

Web 前端技术实训课程——基础实验二

田正祺 软件 01 2020080095 tianzq20@mails.tsinghua.edu.cn

1 使用说明

在 `src/` 目录下运行：

```
pip install -r requirements.txt
py app.py
```

并在浏览器中访问 Flask 所显示的 URL（默认 <http://127.0.0.1:5000/>）。唯一的用户名为 `tianzq20`，密码为 `password`。

2 代码结构

此文档中以下的路径都默认添加 `src/` 前缀。

本项目的 HTML、CSS、Javascript、以及图片文件分别存放在 `templates`、`static/css`、`static/js`、以及 `static/img`。实现后端的 Flask 程序存放在 `app.py` 中。

3 实现

3.1 内容

此项目中有两个网页：“登录”页面，即让用户输入用户名以及密码，以及“成功”页面，即用户成功地登录后可以看到自己的连接时常以及已用流量。它们分别存放在 `templates/login.html` 和 `templates/success.html` 中。后者与实验一相比几乎相同。

3.2 样式

用于为 `login.html` 以及 `success.html` 添加样式的 CSS 文件分别存放在 `static/css/login.css` 以及 `static/css/succeess.css` 中。

3.3 行为

后端是此项目的重点之一，而它的实现在 `app.py` 中使用了 Flask 框架。本项目有两个接口，其 URL 路径分别在 `/login`（或 `/`）以及 `/success`，也有两个状态：连接状态 `connected` 以及已用流量 `usage`。在未连接状态下，用户访问 `/success` 后将会被重定向到 `/login`。`/login` 在接收 GET 请求后就会反应登录页面的 HTML 代码。在接收 POST 请求，即用户名以及密码，就会根据正确性而返回对应的响应。POST 请求的发送是由 `/static/js/login.js` 中的 `connect()` 函数实现的。此函数也负责实现空用户名或空密码的异常处理，以及将用

用户名和密码以 JSON 格式发送给后端。登录成功后重定向到 `/success` 也是 `login.js` 中的 `connect()` 函数实现的。

类似的，在已连接状态下，用户访问 `/login` 后将会被重定向到 `/success`。`/success` 在接收 GET 请求后就会反应成功页面的 HTML 代码。在接收 POST 请求，即用户点击“断开链接”，就会重定向到 `/login`，并将已用流量增加。

4 问题与解决方法

一个我遇到的问题是：已用流量不仅需要通过文字在成功页面上显示，而且也需要改变此页面上的橘色进度条的长度。Flask 所使用的模板可以让后端动态地改变 HTML 中的内容，但较难改变样式，因此无法改变进度条的长度。一个方式是让后端将已用流量传输给 Javascript，并让 Javascript 改变文字以及进度条。比如：后端实现另外一个接口，只返回已用流量。Javascript 就可以发出请求，并从回应中得到已用流量。但是会到一定的程度增加代码量。此项目的解决方法是将已用流量的文字通过模板放在 HTML 中（使用 `usage` 变量），并将 Javascript 读取 HTML 文件中的元素，从而的去已用流量的数值，最后可以改变进度条的长度。