

# OOAD: #1 Inception

2022113 엄정석 / 202211334 이동훈 / 202211384 차민우 / 202211392 최환

# Contents

## 1. Planning

- a. Vision
- b. FR (Functional Requirements)
- c. NFR (Non-Functional Requirements)
- d. Use-Cases
- e. Glossary

## 2. CI/CD 환경 구축

- a. Code & Version Control
- b. CI 환경
- c. Team Communication
- d. Pipeline

# Planning

Vision

# Vision

Customer의 초기 요구사항을 모두 만족하는  
**RVC 소프트웨어의 개발**

# Planning

Functional Requirements

# Functional Requirements

## FR-01. 기본 주행

- 시스템은 구동 시 직진 방향으로 이동하며 흡입(clean) 및 물걸레(mop) 청소 기능을 동시에 수행해야 한다.
- 모든 주행 및 청소 시나리오에서 '장애물 감지 및 회피(FR-02, FR-03)'는 '먼지 감지 및 강화 청소(FR-04)'보다 우선한다. 회피 동작이 시작되는 즉시 진행 중인 청소 모드 및 관련 타이머는 중단 및 초기화된다.

# Functional Requirements

## FR-02. 장애물 회피 - 일반

- 시스템은 전방 센서에 장애물이 감지된 경우, 탈출 경로를 확보하기 위해 판단 결과에 따라 아래와 같이 동작한다.
  - **[조건 A]** 양쪽이 비어있는 경우:  
좌측으로 회전한 후, 전진하며 청소를 재개한다.
  - **[조건 B]** 왼쪽이 장애물로 막힌 경우:  
우측으로 회전한 후, 전진하며 청소를 재개한다.
  - **[조건 C]** 오른쪽이 장애물로 막힌 경우: 좌측으로 회전한 후, 전진하며 청소를 재개한다.
- 모든 회전 동작은 시작 시 정해진 각도를 완전히 수행할 때까지 중단되지 않는다. 회전 동작 중에 추가적인 장애물이 감지되더라도 현재의 회전 동작을 끝까지 완료한 후 다음 상태(전진 또는 재판단)로 전이한다.

# Functional Requirements

## FR-03. 장애물 회피 - 3면 고립 상황 탈출

- 시스템은 전방, 좌측, 우측 센서에 모두 장애물이 감지되어 고립된 경우, 탈출 경로를 확보하기 위해 다음의 절차를 수행해야 한다.
  - a. 초기 회피 (후진): 3면 장애물 감지 즉시 3초 동안 후진한다.
  - b. 측면 공간 탐색: 3초 후진 완료 후, 좌측 및 우측 센서를 확인하여 탈출 가능 여부를 판단한다.
  - c. 조건별 탈출 동작: 판단 결과에 따라 아래와 같이 동작한다.
    - **[조건 A]** 양쪽 모두 빈 공간인 경우:  
좌측으로 회전한 후, 전진하며 청소를 재개한다.
    - **[조건 B]** 한쪽만 빈 공간인 경우:  
장애물이 없는 방향으로 회전한 후, 전진하며 청소를 재개한다.
    - **[조건 C]** 양쪽 모두 장애물로 막힌 경우:  
최대 10회까지 NFR-04의 안전 절차에 따라 위 절차를 반복한다.

# Functional Requirements

## FR-04. 먼지 감지 및 강화 청소 - Power Up 모드

- 시스템은 먼지 감지 시 즉시 흡입력을 강화(Power Up 모드)하며, 감지 시점으로부터 정확히 5초간 해당 상태를 유지한 후 일반 상태로 자동 전환해야 한다.
- Power Up 모드 유지 중 먼지가 재감지될 경우, 진행 중인 타이머를 유지한다.

# Functional Requirements

## FR-05. 환경 인식

- 전방/측면 센서 데이터를 실시간으로 분석하여 장애물 유무와 위치(전면, 좌측, 우측)를 판별해야 한다.

## FR-06. 상태 모니터링

- 현재 주행 상태(전진, 후진, 회전)와 청소 모드(일반, Power up)를 실시간으로 관리해야 한다.

# Planning

## Non-Functional Requirements

# Non-Functional Requirements

## NFR-01. 반응 및 회전 시간

장애물 감지 후 판단 지연 시간은 0.5초 이내여야 하며, 회전 명령 수행 시 0.5초 이내에 130도 회전을 완료해야 한다. 시간은 1초 이내여야 한다.

## NFR-02. 타이머 정밀도

Power Up 모드 유지 시간(5초)의 오차 범위는 0.1초 이내여야 한다.

## NFR-03. 충돌 방지

장애물 감지 센서의 정확도는 99% 이상이어야 하며, 물리적 충격을 최소화하도록 제어되어야 한다.

# Non-Functional Requirements

## NFR-04. 후진 동작 제약 및 시스템 보호

시스템은 고립 상황 탈출을 위해 반복적인 후진을 수행할 수 있으나, 후방 안전 확보를 위해 1회 후진 시간은 3초를 초과할 수 없으며, 각 후진 주기 사이에는 센서 재스캔을 위해 0.5초의 정지 시간을 가져야 한다. 또한, 총 10회 이상의 반복 후진 후에도 경로 미확보 시 시스템은 안전을 위해 에러를 출력하며 동작을 정지한다.

## NFR-05. 하드웨어 독립성

제어 소프트웨어(RVC SW)는 특정 모터 드라이버나 센서 모델에 종속되지 않도록 추상화된 인터페이스를 사용해야 한다.

## NFR-06. 범위 제한

본 설계는 '자동 청소 기능'에 한정하며, 수동 조작이나 배터리 관리 루틴은 포함하지 않는다.

# Planning

## Use-Cases

# Use-Cases

Functional Requirements	Use-Case Number & Name	Actor
FR-01. 기본 주행	1. 기본 주행	RVC
FR-02. 장애물 회피 - 일반	2. 장애물 감지 - 좌회전	RVC
FR-02. 장애물 회피 - 일반	3. 장애물 감지 - 우회전	RVC
FR-03. 장애물 회피 - 3면 고립 상황 탈출	4. 장애물 감지 - 후진	RVC
FR-03. 장애물 회피 - 3면 고립 상황 탈출	5. 최종 탈출 불가	RVC
FR-04. 먼지 감지 및 강화 청소 - Power Up 모드	6. 먼지 감지	RVC

# Use-Cases

<b>Use Case</b>	1. 기본 주행
<b>Actors</b>	RVC
<b>Description</b>	<ul style="list-style-type: none"><li>• RVC가 작동을 시작하면 직진하며 바닥의 모든 구역 청소를 목표로 움직이며 청소를 진행한다.</li></ul>

# Use-Cases

<b>Use Case</b>	2. 장애물 감지 - 좌회전
<b>Actors</b>	RVC
<b>Description</b>	<ul style="list-style-type: none"><li>• RVC가 직진하며 전면과 좌/우측의 장애물 감지 센서를 통해 장애물을 감지한다.</li><li>• 전면, 혹은 전면과 우측 센서가 장애물을 감지하면, RVC는 직진을 멈추고 좌측으로 회전하여 장애물을 피해 경로를 확보한다. 회전 완료 이후 직진 및 청소를 재개한다.</li><li>• 회전 중 추가 장애물이 감지되더라도 중단 없이 회전을 완수한다.</li></ul>

# Use-Cases

<b>Use Case</b>	3. 장애물 감지 - 우회전
<b>Actors</b>	RVC
<b>Description</b>	<ul style="list-style-type: none"><li>• RVC가 직진하며 전면과 좌/우측의 장애물 감지 센서를 통해 장애물을 감지한다.</li><li>• 전면과 좌측 센서가 장애물을 감지하면, RVC는 직진을 멈추고 우측으로 회전하여 장애물을 피해 경로를 확보한다. 회전 완료 이후 직진 및 청소를 재개한다.</li><li>• 회전 중 추가 장애물이 감지되더라도 중단 없이 회전을 완수한다.</li></ul>

# Use-Cases

<b>Use Case</b>	4. 장애물 감지 - 후진
<b>Actors</b>	RVC
<b>Description</b>	<ul style="list-style-type: none"><li>• RVC가 직진하며 전면과 좌/우측의 장애물 감지 센서를 통해 장애물을 감지한다.</li><li>• 전면과 좌측, 우측 센서가 모두 장애물을 감지하면, RVC는 직진을 멈추고 3초간 후진한다.</li><li>• 3초 후진 완료 후, 좌/우측의 장애물 감지 센서를 확인하여 탈출 가능 여부를 파악한다. 판단 결과에 따라 FR-03.c와 같이 동작한다.</li></ul>

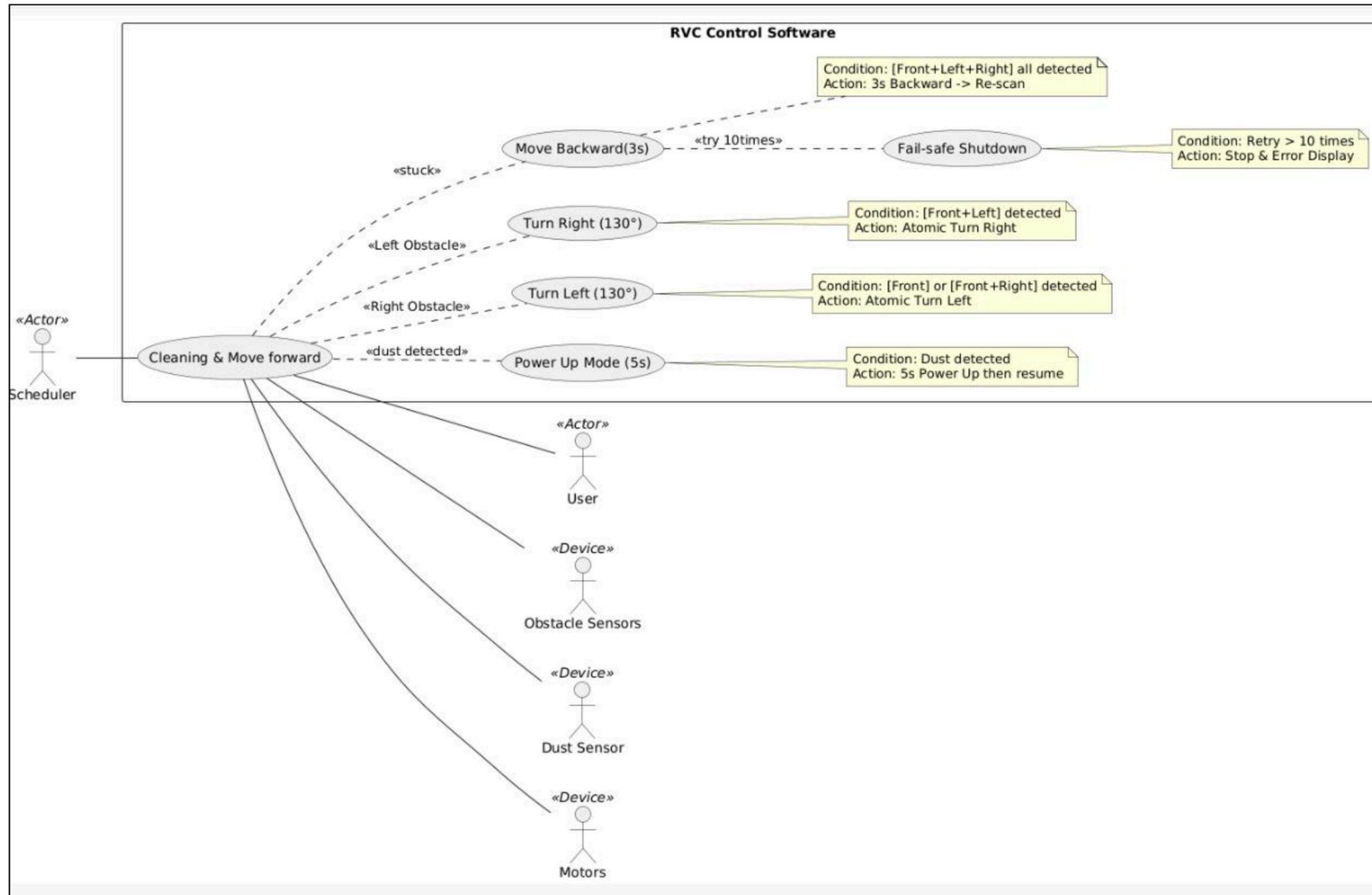
# Use-Cases

<b>Use Case</b>	5. 최종 탈출 불가
<b>Actors</b>	RVC
<b>Description</b>	<ul style="list-style-type: none"><li>RVC가 '4. 장애물 감지 - 후진' 상황을 연속으로 10번 반복한 후에도 회전을 통한 경로를 확보하지 못한 경우 하드웨어 과부하 및 무한 루프 방지를 위해 시스템을 정지한다. 이후 사용자에게 에러 메시지를 출력하고 시스템을 종료한다.</li></ul>

# Use-Cases

<b>Use Case</b>	6. 먼지 감지
<b>Actors</b>	RVC
<b>Description</b>	<ul style="list-style-type: none"><li>• RVC가 청소 중 먼지 감지 센서를 통해 바닥의 먼지를 감지한다.</li><li>• 먼지가 감지되면 Power Up 모드를 5초간 실행한 후 청소 동작으로 복귀한다.</li></ul>

# Use-Cases Diagram



# Planning

## Glossary

# Glossary

## **RVC**

Robotic Vacuum Cleaner

## **Obstacle**

전면, 좌측, 우측 센서에 의해 감지되는 물리적 방해물

## **Dust**

전면, 좌측, 우측 센서에 의해 감지되는 먼지

# Glossary

## Power Up 모드

RVC의 먼지 흡입 강도를 높입니다

## Atomic Operation

회전 동작 시작 시, 외부 센서의 추가 입력이나 방해에 상관 없이 미리 설정된 각도를 반드시 완수하는 제어 방식

# CI/CD 환경 구축

## Code & Version Control

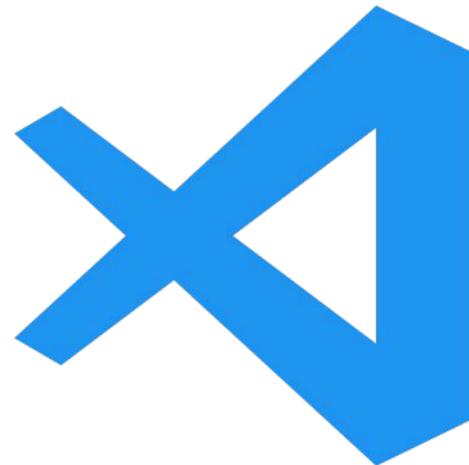
# Code & Version Control

## Language



C++

## IDE



Visual Studio Code

## Version Control



Git & GitHub

# CI/CD 환경 구축

CI 환경

# CI 환경: Build System & Unit Test Tool

## Build System



CMake

## Unit Test Tool



Google Test

# CI 환경: Static Code Analysis

## Static Analysis Tool



SonarQube Cloud

## Static Analysis Tool



cppCheck

## Static Analysis Tool



cppLint

# CI 환경: CI Tool 후보

CI Tool



Github Actions

CI Tool



Jenkins

Github Actions

CI Tool



Github Actions

# CI 환경: CI Tool 결정

## CI Tool



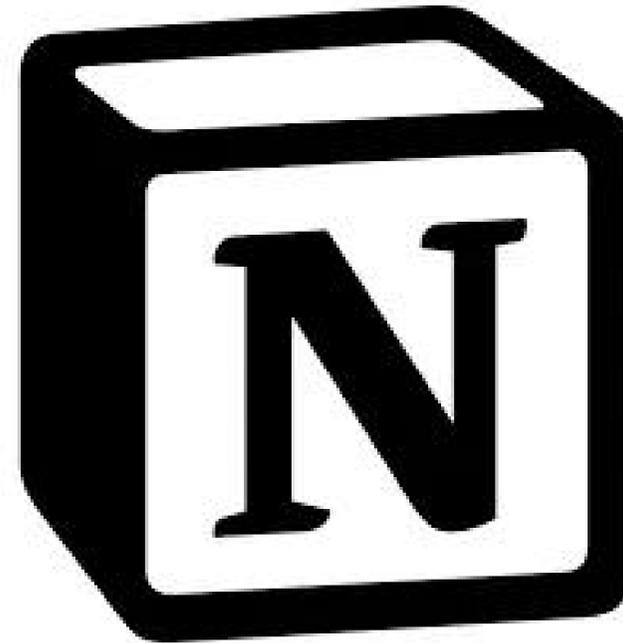
GitHub Actions

# CI/CD 환경 구축

Team Communication

# Team Communication

## Communication Tool

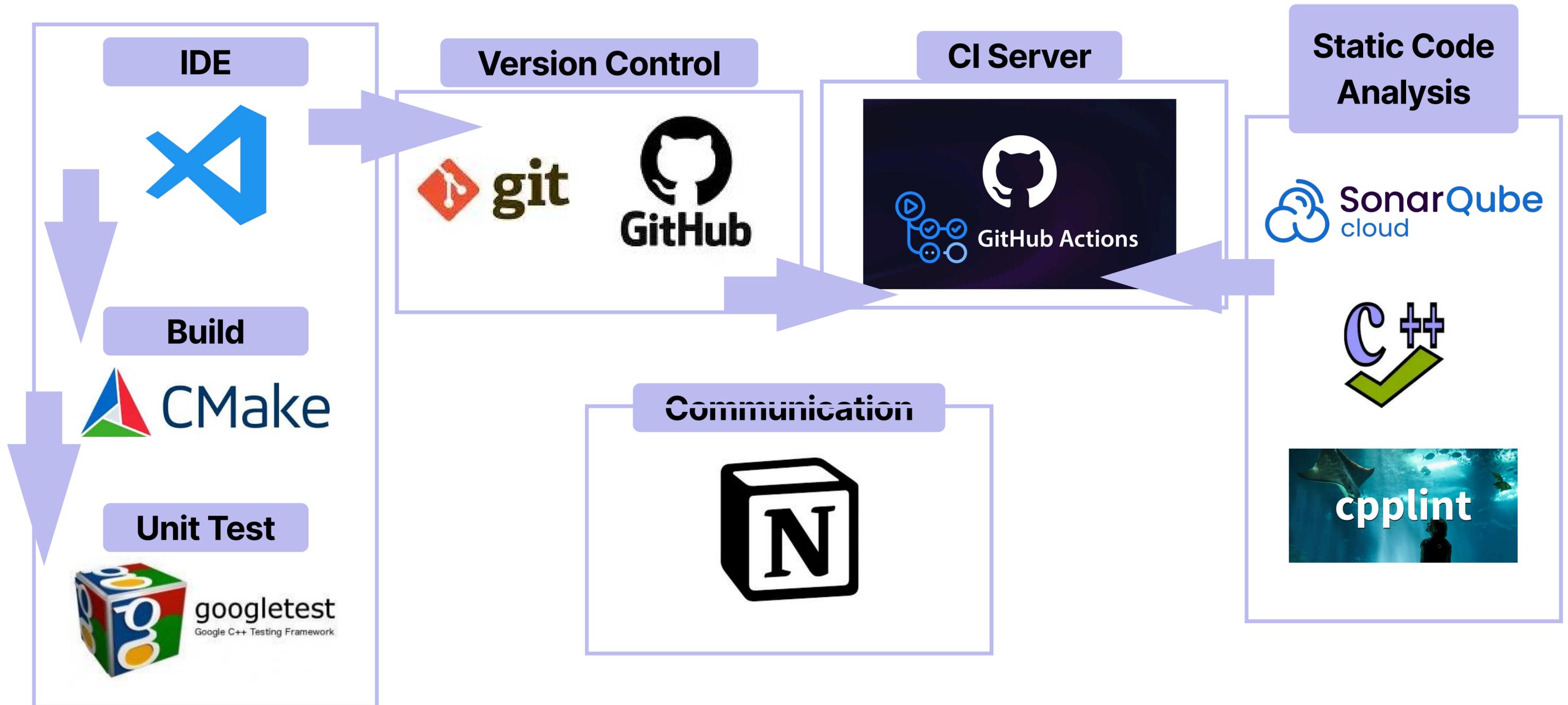


Notion

# CI/CD 환경 구축

Pipeline

# Pipeline



**감사합니다.**