

두려움을 원동력으로 만드는
이종혁입니다.

목차

Contents

01	이력사항	P.03
02	스킬 및 역량	P.04
03	참여 프로젝트	P.05
04	자기소개	P.4
05	수료 내역	P.4

01. 이력사항

이종혁

| 2000.01.15 | 010-7605-5208 | ljh000115@gmail.com |

• 학력사항

2024. 03 ~ 08 KG 에듀원 하이브리드 멀티크라우드 아키텍처

- 자격사항

• 경력사항

2024. 09 – 2025. 09 (주)산엔지니어링 (소프트웨어 엔지니어)

2021. 12 – 2024. 02 다이소 매니저 (아르바이트)

2022. 06 - 2022. 08 이천 SK 하이닉스 공장 건설(보온직 조공)

02. 스킬 및 역량

빅데이터 & 클라우드 서버 & 코딩

[보유 기술]

Language

Python, SQL, HTML, CSS, R studio, SPSS, C, C#, C++, Qml, Skript

OS

Window, Linux

Framework+Library

Flask, tensorflow, Pandas, Numpy, BeautifulSoup

Data Base

Toad for SQL Server, MySQL workbench

Development Tool

Jupyter Notebook, Visual Studio, Visual Studio Code, Anaconda3, Sypder IDE, R studio, SPSS, STM32 IDE, Qt Creator

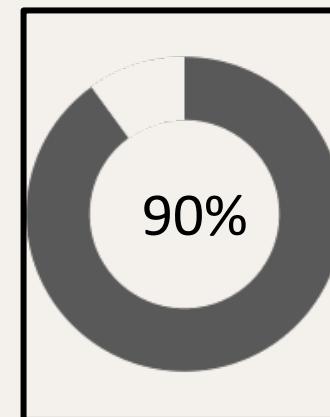
Cloud

GCP, AWS

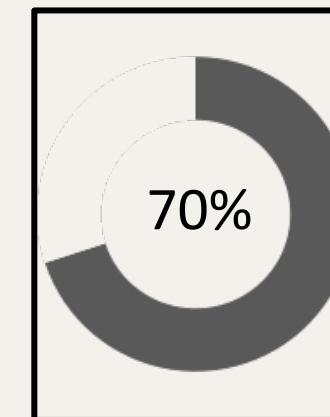
ETC

Powerpoint, Excel, Hanword, Notion, Draw.io, Cisco Packet tracer, MobaXterm

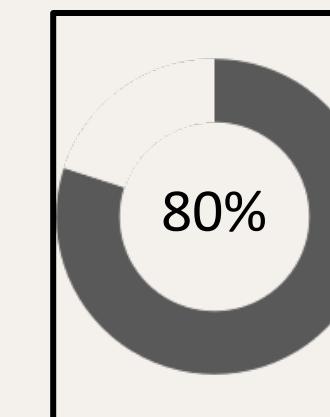
[강점 역량]



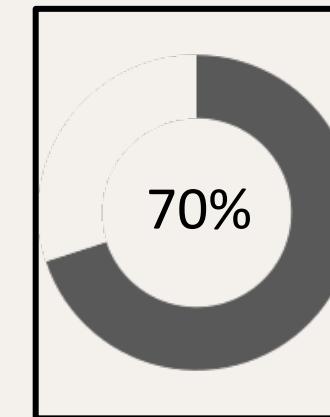
포용력



기획력



분석력



설득력

03. 참여프로젝트

1. Site-to-site vpn 3tire 멀티클라우드

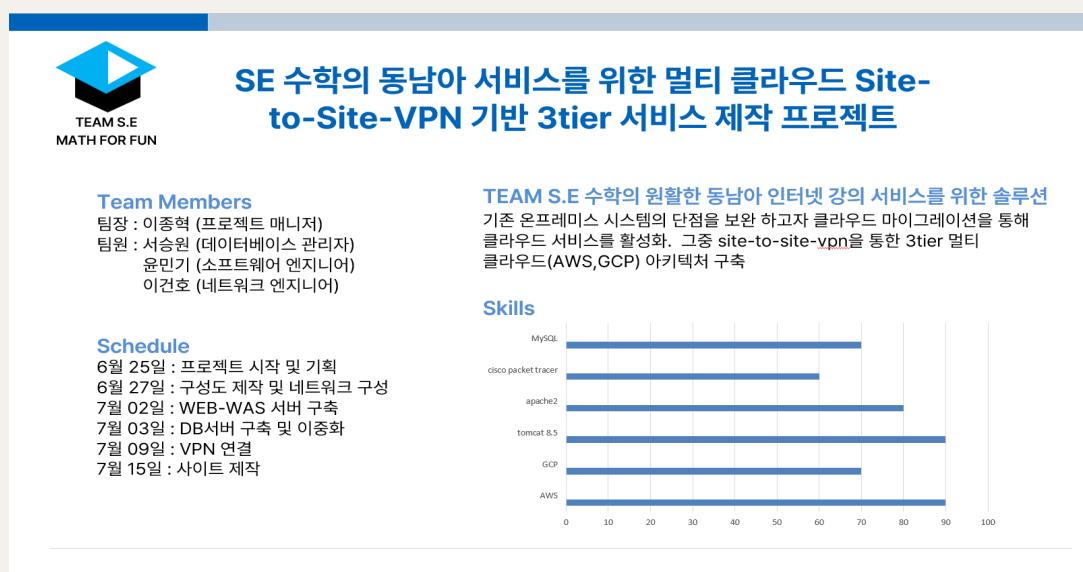
팀원 이종혁(팀장), 서승원, 윤민기, 이건호

역할 Project Manager
DataBase 관리자

제작기간 2024. 06. 25 - 2024. 07. 19

사용 기술

AWS, GCP, Ubuntu, MySQL, Apache2, Tomcat, Slack, CISCO Packet Tracer, MobaXterm, Python, Notion



2. Flask를 이용한 docker-compose 웹 서비스

팀원 이종혁(팀장), 서승원, 윤민기, 이건호

역할 Project Manager
Docker File 제작
DB 제작

제작기간 2024. 08. 12 ~ 2024. 09. 13

사용기술

Docker, Terraform, Flask, Python, Html, MariaDB, GCP, MobaXterm, Github, Notion, Discord



03. 참여프로젝트

3. Python Shooting Game

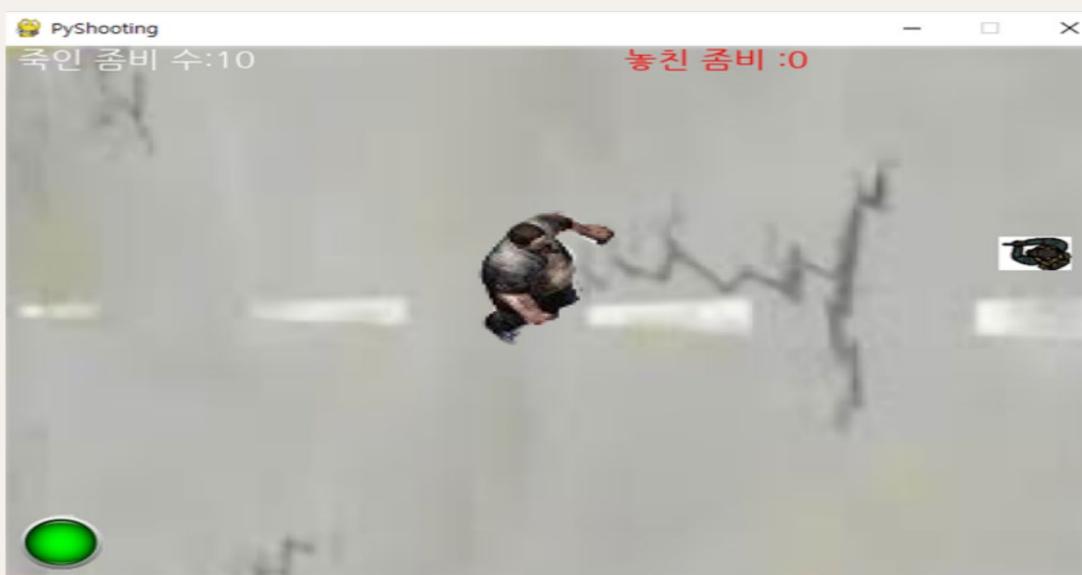
팀원 이종혁(팀장), 윤민기, 이건호

역할 Project Manager
Application Architecture

제작기간 2024. 03. 13 – 2024. 03. 22

사용기술

Visual Studio Code, Python, Pygame



4. 웹데이터 정보 및 실습

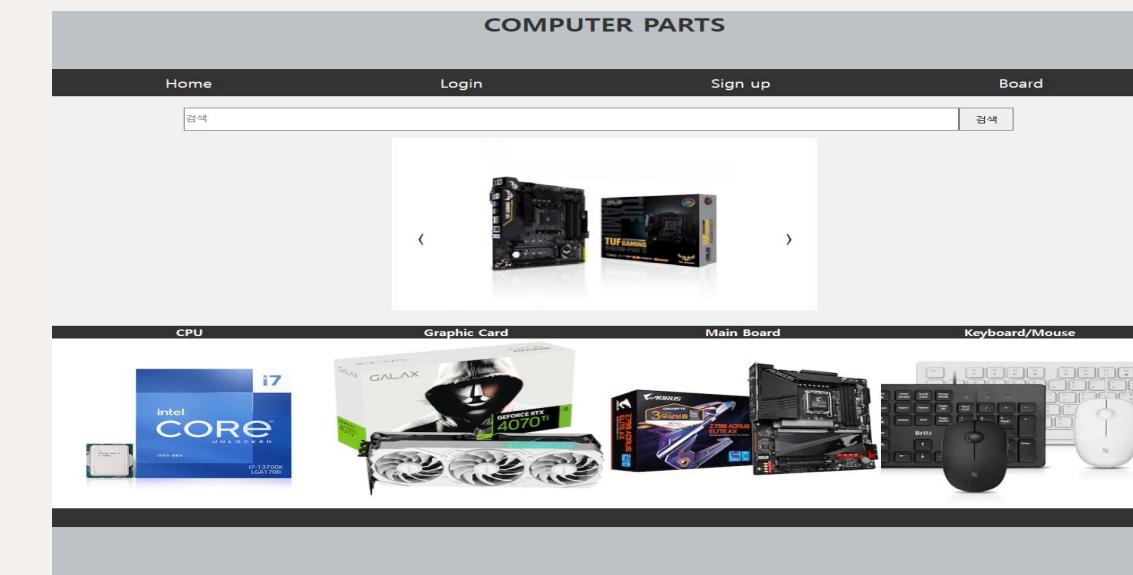
팀원 이종혁(팀장), 김훈종, 노승현, 장준혁

역할 Project Manager
웹 서버, Data base, Data base 서버 제작
게시판 페이지 제작

제작기간 2023. 03. 25 ~ 2023. 08. 01

사용기술

Html, CSS, Java Script, Toad for SQL, GCP,
MobaXterm, Python, Flask, Pymsql



03. 참여프로젝트

5. STM32 PID 제어

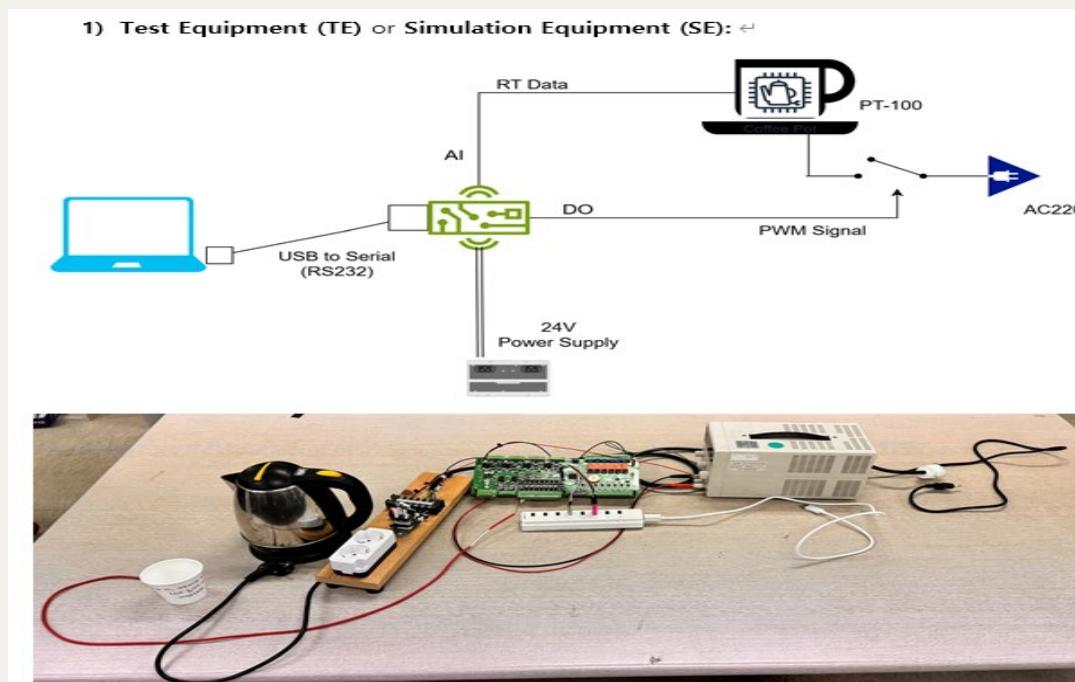
팀원 이종혁

역할 시스템 개발자
코드 테스터

제작기간 2024. 10. 14 – 2024. 12. 16

사용기술

STM32 IDE, C, Draw.io



6. CAN Protocol Monitoring Simulator

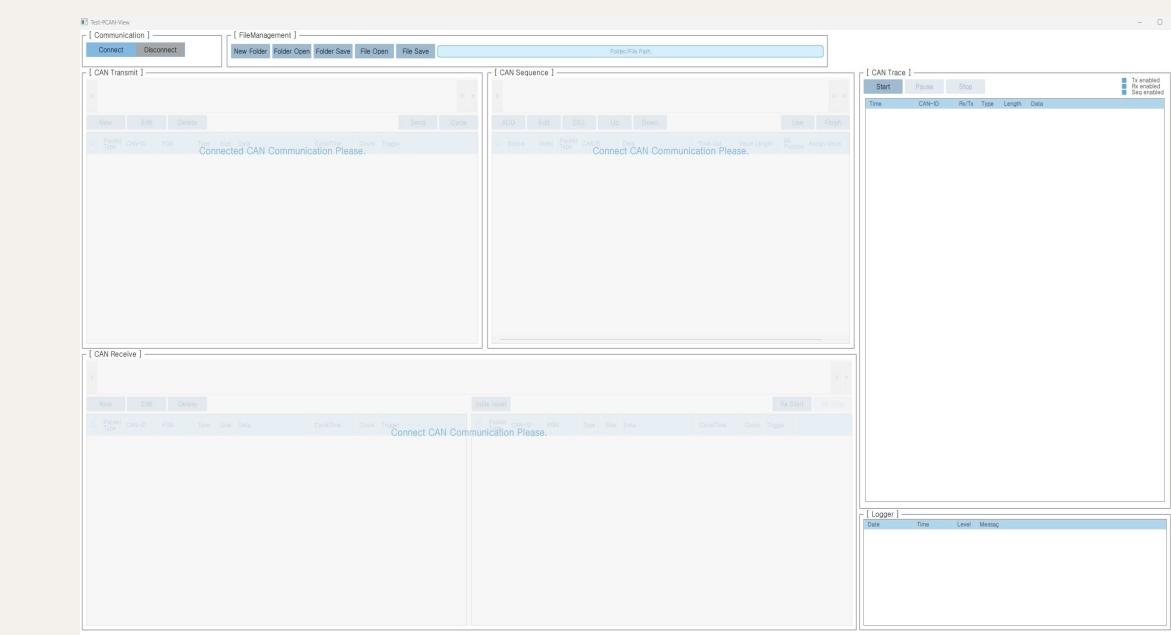
팀원 이종혁, 구승회

역할 CAN Protocol Monitoring Simulator의
Sequence 파트 제작
Trace 필터 제작

제작기간 2025. 04. 02 ~ 2025. 08. 15

사용기술

Qml, C++, PCAN_Basic API, GitHub, QT Creator,
Draw.io



TeamProject1

[Site-by-Site-VPN 3 tier 멀티 클라우드 아키텍처 제작]

OVERVIEW

- 동남아 시장 진출 및 원활한 서비스 제공을 위해 기존 온프레미스 환경을 클라우드 마이그레이션을 통해 동남아 지역에 멀티 클라우드 서비스 서비스 구축 및 온라인 운영

PERIOD

- 2024. 6. 25 ~ 2024. 7. 31 (약 37일)

TECHNOLOGY

- Environment : AWS, GCP, MySQL, Cisco Packet Tracer
- Language : html, jsp
- Tools & OS: ubuntu22.04, mysql, apach2, tomcat 8.5, slack, mobaxterm, notion, draw.io

MY ROLE (팀장 PM)

- 프로젝트 총괄
- GCP DB 파트 제작
- WAS 동영상 파트 제작
- 기획서 및 발표 ppt제작
- 발표



SE 수학의 동남아 서비스를 위한 멀티 클라우드 Site-by-Site-VPN 기반 3tier 서비스 제작 프로젝트

Team Members

- 팀장 : 이종혁 (프로젝트 매니저)
- 팀원 : 서승원 (데이터베이스 관리자)
- 윤민기 (소프트웨어 엔지니어)
- 이건호 (네트워크 엔지니어)

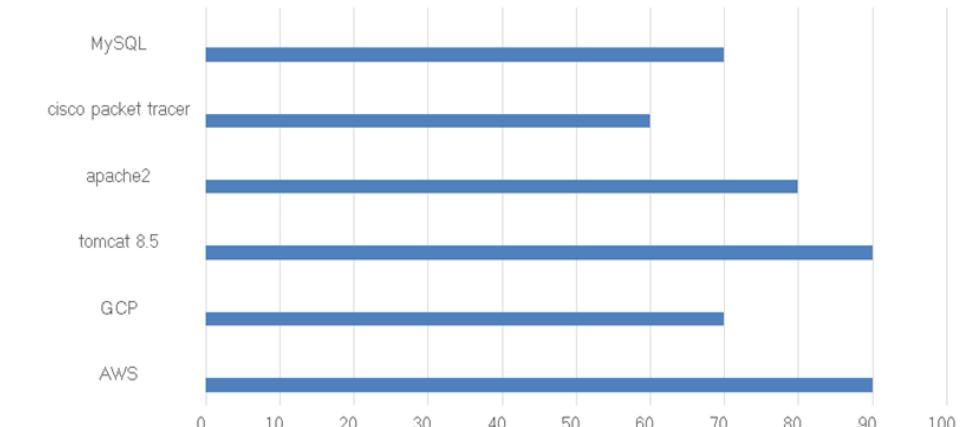
Schedule

- 6월 25일 : 프로젝트 시작 및 기획
- 6월 27일 : 구성도 제작 및 네트워크 구성
- 7월 02일 : WEB-WAS 서버 구축
- 7월 03일 : DB서버 구축 및 이중화
- 7월 09일 : VPN 연결
- 7월 15일 : 사이트 제작

TEAM S.E 수학의 원활한 동남아 인터넷 강의 서비스를 위한 솔루션

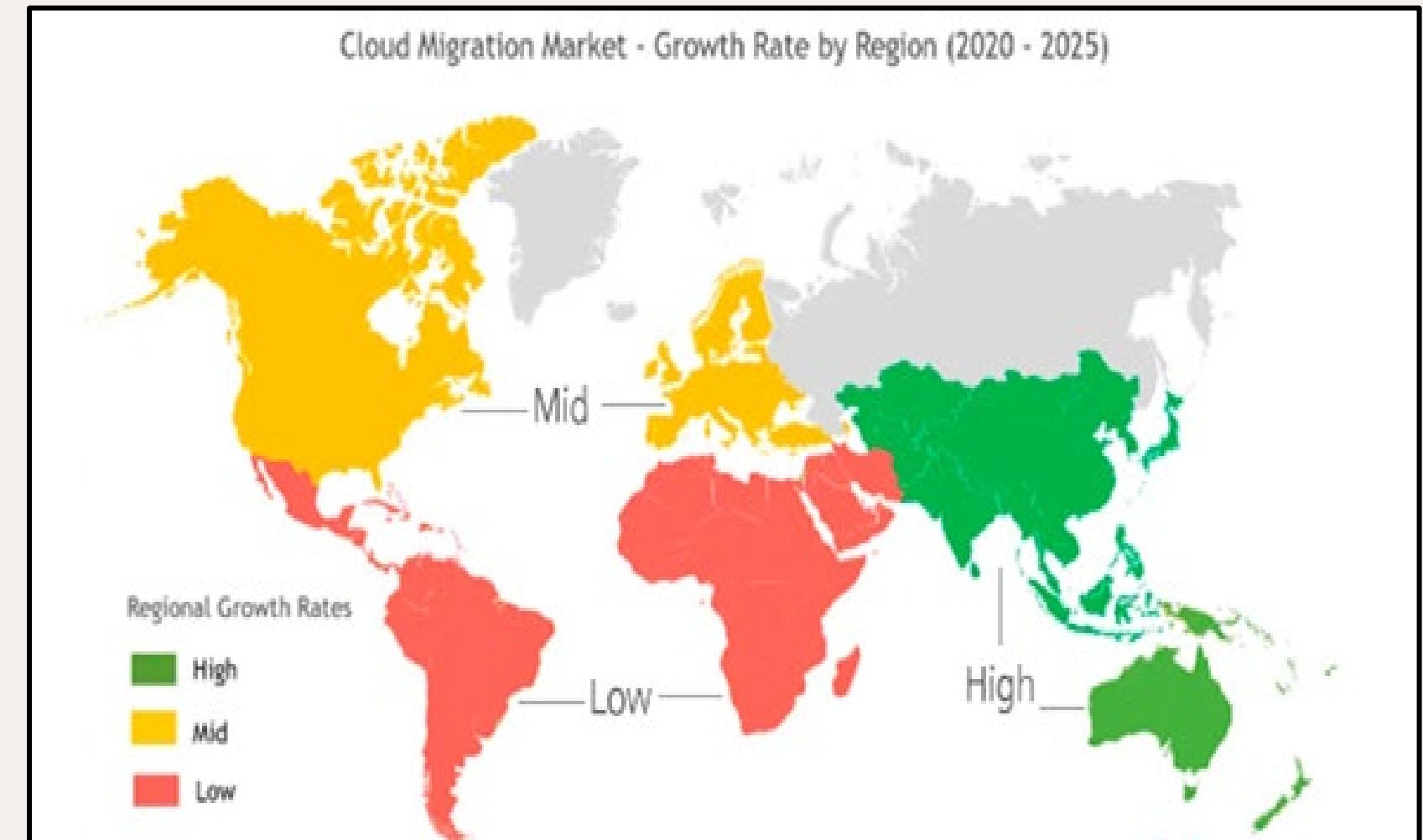
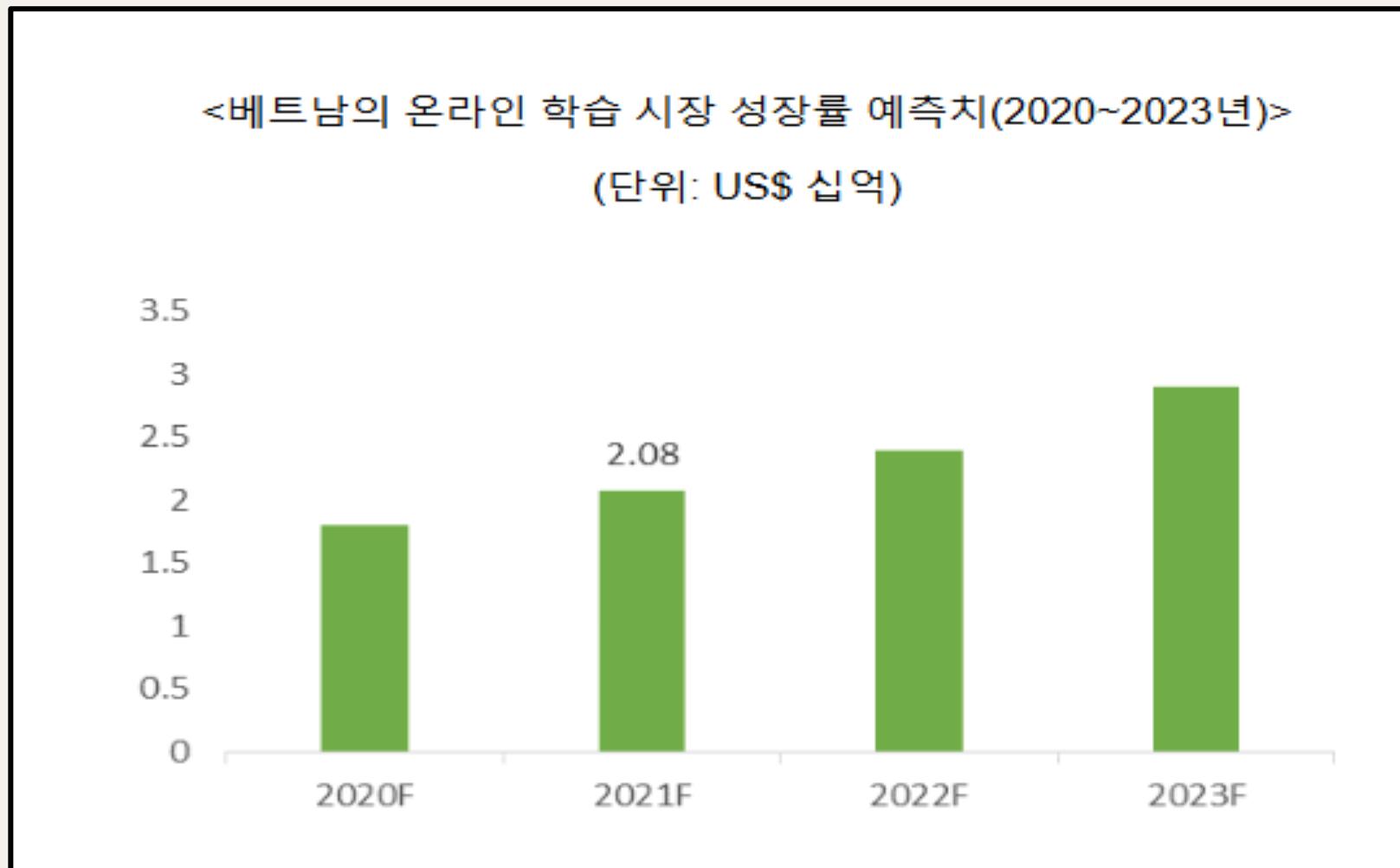
기존 온프레미스 시스템의 단점을 보완하고자 클라우드 마이그레이션을 통해 클라우드 서비스를 활성화. 그중 site-by-site-vpn을 통한 3tier 멀티 클라우드(AWS,VPN) 아키텍처 구축

Skills



TeamProject1

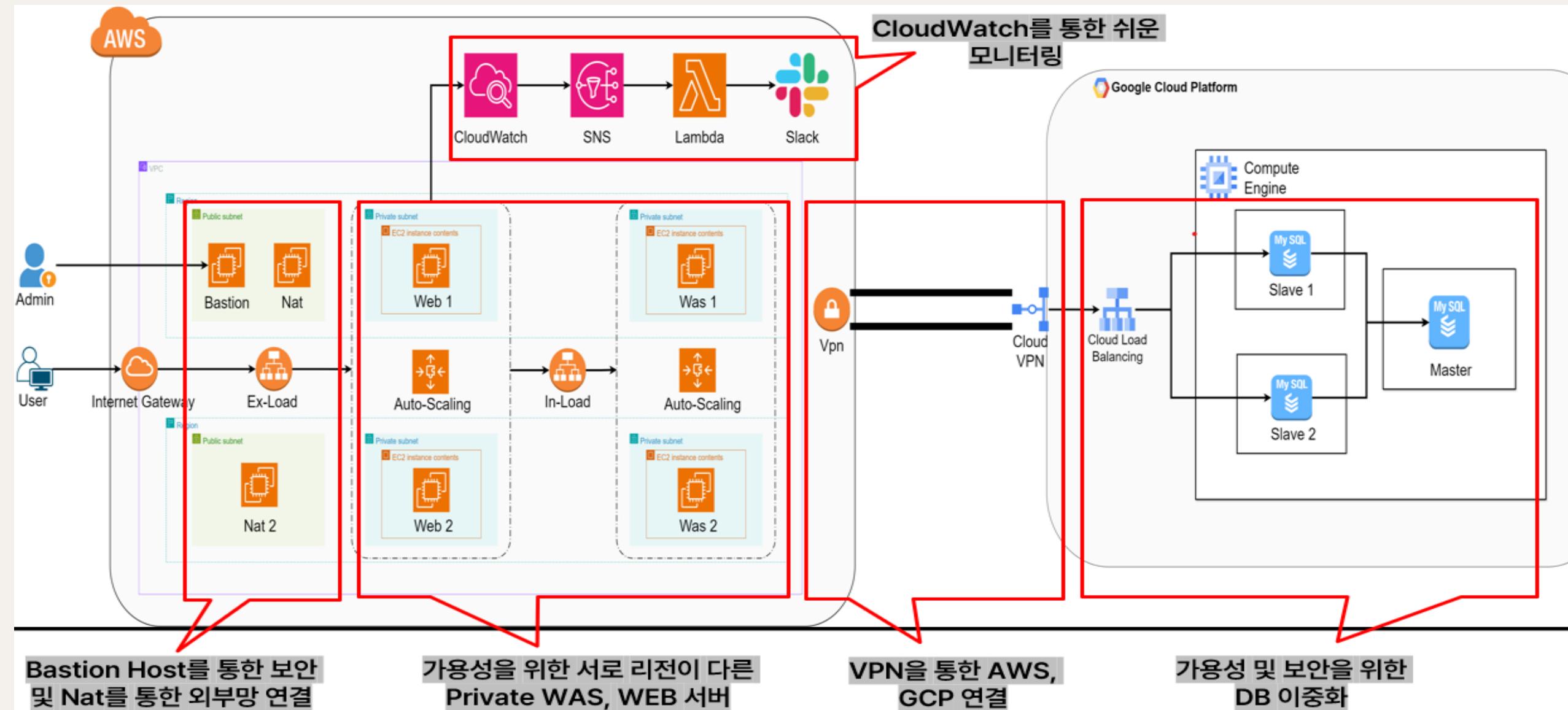
[시나리오]



동남아 온라인 학습 시장의 급성장으로 웹 강의 페이지의 트래픽이 급증, 이를 해결하기 위해 기존 온프레미스 시스템을 클라우드로 이전하고 멀티 클라우드 서비스를 도입하여 트래픽 증가에 효과적으로 대응하고, 서비스의 원활한 운영

TeamProject1

[구성도]

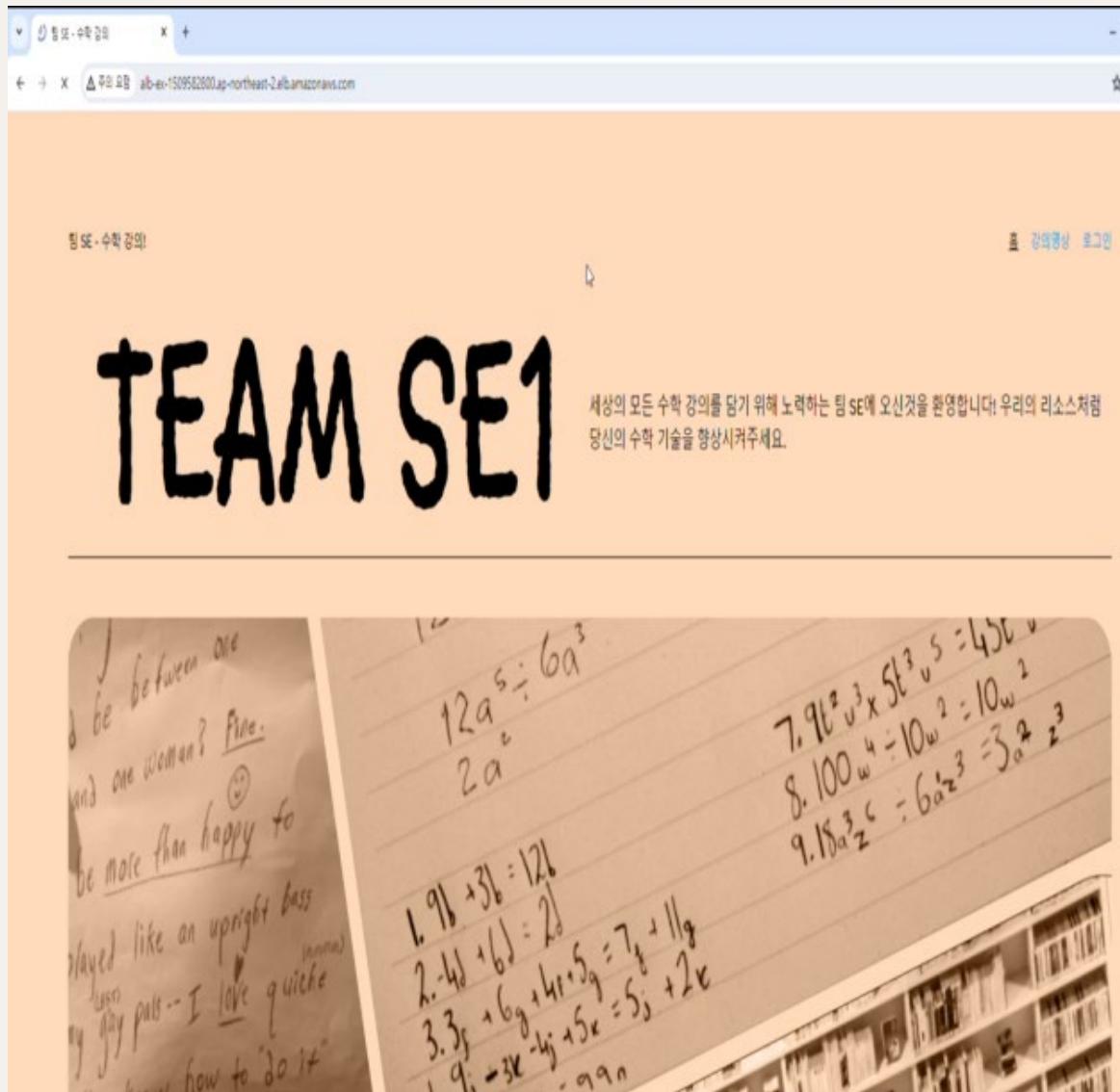


- 가용성 : 서비스 이중화, 가용 영역 분할, Auto Scaling, Cloud Watch를 통한 모니터링
- 비용 : NAT, 인스턴스 사용, 현지 오프레미스 시스템 대신 클라우드 서비스 활용
- 보안 : 보안그룹, VPN, Bastion Host을 통한 보안 강화

TeamProject1

[결과물]

웹 페이지 구성 및 ALB와 연동



회원 목록 페이지 구성 및 DB 연동

The screenshot shows a '회원 목록' (Member List) page. At the top, it says '회원 목록' and '// mysql database 연결'. Below is a table with columns: 번호 (Number), 이름 (Name), 전화번호 (Phone Number), 생성일 (Creation Date), and 관리 (Management). The table lists 15 members. At the bottom of the table, there are buttons for '수정' (Update) and '삭제' (Delete).

번호	이름	전화번호	생성일	관리
33	123	414111	2024-07-22 02:50:54.0	수정 삭제
32	teamse	9876543211	2024-07-22 02:01:57.0	수정 삭제
31	yigeonho	01044454127	2024-07-16 07:28:28.0	수정 삭제
29	Vu Minh Tien	0123-456-817	2024-07-16 07:01:19.0	수정 삭제
28	Hoang Thi Thuy	0123-456-816	2024-07-16 07:01:19.0	수정 삭제
27	Pham Van Dat	0123-456-815	2024-07-16 07:01:19.0	수정 삭제
26	Le Thi Bich	0123-456-814	2024-07-16 07:01:19.0	수정 삭제
25	Tran Duc Khanh	0123-456-813	2024-07-16 07:01:19.0	수정 삭제
24	Nguyen Thi Ly	0123-456-812	2024-07-16 07:01:19.0	수정 삭제
23	Quach Van Linh	0123-456-811	2024-07-16 07:01:19.0	수정 삭제
22	Vuong Thi Yen	0123-456-810	2024-07-16 07:01:19.0	수정 삭제
21	Tan Van Tuan	0123-456-809	2024-07-16 07:01:19.0	수정 삭제
20	Ly Thi Thao	0123-456-808	2024-07-16 07:01:19.0	수정 삭제
19	Chau Duc Huy	0123-456-807	2024-07-16 07:01:19.0	수정 삭제
18	Cao Thi Le	0123-456-806	2024-07-16 07:01:19.0	수정 삭제
17	Truong Van Quan	0123-456-805	2024-07-16 07:01:19.0	수정 삭제
16	Trinh Thi Ngoc	0123-456-804	2024-07-16 07:01:19.0	수정 삭제
15	Phan Minh Tam	0123-456-803	2024-07-16 07:01:19.0	수정 삭제

동영상 재생



TeamProject2

[Flask를 이용한 docker-compose 웹 서비스]

OVERVIEW

-Docker와 Terraform 기술을 활용한 자동 클라우드 아키텍트 제작
파일 생성

PERIOD

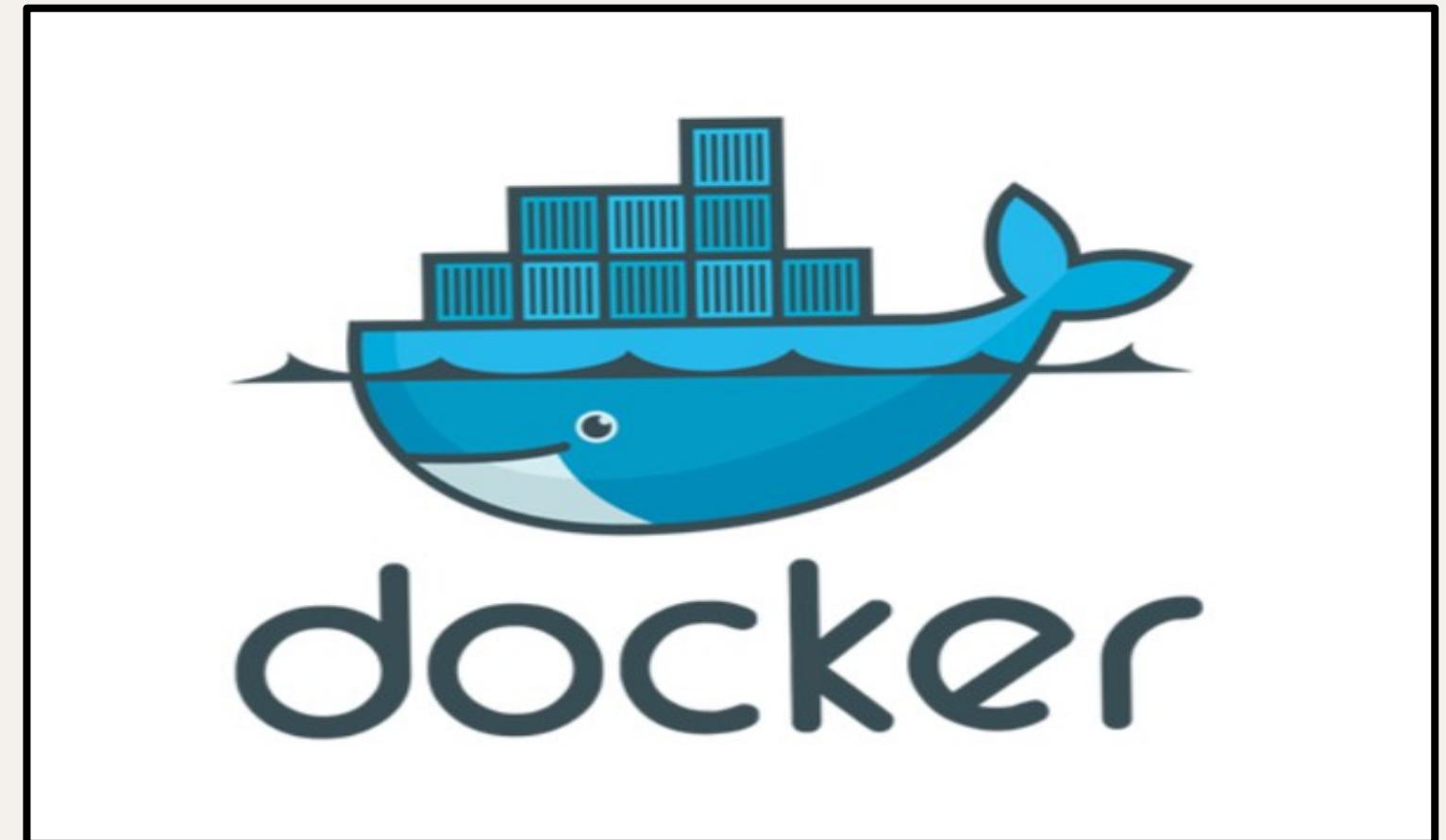
- 2024. 08. 12 ~ 2024. 09. 13 (약 33일)

TECHNOLOGY

- Environment : Docker, Terraform, Github, Flask, GCP
- Language : html, python, docker, sql
- Tools & OS: Docker, Terraform, GCP, maria db, mobaxterm, notion, discord

MY ROLE (팀장 PM)

- 프로젝트 총괄
- DB 생성 파일 제작
- Docker File 제작
- Flask app.py 제작



TeamProject2

[Terraform을 이용한 vpc,vm인스턴스 구성]

Terraform을 이용한 vm인스턴스 구성

VM 인스턴스							
필터 속성 이름 또는 값 입력							
<input type="checkbox"/> 상태	이름 	영역	권장사항	다음에서 사용 중:	내부 IP	외부 IP	연결
<input type="checkbox"/>	 master-instance	us-central1-a			172.16.31.2 (nic0)	34.42.221.80 (nic0)	SSH  
<input type="checkbox"/>	 worker1-instance	europe-west1-c			172.16.32.2 (nic0)	34.77.22.52 (nic0)	SSH  
<input type="checkbox"/>	 worker2-instance	asia-northeast3-a			172.16.33.2 (nic0)	34.64.78.161 (nic0)	SSH  

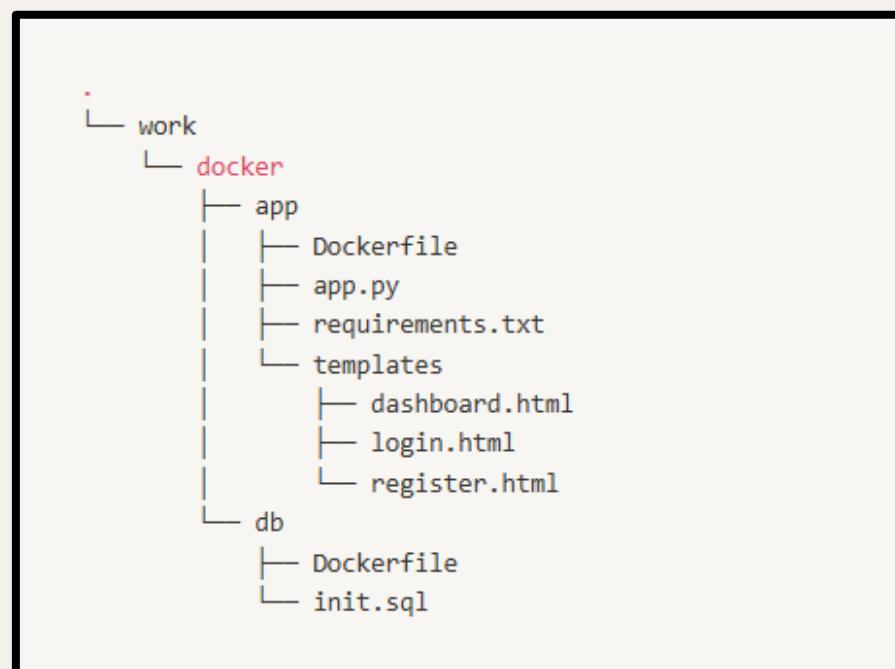
Terraform을 이용한 VPC 네트워크 구성

VPC 네트워크							
필터 속성 이름 또는 값 입력							
이름 	서브넷	MTU 	모드	IPv6 ULA 범위	게이트웨이	방화벽 규칙	전역 동적 라우팅
default	39	1460	자동			4  사용 안함	
se-vpc	3	1460	커스텀			1  사용 안함	

TeamProject2

[Docker 파일 구성]

파일 생성



Docker Hub에 Repositories 생성

Create repository

Namespace: ekchoi3912 Repository Name: mydb

Short description: A short description to identify your repository. If the repository is public, this description is used to index your content on Docker Hub and in search engines, and is visible to users in search results.

Visibility: Using 0 of 1 private repositories. [Get more](#)

Public Appears in Docker Hub search results Private Only visible to you

Create repository

Namespace: ekchoi3912 Repository Name: myapp

Short description: A short description to identify your repository. If the repository is public, this description is used to index your content on Docker Hub and in search engines, and is visible to users in search results.

Visibility: Using 0 of 1 private repositories. [Get more](#)

Public Appears in Docker Hub search results Private Only visible to you

DB 이미지 build 및 push

5-1 DB 이미지 build하기

```
docker build -t Docker허브계정/mydb:v1 ./db
```

<예시> --> 복사하지 마세요
docker build -t ekchoi3912/mydb:v1 ./db

5-2 DB 이미지 Push하기

```
docker push Docker허브계정/mydb:v1
```

<예시> --> 복사하지 마세요
docker push ekchoi3912/mydb:v1

5-3 flask 이미지 build하기

```
docker build -t Docker허브계정/myapp:v1 ./app
```

<예시> --> 복사하지 마세요
docker build -t ekchoi3912/myapp:v1 ./app

5-4 DB 이미지 Push하기

```
docker push Docker허브계정/myapp:v1
```

<예시> --> 복사하지 마세요
docker push ekchoi3912/myapp:v1

TeamProject2

[Docker 파일 구성]

DB Container 실행

6-1 DB Container 실행하기

```
docker run -d --name mydb Docker허브계정/mydb:v1

<예시> --> 복사하지 마세요
docker run -d --name mydb ekchoi3912/mydb:v1
```

6-2 DB Container 실행 상태 확인하기

```
docker ps

ekchoi391201@docker:~/work/docker$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
5f504091594f ekchoi3912/mydb:v1 "docker-entrypoint.s..." 12 minutes ago Up 12 minutes 3306/tcp, 33060/tcp mydb
```

6-3 DB Container IP확인하기

```
docker inspect mydb | grep '"IPAddress":' | awk -F '"' '{print $4}'

ekchoi391201@docker:~/work/docker$ docker inspect mydb | grep '"IPAddress":' | awk -F '"' '{print $4}'
172.17.0.2
172.17.0.2
```

예제에서는 172.17.0.2로 확인이 되었습니다.

App 컨테이너 실행

app 소스 상에서 DB Host를 db-svc로 지정했기 때문에 db-svc를 호출 했을 때 db Container의 IP가 응답할 수 있도록 해야 합니다.

```
app = Flask(__name__)
app.secret_key = 'your_secret_key'

# MySQL 설정
app.config['MYSQL_HOST'] = 'db-svc'
app.config['MYSQL_USER'] = 'frodo'
app.config['MYSQL_PASSWORD'] = 'frodo5020!!!'
app.config['MYSQL_DB'] = 'frodo'
```

```
docker run -d -p 5000:5000 \
--name myapp \
--add-host db-svc:DB컨테이너IP \
도커허브계정/myapp:v1
```

```
<예시> --> 복사하지 마세요
docker run -d -p 5000:5000 \
--name myapp \
--add-host db-svc:172.17.0.2 \
ekchoi3912/myapp:v1
```

7-2 DB Container 실행 상태 확인하기

```
docker ps

ekchoi391201@docker:~/work/docker$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
5f504091594f ekchoi3912/myapp:v1 "docker-entrypoint.s..." 9 minutes ago Up 9 minutes 0.0.0.0:5000->5000/tcp, :::5000->5000/tcp NAMES
myapp
```

7-3 app 접속하기

7-3-1 curl을 통한 간단한 확인

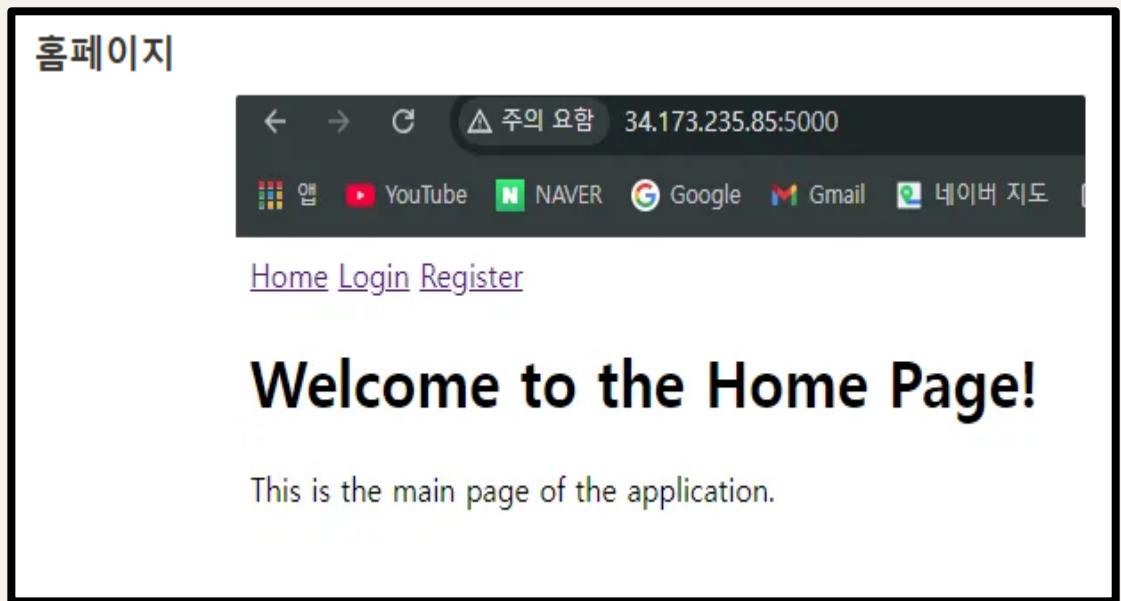
```
curl localhost:5000
```

```
ekchoi391201@docker:~/work/docker$ curl localhost:5000
<!DOCTYPE html>
<html>
<title>Redirecting...</title>
<h1>Redirecting...</h1>
<p>You should be redirected automatically to the target URL: <a href="/login">/login</a>. If not, click the link.</p>
ekchoi391201@docker:~/work/docker$
```

TeamProject2

[결과물]

홈페이지



로그인 페이지

로그인페이지

34.173.235.85:5000/login

Home Login Register

Login

Username

Password

데이터베이스

데이터베이스

```
mysql> select * from user;
+---+-----+-----+-----+
| id | username | email | password |
+---+-----+-----+-----+
| 1 | yigeonho | dlrjsgh0731@gmail.com | 123 |
| 2 | test | test@gmail.com | 123 |
+---+-----+-----+-----+
2 rows in set (0.00 sec)
```

TeamProject3

[Python shooting game]

OVERVIEW

- Python의 pygame 라이브러리를 이용하여 custom 슈팅 게임 만들기

PERIOD

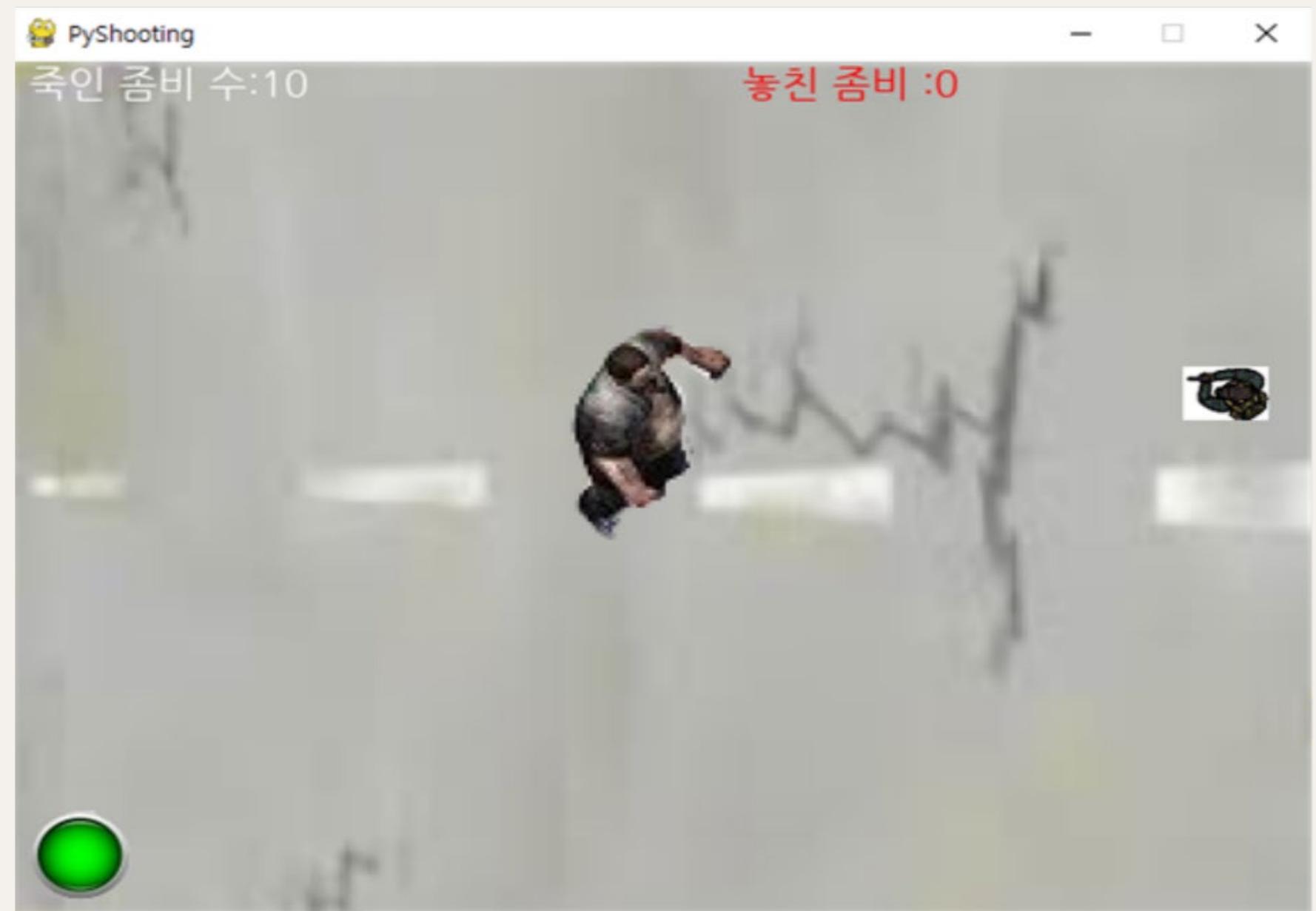
- 2024. 3. 13 ~ 2024. 3. 22 (약 10일)

TECHNOLOGY

- Environment : Visual Studio Code
- Language : Python
- Framework & Library : pygame

MY ROLE (팀장 PM)

- 프로젝트 총괄
- 가로형 게임 함수 제작
- 캐릭터 세로 축 이동 함수 제작
- 게임 난이도 조절을 위한 좀비 수 및 총알 수 조작 함수 제작
- 기획서 및 발표 ppt제작
- 발표



TeamProject4

[웹데이터 정보 및 실습]

OVERVIEW

- Html, CSS, Java Script를 이용하여 웹사이트 제작
- 데이터 수집 및 데이터를 활용할 게시판 제작
- GCP를 통해 DB 및 WEB 서버 제작
- Toad for SQL를 통해 DB 설계 및 연결
- Python Flask를 이용하여 WEB에 페이지 연결

PERIOD

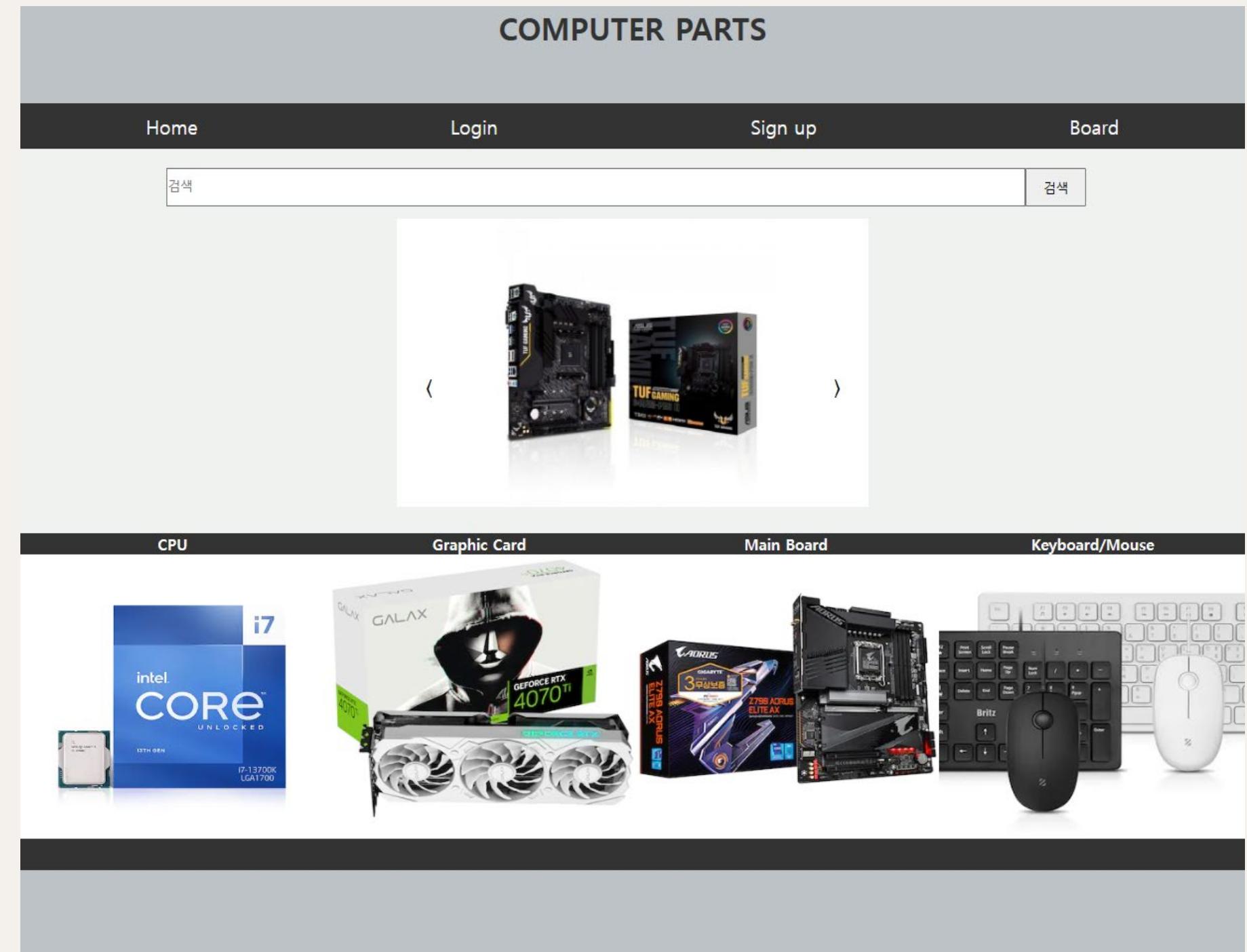
- 2023. 3. 13 ~ 2023. 6. 10 (약 90일)

TECHNOLOGY

- Environment : Visual Studio Code, GCP, Toad for SQL Server
- Language : Python, SQL, HTML, CSS, Java Script
- Framework & Library & Tools: Flask, MobaXterm

MY ROLE (팀장 PM)

- 프로젝트 총괄
- GCP를 통해 WEB, DB 서버 제작
- DB 설계, 제작
- 게시판 페이지 제작
- Flask를 통해 페이지를 웹 서비스에 연동
- 기획서 및 발표 ppt제작
- 발표



TeamProject4

[구성도]



- 메인 페이지 : Flask 를 통한 WEB 서버 연동 및 각각 버튼에 연결된 페이지들 연동
- 로그인 페이지 : 회원가입 가능, 아이디/비번 찾기 가능
- 제품 페이지 : 각 파츠 메인 배너를 통해 접속, 버튼을 통해 물건 담기 기능 추가, 버튼 활성화 시 페이지 하단의 텍스트바에 가격 측정
- 게시판 페이지 : DB 연동을 통해 글 쓰기, 글 수정, 글 삭제 기능 구현, POST 방식으로 데이터 서버에 이동, GET 방식으로 특정 데이터 불러오는 함수 제작

TeamProject4

[결과물]

메인 페이지 구성 및 Flask와 연동

COMPUTER PARTS

Home Login Sign up Board

검색 검색

CPU Graphic Card Main Board Keyboard/Mouse

intel CORE i7 GALAX GEFORCE RTX 4070 Ti GALAX RTX 4070 Ti TUF GAMING AORUS ELITE AX 3.0 RGB

제품 상세 페이지

CPU

인텔 코어i5-13세대 13400F (랩터레이크) (정품) 259,200원

인텔 코어i7-13세대_13700K 548,280원

AMD_라이젠5-4세대_5600X 181,660원

AMD_라이젠_7-5세대_7800X3D 635,530원

Flask를 통해 DB와 연결된 게시판

제품리뷰

번호	제목	글쓴이	작성일
1	[% for i in data_list: %] {{ i[0] }}{{ i[1] }}{{ i[2] }}{{ i[3] }} [% endfor %]		

<< < 1 2 3 4 5 > >>

등록

TeamProject5

[STM32 모듈 기반 통한 PID 제어 온도 센서 제작]

OVERVIEW

- 인베디드 시스템의 기초 및 C언어의 활용한 STM32 모듈 PID 제어 온도 센서 제작

PERIOD

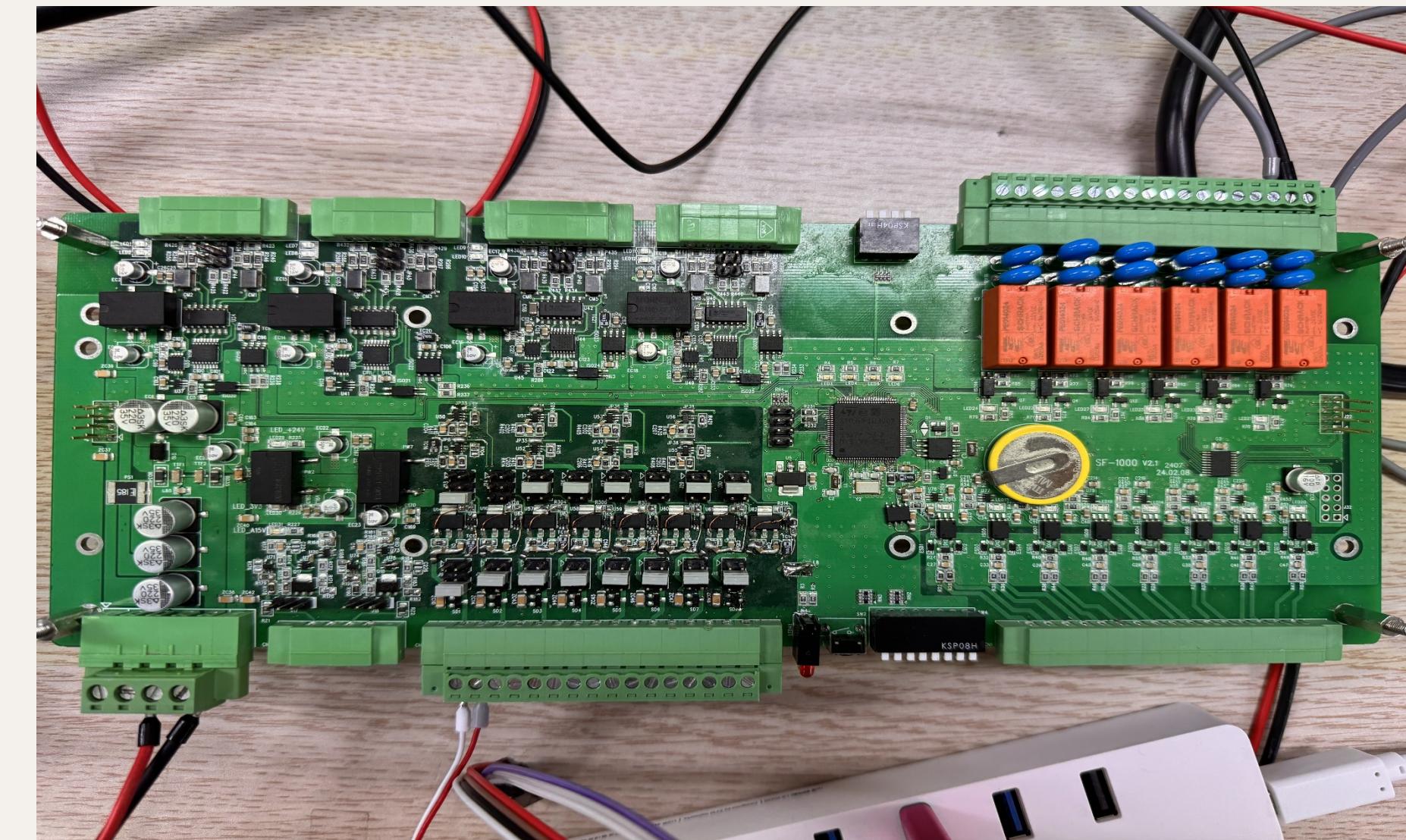
- 2024. 10. 14 ~ 2024. 12. 16 (약 64일)

TECHNOLOGY

- Environment : STM32 IDE,
- Language : C
- Tools & OS: Serial Packet, draw.io

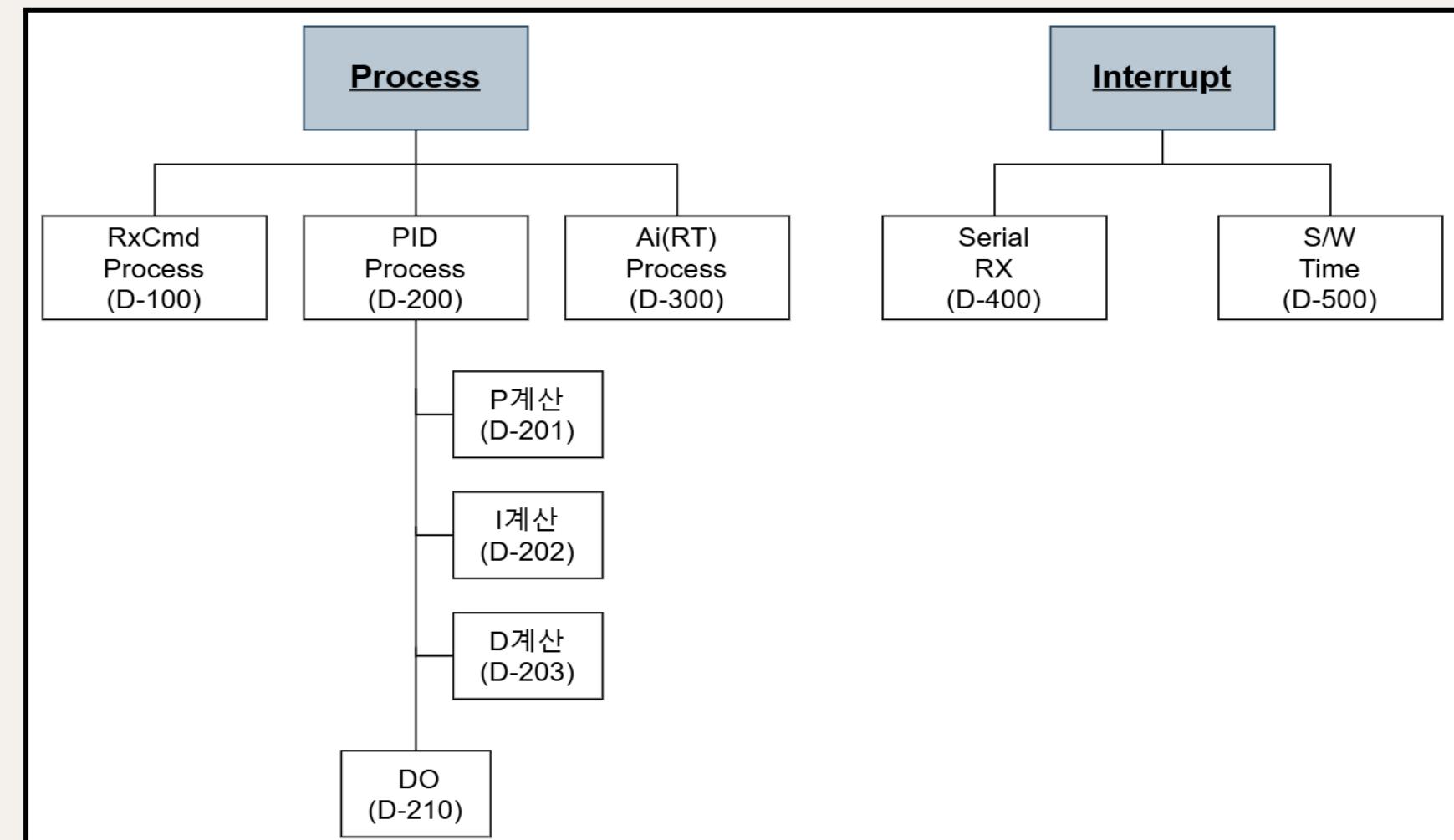
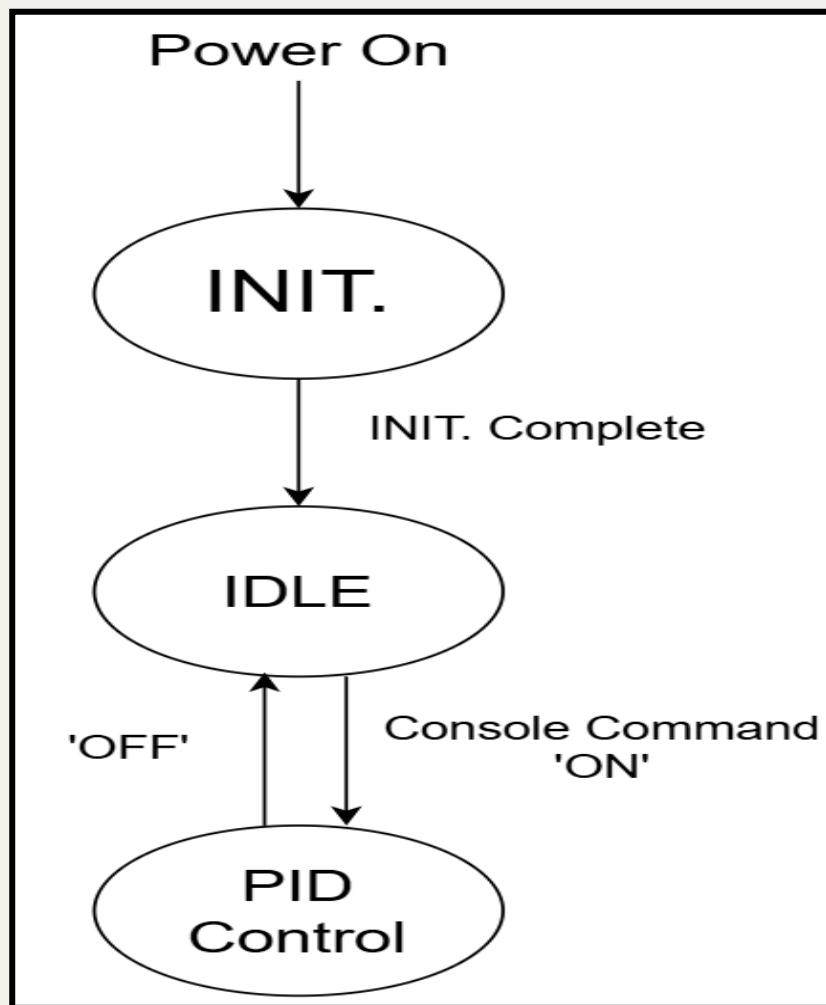
MY ROLE

- 프로젝트 설계
- 사양서 작성
- PID 제어 코드 제작



TeamProject5

[구성도]

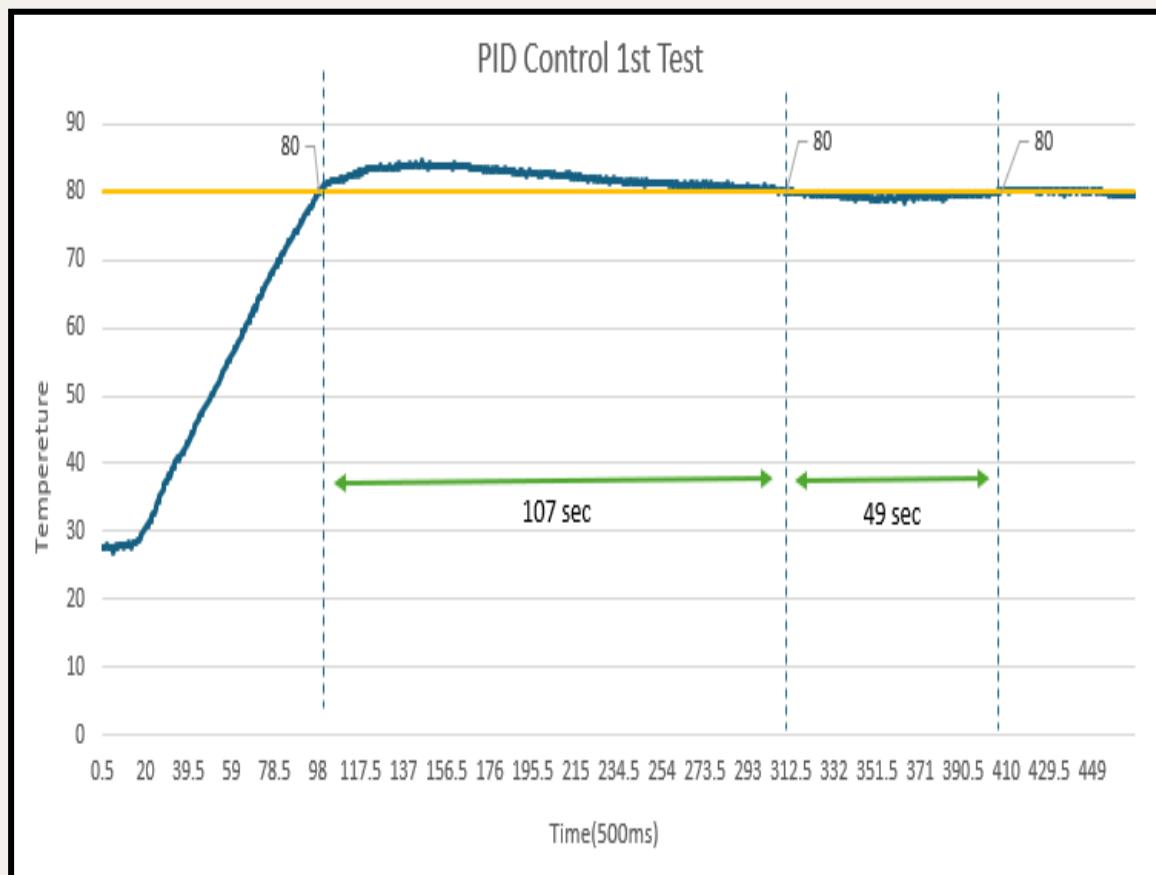


System Operation 시 파워작동으로 INIT. 상태로 돌입. INIT 완료 후 대기 상태에서 사용자의 'ON' Command 작동 시 PID 계산을 통한 온도 측정. 'OFF' 시 다시 대기 상태로 돌입. PID Control은 구체적으로 처음 P계산 이후 I 계산 이후 D 계산을 종합하여 온도의 유지와 직속 적인 상태를 파악해서 온도 측정. 후 Analog Input 처리로 계산 값 전송

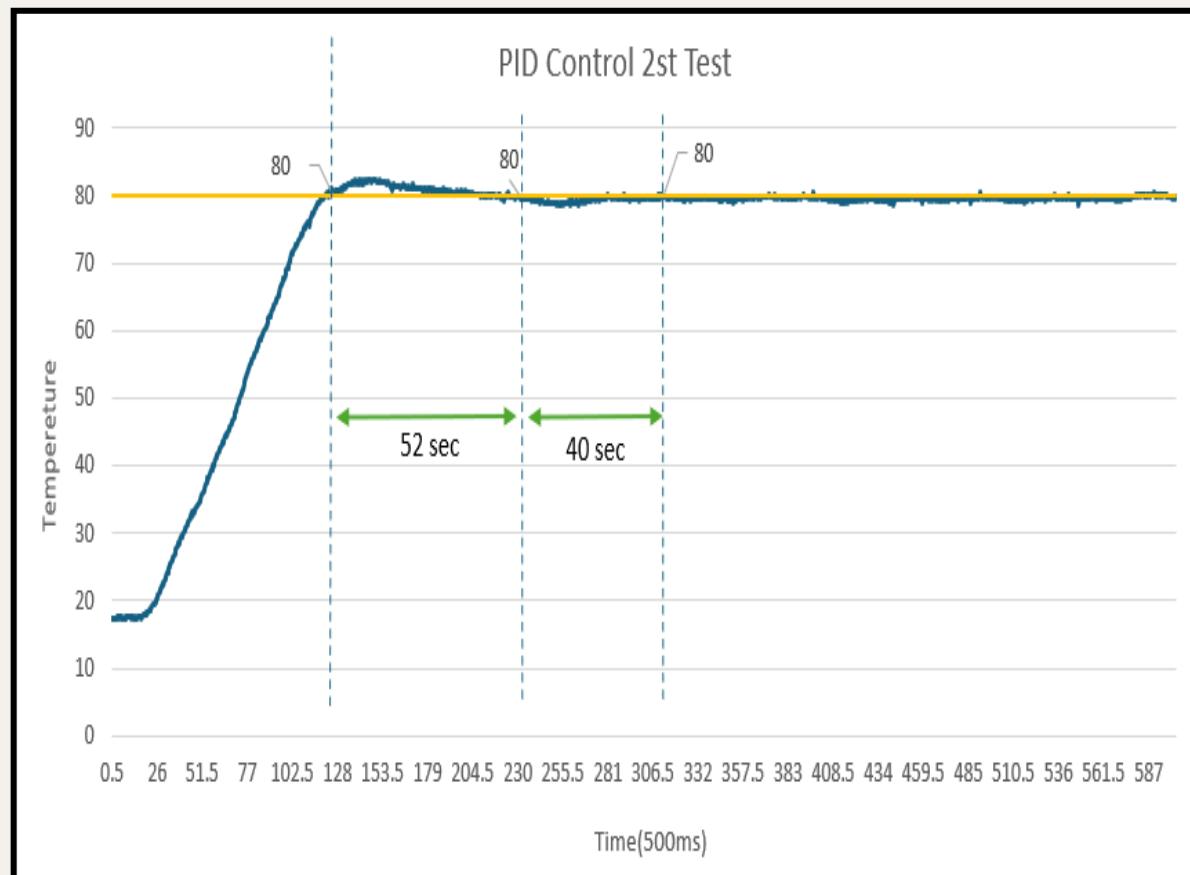
TeamProject5

[결과물]

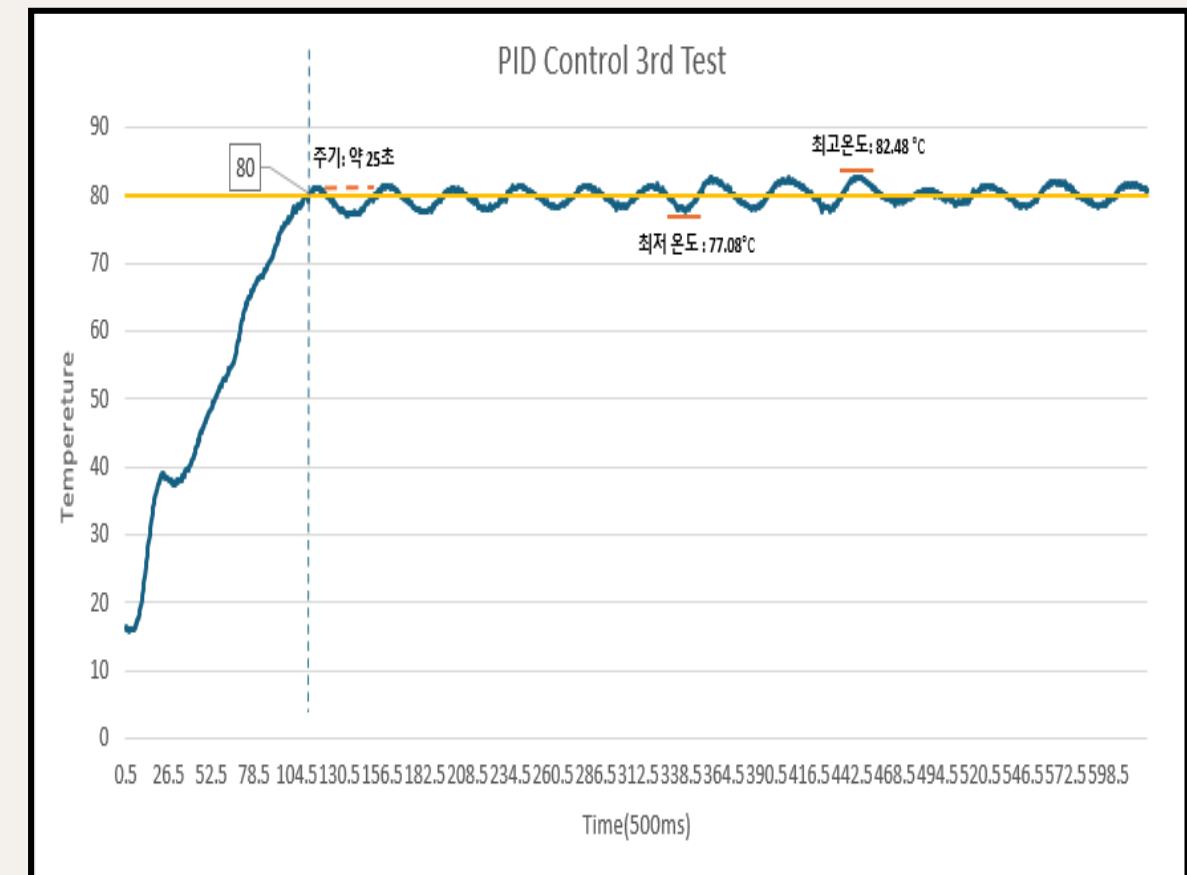
PID Control 1차 테스트 결과



PID Control 2차 테스트 결과



PID Control 3차 테스트 결과



TeamProject6

[CAN Protocol Monitoring Simulator 제작]

OVERVIEW

- 멀티 CAN 통신 시뮬레이터 제작. CAN 통신을 통한 데이터 송신, 수신, 시퀀스 송수신이 가능하고 통신 데이터 조작이 가능해야 함

PERIOD

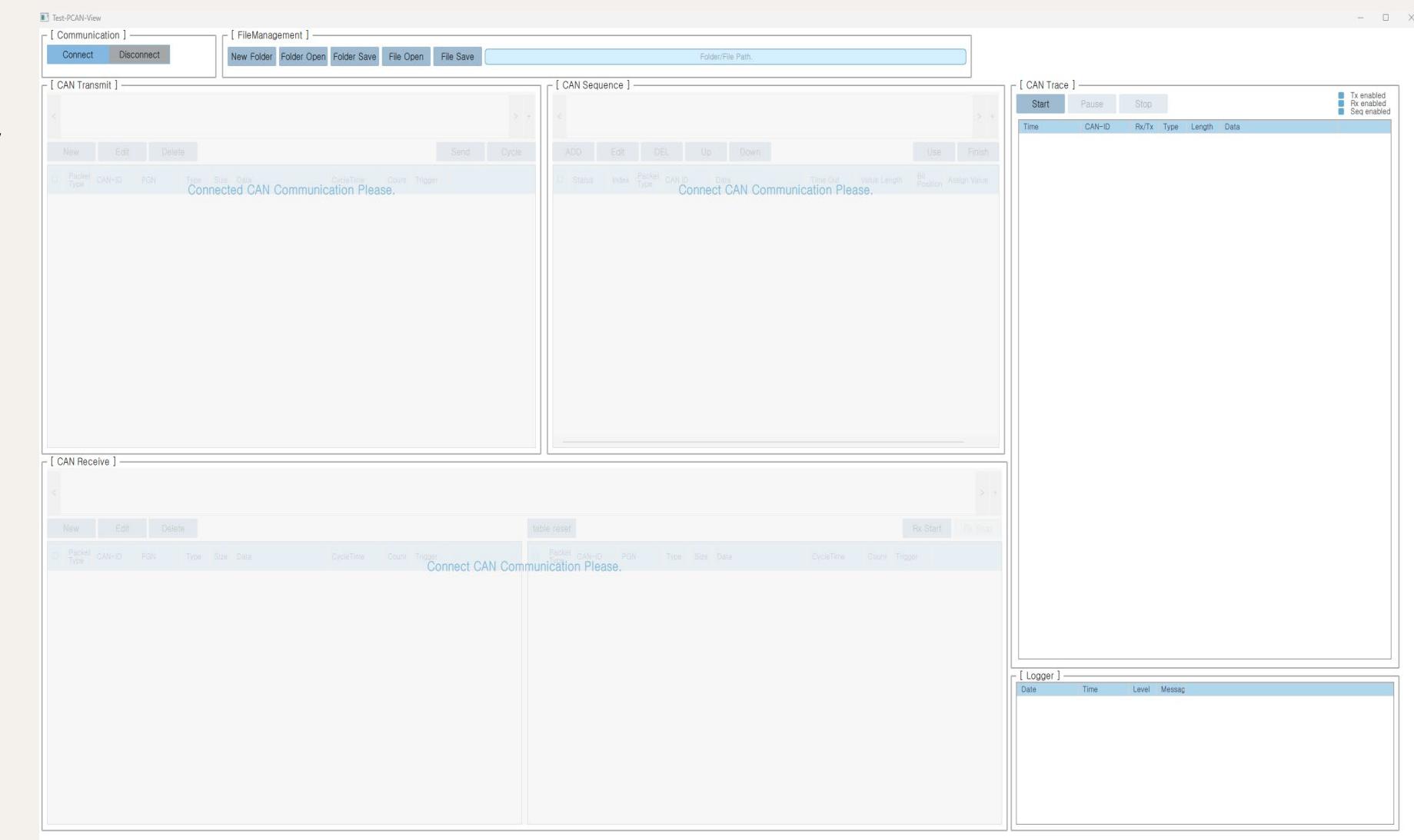
- 2025. 04. 02 ~ 2025. 08. 15 (약 4개월)

TECHNOLOGY

- Environment : Qt Creator
- Language : Qml, C++
- Tools & OS: Github, draw.io, PCAN_Basic API

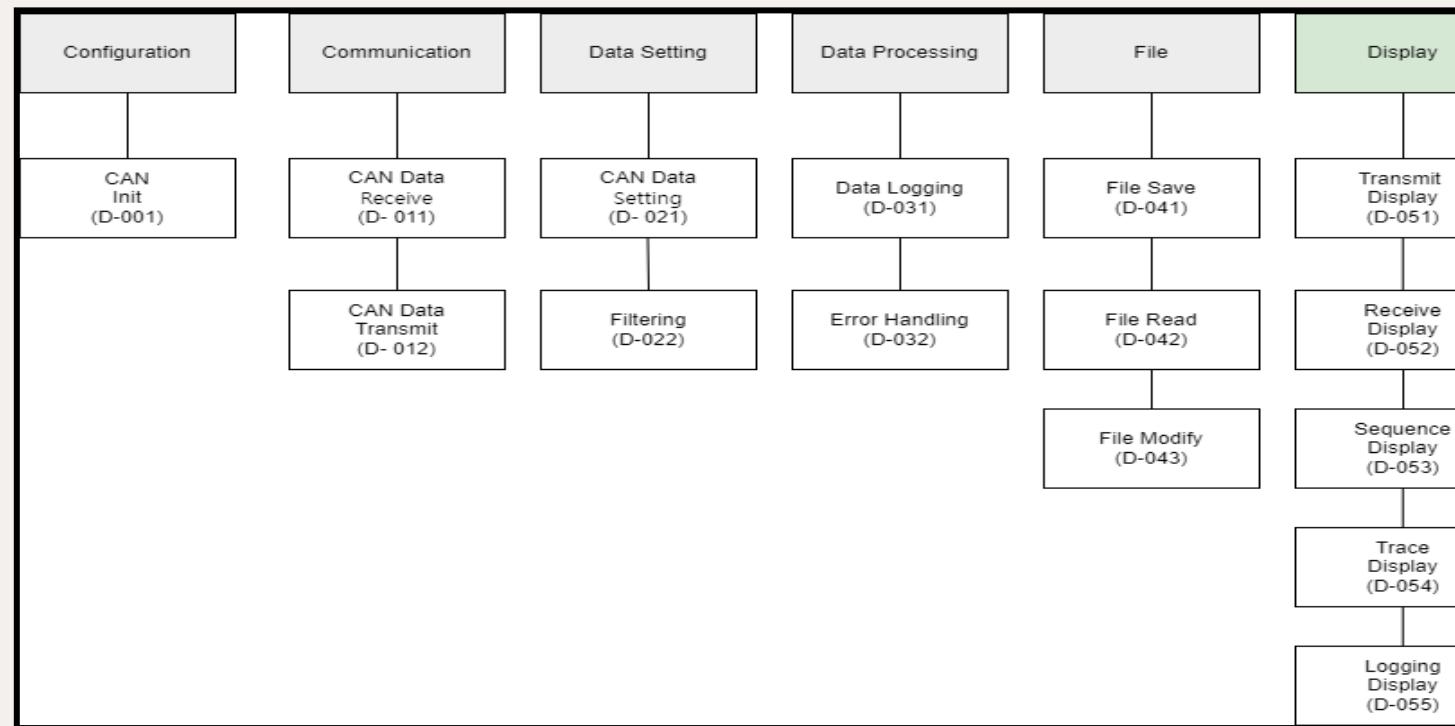
MY ROLE

- 프로젝트 설계
- 요구사항 설계서 제작
- 시퀀스 다이어그램 제작
- 시퀀스 파트 제작
- Trace 필터 제작

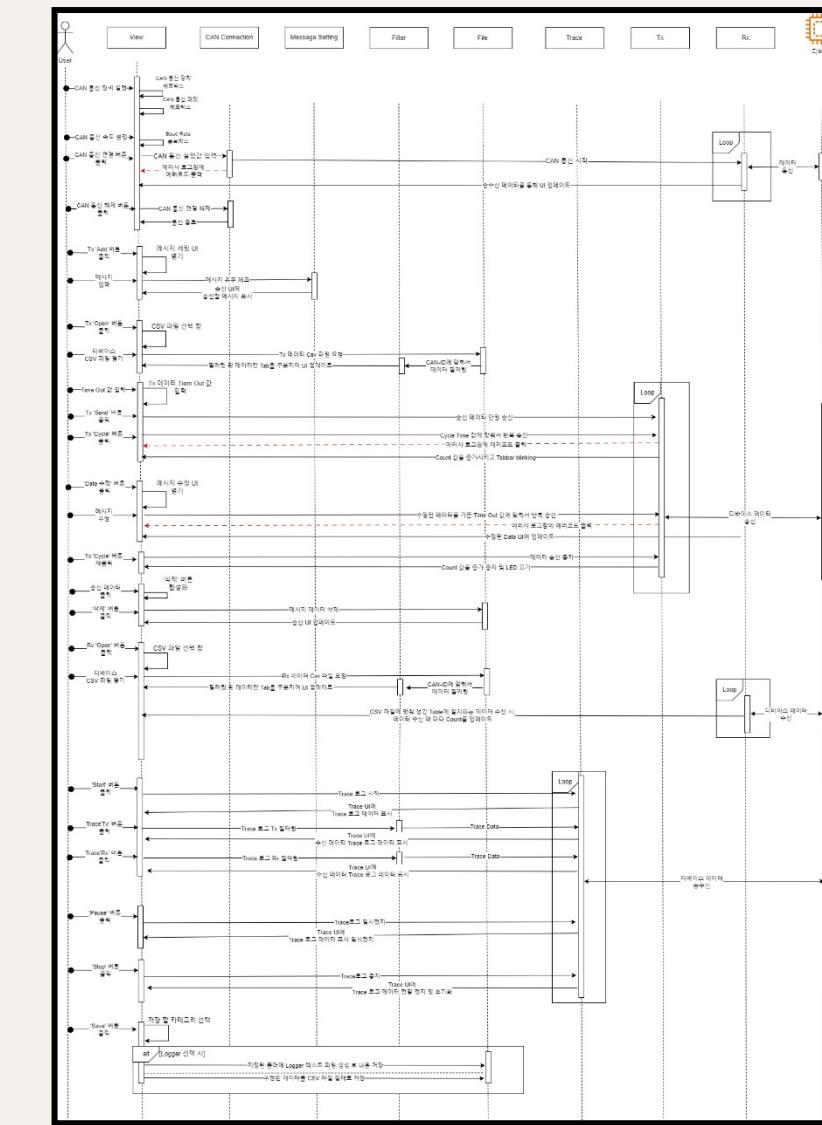


TeamProject6

[구성도]



[프로젝트 WBS]

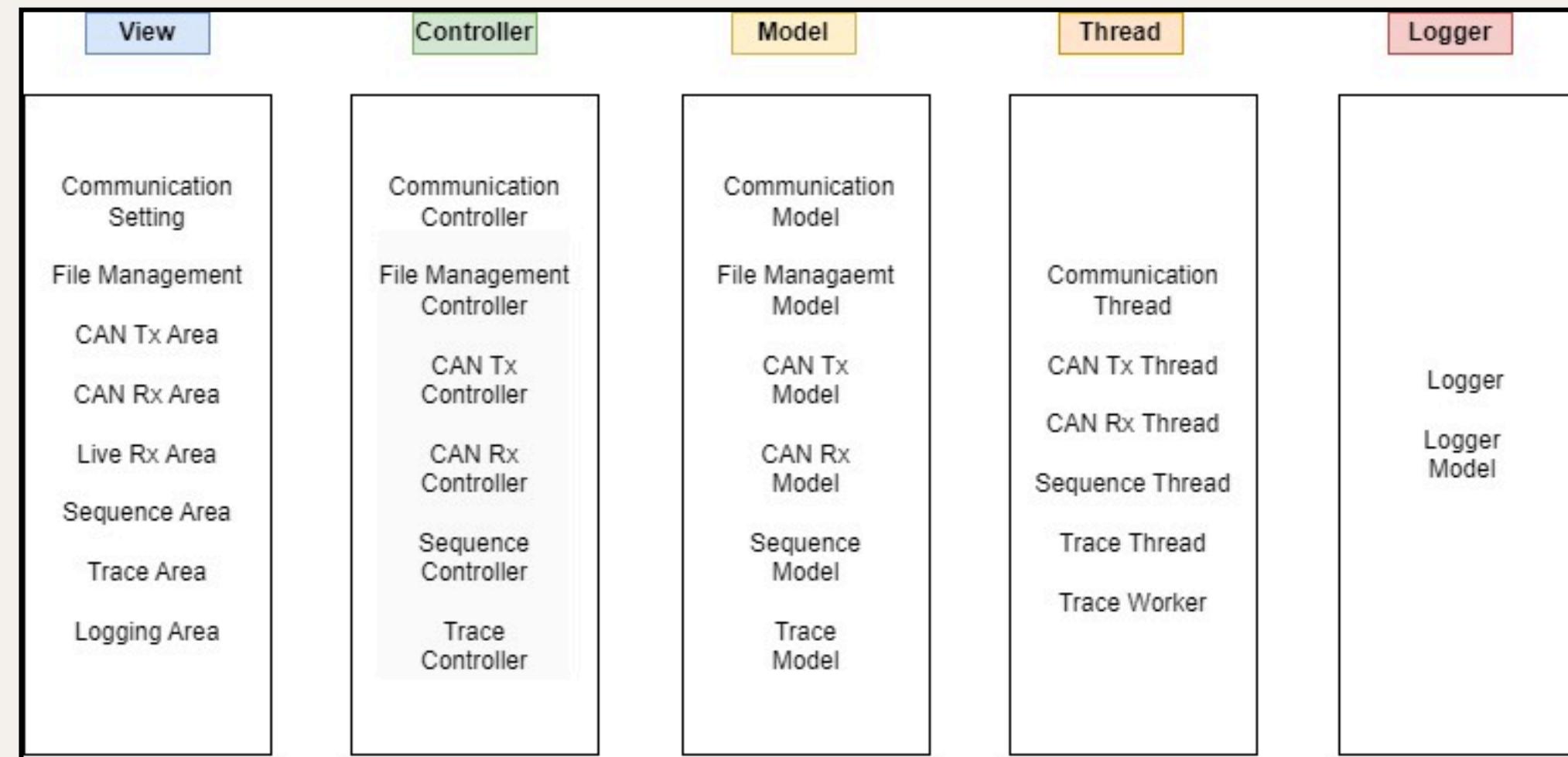


[시퀀스 Diagram]

QT를 통해 PCAN-Basic API를 활용하여 M 대 N CAN 통신이 가능해야 하고, 시퀀스를 통해 요구한 데이터를 송수신 하여 각각 연결된 디바이스에 따라 송신, 수신 데이터를 표기하는 할 수 있는 프로그램이다. 송수신 할 데이터를 CSV 및 Excel 형식으로 불러와 CAN 통신을 통한 데이터 전달이 가능해야 하며 Trace 부분에 실시간 데이터 송수신 정보가 표기된다.

TeamProject6

[구성도]



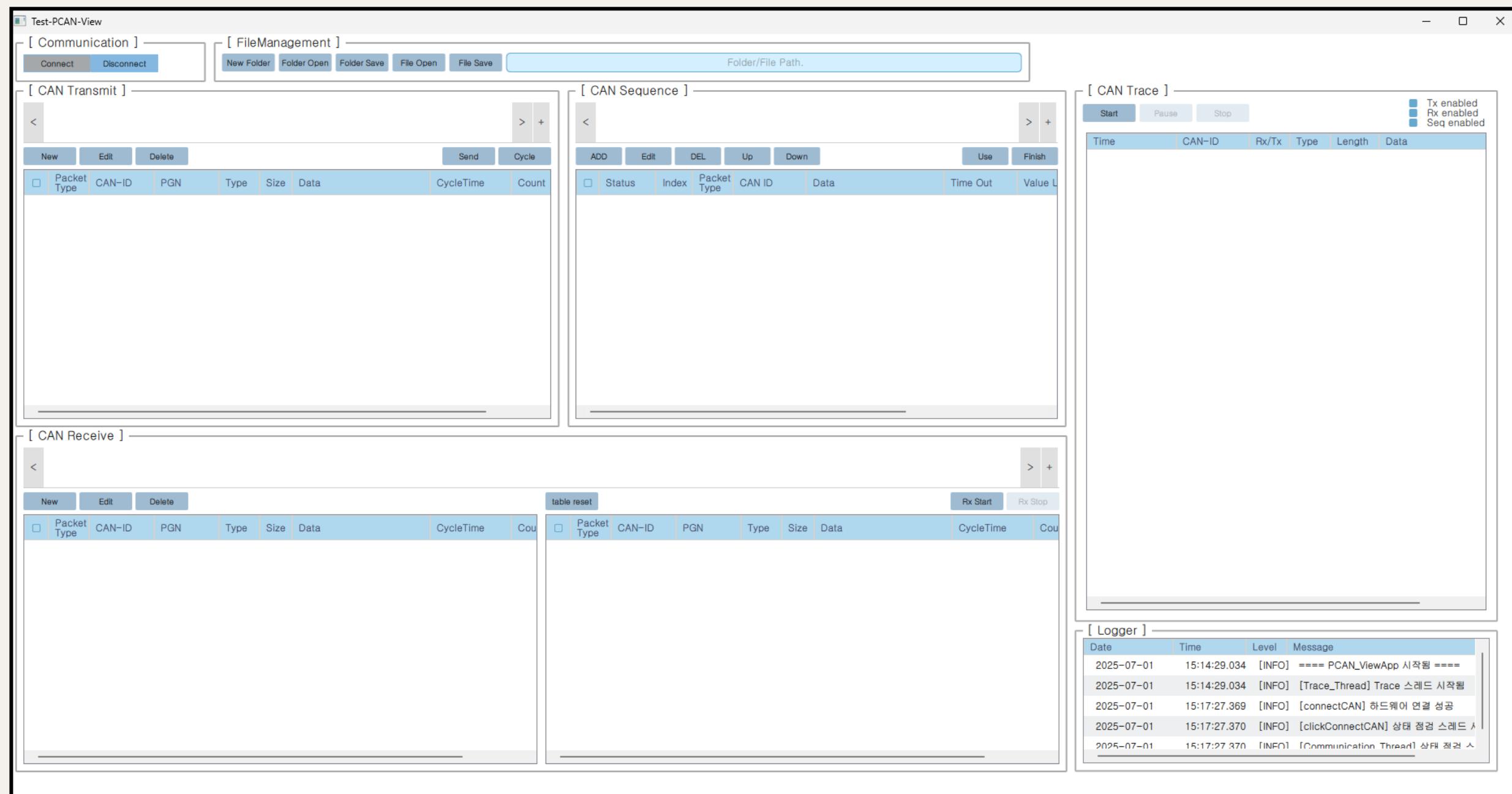
[System Architecture]

- 기본적으로 MVC 패턴은 활용한 프로젝트 빌드에, 시스템 적인 한계를 고려한 Thread 기술 추가를 통한 동시 시스템 처리가 가능한 패턴으로 구현하였다.
- 데이터 송수신 시 실시간으로 데이터의 처리를 확인 및 로그 확인을 위한 Logger 파트를 별도로 구분지어 유지보수에 적합한 형태로 구현하였다.

TeamProject6

[결과물]

CAN Protocol Monitoring Simulator 메인 환면



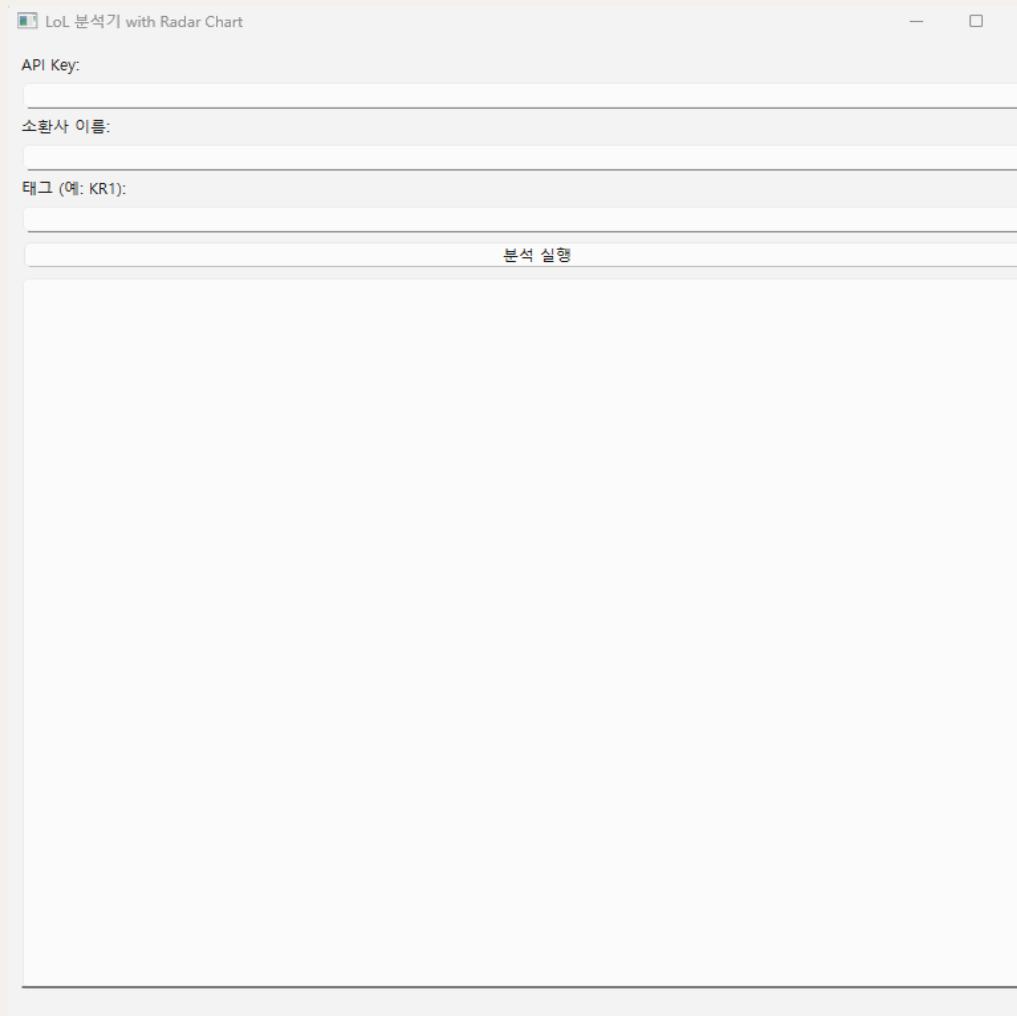
04. 개인프로젝트

1. League of Legends 전적 검색 + 피드백 프로그램 제작

제작기간 2025. 05. 25 - 2025. 06. 20

사용 기술

Visul Studio Code, Python, League of Legends API,
Chat GPT



2. 마인크래프트 멀티 경제 서버 제작 및 운영

제작기간 2025. 07. 12 - 2025. 08. 06

사용 기술

Skript, Minecraft Paper Server,
Chat GPT

```

[17:01:52 INFO]: [ModelEngine] [A] - Zipping Phase: 60,971,398.2ms
[17:01:52 INFO]: [Vault] Checking for Updates ...
[17:01:52 INFO]: [Vault] No new version available
[17:01:52 INFO]: [Skript] Loading variables...
[17:01:52 INFO]: [Skript] Loaded 498 variables in 0.0 seconds
[17:01:52 INFO]: [RoseGarden] An update for PlayerParticles (v8.11) is available! You are running v8.9.
[17:01:52 INFO]: [Quests] $Update to 5.2.8: $bhttps://github.com/PikaMug/Quests
[17:01:53 INFO]: [RoseGarden] An update for RoseLoot (v1.4.0) is available! You are running v1.3.0.
[17:01:53 INFO]: [Skript] All scripts loaded without errors.
[17:01:53 INFO]: [Skript] Loaded 8 scripts with a total of 15 structures in 0.5 seconds
[17:01:53 INFO]: [Skript] Finished loading.
[17:01:53 INFO]: [WealthTax] Scheduled tax collection is running...
[17:01:54 INFO]: [RoseLoot] Registered 195 loot table conditions.
[17:01:54 INFO]: [RoseLoot] Registered 11 loot table types.
[17:01:54 INFO]: [RoseLoot] Registered 19 loot item types.
[17:01:54 INFO]: [RoseLoot] Loaded 0 loot tables.
[17:01:54 INFO]: [Citizens] Loaded 8 NPCs.
[17:01:54 INFO]: [PlaceholderAPI] An update for PlaceholderAPI (v2.11.7) is available at:
[17:01:54 INFO]: [PlaceholderAPI] https://www.spigotmc.org/resources/placeholderapi.6245/
[17:01:54 INFO]: Done (28.059s)! For help, type "help"

```

Personal Project 1

[League of Legends 전적 검색 + 피드백 프로그램 제작]

OVERVIEW

- 라이엇 API를 활용한 전적 검색이 가능하고, 티어 별 분석, 주 포지션/부 포지션 별 분석, 티어 향상을 위한 팁 제공

PERIOD

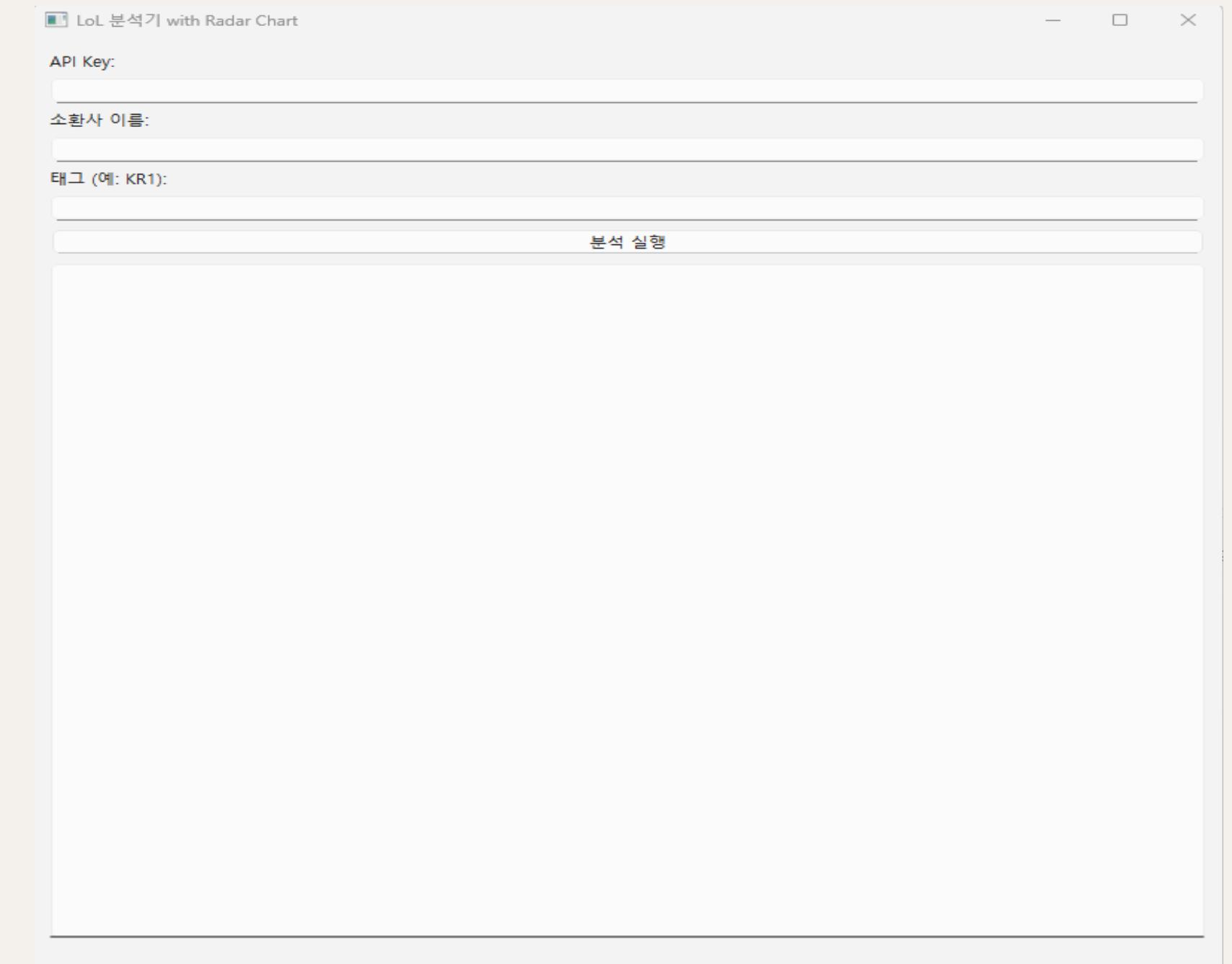
- 2025. 05. 25 - 2025. 06. 20 (약 28일)

TECHNOLOGY

- Environment : Visual Studio Code
- Language : Python
- Tools & OS: League Of Legend API, pychae

MY ROLE

- 프로젝트 설계
- Z-Score 구성
- UI 제작
- S/W 제작
- 데이터 시각화



Personal Project 1

[기획]

zscore_tier.json (Left): A JSON file containing Z-Score data for various game metrics across different tiers (TOP, JUNGLE, etc.). The data includes mean and standard deviation values for metrics like damage, deaths, and kills.

gui_main.py (Right): A Python script for a graphical user interface. It includes imports for Riot API and local files like 'constants.py' and 'match_filter.py'. It defines functions to get PUUID by Riot ID and Summoner info by PUUID, and to get match IDs and data by PUUID.

[티어 및 라인별 기준을 위한 Z-Score]

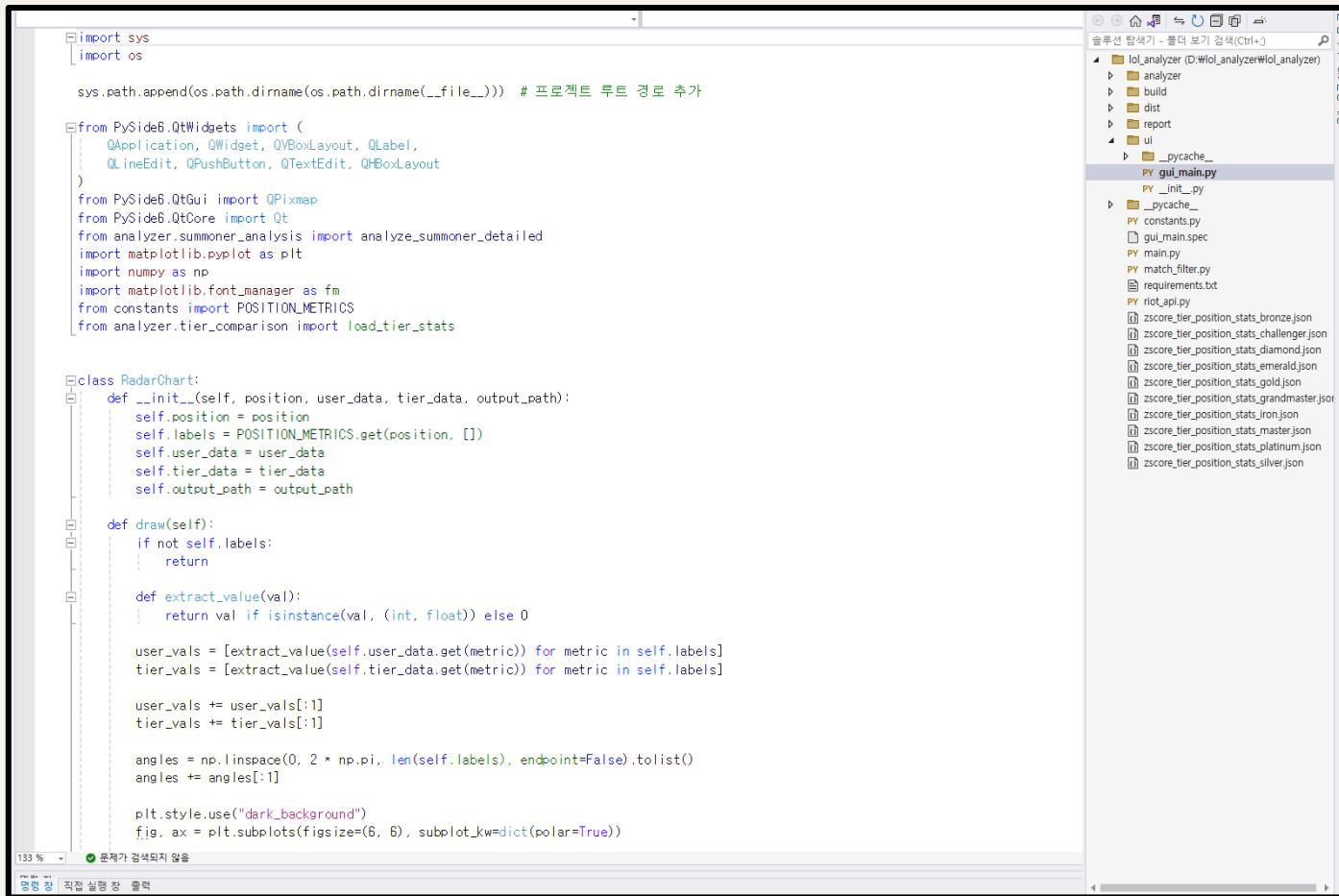
main.py (Right): A Python script for interacting with the Riot API. It includes functions to get PUUID by Riot ID, Summoner info by PUUID, match IDs by PUUID, and match data by match ID. It uses the requests library and defines URLs and headers for API calls.

[라이엇 API를 통해 전적 불러오기]

리그 오브 레전드라는 게임은 랭크 시스템을 도입하여 각 플레이어들마다 개인의 실력마다 랭크를 지정하고 각각 선호하는 역할군이 분할되어 주 역할군과 부 역할군이 나뉘어져 있는 게임입니다. 이 게임의 통계 및 개인 사용자의 실력을 비교하기 위해서 현존하는 모든 랭크 별 및 역할 군별 기준이 필요함으로 Z-Score 방식으로 기준을 잡고 사용자의 티어 및 역할군마다 장단점을 소개하고 다음 랭크로 가기 위해 간단한 팁을 제공하는 형식으로 구현했습니다.

Personal Project 1

[기획]



```

import sys
import os

sys.path.append(os.path.dirname(os.path.dirname(__file__))) # 프로젝트 투트 경로 추가

from PySide6.QtWidgets import (
    QApplication, QWidget, QVBoxLayout, QLabel,
    QLineEdit, QPushButton, QTextEdit, QHBoxLayout
)
from PySide6.QtGui import QPixmap
from PySide6.QtCore import Qt
from analyzer.summoner_analysis import analyze_summoner_detailed
import matplotlib.pyplot as plt
import numpy as np
import matplotlib.font_manager as fm
from constants import POSITION_METRICS
from analyzer.tier_comparison import load_tier_stats

class RadarChart:
    def __init__(self, position, user_data, tier_data, output_path):
        self.position = position
        self.labels = POSITION_METRICS.get(position, [])
        self.user_data = user_data
        self.tier_data = tier_data
        self.output_path = output_path

    def draw(self):
        if not self.labels:
            return

    def extract_value(val):
        if isinstance(val, (int, float)):
            return val
        else:
            return 0

    user_vals = [extract_value(self.user_data.get(metric)) for metric in self.labels]
    tier_vals = [extract_value(self.tier_data.get(metric)) for metric in self.labels]

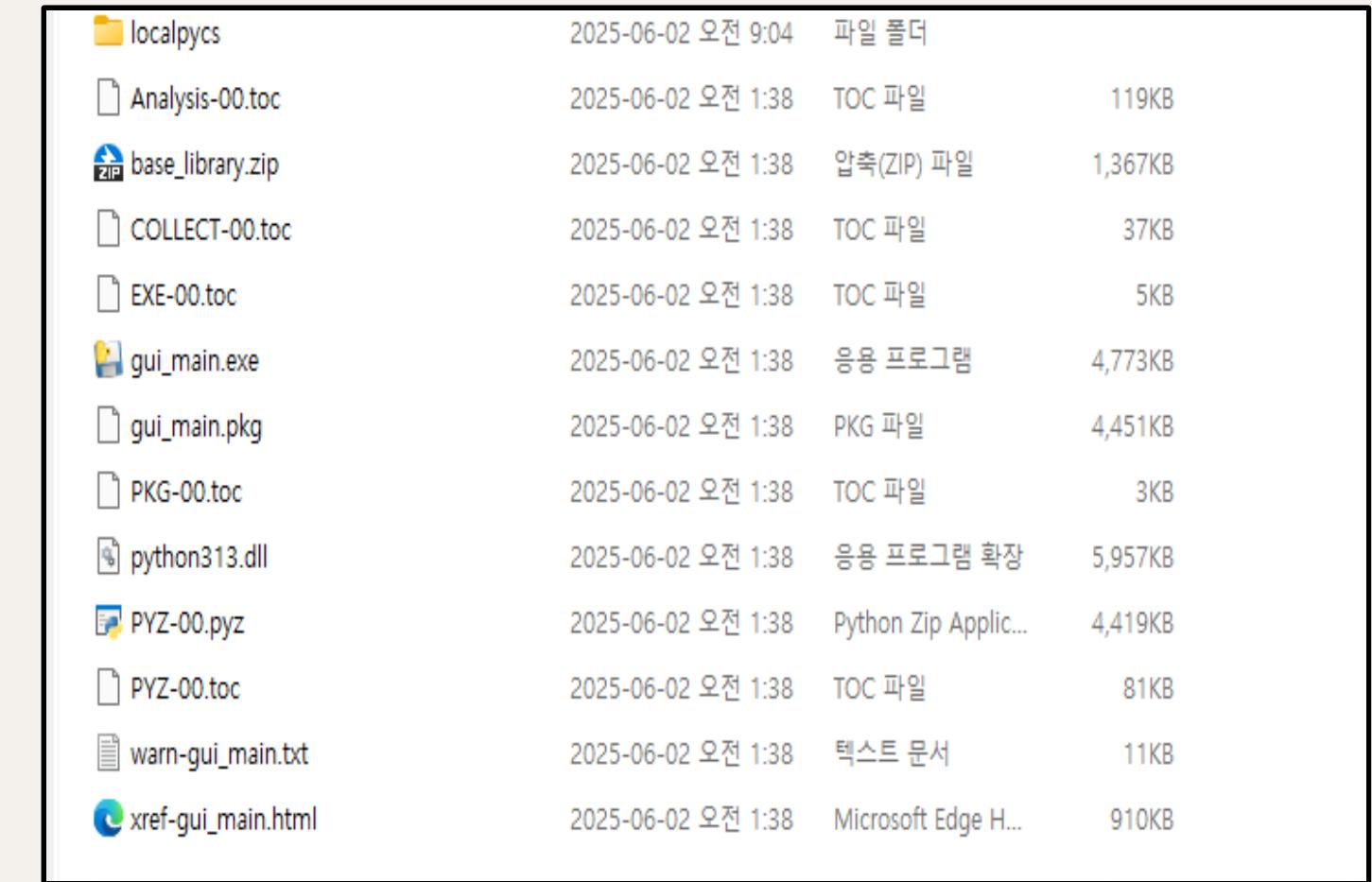
    user_vals += user_vals[-1]
    tier_vals += tier_vals[-1]

    angles = np.linspace(0, 2 * np.pi, len(self.labels), endpoint=False).tolist()
    angles += angles[-1]

    plt.style.use("dark_background")
    fig, ax = plt.subplots(figsize=(6, 6), subplot_kw=dict(polar=True))

```

[외부 서비스를 위한 GUI]



파일/폴더	날짜	타입	크기
localpycs	2025-06-02 오전 9:04	파일 폴더	
Analysis-00.toc	2025-06-02 오전 1:38	TOC 파일	119KB
base_library.zip	2025-06-02 오전 1:38	압축(ZIP) 파일	1,367KB
COLLECT-00.toc	2025-06-02 오전 1:38	TOC 파일	37KB
EXE-00.toc	2025-06-02 오전 1:38	TOC 파일	5KB
gui_main.exe	2025-06-02 오전 1:38	응용 프로그램	4,773KB
gui_main.pkg	2025-06-02 오전 1:38	PKG 파일	4,451KB
PKG-00.toc	2025-06-02 오전 1:38	TOC 파일	3KB
python313.dll	2025-06-02 오전 1:38	응용 프로그램 확장	5,957KB
PYZ-00.pyz	2025-06-02 오전 1:38	Python Zip Applic...	4,419KB
PYZ-00.toc	2025-06-02 오전 1:38	TOC 파일	81KB
warn-gui_main.txt	2025-06-02 오전 1:38	텍스트 문서	11KB
xref-gui_main.html	2025-06-02 오전 1:38	Microsoft Edge H...	910KB

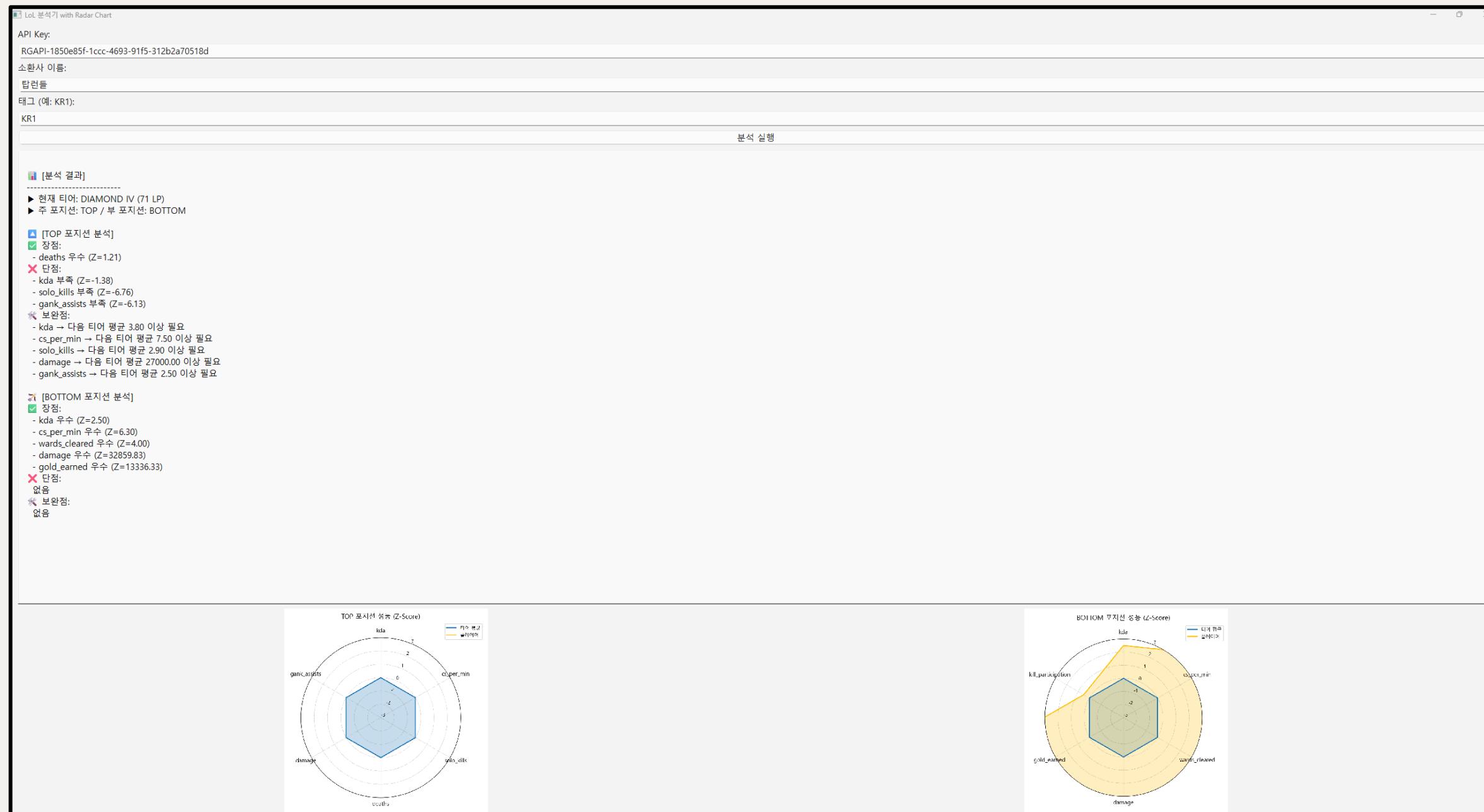
[외부 서비스를 위한 Build 파일]

추후 외부 배포를 위한 GUI 구현과 Build 파일 제작으로 실질적인 서비스 준비 및 서버에 배포시 온라인 서비스를 기획 할 수 있습니다.

Personal Project 1

[결과물]

LOL 분석기 with Radar Chart



Personal Project 2

[마인크래프트 멀티 경제 서버 제작 및 운영]

OVERVIEW

- 경제/도박/환율/퀘스트 시스템이 적용된 마인크래프트 멀티 서버 제작 및 운영

PERIOD

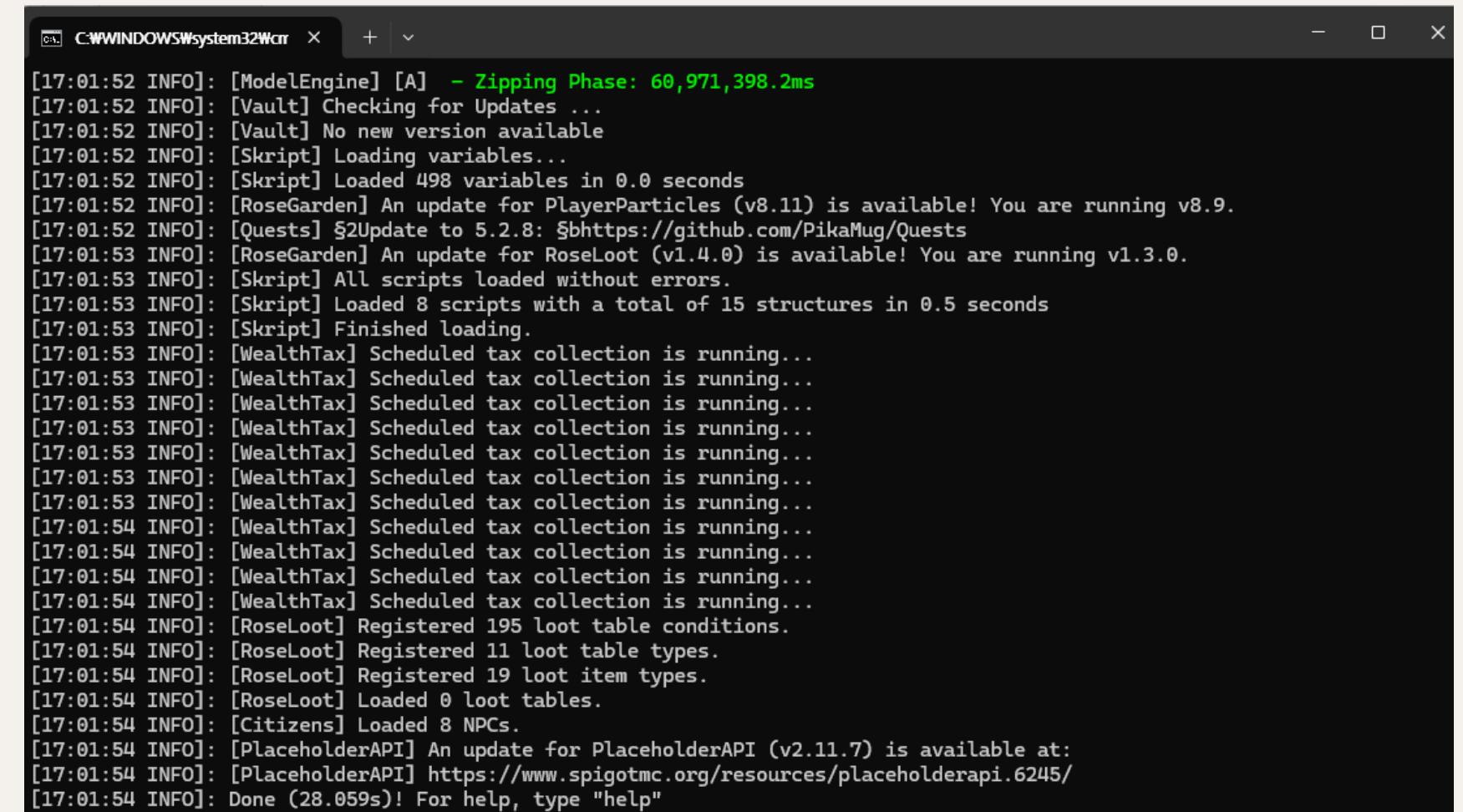
- 2025. 07. 12 - 2025. 08. 06 (약 26일)

TECHNOLOGY

- Environment : Minecraft Paper Server
- Language : Java, Skript
- Tools & OS: Linux, Skript, Plugins

MY ROLE

- 멀티 서버 구현
- IP 포트 포워딩 및 고정 IP 세팅
- Skript 문 제작
- 게임 시스템 제작
- 멀티 서버 운영



```

[17:01:52 INFO]: [ModelEngine] [A] - Zipping Phase: 60,971,398.2ms
[17:01:52 INFO]: [Vault] Checking for Updates ...
[17:01:52 INFO]: [Vault] No new version available
[17:01:52 INFO]: [Skript] Loading variables...
[17:01:52 INFO]: [Skript] Loaded 498 variables in 0.0 seconds
[17:01:52 INFO]: [RoseGarden] An update for PlayerParticles (v8.11) is available! You are running v8.9.
[17:01:52 INFO]: [Quests] §2Update to 5.2.8: §bhttps://github.com/PikaMug/Quests
[17:01:53 INFO]: [RoseGarden] An update for RoseLoot (v1.4.0) is available! You are running v1.3.0.
[17:01:53 INFO]: [Skript] All scripts loaded without errors.
[17:01:53 INFO]: [Skript] Loaded 8 scripts with a total of 15 structures in 0.5 seconds
[17:01:53 INFO]: [Skript] Finished loading.
[17:01:53 INFO]: [WealthTax] Scheduled tax collection is running...
[17:01:54 INFO]: [RoseLoot] Registered 195 loot table conditions.
[17:01:54 INFO]: [RoseLoot] Registered 11 loot table types.
[17:01:54 INFO]: [RoseLoot] Registered 19 loot item types.
[17:01:54 INFO]: [RoseLoot] Loaded 0 loot tables.
[17:01:54 INFO]: [Citizens] Loaded 8 NPCs.
[17:01:54 INFO]: [PlaceholderAPI] An update for PlaceholderAPI (v2.11.7) is available at:
[17:01:54 INFO]: [PlaceholderAPI] https://www.spigotmc.org/resources/placeholderapi.6245/
[17:01:54 INFO]: Done (28.059s)! For help, type "help"

```

Personal Project 2

[기획]

 BlockLocker.jar	2025-07-21 오후 9:10	Executable Jar File	184KB
 Citizens-2.0.37-b3738.jar	2025-07-20 오후 8:17	Executable Jar File	3,731KB
 CrazyCrates-4.9.8.jar	2025-06-25 오후 10:41	Executable Jar File	2,212KB
 DivinityEconomy-3.5.6.jar	2025-06-25 오후 10:43	Executable Jar File	1,511KB
 EconomyShopGUI-6.14.2.jar	2025-07-13 오후 4:35	Executable Jar File	1,551KB
 EssentialsX-2.21.2-dev+33-ff7c952.jar	2025-07-21 오후 9:13	Executable Jar File	4,619KB
 EssentialsXChat-2.21.2-dev+32-041762... jar	2025-07-20 오후 7:53	Executable Jar File	22KB
 InvSee++.jar	2025-07-25 오후 10:53	Executable Jar File	1,636KB
 InvSee++_Clear.jar	2025-07-25 오후 10:53	Executable Jar File	25KB
 InvSee++_Give.jar	2025-07-25 오후 10:53	Executable Jar File	167KB
 LuckPerms-Bukkit-5.5.8.jar	2025-06-21 오후 5:37	Executable Jar File	1,446KB
 ModelEngine-4.0.8.jar	2025-08-03 오전 3:49	Executable Jar File	2,265KB
 PlaceholderAPI-2.11.6.jar	2025-06-25 오후 10:43	Executable Jar File	912KB
 PlayerParticles-8.9.jar	2025-07-30 오후 10:50	Executable Jar File	1,850KB
 ProtocolLib.jar	2025-08-03 오전 3:11	Executable Jar File	10,011KB
 Quests-5.2.3.jar	2025-06-25 오후 10:39	Executable Jar File	4,537KB
 RoseLoot-1.3.0.jar	2025-06-19 오후 8:46	Executable Jar File	927KB
 ShopEditor-1.0.0-SNAPSHOT.jar	2025-07-23 오후 10:03	Executable Jar File	10KB
 Skript-2.11.2.jar	2025-06-19 오후 8:46	Executable Jar File	3,951KB
 skript-yaml-1.7.1.jar	2025-07-23 오후 10:42	Executable Jar File	578KB
 TAB v5.2.5.jar	2025-07-30 오후 10:49	Executable Jar File	3,434KB
 TaxSystem-1.0.2.jar	2025-08-09 오후 10:49	Executable Jar File	73KB
 TestProject250728-1.0-SNAPSHOT.jar	2025-08-08 오전 9:21	Executable Jar File	4KB
 Vault.jar	2025-06-19 오후 8:25	Executable Jar File	266KB
 worldedit-bukkit-7.3.16.jar	2025-08-02 오후 1:47	Executable Jar File	6,031KB
 worldguard-bukkit-7.0.13-dist.jar	2025-08-09 오후 11:17	Executable Jar File	1,139KB

[플러그인 제작 및 관리]

마인크래프트는 기본 사용자가 아이템을 만들며 블록으로 된 세상에 삶을 이어가는 게임입니다. 기본은 싱글 플레이어 게임이지만 친구들과 같이 하기 위해 서버를 운영하고 멀티 플레이 형식으로 운영을 하였고, 기존 게임에 더 큰 재미 요소를 첨가하고자 화폐 기능, 도박 기능, 환율 기능 등을 추가하고 게임의 목표를 만들고자 특정 아이템을 일정 기간 안에 모아서 점수를 획득 하는 형식으로 서버를 제작하고 운영하였습니다.

05. 자기소개

저는 두려움을 원동력으로 삼는 이종혁 입니다

프로젝트 리더의 경험

저의 참여했던 거의 모든 팀 프로젝트를 팀장으로써 참여를 하였습니다. 처음 팀장이란 지책을 맡았을 때, 엄청 부담감과 저의 아이디어로 프로젝트가 진행된다는 부담감이 그게 다가왔었습니다. 하지만 오히려 그러한 상황을 이겨내고 극복해보자 라는 마인드로 일단 부딪혀 봤습니다. 물론 순탄하진 않았고 여러 문제들을 직시 하였지만 오히려 저는 그 문제들을 해결해가면서 배운 게 많다고 생각합니다. 덕분에 팀원들 간의 소통이 얼마나 중요한지, 팀장이라는 지책으로써 어떻게 팀원들을 조율해가며 프로젝트를 성공적으로 완성시켜야 하는지, 그리고 누군가의 가르침 만큼 혼자 고민해가며 문제를 해결하기 위한 과정 또한 얼마나 중요한 경험이 되는지를 배울 수 있었습니다. 이러한 경험들을 현재 저의 능력이 되어 더 이상 부담이나 두려움이 아닌 남들이 쉽게 나서지 못할 때 저를 어필 할 수 있는 강점이 되었습니다.

핵심역량

-클라우드 서버

처음 클라우드 서버를 접하게 된 건 대학생활 중 교수님 추천으로 동기들과 산학연계 실무 프로젝트를 참여할 때 였습니다. AI 판례 챗봇을 만들게 되었고, 챗봇 운영을 위해 필요한 서버와 DB 파트를 맡게 되면서 독학으로 공부하던 중 GCP를 접하게 되었고, GCP를 활용하여 성공적으로 저의 파트를 완성시킬 수 있었습니다. 이때 느낌 성취감이 더더욱 클라우드 서버에 흥미를 불이게 해주었고 졸업 이후 추가적인 공부를 KG 에듀원 아이티뱅크에서 주관한 하이브리드 및 멀티 클라우드 아키텍트 양성 수업을 우수한 성적으로 수료했습니다.

-데이터 베이스

의지가 빈약했던 저는 그냥 대학교 전공이니 졸업만 하자 라는 생각으로 임했지만 군대를 전역하고 학교에 복학 한 이후, 제가 처한 현실에 불안감과 위기감을 느꼈습니다. 하지만 좌절을 하는 것이 아닌 오히려 더 의지를 불태워 전공부터 다시 천천히 확실하게 다져 나가면서 발전해 나가고자 마음을 먹었고, 그 덕분에 복학 이후에는 전공 수업을 다 A 이상의 성적을 내면서 졸업을 했으며 교수님 밑에서 추가적인 일을 하면서 원주 시청에서 주관한 설문조사 데이터 통계 및 정리하는 일까지 참여하며 데이터 활용에 대한 실무를 접했고 전공 교육의 실무 속 활용 방법과 중요도를 깨울 칠 수 있었습니다.

05. 자기소개

저는 두려움을 원동력으로 삼는 이종혁 입니다

핵심역량

-코딩

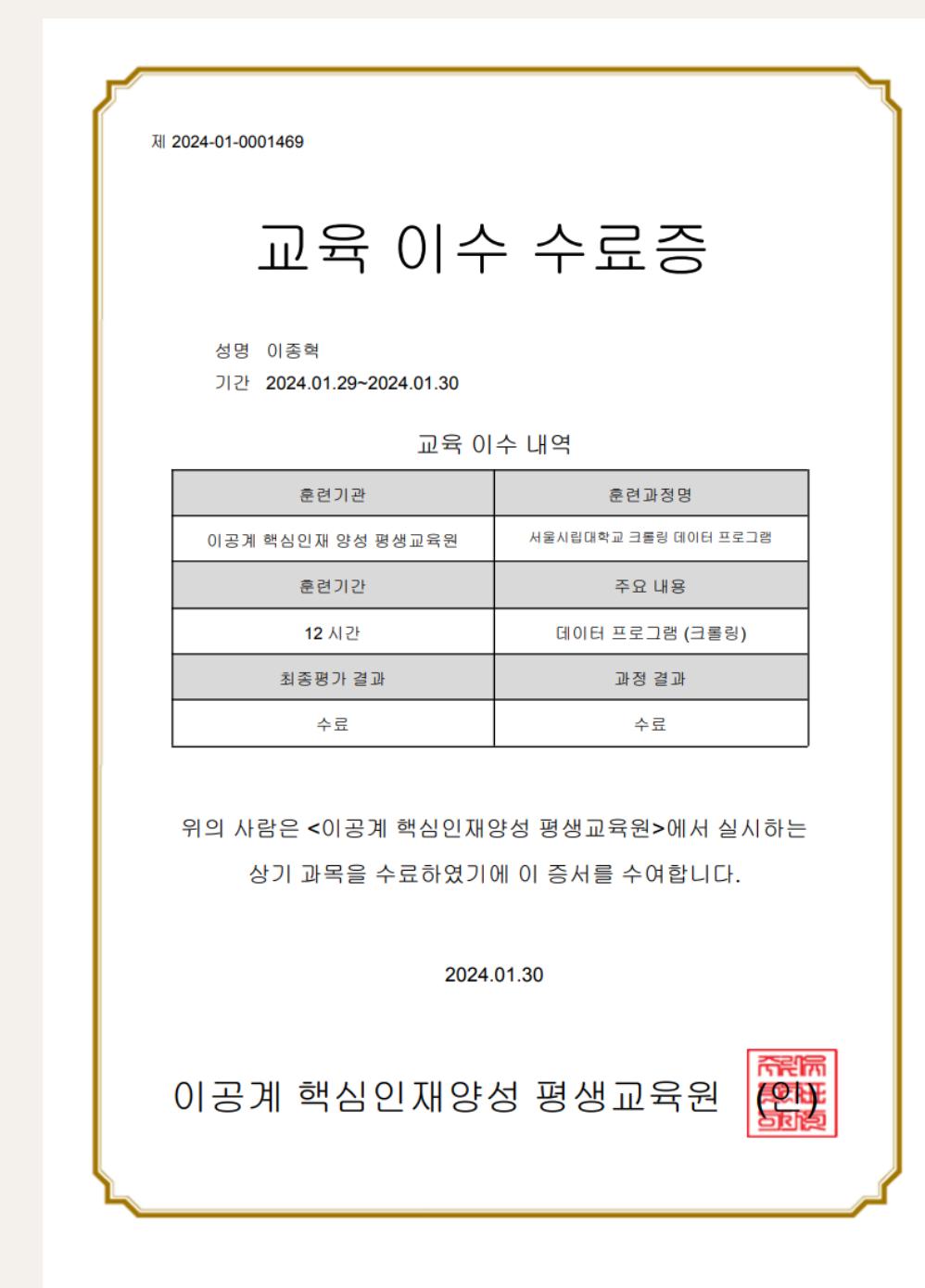
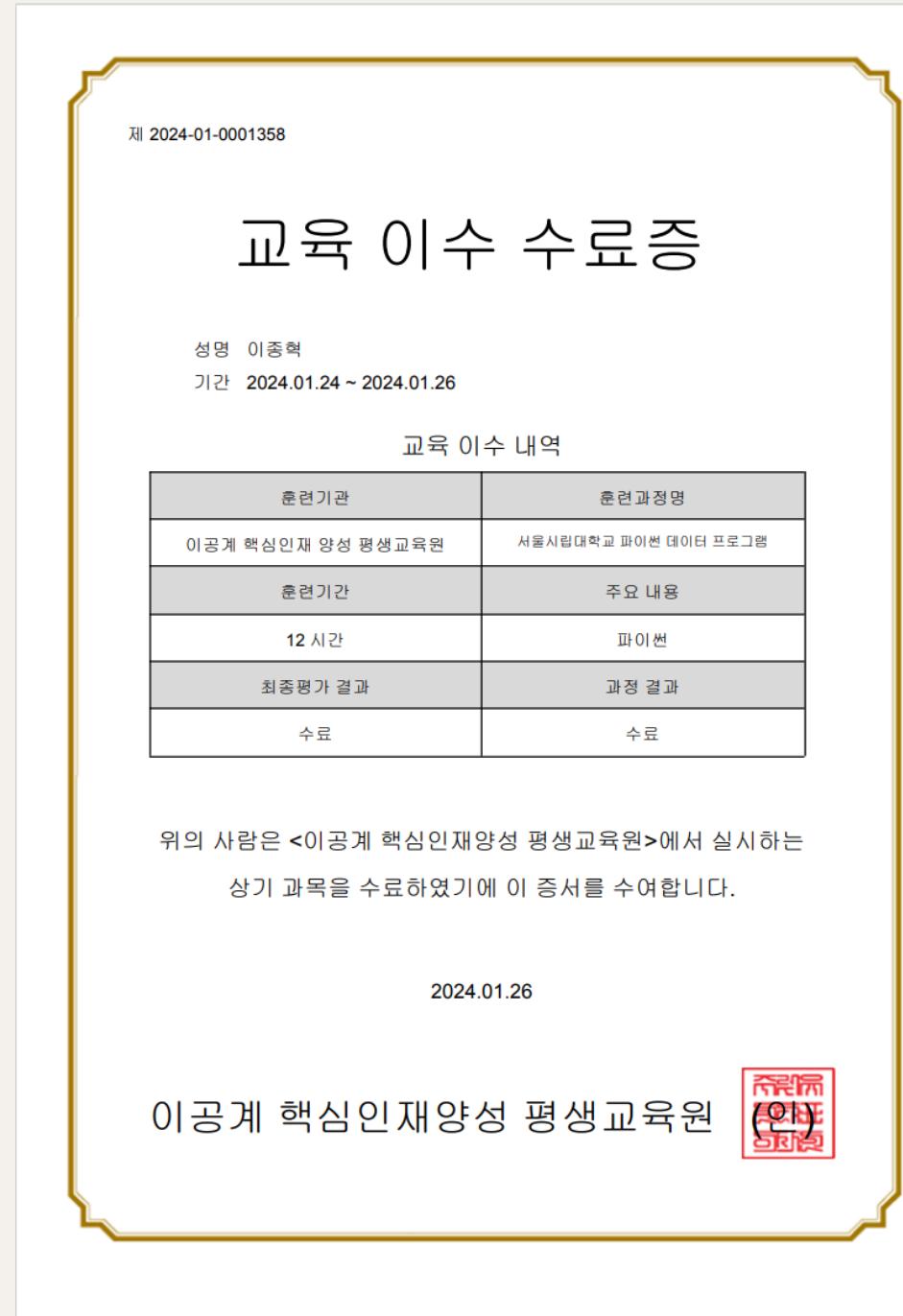
개발자 이신 아버지 밑에서 자라면서 자연스럽게 코딩을 접하게 되었습니다. 아버지는 제가 어렸을 때부터 컴퓨터에 흥미가 있으신 걸 아시고는 간단한 기초 코딩부터, 개발자의 삶이 어떤 건지 장점과 단점을 하나하나 알려주시면서 자연스레 개발자라는 직군이 저한테는 친근하고 가장 가까운 직업이 되었습니다. 그리고 처음으로 취업했던 회사에서는 리눅스 소프트웨어 엔지니어로써 여러 C언어 관련 교육과 실무 프로젝트를 통해 실무 경험을 쌓고 코디의 가능성을 몸소 느끼며 저의 관심사인 서버 혹은 게임과 같이 활용하면 얼마나 무궁무진한 가능성이 존재하는지 깨닳았습니다. 덕분에 저는 호기심이 생기면 일단 저의 생각과 주관대로 개발을 해보고 운영을 해보는 취미가 생겼습니다.

-영어

초등학교 1학년 부터 가족들의 판단으로 해외에 나가서 약 3년 6개월 이란 시간 동안 거주를 하며, 영국식 국제학교를 다녔습니다. 모국어를 제일 많이 습득할 시기인 만 7~9세에 저는 자연스레 2가지의 언어를 배울 수 있었습니다. 학교나 일상에서는 영어를 가족들 한테서는 한국어를 배웠습니다. 또한 저는 국제학교에서 많은 인종과 많은 나라의 사람들을 자연스레 만나고 친해지며 같이 생활을 했던 경험 덕분에 현재도 외국인은 만나고 상대하는 것에 일체 거부감이 없으며 오히려 남들보다 더 자신 있고 좋은 관계를 형성 할 수 있습니다.

06. 수료증

Python 활용 수료증



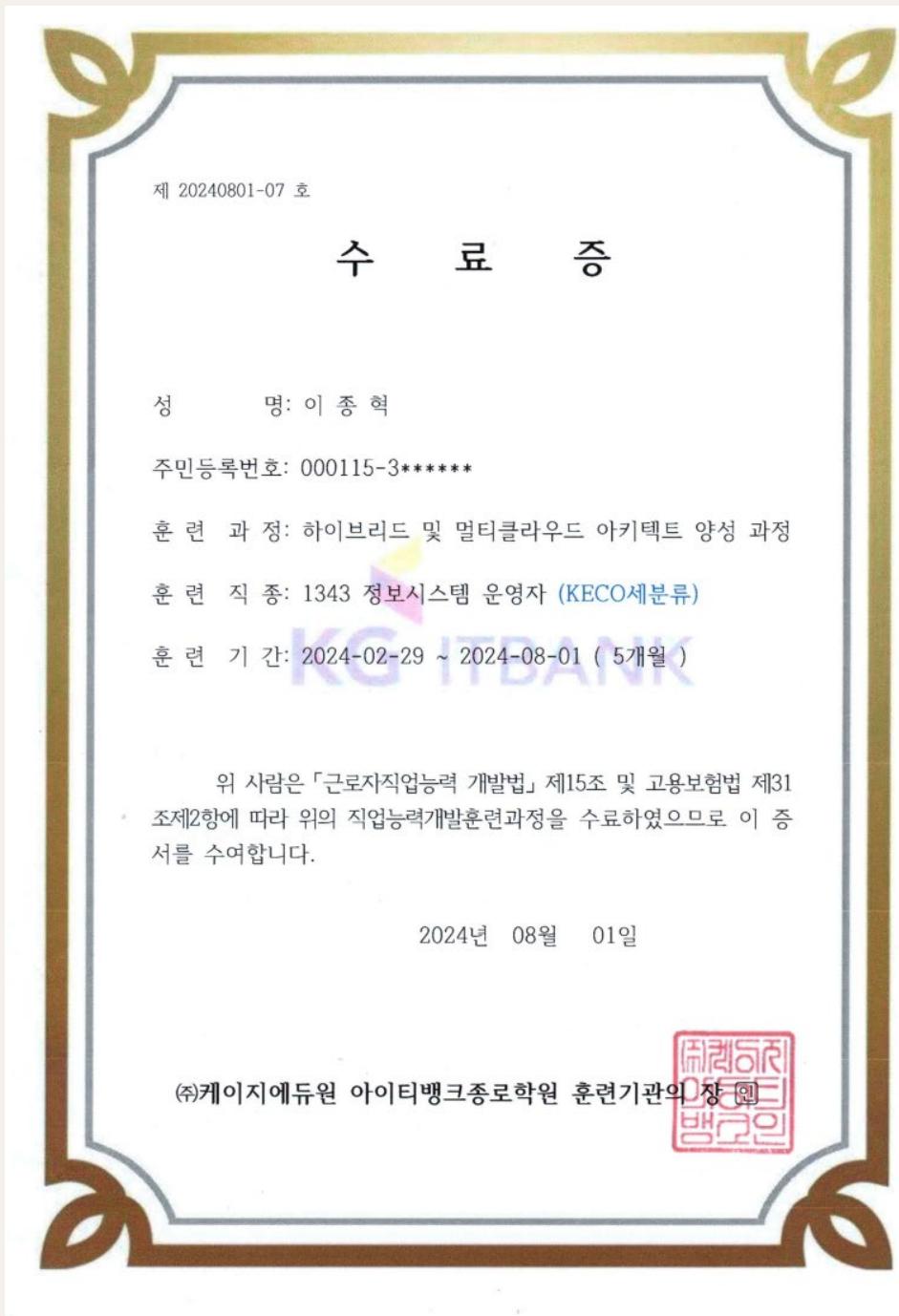
06. 수료증

멀티 클라우드 수료증



06. 수료증

멀티 클라우드 수료증



06. 수료증

OPIC 성적 확인서

시험 응시 확인서

시험구분	OPIC		
성명	이종혁	생년월일	2000.01.15
수험번호	2A0933132576	응시언어	영어
응시일시	2025-11-27 14:20	응시장소	강남시험센터 메이플넥스 역삼 C

위 수험자의 시험 응시를 확인합니다.

(주)멀티캠퍼스

발급일 : 2025.12.12

본 내용은 ACTFL 시험 주관사 (주)멀티캠퍼스를 통하여 정식으로 확인된 결과입니다.

응시 확인서를 위변조하는 행위는 시험관리규정과 관련 법률에 따라

응시자격제한 및 형사 처벌 대상이 될 수 있습니다.

고객센터 1644-0505

읽어주셔서 감사합니다.

Thank you.