

리눅스 프로젝트 과제 안내

본 과제는 한 학기 동안 배운 Shell 기본 명령어, Shell 프로그래밍 그리고 Github를 얼마나 잘 다루는지 확인하는 과제입니다. 2인 1조로 팀을 이뤄 Git, Shell 코드, Shell 명령어를 조합하여 4가지 이상의 기능이 동작하는 소프트웨어를 간단히 구현합니다. 예시로 ps, netstat, git log 등을 간단히 분석하거나 보여주는 툴을 만들 수 있으며, 환경변수 등을 shell을 통해 컨트롤 할 수 있습니다. 또한 Python, C언어로 작성한 소스코드를 shell 코드를 통해 input값을 넣어 실행시키는 기능을 구현할 수 있습니다. (다만, Python, C언어 사용은 필수가 아닙니다.) 위와 같은 기능을 어떻게 누가 구현할지 기획서를 통해 계획하고, 기획서 대로 최대한 구현합니다. 구현 과정에서 발생하는 산출물은 Local git, Remote github를 통해 관리하며 평가는 **하단 평가지표**를 따르므로 참고하면 됩니다.

결과물

1. 기획서 (~11/27)

: Git Repository URL(Public으로 생성할 것), 구현할 기능(4가지 이상), 역할 분배 (기능 구현, 통합, Github 관리 등), 사용할 Linux, Git 기능 등 명세

2. 보고서 (~12/04 18:00)

: 각자 Local Git Graph 캡쳐본, 기획서와 구현된 소프트웨어 비교 등, 평가지표에 따라 평가 가능한 요소 추가

3. 발표자료 (~12/04 18:00)

: 5분 내외 소프트웨어 목적과 기능 설명, 구현 중 어려웠거나 오류가 발생했던 부분 및 해결 방법, 수업 외로 찾아서 도입한 기술 등

프로젝트 평가지표 (총점 20점, 항목당 4점)

1. 기획서, 보고서, 발표자료를 통한 수업 성실도 및 창의성 평가 (5점)

- 이해도: 수업내용(Shell, Git) 등을 얼마나 잘 이해하고 있는가?
- 응용능력: 사용 가능한 기술을 얼마나 유기적으로 조합하였는가?

2. 프로젝트 기능 퀄리티 (5점)

- 제출한 기획서와 기능이 얼마나 일치하는가?
- 수업 내용을 얼마나 다양하게 포함하여 구현했는가? (Shell, Git 등)

3. Git/Github를 얼마나 의미있게 사용했는가? (5점)

- Branch 사용, PR, 의미있는 커밋 등을 얼마나 많이 포함하고 있는가?
- 예상치 못한 충돌이나 오류를 어떻게 해결했는가?

4. Git/Github 사용 조건을 만족하였는가? (5점)

- main (master) branch에 직접 Commit하지 않을 것
- 기능마다 local/remote에 feature branch를 만들고, 산출물은 remote에서 관리할 것