

## Ejercicio 1. Programación (6p)

Se desea implementar una aplicación web SPA para consultar la información de los alumnos de la universidad con la siguiente funcionalidad:

- Al entrar en la aplicación se mostrará un título “Alumnos URJC” y un listado con el nombre de cada alumno en un enlace.
- Cuando el usuario pinche en el nombre de un alumno, la aplicación navegará a una nueva página y mostrará una ficha con los datos de ese usuario (Nombre, Dirección y correo electrónico). El título deberá seguir estando visible.
- Habrá un botón “volver” que al ser pulsado volverá a la página principal.
- Se asume que la base de datos a la que se conecta la aplicación ya tiene información de alumnos. La aplicación no tiene la capacidad de dar de alta alumnos ni modificarlos.

Se pide:

- **A) Implementar el backend de la aplicación web con Spring MVC, SpringData y JPA. (2p)**
  - No es necesario escribir el fichero pom.xml, se puede asumir que tiene todas las dependencias correspondientes y está conectada a una base de datos externa correctamente configurada.
  - No es necesario escribir la clase Application.
  - Es necesario escribir TODOS los demás ficheros de la aplicación.
  - No es necesario incluir los imports en los ficheros Java.
  - No es necesario implementar los getter y setter. Se pueden dejar indicados o se puede usar Lombok.
  - Se valorará especialmente que la API REST esté diseñada de forma correcta (formato de las URLs, uso de los códigos de estado, etc.)
- **B) Implementar el frontend usando Angular (2p)**
  - Se puede asumir que se dispone de un proyecto Angular con todos los ficheros necesarios correctamente configurados (index.html, angular.cli, package.json, tsconfig.json, app.module.ts, etc.)
  - Hay que implementar todos los componentes necesarios y un servicio para las peticiones al backend. Hay que implementar el fichero para configurar las rutas de los componentes.
  - Se usará HTML plano en el template de los componentes.
  - No es necesario incluir los imports en los ficheros TypeScript.
  - Se asumirá que el proxy está correctamente configurado y se pueden usar URLs relativas para acceder a la API REST del backend.
- **C) Empaquetar la aplicación en un contenedor Docker (2p)**
  - Escribe un script (en Bash, Bat o PowerShell) que al ejecutarse construya la imagen daw/alumnos:1.0.0 con la aplicación (tanto backend como frontend) empaquetada. Se deberán implementar los ficheros Dockerfile que sean necesarios.
  - Se asumirá que el código Spring está alojado en una carpeta “backend” y el código Angular está alojado en una carpeta “frontend” y el script y el/los Dockerfile están en la carpeta raíz.
  - Se asumirá que en el sistema donde se ejecute el script o se construya el Dockerfile sólo tiene Docker instalado. No dispone de Node.js ni Maven.
  - Se asumirá que en DockerHub existen las imágenes maven:3.6.0, node:14.0.0 y openjdk-jre:11.0.0