



SI730 - APLICACIONES WEB
PRÁCTICA CALIFICADA 1
2024-1

Sección:

Profesor:

Duración: 100 minutos

Indicaciones:

1. El examen 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 Jelly Bean, inc.

Usted forma parte de una nueva compañía que ha adquirido la franquicia Jelly Bean. Jelly Bean es una empresa que se encarga de ofrecer bebidas en base a Te con sabores exóticos y con muchos toppings. La franquicia como parte de su procedimiento y procesos está exigiendo que se cumplan una serie de requisitos no solo a nivel de implementación de locales, procedimientos y recetas, sino que también tiene requisitos a nivel de T.I.

Como parte de los requisitos de T.I. tenemos que desarrollar una e-commerce lo más pronto posible ya que estamos a punto de tener un incumplimiento. Para tal efecto y como parte de la franquicia, se nos está proporcionando una API que pueda proporcionar el detalle de los productos que se pueden ofrecer, así como las características de los mismos.

Podemos obtener estos datos accediendo al endpoint:

<https://jellybellywikiapi.onrender.com/api/Beans>

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

Se ha encargado a nuestro equipo de desarrollo la creación de una aplicación web que tenga las siguientes características:

- La aplicación debe presentar en la vista home un Toolbar en la parte superior, con el título “Jelly Bean Peru Store”.
- La aplicación debe presentar debajo una vista “Store Products” con cards para las los products. ■ Para cada card, la información del product a mostrar incluye: flavorName, ingredients, glutenFree, sugarFree, seasonal, kosher y image. El url de la imagen está en la propiedad imageUrl del objeto). La información groupName, debe estar ubicada en el footer del card.
- La aplicación debe presentar en la parte inferior de la vista “Store Products”, una sección de footer con la información “Copyright © 2024 Jelly Bean Peru Store, inc. 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.)

Dentro del equipo de desarrollo, se tiene a una persona de Quality Assurance que tomará en cuenta no solo 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 Vue.js, organización y eficiencia del código. Igualmente se tomará en cuenta la aplicación de patrones de diseño.

Restricciones técnicas: Debe generar el proyecto con Vite y utilizar Vue 3. El equipo requiere que la interfaz de usuario esté basada en Material Design utilizando la biblioteca de componentes de UI PrimeVue, mientras que para la comunicación con el backend debe apoyarse en axios

Página 2 de 4

(<https://github.com/axios/axios>). Debe incluir ARIA attributes en las vistas. La interfaz de usuario debe mostrar los textos en inglés. La organización del proyecto debe ser domain-driven y con layered architecture, considerando los sub-dominios public (para elementos o componentes generales de la interfaz de usuario, como home) y commerce (para componentes o elementos relacionados con el e

commerce como products). Aplique buenas prácticas y convenciones aplicadas en clase para nomenclatura lógica y física de clases y componentes. El proyecto de aplicación debe poder abrirse sin problemas en JetBrains WebStorm.

Para la nomenclatura física de componentes, clases e interfaces utilice kebab-case. Aplique en los nombres físicos sufijos que indiquen el tipo de elemento: .component.vue, .service.js, .entity.js. El sufijo component no debe ser parte del nombre del componente cuando se use en templates (Por ejemplo user

list.component.vue debería utilizarse en código de template como <user-list></user-list>). Para la nomenclatura de componentes de PrimeVue dentro de los templates utilice kebab-case.

Asigne el prefijo pv- al momento de incorporar su uso en el proyecto (Por ejemplo, pv-card).

Comente los archivos de código fuente elaborados por usted, incluyendo summary con el propósito y author con su nombre y apellido.

Fuera del alcance:

Sidebar

Internationalization

Routing

Referencias:

<https://fakestoreapi.com/>

<https://router.vuejs.org/>

<https://router.vuejs.org/installation.html>

<https://primevue.org/installation>

<https://primevue.org/button>

<https://primevue.org/toolbar>

<https://primevue.org/toast>

<https://primeflex.org/installation>

<https://primeflex.org/gridsystem>

<https://primeflex.org/flex>

<https://axios-http.com/docs/intro>

Página 3 de 4

Rúbrica de calificación

Sobresaliente (S) Esperado (E) Necesita Mejorar (M) Insuficiente (I) Calificación

Criterio de Calificación

C01. Building y ejecución	ejecución, ésta se inicia sin problemas. La aplicación es accesible en la ruta indicada.	La aplicación no llega a iniciar y concluir. ejecutarse, sin embargo el proceso de building llega a	no llega a concluir. Al cargar el proyecto el proceso de building presenta errores y
---------------------------	--	---	--

	2.0 puntos	1.0 punto	0.5 puntos	0 puntos
C02. User Interface	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 presenta la información solicitada.	La aplicación no presenta la información solicitada.

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 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 El desarrollador organiza el código y los elementos de frontend de la solución, aplicando buenas prácticas de JavaScript y Vue, 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 <u>organización</u> .	El desarrollador aplica en frontend un enfoque domain-driven, la mayoría de convenciones, recomendaciones y buenas prácticas de JavaScript o Vue, o no cumple con todas las restricciones técnicas sobre organización.	El desarrollador aplica en frontend solo algunas de las convenciones, recomendaciones y buenas prácticas de JavaScript defecto. No se evidencia un criterio de	organización para los elementos de la solución, limitándose a la estructura por
--	---	--	--	---

C05. Code Quality Utiliza para el frontend el lenguaje de programación JavaScript, el framework Vue 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	<u>corresponden</u> . Utiliza para el frontend el lenguaje de programación JavaScript y el framework Vue. 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, ó existen ineficiencias en la codificación:	redundancia ó 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 ó criterios de	eficiencia en la codificación, con ausencia de comentarios, ó no aplica patrones de arquitectura ni patrones de diseño, aunque la codificación es funcional. No utiliza el lenguaje de programación JavaScript o no utiliza Vue, o no cumple con la mayoría de restricciones técnicas o el código no es funcional.
--	---	---	---	--

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 <u>como los recursos</u> . El desarrollador aplica en la	mayoría de casos la nomenclatura en inglés y 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	6.00 puntos		

Lima, 16 de Abril del 2024