

Aplicación para dispositivos móviles

Parcial N°1 *Galería SauRa*



Rascioni Miguel - Saurrales Carolina
- DWN3CV – 2023

INFORME

INTRODUCCIÓN

El tema del trabajo es la Galería de Arte; “**SauRa**”. Se podrá acceder a las páginas a través de un menú. En primera instancia estará habilitada por defecto la página del **Home**. Esta página cuenta con el catálogo completo de la galería. Dispone de un filtro para seleccionar por tipo de obra.

El **menú** cuenta con un botón para inicio de sesión. El usuario podrá loguearse al sistema con su usuario=root y clave=123456. De este modo se le habilitará una segunda opción en el menú de “**Agregar Obra**” mediante la cual podrá dar de alta las nuevas obras en la Galería SauRa.

El **footer** cuenta con el ícono de la galería y las redes sociales.



Funcionalidades desarrolladas desde vuejs.

index.html

Se utilizó **v-for**="item in links" y **router-link** :to="item.url" para acceder a las páginas del menú

<router-view></router-view> para visualizar las vistas.

index.js

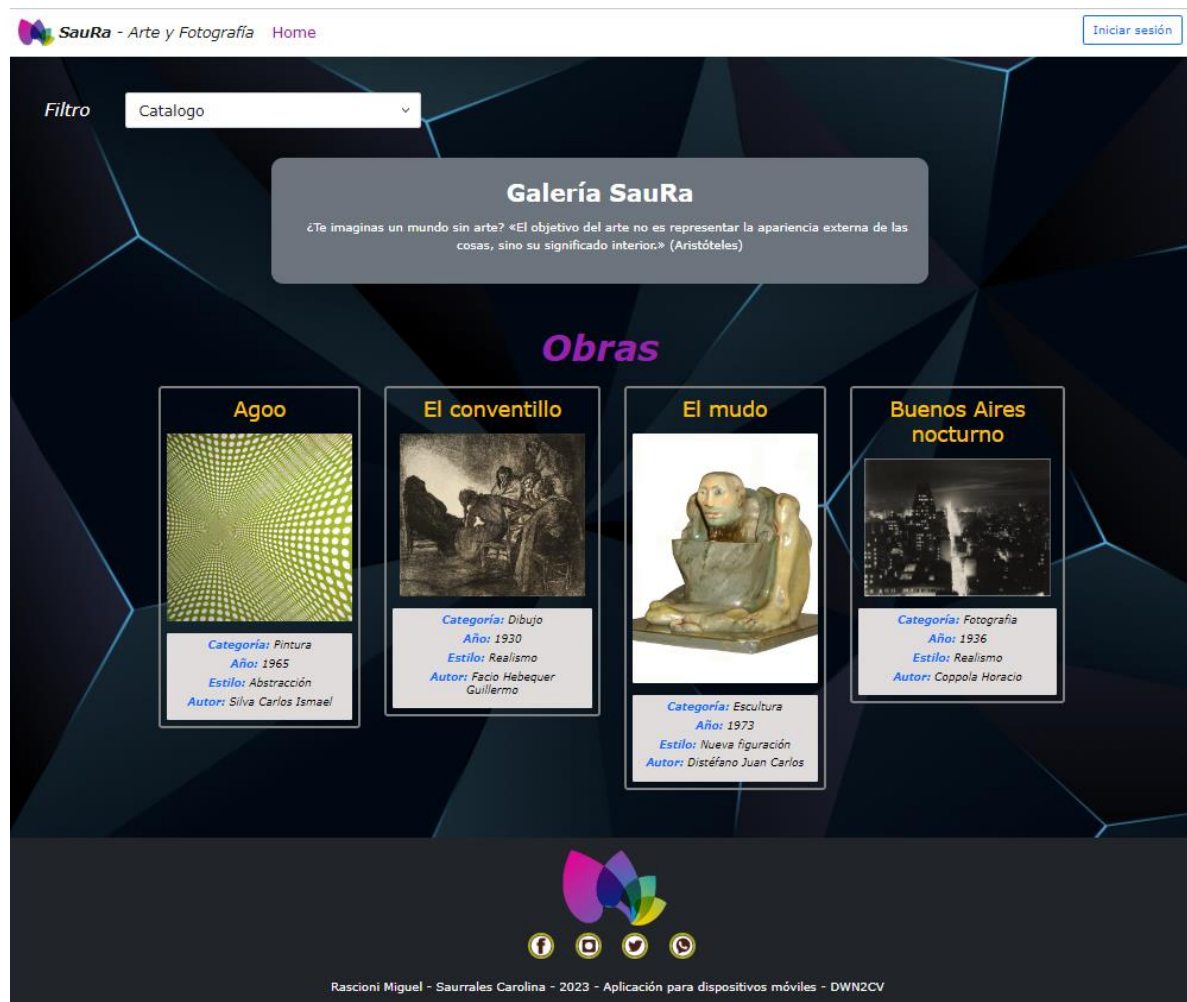
Se declara la **constante** **const routes** = [path: "/", component: home }, { path: "/home", component: home }, { path: "/agregar", component: agregar },]; y **const router = new VueRouter** para las opciones del menú.

Posibles respuestas pensadas para darle al usuario. Flujo de uso de la misma.

En el menú estará activa en color violeta la opción de Home (página de inicio). A la derecha se visualiza el botón de inicio de sesión. Si inicia sesión con su usuario y password podrá acceder a una segunda página “Agregar Obras” En la cual a través de un formulario podrá agregar nuevas obras al catálogo (se explicará en detalle en las próximas páginas)

PRIMER PANTALLA - Home

Por defecto se accede a la página del Home. Al comenzar la sesión se visualiza un menú desplegable para filtrar el catálogo, una breve descripción y cuatro obras, una obra por cada categoría (Pintura, Dibujo, Escultura, Fotografía)



Filtro. Menu desplegable.

Dispone de un filtro para seleccionar su visualización entre Pinturas, Dibujos, Esculturas y Fotografías.

Funcionalidades desarrolladas desde vuejs.

Componente show-categorias.js.

Vue.component('show-obras') para registrar el componente show-categorias.js

Se utilizó **v-model** ="idCategoria" y **v-for**="item in categorias" para mostrar el scroll con las opciones que el usuario puede seleccionar para filtrar por Categoría.

v-if para visualizar una breve descripción de cada categoría filtrada,

Iteración sobre catalogoObras: se utilizó **v-for** para imprimir todos los elementos del array.

LocalStorage: se utilizó **mounted:function()** para recuperar la información de las categorías (scroll de filtrado) y la información del catálogo (cards con imagen y atributos de cada obra)

index.js

Se declaran las const home = { template: `<show-obras></show-obras>` };

Objeto const app= new Vue para armar la aplicación en Vue modularmente: Array **categorías** con toda la información del detalle de categorías. **catalogoObras** con la información de cada card. (título de la obra, imagen y demás atributos). El array de **links** a las opciones del menú.

LocalStorage

Ciclo de vida: **beforeMount** para montar el componente Catálogo y Categoría.

Mounted para marcar en color violeta que página esta acitva.

Posibles respuestas pensadas para darle al usuario. Flujo de uso de la misma.

Por cada una de las opciones que el usuario seleccione, se desplegará a continuación una breve descripción o párrafo. Debajo podrá ver las obras correspondientes a la categoría seleccionada (ej. Escultura). Por cada Obra podrá visualizar el Título, Imagen, categoría, Año, Estilo y Autor.



INICIO DE SESION

Botón de inicio y modal de inicio.

Funcionalidades desarrolladas desde vuejs.

Componente login-modals.js

Vue.component('login-modal') para registrar el componente login-modal.js. Ventana modal.

En panel visible: **v-show** para mostrar de modo condicional el elemento. Si el valor es true mostrará el elemento. Si es false le aplicará la propiedad display: none

En Ingreso usr y passw: **v-on** para vincular un evento al input

En políticas Términos y Condiciones: **v-model** para crear el data binding

v-on para vincular un evento a un botón. Si todas las opciones ingresadas por el usuario son correctas se habilita el botón de inicio de sesión.

props tiene lo que le paso de afuera: props: ['panelLoginVisible', 'ocultarPanelLogin', 'iniciarSesion'],

computed: variables computadas para verificar que haya tildado ambos check box.

methods para validar usuario y password correctos

emit, depende de si se autentico bien o no devuelve true o false: el hijo dispara un evento que va al padre app

ciclos de vida: **beforeMount** para montar el componente y **beforeUpdate** para actualizarlo.

index.html

Se utilizó **login-modal** para hacer visible el panel de login u ocultarlo y para iniciar sesión

@click="toogleLogin">{{ login ? 'Cerrar sesión' : 'Iniciar sesión'}} para verificar el inicio o cierre de sesión

index.js

Methods: para inicio de sesión authenticated (status y iniciarSesion() y toogleLogin () para verificar el inicio o cierre de la sesión.

Posibles respuestas pensadas para darle al usuario. Flujo de uso de la misma.

Si el usuario pasa el cursor sobre la esquina superior derecha, el botón de “Iniciar sesión” cambia de color a celeste pleno

Al hacer click sobre el botón se le despliega la pantalla de logueo.

Debe completar todos los campos correctamente para que se habilite el botón de Iniciar Sesión.

Si alguno es incorrecto, por ejemplo, la clave, el sistema le indicará el error:

Inicio de sesión



Por favor, ingrese su usuario

Por favor, ingrese la contraseña genérica provista.

Para poder continuar, debe aceptar las políticas de privacidad, y los términos y condiciones

- ☒ Políticas de privacidad
- ☒ Términos y condiciones

La contraseña es incorrecta

Iniciar Sesión


Cancelar

Si todos los datos son correctos, el usuario iniciará sesión y a continuación se habilitará en el menú la opción “[Agregar Obra](#)”



SEGUNDA PANTALLA – Agregar obra

Haciendo click en el menú “Agregar Obra” el usuario ingresará a la página donde podrá dar de alta nuevas obras a través de un formulario.



SauRa - Arte y Fotografía Home Agregar Obra Cerrar sesión

Nueva Obra

Título de obra

Archivo.jpg o link a la imagen

Texto alternativo para la imagen

- Seleccionar -

Año de la obra

Estilo artístico

Nombre del artista

Agregar

Rascioni Miguel - Saurrales Carolina - 2023 - Aplicación para dispositivos móviles - DWN2CV

Formulario de Nueva Obra

Funcionalidades desarrolladas desde vuejs.

Componente nueva-obra.js

Vue.component('nueva-obra') para registrar el componente nueva-obra.js

Se utilizó **form v-on:submit.prevent @submit="controlarCampos"** para suministrar el formulario a través del cual el usuario dará de alta la nueva obra.

Validaciones, ejemplo: **v-if="!this.errorTitulo"** debe completar el nombre de la obra. Una vez validados todos los campos, se muestra debajo lo ingresado. Se aplica un **v-if="this.nuevaObra.length"** para validar si se dio de alta alguna obra y poder mostrarla.

LocalStorage

Se utilizó **beforeMount**: para leer de localStorage las Categorías para el dropdown de selección. Para guardar en el localStorage las obras agregadas y luego se puedan visualizar en el Home junto a las demás obras del catálogo se utilizó **beforeDestroy**

methods

Se utilizaron los métodos **agregarObra** para hacer un **push** de todos los datos referentes a cada una de las obras. Y **controlarCampos** para validar que los datos del formulario no tengan errores o estén vacíos. Ej. **this.tituloObra != "" ? this.errorTitulo=true : this.errorTitulo=false;**

index.js

Se declaran las const agregar = { template: `

Posibles respuestas pensadas para darle al usuario. Flujo de uso de la misma.

A través del formulario el usuario podrá dar de alta una nueva obra. Para esto debe llenar la totalidad de los campos a excepción del alt que es opcional.

En el segundo input se debe colocar el nombre del archivo con su extensión que se encuentra alojado en el directorio /img o se puede poner la url de la imagen, ejemplo:

<https://www.bellasartes.gob.ar/media/uploads/coleccion/8053.jpg>

Si alguno de los campos no se completa o el año ingresado no está entre 1 y 9999, se indicará el mensaje de error en el campo que corresponda.



The image shows a web form titled "Nueva Obra" with a dark background and a colorful abstract graphic on the left. The form contains several input fields and a dropdown menu. The following table summarizes the state of each field and any associated error messages:

Field	Value	Error Message
Nombre de la obra	Attika	
URL de la imagen	https://encrypted-tbn0.gstatic.com/im...	
Título de la obra	Pintura Attika	
Categoría	- Seleccionar -	No eligió una categoría
Año de la obra		El año debe estar entre 1 y 9999
Estilo artístico		Debe completar el estilo artístico
Nombre del artista		Debe completar el nombre del artista

At the bottom of the form is a blue button labeled "Agregar".

Si los datos son completados correctamente se mostrará en la tabla inferior la obra que se agregó.

Nueva Obra

- Seleccionar -

Agregar

Obra agregada

Título: Intihuasi II	Portada: https://www.bellasartes.gob.ar/media/uploads/coleccion/8053.jpg
Alternativo: Pintura Intihuasi II	Categoría: Pintura
Año: 1973	Estilo: Abstracción
Artista: Puente Alejandro	

Título: Attika	Portada: https://www.bellasartes.gob.ar/media/uploads/coleccion/7356_edit.jpg
Alternativo: Pintura Attika	Categoría: Pintura
Año: 1957	Estilo: Abstracción
Artista: Vasarely Victor	

Luego el usuario podrá ir al home y verificar en la opción filtrar por Pinturas, que además de la pintura existente al inicio de la sesión, se agregaron las dos últimas que agregó.

SauRa - Arte y Fotografía
 [Home](#)
[Agregar Obra](#)

Cerrar sesión

Filtro

Pintura

Pintura

"Cuando miramos una obra de arte estamos sometidos al recuerdo de una multitud de cosas que para bien o para mal influyen sobre nuestros gustos". (Gombrich)

Obras

Agoo

Categoría: Pintura
Año: 1965
Estilo: Abstracción
Autor: Silva Carlos Ismael

Intihuasi II

Categoría: Pintura
Año: 1973
Estilo: Abstracción
Autor: Puente Alejandro

Attika

Categoría: Pintura
Año: 1957
Estilo: Abstracción
Autor: Vasarely Victor

Rascioni Miguel - Saurrales Carolina - 2023 - Aplicación para dispositivos móviles - DWN2CV

CIERRE DE SESION

Botón de cierre de sesión

Funcionalidades desarrolladas desde vuejs.

index.js


toggleLogin() para verificar el cierre de sesión. Se utiliza el **if/else** para setear las variables de login a false y volver al home: `this.login = false;` `this.$router.push('/home');`
`this.links[1].enable=false;`

index.html

@click="toggleLogin">{{ login ? 'Cerrar sesión' : 'Iniciar sesión' }} para verificar el inicio o cierre de sesión

Posibles respuestas pensadas para darle al usuario. Flujo de uso de la misma.

Si el usuario pasa el cursor sobre la esquina superior derecha, el botón de “Cerrar sesión” cambia de color a gris pleno

 **SauRa** - Arte y Fotografía [Home](#) [Agregar Obra](#)

Cerrar sesión

Al hacer click sobre el botón podrá cerrar su sesión de forma segura.