

기말과제 레포트

|  |  |
| --- | --- |
| 과목명. | 오픈소스SW기초 |
| 담 당. | 윤 종 민 교수님 |
| 제출일. | 2020년 12월 20일 |
| 소프트웨어융합대학 | 소프트웨어학과 |
| 학 번. 2020052551 | |
| 이 름. | 성주원 |

****

목차

기존 프로젝트 분석으로 결정하였습니다

1. 프로젝트 소개
2. 프로젝트 선정 이유
3. 해당 프로젝트의 코드 분석 내용
4. 프로젝트에 적용된 알고리즘 정리
5. 어떤 커뮤니케이션 방법을 취해 프로젝트를 진행시키는지
6. 플로우 차트
7. 각 파일의 용도 및 설명

4. 프로젝트에 관한 개인적인 의견 및 리뷰

1. 프로젝트 소개

‘The Fuck’이란 프로젝트는 MIT 라이선스, 즉 오픈소스 라이선스를 따르는 프로젝트로 MIT 라이선스는 라이선스가 적용되는 소프트웨어는 누구라도 무상으로 제한없이 취급 가능 하지만 저작권 표시 및 라이선스 허가 표시를 소프트웨어의 모든 복제물 또는 중요한 부분에 기재해야 합니다. 또 저자 또는 저작권자는 소프트웨어에 관해서 아무런 책임을 지지 않는 라이선스가 MIT 라이선스입니다. 이 프로젝트는 직전에 났던 에러의 해결 방법을 예측해 자동으로 실행시키는 프로젝트입니다. 물론 자신만의 룰(alias)을 만들어 사용 가능합니다. 이 프로젝트는 100% 파이썬으로 만들어지며 152명의 기여자로 이루어져 있으며 약 220명의 사람들이 쓰고 있습니다. 이 프로그램을 쓸려면 파이썬 3.4버전 이상이, pip, python-dev가 필요합니다. OS 에선 Homebrew를 통해 ‘brew install thefuck’명령어로 설치 가능합니다. 우분투에선 ‘sudo apt install python3-dev python3-pip python3-setuptools’ 와 ‘sudo pip3 install thefuck’으로 설치 가능하며 만일 apt를 업데이트 하지 않았다면 ‘sudo apt update’를 통해 설치 전 업데이트를 해줘야 합니다. 윈도우 같은 경우 ‘pip install thefuck’으로 설치 가능합니다. 업데이트 하는 방법은 ‘pip3 install thefuck –upgrade’를 통해 업데이트가 가능합니다. 현재 ‘Experimental instant mode’로 빠르게 로딩할 수 있는 시스템을 구축하여 수정 중에 있습니다. 이 프로젝트에 기여하는 방법은 크게 두가지로 오류가 나는 부분을 리포트하는 방법과 코딩을 수정하여 오류를 없애거나 추가적인 기능을 추가하는 방법이 있습니다.

밑에 코드들은 ‘The Fuck’을 사용한 예제이다.

➜ apt-get install vim

E: Could not open lock file /var/lib/dpkg/lock - open (13: Permission denied)

E: Unable to lock the administration directory (/var/lib/dpkg/), are you root?

➜ fuck

sudo apt-get install vim [enter/↑/↓/ctrl+c]

[sudo] password for nvbn:

Reading package lists... Done

...

➜ lein rpl

'rpl' is not a task. See 'lein help'.

Did you mean this?

repl

➜ fuck

lein repl [enter/↑/↓/ctrl+c]

nREPL server started on port 54848 on host 127.0.0.1 - nrepl://127.0.0.1:54848

REPL-y 0.3.1

1. 프로젝트 선정 이유

처음에 이름 때문에 거부감이 일었지만 프로젝트의 설명이 저를 이 프로젝트를 보게 만들었습니다. 바로 직전에 발생한 오류를 자동적으로 고쳐주는 프로그램이란 점이 흥미로웠습니다. 개인적으로 프로그램을 짤 때 가장 어려운 점은 오류가 날 때입니다. 오류가 나면 오타일수도 있는 오류에 하루 종일 생각해 시간을 낭비하는 때가 많았습니다. 하지만 이 프로그램을 사용하게 되면 오류에 대한 해결 방법을 바로 실행시킬 수 있으니 저에겐 꿈만 같은 프로그램 같았습니다. 그래서 저는 이 프로그램의 실행 방법을 알고 싶어 조사하게 되었습니다. 또한 제가 그나마 익숙한 언어를 고르라면 파이썬이기 때문에 100% 파이썬으로 이루어진 이 프로젝트에 관해 코드를 본다면 조금이라도 쉽게 이해할 수 있을거라 생각해 알아보게 되었습니다. 또한 이 프로젝트를 알아가면서 이 프로젝트의 이름이 욕인 이유가 대부분 프로그래머들은 열심히 짠 코드에 오류가 발생하면 제 생각엔 대부분 욕을 하기 때문에 이렇게 제목을 지은건가 하는 생각도 들었습니다.

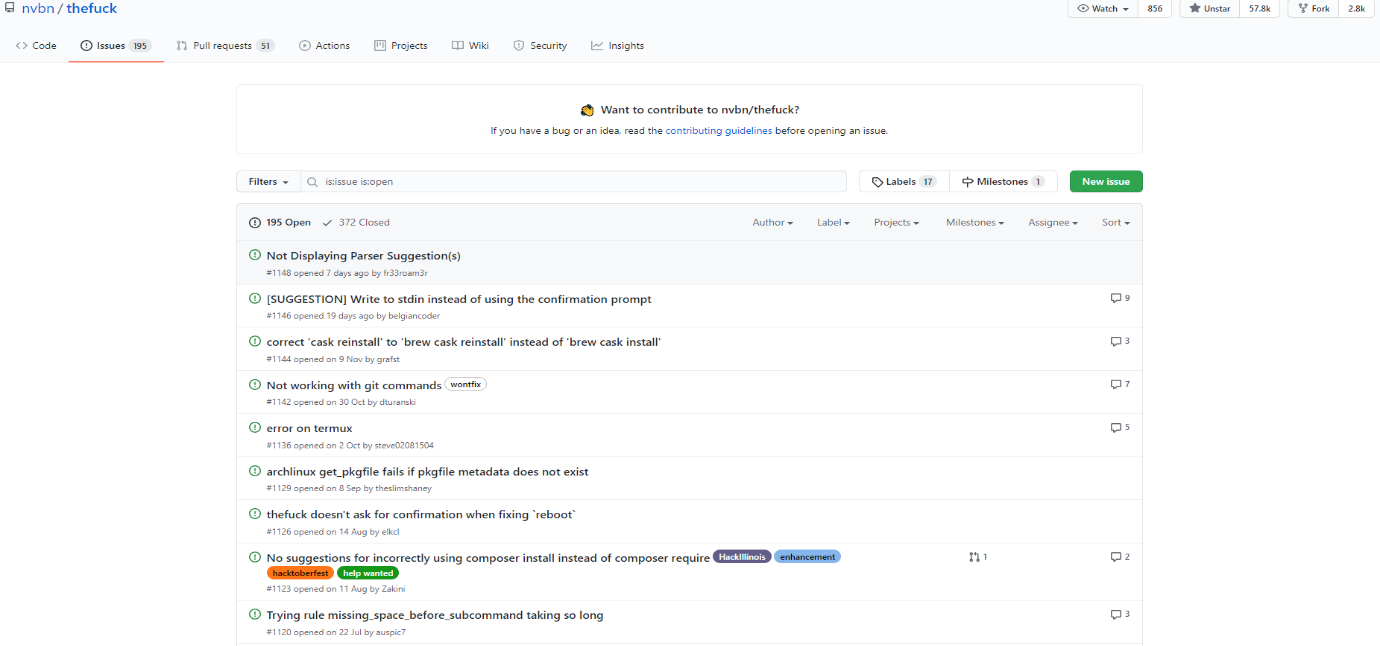
1. 해당 프로젝트 코딩 분석 내용

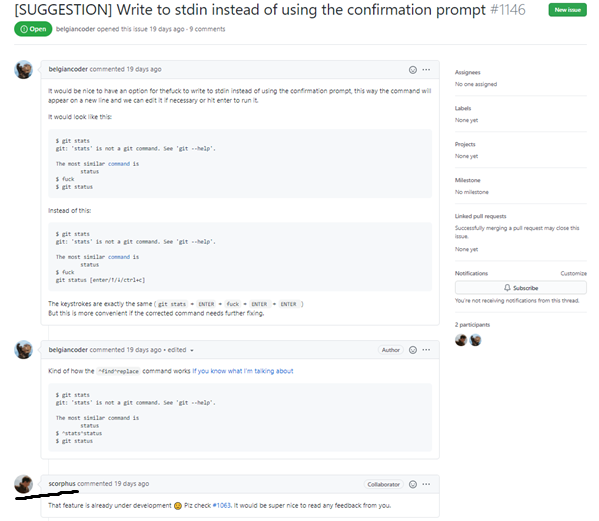
## 프로젝트에 적용된 알고리즘 정리

entrypoints 폴더의 파일에서 오류를 감지하고 exception을 실행시킨 후 시스템에서 권장하는 해결 방법을 입력받습니다. 만일 alias(미리 설정해 놓으면 실행되는 명령)에 저장된 명령이라면 바로 실행시킵니다. exception.py에서 만일 명령줄이 비워져 있거나 rules 폴더 안에 있는 rules에 해당이 안되고 log파일에서 찾지 못하면 예외처리를 합니다. 만일 시스템에서 해결 방법을 못 찾거나 alias로 설정된 명령어가 아니라면 rules에서 찾은 후 corrector.py에서 룰을 적용시킵니다. 만약 못 찾는다면 shell 명령어를 확인하기 위해 shells 폴더에 있는 룰을 확인해봅니다. specific 파일에서 설치 패키지나 권한에 문제가 있는지 확인하고 있으면 고칩니다. 만일 3가지 방법 중에 찾는다면 types.py에서 해결 방법이 적용된 명령줄을 실행시킵니다. 만일 실행시간이 너무 오래 걸리거나 파일을 가져올 수 없는 상태가 되면 재실행 시키거나 메시지로 유저에게 알립니다. 유저가 thefuck을 입력하거나 그에 맞는 옵션을 입력하면 argument\_parser.py에서 전달값을 받아 파이썬 데이터행으로 파싱합니다. 그 후 그에 맞는 설명서를 프린트합니다. 만일 옵션이 존재하지 않으면 옳지 않다면 다시 옵션을 보여주고 에러를 프린트합니다.

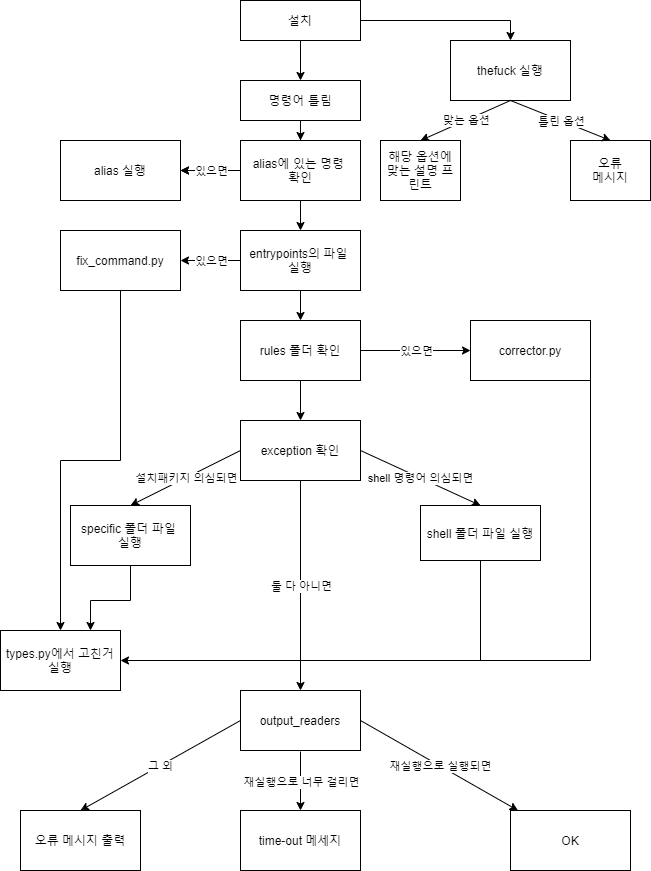
## 어떤 커뮤니케이션 방법으로 프로젝트를 진행시키는지

이 프로젝트에서는 이슈트래커를 통해서 커뮤니케이션을 진행하는 것으로 보입니다. 따로 메일링 리스트나 개별 사이트를 이용하는 것이 안 보였습니다. 이슈트래커의 제목에 bug, suggestion, enhancement, 등의 태그를 붙여 다양한 이슈에 관해서 소통하고 있습니다. README.md에서 Developing 부분에서 CONTRIBUTING.md로 들어가면 기여할 수 있는 방법들이 나열되어 있습니다.





1. 플로우 차트



1. 각 파일 용도 및 설명

.github 폴더는 개발 내용에 관해서 담겨있습니다. test폴더는 thefuck의 파일들을 베껴서 테스트용으로 따로 구분한 폴더입니다. 자세한 파일 내용은 thefuck폴더에서 설명합니다. .editorconfig는 형식 인자를 정의해 놓은 파일입니다. .gitignore은 일시적인 vum file를 저장하기 위해 만든 파일입니다. .travis.yml은 파이썬 2와 링크를 제거하기 위한 파일입니다. CONTRIBUTING.md는 기여하는 방법에 대한 문서입니다. LICENSE.md는 MIT라이선스에 대한 내용이 적혀있습니다. MANIFEST.in은 라이선스 파일과 fastentrypoints.py를 포함시킨 파일입니다. README.md는 thefuck에 대한 설명서입니다. appveyor.yml은 설치 경로와 환경 경로를 검사하는 파일입니다. example.gif는 설명서에 있는 gif입니다. example\_instant\_mode.gif는 빠른 gif용 파일입니다. fastentrypoints.py는 빠른 설치를 위한 셋업 파일입니다. install.sh는 설치 방법이 바뀌었을 때 알려주는 스크립트입니다. release.py는 설치할 때 버전 정보를 알려주는 스크립트입니다. requirement.txt는 설치 및 사용을 위한 프로그램들을 텍스트 파일로 정리해 놓은 파일입니다. setup.cfg는 셋업 마크다운을 위한 파일입니다. setup.py는 설치를 하는 파일입니다. snapcraft.yaml은 thefuck에 관해서 간단히 적어놓은 파일입니다. toxi.ini는 설치 조건을 확인하는 파일입니다.

-thefuck 폴더 안의 파일들에 관해서 설명

\_init\_.py(빈 파일). argument\_parser.py는 입력값을 가공하여 메뉴들 중 하나로 입력 받아오는 역할을 합니다. conf.py는 세팅과 사용자의 전달값의 다리 역할을 하는 파일입니다. const.py는 메뉴들을 보여주며 사용자가 단축키나 명령어, 혹은 방향키를 눌렀을 시 전달값을 보내주는 파일입니다. corrector.py는 thefuck의 핵심 부분이라고 할 수 있는 오류 난 부분을 해결방법을 받아와 바꿔주는 파일입니다. exceptions.py는 사용자가 예상치 못한 명령을 입력했을 시 오류 대신 예외 처리해주는 역할을 하는 프로그램입니다. logs.py는 오류 해결방법에 대한 해결책을 보여주고 확인하고 오류의 해결 방법에 대한 건의 사항을 찾지 못하거나 alias가 설정되어 있지 않거나 중복으로 설정되어 있을 경우 알려주는 파일입니다. (성공해도 알려줍니다) types.py는 특정 명령어에 대한 가이드 라인과 명령어 해결방법에 대한 확인을 받으면 실행시켜주는 파일입니다. ui.py는 ui와 관련된 파일들을 불러와 실제로 실행시키는 파일입니다. utils.py는 오류를 그대로 corrector.py로 가기 전에 alias 검사하고 가공하는 파일입니다. entrypoints 파일에는 제어하는 파일들이 담겨 있습니다. output\_reader.py는 오래 걸리는 명령어나 경고를 프린트해주는 프로그램들로 구성되어 있습니다. rules폴더엔 git과 관련된 오류에 관한 대응 방법들의 파일들이 있고 각종 룰들이 있는 파일들로 구성되어 있습니다. shells 폴더엔 alias에 관한 설명과 명령어들이 저장되는 파일들이 있습니다. specific 파일에는 flake8이란 파이썬 패키지로 코드 스타일을 검사해주는 건데 이를 고쳐서 쓰는 파일들과 yum에 사용 불가능한 명령어을 지원하도록 해주는 파일들로 이루어져 있습니다.

4. 프로젝트에 관한 개인적인 의견 및 리뷰

이 프로젝트는 초보 개발자이건 잘하는 개발자이건 다 유용하게 쓸 수 있는 프로그램이라고 생각합니다. 실제로 코딩에 난 에러가 아닌 콘솔에서 직전에 난 오류를 고쳐주는 프로그램이지만 사람은 누구나 실수를 하기 때문에 이러한 방식으로 오류를 빠르고 효율적으로 찾아내서 고치는 게 좋다고 생각합니다. 또 초보 개발자들은 오류를 모르는 경우가 많아 어느 부분에 오류가 나는지 몰라 고치는 게 쉽지 않은데 이 프로그램을 쓰면 훨씬 빠르고 쉽게 고칠 수 있다는 점에서 좋은거 같습니다. 또한 프로젝트가 152명이라는 많은 사람들에게 만들어지고 계속해서 개발되고 있다는 점이 신기했습니다. 앞으로 사회에 나가서 일하기 전 실력을 키워서 이렇게 효율적인 프로그램을 개발하는 프로젝트에 참가하고 싶다고 생각하게 되었습니다.