

IOS App 的基础使用——连接、定位与通知消息

【操作步骤】

1. 登录与访问
2. ios 组件配置
3. 调用消息通知服务
4. 设置内网访问地址

【参考】

- HomeAssistant IOS App (请使用版本 1.5.0 及以上)
<https://itunes.apple.com/us/app/home-assistant-open-source-home-automation/id1099568401?mt=8>
- HomeAssistant IOS App 文档
<https://www.home-assistant.io/docs/ecosystem/ios/>

iBeacon 定位

【硬件准备】



【操作步骤】

1. 在树莓派上找到 iBeacon 的 uuid、major 和 minor
2. 在 HomeAssistant 中配置 iBeacon

【参考】

- iBeacon 的原理



- 在树莓派上扫描 iBeacon 设备

- 安装命令

```
sudo apt-get install python-dev libbluetooth-dev libcap2-bin  
sudo pip3 install beacontools[scan]
```

- iBeacon 扫描 python 程序

```
import time  
from beacontools import BeaconScanner  
  
def callback(bt_addr, rssi, packet, additional_info):  
    print("<%s, %d> %s %s" % (bt_addr, rssi, packet, additional_info))  
  
scanner = BeaconScanner(callback)  
scanner.start()  
time.sleep(10)  
scanner.stop()
```

- HA 中 iBeacon 的配置

```
# customize.yaml  
zone.home:  
  beacon:  
    uuid: 74278bda-b644-4520-8f0c-720eaf059935  
    major: 65504  
    minor: 61440
```

多媒体通知与静态文件 Web 服务

【操作步骤】

1. 发送特殊提示音的消息
2. 发送自定义提示音的消息
3. 静态文件 web 服务
4. 发送包含图片的消息
5. 发送定位信息地图的消息
6. 发送即时摄像头影像的消息
7. 消息次数限制

【参考】

- 设置提示音参考文档

<https://www.home-assistant.io/docs/ecosystem/ios/notifications/sounds/>

- 演示中发送的一些消息服务数据样例

```
{
  "message": "这是一条特殊提示音的消息",
  "data": {
    "push": {
      "sound": "US-EN-Morgan-Freeman-Wife-Is-Arriving.wav"
    }
  }
}

{
  "message": "这是一条包含图片的消息",
  "data": {
    "attachment": {
      "url": "https://zjs.duckdns.org/local/ARISTOTLE.jpg",
      "content-type": "jpg"
    }
  }
}

{
  "message": "这是一条包含地图的消息",
  "data": {
    "push": {
      "category": "map"
    },
    "action_data": {
      "latitude": "31.3015",
      "longitude": "121.5114"
    }
  }
}

{
  "message": "这是一条摄像头影像的消息",
  "data": {
    "push": {
      "category": "camera"
    },
    "attachment": {
      "content-type": "jpeg"
    },
    "entity_id": "camera.cam5"
  }
}
```

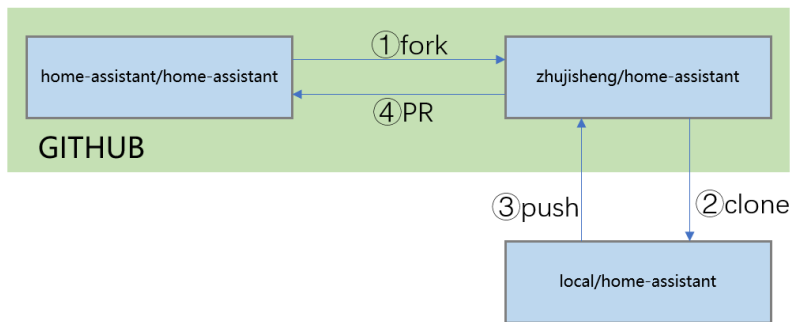
把你的代码贡献给组织

【操作步骤】

1. fork home-assistant 项目
2. 构建本地开发环境
3. 本地库修改与测试
4. 提交你的贡献
5. 提交网站文档修改

【参考】

- git 工具的介绍
<https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000>
- HomeAssistant 开发者指导
https://developers.home-assistant.io/docs/en/development_index.html
- 核心过程



- 构建开发环境命令
 - 安装基础软件库

```
sudo apt update
sudo apt-get install python3-pip python3-dev python3-venv
sudo apt-get install autoconf libssl-dev libxml2-dev libxslt1-dev libjpeg-dev libffi-dev libudev-dev zlib1g-dev
```
 - 创建/进入开发目录

```
mkdir my_development
cd my_development
```
 - 克隆远程库

```
git clone https://github.com/YOUR_GIT_USERNAME/home-assistant.git
cd home-assistant
git remote add upstream https://github.com/home-assistant/home-assistant.git
```
 - 构建 python 虚拟环境，从源码安装 HomeAssistant

```
python3 -m venv .
source bin/activate
pip3 install wheel
script/setup
```
- 修改与本地测试（以增加 image_processing.qrcode 为例）
 - 创建工作分支

```
git checkout -b qrcode
```
 - 修改

```
vi homeassistant/components/image_processing/qrcode.py # 编辑组件程序
script/gen_requirements_all.py # 生成新的 requirements_all.txt
vi .coveragerc # 编辑测试覆盖配置文件
```

➤ 代码规范检查

```
pip3 install -r requirements_test.txt
flake8 homeassistant/components/image_processing/qrcode.py
pydocstyle homeassistant/components/image_processing/qrcode.py
pylint homeassistant/components/image_processing/qrcode.py
```

➤ 完整测试

```
tox # 完整测试命令，运行时间非常长
```

➤ 确认修改

```
git add .
git commit -m "Added some feature" --amend --author="YourName <youremail@xxx.xxx>"
```

● 提交你的贡献

➤ 获取最新官方版本内容

```
git fetch upstream dev # 获取官方的当前分支
git rebase upstream/dev # 调基
```

➤ 将修改发布到你 fork 的项目中

```
git push origin HEAD
```

➤ 创建 Pull Request (在 github 网站上进行)

本视频中创建的 PR 链接: <https://github.com/home-assistant/home-assistant/pull/20215>