Report

2017320202 박지수

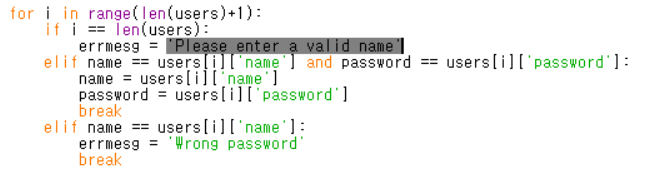
1. 프로젝트 개요 및 전반적인 자기평가

이 프로젝트는 Python을 이용하여 웹 브라우저로 접속 가능한 간단한 talk-server를 개발한 것이며, 수업 자료인 ‘Example: talk’의 기능을 일부 발전시킨 것이다. 과제에서 제시한 사용자의 등록, 참여, 대화, 관리 등의 기능을 전부 포함한다. 이외에도 사용자의 등록이나 참여 시에 발생할 수 있는 몇 가지 오류상황이 처리되었다. 또한 password와 confirm의 입력창에 내용을 입력하였을 경우 보이지 않게 처리하였다. 다만 매 실행마다 사용자가 다시 가입해야 하는 점, admin으로의 접근이 열려있어 악의적인 사용자에 의해 피해를 입을 수 있는 점 등의 약점이 있다.

1. 프로그램 실행 방법

cmd창에서 파이썬과 html 파일들이 들어있는 경로로 이동 후, ‘python app.py’를 입력하면 서버가 활성화된다. 이때 localhost:8080/으로 접속하면 서버의 메인 페이지가 보이며 주소창에 직접 path를 입력하거나 화면의 링크, 버튼의 클릭을 하여 다른 페이지로 이동하는 방식으로 실행한다.

1. 오류상황 및 처리 방식
   1. 로그인할 시에



* + 1. 존재하지 않는 이름일 경우

모든 유저의 name과 입력된 값을 비교하였으나 존재하지 않았을 경우, 에러 메시지로 ‘Please enter a valid name’을 출력하고 로그인 페이지로 다시 돌아간다.

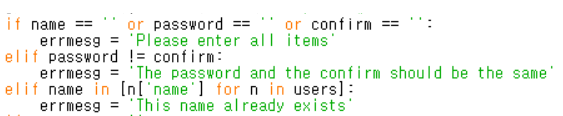
* + 1. 비밀번호가 잘못되었을 경우

일치하는 name의 유저를 찾았으나 비밀번호가 다른 경우, 에러 메시지로 ‘Wrong password’를 출력하고 로그인 페이지로 다시 돌아간다.

* + 1. 모든 항목을 입력하지 않은 경우

user 중 name이나 password가 빈 경우는 존재하지 않으므로, 따로 처리할 필요 없이 위의 ⅰ항목에서 예외 처리된다.

* 1. 회원가입 시에



* + 1. Password와 Confirm의 입력이 다른 경우

‘The password and the confirm should be the same’이라는 에러 메시지를 출력하고, 회원가입 페이지로 다시 돌아간다.

* + 1. 이미 존재하는 사용자의 이름일 경우

회원 정보를 저장해 놓은 리스트 users의 모든 user에 대하여 name이 입력한 값과 같은지 비교한다. 이미 존재하는 이름일 경우, ‘This name already exists’라는 에러 메시지를 출력하고 회원가입 페이지로 다시 돌아간다.

* + 1. 모든 항목을 입력하지 않은 경우

Name, password, confirm 세 가지 항목 중 하나라도 비어 있는 경우 ‘Please enter all items’라는 메시지를 출력하고 회원가입 페이지로 다시 돌아간다.

1. Python 소스코드 파일명: app.py, myutil.py
2. 개발한 class별 간략한 설명
   1. Class MainHandler: 로그인, 가입으로 이동이 가능한 페이지인 mail.html을 렌더링한다.
   2. Class JoinHandler: 로그인하였을 경우를 다루는 클래스이다. 예외처리를 통해서 올바른 경우 join.html로, 그렇지 않은 경우에는 join-fail.html로 연결된다.
   3. Class TalkHandler: 대화상황을 다루는 클래스이다. 메시지를 보낼 때마다 그 정보가 history에 저장된다.
   4. Class LogoutHandler: 로그아웃할 경우를 다루는 클래스이다. 히스토리에 로그아웃했다는 사실이 추가되면서 다른 사용자와 admin이 이를 알 수 있다.
   5. Class RegisterHandler: 회원가입창을 보여주는 클래스이다.
   6. Class ConfirmHandler: user가 회원가입을 하려고 시도했을 때를 다루는 클래스이다. 예외 처리를 통해서 올바른 경우에만 reg-confirm.html로 보내고 사용자로 등록시킨다. 올바르지 않은 경우에는 reg-fail.html로 보내서 각 상황에 맞는 에러 메시지를 보여준다.
   7. Class AdminHandler: 전체 대화 내용을 담은 히스토리를 dump 또는 load할 수 있는 클래스이다. 주소창에 직접 ‘/admin’을 쳐서 이동했을 때만 사용된다.
3. Html 파일명 및 사용된 method 이름
   1. Admin.html: action='/admin' method='post'
   2. Join.html: action='/logout' method='post', action='/talk' method='post'
   3. Join-fail.html: action='/main' method='get'
   4. Logout.html: action="/main" method="get"
   5. Main.html: action="/join" method="post", action="/register" method="get"
   6. Reg-confirm.html: action='/main' method='get'
   7. Reg-fail.html: action='/register' method='get'
   8. Register.html: action='/reg-confirm' method='post'
   9. Talk.html: action='/talk' method='post', action='/logout' method='post'