

고급소프트웨어공학

CTIP 환경 구축

Team #3

컴퓨터공학부 202473136 한범진

컴퓨터공학부 **202371449 김상재**

컴퓨터공학부 202273104 서푸름

컴퓨터공학부 202473137 함시연

목 차

- IDE
- Project Builder
- Unit Test
- Code Version Management
- CI Tool
- Requirements Management & Bug Tracking
- Team Communication

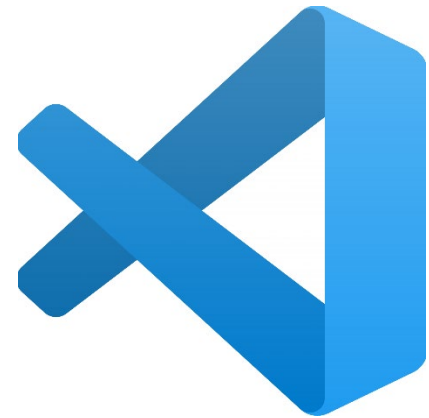
IDE



Visual
Studio



CLion

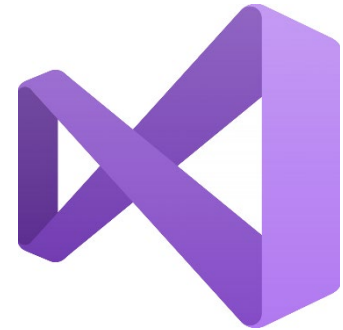


Visual
Studio Code

IDE

- Visual Studio

- Microsoft 사의 C/C++ 개발 도구



- 장점

- IntelliSense, 디버깅, 코드 분석 등 다양한 기능 제공
 - 팀 협업을 위한 기능을 제공하며, 원격 디버깅 가능

- 단점

- 무거운 IDE로, 작업 시 많은 메모리와 디스크 공간 필요
 - Windows 환경만 지원함
 - 다른 OS에서 사용 불가능한 라이브러리 포함 등

IDE

- CLion

- JetBrains 사의 C/C++ 개발 도구



- 장점

- 멀티스레드 디버깅을 지원, CMake와의 통합 가능
 - 스마트 코드 완성과 리팩토링 도구 지원

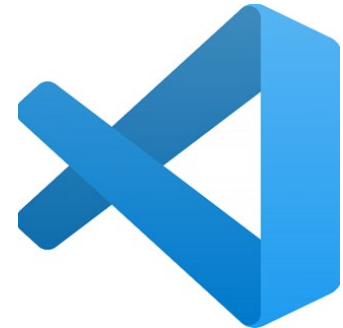
- 단점

- 유료 라이선스가 필요하며, 무료 대안에 비해 비용 부담이 큼
 - Visual Studio 대비, 사용자 경험 부족
 - 개발팀 및 검증팀 모두

IDE

- Visual Studio Code

- Microsoft 사의 코드 편집 도구



- 장점

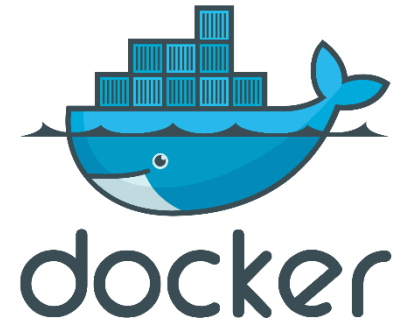
- 가벼운 코드 편집 도구

- 다양한 확장 기능을 통해 여러 기능들을 손쉽게 추가 가능

- 단점

- 편집 도구이므로, 프로그램 제작에 필요한 컴파일 환경을 따로 구축해야함

IDE



- Docker

- 응용에 필요한 모든 요소들을 컨테이너로 패키징
 - 코드, 라이브러리, 설정 파일 등
- 개발, 테스트, 배포를 간소화하는 소프트웨어 플랫폼

- 장점

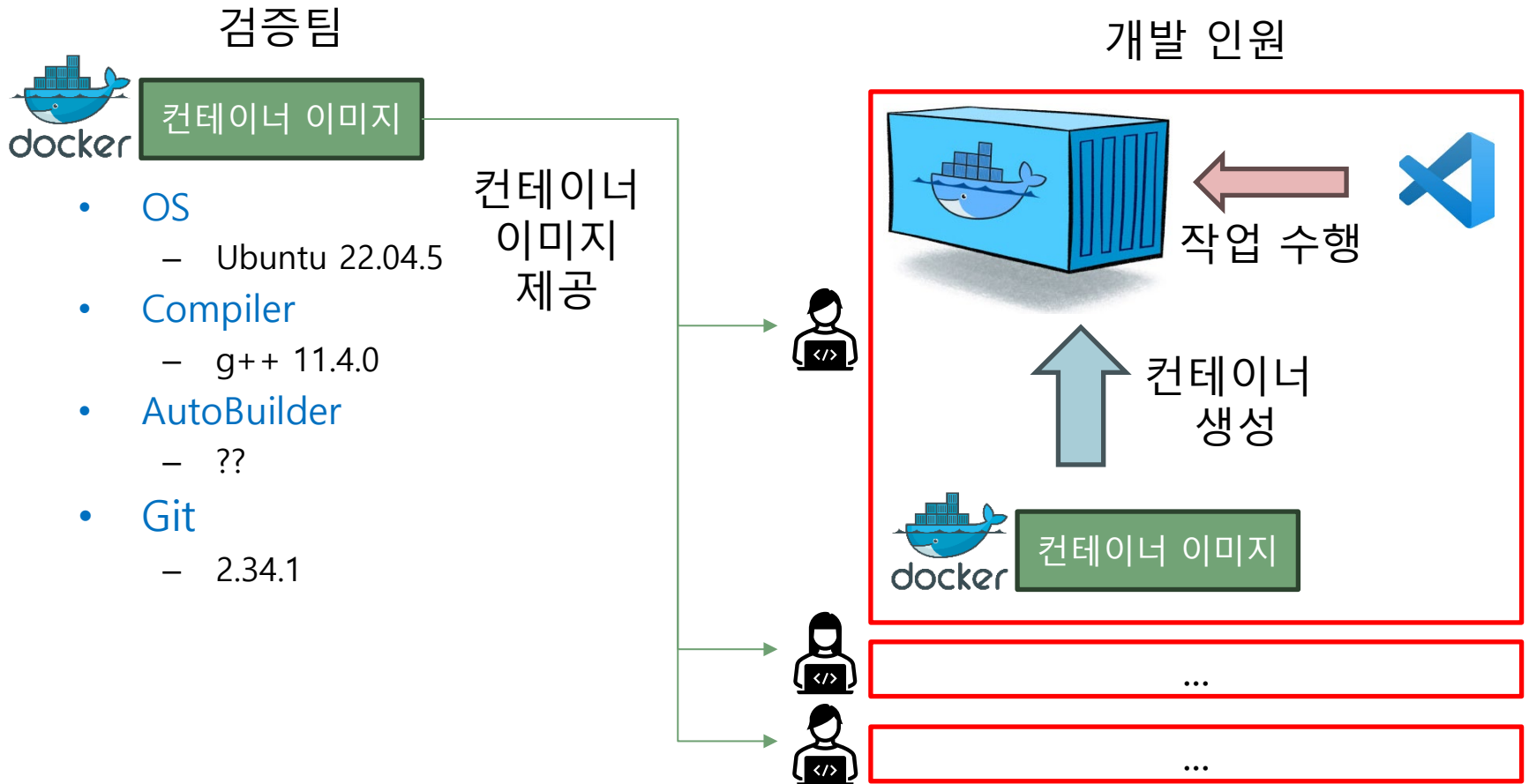
- 개발 환경 및 프로덕션 환경 간의 차이를 제거
 - “내 컴퓨터에서는 잘 되는데” 문제 방지
- 다양한 운영체제 및 CPU 아키텍처 지원 가능
 - 컴파일러 버전 및 Auto Builder 버전만 맞으면 Intel CPU, ARM CPU 등 다양한 CPU 버전도 지원 가능

- 단점

- 컨테이너 이미지를 자체적으로 설계 및 구현해야 함
- 컨테이너 사용을 위한 환경이 초기에 구축되어야 함

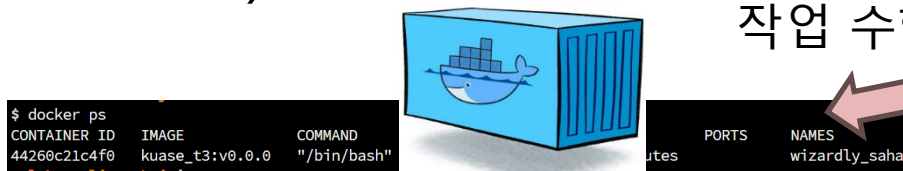
IDE

• Visual Studio Code + Docker



IDE

- Visual Studio Code + Docker
 - 예시)



컨테이너 이미지 기반
실행 중인 컨테이너

↑
컨테이너
생성

```
FROM ubuntu:jammy

RUN sed -i -e 's/archive.ubuntu.com/ftp.kaist.ac.kr/g' \
    -e 's/security.ubuntu.com/ftp.kaist.ac.kr/g' /etc/apt/sources.list

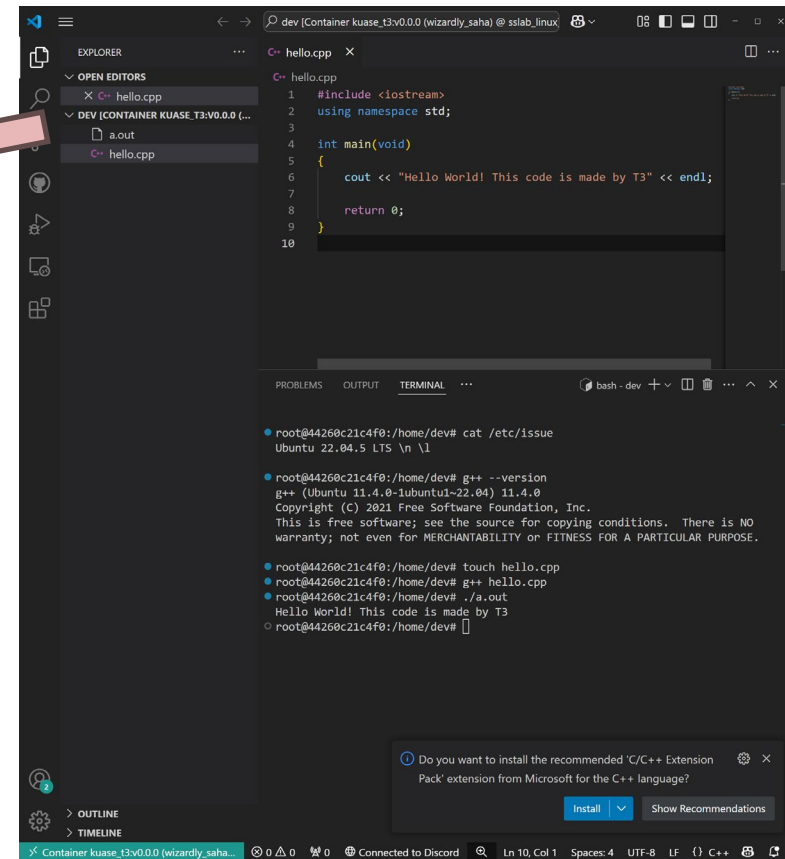
RUN apt-get update

RUN apt-get install vim -y

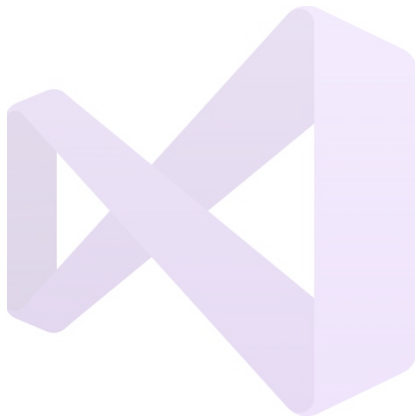
RUN apt-get install g++-11 -y && \
    update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-11 99
```



컨테이너 이미지 생성을 위한 파일



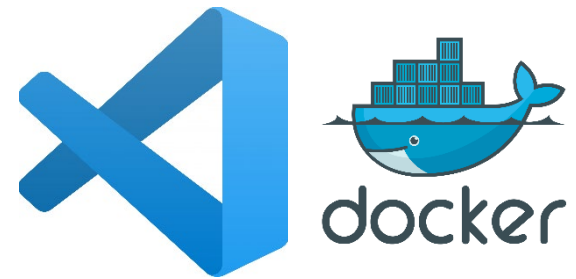
IDE



Visual
Studio

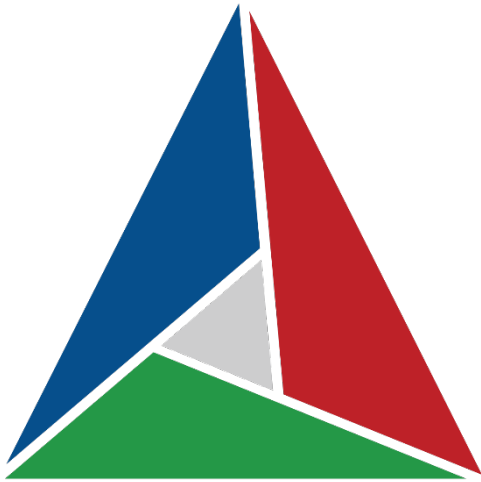


CLion

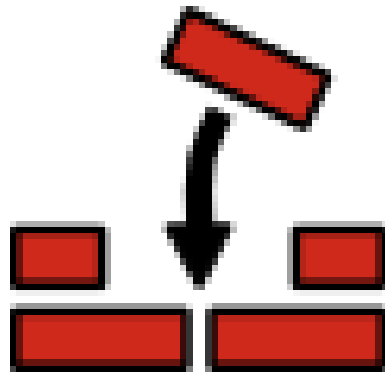


Visual
Studio Code
with
Docker

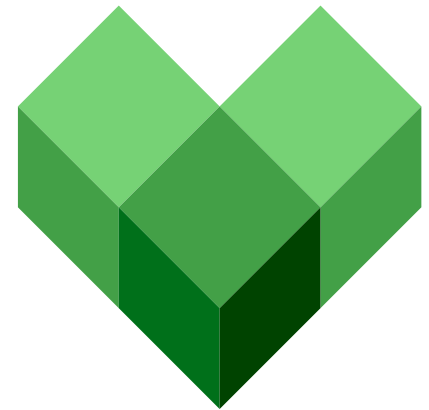
Project Builder



CMake



SCons



Bazel

Project Builder

- Cmake

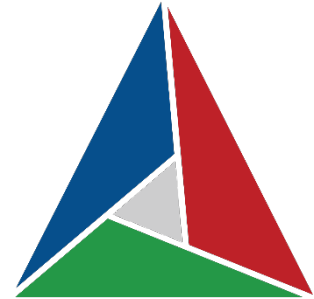
- C++ 프로젝트를 관리하기 위해 널리 사용되는 도구

- 장점

- 높은 호환성을 통해 여러 OS에서 동작함
 - IDE와의 연동성이 높음
 - 공식적으로 지원하는 IDE가 많음

- 단점

- 어려운 난이도의 문법을 필요로 하여 초기에 사용하기 어려움



Project Builder

- SCons

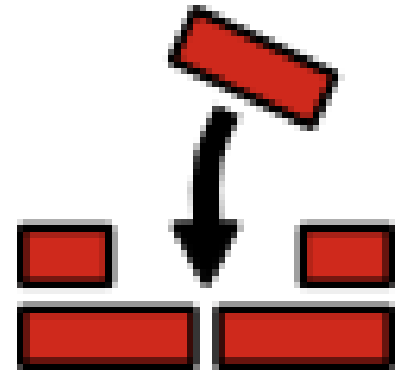
- Python 스크립트를 통해 빌드 방법 및 종속성을 정의

- 장점

- Python 스크립트 기반이므로 유연한 동작 정의 가능
 - 작성 및 동작 디버깅이 용이

- 단점

- Python 기반 동작에서 오는 낮은 IDE 연동성 및 높은 외부 라이브러리 관리 난이도



Project Builder

- Bazel

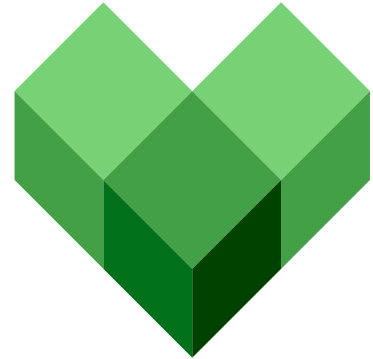
- C++를 포함해 여러 언어를 지원하는 빌드 자동화 도구

- 장점

- 빠른 프로젝트 빌드 속도
 - 대규모 프로젝트 관리에 유리

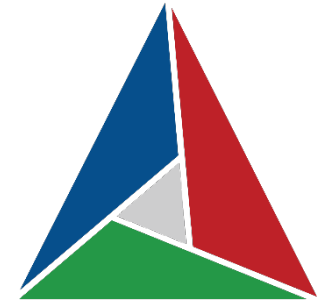
- 단점

- 기존 Makefile등과의 다른 구조로 인한 낮은 IDE 연동성

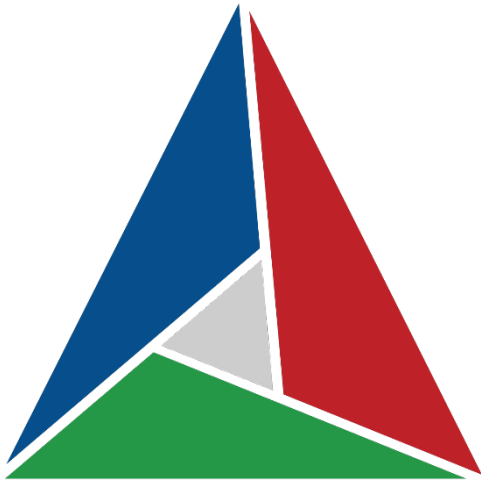


Project Builder

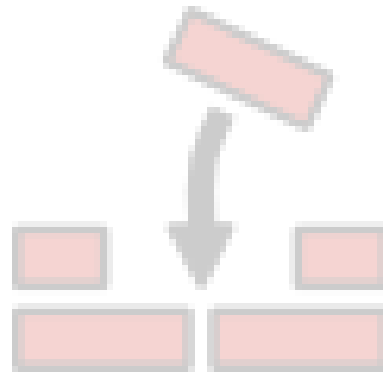
- Cmake + GNU Make
 - 이전 환경에서 정한 IDE와의 연동성 및 환경에 대한 호환성을 생각하여 Cmake로 결정
 - 널리 사용되는 툴인 만큼 안정성 및 호환성, 접근성이 가장 높음
 - Cmake를 통해 프로젝트를 관리하고, 빌드를 위한 백엔드로 Make를 사용



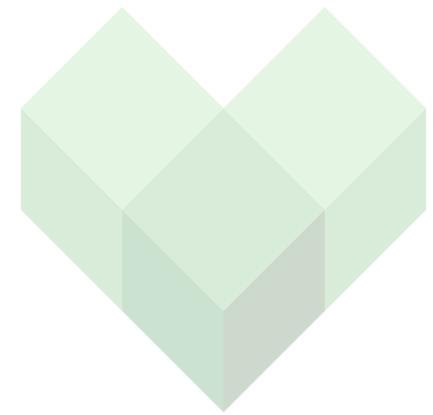
Project Builder



CMake



SCons



Bazel

Unit Test



- GoogleTest

- Google에서 개발한 C++ Unit Test 툴

- 장점

- C++ 테스트 자동화에 필요한 기능들을 광범위하게 지원 (현업수준 테스트에 적합)
 - 대규모 커뮤니티와 수 많은 레퍼런스
 - CI/CD 도구들과 쉬운 연동
 - GitHub stars 13k+, 지속적인 업데이트로 활발한 유지, 관리

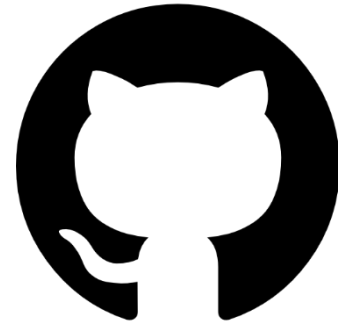
- 단점

- 상대적으로 설정이 복잡하고 무거움
 - 단순 테스트에는 부담이 될 수 있음

Code Version Management

- Github

- 온라인 코드 버전 관리 서비스
- 전 세계에서 가장 범용적으로 사용
- 개발 및 검증팀에서 제일 쉽게 사용할 수 있는 코드 버전 관리 서비스
- 협업, 코드 리뷰, CI/CD 기능까지 제공됨



CI Tool



Jenkins



Github
Actions

CI Tool

- Jenkins

- CI/CD를 지원하는 자동화 서버

- 장점

- 오픈 소스 및 무료
 - 1,700개 이상의 다양한 플러그인 지원
 - 다양한 언어 및 플랫폼 지원

- 단점

- 복잡한 설정
 - 불편한 UI/UX
 - 지원되지 않는 플러그인 의존성 등



CI Tool

- Github Actions

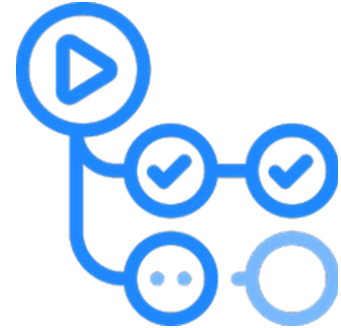
- Github 에서 제공하는 CI/CD 도구

- 장점

- Github 과의 통합이 용이함
 - 코드 푸시, 풀 리퀘스트 생성 등 다양한 작업 자동화 용이
 - 광범위한 액션 및 워크플로우 사용 가능

- 단점

- 디버깅이 어려움
 - 자원이 제한됨
 - Github Actions에서 제공하는 기본 서버의 성능이 낮음



CI Tool

- Github Actions with Docker

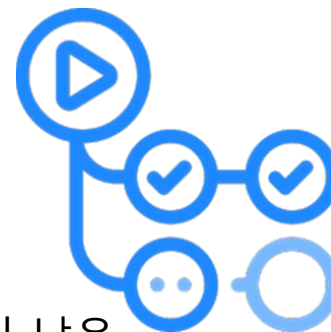
- 단점 개선

- 자원이 제한됨

- Github Actions에서 제공하는 기본 서버의 성능이 낮음

- » 을 개선하기 위해, 온프레미스 서버 구축 (Self-Hosted)

- 빌드 및 유닛 테스트 실패 시 온프레미스 서버 파일에 직접 접근 가능



```
sslab@sslab-kuase:~/actions-runner$ ./run.sh
```

```
✓ Connected to GitHub
```

```
jobs:
  test_job:
    name: Node Information
    runs-on: self-hosted
    steps:
      - name: check the node version
        run: cat /etc/issue && uname -a
      - name: check the node spec
        run: lscpu ; lsmem
```

```
Node Information
succeeded 2 weeks ago in 7s
Q Search logs
> ✓ Set up job
v ✓ check the node version
1 ▶ Run cat /etc/issue && uname -a
4 Ubuntu 22.04.5 LTS \n \l
5
6 Linux sslab-kuase 6.8.0-52-generic #53~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC
2 x86_64 x86_64 x86_64 GNU/Linux
v ✓ check the node spec
1 ▶ Run lscpu ; lsmem
4 Architecture:                x86_64
5 CPU op-mode(s):              32-bit, 64-bit
6 Address sizes:                39 bits physical, 48 bits virtual
7 Byte Order:                   Little Endian
8 CPU(s):                       10
9 On-line CPU(s) list:          0-9
```

온프레미스 환경 구축

워크플로우 파일 예시

워크플로우 실행 결과

CI Tool



Jenkins



Github
Actions

Requirements Management & Bug Tracking



GitHub
Project



Trello



REDMINE

Requirements Management & Bug Tracking

- Github Project

- GitHub와 통합된 요구사항 관리

- 장점

- GitHub Actions로 자동화 연결 가능
 - 코드 리뷰, 변경사항, 요구사항 간 자동 추적성 확보

- 단점

- 복잡한 요구사항 관계표현은 어려움
 - UI 커스터마이징이 제한됨



Requirements Management & Bug Tracking

- Trello

- 카드 기반 협업 도구



- 장점

- 직관적인 UI, 사용법이 매우 쉬움
 - 모바일/웹 환경에서 편리하게 사용 가능

- 단점

- 코드/이슈/빌드와의 연계 부족
 - CI/CD나 Git 연동은 추가 설정이 필요

Requirements Management & Bug Tracking

- REDMINE

- 이슈 중심의 프로젝트 관리 도구

- 장점

- 이슈 기반 요구사항 분해 및 추적 가능
 - 일정, 업무, 담당자 등 복합 관리 기능
 - 확장 플러그인 풍부

- 단점

- 설치 및 세팅의 번거로움
 - GitHub와 별도 연동 필요



Requirements Management & Bug Tracking



GitHub
Project



Trello



REDMINE

Team Communication



Slack



Discord

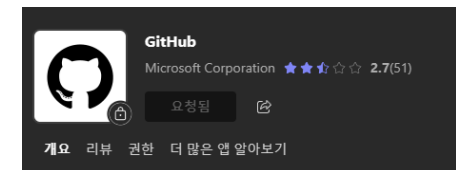


Microsoft
Teams

Team Communication



	Slack	Discord	Teams
음성/영상 통화	유료 버전 필요 1:1은 무료 가능	무료	무료
채팅	○	○	○
CI Server, Github 등의 알림	○	○	연동 어려움 (교내 Teams 미지원)
비용	일부 기능을 위해 유료 필수	무료	무료



Team Communication

Check "#36" [1] [Update cmake-single-platform.yml build와 test job 분리] #36

Summary

Jobs

- build
- test

Run details

Usage

Workflow file

cmake-single-platform.yml

on: push

build	14s
test	13s

Annotations

1 error

- test

An error occurred trying to start process '/usr/bin/bash' with working directory

오후 3:50 앱 SMA_TEST

1 Commit

KUASE-V3/SMA_TEST

ksj_test

6e2559 Update cmake-single-platform.yml

build와 test job 분리 by @s...r

2025-04-11 오후 3:50

오후 3:50 앱 GitHub

[SMA_TEST] build success on ksj_test

오후 3:50 앱 GitHub

[SMA_TEST] GitHub Actions checks failure on ksj_test



Github Actions



Discord

Team Communication



Slack



Discord



Microsoft
Teams

