

uniapp 安卓开机启动原生插件

目 录

使用说明	1
使用方法	1.1
更新日志	1.2
插件方法	2
开启悬浮窗	2.1
设置延迟	2.2

使用方法

介绍

安卓开机启动原生插件，支持自定义延迟时间，是否开启开机启动

联系作者

关注微信公众号可联系作者



插件地址

<https://ext.dcloud.net.cn/plugin?id=14306>

权限

- android.permission.RECEIVE_BOOT_COMPLETED
- android.permission.SYSTEM_ALERT_WINDOW

用法

该插件一般用于终端只有一个程序或自启主程序，该插件不需要引入，只需要集成就可以实现自动启动程序，如果取消自动启动程序移除集成的插件即可，如果您需要设置延迟时间可引入以下代码

```
const module = uni.requireNativePlugin("leven-autoStart-AutoStartModule");
```

安卓8.0以上如果要实现开机启动需要打开以下权限

- 1.自启动权限
- 2.悬浮窗权限

页面内容

```
<template>
  <view>
    <uni-card title="开机自动启动程序插件">
      <view style="margin-bottom: 20px;">
        <button type="primary" @click="openSettingPage">开启悬浮窗</button>
        <button type="primary" @click="setDelay">设置延迟时间</button>
      </view>
      <view>
        <text style="font-size: 14px;">该插件一般用于终端只有一个程序或自启主程序，该插件不
      </view>
      <view>
        <text style="font-size: 14px;">插件使用的是广播通知方式自启动程序，如果不能启动请打
      </view>
      <view>
        <text style="font-size: 14px;">安卓8.0以上要实现开机启动需要打开以下权限：</text>
        <text style="font-size: 14px;">1.自启动</text>
        <text style="font-size: 14px;">2.悬浮窗</text>
      </view>
    </uni-card>
  </view>
</view>
</template>

<script>
  const module = uni.requireNativePlugin("leven-autoStart-AutoStartModule");
  export default {
    data() {
      return {}
    },
    methods: {
      // 打开设置页面
      openSettingPage() {
        module.openSettingPage(res => {
          uni.showToast({
            title: JSON.stringify(res),
            icon: "none"
          })
        })
      },
      // 设置延迟时间
      setDelay() {
```

使用方法

```
module.setDelay({
  delay: 30,
  debug: true,
  isAutoStart: true
}, res => {
  uni.showToast({
    title: JSON.stringify(res),
    icon: "none"
  })
})
}
}
}
</script>

<style>

</style>
```

更新日志

2023-09-05 v1.1.0

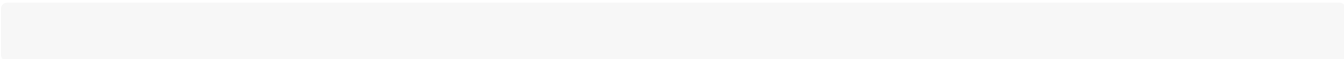
【新增】开启悬浮窗权限【[转到](#)】

2023-8-28

首次发布

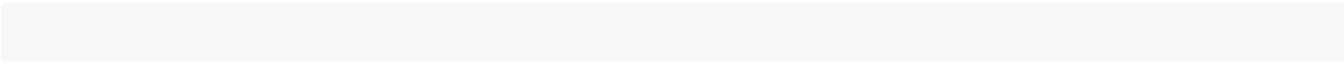
开启悬浮窗

- 用法如下：



- 参数说明
无

- 示例



- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置延迟

方法名

setDelay

用法

- 用法如下：

```
module.setDelay({
  delay: 30,
  debug: true,
  isAutoStart: true
}, res => {
  uni.showToast({
    title: JSON.stringify(res),
    icon: "none"
  })
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
delay	Integer	否	0	延迟时间，单位：秒
debug	Boolean	否	false	是否开启debug模式，如果开启的话会在启动广播时弹出Toast提示消息
isAutoStart	Boolean	否	true	是否开启开机启动

回调

- 示例

设置延迟

```
{
  "message": "设置成功",
  "code": 0,
  "data": {}
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败