



포팅 메뉴얼

FEST 프로젝트는 **Unity 6, ROS2 foxy**를 기반으로 제작되었습니다.

1. 사용 도구

이슈 관리 : Jira
형상 관리 : GitLab
커뮤니케이션 : Notion, MatterMost
디자인 : Figma

2. 개발 환경

2.1 사용 언어

python : 3.8.10(ROS2 foxy), 3.11.5(Fastapi)
C++ : c++14

2.2 시뮬레이션

ROS2 : foxy
Windows : 11
ubuntu : 20.04 LTS
unity : 6000.0.41f1

2.3 인프라

Docker
Docker-Compose
Nginx

2.4 DB

postgreSQL : 17.4
Redis : 3.0.504

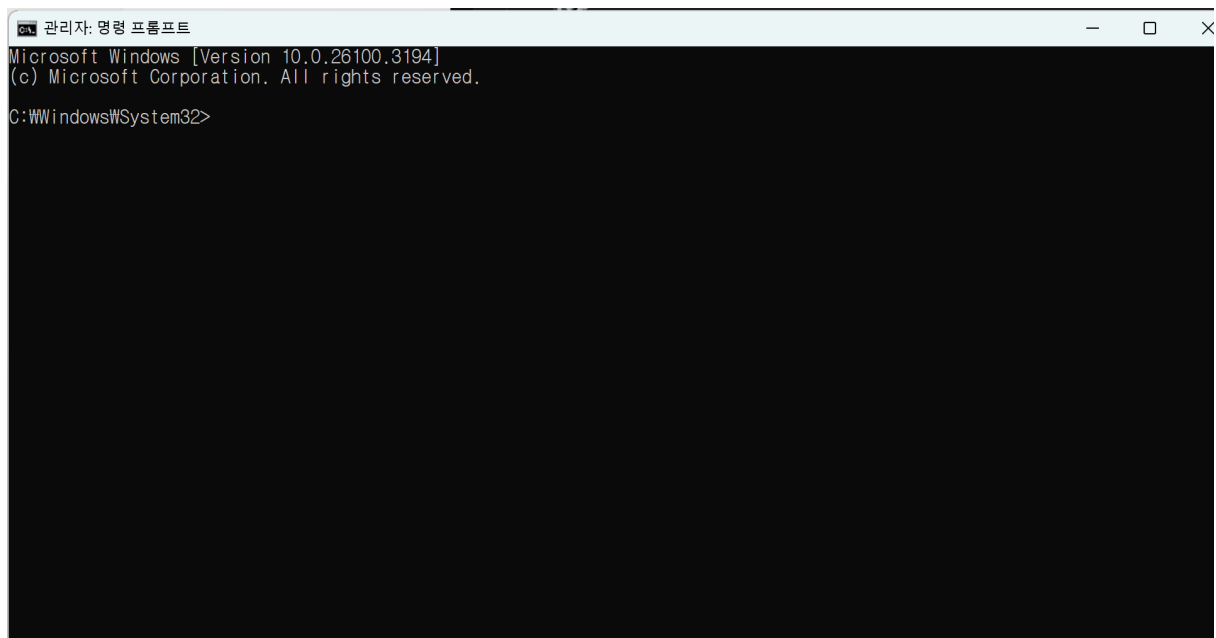
2.5 FE/BE

Vue.js
FastAPI
MQTT(mosquitto)

3. 개발환경 구축

3.1 ROS2

명령프롬프트 실행 후



```
관리자: 명령 프롬프트
Microsoft Windows [Version 10.0.26100.3194]
(c) Microsoft Corporation. All rights reserved.
C:\Windows\System32>
```

python upgrade

choco upgrade -y python --version=3.8.10

visual C++ redistributable upgrade

```
choco install -y vc_redist2013 vc_redist140
```

openssl

```
choco install openssl --version=1.1.1.1400
```

ROS 2 Foxy

ROS 2 Foxy Fitzroy - Patch Release 11

These are the binary packages for ROS 2 Foxy Fitzroy Patch Release 11 (2023-06-20)

For runtime dependencies see the binary package [installation instructions](#).

Your system must be up-to-date to be compatible with the downloaded packages.

Additionally there are [Debian packages](#) for Ubuntu Focal.

Note: ignore the Source code links, because they don't actually contain the source code for ROS 2 (they're auto-generated by GitHub). Instead, grab the binary package for your platform. If you're interested in building from source, consult the [building from source instructions](#).

Assets

7

12 1 12 people reacted

Assets

7

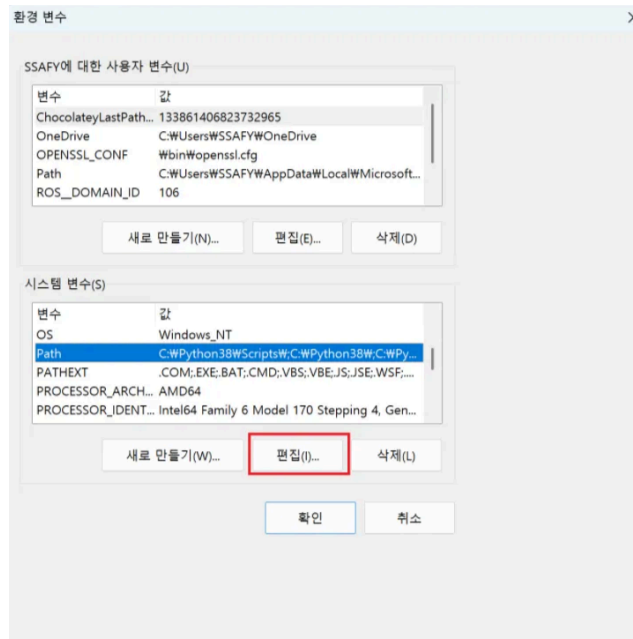
| | | |
|---|---------|--------------|
| ros2-foxy-20230620-linux-centos-amd64.tar.bz2 | 640 MB | Jun 21, 2023 |
| ros2-foxy-20230620-linux-focal-amd64.tar.bz2 | 842 MB | Jun 21, 2023 |
| ros2-foxy-20230620-linux-focal-arm64.tar.bz2 | 666 MB | Jun 21, 2023 |
| ros2-foxy-20230620-windows-debug-amd64.zip | 1000 MB | Jun 21, 2023 |
| ros2-foxy-20230620-windows-release-amd64.zip | 660 MB | Jun 21, 2023 |
| Source code (zip) | | Jun 21, 2023 |
| Source code (tar.gz) | | Jun 21, 2023 |

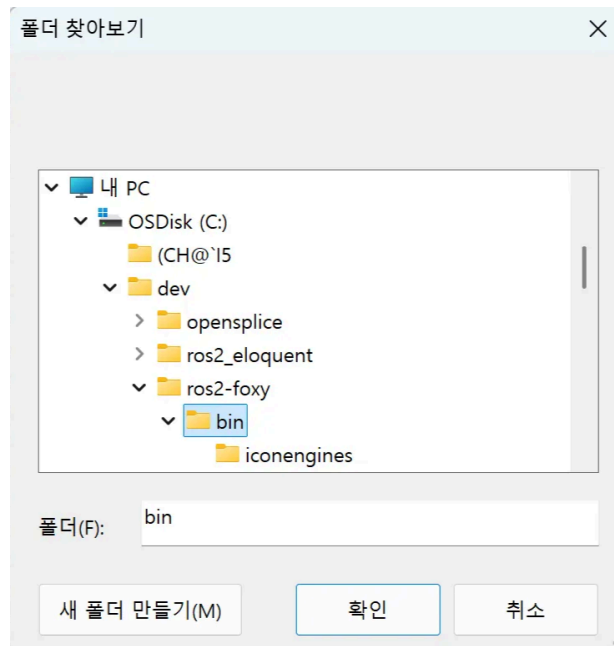
12 1 12 people reacted

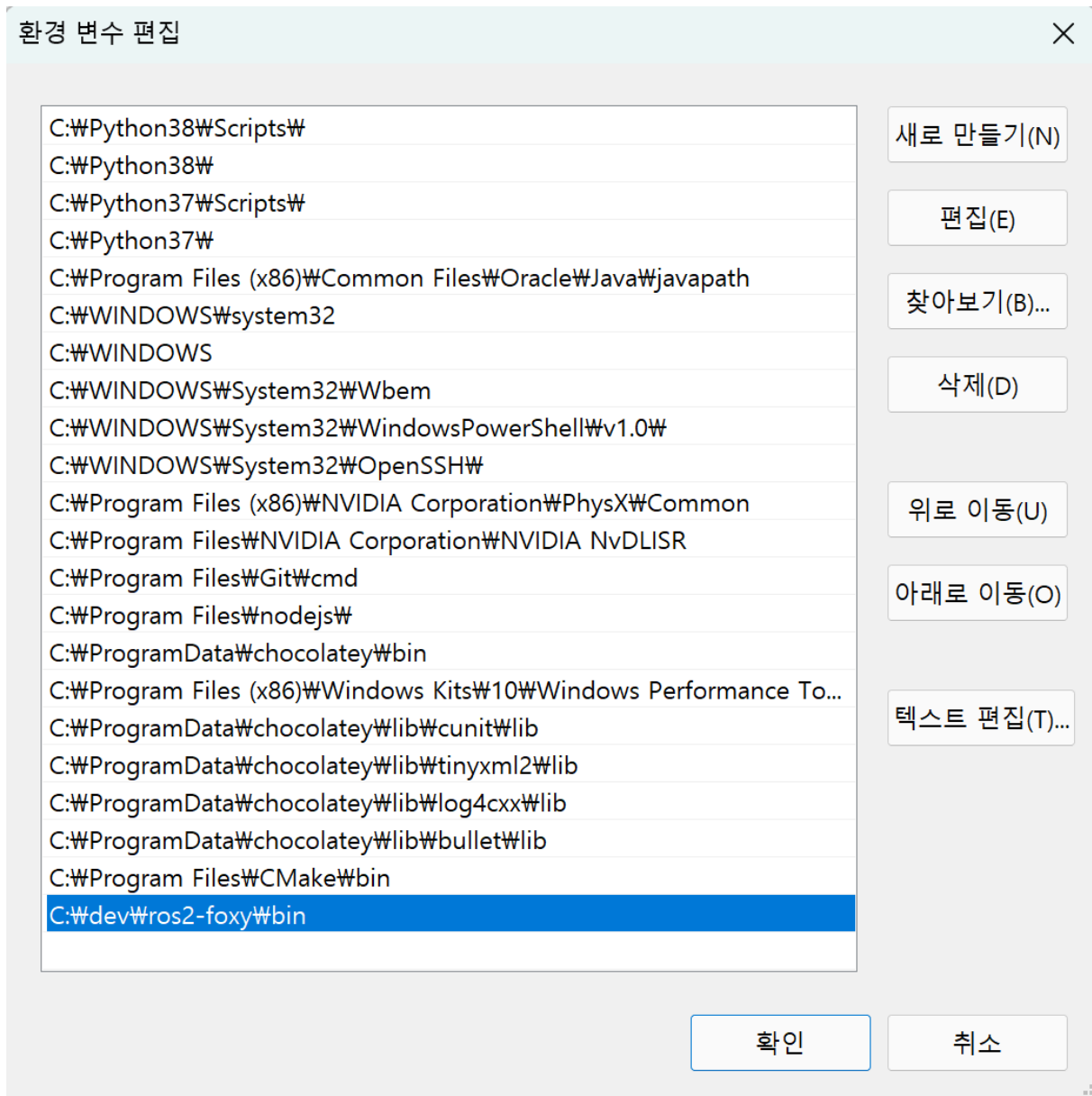
Zip 파일 압축 해제 후 폴더명 ros2_foxy 로 변경

```
C:
├─ dev
│   └─ opensplice
│       └─ ros2-foxy
```

C:\dev\ros2-foxy\bin 환경변수 추가







3.2 Redis

1. port : 6379
2. redis 시스템 환경 변수 path에 등록

C:\Program Files\Redis\

3. redis 클라이언트 실행(cmd)

```
redis-cli
```

3.3 PostgreSQL(17.4)

1. postgis 다운로드

- application stack builder 실행
- postgis 클릭 후 설치

2. 시스템 환경 변수 등록

```
C:\Program Files\PostgreSQL\17\bin
```

3. 터미널에서 postgresql 실행

```
net start postgresql-x64-17
```

```
psql -U postgres # 초기 사용자 로그인, db 설치시 사용한 비밀번호 입력
```

4. 사용자 생성 및 권한 부여

```
CREATE USER 사용자 WITH PASSWORD '사용자 '; # 사용자 생성  
ALTER USER 사용자 CREATEDB; # 사용자에게 db 생성 권한 부여
```

5. 사용자 db 생성 및 사용자 database 에 권한 부여

처음에 로그인 하려는 사용자와 같은 이름의 db가 없으면 해당 사용자로 로그인이 안됨

```
create database 사용자;  
grant all privileges on database 사용자 to 사용자;
```

6. 루트 계정 로그아웃 사용자 로그인

```
\q  
psql -U 사용자
```

7. fest_db 생성

```
create database fest_db;
```

8. fest_db에 postGIS 확장 활성화(루트 계정만 가능) 및 ssafy 사용자에게 사용권한 부여

```
\q # 로그아웃  
psql -U postgres -d fest_db # 루트계정으로 fest_db 연결  
  
-- PostGIS 확장 활성화(postgis는 db 마다 확장을 활성화함)  
CREATE EXTENSION postgis;  
  
GRANT USAGE ON SCHEMA public TO 사용자;  
GRANT ALL ON ALL TABLES IN SCHEMA public TO 사용자;  
GRANT ALL ON ALL SEQUENCES IN SCHEMA public TO 사용자;
```

9. 사용자 계정으로 로그인

```
\q # 로그아웃  
  
psql -U 사용자 -d fest_db
```

10. 테이블 생성

-- 로봇 정보 테이블

```
CREATE TABLE robots (  
    robot_id VARCHAR(50) PRIMARY KEY,  
    name VARCHAR(100),  
    battery INTEGER  
);
```

-- 화재 사고 테이블

```
CREATE TABLE incidents (  
    incident_id VARCHAR(50) PRIMARY KEY,  
    location GEOMETRY(POINT, 4326),  
    detected_at TIMESTAMP,  
    extinguished_at TIMESTAMP,  
    status VARCHAR(20) DEFAULT 'active'  
);
```

-- 로봇 위치 이력 테이블

```
CREATE TABLE robot_positions (  
    id SERIAL PRIMARY KEY,  
    robot_id VARCHAR(50) REFERENCES robots(robot_id),  
    position GEOMETRY(POINT, 4326),  
    orientation FLOAT,  
    recorded_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
CREATE TABLE firefighting_missions (  
    mission_id VARCHAR(50) PRIMARY KEY,  
    robot_id VARCHAR(50) REFERENCES robots(robot_id),  
    incident_id VARCHAR(50) REFERENCES incidents(incident_id),  
    status VARCHAR(20) DEFAULT 'assigned', -- assigned, in_progress, completed  
    assigned_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
    arrived_at TIMESTAMP NULL,  
    completed_at TIMESTAMP NULL  
);
```

3.4 MQTT

1. 다운로드

<https://mosquitto.org/download/>
mosquitto-2.0.21a-install-windows-x64.exe

port : 1883

2. mosquitto 실행파일 시스템 환경변수 path에 등록

C:\Program Files\mosquitto

3. mosquitto 실행

cmd 에서 mosquitto.exe -v 입력

mosquitto.exe -v

3.5 Vue

1. node.js LTS 최신 버전 설치
2. npm install
3. npm run dev

3.6 Cuda

1. nvidia드라이버 다운로드(본인컴퓨터 gpu랑 같은걸로!!!!)

<https://www.nvidia.com/ko-kr/drivers/>

- gpu확인방법

시작 메뉴 > 시스템 정보 > 구성 요소 > 디스플레이
설치확인

PowerShell이나 명령 프롬프트에서
nvidia-smi

2. cuda 다운

<https://developer.nvidia.com/cuda-12-1-0-download-archive>

Select Target Platform

Click on the green buttons that describe your target platform. Only supported platforms will be shown. By downloading and using the software, you agree to fully comply with the terms and conditions of the [CUDA EULA](#).

| | |
|------------------|---|
| Operating System | Linux Windows |
| Architecture | x86_64 |
| Version | 10 11 Server 2016 Server 2019 Server 2022 |
| Installer Type | exe (local) exe (network) |

Download Installer for Windows 10 x86_64

The base installer is available for download below.

| | |
|------------------|--------------------------|
| > Base Installer | Download (3.0 GB) |
|------------------|--------------------------|

Installation Instructions:

3. pytorch설치

```
pip install torch torchvision torchaudio --index-url https://download.pytorch.org
```

4. ros2/ros2_ws/src/perception_py 경로

4. 실행 방법

4.1 배포 파일 FEST simulator.exe 파일 실행

4.2 ros2/ros2_ws/start_launches.bat 실행 후, exe 파일에서 연결 표시 확인

4.3 ros2/ros2_ws/start_goal.bat 실행(경로 탐색)

4.4 ros2/ros2_ws/run_rviz2.bat 실행(시각화 툴)

