PROGRAMACIÓN DE APLICACIONES MÓVILES

PGY4121

Profesor:

DuocUC®

ESCUELA DE INFORMÁTICA Y TELECOMUNICACIONES

Actividad N°3.2: Preparando Nuestra Aplicación

Preparar la aplicación

DuocUC®

Objetivos

Lo que se espera que aprendas en esta Actividad es:

- Preparar la configuración general de la aplicación para su posterior despliegue
- Conocer lo que es Ionic Config y cómo lo podemos usar
- Conocer las distintas configuraciones que podemos realizar a la configuración de la aplicación
- Conocer las distintas plataformas
- Conocer el proceso de firma de la aplicación

DuocUC®

Ionic Config

DuocUC®

PGY4121 Ionic Config

- Al generar el proyecto Ionic Angular se utiliza una configuración inicial el cual nosotros podemos sobreescribir modificando el **app.module.ts**

```
@NgModule({
  ...
  imports: [BrowserModule,
  IonicModule.forRoot(),
  AppRoutingModule
  ],
  providers: [...],
  bootstrap: [AppComponent]
})
export class AppModule {}
```

PGY4121 Ionic Config

- La configuración presentada se deshabilita el efecto ripple en toda la aplicación y se fuerza al modo Material Desing

```
...
IonicModule.forRoot({
      rippleEffect: false,
      mode: 'md'
    })
...
```

PGY4121 Ionic Config - Modes

- Ionic utiliza modos para personalizar el aspecto de los componentes, cada plataforma tiene un modo predeterminado anulable mediante la configuración.

Plataforma	Modo	Descripción
ios	ios	Para iPhone, iPad o iPod utilizara los estilos de iOS
android	md	Para los dispositivos Android utiliza los estilos de Material Desing
core	md	Cualquier otra plataforma que no se ajuste a ninguna plataforma anterior utiliza los estilos de Material Desing

PGY4121 Ionic Config - Modes

- El resultado de esa configuración se traduce en:

Plataforma	Modo	Descripción
ios	ios	<html class="ios">
android	md	<html class="md">
core	md	<html class="md">

El elemento se le modificara la clase (class) al modo indicado

DuocUC®

La Configuración

PGY4121 La configuración

- Se puede establecer una configuración determinada en función a la plataforma en el cual está operando la aplicación de la siguiente forma

```
import { isPlatform, IonicModule } from '@ionic/angular';

const getConfig = () => {
  if (isPlatform('hybrid')) {
    return {
      backButtonText: 'Previous',
      tabButtonLayout: 'label-hide'
    }
  }

  return {
    menuIcon: 'ellipsis-vertical'
  }
}
@NgModule({
  ...
  imports: [
    BrowserModule,
    IonicModule.forRoot(getConfig()),
    AppRoutingModule
  ],
  ...
})
```

DuocUC®

PGY4121 La configuración

- Se puede observar que se declara una constante identificada como getConfig que será una función, función que retorna una configuración específica según la plataforma mediante la función suministrada por '@ionic/angular' llamada isPlatform

```
import { isPlatform, IonicModule } from '@ionic/angular';

const getConfig = () => {
  if (isPlatform('hybrid')) {
    return {
      backButtonText: 'Previous',
      tabButtonLayout: 'label-hide'
    }
  }

  return {
    menuIcon: 'ellipsis-vertical'
  }
}
@NgModule({
  ...
  imports: [
    BrowserModule,
    IonicModule.forRoot(getConfig()),
    AppRoutingModule
  ],
  ...
})
```

DuocUC®

PGY4121 La configuración

- Las plataformas pueden ser (estas son solo algunas)

Plataforma	Descripción
ios	Un dispositivo con Ios
android	Un dispositivo con Android
Cordova	Un dispositivo que ejecuta Cordova
Desktop	Un dispositivo de escritorio
mobileweb	Un navegador web que se ejecuta en un dispositivo móvil
Capacitor	Un dispositivo que ejecuta Capacitor

https://ionicframework.com/docs/angular/platform#platforms

Opciones de Configuración

PGY4121 Opciones de Configuración

» Las opciones de pueden implementar de la siguiente forma

```
...
IonicModule.forRoot({
      rippleEffect: false,
      mode: 'md'
    })
...
```

```
import { isPlatform, IonicModule } from '@ionic/angular';

const getConfig = () => {
  if (isPlatform('hybrid')) {
    return {
      backButtonText: 'Previous',
      tabButtonLayout: 'label-hide'
    }
  }

  return {
    menuIcon: 'ellipsis-vertical'
  }
}
@NgModule({
  ...
  imports: [
    BrowserModule,
    IonicModule.forRoot(getConfig()),
    AppRoutingModule
  ],
  ...
})
```

DuocUC®

PGY4121 Opciones de Configuración

» Algunas opciones son

Configuración	Tipo	Descripción
animated	boolean	Si es true Ionic habilita todas las animaciones y transiciones
backButtonIcon	string	Anula el icono predeterminado en los componentes <ion-back-button>
hardwareBackButton	boolean	Si es true, Ionic responderá al botón de retroceso de hardware del dispositivo

Lista completa en: https://ionicframework.com/docs/angular/config#config-options

DuocUC®

Deployment

PGY4121 Deployment

- Luego de preparar nuestra aplicación, veremos como generar la APK de la aplicación, para eso se les suministrará el documento 3.2.2 Generate Android APK. Pero no te adelantes, primero revisemos el proceso de firma.

Firmando la aplicación

PGY4121 APK sin firmar

- ¿Por qué debemos firmar nuestra aplicación?
  - Debemos firmar nuestra aplicación para identificarnos, indicamos quienes somos, de donde somos entre otras cosas. Es una norma que tiene tanto Google para las publicaciones en Google Play como Apple lo tiene para su App Store.
- ¿Podemos usar una apk sin firmar?
  - Si podemos usar una apk sin firmar, pero lo tenemos que instalar manualmente mediante el apk (debug o prod sin firmar).

Tipos de Firma

PGY4121 Firma Manual

- Uno puede generar una APK (sin firma), y como no se encuentra firmada se puede ocupar, pero no es confiable ni tampoco puede ser subido a los Stores.

- Para esto necesitamos generar un keystore mediante la herramienta keytool de Java

- Luego de tener una llave se firma la aplicación con la llave generada o ya guardada sobre la aplicación

- La aplicación también se puede firmar directamente en Android Studio generando la keystore desde ahí

DuocUC®

PGY4121 Firma Automática

- Para la firma automática se usa Android App Bundle, que es un nuevo formato de carga que incluye todo el código compilado y los recursos de la app, pero delega la generación a Google Play

Android App Bundle

DuocUC®

PGY4121 Android App Bundle

- El nuevo modelo de publicación de aplicaciones de Google Play, denominado Dynamic Delivery, luego utiliza su paquete de aplicaciones para generar y publicar una APK optimizada para la configuración de cada dispositivo de cada usuario por lo que descargan solo el código y recursos que necesita para su aplicación.

Capacitor asume que se hará de esta nueva forma

DuocUC®

PGY4121 Funciones

FUNCIÓN

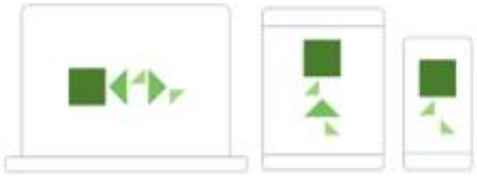
Administración eficaz de actualizaciones

Crea un artefacto que incluya todo el código compilado, los recursos y las bibliotecas nativas de tu app. Ya no tendrás que crear, firmar, cargar y administrar códigos de versión para varios APK.

FUNCIÓN

Obtén los beneficios de una app más liviana

Dynamic Delivery, el modelo de publicación de Google Play, usa tu Android App Bundle a fin de compilar y publicar APK optimizados para cada configuración de dispositivo. De esta manera, los usuarios obtienen una app de menor tamaño sin el código que no necesitan ni los recursos que se usan para otros dispositivos. Controla el tamaño de tu app en el nuevo informe de tamaño de la app en Google Play Console.



FUNCIÓN

Compilación más rápida

Los sistemas de compilación, como Android Studio con Gradle, están optimizados para apps modulares, por lo que permiten una compilación mucho más rápida que las apps monolíticas grandes. Pasarás menos horas esperando y más tiempo diseñando, codificando y probando tu app.

Más información

Más información

https://developer.android.com/platform/technology/app-bundle

PGY4121 Funciones

Menor tamaño gracias a Dynamic Delivery

22% Airbnb

23% Blue Apron

8% Clash of Clans

33% Dropbox

42% Duolingo

35% Fandango

26% Flipkart

28% Jaumo

PGY4121 Funciones

36%

LinkedIn

10%

Minion Rush

57%

Netflix

25%

Ola Cabs

39%

Perigee

20%

redBus

13%

Runtastic

49%

RV AppStudios

13%

Riafy

6%

Shazam

28%

Tinder

18%

TripAdvisor

DuocUC®

Resumen

PGY4121 Resumen

- Revisamos cómo preparar la configuración general de la aplicación mediante el app.module.ts
- Revisamos lo que es Ionic Config y cómo lo podemos usar
- Conocer las distintas configuraciones que podemos realizar a la configuración de la aplicación
- Conocer las distintas plataformas
- Revisamos el por qué firmamos y sus diferentes formas de hacerlo

DuocUC®

Profundiza más!!!

Consultar la documentación oficial:

Plaform:

https://ionicframework.com/docs/angular/platform

Config:

https://ionicframework.com/docs/angular/config