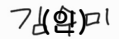



캡스톤디자인 중간보고서

제 목	국문	학교 내 자율주행 로봇 배달 서비스		
	영문	Autonomous robot delivery service in school		
진 행 상 황	중요마일스톤	<p><주요 기능 : 자율 주행 배달 ></p> <ol style="list-style-type: none"> 1. SLAM Mapping: SLAM 작업을 통해 학교 내부의 정밀한 Map를 구축. 2. Localization and Navigation: 현재 위치 파악과 경로 계획 기능을 개발하여, 로봇이 자율적으로 이동할 수 있도록 작업. 3. Server Communication: ROS Bridge와 서버 간의 통신 인프라를 구축하였으며, 실시간 데이터 전송 및 수신 확인. 4. 가상 환경에서의 테스트 진행: 개발된 기능들을 가상 환경(Gazebo) 에서 테스트하여 시스템의 안정성 및 효율성을 검증. 5. Mobile Application Integration: 사용자 인터페이스 개발, 호실 입력 기능을 통해 로봇이 목적지 좌표를 인식하고 정확히 배달을 수행할 수 있도록 생성. 		
	진행상황	<ol style="list-style-type: none"> 1. SLAM 작업 완료: 기존 ROS 패키지의 노드를 활용하여 SLAM 작업을 성공적으로 수행. 이 과정을 통해 학교 내부의 정밀한 지도를 생성 2. Localization & Navigation 작업 완료: SLAM으로 얻은 맵을 기반으로 로봇의 위치 파악 및 경로 계획 작업을 완료. 이를 통해 로봇이 자율적으로 이동할 수 있는 기능을 확보 3. Parameter 수정 작업 완료: 로봇의 최적 성능을 위해 필요한 기존 파라미터들을 수정. 4. 서버 및 통신 인프라 구축: 전용 서버를 구축하고, ROS Bridge를 사용하여 로봇과의 통신 인프라 완료. 5. 가상 환경에서의 테스트 완료: 개발된 기능들을 Gazebo 가상 환경에서 테스트하여, 시스템의 안정성 및 효율성을 확인. 6. YOLO와 SLAM의 정확도 비교 분석: 객체 인식을 위한 YOLO와 SLAM 기술의 정확도를 비교 분석하여, 각 기술의 성능 및 적용 가능성을 평가. 7. 모바일 어플리케이션을 통한 호실 입력 기능 개발 중: 사용자가 모바일 어플리케이션을 통해 원하는 호실을 입력하면, 로봇이 해당 호실을 정확히 찾아가는 기능을 개발 중 		
산출물	요구사항 정의서(별첨 1), 중간보고서(별첨 2)			
팀 구성원	학년	학 번	이 름	연락처(전화번호/이메일)
	4	20191783	이동현	010-7589-8323 / donghoney8323@naver.com
	4	20201727	김수미	010-8235-9619 / ymg12347@gmail.com
	4	20217137	송찬호	010-9975-1941 / thdcksgh8545@gmail.com
<p>컴퓨터공학과의 프로젝트 관리규정에 따라 다음과 같이 요구사항 정의서와 중간보고서를 제출합니다</p> <p style="text-align: center;">2024년 05월 02일</p> <p style="text-align: right;">책임자 : 김수미  지도교수 : 황경호 </p>				

[별첨1]

프로젝트명 : 학교 내 자율주행 로봇 배달 서비스

소프트웨어 요구사항 정의서

Version 1.0

개발 팀원 명(팀리더): 김수미
이동현
송찬호

대표 연락처 : 010-8235-9619

e-mail: ymg12347@gmail.com

목차

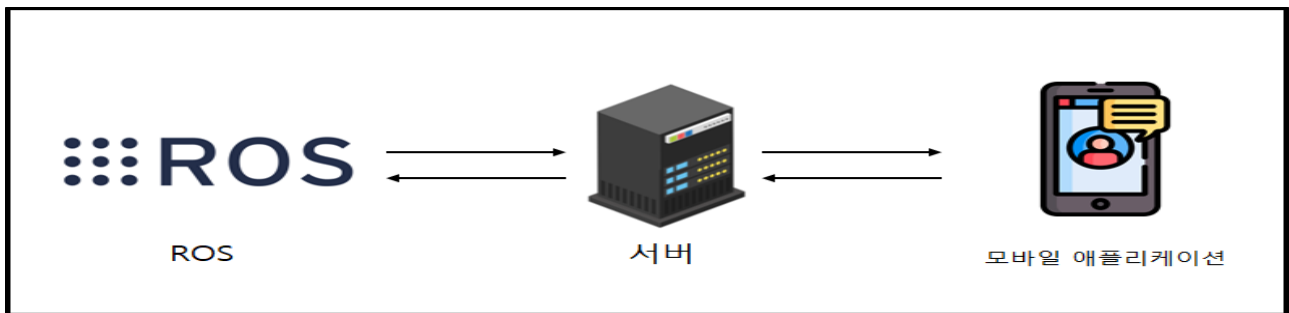
1. 개요
2. 시스템 장비 구성요구사항
3. 기능 요구사항
4. 성능 요구사항
5. 인터페이스 요구사항
6. 데이터 요구사항
7. 테스트 요구사항
8. 보안 요구사항
9. 품질 요구사항
10. 제약 사항
11. 프로젝트 관리 요구사항

1. 시스템 개요

- ROS 기반 자율주행 로봇 하드웨어 구조

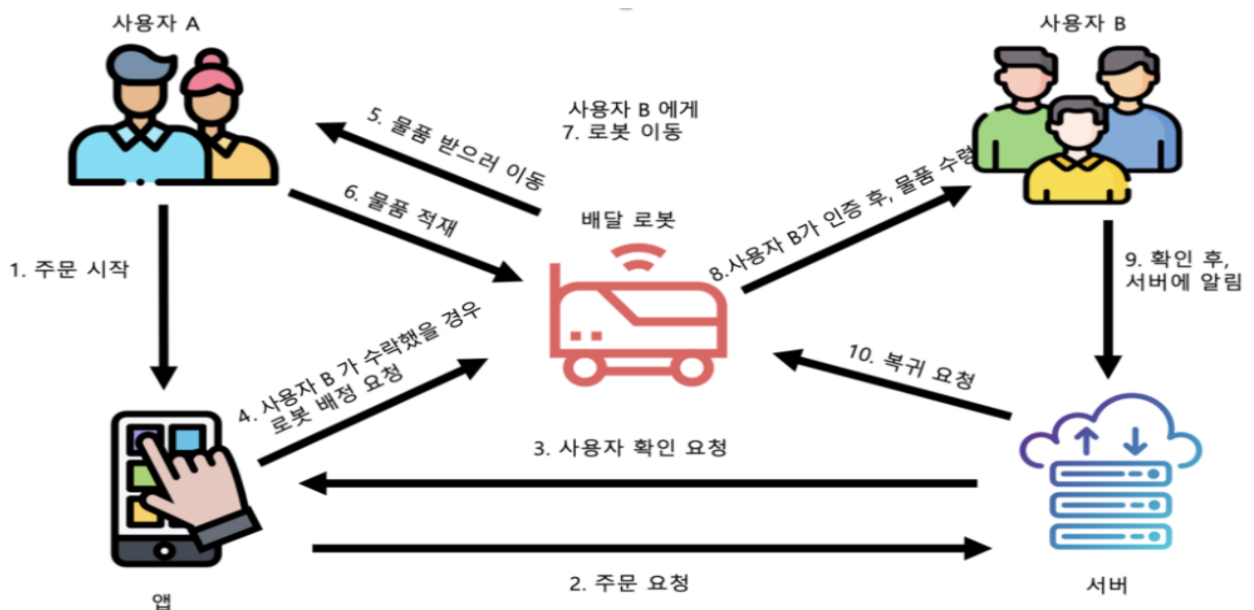


- 자율주행 배달 서비스 파이프라인



- 전체적인 시스템에 대한 시나리오

- 어플리케이션을 통해 주문을 하면 서버를 통해 로봇과 통신하여 로봇이 해당 목표지점까지 배달을 완료.



2. 시스템 장비 구성요구사항

요구사항 고유번호		ECR-001
요구사항 명칭		NVIDIA Jetson Orin Nano
요구사항 분류		시스템 장비 구성
상세 설명	정의	NVIDIA Jetson Orin Nano 개발자 키트에는 SD 카드 슬롯, 참조 캐리어 보드, 사전 조립된 방열판/팬, 19V DC 전원 공급 장치, M.2-Key E 기반 무선 네트워킹 모듈이 탑재된 특수 NVIDIA Orin Nano 8GB 모듈이 포함되어 있습니다. 부팅 가능한 마이크로SD 카드 슬롯 외에도 고속 저장을 위해 캐리어 밑면에 2개의 M.2 Key-M NVMe 소켓이 제공됩니다.
	세부 내용	<p>GPU - 32개의 Tensor 코어가 있는 NVIDIA Ampere 아키텍처</p> <p>CPU - 6코어 Arm Cortex-A78AE v8.2 64비트 CPU 1.5 MB L2 + 4 MB L3</p> <p>메모리 - 8 GB 128비트 LPDDR5 초당 68 GB</p> <p>저장장치 - 마이크로SD 슬롯을 통한 외부 M.2 Key M을 통한 외부 NVMe</p> <p>전력 - 7 W ~ 15 W</p> <p>카메라 - 2x MIPI CSI-2 22핀 카메라 커넥터</p> <p>M.2 Key M - 3세대 x4 PCIe</p> <p>M.2 Key M - 3세대 x2 PCIe</p> <p>M.2 Key E - PCIe (x1), USB 2.0, UART, I2S, I2C</p> <p>USB - A 타입: 4x USB 3.2 2세대 C타입: 디버그 및 디바이스 모드용1x</p> <p>네트워킹 - 1x GbE 커넥터</p> <p>디스플레이 - 디스플레이포트 1.2 (+MST)</p> <p>마이크로SD 슬롯 - 최대 SDR104 모드 UHS-1 카드</p> <p>기타 - 40핀 확장 헤더(UART, SPI, I2S, I2C, GPIO)12핀 버튼 헤더 4핀 팬 헤더 DC 전원 잭</p> <p>크기 - 100 mm x 79 mm x 21 mm(피트, 캐리어 보드, 모듈 및 열 솔루션 포함 높이)</p>
관련 요구사항		.
요구사항 출처		https://developer.nvidia.com/ko-kr/blog/develop-ai-powered-robots-smart-vision-systems-and-more-with-nvidia-jetson-orin-nano-developer-kit/

요구사항 고유번호		ECR-002
요구사항 명칭		RGB-D Camera
요구사항 분류		시스템 장비 구성
요구사항 상세 설명	정의	컬러 이미지와 깊이 정보를 동시에 캡처할 수 있는 카메라입니다. "RGB"는 빨간색, 초록색 및 파란색 채널의 색상 정보를 나타내며, "D"는 각 픽셀에서 카메라까지의 거리를 나타냅니다.
	세부 내용	<p>소프트웨어 지원: RGB-D 카메라와 함께 사용할 수 있는 소프트웨어 및 SDK(Software Development Kit)가 필요합니다.</p> <p>인터페이스 및 호환성: RGB-D 카메라는 일반적으로 USB 또는 기타 인터페이스를 통해 컴퓨터 또는 장치에 연결됩니다.</p> <p>시야: 수평 54 도, 수직 45 도</p> <p>깊이 이미지 크기: VGA(640x480)</p> <p>공간 X/Y 해상도 @ 0.5m: 1mm</p> <p>깊이 Z 해상도 @ 0.5m: <1mm</p> <p>최대 이미지 처리량: (QVGA) 60fps, (VGA) 30fps</p> <p>작동 범위: 0.35-3m</p> <p>컬러 이미지 크기: 1280x960</p> <p>오디오: 내장 마이크-마이크 2 개, 디지털 입력-4 시간</p> <p>데이터 인터페이스: USB 2.0, USB 3.0</p> <p>전력 소비: 2.25W</p> <p>전원 공급 장치: USB 2.0, USB 3.0</p> <p>치수: 18x2.5x3.5cm</p> <p>무게: 0.5lb</p> <p>운영 환경: 실내</p>
관련 요구사항		.
요구사항 출처		https://item.gmarket.co.kr/Item?goodscode=3641652410&GoodsSale=Y&jaehuid=200001169&NaPm=ct%3Dlvpzvltc%7Cci%3Dd9db57a79c4f2e2c04e12e6068aa3cd395cd9ff7%7Ctr%3DsIsl%7Csn%3D24%7Chk%3D6f65e6c6604d0f6fe2a4fd2e00b2451546b805c9

요구사항 고유번호		ECR-003
요구사항 명칭		LIDAR Sensor
요구사항 분류		시스템 장비 구성
요구사항 상세 설명	정의	레이저를 사용하여 대상으로 광을 발사하고, 반사된 광을 측정하여 정확한 거리 및 공간 정보를 제공하는 광학 원격 감지 기술 센서.
	세부 내용	<p>소프트웨어 및 통합 지원: 센서와 함께 제공되는 소프트웨어 및 API(Application Programming Interface)가 있어야 합니다. 이는 센서 데이터를 효과적으로 처리하고 다른 시스템에 통합하는 데 필요합니다.</p> <p>360도 전방향 스캐닝 거리 측정</p> <p>작은 거리 오차, 안정적인 성능 및 높은 정확도</p> <p>거리 측정은 30m 이상</p> <p>강력한 주변광 간섭에 대한 저항성</p> <p>안정적인 성능을 위한 산업용 등급 무브러시 모터 구동</p> <p>레이저 출력이 1등급 레이저 안전 기준을 충족</p> <p>5-12Hz의 적응형 스캐닝 주파수 (사용자 정의 지원)</p> <p>광자성 융합 기술을 이용한 무선 통신, 무선 전원 공급</p> <p>측정 주파수 최대 20kHz (사용자 정의 지원)</p>
관련 요구사항		.
요구사항 출처		https://www.devicemart.co.kr/goods/view?no=12533064&market=nave&NaPm=ct%3Dlvq09sm8%7Cci%3De5aed2bff1cf27dda7b6aaadfcece354173694ae%7Ctr%3Dslct%7Csn%3D876973%7Chk%3Ded998dff474801c47654c9bd5222ae9433afdb

3. 기능 요구사항

요구사항 고유번호		SFR-001
요구사항 명칭		사용자 배달 서비스 요청
요구사항 분류		기능
요구사항 상세 설명	정의	모바일 애플리케이션을 통해 배달 서비스를 이용하기 위해 사용자는 양식을 작성 및 요청
	세부 내용	- 작성된 정보를 서버로 전송
산출정보		.
관련 요구사항		.
요구사항 출처		.

요구사항 고유번호		SFR-002
요구사항 명칭		SLAM & NAVIGATION
요구사항 분류		기능
요구사항 상세 설명	정의	로봇이 외부 센서(LIDAR, 카메라)를 사용하여 주변 환경을 감지하고 이를 기반으로 자신의 위치를 추정하며, 동시에 환경의 구조를 매핑합니다. SLAM에서 생성된 지도를 사용하여 로봇이 목표 지점까지 경로를 계획하고 이동합니다.
	세부 내용	- SLAM 통한 실내 맵 확보 - 서버를 통해 전송받은 정보로 로봇에게 NAVIGATION 좌표 지정 - 로봇이 배달 작업을 마친 후 서버에게 결과 반환 및 사용자에게 알림
산출정보		.
관련 요구사항		.
요구사항 출처		.

요구사항 고유번호		SFR-003
요구사항 명칭		배달 결과 확인
요구사항 분류		기능
요구사항 상세 설명	정의	실시간으로 YOLO를 사용하여 호실 간판 객체를 탐지합니다. 객체의 영역에 있는 간판 이미지 정보를 텍스트 정보로 변환합니다.
	세부 내용	- 로봇이 배달을 통해 최종적으로 도착한 호실이 맞는지 호실 객체 탐지 수행 - 탐지된 영역에 대해 OCR을 통해 텍스트 검출 - 서버를 통해 전송받은 정보와 검출한 텍스트 정보 대조 및 판별 후 서버에 배달 결과 반환
산출정보		.
관련 요구사항		.
요구사항 출처		.

4. 성능 요구사항

요구사항 고유번호		PER-001
요구사항 명칭		통신 지연 시간 최소화
요구사항 분류		성능
요구사항 상세 설명	정의	데이터를 송수신하는 데 걸리는 시간을 최대한 단축하여 시스템의 실시간성과 반응성을 향상시킨다.
	세부 내용	- 모바일 애플리케이션, 서버, 로봇이 배달 서비스 과정에서 단계마다 필요 정보를 주고받기 때문에 통신 지연 시간이 길어질수록 응답속도도 늦어져 좋지 못한 사용자 경험을 만든다.
산출정보		.
관련 요구사항		.
요구사항 출처		.

5. 인터페이스 요구사항

요구사항 고유번호		SIR-001		
요구사항 명칭		입력창		
요구사항 분류		사용자 인터페이스	응락수준	필 수
요구사항 상세 설명	정의	호실 입력창 구현		
	세부 내용	<ul style="list-style-type: none"> - 사용자가 목표로 하는 해당 호실을 입력하는 인터페이스 구현 - 사용자의 현재 위치와 해당 목표 위치를 구분하여 인터페이스를 표시하여 구현 		

요구사항 고유번호		SIR-001		
요구사항 명칭		알림창		
요구사항 분류		사용자 인터페이스	응락수준	필 수
요구사항 상세 설명	정의	도착 시, 알림창 구현		
	세부 내용	<ul style="list-style-type: none"> - 로봇이 사용자에게 도착했다는 알림창을 구현하여 보내는 사람과 받는 사람에게 알림이 가도록 구현 - 알림을 받고 확인을 하면 다시 되돌아가는 인터페이스 구현 		

6. 데이터 요구사항

요구사항 고유번호	DAR-001		
요구사항 명칭	데이터셋 구축		
요구사항 분류	데이터	응락수준	필 수
요구사항 상세 설명	<ul style="list-style-type: none"> - SLAM을 통한 MAP 데이터 구축하여 yaml파일로 저장 - YOLO 작업을 위한 복도와 여러 이미지셋 데이터셋 수집 후 저장 		

요구사항 고유번호	DAR-001		
요구사항 명칭	전처리된 데이터		
요구사항 분류	데이터	응락수준	필 수
요구사항 상세 설명	<ul style="list-style-type: none"> - 저장된 MAP 데이터를 조정하여 보다 세밀한 MAP 구축 - 이미지데이터셋에 여러 albumentations 적용하여 새로운 데이터셋 저장 		

7. 테스트 요구사항

요구사항 고유번호	TER-001		
요구사항 명칭	성능 테스트		
요구사항 분류	테스트	응락수준	필 수
요구사항 상세 설명	<ul style="list-style-type: none">- YOLOv8과 SLAM 중 자율 주행을 할 때, 장애물 회피 능력과 해당 목표지점까지 정확도 성능을 비교하여 평가- 기존의 파라미터값과 인공지능 기술을 앙상블 적용하여 성능을 비교하여 평가		

8. 보안 요구사항

요구사항 고유번호	SER-001
요구사항 명칭	인증 및 권한 보안
요구사항 분류	보안
요구사항 상세 설명	<ul style="list-style-type: none">- 앱에서 필요 없는 기능의 권한이나 개인정보를 요청하지 않도록 함- 학교 구글 계정으로 로그인을 진행하고, 사용자 인증을 수행하여 학교 내 사용자만 앱에 접근할 수 있도록 함

요구사항 고유번호	SER-002
요구사항 명칭	사용자 인터페이스 보안
요구사항 분류	보안
요구사항 상세 설명	<ul style="list-style-type: none">- 시스템 화면에는 권한이나 인증 절차 없이 개인정보가 노출되지 않아야함- 사용자의 인증 상태에 따라 시스템의 화면에 보여지는 정보가 동적으로 변경되어야함

요구사항 고유번호	SER-003
요구사항 명칭	데이터 보안
요구사항 분류	보안
요구사항 상세 설명	<ul style="list-style-type: none">- 데이터베이스 접속은 관리자만 접속 가능해야하며, AWS IAM 통해 데이터베이스 접속 권한을 제어함- 코드를 공유할 때, DB 설정 파일이 같이 올라가지 않도록 해야함

9. 품질 요구사항

요구사항 고유번호		QUR-001
요구사항 명칭		신뢰성
요구사항 분류		품질
요구사항 상세 설명	정의	품질관리
	세부 내용	<ul style="list-style-type: none"> - 로봇이 배달하는 과정에서 장애물 탐지, 호실 찾기 등 자율주행이 원활하게 되어야함 - 배달 물품의 보안을 잘 유지시킬 필요가 있음

요구사항 고유번호		QUR-002
요구사항 명칭		사용성
요구사항 분류		품질
요구사항 상세 설명	정의	품질관리
	세부 내용	<ul style="list-style-type: none"> - 사용자가 시스템을 쉽게 운용할 수 있도록 직관적인 UI/UX 디자인이 필요함 - 로봇으로부터 물품을 수령하거나 로봇에게 물품을 전달할 때, 조작이 어렵지 않아야함

요구사항 고유번호		QUR-003
요구사항 명칭		유지관리성
요구사항 분류		품질
요구사항 상세 설명	정의	품질관리
	세부 내용	<ul style="list-style-type: none"> - 시스템에 문제가 발생할 경우, 문제를 신속하게 해결하기 위한 유지보수 및 복구 방안이 있어야함

10. 제약 사항

요구사항 고유번호		COR-001
요구사항 명칭		시스템 개발 제약사항
요구사항 분류		제약사항
요구사항 상세 설명	세부 내용	<ul style="list-style-type: none"> - 로봇 프로그래밍을 위해서는 ROS 환경을 이용해야하고, 해당 환경에서 특정 프로그래밍 언어(python, C++)를 사용해야함 - 서버 개발을 위해 프레임워크 Spring boot를 사용할 수 있어야하고, 프로그래밍 언어 Java를 다룰 수 있어야함 - 앱 개발을 위해 프레임워크 Flutter를 사용할 수 있어야하고, 프로그래밍 언어 Dart를 다룰 수 있어야함 - 애자일 방법론을 준수해야함

요구사항 고유번호		COR-002
요구사항 명칭		설계 및 구현 제약사항
요구사항 분류		제약사항
요구사항 상세 설명	세부 내용	<ul style="list-style-type: none"> - Jetson nano로는 Yolo, OCR, Slam을 모두 수행하기에 연산량이 부족할 가능성이 있으므로 Jetson Orin Nano을 로봇의 구성품으로 사용할 필요가 있음

요구사항 고유번호		COR-003
요구사항 명칭		데이터 제약사항
요구사항 분류		제약사항
요구사항 상세 설명	세부 내용	<ul style="list-style-type: none"> - 사용자의 개인 정보나 기타 민감한 데이터를 안전하게 보호해야하므로 데이터 암호화, 접근 제어, 백업 및 회복 정책 등을 준수해야함

11. 프로젝트 관리 요구사항

요구사항 고유번호	PMR-001
요구사항 명칭	프로젝트 수행 조직
요구사항 분류	프로젝트 관리
요구사항 세부 내용	<ul style="list-style-type: none"> - 로봇 개발 (송찬호) <ul style="list-style-type: none"> • 로봇 하드웨어 및 소프트웨어 개발 • 로봇 자율주행 알고리즘 적용 - 백엔드 개발 (김수미) <ul style="list-style-type: none"> • 데이터베이스 설계 및 관리 • API 엔드포인트 개발 - 프론트엔드 개발 (이동현) <ul style="list-style-type: none"> • 사용자 인터페이스 개발 • 사용자 입력 기능 구현

요구사항 고유번호	PMR-001
요구사항 명칭	프로젝트 일정계획
요구사항 분류	프로젝트 관리
요구사항 세부 내용	<ul style="list-style-type: none"> - 프로젝트 계획 단계에서는 요구사항 분석 및 명세화, 시스템 설계 및 구조화, 로봇 ROS 개발 환경 설정, 서버 및 앱 개발 환경 설정이 필요하다. 이 단계에서는 요구사항 명세서, 아키텍처 설계서, 프로젝트 일정서 제출한다. - 프로젝트 개발 단계에서는 백엔드 서비스 및 API 개발, 프론트엔드 화면 개발, 자율주행 로봇 하드웨어 및 소프트웨어 개발, 로봇과 서버/서버와 앱 통신 개발이 필수적이다. 이에 따라 개발한 코드 산출물을 제출한다. - 프로젝트 테스트 단계에서는 실제로 로봇 배달 시스템 운영해보며 테스트하고, 문제점을 해결해야한다.

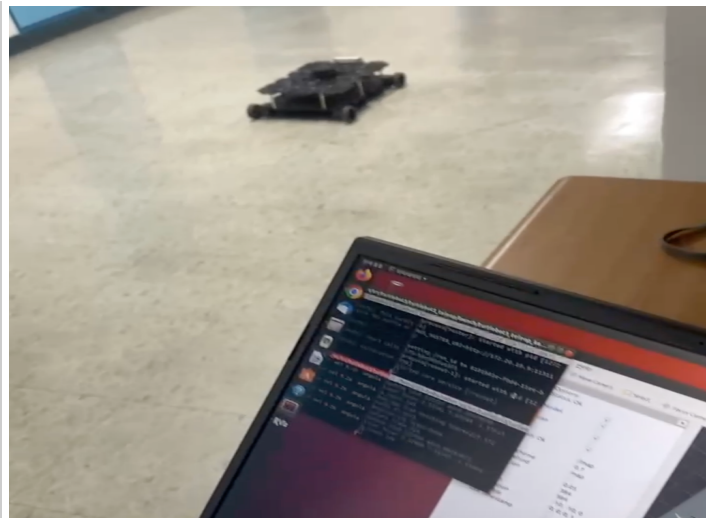
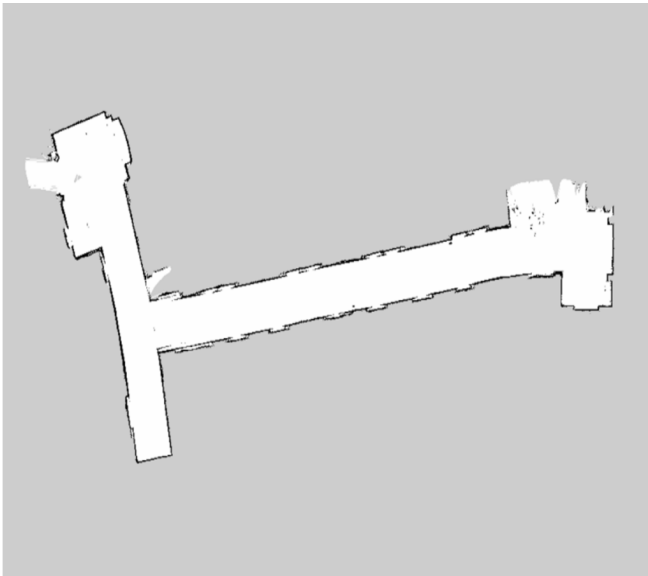
[별첨2]

중간보고서

1. 요구사항 정의서에 명시된 기능에 대하여 현재까지 분석, 설계, 구현(소스코드 작성) 및 테스트한 내용을 기술하시오.

<구현>

- 실제 환경에서 로봇을 사용하여 지도 생성 및 로봇 이동 작업 수행
- SLAM을 이용한 MAP 구축



AMCL을 사용하여 로봇의 위치 추정 및 내비게이션 구현

```
import rospy
from geometry_msgs.msg import PoseWithCovarianceStamped

def pose_callback(data):
    rospy.loginfo("Robot Current Pose: [%f, %f, %f]", data.pose.pose.position.x, data.pose.pose.position.y,
    data.pose.pose.orientation.w)

def listener():
    rospy.init_node('pose_listener', anonymous=True)
    rospy.Subscriber("/amcl_pose", PoseWithCovarianceStamped, pose_callback)
    rospy.spin()

if __name__ == '__main__':
    listener()
```

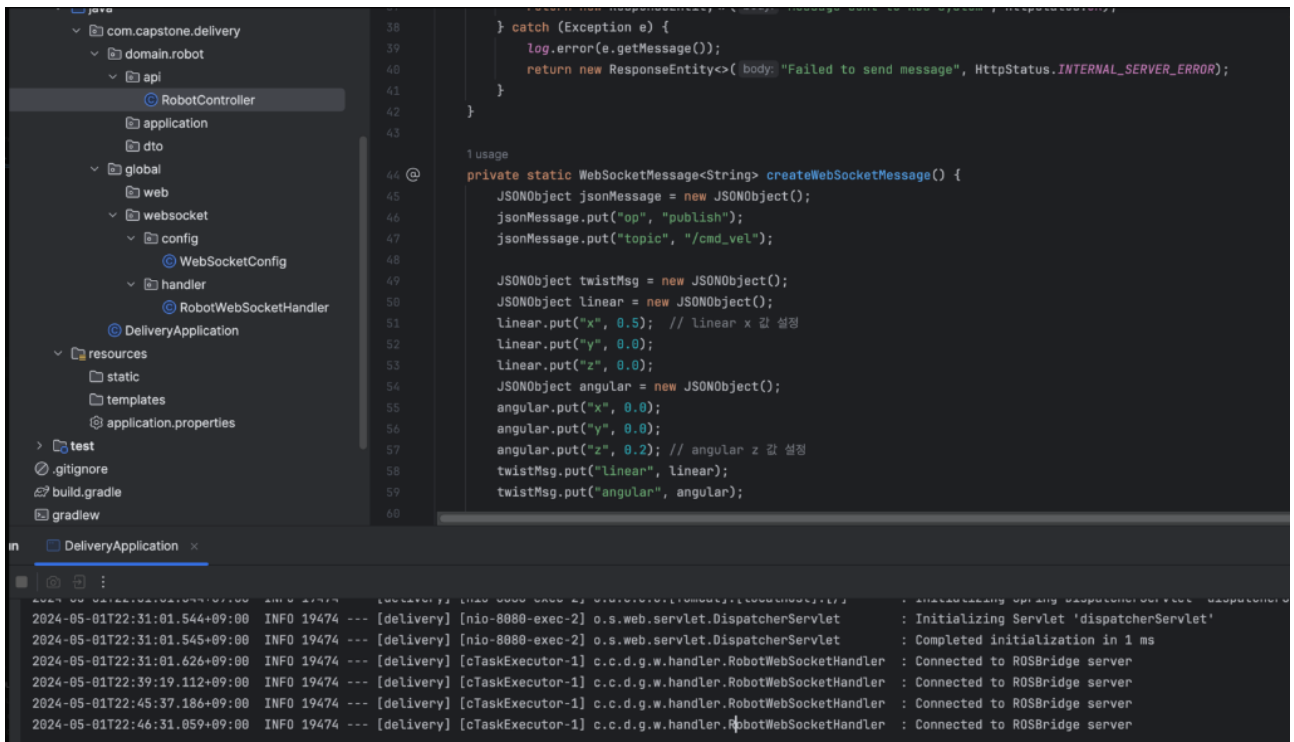
- 로봇의 해당 위치값 (x,y 좌표값)을 얻어냄

```

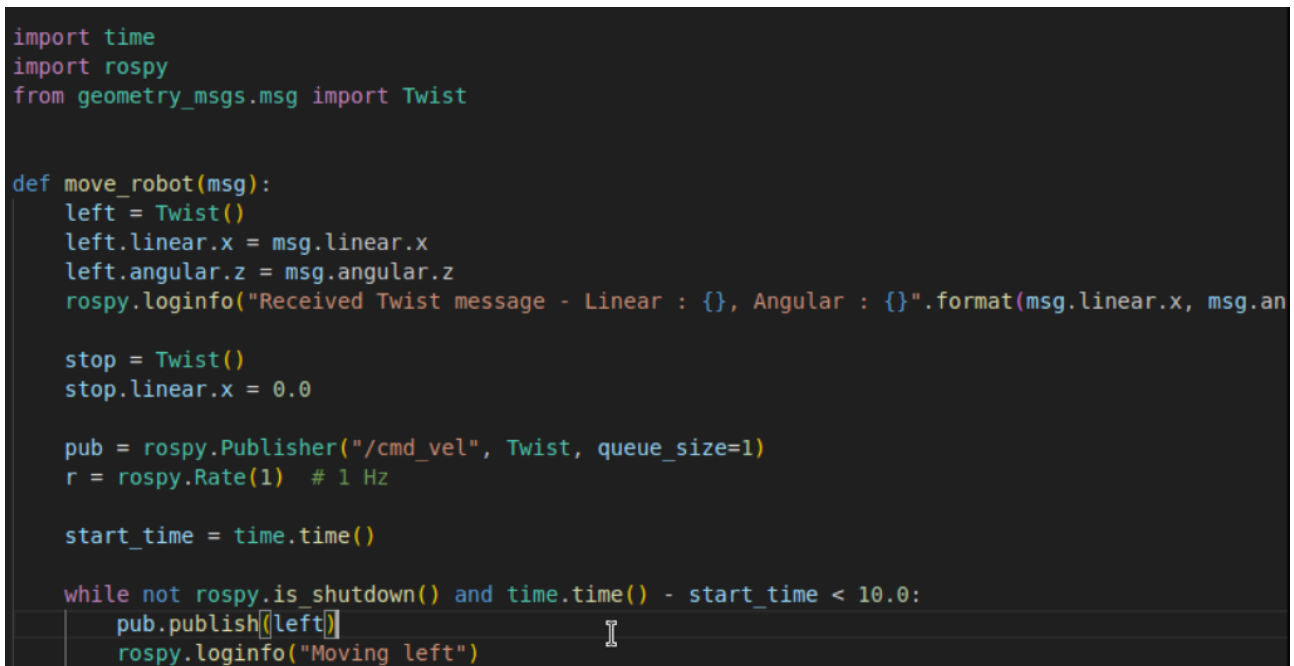
19     '302': {
20         'front': (0.7318, -2.0756, 0, 0, 0, 1),
21         'side': (0.7318, -2.0756, 0, 1, 0, 0),
22         'back': (0.7318, -2.0756, 0, 0, 1, 0)
23     }
24 }
25 def move_to_room(room_id, orientation_type='front', frame="map"):
26     if room_id in room_orientations and orientation_type in room_orientations[room_id]:
27         x, y, qx, qy, qz, qw = room_orientations[room_id][orientation_type]
28
29         rospy.init_node('move_to_room_node', anonymous=True)
30         goal_publisher = rospy.Publisher('/move_base_simple/goal', PoseStamped, queue_size=1)
31
32         goal = PoseStamped()
33         goal.header.seq = 1
34         goal.header.stamp = rospy.Time.now()
35         goal.header.frame_id = frame
36         goal.pose.position.x = x
37         goal.pose.position.y = y
38         goal.pose.orientation.x = qx
39         goal.pose.orientation.y = qy
40         goal.pose.orientation.z = qz
41         goal.pose.orientation.w = qw
42
43         rospy.sleep(1)
44         goal_publisher.publish(goal)
45         rospy.loginfo(f"Goal published for room {room_id} with orientation {orientation_type}: moving to position x={x}, y={y}")
46
47         global status_subscriber
48         status_subscriber = rospy.Subscriber('/move_base/status', GoalStatusArray, check_goal_status)
49     else:
50         rospy.loginfo("Room ID or orientation type not found.")
51
52 # 로봇을 특정 호실로 이동시키는 함수
53 # def move_to_room(room_id, frame="map"):
54 #     if room_id in room_coordinates:
55
56 def check_goal_status(status_msg):
57     if status_msg.status_list:
58         status = status_msg.status_list[-1].status
59         if status == 3:
60             rospy.loginfo("Goal reached. Shutting down.")
61             rospy.signal_shutdown("Goal reached")
62         elif status in [4, 5]:
63             rospy.loginfo("Goal not reached but stopped trying")
64             rospy.signal_shutdown("Stopping attempts to reach goal")
65
66 if __name__ == '__main__':
67     try:
68         room_id = input("Enter the room ID (e.g., '301','302'): ").strip()
69         move_to_room(room_id)
70         rospy.spin()
71     except rospy.ROSInterruptException:
72         rospy.loginfo("ROS interrupt exception")
73     except Exception as e:
74         print(f"An error occurred: {e}")

```

- 호실을 입력하면 로봇이 해당 호실로 찾아가는 기능 구현
- 사용자가 원하는 호실을 입력하면 해당 호실의 좌표값을 받아 이동하는 코드
- ROS Bridge를 통한 서버와 ROS 간의 통신 구현



- ROS에서 메시지를 받아 가상환경에서 로봇을 움직이는 코드 작성



<분석>

- 데이터 탐색적 분석
- SLAM 정확도 및 속도 분석
- YOLO 정확도 및 속도 분석
- 외부 지식 통계
- ROS node 와 서버간의 통신 분석

<설계>

- 자율주행로봇 모델 설계
- SLAM을 통해 globalmap과 여러 costmap 장애물에 대한 정확도 설계
- ROS node와 서버간의 통신 설계
- 사용자들이 보기 쉽고 편리한 인터페이스의 어플리케이션 설계

2. 프로젝트 수행을 위해 적용된 추진전략, 수행 방법의 결과를 작성하고, 만일 적용과정에서 문제점이 도출되었다면 그 문제를 분석하고 해결방안을 기술하시오.

- 프로젝트 시작 전에 팀원들과 스터디를 진행하여 로봇과 ROS 환경에 대한 기본적인 원리와 동작 과정을 이해
- 팀원 간에 정기적인 회의를 통해 프로젝트의 진행 상황을 공유하고, 문제점을 식별하고 해결할 수 있는 방안을 논의
- 팀원 간의 의사소통을 강화하기 위해 적극적으로 정보를 공유하고, 문제가 발생할 때 즉시 해결할 수 있는 방안을 모색
- 업무 분담 및 역할을 명확히 정의하여 혼란을 최소화

캡스톤 디자인 I 중간보고서 채점표

평가도구	평 가 항 목	평 가 점 수				
		1	2	3	4	5
중간 보고서 및 실행 결과	1. 요구사항 정의서(기능, 성능, 인터페이스 등)가 구체적으로 작성되었는가?					
	2. 요구분석, 설계 산출물(모델, 프로토타입 등)의 내용이 충실한가?					
	3. 설계 및 구현 문제를 위해 적용한 이론, 문제해결 방법이 제시되었으며 그 적용이 적합한가?					
	4. 구현된 소프트웨어(또는 이와 동등한 하드웨어 시스템)가 버그 없이 실행되었는가?					
	5. 구현된 소프트웨어(또는 이와 동등한 하드웨어 시스템)의 성능 요구사항은 충족되었는가?					
도구활용	6. 설계 및 구현을 위해 도구가 적절히 활용되었는가?					
	7. 도구의 활용수준(능숙도)은 프로젝트 수행에 적합한가?					
팀원의 업무 및 역할	8. 팀원의 업무분담에 따른 역할 및 협력이 충실히 이루어졌는가? (평가자에 의한 질의)					
	9. 프로젝트 중간 진척상황에 대해 팀원이 충분히 인지하고 있는가?(평가자에 의한 질의)					
합계						
*검토 의견(최종완료 때까지 보완해야할 점에 대해 작성 요망)						
심사위원(소속):		(이름)			(인)	