

API 调用方法

测试网站：<http://api.gvrcraft.com:8000/>
比如：<http://api.gvrcraft.com:8000/api/user/18600028197>

如果出错，返回：
{ "code": "错误类型信息", "message": "补充信息" }

如果成功，返回：
{ "code": "", "message": "补充信息", "data": {...} }

由于BODY是JSON格式，如果有BODY，必须在HEADER中使用 **Content-Type: application/json**

错误类型列表

| 名称 | 说明 | Code |
|--------|---------------------|------|
| 数据无效 | 输出的数据不正确 | 400 |
| 申请失败 | 调用第三方的数据失败，比如：发短信失败 | 401 |
| 数据获取失败 | 调用本地数据失败 | 402 |
| 校验失败 | 各种验证的失败 | 403 |
| 内部错误 | 程序内的未注册错误 | 500 |
| | | |

函数列表

获取用户数据

GET /api/user/{username}?token={token}
username 为用户ID

返回数据参数表

| 名称 | 类型 | 说明 |
|----------|--------|-----------------------------|
| username | string | 用户ID |
| access | string | 微信或者微博登录需要的信息（如果是手机登录，那么为空） |
| nickname | string | 用户显示在界面上的名字 |

| 名称 | 类型 | 说明 |
|---------|--------|---|
| avatar | string | 用户头像，可为imgN表示默认用户头像，或 http://XXXXXX 表示微信微博链接的头像 |
| info | String | 用户签名信息 |
| priv | int | <=0: 被封禁 1: 普通用户 2: 管理员 3: 超级管理员 |
| company | string | 企业ID |

Example

GET /api/user/18600028197?token=UKDFJOE382SFHHFV653AJF2319EKFSJKG

Return

```
{
  "code": "", "message": "", "data": { "username": "18600028197", "access": "", "nickname": "xiaoming", "avatar": "img3", "info": "此人没有介绍", "priv": 1 }
}
```

GET /api/user/?token=UKDFJOE382SFHHFV653AJF2319EKFSJKG

Return

```
{
  "code": "", "message": "", "data": { "username": "18600028197", "access": "", "nickname": "xiaoming", "avatar": "img3", "info": "此人没有介绍", "priv": 1 }
}
```

GET /api/user/hello

Return { "code": "400", "message": "找不到这个用户" }

发送验证码

向手机发送验证码，返回是否成功发送。验证码有效期为 10分钟。

POST /api/sendcode/{PhoneNumber}

PhoneNumber 为手机号码

返回数据参数表

无

Example

Post /api/sendcode/18600028197

Return {"code": "", "message": "", "data": {}}

Post /api/sendcode/12345

Return {"code": "400", "message": "无效的电话号码" }

登录微信或微博用户

POST /api/login2

使用微信或者微博来登录用户

BODY参数列表

| 名称 | 类型 | 是否必选 | 说明 |
|--------|--------|------|------------------------------|
| type | string | Yes | 登录服务商，目前只有 weixin或者 weibo 两种 |
| access | string | Yes | 用户登录微信或微博后返回的信息 |

返回数据参数表

| 名称 | 类型 | 说明 |
|----------|--------|-------------------------|
| token | string | AccessToken字符串，有效时间为1个月 |
| username | string | 用户名（只作为标识，不显示） |
| nickname | string | 用户昵称（显示名） |
| avatar | string | 头像地址 |
| info | string | 其他信息 |

Example

```
POST      /api/login2
BODY      { "type": "weixin", "access": "code:abcde" }
Return    { "code": "", "message": "", "data": { "token":
"UKDFJOE382SFHHFV653AJF2319EKFSJKG", "name": "wx_39128",
"nickname": "xiaoming", "avatar": "http://wx.qlogo.cn/mmopen/
g3MonUZtNHkdmzicllibx6iaFqAc56vxLSUfpb6n5WKSYVY0ChQKkiaJSgQ1dZuTOgvLLrhJbERQ
Q4eMsv84eavHiaiceqxibJxCfHe/0", "info": "", "sex": 1}}
```

```
POST      /api/login2
BODY      { "type": "weibo", "access": "access_token:yuw7328326wdw;uid:abw9230iowri"
Return    {
"code": "",
"message": "",
"data": {
"token": "UKDFJOE382SFHHFV653AJF2319EKFSJKG",
"name": "wb_82923",
"nickname": "XiaoLi",
"avatar": "http://tp1.sinaimg.cn/1404376560/180/0/1",
"info": "Good one",
"sex": 2
```

```
}  
}
```

```
POST      /api/login2  
BODY      { "type": "weixin", "access": "" }  
Return    {"code": "403", "message": "Invalid data"}
```

登录用户

POST /api/login/{Username}/{code}

使用用户名和验证码来登录用户

BODY数据参数表

| 名称 | 类型 | 是否必选 | 说明 |
|----------|--------|------|---------|
| password | string | No | 密码的 MD5 |

返回数据参数表

| 名称 | 类型 | 说明 |
|-------|--------|-------------------------|
| token | string | AccessToken字符串，有效时间为1个月 |

Example

```
POST      /api/login/18600028197/12345  
BODY      { "password": "abc123" }  
Return    { "code": "", "message": "", "data": { "token":  
"UKDFJOE382SFHHFV653AJF2319EKFSJKG" }}
```

```
POST      /api/login/abc/12345  
Return    {"code": "403", "message": "Invalid user or passcode"}
```

设置用户数据

使用token设置用户数据。

PUT /api/user/{username}?token={token}

BODY参数列表

| 名称 | 类型 | 是否必选 | 说明 |
|--------|--------|------|---------------------|
| avatar | string | Yes | 用户头像的JPG文件的Base64 值 |

| 名称 | 类型 | 是否必选 | 说明 |
|----------|--------|------|--|
| nickname | string | No | 用户显示在界面上的名字 |
| avatar | string | No | 用户头像，可为imgN表示默认用户头像，或http://XXXXXX 表示微信微博链接的头像 |
| info | string | No | 用户签名信息 |
| pw | string | No | 密码的md5 |

返回数据参数表

无

Example

```
PUT      /api/user/18600028197?token=UKDFJOE382SFHHFV653AJF2319EKFSJKG
BODY     { "nickname": "xiaoming", "avatar": "img2", "info": "此人没有介绍" }
Return   { "code": "", "message": "用户数据设置完成", "data": {} }
```

```
PUT      /api/user/abcde?token=UKDFJOE382SFHHFV653AJF2319EKFSJKG
BODY     { "nickname": "xiaoming", "avatar": "img2", "info": "此人没有介绍" }
Return   { "code": "400", "message": "用户不存在" }
```

上传用户头像

用户上传自己的头像

POST /api/upload_avatar?token={token}

BODY参数列表

Example

(avatar中应该是jpg文件的base64值)

```
POST      /api/upload_avatar?token=UKDFJOE382SFHHFV653AJF2319EKFSJKG
BODY      { "avatar": "SGVsbG8sd29ybGQ="}
Return    { "code": "", "message": "OK", "data": {} }
```

获取所有视频类型

GET /api/videotypes

返回数据参数表

将返回一个类型数组，其中单个类型的参数如下表

| 名称 | 类型 | 说明 |
|---------|--------|-------------------------|
| name | string | AccessToken字符串，有效时间为1个月 |
| icon | string | 图标链接 |
| display | string | 显示名称 |

Example

```
GET      /api/videotypes
Return    {
  "code": "",
  "message": "",
  "data": { "types": [
    {
      "name": "game",
      "icon": "http://pic.gvrcraft.com/sport.png",
      "display": "赛事"
    },
    {
      "name": "knowledge",
      "icon": "http://pic.gvrcraft.com/knowledge.png",
      "display": "知识"
    }
  ]
}
```

获取某种类型的轮播列表

GET /api/barrier/{videotype}

返回数据参数表

| 名称 | 类型 | 说明 |
|-------------------|--------|--|
| id | string | 视频ID |
| title | string | 视频显示名称 |
| url | string | 视频的链接 |
| duration | int | 视频长度，单位是秒；当这不是视频而是链接时，数据为0；这项数据也用来判断是否是一个链接（是否等于0） |
| description | string | 视频内容描述 |
| preview | string | 视频预览 |
| like | int | 视频的点赞数，链接没有这项数据 |
| viewSeconds | int | 视频的观看秒数总和，链接没有这项数据 |
| currentViewPeople | int | 视频的当前在线观看IP数，链接没有这项数据 |
| totalViewPeople | int | 视频的总观看IP数，链接没有这项数据 |

Example

GET /api/barrier/game

```
Return {
  "code": "",
  "message": "",
  "data": { "barriers": [
    {
      "id": "video1",
      "title": "video1",
      "url": "http://video.gvrcraft.com/water1.mp4",
      "duration": 70,
      "description": "这是个好视频",
      "preview": "http://video.gvrcraft.com/water1.jpg",
      "like": 3456,
      "viewSeconds": 829122,
      "currentViewPeople": 230,
      "totalViewPeople": 574
    },
    {
      "id": "video2",
      "title": "video2",
      "url": "http://gvrcraft.com/me.html?Id",
      "duration": 0,
      "description": "这是个链接",
    }
  ]
}
```

```
    "preview": "http://video.gvrcraft.com/aqua.jpg"
  }
}]
}
```

```
GET      /api/barrier/none
Return   {
  "code": "",
  "message": "",
  "data": { "barriers": []}
}
```

```
GET      /api/barrier/test
Return   {
  "code": "400",
  "message": "无效的类型"
}
```

获取热词

GET /api/hotwords

返回数据为一个字符串列表。

Example

```
GET      /api/hotwords
Return   {
  "code": "",
  "message": "",
  "data": { "words": [
    "good",
    "cool"
  ]}
}
```

获取闪屏页面

GET /api/splash/{deviceType}

deviceType 是 iOS, Android 等

返回数据参数表

| 名称 | 类型 | 说明 |
|------|--------|------------------------|
| url | string | 闪屏的链接，可以是图片 |
| type | string | 链接类型，可以是 video 或者 html |

| 名称 | 类型 | 说明 |
|----------|--------|--|
| param | string | 链接参数,type 为video时, 就是videoid, 表示点击后跳转到这个视频播放; type 为 html时, 就是网页的url, 点击后展示这个网页的内容 |
| duration | int | 闪屏持续时间, 单位是秒 |

Example

GET /api/splash/iOS

```
Return {
  "code": "",
  "message": "",
  "data": {
    "url": "http://video.gvrcraft.com/aqua.jpg",
    "type": "video",
    "param": "video2",
    "duration": 3
  }
}
```

从Videoid获取视频

GET /api/video/{videoid}

返回数据参数表

| 名称 | 类型 | 说明 |
|-------------------|--------|--------------|
| id | string | 视频ID |
| title | string | 视频标题 |
| url | string | 视频的链接 |
| duration | int | 视频长度, 单位是秒 |
| description | string | 视频内容描叙 |
| preview | string | 视频预览 |
| like | int | 视频的点赞数 |
| viewSeconds | int | 视频的观看秒数总和 |
| currentViewPeople | int | 视频的当前在线观看IP数 |
| totalViewPeople | int | 视频的总观看IP数 |
| comment | int | 当前评论数 |
| favorite | int | 视频收藏数 |

Example

GET /api/video/video2

```
Return {
  "code": "",
  "message": "",
  "data": {
    "id": "video2",
    "title": "video2",
    "url": "http://video.gvrcraft.com/aqua.mp4",
    "duration": 80,
    "description": "这也是个好视频",
    "preview": "http://video.gvrcraft.com/aqua.jpg",
    "like": 3476,
    "viewSeconds": 829133,
    "currentViewPeople": 260,
    "totalViewPeople": 594,
    "comment": 2,
    "favorite": 1
  }
}
```

GET /api/video/invalid1

```
Return {
  "code": "400",
  "message": "找不到视频"
}
```

获取某种类型的视频列表

GET /api/videos/{videotype}?page={N}&filter={filter}

获取某种类型的视频列表第N页，如果不加 page 就返回第一页。

videotype 为 all 时，表示列出第一页的所有视频。

如果存在 filter，那么显示 title 或者 description 包含 filter 的所有视频。

返回数据参数表

videos： 视频列表，类型同 GET /api/video/{videoid}

total： 表示一共有多少个数据，类型为 integer

Example

GET /api/videos/game

```
Return {
  "code": "",
  "message": "",
  "data": { "videos": [
    {
      "id": "video1",
      "title": "video1",
      "url": "http://video.gvrcraft.com/water1.mp4",

```

```

    "duration": 70,
    "description": "这是个好视频",
    "preview": "http://video.gvrcraft.com/water1.jpg",
    "like": 3456,
    "viewSeconds": 829122,
    "currentViewPeople": 230,
    "totalViewPeople": 574,
    "comment": 2,
    "favorite": 1
  },
  {
    "id": "video2",
    "title": "video2",
    "url": "http://video.gvrcraft.com/aqua.mp4",
    "duration": 80,
    "description": "这也是个好视频",
    "preview": "http://video.gvrcraft.com/aqua.jpg",
    "like": 3476,
    "viewSeconds": 829133,
    "currentViewPeople": 260,
    "totalViewPeople": 594,
    "comment": 2,
    "favorite": 0
  }
], "total": 2}
}

```

```

GET      /api/videos/all?filter=o2
Return   {
  "code": "",
  "message": "",
  "data": { "videos": [
    {
      "id": "video2",
      "title": "video2",
      "url": "http://video.gvrcraft.com/aqua.mp4",
      "duration": 80,
      "description": "这也是个好视频",
      "preview": "http://video.gvrcraft.com/aqua.jpg",
      "like": 3476,
      "viewSeconds": 829133,
      "currentViewPeople": 260,
      "totalViewPeople": 594,
      "comment": 2,
      "favorite": 1
    }
  ]}, "total": 1}
}

```

```

GET      /api/videos/game?page=2
Return   {
  "code": "",
  "message": "",
  "data": { "videos": [], "total": 2 }
}

```

```

GET      /api/videos/invalid

```

```
Return      {
  "code": "400",
  "message": "无效的类型"
}
```

把某个video加为收藏

POST /api/favorite/{videoid}?token={token}

Videoid 为视频的 ID

BODY数据参数表

无

返回数据参数表

无

Example

```
POST      /api/favorite/video1?token=12345
Return    {
  "code": "",
  "message": "设置成功",
  "data": {}
}
```

把某个video取消收藏

POST /api/unfavorite/{videoid}?token={token}

Videoid 为视频的 ID

当videoid为 all 时，表示取消所有收藏

BODY数据参数表

无

返回数据参数表

无

Example

```
POST      /api/unfavorite/video1?token=12345
Return    {
  "code": "",
  "message": "设置成功",
  "data": {}
}
```

获取某个用户的收藏视频

GET /api/favorite?token={token}

返回数据参数表

favorite: 视频列表，类型同 **GET /api/video/{videoid}**

Example

GET /api/favorite?token=12345

```
Return {
  "code": "",
  "message": "",
  "data": { "favorite": [
    {
      "id": "video1",
      "title": "video1",
      "url": "http://video.gvrcraft.com/water1.mp4",
      "duration": 70,
      "description": "这是个好视频",
      "preview": "http://video.gvrcraft.com/water1.jpg",
      "like": 3456,
      "viewSeconds": 829122,
      "currentViewPeople": 230,
      "totalViewPeople": 574,
      "comment": 2,
      "favorite": 1
    },
    {
      "id": "video2",
      "title": "video2",
      "url": "http://video.gvrcraft.com/aqua.mp4",
      "duration": 80,
      "description": "这也是个好视频",
      "preview": "http://video.gvrcraft.com/aqua.jpg",
      "like": 3476,
      "viewSeconds": 829133,
      "currentViewPeople": 260,
      "totalViewPeople": 594,
      "comment": 2,
      "favorite": 0
    }
  ]}
}
```

GET /api/favorite?token=invalid

Return {"code": "400", "message": "用户不存在"}

评论某个video（弹幕）

POST /api/comment/{videoid}?token=12345

Videoid 为视频的 ID

BODY参数表

| 名称 | 类型 | 是否必选 | 说明 |
|------------------|--------|------|--------------------------------|
| type | int | YES | 评价类型 0: 文字(Default) 1: 赞 |
| message | string | YES | 评价内容 |
| reply | int | NO | 本评论回复了之前哪一条评论 |
| videotime | int | NO | 只在录播中使用，表示现在视频播放到多少秒了 |

返回数据参数表

| 名称 | 类型 | 说明 |
|--------------|-----|-----------------------|
| type | int | 评价类型 0: 文字 1: 赞 |
| count | int | 最新评论总数或点赞总数 |

Example

POST /api/comment/video1
BODY { "message": "hello" }

```
Return        {  
  "code": "",  
  "message": "设置成功",  
  "data": { "type": 0, "count": 5 }  
}
```

POST /api/comment/video1
BODY { "type": 1, "message": "5" }

```
Return        {  
  "code": "",  
  "message": "设置成功",  
  "data": { "type": 0, "count": 523 }  
}
```

获取某个video的评论

GET /api/comment/{videoid}

获取视频的第1页评论（弹幕）

或者

GET /api/comment/{videoid}?page={N}

获取第N页评论（弹幕）

按照时间排序，最新的在最前面。

或者

GET /api/comment/{videoid}?after={TIME}

获取在TIME后出现的评论（弹幕），TIME 为 UTC 时间1970年至今的秒数

返回数据参数表

comments 中列出当前页的所有评论

| 名称 | 类型 | 说明 |
|------------------|--------|-----------------------|
| datetime | string | 时间 |
| nickname | string | 用户 |
| avatar | string | 用户头像 |
| message | string | 评论信息 |
| videotime | int | 只在录播中使用，表示现在视频播放到多少秒了 |

total 中显示所有评论（弹幕）的数量

time 成功返回时的时间，用于 **after** 中

Example

GET /api/comment/video1

```
Return {
  "code": "",
  "message": "",
  "data": {
    "comments": [
      {
        "datetime": "2017-06-21 14:03",
        "nickname": "user1",
        "avatar": "img1",
        "message": "Hello",
        "videotime": 5,
        "datetime": "2017-06-23 16:22",
        "nickname": "user2",
        "avatar": "img3",
        "message": "OK",
        "videotime": 20
      }
    ],
    "total": 2,
    "time": 16223829
  }
}
```

GET /api/comment/video1?after=1000000

```
Return {
  "code": "",
```

```
"message": "",
"data": {
  "comments": [
  ],
  "total": 2, "time": 16223829
}
}
```

获取服务器参数

GET /api/settings/{deviceType}
deviceType 是 iOS, Android

返回数据参数表

| 名称 | 类型 | 说明 |
|--------------|-----|----------|
| liveRefresh | int | 直播下的刷新时间 |
| videoRefresh | int | 录播下的刷新时间 |

Example

```
GET      /api/settings/iOS
Return   {
  "code": "",
  "message": "",
  "data": {
    "liveRefresh": 3,
    "videoRefresh": 30
  }
}
```

用密码登录用户

POST /api/login3/{Username}/{pw}
使用用户名和md5(密码)来登录用户

返回数据参数表

| 名称 | 类型 | 说明 |
|-------|--------|-------------------------|
| token | string | AccessToken字符串，有效时间为1个月 |

Example

```
POST      /api/login3/18600028197/12345
```


Return { "code": "", "message": "", "data": { "token":
"UKDFJOE382SFHHFV653AJF2319EKFSJKG" } }

POST /api/login/abc/12345
Return { "code": "403", "message": "Invalid user or passcode" }

登出用户

POST /api/logout?token={token}
登出当前token，即释放token

获取企业信息

GET /api/company/{company}
获取企业信息

返回数据参数表

| 名称 | 类型 | 说明 |
|-------------|--------|---------------------------|
| name | string | 企业名（Index，不能重复） |
| number | int | 企业人数 |
| admin | string | Admin的帐号ID |
| description | string | 企业描述 |
| info | string | 其他信息 |
| priv | int | 权限（-1：已删除，0：不可用， 1：正常） |

设置企业信息

PUT /api/company/{company}
设置企业信息，包括删除企业（即把priv设为 -1），创建企业，信息同上表

获取视频流信息

GET /api/stream/{streamID}

获取视频流信息

返回数据参数表

| 名称 | 类型 | 说明 |
|-----------------|--------|---|
| id | string | 视频流ID |
| name | string | 视频流名称 |
| owner | string | 流的主人 |
| description | string | 描述 |
| info | string | 其他信息 |
| priv | int | 权限（-1：已删除，0：不可用，1：正常） |
| status | int | -1: inactive 0: rtmp 被推流中 1: 拉流中 |
| rtmp | string | rtmp推流地址 |
| pull | string | 拉流地址 |
| url | string | 观看地址 |
| from | int | 允许使用时间 from |
| to | int | 允许使用时间 to |
| security_status | int | 使用的安全设置（1：加密，2：水印，4：911） |
| vtype | int | 0：正常，1：上下格式，2：鱼眼格式fsxs，3：倒鱼眼rfsxs，4：鱼眼格式2 fsxs2，5：鱼眼格式3 fsxs3 |

设置视频流信息

PUT /api/stream/{streamID}

设置视频流信息，同上表

获取视频点播信息

GET /api/vod/{vodID}

获取视频点播信息

返回数据参数表

| 名称 | 类型 | 说明 |
|--------------|--------|---|
| id | string | 视频点播ID |
| name | string | 名称 |
| owner | string | vod的主人 |
| description | string | 描叙 |
| info | string | 其他信息 |
| priv | int | 权限（-1：已删除，0：不可用，1：正常） |
| dest_stream | string | 如果播放，那么在哪个流播出 |
| video_status | int | 使用的安全设置（1：加密，2：水印，4：压缩，8：911） |
| duration | int | 时长（单位秒） |
| width | int | 宽度 |
| height | int | 高度 |
| vtype | int | 0：正常，1：上下格式，2：鱼眼格式fsxs，3：倒鱼眼rfsxs，4：鱼眼格式2 fsxs2，5：鱼眼格式3 fsxs3 |

设置视频流信息

PUT /api/vod/{vodID}

设置视频流信息，同上表

获取设备信息

GET /api/device/{deviceId}

获取设备信息

返回数据参数表

| 名称 | 类型 | 说明 |
|----------------|--------|----------------------------|
| id | string | 设备ID |
| name | string | 名称 |
| description | string | 描述 |
| sid | string | 安全用字符串 |
| priv | int | 权限（-1：已删除，0：设备不可用或损坏，1：正常） |
| playing_stream | string | 正在播放的流的videoID |
| time | int | 正在播放的流的播放位置 |
| info | string | 其他信息 |

设置视频流信息

PUT /api/device/{vodID}

设置视频流信息，同上表