

# 객체지향개발방법론

## UP - Inception

# INDEX

목차페이지

- 01 Use Case
- 02 Use Case Diagram
- 03 Functional Requirements
- 04 Non-Functional Requirements
- 05 Use Case Diagram
- 06 CI/CD 환경
- 07 CI/CD 결과

# Use Case



|                    |   |
|--------------------|---|
| <b>Use Case</b>    | <b>1. 자동 청소 제어</b>  |
| <b>Actors</b>      | User, Obstacle Sensor, Dust Sensor, Motor, Cleaner  |
| <b>Description</b> | <ul style="list-style-type: none"><li>• 사용자가 시스템을 가동한다.</li><li>• 시스템은 먼지와 장애물 상황에 따라 자율적으로 청소한다.</li></ul> |

# Use Case



|                    |   |
|--------------------|---|
| <b>Use Case</b>    | <b>2. 직선 청소 주행</b>  |
| <b>Actors</b>      | Digital Clock, Motor  |
| <b>Description</b> | <ul style="list-style-type: none"><li>• 시스템은 전방으로 주행하며 바닥면을 청소한다.</li><li>• 별도의 장애물이 없는 한 직선 경로를 유지하며 청소 기능을 수행한다.</li></ul>  |
| <b>Use Case</b>    | <b>3. 장애물 회피</b>  |
| <b>Actors</b>      | Obstacle Sensor, Motor  |
| <b>Description</b> | <ul style="list-style-type: none"><li>• 주행 중 전방의 센서로부터 장애물이 감지된다.</li><li>• 시스템은 전진을 멈추고, 장애물이 없는 좌측 혹은 우측으로 방향을 전환한다.</li><li>• 전환이 완료되면 직선 주행 및 청소를 재개한다.</li></ul> |

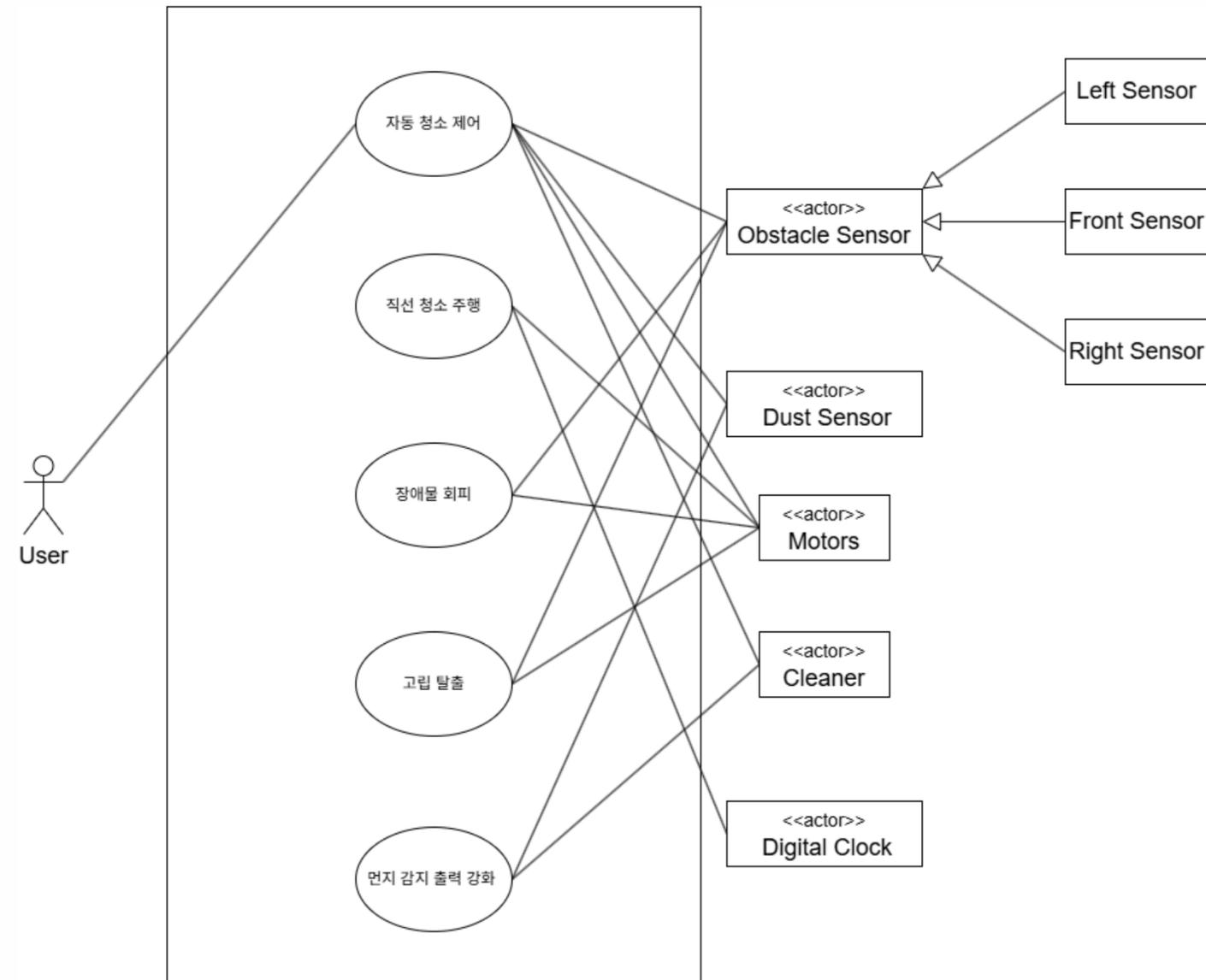
# Use Case



|                    |   |
|--------------------|---|
| <b>Use Case</b>    | <b>4. 고립 탈출</b>   |
| <b>Actors</b>      | Obstacle Sensor, Motor  |
| <b>Description</b> | <ul style="list-style-type: none"><li>• 주행 중 전방, 좌측, 우측 센서로부터 장애물이 감지된다.</li><li>• 시스템은 전진을 멈추고 일정 시간 후진하여 회전 공간을 확보한다.</li><li>• 이후 장애물이 감지되지 않는 방향으로 회전하여 직진 주행 및 청소를 재개한다.</li></ul> |

|                    |   |
|--------------------|---|
| <b>Use Case</b>    | <b>5. 먼지 감지 출력 강화</b>   |
| <b>Actors</b>      | Dust Sensor, Cleaner  |
| <b>Description</b> | <ul style="list-style-type: none"><li>• 주행 중 먼지 센서로부터 먼지가 감지된다.</li><li>• 시스템은 일시적으로 청소기의 흡입 성능을 높인다.</li></ul> |

# Use Case Diagram



# Functional Requirements

| ID      | 요구사항 명칭       | 상세 설명  | 관련 Use Case |
|---------|---------------|--|-------------|
| FR - 01 | 자동 쓸기 및 닦기    | 시스템은 사용자의 명령에 따라 자동으로 바닥 청소와 물걸레질을 수행해야 한다.              | UC - 01     |
| FR - 02 | 직진 주행         | 시스템은 청소 중 장애물이 없는 경우 직진으로 주행해야 한다.                       | UC - 02     |
| FR - 03 | 장애물 감지 및 정지   | 시스템은 주행 중 전방 센서를 통해 장애물을 감지하면 즉시 주행과 청소를 중단해야 한다.        | UC - 03     |
| FR - 04 | 우선순위 회피 주행    | 전방 장애물 발생 시 [우측 → 좌측] 순으로 가용 공간을 확인하여 방향을 전환해야 한다.       | UC - 03     |
| FR - 05 | 복합 장애물 대응     | 전방, 좌, 우가 모두 차단된 경우 후진 후 우측으로 방향을 전환하여 주행을 재개해야 한다.      | UC - 04     |
| FR - 06 | 먼지 감지 및 출력 강화 | 먼지 감지 시 3초 동안 흡입력을 강화한 후 표준 모드로 복귀해야 한다.                 | UC - 05     |
| FR - 07 | 안전 종료         | 시스템은 종료 명령 수신 시, 주행 및 청소 알고리즘을 중단하고 시스템을 종료 상태로 전환해야 한다. | UC - 01     |

# Non-Functional Requirements

| ID       | 구분                     | 상세 설명  |
|----------|------------------------|--|
| NFR - 01 | Performance Efficiency | 센서 데이터를 통한 상황 판단 및 결정 시간은 0.5초 이내여야 한다.          |
| NFR - 02 | Performance Efficiency | 전원 On/Off 프로세스 완료까지 소요 시간은 1초 이내여야 한다.           |
| NFR - 03 | Portability            | 시스템은 리눅스(Linux) 환경에서 정상 작동을 보장해야 한다.             |
| NFR - 04 | Constraints            | 구현 언어는 C++ 을 사용하며, 단위 테스트는 gTest 프레임워크를 사용한다.    |
| NFR - 05 | Constraints            | 하드웨어 상세 로직에 의존하지 않도록 추상화된 소프트웨어 컨트롤러 수준에서 설계한다.  |
| NFR - 06 | Maintainability        | 향후 새로운 센서 추가나 머신러닝 모듈 도입이 용이하도록 모듈화된 구조를 가져야 한다. |

# ✓ CI/CD 환경

