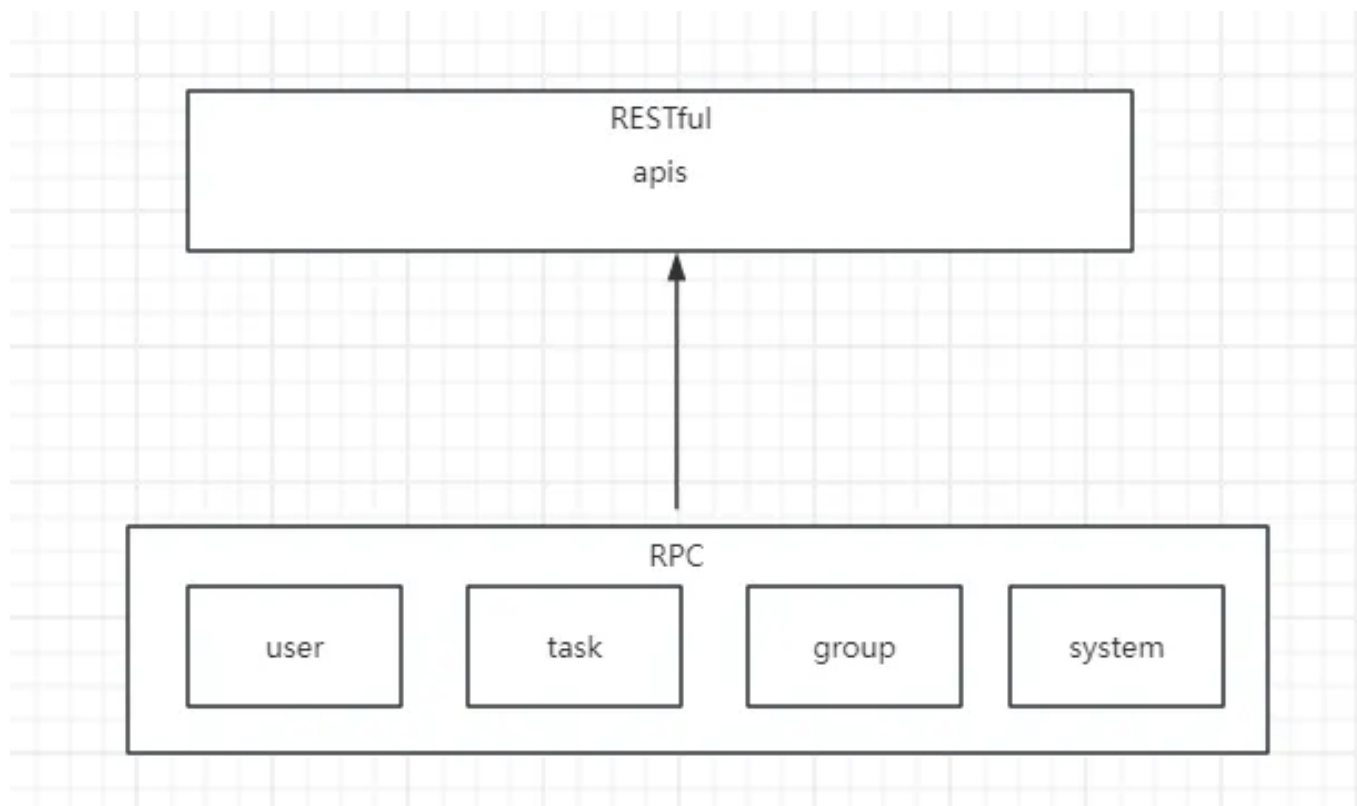


4.系统设计

- 1.总体设计
- 2.用户管理
- 3.机器人功能
- 4.群组管理和系统设置

1.总体设计

该项目采用go-zero框架，模块之间的调用使用RPC协议，向外部暴露一套RESTful风格API，供前端调用



2.用户管理

- 1. 实现用户注册/登录/注销/修改用户信息功能
 - a. 用户名：需要为真实的中文用户名
 - b. 手机号：用于实现群组@功能

- c. 邮箱：用于发送通知信息
 - d. Jenkins账号和密码：用于执行Jenkins任务时身份验证
 - e. 部门信息：
2. 用户登录信息保存90天失效

3.机器人功能

1. 执行HTTP和HTTPS的GET方法接口，参数通过路径传入接口
2. 执行Jenkins Job，执行记录需要是@机器人的人员，不可写死账号信息，导致Jenkins的执行记录混乱。Jenkins Job也可携带入参
3. 执行远程脚本，需要指定执行主机的host/user/password，脚本文件通过web上传至目标服务器，而不是让用户去手动上传
4. 上述3类Task的参数传递遵循以下原则：
 - a. 第一个参数：Task名称
 - b. 其他参数：Task入参
 - c. 参数通过英文";"来进行分割

4.群组管理和系统设置

1. 访问web根目录时，需要先到Mysql去查看该企业有没机器人存在，若有，跳至登录/注册页面，否则新建机器人教程页面，然后通过按钮来跳至机器人配置页面
2. 机器人的企业密钥等信息需要通过web页面来进行配置，配置信息在数据库中应为密文，加盐
3. 不同群组的机器人功能独立，web页面以群组为单位来展示对应群组中的Task
4. 各群组的机器人Token需要展示在群组页面，且仅管理员可见
5. web页面仅展示有该机器人的群组，而不是展示企业的所有群组，通过探针，每10分钟来执行一次接口，获取企业内部群组和机器人信息，该时间间隔可以通过web页面来调整
6. Task页面需要包含近期的Task执行记录信息，默认按照存储14天执行历史，可通过web端配置