TEAM 2

Object Oriented Implementation

강병완 202211248
강현준 202211251
박　완 202211301
정민수 202211365

Table of Content

Update

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(Supportin

Use Case #1

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

Use Case #1

Update

Name	1. 전원 켜기
Actor	User (Primary), SweepingUnit(Supporting), DustSensor(Supporting), ObstacleSensor(Supporting)
Pre-Requisites	RVC의 전원이 꺼져있는 상태이다.
Typical Courses of Events	(R): RVCSystem, (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.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case #2

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

Use Case #2

Update

Name	2. 전원 끄기
Actor	User (Primary), SweepingUnit(Supporting), DustSensor(Supporting), ObstacleSensor(Supporting)
Pre-Requisites	RVC의 전원이 켜져있는 상태이다.
Typical Courses of Events	(R): RVCSystem, (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.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case #3

Name	3. 청소 시작
Actor	User(Primary), Cleaner(Supporting), ObstacleSensor(Supporting), DustSensor(Supporting)
Pre-Requisites	RVC의 전원이 켜져있다.
Typical Courses of Events	(R) : RVCSystem, (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.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case #3

Update

Name	3. 청소 시작
Actor	User(Primary), SweepingUnit(Supporting), ObstacleSensor(Supporting), DustSensor(Supporting)
Pre-Requisites	RVC의 전원이 켜져있다.
Typical Courses of Events	(R): RVCSystem, (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.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case #4

Name	4. 청소 종료
Actor	User(Primary), Cleaner(Supporting), ObstacleSensor(Supporting), DustSensor(Supporting)
Pre-Requisites	RVC가 청소를 진행 중이다.
Typical Courses of Events	(R): RVCSystem, (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.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case #4

Update

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

Use Case #5

Name	5. 청소 시 전진
Actor	Motor(Supporting), Cleaner(Supporting)
Pre-Requisites	RVC가 켜진 상태이다.
Typical Courses of Events	(C) : Cleaner, (M) : Motor, (R) : RVCSystem 1. (R)이 (C)에게 켜져있냐고 물어본다. 2. (C)가 켜져있는 걸 (R)이 인식하면, (R)은 (M)을 전진하도록 한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case #5

Update

Name	5. 전진 이동
Actor	DriveMotor(Supporting), SweepingUnit(Supporting)
Pre-Requisites	RVC가 청소를 진행 중이다.
Typical Courses of Events	(R): RVCSystem, (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.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use Case #6

Name	6. 청소 파워 증폭
Actor	DustSensor(Supporting), Cleaner(Supporting)
Pre-Requisites	RVC가 청소 중이고, Cleaner가 켜져있다.
Typical Courses of Events	(R): RVCSystem, (D): DustSensor, (C):Cleaner 1. (R)이 (D)에게 먼지 감지를 요청한다. 2. 먼지가 감지되었다면 (R)이 (C)의 파워를 증폭모드로 설정한다.
Alternative Courses of Events	Line 2: 먼지가 감지되지 않았다면 (R)이 (C)의 파워를 기본 모드로 설정한다.
Exceptional Courses of Events	N/A

Use Case #6

Update

Name	6. 먼지 감지 및 청소
Actor	DustSensor(Supporting), SweepingUnit (Supporting)
Pre-Requisites	RVC가 청소를 진행 중이다.
Typical Courses of Events	(R): RVCSystem, (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.
Alternative Courses of Events	Line 3: 먼지가 감지되지 않았다면, (R)이 (S)의 파워를 일반 모드로 설정한다.
Exceptional Courses of Events	N/A

Use Case #7

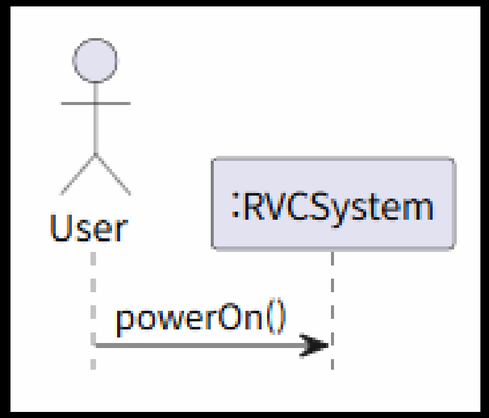
Name	7. 장애물 감지 후 회피
Actor	ObstacleSensor(Supporting), Cleaner(Supporting), Motor(Motor)
Pre-Requisites	RVC가 청소 중이다.
Typical Courses of Events	(C) : Cleaner (M): Motor, (O): ObstacleSensor, (R) : RVCSystem 1. (R)이 (O)에게 장애물 감지를 요청한다. 2. 정면에서 장애물을 감지했다면, (R)이 (C)를 끈다. 3. (R)이 (M)의 이동을 중단한다. 4. 좌측에 장애물이 없다면, (R)은 (M)이 좌측으로 회전하도록 한다. 5. (R)이 (C)를 켠다.
Alternative Courses of Events	Line 4 : 좌측에 장애물이 있고 우측에 장애물이 없다면 (R)은 (M)이 우측으로 회전하도록 한다. Line 4: 좌측 그리고 우측에 장애물이 있다면, (R)은 (M)이 후진 후에 좌측 또는 우측으로 회전하도록 한다.
Exceptional Courses of Events	Line 4~5 : 사방에 장애물이 있다면, 청소를 종료한다.

Use Case #7

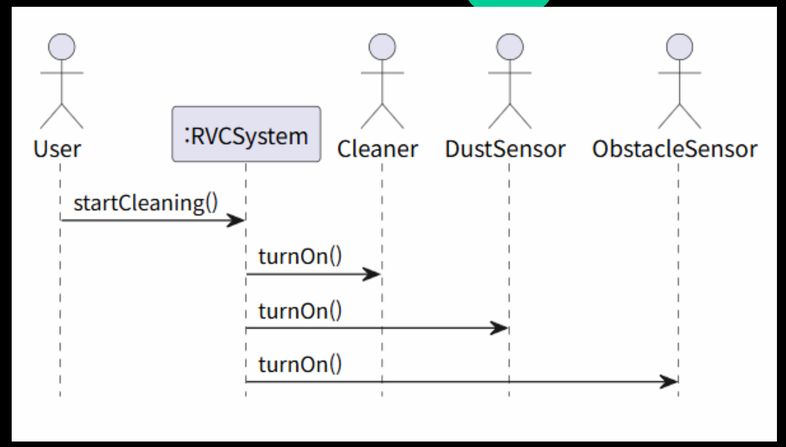
Update

Name	7. 장애물 감지 및 회피
Actor	ObstacleSensor(Supporting), SweepingUnit(Supporting), DriveMotor(Supporting), DustSensor(Supporting)
Pre-Requisites	RVC가 청소 중이다.
Typical Courses of Events	(R): RVCSystem, (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 켠다.
Alternative Courses of Events	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.
Exceptional Courses of Events	Line 5~6: 사방에 장애물이 있다면, (R)은 (D)의 전원을 끄고 청소를 종료한다.

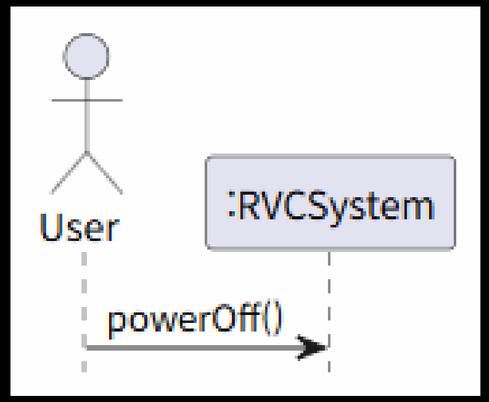
System Sequence Diagram



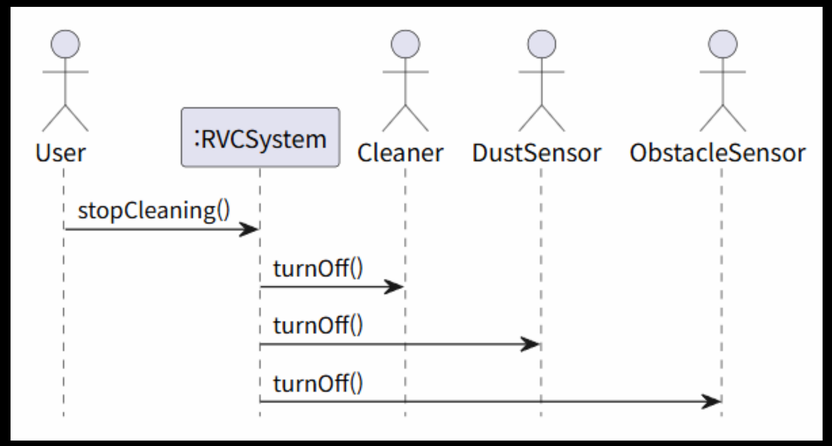
UC #1



UC #3

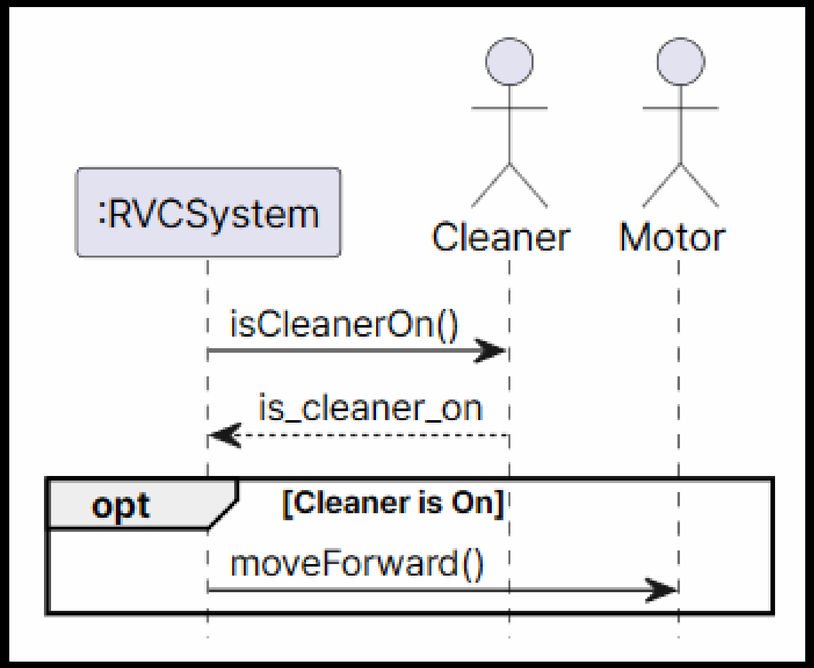


UC #2

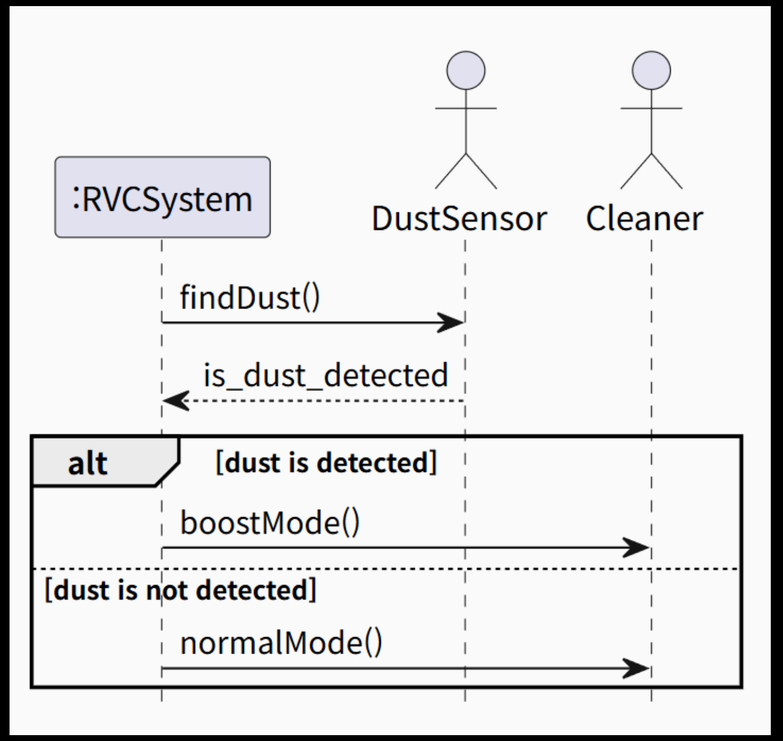


UC #4

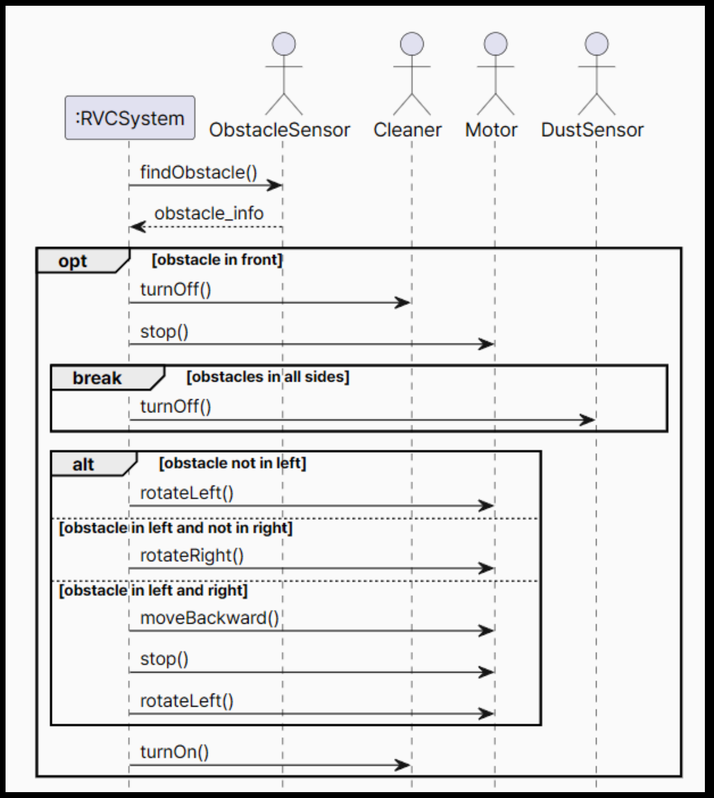
System Sequence Diagram



UC #5



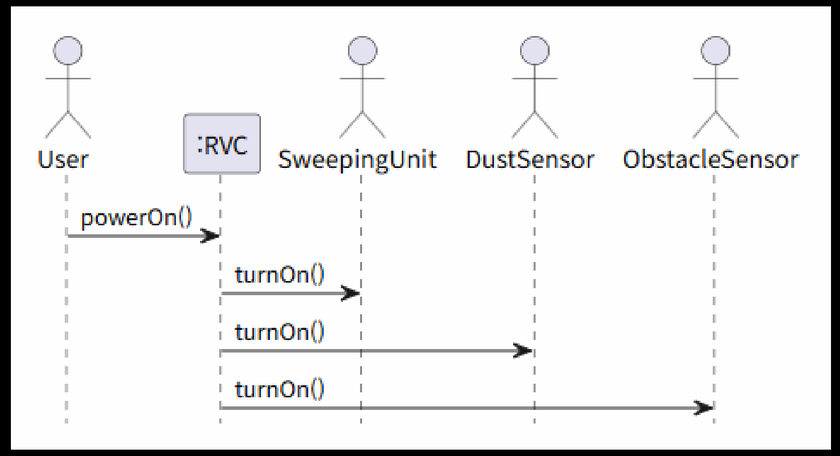
UC #6



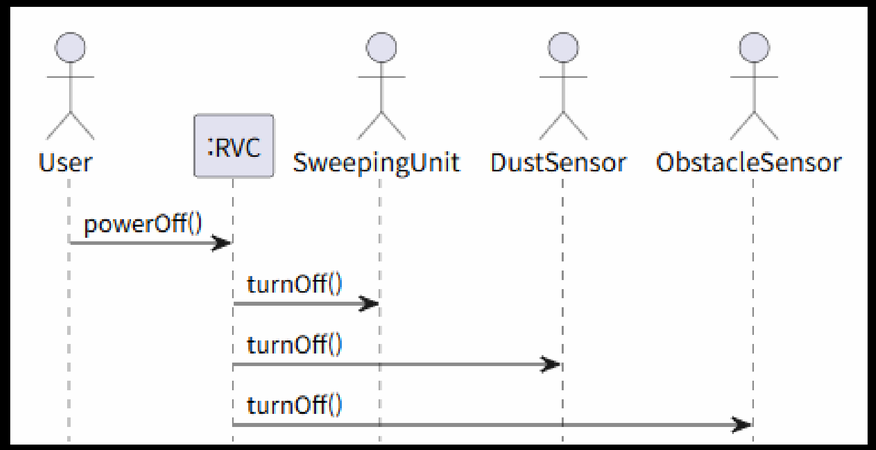
UC #7

System Sequence Diagram

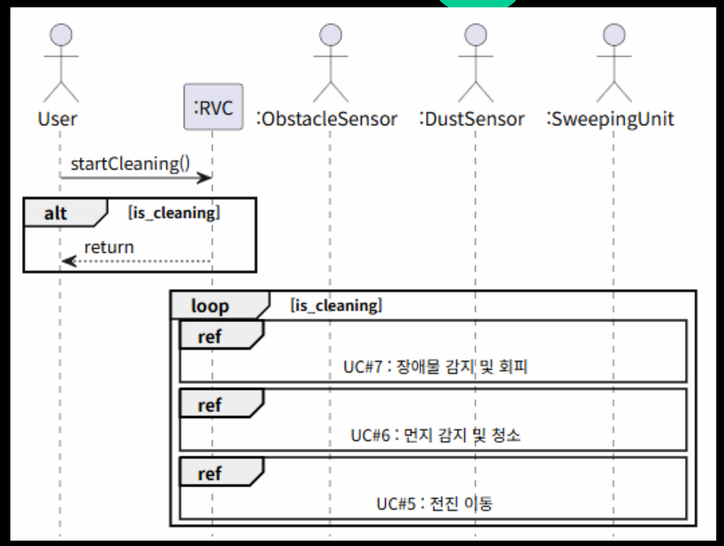
Update



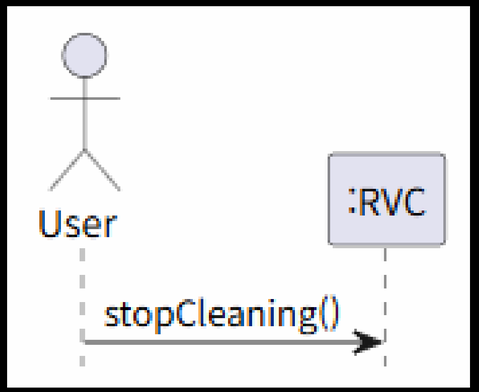
UC #1



UC #2



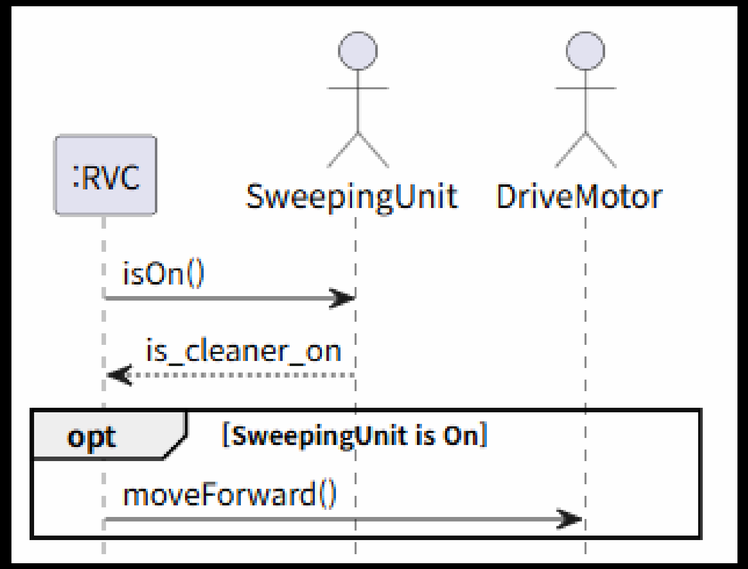
UC #3



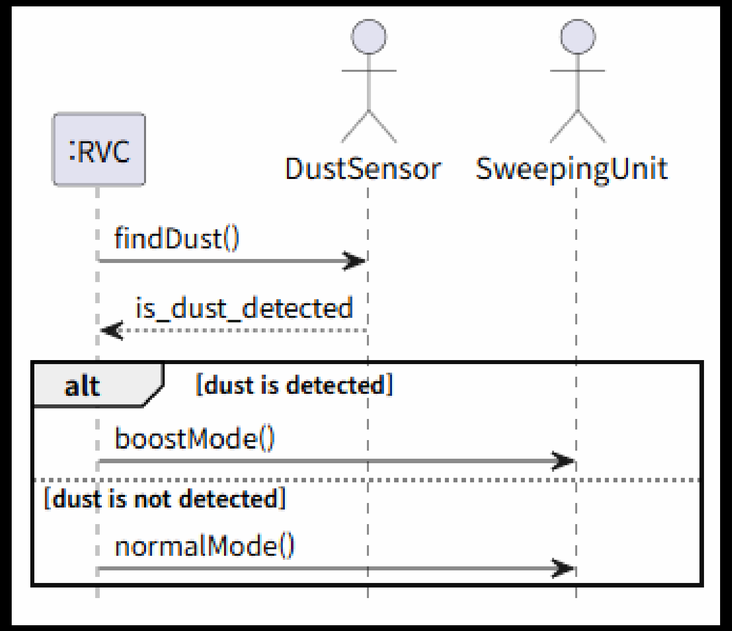
UC #4

System Sequence Diagram

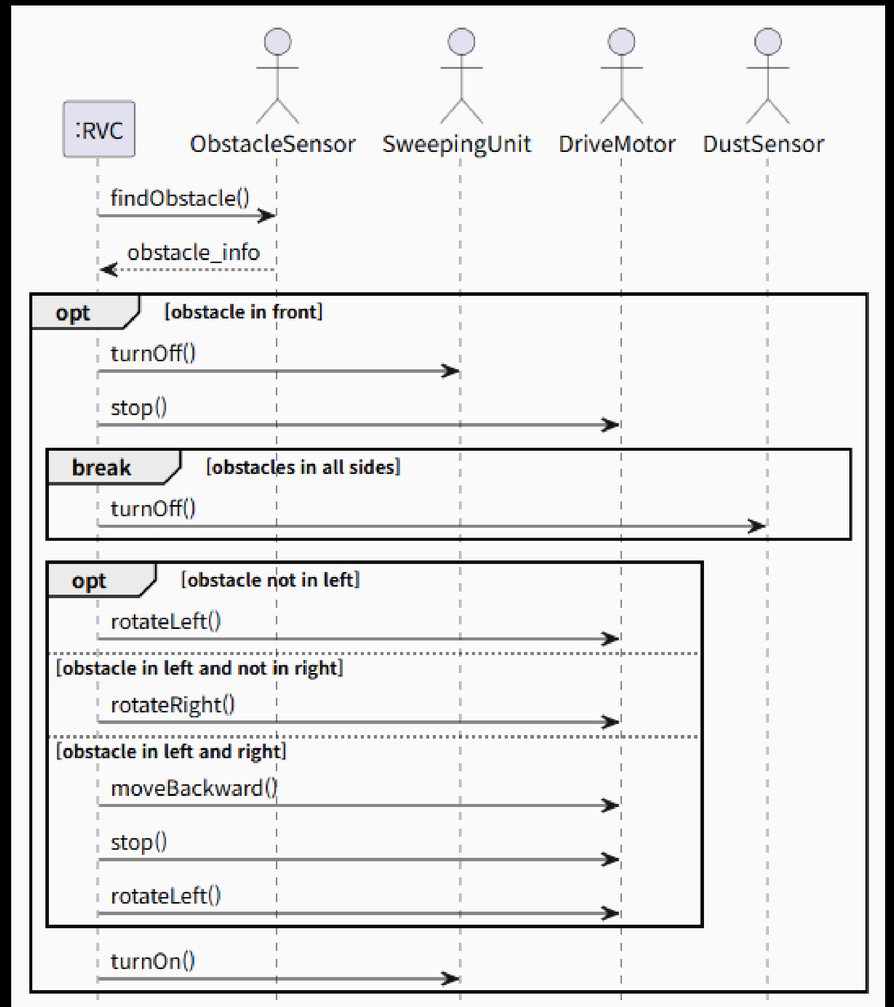
Update



UC #5

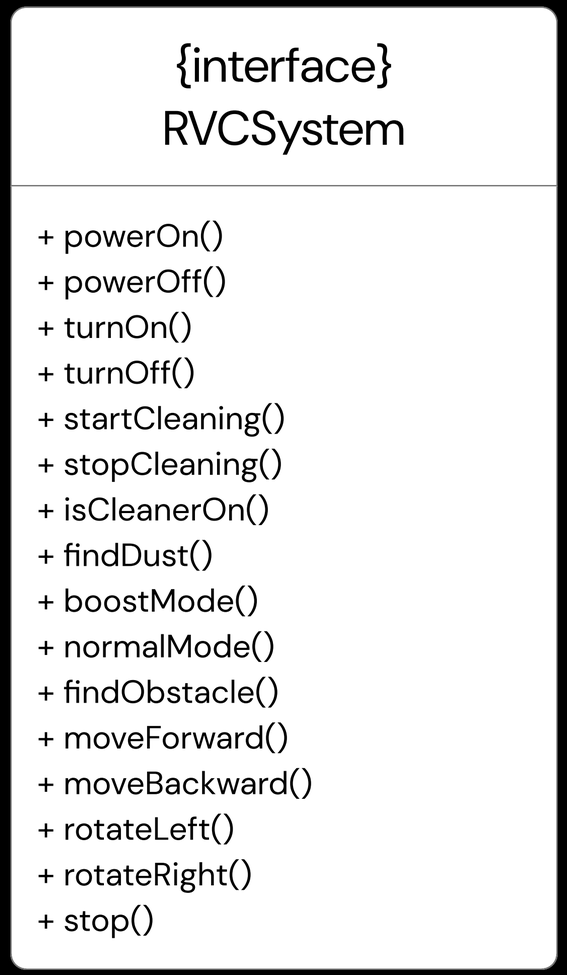


UC #6



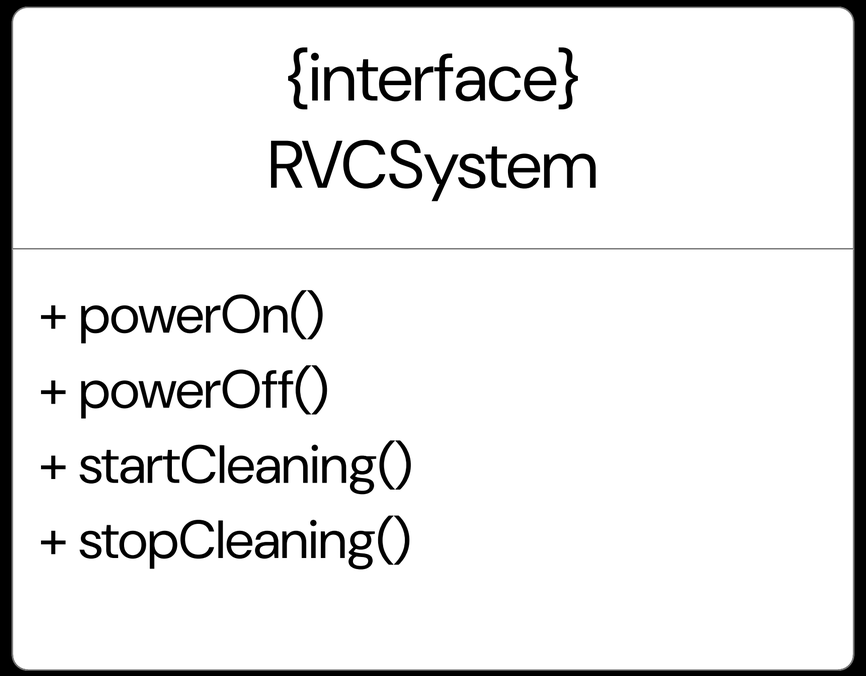
UC #7

System Operations

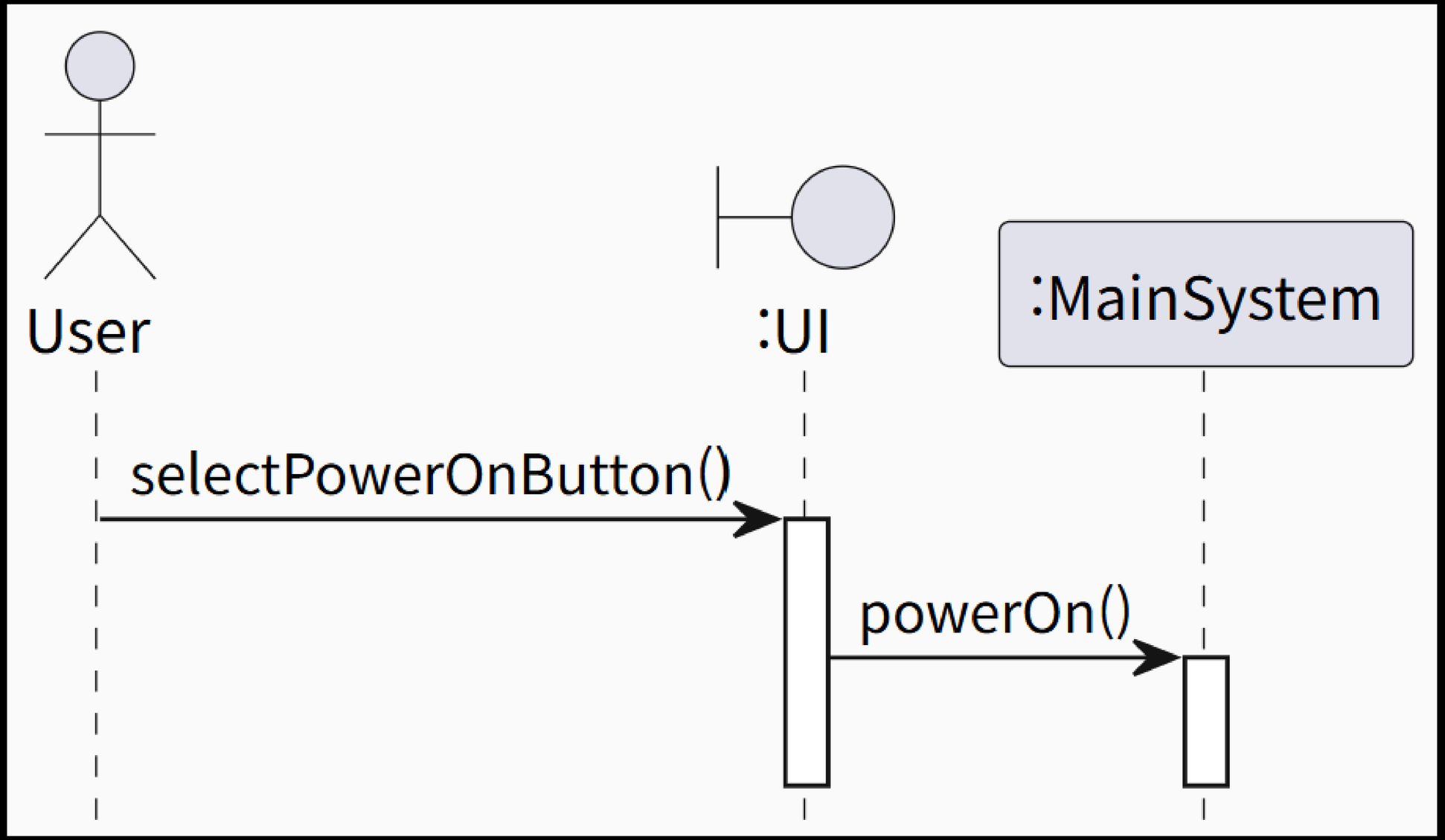


System Operations

Update

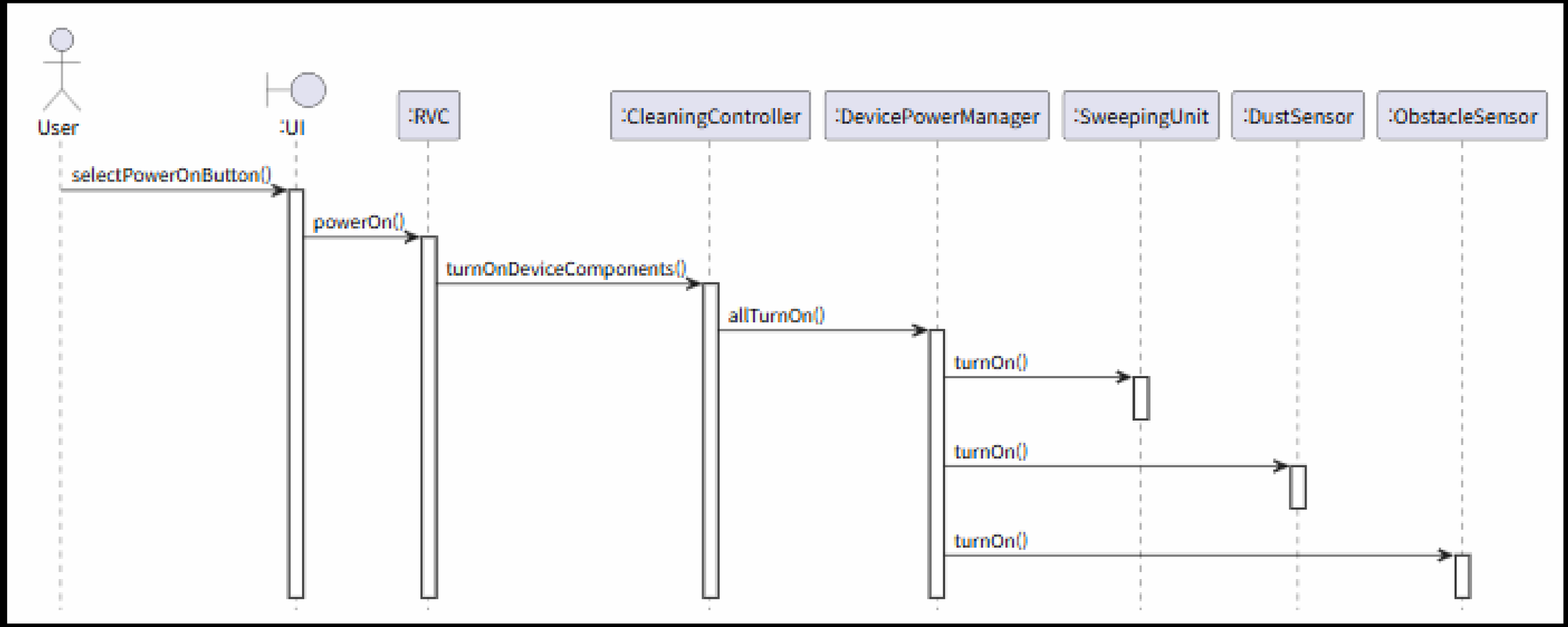


Sequence Diagram : UC #1

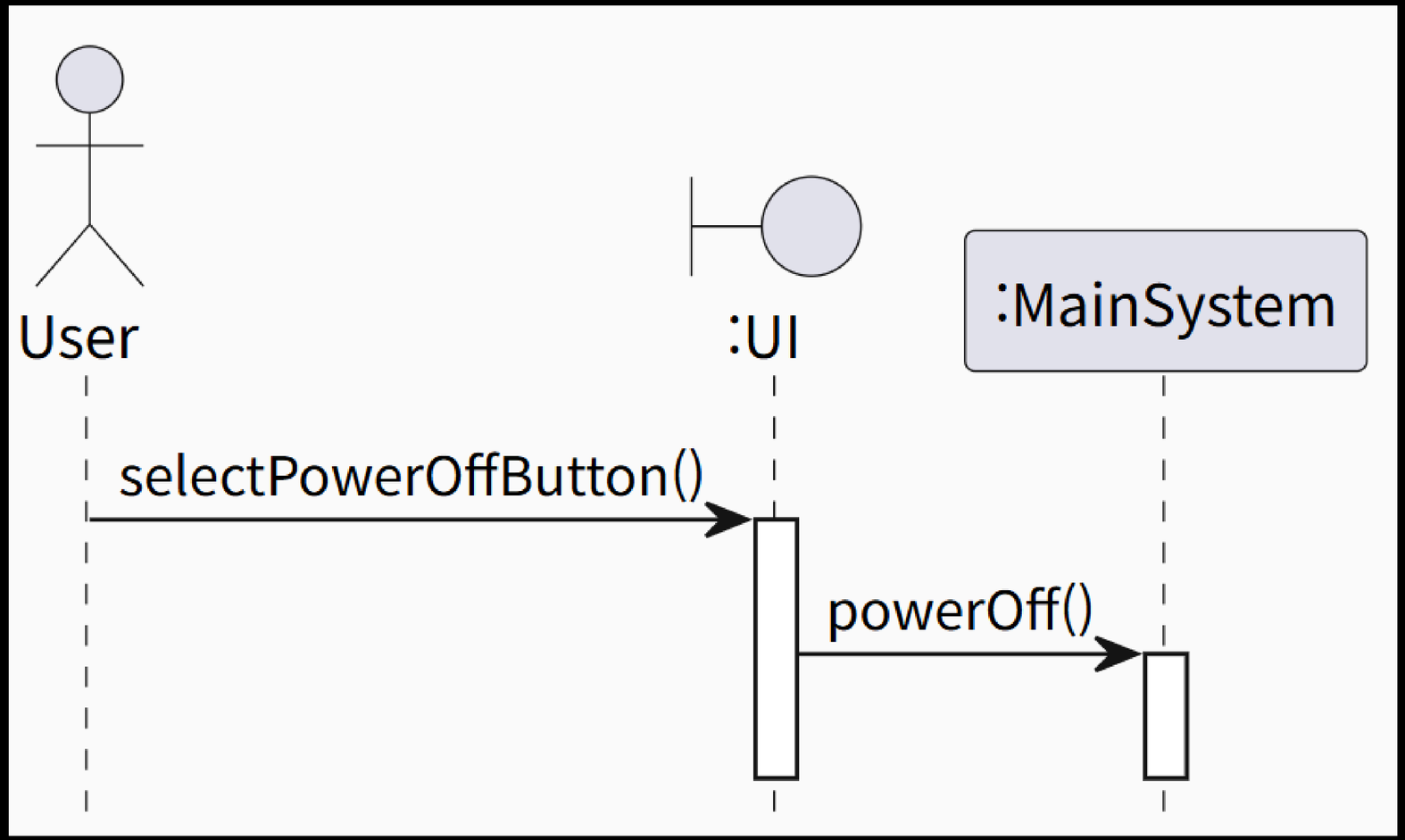


Sequence Diagram : UC #1

Update

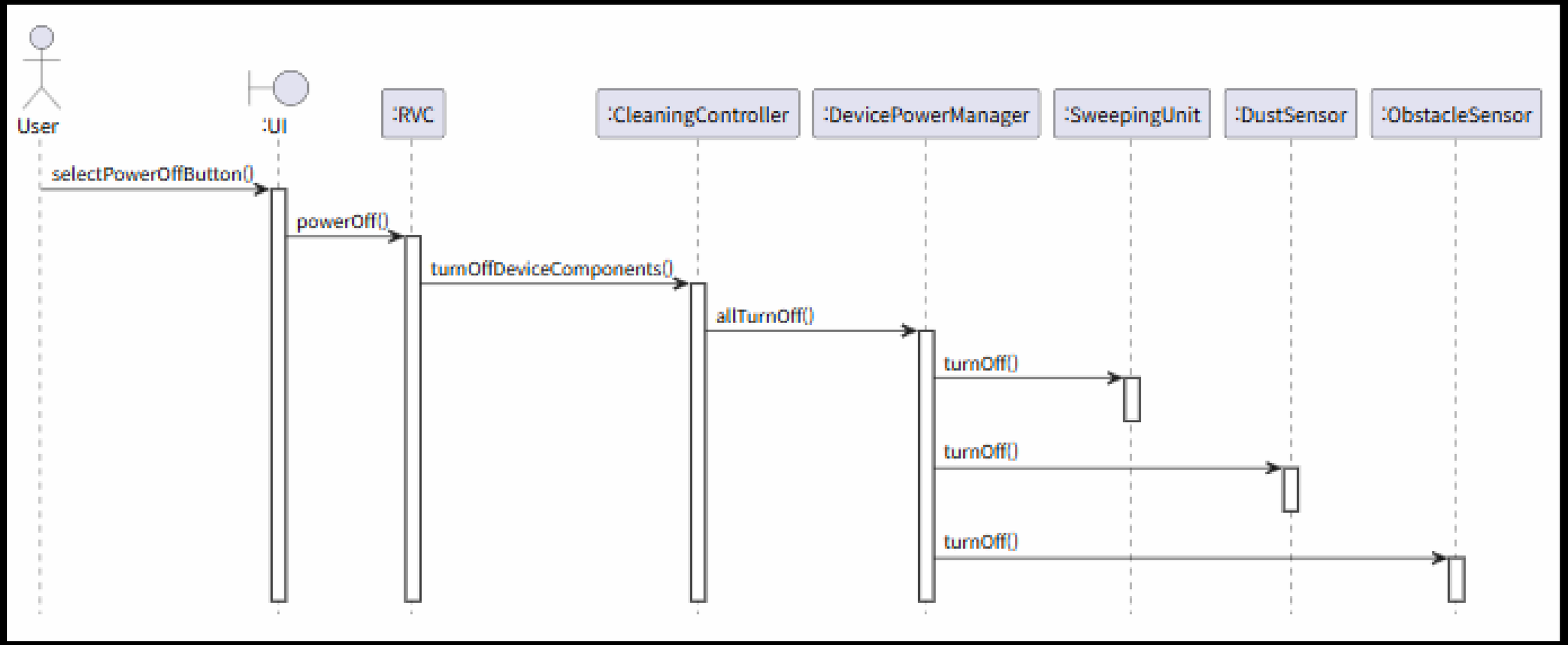


Sequence Diagram : UC #2

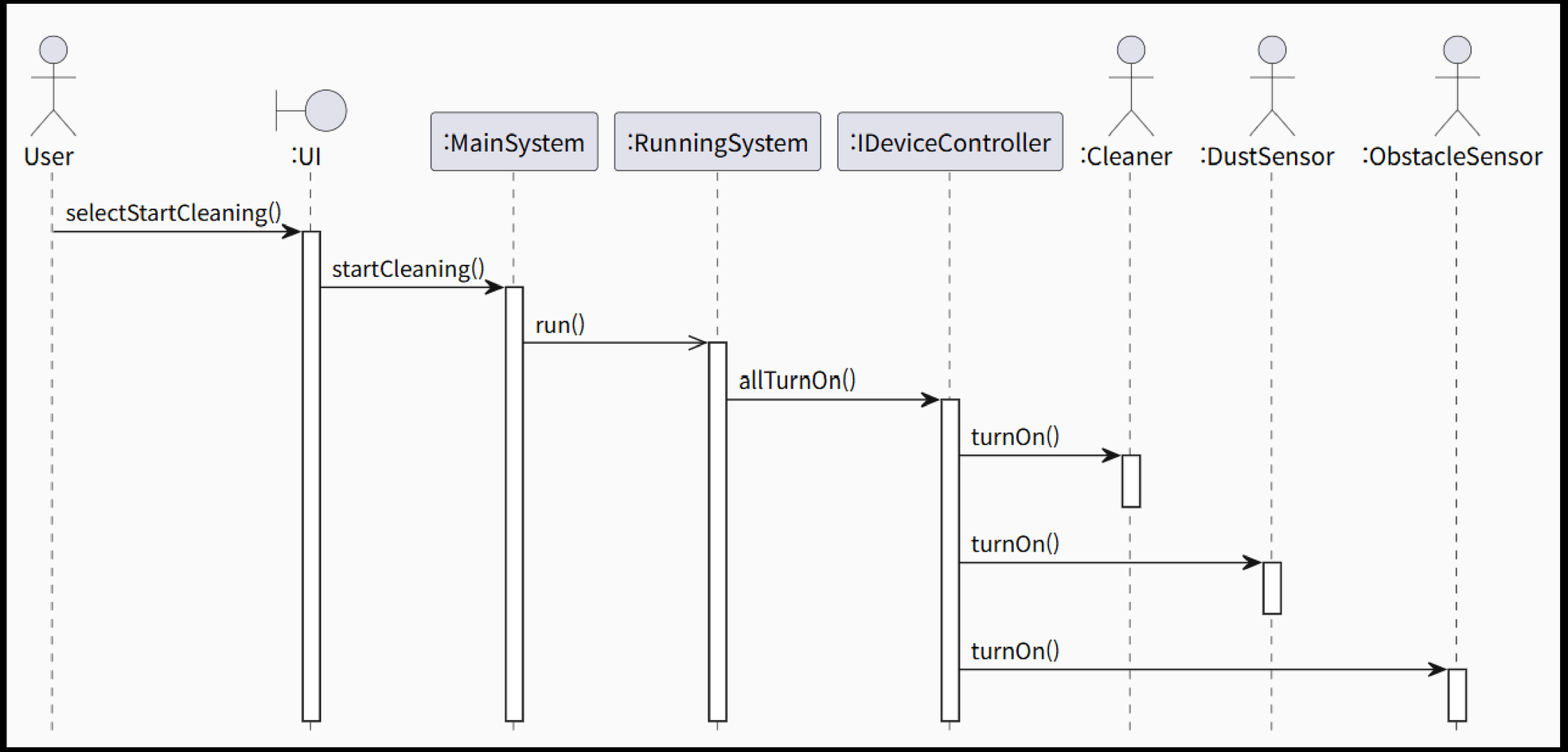


Sequence Diagram : UC #2

Update

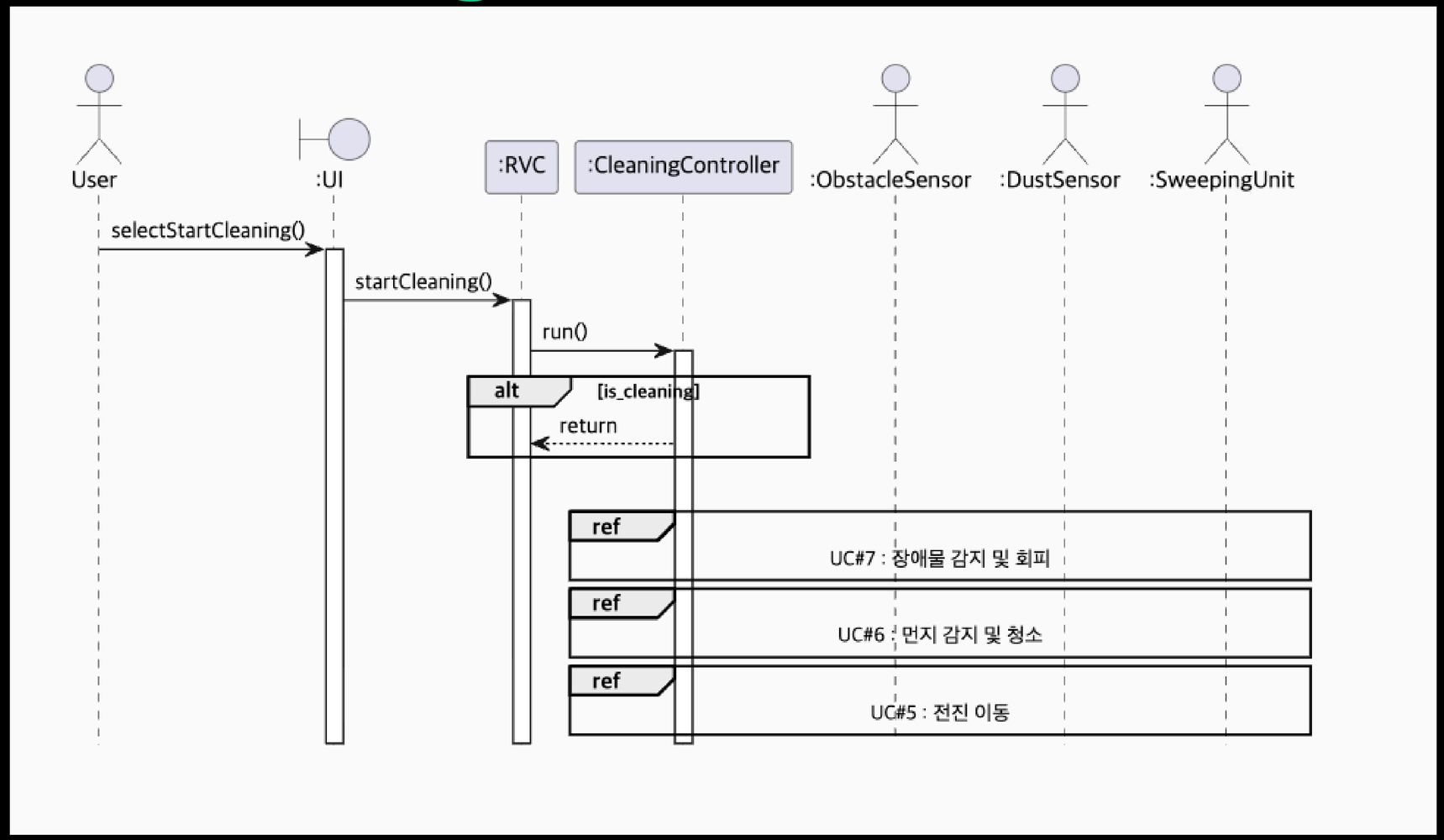


Sequence Diagram : UC #3

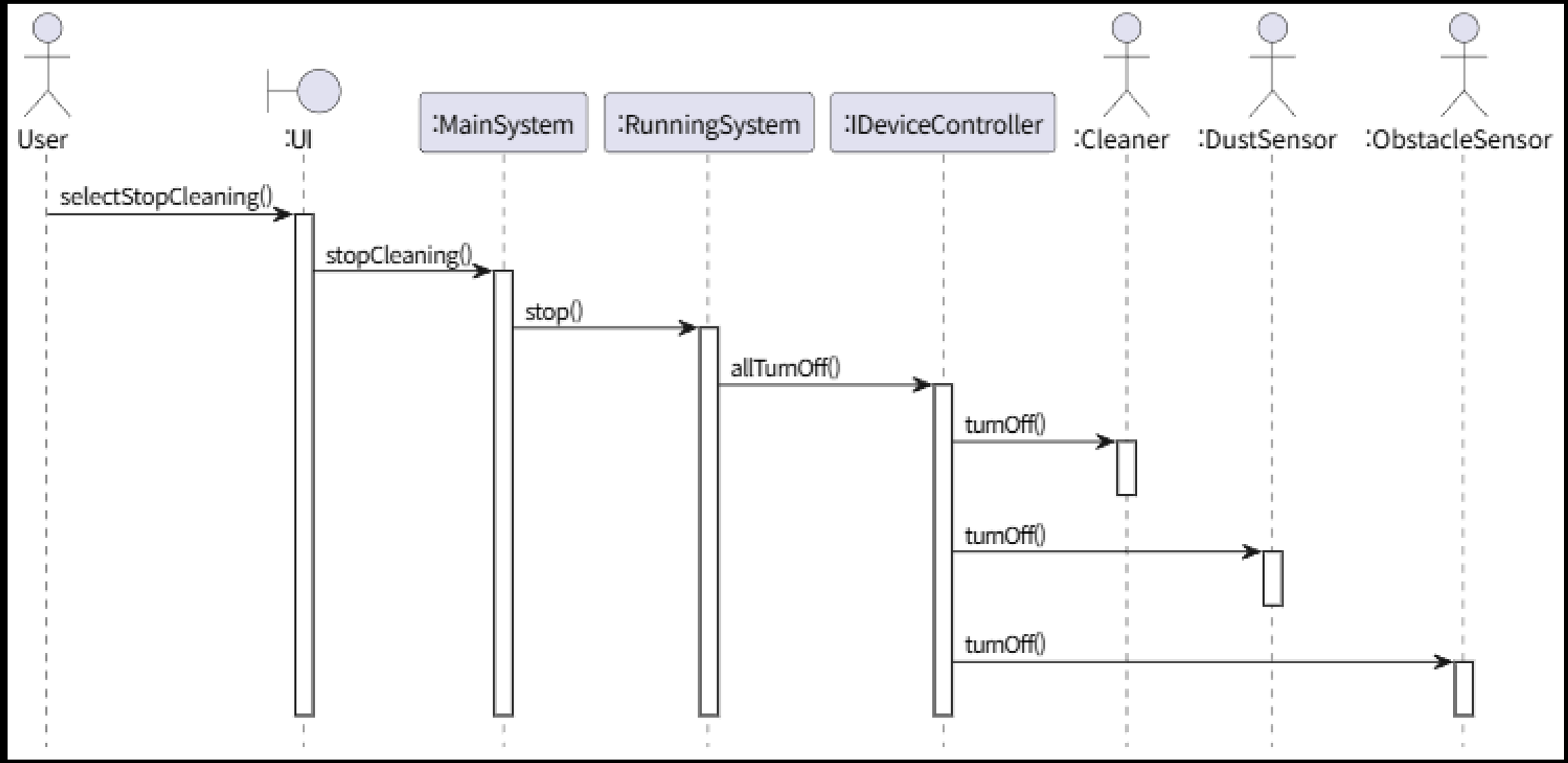


Sequence Diagram : UC #3

Update

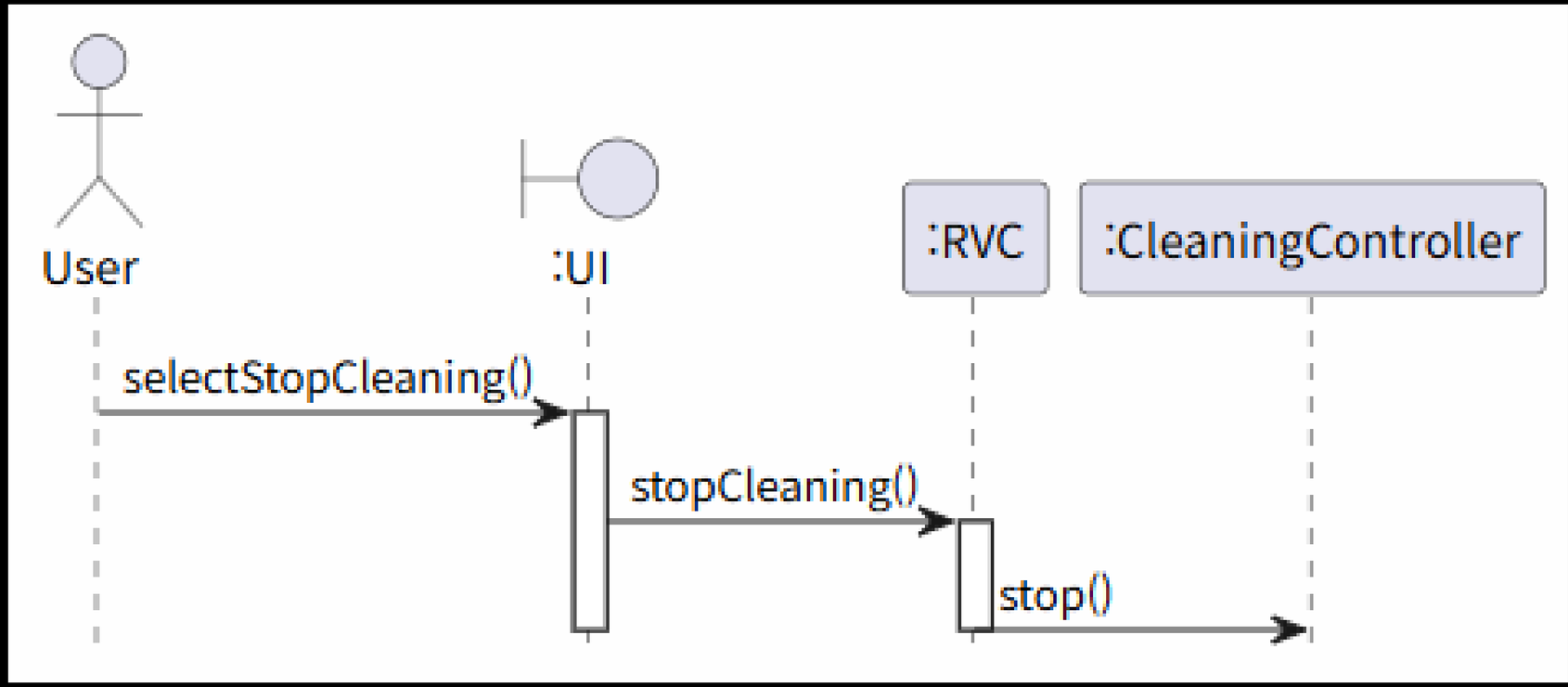


Sequence Diagram : UC #4

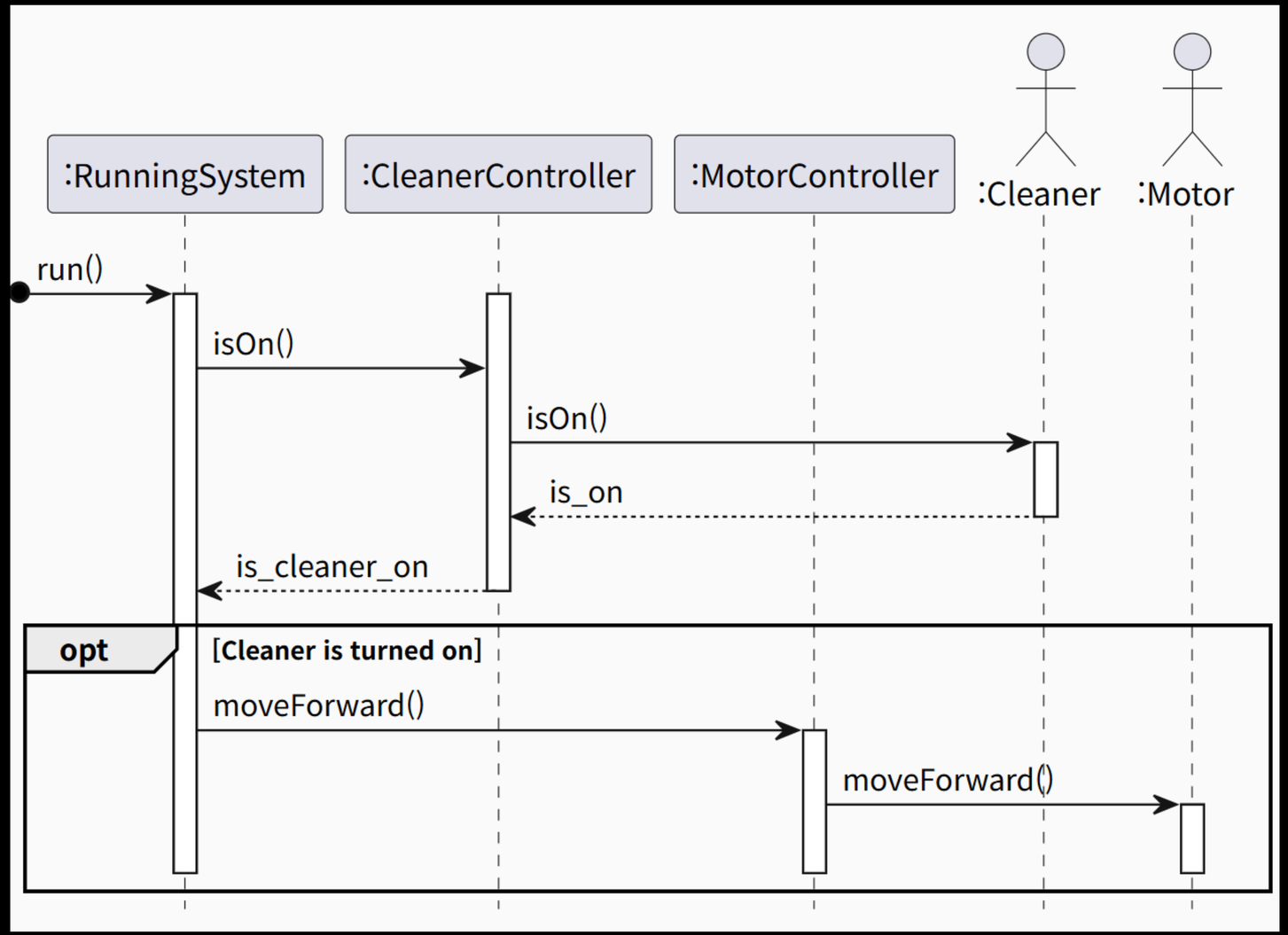


Sequence Diagram : UC #4

Update

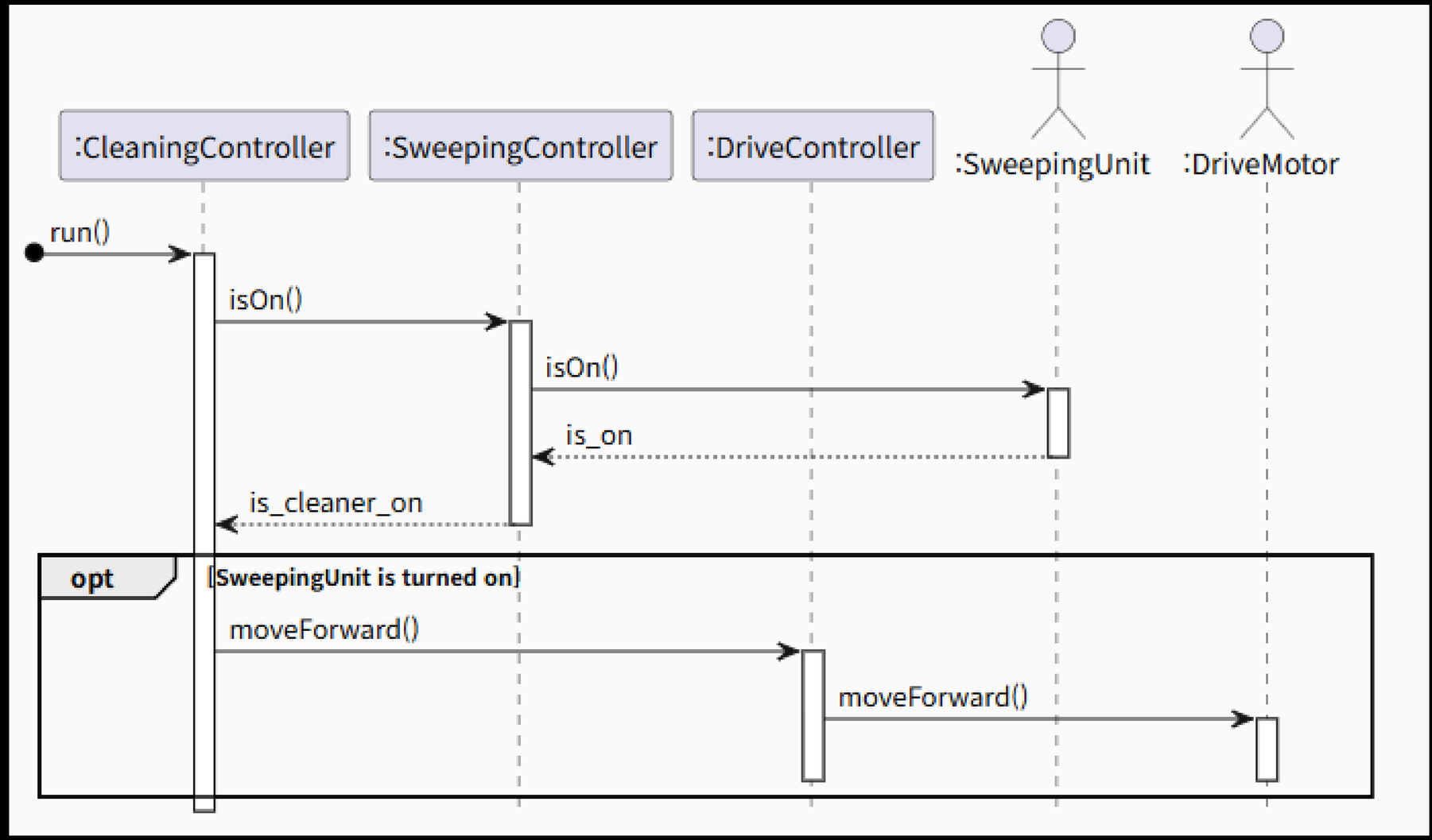


Sequence Diagram : UC #5

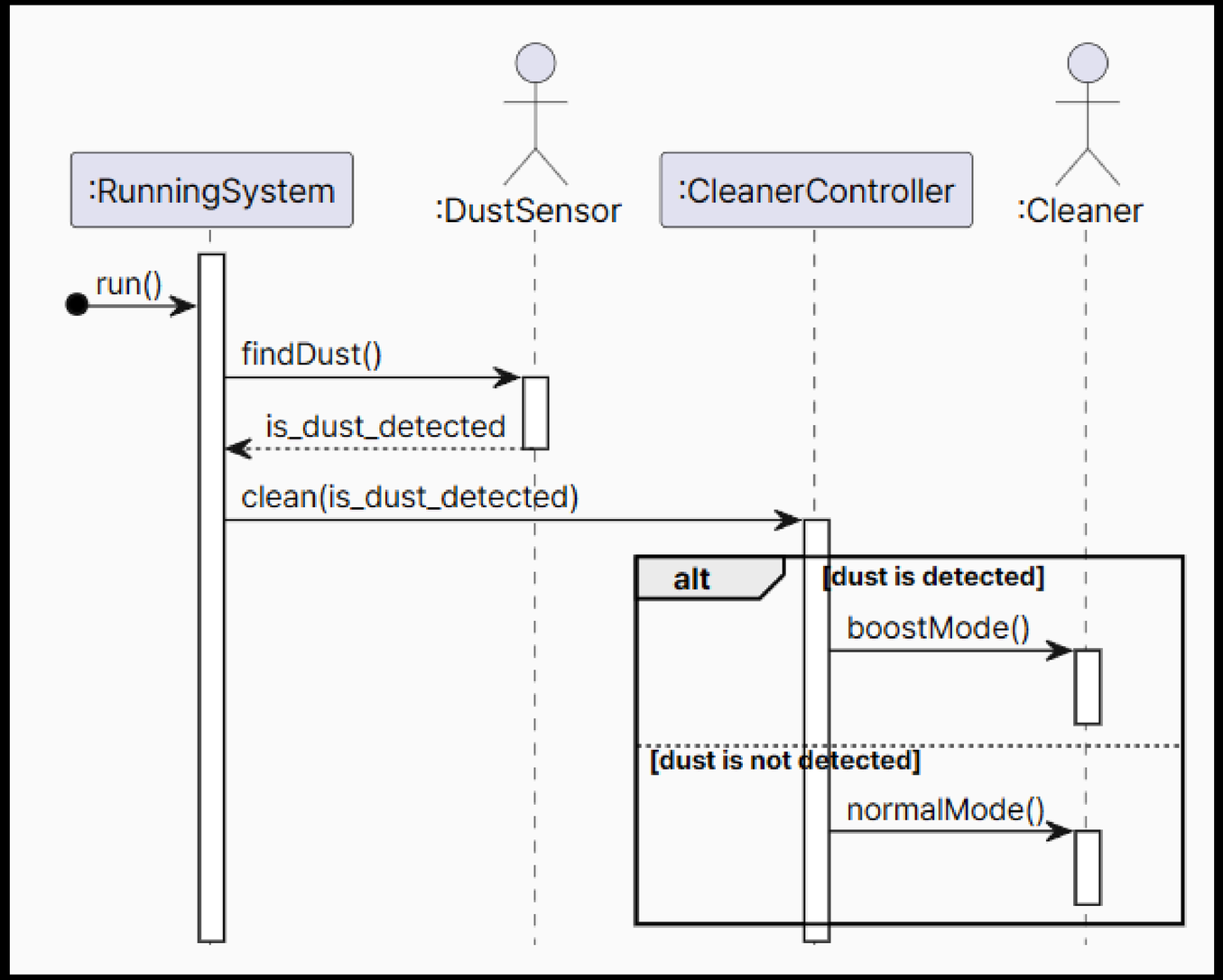


Sequence Diagram : UC #5

Update

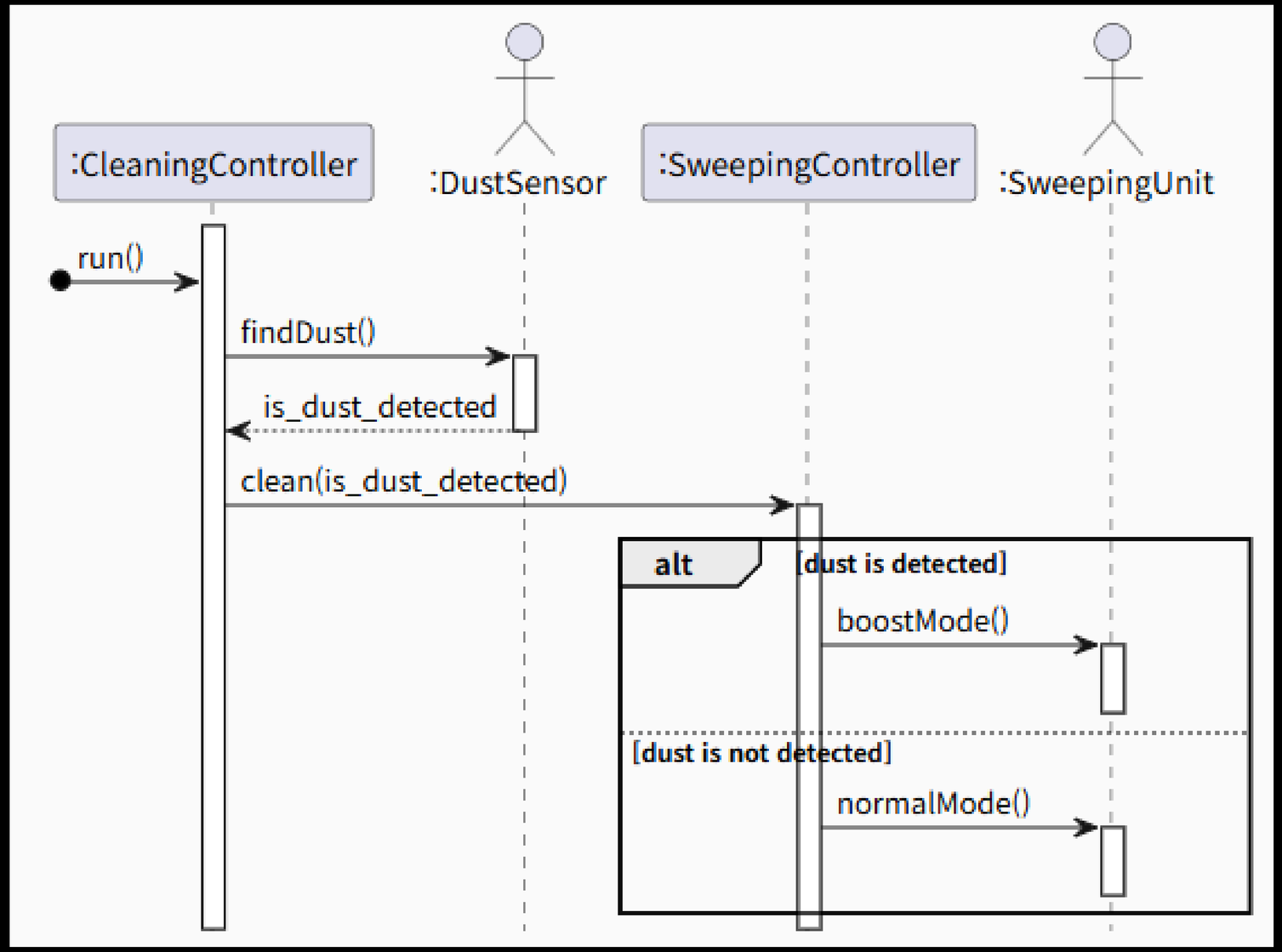


Sequence Diagram : UC #6

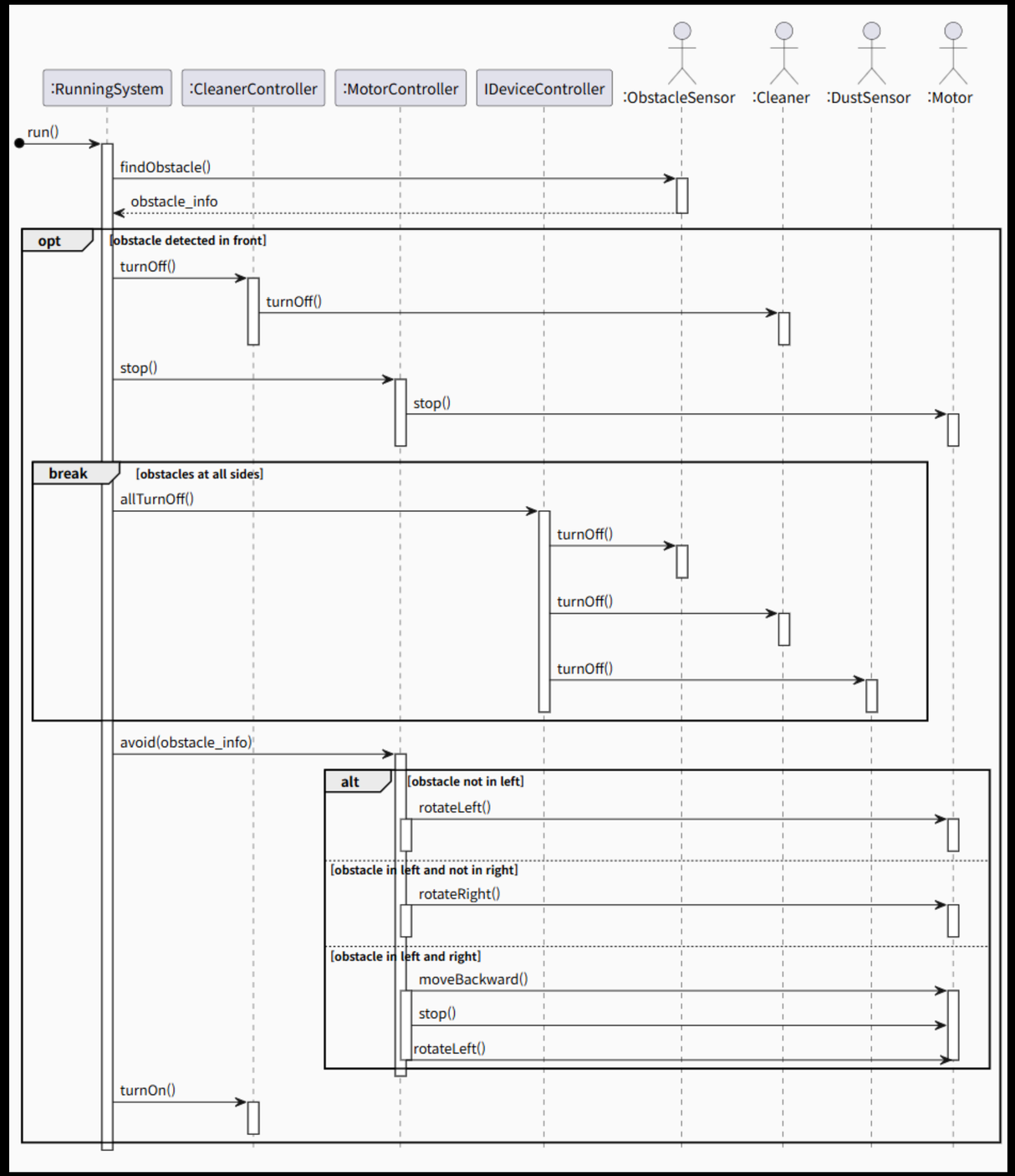


Sequence Diagram : UC #6

Update

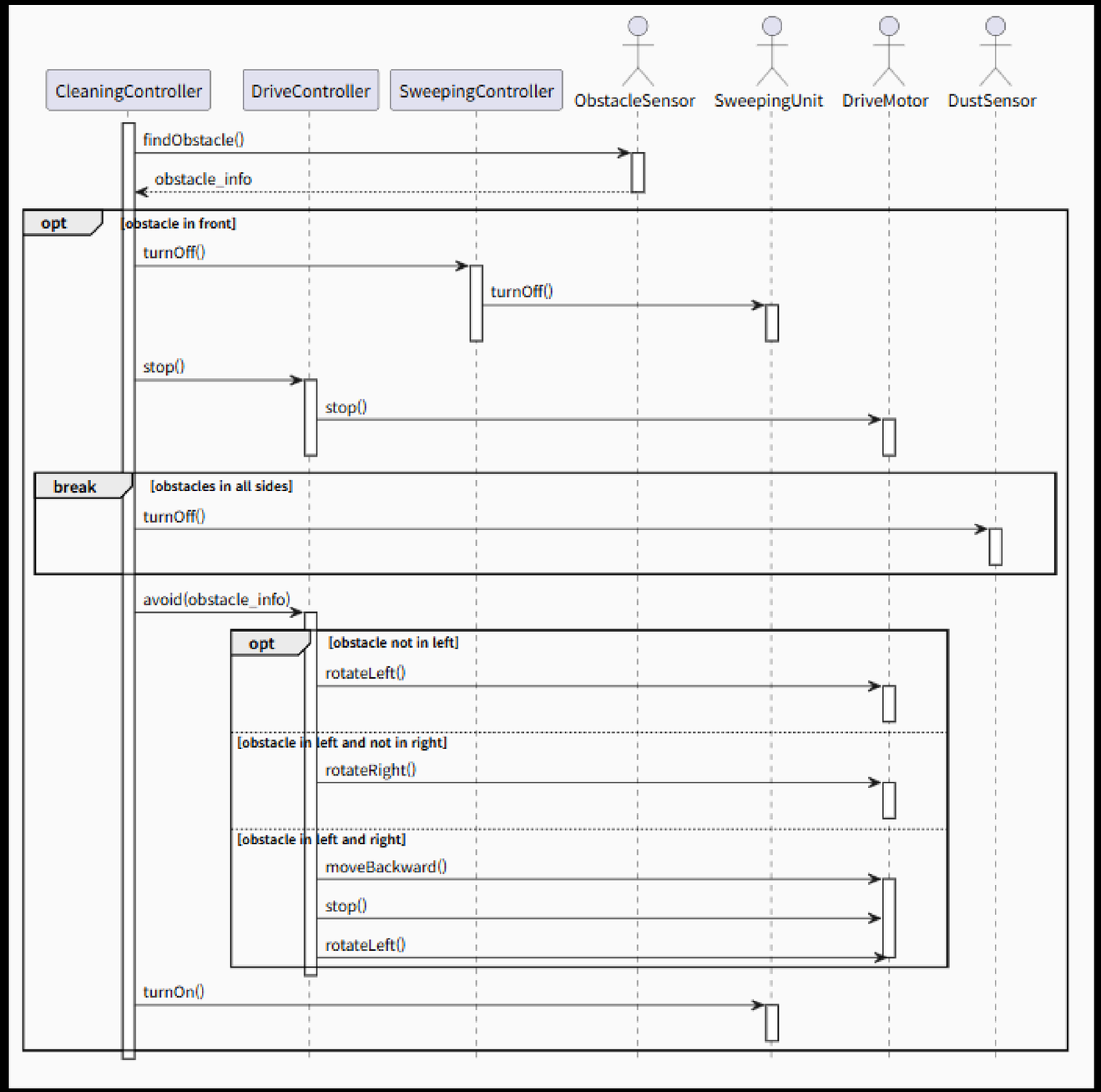


Sequence Diagram : UC #7

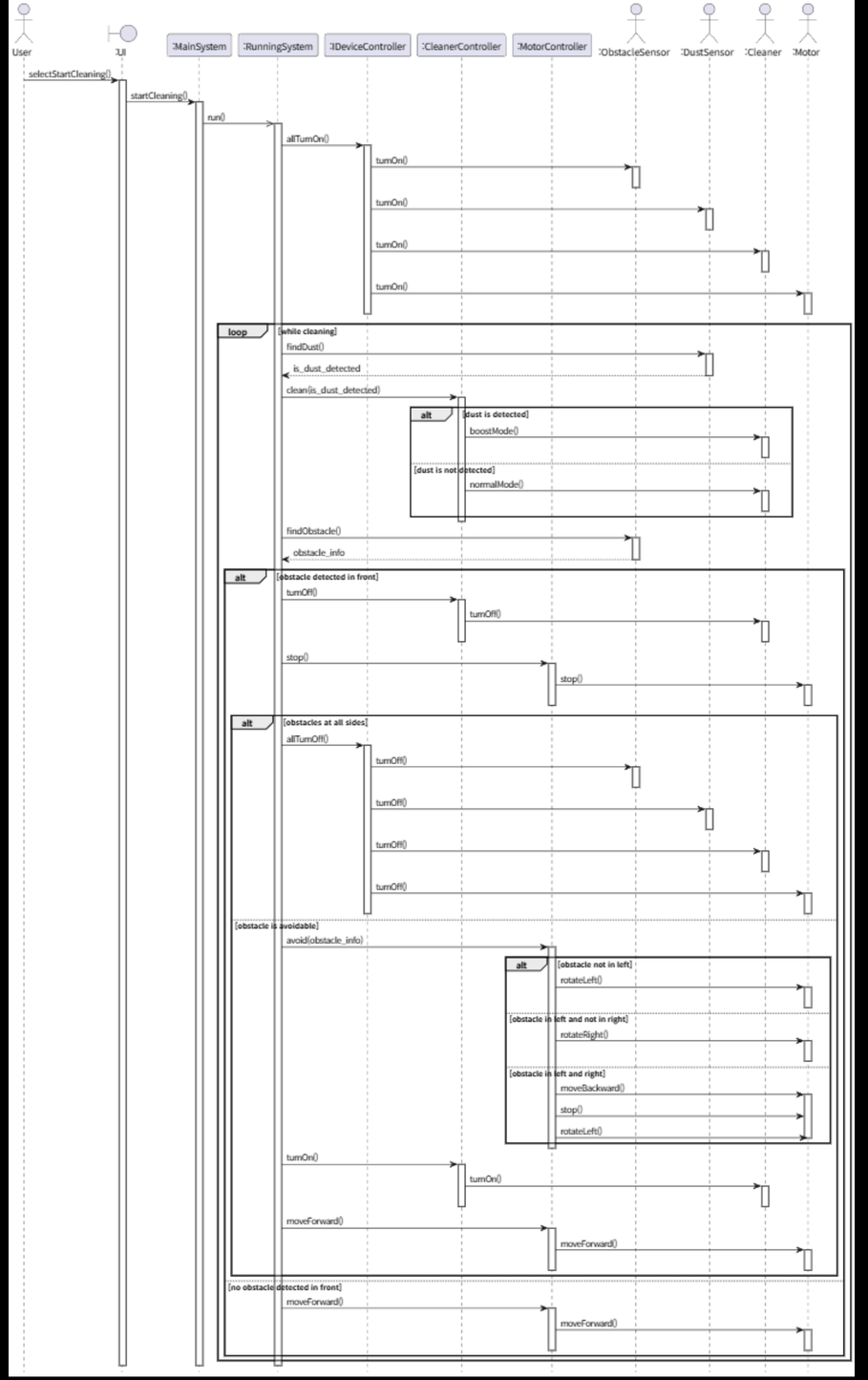


Sequence Diagram : UC #7

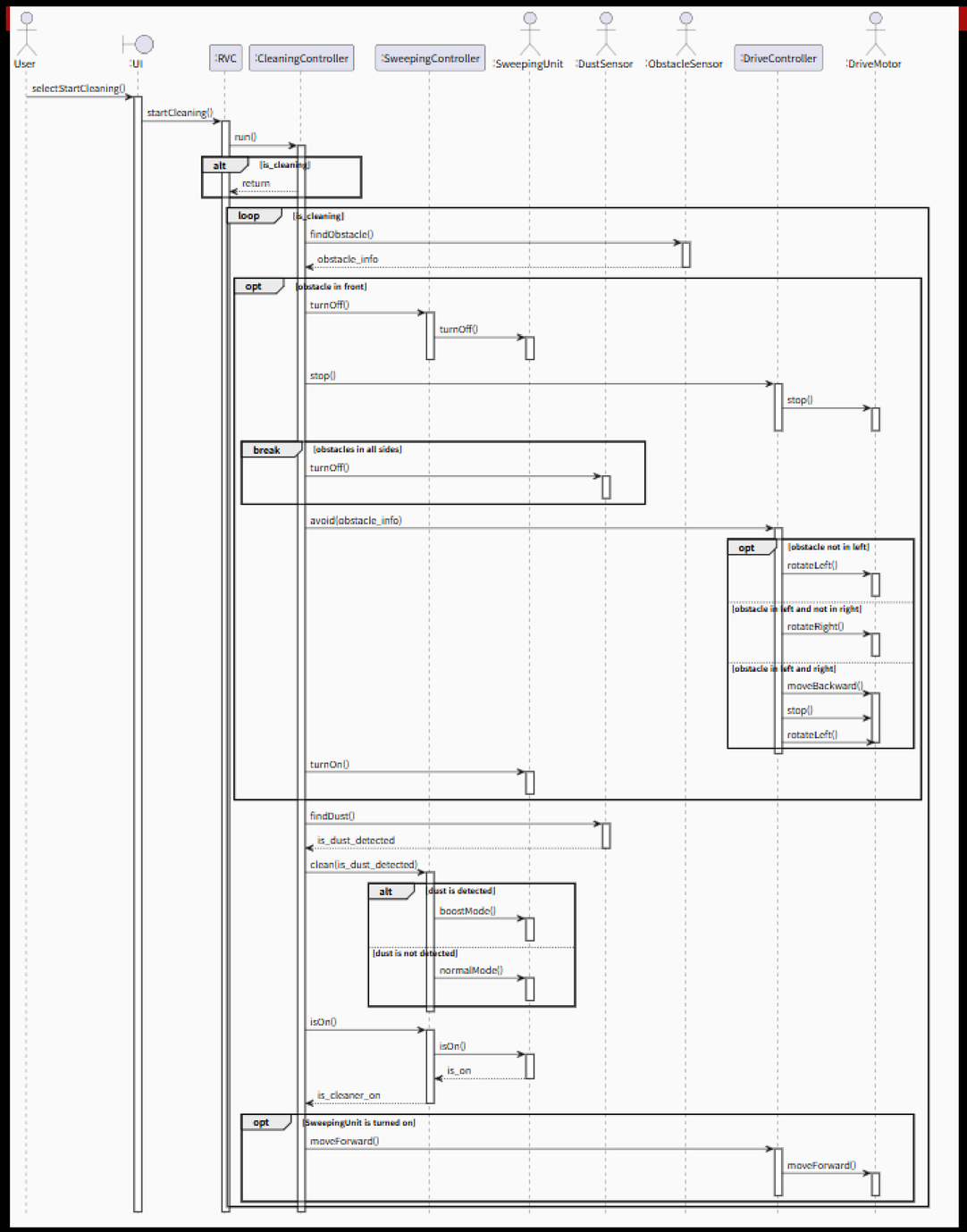
Update



Sequence Diagram : Whole Sequence

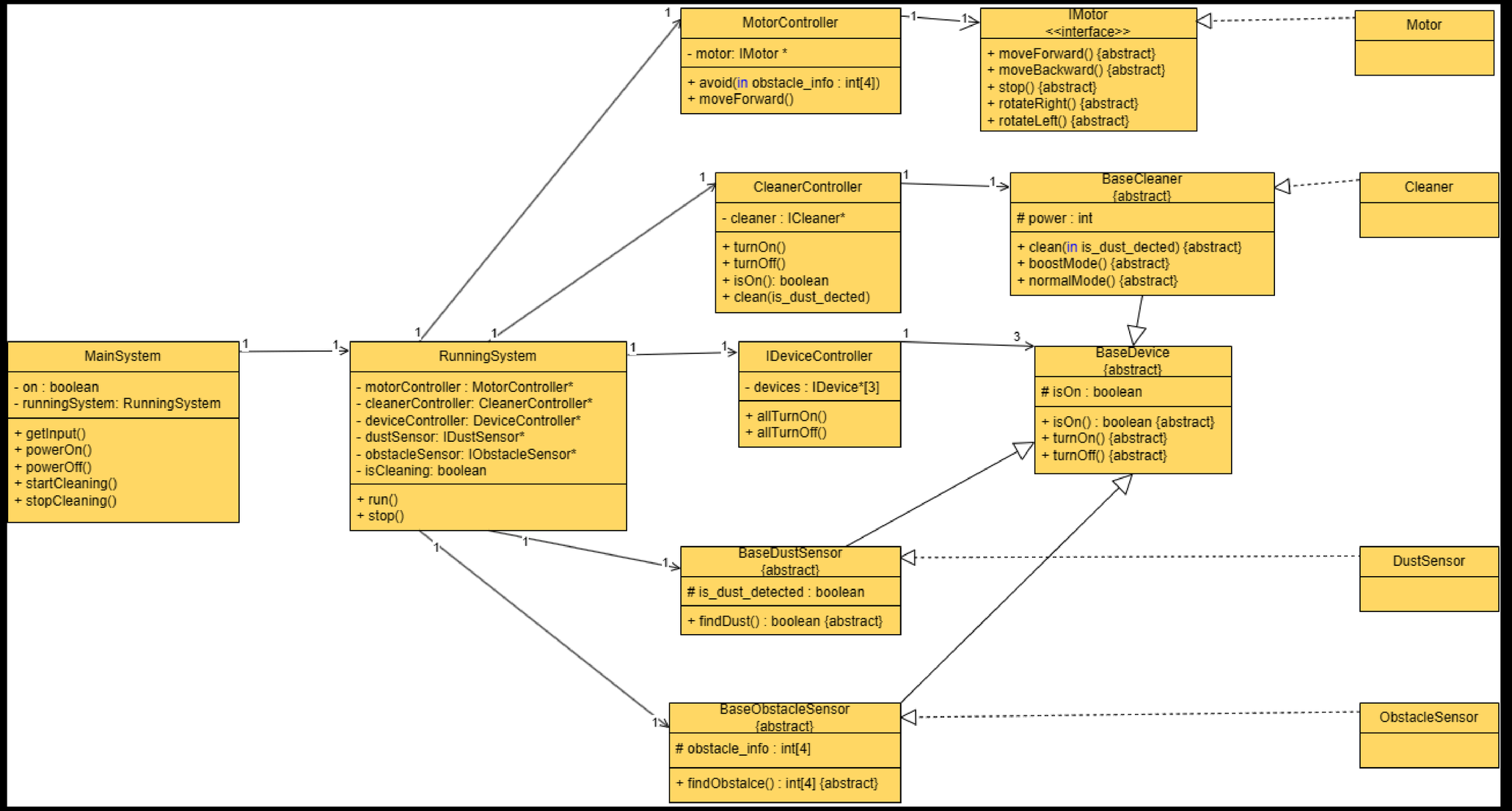


Sequence Diagram : Whole Sequence



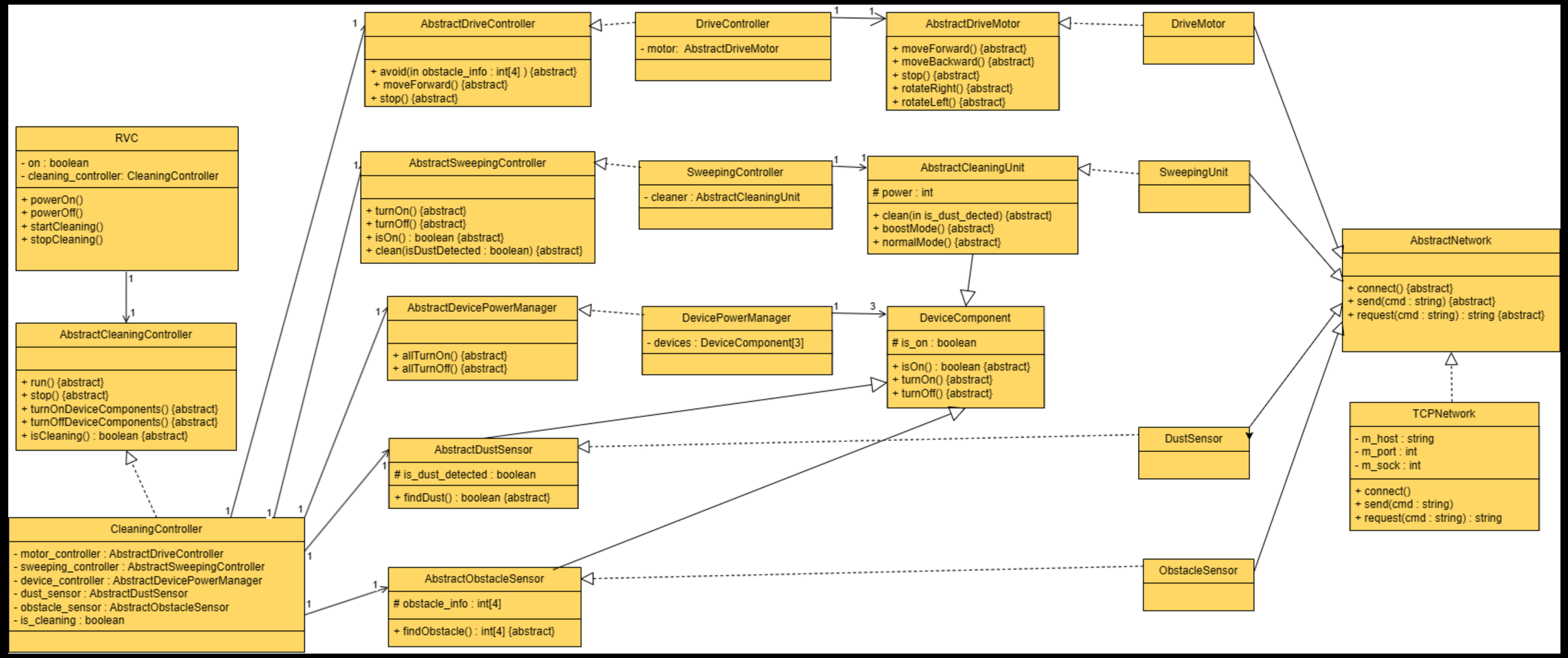
Update

Class Diagram



Class Diagram

Update



CI/CD

Development

IDE

Build Management

CMake

Unit Testing

googletest

Version Control

GitHub

Requirements Management

Jira

Team Communication

Server

ORACLE CLOUD Infrastructure

CI

Continuous Integration

GitHub Action

Build Management

CMake

Unit Testing

googletest

Static Code Analysis

clang-tidy

Coverage

gcovr

CI/CD

Update

Development

IDE

Build Management

CMake

Unit Testing

googletest

Version Control

GitHub

Requirements Management

Jira

Team Communication

Server

GitHub

CI

Continuous Integration

GitHub Actions

Build Management

CMake

Unit Testing

googletest

Static Code Analysis

clang-tidy

Coverage

gcovr

Unit Test

DriveController

Test #	Test Group	Test Name	Expected Result
DC-1	DriveController	Constructor_Nullptr_Throws	`nullptr` 입력 시 `std::invalid_argument` throw
DC-2	DriveController	Avoid_AllZero_Throws	`{0,0,0,0}` 입력 시 `std::invalid_argument` throw, 모터 미호출
DC-3	DriveController	Avoid_FrontBlocked_OthersOpen_Throws	`{0,1,1,1}` 입력 시 `std::invalid_argument` throw, 모터 미호출
DC-4	DriveController	Avoid_AllBlocked_Throws	`{1,1,1,1}` 입력 시 `std::invalid_argument` throw, 모터 미호출
DC-5	DriveController	Avoid_FrontBlocked_LeftOpen_RotateLeft	`{1,0,0,0}` 입력 시 `rotateLeft` ×1, 나머지 ×0
DC-6	DriveController	Avoid_FrontBlocked_LeftOpen_IrrelevantRightBack	`{1,0,1,1}` 입력 시 `rotateLeft` ×1 (left 열림이면 right·back 무관)
DC-7	DriveController	Avoid_FrontLeftBlocked_RightOpen_RotateRight	`{1,1,0,0}` 입력 시 `rotateRight` ×1, 나머지 ×0
DC-8	DriveController	Avoid_RightOpen_BackIrrelevant_RotateRight	`{1,1,0,1}` 입력 시 `rotateRight` ×1 (back 상태 무관)
DC-9	DriveController	Avoid_OnlyBackOpen_BackThenStopThenRotateLeft	`{1,1,1,0}` 입력 시 `moveBackward` ×1, `stop` ×1, `rotateLeft` ×1, 나머지 ×0
DC-10	DriveController	Avoid_OnlyBackOpen_CallOrder	호출 순서: `moveBackward` → `stop` → `rotateLeft`
DC-11	DriveController	MoveForward_CallsMotorOnce	`moveForward()` ×1 → `motor.moveForward` ×1, 나머지 ×0
DC-12	DriveController	Stop_CallsMotorOnce	`stop()` ×1 → `motor.stop` ×1, 나머지 ×0
DC-13	DriveController	MoveForward_ThreeTimes	`moveForward()` ×3 → `motor.moveForward` ×3
DC-14	DriveController	MoveForward_AfterException_StillWorks	throw 발생 후 `moveForward()` 호출 시 예외 없이 정상 호출, `moveForward` ×1

DriveMotor

Test #	Test Group	Test Name	Expected Result
DM-1	DriveMotor	DefaultCtor_NoNetwork_NoSend_NoCrash	`DriveMotor motor` (network=null) 후 5개 메서드 호출해도 crash 없이 종료, `send()` 미호출
DM-2	DriveMotor	MoveForward_SendsCommand	`moveForward()` → `send("MOVE_FORWARD")` 1회
DM-3	DriveMotor	MoveBackward_SendsCommand	`moveBackward()` → `send("MOVE_BACKWARD")` 1회
DM-4	DriveMotor	Stop_SendsCommand	`stop()` → `send("STOP_MOTOR")` 1회
DM-5	DriveMotor	RotateRight_SendsCommand	`rotateRight()` → `send("ROTATE_RIGHT")` 1회
DM-6	DriveMotor	RotateLeft_SendsCommand	`rotateLeft()` → `send("ROTATE_LEFT")` 1회
DM-7	DriveMotor	MoveForward_ThreeTimes_SendsThree	`moveForward()` ×3 → `send("MOVE_FORWARD")` ×3
DM-8	DriveMotor	MixedCalls_SendEachOnce	`moveForward()` → `stop()` → `rotateLeft()` → 각 커맨드 1회씩, 총 3회 전송

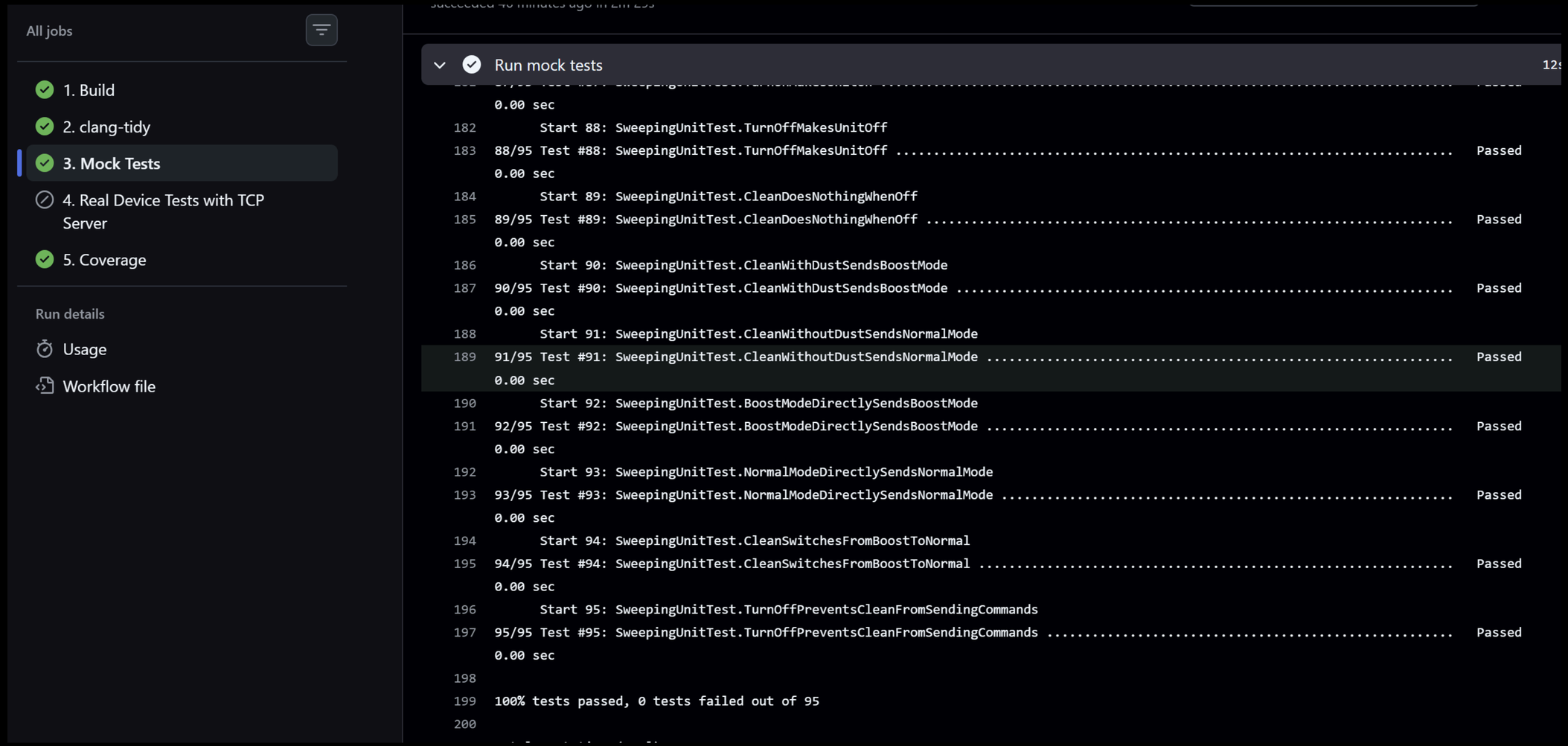
ObstacleSensor

Test #	Test Group	Test Name	Expected Result
OS-1	ObstacleSensor	InitialStateIsOff	생성 직후 `isOn()` == false
OS-2	ObstacleSensor	TurnOnMakesSensorOn	`turnOn()` 후 `isOn()` == true
OS-3	ObstacleSensor	TurnOffMakesSensorOff	`turnOn()` → `turnOff()` 후 `isOn()` == false
OS-4	ObstacleSensor	FindObstacleReturnsZerosWhenPowerIsOff	전원 OFF → `{0,0,0,0}`, 네트워크 미호출
OS-5	ObstacleSensor	FindObstacleReturnsZerosOnInvalidResponse	응답 `"GARBAGE"` → `{0,0,0,0}`
OS-6	ObstacleSensor	FindObstacleReturnsZerosOnEmptyResponse	응답 `""` → `{0,0,0,0}`
OS-7	ObstacleSensor	FindObstacleRequestsNetworkEveryCall	ON 상태에서 2회 호출 → 2회 요청 발생
OS-8	ObstacleSensor	TurnOffPreventsObstacleSensorFromQuerying	`turnOn()` → `turnOff()` 후 호출 → `{0,0,0,0}`, 네트워크 미호출
OS-9	ObstacleSensorValidInputTest	ParsesValidObstacleResponseWhenPowerIsOn	전원 ON 시 16개 조합(2^4) 모두 파싱 결과(front/right/back/left)가 입력과 일치

CleaningController

Test #	Test Group	Test Name	Expected Result
CC-1	CleaningController	ConstructorThrowsWhenDriveControllerIsNull	`drive` 가 `nullptr` 이면 `std::invalid_argument` throw
CC-2	CleaningController	ConstructorThrowsWhenSweepingControllerIsNull	`sweeping` 이 `nullptr` 이면 `std::invalid_argument` throw
CC-3	CleaningController	ConstructorThrowsWhenDustSensorIsNull	`dustSensor` 가 `nullptr` 이면 `std::invalid_argument` throw
CC-4	CleaningController	ConstructorThrowsWhenObstacleSensorIsNull	`obstacleSensor` 가 `nullptr` 이면 `std::invalid_argument` throw
CC-5	CleaningController	RunOnly	(전원 OFF) `run()` 호출 후 `isCleaning()` == false
CC-6	CleaningController	StopOnly	(전원 OFF) `stop()` 호출 후 `isCleaning()` == false
CC-7	CleaningController	RunThenStop	(전원 OFF) `run()` → 종료 대기 → `stop()` 후 `isCleaning()` == false
CC-8	CleaningController	TurnOnThenRun	`turnOnDeviceComponents()` → `run()` 후 종료 시점에 `isCleaning()` == false
CC-9	CleaningController	TurnOnThenRunThenStop	`turnOn` → `run` → `stop` 후 종료 시점에 `isCleaning()` == false
CC-10	CleaningController	TurnOnThenRunThenTurnOff	`turnOn` → `run` → 종료 → `turnOff` 후 `isCleaning()` == false
CC-11	CleaningController	TurnOnThenRunThenStopAgain	`turnOn` → `run` → 종료 → `stop` 후 `isCleaning()` == false
CC-12	CleaningController	TurnOnThenTurnOff	호출 순서: `allTurnOn` → `allTurnOff` (순서 보장)
CC-13	CleaningController	TurnOffThenTurnOn	호출 순서: `allTurnOff` → `allTurnOn` (순서 보장)
CC-14	CleaningController	TurnOnThreeTimes	`turnOn` ×3 → `allTurnOn` ×3

CI Test


All jobs
1. Build
2. clang-tidy
3. Mock Tests
4. Real Device Tests with TCP Server
5. Coverage
Run details
Usage
Workflow file
Run mock tests
0.00 sec
Start 88: SweepingUnitTest.TurnOffMakesUnitOff
88/95 Test #88: SweepingUnitTest.TurnOffMakesUnitOff ..... Passed
0.00 sec
Start 89: SweepingUnitTest.CleanDoesNothingWhenOff
89/95 Test #89: SweepingUnitTest.CleanDoesNothingWhenOff ..... Passed
0.00 sec
Start 90: SweepingUnitTest.CleanWithDustSendsBoostMode
90/95 Test #90: SweepingUnitTest.CleanWithDustSendsBoostMode ..... Passed
0.00 sec
Start 91: SweepingUnitTest.CleanWithoutDustSendsNormalMode
91/95 Test #91: SweepingUnitTest.CleanWithoutDustSendsNormalMode ..... Passed
0.00 sec
Start 92: SweepingUnitTest.BoostModeDirectlySendsBoostMode
92/95 Test #92: SweepingUnitTest.BoostModeDirectlySendsBoostMode ..... Passed
0.00 sec
Start 93: SweepingUnitTest.NormalModeDirectlySendsNormalMode
93/95 Test #93: SweepingUnitTest.NormalModeDirectlySendsNormalMode ..... Passed
0.00 sec
Start 94: SweepingUnitTest.CleanSwitchesFromBoostToNormal
94/95 Test #94: SweepingUnitTest.CleanSwitchesFromBoostToNormal ..... Passed
0.00 sec
Start 95: SweepingUnitTest.TurnOffPreventsCleanFromSendingCommands
95/95 Test #95: SweepingUnitTest.TurnOffPreventsCleanFromSendingCommands ..... Passed
0.00 sec
100% tests passed, 0 tests failed out of 95

Code Review

project/server/server.cpp

```
8  + #include <unistd.h>
9  + #include <vector>
10 +
11 + static volatile bool g_running = true;
```

bakbakwanwan 31 minutes ago — Member

Server는 Background에서 돌리는게 맞을까요?

likescape reviewed 32 minutes ago — View reviewed changes

project/server/server.cpp

```
210 +     }
211 + };
212 +
213 + std::vector<std::string> RvcSimulator::map = {
```

likescape 32 minutes ago — Member

map 제작 방법 확인했습니다.
문제 없는 거 같습니다

likescape reviewed 47 minutes ago — View reviewed changes

project/src/main.cpp

```
64 +     std::cout << "2: startCleaning\n";
65 +     std::cout << "3: stopCleaning\n";
66 +     std::cout << "4: powerOff\n";
67 +     std::cout << "q: quit\n";
```

likescape 47 minutes ago — Member

q는 use case에서 유저의 act로서 고려되지 않았던거 같은데, 테스트를 위한 것일까요?

project/server/server.cpp

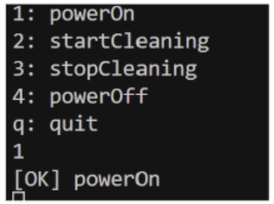
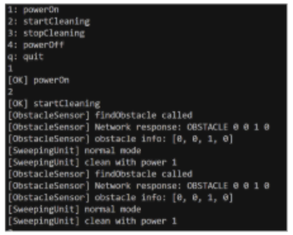
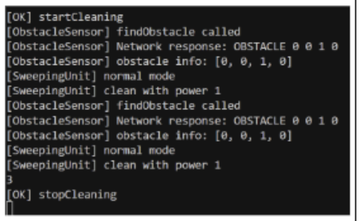
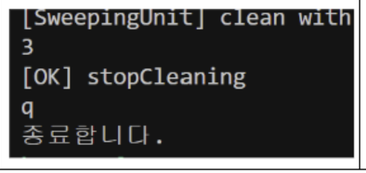
```
8  + #include <unistd.h>
9  + #include <vector>
10 +
11 + static volatile bool g_running = true;
```

bakbakwanwan 32 minutes ago — Member

Server는 Background에서 돌리는게 맞을까요?

System Test

RVC System Test

Test #	Test Scenario	Test Type	Ref #	Actual Result	Expected Result
STC-1	1 입력	pos			power켜기
STC-2	2 입력	neg			power가 켜지지 않아, startcleaning 작동 하지 않음
STC-3	3 입력	neg			startcleaning 중이 아니므로 stopcleaning 작동 하지 않음
STC-4	4 입력	neg			power가 켜지지 않아, power off이 작동하지 않음
STC-5	1 → 2 입력				전원도 켜고 start하였음으로 cleaning을 시작하고 정상적으로 진행되는 로그와 서버의 청소기도 움직임
STC-6	1 → 2 → 3입력				청소가 정상적으로 진행을 멈춤
STC-7	q 입력				ui작동

System Test

STC-8	1 → 2 → 3 ->4 입력			1: powerOn 2: startCleaning 3: stopCleaning 4: powerOff q: quit 1 [OK] powerOn 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 1 1 0 [ObstacleSensor] obstacle info: [0, 1, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 3 [OK] stopCleaning 4 [OK] powerOff	전원이 켜지고 클린 작업을 하고 stop하고 power 꺼짐

STC-9	1 -> 2			==== MAP ==== OOOOOOOOOOOOOOOOOOOO O . O OO O OOO . O OO O O v . OO O . O OO O . OO O O . OO O O OO O . . OO OOOOOOOOOOOOOOOOOOOO ========== [REQUEST] FIND_DUST [RESPONSE] DUST 0 [SweepingUnit] boost mode [SweepingUnit] clean with power 2	청소중 먼지 발견시 boost모드로 진입

STC-10	1 → 2 → 3 → 2 입력			[OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 3 [OK] stopCleaning 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode	
STC-11	4면이 다 막힌 map에서 1 → 2 입력			1: powerOn 2: startCleaning 3: stopCleaning 4: powerOff q: quit 1 [OK] powerOn 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 1 1 [ObstacleSensor] obstacle info: [1, 1, 1, 1] 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 1 1 [ObstacleSensor] obstacle info: [1, 1, 1, 1] 3	

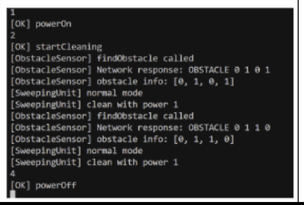
System Test

STC-8	1 → 2 → 3 ->4 입력			1: powerOn 2: startCleaning 3: stopCleaning 4: powerOff q: quit 1 [OK] powerOn 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 1 1 0 [ObstacleSensor] obstacle info: [0, 1, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 3 [OK] stopCleaning 4 [OK] powerOff	전원이 켜지고 클린 작업을 하고 stop하고 power 꺼짐

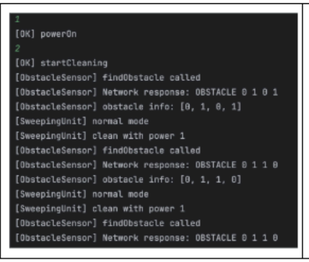
STC-9	1 -> 2			==== MAP ==== OOOOOOOOOOOOOOOOOOOO O . O OO O OOO . O OO O O v . OO O . O OO O . OO O O . OO O O OO O . . OO OOOOOOOOOOOOOOOOOOOO ========== [REQUEST] FIND_DUST [RESPONSE] DUST 0 [SweepingUnit] boost mode [SweepingUnit] clean with power 2	청소중 먼지 발견시 boost모드로 진입

STC-10	1 → 2 → 3 → 2 입력			[OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 3 [OK] stopCleaning 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode [SweepingUnit] clean with power 1 [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 0 0 1 0 [ObstacleSensor] obstacle info: [0, 0, 1, 0] [SweepingUnit] normal mode	
STC-11	4면이 다 막힌 map에서 1 → 2 입력			1: powerOn 2: startCleaning 3: stopCleaning 4: powerOff q: quit 1 [OK] powerOn 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 1 1 [ObstacleSensor] obstacle info: [1, 1, 1, 1] 2 [OK] startCleaning [ObstacleSensor] findObstacle called [ObstacleSensor] Network response: OBSTACLE 1 1 [ObstacleSensor] obstacle info: [1, 1, 1, 1] 3	

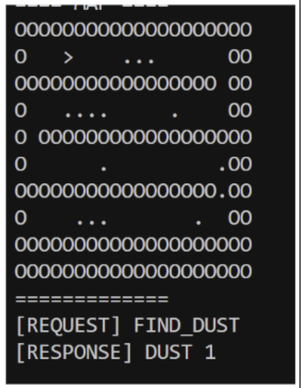
System Test

STC-12	1 → 4 입력				
STC-13	1 → 2 → 4 입력				

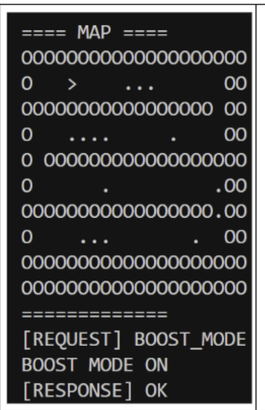
```
1
[OK] powerOn
2
[OK] startCleaning
[ObstacleSensor] findObstacle called
[ObstacleSensor] Network response: OBSTACLE 0 1 0 1
[ObstacleSensor] obstacle info: [0, 1, 0, 1]
[SweepingUnit] normal mode
[SweepingUnit] clean with power 1
[ObstacleSensor] findObstacle called
[ObstacleSensor] Network response: OBSTACLE 0 1 1 0
[ObstacleSensor] obstacle info: [0, 1, 1, 0]
[SweepingUnit] normal mode
[SweepingUnit] clean with power 1
4
[OK] powerOff
```

STC-15	1 → 2 입력 후 map에 먼지 다 치워짐				꾸준히 이동

```
1
[OK] powerOn
2
[OK] startCleaning
[ObstacleSensor] findObstacle called
[ObstacleSensor] Network response: OBSTACLE 0 1 0 1
[ObstacleSensor] obstacle info: [0, 1, 0, 1]
[SweepingUnit] normal mode
[SweepingUnit] clean with power 1
[ObstacleSensor] findObstacle called
[ObstacleSensor] Network response: OBSTACLE 0 1 1 0
[ObstacleSensor] obstacle info: [0, 1, 1, 0]
[SweepingUnit] normal mode
[SweepingUnit] clean with power 1
[ObstacleSensor] findObstacle called
[ObstacleSensor] Network response: OBSTACLE 0 1 1 0
```

STC-16	server에서 dust감지후 boost모드로 변경 요청받음				

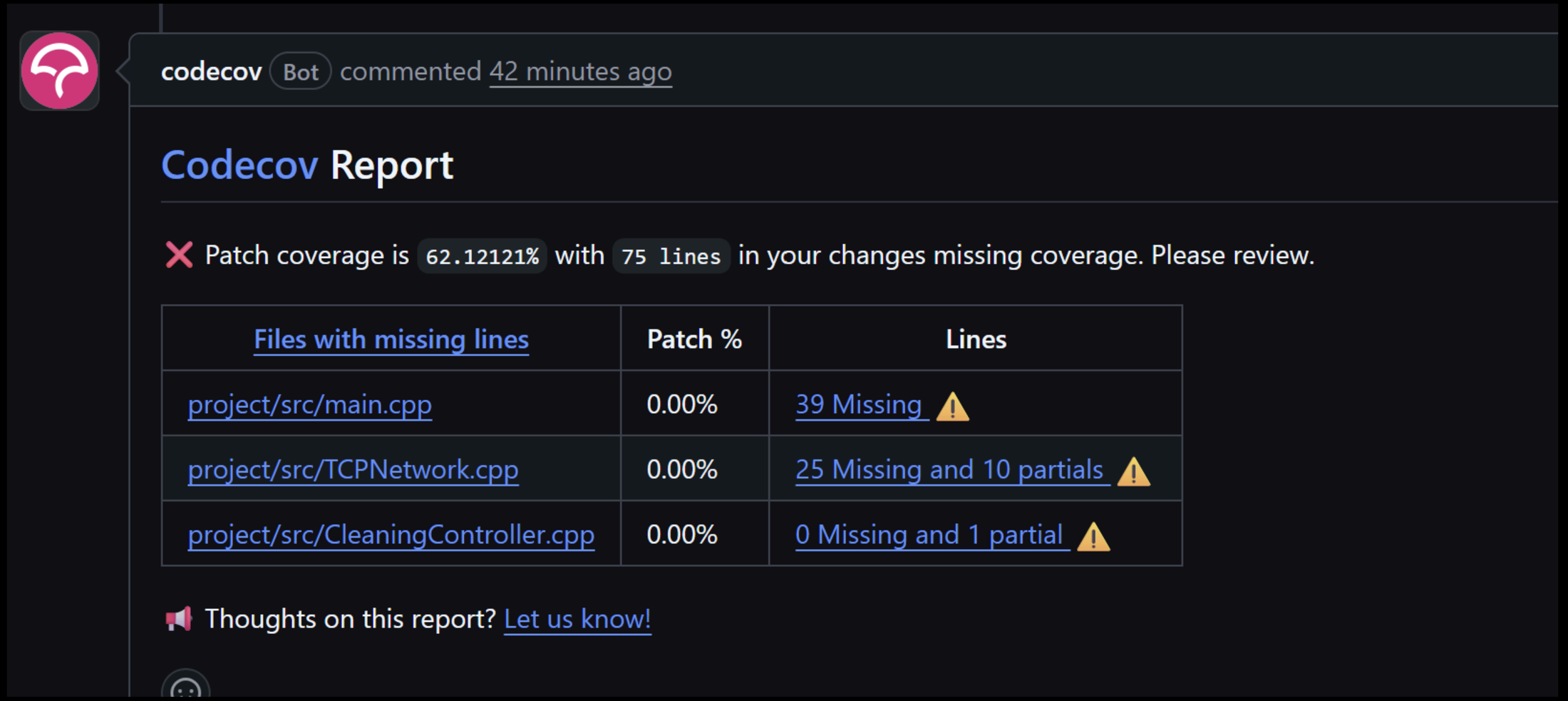
```
OOOOOOOOOOOOOOOOOOOO
O   >     ...      OO
OOOOOOOOOOOOOOOOOO OO
O    ....     .    OO
O OOOOOOOOOOOOOOOOOO
O        .         .OO
OOOOOOOOOOOOOOOOOO.OO
O     ...        .  OO
OOOOOOOOOOOOOOOOOOOO
OOOOOOOOOOOOOOOOOOOO
=============
[REQUEST] FIND_DUST
[RESPONSE] DUST 1
```

```
==== MAP ====
OOOOOOOOOOOOOOOOOOOO
O   >     ...      OO
OOOOOOOOOOOOOOOOOO OO
O    ....     .    OO
O OOOOOOOOOOOOOOOOOO
O        .         .OO
OOOOOOOOOOOOOOOOOO.OO
O     ...        .  OO
OOOOOOOOOOOOOOOOOOOO
OOOOOOOOOOOOOOOOOOOO
=============
[REQUEST] BOOST_MODE
BOOST MODE ON
[RESPONSE] OK
```

STC-17	잘못된 입력				

Coverage



codecov Bot commented 42 minutes ago

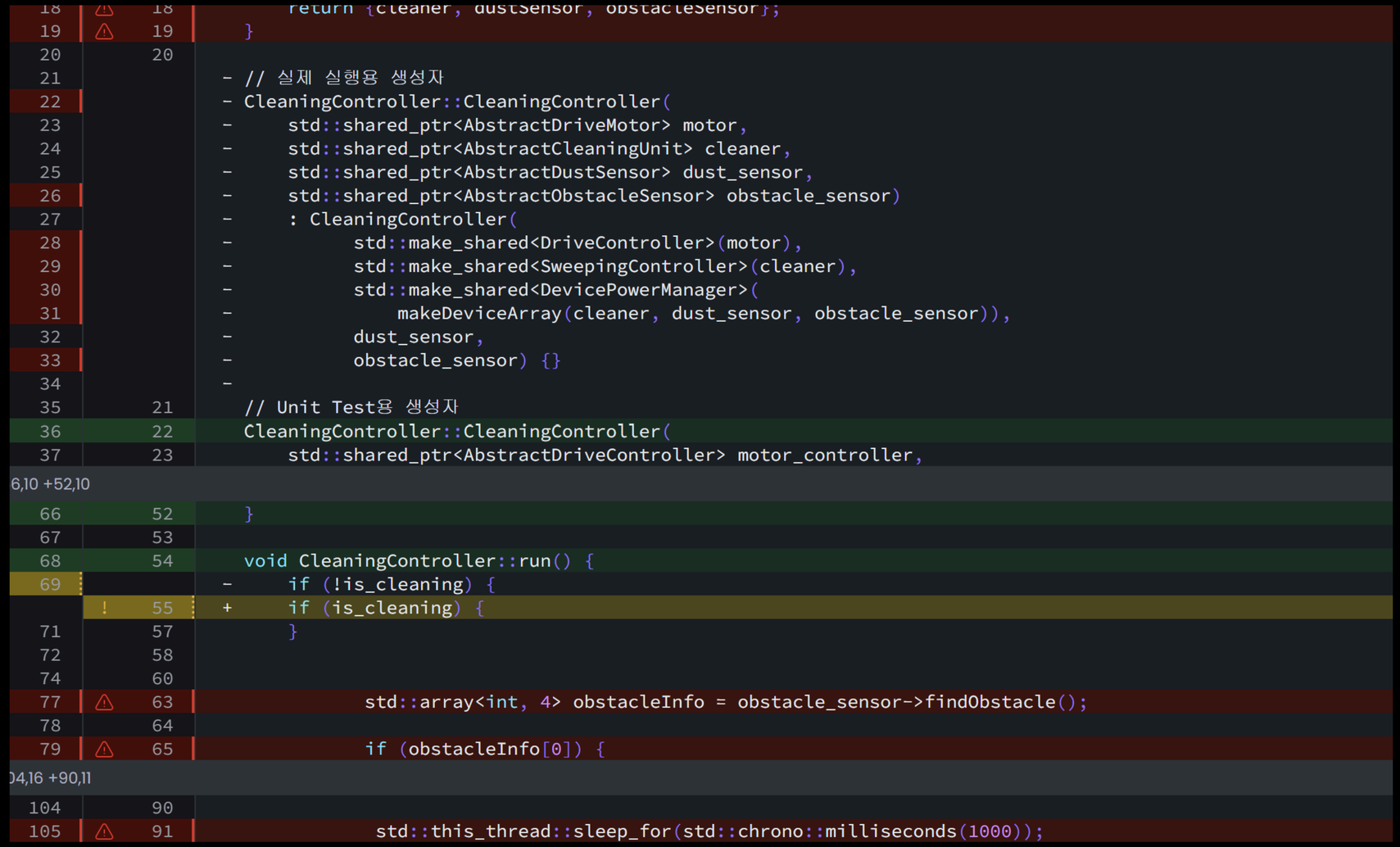
Codecov Report

❌ Patch coverage is `62.12121%` with `75 lines` in your changes missing coverage. Please review.

Files with missing lines	Patch %	Lines
project/src/main.cpp	0.00%	39 Missing ⚠️
project/src/TCPNetwork.cpp	0.00%	25 Missing and 10 partials ⚠️
project/src/CleaningController.cpp	0.00%	0 Missing and 1 partial ⚠️

📢 Thoughts on this report? Let us know!

Coverage


return {cleaner, dustSensor, obstacleSensor};
}
// 실제 실행용 생성자
CleaningController::CleaningController(
std::shared_ptr<AbstractDriveMotor> motor,
std::shared_ptr<AbstractCleaningUnit> cleaner,
std::shared_ptr<AbstractDustSensor> dust_sensor,
std::shared_ptr<AbstractObstacleSensor> obstacle_sensor)
: CleaningController(
std::make_shared<DriveController>(motor),
std::make_shared<SweepingController>(cleaner),
std::make_shared<DevicePowerManager>(
makeDeviceArray(cleaner, dust_sensor, obstacle_sensor)),
dust_sensor,
obstacle_sensor) {}
// Unit Test용 생성자
CleaningController::CleaningController(
std::shared_ptr<AbstractDriveController> motor_controller,
6,10 +52,10
}
void CleaningController::run() {
if (!is_cleaning) {
if (is_cleaning) {
}
std::array<int, 4> obstacleInfo = obstacle_sensor->findObstacle();
if (obstacleInfo[0]) {
04,16 +90,11
std::this_thread::sleep_for(std::chrono::milliseconds(1000));

Coverage

project/src/ObstacleSensor.cpp	0	81.08%	100.00%	+6.08%
project/tests/DriveControllerTest.cpp	0	96.55%	-	0.00%
project/tests/SweepingUnitTest.cpp	0	98.25%	100.00%	+1.69%
project/tests/DustSensorTest.cpp	0	98.85%	100.00%	+2.13%
project/src/DustSensor.cpp	0	94.74%	100.00%	+1.40%
project/src/include/AbstractNetwork.hpp	0	100.00%	-	0.00%
project/tests/ObstacleSensorTest.cpp	0	98.89%	100.00%	+2.01%
project/tests/RVCTest.cpp	0	99.00%	-	0.00%
project/tests/DriveMotorTest.cpp	0	97.78%	100.00%	+2.13%
project/tests/CleangingControllerTest.cpp	0	90.00%	100.00%	+10.00%
project/src/DriveMotor.cpp	0	100.00%	100.00%	0.00%
project/src/SimulatorClient.cpp Deleted file	-	-	-	-
project/src/SweepingUnit.cpp	0	100.00%	100.00%	0.00%

Coverage

project/src/main.cpp	39	0.00%	0.00%	0.00%
project/src/TCPNetwork.cpp	25	0.00%	0.00%	0.00%
project/src/CleaningController.cpp	1	55.17%	0.00%	+14.59%
project/src/ObstacleSensor.cpp	0	81.08%	100.00%	+6.08%
project/tests/DriveControllerTest.cpp	0	96.55%	-	0.00%
project/tests/SweepingUnitTest.cpp	0	98.25%	100.00%	+1.69%
project/tests/DustSensorTest.cpp	0	98.85%	100.00%	+2.13%
project/src/DustSensor.cpp	0	94.74%	100.00%	+1.40%
project/src/include/AbstractNetwork.hpp	0	100.00%	-	0.00%
project/tests/ObstacleSensorTest.cpp	0	98.89%	100.00%	+2.01%
project/tests/RVCTest.cpp	0	99.00%	-	0.00%
project/tests/DriveMotorTest.cpp	0	97.78%	100.00%	+2.13%
project/tests/CleangingControllerTest.cpp	0	90.00%	100.00%	+10.00%
project/src/DriveMotor.cpp	0	100.00%	100.00%	0.00%

Static Code Analysis

project/server/server.cpp:70:5: warning: [modernize-use-nodiscard]

> function 'backOf' should be marked [[nodiscard]]

```
   70 |     Direction backOf(Direction d) const {
      |     ^
      |     [[nodiscard]]
```

project/server/server.cpp:70:15: warning: [modernize-use-trailing-return-type]

> use a trailing return type for this function

```
   70 |     Direction backOf(Direction d) const {
      |               ^
note: this fix will not be applied because it overlaps with another fix
```

project/server/server.cpp:70:15: warning: [readability-convert-member-functions-to-static]

> method 'backOf' can be made static

```
   70 |     Direction backOf(Direction d) const {
      |               ^                   ~~~~~
      |     static
```

- **project/server/server.cpp:195:18:** warning: [readability-identifier-length]

  > loop variable name 'y' is too short, expected at least 2 characters

```
  195 |         for (int y = 0; y < static_cast<int>(map.size()); y++)
      |                     ^
```

- **project/server/server.cpp:196:22:** warning: [readability-identifier-length]

  > loop variable name 'x' is too short, expected at least 2 characters

```
  196 |             for (int x = 0; x < static_cast<int>(map[y].size()
      |                         ^
```

Summary

- System Test – 전원을 키고 청소를 키고, 끄고, 다시 킴 (강병완)
  - 테스트 : 전원을 키고 끄는 키는 과정에서 비정상 작동을 확인
  - 발견 : RVC 전원을 끄는 과정에서 문제가 없음을 확인

- SweepingUnitTest — 전원 꺼진 상태에서 네트워크 명령 전송 (강현준)
  - 테스트 코드: 전원 끈 상태에서 clean(true), clean(false) 호출
  - 발견: isOn() 체크 없이 boostMode() 내부의 StrictMock이 예상치 못한 send() 호출을 검출, 테스트 실패
  - 수정: clean() 진입부에 if (!isOn()) return 가드 추가

- DriveMotorTest — null network 시 crash (박완)
  - 테스트 코드: network 없이 기본 생성자로 DriveMotor 생성 후 전 메서드 호출
  - 발견: m_network가 null인 상태에서 send() 호출 시 에러 발생
  - 수정: 각 메서드 내부에 if (!m_network) return 가드 추가

Thank you!