### Virtual-Try-On Dev Spec

### 一、概述/概念

任务概述：本软件是一款独立的集成式的可视化的试穿衣服软件。可以通过商家上传模特试穿样品图，用户上传自己的身材图，再云端生成出用户试穿的图片，给予用户换装的新体验。便于用户选择最适合自己的服装商品 。本软件的功能列表如下。

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **功能** | **归类** | **优先级** |
| 1 | 上传全身照或者调用本地摄像头，拍摄用户全身照 | 基本型 | 1 |
| 2 | 商家可以上传商品衣物图片 | 基本型 | 1 |
| 3 | 模型生成大量的试穿照，供用户选择 | 基本型 | 1 |
| 4 | 用户可以进行衣服的款式的搜索 | 基本型 | 1 |
| 5 | 生成试穿照的保存、收藏和分享功能 | 无差别 | 2 |
| 6 | 用户的登录和历史记录管理 | 无差别 | 2 |
| 6 | 用户评论功能 | 无差别 | 2 |
| 7 | 对模型进行优化，使得生成图片质量更高 | 无差别 | 2 |
| 8 | 进行模型压缩，使得模型可以在移动端本地运行 | 期望型 | 3 |
| 9 | 对模型进行优化，支持高分辨率的图片生成 | 期望型 | 3 |
| 10 | 自由选择身体姿态进行试穿匹配 | 期望型 | 3 |
| 11 | 对服装试穿结果和购买数据进行统计分析 | 期望型 | 3 |
| 12 | 对用户进行个性化推荐功能 | 惊喜型 | 4 |

表1

### 二、运行环境/总体设计

**运行环境**

* 软件运行环境：全平台 web 端；
  + 本软件基于web的环境进行开发，用户可以在web中进行登录，并上传自己的图片。图片会上载到云端服务器，利用云端服务器中存储的模型对其进行试穿照片的生成；
  + 目前初步确定使用Azure 部署我们的模型和应用。
* 硬件运行环境：独立计算机、手机。

**总体设计**



* 前端：
  + 向后端请求页面与其他资源；
  + 获取试穿照片，提供试穿预览功能；
  + 为用户与商家提供友好的交互。
* 后端：
  + 根据请求向前端返回对应的资源；
  + 和数据库进行交互存储相应的用户、商家、商品信息；
  + 将用户全身照和商品图片发送给云端模型合成虚拟试穿照，并将结果返回给前端。
* 云端：
  + 存储、运行模型；
  + 接收后端传入的服装照片和用户全身照片以生成虚拟试穿照，将结果返回给后端。

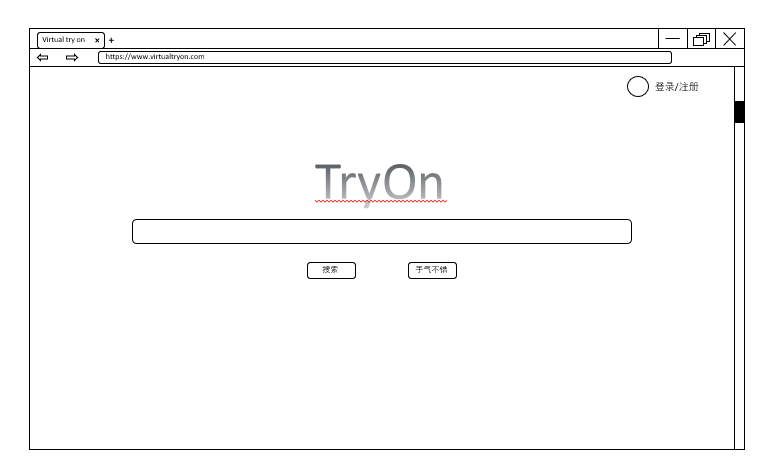
**数据设计**

* 因只需要使用原模型中 pose-to-image 部分的功能，所以只需要使用 DeepFashion 数据集进行训练即可；
* 使用开源工具 OpenPose 估计 DeepFashion 中不同人的pose信息；
* 根据官方提供的划分策略，将数据集划分为训练集和验证集。

### 三、交互/接口/功能设计

（一）用户的接口

**首页**



* 首页为用户提供**搜索**功能，用户可以搜索自己想要虚拟试穿的衣服类型及品牌，例如：秋装，长裤等等；
* 为了增加用户体验，增加了**手气不错**选项，可以随机（或根据用户的浏览记录进行个性化推荐）从数据库中为用户选择一件衣物进行虚拟试穿；
* 右上角提供了用户**登录和注册**的选项，用户可以通过手机号等进行注册，注册账号即可进行登录。

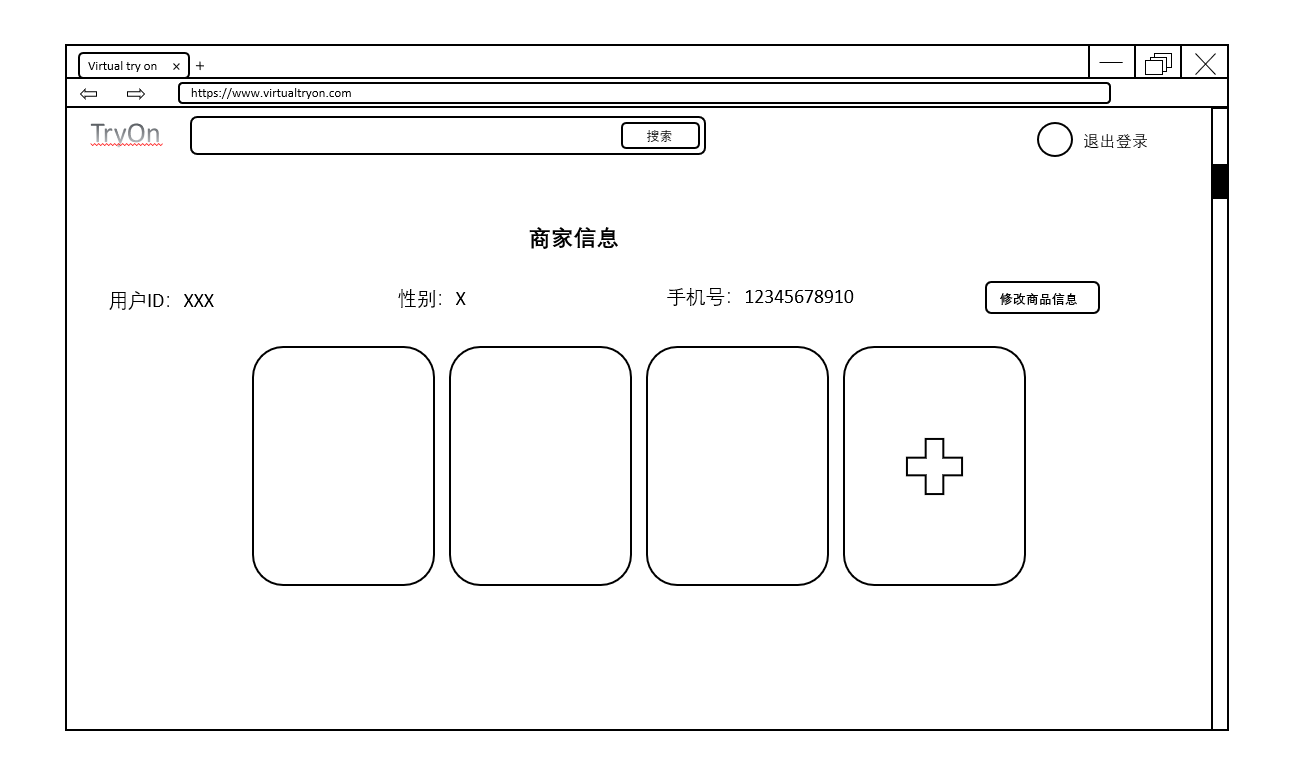
**用户个人主页**



* 登录之后的用户界面中，用户可以提前**上传**自己的全身照，系统会提前将全身照上传至云端，可以更快速地得到虚拟试穿的结果，并提供后文所属的**提前预览**功能；
* 用户可以随时**修改个人信息**，如手机号、性别以及全身照等；
* 如果用户不愿意注册，则开启默认的游客模式，此时将不提供提前预览的功能，只能在商品界面上传自己的全身照进行虚拟试穿。

（二）商家的接口

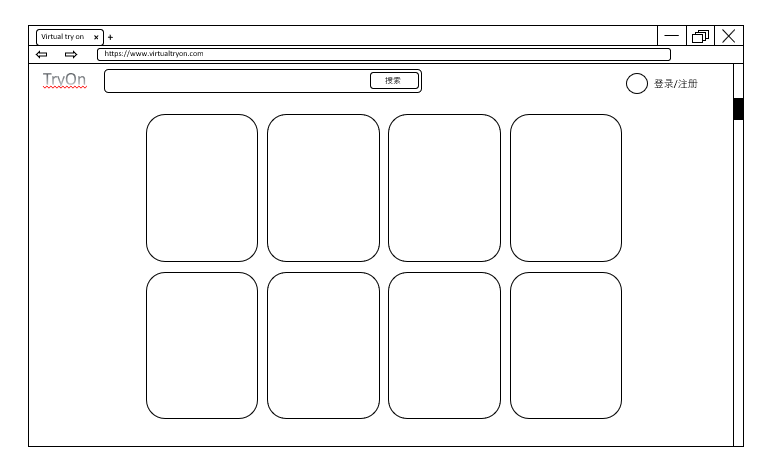
**商家个人主页**



* 登录之后的商家界面中，商家可以上传新的商品、删除已上传的商品或修改商品信息；
* 登录后，商家也可以自行更改手机号等个人信息。

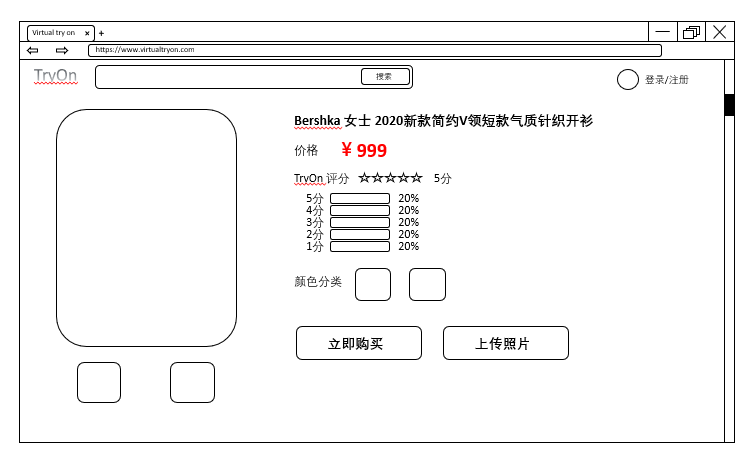
（三）日常使用

**商品浏览页**



* 采用数据流的呈现方式，这样可以最大程度的提供更好的用户体验，也和主流的购物软件保持一致；
* **提前预览**功能：如果用户是登录状态并上传过自己的全身照，那么鼠标悬浮在图片上，就将看得到生成的虚拟试穿图片。

**商品详情页**



* 用户可以上传本人照片，并通过选择服装的颜色类型，来生成对应颜色服饰的虚拟试穿的图片；
* 用户可以为服装和试穿效果**评分**，同时可以预览该服饰的先前评分；
* 点击立即购买会将用户**导流**到第三方购物页；
* 商家最多为每个商品提供的5张商品图片；
* 可以通过图片下方的两张略缩图，选择大图中展示的是原本模特试穿的图片还是虚拟试穿的效果。

### 四、数据库设计

**实体及其属性：**

1. 商品：
   1. 商品ID：主键，后端自动生成，用来唯一标识一个商品；
   2. 商品名称 ：字符串属性，不能为空；
   3. 商品图片 ：图片属性，不能为空；
   4. 商品价格 ：浮点数属性，不能为空；
   5. 购物链接 ：字符串属性，不能为空；
   6. 评分总人数：整数属性，默认为0；
   7. 商品总分：整数属性，用来和评分总人数一起计算商品平均分，默认为0；
   8. 所属商家：外键，参照商家用户名。
2. 用户：
   1. 用户名 ：主键，由用户指定，不可重复，不能为空；
   2. 密码 ：字符串属性，由用户指定，不能为空；
   3. 手机号 ：数字属性，由用户指定，不能为空；
   4. 个人图片 ：图片属性，可以为空，由用户自己上传。
3. 商家：
   1. 用户名：主键，由商家指定，但不可重复，不能为空；
   2. 密码：字符串属性，不能为空；
   3. 手机号：数字属性，由商家指定，不能为空；
   4. 已上传商品：外键，参照商品 ID，级联删除。

### Training/Inference 训练/推理子系统设计

（一）训练子系统：

* 硬件
  + GPU：8 × Nvidia V100s
  + 单机/分布式：单机多卡
* 软件/框架
  + 模型基于PyTorch，也使用了一些第三方库
* 算法/基础模型
  + 使用了CoCosNet的网络模型，在其基础上进行优化和提升
* 指标要求
  + 训练模型所需要的时间：10天之内
  + 模型的参数尺寸：500MB以内

（二）推理子系统

* 硬件
  + GPU：1 × Nvidia V100s
  + 单机/分布式：单机单卡
* 软件/框架
  + 模型基于PyTorch，也使用了一些第三方库
* 指标要求
  + 单次测试模型所需要的时间：100ms级

### 六、日志/试验

**日志**

* 记录用户的生成请求；
* 记录用户的登录请求；
* 记录用户的打分请求。

**试验**

* 基本功能实现完成后提供给部分用户进行测试。

### 退出机制/已知问题

遇到何种情况时，即停止开发工作、放弃项目：

* 无

已知问题：

* 模型无法生成较高分辨率的虚拟试穿图片；
* 模型生成的试穿图片中衣服和身体可能组合地不够理想；
* 可能生成图片并回传的延迟较高；
* 网站服务器对于用户流量的限制上限。

### 工作计划

|  |  |  |
| --- | --- | --- |
| **人员** | **工作** | **初步计划** |
| 毕书显 | 模型优化和部署 | 5周 |
| 王楚鑫 | 模型优化和部署 | 5周 |
| 张博文 | 后端开发 | 5周 |
| 詹佑翊 | 前端开发 | 5周 |