

E:/Documentos/Clases/proyecto/cartilla/app/src/App.tsx

```
import React, { useEffect, useState } from "react";
import {
  IonApp,
  IonContent,
  IonHeader,
  IonIcon,
  IonLabel,
  IonRouterOutlet,
  IonSelect,
  IonSelectOption,
  IonTabBar,
  IonTabButton,
  IonTabs,
  IonTitle,
  IonToolbar,
  IonButton,
  setupIonicReact,
} from "@ionic/react";
import niña from "../assets/Img/girl.webp";
import {
  personOutline,
  chevronDown,
  chevronUp,
  arrowForward,
} from "ionicons/icons";
import { SplashScreen } from "@capacitor/splash-screen";
import { IonReactRouter } from "@ionic/react-router";
import { Redirect, Route } from "react-router";

import { Storage } from "@ionic/storage";

import CustomHome from "../assets/icons/HomeNew.svg";
import CustomActivities from "../assets/icons/ActivitiesNew.svg";
import CustomSearch from "../assets/icons/SerachNew.svg";
import CustomProfile from "../assets/icons/ProfileNew.svg";
import CustomAdd from "../assets/icons/AddNew.svg";

/* Core CSS required for Ionic components to work properly */
import "@ionic/react/css/core.css";
import "@ionic/react/css/normalize.css";
import "@ionic/react/css/structure.css";
import "@ionic/react/css/typography.css";
import "../src/Styles/globals.css";
import "@ionic/react/css/padding.css";
```

```

import "@ionic/react/css/float-elements.css";
import "@ionic/react/css/text-alignment.css";
import "@ionic/react/css/text-transformation.css";
import "@ionic/react/css/flex-utils.css";
import "@ionic/react/css/display.css";
import "@ionic/react/css/palettes/dark.class.css";
/* import '@ionic/react/css/palettes/dark.always.css'; */
/*import "@ionic/react/css/palettes/dark.system.css";*/

import "./theme/variables.css";
import Main from "./pages/main/Main";
import Search from "./pages/search/Search";
import Suggestion from "./pages/suggestion/Suggestion";
import AllActivities from "./components/activities/AllActivities";
import Activities from "./components/activities/Activities";
import AddActivity from "./pages/AddActivity/AddActivity";

setupIonicReact();

const App: React.FC = () => {
  const [storage, setStorage] = useState<Storage | null>(null);
  const [userData, setUserData] = useState<boolean>(false);
  const [selectedOption, setSelectedOption] = useState<string | undefined>();

  const handleSelectChange = (event: CustomEvent) => {
    setSelectedOption(event.detail.value);
  };

  const handleContinue = async () => {
    await saveUserData(true);
    setUserData(true);
  };

  useEffect(() => {
    // Inicializa el almacenamiento y recupera el estado
    const initStorage = async () => {
      const storage = new Storage();
      await storage.create();
      setStorage(storage);

      const savedData = await storage.get("userData");
      if (savedData) {
        setUserData(savedData);
      }
    };
    initStorage();
  });

```

```

        SplashScreen.hide();
    }, []);

    // Función para guardar datos
    const saveUserData = async (data: boolean) => {
        if (storage) {
            await storage.set("userData", data);
            console.log("Datos guardados exitosamente.");
        }
    };

    const customAlertOptions = {
        header: "Rol",
        subHeader: "Selecciona tu rol",
        translucent: true,
    };

    return (
        <IonApp className="basics">
            {userData && (
                <IonReactRouter>
                    <IonTabs>
                        <IonRouterOutlet>
                            <Route exact path="/main" component={Main} />
                            <Route path="/search" component={Search} />
                            <Route path="/allfilter/:subcategory" component={AllActivities} />
                            <Route path="/all" component={AllActivities} />
                            <Route path="/activities/:category/:id" component={Activities} />
                            <Route exact path="/suggestions" component={Suggestion} />
                            <Route exact path="/addActivity" component={AddActivity} />
                            <Redirect exact from="/" to="/main" />
                        </IonRouterOutlet>
                        <IonTabBar className="hometab" slot="bottom">
                            <IonTabButton tab="main" href="/main">
                                <IonIcon icon={CustomHome} />
                            </IonTabButton>
                            <IonTabButton tab="search" href="/search">
                                <IonIcon icon={CustomSearch} />
                            </IonTabButton>
                            <IonTabButton tab="addActivity" href="/addActivity">
                                <IonIcon icon={CustomAdd} />
                            </IonTabButton>
                            <IonTabButton tab="all" href="/all">
                                <IonIcon icon={CustomActivities} />
                            </IonTabButton>
                            <IonTabButton tab="suggestions" href="/suggestions">

```

```

        <IonIcon icon={CustomProfile} />
      </IonTabButton>
    </IonTabBar>
  </IonTabs>
</IonReactRouter>
)}}
{!userData && (
  <>
    <IonContent>
      <div className="select_roll_container">
        <div className="w-full flex flex-col content-center items-center justify-center">
          <h1 className="poppins text-white font-extrabold text-3xl mb-5">Bienvenidos</h1>
          <img src={niña} className="w-[60%] self-center" />
        </div>
        <div className="relative bg-white w-[100%] h-[200px] p-6 rounded-3xl edit_shadow">
          <div className="w-[95%]">
            <IonSelect
              value={selectedOption}
              placeholder="ESCOGER ROL"
              onChange={handleSelectChange}
              className="no-focus-indicator border border-black pl-[10%] rounded-3xl"
              toggleIcon={chevronDown}
              expandedIcon={chevronUp}
              interfaceOptions={customAlertOptions}
            >
              <IonIcon
                slot="start"
                icon={personOutline}
                aria-hidden="true"
                color="medium"
                className="poppins"
              ></IonIcon>
              <IonSelectOption value="1">Padre / Madre</IonSelectOption>
              <IonSelectOption value="2">Maestro</IonSelectOption>
              <IonSelectOption value="3">Cuidador</IonSelectOption>
            </IonSelect>
          </div>
          <div className="rounded-full absolute -bottom-8 overflow-hidden p-0.5 bg-white">
            <IonButton
              onClick={handleContinue}
              disabled={!selectedOption}
              shape="round"
              size="large"
            >
              <IonIcon
                slot="icon-only"

```

```

        icon={arrowForward}
        aria-hidden="true"
        color="light"
      ></IonIcon>
    </IonButton>
  </div>
</div>
</div>
</IonContent>
</>
  )}
</IonApp>
);
};

```

export default App;

E:/Documentos/Clases/proyecto/cartilla/app/src/main.tsx

```

import React from 'react';
import { createRoot } from 'react-dom/client';
import App from './App';
import './src/Styles/globals.css';
import { UserProvider } from './components/context/UserContext';

const container = document.getElementById('root');
const root = createRoot(container!);
root.render(
  <React.StrictMode>
    <UserProvider>
      <App />
    </UserProvider>
  </React.StrictMode>
);

```

E:/Documentos/Clases/proyecto/cartilla/app/src/vite-env.d.ts

```
/// <reference types="vite/client" />
```

E:/Documentos/Clases/proyecto/cartilla/app/src/activities/userData.json

```

{
  "login": false
}

```

E:/Documentos/Clases/proyecto/cartilla/app/src/activities/saltar/saltar.json

```
[
  {
    "id": "1",
    "title": "CONDUCIENDO EL TREN",
    "category": "movimiento",
    "keywords": "coordinación, equilibrio, lateralidad",
    "purpose": "Desarrollar en el niño procesos consciente de movimiento en diferentes direcciones",
    "description": "Se organiza un grupo de 4 a 6 niños (hermanos, otros niños, adultos o padres)",
    "materials": ["Espacio"],
    "variables": "Iniciar en un ritmo suave en donde el adulto decide en que dirección se dirige",
    "psicomotor": [
      {
        "title": "Motor",
        "description": "Gatear, caminar, coordinación, equilibrio, manejo temporoespacial, lateralidad",
      },
      {
        "title": "Cognitivo",
        "description": "Toma de decisiones (maquinista), atención y concentración (niños que se distraen)",
      },
      {
        "title": "Social",
        "description": "Cooperación y colaboración, trabajo en equipo, fortalecer los procesos de comunicación",
      }
    ],
    "extraInfo": [
      {
        "title": "Adaptación a espacios reducidos",
        "description": "Definir una ruta para que los niños se desplacen. Utilizar todas las habilidades",
      }
    ],
    "img": "/img/activities/conduciendoTren.avif"
  },
  {
    "id": "2",
    "title": "VOLANDO AL TIRO AL BLANCO",
    "category": "precisión",
    "keywords": "coordinación fina, lanzamiento",
    "purpose": "Estimular la manipulación de materiales para construcción de elementos de juego",
    "description": "Se elabora un avión de papel el cual será utilizado como elemento (dardo) para el juego",
    "materials": ["Hojas de papel", "un objeto que haga de tiro al blanco"],
    "variables": "El juego puede comenzar con un proceso de aprendizaje de plegado básico o con un objeto que haga de tiro al blanco",
    "psicomotor": [
      {
        "title": "Motor",

```

```

        "description": "Precisión, lanzamiento, coordinación fina y gruesa, manejo temporoes
    },
    {
        "title": "Cognitivo",
        "description": "Atención, concentración, memorización."
    },
    {
        "title": "Social",
        "description": "Cooperación y colaboración, trabajo en equipo, autoestima, persevera
    }
],
"extraInfo": [
    {
        "title": "Adaptación a espacios reducidos",
        "description": "Para esta actividad se puede utilizar cualquier lugar de la casa en
    }
],
"img": "/img/activities/paperplane.jpg"
},
{
    "id": "3",
    "title": "CAMINITO DE JUGUETES",
    "category": "coordinación",
    "keywords": "equilibrio, coordinación visomanual",
    "purpose": "Estimular los procesos de construcción por medio de la manipulación de eleme
    "description": "Utilizando elementos encontrados en casa, el niño, con la ayuda de sus h
    "materials": ["Espacio despejado", "juguetes", "ropa", "calzado", "cobijas"],
    "variables": "El camino puede estar provisto de diferentes distancias y pruebas de paso.
    "psycomotor": [
        {
            "title": "Motor",
            "description": "Precisión, coordinación fina y gruesa, coordinación visomanual, equi
        },
        {
            "title": "Cognitivo",
            "description": "Atención, concentración, creatividad, imaginación, exploración, clas
        },
        {
            "title": "Social",
            "description": "Cooperación y colaboración, trabajo en equipo, toma de decisiones, p
        }
    ],
    "extraInfo": [
        {
            "title": "Adaptación a espacios reducidos",
            "description": "Para esta actividad se puede utilizar todas las instalaciones de la

```

```

    }
  ],
  "img": "/img/activities/camino.webp"
},
{
  "id": "4",
  "title": "EL TOCA TOCA",
  "category": "coordinación",
  "keywords": "equilibrio, coordinación corporal",
  "purpose": "Conocimiento de las partes de mi cuerpo",
  "description": "Trabajando en parejas, se realizan los movimientos necesarios para que l",
  "materials": ["Espacio despejado"],
  "variables": "La actividad puede ser realizada por dos, tres o más niños. El nivel de co",
  "psicomotor": [
    {
      "title": "Motor",
      "description": "Coordinación, equilibrio."
    },
    {
      "title": "Cognitivo",
      "description": "Atención, concentración, creatividad, toma de decisiones."
    },
    {
      "title": "Social",
      "description": "Cooperación y colaboración, trabajo en equipo, perseverancia, lidera
    }
  ],
  "extraInfo": [
    {
      "title": "Adaptación a espacios reducidos",
      "description": "Para esta actividad se puede utilizar cualquier lugar de la casa en
    }
  ],
  "img": "/img/activities/tocatoca.jpg"
},
{
  "id": "5",
  "title": "ZAPATOS LOCOS",
  "category": "construcción",
  "keywords": "coordinación fina, equilibrio",
  "purpose": "Estimular los procesos de construcción de elementos.",
  "description": "Utilizando una cantidad de zapatos, los niños, en coordinación con un ac",
  "materials": ["Espacio", "zapatos"],
  "variables": "La actividad puede realizarse individualmente o en pequeños grupos. Otra a",
  "psicomotor": [
    {

```



```

        "title": "Motor",
        "description": "Coordinación fina, coordinación visomanual."
    },
    {
        "title": "Cognitivo",
        "description": "Atención, concentración, procesos creativos, toma de decisiones."
    },
    {
        "title": "Social",
        "description": "Interacción social, trabajo en equipo, colaboración, cooperación, as
    }
],
"extraInfo": [
    {
        "title": "Adaptación a espacios reducidos",
        "description": "Para esta actividad se puede realizar los en lugares de la casa con
    }
],
"img": "/img/activities/zapatos.webp"
},
{
    "id": "6",
    "title": "LAGO DE LAVA",
    "category": "imaginación",
    "keywords": "coordinación, equilibrio, creatividad",
    "purpose": "Estimular el desarrollo de los patrones básicos de movimientos, además de la
    "description": "El adulto o padre de familia propone una historia en la que los niños se
    "materials": ["Espacio", "hojas de papel", "sillas", "mesas", "cojines", "mantas", "ropa
    "variables": "La actividad puede permitir un reto en el que se formen grupos de trabajo
    "psicomotor": [
        {
            "title": "Motor",
            "description": "Correr, saltar, equilibrio, manejo tempo espacial, coordinación (gru
        },
        {
            "title": "Cognitivo",
            "description": "Atención, concentración, toma de decisiones, creatividad, imaginaci
        },
        {
            "title": "Social",
            "description": "Trabajo en equipo, colaboración, cooperación, interacción social."
        }
    ],
    "extraInfo": [
        {
            "title": "Adaptación a espacios reducidos",

```

```

        "description": "Para esta actividad se puede utilizar algunos espacios de la casa que",
    },
],
"img": "/img/activities/lava.jpg"
},
{
    "id": "7",
    "title": "SOPLANDO Y GOL",
    "category": "respiración",
    "keywords": "coordinación fina, capacidad pulmonar",
    "purpose": "Estimulación de los procesos de sensibilización interna (respiración).",
    "description": "En un espacio delimitado por una cinta con pegamento, se dibuja en el suelo",
    "materials": ["Espacio despejado", "cinta de enmascarar papel", "pelota liviana o de espuma"],
    "variables": "Defina el tamaño de la zona de juego (campo de fútbol) en función del espacio",
    "psicomotor": [
        {
            "title": "Motor",
            "description": "Coordinación fina, estimular la capacidad pulmonar, fortalecer los músculos",
        },
        {
            "title": "Cognitivo",
            "description": "Atención, concentración, creatividad, toma de decisiones."
        },
        {
            "title": "Social",
            "description": "Cooperación y colaboración, trabajo en equipo, liderazgo, fortalecer",
        }
    ],
    "extraInfo": [
        {
            "title": "Adaptación a espacios reducidos",
            "description": "Para esta actividad se puede utilizar cualquier lugar de la casa en el que",
        }
    ],
    "img": "/img/activities/soplando.jpg"
},
{
    "id": "8",
    "title": "TRES, DOS, UNO, DESPEGUE",
    "category": "respiración",
    "keywords": "coordinación fina, capacidad pulmonar",
    "purpose": "Estimulación de los procesos de sensibilización interna (respiración), y la",
    "description": "En un espacio delimitado se ancla una línea de cuerda (hilo, lana o piola)",
    "materials": ["Lana", "hilo o piola", "cinta de enmascarar", "bombas", "pitillos", "cartas"],
    "variables": "Definir el tamaño de la zona de juego donde se puede colocar una línea de",
    "psicomotor": [

```

```

    {
      "title": "Motor",
      "description": "Coordinación fina, estimular la capacidad pulmonar, fortalecer los m
    },
    {
      "title": "Cognitivo",
      "description": "Creatividad, toma de decisiones."
    },
    {
      "title": "Social",
      "description": "Cooperación y colaboración, liderazgo, asumir retos, trabajo en equi
    }
  ],
  "extraInfo": [
    {
      "title": "Adaptación a espacios reducidos",
      "description": "Para esta actividad se puede utilizar uno o varios lugares de la cas
    }
  ],
  "img": "/img/activities/globos.jpg"
},
{
  "id": "9",
  "title": "PASA CON CUIDADO",
  "category": "coordinación",
  "keywords": "equilibrio, manejo segmentario",
  "purpose": "Estimulación del sentido kinestésico, además de promover el sentido de lucha
  "description": "Esta actividad debe contar con un espacio que permita la creación de una
  "materials": ["Pasillo", "marco de una puerta", "Lana", "hilo", "cinta de enmascarar", "
  "variables": "Esta actividad puede realizarse de varias maneras, donde la construcción d
  "psicomotor": [
    {
      "title": "Motor",
      "description": "Coordinación gruesa, equilibrio, coordinación, manejo segmentario."
    },
    {
      "title": "Cognitivo",
      "description": "Atención, concentración, exploración, observación, toma de decisiones
    },
    {
      "title": "Social",
      "description": "Autoestima, cooperación, trabajo en equipo, solidaridad, control em
    }
  ],
  "extraInfo": [
    {

```

```

        "title": "Adaptación a espacios reducidos",
        "description": "Para esta actividad, utilizar un marco de puerta o pasillo de la casa",
      },
    ],
    "img": "/img/activities/red.jpg"
  }
]

```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/Loader.tsx

```

import { ClipLoader } from "react-spinners"

const Loader: React.FC = () => {
  return <div className="h-[100vh] flex flex-row justify-center content-center " >
    <ClipLoader size={60} />
  </div >
}

export default Loader;

```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/MainButton.tsx

```

import {
  IonButton,
  IonCard,
  IonCardContent,
  IonCardHeader,
  IonCardTitle,
} from "@ionic/react";

interface MainButtonProps {
  text: string;
  onClick: any;
  className?: string; // Añadir className a las props
}

const MainButton: React.FC<MainButtonProps> = ({
  text,
  onClick,
  className,
}) => {
  return (
    <IonButton
      className={` ${className} `} // Combinar las clases
      onClick={onClick}
    >

```

```

    >
    <p>{text}</p>
  </IonButton>
);
};

export default MainButton;

```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/activities/activities.css

```

@charset "UTF-8";
@import url("https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=block");

.activityBg {
  background-color: white; /* Morado oscuro */
  color: #fff;
  border-radius: 10px; /* Bordes redondeados para la tarjeta */
  margin-bottom: 20px;
}

.activityTitle {
  color: black; /* Morado claro */
  font-size: 12px; /* Tamaño de fuente aumentado */
  text-align: left;
  letter-spacing: 1.5px;
}

.activityBtn {
  --background: #c074f3; /* Morado medio */
  color: #55069e;
  --background-activated: #8e24aa; /* Morado intermedio más oscuro */
  --background-focused: #8e24aa;
  --background-hover: #8e24aa;
  --color: #fff; /* Blanco para el texto */
  border-radius: 15px;
  margin-top: 10px;
  transition: transform 0.2s, box-shadow 0.2s; /* Transición para hover */
  font-size: 1rem;
  font-weight: bold;
}

.activityBtn:hover {
  transform: translateY(-2px); /* Mover el botón hacia arriba en hover */
}

.activityBtn:active {
  transform: translateY(0); /* Restaurar posición en active */
}

```

```

}

.custom-activitiesbar {
  background: #e85d8b;
  font-size: 18px;
  font-family: "Poppins", sans-serif;
  font-style: normal;
  text-transform: uppercase;
  padding-top: 6px;
  padding-bottom: 6px;
  letter-spacing: 3px;
}

.custom-input {
  padding: 8px; /* Ajusta el padding interno */
  overflow: hidden; /* Oculta el contenido que se desborda */
}

.custom-segment-button {
  --color: #dc5097; /* Color del texto cuando está inactivo */
  --background: transparent; /* Fondo transparente cuando está inactivo */
}

ion-segment-button::part(indicator-background) {
  background: #dc5097;
}

ion-segment-button.md::part(native) {
  color: #000;
}

.segment-button-checked.md::part(native) {
  color: #dc5097;
}

ion-segment-button.md::part(indicator-background) {
  height: 4px;
}/*# sourceMappingURL=activities.css.map */

```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/activities/activities.css.map

```
{
  "version": 3,
  "sources": [
    "activities.css",
    "activities.scss"
  ],
  "names": [],
  "mappings": "AAAA,gBA"
}
```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/activities/activities.scss

```

@import url("https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap");

.activityBg {
    background-color: white; /* Morado oscuro */
    color: #fff;
    border-radius: 10px; /* Bordes redondeados para la tarjeta */
    margin-bottom: 20px;
}

.activityTitle {
    color: black; /* Morado claro */
    font-size: 12px; /* Tamaño de fuente aumentado */
    text-align: left;
    letter-spacing: 1.5px;
}

.activityBtn {
    --background: #c074f3; /* Morado medio */
    color: #55069e;
    --background-activated: #8E24AA; /* Morado intermedio más oscuro */
    --background-focused: #8E24AA;
    --background-hover: #8E24AA;
    --color: #fff; /* Blanco para el texto */
    border-radius: 15px;
    margin-top: 10px;
    transition: transform 0.2s, box-shadow 0.2s; /* Transición para hover */
    font-size: 1rem;
    font-weight: bold;
}

.activityBtn:hover {
    transform: translateY(-2px); /* Mover el botón hacia arriba en hover */
}

.activityBtn:active {
    transform: translateY(0); /* Restaurar posición en active */
}

.custom-activitiesbar{
    background: #e85d8b;
    font-size: 18px;
    font-family: "Poppins", sans-serif;
    font-style: normal;
    text-transform: uppercase;
    padding-top: 6px;
}

```

```

        padding-bottom: 6px;
        letter-spacing: 3px;
    }

    .custom-input {
        padding: 8px; /* Ajusta el padding interno */
        overflow: hidden; /* Oculta el contenido que se desborda */
    }

    .custom-segment-button {
        --color: #dc5097; /* Color del texto cuando está inactivo */
        --background: transparent; /* Fondo transparente cuando está inactivo */
    }

    ion-segment-button::part(indicator-background) {
        background: #dc5097;
    }

    ion-segment-button.md::part(native) {
        color: #000;
    }

    .segment-button-checked.md::part(native) {
        color: #dc5097;
    }

    ion-segment-button.md::part(indicator-background) {
        height: 4px;
    }

```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/activities/Activities.tsx

```

import {
    IonBackButton,
    IonButtons,
    IonCard,
    IonCardContent,
    IonCardHeader,
    IonCardTitle,
    IonChip,
    IonCol,
    IonContent,
    IonGrid,
    IonHeader,
    IonLabel,
    IonPage,

```



```

    IonRow,
    IonSegment,
    IonSegmentButton,
    IonTitle,
    IonToolbar,
  } from "@ionic/react";
import { useEffect, useState } from "react";
import { fetchDataAndSearchById } from "../../utils/loadActivity";
import "../activities.css";
import ActivitiesContent from "../ActivitiesContent";
import { ClipLoader } from "react-spinners";
import { Loader } from "rsuite";

interface ActivitiesProps {
  match: any;
}

interface Psicomotor {
  title: string;
  description: string;
}

interface Activity {
  id: string;
  title: string;
  category: string;
  keywords: string;
  purpose: string;
  description: string;
  materials: string[];
  variables: string;
  psicomotor: Psicomotor[];
  extraInfo: string;
}

const initialActivity: Activity = {
  id: "1",
  title: "Saltar",
  category: "Saltar",
  keywords: "Saltar",
  purpose: "Saltar",
  description: "Saltar",
  materials: ["cuerda"],
  variables: "variables",
  psicomotor: [

```

```

    {
      title: "Motor",
      description: "Motor",
    },
    {
      title: "Cognitivo",
      description: "Cognitivo",
    },
    {
      title: "Social",
      description: "Social",
    },
  ],
  extraInfo: "",
};

const Activities: React.FC<ActivitiesProps> = ({ match }) => {
  const [activity, setActivity] = useState(initialActivity);
  const [showPurpose, setShowPurpose] = useState("activity");
  const [showLoader, setShowLoader] = useState(false);

  useEffect(() => {
    const get = async () => {
      setShowLoader(true);
      let activity: any = await fetch(`https://services.qalaub.com/api/ipapp?id=${match.params.id}`);
      let newActivity = await activity.json();
      newActivity.materials = newActivity.materials.split(',');
      setActivity(newActivity);
      setShowLoader(false);
    };
    get();
  }, []);

  const handleSegment = (ev: any) => {
    setShowPurpose(ev.detail.value);
  };

  return (
    <IonPage>
      <IonHeader>
        <IonToolbar className="custom-toolbar">
          <IonButtons slot="start">
            <IonBackButton default-href="/"></IonBackButton>
          </IonButtons>
          <IonTitle>Descripción</IonTitle>
        </IonToolbar>
      </IonHeader>

```

```

<IonContent>
  {showLoader && < div className="h-[100vh] flex flex-row justify-center content-center" >
    <ClipLoader size={60} />
  </div >}
  {!showLoader &&
    <>
      <IonHeader>
        <IonToolbar>
          <IonSegment value={showPurpose} onIonChange={handleSegment}>
            <IonSegmentButton
              className="custom-segment-button"
              value="activity"
            >
              <IonLabel>Actividad</IonLabel>
            </IonSegmentButton>
          </IonSegment>
        </IonToolbar>
      </IonHeader>
      <div className="bodybgs min-h-full">
        {showPurpose == "activity" ? (
          <section >
            <h1 className="text-center mt-5 font-bold bg-white rounded-lg py-2.5 w-[90%]" >
              {activity.title}
            </h1>
            <ActivitiesContent
              title="Propósito"
              description={activity.purpose}
            />
            <ActivitiesContent
              title="Descripción"
              description={activity.description}
            />
            <div>
              <IonCard>
                <IonCardHeader>
                  <IonCardTitle className="font-bold">Materiales requeridos</IonCardTitle>
                </IonCardHeader>
                <IonCardContent>
                  {activity.materials?.map((el: string) => (
                    <IonChip key={el} outline={true}>
                      {el}
                    </IonChip>
                  ))}
                </IonCardContent>
              </IonCard>
            </div>
          ) : null}
        </div>
      </div>
    </>
  }

```

```

        <ActivitiesContent
          title="Variables"
          description={activity.variables}
        />
      </section>
    ) : (
      <section className="min-h-full justific">
        <h2 className="text-center mt-5 font-bold bg-white rounded-lg py-2.5 w-[90%]">
          {activity.psycomotor?.map((el: any) => (
            <IonCard key={el.title}>
              <IonCardHeader>
                <IonCardTitle className="font-bold">{el.title}</IonCardTitle>
              </IonCardHeader>
              <IonCardContent>{el.description}</IonCardContent>
            </IonCard>
          ))}
        </section>
      )}
    </div>
  </>
</IonContent>
</IonPage>
);
};

```

```
export default Activities;
```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/activities/ActivitiesContent.tsx

```

import {
  IonCard,
  IonCardContent,
  IonCardHeader,
  IonCardTitle,
} from "@ionic/react";
import "../activities.css";

interface ActivitiesContentProps {
  title: string;
  description: string;
}

const ActivitiesContent: React.FC<ActivitiesContentProps> = ({
  title,
  description,

```

```

    }) => {
      return (
        <>
          <IonCard>
            <IonCardHeader className="prueba">
              <IonCardTitle className="font-bold">{title}</IonCardTitle>
            </IonCardHeader>
            <IonCardContent className="text-black">{description}</IonCardContent>
          </IonCard>
        </>
      );
    };
  };

```

```
export default ActivitiesContent;
```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/activities/Activity.tsx

```

import {
  IonButton,
  IonCard,
  IonCardContent,
  IonCardHeader,
  IonCardTitle,
} from "@ionic/react";
import "./activities.css";
import { ActivityType } from "../../utils/types";
import { useHistory } from "react-router";
import { saveUserData } from "../../utils/loadActivity";
import { useContext } from "react";
import { UserContext } from "../../context/UserContext";

interface ActivityProps {
  activity: ActivityType;
}

const Activity: React.FC<ActivityProps> = ({ activity }) => {
  const { update, setUpdate } = useContext(UserContext);
  const history = useHistory();

  const handleClick = () => {
    // Crear un nuevo objeto JSON con los datos de actividad
    const updatedJson = {
      category: activity.category,
      id: activity.id,
    };
    saveUserData(updatedJson);
  };

```

```

        // Redirigir a la nueva ruta
        setUpDate(!update);
        history.push(`/activities/${activity.category}/${activity.id}`);
    };
    return (
        <IonCard
            title={activity.title}
            className="activityBg flex flex-row h-56"
            onClick={handleClick}
        >
        <div className="w-[50%] overflow-hidden">
            <img
                src={activity.img.replace('data:image', 'data:image').replace('base64', ';base64,')}
                alt=""
                className="w-full h-full object-cover"
            />
        </div>
        <div className="flex flex-col w-[50%]">
            <IonCardHeader>
                <IonCardTitle className="activityTitle font-bold">
                    {activity.title}
                </IonCardTitle>
            </IonCardHeader>
            <IonCardContent className="flex flex-col">
                <p className="h-36 overflow-hidden overflow-y-scroll text-black text-justify">
                    {activity.description}
                </p>
            </IonCardContent>
        </div>
    </IonCard>
    );
};

export default Activity;

```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/activities/AllActivities.tsx

```

import {
    IonButton,
    IonCard,
    IonCardContent,
    IonCardHeader,
    IonCardTitle,
    IonContent,
    IonHeader,
    IonPage,

```

```

    IonTitle,
    useIonRouter,
  } from "@ionic/react";
import { useEffect, useState } from "react";
import { fetchAllData } from "../../utils/loadActivity";
import "./activities.css";
import Activity from "./Activity";
import { Loader } from "rsuite";
import { ClipLoader } from "react-spinners";

interface ActivitiesProps {
  match: any;
}

const AllActivities: React.FC<ActivitiesProps> = ({ match }) => {
  const [activities, setActivities] = useState([]);
  const router = useIonRouter();
  const [showLoader, setShowLoader] = useState(false);

  useEffect(() => {
    (async () => {
      setShowLoader(true);
      const subcategory = match.params.subcategory;
      let url = 'https://services.qalaub.com/api/ipapp';
      if (subcategory) url = 'https://services.qalaub.com/api/ipapp/subcategory?subcategory=' + subcategory;
      const res = await fetch(url);
      setActivities(await res.json());
      setShowLoader(false);
    })();
  }, [router.routeInfo]);

  return (
    <IonPage>
      <IonHeader className="w-full h-7">
        <IonTitle class="custom-activitiesbar ion-text-center text-white">
          Actividades
        </IonTitle>
      </IonHeader>

      <IonContent>
        {showLoader && <div className="h-[100vh] flex flex-row justify-center content-center">
          <ClipLoader size={60} />
        </div >}
        {!showLoader && <section className="bodybgs min-h-full">
          {activities?.map((el: any, i: number) => (
            <Activity key={el.title + i} activity={el} />
          ))}
        </section>}
      </IonContent>
    </IonPage>
  );
};

```

```

    ))}
  </section>}

  </IonContent>
</IonPage>
);
};

```

```
export default AllActivities;
```

E:/Documentos/Clases/proyecto/cartilla/app/src/components/context/UserContext.tsx

```

import React, { createContext, useState, ReactNode } from 'react';

// Define los tipos para el contexto
interface UserContextType {
  update: boolean;
  setUpdate: (update: boolean) => void;
}

// Crea el contexto con un valor inicial indefinido
const UserContext = createContext<UserContextType>({} as UserContextType);

// Define el tipo de los props para el proveedor
interface UserProviderProps {
  children: ReactNode;
}

// Define el proveedor del contexto
const UserProvider: React.FC<UserProviderProps> = ({ children }) => {
  const [update, setUpdate] = useState<boolean>(false);

  const data: UserContextType = {
    update, setUpdate
  };

  return (
    <UserContext.Provider value={data}>
      {children}
    </UserContext.Provider>
  );
}

export { UserProvider, UserContext };

```


E:/Documentos/Clases/proyecto/cartilla/app/src/pages/AddActivity/AddActivity.tsx

```
import React, { useState, ChangeEvent, FormEvent } from "react";
import {
  IonInput,
  IonButton,
  IonTextarea,
  IonItem,
  IonLabel,
  IonList,
  IonListHeader,
  IonPage,
  IonContent,
  IonSelect,
  IonSelectOption,
  IonHeader,
  IonTitle,
  IonIcon,
} from "@ionic/react";
import { arrowForward, chevronDown, chevronUp, personOutline } from "ionicons/icons";
import niña from "../../assets/Img/girl.webp";

interface Psicomotor {
  title: string;
  description: string;
}

interface ExtraInfo {
  title: string;
  description: string;
}

interface Activity {
  title: string;
  category: string;
  subcategory: string; // Nueva línea para subcategoría
  keywords: string;
  purpose: string;
  description: string;
  materials: string[];
  variables: string;
  img: string | ArrayBuffer | null;
  psicomotor: Psicomotor[];
  extraInfo: ExtraInfo[];
}
```

```

const customAlertOptions = {
  header: "Codigo de acceso",
  subHeader: "Ingresa el codigo",
  translucent: true,
};

const ActivityForm: React.FC = () => {
  const [activity, setActivity] = useState<Activity>({
    title: "",
    category: "",
    subcategory: "", // Nueva línea para subcategoría
    keywords: "",
    purpose: "",
    description: "",
    materials: [""],
    variables: "",
    img: null,
    psycomotor: [{ title: "", description: "" }],
    extraInfo: [{ title: "", description: "" }],
  });
  const [selectedOption, setSelectedOption] = useState<string | undefined>();
  const [codeA, setCodeA] = useState<string | undefined>(undefined);
  const [pass, setPass] = useState<boolean>(false);

  const handleChange = (
    e: ChangeEvent<
      HTMLInputElement | HTMLTextAreaElement | HTMLIonSelectElement
    >
  ) => {
    const { name, value } = e.target;
    setActivity({ ...activity, [name]: value });
  };

  const handleImageChange = (e: ChangeEvent<HTMLInputElement>) => {
    const file = e.target.files?.[0];
    if (file) {
      const reader = new FileReader();
      reader.onloadend = () => {
        setActivity({ ...activity, img: reader.result });
      };
      reader.readAsDataURL(file);
    }
  };

  const handleArrayChange = (
    index: number,

```

```

        field: "title" | "description",
        value: string,
        type: "psicomotor" | "extraInfo"
    ) => {
        const newArray = [...activity[type]] as (Psicomotor | ExtraInfo)[];
        newArray[index] = { ...newArray[index], [field]: value };
        setActivity({ ...activity, [type]: newArray });
    };

    const handleMaterialChange = (index: number, value: string) => {
        const newMaterials = [...activity.materials];
        newMaterials[index] = value;
        setActivity({ ...activity, materials: newMaterials });
    };

    const addMaterial = () => {
        setActivity({ ...activity, materials: [...activity.materials, ""] });
    };

    const removeMaterial = (index: number) => {
        const newMaterials = [...activity.materials];
        newMaterials.splice(index, 1);
        setActivity({ ...activity, materials: newMaterials });
    };

    const addPsicomotor = () => {
        setActivity({
            ...activity,
            psicomotor: [...activity.psicomotor, { title: "", description: "" }],
        });
    };

    const removePsicomotor = (index: number) => {
        const newPsicomotor = [...activity.psicomotor];
        newPsicomotor.splice(index, 1);
        setActivity({ ...activity, psicomotor: newPsicomotor });
    };

    const addExtraInfo = () => {
        setActivity({
            ...activity,
            extraInfo: [...activity.extraInfo, { title: "", description: "" }],
        });
    };

    const removeExtraInfo = (index: number) => {

```

```

const newExtraInfo = [...activity.extraInfo];
newExtraInfo.splice(index, 1);
setActivity({ ...activity, extraInfo: newExtraInfo });
};

const handleSubmit = async (e: FormEvent) => {
  e.preventDefault();
  try {
    const response = await fetch("https://services.qalaub.com/api/ipapp", {
      method: "POST",
      headers: {
        "Content-Type": "application/json",
      },
      body: JSON.stringify(activity),
    });
    const data = await response.json();
    console.log("Success:", data);
  } catch (error) {
    console.error("Error:", error);
  }
};

const handleContinue = async () => {
  if(codeA == 'AMDIEPCNOWQSM') setPass(true);
};

return (
  <IonPage>
    {pass ?
      <>
        <IonHeader className="w-full h-7">
          <IonTitle class="custom-activitiesbar ion-text-center text-white">
            Agregar actividad
          </IonTitle>
        </IonHeader>
        <IonContent className="bodybgxs">
          <section className="bodybgs min-h-[100vh]">
            <form
              onSubmit={handleSubmit}
              className="bg-white overflow-scroll h-[90%] w-[90%] my-4 rounded-xl border-
            >
              <IonItem className="border-b-[0.5px] flex flex-row p-3">
                <IonLabel position="stacked">Título</IonLabel>
                <IonInput
                  style={{
                    "--padding-start": "10px", // Ajusta el padding desde el inicio

```

```

        "--padding-end": "10px", // Ajusta el padding desde el final
        "--padding-top": "10px", // Ajusta el padding desde arriba
        "--padding-bottom": "10px", // Ajusta el padding desde abajo
        overflow: "hidden", // Oculta el contenido que se desborda
    }}
    className="w-full border-[#E65B8F] border-2 rounded-md my-1.5 custom-inp
    name="title"
    value={activity.title}
    onChange={(e: any) => handleChange(e)}
    placeholder="Ingrese el título"
  />
</IonItem>
<IonItem className="border-b-[0.5px] flex flex-row p-3">
  <IonLabel position="stacked">Categoría</IonLabel>
  <IonSelect
    style={{
      "--padding-start": "10px", // Ajusta el padding desde el inicio
      "--padding-end": "10px", // Ajusta el padding desde el final
      "--padding-top": "10px", // Ajusta el padding desde arriba
      "--padding-bottom": "10px", // Ajusta el padding desde abajo
      overflow: "hidden", // Oculta el contenido que se desborda
    }}
    className="w-full border-[#E65B8F] border-2 rounded-md mt-1.5"
    name="category"
    value={activity.category}
    onChange={(e: any) => handleChange(e)}
    placeholder="Seleccione una categoría"
  >
    <IonSelectOption value="motoras">Motoras</IonSelectOption>
    <IonSelectOption value="pelota">Con Pelota</IonSelectOption>
    { /* Agrega más opciones según sea necesario */ }
  </IonSelect>
</IonItem>
<IonItem className="border-b-[0.5px] flex flex-row p-3">
  <IonLabel position="stacked">Subcategoría</IonLabel>
  <IonSelect
    style={{
      "--padding-start": "10px", // Ajusta el padding desde el inicio
      "--padding-end": "10px", // Ajusta el padding desde el final
      "--padding-top": "10px", // Ajusta el padding desde arriba
      "--padding-bottom": "10px", // Ajusta el padding desde abajo
      overflow: "hidden", // Oculta el contenido que se desborda
    }}
    className="w-full border-[#E65B8F] border-2 rounded-md mt-1.5"
    name="subcategory"
    value={activity.subcategory}
  >

```

```

        onIonChange={(e: any) => handleChange(e)}
        placeholder="Seleccione una subcategoría"
    >
        <IonSelectOption value="desplazamiento">
            Desplazamiento
        </IonSelectOption>
        <IonSelectOption value="salto">Salto</IonSelectOption>
        <IonSelectOption value="pateo">Pateo</IonSelectOption>
        <IonSelectOption value="lanzar">Lanzar</IonSelectOption>
        <IonSelectOption value="recibir">Recibir</IonSelectOption>
        <IonSelectOption value="rebotar">Rebotar</IonSelectOption>
        <IonSelectOption value="golpear">Golpear</IonSelectOption>
        <IonSelectOption value="tiro">Tiro</IonSelectOption>
        <IonSelectOption value="agarrar">Agarrar</IonSelectOption>
    </IonSelect>
</IonItem>
<IonItem className="border-b-[0.5px] flex flex-row p-3">
    <IonLabel position="stacked">Palabras clave</IonLabel>
    <IonTextarea
        style={{
            '--padding-start': '10px', // Ajusta el padding desde el inicio
            '--padding-end': '10px',   // Ajusta el padding desde el final
            '--padding-top': '10px',   // Ajusta el padding desde arriba
            '--padding-bottom': '10px', // Ajusta el padding desde abajo
            'overflow': 'hidden',       // Oculta el contenido que se desborda
        }}
        className="w-full border-[#E65B8F] border-2 rounded-md my-1.5"
        name="keywords"
        value={activity.keywords}
        onIonChange={(e: any) => handleChange(e)}
        placeholder="Ingrese las palabras clave"
    />
</IonItem>
<IonItem className="border-b-[0.5px] flex flex-row p-3">
    <IonLabel position="stacked">Propósito</IonLabel>
    <IonTextarea
        style={{
            '--padding-start': '10px', // Ajusta el padding desde el inicio
            '--padding-end': '10px',   // Ajusta el padding desde el final
            '--padding-top': '10px',   // Ajusta el padding desde arriba
            '--padding-bottom': '10px', // Ajusta el padding desde abajo
            'overflow': 'hidden',       // Oculta el contenido que se desborda
        }}
        className="w-full border-[#E65B8F] border-2 rounded-md my-1.5"
        name="purpose"
        value={activity.purpose}
    >

```

```

        onIonChange={(e: any) => handleChange(e)}
        placeholder="Ingrese el propósito"
    />
</IonItem>
<IonItem className="border-b-[0.5px] flex flex-row p-3">
    <IonLabel position="stacked">Descripción</IonLabel>
    <IonTextarea
        style={{
            '--padding-start': '10px', // Ajusta el padding desde el inicio
            '--padding-end': '10px',   // Ajusta el padding desde el final
            '--padding-top': '10px',   // Ajusta el padding desde arriba
            '--padding-bottom': '10px', // Ajusta el padding desde abajo
            'overflow': 'hidden',       // Oculta el contenido que se desborda
        }}
        className="w-full border-[#E65B8F] border-2 rounded-md my-1.5"
        name="description"
        value={activity.description}
        onIonChange={(e: any) => handleChange(e)}
        placeholder="Ingrese la descripción"
    />
</IonItem>
<IonList className="border-b-[0.5px] flex flex-col p-3">
    <IonListHeader>
        <IonLabel>Materiales</IonLabel>
    </IonListHeader>
    {activity.materials.map((material, index) => (
        <IonItem key={index}>
            <IonInput
                placeholder={`Material ${index + 1}`}
                value={material}
                onIonChange={(e: CustomEvent) =>
                    handleMaterialChange(
                        index,
                        (e.target as HTMLInputElement).value
                    )
                }
            />
            <IonButton
                style={{ "--background": "#E65B8F", "--color": "#fff" }}
                onClick={() => removeMaterial(index)}
            >
                Eliminar
            </IonButton>
        </IonItem>
    ))}
</IonList>
<IonButton

```

```

        style={{ "--background": "#E65B8F", "--color": "#fff" }}
        onClick={addMaterial}
    >
        Agregar Material
    </IonButton>
</IonList>
<IonItem className="border-b-[0.5px] flex flex-row p-3">
    <IonLabel position="stacked">Variables</IonLabel>
    <IonTextarea
        style={{
            '--padding-start': '10px', // Ajusta el padding desde el inicio
            '--padding-end': '10px',   // Ajusta el padding desde el final
            '--padding-top': '10px',   // Ajusta el padding desde arriba
            '--padding-bottom': '10px', // Ajusta el padding desde abajo
            'overflow': 'hidden',       // Oculta el contenido que se desborda
        }}
        className="w-full h-[15vh] border-[#E65B8F] border-2 rounded-md my-1.5"
        name="variables"
        value={activity.variables}
        onIonChange={(e: any) => handleChange(e)}
        placeholder="Ingrese las variables"
    />
</IonItem>
<IonItem className="border-b-[0.5px] flex flex-row p-3">
    <IonLabel position="stacked">Imagen</IonLabel>
    <input
        className="w-full rounded-md my-1.5 flex flex-col"
        type="file"
        onChange={handleImageChange}
    />
</IonItem>
<IonList className="border-b-[0.5px] flex flex-col p-3">
    <IonListHeader>
        <IonLabel>Psicomotor</IonLabel>
    </IonListHeader>
    {activity.psycomotor.map((item, index) => (
        <IonItem key={index}>
            <IonLabel position="stacked">Titulo</IonLabel>
            <IonInput
                style={{
                    '--padding-start': '10px', // Ajusta el padding desde el inicio
                    '--padding-end': '10px',   // Ajusta el padding desde el final
                    '--padding-top': '10px',   // Ajusta el padding desde arriba
                    '--padding-bottom': '10px', // Ajusta el padding desde abajo
                    'overflow': 'hidden',       // Oculta el contenido que se desborda
                }}
            />
        </IonItem>
    ))
}

```



```

        className="w-full border-[#E65B8F] border-2 rounded-md my-1.5"
        placeholder="Título"
        value={item.title}
        onChange={(e: CustomEvent) =>
            handleArrayChange(
                index,
                "title",
                (e.target as HTMLInputElement).value,
                "psicomotor"
            )
        }
    />
<IonLabel position="stacked">Descripción</IonLabel>
<IonTextarea
    style={{
        '--padding-start': '10px', // Ajusta el padding desde el inicio
        '--padding-end': '10px',   // Ajusta el padding desde el final
        '--padding-top': '10px',   // Ajusta el padding desde arriba
        '--padding-bottom': '10px', // Ajusta el padding desde abajo
        'overflow': 'hidden',      // Oculta el contenido que se desborda
    }}
    className="w-full border-[#E65B8F] border-2 rounded-md my-1.5"
    placeholder="Descripción"
    value={item.description}
    onChange={(e: CustomEvent) =>
        handleArrayChange(
            index,
            "description",
            (e.target as HTMLTextAreaElement).value,
            "psicomotor"
        )
    }
/>
<IonButton
    style={{ "--background": "#E65B8F", "--color": "#fff" }}
    onClick={() => removePsicomotor(index)}
>
    Eliminar
</IonButton>
</IonItem>
)}}
<IonButton
    style={{ "--background": "#E65B8F", "--color": "#fff" }}
    onClick={addPsicomotor}
>
    Agregar Psicomotor

```

```

</IonButton>
</IonList>
<IonList className="border-b-[0.5px] flex flex-col p-3">
  <IonListHeader>
    <IonLabel>Información adicional</IonLabel>
  </IonListHeader>
  {activity.extraInfo.map((item, index) => (
    <IonItem key={index}>
      <IonLabel position="stacked">Título</IonLabel>
      <IonInput
        style={{
          '--padding-start': '10px', // Ajusta el padding desde el inicio
          '--padding-end': '10px',   // Ajusta el padding desde el final
          '--padding-top': '10px',   // Ajusta el padding desde arriba
          '--padding-bottom': '10px', // Ajusta el padding desde abajo
          'overflow': 'hidden',      // Oculta el contenido que se desborda
        }}
        className="w-full border-[#E65B8F] border-2 rounded-md my-1.5"
        placeholder="Título"
        value={item.title}
        onIonChange={(e: CustomEvent) =>
          handleArrayChange(
            index,
            "title",
            (e.target as HTMLInputElement).value,
            "extraInfo"
          )
        }
      />
      <IonLabel position="stacked">Descripción</IonLabel>
      <IonTextarea
        style={{
          '--padding-start': '10px', // Ajusta el padding desde el inicio
          '--padding-end': '10px',   // Ajusta el padding desde el final
          '--padding-top': '10px',   // Ajusta el padding desde arriba
          '--padding-bottom': '10px', // Ajusta el padding desde abajo
          'overflow': 'hidden',      // Oculta el contenido que se desborda
        }}
        className="w-full border-[#E65B8F] border-2 rounded-md my-1.5"
        placeholder="Descripción"
        value={item.description}
        onIonChange={(e: CustomEvent) =>
          handleArrayChange(
            index,
            "description",
            (e.target as HTMLTextAreaElement).value,

```

```

        "extraInfo"
      )
    }
  />
  <IonButton
    style={{ "--background": "#E65B8F", "--color": "#fff" }}
    onClick={() => removeExtraInfo(index)}
  >
    Eliminar
  </IonButton>
</IonItem>
)}}
<IonButton
  style={{ "--background": "#E65B8F", "--color": "#fff" }}
  onClick={addExtraInfo}
>
  Agregar Información Adicional
</IonButton>
</IonList>
<div className="border-b-[0.5px] flex flex-col p-3">
  <IonButton
    className="w-full"
    style={{ "--background": "#E65B8F", "--color": "#fff" }}
    type="submit"
  >
    Enviar
  </IonButton>
</div>
</form>
</section>
</IonContent>
</> :
<>
  <IonContent>
    <div className="select_roll_container">
      <div className="w-full flex flex-col content-center items-center justify-cente
        <h1 className="poppins text-white font-extrabold text-3xl mb-5">Acceso para
        <img src={niña} className="w-[60%] self-center" />
      </div>
      <div className="relative bg-white w-[100%] h-[200px] p-6 rounded-3xl edit_shad
        <div className="w-[95%]">
          <IonItem className="border-b-[0.5px] flex flex-row p-3">
            <IonInput
              style={{
                "--padding-start": "10px", // Ajusta el padding desde el inicio
                "--padding-end": "10px", // Ajusta el padding desde el final

```

```

        "--padding-top": "10px", // Ajusta el padding desde arriba
        "--padding-bottom": "10px", // Ajusta el padding desde abajo
        overflow: "hidden", // Oculta el contenido que se desborda
    }}
    className="w-full border-[#E65B8F] border-2 rounded-md my-1.5 custom-1
    name="codigo"
    value={codeA}
    onChange={(e: any) => setCodeA(e.target.value)}
    placeholder="Ingresa el código"
  />
</IonItem>
</div>
<div className="rounded-full absolute -bottom-8 overflow-hidden p-0.5 bg-white" >
  <IonButton
    onClick={handleContinue}
    disabled={!codeA}
    shape="round"
    size="large"
  >
    <IonIcon
      slot="icon-only"
      icon={arrowForward}
      aria-hidden="true"
      color="light"
    ></IonIcon>
  </IonButton>
</div>
</div>
</div>
</IonContent>
</>
</IonPage>
);
};

```

```
export default ActivityForm;
```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/main/main.css

```

@charset "UTF-8";
@import url("https://fonts.googleapis.com/css2?family=Lilita+One&display=swap");
@import url("https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap");
@import url("https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=swap");
.bgcolor {
  --background: #f5f5f5;
}

```

```

.custom-toolbar {
  --background: #e85d8b;
}

.lilita-one-regular {
  font-family: "Lilita One", sans-serif;
  font-weight: 400;
  font-style: normal;
  color: white;
  font-size: xx-large;
}

.why-us {
  position: relative;
  overflow: hidden;
  width: 100%;
  min-height: 22%;
  padding: 15px 15px;
}

.why-us h1 {
  font-size: 20px;
  font-family: "Poppins", sans-serif;
  font-style: normal;
}

.why-us p {
  font-size: 13px;
  font-family: "Roboto", sans-serif;
  font-weight: 400;
  font-style: normal;
  color: white;
  margin-top: 10px;
  display: -webkit-box;
  -webkit-box-orient: vertical;
  -webkit-line-clamp: 3; /* Número de líneas a mostrar antes de truncar */
  overflow: hidden;
  text-overflow: ellipsis;
}

.why-us .custom-home-button {
  --background: #d05281;
  font-family: "Roboto", sans-serif;
  font-weight: 500;
  font-style: normal;
  color: white;
  margin-top: 15px;
}

```

```

.why-us .custom-home-button p {
    font-size: 12px;
    margin: 0;
    padding-left: 20px;
    padding-right: 20px;
}

.why-us::before {
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: url("../../public/img/Home/pexels-shvetsa-3845220.jpg") no-repeat center center;
    background-size: cover;
    filter: blur(2px);
    z-index: -1;
}

.why-us::after {
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: rgba(0, 0, 0, 0.3); /* Ajusta la opacidad para cambiar el nivel de oscurecimiento */
    z-index: -1;
}

.why-us > * {
    position: relative;
    z-index: 1;
}

.buttontruncate {
    overflow: visible;
    margin: 0 10px;
    white-space: nowrap; /* Evita que el texto se rompa en varias líneas */
    overflow: hidden; /* Oculta el desbordamiento del texto */
    text-overflow: ellipsis; /* Añade puntos suspensivos */
    --background: #d05281;
    font-family: "Roboto", sans-serif;
    font-weight: 600;
    font-style: normal;

```

```

        color: white;
        /* main.css */
    }
    .buttontruncate p {
        font-size: 15px;
        margin: 0;
        height: 44px;
        width: 155px;
        display: flex; /* Hacer que el contenedor sea un flex container */
        justify-content: center; /* Centrar horizontalmente */
        align-items: center; /* Centrar verticalmente */
        text-align: center; /* Asegurar que el texto esté centrado */
        padding-left: 10px;
        padding-right: 10px;
    }

    .slidersections {
        background-color: #ffffff;
        padding: 20px 0px;
    }
    .slidersections h2 {
        font-family: "Poppins", sans-serif;
        font-style: normal;
        margin: 20px;
        margin-top: 0;
        font-size: 16px;
        font-weight: bold;
    }

    .populars {
        display: flex;
    }
    .populars h2 {
        margin-top: 20px;
    }

    .select_roll_container {
        --background: none;
        background: linear-gradient(to bottom, #c624ff 20%, #ff864e 100%), linear-gradient(to bottom, #c624ff 20%, #ff864e 100%);
        background-size: 100% 50%, 100% 50%;
        background-position: top, bottom;
        background-repeat: no-repeat;
        min-height: 100vh;
        display: flex;
        flex-direction: column;
        justify-content: center;
    }

```

```

    align-items: center;
    padding: 20px;
}

.select_roll_container ion-select {
    width: 100%;
    max-width: 400px;
}

.select_roll_container ion-button {
    --background: none;
    background: linear-gradient(to bottom, #c624ff -100%, #ff864e 100%);
    border-radius: 100%;
}

.edit_shadow {
    box-shadow: 0 0px 6px 2px rgba(0, 0, 0, 0.4);
}

.edit_shadow2 {
    box-shadow: 0 10px 5px -1px rgba(0, 0, 0, 0.4);
}

.no-focus-indicator ion-select {
    --background: none;
    --box-shadow: none;
    --outline: none;
}

/* También se puede aplicar a los elementos internos */
.no-focus-indicator ion-select .select-text {
    box-shadow: none;
}

.no-focus-indicator ion-select .select-icon {
    display: none;
}

.poppins {
    font-family: "Poppins", sans-serif;
}

.sw-mod .swiper-autoheight, .sw-mod .swiper-autoheight .swiper-slide {
    height: 80px;
}

.sw-mod .swiper-scrollbar {

```



```

    z-index: 10;
}
.sw-mod .swiper-scrollbar-drag {
    --swiper-scrollbar-drag-bg-color: rgb(232, 93, 139);
}/*# sourceMappingURL=main.css.map */

```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/main/main.css.map

```

{"version":3,"sources":["main.css","main.scss"],"names":[],"mappings":"AAAA,gBAAGB;ACAR,+EA

```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/main/main.scss

```

@import url("https://fonts.googleapis.com/css2?family=Lilita+One&display=swap");
@import url("https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap");
@import url("https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=swap");
.bgcolor {
    --background: #f5f5f5;
}

.custom-toolbar {
    --background: #e85d8b;
}

.lilita-one-regular {
    font-family: "Lilita One", sans-serif;
    font-weight: 400;
    font-style: normal;
    color: white;
    font-size: xx-large;
}

.why-us {
    position: relative;
    overflow: hidden;
    width: 100%;
    min-height: 22%;
    padding: 15px 15px;
    h1 {
        font-size: 20px;
        font-family: "Poppins", sans-serif;
        font-style: normal;
    }
    p {
        font-size: 13px;
        font-family: "Roboto", sans-serif;
        font-weight: 400;
    }
}

```

```

    font-style: normal;
    color: white;
    margin-top: 10px;
    display: -webkit-box;
    -webkit-box-orient: vertical;
    -webkit-line-clamp: 3; /* Número de líneas a mostrar antes de truncar */
    overflow: hidden;
    text-overflow: ellipsis;
}

.custom-home-button {
    --background: #d05281;
    font-family: "Roboto", sans-serif;
    font-weight: 500;
    font-style: normal;
    color: white;
    margin-top: 15px;
    p {
        font-size: 12px;
        margin: 0;
        padding-left: 20px;
        padding-right: 20px;
    }
}

}

}

.why-us::before {
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: url("../../public/img/Home/pexels-shvetsa-3845220.jpg") no-repeat center cover;
    background-size: cover;
    filter: blur(2px);
    z-index: -1;
}

.why-us::after {
    content: "";
    position: absolute;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: rgba(0, 0, 0, 0.3); /* Ajusta la opacidad para cambiar el nivel de oscurecimiento */
}

```

```

    z-index: -1;
}

.why-us > * {
    position: relative;
    z-index: 1;
}

.buttontruncate {
    overflow: visible;
    margin: 0 10px;
    white-space: nowrap; /* Evita que el texto se rompa en varias líneas */
    overflow: hidden; /* Oculta el desbordamiento del texto */
    text-overflow: ellipsis; /* Añade puntos suspensivos */
    --background: #d05281;
    font-family: "Roboto", sans-serif;
    font-weight: 600;
    font-style: normal;
    color: white;
    /* main.css */
    p {
        font-size: 15px;
        margin: 0;
        height: 44px;
        width: 155px;
        display: flex; /* Hacer que el contenedor sea un flex container */
        justify-content: center; /* Centrar horizontalmente */
        align-items: center; /* Centrar verticalmente */
        text-align: center; /* Asegurar que el texto esté centrado */
        padding-left: 10px;
        padding-right: 10px;
    }
}

.slidersections {
    background-color: #ffffff;
    padding: 20px 0px;
    h2 {
        font-family: "Poppins", sans-serif;
        font-style: normal;
        margin: 20px;
        margin-top: 0;
        font-size: 16px;
        font-weight: bold;
    }
}

```

```

.populars {
  display: flex;
  h2 {
    margin-top: 20px;
  }
}

.select_roll_container {
  --background: none;
  background: linear-gradient(to bottom, #c624ff 20%, #ff864e 100%),
    linear-gradient(to bottom, transparent 50%, #ffffff 60%);
  background-size:
    100% 50%,
    100% 50%;
  background-position: top, bottom;
  background-repeat: no-repeat;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  padding: 20px;
}

.select_roll_container ion-select {
  width: 100%;
  max-width: 400px;
}

.select_roll_container ion-button {
  --background: none;
  background: linear-gradient(to bottom, #c624ff -100%, #ff864e 100%);
  border-radius: 100%;
}

.edit_shadow {
  box-shadow:
    0 0px 6px 2px rgba(0, 0, 0, 0.4),
}

.edit_shadow2 {
  box-shadow:
    0 10px 5px -1px rgba(0, 0, 0, 0.4)
}

```

```

.no-focus-indicator ion-select {
  --background: none;
  --box-shadow: none;
  --outline: none;
}

/* También se puede aplicar a los elementos internos */
.no-focus-indicator ion-select .select-text {
  box-shadow: none;
}

.no-focus-indicator ion-select .select-icon {
  display: none;
}

.poppins{
  font-family: "Poppins", sans-serif;
}

.sw-mod {
  .swiper-autoheight, .swiper-autoheight .swiper-slide {
    height: 80px;
  }
  .swiper-scrollbar {
    z-index: 10;
  }
  .swiper-scrollbar-drag {
    --swiper-scrollbar-drag-bg-color: rgb(232, 93, 139);
  }
}

```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/main/Main.tsx

```

import {
  IonButton,
  IonContent,
  IonHeader,
  IonIcon,
  IonPage,
  IonText,
  IonTitle,
  IonToolbar,
} from "@ionic/react";
import "../main.css"; // Asegúrate de importar el archivo SCSS

```

```

import { generateSuggestions } from "../../utils/loadActivity";
import { useContext, useEffect, useState } from "react";
import { UserContext } from "../../components/context/UserContext";
import MainButton from "../../components/MainButton";
import { arrowForward } from "ionicons/icons";
import { useHistory } from "react-router";
// import Swiper and modules styles
import 'swiper/css';
import 'swiper/css/navigation';
import 'swiper/css/pagination';
import SwiperHabilities from "./SwiperHabilities";
import SwiperBall from "./SwiperBall";

const Main: React.FC = () => {
  const [suggestions, setSuggestions] = useState<any[]>([]);
  const { update } = useContext(UserContext);
  const history = useHistory();

  useEffect(() => {
    setSuggestions(generateSuggestions());
  }, [update]);

  return (
    <IonPage>
      <IonHeader>
        <IonToolbar className="custom-toolbar">
          <IonTitle className="lilita-one-regular">PIAPP</IonTitle>
        </IonToolbar>
      </IonHeader>
      <IonContent className="bgcolor min-h-full">
        <section className="why-us w-[100%]">
          <h1 className="text-white">
            ¿Por qué esta <span className="text-[#f95c1c]">app?</span>
          </h1>
          <p>
            Imagina un mundo donde los niños aprenden y desarrollan sus
            habilidades psicomotrices jugando. Este libro es tu guía para crear
            actividades lúdicas y educativas, transformando la rutina en una
            emocionante aventura. Promovamos juntos una infancia activa y
            saludable, ¡donde cada día sea una oportunidad para divertirse y
            crecer!
          </p>
          <MainButton
            className="custom-home-button"
            text="VER MÁS"
            onClick={() => {history.push(`/suggestions`)}}
          />
        </section>
      </IonContent>
    </IonPage>
  );
};

```

```

    />
  </section>
  <section className="">
    <div className="slidersections mt-5">
      <h2>Habilidades motoras</h2>
      <div className="sw-mod" >
        <SwiperHabilities />
      </div>
    </div>
    <div className="slidersections mt-5">
      <h2>Categorias de aporte con pelota</h2>
      <div className="sw-mod">
        <SwiperBall />
      </div>
    </div>
  </section>
  <section className="popular-activities">
    <div className="slidersections mt-5 populars">
      <h2>Actividades populares</h2>
      <IonButton
        fill="clear"
        className="text-[#F95715] font-bold text-left text-[10px]"
        onClick={() => {history.push(`/all`)}}
      >
        ver todas
      <IonIcon slot="end" icon={arrowForward}></IonIcon>
    </IonButton>
    </div>
  </section>
</IonContent>
</IonPage>
);
};

```

```
export default Main;
```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/main/SwiperBall.tsx

```

import React from 'react';
import { Swiper, SwiperSlide } from 'swiper/react';
import 'swiper/css';
import { Scrollbar } from 'swiper/modules';
import MainButton from '../../components/MainButton';
import './main.css';
import { useHistory } from 'react-router';
import 'swiper/swiper-bundle.css';

```

```

const SwiperBall: React.FC = () => {
  const history = useHistory();

  const truncateText = (text: string, maxLength: number) => {
    return text.length > maxLength ? `${text.substring(0, maxLength)}...` : text;
  };

  return (
    <Swiper
      modules={[Scrollbar]}
      slidesPerView={1.5}
      spaceBetween={10}
      loop={true}
      autoHeight={true}
      touchStartPreventDefault={false} // Para asegurar que sea deslizable en dispositivos t
      scrollbar={{ draggable: true }} // Activa la scrollbar
      className="h-20"
    >
      <SwiperSlide>
        <MainButton
          text={truncateText("LANZAR", 7)}
          onClick={() => { history.push('/allfilter/lanzar'); }}
          className="buttontruncate rounded-lg"
        />
      </SwiperSlide>
      <SwiperSlide>
        <MainButton
          text={truncateText("PATEO", 7)}
          onClick={() => { history.push('/allfilter/pateo'); }}
          className="buttontruncate rounded-lg"
        />
      </SwiperSlide>
      <SwiperSlide>
        <MainButton
          text={truncateText("RECIBIR", 7)}
          onClick={() => { history.push('/allfilter/recibir'); }}
          className="buttontruncate rounded-lg"
        />
      </SwiperSlide>
      { /* Agrega más SwiperSlides si es necesario */ }
    </Swiper>
  );
};

export default SwiperBall;

```


E:/Documentos/Clases/proyecto/cartilla/app/src/pages/main/SwiperHabilities.tsx

```
import React from "react";
import { Swiper, SwiperSlide } from "swiper/react";
import "swiper/css";
import { Scrollbar } from "swiper/modules";
import MainButton from "../../components/MainButton";
import "./main.css";
import { useHistory } from "react-router";
import "swiper/swiper-bundle.css";

const SwiperHabilities: React.FC = () => {
  const history = useHistory();

  const truncateText = (text: string, maxLength: number) => {
    return text.length > maxLength
      ? `${text.substring(0, maxLength)}...`
      : text;
  };

  return (
    <Swiper
      modules={[Scrollbar]}
      slidesPerView={1.5}
      spaceBetween={10}
      loop={true}
      autoHeight={true}
      touchStartPreventDefault={false} // Asegura que sea deslizable en todos los dispositivos
      scrollbar={{ draggable: true }}
      className="h-20"
    >
      <SwiperSlide>
        <MainButton
          text={truncateText("DESPLAZAMIENTO", 7)}
          onClick={() => {
            history.push("/allfilter/desplazamiento");
          }}
          className="buttontruncate rounded-lg"
        />
      </SwiperSlide>
      <SwiperSlide>
        <MainButton
          text={truncateText("SALTO", 7)}
          onClick={() => {
            history.push("/allfilter/salto");
          }}
        />
      </SwiperSlide>
    </Swiper>
  );
};
```

```

        className="buttontruncate rounded-lg"
      />
    </SwiperSlide>
    <SwiperSlide>
      <MainButton
        text={truncateText("LANZAR", 7)}
        onClick={() => {
          history.push("/allfilter/lanzar");
        }}
        className="buttontruncate rounded-lg"
      />
    </SwiperSlide>
    <SwiperSlide>
      <MainButton
        text={truncateText("RECIBIR", 7)}
        onClick={() => {
          history.push("/allfilter/recibir");
        }}
        className="buttontruncate rounded-lg"
      />
    </SwiperSlide>
    <SwiperSlide>
      <MainButton
        text={truncateText("REBOTAR", 7)}
        onClick={() => {
          history.push("/allfilter/rebotar");
        }}
        className="buttontruncate rounded-lg"
      />
    </SwiperSlide>
    <SwiperSlide>
      <MainButton
        text={truncateText("GOLPEAR", 7)}
        onClick={() => {
          history.push("/allfilter/golpear");
        }}
        className="buttontruncate rounded-lg"
      />
    </SwiperSlide>
  </Swiper>
);
};

export default SwiperHabilities;

```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/search/search.css

```
/* src/pages/main/main.scss */
```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/search/Search.tsx

```
import {
  IonContent,
  IonHeader,
  IonItem,
  IonList,
  IonPage,
  IonSearchbar,
  IonTitle,
} from "@ionic/react";
import { useState } from "react";
import "./search.css";
import { fetchDataAndSearchByKeyword } from "../../utils/loadActivity";
import { ActivityType } from "../../utils/types";
import Activity from "../../components/activities/Activity";

const Search: React.FC = () => {
  const [results, setResults] = useState<ActivityType[]>([]);
  const [showQuery, setShowQuery] = useState(false);

  const handleInput = (ev: Event) => {
    let query = "";
    const target = ev.target as HTMLIonSearchbarElement;
    if (target) query = target.value!.toLowerCase();

    fetchDataAndSearchByKeyword(query) // Asegúrate de proporcionar la categoría
      .then((result) => {
        if (typeof result === 'string') {
          setResults([]);
        } else {
          setResults(result);
          if (result.length > 0 && query !== "") setShowQuery(true);
          else setShowQuery(false);
        }
      })
      .catch((error) => {
      });
  };

  return (
    <IonPage>
```

```

<IonContent className="">
  <section className="min-h-full">
    <div className="min-h-[100vh] flex flex-col ">
      <IonSearchbar
        showClearButton="always"
        animated={true}
        placeholder="¿Qué estás buscando hoy?"
        onIonInput={(ev) => handleInput(ev)}
      ></IonSearchbar>
      {showQuery && results.length == 0 && (
        <IonTitle>No hay actividades para esta palabra</IonTitle>
      )}
      {showQuery && (
        <IonList className="p-0 custom-ionlist">
          {results.map((result, i) => (
            <Activity key={result.title + i} activity={result} />
          ))}
        </IonList>
      )}
    </div>
  </section>

</IonContent>
</IonPage>
);
};

```

```
export default Search;
```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/suggestion/suggestion.css

```

/* src/pages/main/main.scss */
@import url("https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=block");
.bodybgs {
  background-color: #f5f5f5;
  position: relative;
  overflow: hidden;
  display: flex;
  justify-content: start;
  align-items: center;
  flex-direction: column;
}
.bodybgs .infosugg {
  background-color: #ffffff;
  width: 90%;
  border-radius: 20px;
}

```

```

padding: 20px;
margin: 20px 0px;
box-shadow: 2px 2px 2px rgba(48, 48, 48, 0.486);
}
.bodybgs .infosugg h2 {
font-family: "Roboto", sans-serif;
text-align: center;
font-weight: 500;
margin-bottom: 20px;
}
.bodybgs .infosugg .parragraphp1 {
font-family: "Poppins", sans-serif;
font-size: 14px;
text-align: justify;
font-weight: 600;
color: #666666;
}

.bodybgxs {
--background: #f5f5f5;
}/*# sourceMappingURL=suggestion.css.map */

```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/suggestion/suggestion.css.map

```

{"version":3,"sources":["suggestion.scss","suggestion.css"],"names":[],"mappings":"AAAA,6BA

```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/suggestion/suggestion.scss

```

/* src/pages/main/main.scss */
@import url("https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=block");
.bodybgs {
background-color: #f5f5f5;
position: relative;
overflow: hidden;
display: flex;
justify-content: start;
align-items: center;
flex-direction: column;
.infosugg {
background-color: #ffffff;
width: 90%;
border-radius: 20px;
padding: 20px;
margin: 20px 0px;
box-shadow: 2px 2px 2px rgba(48, 48, 48, 0.486);
}
}

```

```

    h2 {
      font-family: "Roboto", sans-serif;
      text-align: center;
      font-weight: 500;
      margin-bottom: 20px;
    }

    .parragraphp1 {
      font-family: "Poppins", sans-serif;
      font-size: 14px;
      text-align: justify;
      font-weight: 600;
      color: #666666;
    }
  }
}

.bodybgxsx {
  --background: #f5f5f5;
}

```

E:/Documentos/Clases/proyecto/cartilla/app/src/pages/suggestion/Suggestion.tsx

```

import {
  IonContent,
  IonHeader,
  IonPage,
  IonText,
  IonTitle,
  IonToolbar,
} from "@ionic/react";
import "../suggestion.css";
import { useHistory } from "react-router";

const Suggestion: React.FC = () => {
  return (
    <IonPage>
      <IonHeader className="w-full h-7">
        <IonTitle class="custom-activitiesbar ion-text-center text-white">
          Información
        </IonTitle>
      </IonHeader>
      <IonContent>
        <section className="bodybgs min-h-full">
          <IonText
            color="dark"

```

```

        className="text-black uppercase font-bold m-5 text-2xl text-center "
    >
</IonText>
<div className="infosugg">
    <h2>Quiénes Somos</h2>
    <p className="parragraphp1">
        ¡Bienvenidos a un viaje de descubrimiento y diversión! Este tomo
        es el resultado de un emocionante proyecto de investigación
        titulado "Aplicación móvil para la estimulación psicomotriz y la
        reducción de hábitos de sedentarismo en la primera infancia",
        llevado a cabo en Fusagasugá, Colombia, y Muzambinho, Brasil.
    </p>
</div>
<div className="infosugg">
<h2>Nuestro Origen</h2>
    <p className="parragraphp1">
        Todo comenzó al observar la realidad en los hogares comunitarios
        tradicionales de bienestar en Fusagasugá, donde la atención a los
        niños es una prioridad. Gracias a la colaboración con las
        dedicadas madres comunitarias del municipio, los entusiastas
        estudiantes de la licenciatura en Educación Básica con énfasis en
        Educación Física, Recreación y Deporte de la Universidad de
        Cundinamarca, y las observaciones realizadas por nuestros curiosos
        investigadores, nació esta cartilla.
    </p>
</div>
<div className="infosugg">
<h2>Por Qué Esta App</h2>
    <p className="parragraphp1">
        Pero, ¿qué hace a esta cartilla tan especial? Más que una simple
        respuesta a las necesidades de la comunidad, es un complemento
        pedagógico diseñado para destacar los maravillosos beneficios de
        la recreación. Queremos que todos reconozcan cómo la diversión y
        el juego pueden estimular la actividad física y, al mismo tiempo,
        reducir el sedentarismo en los más pequeños.
    </p>
</div>
<div className="infosugg">
<h2>Nuestro Compromiso</h2>
    <p className="parragraphp1">
        Imagina un mundo donde los niños corren, saltan y juegan mientras
        aprenden y desarrollan sus habilidades psicomotrices. Este tomo es
        tu guía para crear ese mundo, lleno de actividades lúdicas y
        educativas que transformarán la rutina diaria en una aventura
        emocionante.
    </p>

```

```

    </div>
    <div className="infosugg">
    <h2>Únete a Nosotros</h2>
    <p className="parragraphp1">
        Únete a nosotros en esta misión de promover una infancia activa y
        saludable. Juntos, podemos hacer de cada día una oportunidad para
        aprender, crecer y, sobre todo, ¡divertirse!
    </p>
    </div>
    </section>
    </IonContent>
    </IonPage>
  );
};

```

```
export default Suggestion;
```

E:/Documentos/Clases/proyecto/cartilla/app/src/Styles/globals.css

```

@tailwind base;
@tailwind components;
@tailwind utilities;
/* @import url('styles.css'); */

```

E:/Documentos/Clases/proyecto/cartilla/app/src/Styles/styles.css

```

@import url("https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=block");
.basics {
  font-family: "Roboto", sans-serif;
  font-weight: 400;
  font-style: normal;
}/*# sourceMappingURL=styles.css.map */

```

E:/Documentos/Clases/proyecto/cartilla/app/src/Styles/styles.css.map

```

{"version":3,"sources":["styles.scss","styles.css"],"names":[],"mappings":"AAAQ,6JAAA;AAER;E

```

E:/Documentos/Clases/proyecto/cartilla/app/src/Styles/styles.scss

```

@import url('https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=block');
.basics{
  font-family: "Roboto", sans-serif;
  font-weight: 400;
  font-style: normal;
}

```


E:/Documentos/Clases/proyecto/cartilla/app/src/theme/variables.css

```
@charset "UTF-8";
/* For information on how to create your own theme, please see:
http://ionicframework.com/docs/theming/ */
* {
  font-family: Arial, sans-serif;
}

/* Fondo degradado en toda la barra de pestañas */
.hometab {
  background: #e85d8b;
}
.hometab ion-tab-button {
  background: transparent;
  position: relative;
}

ion-icon {
  color: rgb(85, 6, 158);
}

ion-icon:hover {
  color: rgb(205, 173, 235);
  /* Color azul claro al pasar el mouse */
}

ion-icon.tab-selected {
  color: rgb(28, 1, 53);
  /* Color azul claro cuando el ícono está activo */
}

.bodybgs .infosugg {
  background-color: rgb(85, 6, 158);
  width: 90%;
  border-radius: 20px;
  padding: 20px;
  margin: 10px 0px;
}

.parragraphp1 {
  color: #FFF;
  text-align: left;
  font-size: 20px;
}
```

```

.texth1 {
  text-transform: uppercase;
  font-weight: bold;
  margin: 20px;
  font-size: 25px;
  color: #FFF;
  text-align: center;
}

.activityBg {
  background-color: rgb(85, 6, 158);
  color: #fff;
  border-radius: 20px;
  margin-top: 20px;
}

.activityTitle {
  color: #E1BEE7;
  font-weight: bold;
  font-size: x-large;
  letter-spacing: 5px;
}

.activityBtn {
  background-color: #864196;
}

ion-card img {
  max-height: 300px;
  width: 100%;
}/*# sourceMappingURL=variables.css.map */

```

E:/Documentos/Clases/proyecto/cartilla/app/src/theme/variables.css.map

```
{
  "version": 3,
  "sources": [
    "variables.css",
    "variables.scss"
  ],
  "names": [],
  "mappings": "AAAA,gBAAGL"
}
```

E:/Documentos/Clases/proyecto/cartilla/app/src/theme/variables.scss

```

/* For information on how to create your own theme, please see:
http://ionicframework.com/docs/theming/ */

```

```

* {
  font-family: Arial, sans-serif;
}

```

```

/* Fondo degradado en toda la barra de pestañas */
.hometab {
    background: #e85d8b;
    ion-tab-button {
        background: transparent;
        position: relative;
    }
}

ion-icon {
    color: rgb(85, 6, 158);
}

ion-icon: hover {
    color: rgb(205, 173, 235);
    /* Color azul claro al pasar el mouse */
}

ion-icon.tab-selected {
    color: rgb(28, 1, 53);
    /* Color azul claro cuando el ícono está activo */
}


.bodybgs .infosugg {
    background-color: rgb(85, 6, 158);
    width: 90%;
    border-radius: 20px;
    padding: 20px;
    margin: 10px 0px;
}

.parragraphp1 {
    color: #FFF;
    text-align: left;
    font-size: 20px;
}

.texth1 {
    text-transform: uppercase;
    font-weight: bold;
    margin: 20px;
    font-size: 25px;
    color: #FFF;
    text-align: center;
}

```

```

}

.activityBg {
  background-color: rgb(85, 6, 158);
  color: #fff;
  border-radius: 20px;
  margin-top: 20px;
}

.activityTitle {
  color: #E1BEE7;
  font-weight: bold;
  font-size: x-large;
  letter-spacing: 5px;
}

.activityBtn {
  background-color: #864196;
}

ion-card img {
  max-height: 300px;
  width: 100%;
}

```

E:/Documentos/Clases/proyecto/cartilla/app/src/utils/loadActivity.ts

```

import activities from "../activities/saltar/saltar.json";
import { ActivityType } from "../types";

async function fetchDataAndSearchById(id: string, category: string) {
  try {
    const data = await import(`../activities/saltar/saltar.json`);
    const dataMap = new Map(data.default.map((item: any) => [item.id, item]));
    const result = dataMap.get(id);
    return result ? result : `Item with id ${id} not found.`;
  } catch (error: any) {
    return `Failed to load data for category ${category}: ${error.message}`;
  }
}

function fetchAllData() {
  return activities;
}

async function fetchDataAndSearchByKeyword(keyword: string) {

```

```

try {
  const data = await import(`../activities/saltar/saltar.json`);
  const activities: ActivityType[] = data.default;

  const result = activities.filter(activity => {
    return Object.values(activity).some(value => {
      if (typeof value === 'string') {
        return value.toLowerCase().includes(keyword.toLowerCase());
      } else if (Array.isArray(value)) {
        return value.some(item => {
          if (typeof item === 'string') {
            return item.toLowerCase().includes(keyword.toLowerCase());
          } else if (typeof item === 'object') {
            return Object.values(item).some(subValue =>
              typeof subValue === 'string' && subValue.toLowerCase().includes(keyword.toL
            );
          }
          return false;
        });
      }
      return false;
    });
  });

  return result.length > 0 ? result : `No items found containing "${keyword}".`;
} catch (error: any) {
  return `Failed to load data for category: ${error.message}`;
}
}

// Definición de los tipos
interface UserData {
  category: string;
  id: string;
}

function saveUserData(data: UserData) {
  let jsonData: UserData[] = [];

  // Leer los datos existentes de localStorage
  const fileData = localStorage.getItem('userData');
  if (fileData) {
    try {
      jsonData = JSON.parse(fileData) as UserData[];
    } catch (err) {
      console.error('Error parsing JSON:', err);
    }
  }
}

```

```

    }
  }

  // Agregar los nuevos datos
  jsonData.push(data);

  // Guardar los datos actualizados en localStorage
  localStorage.setItem('userData', JSON.stringify(jsonData));
  console.log('Data saved successfully');
}

function getUserData(): UserData[] {
  const fileData = localStorage.getItem('userData');
  if (fileData) {
    try {
      return JSON.parse(fileData) as UserData[];
    } catch (err) {
      console.error('Error parsing JSON:', err);
      return [];
    }
  }
  return [];
}

function generateSuggestions(): ActivityType[] {
  const interactions = getUserData();
  const categoryCount: { [key: string]: number } = {};
  const keywordCount: { [key: string]: number } = {};

  // Contar las interacciones por categoría y palabras clave
  interactions.forEach(interaction => {
    // Contar categorías
    if (categoryCount[interaction.category]) {
      categoryCount[interaction.category]++;
    } else {
      categoryCount[interaction.category] = 1;
    }

    // Contar palabras clave
    const activity = activities.find(a => a.id === interaction.id);
    if (activity) {
      const keywords = activity.keywords.split(' ');
      keywords.forEach(keyword => {
        if (keywordCount[keyword]) {
          keywordCount[keyword]++;
        } else {

```

```

        keywordCount[keyword] = 1;
    }
    });
}
});

// Encontrar la categoría más frecuente
const mostFrequentCategory = Object.keys(categoryCount).reduce((a, b) => categoryCount[a] > categoryCount[b] ? a : b);

// Encontrar las palabras clave más frecuentes
const mostFrequentKeywords = Object.keys(keywordCount).sort((a, b) => keywordCount[b] - keywordCount[a]);

// Generar sugerencias basadas en la categoría y palabras clave más frecuentes
const suggestions = activities.filter(activity =>
    (activity.category === mostFrequentCategory && mostFrequentKeywords.some(keyword => activity.keywords.includes(keyword)))
);

return suggestions;
}

export {
    fetchDataAndSearchById,
    fetchAllData,
    fetchDataAndSearchByKeyword,
    saveUserData,
    getUserData,
    generateSuggestions
};

```

E:/Documentos/Clases/proyecto/cartilla/app/src/utils/loadUserData.ts

E:/Documentos/Clases/proyecto/cartilla/app/src/utils/types.ts

```

export type ActivityType = {
    id: string;
    title: string;
    category: string;
    keywords: string;
    purpose: string;
    description: string;
    materials: string[];
    variables: string;
    psicomotor: {

```

```
        title: string;
        description: string;
    }[];
    extraInfo: {
        title: string;
        description: string;
    }[];
    img: string;
};
```