

# 图书管理系统

薛振梁 18307130172

IDBS Assignment 3 报告。

## 1. 概况

实现了一个简单的图书管理系统。支持用户管理、添加/删除/更新图书、借阅/返还图书、延长借阅期限、查询图书和借阅历史记录。整个项目分为数据库、后端服务器和前端客户端三个部分。其中后端使用 Go 语言编写，基于 HTTP 协议实现了一个 REST API 架构。前端是一个命令行客户端，使用 Python 3 编写。

项目仓库在 <https://github.com/riteme/catmgr>。简单的环境设置和使用可以参见 GitHub 仓库的 README 文件。

## 2. 数据库

**2.1 数据库模式** 定义了以下几种数据库模式：

```
User(user_id, type_id, name, token)
UserType(type_id, type_name, can_update, can_adduser, can_borrow, can_inspect)
Book(book_id, title, author, isbn, available_count, description, comment)
Record(record_id, user_id, book_id, return_date, borrow_date, deadline, final_deadline)
```

**User** 表用于记录用户信息。其中 **name** 表示用户名，要求数据库中不同的用户的用户名要不同，从而可以用于登录。**token** 是用户密码的 SHA1 值。**type\_id** 是用户类型的 ID。

**UserType** 用于存储预设的用户类型，方便进行权限管理。用户一共有四大权限：

- 更新书籍 (**can\_update**)：表示能否添加/删除书籍，或者更新书籍的信息。
- 添加用户 (**can\_adduser**)：表示能否增加新的用户。
- 借阅书籍 (**can\_borrow**)：表示能否借阅书籍。这里没有限制延长借阅期限和返还书籍的权限，认为这两项权力和借阅书籍是一体的。
- 检查用户 (**can\_inspect**)：一般情况下，我们只允许用户查看自己的借阅记录，不允许其他用户查看。如果授予该权限，则可以查看其他用户的借阅历史。

一般管理员用户会拥有所有的权限，而学生用户只有借阅权限。此外 **type\_name** 只是用于用户类型的注释，没有实际作用。

**Book** 用于存储所有的书籍信息。书籍信息包含标题、作者、ISBN、描述和附言 (**comment**)。附言部分主要用于记录一些书籍的状态信息，比如“书籍丢失”。**Book** 表使用 **available\_count** 记录当且可供借阅的书籍数量。

Record 表记录所有的借阅记录。其中 `borrow_date` 表示书籍借出的日期，`return_date` 表示书籍归还的日期。如果 `return_date` 为空，则说明该记录的书籍还未被归还。`deadline` 表示借阅期限，一般初始为借出日期后 30 天，并且在借阅期限前一周内可以申请延长期限，每次延长 30 天。`final_deadline` 表示最终借阅期限，即无论如何借阅期限不能晚于这个限制，一般为借出日期后 90 天。因此实际上每本书最多延长两次借阅期限。

**2.2 设置数据库** 使用代码仓库中的 `sql/setup.sql` 或 `sql/setup_test.sql` 来初始化仓库。默认的数据库名称为 `library` 或 `library_test`。其中 `sql/setup_test.sql` 包含样例测试数据。

## 3. 后端

**3.1 基本特性** 后端使用 Go 语言编写，用于提供一套基于 HTTP 协议的 REST API，并且负责与数据库进行交互，包含以下几个接口：

- `/new`：增加新的书籍。这里值添加书籍，并不设置书籍的基本信息。
- `/update`：更新书籍的信息。可以更新除书籍 ID 以外的所有信息。通过增加/减少 `available_count` 实现新增/删除图书。
- `/adduser`：新增用户。
- `/show`：索引书籍。支持按图书 ID、ISBN 索引，或者在书籍标题、作者中搜索包含关键字的所有书籍。
- `/list`：检索借阅记录。支持只检索未归还记录和逾期记录。
- `/borrow`：借阅书籍。如果用户有大于 3 条未归还的逾期的借阅记录，则会拒绝图书借阅。
- `/extend`：延长借阅期限。
- `/return`：归还书籍。

以上接口除 `/show` 外，均需要提供用户名/用户 ID 和密码，用于后端进行权限验证。数据交互使用 JSON 编码。目前用户登录信息是直接在中文数据中明文传送的，没有使用 HTTPS，可能存在安全隐患。

代码仓库中 `catmgrd/main.go` 是服务器的主体部分，负责对 HTTP 请求进行处理。`catmgrd/api.go` 是与数据库交互的部分，`catmgrd/api_test.go` 是这一部分的单元测试代码。

**3.2 设置后端** 首先在工作目录下创建 `catmgrd.json` 配置文件，包含以下内容：

```
{
  "username": "root",
  "password": "123456",
  "protocol": "tcp",
  "address": "localhost",
  "port": 3306,
  "database": "library_test"
}
```

这是连接 MySQL 数据库时用到的配置。`username` 和 `password` 为数据库用户的名称和密码。`protocol` 一般为 `tcp`。`address` 和 `port` 指定 MySQL 数据库的地址和端口。`database` 表示要使用的数据库名称。之后使用：

```
make serve
```

运行服务器。默认情况下 `catmgrd` 将监听本地的 10777 端口。

**3.3 单元测试** 使用以下命令运行单元测试：

```
make test
```

在运行单元测试前需要在 `catmgrd` 目录下准备一个 `test_config.json` 配置文件，其内容与 `catmgrd.json` 相同。并且需要准备好测试数据库 `library_test`。

## 4. 前端

**4.1 基本情况** 前端使用 Python 3 编写，提供了一个简单的命令行交互。

```
sudo pip3 install click requests # 依赖项
chmod +x ./catmgr-cli.py         # 执行权限
./catmgr-cli.py --help
```

输出：

```
Usage: catmgr-cli.py [OPTIONS] COMMAND [ARGS]...

Options:
  -v, --verbose  Show more information.
  --help         Show this message and exit.

Commands:
  adduser  Add a new user.
  borrow   Borrow a book.
  extend   Extend deadline.
  list     List borrow history.
  new      Add a new book.
  return   Return a book.
  show     Search for books.
  update   Update book information.
```

具体的操作与后端的接口一一对应。

**4.2 配置文件** 与后端类似，前端可以使用 catmgr.json 文件修改默认设置。

```
{
    "server_url": "http://localhost:10777",
    "user": "root",
    "password": "root"
}
```

如上所示，可以指定后端服务器的地址（默认为 localhost:10777）、设置默认登录的用户和密码。设置默认用户后，当前端提示输入用户名和密码时可以直接确认，从而使用配置文件中的用户名和密码。

注意以上配置不是必须的。

**4.3 用户登录** 大部分命令都需要用户登录。一般情况下按照命令行中的提示输入用户名和密码即可。也可以使用 --user 和 --password 参数传递用户名和密码：

```
./catmgr-cli.py [subcommand] --user ayaya --password 123456 ...
```

上述命令使用用户名“ayaya”和密码“123456”进行登录。

### 4.4 添加新用户

```
./catmgr-cli.py adduser
```

之后按照指示进行用户验证以及设置新用户的用户信息（用户类型、用户名、密码）。成功时会返回新用户的用户 ID。

### 4.5 新增书籍

```
./catmgr-cli.py new
```

成功时返回新增书籍的 ID。该命令不需要提供书籍的信息。后续可以使用 update 命令更新书籍信息。

## 4.6 检索书籍

```
./catmgr-cli.py show --section=title graph
```

上述命令检索所有书名包含“graph”的书籍。除 title 外，section 参数还可以指定为 book\_id、isbn 和 author。

## 4.7 查询借阅历史

```
./catmgr-cli.py list ayaya
```

上述命令列出用户 ayaya 所有的借阅记录，按照时间顺序降序排序。可以使用 --filter=not-returned 来忽略已返还的记录，只显示未返还的记录。或者是 --filter=overdue 来显示未返还且已逾期的记录。

## 4.8 更新书籍

```
./catmgr-cli.py update 1 --diff=-1 --comment="a book lost"
```

上述命令将减少一本书籍 ID 为 1 的书，并且将这本书的附言修改为“a book lost”。此外还可以使用 --title、--author、--isbn 和 --desc（--description 的缩写）来更新对应信息。没有在命令行中指定的项目，不会更新数据库中的内容。

## 4.9 借阅书籍和相关操作

```
./catmgr-cli.py borrow 5  
./catmgr-cli.py extend 57  
./catmgr-cli.py return 57
```

这里 borrow 命令使用的是书籍 ID，而 extend 和 return 用的都是借阅记录的 ID。书籍 ID 可以使用 show 命令查询，借阅记录 ID 可以使用 list 命令查询。