成绩：

南京理工大学

《管理信息系统》

课程作业

|  |  |
| --- | --- |
| 上课学期 | 2023年秋季 |
| 小组成员 | 921127940137，姓名：沈贝宁 |
| 921127940122，姓名：顾翔 |
| 921127940125，姓名：季诚 |
| 921127940133，姓名：那铭心 |
| 921127940156，姓名：张瀚宸 |
| 921127940158，姓名：张浩 |
| 921127940161，姓名：张俊杰 |
| 921127940159，姓名：张浩然 |

2023年 11 月 29 日

**基于智慧信息的高校自助打印管理系统**

**报告撰写的方式**：

√报告内容有AI辅助撰写：商业模式设计章节使用了ChatGPT辅助撰写

**小组成员分工**：

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 学号 | 负责内容 | |
| 沈贝宁 | 921127940137 | **前端设计及代码编写（客户页面，店员打印页面）、系统部署（前端服务器）、书面报告撰写（第五章）** | |
| 顾翔 | 921127940122 | **前端设计及代码编写（注册登录、店员页面）、系统部署（SSL证书, PWA）、书面报告撰写（第五章）** | |
| 季诚 | 921127940125 | **系统调研及整体功能设计、智慧打印店商业模式分析、报告撰写（第一至四章）** | |
| 张瀚宸 | 921127940156 | **后端设计及代码编写（后端服务器，数据库）、系统部署（后端服务器）、书面报告撰写（第五章）** | |
| 那铭心 | 921127940133 | **PPT制作** | **幻灯片的制作是由：小组成员人工完成** |
| 张浩 | 921127940158 | **PPT制作** |
| 张俊杰 | 921127940161 | **PPT制作** |
| 张浩然 | 921127940159 | **PPT制作** |

**目录**

[1. 引言 1](#_Toc152248885)

[2. 系统设计动机 1](#_Toc152248886)

[2.1 背景介绍 1](#_Toc152248887)

[2.2 用户需求 2](#_Toc152248888)

[2.3.当前的信息系统 3](#_Toc152248889)

[3. 商业模式设计 4](#_Toc152248890)

[4. 系统的功能设计 7](#_Toc152248891)

[4.1 系统实现技术：渐进式Web应用（PWA） 8](#_Toc152248892)

[4.2 系统主要功能 8](#_Toc152248893)

[4.3 系统特性：清晰的反馈机制 17](#_Toc152248894)

[5. 系统开发过程介绍 18](#_Toc152248895)

[步骤1：确定系统开发环境和运作模式 18](#_Toc152248896)

[步骤2：前端开发 18](#_Toc152248897)

[步骤3：后端开发 28](#_Toc152248898)

[步骤4：系统部署 39](#_Toc152248899)

[参考文献 40](#_Toc152248900)

1. 引言

在高校环境中，传统的打印店经营管理模式较为落后，店铺的管理多依赖于店铺的直接管理者。在如今信息化、数字化的背景下，这种店铺管理模式显然落后于时代发展，不仅不能够满足当前学生和教职工对打印和文档处理的多样性需求，而且不利于校内打印店的长久经营管理[1]。因此，本项目计划建立一套高校内智慧打印店管理信息系统，以解决当前高校内打印店存在的问题。

本项目系统面向的是高校内学生、教职工等消费用户以及校内打印店的管理者。项目为了逐步构建智慧打印店管理信息系统，首先针对上述用户的需求进行了全面分析。从消费者和管理者两种角度，分别识别了消费者在打印店进行消费的多种需求，诸如自助打印、预约打印等，以及打印店管理者在不同管理环境中的不同需求，包括店铺财务管理、资源管理、员工管理。通过与当前存在的打印店管理信息系统进行比较分析，了解到当前管理信息系统的存在的诸多缺陷。

为了尽量解决当前管理信息系统存在的问题，本项目从商业画布9方面对打印店管理进行了店铺商业模式设计，明确了本系统面向用户、价值主张等要素。根据用户需求对系统进行了多功能设计。因此，本项目的系统致力于在实现打印店的基本功能上，满足不同消费群体的多样性需求，实现打印店管理者对店铺的全方面管理，以弥补当前信息系统的缺陷，实现优化打印店资源管理、改善用户体验，以及提供实时数据分析和报告，以支持更智能的决策制定。

2. 系统设计动机

## 2.1 背景介绍

由于高校属于知识与信息的密集区域，校内学生与教职工一直有着旺盛的打印需求，校内打印店必定能够实现盈利，然而当前打印店店铺管理主要依靠于店铺的管理员，在财务管理、资源管理、订单管理等方面缺乏相应的信息化管理方式，这种经营方式尽管能够保证店铺的正常运行，但是存在人力资源、物力资源等浪费，不利于店铺长期经营。且由于当前打印店侧重于实体店经营，在功能方面尽管当前满足大部分用户的基本需求，但是在面对用户多样化需求时，稍显乏力，在市场竞争中缺少差异化特点。因此，构建一个高效智能且用户友好的智慧打印店管理信息系统是势在必行的，不仅能够提升服务质量和操作效率，同时也是高校信息化建设战略的重要组成部分。

## 2.2 用户需求

智慧打印店信息管理系统涉及两个主要角色：客户（学生、教职工）和店铺管理员（店主、店员）。

**针对客户的需求。**客户在使用打印店系统时，其首要需求是能够满足基本的打印需求，在客户端上传打印订单的过程中，客户需要能够直观地选择打印参数，包括但不限于黑白或彩色打印、打印份数、单双面打印以及纸张大小。由于客户预算各异，客户期望系统能够提供各打印配置下的价格信息，以便能够做出经济合理的选择。此外，为了更好地管理个人的打印信息，客户还需要能够方便地查看历史订单。

然而，在高校环境下，客户面临一些特殊情况，因此客户也需要打印店提供其他方面的服务。因为部分客户的时间安排与打印店的营业时间可能存在冲突，客户需要打印店提供24小时工作的自动打印服务，以满足不同客户的灵活打印时间需求；在期末考试周等打印需求高峰时段，打印店内可能会出现人员拥挤、等待时间长等情况，客户需要预约打印的服务，提前规划并避免长时间的等待；考虑到打印店地理位置相对偏僻，对于一些住宿距离较远的客户，往返打印店的时间成本较高，为减轻时间成本，客户希望能够将打印的文件配送到住宿点，尽管需要支付少量的配送费。

**针对店主的需求**。作为打印店的关键人物，店主需要全面管理店铺，涉及店铺运营、员工管理、资源管理、以及财务管理等多方面。当前打印店管理主要依赖于店主的个人经验，缺乏相应的信息系统来统筹管理店铺。为了更好地管理打印店，店主期望系统能够提供完善的店铺运营数据分析功能，提供店铺的数据分析报表以及用户画像，从而协助店主更有效地经营打印店以及了解用户的消费习惯。

由于打印店需要实时查看客户订单并迅速处理待打印的订单，订单管理是打印店对外业务的关键功能。当前的打印店运营方式仅支持到店打印，客户只能通过添加店主的微信或其他社交帐号，通过通讯工具传输文件。这种经营模式没有对应的数据库存储订单信息，导致订单管理混乱，尤其在高峰期，容易出现漏单、打印错误等问题。故而信息系统需要能够提供统一管理用户订单的功能，其功能包括显示当前待打印的用户订单及订单详情、下载订单、以及在打印完成后切换订单状态。

在当前打印店中，各种店铺运行数据仍停留在纸质表格上，诸如财务信息、物资信息等数据均需要专人负责，管理相对繁琐，常导致记录不准确、无法实时跟踪等情况的发生。因此，店主期望信息系统具有财务管理、员工管理、物资管理等功能，以便管理打印店。

除了以上功能性需求外，用户还需要系统提供美观且易于操作的界面，便于用户能够快速理解系统功能，减少学习成本，享受良好的操作体验。还要有清晰的反馈机制，确保用户能够清楚了解其操作结果，包括登录成功、文件上传成功、支付成功等反馈提示，从而提升整体系统的实用性与用户满意度。

## 2.3.当前的信息系统

**印萌**[2]是广东印萌科技有限公司旗下的一款共享打印软件平台，涵盖自助打印、预约打印和文库等多项功能，覆盖移动端与电脑端。其功能主要分为三个方面：打印模式、专业赋能和引流增值。

在打印模式方面，印萌提供了电脑自助打印、手机自助打印与预约打印服务。用户需要通过相应程序先上传文档，并采用先付款后打印的营业模式；对于预约打印服务，用户可以随时随地上传文件，商家接到订单后按时完成，用户根据取货码到店取货。

在专业赋能方面，印萌提供便捷的文件上传功能，支持微信小程序登录上传文件，如图 1所示。平台还推出了会员卡系统，店铺可以发放电子会员卡，用户先充值后消费，以锁定客源并实现资金快速回收；印萌还为商家提供了财务管理系统，如图 2所示，后台能够自动记录每笔订单数据，自动记录分析每日、每月店铺流水数据。

|  |  |
| --- | --- |
|  |  |

图 1印萌微信小程序端界面



图 2印萌管理端界面

在增值引流方面，印萌首先提供了免单服务，商家可以通过平台投放免单补贴，用户到店打印有机会享受免单服务；其次，印萌还提供打印证件照和论文查重服务，进一步满足用户多样性的打印需求。

综上分析发现，印萌在打印业务上的功能覆盖面广，几乎能够满足上述用户需求，但是仍有部分需求没有满足。首先，印萌的服务范围局限在打印店内，缺乏对外配送服务，客户必须到店取货。其次，客户端基于微信小程序，限制了应用的可见性和用户获取渠道。

3. 商业模式设计

可以绘制出如图 3所示的打印店案例分析图。在这个竞争激烈的行业里，本小组使用信息系统提升客户体验和店铺管理效率。

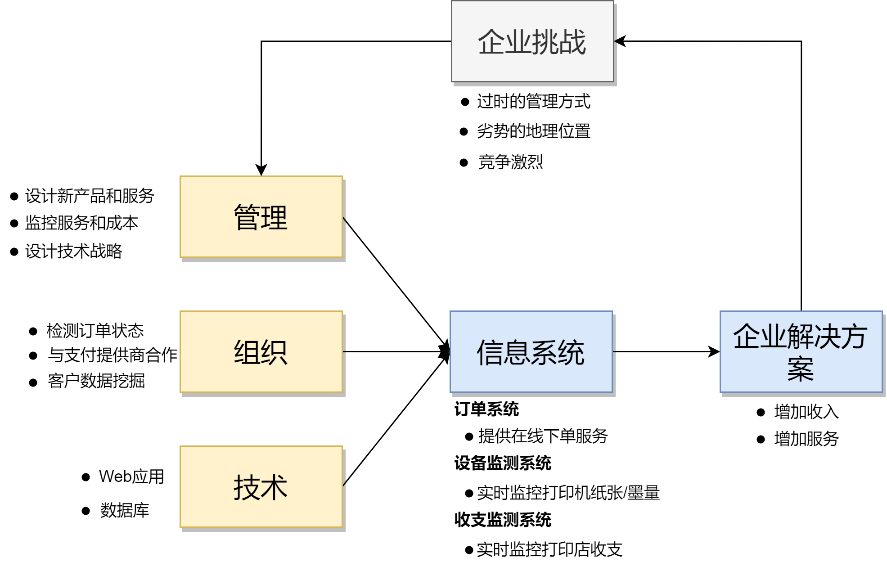


图 3 打印店分析图

项目系统正式上线服务，我们将充分利用商业画布的九大要素，系统地描述和运营商业模式，具体如下：

客户细分

就**客户细分**而言，本小组开发智慧打印店管理信息系统面向高校内部打印市场。**客户**主要是高校内的不同用户群体。可以将校内用户划分成学生与教职工两大主要群体。学生作为主要客户，预算相对有限，通常追求方便、快捷且价格适中的打印服务，可能需要频繁印刷课程作业和实验报告。另一方面，教职工也是可能的用户，预算相对充足，对印刷质量有更高的要求，可能需要定期大批量印刷教材或研究资料。然而，考虑到教职工通常拥有学校配套的打印设备，因此在打印店消费的可能性相对较低。因此本系统的主要目标是满足校内学生群体的多样化打印需求。

* 价值主张

在**价值主张**方面，本智慧打印店系统有清晰的优势。首先，系统为用户提供快捷便利的服务，用户可以随时随地上传文件，并通过预约打印，在自己的空闲时间利用取货码到店取货。这不仅使用户能够轻松快速完成打印需求，还能够避开高峰期，最大程度减少等待时间，提高效率。其次，系统能够满足用户的个性化需求，为用户提供多样的印刷选项，从纸张尺寸、打印份数，到打印类型（黑白、彩印）、打印范围，用户可以根据具体要求进行选择。且在价格方面，系统提供了公开透明的价格公示，用户可以通过平台了解每项服务的费用，在预算范围内完成打印任务。此外，我们的智慧打印店提供配送服务，满足用户不同的时间和地点需求。最后，系统支持多平台客户端，提供直观友好的在线平台，使用户能够方便地使用本系统以完成打印服务。

* 渠道通路

在**渠道通路**方面，本智能打印店管理信息系统通过多渠道方式直达用户。在线方面，我们提供手机客户端、PC客户端以及Web端，为用户提供随时随地的上传文件的方式。同时，线下建立打印店门店，为用户提供更直观、更个性化的服务。

* 客户关系

在**客户关系**方面，我们的策略是开发新用户和维护当前客户。为了吸引新用户，采用积极宣传的措施，通过社交平台运营和组织免单活动等方式提高打印店的校内知名度。此外，我们提供便利的打印服务，促使新用户使用我们的系统。为了维护当前客户关系，留住老用户，我们将采取多重措施。首先，我们将积极响应用户反馈，以确保客户满意度与体验。其次，计划开发VIP服务，为忠实用户提供优质服务，增强客户黏性。

* 营收来源

在营收来源方面，主要是客户打印费用以及配送收入。首先，客户打印费用是我们的主要的营收，通过提供各种印刷选项，我们能够收取客户打印的相应费用。其次，配送收入也是重要收入来源，我们提供便捷的配送服务，满足客户在不同时间和地点的取货需求。

* 成本结构

在成本机构方面，包含店面租赁费用，打印设备与纸张费用，人力成本，平台运行与维护成本。店面租赁费用是经营打印店的固定成本。为了保证打印店能够提供高质量的打印服务，必须投资购买先进的打印设备以及纸张，以满足客户多样打印需求。同时，人力成本是关键部分，包括雇佣店员、客服和技术人员。此外，平台运行与维护成本涉及服务器托管、系统维护等，以确保我们的智慧打印店管理信息系统高效稳定运行

* 核心资源

经营打印店的核心资源包含经验丰富的人力资源、先进的打印机、现代化电脑、优质纸张。为确保业务能高效运作，需要经过业务培训的人员。电脑是运行整个信息管理系统的的基础，让店主能够进行管理订单、财务管理等。

* 关键活动

在打印店的商业模式中，订单管理、在线支付、财务管理、平台管理、营销宣传等关键活动保障业务流程运转。订单管理是打印业务的起始步骤，系统将订单的详细信息有效传输给打印店。财务管理是打印店可持续发展的保障，包括每日收入支出监控，确保财务状况良好。平台管理确保打印店管理信息系统的稳定运行，包括流量监控、技术支持和界面优化。最后营销宣传是吸引新用户的关键活动，可以通过多种途径实现，包括促销活动，社交媒体运行等。

* 重要合作

打印店运营中的重要合作伙伴包括**支付服务提供商、学校、材料供应商和外送平台**。与支付宝、微信等支付服务提供商合作，以确保安全方便的在线支付方式，且便于进行财务管理；与学校的合作对可以更好地服务学生和教职员工，增加打印店的订单数；与材料供应商的合作确保了打印店能够获得高质量的印刷材料，保障印刷质量；与外送平台合作，可以支持订单配送服务，为用户提供更为便捷的取货方式。

综上所述，我们可以绘制本智慧打印店的商业画布。如图 4所示。

图表, 树状图

描述已自动生成

图 4 商业画布

4. 系统的功能设计

本智慧打印店管理信息系统部署在服务器上，采用数据库管理数据，实现了包括账号管理、订单管理、库存管理、财务管理等功能，如图 5所示。

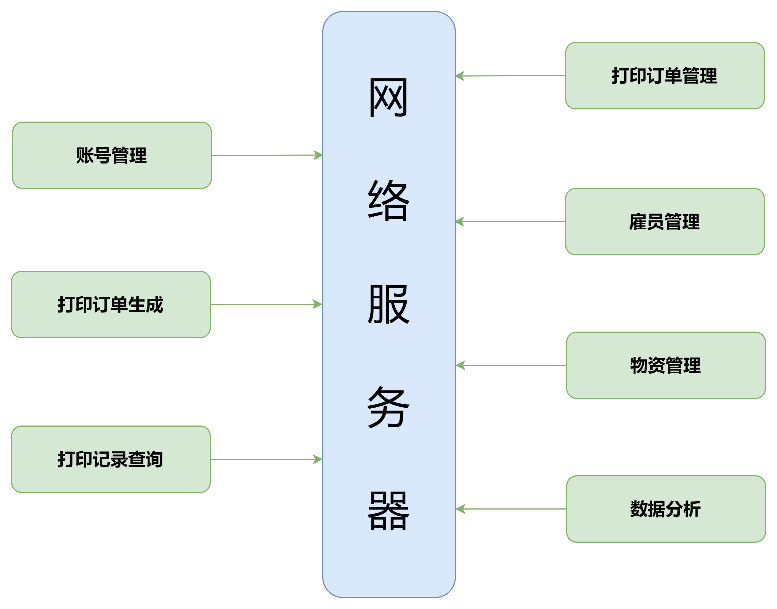


图 5打印店管理系统流程图

## 4.1 系统实现技术：渐进式Web应用（PWA）

本打印店管理信息系统采用渐进式Web应用（PWA）技术，具有跨平台兼容性、自动更新和类似原生应用体验的三大特性。

本系统具有较强的跨平台兼容性，无论是在桌面端、平板还是移动设备上使用系统，用户都能享受到相似的界面和功能，提高了系统的灵活性和可访问性。用户可以通过支持PWA的浏览器（Chrome、Edge）访问本智慧打印店网站，浏览器会自动触发安装提示，用户可以安装PWA应用，在设备上创建应用图标，使用户可以通过点击图标直接启动应用，从而获得与本地应用相似的流畅启动和操作体验，提高了用户对系统的使用感受和满意度，PC端如

图 6所示，移动端如图 7所示。自动更新特性确保系统能够在后台自动检测并应用新的版本，使用户始终体验到最新的功能和性能优化，同时减少了用户的维护成本。

|  |  |
| --- | --- |
| 图形用户界面, 文本, 应用程序  描述已自动生成  图 6 PC端应用安装 | 图形用户界面, 应用程序, 聊天或短信  描述已自动生成  图 7移动端安装 |

## 4.2 系统主要功能

* 账号管理

账号管理功能是智慧打印店最基础的功能，是用户与系统进行交互的入口。本系统具备基本的登录/注册模块，且在登录界面，可以根据用户名自动匹配对应的账户类型，从而登录到相应的“顾客”/“店员”页面。该模块的流程图如图 8所示。

手机屏幕的截图

描述已自动生成

图 8 登录/注册流程图

若客户没有账号，需要进行账号注册，完成填写账号名、密码、确定密码等操作，如图 9所示。系统会判断账号名是否与客户账号数据表中已有账号名重复。若账号名不重复，则将账号/密码写入客户账号数据表，表示注册成功，同时系统需要弹出提示信息，提醒客户账号注册成功；否则系统会弹出提示信息，提醒用户更换非重复的账号名；在拥有账号情况下，客户可以直接登录，需要输入账号/密码，如图 10所示。系统同时检索管理员账号数据表和客户账号数据表，判断账号/密码是否正确，正确则进入账号对应界面。为实现本功能，系统使用数据库存储账号数据，客户账户数据表如图 11所示，每条记录由“客户ID”、“密码”、“客户名”、“寄送地址”组成。管理员账户数据表如图 12所示，每条记录由“管理员ID”、“密码”、“管理员名”组成。

|  |  |
| --- | --- |
| 图 9 注册界面 | 图 10 登录界面 |

表格

描述已自动生成

图 11 客户账户数据表

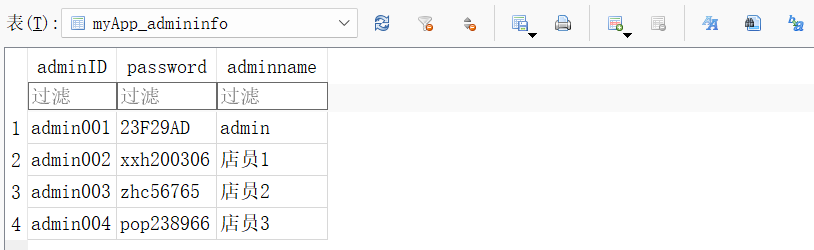


图 12 管理员账户数据表

* 订单管理

本系统支持客户订单管理。客户订单可以用图 13的E-R图表示，即一个用户可以创建多个订单，一个订单中可以有多个文件，一个文件可以出现在多个订单中。本系统将一件订单作为一条记录存储在订单数据表中，订单数据表如图 14所示，每件订单属性包含订单ID、客户ID、订单状态、订单金额、文件名、取货码、文件ID、打印颜色、单双面选择、纸型、预定取货时间、寄送地址、备注。



图 13 实体-关系图

图形用户界面, 文本

描述已自动生成

图 14 订单数据表

客户登录客户界面，如图 15所示。【价格公示】提供了打印店各种打印类型的价格，如图 16所示。

|  |  |
| --- | --- |
| 图 15 客户端界面 | 图 16 价格公示 |

【历史打印订单】页面展示了客户的历次打印订单，点击“详情”还可以进一步查看订单中各打印文件的信息，如图 17、图 18所示。

|  |  |
| --- | --- |
| 图形用户界面, 文本, 应用程序, 聊天或短信  描述已自动生成  图 17 历史订单页 | 图形用户界面, 文本, 应用程序, 聊天或短信  描述已自动生成  图 18 订单详情页 |

客户可以新建打印订单，订单界面如图 19所示。在订单界面，详细地指出了创建订单的具体操作，满足了客户对直观易操作界面的要求。为了满足客户多样性的打印需求，系统支持客户进行打印设置，如图 20所示，预置选项包括纸型（A4、A3）、颜色（黑白、彩色）、单双面、打印范围、打印份数，若客户还有其他需求，可以同时在备注中说明。

|  |  |
| --- | --- |
| 图 19 新建订单 | 图 20 订单设置 |

完成打印设置之后，客户还可以选择预约送货上门时间，能够方便地完成打印任务，避免前往打印店耗费时间。系统将会自动计算订单金额，客户只需提交订单并完成支付即可，如图 21所示。客户可以查看历史打印订单，会显示客户的每一笔订单详情，包括订单创建时间、订单金额、取件码、订单状态等，该功能满足了客户对订单的全面管理需求，为客户提供了方便的订单查阅和追踪功能，如图 22所示。此时订单已经上传至服务器。

|  |  |
| --- | --- |
| 图 21 新建订单 | 图 22 订单详情 |

订单管理模块允许管理员实时管理当前待打印订单。在管理员后台中，每份订单以“订单号”“预定交货时间”、“客户取件码”、“订单状态”的形式展示，如图 23、图 24所示。

|  |  |
| --- | --- |
| 图 23订单管理 | 图 24订单管理 |

管理员可以通过订单状态栏进入对应订单的打印界面，如图 25所示。在该界面中，会显示订单的各项设置以及用户备注。可以在该界面打包下载订单文件。完成打印后，在该界面点击【打印完成】，代表完成该订单的打印任务。此时，订单状态转移到“待配送”状态，如图 26所示。可以点击【待配送】，将订单分配给专送员。

|  |  |
| --- | --- |
| 图 25 订单打印 | 图 26订单管理 |

* 设备管理

系统将打印机状态记录在打印机状态数据表中，如图 27所示。每条状态记录包括打印机ID、打印机名、剩余纸量、剩余墨量、打印机状态，支持管理员实时监控当前打印机的状态。系统提供了直观的可视化视图，使管理员能够一目了然地了解打印设备当前剩余纸量与剩余墨量，确保设备在最佳状态下运行状态，该模块界面如图 28所示。

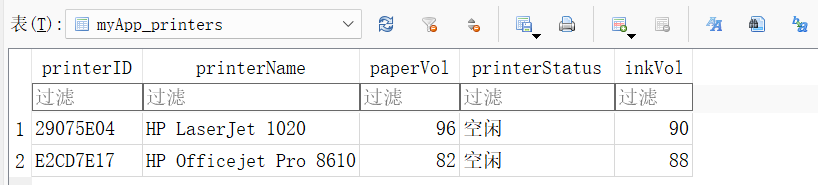


图 27打印机状态数据表

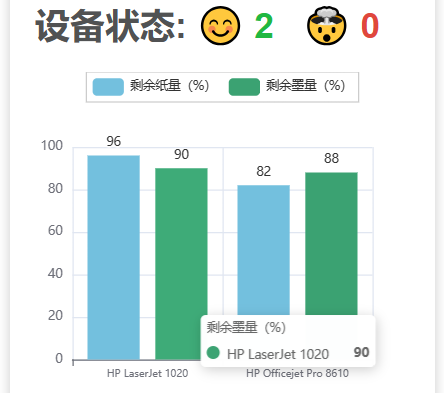


图 28 设备管理

* 财务管理

为了实现财务管理，系统将打印店的所有财务记录存储在财务数据表中，如图 29所示，每条记录由交易ID、交易金额、创建时间、记录类型，财务类型分为收入与支出，收入方面涵盖打印服务收入、配送收入，而支出包括电费、采购支出、员工薪资等。系统会通过订单自动计算打印店当前收入，减少了人工统计工作量。同时，为了更便于管理，系统还提供了直观的可视化图表，支持管理员查看当前打印店的收支情况，使管理员能够更清晰地了解和分析店铺地经济状态，如图 30所示。

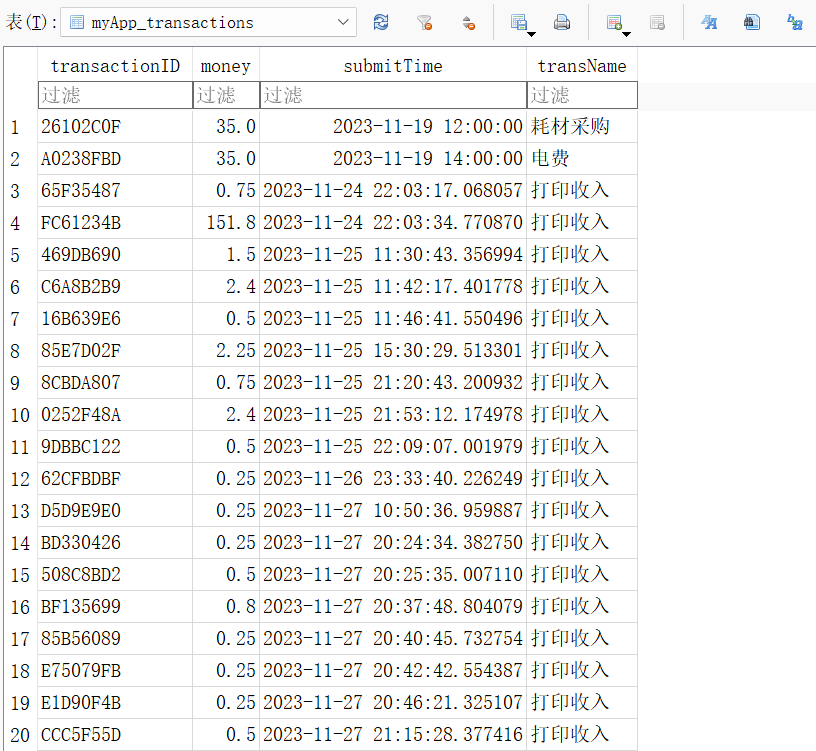


图 29 财务数据表

图表

描述已自动生成

图 30 收支管理

* 员工管理

系统支持管理员对员工进行管理，涉及两个数据表，分别是员工基本信息数据表、员工出勤表。员工基本信息数据表记录了员工的基本信息，每条记录包括姓名、性别、身份证号、联系方式、岗位和入职时间。员工出勤表记录了员工的出勤情况，包括每天上下班时间、迟到早退情况，系统每月会自动计算每位员工的工作时长和加班时长，便于工资计算和绩效评估

* 数据分析

系统支持对打印店运营数据进行多方面分析。通过对用户历史购买数据的深度挖掘，系统能够准确识别不同用户的偏好与需求，以及整个用户群体的特征。这使得在不同时间段能够合理调配店内资源，从而提高人力资源与物力资源的利用率。系统同时具备计算用户留存率和流失率的功能，有助于店铺评估用户忠诚度，同时为管理者提供科学依据，以制定有效的用户留存策略。此外，系统还可以对店内运营数据进行全面分析。通过深入研究不同商品的销售情况，包括热销商品、滞销商品和总销售额等信息，系统协助打印店管理者制定有针对性的采购和促销策略。在执行促销或广告活动后，系统还能够精准分析活动效果，包括吸引新用户数量和提升销售额等方面的数据，以便优化未来的营销策略。

通过对网站流量检测，可以实现用户行为分析，管理员可以方便地登录仪表盘并选择流量监控工具，实时了解网站的访问量、请求来源、地理位置等信息。我们的系统接入了Cloudflare，通过Cloudflare的历史流量分析功能允许管理员查看不同时间段内的访问趋势，如图 31，而流量来源分析则帮助了解用户发现和访问网站的方式。总体而言，通过Cloudflare仪表盘，系统网站得以全面监控和分析流量，确保网站的正常运行和安全性。

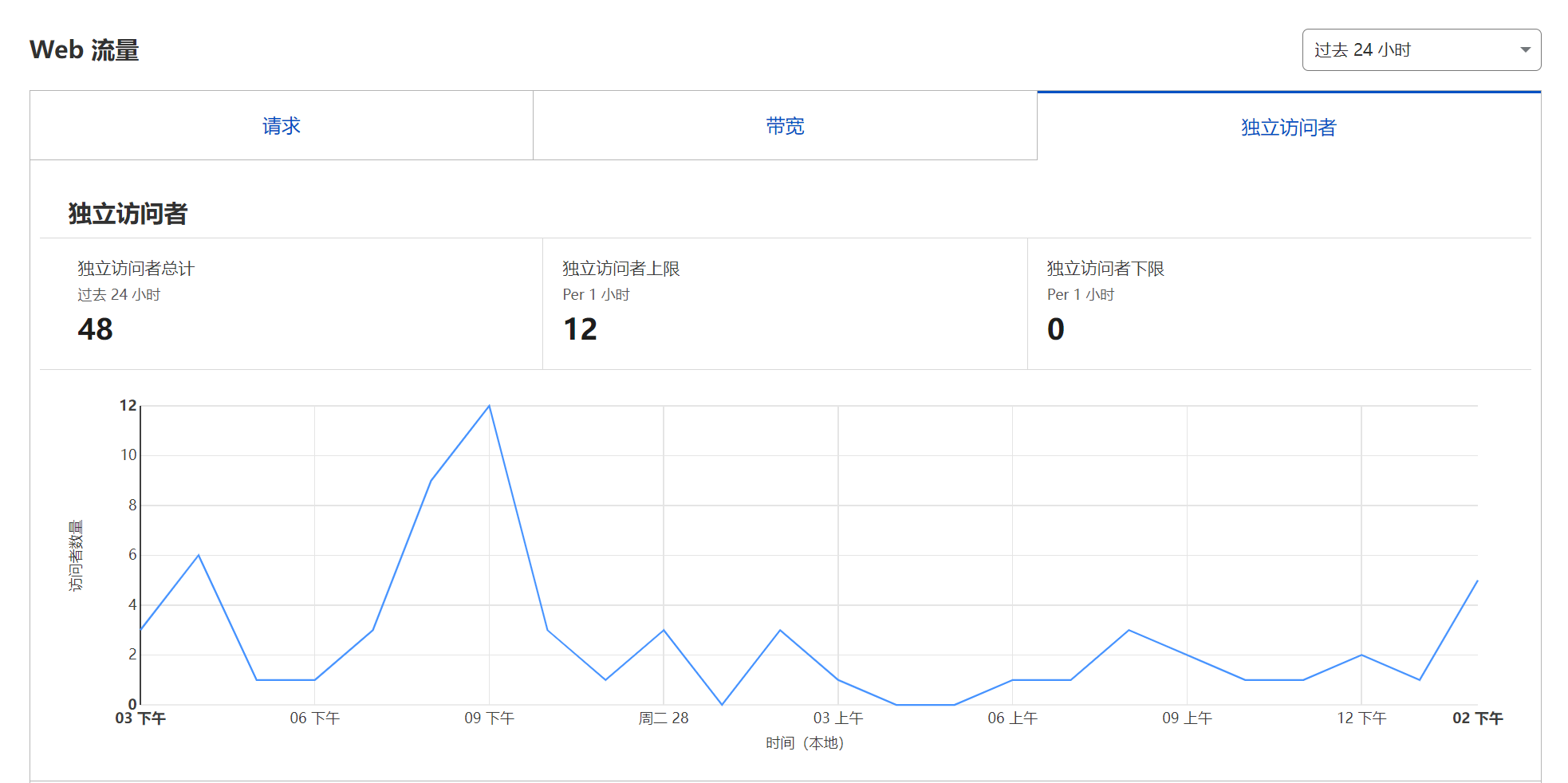


图 31 用户行为分析：密集访问时间分析

## 4.3 系统特性：清晰的反馈机制

系统不仅满足了上述功能性需求，还特别注重了用户体验，提供了清晰的反馈机制以确保用户对每一步操作的结果有明确了解。用户在登录或注册成功时会收到相应的提示。在创建订单过程中，系统会细致地提示用户是否 “未选择要打印的文件”、是否“打印文件为上传/打印设置未完成”，以及成功上传文件提示“文件上传成功”。用户支付成功后，会得到“打印任务已提交”的明确反馈。管理员在处理订单时同样有对应的反馈。系统会在选择配送订单时询问“是否分配专送员”以确定是否配送，确定分配成功后将得到相应提示，确保管理员能够清晰了解并管理整个流程。这一套反馈机制的设计提升了系统的用户友好度，满足了用户对操作可视性的需求。

5. 系统开发过程介绍

本系统完全由小组成员借助相关开发框架的官方文档**独立完成**，开发过程可以分为如下四个步骤。

## 步骤1：确定系统开发环境和运作模式

本系统基于Vue+Django框架开发，在应用上实现了前后端分离。

前端使用Vue，这是一个基于JavaScript的Web应用前端开发框架，用于呈现用户端（包括管理员）界面，接收用户与页面的交互动作并传递给后端。我们使用Nginx作为前端服务器，这是一款轻量级的Web 服务器，可以很便捷地将项目从测试环境部署到服务器中。

后端使用Django，它是一个基于Python的高级Web应用框架，提供了一套强大的工具和功能，用于简化和加速Web应用程序的开发[4]。此外，它提供了强大的对象关系映射（ORM）系统，允许开发者使用Python代码而不是SQL语句来操作数据库。

由此我们构建了层次式风格的系统体系结构：中心是数据存储，即系统用于对外提供服务的所有数据，如用户账号信息、用户打印订单信息；中心向外第一层是数据处理层，包括与数据的各种交互，由后端实现；第二层是用户接口层，用户端的各种操作会在这里被封装成消息传送给后端，后端对消息的响应也会在这里解析；第三层（最外层）是用户界面层，用于呈现用户界面及支持用户与系统的交互。这种体系结构类似于B/S（browser/server）结构，在物理上隔离了客户（请求信息）、程序（处理请求）和数据（被操作），大大减轻了后续编码的工作量。

## 步骤2：前端开发

* 文件组织与编码规范

Vue项目创建后，其基本文件结构如图 32所示。dist文件夹用于项目部署，node\_modules文件夹存放项目中要用到的各种库和工具，src文件夹则是源代码的书写区域。

表格

中度可信度描述已自动生成

图 32 一个一般的Vue项目的文件组织

我们在./src目录下新建文件夹views，进一步在./src/views目录下创建若干个文件夹，用于存放所有制作网页页面的Vue文件，文件夹名表示相应网页的功能。由于项目本身体量不大，规定每一个网页页面都只由一个Vue文件进行描述（命名为index.vue），这样views文件夹可以取代assets和components文件夹。./src/views目录的文件组织如图 33所示。

图片包含 图表

描述已自动生成

图 33 ./src/views目录结构

Vue使用官方路由器Vue Router在应用程序中实现不同网页（视图）之间的跳转。我们把路由配置文件index.js存放在./src/router目录下，部分路由表项如图 34所示。

文本

描述已自动生成

图 34 系统部分路由表项（Login、Register、Homepage\_admin、Homepage\_user分别表示登录页面、注册页面、管理员主页、用户主页）

经过上述改造，本系统前端的文件组织可用图 35进行描述。在编码时，只需要根据需求编写页面文件，并添加相应的路由表项即可。

图片包含 文本

描述已自动生成

图 35 前端项目的文件组织

Vue项目有两种编码方式：选项式API（Options API）和组合式API（Composition API）。这里我们选用了选项式API，主要是因为选项式API的编码风格类似于面向对象编程，易于学习使用；且本项目并不复杂，一定程度上减轻了选项式API代码逻辑复杂较难理解的缺陷。另外，我们还规定代码中的变量、方法等一律采用驼峰命名法（如totalPrice、backHomepage()），代码缩进为2字符等其他编码规范，便于开发人员阅读和使代码美观。

每一份网页页面文件源代码index.vue都按Vue单文件组件（Single-File Component，简称SFC）的格式开发。图 36是SFC的一个示例，可见SFC是网页开发中 HTML、CSS 和 JavaScript 三种语言经典组合的自然延伸。<template>、<script> 和 <style> 三个块在同一个文件中分别封装、组合了组件（网页页面）的视图、逻辑和样式[3]。

文本, 聊天或短信

描述已自动生成

图 36 SFC示例

下面将以项目中的几个重要文件为例，介绍具体的编码过程。

* 两个核心文件：./src/main.js和./vue.config.js

main.js和 vue.config.js 中存放的是Vue项目的基本配置信息和全局导入的库。在vue.config.js中，通过如下代码进行跨域配置：

文本, 信件

描述已自动生成

图 37 配置跨域

main.js则是整个Vue项目的入口文件。主要对项目进行全局的配置，比如创建整个Vue项目的根实例、引入第三方库、设置挂载点等。main.js的代码如图 38所示。

文本

描述已自动生成

图 38 main.js的代码

* 用户提交打印任务页面：./src/views/createPrint/index.vue

用户提交打印任务页面是本系统前端最为复杂的页面。views文件夹中的所有页面Vue文件的代码写法都与其类似，因此本部分将详细介绍该页面的开发过程，之后的登录页面介绍部分则会省略一些细节。

首先设计该网页的视图及样式，即<template>和<style>部分。为了达到适配手机端的效果，第一步我们自定义了一个容器container用于容纳网页中的各小组件，在<style>块中规定其CSS样式，主要设置了container的宽度、背景颜色、边框弧度、阴影等。代码及效果如图 39所示。

图片包含 文本

描述已自动生成图形用户界面, 文本

描述已自动生成

图 39 container的CSS样式（左图）和效果（右图）

第二步，根据提交打印任务这一功能的需要，我们设计了返回用户主页按钮、提交打印任务的步骤条、打印文件上传模块等用户交互部件。这些部件均由Vue的一个开源组件库[Element Plus](https://element-plus.org/zh-CN/)创建，UI较为美观实用。将小组件加入之前定义的container之后，在<style>块中添加对应的CSS样式使小组件的位置、间距等协调、美观。本页面功能相对复杂，小组件较多，这里仅展示部分小组件的代码，如图 40所示。

图片包含 文本

描述已自动生成

图 40 部分小组件的创建代码（从上到下依次为：返回用户主页按钮、提交打印任务的步骤条、打印文件上传模块）

接下来重点介绍处理页面功能逻辑的<script>块。图 41展示了<script>块的结构：开头导入本页面需要用到的工具（node\_modules中的模块）；之后类似于面向对象编程，规定页面中需要处理的各种变量（data(){}部分）和实现页面功能必要的方法（methods:{}部分）；最后利用Vue的生命周期钩子mounted，定义一些在本页面加载时需要执行的简单操作。

图片包含 图形用户界面

描述已自动生成

图 41 <script>块结构

methods中定义了12个方法，能够完整地处理用户提交打印任务页面的所有逻辑。下面将逐一介绍所有方法。

backHomepage()在用户点击“返回”按钮时触发，利用./src/router/index.js中的路由表导航回用户主页；disablePastDates(time)对用户能够选择的预约送达时间作了规定，不能选取早于当前日期的时间；getPrice(i)方法也较为简单，用于计算用户上传的id（由系统设置）为i的文件的打印价格，方法体内首先根据该文件的打印设置计算出每张纸的打印单价，然后把单价依次与打印范围和打印份数相乘，得到该文件的打印总价。

beforeUpload(file, fileList)、submitUpload()、submitSuccess()和submitFailed()这四个方法处理向服务器上传文件这整个过程的逻辑。

beforeUpload(file, fileList)方法在系统向服务器上传文件前触发，将用户上传的待打印文件的文件信息（文件名、文件类型等）存储到uploadedFiles数组中，这主要是为了创建表格，向用户展示已上传文件的文件信息。代码如图 42所示。

文本

中度可信度描述已自动生成

图 42 beforeUpload方法的实现代码

submitUpload()方法体内，语句this.$refs.uploadRef.submit()将用户待打印的文件以二进制流的形式传输向服务器。另外，如果uploadedFiles数组长度为0（即用户未上传任何文件），则向用户提示：“未选择要打印的文件！”。

submitSuccess()和submitFailed()方法顾名思义，分别对文件上传成功和失败的处理。如果成功，则向用户提示：“文件上传成功！”；反之提示：“文件上传失败！”。

除了要向服务器上传用户待打印的文件，还需要上传对应的打印设置（如用A4纸还是用A3纸打印，彩打还是黑白打印）。本系统中用dataStorage数组储存这些打印设置，其中的一个元素即代表一份文件的打印设置。在展示用户已上传文件的文件信息表格（图 43左）中，用户点击每个文件最后的“配置”链接，即可打开文件打印设置对话框（图 43右）。

图形用户界面, 表格

中度可信度描述已自动生成 图形用户界面, 应用程序

描述已自动生成

图 43 已上传文件的文件信息表（左）和打印设置对话框（右）

每个文件都有与之对应的打印设置，需要保存这些打印设置并保证每个文件的打印设置对话框都能够储存用户上次的更改。printSettings(id)和saveSettings()两个方法实现了这一功能：前者在某个文件的打印设置对话框打开前预加载此文件先前为其配置的打印设置（通过读取dataStorage数组），后者则将此次打印设置结果保存到dataStorage数组的相应文件元素中。它们的代码如图 44所示。

文本

描述已自动生成

图 44 printSettings和saveSettings实现代码

最后剩余的3个方法showPay()、payCancel()和submitTask()用于用户提交打印任务时的付款结算。用户点击“提交打印”按钮后首先触发showPay()展示付款对话框。如果用户点击“取消支付”将无事发生，触发payCancel()关闭对话框，可以继续进行打印设置或者直接返回主页；如果用户扫描二维码完成付款，用户点击“已完成支付，继续”按钮，触发事件submitTask()，系统会向服务器提交本次打印任务的各文件打印设置，同时利用Vue Router跳转页面至本次打印任务的详情页面，供用户检查是否有错误。

* 登录页面：./src/views/login/index.vue

登录页面的任务是将用户输入的信息通过API发送给后端，再根据后端返回的内容，做出如下之一的决策：账户密码无效/登录到用户界面/登录到管理员界面。我们在<script>块的methods部分定义了方法doSubmit()，该方法通过axios向后端指定的API传参（用户输入的用户名与密码）和对返回内容进行相应操作来处理登录逻辑。代码如下图所示。

文本

描述已自动生成

图 45 doSubmit的实现代码

在页面设计上，我们使用[Element Plus](https://element-plus.org/zh-CN/)的组件让页面更加美观，具体代码如下：

日程表

描述已自动生成

图 46 登录界面<template>块设计

同样，组件还设置了一些定义在<style>块中的CSS样式。相关代码如下：

文本

描述已自动生成

图 47 登录界面相关css

## 步骤3：后端开发

* 后端开发环境构建

首先，我们需要在控制台终端中输入以下命令以安装Django模块。



然后，我们在控制台终端中输入以下命令，创建一个名为djangoproject的Django工程目录。

 我们进入该目录，在控制台终端中输入以下命令，创建一个名为myApp的Django应用，作为后续进行数据操作的主要工具。

 至此，我们所构建的文件结构如图 48所示。

手机屏幕的截图

描述已自动生成

图 48 Django后端目录结构

为方便后台操作，我们需要在djangoproject/settings.py中添加自己创建的Django应用myApp。

文本

描述已自动生成

图 49 添加Django应用

* 数据库和跨域配置

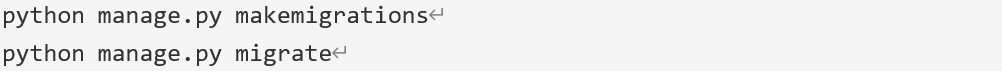
在Django后端开发中，我们采用Django默认的SQLite3数据库。首先，我们需要在工作目录中创建一个SQLite3数据库文件。Django采用的是M（Models，即数据模型）V（Views，即数据视图）T（Templates，即展示模板）架构，在myApp/models.py中创建的每个类即对应SQLite3数据库文件中的一张数据表，类的每个成员变量即为数据表中的相应字段信息。因此我们在myApp/models.py创建一个类UserInfo，用于存储用户的用户名和密码等信息，该类对应数据表的主键为userID字段。

文本

描述已自动生成

图 50 用户信息类

然后，我们在控制台终端中依次输入以下命令，将创建的表同步到数据库文件中。

 上面的命令会在Django的工程目录下自动生成一个名为db.sqlite3的文件，这就是我们的SQLite3数据库文件。

为解决跨域和服务器部署问题，我们需要修改djangoproject/settings.py中的ALLOWED\_HOSTS和CSRF\_TRUSTED\_ORIGINS。

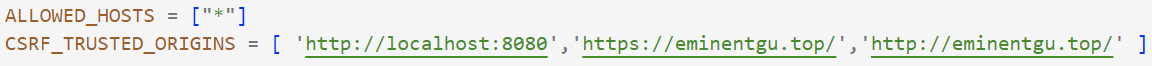


图 51 服务器部署设置

此外，为提高可读性，我们将Django后台的语言改为简体中文，同时将时区改为上海时区，以保证在数据库表中新建记录时生成的时间点准确无误。

图片包含 图示

描述已自动生成

图 52 语言和时区设置

* 数据表构建

仿照上面的步骤，我们在myApp/models.py中创建一个类AdminInfo，用于存储打印店管理员的用户名和密码等信息，该类对应数据表的主键为adminID字段。

文本

描述已自动生成

图 53 管理员信息类

为存储打印机的当前状态信息，我们创建如下的Printers类，其中paperVol和inkVol两个字段分别代表打印机当前剩余的纸量和墨量（以百分数表示），printerStatus字段为打印机当前的状态，如“空闲”“工作中”“故障”等，该类对应数据表的主键为printerID字段。

文本

描述已自动生成

图 54 打印机信息类

为存储用户提交的打印任务，我们创建如下的Tasks类，其中userID和taskID分别为用户的用户名和打印任务的订单号；fileName为需要打印的文件名；submitTime为打印任务的提交时间（当提交一个打印任务时，我们通过设置auto\_now\_add参数为True，该字段的值将被自动填充为打印任务的提交时间）；paper、color、doubleSided为待打印文件的打印配置，分别代表打印纸型、黑白/彩色打印、单面/双面打印；money为待打印文件的打印费用；taskStatus为打印任务的当前状态，分为三种情况：待打印、待配送和已完成；reservedTime为打印任务的预约取货时间；pickCode为打印任务的取件码。该类对应数据表的主键为fileID字段。

需要注意的是，在该类对应的表myApp\_tasks中，每一条记录表示用户提交的一个待打印文件。若用户提交的打印任务中包含了多个文件，则这些文件在userID、taskID、submitTime、taskStatus、reservedTime、pickCode六个字段上的值相同，表示它们来自同一个打印任务。

文本

中度可信度描述已自动生成

图 55 打印任务信息类

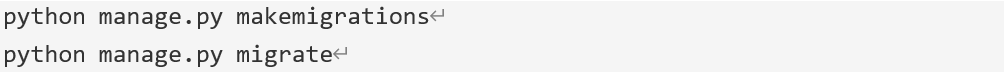
为存储收入和支出条目信息，便于店主了解打印店的资金流向，我们创建如下的Transactions类，其中transName为收支项目名称，submitTime为收支项目的提交时间，money为收支项目的金额。该类对应数据表的主键为transactionID字段，表的每一条记录表示一个收入或支出条目。

图片包含 文本

描述已自动生成

图 56 收支项目信息类

需要注意的是，每当我们对myApp/models.py中的某个类进行修改后，都需要在控制台终端中依次输入以下命令，将改动同步到数据库文件中。每次的修改记录均会以一个.py文件的形式存储在myApp/migrations目录下。



* 数据表操作

在定义了数据存储的模型（即数据表）之后，我们需要根据前端的需求进行相应的数据表操作，并在myApp/views.py中将这些操作连同返回给前端的响应信息一起封装为一个个独立的函数模块，以下是myApp/views.py中定义的各函数模块。

* 用户/管理员注册

图形用户界面, 文本

描述已自动生成

图 57 用户/管理员注册

* 用户/管理员登录

图形用户界面, 文本, 应用程序

描述已自动生成

图 58 用户/管理员登录

* 当日盈利计算

图形用户界面, 文本, 应用程序

描述已自动生成

图 59 计算当日盈利

* 获取当前剩余的打印耗材情况和当日收支情况

文本

描述已自动生成

图 60 获取当前剩余的打印耗材情况和当日收支情况

* 获取打印机当前状态

文本

中度可信度描述已自动生成

图 61 获取打印机当前状态

* 用户提交打印任务

图片包含 文本

描述已自动生成

图 62 用户提交打印任务

* 管理员处理打印任务

文本

描述已自动生成

图 63 管理员处理打印任务

* 用户查看历史订单列表

文本

描述已自动生成

图 64 用户查看历史订单列表

* 用户查看订单详情

图形用户界面, 文本

描述已自动生成

图 65 用户查看订单详情

* 管理员查看待处理（待打印/待配送）订单列表

文本

描述已自动生成

图 66 管理员查看待处理（待打印/待配送）订单列表

* 管理员查看待处理（待打印/待配送）订单详情

文本

描述已自动生成

图 67 管理员查看待处理（待打印/待配送）订单详情

* 模拟打印件配送过程

图形用户界面, 文本, 应用程序

描述已自动生成

图 68 模拟打印件配送过程

* 接收用户上传的文件并暂存

文本

描述已自动生成

图 69 接受用户上传的文件并暂存

* 将指定订单目录下的所有文件名传递给前端，用于下载用户上传的文件

文本

描述已自动生成

图 70 将指定订单目录下的所有文件名传递给前端

* 清除文件暂存区

文本

描述已自动生成

图 71 清除文件暂存区

* 与前端交互的接口定义

为实现Django后端与Vue前端的交互，我们需要在djangoproject/urls.py中配置路由信息，使得后端能够接收到前端发出的请求信息并返回响应。

手机屏幕截图

描述已自动生成

图 72 前端请求URL配置

## 步骤4：系统部署

* 服务器基本环境配置
  + - 1. 服务器选择Vultr提供的1核1g 25g服务器；
      2. 系统为Ubuntu 22.04 x64.

图形用户界面, 应用程序

描述已自动生成

图 73 服务器基本信息

* 域名配置

我们使用Cloudflare作为我们的DNS域名解析商，同时，Cloudflare还可以一定程度上的保护网站免受攻击，以及在服务不可用时提供响应界面，提升用户的体验。

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

图 74 CloudFlare解析域名

* SSL证书配置及PWA

我们配置了SSL证书，我们的网站支持更安全的Https访问。这也让我们的应用可以支持PWA，因此我们的系统可以以“应用程序”的形式安装在Android、IOS、Windows、Linux系统并正常运行。

图形用户界面, 文本, 应用程序

描述已自动生成

图 75 配置ssl证书且支持PWA

* 前端Nginx部署、前端上线

通过Nginx可以轻松部署build的vue项目，同时协调好前后端的跨域问题。本地通过终端命令npm run build打包vue项目，上传服务器后其部署在目录/root/webapp/dist中。

* 后端部署

后端采用Django框架，在服务器上使用conda创建python容器，并安装Django库及其他依赖库。

参考文献

1. 倪星云. 校园打印店经营的可行性分析[J]. 投资与合作, 2021(07):215-216.
2. [印萌自助打印软件\_8000+间文印店免费试用 (inmeng.net)](https://www.inmeng.net/product)
3. Vue官方文档：[单文件组件 | Vue.js (vuejs.org)](https://cn.vuejs.org/guide/scaling-up/sfc.html)
4. Django官方文档：[Django 文档 | Django 文档 | Django (djangoproject.com)](https://docs.djangoproject.com/zh-hans/4.2/)