Tutorial

AngularJS, Ionicjs, Cordova, Android, Firebase

Índice

[AngularJS](#angularjs)

[Ionicjs](#Ionicjs)

[Cordova](#Cordova)

[Android](#Android)

[Firebase](#Firebase)

* AngularJS

Este tutorial está especialmente diseñado para ayudarlo a aprender AngularJS de la manera más rápida y eficiente posible. Primero, aprenderá los conceptos básicos de AngularJS: directivas, expresiones, filtros, módulos y controladores. Entonces aprenderás todo lo demás que necesitas saber sobre AngularJS.

Lo que ya deberías saber Antes de estudiar AngularJS:

Debe tener una comprensión básica de: HTML, CSS, JavaScript.

* Ionicjs

Ionic actualmente requiere AngularJS para trabajar en todo su potencial. Si bien aún puede usar la parte de CSS del marco, se perderá poderosas interacciones de UI, gestos, animaciones y otras cosas.

* Descargue el último lanzamiento oficial de Ionic y comience
* Línea de comando de utilidad y proyecto de semilla

Tenemos una utilidad node.js que se puede usar para iniciar rápidamente un proyecto Ionic. Estamos trabajando para desarrollar la funcionalidad de esta herramienta, pero ahora puede crear rápidamente un proyecto con nuestro proyecto inicial al hacer lo siguiente:

$ npm install -g ionic

$ ionic start myproject --type ionic1

* Qué está incluido

Ionic viene con Javascript y CSS compilados para su aplicación, archivos Sass opcionales y extensiones de JS Framework

* CSS / Sass

El CSS puede funcionar por sí mismo, pero también está diseñado para ser mejorado por el desarrollador. Para simplificar, siempre puede agregar su propio CSS y anular las propiedades predeterminadas. Y para obtener aún más potencia y flexibilidad, el núcleo está escrito con Sass e incluye variables y mixins fácilmente personalizables. Si bien el diseño predeterminado es similar a IOS.

[Índice](#Indice)

* Cordova

Apache Cordova es un marco de desarrollo móvil de código abierto. Permite utilizar las tecnologías estándar web como HTML5, CSS3 y JavaScript para desarrollo multiplataforma, evitando el lenguaje de desarrollo nativo cada plataforma móvil. Aplicaciones ejecutan dentro de envolturas para cada plataforma y dependen de enlaces estándares API para acceder a de cada dispositivo sensores, datos y estado de la red.

Apache Cordova aplicaciones se basan en un común config.xml archivo que proporciona información acerca de la aplicación y especifica los parámetros que afectan a cómo funciona, como si responde a la orientación cambia de puesto. Este archivo se adhiere a la especificación de Empaquetado de la aplicación Web, widget, o de la W3C.

**Nota**: desde la versión 3.0, cuando se crea un proyecto de Córdoba no tiene presente algún plugin. Este es el nuevo comportamiento predeterminado. Algún plugin que desee, incluso los plugins del núcleo, debe agregarse explícitamente.

Vías de desarrollo

* **Flujo de trabajo multiplataforma (CLI):** Use este flujo de trabajo si quieres tu aplicación para ejecutar en los sistemas operativos móviles como sea posible, con poco necesidad específica de la plataforma desarrollo. Este flujo de trabajo se centra en la cordova.
* **Flujo de trabajo centrado en plataforma:** Utilice este flujo de trabajo si desea concentrarse en construir una aplicación para una sola plataforma y necesitan poder modificarlo en un nivel inferior. Tienes que utilizar este enfoque, por ejemplo, si quieres tu aplicación para mezclar los componentes nativos personalizados con componentes Cordova basados en web, como se explica en WebViews incrustación.

Instalación de Cordova

La instalación de Córdoba será diferente dependiendo del flujo de trabajo anterior que usted elige:

* Flujo de trabajo multiplataforma: ver la interfaz de línea de comandos.
* Flujo de trabajo centrado en plataforma: ver las guías de la plataforma.

Después de instalar Cordova, se recomienda que consulte a las guías de plataforma para las plataformas móviles que va a desarrollar para. También se recomienda que también revise la guía de privacidad, seguridad y próximos pasos. Para configurar Cordova, consulte el archivo config.xml. Para acceder a la función nativa en un dispositivo de JavaScript, se refieren a las APLS de Plugin. Y se refieren a las otras guías incluidas según sea necesario.

[Índice](#Indice)

* Android

Android es un sistema operativo de código abierto y basado en Linux para dispositivos móviles como teléfonos inteligentes y tabletas. Android fue desarrollado por Open Handset Alliance, liderado por Google y otras compañías. Este tutorial le enseñará la programación básica de Android y también le mostrará algunos conceptos avanzados relacionados con el desarrollo de aplicaciones de Android.

Audiencia

Este tutorial ha sido preparado para los principiantes para ayudarlos a comprender la programación básica de Android. Después de completar este tutorial, se encontrará en un nivel moderado de experiencia en programación de Android desde donde puede llevarlo a los siguientes niveles.

Requisitos previos

La programación de Android se basa en el lenguaje de programación Java, por lo que si tiene una comprensión básica de la programación Java, será divertido aprender el desarrollo de aplicaciones de Android.

* Firebase

Llaves compatibles

* Realtime

En lugar de las solicitudes HTTP típicas, Firebase Realtime Database utiliza la sincronización de datos: cada vez que los datos cambian, cualquier dispositivo conectado recibe esa actualización en milisegundos.

* Offline

Las aplicaciones de Firebase siguen siendo receptivas incluso cuando están desconectadas porque el SDK de Firebase Realtime Database mantiene sus datos en el disco. Una vez que se restablece la conectividad, el dispositivo cliente recibe los cambios que perdió, sincronizándolo con el estado actual del servidor.

* Accessible from Client Devices

Se puede acceder a Firebase Realtime Database directamente desde un dispositivo móvil o un navegador web; no hay necesidad de un servidor de aplicaciones. La seguridad y la validación de datos están disponibles a través de las Reglas de seguridad de Firebase.

* Scale across multiple databases

Con Firebase Realtime Database en el plan de precios Blaze, puede soportar las necesidades de datos de su aplicación a escala dividiendo sus datos en varias instancias de base de datos en el mismo proyecto de Firebase.

[Índice](#Indice)