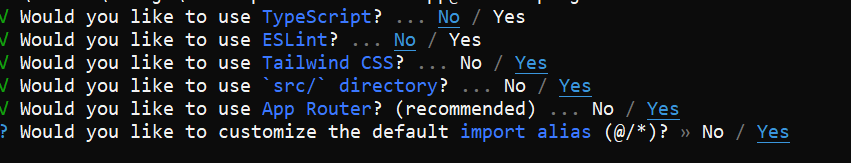
npx create-next-app@latest nextjs-project1



npm i mongoose zod daisyui react-hot-toast

Zod est une bibliothèque de déclaration et de validation de schéma

daisyui : bibliothèque pour Tailwind CSS



 plugins: [require('daisyui')],



/\*\* @type {import('next').NextConfig} \*/

const nextConfig = {

    images:{

        domains:["firebasestorage.googleapis.com","res.cloudinary.com"]

        },

        env: {

            MONGODB\_URI:"mongodb+srv://hassan:messk2015@cluster0.yfhhu.mongodb.net/DataBaseCommerce2024?retryWrites=true&w=majority"

            }

}

module.exports = nextConfig



import mongoose from 'mongoose'

export default async function dbConnect() {

  try {

    await mongoose.connect(process.env.MONGODB\_URI)

  } catch (error) {

    throw new Error('Connection failed!')

  }

}



import mongoose from 'mongoose'

export class Categorie  {

  \_id;

  nomcategorie;

  imagecategorie;

}

const categorieSchema = new mongoose.Schema(

  {

    nomcategorie:{ type: String, required: true,unique:true },

    imagecategorie :{ type: String, required: false }

  },

  {

    timestamps: true,

  }

)

const categorieModel =

  mongoose.models.Categorie || mongoose.model('Categorie', categorieSchema)

export default categorieModel



'use server'

import { revalidatePath } from 'next/cache'

import CategorieModel from './categorieModel'

import dbConnect from './dbConnect'

import { z } from 'zod'

export async function createProduct(prevState, formData) {

  const schema = z.object({

    nomcategorie: z.string().min(3),

    imagecategorie: z.string().min(1),

   })

  const parse = schema.safeParse({

    nomcategorie: formData.get('nomcategorie'),

    imagecategorie: formData.get('imagecategorie'),

  })

  if (!parse.success) {

    console.log(parse.error)

    return { message: 'Form data is not valid' }

  }

  const data = parse.data

  try {

    await dbConnect()

    const product = new CategorieModel(data)

    await product.save()

    revalidatePath('/')

    return { message: `Created product ${data.nomcategorie}` }

  } catch (e) {

    return { message: 'Failed to create product' }

  }

}

export async function deleteProduct(formData) {

  const schema = z.object({

    \_id: z.string().min(1),

    nomcategorie: z.string().min(1),

  })

  const data = schema.parse({

    \_id: formData.get('\_id'),

    nomcategorie: formData.get('nomcategorie'),

  })

  try {

    await dbConnect()

    await CategorieModel.findOneAndDelete({ \_id: data.\_id })

    revalidatePath('/')

    console.log({ message: `Deleted product ${data.nomcategorie}` })

    return { message: `Deleted product ${data.nomcategorie}` }

  } catch (e) {

    return { message: 'Failed to delete product' }

  }

}



import Image from 'next/image'

import { Toaster } from 'react-hot-toast'

import CreateForm from './createCateg'

import dbConnect from '@/lib/dbConnect'

import CategorieModel from '@/lib/categorieModel'

export default async function Home() {

  await dbConnect()

  const products = (await CategorieModel.find({}).sort({

    \_id: -1,

  }))

  return (

    <div className="mx-auto max-w-2xl lg:max-w-7xl">

      <div className="flex justify-between items-center">

        <h1 className="font-bold py-10 text-2xl">Admin Categories</h1>

        <Toaster />

        <CreateForm />

      </div>

      <table className="table">

        <thead>

          <tr>

            <th>Image</th>

            <th>Name</th>

            <th>Actions</th>

          </tr>

        </thead>

        <tbody>

          {products.length === 0 ? (

            <tr>

              <td colSpan={5}>No Category found</td>

            </tr>

          ) : (

            products.map((product) => (

              <tr key={product.\_id}>

                <td>

                  <Image

                    src={product.imagecategorie}

                    alt={product.nomcategorie}

                    width={80}

                    height={80}

                    className="rounded-lg"

                  />

                </td>

                <td>{product.nomcategorie}</td>

               <td>

                </td>

              </tr>

            ))

          )}

        </tbody>

      </table>

    </div>

  )

}



'use client'

import { useFormState, useFormStatus } from 'react-dom'

import { createProduct } from '@/lib/actions'

import { useEffect, useRef } from 'react'

import toast from 'react-hot-toast'

export default function CreateForm() {

  const [state, formAction] = useFormState(createProduct, {

    message: '',

  })

  const { pending } = useFormStatus()

  const ref = useRef()

  useEffect(() => {

    if (state.message.indexOf('Created product') === 0) {

      ;(document.getElementById('my\_modal\_3')).close()

      ref.current?.reset()

      toast(state.message)

    } else if (state.message) {

      toast(state.message)

    }

  }, [state.message])

  return (

    <div>

      <button

        className="btn btn-primary"

        onClick={() =>

          (document.getElementById('my\_modal\_3')).showModal()

        }

      >

        Create Category

      </button>

      <dialog id="my\_modal\_3" className="modal">

        <div className="modal-box">

          <h2 className="tex-2xl font-bold pm-4">Create Category</h2>

          <form ref={ref} action={formAction}>

            <div className="form-control w-full max-w-xs py-4">

              <label htmlFor="nomcategorie">Name</label>

              <input

                type="text"

                id="nomcategorie"

                name="nomcategorie"

                className="input input-bordered w-full max-w-xs"

                required

              />

            </div>

            <div className="form-control w-full max-w-xs py-4">

              <label htmlFor="image">Image</label>

              <input

                type="text"

                id="imagecategorie"

                name="imagecategorie"

                className="input input-bordered w-full max-w-xs"

                required

                defaultValue="https://res.cloudinary.com/iset-sfax/image/upload/v1658754288/images/flash1.jpg.jpg"

              />

            </div>

            <button

              className="btn btn-primary mr-3"

              type="submit"

              disabled={pending}

            >

              Create

            </button>

            <button

              type="button"

              className="btn btn-ghost"

              onClick={() =>

                (document.getElementById('my\_modal\_3')).close()

              }

            >

              Back

            </button>

          </form>

        </div>

      </dialog>

    </div>

  )

}



import Link from 'next/link'

export default function Home() {

  return (

    <main className="flex min-h-screen flex-col items-center justify-between p-24">

      <div className="z-10 max-w-5xl w-full items-center justify-between font-mono text-sm lg:flex">

      <ul>

      <li>

        <Link href="/">Home</Link>

      </li>

      <li>

        <Link href="/categories">Catégorie</Link>

      </li>

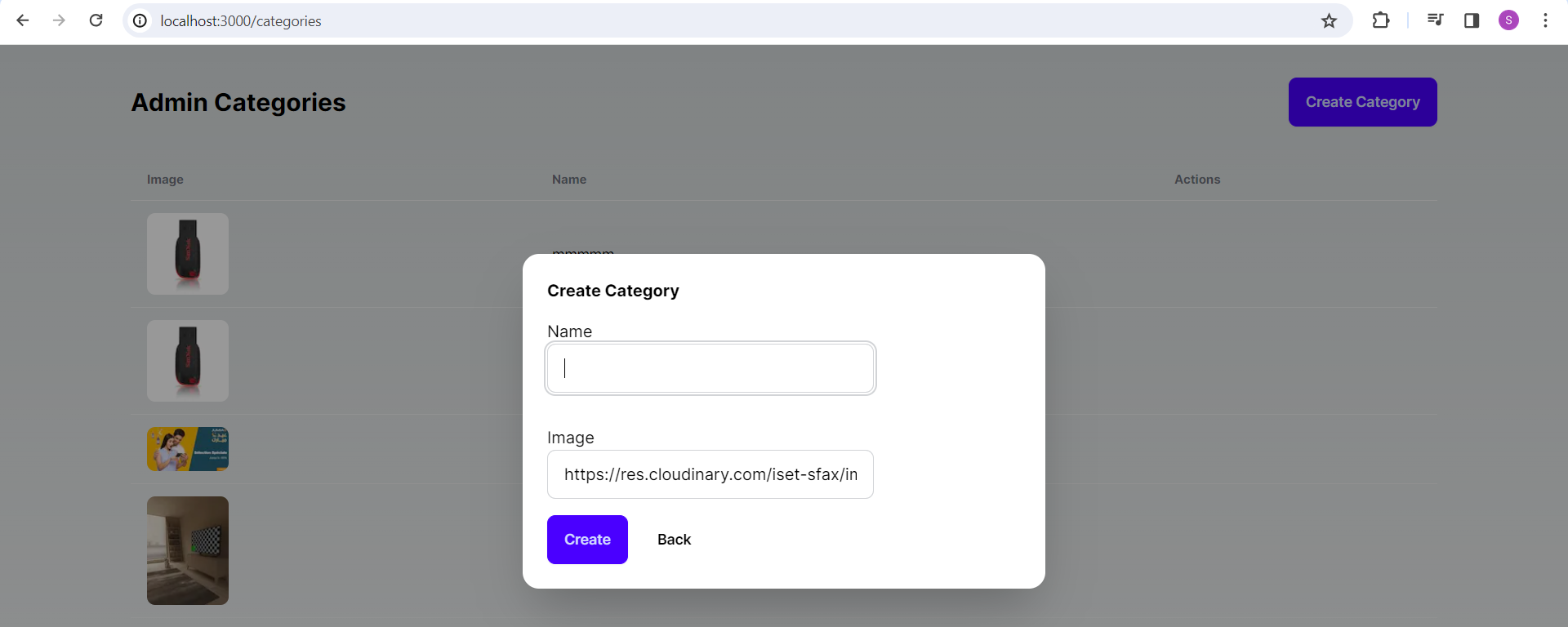
      </ul>

      </div>

    </main>

  )

}

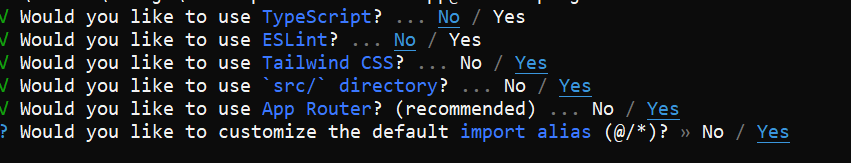


\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* VERSION 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

npx create-next-app@latest nextjs-project1



npm i material-react-table @mui/material @mui/x-date-pickers @mui/icons-material @emotion/react @emotion/styled

npm i react-hot-toast



import ListCategories from "@/components/categories/listCategories"

async function getData() {

    const res = await fetch('http://localhost:3001/api/categories')

    // The return value is \*not\* serialized

    // You can return Date, Map, Set, etc.

    if (!res.ok) {

      // This will activate the closest `error.js` Error Boundary

      throw new Error('Failed to fetch data')

    }

    return res.json()

  }

  export default async function Page() {

    const data = await getData()

   return (

    <main>

    <h1 className="font-bold py-10 text-2xl">Admin Categories</h1>

    <ListCategories categories={data} />

    </main>

   );

  }



'use client'

import React from 'react'

import { MaterialReactTable} from 'material-react-table';

const ListCategories = ({categories}) => {

    const columns = React.useMemo(

        () => [

            {

                accessorKey: 'imagecategorie',

                header: 'Image',

                Cell: ({ cell }) => (

                       <img

                            src={cell.getValue()}

                            width={100}

                            height={120}

                            className="rounded-lg"

                            alt="Image Category"

                        />

                    )

                 },

          {

            accessorKey: 'nomcategorie',

            header: 'Name',

            size: 150,

          },

          {

            accessorKey: '\_id',

            header: 'Actions',

            size: 100,

            Cell: ({ cell, row }) => (

                <div >

                    <button className="btn btn-ghost text-success"

                        onClick={() => {}}

                        size="md"

                    >

                       Edit

                    </button>

                    <button className="btn btn-ghost text-warning"

                        onClick={() => {}}

                        size="md"

                    >

                        Delete

                    </button>

                </div>

            ),

        },

          ],

        [],

      );

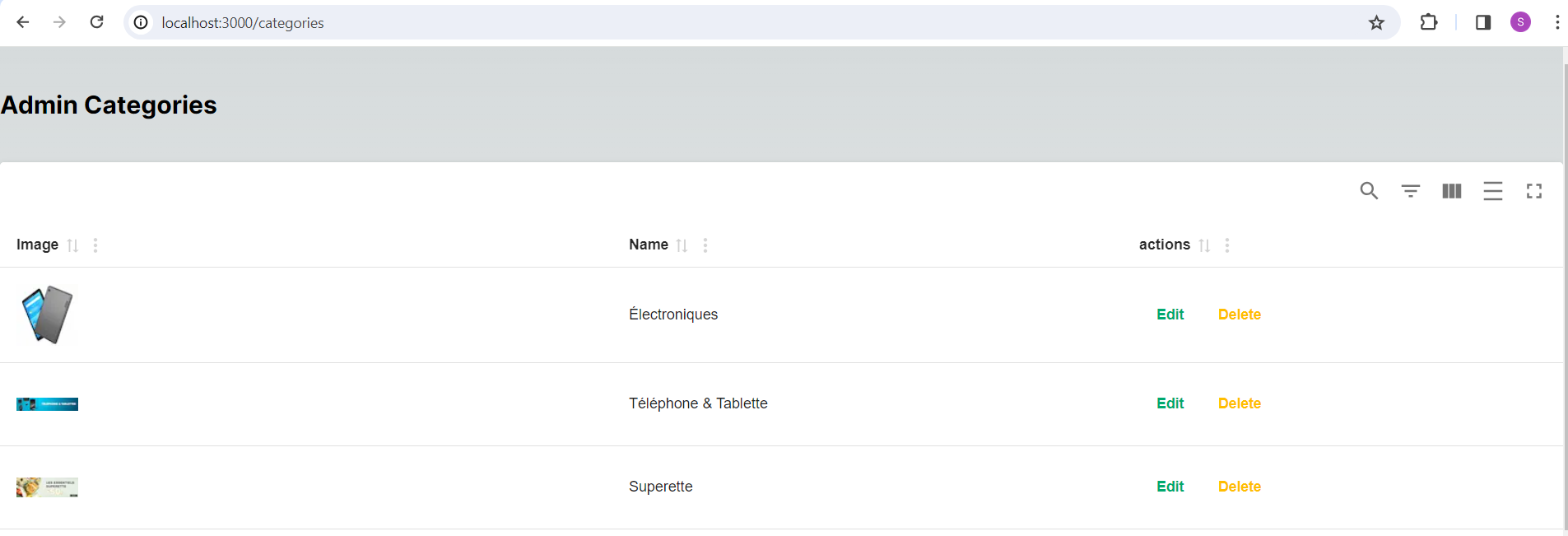
  return (

    <MaterialReactTable columns={columns} data={categories} />

 )

}

export default ListCategories





'use client'

import { useFormState, useFormStatus } from 'react-dom'

import { addCategory} from '@/lib/actionsCategorie'

import { useEffect, useRef } from 'react'

import toast from 'react-hot-toast'

export default function CreateForm() {

  const [state, formAction] = useFormState(addCategory, {

    message: '',

  })

  const { pending } = useFormStatus()

  const ref = useRef()

  useEffect(() => {

   if (state.message){

      (document.getElementById('my\_modal\_3')).close()

      ref.current?.reset()

      toast(state.message)

    }

  }, [state.message])

  return (

    <div>

      <button

        className="btn btn-primary"

        onClick={() =>

          (document.getElementById('my\_modal\_3')).showModal()

        }

      >

        Create Category

      </button>

      <dialog id="my\_modal\_3" className="modal">

        <div className="modal-box">

          <h2 className="tex-2xl font-bold pm-4">Create Category</h2>

          <form ref={ref} action={formAction}>

            <div className="form-control w-full max-w-xs py-4">

              <label htmlFor="nomcategorie">Name</label>

              <input

                type="text"

                id="nomcategorie"

                name="nomcategorie"

                className="input input-bordered w-full max-w-xs"

                required

              />

            </div>

            <div className="form-control w-full max-w-xs py-4">

              <label htmlFor="image">Image</label>

              <input

                type="text"

                id="imagecategorie"

                name="imagecategorie"

                className="input input-bordered w-full max-w-xs"

                required

                defaultValue="https://res.cloudinary.com/iset-sfax/image/upload/v1658754288/images/flash1.jpg.jpg"

              />

            </div>

            <button

              className="btn btn-primary mr-3"

              type="submit"

              disabled={pending}

            >

              Create

            </button>

            <button

              type="button"

              className="btn btn-ghost"

              onClick={() =>

                (document.getElementById('my\_modal\_3')).close()

              }

            >

              Back

            </button>

          </form>

        </div>

      </dialog>

    </div>

  )

}



'use server'

import { revalidatePath } from 'next/cache'

export async function addCategory(prevState, formData) {

    const nomcategorie= formData.get('nomcategorie')

    const imagecategorie= formData.get('imagecategorie')

    const categorie ={

        nomcategorie,

        imagecategorie

    }

    try {

        const response = await fetch("http://localhost:3001/api/categories", {

            method: 'POST',

            headers: {

            'Content-Type': 'application/json',

            },

            body: JSON.stringify(categorie),

        });

        const result = await response.json();

        console.log(result)

        revalidatePath('/categories')

       if(result.nomcategorie) return { message: `Created category ${result.nomcategorie}` }

       else  return { message: 'Failed to create category' }

    } catch (e) {

  return { message: 'Failed to create category' }

}

}



import ListCategories from "@/components/categories/listCategories"

import { Toaster } from 'react-hot-toast'

import CreateForm from './createCateg'

async function getData() {

    const res = await fetch('http://localhost:3001/api/categories')

    // The return value is \*not\* serialized

    // You can return Date, Map, Set, etc.

    if (!res.ok) {

      // This will activate the closest `error.js` Error Boundary

      throw new Error('Failed to fetch data')

    }

    return res.json()

  }

  export default async function Page() {

    const data = await getData()

   return (

    <main>

    <h1 className="font-bold py-10 text-2xl">Admin Categories</h1>

    <Toaster />

    <CreateForm />

    <ListCategories categories={data} />

    </main>

   );

  }

