



APLICACIONES OPEN SOURCE (SI729)
PRÁCTICA CALIFICADA 1
2024-1

Sección: WS53
Profesor: Flores Moroco, Juan Antonio
Duración: 100 minutos
Indicaciones:

1. La práctica calificada consta de 1 pregunta, y tendrá 100 minutos para resolverlas.
 2. La pregunta es de tipo Proyecto de Software y la entrega de su respuesta es a través de envío de archivo empaquetado **.zip** con proyecto de software en la Actividad PC1 (Práctica Calificada 1).
-

Enunciado:

Pregunta 1 (20 p.).

Caso The Star Wars API

Su cliente, The Star Wars API desean promover el uso de su plataforma (<https://the.star.wars.api/>), un RESTful API de alcance mundial, fácil de usar para desarrolladores que brinda acceso a datos públicos.

Para ello, le solicita elaborar una aplicación web que obtenga la información de Actores de la película The Star War, accediendo al endpoint:

<https://swapi.dev/api/people/?page=1&format=json>

Para el desarrollo web de lado web frontend, se ha seleccionado TypeScript como lenguaje de programación y Angular como Frontend Framework.

Se le encarga el desarrollo de una aplicación web que implemente las siguientes características:

- La aplicación debe presentar en la parte superior un Toolbar, con el título “The Star War People”
- La aplicación debe presentar dentro de la vista principal, una vista con título “Actors”, conteniendo un conjunto de Cards o Table que presenten la información de cada uno de los actores disponibles, incluyendo toda la información. El API ofrece arrays de información, pudiendo mostrarse todos los datos o solo el primero elemento.
- La aplicación debe presentar en la parte inferior, una sección de footer con la información “Copyright © 2024 UPC-DAOS. All rights reserved.” en la primera línea. Además debe incluir en la segunda línea del footer la información “Developed by” y los datos del Developer (considérese a usted como autor, indicando código, nombre y apellido.)

El equipo de IT de su cliente tomará en cuenta no sólo el cumplimiento de las características funcionales, sino el diseño de interfaz de usuario, así como la estructura del proyecto, aplicación de convenciones de nomenclatura de objetos de programación en inglés, convenciones de nomenclatura de Angular, organización y eficiencia del código. Igualmente se tomará en cuenta la aplicación de patrones de diseño. Se toma en cuenta accessibility (imágenes con texto alternativo, uso de ARIA attributes) y usability.

Restricciones técnicas: El equipo requiere que la interfaz de usuario esté basada en **Angular Material**, mientras que para la comunicación con el backend debe apoyarse en **HttpClient** (incluido en @angular/common/http). Debe incluir ARIA attributes en las vistas. La interfaz de usuario debe mostrar los textos en **inglés**. El proyecto de aplicación debe poder abrirse sin problemas en JetBrains WebStorm o IntelliJ IDEA.

La organización del proyecto debe ser domain-driven aplicando layered architecture, considerando los sub-dominios **public** para elementos genéricos y **actors** para componentes o elementos relacionados con los people. Aplique buenas prácticas para nomenclatura lógica y física de clases y componentes. Comente los archivos de código fuente elaborados por usted, incluyendo *summary* con el propósito y *author* con su nombre y apellido.

Incluya en el archivo README.md, la información de la aplicación, descripción y su información como autor.

Fuera del alcance:

Sidebar

Internationalization

Routing

Referencias:

<https://angular.io/guide/http>

<https://material.angular.io/components/toolbar/overview>

<https://material.angular.io/components/toolbar/overview#accessibility>

<https://material.angular.io/components/grid-list/overview>

<https://material.angular.io/components/grid-list/overview#accessibility>

<https://material.angular.io/components/card/overview>

<https://material.angular.io/components/card/overview#accessibility>

<https://www.w3.org/TR/wai-aria/#usage>

Criterio de Calificación	Sobresaliente (S)	Esperado (E)	Necesita Mejorar (M)	Insuficiente (I)	Calificación
C01. Building y ejecución	Al abrir el proyecto y ordenar la ejecución, ésta se inicia sin problemas. La aplicación es accesible en la ruta indicada.	La aplicación no llega a iniciar y ejecutarse, sin embargo el proceso de building llega a concluir.	Al cargar el proyecto el proceso de building presenta errores y no llega a concluir.	No elabora solución	
	2.0 puntos	1.0 punto	0.5 puntos	0 puntos	
C02. User Interface	Se evidencia que la interfaz de usuario aplica Responsive Web Design. La interfaz cumple con las características solicitadas para la estructura, elementos de la interfaz de usuario.	La interfaz de usuario no evidencia con claridad que aplica Responsive Web Design, o no cumple con todas las características solicitadas.	La aplicación no implementa una interfaz de usuario con las características solicitadas.	La aplicación no se visualiza o no implementa ninguna de las características para interfaz de usuario.	
	3.0 puntos	2.0 puntos	1.0 puntos	0 puntos	
C03. Features	La vista cumple de forma completa con los requisitos de interfaz de usuario, con todas las características y restricciones indicadas en el enunciado. Se cumple que la información es recuperada desde el endpoint solicitado.	La vista cumple de forma parcial con los requisitos de interfaz de usuario, o con la mayoría de las características y restricciones indicadas en el enunciado. Se cumple que la información es recuperada desde el endpoint solicitado.	La vista cumple de forma parcial con los requisitos de interfaz de usuario, con solo algunas de características y restricciones indicadas en el enunciado, o se cumple que la información es recuperada desde el endpoint solicitado, sin embargo, esta no se visualiza.	No se implementa el feature.	
C04. Code Organization	7.0 puntos El desarrollador organiza el código y los elementos de frontend de la solución, aplicando buenas prácticas de TypeScript y Angular, agrupando los elementos de la solución según convenciones, manteniendo organización de paquetes y carpetas recomendadas por el fabricante y buenas prácticas de la industria de software, bajo un enfoque domain-driven, cumpliendo con las restricciones técnicas sobre organización.	4.5 puntos El desarrollador aplica parcialmente en frontend convenciones, recomendaciones y buenas prácticas de TypeScript o Angular, o no cumple con todas las restricciones técnicas sobre organización.	2 puntos El desarrollador aplica en frontend solo algunas de las convenciones, recomendaciones y buenas prácticas de TypeScript o Angular, o no cumple con las restricciones técnicas sobre organización.	0 puntos No se evidencia un criterio de organización para los elementos de la solución, limitándose a la estructura por defecto.	
	3.0 punto	2.0 puntos	1.0 puntos	0 puntos	
C05. Code Quality	Utiliza para el frontend el lenguaje de programación TypeScript, el framework Angular y cumple con las restricciones técnicas indicadas. La codificación tiene un estilo claro, indentando los bloques de código según los estándares de programación correspondientes al lenguaje, aplicando una lógica consistente en los métodos, condicionales sin escenarios no contemplados, uso adecuado de reutilización de código para evitar redundancia. Aplica patrones de arquitectura y patrones de diseño. Distribuye el código en los niveles correspondientes, asignando lógica de persistencia, lógica de negocio, lógica de control, y transferencia a las interfaces y clases que corresponden.	Utiliza para el frontend el lenguaje de programación TypeScript y el framework Angular. La codificación es funcional, pero cumple de forma parcial con las restricciones técnicas, o sólo aplica parcialmente los estándares de indentación de bloques de código, o existen ineficiencias en la codificación: redundancia o inconsistencias en la lógica de programación. Aplica parcialmente patrones de arquitectura y patrones de diseño, o existe en algunas partes una distribución de la lógica en los niveles incorrectos.	Cumple con solo algunas restricciones técnicas, o no se evidencia aplicación de estándares o criterios de eficiencia en la codificación, con ausencia de comentarios, o no aplica patrones de arquitectura ni patrones de diseño, aunque la codificación es funcional.	No utiliza el lenguaje de programación TypeScript o no utiliza Angular, o no cumple con la mayoría de restricciones técnicas o el código no es funcional.	
	3.0 puntos	2.0 punto	1.0 puntos	0 puntos	
C06. Naming Standards	El desarrollador aplica en todos los nombres de objetos de programación como paquetes, componentes, interfaces, clases, objetos, variables, constantes y métodos la nomenclatura en inglés y la nomenclatura estándar para identificadores de clases, objetos, miembros de programación, así como los recursos.	El desarrollador aplica en la mayoría de casos la nomenclatura estándar para identificadores de clases, objetos, miembros de programación, así como los recursos.	El desarrollador aplica sólo en algunos casos la nomenclatura en inglés y la nomenclatura estándar para identificadores de clases, objetos, miembros de programación, así como los recursos.	El desarrollador no aplica nomenclatura en inglés para los objetos de programación ó recursos.	
	2.0 puntos	1.0 punto	0.5 puntos	0 puntos	
Total	20 puntos	12.5 puntos	5.75 puntos	0 puntos	

Lima, 16 de Abril del 2024