack

C C# C++ CSS3 HTML5 JAVA JAVASCRIPT LINUX MYSQL NODE.JS REACT

기획의도

- 딸깍
- 시간 낭비 없음



```
> GEMINI

Tips for getting started:
1. Ask questions, edit files, or run commands.
2. Be specific for the best results.
3. /help for more information.

Gemini CLI update available! 0.4.1 → 0.5.5
Installed with npm. Attempting to automatically update now...

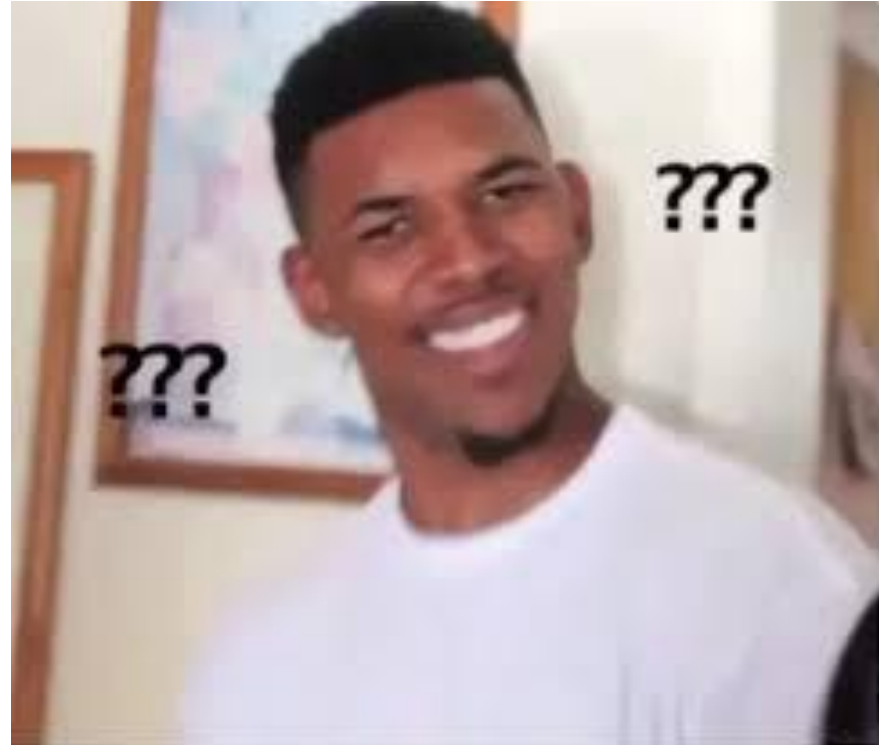
Using: 1 GEMINI.md file | 1 MCP server (ctrl+t to view)

> github.com/????? 해당 링크 참고해서 내 깃허브 프로필 꾸며줘 █

~\OneDrive\Desktop\workspace    no sandbox (see /docs)    gemini-2.5-pro (100% context left)
```

기획의도

- 어떻게??



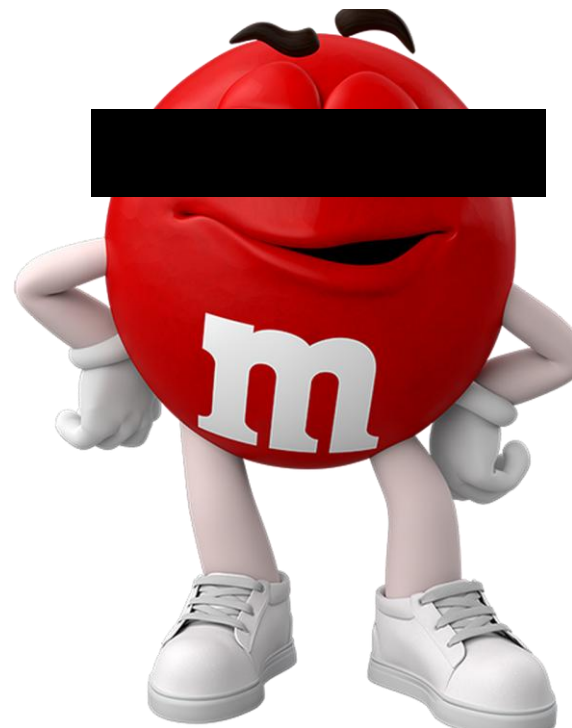
기획의도

- 어떻게??
- MCP 서버를 통해
- 근데 좀 귀찮음



M&M

- MCP 연결 도우미
- Gemini Cli 설치 도우미
- Gemini Cli 실행 보조
- 순혈 C#
- DB 안씀



WBS

날짜	작업 항목	세부 내용
9월 24일	프로젝트 초기 계획 및 분석	1.1. 요구사항 정의: API 키 입력/상태 표시 UI, 성공/실패 알림 기능 정의. 1.2. 기술 스택 및 환경 설정: C# 데스크톱 앱 개발 환경, HttpClient 라이브러리 준비. 1.3. 초기 파일 생성: 프로젝트 구조, README 작성.
9월 25일	핵심 로직 개발: 스크립트 분석	2.1.1. @smithery/cli 소스 코드 분석: 패키지 다운로드 URL, 내부 명령 파악. 2.1.2. @smithery-ai/github 소스 코드 분석: 파일 생성, gemini-cli 설정 방식 파악.
9월 26일	핵심 로직 개발: 파일 다운로드	2.2.1. HttpClient를 사용한 패키지 다운로드 기능 구현. 2.2.2. 다운로드 진행 상황 및 오류 처리 로직 구현.
9월 27일	핵심 로직 개발: 압축 해제 및 파일 조작	2.3.1. System.IO.Compression을 이용한 압축 해제 기능 구현. 2.3.2. 필요한 파일 복사 및 폴더 생성 로직 구현.
9월 28일	핵심 로직 개발: 설정 파일 조작	2.4.1. gemini-cli 설정 파일 경로 탐색 기능 구현. 2.4.2. API 키 및 프로필 정보를 설정 파일에 기록하는 기능 구현.
9월 29일	핵심 로직 개발: 마무리	전체 핵심 로직 코드 검토 및 리팩토링. 이슈 및 버그 초기 수정.
9월 30일	사용자 인터페이스(UI) 개발	3.1. GUI 디자인: API 키 입력창, '설치 시작' 버튼, 상태 표시 UI 디자인 및 구현.
10월 1일	사용자 인터페이스(UI) 개발	3.2. 이벤트 핸들링 구현: '설치 시작' 버튼 클릭 시 핵심 로직 실행, 진행 상태 업데이트 로직 구현.
10월 2일	사용자 인터페이스(UI) 개발	3.3. 유효성 검사: API 키 입력 유효성 검사 기능 추가. UI와 로직 연결 작업.
10월 3일	통합 테스트 및 디버깅	4.1. 기능별 테스트: 각 기능(다운로드, 파일 조작 등)이 개별적으로 정상 작동하는지 확인.
10월 4일	통합 테스트 및 디버깅	4.2. 통합 테스트: 프로그램의 처음부터 끝까지 전체 프로세스가 원활하게 작동하는지 확인.
10월 5일	통합 테스트 및 디버깅	4.3. 버그 수정 및 최적화: 발견된 버그를 수정하고 코드 최적화.
10월 6일	예비일	예상치 못한 문제 발생 시 추가 작업일.
10월 7일	예비일	추가적인 테스트 및 마무리 작업.

WBS

날짜	작업 항목	세부 내용		
9월 24일	프로젝트 초기 계획 및 분석	1.1. 요구사항 정의: API 키 입력/상태 표시 UI, 성공/실패 알림 기능 정의. 1.2. 기술 스택 및 환경 설정: C# 데스크톱 앱 개발 환경, HttpClient 라이브러리 준비. 1.3. 초기 파일 구조 설계.		
9월 25일	핵심 로직 개발: 스크립트 분석	2.1.1. @smithery/cli 소스 코드 분석: 패키지 다운로드 URL, 내부 명령 파싱. 2.1.2. @smithery-ai/cli 소스 코드 분석: 파일 생성, gemini-cli 설정 방식 파악.		
9월 26일	핵심 로직 개발: 파일 다운로드	2.2.1. HttpClient를 사용한 패키지 다운로드 로직 구현. 2.2.2. 로컬 캐시 및 다운로드 진행 상황 및 오류 처리 로직 구현.		
9월 27일	핵심 로직 개발: 압축 해제 및 파일 조직	2.3.1. 7z 및 ZIP compression을 이용한 압축 해제 로직 구현. 2.3.2. 필요한 파일 검사 및 폴더 생성 로직 구현.		
9월 28일	핵심 로직 개발: 설정 파일 조작	3.1.1. gemini-cli 설정 파일 생성 및 읽기 로직 구현. 3.1.2. API 키 및 모델 정보를 설정 파일에 기록하는 기능 구현.		
9월 29일	핵심 로직 개발: 마무리	3.2.1. CLI 명령어 구조화 및 도움말 생성. 3.2.2. 전체 핵심 로직 코드 검토 및 리팩토링. 3.2.3. 이슈 및 버그 초기 수정.		
9월 30일	사용자 인터페이스(UI) 개발	4.1.1. 'gemini-cli' API 키 입력창 설치 시작 버튼, 상태 표시 UI 디자인을 구현.		
10월 1일	사용자 인터페이스(UI) 개발	4.1.2. 이벤트 핸들링 구현: 설치 시작 버튼 클릭 시 핵심 로직 실행, 진행 상태 업데이트 로직 구현.		
10월 2일	사용자 인터페이스(UI) 개발	4.2.1. UI 디자인 완성: API 키 입력, 다운로드 진행률 표시, 로그 출력 영역 등. 4.2.2. 유효성 검사: API 키 형식, 다운로드 경로 검사 기능 추가 및 로직 연결 작업.		
10월 3일	통합 테스트 및 디버깅	5.1.1. 개발 환경: 각 기능 다운로드, 파일 생성 등 개별적으로 테스트 작동하는지 확인.		
10월 4일	통합 테스트 및 디버깅	5.1.2. 통합 테스트: 프로그램의 처음부터 끝까지 전체 프로세스가 원활하게 작동하는지 확인.		
10월 5일	통합 테스트 및 디버깅	5.2.1. 버그 수정 및 최적화: 발견된 버그를 수정하고 코드 최적화.		
10월 6일	예비일	예상치 못한 문제 발생 시 추가 작업일.		
10월 7일	예비일	추가적인 테스트 및 마무리 작업.		

기능정의서 - 메뉴구조도

	구분		기능		세부기능
1	MCP 관리 기능	1-1	MCP 목록 출력	1-1-1	mcp 목록 출력 기능
				1-1-2	mcp 목록 설명 다국어 지원(eng, kor)
		1-2	MCP 추가 기능	1-2-1	mcp 설치를 버튼 하나로 할 수 있도록 하는 기능
				1-2-2	API 키 발급 지원 기능 (하이퍼링크 등으로 간편화)
		1-3	MCP 삭제 기능	1-3-1	setting.json 수정을 통한 mcp 서버 삭제 기능
				1-3-2	MCP 삭제시 저장된 API키도 삭제
2	Gemini Cli 설치 지원	2-1	Gemini Cli 설치	2-1-1	개인 PC의 CMD 환경에서 자동 설치
				2-1-2	Node.js 자동 설치
				2-1-3	설치 과정 중 발생하는 오류 처리
		2-2	Gemini Cli 지원	2-2-1	Gemini Cli 사용 방법 매뉴얼
				2-2-2	Gemini Cli 자동 업데이트 기능

프로토타입

MCP 관리 도구

Model Context Protocol 및 Gemini CLI 관리

언어 ☒ EN

MCP 관리

Gemini CLI

MCP 목록

설치된 MCP 서버를 관리하고 새로운 MCP를 추가할 수 있습니다.

+ MCP 추가

이름	설명	상태	버전	API 키	작업
GitHub MCP	GitHub repository integration	설치됨	v1.2.0	****-****-****-ghp123	삭제
Slack MCP	Slack workspace integration	설치됨	v2.1.0	****-****-****-xoxb456	삭제
Google Drive MCP	Google Drive file management	사용 가능	v1.0.5	-	설치

QnA

- qna

