

전정대 배달 로봇



이 세계 _ 개발 _ 라이프

HYPER

멘토: 김대호 신승희 이원준

멘티: 강지원 신희준 황준태

CONTENT

01. DEFINITION

02. GOALS

03. USED PARTS

04. PROCESS

05. DIFFICULTY

06. REUSLT

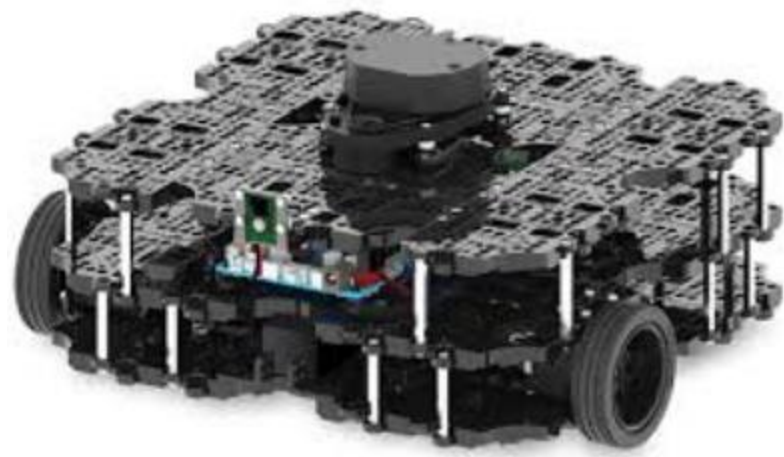
PROJECT DEFINITION



PROJECT GOALS

로봇을 사용하여
주행 중 목표하는 번호판을 인식하면 문 앞에서 정지

USED PARTS



**TurtleBot 3
Waffle**

Lidar 센서, 모터, Raspberry pi 등
다양한 모듈이 포함되어 있는 하드웨어

↔
ROS



Local PC

Ubuntu 18.04가 설치 되어있는 PC

↔
Socket



Raspberry pi

카메라와 연결되어 있으며
RaspberryPi 자체 OS를 사용

PROJECT PROCESS



프로젝트 준비

하드웨어에 사용할 OS 버전 선택과
번호판 인식 알고리즘을 ANPR로
결정했습니다.



환경 구성

로컬 pc와 터틀봇의 Raspberry pi에
Ubuntu 18.04와 ROS를 설치했습니다.



통신 및 제어

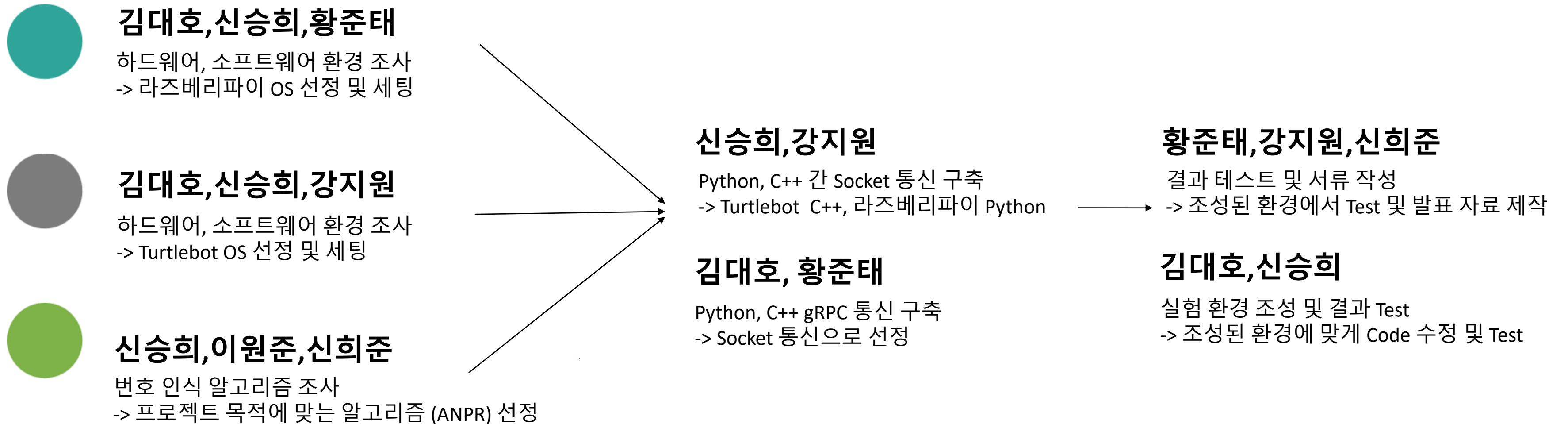
터틀봇과 로컬 pc를 ROS를 사용하여
연결한 뒤 SLAM을 사용하여 지도를
작성했습니다.



번호판 인식

소켓 통신을 통해 Raspberry pi 카메라
에서 처리한 데이터를 PC로 전송하고
데이터가 옳다면 터틀봇으로 전송합니다.

PROJECT PROCESS



PROJECT DEFFICULTY



라즈베리파이 과부화

Turtlebot 에서 주행과 번호 인식을 동시에 진행하기 어렵다.
-> 별도의 라즈베리파이에 카메라를 부착하여 부하를 덜어준다.



라즈베리파이와 Turtlebot 간 사용 언어 다름

Turtlebot 은 C++, 라즈베리파이는 Python 언어를 사용함
-> TCP socket 통신하여 양 언어간 통신이 가능하게 함.



Turtlebot navigation 실행파일만 존재한다

코드를 수정하여 번호 인식 후 주행을 변경할 수 없다.
-> Rviz navigation 프로세스를 kill 하는 방법으로 주행을 변경한다.



Turtlebot 속도가 빠르다

Turtlebot 속도로 번호 인식을 잘 하지 못함.
-> Turtlebot 속도를 낮춰준다.

PROJECT RESULT



PROJECT RESULT

01 제한된 통신 반경

02 딥러닝을 이용한 번호판 인식

03 하드웨어 제약

04 완전한 실내 자율 주행

THANK YOU.

이 세 계 _ 개 발 _ 라 이 프