

Familia Profesional <b>Informática y Telecomunicaciones</b>		Nombre del Ciclo Formativo <b>Título de Técnico Superior en Desarrollo de Aplicaciones Web</b>		
Centro Educativo <b>IES Campanillas (sede PTA)</b>		Módulo Profesional <b>Aplic. Híbridas con Ionic (ofic. Acceso a datos)</b> <b>Código: 0486 N.º de créditos ECTS: 9</b>	Profesor <b>Luis José Sánchez</b>	
Curso lectivo <b>2021 / 2022</b>	Grupo <b>2º DAM</b>	Tipo de documento <b>Requisitos de la aplicación en Ionic</b>	Trimestre <b>Primero</b>	Fecha de exposición <b>15 de diciembre de 2020</b>

## DESARROLLO DE APLICACIÓN HÍBRIDA CON IONIC

Se debe desarrollar, entregar y presentar en clase una aplicación híbrida hecha en **Ionic** que cumpla los requisitos que se especifican en el siguiente apartado. La aplicación debe ser totalmente funcional y debe estar lo suficientemente probada como para no contener errores. Es preferible entregar una aplicación sencilla que funcione perfectamente que una aplicación con muchas funcionalidades que contenga errores. La temática de la aplicación es totalmente libre y puede coincidir con la de otra asignatura.

## REQUISITOS MÍNIMOS

A continuación se especifican los requisitos mínimos que debe incluir la aplicación:

- Páginas y/o componentes que aíslen correctamente las diferentes funcionalidades.
- Servicio/s para grabar y proveer datos a la aplicación.
- Inyección de dependencias.
- Estructuras ngIf y ngFor.
- Clase/s y/o interfaz/interfaces.
- Estilos propios.
- Almacenamiento de datos de manera local (no quita que se puedan utilizar datos en la nube).
- Ventana/s emergente/es.

De forma opcional, la aplicación puede incluir:

- Promesa/s y/o observable/s.
- Consumo de una API desarrollada en NestJs.
- Consumo de APIs de terceros.
- Tuberías (pipes) ya existentes o propias.
- Uso de la cámara de fotos.
- Otras funcionalidades que no se hayan visto en clase.

## **GITHUB**

El código fuente de la aplicación se debe subir a **GitHub**. El fichero **README.md** debe contener el título, una descripción detallada y capturas de pantalla de todas las páginas y ventanas de la aplicación. Es recomendable incluir también GIFs animados mostrando el funcionamiento de la aplicación. El repositorio de la aplicación se debe enlazar desde el repositorio índice que se indicará en clase a tal efecto.

## **VÍDEO EN YOUTUBE (opcional)**

En caso de tener vídeo, deberá estar enlazado tanto desde el repositorio de la aplicación como desde el índice de trabajos. En el vídeo se debe mostrar la aplicación funcionando y se deben explicar sus funcionalidades y las peculiaridades del código; con unos 3 o 4 minutos es suficiente; la duración máxima es de 6 minutos. El vídeo debe estar alojado en Youtube.

## **EXPOSICIÓN**

Cada alumno deberá exponer su aplicación y deberá mostrar su funcionamiento. Así mismo, describirá los pasos que ha seguido en su desarrollo. El profesor puede hacer preguntas durante o al final de la exposición. El tiempo de exposición aproximado estará entre los 5 y 10 minutos.

## **FECHAS**

- Fecha límite de entrega de la aplicación (subida a GitHub): **14 de diciembre a las 23:59h.**
- Exposiciones por orden de lista: el día **15 de diciembre en las horas de clase.**

### **Miércoles 15 de diciembre**

#### **1ª hora**

Alvarado Díaz, Irene  
Brea Rivera, Miguel Ángel  
Buendía López, Jairo  
Cisneros Santana, Pedro  
De Abrisqueta Rangel, Ander  
Estrada Cobo, José Manuel  
Fabián El Belaizi, Daniel

#### **2ª hora**

Galera García, José Ángel  
García Gómez, Sergio  
Gutiérrez Ramírez, Jesús María  
López Lozano, Santos  
López Pérez, María de la Cruz  
Millán Rebollo, Noel  
Millón Cortés, Manuel Alejandro

#### **3ª hora**

Morales Aragón, Marcos  
Reyes Podadera, Pablo Jesús  
Rodríguez-Barbero Verdura, David  
Salmerón Robles, Gonzalo José  
Sevilla Valderrama, Fernando Iván  
Egea Hermoso, Adrián

## **EVALUACIÓN**

Para tener aprobado el trabajo, la aplicación debe estar subida a GitHub en el plazo indicado y cumplir con los requisitos mínimos. A partir de ahí, para la nota se tendrá en cuenta la documentación, la complejidad de la programación, las funcionalidades no obligatorias que se hayan incluido, el vídeo, etc. Cada parte no tiene una nota concreta sino que la aplicación se valora como un todo indivisible.