

# 객체지향개발방법론

## VIBE CODING

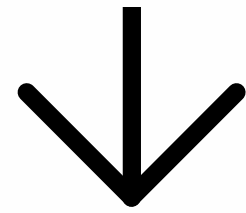
# INDEX

목차페이지

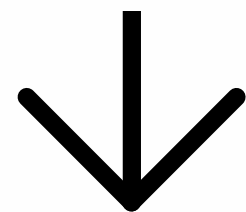
- 01 **작업과정**
- 02 **SRS**
- 03 **USE CASE**
- 04 **SDD**
- 05 **Code**
- 06 **Simulator**
- 07 **SA**
- 08 **CI/CD**
- 09 **검증 과정 예**
- 10 **후기**

## ✓ 작업 과정

PROMPT로  
작업 요청



결과물 최소  
3회 검수 요청



결과물



별도의 hook, command의  
설정은 없음



직접 검수

수정요청

# ✓ SRS



We need to make and System Requirement Specification about RVC Controller or Roomba. You should meet the following conditions for this document. please make it in md file, and include Functional Requirments and Non Functional Requirments.  
...(조건들 설명)

```
### 3.2.1 Object: AutomaticCleaningSession

#### 3.2.1.1 Attributes (direct or inherited)

| ID | Attribute | Description |
|----|-----|-----|
| AT-1.1 | `sessionActive` | Boolean: automatic cleaning session is on or off |
| AT-1.2 | `sessionSource` | Optional: user, schedule, remote TBD |

#### 3.2.1.2 Functions (services / methods)

| ID | Functional requirement |
|----|-----|
| OBJ1-FR-1 | When `sessionActive` becomes true, the Software Controller SHALL enable the automatic cleaning behavior consistent with 3.2.2 (SurfaceCleaningController) and 3.2.4 (NavigationAndEscapeCoordinator). (Legacy) |
| OBJ1-FR-2 | When `sessionActive` is false, the Software Controller SHALL NOT perform autonomous obstacle avoidance or autonomous forward cruise described in 3.2.2 through 3.2.4. (Legacy FR-016) |
| OBJ1-FR-3 | Starting and stopping a session SHALL be deterministic with respect to the session interface (no half TBD in design). |

#### 3.2.1.3 Messages (communications received or sent)

| Direction | Message | Description |
|-----|-----|-----|
| Received | `StartSession` | Asserts `sessionActive` |
| Received | `StopSession` | Negates `sessionActive` |
| Received | `ResumeSession` | Resumes a paused session if the product defines pause/resume; policy TBD (see UC- |
| Received | `requestServiceOrReset` | User service/reset intent when a persistent fault does not recover; handling |
| Sent | (internal) | Notifies SurfaceCleaningController and NavigationAndEscapeCoordinator of session sta
```

# ✓ USE-CASE



Next, please make use case which goes along with SRS you just made. Make it in Casual form and give me md file

```
## UC-07 📄 Resume forward cleaning after a maneuver

**Use Case Name**
Resume forward cleaning after a maneuver

**Actor**
- **Primary:** **Obstacle sensor subsystem** (reports that **forward** is again **clear** / safe per fusion rules).
- **Supporting:** **Drive motors / wheel subsystem** (execute forward command).
- **Supporting:** **Cleaning hardware** (resume or maintain cleaning on).
- **Supporting:** **Home user** (beneficiary).

**Purpose**
After a turn or backup escape completes and the path ahead is viable, return to **forward** travel with **cleaning** on. Do not remain in maneuver state indefinitely.

**Overview**
This use case often appears as the **tail** of UC-03, UC-04, or UC-05: once **obstacle sensors** and fusion say the path is safe, the **system** prefers forward progress and normal cleaning.

**Cross Reference**
SRS 3.2.4 OB34-FR-2, OB34-FR-3, OB34-FR-6; 3.2.2 OB32-FR-1. Legacy: FR-001, FR-004, FR-007.

**Pre-Requisites**
- Session active.
- A maneuver (avoidance or surrounded escape) has completed or forward clearance is granted per fusion **TBD**.
- Forward path evaluated **safe**.

**Typical Courses of Events**
1. **Obstacle sensors** + fusion indicate forward **safe** after maneuver.
2. **System** commands **straight forward** on **drive motors** (OB34-FR-2).
3. **System** ensures **cleaning hardware** is **active** at appropriate power (OB32-FR-1) unless another rule suspends it.
4. Continues as **UC-02**.
```

# ✓ UseCase

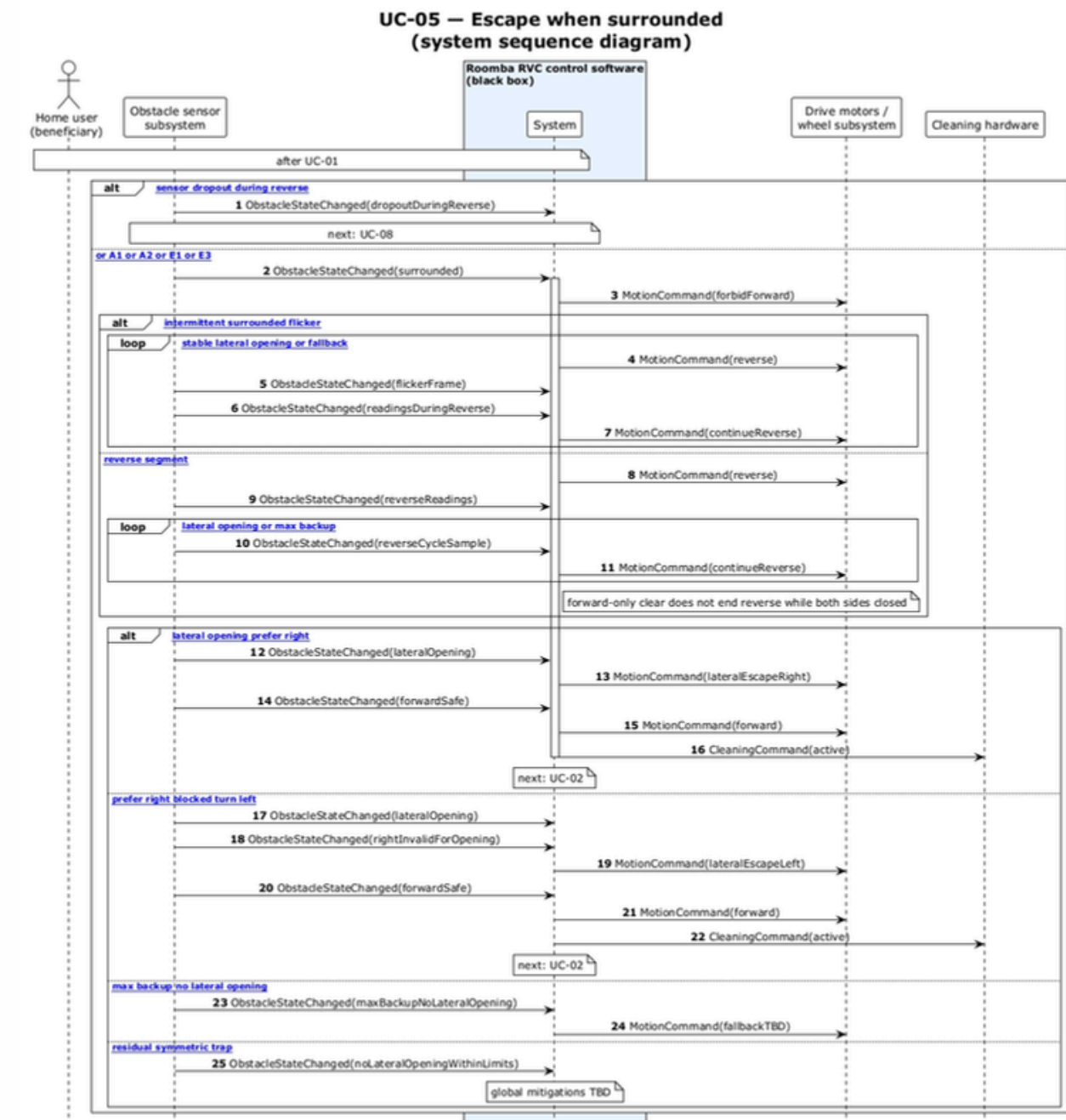


1. Start automatic cleaning session	2. Cruise forward while cleaning	3. Avoid obstacle (forward blocked, not surrounded)
4. Avoid obstacle when right turn blocked	5. Escape when surrounded	6. Boost cleaning when dust is high
7. Resume forward cleaning after a maneuver	8. Handle missing / invalid / stale obstacle data	9. Build consistent fused obstacle picture

# ✓ SDD-SSD



Next, please make System Sequence Diagram which goes along with SRS and Use Case you just made.  
...(조건 설명)

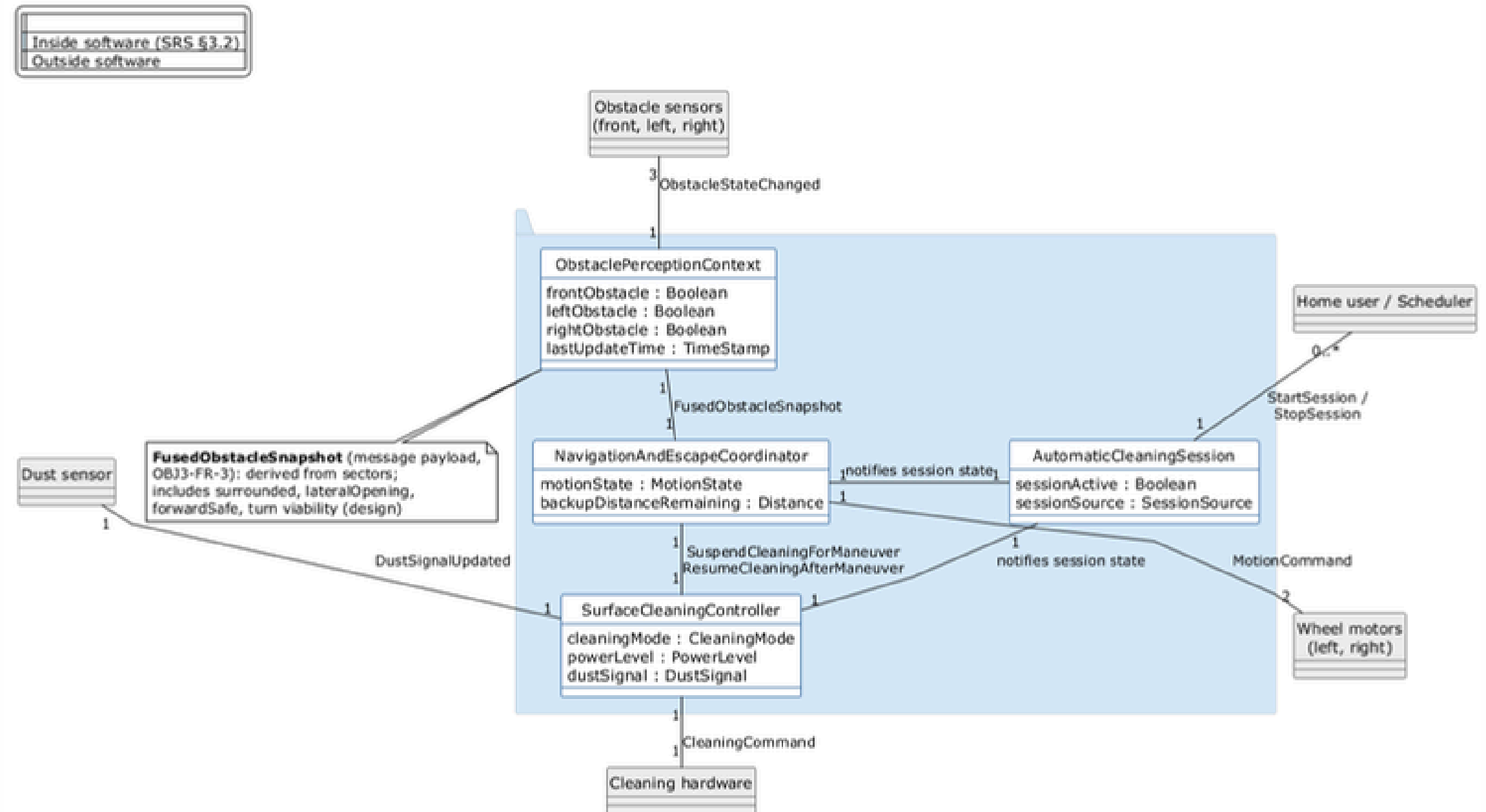


# ✓ SDD-Domain



**Roomba RVC – Domain model  
(names & attributes = SRS §3.2)**

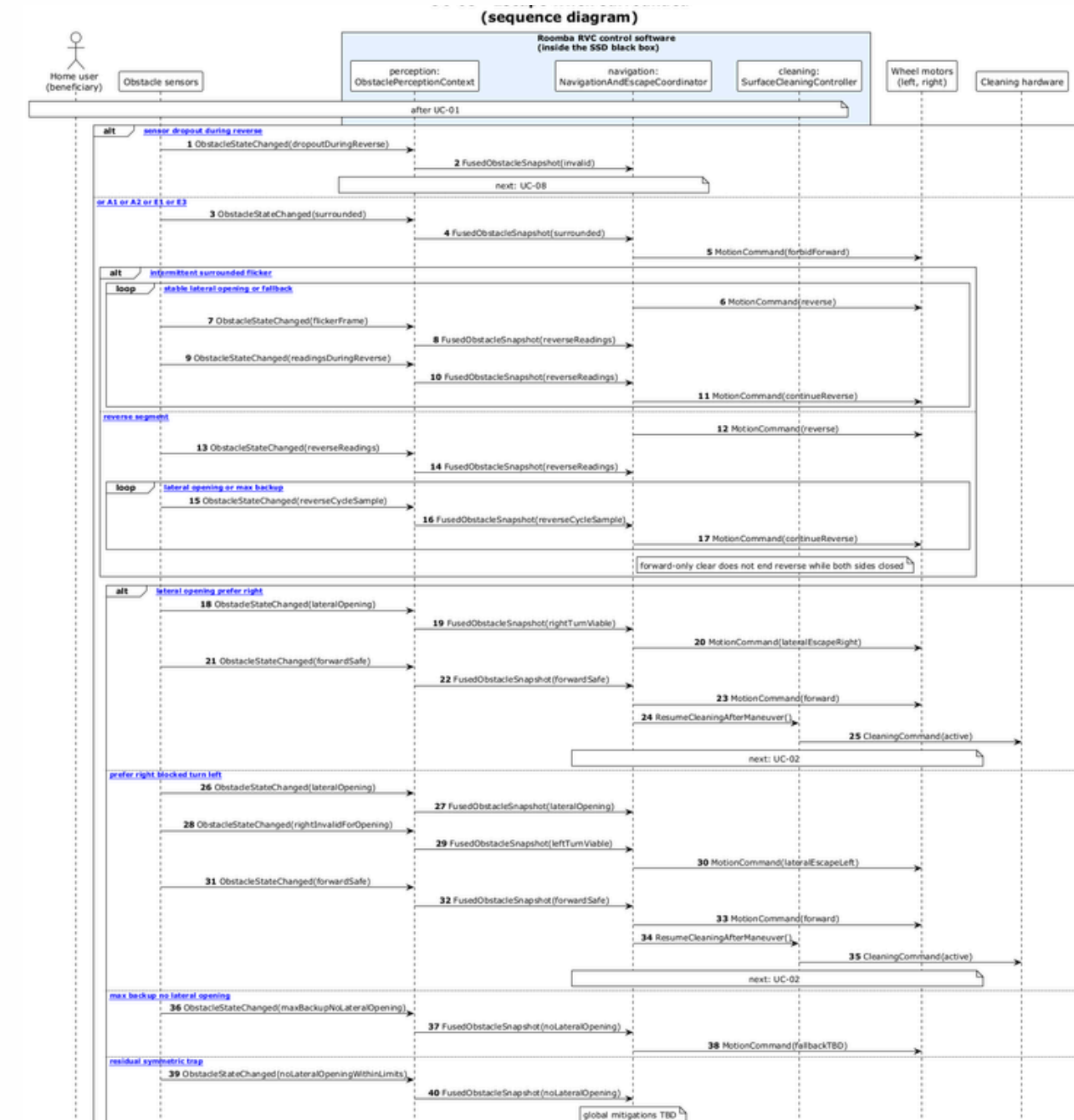
Next, please make Domain Model Diagram based on SRS, Use Case, and System Sequence Diagram.



# ✓ SDD-SD

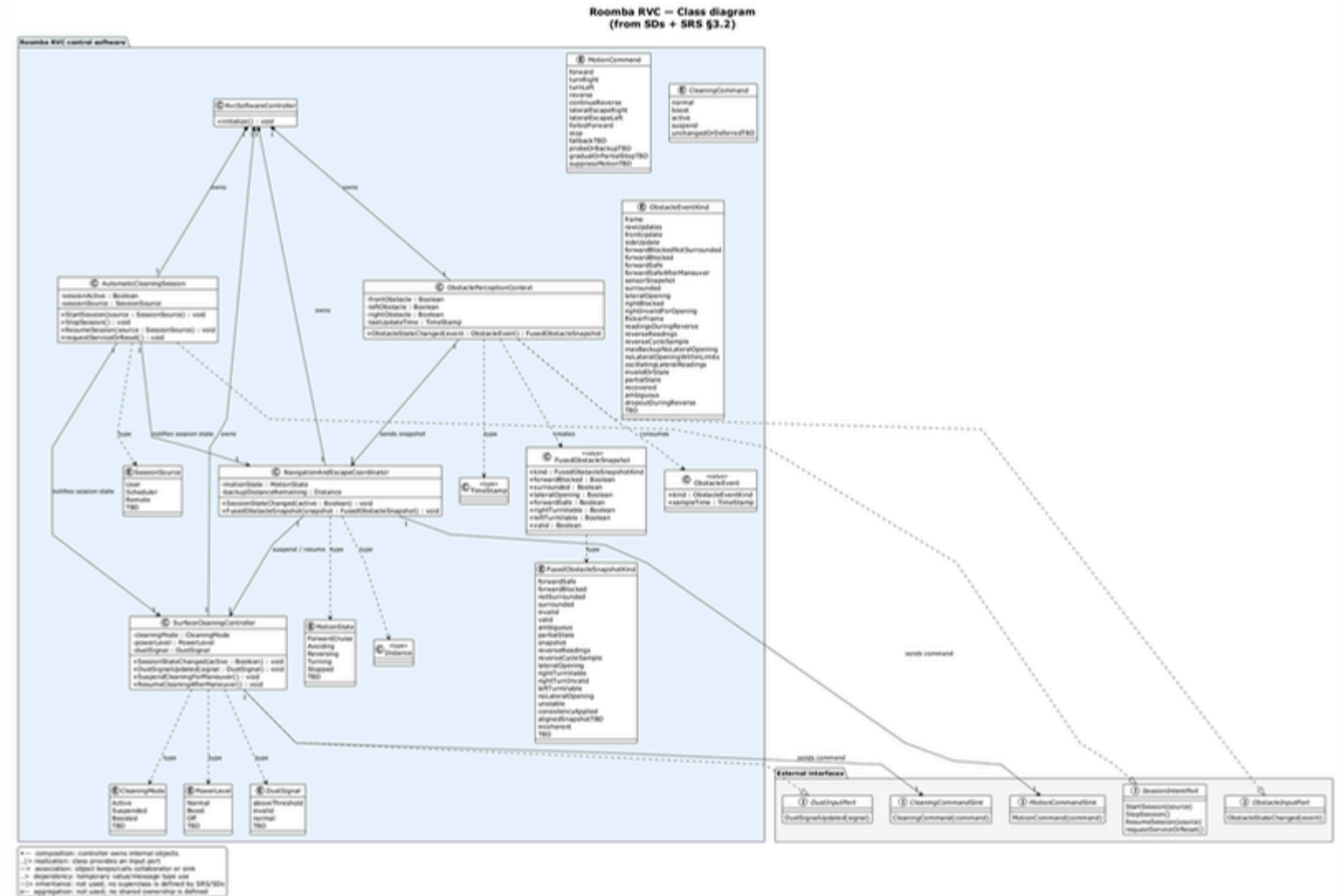


Next, please make  
Sequence Diagram based  
on System Sequence  
Diagram



# ✓ SDD-CLASS

Next, please make Class Diagram based on Sequence Diagram and Domain Model Diagram you just made?



# ✓ Code



Alright! Now would you make code based on Sequence Diagram and Class Diagram? Please keep the operation name and logic same as the documents we made so far.

```
EXPLORER
SRC
  AutomaticCleaningSessio...
  NavigationAndEscapeCoo...
  ObstaclePerceptionConte...
  RvcSoftwareController.cpp
  SurfaceCleaningControlle...

NavigationAndEscapeCoordinator.cpp
90 void NavigationAndEscapeCoordinator::handleSurrounded(const rvc::FusedObstacleSnap
100     if (snapshot.kind == FusedObstacleSnapshotKind::reverseReadings ||
102         motionState_ = MotionState::Reversing;
103         send(MotionCommand::continueReverse);
104         return;
105     }
106
107     if (snapshot.kind == FusedObstacleSnapshotKind::rightTurnViable || snapshot.ri
108         motionState_ = MotionState::Turning;
109         send(MotionCommand::lateralEscapeRight);
110         return;
111     }
112
113     if (snapshot.kind == FusedObstacleSnapshotKind::leftTurnViable || snapshot.lef
114         motionState_ = MotionState::Turning;
115         send(MotionCommand::lateralEscapeLeft);
116         return;
117     }
118
119     if (snapshot.kind == FusedObstacleSnapshotKind::noLateralOpening) {
120         send(MotionCommand::fallbackTBD);
121         return;
122     }
123
124     send(MotionCommand::probeOrBackupTBD);
125 }
```

# ✓ UT



```
ObstaclePerceptionContextTest.cpp > ...
1  #include "rvc/NavigationAndEscapeCoordinator.hpp"
6
7  #include <gtest/gtest.h>
8
9  namespace {
10
11  struct ObstaclePerceptionContextFixture {
12      rvc_test::RecordingMotionSink motionSink;
13      rvc_test::RecordingCleaningSink cleaningSink;
14      rvc::SurfaceCleaningController cleaning{cleaningSink};
15      rvc::NavigationAndEscapeCoordinator navigation{motionSink, cleaning};
16      rvc::ObstaclePerceptionContext perception{navigation};
17  };
18
19  } // namespace
20
21  TEST(ObstaclePerceptionContextTest, ObstacleStateChangedMapsSafeFrameToForwardSafeSnapshot) {
22      ObstaclePerceptionContextFixture fixture;
23
24      const auto snapshot = fixture.perception.ObstacleStateChanged({rvc::ObstacleEventKind::frame, 1});
25
26      EXPECT_EQ(snapshot.kind, rvc::FusedObstacleSnapshotKind::forwardSafe);
27      EXPECT_TRUE(snapshot.forwardSafe);
28      EXPECT_TRUE(snapshot.valid);
29  }
30
31  TEST(ObstaclePerceptionContextTest, ObstacleStateChangedMapsSurroundedEventToSurroundedSnapshot) {
32      ObstaclePerceptionContextFixture fixture;
33
34      const auto snapshot = fixture.perception.ObstacleStateChanged({rvc::ObstacleEventKind::surrounded, 2});
35
36      EXPECT_EQ(snapshot.kind, rvc::FusedObstacleSnapshotKind::surrounded);
37      EXPECT_TRUE(snapshot.forwardBlocked);
38      EXPECT_TRUE(snapshot.surrounded);
39  }
```

Great! Now would you make Google test for Unit testing our code?

# ✓ UT



```
PS C:\Users\cheky\Cursor> ctest --test-dir cpp/build --output-on-failure
Test project C:/Users/cheky/Cursor/cpp/build
  Start 1: AutomaticCleaningSessionTest.StartSessionEnablesNavigationAndCleaningPolicies
1/39 Test #1: AutomaticCleaningSessionTest.StartSessionEnablesNavigationAndCleaningPolicies ..... Passed 0.
  Start 2: AutomaticCleaningSessionTest.StopSessionStopsMotionAndSuspendsCleaning
2/39 Test #2: AutomaticCleaningSessionTest.StopSessionStopsMotionAndSuspendsCleaning ..... Passed 0.
  Start 3: AutomaticCleaningSessionTest.ResumeSessionEnablesPoliciesAfterStop
3/39 Test #3: AutomaticCleaningSessionTest.ResumeSessionEnablesPoliciesAfterStop ..... Passed 0.
  Start 4: AutomaticCleaningSessionTest.RequestServiceOrResetStopsMotionAndSuspendsCleaning
4/39 Test #4: AutomaticCleaningSessionTest.RequestServiceOrResetStopsMotionAndSuspendsCleaning ..... Passed 0.
  Start 5: NavigationAndEscapeCoordinatorTest.SessionStateChangedActiveEnablesPolicyWithoutImmediateCommand
5/39 Test #5: NavigationAndEscapeCoordinatorTest.SessionStateChangedActiveEnablesPolicyWithoutImmediateCommand ... Passed 0.
  Start 6: NavigationAndEscapeCoordinatorTest.SessionStateChangedInactiveStopsMotion
6/39 Test #6: NavigationAndEscapeCoordinatorTest.SessionStateChangedInactiveStopsMotion ..... Passed 0.
  Start 7: NavigationAndEscapeCoordinatorTest.InvalidSnapshotStopsMotionAndSuspendsCleaning
7/39 Test #7: NavigationAndEscapeCoordinatorTest.InvalidSnapshotStopsMotionAndSuspendsCleaning ..... Passed 0.
  Start 8: NavigationAndEscapeCoordinatorTest.PartialStaleSnapshotUsesPartialStopCommand
8/39 Test #8: NavigationAndEscapeCoordinatorTest.PartialStaleSnapshotUsesPartialStopCommand ..... Passed 0.
  Start 9: NavigationAndEscapeCoordinatorTest.SurroundedSnapshotForbidsForwardThenStartsReverse
9/39 Test #9: NavigationAndEscapeCoordinatorTest.SurroundedSnapshotForbidsForwardThenStartsReverse ..... Passed 0.
  Start 10: NavigationAndEscapeCoordinatorTest.ReverseReadingsContinueReverse
10/39 Test #10: NavigationAndEscapeCoordinatorTest.ReverseReadingsContinueReverse ..... Passed 0.
```

## GCC Code Coverage Report

Directory: ./

Coverage: low: ≥ 0% medium: ≥ 75.0% high: ≥ 90.0%

	Coverage	Exec / Excl / Total
Lines:	93.6%	264 / 0 / 282
Functions:	97.3%	36 / 0 / 37
Branches:	79.6%	90 / 0 / 113

### List of functions

File	Lines	Functions	Branches
<a href="#">include/rvc/Ports.hpp</a>	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100.0% 5 / 0 / 5	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100.0% 5 / 0 / 5	<div style="width: 0%;"><div style="width: 0%;"></div></div> -% 0 / 0 / 0
<a href="#">src/AutomaticCleaning...</a>	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100.0% 24 / 0 / 24	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100.0% 6 / 0 / 6	<div style="width: 0%;"><div style="width: 0%;"></div></div> -% 0 / 0 / 0
<a href="#">src/NavigationAndEsc...</a>	<div style="width: 96.8%;"><div style="width: 96.8%;"></div></div> 96.8% 90 / 0 / 93	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100.0% 8 / 0 / 8	<div style="width: 81.1%;"><div style="width: 81.1%;"></div></div> 81.1% 60 / 0 / 74
<a href="#">src/ObstaclePerceptio...</a>	<div style="width: 94.2%;"><div style="width: 94.2%;"></div></div> 94.2% 81 / 0 / 86	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100.0% 3 / 0 / 3	<div style="width: 86.4%;"><div style="width: 86.4%;"></div></div> 86.4% 19 / 0 / 22
<a href="#">src/RvcSoftwareContro...</a>	<div style="width: 90.9%;"><div style="width: 90.9%;"></div></div> 90.9% 20 / 0 / 22	<div style="width: 88.9%;"><div style="width: 88.9%;"></div></div> 88.9% 8 / 0 / 9	<div style="width: 50.0%;"><div style="width: 50%;"></div></div> 50.0% 3 / 0 / 6
<a href="#">src/SurfaceCleaningCo...</a>	<div style="width: 84.6%;"><div style="width: 84.6%;"></div></div> 84.6% 44 / 0 / 52	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100.0% 6 / 0 / 6	<div style="width: 72.7%;"><div style="width: 72.7%;"></div></div> 72.7% 8 / 0 / 11

Would you run Code Coverage Test for it?

# ✓ Simulator



```
Running 30 scripted scenarios
Use --verbose to print actions and command traces for every scenario.

[PASS] TC-01 Initialize enters safe inactive state
[PASS] TC-02 Start and cruise forward while cleaning
[PASS] TC-03 Dust boost returns to normal while active
[PASS] TC-04 Dust boost is deferred while session is inactive
[PASS] TC-05 Invalid dust signal keeps normal cleaning
[PASS] TC-06 Stop session stops motion and suspends cleaning
[PASS] TC-07 Resume after stop re-enters forward cleaning
[PASS] TC-08 Service/reset request enters inactive safe state
[PASS] TC-09 Invalid obstacle data triggers full fault stop
[PASS] TC-10 Partial stale obstacle data uses gradual stop
[PASS] TC-11 Recovery snapshot is accepted without new command
[PASS] TC-12 Forward blocked not surrounded chooses right turn
[PASS] TC-13 Right blocked chooses left turn
[PASS] TC-14 Oscillating lateral readings suppress motion
[PASS] TC-15 Surrounded starts reverse escape
[PASS] TC-16 Reverse readings continue backing up
[PASS] TC-17 Reverse cycle sample continues reverse segment
[PASS] TC-18 Lateral opening escapes right
[PASS] TC-19 Right invalid for opening escapes left
[PASS] TC-20 Max backup without opening uses fallback
[PASS] TC-21 No lateral opening within limits uses fallback
[PASS] TC-22 Dropout during reverse enters UC-08 style stop
[PASS] TC-23 Ambiguous obstacle snapshot waits for later policy
[PASS] TC-24 Raw updates build consistency snapshot only
[PASS] TC-25 Asynchronous side update aligns snapshot only
[PASS] TC-26 Forward safe after maneuver resumes cleaning
[PASS] TC-27 Forward blocked after maneuver trends to obstacle avoidance
[PASS] TC-28 Going back then turning: reverse before right escape
[PASS] TC-29 Going back then left turning after right opening invalid
[PASS] TC-30 Full mission path with boost, trap escape, resume, stop

Summary: 30 / 30 scenarios passed.
```

```
SIMULATOR [P] [C] [M]
rvc_simulator.cpp 2
rvc_simulator.cpp > ...
10 namespace {
205 std::string describeActions(const std::vector<Action>& actions) {
207     for (std::size_t i = 0; i < actions.size(); ++i) {
212     }
213     return result;
214 }
215
216 void runActions(const std::vector<Action>& actions, rvc::RvcSoftwareController& controller) {
217     rvc::TimeStamp time = 1;
218     for (const auto& action : actions) {
219         switch (action.kind) {
220             case ActionKind::Initialize:
221                 controller.initialize();
222                 break;
223             case ActionKind::Start:
224                 controller.sessionIntentPort().StartSession(rvc::SessionSource::User);
225                 break;
226             case ActionKind::Stop:
227                 controller.sessionIntentPort().StopSession();
228         }
229     }
230 }
```

Awesome. Now would you make Simulator to test the program? I would like 30 test case for it.

✓ SA



Summary

All jobs

- Static Analysis
- Build, Test, Simulate

Run details

- Usage
- Workflow file

### Static Analysis

succeeded 5 days ago in 26s

Search logs

- Set up job
- Checkout repository
- Install static analysis tools
- Run cppcheck

```
1 Run cppcheck \  
12 Checking cpp/src/AutomaticCleaningSession.cpp ...  
13 1/5 files checked 8% done  
14 Checking cpp/src/NavigationAndEscapeCoordinator.cpp ...  
15 2/5 files checked 44% done  
16 Checking cpp/src/ObstaclePerceptionContext.cpp ...  
17 3/5 files checked 77% done  
18 Checking cpp/src/RvcSoftwareController.cpp ...  
19 4/5 files checked 85% done  
20 Checking cpp/src/SurfaceCleaningController.cpp ...  
21 5/5 files checked 100% done
```

#### Post Checkout repository

```
1 Post job cleanup.  
2 /usr/bin/git version  
3 git version 2.54.0  
4 Temporarily overriding HOME='/home/runner/work/_temp/07ba2106-5fe6-4342-80b1-60d9ab307e05' before making global git config changes  
5 Adding repository directory to the temporary git global config as a safe directory  
6 /usr/bin/git config --global --add safe.directory /home/runner/work/OOAD-with-AI/OOAD-with-AI  
7 /usr/bin/git config --local --name-only --get-regexp core.sshCommand
```

Finally, would you make  
CI/CD in github? Please  
include at least one Static  
Analysis.

# ✓ CI/CD



Deilise / OOAD-with-AI

<> Code Issues Pull requests **Actions** Projects Wiki Security and quality Insights Settings

← C++ CI

## ✓ Limit static analysis to RVC source code #2

Summary

All jobs

- ✓ Static Analysis
- ✓ Build, Test, Simulate

Run details

- Usage
- Workflow file

Triggered via push 5 days ago

✓ Deilise pushed → 12eed09 main

Status

Success

Total duration

1m 0s

Artifacts

2

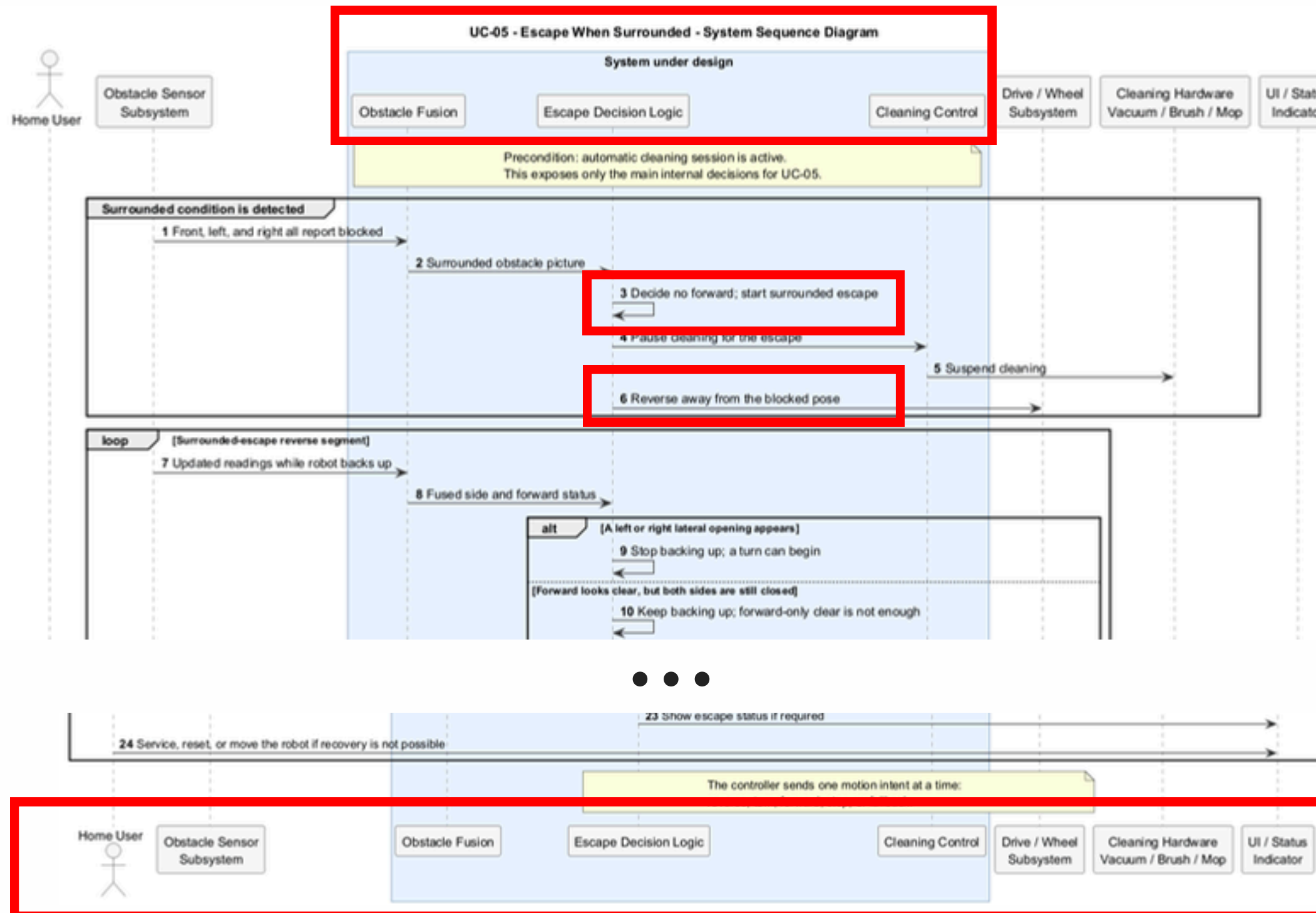
cpp-ci.yml

on: push

✓ Static Analysis 26s

✓ Build, Test, Simulate 28s

# ✓ 검증 과정 - ssd 5 예시

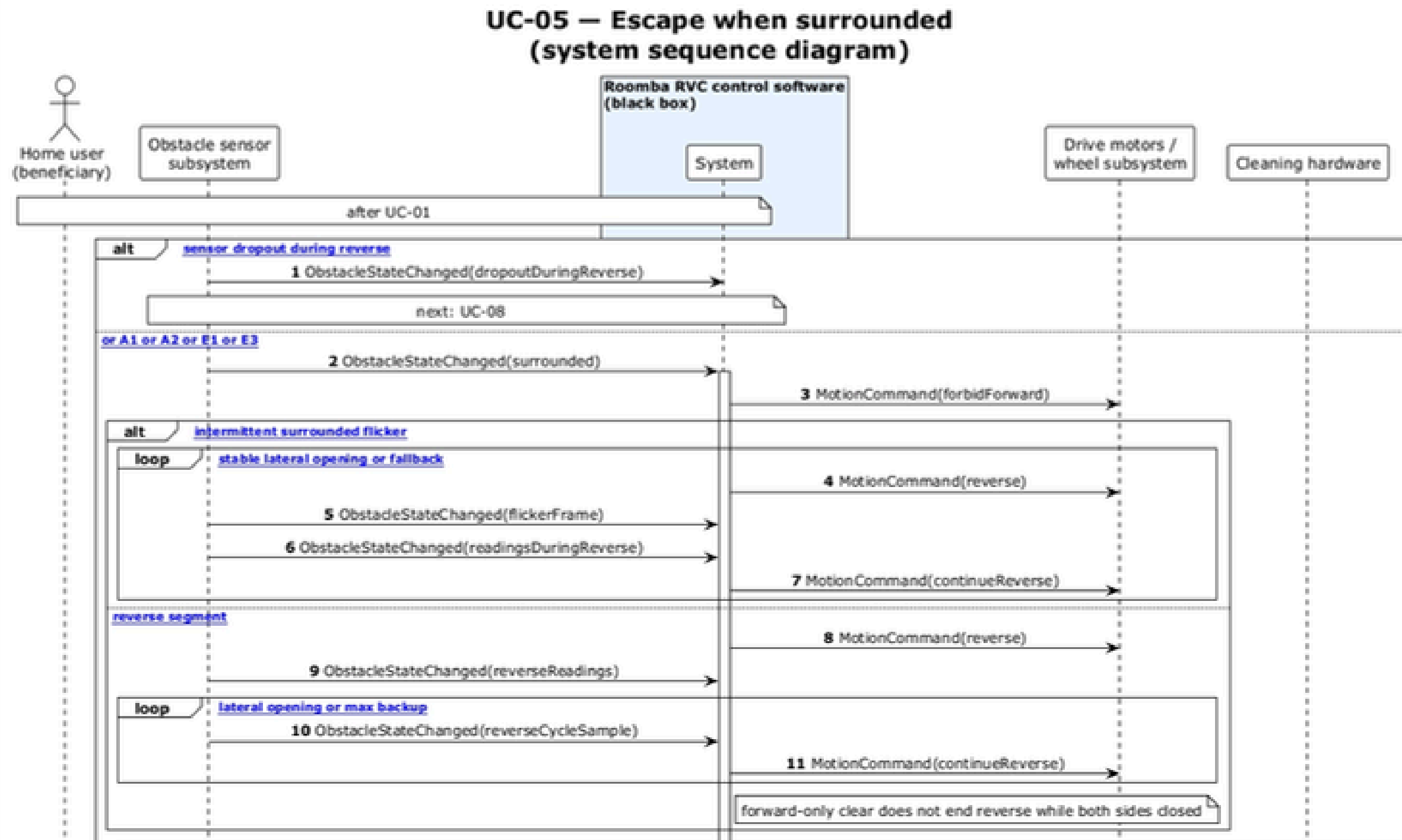


I need the SSD's System part to be black boxed.

Also, you need to make the descriptions on the arrow to be Operation names and arguments, not explanation of what it is doing.

Finally, remove the bottom part of the participants and what is going on inside of the objects.

# ✓ 검증 과정 - ssd 5 예시



# ✓ 검증 과정 - Use Case, Domain Diagram 예



## ## UC-01 - Start Cleaning

Someone starts the robot, either by pressing a button, using an app, or a voice command. The controller marks the cleaning session as active. From this point, the robot can perform various cleaning situations.

If the session is not active, none of the automatic cleaning behaviors are performed.

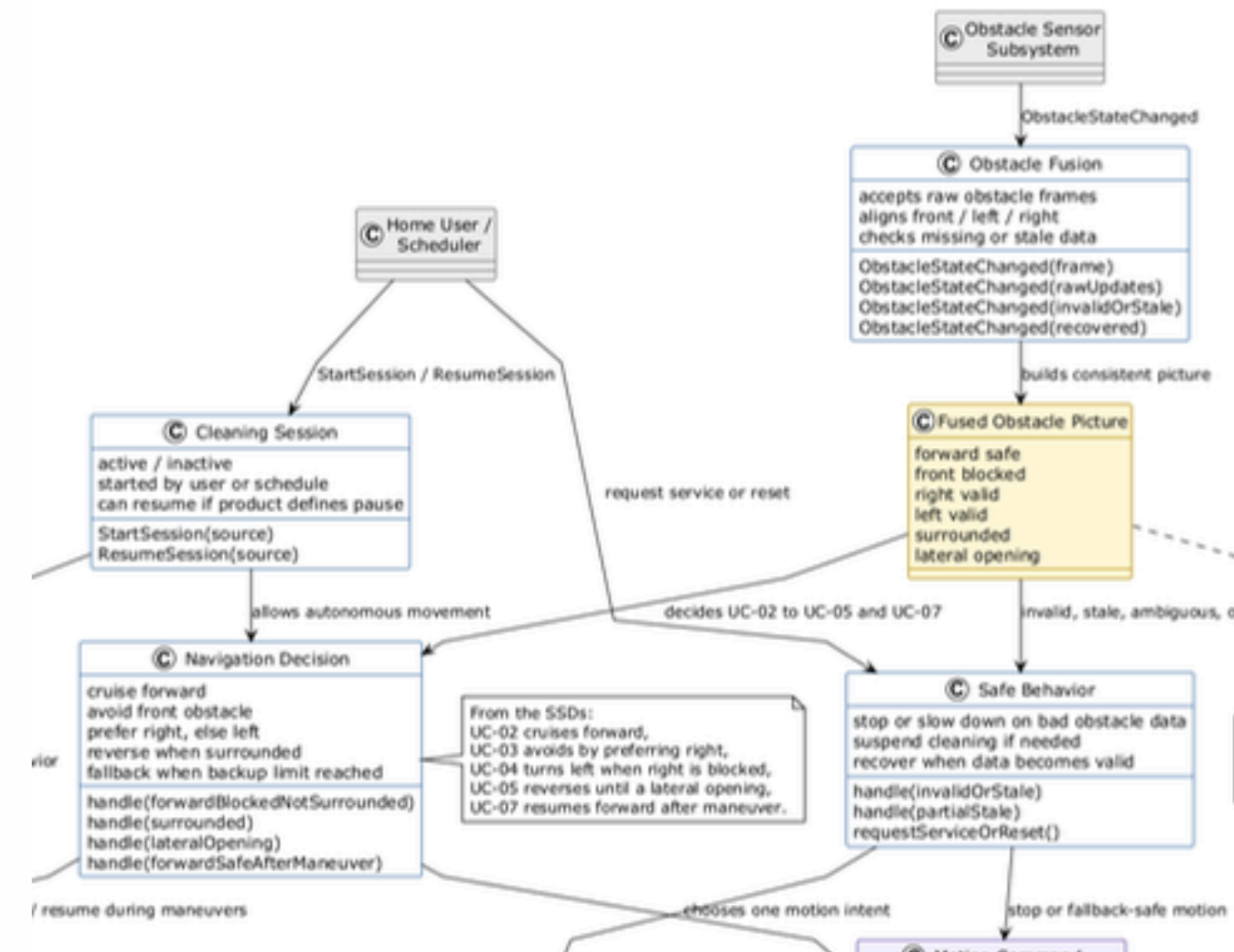
## ## UC-02 - Move Forward While Cleaning

When cleaning is active and nothing is blocking the robot, the controller allows autonomous movement.

This is the normal cruising behavior. The robot stays in this mode until it receives new sensor data.

## ## UC-03 - Avoid a Front Obstacle

Roomba RVC - Domain Diagram from System Sequence Diagrams



# ✓ 검증 과정 - Use Case, Domain Diagram 예

## ## UC-01 – Start automatic cleaning session

### \*\*Use Case Name\*\*

Start automatic cleaning session

### \*\*Actor\*\*

- **Primary:** **Home user** (starts cleaning from UI or phys
  - **Primary (alternate):** **Scheduler / clock** (if the prod
  - **Supporting:** **UI / panel firmware** (delivers start/sto
- (The cleaning control **software** is the **system**, not an*

### \*\*Purpose\*\*

Put the robot into **automatic cleaning** mode so vacuum/mop

### \*\*Overview\*\*

The user or an automated schedule requests a cleaning session (obstacle handling, dust boost) until the session is stopped.

### \*\*Cross Reference\*\*

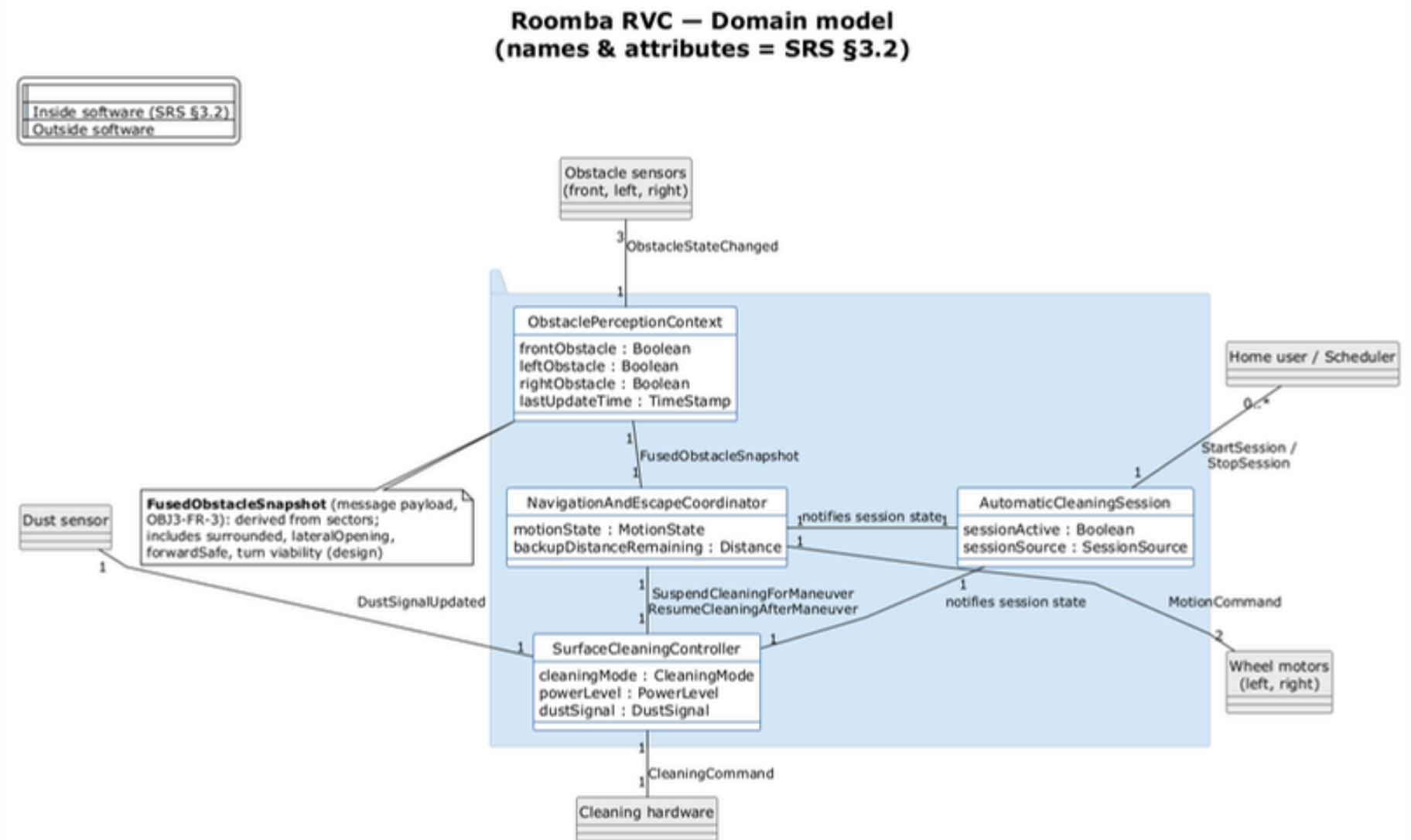
SRS §3.2.1 **AutomaticCleaningSession** (OBJ1-FR-1, OBJ1-FR-2, **FR-001**); §3.1.1 UI-1, UI-2. Legacy: FR-001, FR-015, FR-01

### \*\*Pre-Requisites\*\*

- Robot is powered and not in a factory-locked state that for
- Session **stop** is not asserted.

### \*\*Typical Courses of Events\*\*

1. **Home user** (or **Scheduler**) issues start-cleaning int



# ✓ 후기



## 장점

1. 툴을 몰라도 바로 시작 가능
2. 커서는 프롬프트 옆에 cmd랑 파일 탐색기가 함께 있어 편리
3. plant uml, 렌더러 다운 등 프로그램 다운도 해줌

## 단점

1. 생각보다 맥락, 설명이 필요
2. 채팅 형식, 글씨체와 작은 글씨크기 때문에 과거 내용 다시 찾기 어려움

## ✓ 후기

### 소감

1. 정확한 결과물이 아닌, 요청한 만큼의 그럴싸한 결과물만 냄. 프로젝트 진행 시, 이론과 필요한 내용/과정을 꺾고 있어야겠다 느낌
2. 그래도 지적 1번만 해도 잘 고침
3. AI가 대신 해주니 사용자는 중간 중간 놀게 됨. 작업 환경이 집중력이 떨어지기 쉬운 환경이라고 느낌
4. 결과물 검증하는 시간이 더 오래 걸림
5. 성능이 정말 좋아져서 AI 사용법을 더 익혀야겠다는 생각이 듦.