

# 接口概述

## API 调用说明

本文档中，所有调用智能床服务端接口的请求基于 http 协议，需要携带 header 和 body 进行数据请求和验证，并且所有接口都需要按此规则校验。

## API checksum 校验

以下参数需要放在 Http Request Header 中

参数	参数说明
AppKey	平台分配的 appkey
Nonce	随机数（最大长度 128 个字符,最小长度为 4 个字符）
CurTime	当前 UTC 时间戳，从 1970 年 1 月 1 日 0 点 0 分 0 秒开始到现在的秒数 (String)
CheckSum	SHA1(AppSecret + Nonce + CurTime),三个参数拼接的字符串，进行 SHA1 哈希计算，转化成 16 进制字符(String，小写) AppSecret 为平台分配的 appSecret

CheckSum 有效期：出于安全性考虑，每个 checkSum 用完即失效，每次请求都必须生成新的 checkSum，同时请确认发起请求的服务器是与标准时间同步的，比如有 NTP 服务。

CheckSum 检验失败时会返回 414 错误码，具体参看 [code 状态表](#)。

**重要提示：**本文档中提供的所有接口均面向开发者服务器端调用，用于计算 CheckSum 的 AppSecret 开发者应妥善保管,可在应用的服务器端存储和使用。

计算 CheckSum 的 java 代码举例如下：

```
import java.security.MessageDigest;

public class CheckSumBuilder {
```

```
// 计算并获取 CheckSum
```

```
public static String getChecksum(String appSecret, St
```

```
ring nonce, String curTime) {
```

```
return encode("sha1", appSecret + nonce + curTim
```

```
e);
```

```
}
```

```
// 计算并获取 md5 值
```

```
public static String getMD5(String requestBody) {
```

```
return encode("md5", requestBody);
```

```
}
```

```
private static String encode(String algorithm, String
```

```
value) {
```

```
if (value == null) {
```

```
return null;
```

```
}
```

```
try {
```

```
    MessageDigest messageDigest
```

```
        = MessageDigest.getInstance(algorithm);
```

```
    messageDigest.update(value.getBytes());
```

```
    return getFormattedText(messageDigest.digest
```

```
        ());
```

```
    } catch (Exception e) {
```

```
        throw new RuntimeException(e);
```

```
    }
```

```
}
```

```
private static String getFormattedText(byte[] bytes)
```

```
{
```

```
    int len = bytes.length;
```

```
    StringBuilder buf = new StringBuilder(len * 2);
```

```
for (int j = 0; j < len; j++) {
```

```
    buf.append(HEX_DIGITS[(bytes[j] >> 4) & 0x0
```

```
f]);
```

```
    buf.append(HEX_DIGITS[bytes[j] & 0x0f]);
```

```
}
```

```
return buf.toString();
```

```
}
```

```
private static final char[] HEX_DIGITS = { '0', '1',
```

```
'2', '3', '4', '5',
```

```
'6', '7', '8', '9', 'a', 'b', 'c', 'd', 'e', 'f
```

```
' };
```

```
}
```

## 接口重要声明

1. 所有接口都只支持 POST 请求;

2. 所有接口请求 Content-Type 类型为:  
application/x-www-form-urlencoded;charset=utf-8;
3. 所有接口返回类型为 JSON，同时进行 UTF-8 编码
4. 所有接口请求参数放入 body 中，同时进行 UTF-8 编码

## 接口列表

### 设置回调通知 url 地址

#### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/setnoticeurl
```

```
HTTP/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

#### 接口描述

智能床平台回调通知分为以下几种：

实时设备监测数据、实时报警数据、设备心跳数据、其他类型数据

当设置回调 url 后，智能床平台会将通知数据 post 到设置的 url 地址中（请确保该地址支持接收 utf8 编码的 json 数据），具体回调数据格式请参见 [回调数据一览](#)

参数说明

参数	类型	必须	说明
callbackurl	String	是	需采用 base64 将回调完整 url 进行编码

curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vmel1kglw0803mgq3"
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
"Content-Type: application/x-www-form-urlencoded" -d
'callbackurl=
aHR0cCUzQS8vd3d3LnVzbGVlcGNhcmUuY29tL2FwaS96bmMvc2V0bm90
aWNldXJs' 'http://{智能床接口域名:端
口}/api/znc/setnoticeurl'
```

返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 登录（可选）

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/login HTTP/1.
```

```
1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

可选接口，用于验证和记录请求方的连接信息

参数说明

参数	类型	必须	说明
version	String	是	版本号
clienttype	int	是	客户端类型 1: c# sdk,2:java sdk,3:ios sdk,4: android sdk,5:js sdk,6:其他

curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vmel1kglw0803mgq3"
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
"Content-Type: application/x-www-form-urlencoded" -d '
version=1.0&clienttype=1' 'http://{智能床接口域名:端
口}/api/znc/login'
```

返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"

{
```



```
"code":200
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 退出（可选）

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/loginout HTTP
```

```
/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

可选接口，用于验证和记录请求方的退出信息

## 参数说明

无

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vme11kglw0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "Checksum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
version=1.0&clienttype=1' 'http://{智能床接口域名:端
```

```
口}/api/znc/loginout'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 获取指定病区指定床号对应的设备 ID

### 请求说明

```
POST http://{智能床接口域名:端口}/api/znc/getequipmenti
```

```
d HTTP/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 **Headers** 的设置请参考[接口概述](#)

### 接口描述

获取指定病区指定床位的对应的设备 ID，通常用于外部系统展示某个床位上绑定的设备 ID  
或用于过滤推送的设备数据匹配到指定的床位上

### 参数说明

参数	类型	必须	说明
partno	String	是	病区号
bedno	String	是	床位号

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vme11kg1w0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
partno=0001&bedno=1' 'http://{智能床接口域名:端口}/api/znc/
```

```
getequipmentid'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"equipmentid":"030500000011"
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 获取指定病区指定床号/设备 ID 对应的历史体征（按天）

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/getdaysignhis
```

```
tory HTTP/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

### 接口描述

获取指定病区指定床位/设备 ID 的对应的历史体征分析数据

### 参数说明

参数	类型	必须	说明
partno	String	是	病区号
bedno	String	是	床位号
equipmentid	String	否	可选参数，一旦赋值了 优先去找这个设备 id 对应的记

参数	类型	必须	说明
			录
begindate	String	是	格式 yyyy-MM-dd,例如 2017-05-05
enddate	String	是	格式 yyyy-MM-dd,例如 2017-05-07

### curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vmel1kglw0803mgq3"
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
-H "Checksum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
"Content-Type: application/x-www-form-urlencoded" -d '
partno=0001&bedno=1&begindate=2017-05-05&enddate=2017-05
-07' 'http://{智能床接口域名:端口}/api/znc/
getsigndayhistory'
```

### 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
{
```

```
"code":200,
```

```
"signinfos":
```

```
[
```

```
{ "partno": "病区号", "bedno": "床位号", "username": "病人
```

```
姓名", "avghr": "平均心率", "avgrr": "平均呼吸", "avgmv": "平均体
```

```
动", "turntimes": "翻身总次数", "date": "2017-05-07"},
```

```
{ "partno": "病区号", "bedno": "床位号", "username": "病人
```

```
姓名", "avghr": "平均心率", "avgrr": "平均呼吸", "avgmv": "平均体
```

```
动", "turntimes": "翻身总次数", "date": "2017-05-07"}]
```

```
]
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

# 获取指定病区指定床号/设备 ID 对应的历史体征（按小时）

## 请求说明

POST http://{智能床接口域名: 端口}/api/znc/gethoursigni

story HTTP/1.1

Content-Type:application/x-www-form-urlencoded;charse

t=utf-8

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

获取指定病区指定床位/设备 ID 的对应的历史体征分析数据

## 参数说明

参数	类型	必须	说明
partno	String	是	病区号
bedno	String	是	床位号
equipmentid	String	否	可选参数，一旦赋值了 优先去找这个设备 id 对应的记录
beginhour	String	是	格 式 yyyy-MM-dd-HH-00-00, 例 如 2017-05-05-20-00-00
endhour	String	是	格 式 yyyy-MM-dd-HH-00-00, 例 如 2017-05-05-22-00-00



## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vmel1kg1w0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
partno=0001&bedno=1&beginhour=2017-05-05-20-00-00&endhou
```

```
r=2017-05-05-22-00-00' 'http://{智能床接口域名:端
```

```
口}/api/znc/gethoursignhistory'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"signinfos":
```

```
[
```

```
{
  "partno": "病区号",
  "bedno": "床位号",
  "username": "病人
姓名",
  "avghr": "平均心率",
  "avgrr": "平均呼吸",
  "avgmv": "平均体
动",
  "turntimes": "翻身总次数",
  "hour": "2017-05-05 20:00:00"
},
{
  "partno": "病区号",
  "bedno": "床位号",
  "username": "病人
姓名",
  "avghr": "平均心率",
  "avgrr": "平均呼吸",
  "avgmv": "平均体
动",
  "turntimes": "翻身总次数",
  "hour": "2017-05-05 21:00:00"
}
]
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

# 获取指定病区指定床号/设备 ID 对应的历史在离床

## 请求说明

POST `http://{智能床接口域名: 端口}/api/znc/getbedhistory`

HTTP/1.1

Content-Type: `application/x-www-form-urlencoded; charset=utf-8`

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

获取指定病区指定床位/设备 ID 的对应的历史在离床数据

## 参数说明

参数	类型	必须	说明
partno	String	是	病区号
bedno	String	是	床位号
equipmentid	String	否	可选参数，一旦赋值了 优先去找这个设备 id 对应的记录
begintime	String	是	格 式      yyyy-MM-dd-HH-mm-ss,      例      如 2017-05-05-20-00-00
endtime	String	是	格 式      yyyy-MM-dd-HH-mm-ss,      例      如 2017-05-05-22-00-00

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vme11kg1w0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
partno=0001&bedno=1&begintime=2017-05-05-20-00-00&endtim
```

```
e=2017-05-05-22-00-00' 'http://{智能床接口域名:端
```

```
口}/api/znc/getbedhistory'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"bedinfos":
```

```
[
```

```
{ "partno": "病区号", "bedno": "床位号", "username": "病人
```

```
姓名", "status": 1 // 在床状态 1: 在床 2: 离床, "timespan": "2017
```

```
-05-05 20:00:00 至 2017-05-05 21:00:00" },
```

```
{ "partno": "病区号", "bedno": "床位号", "username": "病人
```

```
姓名", "status": 2 // 在床状态 1: 在床 2: 离床, "timespan": "2017
```

```
-05-05 21:00:00 至 2017-05-05 22:00:00" }
```

```
]
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 获取指定病区指定床号/设备 ID 对应的报警历史

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/getalarmhisto
```

```
ry HTTP/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

获取指定病区指定床位/设备 ID 的对应的历史报警数据

## 参数说明

参数	类型	必须	说明
partno	String	是	病区号
bedno	String	是	床位号
equipmentid	String	否	可选参数，一旦赋值了 优先去找这个设备 id 对应的记录
type	int	是	0:全部 1：在离床报警 2：心率报警 3：呼吸报警 4：体动异常报警 5：翻身提醒
handleflag	int	否	0：未处理 1：已处理
begintime	String	是	格 式 yyyy-MM-dd-HH-mm-ss, 例 如 2017-05-05-20-00-00
endtime	String	是	格 式 yyyy-MM-dd-HH-mm-ss, 例 如 2017-05-05-22-00-00

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vme11kg1w0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
partno=0001&bedno=1&type=0&begintime=2017-05-05-20-00-00
```

```
&endtime=2017-05-05-22-00-00' 'http://{智能床接口域名:端
```

```
口}/api/znc/getalarmhistory'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"alarminfos":
```

```
[
```

	<code>{"partno":"病区号","bedno":"床位号","username":"病人</code>
	<code>姓名","type":1//报警类型,"handleflag":1//处理标志,"alarmtim</code>
	<code>e":"2017-05-05 21:00:00","alarmContent":"病人已经离床超过 3</code>
	<code>0 分钟，请及时关注！"}},</code>
	<code>{"partno":"病区号","bedno":"床位号","username":"病人姓</code>
	<code>名","type":2//报警类型,"handleflag":1//处理标志,"alarmtime</code>
	<code>":"2017-05-05 20:30:00","alarmContent":"病人心率持续超过 12</code>
	<code>0 次/分已经三分钟，请及时关注！"} }</code>
	<code>]</code>
	<code>}</code>

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)



# 获取指定病区指定床号/设备 ID 对应的历史睡眠

## 请求说明

POST http://{智能床接口域名: 端口}/api/znc/getsleephisto

ry HTTP/1.1

Content-Type:application/x-www-form-urlencoded;charse

t=utf-8

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

获取指定病区指定床位/设备 ID 的对应的历史睡眠数据

## 参数说明

参数	类型	必须	说明
partno	String	是	病区号
bedno	String	是	床位号
equipmentid	String	否	可选参数，一旦赋值了 优先去找这个设备 id 对应的记录
date	String	是	睡眠查询日期，格式 yyyy-MM-dd

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vme11kg1w0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
partno=0001&bedno=1&date=2017-05-05' 'http://{智能床接口域
```

```
名:端口}/api/znc/getsleephistory'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"equipmentid":"0305000000011"//设备 id,
```

```
"partno":"0001"//病区编号,
```

```
"bedno":"01"//床位号,
```

```
"signinfo":{
```

```
"maxhr":"90"//最大心率,
```

```
"minhr":"70"//最小心率,
```

```
"avghr":"78"//平均心率,
```

```
"maxrr":"30"//最大呼吸,
```

```
"minrr":"20"//最小呼吸,
```

```
"avgrr":"26"//平均呼吸
```

```
}//体征信息,
```

```
"sleeptimespan":"2017-05-05 21:00:20 至 2017-05-06 06:20:
```

```
00"//睡眠时间,段,
```

```
"onbedtimespan":"2017-05-05 21:00:00 至 2017-05-06 06:20:
```

```
00"//卧床时间段(开始上床睡觉-早上起床),
```

```
"onbedbegttime":"21: 00"//开始上床时间,
```

```
"sleeptideminutes":20//睡眠潜伏期, 单位分钟数,
```

```
"sleepminutes":"5 小时 40 分钟"//睡眠总时长，单位分钟数，
```

```
"onbedminutes":"6 小时"//在床总时长，单位分钟数，
```

```
"deepminutes":120//深睡时长，单位分钟数，
```

```
"lightminutes":220//浅睡时长，单位分钟数，
```

```
"wakeminutes":100//清醒时长，单位分钟数，
```

```
"truntimes":30//翻身总次数，
```

```
"leavebedtimes":1//离床总次数，
```

```
"leavebedinfos":[
```

```
{ "timespan":"2017-05-05 23:45:00 至 2017-05-05 23:55:
```

```
00", "minutes":10}
```

```
]//离床时间段，
```

```
"deepsleepinfos":[
```

```
{ "timespan":"2017-05-05 21:55:00 至 2017-05-05 22:55:
```

```
00", "minutes":60},
```

```
{"timespan":"2017-05-06 01:55:00 至 2017-05-06 02:55:
```

```
00","minutes":60}
```

```
]//深睡时间段,
```

```
"lightsleepinfos":[
```

```
{"timespan":"2017-05-05 22:55:00 至 2017-05-05 23:55:
```

```
00","minutes":60},
```

```
{"timespan":"2017-05-06 00:55:00 至 2017-05-06 01:55:
```

```
00","minutes":60},
```

```
{"timespan":"2017-05-06 03:55:00 至 2017-05-06 05:35:
```

```
00","minutes":100}
```

```
]//浅睡时间段,
```

```
"wakesleepinfos":[
```

```
{"timespan":"2017-05-05 01:00:00 至 2017-05-05 01:50:
```

```
00","minutes":50},
```

```
{"timespan": "2017-05-06 03:00:00 至 2017-05-06 01:50:"
```

```
00", "minutes": 50}
```

```
] // 清醒时间段
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 获取指定病区下未处理的报警信息

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/getunhandleal
```

```
arms HTTP/1.1
```

```
Content-Type: application/x-www-form-urlencoded; charse
```

```
t=utf-8
```

请求中 **Headers** 的设置请参考[接口概述](#)

## 接口描述

获取指定病区指定床位/设备 ID 的对应的历史未处理报警数据

## 参数说明

参数	类型	必须	说明
partno	String	是	病区号

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vmel1kglw0803mgq3"
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
"Content-Type: application/x-www-form-urlencoded" -d '
partno=0001' 'http://{智能床接口域名:端口}/api/znc/getunhandlealarms'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```

{
  "code":200,
  "alarminfos":
  [
    {
      "partno":"病区号", "bedno":"床位号", "username":"病人
      姓名", "type":1//报警类型, "alarmtime":"2017-05-05 21:00:00
      ", "alarmContent":"病人已经离床超过 30 分钟，请及时关注！" },
    {
      "partno":"病区号", "bedno":"床位号", "username":"病人姓
      名", "type":2//报警类型, "alarmtime":"2017-05-05 20:30:00",
      "alarmContent":"病人心率持续超过 120 次/分已经三分钟,请及时关注！
      " }
  ]
}

```

## 主要的返回码

200、404、414、500



具体请参考 [code 状态表](#)

## 获取设备 ID 对应的病区床位信息

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/getbedinfo HT
```

```
TP/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

### 接口描述

获取指定设备 ID 对应的病区床位信息

### 参数说明

参数	类型	必须	说明
equipmentid	String	是	设备 id

### curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vme11kglw0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "Checksum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
equipmentid=030500000011' 'http://{智能床接口域名:端
```

```
口}/api/znc/getbedinfo'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"equipmentid":"030500000011"//设备 id,
```

```
"partno":"0001"//病区编号,
```

```
"bedno":"01"//床位号
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 获取医院下所有病区列表

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/gethospitalpa
```

```
rts HTTP/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

### 接口描述

获取指定设备 ID 对应的病区床位信息

### 参数说明

参数	类型	必须	说明
hospitalcode	String	是	医院编码（一般有智能床预先提供）

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vme11kg1w0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
hospitalcode=0001' 'http://{智能床接口域名:端
```

```
口}/api/znc/gethospitalparts'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"hospitalcode":"0001"//医院编码,
```

```
"hospitalname":"测试医院"//医院名称,
```

```
"partinfos":
```

```
[
```

```
{ "partno": "病区号", "partname": "病区名称" },
```

```
{ "partno": "病区号", "partname": "病区名称" },
```

```
.....//多个病区集合信息
```

```
]
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

## 获取指定病区下的所有床位信息

### 请求说明

```
POST http://{智能床接口域名: 端口}/api/znc/getbedinfosof
```

```
part HTTP/1.1
```

```
Content-Type:application/x-www-form-urlencoded;charse
```

```
t=utf-8
```

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

获取指定设备 ID 对应的病区床位信息

## 参数说明

参数	类型	必须	说明
partno	String	是	病区号

## curl 请求示例

```
curl -X POST -H "AppKey: go9dnk49bkd9jd9vmel1kglw0803mgq3"
```

```
-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"
```

```
-H "CheckSum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
partno=0001' 'http://{智能床接口域名:端
```

```
口}/api/znc/getbedinfosofpart'
```

# 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"

{

  "code":200,

  "partno":"0001"//病区号,

  "partname":"病区名称"//病区名称,

  "roominfos":

  [

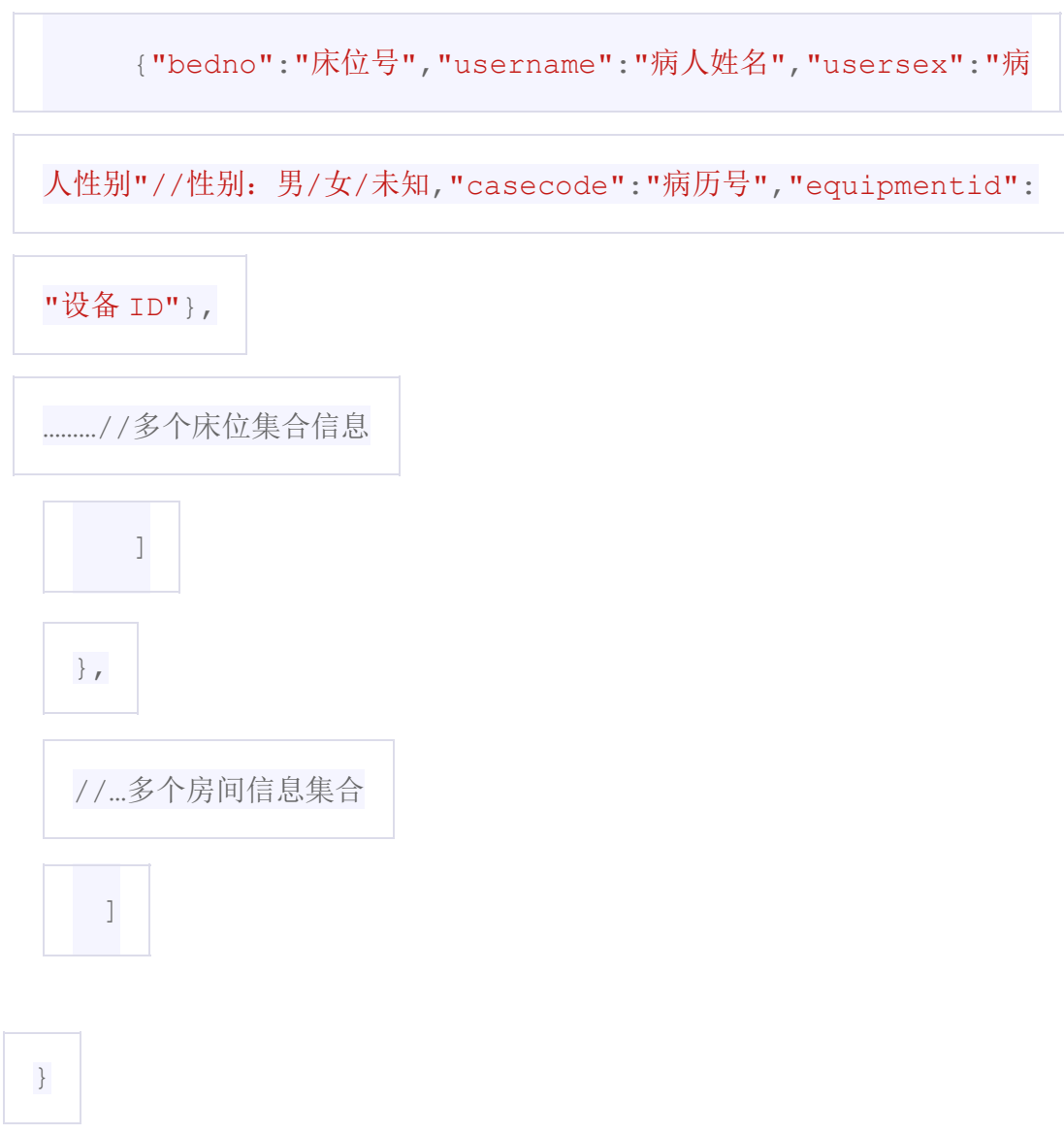
    { "roomno":"1"//房间号,

      "bedinfos":

      [

        { "bedno":"床位号", "username":"病人姓名", "usersex":"病人性别"//性别: 男/女/未知, "casecode":"病历号", "equipmentid":

          "设备 ID"},
```



## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)



# 获取指定病区下指定床位信息

## 请求说明

POST http://{智能床接口域名: 端口}/api/znc/getspecialbed

info HTTP/1.1

Content-Type:application/x-www-form-urlencoded;charse

t=utf-8

请求中 Headers 的设置请参考[接口概述](#)

## 接口描述

获取指定设备 ID 对应的病区床位信息

## 参数说明

参数	类型	必须	说明
partno	String	是	病区号
bedno	String	是	床位号

## curl 请求示例

curl -X POST -H "AppKey: go9dnk49bkd9jd9vmel1kglw0803mgq3"

-H "Nonce: 4tgggergigwow323t23t" -H "CurTime: 1443592222"

```
-H "Checksum: 9e9db3b6c9abb2e1962cf3e6f7316fcc55583f86" -H
```

```
"Content-Type: application/x-www-form-urlencoded" -d '
```

```
partno=0001&bedno=01' 'http://{智能床接口域名:端
```

```
口}/api/znc/getspecialbedinfo'
```

## 返回说明

http 响应:json

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"code":200,
```

```
"partno":"0001"//病区号,
```

```
"partname":"病区名称"//病区名称,
```

```
"roomno":"1"//所属房间号,
```

```
"bedno":"01"//床位号,
```

```
"equipmentid":"030100000011"//绑定设备 id,
```

```
"patientinfo":{
```

```
"usercode": "00001"//用户唯一编码,
```

```
"username": "张三"//病人姓名,
```

```
"usersex": "男"//性别: 男/女/未知,
```

```
"casecode": "his0001"//病区号,
```

```
"address": "地址"//地址,
```

```
"tel": "电话号码"//电话号码
```

```
}//床位用户信息,
```

```
}
```

## 主要的返回码

200、404、414、500

具体请参考 [code 状态表](#)

# 回调数据一览

## 实时设备监测数据

"Content-Type": "application/json; charset=utf-8"
{
"datatype":1//回调数据类型, 1: 实时监测数据 2: 设备心跳数据 3:
报警数据,
"equipmentid":"0305000000011"//设备 id,
"partno":"0001"//病区编号,
"bedno":"01"//床位号,
"bedstatus":1//在离床状态 1: 在床 2: 离床,
"lastedBedTime":"2017-05-06 20:00:00"//最近一次在离床时间
根据在离床状态, 如果当前是在床 则为最近一次上床时间 如果当前是离床
则为最近一次离床时间,
"hr":78//心率,

```
"rr":28//呼吸,
```

```
"mv":2999//体动,
```

```
"bodystatus":1//体姿状态 1: 平躺 2: 侧躺 3: 坐立,
```

```
"reporttimestamp":1443592222//监测数据时间,当前 UTC 时间戳,
```

```
从 1970 年 1 月 1 日 0 点 0 分 0 秒开始到现在的秒数 (String)
```

```
}
```

## 实时设备心跳数据

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"datatype":2//回调数据类型, 1: 实时监测数据 2: 设备心跳数据 3:
```

```
报警数据,
```

```
"equipmentid":"0305000000011"//设备 id,
```

```
"partno":"0001"//病区编号,
```

```
"bedno": "01"//床位号,
```

```
"status": 1//设备状态 1: 正常运行 2: 设备异常 3: 设备断电,
```

```
"reporttimestamp": 1443592222//监测数据时间, 当前 UTC 时间戳,
```

```
从 1970 年 1 月 1 日 0 点 0 分 0 秒开始到现在的秒数 (String)
```

```
}
```

## 实时报警数据

```
"Content-Type": "application/json; charset=utf-8"
```

```
{
```

```
"datatype": 3//回调数据类型, 1: 实时监测数据 2: 设备心跳数据 3:
```

```
报警数据,
```

```
"equipmentid": "0305000000011"//设备 id,
```

```
"partno": "0001"//病区编号,
```

```
"bedno": "01"//床位号,
```

"alarmtype":1//报警类型 1：在离床报警 2：心率报警 3：呼吸报警

4：体动异常报警 5：翻身提醒,

"alarmcontent":"已经离床超过 30 分钟，请及时关注病人情况"//报

警内容,

"alarmTime":"2017-05-05 17:30:30"//报警产生时间,

"reporttimestamp":1443592222//数据推送时间,当前 UTC 时间戳,

从 1970 年 1 月 1 日 0 点 0 分 0 秒开始到现在的秒数 (String)

}

## code 状态表

code	详细描述
200	操作成功
201	客户端版本不对，需升级 sdk
301	接入账户被封禁
302	接入用户名或密码错误
303	API 接口访问受限
315	IP 限制
404	查询结果为空

code	详细描述
405	参数长度过长
408	客户端请求超时
413	客户端认证参数有误
414	请求参数无效
415	客户端网络问题
431	HTTP 重复请求
500	服务器内部错误
503	服务器繁忙
998	解包错误
999	打包错误