

# 软件附加功能说明文档

## 一、前言

- 软件简介：电影可视化分析系统 Cuckoo，具有爬虫、账号管理、可视化数据等功能的 web 系统。
- 文档目的：该 project 用途之一是《软件工程》实验课的 project 提交，除主要功能外，该 project 还实现了一些附加功能，便于 project 的验收，特此说明附加功能的详细内容及实现。

## 二、组织架构

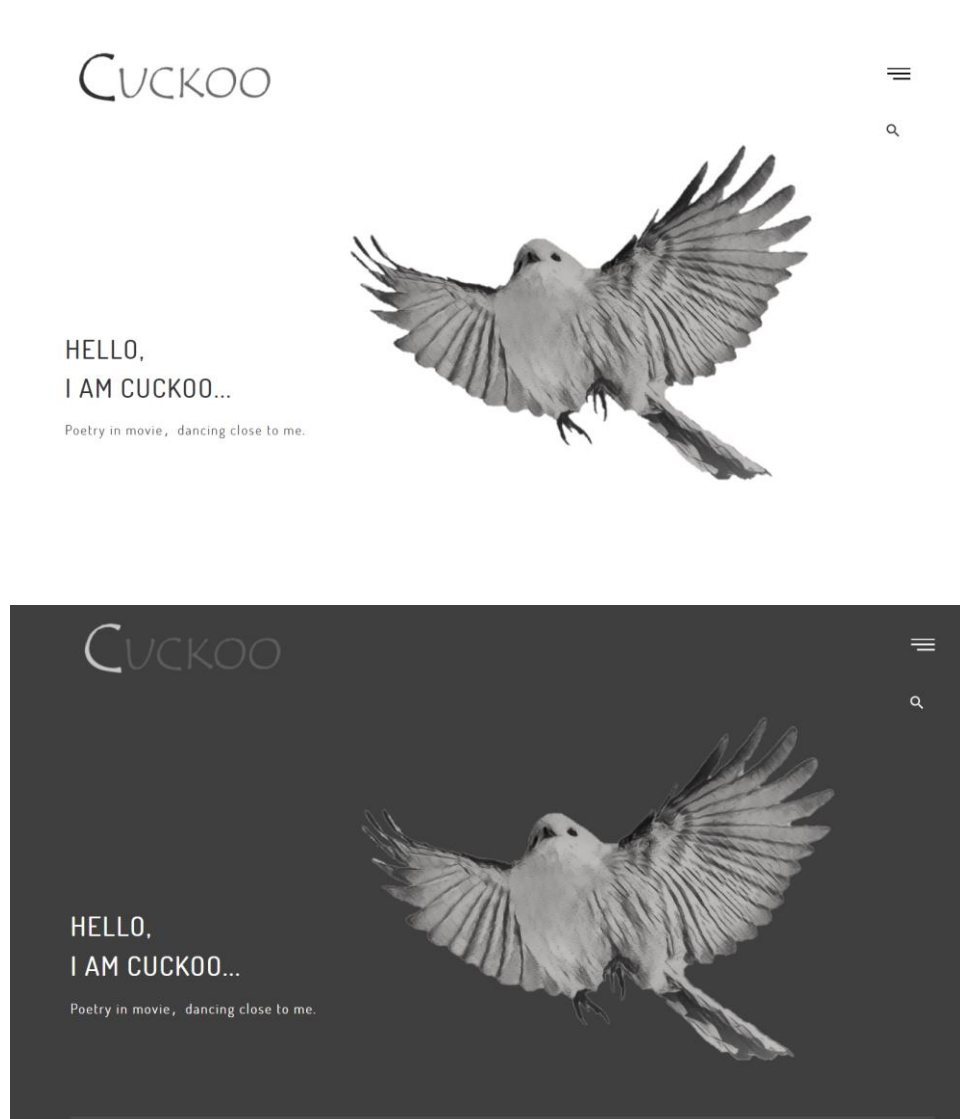
文档将对各个功能模块进行功能展示、实现原理、部分核心源码解释等说明。

## 三、功能介绍

### (一) 界面“护眼模式”

软件界面的单一性会减少用户对软件的好感度，适当增加页面与用户间的交互，可以使用户体验得以提升，除此之外，Cuckoo 是一个提供数据分析及可视化的界面，其最初的核心目的之一便是解决用户对大量繁杂数据的而不知如何整理的问题，使界面更人性化是我们在产品开发时的着重点。“护眼模式”不仅增加了用户与软件间的交互，更增加了软件的人性化，增加用户的体验感。

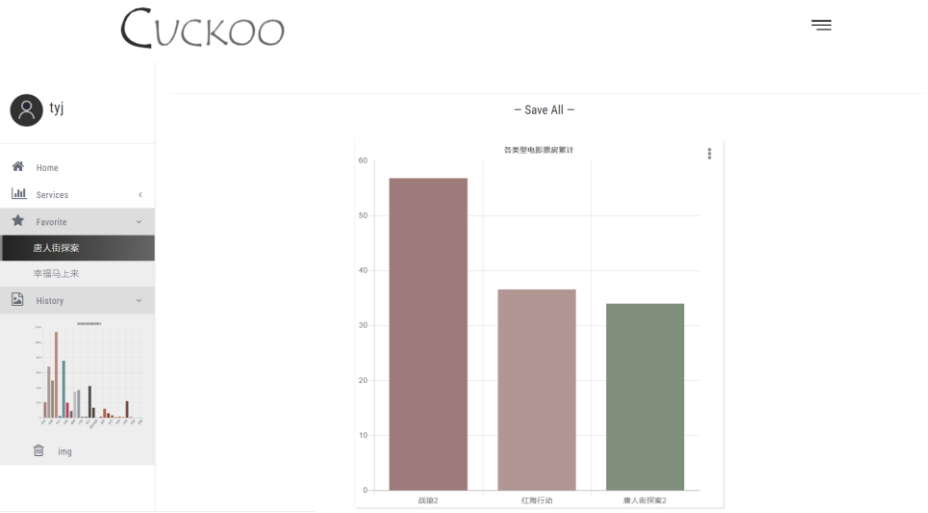
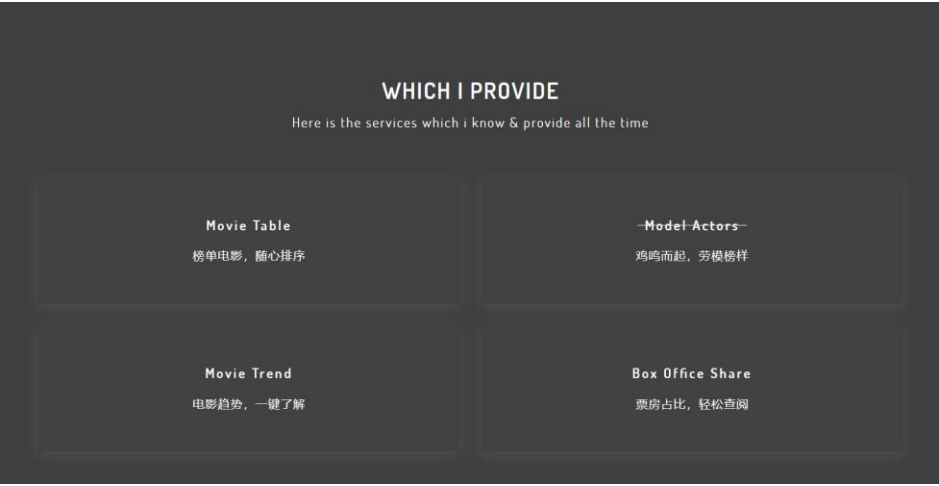
➤ 界面展示

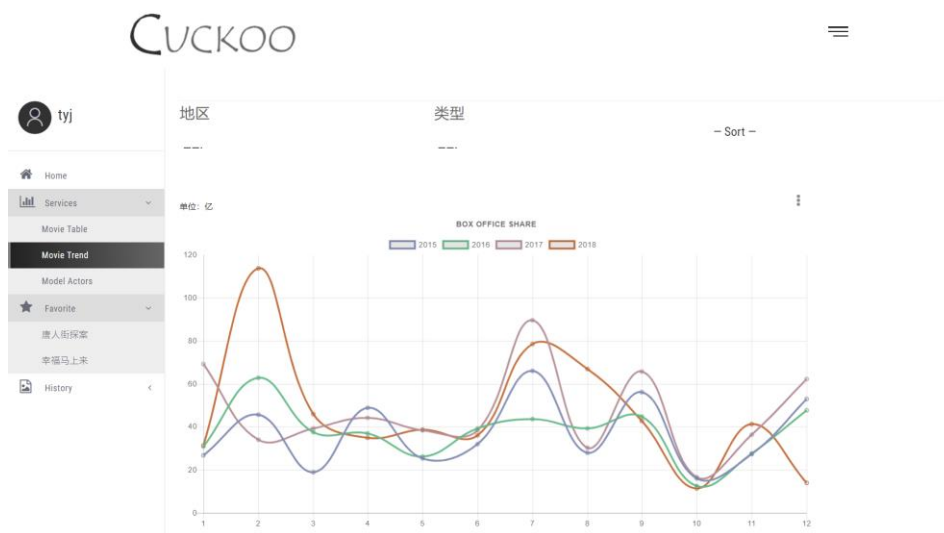


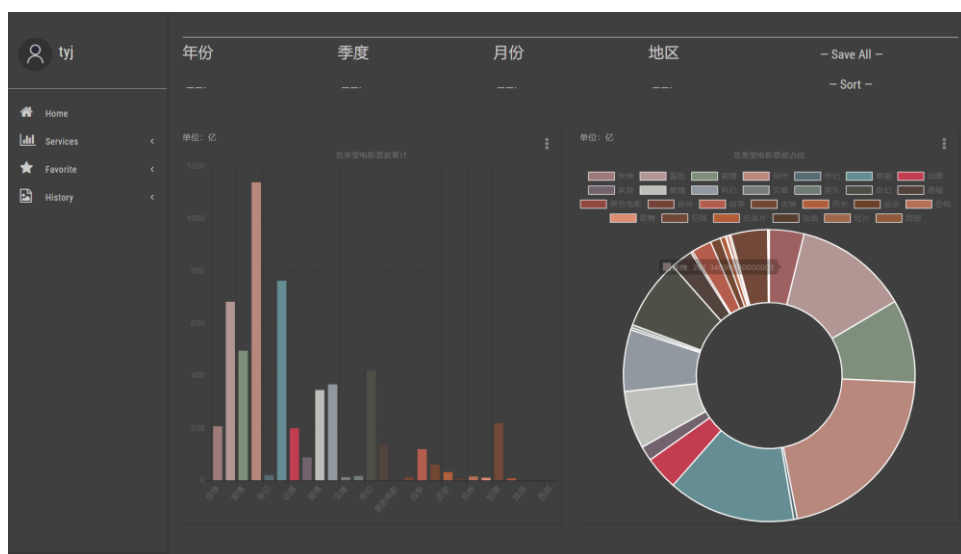
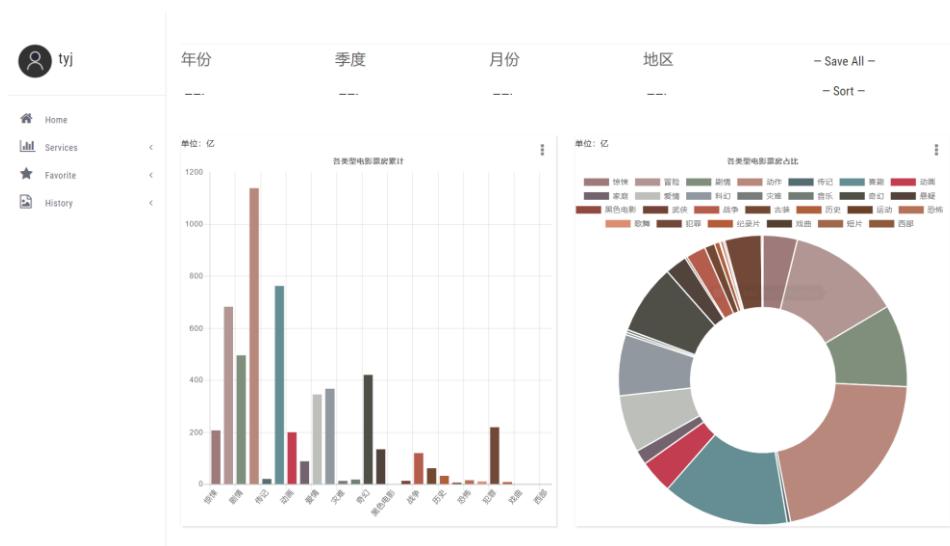
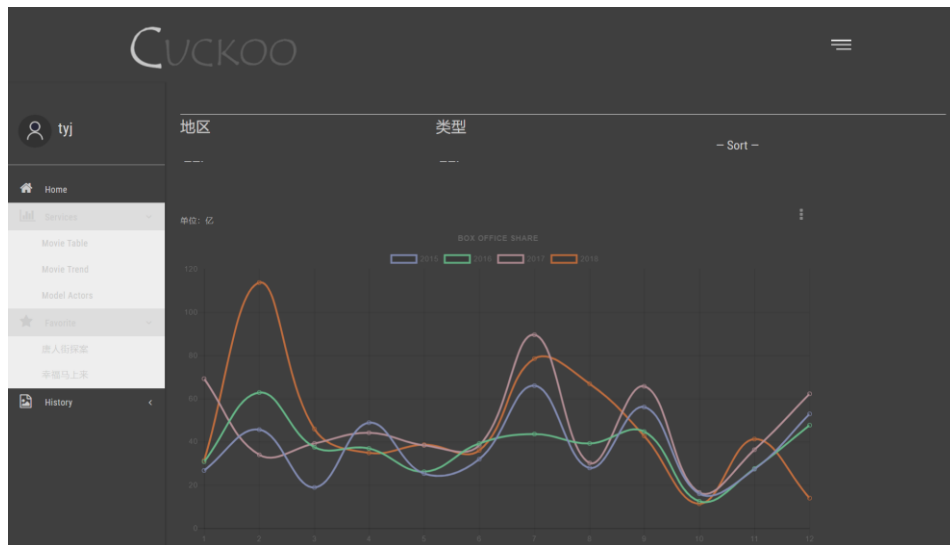
## WHICH I PROVIDE

Here is the services which i know & provide all the time

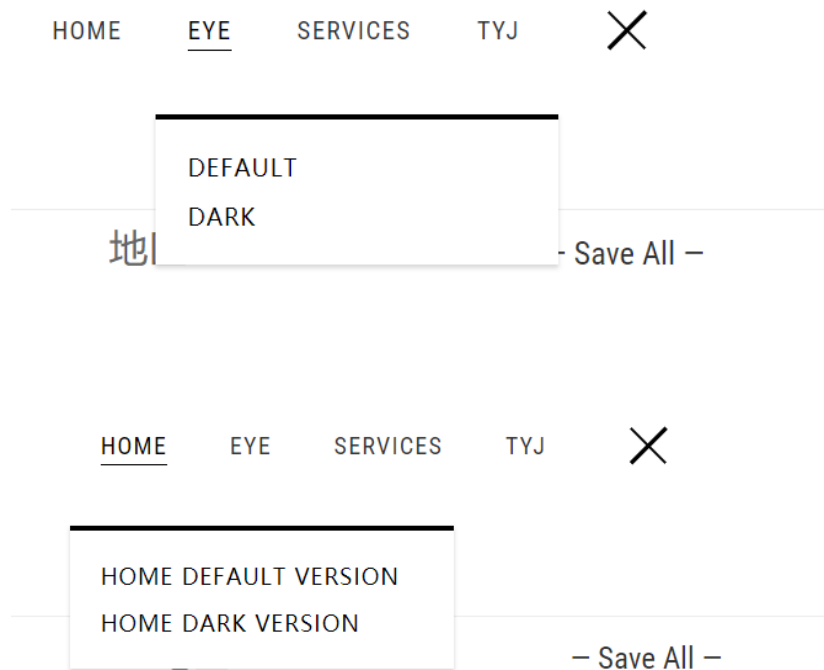
<b>Movie Table</b> 榜单电影，随心排序	<b>Model Actors</b> 鸡鸣而起，劳模榜样
<b>Movie Trend</b> 电影趋势，一键了解	<b>Box Office Share</b> 票房占比，轻松查阅





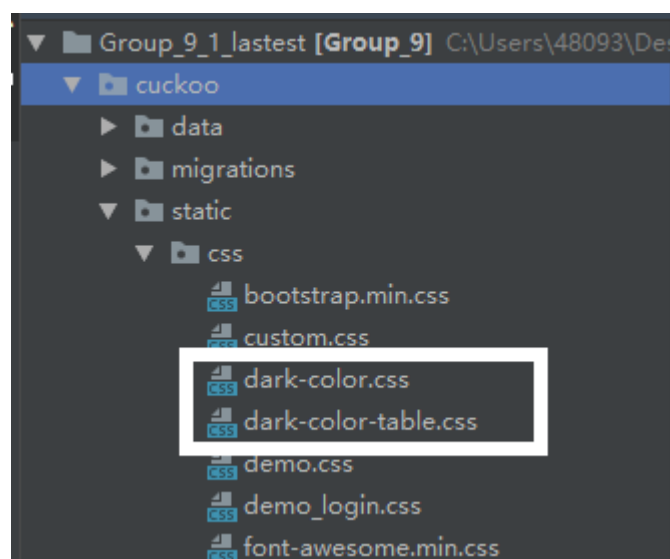


- 使用方法：点击顶部导航栏 EYE 进行“default”“dark”进行切换，或是 HOME 进行不同主页版本的选择。



- 实现原理

通过前端向后台传递参数选择是否调用黑夜模块的 CSS 格式文件对页面进行重新渲染。

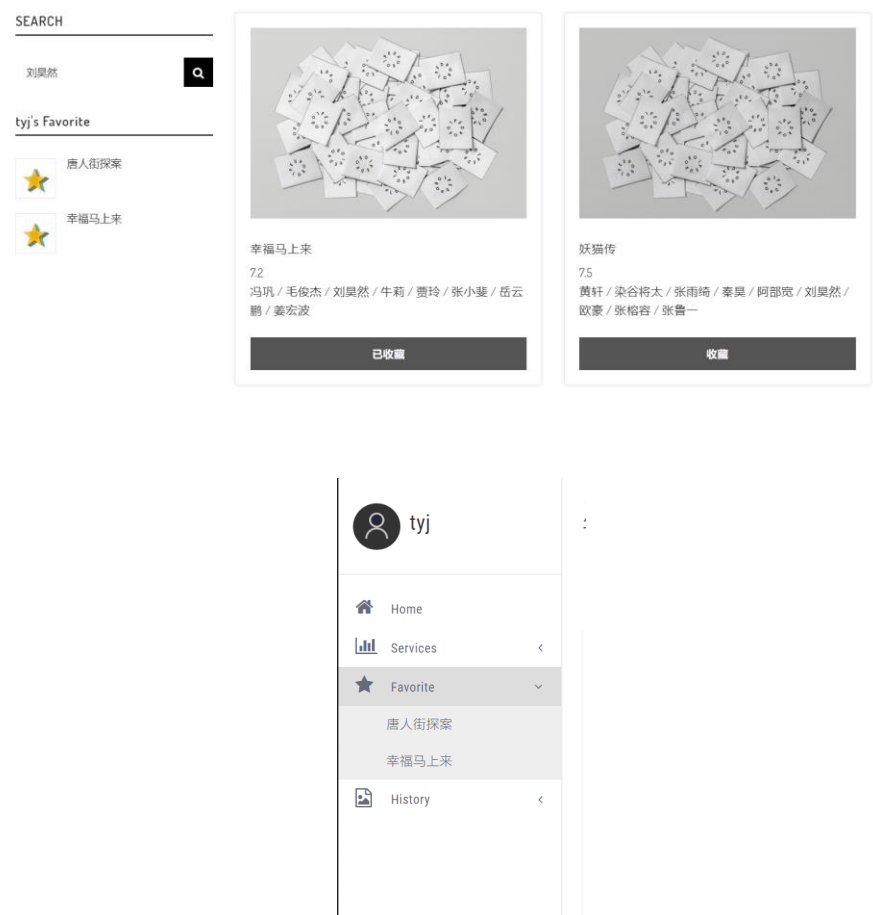


(二) 收藏夹功能

软件可玩性是增加用户粘度的重要因素之一，收藏夹作为用户个性化功能之一，可以增加用户使用过程中的可玩性。同时，增加收藏夹功能使得用户可以主动保存当前浏览记录，方便用户之后得使用，从而让用户产生对软件的依赖感，增加用户粘度。

➤ 界面展示

我们在每一个功能的侧边栏都提供了收藏夹，便于用户使用该功能。用户点击收藏按钮即可对其收藏，按钮变成“已收藏”状态，再按下即可取消收藏，按钮回归“未收藏”状态。



➤ 实现原理

通过构建与用户 model 所关联的收藏夹 model 数据库，使其搜藏功能对注册用户开放。前端点击收藏按钮，获取当前按钮所属条目信息，并

向后端发出一个 ajax 请求，请求后端进行处理，后端收到请求，更新数据库并返回条目状态，前端显示条目状态。

➤ 实现源码：

```
@csrf_exempt
def deal_favorite(request):
    print('ajax chenggong')
    username = request.user
    if username:
        login_message = 'yes'
        user = User.objects.get(username=username)

        movie_name = request.POST['movie_name']
        movie_url = request.POST['url']
        print(movie_name)
        # print(user.favorite_set.all())
        # 从收藏夹中搜索
        the_movie = user.favorite_set.filter(movie_name=movie_name)
        if the_movie: # 如果已经收藏了，就取消收藏
            print(the_movie)
            the_movie.delete()
            message = '取消收藏成功'
        else:
            the_new_movie = user.favorite_set.create(movie_name=movie_name, movie_url=movie_url)
            the_new_movie.save()
            message = '收藏成功'
    else:
        login_message = '您尚未登录'
        message = '收藏失败'
    return JsonResponse({'message': message, 'login_message': login_message})
```

### (三) 模糊搜索功能

对于包含着大量数据的软件来说，增加搜索功能给用户体验上带来的增益是可观的。再者，能否为用户筛选出有效信息是该软件在开发过程中重要考量的因素之一，除了该软件提供的格式化筛选，用户的主动筛选更能使用户获得更高的体验。同时搜索功能的提出，是采用何种搜索方式是值得我们的考量的，数据属性太多，若每次搜索时都让用户选择搜索的属性，势必会影响用户的体验，所以我们采用了模糊搜索的方式，为用户减去大量的选择时间，把时间交给我们。

同时，我们在模糊搜索上的非功能需求进行了考虑，比如搜索速度，对于速度来说，由于数据量及算法的优异性，反应速度在 1s 内。对于涵盖条目较多的搜索项，也能快速出结果（但在观感上，第一次搜索仍会持续 2~3s，该数据是基于开发者电脑配置所有所变化，第二次搜索能加速这一过程）



➤ 功能入口



➤ 界面展示

