2021-1 Capstone Design

11주차 캡스톤 진행 상황

Deeply

Capstone Design Team : 5

17011885 박세정

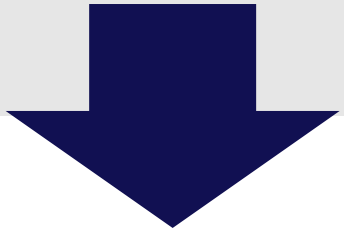
17011869 이혜인

17011757 박미희

10,11주차 진행 목표

1. WebRTC data channel을 열어 명령어 보내기 ✅

2. 받은 명령어를 통해 로봇 제어하기 🔼

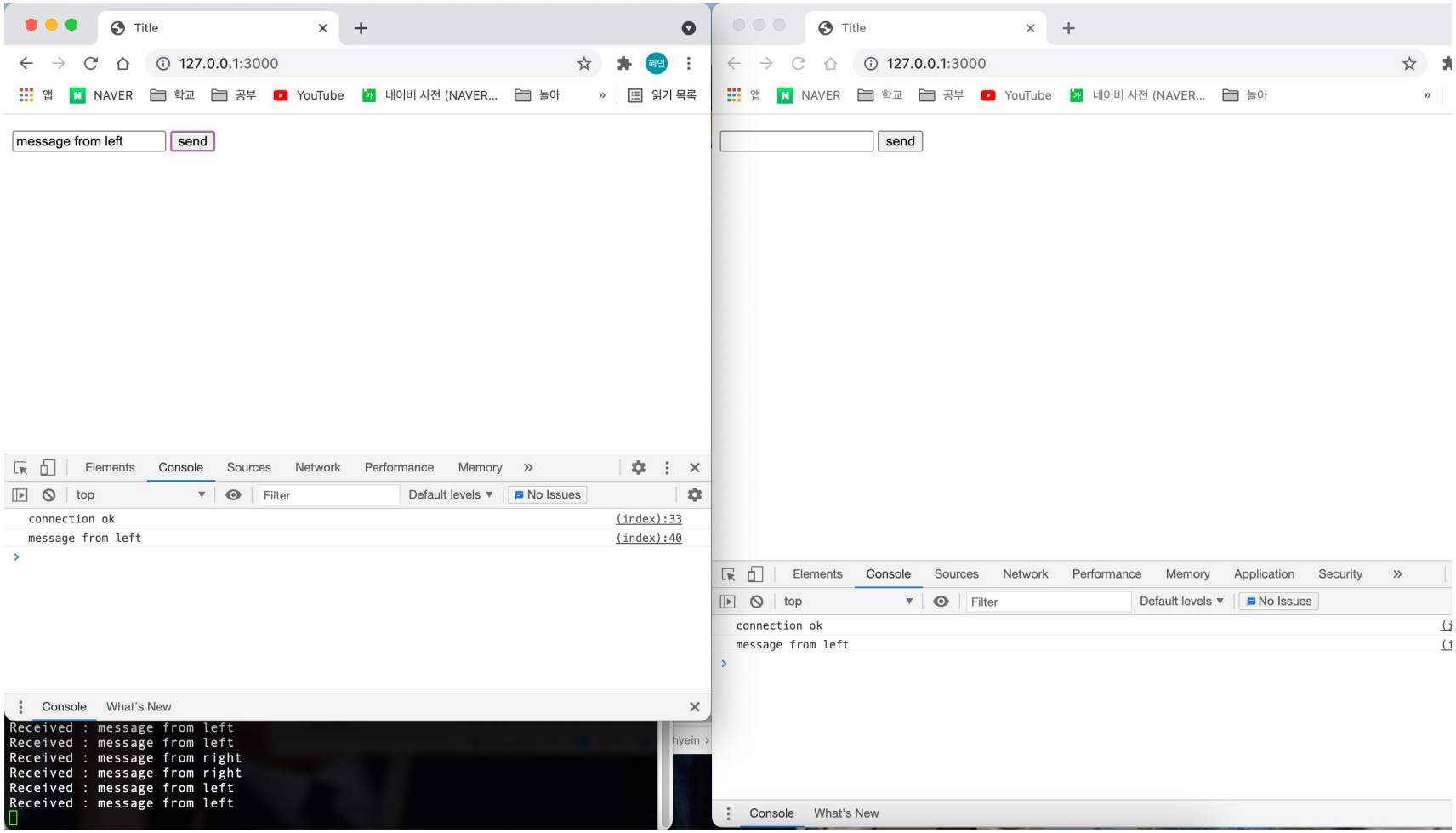


1. WebRTC를 통해 서로 다른 사용자가 메세지 주고 받기

2. 브라우저에서 명령어 받아오기

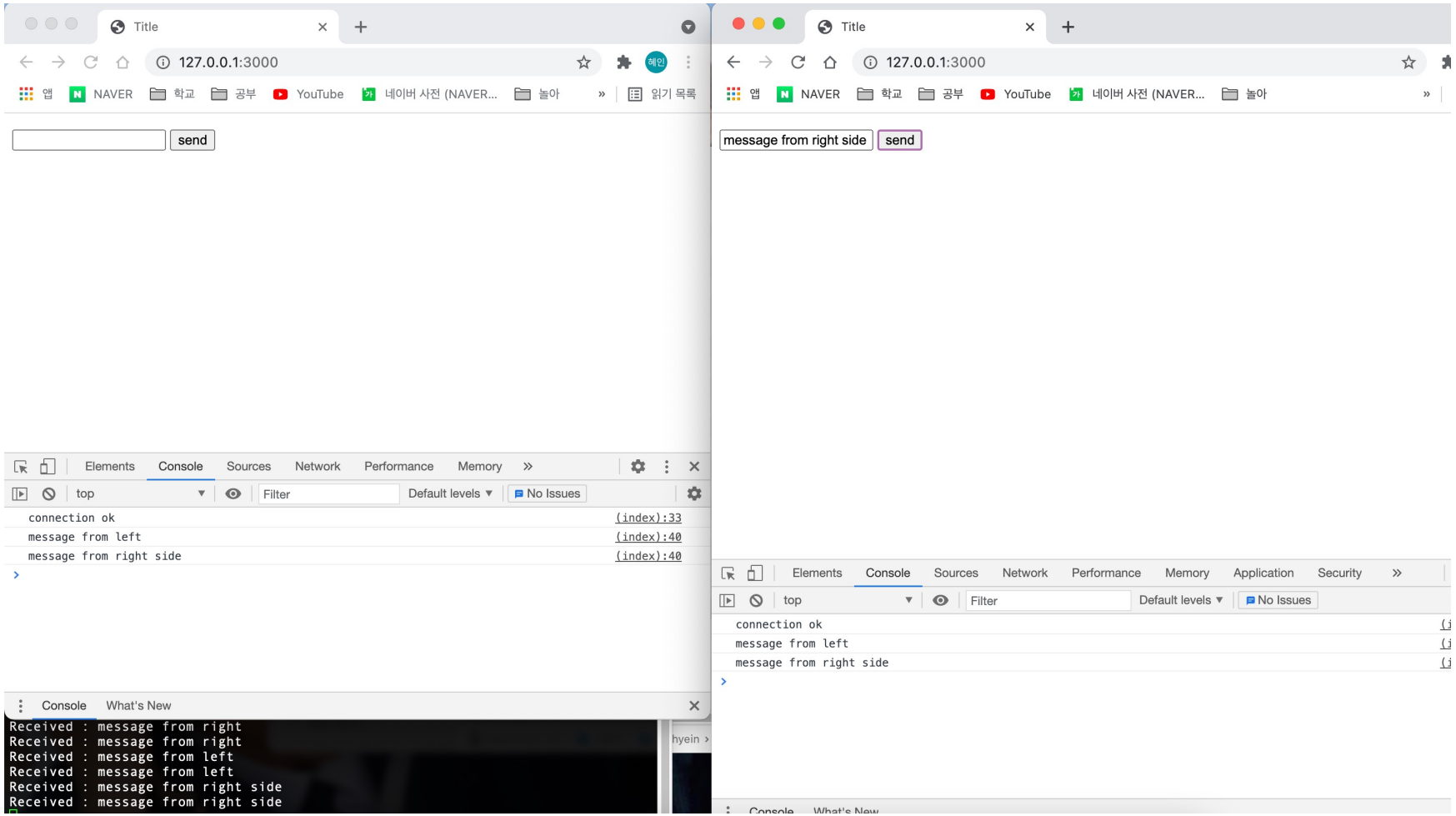
11주차 진행 과정

WebRTC를 통해 메세지 주고 받기



11주차 진행 과정

WebRTC를 통해 메세지 주고 받기

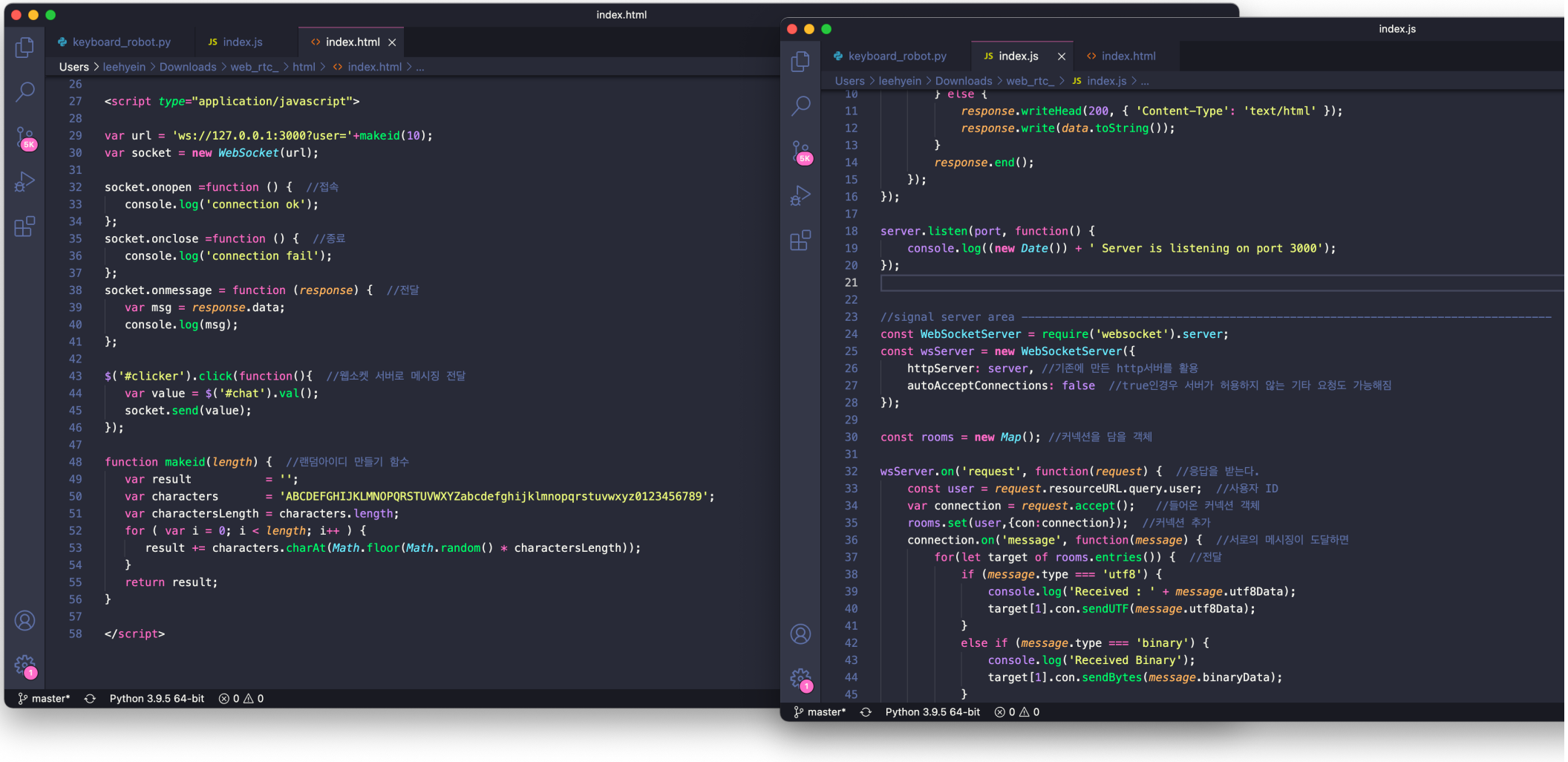


11주차 진행 과정

WebRTC를 통해 메세지 주고 받기

websocket 모듈 이용,

기존 구현한 http 서버를 활용해 웹 소켓 서버 구동



11주차 진행 과정

WebRTC를 통해 메세지 주고 받기

1. 같은 디바이스에서 두 개의 브라우저가 서로 메세지 주고 받음

2. 두 개의 디바이스가 서로 메세지 주고 받음

11주차 진행 과정

브라우저에서 명령어 받아오기

1. RTCBot API 이용: 로봇 원격 제어

https://github.com/dkumor/rtcbot

2. RPi – Browser WebRTC 연결

python, javascript

RTCBot

pypi v0.2.4 npm v0.2.4 docs passing chat on gitter tests passing

RTCBot's purpose is to provide a set of simple modules that help in developing remote-controlled robots in Python, with a focus on the Raspberry Pi.

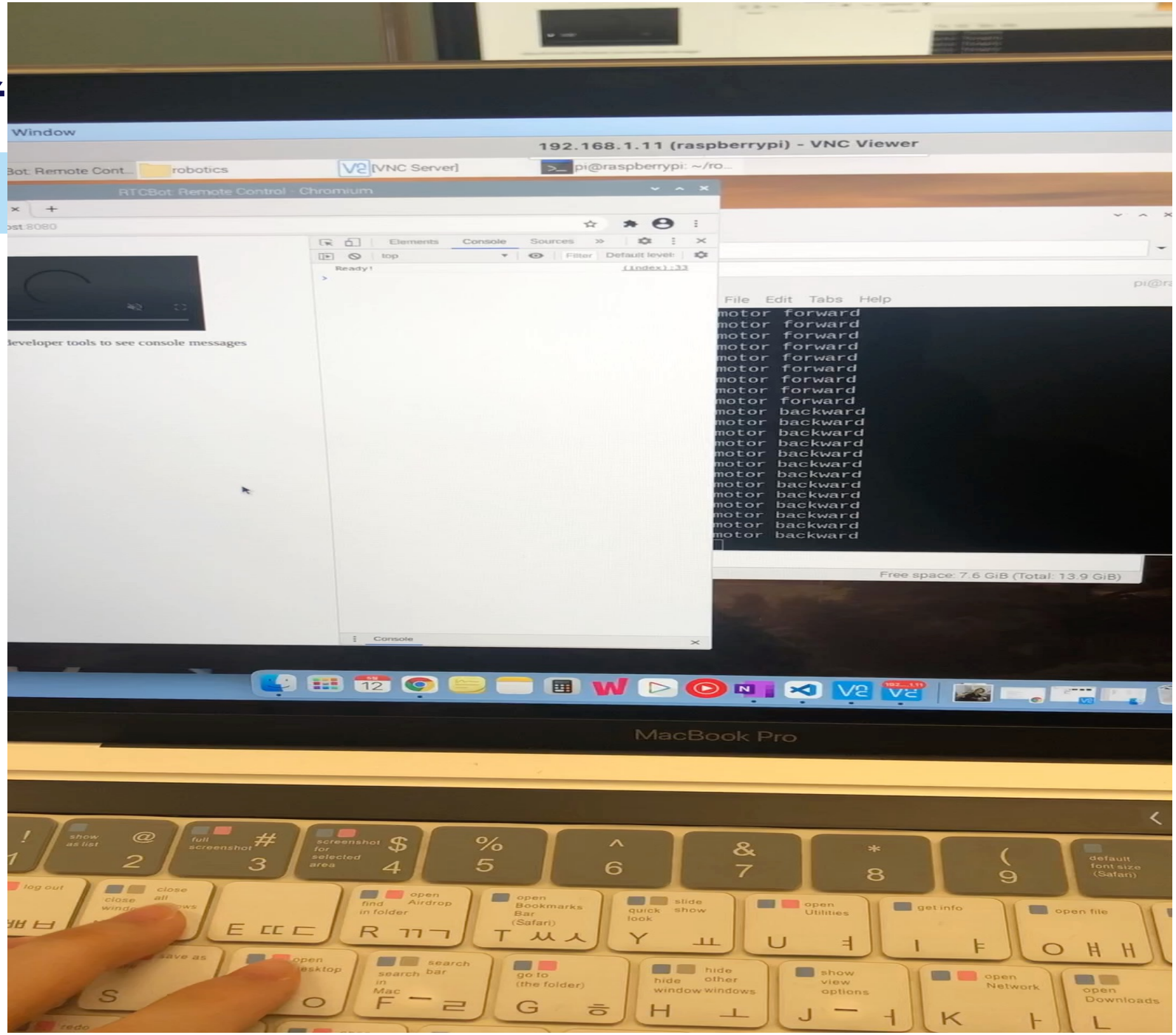
The documentation includes tutorials that guide in developing your robot, starting from a basic connection between a Raspberry Pi and Browser, and encompass creating a video-streaming robot controlled entirely over a 4G mobile connection, all the way to a powerful system that offloads complex computation to a desktop PC in real-time.

All communication happens through WebRTC, using Python 3's asyncio and the wonderful aiortc library, meaning that your robot can be controlled with low latency both from the browser and through Python, even when it is not connected to your local network.

The library is explained piece by piece in the documentation.

See Documentation & Tutorials

11주


Window
192.168.1.11 (raspberrypi) - VNC Viewer
robotics
[VNC Server]
RTCBot: Remote Control - Chromium
Elements
Console
Sources
Ready!
Developer tools to see console messages
File Edit Tabs Help
motor forward
motor backward
Free space: 7.6 GiB (Total: 13.9 GiB)
MacBook Pro

12주차 진행 목표

1. 받은 명령어를 통해 로봇 제어하기
2. custom web app 개선해 영상 스트리밍