

# TEAM 2

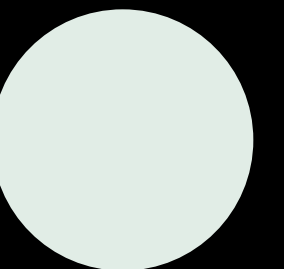
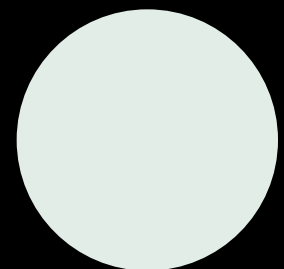
## *Object Oriented Analysis*

강병완 202211248

강현준 202211251

박 완 202211301

정민수 202211365

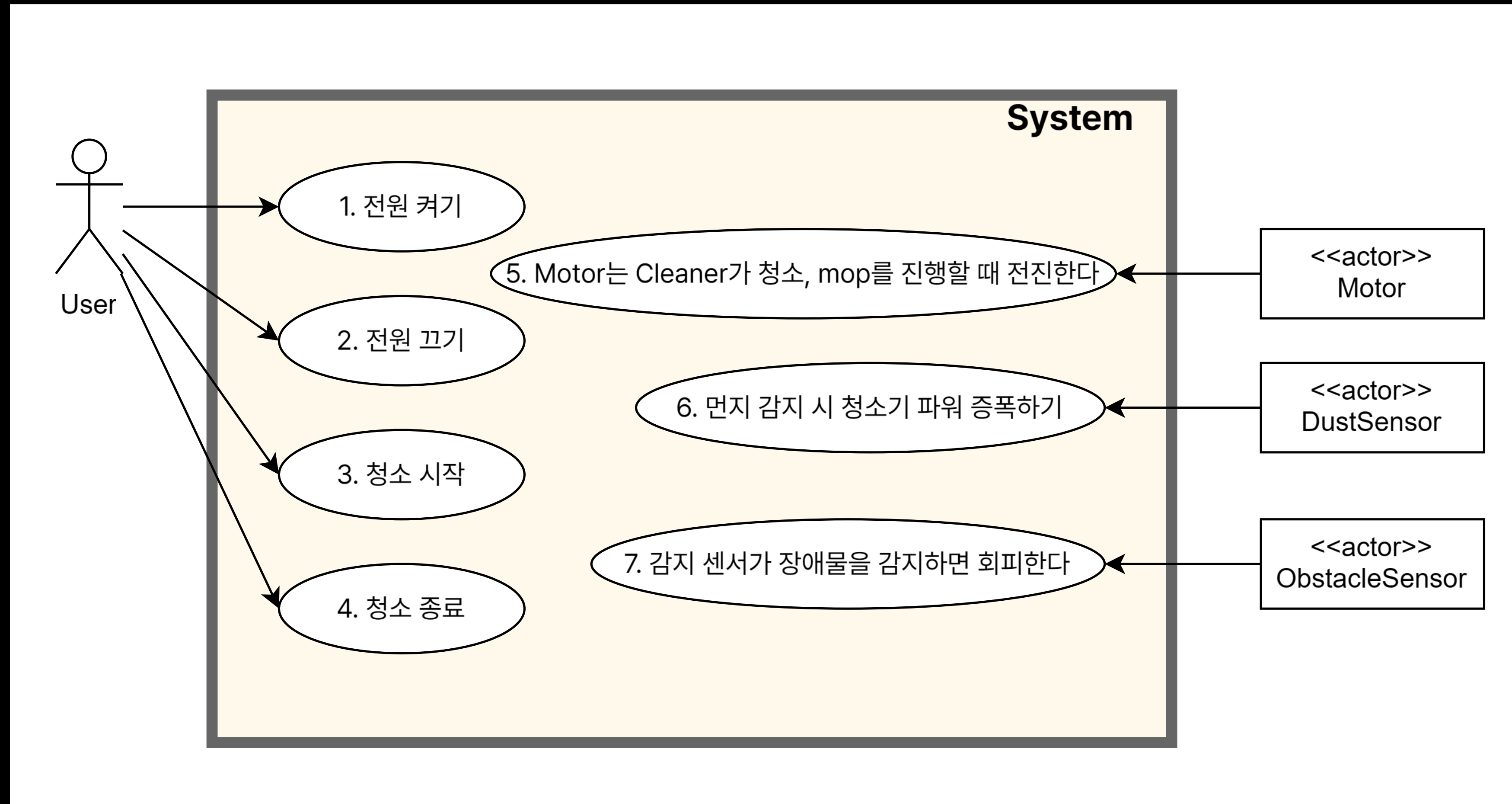


# Table of Content

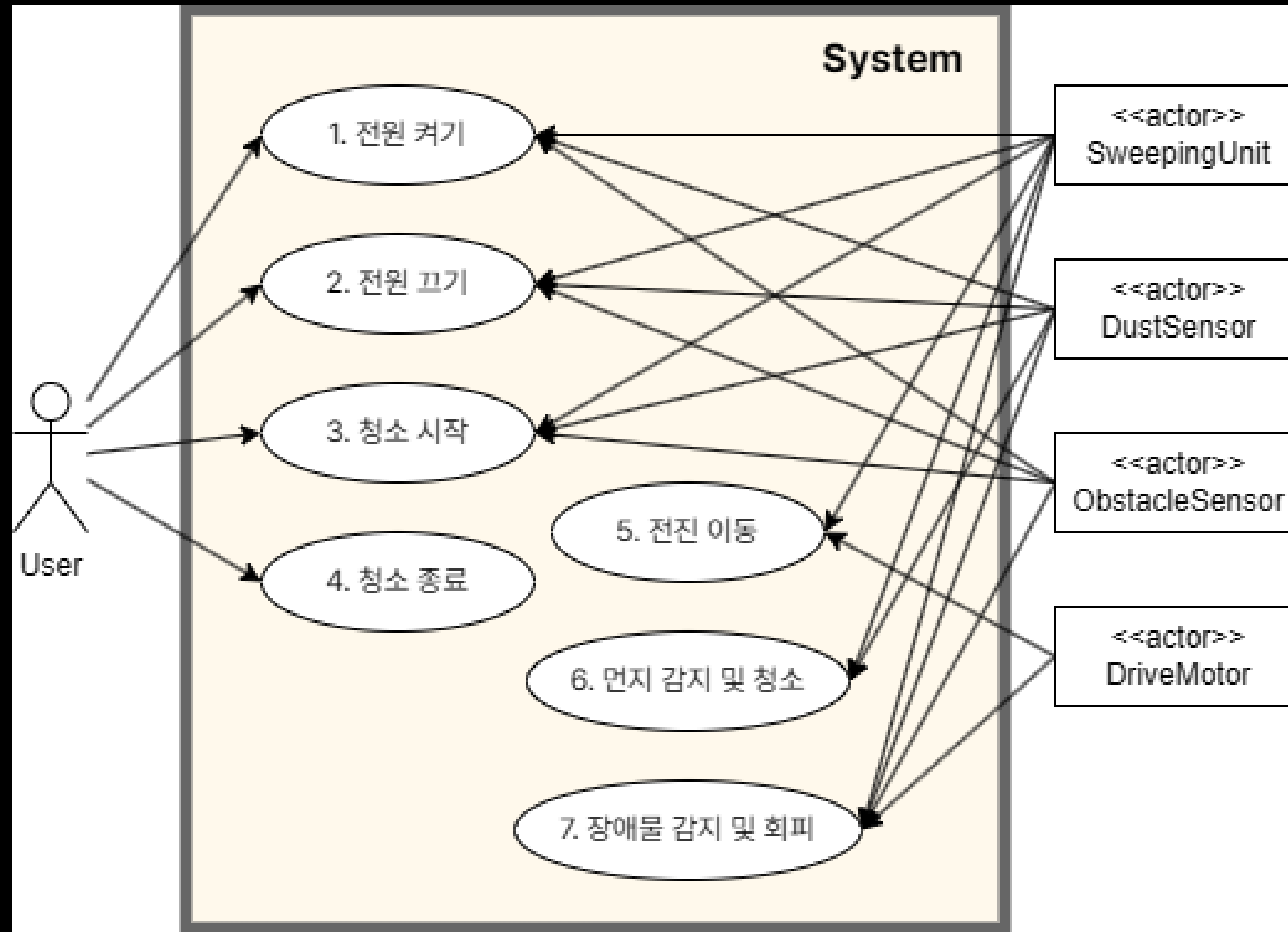
- Changes
- Use Case
- System Sequence Diagram
- System Operations
- Domain Model

# Changes #2

# Use case Diagram



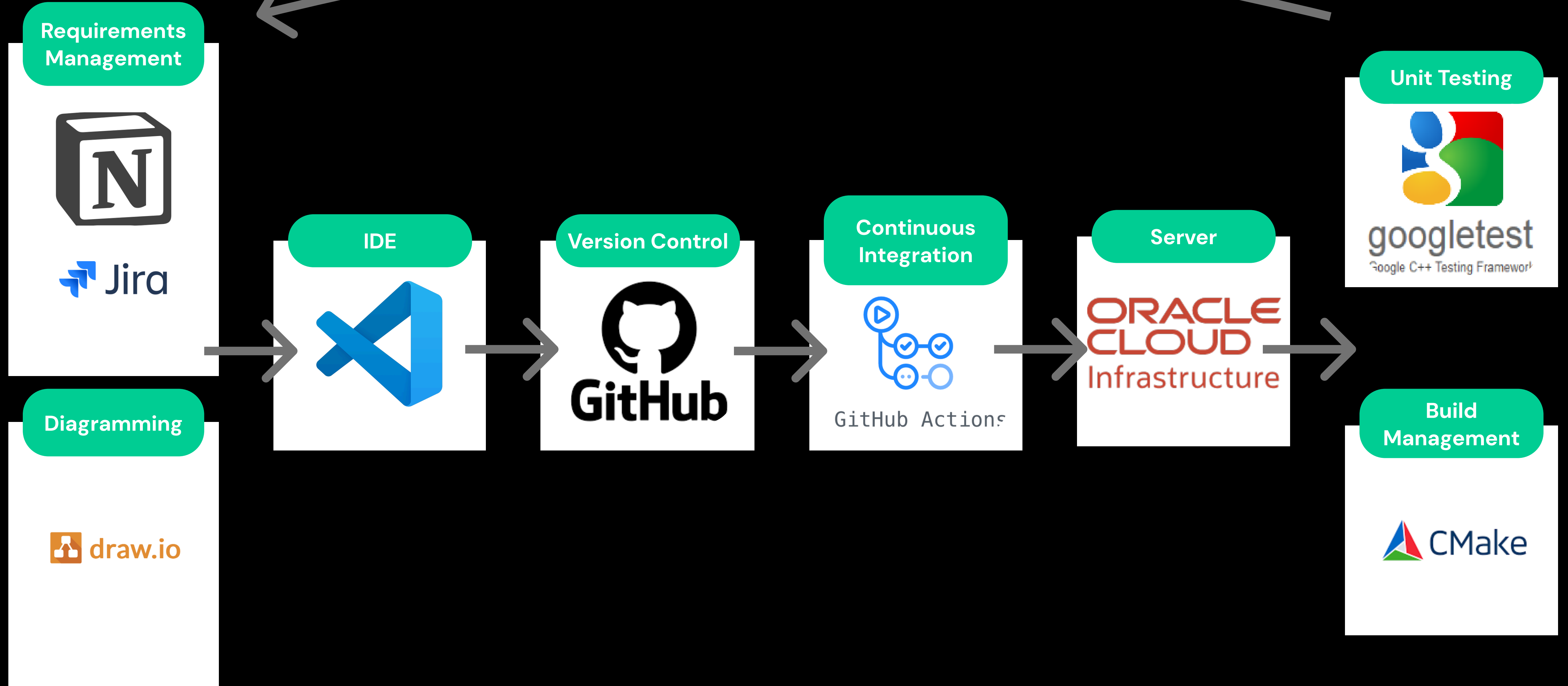
# Use case Diagram



# Changes #3

# CI/CD

Before

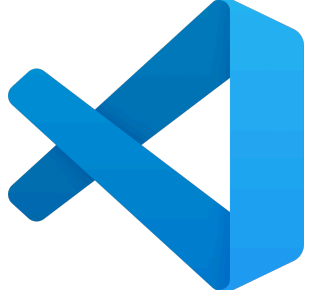


# CI/CD


# Update

## Development


**IDE**



**Build Management**



**Unit Testing**




googletest  
Google C++ Testing Framework

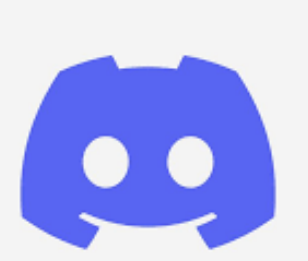
**Version Control**




**Requirements Management**



**Team Communication**




**Server**




**CI**

**Continuous Integration**




GitHub Actions

**Build Management**




**Unit Testing**




googletest  
Google C++ Testing Framework

**Static Code Analysis**



**Coverage**



gcovr

# CI/CD

# Before

Summary

All jobs

- build-and-test

Run details

- Usage
- Workflow file

### build-and-test

succeeded 3 days ago in 1m 6s


Search logs

- > ✓ Set up job 3s
- > ✓ Checkout 4s
- > ✓ Configure 7s
- > ✓ Build 43s
- ▼ ✓ Test 0s
  - 1 ▶ Run `ctest --test-dir project/build --output-on-failure`
  - 4 Internal ctest changing into directory: `/home/ubuntu/actions-runner-00AD/00AD/KONKUK-00AD-2026-1/KONKUK-00AD-2026-1/project/build`
  - 5 Test project `/home/ubuntu/actions-runner-00AD/00AD/KONKUK-00AD-2026-1/KONKUK-00AD-2026-1/project/build`
  - 6 Start 1: MathTest.Add
  - 7 1/2 Test #1: MathTest.Add ..... Passed 0.01 sec
  - 8 Start 2: MathTest.Sub
  - 9 2/2 Test #2: MathTest.Sub ..... Passed 0.01 sec
  - 10
  - 11 100% tests passed, 0 tests failed out of 2
  - 12
  - 13 Total Test time (real) = 0.09 sec
- > ✓ Post Checkout 3s
- > ✓ Complete job 0s

# CI/CD

# UPDATE

🏠 Summary

All jobs 

- ✅ build-test

Run details

- 🕒 Usage
- 📄 Workflow file

### build-test

succeeded 11 hours ago in 4m 2s

- > ✅ Set up job
- > ✅ Checkout
- > ✅ Set up Python
- > ✅ Detect ccache
- ⌚ Cache ccache
- > ✅ Install gcovr
- > ✅ Configure
- > ✅ Build
- > ✅ clang-tidy
- > ✅ Test
- > ✅ Generate coverage report
- > ✅ Upload coverage to Codecov
- > ✅ Post clang-tidy
- > ✅ Post Set up Python
- > ✅ Post Checkout
- > ✅ Complete job

Before

# Use Case

#	Ref. #	Aa Use-Case Name	☰ Actor
1		📄 1. 전원 켜기	User (Primary)
2		📄 2. 전원 끄기	User (Primary)
3		📄 3. 청소 시작	User (Primary) Cleaner(Supporting) DustSensor(Supporting) ObstacleSensor(Supporting)
4		📄 4. 청소 종료	User (Primary) Cleaner(Supporting) DustSensor(Supporting) ObstacleSensor(Supporting)
5		📄 5. Motor는 Cleaner가 켜져 있을 때 전진한다	Motor(Supporting) Cleaner(Supporting)
6		📄 6. 먼지 감지 시 청소기 파워 증폭하기	DustSensor(Supporting) Cleaner(Supporting)
7		📄 7. ObstacleSensor가 장애물을 감지하면 회피한다	ObstacleSensor(Supporting) Motor(Supporting) Cleaner(Supporting)

# Use Case

#	Ref. #	Aa Use-Case Name	☰ Actor
1	📄	1. 전원 켜기	User (Primary) SweepingUnit(Supporting) DustSensor(Supporting) ObstacleSensor(Supporting)
2	📄	2. 전원 끄기	User (Primary) SweepingUnit(Supporting) DustSensor(Supporting) ObstacleSensor(Supporting)
3	📄	3. 청소 시작	User (Primary) SweepingUnit(Supporting) DustSensor(Supporting) ObstacleSensor(Supporting)
4	📄	4. 청소 종료	User (Primary)
5	📄	5. 전진 이동	DriveMotor(Supporting) SweepingUnit(Supporting)
6	📄	6. 먼지 감지 및 청소	DustSensor(Supporting) SweepingUnit(Supporting)
7	📄	7. 장애물 감지 및 회피	ObstacleSensor(Supporting) DriveMotor(Supporting) SweepingUnit(Supporting) DustSensor(Supporting)

# Use Case #1

<b>Name</b>	1. 전원 켜기
<b>Actor</b>	User (Primary)
<b>Pre-Requisites</b>	RVC의 전원이 꺼져있는 상태이다.
<b>Typical Courses of Events</b>	(R) : RVCSystem, (U) : User 1. (U)가 (R)의 전원을 켜다. 2. (R)의 전원이 켜진다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

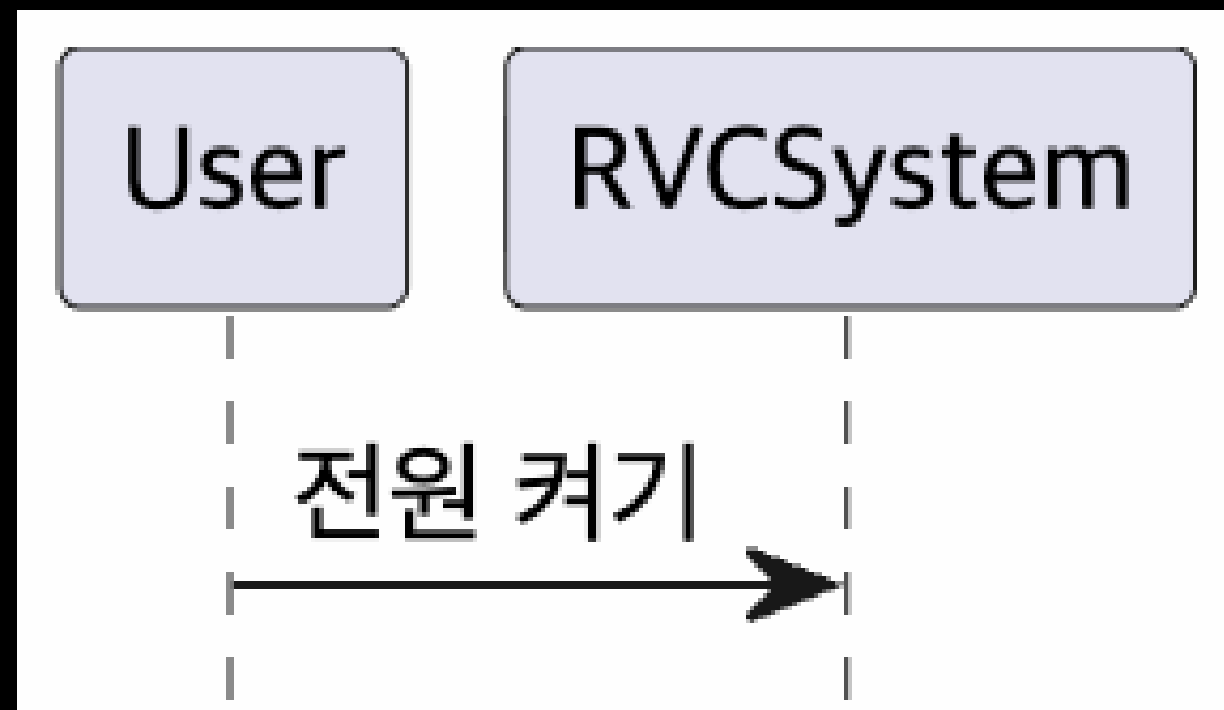
# Use Case #1

Update

<b>Name</b>	1. 전원 켜기
<b>Actor</b>	User (Primary)
<b>Pre-Requisites</b>	RVC의 전원이 꺼져있는 상태이다.
<b>Typical Courses of Events</b>	(R): RVCSytem, (U): User, (S): SweepingUnit, (D): DustSensor, (O): ObstacleSensor 1. (U)가 (R)의 전원을 킨다. 2. (R)의 전원이 켜진다. 3. (R)이 (S)의 전원을 킨다. 4. (R)이 (D)의 전원을 킨다. 5. (R)이 (O)의 전원을 킨다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #1

Before



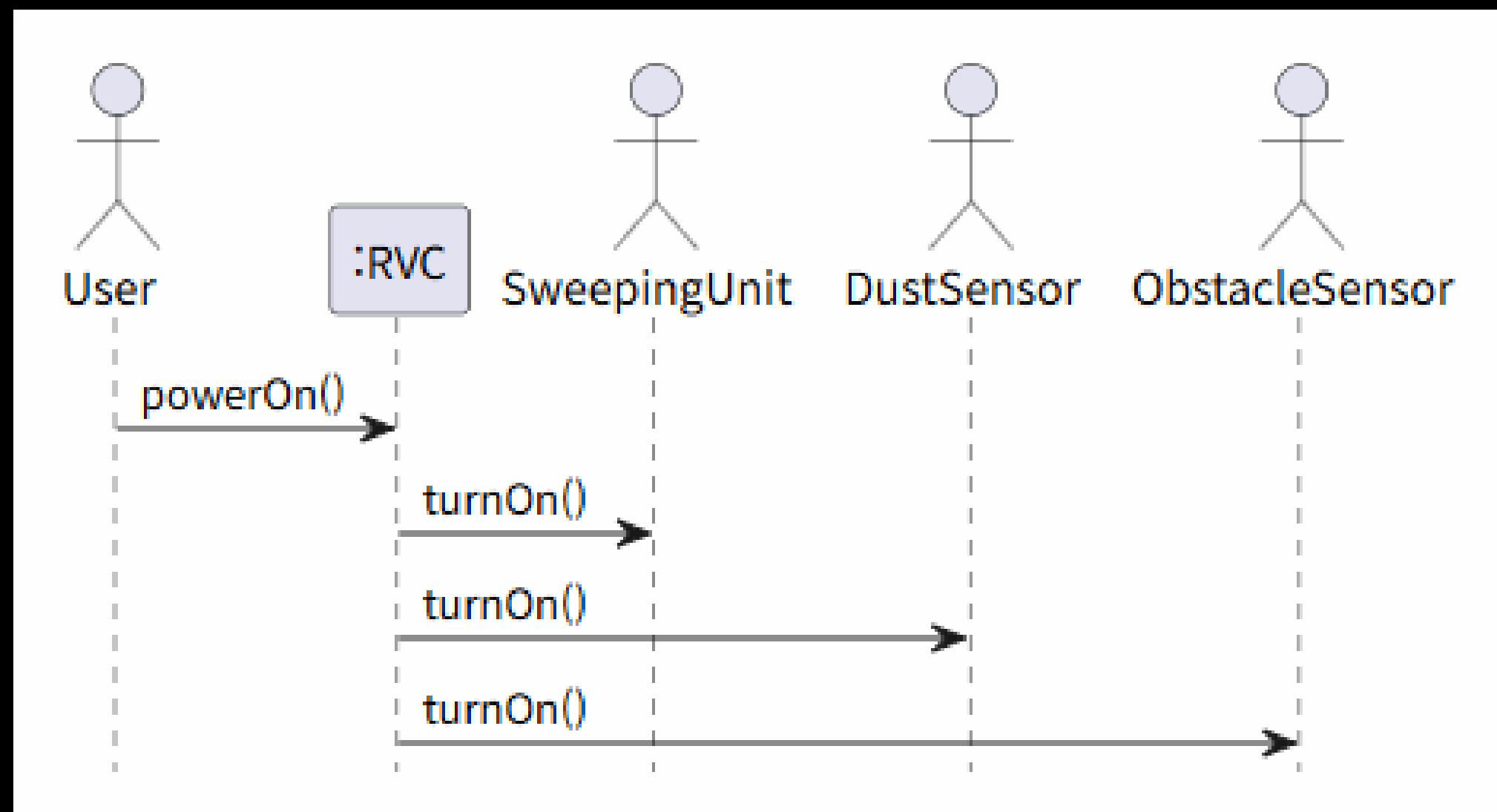
+ 전원 켜기

System Sequence Diagram

System Operations

# Use Case #1

Update



+ powerOn()

System Sequence Diagram

System Operations

# Use Case #2

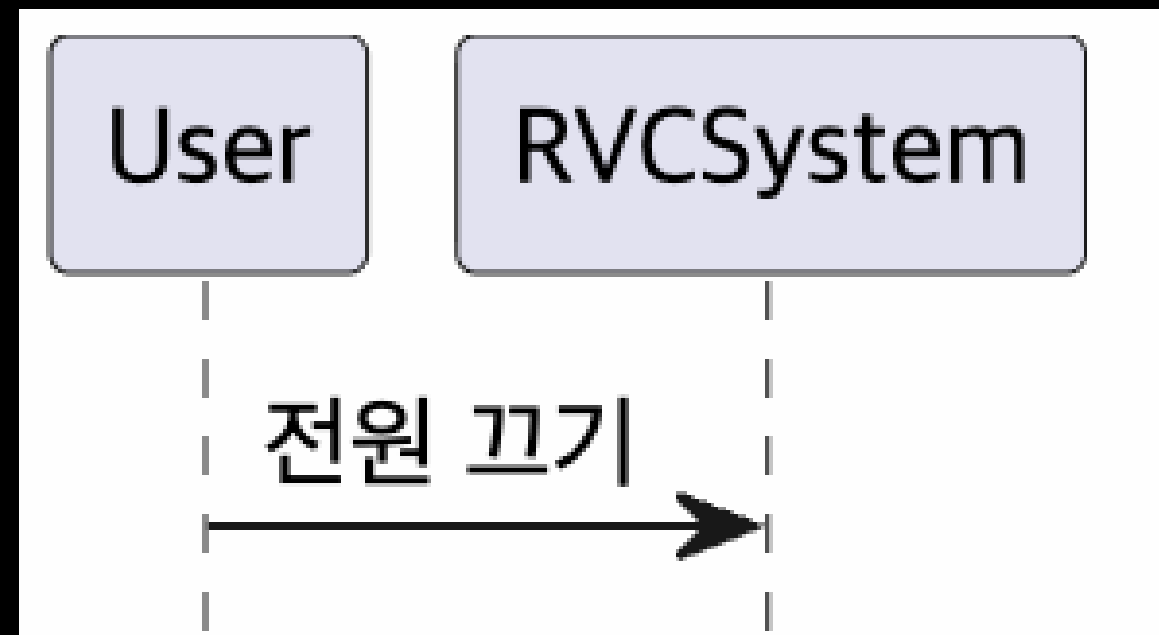
<b>Name</b>	2. 전원 끄기
<b>Actor</b>	User (Primary)
<b>Pre-Requisites</b>	RVC의 전원이 켜져 있음.
<b>Typical Courses of Events</b>	(R) : RVCSsystem, (U) : User 1. (U)가 (R)의 전원을 끈다. 2. (R)의 전원이 꺼진다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #2

<b>Name</b>	2. 전원 끄기
<b>Actor</b>	User (Primary)
<b>Pre-Requisites</b>	RVC의 전원이 켜져있는 상태이다.
<b>Typical Courses of Events</b>	(R): RVCSytem, (U): User, (S): SweepingUnit, (D): DustSensor, (O): ObstacleSensor 1. (U)가 (R)의 전원을 끈다. 2. (R)이 (S)의 전원을 끈다. 3. (R)이 (D)의 전원을 끈다. 4. (R)이 (O)의 전원을 끈다. 5. (R)의 전원이 꺼진다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #2

Before



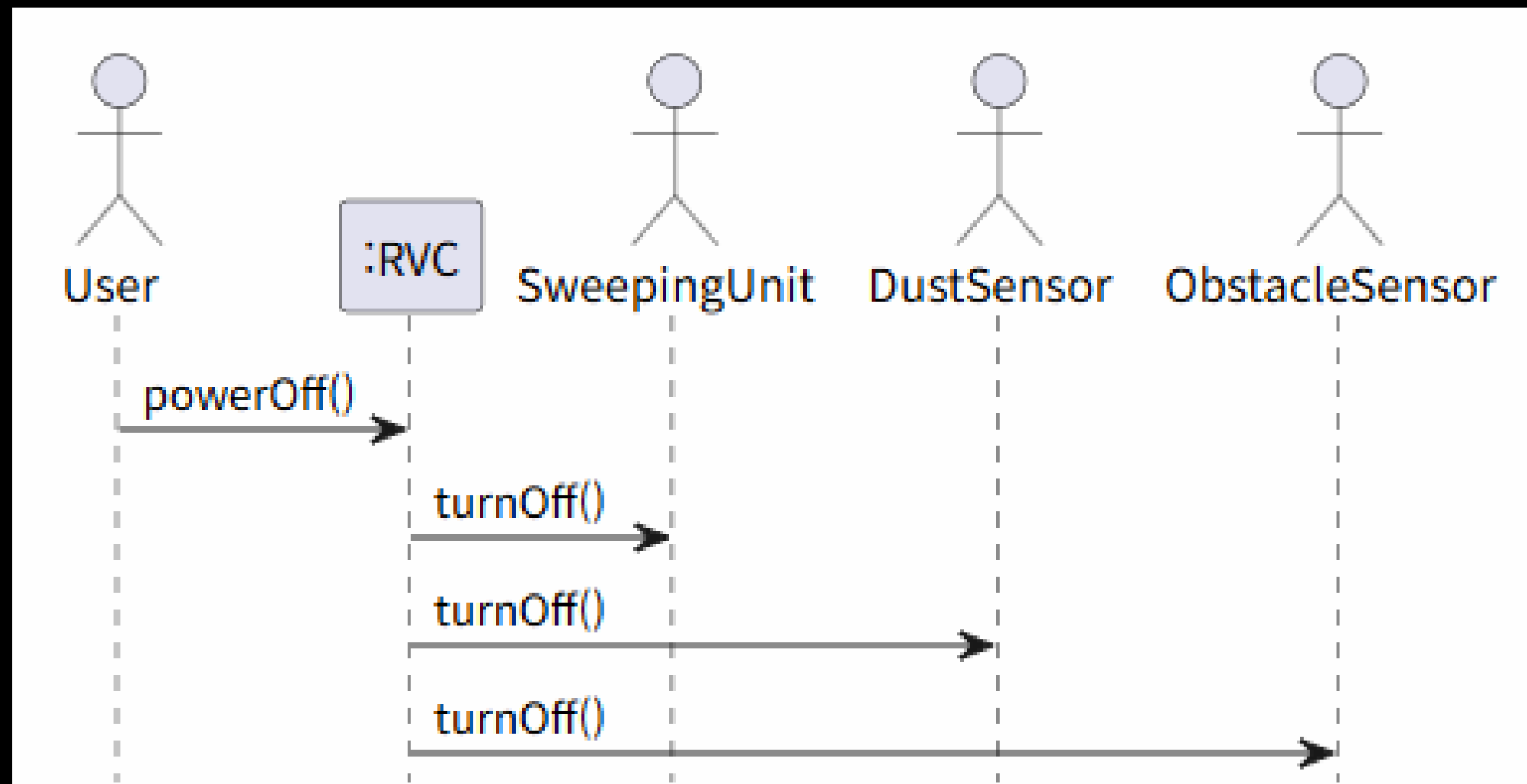
+ 전원 끄기

System Sequence Diagram

System Operations

# Use Case #2

Update



+ powerOff()

System Sequence Diagram

System Operations

# Use Case #3

<b>Name</b>	3. 청소 시작
<b>Actor</b>	User(Primary), Cleaner(Supporting), ObstacleSensor(Supporting), DustSensor(Supporting)
<b>Pre-Requisites</b>	RVC의 전원이 켜져있다.
<b>Typical Courses of Events</b>	(R) : RVCSysyem, (U) : User, (C) : Cleaner, (O) : ObstacleSensor, (D) : DustSensor 1. (U)이 (R)에게 청소시작을 지시 한다. 2. (R)이 (C)를 킨다. 3. (R)이 (D)의 전원을 킨다. 4. (R)이 (O)의 전원을 킨다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

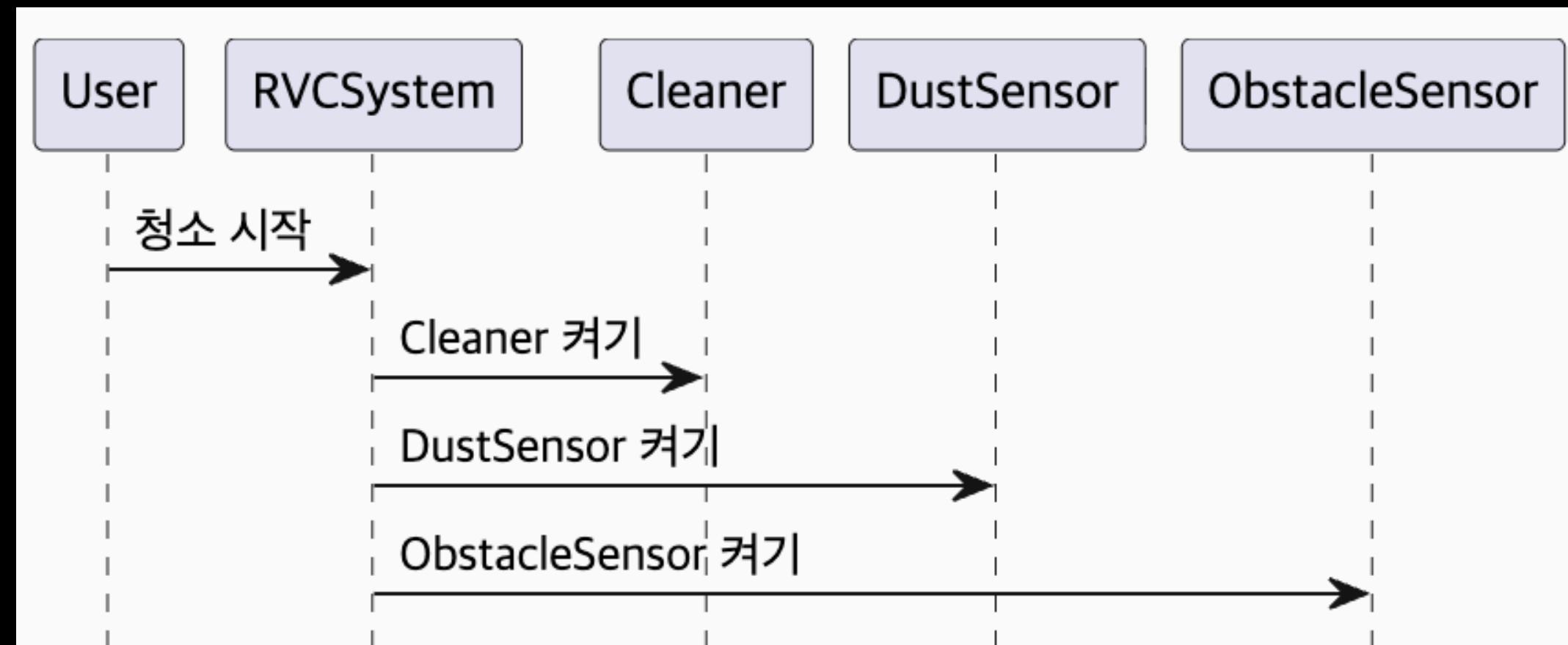
# Use Case #3

Update

<b>Name</b>	3. 청소 시작
<b>Actor</b>	User(Primary), SweepingUnit(Supporting), ObstacleSensor(Supporting), DustSensor(Supporting)
<b>Pre-Requisites</b>	RVC의 전원이 켜져있다.
<b>Typical Courses of Events</b>	(R): RVCSysyem, (U): User 1. (U)가 (R)에게 청소 시작을 지시한다. 2. Ref : UC#7을 시작한다. 3. Ref : UC#6을 시작한다. 4. Ref : UC#5를 시작한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #3

Before



+ 청소 시작

+ Cleaner 켜기

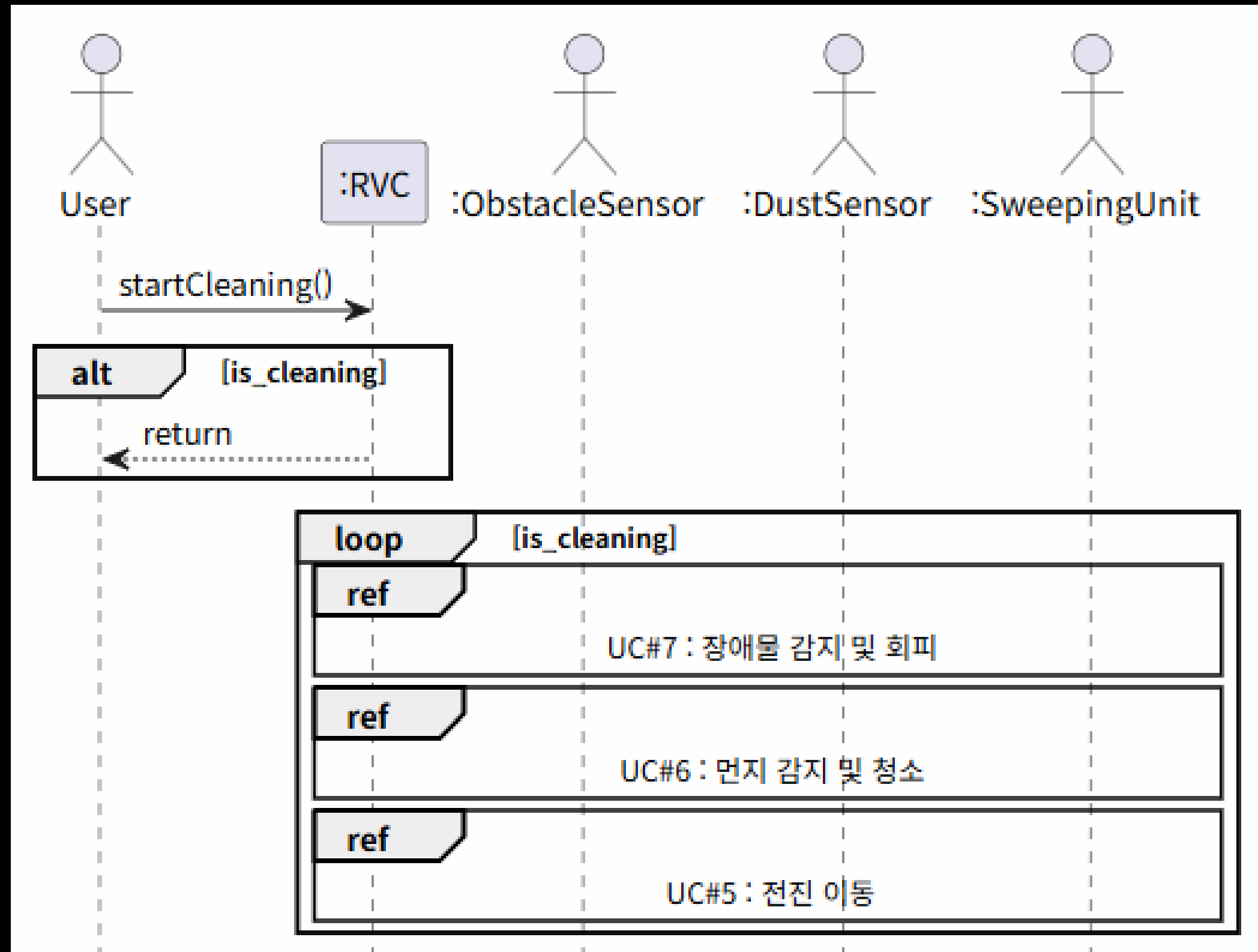
+ DustSensor 켜기

+ ObstacleSensor 켜기

System Sequence Diagram

System Operations

## Use Case #3



+ startCleaning()

System Sequence Diagram

System Operations

# Use Case #4

<b>Name</b>	4. 청소 종료
<b>Actor</b>	User(Primary), Cleaner(Supporting), ObstacleSensor(Supporting), DustSensor(Supporting)
<b>Pre-Requisites</b>	RVC가 청소를 진행 중이다.
<b>Typical Courses of Events</b>	(R): RVCSytem, (U): User, (C): Cleaner, (O): ObstacleSensor, (D): DustSensor 1. (U)가 (R)의 청소를 종료하게 지시한다. 2. (R)이 (C)를 끈다. 3. (R)이 (D)를 끈다. 4. (R)이 (O)를 끈다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #4

<b>Name</b>	4. 청소 종료
<b>Actor</b>	User(Primary)
<b>Pre-Requisites</b>	RVC가 청소를 진행 중이다.
<b>Typical Courses of Events</b>	(R): RVCSytem, (U): User 1. (U)가 (R)에게 청소 중지를 지시한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #4

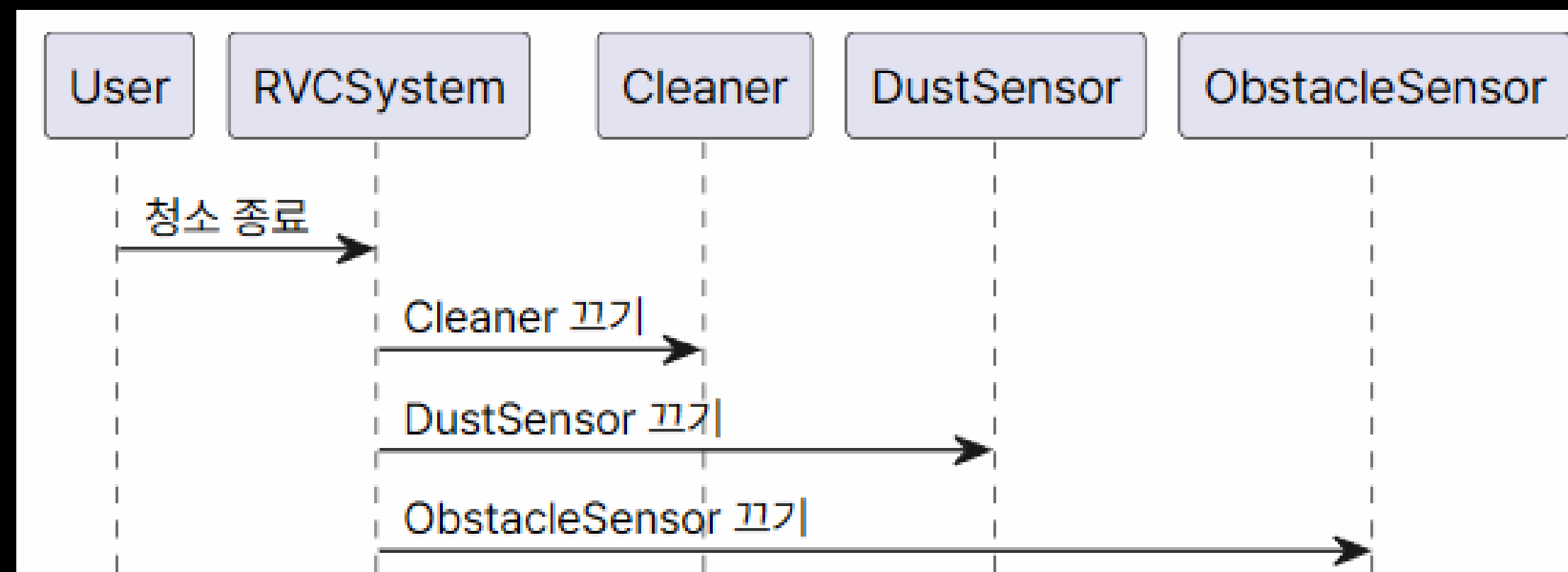
Before

+ 청소 종료

+ Cleaner 끄기

+ DustSensor 끄기

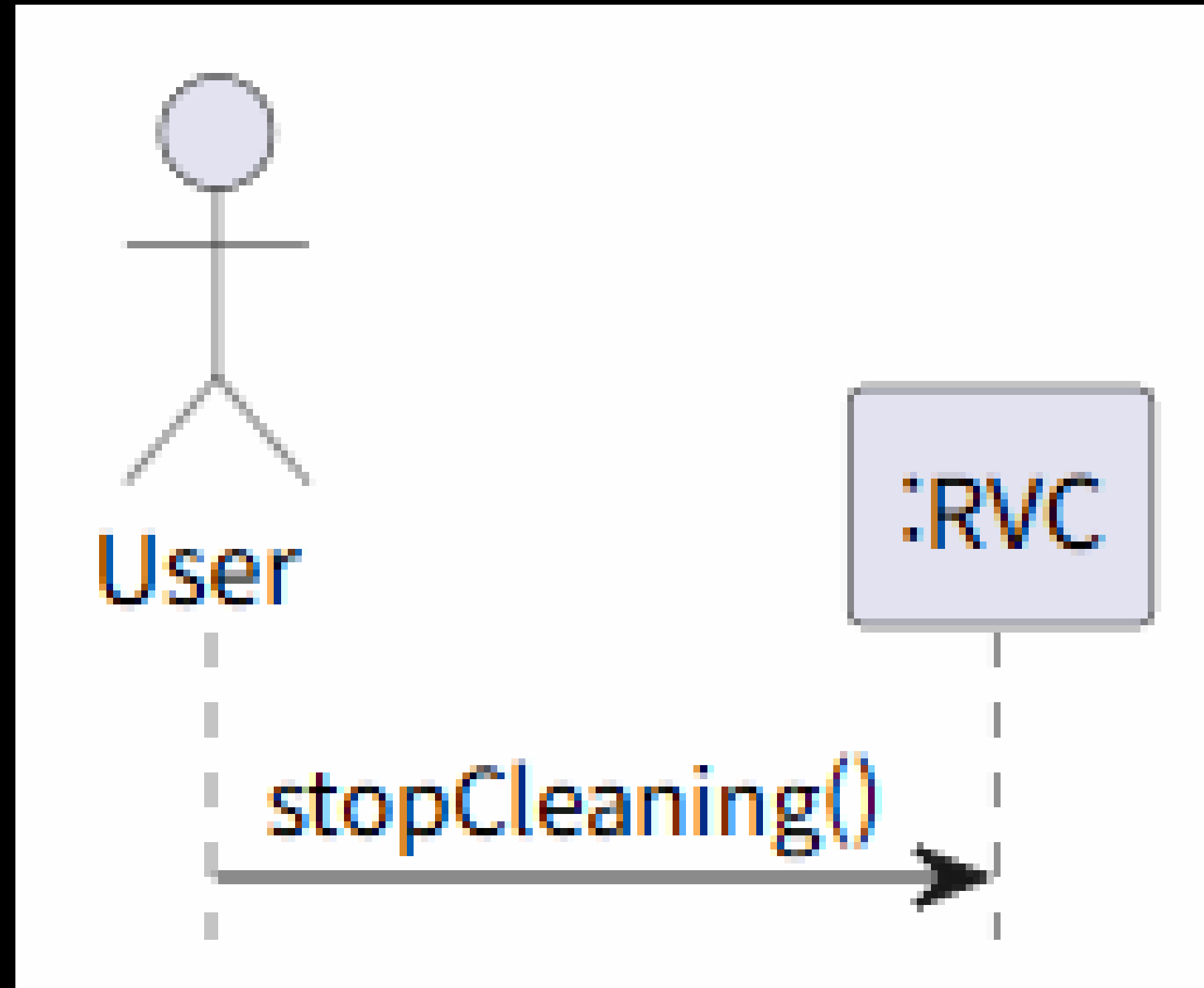
+ ObstacleSensor 끄기



System Sequence Diagram

System Operations

## Use Case #4



+ stopCleaning()

System Sequence Diagram

System Operations

# Use Case #5

Before

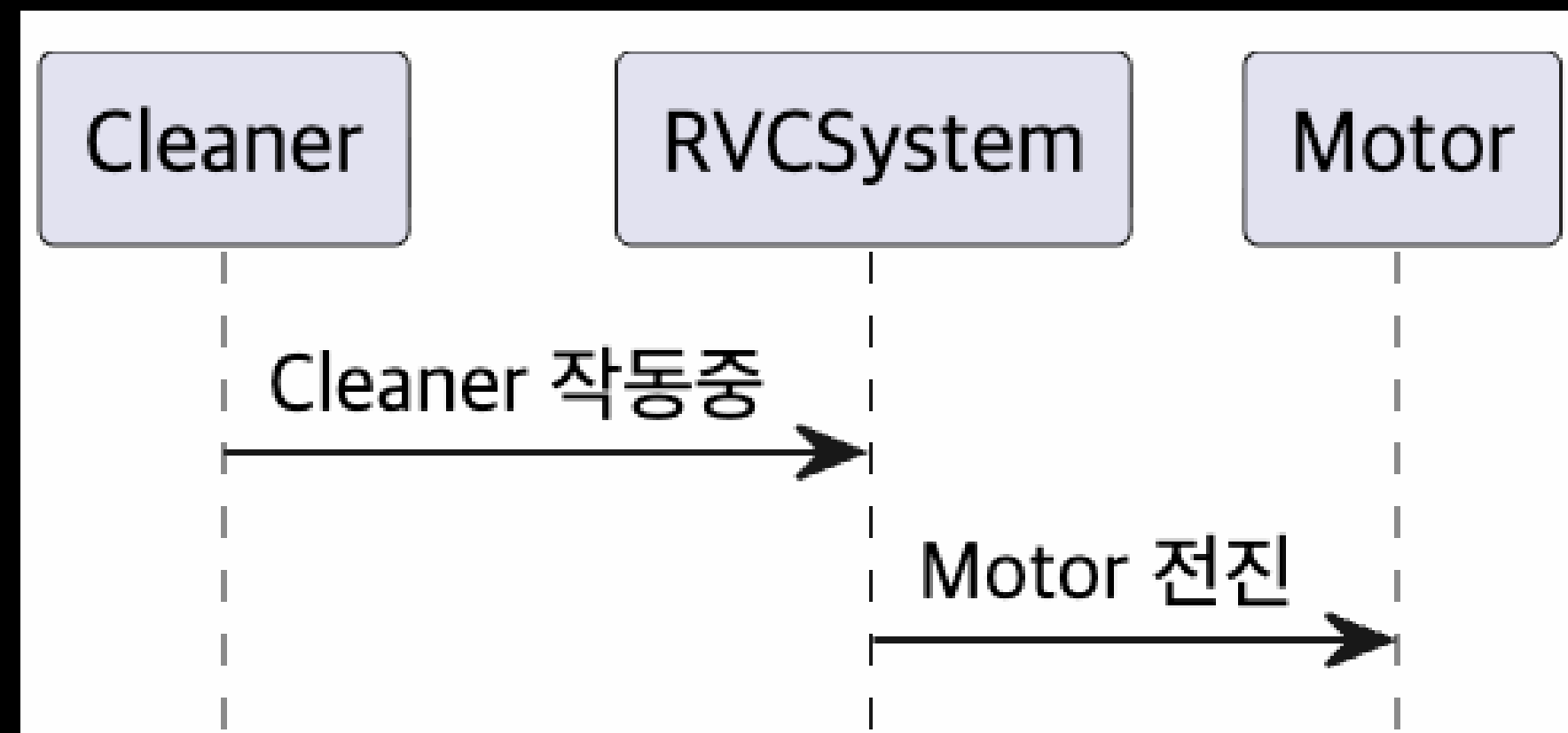
<b>Name</b>	5. Motor는 Cleaner가 켜져 있을 때 전진한다.
<b>Actor</b>	Motor(Supporting), Cleaner(Supporting)
<b>Pre-Requisites</b>	RVC가 켜진 상태이다.
<b>Typical Courses of Events</b>	(C) : Cleaner, (M) : Motor, (R) : RVCSystem 1. (C)가 (R)에게 신호를 보낸다. 2. (R)은 (M)이 전진하도록 한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #5

<b>Name</b>	5. 전진 이동
<b>Actor</b>	DriveMotor(Supporting), SweepingUnit(Supporting)
<b>Pre-Requisites</b>	RVC가 청소를 진행 중이다.
<b>Typical Courses of Events</b>	(R): RVCSysyem, (S): SweepingUnit, (M): DriveMotor 1. (R)이 (S)에게 켜져 있는지 물어본다. 2. (S)가 켜져 있는 것을 (R)이 인식하면, (R)은 (M)을 전진하도록 한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

# Use Case #5

Before



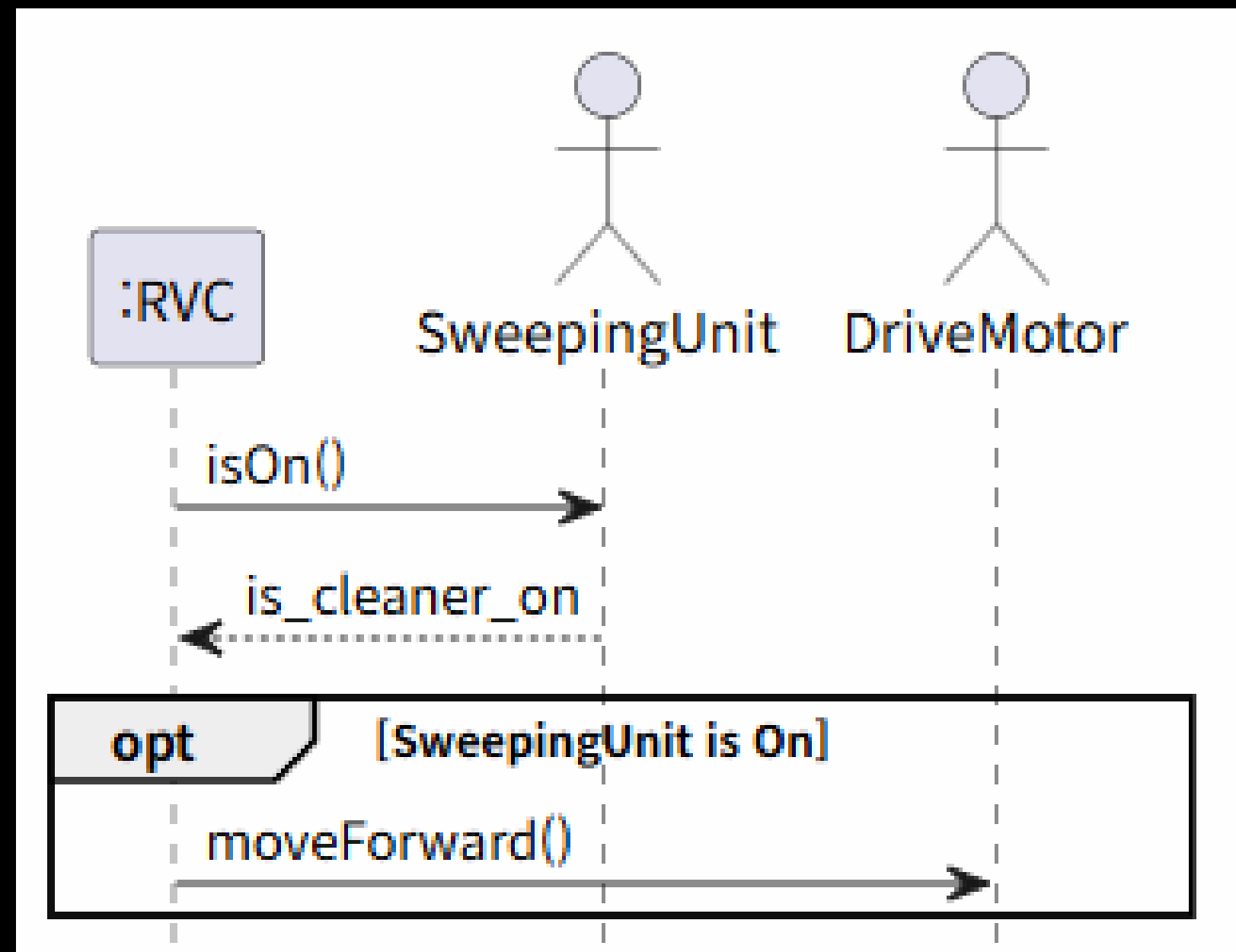
+ Cleaner 작동중

+ Motor 전진

System Sequence Diagram

System Operations

## Use Case #5



System Sequence Diagram

System Operations

# Use Case #6

Before

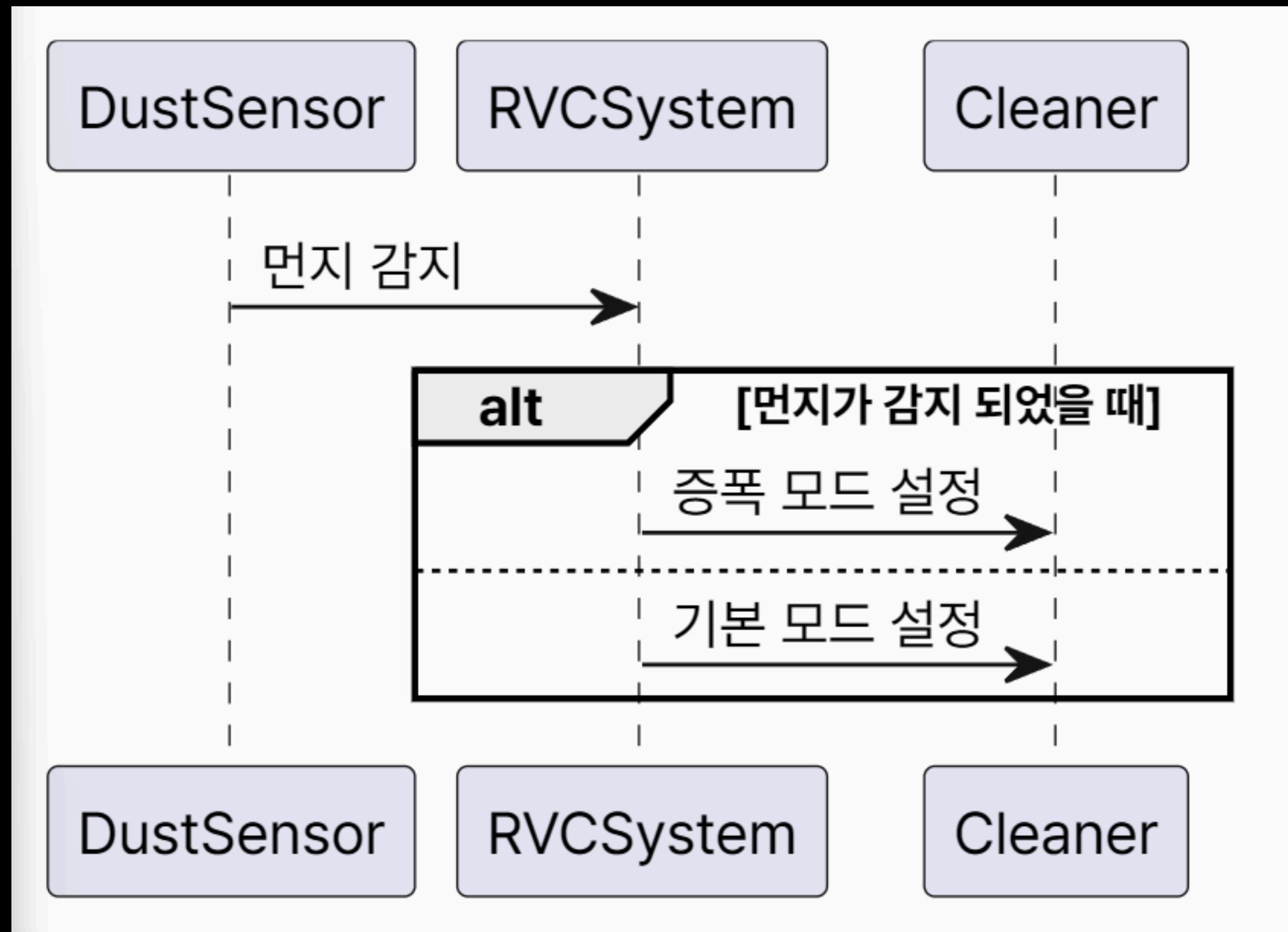
<b>Name</b>	6. 먼지 감지 시 청소기 파워 증폭하기
<b>Actor</b>	DustSensor(Supporting), Cleaner(Supporting)
<b>Pre-Requisites</b>	RVC가 청소 중이다.
<b>Typical Courses of Events</b>	(R): RVCSysyem, (D): DustSensor, (C):Cleaner 1. (D)가 (R)에게 신호를 보낸다. 2. 먼지가 감지되었다면 (R)이 (C)의 파워를 증폭모드로 설정한다.
<b>Alternative Courses of Events</b>	Line 2: 먼지가 감지되지 않았다면 (R)이 (C)의 파워를 기본 모드로 설정한다.
<b>Exceptional Courses of Events</b>	N/A

# Use Case #6

<b>Name</b>	6. 먼지 감지 및 청소
<b>Actor</b>	DustSensor(Supporting), SweepingUnit (Supporting)
<b>Pre-Requisites</b>	RVC가 청소를 진행 중이다.
<b>Typical Courses of Events</b>	(R): RVCSysyem, (D): DustSensor, (S): SweepingUnit 1. (R)이 (D)에게 먼지 감지를 요청한다. 2. (D)가 (R)에게 먼지 감지 여부를 알려준다. 3. 먼지가 감지되었다면, (R)이 (S)의 파워를 증폭 모드로 설정한다.
<b>Alternative Courses of Events</b>	Line 3: 먼지가 감지되지 않았다면, (R)이 (S)의 파워를 일반 모드로 설정한다.
<b>Exceptional Courses of Events</b>	N/A

# Use Case #6

Before



System Sequence Diagram

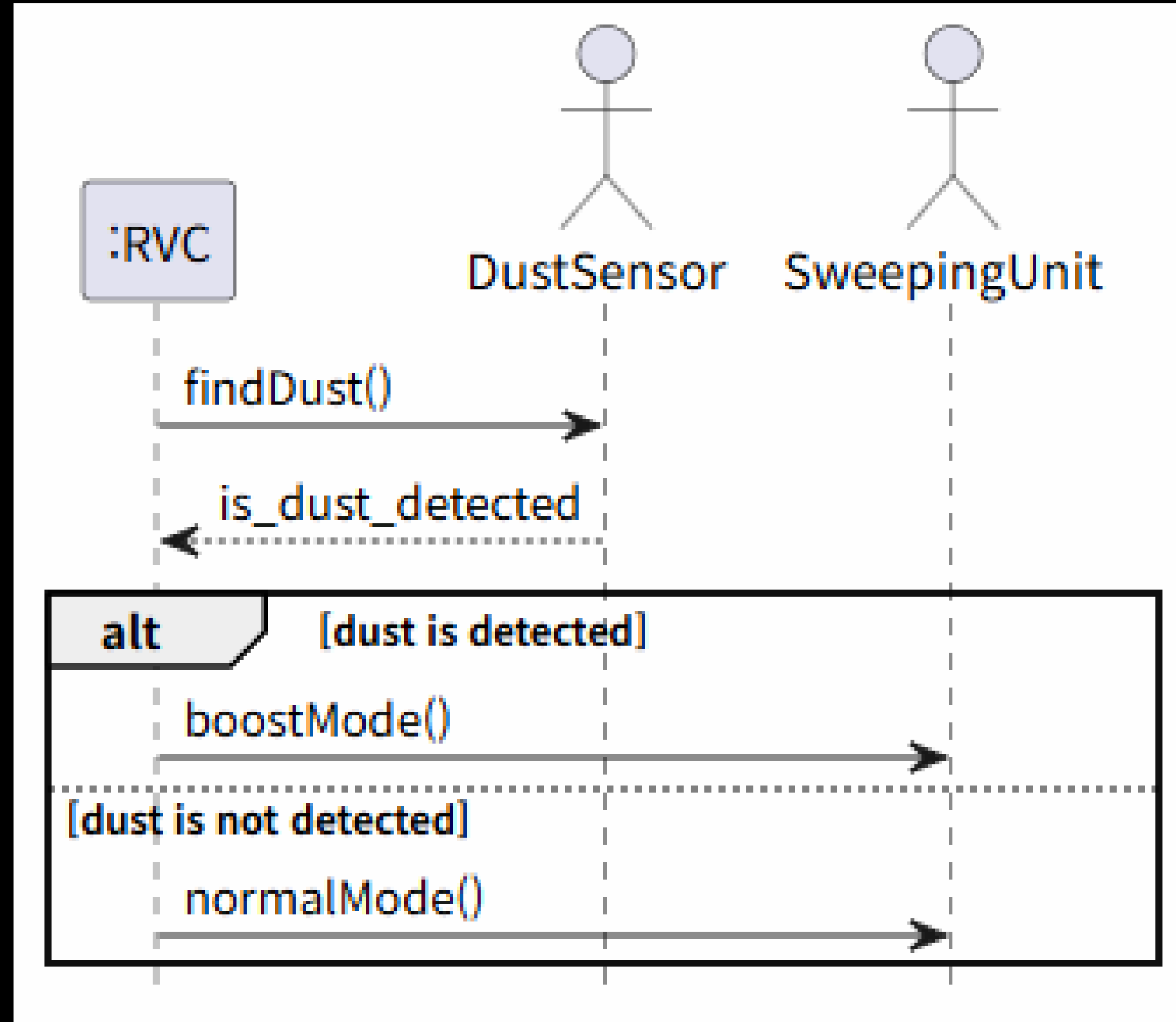
+ 먼지 감지

+ 증폭 모드

+ 기본 모드

System Operations

## Use Case #6



System Sequence Diagram

System Operations

# Use Case #7

Before

<b>Name</b>	7. ObstacleSensor가 장애물을 감지하면 회피한다
<b>Actor</b>	ObstacleSensor(Supporting), Cleaner(Supporting), Motor(Motor)
<b>Pre-Requisites</b>	RVC가 청소 중이다.
<b>Typical Courses of Events</b>	(C) : Cleaner (M): Motor, (O): ObstacleSensor, (R) : RVCSystem 1. (O)가 장애물을 감지해 (R)에게 알린다. 2. (R)이 (C)를 끈다. 3. (R)이 (M)의 이동을 중단하도록 한다. 4. 좌측 또는 우측에 장애물이 없다면, (R)은 (M)이 장애물이 없는 방향으로 회전하도록 한다. 5. (R)이 (C)를 켜다.
<b>Alternative Courses of Events</b>	Line 4: 좌우측에 장애물이 있다면, (R)은 (M)이 후진 후에 좌측 또는 우측으로 회전하도록 한다.
<b>Exceptional Courses of Events</b>	Line 4~5: 사방에 장애물이 있다면, (O)를 끄고 청소를 종료한다.

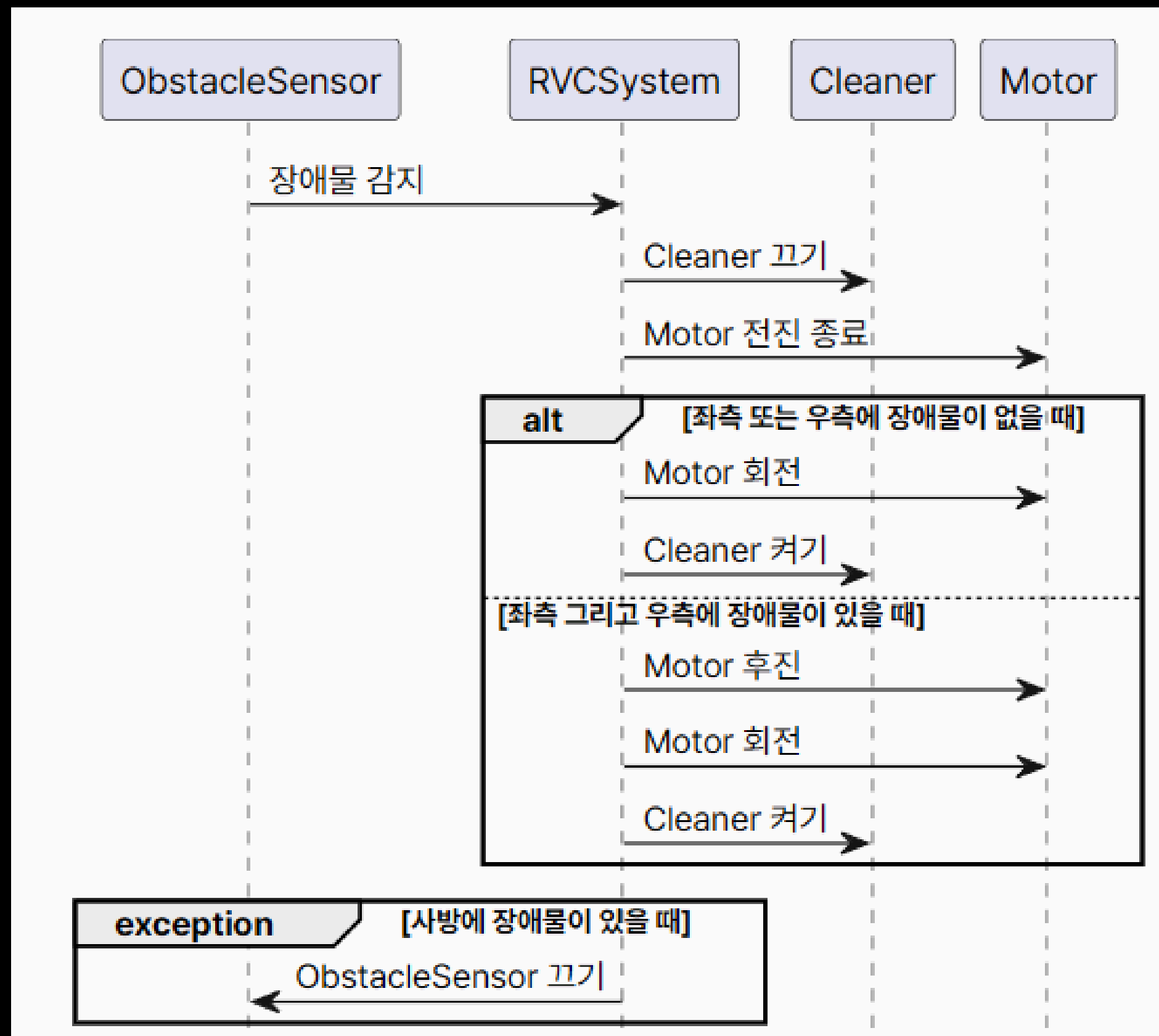
# Use Case #7

Update

<b>Name</b>	7. 장애물 감지 및 회피
<b>Actor</b>	ObstacleSensor(Supporting), SweepingUnit(Supporting), DriveMotor(Motor)
<b>Pre-Requisites</b>	RVC가 청소 중이다.
<b>Typical Courses of Events</b>	(R): RVCSysyem, (S): SweepingUnit, (M): DriveMotor, (O): ObstacleSensor, (D): DustSensor 1. (R)이 (O)에게 장애물 감지를 요청한다. 2. (O)가 (R)에게 장애물 정보를 알려준다. 3. 정면에서 장애물이 감지되었다면, (R)이 (S)를 끈다. 4. (R)이 (M)의 이동을 중단한다. 5. 좌측에 장애물이 없다면, (R)은 (M)이 좌측으로 회전하도록 한다. 6. (R)이 (S)를 켜다.
<b>Alternative Courses of Events</b>	Line 5: 좌측에 장애물이 있고 우측에 장애물이 없다면, (R)은 (M)이 우측으로 회전하도록 한다. Line 5: 좌측과 우측에 모두 장애물이 있다면, (R)은 (M)이 후진하도록 한 뒤 정지시키고 좌측으로 회전하도록 한다.
<b>Exceptional Courses of Events</b>	Line 5~6: 사방에 장애물이 있다면, (R)은 (D)의 전원을 끄고 청소를 종료한다.

# Use Case #7

Before



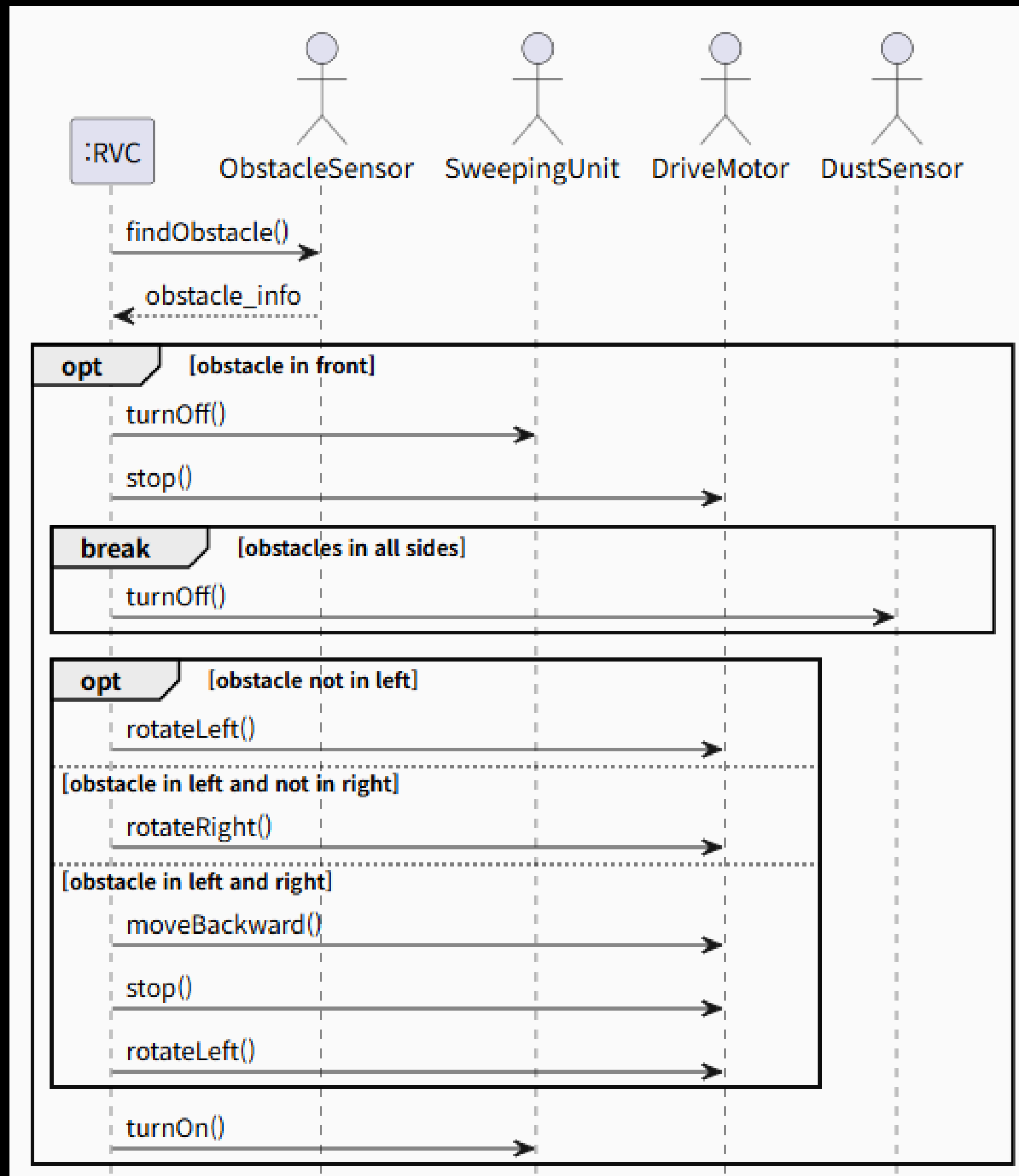
- + 장애물 감지
- + Cleaner 켜기
- + Cleaner 끄기
- + Motor 전진 종료
- + Motor 회전
- + Motor 후진
- + ObstacleSensor 끄기

System Sequence Diagram

System Operations

# Use Case #7

Update

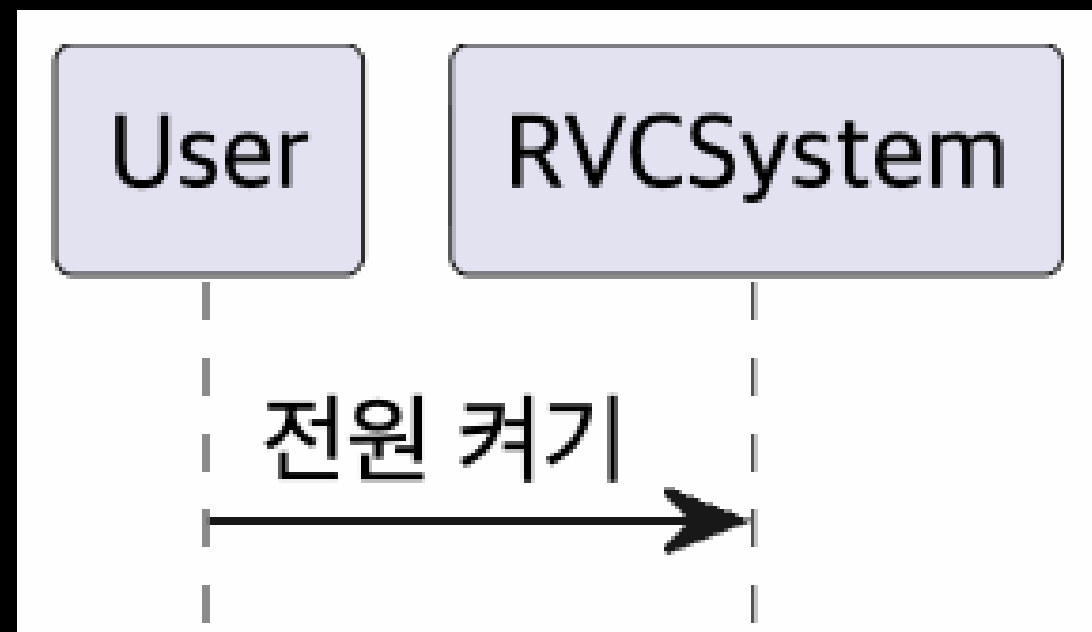


System Sequence Diagram

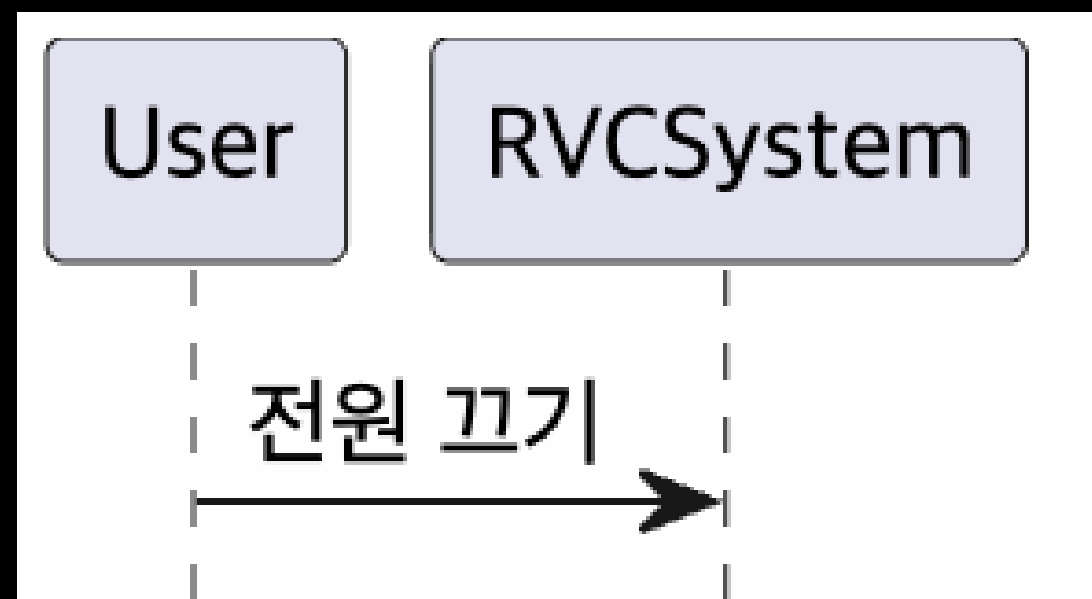
System Operations

Before

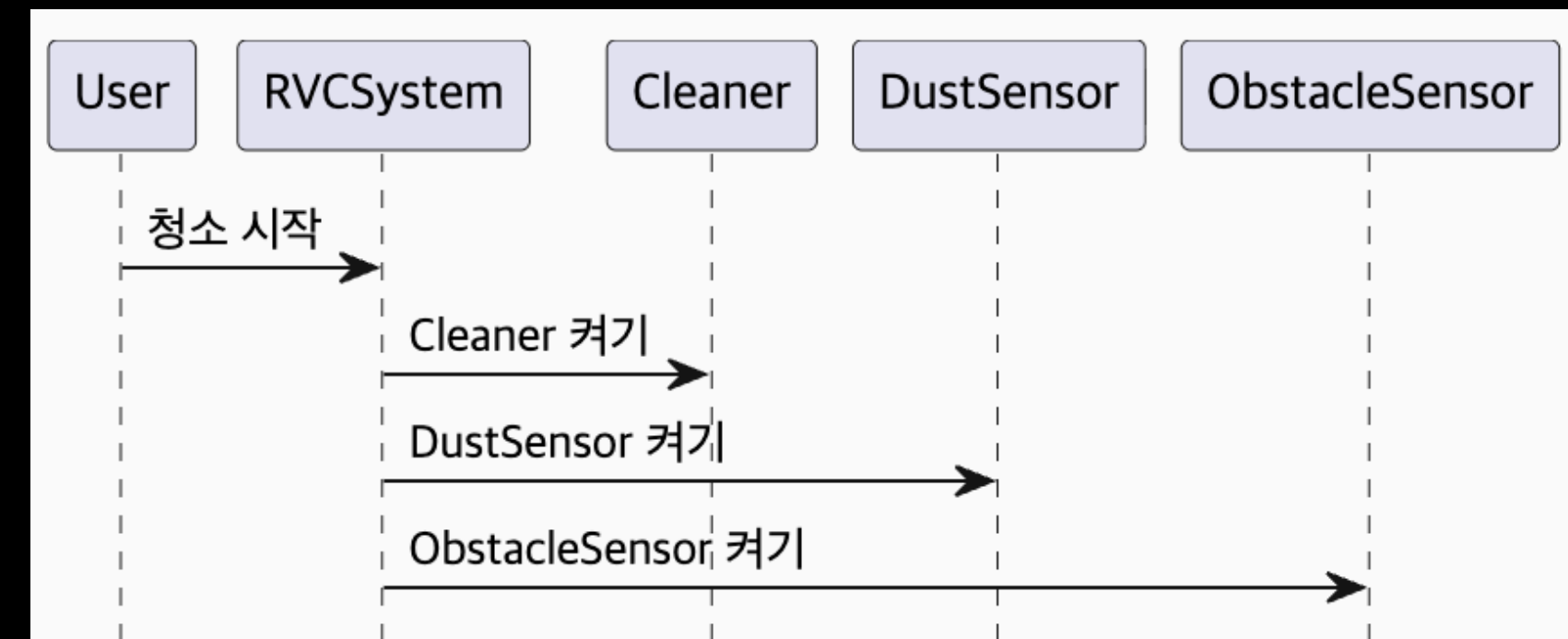
# System Sequence Diagram



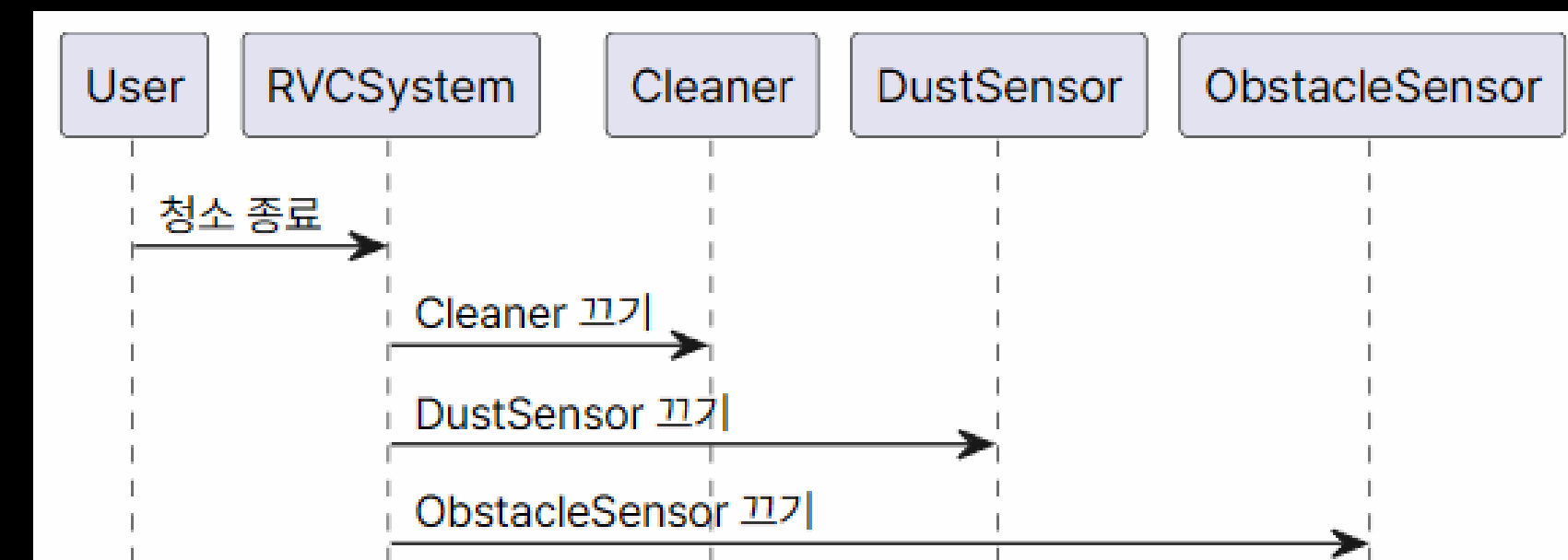
UC #1



UC #2



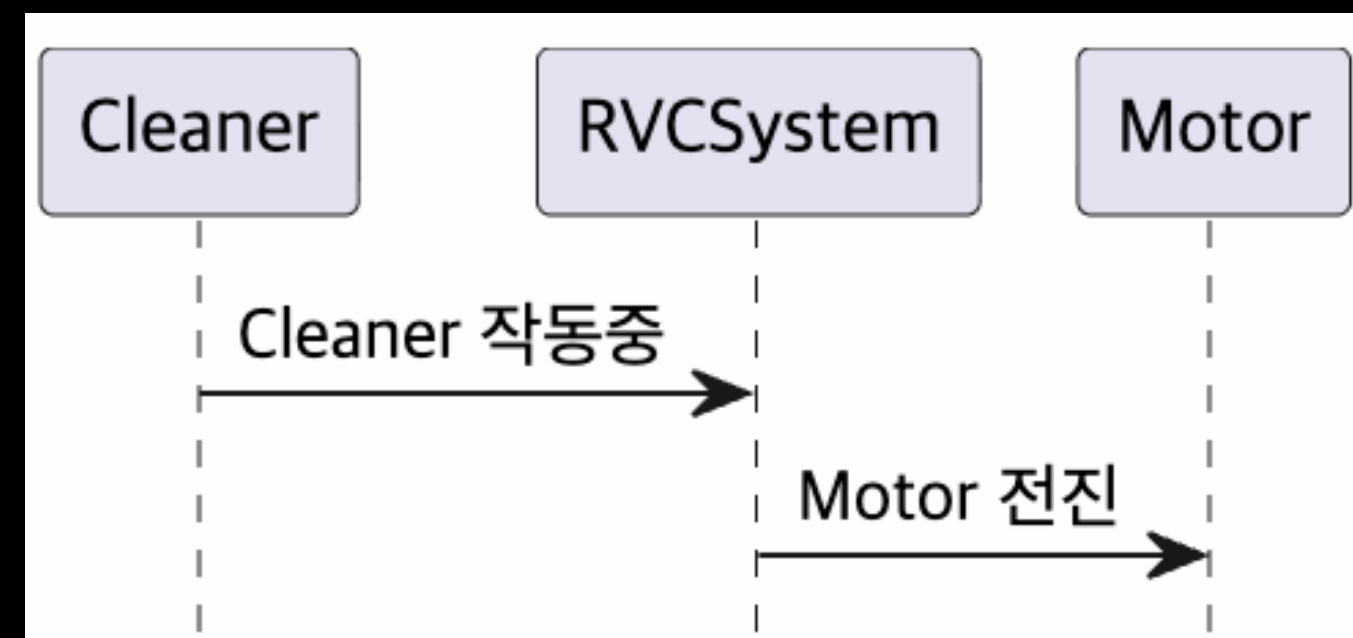
UC #3



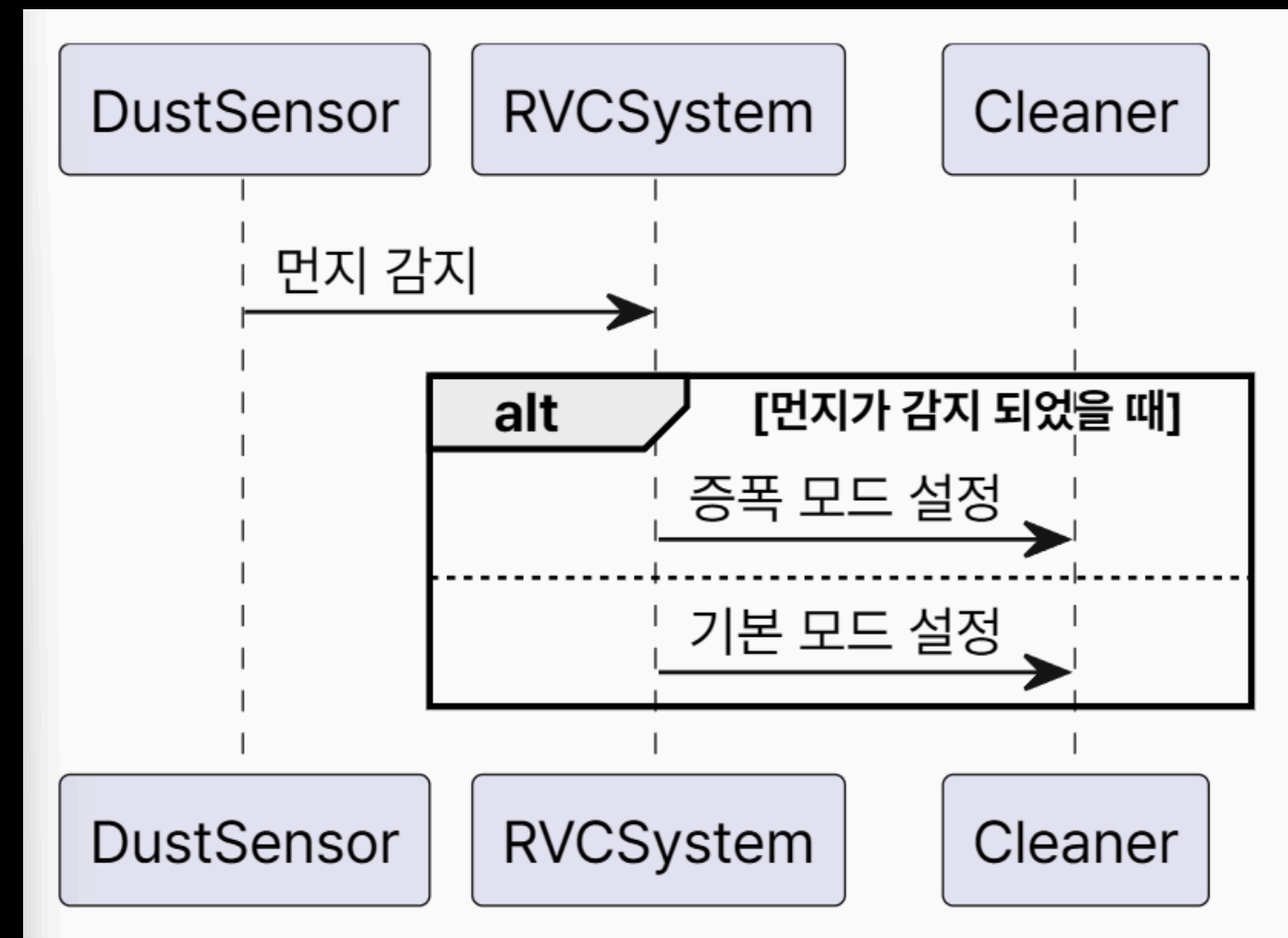
UC #4

Before

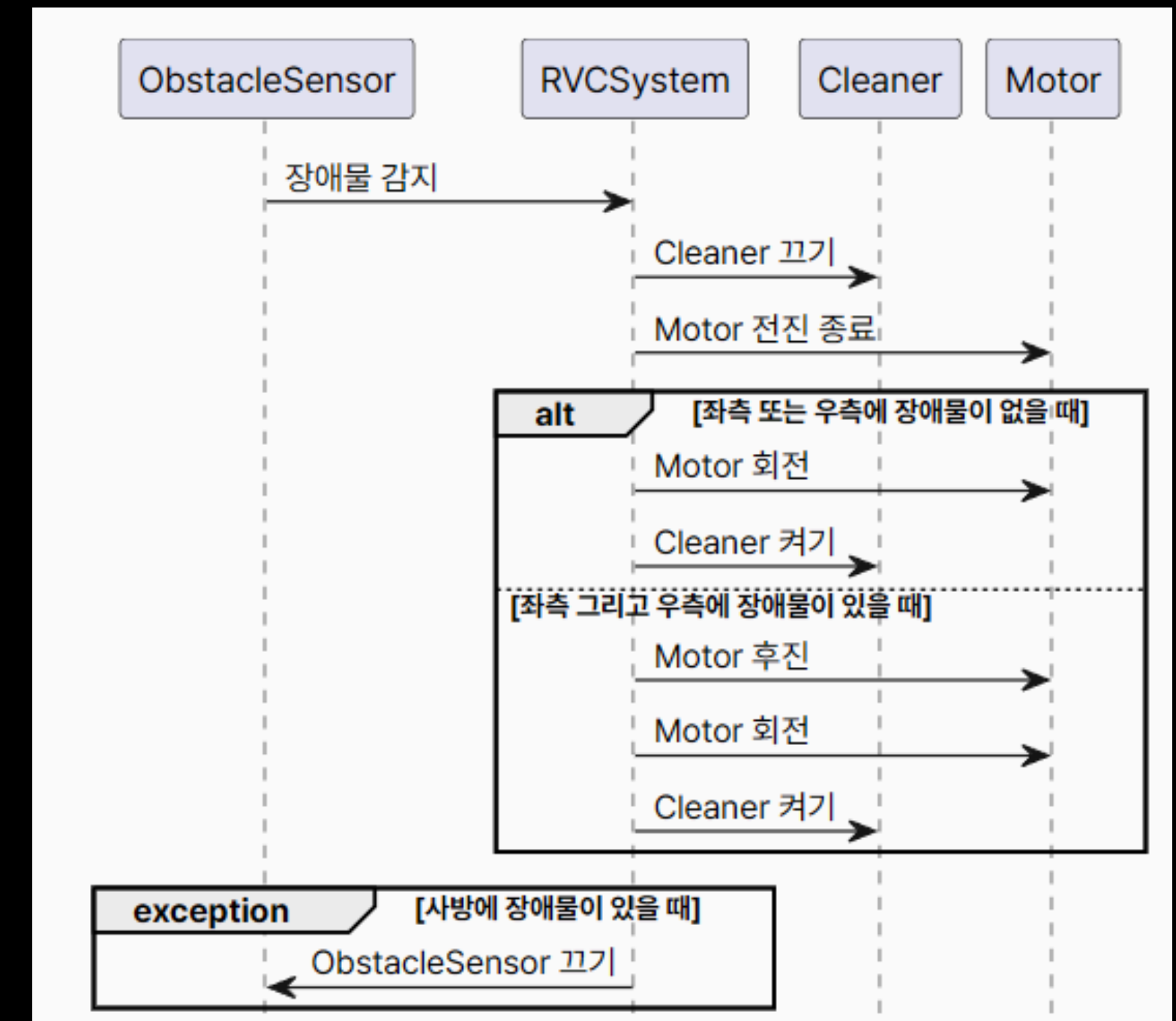
# System Sequence Diagram



UC #5

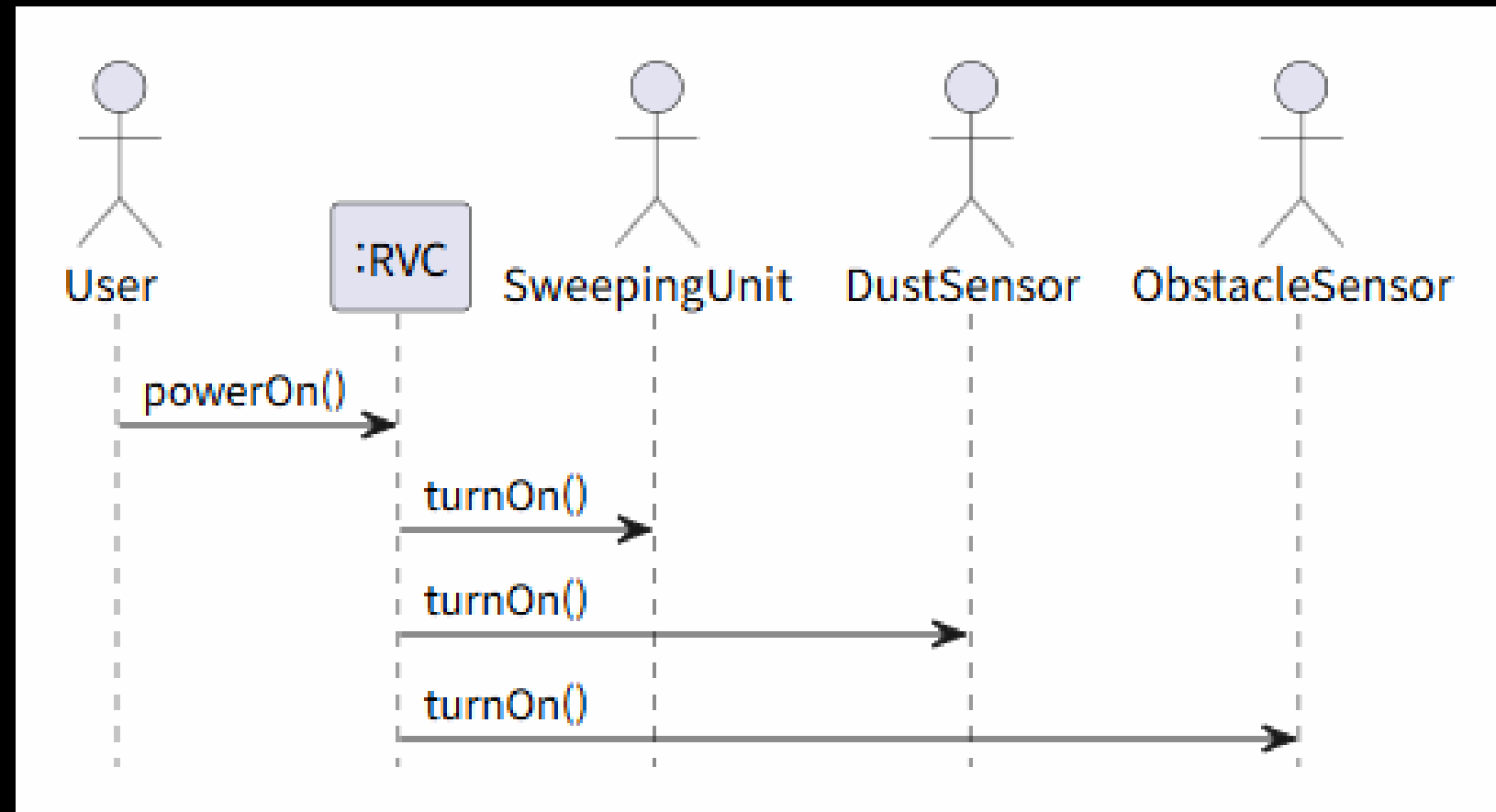


UC #6

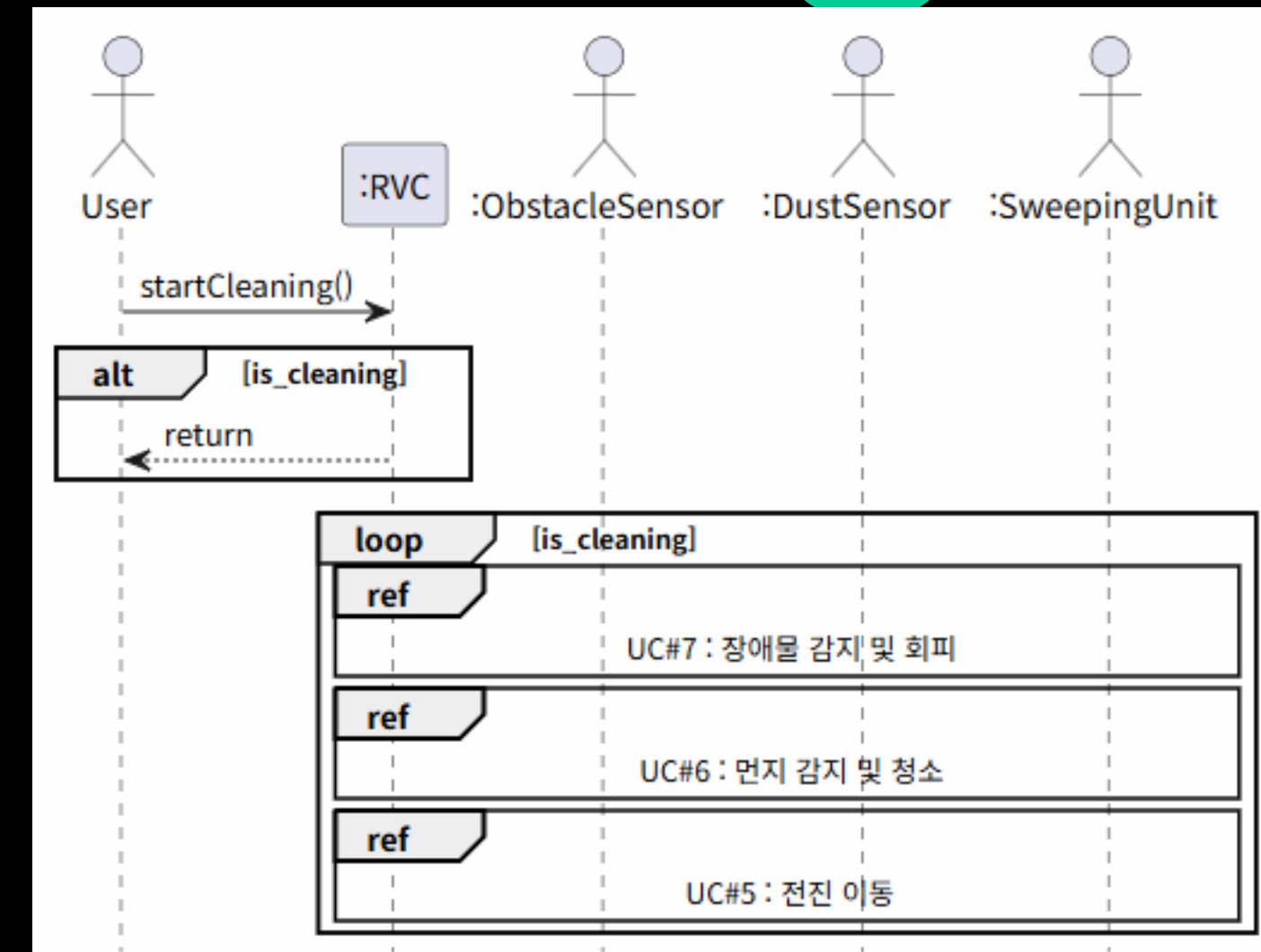


UC #7

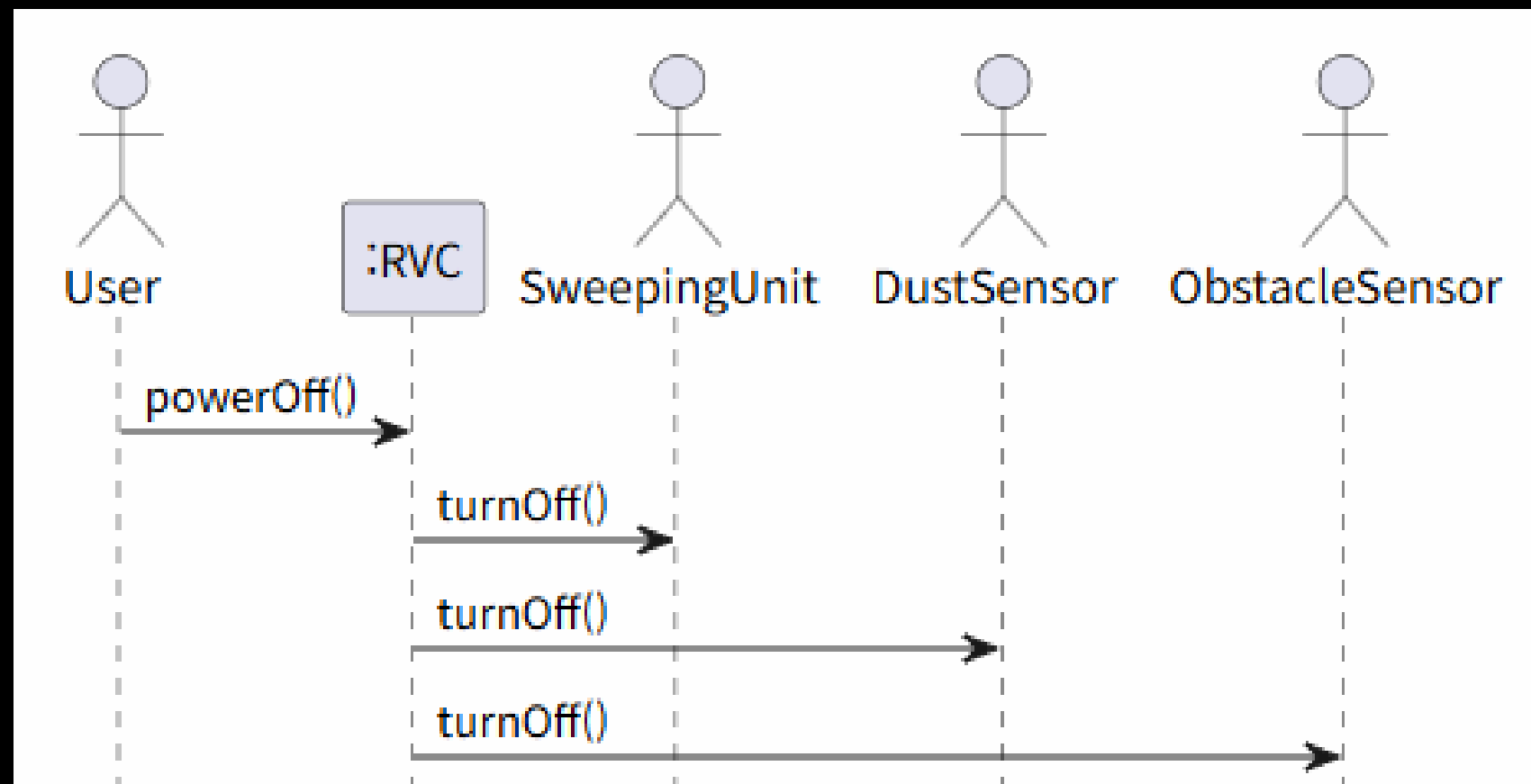
# System Sequence Diagram Update



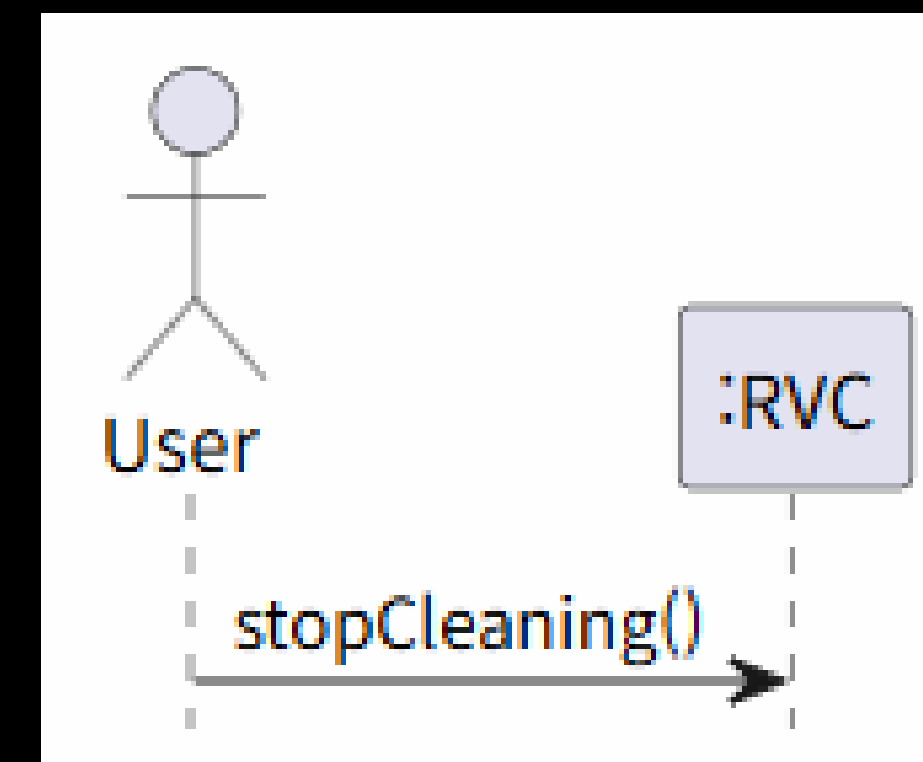
UC #1



UC #3

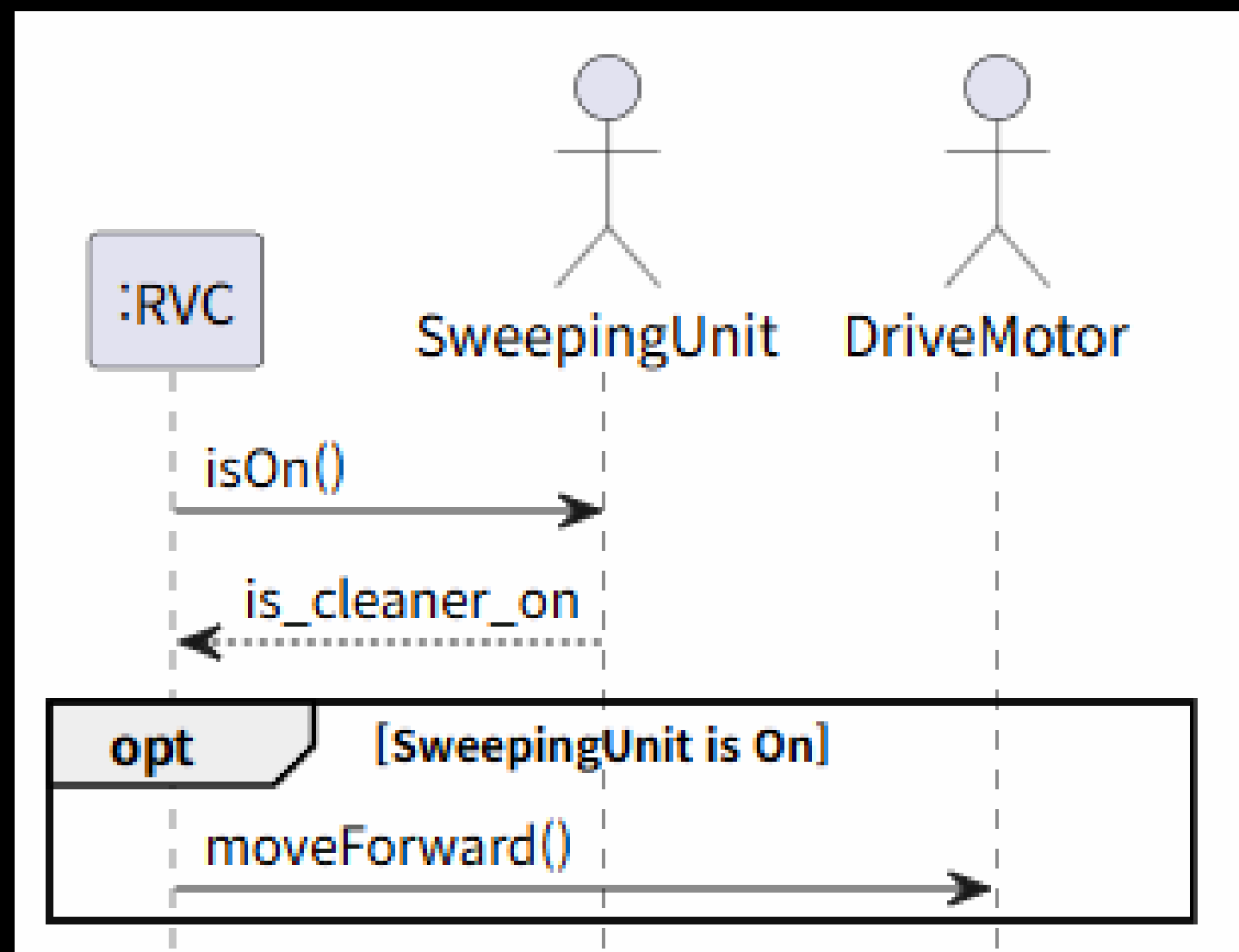


UC #2

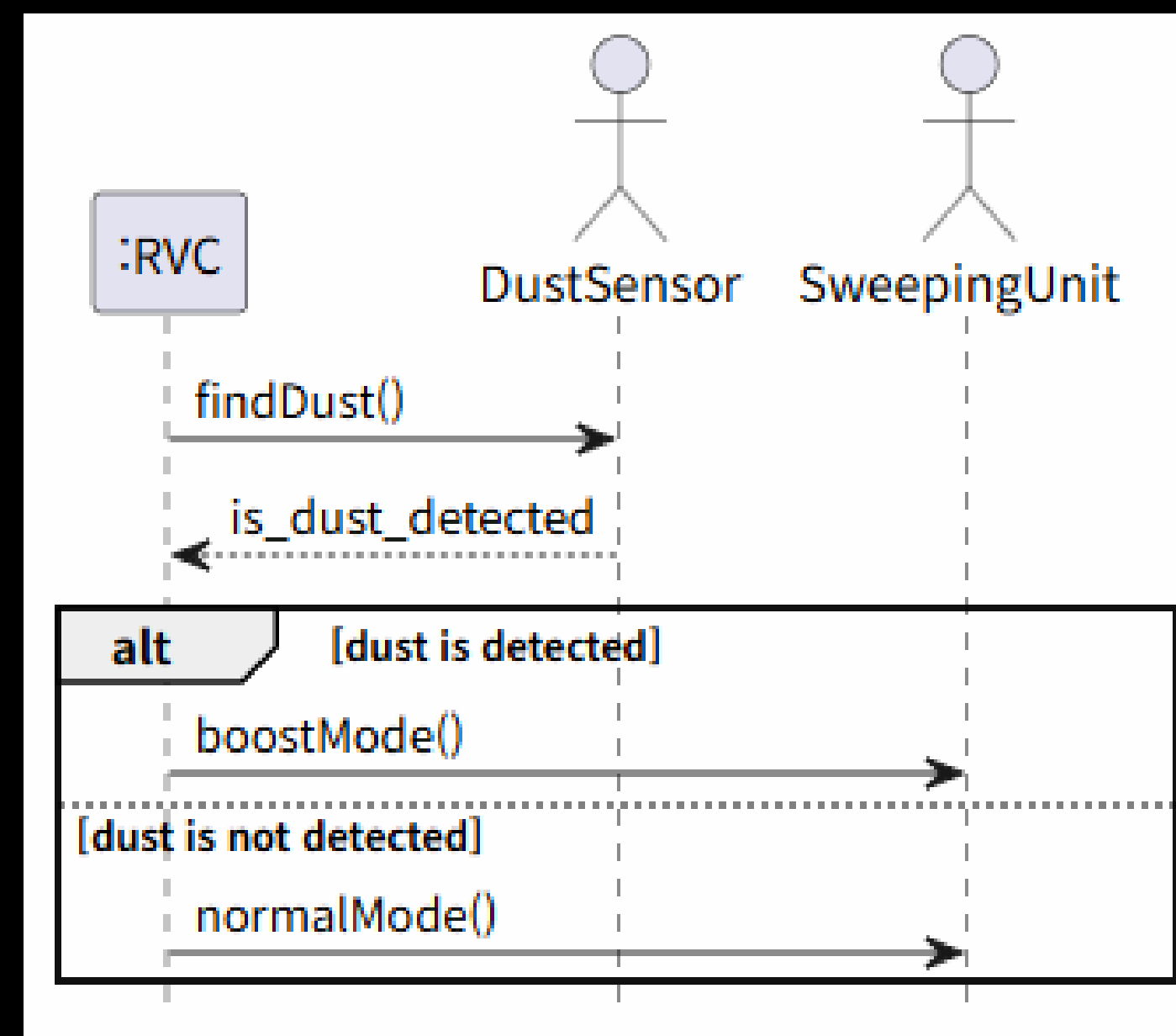


UC #4

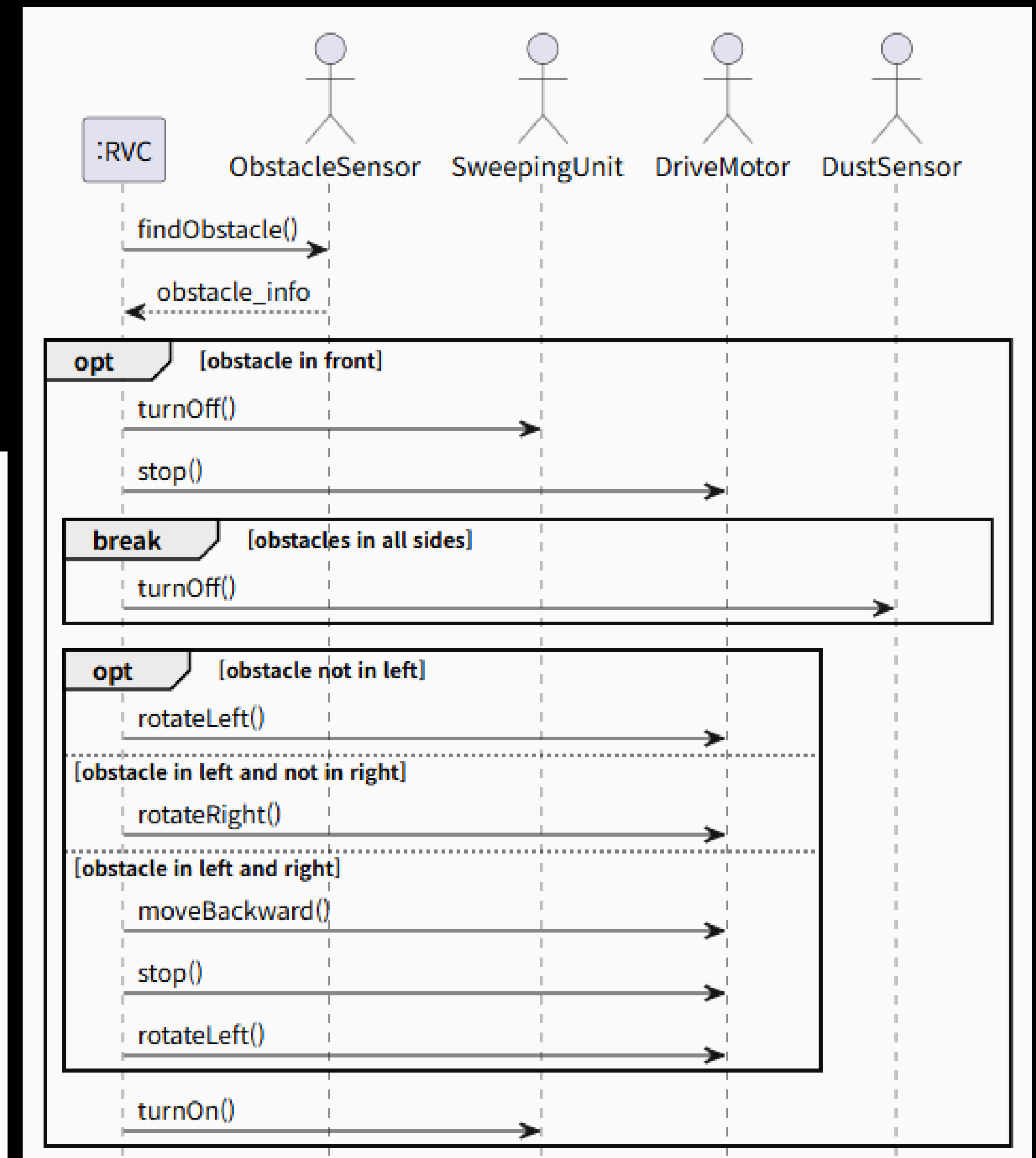
# System Sequence Diagram



UC #5



UC #6



UC #7

# System Operations

Before

+ 전원 켜기

+ 전원 끄기

+ 청소 시작

+ Cleaner 켜기

+ DustSensor 켜기

+ ObstacleSensor 켜기

+ 청소 종료

+ Cleaner 끄기

+ DustSensor 끄기

+ ObstacleSensor 끄기

+ Cleaner 작동중

+ 전진

+ 먼지 감지

+ 증폭 모드

+ 기본 모드

+ 장애물 감지

+ Cleaner 켜기

+ Cleaner 끄기

+ Motor 전진 종료

+ Motor 회전

+ Motor 후진

+ ObstacleSensor 끄기

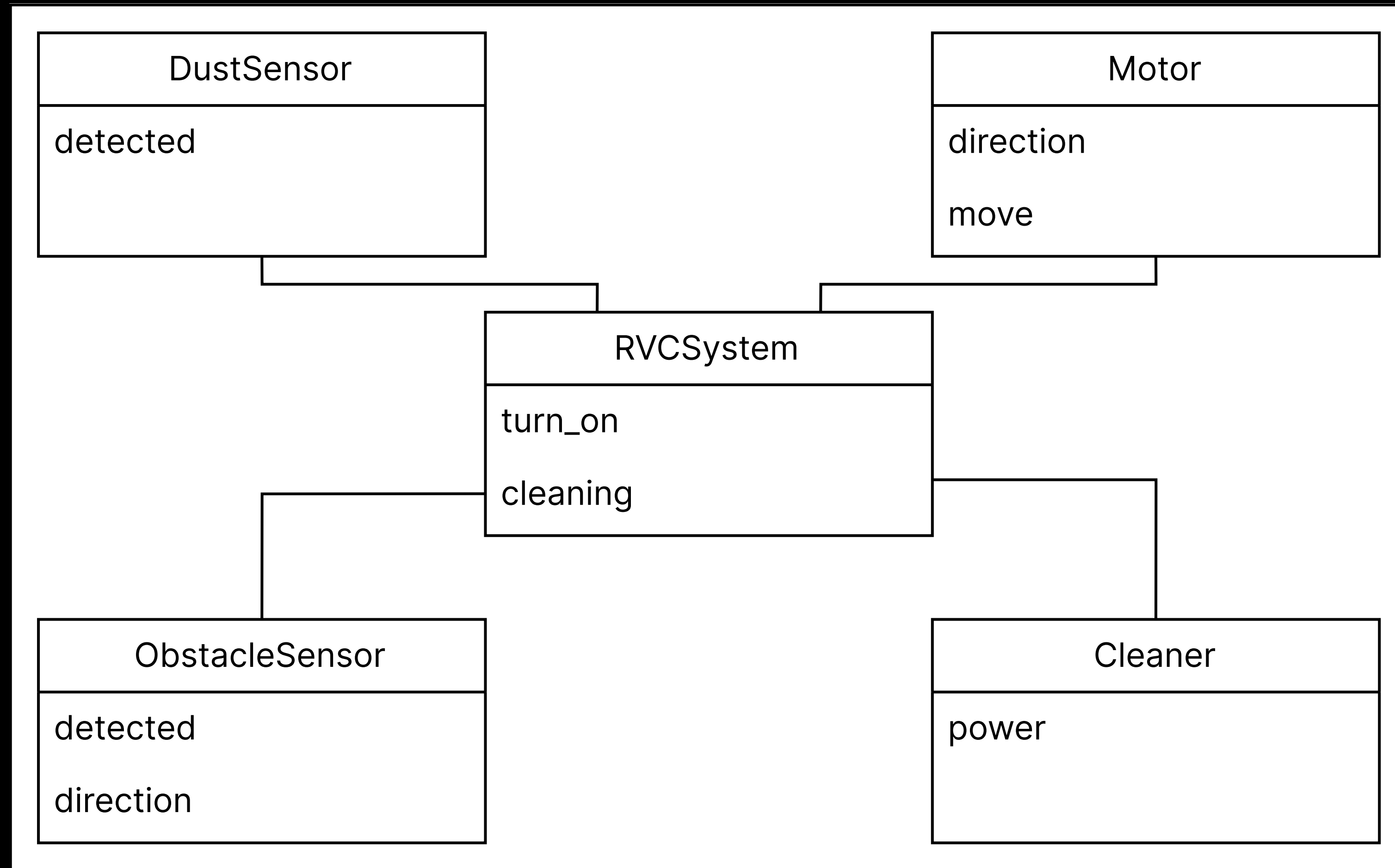
# System Operations

Update

{interface}  
RVCSystem

+ powerOn()  
+ powerOff()  
+ startCleaning()  
+ stopCleaning()

# Domain Model



**Thank**

**you!**

