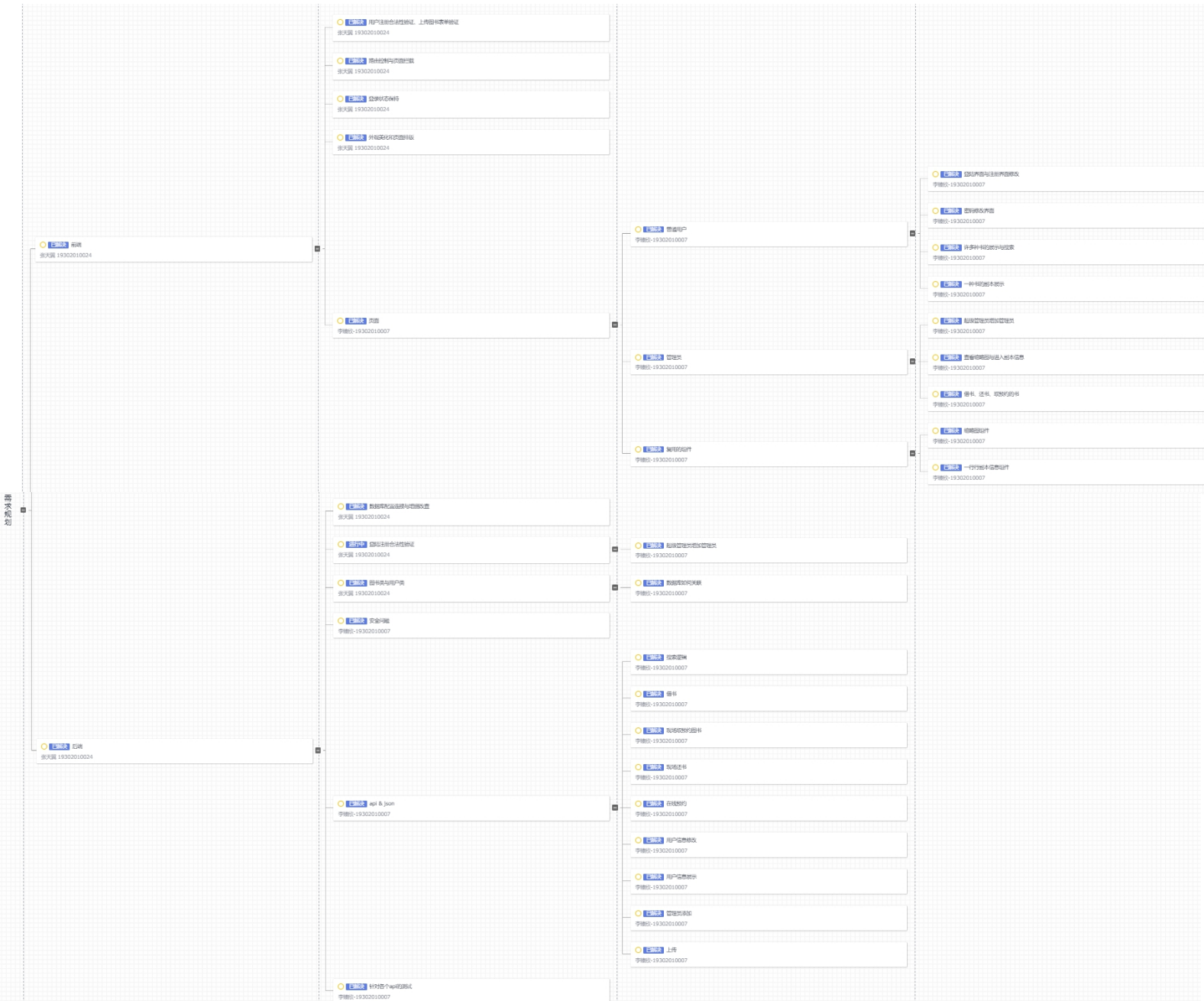


温馨提示：阿里云服务器第一次请求会慢一些

需求规划



二、代码检查结果截图



三、Lab2 截止截图



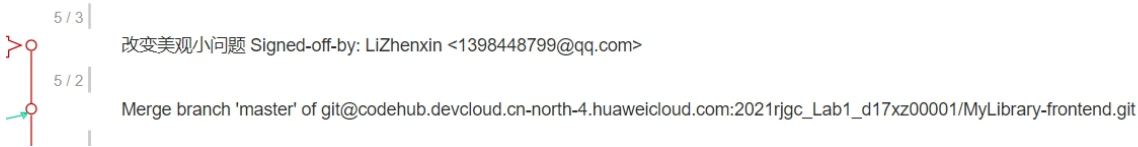
四、Lab3 截止截图

分成前端和后端，此后舍弃原 library 仓库

前端：

标签列表 标签可以标记历史里程碑

标签名	最新提交	更新时间
lab3-finish	cc0d6589 · 改变美观小问题 Signed-off-by: LiZhenxin <1398448799@qq.c...	11分钟前



后端:

标签可以标记历史里程碑

标签名	最新提交	更新时间
lab3-finish	dacaf3c6 · 更新JwtRequestFilter.java	3小时前



五、项目页面截图

用户页（与 lab2 基本相同，新增了诸如个人中心与修改密码）

修改密码

用户名

admin

* 密码

请输入密码

* 修改密码

请输入修改密码

提交

用户信息展示

用户

admin

邮箱

暂无

密码修改

借阅详情:

序号	图书	ISBN	cid	书名	作者	地区	状态	预约时间	借阅时间	归还时间
暂无数据										

修改密码

用户名

admin

* 密码

请输入密码

* 修改密码

请输入修改密码

提交

欢迎您, 管理员

邯郸校区

学生事务

查看书本

图书上新

增加管理员

个人中心

用户界面

退出

主页

今日事务: 暂无

欢迎您, 管理员

邯郸校区

学生事务

书本出借

取预约书

现场借书

书本归还

现场还书

查看书本

图书上新

增加管理员

个人中心

用户界面

现场借书

* 用户名

* 副本编号

	序号	图书	ISBN	cid	书名	作者	地区	状态
暂无数据								

< 1 >

10条/页

共 0 条

前往 1 页

借出选中书本

欢迎您, 管理员

邯郸校区

学生事务

查看书本

图书上新

增加管理员

个人中心

用户界面

退出

个人中心

序号	图书	ISBN	cid	书名	地区	状态	借阅时间	借阅人学号	借阅人工号	借阅负责人	归还时间	归还负责人
1		999-1-999-12345-1	999-1-999-12345-1-001	测试1		borrowed	2021-05-01 17:10:38	19302010007		admin		

< 1 >

10条/页

共 1 条

前往 1 页

六、产品使用说明：

1. 一些小组商量下得到的设定

1.1 新增管理员

用户名：6-32 位的数字或字母

密码：6-32 位的数字或字母

需要超级管理员同时给管理员赋予用户名和密码

1.2 ISBN 的格式

形如 XXX-X-XXX-XXXXX-X 即可

123-1-123-12345-1

1.3 操作记录分为哪些

1. 借阅--借阅时间、借阅人、借阅负责人（管理员）
2. 归还--归还时间、归还负责人（管理员）
3. 预约--预约时间、预约人

1.4 操作记录的查看

对一本单独的书

1. 借阅会消除其预约与归还的消息
2. 预约会消除其借阅与归还的消息
3. 归还不会消除其消息

1.5 对于某一个书本的许多副本的详情页面

1. 普通用户查看书本--一些书本的基本信息，诸如 *ISBN*、*cid*、书名等，以及其状态与所在分馆（若 *borrowed* 则无分馆信息）
 2. 管理员查看书本--所有上述的操作记录
 3. 用户查看个人信息--与自己相关的操作（借阅、预约、归还）的时间，以及书的状态与分馆
 4. 管理员查看个人信息--与自己相关的操作，以及对应的负责人员（借阅、归还，预约与管理员无关）
-

2. 网站使用手册

2.1 用户页与管理员页的分离

1. 普通用户--只能访问用户页面，即导航栏在页面上部的页面，可以修改密码、查看个人信息、查看书本、在点击书本框进入副本表格时可选择预约。
2. 管理员--可以访问用户页面（部分权限不开放，诸如用户的个人信息，普通用户的查看书本、副本表格页），但是可以通过这个页面修改自己的密码，或进入管理员页。
3. 一个特例--超级管理员，都可以访问。
4. 如果想要越级访问，会进入 *notAuthorized* 页面；如果想要乱输 *url*，会进入 *404* 页面

2.2 管理员页的一些操作

由于用户页的书本操作非常简单，点进副本表格页，点击预约按钮即可，接下来介绍管理员如何使用自己的页面。

1. 取预约：输入 **11** 位学号，如果合法且用户有过预约，那么下面的表格会自动展示他预约的书本，选中一些后点按钮即可。
2. 直接借书：此时只能借出 *available*，即在库的书。输入 **11** 位学号，输入 *cid*。每检测到一个合法且库中存在的 *cid* 便会在下方表格自动添加条目，只需要反复输入即可得到一个有很多副本的表格，同样地，选中一些后点击按钮即可。
3. 还书：反复输入 *cid*，自动添加副本，与上一条类似。
4. 图书上新：这和前一个 *lab* 一样。

七、小组实验过程记录

1. 对 Vue 和 Springboot 框架的进一步学习：

在 Lab2 中，小组成员对 vue 和 springboot 框架已经有了一定的了解。但 Lab3 对两种框架的使用都有了更进一步的要求，因此小组成员在进行具体分工之前，首先各自阅读了 Lab3 文档后对可能需要的知识点进行了学习，并对各自工作有了一定的预估。

2. Lab3 项目需求的分析与讨论：

对于需求进行讨论之后，小组确立了此次 lab 所需要实现的各个功能，并对需求中可能产生歧义与不同理解的方面进行统一。同时，由于 Lab3 中后端的实现相对简单，而前端中相关页面的实验较为繁琐，小组对组员的分工进行了重新划分。

3. 对于前后端开发内容的分工与两端接口的约定：

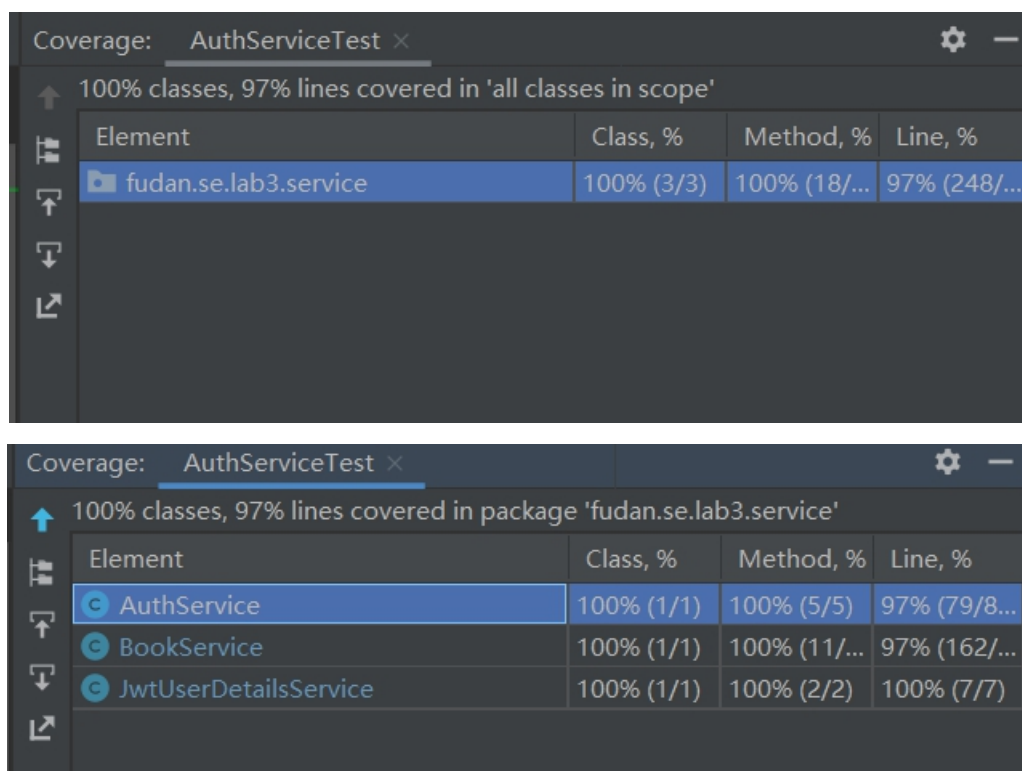
与 Lab2 相同，在对 Lab3 需求进行讨论并完成分工之后，小组在需求规划中建立了思维导图，以便于前端与前端、后端与后端开发同学的合作。前端和后端之间则沿用 Lab2 中的 http 请求。

4. Devcloud 仓库管理：

在明确任务分配之后，小组成员根据自己的任务完成相关代码的编写，并在上传时标注新增、修改和删除的内容，尽可能避免了需要手动合并的情况。同时，在 Lab3 中使用了 tag 功能进行版本管理，以便于项目的部署与检查。

5. 后端的测试：

为保障后端的正确运行，小组对 Service 进行了相应的测试。Lab3 中完成了 AuthService, BookService 与 JwtUserDetailsService 三个 Service 类分别对应了三个测试类对其中的方法进行测试，保障了方法的正确性和覆盖率。



6. 前后端的协同以及服务器的部署：

在完成了修复 bug、根据华为云测试结果修改代码，以及人工调试页面的工作时，由于在 Lab2 中积累的一定经验，Lab3 中对接口的调用统一的较为完善。但在服务器的部署中仍遇到了一些困难。在后端的部署中，发现了无法启动 SpringBoot 服务和无法通过 url 健康测试的问题，在对后端进行修改后进行了修复。

六、小组成员实验总结

张天翼：

图书类的修改

在 Lab3 中根据文档要求对 Book 类进行一定修改，将数据中主键 cid 修改为 String 类型，增加了 isProtoType 属性对书本原型与副本进行区分，并增加了借阅、预约、归还人的学号/工号与进行相应操作的管理员。

上传及预约、借阅、归还逻辑

图书上新功能分为上传原型与增加副本两个方法。上传原型只上传图书的相关属性，但不包含数量、借阅人的信息等，故上传的书本为原型；增加副本方法则负责增加副本的数量，增加的书本具有所有属性。其他方法如预约、归还等则与借阅类似，进行一些基本逻辑判断，如书本状态，预约人信息等，再对图书属性就行相应修改即可。

搜索逻辑

搜索逻辑在 Lab2 的基础上增加了对各种属性的搜索功能，并实现了模糊搜索。同时，根据需求将部分方法的返回调整为书本原型。

后端部署

完成了一些服务器的初始化工作，如安装 jdk, maven, mysql 配置与非 root 用户的添加等。参与了后端部署工作，部署总体按照文档中进行，但仍碰到了 SpringBoot 启动失败、url 健康测试不通过

的问题，在对后端进行相应修正后得到解决。

李臻欣：

小组工作

协助需求的规划与前后端的分工，以及接口的设定。

代码

在具体代码上，主要负责写了用户相关的内容，包括前端与后端。

首先，完成了管理员的增添，以及前后端在分职责登录时的逻辑。此外，前端中设计并增加了管理员界面，配合后端完成了对于书本的取预约、借阅和根据书本信息、人员信息查询的操作功能，后端增加了管理员增加的方法，同时深入地细化了用户与管理员的用户名、密码规则。

部署

参与前端的打包与问题的解决，并且部署到远端的 nginx 中，尝试解决跨域问题。部署所消耗的精力与写代码消耗的精力几乎是等同的，尤其是当出现不知名的问题时，譬如 vue webpack 打包后视图没有被渲染，但是页面逻辑一切正常等等。

陈敏：

前端代码工作：

主要是关于用户相关的内容，修改注册和登录的界面，在注册界面修改用户名格式，登录界面增加管理员的注册功能。同时重构 lab2 的图书展示页面，新增搜索功能，并将图书的展示部分用组件替代，更改展示的风格，实现用户查看借阅的逻辑流程。

在查看副本页面，用组件展示副本，实现预约功能，向后端传送图书信息和节约时间。在用户信息展示页面，从后端获取信息，展示用户的账号、邮箱和用户的预约和借阅的书籍信息。最后新增一个密码修改页面，实现密码的修改，同时也对新密码进行了格式验证。

叶雨昂：

我的工作主要是为 service 类写测试用例，根据 login、register、add、changepassword 等方法各自完成初始化和测试。首先需要了解具体的代码是如何实现的，然后思考可能出现的错误情况，最后检查 line 是否达到了 80% 的覆盖率要求。测试结果是方法本身没有需要修改的地方，因此也无需修改或新增异常处理。