CURSO DESARROLLADOR DE APLICACIONES EN LA NUBE

NIVEL 2

LECCION1

**DESARROLLAR EL FRONT END DE LA APLICACIÓN**

La codificación de la interfaz gráfica de tu aplicación es indispensable porque es la única forma que el usuario pueda utilizarla ya que cuenta con elementos como botones y menús que permiten la interacción; y, a nivel de código te permite vincular su funcionalidad con la plataforma de desarrollo de aplicaciones en la nube

En el siguiente ejemplo desarrollaras la interfaz gráfica de una app web de autenticación que le permita ingresar al usuario por medio de su correo electrónico, a un sistema de compra en línea para adquirir las entradas de un parque acuático

**CODIFICAR LA INTERFAZ GRAFICA**

Para codificar la interfaz grafica de la aplicación, realiza lo siguiente:

Crea una carpeta en mis documentos y nómbrala como PrimeraApp

En la carpeta crear mediante el editor un archivo INDEX.html, este archivo es la pagina de inicio de tu aplicación. En donde utilizaras el lenguaje de marcado de hipertexto para estructurar sus contenidos, por lo que es importante

* **<DOCTYPE html>** para indicar que es un documento tipo “html”
* etiqueta de inicio de **<html>** y su correspondiente cierre **</html>**
* apertura de la etiqueta de encabezado **<head>** y su cierre **</head>**
* tipo de codificación a utilizar **<meta charset=”UTF-8”>** con la finalidad de que los caracteres especiales como la “ñ” y los acentos de las palabras se visualicen de forma correcta cuando los usuarios naveguen en la interfaz de la aplicación
* **<meta name=”viewport” content=”width=device-width”>** para asegurarse que la interfaz gráfica se ajusta a dispositivos de diversos tamaños
* El nombre de tu sitio web entre las etiqueta **<title> MI PRIMERA APP DE AUTENTICACION </title>** para que el texto aparezca en la pestaña del navegador.
* El enlace de tipo de fuente que deseas utilizar en tu sitio -> **<link href=”**[**https://fonts.googleapis.com/css?family=Poppins**](https://fonts.googleapis.com/css?family=Poppins)**” rel=”stylsheets”>**
* Enlace a tu hoja de estilo en cascada o “css”, que es el archivo en el que aplicaras estilos gráficos a las etiquetas html, para dar color a todos los elementos del sitio, además de acomodarlos **<link rel=”stylesheet” type=”text/css” href=”miEstilo.css”>**
* **<body> </body>** cuerpo que contendrá todos los elementos que serán visibles de la interfaz para el usuario final
* En el cuerpo debes escribir <h1> </h1> entre ellas esta el texto del encabezado de tu aplicación web, ejemplo **<h1>Inicia sesión para comprar tus entradas</h1>**
* **Contenedores o <div> </div>** se utilizan para organizar los contenidos de la aplicación. Ejemplo un contenedor general que contendrá una división que en su interior contendrá una imagen, y un div mas llamado principal que contendrá la fotografia de los usuarios que se autentiquen en la app por medio de la cuenta de su correo electrónico, además de la opción recordar

**<div id=contenedor>**

**<div id=”foto”></div>**

**<button id=”logueo”></button>**

**<div id=”principal”></div>**

**<button id=”guardar”></button>**

**</div>**

**ESTILOS DE LA APLICACIÓN**

La aplicación de estilos a las etiquetas de la aplicación html se hace por medio de un documento css, “hojas de estilo en cascada – Cascading Style sheets”

* Crea un nuevo documento con el nombre miEstilo.css y guárdalo en la carpeta anterior.
* Para asignar un color a la pantalla del navegador:

body{

background-color:#040F7F;

}

* Define el color de la fuente, familia tipográfica y alineación del texto para el encabezado h1

h1{

color:#blanco;

font-family:’Poppins’,san-serif;

align:center;

}

* Establece las propiedades en común para los botones “logueo” y “guardar”de mensajes “por medio de correo electrónico” y “recordar”, en el ejemplo ambos botones son de color negro y tamaño de texto de 16 pixeles, letra blanca, borde azul de 1 px, texto centrado
* Ademas el cursor será en forma de una mano, y el botón “guardar” estara debajo de “logueo”

#logueo,#guardar{

background-color:black;

color:white;

font-size:16px;

border: 1px solid#4FB1FF;

text-align:center;

cursor:pointer;

display:inline-block;

}

* Determinar el comportamiento cuando el cursor pase por encima de los botones

#logueo:hover,#guardar:hover{

background-color:#medioblanco;

color:black;

}

* Establece las reglas de estilo para el div que contiene al resto de elementos, en el ejemplo se le asigna un ancho automatico al área con id “contenedor” para que este crezca automáticamente según el nro de elementos en su interior y se deben centrar los botones

#contenedor{

width:auto;

text-align:center;

}

* Coloca una imagen dentro del “div” con id “foto”, ancho del 50%, alto de 400 pixeles, dar valor 0 auto a la propiedad margin, y posición relativa para que la imagen salga centrada dentro del div. Por ultimo utiliza la propiedad backgroun-image: url para invocar a una imagen desde un enlace activo en la web, e indicale con la propiedad bacground-size que la imagen debe adaptarse al tamaño del div, y que la imagen no se repita.

#foto{

widht:50%;

height:400px;

margin:0 auto;

position:relative;

background-image:url(<http://www.ninos-felices.com.mx/w>)

background-size:contain;

bakcgroun-repeat:no repeat;

}

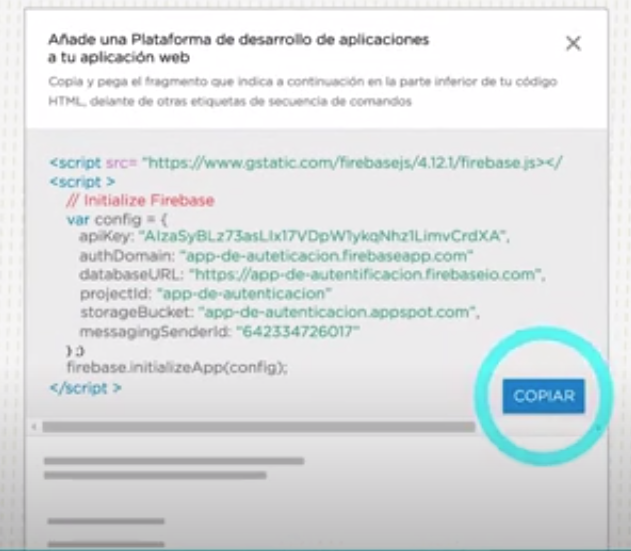
**INCORPORAR LOS SERVICIOS DE LA PLATAFORMA A UNA APLICACIÓN WEB**

La plataforma de desarrollo de aplicaciones en la nube te permite contar con un BACK END para tus aplicaciones, ya sean MOVILES o en WEB, sin necesidad de programar todas las funcionalidades desde cero, para ello es necesario que incorpores sus servicios a tu aplicación mediante los siguientes pasos:

IMPLEMENTACION DE SERVICIOS

* Ingresa al sitio del proveedor de la plataforma y da CLIC en ACCEDER, para acceder a la herramienta por medio de tu correo electrónico

https://console.firebase.google.com/?hl=es-419

* Da clic en IR A LA CONSOLA
* Da clic en AÑADIR PROYECTO, para añadir uno nuevo y nombralo “APP DE AUTENTICACION”
* Selecciona tu país en la lista desplegable, y
* Da clic en el botón CREAR PROYECTO,
* Una vez generado la plataforma te redirecciona al “MENU DE TIPOS DE APLICACONES” que puedes desarrollar en conjunto con esta tecnología, da clic en “AÑADIR A LA APLICACIÓN WEB”
* Copia el fragmento de código que te muestra el navegador en tu archivo “index .html”, es importante que lo ubiques antes del cierre de la etiqueta </body>, para evitar que la aplicación tarde en cargarse debido al peso de los archivos
* Observa que en el código se encuentran 2 <script>, el primero importa la librería de la plataforma para que puedas utilizar todas las herramientas que esta ofrece,
* el 2do son las referencias que la plataforma usa para iniciar sus servicios desde tu aplicación y son:

**APIKEY**.- El número de identificación de tu proyecto

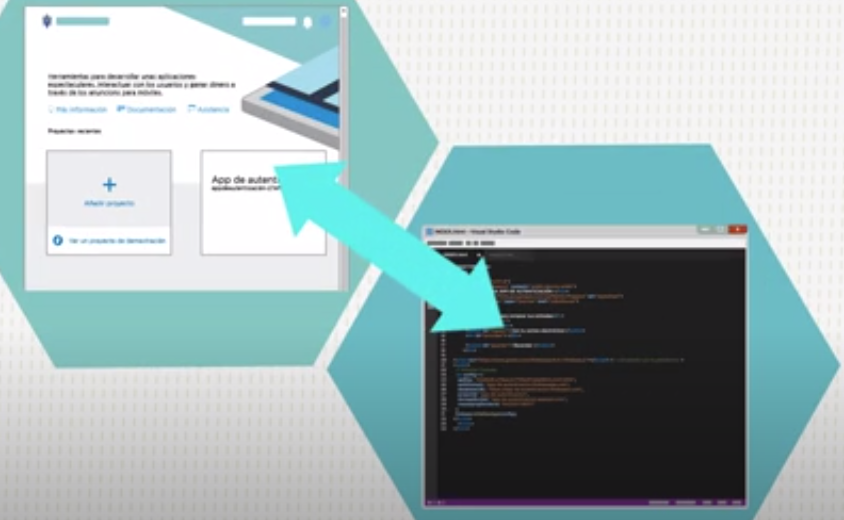
**AUTHDOMAIN**.- Dominio de autenticación del proyecto que te permite utilizar el servicio AUTH

**DATABASEURL**.- Url de la base de datos del proyecto con el que puedes usar el servicio “base de datos en tiempo real” DATABASEURO

**PROJECTID**.- Identificador del proyecto

**STORAGEBUCKET**.- Depósitos de almacenamiento del proyecto con el que puedes utilizar el servicio de almacenamiento en la nube

**MESSAGESENDERID**.- Identificador único de los mensajes de notificación y datos del proyecto con los que es posible utilizar mensajería en la nube.

Al utilizar las referencias de la plataforma, se vinculan entre si el proyecto que creaste en la web y el código de tu aplicación.

**REQUISITOS PREVIOS A LA PROGRAMACION DEL BACK END**

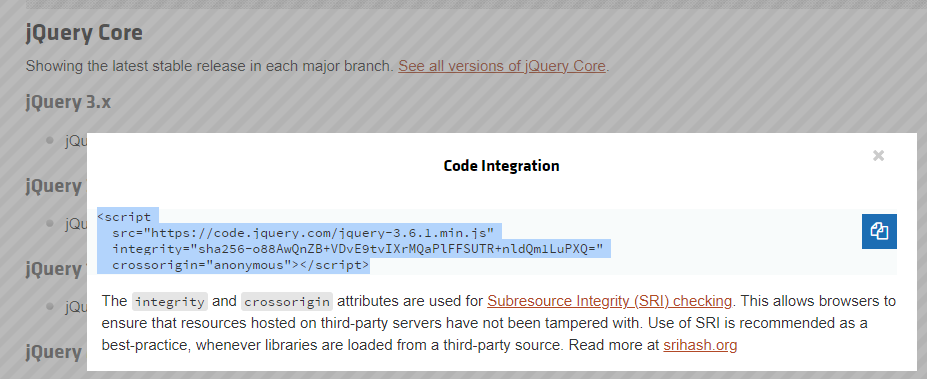
Para que tu aplicación web sea funcional también tienes que codificar el BACK END, es decir programar cada uno de los componentes que harán que esta responda a las peticiones del usuario.

Para lograrlos es indispensable:

* utilizar un lenguaje de programación, como JAVASCRIPT, para codificar la autentificación del usuario
* Instalar un servidor local, con la finalidad de probar su funcionamiento, lo que se hace vinculando la librería JQUERY del lenguaje JAVASCRIPT a tu documento html (index.html) con el objetivo de facilitar la manipulación de sus etiquetas e id durante el desarrollo de las funcionalidades de la aplicación.
* Ejecuta los siguientes pasos con el fin de vincular los recursos de Jquery a tu archivo index.html



1.- Ingresa al buscador y -> JQUERY CDN

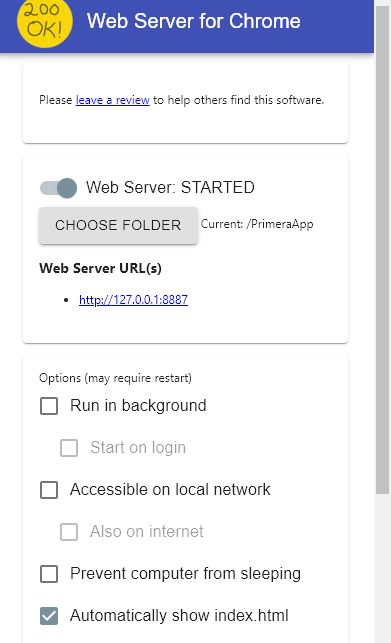
 Da clic en la versión jQuery 3.x minificado

2.- Copia el script antes de los scripts generados por la plataforma de aplicaciones en el archivo index.html

3.- En el navegador busca “extensión web server”

### [Web Server for Chrome - Chrome Web Store](https://chrome.google.com/webstore/detail/web-server-for-chrome/ofhbbkphhbklhfoeikjpcbhemlocgigb?hl=Es)

* Botón INICIAR APLICACION

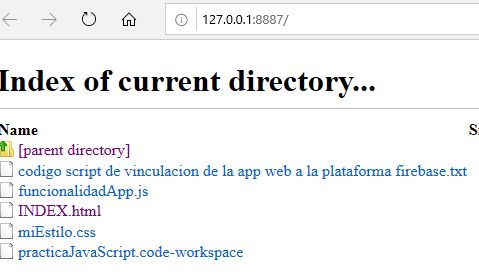


Añadir carpeta PrimeraApp

Es decir seleccionar la carpeta que se desea iniciar como servidor

* Abrir url del servidor: DA CLIC EN EL URL DEL SERVIDOR http://127.0.0.1:8887

http://127.0.0.1:8887/INDEX.html



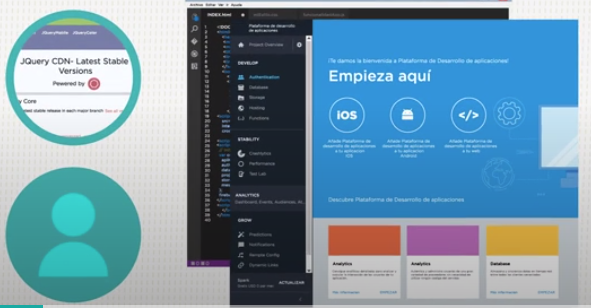
* Por ultimo abrir INDEX.html desde el servidor; y,
* Cambiar 127.0.0.1 a localhost, ya que la plataforma de desarrollo de aplicaciones no reconoce 127.0.0.1

<http://localhost:8887/INDEX.html>

**CODIFICAR EL BACK END DE LA APLICACIÓN**

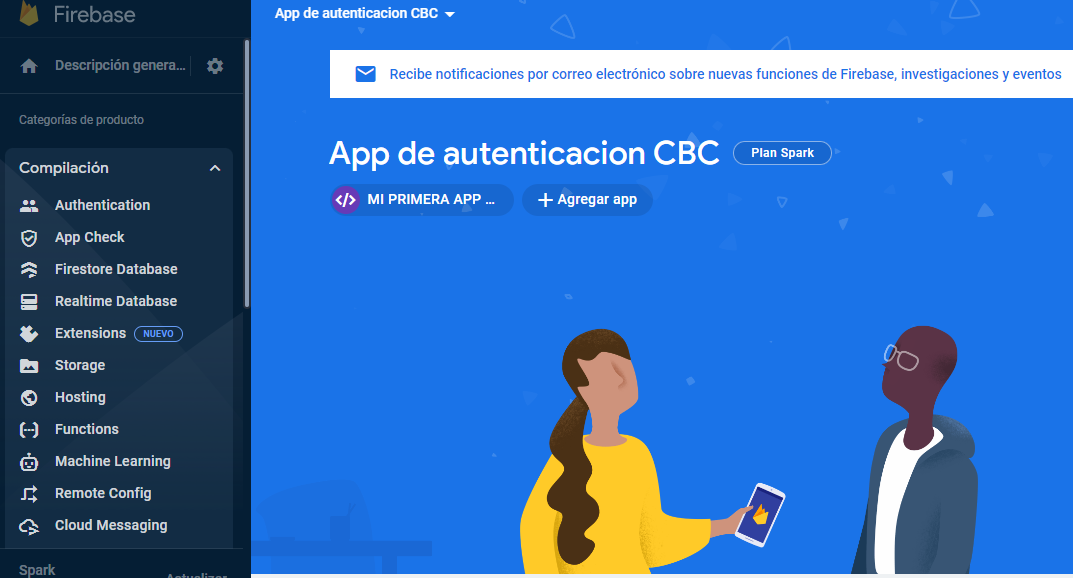
Ya que cuentas con la codificación de la interfaz gráfica, los scripts de la plataforma de desarrollo de aplicaciones, y un servidor local, es momento de programar la funcionalidad de tu aplicación:

En el siguiente ejemplo utilizaras el JQUERY y el servicio de autenticación de usuarios que ofrece la plataforma:

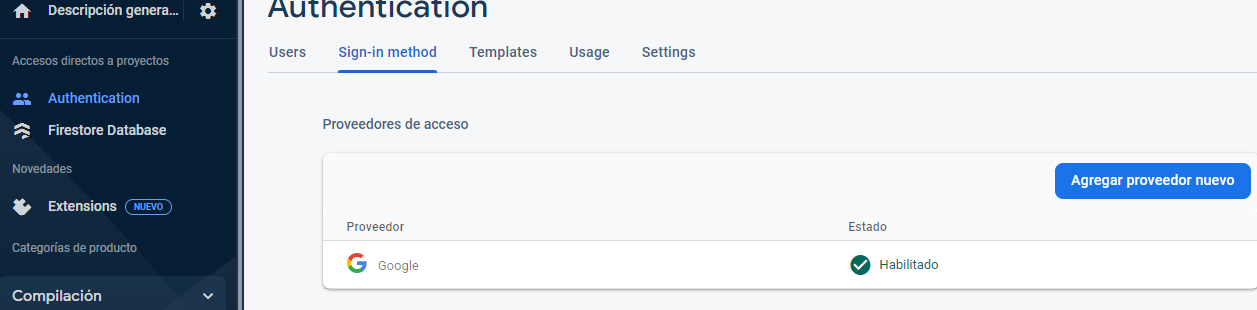


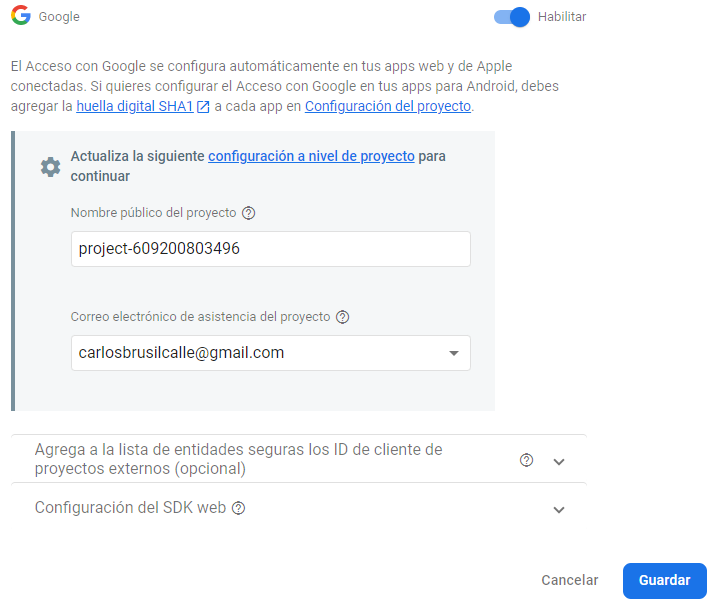
* LOGIN DE LA APLICACIÓN:

Accede a la consola de tu proyecto en la plataforma de desarrollo de aplicación y da clic en la opción AUTENTICACION del menú



Luego: Método de acceso (Sign-in metod), Seleccionar proveedor y GUARDAR





* CREA UN ARCHIVO CON LA EXTENSION JS (editor código VS CODE)

funcionalidaApp.js

* Vincular mediante la etiqueta <script src….> tu index.html al archivo que acabas de crear, la orden tiene que estar antes de cerrar el cuerpo </body>:

<script src=”funcionalidadApp.js”></script>

* CREA UNA VARIABLE QUE APUNTA A TU PROVEEDOR DE APP (en archivo funcionalidad

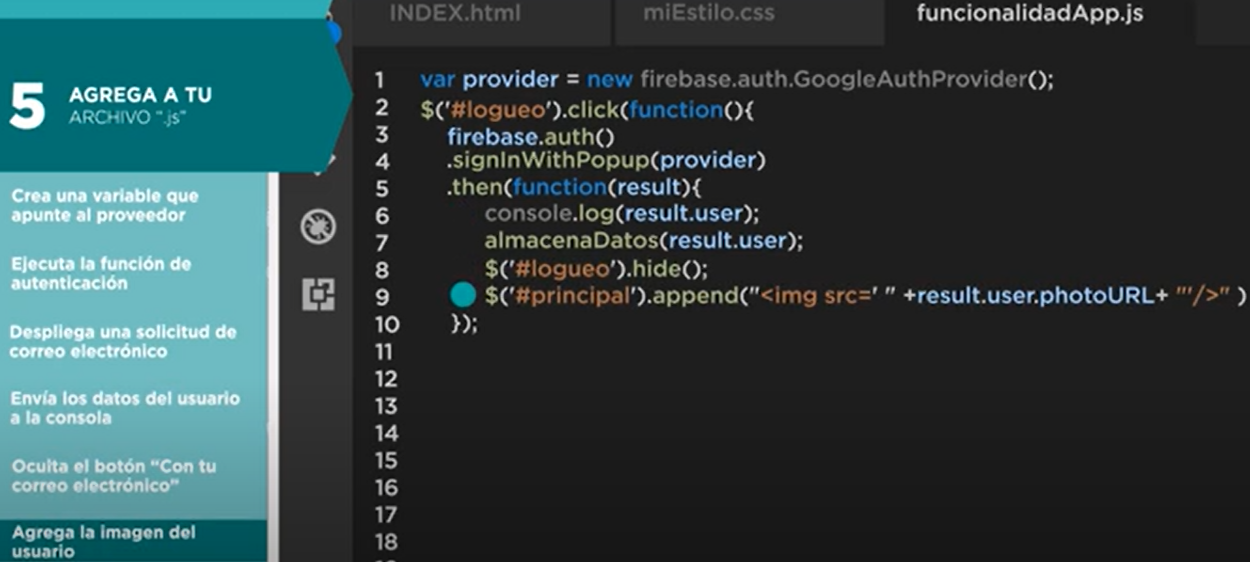
var provider=new firebase.auth.GoogleAuthProvider();

* Ejecuta la funcion de autenticacion desde la Plataforma de desarrollo de app

$(“logueo”.clic(function(){

firebase.auth()

Despliega una ventana donde el proveedor solicita el correo del usuario



.signinWithPopu..

Envia los datos a consola

.console.log(result..

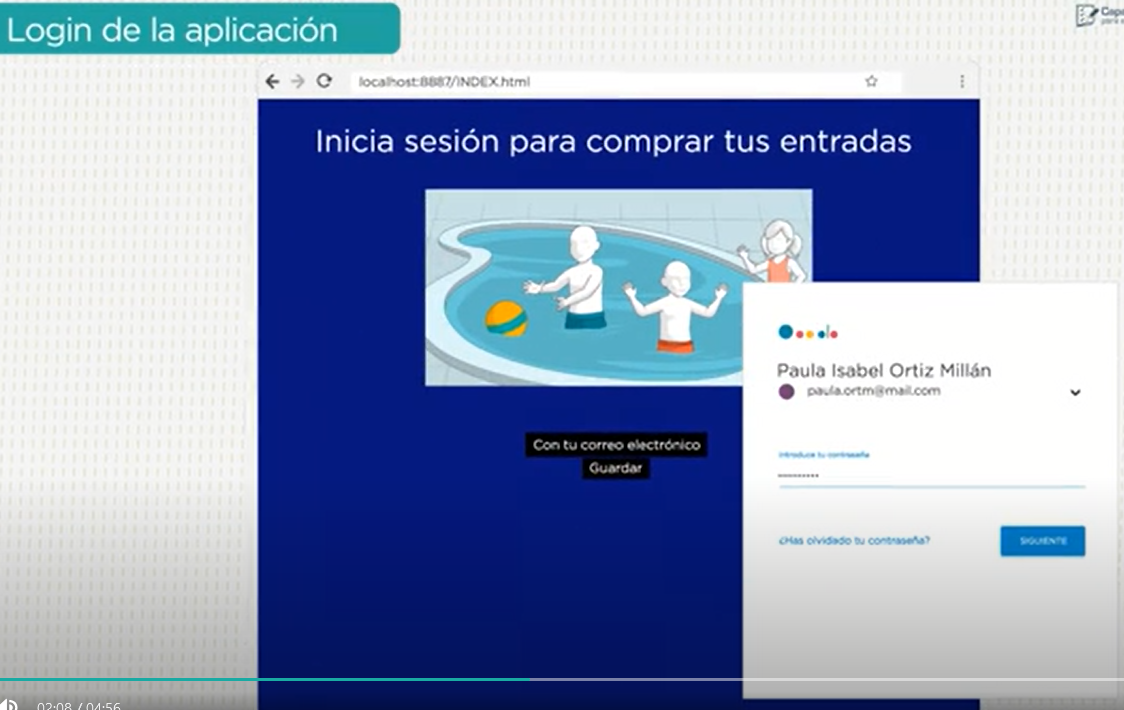
.almacenaDatos(res

Oculta el botón con el método hide()

$(‘#logueo’).hide();

Agregar la imagen del usuario a la division #principal

$(‘#principal’).append(“img.src’”



**ALMACENAR LOS DATOS DEL USUARIO EN LA BASE DE DATOS** que ofrece el proveedor de apps.

Una vez hecho el login es preciso guardar los datos del usuario en el servecio de base de datos en tiempo real:

Accede nuevamente a la consola de tu proyecto en la plataforma de desarrollo de aplicación y da clic en la opción BASE DE DATOS (Firestore Database) del menú

Editar las reglas autorizando a la aplicación pueda leer y escribir

rules\_version = '2';

service cloud.firestore {

match /databases/{database}/documents {

match /{document=\*\*} {

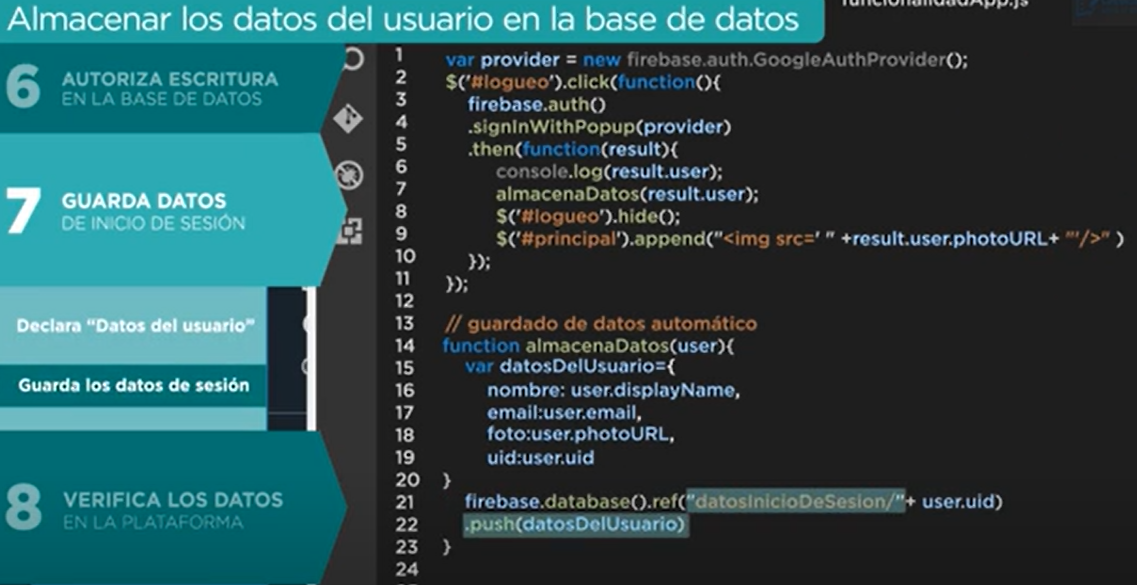
allow read, write: if false;

}

}

}





VERIFICA QUE LOS DATOS DE INICIO DE SESION se encuentren en la base de datos: