import { Link } from 'react-router-dom';

import { MenuItem } from '@components'; // Componente para renderizar cada elemento del menú

import { Image } from '@profabric/react-components'; // Componente de imagen

import styled from 'styled-components'; // Utiliza styled-components para estilizar elementos

import { SidebarSearch } from '@app/components/sidebar-search/SidebarSearch'; // Barra de búsqueda en el sidebar

import i18n from '@app/utils/i18n'; // Para traducciones

import { useAppSelector } from '@app/store/store'; // Hook para acceder al estado global de la aplicación

// Interfaz que define la estructura de un elemento del menú

export interface IMenuItem {

  name: string; // Nombre del elemento del menú

  icon?: string; // Ícono asociado al elemento

  path?: string; // Ruta a la que redirige el menú

  children?: Array<IMenuItem>; // Submenús (si los tiene)

}

// Definición de los elementos del menú

export const MENU: IMenuItem[] = [

  {

    name: i18n.t('menusidebar.label.dashboard'), // Traducción para "Dashboard"

    icon: 'fas fa-tachometer-alt nav-icon', // Ícono de FontAwesome

    path: '/', // Ruta de redirección

  },

  {

    name: i18n.t('menusidebar.label.blank'), // Traducción para "Blank"

    icon: 'fas fa-wrench nav-icon', // Ícono de FontAwesome

    path: '/blank', // Ruta de redirección

  },

  {

    name: i18n.t('menusidebar.label.mainMenu'), // Menú principal con submenús

    icon: 'far fa-caret-square-down nav-icon', // Ícono de FontAwesome

    children: [

      {

        name: i18n.t('menusidebar.label.subMenu'), // Submenú 1

        icon: 'fas fa-hammer nav-icon', // Ícono de FontAwesome

        path: '/sub-menu-1', // Ruta de redirección

      },

      {

        name: i18n.t('menusidebar.label.blank'), // Submenú 2

        icon: 'fas fa-cogs nav-icon', // Ícono de FontAwesome

        path: '/sub-menu-2', // Ruta de redirección

      },

    ],

  },

];

// Estilos para la imagen de la marca utilizando styled-components

const StyledBrandImage = styled(Image)`

  float: left;

  line-height: 0.8;

  margin: -1px 8px 0 6px;

  opacity: 0.8;

  --pf-box-shadow: 0 10px 20px rgba(0, 0, 0, 0.19),

    0 6px 6px rgba(0, 0, 0, 0.23) !important;

`;

// Estilos para la imagen del usuario

const StyledUserImage = styled(Image)`

  --pf-box-shadow: 0 3px 6px #00000029, 0 3px 6px #0000003b !important;

`;

// Componente principal del Sidebar

const MenuSidebar = () => {

  // Accede a información del usuario actual y configuraciones desde el estado global

  const currentUser = useAppSelector((state) => state.auth.currentUser); // Usuario actual

  const sidebarSkin = useAppSelector((state) => state.ui.sidebarSkin); // Tema del sidebar

  const menuItemFlat = useAppSelector((state) => state.ui.menuItemFlat); // Si los ítems deben ser planos

  const menuChildIndent = useAppSelector((state) => state.ui.menuChildIndent); // Si los submenús deben tener indentación

  return (

    // Estructura principal del sidebar

    <aside className={`main-sidebar elevation-4 ${sidebarSkin}`}>

      {/\* Enlace a la página de inicio con la imagen de la marca \*/}

      <Link to="/" className="brand-link">

        <StyledBrandImage

          src="img/acroming.png" // Ruta de la imagen de la marca

          alt="Acroming Ecuador"

          width={33}

          height={33}

          rounded

        />

        <span className="brand-text font-weight-light">Geoportal</span>

      </Link>

      <div className="sidebar">

        {/\* Panel del usuario \*/}

        <div className="user-panel mt-3 pb-3 mb-3 d-flex">

          <div className="image">

            <StyledUserImage

              src={currentUser?.photoURL} // Imagen del usuario actual

              fallbackSrc="/img/default-profile.png" // Imagen por defecto si no hay imagen de usuario

              alt="User"

              width={34}

              height={34}

              rounded

            />

          </div>

          <div className="info">

            <Link to={'/profile'} className="d-block">

              {currentUser?.email} {/\* Email del usuario actual \*/}

            </Link>

          </div>

        </div>

        {/\* Barra de búsqueda dentro del sidebar \*/}

        <div className="form-inline">

          <SidebarSearch /> {/\* Componente para buscar elementos en el menú \*/}

        </div>

        {/\* Menú de navegación \*/}

        <nav className="mt-2" style={{ overflowY: 'hidden' }}>

          <ul

            className={`nav nav-pills nav-sidebar flex-column${

              menuItemFlat ? ' nav-flat' : ''

            }${menuChildIndent ? ' nav-child-indent' : ''}`}

            role="menu"

          >

            {/\* Mapeo de los ítems del menú dinámico \*/}

            {MENU.map((menuItem: IMenuItem) => (

              <MenuItem

                key={menuItem.name + menuItem.path} // Clave única para cada ítem

                menuItem={menuItem} // Pasa el ítem al componente MenuItem

              />

            ))}

          </ul>

        </nav>

      </div>

    </aside>

  );

};

export default MenuSidebar;

**Descripción por secciones:**

1. **Importaciones**:
   * **react-router-dom**: Se usa para la navegación de enlaces entre rutas.
   * **styled-components**: Utilizado para crear estilos personalizados para imágenes.
   * **useAppSelector**: Hook para acceder al estado global usando Redux.
   * **i18n**: Utilizado para manejar traducciones.
2. **Interfaz IMenuItem**: Define la estructura de cada elemento del menú:
   * name: El nombre del ítem que se muestra.
   * icon: El ícono que se muestra junto al nombre.
   * path: La ruta a la que redirige cuando se hace clic.
   * children: Submenús, si los hay.
3. **Constante MENU**:
   * Define los elementos del menú, algunos con submenús, que se traducen usando i18n.
4. **Estilos personalizados**:
   * **StyledBrandImage**: Aplica estilos a la imagen de la marca (logo).
   * **StyledUserImage**: Estilos para la imagen del usuario.
5. **MenuSidebar**:
   * Accede al estado global para obtener el usuario actual, el tema del sidebar, y configuraciones visuales.
   * Renderiza el logo de la empresa, el panel de usuario, una barra de búsqueda, y el menú dinámico basado en los elementos de MENU.

**¿Cómo usar este código?**

Guarda este código como MenuSidebar.tsx y asegúrate de tener los siguientes puntos configurados:

* Las rutas de las imágenes y componentes importados deben estar correctas.
* El MENU se puede ajustar según las necesidades de tu proyecto, añadiendo o eliminando elementos del menú.
* Asegúrate de que el estado global esté bien configurado si usas Redux o algo similar.