

ionic 创建 APP

前面的章节中我们已经学会了 ionic 框架如何导入到项目中。

接下来我们将为大家介绍如何创建一个 ionic APP 应用。

ionic 创建 APP 使用 HTML、CSS 和 Javascript 来构建，所以我们可以创建一个 www 目录，并在目录下创建 index.html 文件，代码如下：

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Todo</title>
    <meta name="viewport" content="initial-scale=1, maximum-scale=1, user-scalable=no, width=device-width">

    <link href="lib/ionic/css/ionic.css" rel="stylesheet">

    <script src="lib/ionic/js/ionic.bundle.js"></script>

    <!-- 在使用 Cordova/PhoneGap 创建的 APP 中包含的文件，由 Cordova/PhoneGap 提供，(开发过程中显示 404) -->
    <script src="cordova.js"></script>
  </head>
  <body>
  </body>
</html>
```

以上代码中，我们引入了 Ionic CSS 文件、Ionic JS 文件及 Ionic AngularJS 扩展 ionic.bundle.js (ionic.bundle.js)。

ionic.bundle.js 文件已经包含了 Ionic 核心 JS、AngularJS、Ionic 的 AngularJS 扩展，如果你需要引入其他 Angular 模块，可以从 lib/js/angular 目录中调用。

cordova.js 是在使用 Cordova/PhoneGap 创建应用时生成的，不需要手动引入，你可以在 Cordova/PhoneGap 项目中找到该文件，所以在开发过程中显示 404 是正常的。

创建 APP

接下来我们来实现一个包含标题、侧边栏、列表等的应用,设计图如下：



创建侧边栏

侧边栏创建使用 `ion-side-menus` 控制器。

编辑我们先前创建的 `index.html` 文件，修改 `<body>` 内的内容，如下所示：

```
<body>
  <ion-side-menus>
    <ion-side-menu-content>
    </ion-side-menu-content>
    <ion-side-menu side="left">
    </ion-side-menu>
  </ion-side-menus>
</body>
```

控制器解析：

- **ion-side-menus**：是一个带有边栏菜单的容器，可以通过点击按钮或者滑动屏幕来展开菜单。
- **ion-side-menu-content**：展示主要内容的容器，可以通过滑动屏幕来展开menu。

- **ion-side-menu** : 存放侧边栏的容器。

初始化 APP

接下来我们创建一个新的 AngularJS 模块，并初始化，代码位于 `www/js/app.js` 中：

```
angular.module('todo', ['ionic'])
```

之后在我们的 `body` 标签中添加 `ng-app` 属性：

```
<body ng-app="todo">
```

在 `index.html` 文件的 `<script src="cordova.js"></script>` 上面引入 `app.js` 文件：

```
<script src="js/app.js"></script>
```

修改 `index.html` 文件 `body` 标签的内容，代码如下所示：

```
<body ng-app="todo">
  <ion-side-menus>

    <!-- 中心内容 -->
    <ion-side-menu-content>
      <ion-header-bar class="bar-dark">
        <h1 class="title">Todo</h1>
      </ion-header-bar>
      <ion-content>
      </ion-content>
    </ion-side-menu-content>

    <!-- 左侧菜单 -->
    <ion-side-menu side="left">
      <ion-header-bar class="bar-dark">
        <h1 class="title">Projects</h1>
      </ion-header-bar>
    </ion-side-menu>

  </ion-side-menus>
</body>
```

尝试一下 »

以上步骤我们已经完成了 ionic 基本 APP 的应用。

创建列表

首先创建一个控制器 **TodoCtrl** :

```
<body ng-app="todo" ng-controller="TodoCtrl">
```

在app.js文件中, 使用控制器定义列表数据 :

```
angular.module('todo', ['ionic'])

.controller('TodoCtrl', function($scope) {
  $scope.tasks = [
    { title: '菜鸟教程' },
    { title: 'www.runoob.com' },
    { title: '菜鸟教程' },
    { title: 'www.runoob.com' }
  ];
});
```

在index.html页面中数据列表我们使用 Angular ng-repeat 来迭代数据 :

```
<!-- 中心内容 -->
<ion-side-menu-content>
  <ion-header-bar class="bar-dark">
    <h1 class="title">Todo</h1>
  </ion-header-bar>
  <ion-content>
    <!-- 列表 -->
    <ion-list>
      <ion-item ng-repeat="task in tasks">
        {{task.title}}
      </ion-item>
    </ion-list>
  </ion-content>
</ion-side-menu-content>
```

修改后 index.html body 标签内代码如下 :

```
<body ng-app="todo" ng-controller="TodoCtrl">
  <ion-side-menus>

  <!-- 中心内容 -->
  <ion-side-menu-content>
    <ion-header-bar class="bar-dark">
      <h1 class="title">Todo</h1>
    </ion-header-bar>
    <ion-content>
```

```
<!-- 列表 -->
<ion-list>
  <ion-item ng-repeat="task in tasks">
    {{task.title}}
  </ion-item>
</ion-list>
</ion-content>
</ion-side-menu-content>

<!-- 左侧菜单 -->
<ion-side-menu side="left">
  <ion-header-bar class="bar-dark">
    <h1 class="title">Projects</h1>
  </ion-header-bar>
</ion-side-menu>

</ion-side-menus>
<script src="http://www.runoob.com/static/ionic/js/app.js"></script>
<script src="cordova.js"></script>
</body>
```

[尝试一下 »](#)

创建添加页面

修改 index.html 在 `</ion-side-menus>` 后添加如下代码:

```
<script id="new-task.html" type="text/ng-template">

  <div class="modal">

    <!-- Modal header bar -->
    <ion-header-bar class="bar-secondary">
      <h1 class="title">New Task</h1>
      <button class="button button-clear button-positive" ng-click="closeNewTask()">Cancel</button>
    </ion-header-bar>

    <!-- Modal content area -->
    <ion-content>

      <form ng-submit="createTask(task)">
        <div class="list">
          <label class="item item-input">
            <input type="text" placeholder="What do you need to do?" ng-model="task.title">
          </label>
        </div>
        <div class="padding">
```

```
        <button type="submit" class="button button-block button-positive">Create Task</button>
      </div>
    </form>

  </ion-content>

</div>

</script>
```

以上代码中我们通过 `<script id="new-task.html" type="text/ng-template">` 定义了新的模板页面。

表单提交时触发 `createTask(task)` 函数。

`ng-model="task.title"` 会将表单的输入数据赋值给 `task` 对象的 `title` 属性。

修改 `<ion-side-menu-content>` 内的内容，代码如下：

```
<!-- 中心内容 -->
<ion-side-menu-content>
<ion-header-bar class="bar-dark">
  <h1 class="title">Todo</h1>
  <!-- 新增按钮 -->
  <button class="button button-icon" ng-click="newTask()">
    <i class="icon ion-compose"></i>
  </button>
</ion-header-bar>
<ion-content>
  <!-- 列表 -->
  <ion-list>
    <ion-item ng-repeat="task in tasks">
      {{task.title}}
    </ion-item>
  </ion-list>
</ion-content>
</ion-side-menu-content>
```

app.js 文件中，控制器代码如下：

```
angular.module('todo', ['ionic'])

.controller('TodoCtrl', function($scope, $ionicModal) {
  $scope.tasks = [
    { title: '菜鸟教程' },
    { title: 'www.runoob.com' },
    { title: '菜鸟教程' },
    { title: 'www.runoob.com' }
  ];
});
```

```
// 创建并载入模型
$ionicModal.fromTemplateUrl('new-task.html', function(modal) {
  $scope.taskModal = modal;
}, {
  scope: $scope,
  animation: 'slide-in-up'
});

// 表单提交时调用
$scope.createTask = function(task) {
  $scope.tasks.push({
    title: task.title
  });
  $scope.taskModal.hide();
  task.title = "";
};

// 打开新增的模型
$scope.newTask = function() {
  $scope.taskModal.show();
};

// 关闭新增的模型
$scope.closeNewTask = function() {
  $scope.taskModal.hide();
};
});
```

创建侧边栏

通过以上步骤我们已经实现了新增功能，现在我们为 app 添加侧边栏功能。

修改 `<ion-side-menu-content>` 内的内容，代码如下：

```
<!-- 中心内容 -->
<ion-side-menu-content>
  <ion-header-bar class="bar-dark">
    <button class="button button-icon" ng-click="toggleProjects()">
      <i class="icon ion-navicon"></i>
    </button>
    <h1 class="title">{{activeProject.title}}</h1>
    <!-- 新增按钮 -->
    <button class="button button-icon" ng-click="newTask()">
      <i class="icon ion-compose"></i>
    </button>
  </ion-header-bar>
  <ion-content scroll="false">
    <ion-list>
      <ion-item ng-repeat="task in activeProject.tasks">
```

```

        {{task.title}}
      </ion-item>
    </ion-list>
  </ion-content>
</ion-side-menu-content>

```

添加侧边栏：

```

<!-- 左边栏 -->
<ion-side-menu side="left">
  <ion-header-bar class="bar-dark">
    <h1 class="title">Projects</h1>
    <button class="button button-icon ion-plus" ng-click="newProject()">
  </button>
</ion-header-bar>
<ion-content scroll="false">
  <ion-list>
    <ion-item ng-repeat="project in projects" ng-click="selectProject(project, $index)" ng-class="{active: activeProject == project}">
      {{project.title}}
    </ion-item>
  </ion-list>
</ion-content>
</ion-side-menu>

```

<ion-item> 中的 ng-class 指令设置菜单激活样式。

这里我们创建一个新的js 文件 app2.js(为了不与前面的混淆)，代码如下：

```

angular.module('todo', ['ionic'])
/**
 * The Projects factory handles saving and loading projects
 * from local storage, and also lets us save and load the
 * last active project index.
 */
.factory('Projects', function() {
  return {
    all: function() {
      var projectString = window.localStorage['projects'];
      if(projectString) {
        return angular.fromJson(projectString);
      }
      return [];
    },
    save: function(projects) {
      window.localStorage['projects'] = angular.toJson(projects);
    },

```



```
newProject: function(projectTitle) {
    // Add a new project
    return {
        title: projectTitle,
        tasks: []
    };
},
getLastActiveIndex: function() {
    return parseInt(window.localStorage['lastActiveProject']) || 0;
},
setLastActiveIndex: function(index) {
    window.localStorage['lastActiveProject'] = index;
}
}
}))

.controller('TodoCtrl', function($scope, $timeout, $ionicModal, Projects, $ionicSideMenuDelegate) {

    // A utility function for creating a new project
    // with the given projectTitle
    var createProject = function(projectTitle) {
        var newProject = Projects.newProject(projectTitle);
        $scope.projects.push(newProject);
        Projects.save($scope.projects);
        $scope.selectProject(newProject, $scope.projects.length-1);
    }

    // Load or initialize projects
    $scope.projects = Projects.all();

    // Grab the last active, or the first project
    $scope.activeProject = $scope.projects[Projects.getLastActiveIndex()];

    // Called to create a new project
    $scope.newProject = function() {
        var projectTitle = prompt('Project name');
        if(projectTitle) {
            createProject(projectTitle);
        }
    };

    // Called to select the given project
    $scope.selectProject = function(project, index) {
        $scope.activeProject = project;
        Projects.setLastActiveIndex(index);
        $ionicSideMenuDelegate.toggleLeft(false);
    };
});
```

```
// Create our modal
$ionicModal.fromTemplateUrl('new-task.html', function(modal) {
  $scope.taskModal = modal;
}, {
  scope: $scope
});

$scope.createTask = function(task) {
  if(!$scope.activeProject || !task) {
    return;
  }
  $scope.activeProject.tasks.push({
    title: task.title
  });
  $scope.taskModal.hide();

  // Inefficient, but save all the projects
  Projects.save($scope.projects);

  task.title = "";
};

$scope.newTask = function() {
  $scope.taskModal.show();
};

$scope.closeNewTask = function() {
  $scope.taskModal.hide();
}

$scope.toggleProjects = function() {
  $ionicSideMenuDelegate.toggleLeft();
};

// Try to create the first project, make sure to defer
// this by using $timeout so everything is initialized
// properly
$timeout(function() {
  if($scope.projects.length == 0) {
    while(true) {
      var projectTitle = prompt('Your first project title:');
      if(projectTitle) {
        createProject(projectTitle);
        break;
      }
    }
  }
});
```

```
});
```

body 中 ion-side-menus 代码如下：

```
<ion-side-menus>

<!-- 中心内容 -->
<ion-side-menu-content>
  <ion-header-bar class="bar-dark">
    <button class="button button-icon" ng-click="toggleProjects()">
      <i class="icon ion-navicon"></i>
    </button>
    <h1 class="title">{{activeProject.title}}</h1>
    <!-- 新增按钮 -->
    <button class="button button-icon" ng-click="newTask()">
      <i class="icon ion-compose"></i>
    </button>
  </ion-header-bar>
  <ion-content scroll="false">
    <ion-list>
      <ion-item ng-repeat="task in activeProject.tasks">
        {{task.title}}
      </ion-item>
    </ion-list>
  </ion-content>
</ion-side-menu-content>

<!-- 左边栏 -->
<ion-side-menu side="left">
  <ion-header-bar class="bar-dark">
    <h1 class="title">Projects</h1>
    <button class="button button-icon ion-plus" ng-click="newProject()">
    </button>
  </ion-header-bar>
  <ion-content scroll="false">
    <ion-list>
      <ion-item ng-repeat="project in projects" ng-click="selectProject(project, $index)" ng-class="{active: activeProject == project}">
        {{project.title}}
      </ion-item>
    </ion-list>
  </ion-content>
</ion-side-menu>

</ion-side-menus>
```

尝试一下 »

← ionic 安装

ionic 头部与底部 →

 点我分享笔记