

---

第三版

---

# Aglaia实验室物资管理系统 需求&设计文档

---

计25 李浩达 丁弼原 张敬卿 荣禹



# 目录

1.项目概述.....	2
基本情况.....	3
开发环境.....	3
小组成员.....	3
功能需求概述.....	4
框架与设计模式.....	4
系统结构设计图.....	5
2.用户模块.....	6
用户类型.....	7
用户状态.....	7
用户权限.....	8
基本功能.....	8
数据模型.....	9
前后端接口.....	9
3.实物模块.....	13
借还流程.....	14

实物状态.....	15
基本功能.....	16
数据模型.....	17
前后端接口.....	17
4.计算模块.....	25
借还流程.....	26
基本功能.....	27
数据模型.....	27
前后端接口.....	28
5.日志模块.....	31
基本功能.....	32
数据模型.....	32
前后端接口.....	33



# 项目概述

该项目初衷为帮助清华大学交叉信息研究院管理实物物资和计算资源，本章将主要介绍项目的基本情况、开发环境、小组成员、功能需求、体系结构设计、框架与设计模式等基本信息。



## 基本情况

交叉信息研究院作为重点科研单位，具有多种科研物资，包括为学生提供的学习物品(包括笔记本电脑、移动硬盘等)，以及科研设备(包括服务器以及其他设备)。这些设备的管理是一个繁重的工作。为了能够更好的服务同学，更好的管理所有资源，需要开发一套物资管理系统，通过本系统，普通用户(主要是学生)应该能够更方便的借用各种物资；而管理员可以通过该系统更好的管理所有物资。预期用户为交叉信息研究院的员工和相关负责人。

## 开发环境

前端CSS/JS: Semantic 0.19.3 <http://semantic-ui.com/>

HTTP 服务器: nginx 1.6.2 <http://nginx.org/>

后端语言: Python 3.4 <https://www.python.org/>

代码风格: Pep8 <https://pypi.python.org/pypi/pep8/>

网络框架: Django 1.7 <https://www.djangoproject.com/>

单元测试: Pyunit <https://wiki.python.org/moin/PyUnit>

## 小组成员

李浩达 [lihaoda@163.com](mailto:lihaoda@163.com) 2012011382

族长，后端攻城狮，反人类的架构狮

丁弼原 [thomasding@outlook.com](mailto:thomasding@outlook.com) 2012011381

前端攻城狮，前端接口设计狮

张敬卿 [zhangjqsmiling@gmail.com](mailto:zhangjqsmiling@gmail.com) 2012011380

前端攻城狮，文档设计狮

荣禹 [rongyu9124@hotmail.com](mailto:rongyu9124@hotmail.com) 2012011358

数据库攻城狮，数据库接口设计狮



# 功能需求概述

\*注：详细的功能说明和状态图请见各个模块

## 1. 用户模块

本模块负责用户账户的管理。除了用户的基本信息外，通过“用户类型”、“用户权限”、“用户状态”三个层次来区分用户并限制用户的操作行为。模块将实现账户的注册与登录、用户个人信息和邮箱的认证与修改、用户类型和权限的修改、发送邮件通知等。

## 2. 实物模块

本模块负责实物资源的管理。针对普通用户，可以查看自己借用的物资清单和审核状态，可以清晰查看所有可借用物资的清单，支持快速筛选出自己需要的物资，支持用户提交借用物资的申请，支持用户归还、报修、挂失已经借用的物资。针对管理员，除了普通用户的功能外，支持审核用户借用、归还、报修、挂失的请求，支持邮件推送通知，支持新增实物和实物类型。

## 3. 计算模块

本模块负责计算资源的管理。针对普通用户，可以查看自己借用的资源清单和审核状态，支持填写自己需要的配置信息向管理员提交借用请求，支持对自己借用的资源向管理员提交配置调整请求，支持提交释放申请。针对管理员，除了普通用户的功能外，可以查看所有的用户请求和正在使用中资源的情况，

可以审核用户的请求，可以看到资源上有重要信息的提示，可以手动新增计算套餐，可以查看当前所有的计算套餐，支持邮件推送通知。

## 4. 日志模块

本模块负责记录日志，且仅对管理员开放查询。支持针对用户操作的日志记录，包括用户修改个人信息，用户所有借还操作；支持针对实物资源的日志记录，包括物品借出、归还、维修、挂失/销毁的状态转移和操作人；支持针对计算资源的日志记录，包括资源的归还、释放的状态转移和操作人；支持针对借还申请的日志记录，部分借还过程可能无法造成物资状态的改变（比如管理员拒绝用户的借用请求），该日志会记录包括这种情况在内的所有和借还相关的操作。管理员可以分层次浏览日志，详细日志会按照类别或者资源本身归类，管理员可以点击感兴趣的类别浏览详细日志记录。

# 框架与设计模式

框架模式：MVT(Model Views Template)

编程方式：主要是面向过程编程（POP），用户管理部分使用面向切面编程（AOP）。



# 系统结构设计图





# 用户模块

用户模块负责用户账户、权限的管理，本章将从用户分类、用户状态、基本功能、数据库模型、前后端接口等方面详细介绍我们的设计思路。



## 用户分类\*

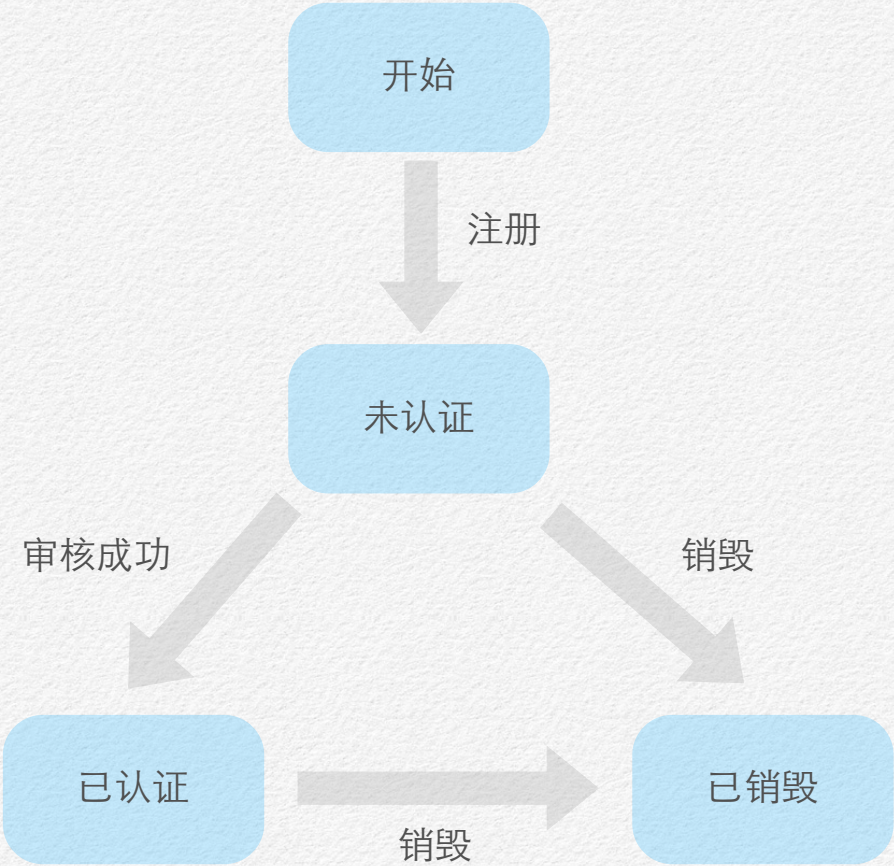
用户类型	类型说明
超级管理员	默认拥有全网站最高权限，包含系统管理、物资管理、普通借还等所有权限；由网站系统内部创建，不开放注册；全网站原则上只能有一个超级管理员，超级管理员可以设置其他用户的权限；
普通管理员	默认拥有部分系统管理、物资管理、普通借还的权限，主要管理物资审核、借还、分配等；由超级管理员直接创建或者升级，不对外开放注册；全网站系统可同时拥有多个普通管理员，彼此之间完全平行，权限相同；
普通用户	默认不具有系统和物资管理的权限；对外开放实名制注册；全网站系统可拥有多个普通用户；

\*注：用户类型与用户权限（后面会详述）是彼此独立的，系统会默认某种类型的用户应该具备某些权限，但是权限可以灵活调整。比如说可以出现个别普通用户的权限与管理员相当的情况，但这应该是极端个例，并且在这种情况下我们建议手动将该普通用户升级为相应的管理员；反之降级亦然。

## 用户状态

用户状态	状态说明
未认证	用户注册完成后或者修改个人信息后将进入此状态，此状态下用户无法正常行使个人的权限，待身份审核通过后可以正常行使权限；
已认证	用户通常状态下应该处于的状态，该状态下可以正常行使权限；
已销毁	用户账户被销毁后处于的状态，该状态不能正常行使权限，该状态不可恢复回未认证或者已认证状态；

## 用户状态转换图





## 用户权限\*

权限名称	权限说明	代码名称*
普通借用	拥有借用物资的权限，所有经过认证的用户均具有该权限	normal
用户身份审核	拥有审核新注册用户的权限，普通管理员和超级管理员默认拥有该权限	user_auth
删除用户	删除某个用户账号的权限（让该用户进入已销毁状态），超级管理员默认拥有该权限	del_user
查看所有用户列表	查看所有用户的列表，普通管理员和超级管理员默认拥有该权限	view_all
修改所有人所在的用户组	修改用户类型（比如有普通用户升级为普通管理员），超级管理员默认拥有该权限	modf_group
修改所有人的敏感信息	尚未实现	modf_key
修改他人的普通信息	尚未实现	modf_normal
修改所有人的权限信息	修改用户的权限（比如让普通用户拥有用户身份审核的权限，虽然我们不建议这样），只有超级管理员拥有默认拥有该权限	modf_perm
实物添加、删除、借用审核	实物物资的管理权限，普通管理员和超级管理员默认拥有该权限	goods_auth
计算资源借用审核	计算资源的管理权限，普通管理员和超级管理员默认拥有该权限	comput_auth

\*注1：用户类型和用户权限彼此是独立的，但不同的用户类型会默认具备一些相应的权限。

\*注2：使用Django自带的Permission Model权限控制模型，content type均设置为account。

## 基本功能

功能	功能说明	权限
用户注册	新用户注册新账户	所有
邮箱验证	新用户验证自己的邮箱	所有
身份审核	管理员审核新用户的真实身份	用户身份审核
销毁账户	管理员销毁用户账户	删除用户
修改个人信息	用户自己或者管理员修改相关用户的个人信息	用户自己；修改他人的敏感/普通信息
用户类型修改	管理员修改用户的用户类型（比如将普通用户升级为管理员）	修改所有人所在的用户组
用户权限修改	管理员修改用户的用户权限（比如让普通用户具备身份审核的权限，虽然我们不建议这样）	修改所有人的权限信息
邮件通知	管理员可以想指定用户发送邮件	查看所有用户列表



# 数据模型

```
class Account (models.Model)
```

类说明： 用户的数据模型，储存了用户账户所需的信息

成员变量：

变量名	变量类型	变量说明
real_name	string	用户真实姓名
department	Department	用户所在部门，自定义的model
tel	string	用户联系方式
status	string(Django.choices)	用户状态
user	User	Django自带的User类
school_id	int	用户学工号
email_auth	Boolean	用户是否通过验证
email_hash	string	用于验证用户的hash码

# 前后端接口

## 登陆界面

前端模板： aglaia/templates/signin.html

后端链接： account.views.show\_signin

### AJAX OPERATION:

1. account.views.do\_signin
- 实现功能： 对用户名和密码进行验证，并完成登陆
- 提交方法： POST
- 传入参数：
  - *username* 输入的用户名
  - *password* 输入的密码
- 返回字符串：
  - ‘ok’ 登陆成功，会重定向到 *account.views.account\_home* 页面
  - ‘mismatch’ 用户名密码错误，登陆失败
2. account.views.do\_verify\_username
- 实验功能： 验证指定用户名是否可用
- 提交方法： POST
- 传入参数：
  - *username* 待验证的用户名
- 返回字符串：
  - ‘ok’ 用户名可用
  - ‘used’ 用户名已经被使用
  - ‘invalid’ 用户名格式错误



### 3. account.views.do\_signup

实验功能：注册新用户

提交方法：POST

传入参数：

- *username* 用户名
- *password* 明文密码
- *realname* 真实姓名
- *number* 学工号
- *tel* 联系电话
- *email* 电子邮件
- *depts* 所在部门（以ASCII逗号分隔的字符串列表） --尚未实现

返回字符串：

- 'ok' 用户名可用
- 'failed' 注册失败

### 4. account.views.do\_lookup\_depts

\*注：尚未实现，开发过程中认为不必要，但接口保留

实验功能：根据指定的搜索条件搜索部门名称

提交方法：POST

传入参数：

- *keyword* 搜索字符串

返回JSON：

```
{
  depts: ['dept1', 'dept2', 'dept3', 'dept4', 'dept5', ...]
}
```

## 用户管理界面

前端模板：aglaia/templates/allaccounts.html

后端链接：account.views.show\_all\_accounts

RENDER CONTEXT (JSON):

- *curpage* 当前所在页面的名字
- *num\_accounts* 该页显示的所有用户的数量
- *num\_pages* 为了显示所有账户所需要的总页数（最小为1）
- *cur\_index* 当前所在页码（从1算起）
- *accounts* 当前页所需要显示的账户列表JSON
  - ✓ *username* 用户名
  - ✓ *realname* 真实姓名
  - ✓ *number* 学工号
  - ✓ *type* 用户类型。可能取值为 'normal', 'manager', 'supervisor'
  - ✓ *email* 电子邮件地址
  - ✓ *email\_verified* 电子邮件是否已经被认证。可能取值为 'true' 和 'false'
  - ✓ *tel* 联系电话
  - ✓ *depts* 所在部门。字符串列表，每个元素是一个部门的名称。
  - ✓ *status* 该用户对象所在状态
    - 可能的取值为 'unauth'(未认证), 'auth'(已认证), 'deleted'(已销毁)
- *all\_perms* 所有可能存在的权限的字符串列表
  - 这个列表中权限的下标和每个账号的权限拥有情况的下标应该一致
- *filter\_type* 用户类型筛选条件
  - 可选值为 'all'(所有用户), 'unauth'(未认证用户), 'auth'(已认证用户)
- *filter\_dept* 部门筛选条件。可选值为部门名称或者None --尚未实现



## AJAX OPERATION:

### 1. account.views.do\_approve\_account

实现功能：实名认证审核通过

提交方法：POST

传入参数：

- *username* 待审核用户名

返回字符串：

- 'ok' 操作完成
- 'denied' 没有这个操作权限或者用户名不存在

### 2. account.views.do\_disapprove\_account

\*注：前端尚未实现，开发过程中认为该功能不是很必要

实验功能：实名认证审核驳回

提交方法：POST

传入参数：

- *username* 待审核的用户名

返回字符串：

- 'ok' 操作完成
- 'denied' 没有这个操作权限或者用户名不存在

### 3. account.views.do\_set\_manager

实现功能：将某个用户设置为普通管理员

提交方法：POST

传入参数：

- *username* 该用户的用户名

返回字符串：

- 'ok' 操作完成
- 'denied' 没有这个操作权限或者用户名不存在

### 7. account.views.do\_set\_permissions

实现功能：修改某个账户的权限

提交方法：POST

传入参数：

- *username* 该用户的用户名
- *permissions* 一个数字串，每位数字按序和all\_perm权限名称相对
  - 数字取值为0（无该权限）或者1（有该权限）

返回字符串：

- 'ok' 操作完成
- 'denied' 没有这个操作权限或者用户名不存在

## 页面基本模板

\*说明：所有以{% extends "base.html" %}开头的HTML文件均继承该模板（除了signin.html等少数页面），该模板设计了各个页面基本的布局（包括菜单栏、右侧隐藏个人信息栏、左边栏、中间主体内容等），因为其中包含了显示和修改个人信息用的右侧个人信息栏，所以将该页面放在用户模块中。

前端模板：aglaia/templates/base.html

RENDER CONTEXT (JSON):

- *user* 当前用户对象JSON
  - ✓ *username* 用户名
  - ✓ *realname* 真实姓名
  - ✓ *number* 学工号
  - ✓ *type* 用户类型
    - 可能取值为'normal'普通用户,'manager'普通管理员,'supervisor'超级管理员



✓ *email* 电子邮件地址

✓ *email\_verified* 电子邮件是否已经被认证, 可能取值为 *'true'* 和 *'false'*

✓ *tel* 联系电话

✓ *depts* 所在部门, 字符串列表, 每个元素是一个部门的名称

✓ *permissions* 拥有权限列表, 每个元素是一个权限的描述

✓ *status* 该用户对象所在状态

- 可能的取值为 *'unauth'* (未认证), *'auth'* (已认证), *'deleted'* (已销毁)

## AJAX OPERATION

### 1.account.views.do\_modify\_account

实现功能: 修改不敏感的账户信息

提交方法: POST

传入参数:

- *tel* 新电话号码
- *depts* 新所在部门 (以ASCII逗号分隔的字符串)
- *realname* 新真实姓名 (未认证用户可以更改)
- *number* 新学工号 (未认证用户可以更改)

返回值字符串:

- *'ok'* 修改成功
- *'invalid'* 新信息的不符合格式
- *'denied'* 没有权限
  - 如未登录用户访问这个URL, 已认证用户不能更改姓名等

### 2.account.views.do\_modify\_email

实现功能: 修改电子邮件, 新电子邮件需要经过验证

提交方法: POST

传入参数:

- *email* 新电子邮件地址

返回值字符串:

- *'ok'* 修改成功
- *'invalid'* 电子邮件格式错误
- *'denied'* 没有权限 (例如未登录用户访问这个URL)

### 3.account.views.do\_modify\_password

实现功能: 修改密码

提交方法: POST

传入参数:

- *old\_password* 旧密码
- *new\_password* 新密码

返回值字符串:

- *'ok'* 修改成功
- *'mismatch'* 旧密码输入错误
- *'invalid'* 新密码不符合密码格式要求
- *'denied'* 没有权限 (例如未登录用户访问这个URL)



# 实物模块

# 3

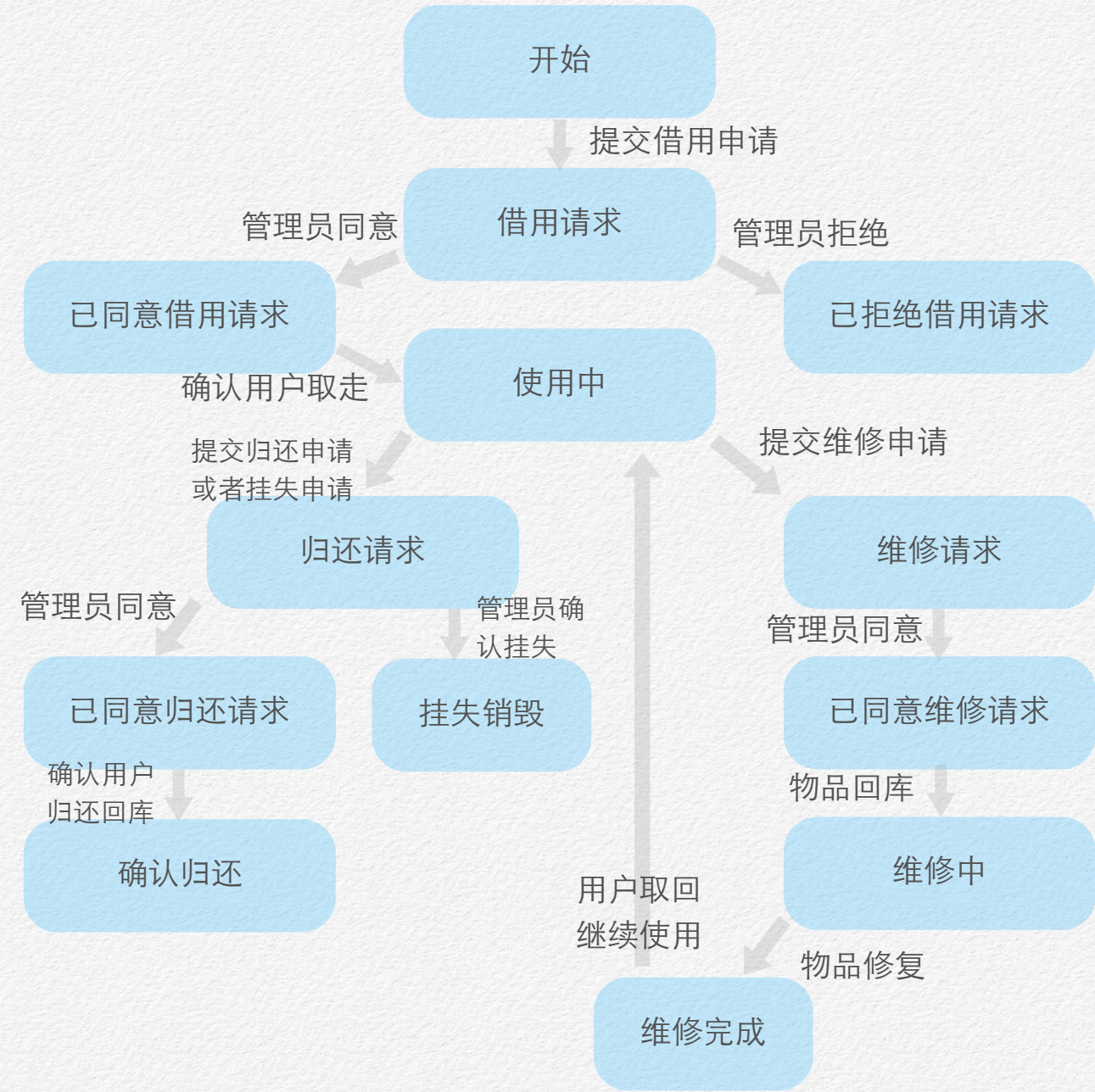
实物模块负责实物资源借用归还、审核请求、新增实物的管理，本章将从借还流程、实物状态、基本功能、数据模型、前后端接口等详细介绍我们的设计思路。



# 借还流程

状态	状态说明
借用请求	用户提交借用实物的请求后进入该状态，等待管理员审核
已拒绝借用请求	拒绝用户的借用申请，借用流程结束
已同意借用请求	同意用户的借用申请，此时物品尚未出库，等待用户前来领取
使用中	用户取走自己借用的物资，正在使用中
归还请求	用户提交归还、挂失实物的请求后进入该状态，等待管理员审核；管理员可以选择物资正常归还回库，也可以选择挂失物品，进入线下赔偿流程
已同意归还请求	同意用户的归还申请，此时物品尚未回库，等待用户前来归还
确认归还	用户归还自己借用的物资，物资已经回库
挂失销毁	管理员审核确认物品丢失，物品进入被销毁的状态，借用流程结束
维修请求	用户提交物品维修的申请，待管理员审核
已同意维修请求	管理员同意维修物品，此时物品尚未回库维修，等待用户送来维修
维修中	用户将物品送回库，物品正在维修
维修完成	物品维修完成，等待用户前来取回继续使用

# 借还流程状态转换图

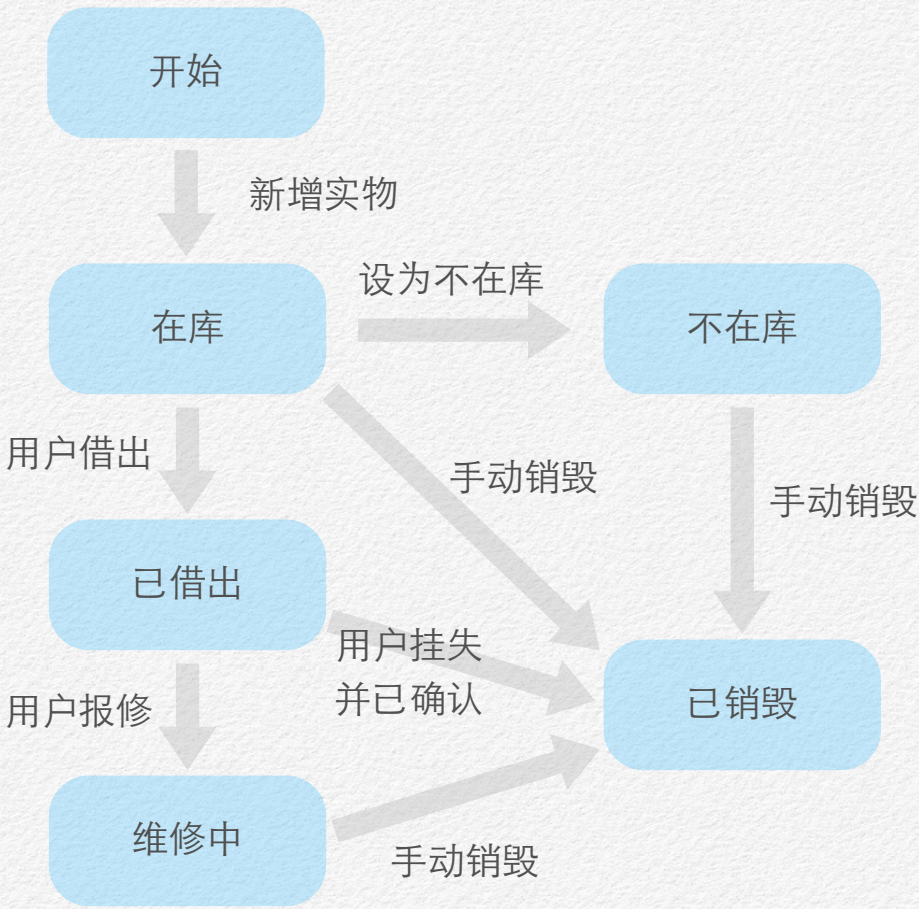




# 实物状态

状态	状态说明
在库	新增的实物默认进入该状态，该状态下的实物资源可以被用户借用
已借出	该状态下的实物物资已经被用户借出，其他用户不能再次借用该物资
维修中	特指已经借出的物资由用户提出维修申请并经过管理员确认回库维修后，进入该状态；原本在库的物资如果需要维修，应设置为‘不在库’的状态
不在库	实物物资没有被任何人借出，但是因为特殊原因不能被借用（如损坏等），管理员可以手动设置物资为‘不在库’状态，也可以设置回‘在库’状态
已销毁	用户挂失或者管理员直接销毁实物资源，进入该状态的实物资源不可恢复，也就是说实物资源的生命以此终结

# 实物状态转换图





# 基本功能

功能	功能说明
个人借还主页	用户可以看到所有自己的请求当前的状态，以及自己借用、报修物资的状态，并可以进行相关操作（后面）
所有实物列表	罗列所有物资，以及它们的状态，支持进行相关操作（后面）
用户提交借用请求	用户提交借用物资的申请（物资列表上进行操作）
用户提交归还请求	用户提交归还物资的申请（个人借还主页上进行操作）
用户提交挂失请求	用户提交挂失物资的申请（个人借还主页上进行操作）
用户提交维修请求	用户提交维修物资的申请（个人借还主页上进行操作）
手动设置实物状态	管理员可以手动将物资设为‘在库’或者‘不在库’的状态（物资列表上操作）
手动销毁实物	管理员可以手动销毁物资（物资列表上操作）
新增实物/实物类型	管理员新增实物资源和实物类型，填写相关属性信息
条件快速筛选实物	用户可以设置筛选条件（名称、实物状态、实物类型）来快速获取自己想要的实物资源
管理员审核请求	管理员审核用户的申请，完成借还流程
管理员维护维修中实物状态	管理员手动维护物资维修的状态，维修完毕后重新交给用户使用
邮件通知	用户提交请求后邮件通知管理员

# 数据模型

class GType(models.Model)

类说明： 用于记录一类物品的属性

成员变量：

变量名	变量类型	变量说明
id	int	id
name	str	GType的名字(最长50字节)
pro_names	str	所有的属性名构成的字符串，不同属性名之间用','隔开

class Goods(models.Model)

类说明： 用于记录一类属性相同的物品的信息

成员变量：

变量名	变量类型	变量说明
id	int	id
name	str	名字(不超过50个字节)
gtype	models.ForeignKey(GType)	对应到GType的多对一关系
pro_values	str	存放物资属性值的字符串,不同属性值之间用sep隔开(sep的定义见下面)



class Borrow(models.Model)

类说明： 用于记录物资借用的信息  
成员变量：

变量名	变量类型	变量说明
id	int	id
account	models.ForeignKey(Account)	物品借用者的account
single	models.ForeignKey(Single)	借用的物品
status	str	物品借用所处的状态
user_note	str	物品借用者的说明
manger_note	str	管理员的说明

class Single(models.Model)

类说明： 用于记录单个物品的信息  
成员变量：

变量名	变量类型	变量说明
sn	str	物品的SN号(不超过60个字节)
goods	models.ForeignKey(Goods)	对应到一个Goods实例的多对一关系
status	str(django.choices)	物品所处的状态
user_name	str	物品借用者的名字(不超过50个字节)
note	str	记录(不超过1000个字节)
manger_note	str	管理员的说明

前后端接口

新增实物页面

前端模板： aglaia/templates/add\_goods.html

后端链接： goods.views.show\_add\_goods

RENDER CONTEXT (JSON):

- base.html要求的RENDER CONTEXT(请见用户模块)
- type\_list 可选类型列表

AJAX OPERATION:

1. goods.views.do\_type\_props

实现功能： 列出某一个type下的每个属性的属性名称

提交方法： POST

传入参数：

- type\_name 类别名字

返回JSON:

- pro\_names 从属性1到属性32的名字列表，若无该属性值为空串

表单ACTION

1. goods.views.add\_goods

实现功能： 添加新物品，完成后应该回到goods\_list.html页面

提交方法： POST

- name 实物资源名称
- type\_name 实物资源所属类别的名字
- pro1\_value 属性1的值
- ...
- pro32\_value 属性32的值
- sn 实物资源sn号
- quantity 实物数量



## 新增实物类型页面

前端模板: `aglaia/templates/add_type.html`

后端链接: `goods.views.show_add_type`

RENDER CONTEXT (JSON):

- `base.html`要求的RENDER CONTEXT(请见用户模块)

表单ACTION

1. `goods.views.add_type`

实现功能: 添加一个新的物品类别

提交方法: POST

- `type_name` 实物资源所属类别的名字
- `pro1_name` 属性1的名字, 如果为空串表示不使用这个属性。
- ...
- `pro32_name` 属性32的名字

## 借还管理页面

\*说明: 其中不仅包含实物资源的借还管理, 也包含了计算资源的借还管理, 计算模块详述请见下一章。本页面信息量较大, 接口较多, 高能预警。

前端模板: `aglaia/templates/borrow.html`

后端链接: `goods.views.show_borrow`

RENDER CONTEXT (JSON):

- `curpage` 当前页面名称。对于`borrow.html`取值为'`borrow`'
- `num_res_used` 正在使用的计算资源数量
- `num_res_borrow` 正在申请借用的计算资源数量
- `num_res_release` 正在申请归还的计算资源数量
- `num_res_modif` 正在申请修改的计算资源数量

- `num_goods_used` 正在使用的实物资源数量
- `num_goods_borrow` 正在申请借用的实物资源数量
- `num_goods_return` 正在申请归还的实物资源数量
- `package_list` 可以使用的套餐列表
- `borrowing_list` 正在借用审核的计算资源列表JSON
  - ✓ `id` 该请求id
  - ✓ `name` 该请求的名字
  - ✓ `note` 管理员与用户相互传递的消息
  - ✓ `package_name` 套餐名字
  - ✓ `sn` 计算资源SN号
  - ✓ `type` 计算资源类型, 字符串: '`real`' (实体机), '`virtual`' (虚拟机)
  - ✓ `cpu` CPU描述
  - ✓ `memory` 内存大小
  - ✓ `disktype` 硬盘类型
  - ✓ `disk` 硬盘大小
  - ✓ `ip` 服务器网络地址
  - ✓ `os` 操作系统
  - ✓ `login` 登录用户名
  - ✓ `initial_password` 初始密码
  - ✓ `period` 资源时间限制
- `inuse_list` 正在使用的计算资源列表
  - ✓ `id` 该请求的编号
  - ✓ `name` 该请求的名字
  - ✓ `package_name` 套餐名字
  - ✓ `sn` 计算资源SN号
  - ✓ `note` 管理员与用户相互传递的消息
  - ✓ `type` 计算资源类型, 字符串: '`real`' (实体机), '`virtual`' (虚拟机)



- ✓ cpu CPU描述
- ✓ memory 内存大小
- ✓ disktype 硬盘类型
- ✓ disk 硬盘大小
- ✓ ip 服务器网络地址
- ✓ os 操作系统
- ✓ login 登录用户名
- ✓ initial\_password 初始密码
- ✓ period 资源时间限制
- modifying\_list 正在修改审核资源列表
  - ✓ id 该请求的编号
  - ✓ name 该请求的名字
  - ✓ note 管理员与用户相互传递的消息
  - ✓ package\_name 套餐名字
  - ✓ sn 计算资源SN号
  - ✓ type 计算资源类型，字符串：'real'（实体机），'virtual'（虚拟机）
  - ✓ cpu CPU描述
  - ✓ memory 内存大小
  - ✓ disktype 硬盘类型
  - ✓ disk 硬盘大小
  - ✓ ip 服务器网络地址
  - ✓ os 操作系统
  - ✓ login 登录用户名
  - ✓ initial\_password 初始密码
  - ✓ period 资源时间限制
- returning\_list 正在释放审核的计算资源列表
  - ✓ id 该请求的编号

- ✓ name 该请求的名字
- ✓ note 管理员与用户相互传递的消息
- ✓ package\_name 套餐名字
- ✓ sn 计算资源SN号
- ✓ type 计算资源类型，字符串：'real'（实体机），'virtual'（虚拟机）
- ✓ cpu CPU描述
- ✓ memory 内存大小
- ✓ disktype 硬盘类型
- ✓ disk 硬盘大小
- ✓ ip 服务器网络地址
- ✓ os 操作系统
- ✓ login 登录用户名
- ✓ initial\_password 初始密码
- ✓ period 资源时间限制
- goods\_borrowing\_list 正在审核借用申请的物品列表（尚未通过审核）
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号
  - ✓ name 被借出物品名
  - ✓ note 管理员消息或者用户说明
- goods\_borrow\_pending\_list 已经审核通过的借用申请的物品列表
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号
  - ✓ name 被借出物品名
  - ✓ note 管理员消息或者用户说明
- goods\_borrow\_failed\_list 审核被拒绝的物品列表
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号



- ✓ name 被借出物品名
- ✓ note 管理员消息
- goods\_inuse\_list 已经借用物品列表（使用中）
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号
  - ✓ name 被借出物品名
  - ✓ note 管理员消息或者用户说明
- goods\_returning\_list 正在审核归还物品列表（审核尚未通过）
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号
  - ✓ name 被借出物品名
  - ✓ note 管理员消息或者用户说明
- goods\_repair\_list 待审核维修申请的物品列表
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号
  - ✓ name 被借出物品名
  - ✓ note 管理员消息或者用户说明
- goods\_repairing\_list 正在维修的物品列表
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号
  - ✓ name 被借出物品名
  - ✓ note 管理员消息或者用户说明
- goods\_repaired\_list 待审核维修申请的物品列表
  - ✓ id 该借用请求的编号
  - ✓ sn 借用物品sn号
  - ✓ name 被借出物品名
  - ✓ note 管理员消息或者用户说明

## AJAX OPERATION:

### 1.computing.views.do\_borrow\_request

实现功能：提交计算资源借用申请

提交方法：POST

传入参数

- type 计算资源类型，字符串：'real'（实体机）,'virtual'（虚拟机）
- name 该请求的名字
- package\_name 套餐名字
- cpu CPU描述
- memory 内存大小
- disktype 硬盘类型
- disk 硬盘大小
- os 操作系统
- reason 用户的申请说明
- period 资源时间限制（应该是expire时间？）
- login 用户名
- initial\_password 初始密码

返回值为字符串

- 'ok' 操作完成
- 'denied' 操作失败

### 2.computing.views.do\_return\_request

实现功能：提交计算资源归还申请

提交方法：POST

传入参数

- id 计算资源请求id

返回值为字符串

- 'ok' 操作完成
- 'denied' 操作失败



### 3.computing.views.do\_modif\_request

实现功能：提交计算资源修改申请

提交方法：POST

传入参数

- id 计算资源请求id
- period 修改后的截止时间
- reason 修改原因

返回值为字符串

- 'ok' 操作完成
- 'denied' 操作失败

### 4.computing.views.do\_get\_package

实现功能：获得某个套餐下的属性值

提交方法：POST

传入参数

- name 套餐名称

返回值为JSON

- type 计算资源类型，字符串：'real'（实体机）,'virtual'（虚拟机）
- cpu CPU描述
- memory 内存大小
- disktype 硬盘类型
- disk 硬盘大小
- os 操作系统

### 表单ACTION

#### 1.goods.views.do\_return\_goods

实现功能：提交归还实物资源的申请（待管理员审核）

提交方法：POST

传入参数

- id 实物资源id

- note 归还说明

#### 2.goods.views.do\_miss\_goods

实现功能：挂失实物资源（待管理员审核）

提交方法：POST

传入参数

- id 实物资源id
- note 挂失说明

#### 3.goods.views.do\_repair\_goods

实现功能：提交维修实物资源申请（待管理员审核）

提交方法：POST

传入参数

- id 实物资源id
- note 挂失说明

#### 4.computing.views.do\_set\_flag

实现功能：标记一个计算资源是否含有重要数据

提交方法：POST

传入参数

- id 同意的计算资源申请编号
- flag 是否包含重要数据,'true' or 'false'



## 实物资源列表页面

说明：该页面支持所有实物资源查询，支持借用操作

前端模板：aglaia/templates/goods\_list.html

后端链接：goods.views.show\_list

显示FUNCTION（GET方法获得如下参数，用于筛选实物）

- status 需要显示处于哪些状态的物品
  - 空串（显示除了已销毁的全部）、all（显示除了已销毁的全部）、
  - available(在库物品)、unavailable(不可用物品)、
  - destroyed(已销毁物品)、borrowed(已借出物品)
- name 实物资源的名字（支持前缀匹配）
- type 筛选的实物资源类型（如电脑、投影仪），空串表示没有限制
- pro1\_value 在筛选类型type确定的情况下有意义：属性1的值
  - 属性1的名字是由类别确定的
- ...
- pro32\_value 属性32的值

RENDER CONTEXT (JSON):

- base.html需要的所有context
- goods\_list 当前筛选状态下的实物资源列表。每个元素有如下属性：
  - ✓ id 实物资源id
  - ✓ name 实物资源的名字
  - ✓ type\_id 实物资源所属类型id
  - ✓ type\_name 实物资源所属类型名字
  - ✓ prop 属性表，共三十二个元素：
    - \*pro\_name 属性i的名字
    - \*pro\_value 属性i的值
  - ✓sn 实物资源sn号

✓ status 实物资源所处状态

- \*'available'(在库) 'unavailable'(不在库)
- \*'borrowed'(已借出) 'destroyed'(已销毁)

✓ user\_id 使用者id

✓ user\_name 使用者用户名

✓ note 处在不可用状态时的描述(仅对不可用物品有意义)

• type\_list 所有物品类型

表单 ACTION

1.goods.views.do\_borrow

实现功能：拥有借用权限的用户提交某个物品的借用申请

提交方法：POST

- id 实物资源id
- note 申请理由

2.goods.views.do\_set\_unavailable

实现功能：拥有权限的用户将某个可用物品设为不可用，并给出理由

提交方法：POST

- id 实物资源id
- note 描述说明

3.goods.views.do\_destroy

实现功能：拥有权限的用户将某个可用物品销毁

提交方法：POST

- id 实物资源id

AJAX OPERATION

1.goods.views.do\_type\_props

实现功能：列出某一个type下的每个属性的属性名称

提交方法：POST



传入参数:

- type\_name 类别名字

返回JSON:

- pro\_names 从属性1到属性32的名字列表，空串代表无该属性

## 实物资源请求审核页面

前端模板: aglaia/templates/goods\_manage.html

后端链接: goods.views.show\_manage

RENDER CONTEXT (JSON):

- base.html要求的render context
- borrow\_requests 借用请求列表
  - ✓ id 请求id
  - ✓ goods\_name 借用物品的名称
  - ✓ sn 借用物品SN
  - ✓ borrower\_name 借用者用户名
- return\_requests 归还请求列表
  - ✓ id 请求id
  - ✓ goods\_name 借用物品名称
  - ✓ sn 借用物品SN
  - ✓ borrower\_name 借用者用户名
- borrow\_pending\_requests 已同意借用的请求列表
  - ✓ id 请求id
  - ✓ goods\_name 借用物品的名称
  - ✓ sn 借用物品SN
  - ✓ borrower\_name 借用者用户名
  - ✓ note 管理员消息

- return\_pending\_requests 已同意归还的请求列表

- ✓ id 请求id
- ✓ goods\_name 借用物品名称
- ✓ sn 借用物品SN
- ✓ borrower\_name 借用者用户名
- ✓ note 管理员消息

- repair\_requests 待审核维修申请的物品列表

- ✓ id 请求id
- ✓ goods\_name 借用物品名称
- ✓ sn 借用物品SN
- ✓ borrower\_name 借用者用户名
- ✓ note 消息

- repairing\_requests 正在维修物品列表

- ✓ id 请求id
- ✓ goods\_name 借用物品名称
- ✓ sn 借用物品SN
- ✓ borrower\_name 借用者用户名
- ✓ note 消息

- repaired\_requests 已维修完成物品列表

- ✓ id 请求id
- ✓ goods\_name 借用物品名称
- ✓ sn 借用物品SN
- ✓ borrower\_name 借用者用户名
- ✓ note 消息

表单 ACTION

1.goods.views.do\_accept\_borrow

实现功能: 接受某个借用申请



提交方法: POST

- id 请求id
- note 管理员信息(比如去何处取物资)

## 2.goods.views.do\_reject\_borrow

实现功能: 拒绝某个借用申请

提交方法: POST

- id 请求id
- note 管理员信息

## 3.goods.views.do\_finish\_borrow

实现功能: 完成某个借用申请 (用户已经取走)

提交方法: POST

- id 请求id

## 4.goods.views.do\_accept\_return

实现功能: 接受归还申请

提交方法: POST

- id 请求id
- note 管理员信息 (比如去何处归还)

## 5.goods.views.do\_finish\_return

实现功能: 完成某个借用申请

提交方法: POST

- id 请求id

## 6.goods.views.do\_accept\_repair

实现功能: 接受维修申请

提交方法: POST

- id 请求id
- note 管理员信息

## 7.goods.views.do\_start\_repair

实现功能: 物品进入维修中 (物资已经回库)

提交方法: POST

- id 请求id
- note 管理员信息

## 8.goods.views.do\_finish\_repair

实现功能: 物资维修完毕

提交方法: POST

- id 请求id
- note 管理员信息

## 9.goods.views.do\_return\_repair

实现功能: 维修后物品归还借用者

提交方法: POST

- id 请求id
- note 管理员信息



# 计算模块

# 4

计算模块负责实体和虚拟计算资源的借还管理，本章将从借还流程、基本功能、数据模型、前后端接口等方面介绍计算模块



借还流程\*

状态	状态说明
借用审核	用户提交借用计算资源的申请，等待管理员审核申请
借用审核失败	管理员拒绝用户的计算资源申请，借用流程结束
使用中	管理员同意用户的计算资源申请，配置好用户所需要的资源，开放给用户使用
归还审核	用户提交使用完毕，提交释放计算资源的请求，待管理员审核
已释放	管理员同意用户的计算资源归还请求，释放计算资源，借用流程结束
配置调整审核	用户申请计算资源配置调整，等待管理员审核
配置调整审核失败	管理员拒绝用户的配置调整申请
配置调整审核成功	管理员同意用户的配置调整申请，并且按用户要求调整完毕

\*注：计算资源的状态和流程一一对应较好，因此不在赘述计算资源的状态。

借还流程状态转换图





# 基本功能

功能	功能说明
计算套餐浏览	管理员可以浏览当前所有计算套餐（给用户提供计算资源配置的参考，方便用户快速选择配置）
添加新套餐	管理员可以手动添加新的计算套餐
申请计算资源	用户提交借用资源的申请
审核借用申请	管理员审核借用申请
申请归还	用户提交释放资源的申请
审核归还申请	管理员审核释放申请，目前只能“同意”
查看使用中资源	管理员可以查看所有正在使用中的计算资源；用户可以查看自己正在使用的计算资源
申请配置调整	用户可以申请调整自己正在使用的计算资源的配置
审核配置调整	管理员审核用户的配置调整请求，可以拒绝请求，或者同意请求并手动更改配置
邮件通知	用户提交请求后邮件通知管理员

# 数据模型

class Computing(models.Model)

类说明： 存储了计算资源所需的信息  
成员变量：

变量名	变量类型	变量说明
id	int	计算资源id
pc_type	str(django.choices)	虚拟机还是实体机
cpu	str	cpu描述
memory	int	内存大小，单位是MB
disk	str(django.choices)	硬盘大小，单位是GB
disk_type	datetime.date	固态硬盘还是机械硬盘
os	str	计算资源的操作系统
sn	str	部分计算资源的sn号
expire_time	str	资源结束使用的时间点
login	str [0-9a-zA-Z]{6,20}	登录计算资源的用户名
password	str [0-9a-zA-Z]{6,20}	登录计算资源的密码
status	str(django.choices)	计算资源当前所处状态
account	ForeignKey(Account)	和Account建立多对一关系
note	str	备注(最长500字节)
address	str	计算资源网络地址
flag	Boolean	计算资源是否含有重要数据



class Package(models.Model)

类说明： 计算资源套餐模型，存储了计算资源套餐所需的信息  
成员变量：

变量名	变量类型	变量说明
id	int	套餐id
name	str	套餐的名字
cpu	str	cpu描述
pc_type	str(django.choices)	虚拟机还是实体机
memory	int	内存大小，单位是MB
disk	str(django.choices)	硬盘大小，单位是GB
disk_type	datetime.date	固态硬盘还是机械硬盘
os	str	计算资源的操作系统

class Server(models.Model)

\*注： 接口保留但用户不再需要该模块

类说明： 存储了服务器所需的信息  
成员变量：

变量名	变量类型	变量说明
id	int	服务器id
status	str(django.choices)	服务器状态
description	str	服务器描述
name	str	服务器名字
configuration	str	服务器配置

前后端接口\*

\*说明： 实物模块中的“借还管理页面”中含有部分计算资源的内容，详细接口信息请见实物模块->前后端接口->借还管理页面

计算资源请求审核页面

前端模板： aglaia/templates/calc\_resource.html

后端链接： goods.views.show\_calc\_resource

RENDER CONTEXT (JSON):

- base.html要求的RENDER CONTEXT(请见用户模块)
- borrowing\_list 借用申请审核列表
  - ✓ id 该请求id
  - ✓ name 该请求的名字
  - ✓ flag 重要资源
  - ✓ package\_name 套餐名字
  - ✓ note 管理员与用户相互传递的消息
  - ✓ type 计算资源类型，字符串：'real'（实体机），'virtual'（虚拟机）
  - ✓ cpu CPU描述
  - ✓ memory 内存大小
  - ✓ disktype 硬盘类型
  - ✓ disk 硬盘大小
  - ✓ ip 服务器网络地址
  - ✓ os 操作系统
  - ✓ login 登录用户名
  - ✓ initial\_password 初始密码
  - ✓ period 资源时间限制



- returning\_list 归还申请审核列表

- ✓ id 该请求id
- ✓ sn 该计算资源的sn号
- ✓ name 该请求的名字
- ✓ flag 重要资源
- ✓ note 管理员与用户相互传递的消息
- ✓ type 计算资源类型，字符串：'real'（实体机）,'virtual'（虚拟机）
- ✓ cpu CPU描述
- ✓ memory 内存大小
- ✓ disktype 硬盘类型
- ✓ disk 硬盘大小
- ✓ ip 服务器网络地址
- ✓ os 操作系统
- ✓ login 登录用户名
- ✓ initial\_password 初始密码
- ✓ period 资源时间限制

- modifying\_list 配置调整审核列表

- ✓ id 该请求id
- ✓ sn 该计算资源的sn号
- ✓ name 该请求的名字
- ✓ flag 重要资源
- ✓ note 管理员与用户相互传递的消息
- ✓ type 计算资源类型，字符串：'real'（实体机）,'virtual'（虚拟机）
- ✓ cpu CPU描述
- ✓ memory 内存大小
- ✓ disktype 硬盘类型
- ✓ disk 硬盘大小

- ✓ ip 服务器网络地址

- ✓ os 操作系统

- ✓ login 登录用户名

- ✓ initial\_password 初始密码

- ✓ period 资源时间限制

- inuse\_list 使用中列表

- ✓ id 该请求id

- ✓ sn 该计算资源的sn号

- ✓ name 该请求的名字

- ✓ flag 重要资源

- ✓ note 管理员与用户相互传递的消息

- ✓ type 计算资源类型，字符串：'real'（实体机）,'virtual'（虚拟机）

- ✓ cpu CPU描述

- ✓ memory 内存大小

- ✓ disktype 硬盘类型

- ✓ disk 硬盘大小

- ✓ ip 服务器网络地址

- ✓ os 操作系统

- ✓ login 登录用户名

- ✓ initial\_password 初始密码

- ✓ period 资源时间限制

- package\_list 套餐列表

- ✓ name 套餐名字

- ✓ type 计算资源类型，字符串：'real'（实体机）,'virtual'（虚拟机）

- ✓ cpu CPU描述

- ✓ memory 内存大小

- ✓ disktype 硬盘类型



✓ disk 硬盘大小

✓ os 操作系统

#### 表单ACTION

##### 1.computing.views.do\_approve\_borrow

实现功能：同意一个计算资源的借用申请，完成后重定向到本页面

提交方法：POST

- *id* 同意的计算资源申请*id*
- *sn* 该计算资源的SN号
- *note* 管理员消息
- *initial\_password* 初始登陆密码
- *login* 登陆用户名
- *ip* IP地址

##### 2.computing.views.do\_disapprove\_borrow

实现功能：拒绝一个计算资源的借用申请，完成后重定向到本页面

提交方法：POST

- *id* 同意的计算资源申请*id*
- *note* 管理员消息

##### 3.computing.views.do\_approve\_return

实现功能：同意一个计算资源的归还申请，完成后重定向到本页面

提交方法：POST

- *id* 同意的计算资源申请*id*
- *note* 管理员消息

##### 4.computing.views.do\_approve\_modify

实现功能：同意一个计算资源的配置调整申请，完成后重定向到本页面

提交方法：POST

- *id* 同意的计算资源申请*id*
- *flag* 重要资源，*true*有重要资源，*false*没有

- *type* 计算资源类型，字符串：'*real*'（实体机），'*virtual*'（虚拟机）
- *cpu* CPU描述
- *memory* 内存大小
- *disktype* 硬盘类型
- *disk* 硬盘大小
- *ip* 服务器网络地址
- *os* 操作系统
- *period* 资源时间限制
- *note* 管理员消息

##### 5.computing.views.do\_disapprove\_modify

实现功能：拒绝一个计算资源的配置调整申请，完成后重定向到本页面

提交方法：POST

- *id* 同意的计算资源申请*id*
- *note* 管理员消息

##### 6.computing.views.do\_create\_package

实现功能：创建一个新的套餐，完成后重定向到本页面

提交方法：POST

- *name* 套餐名字
- *type* 计算资源类型，字符串：'*real*'（实体机），'*virtual*'（虚拟机）
- *cpu* CPU描述
- *memory* 内存大小
- *disktype* 硬盘类型
- *disk* 硬盘大小
- *os* 操作系统

##### 7.computing.views.do\_delete\_modify

实现功能：删除一个新的套餐，完成后重定向到本页面

提交方法：POST

- *name* 套餐名字



# 日志模块



日志模块记录包括管理员在内的所有用户的操作行为，可以查看用户、实物资源、计算资源的操作日志，本章将从基本功能、数据模型、前后端



## 基本功能

功能	功能说明
查看用户操作日志	管理员可以查看某个用户操作的日志记录
查看用户信息修改日志	管理员可以查看某个用户修改个人信息的日志记录
查看用户所有相关日志	管理员可以查看所有跟某个用户相关的日志记录（即上述两项的并集）
查看实物资源日志	管理员可以查看某个实物资源的状态日志，包括改变资源状态的操作人的信息，注意只有实物资源状态发生了改变才会被记录，如果借用请求被拒绝，实物资源状态并没有发生变化，则这次请求不会被记录（但是会被“借还请求日志”记录，请见后面）
查看计算资源日志	与实物资源日志相似，但是是针对计算资源的
查看所有借还请求日志	所有用户提交请求和管理员审核请求都会被记录，即便资源状态可能并没有发生改变（如管理员拒绝用户借用请求这种情况）

## 数据模型

class LogAccount(models.Model)

类说明：记录用户的操作行为

成员变量：

变量名	变量类型	变量说明
user_id	int	操作者id
target	ForeignKey(Account)	被操作对象
action	str	动作类型，包括create、update、delete等
time	DateTime	操作时间
description	str	动作描述，如提出借用申请

class LogBorrow(models.Model)

类说明：记录借还请求的操作

成员变量：

变量名	变量类型	变量说明
user_id	int	操作者id
target	ForeignKey(Borrow)	被操作对象
action	str	动作类型，包括create、update、delete等
time	DateTime	操作时间
description	str	动作描述，如提出借用申请



class LogSingle(models.Model)

类说明： 记录实物资源状态的改变  
成员变量：

变量名	变量类型	变量说明
user_id	int	操作者id
target	ForeignKey(Single)	被操作对象
action	str	动作类型，包括create、update、delete等
time	DateTime	操作时间
description	str	动作描述，如提出借用申请

class LogComputing(models.Model)

类说明： 记录计算资源状态的改变  
成员变量：

变量名	变量类型	变量说明
user_id	int	操作者id
target	ForeignKey(Computing)	被操作对象
action	str	动作类型，包括create、update、delete等
time	DateTime	操作时间
description	str	动作描述，如提出借用申请

前后端接口

显示特定类型对象日志页面

前端模板： aglaia/templates/log.html  
后端链接： log.views.show\_log  
显示FUNCTION （GET方法获得如下参数）

- type 日志类型
  - ‘goods’实物资源 ‘computing’计算资源
  - ‘user’ 用户 ‘borrow’借还请求
- id 日志查询对象的id
- is\_actor 如果查询用户日志，此项表示筛选项
  - ‘true’ 该用户的操作日志
  - ‘false’ 该用户的信息修改日志
  - ‘All’ 所有和该用户相关的日志

RENDER CONTEXT (JSON):

- base.html要求的RENDER CONTEXT(请见用户模块)
- logs 所有日志条目的列表JSON
  - ✓ actor 操作对象
  - ✓ target 被操作对象
  - ✓ action 操作行为
  - ✓ time 操作时间
  - ✓ desc 操作行为描述



## 显示实物资源借还请求

前端模板：aglaia/templates/log\_borrow.html

后端链接：log.views.show\_borrow\_list

RENDER CONTEXT (JSON):

- base.html要求的RENDER CONTEXT(请见用户模块)
- borrows 所有日志条目的列表JSON
  - ✓ id 借用请求的id
  - ✓ sn 借用物品sn号
  - ✓ name 被借出的物品名
  - ✓ note 管理员消息或者用户说明

## 显示计算资源借还请求

前端模板：aglaia/templates/log\_computing.html

后端链接：log.views.show\_computing\_list

RENDER CONTEXT (JSON):

- base.html要求的RENDER CONTEXT(请见用户模块)
- computings 所有日志条目的列表JSON
  - ✓ id 借用请求的id
  - ✓ sn 借用物品sn号
  - ✓ name 被借出的物品名
  - ✓ note 管理员消息或者用户说明
  - ✓ package\_name 套餐名字
  - ✓ type 计算资源类型，取值'real'（实体机），'virtual'（虚拟机）
  - ✓ cpu 资源CPU描述
  - ✓ memory 内存大小

- ✓ disktype 硬盘类型
- ✓ disk 硬盘类型
- ✓ ip 服务器网络地址
- ✓ os 操作系统
- ✓ login 登陆用户名
- ✓ initial\_password 初始密码