

- 实验报告
 - 应用需求介绍
 - 系统设计
 - 系统实现
 - 组员分工
 - 程序演示

实验报告

应用需求介绍

- 面向中小型图书室，提供图书、作者、分类、用户的基础信息管理。
- 支持读者借阅、归还流程以及库存数量的同步变更，便于掌握存量。
- 提供图书搜索、详情查看与书评展示功能，累积读者的反馈信息。
- 后台需具备新增、编辑、删除等管理操作，并对关键动作给予提示。

系统设计

- 整体架构：采用经典 MVC 思路，Flask 充当控制层负责请求路由与业务判断，Jinja2 模板提供视图渲染，psycopg 驱动实现模型层与 openGauss/PostgreSQL 的交互。请求流程依次为浏览器 → Flask 路由 → SQL 查询/事务 → 模板渲染 → HTML 响应；失败时通过 Flash 消息传递错误反馈。
- 模块划分：
 - `app.py`：集中定义蓝本级路由、输入校验、库存更新和借阅/归还事务控制，对数据库操作采用 `with` 语句保障提交或回滚。
 - `config.py`：封装数据库主机、端口、数据库名、账号密码与秘钥，支持从环境变量覆盖，便于部署。
 - `templates/`: `base.html` 提供统一导航与消息区，其余模板 (`index`、`book_detail`、`users`、`authors`、`categories`、`borrowers`) 继承基模板以实现内容块替换。
 - `static/`: `css/style.css` 定义组件样式，`js/` 可放置后续交互脚本，`img/` 预留图标与插画。
- 数据库与 ER 关系：实体与联系如下（参见 `database.sql`）：

- `role(role_id, name)` 与 `user(user_id, role_id, username, email)`: 一对多关系, `role_id` 允许为空并设置 `ON DELETE SET NULL`, 保证删除角色后用户账号仍保留。
 - `author(author_id, name) ↔ book(book_id, author_id, ...)`: 作者与图书一对多, 作者被删除时对应图书的 `author_id` 设为 `NULL`, 便于图书改绑其他作者。
 - `category(category_id, name) ↔ book(category_id)`: 分类与图书一对多, 为搜索与筛选提供维度, 分类删除后图书分类置空。
 - `user` 与 `book` 通过 `borrow(borrow_id, user_id, book_id, borrowed_at, due_at, returned_at)` 形成多对多借阅关系; `ON DELETE CASCADE` 确保删除用户或图书时同步移除借阅记录, 借阅流程中同步维护库存。
 - `user` 与 `book` 通过 `review(review_id, user_id, book_id, rating, content, created_at)` 建立第二条多对多联系, `CHECK(rating BETWEEN 1 AND 5)` 限定评分范围, 评论支持可选内容。
 - 每个实体均设置自增主键, 文本字段施加唯一约束 (如 `role.name`、`category.name`、`book.isbn`) 以防重复。
- **业务与约束策略:** 借阅流程使用事务先校验库存再扣减, 归还时反向加库存并记录时间戳; 新增或编辑图书时若作者名称不存在则自动建档。所有插入操作捕获 `UniqueViolation`, 对用户输入给出错误提示, 分页查询则通过 `LIMIT/OFFSET` 控制结果集。

系统实现

- 书目列表支持关键词检索与分页展示 (每页 5 条), 详情页提供借阅、评论操作及库存状态显示。
- 用户、作者、分类页面提供新增表单与列表视图; 借阅记录页展示最新借阅并支持归还操作。
- 通过 Flask 的 Flash 消息反馈操作结果, 模板统一继承 `base.html` 以复用导航与布局。
- 在 `tests/pagination_test.py` 中提供分页校验脚本, 自动生成测试数据验证首页与借阅页的分页逻辑, 并于结束后清理数据。

组员分工

- 陈登瑞：主导需求梳理、数据库建模、后端路由与借阅流程实现，负责编写 `tests/pagination_test.py` 与部署脚本。
- 何浩：负责界面原型、模板与样式实现，涵盖导航布局、分页展示、表单交互，并整理 `report.md` 与演示素材。

程序演示

- 参考 `README.md` 完成环境准备与依赖安装，初始化数据库后运行 `python app.py` 或 `flask run` 启动应用。
- 浏览器访问 `http://127.0.0.1:5000/` 体验书目搜索、借阅归还、书评提交等流程，截取关键界面截图或录制演示视频。
- 若需对比操作前后，可通过数据库执行 `TRUNCATE TABLE borrow, review, book, author, category, "user" RESTART IDENTITY CASCADE;` 清空记录，再按照流程演示。