|  |
| --- |
| **SSU-Nice-Lecture-Timetable**  **나반, 2팀, [ 20211735, 20211730, 20211731, 20213057 ]** |

|  |  |
| --- | --- |
| **2. 요약**  u-SAINT 강의시간표 조회 페이지의 불편한 점 10가지를 개선하는 것이 목표이다.  현재, 불필요하거나 중요도가 낮은 내용으로 인해 가독성이 매우 떨어지는 상황이다.  중요도를 토대로 열을 재배치하고, 중복된 텍스트를 제거함으로써 보다 간결하고 직관적인 사이트를 완성하는 것이 목표이다.  본 프로젝트를 계기로 u-SAINT가 개선되어 더 이상 외부 서비스에 의존하지 않고 편리하게 학사 관리를 할 수 있기를 바란다. | **3. 대표 그림**  - 중복을 제거하고 아이콘을 활용하여 강의 목록을 훨씬 직관적으로 개선하였다.    그림 1. 강의시간표 조회 페이지 [(원본 이미지)](https://s3.us-west-2.amazonaws.com/secure.notion-static.com/5fe08fc3-27de-42a3-be69-436d47927d05/Untitled.png?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Content-Sha256=UNSIGNED-PAYLOAD&X-Amz-Credential=AKIAT73L2G45EIPT3X45%2F20211205%2Fus-west-2%2Fs3%2Faws4_request&X-Amz-Date=20211205T090913Z&X-Amz-Expires=86400&X-Amz-Signature=b26eca03c0c4389596c4900d6af9e3a242a86745b722c75d04d229c7eb77a407&X-Amz-SignedHeaders=host&response-content-disposition=filename%20%3D%22Untitled.png%22&x-id=GetObject) |

|  |
| --- |
| **4. 서론**  **배경 설명**  매 학기마다 이용하게 되는 강의시간표 조회 페이지의 가독성이 매우 떨어지기 때문에, 에브리타임 등의 외부 서비스를 활용하여 시간표를 짜고, 그 과목을 다시 u-SAINT에서 찾아 신청하는 학생들이 많은 것으로 알고 있다. 이러한 번거로운 과정이 필요 없도록 u-SAINT 강의시간표 조회 페이지를 개선하고자 하였다.  **문제정의 및 해결방안**   1. 본인 학부의 과목을 수동으로 선택하여 조회하여야 함. -> 학적 정보에서 학부와 전공을 불러와서 첫 화면에서 바로 표시되도록 개선함. 2. 전공을 선택하는 드롭다운에서 학부 레벨에 전공명이 들어있는 등 일관적이지 못함. -> 학부레벨에 전공명이 들어가 있는 등의 특이 케이스를 수정함. 3. 몇 년 이상 이용되지 않거나 폐지된 교과 구분의 탭이 여전히 남아있음. -> 평생교육사, 듀얼리스팅과목 등의 탭을 제거하여 탭 목록을 간소화함. 4. 불필요한 공백으로 인하여 중요한 정보가 밀리고 여러 줄로 나뉘어 표시됨. -> 다수의 과목에서 비어있는 열을 뒤로 이동하고, 아이콘으로 간소화하여 표시함. 5. 학부전공별 탭에서 이미 전공을 선택했으므로 표에 수강 대상을 표시할 필요 없음. -> 수강 대상 열을 삭제함. (해당 정보는 강의계획서에서도 확인할 수 있음) 6. 수업 시간이 12:00 ~ 12:50, 13:00 ~ 13:50과 같이 50분 단위로 나뉘어 표시됨. -> 실제 수업은 연속하여 진행되므로 합쳐서 표시함. 7. 교수명이 교수명열에 중복되어 표시되고, 강의시간열에도 추가로 표시됨. -> 강의시간에서 교수명을 제거하고, 중복된 교수명을 수정함. 8. 과목명에 수업방식, 수강대상 등 목적을 벗어난 표현이 함께 적혀있음. -> 과목명만 남기고 나머지 내용은 아이콘 혹은 해당 열로 분리함. 9. 과목마다 칸의 높이가 들쭉날쭉하여 가독성이 매우 떨어짐. -> 4번 문제가 원인이므로 4번을 먼저 해결하고 과목마다 높이를 통일함. 10. 과목 검색에서 학부 과목과 석사, 박사 과목이 뒤섞여 검색됨. -> 학적 정보에서 학부, 석사, 박사인지를 확인하여 해당하는 과목만 표시함.  (전체 과목을 보고 싶을 경우 “전체 과목 표시” 토글을 이용할 수 있음) |

|  |
| --- |
| **5. 본론**  **시스템 개요**    **기술 요소**   * Oracle Cloud Infrastructure: 클라우드 컴퓨팅 플랫폼, 현재 서버와 DB를 본 클라우드 환경에서 운용하고 있음. * Freenom: 무료 도메인 제공, 현재 mathpaul3.gq와 [hoyongee.ga](http://hoyongee.ga/) 도메인을 등록하여 사용하고 있음. * PuTTY: 오픈소스 단말 에뮬레이터 프로그램, SSH로 서버에 접속할 때 사용됨. (저장소: <https://git.tartarus.org/?p=simon/putty.git>) * MySQL: 오픈소스 관계형 데이터베이스 관리 시스템(DBMS) (저장소: <https://github.com/mysql/mysql-server>) * Heidi SQL: 오픈소스 데이터베이스 클라이언트, DB 구성 과정에서 사용됨. (저장소: <https://github.com/HeidiSQL/HeidiSQL>) * VS Code Jupyter Extension: VSC에서 \*.ipynb 파일로 작업하기 위한 오픈소스 프로젝트, 데이터 정제 과정에서 사용됨. (저장소: <https://github.com/Microsoft/vscode-jupyter>) * Pandas: 데이터 조작 및 분석을 위한 오픈소스 파이썬 라이브러리, 데이터 정제 과정에서 사용됨. (저장소: <https://github.com/pandas-dev/pandas>) * Git: 오픈소스 분산 버전 관리 시스템. (저장소: <https://git.kernel.org/pub/scm/git/git.git>) * Vue.js: 오픈소스 프로그레시브 자바스크립트 프레임워크. (저장소: <https://github.com/vuejs/vue-next>) * Express.js: 오픈소스 웹 서버 프레임워크. (저장소: <https://github.com/expressjs/express>) * Node.js: 오픈소스 자바스크립트 런타임. (저장소: <https://github.com/nodejs/node>) * Adobe XD: 상용 UI 디자인 소프트웨어, 프론트엔드 UI 디자인에 사용됨. * Notion: 상용 메모 작성 및 프로젝트 관리 소프트웨어, 프로젝트 계획과 진행 상황 등을 문서화하는데 사용됨.   **개발 단계**   1. 요구사항 정의 - uSAINT 강의시간표의 불편한 점 찾기. → 서론의 10가지 문제점. 2. 프로젝트 설계 – 개선 방법 및 구현 계획 수립, 프로젝트 구조 설계 및 UI 디자인. 3. 프로젝트 구현 - 전반부: 데이터 정제 및 DB 구성, 후반부: 프론트엔드 및 백엔드 개발 4. 프로젝트 시연 – 테스트를 통해 디버깅 진행. 5. 프로젝트 배포 – Docker를 활용하여 배포.   **구현 방법**  uSAINT에서 강의 목록을 가져와서 정규표현식과 Pandas를 통해 필요한 정보를 추출 및 정제하고, csv 형태로 저장하여 DB를 구성한다. 기존 강의시간표의 개선할 점을 토대로 백엔드는 Express.js로, 프론트엔드는 Vue.js로 구성하여 강의시간표 조회 페이지를 구현한다. |

|  |
| --- |
| **6. 결론**  - 보고 내용 요약  - 향후 할일 정리 |

\* 7번 출처 제외 총 3장 이내 (파란색 글은 삭제 할 것), 기한 내에 제출 할 것 (10점)

**7. 출처**

1. 우재남, 이것이 우분투 리눅스다(개정판), 한빛미디어, 2020.
2. 숭실대학교 u-SAINT, <https://saint.ssu.ac.kr/irj/portal>
3. 고려대 수강신청 페이지, [https://sugang.korea.ac.kr](https://sugang.korea.ac.kr/)
4. 국민대 수강신청 페이지, <https://sugang.kookmin.ac.kr/front/enrollment/course-catalog>
5. Python 3.10.1 documentation , [https://docs.python.org/3](https://docs.python.org/3/)
6. Python re library documentation, <https://docs.python.org/3/library/re.html>
7. Python CSV library documentation, <https://docs.python.org/3/library/csv.html>
8. Python JSON library documentation, <https://docs.python.org/3/library/json.html>
9. Pandas documentation, [https://pandas.pydata.org/docs](https://pandas.pydata.org/docs/)
10. regular expressions 101, [https://regex101.com](https://regex101.com/)
11. 점프 투 파이썬 07-2 정규 표현식 시작하기, <https://wikidocs.net/4308>
12. [Python 문법] 정규표현식 (Regular Expressions), [https://nachwon.github.io/regular-expressions](https://nachwon.github.io/regular-expressions/)
13. XD UI Kit, <https://assets.adobe.com/public/595fabd0-da61-49bb-6117-729c5405d84b>
14. Tab Component, <https://bootstrap-vue.org/docs/components/tabs>
15. Icon Hover, <https://developer.mozilla.org/ko/docs/Web/CSS/:hover>
16. Multilevel Accordion, <https://codepen.io/kolya/pen/gOwOjmO>
17. Vue Router, <https://www.youtube.com/watch?v=PQRozQ5fGKo>
18. MySQL 외부 접속, <https://pythonblog.co.kr/blog/8/>
19. 도메인 할당, <https://blog.lael.be/post/6070>
20. 우분투 웹서버 세팅, <https://blog.lael.be/post/73>
21. A Apache2 서버를 nginx로 교체, <https://marshall-ku.com/web/tips/apache2-서버를-nginx로-교체하기>
22. Ubuntu에 nginx 설치하고 vue.js 설치, <https://hplayground.tistory.com/105>
23. LINUX 사용중인 포트, 프로세스 확인, <https://kugancity.tistory.com/entry/리눅스-포트-사용중인-프로세스-확인-방법>
24. LINUX 서비스 목록 확인, 서비스 명령어, <https://fabxoe.tistory.com/130>
25. Vue.js Invalid host header 오류, <https://cubenuri.tistory.com/409>
26. Oracle Cloud 포트 열고 관리하기, <https://kibua20.tistory.com/124>
27. Dockerfile, .dockerignore,   
    [https://docs.docker.com/engine/reference/builder/#dockerignore-file](https://docs.docker.com/engine/reference/builder/)
28. Docker image 변경, 저장: [https://losskatsu.github.io/it-infra/docker03/#도커docker3-이미지-변경-후-저장commit](https://losskatsu.github.io/it-infra/docker03/%23도커docker3-이미지-변경-후-저장commit)

단행본 - 저자, 제목, 출판사, 출판년도.

웹페이지 – 작성자, “제목”, 웹사이트명, 발행날짜, URL.