

3학년 1학기 동아리 활동 보고서

3610양세진

-동아리 홍보 웹 페이지 만들기 활동내용:

나, 차성민, 최준, 서하람 이렇게 네명이서 팀을 꾸려 동아리 홍보 웹 페이지 만들기 활동을 하였습니다.

그리고 각자 웹 템플릿을 찾아 서로의 것들중 가장 디자인이 괜찮은 것으로 활동을 진행하기로 하였습니다.

과반수 의견으로 성민의 웹 템플릿으로 활동을 진행하기로 하였습니다.

넷트리파이(Netlify)라는 무료 웹 호스팅 사이트를 이용하여 우리가 만든 웹 페이지를 호스팅하기로 하였습니다.

각자 웹 템플릿에 있는 페이지에서 어떤 페이지를 담당할지 역할 분담을 하였습니다(역할분담: 서하람-about, 양세진-project, 차성민-home, 최준-sevice).

우리가 인터넷에 접속하면 볼 수 있는 수많은 웹사이트들은 간단하게 이렇게 구성되어 있다.

웹 페이지의 뼈대를 구성하는 **html**, 디자인을 담당하는 **css**, 웹 페이지 내의 기능을 담당하는 **javascript**,

이 외에도 동적인 사이트들은 각종 프레임워크(동적인 웹을 위해 필요), 라이브러리, **DB**(데이터 베이스)등을 갖추고 있는데 저희가 만들 사이트는 정적인 사이트로 **html**, **css**, **javascript**, **scss**정도로만 이루어져 있어 저희가 손댈 부분은 이곳들이고, 마지막으로 웹 호스팅만 하면 배포가 끝나는 것입니다.

저희는 우선 페이지의 **<title>**과 페이지의 상단에 위치한 로고 이미지, 글자, 파비콘(인터넷 웹 브라우저에 표시되는 웹사이트나 웹페이지를 대표하는 이미지)들을 수정하였습니다.

저는 **project.html** 페이지의 동아리 활동 소개를 담당하였습니다. 우선 상단의 **navbar**라는 곳에 **project**라고 되어 있는 부분을 '동아리 활동 소개'로 변경하였습니다.

그리고 나머지 '**project**'라고 되어 있는 부분도 모두 '동아리 활동 소개'로 변경하였습니다.

저는 웹페이지를 수정하는 과정에서 한가지 어려움을 깨닫게 되었는데, 4인이서 웹페이지를 수정하면 각자의 컴퓨터에서 따로 웹 템플릿 파일을 수정하고 있었기 때문에 마지막에 결과물을 합칠때 약간의 번거로움이 생긴다는 것을 깨달았습니다.

그래서 저는 서로 다같이 수정한 파일을 공유하는등 함께 협업할 수 있는 개발 환경의 필요성을 느꼈습니다. 저는 친구에게 이에 대해 물어보았는데 **Netlify**에 이 기능이 있는데 유료 기능이라서 활용하지 못하고 있었다는 사실을 알게 되었습니다.

그래서 저는 이전에 '**git**'이라는 걸 활용하면 협업을 할 수 있다는 것을 본것이 생각나서 팀원들에게 이걸 활용해보는 건 어떠냐고 의견을 물어보았습니다.

답변은 긍정적이었고, 저는 '**git**'에 대해 알아보겠다고 말했습니다.

git은 '소프트웨어 개발에서 코드 및 기타 파일의 변경 사항을 추적하고 관리하는데 사용되는 분산 버전 관리 시스템'이라고 하는데, 이 **git**은 오픈 소스 소프트웨어로 많은 소프트웨어 프로젝트에서 사용되며, 특히 팀 프로젝트에서 코드 협업에 유용하다고 합니다. 추가로 안 사실은 이 '**git**'을 활용하면 코드의 저장 뿐만 아니라, 수정하기 이전 상태의 코드로 돌아가기, 변경 사항 추적하기 등의 다양한 기능을 할 수 있다는 것을 알게 되었습니다.

처음 접해보는 것이라 많이 막막하였는데 일단 무작정 블로그에 올라온 **git**과 **github**(버전 관리와 협업을 위한 코드 웹 호스팅 플랫폼)을 활용하여 협업 환경을 셋팅하는 법을 따라해보기로 하였습니다.

먼저 컴퓨터에 **Git**을 설치하였습니다. 그리고 코드 에디터 **vscode**에 **Git** 명령어를 사용할 수 있도록 셋팅을 하였습니다.

그리고 **github**에 들어가 리포지토리(**Repositories** = 저장소)를 생성한 후 제가 생성한 리포지토리의 **HTTPS**주소를 복사하였습니다.

그리고 **vscode**에 터미널에 **git clone** '복사한 리포지토리 **HTTPS**주소' 이 명령어를 입력하여 제 컴퓨터에 프로젝트를 복제하였습니다.

저는 이 **git**을 이제 막 배우는 단계이기 때문에 최소한의 기능만을 사용하고자 하였습니다(애초에 다른 기능들은 필요치 않기도 하였습니다.)

아래 명령어들을 이용하여 저는 프로젝트를 관리 하였습니다.

git add . (**Git** 저장소에 있는 모든 변경 사항을 스테이징 영역에 추가하는 명령어. 즉, **Git**이 다음 커밋에 포함할 파일들을 준비하는 과정)

git commit -m "변경사항 메모" (**Git** 저장소에서 파일의 변경 사항을 영구적으로 기록하는 과정)

git push (원격 저장소(remote repository)에 코드 변경분을 업로드하기 위해서 사용하는 **Git** 명령어)

제가 코드를 수정하고 마무리할때 **git add .** 명령어를 사용해 변경사항들을 커밋할 준비를 하고, **git -m commit** 명령어를 사용해, 변경사항을 기록하고 간단한 메모를 남겨 제가 어떤 부분을 수정하였는지를 기록하였고, 마지막으로 **git push**를 활용해 제가 만든 리포지토리에 변경된 코드들을 업로드 하는 식으로 **git**을 사용하였습니다.

그리고 이 **git**을 사용할때 주의해야 할 점이 한가지 있는데 다른 사람도 같이 제 저장소를 공유하기 때문에 제가 코드를 수정한 파일을 **push**했다면 다른 사람은 항상 코드를 자신의 컴퓨터에서 수정하기 전에 **git pull** 명령어를 사용하여 업데이트를 해야 한다는 것입니다.

세번째 동아리 활동 시간때 저는 친구의 컴퓨터에 협업할 수 있는 환경을 셋팅하기 위해서 **git**을 설치하였는데 윈도우 **powershell** 터미널에서 **git**명령어를 사용할 수 없는 문제가 발생했습니다.

그래서 이 때문에 다음 단계를 진행하는데 어려움을 겪고 있었는데 **vscode**내에 **git bash** 터미널이 생긴 것을 보고 이것을 이용하여 친구의 컴퓨터도 제 컴퓨터처럼 협업을 할 수 있는 상태로 세팅하였고 잘 커밋되는지 확인하였습니다.

지금 와서 생각해 보면 별거 아닌 간단한 것이지만 그때 당시에는 git에 대한 지식이 아예 없던 때라 친구의 컴퓨터에서 환경을 다 셋팅하고 나서 엄청난 성취감을 느꼈던 것 같습니다.

저는 집에서 project.html 페이지의 남은 요소들을 수정하기 위해 페이지에서 수정하기 좋은 부분은 없는지를 찾아보았습니다. 페이지에 가장 많은 부분을 차지하는 것이 이미지 파일들이었는데 저는 우리 동아리 활동과 코딩에 대한 이야기를 이 이미지가 들어간 곳에 넣으면 좋겠다는 생각이 들었습니다.

저는 카드뉴스 형식으로 아래와 같이 이미지를 만들어 웹사이트에 이미지 요소로 추가하였습니다.



저는 이 사이트에 방문할 저희 동아리 학생들, 그리고 앞으로 저희 코딩 동아리에 들어올 학생들을 고려하여 이 페이지에 카드뉴스를 제작하였습니다.

첫번째 사진은 저희 코딩동아리에서 하는 활동을 알리려는 목적으로 제작하였고,

두번째 사진은 코딩이란 무엇인지에 대해 간략히 적어놓은 글입니다.

세번째 사진은 처음 프로그래밍 언어를 고를때 무엇을 고를지 막막해하는 학생들이 있을 것 같아 참고가 되라고 각 언어들을 몇개 선정한 뒤 그 언어의 특징을 간략히 적어 놓았습니다.

네번째 사진은 코딩을 공부하는 방법을 3가지 정도 적어 놓았고,

다섯 번째 사진은 코딩 관련 진로 지식을 적어놓은 사진인데, 코딩에 관심있는 학생들 중 코딩관련 진로를 희망하는 학생들을 위해 도움이 될 만한 정보들을 간략히 적어 놓은 것입니다.

여섯 번째 사진은 개발자가 가져야 할 역량을 적어 놓은 것입니다. 저는 개발자가 단순히 코딩만 잘해서는 끝나지 않는다고 생각하여 제가 생각하는 개발자가 가져야 할 역량들을 적어놓아 학생들이 이를 보고 자신은 어떤 개발자가 되어야 했는지를 생각하게끔 만들었습니다.

저는 **github**을 둘러보던 도중 이 **github**도 호스팅 서비스가 있다는 것을 알게 되었고 깃헙 **Pages**서비스를 활용해 제 저장소에 있는 동아리 홍보 웹사이트를 호스팅 해보았습니다.

전에 처음 **html**이 웹페이지에서 어떻게 구현되는지를 눈으로 보기 위해서 **vscode** 에디터에 있는 확장 프로그램인 **Live Server**를 활용해 로컬환경(제 컴퓨터)에서 **html**이 어떻게 구현되는지를 확인했었는데 이는 로컬환경에서 구현된 것이라, 실제로 다른 사용자들이 주소창에 제 **ip**를 입력해도 제 웹사이트에 접속할 수 없다는 한계점이 있었습니다. 그래서 그때 당시 저는 어떻게 하면 내가 만든 웹사이트에 다른 사람들이 접속할 수 있도록 할 수 있는건지에 대해 궁금증이 생겼었고, 이에 대해 찾아보니 호스팅 서비스를 이용해야 된다는 것을 알게 되었습니다. 그런데 호스팅 서비스를 이용하기 위해서는 서비스를 해주는 회사에 이용료를 내야 한다는 것을 알고 호스팅을 하는 것은 포기하였습니다. 그런데 **github**같이 무료 호스팅을 지원하는 사이트가 있다는 것을 알게 된 저는 이에 큰 기쁨을 느꼈습니다. 친구에게 제가 호스팅한 코딩 동아리 홍보 사이트에 직접 접속해달라고 부탁하고 정말로 접속되는 것을 본 순간 저는 무언가 드디어 해냈다는 기분을 느낄 수 있었습니다.

저는 이후 계속 웹사이트를 약간씩 수정해가며, 중요한 부분을 수정하고 호스팅중인 웹페이지에서 어떻게 구현되는지를 눈으로 보려고 하는데 갑자기 제가 커밋/**push**한 코드가 적용되지 않는 일이 발생하였습니다.

저는 처음에는 제가 연결한 파일의 상대경로가 잘못되었나 계속 수정-커밋-**push**하며 호스팅 중인 웹페이지의 화면을 확인하였는데 계속 수정사항이 적용되지 않는 것이였습니다. 계속 인터넷이 이러한 문제를 해결하는 방법을 찾다가 **github**의 오류 알림을 보라는 내용이 있어 알림을 보니 **commit**(커밋)을 단시간에 너무 많이 해서 이것이 밀려, 수정사항이 적용되지 않는 오류가 일어난 것임을 알게 되었고, 호스팅 중인 웹사이트를 내렸다 다시 올리니 이를 해결할 수 있었습니다.

문제를 해결하기 위한 과정에서 계속 커밋을 했었는데 이것이 문제를 심화시키고 있었다는 것을 깨닫고는 이 일에 대해 돌아보게 되었습니다. 그리고 또한 커밋은 자주하면 효율적이지 못하고 모든 작업을 끝냈을 때 한번 하는 것이 효율적임을 깨닫게 되었습니다.

[이번 활동을 통해 느낀점]

제가 이번 활동을 통해 느낀점은 역시 웹 개발은 재밌다는 것입니다. 바닥부터 하나하나 직접 저희가 개발하지는 않고 웹 템플릿을 이용했지만, 개발은 만들어진 재료들을 적절히 조합하는 것인만큼, 이번 활동은 저희에게 있어, 또 저 자신에게 있어 아주 의미있는 활동이었다고 생각합니다. 활동 중 저는 **html**에 있는 태그, 그리고 페이지가 다른 페이지와 어떻게 연결되는지 등을 알 수 있었습니다. 또 친구들과 어떻게 하면 서로 작성한 코드를 공유하며 공동으로 협업할 수 있을지를 생각하면서 **git**이란 것을 활용하는 과정에서 실무자들의 업무환경을 아주 간접적으로나마 느낄 수 있었습니다. 또 웹페이지의 여러

부분을 수정하면서 심심치 않게 작은 오류들이 발생했었는데 이것들을 수정해 나가면서 또 새로운 지식을 얻기도 하였습니다. 저는 웹 개발의 매력은 이런것이라고 생각합니다. 내가 직접 생각한 것을 구현하고, 디자인하고, 그것을 다른 사람들과 공유하는 것, 이것 때문에 제가 웹 개발에 관심을 가지게 된것 같습니다.