2023-2 Open-Source Software Project

**오픈소스 소프트웨어**

**프로젝트 중간보고서**

**2023.11.04.**

OSSP 1팀 오소리

2020112521 장 민

2021110765 최효은

2021110472 김미소

**목차**

[**1. 프로젝트 개요** 2](#_Toc150051188)

[**1.1 프로젝트 주제** 3](#_Toc150051189)

[**1.3 선정 오픈소스** 3](#_Toc150051190)

[**1.3.1 선정 오픈소스 정보** 3](#_Toc150051191)

[**1.3.3 선정 이유** 4](#_Toc150051192)

[**1.3.4 개선 방향성** 4](#_Toc150051193)

[**2. 프로젝트 변경 사항** 5](#_Toc150051194)

[**2.1 개발환경 및 라이선스** 5](#_Toc150051195)

[**2.2 프로젝트 개선 및 개발사항** 5](#_Toc150051196)

[**3. 프로젝트 진행 사항** 5](#_Toc150051197)

[**3.1 프론트엔드\_최효은** 5](#_Toc150051198)

[**3.2 데이터베이스\_김미소** 5](#_Toc150051199)

[**3.3 백엔드\_장민** 7](#_Toc150051200)

[**4. 남은 과제** 8](#_Toc150051201)

[**4.1 회원 가입** 8](#_Toc150051202)

[**4.2 졸업 요건 검사** 8](#_Toc150051203)

[**5. 프로젝트 일정** 8](#_Toc150051204)

[**5.1 기존 일정** 9](#_Toc150051205)

[**5.2 일정 변경사항** 9](#_Toc150051206)

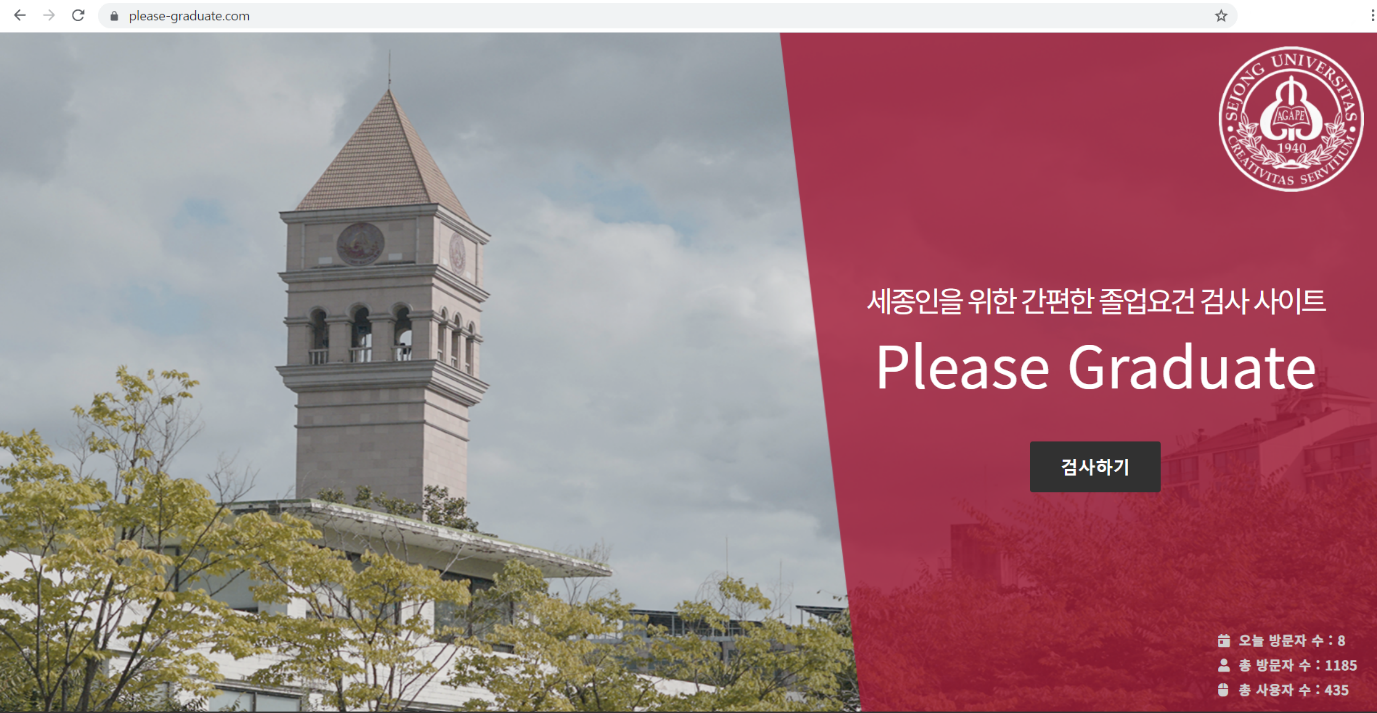
**1. 프로젝트 개요**

**1.1 프로젝트 주제**

본교는 졸업을 위해 공통교양, 기초소양, 전공필수, 영어강의 등의 여러가지 조건을 요구한다. 그러나, 졸업사정을 위해 이렇게 여러가지 요소에 대한 고려가 필요함에도 불구하고 졸업사정관리에 대한 기존 시스템에는 여러가지 한계들이 존재한다.

본 프로젝트는 오픈소스 Please Graduate를 분석하여, 본교의 졸업사정관리 구조에 적합한 형태로 변형하고자 한다. 이를 통해 본교 학생들이 간편하게 자신의 졸업요건 충족여부를 확인할 수 있게 하고 나아가서는 해당 학생의 졸업요건 충족을 위한 향후의 로드맵을 제공하는 것이 본 프로젝트의 목표이다.

**1.2 선정 오픈소스**



**1.2.1 선정 오픈소스 정보**

본 프로젝트를 위해 우리가 선정한 오픈소스는 Django를 이용해 구현한 졸업사정관리 시스템이다. [(Please Graduate)](https://github.com/hanjo8813/PleaseGraduate)

위 오픈소스는 실제 학생들만 사용하도록 학생 인증을 해야만 회원가입을 할 수 있게 되어 있다. 다만, 학생 인증에 학생이 실제로 학교사이트에서 사용하는 학번과 비밀번호를 이용하는 만큼 솔트를 통합한 암호화 해시 함수 Bcrypt와 SSL인증서를 사용해 보안취약점을 보완했다.

졸업요건을 검사할 때는 학사정보시스템에서 기이수성적 엑셀파일을 다운로드 받아 사이트에 업로드하고, 현재 수강하고 있는 과목 또는 향후에 수강하고자 하는 과목을 직접 추가한 뒤에 검사를 진행하도록 되어 있다.

졸업요건의 검사 결과에서는 영역별 달성도를 시각화하여 보여준다. 요건이 충족되지 않은 영역에 대해서는 Recommend 버튼을 통해 부족한 부분을 알려주고 충족을 위해 필요한 강의를 추천해준다.

그 외에도 부가적으로 ‘꿀교양 찾기’라는 기능을 제공하는데, 모든 사용자 데이터를 참조해 수강횟수를 기준으로 가장 많은 학생이 수강한 강의를 정렬해준다.

**1.3.2 선정 이유**

* 졸업사정관리 시스템에 필요한 기능들이 완성도 있게 구현되어 있다.
* 직관적인 UI 설계로 졸업 요건 가운데 충족된 부분과 그렇지 못한 부분을 한눈에 알아볼 수 있게 되어 있다.
* 지속적으로 업데이트가 진행되고 있고, 업데이트 내역이 상세히 정리되어 있다.
* 개발일지를 통해 개발에 대한 기록이 정리되어 있고, 각기의 개발 항목에 대한 설명과 전반적인 코드의 주석이 상세해 내용을 파악하기 용이하다.

**1.3.3 개선 방향성**

1) 추천 강의를 정리할 자체적인 기능의 부재

* 관심 교양 등록 기능 추가

1. 강의 추천 기준의 획일성

* 강의 평가 기능 개발

1. 단기적인 소모성의 문제

* 취업 정보 관련 페이지와의 연동

**2. 프로젝트 변경 사항**

**2.1 개발환경 및 라이선스**

**2.2 프로젝트 개선 및 개발사항**

**3. 프로젝트 진행 사항**

**3.1 프론트엔드\_최효은**

**3.2 데이터베이스\_김미소**

**3.3 백엔드\_장민**

1. 서버 구축

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

1) Amazon Web Service(AWS) EC2 서버를 임대

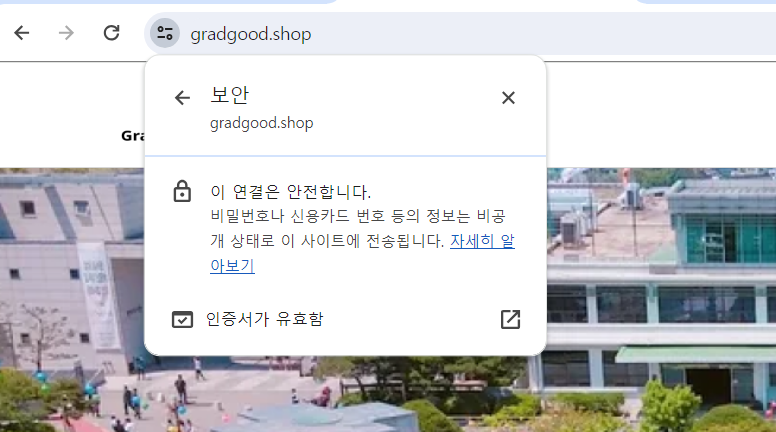
2) uWsgi 서버와 nginx 웹 서버를 Django 프로젝트와 연결해 데몬에 등록

3) static 파일 연결

4) 가비아를 통해 도메인 구입 후 AWS Route 53을 통해 해당 도메인 등록

(<https://gradgood.shop/>)

2. https 설정

****

Let's Encrypt를 사용하여 Nginx에 SSL 적용

1) HTTP (포트 80)으로 들어오는 모든 요청을 HTTPS로 리디렉션했다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

2) SSL 프로토콜은 TLSv1.2와 TLSv1.3을 사용하였고, SHA-384 해시 함수와 SHA-256 해시 함수, ECDHE 키 교환 및 256비트 AES 암호 등을 사용하여 데이터를 암호화하도록 암호화 스위트를 설정해주었다.

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명

3) 해당 SSL 설정의 적용 여부를 확인하기 위해 [www.ssllabs.com](http://www.ssllabs.com)에서 SSL Server Test를 진행하였고, 그를 통해 TLSv1.2와 TLSv1.3이 정상적으로 구동함을 확인했다.

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

3. 로그인 기능

로그인에는 블로피시 암호에 기반을 두고 salt를 통합한 암호화 해시 함수 bcrypt를 사용하였다.



bcrypt 인자는 bytes로 입력을 받기 때문에 인코딩을 진행한 후 salt를 생성하고 해싱을 진행하는 과정을 거쳐 비밀번호를 암호화하고 이를 디코딩하여 다시 문자열로 변환한 뒤 데이터베이스에 저장해주었다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

이후 로그인이 진행될 때는, bcrypt가 제공하는 checkpw() 메소드를 이용해 입력 받은 비밀번호를 암호화해 DB의 암호화된 비밀번호와 비교하였다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**4. 남은 과제**

**4.1 회원 가입**

**4.2 졸업 요건 검사**

**5. 프로젝트 일정**

**5.1 기존 일정텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명**

**5.2 일정 변경사항**