

<오동동>

<Mobile Robot Simulator – ADD_ON>

구현운영 산출물

[요 약]

Mobile Robot Controller 의 ADD_ON 시스템의 구현과 운영 산출물을 기술한다.

주요 산출물로

1. Component Diagram

2. Deployment Diagram

을 작성하였다.

1 개 요

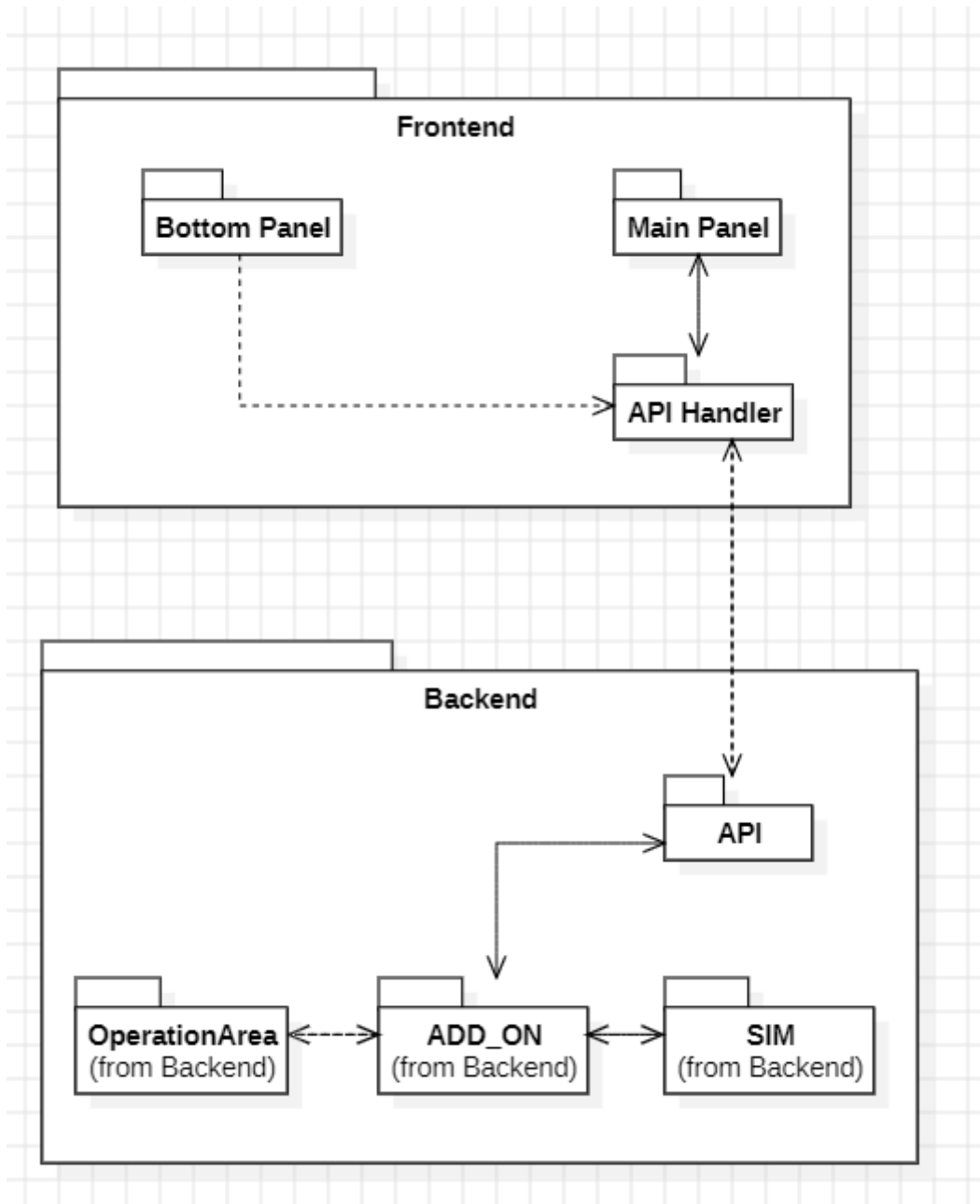
1.1 목 적

본 문서에서는 Mobile Robot Controller 시스템에 대한 설계 산출물을 기술한다.

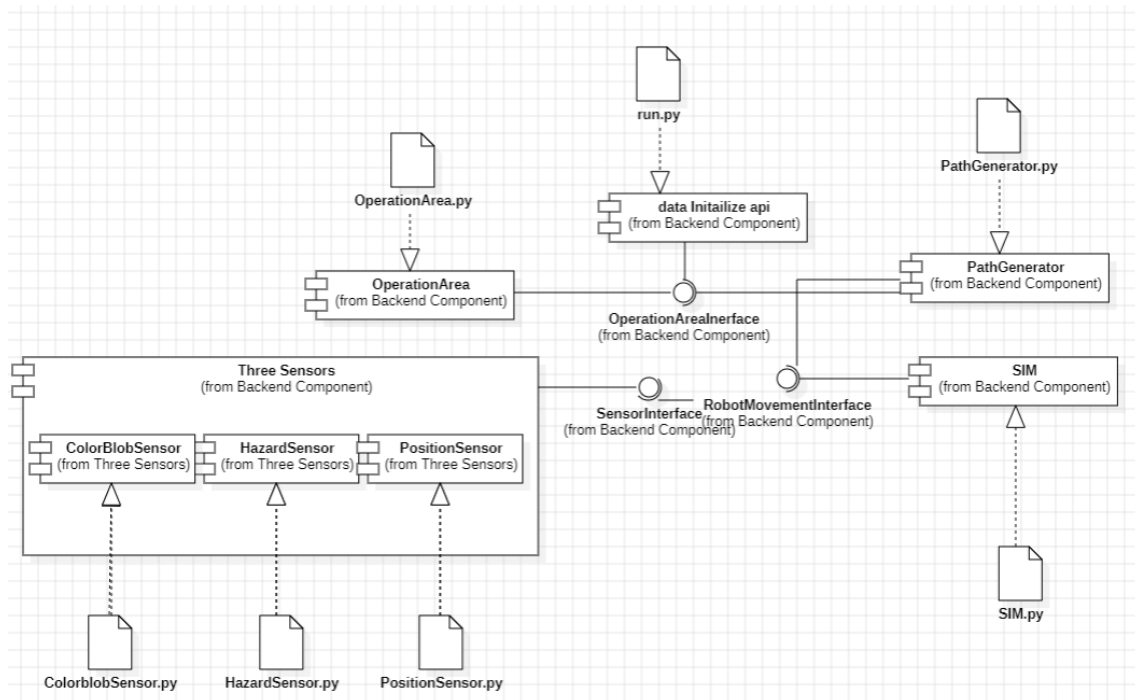
1. **Component Diagram** 을 작성해 컴포넌트 간의 통신을 매개하는 Interface 와, 산출물들을 표현한다.

2. **Deployment Diagram** 을 작성하 시스템의 물리적 구조와 시스템의 배치를 표현한다.

2. Component Diagram



2.1 Backend 컴포넌트 패키지 구성



[OperationArea]

재난 지형 모델들의 데이터를 가지고 있다. 즉, 재난 지형의 크기, 위험 지점, 중요 지역 등을 가지고 있으며, OperationAreaInterface 를 통하여 이 데이터를 다른 컴포넌트에게 제공한다.

OperationArea.py 산출물로 구현된다.

[PathGenerator]

OperationAreaInterface 를 통하여 데이터를 전달받고, 데이터를 처리 한 후, RobotMovementInterface 를 이용하여 SIM 을 컨트롤한다.

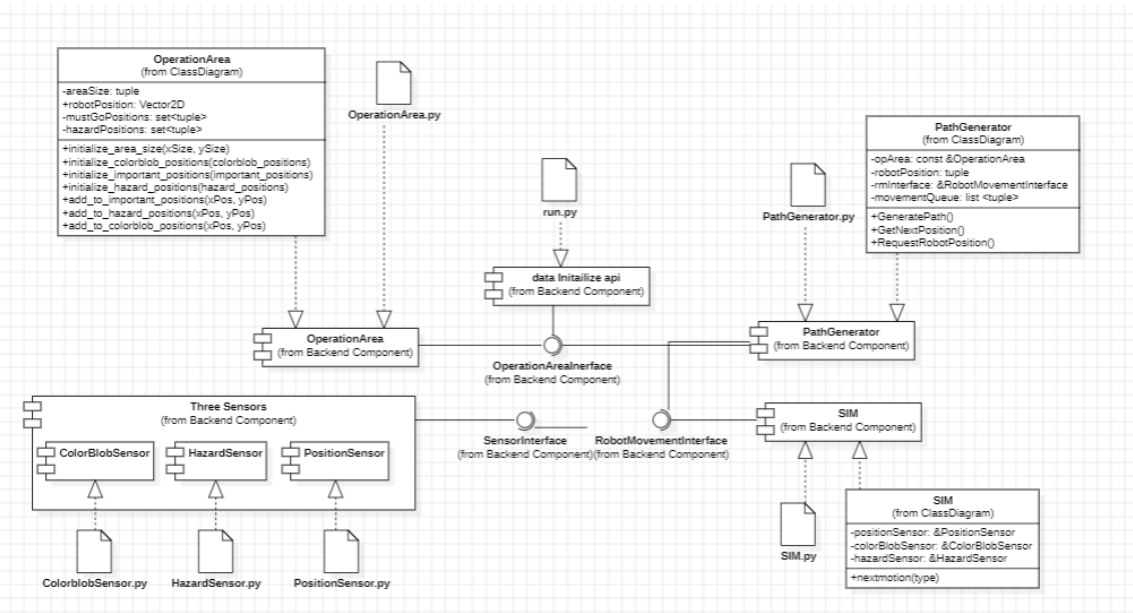
PathGenerator.py 산출물로 구현된다.

[ThreeSensor]

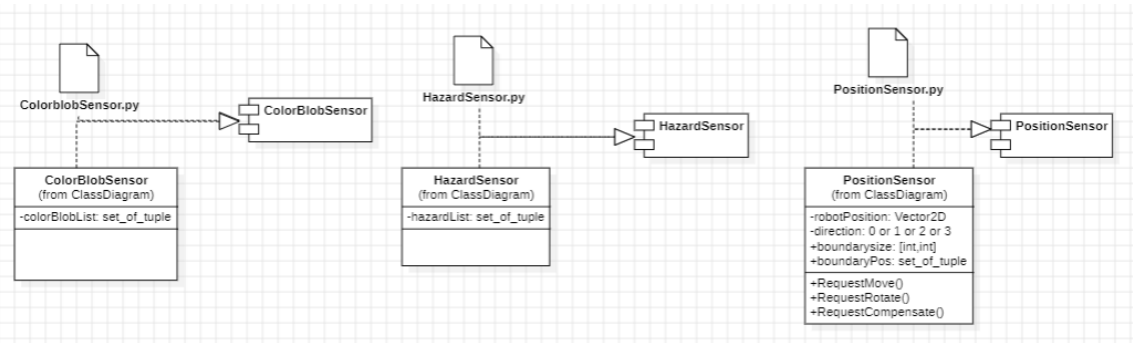
센서들의 동작의 기능을 제공한다. 이 기능은 SensorInterface 를 통해 제공되어진다.

ColorblobSensor.py, HazardSensor.py, PositionSensor.py 산출물로 구현된다.

이들을 바탕으로 컴포넌트들의 클래스를 실현하면 다음과 같이 그려진다.

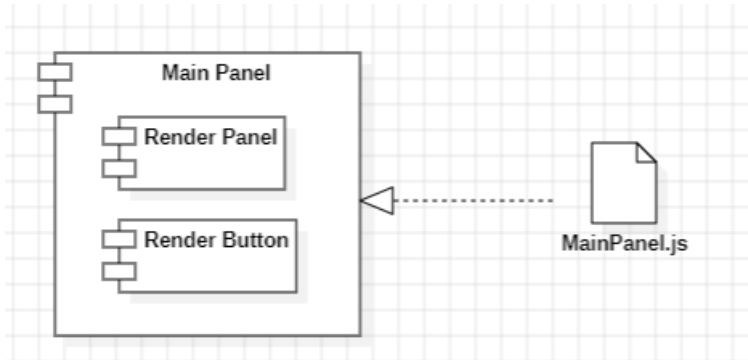


추가로 센서들은 다음과 같이 실현된다.

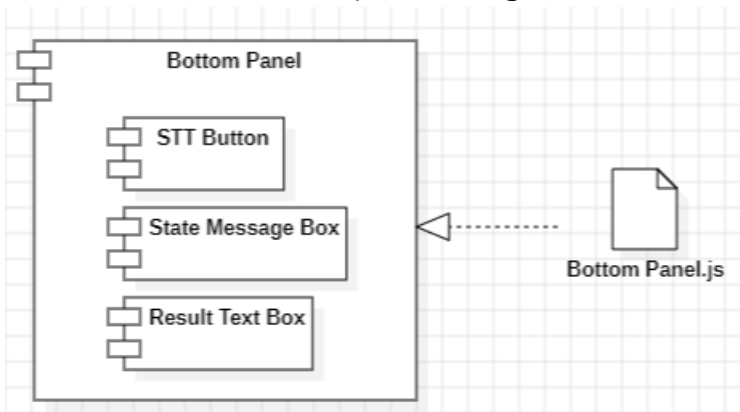


2.2 Frontend 컴포넌트 패키지 구성

2.2.1 MainPanelComponetDiagram



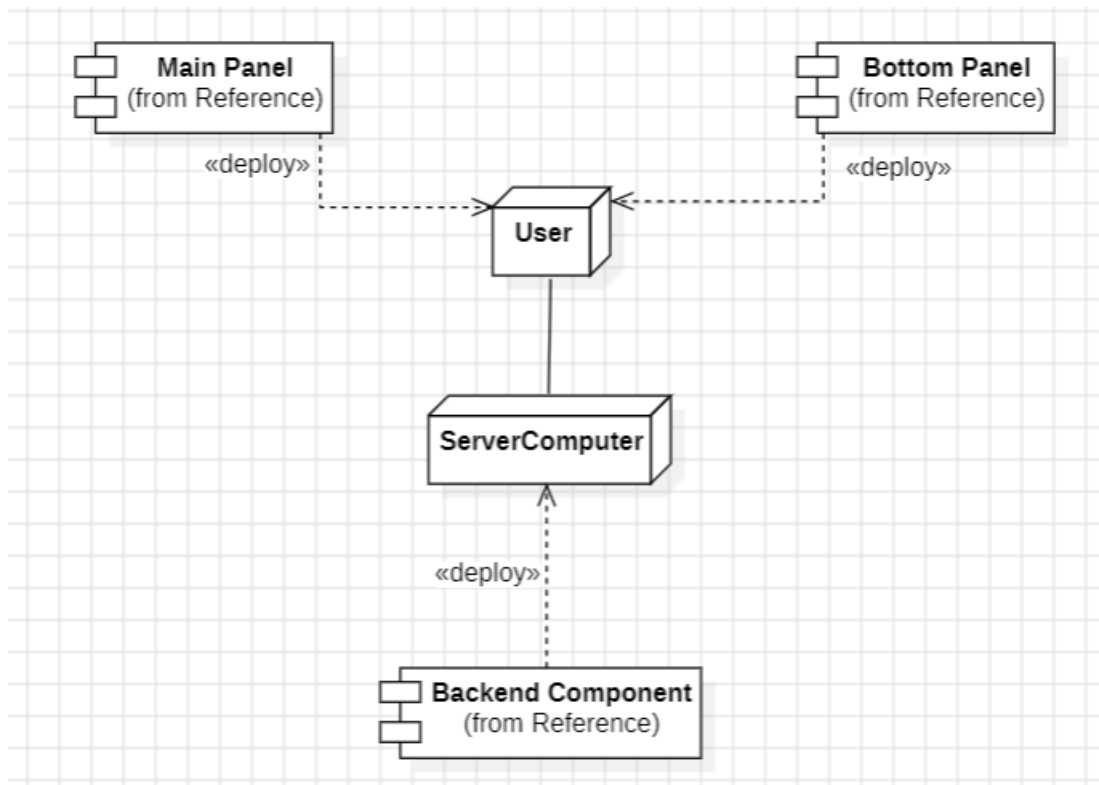
2.2.2 BottomPanelComponentDiagram



프론트엔드 구현에 이용한 React App 은 크게 두 부류 **MainPanel**, **BottomPanel** 로 나누었다.

MainPanel 은 렌더링 및 시각화를 위한 컴포넌트들이 작성되었으며, BottomPanel 은 사용자의 STT 명령을 처리하고 보여주는 컴포넌트들이 작성돼있다.

3. 배치도



Mobile Robot Controller 의 ADD_ON 시스템은 두 개의 컴퓨터에 의해 동작된다.

[User Computer]

User 측에서는 사용자가 시뮬레이션 정보를 입력해 시뮬레이션을 시작할 수 있다. 시뮬레이션 정보는 일정 시간마다 서버와 동기화되며 최신 시뮬레이션 정보들을 화면에 보여준다.

[Server Computer]

Backend Component 를 동작시키기 위한 코드와 프로그램들이 저장되어 있다. User Computer 와 사전에 정의된 API Protocol 의 방식으로 통신을 하여, 사용자로부터 입력을 받으며, Server Computer 에서 이를 연산 한 후,

다시 규약에 맞추어 정보를 반환한다.

[부록]

완성된 제품의 테스트 결과를 기술한다. 테스트 항목으로는

1. TSP 근사 알고리즘 성능 =

[모든 경로 중 최단 거리] / [알고리즘으로 이동한 거리]

이 있다.

Test #1

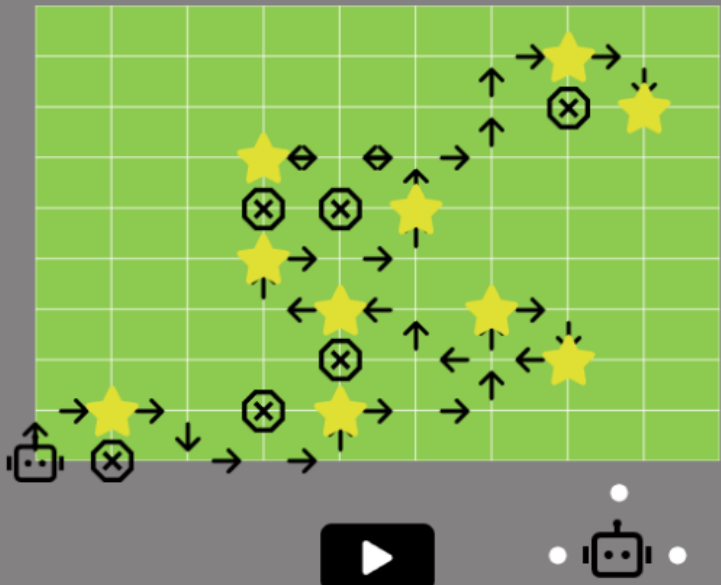


모든 경로에서 계산한 최단 거리 : 23

알고리즘으로 계산한 이동 거리 : 24

성능 : 0.96

Test #2



성능: 0.78

Test #3



[평가]

Test#1 ~ #3 으로부터 얻은 성능의 평균값은 대략 **0.9** 다.

기존 문제인 Open-Revisitable TSP Problem 의 솔루션을 간단히 근사한 것에 비해, 1.0 에 가까운 성능을 보인다. 따라서 **Nearest-Neighbor Algorithm** 으로의 근사는 **상당히 유용**하다고 평가한다.