

# 보안 프로그래밍 및 모의해킹- 서비스 포트 스캐너 구현

1 조 경기에서 남해까지

김용준 김주형 박지원 오성명 정민규

2024.04.12

# 목 차

|                         |    |
|-------------------------|----|
| 1. 개요 .....             | 3  |
| 1.1 프로젝트명 .....         | 3  |
| 1.2 프로젝트 기간 .....       | 3  |
| 1.3 프로젝트 목적 .....       | 3  |
| 1.4 프로젝트 기대 효과 .....    | 3  |
| 2. 프로젝트 범위 .....        | 4  |
| 2.1 작업명세서 .....         | 4  |
| 2.2 기능명세서 .....         | 5  |
| 3. 시스템 구축 환경 .....      | 6  |
| 3.1 하드웨어 구성도 .....      | 6  |
| 3.2 소프트웨어 구성도 .....     | 6  |
| 4. 프로젝트 추진체계 .....      | 7  |
| 4.1 프로젝트 수행 조직도 .....   | 7  |
| 4.2 조직별 역할 .....        | 7  |
| 5. 관리 프로세스 계획 .....     | 8  |
| 5.1 세부 일정 추진 계획 .....   | 8  |
| 5.2 단계별 산출물 .....       | 8  |
| 6. 프로젝트 관리 .....        | 9  |
| 6.1 프로젝트 작업 내용 .....    | 9  |
| 6.2 작업 내용 및 개인 회고 ..... | 12 |

## 1.1 프로젝트 명

본 프로젝트의 명칭은 "서비스 포트 스캐너 구현" 이라 한다.

## 1.2 프로젝트 기간

- 프로젝트 기간 : 2024.03.28 ~ 2024.04.12
- 시스템 오픈 일정 : 예정 일정 없음
- 전체 프로젝트 기간 : 2024.03.28 ~ 2024.04.12

## 1.3 프로젝트 목표

- 입력한 IP 주소를 검색 후 해당 IP 의 열려 있는 Port 스캔 (all-scanning)
- 본인 PC 를 스캔할 경우, Unknown port 에 관해 개인 코멘트 관리 가능
- Port 스캔 진행 시, 열려 있는 Port 에 관해 관련된 상세정보 제공
- 열려 있는 Port 에 대한 취약점 양지

## 1.4 프로젝트 기대 효과

- 내 PC SCAN 을 통해 로컬 PC IP 를 자동 취득, 편의성을 제공함
- 열린 Port 의 상세 정보를 제공함으로 서비스 효율성 제공
- 열린 Port 에 대한 취약점을 양지해 개인 PC 보안에 관한 관심도 높임

## 2. 프로젝트 범위

---

### 2.1 작업 명세서

#### 1. 서비스 포트 스캐너 구현

| 업무         | 업무 범위  |
|------------|--|
| 서비스 스캐너 구현 | <ul style="list-style-type: none"><li>- SYN 스캔에 대한 이해 및 SYN 스캐너 구현</li><li>- 서비스별 판별 방법 설계 및 구현</li><li>- 각 서비스 포트별 스캔에 관한 에러 핸들링</li><li>- 알려진 port 에 관한 DB 설계 및 구축</li><li>- Scan 버튼 클릭시 all-scanning 진행</li><li>- 내 PC 의 IP 주소 취득</li><li>- 열려있는 port 에 관한 Banner 정보 취득 후 취약점 혹은 정보 제공</li><li>- 알려진 port 가 아닌 경우, 개인이 작성한 코멘트 작성 및 정보 제공</li></ul> |
| 문서화        | <ul style="list-style-type: none"><li>- 개발 사양서 작성</li><li>- Notion 을 통한 진행 상황 공유</li></ul>   |

#### 2. 시스템 구축

| 업무     | 업무 범위  |
|--------|--|
| 시스템 구축 | <ol style="list-style-type: none"><li>1. 요구정의 단계<ul style="list-style-type: none"><li>- 개발 사양서 작성</li></ul></li><li>2. 분석 단계<ul style="list-style-type: none"><li>- 시스템 분석</li><li>- 개인 comment 에 관한 DB table 분석</li></ul></li><li>3. 설계 단계<ul style="list-style-type: none"><li>- 서비스 스캐너 설계</li></ul></li><li>4. 구현 단계<ul style="list-style-type: none"><li>- 서비스 스캐너 구현</li></ul></li></ol> |
| 테스트    | <ol style="list-style-type: none"><li>1. 테스트</li></ol>   |

|  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>- 단위 테스트 진행</li> <li>- 통합 테스트 진행</li> </ul> |
|--|--|

## 2.2 기능 명세서

|          |                     |                                  |   |
|----------|---------------------|----------------------------------|---|
| Back-end | well known port 스캐닝 | Multi Threading 구현               | 포트 스캔 속도 향상을 위해 Multi Threading 으로 스캔 코드 구현   |
|          |                     | Well Known port 정보 DB 구현         | port_number, service_name, port_description 에 대한 정보 mysql 에 입력.                           |
|          | unknown port 스캐닝    | Multi Threading 구현               | 포트 스캔 속도 향상을 위해 Multi Threading 으로 스캔 코드 구현   |
|          |                     | unknown port 에 대한 banner 정보 수집   | socket 통신을 통해 얻은 배너 정보를 바탕으로 열린 포트의 서비스가 무엇인지 추측.   |
|          | my scan             | 웹 사이트 접속자의 ip 채집                 | 해당 웹사이트 접속 시 접속자의 ip 를 찾고, 해당 ip 에 대한 스캔을 바로 실시   |
|          | 언노운 정보 저장           | 사용자가 포트에 대해 설명한 내용을 DB 에 전달 및 저장 | 언노운 포트의 경우, 배너 정보만으로 서비스 구별 불확실. 유저가 직접 이 용도로 사용하고 있음을 명시 할 시 해당 정보 db 에 저장 이후 불러옴.       |
|          |                     | 언노운 포트의 경우 ip 대조후 조회             | 언노운 포트의 유저의 설명이 들어가 있는 경우 접속자의 ip 와 검색 대상의 ip 가 동일한 경우에만 디비에 조회 후 출력.                     |
| Frontend | index.html          | 전달할 ip 입력                        | 검색하고자 하는 ip 입력 및 전달. 자신의 ip 를 상단에 보여주며 myscan 시 별도 입력 없이 본인의 ip 전달                        |
|          | result.html         | 스캔 결과 출력                         | 스캔 결과를 출력하고, 웹노운 포트의 경우 db 상에 저장된 정보를 출력. 언노운의 경우 배너 정보를 출력하고 사용자가 정리하여 내용을 넣을 수 있도록 칸 설계 |

### 3. 시스템 구축 환경

---

#### 3.1 하드웨어 구성도

- 비대면으로 진행되어 팀원 각 개인 PC 에서 진행하는 것으로 함

#### 3.2 소프트웨어 구성도

- 본 프로젝트 진행 시에 공통으로 사용하는 소프트웨어에 관한 스택 및 버전 정보로 변경함

|        |        |             |
|--------|--------|-------------|
| Server | Ubuntu | 22.04.4     |
|        | Flask  | 3.0.2       |
| Source | git    | 2.34.1      |
| Client | Python | 3.10.12     |
| Brower | Edge   | xx.xx.xx 이상 |
|        | Chrome | xx.xx.xx 이상 |
|        | Safari | xx.xx.xx 이상 |
| DB     | MySQL  | 8.0.35      |

## 4. 프로젝트 추진체계

### 4.1 프로젝트 수행 조직도

| 팀장  | 팀원  | 팀원  | 팀원  | 팀원  |
|-----|-----|-----|-----|-----|
| 박지원 | 김용준 | 김주형 | 오성명 | 정민규 |

### 4.2 조직별 역할

#### 4.2.1 서비스 스캐너 구현

| 작업내용                | 주담당        | 관련 기능                                      |
|---------------------|------------|--|
| Front-end 개발        | 오성명        | Front-end                                  |
| Front 기능 연결         | 오성명<br>김용준 | Front-end                                  |
| DB 생성 및 구축          | 김주형        | Well-known port 정보 및 Unknown port 코멘트 불러오기 |
| DB 와 연동된 API 제작     | 김주형<br>김용준 | Well-known port 정보 및 Unknown port 코멘트 불러오기 |
| API 연결 및 값 처리       | 김용준<br>박지원 |  |
| 접속 PC IP 취득 및 처리    | 박지원<br>김용준 | 본인 PC port scan 을 위한 IP 취득                 |
| SACN 후 Banner 정보 처리 | 박지원        | Well-known port 가 아닌 경우 정보 취득 후 출력         |
| 멀티스레딩               | 정민규        | Full scanning 기능 최적화                       |
| 에러 핸들링 추가 및 수정      | 정민규        | Error 발생시 처리 구현                            |
| 서버 기본 세팅            | 정민규        |  |

#### 4.2.2 문서화

| 작업내용               | 이름         |
|--------------------|------------|
| 개발 사양서 수정 및 작성     | 김용준<br>오성명 |
| 개발 보고서 및 notion 정리 | 박지원        |

|                 |     |
|-----------------|-----|
| 발표 ppt 제작       | 김주형 |
| 발표 및 발표 스크립트 작성 | 정민규 |

## 5. 관리 프로세스 계획

### 5.1 세부 일정 추진 계획

| 단계 | 활동                | 28 | 29 | 30 | 31 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|----|-------------------|----|----|----|----|---|---|---|---|---|---|---|---|---|----|----|----|
| 계획 | 개발 사양서            |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
|    | 기능 설계서            |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
| 분석 | 소켓 통신 및 TCP/IP 이해 |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
|    | 멀티 스레딩 이해         |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
|    | SYN 스캔 이해         |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
| 설계 | 서비스별 판별 방법        |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
|    | 서비스별 에러 핸들링       |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
| 구현 | 서비스별 코드 구현        |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |
|    | 포트 및 서비스 스캐너      |    |    |    |    |   |   |   |   |   |   |   |   |   |    |    |    |

### 5.2 단계별 산출물

| 단계 | 산출물        | 완료일        | 비고  |
|----|------------|------------|---|
| 계획 | 개발 사양서     | 2024.04.01 | <a href="https://www.notion.so/goorm/1271e1e9c21d4ae49f2becaa4f50f662">https://www.notion.so/goorm/1271e1e9c21d4ae49f2becaa4f50f662</a>                 |
|    | 기능 설계서     | 2024.04.01 | 개발 사양서 내 첨부   |
| 분석 | SYN 이해     | 2024.04.02 | <a href="https://www.notion.so/goorm/Day-3-4-1eb4b21b55e94b589651e5f02f347126">https://www.notion.so/goorm/Day-3-4-1eb4b21b55e94b589651e5f02f347126</a> |
| 설계 | 서비스별 판별 방법 | 2024.04.04 | <a href="https://www.notion.so/goorm/Day-5-6-38464ba7f05e467ca2ffffe0ddbdadb5">https://www.notion.so/goorm/Day-5-6-38464ba7f05e467ca2ffffe0ddbdadb5</a> |
| 구현 | 서비스별 코드 구현 | 2024.04.05 | <a href="https://www.notion.so/goorm/Day-5-6-38464ba7f05e467ca2ffffe0ddbdadb5">https://www.notion.so/goorm/Day-5-6-38464ba7f05e467ca2ffffe0ddbdadb5</a> |



## 6. 프로젝트 관리

---

### 6.1 프로젝트 보고 계획

| 구분   | 보고내용  | 보고주기 및 기한   | 참석자                              |
|------|---|---|----------------------------------|
| 일일보고 | <ul style="list-style-type: none"><li>➤ 전일 작업한 내용 보고</li><li>➤ 전일 작업 중 발생했던 이슈 보고 및 관리</li><li>➤ 당일 작업할 내용 보고</li><li>➤ 변경된 내용 공유</li></ul> | 2024.03.28 ~<br>2024.04.12<br><br>매일 오전 11 시,<br>오후 6 시 (2 회) | 김용준, 김주형<br>박지원, 오성명<br>정민규      |
| 중간보고 | <ul style="list-style-type: none"><li>➤ 당일 기준 진행된 내용 보고</li></ul>   | 2024.04.08  | 정보보호 4 회차<br>플레이어 전원,<br>강기웅 멘토님 |
| 최종보고 | <ul style="list-style-type: none"><li>➤ 프로젝트 완료 내역 보고</li></ul>   | 2024.04.12  | 정보보호 4 회차<br>플레이어 전원,<br>강기웅 멘토님 |

### 6.2 작업 내용 및 개인 회고

#### 6.2.1 작업 내용

- 코드와 관한 내용은 하기 링크에 첨부함

<https://www.notion.so/goorm/f4b82757bffe49e081972043b048a30d>

a. 메인 홈페이지

## GN Port Scanner

My IP Address: 127.0.0.1

Enter IP Address

SCAN My IP Scan

### b. 검색 후 열려있는 port 조회

## GN Port Scan Result

| Port | Status | Service                               | Description  | Note |
|------|--------|---------------------------------------|--|------|
| 137  | closed | NETBIOS Name Service                  | NETBIOS Name Service는 네트워크 이름을 IP 주소로 해석합니다.                           | 🔍    |
| 445  | closed | Microsoft DS (Direct Host)            | Microsoft DS (Direct Host)는 Microsoft Windows 파일 및 프린터 공유를 위한 프로토콜입니다. | 🔍    |
| 69   | closed | TFTP (Trivial File Transfer Protocol) | TFTP (Trivial File Transfer Protocol)는 간단한 파일 전송을 위한 프로토콜입니다.          | 🔍    |
| 1    | closed | Echo                                  | Echo 서비스는 클라이언트가 보낸 데이터를 그대로 다시 반환합니다.                                 | 🔍    |
| 49   | closed | Login Host Protocol                   | Login Host Protocol은 원격 로그인을 제공합니다.                                    | 🔍    |
| 70   | closed | Gopher                                | Gopher는 텍스트 기반의 문서와 파일을 검색하기 위한 프로토콜입니다.                               | 🔍    |
| 67   | closed | Bootstrap Protocol Server             | Bootstrap Protocol은 네트워크 부팅을 위한 IP 할당을 제공합니다.                          | 🔍    |
| 68   | closed | Bootstrap Protocol Client             | Bootstrap Protocol은 네트워크 부팅을 위한 IP 할당을 제공합니다.                          | 🔍    |
| 17   | closed | QOTD                                  | Quote of the Day (QOTD) 서비스는 무작위 인용구를 제공합니다.                           | 🔍    |
| 7    | closed | Echo                                  | Echo 서비스는 클라이언트가 보낸 데이터를 그대로 다시 반환합니다.                                 | 🔍    |
| 18   | closed | Message Send Protocol                 | Message Send Protocol은 간단한 메시지 전송을 위한 프로토콜입니다.                         | 🔍    |
| 37   | closed | Time                                  | Time 서비스는 현재 시간을 제공합니다.  | 🔍    |
| 80   | closed | HTTP (Hypertext Transfer Protocol)    | HTTP (Hypertext Transfer Protocol)는 웹 서버와 클라이언트 간의 통신을 제공합니다.          | 🔍    |
| 21   | closed | FTP - 제어                              | FTP (File Transfer Protocol)는 파일 전송을 위한 프로토콜입니다.                       | 🔍    |
| 11   | closed | Sysstat                               | Sysstat 서비스는 서버의 현재 시스템 상태를 제공합니다.                                     | 🔍    |
| 79   | closed | Finger                                | Finger 서비스는 사용자 정보를 조회합니다.   | 🔍    |

### c. 해당 돋보기 아이콘을 통해 해당 port 정보 조회 가능

GN Port Scan Result

| Port | Status | Service                               | Description  | Note   |
|------|--------|---------------------------------------|--|--|
| 137  | closed | NETBIOS Name Service                  | NETBIOS Name Service는 네트워크 이름을 IP 주소로 해석합니다.                           | ㄱ  |
| 445  | closed | Microsoft-DS (Direct Host)            | Microsoft-DS (Direct Host)는 Microsoft Windows 파일 및 프린터 공유를 위한 프로토콜입니다. | ㄱ  |
| 69   | closed | TFTP (Trivial File Transfer Protocol) | TFTP (Trivial File Transfer Protocol)는 간단한 파일 전송을 위한 프로토콜입니다.          | ㄱ  |
| 1    | closed | Echo                                  | Echo 서비스는 클라이언트가 보낸 데이터를 그대로 다시 반환합니다.                                 | ㄱ  |
| 49   | closed | Login Host Protocol                   | Login Host Protocol은 원격 로그인 제공을 제공합니다.                                 | ㄱ  |
| 70   | closed | Gopher                                | Gopher는 텍스트 기반의 문서와 파일을 검색하기 위한 프로토콜입니다.                               | ㄱ  |
| 67   | closed | Bootstrap Protocol Server             | Bootstrap Protocol은 네트워크 부팅을 위한 IP 할당을 제공합니다.                          | ㄱ  |
| 68   | closed | Bootstrap Protocol Client             | Bootstrap Protocol은 네트워크 부팅을 위한 IP 할당을 제공합니다.                          | ㄱ  |
| 17   | closed | QOTD                                  | Quote of the Day (QOTD) 서비스는 무작위 인용구를 제공합니다.                           | ㄱ  |
| 7    | closed | Echo                                  | Echo 서비스는 클라이언트가 보낸 데이터를 그대로 다시 반환합니다.                                 | ㄱ  |
| 18   | closed | Message Send Protocol                 | Message Send Protocol은 간단한 메시지 전송을 위한 프로토콜입니다.                         | Echo 서비스는 보안 경사를 거치지 않고 모든 데이터를 그대로 반환하므로 악성 코드를 전송하거나 시스템에 대한 공격을 수행하는 데 사용될 수 있습니다. 이는 시스템의 안정성과 보안을 저해할 수 있습니다. |
| 37   | closed | Time                                  | Time 서비스는 현재 시간을 제공합니다.  |  |
| 80   | closed | HTTP (Hypertext Transfer Protocol)    | HTTP (Hypertext Transfer Protocol)는 웹 서버와 클라이언트 간의 통신을 제공합니다.          |  |
| 21   | closed | FTP - 제어                              | FTP (File Transfer Protocol)는 파일 전송을 위한 프로토콜입니다.                       | ㄱ  |
| 11   | closed | Systat                                | Systat 서비스는 서버의 현재 시스템 상태를 제공합니다.                                      | ㄱ  |
| 79   | closed | Finger                                | Finger 서비스는 사용자 정보를 조회합니다.   | ㄱ  |
| 67   | closed | Bootstrap Protocol Server             | Bootstrap Protocol은 네트워크 부팅을 위한 IP 할당을 제공합니다.                          | ㄱ  |
| 68   | closed | Bootstrap Protocol Client             | Bootstrap Protocol은 네트워크 부팅을 위한 IP 할당을 제공합니다.                          | ㄱ  |
| 17   | closed | QOTD                                  | Quote of the Day (QOTD) 서비스는 무작위 인용구를 제공합니다.                           | ㄱ  |
| 7    | closed | Echo                                  | Echo 서비스는 클라이언트가 보낸 데이터를 그대로 다시 반환합니다.                                 | ㄱ  |
| 18   | closed | Message Send Protocol                 | Message Send Protocol은 간단한 메시지 전송을 위한 프로토콜입니다.                         | ㄱ  |
| 37   | closed | Time                                  | Time 서비스는 현재 시간을 제공합니다.  | ㄱ  |
| 80   | closed | HTTP (Hypertext Transfer Protocol)    | HTTP (Hypertext Transfer Protocol)는 웹 서버와 클라이언트 간의 통신을 제공합니다.          | ㄱ  |
| 21   | closed | FTP - 제어                              | FTP (File Transfer Protocol)는 파일 전송을 위한 프로토콜입니다.                       | FTP는 데이터 및 인증 정보를 암호화하지 않고 전송하므로 중간자 공격 및 패킷 스니핑과 같은 위험이 있습니다. 또한, 악의적인 사용자가 파일 전송을 위조하여 악성 코드를 전송할 수 있습니다.        |
| 11   | closed | Systat                                | Systat 서비스는 서버의 현재 시스템 상태를 제공합니다.                                      |  |
| 79   | closed | Finger                                | Finger 서비스는 사용자 정보를 조회합니다.   |  |
| 95   | closed | Supdup                                | Supdup 서비스는 원격 터미널 액세스를 제공합니다.   | ㄱ  |
| 9    | closed | Discard                               | Discard 서비스는 클라이언트가 보낸 데이터를 버립니다.                                      | ㄱ  |
| 43   | closed | Whois                                 | Whois 서비스는 도메인 등록 정보를 조회합니다.   | ㄱ  |
| 53   | closed | DNS (Domain Name System)              | DNS (Domain Name System)은 도메인 이름을 IP 주소로 해석합니다.                        | ㄱ  |
| 71   | closed | NETRJS protocol                       | NETRJS protocol은 원격 작업 실행을 제공합니다.                                      | ㄱ  |
| 42   | closed | Host Name Server Protocol             | Host Name Server Protocol은 호스트 이름과 IP 주소를 매핑합니다.                       | ㄱ  |
| 19   | closed | Character Generator Protocol          | Character Generator Protocol은 무작위 문자열을 생성하여 보냅니다.                      | ㄱ  |

d. Unknown-port 에 개인 comment 작성 및 수정 가능

| Port  | Status | Service                                   | Description |
|-------|--------|---|-------------|
| 1119  | open   | HTTP/1.1 400 Bad RequestConnection: close | None        |
| 5000  | open   | <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTM    | None        |
| 5678  | open   | <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTM    | None        |
| 6090  | open   | HTTP/1.0 400 Bad RequestContent-Type: tex | None        |
| 8380  | open   | HTTP/1.1 200 OKConnection: closeContent-T | None        |
| 33060 | open   |   | None        |

[Submit](#)

### 6.2.2 개인 회고

|     |  |
|-----|--|
| 김용준 | <p>온라인으로 진행해본 첫 프로젝트였고, 이번 과정을 통해 처음 알게 된 분야의 개발이어서 걱정이 많았다. 하지만 우려와 달리 서로가 서로의 부족한 점을 채워 나가면서 최종 결과물을 만들어 낼 수 있었다. 기본 기능 외의 팀원들과 소통하며 생각한 기능이 있었지만, 완전히 다 구현하지 못해 못내 아쉬움이 남는다.</p> <p>첫 시작이라 아쉬움이 남지만 그 경험을 바탕으로 이후의 프로젝트에서는 보다 나은 과정을 통해 보다 매끄러운 결과를 만들어 낼 수 있을 것이라는 자신감이 생겼다.</p> |
| 김주형 | <p>오직 비대면으로만 협업하는 과정을 처음 경험해서 걱정했는데, 팀원 모두가 능동적으로 역할과 책임을 맡았기 때문에 프로젝트가 잘 진행되지 않았나 생각한다. 또한, 처음 공통의 이해가 필요한 부분을 점검하고 진행하여서 더 매끄럽게 진행되었다고 생각한다.</p>   |

|     |  |
|-----|--|
|     | <p>공통 이해와 목표 설정의 중요성에 대해 알 수 있는 경험이었다고 생각한다.</p> <p>또한, 포트 번호에 대한 정보를 검색하는 과정에서 다양한 포트와 취약점에 대해 자세히 배울 수 있었다.</p> <p>다시 한번 백 번 듣는 것보다 한번 보는 것이, 백 번 보는 것보다 한번 해보는 것이 더 배움에 도움이 된다고 느꼈다.</p>  |
| 박지원 | <p>처음 서비스 스캐너를 구현하라고 했을 때 내가 구현할 수 있을까 막막한 마음이었다. 게다가 팀장으로 팀원을 이끌고 가려면 내가 더 아는 게 많아야만 하지 않을까 하는 부담감도 있었다. 그렇지만 프로젝트가 시작되고 나서는 팀원들 모두가 나서서 본인이 아는 정보를 공유하고 하는 과정에서 다른 사람들은 이런 방식으로 이해하고 접근했구나, 내가 아는 정보가 전부가 아니었구나를, 팀장이라고 팀원들을 끌어 나가야만 하는 것이 아니라 서로 함께 배워 나가는 것임을 깨닫는 시간이었다.</p> <p>또한 구현해 나가는 과정에 있어서 팀원들과 소통하는 시간은 걱정했던 것과 달리 매우 즐거웠고 또 얻어가는 게 많았다고 생각한다.</p> <p>스캐너를 구현하기 위해서 다시 한번 알던 지식을 정리하고 공유하는 과정에서 더 확실히 기초를 다질 수 있었고 각 역할을 나누어 개발하는 과정에서는 협업을 하기 위해 필요한 것들을 배울 수 있었다.</p> <p>이번 프로젝트를 진행하며 배운 것들을 바탕으로 다음 프로젝트에는 겁먹지 않고 조금 더 자신감 있게 의견을 제안하고, 프로젝트를 시작하기 전에 조금 더 배경지식을 쌓아가야겠다고 느끼게 된 소중한 시간이었다.</p> |
| 오성명 | <p>처음에는 프로젝트를 어떻게 시작해야될지 막막했지만 서로 어떻게 구현하면 좋을지 어떤 기능을 넣으면 좋을지에 대해서 서로의 생각을 말하면서 점차 프로젝트를 진행할 수 있었다. 부족한 부분도 있었고 오류가 발생해서 안되는 부분도 있었지만 의지할 수 있는 팀원들이 있어서 좋았고 다음 프로젝트에서는 처음보다 더 발전한 모습이 되길..</p> <p>파이팅!</p>   |
| 정민규 | <p>비전공자로서 정말 아는거 하나 없이 시작한 이번 프로젝트에서 AWS 를 관리해보고 코드를 짜보면서 정말 모르는 것도 많았고 헤맨 것도 많았는데, 친절하게 알려주신 팀원들 덕분에 잘 끝마칠 수 있었던 것 같다.</p> <p>멀티스레딩을 적용하고 에러 핸들링을 맡기로 했었는데 기능들을 구현하면서 알고 있던 내용들을 다시 재정립할 수 있는 기회가 되었고, 처음이라 걱정스럽기도 하고, 아쉬웠던 점도 있었지만 좋은 경험이 되어 또 어떤 일의 발판이 될 수 있을거라고 생각한다.</p>   |