

Informe - Final

Para este trabajo se implementaron 1 único html con una SPA que contiene la Home de Natural Sushi y un panel de administración de productos. Queda fuera del alcance de este trabajo la implementación de un sistema login. Podría ser un trabajo futuro. Los productos que se cargan en el panel de control se ven reflejados en la home.

Se utilizó Vue.js, Vue router y Vuetify. Se tomó como base el TP2.

The screenshot shows the main landing page of the Natural Sushi website. At the top, there's a banner with a photo of the restaurant interior and the text "Bienvenido a Natural Sushi". Below the banner, there's a section titled "Nuestros productos" (Our products) with five images of different sushi rolls: California, Japan, Hot Cali, Vegetarian Roll, and Sushi Especial de la casa. Under "Nuestros servicios" (Our services), there are three cards: "Te lo llevamos a tu casa" (Delivery), "Take away", and "Un buen plato en un buen restaurante". At the bottom, there's a section titled "Reseñas" (Reviews) with four user reviews and a button labeled "DEJANOS TU RESERVA" (Leave us a reservation).

Home

Natural es una página de venta de sushis y comidas japonesas. La Home es la primera de las cuatro vistas definidas en la aplicación. Se ha utilizado un v-if para mostrar la sección de productos en el caso de que haya productos cargados.

Un v-for para mostrar cada una de las tarjetas. Se utilizó una Api Php y se consumen dos servicios.

El primer servicio es para obtener los productos.

El otro servicio es para obtener las reseñas que se muestran en la ultima sección de la Home.

El botón "Dejanos tu reseña" lleva a otra vista para dejar un comentario y puntuación.

Panel de administración de productos

La vista Admin contiene el panel de administración de productos permite ver los productos existentes, cargar nuevos, editar y eliminar un producto.

A esta página se podrá acceder a través del menú de la aplicación implementado con el componente v-navigation-drawer de Vuetify.

Se utilizaron directivas v-if, v-for, v-bind, filtros de Vue y componentes.

Los componentes utilizados son 3: product, edit, sushiform.

≡ Natural

Lista de productos

Codigo	Imagen	Nombre	Descripcion	Categoría	Precio	Acciones
123		California	Kanikama-palta	Clásicos	\$ 380	 
234		Tuna Cream	Atún cocido-mayonesa	Especiales	\$ 380	 
142		Miami	Salmón Cocido-Philadelphia y remolacha	Hot Rolls	\$ 760	 
667		San Pablo	Pollo Apanado-Philadelphia-salsa Teriyaki	Clásicos	\$ 380	 



Vista admin

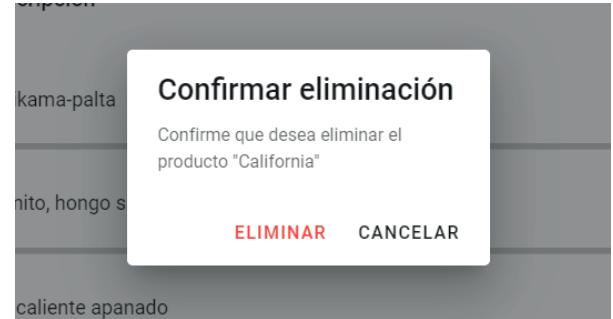
Se muestra una lista de los productos cargados si los hay, y tiene botones para agregar nuevos productos, editar o eliminar. De cada producto se muestra código, imagen, nombre, descripción, categoría y precio. La ruta para acceder esta vista es /admin. El botón de agregar se hizo con un botón flotante de Vuetify, muy común en diseño material. Se agregó la correspondiente confirmación en el botón eliminar.

Para mostrar los productos cargados se utiliza v-for y el componente Product que recibe como prop un producto y dibuja la fila correspondiente. Este componente emite eventos editar y eliminar que son escuchados por su componente padre. Se utilizan dos filtros de Vue: mayuscula y formatoPrecio. Están definidos en main.js; mayuscula pone la primera letra de un texto en mayúscula y formatoPrecio agrega la moneda antes del número.

Lista de productos						
Código	Imagen	Nombre	Descripción	Categoría	Precio	Acciones
1		California	Kanikama-palta	Clásicos	\$ 380.00	 

Componente product-component

Al eliminar un producto, el componente muestra un modal de confirmación, y si el usuario confirma la acción se emite el evento eliminar. La vista Admin escucha este evento y ejecuta su propio método eliminar que elimina el producto y actualiza la base de datos.



Confirmación para eliminar

Edición de productos

Al tocar el botón editar, la vista Admin escucha un evento y ejecuta su propio método editar que establece el dato local editandoCodigo con el código del producto a editar. Dentro del v-for que muestra la lista de productos se utilizan v-if para mostrar el producto o el formulario de edición según corresponda. Si editandoCodigo es igual al código del producto se muestra el componente Edit, sino el componente Product.

1		California	Kanikama-palta	Clásicos	\$ 380.00	 
<div style="display: flex; align-items: center;"> <div style="flex: 1;"> <div>Categoría</div> <input type="text" value="Especiales"/> </div> <div>Vista previa de la imagen</div> </div> <div style="display: flex; align-items: center;">  <div style="margin-left: 10px;"> <div>Imagen del producto</div> <input type="file"/> </div> </div>						
<div style="display: flex; justify-content: space-between;"> <div> <div>Nombre del producto</div> <input type="text" value="Japan"/> </div> <div> <div>Descripción</div> <input type="text" value="Palmito, hongo shiitake, zanahoria con panko exterior."/> </div> <div> <div>Precio</div> <input type="text" value="760.00"/> </div> </div>						
<input type="button" value="LISTO"/>						

Componente edit

El componente Edit recibe como prop el producto a editar y utiliza el componente SushiForm para dibujar el formulario de edición. Le pasa las props producto y modoEdicion y escucha el evento guardar. Cuando este evento sucede se ejecuta el método guardar del componente Edit que emite el evento terminarEdicion.

Cuando la vista Admin escucha este evento vuelve a establecer su dato local editandoCodigo como undefined y vuelve a guardar los datos actualizados en la base de datos.

Creación de productos

El botón Agregar (+) de la vista Admin es un router-link que lleva al path "/add". En esta ruta se muestra la vista Add. Esta vista muestra un botón para volver a la lista con un router-link al path "/admin" y usa el componente SushiForm al igual que el componente de editar.

Natural

Añadir producto

VOLVER

Código del producto
123

Categoría
Hot Rolls

Nombre del producto
Japan

Descripción
Palmito, hongo shitake, zanahoria con panko exterior.

Precio
760

Vista previa de la imagen



Imagen del producto
2021062202...acream.jpg

GUARDAR

Vista Add

Tiene la siguiente data: producto y codigoRepetido. Producto es un JSON con los datos del producto todos en blanco. Y codigoRepetido es un booleano inicialmente establecido en false.

Al usar el componente Form, le pasa las props producto, codigoRepetido y modoEdicion. A diferencia de la vista Edit, en este caso modoEdicion se pasa con valor false. A su vez escucha el evento guardar y ejecuta su método propio guardar que verifica si el código ya existe en otro producto antes de guardar en la base de datos el producto creado. Si ya existe, establece codigoRepetido en true, si no existe agrega el producto a la lista de productos, actualiza la base de datos y redirige a la vista Admin utilizando vue-router.

Componente formulario

El componente SushiForm se encarga de mostrar el formulario para ingresar o modificar los datos de un producto. Incluye validaciones de datos a través de validate de vuertify y utiliza v-on:keydown para restringir el ingreso de algunos caracteres. Para esta entrega, al utilizar base de datos y php se cambió el input de URL de imagen por un input de tipo file, para poder subir una imagen.

Añadir producto

Vista previa de la imagen

Código del producto
Código requerido

Categoría

Nombre del producto
Nombre requerido

Descripción
Descripción requerida

Precio
Precio requerido

Imagen del producto
Imagen requerida

GUARDAR

Componente SushiForm

El componente recibe 3 props. La prop producto es un objeto que contiene todas las propiedades del producto. Cuando el producto es nuevo deben estar vacías. La propiedad modoEdicion se utiliza en un v-if para no mostrar el input del código del producto, ya que este dato no se edita. La propiedad codigoRepetido se usa para mostrar un mensaje de error y un estilo de error para el código cuando ya existe, usando v-if y v-bind:class.

Se utilizan varios atributos en data para cada una de las validaciones de vuety y para inicializar el select de categorías.

Se utiliza `@submit.prevent` para ejecutar el método `guardar` que valida el form con la api js de vuetyf y a su vez emite un evento `guardar` para el padre, que puede ser `create-component` o `edit-component`.

Dejar reseña (Home)

La cuarta y última vista es addReview, se accede a través de la ruta add-review, a partir de un botón en la Home. Tiene una implementación de un formulario para seleccionar una puntuación de 0 a 5 y dejar un comentario. El submit termina impactando en la base de datos. A su vez la Home muestra las reseñas obtenidas a través de la base.



Vista AddReview

Funcionamiento Offline

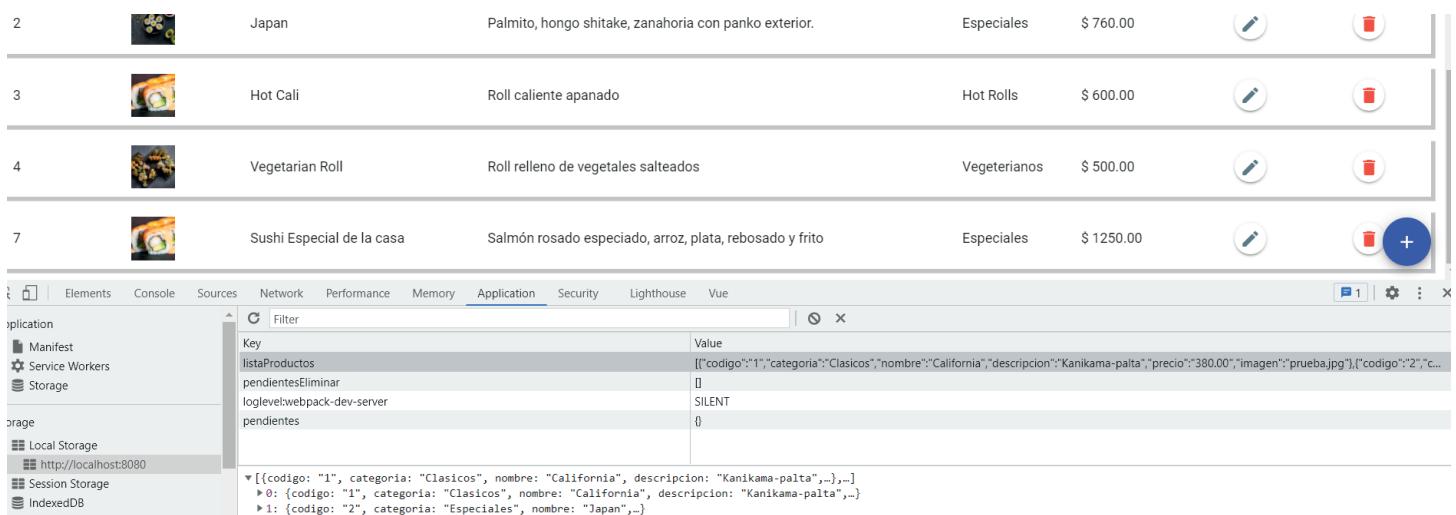
La aplicación utiliza una conexión con el servidor que está implementado con archivos php, que manejan las peticiones HTTP y a su vez leen y escriben en la base de datos. Tanto los productos como las reseñas se guardan en la base de datos.

La administración de productos puede funcionar sin conexión. Tanto la Home como el panel de Administración de productos recuperan los datos de la base de datos para mostrar los productos y guardan una copia en el localstorage del navegador. Luego, si la conexión se pierde momentáneamente, los productos siguen apareciendo.

Además el panel de administración puede seguir funcionando sin conexión. Al cargar un producto nuevo, si no hay conexión el mismo se guarda como pendiente en el localstorage. Al actualizar un producto se hace exactamente lo mismo. Al eliminar un producto, si no hay conexión su código se guarda en una lista de códigos pendientes de eliminar en el localstorage. Para el usuario esto es transparente y se le permite agregar, editar y eliminar productos aún sin conexión.

La aplicación en la vista Admin se suscribe al evento online y comprueba el estado de la conexión. Al estar online, si hay datos pendientes de guardar o de eliminar, actualiza la base de datos según corresponda.

En la implementación actual no funciona correctamente la carga de imágenes offline. Podría ser una mejora futura.



The screenshot shows the Chrome DevTools Application tab with the following details:

- Local Storage:** Key: listaProductos Value: [{"codigo": "1", "categoria": "Clasicos", "nombre": "California", "descripcion": "Kanikama-palta", "precio": "380.00", "Imagen": "prueba.jpg"}, {"codigo": "2", "categoria": "Especiales", "nombre": "Japan", "descripcion": "Palmito, hongo shitake, zanahoria con panko exterior."}]
- Session Storage:** Key: pendientesEliminar Value: []
- IndexedDB:** Key: loglevelwebpack-dev-server Value: SILENT
- IndexedDB:** Key: pendientes Value: []

Datos en localstorage