MANUAL DEL DESARROLLADOR

PROYECTO COCINAPP

Manual del desarrollador

Índice de contenido

Estructura de carpetas del código fuente:

css: contiene el archivo style.css.

docs: contiene todos los archivos jsdoc generados.

images: contiene todas las imágenes locales utilizadas por el código (fondos, iconos…).

js: contiene todos los archivos .js del código (archivos de funciones, de eventos, de configuracion…).

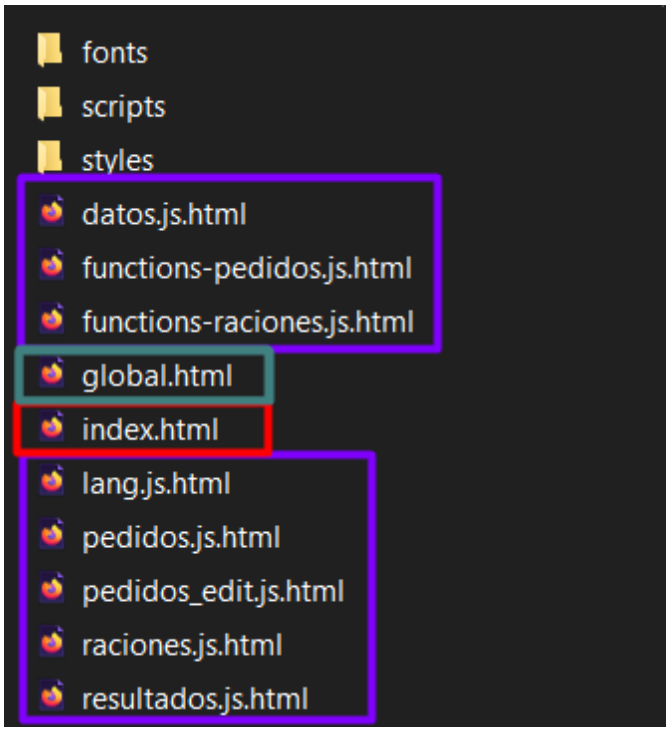
pages: contiene todos los archivos .html a excepción de index.html.

index.html: archivo .html para la página principal de nuestra pagina web.

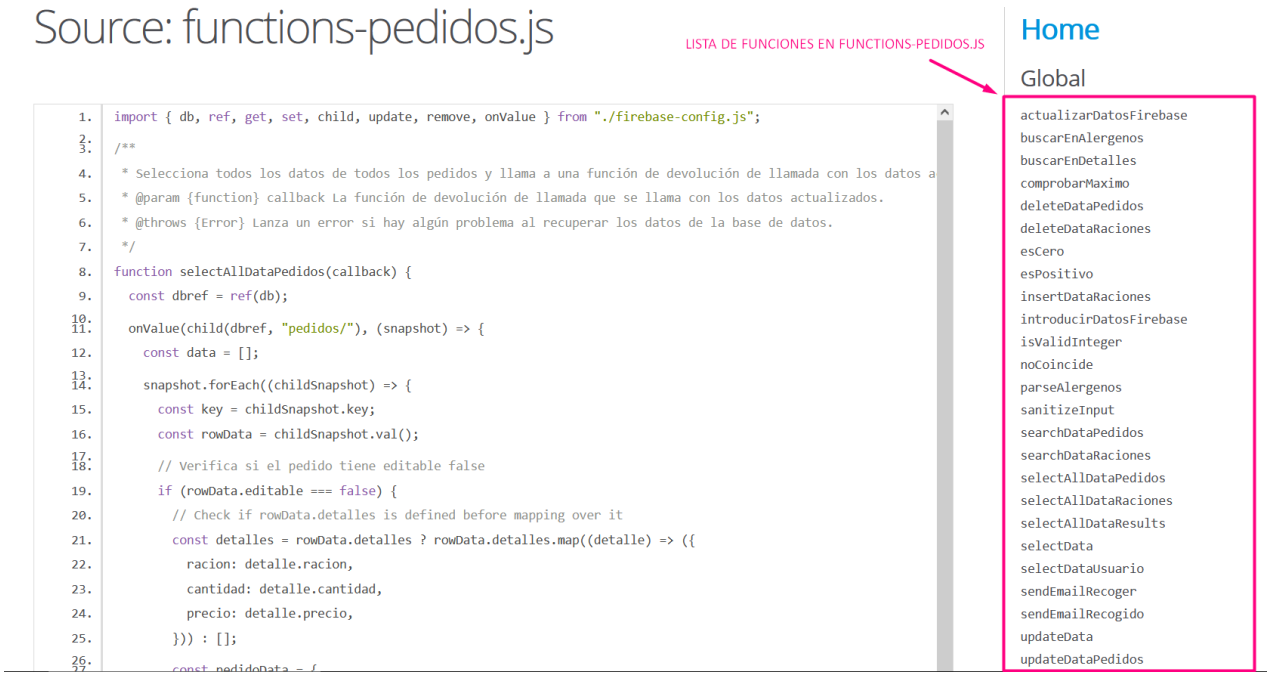
jsdoc.conf.json: configura la generación de código por jsdoc. Aquí hemos configurado que cree la documentación para los archivos de la carpeta js y añada a documentación en docs.

Documentación del código:

Hemos utilizado jsdoc para comentar las funciones de nuestro código. En la carpeta docs se encuentran todos los archivos generados por jsdoc:

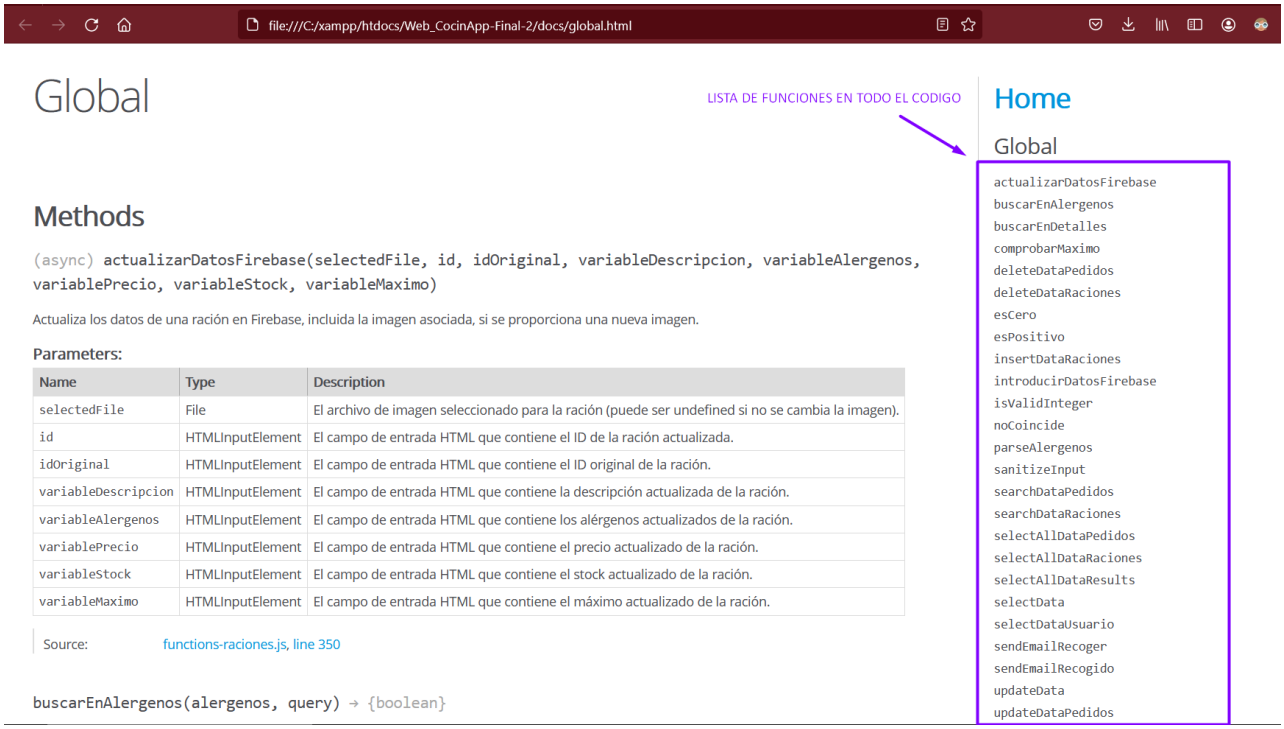


Archivos de código fuente y sus funciones: en estos archivos se muestra el código fuente de cada archivo .js, con la lista de funciones que lo componen a la derecha:



global.html:

Se puede encontrar la lista de todas las funciones en index.html o global.html:



index.html: pagina home de la documentación jsdocs. Muestra una introducción al proyecto y enlaza con las funciones de global.html:

Instrucciones para hostear la página en un servidor:

La documentación detalla sobre la instalación de la página en el servidor se encuentra en el archivo:

Manual_de_desarrollador_servidor.pdf

Cambio de estado editable de los pedidos (Python)

Un pedido creado por un cliente pasara una hora en estado editable para que el cliente pueda editar o eliminar el pedido. Pasado este tiempo, solo cocina podrá editar ciertos datos como el estado, la fecha y la hora, o borrar el pedido.

Para controlar esto, se utiliza una tarea en Python que comprobara cada hora con la base de datos qué pedidos se deben cambiar a no editables y cambiara el valor editable a false.

Solo los pedidos con editable = false aparecerán en la lista de pedidos de nuestra página.

La documentación detalla sobre el funcionamiento e instalación de dicha tarea se encuentra en el archivo:

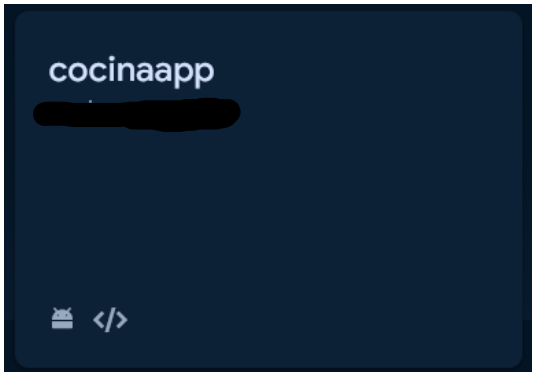
AppCocina_Tarea_actualizar_edicion_Pedido.pdf

Credenciales de firebase (nueva base de datos)

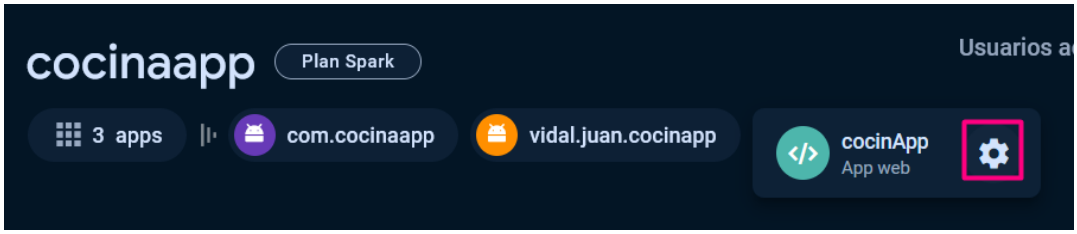
Si en algún momento se cambia la base de datos de firebase que la aplicación esta utilizando, debemos darle las credenciales de conexión para esa base de datos especifica en el archivo firebase-config.js:

```
JS firebase-config.js > ...
//FUNCIONES IMPORTADAS
import { initializeApp } from "https://www.gstatic.com/firebasejs/10.7.0/firebase-app.js";
import { getAnalytics } from "https://www.gstatic.com/firebasejs/10.7.0/firebase-analytics.js";
import { getStorage } from "https://www.gstatic.com/firebasejs/10.7.0/firebase-storage.js";
import { getDatabase, ref, get, set, child, update, remove, onValue,} from "https://www.gstatic.com/firebasejs/10.7.0/firebase-database.js";

// CONFIGURACION DE LA BASE DE DATOS DE FIREBASE
const firebaseConfig = {
  apiKey: "[redacted]",
  authDomain: "[redacted]",
  databaseURL: "[redacted]",
  projectId: "[redacted]",
  storageBucket: "[redacted]",
  messagingSenderId: "[redacted]",
  appId: "[redacted]",
  measurementId: "[redacted]"
};

//INICIARNDO FIREBASE
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
const db = getDatabase();
//OBTENER REFERENCIA AL ALMACENAMIENTO DE FIREBASE
const storage = getStorage(app);//esto es lo nuevo

// EXPORTAMOS LOS OBJETOS DE FIREBASE PARA USARLOS EN OTROS FICHEROS
export { app, analytics, storage, db, ref, get, set, child, update, remove, onValue };
```

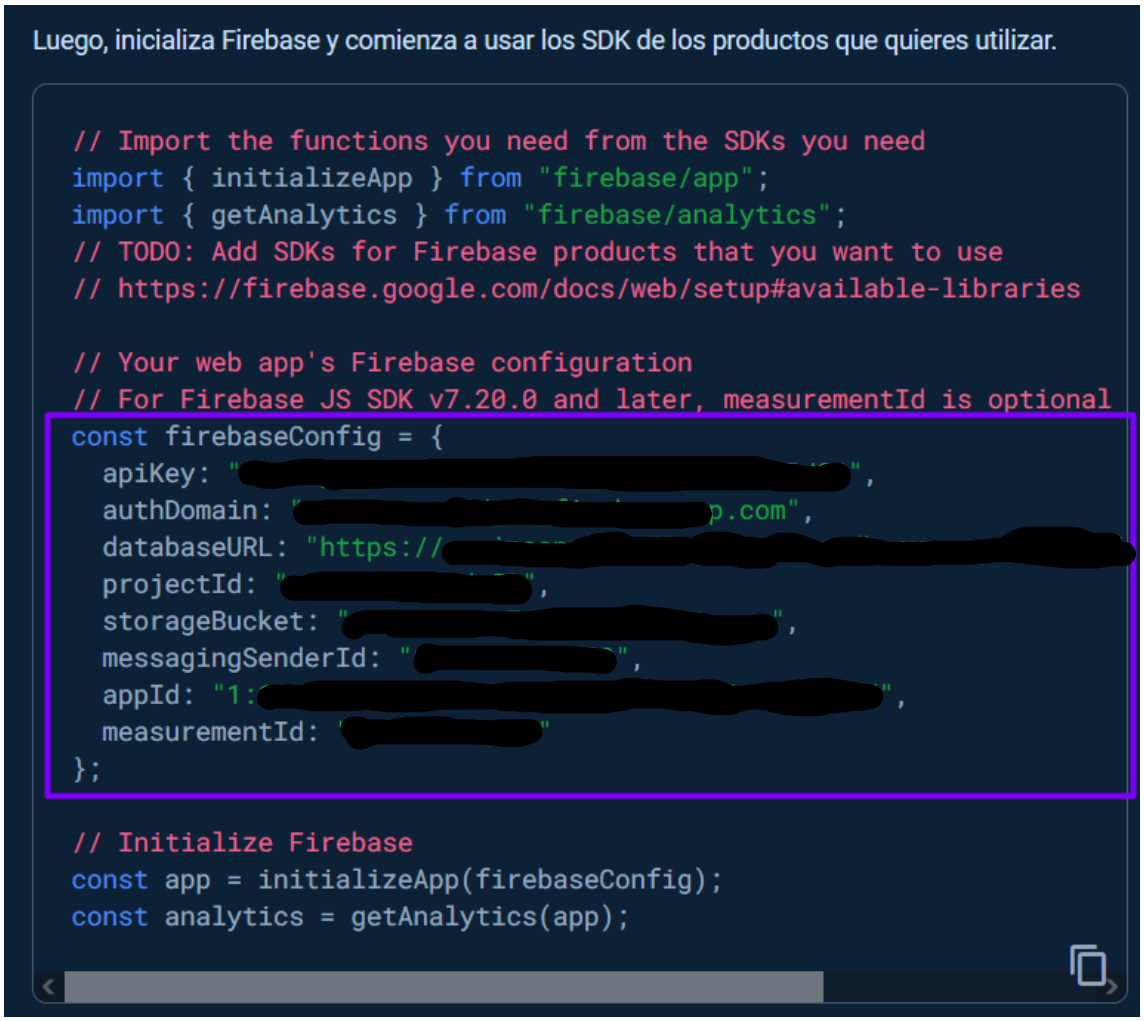
Para encontrar estas credenciales en firebase, se selecciona el proyecto:



Seleccionar la app web e ir a opciones:



Bajamos, y copiamos la constante firebaseConfig:



Cuentas de correo para el login:

Para iniciar sesión en CocinApp, se utilizan los correos especificados en el archivo auth.js:

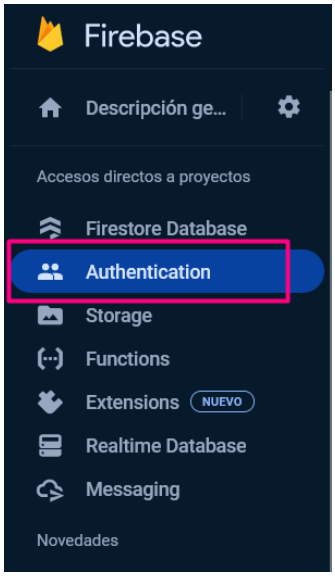
Como dice el código, estos correos son cuentas de usuarios administradores registrados en la base de datos Authentication en el firebase. La contraseña será la contraseña del usuario en Authentication en firebase.

Si se pierde la contraseña de estas cuentas, desde la cuenta de firebase se pueden crear usuarios nuevos en Authentication y borrar los antiguos.

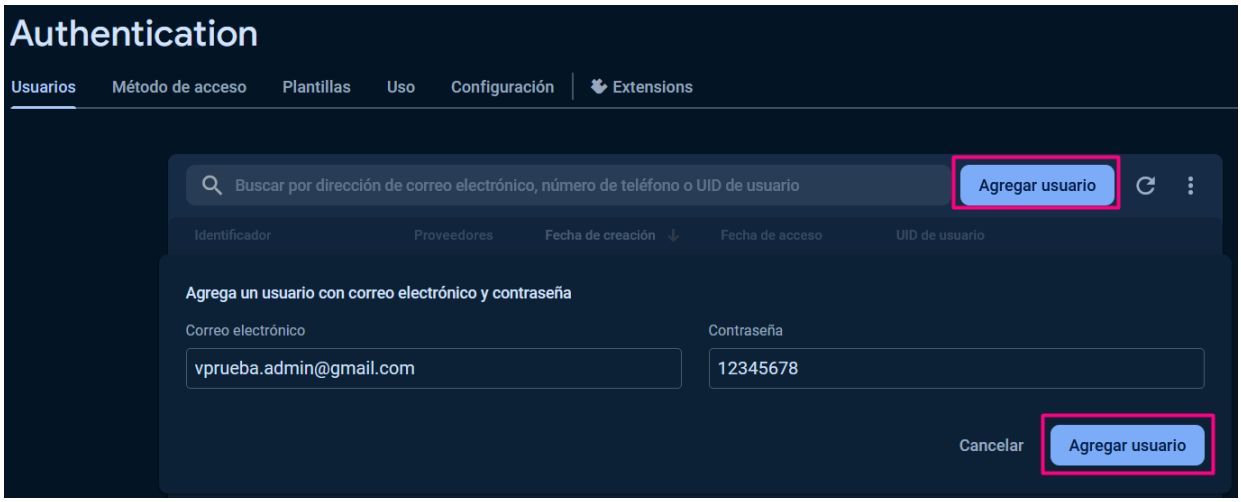
En caso de crear nuevos usuarios habría que cambiar los usuarios de “admin_email” en nuestro código por los nuevos. El código esta creado para manejar cuentas de usuario en forma de correos electrónicos, por lo que los nuevos usuarios deben tener el mismo formato.

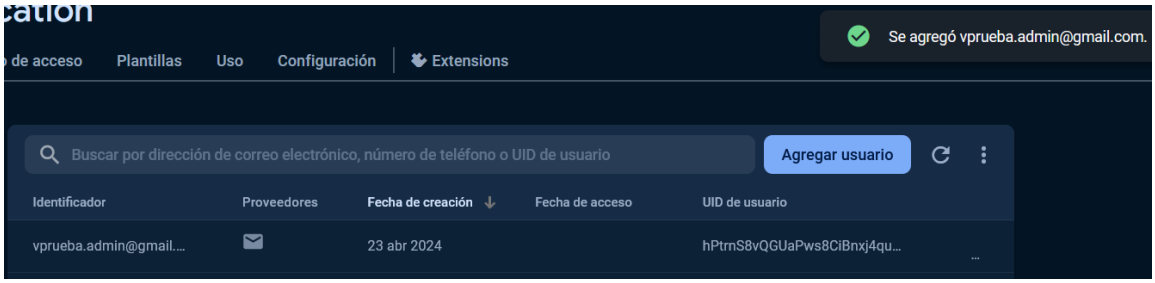
Como crear una nueva cuenta en Authentication:

En la cuenta de firebase vamos a Authentification:



En Authentication, pulsamos en “Agregar usuario”, añadimos el correo que queremos usar y la contraseña, y le damos a agregar:





Ahora se debe añadir ese correo a auth.js, en admin_email, y ya podremos usar nuestro nuevo usuario para iniciar sesión en el login.

Envío de correo al usuario al cambiar estado del pedido:

Cuando en pedidos.html cambiamos el estado de un pedido a recoger o recogido, se utiliza mailto para abrir la aplicación de correo predeterminada de nuestro equipo y generar un correo con el cliente como destinatario, notificándole de los cambios en su pedido.

```
function sendEmailRecogido(senderEmail, recipientEmail, idPedido) {
  console.log("enviando correo");

  // Asunto y cuerpo del email
  const subject = 'Su ha sido recogido';
  const body = `Su pedido con el codigo ${idPedido} ha sido recogido.`;

  // Creando el mailto URI
  const mailtoURI = `mailto:${recipientEmail}?subject=${encodeURIComponent(subject)}&body=${encodeURIComponent(body)}&from=${senderEmail}`;

  // Abre el cliente email por defecto con el email pre-hecho
  window.open(mailtoURI, '_self');
}
```

Es importante que en nuestra aplicación de correo predeterminada estemos logueados con el correo de cocina para que al generar este email utilice el correo de cocina como remitente.

A pesar de que en el código le damos como remitente la cuenta de cocina desde pedidos.js, varios servicios de correo solo permiten enviar correos del remitente que haya iniciado sesión en el servicio, por lo que es importante tenerlo en cuenta.

Correo correspondiente en contacto.html

Hay que tener en cuenta cambiar el correo que aparece en contacto.html si queremos que la cuenta de contacto sea otra:


```
<!-- Botón de email con icono -->
<a href="mailto:[redacted]@educacion.navarra.es" class="btn btn-outline-secondary mb-2">
  <i class="bi bi-envelope"></i>
</a>
<input type="email" class="form-control" id="email" value="[redacted]@educacion.navarra.es" readonly>
</div>
```