

AD 프로젝트 보고서

손원철(20203083)

주제 : 한/영 키가 작동하지 않을 때 작성한 내용을 한글 또는 영어로 바꿔주는 프로그램

개요

1. 구상하게 된 계기
2. SRS (요구사항명세서)
 - (1) 기능적 요구사항
 - (2) 사용자 인터페이스 요구사항
 - (3) 비기능적 요구사항
3. ADS (구조설계서)
4. DDS (상세설계서)
5. 테스트

1. 구상하게 된 계기

다들 한/영키가 먹히지 않아 메모장을 켜고 한/영키를 눌러보기도 하고, 네이버에 'rnralseogkrtry'를 쳐서 '국민대학교로 검색한 결과입니다'가 나오는 것을 이용한 적이 있을 것이다. 이것은 필자의 경험인데, 1년 전에 갑자기 한/영 키가 작동하지 않았다. 일시적인 현상인 줄 알고 넘어갔는데, 재부팅을 해도 해결되지 않아 구글링을 하여 정보를 찾아보았지만 그래도 해결되지 않아서 결국 컴퓨터를 포맷하였다. 이때 고생한 기억이 남아 이 프로그램을 구상하게 되었다. 사실 '한글'이나 'word'에 이러한 기능이 구현되어 있지만 모든 곳에 적용할 수 있는 프로그램을 만들면 좋겠다는 생각을 하였다.

2. SRS

(1) 기능적 요구사항

- (1) - 1. 버튼을 누르면 무슨 작업을 할지 결정하는 상태를 바꾸도록 함
- (1) - 2. 영어에서 한글로 변환하는 경우, 변환 타이밍은 사용자가 space bar나 enter, 또는 이외의 특수문자나 숫자를 입력할 때로 함
- (1) - 3. 한글에서 영어로 변환하는 경우, 즉시 변환하도록 함

(2) 사용자 인터페이스 요구사항

- (2) - 1. 인터페이스 구성 요소는 다음과 같이 한다.
 - (2) - 1 - 1. 윈도우 타이틀 바 (프로그램 이름인 Converter로 함)
 - (2) - 1 - 2. 한/영 변환 버튼들 (현재는 한 -> 영, 영 -> 한 버튼 2개가 될 것임)
 - (2) - 1 - 3. 변환을 멈추도록 하는 버튼
 - (2) - 1 - 4. 상태 표시줄 (사용자가 임의로 수정할 수 없고, 버튼을 클릭해서 일어나는 상태 변화를 표시)

(3) 비기능적 요구사항

python으로 구현하고, PyQt5, threading, inko, keyboard 모듈을 이용한다. inko 모듈은 `python -m pip install inko-py`를 입력하면 설치할 수 있고, keyboard 모듈은 터미널에 `pip3 install keyboard`를 입력하면 설치할 수 있다. PyQt5는 GUI 환경을 제공하고, threading은 코드의 병렬 실행을 도와준다. inko는 영타를 쳤을 때 한글로 변환하는 것을 도

와주는 모듈이고, keyboard는 키보드 이벤트를 관리한다. threading을 사용하는 이유는 keyboard.wait()도 루프이고, GUI도 루프이므로 두 루프가 병렬적으로 실행될 수 있도록 하기 위함이다. 원래

(유니코드 완성형 한글코드) = 0xAC00 + 28*21*(초성 인덱스) + 28*(중성 인덱스) + (종성 인덱스)

```

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27
초 ㄱ ㅋ ㄴ ㄷ ㄹ ㄺ ㄻ ㄼ ㄽ ㄾ ㄿ ㅀ ㅁ ㅂ ㅃ ㅄ ㅅ ㅆ ㅈ ㅊ ㅋ ㆁ ㆂ ㆃ
중 ㅅ ㅈ ㅊ ㅋ ㆁ ㆂ ㆃ ㆄ ㆅ ㆆ ㆇ ㆈ ㆉ ㆊ ㆋ ㆌ ㆍ ㆎ ㆏ ㆐ ㆑ ㆒ ㆓ ㆔ ㆕
종 ㅌ ㄴ ㄷ ㄹ ㅁ ㅂ ㅃ ㅄ ㅅ ㅆ ㅈ ㅊ ㅋ ㆁ ㆂ ㆃ ㆄ ㆅ ㆆ ㆇ ㆈ ㆉ ㆊ ㆋ ㆌ ㆍ ㆎ ㆏ ㆐ ㆑ ㆒ ㆓ ㆔ ㆕

```

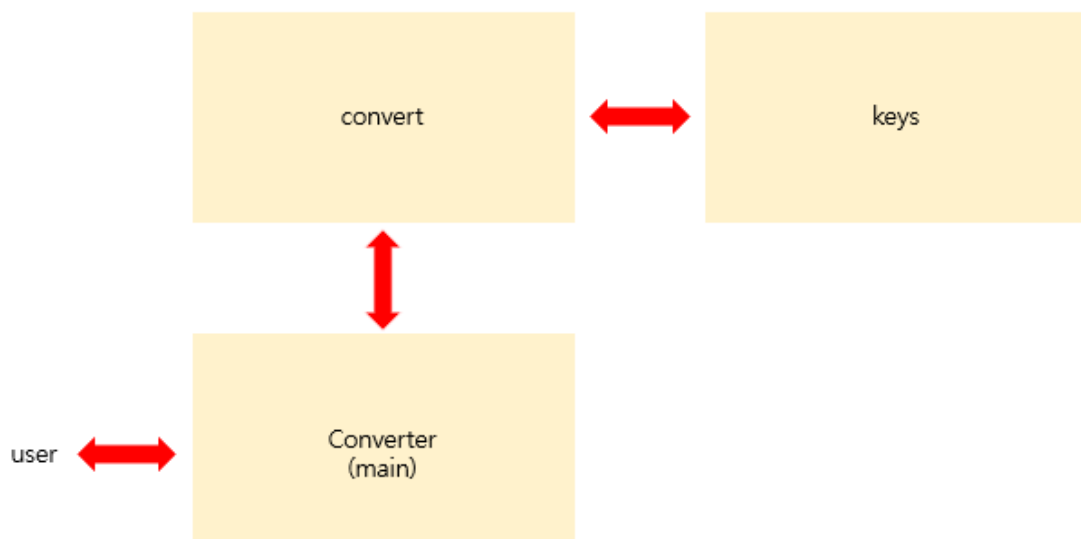
(출처 : <https://blog.naver.com/xsnake/100205421762>)

이를 이용하여 한영 변환 알고리즘을 구현하려고 노력했으나, 나중에 찾아보니 이미 inko라는 모듈이 이를 포함하고 있어서 사용했다. PyQt5에도 키보드 이벤트 처리 기능이 있지만, 이는 GUI가 클릭된 상태(윈도우 최상단에 있는 경우)에만 동작해서 부족함이 있어서 keyboard를 채용하게 되었다.

(keyboard 모듈 참고: <https://github.com/boppreh/keyboard>)

(inko 모듈 참고: <https://github.com/JackCme/inko.py>)

3. ADS



keys: 모든 영어 문자(대소문자 둘 다)가 포함된 문자열 ENG, 키보드에서 눌러도 문자열로 출력되지 않는 것들(예시: esc, f4 등)을 enter와 space를 제외하고 넣어놓은 리스트 SPECIAL을 가짐.

convert: keys의 변수들을 가져와서 키 입력을 분류하고, 한/영 변환을 하는 모듈

converter: GUI를 포함한 main driver. 유저에게 mode 변환 버튼들을 제공하고, 현재 어떤 mode인지 보여줌.

4. DDS

keys.py

| | 이름 | 역할, 설명 |
|-----------|---------|--|
| variables | ENG | 소문자 26자, 대문자 26자를 담아둔 문자열 |
| | SPECIAL | enter와 space를 제외한 esc, fl, end 등의 입력해도 문자열로 출력되지 않는 키들을 모아둔 리스트 |

convert.py

| | 이름 | 역할, 설명 |
|------------|---------|--|
| attributes | __mode | 어떤 언어에서 어떤 언어로 변환할지 결정하도록 하는 문자열 변수 |
| | __input | keyboard로부터 입력받은 값들을 저장하는 문자열 변수 |
| | __inko | 영타를 한글로 변환하는 class 초기화 |
| methods | getMode | __mode를 리턴하는 함수 (디버깅 용도) |
| | setMode | __mode를 set하는 함수 |
| | convert | 키보드 입력을 인자로 받아서 __mode에 따라 언어 변환 작업을 수행함 |

converter.py

| | 이름 | 역할, 설명 |
|-----------|----------------|---|
| class | Button | 버튼을 쓰기 쉽게 상속받아서 override |
| | Converter | 언어 변환 GUI |
| GUI 구성 요소 | 여러 Button들 | setModeAndText를 이용하여 display에 보여주는 것과 mode를 변경하는 Button들 |
| | display | 현재 mode를 보여주는 lineEdit |
| methods | keyboardStart | keyboard 입력을 받아들이는 것도 루프이고, GUI도 루프이므로 병렬 작업을 진행하기 위해 threading을 이용할 수 있도록 분리해둔 함수. hook과 wait을 함 |
| | setModeAndText | 버튼을 누를 때 연결될 함수. 상태를 보여주고 mode를 바꿈 |

5. 테스트

프로그램의 특성상 unittest를 돌리기 어려워서 직접 실행해가며 testcase를 넣어봤습니다. 먼저 convert에서 한글 -> 영어의 경우 영어가 아닌 값들(숫자, 특수문자 등)에는 반응하지 않으므로 소문자 26개, 대문자 26개를 테스트하면 끝이고, 성공했습니다. 영어 -> 한글의 경우 inko 모듈이 문제가 될 가능성은 적지만 그래도 오차가 있는 경우들(예: 안ㄴ여하세요, 없 o므 등)도 정상적으로 변환되는지 테스트했고, 성공했습니다. 영어 -> 한글 변환하는 시점이 올바른지도 테스트하였습니다. 그리고 mode가 Disabled일 때 아무 행동도 하지 않는지도 테스트하였고, 전부 성공했습니다.

converter에서 전체적으로 실행 오류가 일어나지 않는지 테스트하였고, 버튼들이 mode를 잘 바꾸는지도 테스트하여 성공했습니다.