

# 项目开发报告

组长：史浩天 24336100

组员：张婷 23303077、蔡锦 24336003、鲁天昱 24336083 (业绩权重: 1/1/1/1)

## 1. 项目概述

- 基本内容：实现一个校园二手交易平台，提供用户账号、商品浏览和搜索、发布与管理、图片上传等功能，采用前后端分离。
- 技术栈：后端使用 Spring Boot 3.x + MyBatis (SQL Server)，前端使用 Vue 3 + Vite + Element Plus，接口请求通过 Axios 调用。通过简单配置，让上传的图片可以通过固定路径直接访问。
- 分工：
  - 史浩天：负责整体架构设计、后端基础配置、静态资源访问和跨域设置，以及前后端联调。
  - 蔡锦：负责用户模块（登录注册、资料修改、头像上传与展示）。
  - 鲁天昱：负责商品模块（列表/详情/搜索/排序、浏览量/状态）。
  - 张婷：负责发布与管理模块（发布/编辑/删除/我的发布、文件上传接口与前端调用）。
- 知识点/创新/难点：
  - 解决前端访问后端接口的跨域问题，配置本地上传目录的访问路径；文件上传既支持本地选择图片，也支持填写图片链接并下载保存到本地。
  - 商品列表/搜索的前端排序与分页展示，浏览量自增。
  - 用户资料更新与头像默认值兜底；商品图片默认占位。
  - SQL Server 上 MyBatis 映射及关联查询（卖家信息联表）。

## 2. 业务需求与基本模块功能

- 用户模块：注册、登录、获取用户详情、更新联系方式与头像，前端导航/设置页同步本地存储。
- 商品模块：按分类/全部列表、详情（含卖家联系方式）、搜索、价格排序、浏览量统计、状态更新（在售/已售）。
- 发布与管理：发布/编辑商品、上传图片（文件或 URL 转存）、删除、我的发布列表、状态切换。
- 基础配置：数据源/Mapper 扫描、上传目录与静态资源映射、前端代理 /api -> 8080。

## 3. 系统设计与实现

- 体系结构：前后端分离，后端提供接口，前端用 Axios 调用；登录信息保存在 localStorage，根据是否登录来控制页面访问；界面使用 Element Plus 的组件搭建。
- 数据库字典（简化）：
  - user(id, username, password, contact, avatarUrl)
  - goods(id, title, price, description, imgUrl, category, userId, status, viewCount) 在多处查询商品时同时带出卖家的昵称、头像和联系方式。
- 核心后端实现：
  - 用户登录/注册/资料更新，默认头像兜底；资料更新仅回写联系与头像。
  - 商品支持新增、查询、修改、删除，以及按条件搜索和排序，支持上架 / 已售状态切换和浏览量统计；删除商品时需要同时匹配商品 id 和当前用户 id，防止删错别人商品。
  - 文件上传时会限制允许的文件类型，只接收指定格式的图片；同时支持填写图片链接，后端把图片下载到本地并返回可访问的地址。
  - 把 /uploads/ 路径指向本地上传目录，浏览器可以直接通过这个路径访问上传的图片。

- 前端关键点：
  - 基于 Vue 3 + Element Plus 编写登录、商品列表和详情、发布和“我的发布”等页面；对 Axios 做了简单封装，开发环境通过 /api 代理后端接口，请求路径更统一；通过路由配置控制只有登录用户才能访问发布和个人相关页面。

### 3.1 重点模块编码实现

- 用户模块：UserController 负责登录/注册/详情/资料更新，UserService 处理用户名唯一校验、默认头像兜底，UserMapper/UserMapper.xml 负责登录、注册、资料更新与按用户名查询。
- 商品模块：GoodsController 提供分类/全部列表、搜索、详情、发布、更新、删除、状态切换与我的商品；GoodsService 兜底商品默认图、浏览量自增；GoodsMapper.xml 进行联表查询卖家信息并支持搜索、状态更新、删除和浏览量计数。
- 发布与管理：Publish.vue 支持新建和编辑商品（可以上传图片或填写图片链接）；MyGoods.vue 展示当前用户发布的商品列表，支持标记已售/重新上架、编辑和删除，已售商品在图片区域会有“已售”标记；后端的 FileController 负责接收上传的文件，检查文件类型是否符合要求，并把文件保存到本地。
- 前端登录状态：Login.vue 登录成功后把 userId 写入 localStorage；Navbar.vue 读取或清空登录信息，并在资料更新后触发刷新事件；Settings.vue 负责加载和提交用户资料，并同步更新本地存储；router/index.js 根据 localStorage 中是否有 userId 来控制 /publish、/my、/settings 等页面是否可以访问。

## 4. 系统测试

- 后端接口：登录/注册、资料更新、文件上传（本地与 URL）、商品新增/编辑/删除/搜索/状态切换/浏览量。
- 前端流程：登录后跳转与路由守卫，列表筛选/排序/搜索，发布-编辑-删除-标记已售，头像与商品图片上传/展示。
- 异常与提示：断网或接口报错时，页面能否弹出友好提示。

## 5. 项目展望（欠缺功能）

- 账号安全：密码加密存储、登录态校验、接口权限校验。
- 商品与用户：补充创建/更新时间字段，支持商品多图，增加评论区/留言，卖家信用评分。
- 列表与搜索：后端分页与排序，搜索高亮/筛选项，更多分类标签。
- 使用体验：骨架屏/加载态、消息提醒（如已售通知）、移动端细节优化。
- 部署与文件：环境配置模板化，上传文件改用云存储，基础日志与错误记录。