

web期中作业文档

web期中作业文档

概述

前端

后端

部署

基本功能展示

功能1：商品展示

功能2：用户注册登录

功能3：添加购物车

功能4：登录状态结算下单

功能5：登录状态查询历史订单信息

概述

本项目实现了一个简单的 服装购买 网站，采用了前后端分离的开发方式。

前端使用的是 `AJAX` 原始框架，后端使用的是 `flask` 框架。

前端

前端分了 `html`, `js`, `css`, `img` 四个文件夹。

其中 `css` 存放的是一些样式文件，`img` 中存放的是页面渲染需要的一些图片文件，`html`, `js` 中各文件的内容及作用如下表所示：

1. `html`

文件名	内容
index.html	主页面。展示了网站所有待售的服装及其基本信息。
product.html	单个商品的展示页。展示了该服装更详细的信息，并可以将其加入购物车。
cart.html	购物车展示页。展示了已经加入购物车的商品，并可以进行结算。（需登录）
profile.html	注册和登录页面。
order.html	历史订单展示页。展示了该用户的历史购买记录。（需登录）

2. js

文件名	作用
indexApi.js	获取主页面需要的数据
render.js	存有渲染主页面的函数、渲染单个商品展示页的函数和渲染购物车页面的函数
productDetail.js	获取单个页面需要的数据，并执行对该页面的一些操作（如，加入购物车等）
cart.js	执行对该页面的一些操作（如，删除商品、计算总价等）
tapPay.js	执行结算订单的操作
profile.js	执行注册和登录的操作
order.js	获取用户信息，展示用户的历史订单，并可以进行退出登录的操作
localStorage.js	存有localStorage信息

后端

后端用 `flask` 简单的实现了一下，具体接口及其作用如下所示：

接口	作用	输入	输出
/main	获得所有商品信息	无	所有商品信息
/product	根据商品id，获取单个商品的具体信息	商品id	单个商品的具体信息
/post	存入用户的历史订单	用户id，用户订单	无
/register	用户注册	用户名、密码	无
/login	用户登录	用户名、密码	是否存在该用户
/get	根据用户id，获取该用户的历史订单	用户id	该用户的历史订单

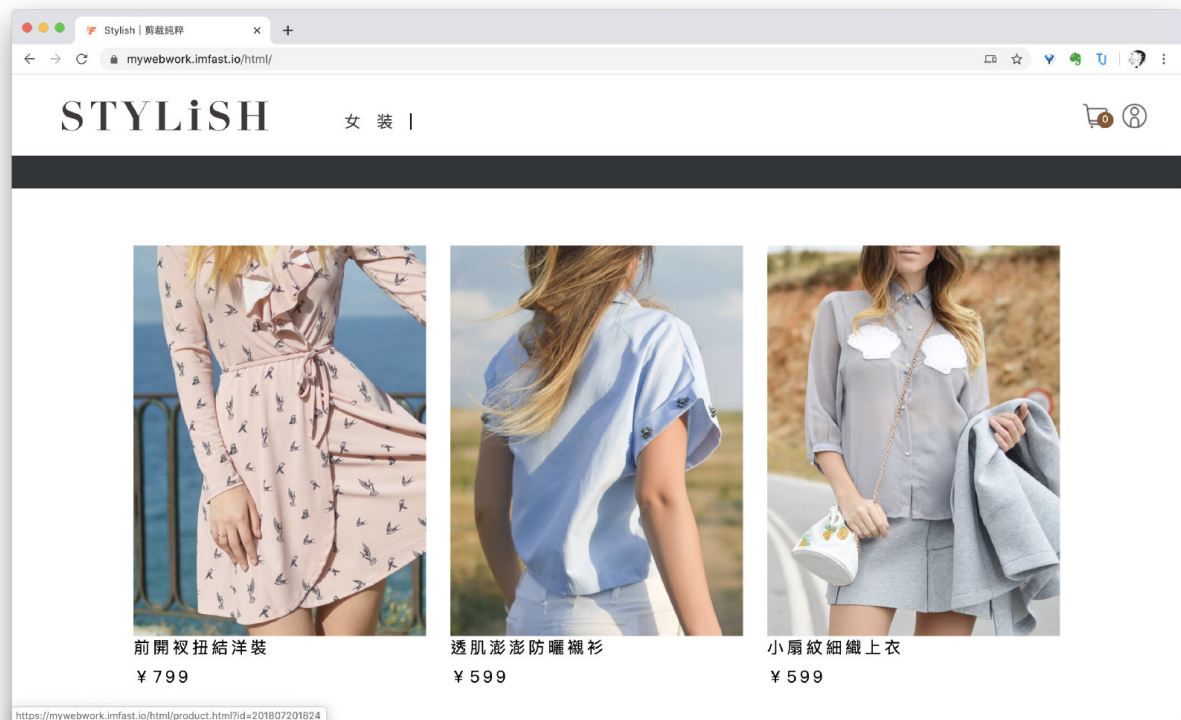
部署

因为服务器不是很稳定，所以没有用 `nginx` 进行部署。

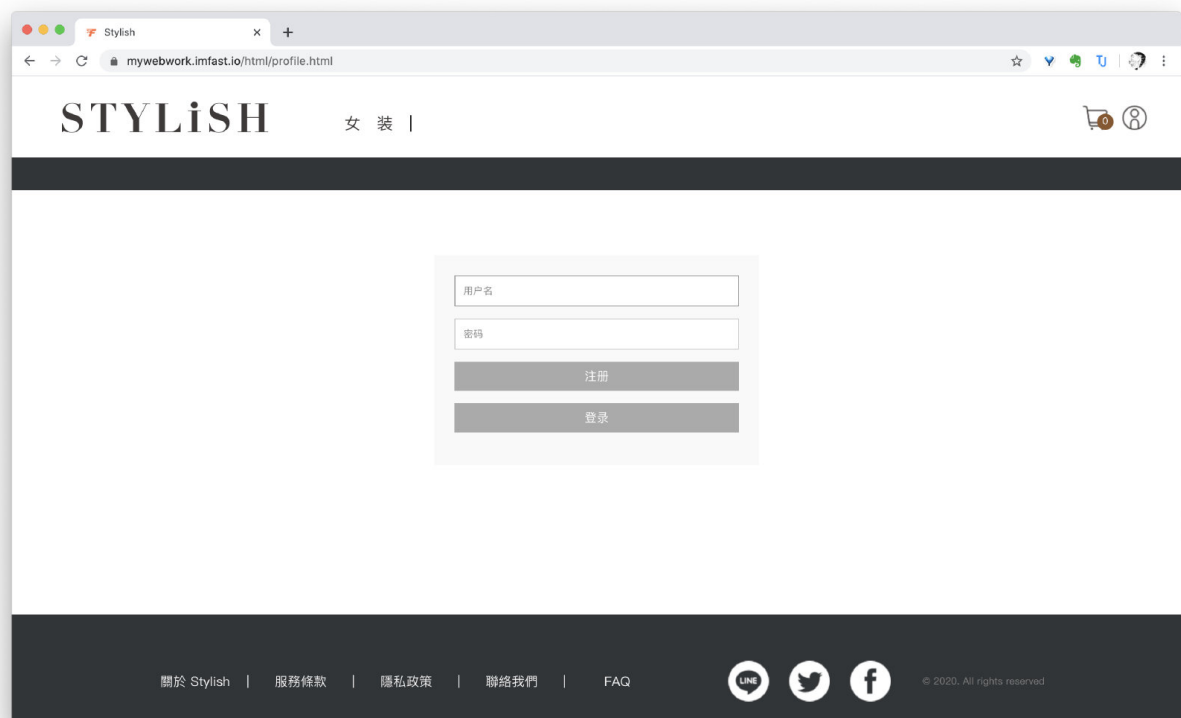
这次用的是 `GitHub page` 和 `fastIO` 进行部署。

基本功能展示

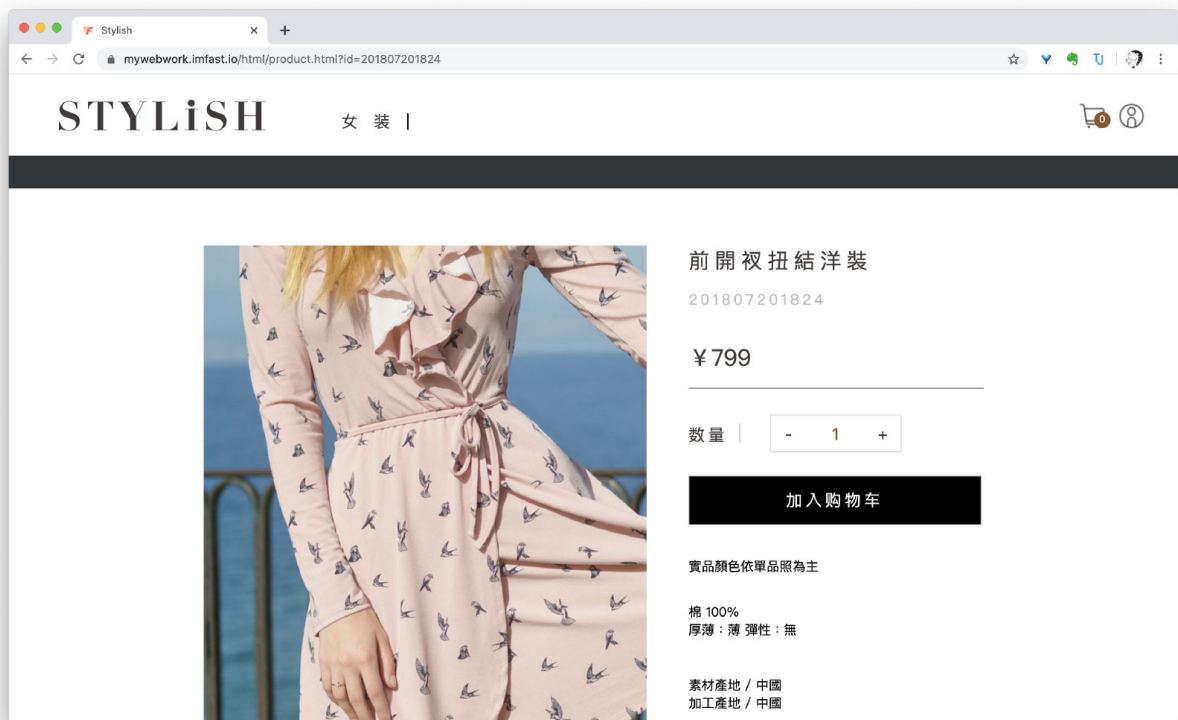
功能1：商品展示



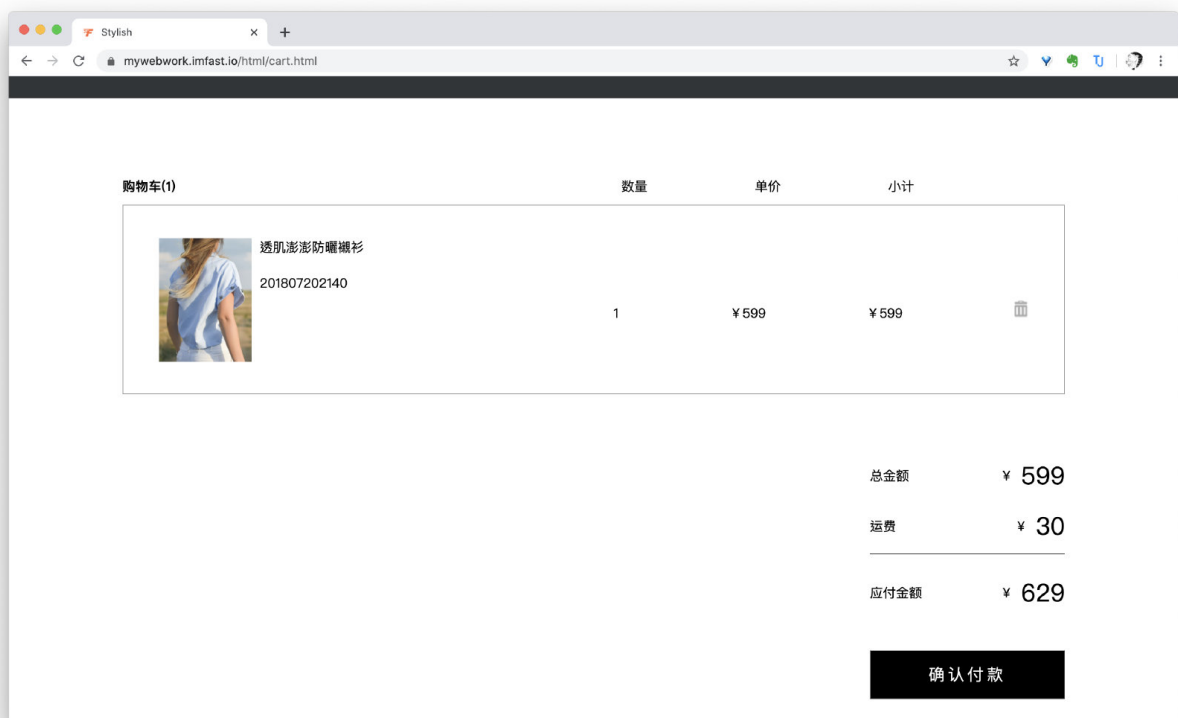
功能2：用户注册登录



功能3：添加购物车



功能4：登录状态结算下单



功能5：登录状态查询历史订单信息

