

# SW 오픈소스 활용

## -Project Proposal-

Prepared by

강 형 철

32190103@dankook.ac.kr

(Dept. of Mobile System Engineering)

April 12, 2023

# 목차

## 1. 프로젝트 배경 및 목표

### 1-1. 프로젝트 기획 배경

### 1-2. 목표 및 기대효과

- 1) 프로젝트 목표 및 기대효과
- 2) 개인 목표

## 2. 요구사항

### 2-1. UI스케치 및 구성요소

- 1) Header
- 2) Body
- 3) Footer

### 2-2. 유저시나리오

## 3. 추진계획

### 3-1. 개발일정

### 3-2. 사용하는 오픈소스(혹은 라이브러리)

- 1) Django
- 2) Python & Python Open Library
- 3) Bootstrap
- 4) AWS(Amazon Web Service)

## 4. 위험 평가 및 대책

### 4-1. 위험요소 및 대책

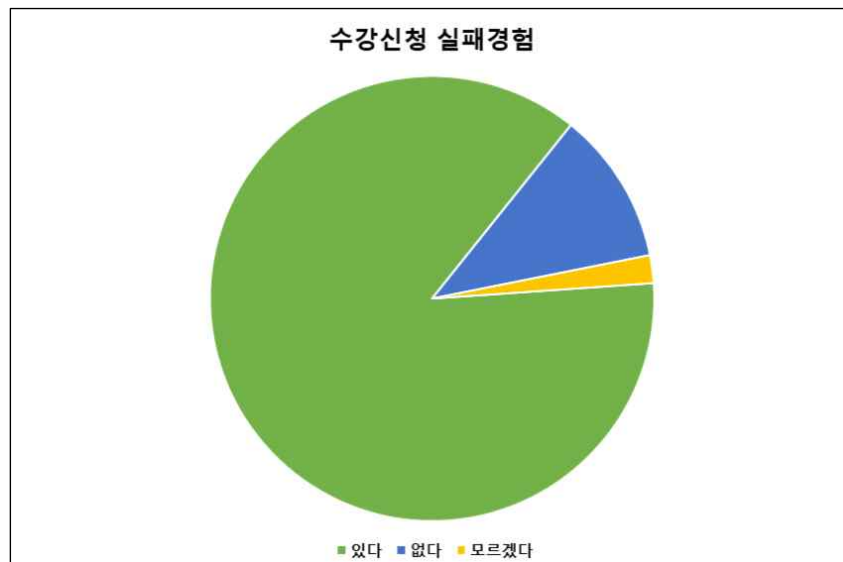
## 5. 주요 용어 설명

## 6. 참고문헌(citations)

# 1. 프로젝트 배경 및 목표

## 1-1. 프로젝트 기획 배경

이 프로젝트는 일단, 수강신청과 관련하여 제가 겪었던 일에서 출발합니다. 2022년 2학기과 2023년 1학기, 군대 전역 이후 복학하여 다닌 두 학기의 수강신청이 너무 힘들었습니다. 듣고 싶었던 강의를 인터넷이 느리거나 서버가 열리는 시간을 잘 확인하지 못하여 원하는 수업을 다 담지 못하였기 때문입니다.



<표> 수강신청 실패경험 설문결과

수강신청 실패경험이 있는지에 대한 설문조사를 진행해본 결과, 위 표와 같이 35명 중에 29명 가까이 수강신청을 실패해서 당황했던 적이 있다고 응답하였습니다. 수강신청과 관련한 문제는 비단 저만 겪었던 일이 아니라 주변 친구들이 모두 겪었던 일이었던 것입니다. 여기서 저는 대학생들이 겪는 수강신청 관련된 문제해결에 도움을 줄 수 있는 사이트를 만들어보려고 생각하였습니다. 물론, 개설강좌의 수강인원수 증가 등과 같은 ‘근본적인 문제’에 대해서는 제가 해결할 수 있는 문제가 아니지만, ‘인터넷 속도’, ‘서버 시간’ 등을 활용해 누구보다 먼저 수강신청을 진행하는 것으로 어느정도 극복이 가능한 문제이기 때문입니다.

우선 수강신청을 한다고 상황을 가정해 보겠습니다. 수강신청을 하기 위해 많은 사람들이 ‘수강신청 서버’에 접속하여 대량의 트래픽이 발생합니다. 이는 서버의 부하를 증가시켜 서버가 다운되거나 접속시간이 길어져, 수강신청 실패로 이어질 수 있습니다. 또한, 서버의 시간과 클라이언트의 시간에 오차가 있을 경우, **클라이언트** 기준 정확한 시간에 수강신청을 하더라도 실패할 수 있는 경우가 있습니다.

이러한 문제점들을 해결하는데 도움을 주기 위해, 이번 프로젝트에서는 Ping과 Delay

를 활용하여 서버 시간을 밀리세컨드 단위로 정확히 계산하여 서버 시간을 정확히 보여주는 사이트를 만들고자 합니다. 추가적으로 유저가 접속하고자 하는 사이트의 **현재 트래픽**을 측정하여 보조 지표로 활용할 수 있도록 해 보다 성공적인 수강신청이 될 수 있도록 도움을 주는 사이트를 만들고자 합니다.

추가적으로 이 프로젝트를 기획할 때 기존 수강신청을 진행할 때 접속해서 사용했던 <https://time.navyism.com/>에서 영감을 받았습니다. 하지만, 저희는 이 사이트에는 없는 여러 가지 추가기능(서버시간 밀리세컨드 단위 표시, 인터넷속도 랭킹, 핑속도 등)을 제공할 생각입니다.

## 1-2. 목표 및 기대효과

### 1) 프로젝트 목표 및 기대효과

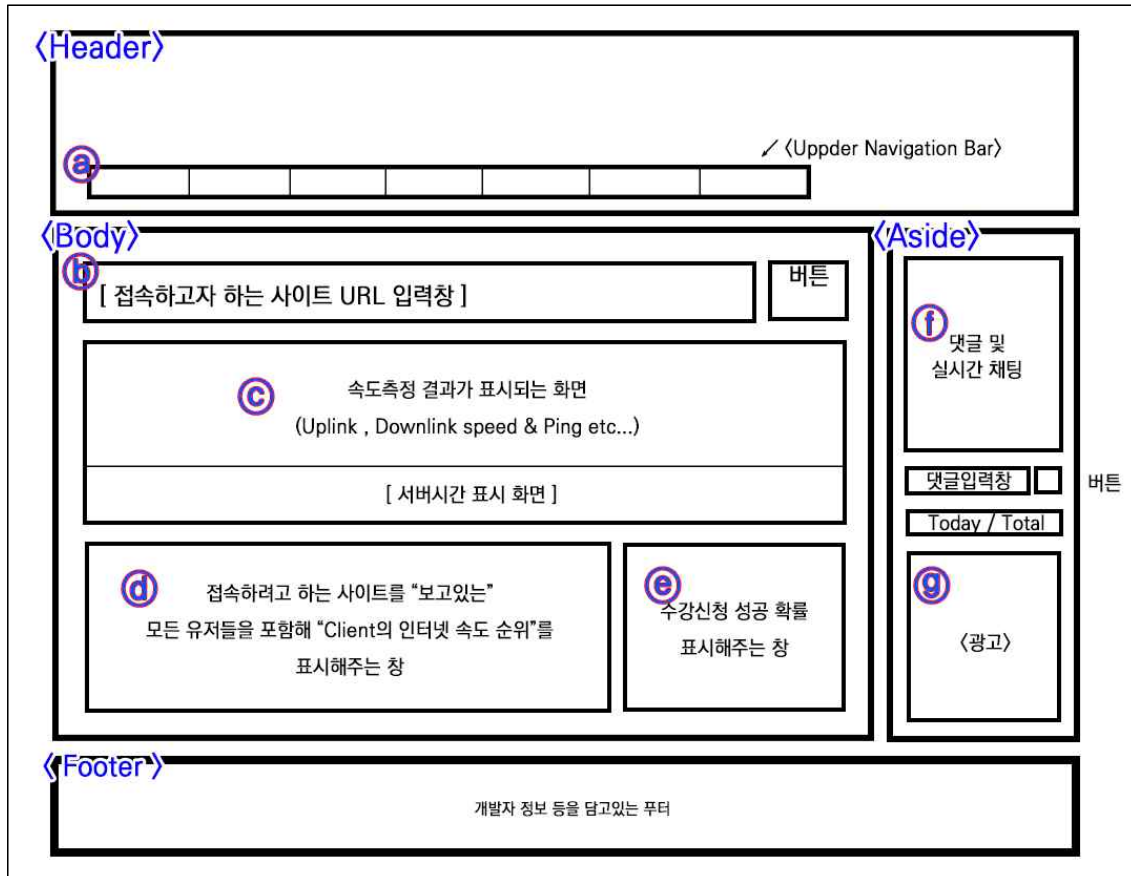
이 프로젝트의 궁극적인 목표는 정확한 서버시간 및 핑, 인터넷 속도, 트래픽을 계산하여 보여주는 사이트를 만들어 수강신청 더 나아가 티켓팅 등 ‘선착순’ 키워드가 필요한 홈페이지 어디서든 쓰일 수 있는 사이트를 만들어 배포하는 것입니다. 이후, 네이비즘(time.navism.com)사이트보다 더 좋은 성능으로 사람들에게 알려져 성공적으로 사이트가 정착된다면 광고 등을 수주받아 소정의 수익을 낼 수 있을 것 이라고 생각합니다.

### 2) 개인 목표

이번 프로젝트를 통한 개인적인 목표는, Django 웹프레임워크 숙달 및 프론트엔드 개발 기술 습득, 백엔드 개발 기술을 익히고 웹호스팅을 시도해 보는 것이 목표입니다. 또한, 부트스트랩 등 오픈소스 라이브러리를 적극 활용하여 자칫 밋밋해 보일 수 있는 웹 UI를 유저 친화적이게 만들어 보는 것입니다.

## 2. 요구사항

### 2-1. UI 스케치 & 구성요소 설명



<그림> UI Scatch

전반적인 UI스캐치는 위 그림과 같습니다. 물론, 개발상황에 따라 달라질 수도 있겠지만, 이번 프로젝트에서 제공하려고 하는 모든 기능을 다 담아보았습니다.

#### 1) Header

<Header> 파트에는 홈페이지 로고나, ③와 같이 [Navigation bar]등을 위치시켜 클라이언트가 이용하고, 저희가 제공할 주요 기능에 쿼메뉴를 제공합니다. 사이트 규모가 커지게 된다면 광고요소를 삽입할 공간이기도 합니다.

#### 2) Body

<Body> 파트에는 이번 프로젝트의 기능을 직접적으로 클라이언트에 제공할 메인파트입니다.

##### 2-1] ⑥ URL 입력창

유저가 접속하고자 하는 사이트(수강신청을 예로들면, <http://sugang.dankook.ac.kr>)

의 URL을 입력하는 곳입니다.

#### 2-2] ㉔ 인터넷 속도측정 표시화면 등

유저가 접속하고자 하는 사이트를 입력하면, 유저의 컴퓨터(Client)와 접속하고자 하는 서버의 인터넷 연결 상태 및 서버시간을 표시합니다. 이때, 핑이나 인터넷 상태를 점검하고 인터넷 딜레이를 계산해서 정확한 서버시간을 밀리세컨드(ms)단위로 표시합니다.

#### 2-3] ㉕ 인터넷 속도 순위

유저를 포함하여 현재 이 사이트를 통해 “접속하고자 하는 사이트”를 보고있는 유저들의 인터넷 속도 등을 계산하여 순위로 나타내 표시합니다.

#### 2-4] ㉖ 수강신청 성공확률 표시창

Ping, 인터넷속도 등 여러 지표를 활용해 사용자의 수강신청 성공확률을 계산한 결과를 표시합니다.

### 3) Aside

사이드바 메뉴들 와 작은 커뮤니티 및 광고배너가 들어올 공간입니다.

#### 3-1] ㉑ 채팅 및 댓글

사이트 접속자들끼리 대화할 수 있는 실시간 채팅기능을 제공할 예정입니다.

#### 3-2] ㉒ 광고배너

광고가 표시될 공간입니다.

### 4) Footer

푸터에는 많은 사이트에서 그러하듯, 개발자 정보 등을 남겨둘 생각입니다.

## 2-2. 사용자 시나리오

### 1) 클라이언트 사용방법(서비스 이용방법)

- ① 유저가 접속하고자 하는 사이트의 URL을 입력한다.
- ② 해당 사이트에 Ping을 전송해 Ping응답을 받아 인터넷 연결상태 및 Delay등을 계산한다.
- ③ 해당 사이트의 서버시간을 받아와, 핑 응답 시간 등을 고려해 ms단위로 표시합니다.
- ④ Uplink & Downlink 속도를 측정하여 인터넷 상태를 점검한다.
- ⑤ Ping, Delay, 인터넷속도 등을 측정한 결과를 토대로 수강신청 성공확률을 표시합니다.
- ⑥ 필요하다면 우측 메뉴바에 있는 채팅창을 통해, 가벼운 대화를 나눌 수 있습니다.
- ⑦ 수강신청 시작 시간 등, 유저가 설정해두면 알림을 받을 수도 있습니다. (추가예정)

### 3. 추진계획

#### 3-1. 개발일정



<그림> 전반적인 개발계획

##### 1) 1차 개발계획 [~4월30일 까지]

UI구성도 및 Html틀을 작성합니다. 이후, 프로그램 아키텍처를 설계하고 제공할 기능들을 모듈화 하여 알고리즘을 설계합니다.

##### 2) 2차 개발계획 [~5월31일 까지]

① 알고리즘 설계도 및 프로그램 아키텍처를 토대로 기능을 구현합니다. 이때, MVC모델을 제공해주는 Django를 통해 웹사이트를 개발합니다. 또한, HTML틀 및 부트스트랩을 활용하여 사용자에게 직접 보여질 부분을 작성합니다.

② MVC모델과 모듈을 결합합니다.

③ HTML Template을 적용하여 최종 완성합니다.

##### 3) 3차 개발계획 [~학기말 까지]

① 최종 배포 및 발표 전까지 테스트를 가해보며 버그를 발견 & 수정합니다.

② 6월 첫째주 주말에 Amazon Web Service를 통해 배포합니다.

#### 3-2. 사용하는 오픈소스(혹은 라이브러리)

이번 프로젝트에서 주로 사용하는 오픈소스는 크게 다섯가지가 있습니다.

[Django, Python, AWS, Boot-strap, Python library]

##### 1) Django

이번 프로젝트 자체가, 웹 어플리케이션을 제작하는 것이기 때문에 겨울방학때 공부

했던 Django를 활용해서 웹 어플리케이션을 제작하자고 결정했습니다. Django에 대하여 간단히 설명하겠습니다.

Django는 파이썬으로 작성된 오픈소스 웹 어플리케이션 프레임워크입니다. 웹 개발에서 필요한 기능들을 미리 구현해 놓았기 때문에 개발자가 필요한 기능들을 쉽게 구현할 수 있습니다. 또한, Django는 Model-View-Controller (MVC) 패턴을 기반으로 한 Model-View-Template (MVT) 패턴을 사용합니다. 이 패턴은 데이터베이스와의 상호작용을 쉽게 만들어주며, 웹 페이지를 구성하는 HTML/CSS/JavaScript 코드와 서버 측 로직을 분리하여 유지보수와 확장성을 높여줍니다. 마지막으로 Django는 데이터베이스 연동, URL 라우팅, 템플릿 엔진, 폼 처리, 인증 등 다양한 기능을 제공합니다. 이 외에도 다양한 패키지와 라이브러리가 있어서, 개발자들은 Django를 이용해 웹 애플리케이션을 빠르고 쉽게 만들 수 있습니다.

위와 같은 이유로 Django를 채택하였습니다.

## 2) Python & Python Open Library

Python을 주 개발 언어로 결정한 이유는, Django가 파이썬으로 작성된 프레임워크이기 때문입니다. 또한, 파이썬은 많은 개발자 개발하여 배포하고 있는 오픈소스 라이브러리가 있기 때문에, 모듈들을 비교적 쉽게 구현하기 위해 선택하였습니다.

## 3) BootStrap

부트스트랩을 이번 프로젝트에 사용하는 오픈소스로 채택한 이유는 부트스트랩(Bootstrap)이 웹 사이트 및 애플리케이션을 구축하기 위한 오픈소스 프레임워크이기 때문입니다. 부트스트랩은 CSS, JavaScript 및 HTML을 사용하여 템플릿 및 인터페이스 요소를 제공하기 때문에, Django에서 작업하는 html 역시 부트스트랩으로 포장하여, 더욱 더 유저편의성을 높인 UI를 제작할 수 있습니다.

## 4) AWS(Amazon Web Service)

이 프로젝트를 통해 웹페이지를 호스팅해야하는데, 이를 위해 AWS를 이용합니다.

AWS(Amazon Web Services)는 아마존닷컴에서 제공하는 클라우드 컴퓨팅 플랫폼으로, 인프라스트럭처, 플랫폼 및 소프트웨어 서비스를 제공합니다. 또한 AWS는 다양한 서비스를 제공하며, 가상 컴퓨터, 데이터베이스, 저장소, 컴퓨팅, 분석, 인공지능, 보안, 애플리케이션 통합 등의 영역에서 다양한 서비스를 제공합니다. 마지막으로 AWS의 핵심 서비스인 Amazon EC2(Elastic Compute Cloud)를 통해, 가상 서버를 생성하고 실행하여 웹페이지를 호스팅 할 생각입니다.



## 4. 위험평가 및 대책

### 4-1. 위험요소 및 대책

웹 페이지를 호스팅할 때는 보안 위험성이 있을 수 있습니다. 이러한 위험성은 다양한 악성 유저들이 웹 페이지의 서비스를 다운시키려고 시도하는 공격들로부터 발생할 수 있습니다. 다음은 그러한 악성 공격의 예시입니다.

#### 1) DDoS 공격

DDoS(Distributed Denial of Service) 공격은 많은 수의 컴퓨터를 사용하여 대상 웹 페이지의 서버에 대량의 데이터를 보내어 서버를 다운시키는 공격입니다. 이를 방지하기 위해서는 적절한 방화벽 및 부하 분산 장치를 사용해야 합니다.

#### 2) 악성 코드 삽입

악성 코드를 삽입하여 사용자의 컴퓨터나 서버를 감염시키는 공격입니다. 이를 방지하기 위해서는 웹 페이지를 제작할 때 보안 취약점을 최소화하고, 코드를 정기적으로 업데이트하는 것이 중요합니다.

#### 3) SQL Injection 공격

SQL Injection 공격은 악의적인 SQL 쿼리를 사용하여 데이터베이스에 대한 액세스를 얻는 공격입니다. 이를 방지하기 위해서는 보안을 강화하는 SQL 쿼리를 작성하고, 웹 애플리케이션을 보호하는 보안 솔루션을 사용해야 합니다.

#### 4) 크로스 사이트 스크립팅(XSS) 공격

XSS 공격은 악성 스크립트를 사용하여 대상 사용자의 브라우저를 조작하는 공격입니다. 이를 방지하기 위해서는 웹 페이지의 입력 폼을 검증하고, 적절한 보안 프로토콜을 사용하는 것이 중요합니다.

## 5. 주요 용어 설명

### 1) Ping

Ping은 컴퓨터 네트워크에서 호스트나 IP 주소를 향해 패킷을 보내고, 해당 호스트로부터 응답을 받아서 네트워크 연결 상태를 확인하는 데 사용되는 유틸리티입니다.

Ping은 ICMP(Internet Control Message Protocol)를 사용하여 호스트 간의 통신을 테스트합니다. ICMP는 네트워크 상태를 모니터링하고, 오류 메시지를 전송하거나, 라우팅 문제를 진단하는 등의 작업에 사용되는 프로토콜입니다.

### 2) Traceroute

Traceroute는 인터넷 네트워크에서 패킷이 전송되는 경로를 추적하고, 패킷 전송 시

간을 측정하는 도구입니다.

Traceroute는 ICMP(Internet Control Message Protocol) 또는 UDP(User Datagram Protocol) 패킷을 사용하여 라우터 간에 전송된 패킷의 경로를 추적합니다. Traceroute는 시작점에서 목적지로 패킷을 보냅니다. 이 패킷은 TTL(Time to Live) 값을 가지며, 이 값은 라우터를 지날 때마다 1씩 감소합니다. 패킷이 TTL 값이 0이 되면 해당 라우터에서는 버려지고, ICMP 혹은 UDP 패킷을 발생시켜 시작점으로 경로 정보를 보냅니다. 이런 과정을 반복하면서 목적지까지 전송되는 경로를 추적합니다.

### 3) Uplink & Downlink

Uplink는 사용자 장치(예: 스마트폰, 노트북 등)가 인터넷 서비스 제공업체(ISP) 또는 통신사와 통신하기 위해 사용하는 연결 경로를 의미합니다. 이 연결 경로는 일반적으로 사용자 장치에서 서버 또는 인터넷으로 데이터를 보내는 데 사용됩니다.

Downlink는 반대로 서버 또는 인터넷에서 사용자 장치로 데이터를 전송하는 것을 의미합니다. 예를 들어, 사용자가 스마트폰으로 웹 페이지를 열 때, 서버에서 스마트폰으로 데이터를 전송하는 과정이 downlink입니다.

## 6. 참고문헌(citations)

1. <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
2. <https://docs.python.org/3/>
3. <https://www.scylladb.com/glossary/ttl-value/>
4. <https://namu.wiki/w/ping>
5. <https://en.wikipedia.org/wiki/Traceroute>