

ionic 开发概要

阿里云物联网套件开发系列 (3)

欧志球

浙江机电职业技术学院

2019 年 9 月 2 日

目录

- 1 Ionic 简介
- 2 第一个 Ionic 程序
 - Ionic 安装配置
- 3 Ionic UI 组件
- 4 Ionic Native 组件



目录

1 Ionic 简介

2 第一个 Ionic 程序

3 Ionic UI 组件

4 Ionic Native 组件



Ionic 是什么?

Ionic 是开源的 UI 组件，用于采用 Web 技术 (HTML, CSS 和 JavaScript) 构建高性能、高质量的移动以及桌面应用。

Ionic 关注前端用户体验、或者 App 的 UI 交互。易学且与 *Angular* 无缝集成，同时还支持：

- Vue



Ionic 是什么?

Ionic 是开源的 UI 组件，用于采用 Web 技术 (HTML, CSS 和 JavaScript) 构建高性能、高质量的移动以及桌面应用。

Ionic 关注前端用户体验、或者 App 的 UI 交互。易学且与 Angular 无缝集成，同时还支持：

- Vue
- React



Ionic 的优势

- 跨平台。一套代码，同时支持在 iOS、Android、PWA 和桌面上运行；



Ionic 的优势

- 跨平台。一套代码，同时支持在 iOS、Android、PWA 和桌面上运行；
- 基于 Web 标准。采用 HTML, CSS 和 JavaScript 开发；



Ionic 的优势

- 跨平台。一套代码，同时支持在 iOS、Android、PWA 和桌面上运行；
- 基于 Web 标准。采用 HTML, CSS 和 JavaScript 开发；
- 设计美观



ionic 的优势

- 跨平台。一套代码，同时支持在 iOS、Android、PWA 和桌面上运行；
- 基于 Web 标准。采用 HTML, CSS 和 JavaScript 开发；
- 设计美观
- 简单易用



ionic 的劣势

- 性能不如原生代码；

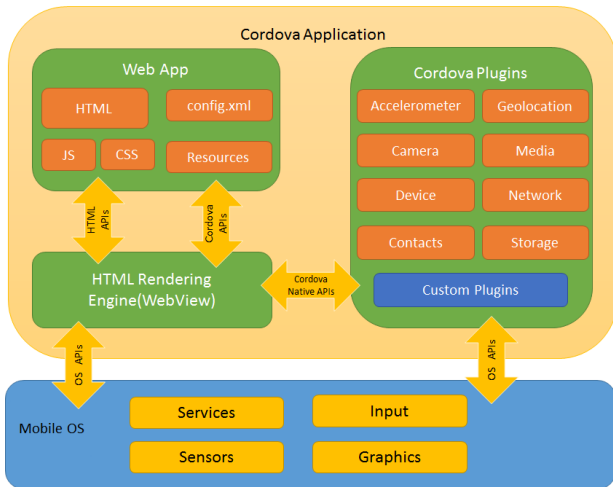


ionic 的劣势

- 性能不如原生代码；
- 某些底层调用或者无法使用，或者需要使用原生代码开发插件。



Ionic 的基本原理—Cordova



目录

- 1 Ionic 简介
- 2 第一个 Ionic 程序
 - Ionic 安装配置
- 3 Ionic UI 组件
- 4 Ionic Native 组件



Ionic Web 所需工具

- Git 用于代码版本管理;



Ionic Web 所需工具

- Git 用于代码版本管理;
- Node.js



Ionic Web 所需工具

- Git 用于代码版本管理;
- Node.js
- 最新版本的 Angular



Ionic Web 所需工具

- Git 用于代码版本管理;
- Node.js
- 最新版本的 Angular
- VSCode 或者其他代码编辑器, 比如 Notepad++



Ionic Web 所需工具

- Git 用于代码版本管理;
- Node.js
- 最新版本的 Angular
- VSCode 或者其他代码编辑器, 比如 Notepad++
- 命令行终端。Windows 下使用 cmd 或者 Powershell, Linux/Mac, 使用任何终端均可。



Ionic 安装和配置

- 假定您已经安装了 Node.js, 则安装步骤如下:

```
npm install -g ionic
```



Ionic 安装和配置

- 假定您已经安装了 Node.js, 则安装步骤如下:

```
npm install -g ionic
```

- 开启一个 App. 以下命令将创建一个 myApp 应用, 该应用以标签页 (Tab) 形式分为多个 UI.

```
ionic start myApp tabs
```



Ionic 安装和配置

- 假定您已经安装了 Node.js, 则安装步骤如下:

```
npm install -g ionic
```

- 开启一个 App. 以下命令将创建一个 myApp 应用, 该应用以标签页 (Tab) 形式分为多个 UI.

```
ionic start myApp tabs
```

- 在浏览器中运行该 App:

```
cd myApp
```

```
ionic serve
```



移动环境下配置—Android

注意：iOS 环境设置有所不同，未列出。具体可以参考[这里](#)。

Android 运行环境配置如下：

- 安装 JDK8;



移动环境下配置—Android

注意：iOS 环境设置有所不同，未列出。具体可以参考[这里](#)。

Android 运行环境配置如下：

- 安装 JDK8;
- 安装 Gradle 环境，参考[这里](#)获取安装信息;



移动环境下配置—Android

注意：iOS 环境设置有所不同，未列出。具体可以参考[这里](#)。

Android 运行环境配置如下：

- 安装 JDK8;
- 安装 Gradle 环境，参考[这里](#)获取安装信息;
- 安装 Android Studio，可以到[这里](#)下载;



移动环境下配置—Android

注意：iOS 环境设置有所不同，未列出。具体可以参考[这里](#)。

Android 运行环境配置如下：

- 安装 JDK8;
- 安装 Gradle 环境，参考[这里](#)获取安装信息;
- 安装 Android Studio，可以到[这里](#)下载;
- 安装 Android SDK. 打开 Android Studio，该 IDE 会提示需要安装 Android SDK。默认是安装最新的 SDK，你也可以在 Configure-SDK Manager 下安装其他 SDK 包;



移动环境下配置—Android

注意：iOS 环境设置有所不同，未列出。具体可以参考[这里](#)。

Android 运行环境配置如下：

- 安装 JDK8;
- 安装 Gradle 环境，参考[这里](#)获取安装信息;
- 安装 Android Studio，可以到[这里](#)下载;
- 安装 Android SDK. 打开 Android Studio，该 IDE 会提示需要安装 Android SDK。默认是安装最新的 SDK，你也可以在 Configure-SDK Manager 下安装其他 SDK 包;
- 配置环境变量，将 ANDROID_SDK_ROOT 设置为你安装了 SDK 的位置；将该位置下 tools/bin,platform-tools 和 emulator 加入 PATH 环境变量中；



移动环境下配置—Android

注意：iOS 环境设置有所不同，未列出。具体可以参考[这里](#)。

Android 运行环境配置如下：

- 安装 JDK8；
- 安装 Gradle 环境，参考[这里](#)获取安装信息；
- 安装 Android Studio，可以到[这里](#)下载；
- 安装 Android SDK. 打开 Android Studio，该 IDE 会提示需要安装 Android SDK。默认是安装最新的 SDK，你也可以在 Configure-SDK Manager 下安装其他 SDK 包；
- 配置环境变量，将 ANDROID_SDK_ROOT 设置为你安装了 SDK 的位置；将该位置下 tools/bin,platform-tools 和 emulator 加入 PATH 环境变量中；
- 如果有手机，请忽略。否则通过 AVD Manager 创建一个 Android 虚拟设备，作为模拟器调试使用。



移动环境下运行—Android

假定你的项目名称为 myApp，请先在终端切换到该目录下，然后运行以下命令：

```
ionic cordova prepare android
```

再通过以下命令在 Android 设备或者模拟器中运行：

```
ionic cordova run android -l
```



移动环境下调试—Android

有以下三种调试方法:

- 使用 Chrome DevTools。在浏览器中输入 `chrome://inspect`，点击对应设备的 App 即可。注意：需要 App 已经启动，且第一次需要翻墙；



移动环境下调试—Android

有以下三种调试方法:

- 使用 Chrome DevTools。在浏览器中输入 `chrome://inspect`，点击对应设备的 App 即可。**注意：需要 App 已经启动，且第一次需要翻墙；**
- 使用 Android Studio。打开 Android Studio，然后选择 View-Too Windows-Logcat；



移动环境下调式—Android

有以下三种调试方法:

- 使用 Chrome DevTools。在浏览器中输入 `chrome://inspect`，点击对应设备的 App 即可。**注意：需要 App 已经启动，且第一次需要翻墙；**
- 使用 Android Studio。打开 Android Studio，然后选择 View-Too Windows-Logcat；
- 使用命令。`adb logcat`



目录

1 Ionic 简介

2 第一个 Ionic 程序

3 Ionic UI 组件

4 Ionic Native 组件



ionic 布局

- 通过 *ion-grid*, *ion-col* 和 *ion-row* 进行网格布局;



ionic 布局

- 通过 *ion-grid*, *ion-col* 和 *ion-row* 进行网格布局;
- 默认每行有 12 列, 可以通过 *size* 调整列大小;



ionic 布局

- 通过 *ion-grid*, *ion-col* 和 *ion-row* 进行网格布局;
- 默认每行有 12 列, 可以通过 *size* 调整列大小;
- 行对齐方式包括: *align-items-start*, *align-items-end*, *align-items-center*;



ionic 布局

- 通过 *ion-grid*, *ion-col* 和 *ion-row* 进行网格布局;
- 默认每行有 12 列, 可以通过 *size* 调整列大小;
- 行对齐方式包括: *align-items-start*, *align-items-end*, *align-items-center*;
- 列对齐方式包括: *align-self-start*, *align-self-end*, *align-self-center*



UI 组件

Ionic 提供了丰富的 UI 组件，包括：

- 告警框 (Alert);



UI 组件

Ionic 提供了丰富的 UI 组件，包括：

- 告警框 (Alert);
- 行为条 (Action Sheet);



UI 组件

Ionic 提供了丰富的 UI 组件，包括：

- 告警框 (Alert);
- 行为条 (Action Sheet);
- 徽章 (Badge)



UI 组件

Ionic 提供了丰富的 UI 组件，包括：

- 告警框 (Alert);
- 行为条 (Action Sheet);
- 徽章 (Badge)
- 按钮 (Button)



UI 组件

Ionic 提供了丰富的 UI 组件，包括：

- 告警框 (Alert);
- 行为条 (Action Sheet);
- 徽章 (Badge)
- 按钮 (Button)
- 卡片 (Card)



UI 组件

Ionic 提供了丰富的 UI 组件，包括：

- 告警框 (Alert);
- 行为条 (Action Sheet);
- 徽章 (Badge)
- 按钮 (Button)
- 卡片 (Card)
- 其他。参考[这里](#)确定可用组件，以及其使用方式。



练习 1

使用 Ionic 的 UI 组件，在手机上实现智慧办公室环境监控数据的展示，包括但不限于：温湿度、投影仪/空调/灯光开关状态，门禁状态等。**注：该题为今年服务外包大赛竞赛题中的一部分。**



练习 2

使用 Ionic 的 UI 组件，在手机上实现智慧办公室用户登录界面及其控制。**注：该题为今年服务外包大赛竞赛题中的一部分，需要用到 Angular 知识。**



目录

- 1 Ionic 简介
- 2 第一个 Ionic 程序
- 3 Ionic UI 组件
- 4 Ionic Native 组件



Native 组件

Ionic 的 Native 组件提供了丰富的功能，用于访问手机的本地特性，比如：传感器数据、相机、蓝牙、支付等等。一些常见的 Native 组件包括：

- 蓝牙通信。包括 BLE、Bluetooth Serial 和 BluetoothLE 等；

参考[这里](#)获取完整的 Native 组件列表以及使用方法。



Native 组件

Ionic 的 Native 组件提供了丰富的功能，用于访问手机的本地特性，比如：传感器数据、相机、蓝牙、支付等等。一些常见的 Native 组件包括：

- 蓝牙通信。包括 BLE、Bluetooth Serial 和 BluetoothLE 等；
- 相机。Camera 和 Camera Preview；

参考[这里](#)获取完整的 Native 组件列表以及使用方法。



Native 组件

Ionic 的 Native 组件提供了丰富的功能，用于访问手机的本地特性，比如：传感器数据、相机、蓝牙、支付等等。一些常见的 Native 组件包括：

- 蓝牙通信。包括 BLE、Bluetooth Serial 和 BluetoothLE 等；
- 相机。Camera 和 Camera Preview；
- HTTP 通信。HTTP；

参考[这里](#)获取完整的 Native 组件列表以及使用方法。



Native 组件

Ionic 的 Native 组件提供了丰富的功能，用于访问手机的本地特性，比如：传感器数据、相机、蓝牙、支付等等。一些常见的 Native 组件包括：

- 蓝牙通信。包括 BLE、Bluetooth Serial 和 BluetoothLE 等；
- 相机。Camera 和 Camera Preview；
- HTTP 通信。HTTP；
- 本地通知。Local Notifications；

参考[这里](#)获取完整的 Native 组件列表以及使用方法。



Native 组件

Ionic 的 Native 组件提供了丰富的功能，用于访问手机的本地特性，比如：传感器数据、相机、蓝牙、支付等等。一些常见的 Native 组件包括：

- 蓝牙通信。包括 BLE、Bluetooth Serial 和 BluetoothLE 等；
- 相机。Camera 和 Camera Preview；
- HTTP 通信。HTTP；
- 本地通知。Local Notifications；
- 媒体相关。Media 和 Media Capture；

参考[这里](#)获取完整的 Native 组件列表以及使用方法。



Native 组件

Ionic 的 Native 组件提供了丰富的功能，用于访问手机的本地特性，比如：传感器数据、相机、蓝牙、支付等等。一些常见的 Native 组件包括：

- 蓝牙通信。包括 BLE、Bluetooth Serial 和 BluetoothLE 等；
- 相机。Camera 和 Camera Preview；
- HTTP 通信。HTTP；
- 本地通知。Local Notifications；
- 媒体相关。Media 和 Media Capture；
- 启动屏幕。Splash Screen

参考[这里](#)获取完整的 Native 组件列表以及使用方法。



练习 3

使用 Ionic 的蓝牙串口组件，实现 Android 手机与 Arduino HC05 的互联互通。控制 LED 灯的状态；



练习 4

使用 Ionic 的 HTTP 组件，实现 Android 手机与 Express Web 服务器的互联互通，从而实现一个完整的登录界面。

- 用户验证在服务器端完成



练习 4

使用 Ionic 的 HTTP 组件，实现 Android 手机与 Express Web 服务器的互联互通，从而实现一个完整的登录界面。

- 用户验证在服务器端完成
- 能提交数据和获取数据；

