



CRUD Firebase React

AGENDA DEL DÍA

Mindset Digital	09:00 am	11:00 am	CRUD Firebase y React
Presentación entregable	09:15 am	12:00 pm	Kahoot del conocimiento
Revisión de proyecto No 4	09:25 am	01:45 pm	Retrospectiva ejercicio individual
Repaso de props	09:30 am	01:40 am	Sesión de dudas
Introducción a Firebase	10:00 am		
Break	10:50 am		



¿QUÉ ES FIREBASE?



Es una plataforma de backend como servicio construida sobre la infraestructura de Google. Entrega funciones propias del backend como:



AUTENTICACIÓN

Inicio de sesión con email, redes sociales, google, etc.



CLOUD MESSAGING

Notificaciones en tiempo real y envío de SMS



FIRESTORE

Bases de datos no SQL con escalamiento automático.



CLOUD STORAGE

Almacenamiento de archivos de todo tamaño.



CLOUD FUNCTIONS

Lógica de negocio que se puede ejecutar en eventos.



HOSTING

Publica tu web con 1GB de espacio gratuito.



Puedes implementar Firebase en web, Android y iOS



Manos a la obra

Buscar en google firebase



**Para acceder a Firebase necesitamos
registrarnos**



Debemos crear un nuevo proyecto



Ahora debemos crear una base de datos

- Seleccionamos Firestore DataBase
- Seleccionamos crear Base de Datos
- Seleccionamos modo de prueba
- Ahora seleccionamos colección
- Agregamos un nombre a nuestra colección. Para este ejemplo pongamos tareas
- Agrega los campos que requieras

Genial, ya tienes tu primer base de datos con Firebase



Configuremos nuestro proyecto

- Seleccionamos descripción general del proyecto
- Seleccionamos la opción web
- Nos pedirá un nombre para la aplicación
- Damos click en registrar
- Genial. Nos indicará el paquete y la configuración de firebase para nuestro proyecto



Instalación de paquetes

```
npm install firebase
```



Ahora debemos crear una nueva app en react

```
npx create-react-app crud
```



Instalación de paquetes

```
npm install firebase
```



Configuremos nuestro archivo de Firebase

- Dentro del proyecto en react, crea una carpeta con el nombre config
- Crea un archivo firebase.js
- copia la configuración



Configuremos nuestro archivo de Firebase

```
import { initializeApp } from 'firebase/app';  
import 'firebase/firestore';  
import { getFirestore } from 'firebase/firestore/lite';
```

```
const firebaseConfig = {  
  apiKey: 'AIzaSyCS6ripONtSFHYdh7bqjWEhPrXJ-erx5Qs',  
  authDomain: 'example-1crud.firebaseio.com',  
  projectId: 'example-1crud',  
  storageBucket: 'example-1crud.appspot.com',  
  messagingSenderId: '689358188786',  
  appId: '1:689358188786:web:18e7281c4f5a9622af606b',  
};
```

```
const app = initializeApp(firebaseConfig);  
export const db = getFirestore(app);
```



Obtener información de la DB de Firebase

```
import { collection, getDocs } from 'firebase/firestore/lite';
```

```
try {  
  const getData = async () => {  
    const DB = collection(db, 'tareas');  
    const task = await getDocs(DB);  
    const taskList = task.docs.map((d) => ({ id: d.id, ...d.data() }));  
    setList(taskList);  
  };  
  
  getData();  
} catch (e) {}
```

En useEffect no olvides usar []



Agregar datos a la colección

```
import { db } from '../config/firebase';
import { collection, getDocs, addDoc, updateDoc, deleteDoc, doc, } from 'firebase/firestore/lite';

const add = async () => {
  try {
    const dataDB = {
      ...form,
    };

    const DB = collection(db, 'tareas');
    const data = await addDoc(DB, dataDB);

    setList([...list, { ...dataDB, id: data.id }]); Agregamos el nuevo elemento creado sin actualizar
    setForm({ nombre: '', fecha: '' }); Limpiamos nuestro formulario
  } catch (e) {
    console.log(e);
  }
};
```



Actualizar datos de la colección

```
import { db } from './config/firebase';
import { collection, getDocs, addDoc, updateDoc, deleteDoc, doc, } from 'firebase/firestore/lite';

const update = async () => {
  const campoNombre = {
    nombre: form.nombre,
  };
  try {
    const taskRef = doc(db, 'tarefas', form.id);
    await updateDoc(taskRef, campoNombre);
    const actualizaRegistro = list.map((item) =>
      item.id === form.id
        ? { id: item.id, nombre: form.nombre, fecha: form.fecha }
        : item
    );
    setList(actualizaRegistro);
  } catch (e) {}
};
```



Eliminar datos de la colección

```
import { db } from './config/firebase';  
import { collection, getDocs, addDoc, updateDoc, deleteDoc, doc, } from 'firebase/firestore/lite';  
  
try {  
  const ref = doc(db, 'tarefas', item.id);  
  await deleteDoc(ref);  
  const filtradoElementos = list.filter((list) => list.id !== item.id);  
  setList(filtradoElementos);  
} catch (e) {  
  console.log(e);  
}
```



