

# GDB 实时数据库系统客户端使用教程

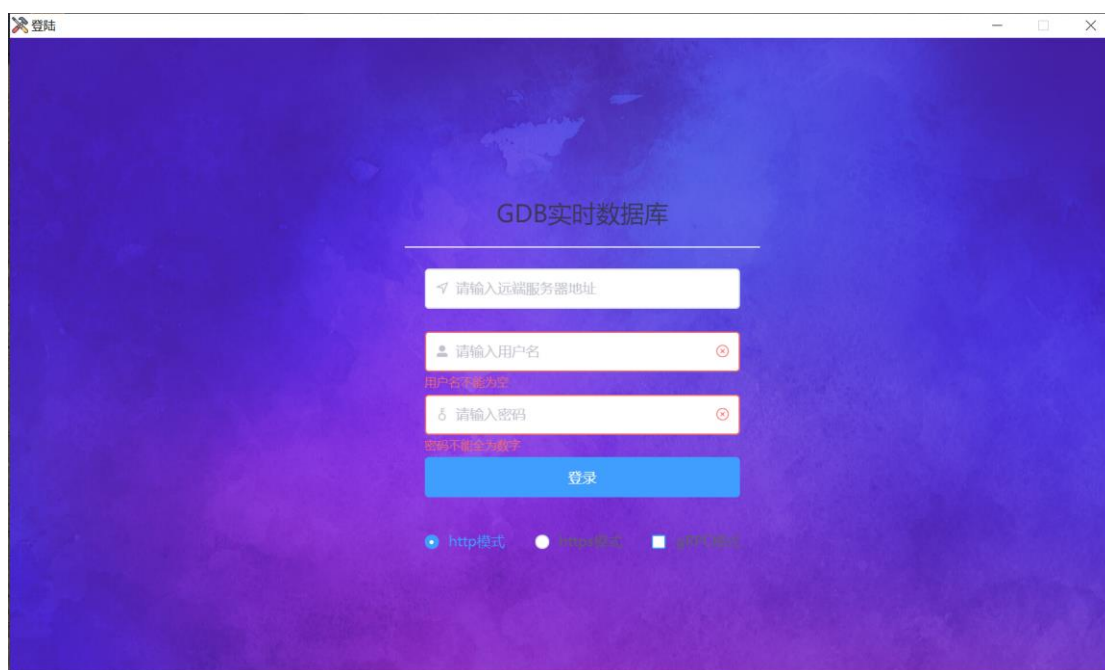
GDB 实时数据库系统客户端使用教程.....	1
1. 简介.....	2
2. 登录界面.....	2
1.1 用户登陆.....	2
1.2 登陆模式.....	3
1.3 Cookie.....	3
3. 首页.....	3
4 分组管理.....	4
4.1 批量加组.....	4
4.2 单个加点.....	5
4.3 批量加点.....	6
4.4 点表下载.....	6
4.5 编辑此组.....	7
4.6 删除此组.....	8
4.7 清空此组.....	8
4.8 历史导入.....	8
4.9 表格管理.....	9
4.10 表格搜索.....	10
5. 二次计算.....	10
5.1 主界面.....	10
5.2 新增计算项.....	11
5.3 编辑计算项.....	12
5.4 开始，停止，删除计算项.....	13
6. 用户手册.....	13
7. 运行日志.....	13
7.1 日志批量删除.....	14
8. 用户管理.....	15
8.1 权限管理.....	15
8.2 页面使用.....	15
9 路由管理.....	17
9.1 路由权限管理.....	17
9.2 页面使用.....	17

# 1.简介

为了让大家可以更方便的使用 GDB 实时数据库，我们提供了客户端应用程序，客户端程序是使用 Electron+Vue 编写而成，具有良好的跨平台支持。您可以下载安装程序之后，直接安装即可使用。在使用客户端的时候需要注意以下几点：

- (1) 在授权模式下，一个用户只能登陆一个客户端（单点登陆）
- (2) 客户端目前不支持根证书认证，如果想换成您自己的证书，需要替换 resources\model\ssl 文件夹下的 gdbServer.crt 和 gdbServer.key 文件（需要注意文件的名称不能修改，必须为 gdbServer 开头）

## 2.登录界面



### 1.1 用户登陆

在使用客户端之前必须先进行登录，客户端有完善的权限管理，对于不同的用户权限不同。Gdb 中的用户分为管理员，普通用户和游客三种。初始的管理员用户名和密码分别为 **admin** 和 **admin@123**。

同时用户在登陆的时候需要输入远程服务器即 gdb 服务的地址，**需要注意的是必须同时输入 ip 地址和端口号，例如 192.168.0.199:8082**

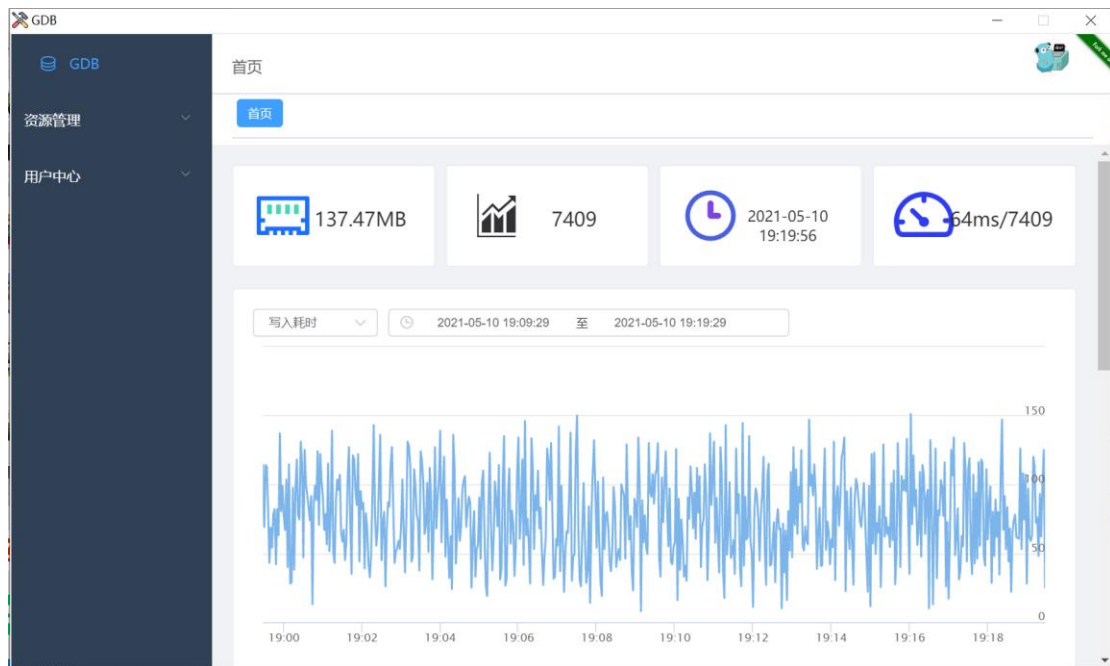
## 1.2 登陆模式

和 gdb 服务相对应，客户端提供了 http,https,http-gRPC,https-gRPC 四种登陆模式。对于 https-gRPC 模式在登陆时需要输入证书的域名（我们提供的证书的默认域名为 gdb.com）。需要注意的是这里选择的模式要和 gdb 服务开启的模式相对应。例如 gdb 服务开启的是 http 模式，那么这里只能选择 http 或者 http-gRPC 模式。

## 1.3 Cookie

客户端默认提供了 cookie 的支持，在用户登陆成功之后，用户的登陆信息会默认保存一天。在一天之后再次登陆即需要重新输入用户信息。

# 3. 首页

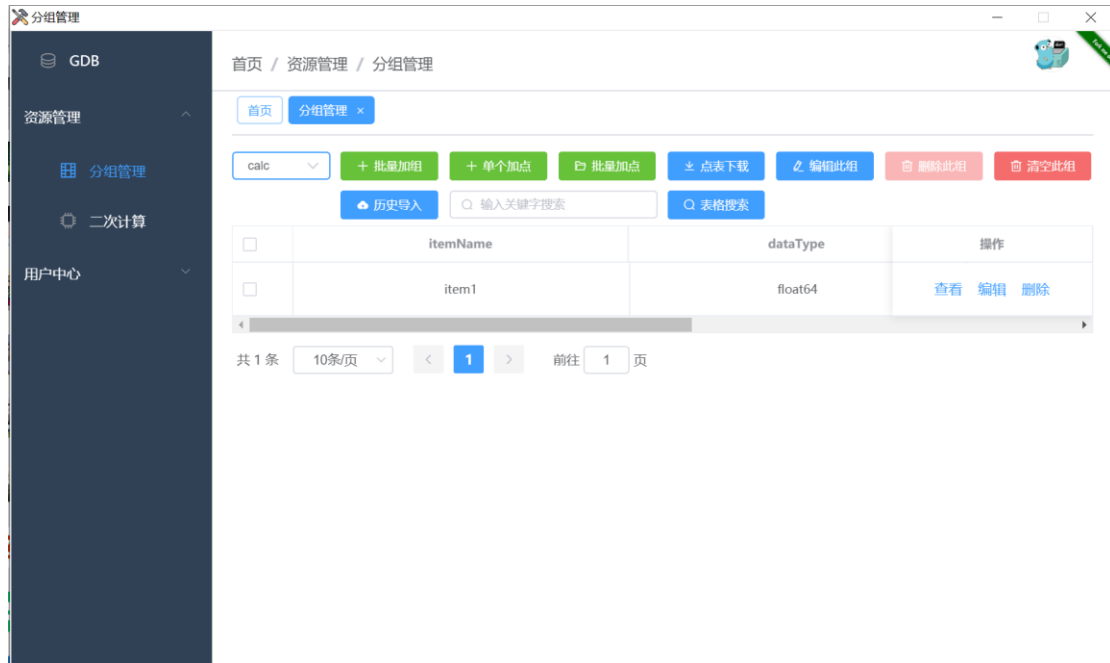


客户端首页展示了 gdb 服务在运行时的一些指标，分别是占用的实时内存，实时写入 item 的个数，最近一次实时数据更新时间，当前实时数据写入速率。

下面展示了写入耗时的历史和内存使用的历史。通过选择时间，用户可以查看这两项指标的历史数据。

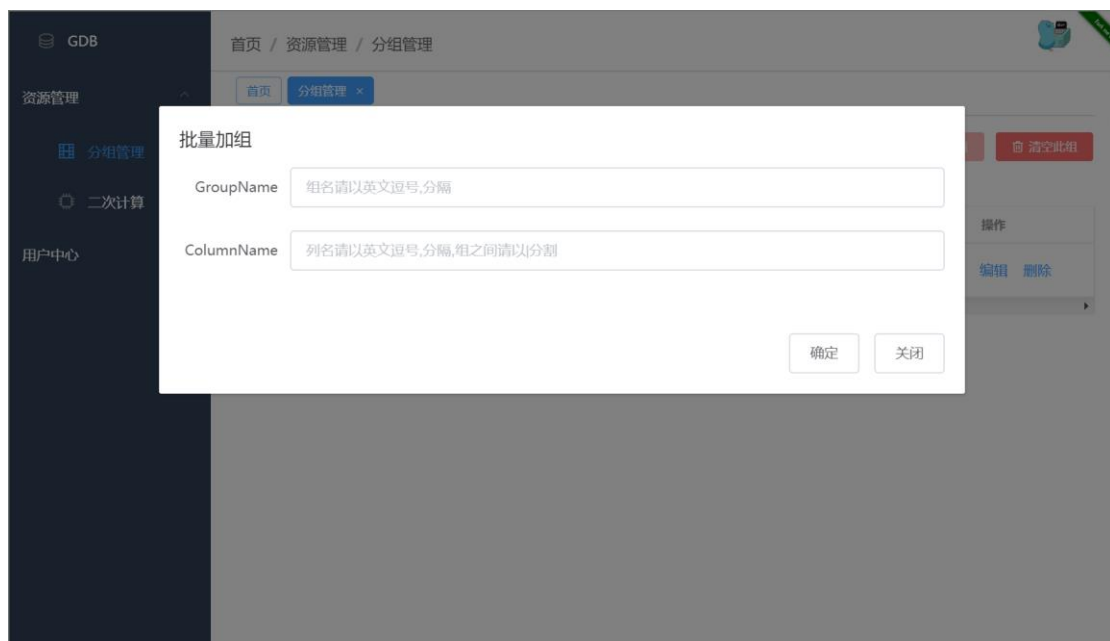
## 4 分组管理

分组管理主要用来管理 gdb 中的组以及组中的 item，其界面如下，需要注意的是 calc 分组是 gdb 客户端自带的，主要用来存储二次计算相关的 item，用户不可删除该组。



### 4.1 批量加组

组是 gdb 服务的基础，在使用之前需要先加组，点击批量加组按钮，弹出的界面如下：

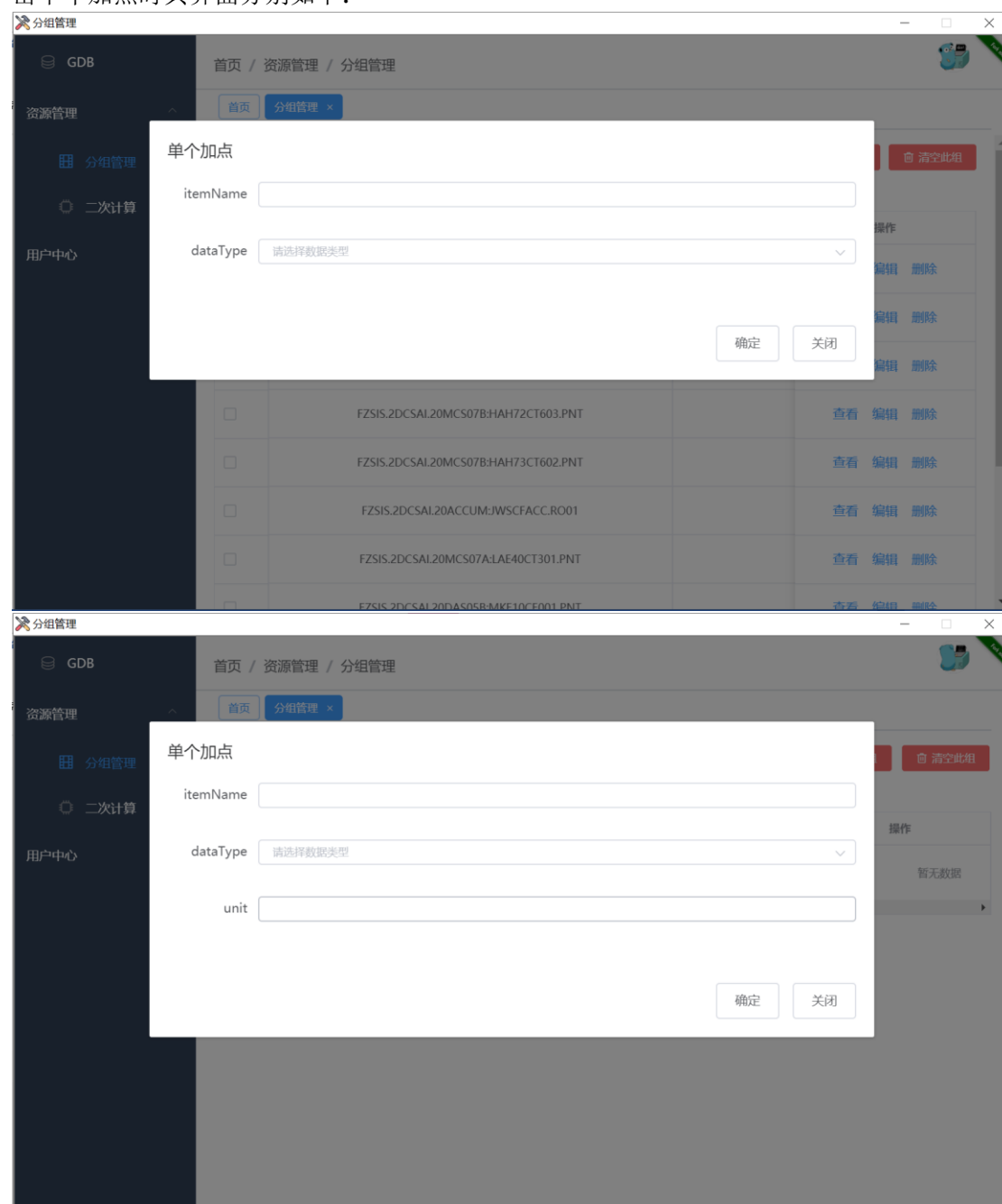


对于需要添加的组名之前我们需要以英文的逗号进行分隔，对于组的列名之间我们需要以英文逗号分隔，对于不同组的列名之间我们需要以|分隔。需要注意的是每个分组 gdb 都默认

有 **id**，**itemName**，**dataType** 三列，分别代表的是每个 **item** 的 **id**，名称以及数据类型，所以在客户端添加分组指定列名时不需要添加这三列，否则会添加失败。同时 **itemName** 是唯一的，不能重复，**dataType** 只能是 **int64**,**float64**,**bool**,**string** 这四个选项之一。并且组名也是不能重复的，重复添加相同的组名也会失败。

## 4.2 单个加点

选择对应的分组，点击单个加点按钮，我们就可以向对应的组里面添加 **item**，我们称之为点。（我在之前添加了 **1DCS**，没有指定自定义列以及 **2DCS**，指定了自定义列 **unit**），所以点击单个加点时其界面分别如下：



我们只需要填入对应的信息，选择 **item** 的数据类型，点击确定即可添加成功。

### 4.3 批量加点

有些时候，单个加点的效率太低，所以我们还提供了批量加点的功能。点击批量加点按钮，界面如下：



我们可以上传后缀为 `xlsx` 的 `excel` 格式的点表，然后点击确定，即可实现批量加点功能。需要注意的是 `excel` 点表的格式必须和对应的 `group` 保持一致。例如，`2DCS` 分组的列为 `id`，`itemName`，`dataType`，`unit`，其中前三列是系统内嵌的，最后一列是用户自定义的。那么对应的点表的 `header` 即第一行必须是 `itemName`，`dataType`，`unit`，之后的每行数据即为对应列的数据。同时 `itemName` 不可重复，`dataType` 只能是 `int64`，`float64`，`string`，`bool` 四者之一。（`id` 为自增列，会自动维护，不需要出现在 `excel` 点表中）。

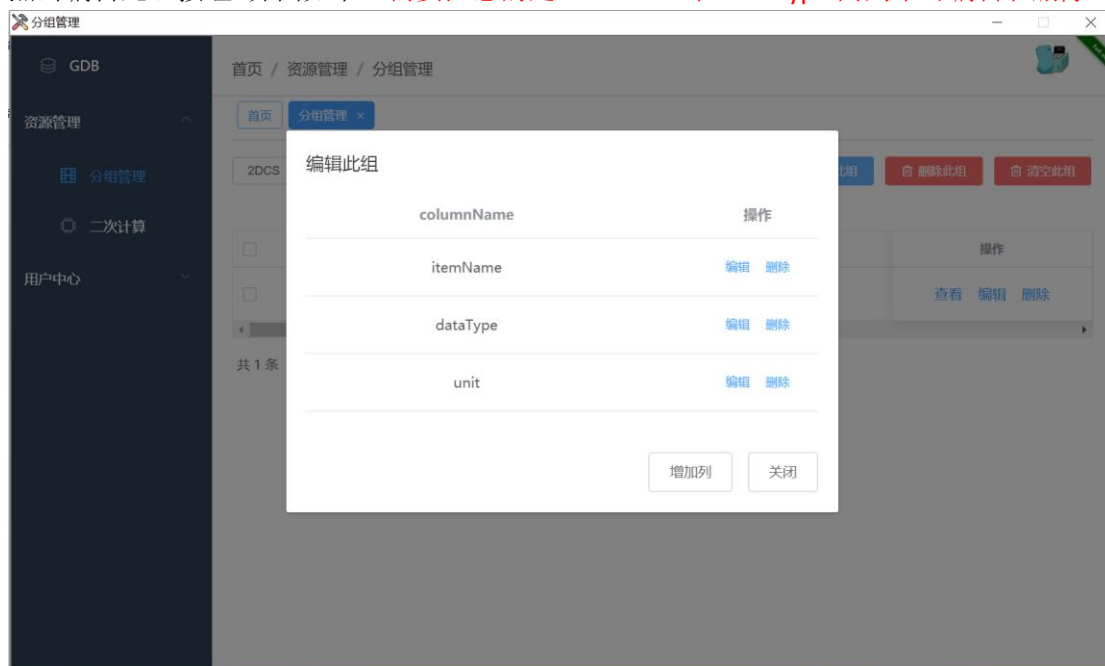
itemName	dataType	unit
Item1	int64	
Item2	float64	
Item3	string	
Item4	bool	

### 4.4 点表下载

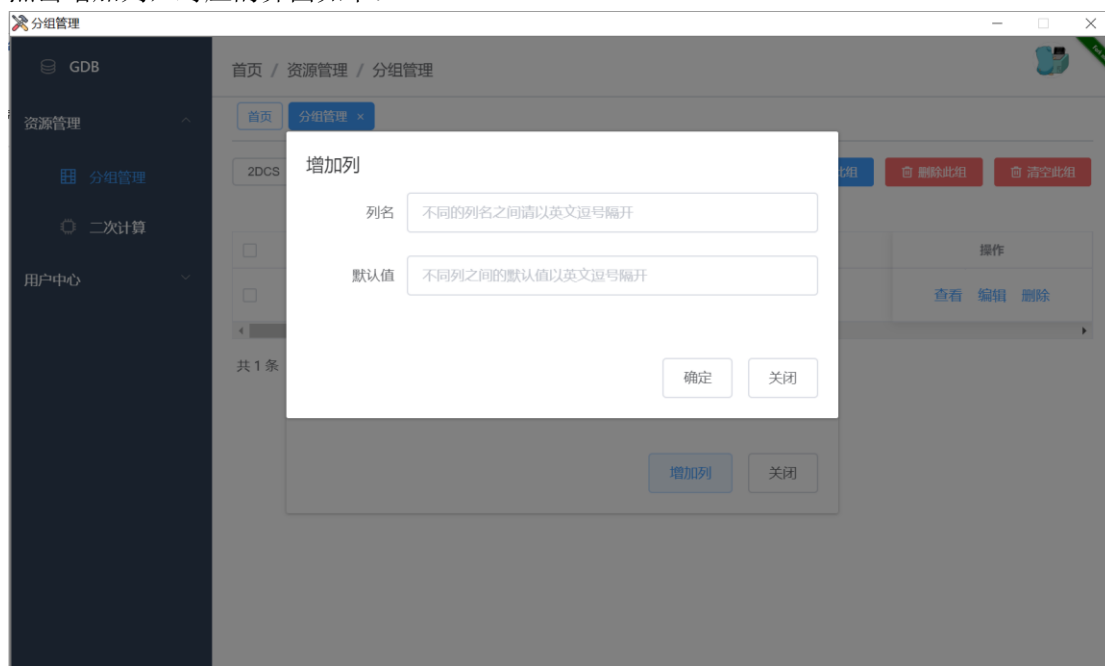
我们也可以在客户端下载分组中所有的 `item`，只需要点击点表下载，然后选择保存路径即可。

## 4.5 编辑此组

有时候我们需要在添加分组之后再去增加、删除或者修改其中的某些列。这个时候我们可以点击编辑此组按钮。界面如下：（需要注意的是 **itemName** 和 **dataType** 两列不可编辑和删除）



我们可以删除某列，编辑对应的列名。（目前不能改变某列的默认值）  
点击增加列，对应的界面如下：



对于需要增加的列之间需要以英文的逗号分隔，对应的默认值之前需要以英文逗号隔开，如果想指定默认值为空字符串，则默认值需要填写 "", 同时默认值必须要是可以被 `json` 反序列化的值，即“默认值”需要可以被 `JSON.parse()` 函数反序列化。需要注意的是列名也是唯一的，重复添加相同的列名会失败。

## 4.6 删除此组

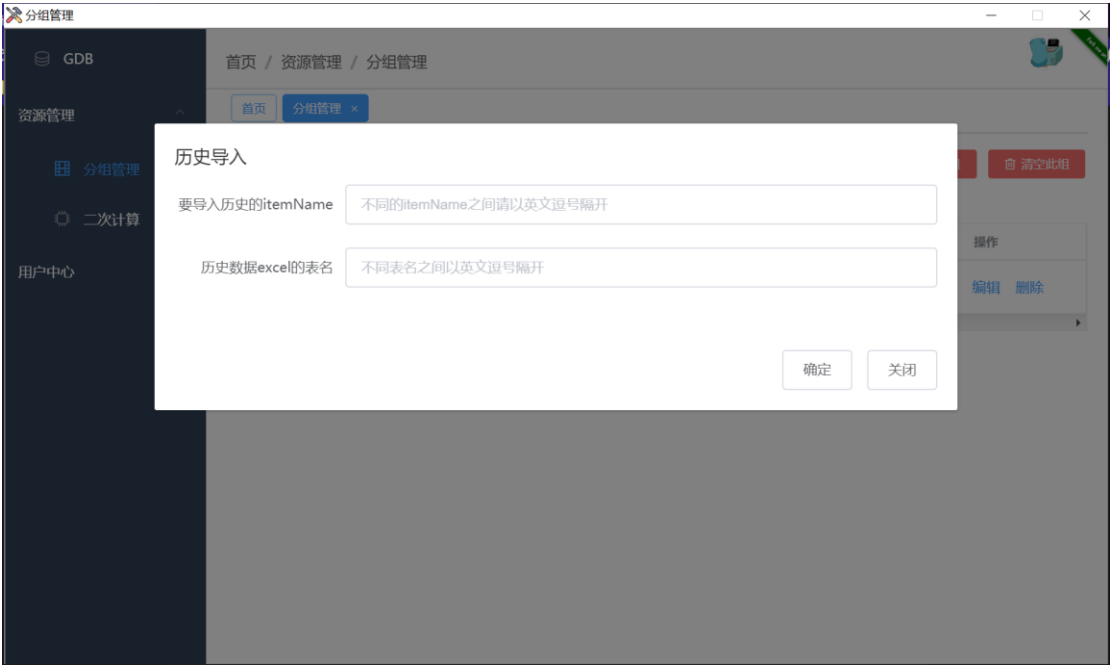
点击删除此组即可以删除该分组（目前历史数据不会被删除）

## 4.7 清空此组

点击清空此组会清空该分组下所有的 item（目前历史数据不会被清除）

## 4.8 历史导入

我们可以批量导入 item 的历史数据，点击历史导入按钮，界面如下：



对于需要历史数据的 item 之间我们需要以英文逗号分隔，对应的 excel 的 sheet 的名称以英文逗号隔开。其中 excel 中数据的格式应该为两列，第一个为时间，第二列为该时间对应的数据（需要注意的是所有导入历史的 item 都必须在对应的组中存在，并且数据类型也要与之对应，否则会导入历史失败）

2020-10-10 16:33:01	10
2020-10-10 16:33:02	20
2020-10-10 16:33:03	30

之后点击确定按钮，等待历史导入完成即可。



## 4.9 表格管理

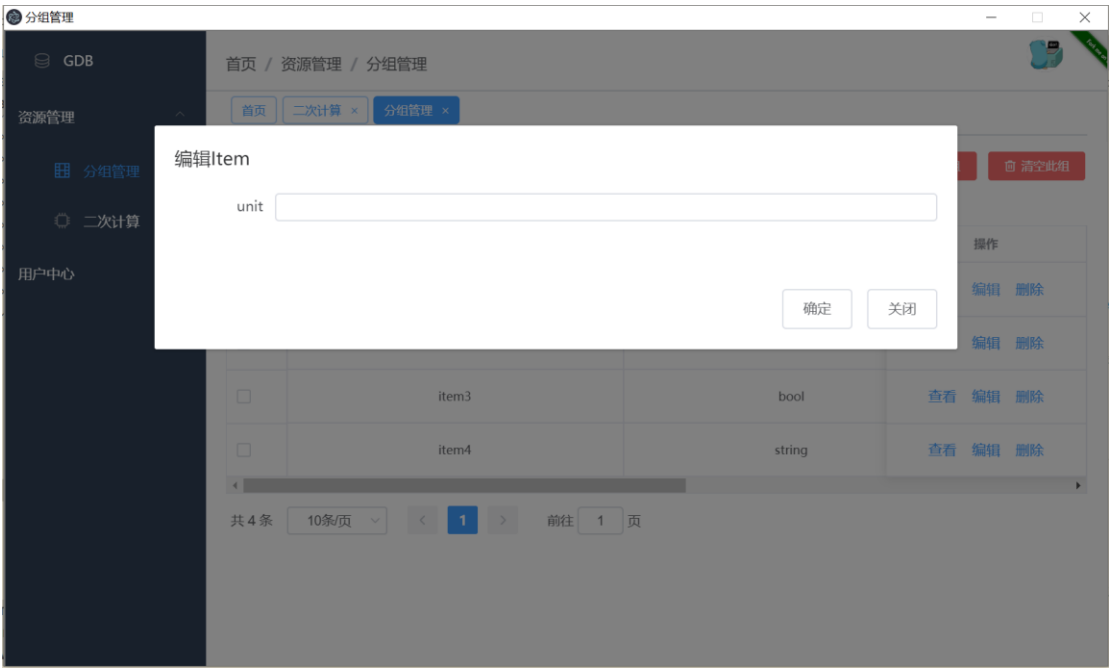
整个分组管理界面中间为对应的数据表格，以表格的形式展现了其中的数据。其界面如下：



其中每个分组的表格战术都有默认的 `itemName`，`dataType`，`realTimeData` 三列，分别代表了 `item` 的名称，数据类型，以及最近一次更新的值。下方为分页操作符，可以选择每页显示的数据数目，当前展示的数据页码以及进行页码的跳转。

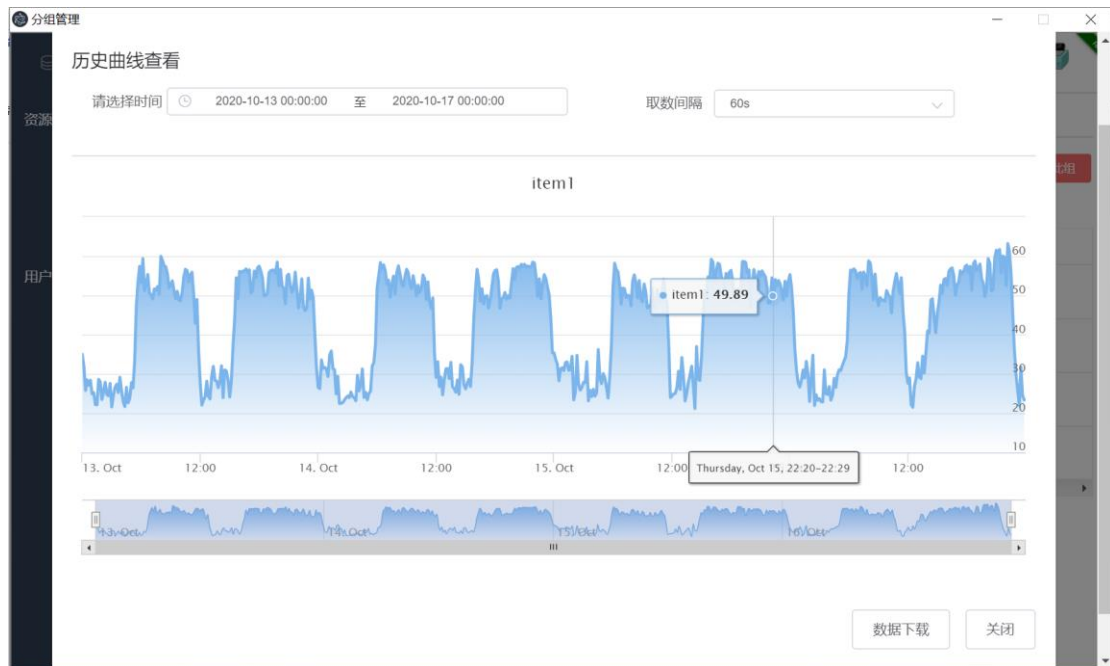
表格右侧为操作列，分别是查看，编辑以及删除按钮。

点击编辑按钮其界面如下：



可以看到，我们可以编辑对应 `item` 自定义列的内容，但是我们不能编辑 `itemName` 以及 `dataType`。

点击查看按钮，可以查看数据类型为 `int64` 或者 `float64` 的 `item` 的历史数据，默认的起始时间为当前时间的前一天，取数间隔为 `60s`，其界面如下：



点击数据下载，可以下载该段时间内的历史数据。

点击删除按钮可以删除对应的 `item`（目前不会删除历史数据）。

## 4.10 表格搜索

页面上方的搜索输入框可以根据 `itemName` 对表格中的数据进行模糊搜索，**需要注意的是必须点击表格搜索按钮才有效果**。如果输入框的内容为空，则查询所有的数据。

# 5. 二次计算

二次计算提供了使用 `JS`（`ES5` 及以下）代码来生成用户自定义数据的功能。用户可以使用任何合法的 `js` 代码来基于已经存在的数据去生成用户想要的数。我们将这种功能称之为二次计算。需要注意的是二次计算的点组为 `calc`，该组不能被删除。

## 5.1 主界面

主页面的表格展示了和二次计算有关的所有内容。分别是 `description`（二次计算的描述），`expression`（二次计算的 `JS` 表达式），`duration`（计算间隔），`status`（计算状态），`errorMessage`（计算过程中的错误信息），`createTime`（创建时间），`updateTime`（最近更新时间）。点击新增计算项可以新增二次计算项，也可以根据 `description` 进行表格内容的搜索。

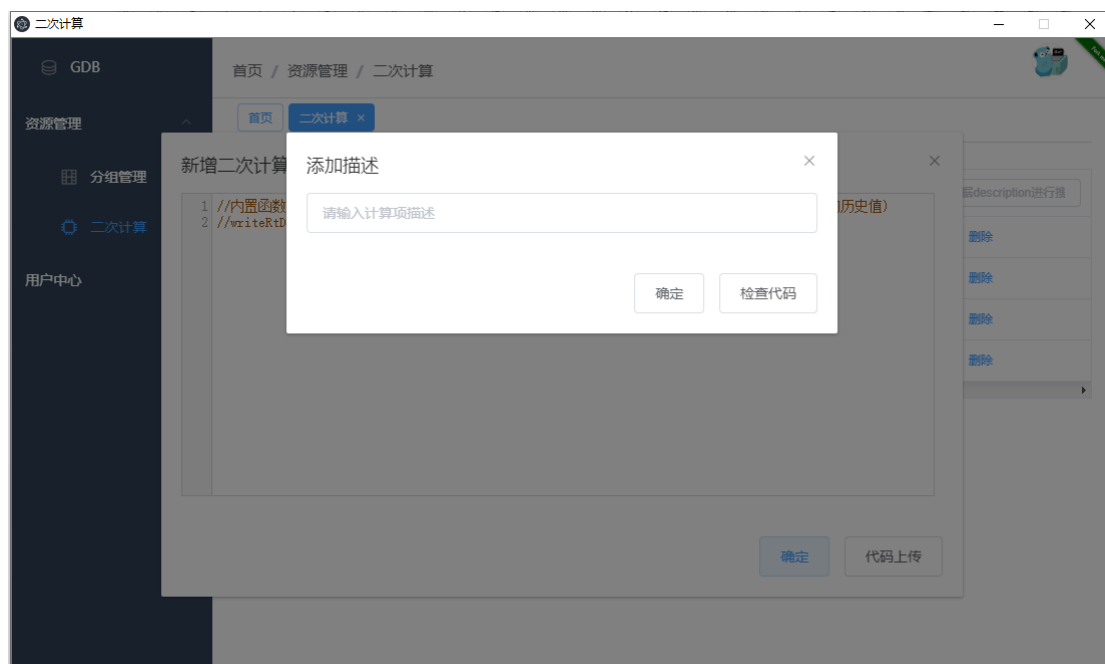


## 5.2 新增计算项

点击新增计算项，新弹出的界面如下：



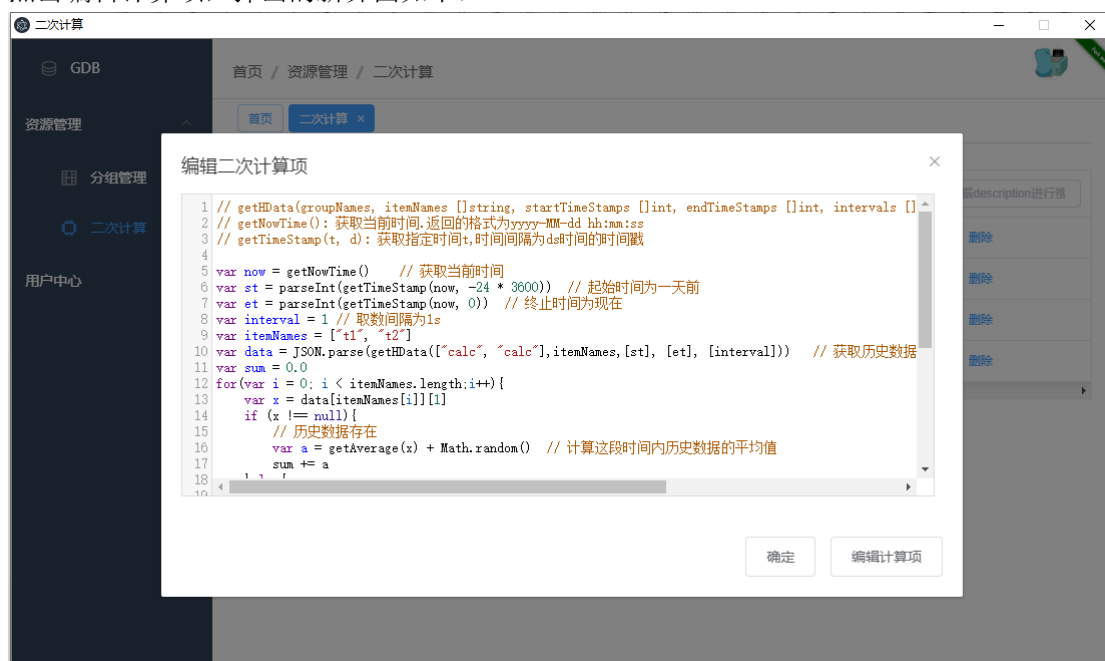
我们可以在这个界面中编写我们的二次计算 JS 代码，可以看到我们已经内置了 6 个基本的函数来帮助用户编写二次计算代码，这 6 个函数主要用来和 gdb 数据库进行交互。同时我们也可以点击代码上传来上传已经写好的 js 代码。  
需要注意的是目前只支持 ES5 及以下的 js 语言，定义变量只能使用 var 关键字。  
当代码输入完毕之后，点击确定按钮，新弹出的界面如下：



点击检查代码可以检查代码是否有错误，输入计算项描述，点击确定，如果代码逻辑无误则可以成功添加计算项。需要注意的是，计算项的描述不能为空。

## 5.3 编辑计算项

点击编辑计算项，弹出的新界面如下：



在这里我们可以对计算项的代码进行修改，点击编辑计算项，可以对计算项的描述以及计算间隔进行修改，需要注意的是计算项的间隔不能小于 1s

之后在该界面点击确定，如果代码逻辑无误即可完成计算项的修改。

需要注意的是计算项的修改以及时间间隔的修改，必须在编辑二次计算项这个页面点击确定之后才会生效。

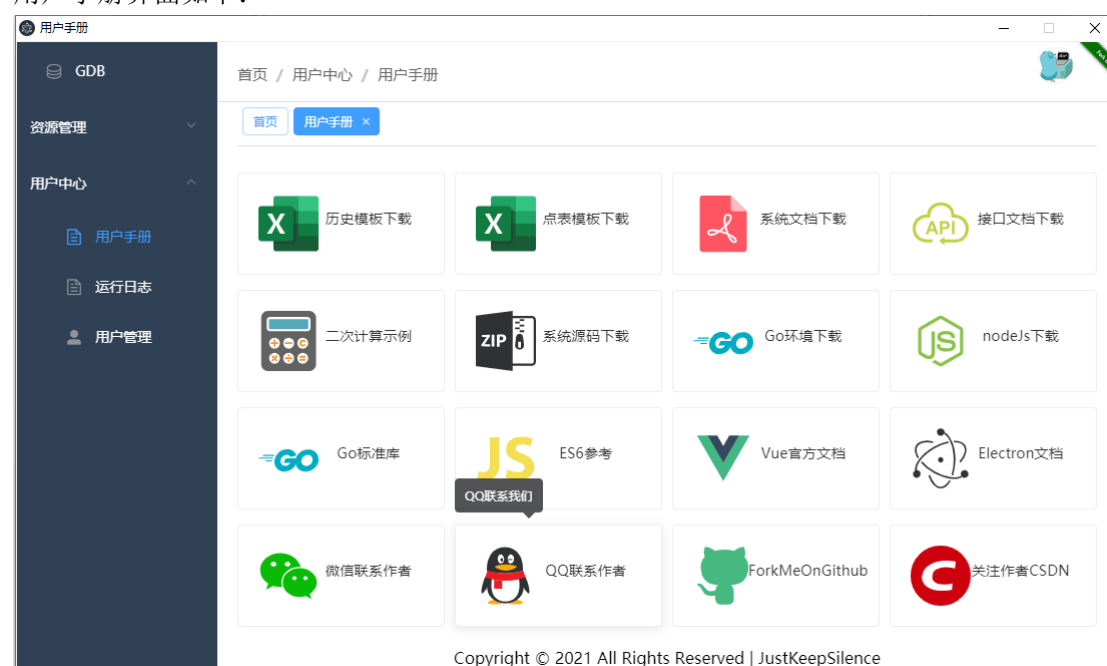
## 5.4 开始，停止，删除计算项

点击开始/停止计算项即可开始或者停止计算项。点击删除即可删除计算项。

需要注意的是，所有对计算项的修改，删除，启停操作，最多在 1 分钟后才会起作用，因为 gdb 内置有计时器，而该计时器的刷新周期就是 1 分钟。

## 6. 用户手册

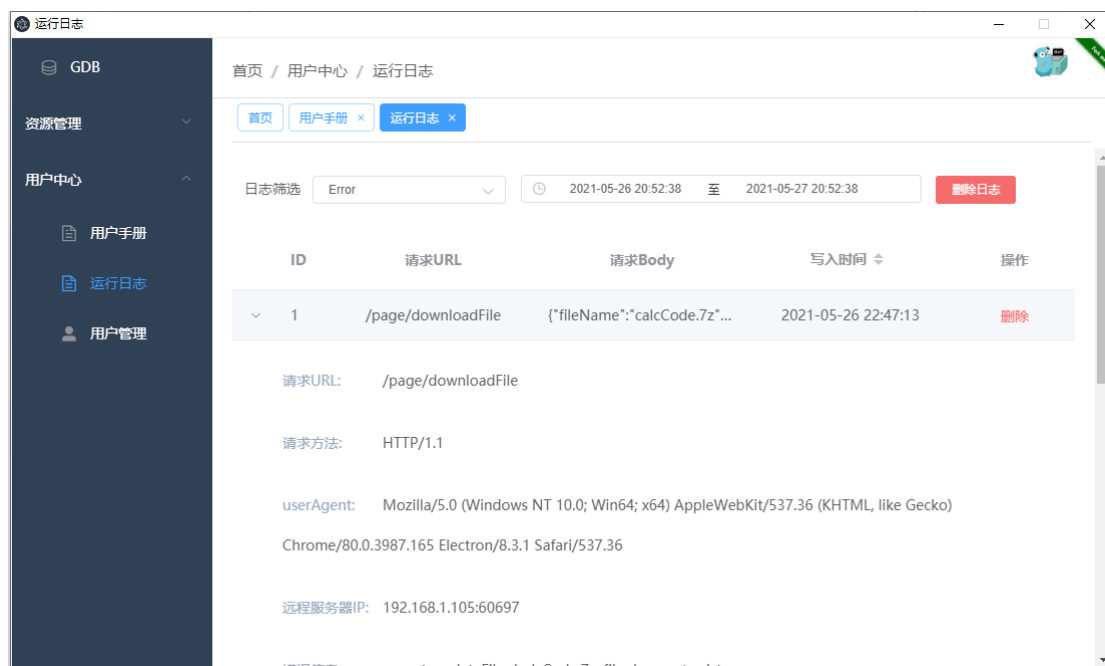
用户手册界面如下：



在这里用户可以下载各种导入所需要的模板，可以下载二次计算代码示例，也可以下载系统的源代码以及环境。当然也可以联系我们。

## 7. 运行日志

运行日志界面如下：



在 gdb 服务器开启日志功能之后，我们就可以在运行日志界面去查看系统运行时的所有请求日志。

日志筛选功能提供了日志的筛选功能，可以选择 all, Info, Error，分别代码查看所有的日志，查看类型为 Info 的日志以及查看类型为 Error 的日志。

对于每条日志我们可以查看其详细的信息，也可以删除它们。

## 7.1 日志批量删除

点击删除日志按钮，弹出的界面如下：



其中非授权日志指的是在 gdb 不需要授权模式运行时所记录的日志，对于这种类型的日志，普通用户和管理员均可以删除。

而其他用户的日志只有管理员可以删除，即普通用户只能删除自己请求的日志以及非授权日

志，管理员可以删除所有用户的日志和非授权日志。

## 8. 用户管理

### 8.1 权限管理

Gdb 中的用户分为三种，分别是管理员，普通用户和游客，分别对应不同的权限，权限列表如下：

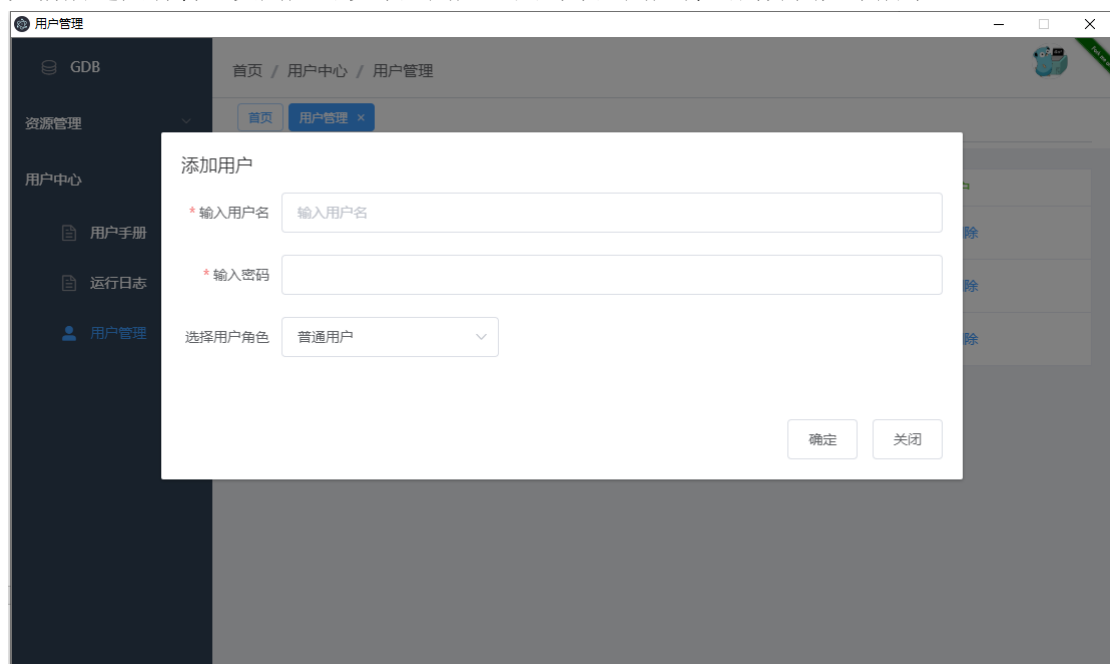
	管理员	普通用户	游客
首页	√	√	无法查看项目来源
分组管理	√	无法编辑，删除，清空分组，无法编辑，删除 item	只能查看
二次计算	√	√	只能查看
用户手册	√	√	只能查看
运行日志	√	只能删除当前用户日志和非授权日志	×
用户管理	√	不能添加，删除用户，并且只能编辑当前用户	×

### 8.2 页面使用

用户管理界面如下：



如前所述只有管理员用户可以添加用户，点击添加用户弹出的界面如下所示：



在这里我们输入用户名（不能为空），密码（不能全为字母或者数字并且不能小于 6 位），并选择用户角色即可添加用户。用户名是唯一的，不可重复添加。

点击编辑用户即可编辑用户（需要注意的是 **admin** 用户不可编辑和删除）。管理员用户可以编辑除了 **admin** 之外的所有用户，普通用户只能编辑自己，在编辑页面可以编辑用户名，更改用户密码和用户角色。

点击删除可以删除用户。



## 9 路由管理

### 9.1 路由权限管理

为了对各个用户可以访问的接口 url 进行更精确的控制，gdb 支持用户路由管理。首先必须明确的是 gdb 中的用户分为两大类，登陆用户和非登陆用户。登陆用户即为在用户管理界面中添加的用户，这些用户可以登陆 gdb 客户端，所以具有一些默认的路由权限，对于这些默认的路由权限用户是不能删除的（否则会影响用户的正常登陆）。非登陆用户即为在路由管理界面添加的用户，这些用户不能登陆 gdb 客户端，只能够访问某些特定的接口 url。这个界面的用户权限和用户管理界面一致，即管理员可以添加用户，编辑和删除所有用户（除了 admin 用户）。普通用户只能编辑自身和非登陆用户。游客没有该页面的权限。

### 9.2 页面使用

路由管理的界面如下：



点击添加用户弹出的界面如下：

添加用户

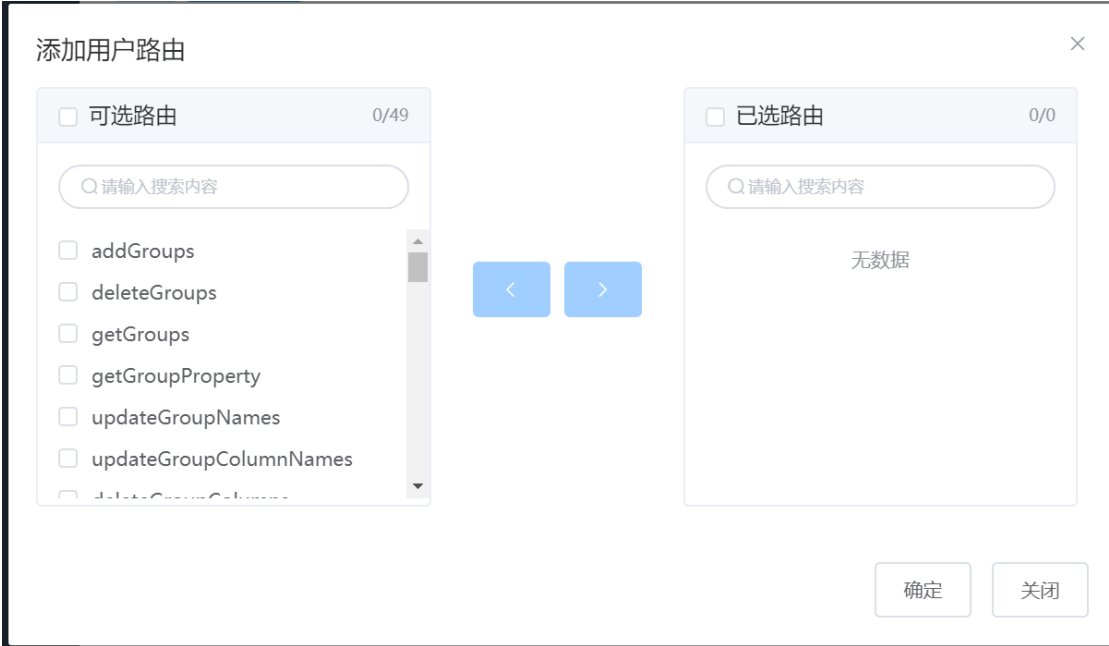
\* 输入用户名

输入用户名

确定

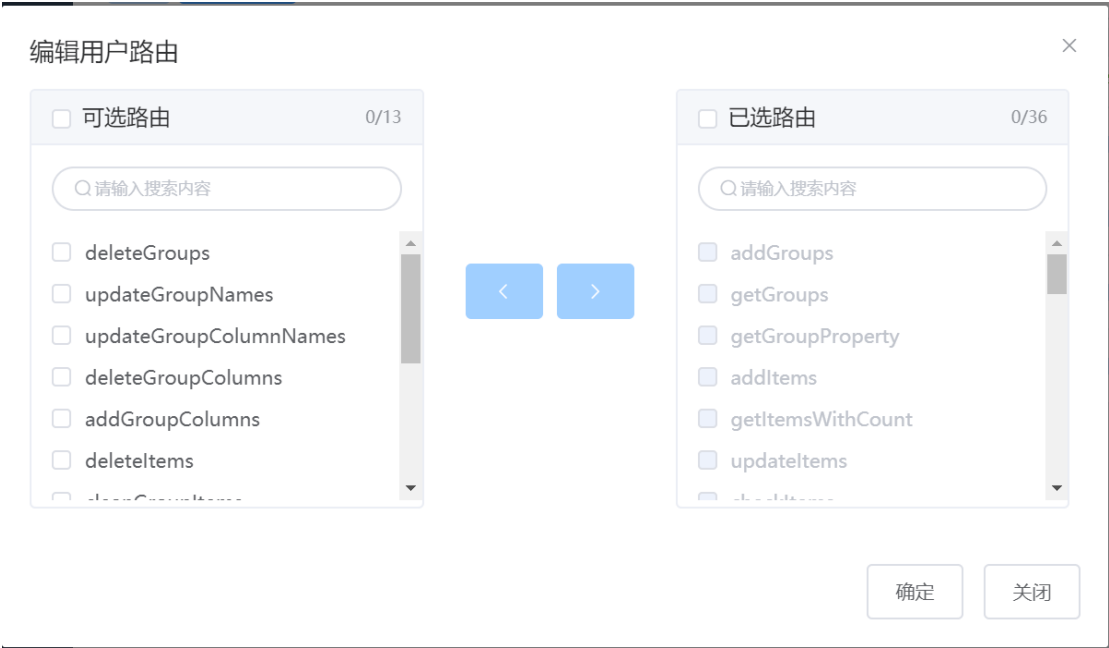
关闭

在输入要添加的用户名之后，点击确定，弹出的界面如下：



左侧为可以添加的用户路由列表，选中之后，点击按钮，即可以添加到右侧，之后点击确定即可成功添加非登陆用户及其对应的路由列表。

点击编辑按钮，即可对用户的路由进行编辑，弹出的界面如下：



如前所述，对于登陆用户，其默认的用户路由是不能更改的。

点击删除即可删除对应的用户，需要注意的是这个界面同样不能删除登陆用户。